DEVELOPMENT OF A PRESSURE BASED INCOMPRESSIBLE FLOW SOLVER FOR WIND TURBINE APPLICATIONS

Akshay Koodly Ravishankara

Development of a pressure based incompressible flow solver for wind turbine applications

Akshay Koodly Ravishankara

Development of a pressure based incompressible flow solver for wind turbine applications

DISSERTATION

to obtain
the degree of doctor at the Universiteit Twente,
on the authority of the rector magnificus,
prof.dr.ir. A Veldkamp,
on account of the decision of the Doctorate Board,
to be publicly defended
on Wednesday 6th October 2021 at 14.45 hours

by

Akshay Koodly Ravishankara

born on the 31st October 1992
in Bangalore, Karnataka, India

This dissertation was approved by:

Supervisor
prof.dr.ir. C.H. Venner

Co-supervisor
dr.ir. E.T.A. van der Weide
dr. Hüseyin Özdemir

Graduation Committee:

Chair/Secretary:	prof.dr.ir H. F. J. M. Koopman
Supervisor:	prof.dr.ir C. H. Venner
Co-supervisor:	dr.ir E. T. A. van der Weide dr. Hüseyin Özdemir
Committee Members:	prof.dr.ir. B.J. Geurts prof.dr. A.R. Thornton dr. V. Magnanimo prof.dr.ir. C.J. Simão Ferreira Prof. Dr. S. Voutsinas dr. R.J.A.M. Stevens

Dedicated to my grandmothers N. Shantalakshmi and H.S. Narasamma.

Contents

Summary

Wind turbine aerodynamics spans a wide range of scales starting from the millimeter thick boundary layers up to atmospheric flows. Analysis of each of these scales have traditionally been performed using simplified or 'engineering' models. However, with simplifications also come uncertainties. In order to reduce the uncertainties and also to improve the design process, more accurate analysis tools are necessary. Computational Fluid Dynamics (CFD) based methods are being used to fill this need. This thesis presents the development and application of an incompressible pressure based solver. The new solver has been developed within the framework of SU2, an open source collection of C++ based software tools for multi-physics analysis.

Wind turbines operate under "incompressible" conditions and the CFD methods are based on solving the Navier-Stokes equations in the incompressible form. The main difficulty in solving the incompressible Navier-Stokes equations lies in resolving the pressure velocity coupling. Various approaches have been proposed over the years to overcome this difficulty and in this work the SIMPLE algorithm is used. The common link among all the approaches to resolve the pressure velocity coupling was the use of staggered grids, where pressure and velocities are stored at cell centers and cell faces respectively. The staggered grid comes with its own disadvantages, especially for modern day industrial problems. A momentum based interpolation technique is used to allow for the SIMPLE algorithm to be applied on collocated grids. The effect of turbulence is modeled using the eddy viscosity based Spalart-Allmaras (SA) and the k-ω Shear Stress Transport (SST) models. Verification of the new solver is carried out using Couette flow problems where analytical solutions to the Navier-Stokes equations can be found. Validation of the new solver is carried out using widely studied problems like flow over a flat plate, backward facing steps and cylinder.

CFD is used for a wide variety of problems in the wind turbine industry. Three such problems are studied in this work. First, the effect of vortex generators on the boundary layer is presented. Integral boundary layer (IBL) based methods are used extensively for quick and accurate analysis of airfoil performance. The effect of vortex generators on the boundary layer is analyzed using the mixing layer theory as a first step towards modeling vortex generators in IBL methods. Another issue gaining importance recently is the effect of leading edge erosion. As the blade surface is exposed to the elements continuously, it is prone to erosion which leads to a reduction in performance. A roughness model is implemented and used to analyze the effect of erosion on aerodynamic performance and boundary layer behavior. Finally, the new solver is used to study the flow past a rotating wind turbine blade.

Samenvatting

De aerodynamica van windturbines omvat een brede diversiteit aan schalen, van de millimeter dikke grenslagen tot atmosferische stromingen. Analyse van elk van deze schalen werd traditioneel grotendeels uitgevoerd met behulp van vereenvoudigde of 'engineering' modellen. Echter, vereenvoudigingen introduceren ook onzekerheden. Om deze onzekerheden te verminderen en ook om het ontwerpproces te verbeteren, zijn nauwkeurigere analyse-methoden nodig. Om in deze behoefte te voorzien, worden methoden gebruikt, die gebaseerd zijn op Computational Fluid Dynamics (CFD). Dit proefschrift presenteert de ontwikkeling en toepassing van een onsamendrukbare, pressure based solver. De nieuwe solver is ontwikkeld binnen het raamwerk van SU2, een open source verzameling van op C++ gebaseerde softwaretools voor multi-disciplinaire analyse.

Windturbines werken onder "onsamendrukbare" omstandigheden en de CFD methoden zijn gebaseerd op het oplossen van de Navier-Stokes vergelijkingen in de onsamendrukbare vorm. De grootste moeilijkheid bij het oplossen van de onsamendrukbare Navier-Stokes-vergelijkingen ligt in het oplossen van de koppeling tussen de druk en snelheden. In de loop der jaren zijn er verschillende benaderingen voorgesteld om deze moeilijkheid te overwinnen en in dit werk wordt het SIMPLE-algoritme gebruikt. De overeenkomst tussen alle benaderingen voor het oplossen van de koppeling tussen druk en snelheden was het gebruik van staggered roosters, waar druk en snelheden worden opgeslagen in respectievelijk het midden en op de randen van de cel. Het staggered rooster heeft zijn eigen nadelen, vooral voor moderne industriële problemen. Een op impuls gebaseerde interpolatietechniek wordt gebruikt om het SIMPLE-algoritme toe te passen op collocated grids. Het effect van turbulentie is gemodelleerd met behulp van de eddy-viscositeit modellen van Spalart-Allmaras (SA) en k-ω Shear Stress Transport (SST). Verificatie van de nieuwe solver wordt uitgevoerd met behulp van Couette-stromingsproblemen waarvoor analytische oplossingen van de Navier-Stokes vergelijkingen gevonden kunnen worden. Validatie van de nieuwe solver wordt uitgevoerd met behulp van veel bestudeerde problemen, zoals de stroming over een vlakke plaat, backward facing step en cilinder.

CFD wordt gebruikt voor een breed scala aan problemen in de windturbine-industrie. In dit werk worden drie van deze problemen bestudeerd. Ten eerste wordt het effect van wervel-generatoren op de grenslaag gepresenteerd. Grenslaag Integraal Methoden (Engelse afkorting IBL) worden veelvuldig gebruikt voor snelle en nauwkeurige analyse van de prestaties van het vleugelprofiel. Het effect van wervel-generatoren op de grenslaag wordt geanalyseerd met behulp van de mixing layer theorie als een eerste stap naar het modelleren van wervel-generatoren in grenslaag-integraalmethoden. Een ander probleem dat recentelijk steeds belangrijker wordt, is het effect van erosie aan de voorkant van het blad. Omdat het bladoppervlak

continu wordt blootgesteld aan de elementen, is het gevoelig voor erosie, wat leidt
tot een vermindering van de prestaties. Een ruwheidsmodel is geïmplementeerd en
gebruikt om het effect van erosie op de aerodynamische prestaties en het gedrag
van de grenslaag te analyseren. Ten slotte wordt de nieuwe solver gebruikt om de
stroming over een draaiend windturbineblad te bestuderen.

Acknowledgement

1

Introduction

Humanity has long harnessed the power of wind for various applications ranging from simple sailing boats to grain mills and water pumps. Some researchers estimate wind energy has been in use for over 2000 years [1]. In the present day, wind energy is poised to play an important role in moving away from fossil fuel based sources of energy.

1.1. Historical overview

Beurskens [2] broadly classifies four different periods in which the use of wind energy evolved into its current form.

1. 600–1890 (classical period) - Classic windmills mainly used as mechanical drives to operate grain mills and other applications. More than $100,000$ windmills were constructed in northwestern Europe. This period ended after the discovery of the steam engine and because of the ready availability of wood and coal.

2. 1890–1930 - Development of the first electricity generating wind turbines. The development of electricity as a source of energy available to everyone led to the use of windmills as an additional possibility for generating electricity. This period saw some basic developments in the field of aerodynamics and control. This period ended as cheaper fossil fuels like oil became more readily available to generate electricity.

3. 1930–1973 - First phase of modern innovation. The necessity of electrifying rural areas and the shortage of energy during the Second World War stimulated new developments. This period saw major advances in the field of aerodynamics.

4. From 1973 to present day - Second phase of innovation and mass production of wind turbines. The energy crisis and environmental problems in combination with technological advances ensure a commercial breakthrough.

The classical wind turbines converted the kinetic energy of wind to mechanical energy directly and were likely vertical. The blades consisted of sailcloth and no control mechanisms were present. Yaw mechanisms were the first piece of technology adopted to align the turbines with the wind direction. Textile strips attached on the wooden blades and the pressure difference across the two sides of the blade kept the strips in place giving an aerodynamic profile. John Smeaton was one of the earliest pioneers in experimenting with aerodynamic efficiency of wind turbines. He introduced the twist in blades (which he termed as "weather"). More improvements to the technology steadily improved the performance of wind turbines but they were superseded by steam power after the invention of steam engines.

In the late 1800s, the rise in popularity of the dynamo reignited interest in using wind power to generate electricity and the earliest wind turbines generated a few kW of power. Development of wind turbines continued in Europe spurred largely by the unpredictability of oil prices during the early 20th century. Denmark, US and Germany were the main hubs of innovations in wind energy in this period. The Smidth Company of Denmark introduced two blade turbines capable of producing 50kW and 70kW. A three bladed 200kW machine was installed in Gedser around the year 1957 (see for example figure 1.1). After the second world war, wind power saw gradual improvement in other areas spurred largely by increasing demand for electricity and reluctance to rely on fossil fuels for all power. The aerodynamic knowledge gained in the aerospace sector also played a key role. However, the progress started to slow as fossil fuels became cheaper and nuclear power emerged as a viable source of energy. And just as steam power had sidelined wind power almost a century earlier, nuclear energy and fossil fuels seemed to do the same until the oil crisis of 1973.

A new energy policy was pursued after the oil crisis to address the over dependence on oil and also to account for the fact that fossil fuels could be exhausted if over used. Environmental problems with the use of fossil fuels and the danger of nuclear energy came to light almost a decade later. Once again, renewable sources of energy like wind, solar and geothermal were actively pursued. It was quickly realized that the only way for wind power to be economical was with large multi megawatt turbines. This phase of development saw large leaps in research in wind turbine design and associated fields like grid connections, wake flows, structures, installation among others. Offshore wind energy was identified as the best candidate for large wind farms. The first offshore farm was erected 2.5 km from the coast of Vindeby in Denmark in 1991 with a total capacity of 4.95 MW comprising 11 450kW turbines. More information about the history of wind turbines can be found in History of Wind by Buerskens [2].

1.2. Wind energy in the current energy climate

While a period of steady but unspectacular growth in wind energy was observed in the early 1990s, in recent times it has grown spectacularly (figure 1.3). Wind energy is expected to play a crucial role in transitioning towards a zero carbon energy sector. Constant improvements in wind turbine design has led to larger and larger wind turbines (figure 1.4). Larger wind turbines (see figure 1.2) have driven installation

(a) A post wind mill in the Netherlands.

(b) Smidth-Aeromotor in Denmark.

Figure 1.1: Wind mills used for mechanical drives (a) and an out of service Smidth-Aeromotor in Denmark with a nominal output of $70kW$ (b) [2].

(a) The $25m$ HAT wind turbine in operation at Petten ($0.4MW$ rated capacity) [2].

(b) The $12MW$ GE Haliade X at the port of Rotterdam [3].

Figure 1.2: Evolution of wind turbines, from producing $< 1MW$ in 1980s (a) to large wind turbines producing $> 10MW$ today (b).

costs per megawatt (MW) down which is reflected in the massive cost reduction for both onshore and offshore wind turbines. The global weighted-average levelized cost of energy (LCOE) of projects using this technology and commissioned in 2019 was USD 0.053/kWh — 9% lower than in 2018 and 39% lower than in 2010, when it was USD 0.086/kWh. Onshore wind energy now consistently outcompetes even the cheapest fossil fuel fired source of electricity, while costs continue to decrease [4].

Research on wind turbine aerodynamics have played a huge role in improving
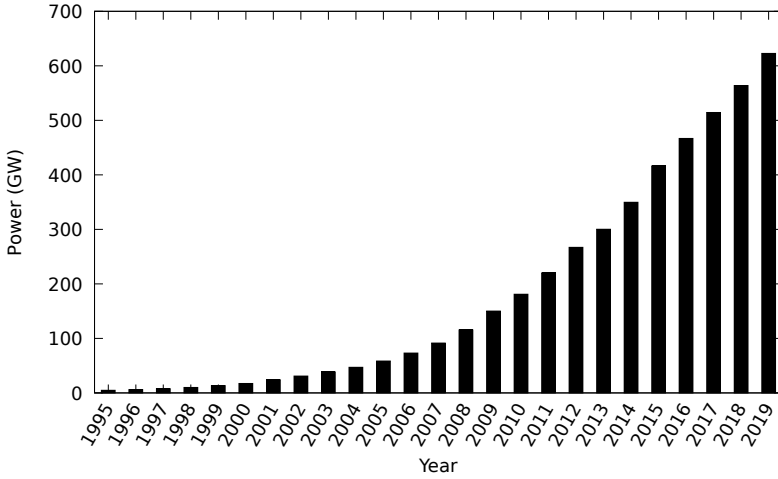
Figure 1.3: Worldwide cumulative installed capacity of wind energy over time.
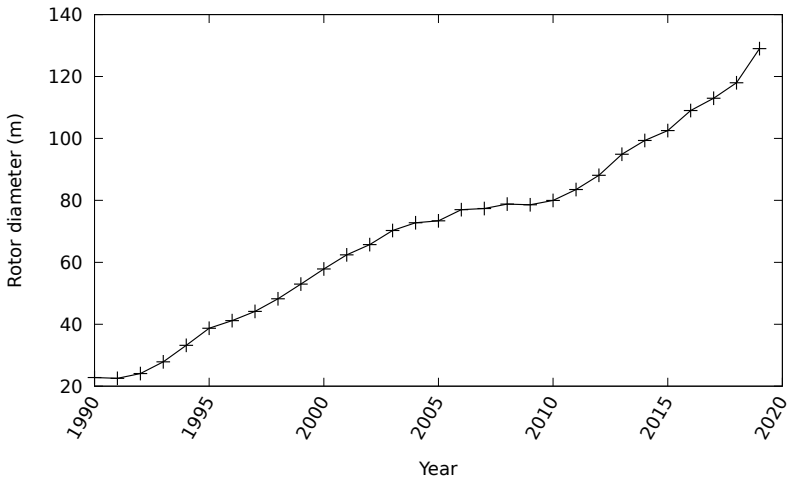
Figure 1.4: Rotor diameter size over years [5].

the performance of turbine blades over the years. Aerodynamicists have a variety of tools at their disposal. Over the years field experiments, wind tunnel measurements and computational tools have all been used to push the envelope of wind turbine aerodynamics. Looking ahead, the wind turbines are likely to become increasingly efficient and complex. To facilitate tackling the upcoming challenges, the tools used for research must also improve. Uncertainties in research tools must be reduced by using higher fidelity tools to keep up with the growth of technology.

1.3. Wind turbine aerodynamics

Wind turbines operate at high Reynolds numbers and low Mach numbers which is somewhat unique compared to other external aerodynamic applications like aerospace, and greatly advantageous in terms of numerical analysis. The high Reynolds number means large regions of the flow can be considered inviscid except for a small region around the body known as the boundary layer (figure 1.5). The low Mach numbers imply that the flow remains incompressible. This combination of conditions have been exploited to develop a wide variety of numerical tools based on simplified forms of the Navier-Stokes equations. On the other hand, aerodynamic analysis of
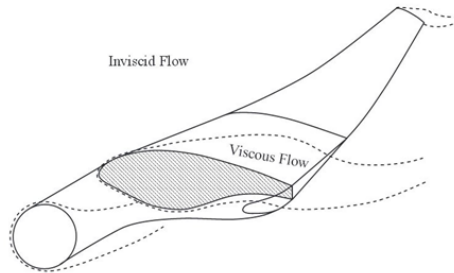
Figure 1.5: Inviscid flow and boundary layer regions [6] for the flow around a wind turbine blade.

wind turbines remains very challenging because of the disparate range of the scales involved (figure 1.6). The relevant length scales range from boundary layers on the turbine blades that are a few millimeters thick all the way upto wind farms that are tens of kilometers long.
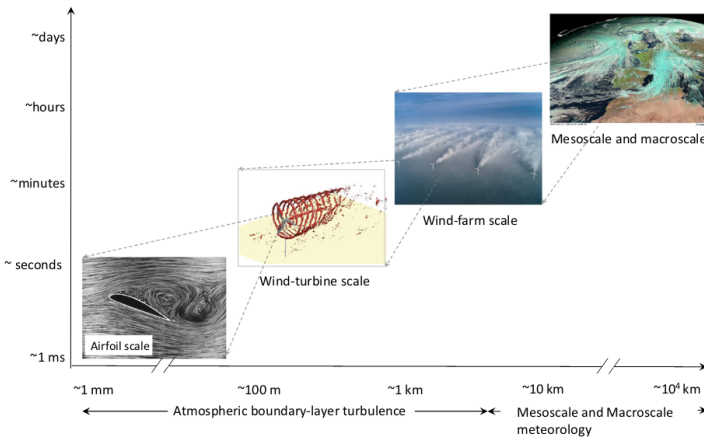
Figure 1.6: Range of scales of flow that are relevant in wind turbine aerodynamics [7].

In the following sections, a review of wind turbine aerodynamics is presented in three parts - aerodynamics of airfoils, aerodynamics of rotors and wind farm

analysis.

1.3.1. Airfoil analysis

Airfoil aerodynamics is at the heart of wind turbine rotor aerodynamics. Two dimensional analysis of airfoils is required by many different rotor design methods. Also analyzing the flow over a simpler two dimensional airfoil can give much needed insight into the physics of wind turbine aerodynamics under complex flow conditions. Due to wind turbines operating at very high Reynolds numbers, the flow around airfoils can be divided into the boundary layer very near the airfoil surface and an inviscid region away from the surface. While the boundary layer region is crucial in many applications, the inviscid analysis can also give useful information especially in attached flow regions. By neglecting the effect of viscosity and assuming the flow to be irrotational, potential flow analysis can be used. Panel methods [6] are very popular for the inviscid analysis of airfoils. The boundary layer can be analyzed separately by solving the simplified boundary layer equations [8]. Integral boundary layer methods can further simplify the two dimensional boundary layer equations into a one dimensional problem. Combination of the boundary layer methods with potential flow methods can give the global flow field. This is known as the interacting boundary layer approach [9] which is used in tools like XFOIL [10] and RFOIL [11].

However, the simplified analysis methods are only valid under attached flow conditions. While inviscid flow methods cannot be used to analyze separated flow, even boundary layer methods loose accuracy under such conditions. Passive flow control devices like vortex generators are widely used to improve the performance of the airfoil and delay separation. Potential flow methods and boundary layer methods cannot be used for such complex geometries readily. With increasing use of wind turbines in adverse weather conditions, rotor blades are also subject to erosion. Modeling such effects are also not yet possible with lower fidelity tools.
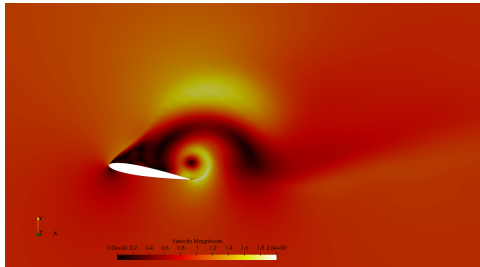
Figure 1.7: Velocity contours for a flow past an airfoil.

Computational Fluid Dynamics (CFD) methods do not have such limitations as the incompressible Navier Stokes equations are solved without any of the simplifying assumptions used for the methods mentioned above. Reynolds Averaged Navier Stokes (RANS) based methods are more widely used in combination with turbulence modeling, but the use of Large Eddy Simulations (LES), and hybrid LES methods

are gaining traction. Some of the challenges for CFD in airfoil aerodynamics are the prediction of laminar to turbulent transition and flow separation.

1.3.2. Rotor modeling

The design of rotor blades is a multi-disciplinary endeavour including aerodynamic analysis and structural design. Focusing on the aerodynamics only, due to faster computational times the Blade Element Momentum (BEM) theory is widely used in the early stages of the design process. The rotor design from BEM is then evaluated by an aeroelastic tool and if the design turns out to be efficient, it is evaluated by more advanced and accurate methods [12] like CFD. The rotor design is carried out with the intention of maximizing Annual Enegy Production (AEP) which is traditionally done by a mixture of designer experience and numerical optimization. Due to this iterative nature of the design process, computational speed at a reasonable accuracy overrides other criteria for selecting numerical tools.

BEM theory is the oldest method [12, 13] that has been constantly improved over the years [14]. The local blade element theory is used in combination with the one dimensional momentum theory. The assumptions are that the flow is inviscid and there are no losses. The rotor plane is assumed to be an ideal and permeable disc that extracts energy [12, 13]. Design of rotor blades is done iteratively based on two dimensional airfoil characteristics which can be found using the methods described in section 1.3.1 .

Vortex wake methods are a more accurate representation of the flow field around rotors. The flow is still assumed to be steady, but the rotor geometry is represented either as a lifting line [15] or a lifting surface [6]. Trailing and shed vortices are computed based on local flow conditions and airfoil characteristics. These vortices are then convected into the wake. In a lifting line model, the blade is represented by a single line with bound vorticity that varies radially. For lifting surfaces, the blade is represented by a zero thickness surface along the camber line instead of assuming that all lift is concentrated along a line, as is done in the lifting line theory. Three dimensional panel methods represent the exact geometry of the blade and are also used for rotor analysis. Similar to the two dimensional scenario, a potential flow solution is sought [6, 16]. All the methods described so far are inviscid and as the fidelity of the representation of rotor increases from the lifting line to lifting surface and finally to the panel method, so does the computational requirements.

CFD has also been used for rotor blade analysis. The first RANS simulations on wind turbine rotors were performed in the late 1990s and early 2000s [17–19]. A detailed historical overview of the use of CFD in rotor modeling can be found in Sumner et al [20]. While the use of RANS turbulence models is still common, there have been some hybrid LES studies on rotors reported recently [21].

The inviscid methods rely to some extent on two dimensional airfoil characteristics and are prone to missing some three dimensional phenomena like rotational augmentation where it is observed that the inboard sections of blades produce lift and drag that is significantly different from the two dimensional characteristics [20, 22]. Performing full three dimensional CFD analysis of rotors overcomes this limitation but that comes with an increase in computational cost. Some authors [23, 24] have

proposed the use of a hybrid between CFD and inviscid vortex based methods. The region close to the rotor is modeled using CFD and the flow outside is modeled as inviscid using the inviscid methods described earlier. Hybrid approaches are especially useful in studying wake aerodynamics.
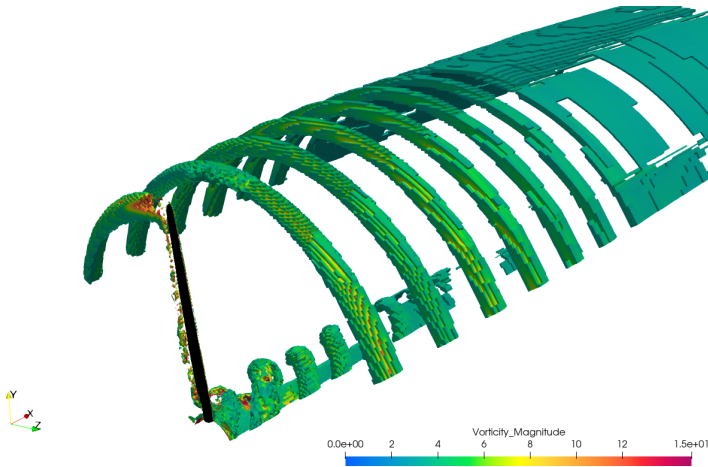
Figure 1.8: Vortex structures behind a turbine blade (periodic CFD simulation).

Another alternative to modeling rotor blades in CFD is the use of an actuator disc (AD) [25, 26]. The AD method is based on the blade element theory and represents the rotor with an equivalent porous surface and it is modeled as a source term that acts on the flow in that region. However, this method is mostly used to model wind farms to study the behavior of wakes which is the subject of the next section.

1.3.3. Wind farms

The wakes emanating from wind turbines are unsteady and highly turbulent. In addition, these wakes also interact with the atmospheric boundary layer which is also dynamic and turbulent. Also wind turbines, especially offshore turbines, are clustered in large wind farms to reduce installation and maintenance costs. Because of the relatively close spacing however, the wake from the turbines interact with each other in addition to the atmospheric boundary layer. Turbines that are downstream of another turbine can see power losses in the range of 40% in full wake conditions and experience increased fatigue loads [7, 27, 28].

The wind turbine wake can be divided into two regions [29] - the near wake region that extends about 2-4 rotor diameters downstream from the turbine and the far wake which is further downstream (see figure 1.9). The near wake region is influenced greatly by wind turbine features like blade profile, nacelle and is highly complex and three dimensional. The far wake region, however, is influenced more by wind turbine parameters such as thrust and power coefficients, incoming flow etc. As the spacing between any two turbines in a wind farm is larger than the near

wake, modeling the far wake accurately is more important than the near wake for understanding wind farm aerodynamics.
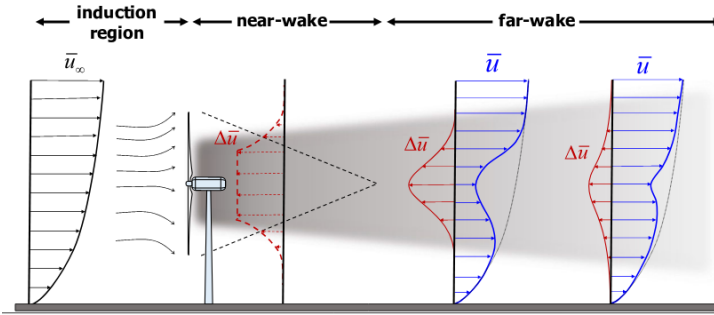
Figure 1.9: Schematic figure from Porté-Agel et al. [7] showing the time averaged flow features resulting from the interaction of a wind turbine with the incoming turbulent (atmospheric) boundary layer.

Numerical modeling of wakes is done either using analytical models or the use of three dimensional CFD analysis where the rotors are represented as an actuator disc or line or surface [28, 30]. Analytical models are used to predict the velocity deficit caused by the wind turbine wakes. These models have a lower accuracy compared to full 3D CFD simulations but have a very low computational cost. Analytical models mostly aim to predict the mean velocity deficit and do not consider the turbulence properties in the wake that can be significantly different from the undisturbed flow field. Thus, they are commonly used for design purposes like optimization of the wind farm layout, control of wind farms and other multi-disciplinary simulations. RANS modeling of wind farms have been extensively studied. Steady state tools like RANS are not suitable for capturing important dynamic wake effects that can significantly affect wind turbine loading. As computational resources have improved, the use of LES to study wind farm aerodynamics have become very common [21, 27].

Reviews of wind turbine wake aerodynamics can be found in literature (for example, Sanderse et al [28], Porté-Agel et al. [7], Stevens and Meneveau [27], Thé and Yu [21]).

Looking ahead
From the preceding section it can be seen that despite the vast range of scales involved in wind turbine aerodynamics, CFD methods are regularly used in all cases. As wind turbines become more complex, the use of a higher fidelity tool like CFD will become more prevalent. Additionally, CFD tools are increasingly being used in multi disciplinary problems like fluid structure interaction [31–33], shape optimization [34, 35], uncertainty quantification [36, 37], aeroacoustics [38, 39] to name a few. It must be noted that the references cited here are only a small selection of the state of the art. While industrial adoption of CFD for the full rotor design is likely far away in the future, CFD methods are increasingly being used earlier in the design process instead of just being used for evaluation purposes.

1.4. Goal

The main aim of this thesis is to develop a new pressure based solver within the framework of the open source multi-physics suite SU2 [40]. The immediate goals are to use this solver for various wind turbine aerodynamics applications like rotor simulations, vortex generator modeling and improving lower fidelity tools. In the long term, the goal is to take advantage of the open source nature of SU2 to not only improve the solver for aerodynamic applications but also to use it as a base for multi disciplinary applications like fluid structure interaction, aeroacoustics and optimization.

1.5. Dissertation overview

This thesis can be broadly divided into two parts - Chapters 2, 3 and 4 are devoted to the implementation details of the new pressure based solver in SU2 while chapters 5, 6 and 7 focus on different applications of CFD in wind energy.

Chapter 2 describes the governing equations for incompressible flow and different methods to solve them. First a short overview of the incompressible Navier Stokes equations is presented. The finite volume method that is commonly used in CFD is described for a general scalar equation highlighting the various numerical aspects. Subsequently, the challenges when dealing with the incompressible flow equations and different methods to overcome them are described. Finally, an overview on turbulence modeling for incompressible flows is given.

Chapter 3 describes the implementation of the governing equations and solution methods outlined in chapter 2 into SU2.

Chapter 4 presents some verification and validation results for the new solver. Verification of the accuracy of the solver is carried out against analytical solutions. Different test cases that replicate different conditions faced in wind turbine aerodynamics are chosen to validate the solver.

Chapter 5 presents the first steps towards modeling vortex generators in integral boundary layer methods. First a conceptual overview of the effect of vortex generators (VGs) on turbulent boundary layers is presented. CFD simulations of VGs on flat plates is then used to introduce the modeling approach.

Chapter 6 presents the effect of leading edge erosion using roughness models for eddy viscosity based RANS turbulence models. The performance of the roughness models is first validated against empirical formulations and experimental data for a flat plate and airfoils. Different methods to characterize roughness numerically are surveyed. Finally, the impact of roughness on turbulent boundary layers are studied and future research required to model roughness in integral boundary layers are presented.

Chapter 7 will present preliminary results from the CFD analysis of the widely studied New Mexico rotor blade.

References

[1] J. K. Kaldellis and D. Zafirakis, *The wind energy (r) evolution: A short review of a long history,* Renewable energy **36**, 1887 (2011).

[2] J. Beurskens, *History of wind energy, chapter 1 of: understanding wind power technology, schaffarczyk ap,* (2014).

[3] @GErenewables, *Twitter post,* (2020).

[4] IRENA, *Renewable Power generation costs in 2019*, Tech. Rep. (International Renewable Energy Agency, Abu Dhabi, 2020).

[5] Statista, *Rotor diameter size of onshore wind turbines worldwide from 1990 to 2019,* (2021).

[6] J. Katz and A. Plotkin, *Low-Speed Aerodynamics*, 2nd ed., Cambridge Aerospace Series (Cambridge University Press, 2001).

[7] F. Porté-Agel, M. Bastankhah, and S. Shamsoddin, *Wind-turbine and wind-farm flows: a review,* Boundary-Layer Meteorology **174**, 1 (2020).

[8] H. Lyon, *Aircraft Engineering and Aerospace Technology*, Vol. 7 (2014) pp. 32–36.

[9] H. Özdemir, *Interacting boundary layer methods and applications,* in *Handbook of Wind Energy Aerodynamics* (Springer International Publishing, 2020) pp. 1–53.

[10] M. Drela, *Xfoil: An analysis and design system for low reynolds number airfoils,* in *Low Reynolds number aerodynamics* (Springer, 1989) pp. 1–12.

[11] R. Van Rooij, *Modification of the boundary layer calculation in rfoil for improved airfoil stall prediction,* (1996).

[12] C. Bak, *Aerodynamic design of wind turbine rotors,* in *Advances in wind turbine blade design and materials*, Woodhead Publishing Series in Energy No. 47 (Woodhead Publishing, 2013).

[13] J. Manwell, J. Mcgowan, and A. L Rogers, *Wind energy explained: Theory, design and application, second edition,* (2006).

[14] J. Schepers, *Engineering models in wind energy aerodynamics: Development, implementation and analysis using dedicated aerodynamic measurements*, Ph.D. thesis, Delft University of Technology (2012).

[15] A. Van Garrel, *Development of a wind turbine aerodynamics simulation module,* (2003).

[16] A. van Garrel, *Multilevel panel method for wind turbine rotor flow simulations*, Ph.D. thesis, University of Twente (2016).

[17] E. P. Duque, W. Johnson, C. van Dam, R. Cortes, and K. Yee, *Numerical predictions of wind turbine power and aerodynamic loads for the NREL phase II combined experiment rotor*, Tech. Rep. (NASA Moffett Field CA AMES Reserach, 2000).

1

[18] N. Sørensen and M. Hansen, *Rotor performance predictions using a navier-stokes method,* in *1998 ASME Wind Energy Symposium* (1998) p. 25.

[19] J. Varela and D. Bercebal, *CFD Calculations of the Flow Around a Wind Turbine Nacelle*, Tech. Rep. (Centro de Investigaciones Energeticas Medioambientales y Tecnologicas (CIEMAT), 1999).

[20] J. Sumner, C. S. Watters, and C. Masson, *Cfd in wind energy: the virtual, multiscale wind tunnel,* Energies **3**, 989 (2010).

[21] J. Thé and H. Yu, *A critical review on the simulations of wind turbine aerodynamics focusing on hybrid rans-les methods,* Energy **138**, 257 (2017).

[22] S. Schreck and M. Robinson, *Rotational augmentation of horizontal axis wind turbine blade aerodynamic response,* Wind Energy: An International Journal for Progress and Applications in Wind Power Conversion Technology **5**, 133 (2002).

[23] G. Xu and L. N. Sankar, *Development of engineering aerodynamics models using a viscous flow methodology on the nrel phase vi rotor,* Wind Energy: An International Journal for Progress and Applications in Wind Power Conversion Technology **5**, 171 (2002).

[24] S. Schmitz and J.-J. Chattot, *A parallelized coupled navier-stokes/vortex-panel solver,* (2005).

[25] H. Aagaard Madsen, *The Actuator cylinder: A flow model for vertical axis wind turbines*, Ph.D. thesis, Aalborg University Centre, Denmark (1982).

[26] R. G. Rajagopalan and J. B. Fanucci, *Finite difference model for vertical axis wind turbines,* Journal of Propulsion and Power **1**, 432 (1985).

[27] R. Stevens and C. Meneveau, *Flow structure and turbulence in wind farms,* Annual review of fluid mechanics **49**, 311 (2017).

[28] B. Sanderse, S. van der Pijl, and B. Koren, *Review of computational fluid dynamics for wind turbine wake aerodynamics,* Wind Energy **14**, 799 (2011).

[29] L. Vermeer, J. N. Sørensen, and A. Crespo, *Wind turbine wake aerodynamics,* Progress in aerospace sciences **39**, 467 (2003).

[30] J. Sørensen and W. Shen, *Numerical Modeling of Wind Turbine Wakes ,* Journal of Fluids Engineering **124**, 393 (2002).

[31] M.-C. Hsu and Y. Bazilevs, *Fluid–structure interaction modeling of wind turbines: simulating the full machine,* Computational Mechanics **50**, 821 (2012).

[32] L. Wang, R. Quant, and A. Kolios, *Fluid structure interaction modelling of horizontal-axis wind turbine blades based on cfd and fea,* Journal of Wind Engineering and Industrial Aerodynamics **158**, 11 (2016).

1

[33] O. Guerri, A. Hamdouni, and A. Sakout, *Fluid structure interaction of wind turbine airfoils,* Wind Engineering **32**, 539 (2008).

[34] M. H. A. Madsen, F. Zahle, N. N. Sørensen, and J. R. R. A. Martins, *Multipoint high-fidelity cfd-based aerodynamic shape optimization of a 10 mw wind turbine,* Wind Energy Science **4**, 163 (2019).

[35] T. Dhert, T. Ashuri, and J. R. R. A. Martins, *Aerodynamic shape optimization of wind turbine blades using a reynolds-averaged navier–stokes model and an adjoint method,* Wind Energy **20**, 909.

[36] A. S. Padron, J. J. Alonso, F. Palacios, M. F. Barone, and M. S. Eldred, *Multi-fidelity uncertainty quantification: Application to a vertical axis wind turbine under an extreme gust,* in *15th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference.*

[37] A. Hsieh, D. C. Maniaci, T. G. Herges, G. Geraci, D. T. Seidl, M. S. Eldred, M. L. Blaylock, and B. C. Houchens, *Multilevel uncertainty quantification using cfd and openfast simulations of the swift facility,* in *AIAA Scitech 2020 Forum.*

[38] K. Luo, S. Zhang, Z. Gao, J. Wang, L. Zhang, R. Yuan, J. Fan, and K. Cen, *Large-eddy simulation and wind-tunnel measurement of aerodynamics and aeroacoustics of a horizontal-axis wind turbine,* Renewable Energy **77**, 351 (2015).

[39] E. H. Caicedo and M. S. Virk, *Aeroacoustics response of wind turbine blade profiles in normal and icing conditions,* Wind Engineering **42**, 243 (2018).

[40] K. Duraisamy, S. Copeland, T. Economon, M. Colonno, A. Aranake, T. Taylor, T. Lukaczyk, F. Palacios, A. Lonkar, J. Hicken, J. Alonso, and A. Campos, *Stanford University Unstructured (SU2): An open-source integrated computational environment for multi-physics simulation and design,* , 1 (2013).

2

Incompressible flow equations

In this chapter the governing equations and solution methods for incompressible flows are presented. The incompressible Navier Stokes equations are first given in dimensional and non-dimensional forms. Subsequently, the finite volume method applied to a general conservation equation is described. The difficulty associated with solving the incompressible flow equations numerically, namely the pressure velocity coupling, is described and the different methods to overcome this issue are presented. Finally, the solution procedure to solve turbulent incompressible flow equations are shown.

2.1. Navier Stokes equations

The governing flow equations for incompressible flow with constant density and viscosity and no heat transfer are

$$\frac{\partial(\rho u)}{\partial x} + \frac{\partial(\rho v)}{\partial y} + \frac{\partial(\rho w)}{\partial z} = 0, \tag{2.1}$$

$$\frac{\partial(\rho u)}{\partial t} + \frac{\partial(\rho uu)}{\partial x} + \frac{\partial(\rho uv)}{\partial y} + \frac{\partial(\rho uw)}{\partial z} = -\frac{\partial p}{\partial x} + \mu\left(\frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y} + \frac{\partial^2 u}{\partial z}\right) \tag{2.2}$$

$$\frac{\partial(\rho v)}{\partial t} + \frac{\partial(\rho uv)}{\partial x} + \frac{\partial(\rho vv)}{\partial y} + \frac{\partial(\rho vw)}{\partial z} = -\frac{\partial p}{\partial y} + \mu\left(\frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y} + \frac{\partial^2 v}{\partial z}\right) \tag{2.3}$$

$$\frac{\partial(\rho w)}{\partial t} + \frac{\partial(\rho wu)}{\partial x} + \frac{\partial(\rho wv)}{\partial y} + \frac{\partial(\rho ww)}{\partial z} = -\frac{\partial p}{\partial z} + \mu\left(\frac{\partial^2 w}{\partial x^2} + \frac{\partial^2 w}{\partial y} + \frac{\partial^2 w}{\partial z}\right) \tag{2.4}$$

where ρ is the constant density of the fluid, u, v and w are the $x-$, $y-$ and $z-$ components of the velocity vector respectively, p is the pressure and μ is the dynamic viscosity, assumed to be constant. Equation 2.1 is the continuity equation or the

15

mass conservation equation. For incompressible flows, this condition reduces to a zero divergence condition for the velocity vector. Equations 2.2, 2.3 and 2.4 are the momentum conservation equations in x, y and z directions respectively. Here it is assumed that the fluid is Newtonian. Under the incompressible flow assumption the energy equation is decoupled from the continuity and momentum equations and is therefore not shown here.

2.1.1. Non-dimensionalization

All the quantities in equations 2.1, 2.2, 2.3 and 2.4 are dimensional and their magnitudes can vary widely. The governing equations can be transformed into a non dimensional form by scaling the variables using reference values. The scaling parameters can be defined as follows

L - reference length (e.g. chord of the airfoil),

V - reference velocity (e.g. the free stream velocity, V_∞),

f - characteristic frequency (e.g., one cycle of a periodic process, or V/L),

p_0 - reference pressure (e.g., dynamic pressure, ρV_∞^2).

With the aid of these characteristic quantities we can define the following non-dimensional variables

$$x^* = \frac{x}{L}, \quad y^* = \frac{y}{L}, \quad z^* = \frac{z}{L},$$

$$u^* = \frac{u}{V}, \quad v^* = \frac{v}{V}, \quad w^* = \frac{w}{V},$$

$$p^* = \frac{p}{p_0} = \frac{p}{\rho V^2}, \quad t^* = tf.$$

Based on dimensional analysis, the following non dimensional numbers can be defined

$$St = \frac{fL}{V}, \tag{2.5}$$

$$Re = \frac{VL}{\nu}, \tag{2.6}$$

where St is known as the Strouhal number and Re is the Reynolds number. $\nu = \frac{\mu}{\rho}$ is the kinematic viscosity. Using the non-dimensional variables and the new non dimensional numbers, equation 2.1 can be written as:

$$\frac{\partial u^*}{\partial x^*} + \frac{\partial v^*}{\partial y^*} + \frac{\partial w^*}{\partial z^*} = 0,$$

and the equations 2.2, 2.3 and 2.4 as

$$St\frac{\partial u^*}{\partial t^*} + \frac{\partial(u^*u^*)}{\partial x^*} + \frac{\partial(u^*v^*)}{\partial y^*} + \frac{\partial(u^*w^*)}{\partial z^*} = -\frac{p_0}{\rho V^2}\frac{\partial p^*}{\partial x*} + \frac{1}{Re}\left(\frac{\partial^2 u^*}{\partial x^{*2}} + \frac{\partial^2 u^*}{\partial y^{*2}} + \frac{\partial^2 u^*}{\partial z^{*2}}\right),$$

$$St\frac{\partial v^*}{\partial t^*} + \frac{\partial(v^*u^*)}{\partial x^*} + \frac{\partial(v^*v^*)}{\partial y^*} + \frac{\partial(v^*w^*)}{\partial z^*} = -\frac{p_0}{\rho V^2}\frac{\partial p^*}{\partial y*} + \frac{1}{Re}\left(\frac{\partial^2 v^*}{\partial x^{*2}} + \frac{\partial^2 v^*}{\partial y^{*2}} + \frac{\partial^2 v^*}{\partial z^{*2}}\right),$$

$$St\frac{\partial w^*}{\partial t^*} + \frac{\partial (w^* u^*)}{\partial x^*} + \frac{\partial (w^* v^*)}{\partial y^*} + \frac{\partial (w^* w^*)}{\partial z^*} = -\frac{p_0}{\rho V^2}\frac{\partial p^*}{\partial z*} + \frac{1}{Re}\left(\frac{\partial^2 v^*}{\partial x^{*2}} + \frac{\partial^2 v^*}{\partial y^{*2}} + \frac{\partial^2 v^*}{\partial z^{*2}}\right).$$

If the reference values are chosen judiciously, the comparison of the different non dimensional quantities can yield information about the relative importance of different flow features. Additionally, matching the non dimensional parameters will allow for comparison of data across different experiments and numerical simulations.

The above equations can be written more compactly using the index notation (see below). In addition to using the index notation, the time and pressure references are chosen as $f = V/L$ and $p_0 = \rho V^2$ leading to the coefficients of the unsteady term and the pressure gradient term to be unity. The $*$ is dropped for the sake of convenience but all quantities shown are non dimensional.

$$\frac{\partial u_i}{\partial x_i} = 0, \tag{2.7}$$

$$\frac{\partial u_i}{\partial t} + \frac{\partial (u_i u_j)}{\partial x_j} = -\frac{\partial p}{\partial x_i} + \frac{1}{Re}\left(\frac{\partial^2 u_i}{\partial x_j \partial x_j}\right). \tag{2.8}$$

Here the velocities are represented by u_i, with $i = 1, 2, 3$ being the three components of the velocity vector and similarly x_i represents the three coordinate directions. Repeated indices indicate a summation over that index. Thus, equation 2.8 represents the three momentum equations.

The rest of the thesis will use index notation to keep the equations compact.

2.2. Numerical methods

The equations described in the previous section cannot be solved analytically except for some simplified cases and are solved numerically. Consider the example of a typical flow problem - the flow past a cylinder. The region of interest extends outwards from the cylinder as shown in figure 2.1. In order to find the velocities and pressures in this region, the physical domain is divided into smaller domains where the partial differential equations can be approximated in a simpler manner. This process is known as *discretization*. Since the partial differential equations are non-linear, care must be taken during the discretization process to preserve all the important flow phenomena in the approximate equations as well.

However, in order to solve the approximate equations numerically, they must converted into algebraic equations. The equations are then solved at some specified points (*nodes* or *grid points*) within each of the smaller domains. This process can be done using many different methods. The earliest approaches used truncated Taylor series approximations of the difference terms known as the finite difference method. A more common approach is the finite volume method where the governing equations are integrated over the smaller discretized regions to obtain an approximate algebraic equation. In this study, the finite volume discretization is used. More information about other discretization techniques can be found in literature (for example [1–3]).
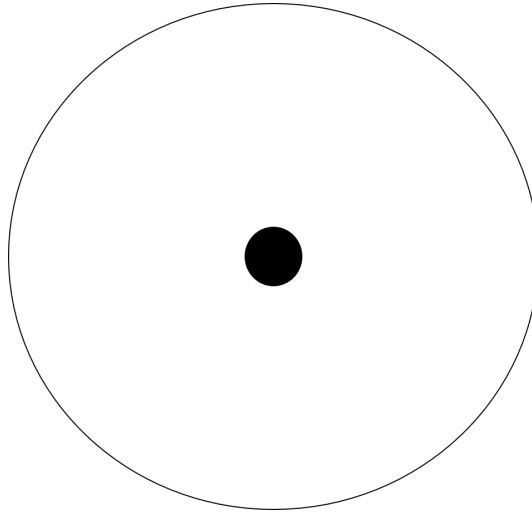
Figure 2.1: Region of interest for flow over a cylinder.

2.2.1. Finite Volume discretization

In this section the finite volume discretization methods will be described briefly. Consider a general conservation equation for any scalar variable ϕ

$$\frac{\partial(\rho\phi)}{\partial t} + \frac{\partial(\rho U_i \phi)}{\partial x_i} = \frac{\partial}{\partial x_i}\left(\Gamma\frac{\partial\phi}{\partial x_i}\right) + Q \tag{2.9}$$

where ρ is the constant fluid density, U_i is the velocity field (assumed constant in this section), Γ is the diffusion coefficient and Q is a source term. First, the physical domain on which the equation 2.9 needs to be solved is discretized into smaller domains known as *control volumes*. For simplicity consider a $2D$ case and define a control volume around a node P as shown in figure 2.2.
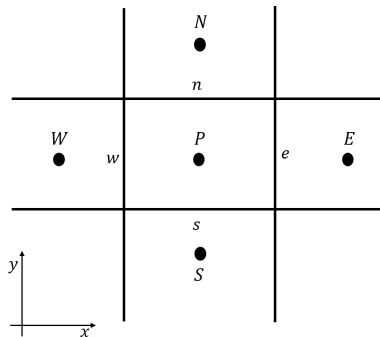
Figure 2.2: $2D$ control volume around a node P.

The control volume is bound on four sides by faces e, w, n and s representing east,

west, north and south respectively. Integrating equation 2.9 on a control volume Ω gives

$$\int_\Omega \frac{\partial(\rho\phi)}{\partial t} d\Omega + \int_\Omega \frac{\partial(\rho U_i \phi)}{\partial x_i} d\Omega = \int_\Omega \frac{\partial}{\partial x_i}\left(\Gamma\frac{\partial\phi}{\partial x_i}\right) d\Omega + \int_\Omega Q d\Omega. \qquad (2.10)$$

Discretization of the viscous term

First consider the diffusion term on the right hand side of equation 2.10. Using the divergence theorem the volume integral can be converted into a surface integral as

$$\int_\Omega \frac{\partial}{\partial x_i}\left(\Gamma\frac{\partial\phi}{\partial x_i}\right) d\Omega = \int_{\partial\Omega}\left(\Gamma\frac{\partial\phi}{\partial x_i}\right) n_i d(\partial\Omega), \qquad (2.11)$$

where $\partial\Omega$ represents the boundary of the control volume and n_i is the unit outward pointing normal of the boundary. In the simplified $2D$ example of figure 2.2, the integral can be written as a summation over the four faces as

$$\int_{\partial\Omega}\left(\Gamma\frac{\partial\phi}{\partial x_i}\right) n_i d(\partial\Omega) = \sum_f \left(\Gamma\frac{\partial\phi}{\partial x_i}\right) n_i \Delta S_f, \qquad (2.12)$$

where $f = e, w, n$ and s, n_i is the corresponding outward unit normal vector of the face f and ΔS_f is the area of the face f. Here the midpoint integration rule is used and the quantities on the face f are computed at the face centroids. Based on the axis directions shown in figure 2.2 the summation can be expanded as

$$\sum_f \left(\Gamma\frac{\partial\phi}{\partial x_i}\right) n_i \Delta S_f = \left(\Gamma\frac{\partial\phi}{\partial x_1}\right)_e \Delta S_e - \left(\Gamma\frac{\partial\phi}{\partial x_1}\right)_w \Delta S_w +$$
$$\left(\Gamma\frac{\partial\phi}{\partial x_2}\right)_n \Delta S_n - \left(\Gamma\frac{\partial\phi}{\partial x_2}\right)_s \Delta S_s. \qquad (2.13)$$

Here x_1 and x_2 are the x and y directions respectively. The derivatives at the faces can be written based on the nodal values of ϕ. For example, considering the neighboring nodes to east of node P in figure 2.2 to be E, the derivative of ϕ at the face e for a constant Γ can be written as

$$\left(\Gamma\frac{\partial\phi}{\partial x_1}\right)_e \Delta S_e = \Gamma\frac{\phi_E - \phi_P}{\Delta_{PE}}\Delta S_e = a_{E,v}\phi_E - a_{E,v}\phi_P, \qquad (2.14)$$

where Δ_{PE} is the distance between the nodes P and E and $a_{E,v}$ is the viscous coefficient of ϕ_E given by

$$a_{E,v} = \Gamma\frac{\Delta S_e}{\Delta_{PE}}.$$

Similar expressions can be written for the other terms in the summation in equation 2.13 to obtain an algebraic relation in terms of the nodal values of ϕ with the viscous coefficients being

$$a_{W,v} = \Gamma\frac{\Delta S_w}{\Delta_{PW}}, \quad a_{N,v} = \Gamma\frac{\Delta S_n}{\Delta_{PN}}, \quad a_{S,v} = \Gamma\frac{\Delta S_s}{\Delta_{PS}},$$

and
$$a_{P,v} = -(a_{E,v} + a_{W,v} + a_{N,v} + a_{S,v}).$$
Here $a_{P,v}$ is the viscous coefficient of ϕ_P.

Discretization of the advective term
Now focusing on the advective term of equation 2.10 and applying the divergence theorem again gives,

$$\int_{\Omega} \frac{\partial(\rho U_i \phi)}{\partial x_i} d\Omega = \int_{\partial\Omega} \rho U_i \phi n_i d(\partial\Omega). \qquad (2.15)$$

Similar to the discretization of the viscous term, the surface integral can be split into a summation over the four bounding faces of the control volume shown in figure 2.2

$$\int_{\partial\Omega} \rho U_i \phi n_i d(\partial\Omega) = \sum_f \left(\rho U_i \phi n_i\right)_f \Delta S_f,$$

$$\sum_f \left(\rho U_i \phi n_i\right)_f \Delta S_f = \left(\rho U_1 \phi\right)_e \Delta S_e - \left(\rho U_1 \phi\right)_w \Delta S_w + \left(\rho U_2 \phi\right)_n \Delta S_n - \left(\rho U_2 \phi\right)_s \Delta S_s,$$

where the same sign convention used for the viscous discretization is also applied here and the mid point integration rule is used to approximate the integral over the face. The term within the brackets can be found using the known velocity field. Determining the value of the scalar variable, ϕ, can be done in various ways. Some of the more common methods are described below.
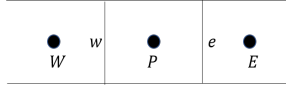
Figure 2.3: $1D$ control volume around node P and neighboring nodes E and W.

Central scheme   In a central scheme, the value of the scalar variable is approximated as the weighted average of the neighboring nodes. For example at the face e and assuming the eastward neighbor of node P in figure 2.3 is E, the value of ϕ is simply
$$\phi_e = \lambda_P \phi_P + \lambda_E \phi_E, \qquad (2.16)$$
where λ_P and λ_E are weighting factors of P and E respectively given by

$$\lambda_P = \frac{d_{Pe}}{d_{PE}}, \quad \lambda_E = \frac{d_{eE}}{d_{PE}}. \qquad (2.17)$$

Here d_{Pe} is the distance from node P to face e, d_{eE} is the distance from face e to node E and d_{PE} is the distance from node P to node E ($d_{PE} = d_{Pe} + d_{eE}$). Thus, the contribution from the face e to the summation can be written in terms of the nodal values of ϕ as

$$\left(\rho U_i \phi\right)_e \Delta S_e = \left(\rho U_1 \Delta S\right)_e \lambda_P \phi_P + \left(\rho U_1 \Delta S\right)_e \lambda_E \phi_E. \qquad (2.18)$$

Hence,

$$a_{E,c} = (\rho U_1 \Delta S)_e \lambda_E.$$

Similarly for the other faces around the control volume of node P in figure 2.2, the coefficients of the nodal values of ϕ can be written as

$$a_{W,c} = -(\rho U_1 \Delta S)_w \lambda_W, \quad a_{N,c} = (\rho U_2 \Delta S)_n \lambda_N, \quad a_{S,c} = -(\rho U_2 \Delta S)_s \lambda_S,$$

and as with the viscous term discretization,

$$a_{P,c} = -(a_{E,c} + a_{W,c} + a_{N,c} + a_{S,c}).$$

Upwind scheme An alternative to the central scheme, is to mimic the physics of the problem and determine the value at the face based on the direction of flow. Denoting the term

$$\rho U_i n_i \Delta S_f = \dot{m}_f,$$

as the mass flux across the face f, the direction of the flow can be found based on the sign of the mass flux. Choosing the value of ϕ as

$$\phi_e = \begin{cases} \phi_P & if \quad \dot{m}_e > 0, \\ \phi_E & if \quad \dot{m}_e < 0, \end{cases} \tag{2.19}$$

will replicate the physics of the flow at the face e. If the flow is in the positive x direction, the mass flux at the face e is positive ($\dot{m}_e > 0$) and the value of the variable at the node P is chosen. On the other hand if the direction of the flow is reversed, the value of ϕ at the node E is chosen. This ensures the physics of the flow is also replicated in the discretized equation. The coefficients of the nodal values of ϕ for the face e are then

$$a_{E,c} = \max(-m_e, 0.0), \quad a_{P,c} = \max(m_e, 0.0). \tag{2.20}$$

Similar expressions can be written at other faces to find the coefficients of ϕ_P and its neighbors.

To examine the difference in behavior of the central and upwind scheme, consider a one dimensional pure advection problem given by

$$\frac{\partial \rho \phi}{\partial t} + \frac{\partial (\rho U \phi)}{\partial x} = 0. \tag{2.21}$$

Let the domain of the problem be $x \in [0, 1]$ and $U = 1$ is the advection velocity and $\rho = 1$ is the density. Assume an initial condition for ϕ as

$$\phi_0 = \begin{cases} 1.0 & if \quad x < 0.25, \\ 0 & if \quad x > 0.25. \end{cases} \tag{2.22}$$

Since no dissipation is present, the initial condition must be advected across the domain without any loss in information. Figure 2.4 shows the numerical results
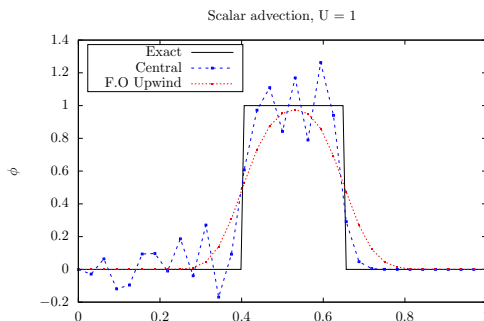
Figure 2.4: Comparison of numerical solutions and exact solution for one dimensional scalar advection equation using central and first order upwind schemes.

from both the upwind and central schemes compared against the exact solution. In this case, a uniform grid with 128 elements was used. The solution shown is at time $t = 0.5$. While the central scheme is more accurate, it is prone to instability which can be seen in the form of the unphysical wiggles that appear around the exact solution. This is understandable since the central discretization does not take the physics of the flow into account. However, while the upwind scheme does not have any wiggles in the solution, there is a loss of information and thus it is not very accurate. The source of these errors can be found by analyzing the truncation errors due for the upwind and central schemes [1, 3]. Expanding the value of ϕ_f around the upwind node U using Taylor series

$$\phi_f = \phi_U + \left(\frac{\partial \phi}{\partial x}\right)_U \Delta x + + \frac{1}{2!} \left(\frac{\partial^2 \phi}{\partial x^2}\right)_U \Delta x^2 + ... \tag{2.23}$$

where Δx is the distance between the upwind node U and the face f. The upwind discretization presented so far only uses the first term in this expansion. The first term where the approximation gets truncated is a function of Δx and thus this scheme is said to be first order accurate. In order to improve the accuracy of the upwind scheme the value at the face can be improved by considering more terms of the Taylor series expansion.

Second order upwind   Instead of using only the value at the upstream node, the approximation of the face value of ϕ can be improved if the gradient of ϕ at the upstream node is also used. The value of ϕ at a face f is reconstructed using both the value of ϕ at the upstream node and its gradient as

$$\phi_f = \phi_U + \left(\frac{\partial \phi}{\partial x}\right)_U \Delta x,$$

where f is the face under consideration, U denotes the upstream node and Δx is the distance from the node U to the face f. In this approximation, the truncation error is of the order (Δ^2) and the value is second order accurate. The solution of

the one dimensional pure advection equation using the second order upwind scheme
is shown in figure 2.5. Clearly, there is less loss of information compared to the first
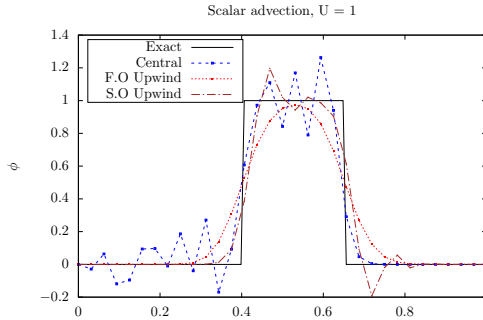
Figure 2.5: Comparison of numerical solution and exact solution for one dimensional scalar
advection equation using central, first and second order upwind schemes.

order approximation and there are fewer wiggles than the central discretization.

Slope limiters However, reconstructing the face value using gradients can also
lead to non-physical oscillations and convergence issues at times. This can be seen
in figure 2.5 at the end of the wave. A monotonic upwind discretization can be found
based on the concept of total variation [2]. The total variation of ϕ in equation 2.21
is given by

$$TV = \int \left| \frac{\partial \phi}{\partial x} \right| dx. \tag{2.24}$$

The total variation of any physically admissible solution does not increase in time [2,
4]. A numerical scheme can produce a physically monotonic solution only if it
ensures that the total variation of the solution does not increase. Such schemes are
called Total Variation Diminishing (TVD) schemes. Based on the concept of total
variation, slope limiters that ensure the solution remains smooth and monotonic near
sharp gradients can be derived [2, 4]. The value of ϕ at a face f is reconstructed
using slope limiters, given by

$$\phi_f = \phi_U + \psi(r) \left(\frac{\partial \phi}{\partial x_i} \right)_U \Delta x,$$

where f is the face under consideration, U denotes the upstream node, Δx is the
distance from the node U to the face f and $\psi(r)$ is the slope limiter function. For
the face e in figure 2.3, r is given by

$$r = \frac{u_P - u_W}{u_E - u_P}.$$

For the van Albada [5] slope limiter, $\psi(r)$ is given by

$$\psi(r) = \frac{r^2 + r}{r^2 + 1}.$$

The solution of the $1D$ scalar advection equation using the van Albada slope limiter is shown in figure 2.6. With the use of the van Albada slope limiter, a more accurate
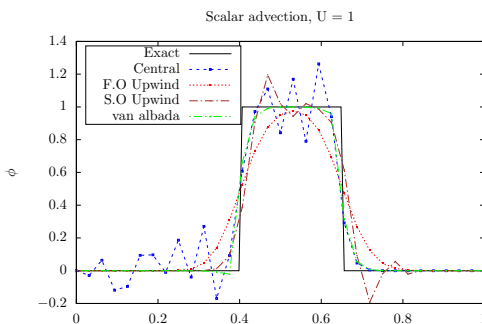
Figure 2.6: Comparison of numerical solution and exact solution for one dimensional scalar advection equation using central, first, second order upwind schemes with and without slope limiter.

representation of the exact solution is found without any oscillations.

Discretization of the source term
The source term, Q, can be discretized using the mid point integration rule. For example on a control volume around a node P shown in figure 2.2, the discretized form of the source term is

$$\int_\Omega Q d\Omega \approx Q_P |\Omega| = b_P. \tag{2.25}$$

This discretization is approximately second order accurate [3].

Discretized equation
The semi discretized form of equation 2.10 for a node P can be written as

$$\int_\Omega \frac{\partial \phi}{\partial t} d\Omega + a_P \phi_P + \sum_{C \in \mathcal{N}(P)} a_C \phi_C = b_P, \tag{2.26}$$

where $C \in \mathcal{N}(P)$ denotes the neighbors of the node P and b_P is the discretized source term contribution. The coefficients, a, are the sum of the advective and viscous contributions

$$a_P = a_{P,c} + a_{P,v}, \quad a_C = a_{C,c} + a_{C,v}.$$

Time discretization
Finally, focusing on the unsteady term in equation 2.10 and using the mid point rule for a control volume around a node P as shown in figure 2.2 gives,

$$\int_\Omega \frac{\partial \phi}{\partial t} d\Omega \approx \frac{\partial \phi}{\partial t} |\Omega|_P. \tag{2.27}$$

The time derivative can be discretized in different ways. The simplest method is to use the Euler time integration schemes. Denoting quantities at a time level t^n as ϕ^n, the approximation for the unsteady term is

$$\frac{\partial \phi}{\partial t}|\Omega|_P \approx \frac{\phi_P^{n+1} - \phi_P^n}{\Delta t}|\Omega|_P. \tag{2.28}$$

There are now two possible approaches to solve equation 2.26. Either all the ϕ values in the spatially discretized terms could be from the current time level t^n giving an explicit formulation or these values could be from the time level t^{n+1} giving an implicit formulation.

Explicit Euler   Using the known values of ϕ^n in the spatial discretization gives

$$\frac{\phi_P^{n+1} - \phi_P^n}{\Delta t}|\Omega|_P + a_P \phi_P^n + \sum_{C \in \mathcal{N}(P)} a_C \phi_C^n = b_P^n. \tag{2.29}$$

The only unknown quantity in this equation is ϕ^{n+1} and this equation can be written in an explicit form to solve for ϕ^{n+1} as

$$\phi_P^{n+1} = \phi_P^n + \frac{\Delta t}{|\Omega|_P}\left(b_P - a_P \phi_P^n - \sum_{C \in \mathcal{N}(P)} a_C \phi_C^n\right),$$

or

$$\phi_P^{n+1} = \phi_P^n - \frac{\Delta t}{|\Omega|_P} R_P^n(\phi), \tag{2.30}$$

where $R^n(\phi)_P$ is the residual at the time level t^n given by

$$R_P^n(\phi) = a_P \phi_P^n + \sum_{C \in \mathcal{N}(P)} a_C \phi_C^n - b_P. \tag{2.31}$$

Implicit Euler   Using the values of ϕ from time level t^{n+1} instead in equation 2.26 gives

$$\frac{\phi_P^{n+1} - \phi_P^n}{\Delta t}|\Omega|_P + a_P \phi_P^{n+1} + \sum_{C \in \mathcal{N}(P)} a_C \phi_C^{n+1} = b_P^n. \tag{2.32}$$

Thus, an implicit equation for ϕ_P^{n+1} is obtained, namely

$$\phi_P^{n+1}\frac{|\Omega|_P}{\Delta t} + a_P \phi_P^{n+1} + \sum_{C \in \mathcal{N}(P)} a_C \phi_C^{n+1} = b_P^n + \phi_P^n \frac{|\Omega|_P}{\Delta t},$$

which can be further simplified as

$$a_P^t \phi_P^{n+1} + \sum_{C \in \mathcal{N}(P)} a_C \phi_C^{n+1} = B_P, \tag{2.33}$$

where the coefficient a_P^t contains the contribution from the spatial discretization as well as temporal discretization and the source term B_P contains the contribution from the previous time step

$$a_P^t = a_P + \frac{|\Omega|_P}{\Delta t}, \quad B_P = b_P^n + \phi_P^n \frac{|\Omega|_P}{\Delta t}. \tag{2.34}$$

Alternately if only a steady state solution is required, using the residual $R_P(\phi)$ defined earlier, but now at the time level t^{n+1} the above equation can be simplified as

$$\frac{\phi^{n+1} - \phi^n}{\Delta t} |\Omega|_P + R_P^{n+1}(\phi) = 0. \tag{2.35}$$

Instead of solving for ϕ^{n+1} directly, the numerator of the time discretization can be written as the update to the solution, $\Delta \phi^n = \phi^{n+1} - \phi^n$, at time level t^n. Thus, the equation 2.35 can also be written as

$$\frac{\Delta \phi^n}{\Delta t} |\Omega|_P + R_P^{n+1}(\phi) = 0. \tag{2.36}$$

The residual $R_P^{n+1}(\phi)$ is also unknown and can be linearized about the time t^n as

$$R_P^{n+1}(\phi) = R_P^n(\phi) + \frac{\partial R_P^n}{\partial t} \Delta t + \mathcal{O}(\Delta t^2),$$

or

$$R_P^{n+1}(\phi) = R_P^n(\phi) + \sum_{C \in \mathcal{N}(P)} \frac{\partial R_P^n(\phi)}{\partial \phi_C} \Delta \phi_C^n + \mathcal{O}(\Delta t^2). \tag{2.37}$$

The term $\sum_{C \in \mathcal{N}(P)} \frac{\partial R_P^n(\phi)}{\partial \phi_C}$ is known as the Jacobian matrix. Thus equation 2.36 can be written as

$$\frac{\Delta \phi^n}{\Delta t} |\Omega|_P + \sum_{C \in \mathcal{N}(P)} \frac{\partial R_P^n(\phi)}{\partial \phi_C} \Delta \phi_C^n = -R_P^n(\phi). \tag{2.38}$$

This can also be written in the form of equation 2.33 as

$$a_P^{t'} \phi_P + \sum_{C \in \mathcal{N}(P)} a_C' \phi_C = B_P', \tag{2.39}$$

However, the coefficients are now different, namely

$$a_P^{t'} = a_P' + \frac{|\Omega|_P}{\Delta t}, \quad B_P' = -R_P^n(\phi). \tag{2.40}$$

Here $a_P^{t,'}$ contains the coefficients of node ϕ_P from the Jacobian matrix and $a_{\mathcal{N}(P)}'$ contains the coefficients of the neighbors of node ϕ_P from the Jacobian matrix.

So far it has been assumed that the coefficients of nodal values of ϕ, a_P are constant. This is true for the general scalar conservation equation (equation 2.9) with constant velocity field U. However, if the velocity field is itself not constant, the coefficients a_P are evaluated using values at the time level t^n.

System of linear algebraic equations   The equation 2.33 or equation 2.39 can be written as a system of linear algebraic equations with the unknowns as the nodal values of ϕ as

$$\mathbf{A}_{ij}\phi_j = b_i, \tag{2.41}$$

where $\mathbf{A}_{ij}$ is the coefficient matrix, ϕ_j is the vector of all the nodal values of ϕ and b_i is the right hand side (RHS). This equation can be solved using different linear solvers for which information can be found in literature [1, 3, 6–8]. The linear solvers available in SU2 are the Flexible Generalized Minimal Residual (FGMRES) [8] and Biconjugate Gradient Stabilized (BiCGSTAB)[6–8] which will also be used in the pressure based solver.

In this section, the procedure to solve a generic partial differential equation of the form equation 2.9 was described. The momentum equations (equation 2.8) closely resemble the generic conservation equation and the discretization described here can be used to solve them. However, when dealing with incompressible flows additional complications arise which will be described next.

2.3. Pressure velocity coupling

For an incompressible flow, the main variables of interest are the velocities, u_i and pressure p (*primitive variables*). The momentum equations (equation 2.8) closely resemble the general scalar equation 2.9. However, unlike the scalar equation the velocity field is unknown and thus the equations are non linear. Additionally, it can be seen that the pressure, which is also unknown, appears as a gradient term in the momentum equations but the continuity equation (equation 2.7) does not contain pressure and cannot be used to solve for it. Thus, despite having four equations for the four unknowns, a direct solution is not possible from the system of equations. This pressure velocity coupling is the biggest challenge in solving the incompressible flow equations [1, 3, 9–11]. For compressible flow problems, the continuity equation acts as an evolution equation for density which can be used in conjunction with the energy equation and thermodynamic relations to obtain the pressure field. However as the continuity equation reduces to a divergence condition on the mass flux for incompressible flows and the energy equation is decoupled there is no direct way to compute the pressure field.

A conceptual interpretation of this scenario would be to consider the implications of the constant density assumption. In any medium, sound travels as pressure and density disturbances. Since density is assumed to be a constant, the speed at which the pressure disturbances must travel is infinite. Thus the effect of any pressure disturbance is felt instantaneously everywhere in the domain. Realistically however, the speed of sound is many times faster than the reference velocities encountered in incompressible problems. This leads to the commonly used condition on the Mach number, $Ma < 0.3$, to determine if a flow is incompressible or not. The Mach number is a non-dimensional number and is defined as the ratio of the reference velocity, V, and the speed of sound, c. Thus at lower Mach numbers the speed of sound is much greater than the reference velocity and pressure disturbances can be assumed to travel much faster than the flow velocity.

This discrepancy in the different propagation speeds of pressure and velocity is reflected in the different natures of the governing equations. Pressure behaves in an elliptic manner propagating in all directions simultaneously and instantly, whereas the velocities behave in a parabolic manner i.e. have a definite marching direction [3]. In order to solve the incompressible flow equations, the two different natures of the pressure and velocity must be resolved.

2.4. Solution methods

One way to overcome the challenges posed by the incompressible form of the Navier Stokes equations is to solve the compressible Navier Stokes equations as they are also valid for incompressible flows. However there are many known drawbacks of this approach. The equations become very stiff at lower Mach numbers leading to poor convergence behavior. Additionally, the excessive numerical diffusion causes a degradation in the accuracy of the solutions [12, 13]. Preconditioning of the compressible Navier Stokes equations [14] can be used to overcome some of the limitations posed by solving the compressible equations, but such methods are more commonly used in flows where a wide range of Mach numbers are observed.

To alleviate the problems with pressure velocity coupling in the incompressible Navier Stokes equations, the pressure can be eliminated from the equations using derived quantities like stream function and vorticity [2, 3] which can then be solved to obtain the flow field. In $2D$, the stream function, ψ, and vorticity, ω, are related to the velocities u and v as

$$u = \frac{\partial \psi}{\partial y}, \quad v = -\frac{\partial \psi}{\partial x} \tag{2.42}$$

and

$$\omega = \frac{\partial u}{\partial y} - \frac{\partial v}{\partial x}. \tag{2.43}$$

Based on these definitions, a Poisson equation for the streamfunction and a conservation equation for vorticity can be found [3]. The nature of the two equations reflect the nature of pressure and velocity but the biggest advantage is that the pressure is eliminated as a dependent variable. However, there are some major drawbacks as well. It is difficult to specify boundary conditions for the stream function and vorticity, especially for complex geometries. Also this method does not generalize well into $3D$ as both vorticity and stream function have three components and thus there are six equations to be solved.

Therefore, the use of the primitive variables (pressure and velocity) is preferred. There are two broad classes of methods to resolve the pressure velocity coupling in primitive variables: the density based approach and the pressure based approach.

An example of the density based method is the pseudo compressibility approach [9, 15, 16] where an artificial speed of sound is introduced in the continuity equation to mimic the compressible flow formulation. The most common approach, especially for constant density flows, is to introduce a time derivative of the pressure in the continuity equation to transform the behavior of the continuity equation from

elliptic to hyperbolic. Equation 2.7 is then becomes

$$\frac{1}{\beta}\frac{\partial p}{\partial t} + \frac{\partial u_i}{\partial x_i} = 0. \tag{2.44}$$

Here β is the artificial compressibility factor. The larger the value of β, the closer the equation 2.44 is to the incompressible equations. Clearly, equation 2.44 is not time accurate and is valid only for steady state problems. A dual time stepping algorithm can overcome this limitation. The artificial compressibility factor, β can be considered as a form of preconditioning for the continuity equation. Its value will have a marked effect on the convergence behavior of the solution. For steady state problems, it can affect the accuracy of the solution via artificial dissipation. This method belongs to the more general approach of pre-conditioned compressible flow solution methods. Economon [12] suggests to precondition all the equations instead of only the continuity equation for robustness and stability of the solver.

The existing incompressible solver in SU2 follows this approach and has been extended to variable-density flows and heat transfer applications [12]. The aim of this thesis is to implement a pressure based solver in SU2. The details of such a solver will be given in the next section.

2.5. Pressure based methods

The difficulty in solving the incompressible equations was explored qualitatively earlier. Numerically it manifests as the well known checkerboard pressure problem [1, 17]. This problem arises from the discretization of the pressure gradient term in the momentum equations. If the pressure gradient is treated as a source
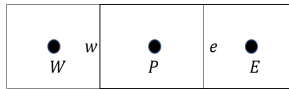
Figure 2.7: One dimensional control volumes around nodes P and the neighboring nodes E and W.

term and integrated using the mid point rule over the control volume like the one shown in figure 2.7, the discretized form can be written as

$$\int_{\Omega}\frac{\partial p}{\partial x_i}d\Omega \approx \frac{\partial p}{\partial x_i}|\Omega|. \tag{2.45}$$

In the simplified one dimensional scenario, for the control volume around a node P with two neighbors E and W to the east and west respectively (figure 2.7), using a central difference scheme the discretized form reduces to

$$\frac{\partial p}{\partial x_i}|\Omega| = \frac{p_E - p_W}{2\Delta x}|\Omega|. \tag{2.46}$$

A uniform grid spacing of Δx is assumed throughout the domain for simplicity. Thus, the discretized pressure gradient term involves only the difference between

alternating nodes. Similarly, the continuity equation

$$\frac{\partial u_i}{\partial x_i} = 0 \tag{2.47}$$

which discretized in a one dimensional scenario for a control volume around node P is simply

$$\int_\Omega \frac{\partial u}{\partial x} d\Omega \approx (u_e - u_w)\Delta S = 0. \tag{2.48}$$

Using the central scheme gives

$$u_E - u_W = 0. \tag{2.49}$$

Once again, the discretized equation around a node P relies only on alternating nodes. Under such conditions any zigzag pressure field will appear uniform and a zigzag velocity field will satisfy the continuity equation. Additionally, if a certain pressure field is found as the solution, infinitely more solutions can be found by adding an arbitrary zigzag variation to the smooth solution.

2.5.1. Staggered grids

Since the pressure terms in the discretized momentum equation was decoupled from the velocity, one potential solution is to stagger the locations where pressure and velocities are computed. In a staggered arrangement, the velocities are stored at the cell faces and pressure is stored at the centroid. In two dimensions, the x component of the velocity is staggered in the x direction and the y component of the velocity is staggered in the y direction as seen in figure 2.8.
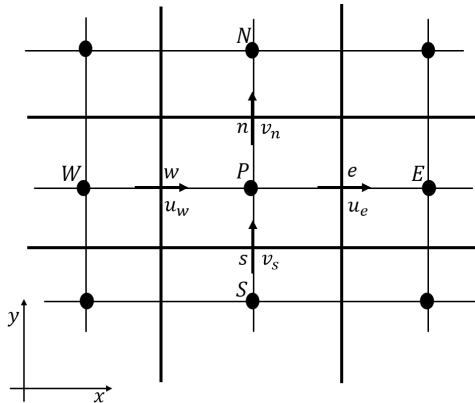
Figure 2.8: Staggered grid showing the locations of pressure and velocity variables in $2D$.

This scheme was first suggested by Harlow and Welch [18] in their marker and cell (MAC) method. One consequence of storing the velocities and pressure at staggered locations is that the control volumes for each of the momentum and continuity equations is now different. In figure 2.8, the control volume for the continuity

equation is formed around node P, for the momentum equation in x direction, the control volume is formed around the mid point of face e (or w) and for the momentum equation in y direction the control volume is formed around the mid point of face n (and s). The pressure gradient term in the momentum equations can be computed directly based on cell values. Also, there is no need to interpolate velocities to the faces since they are available directly. The momentum equations for the velocity components can be solved in a segregated manner treating each of them similar to the scalar equation 2.9. For example, the equation for the x component of velocity on the face e in figure 2.8 will be

$$a_e^u u_e + \sum_{f \in \mathcal{N}(e)} a_f^u u_f = -|\Omega|_e \frac{\partial p}{\partial x_e}, \tag{2.50}$$

where a_e^u is the coefficient of u at the face e, a_f are the coefficients of the neighboring nodes $\mathcal{N}(e)$. The coefficients can be found as described in section 2.2.1. However, unlike the scalar transport equation since the velocity field itself is unknown, the coefficients are a function of the velocities. The pressure gradient term is treated as a constant within the control volume. The gradient can now be found as

$$\frac{\partial p}{\partial x} = \frac{p_E - p_P}{\delta x_{PE}}, \tag{2.51}$$

which can be found without any interpolation since the pressure values are stored on the nodes P and E. Since consecutive nodes are used to find the pressure gradient, the checkerboard problem is eliminated. However, the value of the pressure is not known and needs to be solved for simultaneously. At this stage, in order to find a new equation for the pressure, the divergence of the discretized momentum equations can be taken and along with the continuity equation a Poisson equation for the pressure can be found [3]. The most commonly used procedure to do this is described now.

Equation 2.50 can only be solved if the pressure is either known beforehand or at least estimated. If an estimate of the pressure field is used, the resulting velocity field will also be an estimate. Denoting the velocity estimate by an $*$, the equation 2.50 can be written more accurately as

$$a_e^u u_e^* + \sum_{f \in \mathcal{N}(e)} a_f^u u_f^* = -|\Omega|_e \frac{\partial p^*}{\partial x_e}. \tag{2.52}$$

Similarly, the estimate of pressure is represented by p^*. Denoting the corrections to velocities and pressure by u_e' and p', the corrected velocity and pressure are

$$p = p^* + p', \quad u_e = u_e^* + u_e'. \tag{2.53}$$

Subtracting equation 2.52 from equation 2.50,

$$a_e^u u_e' + \sum_{f \in \mathcal{N}(e)} a_f^u u_f' = -|\Omega|_e \frac{\partial p'}{\partial x_e}. \tag{2.54}$$

Analogous equations can be written for the vertical component of the velocity at the face n in figure 2.8 as

$$a_n^v v_n' + \sum_{f \in \mathcal{N}(n)} a_f^v v_f' = -|\Omega|_n \frac{\partial p'}{\partial y_n}. \tag{2.55}$$

Unlike the momentum equations where the control volume for velocities are defined around cell faces (like e in figure 2.8), the continuity equation is solved using the control volume around a node. In figure 2.8, the discretized continuity equation around the node P is

$$\sum_f \dot{m}_f = \dot{m}_e + \dot{m}_w + \dot{m}_n + \dot{m}_s = 0. \tag{2.56}$$

Since the velocities are stored at the faces, the mass fluxes can be written in terms of the face velocities directly as

$$\dot{m}_e = \rho u_e \Delta S_e = \rho u_e^* \Delta S_e + \rho u_e' \Delta S_e = \dot{m}_e^* + \dot{m}_e', \tag{2.57}$$

where $\dot{m}_e^*$ and $\dot{m}_e'$ is the estimated mass flux and the correction of the mass flux. The continuity equation in terms of the estimated and corrected mass flux is

$$\sum_f (\dot{m}_f^* + \dot{m}_f') = 0,$$

or

$$\sum_f \dot{m}_f' = -\sum_f \dot{m}_f^*. \tag{2.58}$$

Substituting for mass flux correction in terms of the velocity corrections

$$\rho u_e' \Delta S_e - \rho u_w' \Delta S_w + \rho u_n' \Delta S_n - \rho u_s' \Delta S_s = -\sum_f \dot{m}_f^*. \tag{2.59}$$

The estimated mass flux can be computed from the solution of equation 2.52. Substituting for the velocity corrections from equation 2.54 will give a new equation for the pressure corrections which can be used to overcome the pressure velocity coupling. Patankar and Spalding [19] introduced an additional simplification to equation 2.54 by neglecting the term $\sum_{f \in \mathcal{N}(e)} a_f^u u_f'$ giving

$$u_e' = -\frac{|\Omega|_e}{a_e^u} \frac{\partial p'}{\partial x_e}. \tag{2.60}$$

Substituting this simplified equation of the velocity correction in equation 2.59 results in

$$-\left(\rho \frac{|\Omega|_e}{a_e^u} \frac{\partial p'}{\partial x_e} \Delta S_e - \rho \frac{|\Omega|_w}{a_w^u} \frac{\partial p'}{\partial x_w} \Delta S_w + \rho \frac{|\Omega|_n}{a_n^v} \frac{\partial p'}{\partial y_n} \Delta S_n - \rho \frac{|\Omega|_s}{a_s^v} \frac{\partial p'}{\partial y_s} \Delta S_s \right) = \sum_f \dot{m}_f^*. \tag{2.61}$$

Denote the coefficient of the pressure gradient terms as

$$D_f = \rho \frac{|\Omega|_f}{a_f} \Delta S_f, \tag{2.62}$$

where f is the face. Equation 2.61 can then be written as

$$\sum_f -D_f \frac{\partial p'}{\partial x_e} = \sum_f \dot{m}_f^*. \tag{2.63}$$

This equation is a discretized Poisson equation for the pressure correction. This pressure corrections will be used to correct the estimated pressure used in equation 2.52.

Pressure and velocity corrections
Once the pressure corrections are found by solving equation 2.63, the velocity corrections can be found from equation 2.60

$$u_e' = -\frac{|\Omega|_e}{a_e^u} \frac{\partial p'}{\partial x_e}.$$

With the pressure and velocity corrections, the estimated values can now be updated as follows

$$p = p^* + \alpha_p p', \tag{2.64}$$
$$u = u^* + u'. \tag{2.65}$$

Here α_p is an under-relaxation parameter for the pressure correction. At convergence, the estimated velocities will satisfy the continuity equation rendering the right hand side of the equation 2.63 zero which then makes the pressure corrections and subsequently velocity corrections also go to zero. It is necessary to use some under-relaxation in pressure correction to ensure convergence [1, 3, 19]. No under-relaxation is necessary for the velocity corrections.

SIMPLE algorithm
This procedure of using an estimate of the pressure field to find the velocity field using the momentum equations and then correcting both using the continuity equation is known as the Semi-Implicit Method for Pressure-Linked Equations or SIMPLE [19, 20]. The solution algorithm is

1. Guess the pressure field p^*.

2. Solve the momentum equations like equation 2.52 to find the estimated velocities u_i^*.

3. Find the mass flux at the faces m_f^* using the estimated velocities.

4. Assemble the pressure correction equation based on the mass fluxes and the momentum equation.

5. Solve the pressure correction equation (Eq. 2.63) to find the pressure and velocity corrections.

6. Update the pressure and velocity corrections using equations 2.64 and equation 2.65 and repeat till convergence.

The biggest assumption made in this procedure was the neglection of the term $\sum_{f \in \mathcal{N}(e)} a_f^u u_f'$ in equation 2.54. This term contains the information about velocity corrections of the neighboring nodes. However, since the neglected terms are only used in the iterative pressure velocity correction algorithm, it does not have an effect on the final solution. It does however have an effect on the rate of convergence of the iterative procedure and occasionally whether the algorithm converges at all.

SIMPLE family of algorithms

There are many variants of the widely used SIMPLE algorithm [1, 17, 21–24] that can improve the performance.

SIMPLEC   Instead of neglecting the term $\sum_{f \in \mathcal{N}(e)} a_f^u u_f'$ in equation 2.54, it can be approximated to improve the convergence rate [24]. The velocity correction at any node P is assumed to be the weighted average of the corrections at the neighboring points. For example for the velocity correction u_e',

$$u_e' \approx \frac{\sum_{f \in \mathcal{N}(e)} a_f^u u_f'}{\sum_{f \in \mathcal{N}(e)} a_f^u}. \tag{2.66}$$

Thus the relation between velocity and pressure correction will be

$$u_e' = -\frac{|\Omega|_e}{a_e^u + \sum_{f \in \mathcal{N}(e)} a_f^u} \frac{\partial p'}{\partial x_e}. \tag{2.67}$$

This leads to a smaller term being neglected in the pressure correction, thus improving the speed of convergence. There is only a modification of the coefficients of the pressure correction equation compared to the SIMPLE algorithm and the sequence of operations remains the same. However, the speed of convergence in SIMPLE can be recovered [3] if the under-relaxation parameter for the pressure correction, α_p is set to

$$\alpha_p = 1 + \frac{\sum_{f \in \mathcal{N}(e)} a_f^u}{a_e^u}. \tag{2.68}$$

PISO   In this variation, an additional correction step is employed [23]. The same sequence of operations outlined for SIMPLE is followed, and the corrected velocity and pressure field is used to explicitly solve the momentum equations to find a new estimate of the velocity field. Based on the explicit velocity solution, the mass imbalance is computed once again and the pressure correction is solved to find a newer estimate of velocity. The explicit solution recovers a portion of the neglected terms and aids in increasing the rate of convergence.

SIMPLER In SIMPLER [17], the pressure correction is used to correct the velocity field only. This algorithm arose out of an observation that the pressure correction equation is very good at correcting the velocities that satisfies the continuity equation but rather poor at giving a converged pressure field. A new equation for the pressure field is found without neglecting any terms as it was done in SIMPLE. The new pressure equation is analogous to equation 2.61 but contains the pressure itself instead of the pressure corrections and also contains additional terms that involve the neighbors on the right hand side. Equation 2.50 can be written as

$$u_e = -\frac{\sum_{f \in \mathcal{N}(e)} a_f^u u_f}{a_e^u} - \frac{|\Omega|_e}{a_e^u} \frac{\partial p}{\partial x_e}.$$

Define a pseudo-velocity, $\hat{u}_e$ as

$$\hat{u}_e = -\frac{\sum_{f \in \mathcal{N}(e)} a_f^u u_f}{a_e^u},$$

which then gives

$$u_e = \hat{u}_e - \frac{|\Omega|_e}{a_e^u} \frac{\partial p}{\partial x_e}. \tag{2.69}$$

Analogous equations can be written for the other components of the velocity. Substituting the expressions for velocity into the discretized continuity equation around a node P (figure 2.8)

$$-\left(\rho \frac{|\Omega|_e}{a_e^u} \frac{\partial p}{\partial x_e} \Delta S_e - \rho \frac{|\Omega|_w}{a_w^u} \frac{\partial p}{\partial x_w} \Delta S_w + \rho \frac{|\Omega|_n}{a_n^v} \frac{\partial p}{\partial y_n} \Delta S_n - \rho \frac{|\Omega|_s}{a_s^v} \frac{\partial p}{\partial y_s} \Delta S_s \right)$$
$$= -\left((\rho \hat{u}_e \Delta S_e) - (\rho \hat{u}_w \Delta S_w) + (\rho \hat{v}_n \Delta S_n) - (\rho \hat{v}_s \Delta S_s) \right). \tag{2.70}$$

This equation can be solved directly for the pressure if the velocity is available. The algorithm is

1. Guess a velocity field.

2. Calculate the coefficients for the pressure equation 2.70.

3. Solve equation 2.70 to find the pressure field.

4. Treat the new pressure field as the estimated pressure field p^* and solve the momentum equations to find u_e^* and other estimated velocities.

5. Compute the mass imbalance at the faces $\dot{m}_f^*$.

6. Solve the pressure correction equation 2.63.

7. Use the pressure correction equation to solve for the velocity corrections only.

8. Use the updated velocity and repeat from step 2.

Another advantage of SIMPLER is that unlike SIMPLE. no guessed pressure field is required. However, each iteration requires the solution of two Poisson problems which can be more computationally intensive. This method unlike SIMPLE provides the correct pressure field if the initial velocity field is already correct.

Disadvantages of the staggered grid arrangement

Staggering the storage of pressure and velocities has been critical in overcoming the pressure velocity coupling. However, there are many disadvantages of using a staggered grid in problems of practical interest. On Cartesian grids, the staggering procedure is relatively straightforward but increases the memory requirements as the locations of the cell faces have to be stored along with the cell centroids. On non-Cartesian grids, like curvilinear grids, the staggering of velocities itself is very challenging. The cell surfaces are not aligned with the velocity components. One alternative is to store all the components at all cell faces but this comes at a significant cost. The use of curvilinear velocity components can overcome some of the difficulties in storage but the momentum equations in curvilinear coordinates are more complicated. On unstructured grids, which are increasingly common in many practical applications, there is no obvious staggering direction. Storing all velocity components at all faces will allow the use of staggered grid approach for unstructured grids but as mentioned above, it comes with a significant increase in complexity and computational cost. Additionally, Rhie and Chow [25] note that while storing all the components of the velocity at all faces for non-Cartesian grids removed the checkerboard problem in the direction of the grids, the pressure oscillations remained in the diagonal direction.

2.5.2. Collocated grids

The remedy to the disadvantages of the staggered grid arrangement is to use the collocated arrangement for pressure and velocities but to store or compute the mass fluxes at cell faces. However, as evidenced earlier the use of collocated grids could result in oscillatory pressure fields. The main problem with the collocated grid was that the pressure gradient approximation relied on alternate nodes and was insensitive to variation in consecutive nodes. The pressure gradient is typically approximated using a second order central scheme which would yield an approximation that depends on nodes W and E when discretizing the momentum equation for node P. This approximation can be considered as a $2\Delta X$ approximation of the pressure gradient and is insensitive to a ΔX variation of pressure. Staggered grids overcome this deficiency by storing the velocities on faces and creating a ΔX approximation of pressure gradient (see figure 2.9).
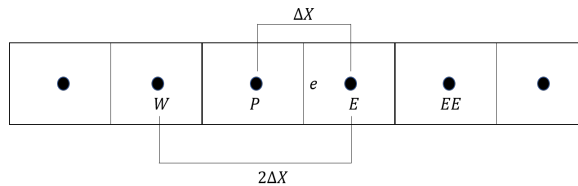
Figure 2.9: Pressure gradient approximation stencils in Rhie Chow interpolation.

Rhie and Chow [25] introduced an interpolation scheme that allowed the use of pressure correction methods like SIMPLE to be used on collocated grids as well. A dissipation term representing the difference between the two ways of computing

the pressure gradient, the ΔX and the $2\Delta X$ approximations, is added to the linear interpolation of the velocity at cell faces. The velocity at the face, $\bar{u}_e$, at a face e in figure 2.9 is approximated by a linear interpolation as

$$\bar{u}_e = \lambda_P u_P + \lambda_E u_E, \tag{2.71}$$

where λ_P and λ_E are the weighting factors for nodes P and E respectively. The new interpolation scheme from Rhie and Chow modify the velocity at the face as

$$u_e = \bar{u}_e + D_f \left(\left(\frac{\partial p}{\partial x} \right)_e - \overline{\left(\frac{\partial p}{\partial x} \right)}_e \right). \tag{2.72}$$

The term D_f was introduced earlier in equation 2.62. The $\bar{u}_e$ is the linearly interpolated velocity in equation 2.71. The first term within the brackets is the ΔX approximation of the pressure gradient and the second term represents the $2\Delta X$ approximation of the pressure gradient. On a uniform $1D$ grid, these derivatives are

$$\left(\frac{\partial p}{\partial x} \right)_e = \frac{p_E - p_P}{\Delta X}, \tag{2.73}$$

and

$$\overline{\left(\frac{\partial p}{\partial x} \right)}_e = \frac{1}{2} \left(\frac{p_E - p_P}{\Delta X} + \frac{p_P - p_W}{\Delta X} \right) = \frac{p_E - p_W}{2\Delta X}. \tag{2.74}$$

The new interpolation scheme, commonly referred to as either the Rhie-Chow interpolation or the momentum interpolation, allows the use of the SIMPLE like algorithms on collocated grids. The solution algorithm is very similar to the staggered grid.

1. Guess the pressure field p^*.

2. Solve the momentum equations like equation 2.52 to find the estimated velocities u_i^*.

3. Use the Rhie-Chow interpolation to compute the velocities at the cell faces.

4. Find the mass flux at the faces m_f^* using the cell face velocities.

5. Assemble the pressure correction equation based on the mass fluxes and the momentum equation.

6. Solve the pressure correction equation (equation 2.63) to find the pressure and velocity corrections.

7. Update the pressure and velocity corrections and repeat till convergence.

Other algorithms based on SIMPLE, like SIMPLEC, SIMPLER and PISO can also be used as long as the mass flux imbalance is computed using the new interpolation scheme.

2.6. Turbulent flows

Most flow problems encountered in practical situations, including the flow past wind turbines, are turbulent in nature. Turbulent flows are highly unsteady with the velocity fields fluctuating rapidly in all three dimensions (usually around a mean value). A laminar flow becomes turbulent due to an instability in the flow that gets amplified, either due to external factors or the flow itself, when above a certain critical Reynolds number. Discussing the nature of turbulence is beyond the scope of this thesis and the emphasis of this section is on modeling the effects of turbulence.

Despite the vastly different natures of laminar and turbulent flows, the Navier Stokes equations presented earlier in equations 2.1, 2.2, 2.3 and 2.4 are valid for both situations. However, turbulent flows involve instantaneous fluctuations in three dimensions, rapid mixing and appears random. Unlike laminar flows, there is an additional mechanism for energy transfer which is usually described as an energy cascade. This process describes the transfer of energy from very large eddies (zones of recirculating motion of the fluid) to smaller ones in a cascading process till the size of the eddies are small enough for viscous dissipation to become relevant.

In order to accurately model the energy cascade process, the numerical methods used must resolve all the scales down to the smallest ones, known as the Kolmogrov scales [26]. The smallest length scales are characterized by the viscosity of the fluid and for flows at high Reynolds numbers such as the flow around a wind turbine, these scales are extremely small. An accurate representation of eddies at this scale would require enormous computational resources. The computational cost of such simulations can be shown to be proportional to approximately Re^3. Such simulations where all the scales of the flow are resolved are known as Direct Numerical Simulations (DNS). Due to the prohibitively expensive computational requirements, DNS can be carried out only for low Reynolds number flows under relatively simple conditions and cannot be applied to problems of practical interest. Thus, approximate approaches that can model the effect of turbulence within reasonable computational costs are sought.

One such approximate solution is to use the Large Eddy Simulations (LES) approach. As the name implies, the larger eddies in a turbulent flow are resolved numerically whereas the statistical nature of the energy cascade process, especially at smaller length scales, is used to model the smaller scales. This lifts some of the more restrictive computational requirements of DNS while still maintaining relatively high accuracy. Despite the still significant computational requirements the use of LES for practical engineering problems is gaining increasing significance.

However, the most commonly used approach to model turbulent flows is based on the solutions of Reynolds Averaged Navier Stokes equations (RANS). The turbulent flow field is split into a time averaged mean flow and a fluctuating component. The Reynolds averaging process brings the computational requirements for most practical problems to a manageable level while maintaining sufficient accuracy. Combinations of RANS and LES like Detached Eddy Simulation (DES) and Delayed Detached Eddy Simulation (DDES) are becoming increasingly common for practical engineering applications as well.

The Reynolds averaging and the resulting equations will be described in detail

in the following section.

2.6.1. Reynolds averaging

Any general variable ϕ in a turbulent flow is three dimensional and varies with time (figure 2.10). ϕ is assumed to be composed of a mean flow component, $\overline{\phi}$ and a
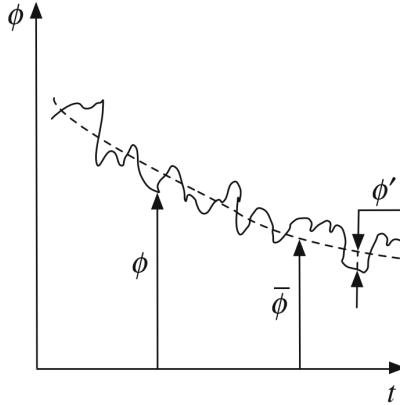
Figure 2.10: Fluctuating and mean variable components for a general variable ϕ [1].

fluctuating component ϕ' as

$$\phi(x_i, t) = \overline{\phi}(x_i, t) + \phi'(x_i, t). \tag{2.75}$$

The mean flow component can be computed using the Reynolds averaging process [1]. Three different approaches are possible

Time averaging The mean flow component is found as

$$\overline{\phi}(x_i, t) = \frac{1}{T} \int_t^{t+T} \phi(x_i, t) dt. \tag{2.76}$$

Here T is the averaging interval which must be large compared to the turbulent fluctuations. This approach is suitable when the mean flow is steady. However, for unsteady mean flows the interval of averaging, T, should be chosen such that it is much larger than the time scale of the fluctuations but smaller than the variation of mean flow. This is admittedly difficult and different type of averaging can be used instead.

Ensemble averaging The ensemble average is defined as

$$\overline{\phi}(x_i, t) = \frac{1}{N} \sum_N \phi(x_i, t), \tag{2.77}$$

where N is the number of different measurements of $\phi(x_i, t)$. For an accurate ensemble average, the value of N must be as large as possible which is not always practical.

Spatial averaging   A spatial average represents the average of a quantity over a space interval Ω.

$$\overline{\phi}(x_i, t) = \frac{1}{\Omega} \int_{\Omega} \phi(x_i, t) dx_i. \tag{2.78}$$

This averaging is typically suitable for homogeneous turbulence.

Properties of Reynolds averaging

For any two variables ϕ and ψ which can be decomposed into a mean flow component $(\overline{\phi}, \overline{\psi})$ and a fluctuating component (ϕ', ψ'), the following rules apply.

$$\begin{aligned}
\overline{\overline{\phi}} &= \overline{\phi}, \\
\overline{\phi'} &= 0, \\
\overline{\frac{\partial \phi}{\partial x_i}} &= \frac{\partial \overline{\phi}}{\partial x_i}, \\
\overline{\phi + \psi} &= \overline{\phi} + \overline{\psi}, \\
\overline{\overline{\phi}\psi'} &= 0, \\
\overline{\phi\psi} &= \overline{\phi}\overline{\psi} + \overline{\phi'\psi'}.
\end{aligned} \tag{2.79}$$

2.6.2. Incompressible RANS equations

The Reynolds averaging can be applied to the velocities, u_i and pressure, p to give

$$\begin{aligned}
u_i &= \overline{u}_i + u'_i, \\
p &= \overline{p} + p'.
\end{aligned} \tag{2.80}$$

Substituting these into the continuity equation (equation 2.7) and taking the time average gives

$$\frac{\partial \overline{u}_i}{\partial x_i} = 0. \tag{2.81}$$

This equation is fundamentally the same as the equation 2.7. The divergence condition now applies only to the mean flow components of velocity. Adopting a similar procedure to the momentum equations (equation 2.8) and taking the time average gives

$$\frac{\partial \overline{u}_i}{\partial t} + \frac{\partial (\overline{u_i u_j})}{\partial x_j} = -\frac{\partial \overline{p}}{\partial x_i} + \frac{1}{Re} \left(\frac{\partial^2 \overline{u}_i}{\partial x_j \partial x_j} \right). \tag{2.82}$$

The unsteady term, pressure gradient term and the viscous diffusion term simplify to a form that is similar to equation 2.8 but due to the non-linearity of the advective

term, special attention is required. Using the property of Reynolds averaging for a product, the advective term can be separated into two components as

$$\frac{\partial \overline{u}_i}{\partial t} + \frac{\partial (\overline{u}_i \overline{u}_j)}{\partial x_j} + \frac{\partial (\overline{u'_i u'_j})}{\partial x_j} = -\frac{\partial \overline{p}}{\partial x_i} + \frac{1}{Re}\left(\frac{\partial^2 \overline{u}_i}{\partial x_j \partial x_j} \right). \tag{2.83}$$

The additional term involving the product of the fluctuating velocity components is unknown and thus the incompressible RANS equations are not closed. This additional term is known as the Reynolds stress tensor and equation 2.83 is usually written as

$$\frac{\partial \overline{u}_i}{\partial t} + \frac{\partial (\overline{u}_i \overline{u}_j)}{\partial x_j} = -\frac{\partial \overline{p}}{\partial x_i} + \frac{1}{Re}\left(\frac{\partial^2 \overline{u}_i}{\partial x_j \partial x_j} \right) - \frac{\partial (\overline{u'_i u'_j})}{\partial x_j}. \tag{2.84}$$

For the commonly used eddy viscosity models, the Reynolds stress tensor is modeled based on the Boussinesq hypothesis [26], where an analogy between the viscous stresses and Reynolds stresses are made. For a Newtonian fluid, the viscous stresses are a linear function of the mean velocity gradients (mean rate of strain) with the constant of proportionality being the dynamic viscosity of the fluid

$$\tau = \mu \left[\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) - \frac{2}{3}\delta_{ij}\frac{\partial u_i}{\partial x_i} \right]. \tag{2.85}$$

Similarly, the Reynolds stresses are assumed to be a linear function of the mean velocity gradients and the proportionality constant being the newly defined turbulent eddy viscosity, μ_t, as

$$\tau^R = -\frac{\partial (\overline{u'_i u'_j})}{\partial x_j} = \mu_t \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right) - \frac{2}{3}\delta_{ij}\left(\rho k + \mu_t \frac{\partial u_i}{\partial x_i} \right). \tag{2.86}$$

Here ρ is the density and k is known as the turbulent kinetic energy defined as

$$k = \frac{1}{2}\overline{u'_i u'_i}. \tag{2.87}$$

For incompressible flows the divergence of the mean flow velocity is zero and the turbulent kinetic energy term is absorbed into the pressure or simply neglected. This leaves only the turbulent eddy viscosity as the new unknown.

Turbulent eddy viscosity models
Using the Boussinesq hypothesis, instead of the unknown Reynolds stress tensor now only a single unknown turbulent eddy viscosity remains. This eddy viscosity can be expressed as a function of a velocity scale (V) and a length scale (l) based on dimensional analysis as

$$\mu_t = C_\mu \rho l V, \tag{2.88}$$

where C_μ is a dimensionless constant. Different eddy viscosity models have been proposed to find these velocity and length scales. The most popular class of these models are the one and two equation turbulence models. An example of a one equation turbulence model is the Spalart-Allmaras model [27] where a scalar transport

equation similar to equation 2.9 is written for a new variable $\tilde{\nu}$. The most common examples of two equation models are the k-ω model from Wilcox [28, 29], the k-ϵ model from Jones and Launder [30, 31] and the k-ω SST model from Menter [32].

Presently, the one equation SA turbulence model and the two equation k-ω SST turbulence model are available in SU2.

2.6.3. Solution procedure

Previously, the solution procedure for the incompressible Navier Stokes equations on a collocated grid was described in section 2.5.2. By solving the turbulence equations in a segregated manner with the flow equations, the solution algorithm does not change significantly. The algorithm for solving the incompressible RANS equations is

1. Guess the pressure field p^*.

2. Solve the momentum equations like equation 2.52 to find the estimated velocities u_i^*.

3. Use the Rhie-Chow interpolation to compute the velocities at the cell faces.

4. Find the mass flux at the faces m_f^* using the cell face velocities.

5. Assemble the pressure correction equation based on the mass fluxes and the momentum equation.

6. Solve the pressure correction equation (equation 2.63) to find the pressure and velocity corrections.

7. Update the pressure and velocity corrections and update the mean flow gradients.

8. Solve the turbulence model equations to find the eddy viscosity μ_t to be used in the next iteration.

9. Repeat till convergence.

More details about the models and their implementation will be presented in the next chapter.

References

[1] F. Moukalled, L. Mangani, and M. Darwish, *Fluid Mechanics and Its Applications The Finite Volume Method in Computational Fluid Dynamics An Advanced Introduction with OpenFOAM® and Matlab®*.

[2] C. Hirsch, *Numerical computation of internal and external flows* (Wiley, 1997).

[3] J. H. Ferziger, M. Perić, and M. Peric, *Vasa*, 3rd ed. (Springer, 2002) p. 423.

[4] R. J. LeVeque *et al.*, *Finite volume methods for hyperbolic problems*, Vol. 31 (Cambridge university press, 2002).

[5] G. D. van Albada, B. van Leer, and J. Roberts, W. W., *A comparative study of computational methods in cosmic gas dynamics,* Astronomy and Astrophysics **108**, 76 (1982).

[6] H. A. Van der Vorst, *Iterative Krylov methods for large linear systems*, 13 (Cambridge University Press, 2003).

[7] G. Strang, *Introduction to linear algebra*, Vol. 3 (Wellesley-Cambridge Press Wellesley, MA, 1993).

[8] Y. Saad and M. H. Schultz, *Gmres: A generalized minimal residual algorithm for solving nonsymmetric linear systems,* SIAM Journal on scientific and statistical computing **7**, 856 (1986).

[9] D. Kwak and C. Kiris, *Success and Challenges in Incompressible Flow Simulation.* (AIAA, 2003).

[10] D. Kwak, C. Kiris, and C. S. Kim, *Computational challenges of viscous incompressible flows,* Computers and Fluids **34**, 283 (2005).

[11] W. Shyy, *Elements of Pressure-Based Computational Algorithms,* Advances in Heat Transfer **24**, 191 (1994).

[12] T. D. Economon, *Simulation and adjoint-based design for variable density incompressible flows with heat transfer,* AIAA Journal , 1 (2019).

[13] W. Barsukow, P. V. Edelmann, C. Klingenberg, F. Miczek, and F. K. Röpke, *A numerical scheme for the compressible low-mach number regime of ideal fluid dynamics,* Journal of Scientific Computing **72**, 623 (2017).

[14] E. Turkel, *Preconditioned methods for solving the incompressible and low speed compressible equations,* Journal of computational physics **72**, 277 (1987).

[15] C. A. Park and D. Kwak, *On the Method of Pseudo Compressibility for Numerically Solving Incompressible Flows.* (AIAA, 1984).

[16] A. J. Chorin, *A numerical method for solving incompressible viscous flow problems,* Journal of computational physics **2**, 12 (1967).

[17] S. Patankar, *Numerical Heat Transfer and Fluid Flow* (CRC Press, 1980).

[18] F.H.Harlow and J.E.Welch, *Numerical Calculation of Time-Dependent Viscous Incompressible Flow with Free Surface,* Physics of Fluids **8** (1965).

[19] S. V. Patankar and D. B. Spalding, *A calculation procedure for heat, mass and momentum transfer in three-dimensional parabolic flows,* International Journal of Heat and Mass Transfer (1972).

[20] L. Caretto, A. Gosman, S. Patankar, and D. Spalding, *Two calculation procedures for steady, three-dimensional flows with recirculation,* in *Proceedings of the third international conference on numerical methods in fluid mechanics* (Springer, 1973) pp. 60–68.

[21] J. Y. Murthy and S. Mathur, *Periodic flow and heat transfer using unstructured meshes,* International Journal for Numerical Methods in Fluids **25**, 659 (2002).

[22] H. Xiao, J. Wang, Z. Liu, and W. Liu, *A consistent SIMPLE algorithm with extra explicit prediction — SIMPLEPC,* International Journal of Heat and Mass Transfer **120**, 1255 (2018).

[23] R.-H. Yen and C.-H. Liu, *Enhancement of the simple algorithm by an additional explicit corrector step,* Numerical Heat Transfer, Part B: Fundamentals **24**, 127 (1993).

[24] J. P. Van Doormaal and G. D. Raithby, *Enhancements of the simple method for predicting incompressible fluid flows,* Numerical heat transfer **7**, 147 (1984).

[25] C. M. Rhie and W. L. Chowt, *Numerical Study of the Turbulent Flow Past an Airfoil with Trailing Edge Separation,* AIAA Journal **21** (1983), 10.2514/3.8284.

[26] S. B. Pope, *Turbulent Flows* (Cambridge University Press, 2000).

[27] P. Spalart and S. Allmaras, *A one-equation turbulence model for aerodynamic flows,* in *30th aerospace sciences meeting and exhibit* (1992) p. 439.

[28] D. C. Wilcox, *Reassessment of the scale-determining equation for advanced turbulence models,* AIAA journal **26**, 1299 (1988).

[29] D. C. Wilcox, *Formulation of the k-ω turbulence model revisited,* AIAA journal **46**, 2823 (2008).

[30] B. E. Launder and B. I. Sharma, *Application of the energy-dissipation model of turbulence to the calculation of flow near a spinning disc,* Letters in heat and mass transfer **1**, 131 (1974).

[31] W. Jones and B. E. Launder, *The prediction of laminarization with a two-equation model of turbulence,* International Journal of Heat and Mass Transfer **15**, 301 (1972).

[32] F. R. Menter, *Two-equation eddy-viscosity turbulence models for engineering applications,* AIAA journal **32**, 1598 (1994).

3

Pressure based incompressible solver in SU2

The previous chapter described the incompressible flow equations and the approach to solve them numerically. This chapter describes the implementation of those numerical methods in SU2. The implementation details of the different boundary conditions, under-relaxation techniques, turbulence models will also be discussed.

The general structure of the partial differential equation (PDE) solved in SU2 is of the form [1]

$$\frac{\partial U}{\partial t} + \frac{\partial F^c}{\partial x_i} - \frac{\partial F^v}{\partial x_i} = Q \quad \text{in} \quad \Omega, \quad t > 0, \tag{3.1}$$

where U is the vector of state variables, F_i^c are the convective fluxes, F_i^v are the viscous fluxes and Q is a source term. In the pressure based solver the momentum equations and the pressure correction equation are solved sequentially. The momentum equation remains in the same form as equation 3.1 and the pressure correction equation is derived based on a combination of the momentum and continuity equations as described in the previous chapter.

3.1. Momentum equation

For the momentum equations, the terms in equation 3.1 are

$$U_i = \rho u_i, \quad F_i^c = \rho u_i u_j, \quad F_i^v = \tau_{ij}, \quad Q = -\frac{\partial p}{\partial x_i} \tag{3.2}$$

where u_i are the components of the velocity vector, ρ is the density, p is the gauge pressure ($p - p_{ref}$, where p_{ref} is a reference pressure) and the viscous stresses for incompressible flow are

$$\tau_{ij} = \mu_{tot} \left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i} \right). \tag{3.3}$$

The total viscosity coefficient, μ_{tot} is the sum of the dynamic viscosity μ_{dyn} and turbulent eddy viscosity μ_{tur}, which is computed via a turbulence model.

3.1.1. Spatial discretization

The spatial discretization of equation 3.1 is performed on an edge based dual grid using the finite volume approach [2–4]. The discretization procedure follows along the same lines as described in section 2.2.1. The control volumes are constructed using a median-dual (vertex-based) scheme [1, 5]. The control volume faces are created exactly midway between the adjacent nodes. An illustration of the construction of the dual cell is shown in figure 3.1. The solution is computed at the nodes shown at the intersection of the solid grid lines.
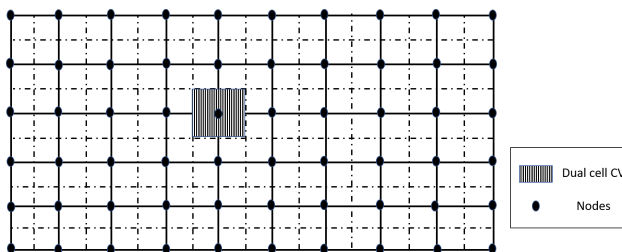
Figure 3.1: Dual cell structure on a 2D domain.

Integrating equation 3.1 over one control volume (CV) with a volume of Ω gives,

$$\int_\Omega \frac{\partial U}{\partial t} d\Omega + \int_\Omega \frac{\partial}{\partial x_i}(F_i^c - F_i^v)d\Omega = -\int_\Omega \frac{\partial P}{\partial x_i}d\Omega. \tag{3.4}$$

Using the divergence theorem on the convective and viscous flux terms results in

$$\int_\Omega \frac{\partial U}{\partial t} d\Omega + \int_{\partial\Omega} (F_i^c - F_i^v)n_i dS = -\int_\Omega \frac{\partial P}{\partial x_i}d\Omega,$$

$$\int_\Omega \frac{\partial U}{\partial t} d\Omega + R(U) = -F^p, \tag{3.5}$$

where $R(U)$ is the residual vector consisting of the discretized convective and viscous fluxes, F^c and F^v. The pressure gradient is treated as a source term and its discretized form, F^p, is found using the mid point integration rule in the CV and is given by

$$\int_\Omega \frac{\partial P}{\partial x_i}d\Omega \approx |\Omega|\frac{\partial P}{\partial x_i} = F^p. \tag{3.6}$$

Discretization of the viscous term   For the dual cell, the surface integral of the viscous fluxes can be transformed into a summation along the CV faces.

$$\int_{\partial\Omega} F_i^v n_i dS \approx \sum_{\partial\Omega} \tau_{ij} n_i \Delta S. \tag{3.7}$$

A central scheme is used to find τ_{ij} at the face f. For the nodes P and F separated across the face f (figure 3.2),

$$\tau_{ij}|_f = \frac{1}{2}\left(\tau_{ij}|_P + \tau_{ij}|_F\right). \tag{3.8}$$

A correction is applied to account for the non orthogonality of the mesh. For a general variable ϕ the derivative in the normal direction is evaluated as

$$\left(\frac{\partial \phi}{\partial n}\right)_f = \frac{\phi_F - \phi_P}{d_{PF}} + \frac{1}{2}\left(\left.\frac{\partial \phi}{\partial x_i}\right|_P + \left.\frac{\partial \phi}{\partial x_i}\right|_F\right)(n_i - \alpha_f s_i), \tag{3.9}$$

where n_i is the face normal, s_i is the normalized vector connecting the cell centers P and F across the face f, d_{PF} is the distance between the nodes P and F and $\alpha_f = s_i n_i$. No correction is applied for the boundary elements. The gradients at the
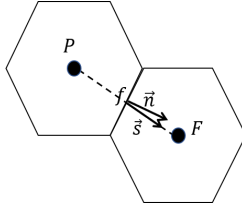
Figure 3.2: Nodes P and F.

cell centers P and F can be computed using either the Green-Gauss or the Least Squares method [2].

Discretization of the convective term The convective fluxes are discretized using a standard upwind scheme and second order accuracy is achieved via reconstruction of variables at the cell faces as described in section 2.2.1. The discretized form of the convective term of the momentum equations is

$$\int_{\partial\Omega} F_i^c n_i dS \approx \sum_{\partial\Omega}(\rho u_i n_i)u_j \Delta S = \sum_{\partial\Omega}\dot{m}_f u_j, \tag{3.10}$$

where $\dot{m}_f$ is the mass flux across each face f along the boundary $(\partial\Omega)$ of the control volume, n_i is the unit outward normal vector of face f and u_j are the velocity components. For instance, on a $1D$ domain (see figure 3.3a) the discretized form of the convective term is

$$\sum_{\partial\Omega}\dot{m}_f u_j = (\dot{m}u)_e + (\dot{m}u)_w. \tag{3.11}$$

and on a 2D domain (see figure 3.3b) the discretized form of the convective term is

$$\sum_{\partial\Omega}\dot{m}_f u_j = (\dot{m}u_j)_e + (\dot{m}u_j)_w + (\dot{m}u_j)_n + (\dot{m}u_j)_s. \tag{3.12}$$

(a) $1D$ control volume around a node P.   (b) $2D$ control volume around a node P.
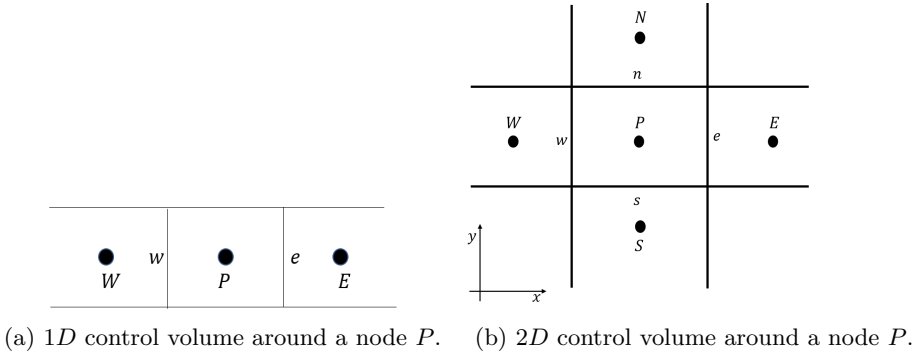
Figure 3.3: Control volume around a node P in (a) $1D$ and (b) $2D$.

Here e,w,n and s represent the east, west, north and south face of the CV respectively and u and v are the horizontal and vertical components of velocity. The velocity at the face is reconstructed from the upwind direction which is determined based on the direction of mass flux at each face. For example, for the face e, let the neighboring node be E. The direction is found as follows. First compute the mass flux across the face, $\dot{m}_e$

$$\dot{m}_e = \frac{1}{2}\rho(u_{i,P} + u_{i,E})n_i.$$

Define two temporary quantities,

$$\dot{m}_P = \frac{1}{2}(\dot{m}_e + |\dot{m}_e|),$$
$$\dot{m}_E = \frac{1}{2}(\dot{m}_e - |\dot{m}_e|).$$

The upwind direction can then be found as,

$$dir_P = \left\lfloor \left| \frac{\dot{m}_P}{|\dot{m}_e|} \right| \right\rceil,$$
$$dir_E = \left\lfloor \left| \frac{\dot{m}_E}{|\dot{m}_e|} \right| \right\rceil.$$

Here $\lfloor \rceil$ represent rounding to the nearest integer. Finally, the velocity at the face e can be found as

$$u_{j,e} = (dir_P)u_{j,P} + (dir_E)u_{j,E}. \tag{3.13}$$

This gives a first order upwind approximation which as seen in section 2.2.1 is not very accurate. The velocities at the nodes, $u_{i,P}$ and $u_{i,E}$ can be reconstructed at the face e to obtain a second order upwind scheme. This option is available to the user. SU2 also has different slope limiters to maintain monotonicity of the upwind scheme. The effect of using a slope limiter was also shown in section 2.2.1.

3.1.2. Time integration: Steady state problems

Steady state problems are solved using a pseudo unsteady method. Instead of using an iterative algorithm the solution is marched in time. For steady state problems, the time integration is done using an implicit Euler scheme. Let the solution at a node P at time t^{n+1} be U_i^{n+1}. Using an implicit Euler scheme on equation 3.5,

$$\int_\Omega \frac{\partial U_P^{n+1}}{\partial t} d\Omega + R_P(U^{n+1}) + F_P^{p,n} \approx \frac{\partial U_P^{n+1}}{\partial t} |\Omega|_P + R_P(U^{n+1}) + F_P^{p,n}. \quad (3.14)$$

A backward Euler time integration scheme is used to discretize the time derivative. The final discretized equation using the procedure outlined in section 2.2.1, can be written as

$$\boldsymbol{A}^n \Delta \mathbf{U}^n = \mathbf{B}^n. \quad (3.15)$$

$\boldsymbol{A}^n$ is the matrix of coefficients at a time level t^n like $\mathbf{a}_{P,i}^n$,

$$\boldsymbol{A}^n = \begin{bmatrix} \vdots & & \vdots & & \vdots \\ & \ddots & & \ddots & \\ \cdots & & \mathbf{a}_{P,i}^n & & \cdots \\ & \ddots & & \ddots & \\ \vdots & & \vdots & & \vdots \end{bmatrix},$$

$\Delta \mathbf{U}^n$ is the vector containing the update to the solution at all the nodes at a time level t^n and $\mathbf{B}^n$ is the vector containing the residual and pressure contributions from all the nodes,

$$\Delta \mathbf{U}^n = \begin{pmatrix} \vdots \\ \Delta U_{P,i}^n \\ \vdots \end{pmatrix}, \qquad \mathbf{B}^n = \begin{pmatrix} \vdots \\ -R(U_{P,i}^n) - F_i^p \\ \vdots \end{pmatrix}, \quad i = 1, 2, 3.$$

A local time stepping scheme is used to accelerate the convergence as each cell advances at a suitable local time step. This local time step is calculated as the minimum of the time steps obtained from convective and viscous terms. The steady solution is obtained faster if larger time steps are used. Also, since time accuracy is not desired when running steady state simulations, the largest possible time step that does not cause the solution to diverge is chosen.

$$\Delta t = min(\Delta t_{conv}, \Delta t_{visc}), \quad (3.16)$$

$$\Delta t_{conv} = CFL \frac{|\Omega|}{\lambda_{conv}}, \qquad \Delta t_{visc} = CFL \frac{|\Omega|^2}{\lambda_{visc}}. \quad (3.17)$$

where CFL is a user defined value and

$$\lambda_{conv} = \sum_f |u_{i,f} n_{i,f}| \Delta S, \quad (3.18)$$

is the sum of the magnitude of the projected face velocity across all the faces of the control volume surrounding a node and

$$\lambda_{visc} = \sum_f C \frac{\mu_{tot}}{\rho} \Delta S^2, \tag{3.19}$$

is the viscous spectral radius [1]. Here C is a constant and is set to $C = 0.25$. In order to avoid the possibility of division by zero, the convective term is changed to

$$\lambda_{conv} = \sum_f |(u_f + u_{ref}) \cdot \vec{n}_f| \Delta S, \tag{3.20}$$

where u_{ref} is a reference velocity.

SIMPLE
As explained in chapter 2, the SIMPLE algorithm is used to calculate the velocity and pressure in an iterative manner. However, since a pseudo time stepping scheme is used for steady state problems, the time steps take the place of iterations. Thus starting from a time step t^n, the velocities and pressure at the next time step t^{n+1} is found as follows. The momentum equations are first solved using the pressure, p^n at time t^n to give an estimated velocity field, u_i^*.

$$u_i^* = u_i^n + \Delta u_i, \tag{3.21}$$

where Δu_i is found from the solution of equation 3.15 as $\Delta u_i = \frac{\Delta U_i}{\rho}$. The estimated velocities and pressure are then corrected based on velocity and pressure corrections.

$$u_i^{n+1} = u_i^* + u_i', \tag{3.22}$$

$$p^{n+1} = p^n + p'. \tag{3.23}$$

The pressure and velocity corrections are found by solving a Poisson equation. The Poisson equation is derived based on the continuity equation and the momentum equations. This procedure is described in the following section.

3.2. Continuity equation
So far, only the momentum equations have been used to find an estimate of the velocity based on the existing pressure field. In order to correct the velocity estimations and the pressure, the continuity equation must be used to close the system of equations. However, since the continuity equation does not contain any pressure terms a new equation for pressure has to be obtained. This new equation for pressure is derived starting from the discrete form of the continuity equation for an incompressible flow

$$\int_\Omega \frac{\partial \rho u_i}{\partial x_i} \approx \sum_{\partial \Omega} \rho u_{f,i} n_{f,i} \Delta S = \sum_{\partial \Omega} \dot{m}_f = 0, \tag{3.24}$$

where u_f is the velocity at a face f, ρ is the fluid density and n_i is the unit outward face normal and ΔS is the area. Using a linear interpolation to find this face velocity

on collocated grids leads to the well known checkerboard problem in the pressure as seen in the previous chapter and thus momentum interpolation techniques are used. Previously, the momentum interpolation proposed by Rhie and Chow [6] was given with only a conceptual explanation. The same interpolation can also be seen as equivalent to writing a pseudo momentum equation at every face with the coefficients linearly interpolated from the momentum equations of the neighboring nodes [2]. Thus, in a sense the momentum interpolation mimics the staggered grid approach on collocated grids. This can also be interpreted as adding a third order derivative of the pressure to stabilize the oscillations in the pressure field [3].

3.2.1. Momentum interpolation of velocities

In this section, equation 2.72 is derived using an alternative approach following Moukalled et al. [2] but adapted to the solution procedure used in SU2. The momentum equation in the x direction for a node P (see figure 3.3a) can be written as

$$\mathbf{a}_P^{n,u}\Delta u_P^n + \sum_{C\in\mathcal{N}(P)} \mathbf{a}_{C,u}^{n,u}\Delta u_C^n = -R(u_P^n) - |\Omega|\left(\frac{\partial p^n}{\partial x}\right)_P$$

where $\mathbf{a}_P^{n,u}$ is the coefficient of u, the velocity along the x direction at the node P, $\mathbf{a}_C^{n,u}$ is the coefficient of the same velocity component at a node C which is a neighbor of node P ($C \in \mathcal{N}(P)$), $R(u_P^n)$ is the residual computed explicitly and $|\Omega|$ is the volume of the control volume around node P. Here $\mathcal{N}(P)$ represents the neighbors the node P. Since the density is assumed to be constant, it is absorbed in to the coefficients $\mathbf{a}_P^{n,u}$ and $\mathbf{a}_C^{n,u}$.

The estimate of the velocity at a node P for time level $n+1$ can be written as

$$u_P^* = u_P^n + \Delta u_P^n = u_P^n - \frac{1}{\mathbf{a}_P^{n,u}}\left(R(u_P^n) + \sum_{C\in\mathcal{N}(P)} \mathbf{a}_{C,u}^n \Delta u_C^n + |\Omega|\left(\frac{\partial p^n}{\partial x}\right)_P\right), \quad (3.25)$$

Let $H(u_P^n)$ denote

$$H(u_P^n) = \frac{1}{\mathbf{a}_P^{n,u}}\left(R(u_P^n) + \sum_{C\in\mathcal{N}(P)} \mathbf{a}_{C,u}^n \Delta u_C^n\right).$$

The velocity estimates at a node P at time level t^{n+1} can now be re-written as

$$u_P^* = u_P^n - H(u_P^n) - \frac{|\Omega|}{\mathbf{a}_P^{n,u}}\left(\frac{\partial p^n}{\partial x}\right)_P. \quad (3.26)$$

Since the pressure gradient used so far is only an estimate, the velocities found using this formula are also an estimate and do not yet satisfy the continuity equation and are thus denoted by u_i^*.

Consider another node E, which is a neighbor of node P across the face e in figure 3.3a. The velocities at node E can be written similar to equation 3.26 as

$$u_E^* = u_E^n - H(u_E^n) - \frac{|\Omega|}{\mathbf{a}_E^{n,u}}\left(\frac{\partial p^n}{\partial x}\right)_E. \quad (3.27)$$

Since the momentum interpolation technique mimics the staggered approach where velocities are stored on cell faces, hypothetically at the face e between P and E, the velocities, $u_{f,i}$ can be written as,

$$u_e^* = u_e^n - H(u_e^n) - \frac{|\Omega|}{\mathbf{a}_e^{n,u}} \left(\frac{\partial p^n}{\partial x}\right)_e. \tag{3.28}$$

The coefficients for the hypothetical momentum equation at the face f are assumed to be interpolated linearly from the neighboring nodes P and E as

$$H(u_e^n) = (\lambda_P H(u_P^n) + \lambda_E H(u_E^n)), \tag{3.29}$$

where λ_P and λ_E are the weighting factors for the interpolation. Since a median dual grid is used for discretization, the faces are always midway between the two nodes. Thus, $\lambda_P = \lambda_E = 1/2$. Substituting for $H_{f,i}^n$, $H_{P,i}^n$ and $H_{E,i}^n$ in equation 3.29 from equations 3.28, 3.26 and 3.27 respectively and expanding the pressure source term from equation 3.6, the velocity at a face f after the momentum equation is

$$u_e^* = u_e^n - (\lambda_P u_P^n + \lambda_E u_E^n) + (\lambda_P u_P^* + \lambda_E u_E^*) - \frac{|\Omega|_f}{\mathbf{a}_e^n}\left(\frac{\partial p^n}{\partial x}\right)_e$$
$$+ \left(\lambda_P \frac{|\Omega|_P}{\mathbf{a}_P^n}\left(\frac{\partial p^n}{\partial x}\right)_P + \lambda_E \frac{|\Omega|_E}{\mathbf{a}_E^n}\left(\frac{\partial p^n}{\partial x}\right)_E\right) \tag{3.30}$$

Let $\boldsymbol{D}_P^{n,u}$ denote

$$\boldsymbol{D}_P^{n,u} = \frac{|\Omega|}{\mathbf{a}_P^{n,u}}. \tag{3.31}$$

Using the new notation, equation 3.30 can be written as

$$u_e^* = u_e^n - (\lambda_P u_P^n + \lambda_E u_E^n) + (\lambda_P u_P^* + \lambda_E u_E^*) - \boldsymbol{D}_e^{n,u}\left(\frac{\partial p^n}{\partial x_i}\right)_e$$
$$+ \left(\lambda_P \boldsymbol{D}_P^{n,u}\left(\frac{\partial p^n}{\partial x}\right)_P + \lambda_E \boldsymbol{D}_E^{n,u}\left(\frac{\partial p^n}{\partial x}\right)_E\right). \tag{3.32}$$

The linear interpolation of the pressure gradient terms from nodes P and E are approximated as

$$\left(\lambda_P \boldsymbol{D}_P^{n,u}\left(\frac{\partial p^n}{\partial x}\right)_P + \lambda_E \boldsymbol{D}_E^{n,u}\left(\frac{\partial p^n}{\partial x}\right)_E\right) = \overline{\boldsymbol{D}}_e^{n,u}\overline{\left(\frac{\partial p^n}{\partial x}\right)}_e. \tag{3.33}$$

The terms under the over bar are linearly interpolated. This approximation is second order accurate [2]. Also the coefficient of the pressure gradient at the face is assumed to be the same as the linearly interpolated value i.e.

$$\boldsymbol{D}_e^{n,u} = \overline{\boldsymbol{D}}_e^{n,u}.$$

Additionally, the linearly interpolated terms are written with an over bar as follows

$$(\lambda_P u_P^* + \lambda_E u_E^*) = \overline{u}_e^*,$$
$$(\lambda_P u_P^n + \lambda_E u_E^n) = \overline{u}_e^n.$$

Equation 3.30 can now be written as

$$u_e^* = \overline{u}_e^* - \overline{\boldsymbol{D}}_e^{n,u}\left(\left(\frac{\partial p^n}{\partial x}\right)_e - \overline{\left(\frac{\partial p^n}{\partial x}\right)}_e\right) + (u_e^n - \overline{u}_e^n). \qquad (3.34)$$

Analogously, the y and z components of the velocity can be written as

$$v_e^* = \overline{v}_e^* - \overline{\boldsymbol{D}}_e^{n,v}\left(\left(\frac{\partial p^n}{\partial y}\right)_e - \overline{\left(\frac{\partial p^n}{\partial y}\right)}_e\right) + (v_e^n - \overline{v}_e^n). \qquad (3.35)$$

$$w_e^* = \overline{w}_e^* - \overline{\boldsymbol{D}}_e^{n,w}\left(\left(\frac{\partial p^n}{\partial z}\right)_e - \overline{\left(\frac{\partial p^n}{\partial z}\right)}_e\right) + (w_e^n - \overline{w}_e^n). \qquad (3.36)$$

Effect of pseudo time stepping

Recall the expression for the face velocity based on the momentum interpolation in Eq. 3.34,

$$u_e^* = \overline{u}_e^* - \overline{\boldsymbol{D}}_e^{n,u}\left(\left(\frac{\partial p^n}{\partial x}\right)_e - \overline{\left(\frac{\partial p^n}{\partial x}\right)}_e\right) + (u_e^n - \overline{u}_e^n).$$

The first term on the RHS is the linearly interpolated velocity estimate and the second term is the difference of the pressure gradients. These two terms are identical to the original interpolation proposed by Rhie and Chow [6]. However, the last term arises as a consequence of using the pseudo time stepping scheme in SU2. This term represents the difference between the corrected velocity at the face u_e^n and the linearly interpolated value $\overline{u}_e^n$ from the previous time step. This difference is equivalent to the difference in pressure gradients from the previous time step. Thus, this term is zero at the start of the iteration and for every subsequent iteration, the difference between the two velocities, u_e^n and $\overline{u}_e^n$ is carried over and added to the next iteration. Not considering this term can lead to oscillations in pressure at small time step sizes [7–12].

In addition to this, the coefficient of the the pressure gradient contains the coefficients of the discretized momentum equations $\mathbf{a}_{e,u_i}^n$ which consists of contributions from time and spatial discretization as,

$$\mathbf{a}_{e,u_i}^n = \boldsymbol{a}_{e,u_i}^{t,n} + \boldsymbol{a}_{e,u_i}^{jac,n}.$$

Equation 3.34 can be rewritten as

$$u_e^* = \overline{u_e^*} - \overline{\frac{|\Omega|}{\boldsymbol{a}_{e,u_i}^{t,n} + \boldsymbol{a}_{e,u_i}^{jac,n}}}\left(\left(\frac{\partial p^n}{\partial x}\right)_e - \overline{\left(\frac{\partial p^n}{\partial x}\right)}_e\right) + (u_e^n - \overline{u}_e^n).$$

The contribution from the time discretization, $\boldsymbol{a}_{e,i}^{t,n}$ depends on the size of the time step Δt which can be changed based on the user defined CFL number. Thus, the final solution will be dependent on the external value of CFL which is undesirable. This is also noted in Cubero and Fueyo [11]. In order to eliminate this dependence the following approach can be adopted. A relaxation factor, RC, is introduced and

multiplied to the time discretization contribution, α_t. When this relaxation factor is set to zero, the solution is independent of CFL. It should be noted that this treatment is not derived analytically and can lead to convergence issues. The RC factor can be changed based on convergence behavior.

$$u_e^* = \overline{u_e^*} - \overline{\frac{|\Omega|}{RC a_{e,u_i}^{t,n} + a_{e,u_i}^{jac,n}} \left(\left(\frac{\partial p^n}{\partial x_i} \right)_e - \overline{\left(\frac{\partial p^n}{\partial x_i} \right)_e} \right)} + (u_e^n - \overline{u}_e^n).$$

3.2.2. Pressure correction equation

To derive the pressure correction equation, first an equation for velocity corrections analogous to equation 3.34 is required. After applying the velocity and pressure corrections, the equation 3.26 becomes

$$u_P^{n+1} = u_P^n - H(u_P^{n+1}) - \frac{|\Omega|}{\mathbf{a}_P^{n,u}} \left(\frac{\partial p}{\partial x} \right)_P^{n+1}. \tag{3.37}$$

Subtracting equation 3.37 from equation 3.26 gives an equation for velocity corrections as

$$u_P' = -H(u_P') - \frac{|\Omega|}{\mathbf{a}_P^{n,u}} \left(\frac{\partial p'}{\partial x} \right)_P. \tag{3.38}$$

Following the same derivation steps outlined above to derive the equation for the velocity estimate at face e, u_e^*, a new equation for the velocity correction at a face e between two nodes P and E (figure 3.3b) can be derived as

$$u_e' = \overline{u_e'} - \overline{\mathbf{D}}_e^{n,u} \left(\left(\frac{\partial p'}{\partial x} \right)_e - \overline{\left(\frac{\partial p'}{\partial x} \right)_e} \right). \tag{3.39}$$

Similarly, the equations for the velocity corrections in the other directions can be written as

$$v_e' = \overline{v_e'} - \overline{\mathbf{D}}_e^{n,v} \left(\left(\frac{\partial p'}{\partial y} \right)_e - \overline{\left(\frac{\partial p'}{\partial y} \right)_e} \right). \tag{3.40}$$

$$w_e' = \overline{w_e'} - \overline{\mathbf{D}}_e^{n,w} \left(\left(\frac{\partial p'}{\partial z} \right)_e - \overline{\left(\frac{\partial p'}{\partial z} \right)_e} \right). \tag{3.41}$$

As before, the terms under the overbar are linearly interpolated. Before deriving the pressure correction equation, a new notation is introduced for the sake of simplicity.

$$\overline{S}_{f,x}^n = \overline{\mathbf{D}}_f^{n,u} n_{f,x} \Delta S, \quad \overline{S}_{f,y}^n = \overline{\mathbf{D}}_f^{n,v} n_{f,y} \Delta S \quad \text{and} \quad \overline{S}_{f,z}^n = \overline{\mathbf{D}}_f^{n,w} n_{f,z} \Delta S. \tag{3.42}$$

Here $n_{f,i}$ is the outward facing unit normal of a face f, ΔS is the area of the face and $\overline{\mathbf{D}}_f^{n,u}, \overline{\mathbf{D}}_f^{n,v}$ and $\overline{\mathbf{D}}_f^{n,w}$ are the coefficients of the pressure gradient difference term in the velocity expressions (equations 3.39, 3.40 and 3.41) and is defined in equation 3.31.

Recall the continuity equation in discrete form equation 3.24

$$\sum_f \dot{m}_f = 0.$$

For a $1D$ control volume like the one shown in figure 3.3a, the summation is over the faces $f = e, w$ and for a $2D$ control volume (figure 3.3b, $f = e, w, n, s$. Rewriting the discrete continuity equation in terms of estimated velocity and velocity corrections gives

$$\sum_f \dot{m}_f = \sum_f (\dot{m}_f^* + \dot{m}_f') = 0, \qquad (3.43)$$

where $\dot{m}_f^*$ and $\dot{m}_f'$, the estimate and correction of the mass flux respectively, are computed as

$$\dot{m}_f^* = \rho u_{f,i}^* n_{f,i} \Delta S,$$
$$\dot{m}_f' = \rho u_{f,i}' n_{f,i} \Delta S,$$

Substituting for the velocity estimates (equations 3.34 to 3.36) and corrections (equations 3.39 to 3.41) in the equation 3.24 gives,

$$\sum_f \rho \overline{u_{f,i}'} n_{f,i} \Delta S - \rho \left(\left(\frac{\partial p'}{\partial x_i} \right)_f - \overline{\left(\frac{\partial p'}{\partial x_i} \right)_f} \right) \overline{S}_{f,i}^n = -\sum_f \dot{m}_f^*. \qquad (3.44)$$

Rearranging this equation by moving the linearly interpolated terms to the RHS gives

$$-\sum_f \rho \overline{S}_{f,i}^n \left(\frac{\partial p'}{\partial x_i} \right)_f = -\sum_f \dot{m}_f^* - \overbrace{\sum_f \rho \overline{u_{f,i}'} n_{f,i} \Delta S - \sum_f \rho \overline{S}_{f,i}^n \overline{\left(\frac{\partial p'}{\partial x_i} \right)_f}}^{\text{neglected in SIMPLE}}. \qquad (3.45)$$

As outlined in section 2.5.1, the terms under the over brace on the RHS of equation 3.45 are neglected in the SIMPLE algorithm. The remaining term on the RHS of equation 3.45 is the mass flux that is calculated using the estimated velocities. Thus, the pressure correction is

$$-\sum_f \rho \overline{S}_{f,i}^n \left(\frac{\partial p'}{\partial x_i} \right)_f = -\sum_f \dot{m}_f^*, \qquad (3.46)$$

Equation 3.46 is a discretized Poisson equation for the pressure correction, p', with the uncorrected mass flux, $\sum_f \dot{m}_f^*$, being the source term. This equation has to be solved sequentially with the momentum equations.

Since the pressure correction equation was derived starting from the continuity equation, the summation of the gradients of the pressure correction are also carried out on the same control volume. Across the control volume shown in figure 3.3a, the summation will be over the faces $f = e, w$ and for the control volume in figure 3.3b,

the summation will be over the faces $f = e, w, n, s$. The discretized pressure correction gradient for face e can be approximated as

$$\left(\frac{\partial p'}{\partial x_i}\right)_f \approx \frac{p'_E - p'_P}{d_{PE}} n_{f,i},$$

where d_{PE} is the total distance between the nodes P and E across the face f and $n_{f,i}$ is the outward unit normal of the face f. In order to find the coefficients of the nodal values of pressure correction, p'_P, the term $\overline{S}^n_{f,i}$ must be calculated at each face f. $\overline{S}_{f,i}$ is calculated at a face f using an over-relaxed approach [2]. The over-relaxed approach increases the contribution of the nodes P and E as the grid non-orthogonality increases. $\overline{S}^n_{f,i}$ is split into the orthogonal ($\overline{E}^n_{f,i}$) and non-orthogonal ($\overline{T}^n_{f,i}$) parts as

$$\overline{S}^n_{f,i} = \overline{E}^n_{f,i} + \overline{T}^n_{f,i}. \tag{3.47}$$

The orthogonal contribution is treated implicitly and the non orthogonal contribution is neglected. The pressure correction equation thus becomes

$$-\sum_f \rho \overline{E}^n_{f,i}\left(\frac{\partial p'}{\partial x_i}\right)_f = -\sum_f \dot{m}^*_f,$$

The coefficients of the nodal values of p' calculated using the over-relaxed approach for the nodes P and E across the face e is

$$a^{p'}_E = (-\rho \Delta S)\frac{(\overline{\boldsymbol{D}}^{n,u}_f n_x)^2 + (\overline{\boldsymbol{D}}^{n,v}_f n_y)^2 + (\overline{\boldsymbol{D}}^{n,w}_f n_z)^2}{\overline{\boldsymbol{D}}^{n,u}_f d_{PE,x} + \overline{\boldsymbol{D}}^{n,v}_f d_{PE,y} + \overline{\boldsymbol{D}}^{n,w}_f d_{PE,z}}, \tag{3.48}$$

where $d_{PE,i}$ is the distance vector between nodes P and E. Analogous to the discretization of the viscous terms in the scalar transport equation in section 2.2.1, the coefficients for all the nodal values can be assembled. Thus, the pressure correction equation can be written as a system of linear equations as

$$a^{p'}_P p'_P + \sum_{C \in \mathcal{N}(P)} a^{p'}_C p'_C = -\sum_{f \in \mathcal{N}(P)} \dot{m}^*_f. \tag{3.49}$$

The mass flux on the RHS is computed as the summation of the mass flux across the faces f around the control volume of the node P. The system of equations for the nodal values of the pressure corrections p'_P can be solved using the linear solvers described in chapter 2. No under-relaxation is used for the Poisson solver. A multigrid method can be applied specifically for the Poisson problem to speed up the convergence, especially for unsteady problems.

3.2.3. Pressure and velocity corrections

Finally, based on the solution of the pressure correction equation, the pressure and velocities at a node P can be corrected as

$$p_P^{n+1} = p_P^* + \alpha_p p', \tag{3.50}$$

$$u_P^{n+1} = u_P^* + \boldsymbol{D}_P^{n,u} \left(\frac{\partial p'}{\partial x} \right)_P,$$

$$v_P^{n+1} = v_P^* + \boldsymbol{D}_P^{n,v} \left(\frac{\partial p'}{\partial y} \right)_P, \tag{3.51}$$

$$w_P^{n+1} = w_P^* + \boldsymbol{D}_P^{n,w} \left(\frac{\partial p'}{\partial w} \right)_P.$$

α_p is the under-relaxation factor. There is no need to under-relax the velocity corrections since the pseudo time stepping method, used to solve the momentum equations, acts as an under-relaxation for the velocities. The choice of the pressure under-relaxation will have an effect on the convergence of the system. As seen in chapter 2, the convergence of the SIMPLE algorithm can be accelerated if the pressure under-relaxation is set to

$$\alpha_p = 1 + \frac{\sum_{f \in \mathcal{N}(P)} \mathbf{a}_{f,u_i}}{\mathbf{a}_{P,u_i}}. \tag{3.52}$$

For a steady state solution, this factor can be simplified in terms of the velocity under-relaxation factor, α_v, as

$$\alpha_p = 1 - \alpha_v, \tag{3.53}$$

In order to find α_v, recall the discretized momentum equation in the x direction for a node P is

$$\mathbf{a}_P^{n,u} \Delta u_P^n + \sum_{C \in \mathcal{N}(P)} \mathbf{a}_C^{n,u} \Delta u_C^n = -R(u_P^n) - |\Omega| \frac{\partial p^n}{\partial x}.$$

$\mathbf{a}_P^{n,u}$ contains contributions from the pseudo time stepping and the Jacobian and can be split as

$$\mathbf{a}_P^{n,u} = \boldsymbol{a}_P^{t,n} + \boldsymbol{a}_P^{jac,n}, \tag{3.54}$$

where $\boldsymbol{a}_P^{t,n}$ is the pseudo time stepping contribution and $\boldsymbol{a}_P^{jac,n}$ is the contribution from the Jacobian. Comparing to a typical under-relaxed equation of the form

$$\frac{\boldsymbol{a}_{P,u_i}^{jac,n}}{\alpha_v} \Delta u_{P,i}^n + \sum_{C \in \mathcal{N}(P)} \mathbf{a}_{C,u_i}^n \Delta u_{C,i}^n = -R(u_{P,i}^n) - F_{P,i}^{p,n},$$

the under-relaxation factor in the pseudo time stepping scheme is

$$\alpha_v = \frac{\boldsymbol{a}_{P,u_i}^{jac,n}}{\boldsymbol{a}_{P,u_i}^{t,n} + \boldsymbol{a}_{P,u_i}^{jac,n}}. \tag{3.55}$$

Thus, the optimum value of the pressure under-relaxation is

$$\alpha_p = 1 - \alpha_v = \frac{\boldsymbol{a}_{P,u_i}^{t,n}}{\boldsymbol{a}_{P,u_i}^{t,n} + \boldsymbol{a}_{P,u_i}^{jac,n}}. \tag{3.56}$$

3.3. Boundary conditions

The control volume formed around interior nodes in the vertex based approach is shown in figure 3.1. The node lies in the center of the control volume. However, because of the vertex based approach, the control volumes around the boundary nodes are different. The node lies on the face of the control volume at the physical boundary as shown in figure 3.4a While the discretization for the interior faces

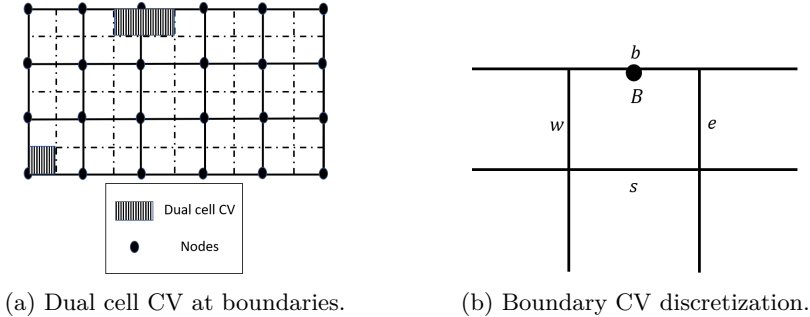(a) Dual cell CV at boundaries.   (b) Boundary CV discretization.

Figure 3.4: Control Volume around boundary nodes.

proceeds as described earlier, the discretization for the boundary face will be presented in this section. Since the boundary node (B) is directly on the boundary face (b) (figure 3.4b), there is no need to use the momentum interpolation to find the velocity at the face, i.e. for any quantity ϕ

$$\phi_b = \phi_B. \tag{3.57}$$

The boundary conditions available are free slip wall, no slip wall, velocity inlet, pressure outlet and symmetry boundaries. The application of each of these boundary conditions for the momentum equations, mass flux computation and pressure correction equation is described below.

3.3.1. Momentum equations

1. Slip wall: This boundary condition specifies a zero normal flux across the boundary (e.g. inviscid wall). For the momentum equations, this is applied as a weak boundary condition with zero flux across the face. The mass flux across the face is also set to zero.

$$\dot{m}_b = 0. \tag{3.58}$$

2. Wall (no-slip): This is a strong boundary condition and is generally used to impose a no slip condition on the velocities at the wall. Since the discretization is vertex-based the boundary node lies on boundary face and thus the velocity boundary condition can be enforced as a Dirichlet boundary condition. The mass flux across the face is set to zero and the velocity at the wall is set to zero or to the specified wall velocity (u_{wall}).

$$\dot{m}_b = 0, \quad u_b = u_{wall}. \tag{3.59}$$

3. Inlet: For a prescribed velocity (u_{in}) at the inlet, the velocity is imposed as a Dirichlet boundary condition similar to the wall boundary. However, the mass flux is not zero but can be easily computed based on the specified velocity u_{in}.

$$\dot{m}_b = \rho u_{i,in} n_i \Delta S_b, \quad u_b = u_{in}. \tag{3.60}$$

4. Outlet: For a specified pressure outlet, a weak boundary condition is applied for the velocities at the outlet. Fully developed flow conditions are assumed at the outlet. Thus, the velocity gradient normal to the outlet surface is zero. Similarly, the mass flux across the face is also computed using the latest estimate of the velocity.

$$\dot{m}_b = \rho u_{i,b} n_i \Delta S_b. \tag{3.61}$$

5. Symmetry: A symmetry boundary does not only imply a zero flux across the face but also a reflection of the solution state across the boundary face. Consequently, a reflected state of the current state is computed and a Neumann boundary condition is applied. The mass flux across the face is set to zero.

6. Far-field: Far-field boundaries are typically used in external flow simulations to denote the free-stream conditions. This is treated as an inlet-outlet type boundary where a Dirichlet condition is used for incoming flow and a Neumann boundary for outgoing flow based on the nature of the flux at the boundary face. The mass flux at the far-field face is computed as

$$\dot{m}_b = \rho u_{i,b} n_i \Delta S_b. \tag{3.62}$$

Depending on the sign of the mass flux, this face is treated as an inlet $(\dot{m}_b < 0)$ or an outlet $(\dot{m}_b > 0)$. Implementation of these two conditions are similar to the inlet and outlet boundary conditions.

7. Periodic: The periodic pair of elements are treated as an internal element by exchanging the flux across the interface. The solution is only computed for the donor node and is transformed back to the receiver node.

3.3.2. Pressure correction equations

If the pressure at a particular boundary is unknown (Euler wall, Wall, Inlet, Symmetry) it is treated as a Neumann boundary and the value of the pressure is updated based on the pressure correction. However, if the value of the pressure is specified (Outlet with a specified pressure), the value of the pressure is fixed and the pressure correction is set to zero as a Dirichlet boundary condition.

3.4. Unsteady problems: Dual time stepping

Unsteady problems are solved with a dual time stepping scheme. The unsteady problem is converted to a steady state problem within each time step which is solved as described previously. A pseudo transient term is added to equation 3.5 as

$$\int_\Omega \frac{\partial U}{\partial \tau} d\Omega + \int_\Omega \frac{\partial U}{\partial t} d\Omega + R(U) = -F^p \rightarrow \int_\Omega \frac{\partial U}{\partial \tau} d\Omega + R^*(U) = 0, \tag{3.63}$$

where τ is the pseudo time variable. The pseudo time term is discretized as explained in section 3.1.2 and the unsteady time term is discretized by a backward Euler scheme. First and second order time integration schemes can be used for the unsteady term. For the first order discretization, $R^*(U)$ is

$$R^*(U) = \frac{U - U^n}{\Delta t}|\Omega| + R(U) + F^{p,n}, \tag{3.64}$$

and for second order,

$$R^*(U) = \frac{3U - 4U^n + U^{n-1}}{2\Delta t}|\Omega| + R(U) + F^{p,n}. \tag{3.65}$$

At the end of the pseudo-steady solution U of equation 3.63 becomes U^{n+1}.

3.5. Moving grids

Recall the general form of equations in SU2 from equation 3.1 is

$$\frac{\partial U}{\partial t} + \frac{\partial F_i^c}{\partial x_i} - \frac{\partial F_i^v}{\partial x_i} = Q \quad \text{in} \quad \Omega, \quad t > 0.$$

3.5.1. Arbitrary Lagrangian Eulerian method

For the Arbitrary Lagrangian and Eulerian (ALE) formulation, the convective term is expressed as

$$F_i^c = \rho(u_i - u_{g,i})u_j, \tag{3.66}$$

where $u_{g,i}$ is the grid velocity. The other terms remain the same. However, the computation of mass flux at the faces of control volumes must also account for the grid movement. Thus, the relative velocity at a face, $u_{rel,f}$, computed using the Rhie-Chow interpolation is

$$u_{rel,f}^* = \overline{u_f}^* - \overline{u}_{f,g} - \overline{\boldsymbol{D}}_f^{n,u}\left(\left(\frac{\partial p'}{\partial x}\right)_f - \overline{\left(\frac{\partial p'}{\partial x}\right)}_f\right) + (u_e^n - \overline{u}_e^n). \tag{3.67}$$

Here $\overline{u}_{f,g}$ is the linearly interpolated grid velocity at the face f.

Rotating reference frame

For steady simulations, in a rotating reference frame with rotation rate Ω_i, the grid velocity can be found as the cross product of the rotation rate vector and the radius vector.

$$u_{g,i} = \epsilon_{ijk}\Omega_i r_j \hat{e}_k. \tag{3.68}$$

The Ω_i is the vector of rotation rate about each of the axis, r_j is the radius vector from the center of rotation, $\hat{e}_k$ is the unit coordinate vector and ϵ_{ijk} is the levi civita tensor. In addition to accounting for grid movement like described above an additional source term is added to the momentum equations

$$Q_{rot} = -\rho\epsilon_{ijk}\Omega_i u_j \hat{e}_k. \tag{3.69}$$

This source term is the cross product of the rotation rate vector, Ω_i, and the velocity vector, u_j.

3.6. Turbulence modeling

Turbulence modeling in SU2 is based on solving the Reynolds Averaged Navier Stokes (RANS) equations. As described in section 2.6.2, the most widely used approach is to use the Boussinesq hypothesis and write the Reynolds stresses in terms of mean flow gradients. This introduces a new unknown, the turbulent eddy viscosity, μ_t. In order to close the system of RANS equations, equations for the turbulent eddy viscosity are solved. Eddy viscosity turbulence models implemented in SU2 are the Spalart-Allmaras (SA) [13] and the Menter Shear Stress Transport (SST) [14] model. These are described in detail in this section.

3.6.1. Spalart-Allmaras (SA)

The SA eddy viscosity model is a one equation model and solves for a scalar variable $\tilde{\nu}$. This scalar is related to the eddy viscosity as

$$\mu_{tur} = \rho \tilde{\nu} f_{v1}. \tag{3.70}$$

Here ρ is the density and f_{v1} is obtained from the turbulence model. Many different versions of the SA turbulence models are available [15]. The general form of the equation in all versions closely resembles the transport equation like equation 2.9. The standard SA model is

$$\frac{\partial \tilde{\nu}}{\partial t} + u_j \frac{\partial \tilde{\nu}}{\partial x_j} = c_{b1}(1 - f_{t2})\tilde{S}\tilde{\nu} - \left[c_{w1}f_w - \frac{c_{b1}}{\kappa^2}f_{t2}\right]\left(\frac{\tilde{\nu}}{d}\right)^2 \tag{3.71}$$

$$+ \frac{1}{\sigma}\left[\frac{\partial}{\partial x_j}\left(\nu_{tot}\frac{\partial \tilde{\nu}}{\partial x_j}\right) + c_{b2}\frac{\partial \tilde{\nu}}{\partial x_i}\frac{\partial \tilde{\nu}}{\partial x_i}\right].$$

The definitions of the functions are

$$f_{v1} = \frac{\chi^3}{\chi^3 + c_{cv1}^2},$$

$$\chi = \frac{\tilde{\nu}}{\nu}, \tag{3.72}$$

where ν is the molecular kinematic viscosity. Additionally,

$$\tilde{S} = \Omega + \frac{\tilde{\nu}}{\kappa^2 d^2}f_{v2}. \tag{3.73}$$

Here Ω is the magnitude of the vorticity, d is the distance from the point to the nearest wall and

$$f_{v2} = 1 - \frac{\chi}{1 + \chi f_{v1}}.$$

f_w is computed as

$$f_w = g\left[\frac{1 + c_{w3}^6}{g^6 + c_{w3}^6}\right]^{1/6}, \tag{3.74}$$

$$g = r + c_{w2}(r^6 - r), \quad r = \min\left(\frac{\tilde{\nu}}{\tilde{S}\kappa^2 d^2}, 10\right).$$

Finally, f_{t2} is computed as

$$f_{t2} = c_{t3}e^{-c_{t4}\chi^2}.$$

The model constants are

$$\begin{aligned}
c_{b1} &= 0.1355, \quad \sigma = 2/3, \quad c_{b2} = 0.622, \\
\kappa &= 0.41, \quad c_{w2} = 0.3, \quad c_{v1} = 7.1, \\
c_{t3} &= 1.2, \quad c_{t4} = 0.5, c_{w1} = \frac{c_{b1}}{\kappa^2} + \frac{1 + c_{b2}}{\sigma}.
\end{aligned}$$
(3.75)

The most widely used version of SA however ignores the trip term f_{t2}. This is referred to as the "no trip term" version of the SA turbulence model and is also used in SU2. In this variation the constant c_{t3} is set to zero. The no trip SA turbulence model can now be written in the general form of equation 3.1 as

$$\frac{\partial U}{\partial t} + \frac{\partial F_i^c}{\partial x_i} - \frac{\partial F_i^v}{\partial x_i} = Q \quad \text{in} \quad \Omega, \quad t > 0,$$

where

$$U = \tilde{\nu}, \qquad F_i^c = u_i\tilde{\nu}, \qquad F_i^v = \frac{(\nu + \tilde{\nu})}{\sigma}\frac{\partial \tilde{\nu}}{\partial x_i},$$

$$Q = c_{b1}\tilde{S}\tilde{\nu} - c_{w1}f_w\Big(\frac{\tilde{\nu}}{d_S}\Big)^2 + \frac{c_{b2}}{\sigma}\Big|\frac{\partial \tilde{\nu}}{\partial x_i}\Big|^2.$$
(3.76)

Discretization
Since the turbulence model equation resembles the general scalar transport equation 2.9, the discretization is carried out as described in section 2.2.1. Since equation 3.76 is solved sequentially after the momentum and pressure correction equations, the previously solved velocity field is used and the advective term, F_i^c is discretized using an upwind scheme. The viscous term, F_i^v, is discretized using a central difference scheme. The source term is discretized using the midpoint integration rule where the gradients are computed with Green-Gauss theorem or the Least Squares method.

Boundary conditions
The most important boundary conditions for the turbulence models are at the walls and far-field boundaries. The boundary condition for the SA turbulence model at the far-field boundaries is

$$\tilde{\nu}_\infty = 3\nu_\infty \quad \text{to} \quad 5\nu_\infty,$$
(3.77)

and on solid walls

$$\tilde{\nu} = 0.$$
(3.78)

3.6.2. Menter Shear Stress Transport (SST)

The Menter Shear Stress Transport equation [14] is a two equation model for finding the turbulent eddy viscosity. The two equations solve for the turbulent kinetic energy, k and specific dissipation rate, ω. This formulation combines two popular two equation models - the k-ω turbulence model [16, 17] and the k-ϵ turbulence model [18, 19] with an additional correction for adverse pressure gradients. The k-ω formulation is used in the inner parts of the boundary layer and k-ϵ formulation in the remaining parts of the flow field. The two equations are

$$\frac{\partial \rho k}{\partial t} + \frac{\partial \rho u_j k}{\partial x_j} = P - \beta^* \rho \omega k + \frac{\partial}{\partial x_j}\left((\mu + \sigma_k \mu_t)\frac{\partial k}{\partial x_j} \right), \tag{3.79}$$

and

$$\frac{\partial \rho \omega}{\partial t} + u_j \frac{\partial \rho u_j \omega}{\partial x_j} = \frac{\partial}{\partial x_j}\left((\mu + \sigma_k \mu_t)\frac{\partial \omega}{\partial x_j} \right) + \frac{\gamma}{\nu_t} P \tag{3.80}$$

$$-\beta \rho \omega^2 + 2(1 - F_1)\frac{\rho \sigma}{\omega}\frac{\partial k}{\partial x_i}\frac{\partial \omega}{\partial x_i}. \tag{3.81}$$

Here P is the production term given by

$$P = \tau_{ij}\frac{\partial u_i}{\partial x_j}. \tag{3.82}$$

where τ_{ij} is the viscous stress tensor (equation 3.3). The turbulent eddy viscosity is computed as

$$\mu_t = \frac{\rho a_1 k}{\max(a_1 \omega, \Omega F_2)}, \tag{3.83}$$

where ρ is density, Ω is the vorticity magnitude, $\Omega = \sqrt{2W_{ij}W_{ij}}$ with

$$W_{ij} = \frac{1}{2}\left(\frac{\partial u_i}{\partial x_j} - \frac{\partial u_j}{\partial x_i} \right).$$

Every constant in the model is a blend of inner and outer value, blended as

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

where ϕ represents any of the constants defined in equation 3.84. The blending functions are computed as

$$F_1 = tanh(arg_1^4),$$

$$arg_1 = min\left[max\left(\frac{\sqrt{k}}{\beta^* \omega d}, \frac{500\nu}{d^2 \omega} \right), \frac{4\rho \sigma_{\omega_2} k}{CD_{k\omega}d^2} \right],$$

$$CD_{k\omega} = max\left(2\sigma_{\omega_2}\frac{1}{\omega}\frac{\partial k}{\partial x_j}\frac{\partial \omega}{\partial x_j}, 10^{-20} \right),$$

$$F_2 = tanh(arg_2^4),$$

$$arg_2 = max\left(2\frac{\sqrt{k}}{\beta^* \omega d}, \frac{500\nu}{d^2 \omega} \right),$$

with d being the distance of any field point to the nearest wall.

The constants of the model are given by

$$a_1 = 0.31, \quad \kappa = 0.41, \quad \beta^* = 0.09,$$
$$\sigma_{k_1} = 0.85, \quad \sigma_{\omega_1} = 0.5, \quad \beta_1 = 0.075,$$
$$\sigma_{k_2} = 1.0, \quad \sigma_{\omega_2} = 0.856, \quad \beta_2 = 0.0828,$$
$$\gamma_1 = \frac{\beta_1}{\beta^*} - \frac{\sigma_{\omega_1}\kappa^2}{\sqrt{\beta^*}}, \quad \gamma_2 = \frac{\beta_2}{\beta^*} - \frac{\sigma_{\omega_2}\kappa^2}{\sqrt{\beta^*}}. \tag{3.84}$$

Following the general form of the equations in equation 3.1, the equations 3.79 and 3.81 can be written as

$$U = \begin{bmatrix} \rho k \\ \rho \omega \end{bmatrix}, F_i^c = \begin{bmatrix} \rho u_i k \\ \rho u_i \omega \end{bmatrix}, F_i^v = \begin{bmatrix} (\mu + \sigma_k \mu_t)\frac{\partial k}{\partial x_i} \\ (\mu + \sigma_\omega \mu_t)\frac{\partial \omega}{\partial x_i} \end{bmatrix}$$

$$Q = \begin{bmatrix} P - \beta^* \rho \omega k \\ \frac{\gamma}{\nu_t} P - \beta \rho \omega^2 + 2(1 - F_1)\frac{\rho\sigma}{\omega}\frac{\partial k}{\partial x_i}\frac{\partial \omega}{\partial x_i} \end{bmatrix}. \tag{3.85}$$

Discretization
Similar to the SA turbulence model, the advective term, F_i^c is discretized using an upwind scheme, the viscous term, F_i^v, using a central scheme and the source term is discretized using the midpoint integration rule with the gradients computed using either the Green Gauss theorem or the Least Squares method.

Boundary conditions
The boundary conditions at the far-field boundaries for the SST $k - \omega$ model are

$$k_\infty = \frac{3}{2}V_\infty^2 TI^2,$$
$$\omega_\infty = \frac{k_\infty}{\nu_\infty \frac{\mu_t}{\mu_{lam}}}. \tag{3.86}$$

Here ν_∞ is the kinematic viscosity in the free stream, V_∞ is the free stream velocity magnitude, ρ is the density and TI is the turbulent intensity. The ratio μ_t/μ_{lam} and turbulent intensity TI are specified as inputs. On solid walls, the boundary conditions are

$$k = 0,$$
$$\omega = 10\frac{6\nu}{\beta_1(\Delta d)^2}. \tag{3.87}$$

Δd is the first cell height and β_1 is a model constant defined earlier.

References

[1] F. Palacios, T. D. Economon, A. Aranake, S. R. Copeland, A. K. Lonkar, T. W. Lukaczyk, D. E. Manosalvas, K. R. Naik, S. Padron, B. Tracey, A. Variyar, and J. J. Alonso, *Stanford university unstructured (SU2): Analysis and design technology for turbulent flows,* in *52nd Aerospace Sciences Meeting* (2014).

[2] F. Moukalled, L. Mangani, M. Darwish, *et al.*, *The finite volume method in computational fluid dynamics*, Vol. 113 (Springer, 2016).

[3] J. H. Ferziger, M. Perić, and M. Peric, *Computational Methods for Fluid Dynamics*, 3rd ed. (Springer, 2002) p. 423.

[4] J. Blazek, *Chapter 5 - unstructured finite volume schemes,* in *Computational Fluid Dynamics: Principles and Applications (Second Edition)*, edited by J. Blazek (Elsevier Science, Oxford, 2005) second edition ed., pp. 131 – 182.

[5] T. J. Barth, *Aspects of unstructured grids and finite-volume solvers for the euler and navier-stokes equations,* In AGARD (1992).

[6] C. M. Rhie and W. L. Chow, *Numerical Study of the Turbulent Flow Past an Airfoil with Trailing Edge Separation,* AIAA Journal **21** (1983).

[7] S. Majumdar, *Role of underrelaxation in momentum interpolation for calculation of flow with nonstaggered grids,* Numerical Heat Transfer **13**, 125 (1988).

[8] S. K. Choi, *Note on the use of momentum interpolation method for unsteady flows,* Numerical Heat Transfer Part a-Applications **36**, 545 (1999).

[9] B. Yu, Y. Kawaguchi, W. Q. Tao, and H. Ozoe, *Checkerboard pressure predictions due to the underrelaxation factor and time step size for a nonstaggered grid with momentum interpolation method,* Numerical Heat Transfer, Part B: Fundamentals **41**, 85 (2002).

[10] W. Z. Shen, J. A. Michelsen, and J. N. Sørensen, *Improved Rhie-Chow interpolation for unsteady flow computations,* AIAA Journal **39**, 2406 (2012).

[11] A. Cubero and N. Fueyo, *A compact momentum interpolation procedure for unsteady flows and relaxation,* Numerical Heat Transfer, Part B: Fundamentals **52**, 471 (2007).

[12] P. Bartholomew, F. Denner, M. H. Abdol-Azis, A. Marquis, and B. G. van Wachem, *Unified formulation of the momentum-weighted interpolation for collocated variable arrangements,* Journal of Computational Physics **375**, 177 (2018).

[13] P. Spalart and S. Allmaras, *A one-equation turbulence model for aerodynamic flows,* in *30th aerospace sciences meeting and exhibit* (1992) p. 439.

[14] F. R. Menter, *Two-equation eddy-viscosity turbulence models for engineering applications,* AIAA journal **32**, 1598 (1994).

[15] C. Rumsey, *Nasa turbulence modeling resource,* `https://turbmodels.larc.nasa.gov/index.html` (2019).

[16] D. C. Wilcox, *Reassessment of the scale-determining equation for advanced turbulence models,* AIAA journal **26**, 1299 (1988).

[17] D. C. Wilcox, *Formulation of the kw turbulence model revisited,* AIAA journal **46**, 2823 (2008).

[18] B. E. Launder and B. I. Sharma, *Application of the energy-dissipation model of turbulence to the calculation of flow near a spinning disc,* Letters in heat and mass transfer **1**, 131 (1974).

[19] W. Jones and B. E. Launder, *The prediction of laminarization with a two-equation model of turbulence,* International journal of heat and mass transfer **15**, 301 (1972).

4

Verification and Validation

In this chapter, the different features of the new pressure based solver will be verified and validated against analytical solutions, experimental data and other references. For verification of the solver, problems with an analytical solution like the Taylor-Couette flow and the plane Poisuelle flow will be used. The validation cases presented will be used to test the different features that are typically encountered during external aerodynamics applications. First, laminar flow problems are used to validate the implementation of the flow solver. Then, turbulent flow problems are presented to validate the coupling of the new flow solver with the existing turbulence solvers. Finally, the unsteady behavior of the new solver is validated. Wherever possible the grids for the numerical simulations are taken from standard sources like the NASA turbulence modeling database [1] or the SU2 test case repository to avoid any potential sources of error from mesh generation.

Laminar and turbulent boundary layers are extremely crucial in wind turbine aerodynamics and the numerical solutions from the new solver are validated against theoretical boundary layer solutions under both circumstances. Flow separation, though undesirable, occurs frequently and the behavior of the new solver under separated conditions is tested for the flow over a backward facing step for both laminar and turbulent flow conditions. Other representative test cases, like the flow over a cylinder and flow past an airfoil are also presented. Finally, two unsteady cases are considered; a laminar flow past a square cylinder and a turbulent flow past a pitching airfoil undergoing dynamic stall.

4.1. Verification

In order to verify the accuracy of the solver, Couette flow or the flow between two solid surfaces is simulated. Two special cases are considered here - the flow between two concentric infinitely long rotating cylinders (also known as Taylor-Couette flow, see the figure 4.1) and a planar flow between two solid infinitely long parallel plates

that are held fixed (also known as plane Poisuelle flow, see figure 4.4a). An analytical solution for the velocity profile can be found for both these cases which is used to calculate the order of convergence of the solver.

While an analytical solution is not available for the laminar flow in a lid driven cavity, reference solutions for velocity can be found in literature [2]. Thus, the solutions for the lid driven cavity is also used to compute the order of accuracy.

Taylor Couette flow
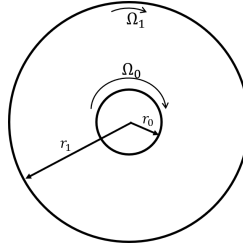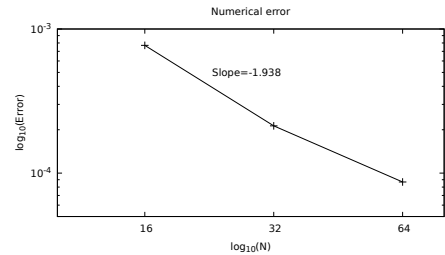The schematic of the Taylor-Couette flow is shown in figure 4.1. The inner cylinder

Figure 4.1: Schematic of the Taylor-Couette flow.

has a radius of r_0 and the outer radius is r_1. Ω_0 and Ω_1 are the angular velocities of the inner and outer cylinders respectively. The analytical solution [3] for the azimuthal velocity as a function of the radius r is

$$u_{ana}(r) = r_0\Omega_0\frac{r_1/r - r/r_1}{r_1/r_0 - r_0/r_1} + r_1\Omega_1\frac{r/r_0 - r_0/r}{r_1/r_0 - r_0/r_1}. \tag{4.1}$$
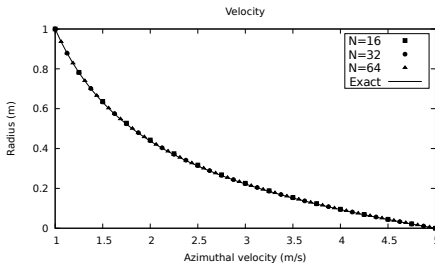
(a) Velocity vs radial distance.   (b) Order of convergence.

Figure 4.2: Grid convergence for the Taylor Couette flow.

The simulation was carried out on a domain with $r_0 = 1m$ and $r_1 = 5m$. The outer wall is held fixed ($\Omega_1 = 0$) and the inner wall is rotating at an angular velocity $\Omega_0 = 1$ rad/s. The two solid walls are treated as moving wall boundaries. Three different grid resolutions are considered with 16, 32 and 64 cells along the radial

direction and with 40 nodes along the circumference of the cylinders. Uniform grid spacing is used in all cases.

Figure 4.2a shows the comparison between the numerical and analytical velocity profile. The numerical solution matches the analytical solution for all the grid resolutions very well. The numerical error is computed as

$$e_{num} = |u_{ana} - u_{num}|.$$

Figure 4.2b shows the log_{10} of the L_2 norm of the error plotted against the number of points in the radial direction. The numerical error decreases at a rate of 1.938 indicating an approximately second order rate of convergence as the grid size is halved.

Convergence behavior The residual history for the grids with 32 and 64 nodes are plotted in figures 4.3a and 4.3b. The residuals of both the velocity components and the mass flux, which serves as the indicator for convergence of the continuity equation, all converge within 1200 iterations for the coarser grid and about 4400 iterations for the fine grid. The two velocity components converge identically because
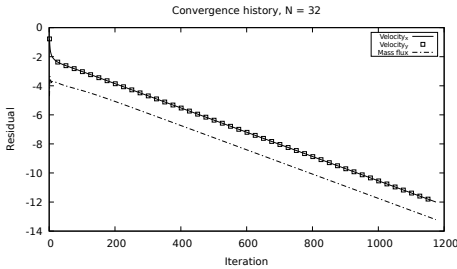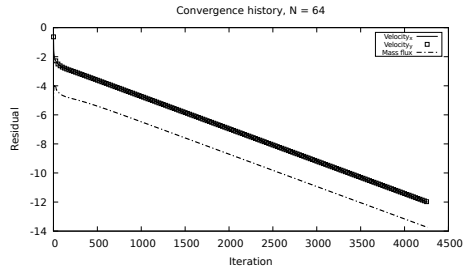
(a) 40×32 grid.   (b) 40×64.

Figure 4.3: Convergence history for the Taylor Couette flow residual.

of rotational symmetry.

Plane Poisuelle flow
The plane Poisuelle flow is the flow between two infinitely long parallel plates that are held fixed (figure 4.4a). The boundary conditions used for the simulation are shown in figure 4.4b. At the inlet boundary, a uniform velocity profile is prescribed, and a zero velocity gradient at the outlet is specified. The outlet pressure is set to zero. A small symmetry region is present immediately after the inlet before the channel starts. The plane Poisuelle flow at a mean flow Reynolds number based on channel width h and inlet velocity U_{in} of $Re = \frac{hU_{in}}{\nu} = 400$ is considered. For fully developed flow conditions, the axial velocity profile at any location y can be computed as

$$u(y) = -\frac{dP}{dx}\frac{1}{2\mu}y(y - h), \qquad 0 \le y \le h \tag{4.2}$$

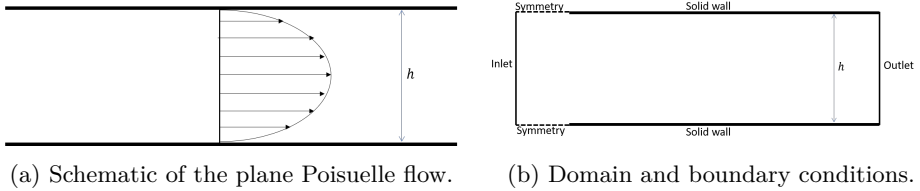(a) Schematic of the plane Poisuelle flow.   (b) Domain and boundary conditions.

Figure 4.4: (a) Schematic of the plane Poisuelle flow and (b) the domain and boundary conditions for the numerical simulation.

where $\frac{dP}{dx}$ is the constant pressure gradient in the streamwise direction, μ is the laminar viscosity and h is the channel width. The domain is $1.25m$ long in the streamwise direction with a symmetry region of $0.25m$ after the inlet boundary. The distance between the two solid walls is $h = 0.125m$ (figure 4.4b). Four different mesh resolutions are chosen with 16, 32, 64 and 128 elements in the direction normal to the solid walls and 100 elements in the streamwise direction.

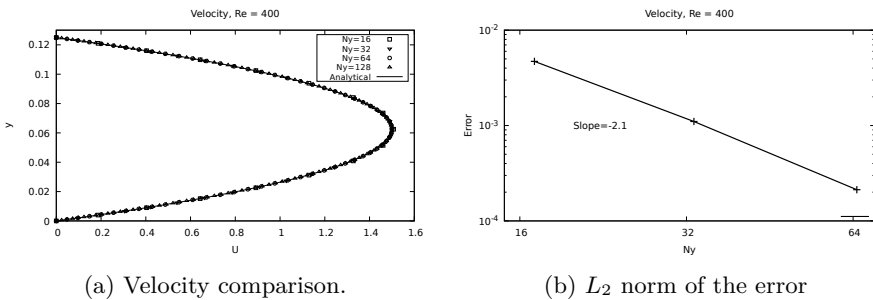(a) Velocity comparison.   (b) L_2 norm of the error

Figure 4.5: (a) Comparison of numerical and analytical results. (b) Error as a function of number of cells.

Figure 4.5a shows the comparison of the streamwise velocity at $x = 0.9$, which is $7.2h$ away from the start of the channel, as a function of the normal distance for different grid resolutions against the analytical solution. The flow is fully developed after roughly $5h$ to $6h$ from the start of the channel. The pressure gradient required to compute the numerical solution is obtained from the results of the finest grid with 128 elements in the normal direction. The numerical results from all the grids match the analytical solution closely. The error between the numerical solution and the analytical solution is computed for the three meshes with 16, 32 and 64 elements. The L_2 norm of the error is plotted against the number of elements in the normal elements in the figure 4.5b on a log scale. The slope of the error curve is 2.1.

Convergence behavior Figure 4.6a shows the convergence history for the two velocity components and the mass flux for the SIMPLE algorithm. Convergence is achieved in 630 iterations. Figure 4.6b shows the comparison of residual conver-
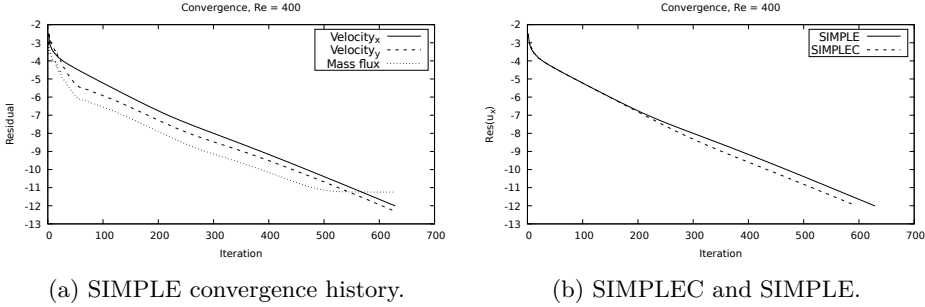
(a) SIMPLE convergence history.   (b) SIMPLEC and SIMPLE.

Figure 4.6: Plane Poisuelle flow (a) residual convergence history on the 100×64 grid and (b) comparsion of convergence behavior for SIMPLE and SIMPLEC for the streamwise velocity on the 100×64 grid.

gence of the streamwise velocity component for the SIMPLE and SIMPLEC iterative methods. Since the pressure corrections are under-relaxed appropriately, there is no significant difference in convergence behavior. However, SIMPLEC does converge slightly faster in this case.

Lid driven cavity

The flow within a lid driven cavity is a commonly used validation problem for CFD solvers. The domain consists of a square cavity with the top wall being moved at a constant velocity along the x axis. This case also serves to test the moving wall boundary condition. While there is no analytical solution for this case, the results from a lid driven cavity are compared against results from Ghia [2]. Ghia [2] solves the vorticity stream function formulation of the incompressible Navier Stokes equations using a multigrid method. The flow is steady and laminar and is a solution of the exact Navier Stokes equation. A square domain $L \times L$ with side $L = 1m$ is used. The top wall of the domain is moving at a constant velocity, U_w, in the x direction. The flow Reynolds number defined based on the wall velocity, U_w and the length of side of the square lid is $Re = \frac{U_w L}{\nu} = 400$. Three different grids are considered with 33, 65 and 129 nodes along each direction. A uniform grid spacing is used throughout the domain. Figure 4.7a shows the x component of the velocity along $x = 0.5$ for the three grids and the benchmark results from Ghia [2]. The numerical results improve against the benchmark solution as the grid is refined. Figure 4.7b shows the L_2 norm of the error between the results from SU2 and the benchmark solution. The error reduces with a slope of 1.89 as the grid spacing is halved.

From both the verification cases with an analytical solution, it is seen that when the number of elements is doubled and subsequently the grid spacing, Δ, is halved, the error is proportional to a factor of approximately $(\Delta)^2$. Also for the lid driven cavity solution where no analytical solution was available, the numerical error when compared against another numerical benchmark result reduced by a factor $(\Delta)^{1.89}$ indicating approximately second order accuracy of the spatial discretization scheme.
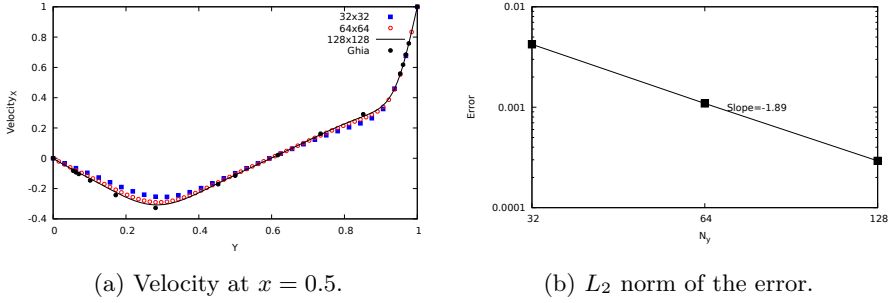
(a) Velocity at $x = 0.5$.   (b) L_2 norm of the error.

Figure 4.7: (a)Velocity profile comparison between numerical results from SU2 and reference solution [2] at $x = 0.5$ and (b) the L_2 norm of the error.

4.2. Validation

In this section the numerical results from the new pressure based solver are validated against either experimental data or reference data from other computational methods. The focus will be to test different flow features that are relevant for external aerodynamics.

4.2.1. Laminar flows

Flow over a flat plate

Understanding the boundary layer is critical for external aerodynamics. To that end, first the laminar flow over a flat plate with no pressure gradient at a Reynolds number of $Re = 4 \times 10^5$ is now considered to analyze the behavior of the new solver in capturing the laminar boundary layer. A semi analytical solution, commonly known as the Blasius solution [4], can be found for the streamwise and normal velocity components under self similar conditions. Self similar solutions can be found by first transforming the coordinates and the velocities as follows

$$x \to c^2 x, \quad y \to cy, \quad u \to u, \quad v \to \frac{v}{c}. \tag{4.3}$$

Here c is any positive constant. Introducing the new similarity variable η and the non dimensional function f as

$$\eta = \frac{y^*}{\delta(x^*)}, \quad \psi = \sqrt{\nu U x^*} f(\eta), \tag{4.4}$$

where $\delta(x)$ is the boundary layer thickness at a location x, U is the free stream velocity, ν is the kinematic viscosity, ψ is the stream function and $f(\eta)$ is a function of the similarity variable only. An ordinary differential equation (ODE) in $f(\eta)$ can then be formed as, see [4]

$$2f''' + f'' f = 0, \tag{4.5}$$

where $'$ denotes differentiation with respect to the similarity variable η. The boundary conditions can be derived from the no slip condition at the wall and the matching

condition at the edge of the boundary layer.

$$u(x,0) = 0, \quad \rightarrow f'(0) = 0,$$
$$v(x,0) = 0, \quad \rightarrow f(0) = 0,$$
$$u(x,\infty) = U \quad \rightarrow f'(\infty) = 1.$$

The velocity components can be written in terms of the stream function ψ and consequently the similarity variable as

$$u(x,y) = \frac{\partial \psi}{\partial y} = U f'(\eta), \quad v(x,y) = -\frac{\partial \psi}{\partial x} = \frac{1}{2}\sqrt{\frac{\nu U}{x}}[\eta f'(\eta) - f(\eta)]. \qquad (4.6)$$

Additionally, the wall shear stress is given by,

$$\tau_w = \mu \left(\frac{\partial u}{\partial y}\right)_{y=0} = 0.332 \mu U \sqrt{\frac{U}{\nu x}}. \qquad (4.7)$$

Using the wall shear stress, the skin friction coefficient, C_f, can be found as

$$C_f = \frac{\tau_w}{\frac{1}{2}\rho U^2}, = 0.664 \frac{0.664}{\sqrt{Re_x}}, \qquad (4.8)$$

where Re_x is the local Reynolds number defined as $Re = \frac{Ux}{\nu}$. Equation 4.5 can be solved numerically to find $f(\eta)$ as a function of η. With that the self similar velocity profiles and the skin friction coefficient can be found which will be used to study the behavior of the new solver in capturing the boundary layer.

The domain used for the numerical simulations is shown in figure 4.8a. A uniform inflow is prescribed and a small inflow region with a symmetry boundary is used before the flat plate begins. Two different meshes are considered. The coarse mesh has 65 nodes in both while the fine mesh has 129 nodes along the streamwise and normal directions. The boundary conditions and the coarse mesh used for the simulation are shown in figures 4.8a and 4.8b. Nodes are clustered near the wall and stretched away from it in the normal direction and clustered around the interface between the symmetry and the wall region and stretched towards the outlet in the streamwise direction (figure 4.8b). The minimum normal grid spacing in the coarse mesh is $1.60 \times 10^{-5}m$ and $8.0 \times 10^{-6}m$ for the fine mesh. Grid spacing at $x = 0$ when the wall begins is $0.001m$ in the coarse mesh and $0.0005m$ for the fine mesh
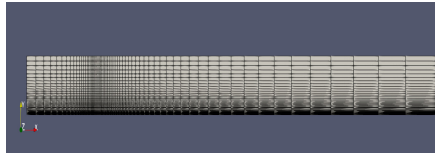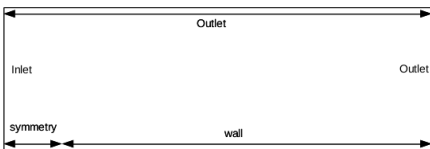
(a) Domain and boundary conditions.   (b) Mesh.

Figure 4.8: Flat plate (a) domain and boundary conditions and (b) the 65×65 mesh.

Figures 4.9a and 4.9b show the comparison of the streamwise and normal velocity components at $x = 0.15m$ against Blasius solution for the two meshes considered. The streamwise component matches very closely with the Blasius solution in both meshes whereas there is a small difference in the normal component of velocity near the edge of the boundary layer. The profile from the fine mesh is closer to the analytical profile. It should also be noted that the normal component of the velocity is significantly smaller than the streamwise component. Figure 4.10 shows the skin

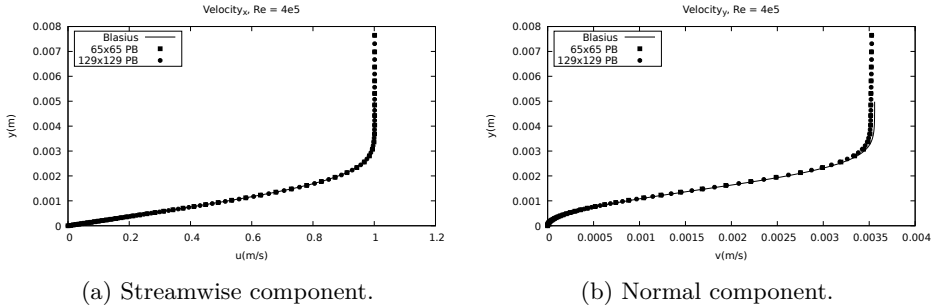(a) Streamwise component.   (b) Normal component.

Figure 4.9: Comparison of streamwise and normal velocity components to the Blasius solution at $x = 0.15$.

friction coefficient from the two meshes compared against the Blasisus solution. A good agreement between the numerical results and theory is found in both cases.
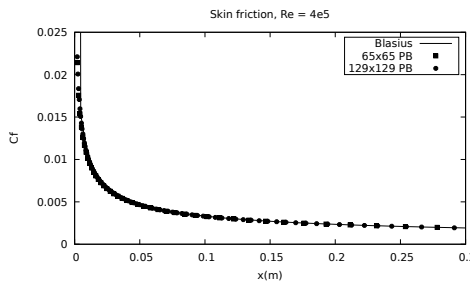

Figure 4.10: Skin friction coefficient for the laminar flow over the flat plate.

Flow over a cylinder

As a simple test case of external aerodynamics, the flow past a circular cylinder is considered. At low Reynolds numbers, the flow remains steady and laminar [5]. The flow separates symmetrically at two points on the cylinder and a recirculation region is formed behind the cylinder. In this study, the drag coefficient and the different flow features (figure 4.11) are compared against a reference numerical solution [5]. L_w corresponds to the length of the wake, (a, b) is the location of the recirculation center and θ is the separation angle on the upper half of the cylinder.
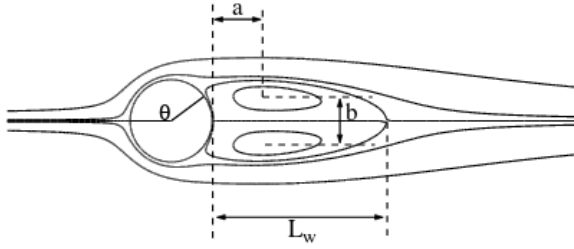
Figure 4.11: Flow features around a cylinder at a Reynolds number $Re = 40$ [5].

The Reynolds number based on the cylinder diameter is $Re = 40$ and a series of five grids are used to study the properties. The diameter of the cylinder is $D = 1m$. The coarsest grid has 65 nodes on the cylinder. The number of nodes are doubled till 1025 nodes on the cylinder. The domain extends for $50D$ to all sides of the cylinder. A freestream boundary condition is imposed on the outer part of the domain. Figure 4.12a shows the drag coefficient (C_d) from the different grids. The values are also listed in table 4.1. The solution converges to a value of 1.499 which is close to the reported value in Gautier et al [5]. Figure 4.12b shows the different flow features. Separation on the upper half occurs around the point $(x, y) = (0.294, 0.404)$ (assuming the center of the cylinder lies at $(x, y) = (0, 0)$) which corresponds to a separation angle of approximately $\theta = 126°$ measured from the leading edge. The center of the recirculation region is at $(1.2, 0.295)$ on the upper half and at $(x, y) = (1.2, -0.295)$ on the lower half. The flow features from SU2 and Gautier et al [5] are also listed in table 4.2. Like the drag coefficient, the flow features match closely with the reference solution. These flow features are computed on a grid with 513 nodes on the cylinder as a grid independent solution is obtained at that point.
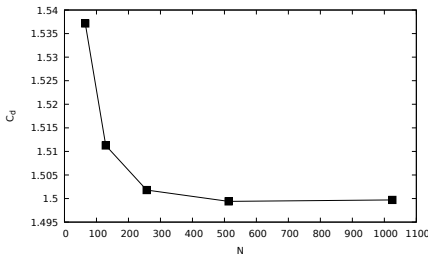
(a) Grid refinement study of the drag coefficient.
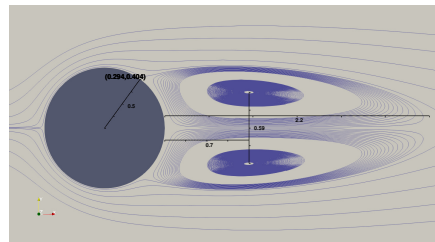
(b) Recirculation region behind the cylinder.

Figure 4.12: Laminar flow over a cylinder at $Re = 40$.

Richardson's extrapolation formula [6]

$$C_d(h/t) - C_d(h/s) \approx \frac{C_d(h/s) - C_d(h)}{s^k - 1} - \frac{C_d(h/t) - C_d(h)}{t^k - 1} \qquad (4.9)$$

Nodes (N)	Drag (C_d)
65	1.537
129	1.511
257	1.502
513	1.499
1025	1.499
Gautier et al [5]	1.49

Table 4.1: Drag coefficient (C_d) for different grid resolutions. N denotes the number of points on the surface of the cylinder.

	SU2	Gautier et al [5]
L_w/D	2.18	2.24
a/D	0.70	0.71
b/D	0.59	0.59
θ	126°	126.4°

Table 4.2: Flow features for flow around a cylinder at a Reynolds number of $Re = 40$.

can be used on the drag values in table 4.1 to find the approximate order of convergence. Here h is the starting grid size, t and s are two integer factors and k is the order of convergence. The grid size h can be found for each case as $\pi D/(N-1)$ where D is the cylinder diameter, πD is the circumference of the cylinder and N is the number of nodes on the cylinder. The results are shown in table 4.3. The drag coefficient converges at a rate of approximately 1.5.

N	t	s	k
65	2	4	1.53
129	2	4	1.58
65	4	8	1.56

Table 4.3: Order of convergence of the drag coefficient using Richardson's extrapolation for the flow over a cylinder at $Re = 40$.

Flow over a backward facing step
Flow separation occurs commonly in external aerodynamics. In the previous case of the flow over the cylinder, the separation location was validated against reference data. In this section, the behavior of the flow within the separated region is tested by analyzing the flow over a backward facing step. The domain consists of an inlet channel which expands into a larger channel across a step. The flow separates at the step and re-attaches downstream along the lower wall. Unlike the flow past a cylinder, here the flow separation occurs at a fixed point, namely at the corner of the step removing any uncertainty in the location of the separation point. Depending on the Reynolds number, a secondary separated region can also occur along the top wall. A Reynolds number of $Re = 800$ based on the step height is considered and

the flow is expected to separate along both the bottom and top walls. The numerical results are compared to results from Gartling [7].

The outline of the domain used is shown in figure 4.13. The step height is $0.5m$ and the channel height is $1m$. The step and the inlet is located on the left boundary at $x = 0$ as shown in the figure 4.13. The step starts at $y = -0.5$ and extends upto $y = 0$. The region from $y = 0$ to $y = 0.5$ is treated as an inlet boundary and a fully developed velocity profile is imposed there. Three different meshes are considered with uniform grid resolutions of 121×17, 241×33 and 481×65 nodes in the streamwise and normal direction, respectively. At the outlet, the pressure is prescribed to be zero.
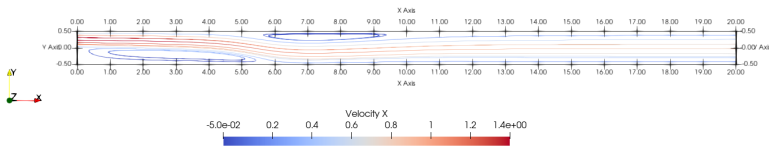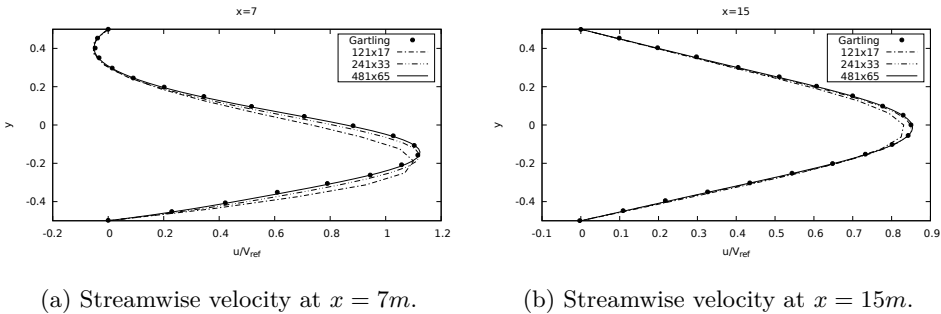

Figure 4.13: Streamlines for the laminar backward facing step at $Re = 800$.

(a) Streamwise velocity at $x = 7m$.   (b) Streamwise velocity at $x = 15m$.

Figure 4.14: Comparison of the streamwise velocity profiles obtained from the numerical results with the literature at (a) $x = 7m$ and (b) $x = 15m$.

Figure 4.14 compares the numerical and experimental results for the streamwise velocity component at two different locations for the three meshes. Figure 4.14a shows the velocity at $x = 7m$. This is within the recirculation region and the flow is separated at the upper wall. The coarse mesh (121×17) performs poorly at this location. The results from the finest mesh (481×65) agree very well with the experimental data. Figure 4.14b shows the same comparison at $x = 15m$. This location is after the flow has reattached on both the upper and lower walls. As a consequence, the performance of the coarse mesh improves and the other two meshes also give very good results. The length of the recirculating zone along the

lower wall is $5.81m$ and along the upper wall is $5.69m$ which match the results from
Gartling [7].

4.2.2. Turbulent flows

Most flows of practical interest are turbulent and in this section the behavior of the
new solver in turbulent conditions will be validated.

Flow over a flat plate

The turbulent boundary layer can be broadly divided into two regions [4, 8]: the
inner region where viscous dissipation is present and the outer region where the
turbulence dissipation dominates completely. The inner region consists of the vis-
cous sub-layer where viscous effects dominate and turbulent effects are absent, a
buffer region where the turbulent stresses start to grow and finally an overlap or
logarithmic region where the turbulent and viscous dissipation match. The overlap
region merges into the outer layer of the boundary layer where viscous effects are
minimal. The velocity profile in the viscous sub-layer and logarithmic region can be
written respectively as

$$u^+ = y^+, \qquad y^+ \leq 5, \tag{4.10}$$

$$u^+ = \frac{1}{\kappa} ln(y^+) + C, \qquad y^+ > 30. \tag{4.11}$$

The region of the boundary layer between $5 \leq y^+ \leq 30$ is the buffer region. In the
above relations, y^+ is the non dimensional wall normal coordinate and u^+ is the
normalized velocity defined as

$$y^+ = \frac{yu_\tau}{\nu}, \qquad u^+ = \frac{u}{u_\tau}, \qquad u_\tau = \sqrt{\frac{\tau_w}{\rho}}.$$

Here u_τ is known as the wall friction velocity and is used as the velocity scale close to
the wall, τ_w is the wall shear stress, ρ is the density, u is the local tangential velocity
and ν is the kinematic viscosity. The constant in equation 4.11 for a smooth wall is
known to be $C = 5.0$.

A turbulent flow over a flat plate is simulated at a Reynolds number of $Re = 5 \times 10^6$ and the results are compared to the standard 2D zero pressure gradient
flat plate validation case from the NASA Turbulence Modeling resource [1]. The
domain used is shown in figure 4.15. Five different grid levels are considered. The
coarsest mesh has 29 points on the wall and 25 points in the normal direction with
an average $y+ \approx 1.7$. The finest mesh has 449 points on the surface and 385 points
in the normal direction with an average $y+ \approx 0.1$. The points are clustered near the
wall to ensure adequate resolution of the turbulent boundary layer near the wall.
The grids used are also from the NASA turbulence modeling database [1].

The skin friction coefficient (C_f) at $x = 0.97m$ for the different grids with the
SA and SST turbulence models are listed in table 4.4. A grid independent solution
is obtained for both turbulence models. Further analysis will be carried out on the
grid with 225 points on the surface. Figures 4.16a and 4.16b show the comparison
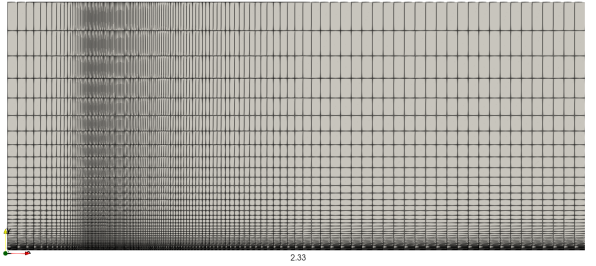
Figure 4.15: Flat plate domain and mesh (137×97 nodes).

N	SA	SST
29	0.00267	0.00258
57	0.00271	0.00265
113	0.00269	0.00269
225	0.00272	0.00270
449	0.00273	0.00272

Table 4.4: Skin friction coefficient (C_f) at $x = 0.97m$ for different grid resolutions with SA and SST. N denotes the number of points on the surface of the flat plate.

(a) SA.   (b) SST.
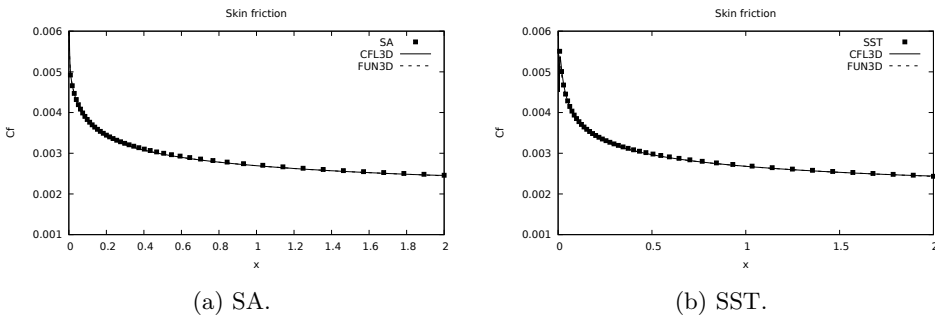
Figure 4.16: Skin friction coefficient, C_f, comparison between the numerical results from SU2, CFL3D and FUN3D [1] for the flow over a flat plate at $Re = 5 \times 10^6$ using the (a) SA and (b) SST turbulence models.

of the skin friction obtained from the SA and SST turbulence models respectively against numerical results from FUN3D and CFL3D [1].

Figures 4.17a and 4.17b show the comparison of the inner velocity profile from the SA and the SST turbulence models against the analytical profiles [4]. The numerical results match the analytical profile closely in both the viscous sublayer and the logarithmic region.

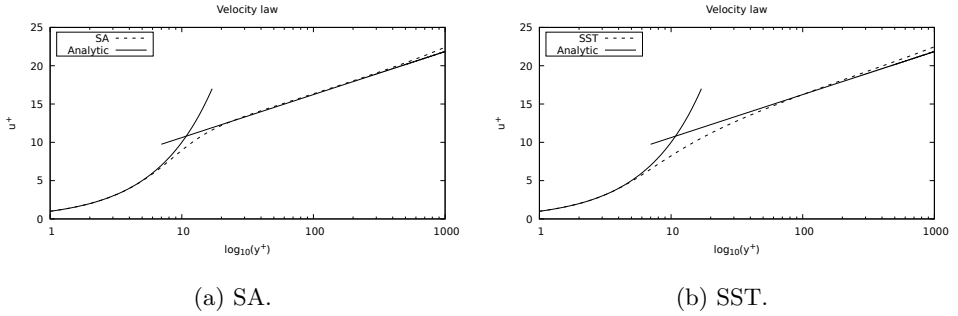(a) SA.   (b) SST.

Figure 4.17: Comparison of the inner velocity profile based on the numerical results from SU2 against analytical results for the turbulent flow over a flat plate at $Re = 5e6$ using the (a) SA and (b) SST turbulence model.

Convergence behavior   Figures 4.18a and 4.18b show the convergence history for the velocity components, mass flux and the eddy viscosity for the SA and turbulent kinetic energy for the SST turbulence model. The velocity components converge smoothly and the residuals drop by more than 6 orders of magnitude for both turbulence models.

(a) SA turbulence model.   (b) SST turbulence model.
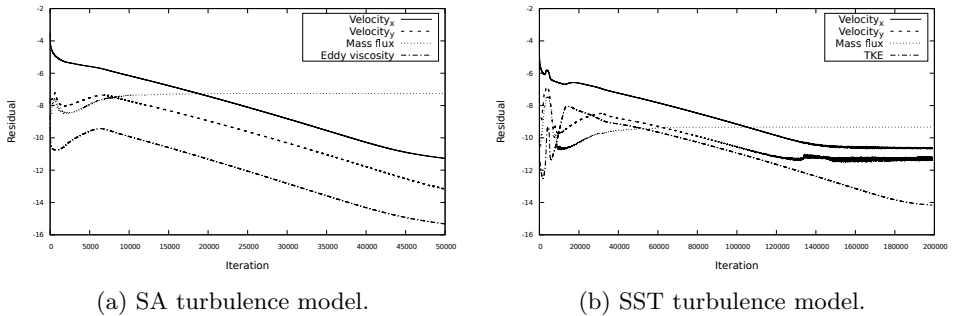
Figure 4.18: Convergence history of the velocity components, mass flux and turbulence variables for the turbulent flow over a flat plate at $Re = 5e6$ using the SA and SST turbulence models.

Backward facing step

The laminar flow over a backward facing step was presented earlier to study the behavior of the new solver in separated regions. In this section, the turbulent flow over a backward facing step at a Reynolds number of $Re_H = 36,000$ based on the step height is presented. The step is $H = 1m$ high and the inlet channel is $9m$ high. As seen previously, the flow separates at the step and reattaches further downstream. Unlike the laminar problem at lower Reynolds numbers, there is only one separated region in the turbulent case. The grids are obtained from the NASA turbulence modeling database [1]. Comparison is made between three grid levels and results from CFL3D [1] and experimental data [9]. The coarsest grid (denoted as SU2 lvl4)

has approximately 5000 nodes, the second grid (SU2 lvl3) has approximately 20000 nodes and the finest grid (SU2 lvl2) has approximately 80000 nodes in total. The coarsest mesh and the domain used is shown in figure 4.19. Nodes are clustered near the step in both the streamwise and normal direction and stretched away from the step.
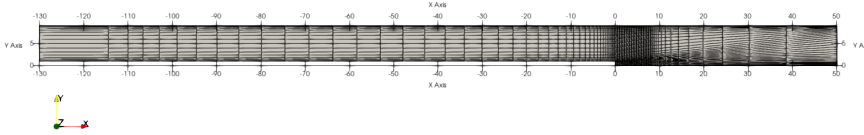
(a) Domain.
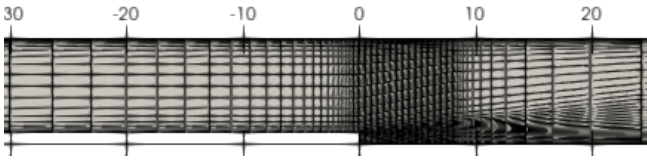
(b) Step.

Figure 4.19: The coarsest mesh for the whole domain (a) and the mesh around the step (b) for the turbulent flow over backward facing step at $Re_H = 36,000$.

Skin friction and reattachment length   Figures 4.20a and 4.21a show the skin friction coefficient in the streamwise direction after the step. Both turbulence models overshoot the minimum skin friction coefficient reported in the experiments. This is also observed in the results from CFL3D. The results from the SA turbulence model recover and are closer to the experimental data compared to CFL3D after $x/H = 10m$. The results from the SST turbulence model are closer to experimental data compared to the SA turbulence model. The minimum value predicted by the SST model is also closer to the experimental data. After the separation zone, the skin friction coefficient from the SST model matches the experimental results and performs better than CFL3D.

Figures 4.20b and 4.21b show the skin friction near the separated region. The reattachment point can be identified by locating where the C_f values become zero for the first time. The SA turbulence model predicts that the flow reattaches at $x/H \approx 6.1$ which is close to the value reported by other CFD codes [1] using SA. The reattachment point predicted by SST is at $x/H \approx 6.54$. The reattachment is observed in the experiment at $x/H = 6.25 \pm 0.1$
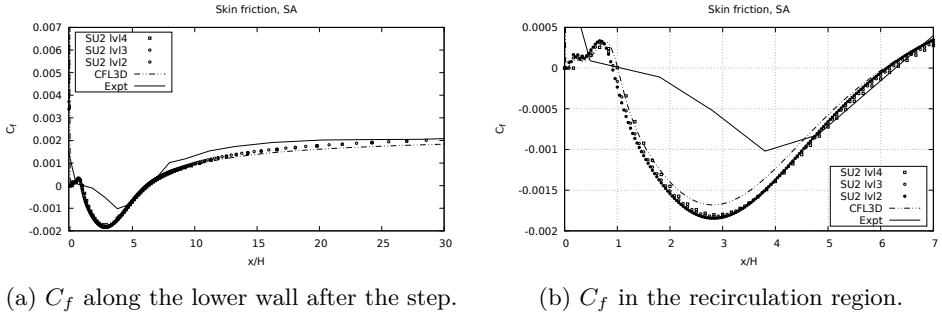
(a) C_f along the lower wall after the step.

(b) C_f in the recirculation region.

Figure 4.20: Skin friction coefficient (C_f) using the SA turbulence model.

(a) C_f along the lower wall after the step.

(b) C_f in the recirculation region.
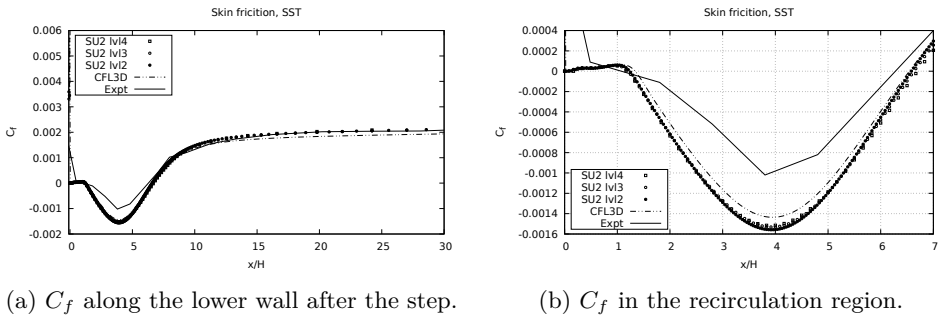
Figure 4.21: Skin friction coefficient using the SST turbulence model.

Velocity profiles The velocity profiles are normalized by a reference velocity, U_{ref} defined as the centerline velocity at $x/H = -4$. Figures 4.22a and 4.22b show

(a) Velocity at $x/H = 1$.

(b) Velocity at $x/H = 4$.

Figure 4.22: Velocity profile using the SA turbulence model (a) $x/h = 1$, (b) $x/H = 4$.

the velocity profile at $x/H = 1$ and $x/H = 4$ using the SA turbulence model on different grids. Both these locations are within the separated region and like the skin friction coefficient results, the velocity profile from the SA turbulence model does not match the experimental data. However, above the step height ($y/H = 1$)

and outside the recirculation zone the results do match the experimental data. At $x/H = 4$, the results of all the grid levels from SU2 match the numerical results from CFL3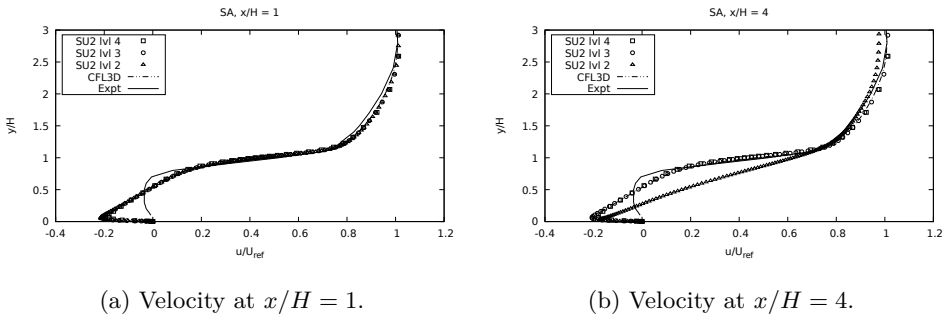D in the recirculation region. Outside the recirculation region, the velocities recover and match the experimental data earlier than CFL3D but the results from the "lvl 2" grid (fine grid) appears to deviate slightly. $x/H = 6$ (figure 4.23a) is

(a) Velocity at $x/H = 6$.   (b) Velocity at $x/H = 10$.
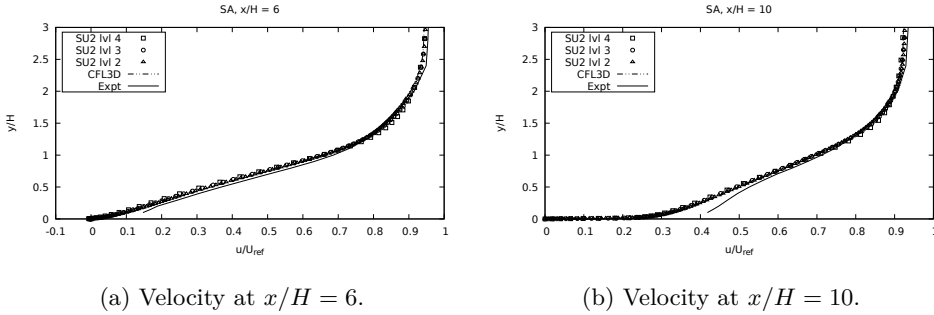
Figure 4.23: Velocity profile using the SA turbulence model (a) $x/h = 6$, (b) $x/H = 10$.

at the edge of the recirculation region and consequently, there is a small mismatch near the wall. The numerical results match the experimental data at $x/H = 6$ much earlier than $x/H = 1$ or $x/H = 4$. $x/H = 10$ (figure 4.23b) is in the attached region of the flow and the velocity profile matches the experimental data for most of the region.

(a) Velocity at $x/H = 1$.   (b) Velocity at $x/H = 4$.
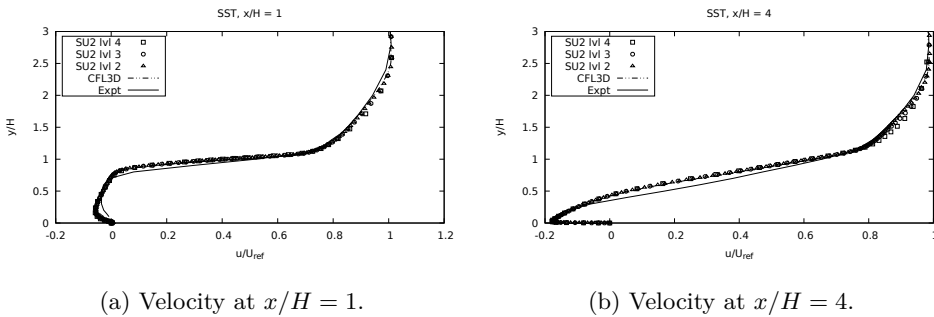
Figure 4.24: Velocity profile using the SST turbulence model (a) $x/h = 1$, (b) $x/H = 4$.

As observed with the skin friction results, the SST turbulence model performs much better in the separated region. The velocity profiles at $x/H = 1$ (figure 4.24a) and $x/H = 4$ (figure 4.24b) are much closer to the experimental data for all the grid levels compared to those from the SA turbulence model. The velocity profiles at $x/H = 6$ (figure 4.25a) and $x/H = 10$ (figure 4.25b) behave similarly to the SA turbulence model.
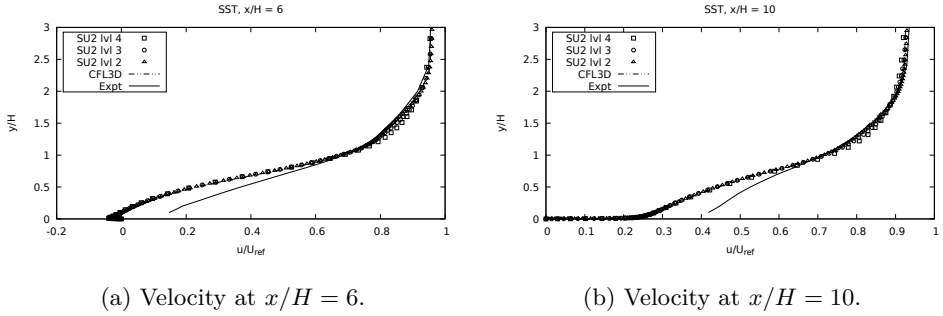
(a) Velocity at $x/H = 6$.   (b) Velocity at $x/H = 10$.

Figure 4.25: Velocity profile using the SST turbulence model (a) $x/h = 6$, (b) $x/H = 10$.

NACA0012 airfoil

Flows over airfoils are a very typical problem in most aerodynamic applications. Lift and drag polars of airfoils used in the different sections of a wind turbine blade are commonly used as input to the lower fidelity tools based on Blade Element Momentum (BEM) theory or lifting line theory which can then be used to analyze the performance of turbine blades. In this section, a fully turbulent flow over a NACA 0012 airfoil employing the SA and the SST turbulence models are compared with the experimental data from Ladson [10]. Experiments were conducted at a range of Reynolds numbers under free and fixed transition conditions. In this study, the results from the fixed transition case at a Reynolds number of $Re = 6 \times 10^6$ are used. The flow was tripped at 0.05% of the chord using a carborundum strip. A grid refinement study is carried out on a series of 4 grids at an angle of attack of $10°$ and the results are tabulated in table 4.5. Based on the grid refinement study,

N	C_l	C_d
129	1.082	0.0119
257	1.090	0.0119
513	1.089	0.0121
1025	1.088	0.0124

Table 4.5: Lift (C_l) and drag (C_d) coefficients for the SA turbulence model. N denotes the number of points on the surface of the airfoil at an angle of attack of $10°$ for NACA 0012.

the grid with 513 points on the airfoil is chosen for further analysis.

Lift and drag coefficients   The numerical results match the experimental data closely for both turbulence models. Figure 4.26 shows the comparison of lift and drag coefficients against experimental data [10] for the SA turbulence model and figure 4.27 shows the results for the SST turbulence model. The results from the SA turbulence model match closely at lower angles of attack but start to deviate at higher angles of attack. The maximum lift coefficient predicted by the SA model is 1.69. On the other hand, the SST turbulence model matches the experimental

data closely at higher angles of attack also. The maximum lift coefficient predicted by the SST turbulence model is 1.60 which is close to the values reported in the experiments. Steady state simulations were carried out until an angle of attack of 17 and higher angles were carried out by employing the unsteady solver. Both the turbulence models over-predict the stall angle compared to the experiments. In the latter case, the reported results are time averaged.
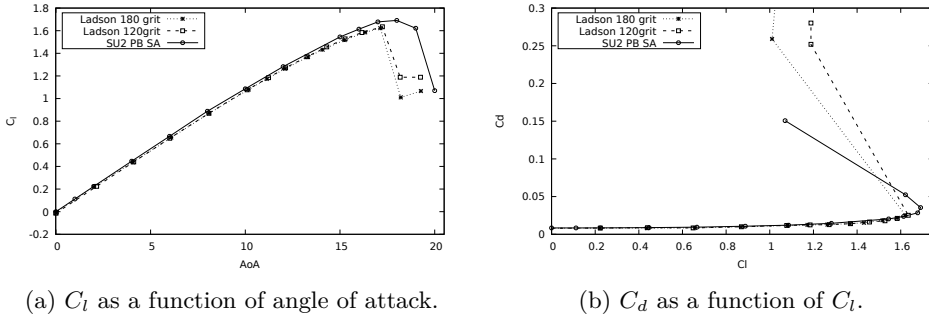

(a) C_l as a function of angle of attack. (b) C_d as a function of C_l.

Figure 4.26: Turbulent flow over the NACA0012 airfoil. A comparison of the numerical solution for the SA RANS turbulence model against the experimental data for the lift coefficient for various angle of attacks (a) and lift to drag (b). Experimental data is listed as Ladson [10] N grit where the number N indicates the grit level of the carborundum paper used.

(a) C_l as a function of angle of attack. (b) C_d as a function of C_l.
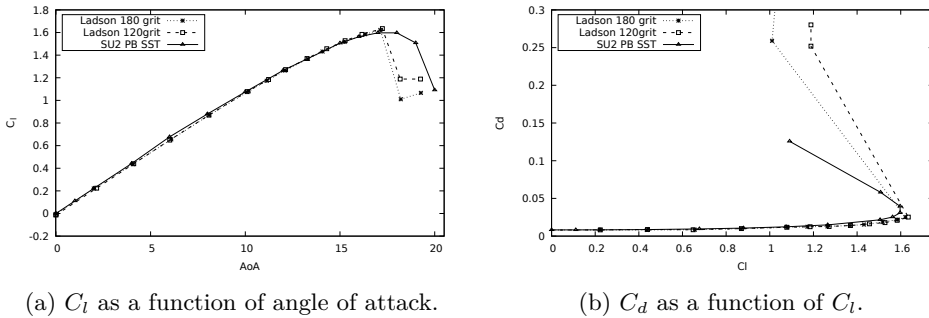
Figure 4.27: Turbulent flow over the NACA0012 airfoil. A comparison of the numerical solution for the SST RANS turbulence model against the experimental data for the lift coefficient for various angle of attacks (a) and lift to drag (b). Experimental data is listed as Ladson [10] N grit where the number N indicates the grit level of the carborundum paper used.

Pressure coefficient The pressure coefficients, C_p, obtained with SU2 are compared against numerical results from CFL3D [1]. The results from the grid with 513 nodes are used. For the sake of clarity of the pictures, the symbols are plotted for only every 4th point. Figure 4.28 shows the comparison of C_p at an angle of attack of 0° for the SA and SST turbulence models. A symmetric distribution is obtained in both cases. Figure 4.29 shows the pressure coefficients at an angle of

(a) C_p from SA.   (b) C_p from SST.

Figure 4.28: Pressure coefficient (C_p) for the SA and SST turbulence models at $AoA = 0°$ for turbulent flow over the NACA 0012 airfoil at a Reynolds number of $Re = 6 \times 10^6$.
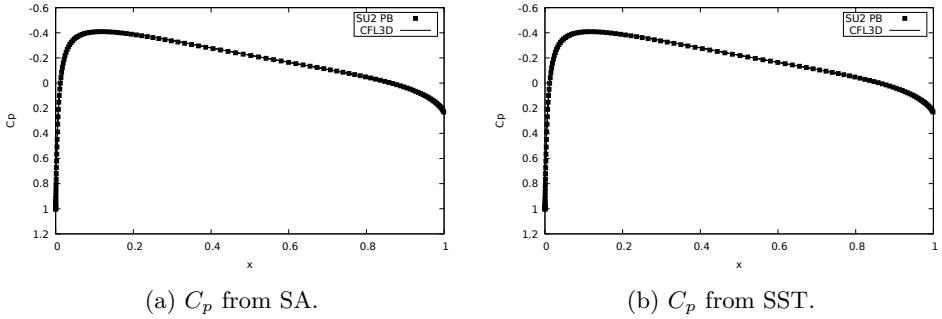
attack of 15°. Results from both the SA and SST model match the results from CFL3D closely. Only the suction side results are available from CFL3D.

(a) C_p from SA.   (b) C_p from SST.
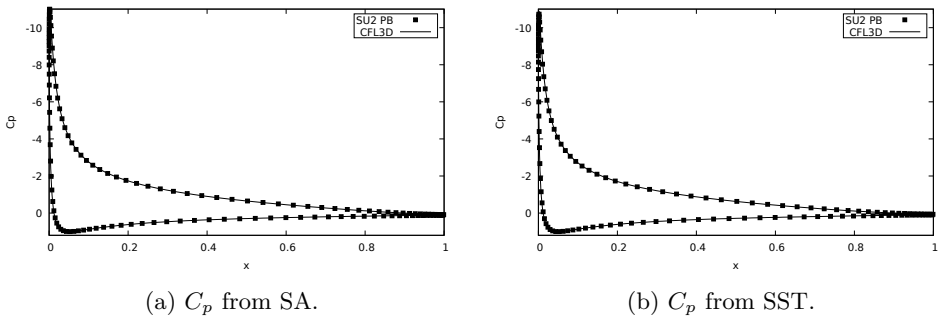
Figure 4.29: Pressure coefficient (C_p) for the SA and SST turbulence models at $AoA = 15°$ for turbulent flow over the NACA 0012 airfoil at a Reynolds number of $Re = 6 \times 10^6$.

Skin friction coefficient   Figure 4.30 shows the comparison of the skin friction coefficient between SU2 and CFL3D at an angle of attack $AoA = 0°$. Once again a symmetric profile is observed for both turbulence models. Figure 4.31 shows the skin friction coefficients at an angle of attack of 15°. Results for both the SA and SST model match the results from CFL3D closely. Only the suction side results are available from CFL3D.

4.2.3. Unsteady flows

Many external aerodynamic flows are unsteady in nature. In this section the unsteady behavior of the pressure based solver under laminar and turbulent flow conditions is tested.
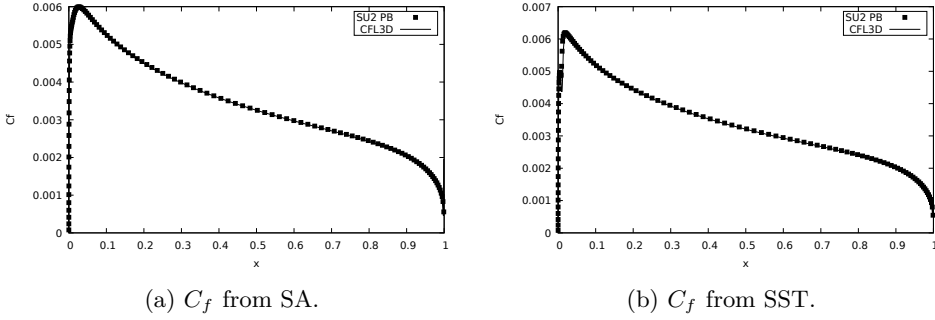
(a) C_f from SA.

(b) C_f from SST.

Figure 4.30: Skin friction coefficient (C_f) for the SA and SST turbulence models at $AoA = 0°$ for turbulent flow over the NACA 0012 airfoil at a Reynolds number of $Re = 6 \times 10^6$.
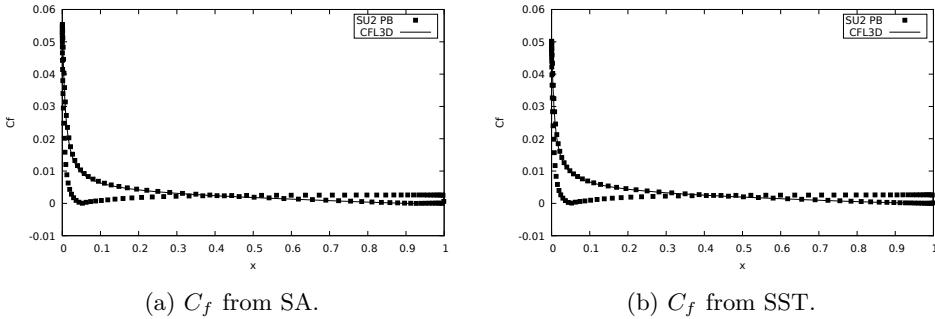

(a) C_f from SA.

(b) C_f from SST.

Figure 4.31: Skin friction coefficient (C_f) for the SA and SST turbulence models at $AoA = 15°$ for turbulent flow over the NACA 0012 airfoil at a Reynolds number of $Re = 6 \times 10^6$.

Square cylinder

In order to validate the unsteady behavior, the flow past a square cylinder that is confined in a channel at a Reynolds number of $Re = 100$ is considered. The flow past a square cylinder has been studied widely under different configurations. At low Reynolds numbers, the flow remains laminar and steady. Kelkar and Patankar [11] determined using a linear stability analysis that the steady flow becomes unstable for Reynolds number of $Re > 53$. At a Reynolds number of $Re = 100$, the flow is unsteady and vorticies are shed from the corners periodically. The flow Reynolds number is defined based on the edge length of the square cylinder, D. The blockage ratio, $B = H/D$, is the ratio of the total height (H) of the confined channel to the edge length of the square cylinder(D). A blockage ratio of $B = 7$ is used in this study. A uniform velocity profile is prescribed at the inlet and the top and bottom walls are treated as slip walls [11]. Fully developed flow is assumed at the outlet and the pressure is set to zero. The domain and the boundary conditions are shown in figure 4.32. Two different meshes are used: a coarse mesh that has 21 nodes on each edge of the cylinder and a fine mesh that has 41 nodes on each edge. The time discretization is carried out using the second order backward Euler scheme.
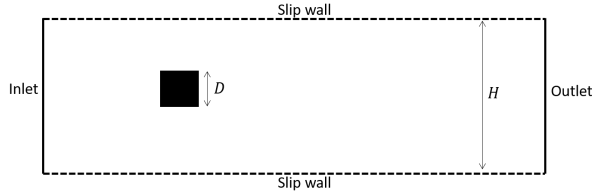
Figure 4.32: Domain and boundary conditions for flow past a square cylinder.

Instantaneous streamlines at two different time instances when the vortex is being shed from one of the two edges are shown in figure 4.2.3. The oscillatory behavior of the flow can be seen in the time histories of the lift coefficient shown in figure 4.34.
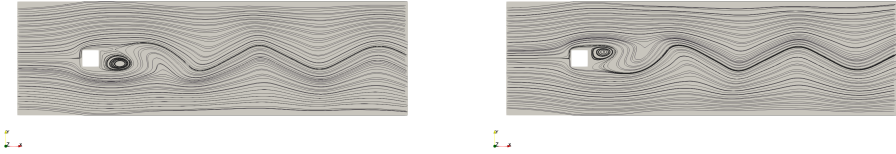
Figure 4.33: Instantaneous streamlines at two different time instances for the flow past a square cylinder.
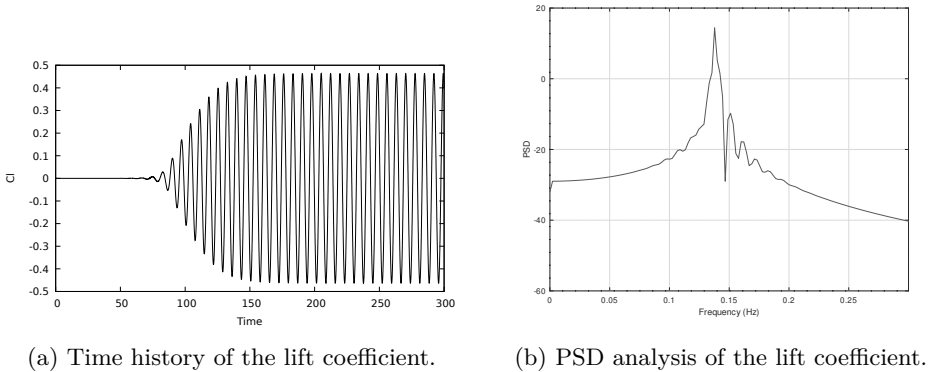
(a) Time history of the lift coefficient.

(b) PSD analysis of the lift coefficient.

Figure 4.34: Variation of lift coefficient with time and PSD analysis for the $N = 41$ and $\Delta t = 0.03s$ case.

The Strouhal number, $St = fD/U$, where f is the frequency of the oscillation in lift, D is the edge length of the square and U is the uniform inlet velocity, is used to characterize the periodicity of vortex shedding. Table 4.6 lists the Strouhal number, St, for the two meshes using different non-dimensional time step sizes. The

Strouhal number found in the literature for the current blockage ratio is 0.126 [11]. The frequency of oscillation of the lift coefficient is found using a power spectral density (PSD) analysis. The resulting Strouhal number of $St = 0.13$ from the fine mesh is close to the reported values in literature.
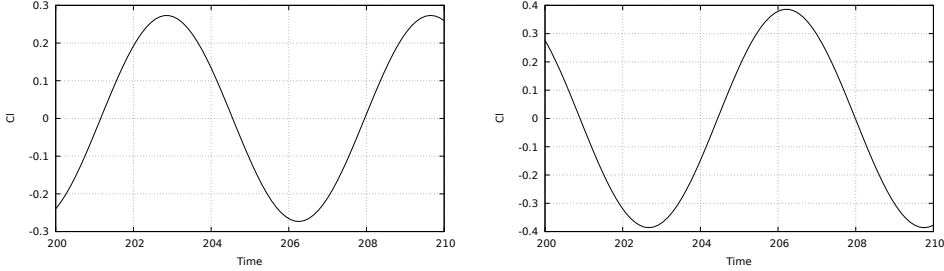
Figure 4.35: Variation of the lift coefficient with time zoomed in to one cycle on the coarse (left) and fine (right) mesh.

N	$\Delta tU/D$	St
21	0.06	0.145
21	0.03	0.144
21	0.015	0.141
41	0.06	0.139
41	0.03	0.138
41	0.015	0.138

Table 4.6: Strouhal number, St, for the two meshes using different non dimensional time steps for the unsteady flow over a square cylinder.

Pitching airfoil
In this section, the unsteady RANS (URANS) capabilities of the pressure based solver is tested by analyzing the NACA 0012 airfoil under dynamic stall conditions. Understanding the dynamic behavior of wind turbine rotors is crucial as they operate in a highly unsteady environment.

Dynamic stall is an unsteady phenomenon that arises as a result of the motion of the airfoil. A pitching airfoil tends to stall at a higher angle of attack than a static one and a hysteresis behavior in the force coefficients is observed as a result of the motion of the airfoil. This of course depends on the amplitude and frequency of pitching. The hysteresis behavior is the result of a lag in phase between the motion of the airfoil and the flow gradients around the airfoil [12]. A vortex can be seen to grow and to be shed from the leading edge region as the airfoil pitches upward. As this vortex moves towards the trailing edge, the lift increases beyond the static stall value and when the vortex passes over and is shed from the trailing edge, a sudden loss in lift is observed. As the airfoil pitches back downward, it remains under deep stall conditions and the flow remains separated causing a hysteresis loop in the force

coefficients with respect to the angle of attack. [13]. As the airfoil continues to pitch downwards, the flow finally reattaches from the leading edge. Based on the period of the airfoil motion and the time scale of the flow, a non dimensional reduced frequency can be defined as

$$k = \frac{\omega c}{2U_\infty},$$

(4.12)

where ω is the frequency associated with airfoil motion like the pitching frequency, c is the airfoil chord and U_∞ is the free stream velocity.

A wind turbine rotor section can undergo dynamic stall behavior under various circumstances during its operation [14]. To define the reduced frequency, an effective velocity, U_{eff},

$$U_{eff} = \sqrt{U_\infty^2 + (r\Omega)^2}$$

(4.13)

is also commonly used instead of the free stream velocity U_∞ for wind turbine applications. Here, r is the radial position of the airfoil section and Ω is the rotation rate of the rotor.

In this section, the dynamic stall behavior of the NACA0012 airfoil at a Reynolds number of $Re = 1 \times 10^6$ is considered. The numerical results are compared to experimental data from McAlister [13]. The grids used for this simulation are the same as those used previously to find the polars and taken from the NASA turbulence modeling database [1]. A grid refinement study is first conducted under steady flow conditions at an angle of attack of 5°. The resulting lift and drag values are listed in table 4.7. Based on the grid refinement study, the grid with $N = 513$ points

N	C_l	C_d
129	0.553	0.0120
257	0.546	0.0119
513	0.545	0.0122

Table 4.7: Lift (C_l) and drag (C_d) coefficients from the SA turbulence model. N denotes the number of points on the surface of the airfoil at an angle of attack of 5°.

on the airfoil is chosen for the unsteady analysis. The non dimensional time step, $\tau = tU_\infty/c$, is set to 0.004 where c is the chord of the airfoil and U_∞ is the free stream velocity. The airfoil is undergoing a pitching motion given by

$$\alpha = \alpha_0 + \alpha_m sin(\omega t),$$

(4.14)

where α is the instantaneous angle of attack, $\alpha_0 = 15°$ is the angle of attack at time $t = 0$, $\alpha_m = 10°$ is the amplitude of the oscillation and ω is the angular frequency of the oscillation. The reduced frequency is set to $k = 0.2$. A first order backward Euler time integration scheme is used. The y^+ of the grid used is less than 1 throughout the airfoil. The pitching motion of the airfoil is simulated by moving the entire mesh as a rigid body.

The lift coefficient as a function of the angle of attack using the SA turbulence model is shown in figure 4.36a. There is a good match between the numerical and experimental data during the upward pitching motion of the airfoil but there

is some difference observed during the downward motion of the airfoil. This is expected as the airfoil is under deep stall conditions and URANS is not accurate enough in this region. The maximum C_l from the numerical results is higher than the experimental values. Figure 4.36b shows the drag coefficient as a function of the angle of attack. Once again, there is a good match between the numerical and experimental data at lower angles of attack in the upswing part of the motion. The stall is predicted to occur later by SA compared to the experiments which is in line with what was observed for steady cases as well. The lift coefficient as a function of

(a) Lift coefficient (C_l).
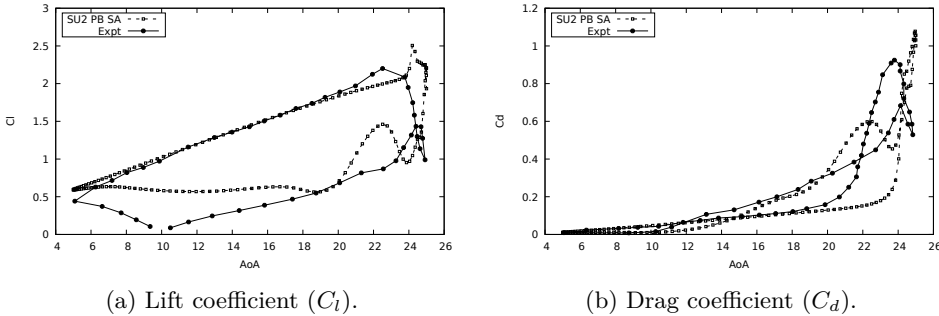

(b) Drag coefficient (C_d).

Figure 4.36: Lift and drag coefficients for a pitching airfoil at Reynolds number $Re = 1 \times 10^6$ using the SA turbulence model.

the angle of attack using the SST turbulence model is shown in figure 4.37a. Similar to the results from the SA turbulence model, there is a good match between the numerical and experimental data during the upward pitching motion of the airfoil. A larger oscillation in lift during the downward motion is observed compared to the SA turbulence model. Similar to the results in the steady state analysis, the maximum C_l from the SST turbulence model is closer to the experimental results compared to the SA turbulence model. Figure 4.37b shows the drag coefficient as a function of the angle of attack. A larger oscillation in drag coefficient compared to the SA turbulence model is observed. The stall is predicted to occur later by the SST compared to the experiments but earlier than the SA model, which is in line with what was observed for steady cases as well.

One of the major reasons for the discrepancy is due to the fact that this was a two dimensional simulation whereas the dynamic stall that occurs during the pitching motion of the airfoil is three dimensional, especially during the downward motion of the airfoil where massive flow separation occurs. Two dimensional URANS methods have a well known limitation in predicting the aerodynamic hysteresis for complex flows [12, 15]. Additionally, the SA turbulence model did not perform as well as the SST turbulence model in capturing the separated flow as evidenced in earlier sections. However, the intention of this test case was to ensure that the new solver behaves as expected under such complex flow conditions.

4.2.4. Rotating flow problems

(a) Lift coefficient (C_l).

(b) Drag coefficient (C_d).
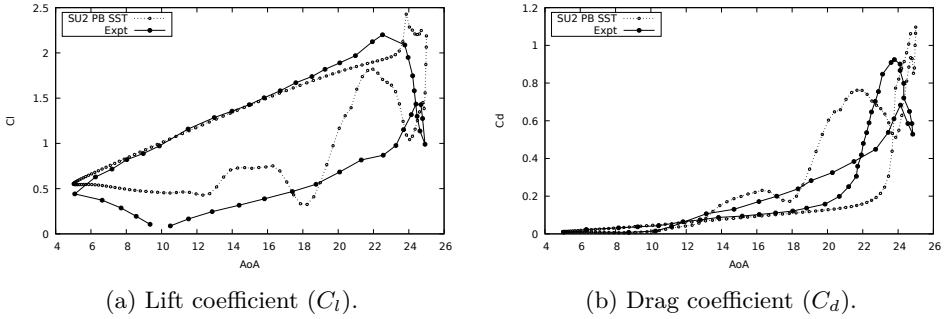
Figure 4.37: Lift and drag coefficients for a pitching airfoil at Reynolds number $Re = 1 \times 10^6$ using the SST turbulence model.

Laminar flow in a rotating rectangular duct

To validate the rotating reference frame implementation, the laminar flow in a rotating rectangular duct at two different rates of rotation is simulated. A long rectangular duct is rotating about one of its sides as shown in the figure 4.38a. The

(a) Schematic [16].

(b) Mesh.
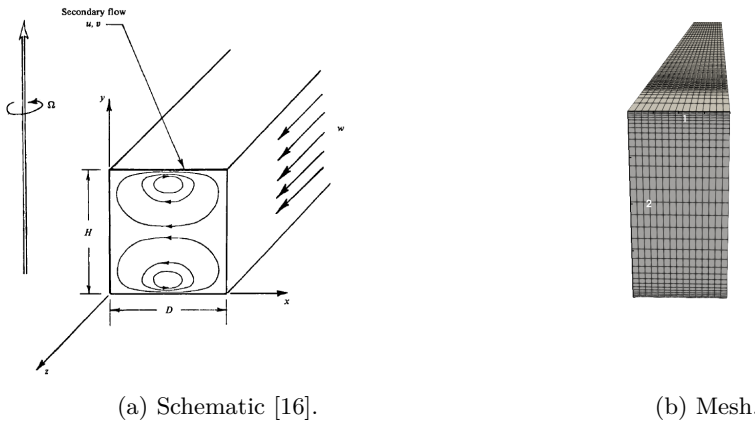
Figure 4.38: Schematic (a) and the mesh (b) for the laminar flow in a rotating low aspect ratio (2 : 1) rectangular duct [16].

relative effects of the rotation rate compared to the axial flow can be characterized by a non dimensional number known as the Rossby number, Ro, which is defined as

$$Ro = \frac{U}{2\Omega D}, \tag{4.15}$$

where U is the reference velocity, Ω is the rotation rate and D is the channel depth (see figure 4.38a. The Rossby number is analogous to the inverse of the tip speed ratio commonly used in wind turbine applications. For low aspect ratio ducts, at low rotation rates and consequently high Rossby numbers, a counter-rotating

double vortex configuration for the secondary flow appears. As the rotation rate increases (and the Rossby number decreases), this double vortex configuration breaks down into four counter rotating vortices. Further increase of rotation rate leads to a restabilization of the secondary flow into a slightly asymmetric double vortex configuration [16].

In this study, a 2×1 rectangular duct is chosen. Two different flow conditions are studied: first at a moderate rate of rotation at a Reynolds number of $Re = 235$ and Rossby number of $Ro = 50.5$ and the second at a relatively rapid rate of rotation at a Reynolds number of $Re = 86$ and $Ro - 1.85$. The depth of the duct is $D = 1m$, the height, $H = 2m$ and the axial length, $L = 25m$, is chosen to be sufficiently long to ensure the flow becomes fully developed and independent of the axial coordinate. Three different grids are used and the numerical results are compared to the results from Speziale [16]. The coarsest grid has 16 nodes along the width and 32 nodes along the height of the duct. The number of nodes along the width and height are doubled to 32 and 64 respectively in the refined grid and doubled again to 64 and 128 respectively in the finest grid. Speziale uses a modified vorticity stream function formulation of the incompressible Navier Stokes equations to resolve the pressure velocity coupling.

Moderate rotation rate

The Reynolds number for this case is $Re = 235$ and the Rossby number is $Ro = 50.5$. The rotation rate is moderately small and thus the axial velocity profile deviates from the symmetric parabolic profile. The axial velocity profiles along the horizontal and vertical centerlines in the fully developed region is shown in figures 4.39a and 4.39b. The axial velocity along the horizontal line matches the numerical results from Speziale [16] but the velocity along the vertical centerline is over predicted in all the three grids compared to the numerical results from Speziale. However, the velocity profiles behave in the same qualitative way compared to the non rotating simulations.

(a) Horizontal centerline.　　　　(b) Vertical centerline.

Figure 4.39: Axial velocity profiles for the laminar flow in a rotating rectangular duct at a Reynolds number of $Re = 235$ and Rossby number of $Ro = 50.5$ along the horizontal (a) and vertical (b) centerline in the fully developed region.

Rapid rotation rate

The Reynolds number for this case is $Re = 86$ and the Rossby number is $Ro = 1.85$. The axial velocity along the horizontal and vertical centerlines in the fully developed region is shown in figures 4.40a and 4.40b respectively. The numerical results are closer to the numerical results from Speziale [16] along the vertical centerline compared to the moderate rotation case. However, a deviation is observed in the velocity profile along the horizontal centerline. In order to obtain a converged solution for this case, the simulation was first initialized at a lower rotation rate. The rotation rate was increased gradually till the desired value was reached.
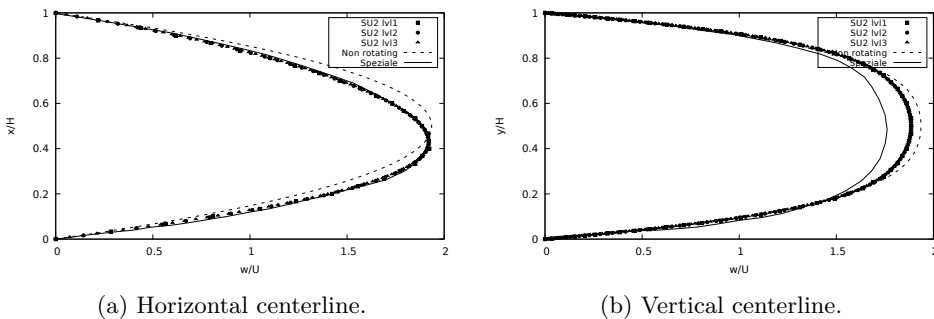

(a) Horizontal centerline.

(b) Vertical centerline.
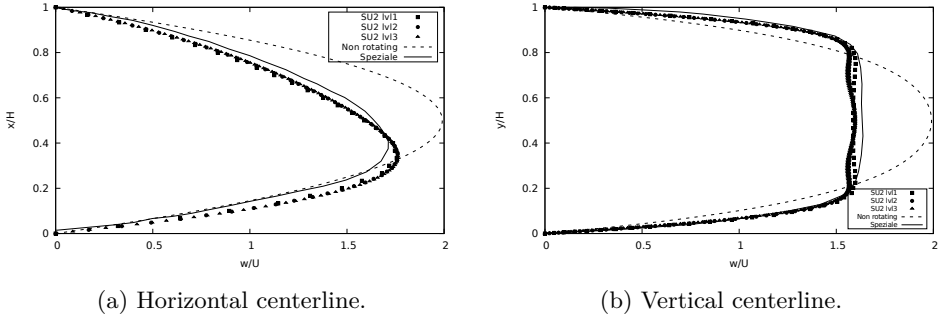
Figure 4.40: Axial velocity profiles for the laminar flow in a rotating rectangular duct at a Reynolds number of $Re = 86$ and Rossby number of $Ro = 1.85$ along the horizontal (a) and vertical (b) centerline in the fully developed region.

The mismatch in the results could be partially explained by the velocity reference values used. The reference value of the velocity used by Sepziale [16] is not very clear and the numerical results from SU2 were non dimensionalized using the inlet velocity.

4.3. Conclusions

The new pressure based solver is shown to be second order accurate in space by verifying the numerical results against analytical solutions for the Taylor-Couette flow and the plane Poisuelle flow problems. Subsequently, the new solver was validated against experimental and other reference data. The new solver is capable of modeling laminar and turbulent flows under different circumstances and can also be used in unsteady conditions where complex flow phenomena occur.

References

[1] C. Rumsey, *Nasa turbulence modeling resource,* https://turbmodels.larc.nasa.gov/index.html (2019).

[2] G. Ghia and Shin, *High-Re solutions for incompressible flow using Navier-Stokes equations and a multigrid method,* Journal of Computational Physics **48** (1982).

[3] G. I. Taylor, *Stability of a viscous liquid contained between two rotating cylinders,* Philosophical Transactions of the Royal Society of London. Series A, Containing Papers of a Mathematical or Physical Character **223**, 289 (1923).

[4] H. Schlichting and K. Gersten, *Boundary-layer theory* (Springer, 2016).

[5] R. Gautier, D. Biau, and E. Lamballais, *A reference solution of the flow over a circular cylinder at Re=40,* Computers & Fluids **75**, 103 (2013).

[6] L. F. Richardson and R. T. Glazebrook, *The approximate arithmetical solution by finite differences of physical problems involving differential equations, with an application to the stresses in a masonry dam,* Philosophical Transactions of the Royal Society of London. Series A, Containing Papers of a Mathematical or Physical Character **210**, 307 (1911).

[7] D. K. Gartling, *A test problem for outflow boundary conditions—flow over a backward-facing step,* International Journal for Numerical Methods in Fluids **11**, 953 (1990).

[8] S. B. Pope, *Turbulent Flows* (Cambridge University Press, 2000).

[9] D. M. Driver and H. L. Seegmiller, *Features of a reattaching turbulent shear layer in divergent channelflow,* AIAA journal **23**, 163 (1985).

[10] C. L. Ladson, *Effects of independent variation of Mach and Reynolds numbers on the low-speed aerodynamic characteristics of the NACA 0012 airfoil section*, Technical Report NASA-TM-4074 (NASA, 1988).

[11] K. M. Kelkar and S. V. Patankar, *Numerical prediction of vortex shedding behind a square cylinder,* International Journal for Numerical Methods in Fluids **14**, 327 (1992).

[12] G. Martinat, M. Braza, Y. Hoarau, and G. Harran, *Turbulence modelling of the flow past a pitching naca0012 airfoil at 105 and 106 reynolds numbers,* Journal of Fluids and Structures **24**, 1294 (2008).

[13] K. W. McAlister, L. W. Carr, and W. J. McCroskey, *Dynamic stall experiments on the NACA 0012 airfoil*, Tech. Rep. TP 1100 (NASA, 1978).

[14] Y. Kim and Z.-T. Xie, *Modelling the effect of freestream turbulence on dynamic stall of wind turbine blades,* Computers & Fluids **129**, 53 (2016).

[15] S. Wang, D. B. Ingham, L. Ma, M. Pourkashanian, and Z. Tao, *Numerical investigations on dynamic stall of low reynolds number flow around oscillating airfoils,* Computers & fluids **39**, 1529 (2010).

[16] C. G. Speziale, *Numerical study of viscous flow in rotating rectangular ducts,* Journal of Fluid Mechanics **122**, 251–271 (1982).

4

5

Wind energy applications: Effect of Vortex Generators on a turbulent boundary layer

In this chapter the effect of a vortex generator (VG) on the turbulent boundary layer is studied. 3D CFD simulations of the flow past a VG on a flat plate are carried out and the resulting boundary layer is compared to the standard turbulent boundary layer. As a first step towards modeling the effect of VGs in integral boundary layer (IBL) theory based methods, a 2D CFD simulation is carried out separately where the VG is represented by a solid wall of zero thickness normal to the flat plate. This simulation is used to understand the limitations that arise out of modeling VGs in a 2D boundary layer which would be the case when using IBL methods. Additionally, IBL methods require closure relations in order to find a solution to the boundary layer. Different closure relations are used for laminar and turbulent boundary layer because of the difference in the nature of the two boundary layers. The introduction of a VG will further modify the nature of the boundary layer and thus a different parametrization is required. The mixing layer will be used as a first step towards forming such a parameterization.

5.1. Introduction

Vortex generators (VGs) are commonly used to improve the performance of wind turbine blades. The concept of a vortex generator is quite old and there has been a considerable amount of research into analyzing the effects of vortex generators [2, 3].

Vortex generators energize the boundary layer and allow for the boundary layer to remain attached for longer [4, 5] and have been successfully used to increase lift produced by airfoil sections and delay stall [6]. Prior research on modeling VGs in CFD methods can be found in eg. references [7] and [8]. Typically, VGs are modeled by performing a Computational Fluid Dynamics(CFD) analysis of wind turbine blade sections either using a body fitted mesh or by the BAY model [9] or its variations. [10] Alternatively, other approaches using vortex strengths are also available [11].

However, the above mentioned modeling methods all require expensive CFD simulations. Performing such $3D$ CFD simulations during the design phase of wind turbines is prohibitively expensive and a simpler solution can be advantageous. For high Reynolds number flows, like the flow past wind turbine rotors, the effect of viscosity is confined to a small region around the body known as the boundary layer and the flow is largely inviscid further away from the body. The Navier-Stokes equations can thus be simplified for the two regions with matching solutions at the interface, allowing for faster computational times. The flow away from the body can be described by the inviscid Euler equations and can be approximated by faster methods like a panel method. The boundary layer equations in the vicinity of the body can be derived from the Navier-Stokes equations in the limit of large Reynolds numbers. The boundary layer equations can then be further simplified by integrating along the wall normal direction to obtain the integral boundary layer equations. Fast solution methods based on the solution of integral boundary layer equations together with the inviscid potential equations coupled with an appropriate viscous-inviscid interaction scheme [12–14] (also called interacting boundary layer method) are routinely used to optimize shape of airfoils. Incorporating the effects of blade add-ons like VGs into such tools will allow for better and more efficient designs at a fraction of the cost compared to full three-dimensional CFD methods.

The main difficulty of this task is that while the vortices induced by the VGs are three dimensional in nature, the integral boundary layer methods are typically used to model two dimensional boundary layers. In this study, the nature of the turbulent boundary layer behind VGs are examined and the requirements for an approximate model that can capture the effects of VGs on a two-dimensional integral boundary layer method are identified.

In the section 5.2 the boundary layer separation is examined for laminar and turbulent flows. Section 5.3 presents the results from the $3D$ CFD simulations of the flow past a VG on a flat plate. The similarities and differences of representing the VG in $2D$ will also be examined. In section 5.4, a mixing layer parametrization is used to explore the qualitative behavior of the $3D$ and $2D$ turbulent boundary layers. Section 5.5 will briefly describe the integral boundary layer methods and the methods used to derive the requisite closure relations for laminar and turbulent flows. Further the differences in the boundary layer due to the VG will be highlighted.

5.2. The boundary layer

To understand how the VG can affect the boundary layer, a brief overview of the behavior of laminar and turbulent boundary layers is presented. Finally, the effect of the VG on a turbulent boundary layer is examined qualitatively.

5.2.1. Laminar boundary layer

The boundary layer plays a crucial role in determining the flow behavior at high Reynolds number. Due to the no-slip condition at the wall, the incoming flow slows down and the velocities are zero at the wall and gradually away from the wall increases until it matches the edge velocity, u_e, at the edge of the boundary layer (approximately the free-stream velocity). The distance from the wall to the edge of the boundary layer is known as the boundary layer thickness, δ. Based on the boundary layer theory, and assuming the flow to be mostly aligned with the x direction, the boundary layer equations for steady two dimensional flows can be derived starting from the non-dimensional Navier-Stokes equations (equations 2.7 and 2.8 in chapter 2) and are given by

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0, \tag{5.1}$$

$$\frac{\partial u^2}{\partial x} + \frac{\partial uv}{\partial y} = -\frac{1}{\rho}\frac{\partial p}{\partial x} + \nu\frac{\partial^2 u}{\partial y^2}, \tag{5.2}$$

$$\frac{\partial p}{\partial y} = 0. \tag{5.3}$$

The equation 5.1 is the continuity equation which remains unchanged under the assumption of large Re numbers ($Re >> 1$). The momentum equation in the direction normal to the wall (y) is reduced to the condition that the normal pressure gradient is zero across the boundary layer (equation 5.3). Thus the pressure in the boundary layer is imposed on it by the flow outside the boundary layer. And this flow outside the boundary layer is governed by the inviscid Euler equation as viscous effects are negligible in this region. Applying the momentum equation, equation 5.2, at the edge of the boundary layer, where $v = 0$, an equation for for pressure gradient in terms of edge velocity can be found as

$$\frac{1}{\rho}\frac{\partial p}{\partial x} = -\frac{\partial u_e^2}{\partial x}. \tag{5.4}$$

This relation is essentially the Bernoulli equation for inviscid flow along a streamline which is expected to be valid at the edge of the boundary layer where the flow becomes inviscid and viscous stresses are negligible.

Depending on the external pressure gradient, the slow moving boundary layer can either remain attached or separate from the wall. Boundary layers are prone to separation under adverse pressure gradients (figure 5.1). Consider a simplified analysis and neglect the normal (y) component of velocity, v, for the momentum

equation 5.2 (this assumption is not far from reality in most boundary layer flows)

$$u\frac{\partial u}{\partial x} = -\frac{\partial p}{\partial x} + \nu\frac{\partial^2 u}{\partial y^2}. \tag{5.5}$$

In an adverse pressure gradient, where the pressure p is increasing, the pressure gradient, $\frac{\partial P}{\partial x}$ is positive and only the shear stress contribution is preventing the flow from separating. Very near to the wall, the shear stress is essentially the wall stress, τ_w and the flow will separate once the wall stress contribution is not enough to overcome the adverse gradient. In the absence of a pressure gradient (like flat plates), $\frac{\partial p}{\partial x} = 0$, separation occurs if the wall stress becomes zero ($\tau_w = 0$) or an inflection point appears in the boundary layer profile.
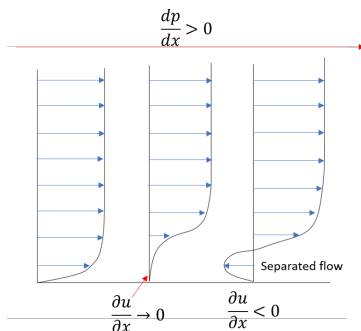

Figure 5.1: A schematic visualization of the boundary layer separation.

5.2.2. Turbulent boundary layer

Adding turbulence to flow has a significant effect on the boundary layers. Due to increased energy transfer within the flow, there is now a significant amount of energy even in the boundary layer. However, due to the no-slip condition the velocity at the wall must still go to zero which leads to an even larger gradient very near the wall. Using the Reynolds averaging procedure (see section 2.6.1), the turbulent Reynolds stresses are represented in terms of the eddy viscosity, ν_t, and the mean flow gradients (or strain rate) and the equation 5.5 becomes

$$u\frac{\partial u}{\partial x} = -\frac{1}{\rho}\frac{\partial p}{\partial x} + (\nu + \nu_t)\frac{\partial^2 u}{\partial y^2}. \tag{5.6}$$

Since the eddy viscosity is positive and the wall shear stress is larger, it is easy to see that the turbulent flow separates later than laminar flow and can overcome larger adverse pressure gradients.

5.2.3. Vortex generator in the boundary layer

Adding a vortex generator (VG) has an analogous effect on turbulent boundary layers as the introduction of turbulence had on laminar boundary layers [3]. As the

name suggests, a vortex generator generates vortices that entrain high energy fluid from outside of the boundary layer and mix it with the boundary layer flow. The additional momentum and energy introduced to the boundary layer help to overcome more severe adverse pressure gradients and the boundary layer can remain attached for longer [3].

The vortex generator is assumed to be submerged in the boundary layer (low-profile VG [2] or sub boundary layer VG [15]), but to extend well into the outer layer of the boundary layer. For instance, the height of the vortex generators used in experiments typically range from $5mm$ [6] to about $10mm$ [16]. From the operational Reynolds numbers in the experiments, one can compute the height of the vortex generators to be approximately between $y^+ = 1000$ and $y^+ = 2050$ which is far away from the inner boundary layer where viscous stresses from the wall are dominant. However, the additional shear layer introduced due to mixing will now create a new region within the boundary layer where viscous stresses become significant again away from the wall. Unlike a fully turbulent boundary layer where the outer layer is completely dominated by Reynolds stresses, the VG will introduce additional mean flow gradients and viscous stresses in the outer layer as well which will also lead to an increase in the production of turbulent kinetic energy within the boundary layer.

Thus the presence of mixing induced by VGs will increase viscous dissipation due to the additional mean flow gradients. The new eddy viscosity, ν_t' (which is different than ν_t), will also increase due to additional production of turbulent kinetic energy. Thus the extra viscous and eddy dissipation will lead to a more controlled rate of growth of the boundary layer and delay separation. This can also be verified by looking at the simplified relations in equations 5.5 and 5.6. If the VG were to introduce an additional term to counter the pressure gradient,

$$u\frac{\partial u}{\partial x} = -\frac{1}{\rho}\frac{\partial p}{\partial x} + (\nu + \nu_t')\frac{\partial^2 u}{\partial y^2} + \text{viscous dissipation due to VG.} \qquad (5.7)$$

The additional viscous dissipation and a larger ν_t' can help the boundary layer to overcome an even larger adverse pressure gradient and remain attached for longer than the turbulent boundary layer (equation 5.6). In order to analyze the effect of VGs more quantitatively, numerical simulations are performed and are described in the following section.

5.3. Numerical simulation

5.3.1. Three dimensional simulation

The pressure based solver is used to simulate the the flow over a flat plate with vortex generator. The geometry of the vortex generator geometry is based on the experimental study of Baldacchino et al [15]. The height of the vortex generators is $h = 5mm$. An array of 15 pairs of vortex generators were used in the experiments. Figure 5.2a shows the schematic of the rectangular vortex generators used in this study. The distance between the trailing edges of the pair of vortex generators is $d = 12.5mm$, the distance between the pairs of vortex generators is $D = 30mm$ and the incidence angle is $\beta = 18°$ (see figure 5.2b).

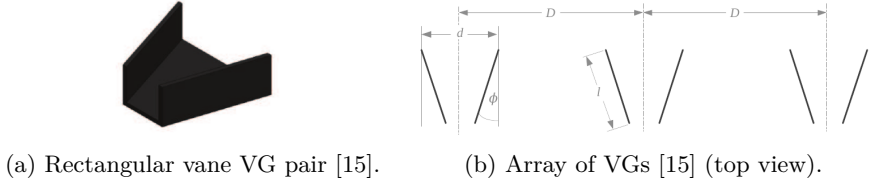(a) Rectangular vane VG pair [15].   (b) Array of VGs [15] (top view).

Figure 5.2: Schematic of a pair of rectangular vortex generators (a) and the mesh of the rectangular pair used in the $3D$ CFD simulations (b).

For the numerical study, only one pair used and a periodic boundary condition is applied on the spanwise boundaries. The domain and the other boundary conditions used for the simulation are shown in figure 5.3a and the vortex generators in figure 5.3b. The trailing edge of the vortex generator is placed at $x_{VG} = 985mm$ from the start of the plate. The domain is $2.1m$ long and $0.25m$ high. There are 82 points in the normal direction with a minimum grid spacing at the wall of $1 \times 10^{-5}m$. There are 816 nodes in the stream wise direction with the nodes clustered in the vicinity of the vortex generator. A simulation without vortex generators on the same domain with the same grid resolutions was also conducted.
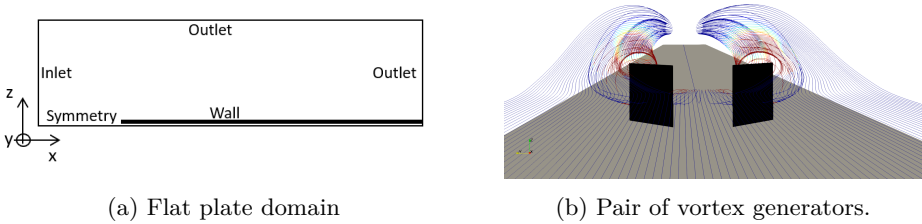

(a) Flat plate domain   (b) Pair of vortex generators.

Figure 5.3: Computational domain used for flat plate simulations (a), VG model (b).

Results
The results for the simulations with VGs are denoted as 'VG' and the simulations without VG as 'Clean'. The results are presented in terms of the scaled stream wise location, s_{VG}, given by

$$s_{VG} = \frac{x - x_{VG}}{h_{VG}},$$

where x_{VG} the location of the trailing edge of the VG and h_{VG} is the height of the VG. The results are presented in different planes as shown in figure 5.4. Here $y = 0$ represents the plane between the VG pair, $y = TE$ represents the plane through the trailing edge of the VG and $y = D/3$ represents the plane $D/3$ away from the centerline. The solution is also extracted along the planes $y = -TE$ and $y = -D/3$ that are symmetrically away from the $y = 0$ plane, as $y = TE$ and $y = D/3$.

Figure 5.5 shows the velocity profile from the clean and VG cases at different streamwise locations downstream of the VG. The velocity profiles in all the planes
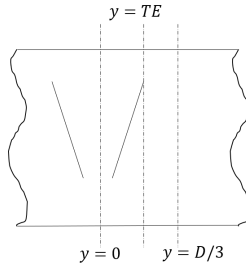
Figure 5.4: Different planes along which the information is presented.

deviate from the clean boundary layer. Figure 5.5a shows the velocity profiles at a location $s_{VG} = -3$ which is located upstream of the leading edge of the VG. As the flow approaches the VGs, deviations are observed along the $y = \pm TE$ planes. Relatively less deviation is observed along other planes because the VG is only physically present along the $y = \pm TE$ plane. Figure 5.5b shows the velocity profiles at a distance of $s_{VG} = 10$. The velocity near the wall is much larger than the clean simulation along the centerline $y = 0$. Along the $y = \pm TE$ plane, the velocity near the wall is higher but the profile around the $z = h_{VG}$ level is still recovering after encountering the trailing edge of the VG. A sharp shear layer appears at this height. A velocity deficit appears along the $y = \pm D/3$ around $z = h_{VG}$ while the profile near the wall is similar to the clean boundary layer. The shear layers formed are confined to a height of $2h_{VG}$. Figure 5.5c shows the velocity along the same planes at a location further downstream, $s_{VG} = 25$. The effect of the VG on the velocity profiles is now spread out over a larger region. The velocity near the wall along the $y = 0$ plane remains higher than the clean boundary layer. The velocity profile along the $y = \pm TE$ planes near the wall is also larger than the clean boundary layer and the shear layer formed due the trailing edge of the VG is now more diffuse. Further downstream at $s_{VG} = 50$ (figure 5.5d), the effect of the VG has spread over an even larger distance in the direction normal to the wall. The velocity close to the wall is larger than the clean simulations along all the planes and shear layers formed have diffused. The effect of the VG can now be seen upto a height of approximately $4h_{VG}$ along all the planes. The deviations in the velocity profiles in all the planes are similar to those reported in Baldacchino et al [15].

Figure 5.6 shows the spanwise variation of the velocity at different streamwise locations at three different heights normal to the wall. The first two locations at a height of $z = h_{VG}$ and $z = 2h_{VG}$ are well within the boundary layer and the third location at a height of $z = 6h_{VG}$ is just outside the edge of the boundary layer. Figure 5.6a shows the velocity profiles just upstream of the VG. The flow has slowed down in the vicinity of the VG at a height of $z = h_{VG}$ while the other two locations remain largely unaffected. Figure 5.6b shows the velocity profiles at a location $s_{VG} = 10$ and the effect of the vortices can be seen clearly at heights $z = h_{VG}$ and $z = 2h_{VG}$. The two troughs in velocity observed at $z = h_{VG}$ correspond to the $y = \pm TE$ planes. At $z = 2h_{VG}$, the flow is affected only in the region between $y = 0$ and $y = \pm TE$. This was also observed in the normal velocity profiles in
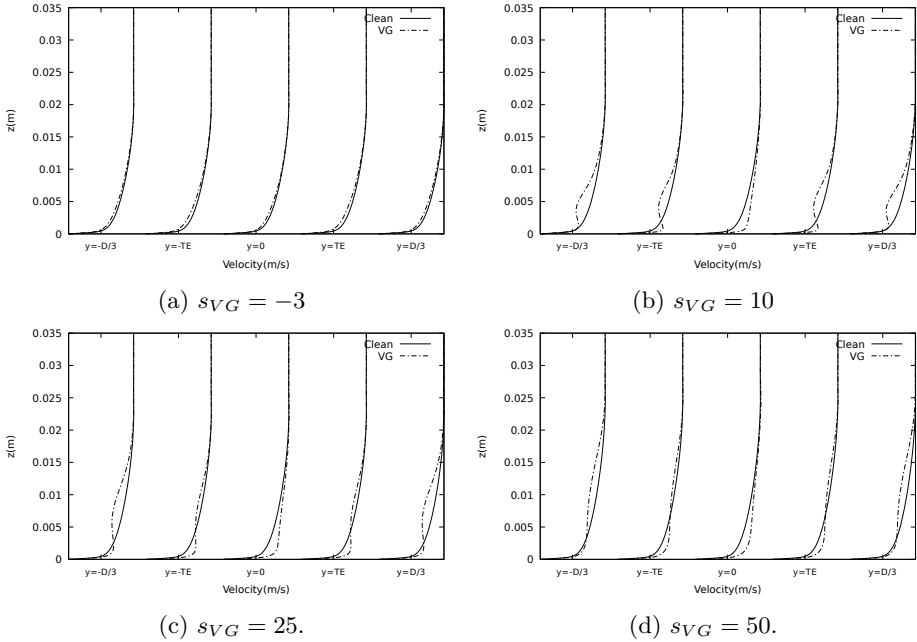
(a) $s_{VG} = -3$

(b) $s_{VG} = 10$

(c) $s_{VG} = 25$.

(d) $s_{VG} = 50$.

Figure 5.5: Boundary layer velocity profile from the $3D$ simulations at different spanwise locations.

figure 5.5b. At a location $s_{VG} = 25$ (figure 5.6c), the vortices appear to have spread out wider beyond the $y = \pm TE$ planes and a greater influence is observed at $z = 2h_{VG}$. Further downstream at $s_{VG} = 50$ (figure 5.6d) the effect of the vortices is more diffuse at $z = h_{VG}$, but stronger at $z = 2h_{VG}$. The velocity just outside the boundary layer at $z = 6h_{VG}$ remains unaffected at all the streamwise locations.

Discussion   As expected, the flow is symmetric about the centerline, $y = 0$. While such a symmetry is natural at the location of the VG due to the periodic array of VGs, this symmetry extends far downstream after the VG as seen in figure 5.6d. Similar analysis must be performed for airfoils with VGs to verify this behavior. The spanwise variation in the flow appears to be dominated by the orientation of the VG upstream. This strong influence of the VG orientation on the flow far downstream can be useful to understand the behavior of the turbulent boundary layer using faster methods using simplified equations like the integral boundary layer equations.

The integral boundary layer methods are used for two dimensional flow analysis and while the flow behind a VG is not strictly two dimensional, the spanwise interaction occurs only around the VG and the flow downstream develops as a two dimensional boundary layer. In order to understand the differences between the $3D$ flow field around a VG and a strictly two dimensional boundary layer as modeled using IBL methods, a $2D$ simulation of only the $y = TE$ plane is carried out and
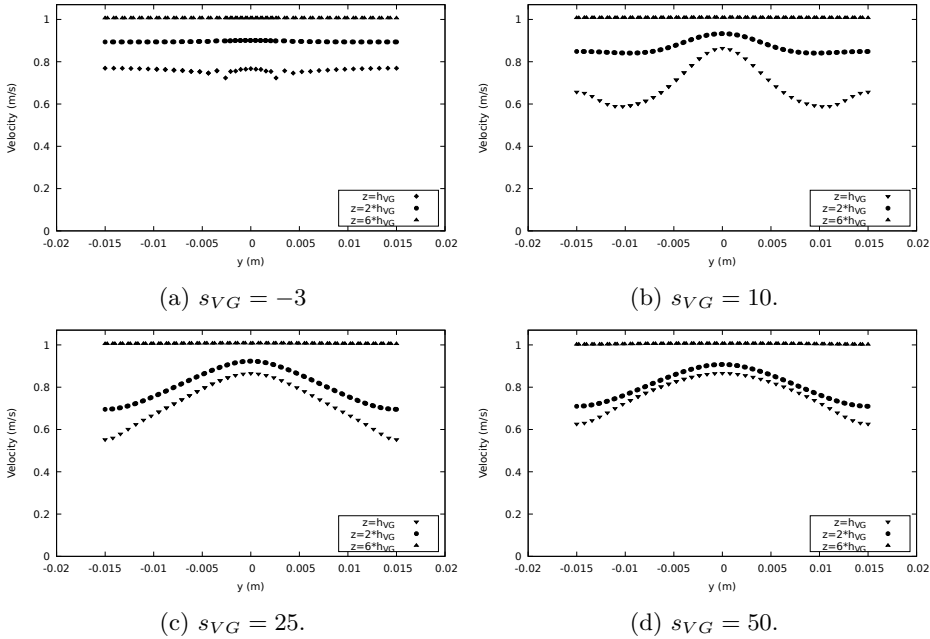
(a) $s_{VG} = -3$.

(b) $s_{VG} = 10$.

(c) $s_{VG} = 25$.

(d) $s_{VG} = 50$.

Figure 5.6: Results from the $3D$ simulations along different planes at different spanwise locations.

described in the following section.

### 5.3.2. Two dimensional numerical simulation

$2D$ CFD simulations are performed under the same conditions as before using the Spalarat-Allmaras turbulence models for steady, incompressible flow conditions, first for no VG (clean) cases and then for cases including the VG under the same flow conditions as the $3D$ simulation. The flat plate is $4.0m$ long and the VG is placed at $x_{VG} = 0.985m$ corresponding to the $y = TE$ plane. The trailing edge of the VG is represented in the $2D$ domain as a zero thickness solid wall of the same height, h_{VG} perpendicular to the flat plate. The boundary conditions used are the same as those shown in figure 5.3a.

Figure 5.7a shows that the effect of the VG on the velocity profile are observed as far upstream as $-10h_{VG}$ and extends up to approximately $100h_{VG}$ (figure 5.7b). However, significant differences in the velocity profile are observed over a much narrower range. In figure 5.8 the comparison between the $2D$ and $3D$ simulations can be seen. Evidently, the behavior of the near wall region is completely different in the $2D$ and $3D$ cases very close to the VG as seen in figure 5.8a, but the profiles appear to behave similarly albeit with an offset as the flow moves further downstream in figure 5.8b.

Figure 5.9a shows the eddy viscosity in the boundary layer for the $2D$ simulation at various locations upstream and downstream of the VG. The effect of the VG is felt

(a) Before VG

(b) Downstream of VG
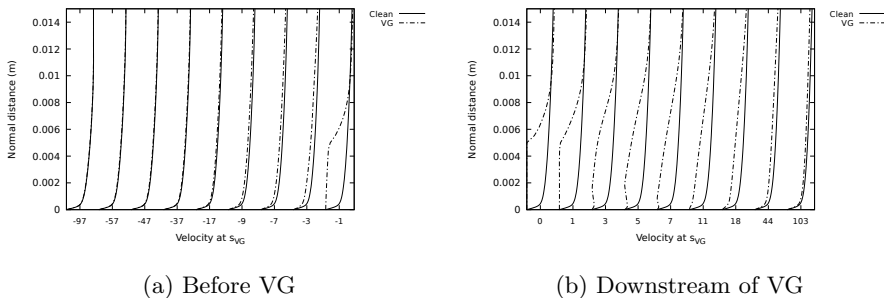
Figure 5.7: Comparison of velocity profiles with and without VG at various x locations.

(a) around VG

(b) downstream of VG
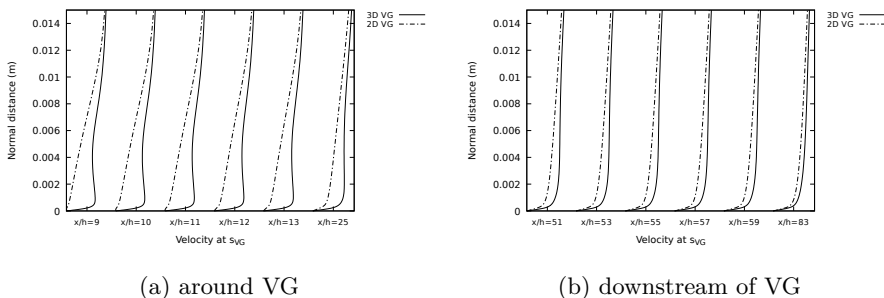
Figure 5.8: Comparison of velocity profiles with VG between $2D$ and $3D$ simulations at various x locations.

slightly upstream starting in the $2D$ simulation. Figure 5.9b shows the comparison of the eddy viscosity in the boundary layer between the $2D$ simulation and the $3D$ simulation along the $y = TE$ plane. Once again, near the VG, the behavior of the two cases are very different. As the flow moves further downstream, both cases behave qualitatively similarly but there is a difference in the magnitudes of the eddy viscosity.

(a) $2D$ clean and VG comparison

(b) $2D$ and $3D$ comparison
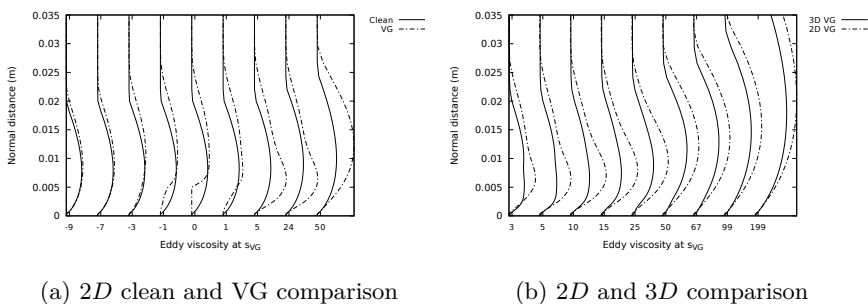
Figure 5.9: Comparison of eddy viscosity profiles between (a) clean and VG simulations and (b) $2D$ and $3D$ simulations at various x locations.

The velocity and eddy viscosity results, especially in the near vicinity of the VG indicate that the mixing process in this region are very different in $2D$ and $3D$ cases. However, encouragingly, the results further downstream display similar qualitative behavior. While representing the effect of the VG in $2D$ by a solid wall of zero thickness is not accurate, the intention of this simulation was to check if the same mixing process that occurs due to the VG can be approximated to some degree in $2D$. To verify this, a brief introduction of the mixing layer is given below and the results from the two simulations are examined further.

5.4. Mixing layer

The plane mixing layer is a free shear flow and is widely studied [3, 17]. Earlier work (e.g., see, [18],[19] and [20]) in studying mixing layers have reported some correlations for parameters like momentum thickness (of the mixing layer), width of mixing region, eddy diffusivity and spreading rate under zero pressure gradient and adverse pressure gradients [19, 20]. However, most of these studies are conducted in the context of free shear flows and assume that the mixing layer can spread without any constraints. This will not be the case here as the wall will significantly impact the spreading of the mixing layer in one direction. As no external pressure gradient exists in the flow over a flat plate, a self similar velocity profile can be expected to form within the boundary layer region [18]. However, due to presence of the wall, the similarity profiles are likely to be very different compared to the standard plane mixing layer.

For a mixing layer [17] (figure 5.10), two imposed velocities, U_h and U_l of the two parallel streams can be defined (figure 5.10). Based on these velocities, a character-
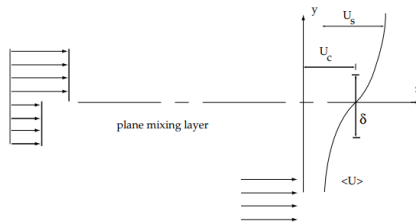

Figure 5.10: An example of a plane mixing layer.

istic convection velocity, U_c, and a characteristic velocity difference, U_s, is defined as,

$$U_c \equiv \frac{1}{2}(U_h + U_l),$$
$$U_s \equiv U_h - U_l.$$

In the present case, the mixing layer can be assumed to form in the vicinity of the VG as the flow above $y = h_{VG}$ mixes with the flow below it. The characteristic

velocities, U_h and U_l, are however not uniform and need to be computed. These characteristic velocities are found as follows,

$$U_l = \frac{\int_0^{h_{VG}}(\rho u)u dy}{\int_0^{h_{VG}}(\rho u)dy}, \tag{5.8}$$

$$U_h = \frac{\int_{h_{VG}}^{\delta}(\rho u)u dy}{\int_{h_{VG}}^{\delta}(\rho u)dy}. \tag{5.9}$$

In both equations, the numerator represents the momentum flux and the denominator represents the mass flux within the bounds of integration. Thus, U_h and U_l can be viewed as the average velocity with which the mass flux is convected in the boundary layer. The stream wise location where this integration is carried out is important since the mixing layer is assumed to form immediately after the VG.

To define the characteristic width of the mixing layer based on the local mean velocity, U, a new weighting factor, α, is introduced such that

$$U = U_l + \alpha(U_h - U_l), \tag{5.10}$$

and then the width, $w(x)$, defined as

$$w(x) = z_{\alpha=0.9}(x) - z_{\alpha=0.1}(x), \tag{5.11}$$

and a reference lateral position is defined as

$$\overline{w}(x) = \frac{1}{2}(z_{\alpha=0.9}(x) + z_{\alpha=0.1}(x)). \tag{5.12}$$

Here $z_{\alpha=0.9}$ represents the location where the local velocity can be found by setting $\alpha = 0.9$ in equation 5.10 and analogously $z_{\alpha=0.1}$ is the location where $\alpha = 0.1$. Based on these definitions, a scaled wall normal distance can be defined as,

$$\xi = \frac{z - \overline{w}(x) - h_{VG}}{w(x)}, \tag{5.13}$$

and the scaled velocity as

$$f(\xi) = \frac{U - U_c}{U_s}. \tag{5.14}$$

For a plane mixing layer, the scaled velocity must be self similar as shown in figure 5.11. The scaled velocity profile is bound by the two velocities of the two streams, U_h and U_l. The upper half of the scaled profile tends towards the velocity U_h and the lower half tends towards U_l. These equations will now be applied to the results from the $2D$ and the $3D$ simulations. The characteristic velocities U_h and U_l are computed at the start of the mixing layer which is assumed to be at $s_{VG} = 3$. x_{VG} is assumed to be the same on all the planes for the $3D$ simulation and is set to $x_{VG} = 0.985m$ corresponding to the TE location the $y = \pm TE$ plane, see figure 5.4.
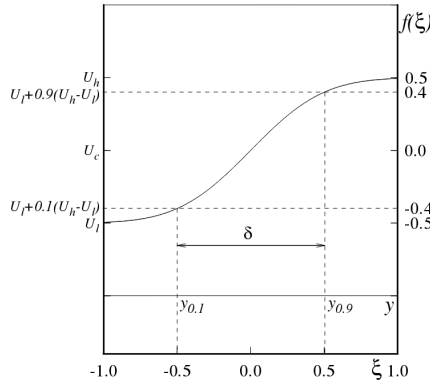
Figure 5.11: Example of a scaled velocity profile in a mixing layer [17].

5.4.1. Three dimensional simulation

The velocities at three different spanwise planes are scaled according to the mixing layer relations and are presented in this section.

$y = 0$ Figure 5.12 shows the scaled velocity as a function of the scaled distance in the plane $y = 0$. All the scaled velocity profiles collapse on a single curve, especially away from the VG location. The collapse is not as good as the analytical example because the two mixing velocities considered here are not uniform. The scaled velocities deviate from the plane mixing layer profile in the lower half as the effect of the wall is dominant.

(a) $s_{VG} = 3$ to $s_{VG} = 12$.   (b) $s_{VG} = 14$ to $s_{VG} = 30$.

Figure 5.12: Scaled velocity and cross stream distance on the $y = 0$ plane.

$y = \pm TE$ Figure 5.13 shows the scaled velocity as a function of the scaled distance in the plane $y = \pm TE$. The collapse of the velocities is not as close as in the $y = 0$ plane. The deviation is stronger in the lower half of the profile, which is nearer to the wall. Once again, the velocities in the upper half are more similar as the mixing effect is more dominant than the presence of the wall.

(a) $s_{VG} = 3$ to $s_{VG} = 12$.   (b) $s_{VG} = 14$ to $s_{VG} = 30$.
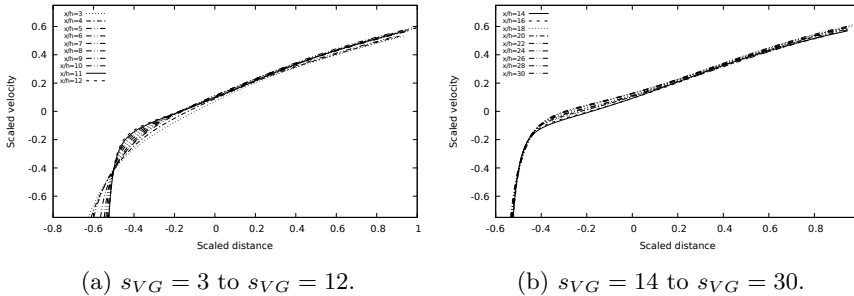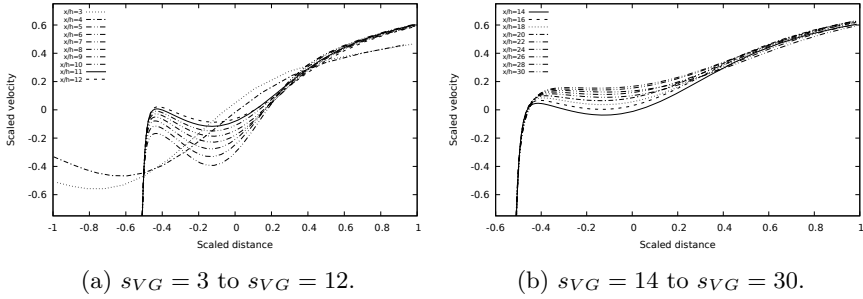
Figure 5.13: Scaled velocity and cross stream distance on the $y = \pm TE$ plane.

$y = \pm D/3$ Figure 5.14 shows the scaled velocity as a function of the scaled distance in the plane $y = \pm D/3$. The collapse of the velocity profiles is better on this plane. There is some deviation in the lower half where the flow encounters the wall, but it is not as much as observed in the $y = \pm TE$ plane.

(a) $s_{VG} = 3$ to $s_{VG} = 12$.   (b) $s_{VG} = 14$ to $s_{VG} = 30$.
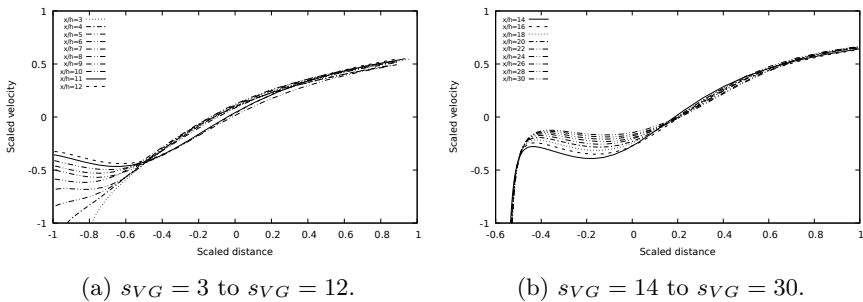
Figure 5.14: Scaled velocity and cross stream distance on the $y = \pm D/3$ plane.

While the scaled velocities do not behave exactly as a plane mixing layer, self similarity is still observed to some extent in all the planes so far. The plane $y = 0$ displays the strongest self similarity indicating the mixing to be the strongest in this plane. In the other planes, the scaled velocities deviate considerably in the lower half of the mixing layer (when the scaled distance is negative). This is due to increasing influence of the wall. The upper half of the scaled profiles match the analytical profile more closely as mixing effects dominate in this region. The scaled velocity profile is bound by the edge velocity of the boundary layer which is close to the value of U_h whereas the velocities in the lower half of the profile are not bound by U_l and reach zero at the wall. Thus, as the effect of the wall becomes more prominent, the scaled velocity profile deviates further away from the analytical one. Also, at the local Reynolds numbers, the y^+ at the h_{VG} is approximately 500. Thus, part of the lower half of the mixing layer is likely interfering with the inner layer of the boundary layer.

5.4.2. Two dimensional mixing layer

Now the plane mixing layer scaling is applied to the results from the $2D$ simulation to compare the difference between the qualitative behavior between the $2D$ and $3D$ cases. The scaled velocity profiles for the $2D$ simulation are shown in figure 5.15. In figure 5.15a, the scaled velocities all collapse on top of each other, however just as with the $3D$ case, the shape of the collapsed curve is different from the standard mixing layer due to the non uniform velocities at the start of the mixing layer. As the flow moves downstream, the width of the mixing layer grows and the presence of the wall inhibits the development leading to truncated profiles as seen in figure 5.15b.

(a) $s_{VG} = 3$ to $s_{VG} = 10$.   (b) $s_{VG} = 11$ to $s_{VG} = 20$
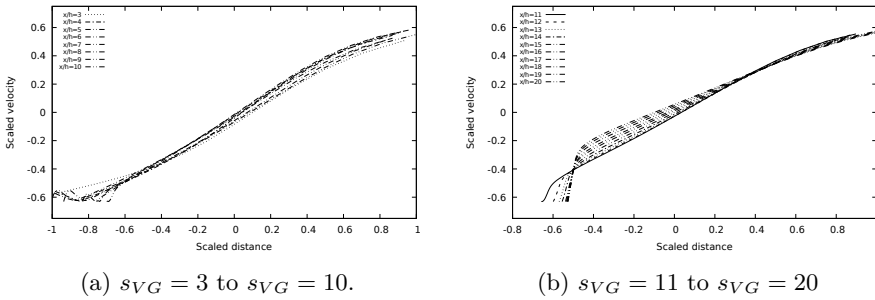
Figure 5.15: Scaled velocity and cross stream distance for the $2D$ simulation.

The scaled velocity profiles show a much closer resemblance to the plane mixing layer in $2D$ than $3D$ which is expected. However away from the wall, both $2D$ and $3D$ simulations show a strong self similarity indicating that the mixing behavior is dominant and is captured in both cases. These results indicate that the mixing layer equations can be used to derive a parametric relation for the velocities. However, as seen in the velocity and eddy viscosity profiles, the behavior in the near wall region is completely different in the $2D$ and $3D$ simulations.

Before exploring the methods to model the effect of vortex generators on turbulent boundary layers using a $2D$ integral boundary layer method, a brief introduction of the integral boundary layer equations is presented.

5.5. Integral boundary layer equations

The integral boundary layer (IBL) equations can be obtained by taking the zeroth and the first moment of the boundary layer equations and integrating them across the direction normal to the wall. The nth moment of the boundary layer equation is defined as [21]

$$[\text{Eq. 5.2}] \times (n+1)u^n - [\text{Eq. 5.1}] \times (u_e^{n+1} - u^{n+1}) = 0. \tag{5.15}$$

Integrating the boundary layer equations in the direction normal to the wall reduces their dimensionality by one. New integral thicknesses are introduced as

$$\delta^* = \int_0^\delta \left(1 - \frac{u}{u_e}\right) dy, \tag{5.16}$$

$$\theta = \int_0^\delta \frac{u}{u_e}\left(1 - \frac{u}{u_e}\right) dy, \tag{5.17}$$

$$\delta^k = \int_0^\delta \frac{u}{u_e}\left(1 - \frac{u^2}{u_e^2}\right) dy. \tag{5.18}$$

Here δ^* is the displacement thickness, θ is the momentum thickness and δ^k is the kinetic energy thickness. The corresponding shape factors are defined as

$$H = \frac{\delta^*}{\theta}, \tag{5.19}$$

$$H^k = \frac{\delta^k}{\theta}. \tag{5.20}$$

H and H^k are called shape factors. The integral boundary layer equations are written in terms of the integral thicknesses and shape factors.

Momentum integral equation The momentum integral equation is obtained by taking the zeroth moment of the boundary layer equations where the $n = 0$ in equation 5.15.

$$\frac{du_e\theta}{dx} = \frac{C_f}{2}u_e - (\delta^* + \theta)\frac{du_e}{dx}. \tag{5.21}$$

Here u_e is the velocity at the edge of the boundary layer, δ^* and θ are the displacement and momentum thickness respectively defined in equations 5.16 and 5.17. C_f is the skin friction coefficient and is defined as

$$C_f = \frac{\tau_w}{\frac{1}{2}\rho u_e^2}, \tag{5.22}$$

where τ_w is the wall shear stress. Using the shape factor defined in equation 5.19, the momentum integral equation can also be written as

$$\frac{d\theta}{dx} + (2 + H)\frac{\theta}{u_e}\frac{du_e}{dx} = \frac{C_f}{2}. \tag{5.23}$$

Kinetic energy integral equation The kinetic energy integral equation is obtained by taking the first moment of the boundary layer equations where $n = 1$ in equation 5.15. Using the energy thickness definition in equation 5.18 the following equation is obtained,

$$\frac{du_e^3\delta^k}{dx} = 2C_D u_e^3, \tag{5.24}$$

where δ^k is the kinetic energy thickness defined in equation 5.18. The dissipation coefficient, C_D, is defined as

$$D = \int_0^\delta 2\tau \frac{\partial u}{\partial y} dy,$$

$$C_D = \frac{D}{1/2\rho u_e^3}.$$

Using the shape factor defined in equation 5.20 and equation 5.23, the kinetic energy integral equation can also be written as

$$\theta \frac{dH^k}{dx} + H^k(1-H)\frac{\theta}{u_e}\frac{du_e}{dx} = 2C_D - H^k \frac{C_f}{2}. \tag{5.25}$$

5.5.1. Closure relations

The new variables introduced in equations 5.23 and 5.25 are θ, δ^*, δ^k, C_f and C_D. Using the shape factors H and H^k will only replace the variables δ^* and δ^k. The edge velocity u_e can be found using an inviscid flow analysis and is considered to be a known quantity in this system of equations. Equations 5.23 and 5.25 make no assumptions about whether the flow is laminar or turbulent and are valid in both cases. However despite starting from a closed system of equations, there are more unknowns than the number of equations and this system of equations is not closed. To close the system of equations, two dependent variables can be chosen from the equations and closure relations must be defined for the other variables in terms of the two chosen dependent variables. Typically, the momentum and displacement thicknesses, θ and δ^*, or θ and the shape factor H are chosen. For ease of analysis, closure relations are defined based on the non-dimensional shape factor H and the Reynolds number based on momentum thickness, Re_θ, as

$$Re_\theta = \frac{\theta u_e}{\nu}.$$

As noted earlier, despite starting from a closed system of equations the integral boundary layer equations are not closed. This situation is a result of the integration of the equations. While integrating the Navier-Stokes equations along the direction normal to the wall, the details of the flow in the normal direction is lost. Thus, in order to close the integral boundary layer equations, a velocity profile family of one or more parameters that represent the flows under consideration must be used to account for the simplification introduced during the integration procedure.

Laminar flow closure relations

Previously in section 4.2.1, a semi analytical velocity profile based on the solution to the Blasisus equations was shown. However, the solution to Blasisus equation is valid only for the flow over a flat plate and a more general profile is required. The Falkner-Skan solution to the laminar boundary layer equations can be used to find the laminar closure relations. Consider the flow over a wedge with an angle of $\pi\beta/2$ with an incoming velocity of U_0 as shown in figure 5.16. The edge velocity along
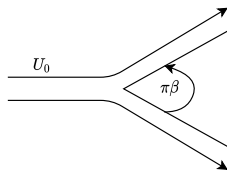
Figure 5.16: A schematic of the flow past a wedge.

the solid surface is assumed to follow the relation

$$u_e(x) = U_0 \left(\frac{x}{L}\right)^m, \tag{5.26}$$

where L is a characteristic length scale and m is a dimensionless constant which is determined by the wedge angle as

$$\beta = \frac{2m}{m+1}. \tag{5.27}$$

$\beta = 0$ corresponds to the Blasisus solution for flow over a flat plate. Drela [22] solved the Falkner-Skan equation using a prescribed shape parameter to obtain the laminar closure relations as follows

$$H^k = \begin{cases} 1.515 + 0.076\frac{(4-H)^2}{H}, & H < 4, \\ 1.515 + 0.040\frac{(H-4)^2}{H}, & H > 4. \end{cases} \tag{5.28}$$

$$Re_\theta \frac{C_f}{2} = \begin{cases} -0.067 + 0.01977\frac{(7.4-H)^2}{H-1}, & H < 7.4, \\ -0.067 + 0.022 \left(1 - \frac{1.4}{H-6}\right)^2, & H > 7.4. \end{cases} \tag{5.29}$$

$$Re_\theta \frac{2C_D}{H^k} = \begin{cases} 0.207 + 0.00205(4-H)^{5.5}, & H < 4, \\ 0.207 - 0.003\frac{(H-4)^2}{(1+0.02(H-4)^2)}, & H > 4. \end{cases} \tag{5.30}$$

Turbulent flow closure relations

Unlike laminar flows, no analytical or single parameter self similar solutions can be found for the velocity profile in a turbulent boundary layer. As seen in section 4.2.2, the turbulent boundary layer consists of a two layer structure. In the inner layer both turbulent and molecular viscous stresses are relevant whereas in the the outer layer, only the turbulent stress are important. Thus, the inner layer thickness scales differently compared to the outer layer.

As seen in section 4.2.2, the velocity profile in the inner layer of the boundary layer is defined based on the friction velocity, u_τ, which in turn depends on the wall shear stress. Thus, the wall shear stress is the primary scaling factor in the inner layer of a turbulent boundary layer [22] and will be used to define other quantities. The most commonly used formula to compute the skin friction for turbulent

boundary layers is the formula derived by Swafford [23]

$$C_f = \frac{0.3 \exp(-1.33H)}{(log_{10}Re_\theta)^{1.74+0.31H}} + 0.00011 \left(tanh \left(4.0 - \frac{H}{0.875} \right) - 1.0 \right). \tag{5.31}$$

Drela [22] derived the closure relation for the kinetic energy shape factor H^k using the combination of the velocity profile in the inner and outer layers of the turbulent boundary layer. The velocity profile used was first derived by Swafford [23] as

$$\frac{u}{u_e} = \frac{u_\tau}{u_e} \frac{s}{0.09} tan^{-1}(0.09y^+) + \left(1 - \frac{u_\tau}{u_e} \frac{s\pi}{0.18} \right) tanh^{1/2} \left(a \left(\frac{y}{\theta} \right)^b \right), \tag{5.32}$$

where

$$\frac{u_\tau}{u_e} = \left| \frac{C_f}{2} \right|^{\frac{1}{2}}, \quad s = \frac{C_f}{|C_f|}, \quad y^+ = \frac{u_\tau y}{\nu}.$$

a and b are two constants which are calculated for a given δ^* and θ by using the skin friction formula in equation 5.31 and substituting the velocity profile in the definitiions of δ^* and θ. The closure relation for the kinetic energy shape factor obtained by Drela [22] is

$$H^k = \begin{cases} 1.505 + \frac{4}{Re_\theta} + \left(0.165 - \frac{1.6}{\sqrt{Re_\theta}} \right) \frac{(H_0-H)^{1.6}}{H}, & H < H_0 \\ \\ 1.505 + \frac{4}{Re_\theta} + (H_0 - H)^2 \left(\frac{0.04}{H} + 0.007 \frac{ln(Re_\theta)}{(H-H_0+\frac{4}{ln(Re_\theta)})^2} \right), & H > H_0. \end{cases} \tag{5.33}$$

where

$$H_0 = 3.0 + \frac{400}{Re_\theta}.$$

Two distinct approaches can be used to find the closure relations for the dissipation coefficient. The first approach assumes that there are two distinct contributions to the integral dissipation coefficient - one from the wall layer and one from the outer layer (also known as the wake layer). One such closure relation from Le Balleur (taken from Drela [22]) is

$$C_D = \frac{C_f}{2} \frac{u_s}{u_e} + \frac{K\pi^2}{16} \left(1 - \frac{u_s}{u_e} \right)^3. \tag{5.34}$$

The second approach [22] is based on the concept of an equilibrium boundary layer. A new pressure gradient parameter analogous to the Falkner-Skan pressure gradient parameter and a modified shape parameter are introduced for turbulent flows. The turbulent boundary layers can be shown to be self similar when using the new parameters and closure relations can be derived similar to the laminar flow closures. The new pressure gradient parameter, β is given by

$$\beta = -\frac{2}{C_f} \frac{\delta^*}{u_e} \frac{du_e}{dx}. \tag{5.35}$$

Experimental evidence shows that if the new pressure gradient parameter is constant, the modified shape factor, G,

$$G = \frac{H-1}{H}\sqrt{\frac{2}{C_f}} \tag{5.36}$$

is also constant. The empirical relation between G and β is [22]

$$G = 6.7\sqrt{1 + 0.75\beta}. \tag{5.37}$$

Using the expressions for G, β and assuming equilibrium conditions in the boundary layer (i.e. constant shape factor), the closure relation for C_D is given by

$$\frac{2C_D}{H^k} = \frac{C_f}{2}\left(\frac{4}{H} - 1\right)\frac{1}{3} + 0.03\left(\frac{H-1}{H}\right)^3. \tag{5.38}$$

Both the closure relations presented so far for C_D are based on equilibrium flow conditions. As discussed earlier in section 5.2.2, in a turbulent flow additional stresses appear as a result of the Reynolds averaging of the turbulent fluctuations and is given by,

$$\tau = \mu\frac{\partial u}{\partial y} - \rho\overline{u'v'}. \tag{5.39}$$

Under the equilibrium boundary layer assumption, the closure relations for the dissipation coefficient, C_D, already involve a velocity gradient weighted integral of the Reynolds stress [22]. However such equilibrium profiles assume only a local dependence of boundary layer parameters on Reynolds stresses. This assumption does not hold in many situations like flows under an adverse pressure gradient that is increasing downstream or in flows were an adverse pressure gradient is suddenly removed, both of which can occur around airfoil trailing edges.

To account for upstream history effects another equation known as the shear lag equation [12, 13] is added to the system of equations. First, a new non dimensional quantity, C_τ, the maximum Reynolds shear stress coefficient is introduced as,

$$C_\tau = \frac{1}{u_e^2}(-\overline{u'v'})_{max}. \tag{5.40}$$

Then it is assumed that the maximum Reynolds shear stress point is representative of the Reynolds shear stress level of the entire boundary layer. The local shear stress is obtained by solving a shear-stress transport equation. A parameter, the equilibrium shear stress coefficient, $C_{\tau_{EQ}}$, is defined as the value of the shear stress coefficient which would occur if the local boundary layer was part of an equilibrium flow [12, 22]. The shear lag equation is defined as

$$\frac{\delta}{C_\tau}\frac{dC_\tau}{dx} = 2a_1\frac{u_e}{u}\frac{\delta}{L}(C_{\tau_{EQ}}^{1/2} - C_\tau^{1/2}). \tag{5.41}$$

The values of the constants commonly used are [12, 13, 22]

$$a_1 = 0.15, \quad \frac{u_e}{u} = 1.5, \quad \frac{L}{\delta} = 0.08.$$

Here L represents the conventional mixing length used in turbulence modeling.

5.5.2. Effect of vortex generators

Boundary layer thickness

The effect of VG on the integral boundary layer parameters is examined by extracting the displacement and momentum thicknesses and shape factors from the CFD simulations. The edge of the boundary layer is located based on vorticity magnitude and then the displacement, momentum thickness and other boundary layer parameters are calculated by integrating the velocity profiles numerically. The extracted thicknesses are shown in figures 5.17a and 5.17b. There is a significant increase in the displacement and a drop in momentum thickness at the $x_{VG} = 0.985m$. Once again, the planes $y = \pm TE$ and $y = \pm D/3$ are symmetric. As the flow moves downstream all the planes tend towards a similar value. Figures 5.18a and 5.18b
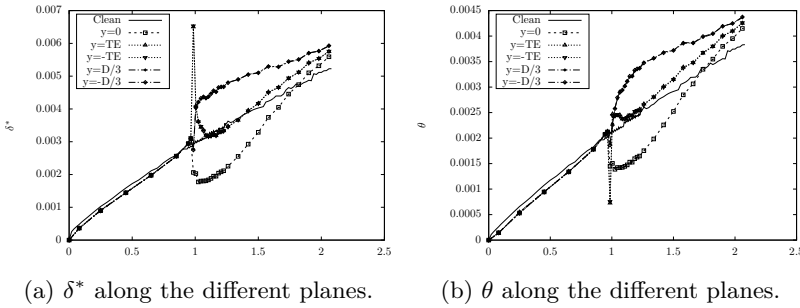

(a) δ^* along the different planes.   (b) θ along the different planes.

Figure 5.17: Displacement and momentum thickness along the flat plate computed from the $3D$ CFD simulations.

show the shape factors H and H^k computed from the CFD simulations. The shape factor along the planes $y = \pm D/3$ is very close to the clean simulation whereas the other planes deviate significantly in the vicinity of the VG and then match the clean simulation as the flow moves downstream.
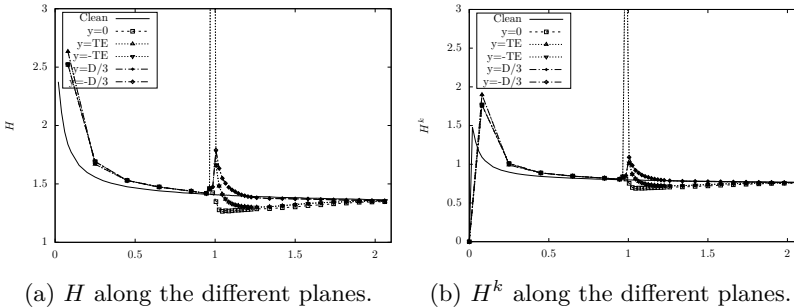

(a) H along the different planes.   (b) H^k along the different planes.

Figure 5.18: Shape factors H and H^k along the flat plate computed from the $3D$ CFD simulations.

Figures 5.19a and 5.19b show the integral boundary layer quantities computed from the $2D$ CFD simulation. While a sharp increase in the displacement thickness similar to the $3D$ simulation is observed, the behavior of the momentum thickness

is different. The shape factors H and H^k behave similarly in both the $2D$ and $3D$ simulations. The values of H and H^k from both the $2D$ and $3D$ simulations tend towards approximately 1.34 and 0.78 away from the VG.

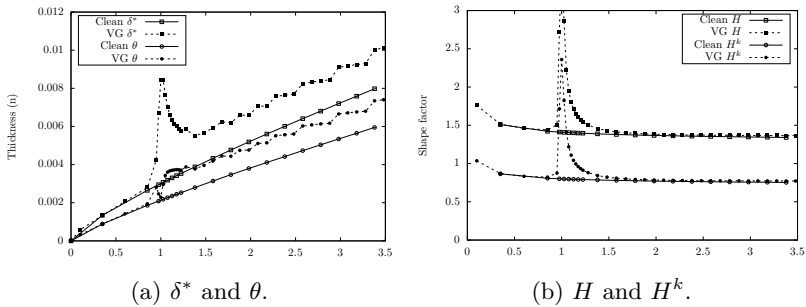
(a) δ^* and θ.

(b) H and H^k.

Figure 5.19: Integral boundary layer parameters along the flat plate computed from the $2D$ CFD simulations.

From these results, it appears that the $2D$ simulation matches the behavior of the integral boundary layer properties of the $3D$ simulation except in the immediate vicinity of the VG.

Closure relations

Figure 5.20 shows the skin friction coefficient from the numerical simulations with and without VGs. For the VG simulations, the skin friction coefficient is shown in five different planes - $y = 0$, $y = \pm TE$ and $y = \pm D/3$. Upstream of the VG location, the skin friction coefficient behaves identically with and without VGs in all the planes considered. There is a sharp spike at the location of the VG and subsequently, the C_f is significantly different to the clean simulations. Symmetry is observed in the C_f results around $y = 0$, similar to the velocity profiles. The
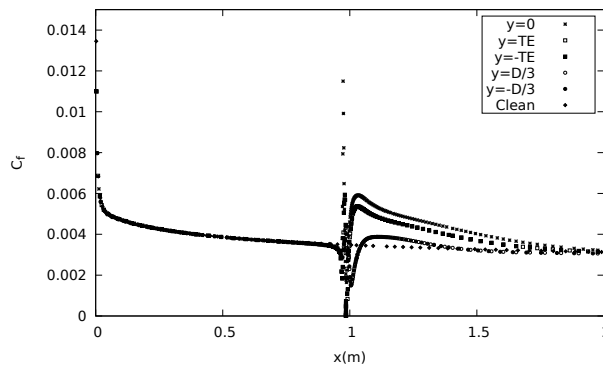

Figure 5.20: Comparison of skin friction coefficient (C_f) with and without a VG.

deviation from the clean results are the largest in the $y = 0$ plane and lowest in the

$y = \pm D/3$ planes. As the flow moves downstream, the C_f from the VG simulations start to match the clean simulation results. The C_f along the planes $y = \pm D/3$ match the clean results earlier than the other planes. From these results, there appears to be a region immediately downstream of the VG where the behavior of the turbulent boundary layer even in the near wall region is altered and this behavior recovers back to the standard turbulent boundary layer further downstream.

While the near wall region or the inner boundary layer recovers to a profile similar to the clean boundary layer, the behavior in the outer layer is yet to be examined. Figure 5.21 shows the eddy viscosity computed from the CFD simulations at different streamwise locations in the $y = 0$, $y = \pm TE$ and $y = \pm D/3$ planes. As expected no difference between the clean and VG eddy viscosity in the outer layer is observed at $s_{VG} = -3$. As the flow moves downstream, the outer layer is significantly different in the VG simulation. At $s_{VG} = 10$ and $s_{VG} = 50$ (figures 5.21c and 5.21c), the eddy viscosity profiles are symmetric around $y = 0$. Further downstream, the eddy viscosity remains different at $s_{VG} = 199$ as seen in figure 5.21d. However, at this location no difference between the planes is observed.
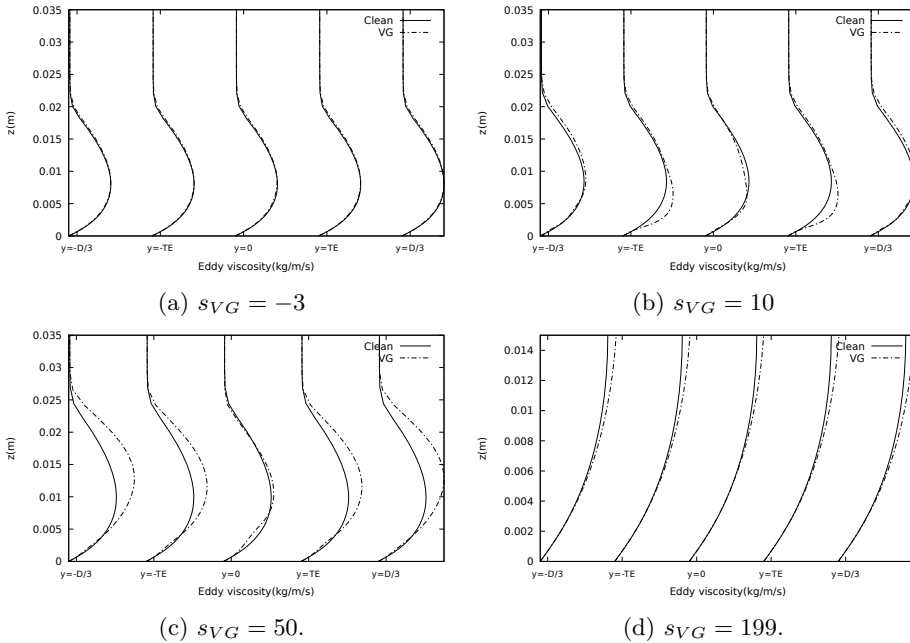

(a) $s_{VG} = -3$

(b) $s_{VG} = 10$

(c) $s_{VG} = 50$.

(d) $s_{VG} = 199$.

Figure 5.21: Boundary layer velocity profile from the $3D$ simulations at different spanwise locations.

The presence of the VG causes a deviation in not only the outer layer of the turbulent boundary layer, but also in the near wall region or the inner layer. However, the inner layer appears to recover back to the standard turbulent boundary layer, but the outer layer remains different. As seen in figure 5.20, the skin friction coefficient, which is used as a scaling parameter for the inner layer, is different and

equation 5.31 is no longer valid. Similarly, since the eddy viscosity profiles are now different, the Reynolds stresses in the boundary layer will behave differently and the shear lag equation needs to be re-examined.

In order to derive these closure relations the mixing layer scaling shown in section 5.4 can be used. However, in order to derive such relations, more data from numerical simulations and experiments of VGs at different Reynolds numbers and configurations are required.

5.6. Conclusions and future work

CFD simulations of the flow over a vortex generator(VG) on a flat plate were performed using the new pressure based solver. The nature of the turbulent boundary layer in the presence of a VG was examined and compared to a clean boundary layer. It was observed that while the boundary layer is three dimensional, the spanwise variation of the velocity in the boundary layer is determined by the orientation of the VG. Spanwise symmetry was observed not only in the vicinity of the VG but also far downstream. This symmetry can be taken advantage of to model the effect of VGs in simpler two dimensional methods. To identify the difference between a two dimensional approximation and the fully resolved VG simulation in $3D$, a $2D$ CFD simulation of one of the planes of the $3D$ domain was carried out. The $2D$ boundary layer with the VG (represented by a zero thickness line) behaves differently in the vicinity of the VG and in the near wall region. However, away from the wall, both the $2D$ and $3D$ boundary layers show similar qualitative behavior downstream of the VG. To better understand the qualitative behavior of the two boundary layers,the velocities in the boundary layers were scaled using the plane mixing layer relations. Since the vortex generators mixes high energy fluid from outside the boundary layer with the slower moving boundary layer the velocities are likely to resemble a mixing layer. The scaled velocities within this embedded mixing layer behave differently than a standard plane mixing layer especially in near the wall. Away from the wall, the scaled velocities show very good self symmetry. This self symmetry was observed along the different planes in the $3D$ simulation and in the $2D$ simulation. The velocities in the mid plane between the two VGs resembled the plane mixing layer more closely than in other planes.

This velocity scaling can be taken advantage of to model the effect of VGs using simpler two dimensional methods based on the integral boundary layer (IBL) equations. To solve the IBL equations closure relations need to be derived. These closure relations are currently based on velocity profiles observed in standard turbulent boundary layers. To derive new closure relations based on the mixing layer scaling, more numerical simulations and experiments need to be carried out at different Reynolds numbers and VG configurations.

References

[1] A. K. Ravishankara, H. Özdemir, and A. Franco, *Towards a vortex generator model for integral boundary layer methods,* in *AIAA Scitech 2019 Forum.*

[2] J. C. Lin, *Review of research on low-profile vortex generators to control boundary-layer separation,* Progress in Aerospace Sciences **38**, 389 (2002).

[3] G. B. Schubauer and W. G. Spangenberg, *Forced mixing in boundary layers,* Journal of Fluid Mechanics **8** (1960).

[4] A. D. Cutler and P. Bradshaw, *Strong vortex/boundary layer interactions Part II,* Experiments in Fluids (1993).

[5] A. D. Cutler and P. Bradshaw, *Strong vortex/boundary layer interactions Part I,* Experiments in Fluids (1993).

[6] P. Fuglsang, I. Antoniou, K. Dahl, and H. Aagaard Madsen, *Wind tunnel tests of the FFA-W3-241, FFA-W3-301 and NACA 63-430 airfoils* (1998).

[7] L. Florentie, A. H. van Zuijlen, S. J. Hulshoff, and H. Bijl, *Effectiveness of Side Force Models for Flow Simulations Downstream of Vortex Generators,* AIAA Journal **55**, 1373 (2017).

[8] U. Fernandez, C. M. Velte, P.-E. Rethore, and N. N. Sørensen, *Self similarity and helical symmetry in vortex generator flow simulations,* (The science of making torque from wind, 2012).

[9] E. E. Bender, B. H. Anderson, and P. J. Yagle, *Vortex generator modeling for navier-stokes codes,* ASME Paper FEDSM99-6919 (1999).

[10] A. Jirasek, *Vortex-generator model and its application to flow control,* Journal of Aircraft **42**, 1486 (2005).

[11] U. Fernandez, P.-E. Réthoré, N. Sørensen, C. Velte, F. Zahle, and E. Egusquiza, *Comparison of four different models of vortex generators,* in *Proceedings of EWEA 2012 - European Wind Energy Conference & Exhibition* (European Wind Energy Association (EWEA), 2012) eWEA 2012 - European Wind Energy Conference & Exhibition, EWEC 2012 ; Conference date: 16-04-2012 Through 19-04-2012.

[12] M. Drela, *Xfoil: An analysis and design system for low reynolds number airfoils,* in *Low Reynolds number aerodynamics* (Springer, 1989) pp. 1–12.

[13] G. Ramanujam, H. Özdemir, and H. Hoeijmakers, *Improving airfoil drag prediction,* AIAA SciTech 2016 (AIAA, 2016).

[14] H. Özdemir, A. van Garrel, A. Koodly Ravishankara, F. Passalacqua, and H. Seubers, *Unsteady interacting boundary layer method,* in *35th Wind Energy Symposium* (2017) p. 2003.

[15] D. Baldacchino, *Experimental investigation of low profile vortex generators in a boundary layer wind tunnel,* in *AVATAR, Task 3.1: Models for flow devices and flow control* (2015).

[16] K. A. Waithef, *Source term model for vortex generator vanes in a navier-stokes computer code,* (2004).

[17] S. B. Pope, *Turbulent Flows* (Cambridge University Press, 2000).

[18] M. R. Rebello, *Analytical and Experimental Investigation of a turbulent mixing layer of different gases in a pressure gradient*, Ph.D. thesis, California Institute of Technology (1973).

[19] O. Konig, J. Schuluter, and H. E. Fiedler, *Turbulent Mixing Layer in Adverse Pressure Gradient,* in *Progress in Fluid Flow Research: Turbulence and Applied MHD* (AIAA).

[20] C. M. Sabin, *An analytical and experimental study of the plane, incompressible, turbulent free-shear layer with arbitrary velocity ratio and pressure gradient,* Journal of Basic Engineering **87**, 421 (1965).

[21] M. Matsushita, S. Murata, and T. Akamatsu, *Studies on boundary-layer separation in unsteady flows using an itegral method,* Journal of Fluid Mechanics **149**, 477 (1984).

[22] M. Drela, *Two-dimensional transonic aerodynamic design and analysis using the Euler equations*, Ph.D. thesis, Massachusetts Institute of Technology (1986).

[23] T. Swafford, *Calculation of Skin Friction in Two-Dimensional, Transonic, Turbulent Flow*, Tech. Rep. (ARO INC ARNOLD AFS TN, 1979).

6

Wind energy applications: Roughness modeling

The surface of wind turbine blades are prone to degradation due to exposure to the elements. Rain, hail, insects are among the many causes of turbine blade degradation or erosion. Surface degradation of the wind turbine blades leads to a reduction in the aerodynamic performance, resulting in power losses. The effect of surface degradation is studied by modeling the turbine blade as a rough surface. Surface roughness can be positive (insects or other foreign objects) or negative (erosion, delamination). The individual roughness elements are however very small and it is not always feasible to study the actual degraded surface. Thus various roughness models have been proposed in the literature which eliminate the need to accurately model the degraded surface by representing erosion with a virtual surface and modeling the effect of erosion on the flow quantities near the eroded surface. In this study, the reduction in performance of airfoils due to leading edge roughness is quantified. Different roughness models are investigated and evaluated against theoretical models. Additionally, the effect of roughness on different integral boundary layer quantities like displacement thickness, momentum thickness and skin friction are presented.

6.1. Introduction

Leading edge erosion is an issue of growing concern in the wind turbine industry in recent years. The combination of growth in the size of wind turbines, increased offshore installations, especially in locations with more adverse weather conditions, has made this subject crucial to the industry [2]. Erosion of turbine blades are largely caused by rain, hailstones, accumulation of contaminants and tends to change the shape of the airfoils. This leads to a reduction in aerodynamic performance of the affected sections. Han[3] presented the effects of contamination of the airfoil

---

used at blade tips on a 5 MW NREL turbine blade using CFD simulations. They report a worst case scenario where the Annual Energy Production (AEP) drops by 3.7%. Herring[2] presents a thorough review on the growing importance of leading edge erosion and different coating alternatives to reduce the impact of erosion. A wide range of drop in AEP, from about 25% to about 3.7%, is reported and the authors suggest it is due to different operating conditions and roughness levels used to evaluate the impact of erosion. The authors also note that repair of moderate erosion can recover the AEP by about 2%.

In order to quantify the adverse effects of roughness the flow around the turbine blades should be investigated. Laminar flow tends to transition to turbulent flow prematurely in presence of roughness. A review of experimental approaches to model roughness and its effect on transition can be found in Ehrmann et al [4]. Langel et al. [5] performed experiments on two airfoils by adding cut vinyl decals and focused on $100 < Re_k < 400$, where Re_k is the Reynolds number based on roughness height k. They also present a numerical approach to model the effect of roughness on transition by adding a scalar field variable. The new scalar variable is used to modify the $\gamma - Re_\theta$ transition model [6]. Sareen et al. [7] note that most of the experimental studies on roughness use strips or zigzag tapes to simulate real roughness and not many studies exist on negative roughness like erosion where material is lost from the blade.

Apart from causing early transition, the nature of the turbulent boundary layer also changes due to roughness. Skin friction increases and a shift in the velocity profile in the inner part of the boundary layer is observed. The additional dissipation near the roughness elements leads to thickening of the boundary layer which can make the boundary layer prone to early separation.

The concept of equivalent sand grain roughness is widely used in turbulence models to account for the effect of roughness on turbulent boundary layers. Nikuradse[8] performed experiments to measure pressure losses across pipes due to roughness, which forms the basis of the sand grain roughness concept. Nikuradse provided relations for the loss in pressure head (friction) and the velocity shift as a function of sand grain roughness heights. Real roughness is first converted to equivalent sand grain roughness when using the roughness models for RANS turbulence models. Typically the rough surface is replaced by a smooth surface and the effect of roughness is modeled as extra dissipation in the inner boundary layer.

Integral boundary layer based tools like RFOIL[9] are used extensively in the wind energy community for quick and accurate analysis of airfoil performance, especially in combination with other methods like Blade Element Momentum theory, to obtain the power output of wind turbines in a relatively inexpensive manner. However, it is restricted mainly to clean airfoils due to lack of research on developing roughness models for integral boundary layer methods. Olsen et al[10] recently proposed a new closure relation for skin friction in the presence of roughness. The authors also note that further work is necessary to refine their study.

In this study, roughness models for the SA and SST $k - \omega$ turbulence models are implemented in the open source tool SU2[11]. The grid requirements and the accuracy of the two models are examined and validated against experimental data.

Two airfoils are considered - NACA 65_2215 and a popular wind turbine airfoil DU-96-W-180. The NACA 65_2215 airfoil has been used for validating roughness models earlier[12][13]. Sareen et al. [7] performed experiments on the DU-W-96-180 with 'negative' roughness. Thus different ways to obtain equivalent sand grain roughness for 'negative' roughness are also examined in this paper. The numerical solution of the RANS equations is then used to analyze the behaviour of the turbulent boundary layer and the various integral boundary layer quantities in the presence of roughness as well as to analyze the integral boundary layer parameters in order to improve roughness modeling in integral boundary layer methods.

The organization of the paper is as follows: the two different roughness models for RANS are presented in section 6.2, validation cases for the roughness models are presented in section 6.3. Based on the results in section 6.3, the SA roughness model is validated against experiments on airfoils in section 6.4. In section 6.5, the effect of roughness on various integral boundary layer properties is analysed. The conclusions are presented in section 6.6.

6.2. Roughness modeling

To motivate the roughness model used in this study, a brief introduction of turbulent boundary layers and the impact of roughness is presented below.

The turbulent boundary layer can be broadly divided into two regions [14, 15]; the inner region where viscous dissipation is comparable to the turbulent dissipation and the outer region where turbulence dissipation dominates completely. The inner region can be further subdivided into three regions - the viscous sub-layer where viscous effects dominate and turbulent effects are absent, a buffer region where the turbulent stresses start to grow and finally an overlap region or a logarithmic region where the turbulent and viscous dissipation match. The overlap region leads into the outer layer of the boundary layer where viscous effects are minimal. The velocity profile in the viscous sub-layer and overlap region can be written respectively as

$$u^+ = y^+, \qquad y^+ \leq 5, \tag{6.1}$$

$$u^+ = \frac{1}{\kappa} ln(y^+) + C, \qquad y^+ > 30. \tag{6.2}$$

The region of the boundary layer between $5 \leq y^+ \leq 30$ is the buffer region. In the above relations, y^+ is the non dimensional wall normal coordinate and u^+ is the normalized velocity defined as

$$y^+ = \frac{y u_\tau}{\nu}, \qquad u^+ = \frac{u}{u_\tau}, \qquad u_\tau = \sqrt{\frac{\tau_w}{\rho}}.$$

Here u_τ is known as the wall friction velocity and is used as the velocity scale close to the wall, τ_w is the wall shear stress, ρ is the density, u is the local velocity and ν is the kinematic viscosity. The constant in equation 6.2 for a smooth wall is known to be $C = 5.0$.

The presence of surface roughness on the wall alters the nature of the velocity distribution near the wall. The roughness elements will introduce new turbulent fluctuations in the flow increasing the skin friction. Typically, a standardized notion of roughness known as the "equivalent sand grain roughness height (k_s)" is used to denote roughness of a wall [8, 15, 16]. A given physical roughness distribution is converted into the "equivalent sand grain roughness height" using empirical correlations [17–19]. A more detailed review is presented in section 6.4.3. Based on the non dimensional roughness height,

$$k_s^+ = k_s u_\tau / \nu, \tag{6.3}$$

three regimes of roughness can be identified [15]. If the roughness elements are within the viscous sub-layer ($k_s^+ \leq 5$, hydraulically smooth), the effect of roughness is not relevant and there is no difference with the smooth velocity profile. As the height of the roughness element increases ($5 \leq k_s^+ \leq 70$, transitionally rough), a shift in the velocity profile is observed. Once the roughness elements are fully within the overlap region ($k_s^+ > 70$, fully rough), the viscous sub-layer plays no part and the flow is in the fully rough regime. It must be noted here that the equivalent sand grain roughness concept typically applies only to the commonly observed distributed roughness ($k-$ type roughness [20]) and not to isolated roughness elements. To reproduce the proper shift Δu^+ in the boundary layer velocity profiles, turbulence models typically increase the eddy viscosity dissipation within the inner part of the boundary layer [16]. Aupoix et al. [16], identify two methods to accomplish this with eddy viscosity based turbulence models (e.g SA and SST):

1. Finite eddy viscosity at the wall which can be interpreted as using a virtual wall to represent roughness and

2. Zero eddy viscosity at the wall where the origin of the wall is at the bottom of roughness but turbulence damping in the wall region is reduced.

With this background on roughness modeling in turbulent boundary layers, roughness models for the SA and SST turbulence models are presented.

6.2.1. Roughness modification for SA model
The roughness modification proposed by Boeing [16, 21] is considered in this section. An alternate modification was also proposed by ONERA in Aupoix et al. [16], but is not considered since it requires the additional input of the friction velocity. The effect of roughness is accounted for by shifting the virtual wall to the top of the roughness element. This can be achieved by offsetting the distance to the wall everywhere. The changes to the turbulence model are

$$d_{new} = d_{min} + 0.03k_s, \tag{6.4}$$

$$\chi = \frac{\tilde{\nu}}{\nu} + c_{R1}\frac{k_s}{d_{new}}, \tag{6.5}$$

$$f_{v2} = 1 - \frac{\tilde{\nu}}{\nu + \tilde{\nu}f_{v1}}. \tag{6.6}$$

with $c_{R1} = 0.5$. The eddy viscosity at the wall is now changed from $\tilde{\nu} = 0$ to a non-zero value by using a mixed (Robin) boundary condition at the wall,

$$\frac{\partial \tilde{\nu}}{\partial n}\bigg|_{wall} = \frac{\tilde{\nu}_{wall}}{0.03 k_s}, \tag{6.7}$$

where $\frac{\partial \tilde{\nu}}{\partial n}$ is the gradient of $\tilde{\nu}$ in the direction normal to the wall.

6.2.2. Roughness modification for SST model

The effect of roughness can be accounted for in the $k - \omega$ SST turbulence model by modifying the boundary conditions at the wall as [22]

$$k_{rough} = 0, \tag{6.8}$$

$$\omega_{rough} = \frac{(\mu_\tau)^2 S_R}{\nu}, \tag{6.9}$$

where

$$S_R = \begin{cases} (\frac{50}{k_s^+})^2, & k_s^+ \leq 25, \\ (\frac{100}{k_s^+}), & k_s^+ > 25. \end{cases}$$

From equation 3.83, the eddy viscosity remains zero at the wall, but there is an increase in turbulence dissipation compared to the clean boundary conditions. Here k_{rough} is the turbulent kinetic energy and k_s^+ is the non dimensional equivalent sand grain roughness height defined in equation 6.3.

The two roughness models are implemented in SU2 and are validated below.

6.3. Model validation

6.3.1. Turbulent flow over a 2-D flat plate

Grid refinement study

Turbulent flow over a flat plate with different roughness heights is simulated with the SA and the SST turbulence models and their respective roughness corrections presented above. The flat plate domain is $2m$ long and $1m$ high and $Re = 6.0 \times 10^6$. A grid refinement study is carried out for the geometry under clean and three roughness levels. There are 57, 113 and 225 points on the surface of the 2-D flat plate for the three grids. The minimum grid spacing is $\Delta y_1 \approx 2 \times 10^{-6} m$. A second set of grids are made with same geometry and same number of points on the surface but with a minimum grid spacing of $\Delta y_2 \approx 3 \times 10^{-8} m$ for the SST roughness model. A growth ratio of 1.09 is used in the normal direction. The skin friction values computed at $x = 0.93m$ are tabulated in table 6.1. Three different roughness heights, $k_s = 1.23 \times 10^{-4} m$, $k_s = 2.46 \times 10^{-4} m$ and $k_s = 9.48 \times 10^{-4} m$ are tested. The k_s^+ values are around 25, 50 and 200 respectively. A grid spacing of Δy_1 gives $y^+ \approx 0.3$ at $x = 0.93$ under clean conditions. As seen in table 6.1, a grid independent solution is obtained for the clean case for both turbulence models with this minimum grid spacing. The SA roughness model gives largely grid independent result for all the roughness heights. However, a grid independent solution is not

$k_s(m)$	N	SA	SST(Δy_1)	SST(Δy_2)
Clean	57	0.00273	0.00267	0.00272
	113	0.00274	0.00271	0.00274
	225	0.00274	0.00273	0.00274
1.23×10^{-4}	57	0.00369	0.00335	0.00346
	113	0.00382	0.00341	0.00346
	225	0.00382	0.00344	0.00346
2.46×10^{-4}	57	0.00451	0.00348	0.00374
	113	0.00457	0.00361	0.00374
	225	0.00457	0.00368	0.00374
9.84×10^{-4}	57	0.00605	0.00375	0.00424
	113	0.00599	0.00392	0.00425
	225	0.00593	0.00413	0.00425

Table 6.1: Skin friction (C_f) at $x = 0.93m$ for different grid resolutions and roughness levels with SA and SST. N denotes the number of points on the surface of the flat plate.

possible under rough conditions with the SST model. The variation is marginal at low roughness heights and increases as the roughness height increases. With the grid spacing of Δy_2, grid independent solutions at different roughness heights are obtained with the SST model as well. The y^+ under clean conditions for this grid spacing is 0.006.

The first two roughness heights are in the transitional roughness regime while the third roughness height is in the fully rough regime. The SA roughness modification gives a grid independent solution with a minimum $y^+ \approx 0.3$ whereas the SST roughness model fails to do so in the fully rough regime. This is likely due to how the roughness modification is introduced in the two models. The eddy viscosity at the wall is directly modified in SA but in SST it remains zero. In the fully rough regime, there is a non-zero eddy viscosity in the inner region of the boundary layer where the viscous sub layer previously existed. Since the eddy viscosity is still zero at the wall for the SST roughness modification, to capture the steep increase in eddy viscosity a finer mesh is likely required compared to the SA roughness modification.

Velocity profiles
Despite the finer mesh the skin friction values predicted by the SST roughness model do not match those from the SA model especially under fully rough conditions (table 6.1). The velocity profiles in the inner boundary layer are now investigated to determine the accuracy of the two models. The velocity profiles for different roughness heights are presented in figures 6.1 and 6.2. The profiles are computed based on the grid independent results i.e. with a grid spacing of Δy_1 for SA and Δy_2 for SST. From figures 6.1 and 6.2, we can see that the clean case matches the viscous sub-layer and log law in the overlap region closely for both the SA and SST models. Further, increasing the equivalent roughness height has the predicted effect of a shift of the velocity profile away from the clean case and once $k_s^+ > 70$, the viscous sub-layer disappears. To further verify the two results, a comparison

is made with the empirical shift in velocity profile as proposed by Nikuradse [23] shown below.

$$u^+ = \frac{1}{\kappa} log(\frac{y^+}{k_s^+}) + B,$$ (6.10)

where $\kappa = 0.40$ and the shift B is given by

$$1 < k_s^+ < 3.5, \quad B = 5.5 + \frac{1}{\kappa} log(k_s^+),$$
$$3.5 < k_s^+ < 7, \, B = 6.59 + 1.52 log(k_s^+),$$
$$7 < k_s^+ < 14, \qquad B = 9.58,$$
$$14 < k_s^+ < 68, \, B = 11.5 - 0.7 log(k_s^+),$$
$$68 < k_s^+, \qquad B = 8.48.$$

Comparing the empirical predictions of the velocity shift (figure 6.1), a slight over-prediction is observed in the transitionally rough region by the SA roughness model. This was also reported in Knopp et al[13]. The SST roughness model does not perform as well as the SA model especially in the fully rough regime (figure 6.2) despite using a much finer grid. The limitations in the $k - \omega$ SST roughness model are also reported elsewhere [13, 23, 24]. It must be noted that various corrections for the SST roughness model have been proposed (for example [13, 23]) but are not investigated in the current study.
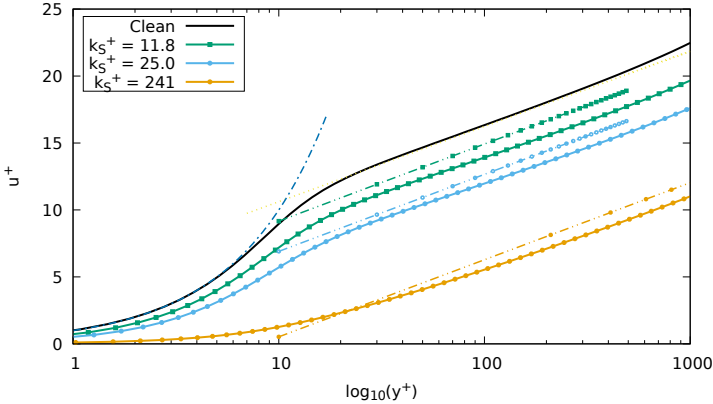
Figure 6.1: A comparison of velocity shifts obtained from the SA model to the theoretical value. Numerical results shown in solid and theoretical results shown by dashed lines.

6.3.2. Blanchard experiments

In this section, the two roughness models are compared to the experimental data from Blanchard obtained from Aupoix et al [16]. The sand grain roughness height was $4.25 \times 10^{-4} m$. With an incoming velocity of $45 ms^{-1}$, the simulation is carried out on a $2m$ long flat plate. The resulting Reynolds number is $Re = 6.46 \times 10^6$. The
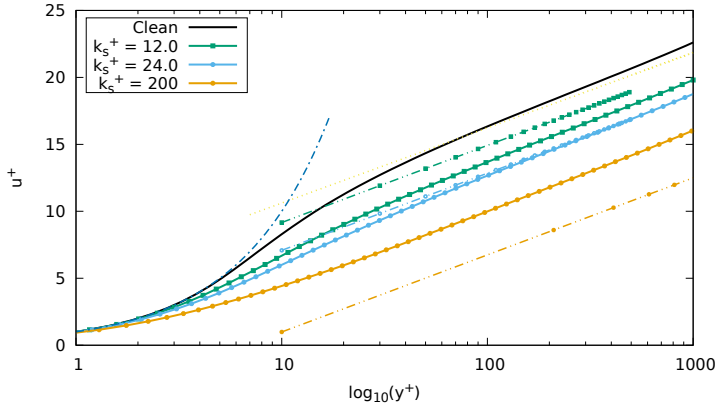
Figure 6.2: A comparison of velocity shifts obtained from the SST model to the theoretical value.Numerical results shown in solid and theoretical results shown by dashed lines.

y^+ of the mesh used is less than 0.4 throughout the domain for the SA roughness model and less than 0.007 for the SST roughness model. The comparison is shown in figure 6.3. Both the SA and SST models predict a higher skin friction compared to the clean flat plate but the results from the SA roughness model are significantly closer to the experimental data. The resulting $k_s^+ \approx 150$ makes the flow fully rough. As seen in figure 6.2, the SST roughness model performs poorly in this regime which results in an underprediction of the skin friction.
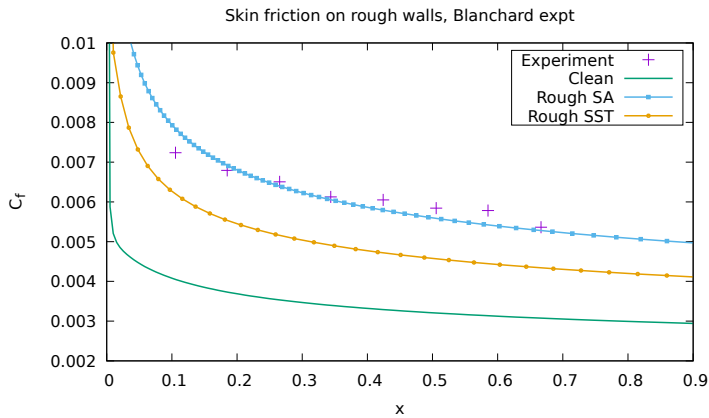
Figure 6.3: Comparison of skin friction coefficient (C_f) from SST and SA roughness models to experimental data from Blanchard[16].

6.4. Roughness on airfoil sections

As seen in section 6.3.1, a very fine grid in the wall normal direction is required for the SST roughness model compared to the SA roughness model, which gives grid independent results with meshes comparable to the clean cases. Additionally, despite the fine grid the SST roughness model performed poorly compared to the SA roughness model in predicting the skin friction for the flat plate. Therefore in the following sections only the SA roughness model will be used. A chord length (c) of $1m$ is assumed and the roughness values are normalized by the chord length.

6.4.1. NACA 65_2215

In this section the SA model is further validated against the NACA 65_2215 airfoil. The Reynolds number is $Re = 2.6 \times 10^6$ and the roughness covers the entire upper surface and on the lower surface from the leading edge up to $x/c = 0.15$. Three roughness heights $k_s/c = 1.54 \times 10^{-4}$, $k_s/c = 3.08 \times 10^{-4}$ and $k_s/c = 1.23 \times 10^{-3}$ are considered here. Clean experiments were performed by Abbot and von Doenhoff [25]. Ljungstrom performed experiments with different roughness heights on the NACA 65_2A215 airfoil, a closely related airfoil. These experiments have been used to validate roughness models by Knopp [13] and Hellsten [12] previously. The experimental data are also extracted from Knopp and Hellsten.

Grid details

A two dimensional C-grid topology (figure 6.4) is used for all the simulations. A grid refinement study is carried out at an angle of attack of $8°$ on meshes with 150, 250 and 450 nodes on the airfoil surface. A $y^+ \approx 0.3$ is maintained for the three grids. A growth ratio of 1.09 is used within the boundary layer. The computational domain extends to 150 chord lengths in all directions . The grid is shown in figure 6.5. The resulting lift and drag coefficients are listed in table 6.2. Since no appreciable difference is observed between the results on the grids with 250 and 450 points (see table 6.2), the grid with 250 points on the airfoil was used for further computations. The far field and wall boundary conditions are applied at the outer edge of the domain and on the airfoil respectively.
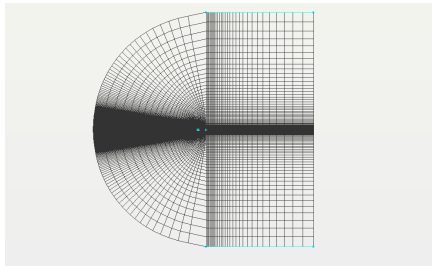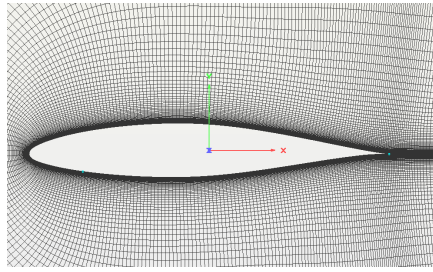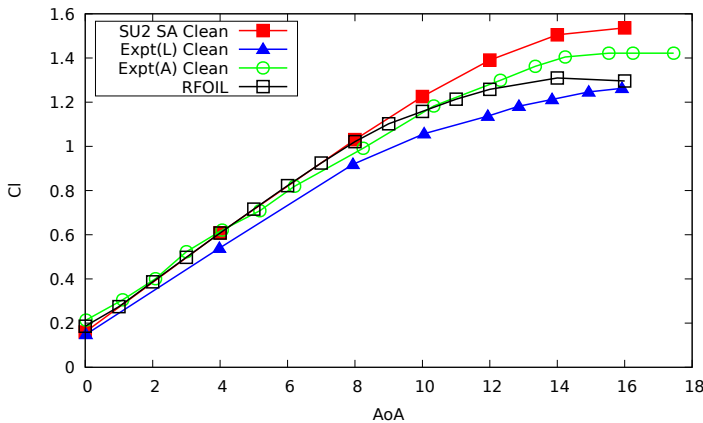

Figure 6.4: Grid used for NACA 65_2215 simulations.

Figure 6.5: Zoom of the grid around NACA 65_2215 airfoil.

N	C_l	C_d
150	1.0273	0.0149
250	1.0336	0.0141
450	1.0346	0.0138

Table 6.2: Lift and Drag coefficients with different grid resolutions for the NACA 65_2215 airfoil.

Figure 6.6: Comparison of NACA 65_2215 polars against experiments and numerical results from SU2 and RFOIL. Expt(L) refers to results from Ljungstrom and Expt(A) from Abbot and von Doenhoff[25].

Clean Results

Figure 6.6 shows the comparison of the numerical results from the SA model under clean conditions. The results from SU2 compare very well against results from RFOIL [9] and the experiments from Abbot [25] at lower angles of attack, but SU2 overpredicts the maximum lift. This could be due to a later prediction of the flow separation by the SA turbulence model compared to the experiments. Since no experimental pressure data is available, this cannot be confirmed. However, the lift values reported by Ljungstrom are significantly lower. Since the two airfoils under consideration are supposed to be very similar, Hellsten [12] concludes that lift values

reported by Ljungstrom are too low likely due to imperfections from a retracted flap in the airfoil geometry setup. The absolute values of the lift coefficients do not compare well against the experimental data from Ljungstrom, but considering the comments of Hellsten the absolute lift coefficient values are not comparable under clean or rough conditions. The maximum lift is observed around an angle of attack of 16° for the clean case in both numerical and experimental data.
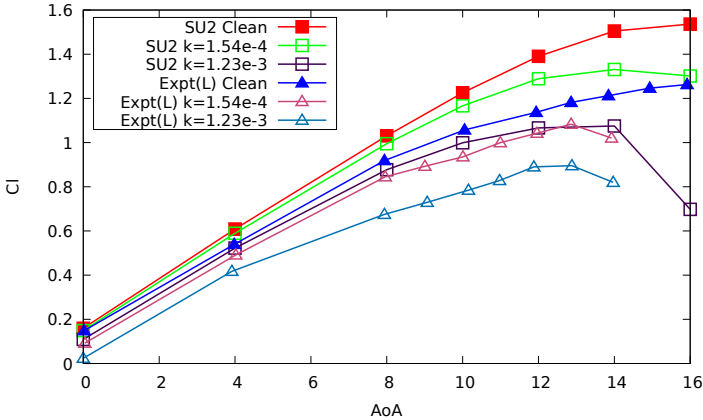

Figure 6.7: Comparison of NACA 65_2215 polars against experiments and numerical results with different roughness heights. Expt(L) refers to results from Ljungstrom.

Rough Results

In figure 6.7 the predicted lift coefficients with different roughness heights are shown. With increasing roughness, the maximum lift value and the angle at which this occurs decrease. Based on the computed skin friction values at an angle of attack of 8°, k_s^+ varies from 70 to about 850. These values suggest the wall is likely to be fully rough but it will vary depending on the flow conditions. As noted earlier, the absolute values of the lift coefficients do not match, but the relative drop of lift from SU2 matches closely with the experiments (table 6.3). However, SU2 predicts a higher value for the angle at which the maximum lift occurs compared to experiments. This is again likely due to the later prediction of the separation location by the SA model.

k_s/c	Experiment	SU2
1.54×10^{-4}	14.22	13.38
3.08×10^{-4}	22.20	19.50
1.23×10^{-3}	29.08	30.03

Table 6.3: Reduction in maximum lift (%) observed in experiments and SU2 for different roughness heights.

6.4.2. DU-96-W300

A typical wind turbine airfoil, DU-97-300, is chosen to test the effect of rough-
ness and verify if VGs can alleviate the anticipated drop in performance. This
choice is motivated by the availability of experimental data for clean and VG cases.
The geometry of the VG is chosen from the AVATAR experimental database (fig-
ure 6.8) [26, 27]. The simulations are carried out at $Re = 2.0 \times 10^6$ for both clean
and VG cases, for which a 3D mesh is generated where the airfoil is extruded in
span direction and a body-fitted mesh is generated around the VG geometry. A
symmetry boundary condition is used on the spanwise extrusion boundaries.

The following cases are considered:

1. Airfoil with no roughness or VGs under fully turbulent conditions (denoted as
   'clean'),

2. Airfoil with VG under fully turbulent conditions ('VG'),

3. Airfoil with roughness ('rough') and

4. Airfoil with VG and roughness ('VGrough').

For the clean airfoil, a grid refinement study is carried out at $AoA = 2.5°$, which
corresponds to the design angle of attack of this airfoil section on the AVATAR
reference turbine blade under normal operating conditions (incoming wind speed of
$10m/s$). The coarsest grid has 128 points (lvl1), the reference grid (lvl2) has 300
points and the finest grid (lvl3) has 512 points on the airfoil and 4 points in the
span direction. Figure 6.8 shows the vortex generator on the airfoil section and
details of the geometry. For the airfoil with VG (zero thickness), 1000 points are
used on the airfoil and 15 points in the span direction (a maximum aspect ratio
of 3 and an average of 1.15 is maintained on the airfoil surface) and no refinement
study is made. The VG geometry is also shown in figure 6.8. The corresponding
dimensions are $h = 5mm$, $D = 35mm$, $d = 17.5mm$ and $\beta = 15°$. The chord length
of the airfoil is $0.65m$ and the VG is placed at 20% chord on the upper surface
of the airfoil [27]. Since a symmetry boundary condition is used on the extrusion
boundaries, the geometry represents a row of counter-rotating VGs as shown in the
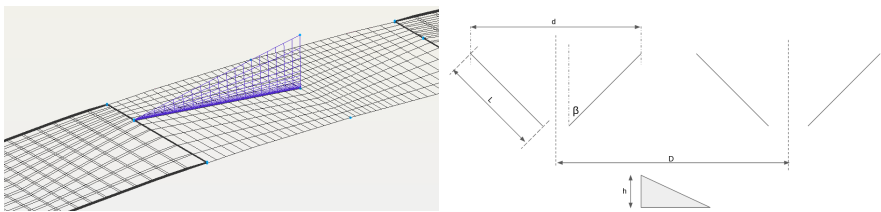right part of figure 6.8.

Figure 6.8: VG on the airfoil surface (left) and VG geometry details [27](right)

Figure 6.9 shows the pressure coefficient and skin friction coefficient along the
airfoil obtained from the three grids. The results from reference grid and fine grid are
almost identical and thus the reference grid will be used for further computations.

The resulting lift and drag coefficients are listed in table 6.4. Only fully turbulent cases are considered for comparison here because the roughness model does not predict the early onset of transition.

Name	N	C_l	C_d
lvl1	128	0.5308	0.0180
lvl2	300	0.5011	0.0161
lvl3	512	0.5077	0.0160

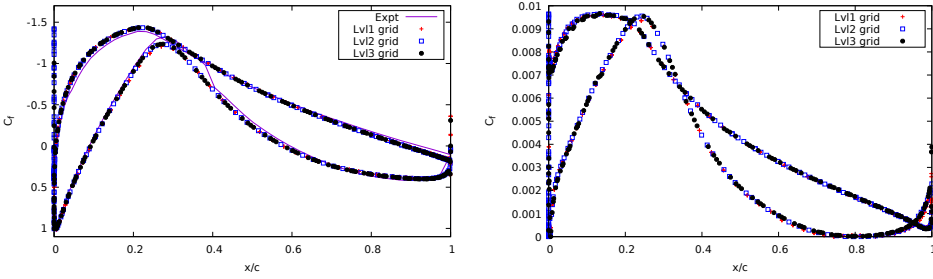Table 6.4: Lift and Drag coefficients with different grid resolutions for the DU-97-W300 airfoil in clean conditions.

Figure 6.9: Comparison of the pressure coefficient (left) and the skin friction coefficient (right) at an $AoA = 2.5°$, $Re = 2.0 \times 10^6$ for different grid resolutions in clean conditions.
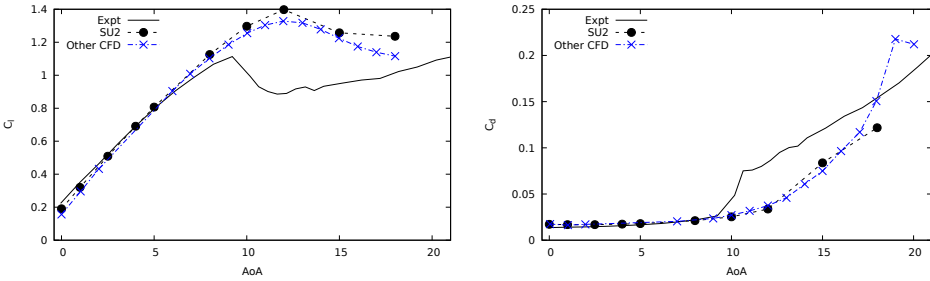
Clean polars

Figure 6.10: Lift (left) and drag (right) polars for the fully turbulent(clean) case at $Re = 2.0 \times 10^6$ in clean conditions.

Figure 6.10 shows the lift and drag polars from SU2 and the experimental data from Baldaccino [27]. Additionally, the lift data from other CFD methods obtained from the Avatar report [26] is also given. The maximum lift angle and the maximum C_l is over estimated by CFD compared to experiments. However, the results from SU2 are in close agreement to those reported by Ellipsys in AVATAR [28] (task 3.2). Similar behavior is observed for C_d as well. The SA model predicts the separation to

occur later than the experiments which results in poor performance at higher angles of attack and over prediction of the maximum lift. While the use of pseudo time stepping scheme helps to overcome some of the convergence issues that a purely steady-state solver would face at higher angles of attack, accuracy of the results remains poor.
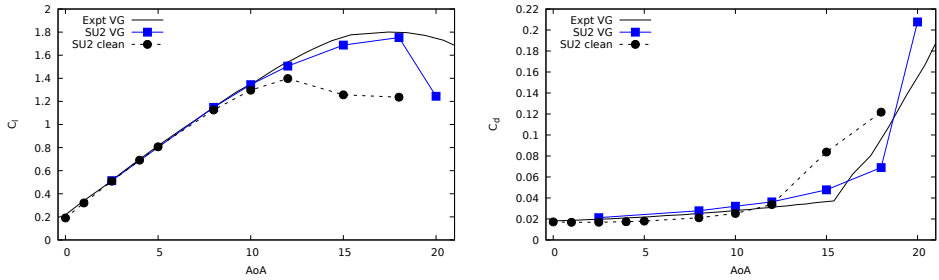
VG polars

Figure 6.11: Lift (left) and drag (right) polars for the fully turbulent case at $Re = 2.0 \times 10^6$ with VG (VG).

Figure 6.11 shows the comparison of the lift and drag polars from SU2 with experimental data [27] at $Re = 2 \times 10^6$ under fully turbulent conditions. Good agreement between the numerical and experimental data is observed at lower angles of attack. SU2 underpredicts the value of the maximum C_l but the stall angle is over predicted. In section 6.4.2, the stall angle predicted by SU2 is around $12°$ which is higher than the experimentally obtained value. From figure 6.11 we observe that the addition of the VG has delayed the stall until an $AoA = 18°$, as expected. A very close match is observed at lower angles but deviations increase at higher angles of attack. Looking at the drag polar on the right the SA model once again predicts separation to occur later than the experiments. However, the maximum lift and the stall angle prediction is much better with VGs than compared to the fully turbulent clean case.

Roughness effects
Determination of the appropriate value of the roughness height, k, is difficult due to lack of experimental data for the airfoil under consideration in rough conditions. Additionally, since no transition model is used in this study, the roughness height used must ideally trigger a very early onset of transition to ensure the flow remains turbulent over the airfoil. Several studies on isolated 3-D roughness elements have reported a critical $Re_{k,crit} > 600$ [20] based on the roughness height which induces larger instabilities in the flow that trigger transition at the location of roughness or even upstream. The study on critical values for distributed roughness is an ongoing research problem [20]. In this case, the roughness height is set to ensure that $Re_k = 800$. Once the roughness height, k, is defined, an equivalent sand grain roughness height, k_s, must be estimated. Langel et al. [29] assume $k_s/k = 1$

for densely packed roughness distribution and a lower value of $k_s/k \approx 0.47$ for lower density (15% distribution density). Aupoix et al. [16] use correlations from Dirling [18] to estimate k_s/k. Following the Dirling's correlation and assuming the distributed roughness to be closely spaced we find $k_s/k \approx 0.539$ which is used to specify the input for the turbulence model considered in this study. Based on these estimates, $k_s/c = 400.0 \times 10^{-6}$ is used. In order to mimic leading edge erosion, the airfoil surface from the leading edge to $x/c = 0.13$ on the pressure side and from leading edge to $x/c = 0.02$ on the suction side is assumed to be rough.
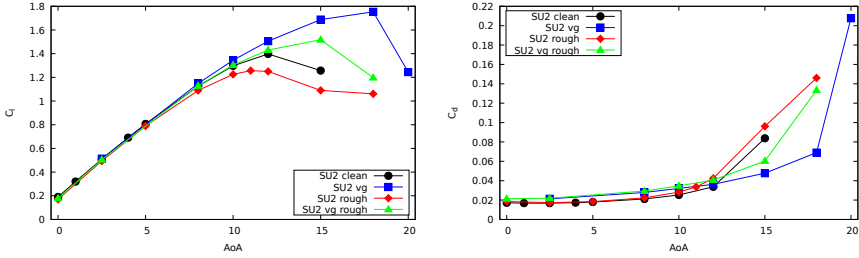

Figure 6.12: Lift (left) and drag (right) polars for the fully turbulent case at $Re = 2.0 \times 10^6$ under different conditions ('clean' -black, 'VG' - blue, 'rough' - red, 'VGrough' - green).

The addition of roughness causes a reduction in the lift and an increase in drag compared to the clean case (the black and red curves in figure 6.12). Based on the computed wall shear stress values the resulting $k_s^+ \approx 240$ corresponds to the fully rough regime. Despite a fairly moderate choice of roughness height (keeping transition in mind), the flow is already in the fully rough regime. Additionally, the airfoil appears to stall slightly earlier due to presence of roughness. Adding a VG on the rough airfoil appears to counteract some of the negative effects of roughness by increasing the lift, however the drag increases further. The VG does delay the stall and the airfoil now stalls at approximately 15°, even with leading edge roughness.
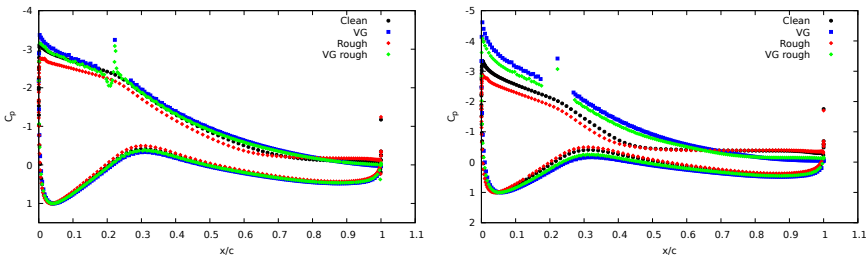

Figure 6.13: Pressure coefficient (C_p) distribution under different cases at $AoA = 12°$ (left) and $AoA = 15°$ (right).

A clearer picture emerges when we investigate the pressure coefficient in different cases (figure 6.13). At an $AoA = 12°$, the fully turbulent (clean) flow is separated close to the trailing edge around $x/c = 0.9$. This angle of attack also corresponds to the maximum C_l. Under rough conditions, the flow separates much earlier however,

the VG helps the flow to remain attached throughout. The difference is clearer when examining the C_p for $AoA = 15°$ in figure 6.13. Under both 'clean' and 'rough' conditions, the airfoil is under stall. The flow remains attached longer with the VG as expected under both clean and rough conditions.
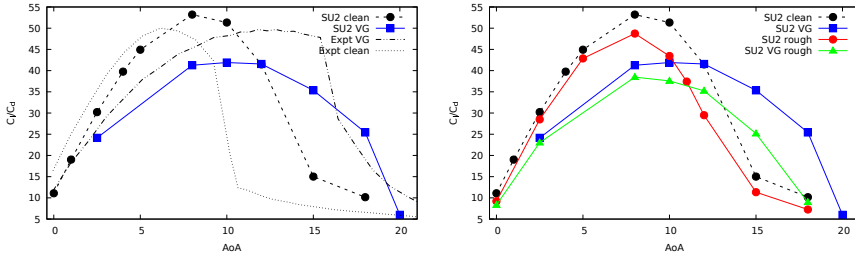
Aerodynamic efficiency

Figure 6.14: Comparison of aerodynamic efficiency for the different cases.

Figure 6.14 shows the aerodynamic efficiencies under different conditions. On the left, the comparison between numerical SU2 results and experiments are shown. There is an under prediction of efficiency at lower angles of attack due to over prediction of the drag. As also seen in the lift polar results, there is an over prediction in maximum efficiency and the angle where it occurs. However, in the VG case, there is a consistent under prediction in efficiency due to over prediction of the drag. Comparing the clean and VG cases, the efficiency at lower angles is lower with VG due to additional drag but at higher angles of attack (beyond stall), the efficiency with VGs remains high as expected. On the right, the comparison of efficiencies with roughness is shown. Due to roughness, a reduction in efficiency is observed both with and without VGs as expected. The maximum efficiency is also reduced in both cases. At higher angles of attack, the VG increases the efficiency even under rough conditions.

6.4.3. DU 96-W-180
In this section, the SA roughness model is applied to the DU96-W-180 airfoil. This is an 18% thick airfoil developed by Delft University [30] and is widely used in the wind energy community. Sareen et al [7] performed experiments on this airfoil at different Reynolds numbers under different stages and types of erosion. They determine the levels of erosion based on photographs of eroded blades. In this study, the leading edge erosion due to pits and gouges (see figure 6.15) under the two most severe stages are considered at $Re = 1.85 \times 10^6$. These cases correspond to Type B stage 3 and stage 4 as reported in [7].

The depths of pits and gouges for these cases are respectively $0.51mm$ and $2.54mm$. The pits and gouges have equal depths and diameters. The chord-wise extent of the pits and gouges are 10% on the upper surface and 13% on the lower surface. The number of pits and gouges on the lower surface is also 1.3 times that

on the upper surface. In stage 3 there are 400 pits and 200 gouges on the upper surface and in stage 4 there are 800 pits and 400 gouges on the upper surface.
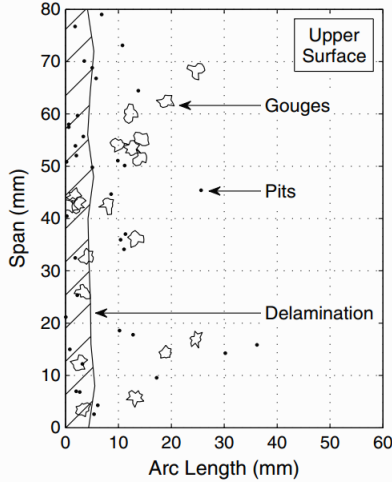

Figure 6.15: Illustration of pits, gouges and delamination of a turbine blade from Sareen et al[7].

Grid details

As seen in section 6.3.1, the SA roughness model requires a wall normal grid spacing that corresponds to $y^+ \approx 0.3$ under clean conditions to obtain grid converged results in rough conditions. Thus, this minimum grid spacing is maintained. A grid refinement study is carried out at an angle of attack of $8°$ with $N = 125, 250, 500$ and 750 points along the airfoil. A growth ratio of 1.09 is used in the normal direction. The resulting lift and drag coefficients are listed in table 6.5 along with the fully turbulent results obtained from RFOIL [9]. Based on these results the grid with $N = 500$ points on the airfoil is chosen for further analysis.

N	C_l	C_d
125	1.028934	0.020944
250	1.065950	0.016588
500	1.069648	0.015781
750	1.069287	0.015704
RFOIL	1.054832	0.015551

Table 6.5: Lift and Drag coefficients with different grid resolutions for the DU95-W-180 airfoil at an angle of attack of $8°$.

Clean results

A baseline case of fully turbulent flow is considered first. A transition model is not considered since the effect of roughness on transition is not implemented. Fig-
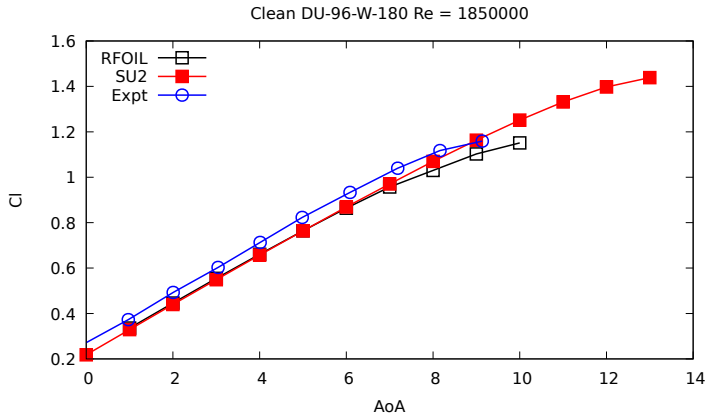
Figure 6.16: Comparison of lift coefficient (C_l) against angle of attack for fully turbulent flow against experimental data.

ure 6.16 shows the lift coefficient at different angles of attack from SU2 and RFOIL under fully turbulent conditions compared to experimental data. Since no mention of tripping the flow is made in [7], it is likely that the flow is not fully turbulent but transitional, especially at lower angles of attack. Consequently, a consistent underprediction of lift is observed in both numerical tools. The results from SU2 and RFOIL match closely in the linear region and deviate at higher angles of attack. Figure 6.17 shows the comparison of lift and drag coefficients of the two numerical
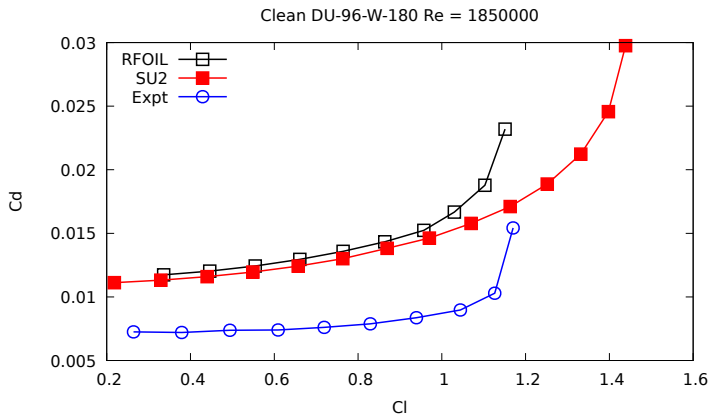

Figure 6.17: Comparison of lift coefficient (C_l) against drag coefficient (C_d) for fully turbulent flow against experimental data.

results from SU2 and RFOIL with the experimental data. Once again, since the experimental flow conditions were not fully turbulent, there is a consistent over-prediction of the drag coefficient by both SU2 and RFOIL. As seen in figure 6.16,

there is good agreement between the numerical results at lower angles of attack. However, RFOIL predicts increasing flow separation to occur from an $AoA = 9°$, which is close to what is observed in the experiment but this is not predicted by SU2. This is likely due to the poor prediction of separation by the SA turbulence model, which was also observed earlier.

Equivalent sand grain roughness

The roughness height, k, is usually defined as the height or depth of roughness elements on the surface, for example, the depth of pits and gouges in figure 6.15. Determination of the equivalent sand grain roughness height, k_s, from the roughness height k is an active area of research. The roughness density parameter, Λ_s, is widely used in literature as a means of relating geometric surface roughness with equivalent sand grain roughness

$$\Lambda_s = \frac{S}{S_f}\left(\frac{A_f}{A_s}\right)^{1.6},\tag{6.11}$$

where S is the total wall area where roughness is present, S_f is the roughness frontal area, A_f is the frontal area of a single roughness element, and A_s is the surface area of a single roughness element in the direction of the flow. Based on data from Schlichtling's experiments, Danberg and Sigal [31] proposed the following relations for 2-D

$$\frac{k_s}{k} = \begin{cases} 3.21 \times 10^{-3}\Lambda_s^{4.935}, & 1.4 \leq \Lambda_s \leq 4.89, \\ 8, & 4.89 \leq \Lambda_s \leq 13.25, \\ 151.71\Lambda_s^{-1.1379}, & 13.25 \leq \Lambda_s \leq 100, \end{cases}\tag{6.12}$$

and 3-D

$$\frac{k_s}{k} = 160.77\Lambda_s^{-1.3376}, \quad 16 \leq \Lambda_s \leq 200.\tag{6.13}$$

Van Rij et al. [32] generalized the roughness shape factor A_f/A_s for irregular 3-D roughness elements as S_f/S_w where S_f is the total frontal area of all roughness elements and S_w is the total wetted area of all roughness elements and proposed the following relation

$$\frac{k_s}{k} = \begin{cases} 1.58 \times 10^{-5}\Lambda_s^{5.683}, & \Lambda_s \leq 7.84, \\ 1.802\Lambda_s^{0.03038}, & 7.84 \leq \Lambda_s \leq 28.12, \\ 255.5\Lambda_s^{-1.454}, & 28.12 \leq \Lambda_s. \end{cases}\tag{6.14}$$

McClain [33] used the discrete element method approach and proposed a single relation as

$$\frac{k_s}{k} = 927.317\Lambda_s^{-1.669}.\tag{6.15}$$

However, these correlations are mainly derived by adding roughness elements like spheres, cones and hemispheres and their validity for negative roughness like pits and gouges is not clear. Various researchers have used statistical representations of rough surfaces in combination with experiments and numerical simulations using LES and DNS to obtain more general correlations based on rms height (k_{rms}),

skewness (Sk) and higher order moments of the rough surface height probability density functions. The k_{rms} and skewness Sk can be computed as

$$k_{rms} = \sqrt{\frac{1}{N}\sum_i k_i^2}, \quad Sk = \frac{1}{N}\sum_i \left(\frac{k_i}{k_{rms}}\right)^3, \tag{6.16}$$

where k_i are the heights or depths of individual roughness elements, for example a pit, and N is the total number of such roughness elements. Flack and Schultz [34] proposed

$$k_s = 4.43 k_{rms}(1 + Sk)^{1.37} \tag{6.17}$$

but note that it is not very general since it does not include information about roughness density. In a more recent study, Flack et al [35] proposed different relations for different types of skewness as

$$k_s = 2.48 k_{rms}(1 + Sk)^{2.24} \tag{6.18}$$

for positive skewness,

$$k_s = 2.73 k_{rms}(2 + Sk)^{-0.45} \tag{6.19}$$

for negative skewness and

$$k_s = 2.11 k_{rms} \tag{6.20}$$

for zero skewness. They also note that negatively skewed surfaces like those with pits and gouges have a smaller impact on drag due to roughness elements than positive skewness. Flack and Schultz [36] and Forooghi et al. [37] also note that another parameter that accounts for sparse roughness is necessary and propose a relation of the form

$$k_s/k_z = F(Sk)G(ES), \tag{6.21}$$

where ES is the effective slope which is related to the solidity of roughness (λ) as $ES = 2\lambda$ and k_z is related to k_{rms} as $k_z = 4.4 k_{rms}$. Note that solidity is defined as the ratio of total roughness frontal area (S_f) to total wall area (S) i.e. $\lambda = S_f/S$. They recommend

$$F(Sk) = 0.67 Sk^2 + 0.93 Sk + 1.3 \tag{6.22}$$

and

$$G(ES) = 1.07(1 - e^{-3.5ES}). \tag{6.23}$$

In this study, equations. 6.21, 6.22 and 6.23 suggested by [37] are used.

Roughness definition
Sareen et al. [7] create different amounts of roughness on the upper and lower surface with the lower surface being 1.3 times rougher than the upper surface. For the type B stage 3 erosion level they add 400 pits and 200 gouges on the upper surface and 520 pits and 260 gouges on the lower surface. In stage 4 the number of pits and gouges are doubled both on the upper and lower surfaces. The rough surface extends from the leading edge to $x/c = 0.1$ on the upper surface and from the leading edge to $x/c = 0.13$ on the lower surface in both cases. The computed statistics are listed in table 6.6.

	Stage 3	Stage 4
k_{rms}	$1.524mm$	$1.524mm$
Sk	-1.56695	-1.56695
ES	0.0563	0.1126
k_s/c	0.00418	0.00760

Table 6.6: Roughness definition for DU-96-W-180 based on Sareen et al[7].

Rough results

Figure 6.18 shows the comparison of the lift coefficient as a function of the angle of attack between SU2 and experiments for stage 3 erosion. There is a small underprediction of lift at lower angles of attack, similar to what was observed in the clean case. This is likely due to the flow still being mildly transitional at lower angles of attack. With increasing angle of attack, the prediction from SU2 matches the experimental data quite closely, likely due to the flow becoming fully turbulent in the experiment. Figure 6.19 shows the drag and lift coefficients. Once again,
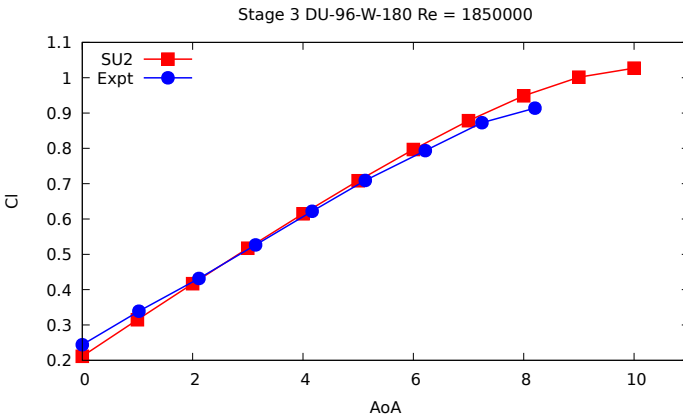
Figure 6.18: Comparison of lift coefficient (C_l) against angle of attack for fully turbulent flow against experimental data (stage 3 see table 6.6).

the numerical results from SU2 overpredict the drag compared to the experimental data. Flow separation starts to occur before $AoA \approx 8°$ in the experiments whereas SU2 does not predict separation till after $AoA \approx 9°$.

Figure 6.20 shows the comparison of the lift coefficient as a function of the angle of attack between SU2 and experiments forstage 4 erosion. The numerical results agree with the experiments more closely compared to stage 3 likely due to the flow being fully turbulent due to the higher roughness level. Figure 6.21 shows the drag and lift coefficients. Once again, numerical results from SU2 overpredict the drag compared to the experimental data.

Figures 6.19 and 6.21 also show the lift and drag values in clean conditions. The increase in drag even at lower angles of attack can be seen clearly. The maximum lift
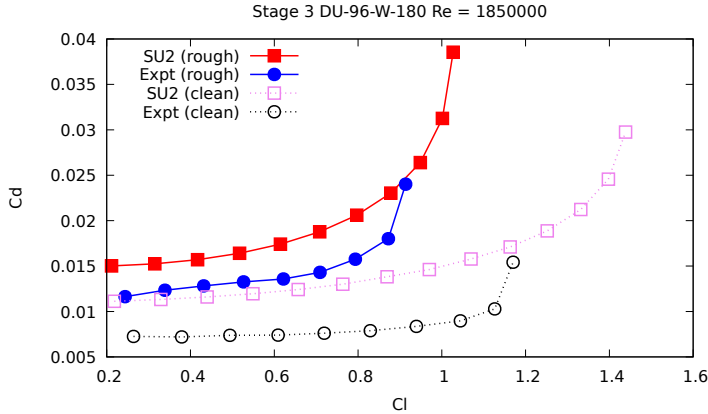
Stage 3 DU-96-W-180 Re = 1850000

Figure 6.19: Comparison of lift coefficient (C_l) against drag coefficient (C_d) for fully turbulent flow against experimental data (stage 3 see table 6.6).
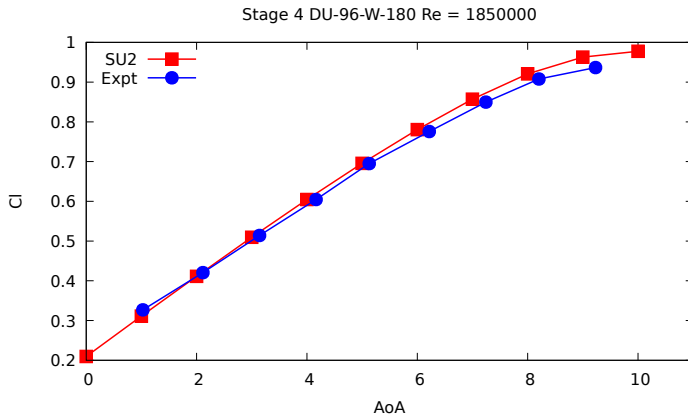
Stage 4 DU-96-W-180 Re = 1850000

Figure 6.20: Comparison of lift coefficient (C_l) against angle of attack for fully turbulent flow against experimental data (stage 4 see table 6.6).

also decreases in rough conditions for both roughness levels considered. However, since Sareen et al. [7] do not report lift and drag values at higher angles of attack, the magnitude of reduction cannot be compared. It is very likely that the airfoil will stall earlier for both the roughness cases compared to the clean conditions.

Discussion　　In this section the SA roughness model was first validated against experiments on the NACA 65_2215 airfoil with a given equivalent sand grain roughness. The SA model predicted the drop in lift very closely compared to the experiments. Subsequently, the SA model was used on the DU-96-W-180 airfoil with 'negative' roughness. It was seen that a statistical description of the surface is required to
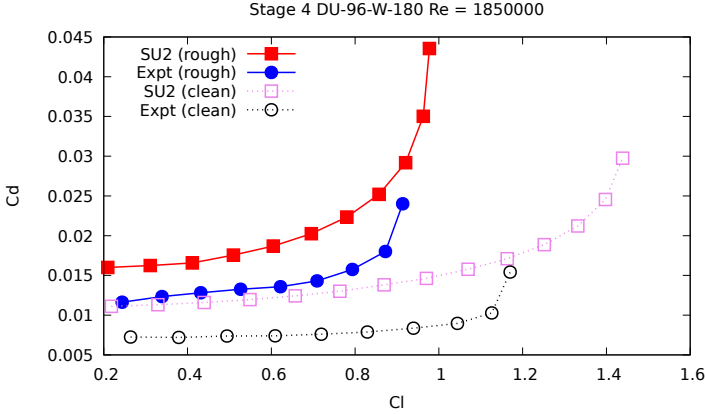
Figure 6.21: Comparison of lift coefficient (C_l) against drag coefficient (C_d) for fully turbulent flow against experimental data (stage 4 see table 6.6).

accurately calculate the equivalent sand grain roughness. Results under clean conditions differed from the experiments likely due to the the experiments not being under fully turbulent conditions, but the numerical results, especially lift coefficient, matched closely with the experimental data under roughness when the flow is likely fully turbulent. It was seen that roughness causes a considerable reduction in lift and increase in drag and can lead to premature stalling of the airfoils.

6.5. Boundary layer analysis

Since the NACA 65_2215 airfoil has a larger rough surface than the DU-96-W-180, it is chosen for the boundary layer analysis. The boundary layer parameters for SU2 are computed by extracting the velocity vector along surface normals at various points along the airfoil. The edge of the boundary layer is assumed to be at the location where the ratio of the magnitude of the vorticity at that location to the value at the wall is less than 10^{-4}. In this section the effect of roughness on the boundary layer properties of airfoils will be investigated.

6.5.1. Integral boundary layer methods

Viscous inviscid interaction methods like RFOIL [9] are widely used to analyze the performance of airfoils. Panel methods are typically used to model the inviscid part of the flow and integral boundary layer methods for the viscous part. Integral boundary layer equations are obtained by integrating the boundary layer equations in the direction normal to the wall. More details on deriving the governing equations can be found in [38, 39]. The new integral quantities introduced are displacement thickness δ^*, momentum thickness θ and kinetic energy thickness δ_k.

$$\delta^* = \int_0^\delta \left(1 - \frac{u}{u_e}\right) dy, \quad \theta = \int_0^\delta \frac{u}{u_e}\left(1 - \frac{u}{u_e}\right) dy.$$

$$\delta_k = \int_0^\delta \frac{u}{u_e} \left(1 - \left(\frac{u}{u_e} \right)^2 \right) dy, \tag{6.24}$$

Here u is the local velocity, δ is the boundary layer thickness, u_e is the velocity magnitude at the edge of the boundary layer and y is the wall normal direction. Further, the following shape parameters are defined

$$H = \frac{\delta^*}{\theta}, \quad H_k = \frac{\delta_k}{\theta}. \tag{6.25}$$

The governing equations, resulting from integration of the continuity and momentum equations, used in RFOIL are

$$\frac{d\theta}{dx} + (2 + H - M_e^2) \frac{\theta}{u_e} \frac{du_e}{dx} = C_f/2, \tag{6.26}$$

$$\theta \frac{dH_k}{dx} + (2H^{**} + H_k(1 - H)) \frac{\theta}{u_e} \frac{du_e}{dx} = 2C_D - H_k C_f/2. \tag{6.27}$$

Note that other formulations of the integral boundary layer equations are used in other tools [39]. In order to close the equations, closure relations [38, 39] are defined for the kinetic energy shape factor H_k, the skin friction coefficient C_f and the dissipation coefficient C_D. The closure relations are different for laminar and turbulent flows. For turbulent flows, an additional equation for lag in Reynolds shear stress (C_τ) is also solved. H^{**} is a shape factor based on the variation of density within the boundary layer and M_e is the Mach number of the external flow. Both can be ignored for incompressible flows. These closure relations are defined in terms of the shape factors introduced earlier and the Reynolds number based on momentum thickness Re_θ, where $Re_\theta = u_e \theta / \nu$ with ν being the kinematic viscosity. In the following sections, the effect of roughness on the different thicknesses, shape factors and closure relations are examined.

6.5.2. Clean results

First the calculated integral boundary layer quantities from SU2 under clean conditions are compared against the RFOIL results. It must be noted that the $X-$axis of all the plots in this section range from $x/c = 0.025$ to $x/c = 1$ to avoid the stagnation region. Figure 6.22 shows the displacement thickness on both the upper and lower sides at angles of attack of $0°$ and $4°$. The calculated displacement thickness matches closely with the values from RFOIL with some deviation near the trailing edge in both cases. The momentum thickness is slightly overpredicted by SU2 after $x/c = 0.4$ at an angle of attack of $0°$ but matches closely for an angle of attack of $4°$ as seen in figure 6.23. The comparisons of the shape factors are shown in figure 6.24. The shape factor is larger for $AoA = 4°$ compared to $AoA = 0°$ indicating a thicker boundary layer as the angle of attack increases. While the computed shape factors from RFOIL and SU2 do not match exactly, both display similar behavior initially decreasing towards the middle of the airfoil and increasing near the trailing edge.
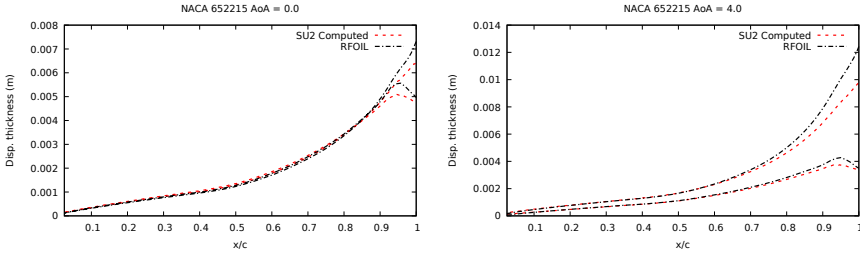
Figure 6.22: Displacement thickness (δ^*) from SU2 and RFOIL at an angle of attack of $0°$ (left) and $4°$ (right) for the NACA 65_2215 airfoil.
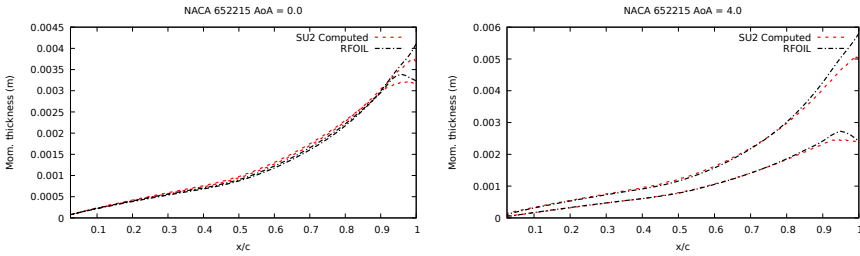

Figure 6.23: Momentum thickness (θ) from SU2 and RFOIL at an angle of attack of $0°$ (left) and $4°$ (right) for the NACA 65_2215 airfoil.
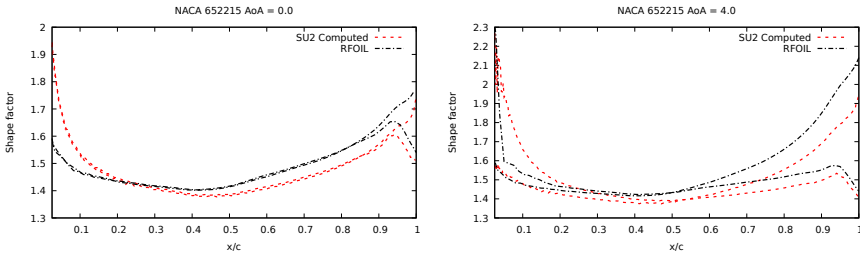

Figure 6.24: Shape factor (H) from SU2 and RFOIL at an angle of attack of $0°$ (left) and $4°$ (right) for the NACA 65_2215 airfoil.

In RFOIL the local skin friction coefficient is computed as [38]

$$C_f = \frac{0.3 \exp(-1.33H)}{(log_{10}Re_\theta)^{1.74+0.31H}} +$$
$$0.00011\Big(tanh\Big(4.0 - \frac{H}{0.875}\Big) - 1.0\Big). \tag{6.28}$$

Here Re_θ is the local Reynolds number based on momentum thickness θ. Figure 6.25 shows the comparison of the skin friction coefficient between RFOIL, the values originally reported by SU2 for the RANS computation (denoted as 'SU2 original') and the skin friction calculated based on the computed integral boundary layer quantities

in equation 6.28 (denoted as 'SU2 Computed'). The C_f computed from the integral quantities using equation 6.28 match the SU2 RANS solution and RFOIL results quite well after $x/c = 0.25$. The mismatch near the leading edge for $AoA = 0°$ is likely due to errors in computing the integral quantities near the stagnation region.
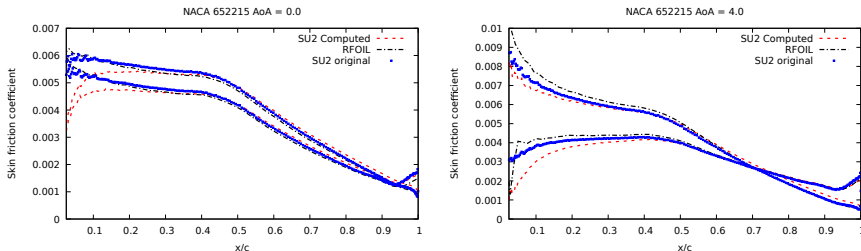

Figure 6.25: Skin friction coefficient (C_f) from SU2 and RFOIL at an angle of attack of $0°$ (left) and $4°$ (right) for the NACA 65_2215 airfoil.

6.5.3. Rough results

Since only the entire upper surface is rough, only the results for the upper surface only are presented in this section. Figures 6.26 and 6.27 show the displacement and momentum thickness for different roughness levels compared to the clean case. As expected, these thicknesses increase with increasing roughness. A very steep increase is observed in the momentum thickness near the trailing edge for the largest roughness.
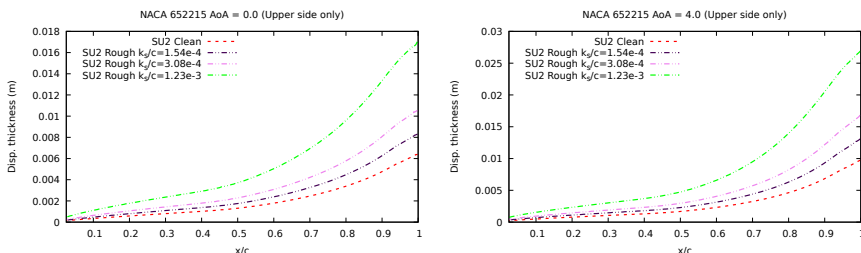

Figure 6.26: Displacement thickness (δ^*) from SU2 under different roughness levels at an angle of attack of $0°$ (left) and $4°$ (right) for the NACA 65_2215 airfoil.

Figure 6.28 shows the shape factors for different roughness levels compared to the clean case at $AoA = 0°$ and $AoA = 4°$. The shape factor increases for all roughness levels with the largest increase for $k_s = 1.23 \times 10^{-3}$. The maximum k_s^+ values varies with angle of attack. At an angle of attack of $0°$, the k_s^+ are 25, 75 and 286 indicating that the flow is in the transitional rough region for the two lower roughness levels and is fully rough for the highest roughness level. However, at an angle of attack of $4°$, the maximum k_s^+ values are 75, 180 and 750 indicating that
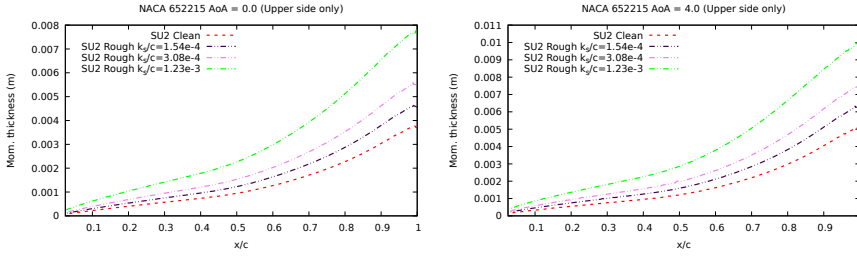
Figure 6.27: Momentum thickness (θ) from SU2 under different roughness levels at an angle of attack of $0°$ (left) and $4°$ (right) for the NACA 65_2215 airfoil.

the flow is fully rough for the $k_s/c = 3.08 \times 10^{-4}$ case also. From figure 6.28 it is seen that the behavior of the shape factor in the $k_s/c = 3.08 \times 10^{-4}$ case is similar for both angles of attack despite one being transitionally rough and the other fully rough.
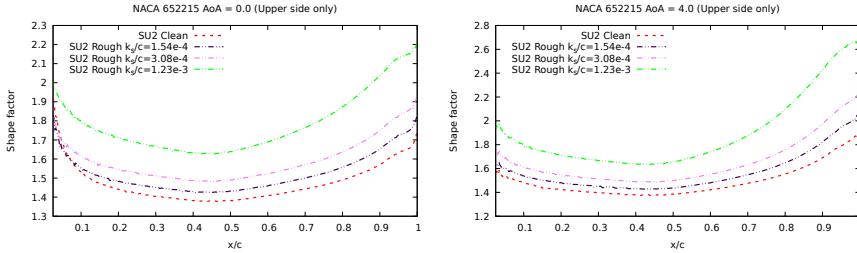

Figure 6.28: Shape factor (H) from SU2 under different roughness levels at an angle of attack of $0°$ (left) and $4°$ (right) for the NACA 65_2215 airfoil.

Skin friction coefficient

Equation 6.28 will not be valid here as the properties of the boundary layer change due to roughness. Olsen et al. [10] suggested a new closure relation for skin friction for rough surfaces including the Reynolds number based on roughness height, $Re_k = u_e k/\nu$ as

$$C_f = \frac{0.9 \exp(-2.4H)}{(|log_{10}Re_\theta - log_{10}Re_k + 1.11|)^{2.45-0.15H}}. \tag{6.29}$$

Figure 6.29 shows the skin friction from equations 6.28 and 6.29, clean and rough SU2 results at angles of attack of $0°$ (left) and $4°$ (right). Clearly equation 6.28 is not valid for rough surfaces. The new closure relation proposed by Olsen et al. appears to overpredict the skin friction. However, since the computed Re_k for the first two roughness levels are approximately 400 and 800, it is outside the range of the data used by the authors in their study. The third roughness level has an average $Re_k \approx 3000$ and is within the valid range of data used to derive the model.
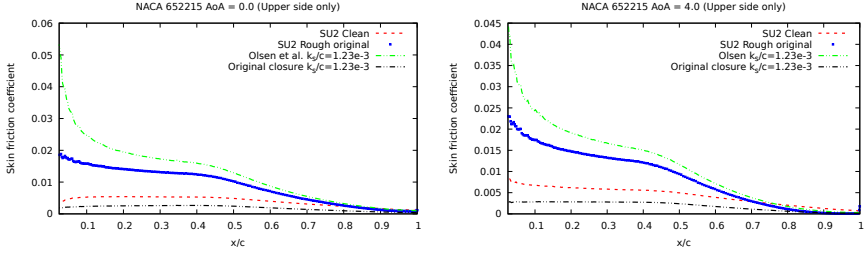
Figure 6.29: Skin friction coefficient (C_f) comparison between RANS solution from SU2 (clean in red and rough in blue) and closure relations from Olsen et al.[10] and RFOIL[9] at an angle of attack of $0°$ (top) and $4°$ (bottom) for the NACA 65_2215 airfoil.

The authors report convergence difficulties when roughness was applied to regions before $x/c = 0.6$ and from the figure 6.29 it can be seen that C_f is overpredicted by a significant amount in that region and is closer to the values reported by SU2 after $x/c = 0.6$.

Kinetic energy shape factor

As seen above the closure sets for skin friction are not valid for rough airfoils. The other closure relation defined in terms of H and Re_θ is for the kinetic energy shape factor H_k. Closure relations for other quantities are defined in terms of C_f and H_k. Thus, the validity of the H_k closure is examined here in detail. For turbulent flows in RFOIL the following closure relations are used to compute H_k. First define

$$H_0 = \begin{cases} 3.0 + \frac{400}{Re_\theta}, & Re_\theta > 400, \\ 4.0 & Re_\theta \leq 400. \end{cases} \tag{6.30}$$

Then for $H < H_0$

$$H_k = \left(0.5 - \frac{4.0}{Re_\theta}\right)\left(\frac{H_0 - H}{H_0 - 1}\right)^2 \frac{1.5}{H + 0.5} + 1.5 + \frac{4}{Re_\theta}, \tag{6.31}$$

otherwise

$$H_k = 1.5 + \frac{4.0}{Re_\theta} + (H - H_0)^2 \left[\frac{0.04}{Re_\theta} + 0.007\frac{lnRe_\theta}{\left(H - H_0 + \frac{4}{Re_\theta}\right)^2}\right] \tag{6.32}$$

The computed H_k based on equation 6.25 (denoted by symbols) and those based on the closure relations in equations 6.31 and 6.32 (denoted by solid lines) are shown in figure 6.30. The computed values agree with the closure relations closely for the clean case and also for the two lowest roughness cases. However, as the level of roughness increases the closure relation does not predict H_k accurately. The Re_{k_s} of the first two roughness cases are approximately 400 and 800, indicating that the closure sets are likely valid for small roughness levels but deviate for higher roughness levels. The deviation observed in the third roughness level is also much

less than the deviation observed for the skin friction coefficient. Figure 6.31 shows the variation of H_k for a higher angle of attack of $12°$. From the top figure it is seen that the behavior of H_k is similar to that observed for lower angles of attack when the flow is attached. However, as the bottom figure shows, the deviation increases for all roughness levels when the flow separates. The wiggles observed are likely an artifact of how the edge of the boundary layer is detected during the post processing.
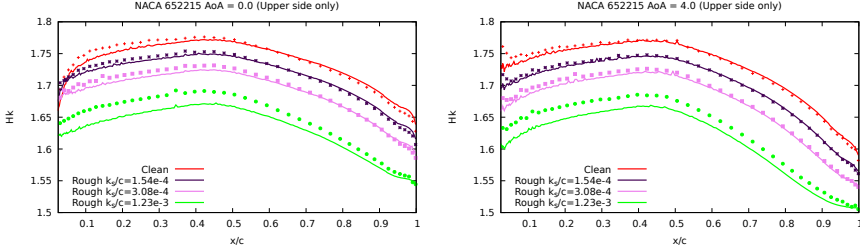

Figure 6.30: H_k from SU2 under different roughness levels at an angle of attack of $0°$ (left) and $4°$ (right) for the NACA 65_2215 airfoil. Computed values (equation 6.25) shown as symbols and result from the closure relations (equations 6.31 and 6.32) as solid lines.
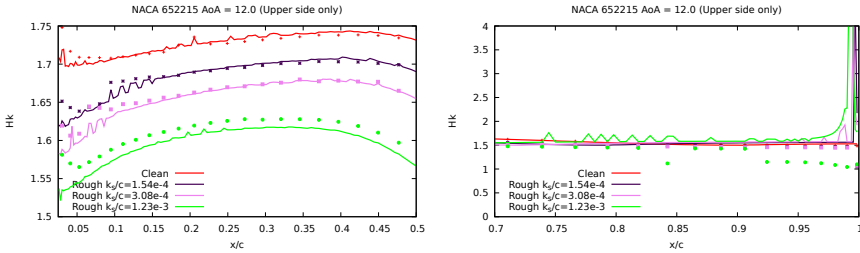

Figure 6.31: H_k from SU2 under different roughness levels at an angle of attack of $12°$ for the NACA 65_2215 airfoil. Left figure shows the plot from $x/c = 0.025$ to $x/c = 0.5$. Right figure shows the zoomed in region around the TE for all roughness levels. Computed values (equation 6.25) shown as symbols and result from the closure relations (equations 6.31 and 6.32) as solid lines.

Since the closure relations for the dissipation coefficient (C_D) and for turbulent flows the Reynolds shear stress coefficient (C_τ) are based on H, Re_θ, C_f and H_k, all of which change with roughness, new closure relations need to be defined. Thus, in order to model roughness in integral boundary layer method based tools like RFOIL, new closure relations need to be derived for all of the above quantities.

Discussion   In this section, the different integral boundary layer quantities and closure relations used in RFOIL are examined under clean and rough conditions. Three different roughness levels were considered corresponding to Re_{k_s} of approximately 400, 800 and 3000. The boundary layer thicknesses increase due to roughness

and the shape factor is also higher. A larger shape factor typically implies a thicker boundary layer that is prone to separation. The shape factor remained less than 2 for the lower two roughness levels but for the highest roughness level the shape factor neared the value for separation even at low angles of attack. Additionally, it was seen that the variation of the shape factor followed the variation of Re_{k_s} more closely than the variation of k_s^+.

Closure relations are crucial for an accurate solution in integral boundary layer methods. The performance of the closures for skin friction and kinetic energy shape factor was examined under rough conditions. The closure relation for skin friction underpredicted significantly for all cases. The new closure relation proposed by Olsen et al. [10] was observed to overpredict the skin friction. The kinetic energy shape factor closure relation was less sensitive to roughness and showed significant deviation only for the largest roughness case and separated flow.

6.6. Conclusions and future outlook

Roughness models for two RANS turbulence models, SA and SST, were implemented in SU2 and the accuracy was examined via grid refinement. The models were validated against empirical models for the shift in velocity profiles in the boundary layer and experimental skin friction data for flat plates. It was seen that that the SST roughness model required a much finer grid compared to the SA roughness model to give a grid independent solutions. However, despite the finer grid the results from the SST roughness model did not match the experimental data or the empirical models under fully rough conditions, unlike the SA roughness model. Based on these results the SA roughness model was further validated against experimental data on two different airfoils. The SA model predicted the reduction in lift for different roughness levels accurately for the NACA 65_2215 airfoil. The SA model was also validated against an experiment with negative roughness (pits and gouges) on the DU-96-W-180 airfoil. Encouraging results were observed for both roughness levels tested. The statistical method to determine the equivalent sand grain roughness proved to be accurate. Some differences were observed in the clean simulation, most likely due to the fact the simulations were run under fully turbulent conditions, unlike the experiments.

Further, the behavior of different integral boundary layer properties like displacement thickness, momentum thickness, shape factors and closures were investigated for the NACA 65_2215 airfoil. The existing skin friction closure relations for clean surfaces greatly underpredict the skin friction (C_f) and are not valid for rough surfaces. However, the closure relation for the kinetic energy shape factor (H_k) performed well for low roughness levels ($Re_{k_s} < 1000$)) but deviated at higher roughness levels and under separation. The deviation was only marginal compared to the skin friction closure relation. However, since the closure relations for other quantities like the dissipation coefficient and the Reynolds shear stress coefficient depend on C_f and H_k new closure relations will be needed in order to simulate rough surfaces in integral boundary layer tools like RFOIL.

The main focus of this study was on the effect of roughness on turbulent boundary layers. For laminar boundary layers, roughness leads to premature transition to

turbulence. In order to fully capture the effect of roughness, the effect on transition will also be considered in the future. Further, more boundary layer data at different roughness levels are needed to derive new closure relations for integral boundary layer methods.

References

[1] A. Koodly Ravishankara, H. Özdemir, and E. van der Weide, *Analysis of leading edge erosion effects on turbulent flow over airfoils,* Renewable energy **172**, 765 (2021).

[2] R. Herring, K. Dyer, F. Martin, and C. Ward, *The increasing importance of leading edge erosion and a review of existing protection solutions,* Renewable and Sustainable Energy Reviews **115**, 109382 (2019).

[3] W. Han, J. Kim, and B. Kim, *Effects of contamination and erosion at the leading edge of blade tip airfoils on the annual energy production of wind turbines,* Renewable Energy **115**, 817 (2018).

[4] R. S. Ehrmann, B. Wilcox, E. B. White, and D. C. Maniaci, *Effect of Surface Roughness on Wind Turbine Performance.*, Tech. Rep. (Sandia National Lab.(SNL-NM), Albuquerque, NM (United States), 2017).

[5] C. M. Langel, R. Chow, C. Van Dam, and D. C. Maniaci, *RANS Based Methodology for Predicting the Influence of Leading Edge Erosion on Airfoil Performance*, Tech. Rep. (Sandia National Lab.(SNL-NM), Albuquerque, NM (United States), 2017).

[6] F. Menter, R. Langtry, and S. Völker, *Transition modelling for general purpose cfd codes,* Flow, turbulence and combustion **77**, 277 (2006).

[7] A. Sareen, C. A. Sapre, and M. S. Selig, *Effects of leading edge erosion on wind turbine blade performance,* Wind Energy **17**, 1531 (2014).

[8] J. Nikuradse *et al.*, *Laws of flow in rough pipes,* (1950).

[9] R. Van Rooij, *Modification of the boundary layer calculation in rfoil for improved airfoil stall prediction,* (1996).

[10] A. S. Olsen, N. Ramos-García, and C. Bak, *Improved roughness model for turbulent flow in 2d viscous-inviscid panel methods,* Wind Energy **23**, 608 (2020).

[11] F. Palacios, S. Padron, B. Tracey, D. E. Manosalvas, A. Aranake, S. R. Copeland, A. Variyar, J. J. Alonso, T. W. Lukaczyk, A. K. Lonkar, K. R. Naik, and T. D. Economon, *Stanford University Unstructured (SU2): Analysis and Design Technology for Turbulent Flows,* , 1 (2014).

[12] A. Hellsten and S. Laine, *Extension of the k-omega-sst turbulence model for flows over rough surfaces,* in *22nd Atmospheric Flight Mechanics Conference* (1997) p. 3577.

[13] T. Knopp, B. Eisfeld, and J. B. Calvo, *A new extension for k–ω turbulence models to account for wall roughness,* International Journal of Heat and Fluid Flow **30**, 54 (2009).

[14] S. B. Pope, *Turbulent Flows* (Cambridge University Press, 2000).

[15] H. Schlichting and K. Gersten, *Boundary-layer theory* (Springer, 2016).

[16] B. Aupoix and P. Spalart, *Extensions of the spalart–allmaras turbulence model to account for wall roughness,* International Journal of Heat and Fluid Flow **24**, 454 (2003).

[17] A. L. Braslow and E. C. Knox, *Simplified method for determination of critical height of distributed roughness particles for boundary-layer transition at mach numbers from 0 to 5,* (1958).

[18] R. Dirling, Jr, *A method for computing roughwall heat transfer rates on reentry nosetips,* in *8th Thermophysics Conference* (1973) p. 763.

[19] R. Grabow and C. White, *Surface roughness effects nosetip ablation characteristics,* AIAA Journal **13**, 605 (1975).

[20] C. M. Langel, R. Chow, C. P. v. Dam, M. A. Rumsey, D. C. Maniaci, R. S. Ehrmann, and E. B. White, *A computational approach to simulating the effects of realistic surface roughness on boundary layer transition,* (52nd Aerospace Sciences Meeting, 2014).

[21] P. Spalart, *Trends in turbulence treatments,* in *Fluids 2000 Conference and Exhibit,* p. 2306.

[22] D. C. Wilcox, *Turbulence modeling for cfd. la canada, ca: Dcw industries,* Inc, November (2006).

[23] B. Aupoix, *Roughness corrections for the k–ω shear stress transport model: status and proposals,* Journal of Fluids Engineering **137** (2015).

[24] B. Aupoix, *Wall Roughness Modelling with k-w STT Model,* in *10th International ERCOFTAC Symposium on Engineering Turbulence Modelling and Measurements* (Marbella, Spain, 2014).

[25] I. H. Abbott and A. E. Von Doenhoff, *Theory of wing sections: including a summary of airfoil data* (Courier Corporation, 2012).

[26] J. Schepers, K. Boorsma, N. Sørensen, Voutsinas, G. Sieros, H. Rahimi, H. Heisselmann, E. Jost, T. Lutz, T. Maeder, A. Gonzalez, C. Ferreira, B. Stoevesandt, G. Barakos, N. Lampropoulos, A. Croce, and J. Madsen, *Final results from the EU project AVATAR: Aerodynamic modelling of 10 MW wind turbines,* Journal of Physics: Conference Series **1037**, 022013 (2018).

[27] D. Baldacchino, C. Ferreira, D. D. Tavernier, W. Timmer, and G. van Bussel, *Experimental parameter study for passive vortex generators on a 30% thick airfoil,* Wind Energy **21**, 745 (2018).

[28] AVATAR, *Avatar: Results and deliverables,* (2017), `http://eera-avatar.eu/publications-results-and-links/index.html`, Last accessed on 2020-02-24.

[29] C. M. Langel, R. Chow, O. F. Hurley, C. P. v. Dam, R. S. Ehrmann, E. B. White, and D. Maniaci, *Analysis of the impact of leading edge surface degradation on wind turbine performance,* (AIAA SciTech, 33rd Wind Energy Symposium, Florida, USA, 2015).

[30] W. Timmer and R. Van Rooij, *Summary of the delft university wind turbine dedicated airfoils,* J. Sol. Energy Eng. **125**, 488 (2003).

[31] J. E. Danberg and A. Sigal, *Analysis of turbulent boundary-layer over rough surfaces with application to projectile aerodynamics*, Tech. Rep. (Army Ballistic Research Lab Aberdeen Proving Ground MD, 1988).

[32] J. A. Van Rij, B. Belnap, and P. Ligrani, *Analysis and experiments on three-dimensional, irregular surface roughness,* J. Fluids Eng. **124**, 671 (2002).

[33] S. T. McClain, S. P. Collins, B. K. Hodge, and J. P. Bons, *The Importance of the Mean Elevation in Predicting Skin Friction for Flow Over Closely Packed Surface Roughness,* Journal of Fluids Engineering **128**, 579 (2005).

[34] K. A. Flack and M. P. Schultz, *Roughness effects on wall-bounded turbulent flows,* Physics of Fluids **26**, 101305 (2014).

[35] K. Flack, M. Schultz, and J. Barros, *Skin friction measurements of systematically-varied roughness: Probing the role of roughness amplitude and skewness,* Flow, Turbulence and Combustion **104**, 317 (2020).

[36] K. A. Flack and M. P. Schultz, *Review of hydraulic roughness scales in the fully rough regime,* Journal of Fluids Engineering **132** (2010).

[37] P. Forooghi, A. Stroh, F. Magagnato, S. Jakirlić, and B. Frohnapfel, *Toward a universal roughness correlation,* Journal of Fluids Engineering **139** (2017).

[38] M. Drela, *Two-dimensional transonic aerodynamic design and analysis using the Euler equations*, Tech. Rep. (Cambridge, Mass.: Gas Turbine Laboratory, Massachusetts Institute of Technology, 1986).

[39] H. Özdemir, *Interacting boundary layer methods and applications,* in *Handbook of Wind Energy Aerodynamics*, edited by B. Stoevesandt, G. Schepers, P. Fuglsang, and S. Yuping (Springer International Publishing, Cham, 2020) pp. 1–53.

6

7

Wind energy applications: MEXICO rotor

In this chapter, the flow past the MEXICO rotor is simulated with the new pressure based solver in SU2 using the SA and SST turbulence models on two different grids. The resulting pressure coefficient, normal and tangential loads, axial velocity traces and skin friction coefficient is compared against experimental data and numerical results from OpenFOAM and EllipSys. Good agreement is found between the results from SU2 and experimental data for pressure coefficient and loads. Finally, some of the numerical issues faced during this study are discussed.

7.1. Introduction

Modeling the flow past a wind turbine rotor has mostly been done using inviscid methods based on lifting line theory or lifting surface theory which use two dimensional airfoil data to predict the rotor performance. The airfoil data from such methods are usually computed from two dimensional methods like RFOIL. Two dimensional methods cannot accurately replicate three dimensional aerodynamics like rotational effects on the flow [1, 2]. CFD modeling of rotors was first reported by Sorenson [3], Duque[4] and Varela [5]. Since then, CFD has been used widely for rotor modeling. While the first attempts used the RANS turbulence models, the use of hybrid LES-RANS modeling is also gaining popularity [6]. In this chapter, the performance of the new pressure based solver for modeling turbine rotors is tested by simulating the rotor studied in the MEXICO (Model experiments in Controlled Conditions) project.

The MEXICO rotor is a three bladed wind turbine with a diameter of $4.5m$. The experimental campaigns were carried out in the open section ($9.5 \times 9.5m^2$) of the Large Scale Facility of the DNW (German-Netherlands) wind tunnel [7, 8] (see

(a) MEXICO experiment set up [7].
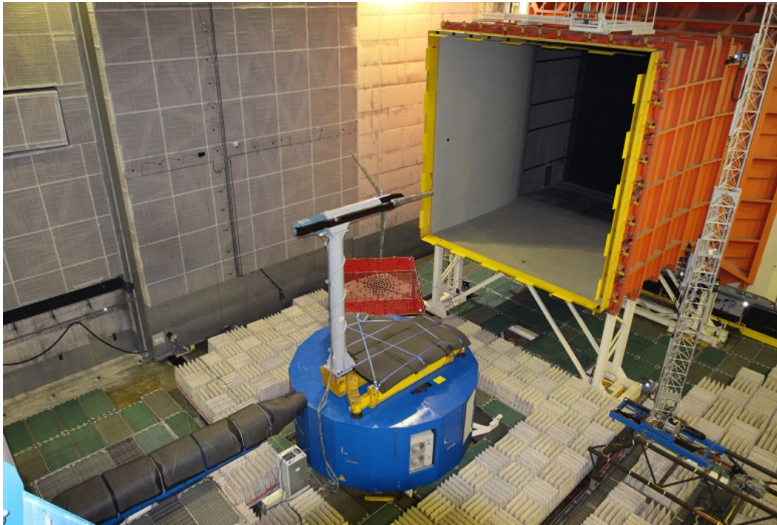
(b) New MEXICO experiment setup [8].

Figure 7.1: Setup of the rotor in the two MEXICO experimental campaigns.

figure 7.1). Measurements were made for different tip speed ratios under different operating conditions. In this chapter a tip speed ratio,

$$\lambda = \frac{\Omega R}{U_\infty}, \tag{7.1}$$

of $\lambda = 6.7$ is chosen. Here Ω is the rotational rate of the wind turbine blade expressed in rad/s and R is the radius of the rotor. The resulting tunnel velocity is $U_\infty = 15m/s$. This case corresponds to the design conditions of the rotor blade. The details of the turbine blade and the operating conditions considered for the numerical simulation is listed in table 7.1. The results from SU2 are compared to the data from

Number of blades, Z	3
Rotor diameter, $D(m)$	4.5
Pitch angle, θ (°)	-2.3
Yaw angle, Ψ (°)	0.0
Rotational speed, $n(rpm)$	424.5
Wind speed, $U_\infty(m/s)$	15
Tip speed ratio, λ	6.7

Table 7.1: MEXICO rotor details and flow conditions.

the second campaign and the results from numerical simulations using OpenFOAM and EllipSys. The results of EllipSys are obtained from Scheppers et al. [9] and the results from OpenFOAM were obtained from the simulations performed by a colleague [10].

7.2. Numerical set up

Flow past wind turbine rotors are generally unsteady and turbulent. The rotation of the turbine blade introduces additional complications. In order to simplify the computational effort, the simulation is carried out in a reference frame that rotates with the wind turbine rotor. The governing equations for simulations in a moving reference frame were given in section 3.5. In this case, the grid is rotating at a constant rotation rate, Ω, which is the rotational speed of the rotor blade. Additionally, since the three rotors are identical, instead of modeling all three rotors, a periodic simulation of only one rotor is considered.

The domain used for the simulation is shown in figure 7.2. In the streamwise direction, the domain extends from $-2.5D$ upstream of the rotor to $5.5D$ downstream of the turbine. In the radial direction, the domain consists of a cylinder of radius $1.5D$ around the hub. However, since periodic boundary conditions are used, only a $120°$ sector of the cylinder as shown in figure 7.2 is modeled.

Free stream boundary conditions are used on the streamwise boundaries and the top of the cylinder. A solid wall boundary condition is used on the rotor blade and hub. Periodic boundary conditions are used on the two faces of the $120°$ sectors. The reference frame is rotating at a constant rate of $\Omega = 44.51 rad/s$. However, the hub has no rotational motion and is handled via a boundary condition. The simulations are carried out using both the SA and k-ω SST turbulence models. The free stream velocity is set to $U_\infty = 15m/s$.

Two different grids are considered. The fine mesh has 160 nodes along the chord of the rotor and 388 nodes along the span of the rotor. A $y^+ < 1$ on the rotor surface is maintained around the rotor. The coarse mesh has 60 nodes along the
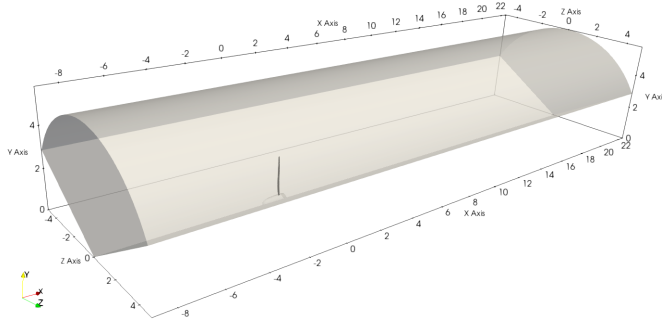
Figure 7.2: Domain of the MEXICO rotor case.

chord and 305 nodes along the span. A $y^+ < 1$ is maintained in the coarse grid as well. The height of the first cell is $4.0 \times 10^{-6}m$ in both meshes. A bounding box
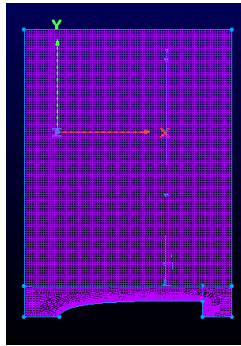

Figure 7.3: Bounding box around the rotor.

of height $1.75m \times 0.8m \times 2.5m$ is created around the rotor (figure 7.3). A hybrid meshing strategy (figure 7.4a) is used within this bounding box. The boundary layer mesh consists of hexahedral cells (figure 7.4b). A total of 54 layers of hexahedral cells were created in the fine mesh and 50 layers in the coarse mesh. Subsequently, tetrahedral cells and pyramid cells fill the bounding box. The faces of the bounding box are then extruded in the streamwise and radial directions using hexahedral cells only. The fine mesh contains 25.4×10^6 cells and the coarse mesh contains 14.9×10^6 cells. The commercial grid generation package Pointwise is used for meshing.

7.3. CFD Results

Preliminary results from the pressure based solver of the pressure coefficients compared to data from experiments and another CFD code EllipSys are shown below. Full convergence was difficult to achieve due to the complexity around the hub re-
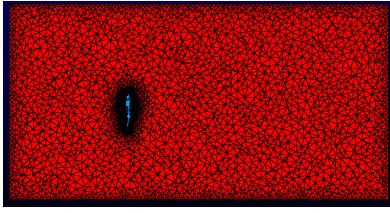
(a) Hybrid meshing around the rotor.
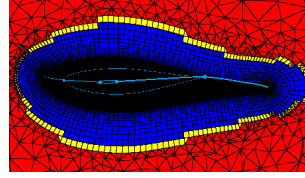
(b) Boundary layer mesh.

Figure 7.4: Meshing strategy around the rotor. Images from the fine grid.

gion. The OpenFOAM simulations were done on the same fine mesh that was used in SU2.

7.3.1. Streamlines

Flow streamlines on the surface on the pressure side of the rotor blade from the SA turbulence model is shown in figure 7.5a and the k-ω SST turbulence model in figure 7.5b. Flow separation is observed near the root of the rotor. The flow remains attached near the tip. The k-ω SST turbulence model predicts a slightly large separation compared to the SA turbulence model. The skin friction coefficient

(a) SA turbulence model.

(b) k-ω SST turbulence model.

Figure 7.5: Skin friction coefficient streamlines using line convolution integral method.

streamlines are generated using ParaView's Surface Line Integration Convolution (LIC) method.

7.3.2. Pressure coefficient

The pressure coefficients at different radial positions are presented below. The fine grid is denoted as "lvl1" and the coarse grid as "lvl2" in the following section. The

pressure coefficients are normalized using the local reference velocity at a radius r, U_{rlocal}, given by

$$C_p = \frac{p - p_\infty}{\frac{1}{2}\rho U_{rlocal}^2},$$

$$U_{rlocal} = \sqrt{U_\infty^2 + (\Omega r)^2}.$$

$r/R = 0.25$: Figure 7.6a shows the comparison of C_p from the fine and the coarse grids at a radial location of $r/R = 0.25$. Results from both the grids show similar behavior as the results from other numerical methods. The coarse grid results on the suction side are generally higher than the fine grid. Similar behavior is observed in the results from the SST turbulence model shown in figure 7.6b. The numerical results from all the codes differ significantly from the experimental data as there is massive flow separation in this region.
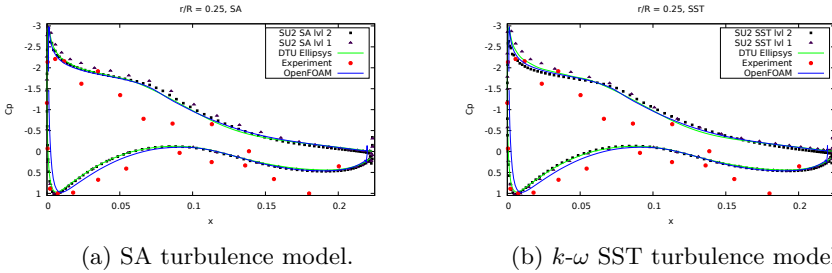

(a) SA turbulence model.   (b) k-ω SST turbulence model.

Figure 7.6: Pressure coefficient at $r/R = 0.25$ from the SA and the k-ω SST turbulence models.

$r/R = 0.35$: Figures 7.7a and 7.7b show the C_p at a radial location of $r/R = 0.35$. Results from both the fine and coarse grids using both turbulence models show the same behavior as the other numerical codes. There is an over prediction of the peak suction pressure in all the numerical methods. However, the results from the SST turbulence model matches the experimental data more closely near the suction peak compared to the SA model.

$r/R = 0.60$: Figures 7.8a and 7.8b show the C_p at a radial location of $r/R = 0.6$. There is a deviation near the suction peak from the experimental data in the results from the fine grid using both the SA and SST turbulence models. This mismatch is observed in other numerical results too. The results from the coarse grid using both the SA and SST turbulence models deviate more significantly than the fine mesh and does not follow the same trend as other numerical methods. However, all the numerical methods match the experimental data on the pressure side.

$r/R = 0.82$: Figures 7.9a and 7.9b show the C_p at a radial location of $r/R = 0.82$. In this case, all the numerical methods under predict the suction peak but match the
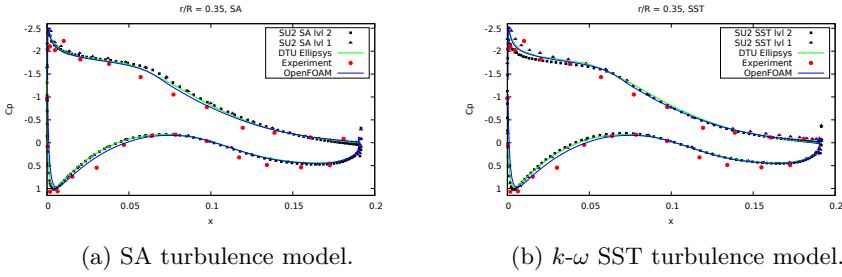
(a) SA turbulence model.

(b) k-ω SST turbulence model.

Figure 7.7: Pressure coefficient at $r/R = 0.35$ from the SA and the k-ω SST turbulence models.

(a) SA turbulence model.

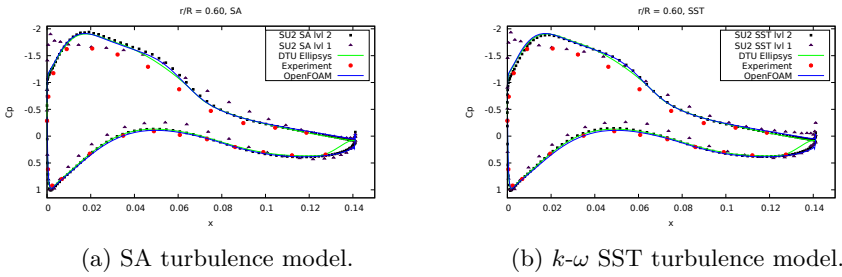(b) k-ω SST turbulence model.

Figure 7.8: Pressure coefficient at $r/R = 0.60$ from the SA and the k-ω SST turbulence models.

experimental data closely immediately after the suction peak. The results from the fine grid matches those from other numerical methods closely. The coarse grid over predicts the pressure values near the leading edge on both the suction and pressure side.
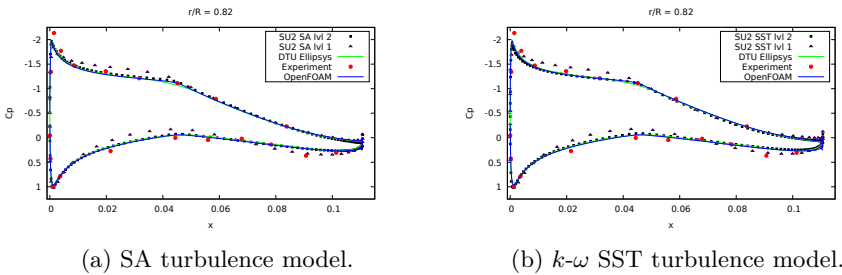

(a) SA turbulence model.

(b) k-ω SST turbulence model.

Figure 7.9: Pressure coefficient at $r/R = 0.82$ from the SA and the k-ω SST turbulence models.

$r/R = 0.92$: Figures 7.10a and 7.10b show the C_p at a radial location of $r/R = 0.92$. Once again, the suction peak is under predicted in all the numerical results compared to the experimental data. The results from the fine and coarse grid match the experimental results and other numerical results closely after the suction peak.
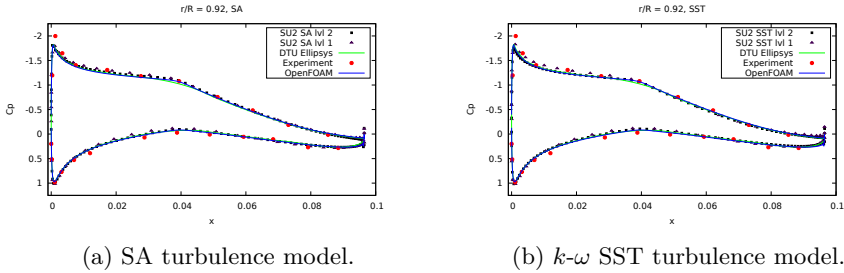
(a) SA turbulence model.   (b) k-ω SST turbulence model.

Figure 7.10: Pressure coefficient at $r/R = 0.92$ from the SA and the k-ω SST turbulence models.

7.3.3. Loads

The normal and tangential loads from the fine mesh using both the SA and k-ω SST turbulence models are compared to data from the experiments and results from the two other numerical tools - OpenFOAM and EllipSys. The normal and tangential forces are found by integrating the pressure and skin friction coefficients along the chord at the five radial locations listed earlier. The loads are the dimensionalized based on local velocity, density and free stream pressure. All the numerical results show similar trends in predicting the normal and tangential loads.

Normal loads   Figure 7.11a shows the normal loads computed using the different numerical methods compared to the experimental data. The results from the SA turbulence model appears to predict consistently higher loads than the SST turbulence model. A close agreement with experimental data is observed near the root for all the numerical results, but starts to deviate at higher radii. An under prediction of normal loads near the tip is observed. There is a dip in the experimental data at the $r/R = 0.6$ section which could be an error in the measurements.
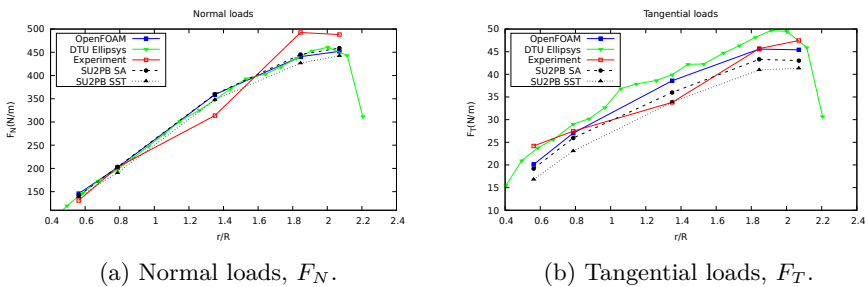

(a) Normal loads, F_N.   (b) Tangential loads, F_T.

Figure 7.11: Loads calculated using the numerical methods compared to experimental data.

Tangential loads   Figure 7.11b shows the tangential loads computed using the different numerical methods compared to the experimental data. Once again, the results from the SA turbulence model appears to predict consistently higher loads

than the SST turbulence model. However, the results from the SA turbulence model match the other numerical results. A relatively close agreement between experimental data and numerical results along the rotor blade is obtained, except at $r/R = 0.6$. Though the results from the SST turbulence model match the experimental data at this point, it is likely a coincidence since the trends are different.

7.3.4. Axial velocity traces

The axial velocity at two radial locations, $r/R = 0.61$ (inboard) and $r/R = 0.82$ (outboard), is compared to the experimental data and the numerical results from EllipSys. This is a preliminary result, especially for the SST turbulence model, since convergence issues were experienced which will be described later. The velocity in the inboard region is under predicted for both the SA and SST turbulence models. However, the results from the SA turbulence model matches closely with the experimental and other numerical results in the outboard region.
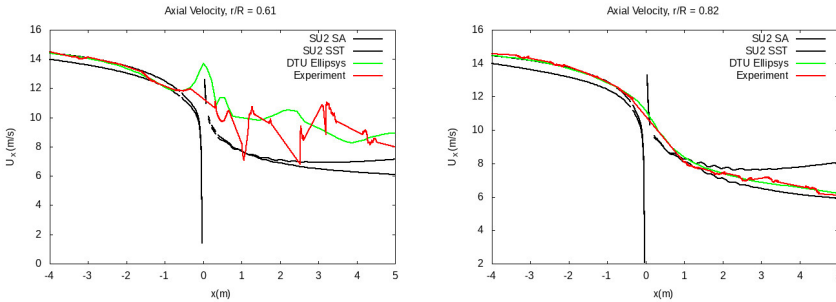
(a) Axial velocity trace at $r/R = 0.61$.   (b) Axial velocity trace at $r/R = 0.82$.

Figure 7.12: Axial velocity comparison in the induction and near wake region.

7.3.5. Skin friction coefficients

Figures 7.13a to 7.14b show the comparison of the skin friction coefficient between the SU2 results from the two turbulence models and OpenFOAM. While both the turbulence models predict the trend in the skin friction coefficient similar to the results from OpenFOAM, wiggles are observed near the leading edge in all cases.

7.4. Numerical issues

The numerical simulation was first initialized at a lower rotational rate, starting from $\Omega = 10 rad/s$, and increased gradually to $\Omega = 25 rad/s, 35 rad/s$ and $\Omega = 40 rad/s$ before reaching $\Omega = 44.51 rad/s$. Only an initialization was sought at intermediate rotational speeds, and thus the simulations were not run to reach a steady state at these intermediate values.

Two numerical issues were observed during the simulation. The major issue was the modeling of the hub. Since the hub was modeled as a solid non-rotating object in a rotating reference frame, convergence was hindered and at times not possible
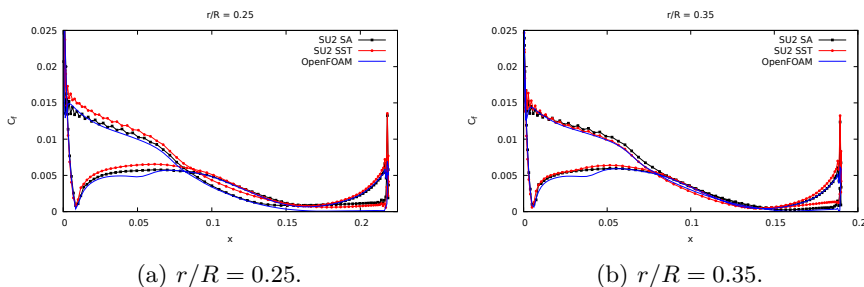
(a) $r/R = 0.25$. (b) $r/R = 0.35$.

Figure 7.13: Skin friction coefficient (C_f) comparison between results from SU2 and OpenFOAM.

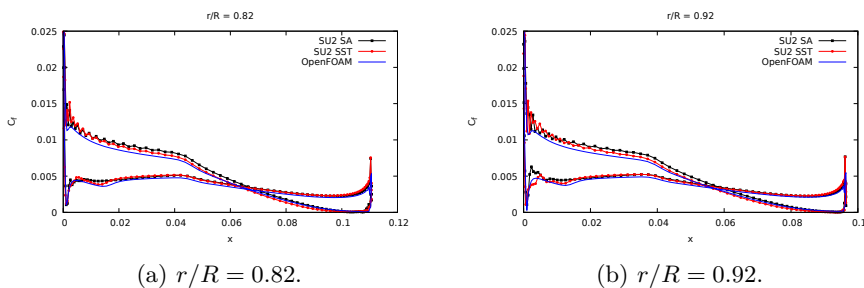(a) $r/R = 0.82$. (b) $r/R = 0.92$.

Figure 7.14: Skin friction coefficient (C_f) comparison between results from SU2 and OpenFOAM.

7

because of the nature of the cells near the hub. This was caused because of the hybrid meshing technique used to generate the grids. The hybrid meshing technique creates a region of cells that are heavily skewed near the hub which coupled with the fact that the hub is stationary in a rotating reference frame led to strong gradients in velocity.

One potential solution to this is to model the rotor only as the intention of performing these simulations is to study the performance of the rotor and the near wake region. Alternately, the hub can be treated as a slip wall to ease the convergence issues as was done in Zhang [11].

The second numerical issue is the wiggles observed in the skin friction coefficient plots. The wiggles are non physical but were observed even when using first order upwind schemes with no reconstruction of velocities. Similar behavior was observed when performing two dimensional simulations on certain airfoils as well. A deeper investigation is necessary to determine the cause of this anomaly.

7.5. Conclusion

The new pressure based incompressible solver has been used to simulate the flow past the MEXICO rotor. Two different grid resolutions were considered. Results from the fine grid using the SA and k-ω SST turbulence model match results from other numerical tools and are in agreement with experimental data. Discrepancies

in the pressure coefficient and loads observed between the results from the new pressure based solver and the experimental results are similar to those observed in other numerical tools as well.

The pressure based solver is capable of simulating flow past a wind turbine with reasonable accuracy. However, numerical issues still persist and improvements must be made to make the solver more robust.

References

[1] J. Sumner, C. S. Watters, and C. Masson, *Cfd in wind energy: the virtual, multiscale wind tunnel,* Energies **3**, 989 (2010).

[2] S. Schreck and M. Robinson, *Rotational augmentation of horizontal axis wind turbine blade aerodynamic response,* Wind Energy: An International Journal for Progress and Applications in Wind Power Conversion Technology **5**, 133 (2002).

[3] N. Sørensen and M. Hansen, *Rotor performance predictions using a navier-stokes method,* in *1998 ASME Wind Energy Symposium* (1998) p. 25.

[4] E. P. Duque, W. Johnson, C. van Dam, R. Cortes, and K. Yee, *Numerical predictions of wind turbine power and aerodynamic loads for the NREL phase II combined experiment rotor*, Tech. Rep. (NASA Moffett Field CA AMES Reserach, 2000).

[5] J. Varela and D. Bercebal, *CFD Calculations of the Flow Around a Wind Turbine Nacelle*, Tech. Rep. (Centro de Investigaciones Energeticas Medioambientales y Tecnologicas (CIEMAT), 1999).

[6] J. Thé and H. Yu, *A critical review on the simulations of wind turbine aerodynamics focusing on hybrid rans-les methods,* Energy **138**, 257 (2017).

[7] H. Snel, J. G. Schepers, and B. Montgomerie, *The MEXICO project (model experiments in controlled conditions): The database and first results of data processing and interpretation,* Journal of Physics: Conference Series **75**, 012014 (2007).

[8] K. Boorsma and J. Schepers, *New mexico experiment, preliminary overview with initial validation.*, Tech. Rep. ECN E 14-048 (Energy Research Center of the Netherlands, ECN, 2014).

[9] J. Schepers, T. Lutz, K. Boorsma, S. Gomez-Iradi, I. Herraez, L. Oggiano, H. Rahimi, P. Schaffarczyk, G. Pirrung, H. A. Madsen, *et al.*, *Final report of IEA wind task 29 Mexnext (Phase 3)*, Tech. Rep. (Energy Research Center for the Netherlands (ECN), 2018).

[10] M. Caboni, personal communication.

[11] Y. Zhang, *Wind turbine rotor aerodynamics: The IEA MEXICO rotor explained*, Ph.D. thesis, Delft University of Technology (2017).

7

Conclusion

This thesis consists of two parts. The first part of the thesis from chapters 1 to 4 present the development and validation of the new pressure based solver. The second part of the thesis from chapters 5 to 7 present three typical wind turbine aerodynamics problems that are solved using the newly developed solver.

Development and validation

A new pressure based solver has been implemented within the framework of SU2 with the intention of using it as a base for further research into wind turbine design and analysis. A finite volume discretization scheme is used and the new solver has been shown to be second order accurate in space and is capable of simulating a wide array of problems including steady, unsteady and rotating flow problems. The accuracy of the solver has been verified against analytical solutions for simplified flows. Validation of the solver has been done using widely used test cases like laminar and turbulent flow over a flat plate, backward facing step, turbulent flow over an airfoil and an airfoil undergoing pitching motion.

Wind turbine aerodynamics applications

In the second part of the thesis, three typical wind turbine aerodynamics applications are presented - flow past vortex generators, effect of leading edge erosion and flow past a rotor blade.

Vortex generator modeling

As a first step towards modeling vortex generators (VGs) in integral boundary layer equation based methods, the flow past a pair of VGs on a flat plate is simulated using the pressure based solver. Since, the integral boundary layer equations are $1D$ and the flow $3D$, a reduction in dimension is necessary. To this end, the difference between a $3D$ flow field and a two dimensional approximation is studied. The two dimensional flow field can then be used to derive the closure relations necessary for the IBL equations analogous to the laminar and turbulent boundary layers.

Effect of leading edge erosion

The roughness model has been validated against an empirical boundary layer profile for rough surfaces. Subsequently, the roughness model has been applied to flow over airfoils to analyze the impact of erosion on aerodynamic efficiency and turbulent boundary layer quantities like displacement thickness, momentum thickness and skin friction. Additionally, methods to convert observed roughness into equivalent sand grain roughness are presented.

Rotor simulations using CFD

CFD simulations of full rotors are becoming increasingly common and in order test the capability of the new pressure based solver, the flow past the rotor blade used in the New MEXICO experimental campaign has been simulated. Design conditions with incoming wind speed of 15 m/s and a tip speed ratio of 6.7 is used. The resulting pressure distributions and loads matched the experimental results and other numerical data closely. Fully converged results for velocity were not yet obtained.

Future work

While the solver has been successfully used for a variety of applications, many improvements are still necessary. Some of the improvements are

- Improve robustness for skewed cells - Skewed cells are commonly observed in industrial problems (like rotors, thick airfoils) and cause many issues with convergence. In order to obtain a good solution for such problems, the handling of the skewed cells especially in the Poisson solver must be improved.

- Multigrid solver for the Poisson equation to improve convergence speed - The solution of the pressure Poisson problem is a crucial part of the solution algorithm for pressure based flow solvers. Multigrid methods are commonly used to solve Poisson equations and greatly improve solution time. This can be crucial for unsteady problems where the Poisson problem has to be solved multiple times within a time step.

- Relaxation of the solution in the initial stages of the iterative process - Since the momentum and pressure Poisson equations are solved in a decoupled manner, large changes in the solution can lead to divergence especially during the initialization phase. Large changes in the solution can occur either as a result of the problem specification, bad initial conditions or the use of large CFL numbers. Regardless of the cause, the robustness of the solver must be improved if the solver will have to be used for state of the art problems.

- Implement a low dissipation convective discretization scheme in order to perform Large Eddy Simulations (LES) - The second order upwind scheme, widely used for RANS simulations, introduces a relatively large amount of artificial dissipation to the solution. While this artificial dissipation can stabilize RANS problems, it is very inaccurate for performing Large Eddy Simulations. A low dissipation scheme like central differencing or higher order upwind schemes will need to be implemented in order to use the pressure based solver for LES.

Curriculum Vitae

Akshay Koodly Ravishankara, born on 31 October 1992, hails from the city of Bengaluru in the state of Karnataka, India. After finishing his secondary school at Kendriya Vidyalaya, Hebbal in Bengaluru, he started his under graduate degree (B. Tech) in Mechanical Engineering at the National Institute of Technology Karnataka (NITK), Surathkal, India in 2010. With a keen interest in aerodynamics, he focused on fluid dynamics during his bachelors degree. His first foray into computational fluid dynamics came during an internship on simulating the flow through a compressor at TU Bergakademie Freiberg in Germany. With a mixture of passion for aerodynamics from his early days and a desire to further explore the field of computational fluid dynamics, he chose to pursue his Masters degree (M.S) in Aeronautics and Astronautics Engineering from Purdue University, West Lafayette, USA in 2014. He majored in aerodynamics with a minor in propulsion during his Masters studies. In 2015 he got an opportunity to pursue an internship at the wind energy unit of ECN in The Netherlands on computational aerodynamics. With a desire to work on renewable energy technology while continuing to study aerodynamics, he decided to focus on wind turbine aerodynamics for his PhD project at the University of Twente in The Netherlands in 2017. This project was supported by the energy transition department of TNO (previously ECN) where he is presently employed as a Junior Scientist.

List of Publications

Peer reviewed Publications

- **A. Koodly Ravishankara**, H. Özdemir and E.T.A. van der Weide, (2021). Analysis of leading edge erosion effects on turbulent flow over airfoils. Renewable energy, 172, 765-779.

- **A. Koodly Ravishankara**, I. Bakhmet and H. Özdemir. Estimation of roughness effects on wind turbine blades with vortex generators. Journal of Physics: Conference Series. Vol. 1618. No. 5. IOP Publishing, 2020.

Conference Proceedings

- **A. Koodly Ravishankara**, H. Özdemir, and E.T.A van der Weide (2020). Implementation of a pressure based incompressible flow solver in SU2 for wind turbine applications. In AIAA Scitech 2020 Forum (p. 0992).

- **A. Koodly Ravishankara**, H. Özdemir and A. Franco (2019). Towards a vortex generator model for integral boundary layer methods. January, 1, 16.