A PARABOLIZED NAVIER-STOKES METHOD FOR WIND FARM APPLICATIONS

By

Anshul Mittal

Lafayette K. Taylor
Professor (Retired) of Computational Engineering
Committee Chair

Kidambi Sreenivas
Research Professor of Computational Engineering
Committee Co-Chair

W. Roger Briley
Professor Emeritus of Computational Engineering
Committee Member

William K. Anderson
Professor (Retired) of Computational Engineering
Committee Member

D. Stephen Nichols, III
Associate Research Professor of Aerospace Engineering (Auburn University)
Committee Member

John “Matt” Matthews
Associate Professor of Mathematics
Committee Member

A PARABOLIZED NAVIER-STOKES METHOD FOR WIND FARM APPLICATIONS

By

Anshul Mittal

A Dissertation Submitted to the Faculty of the University of Tennessee at Chattanooga in Partial Fulfillment of the Requirements of the Degree of Doctor of Philosophy in Computational Engineering

The University of Tennessee at Chattanooga
Chattanooga, Tennessee

December 2015

ABSTRACT

Fluid flow simulations play an important role in the wind industry. With the development of large wind farms, flow simulations through an entire wind farm are becoming a necessity. These are required for designing the layout of new wind farms in the development stage and for forecasting power production from the existing ones for operational purposes. Conventional Navier-Stokes simulations (commonly referred to as CFD simulations) are computationally very expensive since they require a supercomputer with runtimes of several weeks.

A Parabolized Navier-Stokes (PNS) method is developed and implemented in this study. The developed PNS method requires less stringent approximations as compared to the existing parabolic methods and incorporates more physics. A wind turbine model is developed and coupled with the PNS method for simulating wind turbines and entire wind farms. The wind turbine model is adapted for spatial marching and is based on the Actuator Line rotor model and Blade Element Momentum theory. The developed PNS method is validated and verified using several test cases and a wind farm simulation on a desktop computer has a runtime of only several hours.

ACKNOWLEDGMENTS

The research presented here was supported by the National Science Foundation (NSF) through a grant (CBET # 1236124) with Dr. Ram Gupta and Dr. Gregory Rorrer as technical monitors. Support was also provided by THEC. This support is gratefully acknowledged.

Images were created and data extractions carried out using Fieldview, licenses for which were provided by Intelligent Light through their University Partners Program.

LIST OF PUBLICATIONS RESULTING FROM THIS RESEARCH

1. **Mittal, A.**, Taylor, L.K., Sreenivas, K., and Arabshahi, A., "Investigation of Two Analytical Wake Models Using Data From Wind Farms," Proceedings of the ASME 2011 International Mechanical Engineering Congress & Exposition, Denver, Co., November 11-17, 2011.

2. **Mittal, A.**, and Taylor, L.K., "Optimization of Large Wind Farms Using a Genetic Algorithm," Proceedings of the ASME 2012 International Mechanical Engineering Congress & Exposition, Houston, TX, November 9-15, 2012.

3. Sreenivas, K., Hilbert, C.B., **Mittal, A.**, Hereth, L., and Taylor, L.K., "High-Fidelity Computational Simulation of the Wake Characteristics of a Model Wind Turbine," 31st AIAA Applied Aerodynamics Conference, San Diego, CA, June 2013, AIAA Paper 2013-2416.

4. **Mittal, A.**, Sreenivas, K., and Taylor, L.K., "Exploration of Modal Decomposition Techniques for Wind Turbines," AIAA-2014-1398, SciTech 2014, National Harbor, MD, January 2014.

5. **Mittal, A.**, Sreenivas, K., Taylor, L.K., Hereth, L., Hilbert, C.B., and Hyams, D.G., "Investigation of Rotor Models for Wind Turbine Simulations," 32nd AIAA Applied Aerodynamics Conference, Atlanta, GA, June 2014, AIAA Paper 2014-2280.

6. Sreenivas, K., **Mittal, A.**, Hereth, L., Taylor, L.K., and Hilbert, C.B., "High-Fidelity Computational Simulation of the Interaction between Tandem Wind Turbines," 32nd AIAA Applied Aerodynamics Conference, June 2014, AIAA Paper 2014-2278.

7. **Mittal, A.**, Sreenivas, K., Taylor, L.K., and Hereth, L., "Improvements to the Actuator Line Modeling for Wind Turbines," AIAA Paper 2015-0216, 33rd Wind Energy Symposium, AIAA SciTech 2015.

8. **Mittal, A.**, Briley, W.R., Taylor, L.K., and Sreenivas, K., "A Parabolic Method without Pressure Approximations for Wind Turbines," AIAA Paper 2015-0728, 33rd Wind Energy Symposium, AIAA SciTech 2015.

9. Sreenivas, K., **Mittal, A.**, Hereth, L., and Taylor, L.K., "Computational Simulation of the Interaction between Tandem Wind Turbines with Offset," AIAA Paper 2015-0224, 33rd Wind Energy Symposium, AIAA SciTech 2015.

10. **Mittal, A.**, Briley, W.R., Taylor, L.K., and Sreenivas, K., "A Parabolic Method without Pressure Approximations for a Wind Farm," EWEA Offshore 2015, Copenhagen: EWEA, 2015.

11. **Mittal, A.**, Taylor, L.K., Sreenivas, K., Briley, W.R., and Nichols, D.S., "Extension of a Parabolic Method without Pressure Approximations for Wind Turbines in ABL Flows," AIAA Paper 2015-3391, 33rd AIAA Applied Aerodynamics Conference, Dallas, TX, June 2015.

12. **Mittal, A.**, Sreenivas, K., Briley, W.R., and Taylor, L.K., "A Parabolic Method for Accurate and Efficient Wind Farm Simulation," AIAA Paper 2015-2268, 33rd AIAA Applied Aerodynamics Conference, Dallas, TX, June 2015.

13. Hassan, W.E., Sreenivas, K., **Mittal, A.**, Taylor, L.K., and Hereth, L. "Blade Resolved Simulation for a Wind Farm," AIAA Paper 2015-2269, 33rd AIAA Applied Aerodynamics Conference, Dallas, TX, June 2015.

14. **Mittal, A.**, Sreenivas, K., Taylor, L.K., Hereth, L., and Hilbert, C.B. "Blade-Resolved Simulations of a Model Wind Turbine: Effect of Temporal Convergence", accepted at Wind Energy.

15. **Mittal, A.**, Briley, W.R., Sreenivas, K., and Taylor, L.K. "A Parabolic Velocity-Decomposition Method for Wind Turbines", submitted to Journal of Computational Physics.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

NOMENCLATURE

ϵ,	Projection width, distance over which forces are distributed
c,	Chord length
Δr,	Grid spacing
TSR,	Tip Speed Ratio
ϵ_c ϵ_c,	Projection width along the chord
ϵ_s,	Projection width along the span of the blade
r_c,	Distance of a grid point from the actuator point along the chord
r_s,	Distance of a grid point from the actuator point along the span of the blade
$\vec{F}$,	Aerodynamic force computed using FAST
$\vec{f}$,	Projected force on a control volume
Δb,	Distance between two adjacent actuator points
C_P, C_T,	Power and thrust coefficients
PNS,	Parabolized Navier-Stokes
AL,	Actuator Line
L,	Reference length, length scale in the *x*-direction
D,	Turbine diameter
u,	*x*-direction velocity
ν,	Kinematic viscosity
ρ,	Reference density
Re,	Reynolds number, uL/ν
p,	Pressure
v, *w*,	Velocities in *y* and *z* directions (transverse plane)
x_D,	Distance from the turbine nondimensionalized by the turbine diameter
r,	Distance between polar grid point and computational grid point
$\vec{b}$,	Body force per unit volume: $b_x\hat{i} + b_y\hat{j} + b_z\hat{k}$
h,	Distance between boundary node and first point off the boundary

CHAPTER 1

INTRODUCTION

Wind energy has become a promising renewable energy resource over the decades. Electricity generation from wind turbines helps in reducing greenhouse gas emissions, fossil fuel consumption, and fresh water usage. The total installed wind capacity worldwide was 369 GW at the end of 2014 which is 3 % of the global electricity production.[1] In the European Union, wind energy provides more than 10 % of the electricity.[1] The industry has matured and progressed to a level where large wind farms are being developed. As an example, the largest offshore wind farm is the London Array wind farm with a capacity of 630 MW.

Computer simulations are extensively used in the wind industry. They are used for designing blades, generators, wind farm layouts (micrositing), analyzing and optimizing electrical grids, and power forecasting. Numerical simulations for estimating power produced by a wind farm are required for both micrositing and forecasting purposes and must account for wake losses in the wind farm. Due to the proximity of the turbines in a farm (typically 5-7 rotor diameters), wakes lead to reduced power production and increased dynamic loads on the blades.

Different types of models exist to simulate the flow through a wind farm. These models account for varying degrees of physical modeling assumptions and their computational cost varies greatly. On one end of the spectrum are the analytical wake models which attempt to predict the velocity and other flow variables in the wake. They typically use a set of explicit equations or analytical expressions and their runtime ranges from milliseconds to a few seconds (see Table 1.1). Some prominent models were developed by Jensen[2,3], Ishihara[4] and Frandsen[5]. These models have limited applicability as all the

relevant physics is not included and they might have to be tuned to each specific site. At the other end of the spectrum are the CFD models, which solve the governing equations (Navier-Stokes) to compute the flow field. These models are available with varying degree of approximations ranging from Large Eddy Simulation (LES) to Reynolds Averaged Navier-Stokes (RANS) formulation. These are commonly referred to as high-fidelity models and require a supercomputer/cluster with runtimes ranging from days to several weeks (see Table 1.1). The user has the option of choosing how to model the wind turbine rotor. An obvious choice is to include the detailed blade geometry in the simulation. However, that increases the mesh size and computational cost considerably. Over the years, several rotor models have been developed to reduce this cost. These include Actuator Disk[6] (AD) and Actuator Line[7] (AL) models.

Somewhere in the middle of the spectrum are the parabolic models, also known as the Field models. These solve a simplified version of the Navier-Stokes equation. Models by Ainslie[8,9] and Crespo et al.[10] are popular in the wind industry. The equation set is parabolized to make it suitable for spatial marching. These models make fewer approximations and thus, are more accurate than the analytical wake models while being significantly cheaper than the RANS/LES simulations. Runtimes for these models range from minutes to several hours (see Table 1.1).

Table 1.1 Comparison of models in terms of fidelity and cost

Simulation Model	Fidelity of the model	Computational cost (runtime for a wind farm)
RANS/LES	High	Several days to few weeks
Field Models (Parabolic)	Medium	Minutes to hours
Analytical Models	Low	Few seconds

A. Motivation

This study is motivated by the need for higher fidelity but low cost simulation methods applicable to wind turbines and wind farms. This work includes an investigation of existing methods as well as the development of new methods with increased accuracy and low computational cost for wind farm simulations. The focus is on the Actuator Line (AL) rotor model and Parabolized Navier-Stokes (PNS) method.

Simulations of large wind farms result in large computational grids due to the increase in the number of blades (typically three per turbine), the need to discretize the blade geometry and resolve the turbulent boundary layer. Actuator Disc (AD) and Actuator Line (AL) models offer an alternative to discretizing the blade geometry (typically referred to as Fully Resolved method). By replacing the turbine blades by a set of points in the form of a disc (AD) or lines (AL), there is no need to resolve the blade geometry. Although discretizing the blade geometry is more accurate, avoiding that discretization leads to ease in grid generation and reduction in mesh size thereby reducing the overall time to solution. The AL model has become popular in the wind community in the past decade. An existing AL model is investigated and implemented in the in-house code, *Tenasi*. Several researchers have reported various issues with the state-of-the-art in AL models. These issues are investigated and potential improvements are developed as part of this study.

A new PNS model is proposed that mitigates some of the issues with the existing parabolic models. An increase in fidelity is also expected due to a higher fidelity wind turbine model.

B. Actuator Line Model

In the AL model each blade is modeled separately by a number of line segments and the aerodynamic forces are imposed for each segment. An open source code, FAST[11], developed by NREL, is commonly used for computing the aerodynamic forces by using the 2D airfoil data and applying 3D

corrections. A similar model is an Actuator Disk (AD) model in which the rotor is modeled as a disk and the forces are imposed over the entire disk.

Over the years, the Actuator Line (AL) method has become a commonly used simulation technique. Numerous studies[12–14] have been published utilizing the AL method. The AL method was originally developed by Sørensen and Shen[7]. One of the issues highlighted by many researchers is the solution dependence on the grid resolution and projection width, which is the distance/region over which the forces are distributed. Troldborg et al.[13] investigated the wake using different rotor modeling methods (fully resolved, AD and AL). For uniform turbulent inflow, all three methods gave similar results and suggested that the projection width should be at least twice the grid spacing. Martinez et al.[12] found that the predicted power varies significantly when the ratio of projection width to the grid spacing is increased beyond the specified minimum of 2.

Shives and Crawford[15] suggested choosing the projection width based on the chord of the blade which is more physical. It was proposed that the projection width be a quarter of the chord length. In a similar work by Jha et al.[16], an elliptic chord distribution was found for the blade and the projection width was determined using that chord distribution, where the value of the constant depended on the grid resolution. For a user interested in carrying out a simulation, the number of actuator points is still a free parameter. Although the focus of the Jha et al. study[16] was to improve the prediction of loads on the blade, the power is over-estimated by 15 % for the NREL Phase VI rotor and this trend carries over for the NREL 5-MW wind turbine where the over-estimation is about 8 %. Moreover, another issue with the above-described approaches arises while projecting forces. As one moves along the span of the blade, the distance over which the force is projected (projection width) is only a function of the chord length and not the size of the blade/actuator element. This results in forces being distributed over a distance that is not physically tied to the size of the element.

In this study, the AL model is implemented in the in-house code, *Tenasi*, and the above-discussed issues are investigated. Several potential solutions for improving the AL model are proposed and investigated.

C. Parabolic Navier-Stokes Model

Before the advent of supercomputers, much research was done to approximate or simplify the Navier-Stokes equations to reduce runtimes, especially for steady-flow problems. A popular approach was to introduce approximations that result in a parabolic equation set that can be solved by spatial marching methods. Spatial marching approaches require significantly less computer resources, and a desktop computer is often sufficient rather than a supercomputer/cluster. The runtime is reduced because steady-state solutions require only a single marching pass from the inflow boundary to the outflow boundary.

Boundary-layer equations are parabolic and their deduction by Prandtl was an important advancement in fluid dynamics.[17] Interest in three-dimensional parabolic methods still exists in various fields, mainly motivated by the high computational cost of the Navier-Stokes simulations. The wind industry is such an example because supercomputer simulations of flow through a wind turbine or a wind farm are very expensive, and runtimes range from several days for a single turbine to weeks for a wind farm[14,18–20].

One of the first models for application in wind energy was developed by Sforza et al.[21,22] It is a linearized momentum equation with constant convective velocity and constant eddy viscosity. The two most popular parabolic models were developed by Ainslie[8,9] and Crespo et al.[10] The Ainslie model assumes an axisymmetric wake and neglects the streamwise pressure gradients and the swirl in the wake. The turbulent shear stresses are modeled using an eddy-viscosity closure scheme. These simplifying assumptions are not suitable for the near-wake region, and the solutions are initiated in the

far-wake region by specifying the velocity deficit (wake characterization) at a location two or more rotor diameters downstream. This model does not account for ground effects or an atmospheric boundary layer.

The model developed by Crespo et al.[10] is called the UPMWAKE model. The wind turbine is assumed to be immersed in a non-uniform surface layer described by analytical expressions that account for the effects of the atmospheric stability and the surface roughness. The Navier-Stokes equations are made parabolic for the far-wake region by neglecting streamwise viscous-diffusion and axial pressure-gradient terms. The system of equations is closed using a k-ε turbulence model and is solved numerically using the SIMPLE algorithm. This model also excludes the near-wake region and is initiated by specifying the velocity deficit created by the rotor at a location two or more rotor diameters downstream. Atmospheric stability effects are modeled using a Boussinesq approximation. The UPMWAKE methodology has been improved by Schepers et al.[23] and the resulting model is known as WAKEFARM. It also models the far-wake region and uses a database of experimental velocity profiles to initialize the far-wake solution at a cross-section located 2.25 rotor diameters downstream of the wind turbine.

All of these parabolic models omit streamwise pressure gradients to obtain a parabolic approximation, and they are applied to the far-wake region by initializing the solution at a certain distance downstream of the wind turbine. The required starting velocity profiles are thus important and must be chosen to model near-wake flow effects adequately. Schepers et al.[23] used a free-vortex method to calculate an approximation for streamwise pressure gradients in the far-wake region. This approximation is introduced by prescribing source terms to represent the pressure-gradient field within the framework of the WAKEFARM model.

In the present work, a new high-fidelity parabolic model combined with a state-of-the-art Actuator Line (AL) wind turbine model is developed and adapted to spatial marching. The high-fidelity parabolic model employs order-of-magnitude estimates from boundary-layer theory to obtain parabolic equations applicable to steady three-dimensional primary/secondary flow problems. The effect of wind turbines

and their upstream influence is modeled as source terms incorporating time-averaged aerodynamic forces from the actuator line model.

CHAPTER 2

ACTUATOR LINE MODEL

In this chapter the Actuator Line model is described along with some of the issues encountered by users. Potential solutions of some of these issues are investigated and results are discussed here. In an AL model, each blade is discretized by line segments and the center of each line segment is called an actuator or blade point. Aerodynamic forces, computed using the airfoil data, are utilized to model the effect of the blade in a CFD simulation. An AL model offers higher fidelity than an AD model because the blades are modeled distinctly and the AL method captures individual blade effects like root and tip vortices.

In the present work, aerodynamic forces required for the use of the AL model are computed using FAST[11], an open-source code developed by NREL. FAST (Fatigue, Aerodynamics, Structures and Turbulence) is a comprehensive aeroelastic simulator capable of predicting both the extreme and fatigue loads of two- and three-bladed horizontal-axis wind turbines. All the features except the aerodynamic force computation are switched off using the appropriate input flags/parameters. The aerodynamic forces are computed using look-up tables which provide steady-state lift and drag forces for the blade cross-section based on the airfoil, relative wind speed and angle of attack. The lift and drag coefficients are supplied by the user through the input files and are utilized by FAST for computing the aerodynamic forces at each actuator point. FAST performs a time-accurate simulation based on the time step size selected by the user and computes the position of the blades and the forces on them at each time step. Since FAST was not designed to be coupled with a CFD code, a modified version of FAST, also from NREL, available in the SOWFA[24] suite is utilized.

A. AL Model, Issues and Potential Improvements

The SimCenter's in-house CFD code, *Tenasi*, is coupled with the modified version of FAST for simulating wind turbines. The aerodynamics forces computed by FAST are transferred along with the position of all the actuator points at each time step to the CFD code. FAST requires a velocity field to compute the aerodynamics forces. For an accurate two-way coupling, the velocity field should be supplied from the CFD code and thus, at each time step, velocity field is also transferred. The modified version of FAST code has specific function calls for (1) supplying velocities at the actuator points and (2) requesting position and forces at the actuator points. These function calls are utilized through a Python script to couple the in-house code *Tenasi* which provides Python hooks to implement various functionalities including modifying the residual (for incorporating body force terms). There are several potential issues with an AL model when compared against a fully-resolved Navier-Stokes model. These are discussed below with potential solutions.

1. Hub and Tower Effects in an AL Model

The standard AL model only describes a model for the blades of a wind turbine. There is no prescribed way of including a hub, nacelle and tower in a standard AL simulation. Most, if not all, AL simulations do not model the hub, nacelle and tower in any way. However, these components/entities, even though they do not contribute to torque production, do affect the wake of the turbine and thus, would affect the performance of any downstream turbines operating in the wake. In the absence of a hub and nacelle, the flow escapes through the center of the rotor and this effect shows up as a peak in the velocity profile at the wake center. The effect of a tower is more complicated since the wake of the turbine interacts with the tower and alters the wake velocity and turbulence characteristics.

For these reasons, it is important to include the effect of hub, nacelle and tower in wind turbine simulations. There are potentially three ways of including these and they vary in geometry representation and associated flow physics.

1. An accurate way is to represent the actual hub, nacelle and tower geometry and generate a grid that resolves them.[25] The mesh should be sufficiently fine near the geometry to resolve the flow near the viscous surfaces. Thus, in this approach an AL model is used for the blades but the hub, nacelle and tower are treated as in a fully resolved rotor simulation. The advantage is that the geometry is represented accurately and the flow physics is captured with higher fidelity.

2. Another option is a minor variation of the previous approach. The geometry may be included but the grid can be coarse enough to not resolve the flow near the viscous surfaces. This offers the advantage of reduced grid size since the boundary layer is not resolved. In this case, the geometry is still represented accurately but the flow physics would not be captured with higher fidelity. The bulk effects of these entities would still be picked-up.

3. The third option is to have an AL type model for these entities (hub, nacelle and tower). Such a model has been developed by the researchers[26] at NREL. Like the AL model for blades, forces are imposed on the flow to model and simulate the effect of these entities. It should be noted that in such a model neither the geometry is represented nor is the associated flow physics captured or resolved with high fidelity. But this method might be sufficient for incorporating gross flow effects.

In terms of the computational cost, the first method is the most expensive among the three discussed above. The second method is not investigated in this study but the cost of the second and the third method should be comparable if not similar. The first method is investigated in the present work. For the first method, the user has a choice of either using wall functions or integrating up to the wall to resolve the viscous sub-layer of the boundary layer. In all the work presented in this study, all the sub-layers of the boundary layer including the viscous sub-layer are resolved and wall functions are not utilized.

2. Determination of Velocity Field

The lift and drag coefficients are determined from the 2D airfoil data (corrected for 3D effects) by utilizing the angle of attack. The lift and drag forces are computed using the lift and drag coefficients and the freestream velocity magnitude. In the standard AL model, the velocity field is determined from the CFD code at the location of the actuator point[12]. This is used to compute the angle of attack and the velocity magnitude which are required for computing the lift and drag forces.

An issue with this approach is that as the actuator line (blade) rotates through the CFD mesh, the distance between the nearest CFD node and the actuator point changes. Considering a grid with an equal spacing of a in all the three directions, the distance between an actuator point and its nearest node can vary from 0 to $\sqrt{3}a/2$. The velocity returned to FAST is not affected by this distance. It can be hypothesized that the effect of not accounting for this distance manifests itself as fluctuations or noise in the simulation leading to increased standard deviation of, for example, power and thrust values. Please note that the distance between the nodes and the actuator point is taken into account when distributing the forces (discussed in the next sub-section).

In this study, three approaches were investigated to alleviate this problem and compared to the existing (baseline) approach of using the velocity from the nearest node. They are (1) Extrapolation of the velocity components from the nearest node using available gradient information (2) Averaging of velocity vector (no extrapolation) from the nearest neighbors (as opposed to just the nearest node) and (3) Averaging of the extrapolated velocity vector (using gradient information) from the nearest neighbors.

3. Force Distribution

The aerodynamic forces obtained from FAST need to be distributed to obtain a continuous distribution over the CFD mesh in the region where the blade would have been. If the forces are not

distributed and only applied to the nearest CFD node, the resulting flowfield would be non-physical (due to discontinuous force distribution) and might even lead to numerical instability.

In the standard AL model, a 3D Gaussian distribution is used for projecting the forces onto the CFD mesh. Equation (2.1) gives the expression for projecting the force where *r* is the distance between an actuator point and a CFD node. Here $\vec{F}$ is the aerodynamic force computed using FAST and $\vec{f}$ is the projected force on a control volume.

$$\vec{f} = \frac{\vec{F}}{\epsilon^3 \pi^{3/2}} \exp\left[-\left(r/\epsilon \right)^2 \right] \tag{2.1}$$

The distance is nondimensionalized by the projection width, ϵ, and determines the extent to which the force is distributed. The AL model was used with a constant value of the projection width by the researchers initially[7,12,27]. This results in the forces being distributed too far, especially near the blade tip. An improvement was developed where the projection width would be a function of the chord length. Shives and Crawford[15] suggested the projection width be a function of the local chord whereas Jha et al.[16] based it on an elliptical chord (found/determined separately). This improved the AL model (and the results) by distributing the forces over a region which is similar to the blade surface area but resulted in another issue.

When the force is distributed based on a projection width which is determined by the local chord length, the region over which the force gets distributed changes as the chord changes (typically decreases along the span from root to tip of a wind turbine blade). The width of actuator elements is typically constant for AL simulations and the force projection does not account for the width of the actuator element. This results in forces being distributed over larger distance along the span (affecting larger number of actuator elements) near the root of the blade whereas the forces are distributed over a smaller distance along the span and affect fewer actuator elements near the tip of the blade. To remedy this situation, we proposed an improvement to the standard AL model where the forces are distributed by utilizing two different projection widths, one determined by the local chord length and

one determined by the size of the actuator element. The proposed method of projecting forces is given by Eq. (2.2) where r_s is the distance between an actuator point and a CFD node along the span of the blade and r_c is the distance perpendicular to the span in the plane of the chord. Thus, $r^2 = r_c^2 + r_s^2$. It should be noted that if the two projection widths are the same then the expression collapses to the one given above by Eq. (2.1).

$$\vec{f} = \frac{\vec{F}}{\epsilon_s \epsilon_c^2 \pi^{3/2}} \exp\left[-\left(r_c / \epsilon_c \right)^2 - \left(r_s / \epsilon_s \right)^2 \right] \tag{2.2}$$

B. Parameters for AL Simulations

For conducting an AL simulation, several parameters related to the AL model need to be determined. These include the CFD grid spacing in axial (Δx) and transverse ($\Delta y, \Delta z$) directions, projection width, number of actuator points and the time step size for the CFD simulation. We start with the blade information and find a representative chord (for standard AL model) or smallest chord, typically near the tip of the blade (for proposed AL model). The projection width is computed based on the selected chord by utilizing $\epsilon = 0.25c$ [15]. For stability reasons, researchers[12,27] found that $\epsilon \geq 2\Delta x$. Using the equality, spacing in the axial direction (Δx) is computed. There is no clear rule or method of determining the grid spacing in the transverse directions. It has been suggested to try different spacing values and pick the one which gives accurate results. The transverse spacing can be chosen to be of similar magnitude as the axial spacing. In this study, the transverse spacing is chosen to be either equal to or close to axial spacing. A structured grid is utilized in the rotor region for AL simulations.

For an AL simulation, discretization of the blade is also a parameter. The spacing between the actuator points, Δb, is suggested[12,28] to be related to the transverse spacing by $\Delta b \approx 0.75\Delta y$. Using this expression, spacing between the actuator points can be determined and thus, the number of actuator points needed to represent the blade. A restriction for AL simulations is the limit on the time step size for CFD simulations. The time step size should be small enough that no actuator point should jump a

cell/control volume in one time step. The time step size can be computed based on the tip speed and the transverse grid spacing values used. The recipe for computing the parameters for an AL simulation is as follows:

- Select the chord (refer to the text above)
- Compute projection width, $\epsilon = 0.25c$
- Compute axial spacing, $\Delta x = 0.5\epsilon$
- Compute transverse spacing, $\Delta y = \Delta z \approx \Delta x$
- Calculate blade discretization, $\Delta b \approx 0.75\Delta y$
- Compute time step size based on transverse spacing

C. Test case for Validation: BT1

The geometry[29] shown in Figure 2.1 is a realistic model of a modern three bladed wind turbine. The rotor diameter is 0.894 m and it sits on top of a stepped tower consisting of 4 cylinders of different diameters. The nacelle is also circular with a diameter of 90 mm. It has an almost semispherical hub cover at the front and back. The model was tested in a wind tunnel that has a test section which is 11.15 m long. The height varies from 1.801 m to 1.851 m to achieve zero pressure gradients at 14 m/s flow speed. The width of the test section is 2.71 m. The turbine has 3 blades and they are made up of 14 % thick NREL S826 airfoils. The inflow velocity was maintained at 10 m/s while the turbine RPM was varied in order to achieve the desired tip speed ratio. Experimental data (mean velocities as well as Reynolds stresses) were obtained at 1, 3 and 5 diameters downstream of the wind turbine for tip speed ratios of 3, 6 and 10.

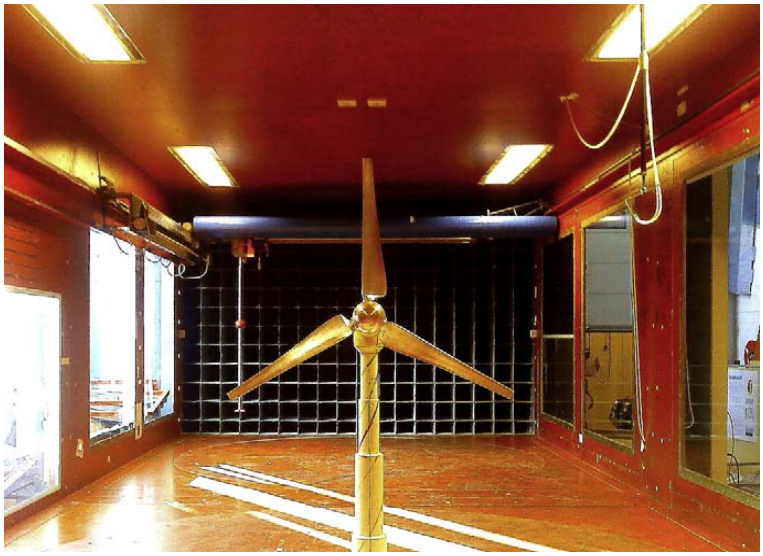


Figure 2.1 BT1 Experimental setup

D. Results

The proposed solutions for the three issues discussed earlier in this chapter are investigated here and the results are presented. BT1 test case is selected for testing since the experimental measurements and the airfoil data is available.

1. Hub, Nacelle and Tower Effects

The effect of hub, nacelle and tower in an AL simulation is evaluated and presented here. These entities are represented accurately and a fine enough grid is generated to resolve the geometry and the flow physics around them. This corresponds to the first method as discussed earlier. Results for the BT1 test case are presented here comparing the experimental data with the fully resolved RANS simulations (includes hub, nacelle and tower) and two different AL simulations. Wind tunnel walls are included in all the simulations. Both the AL simulations include the tower but only one includes the hub and nacelle entities. Spalart-Allmaras turbulence model is used for turbulence closure for all the simulations.

The grid for the fully resolved simulations had 32.2×10^6 nodes. A uniformly resolved grid block was included downstream of the rotor to ensure reasonable wake resolution. Time step equivalent to 360 steps per rotor revolution was used and 10 Newton sub-iterations were utilized for time accuracy.

For AL simulations, the recipe for determining all the parameters is followed by selecting a representative chord of 0.036201 m (at 68.8 % span). A structured grid block was included in place of the rotor with a spacing of 5 mm in all directions. The final grid with the hub, nacelle and tower had 17.9×10^6 nodes whereas the grid without the hub and nacelle had 17.1×10^6 nodes. A time step of 0.0001 seconds for the case of TSR = 6 was used. Blades are discretized using 105 actuator points and a projection width of 8 mm is used. It should be noted that if the recipe for computing AL parameters is followed, the projection width would be 9 mm. However, at the time these simulations were conducted, there was a bug in the code which resulted in forces being 1.2 times less (due to the density of air). Since, this was not known at that time, the projection width was adjusted. This shows the potential pitfall of the AL method where any amount of power/thrust can be obtained by varying the parameters. The results presented in the next two sections were obtained after correcting the bug and the projection width used is 9 mm for those cases. Please refer to Sreenivas et al.[30] and Mittal et al.[31] for details of the fully resolved simulations and Mittal et al.[25] for AL simulations details.

Power and thrust coefficients for various tip-speed-ratios are presented in Figure 2.2. The results include AL simulations (with and without the hub and nacelle), fully resolved simulations (CFD) and the experimental data.

As can be seen from the figure, the overall trends in thrust and power are predicted well. The agreement is worst for TSR = 10. One of the reasons could be that the solution has not been run long enough because of the large computational cost. Due to the setup required for the AL simulations, it is not possible to utilize local time-stepping to convect the flow features efficiently. This very likely caused the wake to not propagate downstream enough. Local time-stepping is a convergence acceleration technique in which the time step is allowed to vary among the control volumes depending upon the

local flow physics and the size of the control volume. A constant CFL value is typically utilized for local time-stepping. Another plausible cause for discrepancy is that the airfoil data used was for Reynolds number of 100,000 only. The Reynolds number based on the chord at tip for the design case (TSR = 6) is 103,600. It goes up to a value of 172, 667 for TSR of 10. This change is the Reynolds number and any flow separation on the blade not accounted by the 3D corrections to the airfoil data could be a cause of the discrepancy.

Note that the thrust and power coefficient presented are computed/reported by FAST and as a result, do not account for the thrust force experienced by the hub and nacelle (where they are included). The magnitudes of the thrust and torque coefficients from this study are similar to that obtained by participants in the BT1 workshop[29].

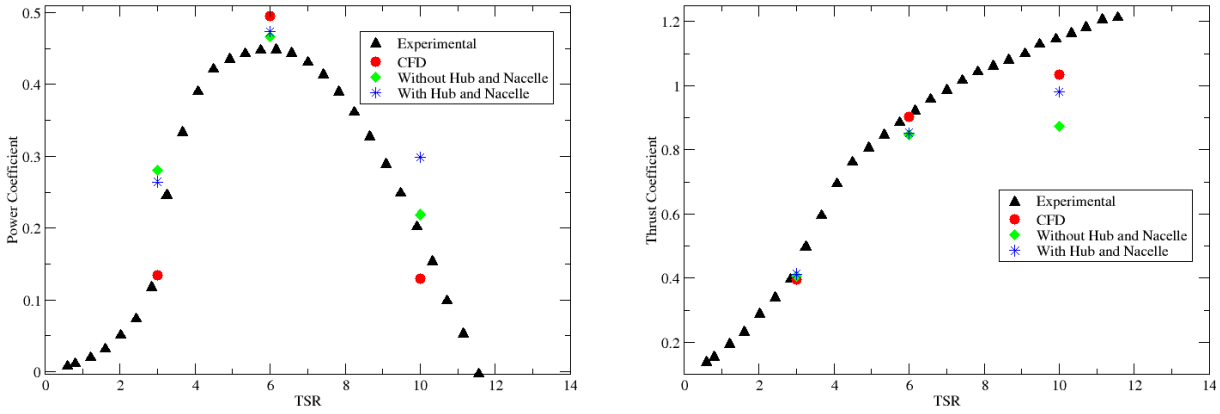


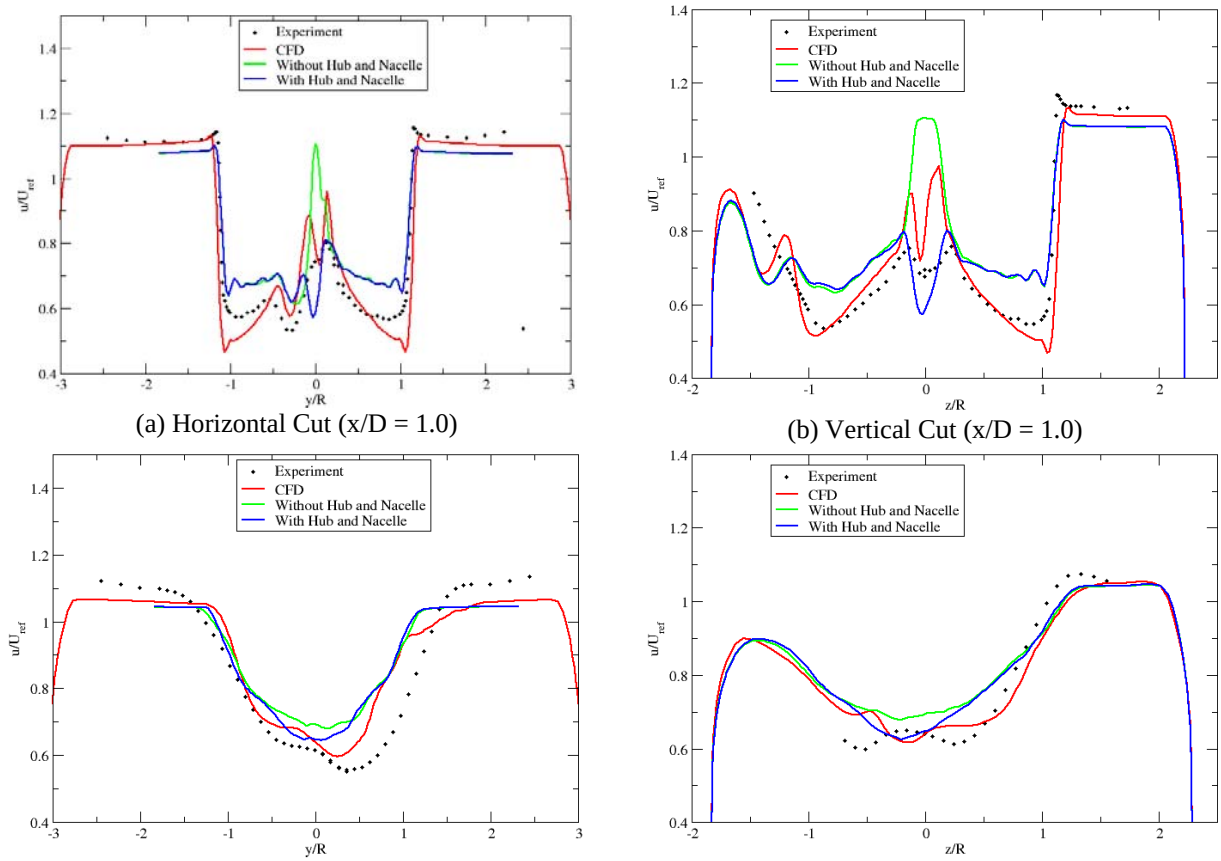
Figure 2.2 Comparison of power and thrust coefficients

As part of the BT1 experiment, wake data was collected for the BT1 turbine at three different locations. These locations corresponded to x/D = 1, 3, and 5, corresponding to near-wake, intermediate-wake and far-wake locations. The typical spacing for wind turbines in the field is about 5D. The experimental wake surveys included horizontal and vertical sweeps providing a three-dimensional view of the wake behind the turbine.

Time-averaged (over five rotor revolutions) computed results (AL simulations with and without hub and nacelle) are compared to experimental data and CFD results (fully resolved) for the first and last stations downstream of the wind turbine. In each figure, the horizontal and vertical wake surveys, grouped by the axial station, are shown next to each other. The quantity plotted in these figures is the wake velocity. Also, for brevity, results are presented for the design TSR only. Complete set of results are available in Mittal et al.[25]

The comparison shown here (Figure 2.3) is for the design TSR. The advantage of having the hub and the nacelle is clearly evident in the wake data. In the absence of the hub and the nacelle, flow escapes through the center of the turbine and shows up as a jet in the wake (Figure 2.3(a) and (b)). The effect is visible up to 5D downstream. The asymmetry present in the experimental data at y/R = 0.2 or so (Figure 2.3(a)) is captured by the CFD simulations as well as the AL simulations. This asymmetry appears to be a consequence of the interaction between the turbine and the tower, since a turbine without the tower typically produces a symmetric velocity profile downstream. This asymmetry was not captured by most participants in the BT1 workshop as most did not include the tower in their computations.

The velocity predicted by the AL simulations is higher than the experimental data and/or CFD results at all the stations. A likely cause is the bug in the code which resulted in less force being applied. Moreover, since the agreement was good for the power coefficient, it is believed that the solution needs to be run for more time steps. Most of the peaks and wake characteristics are captured to some extent in the AL simulations. The wake spreading rate is captured quite well in the horizontal direction, while it is reasonable in the vertical direction.



(a) Horizontal Cut (x/D = 1.0)

(b) Vertical Cut (x/D = 1.0)

(c) Horizontal Cut (x/D = 5.0)

(d) Vertical Cut (x/D = 5.0)

Figure 2.3 Comparison of velocity profiles downstream of the wind turbine (TSR = 6)

2. Determination of Velocity Field

The different approaches for determining the velocity for the AL model are investigated here. The standard AL model is utilized for projecting the forces onto the CFD grid. The AL parameters and the grid spacing values are computed as discussed in the previous section. A representative chord of 0.036201 m at a span of 68.8 % (radius of 0.3075 m) is selected for computing the AL parameters. Values of all the parameters and grid spacing values are given below in mm except for the number of actuator points.

$$\begin{aligned} \Delta x &= 4.5 \\ \Delta b &= 3.75 \\ \epsilon &= 9.0 \\ \Delta y &= \Delta z = 5.0 \\ N_b &= 105 \end{aligned} \tag{2.3}$$

The grid for the simulations had a total of 17.8 million nodes. The time step was selected to be 0.0001 seconds (equivalent to 468 steps per rotor revolution) to ensure that an actuator point (near the tip) does not jump a cell moving from a time step to the next. The simulation was carried out for a total of 5000 time steps. The time-averaged values of the power coefficient (C_P) and thrust coefficient (C_T) and their standard deviations are presented in the Table 2.1. The results are time averaged over the last 2000 time steps (over 4 rotor revolutions). Note that the values of the thrust coefficient are expected to be lower than the experimental data because the drag force on the hub is not included in the results presented in the table.

The table gives the results for a total of six cases. The first is the baseline where the velocity from the nearest node is used and the second is when gradient information is utilized to extrapolate the velocity from the nearest node. The other four cases are for when the velocity from the nearest neighbors (two different sets of neighbors) is averaged with and without utilizing the gradient information. Immediate neighbors of the nearest node would be another 6 nodes in ±*x*, ±*y* and ±*z* directions. This would give the first set of nearest neighbor nodes. Going one more level, the next set of nearest neighbors would give another 14 nodes, making the total number of nearest neighbor nodes to be 21. From the results in the table, it is seen that both the power and thrust coefficient values decrease compared to the baseline case. The exception, for power coefficient, being the case where velocity is averaged from the 21 nearest neighbors and for thrust coefficient, being both the cases when the velocity is averaged (no use of gradients).

Table 2.1 Power and thrust coefficients for various approaches to determine input velocity to FAST

Approach	C_P	Standard deviation	Coefficient of Variance	C_T	Standard deviation	Coefficient of Variance
Experimental	0.4480	-	-	0.9100	-	-
Velocity from the nearest node	0.4604	0.03995	0.08677	0.8440	0.05743	0.06804
Extrapolated (using gradients) velocity from the nearest node	0.4379	0.03531	0.08063	0.8331	0.05232	0.06280
Velocity averaged from the 7 nearest neighbors	0.4497	0.03455	0.07683	0.8499	0.05055	0.05948
Velocity averaged from the 21 nearest neighbors	0.4767	0.03233	0.06782	0.8643	0.04529	0.05240
Extrapolated (using gradients) velocity averaged from the 7 nearest neighbors	0.4204	0.03330	0.07921	0.8154	0.05014	0.06149
Extrapolated (using gradients) velocity averaged from the 21 nearest neighbors	0.4179	0.03446	0.08246	0.8072	0.05207	0.06451

But a more significant observation is that the standard deviation is reduced for all the cases compared to the baseline case. However, since the mean values of the power and thrust coefficients have gone down, reduction in the standard deviation may not show the true picture. On comparing the coefficient of variation (ratio of standard deviation and mean value), a nondimensional standard

deviation, it is found that the coefficient of variation is also reduced for all the cases compared to the baseline case (both for power and thrust coefficients).

The results show that the two approaches (averaging velocity and utilizing gradient information) work by reducing the standard deviation in both the power and thrust coefficients. It was hypothesized that the fluctuation/noise in the power and thrust coefficients was due to neglecting the varying distance between the CFD node and the actuator point. However, it is surprising and still not clear as to when both these approaches are combined (averaging the velocities extrapolated using the gradients), the results do not show the expected trends. The predicted power and thrust coefficients are considerably lower than the experimental as well as other simulation approaches. One plausible reason could be that the gradient information is not accurate to be utilized to extrapolate to large distances (more than the nearest neighbor). The case where velocities from 21 nearest neighbors are averaged also has large error and could be because some of the 21 nodes are quite far from the actuator point where velocities are higher. This leads to higher averaged velocity resulting in higher power and thrust values. For further details, refer to Mittal et al.[32]

3. Force Distribution

In this section, the standard and the proposed models are compared. For all the simulations conducted, the projection width, for both the standard and the proposed models, was computed as a function of the local blade properties. For the BT1 case, the minimum chord of 0.025926 m is at the tip of the blade. For stability, spacing in the axial direction should be smaller than half of the projection width. Thus, the axial spacing, Δx , is 3.2 mm. Spacing of 5 mm was used for both the *y* and *z* directions. A total of six simulations were conducted, three with the original AL model and three with the proposed AL model. The blade discretization (number of actuator points) was varied for the three simulations for both the models. The number of actuator points on the blade was 27, 53 and 105. The projection width

for the original model was calculated as a function of the local chord, $\epsilon = 0.25c$. For the proposed model, chord based projection width was calculated in the same way as for the original model, $\epsilon_c = 0.25c$. The projection width in the span direction was set to twice the width of the actuator element.

The grid for the simulations had a total of 18.3 million nodes. The time step was selected to be 0.0001 seconds to ensure that an actuator point (near the tip) does not jump a cell moving from a time step to the next. The simulation was carried out for a total of 9362 time steps (20 rotor revolutions). The experimentally measured values of the power and thrust coefficients are 0.448 and 0.91 respectively. Note that the values of the thrust coefficient are expected to be lower than the experimental data because the drag force on the hub is not included in the results presented in the table.

Table 2.2 Results comparing the original and the proposed AL models

Number of actuator points	Power coefficient		Thrust coefficient	
	Proposed AL model	Original AL model	Proposed AL model	Original AL model
27	0.2681	0.2447	0.6710	0.6330
53	0.3440	0.3382	0.7636	0.7566
105	0.4403	0.4381	0.8623	0.8616

Power and thrust coefficients are compared across different blade discretizations and for the two AL models in Table 2.2. The results are time averaged over the last 9 rotor revolutions. The power and thrust coefficients predicted by the proposed AL model are always higher than the predictions by the original model. Moreover, as the width of the actuator element increases, the values of the power and thrust coefficients reduce significantly but the proposed AL model predictions are better than the

original AL model. For example, power coefficient varies from 0.2681 to 0.4403 for the proposed model. This shows the huge errors in the computational results and highlights the issue of grid dependent solutions, in this case, the ratio of the actuator element width to the grid spacing. Detailed information is available in Mittal et al.[32]

The objective behind the proposed AL model was to reduce the grid sensitivity but based on the results presented above, it is clear that the grid dependence is very strong and must be investigated further. The takeaway from this section is that to conduct an AL simulation, the grid should be designed keeping $\Delta x \leq 0.5\epsilon_c$ and $\Delta y = \Delta z \sim 1.3\overline{3}\Delta b$ as guides. The proposed model seems to perform better (gives slightly more accurate results) than the original AL model.

The choice of the constant for computing the projection width along the span using the width of the actuator element needs further attention. For the results presented, the constant was 2.0. The effect of this parameter has not been evaluated and should be investigated.

E. Summary and Lessons Learned

In this chapter, several aspects of AL modeling were discussed and potential improvements for many issues were investigated. The hub, nacelle and tower are important for wake characteristics and should be modeled when there is interest in the wakes of wind turbines. The high fidelity approach of modeling these entities is probably the most expensive and a comparison of the three approaches in terms of fidelity and computational cost would benefit the wind energy community.

Grid/AL parameter dependent solutions are an issue with AL simulations. This is to be expected as this is due to the decoupling of the blades from the flow. An attempt was made here to better understand how to select these parameters/grid spacing values and to synthesize all the information pertaining to this but this issue is not completely resolved. In other words, it is still not possible to have a completely deterministic way of computing all the parameters/grid spacing values for a simulation.

The time step restriction for the AL simulations takes away some of the benefit of reduced grid size. However, many researchers use an explicit method for time advancement due to the restrictive time step size. This works great for cases where the viscous spacing near the wall is absent since the control volumes near a viscous wall would require a very small time step. Implicit methods are ideal for such scenarios. Local time-stepping is typically utilized in CFD simulations to accelerate the convergence and convect the flow features quickly. However, due to the decoupling in AL simulations, local time-stepping cannot be utilized in an AL simulation which takes away some of the benefits of the reduced grid size.

CHAPTER 3

DEVELOPMENT OF PARABOLIZED NAVIER-STOKES MODEL FOR WIND TURBINES

In this chapter, development of the Parabolized Navier-Stokes (PNS) model is discussed in detail. The theory and the literature review for this model are described. This is followed by the derivation of the governing equations. The assumptions and approximations made to the Navier-Stokes equations are also discussed. Since the flow through wind turbines is primarily incompressible, the incompressible form of the Navier-Stokes equations is utilized and the developed PNS model is applicable for incompressible flows.

A. Background and Theory of the PNS Model

A new PNS model applicable to wind turbines is developed here based on the three-dimensional viscous primary/secondary flow approximation developed by Briley et al.[33,34] This approximation employs a vector decomposition of the secondary flow velocity. This is done to isolate a small velocity vector analogous to the surface-normal velocity component of boundary-layer theory and separate it from the large rotational component associated with streamwise vorticity. Neglecting this small component in the transverse momentum equations produces a parabolic set of equations without any approximations for the pressure-gradient terms, allowing the pressure field to be solved as a dependent variable.

The present model is formulated to allow the solution to be initialized well upstream of a turbine and to continue through the turbine and its near-wake and far-wake regions. This is accomplished by treating the flow field as a steady flow in which the turbine is modeled by specified source terms that

alter the velocity, energy and vorticity of the incident wind flow. The turbine source-term model incorporates time-averaged aerodynamic forces predicted by a BEM (Blade Element Momentum) model, FAST[11], developed at NREL. To account for the upstream flow effects, the time-averaged forces are distributed to simulate the slowing down of flow before the turbine and thus making this model applicable throughout the entire flow field, both upstream and downstream of the turbine. This parabolized model thereby avoids the need for far-wake starting conditions that model the turbine and near-wake region.

The three-dimensional primary/secondary flow approximation is based on the concept of a primary flow direction (straight or curved) whose velocity component does not reverse direction or encounter blunt surfaces with stagnation points. Examples include internal flows in smoothly curved ducts and external flows past smooth surfaces that have no flow separation. In the case of a wind farm, the prevailing wind defines the primary-flow direction, and each turbine is modeled using time-averaged source terms that account for alterations of the velocity, energy and vorticity of the incident wind flow caused by the turbine.

To obtain a parabolized Navier-Stokes model, approximations have to be made to both the viscous and inviscid terms of the Navier-Stokes equations. The present viscous approximation neglects streamwise diffusion in a manner similar to that of previous and existing models, but the inviscid terms are approximated differently. The eddy-viscosity model of Ainslie and the UPMWAKE and WAKEFARM models all prescribe the streamwise pressure gradient as a known source term, whether zero when neglected or non-zero when computed with the free-vortex method. In the present model, the primary-flow momentum equation is unaltered, and instead the inviscid convection terms in the transverse momentum equations are modified following a vector decomposition of the secondary-flow velocity vector into rotational and irrotational component vectors. Specifically, the equations are parabolized by neglecting the irrotational secondary-velocity components in the transverse momentum equations that govern streamwise vorticity, based on order-of-magnitude estimates analogous to those of boundary-

layer theory. The irrotational secondary-velocity vector is thereby determined by the continuity equation, and the rotational secondary-velocity vector is determined by the streamwise vorticity.

This primary/secondary flow analysis has been assessed and validated for flow in ducts and 90-degree bends by comparison with both experimental measurements and Navier-Stokes solutions[33,34], as well as in turbofan-exhaust lobed-mixer geometries.[35] This methodology has been shown to be very accurate when applied to flow problems in which the underlying assumptions of high Reynolds number, identifiable primary-flow direction, and no occurrence of reversed primary flow are satisfied. In the case of wind turbine flows, parabolized Navier-Stokes approaches cannot model blade stagnation points or separation, and consequently the turbines are modeled using a time-averaged actuator line method analogous to an actuator disc method.

B. Governing Equations

The nondimensional form of the continuity and the Navier-Stokes equations for steady and incompressible flow is given in Eq. (3.1). The dimensional reference quantities are *L* (length scale in the primary flow direction), *u* (velocity in the primary flow direction), ρu^2 (pressure), and ν (kinematic viscosity). The Reynolds number ($Re = uL/\nu$) is assumed to be large, and the primary flow direction is the *x* direction. ρ is the reference density and $\vec{b}$ is the body force per unit volume.

$$\begin{aligned} \nabla \cdot \vec{U} &= 0 \\ \vec{U} \cdot \nabla \vec{U} + \nabla p &= \frac{1}{Re}\left(\nabla^2 \vec{U}\right) + \vec{b} \end{aligned} \tag{3.1}$$

The velocity vector $\vec{U} = \vec{U}_p + \vec{V}_s$ is written as the sum of primary and secondary flow velocities such that $\vec{U}_p = u\hat{i}$ and $\vec{V}_s = v\hat{j} + w\hat{k}$. A surface gradient operator is defined for the transverse *y-z* plane as $\nabla_s(\cdot) = \hat{j}\,\partial(\cdot)/\partial y + \hat{k}\,\partial(\cdot)/\partial z$. A Helmholtz vector decomposition is introduced for the secondary-flow velocity such that

$$\vec{V}_s = \nabla_s \phi + \nabla_s \times \left(\hat{i} \psi \right) = \vec{V}_\phi + \vec{V}_\psi$$

$$\vec{V}_\phi = \hat{j} v_\phi + \hat{k} w_\phi = \hat{j} \frac{\partial \phi}{\partial y} + \hat{k} \frac{\partial \phi}{\partial z} \tag{3.2}$$

$$\vec{V}_\psi = \hat{j} v_\psi + \hat{k} w_\psi = \hat{j} \frac{\partial \psi}{\partial z} - \hat{k} \frac{\partial \psi}{\partial y}$$

Substituting the secondary velocities from Eq. (3.2) reduces the continuity equation to the following without any approximations.

$$\frac{\partial u}{\partial x} + \frac{\partial^2 \phi}{\partial y^2} + \frac{\partial^2 \phi}{\partial z^2} = 0$$

In the momentum equations, the primary/secondary flow approximations assume that $\left| \vec{V}_\phi \right| \ll 1$ and that surface shear layers are thin, so that the following order-of-magnitude estimates are appropriate for high Reynolds number:

- Throughout the flow field:

  $u,\ p,\ \partial(\cdot)/\partial x,\ v_\psi,\ w_\psi = O(1)$
  $v_\phi,\ w_\phi = O\left(Re_x^{-1/2} \right)$

- In rotational inviscid regions: $\partial(\cdot)/\partial y,\ \partial(\cdot)/\partial z = O(1)$

- In thin high-shear regions near solid surfaces whose surface-normal unit vector is $\vec{n}$:

  $$\vec{n} \cdot \nabla = \frac{\partial(\cdot)}{\partial n} = O\left(Re_x^{1/2} \right), \quad \nabla - \frac{\partial(\cdot)}{\partial n} \cdot \vec{n} = O(1)$$

Note that these scalings can be reduced to those of three-dimensional boundary-layer theory in which either *y* or *z* is the surface normal coordinate. Based on these scalings, the momentum equations can be written as in Eq. (3.3). Any viscous term containing $\partial/\partial x$ has been neglected based on the scalings. The streamwise vorticity is given by $\Omega_x = \hat{i} \cdot \left(\nabla \times \vec{U} \right) = -\nabla_s^2 \psi$. The continuity equation remains unchanged.

$$\vec{U}\cdot\nabla u+\frac{\partial p}{\partial x}=\frac{1}{Re}\left(\frac{\partial^2 u}{\partial y^2}+\frac{\partial^2 u}{\partial z^2}\right)+b_x$$

$$\vec{U}\cdot\nabla v_\psi+\frac{\partial p}{\partial y}=\frac{1}{Re}\left(-\frac{\partial \Omega_x}{\partial z}\right)+b_y \tag{3.3}$$

$$\vec{U}\cdot\nabla w_\psi+\frac{\partial p}{\partial z}=\frac{1}{Re}\left(\frac{\partial \Omega_x}{\partial y}\right)+b_z$$

To use spatial marching methods, the equation set should be non-elliptic in the spatial marching direction. The characteristic analysis of these equations was carried out by Briley and McDonald[33], where they showed that both the inviscid and the viscous terms are non-elliptic and well posed for spatial-marching solution. The source terms do not affect the characteristic analysis. Since solving this set of equations is difficult due to the fact that there is no equation for pressure, these equations are reformulated. This reformulation is carried out by replacing the two transverse momentum equations by their surface divergence and surface curl, and some subsequent manipulation. The reformulated equations governing the five dependent variables $\Omega_x, \psi, p, u, \phi$ are given below.

$$\begin{gathered}
\vec{U}\cdot\nabla\Omega_x+NL=\frac{1}{Re}\left(\frac{\partial^2\Omega_x}{\partial y^2}+\frac{\partial^2\Omega_x}{\partial z^2}\right)+\frac{\partial b_z}{\partial y}-\frac{\partial b_y}{\partial z}\\
\left(\frac{\partial^2\psi}{\partial y^2}+\frac{\partial^2\psi}{\partial z^2}\right)=-\Omega_x\\
\left(\frac{\partial^2 p}{\partial y^2}+\frac{\partial^2 p}{\partial z^2}\right)=-\frac{\partial}{\partial y}\left(\vec{U}\cdot\nabla\right)v_\psi-\frac{\partial}{\partial z}\left(\vec{U}\cdot\nabla\right)w_\psi+\frac{\partial b_y}{\partial y}+\frac{\partial b_z}{\partial z}\\
\vec{U}\cdot\nabla u+\frac{\partial p}{\partial x}=\frac{1}{Re}\left(\frac{\partial^2 u}{\partial y^2}+\frac{\partial^2 u}{\partial z^2}\right)+b_x\\
\left(\frac{\partial^2\phi}{\partial y^2}+\frac{\partial^2\phi}{\partial z^2}\right)=-\frac{\partial u}{\partial x}
\end{gathered} \tag{3.4}$$

where

$$NL=\left(\frac{\partial u}{\partial y}\frac{\partial w_\psi}{\partial x}+\frac{\partial v}{\partial y}\frac{\partial w_\psi}{\partial y}+\frac{\partial w}{\partial y}\frac{\partial w_\psi}{\partial z}\right)-\left(\frac{\partial u}{\partial z}\frac{\partial v_\psi}{\partial x}+\frac{\partial v}{\partial z}\frac{\partial v_\psi}{\partial y}+\frac{\partial w}{\partial z}\frac{\partial v_\psi}{\partial z}\right)$$

The equation set (Eq. (3.4)) gives the PNS equations for the mean flow and would be solved for carrying out simulations presented in this work.

C. Solution Methodology

The system of equations given in Eq. (3.4) is solved using a sequentially block-decoupled, semi-implicit algorithm. An efficient semi-implicit algorithm is constructed by modifying the system of equations written as a fully-implicit backward-difference semi-discrete approximation. Let $\left(\cdot\right)^{n}$ and $\left(\cdot\right)^{n+1}$ denote explicit and implicit levels of the marching discretization. The nonlinear convective operator is linearized as $\vec{U}^{n}\cdot\nabla\left(\cdot\right)^{n+1}$, where $\vec{U}^{n}=\vec{U}_{p}^{n}+\vec{V}_{s}^{n}$. The second modification is an explicit evaluation of the nonlinear terms NL^{n} in the vorticity equation using values known from a previous step in the solution sequence. The result is the following semi-discrete system:

$$
\begin{aligned}
\vec{U}^{n}\cdot\nabla\Omega_{x}^{n+1}-\frac{1}{Re}\left(\frac{\partial^{2}\Omega_{x}^{n+1}}{\partial y^{2}}+\frac{\partial^{2}\Omega_{x}^{n+1}}{\partial z^{2}}\right)&=\frac{\partial b_{z}^{n+1}}{\partial y}-\frac{\partial b_{y}^{n+1}}{\partial z}-NL^{n}\\
\left(\frac{\partial^{2}\psi^{n+1}}{\partial y^{2}}+\frac{\partial^{2}\psi^{n+1}}{\partial z^{2}}\right)+\Omega_{x}^{n+1}&=0\\
\left(\frac{\partial^{2}p^{n+1}}{\partial y^{2}}+\frac{\partial^{2}p^{n+1}}{\partial z^{2}}\right)&=-\frac{\partial}{\partial y}\left(\vec{U}^{n}\cdot\nabla\right)v_{\psi}^{n+1}-\frac{\partial}{\partial z}\left(\vec{U}^{n}\cdot\nabla\right)w_{\psi}^{n+1}+\frac{\partial b_{y}^{n+1}}{\partial y}+\frac{\partial b_{z}^{n+1}}{\partial z}\\
\vec{U}^{n}\cdot\nabla u^{n+1}-\frac{1}{Re}\left(\frac{\partial^{2}u^{n+1}}{\partial y^{2}}+\frac{\partial^{2}u^{n+1}}{\partial z^{2}}\right)&=b_{x}^{n+1}-\frac{\partial p^{n+1}}{\partial x}\\
\left(\frac{\partial^{2}\phi^{n+1}}{\partial y^{2}}+\frac{\partial^{2}\phi^{n+1}}{\partial z^{2}}\right)&=-\frac{\partial u^{n+1}}{\partial x}
\end{aligned}
\qquad (3.5)
$$

Note that all of the equations are linear in the implicit unknowns $\Omega_{x}^{n+1},\psi^{n+1},p^{n+1},u^{n+1},\phi^{n+1}$. Furthermore, the highest-order terms in each of equations (Eq. (3.5)) is the Laplacian of one of these dependent variables. If the equations are ordered as in Eq. (3.5) above, these Laplacian terms fall on the diagonal of the system. Finally, without considering the boundary conditions these equations would constitute a sequentially-decoupled system of equations in the ordered variable set $\left[\Omega_{x}^{n+1},\psi^{n+1},p^{n+1},u^{n+1},\phi^{n+1}\right]$. However, boundary conditions for all the velocity components at a viscous or no-slip boundary are usually required. The implicit treatment of the tangential velocity boundary condition linearly couples the Ω_{x}^{n+1} and ψ^{n+1} variables at the no-slip boundary. Therefore, the vorticity

and stream function equations are solved as a coupled 2×2 equation set, and equations (Eq. (3.5)) are thus solved as a sequentially block-decoupled, second-order, linear system in the ordered variable set $\left[\Omega_x^{n+1}, \psi^{n+1}\right]^T, \left[p^{n+1}\right], \left[u^{n+1}\right], \left[\phi^{n+1}\right]$. The diagram in Figure 3.1 illustrates the sequential block-decoupled structure of the equations and indicates the order in which the equations are solved. The dependent variables for each equation are shown inside the blocks.

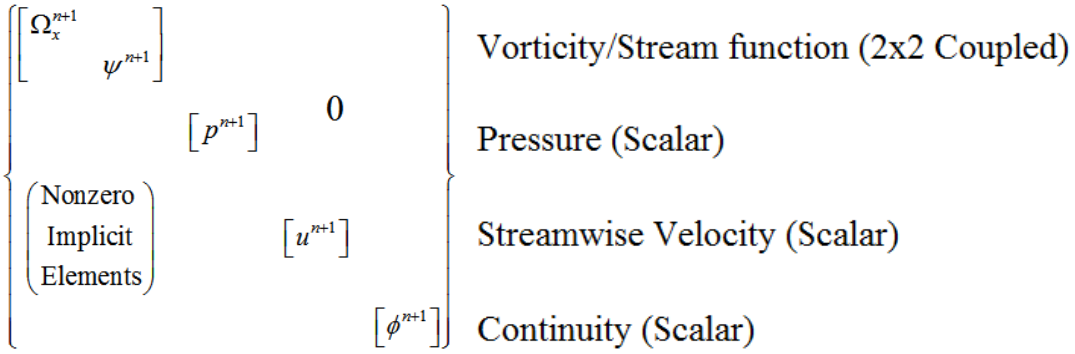


Figure 3.1 Diagram of sequential block-decoupling of the equations with their dependent variables

Given an initial or known solution at a spatial step, *n*, the solution at the next marching step, *n+1*, can be computed using a spatially discretized approximation of Eq. (3.5). First, the vorticity and the stream function equations are solved as a coupled linear system for $\Omega_x^{n+1}, \psi^{n+1}$. Next the pressure equation is solved for p^{n+1} using secondary velocity components $\vec{V}_{\psi}^{\,n+1}$ evaluated from definitions in Eq. (3.2) with the just-computed stream function values ψ^{n+1}. The *x*-momentum equation is then solved for u^{n+1} using the updated pressure field p^{n+1}. Note that in the case of internal flows[33,34], the Neumann boundary conditions imply an arbitrary constant for p^{n+1} that can be evaluated by secant iteration to preserve the global mass flow rate. Finally, the continuity equation is solved for ϕ^{n+1} using updated values of u^{n+1}, and the secondary velocity components $\vec{V}_{\phi}^{\,n+1}$ are updated using ϕ^{n+1}.

A structured or unstructured grid can be used for the spatial discretization of Eq. (3.4). In this study, two different implementations were carried out, one utilizing structured grids and the other utilizing unstructured grids. A finite difference methodology is used for solving the equations. The first implementation used unstructured grids (triangles and quadrilaterals) for the cross-sectional plane (*y*-*z* plane). Derivatives in the transverse plane were computed using a weighted linear least-squares method[36,37], and streamwise derivatives were obtained using the standard 2-point backward-difference expression. The unstructured grid offers the advantage of having fine spacing in the regions of interest (in the turbine rotor region) and coarsening quickly away from the regions of interest. However, it was found that for a general unstructured mesh, the resulting discretization of the Laplacian operator does not yield a diagonally dominant matrix. As a result, typical point iterative solvers like symmetric Gauss-Seidal cannot be used. Krylov subspace based methods like GMRES must be used for solving such systems.

The next implementation utilized structured grids for the cross-sectional plane and the derivatives in the plane were computed using central difference approximations. This implementation improved the runtime and required less memory. The advantages arise from the fact that the spatial discretization resulted in a 5-point stencil for structured grids as compared to a larger stencil for unstructured grids. A smaller stencil requires less memory and less computations in the linear solver.

The linear discretized 2×2 and scalar systems can be solved using any linear solver of choice. In the present implementation, symmetric Gauss-Seidel iteration is utilized. The solution is obtained by spatial marching in the *x*-direction from the inflow boundary to a specified outflow location. Turbulence is modeled using an eddy viscosity model like the one equation Spalart-Allmaras model.

The initial solution at the inflow boundary is first specified, and boundary conditions are required for all the boundaries in the transverse plane. Three different boundary types arise in the present simulations: farfield, viscous-wall, and symmetry conditions. The equations for these are given in Eq. (3.6) with $\hat{n}$ denoting the unit normal to the boundary and indices *w* and *i* denoting nodes on the

boundary and first node off the boundary respectively. Here *h* denotes the distance between a boundary node and the first interior node from that boundary node.

Farfield	Viscous wall	Symmetry
	$u = \psi = 0$	
	$\frac{\partial \phi}{\partial n} = 0$	
$\Omega_x = \Omega_{x\infty}$		$\frac{\partial \Omega_x}{\partial n} = \frac{\partial p}{\partial n} = \frac{\partial u}{\partial n} = \frac{\partial \phi}{\partial n} = 0$
$p = p_\infty$	$\hat{n} = \hat{j} : \frac{\partial p}{\partial y} = \frac{\partial \Omega_x}{\partial z}$ *and* $\Omega_{xw} + \frac{2}{h^2}\psi_i = \frac{2}{h}\frac{\partial \phi}{\partial z}$	
$u = u_\infty$		$\psi = 0$
$\psi = \phi = 0$	$\hat{n} = \hat{k} : \frac{\partial p}{\partial z} = -\frac{\partial \Omega_x}{\partial y}$ *and* $\Omega_{xw} + \frac{2}{h^2}\psi_i = -\frac{2}{h}\frac{\partial \phi}{\partial y}$	

(3.6)

D. Wind Turbine Model

The wind turbine is modeled using a Blade Element Momentum method (BEM) by utilizing the open-source code FAST[11] from NREL. The SOWFA[24] suite–a modified version of FAST, also from NREL–is designed for use with the CFD code OpenFOAM[38] for modeling a wind turbine using the Actuator Line method. The API of the SOWFA suite is utilized by the PNS model to transfer velocities to and receive forces from SOWFA. It should be noted that the PNS model developed is for steady incompressible flow. FAST simulates a wind turbine by utilizing a time-marching algorithm and reports position of the blades and forces on them at each time step. Since, it is not possible to include time varying blade positions in the PNS model, the forces are time averaged to get an effect analogous to an Actuator Disk. In order to incorporate the results from FAST, the following sub-models are required.

1. Gaussian distribution for approximate axial extent of upstream effects

A consequence of the parabolic model is that the flow upstream of the turbine does not feel the effect of the turbine and therefore, does not slow down. To approximate the upstream influence of the turbine, the time-averaged forces computed by FAST are distributed some distance upstream of the

turbine. This distance is known to be of the order one, i.e., about the size of the turbine rotor. Based on results from Navier-Stokes simulations, this physical length is set to one rotor diameter. Note that this length is set *a priori*, i.e., before the start of the simulation. A Gaussian distribution, as given in Eq. (3.7), is utilized for distributing the force in the axial direction. Eq. (3.7) gives the force for a node that is x_D distance away from the turbine in the streamwise/axial direction. x_D is the distance from the turbine nondimensionalized by the turbine diameter. The force application is done in a conservative fashion such that the sum of the forces applied at all the planes for a node is equal to the time-averaged forces computed for that node.

$$\vec{f}_p = \vec{F} \exp\left[-x_D^2\right] \tag{3.7}$$

2. Gaussian in-plane interpolation

The forces computed by FAST are time averaged and stored on a polar grid with a user-specified number of circumferential points and radial points. The number of circumferential points should be enough (at least 36) to ensure sufficient resolution in the circumferential direction. The number of radial points is equal to the number of points used to discretize the blade in FAST. Since the polar grid points may not coincide with the PNS computational grid, forces for the computational grid are interpolated by assuming a Gaussian distribution as in Eq. (3.8). Here *r* is the distance between a computational node and a polar grid point. This distance is nondimensionalized by the projection width, ϵ. The force interpolation is done in a conservative fashion.

$$\vec{f} = \vec{f}_p \exp\left[-\left(r/\epsilon\right)^2\right] \tag{3.8}$$

3. Simulation Process

At each spatial marching step, the distance to the downstream wind turbine is computed, and if it crosses a threshold value of one rotor diameter (1 D), the following occurs: The velocity field from the

previous plane (Figure 3.2) is used to run FAST for a user-specified number of revolutions of the turbine (typically 4). The induction feature is enabled in FAST to simulate the slowing down of the flow as it approaches the turbine to obtain accurate aerodynamic forces. The tip-loss and hub-loss models are switched-on in FAST to account for these losses. The forces computed by FAST are time averaged over the certain number of revolutions to neglect/discount the effect of start-up transients (typically 2). The time-averaged forces are stored on a polar grid as described above. It should be noted that the FAST program is called only once per turbine to simulate the turbine and compute the forces. These forces are distributed as the PNS simulation progresses.

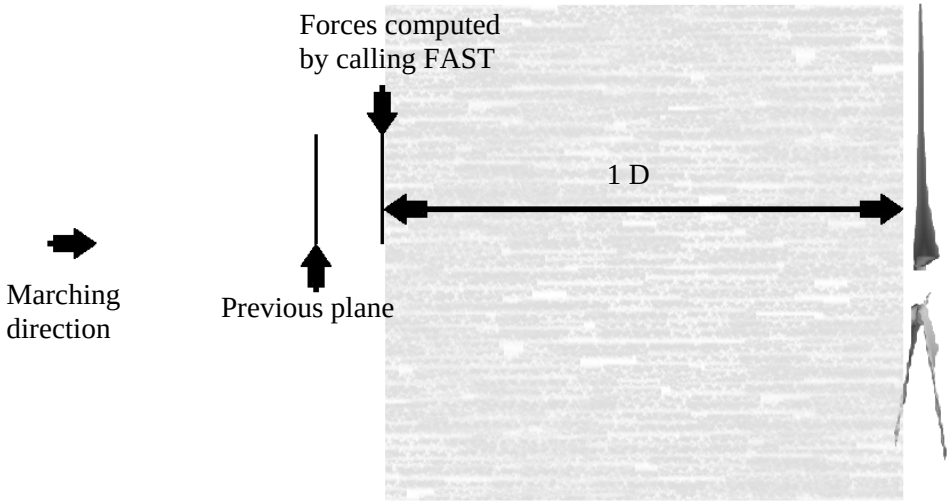


Figure 3.2 Illustration showing the location where FAST is called. Forces are distributed over the shaded region

The time-averaged forces $\vec{F}$ returned to the PNS code are stored for the marching solution. To approximate the upstream-influence effect of the turbine, the forces are distributed among all planes from 1 rotor diameter upstream of the turbine to the turbine itself using a Gaussian distribution (shown by the shaded region in Figure 3.2). At each plane, Gaussian in-plane interpolation is carried out to transfer the forces from the polar grid to the PNS computational grid as discussed above. It should be

noted that in the present model, the pressure field does not react to the streamwise source term. However, the effect of the secondary velocities and the transverse source terms on the pressure field is captured.

CHAPTER 4

VERIFICATION AND VALIDATION OF THE PNS MODEL

In this chapter the developed PNS model is validated and verified using several test cases. The Spalart-Allmaras model[39] is used for turbulence closure. Three test cases are given here for basic validation. The first two test cases involve simulating flow over a flat plate. Laminar and turbulent boundary layer solutions are well known for a flat plate and are used for validation purposes. For these cases, the model gives excellent agreement with theoretical/experimental results. The third test case is the convection of a streamwise vortex for which the computed PNS solution is compared with a full Navier-Stokes solution.

As a practical application example, the PNS model is used to simulate the flow through the NREL offshore 5-MW baseline wind turbine[40], and the computed solutions are compared with the blade-resolved Navier-Stokes solutions. All the cases presented here were run with both the unstructured and structured codes. These results have been presented in Mittal et al.[41] and Mittal et al.[42]

A. Laminar flow over a flat plate

Laminar flow over a flat plate is simulated for a Reynolds number of 20,000. The step size is chosen as 0.01 m for marching in the streamwise direction. The first point off the wall is 10^{-3} m and the mesh contains 671 nodes (61 nodes in y and 11 nodes in z directions). To avoid the singularity at the leading edge of the plate, the solution (velocity profile) is initialized at planes corresponding to x=0.01 m and x=0.02 m. The boundary layer thickness (for initializing the solution) is computed using Eq. (4.1) and only

the streamwise velocity is specified using Eq. (4.2). Note that the velocity normal to the wall/plate (*y*-velocity) is initialized to zero at both the planes.

$$\delta = 4.91x / \sqrt{\mathrm{Re}_x} \tag{4.1}$$

$$u = 1.5\left(\frac{y}{\delta}\right) - 0.5\left(\frac{y}{\delta}\right)^3 \tag{4.2}$$

The computations were conducted for a total of 200 steps and the final solution at *x*=2 m is shown along with the RANS solution and the Blasius solution. Figure 4.1 shows a plot of the nondimensional *x*-velocity against the similarity variable ($\eta = y\sqrt{u_\infty / 2\nu x}$). The agreement, as can be seen, is good. The boundary layer thickness is estimated to be around 0.050 m from the computed solution at *x*=2 m. This is higher as compared to a value computed using Eq. (4.1), which is 0.0491 m. The error of 1.8 % is most likely due to the incorrect initialization of the wall-normal velocity (*y*-velocity).

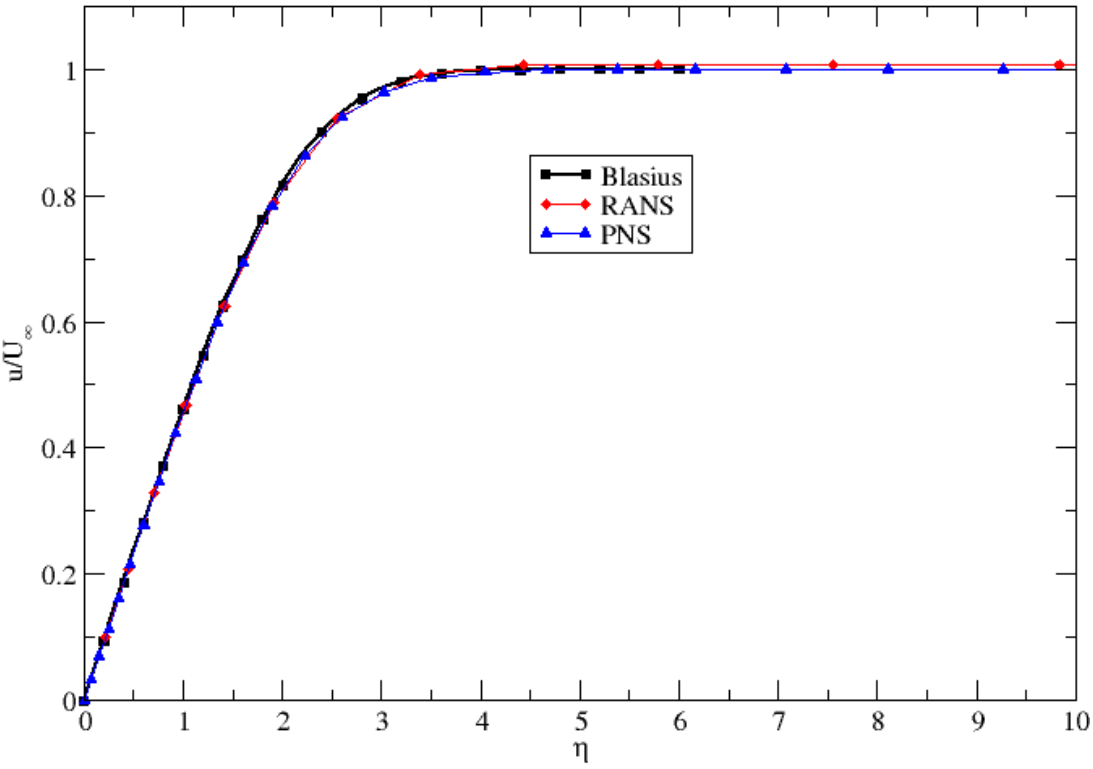


Figure 4.1 Plot of similarity variable against the *x*-velocity for laminar flow over a flat plate

B. Turbulent flow over a flat plate

For turbulent closure, the Spalart-Allmaras[39] model is utilized. The turbulence model is updated after each marching step of the mean-flow (PNS) equations. The Reynolds number for the flow is 10^6. The total number of nodes is 793 (61 in *y* and 13 in *z* directions) and the off-the-wall spacing is 10^{-5} m. The solution is marched for a total of 100 steps (step size of 0.01 m). However, the marching is started at *x*=5 m from the leading edge. This was done to avoid the transition region and ensure that the flow is fully turbulent. The solution is initialized at planes corresponding to *x*=5.01 m and *x*=5.02 m with the Coles' law of the wake (Eq. (4.3)). The value of the Coles' parameter, Π, is 0.45 and boundary layer thickness is assumed to be 0.06 m. The boundary layer thickness was chosen due to the inaccuracies in the expressions that estimate turbulent boundary layer thickness.[43]

$$u^+ = \frac{1}{\kappa}\ln\left(y^+\right) + B + \frac{2\Pi}{\kappa}\sin^2\left(\frac{\pi y}{2\delta}\right) \tag{4.3}$$

The initialized solution at *x*=5.01 m and the computed solution at *x*=6 m are plotted against the law of the wall in Figure 4.2. The velocity profiles obtained from various locations on the plate were found to be self-similar. The skin friction coefficient computed at *x*=6 m is $2.68\cdot10^{-3}$. The value of skin friction coefficient calculated using Eqs. (4.4) and (4.5) is 2.90×10^{-3} and 2.61×10^{-3} respectively. The agreement of the PNS solution is excellent for the velocity profile and within range of various expressions for the skin friction coefficient. The boundary layer thickness is estimated to be about 0.068 m at *x*=6 m.

$$C_f = 0.027\mathrm{Re}_x^{-1/7} \tag{4.4}$$

$$C_f = 0.0592\mathrm{Re}_x^{-1/5} \tag{4.5}$$

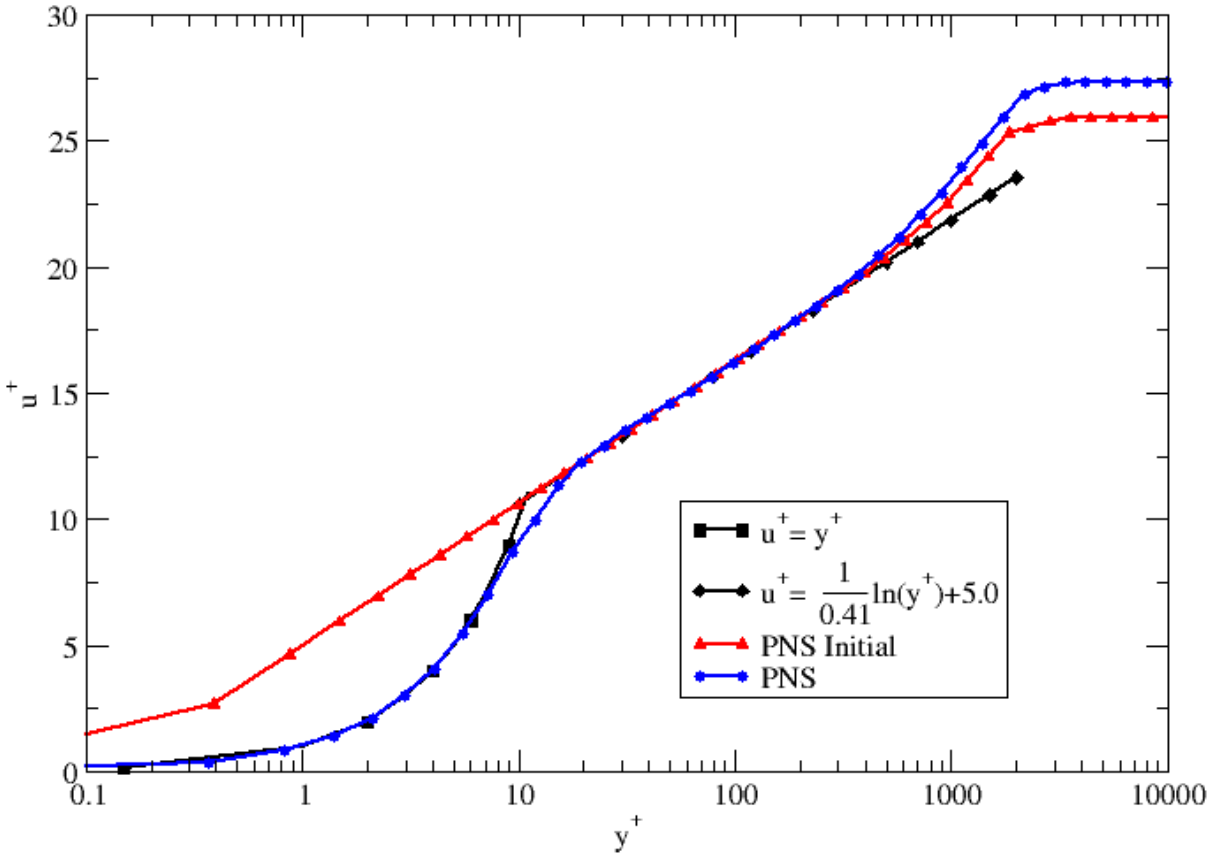


Figure 4.2 *X*-velocity plot for the turbulent flow over a flat plate

C. Convection of a vortex

In the previous two test cases, the streamwise vorticity and the stream function were identically zero in the entire flow field. This test case is designed to test the equations for these two variables. A streamwise vortex is created by imposing forces in the transverse plane (*y*-*z* plane). The vortex is convected downstream (+*x*) by a uniform axial flow. This is akin to a tip vortex at a wing tip being generated continuously and convected by the freestream flow. The grid for this case extends 1 m in *y* and *z* directions and has 41 points (uniformly spaced) in both the directions. The center of the vortex is at (0.5, 0.5) with a radius of 0.1 m. The force per unit volume applied for creating the vortex is 12.25 N/m^3 (0.1 nondimensional units). The marching step size is 0.01 m and the forces are applied on five

consecutive planes from 0.03 m to 0.07 m. The Reynolds number for the flow is 6.8×10^5 and the flow is assumed laminar.

To assess the accuracy of the computed results, the same flow conditions are simulated using a Navier-Stokes code with the same forces applied as source terms to generate the vortex. The grid for the RANS simulation is generated by extruding the planar grid (for PNS) with the spatial marching step size (0.01 m). This ensured that the spacing in all the three directions (Δx, Δy and Δz) is exactly the same for both the simulations. The RANS simulation was run as a steady-state problem with local time-stepping.

Table 4.1 compares the PNS and the RANS results. RANS results are presented for three different spatially accurate schemes: second order, fifth order and seventh order. The maximum and minimum of *y*-velocity and streamwise vorticity at three downstream locations is presented in the table. An initial observation from the data is that the vortex dissipates as it convects downstream, as expected. Figure 4.3 shows a plot of the *y*-velocity along a horizontal line from outside the vortex to its center at three different downstream locations. The plot compares the PNS solution with the seventh order RANS solution. Both (Table 4.1 and Figure 4.3) show that the dissipation is minimal, if any, based on the PNS results. The amount of dissipation varies significantly for the RANS results with large dissipation for the second order scheme and least for the seventh order scheme. As the order of accuracy is increased, the trend of the RANS solutions is to tend towards the values computed using the PNS model. It is also observed the streamwise vorticity magnitudes predicted by the RANS simulation are significantly lower than the PNS model predictions. Since, the flow is treated as laminar, only molecular viscosity should be dissipating the vortex apart from the numerical diffusion. The vortex should dissipate as it convects downstream. However, in the RANS simulations, the numerical diffusion seems to be playing a substantial role and is evident as the order of accuracy is increased (Table 4.1).

Table 4.1 Comparison of Parabolized Navier-Stokes and Navier-Stokes results for the vortex convection case at 1, 5 and 9 diameters downstream locations

	2^{nd} order spatially accurate scheme		5^{th} order spatially accurate scheme		7^{th} order spatially accurate scheme		7^{th} order spatially accurate scheme (finer grid)		PNS	
Location (m)	Max.	Min.	Max.	Min.	Max.	Min.	Max.	Min.	Max.	Min.
					y-velocity $\left(\times 10^{-2}\right)$					
0.2	5.184	-5.313	5.372	-5.603	5.271	-5.698	5.162	-5.359	5.584	-5.695
1.0	3.994	-3.995	4.901	-5.031	5.135	-5.466	5.248	-5.262	5.585	-5.692
1.8	3.336	-3.318	4.484	-4.553	4.884	-5.116	5.255	-5.357	5.595	-5.692
					Streamwise vorticity					
0.2	0.8014	-2.948	0.8011	-3.104	0.8214	-3.124	1.810	-6.030	1.109	-3.999
1.0	0.7460	-2.162	0.7090	-2.818	0.6874	-3.051	1.594	-5.315	1.110	-3.993
1.8	0.6604	-1.780	0.7324	-2.561	0.6970	-2.949	1.449	-4.699	1.109	-3.987

To further confirm this, another RANS simulation was conducted for the same flow problem on a finer grid. The finer grid had reduced grid spacing (81 points in both the *y* and *z* directions as compared to 41 in the original grid) in the *y* and *z* directions but exactly the same spacing in the streamwise direction. The results of the RANS simulation with the seventh order spatially accurate scheme are also presented in Table 4.1. The *y*-velocity is in good agreement with the PNS results. However, the streamwise vorticity is considerably higher than the PNS values. This is because of the discontinuous force distribution applied to generate the vortex. The magnitude of the force is exactly zero at the center of the vortex and immediately outside the vortex radius. As a result, even though the velocity field is similar for the simulations on coarse and fine grids, due to the spacing differences, the gradients of the same velocity field are different, specifically, higher for the fine grid. Thus, the streamwise vorticity magnitudes are higher on the finer grid. This was confirmed by conducting simulations (both Navier-Stokes and PNS) on successively finer grids.

Since this test case is fictitious, it is not possible to compare the computed results with experimental or theoretical data, as in the previous test cases. We believe that the PNS results are accurate because the vorticity formulation is more accurate due to the fact that the derivatives of the velocity are dependent variables instead of the velocity itself. The truncation error acts on the velocity field in RANS and on the streamwise vorticity in PNS. In fact, two-dimensional vorticity/stream-function formulations were popular in the past but were found to be inconvenient in three dimensions.

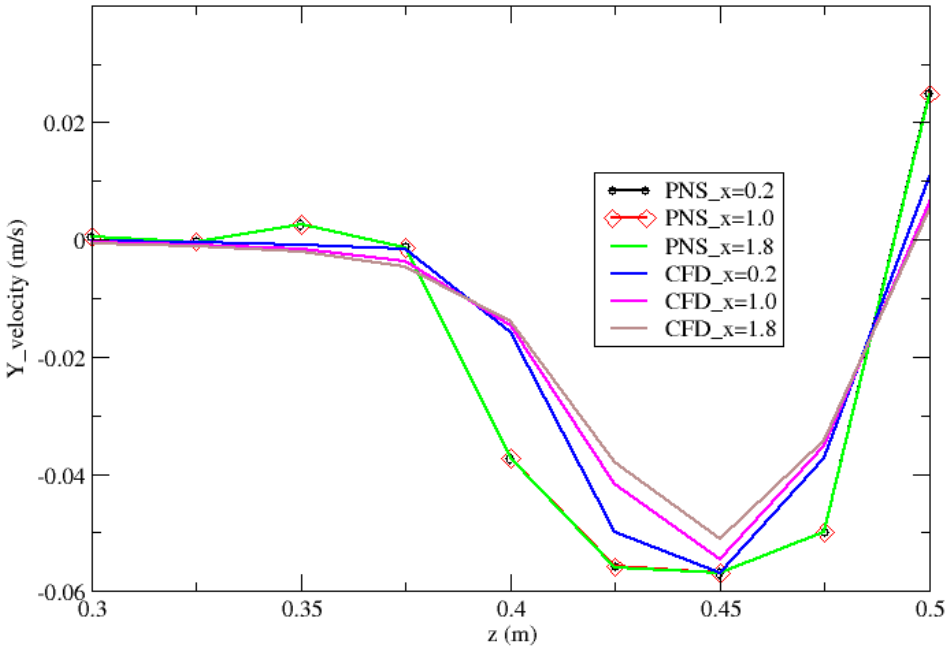


Figure 4.3 Comparison of *y*-velocity for the PNS and RANS solutions (7^{th} order) at different downstream locations (41 points in *y* and *z* directions)

D. Wind turbine

The NREL offshore 5-MW baseline wind turbine[40] is a representative utility-scale multi-megawatt turbine. It is a conventional three-bladed upwind, variable-speed, variable blade-pitch-to-feather controlled turbine. The design is closely related to the Repower 5MW turbine as well as other

conceptual models from WindPACT, RECOFF and DOWEC projects. The rotor diameter is 126 m with a hub height of 90 m. The turbine design includes a rotor tilt of 5º and a blade precone of 2.5º. The rated wind speed is 11.4 m/s, at which the rotor rotates at 12.1 rpm. The blade is 61.5 m long and is made up of different airfoil sections. The root section is a circle/cylinder which transitions into different DU airfoils. The latter part of the blade (from 44.55 m till tip) is comprised of NACA64 airfoil.

The present PNS model is used to simulate flow through the NREL turbine. The simulation is conducted at the design point with a uniform inflow wind speed of 11.4 m/s. The grid extends to 500 m in *y* and *z* directions from the middle of the plane where the wind turbine is assumed to be located. Thus, the boundaries are approximately 4 rotor diameters away from the wind turbine center. The grid contains a total of 10,201 nodes with spacing between the nodes ranging from 40 m at the boundaries to 4.05 m near the center. The spatial marching step size is 5 m and the simulation is run for a total of 200 steps (1000 m). The turbine is assumed to be 150 m from the inflow boundary.

Table 4.2 Comparison of thrust and torque values obtained using different simulation methods

	Reported by NREL	RANS simulation	PNS simulation
Thrust (kN)	800	762	741
Torque (kNm)	4180	4131	4311

To assess the accuracy of the computed results, a blade-resolved Navier-Stokes simulation of the same wind turbine was conducted at the design point. Table 4.2 compares the torque and thrust values obtained using the PNS model against the RANS results and results reported by the NREL[40] (obtained using a BEM method). The torque predicted by the PNS simulation is higher than the NREL reported and the RANS computed values. This is due to the discretization used for the blades in FAST. For this PNS simulation, the blades were discretized using 17 points. The higher torque value due to the

discretization was confirmed by running FAST in stand-alone mode. A discretization of 62 points was tested with stand-alone FAST and confirmed the above-mentioned theory. However, a 62 point discretization was not utilized for this demonstration case because that would require a considerably finer mesh. The errors due to a coarser blade discretization were deemed acceptable for this demonstration case. Figure 4.4 and Figure 4.5 compare the mesh at a plane downstream of the turbine for the PNS and the RANS simulations respectively. The meshes shown are clipped to the size of the turbine to highlight the difference in the resolution required for the two methodologies.

To further assess the accuracy of the PNS model, streamwise velocity is compared downstream of the turbine in the wake region. The mesh for the RANS simulation contained a uniform block downstream of the turbine extending from the rotor to five diameters to resolve the wake. Figure 4.6 shows the contour plots of the axial (streamwise) velocity on an axial plane passing through the center of the turbine. The dissipation is immense for the RANS solution after the grid refinement ends at five diameters downstream. There is no apparent dissipation in the PNS solutions. Figure 4.7 compares the axial velocity at a plane two rotor diameters downstream of the turbine between the RANS solution and the PNS solution. The velocity is higher at the center of the wake due to the absence of the hub. Moreover, since only 36 circumferential stations were used for storing the forces, a large projection width (for Gaussian distribution) had to be utilized to ensure a reasonably smooth flow field. This effect coupled with the absence of the hub leads to a weaker wake as seen in the figure. It is not possible to include the hub geometry in the simulation since stagnation points violate the assumptions of the present model. The overall character/nature of the wake is captured quite well by the PNS. Also, since the turbine is modeled by applying forces averaged over the entire rotor disk, the tip and root vortices are absent.

It was found that the turbulence model was not generating any turbulence. There are a couple of reasons for this. First, since the forces are averaged over the entire rotor disk, the velocity gradients, which play a critical role in turbulence production, in the flow field are small. Second, since there are no

solid/viscous/no-slip walls present for the blades, the distance to the nearest no-slip wall in the turbulence model is set to infinity. Moreover, strong velocity gradients near the blade are due to the no-slip boundary condition and do not arise in the flow field in the absence of the viscous wall. Thus, the combined effect is that the turbulence production is negligible for this flow simulation. Mittal et al.[41] have discussed this issue in further detail.

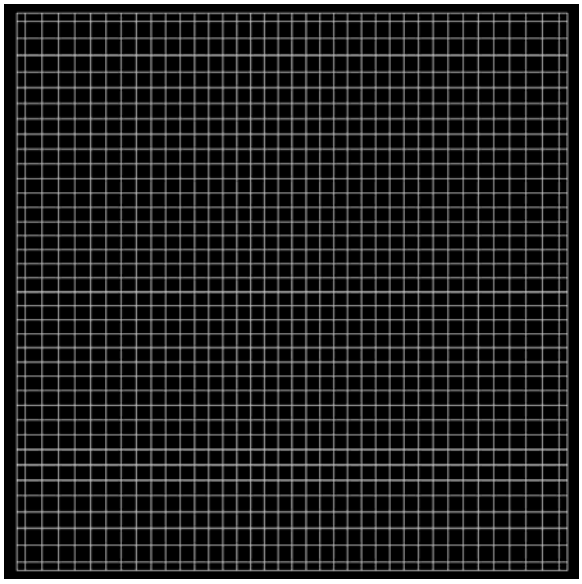


Figure 4.4 Cross-section of the mesh downstream of the turbine for PNS

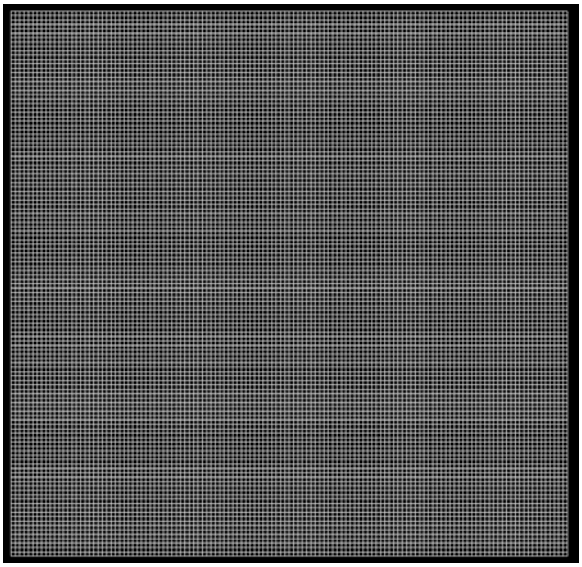


Figure 4.5 Cross-section of the mesh downstream of the turbine for RANS

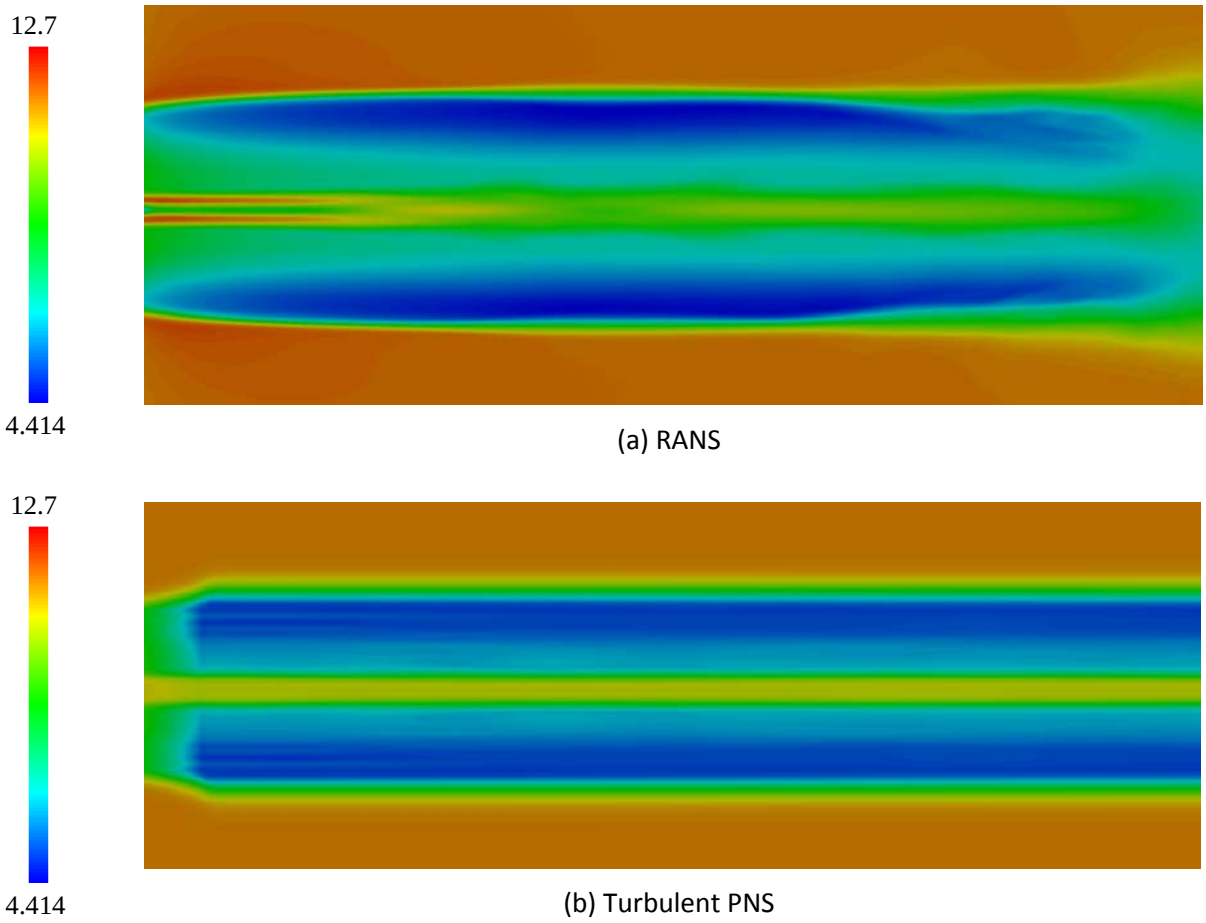


Figure 4.6 Contours of axial velocity comparing the wake between RANS and PNS solutions

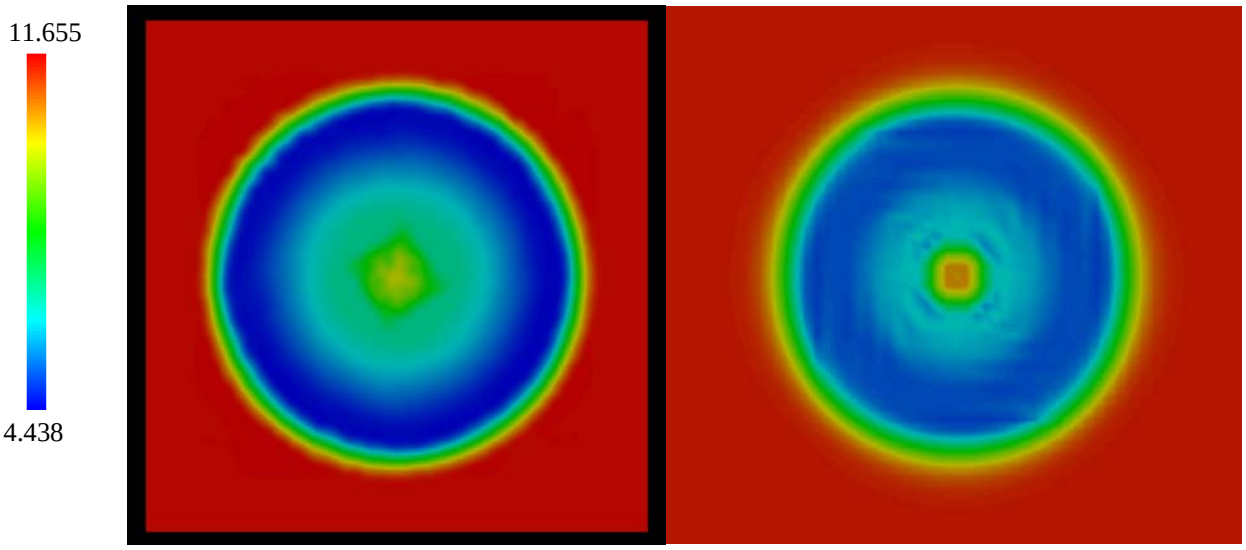


Figure 4.7 Contours of streamwise (*x*-velocity) velocity at two rotor diameters downstream of the wind turbine

The runtime of the PNS simulation was 88 minutes on a single core of a desktop computer. The Navier-Stokes simulation required 48 cores of the in-house cluster (with processors comparable to the ones on the desktop computer). For a single wind turbine, the simulation can be run as a steady-state simulation using a relative reference frame and would take about 28 hours. However, for more than one wind turbine, an unsteady simulation is necessary and for the above-discussed case, would take over 202 hours. Thus, the computational cost for steady-state and unsteady simulations are 1,344 and 9,700 processor hours respectively.

CHAPTER 5

TOWARDS WIND FARM SIMULATIONS USING THE PNS MODEL

In the previous chapter, the developed PNS model was validated and verified using several test cases. The motivation behind the development of the PNS model was to conduct flow simulations of entire wind farms efficiently, i.e., with high fidelity and low cost. In this chapter, results obtained by the application of the developed PNS model are presented. Flow simulation through a wind farm of 5 wind turbines is presented first. The next section demonstrates the utility and usability of the PNS model for layout optimization of the wind farm. The last section discusses the issues related to modeling the Atmospheric Boundary Layer (ABL) in RANS and PNS codes' context.

A. Wind Farm

A wind farm of five NREL 5 MW wind turbines is simulated using the PNS method. Three turbines are located upstream in a row with 1.5 rotor diameter lateral distance. The second row of the other two turbines is located 5 rotor diameters downstream of the first row. The layout of the wind turbines in the wind farm is shown in Figure 5.1. Table 5.1 gives the locations of the turbines. The *y* coordinate value is 0.0 m for all the turbines.

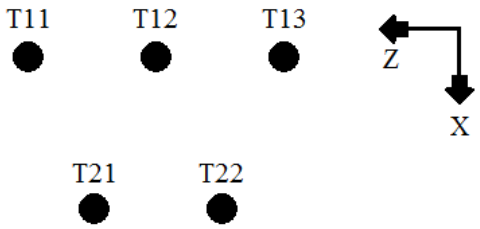


Figure 5.1 Layout of the turbines in the wind farm

Table 5.1 Location of the turbines in the wind farm

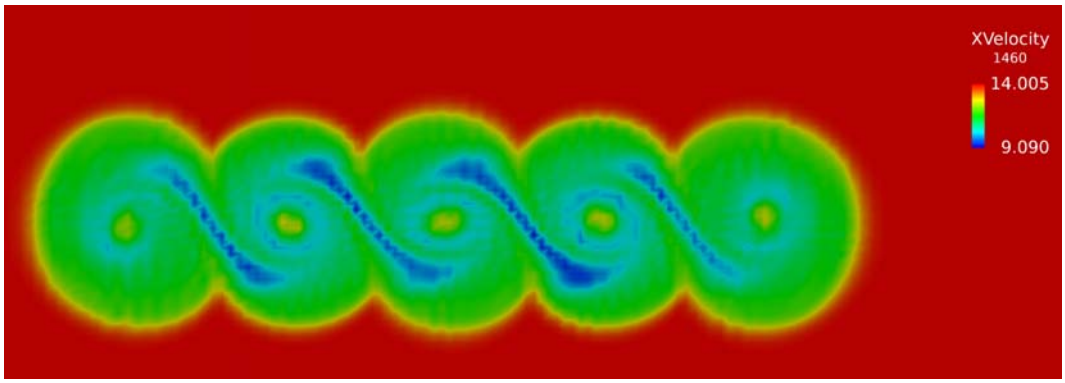
Turbine Id	T11	T12	T13	T21	T22
z coordinate (m)	-189.0	0.0	189.0	-94.5	94.5

The domain of the simulation in the transverse plane (*yz* plane) extends from -1000 m to 1000 m in the *y* direction and from -1500 m to 1500 m in the *z* direction. A structured mesh is utilized with 151 points in the *y* direction and 201 points in the *z* direction. The first row is 200 m from the inflow boundary. The marching step size is 10 m and the simulation is conducted for 200 steps covering 2000 m. The incoming wind speed is 14 m/s and since this is higher than the rated wind speed (11.4 m/s), the blades of all the turbines are pitched at 8.7 degrees. For the purpose of this study, a controller for the wind turbine is not utilized and thus, the second row turbines may be under-performing. Table 5.2 gives the torque and thrust values for all the turbines in the wind farm. The power production of the second row turbines is slightly less than that of the first row turbines due to the staggered layout which reduces the impact of the wake on downstream turbines and the blades being pitched which lowers the power extraction capabilities of the turbines thus, weakening the wakes.

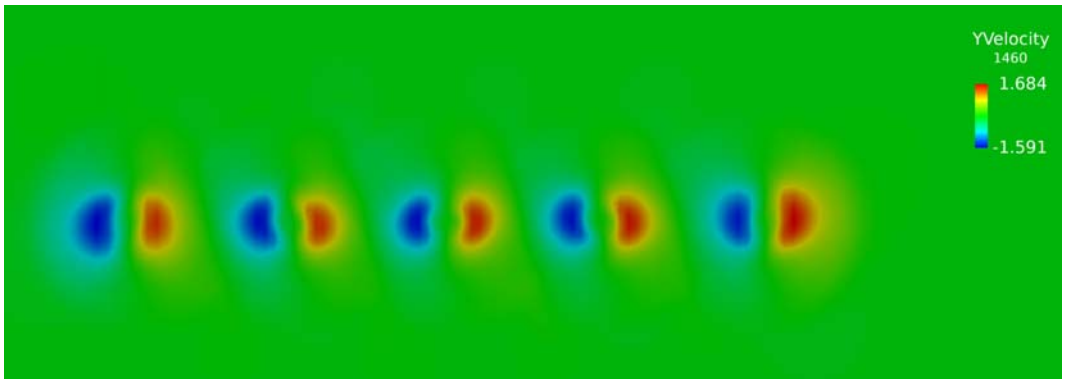
Table 5.2 Thrust and power values for the turbines

	T11	T12	T13	T21	T22
Power (kW)	5349	5349	5349	5079	5101
Thrust (kN)	460.7	460.7	460.7	459.9	461.2

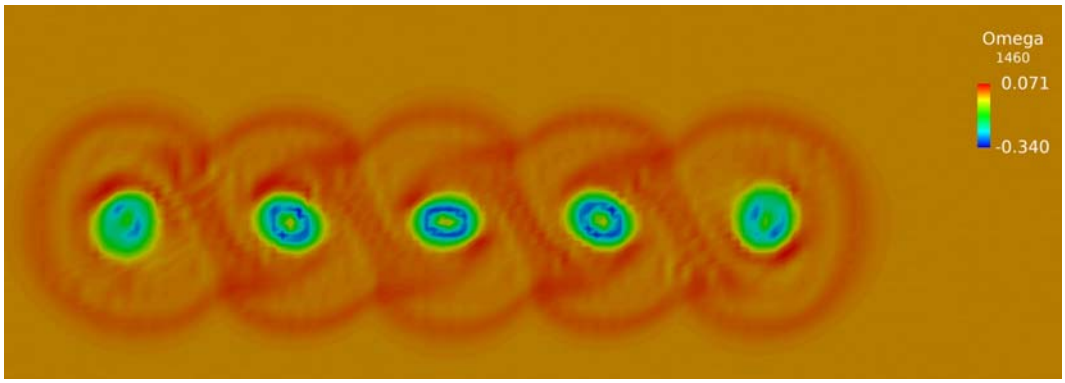
Figure 5.2 (a) shows the contours of the *x*-velocity (streamwise) at five rotor diameters downstream of the second row of the turbines. Five wakes are seen corresponding to the five turbines. As explained above, since the power produced is similar, the wake deficits are also similar. Since the PNS model simulates the rotation of the wake, the low velocity regions (blue) are seen stretched and being mixed with the higher velocity flow. Figure 5.2 (b) shows the contours of the *y*-velocity. This gives an estimate about the rotational speed of the flow in the wake region (about 1.6 m/s). The third image (Figure 5.2 (c)) shows the contours of the streamwise vorticity. The negative values near the hub/root of the blade show that the root vortices are rotating in the opposite direction. Although the results presented in this section are not compared with RANS simulations, Mittal et al.[44] have compared the PNS method with the blade-resolved RANS simulations for an array of wind turbines. Results for a different set of parameters for a wind farm using the PNS method are presented in Mittal et al.[45] Figure 5.3 shows the contours of streamwise velocity on an axial plane passing through the center of the turbines. The gray lines highlight the location of the first and second row of the wind turbines in the wind farm.



(a) *X*-velocity



(b) *Y*-velocity



(c) Streamwise vorticity

Figure 5.2 Contours of flow variables 5 rotor diameters downstream of the second row of turbines

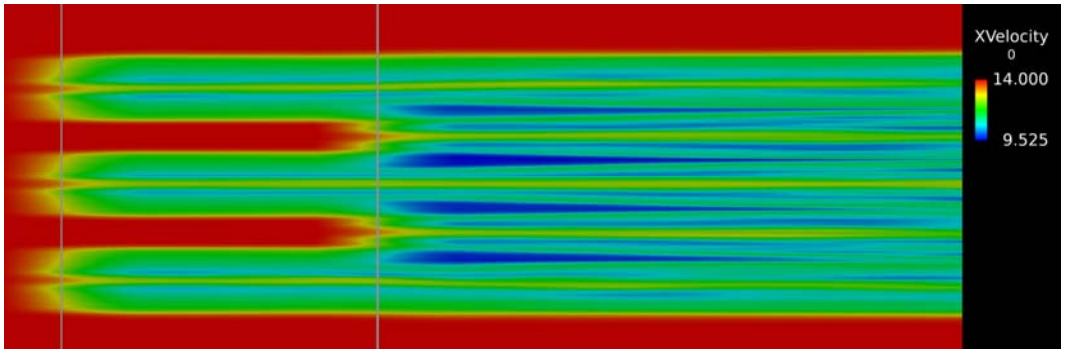


Figure 5.3 Contours of streamwise velocity on a plane passing through the center of the turbines

B. Optimization

Flow simulations of wind farms are used with optimization algorithms for two purposes in the wind industry. First, at the time of development of the wind farm, optimization is conducted to determine optimal layout (positioning of the wind turbines) of the wind farm in terms of minimizing the cost (capital investment and maintenance costs) and maximizing the power output (in turn revenue). Second, once a wind farm is operational, performance of the farm can be improved by carrying out optimization of the operational parameters like the yaw of the turbines or pitch setting of the blades.

In this section, the applicability and suitability of the developed PNS model for optimization purposes is explored. A simple test case with two NREL 5-MW wind turbines in tandem setting is utilized. These turbines are placed five rotor diameters apart with no offset. The incoming wind speed is 14 m/s. Both the turbine's blades are pitched at 8.7 degrees.

The sensitivity derivatives are computed by using a one-sided finite difference approximation. The turbine location is perturbed by 5 m for computing the sensitivity derivatives since the minimum grid spacing is around 4 m. The large perturbation distance is also a consequence of the scale of the problem. Figure 5.4 shows the power produced by the second turbine as it moves out of the wake of the first turbine. The second turbine produces minimum power when it is at the center of the first turbine's wake. Once the turbine has completely moved out of the wake, the power produced levels-off to a

constant value. The sensitivity derivatives for the second turbine's power output are shown in Figure 5.5. It should be noted that for this case, since the first turbine always sees a clean flow, these sensitivity derivatives optimize the total power output of the array of turbines. A peak is seen in the sensitivity derivative plot at the 70 m location. There are few factors at play here. First is the low velocity region at high span locations of the blade (blue region in Figure 4.7). Because of this region, the gain in power output is small when the turbine is moved by 10 or 20 m since the main torque producing part of the blade is still seeing low velocity. Secondly, as the turbine moves out of the wake, the area overlap (between the turbine rotor and the wake) decreases nonlinearly which results in large gains in power output. Each simulation's cost (evaluation of power output for one layout) was around 30 minutes on a single core of a desktop computer.

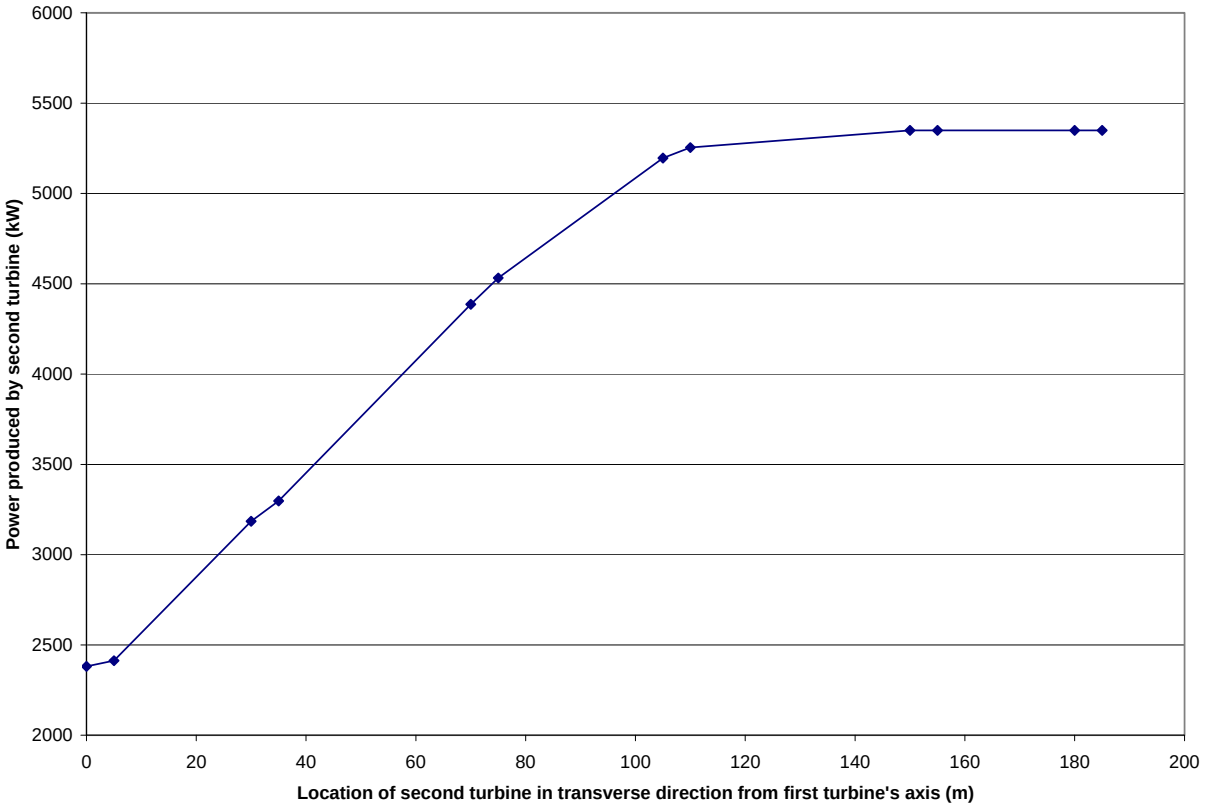


Figure 5.4 Power produced by second turbine as a function of its location

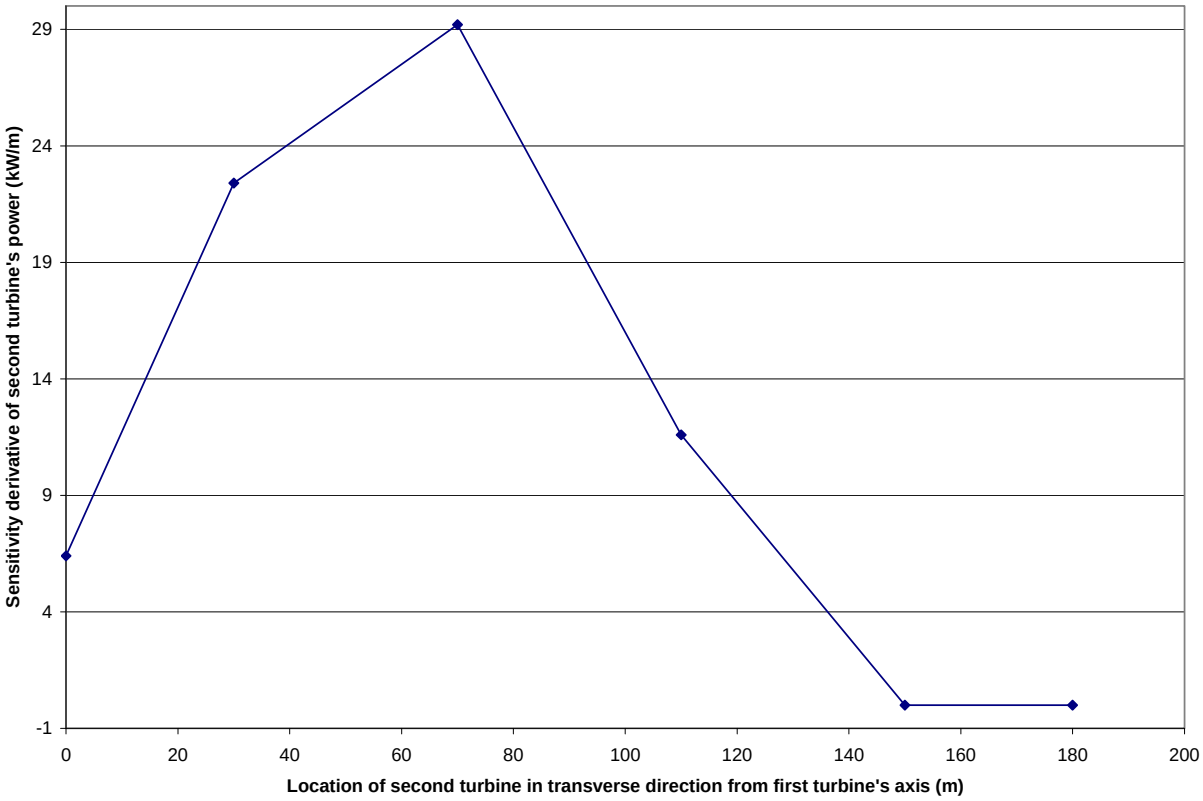


Figure 5.5 Sensitivity derivative of second turbine's power as a function of its location

The second test case involved three turbines in a configuration similar to as shown in Figure 5.1. The turbines labeled T11, T12 and T21 are considered in this test case with the layout positioned such that the T21 turbine is at z = 0 m when right in the middle of the T11 and T12 turbines. As in the previous case, incoming wind speed is 14 m/s with blades pitched at 8.7 degrees for all the turbines. Also the turbine location is perturbed by 5 m for computing the sensitivity derivatives.

The power produced by the third turbine (T21) is shown in Figure 5.6 as a function of its location. The power produced increases as the turbine moves out of the wake of the left turbine (T11) and reaches a maximum value in the middle. As the turbine moves into the wake of the right turbine (T12), the power produced decreases showing the impact of the wake. This behavior is expected as well as intuitive and is confirmed by the sensitivity derivatives shown in Figure 5.7. The sensitivity derivatives are positive while

the turbine (T21) is moving out of the wake of the left turbine (T11), reaching a zero value corresponding to the location of maximum power production, and then going negative when the T11 turbine moves into the wake of the right turbine (T12). Each simulation's cost (evaluation of power output for one layout) was more than 90 minutes on a single core of a desktop computer

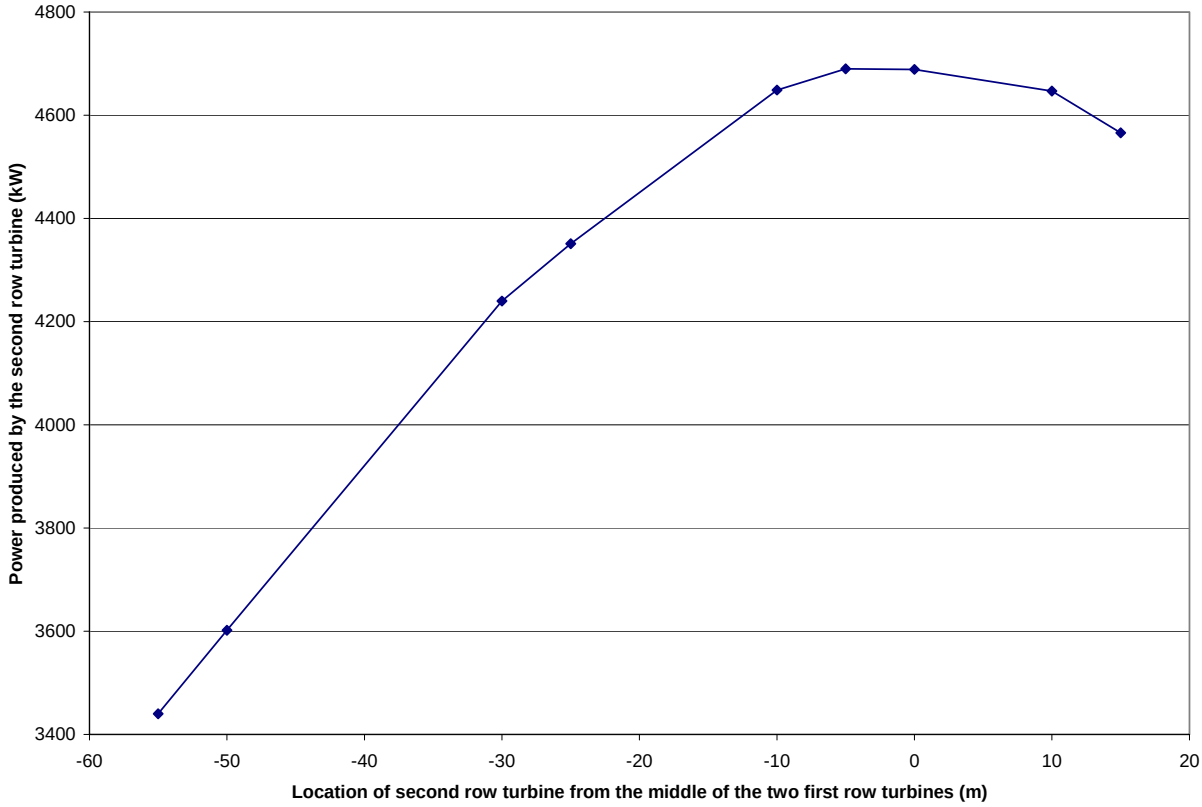


Figure 5.6 Power produced by third turbine as a function of its location

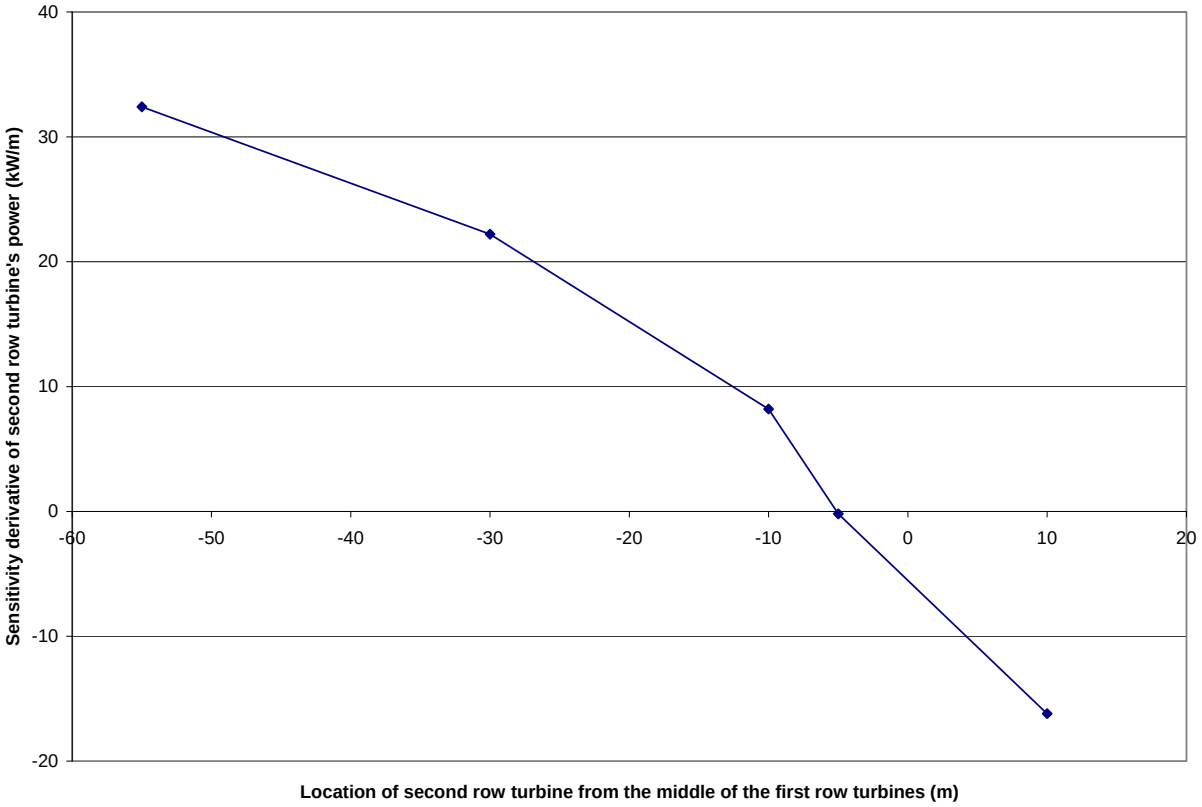


Figure 5.7 Sensitivity derivative of third turbine's power as a function of its location

Given the cost of each PNS simulation (on the order of hours), carrying out layout optimization for a large wind farm from scratch might be very expensive since an optimization of that level typically requires thousands of function evaluations. Analytical wake models can be useful if used to carry out a coarse optimization and then using a higher fidelity model (like PNS) to refine the layout and accurately estimate the power output of a wind farm. Similarly, optimization involving the turbine parameters such as yaw angle and blade pitch can be carried out to improve the overall wind farm output during operation of a wind farm.

C. Atmospheric Boundary Layer

The Atmospheric Boundary Layer (ABL) height ranges from few hundred meters to several thousand meters. Since, the hub height of wind turbines has crossed the hundred meter mark mainly driven by the ever increasing rotor size, the effect of the ABL on the power production and performance of the wind turbines is significant. Thus, it is imperative to model the ABL appropriately and include it in the simulations of the wind turbines and farms.

In a typical CFD simulation involving boundary layers, a fine enough mesh is utilized to resolve all the sub-layers of the turbulent boundary layer (viscous sub-layer, buffer layer, log layer and outer layer) and accurately capture the physics. However, in the case of the ABL, resolving these sub-layers becomes computationally very expensive due to the range of the scales involved and the size of the grid. If coarser grid spacing is used, the standard turbulence models do not predict the turbulence production accurately and the ABL velocity profile is not sustained.

In order to simulate ABL flows without having to resort to resolve all the sub-layers, researchers at the SimCenter have modified the two-equation SST $k-\omega$ model[46] turbulence model. The standard model is modified to provide a neutral ABL that is of consistent thickness throughout the computational domain in addition to maintaining a consistent level of turbulence in the field. The details about the modifications is available in Mittal et al.[47] The effect of the modified turbulence model can be seen from the contour plot of turbulent kinetic energy in Figure 5.8 where the characteristic peak in the turbulent kinetic energy near the ground is seen. This simulation was conducted using the SimCenter's in-house code, *Tenasi*. It should be noted that if the modified turbulence model is not used, the velocity profile or the turbulence level decay very fast through the computational domain.

Figure 5.8 Contour plot of turbulent kinetic energy for neutral ABL

It is an entirely different story for the PNS model. As was reported for the streamwise vortex convection test case in the previous chapter, the dissipation in the PNS model is minimal. As a result, an imposed neutral ABL velocity profile does not decay even without a turbulence model in a PNS simulation. This was ascertained by conducting a simulation in the absence of wind turbines. Figure 5.9 shows the contours of streamwise velocity indicating that the boundary layer thickness is constant and does not change as the simulation marches in the *x* direction.

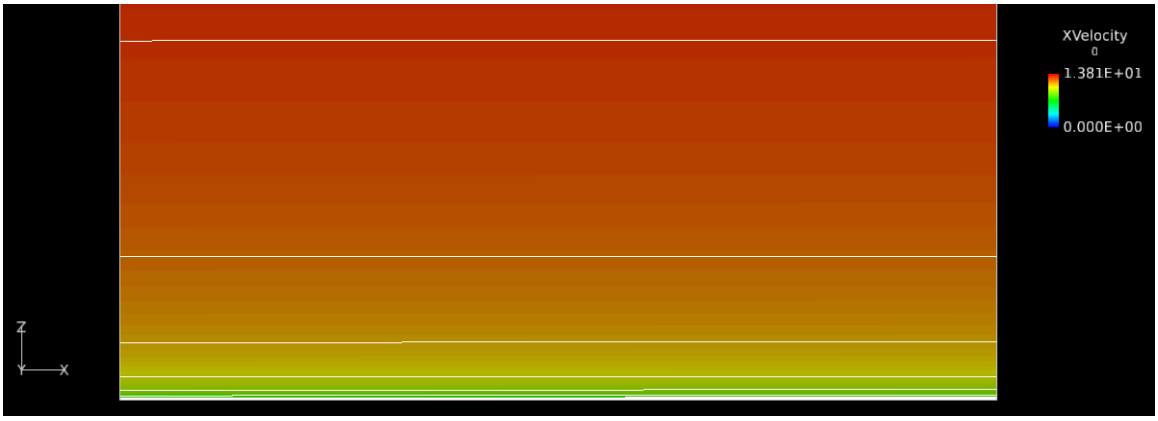


Figure 5.9 Contours of *x*-velocity (streamwise) indicating constant boundary layer thickness

Having ensured that the velocity profile sustains across the entire computational domain, a wind turbine (NREL 5-MW) immersed in the ABL was simulated. The domain of the simulation in transverse plane (*yz* plane) extends from -500 m to 500 m in the *y* direction and from 0 m to 1000 m in the *z* direction. A structured mesh is utilized with 101 points in the *y* and *z* directions. The turbine is located 200 m from the inflow boundary. The marching step size is 10 m and the simulation is conducted for 200

steps covering 2000 m. The boundary layer profile is such that the incoming wind speed of 11.4 m/s occurs at the hub height of 90 m. The blades of the turbine are not pitched and 0 degree pitch angle is maintained.

The torque and thrust produced by the turbine are 4261 kNm and 737 kN respectively. The torque produced is 1.1 % less than the torque production for uniform inflow case. Figure 5.10 shows the contours of streamwise/axial velocity. The first image highlights the wake five rotor diameters downstream of the turbine whereas the second image shows the development of the wake from the turbine location (marked by white line). The PNS simulation was run on a Linux desktop computer and took 57 minutes. Presence of an ABL has negligible effect of the computational cost and the runtime.

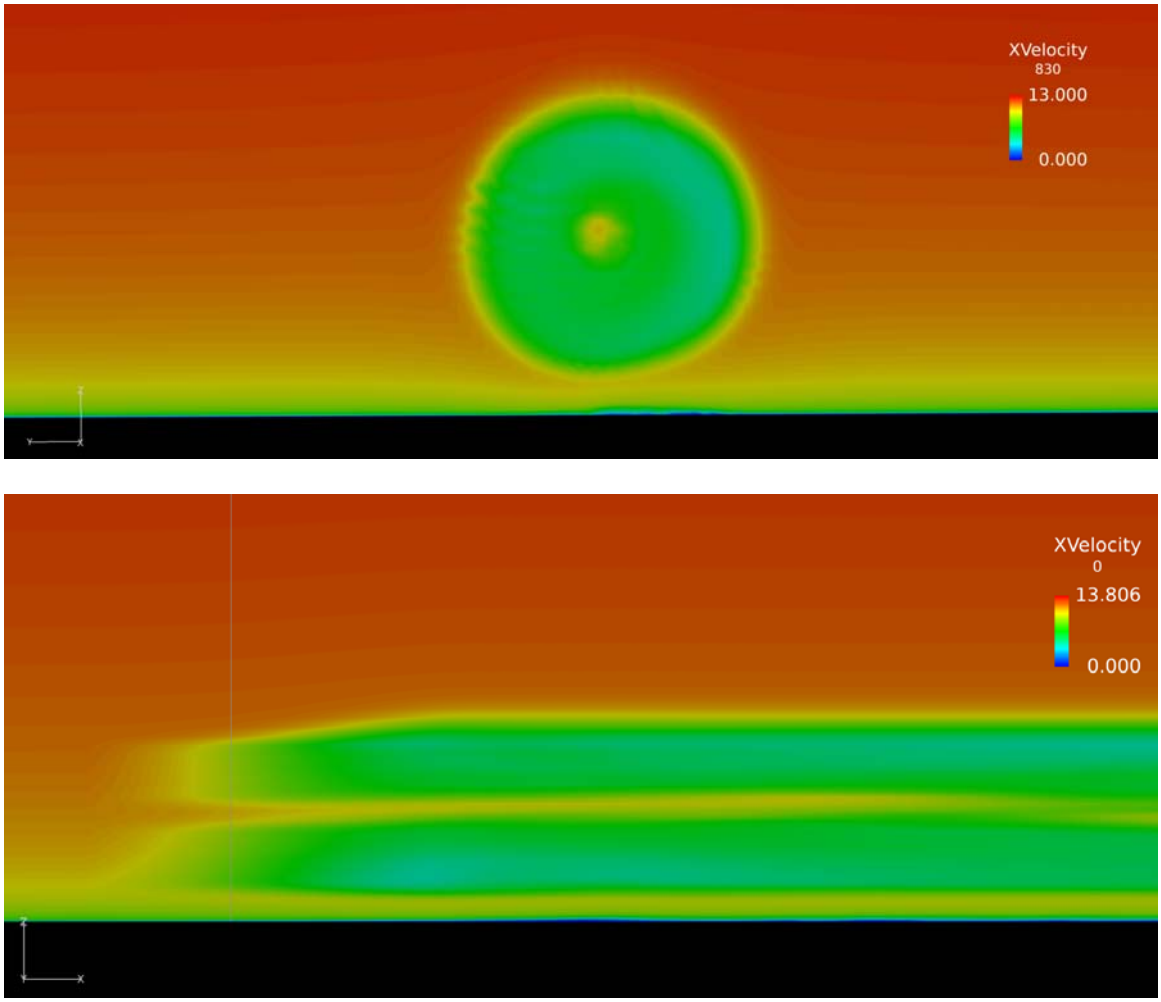


Figure 5.10 Contours of streamwise velocity showing the wake of a turbine in neutral ABL

CHAPTER 6

SUMMARY AND CONCLUSIONS

This study was motivated by the need for high fidelity low cost flow simulations of wind farms. Actuator Line rotor model was studied and then implemented in the SimCenter's in-house code, *Tenasi*. A new Parabolized Navier-Stokes method in conjunction with a suitable wind turbine model was developed and implemented to simulate entire wind farms.

Actuator Line method is an approximation or a model for incorporating the blades' effects in a simulation and was developed to keep the grid size small and avoid very small sized cells that might limit the time step size. As the AL model has matured, the need for modeling the hub, nacelle and tower has been realized. Various methods/models are now available and have been discussed and/or investigated in this study.

In this study, an attempt has been made to synthesize all the information on determining the values of AL parameters required for conducting a simulation. Grid dependent solutions are an issue with AL simulations. This is due to the fact that blades are not a part of the simulation and are modeled using a BEM method. An attempt was made to better understand this grid dependency and reduce the sensitivity of the solutions. A few refinements/potential improvements to the AL model (determining velocity field for aerodynamic force calculation and force distribution function) are investigated in this study and, in the limited scope of the work conducted here, have shown potential. An exhaustive investigation would benefit the research community.

The present Parabolized Navier-Stokes (PNS) model has been validated using several test cases. The PNS model was verified for a wind turbine by comparing the computed results with the RANS solutions.

There are at least two advantages of the PNS model when compared with the existing parabolic models. First, since pressure is a dependent variable, it is computed throughout the flow field and does not require additional simulation or experimental data for modeling the near-wake region. Second, since the wind turbine is modeled as an Actuator Disk, the simulation procedure can continue through the turbine without the need to know the velocity downstream of the turbine experimentally. All the PNS simulations are and can be carried out on a single core of a desktop computer.

The PNS model is an attractive choice for wind farm simulations due to high fidelity and low cost of the model. It can be used for optimization purposes also as demonstrated in this study. Atmospheric Boundary Layer physics can be incorporated in the PNS model in a straight-forward manner. A neutral ABL velocity profile case is discussed in this study. An equation for temperature/energy can be added easily for stable and unstable stratification conditions.

Turbulence modeling has turned out to be an issue. Typical turbulence models for RANS equations do not work with the PNS because of the absence of strong gradients which arise only near a viscous wall. A modified turbulence model is required that accounts for rotor generated and atmospheric turbulence for an Actuator Disc type model of a wind turbine. An appropriate turbulence model would provide accurate wake dissipation and velocity recovery characteristics in varying atmospheric conditions, thus, making the PNS model a great tool for analysis of large wind farms.

The PNS implementation for structured grids was found to be more efficient in terms of both the memory utilization and the runtime. One of the bottlenecks in the PNS model is the amount of time spent in the linear solve routine. In the present study, Symmetric Gauss-Seidel (SGS) algorithm is utilized. A better and faster solver would potentially reduce the runtime of the PNS method even further.

REFERENCES

[1] "Wind in Numbers," *Global Wind Energy Council* Available: http://www.gwec.net/global-figures/wind-in-numbers/.

[2] Jensen, N. O., *A note on wind generator interaction*, Roshkilde, Denmark: 1983.

[3] Katic, I., Højstrup, J., and Jensen, N. O., "A Simple Model for Cluster Efficiency," *EWEC 86*, W.C.O.T.E.C. Palz and E.I.N.E.B. Sesto, eds., A. Raguzzi, Bookshop for Scientific Publications, 1986, pp. 407–410.

[4] Ishihara, T., Yamaguchi, A., and Fujino, Y., "Development of a New Wake Model Based on a Wind Tunnel Experiment," *Global Wind Power*, 2004.

[5] Frandsen, S. T., Barthelmie, R. J., Rathmann, O., Jorgensen, H. E., Badger, J., Hansen, K., Ott, S., Re, Larsen, S. E., and Jensen, L. E., "The making of a second-generation wind farm efficiency model-complex," *Wind Energy*, vol. 12, 2009, pp. 445–458.

[6] Mikkelsen, R., "Actuator disc methods applied to wind turbines," 2003.

[7] Sørensen, J. N., and Shen, W. Z., "Numerical Modeling of Wind Turbine Wakes," *Journal of Fluids Engineering*, vol. 124, May. 2002, pp. 393–399.

[8] Ainslie, J. F., "Wake Modelling and Prediction of Turbulent Properties," *Proceedings of the 8th BWEA Wind Energy Conference*, 1986, pp. 115–119.

[9] Ainslie, J. F., "Calculating the Flowfield in the Wake of Wind Turbines," *Journal of Wind Engineering and Industrial Aerodynamics*, vol. 27, 1988, pp. 213–224.

[10] Crespo, A., Manuel, F., Moreno, D., Fraga, E., and Hernandez, J., "Numerical Analysis of Wind Turbine Wakes," *Proceedings Delphi Workshop On Wind Energy Applications*, G. Bergeles and J. Chadjivassiliadis, eds., 1985, pp. 15–25.

[11] Jonkman, J. M., and Buhl Jr., M. L., "FAST User's Guide," NREL/EL-500-29798, Golden, CO, 2005.

[12] Martinez, L. A., Leonardi, S., Churchfield, M. J., and Moriarty, P. J., "A Comparison of Actuator Disc and Actuator Line Wind Turbine Models and Best Practises for their Use," *50th AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition*, Nashville: 2012.

[13] Troldborg, N., Zahle, F., Réthoré, P. E., and Sørensen, N. N., "Comparison of the wake of different types of wind turbine CFD models," *50th AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition*, Nashville, TN: 2012.

[14] Churchfield, M. J., Lee, S., Moriarty, P. J., Martinez, L. A., Leonardi, S., Vijayakumar, G., and Brasseur, J. G., "A Large-Eddy Simulation of Wind-Plant Aerodynamics," *50th AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition*, Nashville: 2012.

[15] Shives, M., and Crawford, C., "Mesh and load distribution requirements for actuator line CFD simulations," *Wind Energy*, vol. 16, Nov. 2013, pp. 1183–1196.

[16] Jha, P., Churchfield, M., Moriarty, P., and Schmitz, S., "Accuracy of State-of-the-Art Actuator-Line Modeling for Wind Turbine Wakes," *51st AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition*, American Institute of Aeronautics and Astronautics, 2013.

[17] Schlichting, H., and Gersten, K., *Boundary-Layer Theory*, Springer-Verlag Berlin Heidelberg, 2000.

[18] Creech, A., Früh, W.-G., and Maguire, A. E., "Simulations of an offshore wind farm using large eddy simulation and a torque-controlled actuator disc model," *arXiv preprint arXiv:1410.3650*, 2014.

[19] Creech, A. C. W., Früh, W.-G., and Maguire, A. E., "High-resolution CFD modelling of Lillgrund Wind farm," *International Conference on Renewable Energies and Power Quality, Bilbao (Spain)*, 2013.

[20] Sitaraman, J., Mavriplis, D. J., and Duque, E. P., "Wind Farm simulations using a Full Rotor Model for Wind Turbines," *32nd ASME Wind Energy Symposium*, American Institute of Aeronautics and Astronautics, 2014.

[21] Sforza, P. M., Sheerin, P., and Smorto, M., "Three dimensional wakes of simulated wind turbines," *AIAA Journal*, vol. 19, 1981, pp. 1101–1107.

[22] Sforza, P. M., Stasi, W., Smorto, M., and Sheerin, P., "Wind turbine generator wakes," *AIAA Aerospace Sciences Meeting*, 1979.

[23] Schepers, J. G., and Pijl, S. P. Van Der, "Improved modelling of wake aerodynamics and assessment of new farm control strategies," *Journal of Physics Conference Series*, vol. 75, 2007, p. 012039.

[24] Churchfield, M., and Lee, S., "NWTC Design Codes (SOWFA)" Available: http://wind.nrel.gov/designcodes/simulators/SOWFA/.

[25] Mittal, A., Sreenivas, K., Taylor, L. K., Hereth, L., Hilbert, C. B., and Hyams, D. G., "Investigation of Rotor Models for Wind Turbine Simulations," *32nd AIAA Applied Aerodynamics Conference*, American Institute of Aeronautics and Astronautics, 2014.

[26] Churchfield, M. J., Lee, S., Schmitz, S., and Wang, Z., "Modeling Wind Turbine Tower and Nacelle Effects within an Actuator Line Model," *33rd Wind Energy Symposium*, Orlando: American Institute of Aeronautics and Astronautics, 2015.

[27] Troldborg, N., "Actuator Line Modeling of Wind Turbine Wakes," Technical University of Denmark, 2009.

[28] Nodeland, A. M. I., "Wake modelling using an actuator disk model in openFOAM," Institutt for energi-og prosessteknikk, 2013.

[29] Krogstad, P.-Å., and Eriksen, P. E., "'Blind test' calculations of the performance and wake development for a model wind turbine," *Renewable Energy*, vol. 50, Feb. 2013, pp. 325–333.

[30] Sreenivas, K., Hilbert, C. B., Mittal, A., Hereth, L., and Taylor, L. K., "High-Fidelity Computational Simulation of the Wake Characteristics of a Model Wind Turbine," 2013.

[31] Mittal, A., Sreenivas, K., Taylor, L. K., Hereth, L., and Hilbert, C. B., "Blade-Resolved Simulations of a Model Wind Turbine: Effect of Temporal Convergence," *accepted at Wind Energy*.

[32] Mittal, A., Sreenivas, K., Taylor, L. K., and Hereth, L., "Improvements to the Actuator Line Modeling for Wind Turbines," *33rd Wind Energy Symposium*, Orlando: American Institute of Aeronautics and Astronautics, 2015.

[33] Briley, W. R., and McDonald, H., "Three-dimensional viscous flows with large secondary velocity," *Journal of Fluid Mechanics*, vol. 144, 1984, pp. 47–77.

[34] Govindan, T. R., Briley, W. R., and McDonald, H., "General Three-Dimensional Viscous Primary/Secondary Flow Analysis," *AIAA Journal*, vol. 29, 1991, pp. 361–390.

[35] Kreskovsky, J. P., Briley, W. R., and McDonald, H., "Investigation of Mixing in a Turbofan Exhaust Duct, Part I: Analysis and Computational Procedure," *AIAA Journal*, vol. 22, Mar. 1984, pp. 374–382.

[36] Hyams, D. G., "An investigation of parallel implicit solution algorithms for incompressible flows on unstructured topologies," Mississippi State University, 2000.

[37] Anderson, W. K., and Bonhaus, D. L., "An Implicit Upwind Algorithm for Computing Turbulent Flows on Unstructured Grids," *Computer and Fluids*, vol. 23, Jan. 1994, pp. 1–21.

[38] "OpenFOAM | The OpenFOAM Foundation" Available: http://openfoam.org/.

[39] Spalart, P., and Allmaras, S., "A one-equation turbulence model for aerodynamic flows," *30th Aerospace Sciences Meeting and Exhibit*, American Institute of Aeronautics and Astronautics, 1992.

[40] Jonkman, J. M., Butterfield, S., Musial, W., and Scott, G., "Definition of a 5-MW Reference Wind Turbine for Offshore System Development," NREL/TP-500-38060, Golden, CO, 2009.

[41] Mittal, A., Briley, W. R., Taylor, L. K., and Sreenivas, K., "A Parabolic Method without Pressure Approximations for Wind Turbines," *33rd Wind Energy Symposium*, American Institute of Aeronautics and Astronautics, 2015.

[42] Mittal, A., Briley, W. R., Sreenivas, K., and Taylor, L. K., "A Parabolic Velocity-Decomposition Method for Wind Turbines," *submitted to Journal of Computational Physics*.

[43] Briley, W. R., "Private Communication," 2014.

[44] Mittal, A., Briley, W. R., Taylor, L. K., and Sreenivas, K., "A Parabolic Method without Pressure Approximations for a Wind Farm," *EWEA Offshore 2015*, Copenhagen: EWEA, 2015.

[45] Mittal, A., Sreenivas, K., Briley, W. R., and Taylor, L. K., "A Parabolic Method for Accurate and Efficient Wind Farm Simulation," *33rd AIAA Applied Aerodynamics Conference*, American Institute of Aeronautics and Astronautics, 2015.

[46] Menter, F. R., "Two-Equation Eddy-Viscosity Turbulence Models for Engineering Applications," *AIAA*, vol. 32, 1994, pp. 1598–1605.

[47] Mittal, A., Taylor, L. K., Sreenivas, K., Briley, W. R., and Nichols, D. S., "Extension of a Parabolic Method without Pressure Approximations for Wind Turbines in ABL Flows," *33rd AIAA Applied Aerodynamics Conference*, American Institute of Aeronautics and Astronautics, 2015.

VITA

Anshul Mittal was born in Jaipur, Rajasthan, India. He completed high school education in Jaipur. He obtained a Bachelor's Degree in Mechanical Engineering from the Indian Institute of Technology Roorkee in 2007. Anshul Mittal then came to USA to pursue Master's Degree in Mechanical Engineering from the Case Western Reserve University in Cleveland, Ohio. Soon after graduating in 2010, he joined the SimCenter to pursue a Doctor of Philosophy in Computational Engineering. The degree of Doctor of Philosophy was awarded in December 2015.