National Technical University of Athens

Department of Naval Architecture and Marine Engineering
Laboratory for Ship and Marine Hydrodynamics

Implementation of transition modelling on an Incompressible Solver

---

Diploma Thesis

Anastasios Magoulas

Thesis Supervisor: Prof. G.Papadakis
Committee Member: Prof. G.Triantafyllou
Committee Member: Prof. G.Grigoropoulos

Athens,October 2020

Acknowledgments

This thesis could not have been completed without the direct or indirect assistance of many people, so I would like to take this opportunity to express my gratitude.

First of all, I would like to thank Professor Georgios Papadakis for his targeted guidance and his willingness to share his knowledge during the time that I authored this thesis. Also I would like to thank him for piquing my interest in the study of Computational Fluid Dynamics.

Special thank to Mr. Kalantzis Konstantinos with whom I collaborated for a part of this work.

Setting coworkers aside, I would like to thank my family members, Nikos, Antonis, Ioanna and Sofia for their unlimited support during this time and especially my father Antonios who , by offering his academic experience, helped me reshape my way of thinking.

Lastly, I would like to express my sincere gratitude to Antonis, Dimitris, Mika, Vaggelis and Christina for offering their invaluable company during this time.

Abstract

The aim of the present work was to asses the performance of the modelling of transition from laminar to turbulent flow. For this purpose an incompressible solver was utilized, where the velocity and pressure fields were linked with the aid of the Artificial Compressibility Method. From all the transition models available in the literature, the Langtry and Menter $\gamma - \mathrm{Re}_\theta$ model was selected, which is a two-equation model and belongs to the family of Local Correlation Based Transition Models.

For validation purposes, simulations were performed on two dimensions for two airfoils typically used in the wind power generation industry, as well as a three-dimensional propeller. The results were then compared to experiments available in the literature in order to assess how well the solver captures the complicated phenomenon of transition.

The simulation for the two-dimensional airfoils indicated that the results are in good agreement with the experimental ones only for small angles of attack, while in higher angles of attack, the effect of strong three dimensional phenomena lead to discrepancies between the results of the solver and the experimental ones.

For the case of the three dimensional propeller, the solver managed to capture the correct slope for the curves on the Open Water Diagram, while a corresponding Fully Turbulent Solver failed to capture them.

Table of Contents

Lists of Figures and Tables

Tables **Page**

1. Introduction

Since the 17th century science and engineering have developed on two parallel paths, the first being pure experiment and the second being pure theory. In the recent years, with the increase of available computer power a third alternative has emerged as a compromise between the two methods discussed above, Computational Dynamics. The same applies for the sector of Fluid Dynamics.

Computational Fluid Dynamics or CFD is the analysis of systems involving fluid flow, heat transfer and associated phenomena, as chemical reactions by means of a computer-based simulation [1]. CFD combines disciplines from Fluid Dynamics, Mathematics and Computer Science in order to simulate fluid flows. The emergence of CFD helped reduce the cost of designing aerodynamic bodies, since for some cases the cost of an experimental test is prohibitive. Moreover, not all flow regimes can be simulated on an experimental facility. However, it should be noted that CFD should not be considered as a replacement of theory or experimental testing, it is just an additional, supplementary tool for verification and validation.

In historical terms, at the first stages of its application, CFD could only be used for simple two-dimensional inviscid flows, however at present time, with the vast computer power available, complex three-dimensional flows can be simulated. The need for CFD emerged in the 60s and 70s in the aerodynamics industry and was used only in high technology engineering areas, but since then it has been used as an educational, design and research tool in the following sectors:

- Automotive Industry and Internal Combustion Engines
- Turbomachinery
- Civil Engineering
- Naval Architecture
- Chemical Engineering
- Meteorology and Oceanography
- Biomedical Engineering
- Sports Engineering

In general, a Navier Stokes CFD Solver, like the one used in the context of this work, divides the computational domain into a set of smaller control volumes and solves the Navier Stokes equations at each control volume using some values from the neighbouring control volumes. This approach, which is called Direct Numeric Simulation is, even with today's computers, possible only for Laminar flows, where the Reynolds Numbers are low. The need to simulate also turbulent flows, some of which are of great engineering significance, led to the appearance of turbulence models, which tried to capture the effects of turbulence without raising the computational cost to preventive levels. However, it is worth mentioning that different turbulence models can give different solution accuracy for the same problem and also different models are better for different flow regimes. Consequently, it is important that the CFD results are interpreted, bearing in mind the drawbacks and the strengths of the turbulence model used.

Another limitation of the above-mentioned turbulence models is that they treat the flow as either fully laminar or fully turbulent. In the recent years a substantial amount of scientific effort has been put into firstly understanding the complex phenomena of transition from laminar to turbulent flow and secondly into finding suitable models for implementing transition in the CFD Simulations.

Transition occurs through different mechanisms for different applications [2]. In aerodynamic flows transition is caused typically by flow instabilities (Tollmein-Schlicthing waves), whose amplitude grows exponentially resulting in an non-linear breakdown to turbulence. This mechanism is termed Natural Transition and is illustrated in Figure 1. On turbomachinery the main transition mechanism is by-pass transition [3],[4], where the high levels of freestream turbulence, caused by the upstream blade rows, force the laminar boundary layer into transition far upstream of the natural transition

location. Lastly, a third type of transition is Separation Induced Transition, where a laminar boundary layer separates and then gets re-attached as turbulent [5].

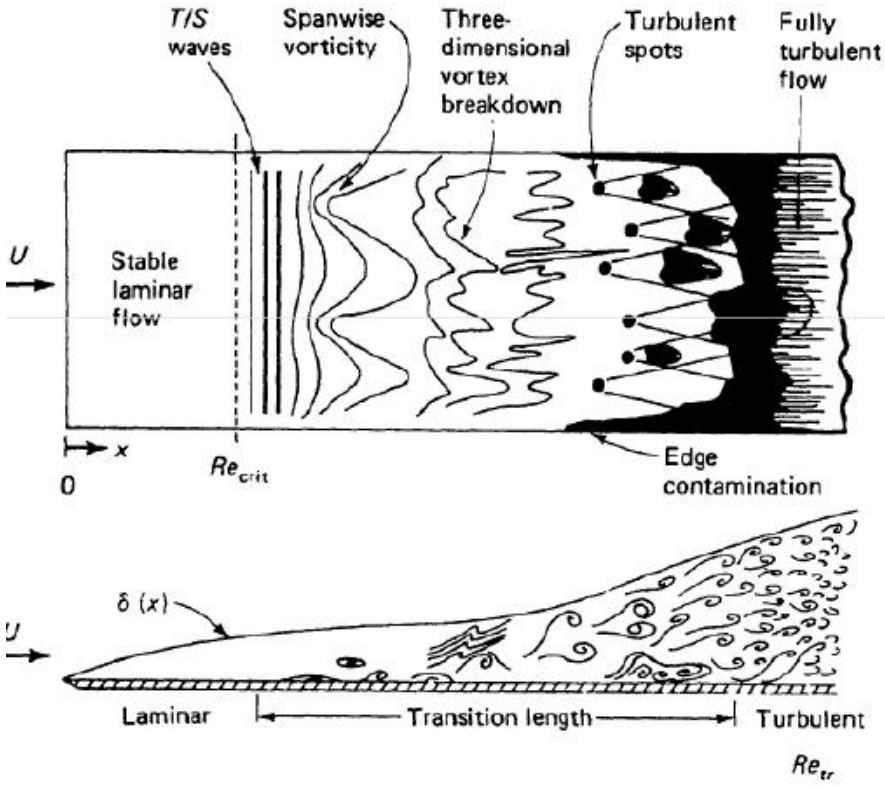

Figure 1. Natural Transition from laminar to turbulent flaw. Source: [6]

Except for the decision about which turbulence model must be used for a certain application, a CFD user must also decide whether the utilized code will account for fluid compressibility. If the flowing medium is air, at low speeds (speed much lower than the speed of sound) the effects of compressibility are negligible. On the other hand, at speeds close to the speed of sound or higher the effects of compressibility become very important. From a mathematical point of view, the compressible Navier-Stokes Equations can be used with no problem as far as coupling of velocity and the pressure, whereas in the incompressible equations the density is constant and by definition the velocity is not linked to the pressure. The above-mentioned problem is solved by the method of Artificial Compressibility by introducing a pressure derivative in the continuity equation and marching it through pseudo time until a steady state is reached. In this case, the original incompressible equations are recovered. The Artificial Compressibility Method was first introduced by Chorin in 1967 [7] and has been since improved by many researchers.

Thesis Scope

The scope of this work is to examine the implementation of transition modelling on an incompressible solver. The results of the simulations were compared to experimental ones available in the literature. The Langtry and Menter $\gamma - \mathrm{Re}_\theta$ transition model was selected, which belongs to the family of Local Correlation Based Transition Models. The work of K. Diakakis [8] who implemented transition

models on a compressible solver, served as a basis for the present thesis. The utilized CFD solver which was the so called MaPFLow, which was developed by G. Papadakis in his PhD thesis in the National Technical University of Athens [9].

2. CFD Solver MaPFlow

Introduction

The purpose of this chapter is to present, in some detail, the MaPFlow solver, which was used for the CFD simulations . On section 2.1 the basic governing equations of CFD are presented. Section 2.2 gives the principles for solving these equations, including the various spatial discretization schemes on sub-sections 2.2.3 and 2.2.4, as well as the temporal discretization on sub-section 2.2.7. The various boundary conditions utilized are included in Section 2.3. Section 2.4 discusses the physics of turbulence and the Turbulence Model used in the context of this work. Transition modelling is presented in Section 2.5. Lastly, the solution of the system of equations is included in Section 2.6.

2.1 Governing Equations

The governing equations of fluid dynamics are derived from the following conservation laws of physics:

- Conservation of mass
- Newton's second law, which states that the rate of change of momentum equals the sum of forces on a fluid particle
- The first law of thermodynamics, which states that the rate of change of energy is equal to the sum of the rate of heat addition to and the rate of work done on a fluid particle.

In a compressible solver, the continuity equation is in essence a transport equation for the density, while the energy equation serves as a transport equation for the temperature; then by utilizing the state equation, the pressure can be computed. On the other hand, in an incompressible solver, the density is constant and therefore not linked to the pressure, consequently coupling the velocity and the pressure is a problem for the solution of the flow field. A remedy for the above mentioned issue is offered by the Artificial Compressibility Method. In the context of this method, a fictitious time derivative of pressure is added in the continuity equation and the solution is marched in pseudotime 10]. When a steady state solution is achieved, this time derivative vanishes and the original incompressible equations are recovered. It is clear from the above that in the case of an incompressible solver, no energy equation is solved.

Let Ω denote an arbitrary control volume and $\partial\Omega$ its boundary. The governing equation for an incompressible flow for cell i is given by:

$$\frac{\partial(W_i\Omega_t)}{\partial\tau}+\mathrm{E}\frac{\partial(W_i\Omega_t)}{\partial t}+\oint_{\partial\Omega_t}\left(\vec{F}_{INV}-\vec{F}_{VIS}\right)\cdot\hat{n}dS=0 \qquad (2.1)$$

Where $\bar{W}$ is the vector of the conservative flow variables:

$$\vec{W} = \begin{pmatrix} p \\ u \\ v \\ w \end{pmatrix} \tag{2.2}$$

Where p denotes the pressure, (u, v, w) are the components of the velocity vector and τ is the pseudo-time.

E is a matrix given by:

$$E = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \tag{2.3}$$

$\vec{F}_{INV}$ and $\vec{F}_{VIS}$ are the vectors of the Inviscid and Viscous Fluxes respectively:

$$\vec{F}_{INV} \cdot \hat{n} = \begin{pmatrix} \beta\left(u \cdot \hat{n}_x + v \cdot \hat{n}_y + w \cdot \hat{n}_z\right) \\ \left(u^2 + p\right) \cdot \hat{n}_x + u \cdot v \cdot \hat{n}_y + u \cdot w \cdot \hat{n}_z \\ v \cdot u \cdot \hat{n}_x + \left(v^2 + p\right) \cdot \hat{n}_y + v \cdot w \cdot \hat{n}_z \\ w \cdot u \cdot \hat{n}_x + w \cdot v \cdot \hat{n}_y + \left(w^2 + p\right) \cdot \hat{n}_z \end{pmatrix} \tag{2.4}$$

$$\vec{F}_V = \begin{pmatrix} 0 \\ \tau_{xx} \cdot \hat{n}_x + \tau_{xy} \cdot \hat{n}_y + \tau_{xz} \cdot \hat{n}_z \\ \tau_{yx} \cdot \hat{n}_x + \tau_{yy} \cdot \hat{n}_y + \tau_{yz} \cdot \hat{n}_z \\ \tau_{zx} \cdot \hat{n}_x + \tau_{zy} \cdot \hat{n}_y + \tau_{zz} \cdot \hat{n}_z \end{pmatrix} \tag{2.5}$$

Where $\hat{n} = \left(\hat{n}_x, \hat{n}_y, \hat{n}_z\right)$ is the unit normal vector and β is the artificial compressibility or pseudo compressibility parameter. The value of this parameter, controls the speed at which the pseudo waves, introduced by this formulation, travel.

2.2 Principles of solution of the governing equations

2.2.1 The Finite Volume Method

In the Finite Volume Method the computational domain is divided into a finite number of control volumes. The variables are calculated in the centre of each control volume and the equations solved represent the conservation of relevant properties in each control volume. In this process, the values of the variables at the surfaces of the control volume are needed; these values are computed by interpolation from the values at the control volume centers.This division of the computational domain is achieved by creating a mesh or grid. Each cell of this grid constitutes a control volume. Hereby, the terms cell and control volume will be used interchangeably.

Discretization is the process of transforming the governing Partial Differential Equations and initial and boundary conditions into a system of discrete algebraic equations. In this way, instead of obtaining an analytical solution in every point of the computational domain, the solution is computed at discrete points in the geometrical domain. The Finite Volume Method Method results in obtaining an algebraic equation for each of the control volumes, in which a number of the neighbouring nodal values appear. MapFlow utilizes the so called-method of lines [11], where the governing equations are first discretized in space and then advanced in time starting from a known initial solution.

2.2.2 Spatial Discretization

Assuming that the control volume does not change in time, the time derivative of the conservative variable vector can be cast in the form:

$$\frac{\partial}{\partial t}\int_{\Omega}\vec{W}d\Omega = \Omega\frac{\partial\vec{W}}{\partial t} \tag{2.6}$$

Consequently, equation (2.1) becomes:

$$\frac{\partial\vec{W}}{\partial\tau} + E\cdot\frac{\partial\vec{W}}{\partial t} = -\frac{1}{\Omega}\left[\oint_{\partial\Omega}\left(\vec{F}_{INV} - \vec{F}_{VIS}\right)dS\right] \tag{2.7}$$

The surface integral on the right-hand side of the above equation is replaced with the sum of fluxes crossing the faces of the control volume. This approximation is called spatial discretization. An assumption is made that the flux is constant along each individual face and it is computed at the center of the face.

Let $\Omega_{I,J,K}$ denote a particular control volume, then the equation (2.7) becomes:

$$\frac{dW_{I,J,K}}{d\tau} + \mathrm{E}\cdot\frac{dW_{I,J,K}}{dt} = -\frac{1}{\Omega_{I,J,K}}\left[\sum_{m=1}^{N_F}\left(\vec{F}_{INV} - \vec{F}_{VIS}\right)_m\Delta S_m\right] \tag{2.8}$$

Where N_F is the number of faces of the control volume and ΔS_m is the area of face m.

As a result of the spatial discretization, we obtain a system of partial differential equations that are hyperbolic in time, which means we have to start from an initial solution and advance the equations in time.

2.2.3 Variable reconstruction

From the above it it clear that the values of the conservative variables must be known at all cell faces in order to solve the equations, but the variables stored refer to the center of each control volume. This means that the values on the cell faces have to be suitably interpolated from the values on the cell centers. This process is termed Variable Reconstruction.

Before presenting the various discretization schemes it is appropriate to define the Stencil or Computational Molecule. The Stencil consists of all the cell centers that a particular scheme uses to interpolate the fluxes on the cell faces.

In this work, three of the above mentioned discretization schemes where utilized with different orders of accuracy. The MUSCL scheme the PLR scheme and the QUICK scheme.

2.2.4.1 Van Leer's MUSCL scheme.

The Monotone Upstream-Centered Schemes for Conservation Laws, was first proposed by Van Leer [12] and utilizes the following stencil.

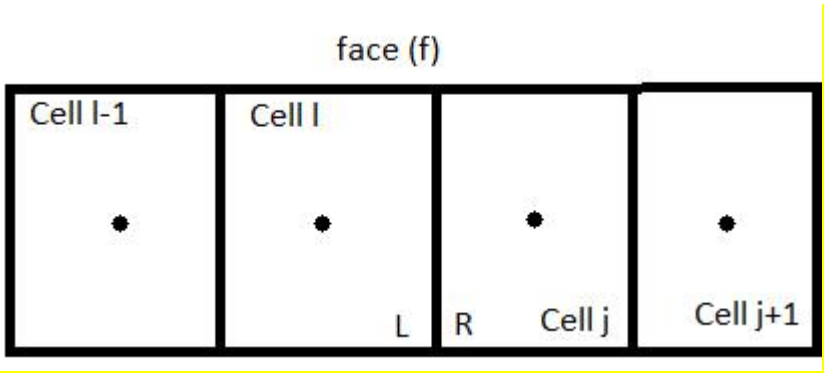

Figure 2. Four-cell stencil for the MUSCL and QUICK schemes

According to the MUSCL scheme the left and right states for the cell face (f) are computed as follows:

$$\vec{W}_L = \vec{W}_I + \frac{1}{4}\left((1-\kappa)(\vec{W}_I - \vec{W}_{I-1})\phi(r_{I-1}) + (1+\kappa)(\vec{W}_j - \vec{W}_I)\phi(r_I)\right) \quad (2.9)$$

$$\vec{W}_R = \vec{W}_J + \frac{1}{4}\left((1-\kappa)(\vec{W}_{J+1} - \vec{W}_J)\phi(r_J) + (1+\kappa)(\vec{W}_j - \vec{W}_I)\phi(r_{J+1})\right) \quad (2.10)$$

Where $k = \frac{1}{3}$ and r denotes the ratio of the backwards difference operator to the forward difference operator:

$$r_I = \frac{\bar{W}_I - \bar{W}_{I-1}}{\bar{W}_J - \bar{W}_I} \tag{2.11}$$

If the denominator is lower than 0, then $r_I=1$. Finally $\varphi(r)$ is called the Limiter Function. Limiters are further discussed in the following paragraph.

Second and higher order accurate schemes can present non-realistic wiggles (oscillations) in the solution, under certain flow conditions like, for example, in regions of high gradients. It is commonly accepted that in order for a scheme to not present these unwanted oscillations it must be monotonicity preserving. This means that it must not create new extrema during the time evolution and also the values of an already existing local minimum must be non-decreasing and for a local maximum they must be non-increasing. On upwind-based schemes, the limiter must become zero in regions of strong discontinuities, in order for the scheme to switch to pure upwind discretization, which is guaranteed to be monotonicity preserving. On the other hand, on smooth flow regions the limiter should turn to 1 and give the original unlimited value. Two limiters can be used in the MaPFlow solver for the MUSCL scheme, the limiter of van Albada [13] or the Superbee limiter [14].

The van Albada limiter reads:

$$\phi(r_I) = \frac{r_I^{\,2} + r_I}{r_I^{\,2} + 1} \tag{2.12}$$

While the Superbee limiter reads:

$$\phi(r_I) = \max\left(0, \min(2r_I, 1), \min(r_I, 2)\right) \tag{2.13}$$

2.2.4.2 PLR

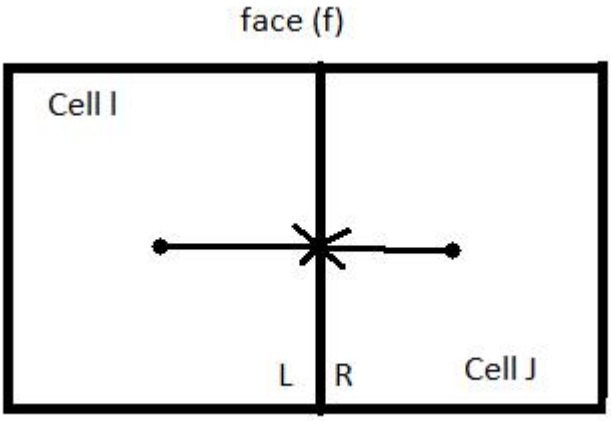

Figure 3. Two- cell stencil used in the PLR scheme

The Piecewise Linear Reconstruction scheme, which was first introduced by Barth and Jespersen [15], utilizes the stencil in Figure 2. The solution is considered to be linearly distributed on each cell and the left and right states are obtained by:

$$\begin{aligned} \vec{W}_L &= \vec{W}_I + \psi_I \left(\nabla \vec{W}_I * \vec{r}_L \right) \\ \vec{W}_R &= \vec{W}_J - \psi_J \left(\nabla \vec{W}_J * \vec{r}_R \right) \end{aligned} \tag{2.14}$$

Where $\vec{W}$ is the primitive variable vector:

$$\vec{W} = \begin{pmatrix} p \\ u \\ v \\ w \end{pmatrix} \tag{2.15}$$

$\vec{r}_L$ and $\vec{r}_R$ are vectors from the cell centers of cells I, J respectively to the cell face.

$\nabla \vec{W}$ is the gradient of W:

$$\nabla \vec{W} = \left[\frac{\partial \vec{W}}{\partial x} \quad \frac{\partial \vec{W}}{\partial y} \quad \frac{\partial \vec{W}}{\partial z} \right]^T \tag{2.16}$$

In the context of the MaPFlow solver the above-mentioned gradients can be computed using either the Green-Gauss Approach or the Least Squares Approach.

Finally, ψ denotes the Venkatakrishnan's limiter [16], [17] given by:

$$\psi_i = \min_j \left\{ \begin{array}{lll} \dfrac{1}{\Delta_2}\left[\dfrac{\left(\Delta^2_{I,\max}+\varepsilon^2\right)\Delta_2 + 2\Delta_2^2\Delta_{1,\max}}{\Delta^2_{1,\max}+2\Delta_2^2+\Delta_{1,\max}\Delta_2+\varepsilon^2}\right] & if & \Delta_2 > 0 \\ \dfrac{1}{\Delta_2}\left[\dfrac{\left(\Delta^2_{I,\min}+\varepsilon^2\right)\Delta_2 + 2\Delta_2^2\Delta_{1,\min}}{\Delta^2_{1,\min}+2\Delta_2^2+\Delta_{1,\min}\Delta_2+\varepsilon^2}\right] & if & \Delta_2 < 0 \\ 1 & if & \Delta_2 = 0 \end{array} \right\} \qquad (2.17)$$

Where:

$$\begin{aligned} \Delta_{1,\max} &= U_{\max} - U_i \\ \Delta_{1,\min} &= U_{\min} - U_i \end{aligned} \qquad (2.18)$$

In the above equations $U_{\max}$ and $U_{\min}$ are the maximum and minimum values, respectively, of all surrounding nodes J including the node I itself.

The purpose of ε^2 in the above equations is to control limiter strictness. For large values of ε^2 the limiter function will take values around unity and no limiter would be imposed on the gradient. This may lead to unrealistic oscillations in the solution. On the other hand, if ε^2 is close to zero it results to full limiting and this may stall the convergence. It is now common practice for ε^2 to take values proportional to the mesh length scale.

$$\varepsilon^2 = \left(K\Delta h\right)^3 \qquad (2.19)$$

Where Δh is the cube root of the volume (in 3D) or the square root of the surface (in 2D) of the control volume.

The proportionality factor K is a free parameter; for small values of K the limiter is strict and PLR becomes first order accurate, while for large values of K the scheme is unlimited. Usually K=5 is chosen.

2.2.4.3 Leonard's QUICK scheme

The Quadratic Upstream Interpolation for Convective Kinetics (QUICK) of Leonard [18] uses a three-point upstream weighted quadratic function to calculate the values on the cell faces. This means that the computational grid consists of two bracketing nodes (one at each side of the cell) and an extra node in the upstream direction. It is third order accurate based on the Taylor series truncation error on a uniform mesh (see Appendix A).

The QUICK scheme has possibly problems in the implementation of boundary conditions, and can be proven to be unsteady for computing turbulent quantities.

The left and right states are computed as follows:

Left state

If $\left|\vec{W}_J - \vec{W}_{I-1}\right| \le 10^{-8}$, then:

$$\vec{W}_L = \frac{1}{2}\left(\vec{W}_I + \vec{W}_J\right) - \frac{1}{8}\left(\vec{W}_J - 2\vec{W}_I + \vec{W}_{I-1}\right) \tag{2.20}$$

Else if $\left|\vec{W}_j - 2\vec{W}_I + \vec{W}_{I-1}\right| \le 0.3\left|\vec{W}_j - \vec{W}_{I-1}\right|$, then:

$$\vec{W}_L = \frac{1}{2}\left(\vec{W}_I + \vec{W}_J\right) - \frac{1}{8}\left(\vec{W}_J - 2\vec{W}_I + \vec{W}_{I-1}\right) \tag{2.21}$$

Else:

$$\vec{W}_{I,nud} = \frac{\vec{W}_I - \vec{W}_{I-1}}{\vec{W}_J - \vec{W}_{I-1}} \tag{2.22}$$

$$\text{If } \vec{W}_{I,nvd} \ge 1.5 \quad or \quad \vec{W}_{I,nvd} \le -1 \quad or \quad 0.35 \le \vec{W}_{I,nvd} \le 0.65$$
$$\vec{W}_{face} = 0.75 + 0.75\left(\vec{W}_{I,nvd} - 0.5\right) \tag{2.23}$$

$$\text{Else if } -1 < \vec{W}_{I,nvd} \le 0$$
$$\vec{W}_{face} = 0.375\vec{W}_{I,nvd} \tag{2.24}$$

$$\text{Else if } 0 < W_{I,nvd} \le 0.35 \quad or \quad 0.65 < W_{I,nvd} \le 1$$

$$\vec{W}_{face} = \frac{\sqrt{\vec{W}_{I,nvd}\left(1 - \vec{W}_{I,nvd}\right)^3} - \vec{W}^2_{I,nvd}}{1 - 2\vec{W}_{I,nvd}} \tag{2.25}$$

Else if $1 < \vec{W}_{I,nvd} \le 1.5$

$$\vec{W}_{face} = \vec{W}_{I,nvd}$$
$$\vec{W}_L = \vec{W}_{I-1} + \left(\vec{W}_J - \vec{W}_{I-1}\right)\vec{W}_{face} \tag{2.26}$$

Right state

If $\left|\vec{W}_J - \vec{W}_{J+1}\right| \le 10^{-8}$, then:

$$\vec{W}_R = \frac{1}{2}\left(\vec{W}_J + \vec{W}_I\right) - \frac{1}{8}\left(\vec{W}_I - 2\vec{W}_J + \vec{W}_{J+1}\right) \tag{2.27}$$

Else if $\left|\vec{W}_I - 2\vec{W}_J + \vec{W}_{J+1}\right| \le 0.3\left|\vec{W}_I - \vec{W}_{J+1}\right|$, then:

$$\vec{W}_R = \frac{1}{2}\left(\vec{W}_J + \vec{W}_I\right) - \frac{1}{8}\left(\vec{W}_I - 2\vec{W}_J + \vec{W}_{J+1}\right) \tag{2.28}$$

Else:

$$\vec{W}_{J,nud} = \frac{\vec{W}_J - \vec{W}_{J+1}}{\vec{W}_I - \vec{W}_{J+1}} \tag{2.29}$$

If $\vec{W}_{J,nvd} \ge 1.5 \quad or \quad \vec{W}_{J,nvd} \le -1 \quad or \quad 0.35 \le \vec{W}_{J,nvd} \le 0.65$

$$\vec{W}_{face} = 0.75 + 0.75\left(\vec{W}_{J,nvd} - 0.5\right) \tag{2.30}$$

Else if $-1 < \vec{W}_{J,nvd} \le 0$

$$\vec{W}_{face} = 0.375\vec{W}_{J,nvd} \tag{2.31}$$

Else if $0 < W_{J,nvd} \le 0.35 \quad or \quad 0.65 < W_{J,nvd} \le 1$

$$\vec{W}_{face} = \frac{\sqrt{\vec{W}_{J,nvd}\left(1-\vec{W}_{J,nvd}\right)^3} - \vec{W}^2_{J,nvd}}{1-2\vec{W}_{J,nvd}} \tag{2.32}$$

Else if $1 < \vec{W}_{J,nvd} \le 1.5$

$$\vec{W}_{face} = \vec{W}_{J,nvd}$$
$$\vec{W}_R = \vec{W}_{J+1} + \left(\vec{W}_I - \vec{W}_{J+1}\right)\vec{W}_{face} \tag{2.33}$$

2.2.5 Discretization of the Convective Fluxes

As a result of the variable reconstruction process described above, the generally discontinuous Left and Right states are computed. Consequently, a method to combine these two values and evaluate the convective fluxes is required. In the context of this work, this was made possible with the aid of a Flux-Difference Splitting Scheme.

Flux Difference Splitting Schemes, which were first introduced by Godunov [19], evaluate the fluxes by solving the Riemann (shock tube) problem [20]. This family of schemes accounts for the physically correct transfer of information throughout the flow. In particular, MaPFlow utilizes Roe's approximate Riemann Solver [21] which offers high accuracy for boundary layers and good treatment of shocks.

Roe's solver approximates the convective fluxes as a sum of wave contributions:

$$\left(\vec{F}_C\right)_{I+\frac{1}{2}} = \frac{1}{2}\left[\vec{F}_C\left(\vec{W}_R\right) + \vec{F}_C\left(\vec{W}_L\right) - \left|A_{ROE}\right|_{I+\frac{1}{2}}\left(\vec{W}_R - \vec{W}_L\right)\right] \tag{2.34}$$

In the above equations L and R denote the Left and Right states, respectively, and A_{ROE} is the Roe Matrix.

The Roe matrix is identical to the Convective Flux Jacobian with the difference that all flow variables are replaced with their Roe-averaged counterparts:

$$
\begin{aligned}
\widetilde{\rho} &= \sqrt{\rho_L \rho_R} \\
\widetilde{u} &= \frac{u_L\sqrt{\rho_L} + u_R\sqrt{\rho_R}}{\sqrt{\rho_L} + \sqrt{\rho_R}} \\
\widetilde{v} &= \frac{v_L\sqrt{\rho_L} + v_R\sqrt{\rho_R}}{\sqrt{\rho_L} + \sqrt{\rho_R}} \\
\widetilde{w} &= \frac{w_L\sqrt{\rho_L} + w_R\sqrt{\rho_R}}{\sqrt{\rho_L} + \sqrt{\rho_R}} \\
\widetilde{H} &= \frac{H_L\sqrt{\rho_L} + H_R\sqrt{\rho_R}}{\sqrt{\rho_L} + \sqrt{\rho_R}} \\
\widetilde{c} &= \sqrt{(\gamma-1)\left(\widetilde{\mathrm{H}} - \frac{\widetilde{q}^2}{2}\right)} \\
\widetilde{V} &= \widetilde{u}n_x + \widetilde{v}n_y + \widetilde{w}n_z \\
\widetilde{q} &= \widetilde{u}^2 + \widetilde{v}^2 + \widetilde{w}^2
\end{aligned}
\tag{2.35}
$$

2.2.6 Discretization of the Viscous Fluxes

The control volume for the viscous fluxes is chosen to be the same as for the convective fluxes. For the reconstruction of the variables, the values on the cell face are obtained by simple central averaging. In particular, for the face between cells *I* and *J*:

$$
\vec{V}_{I,J} = \frac{1}{2}\left(\vec{V}_I + \vec{V}_j\right) \tag{2.36}
$$

In order to calculate the viscous fluxes, apart from computing the values at the cell face we also have to compute the gradients of certain quantities. Assuming that the gradient on the centers of the cells, *I* and *J,* are already approximated, either by the Green- Gauss method or by the least squares method, the gradient at the face between cells *I* and *J* is given by:

$$
\nabla\vec{V}_{I,J} = \overline{\nabla\vec{V}_{I,J}} - \left[\overline{\nabla\vec{V}_{I,J}}\vec{t}_{I,J} - \left(\frac{\partial\vec{V}}{\partial l}\right)_{I,J}\right]\vec{t}_{I,J} \tag{2.37}
$$

Where $\overline{\nabla\vec{V}_{I,J}}$ is the mean gradient:

$$
\overline{\nabla\vec{V}_{I,J}} = \frac{1}{2}\left(\nabla\vec{V}_I + \nabla\vec{V}_J\right) \tag{2.38}
$$

$\left(\frac{\partial \vec{V}}{\partial l}\right)_{I,J}$ is a first order approximation of the directional derivative:

$$\left(\frac{\partial \vec{V}}{\partial l}\right)_{I,J} = \frac{\vec{V}_J - \vec{V}_I}{l_{I,J}} \tag{2.39}$$

Finally, $l_{i,j}$ is the distance between the cell centers and $\vec{t}_{i,j}$ is the unit normal vector pointing from cell center *I* to cell center *J*.

2.2.7 Temporal Discretization

Spatial discretization results, for each control volume, in the following equation:

$$\frac{d\left(\Omega_I \vec{W}_I\right)}{dt} = -\vec{R}_I \tag{2.40}$$

The temporal discretization schemes are divided to implicit and explicit schemes. Implicit schemes rely on already known variables, namely the vector $\vec{W}^n$ and the corresponding residual $\vec{R}^n$ to find the solution on the next time level. As the name suggests, the equations can be solved explicitly from already known values. As a result, for a given time step explicit schemes are easier to implement than implicit schemes and require less computational effort. On the other hand, there is a maximum time step that can be employed, which is dictated by stability criteria.

Implicit solutions use $\vec{R}\left(\vec{W}^{n+1}\right) = \vec{R}^{n+1}$ to obtain the solution of the new time level. These schemes involve solving a large system of equations simultaneously and have, for a given time step, larger computational cost than implicit ones. However, larger time steps can be used without stability problems. Since $\vec{R}^{n+1}$ is unknown, the following linear approximation is used:

$$\vec{R}^{n+1} \approx \vec{R}^n + \left(\frac{\partial \vec{R}}{\partial \vec{W}}\right)_I \Delta \vec{W}^n \tag{2.41}$$

Where:

$$\Delta \vec{W}^n = \vec{W}^{n+1} - \vec{W}^n \tag{2.42}$$

And $\left(\partial \vec{R} / \partial \vec{W}\right)$ is the flux Jacobian.

MaPFlow utilizes a finite difference scheme for the time derivative [22]:

$$\frac{1}{\Delta t}\left[\phi_{n+1}\left(\Omega\vec{W}\right)^{n+1}+\phi_{n}\left(\Omega\vec{W}\right)^{n}+\phi_{n-1}\left(\Omega\vec{W}\right)^{n-1}+\phi_{n-2}\left(\Omega\vec{W}\right)^{n-2}+.....\right]=-R^{n+1} \quad (2.43)$$

The accuracy of the above Backwards Differencing Formula (BDF) depends on the choice of ϕ_n.
BDF2OPT is a class of second order accurate backwards difference schemes, with half the error of a conventional 2nd order scheme. [23]

In Table 1, below, the values of ϕ_n are given , which guarantee a certain order of accuracy.

Order of Accuracy	ϕ_{n+1}	ϕ_n	ϕ_{n-1}	ϕ_{n-2}
1st	1	-1	0	0
2nd	3/2	-2	1/2	0
3rd	11/6	-3	3/2	-1/3
BDF2OPT	$3/2-\phi_{n-2}$	$-2+3\phi_{n-2}$	$1/2-3\phi_{n-2}$	-0.58/3

Table 1. Values of ϕ for a certain order of accuracy

2.2.7.1 Steady State Problems

Even in the case of steady state problems, the form of the equations to be solved requires marching of the solution in pseudo-time until an acceptable level of convergence is reached. For this purpose, MaPFlow utilizes a first order accurate scheme by linearizing R^{n+1} on equation (2.43) around the current time level:

$$\frac{\left(\Omega_I\Delta\vec{W}_I^{\,n}\right)}{\Delta t_I}=\vec{R}_I^{\,n}+\left(\frac{\partial\vec{R}}{\partial\vec{W}}\right)_I\Delta\vec{W}_I^{\,n} \quad (2.44)$$

Rearranging the terms on the above equation gives:

$$\left[\frac{\Omega_I}{\Delta t_I}+\left(\frac{\partial\vec{R}}{\partial\vec{W}}\right)_I\right]\Delta\vec{W}_I^{\,n}=-\vec{R}_I^{\,n} \quad (2.45)$$

The term inside the square brackets is denoted the **Implicit Operator**.

2.2.7.2 Local time stepping

Local Time Stepping [24] is a method where, in each cell, the maximum allowable timestep is used in order to accelerate convergence. The maximum allowable timestep is imposed by the Courant-Friedrichs-Lewy (CFL) condition [25]. It states that the Domain of Dependence of the numerical method has to include the Domain of Dependence of the partial differential equation. This

practically means that the time step must be equal or smaller than the time required to transport information across the stencil of the spatial discretization scheme. In this work the timestep was computed by:

$$\Delta t_I = CFL \frac{\Omega_{\mathrm{I}}}{\left(\widehat{\Lambda}_c + C\widehat{\Lambda}_v\right)_I} \tag{2.46}$$

Where $\widehat{\Lambda}_c$ and $\widehat{\Lambda}_v$ are a representation of the sum of convective and viscous spectral radii (Eigenvalues) respectively over all faces of the control volume, given by:

$$\begin{aligned}\left(\widehat{\Lambda}_c\right)_I &= \sum_{J=1}^{N_F}\left(\left|\vec{u}_{IJ}\vec{n}_{IJ}\right| + c_{IJ}\right)\Delta S_{IJ} \\ \left(\widehat{\Lambda}_v\right)_I &= \frac{1}{\Omega_{\mathrm{I}}}\sum_{J=1}^{N_F}\left[\max\left(\frac{3}{3\rho_{I,J}}, \frac{\gamma_{I,J}}{\rho_{I,J}}\right)\left(\frac{\mu_L}{\mathrm{Pr}_L} + \frac{\mu_T}{\mathrm{Pr}_T}\right)_{IJ}\left(\Delta S_{IJ}\right)^2\right]\end{aligned} \tag{2.47}$$

2.2.7.3 Time True Problems, Dual Time Stepping

Regarding unsteady problems, the Dual Step approach is used [26]. In the context of this method, on each physical time step the equations are marched on a different "pseudo-time level" and the corresponding time variable is denoted τ. The variables on the pseudo-time level are denoted W^* and they do not satisfy the original unsteady problem unless convergence is reached. It should be noted that the vector W^*, does not contain the pressure term.Essentially, for every physical time level, a steady problem is solved in the pseudo-time:

$$\frac{\partial\left(\Omega^{n+1}\vec{W}^*\right)}{\partial\tau} = -\left(\vec{R}^*\right)^{l+1} \tag{2.48}$$

Where $(l+1)$ is the new pseudo-time level. By setting:

$$\vec{R}^* = \frac{\partial\left(\Omega\vec{W}^*\right)}{\partial t}R\left(\vec{W}^*\right) \tag{2.49}$$

equation (2.48) becomes:

$$\frac{\partial\left(\Omega^{n+1}\vec{W}^*\right)}{\partial\tau} + \frac{\partial\left(\Omega\vec{W}^*\right)}{\partial t} = -R\left(\vec{W}^*\right) \tag{2.50}$$

When the process converges, $R^* = 0, W^* = W$ and the unsteady problem is satisfied.

Discretizing the above equation gives:

$$\frac{\Omega^{n+1}\Delta W^{*l}}{\Delta\tau}+\frac{1}{\Delta t}\left[\phi_{n+1}\left(\Omega\vec{W}^{*}\right)^{n+1}+\phi_{n}\left(\Omega\vec{W}\right)^{n}+\phi_{n-1}\left(\Omega\vec{W}\right)^{n-1}\right]=-R^{l+1}\Rightarrow$$
$$\frac{\Omega^{n+1}\Delta W^{*l}}{\Delta\tau}=-R^{*l+1} \tag{2.51}$$

With:

$$\Delta W^{*l}=W^{*l+1}-W^{*l},$$
$$R^{*l+1}=R^{l+1}-\frac{1}{\Delta t}\left[\phi_{n+1}\left(\Omega\vec{W}^{*}\right)^{n+1}+\phi_{n}\left(\Omega\vec{W}\ \right)^{n}+\phi_{n-1}\left(\Omega\vec{W}\right)^{n-1}\right] \tag{2.52}$$

The implementation of an implicit scheme in the Dual Time-Step approach involves linearizing the unsteady residual R^{*l+1}:

$$R^{*l+1}\approx R^{*l}+\left(\frac{\partial\vec{R}}{\partial\vec{W}^{*}}\right)_{l}\Delta\vec{W}^{l}\Rightarrow \tag{2.53}$$
$$R^{*l+1}=R^{*l}-\frac{1}{\Delta t}\left[\phi_{n+1}\left(\Omega\vec{W}^{*}\right)^{n+1}+\phi_{n}\left(\Omega\vec{W}\right)^{n}+\phi_{n-1}\left(\Omega\vec{W}\right)^{n-1}\right]+\frac{\partial\vec{R}}{\partial\vec{W}^{*}}\Delta\vec{W}^{*}-\phi_{n+1}\frac{\Omega^{n+1}}{\Delta t}\Delta\vec{W}^{*}$$

Where the ϕ coefficients are taken according to Table 1. By inserting equation (2.53) to equation (2.51) the following form is obtained:

$$\left[\frac{\Omega^{n+1}}{\Delta\tau}+\phi_{n+1}\frac{\Omega^{n+1}}{\Delta t}+\frac{\partial\vec{R}}{\partial\vec{W}^{*}}\right]\Delta\vec{W}^{*l}=-R^{l}-Q_{dual}^{l} \tag{2.54}$$

where Q_{dual}^{l} is a source-like term for the dual time stepping technique given by:

$$Q_{dual}^{l}=\frac{1}{\Delta t}\left[\phi_{n+1}\left(\Omega\vec{W}^{*}\right)^{n+1}+\phi_{n}\left(\Omega\vec{W}\right)^{n}+\phi_{n-1}\left(\Omega\vec{W}\right)^{n-1}\right] \tag{2.55}$$

The local time-stepping technique mentioned before is also used for defining the “pseudo-time” step.

2.3 Boundary Conditions

The implementation of correct boundary conditions has great significance as it can completely change the acquired solution. On external aerodynamics the following boundary conditions are used:

- Solid Wall Boundaries
- Far Field Boundaries
- Symmetry Boundaries
- Periodic Boundaries

Before introducing the specific boundary conditions, it is appropriate to discuss the concept of dummy cells. Dummy cells are virtual cells outside the computational domain and their purpose is to simplify the computation of the fluxes across the boundaries; this is achieved by extending the computational stencil for the spatial discretization outside the boundaries. As a result, the same discretization schemes can be used for the cells in the boundary with the cells inside the domain.

2.3.1 Solid Wall Boundary Conditions

For inviscid walls, since there is no friction, the boundary condition is that the velocity must be tangential to the surface and hence the velocity normal to the wall is zero.

$$\left(\vec{u} - \vec{u}_g\right)\vec{n} = 0 \tag{2.56}$$

Where $\vec{u}_g$ denotes the mesh velocity. Pressure is taken equal to that of the cell next to the wall.

$$p_w = p_I \tag{2.57}$$

On viscous walls the no-slip condition is applied, which states that there is no relative velocity between the surface and the fluid:

$$\vec{u} = \vec{u}_g \tag{2.58}$$

Density and pressure are taken equal to those of the cell next to the wall, as in the case of an inviscid flow.

For both viscous and inviscid walls the convective flux is taken equal to:

$$\vec{F}_{cwall} = \begin{pmatrix} 0 \\ n_x p_w \\ n_y p_w \\ n_z p_w \\ p_w V_g \end{pmatrix} \tag{2.59}$$

Where:

$$V_g = \vec{u}_g \vec{n} \tag{2.60}$$

2.3.2 Farfield Boundaries

A necessary requirement for the farfield boundary conditions is that they must not reflect back any outgoing disturbances [26]. This is extremely important for sub-sonic flows where due to the elliptic nature of the equations, incorrect implementation could result to reduced accuracy and slow convergence.

The sign of each eigenvalue of the flux Jacobian determines whether the information is going in or out of the computational domain. Kreiss [27] proved that the number of conditions that should be posed from outside the boundary is equal to the number of incoming characteristics. The remaining values must be determined from the solution inside the domain.

2.3.3 Symmetry Boundary Conditions

When the flow is symmetric with respect to a line or plane, the solution is computed only in a half of the domain. The necessary boundary conditions are: a) there is no flux across the symmetry line /plane, and b) gradients normal to the symmetry line /plane must be zero. These conditions are translated to the following mathematical form:

$$\begin{aligned} \vec{u} \cdot \vec{n} &= 0 \\ \nabla \vec{U} \cdot \vec{n} &= 0 \end{aligned} \tag{2.61}$$

2.3.4 Periodic Boundary Conditions

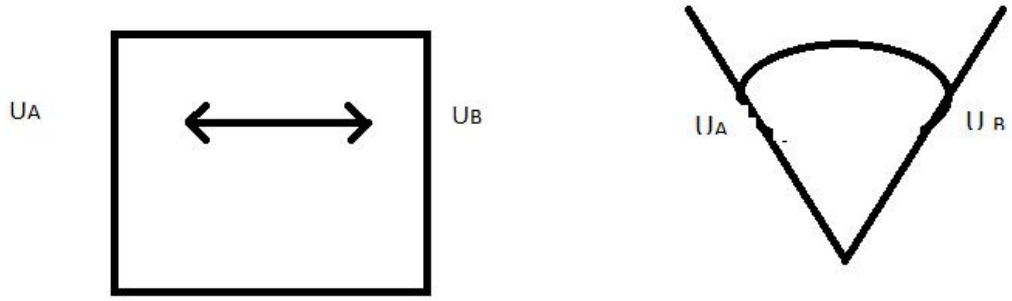

Figure 4. Translational periodicity (Left) and rotational periodicity (Right)

If flow field is periodic, the equations can be solved only in one period. There are two types of periodicity, the first one is translational periodicity, which means that the one periodic boundary can be transformed to the other periodic boundary by pure coordinate translation, while in the second type, rotational periodicity, the periodic boundary was generated with coordinate rotation.

The periodic boundary conditions are implemented by utilizing the dummy cell concept. Let A, B denote two periodic surfaces as illustrated in Figure 4. For every ghost cell in contact with A, there is a corresponding ghost cell in contact with B. Scalar quantities are taken to be equal on both dummy cells, while vector quantities follow the motion required to collapse A on B. This motion is expressed by a transformation matrix R_A, so:

$$U_A = U_B$$
$$\vec{U}_A = R_A \vec{U}_B \qquad (2.62)$$

In translational periodicity $R_A = I$, while for rotational periodicity R_A is the corresponding rotation matrix.

2.4 Turbulence Modelling

Turbulence is a state of motion where all the flow properties vary in a random and chaotic way. A necessary condition for the turbulent flow regime is that the Reynolds number, which gives a measurement of the relative importance of inertial forces and viscous forces, is above a certain value termed Critical Reynolds Number (Re_{cr}).

Turbulence is manifested by the appearance of rotational flow structures called turbulent eddies. These eddies have a wide range of characteristic length scales and velocity scales. For that reason, a direct simulation of the time-dependent Navier-Stokes equations called Direct Numeric Simulation (DNS) is impossible for complex flow structures, as the grid would have to be sufficiently fine to account for even the turbulent eddies with the smallest length scales and the time step would have to be sufficiently small in order to account for the fastest eddies.

A remedy for the above problem is presented by the Reynolds decomposition [28]. The flow quantities are split into their mean and fluctuating parts $u_i^{'}$, given respectively by the equations (2.63) and (2.64):

$$\overline{u}_i = \lim_{T\to\infty} \frac{1}{T} \int_t^{t+T} u_i dt \tag{2.63}$$

$$u_i = \overline{u}_i + u_i' \tag{2.64}$$

Where:

$$\overline{u_i'} = 0$$
$$\overline{u_i' u_j'} = 0$$

The above is known as Reynold's time averaging and is suitable for statistically stationary turbulence. In practice, $T \to \infty$ means that T must be larger than the typical time scale of the turbulent fluctuations.

By applying Reynolds averaging to the incompressible Navier-Stokes equations, the following equations for mass and momentum transfer are obtained:

$$\frac{\partial \overline{u}_i}{\partial x_i} = 0$$
$$\rho \frac{\partial \overline{u}_i}{\partial t} + \rho \overline{u}_j \frac{\partial \overline{u}_i}{\partial x_j} = -\frac{\partial \overline{p}}{\partial x_i} + \frac{\partial}{\partial x_j}\left(\overline{\tau}_{ij} - \rho \overline{u_i' u_j'}\right) \tag{2.65}$$

These are known as the Reynolds Averaged Navier Stokes equations (RANS). It can be easily observed that the RANS equations are in fact identical to the Navier Stokes equations for an incompressible fluid with the addition of one term:

$$\tau_{ij}^R = -\rho \overline{u_i' u_j'} \tag{2.66}$$

This term is denoted Reynolds stress tensor and expresses the transfer of momentum due to turbulent fluctuations. The laminar viscous stresses are computed according to Newton's Law of Viscocity, which states that the viscous stress tensor is proportional to the Strain Rate Tensor:

$$\overline{\tau}_{ij} = 2\mu \overline{S}_{ij} = \mu\left(\frac{\partial \overline{u}_i}{\partial x_j} + \frac{\partial \overline{u}_j}{\partial x_i}\right) \tag{2.67}$$

The Reynolds stress tensor consists in three dimensions, of nine components:

$$\tau_{ij}^{R} = \begin{pmatrix} \rho\overline{(u_1')^2} & \rho\overline{u_1'u_2'} & \rho\overline{u_1'u_3'} \\ \rho\overline{u_2'u_1'} & \rho\overline{(u_2')^2} & \rho\overline{u_2'u_3'} \\ \rho\overline{u_3'u_1'} & \rho\overline{u_3'u_2'} & \rho\overline{(u_3')^2} \end{pmatrix} \tag{2.68}$$

But since:

$$\overline{u_i'u_j'} = \overline{u_j'u_i'} \tag{2.69}$$

there are six additional unknowns, so six additional equations are needed in order to close the system.

The Boussinesq Approximation

The turbulence models implemented in MaPFlow are based on the Boussinesq Approximation or Eddy Viscosity Hypothesis [29], [30]. The Boussinesq Approximation states that the Reynolds stresses are proportional to mean rates of deformation, as in a laminar flow. The proportionality factor is called the Eddy Viscosity (μ_{T}):

$$\tau_{ij}^{R} = -\rho u_i'u_j' = 2\mu_t \overline{S}_{ij} - \frac{2}{3}\rho K \delta_{ij} \tag{2.70}$$

where $\overline{S}_{i,j}$ is the Reynolds Average strain rate tensor, K is the turbulent kinetic energy and δ_{ij} is Kronecker Delta ($\delta_{ij} = 1$ if $i = j$ and $\delta_{ij} = 0$ if $i \neq j$).

By introducing the Eddy Viscosity Hypothesis, the problem caused by the appearance of turbulent fluctuations on the governing equations is reduced to finding a suitable model for computing μ_{T} throughout the computational domain.

Menter k-ω SST Turbulence Model

Menter's K-ω SST (Shear Stress Transport) turbulence model [31], [32] is a first order closure of the Reynolds Averaged Navier Stokes Equations. It is a combination of Wilcox K-ω model [33], [34] and a high Reynolds number k-ε model (transformed in the k-ω formulation). More precisely, Wilcox K-ω model is used in the viscous sublayer of the boundary layer and in the logarithmic region. This is so because in the former case, the fact that the K-ω model doesn't need a damping function makes it more stable and in the latter case it gives better results for adverse pressure gradient flows and compressible flows. On the other hand, it switches to a K-ε model in the wake region of the boundary layer because the K-ω is sensitive to the freestream value of ω.

Two transport equations are solved, one for the turbulent kinetic energy K and one for the specific dissipation rate ω:

$$\frac{\partial \rho \mathrm{K}}{\partial t}+\frac{\partial}{\partial x_j}\left(\rho u_j K\right)=\frac{\partial}{\partial x_j}\left[\left(\mu_L+\sigma_{\mathrm{K}}\mu_{\mathrm{T}}\right)\frac{\partial \mathrm{K}}{\partial x_j}\right]+\tau_{ij}^F S_{ij}-\beta^* \rho\omega \mathrm{K}$$

$$\frac{\partial \rho \omega}{\partial t}+\frac{\partial}{\partial x_j}\left(\rho u_j \omega\right)=\frac{\partial}{\partial x_j}\left[\left(\mu_L+\sigma_{\omega}\mu_{\mathrm{T}}\right)\frac{\partial \omega}{\partial x_j}\right]+\frac{C_{\omega}\rho}{\mu_{\mathrm{T}}}\tau_{ij}^F S_{ij}-\beta\rho\omega^2+2\left(1-F_1\right)\frac{\rho\sigma_{\omega 2}}{\omega} \quad (2.71)$$

Where τ_{ij}^F is the Favre averaged turbulent stress tensor and S_{ij} is the stress tensor.

The second term on the right-hand side of the first equation is the Production term. In [35] it is recommended to limit the production term so it is replaced by:

$$P=\min\left(P, \beta^* \rho k \omega\right) \quad (2.72)$$

F_1 is a blending function between the k-ω and the k-ε models that takes the form of a hyperbolic tangent in order to ensure smooth transition between the two models. In particular, if $F_1=1$, the model is K-ε and if $F_1=0$ the model is K-ω.

F_1 is given by:

$$F_1=\tanh(\arg_1^4)$$
$$\arg_1=\min\left[\max\left(\frac{\sqrt{k}}{\beta\omega d}, \frac{500\nu}{\omega d^2}, \frac{4\rho\sigma_{\omega 2}\kappa}{CD_{K\omega}d^2}\right)\right] \quad (2.73)$$

Where $\nu=\frac{\mu_{\mathrm{T}}}{\rho}$, d is the distance to the nearest wall and $CD_{k\omega}$ is the positive part of the cross-diffusion term in (2.78) given by:

$$CD_{k\omega}=\max\left(2\rho\sigma_{\omega 2}\frac{1}{\omega}\frac{\partial k}{\partial x_j}\frac{\partial \omega}{\partial x_j}, 10^{-20}\right) \quad (2.74)$$

F_1 is also used as a blending function for the empirical constants of the models with ϕ_1 presenting the values of the constants for the k-ω model (hereby stated as inner constants) and ϕ_2 the values of the constants for Menter's modified k-ε model (hereby stated as outer constants).

$$\phi=F_1\phi_1+\left(1-F_1\right)\phi_2 \quad (2.75)$$

The inner constants are:

$$\gamma_1 = \frac{\beta_1}{\beta^*} - \frac{\sigma_{\omega 1}\kappa^2}{\sqrt{\beta^*}}$$
$$\sigma_{\kappa 1} = 0.85$$
$$\sigma_{\omega 1} = 0.5$$
$$\beta_1 = 0.075 \qquad (2.76)$$
$$\beta^* = 0.09$$
$$\kappa = 0.41$$
$$\alpha_1 = 0.31$$

The outer constants are:

$$\gamma_2 = \frac{\beta_2}{\beta^*} - \frac{\sigma_{\omega 2}\kappa^2}{\sqrt{\beta^*}}$$
$$\sigma_{\kappa 2} = 1$$
$$\sigma_{\omega 2} = 0.856 \qquad (2.77)$$
$$\beta_2 = 0.0828$$

The eddy viscosity is limited because it was observed that it overpredicts the wall shear stress and the implementation of a limiter gives better agreement with experimental measurements of separated flows [36]:

$$\mu_T = \frac{\rho\alpha_1 k}{\max(\alpha_1\omega, F_2\Omega)} \qquad (2.78)$$

Where Ω is the vorticity magnitude and F_2 is another blending function:

$$\Omega = \sqrt{2W_{ij}W_{ij}}$$
$$W_{ij} = \frac{1}{2}\left(\frac{\partial u_j}{\partial x_i} - \frac{\partial u_i}{\partial x_j}\right)$$
$$F_2 = \tanh(\arg_2) \qquad (2.79)$$
$$\arg_2 = \max\left(\frac{2\sqrt{k}}{\beta^*\omega d}, \frac{500\nu}{d^2\omega}\right)$$

Boundary Conditions

The boundary conditions as defined by [31], are for the freestream values:

$$\omega_\infty = C_1 \frac{U_\infty}{L}$$
$$k_\infty = \frac{10^{-C_2} U_\infty^2}{\mathrm{Re}_L} \tag{2.80}$$

Where L is the approximate length of the computational domain and $1 \le C_1 \le 10$ and $1 \le C_2 \le 5$.

The values on the wall are:

$$\omega_{wall} = \frac{6\nu}{\beta_1 d_1^2}$$
$$k_{wall} = 0 \tag{2.81}$$

With d_1 being the distance to the closest wall.

2.5 Transition Modelling. The $\gamma - \mathrm{Re}_\theta$ model.

Transition models are a family of turbulence models that account for the effect of the transitional region between the laminar and the turbulent flow. They are very effective on aerodynamic applications, where a significant proportion of the boundary layer is laminar.

For the scope of this work, the $\gamma - \mathrm{Re}_\theta$ model was used. It was proposed by Langtry and Menter [37], and further corrected in [38], [39], [40]. This model can be derived from the K-ω SST model discussed in Section 2.4 by modifying the Source (P_K) and Destruction (D_K) term in equation 2.78, as well as the blending function F_1. The modifications are as follows:

$$\widetilde{P}_K = \gamma P_K \tag{2.82}$$

$$\widetilde{D}_K = \min\left(\max\left(\gamma, 0.1\right), 1.0\right) D_K \tag{2.83}$$

The variable γ in the above equations is called Turbulence Intermittency and expresses the percentage of time that turbulent fluctuations are present in the boundary layer. This variable determines whether the flow is laminar ($\gamma = 0$), fully turbulent ($\gamma = 1$) or transitional from laminar

to turbulent ($0 < \gamma < 1$). It should be noted that the freestream value of γ is taken equal to 1 instead of 0 as someone would except. In other words, the flow is considered turbulent everywhere except of a small laminar region.

On equation 2.81 when $\gamma = 1$ the destruction term takes its original value from the K-ω model and when $\gamma = 0$ (laminar region), $\widetilde{D}_K = 0.1 D_K$. This limiter ensures that the destruction term never falls under 10% of its original value, so the wall still damps turbulence even if the flow is laminar.

The blending function F_1 was modified because it was observed that in the boundary layer it would sometimes switch to 0 (which means that the turbulence model becomes K-ε); this must be avoided because the K-ε model does not have good near-wall behaviour. The modified blending function is given by:

$$\begin{aligned} F_1 &= \max\left(F_{1,original}, F_3\right) \\ F_3 &= e^{-\left(\frac{R_{dw}}{120}\right)^8} \\ R_{dw} &= \frac{\rho d_w \sqrt{k}}{\mu} \end{aligned} \tag{2.84}$$

The turbulence Intermittency is calculated at every point of the computational domain using a transport equation:

$$\frac{\partial(\rho\gamma)}{\partial t} + \frac{\left(\rho U_j \gamma\right)}{\partial x_J} = P_\gamma - \mathrm{E}_\gamma + \frac{\partial}{\partial x_J}\left[\left(\mu + \frac{\mu_t}{\sigma_f}\frac{\partial\gamma}{\partial x_J}\right)\right] \tag{2.85}$$

The production of intermittency P_γ is given by:

$$P_\gamma = F_{length}\left[\gamma F_{onset}\right]^{0.5} c_{a1}\rho S\left(1 - c_{e1}\gamma\right) \tag{2.86}$$

Where S is the magnitude of the strain rate, F_{length} is an empirical correlation that controls the strength of production and hence the length of the transition region and F_{onset} is a function that switches on the production of intermittency. Both functions are dimensionless.

The destruction term of equation 2.92 is given by:

$$E_\gamma = c_{a2}\rho\Omega\gamma F_{turb}\left(c_{e2}\gamma - 1\right) \tag{2.87}$$

Where Ω is the vorticity magnitude.

F_{onset} is given by:

$$
\begin{aligned}
F_{onset} &= \max\left(F_{onset2} - F_{onset3}, 0\right) \\
F_{onset3} &= \max\left(1 - \left(\frac{R_T}{2.5}\right)^3, 0\right) \\
F_{onset2} &= \min\left(\max\left(F_{onset1}, F_{onset1}^4\right), 2.0\right) \\
F_{onset1} &= \frac{\mathrm{Re}_v}{2.193\,\mathrm{Re}_{\theta c}}
\end{aligned}
\tag{2.88}
$$

$$
F_{turb} = e^{-\left(\frac{R_T}{4}\right)^4} \tag{2.89}
$$

$$
\mathrm{Re}_v = \frac{\rho d_w^2}{\mu} S \tag{2.90}
$$

$$
R_T = \frac{\rho k}{\mu \omega} \tag{2.91}
$$

$$
\begin{aligned}
\mathrm{Re}_{\theta c} &= f\left(\widetilde{\mathrm{Re}}_{\theta t}\right) \\
F_{length} &= f\left(\widetilde{\mathrm{Re}}_{\theta t}\right)
\end{aligned}
\tag{2.92}
$$

The constants appearing in the above equations are given by:

$$
\begin{aligned}
c_{e1} &= 1.0 \\
c_{a1} &= 2.0 \\
c_{e2} &= 50 \\
c_{a2} &= 0.06 \\
\sigma_f &= 1.0
\end{aligned}
\tag{2.93}
$$

In the variation of the $\gamma - \mathrm{Re}_\theta$ model, which is utilized in MaPFlow, γ in equation (2.82) is replaced with γ_{eff} given by:

$$
\gamma_{eff} = \max\left(\gamma, \gamma_{sep}\right) \tag{2.94}
$$

Where γ_{sep} is a modified turbulence intermittency in order to predict separation induced transition:

$$\gamma_{sep} = \min\left(s_1 \max\left[0, \left(\frac{\mathrm{Re}_v}{3.235\,\mathrm{Re}_{\theta c}} \right) - 1 \right] F_{reattach}, 2 \right) F_{\theta t}$$

$$F_{reattach} = e^{-\left(\frac{R_T}{20}\right)^4} \qquad (2.95)$$

$$s_1 = 2$$

At this point, it is appropriate to define the Momentum tThickness Reynolds Number:

$$\mathrm{Re}_\theta = \frac{\rho U \theta}{\mu} \qquad (2.96)$$

Where θ is the momentum thickness defined as:

$$\theta = \int_0^\infty \frac{\rho U}{\rho_\infty U_\infty} \left(1 - \frac{U}{U_\infty} \right) dy \qquad (2.97)$$

On equations 2.92 $\mathrm{Re}_{\theta c}$ is the Momentum Thickness Reynolds Number where the intermittency starts to increase, while $\mathrm{Re}_{\theta t}$ is the Momentum Thickness Reynolds Number where transition occurs. Both of these values are calculated from empirical correlations.

Furthermore, the $\gamma - \mathrm{Re}_\theta$ model introduces a new variable $\widetilde{\mathrm{Re}}_{\theta t}$, which has a value at every point in the computational domain and is called Transition Momentum Thickness Reynolds Number. A transport equation is solved for $\widetilde{\mathrm{Re}}_{\theta t}$ given by:

$$\frac{\partial\left(\rho\,\widetilde{\mathrm{Re}}_{\theta t}\right)}{\partial t} + \frac{\partial\left(\rho U_J\,\widetilde{\mathrm{Re}}_{\theta t}\right)}{\partial x_J} = P_{\theta t} + \frac{\partial}{\partial x_J}\left[\sigma_{\theta t}\left(\mu + \mu_t\right) \frac{\partial\,\widetilde{\mathrm{Re}}_{\theta t}}{\partial x_J} \right] \qquad (2.98)$$

It should be noted that this transport equation diffuses this value towards the wall in contrary to conventional transport equations, where the effect of the wall diffuses out away from the wall. More specifically, the freestream value of $\widetilde{\mathrm{Re}}_{\theta t}$, which is computed with the aid of an empirical correlation diffuses towards the wall. This is made possible by the form of the production term $P_{\theta t}$:

$$P_{\theta t} = c_{\theta t} \frac{\rho}{t} \left(\mathrm{Re}_{\theta t} - \widetilde{\mathrm{Re}}_{\theta t} \right) \left(1.0 - F_{\theta t} \right) \qquad (2.99)$$

$$t = \frac{500\mu}{\rho U^2} \qquad (2.100)$$

Where t is a time scale, which is used for dimensional reasons.

The production term is used to set the value of $\widetilde{\mathrm{Re}}_{\theta t}$ equal to the freestream value, everywhere except from the boundary layer. This is made possible by the blending function $F_{\theta t}$.

In the boundary layer this blending function takes the value of one and consequently the production term is set to zero, while at the freestream it takes the value of zero, forcing $\widetilde{\mathrm{Re}}_{\theta t}$ to become equal to the freestream value. The limiter is obtained by the following equation:

$$
\begin{aligned}
F_{\theta t} &= \min\left(\max\left(F_{wake} e^{-\left(\frac{-y}{\delta}\right)^4}, 1.0 - \left(\frac{\gamma - \frac{1}{c_{e2}}}{1.0 - \frac{1}{c_{e2}}} \right)^2 \right), 1.0 \right) \\
\theta_{BL} &= \frac{\widetilde{\mathrm{Re}}_{\theta t}\, \mu}{\rho U} \\
\delta_{BL} &= \frac{15}{2}\theta_{BL} \\
\delta &= \frac{50 \Omega d_w}{U}\delta_{BL} \\
\mathrm{Re}_{\omega} &= \frac{\rho \omega d_{\omega}^2}{\mu} \\
F_{wake} &= e^{-\left(\frac{\mathrm{Re}_{\omega}}{1\mathrm{E}+5}\right)^2}
\end{aligned}
\tag{2.101}
$$

F_{wake} is a blending function that ensures that $F_{\theta t}$ is not activated in the wake regions downstream of solid bodies. Lastly, the constants in the above equations are taken as follows:

$$
\begin{aligned}
c_{\theta t} &= 0.03 \\
\sigma_{\theta t} &= 2.0
\end{aligned}
\tag{2.102}
$$

Empirical Correlations for $\mathrm{Re}_{\theta c}$ and F_{length}

For the calculation of $\mathrm{Re}_{\theta c}$ and F_{length} the Langtry correlations [38] were used:

$$
F_{length} = \left\{ \begin{array}{c}
\left[398.189 \cdot 10^{-1} + \left(-119.270 \cdot 10^{-4} \right) \widetilde{\mathrm{Re}}_{\theta t} + \left(-132.567 \cdot 10^{-6} \right) \widetilde{\mathrm{Re}}_{\theta t}^2 \right], \widetilde{\mathrm{Re}}_{\theta t} < 400 \\
\left[263.404 + \left(-123.939 \cdot 10^{-2} \right) \widetilde{\mathrm{Re}}_{\theta t} + \left(194.548 \cdot 10^{-5} \right) \widetilde{\mathrm{Re}}_{\theta t}^2 + \left(-101.695 \cdot 10^{-8} \right) \widetilde{\mathrm{Re}}_{\theta t}^3 \right], 400 \le \widetilde{\mathrm{Re}}_{\theta t} < 596 \\
\left[0.5 - \left(\widetilde{\mathrm{Re}}_{\theta t} - 596.0 \right) \cdot 3.0 \cdot 10^{-4} \right], 596 \le \widetilde{\mathrm{Re}}_{\theta t} < 1200 \\
\left[0.3188 \right], 1200 \le \widetilde{\mathrm{Re}}_{\theta t}
\end{array} \right\}
\tag{2.103}
$$

$$\mathrm{Re}_{\theta c} = \begin{Bmatrix} \left[\begin{array}{l} \tilde{\mathrm{R}}\mathrm{e}_{\theta t} - \left(396.035 \cdot 10^{-2}\right) + \left(-120.656 \cdot 10^{-4}\right)\tilde{\mathrm{R}}\mathrm{e}_{\theta t} + \left(868.230 \cdot 10^{-6}\right)\tilde{\mathrm{R}}\mathrm{e}_{\theta t}^{2} \\ + \left(-696.506 \cdot 10^{-9}\right)\tilde{\mathrm{R}}\mathrm{e}_{\theta t}^{3} + \left(174.105 \cdot 10^{-12}\right)\tilde{\mathrm{R}}\mathrm{e}_{\theta t}^{4} \end{array} \right], \tilde{\mathrm{R}}\mathrm{e}_{\theta t} \le 1870 \\ \left[\tilde{\mathrm{R}}\mathrm{e}_{\theta t} - 593.11 + \left(\tilde{\mathrm{R}}\mathrm{e}_{\theta t} - 1870.0\right) \cdot 0.482\right], \tilde{\mathrm{R}}\mathrm{e}_{\theta t} > 1870 \end{Bmatrix} \quad (2.104)$$

Langtry also proposed a correction on F_{length} [40]:

$$\begin{aligned} F_{length} &= F_{length}\left(1 - F_{sublayer} + 40.0 F_{sublayer}\right) \\ F_{sublayer} &= e^{-\left(\frac{R_w}{0.4}\right)^2} \\ R_w &= \frac{\rho d_w^2 \omega}{500\mu} \end{aligned} \quad (2.105)$$

Empirical Correlations for $\mathrm{Re}_{\theta t}$

On the available correlations the $Re_{\theta t}$ is a function of the Turbulence Intensity (Tu) and of a scalar measure of the Steamwise Pressure Gradient (λ_θ).

The Turbulence Intensity is given by:

$$Tu = 100\frac{\sqrt{\frac{2k}{3}}}{U} \quad (2.106)$$

While λ_θ is given by:

$$\lambda_\theta = \frac{\rho\theta^2}{\mu}\frac{dU}{ds} \quad (2.107)$$

On the above equation $\frac{dU}{ds}$ is the velocity gradient along the steamwise direction and it is computed by:

$$U = \sqrt{u^2 + v^2 + w^2} \quad (2.108)$$

$$\frac{dU}{dx}=\frac{1}{2}\left(u^2+v^2+w^2\right)^{-\frac{1}{2}}\cdot\left[2u\frac{du}{dx}+2v\frac{dv}{dx}+2w\frac{dw}{dx}\right]$$
$$\frac{dU}{dy}=\frac{1}{2}\left(u^2+v^2+w^2\right)^{-\frac{1}{2}}\cdot\left[2u\frac{du}{dy}+2v\frac{dv}{dy}+2w\frac{dw}{dy}\right]$$
$$\frac{dU}{dz}=\frac{1}{2}\left(u^2+v^2+w^2\right)^{-\frac{1}{2}}\cdot\left[2u\frac{du}{dz}+2v\frac{dv}{dz}+2w\frac{dw}{dz}\right] \quad (2.109)$$
$$\frac{dU}{ds}=\left[\left(u/U\right)\frac{dU}{dx}+\left(v/U\right)\frac{dU}{dy}+\left(w/U\right)\frac{dU}{dz}\right]$$

In this work the Langtry Empirical Correlation [38] was used as follows:

$$\mathrm{Re}_{\theta t}=\left\{\begin{array}{c}\left[1173.51-589.428Tu+\frac{0.2196}{u^2}\right]F(\lambda_\theta),Tu\le 1.3\\ 331.50\left[Tu-0.5658\right]^{-0.671}F(\lambda_\theta),Tu>1.3\end{array}\right\} \quad (2.110)$$

$$F\left(\lambda_\theta\right)=\left\{\begin{array}{c}1-\left[-12.986\lambda_\theta-123.66\lambda_\theta^2-405.689\lambda_\theta^3\right]e^{-\left[\frac{Tu}{1.5}\right]^{1.5}},\lambda_\theta\le 0\\ 1+0.275\left[1-e^{[-35.0\lambda_\theta]}\right]e^{\left[\frac{-Tu}{0.5}\right]},\lambda_\theta>0\end{array}\right\} \quad (2.111)$$

2.6 Solution of the Discretised Equations

The result of the discretization process is that the partial differential governing equations are transformed into a system of algebraic equations in the following form:

$$AX=B \quad (2.112)$$

In order to solve this system of equations a numerical process has to be employed. There are two families of numerical solution methods, direct and indirect or iterative methods. Direct methods are accurate but for a large system of equations the computational cost is high, as well as the memory requirements, because for a system of N unknowns and N equations, all NxN coefficients need to be stored. On the other hand, iterative solvers begin with an initial guessed solution and by the repeated application of a simple algorithm, the solution in constantly improved until a certain criterion of convergence is met. Iterative methods have limited memory requirements, since only the non-zero coefficients need to be stored; also, they are suitable for parallel computing while direct methods are not.

As shown before, the discretized form of the equations for cell *I* is:

$$\left[\frac{(D)_I}{\Delta t_i}+\left(\frac{\partial R}{\partial U}\right)_I\right]\Delta\vec{U}_I^n=-\vec{R}_I^n \tag{2.113}$$

In these equations only one term refers to the cell in consideration, so the following splitting is employed:

$$D_I\Delta\vec{U}_I^n+O_I\sum\Delta\vec{U}_J^n=-\vec{R}_I^n \tag{2.114}$$

The first term in equation 2.121 is block diagonal and includes the variables at cell *I*, while the second term contains the off-diagonal contributions, that involve the cells surrounding cell *I.*

Jacobi Iterative solver

The Jacobi iterative solver begins with an initial solution and it improves it in every iteration using the following formula:

$$D_I\Delta\vec{U}_I^{n,k+1}=-\vec{R}_I^n-O_I\sum\Delta\vec{U}_J^{n,k} \tag{2.115}$$

Where k is the number of the current iteration.

Gauss - Seidel Solver

Another iterative solver is the Gauss Seidel solver, it is very similar to Jacobi's method, with the exception that the off-diagonal terms are calculated with the current update for $\vec{U}$:

$$D_I\Delta\vec{U}_I^{n,k+1}=-\vec{R}_I^n-O_I\sum\Delta\vec{U}_L^{n,k+1}-O_I\sum\Delta\vec{U}_J^{n,k} \tag{2.116}$$

Where $\vec{U}_L$ contains the cell values that have already been updated in the $k+1$ iteration.

The Gauss Seidel method can prove to be twice as fast as the Jacobi Method, however its performance is strongly dependent on the form of matrix *A* in equation 2.112.
If the matrix is banded, which means that its non-zero entries are confined in a diagonal band, the matrix can be split in an Upper and Lower part and Gauss - Seidel becomes:

$$D_I\Delta\vec{U}_I^{n,k+1}=-\vec{R}_I^n-O_I\sum\Delta\vec{U}_L^{n,k+1}-O_I\sum\Delta\vec{U}_R^{n,k} \tag{2.117}$$

However, if *A* is considerably sparse the Gauss Seidel solver has the same convergence properties as the Jacobi Solver [41].

In the case of structured meshes, the matrix is inherently banded and the Gauss - Seidel method behaves well. In unstructured grids, the bandwidth of the matrix depends on the cell numbering, by using the Reverse Cuthill-McKee (RCM) reordering scheme [*ibid*], the bandwidth of the matrix is effectively reduced, and the performance of the Gauss- Seidel method is enhanced.

Chapter Three: Model Validation

3.0 Introduction

In this chapter three different test cases were simulated using the MaPFlow solver and the obtained results were compared to experimental values available in the literature. The three test cases consist of two airfoils typically used in the wind energy sector, namely the DU-00-W212 and the S827, in addition to the VP1304 controllable pitch propeller. Section 3.1 pertains to the DU-00-W212 airfoil, with Subsection 3.1.1 briefly discussing the experimental layout and Subsection 3.1.2 the results of the simulation. Similarly, for the S827 airfoil the experimental procedure is described on Subsection 3.2.1 and the simulation results are presented on Subsection 3.2.2. Finally, the VP1304 propeller is discussed on Section 3.3.

3.1 DU-00-W212 airfoil at $\mathrm{Re} = 6 \cdot 10^6$

3.1.1 Experiment Layout

The DU-00-W212 airfoil was measured by O. Ceyhan and O. Pires [42] at the DNW-HDG low speed wind tunnel at Goettingen, Germany.

The layout of the wind tunnel can be seen on Figure 5.

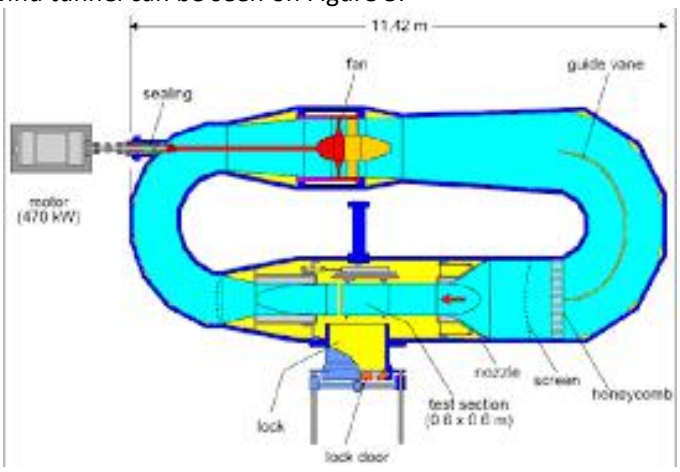

Figure 5. The DNW-HDG wind tunnel. Source: [42]

The model was tested on different Reynolds numbers ranging from 3 x 10^6 to 15 x 10^6, while maintaining the Mach Number below 0.1, in order to determine the effect of different Reynolds numbers on the aerodynamic characteristics. The Lift Coefficients were computed by integrating the

pressure distribution over the airfoil, which was measured by 90 pressure taps mounted on the model. The drag coefficients were computed with the aid of a wake rake. A wake rake is a device that measures the static pressure and the total pressure on the wake region of the flow. By comparing these values to the free stream total pressure, the pressure deficit is computed, which is directly related to the profile drag. The rake on this experiment had 118 total and 8 static pressure probes and it moved with a traversing mechanism across the width of the wake. Experimental results were obtained for both free transition and transition tripping. However, in this work only the 'clean condition' results were utilized.

3.1.2 Results

Using the MaPFlow software two simulations were carried out, one with the compressible solver and one with the incompressible, with the Reynolds number set to 6 x 10^6 and employing the $\gamma - \mathrm{Re}_\theta$ transition model. The Mach number was set to 0.15 and therefore it is expected that the compressible and incompressible solvers will have similar results, since for this Mach number there must be minimal compressibility effects.

For both simulations two structured, body-fitted O-grids were utilized. The coarser grid, hereby referred to as "Grid L2"consists of 4 blocks with around 4600 cells each, the finer grid, hereby referred to as "Grid L1", also consists of 4 blocks but with around 18500 cells in each block. The outer boundary on both meshes was put 40 chords away from the airfoil in order to avoid far-field influence on the calculations.

Figures 5 and 6 illustrate the L2 grid, while figures 7 and 8 illustrate the L1 grid.

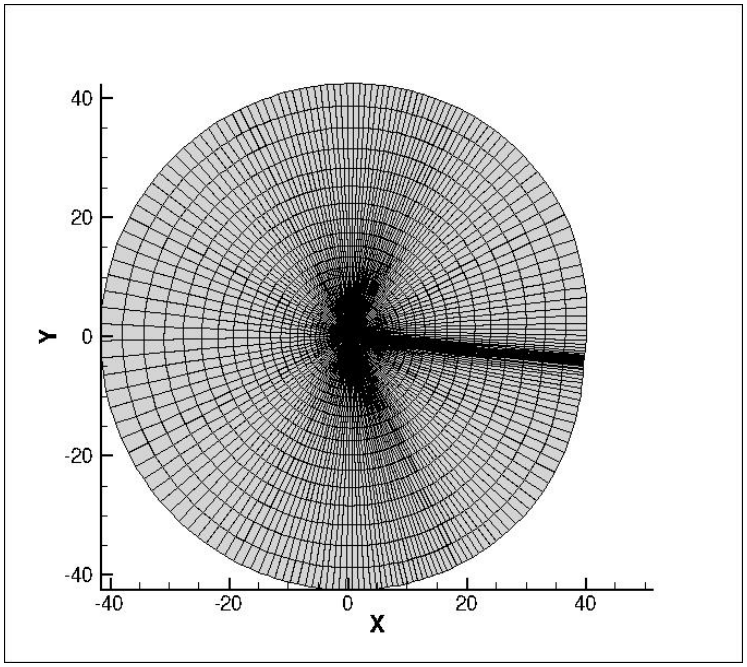

Figure 6. Grid L2: View of the entire mesh

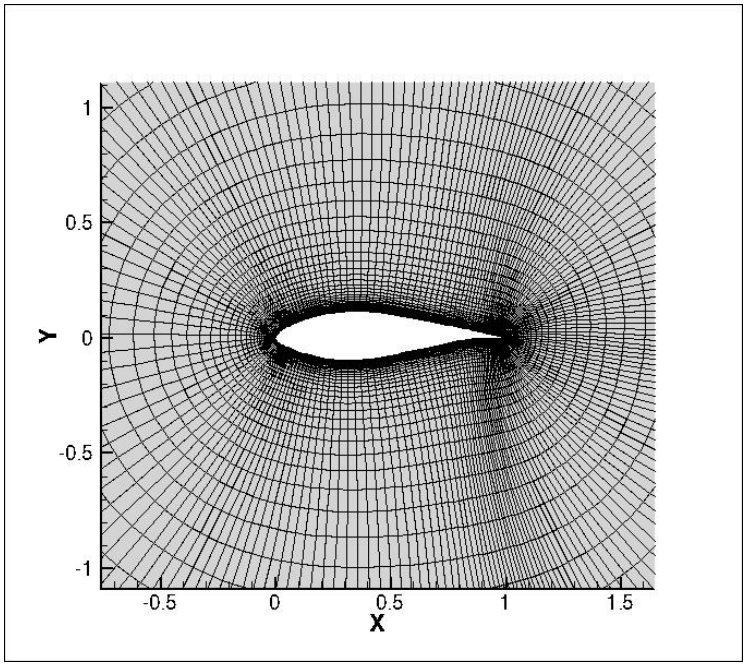

Figure 7. Grid L2: View of near airfoil region

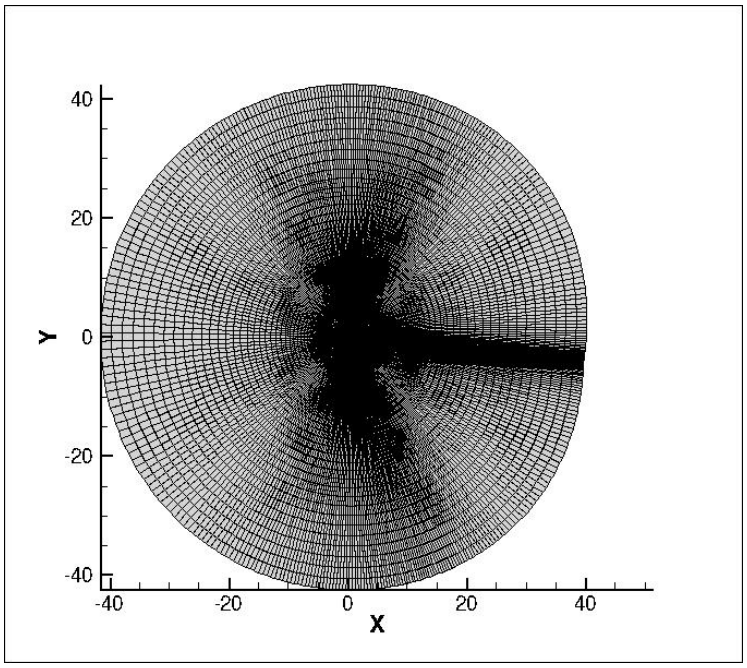

Figure 8. Grid L1: View of the entire mesh

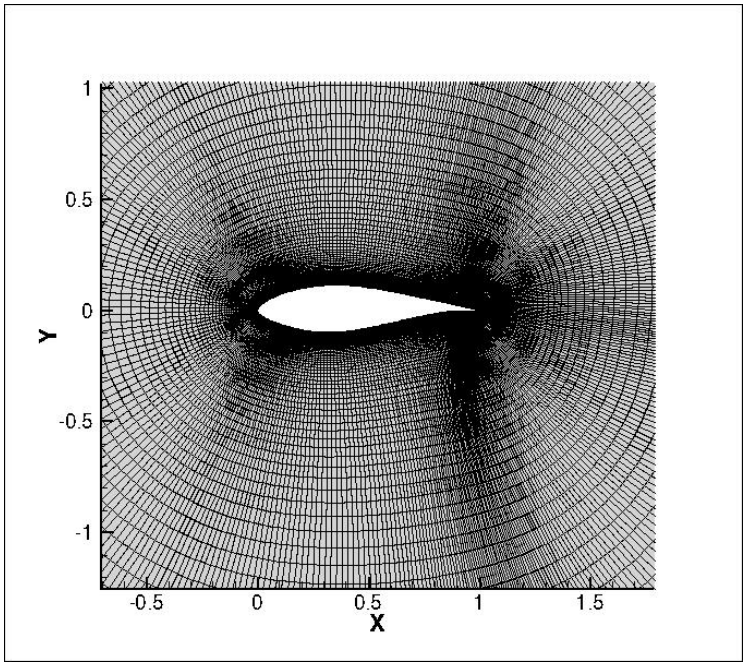

Figure 9. Grid L1: View of near airfoil region

Grid Independence Study

Prior to obtaining any results, a grid independence study was performed between the L2 and L1 meshes for the incompressible solver. Figure 9 illustrates the computed Lift Coefficients in respect to the angle of attack, while figure 10 shows the drag coefficient in respect to the lift coefficient.

The results for the Lift Coefficient were satisfactory, however there was a deviation between the results of the two meshes in the CL-CD diagram, which indicates that Grid Independence has not been achieved. Due to limited time and computer resources for creating a mesh finer than L1, it was decided that the results for the L1 grid will be used for the validation of the model.

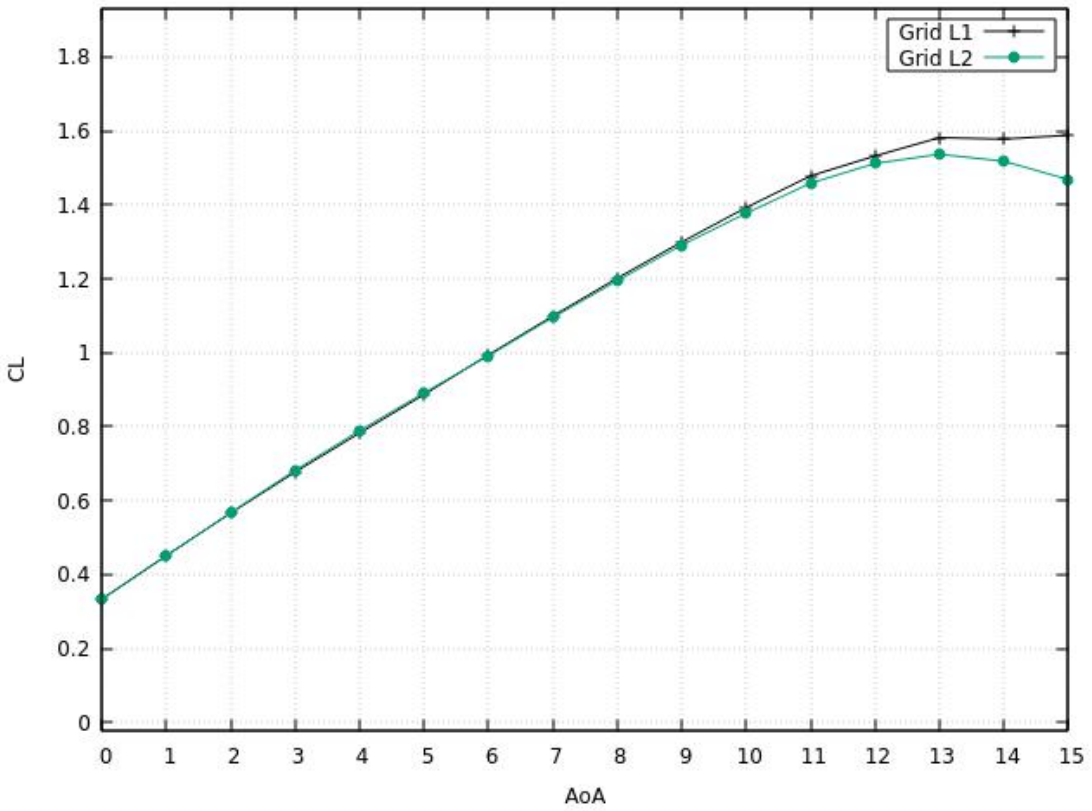

Figure 9. Grid Independence Study CL-α

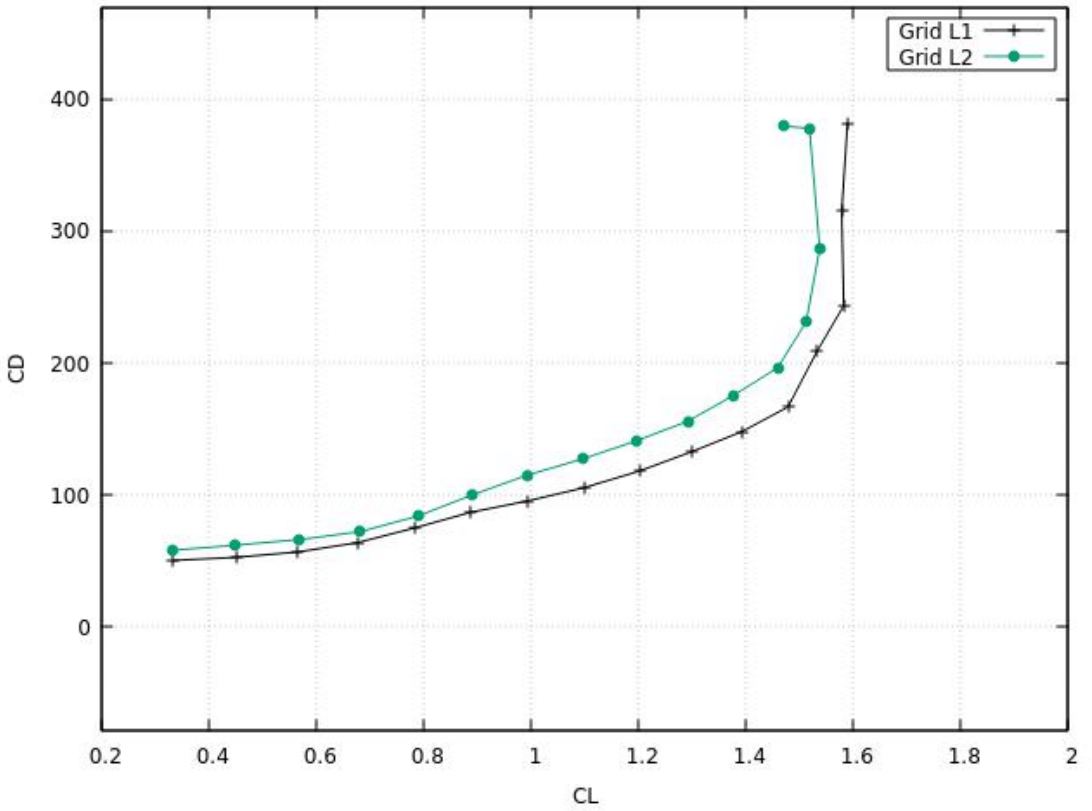

Figure 10. Grid Independence Study CL-CD

Simulation Results

Figure 11 shows the Lift Coefficient against the Angle of Attack for the incompressible and compressible solvers, in addition to the result from the experimental measurements. Generally, it can be seen that the agreement with the experimental results is satisfactory for both solvers.
More precisely, both solvers slightly overpredict the Lift Coefficient for angles from 0 degrees to 5 degrees, then for angles from 5 degrees to 8 degrees the results are quite similar with the experimental ones. For angles from 9 degrees and above there is a discrepancy with the experimental results, which becomes larger as the Angle of Attack increases, and eventually the solvers do not capture the reduction of the Lift Coefficient in the post-stall region. This discrepancy between the experimental results and the ones from the simulations for higher Angles of Attack, can be attributed to strong three-dimensional effects that can't be captured by a two-dimensional solver. For the same reason, simulations for angles greater than 15 degrees were not performed although experimental results are available.

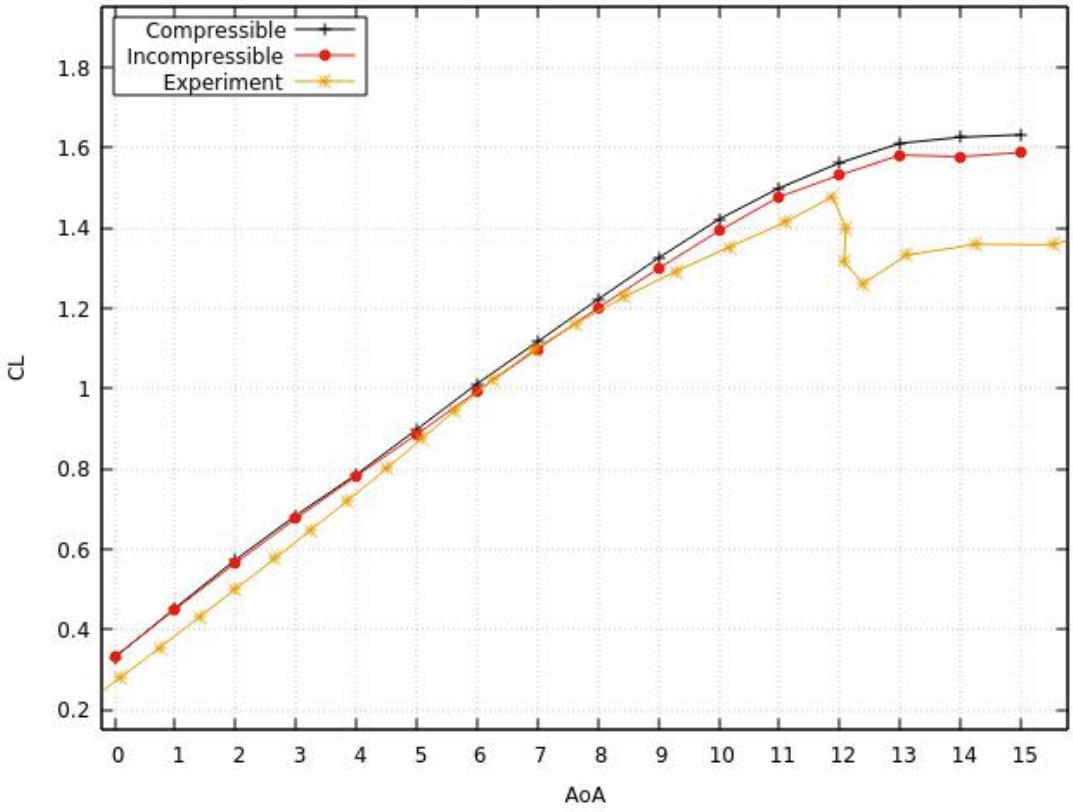

Figure 11. Lift coefficient against Angle of Attack

The Lift Coefficients against the Drag Coefficients are presented on figure 12. Although the results of the incompressible and the compressible solvers are in good agreement with each other, there is a serious deviation from the experimental results, especially outside the laminar drag bucket, where for high lift coefficients the drag coefficient is extremely overpredicted. Again, this can be a result of strong three-dimensional phenomena on higher Angles of Attack. It can also be seen that the models predict transition at lower values of CL.

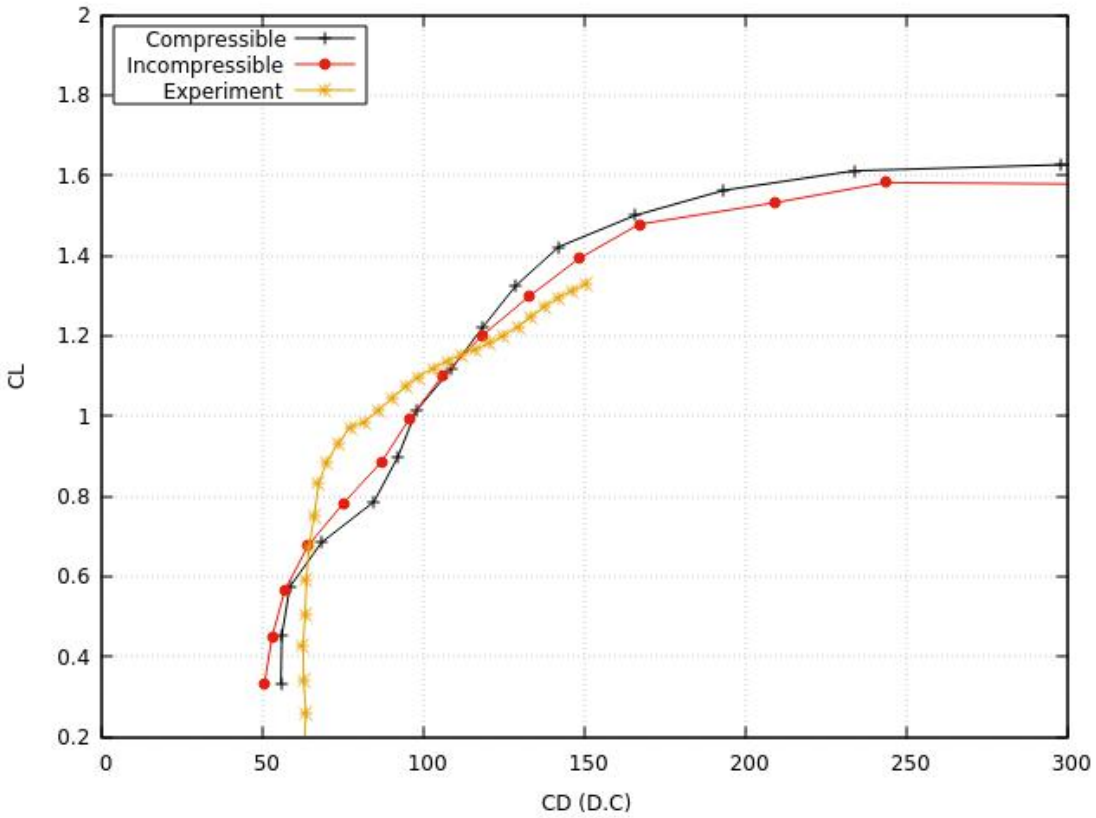

Figure 12. Lift Coefficient Against Drag Coefficient (in Drag Counts)

Figure 13 Illustrates the surface pressure distribution on the pressure and suction sides of the airfoil for an Angle of Attack of 2 degrees. Both solvers correctly predict the pressure distribution with the exception of a region on the pressure side ranging from 0.2 to 0.4 of the chord, where there is a slight deviation between the results from the solvers and the experimental ones. This region is connected with the transition from laminar flow to turbulence and it can be seen that the transition is earlier on the CFD simulation results.

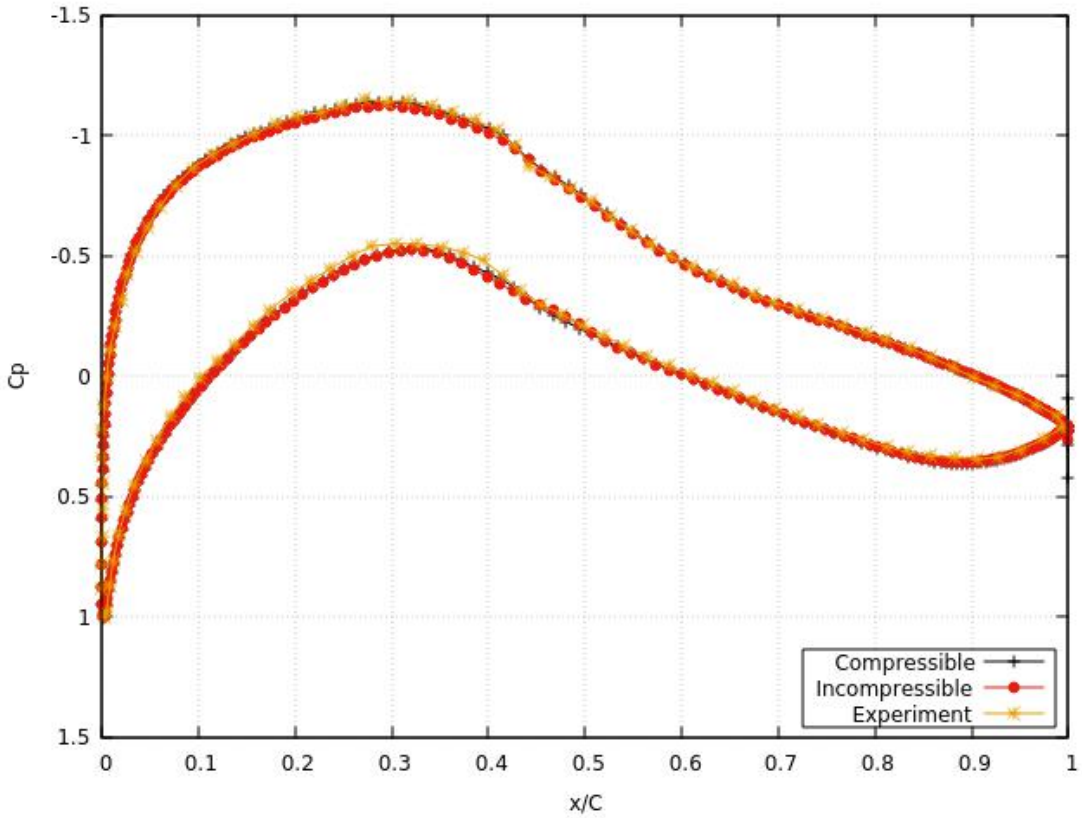

Figure 13. Pressure Distribution for 2 degrees AoA

Lastly, since there were no experimental results available for the transition locations on all Angles of Attack, a comparison was made between the results of the Compressible and Incompressible Solvers. The following figures (14-21) illustrate the transition locations for the upper and lower sides of the airfoil, as well as the effective turbulence intermittency contours in these locations for 0, 4 and 8 degrees Angle of Attack.

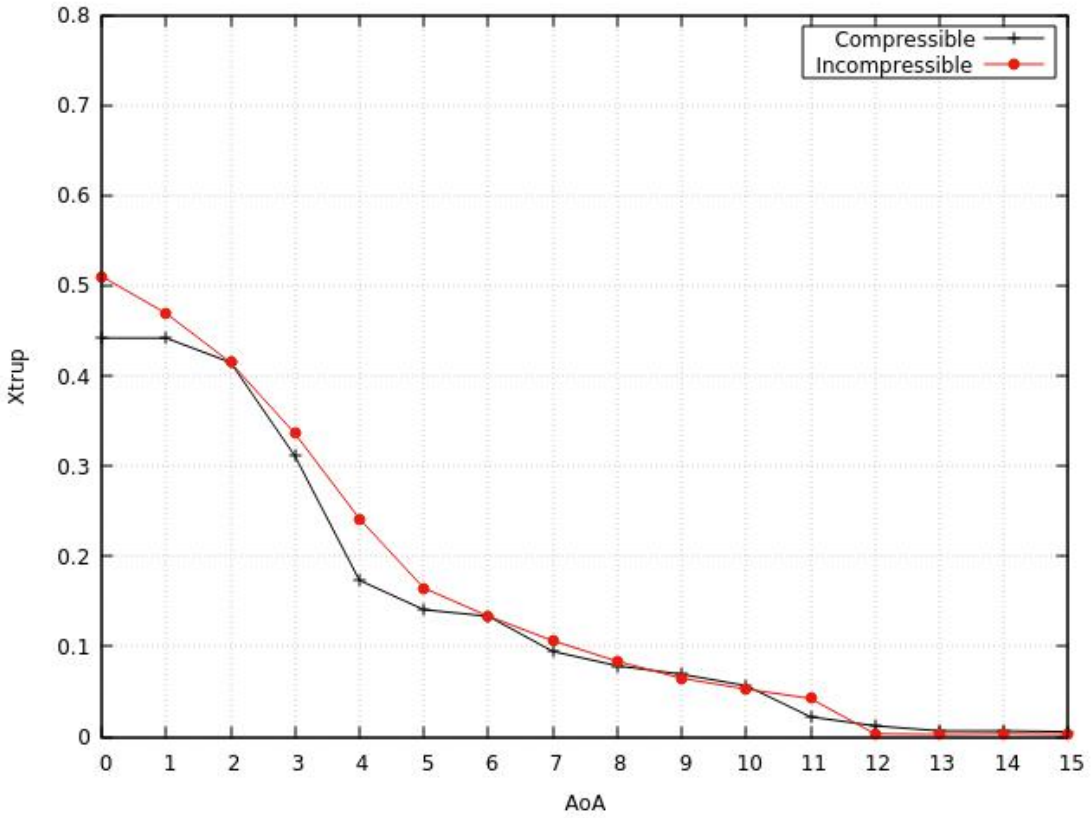

Figure 14. Transition Location Upper Side

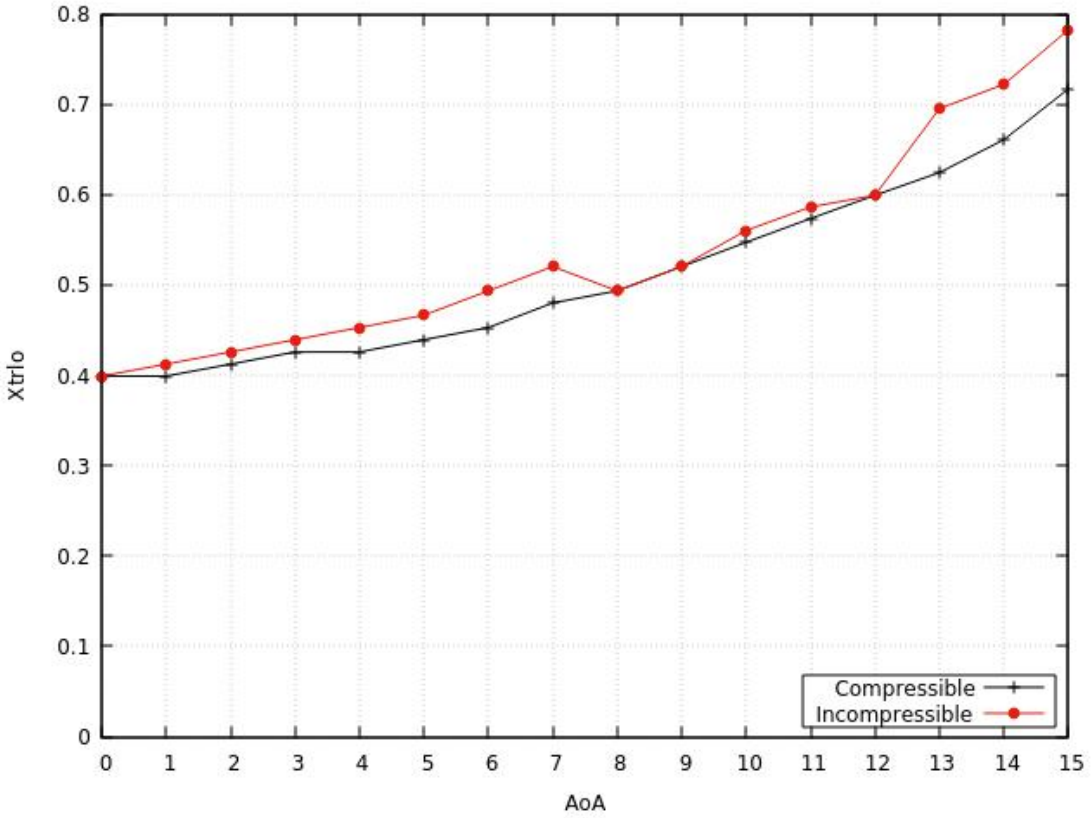

Figure 15. Transition Location Lower Side

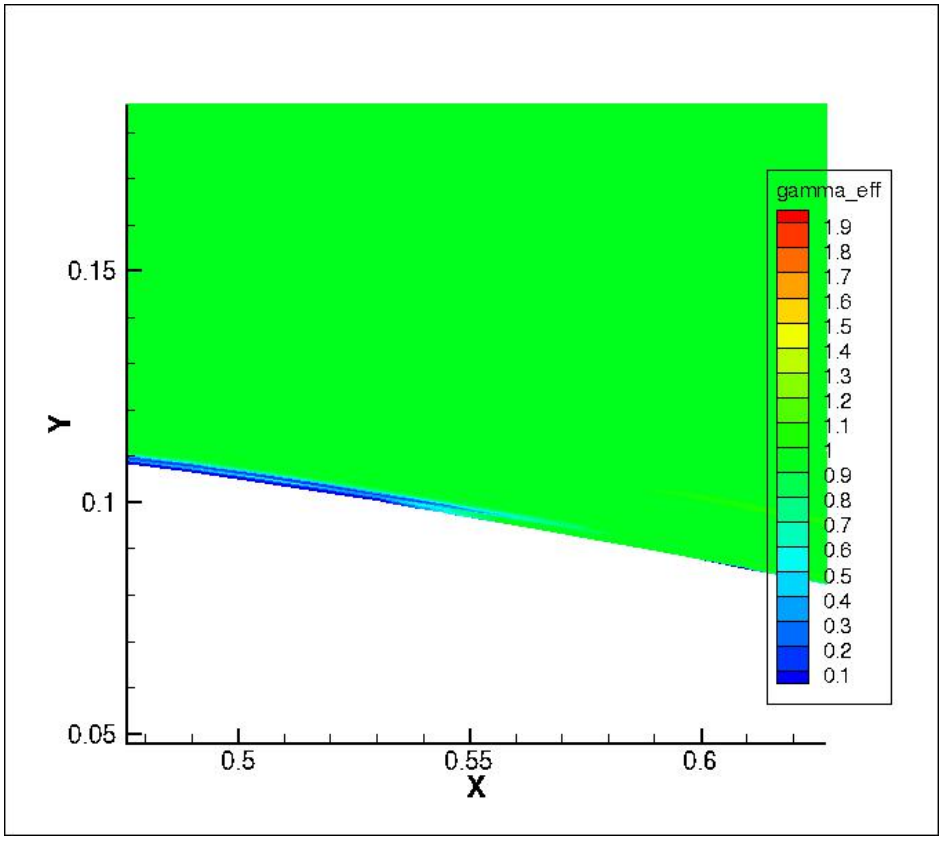

Figure 16. Effective Turbulence Intermittency (γ_{eff}) 0 degrees Upper Side

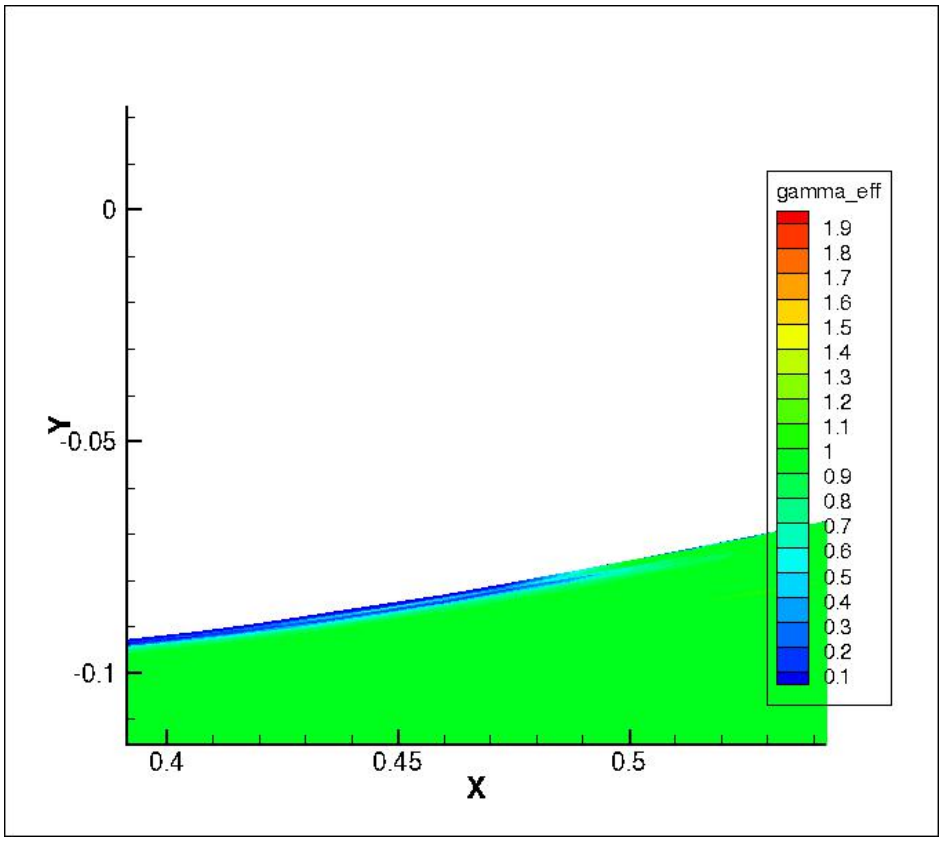

Figure 17. Effective Turbulence Intermittency (γ_{eff}) 0 degrees Lower Side

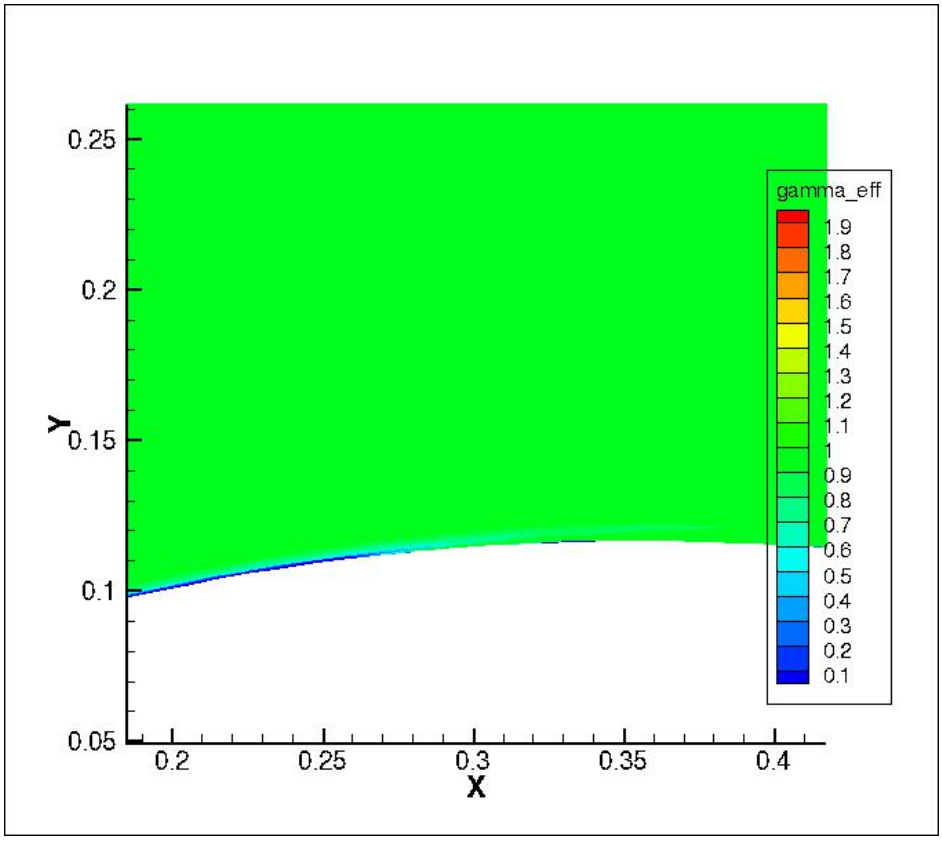

Figure 18. Effective Turbulence Intermittency (γ_{eff}) 4 degrees Upper Side

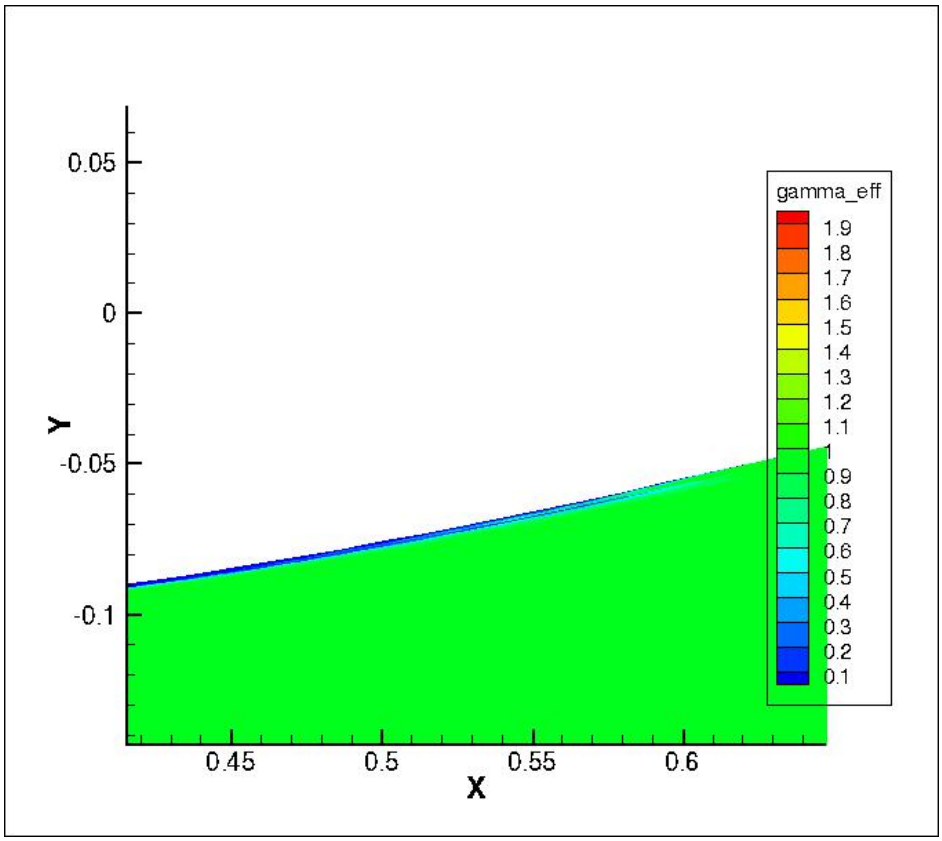

Figure 19. Effective Turbulence Intermittency (γ_{eff}) 4 degrees Lower Side

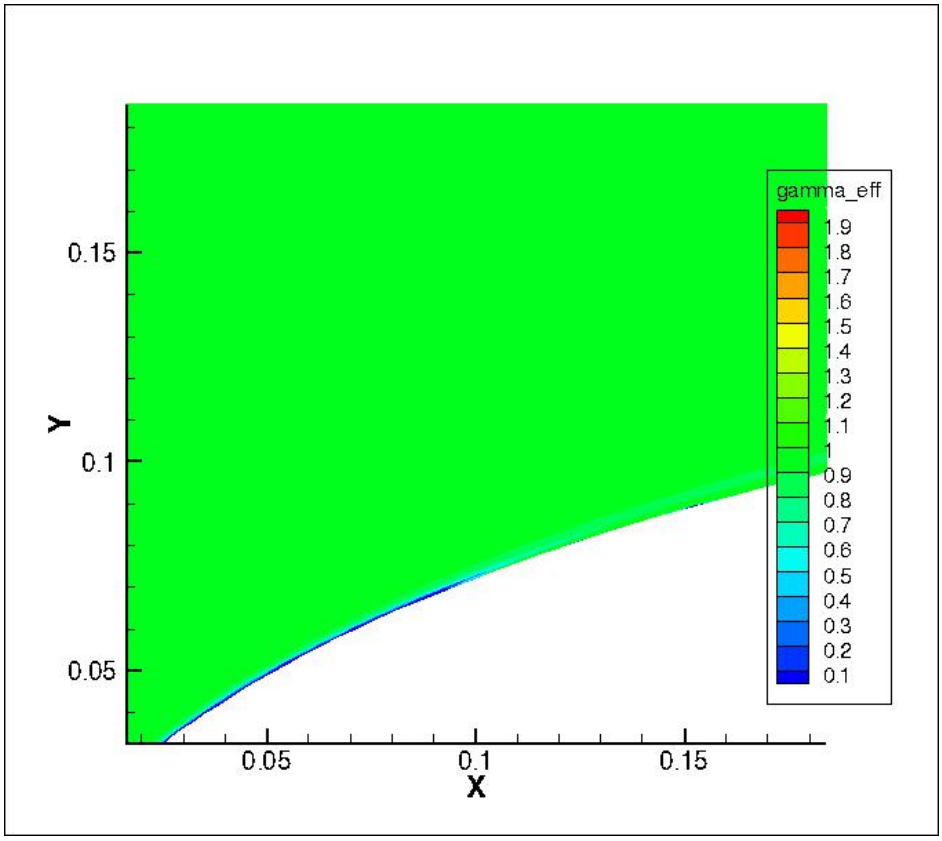

Figure 20. Effective Turbulence Intermittency (γ_{eff}) 8 degrees Upper Side

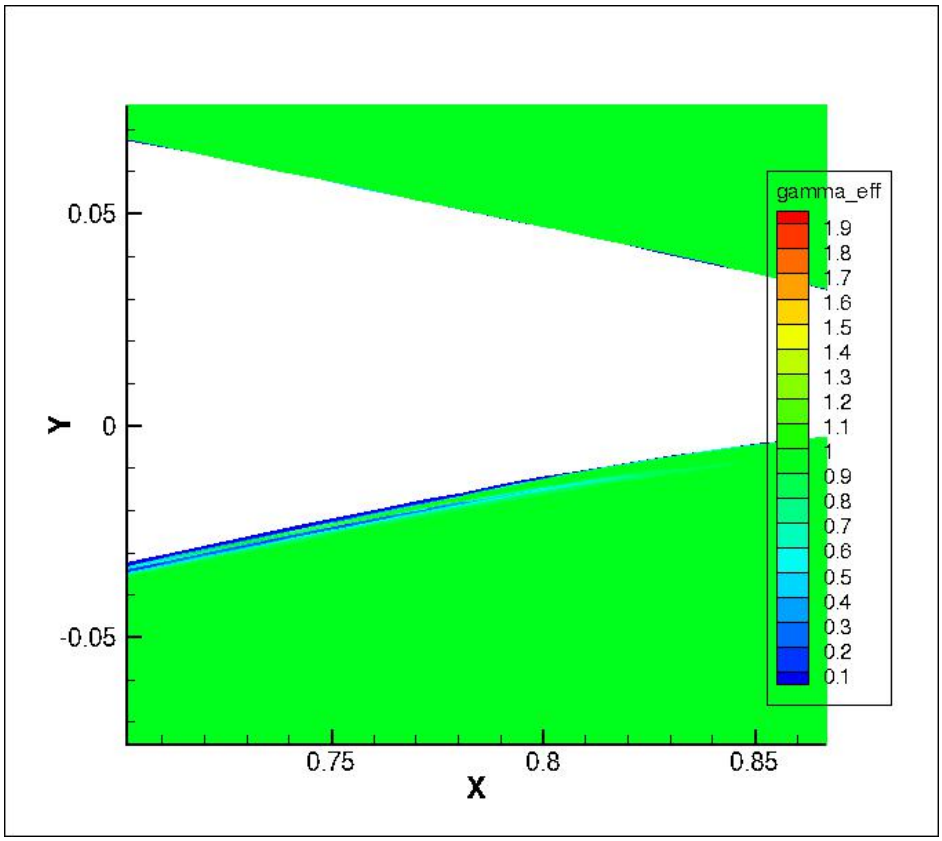

Figure 21. Effective Turbulence Intermittency (γ_{eff}) 8 degrees Lower Side

3.2 S827 airfoil at $\mathrm{Re} = 3 \cdot 10^6, 4 \cdot 10^6$

3.2.1 Experiment Layout

The S827 airfoil is a NREL S-series airfoil, which is frequently used for transition model validation, since its physics are strongly characterized by transition phenomena. It was designed and measured by Sommers [43] at the Nasa-Langley Low Turbulence Pressure Tunnel (LTPT). The airfoil was designed so it will deliberately stall above a certain Angle of Attack, but the separation is restricted to a 'separation ramp'. As a result, the airfoil can continue operating with reduced lift giving the $C_L - a$ diagram a distinctive two-slope form.

The airfoil was designed to meet a certain desired pressure distribution profile using the Eppler Airfoil Design and Analysis Code. The LTPT is illustrated in figure 21. Experimental tests were made for Reynolds Numbers ranging from $1 \cdot 10^6$ to $6 \cdot 10^6$ with both free transition and transition fixed by roughness, however for in this work only the free transition results were utilized. Like for the DU-00-W212 airfoil a wake survey probe was used to calculate the Drag Coefficient, while the Lift Coefficients and the pressure distribution at various angles were measured with the aid of pressure transducers mounted on the model.

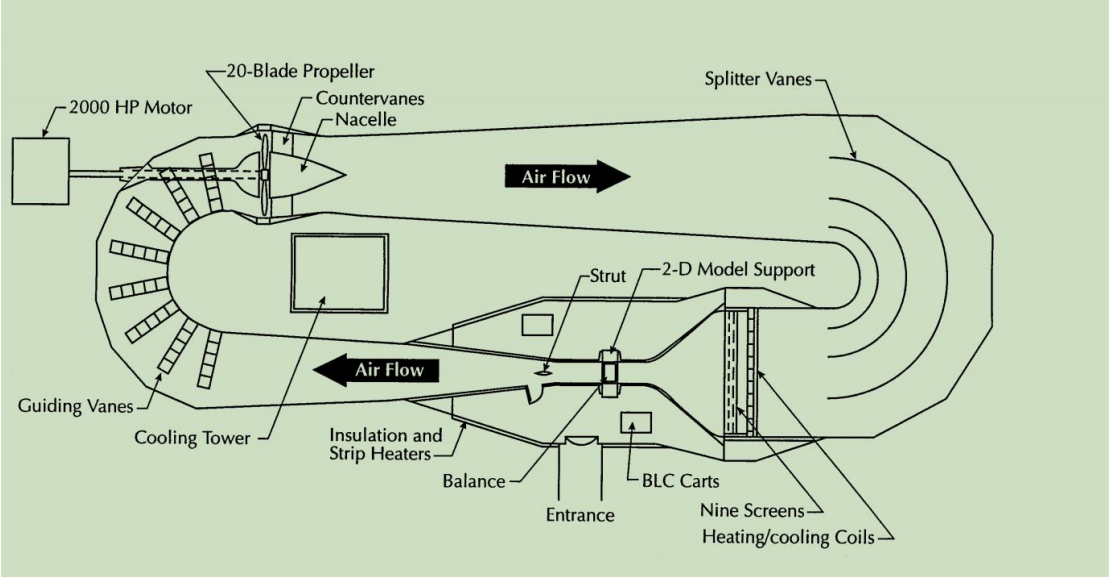

Figure 22. The Nasa-Langley Low Turbulence Pressure Tunnel

3.2.2 Results

Simulations were carried out using only the Incompressible solver for $\mathrm{Re} = 3 \cdot 10^6$ and $\mathrm{Re} = 4 \cdot 10^6$. For both Reynolds Numbers the Mach Number was set to 0.1 and the freestream turbulence intensity to 0.05%.

For both Reynolds Numbers a structured, body-fitted O-grid was utilized. The grid consists of 4 blocks with approximately 32800 cells each and the far-field boundary was put 1000 chords away from the airfoil. The grid can be seen on figures 23 and 24.

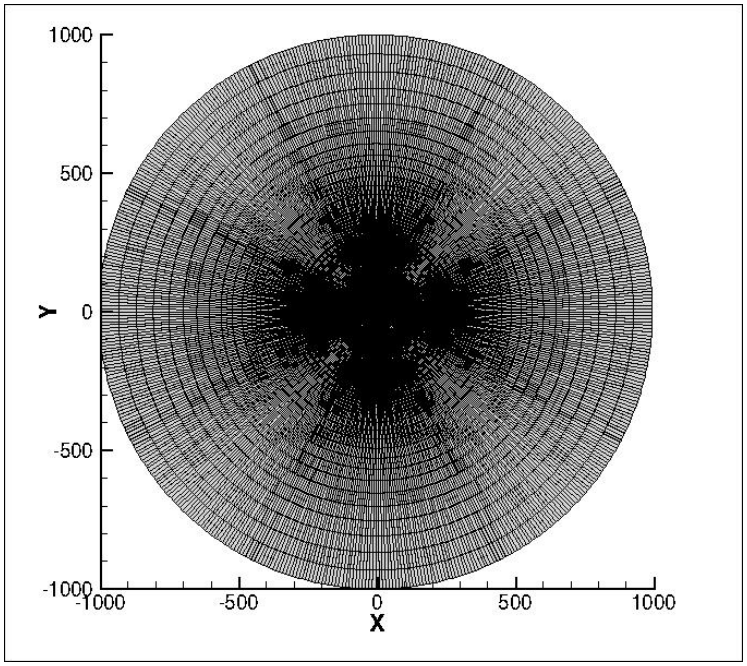

Figure 23. View of the Entire Mesh

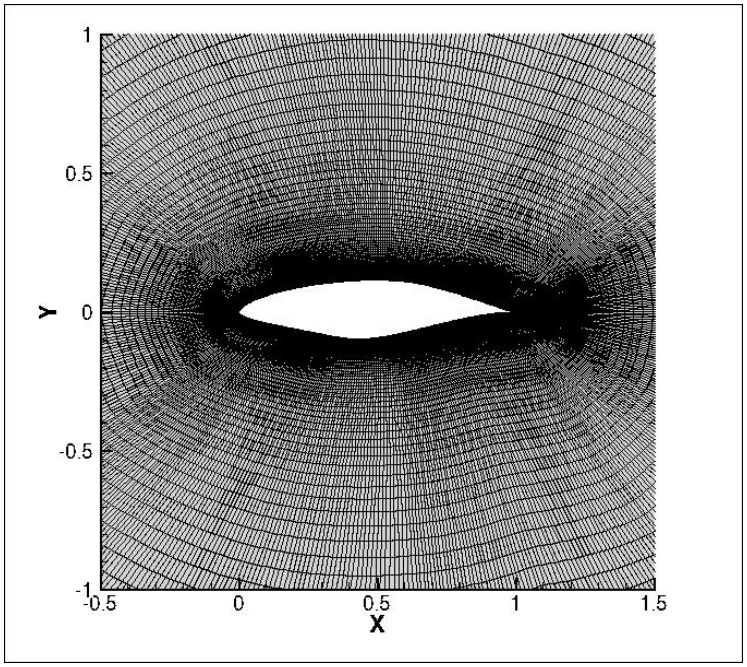

Figure 24. View of near airfoil region

The Lift Coefficients for Angles of Attack ranging from -5 to 20 degrees are plotted on figure 25 for $\mathrm{Re} = 3 \cdot 10^6$. The results are quite close to the experiment for angles from -4 to 4 with the model slightly overpredicting the slope of the curve. The results of the model indicate that the Angle of Attack for stall onset and the maximum Lift Coefficient predicted are significantly higher than the corresponding experimental results. For Angles of Attack greater than 10 the deviation from the experimental results increases as the Angle of Attack increases and eventually the model fails to capture the two-slope form of the experimental results, although it correctly captured the increase of

the Lift Coefficient after the stall. Like for the DU-00-W212 airfoil, the failure to capture the characteristics of the flow at higher Angles of Attack can be attributed to strong three-dimensional phenomena.

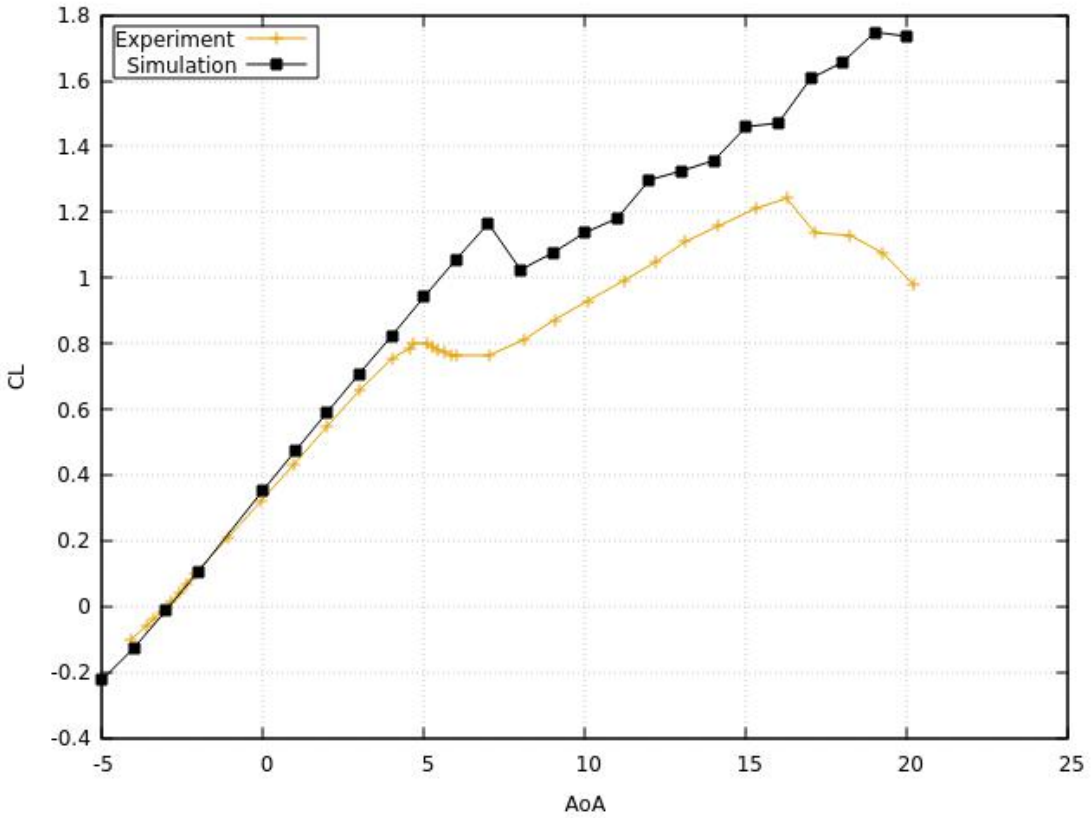

Figure 25. Lift Coefficient against Angle of Attack

The drag polars for $\mathrm{Re} = 3 \cdot 10^6$ are presented on Figure 26. The agreement with the experimental results is good inside the laminar drag bucket. However, the model significantly overpredicts the Lift Coefficient at which boundary layer transition to mostly turbulent flow starts. This was expected, due to the results for the lift coefficient presented above, as the model predicts the onset of stall at higher values of CL. Experimental results for high Angles of Attack were not taken into account for reasons explained above.

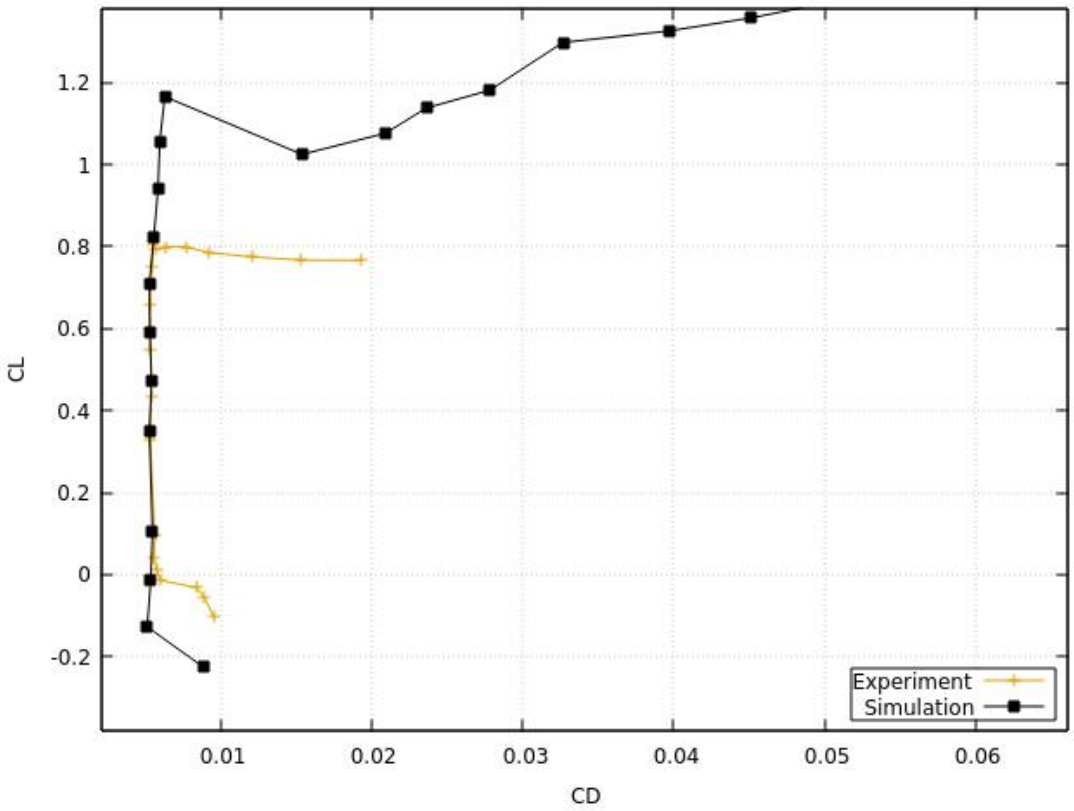

Figure 26. Lift Coefficient against Drag Coefficient

For $\mathrm{Re} = 4 \cdot 10^6$ the results of the simulation presented similar behavior with the ones for $\mathrm{Re} = 3 \cdot 10^6$. More precisely, for lower Angles of Attack the results were in good agreement with the experiment. The solver overpredicts the angle at which stall occurs, as well as the maximum Lift Coefficient. From 19 degrees to 20 degrees the Lift Coefficient is reduced, which might indicate that if simulations were made for angles of attack greater than 20, the solver might have been able to capture the characteristic two-slope form. The results are presented on Figure 27.

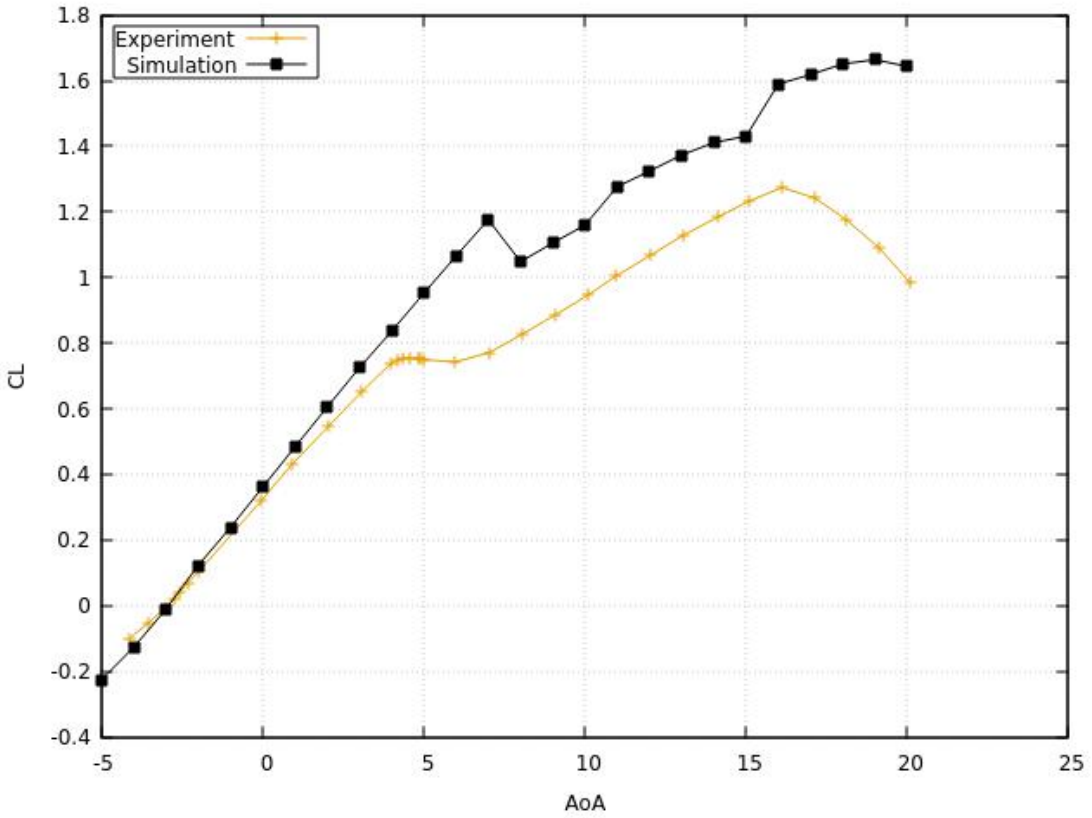

Figure 27. Lift Coefficient against Angle of Attack

The drag polars for $\mathrm{Re} = 4 \cdot 10^6$ are presented on Figure 28. The model, also in this case, overpredicts the Lift Coefficient at the Drag Bucket Corner.

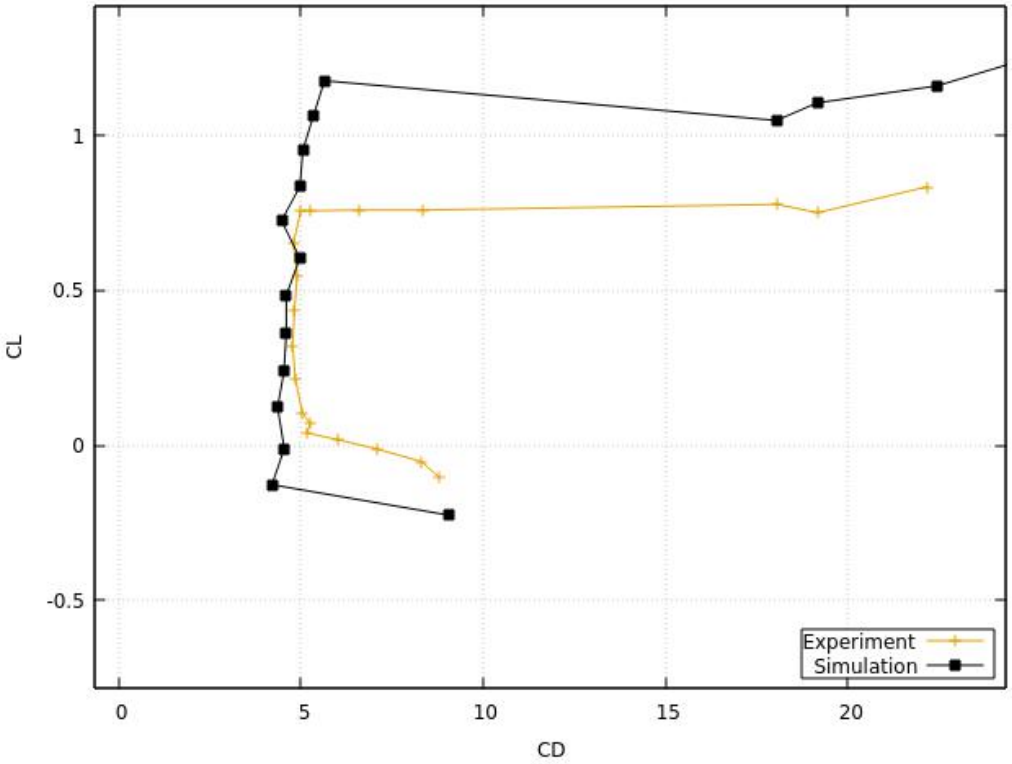

Figure 28. Lift Coefficient against Drag Coefficient

The surface pressure distributions for Angles of Attack 0, 4 and 8 are plotted on figures 29-31.

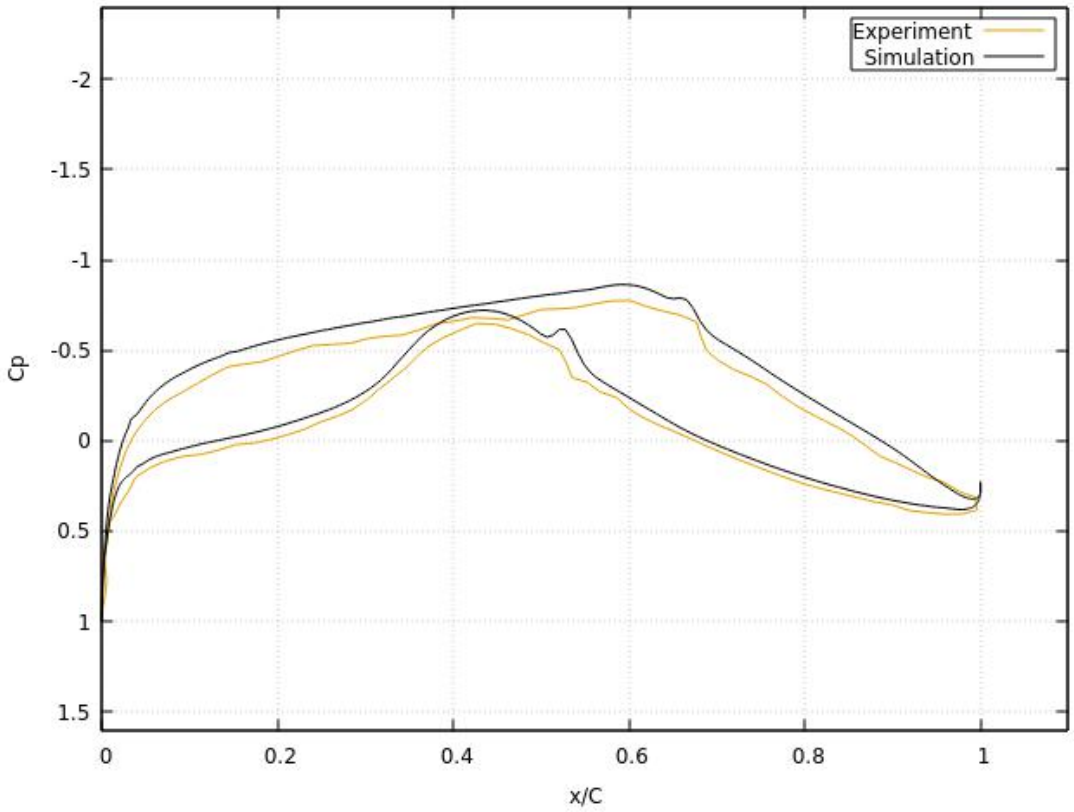

Figure 29. Surface Pressure Distribution, 0 degrees

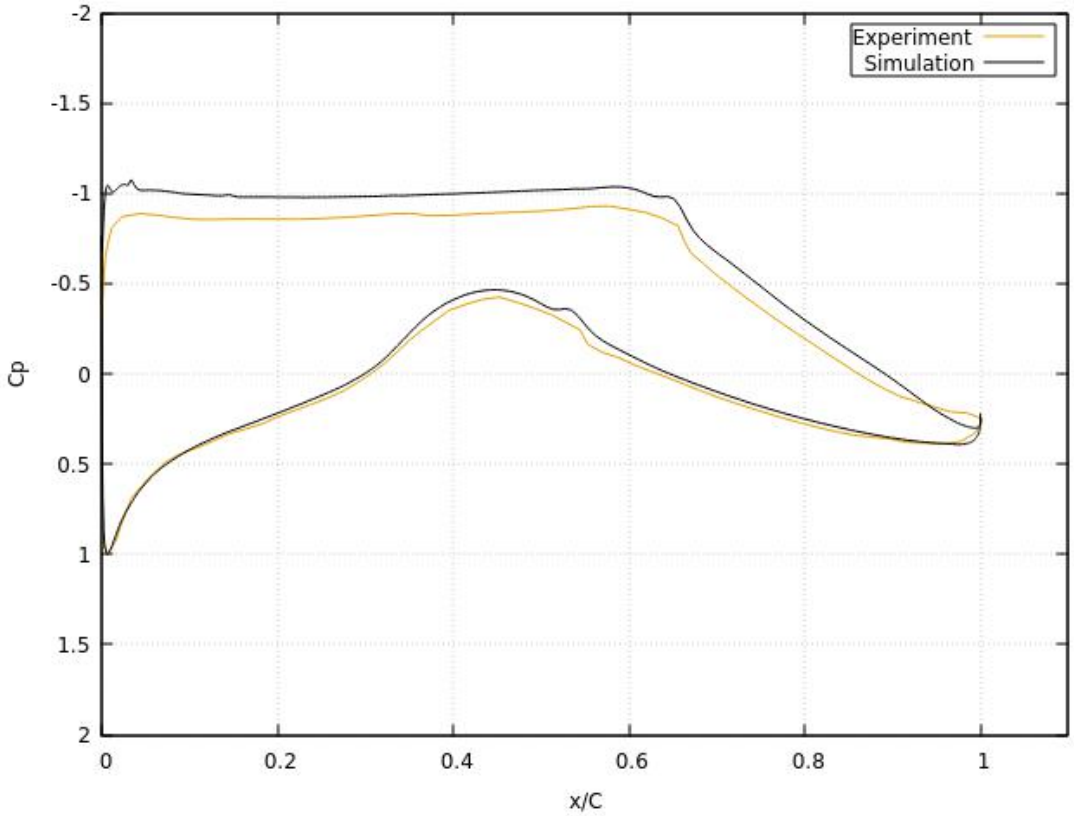

Figure 30. Surface Pressure Distribution, 4 degrees

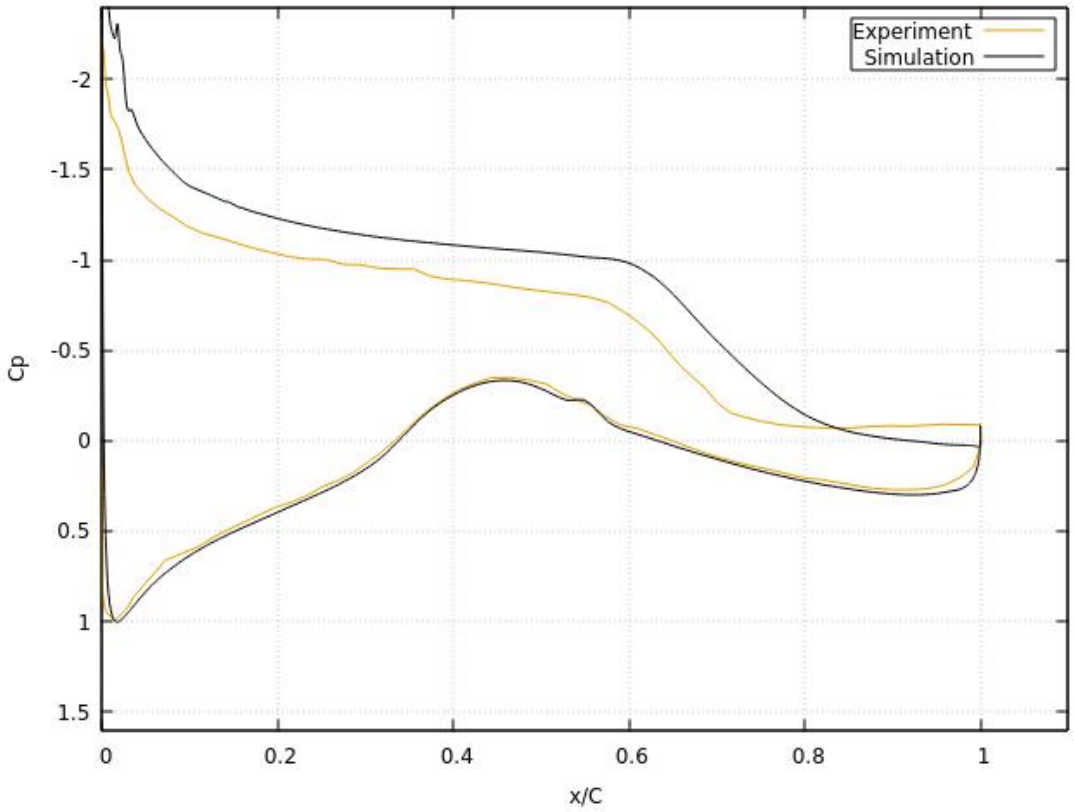

Figure 31. Surface Pressure Distribution, 8 degrees

It is clear from the above plots that the results of the simulation are in good agreement with the experiment for the lower side of the airfoil for all Angles of Attack. For the upper part of the airfoil there is a discrepancy between the results, which becomes larger as the Angle of Attack increases.

3.3 VP1304 Controllable Pitch Propeller

As a third, and final, test case a three-dimensional propeller was selected in order to assess how well the model will behave in this case and whether it will capture the three-dimensional effects correctly.

For this reason, simulations were performed on the VP1304 Controllable Pitch Propeller. The Potsdam Model Basin has published experimental results for this propeller under the name Potsdam Propeller Test Case (PPTC) in order to assist in model validation. The results of the simulation were compared with simulations made by Kalantzis [44] without transition modelling, as well as with experimental data, obtained by towing tank tests on the SVA Potsdam.

The basic geometric characteristics of the propeller are given on Table 2

Propeller diameter	D	mm	250.000
Pitch at $\frac{r}{R} = 0.7$	$P_{0.7}$	mm	408.705

Pitch at $\frac{r}{R}=0.75$	$P_{0.75}$	mm	407.3804
Mean Pitch	P_{mean}	mm	391.8812
Chord Length at $\frac{r}{R}=0.7$	$C_{0.7}$	mm	104.1670
Chord Length at $\frac{r}{R}=0.75$	$C_{0.75}$	mm	106.3476
Thickness at $\frac{r}{R}=0.75$	$t_{0.75}$	mm	3.7916
Pitch Ratio	$\frac{P_{0.7}}{D}$	-	1.6350
Mean Pitch Ratio	$\frac{P_{mean}}{D}$	-	1.5675
Area Ratio	$\frac{A_E}{A_0}$	-	0.7790
Skew	θ_{eff}	°	18.8
Hub Diameter Ratio	$\frac{d_h}{D}$	-	0.1500
Number of Blades	z	-	5
Direction of Rotation	-	-	Right Handed (SCS)

Table 2. Propeller geometric characteristics

For the simulations the one-blade approach was followed. The computational domain is cylindrical with the inlet located at a 3D distance away from the propeller plane, while the outlet is at a distance of 5D. Since the pattern of the flow is expected to have a rotational periodicity the computational domain was selected as a blade passage with 72° periodicity in the circumferential direction. The unstructured mesh utilized has 3.4×10^6 cells, is locally refined around the propeller and was generated by Kalantzis [44]. It should be noted that prior to the simulations a grid independence study was performed for the case of modelling without transition with satisfactory results [ibid].

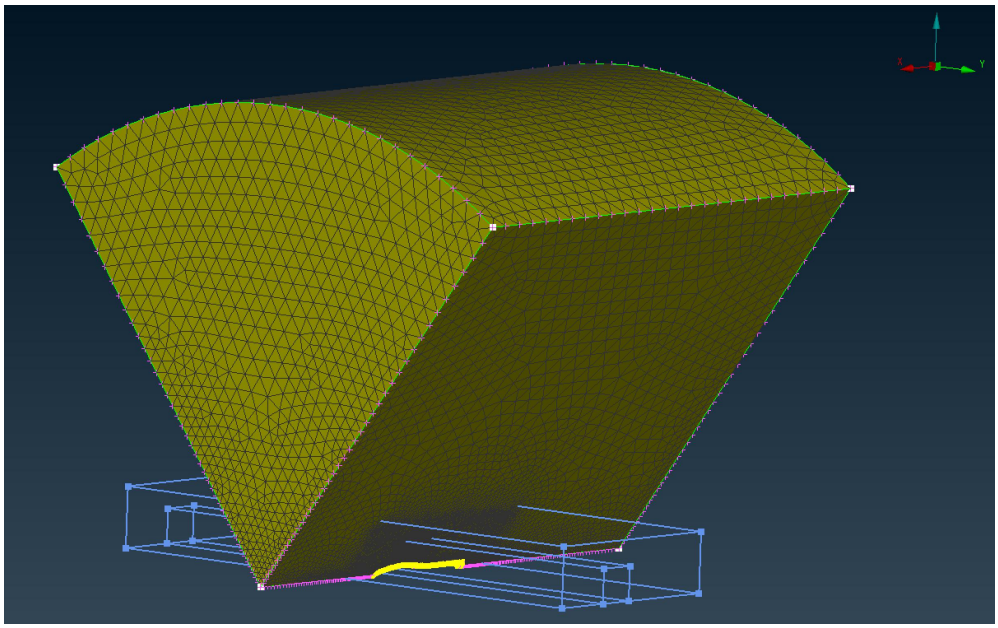

Figure 32. Mesh General view, refinement regions appearing as blue boxes. Source: [44]

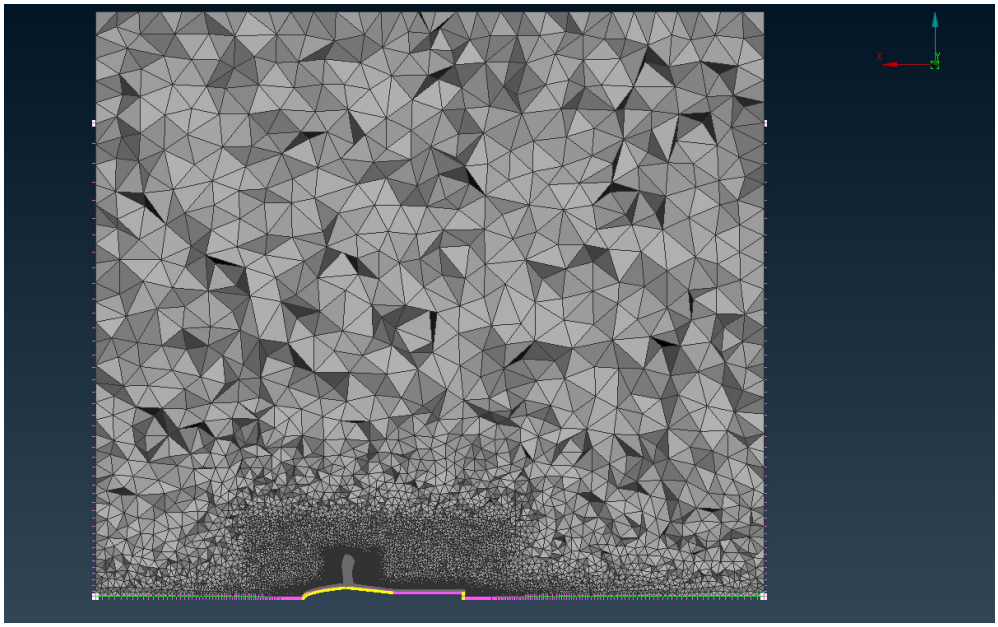

Figure 33. Sectional View of the Mesh. Source: [44]

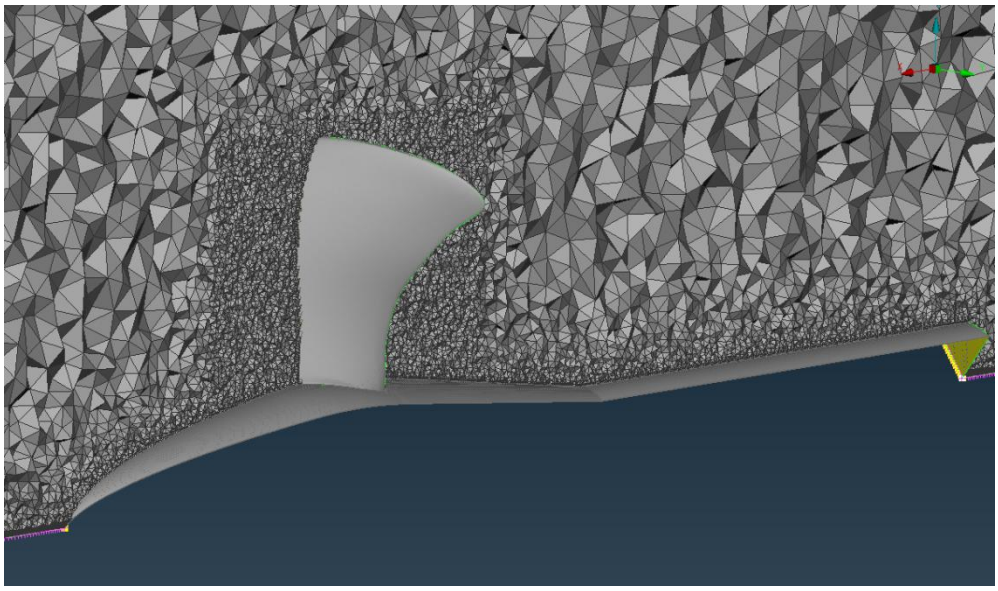

Figure 34. Mesh around the Propeller and the Hub. Source: [44]

For the simulation the revolutions were kept constant, while the inlet velocity was changed to achieve a certain Advance Coefficient each time. The flow conditions were selected to be the same with the experiment and are given on Table 3.

Water Density	ρ	$\frac{kg}{m^3}$	968.67
Kinematic Viscosity	v	$\frac{m^2}{s}$	1.07×10-6
Number of revolutions	n	$\frac{1}{s}$	15

Advance Velocity	V_A	$\frac{m^2}{s}$	0.75-5.25

Table 3. Flow Conditions Tested

Regarding the boundary conditions, a Dirichlet condition for the velocity is applied on the inlet. For the propeller and the hub the no-slip condition is applied. Lastly, the outlet boundary is considered a pressure outlet.

Simulations were carried out for values of the Advance Coefficient 0.6, 0.8, 1, 1.2 and 1.4 using the $\gamma - \mathrm{Re}_\theta$ and were compared to the corresponding results when the k-ω SST model is used, as well with the available experimental results.

The results for both cases are presented on Table 4, along with their deviation from the experiments and are further illustrated on figures 35 to 37 as bar charts. The open water diagram obtained from the above results is plotted on figure 38.

J	kt	10kq	η	ΔKT	Δ10KQ	Δη	ΔKT%	Δ10KQ%	Δη%	
0,6	0,626809	1,43814	0,416203224	0,002224406	0,041766862	-0,010927846	0,354877866	2,90422785	-2,625603492	Without Transition
	0,63313	1,440136	0,419817722	0,008545406	0,043762862	-0,007313348	1,349708022	3,038800669	-1,742029449	With Transition
0,8	0,5031709	1,198859	0,534389021	-0,000254198	0,020852046	-0,009734033	-0,050519138	1,739324341	-1,821525719	Without Transition
	0,5264102	1,24095	0,540107404	0,022985102	0,062943046	-0,00401565	4,366386214	5,072166195	-0,743491085	With Transition
1	0,379235	0,967312	0,623967498	-0,010864	-0,007567	-0,012892914	-2,864714491	-0,78227087	-2,066279716	Without Transition
	0,4089	1,0151	0,641103894	0,018801	0,040221	0,004243482	4,597945708	3,962269727	0,661902337	With Transition
1,2	0,2638446	0,7374526	0,683306381	-0,018634226	-0,038595874	-0,011875473	-7,062576077	-5,233675168	-1,737942637	Without Transition
	0,294736	0,787401	0,714888977	0,012257174	0,011352526	0,019707123	4,158696053	1,441771905	2,756669028	With Transition
1,4	0,1470344	0,497297	0,658796498	-0,024676018	-0,061525098	-0,025857645	-16,7824792	-12,37190202	-3,924982225	Without Transition
	0,1752519	0,537469	0,726536575	0,003541482	-0,021353098	0,041882432	2,020795438	-3,972898456	5,764669431	With Transition

Table 4. Results of the simulations

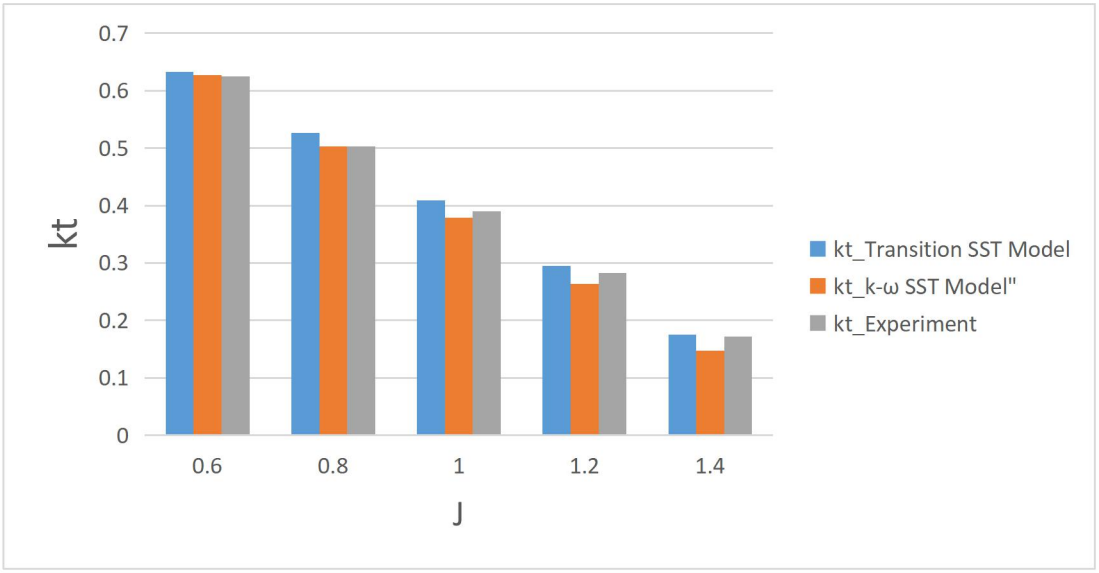

Figure 35. Thrust Coefficient against Advance Coefficient

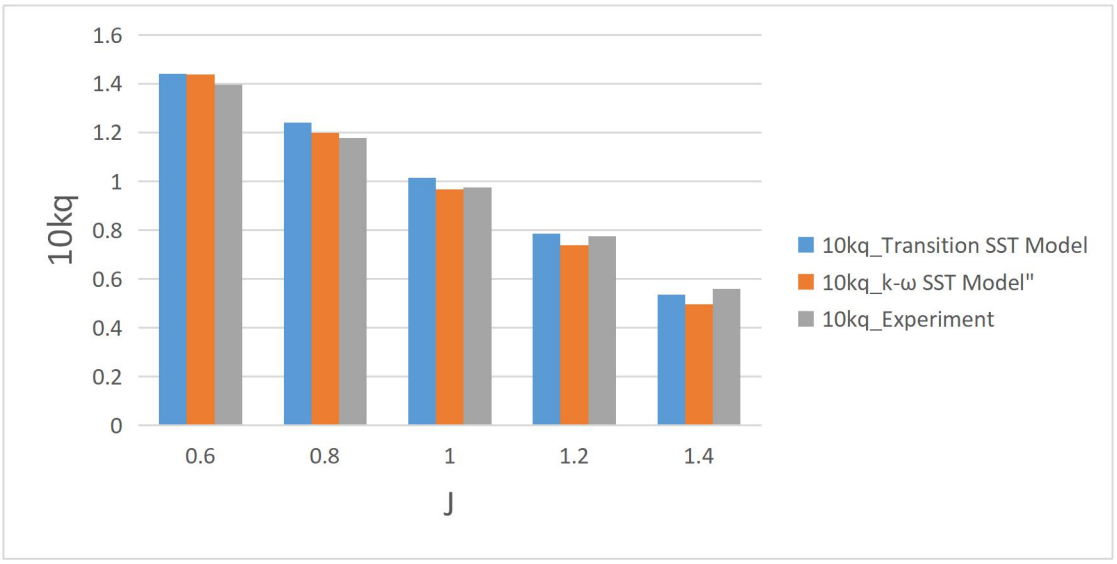

Figure 36. Torque Coefficient against Advance Coefficient

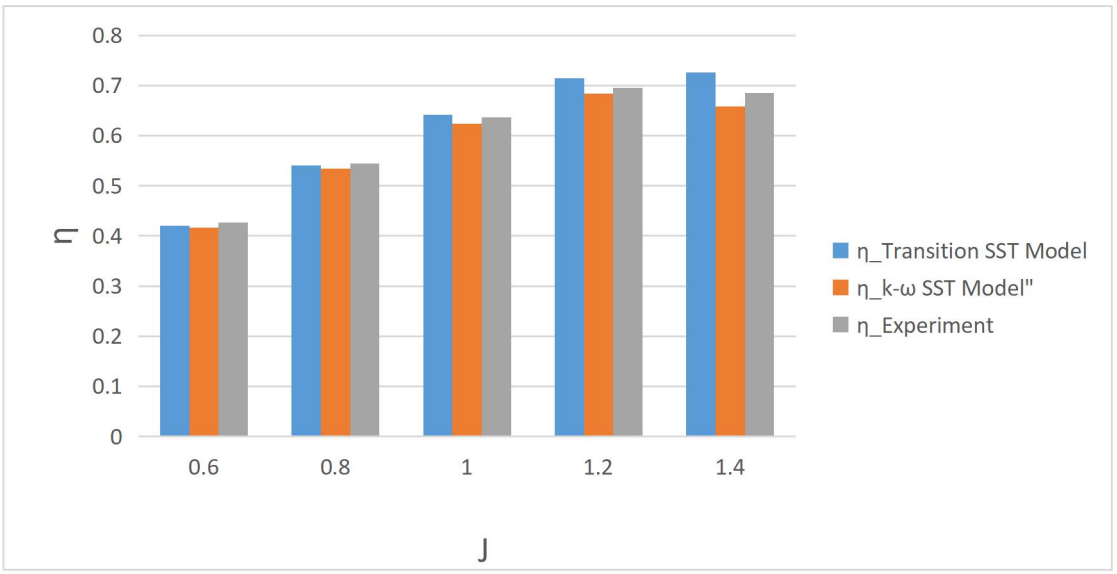

Figure 37. Propeller Efficiency against Advance Coefficient

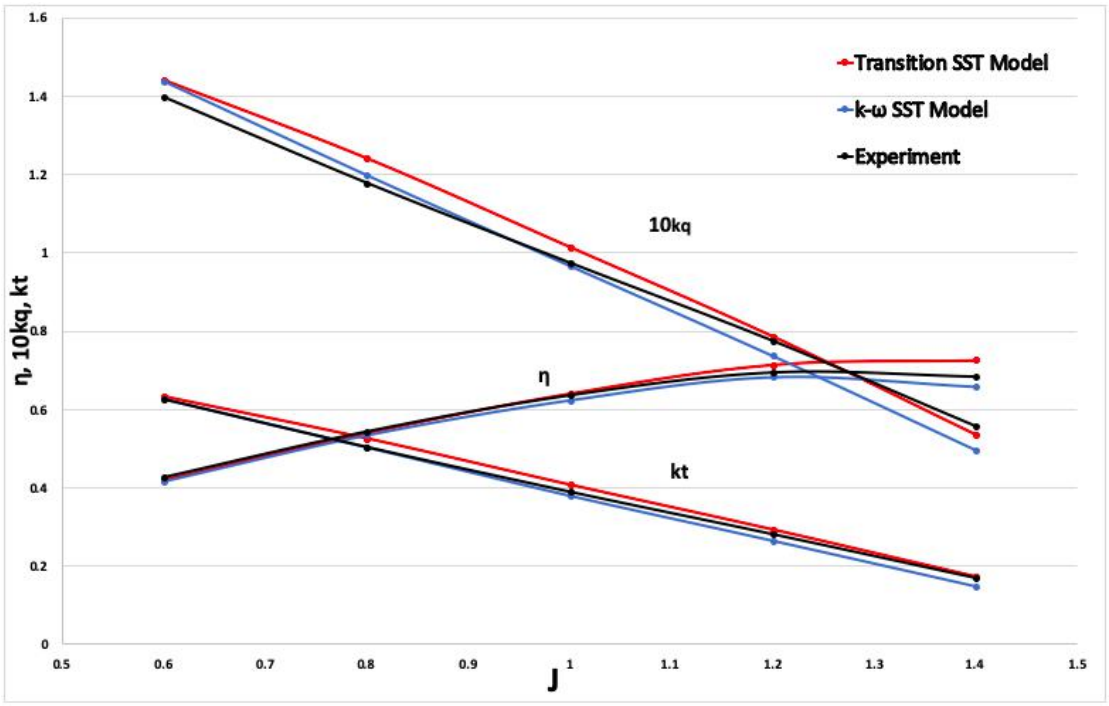

Figure 38. Open Water Diagram

First of all, the results have a maximum percentage difference of 4.59% for k_T , a 5.07 % for $10k_Q$ and a maximum of 5.76% for η for the $\gamma - \mathrm{Re}_\theta$ model. The transition model gives higher values of both propeller Thrust Coefficient (k_T) and Torque Coefficient (k_Q) compared to the k-ω-SST model at almost all propeller loadings. This can be attributed to an increment in the pressure component and a minute decrease of the friction component of the blade forces, when the transition model is used compared to fully turbulent flow assumptions, similar results were obtained in [45],[46]. In addition, for k_T and k_Q the results with $\gamma - \mathrm{Re}_\theta$ are closer to the experiment at j=1.2 and 1.4. This is because of the fact that, as the advance coefficient takes higher values the Angle of Attack decreases and there are more laminar regions at the propeller blade and the transition modelling has greater influence. Also, the resulting open water efficiency (η_0), which is calculated with $\gamma - \mathrm{Re}_\theta$, is closer to the experiment until j=1.

Regarding the slope of the curves, the model captured the slope correctly for all three curves,while the fully turbulent model failed.

4. Concluding Remarks

As far as the two-dimensional test cases are concerned, the MapFlow solver presented reliable results only for low Angles of Attack. On the contrary in higher Angles of Attack in all cases tested the solver severely overpredicts the Lift Coefficient. For the drag bucket corners no pattern can be traced, since for the DU-00-W212 airfoil the solver predicts the corners at lower values of CL than the experiment, while for the S827 for both Reynolds Numbers the Lift Coefficient at the Drag Bucket corners is overpredicted. It should be also mentioned that the solver gave results that were in very good agreement with the compressible one in all Angles of Attack, which was the basic purpose of this thesis, that is to examine whether the implementation of the Artificial Compressibility method will significantly alter the results of the solver.

For the three-dimensional case the simulation with transition modelling gave better results than the fully turbulent simulation only for two Advance Coefficients. This is contrary to what was expected, that is the impression that the simulation with transition modelling will give better results for all cases. This issue deserves further investigation. On the other hand the solver gave better results for the slope of the curves in comparison with the fully turbulent case.

Future Work Recommendations

The main recommendations that can be drawn from this work refer to the advisability of employing:

- Combination of Transition Modelling with higher order Spatial Discretization schemes like MUSCL or QUICK.
- Further assessment of the performance of the model on three-dimensional cases.
- Assessment of the influence that the choice of freestream Turbulence Intensity has on the results obtained.
- Application of other transition models like the e^N, the γ and the AFT in combination with an incompressible solver.

Appendix A: Order of accuracy based on the Taylor series truncation error.

Let f(x) a function of x. The Taylor series expansion of f(x) at the point (x+Δx) is given by:

$$f(x+\Delta x)=f(x)+\left(\frac{\partial f}{\partial x}\right)_x \Delta x+\left(\frac{\partial^2 f}{\partial x^2}\right)_x \frac{\Delta x^2}{2}+\left(\frac{\partial^3 f}{\partial x^3}\right)_x \frac{\Delta x^3}{6}+..... \qquad (2.118)$$

If we consider Δx to be the grid spacing, with some mathematical manipulation it can be shown that all the discretization schemes mentioned above can be written in the form of a Taylor series expansion, by neglecting all the factors containing Δx^n from a certain n and above.
The number n determines the order of accuracy of a certain scheme and expresses the rate of error decrease, as Δx is reduced. For example, if n=2 the scheme is second order accurate and it neglects terms containing Δx in the power of 2 and above.

Bibliography

[1] H. Veerstek, W. Malalasekra : An Introduction to Computational Fluid Dynamics The Finite Volume Method, 2nd Edition ,2006
[2] F. R. Menter, R. Langtry, S. Volker, P. G. Huang, Transition Modelling for General Porpuse CFD Codes, Engineering Turbulence Modelling and Experiments 6, (2005)
[3] Mayle R.E., Shulz A.: The Path to predicting bypass transition, ASME J. Turbomach. 119, pp. 405-411 (1997)
[4] Morkovin M.V. : On the many faces of transition, Wells C.S Viscous Drag Reduction, pp 1-3, Plenum, New York (1969)
[5] Mayle R.E. : Transition in a separation bubble, J. Turbomach. 118, pp. 752-759, (1996)
[6] F. White : Viscous Fluid Flow (1996)
[7] Chorin J. : A Numerical Method for Solving Incompressible Viscous Problems
[8] Diakakis K. : Computational analysis of transitional and massively separated flows with application to wind turbines, National Technical University of Athens, (2019)
[9] G. Papadakis S. Voutsinas : In view of accelerating cfd simulations through coupling with vortex particle approximations, The Science of Making Torque from Wind Journal of Physics, Conference Series (2014)
[10] D. Kwak, C.C. Kiris : Computation of Viscous Incompressible flows., Springer (2011)
[11] Richtmeier P.D, Morton K.W : Difference Methods for initial Value Problems, 2nd Edition, 1967
[12] Van Leer, B: Towards the Ultimate Conservative Difference Scheme V. A second order sequel to Godunov's method, J. Computational Physics, 127(1996), pp. 101-136
[13] G.D. van Albada, B. van Leer,W.W. Robert, A comparative Study of Computational Methods
[14] P.L. Roe, Characteristic based Schemes for the Euler equations , Annual Review of Fluid Mechanics 18 (1986), pp. 337-365
[15] Barth, T.J.; Jespersen, D.C.: The Design and Application of Upwind Schemes on Unstructured Meshes. AIAA Paper 89-0366, 1989.

[16] Venkatakrishnan, V.: On the Accuracy of Limiters and Convergence to Steady State Solutions. AIAA Paper 93-0880, 1993.
[17] Venkatakrishnan, V.: Convergence to Steady-State Solutions of the Euler Equations on Unstructured Grids with Limiters. J. Computational Physics, 118 (1995), pp. 120-130.
[18] B.P. Leonard, A stable and accurate convective modelling procedure based on quadratic upstream interpolation, Computer Methods in Applied Mechanics and Engineering 19 ,1979,pp. 59-98
[19] Godunov, S.K.: A Difference Scheme for Numerical Computation Discontinuous Solution of Hydrodynamic Equations. Math. Sbornik (in Russian),47 (1959), pp. 271-306; translated US Joint Publ. Res. Service, JPRS 7226, 1969.
[20] Anderson, John D., Jr.: Modern Compressible Flow: With Historical Perspective, 2d ed., McGraw- Hill, New York, 1990 pp 291-298
[21] Roe, P.L.: Approximate Riemann Solvers, Parameter Vectors, and Difference Schemes. J. Computational Physics, 43 (1981), pp. 357-372.

[22] R. Biedron, V. Vatsa, H. Atkins, Simulation of Unsteady Flows Using an Unstructured Navier-Stokes Solver on Moving and Stationary Grids, in: 23rd AIAA Applied Aerodynamics Conference, June, American Institute of Aeronautics and astronautics,Virginia,2005
[23] V.Vatsa, M. Carpenter, D. Lockard, Re-evaluation of an Optimized Second Order Backwards Difference (BDF2OPT) Scheme for Unsteady Flow Applications in: 48th AIAA Aerospace Sciences

Meeting Including the New Horizons Forum and Aerospace Exposition, American Institute of Aeronautics and Astronautics, Reston, Virginia, 2010, pp 1-15
[24] D. J. Mavriplis V. Venkatakrishnan, Agglomeration Multigrid for Viscous Turbulent Flows, ICASE report 24 (1994), pp. 553-570
[25] Courant, R.; F'riedrichs, K.O.; Lewy, H.: Uber die partiellen Differenzengleichungen der mathematischen Physik. Math. Ann., 100 (1928), pp. 32-74. Transl.: On the Partial Difference Equations of Mathematical Physics. IBM Journal, 11 (1967), pp. 215-234.
[26] Mazaheri, K.; Roe, P.L.: Numerical Wave Propagation and Steady-State Solutions: Soft Wall and Outer Boundary Conditions. AIAA Journal, 35 (1997) , pp. 965-975.
[27] Kreiss, H.O.: Initial Boundary Value Problems for Hyperbolic Systems. Comm. Pure Appl. Math., 23 (1970), pp. 277-298.
[28] Reynolds, 0.: On the Dynamical Theory of Incompressible Viscous Fluids and the Determination of the Criterion. Phil. Trans. Roy. SOC., London, Series A 186 (1895), pp. 123-164.
[29] Boussinesq, J.: Essai sur la the'orie des eaux courantes. Mem. Pres. Acad. Sci., XXIII, 46, Paris, 1877
[30] Boussinesq, J.: Theorie de l' 6coulement tourbillonant et tumubteur des liquides dans les lits rectiligues. Comptes Rendus de 1' Acad. des Sciences,CXXII (1896), p. 1293.
[31] Menter, F.R.: Two-Equation Eddy- Viscosity Turbulence Models for Engineering Applications. AIAA Paper 93-2906, 1993; also AIAA Journal, 32 (1994) pp. 1598-1605.
[32] Menter, F.R.; Rumsey, L.C.: Assessment of Two-Equation Turbulence Models for Transonic Flows. AIAA Paper 94-2343, 1994.
[33] Wilcox, D.C.: Reassessment of the Scale-Determining Equation for Advanced Turbulence Models. AIAA Journal, 26 (1988), pp. 1299-1310.
[34] Wilcox, D.C.: Turbulence Modeling for CFD. Published by DCW Industries, Inc., La Cafiada, California, USA, 1993.
[35] F. Menter, Zonal Two Equation k-w Turbulence Models for Aerodynamic Flows, in: 23rd Fluid Dynamics, Plasma Dynamics and Lasers Conference, American Institute of Aeronautics and Astronautics, Reston, Virginia, 1993.
[36] Apsley, D. D., Leschziner, M. A. Advanced turbulence modelling of separated flow in a diffuser Flow, Turbulence and Combustion 63, (2000), pp. 81-112
[37] F. Langtry, R.B., Menter, Transition Modelling for General CFD Applications in Aeronautics, AIAA 2005-552 (2005).
[38] R.B. Langtry, F.R Menter, Correlation-Based Transition Modelling for Unstructured Parallelized Computational Fluid Dynamics Codes, AIAA Journal 47 (2009), pp. 2894-2906
[39] F.R. Menter, R. B. Langtry, S. R. Likki, Y. B. Suzen, P. G. Huang, S. Volker, A Correlation-Based Transition Model Using Local Variables - Part I: Model Formulation, Journal of Turbomachinery 128 (2006).
[40] F. R. Menter, R. Langtry, S. Volker, P. G. Huang, Transition Modelling for General Purpose CFD Codes, Engineering Turbulence Modelling and Experiments 6, (2005)
[41] Y. Saad, Iterative Methods for Sparse Linear Systems Second Edition Yousef Saad (2003)
[42] O. Pires, X. Munduate, O. Ceyhan, M. Jacobs, H. Snel, Analysis of high Reynolds Number Effects on a wind turbine airfoil using 2D wind tunnel test data , Journal of Physics Conference Series 753 (2016)
[43] D. M. Somers, Design and Experimental Results for the S827 Airfoil, Technical Report January,NREL, 1988
[44] K.N. Kalantzis, Numerical Simulation of propeller open water characteristics using CFD, National Technical University of Athens,2020
[45] A. Bhattacharyya, Influence of Flow Transition on Open and Ducted Propellers,(2015)
[46] J.A.C. Falcao de Campos, J. Baltazar, D. Rijpkema, On the use of the γ-Re_θ Model for the Prediction of the Propeller Performance at Model-Scale