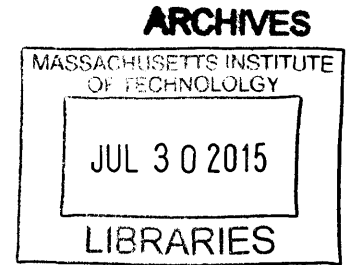# Turbulence Models for the Numerical Prediction of Transitional Flows with RANSE

by

Mert GOKDEPE

B.S., Turkish Naval Academy (2011)

Submitted to the Department of Mechanical Engineering
in partial fulfillment of the requirements for the degrees of

Master of Science in Naval Architecture and Marine Engineering

and

Master of Science in Mechanical Engineering

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

June 2015

© Massachusetts Institute of Technology 2015. All rights reserved.

Author . . . .
Signature redacted
Department of Mechanical Engineering
May 8, 2015

Certified by . .
Signature redacted
Stefano Brizzolara
Research Scientist and Lecturer
Assistant Director for Research MIT Sea Grant
Thesis Supervisor

Accepted by
Signature redacted
David E. Hardt
Chairman, Department Committee on Graduate Students
Department of Mechanical Engineering

Turbulence Models for the Numerical Prediction of Transitional Flows with RANSE

by

Mert GOKDEPE

Submitted to the Department of Mechanical Engineering
on May 8, 2015, in partial fulfillment of the
requirements for the degrees of
Master of Science in Naval Architecture and Marine Engineering
and
Master of Science in Mechanical Engineering

Abstract

Research on turbulence modeling in naval architecture has extensively increased in importance over the years and it is now considered one of the most important ways to accurately compute high Reynolds number flows with Reynolds Averaged Navier-Stokes Equations (RANSE) solvers. In naval architecture, turbulence models are necessary to solve typical hydrodynamic problems both in model scale, where Re=O(106), and in full scale, typically Re=O(108), since direct numerical simulations are not possible in these cases. This thesis aims to study the performance of different turbulence models to predict the laminar-turbulent transitional flow in the boundary layer of streamlined bodies. Starting with a systematic study on a flat plate and arriving to transitional flow airfoils like the NACA 65_1-213 a=0.5. The RANSE solver is built on the libraries of OpenFOAM(Open Field Operation and Manipulation) which is a free, open source CFD program which enables a large group of users to solve broad range of problems. Turbulence models considered range from one equation models such Spalart-Allmaras, two-equation models such as k-epsilon, k-omegaSST, three-equation model kkl-omega as RANS solvers, LES solvers and DES Solvers. The validation of OpenFoam based solver and the different turbulence models is made on the prediction of the friction and pressure drag components as well as lift predictions. In particular, the capability of the turbulent models to capture the transition between laminar and turbulent regime plays a vital role in engineering applications. Four different turbulence models are used in this scope: k-epsilon, k-omegaSST, Spalart-Allmaras and kkl-omega in conjunction with different wall functions. The flat plate case was simulated with all of these turbulence models by using the pimpleFoam transient solver and the hydrofoil case was tested with the kkl-omega and kOmegaSST models by using simpleFoam steady-state solver. The kkl-omega t.m. is one of the newest transition models and it was developed to superior to the other models since it provides the transition region information. Its current implementation in OpenFOAM significantly underestimates the skin friction and the onset of the transition point.

3

We propose a series of modifications which we implemented on model equations and empirical parameters. These changes improve the prediction accuracy of the frictional drag component in transitional flows.

Thesis Supervisor: Stefano Brizzolara
Title: Research Scientist and Lecturer
Assistant Director for Research MIT Sea Grant

Acknowledgments

First and foremost, l would like to thank the Turkish Navy and Turkish Government for the opportunity to attend graduate studies at MIT. I would also like to thank Professor Stefano Brizzolara and Dr. Luca Bonfiglio for their expertise and insight that guided my research. Through their academic mentoring, I discovered the true meaning of research and gained vast experience from their knowledge and dedicated support.

Furthermore, I would also like to show my sincere gratitude to Prof. Aydin Salci, Prof. Ferda Goksin, Prof. Sakir Bal, RDML Nurhan Kayhaoglu, TN (ret) for their instruction during my undergraduate years, without which MIT would not be possible. I also want to thank my predecessor LT Ilkay Ozer Erselcan, TN, who inspired me to apply for graduate studies at MIT. Special thanks goes to the 2N officers CAPT Joseph Harbor, USN, CDR Jerod Ketcham, USN, CDR Weston Gray, USN, CAPT Mark Thomas, USN (ret) and CAPT Raymond "Chip" Mccord, USN (ret) for their guidance and wisdom throughout my academic years at MIT.

Finally, l would like to express my loving appreciation to my parents Mahmut and Gunnur Gokdepe, as well as my brother Murat Gokdepe, for their ardent support throughout my life.

Contents

List of Figures

List of Tables

13

Chapter 1

Introduction to Numerical Methods

A vast amount of research has been made on computational fluid dynamics (CFD). The focal point in computational fluid dynamics is to numerically solve the Navier Stokes equations which include the transport of mass, momentum and energy in the flows. The governing equations of a compressible Newtonian fluid are as follows [24]: Conservation of mass for all fluids:

$$\frac{\partial \rho}{\partial t} + div(\rho u) = 0 \tag{1.1}$$

Conservation of x-momentum:

$$\frac{\partial(\rho u)}{\partial t} + div(\rho u \mathbf{u}) = -\frac{\partial p}{\partial x} + div(\nu\ grad\ u) + S_{Mx} \tag{1.2}$$

Conservation of y-momentum:

$$\frac{\partial(\rho v)}{\partial t} + div(\rho v \mathbf{u}) = -\frac{\partial p}{\partial y} + div(\nu\ grad\ v) + S_{My} \tag{1.3}$$

Conservation of z-momentum:

$$\frac{\partial(\rho w)}{\partial t} + div(\rho w \mathbf{u}) = -\frac{\partial p}{\partial z} + div(\nu\ grad\ w) + S_{Mz} \tag{1.4}$$

15

Conservation of energy:

$$\frac{\partial(\rho i)}{\partial t} + div(\rho i \mathbf{u}) = - - p\, div\, u + div(k\, grad\, T) + \Phi + S_i \qquad (1.5)$$

where S_M is the momentum source and Φ is the dissipation function.

The conservation equations turn into partial differential equations (PDEs) when they are applied to control volumes with an infinitesimal size in a fluid flow as shown above. However, it is almost impossible to apply analytical solutions to these transport equations due to the nonlinear and 3D equations, existence of complex solution domains [10]. It is for this reason that numerical methods are needed at this point and discretization method is used to approximate the PDEs by using the algebraic equations. In discretization method, the equations are solved in domains of limited extension to get results at discrete locations in space and time. In principle, the more discreatizations are qualified, the more accuaracy can be obtained from the numerical methods. A particular version of N-S equations nowadays commonly used for industrial applications is the Reynolds Averaged Navier-Stokes. he main object of study in this thesis which concentrates on turbulence models: specifically their ability of capturing the main flow characteristics near critical laminar-turbulent transition. From a high level of abstraction, the process of a CFD based study follows these steps [23]:

1. Problem statement: information about the flow

2. Mathematical model: Initial Boundary Value Problem(IBVP)=Partial Differential Equations(PDE)+Initial Conditions(IC)+Boundary Conditions(BC)

3. Mesh generation: nodes/cells, time instants

4. Space discretization: coupled ordinary differential equation(ODE)/differential algebraic equation(DAE) systems

5. Time discretization

6. Iterative solver: discrete function values

16

7. CFD software: implementation, debugging

8. Simulation run: parameters, stopping criteria

9. Postprocessing: visualization, analysis of data

10. Verification: model validation, adjustment

Converting the PDEs into algebraic equations thorugh a discretization process using the following algorithm[23]:

1. Mesh generation

   - structured or unstructured, triangular or quadrilateral
   - CAD tools+grid generators
   - mesh size, adaptive refinement in "interesting" flow regions

2. Space discretization (approximation of spatial derivatives)

   - finite differences/volumes/elements
   - high- vs. low-order approximations

3. Time discretization (approximation of temporal derivatives)

   - explicit vs. implicit schemes, stability constraints
   - local time-stepping, adaptive time step control

The studies presented in this thesis are based on a finite volume solver, which again corresponds to the most widely diffused method to address fluid dynamic problems which involve incompressible fluids, such as water in our case, and complex boundary shapes and conditions. It can be very hard and costly to run experiments each time since they require so much equipment and manpower. Therefore, CFD methods are superior to experiments especially when wisely combined with limited sets of experiments may offer possibilities of analysis and interpretation of the fluid dynamic flows that could be hardly met by experiments alone. It can be seen from the comparison in Table 1.1 [23].

Experiments	Simulations
Quantitative description of the flows using measurements	Quantitaive prediction of the flows using CFD software
• for one quantity at a time	• for all desired quantities
• at a limited number of points and time instants	• with high resolution in space and time
• for a laborator-scale model	• for the actual flow domain
• for a limited range of problems and operating conditions	• for virtually any problem and realistic operating conditions
Error sources: measurement errors,flow disturbances by the probes	Error sources: modeling, discretization, iteration, implementation

Table 1.1: Comparison of experiments with simulations

1.1 What is OpenFoam?

The OpenFOAM (Open Field Operation and Manipulation) is a free, open source CFD program which enables a large group of users such as engineers, scientists, academics and commerical organizations to solve broad range of problems including complex fluid flows, solid dynamics and electromagnetics. It includes preprocessing, processing and post-processing libraries and utilities. It has meshing tools such as blockMeshDict and SnappyHexMesh and is also compatible with other CAD engines such as GMSH, Fluent, Ansys. The meshes created with other mesh programs can be converted to an openFoam language by using the utilities including gmshToFoam. The cases in openFoam can be run in parallel processors by using decomposePArDict file and can be integrated by using reconstructPar command. Open source means that the users can use their own files of openFoam by modifying the existing files in the C++ library and it enables the users to compile their own solvers. It follows a highly modular code design in which collections of functionality (e.g. numerical methods, meshing, physical models, etc.) are each compiled into their own shared library. There are quite many applications and utilities in OpenFOAM which can be used for specific engineering problems including meshing, data visualation, pre-processing and

post-processing [2].
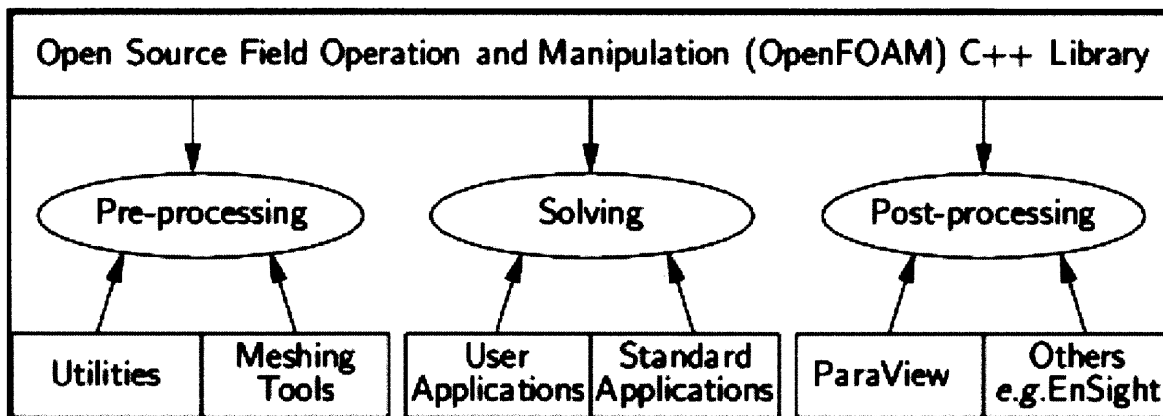
Figure 1-1: General Structure of OpenFoam[2]

1.2   What is GMSH?

Gmsh is a 3D mesh generator which uses finite element method with a CAD engine and post-processor in it. Its goal is to be a fast and user-friendly grid generator and to provide advanced visualization capabilities. Gmsh has four modules: geometry, mesh, solver and post-processing[1]. It is also compatible with openFoam and the meshes that are created in Gmsh can easily be converted to openFoam language.

1.3   Organization

In this research, firstly, the four different turbulence models which were k-epsilon, k-omegaSST, Spallart-Allmaras and kkl-omega are used to estimate the skin friction coefficient on flat plate to be able to find the most proper case setting by comparing the CFD results with experimental results. Then, the goal is to apply those settings to NACA $65_1 - 213$ a=0.5 hydrofoil test case to estimate the total drag, lift, local skin and pressure coefficients along the hydrofoil.

19

1.3.1 Chapter 1

A short description of Numerical Methods, Computational Fluid Dynamics (CFD), OpenFoam, GMSH and organization.

1.3.2 Chapter 2

First and foremost, a theoretical review of the turbulence models including the k-epsilon, Menter k-omegaSST, Spalart-Allmaras, kkl-omega and the derivations of their governing equations are described. Then, the theory of laminar and turbulent boundary layers on a flat plate is investigated. After that, the case set-up for the flat plate was evaluated and model parameters for each model are defined. The study for the difficulties of flat plate is evaluated and the drag coefficient results are compared with the experimental results. For the experimental results, Blasius laminar boundary layer, Schoenherr and ITTC'57 formulas are used.

1.3.3 Chapter 3

NACA 65_1-213 a=0.5 hydrofoil is tested with kkl-omega and Menter k-omegaSST turbulence models with SIMPLE solver and steady- state case in OpenFoam and then, the comparison of the openFoam results with experimental results and XFOIL results is conducted.

1.3.4 Chapter 4

The conclusion remarks of the thesis and some proposal for further studies.

Chapter 2

Introduction to Turbulence Models

"Turbulence is an irregular motion which in general makes its appearance in fluids, gaseous or liquid, when they flow past solid surfaces or even when neighboring streams of the same fluid flow past or over one another" This is the definition of turbulence modeling provided by Taylor and Von Karman in 1937 [9].

Research on turbulence modeling has extensively increased in importance over the years and it is now considered one of the most important aspects to get the accurate computation of high Reynolds number flows with Reynolds Averaged Navier-Stokes Equations (RANSE) solvers. In naval architecture turbulence models are necessary to solve typical hydrodynamic problems both in model scale, where Re=O(106), and in full scale, typically Re=O(108), since direct numerical simulations are not possible in these cases. It is important to study the performance of different turbulence models for the prediction of the laminar-turbulent transitional flow in the boundary layer of streamlined bodies of interest in naval architecture. Turbulence models are considered within a range from one equation models such Spalart-Allmaras, two-equation models such as k-epsilon, k-omegaSST, three-equation model kkl-omega as RANS solvers, LES solvers and DES Solvers. The validation of OpenFoam based solver and the different turbulence models is made on the prediction of the friction and pressure drag components as well as lift predictions and in particular on the capability of the turbulent models to capture the transition between laminar and turbulent regime. In this research, only RANS solvers are used. Before moving forward to the numerical

analysis of turbulence modeling, it is significant to inverstigate the history it. The following history is provided by Ismail Celik [9]:

"The origin of the time-averaged Navier-Stokes equations dates back to the late nineteenth century when Reynolds (1895) published results from his research on turbulence. The earliest attempts at developing a mathematical description of the turbulent stresses, which is the core of the closure problem, were performed by Boussinesq (1877) with the introduction of the eddy viscosity concept. Neither of these authors, however, attempted to solve the time-averaged Navier-Stokes equations in any kind of systematic manner. More information regarding the physics of viscous flow was still required, until Prandtl's discovery of the boundary layer in 1904. Prandtl (1925) later introduced the concept of the mixing-length model, which prescribed an algebraic relation for the turbulent stresses. This early development was the cornerstone for nearly all turbulence modeling efforts for the next twenty years. The mixing length model is now known as an algebraic, or zero-equation model. To develop a more realistic mathematical model of the turbulent stresses, Prandtl (1945) introduced the first one-equation model by proposing that the eddy viscosity depends on the turbulent kinetic energy, k, solving a differential equation to approximate the exact equation for k. This one equation model improved the turbulence predictions by taking into account the effects of flow history. The problem of specifying a turbulence length scale still remained. This info, which can be thought of as a characteristic scale of the turbulent eddies, changes for different flows, and thus is required for a more complete description of the turbulence. A more complete model would be one that can be applied to a given turbulent flow by prescribing boundary and/or initial conditions. Kolmogorov (1942) introduced the first complete turbulence model, by modeling the turbulent kinetic energy k, and introducing a second parameter ω that he referred to as the rate of dissipation of energy per unit volume and time. This two-equation model, termed the k-ω model, used the reciprocal of ω as the turbulence time scale, while the quantity ω 1 2 k served as a turbulence length scale, solving a differential equation for *omega* similar to the solution method for k. Because of the complexity of the mathematics, which required the solution of nonlinear differential equations, it went

virtually without application for many years, before the availability of computers. Rotta (1951) pioneered the use of the Boussinesq approximation in turbulence models to solve for the Reynolds stresses. This approach is called a second-order or second-moment closure. Such models naturally incorporate non-local and history effects, such as streamline curvature and body forces. The previous eddy viscosity models failed to account for such effects. For a threedimensional flow, these second-order closure models introduce seven equations, one for a turbulence length scale, and six for the Reynolds stresses. As with Kolmogorov's k-ω model, the complex nature of this model awaited adequate computer resources.

Thus, by the early 1950's, four main categories of turbulence models had developed:
(1) Algebraic (Zero-Equation) Models
(2) One-Equation Models
(3) Two-Equation Models
(4) Second-Order Closure Models
With increased computer capabilities beginning in the 1960's, further development of all four of these classes of turbulence models has occurred [9]."

RANS equations focus on the mean flow and the effects of turbulence on mean flow properties. It is basically time averaged equations so that the fluctuations in the flow are discarded and averages can be solved directly. The major components of RANS equations are reynolds averaged continuity equations, momentum equations and reynolds averaged stress equations [24].

RANS equations are derived by applying time averages to Navier-Stokes equations. The equations are averaged by inserting [5] $u_i = \overline{u}_i + u'$ and $p = \overline{p} + p'$

Averaged continuity for incompressible flow is given by:

$$\frac{\partial \rho \overline{u}_i}{\partial x_i} = 0 \tag{2.1}$$

Averaged momentum equation for incompressible flow is as follows [24]:

$$\frac{\partial \rho \overline{u}_i}{\partial t} + \frac{\partial (\rho \overline{u}_i \overline{u}_j + \rho \overline{u_i u_j})}{\partial x_j} = -\frac{\partial \overline{p}}{\partial x_i} + \frac{\partial \overline{\tau}_{ij}}{\partial x_j} \tag{2.2}$$

23

where $\bar{\tau}_{ij} = \mu(\frac{\partial \bar{u}_i}{\partial x_j} + \frac{\partial \bar{u}_j}{\partial x_i})$

2.1 Overview of the Turbulence Models

2.1.1 The k-Epsilon model

The k-epsilon model was first proposed by Harlow and Nakayama in 1968 [14] where k is the turbulent kinetic energy and epsilon is the dissipation rate of turbulent kinetic energy k. This model is used for systems that affect the turbulent kinetic energy [24]. The governing equations of the model can be obtained from avreaged Navier-Stokes equations for incompressible flows which was described before and it is as follows [22]:

$$\rho \frac{\partial \bar{u}_i}{\partial t} + \sum_j \left(\rho \bar{u}_j \frac{\partial \bar{u}_i}{\partial x_j} + \rho \overline{\frac{\partial u_i'}{x_j} u_j'} \right) = -\frac{\partial \bar{p}}{\partial x_i} + \sum_j \frac{\partial \bar{\tau}_{ij}}{\partial x_j} \tag{2.3}$$

where u is the velocity field and p is the pressure field. If NS equation is multiplied by u_i and then the resulting equation is averaged, the following formula can be derived:

$$\rho \overline{\frac{\partial u_i}{\partial t} u_i} + \sum_j \overline{u_j \frac{\partial \rho u_i}{\partial x_j} u_i} = -\overline{\frac{\partial p}{\partial x_i} u_i} + \sum_j \overline{\frac{\partial \tau_{ij}}{\partial x_j} u_i} \tag{2.4}$$

When averaged NS equation was multiplied by $\bar{u}_i$, the result is:

$$\rho \frac{\partial \bar{u}_i}{\partial t} \bar{u}_i + \sum_j \left(\rho \bar{u}_j \frac{\partial \bar{u}_i}{\partial x_j} \bar{u}_i + \rho \overline{\frac{\partial u_i'}{x_j} u_j'} \bar{u}_i \right) = -\frac{\partial \bar{p}}{\partial x_i} \bar{u}_i + \sum_j \frac{\partial \bar{\tau}_{ij}}{\partial x_j} \bar{u}_i \tag{2.5}$$

$\tau_{ij} = -\rho \overline{u_i u_j}$, so the formula takes the form of:

$$\rho \frac{\partial \bar{u}_i}{\partial t} + \rho \sum_j \left(\bar{u}_j \frac{\partial \bar{u}_i}{\partial x_j} \bar{u}_i \right) = -\frac{\partial \bar{p}}{\partial x_i} \bar{u}_i + \sum_j \left(\frac{\partial \bar{\tau}_{ij}}{\partial x_j} \bar{u}_i + \frac{\partial \tau_{ij}}{\partial x_j} \bar{u}_i \right) \tag{2.6}$$

When equation 2.6 is subtracted from equation 2.5, the result is given by:

$$\rho \overline{\frac{\partial u_i'}{\partial t} u_i'} + \rho \sum_j \left(\overline{u_j \frac{\partial u_i}{\partial x_j} u_i} - \overline{u_j \frac{\partial \bar{u}_i}{\partial x_j} \bar{u}_i} \right) = -\overline{\frac{\partial p'}{\partial x_i} u_i'} + \sum_j \left(\overline{\frac{\partial \tau_{ij}'}{\partial x_j} u_i'} - \frac{\partial \tau_{ij}}{\partial x_j} \bar{u}_i \right) \tag{2.7}$$

24

When the averaging rules are applied, the result is:

$$\rho\overline{\frac{\partial u_i'}{\partial t}u_i'}+\rho\sum_j\overline{u_j'\frac{\partial u_i'}{\partial x_j}u_i'}+\rho\sum_j\left(\overline{\frac{\partial u_i'}{\partial x_j}u_i'u_j'}+\overline{\frac{\partial u_i'u_j'}{\partial x_j}\overline{u}_i}+\overline{u_j'u_i'\frac{\partial \overline{u}_i'}{\partial x_j}}\right)=-\overline{\frac{\partial p'}{\partial x_i}u_i'}-\sum_j\left(\overline{\frac{\partial u_i'}{\partial x_j}\tau_{ij}'}+\overline{\frac{\partial \tau_{ij}}{\partial x_j}\overline{u}_i}\right)$$

(2.8)

This formula can also be written as:

$$\frac{\rho}{2}\left(\overline{\frac{\partial (u_i')^2}{\partial t}}+\sum_j\overline{\frac{\partial (u_i')^2}{\partial x_j}u_j'}\right)=-\overline{\frac{\partial p'}{\partial x_i}u_i'}-\frac{\rho}{2}\sum_j\frac{\partial}{\partial x_j}\overline{\left(u_i'^2u_j'^2\right)}-\sum_j\left(\overline{\frac{\partial u_i'}{\partial x_j}\tau_{ij}'}+\rho\overline{u_j'u_i'}\frac{\partial \overline{u}_i'}{\partial x_j}\right)$$

(2.9)

The instantaneous kinetic energy k(t) of a turbulent flow is the sum of the mean kinetic energy $K=\frac{1}{2}(U^2+v^2+w^2)$ and the turbulent kinetic energy is $k=\frac{1}{2}(\overline{u'^2}+\overline{v'^2}+\overline{w'^2})$. The governing equations for k can be represented as follows [24]:

$$\frac{\partial (\rho k)}{\partial t}+div(\rho kU)=div(-\overline{p'u'}+2\mu\overline{u's_{ij}'}-\rho\frac{1}{2}\overline{u_i'.u_i'u_j'})-2\mu\overline{s_{ij}'.s_{ij}'}+\rho\overline{u_i'u_j'}.s_{ij}$$ (2.10)

The viscous stress effects on k have two parts: $2\mu U s_{ij}$ is the transport of k because of the viscous stresses and $2\mu s_{ij}.s_{ij}$ is the viscous dissipation of the kinetic energy k. The terms $\rho U\overline{u_i'u_j'}$ and $\rho\overline{u_i'u_j'}.s_{ij}$ consists of Reynolds stresses and the first one the transport of k dues to the Reynolds stresses and the second one is the total decrease of k which occurs because of deformation. In high reynolds numbers, the transport of k and the total decrease of k are quite larger than the viscous parts of the equation [24].

For the numerical analysis in openFoam, three equations are needed to be defined for the k-ϵ model. One of them is k, one is ε and the last one is ν_t. k and ε are used to define velocity scale and length scale but turbulent length scale will be be used to calculate ε. The following formulas are used to define the turbulent length scale:

$$\delta_{0.99}=\frac{0.374l}{Re^{\frac{1}{5}}}$$

(2.11)

25

$$l = 0.4\delta_{0.99} \tag{2.12}$$

where $\delta_{0.99}$ is boundary layer thickness and l is turbulent length scale.

The other parameter that has to be defined is the turbulent intensity which defines the strength of vorticity at the inlet region. It was selected depending of the turbulent intensity that was used in the experiments. The final step is defining the k, ϵ and ν_t values. The following formulas are used to calculate each value for each reynolds number [24]:

$$k = \frac{2}{3}(U_{ref}I)^2 \tag{2.13}$$

$$\varepsilon = C_\mu^{\frac{3}{4}} \frac{k^{\frac{3}{2}}}{l} \tag{2.14}$$

$$\mu_t = \sqrt{\frac{3}{2}} U I l \tag{2.15}$$

where $C_\mu = 0.09$.

The k-ε model needs to integrate the model equations right through the wall but at high reynolds number, it avoids this by making use of the universal behavior of near-wall flows. The k - ε model is one of the most widely used turbulent models but it has advantages and disadvantages depending on which case to be used in [24].

Advantages:

- simplest turbulence model for which only initial and/or boundary conditions need to be supplied

- excellent performace for many industrially relevant flows

- well established, the most widely validated turbulence model

Disadvantages:

- more expensive to implement than mixing length model

- poor performance in some unconfined flows, flows with large extra strains (e.g. swirling flows), rotating flows, flows driven by unisotropy of normal Reynolds stresses

2.1.2 Menter SST k-omega model

k-OmegaSST model is a model to be used in the sublayer of the boundary layer. The difference of this model from the other models is that it does not include damping functions and it is superior wilcox $k - \omega$ model since it is more accuarate. The $k - \varepsilon$ model is independent from the freestream values in the outer region of the boundary layer and Menter [20] used the $k - \varepsilon$ formulation to propose the new model. The governing equation for k-omegaSST model is as follows [24]:

$$\frac{\partial(\rho\omega)}{\partial t}+div(\rho\omega U) = div[(\mu+\frac{\mu_t}{\sigma_{\omega,1}})grad(\omega)]+\gamma_2(2\rho S_{ij}.S_{ij}-\frac{2}{3}\rho\omega\frac{\partial U_i}{\partial x_j}\delta_{ij})-\beta_2\rho\omega^2+2\frac{\rho}{\sigma_{\omega,2}\omega}\frac{\partial k}{\partial x_k}\frac{\partial \omega}{\partial x_k}$$

$$(2.16)$$

The k-ω model also has two equations and eddy viscosity which have to be defined for CFD analysis. The formulas are as follows [20]:

$$k = \frac{2}{3}(U_{ref}I)^2 \qquad (2.17)$$

$$\omega = C_\mu^{-1/4}\frac{\sqrt{k}}{l} \qquad (2.18)$$

$$\nu_t = \frac{k}{\omega} \qquad (2.19)$$

Model constants are same as the ones provided in k-ε model. These equations are used for free stream. Considering the wall conditions, the following formulas are proposed [20]:

$$k_{wall} = 0 \qquad (2.20)$$

27

$$\omega_{wall} = 10\frac{6\nu}{\beta_1(\Delta y)^2} \tag{2.21}$$

where β_1=0.075 and Δy is the distance to the center of the first cell.

2.1.3   Spalart-Allmaras Model

The Spalart-Allmaras model is one of the one equation models and it includes only one transport equation for kinematic eddy viscosity parameter $\tilde{\nu}$. The Spalart-Almaras model provides promising results for external aerodynamics [24]. There has been made modifications on Spalart-Allmaras model but the baseline model will be discussed in this research. The transport equation and Reynolds stresses are as follows [7]:

$$\frac{D\tilde{\nu}}{Dt} = P - D + T + \frac{1}{\sigma}\left[\nabla.((\nu + \tilde{\nu})\Delta\tilde{\nu}) + c_{b2}(\nabla\tilde{n}u)^2\right] \tag{2.22}$$

where P is the production term, D is the wall destruction term and T is the trip term and they are given by:

$$P = c_{b1}(1 - f_{t2})\tilde{S}\tilde{\nu} \tag{2.23}$$

$$D = \left(c_{w1}f_w - \frac{c_{b1}}{\kappa^2 d^2}f_{t2}\right)\left[\frac{\tilde{\nu}}{d}\right]^2 \tag{2.24}$$

$$T = f_{t1}(\Delta u)^2 \tag{2.25}$$

where $\tilde{S}$ is the modified vorticity and it is given by:

$$\tilde{S} \equiv S + \frac{\tilde{\nu}}{\kappa^2 d^2}f_{v2} \tag{2.26}$$

$$f_{v1} = 1 - \frac{\chi}{1 + \chi f_{v1}} \tag{2.27}$$

d represents the wall spacing of the first cell from the wall and S represents the

28

vorticity.

$$f_w = g \left[\frac{1 + c_{w3}^3}{g^6 + c_{w3}^3} \right]^{1/6} \tag{2.28}$$

$$g = r + c_{w2}(r^6 - r) \tag{2.29}$$

$$r = min \left(\frac{\tilde{\nu}}{\tilde{S}\kappa^2 d^2}, r_{lim} \right) \tag{2.30}$$

$$f_{t1} = c_{t1} g_t exp \left(-c_{t2} \frac{\omega_t}{\Delta u^2} [d^2 + g_t^2 d_t^2] \right) \tag{2.31}$$

$$f_{t2} = c_{t3} exp(-c_{t4}\chi^2) \tag{2.32}$$

where $g_t = min \left(0.1, \frac{\Delta}{\omega_t \Delta x} \right)$, d_t is the legth between the wall and the Δu is the relative velocity difference in reference to the trip point, Δx is streamwise cell distance at the trip and ω_t is the trip vorticity. f_{v1}, f_{v2} and f_w are wall damping functions. The model constants are listed in Table 2.1.

σ_v	0.667
κ	0.71
C_{b1}	0.1355
C_{b2}	0.622
C_{w1}	$C_{b1}\kappa^2 \left(\frac{1+C_{b2}}{\sigma_v} \right)$
C_{w2}	0.3
C_{w3}	2
C_{t1}	1
C_{t2}	2
C_{t3}	1.2
C_{t4}	0.5
r_{lim}	10

Table 2.1: The Spalart-Allmaras model constants

In Spalart-Allmaras model, two new variables are defined for the buffer layer and viscous sublayer. They are $\tilde{\nu}$ and $\chi \equiv \frac{\tilde{\nu}}{\nu}$. $\tilde{\nu}$ was chosen instead of U as a transported

29

quantity because $\tilde{\nu}$ operates better in near-wall region and it is easier to determine compared to U. Therefore, a finer mesh is not required for Spalart-Allmaras model in contrast to k-omega and k-epsilon models. The eddy viscosity ν_t equals kyu_t in the log layer but it does not in the buffer layer. Due this reason, Spalart-Allmaras proposed the variable $\tilde{\nu}$. Therefore, Spalart-Allmaras introduced the following equations [17]:

$$\nu_t = \tilde{\nu} f_{v1} \tag{2.33}$$

$$f_{v1} = \frac{\chi^3}{\chi^3 + c_{v1}{}^3} \tag{2.34}$$

where $\chi = \frac{\tilde{\nu}}{\nu}$ and ν is the molecular viscosity.

2.1.4 The kkl-omega Model

Overview and development of the model

The calculation of the laminar boundary layers are quite easy and direct since turbulence effects are not taken into account for the calculations. However, it is not very straightforward to compute the turbulent boundary layers due to the existing unsteady disturbances but the developments in numerical methods such as the ones in Reynolds averaging provided considerable capabilities about prediction of the behaviors of the fully turbulent flows. There is also a transition region between laminar and fully turbulent which is not quite easy to deal with because disturbances which cause instability began to occur and they develop as the Reynolds number increases. In some cases, the boundary layer might be laminar on major part of the wall and fully turbulent approaches might lead to wrong computations. It is at umost importance to corretly predict the onset of the transition since it has a big effect on engineering applications [25]. The effects of transition can be listed as follows [11]:

- **Wall Shear Stress**

  - Higher wall shear for turbulent flows (more resistance in pipe flow, higher drag for airfoils,...)

- **Heat Transfer**

  - Heat transfer is strongly dependant on state of boundary layer

  - Much higher heat transfer in turbulent boundary layer

- **Separation Behavior**

  - Separation point/line can change drastically between laminar and turbulent flows

  - Turbulent flow much more robust than laminar flow. Stays attached even at larger pressure gradients
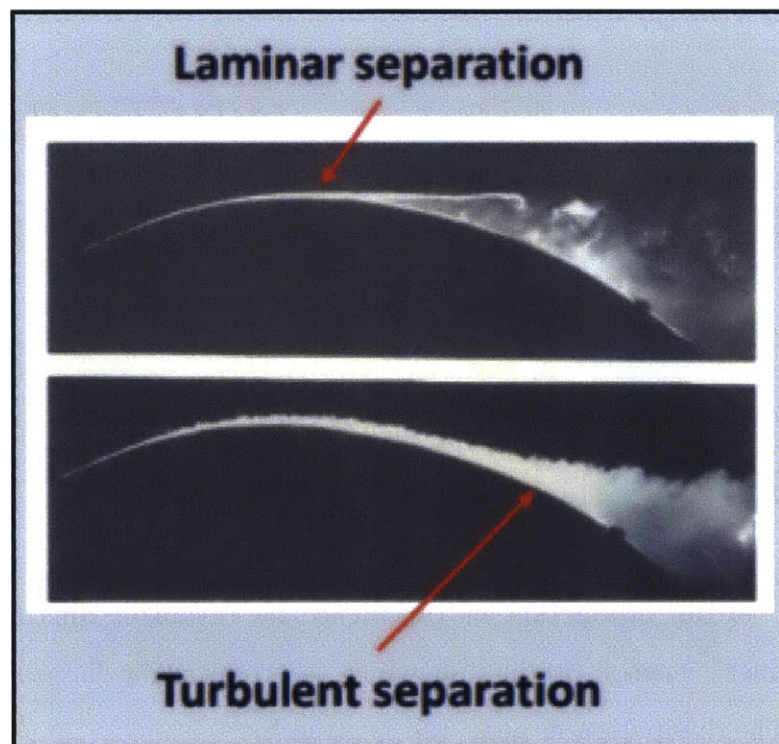
Figure 2-1: Laminar and Turbulent Separation [11]

- **Efficiency**

  - Axial turbo machines perform different in laminar and turbulent stage

  - Wind turbines have different characteristics

  - Small scale devices change characteristics depending on flow regime

31

Modes of Transition

The transition is divided into three categories which are natural transition, bypass transition and separated flow transition. [25]

Derivation of Model Equations

This model is a three equation, eddy-viscosity type. In addiion to the variables such as U and P, the turbulent kinetic energy k_t, the laminar kinetic energy k_l and inverse turbulent time scale ω are used to solve the transport equations. The transport equations are defined by [25]:

$$\frac{Dk_t}{Dt} = P_{K_t} + R_{BP} + R_{NAT} - \omega k_t - D_T + \frac{\partial}{\partial x_j}\left[\left(\nu + \frac{\alpha_T}{\alpha_K}\right)\frac{\partial k_t}{\partial x_j}\right] \qquad (2.35)$$

$$\frac{Dk_L}{Dt} = P_{K_L} - R_{BP} - R_{NAT} - D_L + \frac{\partial}{\partial x_j}\left[\nu\frac{\partial k_L}{\partial x_j}\right] \qquad (2.36)$$

$$\frac{D\omega}{Dt} = C_{\omega 1}\frac{\omega}{k_T}P_{k_t} + \left(\frac{C_{\omega R}}{f_W} - 1\right)\frac{\omega}{k_T}\left(R_{BP} + R_{NAT}\right) - C_{\omega 2}\omega^2 + C_{\omega 3}f_\omega\alpha_T\frac{\sqrt{k_T}}{d^3} + \frac{\partial}{\partial x_j}\left[\left(\nu + \frac{\alpha_T}{\alpha_\omega}\right)\frac{\partial\omega}{\partial x_j}\right]$$
$$(2.37)$$

where P_{K_T} is the production of turbulence by turbulent fluctuations, P_{K_L} is the laminar kinetic energy production which is produced by large scale turbulent fluctuations. The dissipation in near wall is given by:

$$D_T = \nu\frac{\partial\sqrt{k_t}}{\partial x_j}\frac{\partial\sqrt{k_T}}{\partial x_j} \qquad (2.38)$$

$$D_L = \nu\frac{\partial\sqrt{k_L}}{\partial x_j}\frac{\partial\sqrt{k_L}}{\partial x_j} \qquad (2.39)$$

R_{BP} is the averaged effect of the streamwise fluctuations breakdown into turbulence during bypass transition and it is given by:

32

$$R_{BP} = C_R \beta_{BP} k_L \omega / f_W \qquad (2.40)$$

where β_{BP} is the threshold function and it controls the bypass transition. It is given by:

$$\beta_{BP} = 1 - exp\left(-\frac{\phi_{BP}}{A_{BP}}\right) \qquad (2.41)$$

$$\phi_{BP} = max\left[\left(\frac{k_T}{\nu\Omega} - C_{BP,crit}\right), 0\right] \qquad (2.42)$$

R_{NAT} is the natural production term and it is defined by:

$$R_{NAT} = C_{R,NAT} \beta_{NAT} k_L \Omega \qquad (2.43)$$

$$\beta_{NAT} = 1 - exp\left(-\frac{\phi_{NAT}}{A_{NAT}}\right) \qquad (2.44)$$

The damping function f_ω is defined by

$$f_\omega = 1 - exp\left[0.41\left(\frac{\lambda_{eff}}{\lambda_T}\right)^4\right] \qquad (2.45)$$

The production of turbulence P_{K_T} is given by:

$$P_{K_T} = \nu_{T,s} S^2 \qquad (2.46)$$

where $\nu_{T,s}$ is small scale turbulent viscosity and it is defined by:

$$\nu_{T,s} = f_v f_{INT} C_\mu \sqrt{k_{T,s}} \lambda_{eff} \qquad (2.47)$$

where f_v and f_{INT} are damping functions.

$$C_\mu = \frac{1}{A_0 + A_S(S/\omega)} \qquad (2.48)$$

33

and

$$f_v = 1 - exp\left(-\frac{\sqrt{Re_T}}{A_v}\right) \tag{2.49}$$

where Re_T is the turbulent Reynolds number which is defined by:

$$Re_T = \frac{f_W{}^2 k_T}{\nu\omega} \tag{2.50}$$

f_{INT} is the damping function which is caused by intermittency and it is given by:

$$f_{INT} = min\left(\frac{k_L}{C_{INT}k_{TOT}}, 1\right) \tag{2.51}$$

P_{k_L} is the laminar kinetic energy production and it is caused by large scale turbulent fluctuations. It is given by:

$$P_{k_L} = \nu_{T,l}S^2 \tag{2.52}$$

where $\nu_{T,l}$ is the large-scale turbulent viscosity and it is defined as follows:

$$\nu_{T,l} = min\left(f_{\tau,l}C_{11}\left(\frac{\Omega\lambda_{eff}{}^2}{\nu}\right)\sqrt{k_{T,l}}\lambda_{eff} + \beta_{TS}C_{12}Re_\Omega d^2\Omega, \frac{0.5(k_L + k_{t,l})}{S}\right) \tag{2.53}$$

where $Re_\Omega = \frac{d^2\Omega}{\nu}$

$f_{\tau,l}$ is the time-sacale-based damping function and it is given by:

$$f_{\tau,l} = 1 - exp\left[-C_{\tau,l}\frac{k_{T,l}}{\lambda_{eff}{}^2\Omega^2}\right] \tag{2.54}$$

and

$$\beta_{TS} = 1 - exp\left(-\frac{max(Re_\Omega - C_{TS,crit}, 0)^2}{A_{TS}}\right) \tag{2.55}$$

$$\phi_{NAT} = max\left[(Re_\Omega - C_{NAT,crit}/f_{NAT,crit}), 0\right] \tag{2.56}$$

34

The sum of the large scale eddy viscosity and small scale eddy viscosity is equal to total eddy viscosity.

$$\nu_{TOT} = \nu_{T,s} + \nu_{T,l} \tag{2.57}$$

$$\alpha_{\theta,TOT} = f_W \left(\frac{k_t}{k_{TOT}} \right) \frac{\nu_{T,s}}{Pr_\theta} + (1 - f_W) C_{\alpha,\theta} \sqrt{k_T} \lambda_{eff} \tag{2.58}$$

the turbulent scalar diffusivitiy is given by:

$$\alpha_T = f_v C_{\mu,std} \sqrt{k_{T,s}} \lambda_{eff} \tag{2.59}$$

and the total kinetic energy is given by:

$$k_{TOT} = k_T + k_L \tag{2.60}$$

The contribution of laminar and turbulent fluctuations to the mean flow and eddy thermal diffusivity is as shown below:

$$-\overline{u_i u_j} = \nu_{TOT} \left(\frac{\partial U_i}{\partial x_j} + \frac{\partial U_j}{\partial x_i} \right) - \frac{2}{3} k_{TOT} \delta_{ij} \tag{2.61}$$

$$-\overline{u_i \theta} = \alpha_{\theta,TOT} \frac{\partial \theta}{\partial x_i} \tag{2.62}$$

The turbulent kinetic energy k_T is divided into two parts in the region close to the wall. The first one is small scale energy $k_{T,s}$ and it has effects on the production of turbulence. The second one is large scale energy $k_{T,l}$ and it has effects on the laminar kinetic energy production. The effective length scale λ_{eff} and the turbulent length scale λ_T is defined as shown below:

$$\lambda_{eff} = min \left(C_\lambda d, \lambda_T \right) \tag{2.63}$$

$$\lambda_T = \frac{\sqrt{k_T}}{\omega} \tag{2.64}$$

35

The small scale and large scale energies are given by:

$$k_{T,s} = f_{SS} f_W k_T \tag{2.65}$$

$$f_W = \frac{\lambda_{eff}}{\lambda_T} \tag{2.66}$$

$$f_{SS} = exp\left[-\left(\frac{C_{SS}\nu\Omega}{k_T}\right)^2\right] \tag{2.67}$$

Model Constants

A_0	4.04	A_s	2.12	σ_ω	1.17
A_ν	6.75	A_{BP}	0.6	A_{NAT}	200
A_{TS}	200	$C_{BP,crit}$	1.2	C_{NC}	0.1
$C_{NAT,crit}$	1250	C_{INT}	0.75	$C_{TS,crit}$	1000
$C_{R,NAT}$	0.02	C_{11}	3.4e-6	C_{12}	1e-10
C_R	0.12	$C_{\alpha,\theta}$	0.035	C_{SS}	1.5
$C_{\tau,l}$	4360	$C_{\omega,1}$	0.44	$C_{\omega,2}$	0.92
$C_{\omega,3}$	0.3	$C_{\omega,R}$	1.5	C_λ	2.495
$C_{\mu,std}$	0.09	Pr_θ	0.85	σ_k	1

Table 2.2: The kkl-omega model constants

2.2 Flat Plate Theory

In this research, firstly, flat plate case will be simulated to understand the difference between each turbulence model and how accurately they can predict the friction coefficients since it is the simplest case to test. Before moving forward to the case runs, it is important to investigate the laminar and turbulent boundary layers on a flat plate.

2.2.1 Blasius' Laminar Boundary Layer

Blasius Laminar Boundar layer theorem was used to compare the the friction coefficient results obtained from openFoam. Therefore, the first step to derive the friction frag formula and investigate the basic aspects of laminar flow. Blasius laminar boundary layer forms over a steady flow past 2D flat plate at Reynolds numbers less than 5e5. The derivation of the blasius laminar boundary layer with the assumptions of 2D steady flow over a plate, horizontal velocity is equal to the undisturbed velocity, vertical velocity is 0 and $\frac{dp}{dx} = 0$ is as follows [6]:

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0 \qquad (2.68)$$

$$u\frac{\partial u}{\partial x} + v\frac{\partial u}{\partial y} = v\frac{\partial^2 u}{\partial y^2} \qquad (2.69)$$

The next step is to define the boundary conditions to be able to solve the equations. No slip boundary condition on the wall so that $u = v = 0$ on the wall. Vertical velocity $v = V = 0$ and horizontal velocity is equal to the undisturbed farstream velocity $U = U_0$ outside the boundary layer.
Using the similarity parameters, the following solution is given

$$\frac{u}{U} \; ; \; \eta \equiv y\sqrt{\frac{U_0}{vx}} \Leftrightarrow \frac{y}{x} = \frac{\eta}{\sqrt{Re_x}} \Leftrightarrow y = \eta\sqrt{\frac{vx}{U_0}} \text{ so that } \frac{u(x,y)}{U_0} = F(\eta)$$

The next step is to define the properties of Blasius Laminar Boundary Layer. δ^* is the displacement thickness of which the formula is: They are expressed as:

$$\delta^* \equiv \int_0 \infty \left(1 - \frac{u}{U}\right) dy \qquad (2.70)$$

θ is the momentum thickness of which the formula is expressed as:

$$\theta \equiv \int_0 \infty \frac{u}{U} \left(1 - \frac{u}{U}\right) dy \qquad (2.71)$$

$$\frac{u(x,y)}{U_0} = F(\eta); \eta = y\sqrt{\frac{U_0}{\nu x}}; y \equiv \eta\sqrt{\frac{\nu x}{U_0}}; \frac{y}{x} = \frac{\eta}{\sqrt{Re_x}} \qquad (2.72)$$

$$\delta \equiv \frac{\nu x}{U_0} \qquad (2.73)$$

$$\delta_{0.99} \cong 4.9\sqrt{\frac{\nu x}{U_0}} \qquad (2.74)$$

$$\delta^* \cong 1.72\sqrt{\frac{\nu x}{U_0}} \qquad (2.75)$$

$$\theta \cong 0.064\sqrt{\frac{\nu x}{U_0}} \qquad (2.76)$$

where $\delta \propto \sqrt{x}$, $\delta \propto \frac{1}{\sqrt{Re_x}}$ and $\frac{\delta}{x} \propto \sqrt{\frac{\nu}{U_0 x}} = \frac{1}{\sqrt{Re_x}}$

$$\tau_0 \equiv \tau_w \cong 0.332\rho U_0^2 \left(\frac{U_0 x}{\nu}\right)^{-\frac{1}{2}} = 0.332(\rho u_0^2)Re_x^{-1/2} \qquad (2.77)$$

where $\tau_0 \propto \frac{1}{\sqrt{Re_x}}$ and $\tau_0 \propto U_0^{3/2}$

The total drag D on a plate is given by:

$$D = B\int_0^L \tau_0 dx \cong 0.664(\rho U_0^2)(BL)\left(\frac{U_0 L}{\nu}\right)^{-1/2} \qquad (2.78)$$

where $D \propto \sqrt{L}, \quad D \propto U^{3/2}$

The friction coefficient is as follows:

$$C_f = \frac{D}{\frac{1}{2}(\rho U_0^2)(BL)} \cong \frac{1.328}{\sqrt{Re_L}} \qquad (2.79)$$

where $C_f \propto \frac{1}{\sqrt{L}}$ and $C_f \propto \frac{1}{\sqrt{U}}$

2.2.2   Turbulent Boundary Layer on a Flat Plate

Turbulent boundary layers are more challenging to be modeled compared to laminar boundary layers since there are disturbance effects in turbulent flow. There
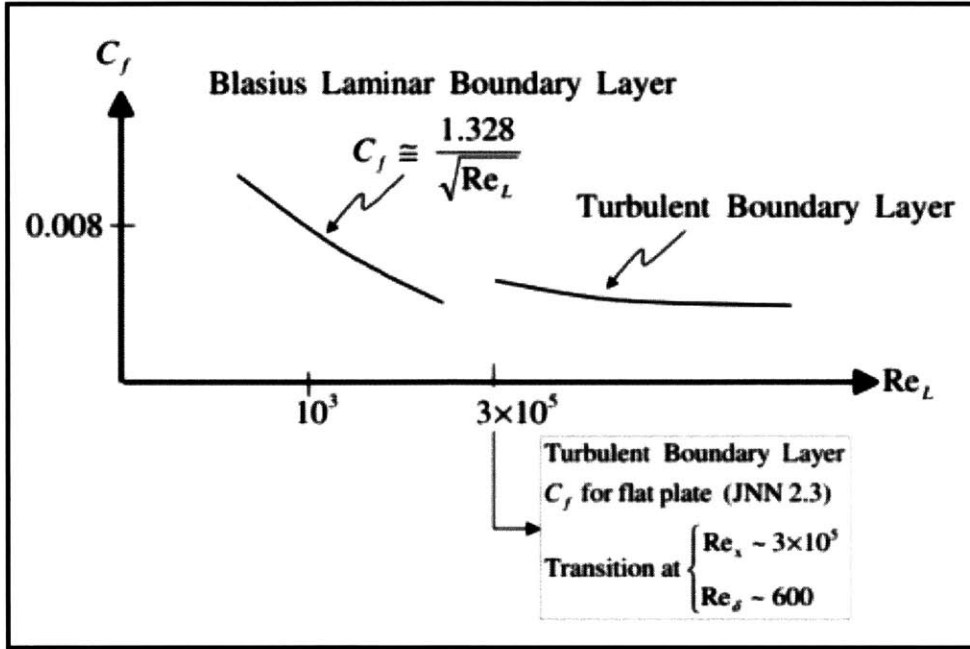
Figure 2-2: Skin Friction vs Re [6]

are several ways of calculating the friction coefficient for the turbulent case depending on the approximate velocity profile used. There are different emprical velocity profiles avaliable such as $(1/7)^{th}$ power velocity profile law or logarithmic velocity profile law. If the $(1/7)^{th}$ power velocity profile law is used, the skin friction on a flat plate can be derived as follows [6]:

$$\frac{\overline{u}}{U_0} = \left(\frac{y}{\delta}\right)^{1/7}$$ (2.80)

where δ is the boundary layer thickness and $\overline{u}$ is the averaged velocity.

$$\delta^* = \frac{\delta}{8}$$ (2.81)

$$\theta = \frac{7}{72}\delta \cong 0.0972\delta$$ (2.82)

The friction law of Blasius for pipe flows is expressed as:

$$\frac{\tau_0}{\rho U_0^2} = 0.227 \left(\frac{U_0\delta}{\nu}\right)^{-1/4}$$ (2.83)

39

By using the same assumptions which were used in laminar flow such as the horizontal velocity U(x) is equal to the undisturbed velocity U_0 and $\frac{dp}{dx} = 0$, the following formula can be obtained where δ^*, θ, τ_0, U_0 are substituted in Von Karman's moment equation.

$$\frac{\tau_0}{\rho U_0^2} = \frac{d}{dx}(\theta) \Rightarrow 0.0227 \left(\frac{U_0 \delta}{\nu}\right)^{-1/4} = \frac{7}{24}\frac{d\delta}{dx} \tag{2.84}$$

Assuming that the turbulent boundary layer starts at $x = 0$, the formula takes the form of:

$$\delta(x) \cong 0.373x \left(\frac{U_0 x}{\nu}\right)^{-1/5} \rightarrow \frac{\delta}{x} \cong 0.373 Re_x^{-1/5} \tag{2.85}$$

So, the total drag D for the flat plate is:

$$D = 0.036(\rho U_0^2)BLRe_L^{-1/5} \tag{2.86}$$

$$C_f = \frac{D}{\frac{1}{2}\rho U_0^2 BL} = 0.073 Re_L^{-1/5} \tag{2.87}$$

If the logarithmic velocity profile law used, Schoenherr formula can be obtained which will be used for comparison of the results in this thesis. The following equations are used for logarithmic velocity profile law [21]:

$$\frac{u}{u_\tau} = 2.5 log\left(\frac{u_\tau y}{\nu}\right) + 5.1 \quad u_\tau y/\nu > 30 \tag{2.88}$$

$$\frac{U - u}{u_\tau} = -2.5 log\left(\frac{y}{\delta}\right) + 2.35 \quad u_\tau y/\delta < 0.15 \tag{2.89}$$

When these two equations are summed, the equation for the friction vlocity u_τ can be derived and it is as follows:

$$\frac{U}{u_\tau} = Alog\frac{u_\tau \delta}{\nu} + (C_1 + C_2) \tag{2.90}$$

The other equation to be used is Von Karman Momentum equation and it is given

by:

$$\frac{\tau_{xy}}{\rho} = \frac{d}{dx}(U^2\theta) + U\delta^*\frac{dU}{dx} \tag{2.91}$$

Assuming that the isotropic relation is satisfied such as $\overline{u'^2} = \overline{v'^2}$ and the pressure radient term is discarded, using the relation $u_\tau \equiv [\tau_{xy}(x,0)/\rho]^{1/2} \equiv (\tau_o\rho)^{1/2}$, the following eqaution can be derived:

$$\tau_0 = \rho U^2 d\theta/dx = \rho u_\tau^2 \tag{2.92}$$

Using the definitions of momentum thickness and boundary layer thickness, the result is:

$$\delta^*/\delta = I(u_{t=\tau}/U) \tag{2.93}$$

$$\theta/\delta = I(u_{t=\tau}/U) - J(u_{t=\tau}/U))^2 \tag{2.94}$$

where

$$I = \int_0^1 f_2(y/\delta)d(y/\delta) \simeq \int_0^1 \left(\frac{U-u}{u_\tau}\right) d(y/\delta) \tag{2.95}$$

$$I = \int_0^1 [f_2(y/\delta)]^2 d(y/\delta) \simeq \int_0^1 \left(\frac{U-u}{u_\tau}\right)^2 d(y/\delta) \tag{2.96}$$

The final local frictional-drag coefficient can be derived by using these equations and it is defined as $c_f = \frac{\tau_0}{\frac{1}{2}\rho U^2}$. The following equation can be satisfied by the local friction coefficient.

$$\frac{1}{\sqrt{c_f}} = 2^{-1/2}Alog(R_x c_f) + C_3 \tag{2.97}$$

The total drag coefficient can be derived by integrating the locl drag coeffcieint along the plate.

41

$$C_f = \frac{1}{l} \int_0^l c_f(x)dx \tag{2.98}$$

This is the the formula that Schoenherr flat plate friction coefficient formula is based on and the Scheonherr formula is:

$$\frac{1}{\sqrt{c_f}} = 1.79 log(R_l C_f) = 4.13 log_{10}(R_l C_f) \tag{2.99}$$

2.3 Turbulence Modeling in OpenFoam

The analysis consists of pre-processing which includes mesh generation, calculating the initial freestream values and setting up the cases, processing which is running the cases in openFoam and post-processing which is the analysis of the results. The steps of modellig a case in openFoam can be seen in Figure 2-3. The test cases are evaluated with the turbulence models discussed above to determine the best model for the cases.

2.3.1 OpenFoam Structure

The general structure of a case in openFoam consists of three main directories which are system, constant and 0 (or time). Each directory has minimum set of files which can be seen in Figure 2-4. Constant directory defines the case and it contains files such as transportProperties, turbulenceProperties and RASProperites for RAS model. In RASProperties file, the type of the model and inclusion of the turbulence effects, in transportProperties file, the type of the transport model and values of the flow properties such as ν and *rho* and in turbulenceProperties file the simulation type are defined. Constant directory also includes the polyMesh file which includes mesh properties such as boundary conditions. The system directory includes the main files such controlDict, fvSchemes and fvSolution. If the case will be run with more than 1 processor, decomposePArdict file should also be put in system directory to determine the number of the subdomains to be used. System directory is mainly about the solution procedure. The controlDict files defines the starting and ending times,

42
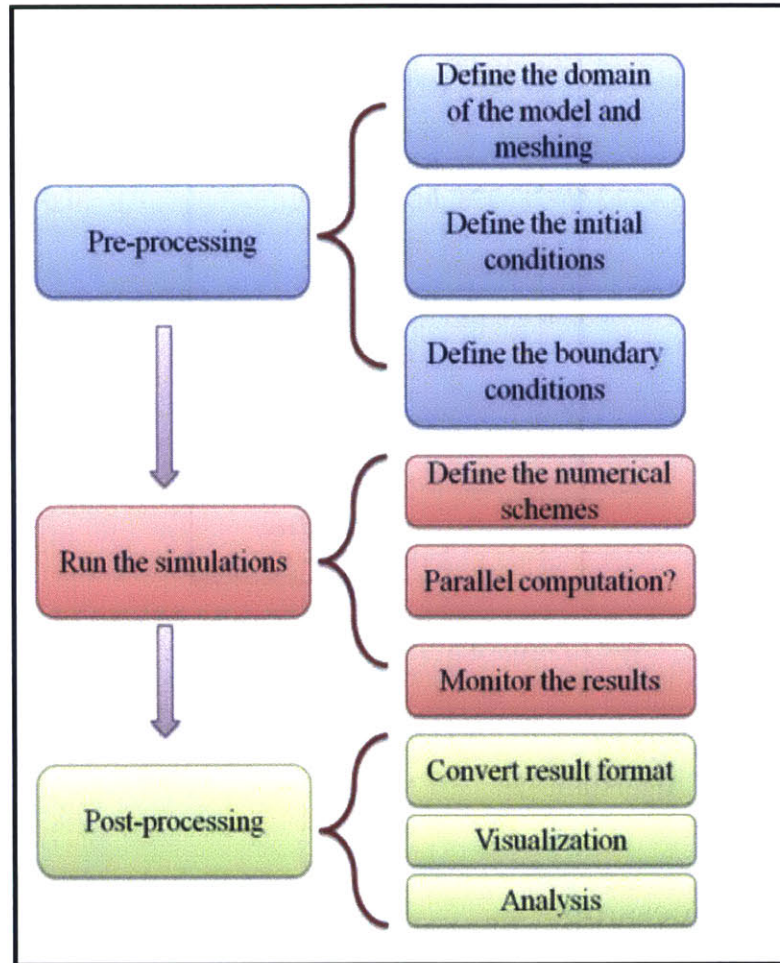
Figure 2-3: Modeling steps in openFoam [18]

tmestep of the simulation and the output of the simulation such as forces. fvSolution file includes the solver parameters, tolerances and solution algorithm. fvSchemes is the file where discretization schemes are set for the case. The type of the files in 0 directory depends on the model to be used. It contains the model parameters, in example, omega, nut, nuTilda, epsilon k etc. The files define the initial farstream values and boundary conditions for the case [2].

Solvers

For this thesis, pimpleFoam solver is used for the flat plate case and simpleFoam solver is used for the hydrofoil case. SimpleFoam is a steady-state solver for the incompressible, turbulent flows whereas pimpleFoam is a large time-step transient

43
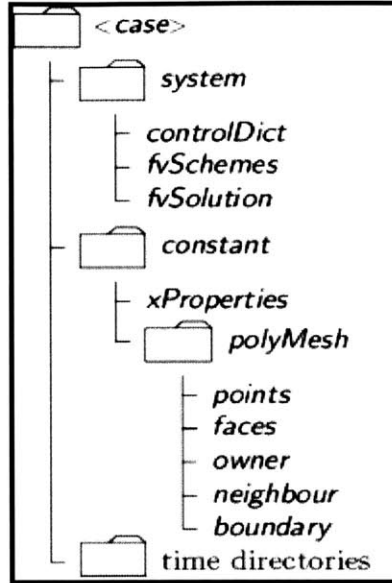
Figure 2-4: OpenFoam case directory [2]

solver for incompressible flows and it is a combination of PISO and SIMPLE solvers. It uses the PIMPLE algortihm [2].

2.3.2 Pre-Processing

Law of the Wall and Mesh Generation

Near the wall, the fluid behavior and the turbulence structure are very different from the ones in freestream. The major problem in near-wall region is the inverted energy cascade. Small vortices start to occur from the wall and they turn into bigger vortices by the time. This is not considered in standard modeling approach. The point of interest in simulations is the drag instead of dealing with the occurances in near-wall region. Therefore, wall functions are used to skip the near-wall region and to get better results from the simulations[16]. Also, modeling of the mesh near the wall is considerably important to get satisfying results in CFD analysis. The flow is affected by the viscous effects near the wall and the following formula is derived by using the dimensional analysis [24].

$$u^+ = \frac{U}{u_t} = f(\frac{\rho u_t y}{\mu}) = f(y^+)$$

44

This relation is called the law of the wall and y^+ is non-dimensional wall distance. Depending on the y^+ value, near the wall is divided into three regions. The region where $y^+ <7$ is called viscous sublayer, the region where $7<y^+<30$ is called buffer layer and the region where $30<y^+$ is called log-layer. These three regions can be seen in Figure 2-5.
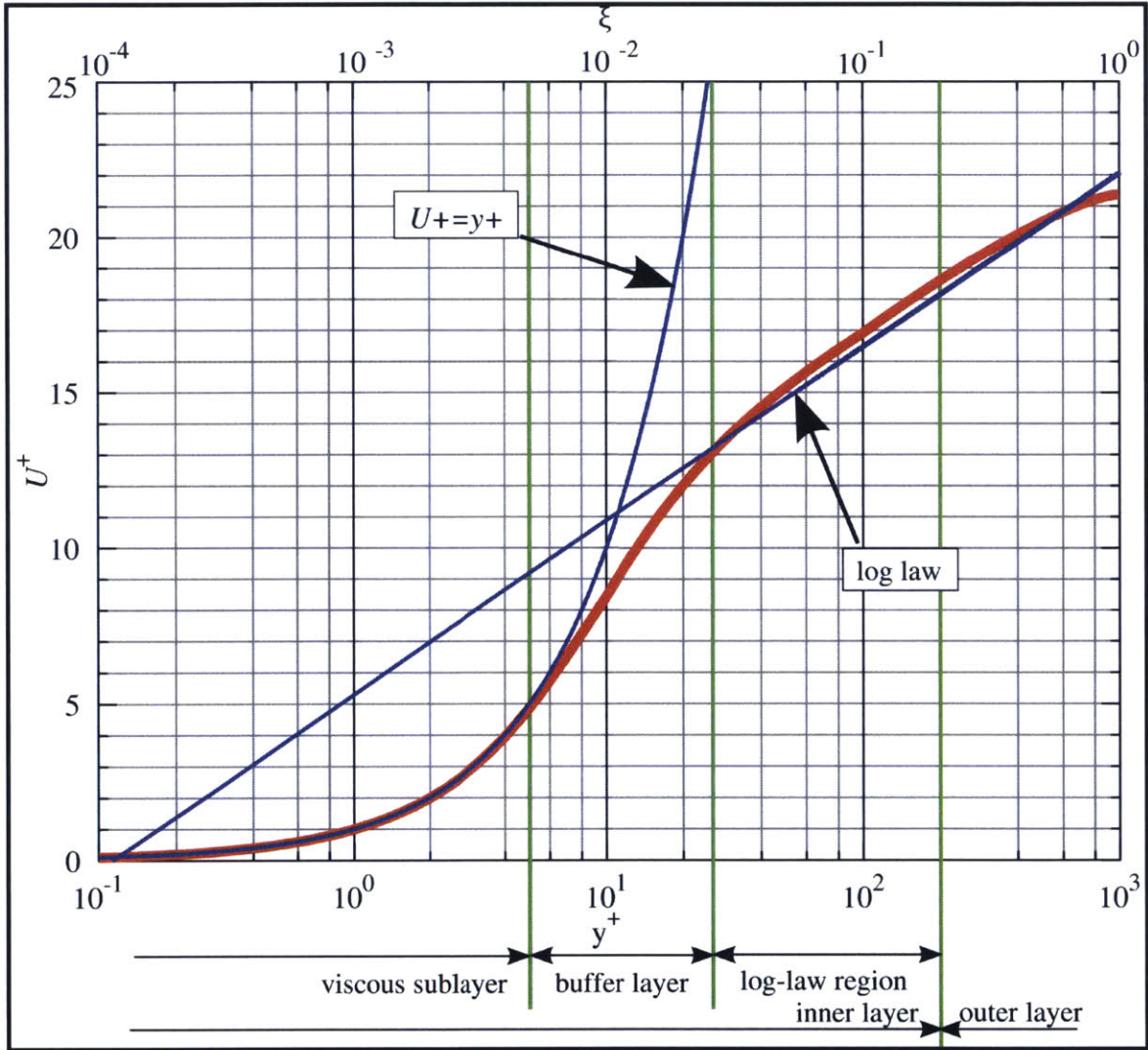
Figure 2-5: Law of the wall

During the mesh creation process, y^+ estimation is used to define the height of the first mesh by using the y^+ calculator [4]. The calculator uses the following formulas. For the estimation of C_f, there are many formulas available. yplus calculator uses

$C_f = 0.026 Re^{-(\frac{1}{7})}$ to estimate the friction coefficient.

$$\tau_{wall} = \frac{\rho C_f U^2}{2}$$

$$u_t = \sqrt{\frac{\tau_{wall}}{\rho}}$$

$$\Delta y = \frac{y^+ \mu}{\rho u_t}$$

where y is the distance to first cell from the wall in yplus calculator but it is the distance to the center of the first cell in openFoam, τ_{wall} is the wall shear stress, u_t is friction velocity, ρ is the fluid density and ν is the kinematic viscosity and it was $10^{-6} m^2/s^2$.

Firstly, y^+ was assumed to be 30 at Reynolds number 3e6 and the distance to the center of the first cell was calculated to be 2.54e-4. The total length of the mesh was 1 m. As it was mentioned before, openFoam uses the distance to the center of the cell for the calculation of y^+. Therefore, the calculated value in yplus calculator was multiplied by 2 two find the wall spacing and it was 5.08e-4. Gmsh was used to create the meshes and square shaped cells were tried to be created especially in near-wall region. The first mesh can be seen in Figure 2-6 and Figure 2-7.
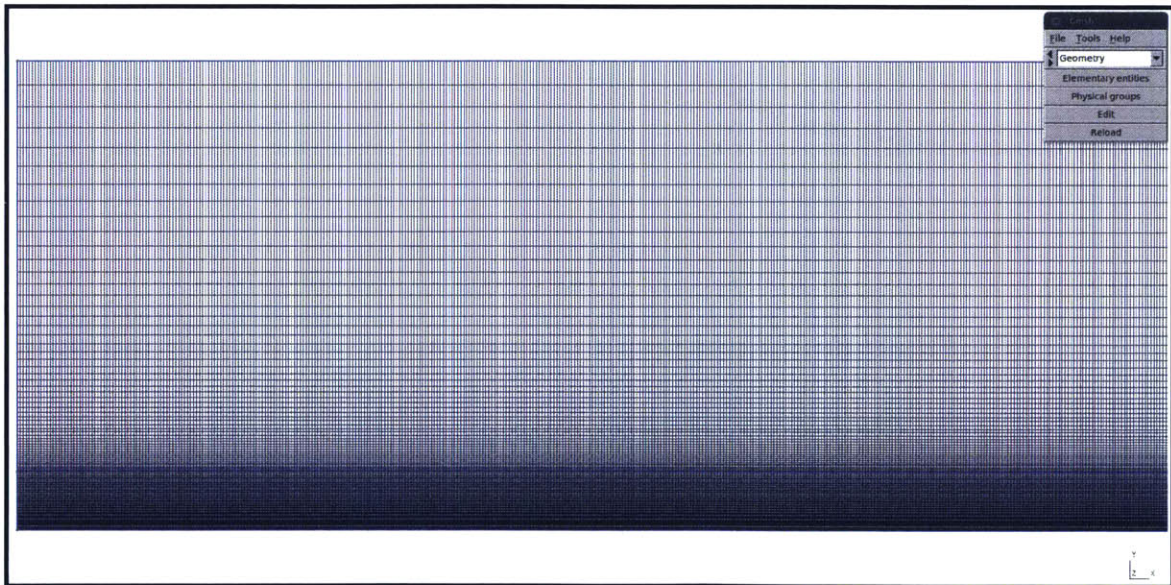
Figure 2-6: Mesh with wall spacing 5e-4 (y^+=30 at Re:3e6)

46
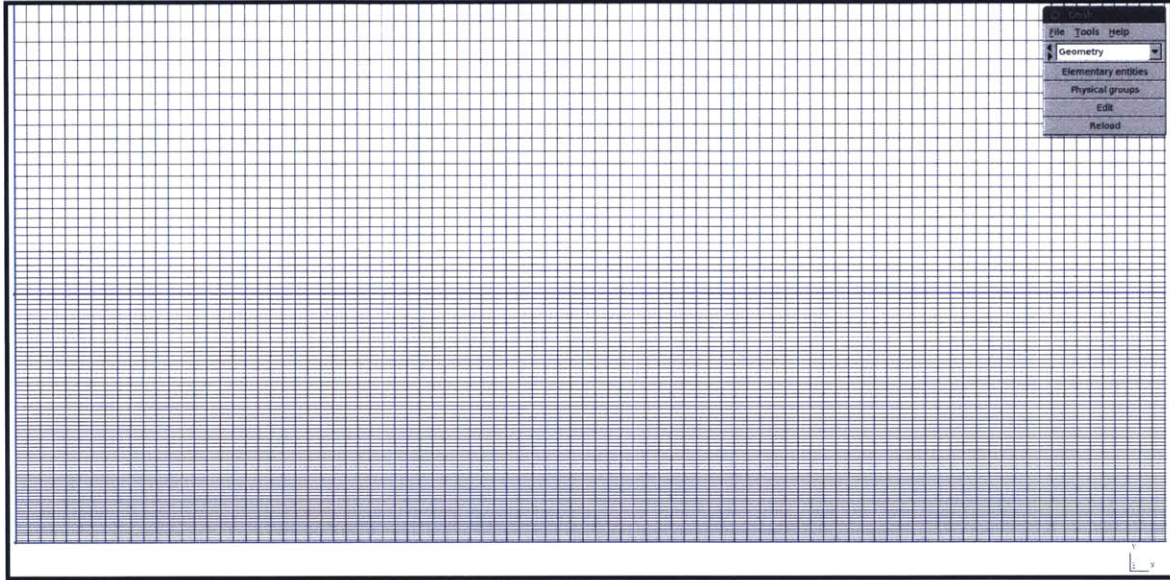
Figure 2-7: Mesh with wall spacing 5e-4 (y^+=30 at Re:3e6)

Then another mesh was created with y^+ is 15 at Reynolds number 1e5 (The equivalent y^+ was calculated to be 707 at Re:3e6). The same formulas were used to calculate the distance to the center of the first cell and it was 2.99e-3. Then, this value was multiplied by 2 and wall spacing was 5.988e-3. The mesh can be seen in Figure 2-8.
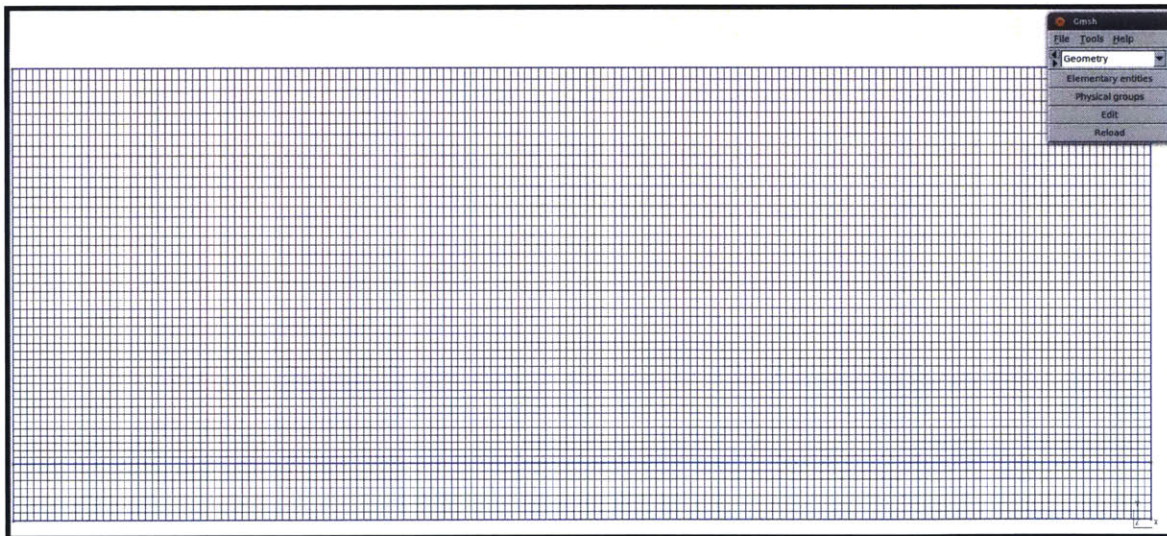
Figure 2-8: Mesh with wall spacing 5.988e-3 (y^+=707 at Re:3e6)

Finally, another mesh was created with y^+ is 30 at Reynolds number 1e5 (The

equivalent y^+ was calculated to be 1414 at Re:3e6. The same formulas were used to calculate the distance to the center of the first cell and it was 7.06e-4. Then, this value was multiplied by 2 and wall spacing was 1.41e-3. This mesh is quite coarse and it can be seen in Figure 2-9.
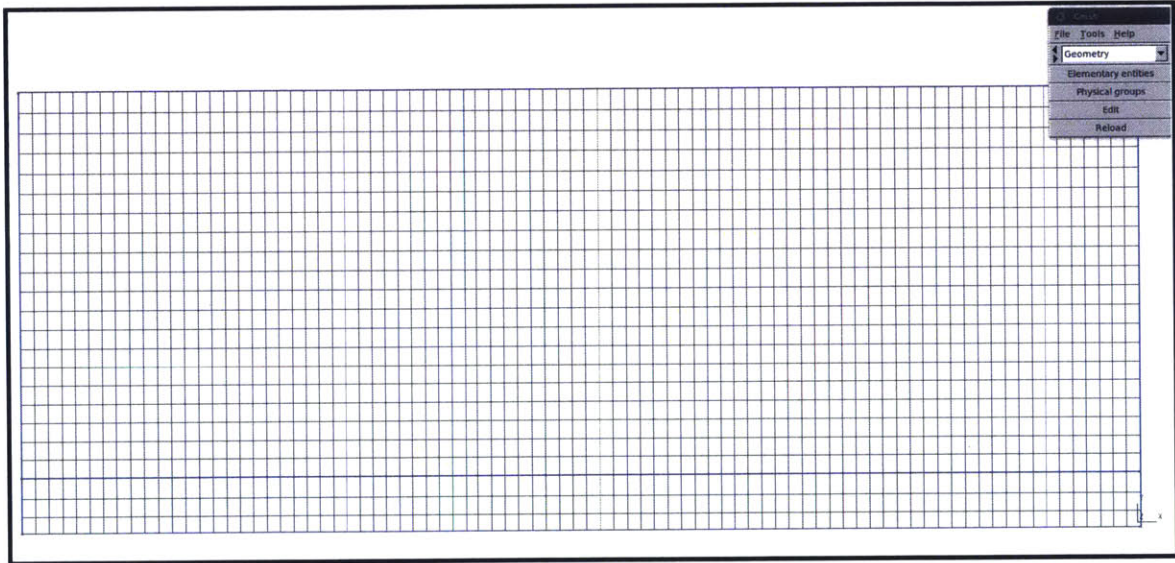
Figure 2-9: Mesh with wall spacing 1.41e-3 (y^+=1414 at Re:3e6)

As it can be seen in the figures that the meshes in Figure 2-8 and Figure 2-9 are coarser than the mesh in Figure 2-7. These three meshes were the main ones which will used for all Reynolds numbers ranging from 1e5 to 1e7. As well as these, other meshes are created specifically for Reynolds 1e5 and 3e6 to observe the effects of different meshes. The proporties of the meshes such as wall spacing, number of cells in z direction, progression in z direction, number of cells in x direction and progression in x direction can be seen in Table 2.3.

2.3.3 Processing

The important parts of running the simulations are to define the parameters of the turbulence models correctly, to choose appropriate boundary conditions and wall functions. In most high-Reynolds-number flows, the wall function approach substantially saves time because the viscosity effects near the wall changes very quick and this effect can be discarded by using the wall functions. The significant point

number	yplus	at Re	wall spacing	cells in y	prog.	cells in x	prog
1	0.1	1.000E+05	3.992E-05	197	1.015449	300	1
2	0.5	1.000E+05	1.996E-04	359	1.011974	1000	1
3	1	1.000E+05	3.992E-04	76	1.012261	300	1
4	15	1.000E+05	5.988E-03	7	1.010537	100	1
5	30	1.000E+05	1.198E-02	4	1.118344	84	1
6	0.5	3.000E+06	8.483E-06	286	1.016088	300	1
7	0.75	3.000E+06	1.27E-05	327	1.011888	300	1
8	0.9	3.000E+06	1.53E-05	313	1.011837	300	1
9	0.95	3.000E+06	1.61E-05	309	1.011819	300	1
10	1	3.000E+06	1.70E-05	305	1.011802	300	1
11	15	3.000E+06	2.545E-04	92	1.0125214	1000	1
12	30	3.000E+06	5.080E-04	69	1.010021	200	1
13	50	3.000E+06	8.483E-04	54	1.003096	800	1
14	70	3.000E+06	1.188E-03	32	1.016924	600	1
15	100	3.000E+06	1.697E-03	27	1.00632	589	1
16	200	3.000E+06	3.393E-03	12	1.035705	295	1

Table 2.3: The properties of the other meshes
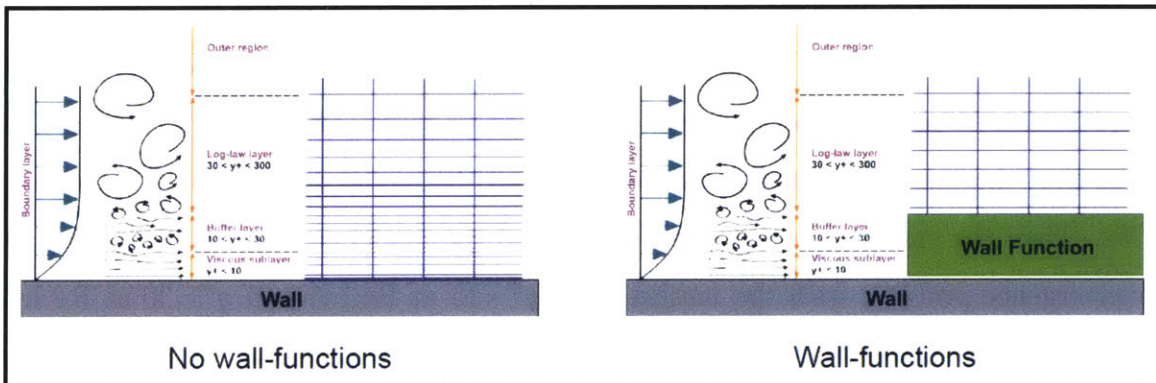
Figure 2-10: The represantation of wall function

at this level is to select an appropriate y^+ value to be able to use the wall functions properly.

No Turbulent Case

Firslty, Blasius formula was used to have an idea about the friction drag values. Blasius formula is given by:

$$C_f = \frac{1.328}{\sqrt{Re}}$$

The boundary conditions for this case can be seen in Table 2.4.

Mesh Part	Boundary Condition
top	symmetryPlane
inlet	fixedValue
outlet	zeroGradient
plate	fixedValue

Table 2.4: No Turbulent case boundary conditions

Setting up k-epsilon Case

The meshes 5 and 12 which were listed in Table 2.3 were used for this model. Firstly, the turbulent intensity was selected very high and the model did not converge. Then the turbulent intensity was lowered to 1% and then the model converged except at Reynolds numbers 1e5 and 2e5. The turbulent intensity for Re=1e5 was increased and decreased but it did not converge at any of those cases. The problem with Re=2e5 was that the courant number went too high and the model did not converge at time 70s. Then the endTime was changed to 50 and the model converged. There was no convergence problem with the meshes with y^+=15 at Re:1e5 and y^+=30 at Re:1e5. These results showed that the k-epsilon model is very sensitive to the y^+ value and the values of other model parameters. The initial freestream values of the model parameters for each Reynolds number can be seen in Table 2.5.

Setting up k-omega Case

The meshes 5 and 12 which were listed in Table 2.3 were used for this model. The formulas which were provided in the previous section were used to calculate k, omega and nut values and the results can be seen in Table 2.6. Wall boundary conditions for the mesh 12: $k_{wall} = 0$ and $\omega_{wall} = 1.28e4$ were used. Wall boundary conditions for mesh 5 were as follows: $k_{wall} = 0$ and $\omega_{wall} = 89.24114$.

Re	k	nut	Epsilon
1.00E+05	1.500E-06	1.832E-05	2.018E-08
2.00E+05	6.000E-06	3.190E-05	1.854E-07
3.00E+05	1.350E-05	4.412E-05	6.787E-07
4.00E+05	2.400E-05	5.554E-05	1.704E-06
5.00E+05	3.750E-05	6.640E-05	3.480E-06
6.00E+05	5.400E-05	7.682E-05	6.237E-06
7.00E+05	7.350E-05	8.691E-05	1.021E-05
8.00E+05	9.600E-05	9.671E-05	1.566E-05
9.00E+05	1.215E-04	1.063E-04	2.283E-05
1.00E+06	1.500E-04	1.156E-04	3.198E-05
2.00E+06	6.000E-04	2.013E-04	2.939E-04
3.00E+06	1.350E-03	2.784E-04	1.076E-03
4.00E+06	2.400E-03	3.504E-04	2.701E-03
5.00E+06	3.750E-03	4.189E-04	5.516E-03
6.00E+06	5.400E-03	4.847E-04	9.885E-03
7.00E+06	7.350E-03	5.483E-04	1.619E-02
8.00E+06	9.600E-03	6.102E-04	2.482E-02
9.00E+06	1.215E-02	6.705E-04	3.618E-02
1.00E+07	1.500E-02	7.294E-04	5.069E-02

Table 2.5: The k-epsilon model initial freestream values for the flat plate case

Setting up the Spalart-Allmaras Case

For Spalart-Allmaras case in openFoam, the files nut, nuTilda, p and U were created in the 0 directory. Considering the boundary conditions, $\tilde{\nu} = \nu = 0$ and nutUSpaldingWallFunction were used at the wall. For freestream, there were 2 boundary conditions to be used. They were as follows [7]:

freestream (fully turbulent): $\frac{\tilde{\nu}}{\nu} = 3 - 5$ $\left(\frac{\tilde{\nu}}{\nu} \approx 0.2 - 1.3 \right)$

freestream (tripped): $\frac{\tilde{\nu}}{\nu} << 1$ Considering these boundary conditions; three variations are created and these 3 variations are used for mesh 5. The kinematic viscosity is defined as 1e-6. Variation 0 is with the tripped freestream boundary condition and the value for internalField is chosen as 0 for both nut and nuTilda. Variation 1 is with the turbulent freestream boundary conditon and te upper limits were chosen. Therefore, internalField value for nut is 5e-6 and 1.3e-6 for nuTilda. Variation 3 is also the turbulent freestream boundary condtion and the lower limits were chosen for this case. The internalField value is 3e-6 for nut and 2e-7 for nuTilda. In the

Re	k	Omega	nut
1.00E+05	1.500E-06	1.49E-01	1.004E-05
2.00E+05	6.000E-06	3.43E-01	1.747E-05
3.00E+05	1.350E-05	5.59E-01	2.417E-05
4.00E+05	2.400E-05	7.89E-01	3.042E-05
5.00E+05	3.750E-05	1.03E+00	1.000E-08
6.00E+05	5.400E-05	1.28E+00	4.208E-05
7.00E+05	7.350E-05	1.54E+00	4.760E-05
8.00E+05	9.600E-05	1.81E+00	5.297E-05
9.00E+05	1.215E-04	2.09E+00	5.820E-05
1.00E+06	1.500E-04	2.37E+00	6.332E-05
2.00E+06	6.000E-04	5.44E+00	1.102E-04
3.00E+06	1.350E-03	8.85E+00	1.525E-04
4.00E+06	2.400E-03	1.25E+01	1.919E-04
5.00E+06	3.750E-03	1.63E+01	2.295E-04
6.00E+06	5.400E-03	2.03E+01	2.655E-04
7.00E+06	7.350E-03	2.45E+01	3.003E-04
8.00E+06	9.600E-03	2.87E+01	3.342E-04
9.00E+06	1.215E-02	3.31E+01	3.672E-04
1.00E+07	1.500E-02	3.75E+01	3.995E-04

Table 2.6: The k-omegaSST model model initial freestream values for the flat plate case

Spalart-Allmaras model, y^+ should be selected either less than 1 or greater than 30. Therefore, different meshes are created and a mesh with $y^+=30$ at Reynolds number 1e5 and meshes with $y^+ =< 1$ are created. The mesh with $y^+=30$ is tested with three different boundary conditions as listed in the following table and it is the reason that they are called variation 1, variation 2 and variation 3. The mesh with $y^+ =< 1$ is tested with the boundary conditions as listed in variation 3. The boundary conditions can be seen in Table 2.7:.

	nut freestream	nuTilda freestream	nut wall	nuTilda wall
Variation 1	0	0	0	0
Variation 2	5.00E-06	1.30E-06	0	0
Variation 3	3.00E-06	2.00E-07	0	0

Table 2.7: The Spalart-Allmaras model initial freestream values for the flat plate case

Setting up the kkl-omega Case

For this case, the same k, nut, and omea values which are calculated for the k-omega model were used in kkl-omega model. k file is changed to kt and kl freestream value was set to 0. This case was the most complicated one to prepare compared to the other models. For kkl-omega model, the y^+ value has to be selected less than 1 to be able to observe transition. A mesh with y^+ greater than 1 was still tested to prove that claim. This model was tested with mesh 5 and 6. Another important part of this model is the mesh creation process. It is stated in ANSYS Fluent 12.0 User's Guide [3] that mesh refinement at leading edge and near-wall region is important to be able to accurately observe the transition. It is also mentioned that it is crucial to do mesh refinement in the areas where the laminar separation begins to occur. As well as these, turbulent intensity also has a major effect on the model so that the inlet turbulence has to carefully be defined. The final and one of the major points in the guide stated as follows: "Finally, the decay of turbulence from the inlet to the leading edge of the device should always be estimated before running a solution as this can have a large effect on the predicted transition location [11]." It can be understood from this statement that the type of the mesh plays an important role on the model. For example, the leading edge of the flat plate used in mesh 5 and 6 begins from the inlet location. Therefore, another mesh which the leading edge of the mesh begins it 0.05 m away from the inlet region was created to be able observe the effect of different meshes on the simulations and this mesh was exactly the same mesh which was used by Jiri Furst in his simulations [13]. The total length of the plate is this mesh was 2.9 m.

2.4 Analysis of The Results

2.4.1 Convergence Criteria

In this part of the report, the C_d and C_f values are plotted in gnuplot as a function of time by using the forceCoeffs.dat file to determine which model converges faster

than the others and to make sure that the models converged. The cases with mesh 12 Reynolds number 7e5 are selected as an example for no turbulent, k-omega and k-espilon models. It is obvious from the following figures that the model converges very fast. For Spalart-Allmaras and kkl-omega models, the cases with mesh 6 at Reynolds number 3e6 are selected. These models take longer to be converged compared to the other models. The reason could be that the mesh which is used in these models is much finer than the one used in no turbulent, k-omega and k-epsilon models. Convergence of the models can be seen in the following figures.
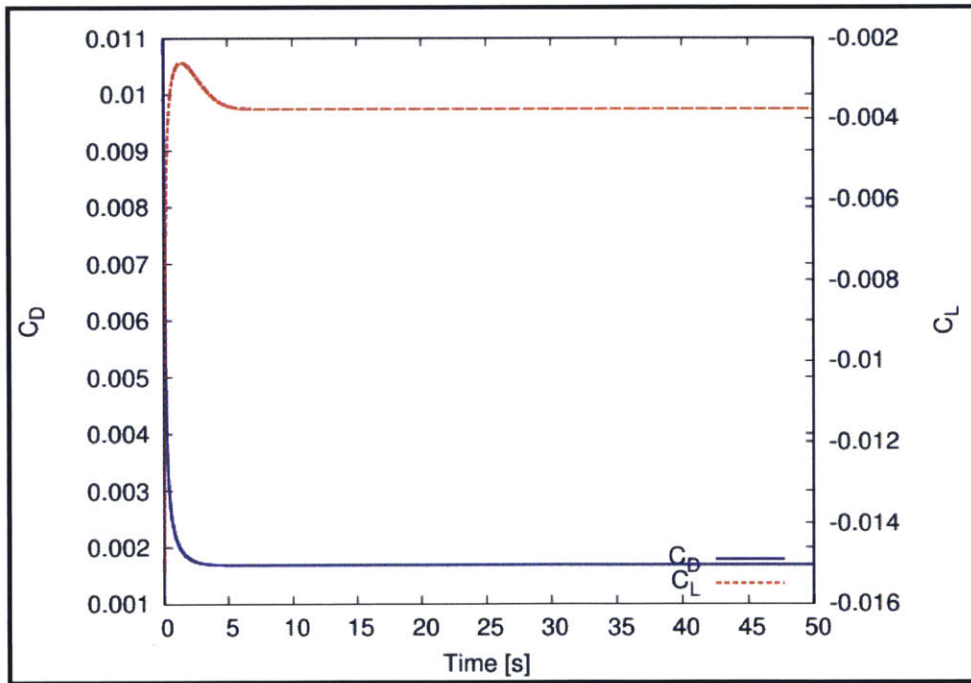
Figure 2-11: No turbulent case convergence
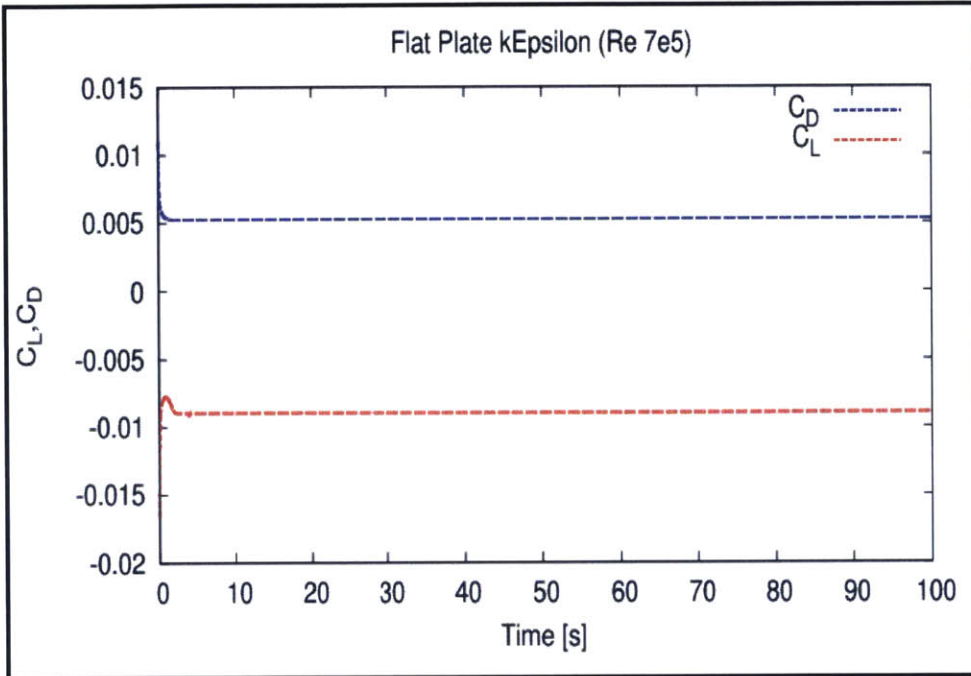
Figure 2-12: The k-epsilon case convergence
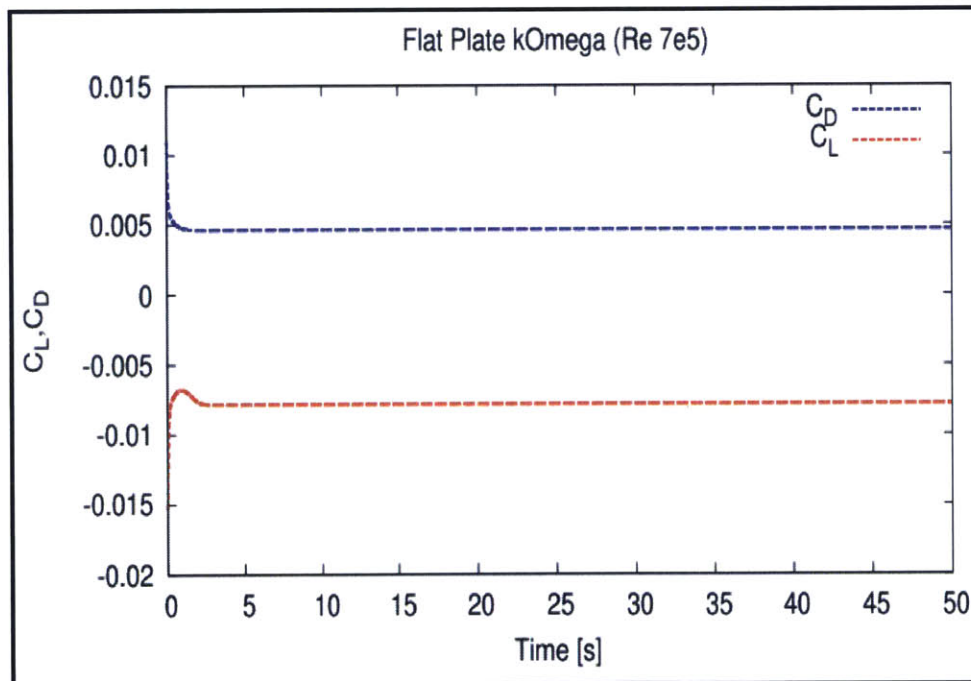
Figure 2-13: The k-omegaSST case convergence
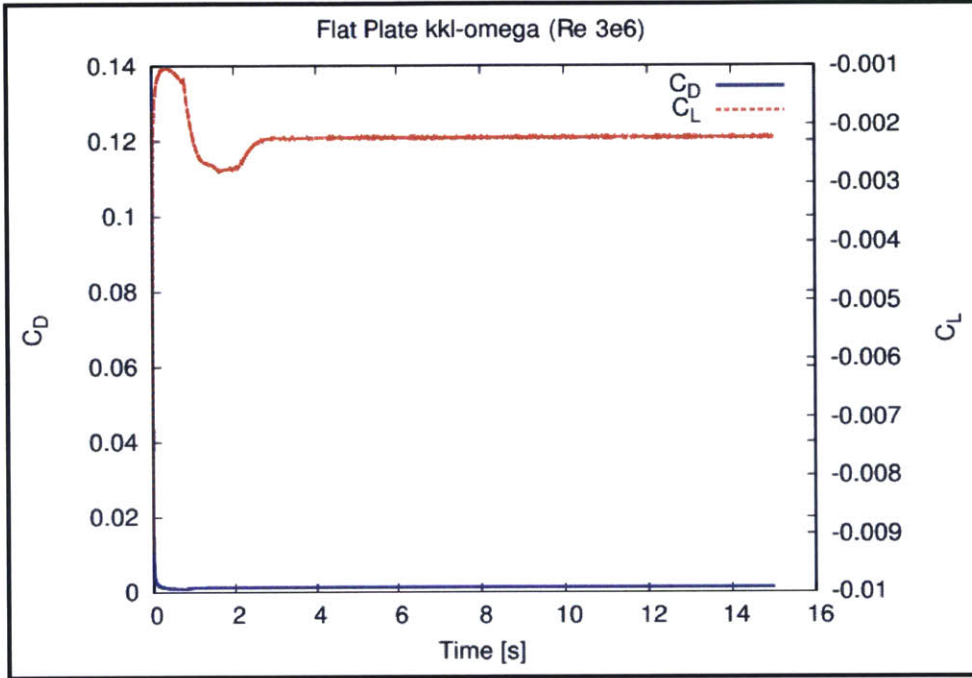
Figure 2-14: The kkl-omega case convergence
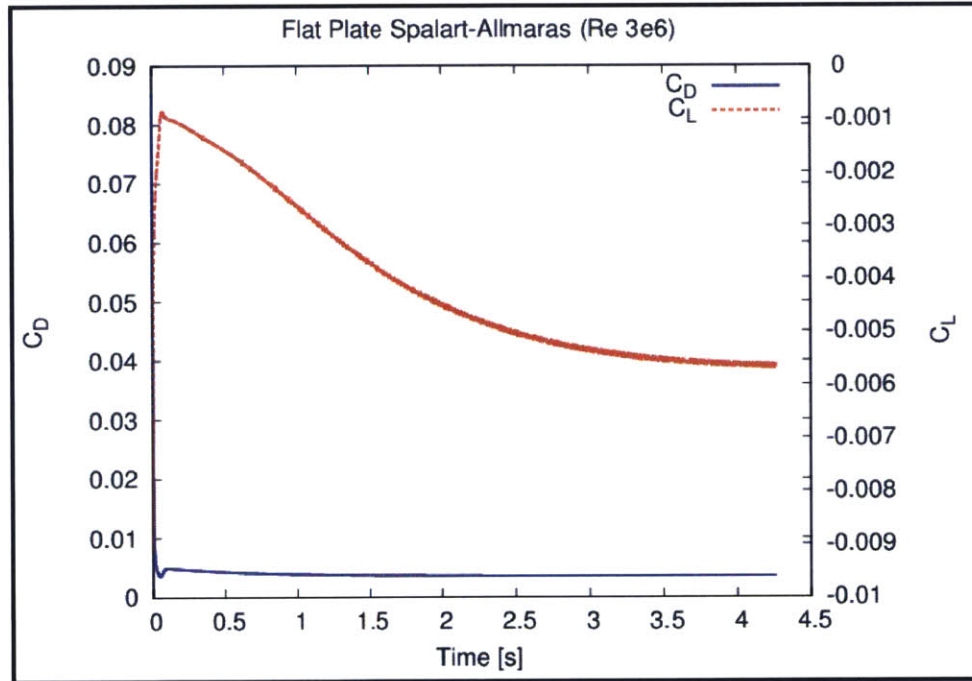
Figure 2-15: The Spalart-Allmaras case convergence

2.4.2 No Turbulent Case

The mesh 12 is used for all of the Reynolds numbers and the results can be seen in Table 2.9. The results were supposed to be close to the Schoenherr value especially

at low Reynolds numbers but they are not very close to the each other. The case is also tested with meshes 4 and 5 but the results were not close to the Schoenherr value due to the fact that those meshes were quite coarse.

Re	Cf Blasius	Cf OpenFoam
1.00E+05	4.20E-03	4.83E-03
2.00E+05	2.97E-03	3.28E-03
3.00E+05	2.42E-03	2.63E-03
4.00E+05	2.10E-03	2.26E-03
5.00E+05	1.88E-03	2.01E-03
6.00E+05	1.71E-03	1.82E-03
7.00E+05	1.59E-03	1.69E-03
8.00E+05	1.48E-03	1.57E-03
9.00E+05	1.40E-03	1.48E-03
1.00E+06	1.33E-03	1.41E-03
2.00E+06	9.39E-04	1.00E-03
3.00E+06	7.67E-04	8.27E-04
4.00E+06	6.64E-04	7.24E-04
5.00E+06	5.94E-04	6.54E-04
6.00E+06	5.42E-04	6.03E-04
7.00E+06	5.02E-04	5.63E-04
8.00E+06	4.70E-04	5.32E-04
9.00E+06	4.43E-04	5.07E-04
1.00E+07	4.20E-04	4.86E-04

Table 2.8: Friction drag coefficient results of the no turbulent case with mesh 12

Therefore, other meshes might be needed for both no turbulent and turbulent cases. As a result of this, different meshes for different y^+ values at Reynolds number 1e5 were created and the results can be seen in Table 2.9.

y^+ at Re:1e5	Cf for Re:1e5	Cf for Re:2e5
1	4.87E-03	3.32E-03
3	4.71E-03	3.25E-03
5	4.72E-03	3.33E-03
7	4.79E-03	3.43E-03

Table 2.9: Skin friction coefficient results for no turbulent case with different meshes

Re	Cf Blasius	Cf OpenFoam	percentage of difference
1.00E+05	4.20E-03	4.83E-03	15.01%
2.00E+05	2.97E-03	3.28E-03	10.46%
3.00E+05	2.42E-03	2.63E-03	8.48%
4.00E+05	2.10E-03	2.26E-03	7.43%
5.00E+05	1.88E-03	2.01E-03	6.81%
6.00E+05	1.71E-03	1.82E-03	6.42%
7.00E+05	1.59E-03	1.69E-03	6.19%
8.00E+05	1.48E-03	1.57E-03	6.06%
9.00E+05	1.40E-03	1.48E-03	5.99%
1.00E+06	1.33E-03	1.41E-03	5.96%
2.00E+06	9.39E-04	1.00E-03	6.71%
3.00E+06	7.67E-04	8.27E-04	7.88%
4.00E+06	6.64E-04	7.24E-04	9.03%
5.00E+06	5.94E-04	6.54E-04	10.10%
6.00E+06	5.42E-04	6.03E-04	11.15%
7.00E+06	5.02E-04	5.63E-04	12.21%
8.00E+06	4.70E-04	5.32E-04	13.31%
9.00E+06	4.43E-04	5.07E-04	14.46%
1.00E+07	4.20E-04	4.86E-04	15.63%

Table 2.10: Percentage of difference for the no turbulent case with mesh 12
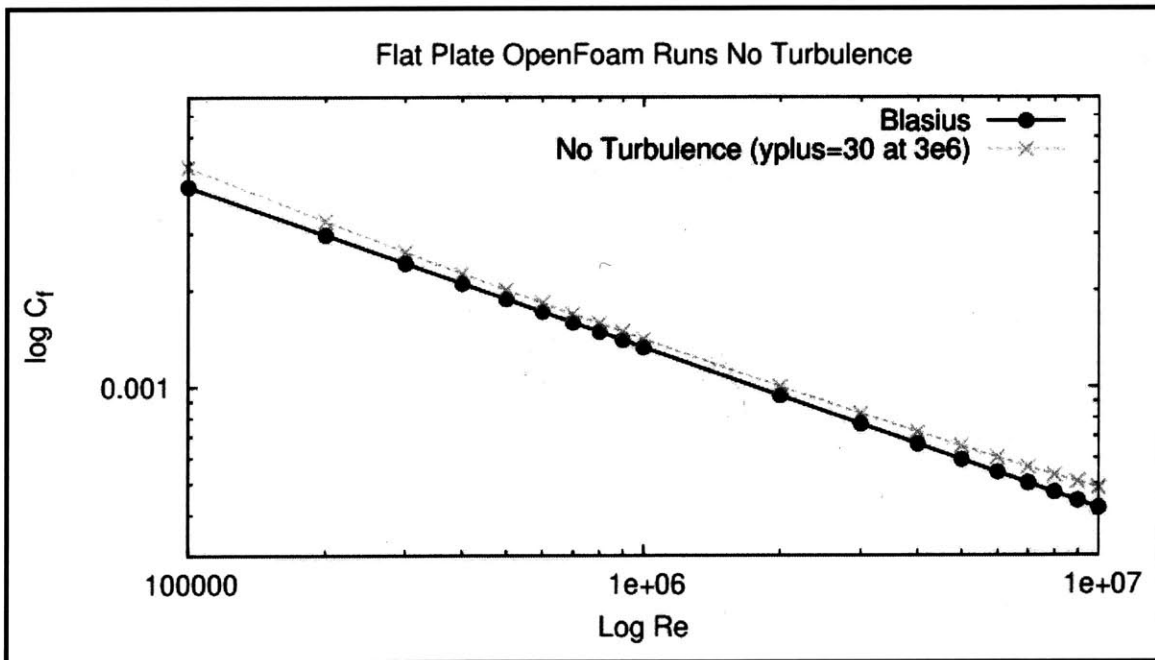
Figure 2-16: No turbulent case skin friction comparison with the Blasius line

2.4.3 The k-omega and k-epsilon Cases

For the k-epsilon and k-omega cases, four different variations were used. Variation 1 was the mesh 4, variation 2 was the mesh 5, variation 4 was the mesh with 12. For these cases, nutkWallFunction was used as wall boundary condition in nut file. Variation 3 was the mesh 12 again but nutkWallFunction was used as a wall boundary condition in nut file. The results and percentage of the differences between Scheonherr value and openFoam runs for k-epsilon, k-omega cases can be seen in the following tables and figures.

Re	Scheonherr	ITTC57	Var 1	Var 2	Var 3	Var 4
1.E+05	7.18E-03	8.33E-03	8.12E-03	7.62E-03		
2.E+05	6.14E-03	6.88E-03	6.62E-03	6.35E-03	9.44E-03	9.48E-03
3.E+05	5.62E-03	6.20E-03	5.98E-03	5.76E-03	8.06E-03	8.16E-03
4.E+05	5.29E-03	5.78E-03	5.59E-03	5.40E-03	7.09E-03	7.26E-03
5.E+05	5.06E-03	5.48E-03	5.32E-03	5.15E-03	6.35E-03	6.62E-03
6.E+05	4.87E-03	5.25E-03	5.12E-03	4.95E-03	5.76E-03	6.15E-03
7.E+05	4.73E-03	5.07E-03	4.96E-03	4.79E-03	5.28E-03	5.78E-03
8.E+05	4.60E-03	4.92E-03	4.82E-03	4.66E-03	4.90E-03	5.50E-03
9.E+05	4.50E-03	4.80E-03	4.71E-03	4.55E-03	4.58E-03	5.27E-03
1.E+06	4.41E-03	4.69E-03	4.61E-03	4.46E-03	4.34E-03	5.08E-03
2.E+06	3.87E-03	4.05E-03	4.04E-03	3.90E-03	3.78E-03	4.15E-03
3.E+06	3.60E-03	3.74E-03	3.76E-03	3.62E-03	3.53E-03	3.77E-03
4.E+06	3.42E-03	3.54E-03	3.57E-03	3.44E-03	3.36E-03	3.77E-03
5.E+06	3.29E-03	3.40E-03	3.44E-03	3.31E-03	3.24E-03	3.40E-03
6.E+06	3.19E-03	3.29E-03	3.33E-03	3.21E-03	3.14E-03	3.28E-03
7.E+06	3.11E-03	3.19E-03	3.25E-03	3.12E-03	3.06E-03	3.19E-03
8.E+06	3.04E-03	3.12E-03	3.18E-03	3.05E-03	3.00E-03	3.12E-03
9.E+06	2.99E-03	3.06E-03	3.12E-03	2.99E-03	2.95E-03	3.05E-03
1.E+07	2.93E-03	3.00E-03	3.06E-03	2.94E-03	2.90E-03	3.00E-03

Table 2.11: The k-epsilon model skin fricton coefficient results

As it can be seen from te results, the cases with the mesh 12 provided promising results but that mesh was is too coarse to be used in CFD analysis. It is concluded from these results tht wall functions has an important role in numerical simulations so that they provide good results.

Re	Var 1	Var 2	Var 3	Var 4
1.00E+05	13.04%	6.17%		
2.00E+05	7.86%	3.44%	53.77%	54.42%
3.00E+05	6.47%	2.60%	43.50%	45.22%
4.00E+05	5.68%	2.11%	33.93%	37.23%
5.00E+05	5.27%	1.79%	25.52%	31.03%
6.00E+05	5.01%	1.58%	18.17%	26.13%
7.00E+05	4.85%	1.42%	11.75%	22.33%
8.00E+05	4.73%	1.30%	6.32%	19.37%
9.00E+05	4.65%	1.21%	1.73%	17.03%
1.00E+06	4.60%	1.14%	1.57%	15.16%
2.00E+06	4.49%	0.85%	2.16%	7.17%
3.00E+06	4.47%	0.68%	1.97%	4.80%
4.00E+06	4.42%	0.54%	1.87%	10.17%
5.00E+06	4.41%	0.45%	1.73%	3.13%
6.00E+06	4.42%	0.39%	1.62%	2.77%
7.00E+06	4.42%	0.35%	1.52%	2.54%
8.00E+06	4.43%	0.31%	1.42%	2.39%
9.00E+06	4.44%	0.28%	1.33%	2.28%
1.00E+07	4.45%	0.25%	1.26%	2.20%

Table 2.12: The k-epsilon model percentage of differences
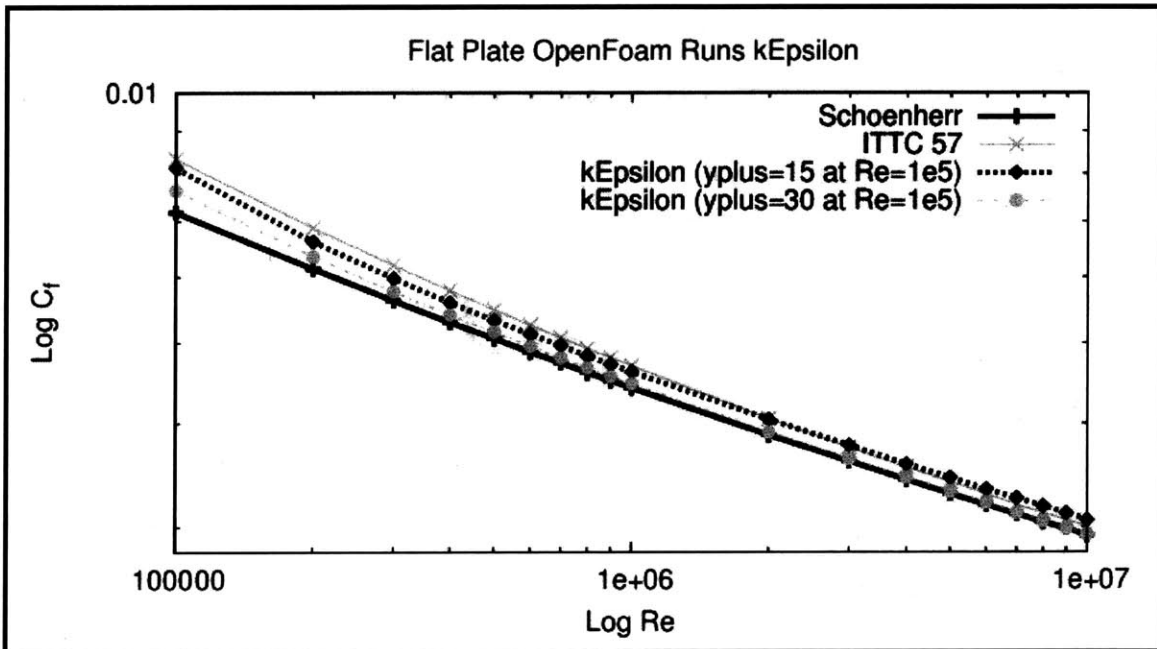
Figure 2-17: The k-epsilon model skin friction comparison with Scheonherr and ITTC'57 lines
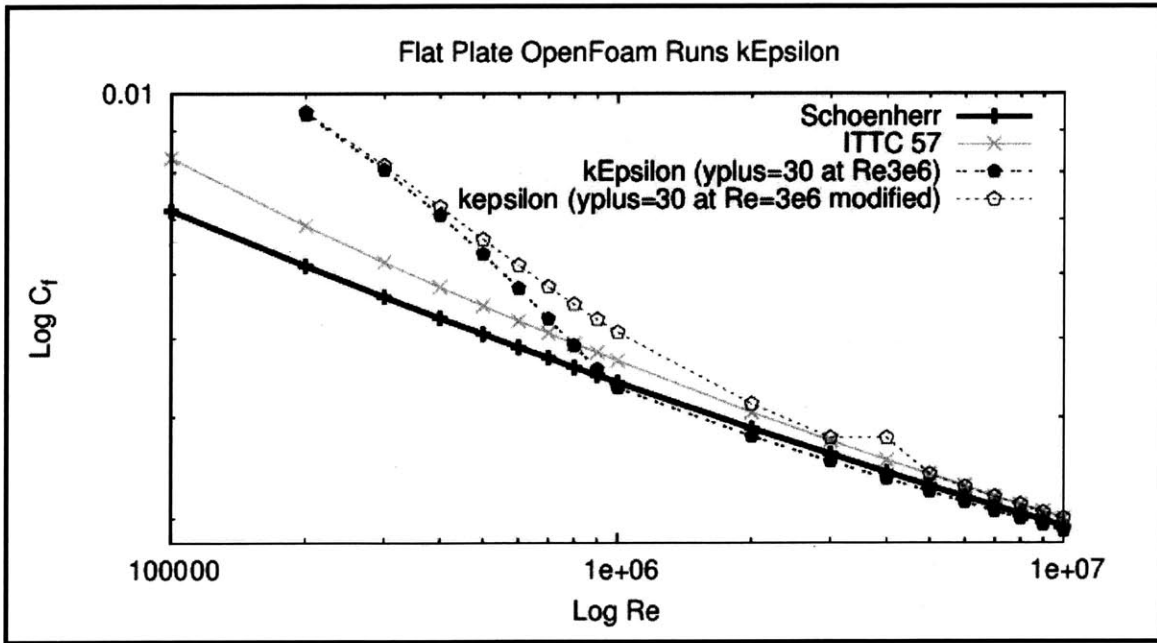
Figure 2-18: The k-epsilon model skin friction comparison with Scheonherr and ITTC'57 lines

Re	Scheonherr	ITTC57	Var 1	Var 2	Var 3	Var 4
1.E+05	7.18E-03	8.33E-03	7.66E-03	7.48E-03	6.77E-03	6.77E-03
2.E+05	6.14E-03	6.88E-03	6.28E-03	6.25E-03	5.75E-03	5.77E-03
3.E+05	5.62E-03	6.20E-03	5.69E-03	5.69E-03	5.59E-03	5.64E-03
4.E+05	5.29E-03	5.78E-03	5.33E-03	5.34E-03	5.47E-03	5.57E-03
5.E+05	5.06E-03	5.48E-03	5.08E-03	5.09E-03	5.21E-03	5.39E-03
6.E+05	4.87E-03	5.25E-03	4.89E-03	4.90E-03	4.92E-03	5.20E-03
7.E+05	4.73E-03	5.07E-03	4.75E-03	4.74E-03	4.63E-03	5.01E-03
8.E+05	4.60E-03	4.92E-03	4.62E-03	4.62E-03	4.37E-03	4.85E-03
9.E+05	4.50E-03	4.80E-03	4.52E-03	4.51E-03	4.13E-03	4.71E-03
1.E+06	4.41E-03	4.69E-03	4.43E-03	4.42E-03	3.92E-03	4.59E-03
2.E+06	3.87E-03	4.05E-03	3.90E-03	3.87E-03	3.45E-03	3.89E-03
3.E+06	3.60E-03	3.74E-03	3.64E-03	3.60E-03	3.30E-03	3.56E-03
4.E+06	3.42E-03	3.54E-03	3.47E-03	3.42E-03	3.17E-03	3.36E-03
5.E+06	3.29E-03	3.40E-03	3.34E-03	3.29E-03	3.07E-03	3.22E-03
6.E+06	3.19E-03	3.29E-03	3.25E-03	3.19E-03	2.99E-03	3.11E-03
7.E+06	3.11E-03	3.19E-03	3.17E-03	3.11E-03	2.93E-03	3.03E-03
8.E+06	3.04E-03	3.12E-03	3.10E-03	3.04E-03	2.87E-03	2.96E-03
9.E+06	2.99E-03	3.06E-03	3.04E-03	2.98E-03	2.82E-03	2.90E-03
1.E+07	2.93E-03	3.00E-03	2.99E-03	2.93E-03	2.78E-03	2.85E-03

Table 2.13: The k-omegaSST model skin fricton coeficient results

Re	Var 1	Var 2	Var 3	Var 4
1.E+05	6.69%	4.12%	5.72%	5.64%
2.E+05	2.36%	1.81%	6.38%	6.05%
3.E+05	1.29%	1.24%	0.45%	0.47%
4.E+05	0.74%	1.00%	3.34%	5.32%
5.E+05	0.52%	0.60%	3.07%	6.63%
6.E+05	0.43%	0.45%	0.99%	6.62%
7.E+05	0.40%	0.35%	2.03%	5.94%
8.E+05	0.40%	0.28%	5.05%	5.39%
9.E+05	0.43%	0.22%	8.22%	4.72%
1.E+06	0.47%	0.19%	11.11%	4.09%
2.E+06	0.94%	0.07%	10.91%	0.50%
3.E+06	1.23%	0.01%	8.28%	0.95%
4.E+06	1.39%	0.11%	7.23%	1.74%
5.E+06	1.53%	0.15%	6.64%	2.18%
6.E+06	1.64%	0.18%	6.26%	2.45%
7.E+06	1.74%	0.21%	5.96%	2.63%
8.E+06	1.83%	0.22%	5.73%	2.75%
9.E+06	1.90%	0.24%	5.56%	2.83%
1.E+07	1.97%	0.25%	5.41%	2.89%

Table 2.14: The k-omegaSST model percentage of differences
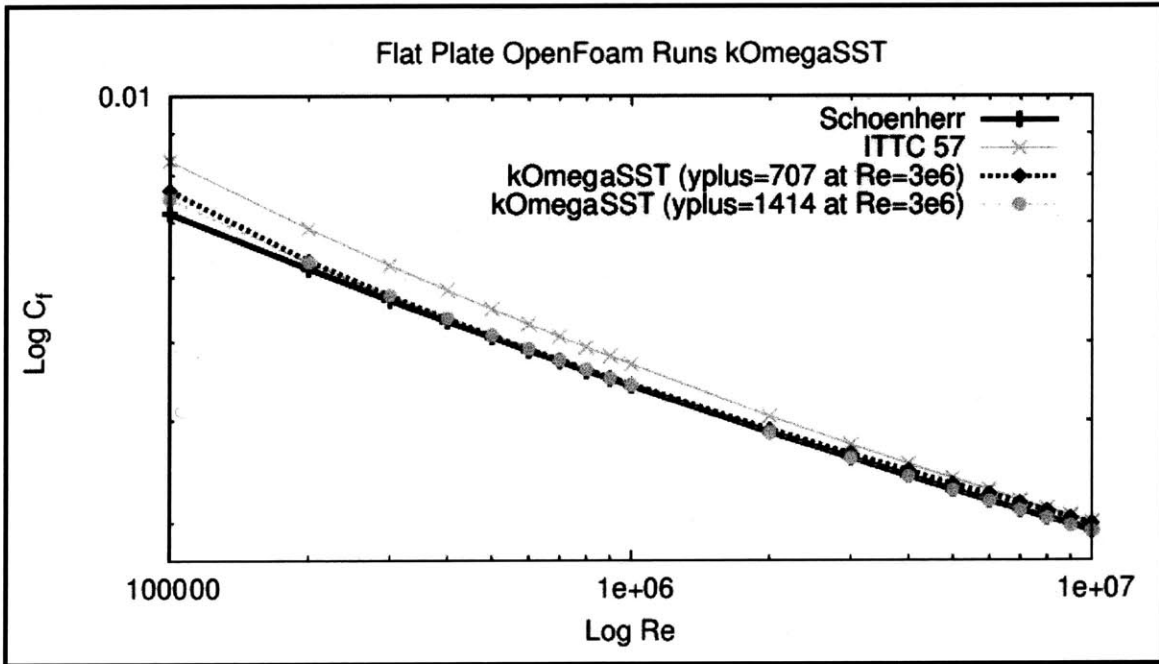
Figure 2-19: The k-omegaSST model skin friction comparison with Scheonherr and ITTC'57 lines
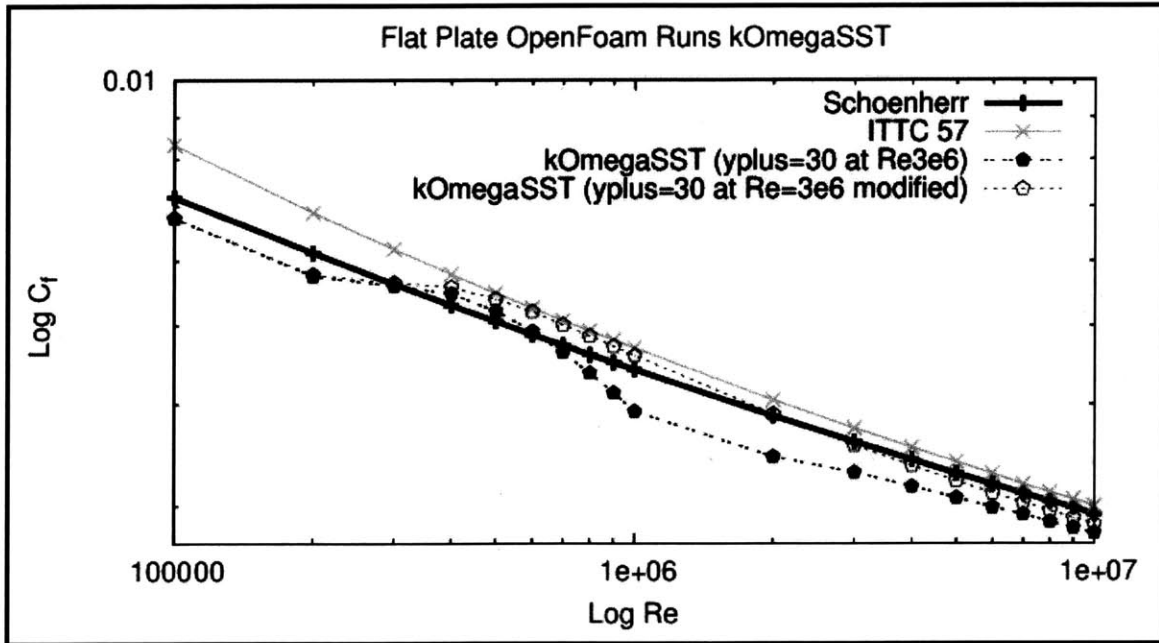
Figure 2-20: The k-omegaSST model skin friction comparison with Scheonherr and ITTC'57 lines

2.4.4 Spalart-Allmaras Case

As mentioned before, 3 different variations were created for the Spalart-Allmaras case. The results for thesse 3 variations can be seen in Table 2.15. It is obvious from the results that Variation 2 provided more promising results compared to the other variations. The results were very close to the Schoenherr value but the problem with this mesh was that the mesh was quite coarse. Therefore, other meshes with more reasonable y^+ values created. The meshes with y^+ less than 1 also provided close results to he Schoenher value and they can be seen in Table 2.17. For these meshes, the boundary conditions of Variation 2 were used.

Re	Schoenherr	ITTC'57	Variation 1	Variation 2	Variaton 3
1.00E+05	7.18E-03	8.33E-03	7.30E-03	7.45E-03	7.33E-03
2.00E+05	6.14E-03	6.88E-03	6.15E-03	6.25E-03	6.17E-03
3.00E+05	5.62E-03	6.20E-03	5.62E-03	5.69E-03	5.63E-03
4.00E+05	5.29E-03	5.78E-03	5.28E-03	5.34E-03	5.29E-03
5.00E+05	5.06E-03	5.48E-03	5.04E-03	5.09E-03	5.05E-03
6.00E+05	4.87E-03	5.25E-03	4.86E-03	4.90E-03	4.86E-03
7.00E+05	4.73E-03	5.07E-03	4.71E-03	4.74E-03	4.72E-03
8.00E+05	4.60E-03	4.92E-03	4.59E-03	4.62E-03	4.59E-03
9.00E+05	4.50E-03	4.80E-03	4.48E-03	4.51E-03	4.49E-03
1.00E+06	4.41E-03	4.69E-03	4.39E-03	4.42E-03	4.40E-03
2.00E+06	3.87E-03	4.05E-03	3.86E-03	3.87E-03	3.86E-03
3.00E+06	3.60E-03	3.74E-03	3.59E-03	3.60E-03	3.59E-03
4.00E+06	3.42E-03	3.54E-03	3.41E-03	3.42E-03	3.41E-03
5.00E+06	3.29E-03	3.40E-03	3.28E-03	3.29E-03	3.28E-03
6.00E+06	3.19E-03	3.29E-03	3.18E-03	3.19E-03	3.18E-03
7.00E+06	3.11E-03	3.19E-03	3.10E-03	3.11E-03	3.10E-03
8.00E+06	3.04E-03	3.12E-03	3.03E-03	3.04E-03	3.03E-03
9.00E+06	2.99E-03	3.06E-03	2.98E-03	2.98E-03	2.98E-03
1.00E+07	2.93E-03	3.00E-03	2.92E-03	2.93E-03	2.92E-03

Table 2.15: The Spalart-Allmaras model skin fricton coefficient results
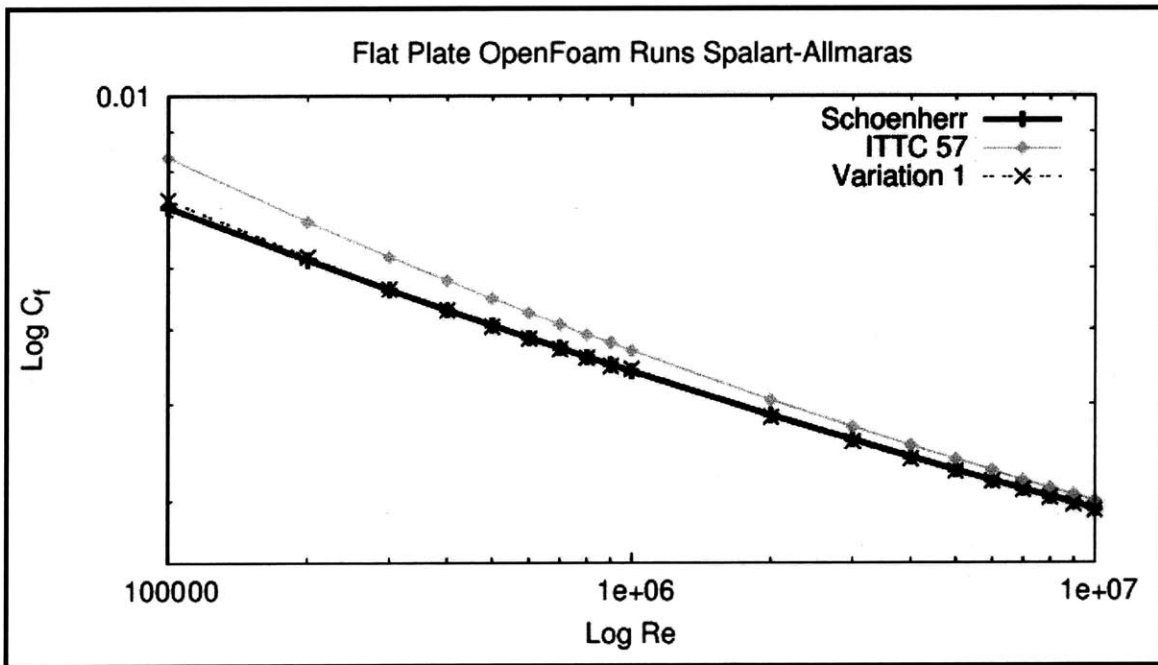
Figure 2-21: The Spalart Allmaras model variation 1 case skin friction comparison with Scheonherr and ITTC'57 lines

Re	Difference Var1	Difference Var2	Difference Var2
1.00E+05	1.69%	3.84%	2.05%
2.00E+05	0.27%	1.83%	0.56%
3.00E+05	0.04%	1.28%	0.27%
4.00E+05	0.19%	0.86%	0.00%
5.00E+05	0.29%	0.62%	0.12%
6.00E+05	0.35%	0.47%	0.19%
7.00E+05	0.37%	0.37%	0.24%
8.00E+05	0.39%	0.29%	0.27%
9.00E+05	0.40%	0.24%	0.28%
1.00E+06	0.40%	0.20%	0.29%
2.00E+06	0.29%	0.09%	0.22%
3.00E+06	0.28%	0.01%	0.23%
4.00E+06	0.32%	0.08%	0.28%
5.00E+06	0.33%	0.13%	0.29%
6.00E+06	0.33%	0.16%	0.30%
7.00E+06	0.33%	0.18%	0.31%
8.00E+06	0.33%	0.20%	0.31%
9.00E+06	0.34%	0.21%	0.31%
1.00E+07	0.34%	0.22%	0.32%

Table 2.16: The Spalart-Allmaras model percentage of differences
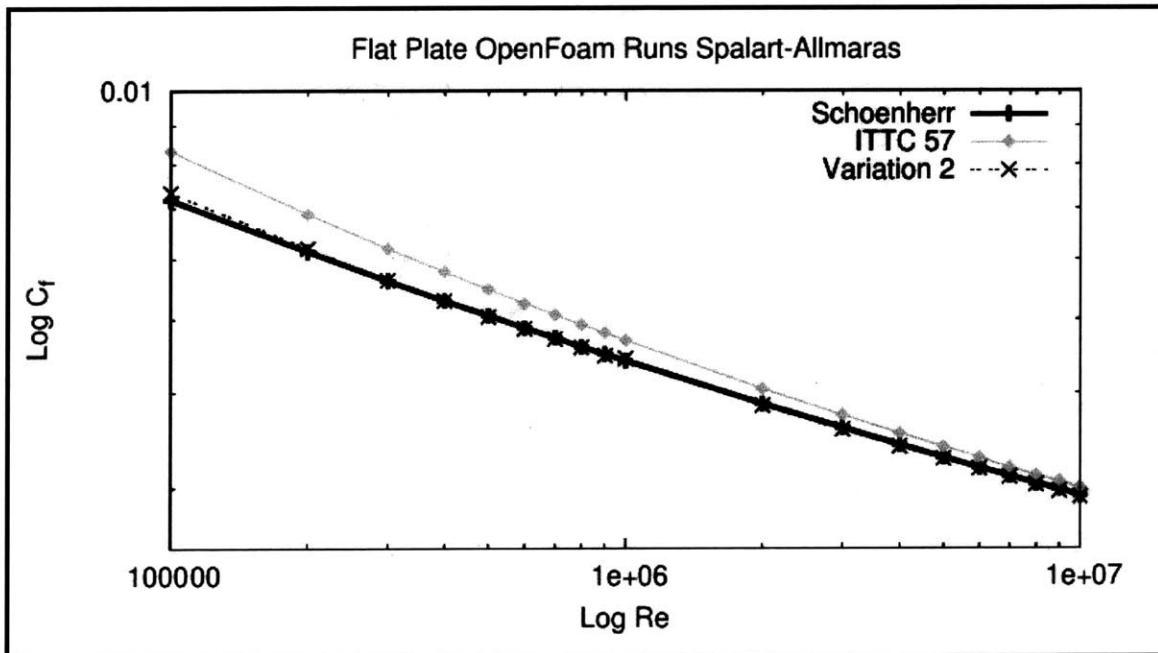
Figure 2-22: The Spalart Allmaras model variation 2 case skin friction comparison with Scheonherr and ITTC'57 lines
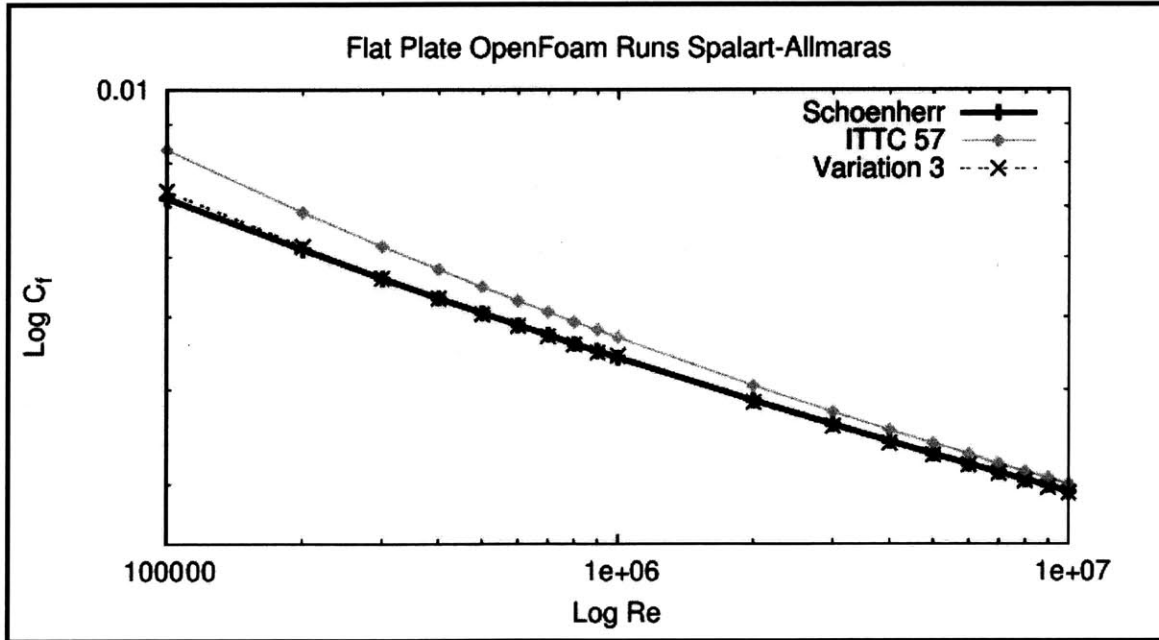
Figure 2-23: The Spalart Allmaras model variation 3 case skin friction comparison with Scheonherr and ITTC'57 lines

yplus	Cf Schoenherr	Cf OpenFoam
0.5	3.596E-03	3.512E-03
0.75	3.596E-03	3.522E-03
0.9	3.596E-03	3.513E-03

Table 2.17: The Spalart-Allmaras model skin fricton coefficient results

2.4.5 kkl-omega Case

For the kkl-omega case, the mesh 5 was used and the results were very close to the Schoenherr line especially at high Reynolds numbers but the model did not provide a transition from laminar region to turbulent region. Therefore, the cases were run with lower yplus values to be able to observe a transition. The results of case with the mesh 5 can be seen in Table 2.18.

When y^+ selected less then 1, the transition can be observed ove the plate but the model underestimates the skin frcition coefficient. The transition can be seen in the following Figure 2-26.

As it can be seen from the Figure 2-26, current version of the kkl-omega model cannot estimate the skin friction coefficient right. The problem with the model is that the model is programmed according to the article which was proposed in 2008

66

Re	Schoenherr	ITTC'57	kkl-omega	Percentage of difference
1.00E+05	7.18E-03	8.33E-03	7.34E-03	2.26%
2.00E+05	6.14E-03	6.88E-03	6.20E-03	0.98%
3.00E+05	5.62E-03	6.20E-03	5.63E-03	0.23%
4.00E+05	5.29E-03	5.78E-03	5.29E-03	0.07%
5.00E+05	5.06E-03	5.48E-03	5.04E-03	0.22%
6.00E+05	4.87E-03	5.25E-03	4.86E-03	0.31%
7.00E+05	4.73E-03	5.07E-03	4.71E-03	0.37%
8.00E+05	4.60E-03	4.92E-03	4.59E-03	0.41%
9.00E+05	4.50E-03	4.80E-03	4.48E-03	0.43%
1.00E+06	4.41E-03	4.69E-03	4.44E-03	0.68%
2.00E+06	3.87E-03	4.05E-03	3.85E-03	0.42%
3.00E+06	3.60E-03	3.74E-03	3.58E-03	0.44%
4.00E+06	3.42E-03	3.54E-03	3.40E-03	0.50%
5.00E+06	3.29E-03	3.40E-03	3.28E-03	0.52%
6.00E+06	3.19E-03	3.29E-03	3.18E-03	0.53%
7.00E+06	3.11E-03	3.19E-03	3.10E-03	0.53%
8.00E+06	3.04E-03	3.12E-03	3.03E-03	0.54%
9.00E+06	2.99E-03	3.06E-03	2.97E-03	0.54%
1.00E+07	2.93E-03	3.00E-03	2.92E-03	0.54%

Table 2.18: The kkl-omega model skin fricton coefficient results
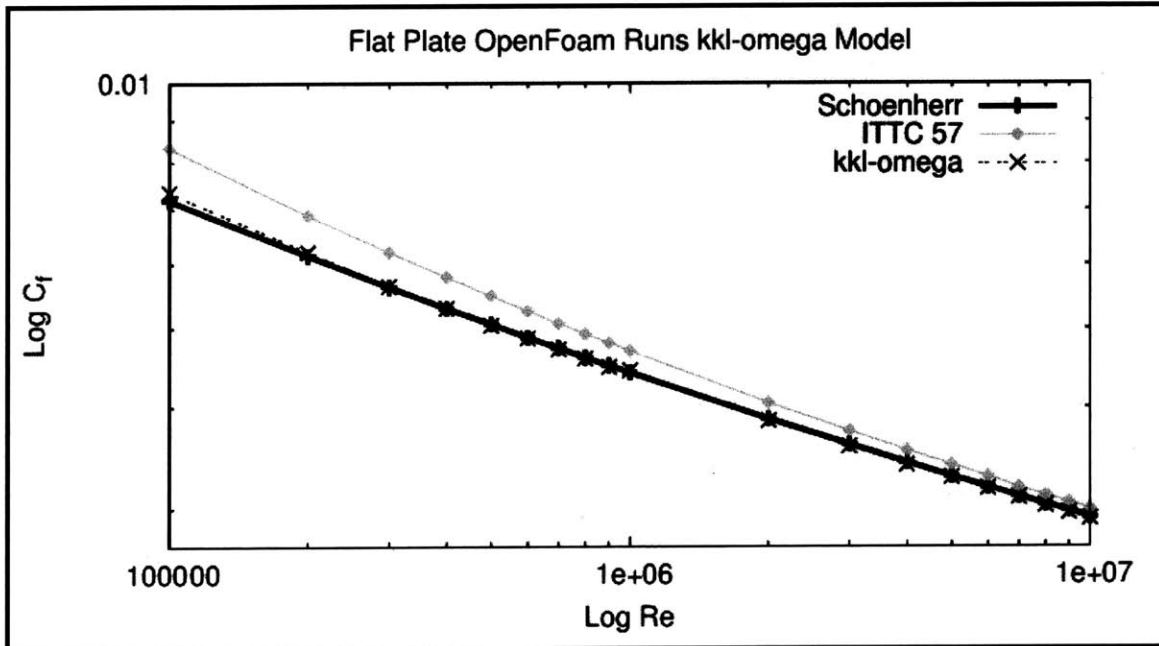
Figure 2-24: The kkl-omega model skin friction comparison with Scheonherr and ITTC'57 lines
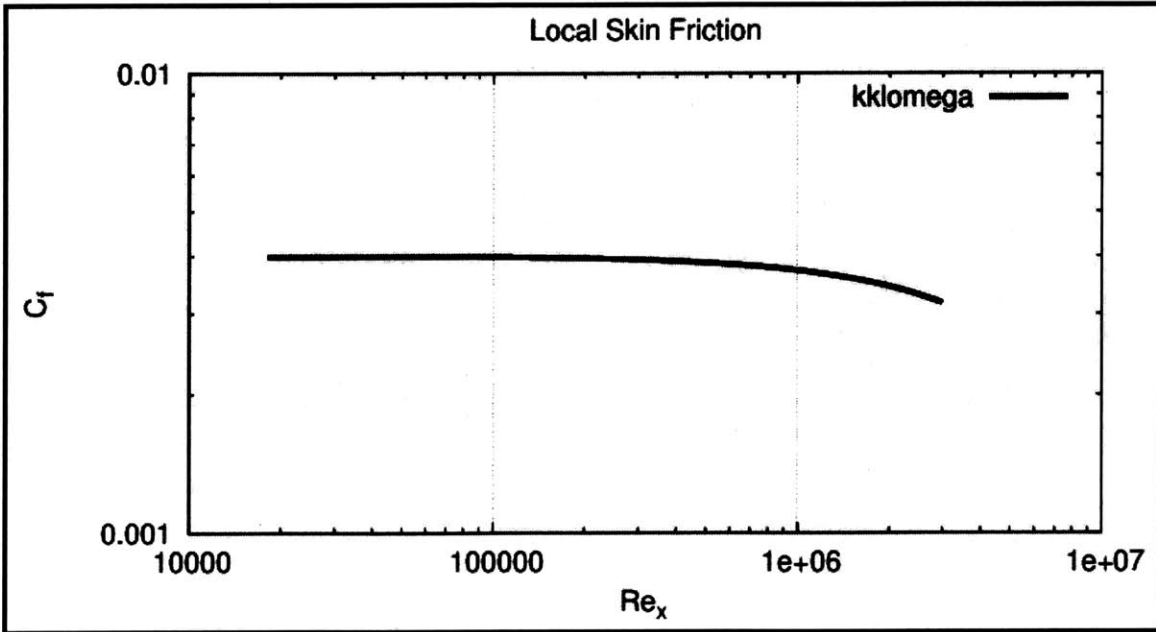
Figure 2-25: The kkl-omega model local skin friction plot
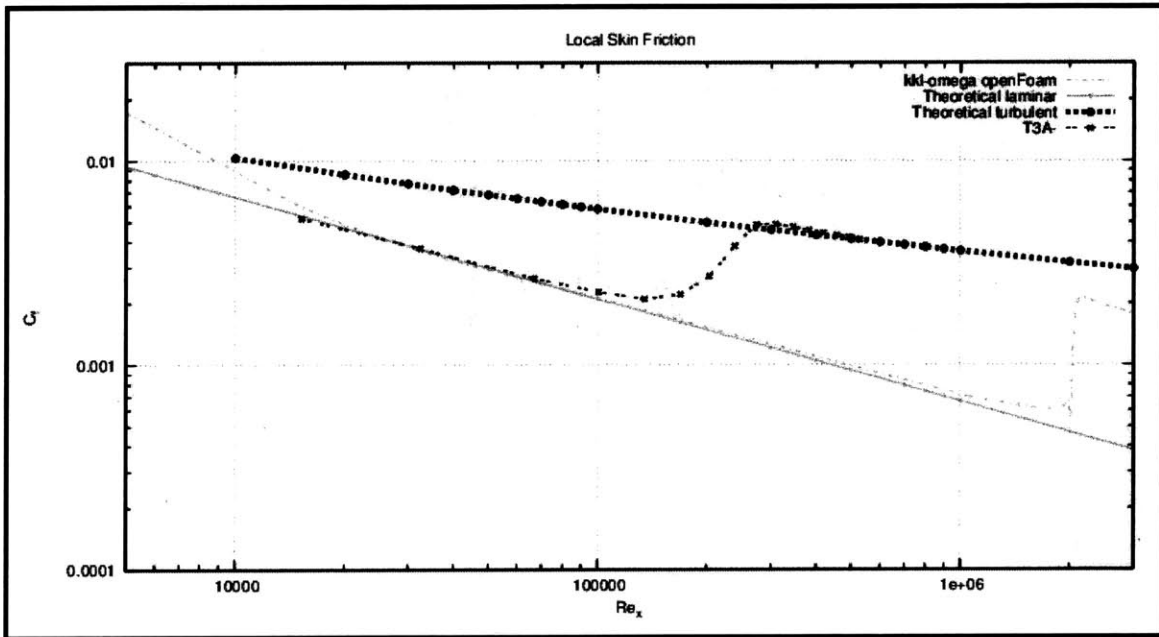
Figure 2-26: Skin Friction vs. Local Reynolds

[25]. It proposed by Jiri Furst [13] that the library should be compiled by using the formulas in the article which was proposed by Walters [26] in 2005. The formulas in paper 2008 are different from the ones in the paper proposed in 2005. The first difference is the f_{INT} formula. The current version of the formula is:

$$f_{INT} = MIN\left(\frac{k_L}{C_{INT}k_{TOT}}, 1\right) \tag{2.100}$$

But in the correct version of the formula, k_t is used instead of k_L. The correct version of the formula is as follows:

$$f_{INT} = MIN\left(\frac{k_T}{C_{INT}k_{TOT}}, 1\right) \tag{2.101}$$

The second difference between those two articles is the damping function f_W. The current version of the formula is as follows:

$$f_W = \frac{\lambda_{eff}}{\lambda_T} \tag{2.102}$$

The right version of the formula is as follows:

$$f_W = \left(\frac{\lambda_{eff}}{\lambda_T}\right)^{2/3} \tag{2.103}$$

The third difference is that the coefficient $C_{\omega 2}$ is not multiplied with F_W^2. The correct version is as follows:

$$C_{\omega 2} = 0.92 f_W^2 \tag{2.104}$$

The last difference is the dissipation formulas. This correction was not applied in the article proposed by Jiri Furst [13]. The current version of the formulas is as follows:

$$D_T = \nu \frac{\partial\sqrt{k_t}}{\partial x_j} \frac{\partial\sqrt{k_T}}{\partial x_j} \tag{2.105}$$

$$D_L = \nu \frac{\partial\sqrt{k_L}}{\partial x_j} \frac{\partial\sqrt{k_L}}{\partial x_j} \tag{2.106}$$

These formulas are multiplied by 2 in the article 2005 and the correct version is as follows:

$$D_T = 2\nu \frac{\partial \sqrt{k_t}}{\partial x_j} \frac{\partial \sqrt{k_T}}{\partial x_j} \tag{2.107}$$

$$D_L = 2\nu \frac{\partial \sqrt{k_L}}{\partial x_j} \frac{\partial \sqrt{k_L}}{\partial x_j} \tag{2.108}$$

Considering these corrections, the modifications are applied to current model in openFoam 2.3.0 and te new library is recompiled in openFoam and exactly the same mesh which was used by Jiri Furst [13] was tested. The case was set up with turbulent intensity 1% and tested with the current version of the model as it can be seen in Figure 2-27. Then, exactly the same case was tested with turbulent intensity 3.043% and it was run with the implemented version of the model. For comparison, ERCOFTAC test case T3A [15] was used.
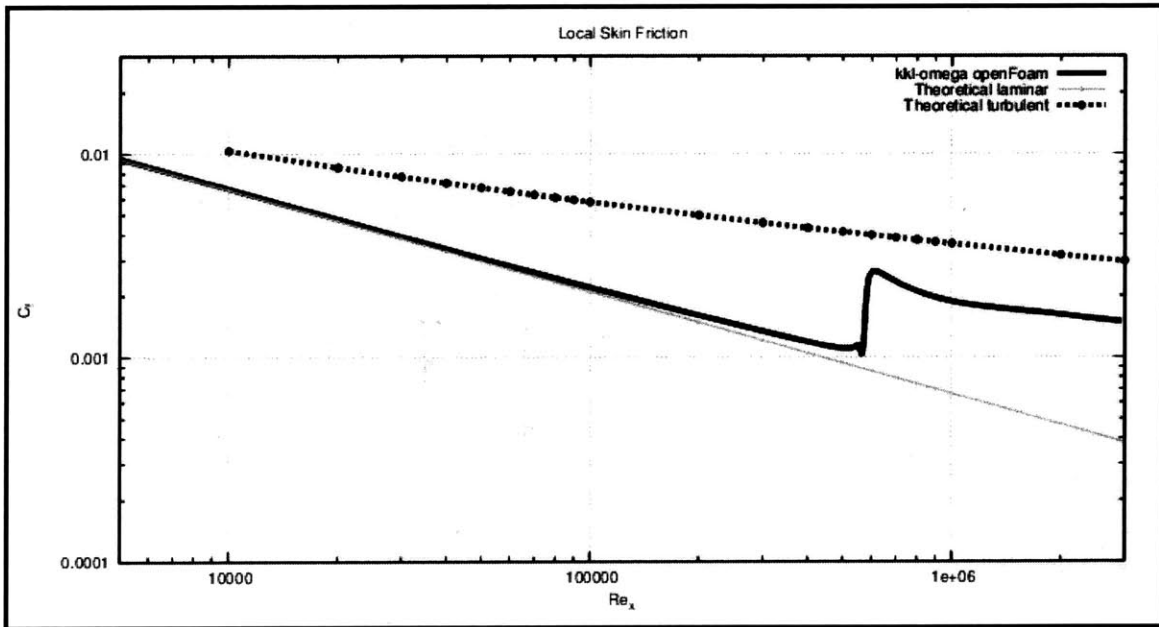
Figure 2-27: Local skin friction plot with the current version of the model

The implemented version of the model provided very close results to te experimental results as it can be seen in Figure 2-28.
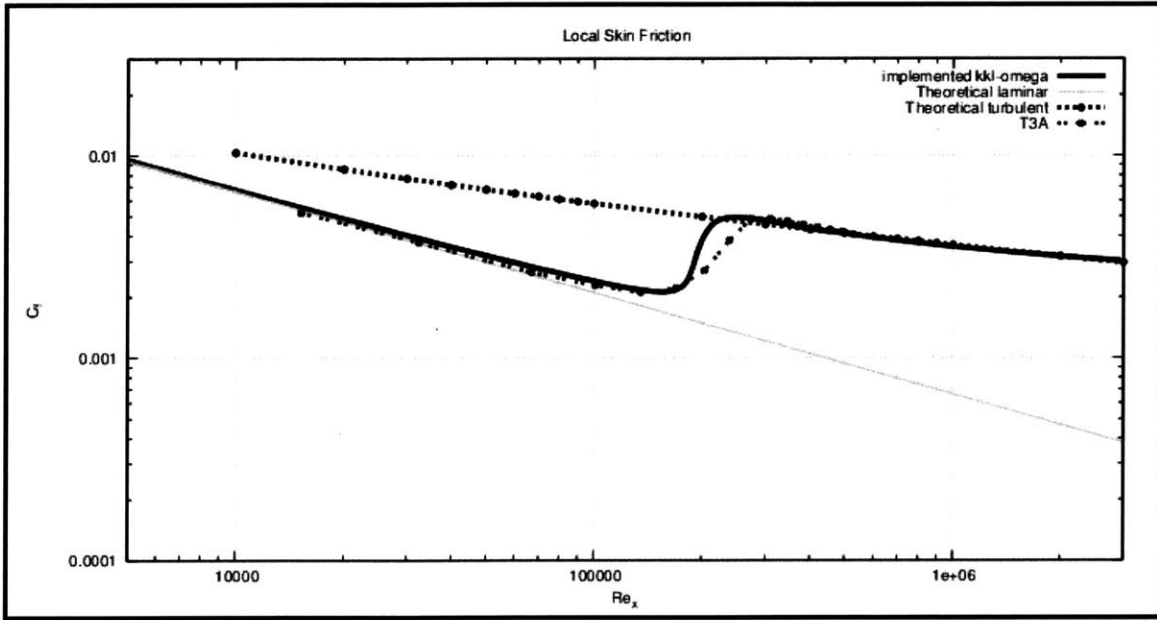
Figure 2-28: Local Skin Friction plot with the implemented model

Cases with Other Meshes

As discussed in the previous parts, the meshes were quite coarse although they provided close results. Therefore, other meshes with lower yplus values to get close results to the Schoenherr value. The yplus was also selected less than 1 for kkl-omega model to be able to observe transition in the friction drag graph. These meshes provided transition from laminar to turbulent regions. The results for Reynolds number 1e5 and 3e6 can be seen in the following table:

yplus at Re:3e6	Cf k-epsilon	Cf k-omega	Cf Spalart-Allmaras	Cf kkl-omega
50	3.720E-03	3.521E-03	3.890E-03	1.058E-03
70	3.706E-03	3.493E-03	3.880E-03	1.198E-03
100	3.709E-03	3.470E-03	3.943E-03	3.745E-03
200	3.737E-03	3.483E-03	3.965E-03	4.139E-03

Table 2.19: The friction coefficient values for different meshes

As it can be seen from the results, the closest results to the Schoenherr value was obtained with the mesh which has the yplus value 30 at Reynolds number 1e5. This was quite a coarse mesh but it provided good results. For the kkl-omega case, different meshes were created for both Reynolds number 1e5 and 3e6 and also the turbulent intensity was chosen as 0.2%. For Reynolds number 1e5, the results for

different yplus values can be seen in the following table:

yplus	at Re:1e5	Blausius Cf value	Cf kkl-omega
0.5	1.00E+05	4.20E-03	4.85E-03
1	1.00E+05	4.20E-03	4.82E-03
3	1.00E+05	4.20E-03	4.71E-03
5	1.00E+05	4.20E-03	4.70E-03
7	1.00E+05	4.20E-03	4.95E-03

Table 2.20: The friction coefficient values for different meshes

Velocity Profiles in Boundary Layer

The last part of the post-processing was plotting the velocity profile at the trailing of the wall. Firstly, the theoretical profile where $\frac{y}{\delta}$ is at y axis and $\frac{u}{U}$ is at the x axis was plotted by using the following formulas [21]:

$$\frac{u}{U} = \left(\frac{y}{\delta}\right)^{\left(\frac{1}{7}\right)}$$

where y was the distance to the center of each cell.

For the boundary layer thickness, the following formula was used [6]:

$$\frac{\delta^*}{x} = 0.373(Re_x)^{\left(-\frac{1}{5}\right)}$$

The calculations were made for the trailing edge of the plate. Therefore, x was selected to be equal to 1.

Then the velocity profile by using the values which openFoam provided was plotted. For this part, initially the sampleDict file was created. It was not possible to define a single point in sampleDict file. Therefore, the starting point was selected to be 0.99 and the end point was selected to be 1. Also, the distances to the center of each cell was calculated as an input in sampleDict file. After the simulations were done, sample command was run in terminal and openFoam calculated the local velocity values for each cell in the boundary layer. The velocity profiles were plotted for three different Reynolds number which are 1e5, 3e6 and 1e7. The plots which can

be seen in Figure: 1-12, Figure: 1-13, Figure: 1-14 were created for each turbulence model with $y^+=30$ at Re:1e5 ($y^+=1414$ at Re:3e6).
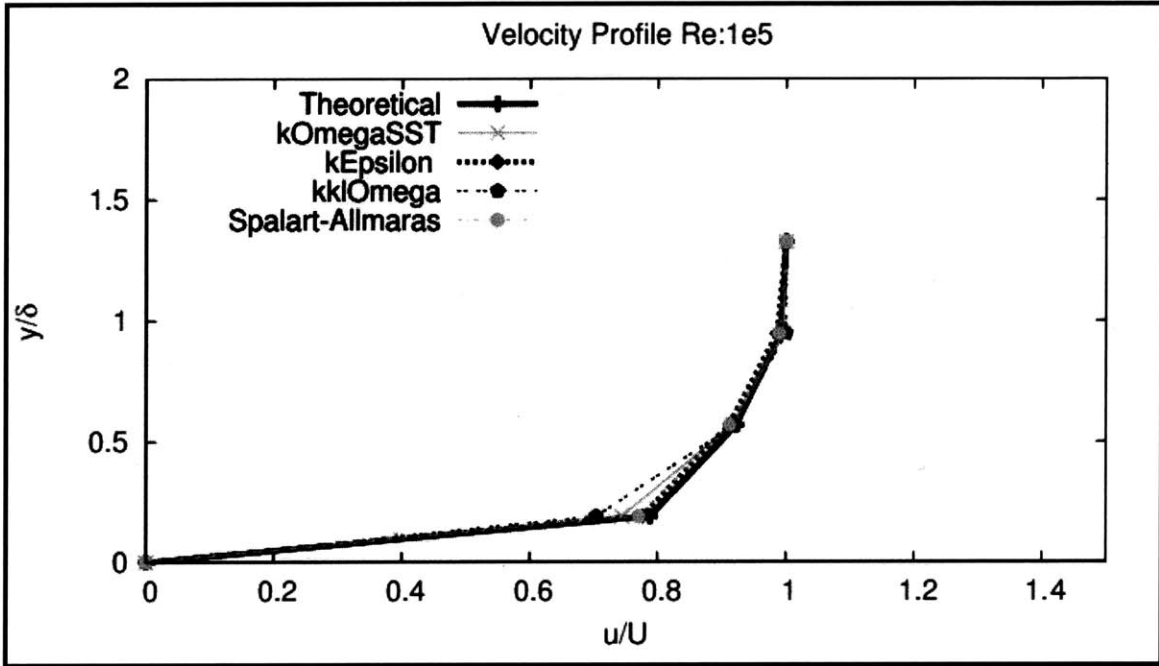
Figure 2-29: Velocity profile at the trailing edge of the plate
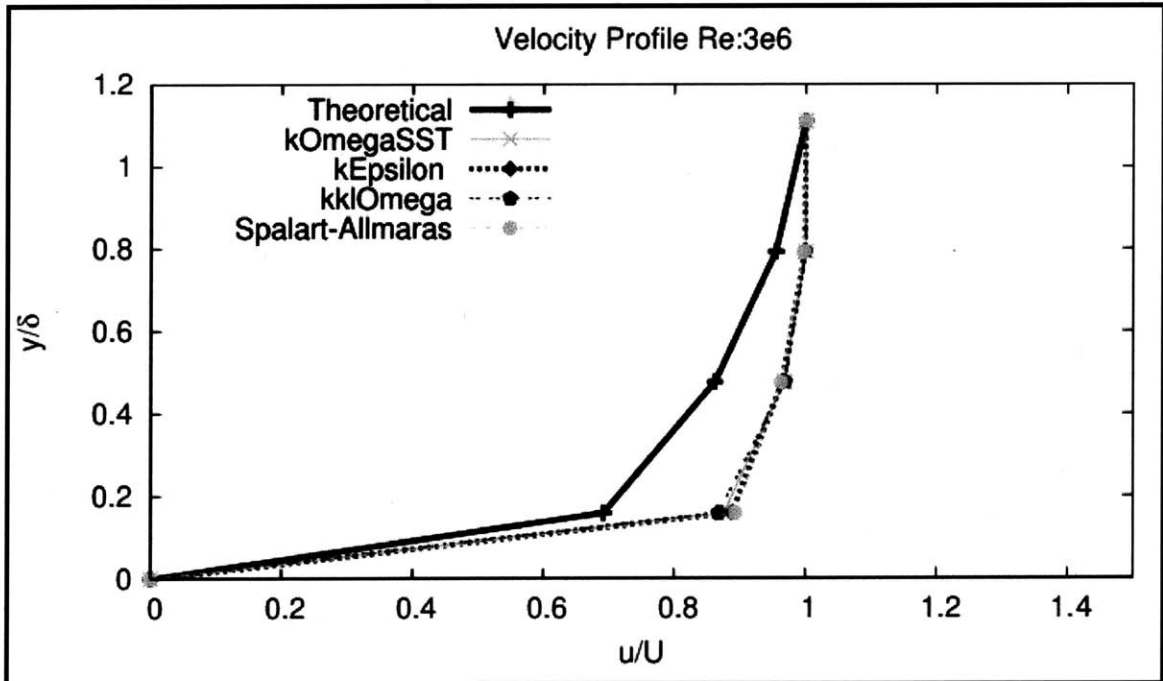
Figure 2-30: Velocity profile at the trailing edge of the plate
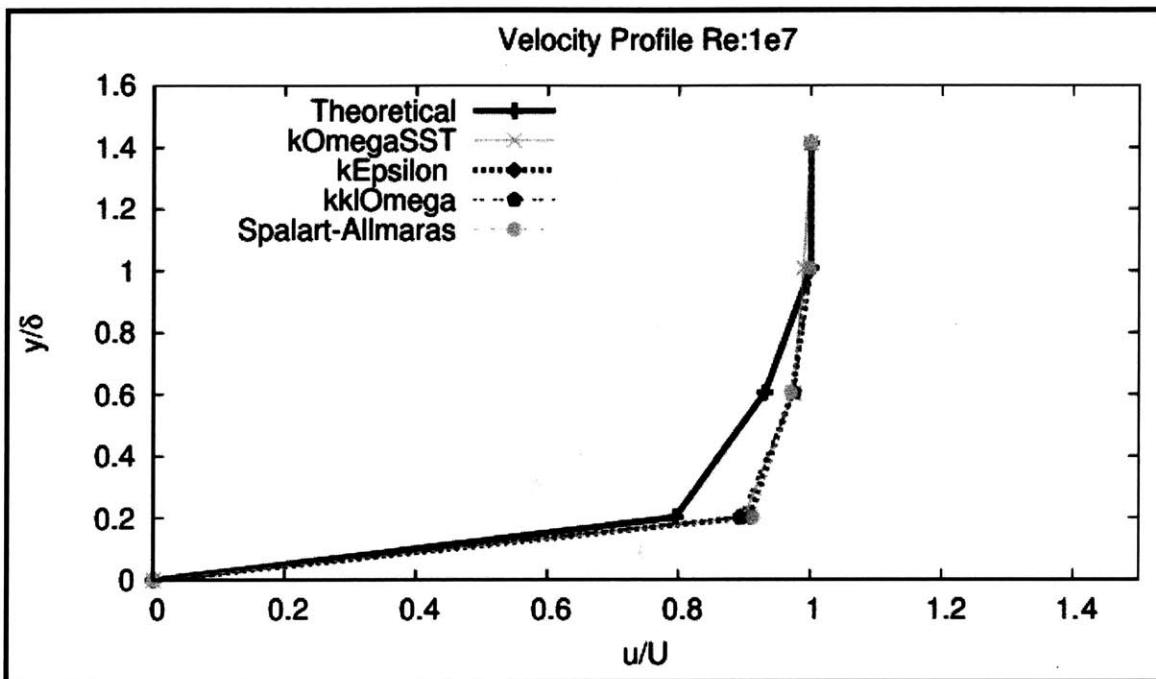
Figure 2-31: Velocity profile at the trailing edge of the plate

Mesh Sensitivity Analysis
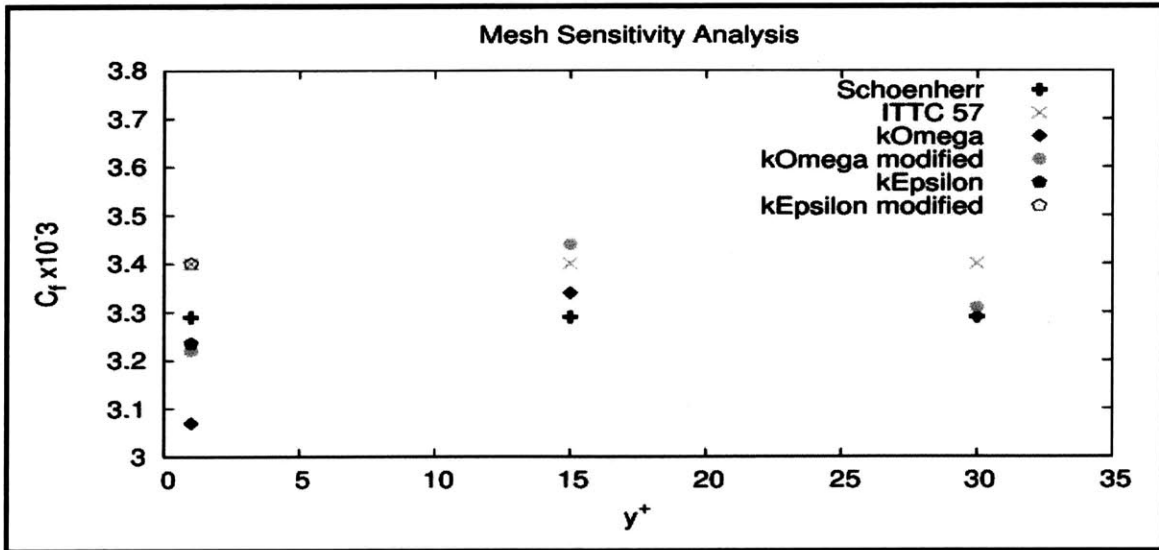
Figure 2-32: Mesh sensitivity analysis at Re:5e5
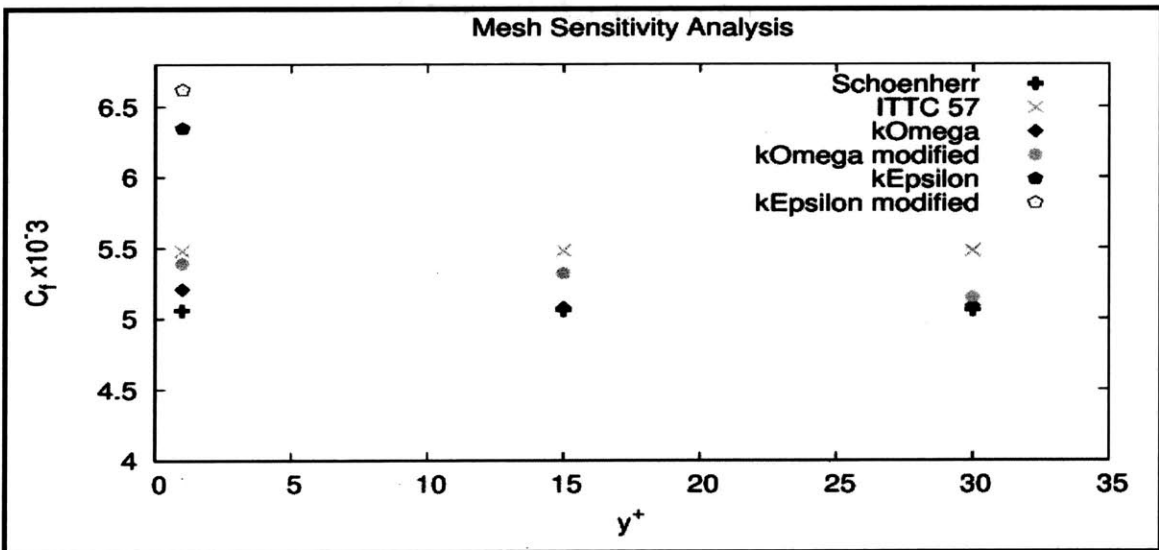
Figure 2-33: Mesh sensitivity analysis at Re:6e6

Chapter 3

NACA $65_1 - 213$ a=0.5 Airfoil Test Case

3.1   History of NACA Airfoils

NACA (National Advisory Commitee for Aeronautics) airfoils were tested in the Langley variable-density tunnel in 1929 and this was the starting point of the historical development of NACA airfoils. This family of airfoil was named NACA 4-digit hydrofoils and all of them had the same thickness distribution. Then another series of airfoils were created with the same thickness distribution but with different maximum camber location. These airfoils were called 5-digit airfoils. Then new methods were developed and NACA 6-digit series airfoils were created. The method aimed to combine the mean line with thickness distributions. The combining process can be seen in Figure 3-1.

3.2   Numbering System of NACA Airfoils

NACA 6-series of airfoils have 6 digits and each digit has a meaning as described below for NACA $65_1 - 213$ a=0.5 [8] :

6= Designation of the series

5 = chordwise position of minimum pressure in tenths of the chord behind the leading
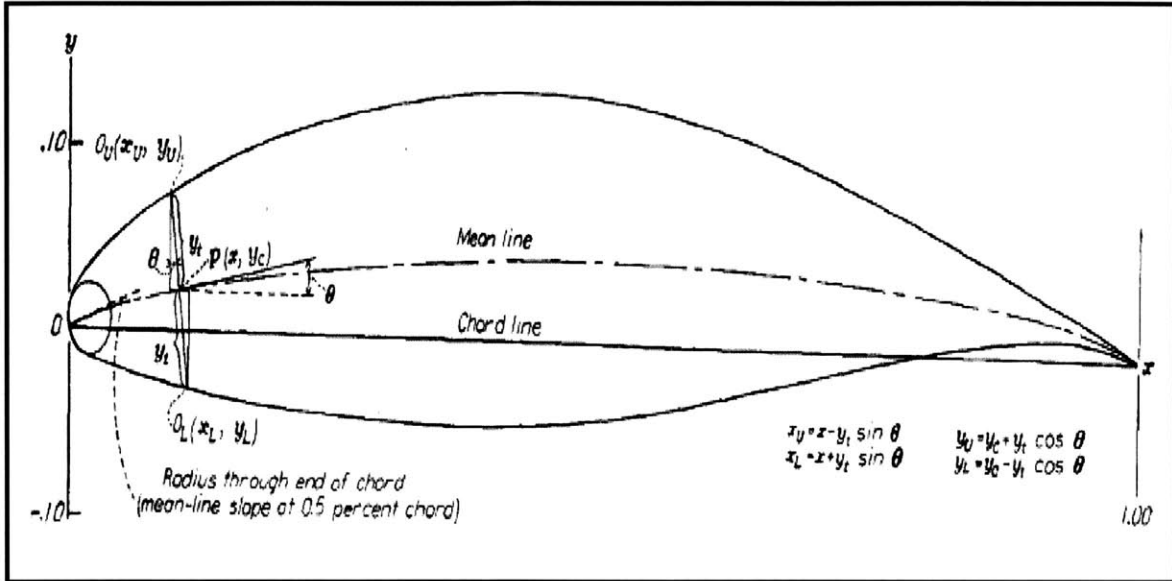
Figure 3-1: Combining the mean line and thickness distributions [19]

edge for the basic symmetrical section at zero lift

1= Gives the range of lift coefficient in which favorable pressure gradients exist on both surfaces

2 = Design lift coefficient in tenths

13 = Airfoil thickness in percent of chord

0.5=type of the mean line used

3.3 Mesh Generation and Case Set up

The geometry of the airofoil was created in AutoCAD by using the points provided in the NACA report. The geometry of NACA $65_1 - 213$ a=0.5 airfoil can be seen in Figure 3-2. The points were exported as a .dat files and the x and y coordinates was separated from each other with commas. Then, pline command was used in AutoCAD and the points were copied from the .dat file and pasted in AutoCAD command line. The point (0.00132,0.0075) was an extreme point therefore it was eliminated from the drawing. Then, curve fit command was applied to the airfoil geometry in AutoCAD. After that, the airofil was divided into 830 points and the points were extracted from

AutoCAD by using the EATTEXT command. Then, the points were used in GMSH to create the mesh. Three different meshes were created with $y^+ = 1$ and 250k cells at Reynolds number 3e6 with angles of attack 0, 2 and 6 as the meshes can be seen in the following figurres. After the meshes were created, the cases for both kOmegaSST and kkl-omega were set up with steady state. The files of kkl-omega case are provided in appendix A.
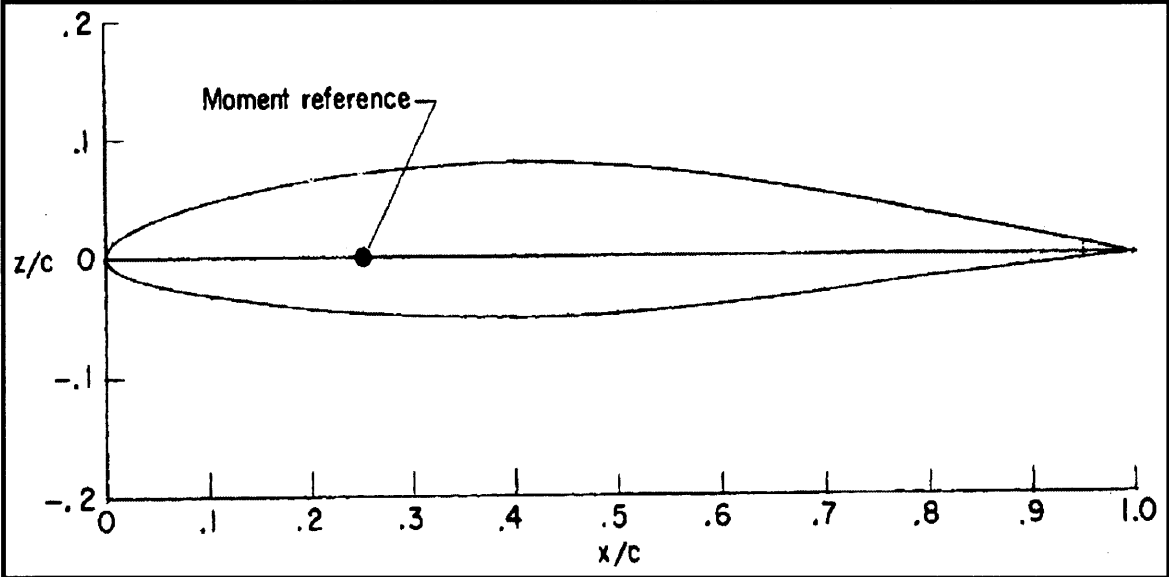
Figure 3-2: Geometry of NACA $65_1 - 213$ a=0.5 airfoil [8]
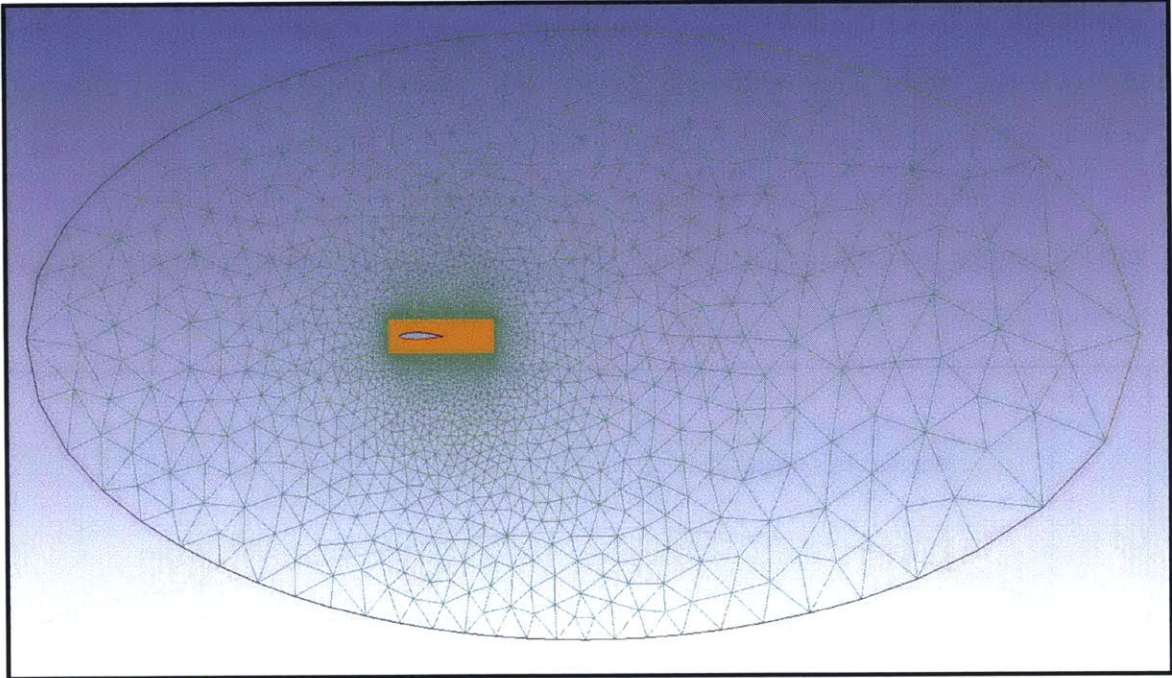
Figure 3-3: NACA $65_1 - 213$ a=0.5 airfoil computational domain
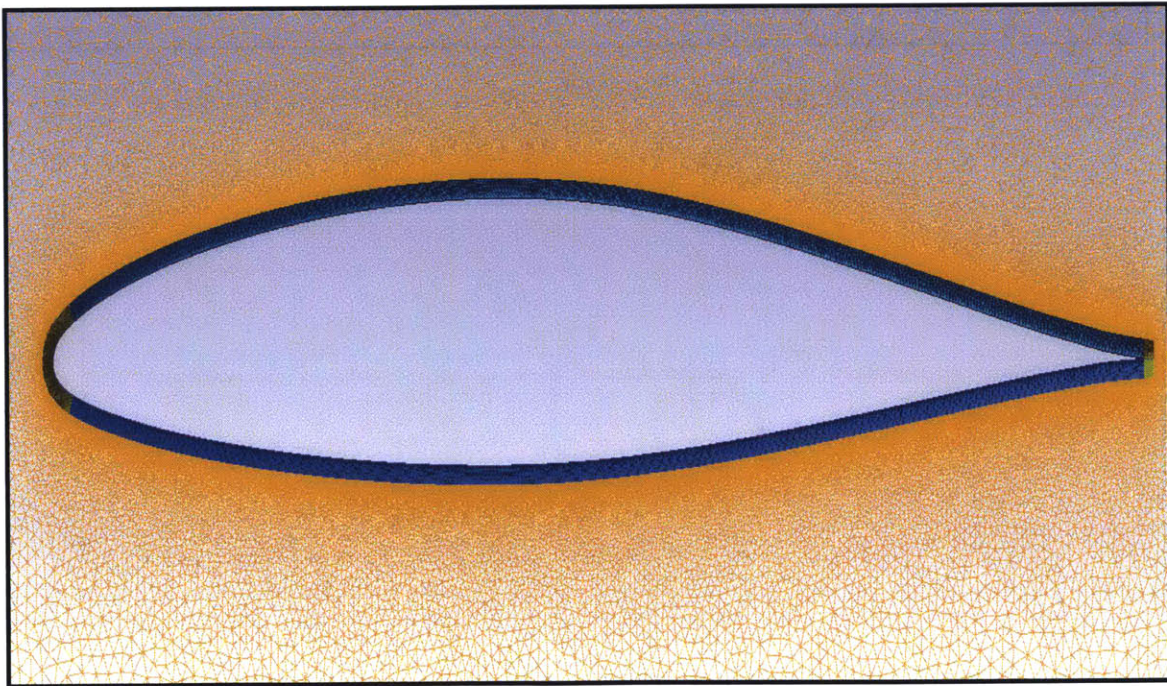
Figure 3-4: NACA $65_1 - 213$ a=0.5 airfoil grid structure in the near-wall region
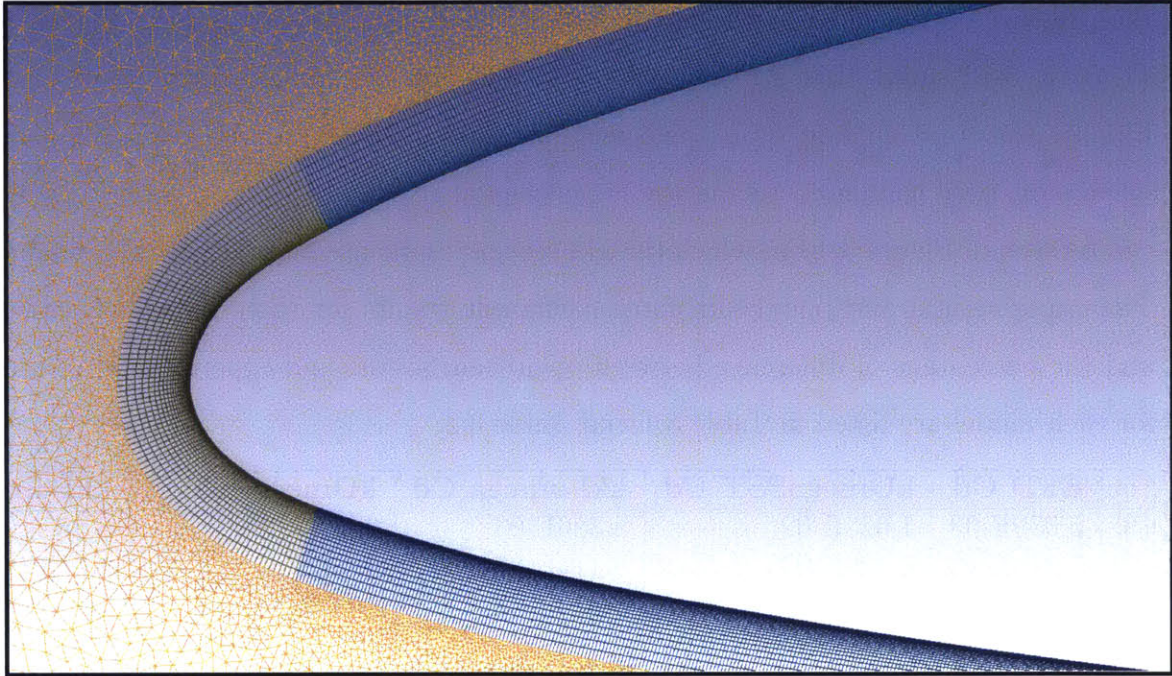
Figure 3-5: NACA $65_1 - 213$ a=0.5 airfoil grid structure at the leading edge

3.4 Analysis of the Results

In this section, the results of the numerical simulation of NACA $65_1 - 213$ a=0.5 with both kOmegaSST and kkl-omega models was analyzed. For comparison, XFOIL results and experimental results [8] were used. There was no information available for the turbulent intesnity therefore it was assumed to be 1%. The total chord length was 0.6063 m and the initial freesream values and boundary conditionscan be seen in the files presented in Appendix A. First all, convergence study was investigated and Figure:3-6, 3-7, 3-8 show that the model converges at more or less 5500 iterations. After that, the local skin frictions were plotted along the foil to determine the transition point. Figures 3-9 3-10, 3-11 show that XFOIL and openFoam results follow the same line on both laminar and fully turbulent regions. The only difference between those results is the onset of the tranisition. Transitiion point depend on the turbulent intensity Tu defined at the inlet region. The difference between those programs might have occured due to this reason. There were no experimental results available for the local skin friction and the onset point of the tranisition. Then,

the pressure coefficient over the foil was plotted for each angle of attack. Figure 3-11, 3-12, 3-12 shows that XFOIL and openFoam provides almost the same results. Finally, the total friction drag coefficient and lift coefficient results obtained from openFoam were compared to the the experimental results. The experimetal results can be seen in Figure 3-13 in which the kkl-omega results are plotted with red points. The experimental and openFoam force coefficient results for three angles of attack and the percentage of difference between openFoam results and experimental results for each model are listed in Table 3.1 and Table 3.2.

α	EFD Cd	kOmegaSST Cd	kkl-omega Cd	kOmegaSST %	kkl-omega %
0	5.039E-03	1.025E-02	5.386E-03	103.43%	6.88%
2	7.319E-03	8.309E-03	6.114E-03	13.52%	16.47%
6	1.100E-02	3.281E-02	1.256E-02	198.30%	14.15%

Table 3.1: Comparison of the drag coefficient results for NACA $65_1 - 213$ a=0.5 airfoil

α	EFD Cl	kOmegaSST Cl	kkl-omega Cl	kOmegaSST %	kkl-omega %
0	1.400E-01	2.094E-01	1.505E-01	49.59%	7.52%
2	3.850E-01	2.445E-01	3.788E-01	36.50%	1.61%
6	8.1690E-01	2.029E-01	8.090E-01	74.57%	0.97%

Table 3.2: Comparison of the lift coefficient results for NACA $65_1 - 213$ a=0.5 airfoil
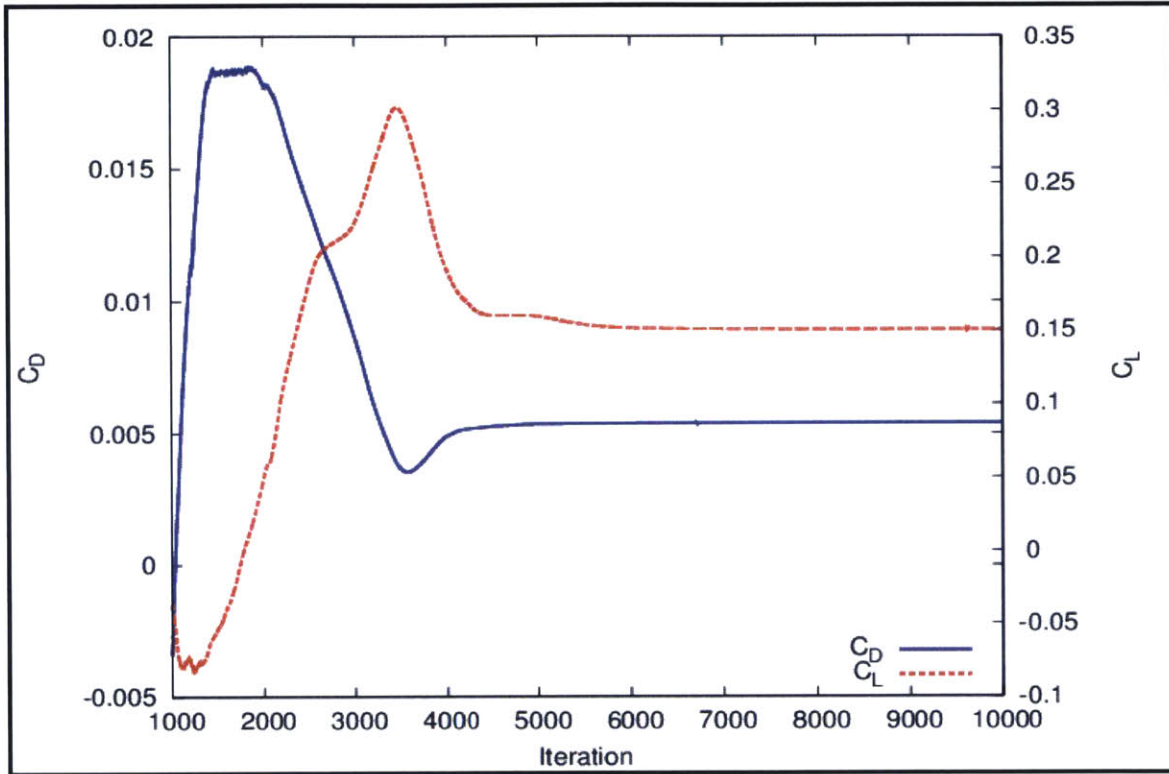
Figure 3-6: NACA $65_1 - 213$ a=0.5 airfoil $\alpha = 0$ convergence
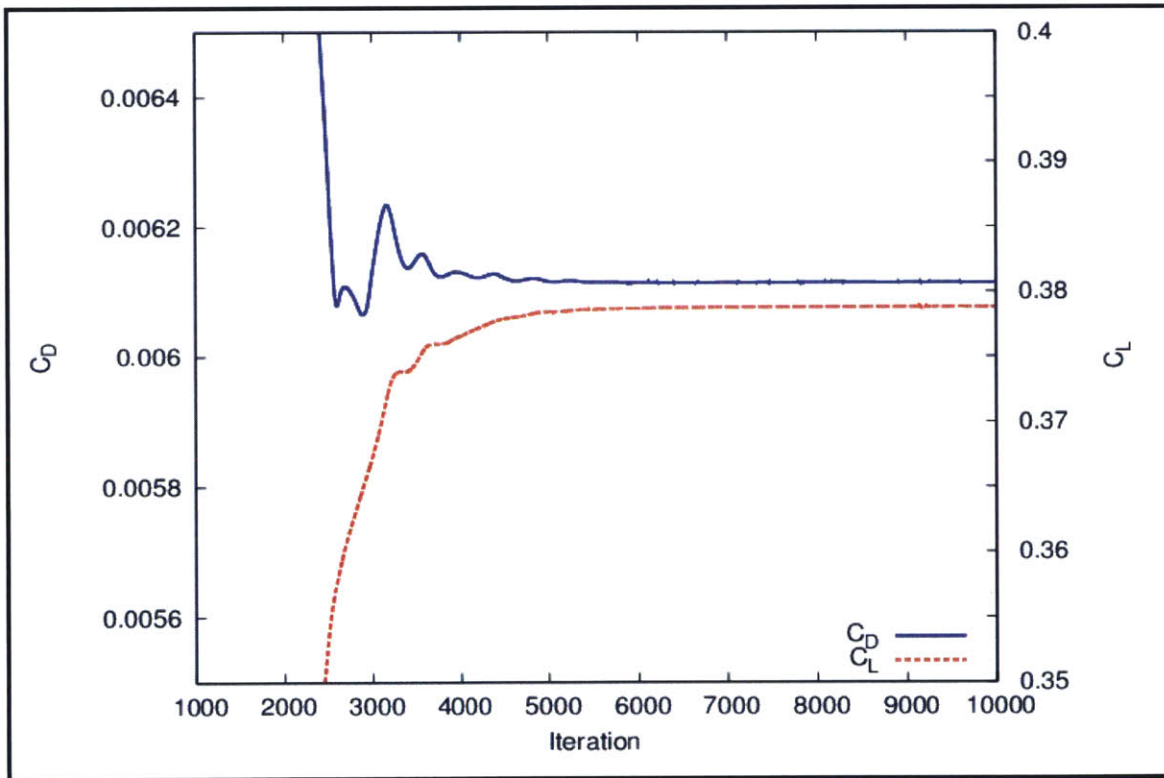
Figure 3-7: NACA $65_1 - 213$ a=0.5 airfoil $\alpha = 2$ convergence
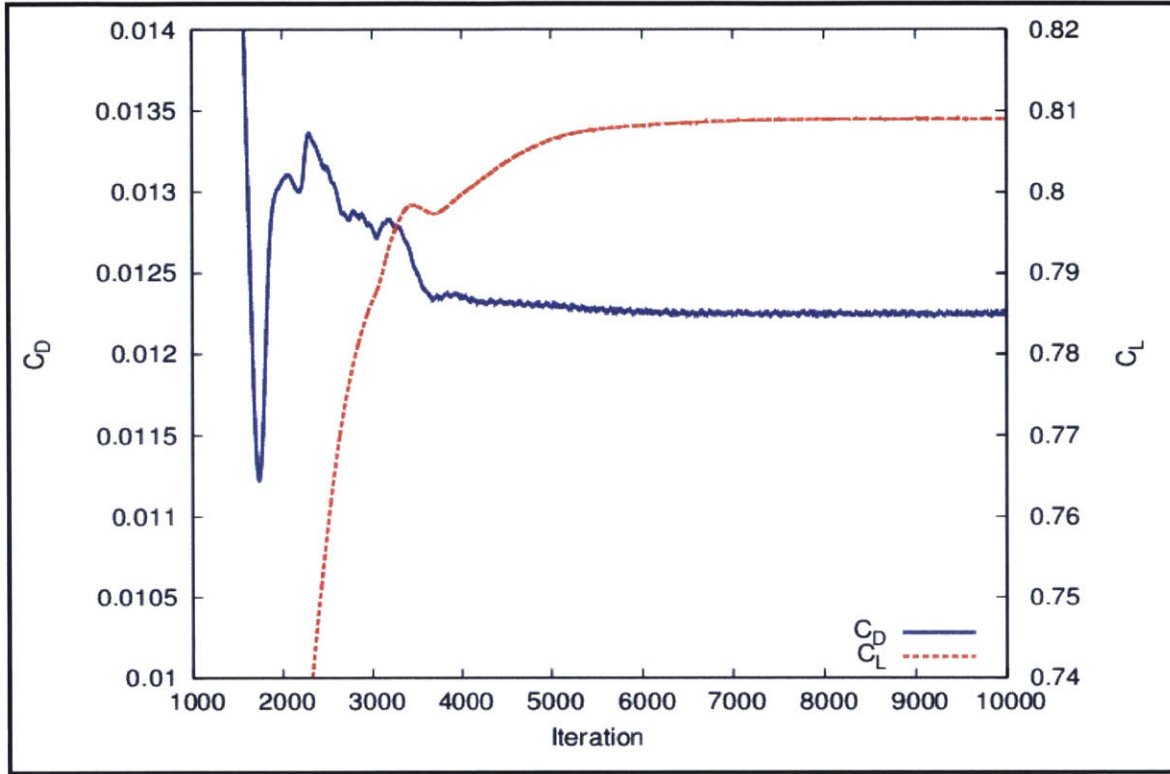
Figure 3-8: NACA $65_1 - 213$ a=0.5 airfoil $\alpha = 6$ convergence
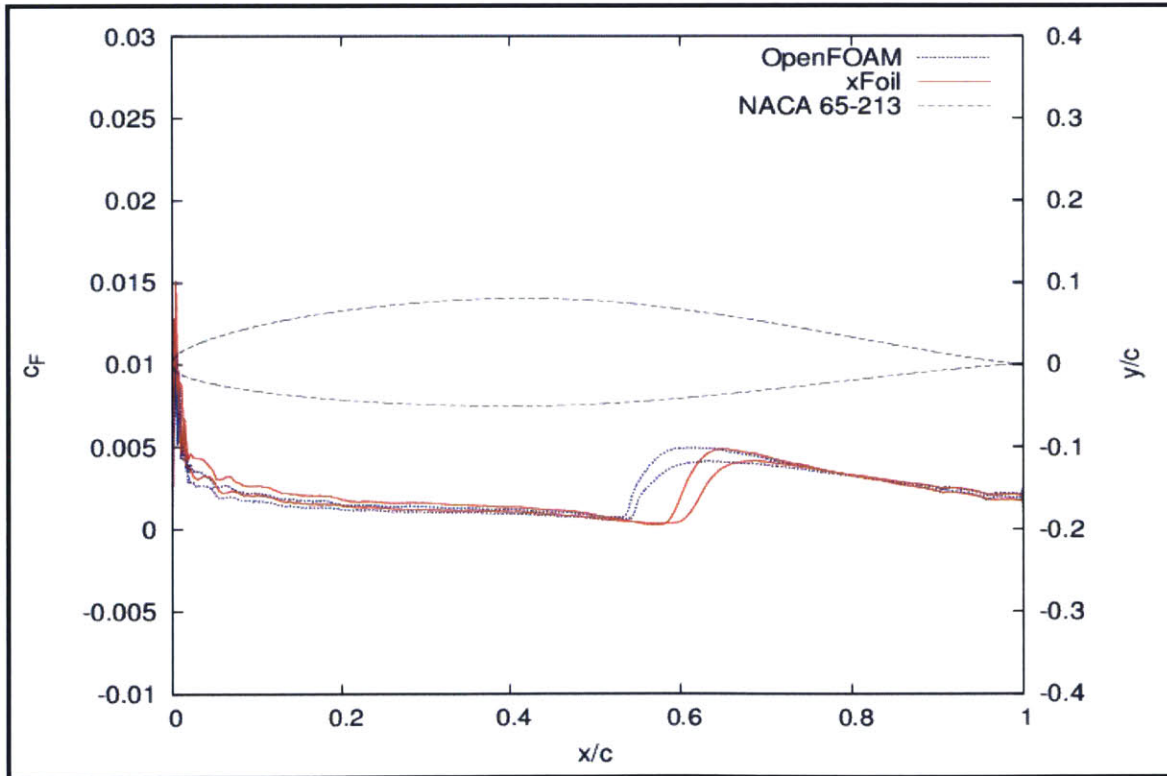
Figure 3-9: NACA $65_1 - 213$ a=0.5 airfoil $\alpha = 0$ local skin friction plot
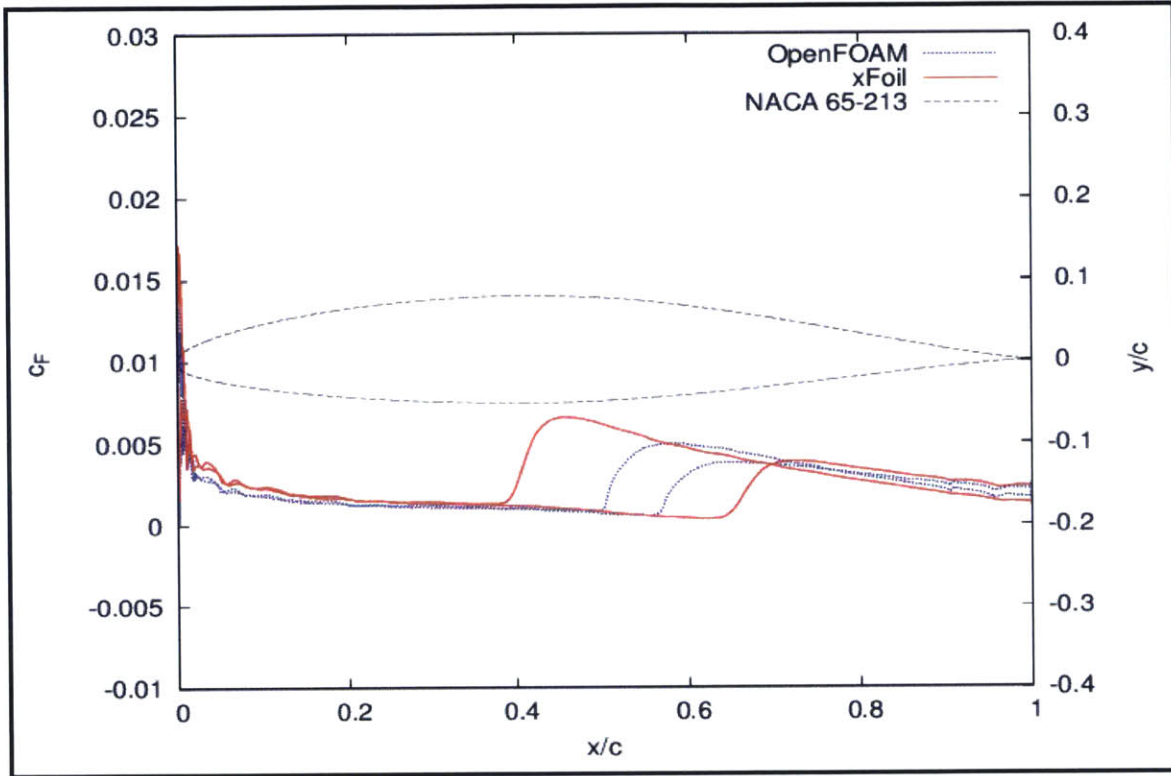
Figure 3-10: NACA $65_1 - 213$ a=0.5 airfoil $\alpha = 2$ local skin friction plot
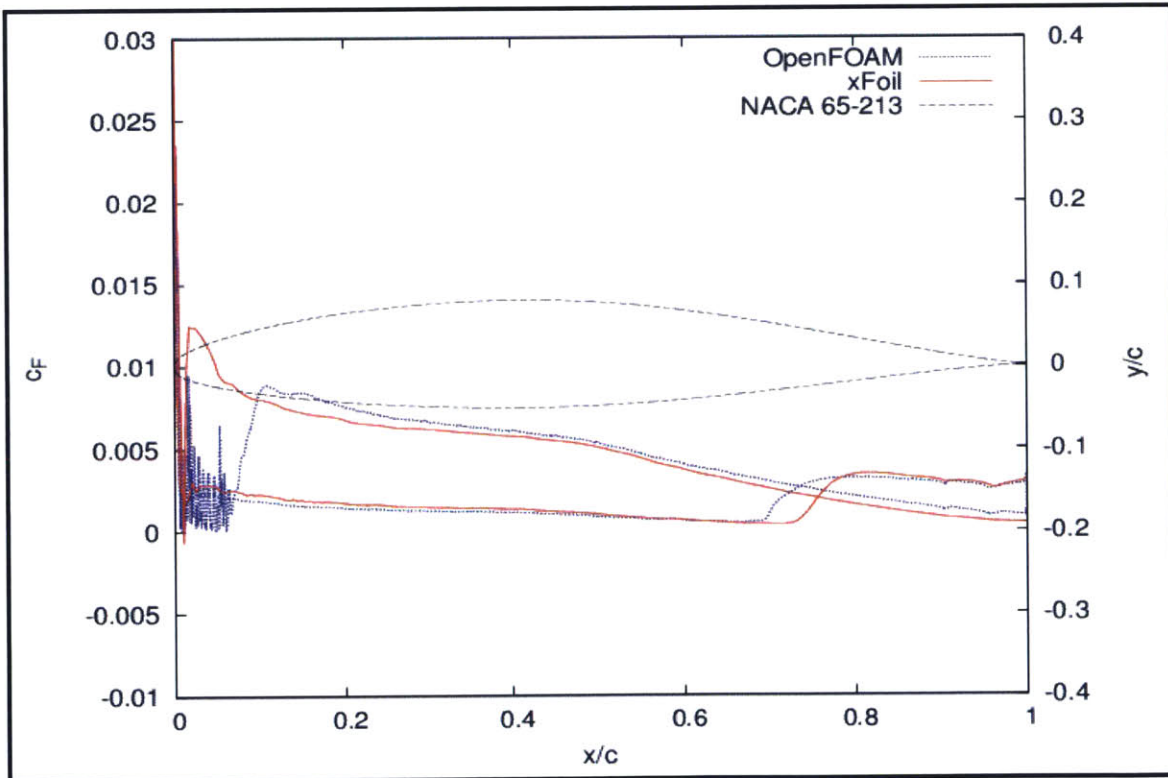
Figure 3-11: NACA $65_1 - 213$ a=0.5 airfoil $\alpha = 6$ local skin friction plot

85
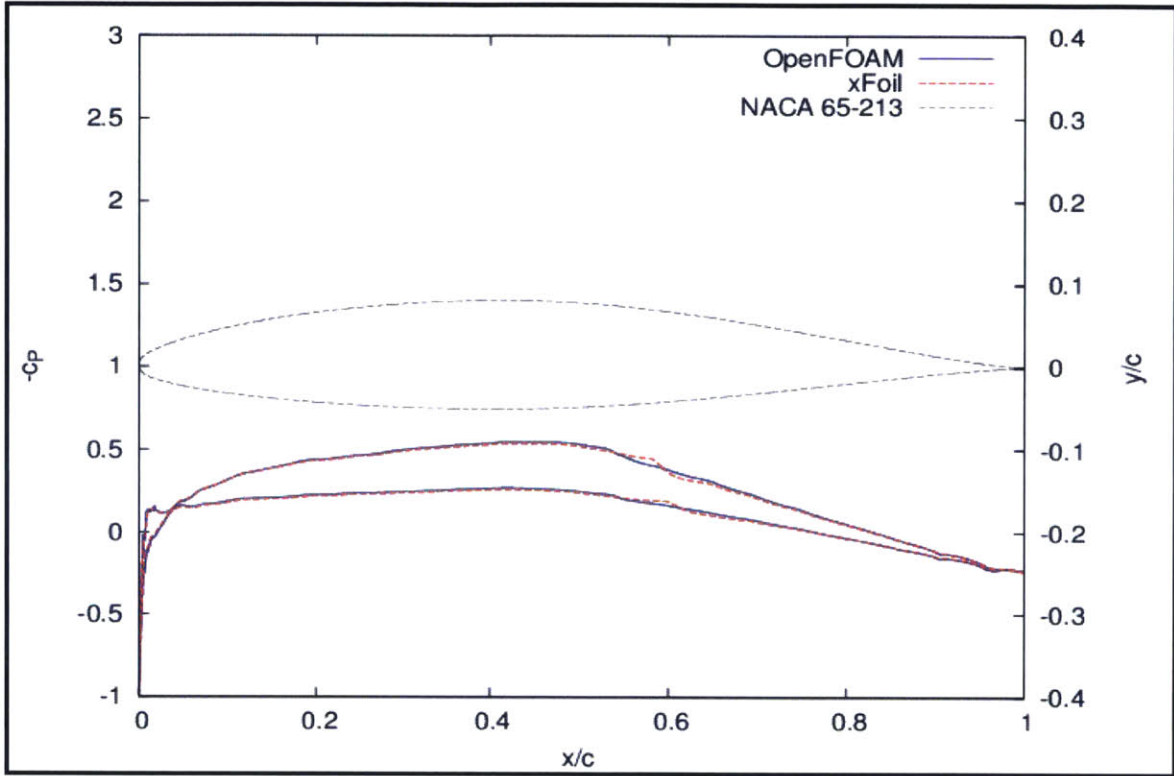
Figure 3-12: NACA $65_1 - 213$ a=0.5 airfoil $\alpha = 0$ pressure coefficient plot
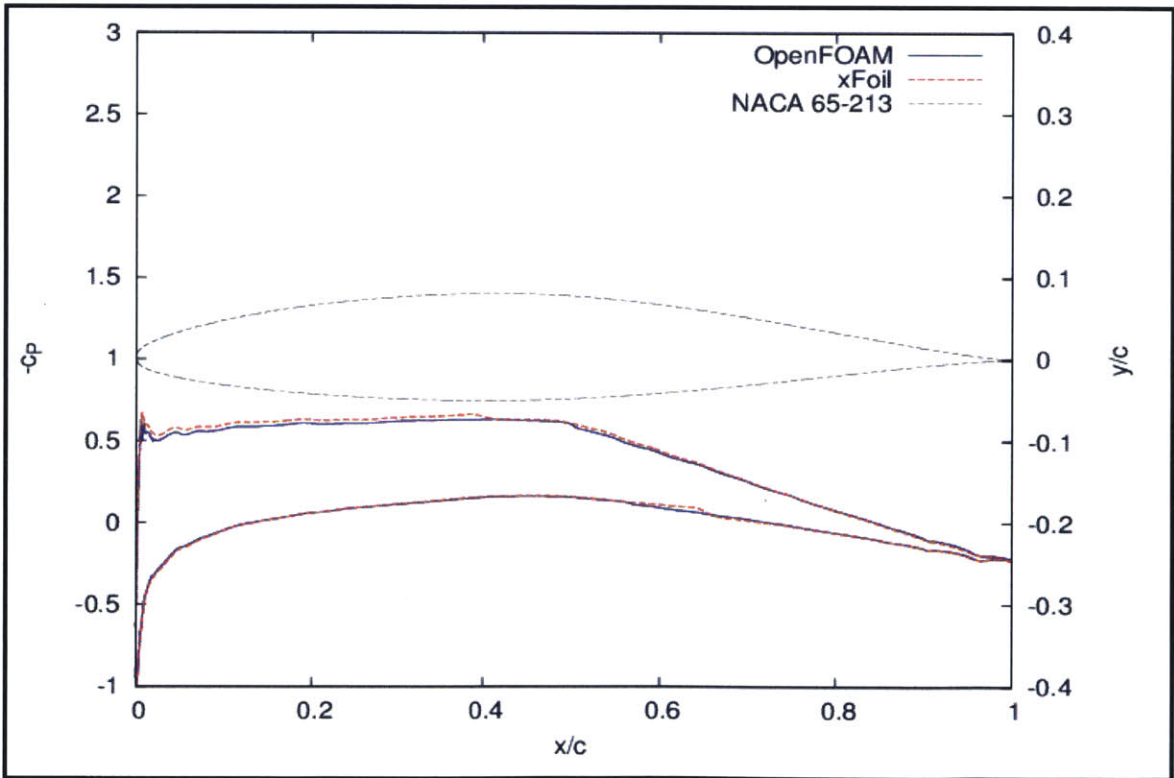
Figure 3-13: NACA $65_1 - 213$ a=0.5 airfoil $\alpha = 2$ pressure coefficient plot
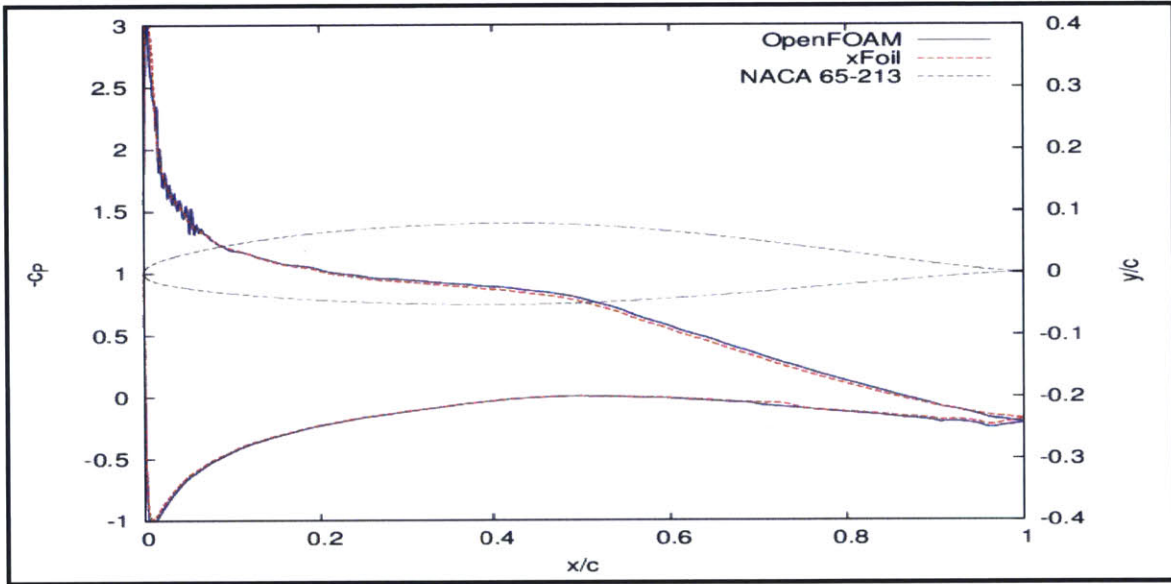
Figure 3-14: NACA $65_1 - 213$ a=0.5 airfoil $\alpha = 6$ pressure coefficient plot
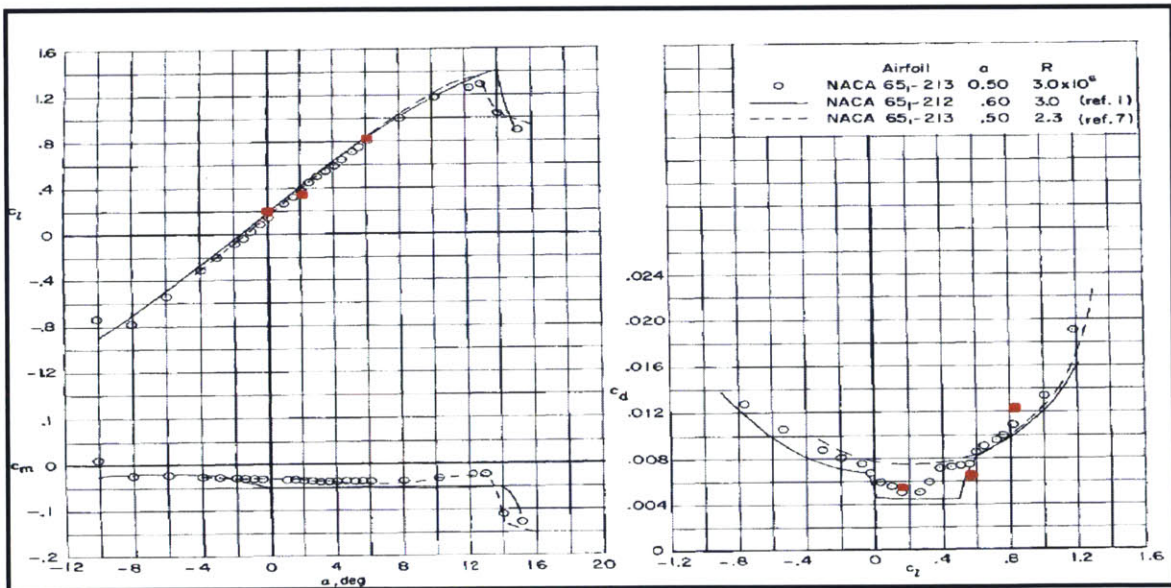
Figure 3-15: NACA $65_1 - 213$ a=0.5 airfoil experimental and openFoam kkl-omega model results [8]

Chapter 4

Conclusions

The key of this research is the implementation of a new kkl-omega model in openFoam. The implemented version of the model provides right transition and good drag coefficient results for the flat plate which the current version is not able to do. Our version also gives good results for the drag coefficient and lift coefficient of NACA $65_1 - 213$ a=0.5 hydrofoil. Furthermore, the kOmegaSST model cannot estimate right values for NACA $65_1 - 213$ a=0.5 hydrofoil lift and drag coefficients. From pressure coefficient standpoint, openFoam results and Xfoil results are nearly identical. From local skin friction coefficient standpoint, XFOIL and openFoam results follow more or less the same line in laminar and fully turbulent regions. The only difference is onset of the transition point. This difference might be caused by the way of using the turbulent intensity in each program. OpenFoam drag coefficient and lift coefficient results are very close to the experimental results but the turbulet intensity value that was used in the experiments is not provided in the experimental results. We assume that turbulent intensity Tu=1% was used in the experiments. We propose in this thesis that the implemented version of the kkl-omega model will provide very close results for each Reynolds number and angle of attack for any hydrofoil.

For the future work, grids with different resolution and angles of attack should be tested and compared to the experimental results in order to evaluate the effect of the grid resolution and performance of the solvers. Also, in this thesis, only the 2-D case is considered. So, the 3-D case including the cavitation and free surface effects

should also be tested for higher fidelity.

Appendix A

OpenFoam Directory Files

A.1 0 (Time) Directory

A.1.1 kl file

```
/*--------------------------------*- C++ -*----------------------------------*\
| =========                 |
|
| \\      /  F ield         | OpenFOAM: The Open Source CFD Toolbox
|
|  \\    /   O peration     | Version:  2.3.0
|
|   \\  /    A nd           | Web:      www.OpenFOAM.org
|
|    \\/     M anipulation  |
|
\*---------------------------------------------------------------------------*/
FoamFile
{
    version     2.0;
    format      ascii;
    class       volScalarField;
    location    "0";
```

```
    object        k1;
}
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //

dimensions        [ 0 2 -2 0 0 0 0 ];

internalField    uniform 3.672e-3;

boundaryField
{
    domain
    {
        type            inletOutlet;
        inletValue      $internalField;
    }
    foil
    {
        type            fixedValue;
        value           uniform 0;
    }
    defaultFaces
    {
        type            empty;
    }
}


// ************************************************************************* //
```

A.1.2   kt File

```
/*--------------------------------*- C++ -*----------------------------------*\
| =========                 |
|
| \\      / F ield          | OpenFOAM: The Open Source CFD Toolbox
|
| \\    / O peration        | Version:  2.3.0
|
|  \\  / A nd               | Web:      www.OpenFOAM.org
|
|   \\/    M anipulation  |
|
\*---------------------------------------------------------------------------*/
FoamFile
{
    version     2.0;
    format      ascii;
    class       volScalarField;
    location    "0";
    object      kt;
}
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //

dimensions      [ 0 2 -2 0 0 0 0 ];

internalField   uniform 3.672e-3;

boundaryField
{
    domain
    {
        type            inletOutlet;
        inletValue      $internalField;
    }
```

```
    foil
    {
        type            fixedValue;
        value           uniform 0;
    }
    defaultFaces
    {
        type            empty;
    }
}



// ************************************************************************* //
```

A.1.3   omega File

```
/*--------------------------------*- C++ -*----------------------------------*\
| =========                 |
|
| \\      /  F ield         | OpenFOAM: The Open Source CFD Toolbox
|
|  \\    /   O peration     | Version:  2.1.0
|
|   \\  /    A nd           | Web:      www.OpenFOAM.org
|
|    \\/     M anipulation  |
|
\*---------------------------------------------------------------------------*/
FoamFile
{
    version     2.0;
    format      ascii;
    class       volScalarField;
    object      omega;
}
```

```
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //


dimensions      [0 0 -1 0 0 0 0];


internalField   uniform 24.0837;


boundaryField
{
    domain
    {
        type            inletOutlet;
        inletValue      $internalField;
    }
    foil
    {
        type            zeroGradient;
    }

    defaultFaces
    {
        type            empty;
    }
}



// ************************************************************************* //
```

A.1.4 nut File

```
/*--------------------------------*- C++ -*----------------------------------*\
| =========                 |                                                 |
|                                                                             |
| \\      /  F ield         | OpenFOAM: The Open Source CFD Toolbox           |
|                                                                             |
|  \\    /   O peration     | Version:  2.1.0                                 |
|                                                                             |
|   \\  /    A nd           | Web:      www.OpenFOAM.org                      |
|                                                                             |
|    \\/     M anipulation  |                                                 |
|                                                                             |
\*---------------------------------------------------------------------------*/
FoamFile
{
    version     2.0;
    format      ascii;
    class       volScalarField;
    location    "0";
    object      nut;
}
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //

dimensions      [0 2 -1 0 0 0 0];

internalField   uniform  0;

boundaryField
{
    domain
    {
        type            calculated;
        inletValue      $internalField;
    }
```

```
foil
{
    type            fixedValue ;
    value           uniform 0;
}

defaultFaces
{
    type            empty;
}
}


// ************************************************************************* //
```

A.1.5   p File

```
/*--------------------------------*- C++ -*----------------------------------*\
| =========                 |
|
| \\      /  F ield         | OpenFOAM: The Open Source CFD Toolbox
|
|  \\    /   O peration     | Version:  1.7.1
|
|   \\  /    A nd           | Web:      www.OpenFOAM.com
|
|    \\/     M anipulation  |
|
\*---------------------------------------------------------------------------*/
FoamFile
{
    version     2.0;
    format      ascii;
    class       volScalarField;
    object      p;
}
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //

dimensions      [0 2 -2 0 0 0 0];

internalField   uniform 0;

boundaryField
{
    domain
    {
        type            outletInlet;
        outletValue $internalField;
    }
    foil
```

```
    {
        type              zeroGradient;
    }


    defaultFaces
    {
        type              empty;
    }
}


// ************************************************************************* //
```

A.1.6   U File

```
/*--------------------------------*- C++ -*----------------------------------*\
| =========                 |
|
| \\      / F ield          | OpenFOAM: The Open Source CFD Toolbox
|
|  \\    /   O peration      | Version:  1.7.1
|
|   \\  /    A nd            | Web:      www.OpenFOAM.com
|
|    \\/     M anipulation   |
|
\*---------------------------------------------------------------------------*/
FoamFile
{
    version     2.0;
    format      ascii;
    class       volVectorField;
    object      U;
}
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //
```

```
dimensions      [0 1 -1 0 0 0 0];


internalField   uniform (0  4.945031784068529 0.172684315060823);


boundaryField
{
    domain
    {
        type            inletOutlet;
        inletValue      $internalField;
    }
    foil
    {
        type            fixedValue;
        value           uniform (0 0 0);
    }

    defaultFaces
    {
        type            empty;
    }
}


// ************************************************************************* //
```

A.2   System Directory

A.2.1   controlDict File

```
/*--------------------------------*- C++ -*----------------------------------*\
| =========                 |
|                           |
| \\      /  F ield         | OpenFOAM: The Open Source CFD Toolbox
|                           |
|  \\    /   O peration     | Version:  1.7.0
|                           |
|   \\  /    A nd           | Web:      www.OpenFOAM.com
|                           |
|    \\/     M anipulation  |
|                           |
\*---------------------------------------------------------------------------*/
FoamFile
{
    version     2.0;
    format      ascii;
    class       dictionary;
    location    "system";
    object      controlDict;
}
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //

application     simpleFoam;

startFrom       latestTime;

startTime       0;

stopAt          endTime;

endTime         50;
```

```
deltaT            0.01;

writeControl     adjustableRunTime;

writeInterval    5;

purgeWrite       0;

writeFormat      ascii;

writePrecision   7;

writeCompression uncompressed;

timeFormat       general;

timePrecision    7;

runTimeModifiable yes;

adjustTimeStep   yes;

maxCo            0.5;
maxDeltaT        1;
maxAlphaCo  0.5;

functions
{
forces
{
type forceCoeffs;
functionObjectLibs ( "libforces.so" );
outputControl timeStep;
outputInterval 1;
```

```
patches
(
foil
);
pName p;
UName U;
rhoName rhoInf;
log true;
rhoInf 1000;
liftDir (0 -0.034899496702501 0.999390827019096);
dragDir (0 0.999390827019096 0.034899496702501);
CofR (0 0 0);
pitchAxis (0 0 0);
magUInf 4.948046;
lRef 0.6063;
Aref 0.006063;
}
};


libs
(
"libmykk1Omega1.so"
    "libOpenFOAM.so"
    "libincompressibleTurbulenceModel.so"
    "libincompressibleRASModels.so"
    "libforces.so"
    "libfiniteVolume.so"
  );



// ******************************************************************* //
```

A.2.2 fvSchemes File

```
/*--------------------------------*- C++ -*----------------------------------*\
| =========                 |                                                  |
|                           |                                                  |
| \\      /  F ield         | OpenFOAM: The Open Source CFD Toolbox            |
|                           |                                                  |
| \\    /   O peration      | Version:  1.6                                    |
|                           |                                                  |
|  \\  /    A nd            | Web:      www.OpenFOAM.org                       |
|                           |                                                  |
|   \\/     M anipulation   |                                                  |
|                           |                                                  |
\*---------------------------------------------------------------------------*/
FoamFile
{
    version     2.0;
    format      binary;
    class       dictionary;
    location    "system";
    object      fvSchemes;
}
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //

ddtSchemes
{
    default         steadyState;
}

gradSchemes
{
    limitedGauss    cellLimited Gauss linear 1;
    default         Gauss linear;
    grad(U)         Gauss linear;
    grad(p)         Gauss linear;
```

104

```
    }

divSchemes
{
    default            none;
    div(phi,U)         bounded Gauss linearUpwindV limitedGauss;
    div((nuEff*dev(T(grad(U))))) Gauss linear;
    div(phi,kt)        bounded Gauss limitedLinear 1;
    div(phi,kl)        bounded Gauss limitedLinear 1;
    div(phi,omega)     bounded Gauss limitedLinear 1;
    div(U,p)           Gauss linear;
}


laplacianSchemes
{
    default Gauss linear corrected;
    laplacian(muEff,U) Gauss linear corrected;
    laplacian(alphaEff,h) Gauss linear corrected;
    laplacian((rho*rAU),p) Gauss linear corrected;
    laplacian(DepsilonEff,epsilon) Gauss linear corrected;
    laplacian(DkEff,k) Gauss linear corrected;
    laplacian(1,p)   Gauss linear corrected;
    laplacian((rho*(1|A(U))),p) Gauss linear corrected;
    laplacian(DomegaEff,omega)    Gauss linear corrected;
}


interpolationSchemes
{
    default            linear;
    div(U,p)           upwind phi;
}


snGradSchemes
{
    default            corrected;
}
```

```
fluxRequired
{
    default         no;
    p               ;
}



// ***************************************************************************** //
```

A.2.3   fvSolution File

```
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //


/*--------------------------------*- C++ -*----------------------------------*\
| =========                 |
|
| \\      /  F ield         | OpenFOAM: The Open Source CFD Toolbox
|
|  \\    /   O peration     | Version:  1.6
|
|   \\  /    A nd           | Web:      www.OpenFOAM.org
|
|    \\/     M anipulation  |
|
\*---------------------------------------------------------------------------*/
FoamFile
{
    version     2.0;
    format      ascii;
    class       dictionary;
    location    "system";
    object      fvSolution;
}
// * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * * //
```

```
solvers
{
    p
    {
        solver          GAMG;
        tolerance       1e-06;
        relTol          1.e-3;
        smoother        GaussSeidel;
        nPreSweeps      2;
        nPostSweeps     2;
        cacheAgglomeration true;
        nCellsInCoarsestLevel 10;
        agglomerator    faceAreaPair;
        mergeLevels     1;
    }
     pFinal
    {
        solver          GAMG;
        tolerance       1e-06;
        relTol          1.e-3;
        smoother        GaussSeidel;
        nPreSweeps      2;
        nPostSweeps     2;
        cacheAgglomeration true;
        nCellsInCoarsestLevel 10;
        agglomerator    faceAreaPair;
        mergeLevels     1;
    }

    "(U|kt|kl|omega)"
    {
        solver          smoothSolver;
        smoother        GaussSeidel;
        nSweeps         2;
        tolerance       1e-08;
```

```
        relTol          1.e-3;
    }
 "(U|kt|kl|omega)Final"
    {
        solver          smoothSolver;
        smoother        GaussSeidel;
        nSweeps         2;
        tolerance       1e-08;
        relTol          1.e-3;
    }
}


SIMPLE
{
    momentumPredictor   yes;
    correctPhi          no;
    nOuterCorrectors    1;
    nCorrectors         2;
    nNonOrthogonalCorrectors 1;
    MaxCo           1;
    pRefCell            0;
    pRefValue           0;
    residualControl
     {
        p               1e-6;
        U               1e-6;
        nut             1e-6;
        kl               1e-6;
        kt               1e-6;
        omega            1e-6;
    }

}
relaxationFactors
{
    p                   0.3;
```

```
    U               0.7;

    kl              0.8;

    kt              0.8;

    omega       0.8;

}
```

// ** //

Bibliography

[1] About gmsh, $http : //geuz.org/gmsh/$. Accessed: 2014-11-30.

[2] About openfoam, $http : //www.openfoam.org/$. Accessed: 2014-11-30.

[3] Ansys fluent user's guide. Accessed: 2014-12-02.

[4] yplus calculator, $http : //www.pointwise.com/yplus/$. Accessed: 2014-12-02.

[5] MIT 2.29 Fall 2011. Lecture 28. Technical report, MIT, 2011.

[6] MIT 2.20 Fall 2014. Hydrodynamics lecture notes. Technical report, MIT, 2014.

[7] Johnson F. T. Spalart P.R. Allmaras, S. R. Modifications and clarifications for the implementation of the spalart-allmaras turbulence model. Aug 2012.

[8] McGhee Robert J. Beasley, William D. Experimental and theoretical low speed aerodynamic characteristic of the naca 65,1-213, a=0.5 airfoil. Technical report, NASA.

[9] Ismail B. Celik. Introductory turbulence modeling. Technical report, West Virginia University, Mechanical Engineering Department, 1999.

[10] Anil W. Date. *Introduction to Computational Fluid Dynamics*. CAmbridge University Press, 2005.

[11] Gilles Eggenspieler. Modelling laminar-turbulent transition processes. Technical report, ANSYS.

[12] Peric M. Ferziger J.H. *Computational Methods for Fluid Dynamics 3rd Edition*. Springer, 2002.

[13] J. Furst. Numerical simulation of transitional flows with laminar kinetic energy. 2013.

[14] Nakayama P.I. Harlow, F. R. Transport of turbulence energy decay rate. 1968.

[15] Coupland J. Ercoftac special interest group on laminar to turbulent transition and retransition: T3a case. Technical report, 1990.

[16] H. Jasak. Numap-foam summer school turbulence modelling for cfd lecture. Technical report, University of Zagreb, Crotia, 15.

[17] Teymour Javaherchi. Review of spalart-allmaras turbulence model and its modifications. March 2010.

[18] Xiaofeng Liu. Modeling of earth surface dynamics and related problems using openfoam. Technical report, University of Texas San Antonio, Texas, 2013.

[19] Stanford University Applied Aeordynamics Course Material. The naca airfoil series. Technical report.

[20] F. R. Menter. Two-equation eddy-viscosity turbulence models for engineering applications. pages 330–331, 1994.

[21] J. N. Newman. *Marine Hydrodynamics*. The MIT Press, 1977.

[22] Colleen D. Scott-Pomerantz. The k-epsilon model in the theory of turbulence. Technical report, University of Pittsburgh, 2004.

[23] Introduction to CFD. Lecture 1. Technical report, Institute of Applied Mathematics, University of Dortmund.

[24] H K Versteeg and Malalasekera W. *An introduction to Computational Fluid Dynamics , The Finite Volume Method Second edition*. Pearson Education Limited 1995, 2007.

[25] Leylek J.H. Walters D.K. A three-equation eddy-viscosity model for reynolds-averaged navier-stokes simulations of transitional flow. December 2008.

[26] Leylek J.H. Walters D.K. Computational fluid dynamics study of wake-induced transition on a compressor-like flat plate. January 2005.