COMPUTATIONAL ANALYSIS OF PLANAR WINGS
DESIGNED FOR OPTIMUM SPAN-LOAD

BY
PRATEEK RANJAN

THESIS

Submitted in partial fulfillment of the requirements
for the degree of Master of Science in Aerospace Engineering
in the Graduate College of the
University of Illinois at Urbana-Champaign, 2016

Urbana, Illinois

Adviser:
Professor Phillip J. Ansell
Abstract

A computational analysis of three span-optimized wings was conducted using an open-source CFD tool. Simulations were carried out at $Re_T = 450,000$ in a semi-spherical domain consisting of unstructured tetrahedra close to the wing surface and pyramids in the farfield region. Simulations were carried out in both steady state and semi-transient states to predict flow transition. A comparative study of different turbulence models revealed $k - \omega SST$ and $k - k_L - \omega$ to be the most suitable turbulence models for this study. The model accuracy was determined using validations with experimental data from a previous study. The required accuracy was achieved using the most appropriate mesh resolution for all three wing designs and second order discretization schemes.

Computational results indicated different drag characteristics between the three span-load optimized wings at the design $C_L$. The Viscous Optimized Wing produced the minimum drag while the Elliptic Wing produced the largest drag at design $C_L$. The Inviscid Optimized Wing had the largest aspect ratio but still produced lesser drag when compared to the Elliptical Wing. Surface flow visualization indicated different flow transition characteristics for the three wings. These differences were attributed to the twist distributions specific to each wing. The Inviscid Optimized wing was observed to have largest laminar boundary-layer region at design angle of attack.

Qualitative wake analysis indicated different wake characteristics for each wing, attributed to the different span-loads. The Elliptic Wing had the most aggressive wake roll-up. Much lesser wake roll-up was observed for the Inviscid and Viscous Optimized Wings. The largest wake cross-section was observed for the Elliptic Wing, while the smallest wake cross-section was observed for the Inviscid Optimized Wing.
I would like to thank my advisor, Professor Phillip J. Ansell for giving me the opportunity to work and study in his research group. Professor Ansell supported me in all my endeavors and steered me in the right direction when necessary. Without his guidance and expertise, this study would have never been possible. I would also like to thank the professionals at Pointwise for their timely support and guidance in mesh generation. I am grateful to my family for giving me the courage to keep moving forward during toughest of times. Finally, I would like to thank Ms. Shubha Jain for her patience and curiosity which forced me to reason the observations in this study.
Table of Contents

List of Figures ................................................................. vi
List of Tables ................................................................. ix
1 Chapter 1. Introduction ....................................................... 1
  1.1 History ................................................................. 1
  1.2 Overview ............................................................. 3
  1.3 Vortex Drag: Fundamentals ........................................... 6
  1.4 Vortex Drag Reduction: A multidisciplinary approach .......... 7
    1.4.1 Wing Design Optimization Method ......................... 9
    1.4.2 Resulting wing design ...................................... 12
2 Chapter 2. Computational Methods ....................................... 18
  2.1 OpenFOAM ............................................................. 19
    2.1.1 0 directory .................................................. 20
    2.1.2 Constant directory .......................................... 22
    2.1.3 System directory ........................................... 23
  2.2 Computational Domain ............................................... 25
    2.2.1 Mesh quality ................................................ 25
    2.2.2 Grid Classification ......................................... 27
    2.2.3 Mesh generation ............................................ 30
  2.3 Turbulence Modeling .................................................. 40
    2.3.1 Direct Numerical Simulation ................................ 42
    2.3.2 Large Eddy Simulation .................................... 43
    2.3.3 Reynolds Averaged Navier-Stokes .......................... 44
    2.3.4 Overview of Turbulence Models ......................... 46
    2.3.5 kkl -omega model (k – kl – ω) ............................ 51
2.4 Numerical algorithms and Schemes ............................... 54
  2.4.1 SIMPLE algorithm ........................................ 54
  2.4.2 PISO algorithm ............................................ 55

3 Chapter 3. Results and Discussions ............................... 57
  3.1 Revisit: Wing Designs ........................................ 57
  3.2 Data validation ............................................... 60
  3.3 Wing Performance ............................................ 64
  3.4 Surface visualization ......................................... 70
  3.5 Wake Analysis ................................................ 73
  3.6 Solution Convergence ......................................... 84

4 Chapter 4. Conclusion ............................................. 87
  4.1 Summary ..................................................... 87
  4.2 Conclusions .................................................. 92
  4.3 Recommendations ............................................ 93

Bibliography .......................................................... 95
List of Figures

1.1 Elliptic Wing spanload .................................................. 14
1.2 Ratio of Induced drag of Inviscid Optimized Wing relative to elliptically
    loaded wing with same wing-root bending moment and lift ................. 14
1.3 Ratio of total drag of viscous optimized wing relative to elliptically-loaded
    wing ................................................................. 16
1.4 Optimized Spanloads ..................................................... 17
1.5 Twist distribution at Design $C_L$ ..................................... 17

2.1 Dictionary file for $U$ .................................................... 21
2.2 Dictionary file for $boundary$ .......................................... 23
2.3 Dictionary file for $controlDict$ ....................................... 24
2.4 Structured grid for a NACA 0015 wing section ....................... 28
2.5 Unstructured grid for a NACA 0015 wing section .................... 28
2.6 Hybrid grid for a NACA 0015 wing section ........................... 28
2.7 Mesh quality report for Elliptic Wing case ........................... 30
2.8 Structured mesh for NACA6412 wing section made using $snappyHexMesh$ 32
2.9 Mesh development stages in Pointwise for elliptical wing ............. 35
2.10 Slice view at midspan location of Elliptic Wing .................... 36
2.11 Leading and trailing edges for Elliptic Wing ........................ 36
2.12 Quality metrics for Elliptic Wing mesh ............................. 37
2.13 Quality metrics for Viscous Optimized wing mesh .................. 38
2.14 Quality metrics for Inviscid Optimized Wing mesh .................. 39
2.15 Visualization for $U_1(t)$ .............................................. 40
2.16 Visualization for $U_2(t)$ .............................................. 41
3.30 Velocity Contour plot at 20 % \( \bar{c} \) for the Viscous Optimized Wing . . . . . . . . 77
3.31 Velocity Contour plot at 40 % \( \bar{c} \) for the Elliptic Wing . . . . . . . . . . . . . . . . 78
3.32 Velocity Contour plot at 40 % \( \bar{c} \) for the Inviscid Optimized Wing . . . . . . . 78
3.33 Velocity Contour plot at 40 % \( \bar{c} \) for the Viscous Optimized Wing . . . . . . . . 78
3.34 Velocity Contour plot at 60 % \( \bar{c} \) for the Elliptic Wing . . . . . . . . . . . . . . . . 79
3.35 Velocity Contour plot at 60 % \( \bar{c} \) for the Inviscid Optimized Wing . . . . . . . 79
3.36 Velocity Contour plot at 60 % \( \bar{c} \) for the Viscous Optimized Wing . . . . . . . 79
3.37 Vorticity Contour plot at 5 % \( \bar{c} \) for the Elliptic Wing . . . . . . . . . . . . . . . . . 80
3.38 Vorticity Contour plot at 5 % \( \bar{c} \) for the Inviscid Optimized Wing . . . . . . . . . 80
3.39 Vorticity Contour plot at 5 % \( \bar{c} \) for the Viscous Optimized Wing . . . . . . . . 80
3.40 Vorticity Contour plot at 20 % \( \bar{c} \) for the Elliptic Wing . . . . . . . . . . . . . . . . 81
3.41 Vorticity Contour plot at 20 % \( \bar{c} \) for the Inviscid Optimized Wing . . . . . . . 81
3.42 Vorticity Contour plot at 20 % \( \bar{c} \) for the Viscous Optimized Wing . . . . . . . 81
3.43 Vorticity Contour plot at 40 % \( \bar{c} \) for the Elliptic Wing . . . . . . . . . . . . . . . . . 82
3.44 Vorticity Contour plot at 40 % \( \bar{c} \) for the Inviscid Optimized Wing . . . . . . . 82
3.45 Vorticity Contour plot at 40 % \( \bar{c} \) for the Viscous Optimized Wing . . . . . . . 82
3.46 Vorticity Contour plot at 60 % \( \bar{c} \) for the Elliptic Wing . . . . . . . . . . . . . . . . 83
3.47 Vorticity Contour plot at 60 % \( \bar{c} \) for the Inviscid Optimized Wing . . . . . . . 83
3.48 Vorticity Contour plot at 60 % \( \bar{c} \) for the Viscous Optimized Wing . . . . . . . 83
3.49 Final Residuals for Elliptic Wing Simulation variables . . . . . . . . . . . . . . . . . . . . . . . 86
3.50 Final Residuals for Inviscid Optimized Wing Simulation variables . . . . . . . . . 86
3.51 Final Residuals for Viscous Optimized Wing Simulation variables . . . . . . . . 86
## List of Tables

1.1 Aerodynamic and geometric parameters for wing designs .................................. 13

2.1 Primitive and derived patch field types ......................................................... 22

2.2 Maximum number of layers added using snappyHexMeshDict ......................... 32

3.1 Reference Angle of Attack at Design $C_L = 0.439$ ....................................... 58

3.2 Drag predictions by optimal spanload configurations at design $C_L$ .............. 65

4.1 Summary of Lift and Drag characteristics for Elliptic Wing .......................... 90

4.2 Summary of Lift and Drag characteristics for Inviscid Optimized Wing ........ 91

4.3 Summary of Lift and Drag characteristics for Viscous Optimized Wing .......... 92
Chapter 1

Introduction

1.1 History

Ever since mankind’s first powered and sustained flight in 1903, a vast number of studies have been conducted to describe and analyze the flow around streamlined bodies and its resulting effect of producing lift have been studied. The essence of creating a heavier than air (HTA) machine and achieving a sustained flight for a mere thirty seconds relied on the sound engineering practices and mathematical framework on which the craft was built. Such foundations in engineering which are now the cornerstones of aerodynamics, were well appreciated by a few during the time of inception of human powered flight. The evidence of a thorough or perhaps near accurate understanding of flight physics such as wing and airfoil behavior is evident from the following remarks (McCormick, 1995) made by Wilbur Wright at a meeting in Chicago in 1901.

1. That the ratio of drift to lift in well-shaped surfaces is less at angles of incidence of five degrees to 12 degrees than at angle of three degrees.

2. That in arched surfaces the center of pressure at 90 degrees is near the center of the surface, but moves slowly forward as the angle becomes less, till a critical angle varying with the shape and depth of the curve is reached, after which it moves rapidly toward the rear till the angle of no lift is found.

3. That a pair of superposed or tandem surfaces, has less lift in proportion to drift than either surface separately, even after making allowance for weight and head resistance of the connections.
Statements such those provided above indicated the presence of a thorough understanding of governing physics which lead to the design and development of the Wright Flyer (McFarland, 1953). A majority of the Wright Brother’s successes are attributed to the multiple wind tunnel experiments conducted involving different airfoil shapes and sizes as well as the previous works of Octave Chanute (Chanute, 1894) and Otto Lilienthal (McFarland, 1953). Earlier designs of the Wright Flyer were derived from the results of numerous gliding experiments conducted by Lilienthal (Lilienthal, 1889). A greater impact of their study and active flight control experiments served as an instrumental factor in the design of the flight control system for the Wright Flyer.

The Wright brothers were years ahead of their counterparts at that time due to ample amount of wind tunnel data collected by them in the late 1890s and early 1900s. Almost three years after the first controllable, sustained flight, similar technology was demonstrated by Charles and Gabriel Voisin (Harris, 1970). A newer design built by the Voisin brothers offered a sustained flight duration much greater than its previous counterparts, offering a range of roughly 2000 meters (McCormick, 1995). By the end of 1907, five such HTA technology demonstrators had been built successfully.

Further developments in theoretical and experimental principles of aerodynamics and fluid mechanics offer a rational way to explain the governing physics for design of early HTA crafts. All such principles have evolved from the first argument that the resistance of a moving body through a fluid depends on the fluid density, area of the body and square of the velocity, made by Sir Isaac Newton (Robertson, 1965). The concept of internal pressure, proposed and introduced by Daniel Bernouilli in 1738 allowed him to apply momentum principles to infinitesimal fluid elements. Using a firm mathematical base, Leonard Euler in 1755 further developed the works of Bernouilli for application to Hydrodynamics and proposed the internal pressure concepts across streamlines resulting in the pressure-velocity relationship that is commonly known today as the Bernoulli’s equation.

Due to the late inception of human controlled flights, many aerodynamic theories such as those by Fredrick Lanchester (Lanchester, 1907) were not published until 1907. However his study was made available 4 years late since the original Wright Flyer was built on the same principles but deduced from experimental results. Another study by Lanchester which focused on aircraft stability was not made available until 1908. Lanchester’s work
laid a foundation for the early versions of *Wing Theory* which was further developed after the introduction of a formalized boundary layer concept by Ludwig Prandtl (Prandtl, 1919). With the importance of airfoil study already emphasized from the experiments of the Wright Brothers, Nikolai Ergorovich Joukowski (spelling anglicized) in 1917 presented a comprehensive work on analytical methods for airfoil behaviors.

A prevailing design limiting factor, aerodynamic drag, has been a limiting factor for aircrafts since its inception. Various studies (Hefner and Bushnell, 1977; Bushnell, 1990; Viswanath, 2002; Garratt, 1977) have investigated sources of aerodynamic drag and methods of minimization. Efforts to minimize skin friction drag have progressed broadly towards flow field control near the walls. This refers to delay of laminar to turbulent flow transition and breakup of large eddies using riblets and other LEBU devices (Viswanath, 2002). Reduction of drag due to lift or induced drag is often approached by innovated wing designs which ultimately alter the vortex shedding properties in the downstream (Frediani, 2005). The importance of aircraft drag reduction has had an ever increasing effect on fuel consumption and flight performance. With the advent of computing methods, studies such as F. Palacious et al. (2015) have used integrated multidisciplinary tools (Palacious et al., 2013) for aircraft design.

### 1.2 Overview

From a pure physical standpoint, transportation of an object in fluid is highly inefficient due to the presence of drag, as some form of energy expenditure is required to move an object from one point to another. Increasing the efficiency of this process by reducing drag has been the focus of researchers since the early 1900s. For example, the airline industry budgets millions of dollars towards fuel consumption annually. Since the fuel costs are substantial, reduction of aerodynamic drag can result in huge savings in the airline industry. Hence a thorough investigation and application of drag reduction techniques is of great interest in both academic and industrial sectors. Studies such as Holmberg et al. (2012); Weideman (1996); Longmuir and Ahmed (2009); Collins (1982) emphasize the importance of drag reduction from an economic point of view.

At subsonic flight conditions, aerodynamic drag has two main components, firstly due to viscous effects which can also lead to drag in the form of pressure components (from
Kutta Condition), and secondly due to generation of lift, referred to as Induced Drag. Induced drag is a result of non uniform distribution of lift on a lifting surface such as wings, where lift varies from root to tip. Due to pressure differences between upper and lower surfaces, vortices are formed at the wing tips. The tip vortex causes an induced flow which increases the local angle of attack of the wing, resulting in an additional downstream facing component to the force acting on the wing. Since the drag is a result of "induced" action of the tip vortices, hence the name induced drag is given.

Aerodynamic drag on an aircraft can be expressed as:

\[ D = k_1 + k_2 \frac{L^2}{q}, \]  

(1.1)

where, \( k_1 \) and \( k_2 \) are functions of the aircraft geometry, \( L \) is the lift produced to sustain flight and \( q \) is the dynamic pressure:

\[ q = \frac{1}{2} \rho V^2 \]  

(1.2)

Based on the study by Kroo (2001), the Induced Drag contributes to almost half of the total drag in flight conditions for maximum \( L/D \) ratio. However, the author believes that this contribution is very design and condition specific. From Eq.1.1, at lower flight speeds, the induced drag increases. For a standard transport aircraft, typical \( C_L \) values can be estimated at 0.5 for cruise and 2.0 for takeoff. During takeoff, the induced drag contributes to about 70-80% of the total drag (Kroo, 2001). From an aircraft performance point of view, one might argue the take off and landing time to be significantly lesser as compared to the cruise flight time. However, a decrease in drag during take off and landing can improve aircraft flight characteristics such as low speed climb performance, engine-out climb and take-off noise, all of which are critical for aircraft design and certification. For aircrafts not limited by parameters such as climb gradient and turning rate, lower induced drag results in better handling qualities and an increase in flight time and endurance.

The effect of reduction of induced drag is seen not only in low speed aircrafts but also to a certain extend in high speed high performance aircrafts and hydrodynamic vessels such as in design of keels. Considering the objective of this study, discussions from here on shall pertain to aircrafts in low speed regime only.

Aerodynamic drag on an aircraft can be expressed as follows in dimensional form
Kroo, 2001):

\[ C_D = \frac{D}{qS} = C_{Dp} + \frac{C_l^2}{\pi A Re} + \Delta C_{Dc} \]  

(1.3)

where AR is the aspect ratio expressed as:

\[ AR = \frac{b^2}{S} \]  

(1.4)

where \( S \) is the surface area and \( b \) is the span. \( e \) is the Oswald’s efficiency factor discussed in Shevell (1983). In Eq.1.3, the first drag component \( C_{Dp} \) is the parasite drag which consists of viscous drag and associated pressure drag. The second term is the induced drag caused due to shed circulation from the wing and in most of the cases, due to generation of lift. This lift dependent drag varies quadratically with lift. The third term is an additional drag term valid for compressible flow regimes, especially at transonic speeds. Although Eq.1.3 offers a simple and clear accounting of various drag components, its accuracy is somewhat challenged. For example, due to viscous effects, especially at low speeds, the value of \( e \) in the induced drag term may change. The above equation further breaks down when propulsion units such as propellers are added. As a result of such additions, the induced drag is now modified for induced vorticity from propeller blades.

Decomposition of aerodynamic drag in Eq.1.3 serves as an important starting point for aircraft design, especially for lifting surfaces such as wings. With a simple term-by-term representation of different types of drag, each can be approached with an objective of minimization. Achieving minimum vortex drag or induced drag has been a challenging design constraint for many years. A general review of historical perspective and methods on vortex drag reduction is presented in Henderson and Holmes (1989). Another such review, focused on drag reduction concepts in nature is proposed by Bunshell and Moore (1991). A review of Wing Theory by Eppler (Eppler, 1987) highlighted the framework upon which the theory was based. Numerous studies were conducted as a part of Advisory Group for Aerospace Research and Development (AGARD) on drag prediction and reduction (Thomas, 1985; Hefner and Bushnell, 1977) and futuristic aircraft design and an overview of drag reduction concepts (Poisson-Quinton, 1985). Studies by Dam (1985) and R.M. Cummings and Shrinivas (1996) emphasized the importance of experimental

1.3 Vortex Drag: Fundamentals

Before the inception of human powered flight and any relevant theory which could have served as a foundation in understanding aerodynamic drag, it was believed that a component of drag was a quadratic function in lift at a given speed. Assuming the pressure forces to be normal to the lifting surface, Chanute (1894) and Lilienthal (1889) cited this result based on Eq. 1.5 derived from their experiments.

\[ C_D \approx C_{D0} + C_L \alpha \]  \hspace{1cm} (1.5)

The first comprehensive work on vortex drag was proposed by Lanchester (1907) and Prandtl (1919) with a description of a far-field view in terms of energy present in the downstream plane of vortex wake. Based on this formulation, lifting line theory was introduced by Prandtl (Prandtl, 1919). The initial model for lifting line theory was further developed by his students Betz, Munk and Trefftz (Anderson, 1997). The lifting line theory model provided a simple and powerful way of estimating the lift distribution on a wing surface with a defined chord and twist distribution. Models developed after its first introduction allowed for computing vortex drag or as Munk suggested, induced drag. Based on the study by Munk (1923), the minimum induced drag for a lifting surface could be achieved with a constant wake-induced down wash distribution in the farfield. Such a wing would have an elliptical lift distribution and would be attributed with an induced drag expressed as:

\[ C_{Di} = \frac{C_L^2}{\pi AR} \]  \hspace{1cm} (1.6)

Eq. 1.6 defines the second term in Eq. 1.3. This expression indicates the dependence of \( C_{Di} \) on \( C_L \). With available values for \( C_L \) and \( C_D \) from computational or experimental studies, remaining parameters in Eq. 1.3 can be estimated. Estimates for Oswald’s efficient
factor and $C_{Dp}$ can be made from the slopes of $C_D$ vs. $C_L^2$ plots. This method of estimation has been used successfully in numerous wind tunnels and flight test experiments. Such studies have indicated boat sails to have an $e$ between 0.70 and 0.80 (Brown, 1972) and between 0.65 and 0.75 for a small general aviation aircraft such as the Cessna 172. The evident difference between the obtained values of $e$ and the theoretical maximum of 1 indicates the lift-dependent drag to be comprised of more than just the vortex drag. This departure of $e$ from unity is observed as $C_L$ increases. At higher $C_L$, the viscous-related pressure drag increases along with skin friction drag, which in turn affects the lift-dependent drag. This effect is pronounced in high aspect ratio wings. Another difficulty in estimating $e$ is due to the fact that vortex drag can exist even when the net $C_L$ is zero. For example, in a twisted wing, although the lift is zero, portions of the wing with incidence (negative or positive) can interfere with neighboring regions on the wing.

By incorporating wing twist distribution, a more general expression for induced drag can be expressed as (Kroo, 2001):

$$C_{Di} = \frac{C_L^2}{\pi AR u} + \theta C_L \nu + \theta^2 w = \frac{C_L^2}{\pi AR e_{inviscid}}$$  \hspace{1cm} (1.7)

With a pre-defined twist distribution, appropriate amplitude $\theta$ can be chosen to reduce the vortex drag at a given design $C_L$. A more comprehensive and physical description of vortex drag can be found in McCormick (1995); Anderson (1997) and Kroo (2001).

### 1.4 Vortex Drag Reduction: A multidisciplinary approach

Efforts to reduce induced drag was first made as early as 1897 by Lanchester (Whitecomb, 1994) wherein he proposed wing end plates as a patent design. Numerous theoretical studies detailing methods of vortex drag reduction have since been presented. Early theoretical studies provided a mathematical framework (Reid, 1924; Hempke, 1927; Mangler, 1938) for wings with reduced vortex drag. With the advent of computing technology, the 1960s and thereafter saw introduction of new methods such as the Vortex Lattice Method (VLM) for analysis of reduced induced drag configurations. To reiterate from previous sections, the most direct way of approaching induced drag minimization was the elimination of near field vortices such as those at the wing tips and trailing vorticity downstream of the wing. Lanchester’s vortex trunks (Lanchester, 1907) is one way to look at the
vortex system of a wing and is considered to be an effective approach to this date. Results from Prandtl’s work on minimum induced drag for planar wings serves as a baseline for all drag reduction studies. Since shedding circulation is required for producing lift, minimization of vortex drag in turn reduces $C_L$ itself. More subtle alterations to the wing design are required such as a multidisciplinary design approach.

The wing root bending moment has been used as a surrogate variable to account for structural weight. The wing weight scales proportionally with this variable. In early multidisciplinary studies, a reduction in induced drag was achieved by increasing the span (relieving the fixed span constraint proposed by Prandtl and Munk) and transitioning from an elliptic to a non-elliptic spanload. The first analytical solution for a wing with fixed lift, reduced induced drag and integrated bending moment across the span was proposed by Prandtl (Prandtl, 1919). Alternative models were developed using the wing root bending moment instead of span integrated bending moment, as proposed and derived by Jones (1950) and Klein and Viswanathan (1973). Klein and Viswanathan (1975) later introduced a spanload which minimized induced drag keeping constant lift fixed and structural weight which scaled with span integrated bending moment and shear. Lobert (1981) proposed a spanload solution for minimizing induced drag by constraining the bending moment per unit sectional thickness.

In addition to the above stated studies which focused on induced drag minimization using fixed lift and bending moment constraints for structural considerations, several studies have discussed have also incorporated different multidisciplinary design constraints such as wing tip devices (Deyoung., 1979; McGeer, 1984; Iglesias and Mason, 2001; Kroo, 1984; Wakayama and Kroo, 1995; Pate and German, 2013; Verstraeten and Slingerland, 2009; Takahashi, 2012), non-planar wings and multiple surfaces. Though most of these studies have focused on minimization of induced drag, some studies have included profile drag coefficients into a spanload optimization routine as a quadratic fit. Examples of optimization studies involving viscous factors contributing towards drag include Wakayama and Kroo (1995); Ning and Kroo (2010) and Verstraeten and Slingerland (2009).

With the development of improved computing technology over the past few decades, MDO has been applied for overall drag reductions for full aircraft configurations. Modern MDO tools such as those discussed by Palacios et al. (2013), couple two different solution matrices, one for aircraft structural solution using FEA methods and another for
flow solution using FVM methods, These two matrices are coupled before running an optimization routine. The resulting configurations are usually known to meet all structural and aerodynamic design requirements. Given the large design spaces, such as in the example of MDO of a full aircraft configuration (Palacious et al., 2012), a trade-off between solution quality and computational expenses need to be made. This hence results in the wide usage of low fidelity design methods.

Though there exist a large number of theoretical models for induced drag minimization, the effect of a certain spanload on the total aerodynamic characteristics of a wing cannot be fully understood completely using only inviscid considerations. More so, the applicability of such inviscid models when applied to a full aircraft design becomes questionable due to lack of studies discussing its physical fidelity. One such experimental study by Wroblewski and Ansell (2016) discusses the implementation and analysis of three spanload optimized wings. Extensive aerodynamic data such as wing lift, drag and spanloads are presented in order to understand the manifestation of viscous effects in three different spanload wing designs. However these wing designs were created using an inviscid approach and viscous-corrected optimization. In order to compare the wing drag across multiple levels of fidelity.

The present study supplements the study by Wroblewski and Ansell (2016) by presenting detailed flow field data including transition physics. Since the objective of the study is to understand the applicability of computational techniques to understand the viscous effects in low to medium angles of attack, the scope of analysis is limited to pre-stall regime only. Also, first and second order numerics are used as a trade off between needed prediction accuracy and computational costs.

1.4.1 Wing Design Optimization Method

The spanloads for the three wing were calculated using a discrete vortex element method with an embedded Trefftz plane analysis routine. This method is essentially derived from the classic baseline approach proposed by Blackwell (1976). Similar design routines have been discussed and used for optimization studies by Kroo (1984) and Iglesias and Mason (2001). Recent studies such as those by Ning and Kroo (2010) and Verstraeten and Slingerland (2009) also present application of this routine.

This section presents the optimization routine briefly with the key steps, similar to
what is presented in Wroblewski and Ansell (2016). A comprehensive description can be found in previously stated literature.

The induced flow velocities can be estimated from a crossflow plane located infinitely downstream of the wing. The flow velocities are computed normal to the crossflow plane, a method commonly referred to as the Trefftz plane analysis method. A constant wake is assumed extending from the wing trailing edge until the crossflow plane. This approach allows for the induced velocities computed at a downstream region to be projected back to the wing lifting line, hence allowing for forces and moments to be computed. The local bound circulation can be computed using the Kutta-Joukowski theorem using the expression:

\[
\frac{\Gamma_j}{V_\infty} = \frac{C_l c}{2}
\]  

(1.8)

Upon normalization, the spanload becomes:

\[
\gamma_j = \frac{(C_l c)_j}{c_{avg}}
\]  

(1.9)

where \(c_{avg}\) is the mean geometric chord.

From the Biort-Savart law, the induced velocity on the Trefftz plane is twice that in the near field. This difference can be attributed to the presence of infinite vortex lines in the farfield as opposed to series of semi infinite vortex lines present in the near field. The induced velocities on the Trefftz plane can be expressed in terms of spanloads, since at each control point on the crossflow plane the induced velocities are dependent on spanloads. Due to this dependence, the induced drag expression from the Kutta Joukowski theorem can be expressed in terms of spanloads. Similarly, in the expression of lift, the circulation term can be rewritten.

\[
C_{D_i} = \sum_{i=1}^{N} \sum_{j=1}^{N} \gamma_i \gamma_j s_i A_{ij}
\]  

(1.10)

\[
C_L = 2 \sum_{j=1}^{N} \gamma_j s_i,
\]  

(1.11)

where, \(A_{ij}\) is the aerodynamic influence constant and \(s_i\) is the normalized semi width
of a given vortex pair.

The method of Lagrange multipliers, discussed in Wroblewski and Ansell (2016), is used to compute a spanload that minimizes the induced drag under a design constraint. A cost function $J$ is introduced to minimize the induced drag at a fixed $C_L$. This may be expressed as:

$$ J = \left( C_D + \lambda C_L \left( C_L - C_{L_{ref}} \right) \right) \quad (1.12) $$

The cost function $J$ is minimized when its partial derivatives with respect to each independent variable equals zero. Substituting Eq.1.10 and Eq.1.11 into Eq.1.12, a linear systems of equations are obtained which can be solved for the optimum spanload and Lagrange multiplier for the lift constraint ($\lambda_{CL}$).

$$ \begin{bmatrix} \frac{\partial J}{\partial \gamma_j} \\ \frac{\partial J}{\partial \lambda_{CL}} \\ \frac{\partial J}{\partial \lambda_{CB}} \end{bmatrix} = [0] \quad (1.13) $$

To account for structural design parameters, surrogate variables such as the wing root bending moment (WRBM) can be incorporated into Eq.1.12, expressed as follows:

$$ J = C_D + \lambda_{CL} \left( C_L - C_{L_{ref}} \right) + \lambda_{CB} \left( C_B - C_{B_{ref}} \right) \quad (1.14) $$

where,

$$ C_B = \frac{1}{2} \sum_{j=1}^{N} \gamma_j s_j \left( \frac{2y_j}{b} \right) \quad (1.15) $$

The linear system of equations now becomes

$$ \begin{bmatrix} \frac{\partial J}{\partial \gamma_j} \\ \frac{\partial J}{\partial \lambda_{CL}} \\ \frac{\partial J}{\partial \lambda_{CB}} \end{bmatrix} = [0] \quad (1.16) $$

The sum of the induced and profile drag terms were also combined to determine the influence of profile drag on the optimum spanload, similar to work of Kroo (1984) and Verstraeten and Slingerland (2009). The profile drag coefficient, $C_{d,p}$ is assumed to vary quadratically with the sectional lift coefficient, based on the expression:
\[
C_{d,p} = C_{d,0} + C_{d,2}C_l^2
\]  
(1.17)

A Reynolds number dependence was also incorporated into the optimization routine. 
\(C_{d,0}\) and \(C_{d,2}\) were computed using a least squares fit. XFOIL (Drela and Youngren, 2001), an airfoil analysis tool which couples integral boundary method and inviscid flow solutions was used to determine profile drag coefficients for the optimization routine. Taking into consideration the aeroelastic effects and structural limitations, especially for high AR wings such as those under consideration, a 15% thick airfoil section was used. A standard symmetric NACA 0015 airfoil geometry was chosen for the wing design. By using a well-established and understood airfoil section such as the NACA 0015, the wings could be designed for the sole purpose of understanding the effects of viscous effects and various design constraints on wing spanloads obtained in actual conditions.

The total drag of the wing was obtained by adding the profile and induced drag components, resulting in the expression:

\[
C_D = \sum_{i=1}^{N} \gamma_i \gamma_j s_i A_{ij} + \sum_{i=1}^{N} \left[ C_{d,0,i} + C_{d,2,i} \left( \frac{\gamma c_{avg}}{c} \right)_i^2 \right]
\]  
(1.18)

The profile drag in Eq.1.18 was obtained using a strip theory approach. The final expression for the cost function incorporating all design constraints (fixed lift and wing root bending moment), to minimize the total drag becomes:

\[
J = C_D + \lambda_{C_L}(C_L - C_{L_{ref}}) + \lambda_{C_B}(C_B - C_{B_{ref}})
\]  
(1.19)

1.4.2 Resulting wing design

The optimization routine discussed in the previous section was used to design and develop three wing designs with optimized spanloads. Aerodynamic and geometric parameters for the resulting wing designs are listed in Table 1.1.
Design Parameters

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>$C_L$</td>
<td>0.439</td>
</tr>
<tr>
<td>$\frac{b}{2}C_B$</td>
<td>0.3</td>
</tr>
<tr>
<td>$Re\tau$</td>
<td>450,000</td>
</tr>
<tr>
<td>Airfoil</td>
<td>NACA 0015</td>
</tr>
<tr>
<td>Planform Taper Ratio ($\lambda$)</td>
<td>0.5</td>
</tr>
<tr>
<td>Wing Reference Area (S)</td>
<td>0.128 $m^2$</td>
</tr>
</tbody>
</table>

Table 1.1: Aerodynamic and geometric parameters for wing designs

From table 1.1, each wing features a different span and aspect ratio depending upon the design constraints imposed. However, the same wing area, planform shape, wing lift coefficient and Reynolds number was kept for all three designs. Since all three wings have the same area with different spans, the mean aerodynamic chord varied for each wing. Computations were hence carried out by adjusting the kinematic viscosity of the flow, in order to keep the Reynolds number constant for all three cases.

Elliptic Wing

The first spanload design was optimized using cost function $J$ from Eq.1.12. The wing design was approached with the objective of minimizing induced drag for a given $C_L$ and span. This constraint resulted in an elliptic spanload, hence referred to as the Elliptic Wing. Fig.1.1 represents the obtained spanload along with the elliptical spanload proposed by Prandtl (1919), as a function of the normalized semispan ($\eta$). The elliptical wing is considered as the baseline wing design in the current study.
Inviscid Optimized Wing

The second spanload design was optimized using cost function $J$ from Eq.1.14. The wing design was approached with an object to minimize the induced drag for a fixed lift and wing root bending moment. Fig.1.2 represents the ratio of induced drag for the Inviscid Optimized Wing to that of an elliptical wing ($C_{D,i}/C_{D,ie}$) with the same lift and wing root bending moment for a range of span ratios ($b/b_e$).

![Figure 1.2: Ratio of Induced drag of Inviscid Optimized Wing relative to elliptically loaded wing with same wing-root bending moment and lift](image)

From Fig.1.2, the induced drag ratio is observed to decrease with increasing span ratio but never reaches a minimum. This result is obtained due to the inviscid assumption made in the discrete vortex element approach, where the span may extend up to infinity.
without an increase in profile drag. With the limit of the infinite span, the induced drag of the wing would approach zero. However negative loading on the wing tips would be present in addition to the span bending moment exceeding the value at the root (Jones, 1950). The proposed design in Klein and Viswanathan (1973) is based on the span ratio which corresponds to the inflection point in Fig.1.2, i.e $b/b_c = 4/3$. As a result, the current wing design also uses this span ratio.

**Viscous Optimized Wing**

The third spanload design was optimized using the cost function $J$ from Eq.1.19. The optimum spanload was designed in a way that it minimizes the total drag of the wing as opposed to induced drag in the previous two designs. A strip theory approach was used to first calculate the total drag, followed by calculating its ratio to the Elliptic Wing drag ($C_D/C_{D,e}$). Fig.1.3 represents the total drag ratio as a function of the varying span ratio. For comparison, drag ratios from previous studies such as those by Kroo (1984) and Verstraeten and Slingerland (2009) are also presented. Slight differences in drag ratios are expected due to the difference in design constraints used in these studies. For example, Verstraeten and Slingerland (2009) calculated the total drag of a tapered wing with $\lambda = 0.5$ with wing tip extensions. Similarly Kroo (1984) determined the total drag ratio for a wing with $\lambda = 0.3$ with design constraints different than those under current consideration. Another factor which may contribute to the evident differences is the profile drag coefficient used, resulting from differences in airfoil. Taking into to account these sources of differences, the behavior of drag ratios with increases in span ratio is consistent with the literature.
From Fig.1.3, a minimum predicted drag is estimated for a span ratio of 1.21. For a span ratio of unity, the drag ratio is nearly unity which indicates that, while profile drag contribution to the total drag is considered, the optimal design corresponds to the twist distribution for the elliptic spanload.

Fig.1.4 illustrates the spanloads for all three wing designs. As expected, the Elliptic Wing consists of an elliptic spanload since it is optimized for a fixed $C_L$ and span. The spanload distribution for the Inviscid Optimized case features relatively higher loading at the wing root and lower loading at the wing tips when compared to the elliptical loading. Spanload design for the Viscous Optimized Wing is observed to lie in region between the Elliptic and Inviscid Optimized Wing spanloads. This trend occurs due to the decrease in induced drag with increase in span being offset by increase in profile drag. Fig.1.5 represents the twist distributions used to achieve optimal spanloads at design $C_L$. 

---

Figure 1.3: Ratio of total drag of viscous optimized wing relative to elliptically-loaded wing
Figure 1.4: Optimized Spanloads

Figure 1.5: Twist distribution at Design $C_L$
Chapter 2

Computational Methods

Introduction

This study employs computational methods to understand and investigate the near and farfield flow physics about three dimensional wing geometries at low Reynolds number flow regime. Given the predominant viscous damping effects and the evolution of the flow from laminar to turbulent associated with this flow regime, the selection of a robust and effective numerical solver package was crucial. While there exists a large number of commercial and OpenSource CFD codes, the current study progressed in two directions, one towards understanding the numerics of computational techniques and another towards using the code as an investigative tool for analyzing the design’s performance. Commercial flow solvers such as ANSYS FLUENT, StarCCM, AccuSolve to name a few, provide a robust and powerful set of algorithms for solving a diverse range of flow physics. However, such software packages provide a little or no access to the governing source code. This inhibits the understanding of different numerical techniques that make up the mathematical framework for a given solver. Designed for a target audience in the Industrial sector, the usability is simplified at the cost of restricted source code modification. This prevalent feature of commercial CFD codes make them an unlikely choice for many academic studies. A CFD code may be classified as ”open source” if the source code is available for modifications to the user. Another feature of such a code is the collaborative development of the source code by users present all over the CFD community. Such collaborations are reviewed, benchmarked and released as newer versions of the CFD code (Frust, 2012).
However, such codes may be subjected to limitations. For example, an open source generally lacks a user friendly GUI and support. Another limitation is the standardization of different flow solvers with the CFD code itself. Due to the collaborative nature of the source code, certain algorithms are often found to deliver inaccurate results (Gomez et al., 2014).

Taking into considerations the aforementioned advantages and disadvantages of open source CFD codes, this study carefully employs a robust CFD code for computations of the flow about the three wings desired. In the sections to follow, a comprehensive overview of the flow solver, turbulence models and employed numerics are presented. A complete description of the CFD code and governing equations is beyond the scope of this thesis. However adequate references will be provided to make the reading as comprehensive as possible.

2.1 OpenFOAM

Open Field Operation and Manipulation, commonly known as OpenFOAM, is a finite Volume (Kim et al., 2001) unstructured flow solver. Written in C++ programming language as an Open Source object oriented library, the code addresses a diverse range of physics in Computational Continuum Mechanics (hereafter CCM). Initially written and applied to fundamental low speed fluid mechanics test cases in the late 1980s at Imperial College, London, the code has since evolved and mirrors the maturity of many commercial packages available today (Balogh et al., 2012; Kassem et al., 2011; Lysenko et al., 2013; Vatani and Mohammad, 2013; Kanoria and Chandar, 2015; Drikakis et al., 2016). Being one of the first flow solvers of its kind, the collaboration to the source code (hereafter SRC) has been an instrumental factor in its rapid evolution. The current version of OpenFOAM addresses the two previously-stated limitations of Open Source CFD solvers. While the native distribution still operates with command-based tools, certain third party GUI distributions such as Visual-CFD and HELYX-OS reduce the initially steep learning curve.

OpenFOAM’s architecture is based on a object oriented framework. The basic ideology behind the design of OpenFOAM can be explained using the representation of the kinetic energy equation in a Reynolds Averaged Navier-Stokes (RANS) model (Wilcox, 1994):
\[ \frac{\partial k}{\partial t} + \nabla \cdot (uk) - \nabla \cdot [(\nu + \nu_t)\nabla k] = \nu_t \left[ \frac{1}{2} (\nabla u + \nabla u^T) \right]^2 - \frac{\epsilon_0}{k_0} k \]  

(2.1)

The above equation is represented as follows in the OpenFOAM SRC:

```c
solve
{
    fvm::ddt(k) + fvm::div(phi, k)
    == nut*magSqr(symm(fvc::grad(U)))
    fvm::SP(epsilon/k, k)
};
```

The correspondence between Eq.2.1 and above script is clear. With a clear and thorough knowledge of basic fluid governing equations along with a preliminary programming knowledge in C++ programming, the use of a support manual seems trivial. A comprehensive description of OpenFOAM SRC is present in the user guide (Greenshields, 2015).

From an application point of view, the OpenFOAM architecture consists of a predefined solver specific-case file which consists of the following three sub directories, discussed in detail in the sections to follow.

### 2.1.1 0 directory

The 0 sub directory, as the name suggests, consists of initial values and boundary conditions for all scalar and vector fields required for solving the flow field. For example, for a simple 2 dimensional laminar incompressible flow problem (Bayraktar et al., 2012), two flow field variables, U (velocity), p (pressure) and \( \mu \) (viscosity) are needed to effectively resolve the flow field. More flow parameters such as turbulent kinetic energy, \( k \), may be solved for to quantify turbulence. Each vector or scalar field is specified using a dictionary file in the 0 sub directory. A dictionary format, in simpler terms, can be defined as a
script file with appropriate header used with OpenFOAM. Fig.2.1 illustrates a dictionary file for $U$. For the sake of brevity, only a partial view of the dictionary file is shown below.

![Dictionary file for $U$](image)

Figure 2.1: Dictionary file for $U$

The dictionary file serves as an important tool for assigning initial values and boundary conditions for the corresponding flow variables. For example, in Fig.2.1, the internal field refers to the initial velocity field for $U$. For a steady state simulation, the Reynolds number is based on the mean aerodynamic chord of the wing. For simplicity, the kinematic viscosity $\nu$ is adjusted, in order to keep the velocity as $1\text{ms}^{-1}$ at the desired Re. The "farfield" refers to a patch name assigned to a set of faces on the computational domain. A freestream boundary condition is applied to the patch farfield with a flow velocity of magnitude $1\text{ms}^{-1}$. For a change in velocity vector, such as in the case of change of angle of attack, the flow magnitude is changed to corresponding sine and cosine components. In addition to farfield, the computational domain consists of patches $wing$ and $sym$. Each of these patch names will be discussed in the upcoming sections. Similar assignments are made for each of the three patches in the domain for $U$, $p$, $nut$, $k$, epsilon and $nuTilda$. 

21
The turbulent parameters mentioned here will be discussed in detail in later sections. Table 2.1 consists of a brief description of different boundary conditions assigned to each of the three computational patches in this study.

<table>
<thead>
<tr>
<th>Type</th>
<th>Description for patch field</th>
<th>Data to specify</th>
</tr>
</thead>
<tbody>
<tr>
<td>fixedValue</td>
<td>Value of $\phi$ is specified</td>
<td>value</td>
</tr>
<tr>
<td>zeroGradient</td>
<td>Normal gradient of $\phi$ is zero</td>
<td>–</td>
</tr>
<tr>
<td>symmetry</td>
<td>Slip Wall</td>
<td>–</td>
</tr>
<tr>
<td>calculated</td>
<td>$\phi$ is derived from other fields</td>
<td>refValue</td>
</tr>
<tr>
<td>freestream</td>
<td>Switches between fixedValue and zeroGradient depending upon the direction of $\phi$</td>
<td>freeStreamValue</td>
</tr>
</tbody>
</table>

Table 2.1: Primitive and derived patch field types

2.1.2 Constant directory

The constant sub directory, as the name suggests, consists of elements of the case which do not change over time. This includes the polyMesh sub directory which consists of the mesh files in OpenFOAM native format. In addition to this, two dictionary files are found in the constant sub directory, RASProperties and transportProperties, for definition of selection and definition of a turbulence model and kinematic viscosity, respectively. RASProperties dictionary file allows the user to select a suitable turbulence model and change any empirical constant for the turbulence model. Depending on if the flow is compressible or incompressible, single phase or multiphase, the transportProperties dictionary file allows the user to assign different transport equations and volume fraction. The polyMesh sub directory stores the mesh file in native OpenFOAM format. Unlike many commercial packages, OpenFOAM offers the flexibility of changing mesh attributes after the mesh is imported using dictionary files for mesh points, mesh faces, mesh boundary and neighbor cells. Any incorrect assignment of patch name can be easily changed by editing the boundary dictionary file. Fig.2.2 illustrates the boundary dictionary file which specifies the patch type, number of faces per patch and index number of the start face for each patch. With the exception of the face attributes, patch properties can be
easily changed. This functionality has been crucial for the current study as it allows mesh manipulation after mesh import from a third party software (discussed in detail in upcoming section).

```
{
    faRField
    
    
    
    type patch;
    nFaces 72388;
    startFace 18912588;

    sym
    
    {type symmetry;
      inGroups 7{symmetry};
      nFaces 46471;
      startFace 10984918;
    }

    using
    
    {type wall;
      inGroups 1{wall};
      nFaces 81627;
      startFace 14402989;
    }

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

Figure 2.2: Dictionary file for boundary

### 2.1.3 System directory

The system sub directory consists of a set of dictionary files which control the numerical details of the solver. The `controlDict` dictionary is used for defining the simulation runtime control in addition to start and end times. This dictionary file allows the user to include `funtionObjects` within the dictionary to perform data sampling and force calculation using computed pressure and velocity fields. Fig.2.3 illustrates the controlDict file used in the steady state computations for the current study. The function object used for force calculation is "libforces.so".
The *fvSchemes* and *fvSolution* dictionary files allow the user to setup numeric schemes and assign pre-conditioners and matrix solvers for each flow field variable respectively. Depending upon factors such complexity of flow physics, mesh resolution, and chosen solver, the two finite volume (fv) dictionary files allow the user to make changes during simulation runtime. A detailed description of numeric solvers employed in the study will be discussed in the upcoming sections.
2.2 Computational Domain

To re-iterate, the current study focuses on flow field analysis around three span-load optimized wings. Spatial Discretization plays a crucial role in determining the accuracy of numerical results. In a finite volume solver such as OpenFOAM, a computational domain consists of three dimensional cells. These cells may vary in shapes and sizes depending upon the type of mesh constructed (Barth, 1992; Loehner, 1993; Frohlich et al., 1998). The mesh generation procedure plays a crucial role in achieving the objective of this study. Each wing, though subjected to similar flow conditions, has different geometrical properties. Such differences result in different meshing approaches within the same mesh generator. In the following sections, mesh quality metrics critical for obtaining a good mesh are discussed. In addition to this, different types of computational meshes and rationale for selecting a type of mesh for this study are also discussed.

2.2.1 Mesh quality

The following factors play an important role in measuring the overall quality of a mesh:

1. Convergence rate: Higher quality meshes will typically converge faster than a low quality mesh. A poor mesh, for example may consist of insufficient grid points in the boundary layer vicinity, which may lead to a delayed convergence or divergence of solution.

2. Accuracy of solution: Grid validation studies indicate that a better mesh quality may result in a more accurate solution. For a low quality mesh, insufficient or misaligned grid points may result in over or under prediction of flow field data.

3. Computation cost: The cost of computing depends mainly on the number of grid points in a mesh. A good mesh should have the right amount of cells necessary to resolve the flow accurately. Any reduction or increase in the number of cells may result in either compromise of solution accuracy or increased computation cost.

The above stated attributes are necessary in constructing a good mesh. However, there exist a set of quality metrics which play an instrumental role in any mesh generation procedure. A mesh which satisfies these quality metrics can automatically be attributed with the above three mesh quality parameters. The quality metrics are as follows:
1. Aspect ratio: This parameter refers to the ratio of the longest side of a cell to its shortest side. For mathematical stability, the aspect ratio should ideally be equal to one. However, generating a mesh with all cells of aspect ratio of 1 is difficult. More so, this scenario would result in a mesh with no cell compression or high resolution in areas where flow gradients are high. Cells of aspect ratio 1 may also result in poor resolution of geometrical features of the native CAD geometry, thus altering the underlying shape of the mesh. OpenFOAM allows a the highest mesh aspect ratio of 1000. Hence a compression ratio of 1000:1 can be achieved near the boundary layer region. As the flow Reynolds number decreases, the compression ratio increases to resolve the viscous damping effects in the near wall region. Any value of aspect ratio greater than 1000 can result in interpolation errors which may result in divergence of the solution.

2. Skewness: Skewness corresponds to the internal angle between the two vertices of a mesh. For a unstructured computational domain consisting of tetrahedral cells, the skewness can be expressed as:

\[
Skewness = \frac{\text{reference optimal cell size} - \text{cell size}}{\text{reference optimal cell size}}
\]  

(2.2)

For prisms and pyramids, the skewness may be expressed as:

\[
Skewness = \max \left[ \frac{\theta_{\text{max}} - 90}{90}, \frac{90 - \theta_{\text{min}}}{90} \right],
\]

(2.3)

where \(\theta_{\text{max}}\) and \(\theta_{\text{min}}\) is the largest and smallest angle in a face or cell respectively.

In OpenFOAM, the skew limit for any given cell is 0.9. A value greater than 1 results in cell interpolation error. Even with the use of corrector schemes (Moukalled et al., 2015), the solution convergence may slow down or even diverge if the number of skewed cells is large.

3. Non-Orthogonality: This parameter refers to the misalignment of cells at an angle greater than 90° with respect to adjacent surface. Cell non-orthogonality in a finite volume solver such as OpenFOAM results in solution break down when computing gradients in such cells (Lien et al., 1996). For three dimensional meshes such as the
wing meshes, non orthogonal cells are produced close to the trailing edges. However, the grid is classified as acceptable if the mesh non-orthogonality is below 70°. In the current study, with the use of appropriate corrector schemes, meshes with cell non-orthogonality up to 88° have seen to yield good results. Any degree greater than 88° results in solution divergence.

2.2.2 Grid Classification

Computational grid may be classified as follows:

1. Structured grids: Structured grids consist of cells arranged with regular connectivity. Such cells are shaped quadrilateral for a 2D domain and hexahedral for a 3D domain. Due to the regular connectivity, structured meshes are highly space efficient, resulting in a mesh with the correct number of cells to obtain a stable and accurate solution with significantly reduced computation costs (Badcock et al., 2000). However, the application of a purely structured mesh is limited. Mostly used for symmetric geometries, application of a purely structured mesh for complex 3D geometries may result in skewed cells.

2. Unstructured grids: Unstructured grids consist of cells with irregular connectivity. Such cells may be shaped as triangles in a 2D domain and as tetrahedral, pyramids, and prisms in a 3D domain. Due to the complex structure and irregular arrangement of cells, these grids are highly space inefficient, resulting in slower convergence rate when compare to that of a structured mesh. Such grids find vast applications due to their adaptability to any given geometry shape and size. Due to this feature, the cells produced are relatively less skewed. However, the number and degree of non-orthogonal cells increase when compared to the structured domains. Hence, with careful implementation of meshing algorithms (Ruppert, 1992; Shewchuk, 2002), an acceptable mesh can be obtained.

3. Hybrid grids: A hybrid grid is a combination of both unstructured and structured domains. Generally, a hybrid mesh consists of a structured domain in the region close to the walls and unstructured domains in the farfield region. One motivation behind generating such meshes is the reduced number or even avoidance of non orthogonal cells close to the walls.
Figs. 2.4, 2.5 and 2.6 illustrate the three mesh types for a NACA 0015 wing section.

For the current study, a three dimensional unstructured mesh was chosen for the flow.
field computations. Although a hexahedral structured mesh would have been ideal, the complex geometry of all three wings resulted in skewed cells in the wing tip region. Additionally, since the scope of the current study was to perform flow field analysis at low, pre-stall angles of attack, generation of a purely structured grid was time expensive and was deemed unnecessary, since unstructured grids provided similar solution accuracy. An O-grid topology was chosen over a C-grid and H-grid topology due to factors such as re-constructibility and cell conservation. Due to the irregular arrangement of cells in the computational domain, a large number of skewed and non-orthogonal cells were initially obtained. However, by adjusting the growth rate in the near wall region and reducing the collision buffer (Pointwise), the required quality metrics were achieved. Fig. 2.7 shows the mesh quality report from OpenFOAM.
Figure 2.7: Mesh quality report for Elliptic Wing case

2.2.3 Mesh generation

Mesh generation in CFD is typically performed with the help of pre-defined algorithms. Such algorithms (Smith, 1982) may follow a purely algebraic method, differential equations method (Thompson, 1985) or a variations method. All three dimensional meshes generated for this study use Hyperbolic schemes (Steger and Chaussee, 1980). The choice of numerical scheme chosen was purely dictated by the geometrical parameters for
all three wings and the ease of mesh generation in the mesh generator. More on mesh generation algorithms can be found in Thompson et al. (1997).

**snappyHexMesh**

In order to maintain consistency in the process of mesh generation, implementation, and obtaining solutions, *snappyHexMesh*, a structured mesh generator distributed with OpenFOAM was used. Mesh generated using is solver consists mainly of hexahedral cells and pyramids at locations where mesh structured domains cannot resolve the complex geometry. The main advantage of using such a solver is twofold. First, the generation of structured surface domains and volumetric blocks. Secondly, the quality metrics, which are critical to the overall quality of the mesh, are automatically fulfilled by every mesh generated by *snappyHexMesh*. This feature highlights the importance of consistency in the process mentioned earlier. The mesh generating algorithm was performed the following steps:

1. Geometry import: Geometry imported in *stl* format.
3. Mesh snapping: Surface and volume mesh are formed by snapping cells to the edges.
4. Layer addition: Addition of cells close to the walls to resolve the boundary layer physics.

While *snappyHexMesh* is an effective meshing tool, its use in the current study was limited due to its implementation based on a dictionary file rather than a graphical user interface. In addition to this limitation, the lack of proper documentation was a crucial factor which lead to the use of a third party mesh solver for this study. Based on available literature (Higuera et al., 2013; Pruna and Merchant, 2013), steps 1, 2 and 3 were implemented accurately in *snappyHexMesh*. One crucial limitation, the layer addition step, which is necessary for resolving boundary layer physics such as flow separation and transition physics, could not be implemented successfully. A significant amount of parametric study was conducted as a part of this study to understand the various factors within the algorithm which may control the layer addition process. Table 2.2 below lists results of this parametric study which yielded the maximum number of layers.
Figure 2.8: Structured mesh for NACA6412 wing section made using \textit{snappyHexMesh}

<table>
<thead>
<tr>
<th>addLayer control variables</th>
<th>Assigned values</th>
<th>Remark</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of layers</td>
<td>10</td>
<td>All layers added</td>
</tr>
<tr>
<td>Max layer size</td>
<td>0.3 m</td>
<td>Obtained 0.24 m</td>
</tr>
<tr>
<td>Min layer size</td>
<td>1e-05 m</td>
<td>Obtained 1.29e-03 m</td>
</tr>
<tr>
<td>Exp ratio</td>
<td>1.2</td>
<td>Implemented successfully</td>
</tr>
<tr>
<td>Feature angle</td>
<td>$60^\circ$</td>
<td>–</td>
</tr>
<tr>
<td>Medal Ratio</td>
<td>0.3</td>
<td>–</td>
</tr>
</tbody>
</table>

Table 2.2: Maximum number of layers added using \textit{snappyHexMeshDict}

As observed above, the maximum number of layers that could be added to the mesh were limited to 10. This resulted in high computed values of turbulent parameters close to the wall and into the farfield. The Meshes obtained from \textit{snappyHexMesh} were observed to yield stable and accurate solutions in the low angle of attack regime. However, with an increase in angle of attack, the near-wall flow gradients increased and solution diverged after a few hundred steps resulting in an unstable solution.

Hence, to summarize, although \textit{snappyHexMesh} preserves consistency in the entire CFD process, the limited number of layers added enables the mesh to be used with the wall function approach (Blocken et al., 2007; Rakopoulos et al., 2010; T.J.Craft et al., 2002; Guardo et al., 2005). Since the current study intends to use the integration method
in the near-wall region (Hartwanger and Horvat, 2008), a refined mesh near the wall was necessary. As a result, a third party meshing tool, Pointwise, was instead used in the current study.

**Pointwise**

Pointwise is a commercial CFD mesh generator which allows the user to generate high quality structured and unstructured meshes. Based on a core meshing methodology different from *snappyHexMesh*, Pointwise allows the user to control and implement certain meshing parameters which are critical in controlling the overall quality of the mesh and also the required quality metrics.

In the current study, the geometries were imported into Pointwise in the IGES (Liu et al., 1996) format. The IGES format imports the edges of wing geometry which makes the surface creation easier. While this import approach is case specific, it was deemed appropriate for all three geometries in the current study. Pointwise works on grid point methodology as opposed to the "surface-snapping" mesh generation approach in *snappyHexMesh*. The following steps describe the meshing procedure in Pointwise:

Step 1: The first step involves the import of geometry in IGES format, producing results in a wireframe of the wing with database entities defining the edges of the geometry.

Step 2: Each database entity is assigned a number of grid points with defined grid spacing. The grid spacing or the spacing constraint allows the compression or expansion of grids at certain locations thus preserving the number of global cells.

Step 3: All the database entities from step 2 are selected to form a surface mesh or domain. This may be unstructured or structured depending upon the underlying geometry.

Step 4: This step involves the creation of the farfield domain. Initially defined as a database entity, the circular database entity is first mirrored to complete a circle and then revolved to form a semi-hemispherical domain. All new database entities are joined together to form a unstructured domain. The last part of this step requires a domain assembly between the surface domain from step 3 and the new farfield domain from step 4.
Step 5: This step is considered the most crucial of all since it defines the mesh fidelity in terms of capturing and resolving the near-wall physics. The surface mesh and farfield domains are selected to form an empty block or a volumetric part of the mesh with no cells to begin with. Depending upon the flow Reynolds number, an initial cell height is chosen with an appropriate expansion ratio. Three dimensional anisotropic extrusion is done at the wing surface towards the farfield volume. This extrusion allows the generation of high aspect ratio cells which are required to maintain a yplus (Ai and Mak, 2013; Kim et al., 2001; Laccarino, 2001) value close to unity. Once the T-rex (Tetrahedral extruded anisotropic) cells reach isotropy, further addition of T-Rex cells is stopped and remaining volume is populated with pyramids (Steinbrenner and J.Abelanet, 2007).

Step 7: The last step involves the boundary conditions assignment. Each patch is defined separately and is assigned a patch type based on those listed in Table 2.1. Fig.2.9 represents the various steps involved in the meshing procedure in Pointwise. The representations illustrate the steps involved in mesh development for the Elliptic Wing case.
Fig. 2.9 represents the mid-span slice view for Elliptic Wing mesh after import into the OpenFOAM framework. A baffle is added to the wing starting from the trailing edge and extending up to five chord lengths downstream. A baffle is a thin surface of cells from which cells in both upward and downward directions are extruded. The main purpose for adding a baffle was to perform high quality wake analysis further into the study. Fig. 2.11 represents a close up of the slice view of wing leading and trailing edges.
Figs. 2.12, 2.13 and 2.14 illustrate the three quality metric statistics for each of the three wing meshes. Since the three geometries are significantly different due to factors such as geometrical dimensions and features such as twist distributions, different meshing parameters were established for each of the three wings. Changes in these meshing parameters lead to different ranges of centroid skewness and maximum included angle among the three meshes, as is clearly evident from the figures below.
Figure 2.12: Quality metrics for Elliptic Wing mesh
Figure 2.13: Quality metrics for Viscous Optimized wing mesh
Figure 2.14: Quality metrics for Inviscid Optimized Wing mesh
2.3 Turbulence Modeling

Turbulence is a fluid property which attributes to the chaotic property changes. Such changes include variation in pressure and velocity fields over time, characterized with low momentum diffusion and high momentum convection. A turbulent flow has two key properties: Irregularity and Diffusivity. Turbulent flows are highly irregular in spatial and temporal frames. Due to this irregular nature, turbulent problems are studied statistically instead of deterministically. Diffusivity in turbulent flows correspond to the available energy in the flow which cause homogenization or mixing in the flow field. Figs.2.15 and 2.16 represent the velocity fields, \( U_1(t) \) and \( U_2(t) \), from a backward facing CFD benchmark problem (B.F.Armaly et al., 2006; Hung et al., 1997; Gartling, 1990), obtained using Large Eddy Simulation (LES) (Gartling, 1990; Moeng and Sullivan, 2015; J.Bardina et al., 1980; P.Saguat, 1998). Visualizations are carried out at two different time steps to illustrate the chaotic nature of the fluid which evolve over time. Both visualizations indicate significant fluctuations in the flow field and by no means the flow field is periodic. Figs.2.17 and 2.18 represent the mean flow fields, \( \langle U_1 \rangle \) and \( \langle U_2 \rangle \), at the same time steps. The mean flow field visualization is devoid of any fluctuations since the flow variables are statistically averaged over time.

![Figure 2.15: Visualization for \( U_1(t) \)](image)
In turbulence modeling, the ultimate objective is to obtain a numeric code which can quantify the rapidly changing flow variables. The turbulent velocity field $U(x, t)$

Figure 2.16: Visualization for $U_2(t)$

Figure 2.17: Visualization for $\langle U_1 \rangle$

Figure 2.18: Visualization for $\langle U_2 \rangle$
is three dimensional, non-periodic and time dependent. The largest turbulent eddies formed correspond to the characteristic width of the flow, which is in turn dependent on the geometrical boundaries and are hence not universal. Characterization of turbulent flow eddies is based on a set of time and length scales (Yang and U. Lei, 1998; Chamorro et al., 2012). The Kolmogorov lengthscale, represents the smallest scales in the viscous sub-layer range spectrum, decrease as $Re^{-3/4}$. Relative to the Integral scales, the Kolmogorov timescale decreases as $Re^{-1/2}$. Challenges in modeling arise from the pressure gradient term and the nonlinear convective term in Navier-Stokes equations. Based on the solution to the Poisson equation (Eq. 2.4), the pressure gradient term is both non-local and nonlinear when expressed in terms of velocity (J. Kim and P. Moin, 1985; U. Ghia et al., 1982).

\[ p(x, t) = p^h(x, t) + \left( -\frac{1}{4\pi} \int \int_{y} \frac{\partial U_i \partial U_j}{\partial x_j \partial x_i} \frac{dy}{|x - y|} \right)_{y, t} \]  

(2.4)

where, $p^h$ is the harmonic function dependent on the defined flow boundary conditions.

As discussed previously, in a turbulent flow field simulation, a time dependent velocity field is studied by solving a set of equations which represent $U(x, t)$ for one realization of the turbulent flow. In general, turbulence models are employed to solve for mean quantities such as $\langle U \rangle$ and $\epsilon$. The following three approaches exist in turbulence modeling:

### 2.3.1 Direct Numerical Simulation

A simulation approach wherein the Navier-Stokes equations are solved numerically for all temporal and spatial scales. DNS does not use any turbulence model and hence attempts to solve the Navier-Stokes equation directly, resolving the smallest dissipative Kolmogorov micro-scales (Yang and U. Lei, 1998; Chamorro et al., 2012) and the largest integral scales corresponding to the geometric length $L$. Although the simplest approach in numerical modeling, this method is computationally expensive since the mesh requirements are extremely high to resolve all scales of turbulence.

\[ Nh > L \]  

(2.5)
Eq.2.5 expresses the relation between the number of points $N$ in any given direction with increments $h$ and integral length scale $L$. To resolve the micro scales, the computational domain must be in accordance with the Kolmogorov length scale as expressed in Eq.2.6.

$$h \leq \eta$$  \hspace{1cm} (2.6)

where the Kolmogorov scale $\eta$ is expressed as

$$\eta = (\nu^3/\epsilon)^{1/4}$$  \hspace{1cm} (2.7)

where $\nu$ is the kinematic viscosity and $\epsilon$ is the turbulent kinetic energy.

For a three dimensional flow field analysis, $N^3$ points are needed, thus resulting in computational cost approximately in the order of $Re^3$. Although DNS offers the computational results with unparalleled accuracy (Wilcox, 1994; Moser et al., 1999), the large computational costs involved and the available time frame for the current study make it an unsuitable option for flow modeling.

2.3.2 Large Eddy Simulation

LES, first developed and studied by Smagorinsky (1963) and Deardoff (1970), is a modeling approach which resolves only a certain range of time and length scales of turbulence. This method was developed in the 1960s and explored further from the 1970’s and onwards as an alternative to the computationally expensive DNS. In LES, a low pass filter (Wilcox, 1994) is applied to the flow field for spatial and temporal filtering. Consider a spatial and temporal field $\phi(x, t)$. The filtered field may hence be expressed as (Pope, 2000; P.Saguat, 1998):

$$\bar{\phi}(x, t) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \phi(r, t')G(x-r, t-t')dt'dr$$  \hspace{1cm} (2.8)

where, $G$ is the convolution kernel and can be expressed as

$$\bar{\phi} = G * \phi$$  \hspace{1cm} (2.9)

Here $\bar{\phi}$ is the filtered field. The governing equations for LES are obtained by applying the aforementioned low pass filter to the continuity and Navier Stokes Equation (Batch-
For incompressible flows, the continuity equation (Pedlosky, 1987) may be expressed as

\[ \frac{\partial \rho}{\partial t} + \nabla \cdot (\rho \mathbf{u}) = 0 \quad (2.10) \]

After low pass filtering, Eq.2.10 becomes

\[ \frac{\partial \bar{u}_i}{\partial x_i} = 0 \quad (2.11) \]

Similarly, the filtered Navier Stokes equations (Wilcox, 1994) take the form

\[ \frac{\partial \bar{u}_i}{\partial t} + \frac{\partial}{\partial x_j}(\bar{u}_i \bar{u}_j) = -\frac{1}{\rho} \frac{\partial \bar{p}}{\partial x_i} + \nu \left( \frac{\partial \bar{u}_i}{\partial x_j} + \frac{\partial \bar{u}_j}{\partial x_i} \right) = -\frac{1}{\rho} \frac{\partial \bar{p}}{\partial x_i} + 2\nu \frac{\partial}{\partial x_j} S_{ij} \quad (2.12) \]

where, \( S_{ij} \) is the rate of strain tensor and \( \bar{p} \) is the filtered pressure. The main challenge in LES modeling arises due to the nonlinear advection term \( \bar{u}_i \bar{u}_j \). The unfiltered velocity field \( U(x,t) \) is required to compute the advection term. Leonard (1974) discusses this problem comprehensively.

LES implementation in OpenFOAM is done using the sub grid scale models (Bensow and Goran, 2010). Although this approach is more robust than the RANS-based solvers, the computational costs involved are high due to high resolution mesh requirements. In addition to this high cost, the object of this study is to investigate the aerodynamic performance of the different wing geometries rather than turbulent flow characterization for which LES would have been a suitable candidate.

### 2.3.3 Reynolds Averaged Navier-Stokes

The most widely used modeling approach, the Reynolds time Averaged Navier Stokes equation model decomposes the flow field into fluctuating and time averaged quantities (Reynolds, 1883). Due to the time averaging, this modeling approach resolves turbulent flow fields in an effective yet computationally in expensive manner. Eq.2.13 represents the mean momentum equation with the turbulent viscosity hypothesis (Wilcox, 1994).

\[ \frac{D}{Dt} \langle U_j \rangle = \frac{\partial}{\partial x_i} \left[ \nu_{eff} \left( \frac{\partial \langle U_j \rangle}{\partial x_j} + \frac{\partial \langle U_j \rangle}{\partial x_i} \right) \right] - \frac{1}{\rho} \frac{\partial \bar{p}}{\partial x_j} \langle \bar{p} \rangle + \frac{2}{3} \rho k, \quad (2.13) \]
where $\nu_{eff}$ is the effective viscosity

$$\nu_{eff}(x, t) = \nu + \nu_T(x, t), \quad (2.14)$$

and $(\langle p \rangle + \frac{2}{3}\rho k)$ is the modified mean pressure.

The unknown Reynolds stresses in Eq.2.13 are usually computed from a suitable turbulence model using Reynolds Stress transport equations or turbulent viscosity hypothesis. Since the current study employs one and two equation algebraic models, Reynolds Stress transport models are not considered.

According to turbulent viscosity hypothesis, the Reynolds stresses in Eq.2.13 are expressed as

$$\langle u_i u_j \rangle = \frac{2}{3} k \delta_{ij} - \nu_T \left( \frac{\partial \langle U_i \rangle}{\partial x_j} + \frac{\partial \langle U_j \rangle}{\partial x_i} \right), \quad (2.15)$$

and for a simple shear flow, the Reynolds shear stress becomes

$$\langle uv \rangle = -\nu_T \frac{\partial \langle U \rangle}{\partial y} \quad (2.16)$$

From Eq.2.12, 2.13 and given $\nu_T(x, t)$, a reasonable closure to the Reynolds equations is achieved. However, the RANS-based approach is associated with limitations in terms of applicability to various flow types. The main reason behind such a limitation is the poor agreement with the turbulent viscosity hypothesis. This limitation can be viewed in two parts. First the intrinsic assumption (Wilcox, 1994), which highlights the failure of the hypothesis in an axisymmetric contraction case (Uberoi, 1956). Secondly, the specific assumption (Wilcox, 1994), which highlights the relationship between Reynolds Stresses and mean velocity gradients as expressed in Eq 2.13.

With the advent of computing technology and the need to solve and analyze flow fields numerically, various turbulence models have been developed. Celik and Wolfgang (1988) in their study of Turbulence Modeling, comprehensively describe the evolution of such models. The turbulence models used toady may be categorized as follows:

1. Algebraic Models
2. One-Equation Models
3. Two-Equation Models

4. Second Order Closure Models

Prandtl (1945) proposed a one equation turbulence model wherein the eddy viscosity was a function of the turbulent kinetic energy $k$. This model solved a differential equation to approximate an exact solution for $k$. Various computational studies (Celik and Wolfgang, 1988) indicated considerable improvements in turbulence prediction. However, one limitation of the model was the absence of turbulence length scale (Yang and U.Lei, 1998). A more descriptive modeling of turbulence was required in order to characterize the length scale, which corresponds to the size of different eddies as this scale can be significantly, different for different flow conditions and types. Kolmogorov (1942) proposed a complete model for turbulence which modeled the turbulent kinetic energy $k$ and an additional parameter $\omega$ which was referred to as the specific energy dissipation rate per unit volume and time. This model, is more commonly referred to as the $k$-$\omega$ model (Menter, 1992). The turbulence time scale was incorporated into the model as the reciprocal of $\omega$. Given the complex modeling approach this model did not see much application until computing technology progressed in the late 1970s. Since then, various modifications have been made to the original model and new turbulence models incorporating a shear stress transport equation (Menter, 1992) and laminar and turbulent kinetic energy (Walters and Cokljat, 2008; Frust, 2012) have been introduced.

In the following section, a brief overview of the turbulence models used in this study is presented. Although a large number of studies have benchmarked each of these 4 models (Rumsey et al., 2010; Fernandes et al., 2015; Crivellini et al., 2013; Chen and Kim, 1987; Walters and Cokljat, 2008; Khaled, 2015) preliminary computational investigation was carried out to select the most appropriate model. Five factors, namely, level of description, accuracy, completeness, computational cost and applicability were used for model appraisal and selection. More on these factors is presented in Pope (2000).

2.3.4 Overview of Turbulence Models

Spalart-Allmaras model (SA)

The Spalart-Allmaras model is a one-equation model, developed at Boeing, with a sole objective of investigating external aerodynamics problems (Spalart and Allmaras, 1992).
Multiple modifications have been made to the model over time (Allmaras and Johnson, 2012). This evolution of the SA model is also seen in the earlier versions of OpenFOAM such as version 1.6 and version 2.4. The approach here is to solve for the kinematic eddy viscosity parameter $\tilde{\nu}$. The transport equation for the SA model is expressed as (Spalart and Allmaras, 1992):

$$\frac{D\tilde{\nu}}{Dt} = P - D + T + \frac{1}{\sigma} \left[ \nabla \cdot \left( \left( \nu + \tilde{\nu} \right) \Delta \tilde{\nu} \right) + c_{b2} \left( \nabla \tilde{u} \right)^2 \right]$$  \hspace{1cm} (2.17)

where, $P$ and $D$ are the production and destruction terms respectively and are expressed as:

$$P = c_{b1} (1 - f_{t2}) \tilde{S} \tilde{\nu},$$ \hspace{1cm} (2.18)

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

$$\tilde{S} \equiv S + \frac{\tilde{\nu}}{k^2d^2 \tilde{f}_v}$$ \hspace{1cm} (2.19)

and

$$D = \left( c_{w1} f_w - \frac{c_{b1}}{k^2 d^2 \tilde{f}_t} \right) \left[ \frac{\tilde{\nu}}{d} \right]^2$$ \hspace{1cm} (2.20)

The trip term $T$ is expressed as:

$$T = f_{t1} (\Delta U)^2$$ \hspace{1cm} (2.21)

Here, $d$ represents the first cell height adjacent to the wall and $S$ is the vorticity. In Eq.2.17,

$$f_{v1} = 1 - \frac{\chi}{1 + \chi f_{v1}},$$ \hspace{1cm} (2.22)

$$f_w = g \left[ \frac{1 + c_{w3}^3}{g^6 + c_{w3}^6} \right]^{1/6},$$ \hspace{1cm} (2.23)

$$g = r + c_{w2} (r^6 - r),$$ \hspace{1cm} (2.24)
\( r = \min\left( \frac{\tilde{\nu}}{Sk^2d^2}, r_{\text{lim}} \right) \) \hspace{1cm} (2.25)

\[ f_{t1} = c_{t1}g_{t}\exp\left( -c_{t2} \frac{\omega_{t}}{\Delta u^{2}} [d^{2} + g_{t}^{2}d_{t}^{2}] \right) \] \hspace{1cm} (2.26)

\[ f_{t2} = c_{t3}\exp(-c_{t4}\chi^{2}) \] \hspace{1cm} (2.27)

and

\[ g_{t} = \min\left( 0.1, \frac{\Delta}{\omega_{t}\Delta x} \right) \] \hspace{1cm} (2.28)

Here \( \Delta u \) is the difference in relative velocity at the trip point, \( \Delta x \) is the stream wise grid spacing at the trip point and \( \omega_{t} \) is the vorticity at trip location. The wall damping functions are denoted by \( f_{t1}, f_{t2} \) and \( f_{w} \). The model constants used in the current SA model version can be found in (Greenshields, 2015). The eddy viscosity ratio \( \chi \) is used to estimate the overall turbulent nature of the flow. From (Spalart and Allmaras, 1992), for \( 3 \leq \chi \leq 5 \), the flow is considered to range from borderline turbulent to fully turbulent. It should be noted that such estimations are valid only as an initial guess. \( \tilde{\nu} \) may be initially estimated using:

\[ \tilde{\nu} = \sqrt{\frac{3}{2}IU_{Avg}l}, \] \hspace{1cm} (2.29)

where, \( I \) is the turbulent intensity, \( l \) is the length scale and \( U_{Avg} \) is the average flow field velocity.

Variables \( \tilde{\nu} \) and \( \chi \) are relatively easy to compute in the near-wall region as compared to \( U \). This hence results in a lower mesh resolution for the SA model when compared to the two-equation models. Ease of computation and faster numerics where the two motivating factors behind the development of SA at an industrial setting such as Boeing.

**k-Epsilon model (k-\( \epsilon \))**

Developed by Jones and Launder (Jones and Launder, 1973), the k-Epsilon Model is a two equation model which as the name suggests, solves two equations, one each for turbulent kinetic energy \( k \) and turbulent dissipation rate \( \epsilon \). For the sake of brevity, the
derivation of governing equations from averaged Navier Stokes equations is not discussed here and can be found in Jones and Launder (1973).

At any given time, the kinetic energy \( k_t \) of a flow field equals the summation of the mean kinetic energy \( K \) and turbulent kinetic energy \( k \). The governing equation for \( k \) is expressed as (Jones and Launder, 1973):

\[
\frac{\partial (\rho k)}{\partial t} + \text{div}(\rho k \mathbf{u}) = \text{div}(-p' \mathbf{u}' + 2\mu s_{ij} - \frac{1}{2} u_i' u_i' u_j') - 2\mu s_{ij} \cdot s_{ij} + \rho u_j' u_j' \cdot s_{ij} \quad (2.30)
\]

\( k \) is influenced by viscous stresses in two ways: the transport of \( k \) \( 2\mu U s_{ij} \) and viscous dissipation \( 2\mu s_{ij} \cdot s_{ij} \). In addition to the viscous contributions, the term \( \rho U u_j' u_j' \) signifies the transport of \( k \) due to Reynolds shear stress. The second term \( \rho u_j' u_j' \cdot s_{ij} \) is the decrease in \( k \) due to deformation. As the flow Reynolds number increases, the transport of \( k \) and decrease in \( k \) become significant as compared to the viscous part (Wilcox, 1994).

The \( k - \epsilon \) model in OpenFOAM is one of the most widely used turbulence models. Three equations, one for \( k \), \( \epsilon \), and \( n_t \), are solved. The length and velocity scales are defined using \( k \) and \( \epsilon \), respectively. The turbulent length scale is defined as a function of flow Reynolds number:

\[
\delta_{0.99} = \frac{0.374 l}{Re^{\frac{1}{2}}}, \quad (2.31)
\]

where \( l \) is the turbulent length scale and \( \delta_{0.99} \) is the boundary layer thickness. The turbulent intensity \( I \) defines the vorticity strength at the inlet. In the current study, the turbulence intensity is matched to the wind tunnel \( I \) for near accurate flow analysis. The following expressions are used to estimate initial values for \( k \), \( \epsilon \) and \( n_t \):

\[
k = \frac{2}{3} (U_{ref} I)^2 \quad (2.32)
\]

\[
\epsilon = C_\mu \frac{k^{\frac{3}{2}}}{l} \quad (2.33)
\]

\[
n_t = 1.2 \tilde{\nu} \quad (2.34)
\]

Here, \( \tilde{\nu} \) is obtained from Eq.2.29. It should be noted that Eq.2.34 expresses a rough
approximation for $\nu_t$.

The $k-\epsilon$ model used in this study utilizes an integration method in the wall proximity rather than the wall function approach, thus avoiding the universal flow behavior assumption near the boundary layer. As a result an increase in computational costs and high mesh resolution requirements are produced.

**k-omega SST model ($k-\omega$ - SST)**

A two equation model first proposed by Menter (Florian Menter, 2006), the $k-\omega$ SST model is used in the sub layer region of boundary layer. This model computes turbulent flow field without any damping functions and is hence superior to the $k-\omega$ models. Based on the discussion on the $k-\epsilon$ model and its independent behavior from the freestream values, the $k-\omega$ SST model is based on similar framework. The governing equation is expressed as (Florian Menter, 2006):

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

(2.35)

Similar to the $k-\epsilon$ model, the $k-\omega$ SST model initializes the flow field using the following three variables:

$$k = \frac{2}{3} (U_{ref} \bar{l})^2 \quad (2.36)$$

$$\omega = C_{\mu}^{-1/4} \frac{\sqrt{k}}{l} \quad (2.37)$$

$$\nu_t = \frac{k}{\omega} \quad (2.38)$$

Eqs.2.34 and 2.38 yield the same value of $\nu_t$. The above parameters are defined for the freestream. Since the above expressions are a function of $U$, appropriate boundary conditions are applied near the wall. While the $k-\omega$ SST model can model the near wall flow field using wall function approach (Florian Menter, 2006), the current study employs the model using integration method.
2.3.5 kkl-omega model \((k - kl - \omega)\)

Modeling of laminar flows is relatively straightforward when compared to highly complex turbulent flows. For a laminar flow simulation, the solver approximates the Navier-Stokes equations without the use of a turbulence model. However, for a turbulent boundary layer, highly analytical models such as those discussed previously are used. Though most of these models deliver near accurate results, no one model can be considered a good fit for all kinds of flow regimes. Another shortcoming of these models is their inability to predict flow transition from laminar to turbulent regimes. This flow transition behavior is considered crucial for many engineering design problems such as design of wind turbines, wings (Khaled, 2015) and problems involving heat transfer and losses (Peng and Peterson, 1996). Studies in the past have attempted to modify the basic models (Wilcox, 1994), TSL model of Zheng et al. (1998) and have delivered near accurate results. However, recent studies (Florian Menter, 2006) have shown that these models fail to resolve all the factors that lead to transition. Algebraic models proposed by (Straka and Pnhoda, 2010), which are based on empirical correlations, predict transition flows with a simple approach and acceptable accuracy. However, application of these models are limited to structured grids mainly due to non-local information such as momentum boundary layer thickness which are difficult to compute. Hence this limits the application to academic codes working on structured meshes.

The \(kkl - \omega\) model is a three equation RANS-based flow transition model first proposed by Walters and Cokljat (Walters and Cokljat, 2008) and later developed by Frust (Frust, 2012). Based on the low Reynolds number \(k - \omega\) two equation model with an additional third equation for kinetic laminar energy \(k_L\), this model overcomes the limitation of previously mentioned models due to local formulation. The current implementation of the \(kkl - \omega\) model is of the latest corrected version as proposed by Frust (Frust, 2012). Previous versions of this model as proposed earlier by Walters and Cokljat (Walters and Cokljat, 2008) under predicted the wall shear stress in the post transition regime. The transport equation may be expressed as (Walters and Cokljat, 2008):

\[
\frac{Dk_t}{Dt} = P_{K_t} + R_{BP} + R_{NAT} - \omega k_t - D_t + \frac{\partial}{\partial x_i} \left[ \left( \nu + \frac{\alpha T}{\alpha K} \right) \frac{\partial k_t}{\partial x_j} \right] \tag{2.39}
\]
\[
\frac{Dk_L}{Dt} = P_{KL} - R_{BP} - R_{NAT} - D_L + \frac{\partial}{\partial x_j} \left[ \nu \frac{\partial k_L}{\partial x_j} \right]
\]  \hspace{1cm} (2.40)

\[
\frac{D\omega}{Dt} = C_{\omega_1} \frac{\omega}{k_t} P_{k_t} + \left( C_{\omega R} \frac{1}{f_W} \right) \frac{\omega}{k_T} (R_{BP} + R_{NAT}) - C_{\omega_2} \omega^2 + C_{\omega_3} f_\omega \alpha_T \sqrt{k_T} \frac{\partial}{\partial x_j} \left[ \left( \nu + \frac{\alpha_T}{\alpha_\omega} \right) \frac{\partial \omega}{\partial x_j} \right]
\]  \hspace{1cm} (2.41)

Above, \( R_{BP} \) is referred to as the averaged effect of fluctuations breakdown into turbulence at bypass transition and may be expressed as:

\[
R_{BP} = C_R \beta_{BP} k_L \omega / f_W,
\]  \hspace{1cm} (2.42)

where the threshold function \( \beta_{BP} \) responsible for controlling bypass transition is expressed as:

\[
\beta_{BP} = 1 - \exp \left( \frac{\phi_{BP}}{A_{BP}} \right),
\]  \hspace{1cm} (2.43)

where

\[
\phi_{BP} = \max \left[ \left( \frac{k_T}{\nu \Omega} - C_{B,P_{crit}} \right), 0 \right]
\]  \hspace{1cm} (2.44)

The natural production term \( R_{NAT} \) is defined as:

\[
R_{NAT} = C_{R,NAT} \beta_{NAT} k_L \omega
\]  \hspace{1cm} (2.45)

where,

\[
\beta_{NAT} = 1 - \exp \left( - \frac{\phi_{NAT}}{A_{NAT}} \right)
\]  \hspace{1cm} (2.46)

Near-wall dissipations are expressed as:

\[
D_T = 2\nu \frac{\partial \sqrt{k_t}}{\partial x_j} \frac{\partial \sqrt{k_T}}{\partial x_j}
\]  \hspace{1cm} (2.47)

\[
D_L = 2\nu \frac{\partial \sqrt{k_L}}{\partial x_j} \frac{\partial \sqrt{k_L}}{\partial x_j}
\]  \hspace{1cm} (2.48)

where the subscripts \( T \) and \( L \) refer to turbulent and laminar regimes respectively.
The damping function is defined as:

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

(2.49)

The production terms for turbulent kinetic energy and laminar kinetic energy may be expressed respectively as:

\[ P_{KT} = \nu_{T,s} \dot{S}^2 \]  

(2.50)

and

\[ P_{kL} = \nu_{T,l} \dot{S}^2 \]  

(2.51)

The small scale turbulent viscosity is expressed as:

\[ \nu_{T,s} = f_v f_{INT} C_\mu \sqrt{k_{T,s} \lambda_{eff}}, \]  

(2.52)

where \( f_v \) and \( f_{INT} \) are damping functions and are defined as:

\[ f_v = 1 - \exp \left( - \frac{\sqrt{Re_T}}{A_v} \right), \]  

(2.53)

and

\[ f_{INT} = \min \left( \frac{k_T}{C_{INT} k_{TOT}}, 1 \right) \]  

(2.54)

The large scale turbulent viscosity is defined as:

\[ \nu_{T,l} = \min \left( f_{\tau,l} C_{11} \left( \frac{\Omega_{\lambda_{eff}}^2}{\nu} \right) \sqrt{k_{T,l} \lambda_{eff} + \beta_{T,S} C_{12} Re_\Omega \Omega^2}, \frac{0.5(k_L + k_{\lambda,l})}{S} \right). \]  

(2.55)

where, the time scale damping function \( f_{\tau,l} \) is expressed as:

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

(2.56)

and

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

(2.57)

The large scale and small scale eddy viscosities contribute towards the total eddy viscosity as:

\[ \nu_{TOT} = \nu_{T,s} + \nu_{T,l} \]  

(2.58)
\[
\alpha_{\theta,TOT} = f_W \left( \frac{k_i}{k_{TOT}} \right) \frac{\nu_{T,s}}{Pr_\theta} + (1 - f_W) C_{\alpha,\theta} \sqrt{k_T \lambda_{\text{eff}}} \quad (2.59)
\]

The total kinetic energy is a summation of laminar and turbulent kinetic energies and may be expressed as:

\[ k_{TOT} = k_T + k_L \quad (2.60) \]

The mean flow is effected by the laminar and turbulent fluctuations as:

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

and thermal diffusivity as:

\[ -\overline{u_i \theta} = \alpha_{\theta,TOT} \frac{\partial \theta}{\partial x_i} \quad (2.62) \]

Turbulent kinetic energy consists of two parts in the near-wall region. These parts are responsible for production of turbulence and laminar kinetic energy are known as small scale and large scale energies respectively. The effective time and length scales are defined as:

\[ \lambda_T = \frac{\sqrt{K_T}}{\omega} \quad (2.63) \]

and

\[ \lambda_{\text{eff}} = \min(C\lambda d, \lambda_T) \quad (2.64) \]

A comprehensive description of the model and model constants are presented in Frust (Frust, 2012).

### 2.4 Numerical algorithms and Schemes

This study analyzes flow fields around three different wing geometries first using a steady state solver followed by a transient solver to emulate transition flow physics. The following section presents a brief overview of two algorithms used in this study.

#### 2.4.1 SIMPLE algorithm

SIMPLE or Semi-Implicit Method for Pressure Linked Equations is a numerical algorithm used for steady state flow field computations. First introduced by Brian Spalding
and Suhas Patankar (Patankar and Spalding, 1972) and later developed by Spalding (1972) and Patankar (1975), is an iterative algorithm which follows the listed sequence of operations:

1. Guess the initial pressure field $p^*$. 
2. Solve momentum equations to obtain velocity field estimates $u^*$, $v^*$ and $w^*$. 
3. Solve the $p'$ equation. 
4. Compute $p$ from the equation below by adding $p'$ to $p^*$.

$$p = p^* + p'$$

(2.65)

This step is called the pressure correction. 
5. Perform velocity corrections to calculate values of $u$, $v$, and $w$ from their starred values. 
6. Solve discretized equations for $\phi$ such as turbulence quantities and temperature fields that influence the fluid properties and source terms. If the given $\phi$ does not influence the flow properties, it is calculated after the flow solution has converged.
7. The corrected pressure $p$ is now the new estimated pressure $p^*$. Steps 2 and onwards are repeated until a stable converged solution is obtained.

### 2.4.2 PISO algorithm

PISO or Pressure Implicit with Splitting Operator, first proposed by Issa in 1986 (Issa et al., 1986), is an extension of the SIMPLE algorithm which computed flow fields without iterations. The algorithm is based on a pressure-velocity calculation procedure for the Navier-Stokes equations and is generally applied for the computation of unsteady flow fields in both compressible and incompressible flow regimes. The PISO algorithm consists of one predictor step and two corrector steps and satisfies the mass conservation using predictor-corrector steps. The sequence of operation can be listed as follows:

1. Define boundary conditions. 
2. Obtain intermediate velocity field by solving the discretized momentum equations.
3. Solve for mass fluxes at cell faces.

4. Solve pressure equation.

5. Correct for new mass fluxes at cell faces.

6. Use new pressure fields to recompute and correct velocities.

7. Update and correct boundary conditions.

8. Repeat from step 3 for a pre-defined number of times.

9. Repeat from step 1 after increasing the time step.

Similar to the SIMPLE algorithm, steps 4 and 5 are repeated to correct for mesh non-orthogonality.
Chapter 3

Results and Discussions

3.1 Revisit: Wing Designs

Section 1 presented a comprehensive description of three optimized spanloads and design approach employed. A mathematical base work was presented which justified the process of optimization used to minimize induced and total drag for the wings. The current section presents and discusses the results obtained from computational analysis of the three designs and attempts to provide a rational explanation for the many factors which may affect the performance of these wings.

Reiterating from previous sections, each of the three wings were designed with a common objective of minimizing drag under a different set of design constraints. For example, the Viscous Optimized Wing was designed to minimize the total drag, the Elliptical and Inviscid Optimized wings were designed to minimize the induced drag. The three wings entail different spanloads to achieve the minimum drag condition, which was obtained through the use of a geometric twist at the design lift coefficient. As mentioned previously, the computational domain for the wings is of a semi spherical shape, considering only the semispan and using a no-slip wall acting as a symmetry plane. The root angle of attack was taken as the reference angle of attack for all simulations. Since the twist distribution varies for the three designs, the $\alpha_{\text{root}}$ was also different for each wing case (Wroblewski and Ansell, 2016). Table. 3.1 lists the root angle of attack at the design $C_L$ for the three wings. Since the $\alpha_{\text{root}}$ is different for each design, the zero lift $\alpha$ also varies for each model.
<table>
<thead>
<tr>
<th>Spanloads</th>
<th>$\alpha_{\text{root}}$ (deg)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Elliptic Wing</td>
<td>5.76</td>
</tr>
<tr>
<td>Inviscid Optimized</td>
<td>8.54</td>
</tr>
<tr>
<td>Viscous Optimized</td>
<td>7.77</td>
</tr>
</tbody>
</table>

Table 3.1: Reference Angle of Attack at Design $C_L = 0.439$

Fig.3.1 represents computer generated models for the three designs. A discussion on geometrical parameters is omitted in this section as they have been comprehensively described in Chapter 1.
Considering the twofold objective of this study, firstly understanding CFD as a high fidelity tool for design analysis and secondly to study the performance of the optimized wings, the following sections are presented in a manner centered around these objectives. The following section presents a comparison of computational predictions for the three designs with the experimental data set from Wroblewski and Ansell (2016). Thereafter, the computational fluid dynamic predictions are used to describe the key flowfield features which produce the observed wing performance.
3.2 Data validation

Validation of fluid dynamic simulations is of key importance for studies attempting to solve complex flowfields such as in this study. Although a degree of error or rather over or under-prediction is expected in any given CFD study, validation from experimental data helps in understanding the sources of these deviations and provides a scope of improvement for both experiments and numeric code.

As discussed in Chapter 2, the current study approximates the performance variables in three steps. First, a first order pressure based scheme is employed for its stability and relatively good convergence time. The second step uses the flow field from the first step as an initial field and employs second order numeric schemes for their higher accuracy. Initialization of flowfield is crucial since second order schemes are known to be unstable and may lead to poor convergence. Although second order schemes require higher mesh resolution due to their low diffusivity, the mesh resolution from step 1 was dense enough for the second order linear schemes. Step 3 is an independent step as simulations are now run using a semi-transient solver. Time derivatives are initially required for the simulations in order to predict low order fluctuations in the flow field which eventually lead to flow transition.

![Figure 3.2: Elliptic wing: $C_L$ characteristics](image)

Figure 3.2: Elliptic wing: $C_L$ characteristics
Figs. 3.2 and 3.3 represents $C_L$ and $C_D$ characteristics for the Elliptical wing. Data is obtained from three sets of simulations, the first two using the steady state SIMPLE algorithm and the last using the semi-transient PIMPLE algorithm. As seen from Figs. 3.2 and 3.3, the degree of accuracy in prediction increases significantly as numeric scheme is changed from upwind to linear upwind. It should be noted that numeric schemes play an important part in the solution’s accuracy. However, other factors such as mesh resolution, tolerances, relaxation factors and pre-conditioners make up the list of parameters which are also crucial for a stable and feasible solution. The $C_L$ prediction is considered to be consistent with the experimental data (Wroblewski and Ansell, 2016). The lift coefficient is nearly identical to the experimental value, though minor variations may be attributed to turbulent flow field calculations in areas of high gradients and lack of cells especially in regions where cells transit from tetrahedral to triangles. The generic trend in the $C_L$ curve is also the same for all three schemes. However, the first order schemes over-predict the $C_L$ at higher angles of attack due to factors such as high diffusivity, which in turn increases the degree of inaccuracy or error. $C_L$ estimates from the steady-state simulations with second-order schemes are also similar to the predictions from the semi-transient simulations based (PIMPLE solver) on second-order schemes.

The $C_D$ estimation is greatly affected from the choice of solver and numeric schemes. As observed, the degree of over prediction decreases by more than 60% when comparing curves from first-order steady-state simulations with those from transient simulations. A
clear result of smearing of flow variables such as velocity and pressure or high diffusion in
first order schemes contributes to the high over prediction. This diffusivity of variables is
reduced vastly with the use of second-order schemes. Turbulent models used in the first
two sets of computations also assume a fully developed turbulent flow field. Hence, low
Reynolds effects such as lower skin friction at walls due to laminar boundary conditions
are neglected, and a turbulent boundary layer is assumed throughout. With the use of
$k - k_L - \omega$ turbulence model, flow transition is taken into consideration. This method
allows for skin friction computation across the laminar boundary layer region. The result
of including the laminar flow contribution is clearly seen in the $C_D$ estimates. At lower
angles of attack, the $C_D$ prediction is highly consistent with the experimental data.
A deviation from this degree of prediction is observed, however, as the $\alpha_{root}$ increases
towards both extremities on the plot.

Figure 3.4: Inviscid Optimized wing: $C_L$ characteristics
Figs. 3.4 and 3.5 represents the $C_L$ and $C_D$ predictions for the Inviscid Optimized wing. Similar to the observations made in Figs. 3.2 and 3.3, the degree of accuracy is increased with the change of solvers from first to second-order and from steady-state to transient. In Fig. 3.4, for $\alpha_{root}$ greater than 6 degrees, a sudden jump in $C_L$ prediction is observed with the use of the upwind scheme. However, $C_D$ estimates for this region tend to be stable and follow the general trend of the experimental data. It is assumed that at high angles of attack, the flow gradients significantly increase, causing the flowfield prediction using first order schemes become unreliable. In Fig. 3.5, for the second-order case, a small jump in drag prediction is observed at $\alpha_{root}$ 6 degree.
Figure 3.7: Viscous Optimized wing: $C_D$ characteristics

Figs.3.6 and 3.7 represent the $C_L$ and $C_D$ estimates for the Viscous Optimized Wing. Similar to the observations made in Figs.3.2, 3.3, 3.4 and 3.5, the general trend of data suggests an increase in the prediction accuracy with the use of second-order schemes. Fig.3.6 and Fig.3.7 indicate the presence of kinks or uneven curves at higher positive $\alpha_{\text{root}}$. Relatively smaller kinks are visible for negative $\alpha_{\text{root}}$. Though mesh inadequacies would have been a good contributing factor for such trends in data, solution logs suggest a stable simulation with pressure and velocity field residuals in the order of $10^{-6}$ and $10^{-7}$ respectively. Such lower residuals thus indicate an accurate solution.

From Figs.3.2, 3.3, 3.4 and 3.5, 3.6 and 3.7, transient simulations are shown to predict aerodynamic characteristics with the best degree of accuracy for all three wing models. Though first order schemes tend to be computationally inexpensive, the large over prediction in $C_D$ data obtained makes it an unsuitable option for design analysis. Second order schemes in steady state framework have shown to estimate values with a greater accuracy, but they fail to emulate the flow transition physics which are characteristic of flows at low Reynolds numbers. An interesting observation is made from above stated figures, especially from Fig3.7.

### 3.3 Wing Performance

The current section presents a detailed discussion on the aerodynamic performance of the three designs using computational data. Basic performance attributes are discussed along
with experimental observations from Wroblewski and Ansell (2016). Due to the extensive near and farfield data available from simulations, additional performance attributes such as the dynamics of the laminar separation bubble and its effect on lifting and drag characteristics are also presented. Figs. 3.8, 3.9, 3.10, 3.11, 3.12 and 3.13 present the aerodynamic characteristics of the three designs. Data plotted in figures below correspond to the transition modeled data set which will be used for final performance evaluation. Table 3.2 lists the drag predictions produced by the finite vortex element design code for the three designs with computational predictions of the current study.

<table>
<thead>
<tr>
<th>Spanload</th>
<th>$C_{d,i}$</th>
<th>$C_{D,p}$</th>
<th>$C_D$</th>
<th>$C_{D,comp}$</th>
<th>$\epsilon$</th>
<th>AR</th>
</tr>
</thead>
<tbody>
<tr>
<td>Elliptic</td>
<td>0.00942</td>
<td>0.00929</td>
<td>0.01871</td>
<td>0.02113</td>
<td>1</td>
<td>6.5</td>
</tr>
<tr>
<td>Inviscid Optimized</td>
<td>0.00796</td>
<td>0.00954</td>
<td>0.01750</td>
<td>0.02079</td>
<td>0.66</td>
<td>11.6</td>
</tr>
<tr>
<td>Viscous Optimized</td>
<td>0.00798</td>
<td>0.00941</td>
<td>0.01739</td>
<td>0.01974</td>
<td>0.806</td>
<td>9.5</td>
</tr>
</tbody>
</table>

Table 3.2: Drag predictions by optimal spanload configurations at design $C_L$

![Figure 3.8: Elliptic wing: $C_L$ characteristics](image-url)
Figure 3.9: Elliptic wing: $C_D$ characteristics

Figure 3.10: Inviscid Optimized wing: $C_L$ characteristics
Figure 3.11: Inviscid Optimized wing: $C_D$ characteristics

Figure 3.12: Viscous Optimized wing: $C_L$ characteristics
Figs. 3.14 and 3.15 present the performance characteristics for the three wing designs. All simulations are carried out in a manner such that a consistent Reynolds number is maintained among all three cases. From Fig. 3.14, the lift slope is observed to become steeper as the $AR$ increases. Thus, the Elliptic Wing has the shallowest lift slope amongst all three designs and the Inviscid Optimized wing is associated with the steepest slope. Due to the different twist distributions, the $\alpha_{root}$ for which $C_L = 0$ is different. While the elliptic wing has zero lift at approximately $\alpha_{root} = 0.3^\circ$, the inviscid and viscous optimized wings have zero lift at $\alpha_{root} = 3.1^\circ$ and $\alpha_{root} = 2.8^\circ$ respectively. Drag characteristics predicted computationally are observed to be in accordance with experimental measurements and predicted values (Wroblewski and Ansell, 2016). The Elliptic wing possesses the highest drag coefficient amongst all three wings and Viscous Optimized wing has the smallest total drag at the design $C_L = 0.439$. This observation is in agreement with the predicted values from Table 3.2.

Fig 3.15 indicates a lower drag for Viscous Optimized Wing when compared to Inviscid Optimized Wing at low angles of attack. At higher angles of attack on the drag polar, the viscous optimized drag is observed to have the lowest drag. This observation is currently in conflict with the experimental behavior. Though there exist a multitude of factors resulting in this deviation, the author suspects the over prediction of induced drag for the Inviscid Optimized Wing to be a crucial factor. The Elliptic Wing is observed to have the lowest drag at $C_L = 0$, followed by Viscous Optimized Wing and Inviscid Optimized
Wing.

While the computational prediction is in agreement with experimental characteristics for the design angle of attack, there exist certain instances when computational data is in conflict. One such example is the incorrect estimation of drag, especially in the case of the Inviscid Optimized Wing. A second example is the inability of the flow solver to predict flow gradients at high angles of attack. Another factor which the author believes may be a crucial contributing factor is the presence of wall interference effects and corresponding corrections that were made during the experimental study. Based on available literature (Wroblewski and Ansell, 2016), such factors were not included as wing design constraints.

Figure 3.14: $C_L$ characteristics of wing designs

Figure 3.15: $C_D$ characteristics of wing designs
3.4 Surface visualization

This section emphasizes the flow transition physics at $Re_z = 450,000$ for all three wing designs. The transitional behavior of the boundary layer is visualized using surface line integration convolution of the wall shear stress variable (Cabral and Leedom, 1993; Stalling and Hege, 1995; Interrante, 1997). Surface flow features such as point of attachment, stagnation, transition and separation are easily distinguishable in such methods with use of saddle points, nodes and with an overlay of field variable contour plot which present the corresponding color spectrum to the plot.

Figure 3.16: Surface flow visualization for the Elliptic Wing
Figs. 3.16, 3.17 and 3.18 illustrate the shear stress on the wing surfaces for the three designs. As seen from the figures above, a series of saddle points indicate the presence of a laminar separation bubble across the wing span. The shown visualizations are for wings at the design condition of $C_L = 0.439$. The pattern of transition region along the span is different for all three design. Such differences can be attributed to the different twist distribution for each design. At span stations with higher local $\alpha$, the pressure and shear gradients are higher, causing an early onset of transition as compared to the outboard sections of the wings due to washout. The trends in transition physics along the span shown above thus correspond to the wing twist distribution and induced angle of attack effects. Based on the visualizations, the Inviscid Optimized Wing has the largest laminar flow region with the smallest maximum shear stress magnitude at design $C_L$, thus resulting in a lower drag. This observation is currently in agreement with the predicted design values. Also, no considerable magnitude of span-wise flow is observed for any of the designs, though a slight influence of the tip vortex is observed for all three wings.
Qualitatively, the Elliptic Wing is shown to have the largest influence from wing tip vortex.

Figs. 3.19, 3.20 and 3.21 present $C_p$ distributions for three different span stations, directed from the inboard to outboard regions. The above plots provide an additional understanding of the influence of the laminar separation bubble along the span. For example, in Fig. 3.20, laminar separation bubble is observed as small pressure plateau on the suction side. At 10% span, the separation bubble is observed at a location 18% chord from the leading edge. At midspan, the separation bubble moves back to 25% chord and further towards the trailing edge at 40% chord in the outboard span station. Similar trends are observed in other two wing designs. The $C_p$ distributions are also indicative of the span wise lift distribution. A general trend suggests a decrease in sectional $C_l$ with further distance in the outboard direction. This may be attributed to the increase in washout towards the outboard sections.

![Figure 3.19: Spanwise $C_P$ distribution for the Elliptic Wing](image)
3.5 Wake Analysis

A Wake analysis was conducted for all three wing designs at the design $C_L = 0.439$ and $Re_{\tau} = 450,000$. Wake surveys were performed at 4 different stations downstream of the wing using computational data slicing tools. The survey stations were limited to $0.6c$ due to the increase in computational costs since wake studies require high resolution baffles across the areas of interest. Velocity and vorticity are computed to determine the momentum deficit and shed circulation across the wake which ultimately affect the lift and drag characteristics of the wing.

Figs.3.22, 3.23 and 3.24 show velocity iso-contours for the three wings at design $C_L$. All
plots are indicative of velocity fields in the near surface region, thus indicating a velocity jump in the suction region and stagnation at leading edge. Trailing edge recirculation regions are not visible due to the thickness of contour plots. The iso-contours suggest a relatively larger wing tip vortex for the Elliptic Wing and significantly smaller vortices for the Inviscid Optimized and Viscous Optimized Wings.

Figure 3.22: Iso-surface contours for the Elliptic Wing
Figure 3.23: Iso-surface contours for the Inviscid Optimized Wing

Figure 3.24: Iso-surface contours for the Viscous Optimized Wing
Figure 3.25: Velocity Contour plot at 5 % c for the Elliptic Wing

Figure 3.26: Velocity Contour plot at 5 % c for the Inviscid Optimized Wing

Figure 3.27: Velocity Contour plot at 5 % c for the Viscous Optimized Wing
Figure 3.28: Velocity Contour plot at 20% \( \tau \) for the Elliptic Wing

Figure 3.29: Velocity Contour plot at 20% \( \tau \) for the Inviscid Optimized Wing

Figure 3.30: Velocity Contour plot at 20% \( \tau \) for the Viscous Optimized Wing
Figure 3.31: Velocity Contour plot at 40 % $\tau$ for the Elliptic Wing

Figure 3.32: Velocity Contour plot at 40 % $\tau$ for the Inviscid Optimized Wing

Figure 3.33: Velocity Contour plot at 40 % $\tau$ for the Viscous Optimized Wing
Figure 3.34: Velocity Contour plot at 60% \( c \) for the Elliptic Wing

Figure 3.35: Velocity Contour plot at 60% \( c \) for the Inviscid Optimized Wing

Figure 3.36: Velocity Contour plot at 60% \( c \) for the Viscous Optimized Wing
Figure 3.37: Vorticity Contour plot at 5 % $\tau$ for the Elliptic Wing

Figure 3.38: Vorticity Contour plot at 5 % $\tau$ for the Inviscid Optimized Wing

Figure 3.39: Vorticity Contour plot at 5 % $\tau$ for the Viscous Optimized Wing
Figure 3.40: Vorticity Contour plot at 20 % $\tau$ for the Elliptic Wing

Figure 3.41: Vorticity Contour plot at 20 % $\tau$ for the Inviscid Optimized Wing

Figure 3.42: Vorticity Contour plot at 20 % $\tau$ for the Viscous Optimized Wing
Figure 3.43: Vorticity Contour plot at 40 % τ for the Elliptic Wing

Figure 3.44: Vorticity Contour plot at 40 % τ for the Inviscid Optimized Wing

Figure 3.45: Vorticity Contour plot at 40 % τ for the Viscous Optimized Wing
The wake analysis velocity and vorticity contour plots are presented in Figs.3.25,
3.37, 3.26, 3.38, 3.27 and 3.39, among others shown above. A general trend observed is the upward deflection of wake produced by the Inviscid and Viscous Optimized wings. The Elliptical Wing in this case, produces a relatively planar wake. These deflection characteristics of the Inviscid and Viscous Optimized wings are due to larger span-wise twist distributions. Based on the velocity contours, the velocity deficit can be seen with a minimum velocity zone at the center of the wake and a transition to the freestream velocity across the wake from the suction side. With the increase in downstream station, these characteristic features diminish. In addition to these observations, the Inviscid Optimized Wing produces the wake with the thinnest cross section, while the Elliptic Wing produces the wake with the thickest cross section.

The vorticity contours suggest the Elliptic Wing produces the largest tip vortex among all three wings. This observation is in accordance with the iso-contour plots. The Inviscid and Viscous Optimized wings have significantly smaller tip vortices. This may be attributed to the different spanload designs for each wing, especially local to the wing tip. The Elliptic Wing has a relatively higher gradient or decrease in bound circulation as compared to the other two wings. This results in a larger tip vortex since the decrease in bound circulation is fed to the strength of the vortex sheet. This increased strength in the vortex sheet across this region results in much greater vortex roll up at wing tip. The Experimental study by Wroblewski and Ansell (2016) also suggests a smaller tip vortex for the Inviscid Optimized design when compared to Viscous Optimized design. For reasons unknown, computational results fail to capture any tip vortex features for these designs. Vorticity contours plotted at the station closest to the trailing edge suggest different vortex shedding characteristics for the three wings. For the Elliptic Wing, inboard sections produce greater vorticity than the outboard sections. The Inviscid Optimized Wing produces maximum vorticity from regions between 75 \% to 90 \% span, with similar trends observed for the Viscous Optimized Wing.

3.6 Solution Convergence

Credibility of any computational study depends on the simulation’s stability and accuracy. As mentioned in Chapter 2, factors such as mesh quality, turbulence model, flow solvers, relaxation factors and numeric schemes play a crucial role in determining stability
and accuracy. Re-iterating, the current study entails flow simulations at a low Reynolds number of 450,000. This results in the selection of a certain flow solver type and turbulence model. In order to capture flow physics accurately, a flow transition model was chosen. Hence the flow solver solves for turbulence flow variables $k$, $\nu Tilda$ and $omega$, in addition to flow field variables $U$ and $p$. Each flow variable is solved for a certain number of iterations with inner loops until the residual, the difference in predicted flow field variables between two time steps, is below a certain acceptable limit. This limit corresponds to the accuracy of the solution and the ease with which the solution attains this accuracy refers to the solution stability. For a well poised simulation, each flow variable must meet this criterion to deliver feasible results.

Figs.3.49, 3.50 and 3.51 show the convergence of the solution for the three wing cases at design $C_L$. Plots indicate a quick convergence of turbulence and velocity field variables. Pressure however, converges at a later stage. Fig.3.49 shows a better convergence for the Elliptic Wing as compared to other wing cases. The Inviscid Optimized Case converges relatively slower than other two cases. This delay in convergence is primarily because of the higher mesh resolution required due to the geometric properties. Simulations are run for about 3000 iterations to ensure no oscillatory behavior is observed.
Figure 3.49: Final Residuals for Elliptic Wing Simulation variables

Figure 3.50: Final Residuals for Inviscid Optimized Wing Simulation variables

Figure 3.51: Final Residuals for Viscous Optimized Wing Simulation variables
Chapter 4

Conclusion

4.1 Summary

This study employed computational techniques to analyze and understand the performance characteristics of three span optimized wings at $Re_{\tau} = 450,000$. An open-source flow solver tool, OpenFOAM, was used. Access to source code allowed for a thorough understanding of solver algorithm and the mathematical framework upon which the turbulence models were based. This feature was found crucial in understating the flow transition turbulence model $k - k_l - \omega$, which has evolved over the past 5 years. Though the initial intent in this study was to use a structured open-source mesh generation tool, certain factors such as lack of support documentation, literature and inability to add boundary-layer cells to resolve flow gradients at the wall lead to the use of a high fidelity third party meshing tool called Pointwise. Based on a geometry feature approach in meshing rather than the cell snapping routine, this process was found to be 70 % less computationally expensive than the open-source mesh generator. In addition to this, the most distinctive advantage of using Pointwise was the ability to add adequate cells layers (approx. 100 layers) in the boundary-layer region. Although a structured domain would have been ideal and computationally inexpensive, complexity in mesh generation and available time frame lead to the generation of unstructured computational domains. A semi-spherical domain was used instead of the conventional wind tunnel rectangular-box type due to ease of mesh generation and to avoid the usage of unconventional boundary conditions at the wall at high angles of attack. A comparative study was conducted for
these two domain types and no differences were observed in the simulations. The choice of turbulence models for this study was based on available literature and an initial set of simulations for one wing design case, comparing results from a number of turbulence models. Apart from the transition turbulence model, simulations were carried out in the linear regime for Spalart-AllMaras, $k-\epsilon$ and $k-\omega-SST$ two equation models. Based on the flow regime and available mesh quality, $k-\omega-SST$ yielded the best prediction, but it was found to be relatively computationally expensive when compared to its one equation counter part, Spalart-AllMaras model. Simulations were carried out in a three-phase process, starting from steady state pressure based SIMPLE solver using first order upwind schemes, followed by the use of second order linear-upwind schemes and transient solver, PIMPLE, with similar numeric settings. Simulations were run in the linear lift regime for the wing geometries due to two main reasons. First, the design space for the three wing designs was in low-moderate angles of attack. Second, with the available time frame and computational resources, higher-order simulations more suitable to model onset of stall, such as Large Eddy Simulation were not possible.

The performance characteristics for the three wing designs were studied using the computational techniques summarized above. Each wing was optimized using method of Lagrange Multiplier with a different cost function, $J$. This resulted in three spanload designs, each achieved with a specified twist distribution specific to the wing geometry. The Elliptic Wing was considered as a baseline case with an elliptic spanload, thus minimizing induced drag. A surrogate variable for structural weight consideration was added to the cost function which resulted in the Inviscid Optimized design. With the largest AR, the obtained spanload was designed to minimize induced drag. The Viscous Optimized Wing was designed with an objective to minimize the total overall drag. A profile drag term was added to the cost function for this design. Since each wing possessed different geometrical attributes, different meshing routines were used for each of them. For these wing geometries, performance analysis indicated lowest drag for the Viscous Optimized Wing at design $C_L = 0.439$ and the highest drag for the Elliptic Wing. However, for $C_L = 0$, the Elliptic Wing had the least drag. At higher angles of attack, the Viscous Optimized Wing was observed to have the smallest drag. This was found to be in conflict with the experimental results. A relatively higher degree of inaccuracy in drag prediction was attributed to this observation. A characterization of the wing drag into profile and induced drag
contributions was not made in this study. Surface flow visualization techniques were used to study the formation and location of a laminar separation bubble. Due to different twist distributions for the three designs, each wing was observed to have a different pattern for onset of turbulence along the span. No significant span-wise flow was observed, though a small influence from the tip vortex was present for the Elliptic Wing. Qualitative wake analysis suggested a substantially larger tip vortex for the Elliptic Wing as compared to the other two designs. This was explained based on the aggressive bound circulation spanwise gradient at the wing tip for the Elliptic Wing. A non planar wake was observed for the Inviscid Optimized and Viscous Optimized wings, which was deflected upwards due to the aggressive washout at the tips. The Elliptic Wing was observed to produce a planar wake. Velocity contours also indicated a maximum wake thickness for the Elliptic Wing and a minimum for the Inviscid Optimized Wing. Vorticity contours at different stations downstream illustrated the span-wise contributions to vorticity for each wing, indicating greater vorticity distributions from the inboard sections of the Elliptic wing. The performance characteristics are summarized in Tables 4.1, 4.2 and 4.3.
### Computational Data: $k - k_L - \omega$ model

<table>
<thead>
<tr>
<th>$\alpha_{root}$</th>
<th>$C_L$</th>
<th>$C_D$</th>
<th>$L/D$</th>
</tr>
</thead>
<tbody>
<tr>
<td>-15</td>
<td>-0.8949</td>
<td>0.1260</td>
<td>-11.0383</td>
</tr>
<tr>
<td>-13</td>
<td>-0.8205</td>
<td>0.0644</td>
<td>-12.7022</td>
</tr>
<tr>
<td>-11</td>
<td>-0.7319</td>
<td>0.0493</td>
<td>-14.8380</td>
</tr>
<tr>
<td>-9</td>
<td>-0.6243</td>
<td>0.0360</td>
<td>-17.3244</td>
</tr>
<tr>
<td>-7</td>
<td>-0.5015</td>
<td>0.0251</td>
<td>-19.9166</td>
</tr>
<tr>
<td>-5</td>
<td>-0.3668</td>
<td>0.0173</td>
<td>-21.1547</td>
</tr>
<tr>
<td>-3</td>
<td>-0.2226</td>
<td>0.0123</td>
<td>-18.0086</td>
</tr>
<tr>
<td>-1</td>
<td>-0.0979</td>
<td>0.0101</td>
<td>-9.6729</td>
</tr>
<tr>
<td>1</td>
<td>0.0658</td>
<td>0.0105</td>
<td>6.2630</td>
</tr>
<tr>
<td>3</td>
<td>0.2296</td>
<td>0.0135</td>
<td>16.9142</td>
</tr>
<tr>
<td>5</td>
<td>0.3913</td>
<td>0.0192</td>
<td>20.3465</td>
</tr>
<tr>
<td>5.5</td>
<td>0.4295</td>
<td>0.0211</td>
<td>20.3150</td>
</tr>
<tr>
<td>5.75</td>
<td>0.4482</td>
<td>0.0221</td>
<td>20.2234</td>
</tr>
<tr>
<td>7</td>
<td>0.5415</td>
<td>0.0279</td>
<td>19.3571</td>
</tr>
<tr>
<td>9</td>
<td>0.6844</td>
<td>0.0393</td>
<td>17.4113</td>
</tr>
<tr>
<td>11</td>
<td>0.8168</td>
<td>0.0528</td>
<td>15.4473</td>
</tr>
<tr>
<td>13</td>
<td>0.9325</td>
<td>0.0682</td>
<td>13.6708</td>
</tr>
<tr>
<td>15</td>
<td>1.0261</td>
<td>0.08442</td>
<td>12.1634</td>
</tr>
</tbody>
</table>

Table 4.1: Summary of Lift and Drag characteristics for Elliptic Wing
<table>
<thead>
<tr>
<th>$\alpha_{\text{root}}$</th>
<th>$C_L$</th>
<th>$C_D$</th>
<th>$L/D$</th>
</tr>
</thead>
<tbody>
<tr>
<td>-12.2</td>
<td>-0.9251</td>
<td>0.0962</td>
<td>-9.6132</td>
</tr>
<tr>
<td>-10.2</td>
<td>-0.8900</td>
<td>0.0713</td>
<td>-12.4667</td>
</tr>
<tr>
<td>-8.2</td>
<td>-0.8139</td>
<td>0.0525</td>
<td>-15.4785</td>
</tr>
<tr>
<td>-6.2</td>
<td>-0.7107</td>
<td>0.0407</td>
<td>-17.4337</td>
</tr>
<tr>
<td>-4.2</td>
<td>-0.5847</td>
<td>0.0302</td>
<td>-19.3101</td>
</tr>
<tr>
<td>-2.2</td>
<td>-0.4400</td>
<td>0.0225</td>
<td>-19.5149</td>
</tr>
<tr>
<td>-0.2</td>
<td>-0.2823</td>
<td>0.0170</td>
<td>-16.6002</td>
</tr>
<tr>
<td>1.8</td>
<td>-0.1141</td>
<td>0.0139</td>
<td>-8.1969</td>
</tr>
<tr>
<td>3.8</td>
<td>0.0596</td>
<td>0.0131</td>
<td>4.5314</td>
</tr>
<tr>
<td>5.8</td>
<td>0.2355</td>
<td>0.0149</td>
<td>15.7139</td>
</tr>
<tr>
<td>7.3</td>
<td>0.407</td>
<td>0.0207</td>
<td>19.6135</td>
</tr>
<tr>
<td>8.3</td>
<td>0.4430</td>
<td>0.0261</td>
<td>17.1994</td>
</tr>
<tr>
<td>9.8</td>
<td>0.5731</td>
<td>0.0349</td>
<td>16.4151</td>
</tr>
<tr>
<td>11.8</td>
<td>0.7270</td>
<td>0.0458</td>
<td>15.8680</td>
</tr>
<tr>
<td>13.8</td>
<td>0.8677</td>
<td>0.0586</td>
<td>14.7886</td>
</tr>
<tr>
<td>15.8</td>
<td>0.9864</td>
<td>0.063</td>
<td>15.6576</td>
</tr>
</tbody>
</table>

Table 4.2: Summary of Lift and Drag characteristics for Inviscid Optimized Wing
### Table 4.3: Summary of Lift and Drag characteristics for Viscous Optimized Wing

<table>
<thead>
<tr>
<th>(\alpha_{root})</th>
<th>(C_L)</th>
<th>(C_D)</th>
<th>(L/D)</th>
</tr>
</thead>
<tbody>
<tr>
<td>-12.77</td>
<td>-0.9559</td>
<td>0.1029</td>
<td>-9.2836</td>
</tr>
<tr>
<td>-10.77</td>
<td>-0.9017</td>
<td>0.0713</td>
<td>-12.6382</td>
</tr>
<tr>
<td>-8.77</td>
<td>-0.859</td>
<td>0.0536</td>
<td>-16.0188</td>
</tr>
<tr>
<td>-6.77</td>
<td>-0.7767</td>
<td>0.0409</td>
<td>-18.9413</td>
</tr>
<tr>
<td>-4.77</td>
<td>-0.654</td>
<td>0.0300</td>
<td>-21.7933</td>
</tr>
<tr>
<td>-2.77</td>
<td>-0.5052</td>
<td>0.0214</td>
<td>-23.5677</td>
</tr>
<tr>
<td>-0.77</td>
<td>-0.3350</td>
<td>0.0156</td>
<td>-21.4493</td>
</tr>
<tr>
<td>1.23</td>
<td>-0.1523</td>
<td>0.0126</td>
<td>-12.0868</td>
</tr>
<tr>
<td>3.23</td>
<td>0.0371</td>
<td>0.0122</td>
<td>3.0212</td>
</tr>
<tr>
<td>5.23</td>
<td>0.2289</td>
<td>0.0147</td>
<td>15.5719</td>
</tr>
<tr>
<td>7.23</td>
<td>0.4305</td>
<td>0.0197</td>
<td>21.8100</td>
</tr>
<tr>
<td>8</td>
<td>0.4871</td>
<td>0.0213</td>
<td>22.7760</td>
</tr>
<tr>
<td>9.23</td>
<td>0.6035</td>
<td>0.0271</td>
<td>22.2333</td>
</tr>
<tr>
<td>11.23</td>
<td>0.741</td>
<td>0.0368</td>
<td>20.1251</td>
</tr>
<tr>
<td>13.23</td>
<td>0.9154</td>
<td>0.0375</td>
<td>24.3732</td>
</tr>
<tr>
<td>15.23</td>
<td>0.9982</td>
<td>0.0632</td>
<td>15.7750</td>
</tr>
</tbody>
</table>

### 4.2 Conclusions

The following conclusions were made from this study:

1. Wing performance characteristics are effectively predicted using both steady state and transient flow solvers. Lift characteristics are predicted accurately in the linear regime while drag is over predicted by a certain extent. Higher deviations are observed in the near-stall regime.

2. Both RANS and URANS flow solvers are employed for analysis. Given the design envelope for all three wings, no significant differences are observed between steady-state and transient results, except the prediction of flow transition by the transient solver. In other words, both near and farfield flow characteristics are similar with the two solvers.

3. Tetrahedral extruded cells at the boundary layer allow for multiple layers to be
added, thus allowing for a mesh with $y^+ = 1$. This in turn allows for a more accurate prediction of the laminar separation bubble and flow separation points.

4. Decisions on turbulence models and numerical discretization schemes were made based on available computing resources and time window. A One-equation model, such as SA, present less accurate results than the two equation $k-\omega-SST$ model, but achieve convergence in half the number of iterations. Second-order upwind schemes are highly unstable without initialization. Hence data from first order schemes are used to initialize simulations.

5. Computational data indicates a minimum drag for Viscous Optimized Wing at the design $C_L$. The Elliptic Wing on the other hand has minimum drag at negative $\alpha_{\text{root}}$ and at $C_L = 0$. Inviscid Optimized Wing, even though with the highest AR, still produces lesser drag than Elliptic Wing.

6. Surface flow visualization provided reasonable flow transition features for the three designs. Different spanwise flow transition patterns are observed for each case. While the Elliptic Wing has a relatively straight transition line, the Inviscid and Viscous Optimized Wings were characterized by a skewed or uneven transition line along the span. This observation is attributed to the twist distribution specific to each model.

7. Qualitative wake analysis at stations downstream to the wings indicate different vortex shedding features for each wing. A strong tip vortex is observed in the iso-contour plot for the Elliptic Wing. Relatively smaller tip vortex systems are observed for the other two wings. In addition to this feature, each wing possesses a different wake thickness at the same station. The thickest wake cross section is observed for the Elliptic Wing. Spanwise contributions to vorticity suggests a higher vortex shedding contribution from the inboard sections of the Elliptic wing.

### 4.3 Recommendations

The following recommendations are made in order to expand the current study.

1. Simulations based on RANS and URANS based solves provide time averaged turbulent flow fields. Higher fidelity computations such as those based on Large Eddy
Simulations can provide additional characterization of turbulent flow field at the design $C_L$.

2. A quantitative analysis of the wake survey will allow for estimating additional near-field flow features such as induced velocity both upstream near the leading edge and downstream from the trailing edge, bound circulation and evolution of wake trace at stations further downstream.

3. For multiple angles of attack within the design space, the dynamics of the laminar separation bubble can be studied. Features such as separation bubble size, length and position along the chord can be instrumental in understanding the wing performance at low Reynolds number regime.

4. The current study employs only one type of low Reynolds number turbulence model. A comprehensive study describing the use of other low Reynolds number turbulence models, such as Launder-Sharma $k - \epsilon$, can facilitate more accurate low Reynolds number flow fields.

5. A parametric study for different second and higher order numeric schemes is recommended for better understanding the numeric setup, allowing faster execution of runs.

6. Designs are analyzed in the linear regime. Computation of near-stall and post-stall characteristics using more advanced CFD techniques such as Large Eddy Simulation will be effective.

7. Wing models are designed using an Inviscid Optimization routine. A multidisciplinary approach with viscous considerations to determine an optimum twist distribution can be used to provide a step forward from this study.
Bibliography


O. Lilienthal. *Der Vogelflug als Grundlage der Fliegekunst (Bird flight as the basis of Aviation)*. London, UK, 1889.


M. Munk. The minimum induced drag of aerofoils. NACA Rep, 121, 1923.


Jones. The spanwise distribution of lift for minimum induced drag of wings having a given lift and a given bending moment. NACA TN, 2249, 1950.


Evren Bayraktar, Otto Mierka, and Stefan Turek. Benchmark computations of three
dimensional laminar flow around a cylinder with openfoam. *International Journal of

Timothy J. Barth. Aspects of unstructured grids and finite-volume solvers for the euler

Rainald Loehner. Matching semi-structured and unstructured grids for navier-stokes

J. Frohlich, W. Rodi, Ph. Kessler, S. Parpais, J.P. Bertoglio, and D. Laurence. Large
eddy simulation of flow around circular cylinders on structured and unstructured grids.

F. Moukalled, L. Mangani, and M. Darwish. Gradient computation. *Fluid Mechanics

F.S. Lien, W.L. Chein, and M.A. Leschziner. A multiblock implementation of a non-

K.J. Badcock, B.E. Richards, and M.A. Woodgate. Elements of computational fluid
dynamics on block structured grids using implicit solvers. *Progress in Aerospace Studies*,

Jim Ruppert. A new and simple algorithm for quality two dimensional mesh generation.

Jonathan Richard Shewchuk. Delaunay refinement algorithms for triangular mesh gen-

title = Pointwise.


Joe F. Thompson. A survey of dynamically-adaptive grids in the numerical solution of


