University of Cincinnati

Date: 5/3/2011

I, Soumitr Dev, hereby submit this original work as part of the requirements for the degree of Master of Science in Mechanical Engineering.

It is entitled:
Wind Turbine Blade Design System - Aerodynamic and Structural Analysis

Student's name: Soumitr Dev

This work and its defense approved by:

Committee chair: Mark Turner, ScD

Committee member: Shaaban Abdallah, PhD

Committee member: Urmila Ghia, PhD

Committee member: Milind Jog, PhD
Wind Turbine Blade Design System – Aerodynamic and Structural Analysis

A Thesis submitted to the
Graduate School
of the University of Cincinnati
in partial fulfillment of the
requirements for the degree of

Master of Science

in the School of Dynamic Systems
of the College of Engineering and Applied Sciences
by

Soumitr Dey

B.E., Anna University
May, 2005

Committee Chair: Dr. Mark. G. Turner Sc.D
Abstract

The ever increasing need for energy and the depletion of non-renewable energy resources has led to more advancement in the "Green Energy" field, including wind energy. An improvement in performance of a Wind Turbine will enhance its economic viability, which can be achieved by better aerodynamic designs. In the present study, a design system that has been under development for gas turbine turbomachinery has been modified for designing wind turbine blades. This is a very different approach for wind turbine blade design, but will allow it to benefit from the features inherent in the geometry flexibility and broad design space of the presented system. It starts with key overall design parameters and a low-fidelity model that is used to create the initial geometry parameters. The low-fidelity system includes the axisymmetric solver with loss models, T-Axi (Turbomachinery-AXIsymmetric), MISES blade-to-blade solver and 2D wing analysis code XFLR5. The geometry parameters are used to define sections along the span of the blade and connected to the CAD model of the wind turbine blade through CAPRI (Computational Analysis PRogramming Interface), a CAD neutral API that facilitates the use of parametric geometry definition with CAD. Either the sections or the CAD geometry is then available for CFD and Finite Element Analysis.

The GE 1.5sle MW wind turbine and NERL NASA Phase VI wind turbine have been used as test cases. Details of the design system application are described, and the resulting wind turbine geometry and conditions are compared to the published results of the GE and NREL wind turbines. A 2D wing analysis code XFLR5, is used for to compare results from 2D analysis to blade-to-blade analysis and the 3D CFD analysis. This kind of comparison concludes that, from hub to 25% of the span blade to blade effects or the cascade effect has to be considered, from 25% to 75%, the blade acts as a 2d wing and from 75% to the tip 3D and tip effects have to be taken into account for design considerations. In addition, the benefits of this approach for wind turbine design and future efforts are discussed.
Acknowledgments

The author would like to thank his thesis and research advisor, Dr. Mark G. Turner, for his guidance and endless supply of ideas. I would also like to thank Dr. Milind Jog for being my graduate advisor and a member of my defense committee. I would also like to thank Dr. Urmila Ghia and Dr. Shaaban A. Abdallah for accepting to be on my defense committee. I realize that you have a busy schedule, making this type of time commitment very difficult.

The author would like to thank NASA for funding and support the research project through MDAO of Turbomachinery with Emphasis on Component Optimization. Additional thanks goes to the research team at Avectec for their organization of the MDAO project meetings and telecoms. A special thanks to Kevin Park for helping me out on numerous occasions with FORTRAN coding. Thanks to Rob Ogden for helping me troubleshoot many license and system issues. I would like to thank Kiran Siddapaji and Kedarnath Sairum (KD) for all the brainstorming sessions for my research. Thanks to Robert Kanapke for providing support and help in using Fine Turbo and ASGARD. Special thanks to other members of the UC Gas Turbine Simulation Laboratory for their help and support.

I would like to thank Dr. Ali Merchant for his support on troubleshooting CAPRI issues and providing know how for development of codes on top of CAPRI interface.

I would also like to thank the NUMECA support team (Roque Lopez and Autumn Fjeld) for helping me out numerous times with Fine Turbo troubleshooting.

Thanks to my MOM and DAD for all of their support.
# Contents

Abstract

Acknowledgments

Contents

List of Tables

List of Figures

Nomenclature

1 Introduction

1.1 Motivation

1.2 Overview

2 Literature Review

2.1 Historical Background

2.2 Recent Work

3 Velocity Triangles

4 Axisymmetric solver - T-Axi

5 Wind Turbine Design Using T-Axi

6 Cascade Analysis using MISES

6.1 Issues with Geometry Definition

6.2 Wind Turbine Design and Analysis using MISES
7 Wing Analysis using XFLR5

8 3D-CFD Analysis using Fine Turbo
   8.1 Case setup ................................................................. 41
   8.2 Grid Generation using AutoGrid and Wind Turbine module in Fine/Turbo .................. 42
      8.2.1 Quality Criterion Definitions .................................. 48
      8.2.2 Boundary Conditions ............................................. 49
      8.2.3 Flow solver - EURANUS ......................................... 50
   8.3 Convergence History .................................................... 52
   8.4 Grid-Dependency Study .................................................. 53
   8.5 Results ................................................................. 54

9 FEA stress and Modal Analysis ................................. 64

10 Conclusion .............................................................. 68
   10.1 Summary ............................................................... 68
   10.2 Future Work ........................................................... 68

References ................................................................. 70

Appendix ........................................................................ 74

A Geometry Development .............................................. 75
   A.1 Parametric Geometry Development using CAPRI ......................... 75
   A.2 One-dimensional Grid Clustering ................................... 75

B Drive_spline Code ....................................................... 78
   B.1 C Program ................................................................. 78

C Read_din Code .......................................................... 93
   C.1 Fortran Program ......................................................... 93

D ANSYS Code ............................................................. 103
   D.1 Fortran Program ....................................................... 103

E Output ....................................................................... 113
   E.1 Sample Output .......................................................... 113
F ANSYS Output File 124
   F.1 Fortran Program .................................................. 124

G INIT file 137
   G.1 TTDES Input ....................................................... 137

H STAGE file 138
   H.1 TTDES Input ....................................................... 138

I TTDES-Results file 139
   I.1 TTDES - Results .................................................. 139

J TAXI Input 141
   J.1 stack ............................................................... 141
   J.2 Walls ............................................................. 143

K geomturbo_creator Code 144
   K.1 Fortran Program .................................................. 144

L MISES Input files 152
   L.1 Blade File ....................................................... 152
   L.2 Ises File .......................................................... 154

Index 155
List of Tables

4.1 INIT file. ................................................................. 16
4.2 Stage file. ................................................................. 16

5.1 GE 1.5sle MW Technical Specifications ([17]). .................................................. 22
5.2 GE wind turbine derived data input for TTDES. .................................................. 22
5.3 NREL Phase VI wind turbine specifications([31]). ................................................ 23
5.4 NREL Phase VI wind turbine derived data input for TTDES. ................................ 23
5.5 T-Axi Turbine parameter results for GE reverse engineered wind turbine blade. .... 27
5.6 T-Axi Turbine parameter results for NREL reverse engineered wind turbine blade. ... 27

6.1 $C_y$, $C_x$ values of different profiles from MISES at 14° at mid span. ................. 34
6.2 $C_l$, $C_d$ values of different profiles from MISES at 14° at mid span. .................. 34
6.3 Comparison between transition locations for NREL-S809 profile from MISES at 14° at mid span, using AGS and $e^{ni}$ modes. .................................................. 35
6.4 Comparison between lift and drag coefficient for NREL-S809 profile from MISES at 14° at mid span, using AGS and $e^{ni}$ modes. .................................................. 35

8.1 Grid quality report values ................................................................. 49
8.2 Number of grid points for each grid configuration................................................ 53
8.3 $Y^±$ range for each grid configuration.................................................. 54
8.4 Power output from different grid arrangements................................................ 54
8.5 Loss model values at mid span for different grid levels(NUMECA-3D). .............. 55
8.6 Loss model values at mid span.................................................. 58

9.1 First five natural Frequencies in parked condition(GE reverse engineered blade). ...... 67
# List of Figures

1.1 Lift and Drag on a wind-turbine blade profile. ........................................... 2
1.2 Process Flowchart for wind turbine design system. ................................. 3

2.1 Horizontal Axis Wind Turbine([52]). ............................................................ 7
2.2 Vertical Axis Wind Turbine([52]). ............................................................... 8

3.1 velocity triangle ......................................................................................... 11
3.2 Nomenclature of a wind turbine blade profile. ........................................... 12
3.3 Nomenclature of an axial turbine blade profile. ......................................... 12
3.4 Velocity triangle used in- T-Axi. ................................................................. 13
3.5 Velocity triangle used in- T-Axi. ................................................................. 13
3.6 Velocity triangle of wind turbine (x shown on the axis should be Z)([25]). ....... 13
3.7 Velocity triangle at rotor plane(from[24]). .................................................. 14
3.8 Velocity triangle for downstream and upstream direction (from[24]). .......... 14

4.1 T-Axi Flow Chart. ...................................................................................... 18
4.2 T-Axi screen capture during solution of 5 stage EEE LPT. ......................... 19
4.3 3D representation of a 5 Stage EEE LPT. .................................................. 20

5.1 Power curve -GE 1.5sle MW(from [17]). .................................................... 22
5.2 Variation of \( rV_\theta \) vs radial location for GE- reverse engineered wind turbine. ... 24
5.3 Variation of \( rV_\theta \) vs radial location for NREL- reverse engineered wind turbine. ... 25
5.4 Comparison of \( c/r \) vs radial locations for GE-reverse engineered wind turbine blade to Griffin report([3]). ......................................................... 26
5.5 Comparison of \( c/r \) vs radial locations for NREL-reverse engineered wind turbine blade to NREL report([33]). .................................................. 26
5.6 CAD Blade Design (NREL Phase VI blade) [31]. ...................................... 27
<table>
<thead>
<tr>
<th>Section</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>5.7</td>
<td>NREL reverse engineered Wind turbine Blade</td>
<td>28</td>
</tr>
<tr>
<td>5.8</td>
<td>GE Windturbine Blade([17])</td>
<td>28</td>
</tr>
<tr>
<td>5.9</td>
<td>GE reverse engineered Windturbine Blade</td>
<td>28</td>
</tr>
<tr>
<td>6.1</td>
<td>Comparison of blade profiles at mid span- NUMECA-3D and NREL-S809.</td>
<td>30</td>
</tr>
<tr>
<td>6.2</td>
<td>Blade profile of NUMECA-3D at midspan.</td>
<td>30</td>
</tr>
<tr>
<td>6.3</td>
<td>MISES initial settings.</td>
<td>31</td>
</tr>
<tr>
<td>6.4</td>
<td>$C_l$ vs Pitch/Chord.</td>
<td>32</td>
</tr>
<tr>
<td>6.5</td>
<td>Pitch/Chord vs $r/r_{tip}$.</td>
<td>33</td>
</tr>
<tr>
<td>6.6</td>
<td>Profile comparisons at mid span.</td>
<td>33</td>
</tr>
<tr>
<td>6.7</td>
<td>$C_p$ comparisons at mid span from MISES.</td>
<td>34</td>
</tr>
<tr>
<td>6.8</td>
<td>$C_p$ comparisons at mid span from MISES, for S809-NREL using AGS and Fully-Turbulent modes.</td>
<td>35</td>
</tr>
<tr>
<td>6.9</td>
<td>Shape factor plot from MISES at mid span.</td>
<td>36</td>
</tr>
<tr>
<td>6.10</td>
<td>$Re_\theta$ plots at mid span.</td>
<td>37</td>
</tr>
<tr>
<td>6.11</td>
<td>Co-efficient of friction plots at mid span.</td>
<td>38</td>
</tr>
<tr>
<td>7.1</td>
<td>$C_p$ comparisons at mid span from XFLR5.</td>
<td>39</td>
</tr>
<tr>
<td>7.2</td>
<td>$C_p$ comparisons at mid span between XFLR5 and MISES ($e^n$).</td>
<td>40</td>
</tr>
<tr>
<td>8.1</td>
<td>Basic nomenclature in Autogrid.</td>
<td>42</td>
</tr>
<tr>
<td>8.2</td>
<td>Geometry import into Autogrid.</td>
<td>42</td>
</tr>
<tr>
<td>8.3</td>
<td>Turbomachinery domain.</td>
<td>43</td>
</tr>
<tr>
<td>8.4</td>
<td>Computational Domain.</td>
<td>44</td>
</tr>
<tr>
<td>8.5</td>
<td>Wizard mesh topology [37].</td>
<td>44</td>
</tr>
<tr>
<td>8.6</td>
<td>Computational grid at mid-span.</td>
<td>44</td>
</tr>
<tr>
<td>8.7</td>
<td>Far field height.</td>
<td>45</td>
</tr>
<tr>
<td>8.8</td>
<td>B2B mesh mid span.</td>
<td>46</td>
</tr>
<tr>
<td>8.9</td>
<td>Flow path definition.</td>
<td>47</td>
</tr>
<tr>
<td>8.10</td>
<td>Grid quality report.</td>
<td>48</td>
</tr>
<tr>
<td>8.11</td>
<td>Aspect ratio [37].</td>
<td>49</td>
</tr>
<tr>
<td>8.12</td>
<td>Expansion ratio [37].</td>
<td>49</td>
</tr>
<tr>
<td>8.13</td>
<td>Periodic Boundary.</td>
<td>50</td>
</tr>
<tr>
<td>8.14</td>
<td>Euler wall condition.</td>
<td>50</td>
</tr>
</tbody>
</table>
# Nomenclature

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>Cross-sectional area</td>
</tr>
<tr>
<td>a</td>
<td>Axial induction factor</td>
</tr>
<tr>
<td>a'</td>
<td>Angular induction factor</td>
</tr>
<tr>
<td>$a_0$</td>
<td>Speed of sound</td>
</tr>
<tr>
<td>b</td>
<td>Stream tube thickness</td>
</tr>
<tr>
<td>C</td>
<td>Absolute velocity downstream</td>
</tr>
<tr>
<td>$C_a$</td>
<td>Axial component of velocity</td>
</tr>
<tr>
<td>$C_\theta$</td>
<td>Tangential component of velocity</td>
</tr>
<tr>
<td>$C_d$</td>
<td>Coefficient of drag</td>
</tr>
<tr>
<td>$C_f$</td>
<td>Skin friction coefficient</td>
</tr>
<tr>
<td>$C_l$</td>
<td>Coefficient of lift</td>
</tr>
<tr>
<td>$c_p$</td>
<td>Coefficient of performance, Coefficient of power</td>
</tr>
<tr>
<td>$C_p$</td>
<td>Coefficient of pressure</td>
</tr>
<tr>
<td>$C_p$</td>
<td>Specific heat constant pressure = 1.005 kJ/Kg K.</td>
</tr>
<tr>
<td>$C_x$</td>
<td>Force in x direction/(mass V1)</td>
</tr>
<tr>
<td>$C_y$</td>
<td>Torque about y/(mass r1 V1) = d(r V\theta)/(r1 V1)</td>
</tr>
<tr>
<td>$c_x$</td>
<td>Axial Chord</td>
</tr>
<tr>
<td>D</td>
<td>Diameter</td>
</tr>
</tbody>
</table>
\( D_h \)  Hydraulic Diameter

\( F \)  Force

\( f \)  Frequency (Hz)

\( H \)  Shape factor

\( \tilde{i} \)  Angle of incidence

\( M \)  Mach number

\( \dot{m} \)  Mass flow rate

\( m' \)  \( \frac{\dot{m}}{r} \)  (2-D cascade)

\( n \)  number of blades

\( P \)  Rated Power

\( P_0 \)  Total Pressure

\( PR \)  Total Pressure Ratio

\( P_s \)  Static Pressure

\( p \)  pitch = \( \frac{2\pi r}{n} \)

\( Q \)  Differential Torque

\( R \)  Specific gas constant

\( Re \)  Reynolds number

\( Re_\theta \)  Reynolds number based on momentum thickness

\( Re_D \)  Reynolds number based on hydraulic diameter

\( rV_\theta \)  Angular momentum

\( S \)  Span = \( \frac{r - r_{hub}}{r_{tip} - r_{hub}} \)

\( s \)  Blade Spacing = \( \frac{2\pi r}{n} \)

\( T \)  Torque
TR  Total Temperature Ratio
Th  Thrust
Ts  Static Temperature
TT  Total Temperature
U   Rotor Wheel speed
\vec{U}  Rotor Wheel speed vector
u   Axial Velocity
\mean{u}{+}  mean axial velocity / shear velocity
\vec{V}  Axial velocity vector
Vz   Axial velocity
V\theta  Tangential velocity
Vr   Radial velocity
V_{rel,1}  Relative velocity upstream
V_{rel,2}  Relative velocity downstream
V_{rot}  Rotational Velocity
W   Relative Velocity
\vec{W}  Relative Velocity vector
Z   Zweifel Coefficient = 2 \times \frac{\tan \alpha_1 - \tan \alpha_1 \cos^2 \alpha_2}{\cos \alpha_2}
z   Axial direction

Greek Symbols
\alpha   Absolute flow angle, also airfoil angle of attack
\beta   Relative flow angle
\tilde{\beta}   Relative Flow angle
ψ  work coefficient = \( \frac{dh}{2U^2} \)

\( \phi \)  Flow coefficient = \( \frac{V_z}{U} \)

\( \tau_w \)  Wall shear stress

\( \theta \)  Tangential direction

\( \gamma \)  Ratio of specific heats

\( \eta \)  Efficiency

\( \rho \)  Density

\( \omega \)  Rotor angular Velocity

\( \lambda \)  Tip speed ratio

\( \zeta \)  Total Pressure loss coefficient \( \frac{P_{in} - P_{exit}}{P_{in} - P_{exit}} \)

\( \zeta \)  Zeta or Energy or Enthalphy loss coefficient \( \frac{P_{rel in} - P_{rel exit}}{P_{rel in} - P_{rel exit}} \)

\( \tilde{\zeta} \)  Stagger angle

\( \phi \)  Meridional flow angle = \( \text{atan} \frac{V_r}{V_z} \)

\( \psi \)  Twist angle

\( \mu \)  Viscosity of fluid

\( \nu \)  Kinematic viscosity of the fluid. \((m^2/s)\)

3DBGB-NREL  Blade section generated through 3DBGB code with default sections.

NUMECA-3D  Blade profile from NUMECA geometry.

NREL-S809  Blade profile from NREL NASA Phase VI airfoil report of S809.

Annotation

\( \tilde{P}_{sg} \)  Normalised quantity

\( \bar{\tau} \)  Area Averaged quantity

\( \tau \)  Mass Averaged quantity
Chapter 1

Introduction

The limited availability of non-renewable natural resources, changes in environmental conditions, and increasing energy costs, have led to an increase in investments in the renewable energy sector. The 1973 oil crisis suddenly created interest in wind energy for many countries intending to reduce dependency on oil imports. Various countries started national research programs for development of wind turbines. The National Wind Technology Center (NWTC), which is a part of National Renewable Energy Lab (NREL), is one such organization in United states, which was instituted for the technical development and large-scale deployment of wind power in the US. The investment since the 90’s in the wind turbine industry has been more than US $1 trillion [24],[5]. This investment has increased by approximately 20% every year since then [5] . [52].

The important question that arises from the above discussion is “Why use Wind Power”? Some of the obvious answers that come to mind are, wind is a free resource and, unlike coal or natural gas, it does not have to be imported. The increase in carbon and green house gas levels in the environment, has led to climatic changes, green-house effects, and depletion of ozone layers. This gave rise to the concept of reduction of carbon footprint and had led to what is called the implementation of “Carbon-Tax”. Europe has been one the few continents to successfully implement this tax. This led people to move away from convectional methods of power generation using fossil fuels to generating power from renewable energy resources and make them more efficient. The wind energy generation sector has also helped in the creation of many job opportunities. Given all of the above advantages of wind energy, it has some drawbacks. Important facts are how much of the wind energy available can be converted into useful power and the cost of production. In a highly windy site, this cost ($/KWh) is comparable to the production cost from non-renewable resources, but the same is not true for other areas. Thus, efforts are being made to harness the available wind energy in the most effective way.
Wind turbines are mounted at a height of 80 feet or more above the ground level where they can take advantage of the faster and less turbulent winds. Wind turbines tap the wind’s energy with their large blades, converting the kinetic energy available in the wind to a useful form of energy. Usually two or three blades are mounted on a shaft to form a rotor. When the wind blows, low-pressure forms on the downwind side of the blade. The low-pressure causes the rotor to turn through lift on the airfoil. The force of lift is much stronger than drag, which is the wind’s force against the front side of the blade. The combined lift and drag causes the rotor to spin along with a generator to generate electricity [30]. Fig. 1.1 shows the lift and drag pictorially on a wind turbine blade profile as explained.

Figure 1.1: Lift and Drag on a wind-turbine blade profile.

1.1 Motivation

Because of the increasing need for energy and to reduce the need for non-renewable energy resources, efforts are being made to utilize renewable energy to a great extent. Wind energy is one such abundant resource, and huge efforts are under way to make the available wind turbines more efficient and more economical to operate.

The wind turbine efficiency improvement can be subdivided into the following areas:

• Aerodynamic design of the Wind Turbine

• Control system design

• Mechanics and Dynamics

• System design and control
• Energy system Economics

In the research presented, an attempt has been made to develop a wind turbine blade design geometry system, keeping in mind the aerodynamics and structural aspects of a wind turbine blade. The access to the source code of T-AXI [50], [49], [8], an axi-symmetric design tool for complete turbomachinery geometry design, made it easier to modify parameters and develop a blade geometry modeler 3DBGB [45] for generating blade profiles for analysis. Thus, a turbomachinery approach to wind turbine blade design system is proposed and discussed in the current research. The design process starts by bootstrapping flow parameters into a low fidelity tool T-AXI and generating blade profiles by using 3DBGB. CAPRI [22], [32] embedded in the Drive_spline code was used to make the geometry consistent with a CAD package. The blade geometry thus produced was analyzed in MISES and high fidelity 3D-CFD tool FINE/Turbo. A script written as a subroutine to the main code, Drive_spline, generates an ANSYS file(ANSYS.ain), which automates the generation of a meshed part, ready for any kind of FEA study.

The flow chart for the process followed during this approach is shown Fig. 1.2.

![Figure 1.2: Process Flowchart for wind turbine design system.](image-url)
1.2 Overview

This thesis presents a turbomachinery approach for wind turbine blade design for horizontal axis wind turbines. Discussions on the process, shown in Fig. 1.2, is detailed in Chapter 5. This chapter also includes the details for usage of T-AXI as a design tool for wind turbine blade design. Comparison of geometry available in literature is also presented in the same chapter. A brief discussion on T-AXI as a turbine design tool is included in Chapter 4. In Chapter 6, the use of MISES to analyze the blade profiles is detailed, and a comparison of aerodynamic data available is made, to show where cascade effects matters in such kind of machines. In Chapter 7, the use of a wing analysis code, XFLR5 for wind turbine blade is explained and a 2D application of wind turbines is explored. Chapter 8 explains the 3D CFD analysis details using Fine/Turbo. Chapter 9 discusses modal analysis of wind turbine blades and FEA results for the blade model generated through 3DBGB. The final Chapter 10 includes a summary and future directions.
Chapter 2

Literature Review

Blade Design and Analysis is one of the most complicated and important aspects of current wind turbine technology. Today engineers strive to design blades that extract as much energy from the wind as possible throughout a range of wind speeds and gusts, yet be durable, quiet and economical, as quoted by KID wind Project with National Renewable Energy Laboratory (NREL) [40].

2.1 Historical Background

Wind Turbine design is reported to date back to ancient Persia [20]. The first recorded wind mills built were made by the Persians in 900 A.D. and they were Horizontal Axes wind turbines. Wind Energy made its appearance in Europe during the middle ages [30]. In the 19th century, the interest of harnessing the power of wind grew and developments were made for electricity generation from wind. Sir William Thomson [48] in 1881 stated, in his address on energy sources to mathematical and physical science section of the British Association for the Advancement of science, to use wind mills to drive a generator to charge storage batteries as accumulators [46]. The development of airplanes in the mid twentieth century for advancements in design and analysis of better propellers was a huge benefit to the wind turbine advancements in terms of the better blade design for wind turbines. During this time, Professor Albert Betz developed the Betz Limit [6] pertaining to the energy in wind that could be achievable.

Design and development of wind turbine blades has been performed by a number of authors, who have derived methods for predicting the airfoil shapes, profiles, and characteristics for the same. Betz [6] and Glauert [19],[18] had originally developed the classical analysis of horizontal wind turbines. Subsequently, Wilson and Lissaman [41],[42], Wilson et al.[43], and de Varies[11] developed computerized codes for this theory. The Momentum theory and Blade element theory combined formed the backbone
of all the calculations made in these codes. Momentum theory, as the name suggests, is based on the control volume analysis of the forces on the blade based on the conservation of angular and linear momentum. Equations are derived for thrust and differential torque from conservation of linear momentum and angular momentum respectively [30]:

\[ d\bar{Th} = \rho U^2 4a(1 - a)\pi rdr \]  \hspace{1cm} (2.1)

\[ d\bar{Q} = 4a'(1 - a)\rho U\pi r^3\Omega dr \]  \hspace{1cm} (2.2)

Blade element theory deals with the lift and drag coefficients and angle of attack on the wind turbine blades [30]. This analysis assumes that

- There is no aerodynamic interaction between elements
- The forces on the blades are determined solely by lift and drag of the airfoil shapes of the blade

Wind Turbines can be broadly classified under two main systems, according to their orientation as Horizontal-Axis Wind Turbines (HAWTs) (see Fig. 2.1) and Vertical Axis Wind Turbines (VAWTs) (see Fig. 2.2). Furthermore, the HAWTs are classified with respect to their rotor orientation (Upwind or Downwind), Blade articulation (rigid or teetering) and number of blades on the rotor (two or three) [30]. The present research work is concentrated on the design, development and analysis of Horizontal-axis wind turbine blades.
Figure 2.1: Horizontal Axis Wind Turbine([52]).
Figure 2.2: Vertical Axis Wind Turbine([52]).
2.2 Recent Work

A number of performance codes have been developed to increase the efficiency of horizontal axis wind turbines. These codes were mostly developed by the National Energy Laboratory in Boulder, CO, and only a few are available for general use. The aerodynamic codes pertaining to wind turbines developed by NREL are:

- WT_Pref [28]
- YawDyn [23]
- AeroDyn [23]
- FAST_AD [43]

Some authors such as Hansen and Butterfield [1] have also been inspired by the helicopter industry to use the Vortex wake methods, to understand the 3-dimensional boundary layer effects on the horizontal axis wind turbines. Computational fluid dynamics, analysis of the wind turbines has been performed by many authors, of which work done by Sorenson and Michelsen[35], and Duque et al.[39], are notable. Selig and Tangler[34] have developed the code PROPID to perform the blade design of the wind turbines from the reverse approach, that is inputing the aerodynamic quantities and then getting out the blade shapes and corresponding geometry from it. This code has been used by the wind turbine industry to successfully design wind turbine blades. In the present research, a similar reverse engineering method is followed, i.e, developing blade shapes through analyzing the aerodynamic quantities associated with wind turbines.
Chapter 3

Velocity Triangles

The analysis of the effects of the blade rows on the airflow is determined by velocity triangles. Figure 3.1 shows a schematic view of a single stage axial turbine. A wind turbine can be considered as a single stage turbine with no stator row. Thus, the velocity triangle applicable to a single stage axial turbine, can be well compared to a wind turbine velocity triangle. Figures 3.2 and 3.3 show the nomenclature used for wind turbine blades and the comparison to an axial turbine blade. The velocity triangle of a wind turbine is shown in Figures 3.4 and 3.5, where ‘z’ is used to describe the axial and ‘r’ for radial axises respectively. Figure 3.6 shows the velocity triangle as described in Ingram’s[25] book for a convectional horizontal axis wind turbine blade. The blade speed $\vec{U}$ is drawn according to the wind velocity and the third closing side of the velocity triangle is the relative velocity $\vec{W}$. The relative velocity makes an angle $\beta$ with the blade called the relative flow angle. Thus, applying vector triangles to the above configuration, we have the following equations:

$$\vec{V} = \vec{W} + \vec{U}$$  \hspace{1cm} (3.1)

$$\vec{U} = \vec{\Omega} \times \vec{r} = \omega r \hat{\theta}$$  \hspace{1cm} (3.2)

$$W_\theta = V_\theta - U = V_\theta - \omega r$$  \hspace{1cm} (3.3)

$$\phi = \tan^{-1}\left(\frac{V_r}{V_z}\right)$$  \hspace{1cm} (3.4)

$$(V_m)^2 = (V_z)^2 + (V_r)^2$$  \hspace{1cm} (3.5)

$$\alpha_m = \tan^{-1}\left(\frac{V_\theta}{V_m}\right)$$  \hspace{1cm} (3.6)

$$\beta_m = \tan^{-1}\left(\frac{W_\theta}{V_m}\right)$$  \hspace{1cm} (3.7)
The sign convention as described means that positive rotation is clockwise (aft looking forward). Figures 3.7 and 3.8 show the velocity triangle of the wind turbine blade section downstream and upstream, taken at rotor plane respectively.
Figure 3.2: Nomenclature of a wind turbine blade profile.

Figure 3.3: Nomenclature of an axial turbine blade profile.
Figure 3.4: Velocity triangle used in T-Axi.

Figure 3.5: Velocity triangle used in T-Axi.

Figure 3.6: Velocity triangle of wind turbine (x shown on the axis should be Z)([25]).
Figure 3.7: Velocity triangle at rotor plane (from [24]).

Figure 3.8: Velocity triangle for downstream and upstream direction (from [24]).
Chapter 4

Axisymmetric solver - T-Axi

T-Axi (Turbomachinery Axi-symmetry solver) is an axisymmetric solver developed by Mark Turner, Ali Merchant, Dario Bruna [50][49][8]. Traditional axisymmetric solvers use the streamline curvature method as described by Smith[29] and Novak[36]. T-axi follows the algorithm similar to MISES[13][54]. The code has loss models incorporated and, to make it robust, the loss is smeared out just downstream of the trailing edge. This has a spanwise mixing effect as described in Adkins and Smith[2]. The schematic of the code is shown in Fig. 4.1.

T-T-DES is a 1-D turbine design code for automating the turbine design process. The input files for TTDES are the “INIT” and ’STAGE’ files that are in simple text format and the values generally are created by hand calculation for boot strapping the initial values to T-Axi. As the name suggests, INIT stands for initial values that are used to initialize the calculations along with Stage file that contains more details on the flow parameters per stage. The INIT and Stage file options are enumerated in Tables 4.1 and 4.2. Sample INIT and Stage file used are included in Appendices G and H.

After the initial run using the init and stage file is done, a ttdes-result file is created that has one-dimensional details of the turbine with respect to the physical design aspect of the blade rows specified. The ttdes-result file for a sample case is shown in Appendix I. The accuracy of the computations depends on the initial values specified. Euler’s Turbomachinery equations form the basis for the computations in TTdes. A free vortex assumption, that is keeping \( rV_\theta \) values constant, is made. The T-T-DES code yields a stack and walls file that has the definition of the radial and axial locations of the blade, Wheel speed, \( rV_\theta \), normalized mass flow, Reynolds number, Boundary layer and Viscous flag options. The walls file has the definition of the hub and casing. T-Axi code is developed around multiple interacting streamtube Euler equations, where the axisymmetric equations are discretized in strong conservative form of meridional streamline grid, the same discretization as used in MISES developed by Drela and Youngren [54],
### Table 4.1: INIT file.

<table>
<thead>
<tr>
<th>Units</th>
<th>1-SI: International system</th>
<th>2-EN: English system</th>
</tr>
</thead>
<tbody>
<tr>
<td>Design options</td>
<td>$\alpha_3$</td>
<td></td>
</tr>
<tr>
<td></td>
<td>$\alpha_2$</td>
<td></td>
</tr>
<tr>
<td></td>
<td>M-2</td>
<td></td>
</tr>
<tr>
<td></td>
<td>TT3</td>
<td></td>
</tr>
<tr>
<td>Number Stages</td>
<td>Maximum stage number: 20</td>
<td></td>
</tr>
<tr>
<td>Mass flow rate</td>
<td>SI: [kg/s] - EN: [lbm/s]</td>
<td></td>
</tr>
<tr>
<td>Inlet Total Pressure</td>
<td>SI: [Pa] - EN: [psi]</td>
<td></td>
</tr>
<tr>
<td>Inlet Total Temperature</td>
<td>SI: [K] - EN: [R]</td>
<td></td>
</tr>
<tr>
<td>Alpha 1 (Absolute angle)-First stage</td>
<td>[deg]</td>
<td></td>
</tr>
<tr>
<td>Mach 1-First stage</td>
<td>[-]</td>
<td></td>
</tr>
<tr>
<td>Ratio of specific heats</td>
<td>[-]</td>
<td></td>
</tr>
<tr>
<td>Gas constant</td>
<td>SI: [kJ/(kg K)] - EN: [ft^2/(s^2R)]</td>
<td></td>
</tr>
<tr>
<td>Ratio of clearance</td>
<td>[-]</td>
<td></td>
</tr>
<tr>
<td>Radius if constant hub,mean or tip radii design</td>
<td>SI: [m] - EN: [in]</td>
<td></td>
</tr>
</tbody>
</table>

### Table 4.2: Stage file.

<table>
<thead>
<tr>
<th>Stage Number</th>
<th>[-]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Absolute Angle (Exit)</td>
<td>[-]</td>
</tr>
<tr>
<td>Mach number (Exit)</td>
<td>[-]</td>
</tr>
<tr>
<td>Stator Zweifel number</td>
<td>[-]</td>
</tr>
<tr>
<td>Rotor Zweifel number</td>
<td>[-]</td>
</tr>
<tr>
<td>Stator Phi coefficient</td>
<td>[-]</td>
</tr>
<tr>
<td>Rotor Phi coefficient</td>
<td>[-]</td>
</tr>
<tr>
<td>Stator Aspect Ratio</td>
<td>[-]</td>
</tr>
<tr>
<td>Rotor Aspect Ratio</td>
<td>[-]</td>
</tr>
<tr>
<td>Rotor Axial Velocity Ratio</td>
<td>[-]</td>
</tr>
<tr>
<td>Stator Row space coefficient</td>
<td>[-]</td>
</tr>
<tr>
<td>Rotor Row space coefficient</td>
<td>[-]</td>
</tr>
<tr>
<td>Stator mean Radius</td>
<td>[-]</td>
</tr>
<tr>
<td>Rotor mean Radius</td>
<td>[-]</td>
</tr>
</tbody>
</table>
and Drela and Giles [13].

The stack and walls file are inputs for T-Axi calculations. Also, for making the calculations more accurate and realistic, the values of radial and axial locations are interpolated and input by hand along with the corresponding \( rV_\theta \) values. The \( rV_\theta \) values can be varied for making the results more realistic. The stack and walls files are attached in Appendix J, used in present design process. The Loss models for the turbine included in T-Axi are:

- Profile Loss
- Trailing Edge Loss
- Trailing Edge Shock Loss
- Clearance Loss (This does not adequately account for the flow over the tip of a wind-turbine)
- End Wall Loss
- Secondary Flow Loss

The T-Axi solution results in a blade data file that will be input to the blade geometry modeler ‘3DBGB’, an in-house developed code. The output of 3DBGB are blade profile sections, that was used to create the actual blade geometry. This is then imported to UniGraphics, a CAD tool for creating actual parametric base part files for further analysis. Fig. 4.2 shows a screen capture during execution of the axisymmetric solver, T-AXI. The first chart indicated in the figure, shows the distribution of flow variables such as Mach number, static pressure, stagnation pressure and axial velocity \( (Q_x) \). The second chart in the same figure shows the distribution of angular momentum, as input in the stack file. Fig. 4.3 shows the 3D model of a five stage EEE LPT [9] generated by the 3D viewer of T-Axi. The T-Axi code can be downloaded with its tutorial and relevant technical papers from www.gtsl.ase.uc.edu.
Figure 4.1: T-Axi Flow Chart.
Figure 4.2: T-Axi screen capture during solution of 5 stage EEE LPT.
Figure 4.3: 3D representation of a 5 Stage EEE LPT.
Chapter 5

Wind Turbine Design Using T-Axi

A wind turbine blade can be considered a huge turbine blade without the casing. Thus, the ‘TT-Des’ module of T-AXI is used to initialize turbine blade flow parameters. First, TT-Des is executed with ‘INIT’ and ‘Stage’ files. These are text format files, where flow parameters can be input or changed to get desired results. This allows bootstrapping the calculations initially and values for T-Axi execution is obtained. The details of the flow parameters as published in the technical specification hand book of G.E 1.5sle MW [17] and NREL phase VI blade [31] are used as test cases to reverse engineer the blade shape from these parameters.

The parameters included in the ‘INIT’ file are either in (1)SI or (2)English units. These parameters include Design options, number of stages, mass flow rate, RPM, inlet total pressure, inlet total temperature, inlet angle alpha, inlet Mach number, inlet duct length/N1 which is the axial width ratio (0 in the case of a wind turbine), hub and casing slope upstream (0,0 in the case of wind turbine), ratio of specific heats, gas constant, ratio of clearance/tip radius or clearance/hub radius for all rotors and stators (not the first and last stator). The power vs wind speed graph from the technical specification of GE 1.5sle MW gives the operating power for specific wind speeds, as shown in Fig. 5.1. The power used in all the calculations for the GE reverse engineered wind turbine blade is 0.35MW that is obtained from the power curve (green curve denoting 1.5sle) Fig. 5.1, and for the NREL Phase VI blade is 5.88KW. The parameters taken from the GE specification are shown in Tables 5.1 and 5.2, and NREL specifications are shown in Tables 5.3 and 5.4.

The values of mass flow rate and initial Mach number are calculated using equations (5.1) and (5.2)

\[ \dot{m} = \rho AV_2 \]  

(5.1)
Table 5.1: GE 1.5sle MW Technical Specifications ([17]).

<table>
<thead>
<tr>
<th>Design Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rated Power</td>
<td>1.5 MW</td>
</tr>
<tr>
<td>Power @7m/s</td>
<td>0.35 MW</td>
</tr>
<tr>
<td>Rotor Diameter</td>
<td>77 m</td>
</tr>
<tr>
<td>Average Wind Speed</td>
<td>7 m/s</td>
</tr>
<tr>
<td>Rated Rotor RPM</td>
<td>20.4 rpm</td>
</tr>
<tr>
<td>Rotor RPM @ 7m/s</td>
<td>12 rpm</td>
</tr>
<tr>
<td>Hub Height</td>
<td>80 m</td>
</tr>
</tbody>
</table>

\[
A = \Pi((r_{tip})^2 - (r_{hub})^2) \quad (5.2)
\]

\[
M = \frac{V_z}{(\sqrt{\gamma RT_s})} \quad (5.3)
\]

Similarly, the ‘Stage’ file, as the name suggests, has the stage details of the turbine. This includes \( \alpha_2 \), exit Mach number and rotor and stator parameters such as Zweifel numbers, phi coefficients, aspect ratios, axial velocity ratios, space coefficients, mean radii.

Table 5.2: GE wind turbine derived data input for TTDES.

<table>
<thead>
<tr>
<th>Derived Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>( r_{tip} )</td>
<td>37 m</td>
</tr>
<tr>
<td>Area</td>
<td>4657 \text{ m}^2</td>
</tr>
<tr>
<td>Average Wind Speed</td>
<td>7 m/s</td>
</tr>
<tr>
<td>Rated Rotor RPM @ 7m/s</td>
<td>12 rpm</td>
</tr>
<tr>
<td>Mach Number</td>
<td>0.0205</td>
</tr>
<tr>
<td>( m )</td>
<td>39933.775 \text{ Kg/s}</td>
</tr>
</tbody>
</table>
Table 5.3: NREL Phase VI wind turbine specifications([31]).

<table>
<thead>
<tr>
<th>Design Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Rated Power</td>
<td>20 kw</td>
</tr>
<tr>
<td>Power @7 m/s</td>
<td>5.9 kw</td>
</tr>
<tr>
<td>Rotor Diameter</td>
<td>10 m</td>
</tr>
<tr>
<td>Average Wind Speed</td>
<td>7 m/s</td>
</tr>
<tr>
<td>Rated Rotor RPM @ 7 m/s</td>
<td>71 rpm</td>
</tr>
<tr>
<td>Hub Height</td>
<td>17 m</td>
</tr>
</tbody>
</table>

Table 5.4: NREL Phase VI wind turbine derived data input for TTDES.

<table>
<thead>
<tr>
<th>Derived Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>r_tip</td>
<td>5.029 m</td>
</tr>
<tr>
<td>Area</td>
<td>79.45 m²</td>
</tr>
<tr>
<td>Average Wind Speed</td>
<td>7 m/s</td>
</tr>
<tr>
<td>Rated Rotor RPM @ 7 m/s</td>
<td>71 rpm</td>
</tr>
<tr>
<td>Mach Number</td>
<td>0.0205</td>
</tr>
<tr>
<td>$\dot{m}$</td>
<td>682 Kg/s</td>
</tr>
</tbody>
</table>

'TT-DES' is the turbine design code for automating the turbine design process. After the initial run using 'INIT' and 'Stage' file, a TT-Des result file is created that has all details of the turbine, including the 1-D design data of the blade rows specified. Euler’s Turbomachinery equations form the base for the computations in TT-Des. The ‘TT-DES’ run yields a stack and walls file. The 'stack' file has the definitions of the radial and axial locations of the blade, wheel speed, $rV_\theta$, normalized mass flow, Reynolds number. It also includes boundary layer and viscous options, to choose from. The walls file has the definition of the hub and casing co-ordinates and forms the grid file. T-Axi code is developed around multiple interacting streamtube Euler equations, where the axisymmetric equations are discretized in strong conservative form of meridional streamline grid, similar to Turbomachinery design codes developed by Drela and Giles [13] and Drela and Youngren [54].

The stack and the walls file are inputs for T-Axi calculations. To make the calculations more accurate and realistic, the values of radial and axial locations are interpolated and input by hand along with the corresponding '$rV_\theta$' values, thus making it a forced vortex method. The '$rV_\theta$' value is calculated from Euler’s Turbomachinery equations. The power is given by:

$$P = \dot{m} c_p \Delta T_T$$  \hspace{1cm} (5.4)

which is also equal to:

$$P = \dot{m} \omega (\Delta rV_\theta)$$  \hspace{1cm} (5.5)
Thus, from equations (5.4) and (5.5)

$$\Delta rV_\theta = \frac{c_p \Delta T_T}{\omega}$$

(5.6)

Equation (5.6) yields the average 'rV_\theta', which is then non-dimensionalized using the reference length and the reference velocity and then input as spanwise distribution in the stack file. T-Axi uses the r_{tip} value as the reference length and the reference speed of sound (a_0) as reference velocity. The total temperature change is calculated using the equation (5.4). The \Delta T_T thus calculated is also used to get the values of TR and ideal PR given by the equations

$$TR = \frac{\Delta T_T}{T_{in}} + 1$$

(5.7)

$$PR_{ideal} = (TR)^{\frac{\gamma}{\gamma - 1}}$$

(5.8)

Figs. 5.2 and 5.3 shows the variation of 'rV_\theta' over the span for the GE-reverse engineered blade and NREL reverse engineered case, respectively. The profile variation, along the span of a wind turbine blade gradually transitions from a circular section to an airfoil shape at about 25% of the span. Thus, the work done by the lower span region is assumed to be zero and hence 'rV_\theta' values are zero in this zone. The torque value at the tip is zero, thus the value at the tip for 'rV_\theta' ends at zero. The work done from 25% till 80% is kept constant, which attributes to the constant profile of 'rV_\theta' in that region and, there after, it gradually decreases to zero at the tip.

![Figure 5.2: Variation of rV_\theta vs radial location for GE-reverse engineered wind turbine.](image-url)
The blade-data file coupled with 3DBGB[45] solver, an in-house written code, that yields a ’3DBGB’ file, contains the sectional information of the blade in X,Y,Z co-ordinates. This file can be imported to a CAD system to generate and loft these sections to form a base blade part file. During the formation of initial CAD data, the part file is parameterized, with maximum geometrical data, which includes the spline information too. The ’Blade-data’ file includes non-dimensional values of various flow and design parameters along with inlet and outlet definitions of the blade at various radial stations. It also calculates the adiabatic efficiency along with the losses. Some of the design parameters included, which are used to compare with available literature, are aspect ratio, the chord and flow angle distribution $\beta$.

Fig. 5.4 compares the $c/r_{tip}$ value distribution, along the non dimensional radial stations for the GE reverse engineered blade to the Griffin’s report [3] for 1.5MW wind turbine blade. The results for the GE reverse engineered blade match very well to the chord distribution as shown in the literature. Fig. 5.5 shows the $c/r_{tip}$ distribution along the radial stations for the NREL reverse engineered blade to the NREL phase VI wind turbine report [33]. These results produced are also a good match to the published data in the NREL phase VI report [33].

Wind turbine rotor performance is often characterized by its power coefficient $c_p$, also known as the Betz limit [6]. This is derived from the axial induction factor, $a$, given by the formula(5.9)[30].

$$c_p = 4a(1 - a)^2$$ (5.9)
Axial induction factor, \( a \), is defined as the fractional difference between the wind velocity between free stream and the rotor plane. It is calculated by Equation 5.10 [30].

\[
4a(1 - a)^2 = \frac{P}{\frac{1}{2} \rho V_z^3 A}
\]  

(5.10)

In other words, \( c_p \) is the ratio of Power to Power in the wind. The ideal value of \( c_p \), is 0.59 as described by Betz [6]. Today, the most efficient wind turbine have \( c_p \) ranging from 0.3 to 0.4. Solving the cubic equation for \( a \), for Power of 0.35MW at 7m/s, yields \( a = 1.5, 0.247, 0.247 \). \( a \neq 1.5 \), as this factor lies between \( 0 \leq a \leq 1 \), thus taking, \( a \sim 0.247 \) yields \( c_p \) as 0.5602. \( c_p \) can also be calculated from the inlet and exit velocities using the formula (5.11).

\[
c_p = \frac{(1 + \frac{V_{out}}{V_{in}})(1 - (\frac{V_{out}}{V_{in}})^2)}{2}
\]  

(5.11)

Inlet and exit velocities are obtained from the blade data file. \( c_p \) thus obtained falls in the range of 0.32 to 0.38. The tip speed ratio \( \lambda \), which is defined as the ratio of blade tip speed to the free stream velocity of the wind, is also calculated indirectly in the blade-data file.

\[
\lambda = \frac{\omega R_{tip}}{V_z}
\]  

(5.12)

The tip-speed ratio recommended for a 3 bladed -Horizontal axis wind turbine is generally more than
Table 5.5: T-Axi Turbine parameter results for GE reverse engineered wind turbine blade.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>PR</td>
<td>1.00011701458166</td>
</tr>
<tr>
<td>TR</td>
<td>1.00003027828626</td>
</tr>
<tr>
<td>$\eta_{adiabatic}$</td>
<td>84%</td>
</tr>
<tr>
<td>$\frac{d\theta}{d\theta_f}$</td>
<td>1.231373299899381E-002</td>
</tr>
<tr>
<td>$\frac{V_z}{U}$</td>
<td>0.212090384820450</td>
</tr>
</tbody>
</table>

Table 5.6: T-Axi Turbine parameter results for NREL reverse engineered wind turbine blade.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>PR</td>
<td>1.00011274364434</td>
</tr>
<tr>
<td>TR</td>
<td>1.00002904094322</td>
</tr>
<tr>
<td>$\eta_{adiabatic}$</td>
<td>88%</td>
</tr>
<tr>
<td>$\frac{d\theta}{d\theta_f}$</td>
<td>3.759218424011611E-002</td>
</tr>
<tr>
<td>$\frac{V_z}{U}$</td>
<td>0.276205283996532</td>
</tr>
</tbody>
</table>

4[30]. The present GE reverse engineered blade design presented has a tip speed ratio of 7. The axial turbine values for the GE and NREL reverse engineered wind turbine blade are shown in Tables 5.5 and 5.6, obtained from the T-Axi runs.

The CAD part files of NREL Phase VI blade and GE are shown in Figs. 5.6 and 5.8 and the reverse engineered CAD part files generated shown in Figs. 5.7 and 5.9.

Figure 5.6: CAD Blade Design (NREL Phase VI blade) [31].
Figure 5.7: NREL reverse engineered Wind turbine Blade

Figure 5.8: GE Windturbine Blade([17]).

Figure 5.9: GE reverse engineered Windturbine Blade.
Chapter 6

Cascade Analysis using MISES

MISES is a viscous/inviscid cascade solver and design system created by Mark Drela and Harold Youngren [55]. The program is a complete CFD procedure from geometry definition to post processing tools. It is a quasi-3D computational method used for design and analysis of airfoils for axial turbomachinery designs. It has a finite volume approach to flow discretization. The inviscid flow is described by Euler’s equations and viscous effects are modeled using integral boundary layer equations. The coupled system of the nonlinear equations is solved by a Newton-Raphson technique. MISES also uses the Abu-Ghannam/Shaw (AGS) [7],[15] for transition prediction.

6.1 Issues with Geometry Definition

The accuracy of any flow simulation depends on many factors and one of the important factor is getting the right geometry into the flow solver. This part is however one of the most challenging tasks as the definition of geometry is not accurate always. In this thesis, a reverse engineering approach was followed and the geometry as published in the literature was tried to match. This process of matching the geometry or rather getting close to the real geometry was the most challenging task and led to errors between the results. The 3D geometry for the NREL test case, was provided by NUMECA which is in open domain and the blade profile data or rather the airfoil geometry was derived from the NREL NASA Phase VI report[33]. Fig. 6.1 shows the difference between the blade profiles at 50% span. The geometry derived from the 3D data initially had stagger as shown in Fig. 6.2. This stagger had to be removed, the profile rotated, translated, and scaled to match the profile geometry derived from the NREL-S809 blade profile report. Thus, a small mismatch was found in the profile geometry, between the two blade profiles of Numeca-3D and NREL-S809 data. This mis-match leads to errors in comparisons made between
aerodynamic quantities between both the profiles.

Figure 6.1: Comparison of blade profiles at mid span - NUMECA-3D and NREL-S809.

Figure 6.2: Blade profile of NUMECA-3D at midspan.

6.2 Wind Turbine Design and Analysis using MISES

The 3DBGB code generates the 'blade' and 'ises' files that form the input to the MISES analysis and redesign. As discussed earlier, MISES is a cascade solver with the ability to have boundary layer coupling included during execution. This is achieved by a Reynolds number input for the blade section in the 'ises' file. The inputs used for the 3DBGB-NREL blade that is 'blade.case' file and corresponding 'ises' file is attached in Appendix L. The blade coordinates are the m’, θ points on the blade surface, that starts from the trailing edge and then goes round the leading edge back to the trailing edge, but is not closed, so that a blunt trailing edge is achieved. This is done to incorporate the Kutta condition over finite thickness. Fig. 6.3(a) shows the cascade arrangement for the MISES setup. The pitch between the blade sections forms the circumferential separation of the cascade. The pitch value is set in the blade file. The ‘iset’ command along with the case extension sets up the case to run in MISES. This creates the grid file for the cascade as shown in Fig. 6.3(b).
An analysis of the $C_l$ distribution along the pitch/chord ratio is done on the NREL-S809 sections derived from the NREL NASA Phase VI report [31] to see where the blade effects start to become 2D. MISES results give $C_y$ which can be correlated to $C_l$ using the following derivation shown:

$$C_y = Torque \text{ about } y/(mass \ r \ V_1)$$

(6.1)

as presented in [55].

$$C_y = (Fr)/\rho V_1 A V_1 r$$

(6.2)
\[ F = \rho V_1 bpV_1 \]  
(6.3)

\[ A = bp \]  
(6.4)

\[ = \frac{2\Pi r}{n} \]  
(6.5)

\[ F = pC_y\rho V_1^2 \]  
(6.6)

\[ C_l = F/\frac{1}{2}\rho V_1^2 \]  
(6.7)

Thus, substituting for 'F' in equation 6.7 from equation 6.2, gives:

\[ C_l = 2pC_y \]  
(6.8)

The variation of \( C_l \) vs pitch/chord ratio starts to asymptote. Fig. 6.4 shows that the \( C_l \) asymptotes as it reaches a pitch/chord ratio of 8.5. The value of \( C_l \) is calculated using the equation (6.8). Fig. 6.5 shows the pitch to chord ratio of NREL blade. The variation shows a high pitch to chord value spanwise, which is expected behavior for a wind turbine. Thus, cascade analysis carried out on machines with high pitch to chord ratios, leads to the fact, the blade to blade effects are not valid and the wind turbine blade starts to behave as a wing. MISES analysis on the airfoils yields various aerodynamic quantities and were used to compare NUMECA-3D, NREL-S809, and the 3DBGB-NREL blade sections at midspan. Fig. 6.6 shows the comparison of blade section at 50% span for NUMECA-3D, NREL-S809, and 3DBGB-NREL cases. The other quantities of interest are the \( C_y, C_x \) values at a given incidence angle, \( C_p, C_f, H, \) and \( Re_\theta \). The NREL blade section uses an 'S809' airfoil and many authors have done extensive analysis on this airfoil. The NREL NASA Phase VI[31] details an extensive report on it and has been used for comparison. The

Figure 6.4: \( C_l \) vs Pitch/Chord.
Figure 6.5: Pitch/Chord vs r/t_tip.

Figure 6.6: Profile comparisons at mid span.

C_y for this airfoil is maximum at a incidence angle of 14°. Thus, all the sections derived, i.e, NUMECA-3D blade airfoil, NREL-S809 airfoil, and the 3DBGB-NREL airfoil are analyzed at this incidence angle. The values of C_y, C_x and \( \frac{C_y}{C_x} \) for each profile run through MISES at 14° are tabulated in Table (6.1). The values of C_l, C_d and \( \frac{C_l}{C_d} \) calculated using the equation (6.8) are tabulated in Table (6.2).

The 'IPILOT' option in MISES is a general post-processor. This option is used to get the various plots of aerodynamic quantities. The first comparison done on all the blade profile shapes at 50% span is the C_p plot. Figure 6.7 shows the C_p comparision for all the three profiles described. This figure shows how well the C_p compare to each other. The transtion option used, was AGS bypass transtion. Fig. 6.8

33
Table 6.1: \( C_y, C_x \) values of different profiles from Mises at 14\(^\circ\) at mid span.

<table>
<thead>
<tr>
<th>Case</th>
<th>( C_y )</th>
<th>( C_x )</th>
<th>( \frac{C_y}{C_x} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUMECA-3D</td>
<td>0.0636</td>
<td>-0.0075</td>
<td>8.48</td>
</tr>
<tr>
<td>NREL-S809</td>
<td>0.0558</td>
<td>-0.0072</td>
<td>7.75</td>
</tr>
<tr>
<td>3DBGB-NREL</td>
<td>0.0623</td>
<td>-0.0077</td>
<td>8.09</td>
</tr>
</tbody>
</table>

Table 6.2: \( C_l, C_d \) values of different profiles from Mises at 14\(^\circ\) at mid span.

<table>
<thead>
<tr>
<th>Case</th>
<th>( C_l )</th>
<th>( C_d )</th>
<th>( \frac{C_l}{C_d} )</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUMECA-3D</td>
<td>1.272</td>
<td>0.150</td>
<td>8.48</td>
</tr>
<tr>
<td>NREL-S809</td>
<td>1.116</td>
<td>0.144</td>
<td>7.75</td>
</tr>
<tr>
<td>3DBGB-NREL</td>
<td>1.246</td>
<td>0.154</td>
<td>8.09</td>
</tr>
</tbody>
</table>

shows the \( C_p \) comparision between the two modes, AGS and fully-trbulent, executed through Mises on S809-NREL profile. Table 6.3 tabulates the Xtr1 and Xtr2 values, which denote the transtion locations on suction and pressure side respectively, for NREL-S809 profile, executed using AGS and \( e^n \) transtion modes in Mises, at an incidence angle of 14\(^\circ\). The Xtr1 and Xtr2 are in terms of \% chord values. The lift and drag coefficients from both the modes of transtion in Mises are tabulated in Table 6.4.

Figs. 6.9,6.10, and 6.11 shows the Mises output plots of shape factor, momentum thickness Reynolds number, and skin friction coefficient for all the three profiles.

Figure 6.7: \( C_p \) comparisons at mid span from Mises.
Figure 6.8: $C_p$ comparisons at mid span from MISES, for S809-NREL using AGS and Fully-Turbulent modes.

Table 6.3: Comparison between transition locations for NREL-S809 profile from MISES at 14° at mid span, using AGS and $e^n$ modes.

<table>
<thead>
<tr>
<th>Case</th>
<th>Xtr1</th>
<th>Xtr2</th>
</tr>
</thead>
<tbody>
<tr>
<td>AGS</td>
<td>0.0008</td>
<td>0.2756</td>
</tr>
<tr>
<td>$e^n$</td>
<td>0.0029</td>
<td>0.5559</td>
</tr>
</tbody>
</table>

Table 6.4: Comparison between lift and drag coefficient for NREL-S809 profile from MISES at 14° at mid span, using AGS and $e^n$ modes.

<table>
<thead>
<tr>
<th>Case</th>
<th>$C_l$</th>
<th>$C_d$</th>
</tr>
</thead>
<tbody>
<tr>
<td>AGS</td>
<td>1.0035666</td>
<td>0.1305972</td>
</tr>
<tr>
<td>$e^n$</td>
<td>1.0518768</td>
<td>0.133821</td>
</tr>
</tbody>
</table>
Figure 6.9: Shape factor plot from Mises at mid span.
Figure 6.10: \( \text{Re}_\theta \) plots at mid span.
Figure 6.11: Co-efficient of friction plots at mid span.
Chapter 7

Wing Analysis using XFLR5

XFLR5[53] is an open source code, used for analysis of wings and airfoils, and is based on XFOIL[14]. XFLR5 is easy to use and no background on how to run XFOIL is needed. This code was used to see the correlation between the MISES analysis and 2D wing analysis. Thus, all the profiles were analyzed using XFLR5, with the same conditions as analyzed in MISES. XFLR5 uses XFOIL code as its base, thus the blade files which were used for analyzing in MISES, could be reused. The $C_p$ plot for each profile was generated from this code. Fig. 7.1 shows a comparison of the $C_p$ thus generated and plotted against $m'$, for the three profiles analyzed through XFLR5 and its comparison to 3D-CFD result (described in the next chapter). The analysis shows the correlation of a 2D wing airfoil analysis to a 3D-CFD analysis for the wind-turbine blade, showing the 2D nature of such kind of machines. This fact is further discussed with the help of 3D-CFD results in detail, in chapter 8.

![Figure 7.1: $C_p$ comparisons at mid span from XFLR5.](image)

39
XFLR5 includes the free transition mode, or $e^\text{nu}$ transition option. Thus the NREL-S809 profile analyzed in MISES and XFLR5 with the same transition mode and the $C_p$ plots compared, as shown in Fig. 7.2.

Figure 7.2: $C_p$ comparisons at mid span between XFLR5 and MISES ($e^\text{nu}$).
Chapter 8

3D-CFD Analysis using Fine Turbo

8.1 Case setup

The first step for setting calculations in a 3D CFD tool is to generate a mesh for the given geometry. The geometry complexity defines the need for the mesh type to be structured or unstructured. Structured meshes are cumbersome and difficult to generate sometimes, but provide good accuracy where the flow is aligned and allows better boundary layer resolution. Unstructured meshes are relatively easy to generate and are independent of geometry complexity. They also generate less points as compared to structured meshes. The Autogrid module in Fine/turbo is a good structured mesh generator, which has the ability to adapt structured meshes automatically for various turbomachinery designs. Thus, this module is used along with the Wind-Turbine wizard to generate the mesh for the defined problem. Once the part file is generated, it can then be exported as a parasolid part file into a high fidelity 3D CFD code Fine/Turbo developed by NUMECA [37]. Alternatively blade sections generated from 3DBGB can be used to generate a ’geomturbo’ file, which is a native format to Fine /Turbo and can be readily imported with hub and casing definitions into AutoGrid. Fig. 8.1 shows the basic nomenclature used in Autogrid module. Using its automated mesh generation tool AutoGrid and using the ’Wind-Turbine wizard’ module, the mesh is generated for the blade file along with suitable boundary conditions. This mesh file is then analyzed for various aerodynamic aspects in the flow solver. The loss calculated through MISES, namely $\zeta$ loss is then compared to calculated results generated from Fine/Turbo. The $C_p$ plot is generated and is compared with $C_p$ plot generated through MISES for specific blade sections.
8.2 Grid Generation using AutoGrid and Wind Turbine module in Fine/- Turbo

To automate the geometry import with the hub and shroud definition into Autogrid, a code 'geom-turbo_creator' (K), written in FORTRAN 9x is used. This uses the walls file obtained from TTDES and section file generated from 3DBGB to generate a '.geomturbo' file, which is a native format to Fine/Turbo Autogrid module. This file is used to initialize the grid parameters, and geometry definition in Autogrid. Here the ‘Wind-turbine wizard’ module is used to create the required grid for the analysis. First the geometry imported into Autogrid is checked for irregularities, hub and shroud definitions and the blade intersection to the hub and shroud with the use of the wind turbine wizard module. After the geometry has passed the ‘geometry check’, it comes up on the screen as shown in Fig. 8.2. The basic nomenclature of turbomachinery configuration domain in Fine /Turbo is as described in Fig. 8.3. Then
the row properties are input, i.e., periodicity (number of blades) and rotation speed (RPM) by accessing the row list on the left top corner. The RPM value is negative in this case (NREL) to account for the direction of rotation. The next tab would ask for the type of the analysis (choose wind turbine). The shroud, Upstream, Downstream and Far field limits are set using the following relations [37].

\[ L_{\text{upstream}} = \text{abs}(\text{BladeHeight} \times \text{FarField Zminvalue}) \]  

\[ L_{\text{downstream}} = \text{abs}(\text{BladeHeight} \times \text{FarField Zmaxvalue}) \]  

\[ R_{\text{shroud}} = R_{\text{tip}} - (1e^{-5})(\text{Bladeheight}) \text{ [by default]} \]  

\[ H = \text{Farfield Rvalue} \times \text{Bladeheight} \]  

The computational domain for the current simulation is as shown in Fig. 8.4. Fig 8.5 shows the terminology of the topology used around the blade, generated through the wind turbine wizard mode. Fig. 8.6 shows the grid used for current simulation around the blade generated through the wizard mode. The mesh generated through the wizard mode uses transfinite interpolation for all the blocks, except the skin block or the mesh closest to the blade surface, where a hyperbolic mesh generation is used. For the far-field boundary, the maximum expansion ratio in the radial direction is 1.8 and in the axial direction is 1. The type of clustering used is uniform.

After the initial mesh parameters are input, an important parameter is the width of the first cell close to the wall. It is input using the Blasius Equation for turbulent flow [37],

\[ y_{\text{wall}} = 6 \left( \frac{V_{\text{ref}}}{\nu} \right)^{\frac{1}{2}} \left( \frac{L_{\text{ref}}}{2} \right)^{\frac{1}{2}} y_{\text{1}}^{+} \]
Figure 8.4: Computational Domain.

Figure 8.5: Wizard mesh topology [37].

Figure 8.6: Computational grid at mid-span.

where:

- $y_{wall}$ Distance of the nearest grid point to the wall. (meter)
- $V_{ref}$ Reference velocity of the flow. (m/s)
- $\nu$ Kinematic viscosity of the fluid. ($m^2/s$)
Figure 8.7: Far field height.

- \( L_{ref} \) Reference Length (meters)
- \( y^+ \) Non-dimensional Value

The above mentioned equation is derived from the Blasius pipe turbulent flow equation and then relating the \( y^+ \) definition as derived below. The Blasius equation for pipe flow is given by:

\[
C_f = 0.0791 Re_D^{1/4} \tag{8.6}
\]

\[
\tau_w = 0.0396 \frac{\rho^{3/4} V_{ref}^{7/4} \mu^{1/4}}{D_h^{1/4}} [51] \tag{8.7}
\]

\[
\tau_w = \frac{2^{1/4}}{36} \frac{\rho V_{ref}^{7/4} \mu^{1/4}}{\rho^{1/4} \rho^{1/4} L_{ref}^{1/4}} \tag{8.8}
\]

\[
\tau_w = \frac{2^{1/4}}{36} \frac{\rho V_{ref}^{7/4} \mu^{1/4}}{\rho^{1/4} L_{ref}^{1/4}} \tag{8.9}
\]

\[
Y^+ = \frac{u^+ y}{\nu} \tag{8.10}
\]

\[
y_{wall} = \frac{\nu}{u^+ y^+} \tag{8.11}
\]

\[
u^+ = \frac{\tau_w}{\rho} \tag{8.12}
\]

\[
\nu = \frac{\mu}{\rho} \tag{8.13}
\]

\[
y_{wall} = \nu \frac{\rho}{\tau_w} y^+ \tag{8.14}
\]
Thus, plugging the derived equation for $\tau_w$ from equation (8.9) in equation (8.14),

$$y_{wall} = \sqrt{\frac{\nu^2 \rho^{36}}{2^{1/4} \rho V_{ref}^7}} \nu^{1/4} L_{ref}^{-1/4} y^+$$  \hspace{1cm} (8.15)

$$= 6 \sqrt{\frac{L_{ref} 2^{1/4} \nu^{7/4}}{V_{ref}^{7/4}}} y^+$$  \hspace{1cm} (8.16)

Thus,

$$y_{wall} = 6 \left( \frac{V_{ref}}{\nu} \right)^{-7/8} \left( \frac{L_{ref}}{2} \right)^{1/8} y^+$$  \hspace{1cm} (8.17)

The $y^+$ value as recommended by NUMECA to locate the nearest grid point is $1<y^+<10$. A $y^+$ value of 5 is initially assumed to calculate the $y_{wall}$ value and input during the grid generation as 'Cell width' option under 'Row Mesh Control' in Autogrid. The next step is to generate the blade-to-blade mesh (B2B) view. The view can be set to desired span or active layer using % of the span which is 0% at the hub to 100% at the tip. Fig. 8.8 shows the B2B mesh at 50% span of the NREL wind turbine blade.

The stacking of blade to blade meshes on the surface of revolution generated from meridional curves called the flow paths generates the 3D mesh. The flow path definition is set or modified by accessing the 'Mesh control' tab and the 'Flow Path Control' option. The Fig. 8.9 shows the flow path control view for NREL blade. Once the base mesh is created, the following mesh refinement is done to get rid of negative cells. Generating the mesh using the wizard mode will automatically select the 'Wind Turbine (WT) High Staggered Option' for the geometry, however this causes problems in the area near

![Figure 8.8: B2B mesh mid span.](image)

46
the hub where the blade is not highly staggered. This can be changed using the following steps: Click on the 'B2B' button in the row of buttons above the graphics window, click on topology to expand this section, select Wind Turbine (WT). Due to the highly curved shape of the blade the 'stick leading and trailing edge' (this should be stitch leading and trailing edge). Right click on Main Blade, go to Expand Geometry, select 'stick leading and trailing edge' and Apply after this refinement to the geometry, the following adjustments to the mesh generation is done. Change the flow paths distribution: Under 'Row Mesh Control' increase the 'Flow Paths Number' to 125 (you can change this value for desired results) then click on 'Flow Paths Control' to open the dialog box and increase the 'Percentage of Mid-flow Cells' to 70%. This will distribute the flow paths to the center and provide a denser flow path distribution around the area of the blade that sharply changes shape. Under 'Row Mesh Control', change the 'Span Interpolation' to 0%. This will eliminate interpolation that is done between the flow paths and produce flow paths which conform better to the blade.

After the mesh refinement, 3D grid generation is done and the project file saved. At the end of the 3D grid generation, Fine/Turbo generates a grid quality report that shows various aspects of mesh. These parameters define the quality of mesh generated in terms of minimum and maximum value of expansion ratio, aspect ratio and cell skewness. The negative cells are also detected, if any and will show up during the grid quality report generation. Fig. 8.10 shows the grid quality report generated for the current design.
8.2.1 Quality Criterion Definitions

As discussed before the grid quality report has the quality criterion for the mesh generated. Each CFD code has an unique way to look at these criteria. The following are the criteria defined by NUMECA for Fine /Turbo with their definitions and limits:

- **Orthogonality** The measure of minimum angle between edges of the element for a 2D case and measure of the cell angle relative to the face for 3D. Limits $25 < \text{angle} < 90$. The angle is measured in degrees. If the angle between the two edges is more than 90, then the value taken into account is $180 - \text{real angle}$.

- **Aspect Ratio** The amount of stretching the mesh elements are subjected to, considering a single cell.

$$ Aspect\text{ratio} = \frac{\max(x, y)}{\min(x, y)} \quad [37] $$

The definition of $x$ and $y$ is as shown in the Fig. 8.11. Limits $1 < \text{Aspect ratio} < 5e5$.

- **Expansion Ratio** The measure of size variation between the adjacent cells. The Fig. 8.12 shows the definition of expansion ratio as used in Fine /Turbo. Limit $1 < \text{Expansion Ratio} < 5$.

The values of above mentioned quantities are tabulated in table 8.1 for the current mesh generated for the NREL blade. These values are well in the limits of the values as recommended by NUMECA.
Figure 8.11: Aspect ratio [37].

Figure 8.12: Expansion ratio [37].

Table 8.1: Grid quality report values

<table>
<thead>
<tr>
<th>Quality Parameter</th>
<th>min Value</th>
<th>max Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Aspect Ratio</td>
<td>1</td>
<td>3e5</td>
</tr>
<tr>
<td>Expansion ratio</td>
<td>1</td>
<td>3.4</td>
</tr>
<tr>
<td>Orthogonality</td>
<td>36</td>
<td>90</td>
</tr>
</tbody>
</table>

8.2.2 Boundary Conditions

During the 3D mesh generation using the ‘Wind Turbine’ wizard, default boundary conditions are applied to the grid. These include:

- Periodic Matching When the physical geometry and the expected flow pattern have a periodically repeating nature. The connection becomes matching if the mesh points along the connected patches are identical. Fig. 8.13 shows the periodic conditions applied to the grid. Generally no user input is required.

- Solid Wall Two options available, namely Euler and Navier-Stokes wall boundary. The Euler’s wall is the inviscid case and no parameter is requested for this case. The Euler’s condition is applied to the walls as shown in Fig. 8.14 for the current mesh. For the rotating parts, Navier-Stokes wall condition is applied, as shown in Fig. 8.15. The Navier-stokes wall has two options,
Cylindrical or Cartesian conditions. For the current case, cylindrical conditions are applied with adiabatic walls and constant rotation speed as input (-71 rpm for NREL).

- External (Far Field) Condition To deal with external flow conditions the far-field boundary or the external boundary conditions are applied. The inputs required in this condition include static pressure (Pa), static temperature (K), velocity ($V_z$) (m/s), turbulent viscosity (m$^2$/s). Fig. 8.16 shows the far-field condition.

8.2.3 Flow solver - EURANUS

After the mesh is generated, along with proper boundary conditions, the main flow solver module 'Fine' is used to impose additional conditions for the flow solver 'EURANUS'. The parameters in the flow management include:
• Configuration

  – Fluid Model  Input the fluid properties to be used by computation. Current case uses ’Air (perfect gas)’ as the fluid model.
  – Flow Model  Characteristics of the flow :
    † Time configuration  Steady
    † Mathematical Model
      † Mathematical Model  Turbulent Navier-Stokes
      † Modeling of turbulence  Spalart-Allmaras
    † Low speed flow  (M<0.3)
  – Rotating Machinery  Defines the rotational speed of the grid blocks generated. The value is input by the user in rpm.
• Optional Models Additional models if needed can be used. The option of selecting 'Transition Model' with AGS modeling is available here.

• Boundary Conditions The Boundary condition already generated in Autogrid show up for additional input definition if required.

• Numerical Model Allows to define CFL number, grid level (coarse or fine) and preconditioning parameters.

• Initial Solution Provides the possibility to start a computation from the values specified.

• Outputs Required outputs can be checked and viewed in the post processor.

• Computation Steering This option sets the control variables, i.e., maximum number of iterations to be performed along with the convergence criteria.

After the required parameters set, the case file is saved and the flow solver is executed. The convergence history can be monitored in the graphics window and required quantities be selected for output against the number of iterations being performed. Mostly the global residuals convergence is monitored and as it reaches the convergence criteria set, the flow solver stops. EURANUS creates a '.run' file that has solution as desired outputs and is the input for the post processing tool 'CFVIEW'. Also a '.CGNS' file is created that can be loaded to other post processing tools such as 'TechPlot', ‘ASGARD’ (in-house CFD post-processing tool).

8.3 Convergence History

The case was set to run on a Linux based cluster available in the Aerospace Engineering department, University of Cincinnati. The computations were run on 4 dual core Pentium 2.4 Ghz processors and took 1 day for a ’111’ grid solution to converge. The convergence plots of the global residual against the number of iterations is shown in Fig. 8.17. It is seen that, after 5000 iterations the change in global residuals against the number of iterations is minimum, but does not settle. Fig. 8.18 shows the torque convergence against the number of iterations. This plot confirms that even though the global residuals have not converged over 15000 iterations, the change in torque values have stabilized. Since the boundary condition imposed for this problem does not have an inlet or outlet boundary condition imposed, mass inflow and outflow cannot be monitored. Thus, other values like axial thrust and torque are the values that can be monitored for determining flow convergence.
8.4 Grid-Dependency Study

Three grids designs, '222', '111', '000' were used to see the grid dependency of the case on the NUMECA-3D blade. The terminology of the grid is native to Fine/ Turbo, where the finest grid that can be accessed is '000', '111' being the intermediate and '222' being the coarsest. The number of grid points for each grid is tabulated below for NUMECA-3D case in table (8.2). Table 8.3, shows the range of $Y^+$ values on the blade surface for each of the grid configuration.

<table>
<thead>
<tr>
<th>Grid definition</th>
<th>Number of grid points</th>
</tr>
</thead>
<tbody>
<tr>
<td>222</td>
<td>0.4 Million</td>
</tr>
<tr>
<td>111</td>
<td>3.2 Million</td>
</tr>
<tr>
<td>000</td>
<td>27 Million</td>
</tr>
</tbody>
</table>

The coarse grid '222' converges after 5596 iterations as compared to 15000 iterations taken by the '111' grid. The '222' grid solution takes 3 hours running parallel on four dual core machines with 2.4Ghz
Table 8.3: $Y^+$ range for each grid configuration.

<table>
<thead>
<tr>
<th>$Y^+$ values</th>
<th>Range</th>
</tr>
</thead>
<tbody>
<tr>
<td>222</td>
<td>0 to 6</td>
</tr>
<tr>
<td>111</td>
<td>0 to 2.5</td>
</tr>
<tr>
<td>000</td>
<td>0 to 1.5</td>
</tr>
</tbody>
</table>

Pentium processors as compared to 1 day for ’111’ grid. Fig. 8.19 shows the convergence plot of ’222’ grid case. The finest grid ’000’ has 27 million grid points. This grid took 4.5 days to complete the same number of iterations, running on the same cluster configuration. The input to the fine grid solution was the converged ’111’ solution. As described before, the change in torque and axial thrust was monitored, for convergence. The $C_p$ plot was also output for each grid and is shown in Fig. 8.20. Other quantity of interest was the torque variation between each grid level and thus generate a corresponding power output. The power generated out of the NREL NASA Phase VI machine at 7m/s is 5.9Kw [10]. The table (8.4) gives the comparison of power outputs computed using the equation 8.20, from various grid arrangements. It is seen that ’000’ grid generates highest power. A comparison of NREL data to the NUMECA-3D data is not entirely appropriate due to the fact that, both geometries are not the same, and had already been discussed in chapter 6. Also, the $C_l/C_d$ value is more for the NUMECA geometry than that of the NREL-S809 geometry, as computed in MISES (6.2). This also leads to the conclusion that the power output would be higher for the NUMECA geometry. The table 8.5 tabulates the $\zeta$ values calculated using the equation 8.21 for each grid level. The ’000’ grid has the lowest value and ’222’ has the highest. As $\zeta$ is a loss coefficient value, it would affect the power output and is clearly seen form table (8.4), ’000’ has the maximum power output and ’222’ has the least. The other comparison is the $C_p$ plot between all the grid levels as shown in Fig. 8.20. A comparison between the $C_p$ plots for NUMECA-3D, generated from MISES and ’000’ grid case is also shown in Fig. 8.21.

Table 8.4: Power output from different grid arrangements.

<table>
<thead>
<tr>
<th>Grid definition</th>
<th>Power (Kw)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Power at 7m/s</td>
<td>5.9</td>
</tr>
<tr>
<td>222</td>
<td>4.6930</td>
</tr>
<tr>
<td>111</td>
<td>5.88374</td>
</tr>
<tr>
<td>000</td>
<td>6.197</td>
</tr>
</tbody>
</table>

8.5 Results

Fine/Turbo has a post-processing module called ’CFVIEW’. The ’run’ file generated by EURANUS is loaded in CFVIEW. The desired flow quantities that were selected to be output during the flow solver
Figure 8.19: Global residual plot of '222' grid.

Figure 8.20: $C_p$ comparison of different grids at mid span (NUMECA-3D).

Table 8.5: Loss model values at mid span for different grid levels (NUMECA-3D).

<table>
<thead>
<tr>
<th>Grid Level</th>
<th>( \zeta )</th>
</tr>
</thead>
<tbody>
<tr>
<td>000</td>
<td>0.002751</td>
</tr>
<tr>
<td>111</td>
<td>0.002812</td>
</tr>
<tr>
<td>222</td>
<td>0.003317</td>
</tr>
<tr>
<td>MISES</td>
<td>0.002139</td>
</tr>
</tbody>
</table>

execution shows up in the graphics window and can be selected for contour plots or line plots. If required, new quantities are defined and the flow solver is executed with one iteration. This calculates the new quantity and shows up in CFVIEW. Fig. 8.22 shows the $Y^+$ values on the blade surface. The $Y^+$ value was guessed and the $y_{wall}$ value was input initially during grid generation. The figure shows the maximum $Y^+$ value to be 2.5 which is well in the limit as recommended by NUMECA.

The torque value obtained is used to calculate the power given by:

\[
P = T\omega
\]  
(8.19)
Figure 8.21: \( C_p \) comparison between '000' grid and MISES.

![Comparison between '000' grid and MISES](image)

Figure 8.22: \( Y^+ \) values.

![Y+ values](image)

The power thus calculated by equation 8.19 is 5.88374 Kw for the NUMECA-3D case.

To generate the \( C_p \) plots, a user-defined quantity is input in CFVIEW. The \( C_p \) formulation is kept same as derived from MISES and is given by:

\[
C_p = \frac{1}{2} \rho \left[ \left( \omega r \right)^2 + V_{ref}^2 \right]
\]  \hspace{1cm} (8.20)

In the equation 8.20 the value of \( V_1 \) is the velocity derived from the input Mach number in MISES input file, 'ISES' for the particular blade profile. Fig. 8.24 shows the \( C_p \) plot derived from CFVIEW output and its comparison with MISES runs on the 3DBGB-NREL and NUMECA-3D blade profile. The Fig. 8.23
shows the $C_p$ plot, both Cartesian and blade section view in CFVIEW, for NUMECA-3D blade profile at mid-span.

Figure 8.23: $C_p$ plot at 50% span.

Figure 8.24: $C_p$ comparison at mid span.

ζ loss values computed in MISES is compared to the values computed from 3D CFD calculations. The equation for the loss coefficients is as described in Denton’s paper [12] for loss models in turbomachinery. The following equation is input in CFVIEW as user-defined quantities.
The values of loss model computed in CFVIEW using equation 8.21 and MISES. The values of total pressures were mass averaged and static pressures were area averaged, used in the above computations. The value of \( P_{t rel} \) is the value of the absolute value of pressure as specified in the initial input of the calculation, which is 101325 (Pa). As, mentioned earlier, MISES has the option of transition, to be set as bypass transition using AGS condition, \( e^n \) transition, and specified location (Fully turbulent specified at leading edge). Fine/turbo too has the option of choosing from AGS transition model and fully turbulent model. Thus, both the cases were executed in Fine/Turbo and MISES, and the loss value compared respectively is shown in table 8.6.

<table>
<thead>
<tr>
<th>Case</th>
<th>( \zeta )</th>
</tr>
</thead>
<tbody>
<tr>
<td>NUMECA-3D Fine/Turbo- AGS</td>
<td>0.002721</td>
</tr>
<tr>
<td>NUMECA-3D-MISES- AGS</td>
<td>0.002384</td>
</tr>
<tr>
<td>NUMECA-3D Fine/Turbo- Fully Turbulent</td>
<td>0.002814</td>
</tr>
<tr>
<td>NUMECA-3D-MISES- Fully Turbulent</td>
<td>0.002513</td>
</tr>
</tbody>
</table>

Figs. 8.25 and 8.26 show radial velocity and \( \phi \) angle contour plots, generated using ASGARD. For a 2D profile the radial velocity is zero. From the radial velocity plot it is evident that from 25\% to 75\%, the wind turbine acts as a 2D wing. The \( \phi \) angles plot also confirm this argument. \( \phi \) angle is defined as:

\[
\phi = \arctan\left(\frac{V_r}{V_z}\right)
\]  

(8.22)

The \( \phi \) angles plot shows high angles towards the tip, and this is due to 3D and tip effects. Thus, from 75\% to tip, 3D and tip effects are to be considered. From hub to 25\% of the span, blade to blade effects are to be considered. Fig. 8.27 shows 2D line contour plot of the \( \phi \) angles. This plot shows the value of \( \phi \) angle is high from 75\% to tip, confirming the fact that from 75\% of the span to tip, 3D tip effects must be considered.

A comparison of tip vortices generated, between a wing and wind turbine blade is shown in Figs. 8.28 and 8.29. This clearly shows the difference between the vortex formed behind a 2D wing and a wind turbine blade. This confirms that, a for a wind turbine blade, it is necessary to take into account the rotational effects to account for tip vortices and tip design. Thus, an assumption of considering a wind turbine blade to act totally as a 2D wing will not hold good. Fig. 8.30 shows the contour plot of static pressure with corresponding line plot. Fig. 8.30(c) shows the non-dimensional plot of static pressure,
(a) Contour plot of radial velocity.  
(b) Contour plot range.

Figure 8.25: Radial Velocity plot for NUMECA-3D.

(a) Contour plot of Phi angle.  
(b) Contour plot range.

Figure 8.26: Phi angle plot for NUMECA-3D.
Figure 8.27: 2D line plot of phi angles for NUMECA-3D.

normalized by:

\[ P_{sg} = \frac{P_s - P_{ref}}{0.5\rho V_{ref}^2} \]  \hspace{1cm} (8.23)

Figure 8.28: Wing tip vortex ([26]).

Fig.8.31 shows the spanwise \( rV_\theta \) distribution for the 3D CFD solution of NUMECA-3D, for the '000' grid case. This plot shows that the values of \( rV_\theta \) is zero till the circular section till 25% of span, has a distribution of \( rV_{\theta,\text{beta}} \) values from 25% to 75% and then has goes to zero at the tip. The assumption made in earlier chapter (5), to design a wind turbine blade out of T-Axi was similar, but the distribution of \( rV_\theta \) from 25% to 75% was kept constant. A \( rV_\theta \) value relates to the loading or work done from that
section, thus, from hub till 25% span of the blade improvements can be made and work can be extracted from this part of the section too. All the plots are taken at 0.3 m downstream, from the central axis of the blade. Fig. 8.31(c) shows the normalized plots of $rV_\theta$ using the following relations:

$$rV_\theta = \frac{rV_\theta}{r_{tip}V_z}$$  \hspace{1cm} (8.24)

The reference quantities are the freestream static values, where, $P_{ref} = 101295$ Pa, $T_{ref} = 288.12$ K, $C_p$ is the specific at heat constant pressure = 1.005 kJ/Kg K, and the $V_{ref}$ is 7m/s.
Figure 8.30: Area averaged plot for static-pressure for NUMECA-3D in meridional view.
Figure 8.31: Mass Averaged plot for $rV_\theta$ for NUMECA-3D in meridional view.
Chapter 9

FEA stress and Modal Analysis

All mechanical structures have their own natural frequencies called resonating frequency, which when attained, causes failure in the structure. Thus, these frequencies must be avoided by design changes and selecting suitable material properties. In wind turbine blades, it is the flutter that causes the most vibration effects in the structure that has must be avoided, for the structural integrity of the mechanism. Also this is one of the contributing factors for the noise. Thus the various resonant frequencies of the Wind Turbine blade must be found and avoided. The process of obtaining the inherent dynamic characteristics of the system in the form of its various natural frequencies, mode shapes and damping factors in the form of mathematical equations is called the modal analysis of the system. Thus, each mode is described in form of its corresponding frequencies or mode shapes using these equations [16]. The wind turbine blade thus was analyzed for its various mode shapes to account for the flutter and acoustics [4], [38]. Sample modal output from ANSYS for the continuous slope disk is attached in the Appendix F. A subroutine ANSYS_WRITER D is written to output a ’ANSYS.AIN’ file which gives an ANSYS Parametric Design Language (APDL) script, which opened in ANSYS, automatically generates meshed part file that is ready for any kind of FEA study to be done in ANSYS. The Meshed Wind Turbine Blade, with 8 node hexahedron brick 185 element are shown in Fig.9.1. The first five mode shapes for the GE reverse engineered Wind Turbine blade is shown in Fig. 9.2. Table 9.1 shows the first five natural frequencies of the GE reverse engineered blade, when simulated as a cantilever beam which is rotating. The material used for the test case was Aluminum to demonstrate the capability, although most of the wind turbines are made of fiberglass or other types of composites. The GE 1.5sle wind turbine is rated for a range of wind speeds (3.5 m/s - 25 m/s) [17]. Thus, it will have different rpms associated with these wind speeds, as the angular velocity is directly proportional to the wind speeds, and is given by the following
The fundamental frequency calculated from angular velocity, is given by the correlation:

\[ f = \frac{\omega}{60} \]  \hspace{2cm} (9.2)

Thus, calculation the rotational frequency using the Equation (9.2), yields a range \((0.2 \leq f \leq 1.433)\). The value of first modal frequency, as tabulated in Table 9.1 is well below the resonant frequency ranges.

Structural analysis for the above case was executed and the Von-Mises plot is shown in Fig. 9.3. Von-Mises stress is often used to estimate the yield criteria of materials. The von-mises criterion states that, failure will occur, if the von-mises stress reaches a critical limit or yield strength of the material. Thus, FEA analysis identifies the areas where this value is attained, is analyzed and avoided by design changes or strengthen the areas of high stress. For the present study aluminum was used as the material. The yield strength of aluminum is 414 Mpa. From Fig 9.3, the maximum value of the von-mises stress is 47.417 Mpa (SMX), which is less than \(1/3\) times the yield strength. The importance of the above exercise was to show the FEA analysis part of the proposed design system. Also, the above case was executed as a solid body, and simulated like a cantilever beam problem, to make it easier and show the capability of FEA coupling to the system proposed.

Figure 9.1: Meshed GE reverse engineered wind-turbine Blade with 8 node Hexahedron Brick 185 element.
Figure 9.2: First five Modal solution of GE reverse engineered blade.
Table 9.1: First five natural Frequencies in parked condition (GE reverse engineered blade).

<table>
<thead>
<tr>
<th>Mode &amp; Frequencies (Hz)</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1st (flapwise)</td>
<td>4.133 Hz</td>
</tr>
<tr>
<td>2nd (edgewise)</td>
<td>5.577 Hz</td>
</tr>
<tr>
<td>3rd (flapwise)</td>
<td>18.318 Hz</td>
</tr>
<tr>
<td>4th (mixed)</td>
<td>20.778 Hz</td>
</tr>
<tr>
<td>5th (mixed)</td>
<td>26.648 Hz</td>
</tr>
</tbody>
</table>

Figure 9.3: Von-Mises Stress plot on GE reverse engineered blade.
Chapter 10

Conclusion

10.1 Summary

An approach to the wind turbine design from a turbomachinery perspective is presented that can leverage many of the design codes and processes developed for axial turbines, open rotors, axial compressors, and fans. The multi-disciplinary design system has the ability for geometry creation and analysis for axial compressors. It is being adapted for wind turbines which have their own unique issues. The multi-disciplinary approach makes it easy to address a vast number of aerodynamic and structural issues. A parametric design tool for geometry has been developed that will help implement quick design changes from a command line input. The system developed was investigated with the in-house turbomachinery axisymmetric solver, T-Axi, as it was easy to change the source code to suit our needs for wind turbine design. For a conventional horizontal axis wind turbine, the analysis shows:

- The lower 25% of span should account for cascade effects.
- From 25% - 75% span, the wind turbine can be assumed 2D and isolated (wing theory applicable).
- The upper 25% of span has significant 3D effects including the effects of rotation (Coriolis and centrifugal).

10.2 Future Work

Capabilities from a turbomachinery design system have been adapted for use for wind turbines. This approach can add understanding of wind turbines from classical turbomachinery methods. This design system is a foundation and is now extendable. It will be easy to add unique tip treatments, as well as new airfoil sections. The system has been part of an optimization loop, and can easily fit into an optimization
environment. The future work should include the validation of the tool with other available tools for design of wind turbine blade. The presented code should also be tied to an acoustic module for noise prediction from the blades and ways for reduction through design changes. Also other modifications to the blade design such as a parametric tip design for reducing the tip noise effects and improving efficiency will be possible.

Now that a wind turbine blade design system has been established using axial turbomachinery concepts, further analysis to include available blade profiles (NREL airfoils for HWATs [47]) can be used in the code to take the advantage of designing the wind turbine blade in a more realistic manner. Wind turbines must deal with off-design and pitch changes which make them different from axial machines. A methodology that combines the wind turbine feature using T-AXI and conventional Blade element method should be developed.

Distortion analysis to understand the earth’s boundary layer effects and the effects of the pylon on the rotating blades of the wind turbine should also be developed. This should be possible with the Non-Linear Harmonic capability in the 3D CFD code Fine/Turbo, but needs some development.
References


Appendix A

Geometry Development

A.1 Parametric Geometry Development using CAPRI

Computational Analysis Programming Interface) [22],[32] for changing the parameters of the base file through command line and generating a new part file with the changed parameters readily available for further use for CFD or FEA. The basic outline is as shown in Fig. A.1.

The Code is attached in Appendix. B

![Image of CAPRI Flow Chart][32]

Figure A.1: CAPRI Flow Chart [32]

A.2 One-dimensional Grid Clustering

To generate good results from a numerical solution to describe a physical process, a good grid definition must be given to the solver, such that discrete volumes or elements can be identified where conservation laws can be applied. Thus a good quality of grid results in a better solution and conversely a poor quality of grid results in poor results [27].
In this problem of continuous slope disk, the radius and thickness distribution is evenly spaced to form a spline, connecting each coordinate point. The equation governing the relationship between the thickness and radii is given by

\[ t(r) = C_1 r^4 + C_2 r^3 + C_3 r^2 + C_4 r + C_5 \]  
(A.1)

Here ‘t’ stands for thickness and ‘r’ for the radii at different points generated by the code.

The five coefficients \( C_1 \) to \( C_5 \) define the general shape of the curve and can be found from the prescribed slope of the line at two end points and known value of ‘t’ at each control point [21].

Here the distribution as shown in Gutzwiller’s thesis [21] is equal over the length. The curved portion and the straight portion of the disk have equal points. The need is to define the curve more accurately than the straight portion, hence to provide more grid points on the curved part than the straight part.

This is achieved by one dimensional grid clustering given by:

\[ \xi_{xx} = P \]  
(A.2)

Where \( \xi \) is the coordination transformation of ‘X’ axis in \( \xi \) direction. ‘P’ is the source term that makes the one dimensional clustering [27]. The basic idea is to provide clustering at high curvature zones. This is given by:

\[ \frac{d^2 t}{dr^2} \]  
(A.3)

The curvature term here forms the source term. The relation between ‘t’ and ‘r’ is already defined in A.1. Thus the equation becomes:

\[ \frac{(\partial^2 r)}{(\partial \xi^2)} = \frac{d^2 t}{dr^2} \]  
(A.4)

This on solving reduces to:

\[ r_{\xi \xi} = -(r_i^2) \frac{d^2 t}{dr^2} \]  
(A.5)

discretizing the equations with second order accurate central differentiation on LHS and RHS for \( r_{\xi \xi} \) and \( r_\xi \) terms and taking partial differentiation for the ‘t’ term in equation A.1, gives:

\[ \frac{r_{i-1} - 2r_i + r_{i+1}}{(\xi_{i+1} - \xi_i)^2} = -\frac{(r_{i+1} - r_i)}{(\xi_{i+1} - \xi_i)} \left(12C_1 r_i^2 + 6C_2 r_i + 2C_3\right) \]  
(A.6)

This system yields a Tridiagonal Matrix system of equations and solving it gives the distribution of the radial points and corresponding thickness coordinate values are got upon substituting it in equation A.1.
The Code is attached in the Appendix C.
Appendix B

Drive_spline Code

B.1 C Program

/*
***************************************************************************

drive_spline — Soumitr Dey
Base Code Used is DriveMM provided by CADNEXUS

Drives the Master Model through CAPRI
It reads the .din file of the specified disk, (as of now given as Web.din)
Regenerates Automatically with the new parameters, & saves as New_disk.prt

Usage: "drive_spline Modeler partName"

Info is written to standard output therefore piping &
redirection can be used for both input & output

Spline modification is implemented (has issues with CAPRI!! , fixed --> 10/26/09)
*/
Legal Disclaimer ——

All code is presented for reference only. User accepts total responsibility and risk for using code examples and applications derived from the code examples.

---

```c
#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <math.h>
#include "capri.h"
#include "capri_codes.h"

// allocates the length of the array for fortran

#define MAX 100

extern void gi_out( char *str, ... );

/*Fortran call to read .din file */

void read_din_(int *TYPE, float *dRqR, float *wWeb, float *dRqB, float *wBor,
              float *rBor, float *dsf, float *Pr1, float *Pr2, float *Pt1, float *Pt2,
              float *Pt3, float *S12, int *dead_weight_flag, float *RPMo, float *BiRr,
              Tadder, float *RimT, float *TAxis, float *sfact, float *Enum, float *BiRr2,
              float *BeRr, float *wRim, float *Bspn, int *Bshr, float *Brho, float *Bthp,
              float *RtqB, float *Bscale, float *rBor_min, float t[], float r[]);

int
CAPRI_MAIN(int argc, char *argv[])
int i, j, k, status, mmindex, np, mfile, type, len, bra, bflg, done
    , npar, nbra, model, regn=0;
    units, nvol, spar[200], omdl;
int handle, l=1;
char *name, filename[100], lname[100], *string, *CADtype, c;
double val, *reals, lreal[2], *crd, *spts;
spts = malloc(2000);
char *server = 0;
FILE *fp, *tmp;
/* specify not to automatically tessellate */
gi_putenv("CAPRItess=OFF");
/* activate spline support */
//gi_putenv("CAPRI spline=NEW");
/* Check for client/server data */
server = getenv("capriSERVER");
/* start CAPRI */
status = gi_uStart();
gi_out(" gi_uStart status = %d ", status);
if (status != 0) exit(1);
/* load the specified parts */
/* Command Modeler name Model*/
status = gi_uLoadModel(server, argv[1], argv[2]);
printf("argv[1] : %s ", argv[1]);
gi_out(" Part %s: gi_uLoadModel status = %d ", argv[2], status);
/* look at a single Volume / No assemblies! */
/*
    if (gi_uNumVolumes() != 1) {
        gi_out(" ERROR: Only works for a single volume! ");
        gi_uStop(0);
*/
```c
    return 1;
  }
*/

/* get the MM handle from Volume 1 — must be 1 */

mmindex = gi_fMasterModel(status,0);
if (mmindex != 1) {
  gi_out(" gi_fMasterModel status = %d ", status);
  gi_uStop(0);
  return 1;
}

/* get the sizes */

if (mmindex != 1) {
  gi_out(" gi_fMasterModel status = %d ", status);
  gi_uStop(0);
  return 1;
}

/* get the sizes */

if (mmindex != 1) {
  gi_out(" gi_fMasterModel status = %d ", status);
  gi_uStop(0);
  return 1;
}

/* output tree */

for (i = 0; i <= nbra; i++) {
  gi_fGetBranch(mmindex, i, &name, &CADtype, &status, &parent,
                &nchild, &children);
  gi_out(" branch %d: %s (%s), sup = %d ",
          i, name, CADtype, status);
}

/* output parameters */

for (i = 1; i <= npar; i++) {
  gi_fGetParam(mmindex, i, &name, &type, &bflg, &len, &bra);
  gi_out(" param #%d: %s, type = %d, len = %d, bra = %d ",
          i, name, type, len, bra);
  if (type == 0) {
    gi_fGetBool(mmindex, i, &bools);
```
else if (type == 1) {
    gi_fGetInteger(mmindex, i, &ints, lints);
    gi_out(" value is %d ", ints[0]);
}
else if (type == 2) {
    gi_fGetReal(mmindex, i, &reals, lreal, &units);
    for (k = 0; k < len; k++)
        gi_out(" value is %le ", reals[k]);
}
else if (type == 3) {
    gi_fGetString(mmindex, i, &string);
    gi_out(" value is %s ", string);
}
else if (type == 4) {
    gi_fGetSpline(mmindex, i, &deg, &fix, &reals);
    for (k = 0; k < len; k++)
        gi_out(" %.3d %le %le ", k+1, reals[2*k], reals[2*k+1]);
}

// File open for getting the changed parameters
// Calls an external fortran subroutine & gets the values form the .din ← file saved

int Bshr,dead_weight_flag,q,TYPE;
char C1,C2,C3,C4,C5,C6,C7,MATNAME,Bfilename;
float dRqR,wWeb,dRqB,wBor,rBor,dsf,Pr1,Pr2,Pt1,Pt2,Pt3,S12,m_b,r_cg_b,RPMo;
float Tadder,RimT,TAxis,sfact,Bnum,BiRr,BeRr,wRim,Bspn,Brho,Bthp,RtqB,←
wBor_max,rBor_min,yscale;
float r[MAX]; // Radial points array
float t[MAX]; // Thickness points array

// get changes

gi_out(" CHANGES: ");
done = 0;

printf("C Calling Fortran Subroutine ,\n\n\n");
Function Call of the Fortran subroutine for .din file

read_din_(&TYPE,&dRqR,&wWeb,&dRqB,&wBor,&rBor,&dsf,&Pr1,&Pr2,&Pt1,&Pt2,&
Pt3,&S12,
&dead_weight_flag,&RPMo,&Tadder,&RimT,&TAxis,
&sfact,&Bnum,&BiRr,&BeRr,&wRim,&Bspn,&Bshr,&Brho,&Bthp,&
RtqB,&Bscale,&rBor_min,t,r);
printf("done \n");
printf("t: %le \n", t[2]);
printf("r: %le \n",r[2]);

printf("TYPE : %d \n\n",TYPE);

/* deal with branches */
/* if ((strcmp(lname,"BRAN") == 0) || (strcmp(lname,"BRAI") == 0)) {
if (strcmp(lname,"BRAN") == 0) {
  if (scanf("%s", lname) == 0) break;
  for (i = 1; i <= nbra; i++) {
    gi_fGetBranch(mmindex, i, &name, &CADtype, &status, &parent,
    &nchild, &children);
    if (strcmp(name, lname) == 0) break;
  }
  if (i > nbra) {
    gi_out(" ERROR: Branch %s not found! ", lname);
    continue;
  }
} else {
  if (scanf("%d", &i) == 0) break;
  if ((i < 0) || (i > nbra)) {
    gi_out(" ERROR: Branch %d out of range (1-%d)! ", i, nbra);
    continue;
  }
  gi_fGetBranch(mmindex, i, &name, &CADtype, &status, &parent,
  &nchild, &children);
```c
if (scanf("%d", &status) == 0) break;
if ((status != 0) && (status != 1)) {
    gi_out(" ERROR: Suppress out of range -> %d!", status);
    continue;
}

if (gi_fSetSupress(mmindex, i, status)) {
    gi_out(" %s status now = %d", name, status);
} /*
   // deal with parameters
   // Changed Parameters from .din file

   // 1 = web disk, 2 = Hyperbolic disk 3 = Ring disk 4 = Continuos slope ← disk

switch(TYPE) {
    case 1:
        printf("Type of disk is Web disk \n\n ");
        printf ("The Changed values of new .din file : \n\n\n ");
        i=1;
        len=50;
        for (k = 0; k < 50; k++)
            spts[2*k] = -t[k];
        spts[2*k+1] = r[k];
        gi_fSetSpline(mmindex, i, len, spts);
        gi_out("The changed spline data 1 is as below:");
        for (k = 0; k < 50; k++)
            gi_out(" %.3d %le %le ", k+1, spts[2*k], spts[2*k+1]);
        i=2;
        len=50;
        for (k = 0; k < 50; k++)
            spts[2*k] = t[k];
        spts[2*k+1] = r[k];
```
})
gi_fSetSpline(mmindex, i, len, spts);
gi_out("The changed spline data 2 is as below:");
for (k = 0; k < 50; k++)
gi_out("%3d %le %le ", k+1, spts[2*k], spts[2*k+1]);

i=9;
val = rBor_min;
gi_fSetReal(mmindex, i, &val);
gi_out("RBOR value now = %le ", name, val);

i=10;
val = Pt1;
gi_fSetReal(mmindex, i, &val);
gi_out("Pt1 value now = %le ", name, val);

i=15;
val = r[49];
gi_fSetReal(mmindex, i, &val);
gi_out("D1 value now = %le ", name, val);

i=17;
val = BiRr*1;
gi_fSetReal(mmindex, i, &val);
gi_out("BIIR value now = %le ", name, val);

i=19;
val = wRim;
gi_fSetReal(mmindex, i, &val);
gi_out("WRIM value now = %le ", name, val);

i=14;
val = RtqB;
gi_fSetReal(mmindex, i, &val);
gi_out("RTQB value now = %le ", name, val);

i=24;
val = BeRr*1;
gi_fSetReal(mmindex, i, &val);
gi_out("BERR value now = %le ", name, val);

i=25;
val = Bspn;
255   gi_fSetReal(mmindex, i, &val);
256   gi_out("BSPN value now = %le ", name, val);
257   break;
258
259   case 3 : printf("Type of disk is Ring disk \n\n ");
260       printf("The Changed values of new .din file : \n\n\n");
261       i=1;
262       len=50;
263       for (k = 0; k < 50; k++){
264          spts[2*k] = -t[k];
265          spts[2*k+1] = r[k];
266       }
267   gi_fSetSpline(mmindex, i, len, spts);
268   gi_out("The changed spline data 1 is as below:");
269       for (k = 0; k < 50; k++)
270          gi_out(" %.3d %le %le ", k+1, spts[2*k], spts[2*k+1]);
271
272   i=2;
273   len=50;
274       for (k = 0; k < 50; k++){
275          spts[2*k] = t[k];
276          spts[2*k+1] = r[k];
277       }
278   gi_fSetSpline(mmindex, i, len, spts);
279   gi_out("The changed spline data 2 is as below:");
280       for (k = 0; k < 50; k++)
281          gi_out(" %.3d %le %le ", k+1, spts[2*k], spts[2*k+1]);
282
283   i=9;
284   val = rBor_min;
285   gi_fSetReal(mmindex, i, &val);
286   gi_out("RBOR value now = %le ", name, val);
287   i=10;
288   val = Pt1;
289   gi_fSetReal(mmindex, i, &val);
290   gi_out("Pt1 value now = %le ", name, val);
i=15;
val = r[49];
gi_fSetReal(mmindex, i, &val);
gi_out("D1 value now = %le ", name, val);
i=17;
val = BiRr*1;
gi_fSetReal(mmindex, i, &val);
gi_out("BIRR value now = %le ", name, val);
i=19;
val = wRim;
gi_fSetReal(mmindex, i, &val);
gi_out("WRIM value now = %le ", name, val);
i=14;
val = RtqB;
gi_fSetReal(mmindex, i, &val);
gi_out("RTQB value now = %le ", name, val);
i=24;
val = BeRr*1;
gi_fSetReal(mmindex, i, &val);
gi_out("BERR value now = %le ", name, val);
i=25;
val = Bspn;
gi_fSetReal(mmindex, i, &val);
gi_out("BSPN value now = %le ", name, val);
break;

case 4: printf("Type of disk is Continuos slope disk \n\n ");
printf ("The Changed values of new .din file : ,\n\n");

// Setting the new spline points
i=1;
len=50;
for (k = 0; k < 50; k++){
spts[2*k] = -t[k];
spts[2*k+1] = r[k];
}
gi_fSetSpline(mmindex, i, len, spts);

gi_out("The changed spline data 1 is as below:");

for (k = 0; k < 50; k++)
gi_out(" %.3d %le %le ", k+1, spts[2*k], spts[2*k+1]);

i=2;
len=50;
for (k = 0; k < 50; k++){
    spts[2*k] = t[k];
    spts[2*k+1] = r[k];
}
gi_fSetSpline(mmindex, i, len, spts);

for (k = 0; k < 50; k++)
gi_out(" %.3d %le %le ", k+1, spts[2*k], spts[2*k+1]);

i=9;
val = rBor_min;
gi_fSetReal(mmindex, i, &val);

i=10;
val = Pt1;
gi_fSetReal(mmindex, i, &val);

i=15;
val = r[49];
gi_fSetReal(mmindex, i, &val);

i=17;
val = BiRr*1;
gi_fSetReal(mmindex, i, &val);

i=19;
val = wRim;
gi_fSetReal(mmindex, i, &val);

i=19;
val = wRim;
gi_fSetReal(mmindex, i, &val);
i=14;
val = RtqB;
gi_fSetReal(mmindex, i, &val);
gi_out("RTQB value now = %le ", name, val);
i=24;
val = BeRr*l;
gi_fSetReal(mmindex, i, &val);
gi_out("BERR value now = %le ", name, val);
i=25;
val = Bspn;
gi_fSetReal(mmindex, i, &val);
gi_out("BSPN value now = %le ", name, val);
break;

// printf("done\n");
//do {
// if (scanf("%s", lname) == 0) break;
// if ((strcmp(lname,"PARN") != 0) && (strcmp(lname,"PARI") != 0) &&
// (strcmp(lname,"SAVE") != 0) && (strcmp(lname,"EXIT") != 0) &&
// (strcmp(lname,"RECN") != 0)) continue;

/* if ((strcmp(lname,"PARN") == 0) || (strcmp(lname,"PARI") == 0)) */{
if (strcmp(lname,"PARN") == 0) {
if (scanf("%s", lname) == 0) break;
}
for (i = 1; i <= npar; i++) {
    gi_fGetParam(mmindex, i, &name, &type, &bflg, &len, &bra);
    if (strcmp(name,lname) == 0) break;
}
if (i > npar) {
    gi_out(" ERROR: Parameter %s not found! ", lname);
    continue;
}
} else {
    if (scanf("%d", &i) == 0) break;
    if ((i < 0) || (i > npar)) {
        gi_out("ERROR: Parameter %d out of range (1-%d)! ", i, npar);
        continue;
    }
    gi_fGetParam(mmindex, i, &name, &type, &bflg, &len, &bra);
} 
if (len < 0) len = -len;

/* set the value
 if (type == 1)
    if (len == 1) {
        if (scanf("%d", &k) == 0) break;
        gi_fSetInteger(mmindex, i, &k);
        gi_out("%s value now = %d", name, k);
    }
 if (type == 2)
    if (len == 1) {
        if (scanf("%le", &val) == 0) break;
        gi_fSetReal(mmindex, i, &val);
        gi_out("%s value now = %le", name, val);
    }
 if (type == 3) {
    if (scanf("%s", lname) == 0) break;
    gi_fSetString(mmindex, i, lname);
    gi_out("%s value now = %s", name, lname);
}
} */

} 

} /*

// Regenerates with the new data
omdl = gi_fRegenerate(mmindex);


//Save the new part as New_disk

90
if (omdl > CAPRI_SUCCESS) {
    strcpy(lname, "New_disk");
    status = gi_uSaveModel(omdl, lname);
    gi_out(" gi_uSaveModel saved as %s ,status = %d \n\n", lname, status);
    if (omdl!=model) gi_uRelModel(omdl);
} 
else {
    status = gi_uSaveModel(omdl, lname);
    gi_out(" gi_uSaveModel saved as %s ,status = %d \n\n", lname, status);
    if (omdl!=model) gi_uRelModel(omdl);
    regn=0;
}

if (strcmp(lname, "SAVE") == 0) {
    // make the new part
    if (regn == 0) {
        omdl = gi_fRegenerate(mmindex);
        gi_out(" gi_fRegenerate status = %d ", status);
        if (omdl > CAPRI_SUCCESS) {
            scanf("%s", lname);
            status = gi_uSaveModel(omdl, lname);
            gi_out(" gi_uSaveModel saved as %s ,status = %d ", lname, status);
            if (omdl!=model) gi_uRelModel(omdl);
        }
    } else {
        // scanf("%s", lname);
        status = gi_uSaveModel(omdl, lname);
        gi_out(" gi_uSaveModel saved as %s ,status = %d ", lname, status);
        if (omdl!=model) gi_uRelModel(omdl);
        regn=0;
    }
}

if (strcmp(lname, "REGN") == 0) {
    // make the new part
    omdl = gi_fRegenerate(mmindex);
gi_out(" gi_fRegenerate status = %d ", omdl);
regn=1;
}

if (strcmp(lname,"EXIT") == 0) break;
}
while (done == 0);
gi_out(" ");

if (done == 0) {
    gi_uStop(0);
    return 1;
} /*
printf("The Ansys File is Saved as Ansys.ain ,\n\n");
    gi_uStop(0);
return 0;
}"
Appendix C

Read_din Code

C.1 Fortran Program

! File to read the .din folder

subroutine read_din (type, dRqR, dRgB, wBor, rBor, dsf, Pr1, Pr2, Pt1, Pt2, Pt3, S12, &
    dead_weight_flag, RPMo, Tadder, RimT, TAxis, &
    sfact, Bnum, BiRr, BeRr, wRim, Bspn, Bshr, Brho, Bthp, &
    RtqB, Bscale, &
    rBor_min, t, r)

implicit none

real :: dRqR, dRgB, wBor, rBor, dsf, Pr1, Pr2, Pt1, Pt2, Pt3, S12
real :: M_B, R_CG_B, RPMO, TADDER, RIMT, TAXIS, SFACT, BNUM, BIRR, BERR, WRIM
real :: BSPN, BRHO, BTHP, RTQB, WBOR_MAX, RBOR_MIN, BSCALE, tempr
integer :: BSHR, type, DEAD_WEIGHT_FLAG, rinc, tinc
character*60fname, fext, temp
character (len=60) :: MATNAME, iname, wRimC, RimTC, TAxisC, RtqBC, RPMoC, BnumC, ← ARG1, &
    BspnC, BrhoC, BiRrC, BeRrC, sfactC, TadderC, wBor_maxC, rBor_minC, &
    BthpC, m_bC, r_cg_Bc, C1, C2, C3, C4, C5, C6, C7
character (len=256) :: Bfilename
real, dimension(200) :: r, t, rmax1, rmin1, tmax1, tmin1
integer :: j, n, num, i, n1, k
NAMELIST /input/output/  C1, C2 ,type,C3,dRqR, wWeb, dRqB, wBor, rBor, dsf, Pr1, ←
   Pr2, Pt1, Pt2, Pt3, S12, &
   C4, MATNAME, C5, dead_weight_flag, m_b, r_cg_b, C6, ←
   RPMo, Tadder, RimT, TAxis, &
   sfact, C7, Bnum, BiRr, BeRr, wRim, Bspn, Bshr, Brho, Bthp, ←
   RtqB, wBor_max, rBor_min, Bfilename, Bscale

fname = 'Web' // '.din'
open(2, file=fname, status='old')
read(2, NML=INPUT_OUTPUT)

close(2)

if(type .eq. 2) THEN
   ! Hyperbolic disk to be done
   stop
else if(type .eq. 4) then
   ! Continuous Slope Disk
   num=50
   call david_disk(wRim, BiRr, BeRr, Pr1, Pr2, Pt1, Pt2, Pt3, num, r, t, RtqB, Bspn, ←
                  S12)
   !fname = 'C_s' // '.tab'
   ! print *, 'Fname : ', trim (fname)
   ! open(5, file=fname)
   ! print *, 'temp : ', temp

   ! Checks in the file for the character r(m) & jumps one line
   ! do while (temp .ne. 'r(m)')
   ! read (5, *) temp
   ! enddo
   ! print *, 'temp : ', temp
n= 50

do j=1,n
  !read(5,*)r_r(j),t_t(j)
t(j) = 1000*t(j)/2.
r(j) = 1000*r(j)
enddo

!print*,t_t(i),r_r(i)
!close(5)
endif

return

call ANSYS_WRITER (r,t,r_cg_b,m_b,num,RPMo,Tadder,RimT,TAxis,Bnum,MATNAME)
end subroutine read_din

! *************DISK GENERATION SUBROUTINE – Modified DAVID’S METHOD (by ➔
            Soumit 11/25/09)*************************
subroutine david_disk(wRim,BiRr,BeRr,Pr1,Pr2,Pt1,Pt2,Pt3,num,r,t,RtqB,Bspn➔
                      ,S12)
implicit none
integer :: i,i1,i2,n1,n2,n3,j,k
integer,intent(in out)::num
real, dimension(num), intent(in out) :: r,t
real, intent(in out) :: Pr1,Pr2,Pt1,Pt2,Pt3,wRim,BiRr,BeRr,RtqB,Bspn,S12
real :: r1,r2,r3,r4,t1,t2,t3,t4,SecP,S1,S2,S3,S4,C1,C2,C3,C4,C5,tm,x,xm➔
        ,S34,Pr3,p2,p3
real, dimension (num) :: a,b,d,RHS,x1,p

!Calculates control points in true dimensions
Pr3 = 0.95
r4=((BiRr+BeRr)/2.−RtqB*Bspn)
r3=Pr3*(r4)
rl=Pr1*(r4)
r2=rl+Pr2*(r3-rl)
t1=Pt1*wRim
85 t2=Pt2*wRim
86 t3=Pt3*wRim
87 t4=wRim
88
89 S2=(t3-t2)/(1.)
90 S3=S2
91 S4=0.0
92
93 n1=22 ! no of points on the curved section r1 -- r2
94 n3=n1 ! no of points on the curved section r3 -- r4
95 n2=6 ! no of points on the straight section r2 -- r3
96
97 ! Initial Radial Points Created
98 do i=1,num,1
99 r(i) = r1+(i*1.0-1.)/(num*1.0-1.)*(r4-r1)
100 enddo
101
102 ! Builds up thickness values
103 p=0
104 t=0
105 call thickness(num,r1,r2,r3,r4,t1,t2,t3,t4,S12,S1,S2,S3,S4,r,p,t)
106
107 ! print *, 'r1:',r1,'r2:',r2,'r3:',r3,'r4:',r4
108 ! print *, 'P=',
109 ! do i=1,num,1
110 ! print *, p(i)
111 ! enddo
112
113 ! print *, ''
114 ! print *, 'Initial R,T'
115 ! do i=1,num,1
116 ! print *, 'i, r(i), t(i)
117 ! enddo
118
119
1 One dimensional Clustering Algorithm
2 Xzz = (−Xz)³ * P
3 X & z transformation coordinates
4 P Source term form one dimensional elliptic clustering
5 do i=1,num,1
6  if (i==1) then
7    d(i)=1!BC
8    a(i)=0!BC
9    RHS(i)=r1
10 endif
11 endif (i==num) then
12    d(i)=1!BC
13    b(i)=0!BC
14    RHS(i)=r4
15 endif (i==i1) then
16    a(i)=0
17    d(i)=1
18    b(i)=0
19    RHS(i)=r2
20 endif (i==i2) then
21    a(i)=0
22    d(i)=1
23    b(i)=0
24    RHS(i)=r3
25 else
26    a(i)=1
27    d(i)=-2
28    b(i)=1
29    RHS(i)= p(i)*((r(i-1)−r(i+1))**3)
30 endif
31 enddo
156  ! print * , ''
157  ! print * , 'r1 : ' , r1 , ' r2 : ' , r2 , ' r3 : ' , r3 , ' r4 : ' , r4
158  ! print * , ''
159  ! print * , 'RHS= ' 
160  ! do i=1,num,1
161  !       print * , RHS(i)
162  ! enddo
163
164  ! sets up new radial stations with clustering at the curved sections
165  call thomas(a,b,d,RHS,num)
166  do i=1,num,1
167     r(i) = RHS(i)
168  enddo
169
170  ! sets up new radial stations and clusters the points on the curved ←
171  ! sections
172  do i=1,num,1
173     if ( i .LT. n1) then
174     r(i) = r1+(i*1.0-1.)/(n1*1.0-1.)*(r2-r1)
175     elseif ((i .EQ. n1)) then
176     r(i) = (r2+r(i-1))/2.
177     elseif ((i .GT. n1).AND. (i .LT. (n1+n2))) then
178     r(i) = r2 + ((i-n1)*1.0-1.)/(n2*1.0-1.)*(r3-r2)
179     ! elseif ((i .EQ.(n1+n2))) then
180     ! r(i) = (r3+r(i-1))/2.
181     elseif ((i .GE. (n1+n2)).AND.(i .LE. num)) then
182     r(i) = r3+((i-(n1+n2))*1.0-1.)/(n3*1.0-1.)*(r4-r3)
183
endif
enddo

! print *,''
! print *,'RHS1='
! do i=1,num,1
! print *,RHS(i)
! enddo
!
call thickness(num,r1,r2,r3,r4,t1,t2,t3,t4,S12,S1,S2,S3,S4,r,p,t)

! print *,''
! print *,'Final R,T'
! print *,'r, t'
! do i=1,num,1
! print *,i,r(i),'' ',t(i)
! enddo
!
! to write certain values to a file format
! OPEN(10, file = 'datfile.dat')
! WRITE(10,*) 'Final R,T'
! do i=1,num,1
! WRITE(10,*) i, '' ',r(i),'' ',t(i)
! enddo
! CLOSE(10)
!
end subroutine david_disk

!******** Thickness Builder***********
subroutine thickness(num,r1,r2,r3,r4,t1,t2,t3,t4,S12,S1,S2,S3,S4,r,p,t)

real, dimension(num), intent(in out) :: r,t,p
integer, intent(in out) :: num
real, intent(in out) :: r1,r2,r3,r4,t1,t2,t3,t4,S12,S1,S2,S3,S4
integer :: i,i2,i1
real :: SecP,C1,C2,C3,C4,C5,tm,x,xm,S34,Pr3,p2,p3
do i=1,num,1
  if ((r(i) .GE. r1) .AND. (r(i) .LT. r2)) then
    tm=(t2-t1)*(S12)+t1
    xm =0.5
    S1=-(t1-tm)/(xm)
    C1=t1
    C2=S1
    C3=(-t1+tm*S1*xm+(4*t1-4*t2+3*S1+S2)*xm**3-(3*t1-3*t2+2*S1+S2)*xm**2)/((-1+xm)**2*xm**2)
    C4=(2*t1-2*tm+2*S1*xm-(4*t1-4*t2+3*S1+S2)*xm**2+(2*t1-2*t2+S1+S2)*xm**2)/((-1+xm)**2*xm**2)
    C5=(-t1+tm*S1*xm+(3*t1-3*t2+2*S1+S2)*xm**2-(2*t1-2*t2+S1+S2)*xm**3)**2/(((-1+xm)**2*xm**2)
    x = (r(i)-r1)/(r2-r1)
    t(i) = C1+C2*x+C3*x**2+C4*x**3+C5*x**4
    p2= (1/(r2-r1))*2
    p(i)= p2*(((1.5*C5)*x**2)+((0.75*C4)*x)+(0.25*C3))
    i1 = i
  elseif ((r(i) .GE. r2) .AND. (r(i) .LT. r3)) then
    SecP = (r(i) - r2)/(r3-r2)
    t(i) = t2+(t3-t2)*SecP
    p(i)=0.0
    i2 = i
  elseif ((r(i) .GE. r3)) then
    S34 = 0.5
    tm=(t4-t3)*(S34)+t3
    C1=t3
    C2=S3
    C3=(-t3+tm*S3*S34+(4*t3-4*t4+3*S3+S4)*S34**3-(3*t3-3*t4+2*S3+S4)*S34**2)/((-1+S34)**2*S34**2)
    C4=(2*t3-2*tm+2*S3*S34-(4*t3-4*t4+3*S3+S4)*S34**2+(2*t3-2*t4+S3+S4)*S34**4)/((-1+S34)**2*S34**2)
    C5=(-t3+tm*S3*S34+(3*t3-3*t4+2*S3+S4)*S34**2-(2*t3-2*t4+S3+S4)*S34**2)
  endif
x = (r(i) - r3) / (r4 - r3)
t(i) = C1 + C2 + x3 + C3 + x2 + C4 + x + C5 + x2 + C6 + x

p3 = (1 / (r4 - r3)) ** 2

p(i) = p3 * (((1.5 * C5) * x2) + ((0.75 * C4) * x) + (0.25 * C3))

end subroutine

subroutine thomas(a, b, d, RHS, num)

integer, intent(in out) :: num
real, dimension (num), intent (in out) :: a, b, d, RHS
real(KIND=8) :: R
integer :: i, j

**************THOMAS SOLVER**********************

! SOLVES A TRI–DIAGONAL SYSTEM
! b(i)*r(i−1) + d(i)*r(i)+a(i)* r(i+1) = rhs(i)
! [ d1 a1 ... ]
! [ b2 d2 a2 ... ]
! [ .... b3 d3 a3 .... ]
! [ ...... ..... b4 d4 a4 ]
! [ ...... ...... b5 d5 ]
! a--> upper diagonal
! d--> diagonal
! b--> lower diagonal
! RHS --> right hand side

subroutine thomas(a, b, d, RHS, num)

implicit none
integer, intent(in out) :: num
real, dimension (num), intent (in out) :: a, b, d, RHS
real(KIND=8) :: R
integer :: i, j

**************SOLVES THE RESULTING TRIDIAGONAL SYSTEM**************
!! Establish Upper Triangular Matrix

if (num .eq. 1) then
    RHS(1)= RHS(1)/d(1)
endif

do  i=2,num
    if (d(i-1) .eq. 0.0) then
        print*,"ERROR: Zero in the denominator"
        stop
    endif
    R = b(i)/d(i-1)
    d(i) = d(i) - R*a(i-1)
    RHS(i) = RHS(i) - R*RHS(i-1)
enddo

! Back Substitution
RHS(num) = RHS(num)/d(num)
j=num
do  i=2,num
    j=num + 1 -i
    RHS(j) = (RHS(j)-a(j)*RHS(j+1))/d(j)
endo
end subroutine thomas
Appendix D

ANSYS Code

D.1 Fortran Program

! ------------------------SAVE ANSYS FILE-------------------------------

subroutine ANSYS_WRITER (r,t,r_cg_b,m_b,num,RPMo,Tadder,RimT,TAxis,Bnum,"MATNAME")

implicit none

integer :: i,istat
real, dimension(200), intent(in) :: r,t
real, dimension(:), allocatable :: e_array,t_array,rho_array,v_array,"alpha_array, &
                                 fty_array,ftu_array
real,intent(in) :: m_b,r_cg_b,RPMo,TAxis,RimT,Tadder,Bnum
real :: ttemp,stemp,omega,pi,atemp,btemp
!
real :: ISOFLAG,ET_EL,STT_STL,SCT_STT,SCL_STL,AT_AL
integer :: num,num2
character (len=256) :: Afilename,fname
character (len=256) :: line,r1,r2,t1,t2,temp1,temp2
!
character (len=60),intent(in) :: MATNAME
!

****************************************************************************************************************************
! Opens the material data file to read the values from it
! Change the fnmae command to the directory where the MATNAME file is for further use

```
fname="/home/soumitr/CADNEXUS/CADNexux/V3.12_LINUX64/Examples/Examples/
" // MATNAME
```

```
open(10, file=fname, status='old')
read(10,*)
read(10,*)
read(10,*)
read(10,*)
read(10,*)
read(10,*)
read(10,*)
read(10,*)
```

```
um2=1
```

```
do
read(10,*,iostat=istat)
if(istat == -1) then
exit
endif
num2=num2+1
enddo
```

```
close(10)
```

```
allocate(t_array(num2))
allocate(e_array(num2))
allocate(v_array(num2))
allocate(alpha_array(num2))
allocate(rho_array(num2))
allocate(fty_array(num2))
allocate(ftu_array(num2))
```

```
open(20, file=fname )
read(20,*)
read(20,*)
```
read(20,*)
read(20,*)
read(20,*)ISOFLAG,ET_EL,STT_STL,SCT_STT,SCL_STL,AT_AL
read(20,*)
do i = 1,num2,1
read(20,*,iostat=istat)t_array(i),alpha_array(i),v_array(i),e_array(i), &
fty_array(i),ftu_array(i),rho_array(i)
enddo
close(20)

! Writing out the ANSYS .AIN file
OPEN (UNIT=10,file = 'Ansys.ain')
write(10,*) "'/prep?"
pi = 4*ATAN(1.0)
omega = (RPMo*2.*pi)/60.

! MAKES MATERIAL FILE
do i = 1,SIZE(t_array)-1,1
write(temp1,*)t_array(i)
line = "MPTEMP," // trim(temp1)
! print*,trim(line)
write(10,*) trim(line)
enddo
do i = 1,SIZE(e_array)-1,1
write(temp1,*)abs(e_array(i)*10**9)
line = "MPDATA,EX,1," // trim(temp1)
write(10,*) trim(line)
line = "MPDATA,EX,2," // trim(temp1)
write(10,*) trim(line)
write(temp1,*)abs(alpha_array(i)/1000000.)
line = "MPDATA,ALPX,1," // trim(temp1)
write(10,*) trim(line)
line = "MPDATA,ALPX,2," // trim(temp1)
write(10,*) trim(line)
write(temp1,*) abs(v_array(i))
line = "MPDATA,PRXY,1," // trim(temp1)
write(10,*) trim(line)
line = "MPDATA,PRXY,2," // trim(temp1)
write(10,*) trim(line)
write(temp1,*) abs(rho_array(i))
line = "MPDATA,DENS,1," // trim(temp1)
write(10,*) trim(line)
line = "MPDATA,DENS,2,,0"
write(10,*) trim(line)
enddo

print*, r_cg_b
write(temp1,*) r_cg_b
line = "*set,r_cg_b," // trim(temp1)
write(10,*) trim(line)
write(temp1,*)(m_b*Bnum)
line = "*set,total_dead_weight," // trim(temp1)
write(10,*) trim(line)
write(temp1,*)(r(num)/1000.)
line = "*set,rimx," // trim(temp1)
write(10,*) trim(line)
write(10,*) "CSYS,0"
write(10,*) "ET,1,MESH200,6"
write(10,*) "ET,2,MASS21"
write(10,*) "ET,3,PLANE42,,,1"
write(10,*) "ET,4,PIPE16"
write(10,*) "ET,5,SOLID185"
write(10,*) " "
write(10,*)  "R,1,total_dead_weight,total_dead_weight,←
           total_dead_weight"
write(10,*)  "R,2,.1,.05"
write(10,*)  " "

! BUILDS UP DISK SHAPE WITH KEYPOINTS
do i=1,num,1
write(r1,*)(r(i)/1000.)
ttemp = (t(i)/1000.)
write(t1,*)ttemp
line = "k,," // trim(r1) // "," //trim(t1)
write(10,*) trim(line)
enddo

do i=num,1,-1
write(r1,*)(r(i)/1000.)
ttemp = -(t(i)/1000.)
write(t1,*)ttemp
line = "k,," // trim(r1) // "," //trim(t1)
write(10,*) trim(line)
enddo

! CONNECTS KEYPOINTS WITH LINES & SIZES MESH
do i=1,(num*2-1),1
write(temp1,*)i
write(temp2,*)(i+1)
line = "l," // trim(temp1) // "," // trim(temp2)
write(10,*) trim(line)
 enddo
write(temp1,*)(num*2)
line = "l,1," // trim(temp1)
   write(10,*) trim(line)
write(10,*)  "lsel,all"
write(10,*)  "lesize,all,,1"
write(10,*)  "lsel,none"
write(temp1,*)(num)
line = "lsel,s,line,," // trim(temp1)
    write(10,*) trim(line)
write(temp1,*)(2*num)
line = "lsel,a,line,," // trim(temp1)
    write(10,*) trim(line)
write(10,*) "/rep"
stemp = (((r(num)-r(1))/1000.)*(1./num))
write(temp1,*)stemp
line = "lesize,all," // trim(temp1) // ",,,1,,,1,0,"
write(10,*) trim(line)
write(10,*) " 

! CREATES AREAS AND MESHES WITH PLANE14 ELEMENTS
write(10,*) "allsel"
write(10,*) "al,all"
write(10,*) " 

write(10,*) atemp
write(temp1,*)atemp
line = "aesize,all," // trim(temp1)
write(10,*) trim(line)
write(10,*) "type,3"
write(10,*) "mat,1"
write(10,*) "real,1"
write(10,*) "amesh,all"
write(10,*) "arefine,all,,,1"
write(10,*) "type,5"

! CREATES VOLUMES AND MESHES WITH SOLID185 ELEMENTS
write(temp1,*)atemp
line ="k, 0,"//trim(temp1)
write(10,*) trim(line)
write(temp1,*) btemp
line = "k,, 0," // trim(temp1)
write(10,*) trim(line)
write(10,*) "vrotate,all,,/,,101,102,360,10"
write(10,*) "vsweep,all,,1"

! ! POINT MASS FOR THE DEAD WEIGHT
!
write(10,*) ""
write(10,*) "nse1,none"
write(10,*) "type,2"
write(temp1,*) r_cg_b

line = "N.," // trim(temp1) // ",,0,0"
write(10,*) trim(line)
write(10,*) "*get,deadweight,NODE,NUM,MAX"
write(10,*) "E,deadweight"
write(temp1,*) (num)
line = "lse1,s,line,," // trim(temp1)
write(10,*) trim(line)
write(10,*) "nse1"
write(10,*) "ks11"

write(10,*) "type,4"
write(10,*) "real,2"
write(10,*) "mat,2"

write(10,*) "*get,numnodes,NODE,0,count"
write(10,*) "*do,i,1,numnodes"
write(10,*) "*get,nodenum,NODE, ,num,MIN"
write(10,*) "nse1,a,node,,deadweight"
write(10,*) "E,deadweight,nodenum"
write(10,*) "nse1,u,node,,deadweight"
write(10,*) "nse1,u,node,,nodenum"
write(10,*) "*endo"
! write(10,*)  "nslk"
!
write(10,*)  "* get, numnodes, NODE, 0, count"
!
write(10,*)  "* do, i, 1, numnodes"
!
write(10,*)  "* get, nodenum, NODE, , num, MIN"
!
write(10,*)  "nsel, a, node, , deadweight"
!
write(10,*)  "E, deadweight, nodenum"
!
write(10,*)  "nsel, u, node, , deadweight"
!
write(10,*)  "nsel, u, node, , nodenum"
!
write(10,*)  "* enddo"
!
write(10,*)  "allsel"
!
!
! APPLY ROTATIONAL LOADING AND TEMPERATURE GRADIENT.

write(temp1,*)(omega)
!
line = "omega,"  // trim(temp1)
!
write(10,*)  trim(line)
!
write(temp1,*)(Taxis)
!
line = "SET,Taxis,"  // trim(temp1)
!
write(10,*)  trim(line)
!
write(temp1,*)(RimT)
!
line = "*SET,RimT,"  // trim(temp1)
!
write(10,*)  trim(line)
!
write(temp1,*)(TAdder)
!
line = "*SET,TAdder,"  // trim(temp1)
!
write(10,*)  trim(line)
!
write(10,*)  "allsel"
!
write(10,*)  "* get, nodemax, NODE, , NUM, MAX"
!
write(10,*)  "* do, i, 1, nodemax"
!
write(10,*)  "nsel, s, node, , i"
!
write(10,*)  "* get, nodex, NODE, i, mxloc, x"
!
write(10,*)  "nodetemp = Taxis + TAdder + (RimT-Taxis)*(nodex/(rimx))"
!
write(10,*)  "BF, i, temp, nodetemp"
! APPLY ROTATIONAL LOADING AND TEMPERATURE GRADIENT.
!
!
!! MACRO — PLOT VM STRESSES ALONG CENTERLINE
!
! write(10,*) "CREATE,CENTERSTRESS,mac"
! write(10,*) "POST1"
! write(10,*) "PATH,CENTER,2,30,500"
! write(r1,*) (r(1)/1000.)
! write(r2,*) (r(num)/1000.)
! line = "PPATH,1,," // trim(r1) // ",0,0"
! write(10,*) line
! line = "PPATH,2,," // trim(r2) // ",0,0"
! write(10,*) line
! write(10,*) "PDEF,VonMises,S,EQV"
! write(10,*) "PDEF,Sigma_R,S,X"
! write(10,*) "PDEF,Sigma_T,S,Z"
! write(10,*) "Sigma_Z,S,Y"
! write(10,*) "PLPATH,VonMises,Sigma_R,Sigma_T,Sigma_Z"
! write(10,*) "*END"
!
!
!! MACRO — PLOTS 2D DISK
!
write(10,*) "CREATE,PLOTDISK2,mac"
write(10,*) "/EXPAND,0"
write(10,*) "/rep"
write(10,*) "*END"
!
!
!BUILDS UP CUSTOM ANSYS TOOLBAR.
write(10,*)  "/NOPR"
write(10,*)  "*ABB,SAVE_DB ,SAVE"
write(10,*)  "*ABB,RESUM_DB,RESUME"
write(10,*)  "*ABB,QUIT ,Fnc_/EXIT"
write(10,*)  "*ABB,POWRGRPH,Fnc_/GRAPHICS"
write(10,*)  "*ABB,REFRESH ,/REP"
write(10,*)  "*ABB,2D-PLOT ,PLOTDISK2"
write(10,*)  "*ABB,VM- PLOT ,STRESSPLOT"
write(10,*)  "*ABB,VM-CENT ,CENTERSTRESS"
write(10,*)  "/GO"

CLOSE (10)

end subroutine ANSYS_WRITER
Appendix E

Output

E.1 Sample Output

1  CAPRI Info: Dynamic Back−End Loader Ver 3.12
2          Built on <-Mar 5 2010->, build number 6810
3  gi_uStart status = 0
4  CAPRI Info: Attempting to load capriUG.so for NX6
5  CAPRI Info: Loading UniGraphics NX6 Back−End Ver 3.12
6          Built on <-Mar 5 2010->, build number 6810
7  File→ /home/soumitr/CADNEXUS/CADNexus_V3.12_LINUX64/lib/→
capiUG.so
8  CAPRI Warning: Opening file→ new_continuous.capri (gi_readbound)
9                  No Boundary data found!
10
11  CAPRI Warning: Vol 1−IO state = 0 (gi_fillbound)
12  CAPRI Warning: IO state = 0 (gi_closebound)
14  Feature Tree Information
15  branch 0: new_continuous (ROOT),  sup = −1
16  branch 1: DATUM_CSYS(0) (DATUM_CSYS),  sup = 0
17  branch 2: SKETCH_000:SKETCH(33) (SKETCH),  sup = 0
18  branch 3: REVOLVED(35) (SWP104),  sup = 0
19  param #1: Spline61,  type = 4,  len = 50,  bra = 2
<p>| | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>21</td>
<td>001</td>
<td>$-1.522156e+01$</td>
<td>$9.000000e+01$</td>
<td></td>
</tr>
<tr>
<td>22</td>
<td>002</td>
<td>$-1.481840e+01$</td>
<td>$9.285480e+01$</td>
<td></td>
</tr>
<tr>
<td>23</td>
<td>003</td>
<td>$-1.439720e+01$</td>
<td>$9.570950e+01$</td>
<td></td>
</tr>
<tr>
<td>24</td>
<td>004</td>
<td>$-1.396900e+01$</td>
<td>$9.856430e+01$</td>
<td></td>
</tr>
<tr>
<td>25</td>
<td>005</td>
<td>$-1.344560e+01$</td>
<td>$1.014190e+02$</td>
<td></td>
</tr>
<tr>
<td>26</td>
<td>006</td>
<td>$-1.276710e+01$</td>
<td>$1.042738e+02$</td>
<td></td>
</tr>
<tr>
<td>27</td>
<td>007</td>
<td>$-1.190150e+01$</td>
<td>$1.071286e+02$</td>
<td></td>
</tr>
<tr>
<td>28</td>
<td>008</td>
<td>$-1.084480e+01$</td>
<td>$1.099833e+02$</td>
<td></td>
</tr>
<tr>
<td>29</td>
<td>009</td>
<td>$-9.621100e+00$</td>
<td>$1.128381e+02$</td>
<td></td>
</tr>
<tr>
<td>30</td>
<td>010</td>
<td>$-8.282700e+00$</td>
<td>$1.156928e+02$</td>
<td></td>
</tr>
<tr>
<td>31</td>
<td>011</td>
<td>$-6.909500e+00$</td>
<td>$1.185476e+02$</td>
<td></td>
</tr>
<tr>
<td>32</td>
<td>012</td>
<td>$-5.609800e+00$</td>
<td>$1.214024e+02$</td>
<td></td>
</tr>
<tr>
<td>33</td>
<td>013</td>
<td>$-4.519900e+00$</td>
<td>$1.242571e+02$</td>
<td></td>
</tr>
<tr>
<td>34</td>
<td>014</td>
<td>$-3.804000e+00$</td>
<td>$1.271119e+02$</td>
<td></td>
</tr>
<tr>
<td>35</td>
<td>015</td>
<td>$-3.619600e+00$</td>
<td>$1.299666e+02$</td>
<td></td>
</tr>
<tr>
<td>36</td>
<td>016</td>
<td>$-3.591100e+00$</td>
<td>$1.328214e+02$</td>
<td></td>
</tr>
<tr>
<td>37</td>
<td>017</td>
<td>$-3.562600e+00$</td>
<td>$1.356761e+02$</td>
<td></td>
</tr>
<tr>
<td>38</td>
<td>018</td>
<td>$-3.534000e+00$</td>
<td>$1.385309e+02$</td>
<td></td>
</tr>
<tr>
<td>39</td>
<td>019</td>
<td>$-3.505500e+00$</td>
<td>$1.413857e+02$</td>
<td></td>
</tr>
<tr>
<td>40</td>
<td>020</td>
<td>$-3.477000e+00$</td>
<td>$1.442404e+02$</td>
<td></td>
</tr>
<tr>
<td>41</td>
<td>021</td>
<td>$-3.448500e+00$</td>
<td>$1.470952e+02$</td>
<td></td>
</tr>
<tr>
<td>42</td>
<td>022</td>
<td>$-3.419900e+00$</td>
<td>$1.499499e+02$</td>
<td></td>
</tr>
<tr>
<td>43</td>
<td>023</td>
<td>$-3.391400e+00$</td>
<td>$1.528047e+02$</td>
<td></td>
</tr>
<tr>
<td>44</td>
<td>024</td>
<td>$-3.362900e+00$</td>
<td>$1.556595e+02$</td>
<td></td>
</tr>
<tr>
<td>45</td>
<td>025</td>
<td>$-3.334300e+00$</td>
<td>$1.585142e+02$</td>
<td></td>
</tr>
<tr>
<td>46</td>
<td>026</td>
<td>$-3.305800e+00$</td>
<td>$1.613690e+02$</td>
<td></td>
</tr>
<tr>
<td>47</td>
<td>027</td>
<td>$-3.277300e+00$</td>
<td>$1.642237e+02$</td>
<td></td>
</tr>
<tr>
<td>48</td>
<td>028</td>
<td>$-3.248800e+00$</td>
<td>$1.670785e+02$</td>
<td></td>
</tr>
<tr>
<td>49</td>
<td>029</td>
<td>$-3.220200e+00$</td>
<td>$1.699333e+02$</td>
<td></td>
</tr>
<tr>
<td>50</td>
<td>030</td>
<td>$-3.191700e+00$</td>
<td>$1.727880e+02$</td>
<td></td>
</tr>
<tr>
<td>51</td>
<td>031</td>
<td>$-3.163200e+00$</td>
<td>$1.756428e+02$</td>
<td></td>
</tr>
<tr>
<td>52</td>
<td>032</td>
<td>$-3.134700e+00$</td>
<td>$1.784975e+02$</td>
<td></td>
</tr>
<tr>
<td>53</td>
<td>033</td>
<td>$-3.106100e+00$</td>
<td>$1.813523e+02$</td>
<td></td>
</tr>
<tr>
<td>54</td>
<td>034</td>
<td>$-3.077600e+00$</td>
<td>$1.842071e+02$</td>
<td></td>
</tr>
<tr>
<td>55</td>
<td>035</td>
<td>$-3.049100e+00$</td>
<td>$1.870618e+02$</td>
<td></td>
</tr>
<tr>
<td>56</td>
<td>036</td>
<td>$-3.020600e+00$</td>
<td>$1.899166e+02$</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
<td></td>
</tr>
<tr>
<td>57</td>
<td>037</td>
<td>-2.992000e+00</td>
<td>1.927713e+02</td>
<td></td>
</tr>
<tr>
<td>58</td>
<td>038</td>
<td>-2.963500e+00</td>
<td>1.956261e+02</td>
<td></td>
</tr>
<tr>
<td>59</td>
<td>039</td>
<td>-2.935000e+00</td>
<td>1.984809e+02</td>
<td></td>
</tr>
<tr>
<td>60</td>
<td>040</td>
<td>-2.906500e+00</td>
<td>2.013356e+02</td>
<td></td>
</tr>
<tr>
<td>61</td>
<td>041</td>
<td>-2.877900e+00</td>
<td>2.041904e+02</td>
<td></td>
</tr>
<tr>
<td>62</td>
<td>042</td>
<td>-2.849400e+00</td>
<td>2.070451e+02</td>
<td></td>
</tr>
<tr>
<td>63</td>
<td>043</td>
<td>-2.820900e+00</td>
<td>2.098999e+02</td>
<td></td>
</tr>
<tr>
<td>64</td>
<td>044</td>
<td>-2.792400e+00</td>
<td>2.127546e+02</td>
<td></td>
</tr>
<tr>
<td>65</td>
<td>045</td>
<td>-2.763800e+00</td>
<td>2.156094e+02</td>
<td></td>
</tr>
<tr>
<td>66</td>
<td>046</td>
<td>-2.735300e+00</td>
<td>2.184642e+02</td>
<td></td>
</tr>
<tr>
<td>67</td>
<td>047</td>
<td>-3.434800e+00</td>
<td>2.213189e+02</td>
<td></td>
</tr>
<tr>
<td>68</td>
<td>048</td>
<td>-7.745400e+00</td>
<td>2.241737e+02</td>
<td></td>
</tr>
<tr>
<td>69</td>
<td>049</td>
<td>-1.339190e+01</td>
<td>2.270284e+02</td>
<td></td>
</tr>
<tr>
<td>70</td>
<td>050</td>
<td>-1.800000e+01</td>
<td>2.313106e+02</td>
<td></td>
</tr>
</tbody>
</table>

param #2: Spline60, type = 4, len = 50, bra = 2

<p>| | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>71</td>
<td>001</td>
<td>1.522156e+01</td>
<td>9.000000e+01</td>
</tr>
<tr>
<td>72</td>
<td>002</td>
<td>1.481840e+01</td>
<td>9.285480e+01</td>
</tr>
<tr>
<td>73</td>
<td>003</td>
<td>1.439720e+01</td>
<td>9.570950e+01</td>
</tr>
<tr>
<td>74</td>
<td>004</td>
<td>1.396900e+01</td>
<td>9.856430e+01</td>
</tr>
<tr>
<td>75</td>
<td>005</td>
<td>1.344560e+01</td>
<td>1.014190e+02</td>
</tr>
<tr>
<td>76</td>
<td>006</td>
<td>1.276710e+01</td>
<td>1.042738e+02</td>
</tr>
<tr>
<td>77</td>
<td>007</td>
<td>1.190150e+01</td>
<td>1.071286e+02</td>
</tr>
<tr>
<td>78</td>
<td>008</td>
<td>1.084480e+01</td>
<td>1.099833e+02</td>
</tr>
<tr>
<td>79</td>
<td>009</td>
<td>9.621100e+00</td>
<td>1.128381e+02</td>
</tr>
<tr>
<td>80</td>
<td>010</td>
<td>8.282700e+00</td>
<td>1.156928e+02</td>
</tr>
<tr>
<td>81</td>
<td>011</td>
<td>6.909500e+00</td>
<td>1.185476e+02</td>
</tr>
<tr>
<td>82</td>
<td>012</td>
<td>5.609800e+00</td>
<td>1.214024e+02</td>
</tr>
<tr>
<td>83</td>
<td>013</td>
<td>4.519900e+00</td>
<td>1.242571e+02</td>
</tr>
<tr>
<td>84</td>
<td>014</td>
<td>3.804000e+00</td>
<td>1.271119e+02</td>
</tr>
<tr>
<td>85</td>
<td>015</td>
<td>3.619600e+00</td>
<td>1.299666e+02</td>
</tr>
<tr>
<td>86</td>
<td>016</td>
<td>3.591100e+00</td>
<td>1.328214e+02</td>
</tr>
<tr>
<td>87</td>
<td>017</td>
<td>3.562600e+00</td>
<td>1.356761e+02</td>
</tr>
<tr>
<td>88</td>
<td>018</td>
<td>3.534000e+00</td>
<td>1.385309e+02</td>
</tr>
<tr>
<td>89</td>
<td>019</td>
<td>3.505500e+00</td>
<td>1.413857e+02</td>
</tr>
<tr>
<td>90</td>
<td>020</td>
<td>3.477000e+00</td>
<td>1.442404e+02</td>
</tr>
<tr>
<td>91</td>
<td>021</td>
<td>3.448500e+00</td>
<td>1.470952e+02</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>---</td>
<td>---</td>
<td>---</td>
</tr>
<tr>
<td>93</td>
<td>022</td>
<td>3.419900e+00</td>
<td>1.499499e+02</td>
</tr>
<tr>
<td>94</td>
<td>023</td>
<td>3.391400e+00</td>
<td>1.528047e+02</td>
</tr>
<tr>
<td>95</td>
<td>024</td>
<td>3.362900e+00</td>
<td>1.556595e+02</td>
</tr>
<tr>
<td>96</td>
<td>025</td>
<td>3.334300e+00</td>
<td>1.585142e+02</td>
</tr>
<tr>
<td>97</td>
<td>026</td>
<td>3.305800e+00</td>
<td>1.613690e+02</td>
</tr>
<tr>
<td>98</td>
<td>027</td>
<td>3.277300e+00</td>
<td>1.642237e+02</td>
</tr>
<tr>
<td>99</td>
<td>028</td>
<td>3.248800e+00</td>
<td>1.670785e+02</td>
</tr>
<tr>
<td>100</td>
<td>029</td>
<td>3.220200e+00</td>
<td>1.699333e+02</td>
</tr>
<tr>
<td>101</td>
<td>030</td>
<td>3.191700e+00</td>
<td>1.727880e+02</td>
</tr>
<tr>
<td>102</td>
<td>031</td>
<td>3.163200e+00</td>
<td>1.756428e+02</td>
</tr>
<tr>
<td>103</td>
<td>032</td>
<td>3.134700e+00</td>
<td>1.784975e+02</td>
</tr>
<tr>
<td>104</td>
<td>033</td>
<td>3.106100e+00</td>
<td>1.813523e+02</td>
</tr>
<tr>
<td>105</td>
<td>034</td>
<td>3.077600e+00</td>
<td>1.842071e+02</td>
</tr>
<tr>
<td>106</td>
<td>035</td>
<td>3.049100e+00</td>
<td>1.870618e+02</td>
</tr>
<tr>
<td>107</td>
<td>036</td>
<td>3.020600e+00</td>
<td>1.899166e+02</td>
</tr>
<tr>
<td>108</td>
<td>037</td>
<td>2.992000e+00</td>
<td>1.927713e+02</td>
</tr>
<tr>
<td>109</td>
<td>038</td>
<td>2.963500e+00</td>
<td>1.956261e+02</td>
</tr>
<tr>
<td>110</td>
<td>039</td>
<td>2.935000e+00</td>
<td>1.984809e+02</td>
</tr>
<tr>
<td>111</td>
<td>040</td>
<td>2.906500e+00</td>
<td>2.013356e+02</td>
</tr>
<tr>
<td>112</td>
<td>041</td>
<td>2.877900e+00</td>
<td>2.041904e+02</td>
</tr>
<tr>
<td>113</td>
<td>042</td>
<td>2.849400e+00</td>
<td>2.070451e+02</td>
</tr>
<tr>
<td>114</td>
<td>043</td>
<td>2.820900e+00</td>
<td>2.098999e+02</td>
</tr>
<tr>
<td>115</td>
<td>044</td>
<td>2.792400e+00</td>
<td>2.127546e+02</td>
</tr>
<tr>
<td>116</td>
<td>045</td>
<td>2.763800e+00</td>
<td>2.156094e+02</td>
</tr>
<tr>
<td>117</td>
<td>046</td>
<td>2.735300e+00</td>
<td>2.184642e+02</td>
</tr>
<tr>
<td>118</td>
<td>047</td>
<td>2.434800e+00</td>
<td>2.213189e+02</td>
</tr>
<tr>
<td>119</td>
<td>048</td>
<td>7.745400e+00</td>
<td>2.241737e+02</td>
</tr>
<tr>
<td>120</td>
<td>049</td>
<td>1.339190e+01</td>
<td>2.270284e+02</td>
</tr>
<tr>
<td>121</td>
<td>050</td>
<td>1.800000e+01</td>
<td>2.313106e+02</td>
</tr>
</tbody>
</table>

param #3: pl298.  
  type = 2,  len = 1,  bra = 2  
  value is 1.945657e+01

param #4: DiskBoreWidth.  
  type = 2,  len = 1,  bra = 2  
  value is 3.044312e+01

param #5: p967.  
  type = 2,  len = 1,  bra = 0  
  value is 9.000000e+01

param #6: pl187.  
  type = 2,  len = 1,  bra = 2
value is 9.000000e+01
param #7: p969,  type = 2,  len = 1,  bra = 0
 value is 2.470000e+02
param #8: p1209,  type = 2,  len = 1,  bra = 3
 value is 3.600000e+02
param #9: RBOR,  type = 2,  len = 1,  bra = 2
 value is 9.000000e+01
param #10: PT1,  type = 2,  len = 1,  bra = 2
 value is 8.456421e-01
param #11: p1215__y,  type = 2,  len = 1,  bra = 0
 value is 0.000000e+00
param #12: p857,  type = 2,  len = 1,  bra = 0
 value is 1.700000e+00
param #13: p1300,  type = 2,  len = 1,  bra = 2
 value is 2.315981e+02
param #14: RTQB,  type = 2,  len = 1,  bra = 2
 value is 2.500000e-01
param #15: D1,  type = 2,  len = 1,  bra = 2
 value is 2.315981e+02
param #16: p970,  type = 2,  len = 1,  bra = 0
 value is 2.540000e+02
param #17: BIRR,  type = 2,  len = 1,  bra = 2
 value is 2.478593e+02
param #18: p1210,  type = 2,  len = 1,  bra = 3
 value is 0.000000e+00
param #19: WRIM,  type = 2,  len = 1,  bra = 2
 value is 3.600000e+01
param #20: p1214__x,  type = 2,  len = 1,  bra = 0
 value is 0.000000e+00
param #21: p1154,  type = 2,  len = 1,  bra = 2
 value is 1.522156e+01
param #22: p1216__z,  type = 2,  len = 1,  bra = 0
 value is 0.000000e+00
param #23: BladeRootHeight,  type = 2,  len = 1,  bra = 2
 value is 1.945657e+01
param #24: BERR,  type = 2,  len = 1,  bra = 2
value is 2.536750e+02

param #25: BSPN,  type = 2,  len = 1,  bra = 2
value is 7.782629e+01

param #26: p254,  type = 2,  len = 1,  bra = 0
value is 9.000000e+01

param #27: p255,  type = 2,  len = 1,  bra = 0
value is 2.313106e+02

CHANGES:

C Calling Fortran Subroutine ,
done

TYPE : 4

Type of disk is Continuos slope disk

The Changed values of new .din file : ,
The changed spline data 1 is as below:

001 -1.534851e+01  9.000001e+01
002 -1.498511e+01  9.185352e+01
003 -1.468704e+01  9.370701e+01
004 -1.441850e+01  9.556052e+01
005 -1.414866e+01  9.741402e+01
006 -1.385168e+01  9.926752e+01
007 -1.350671e+01  1.011210e+02
008 -1.309785e+01  1.029745e+02
009 -1.261421e+01  1.048280e+02
010 -1.204988e+01  1.066815e+02
011 -1.140392e+01  1.085350e+02
012 -1.068038e+01  1.103885e+02
013 -9.888277e+00  1.122420e+02
014 -9.041639e+00  1.140955e+02
015 -8.159446e+00  1.159490e+02
016 -7.265672e+00  1.178025e+02
017 -6.389278e+00  1.196560e+02
The changed spline data 2 is as below:

<p>| | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>018</td>
<td>-5.564188e+00</td>
<td>1.215095e+02</td>
<td></td>
</tr>
<tr>
<td></td>
<td>019</td>
<td>-4.829332e+00</td>
<td>1.233630e+02</td>
<td></td>
</tr>
<tr>
<td></td>
<td>020</td>
<td>-4.228607e+00</td>
<td>1.252165e+02</td>
<td></td>
</tr>
<tr>
<td></td>
<td>021</td>
<td>-3.810881e+00</td>
<td>1.270700e+02</td>
<td></td>
</tr>
<tr>
<td></td>
<td>022</td>
<td>-3.687289e+00</td>
<td>1.279968e+02</td>
<td></td>
</tr>
<tr>
<td></td>
<td>023</td>
<td>-3.630026e+00</td>
<td>1.289235e+02</td>
<td></td>
</tr>
<tr>
<td></td>
<td>024</td>
<td>-3.448524e+00</td>
<td>1.470878e+02</td>
<td></td>
</tr>
<tr>
<td></td>
<td>025</td>
<td>-3.267023e+00</td>
<td>1.652522e+02</td>
<td></td>
</tr>
<tr>
<td></td>
<td>026</td>
<td>-3.085522e+00</td>
<td>1.834165e+02</td>
<td></td>
</tr>
<tr>
<td></td>
<td>027</td>
<td>-2.904021e+00</td>
<td>2.015808e+02</td>
<td></td>
</tr>
<tr>
<td></td>
<td>028</td>
<td>-2.728023e+00</td>
<td>2.191943e+02</td>
<td></td>
</tr>
<tr>
<td></td>
<td>029</td>
<td>-2.722519e+00</td>
<td>2.197451e+02</td>
<td></td>
</tr>
<tr>
<td></td>
<td>030</td>
<td>-2.788676e+00</td>
<td>2.202958e+02</td>
<td></td>
</tr>
<tr>
<td></td>
<td>031</td>
<td>-3.058390e+00</td>
<td>2.208465e+02</td>
<td></td>
</tr>
<tr>
<td></td>
<td>032</td>
<td>-3.509082e+00</td>
<td>2.213973e+02</td>
<td></td>
</tr>
<tr>
<td></td>
<td>033</td>
<td>-4.118357e+00</td>
<td>2.219480e+02</td>
<td></td>
</tr>
<tr>
<td></td>
<td>034</td>
<td>-4.864102e+00</td>
<td>2.224988e+02</td>
<td></td>
</tr>
<tr>
<td></td>
<td>035</td>
<td>-5.724355e+00</td>
<td>2.230495e+02</td>
<td></td>
</tr>
<tr>
<td></td>
<td>036</td>
<td>-6.677464e+00</td>
<td>2.236003e+02</td>
<td></td>
</tr>
<tr>
<td></td>
<td>037</td>
<td>-7.701903e+00</td>
<td>2.241510e+02</td>
<td></td>
</tr>
<tr>
<td></td>
<td>038</td>
<td>-8.776478e+00</td>
<td>2.247017e+02</td>
<td></td>
</tr>
<tr>
<td></td>
<td>039</td>
<td>-9.880102e+00</td>
<td>2.252525e+02</td>
<td></td>
</tr>
<tr>
<td></td>
<td>040</td>
<td>-1.099203e+01</td>
<td>2.258032e+02</td>
<td></td>
</tr>
<tr>
<td></td>
<td>041</td>
<td>-1.209164e+01</td>
<td>2.263539e+02</td>
<td></td>
</tr>
<tr>
<td></td>
<td>042</td>
<td>-1.315862e+01</td>
<td>2.269047e+02</td>
<td></td>
</tr>
<tr>
<td></td>
<td>043</td>
<td>-1.417277e+01</td>
<td>2.274554e+02</td>
<td></td>
</tr>
<tr>
<td></td>
<td>044</td>
<td>-1.511428e+01</td>
<td>2.280062e+02</td>
<td></td>
</tr>
<tr>
<td></td>
<td>045</td>
<td>-1.596338e+01</td>
<td>2.285569e+02</td>
<td></td>
</tr>
<tr>
<td></td>
<td>046</td>
<td>-1.670066e+01</td>
<td>2.291077e+02</td>
<td></td>
</tr>
<tr>
<td></td>
<td>047</td>
<td>-1.730687e+01</td>
<td>2.296584e+02</td>
<td></td>
</tr>
<tr>
<td></td>
<td>048</td>
<td>-1.776301e+01</td>
<td>2.302091e+02</td>
<td></td>
</tr>
<tr>
<td></td>
<td>049</td>
<td>-1.805028e+01</td>
<td>2.307599e+02</td>
<td></td>
</tr>
<tr>
<td></td>
<td>050</td>
<td>-1.815013e+01</td>
<td>2.313106e+02</td>
<td></td>
</tr>
</tbody>
</table>

001 1.534851e+01 9.000001e+01
002 1.498511e+01 9.185352e+01
<p>| | | | | | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>237</td>
<td>003</td>
<td>1.468704e+01</td>
<td>9.370701e+01</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>238</td>
<td>004</td>
<td>1.441850e+01</td>
<td>9.556052e+01</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>239</td>
<td>005</td>
<td>1.414866e+01</td>
<td>9.741402e+01</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>240</td>
<td>006</td>
<td>1.385168e+01</td>
<td>9.926752e+01</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>241</td>
<td>007</td>
<td>1.350671e+01</td>
<td>1.011210e+02</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>242</td>
<td>008</td>
<td>1.309785e+01</td>
<td>1.029745e+02</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>243</td>
<td>009</td>
<td>1.261421e+01</td>
<td>1.048280e+02</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>244</td>
<td>010</td>
<td>1.204988e+01</td>
<td>1.066815e+02</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>245</td>
<td>011</td>
<td>1.140392e+01</td>
<td>1.085350e+02</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>246</td>
<td>012</td>
<td>1.068038e+01</td>
<td>1.103885e+02</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>247</td>
<td>013</td>
<td>9.888277e+00</td>
<td>1.122420e+02</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>248</td>
<td>014</td>
<td>9.041639e+00</td>
<td>1.140955e+02</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>249</td>
<td>015</td>
<td>8.159446e+00</td>
<td>1.159490e+02</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>250</td>
<td>016</td>
<td>7.265672e+00</td>
<td>1.178025e+02</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>251</td>
<td>017</td>
<td>6.389278e+00</td>
<td>1.196560e+02</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>252</td>
<td>018</td>
<td>5.564188e+00</td>
<td>1.215095e+02</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>253</td>
<td>019</td>
<td>4.829332e+00</td>
<td>1.233630e+02</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>254</td>
<td>020</td>
<td>4.228607e+00</td>
<td>1.252165e+02</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>255</td>
<td>021</td>
<td>3.810881e+00</td>
<td>1.270700e+02</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>256</td>
<td>022</td>
<td>3.687289e+00</td>
<td>1.279968e+02</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>257</td>
<td>023</td>
<td>3.630026e+00</td>
<td>1.289235e+02</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>258</td>
<td>024</td>
<td>3.448524e+00</td>
<td>1.470878e+02</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>259</td>
<td>025</td>
<td>3.267023e+00</td>
<td>1.652522e+02</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>260</td>
<td>026</td>
<td>3.085522e+00</td>
<td>1.834165e+02</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>261</td>
<td>027</td>
<td>2.904021e+00</td>
<td>2.015808e+02</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>262</td>
<td>028</td>
<td>2.728023e+00</td>
<td>2.191943e+02</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>263</td>
<td>029</td>
<td>2.722519e+00</td>
<td>2.197451e+02</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>264</td>
<td>030</td>
<td>2.788676e+00</td>
<td>2.202958e+02</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>265</td>
<td>031</td>
<td>3.058390e+00</td>
<td>2.208465e+02</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>266</td>
<td>032</td>
<td>3.509082e+00</td>
<td>2.213973e+02</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>267</td>
<td>033</td>
<td>4.118357e+00</td>
<td>2.219480e+02</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>268</td>
<td>034</td>
<td>4.864102e+00</td>
<td>2.224988e+02</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>269</td>
<td>035</td>
<td>5.724355e+00</td>
<td>2.230495e+02</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>270</td>
<td>036</td>
<td>6.677464e+00</td>
<td>2.236003e+02</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>271</td>
<td>037</td>
<td>7.701903e+00</td>
<td>2.241510e+02</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>272</td>
<td>038</td>
<td>8.776478e+00</td>
<td>2.247017e+02</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
The Ansys **File** is Saved as Ansys.ain.
Figure E.1: Disk Before Clustering at the curved section

Figure E.2: Disk After Clustering at the curved section

Figure E.3: Rotor 3 Disk of EEE
Figure E.4: Rotor 4 Disk of EEE after running through the code
Appendix F

ANSYS Output File

F.1 Fortran Program

```fortran
1 /prep7
2 MPTEMP,, -17.780001
3 MPTEMP,, 37.779999
4 MPTEMP,, 93.330002
5 MPTEMP,, 148.89000
6 MPTEMP,, 204.44000
7 MPTEMP,, 260.00000
8 MPTEMP,, 315.56000
9 MPTEMP,, 371.10999
10 MPTEMP,, 426.67001
11 MPTEMP,, 482.22000
12 MPTEMP,, 537.78003
13 MPTEMP,, 593.33002
14 MPTEMP,, 648.89001
15 MPTEMP,, 704.44000
16 MPTEMP,, 760.00000
17 MPTEMP,, 815.56000
18 MPTEMP,, 871.10999
19 MPTEMP,, 926.66998
20 MPTEMP,, 982.21997
21 MPTEMP,, 1037.7800
```
<table>
<thead>
<tr>
<th>Line</th>
<th>Description</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>94</td>
<td>MPDATA, EX, 1,</td>
<td>9.2300012E+10</td>
</tr>
<tr>
<td>95</td>
<td>MPDATA, EX, 2,</td>
<td>9.2300012E+10</td>
</tr>
<tr>
<td>96</td>
<td>MPDATA, ALPX, 1,</td>
<td>1.00799998E−05</td>
</tr>
<tr>
<td>97</td>
<td>MPDATA, ALPX, 2,</td>
<td>1.00799998E−05</td>
</tr>
<tr>
<td>98</td>
<td>MPDATA, PRXY, 1,</td>
<td>0.31000000</td>
</tr>
<tr>
<td>99</td>
<td>MPDATA, PRXY, 2,</td>
<td>0.31000000</td>
</tr>
<tr>
<td>100</td>
<td>MPDATA, DENS, 1,</td>
<td>4428.7798</td>
</tr>
<tr>
<td>101</td>
<td>MPDATA, DENS, 2,</td>
<td>0</td>
</tr>
<tr>
<td>102</td>
<td>MPDATA, EX, 1,</td>
<td>5.82599967E+10</td>
</tr>
<tr>
<td>103</td>
<td>MPDATA, EX, 2,</td>
<td>5.82599967E+10</td>
</tr>
<tr>
<td>104</td>
<td>MPDATA, ALPX, 1,</td>
<td>1.01699998E−05</td>
</tr>
<tr>
<td>105</td>
<td>MPDATA, ALPX, 2,</td>
<td>1.01699998E−05</td>
</tr>
<tr>
<td>106</td>
<td>MPDATA, PRXY, 1,</td>
<td>0.31000000</td>
</tr>
<tr>
<td>107</td>
<td>MPDATA, PRXY, 2,</td>
<td>0.31000000</td>
</tr>
<tr>
<td>108</td>
<td>MPDATA, DENS, 1,</td>
<td>4428.7798</td>
</tr>
<tr>
<td>109</td>
<td>MPDATA, DENS, 2,</td>
<td>0</td>
</tr>
<tr>
<td>110</td>
<td>MPDATA, EX, 1,</td>
<td>3.729000004E+10</td>
</tr>
<tr>
<td>111</td>
<td>MPDATA, EX, 2,</td>
<td>3.729000004E+10</td>
</tr>
<tr>
<td>112</td>
<td>MPDATA, ALPX, 1,</td>
<td>1.02600006E−05</td>
</tr>
<tr>
<td>113</td>
<td>MPDATA, ALPX, 2,</td>
<td>1.02600006E−05</td>
</tr>
<tr>
<td>114</td>
<td>MPDATA, PRXY, 1,</td>
<td>0.31000000</td>
</tr>
<tr>
<td>115</td>
<td>MPDATA, PRXY, 2,</td>
<td>0.31000000</td>
</tr>
<tr>
<td>116</td>
<td>MPDATA, DENS, 1,</td>
<td>4428.7798</td>
</tr>
<tr>
<td>117</td>
<td>MPDATA, DENS, 2,</td>
<td>0</td>
</tr>
<tr>
<td>118</td>
<td>MPDATA, EX, 1,</td>
<td>1.63099996E+10</td>
</tr>
<tr>
<td>119</td>
<td>MPDATA, EX, 2,</td>
<td>1.63099996E+10</td>
</tr>
<tr>
<td>120</td>
<td>MPDATA, ALPX, 1,</td>
<td>1.02600006E−05</td>
</tr>
<tr>
<td>121</td>
<td>MPDATA, ALPX, 2,</td>
<td>1.02600006E−05</td>
</tr>
<tr>
<td>122</td>
<td>MPDATA, PRXY, 1,</td>
<td>0.31000000</td>
</tr>
<tr>
<td>123</td>
<td>MPDATA, PRXY, 2,</td>
<td>0.31000000</td>
</tr>
<tr>
<td>124</td>
<td>MPDATA, DENS, 1,</td>
<td>4428.7798</td>
</tr>
<tr>
<td>125</td>
<td>MPDATA, DENS, 2,</td>
<td>0</td>
</tr>
<tr>
<td>126</td>
<td>MPDATA, EX, 1,</td>
<td>1.00000000E+09</td>
</tr>
<tr>
<td>127</td>
<td>MPDATA, EX, 2,</td>
<td>1.00000000E+09</td>
</tr>
<tr>
<td>128</td>
<td>MPDATA, ALPX, 1,</td>
<td>1.02600006E−05</td>
</tr>
<tr>
<td>129</td>
<td>MPDATA, ALPX, 2,</td>
<td>1.02600006E−05</td>
</tr>
</tbody>
</table>

127
130  MPDATA, PRXY, 1., 0.31000000
131  MPDATA, PRXY, 2., 0.31000000
132  MPDATA, DENS, 1., 4428.7798
133  MPDATA, DENS, 2., 0
134  MPDATA, EX, 1., 1.00000000E+09
135  MPDATA, EX, 2., 1.00000000E+09
136  MPDATA, ALPX, 1., 1.03500006E−05
137  MPDATA, ALPX, 2., 1.03500006E−05
138  MPDATA, PRXY, 1., 0.31000000
139  MPDATA, PRXY, 2., 0.31000000
140  MPDATA, DENS, 1., 4428.7798
141  MPDATA, DENS, 2., 0
142  MPDATA, EX, 1., 1.00000000E+09
143  MPDATA, EX, 2., 1.00000000E+09
144  MPDATA, ALPX, 1., 1.03500006E−05
145  MPDATA, ALPX, 2., 1.03500006E−05
146  MPDATA, PRXY, 1., 0.31000000
147  MPDATA, PRXY, 2., 0.31000000
148  MPDATA, DENS, 1., 4428.7798
149  MPDATA, DENS, 2., 0
150  MPDATA, EX, 1., 1.00000000E+09
151  MPDATA, EX, 2., 1.00000000E+09
152  MPDATA, ALPX, 1., 1.03500006E−05
153  MPDATA, ALPX, 2., 1.03500006E−05
154  MPDATA, PRXY, 1., 0.31000000
155  MPDATA, PRXY, 2., 0.31000000
156  MPDATA, DENS, 1., 4428.7798
157  MPDATA, DENS, 2., 0
158  MPDATA, EX, 1., 1.00000000E+09
159  MPDATA, EX, 2., 1.00000000E+09
160  MPDATA, ALPX, 1., 1.03500006E−05
161  MPDATA, ALPX, 2., 1.03500006E−05
162  MPDATA, PRXY, 1., 0.31000000
163  MPDATA, PRXY, 2., 0.31000000
164  MPDATA, DENS, 1., 4428.7798
165  MPDATA, DENS, 2., 0
set, r_cg_b, 0.25027418
set, total_dead_weight, 3.8996267
set, rimx, 0.23131061

CSYS, 0

ET, 1, MESH200, 6
ET, 2, MASS21
ET, 3, PLANE42, 1
ET, 4, PIPE16
ET, 5, SOLID185

R, 1, total_dead_weight, total_dead_weight, total_dead_weight
R, 2, 1, 0.05

k, 9.000000110E-02, 1.53485136E-02
k, 9.18535143E-02, 1.49851115E-02
k, 9.37070101E-02, 1.46870436E-02
k, 9.55605134E-02, 1.44185014E-02
k, 9.74140167E-02, 1.41486619E-02
k, 9.92675200E-02, 1.38516836E-02
k, 0.10112102, 1.35067059E-02
0.10297452, 1.30978478E-02
0.10482802, 1.26142092E-02
0.10668153, 1.20498780E-02
0.10853502, 1.14039201E-02
0.11038852, 1.06803766E-02
0.11224203, 9.88277397E-03
0.11409552, 9.04163904E-03
0.11594903, 8.15944560E-03
0.11780253, 7.26567209E-03
0.11965603, 6.38927799E-03
0.12150953, 5.63418858E-03
0.12336303, 4.82933223E-03
0.12521653, 4.22860682E-03
0.12707004, 3.81088071E-03
0.12799679, 3.68728861E-03
0.12892354, 3.63002578E-03
0.14708784, 3.44852451E-03
0.16525215, 3.26702325E-03
0.18341646, 3.08552198E-03
0.20158076, 2.90402072E-03
0.21919432, 2.72802263E-03
0.21974507, 2.72251945E-03
0.22029582, 2.78867572E-03
0.22084655, 3.05839999E-03
0.22139730, 3.50908190E-03
0.22194803, 4.11835732E-03
0.22249877, 4.86410223E-03
0.22304951, 5.72435511E-03
0.22360025, 6.67746412E-03
0.22415099, 7.70190312E-03
0.22470173, 8.77647754E-03
0.22525248, 9.88010224E-03
0.22580321, 1.09920343E-02
0.22635394, 1.20916367E-02
0.22690469, 1.31586157E-02
0.22745542, 1.41727747E-02
<table>
<thead>
<tr>
<th>k,</th>
<th>0.22800617</th>
<th>1.51142767E-02</th>
</tr>
</thead>
<tbody>
<tr>
<td>k,</td>
<td>0.22855690</td>
<td>1.59633756E-02</td>
</tr>
<tr>
<td>k,</td>
<td>0.22910765</td>
<td>1.67006627E-02</td>
</tr>
<tr>
<td>k,</td>
<td>0.22965838</td>
<td>1.73068680E-02</td>
</tr>
<tr>
<td>k,</td>
<td>0.23020913</td>
<td>1.77630093E-02</td>
</tr>
<tr>
<td>k,</td>
<td>0.23075986</td>
<td>1.80502776E-02</td>
</tr>
<tr>
<td>k,</td>
<td>0.23131061</td>
<td>1.81501247E-02</td>
</tr>
<tr>
<td>k,</td>
<td>0.2320913</td>
<td>1.77630093E-02</td>
</tr>
<tr>
<td>k,</td>
<td>0.2320913</td>
<td>1.80502776E-02</td>
</tr>
<tr>
<td>k,</td>
<td>0.22965838</td>
<td>1.73068680E-02</td>
</tr>
<tr>
<td>k,</td>
<td>0.22910765</td>
<td>1.67006627E-02</td>
</tr>
<tr>
<td>k,</td>
<td>0.22855690</td>
<td>1.59633756E-02</td>
</tr>
<tr>
<td>k,</td>
<td>0.22745542</td>
<td>1.41727747E-02</td>
</tr>
<tr>
<td>k,</td>
<td>0.22690469</td>
<td>1.31586157E-02</td>
</tr>
<tr>
<td>k,</td>
<td>0.22635394</td>
<td>1.20916367E-02</td>
</tr>
<tr>
<td>k,</td>
<td>0.22580321</td>
<td>1.09920343E-02</td>
</tr>
<tr>
<td>k,</td>
<td>0.22525248</td>
<td>9.88010224E-03</td>
</tr>
<tr>
<td>k,</td>
<td>0.22470173</td>
<td>8.77647754E-03</td>
</tr>
<tr>
<td>k,</td>
<td>0.22415099</td>
<td>7.70190312E-03</td>
</tr>
<tr>
<td>k,</td>
<td>0.22360025</td>
<td>6.67746412E-03</td>
</tr>
<tr>
<td>k,</td>
<td>0.22304951</td>
<td>5.72435511E-03</td>
</tr>
<tr>
<td>k,</td>
<td>0.22249877</td>
<td>4.86410223E-03</td>
</tr>
<tr>
<td>k,</td>
<td>0.22194803</td>
<td>4.11835732E-03</td>
</tr>
<tr>
<td>k,</td>
<td>0.22139730</td>
<td>3.50908190E-03</td>
</tr>
<tr>
<td>k,</td>
<td>0.22084655</td>
<td>3.05838999E-03</td>
</tr>
<tr>
<td>k,</td>
<td>0.22029582</td>
<td>2.78867572E-03</td>
</tr>
<tr>
<td>k,</td>
<td>0.21974507</td>
<td>2.72251945E-03</td>
</tr>
<tr>
<td>k,</td>
<td>0.21919432</td>
<td>2.72802263E-03</td>
</tr>
<tr>
<td>k,</td>
<td>0.21058076</td>
<td>2.90402072E-03</td>
</tr>
<tr>
<td>k,</td>
<td>0.18341646</td>
<td>3.08552198E-03</td>
</tr>
<tr>
<td>k,</td>
<td>0.16525215</td>
<td>3.26702325E-03</td>
</tr>
<tr>
<td>k,</td>
<td>0.14708784</td>
<td>3.44852451E-03</td>
</tr>
<tr>
<td>k,</td>
<td>0.12892354</td>
<td>3.63002578E-03</td>
</tr>
<tr>
<td>k,</td>
<td>0.12799679</td>
<td>3.68728861E-03</td>
</tr>
<tr>
<td>(k)</td>
<td>(0.12707004)</td>
<td>(-3.81088071\times10^{-3})</td>
</tr>
<tr>
<td>-------</td>
<td>----------------</td>
<td>-----------------------------</td>
</tr>
<tr>
<td>(k)</td>
<td>(0.12521653)</td>
<td>(-4.22860682\times10^{-3})</td>
</tr>
<tr>
<td>(k)</td>
<td>(0.12336303)</td>
<td>(-4.82933223\times10^{-3})</td>
</tr>
<tr>
<td>(k)</td>
<td>(0.12150953)</td>
<td>(-5.56418858\times10^{-3})</td>
</tr>
<tr>
<td>(k)</td>
<td>(0.11965603)</td>
<td>(-6.38927799\times10^{-3})</td>
</tr>
<tr>
<td>(k)</td>
<td>(0.11780253)</td>
<td>(-7.26567209\times10^{-3})</td>
</tr>
<tr>
<td>(k)</td>
<td>(0.11594903)</td>
<td>(-8.15944560\times10^{-3})</td>
</tr>
<tr>
<td>(k)</td>
<td>(0.11409552)</td>
<td>(-9.04163904\times10^{-3})</td>
</tr>
<tr>
<td>(k)</td>
<td>(0.11224203)</td>
<td>(-9.88827739\times10^{-3})</td>
</tr>
<tr>
<td>(k)</td>
<td>(0.11038852)</td>
<td>(-1.06803766\times10^{-2})</td>
</tr>
<tr>
<td>(k)</td>
<td>(0.10853502)</td>
<td>(-1.4039201\times10^{-2})</td>
</tr>
<tr>
<td>(k)</td>
<td>(0.10668153)</td>
<td>(-1.20498780\times10^{-2})</td>
</tr>
<tr>
<td>(k)</td>
<td>(0.10482802)</td>
<td>(-1.26142092\times10^{-2})</td>
</tr>
<tr>
<td>(k)</td>
<td>(0.10297452)</td>
<td>(-1.30978478\times10^{-2})</td>
</tr>
<tr>
<td>(k)</td>
<td>(0.10112102)</td>
<td>(-1.35067059\times10^{-2})</td>
</tr>
<tr>
<td>(k)</td>
<td>(9.92675200\times10^{-2})</td>
<td>(-1.38516836\times10^{-2})</td>
</tr>
<tr>
<td>(k)</td>
<td>(9.74140167\times10^{-2})</td>
<td>(-1.41486619\times10^{-2})</td>
</tr>
<tr>
<td>(k)</td>
<td>(9.55605134\times10^{-2})</td>
<td>(-1.44185014\times10^{-2})</td>
</tr>
<tr>
<td>(k)</td>
<td>(9.37070101\times10^{-2})</td>
<td>(-1.46870436\times10^{-2})</td>
</tr>
<tr>
<td>(k)</td>
<td>(9.18535143\times10^{-2})</td>
<td>(-1.49851115\times10^{-2})</td>
</tr>
<tr>
<td>(k)</td>
<td>(9.00000110\times10^{-2})</td>
<td>(-1.53485136\times10^{-2})</td>
</tr>
</tbody>
</table>
l, 88, 89
l, 89, 90
l, 90, 91
l, 91, 92
l, 92, 93
l, 93, 94
l, 94, 95
l, 95, 96
l, 96, 97
l, 97, 98
l, 98, 99
l, 99, 100
l, 1, 100
lsel, all
lesize, all, 1
lsel, none
lsel, s, line,, 50
lsel, a, line,, 100
/rep
lesize, all, 2.82621197E−03,...,1,...,0.
allsel
al, all
aesize, all, 2.82621197E−03
type, 3
mat, 1
real, 1
amesh, all
arefine, all,...,1
type, 5
k,, 0, 1.81501247E−02
k,, 0, −1.81501247E−02
vrotate, all,...,101,102,360,10
vsweep, all, 1
omega,, 1288.3672
418  *SET,Taxis,  145.77992
419  *SET,rimT,   145.77992
420  *SET,TAdder, 180.00000
421  *CREATE,PLOTDISK2,mac
422  /EXPAND,0
423  /rep
424  *END
425  /NOPR
426  *ABB,SAVE_DB ,SAVE
427  *ABB,RESUM_DB,RESUME
428  *ABB,QUIT ,Fnc_/EXIT
429  *ABB,POWRGRPH,Fnc_/GRAPHICS
430  *ABB,REFRESH ,/REP
431  *ABB,2D=PLOT ,PLOTDISK2
432  /GO
Appendix G

INIT file

G.1 TTDES Input

1 1 ! Units [1-SI:2-English]
2 2 1 1 1 ! Design Options [Unknown|Known: 1:(alpha2←
alpha3,M2,TT3), 2:(alpha3|alpha2,M2,TT3), 3:(M2|TT3,alpha2,alpha3), ←
4:(TT3|M2,alpha2,alpha3)]
3 1 ! N Stages
4 39933.775 ! Mass Flow Rate [kg/s]
5 12 ! RPM [rpm]
6 101325 ! Inlet Total Pressure [Pa]
7 288.15 ! Inlet Total Temperature [K]
8 0. ! Alpha 1 - First Stage [deg]
9 0.02059 ! Mach 1 - First Stage [-]
10 0 ! Inlet Duct Length/N1 Axial Width Ratio [-]
11 0 0 ! Hub and Casing slope upstream of N1 [deg]
12 1.401 ! Ratio of Specific Heats [-]
13 0.287 ! Gas Constant [kJ/kg*K]
14 0.0009 ! Ratio of clearance/(tip radius) or clearance/(hub radius←
) for all rotors and stator (not IGV or last stator)
# Appendix H

## STAGE file

### H.1 TTDES Input

<table>
<thead>
<tr>
<th>Stage Data</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>! Alpha 2</td>
<td>[deg]</td>
</tr>
<tr>
<td>! Mach 2</td>
<td>[-]</td>
</tr>
<tr>
<td>! Total Temperature 3</td>
<td>[K]</td>
</tr>
<tr>
<td>! Stator Zweifel Number</td>
<td>[-]</td>
</tr>
<tr>
<td>! Rotor Zweifel Number</td>
<td>[-]</td>
</tr>
<tr>
<td>! Stator Phi Coef.</td>
<td>[-]</td>
</tr>
<tr>
<td>! Rotor Phi Coef.</td>
<td>[-]</td>
</tr>
<tr>
<td>! Stator Aspect Ratio</td>
<td>[-]</td>
</tr>
<tr>
<td>! Rotor Aspect Ratio</td>
<td>[-]</td>
</tr>
<tr>
<td>! Rotor Axial Velocity Ratio</td>
<td>[-]</td>
</tr>
<tr>
<td>! Stator Row Space Coef.</td>
<td>[-]</td>
</tr>
<tr>
<td>! Rotor Row Space Coef.</td>
<td>[-]</td>
</tr>
<tr>
<td>! Stator Mean Radius</td>
<td>[m]</td>
</tr>
<tr>
<td>! Rotor Mean Radius</td>
<td>[m]</td>
</tr>
</tbody>
</table>
Appendix I

TTDES-Results file

I.1 TTDES - Results

1 T-T DES output for case: wt
2 The date for this run is (mm/dd/year): 12/20/2010
3 The time for this run is 16:37 and 55 seconds
4
5 ————STAGE 1———
6
7 Areas (m^2): 1: 4651.7 2: 4651.8 3: 4671.2
8 Heights (m): 1: 37.184 2: 37.185 3: 37.340
9 Stator Chord: 371.85 m, Number of stator blades: 1
10 Rotor Chord: 1.3127 m, Number of rotor blades: 4
11 Stage Pressure Ratio: 1.0042
12 Stage Temperature Ratio: 1.0000
13 Stage Adiabatic Efficiency: 0.28673E-01
14 Degree of Reaction: hub: 1.0000 pitch: 1.0000 tip: 1.0000
15 Stage Load Coefficient: 0.79925E-02
16 Stage Flow Coefficient: 0.28011
17
18 1h 1m 1t 2h 2m 2t 2Rm 3←
19
20 Rm 3h 3m 3t
21
139
<table>
<thead>
<tr>
<th>No.</th>
<th>Symbol</th>
<th>Unit</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>20</td>
<td>TT</td>
<td>Kcl</td>
<td>288.1 288.1 288.1 288.1 288.1 288.1 288.1 288.1 288.5 288.1 288.1 288.1 288.1 288.1 288.1</td>
</tr>
<tr>
<td>21</td>
<td>Ts</td>
<td>Kcl</td>
<td>288.1 288.1 288.1 288.1 288.1 288.1 288.1 288.1 288.1</td>
</tr>
<tr>
<td>22</td>
<td>PT</td>
<td>kPa</td>
<td>101.33 101.33 101.33 101.32 101.32 101.32 101.71 101.29 100.90 100.90 100.90</td>
</tr>
<tr>
<td>23</td>
<td>Ps</td>
<td>kPa</td>
<td>101.29 101.29 101.29 101.29 101.29 101.29 101.29 101.29 101.29 100.87 100.84 100.87 100.87</td>
</tr>
<tr>
<td>24</td>
<td>Mach</td>
<td></td>
<td>0.021 0.021 0.021 0.021 0.021 0.021 0.021 0.076 0.028 0.021 0.021</td>
</tr>
<tr>
<td>25</td>
<td>Vel</td>
<td>m/s</td>
<td>7.0 7.0 7.0 7.0 7.0 7.0 7.0 26.0 9.5 7.0 7.0 7.0</td>
</tr>
<tr>
<td>26</td>
<td>Ax V</td>
<td>m/s</td>
<td>7.0 7.0 7.0 7.0 7.0 7.0 7.0 7.0 7.0 7.0 7.0 7.0 7.0</td>
</tr>
<tr>
<td>27</td>
<td>TangV</td>
<td>m/s</td>
<td>0.0 0.0 0.0 0.0 0.0 0.0 0.0 0.0 25.4 6.4 0.4 0.2</td>
</tr>
<tr>
<td>28</td>
<td>alpha/beta</td>
<td></td>
<td>0.00 0.00 0.00 0.00 0.00 0.00 0.00 0.00 74.59 42.50 3.27 1.69</td>
</tr>
</tbody>
</table>

--- AXIAL FLOW TURBINE ---

- Turbine Pressure Ratio: 1.0042
- Turbine Temperature Ratio: 1.0000
- Turbine Adiabatic Efficiency: 0.28673E−01
Appendix J

TAXI Input

J.1 stack

1. T-T DES output, extension: wt
2. The overall information for each blade row is
3. Number of blades rotation_speed Hub_cl Tip_cl
4. Geometry, loss, blk, rV_theta are input starting at the hub
5. r_le r_te x_le x_te Loss Blk rV_theta
6. 1.401000  518.669983  1  0.064592  0.8973709E+09  0  1
7. Above are gam, TT_in(deg R), type(0-conv, 1-turb), ma, Re, BL switch & ← loss flag
8.  3 -0.142144  0.000000  0.000900
9.  0.02800  0.02800  9.65819  9.68679  0.0  0.000  0.000000
10. 0.08227  0.08227  9.65885  9.68613  0.0  0.000  0.000000
11. 0.13656  0.13656  9.65951  9.68547  0.0  0.000  0.000000
12. 0.19083  0.19083  9.66018  9.68481  0.0  0.000  0.000000
13. 0.24511  0.24511  9.66084  9.69920  0.0  0.000  0.00045
14. 0.29939  0.29939  9.66150  9.69249  0.0  0.000  0.00066
15. 0.35367  0.35367  9.66216  9.69112  0.0  0.000  0.00066
16. 0.40794  0.40794  9.66282  9.68476  0.0  0.000  0.00066
17. 0.46222  0.46222  9.66348  9.68214  0.0  0.000  0.00066
18. 0.51650  0.51650  9.66414  9.68009  0.0  0.000  0.00066
19. 0.57078  0.57078  9.66481  9.67748  0.0  0.000  0.00066
20. 0.62506  0.62506  9.66547  9.67521  0.0  0.000  0.00066
<table>
<thead>
<tr>
<th></th>
<th>0.67933</th>
<th>0.67933</th>
<th>9.66613</th>
<th>9.67521</th>
<th>0.0</th>
<th>0.000</th>
<th>0.00066</th>
</tr>
</thead>
<tbody>
<tr>
<td>21</td>
<td>0.73361</td>
<td>0.73361</td>
<td>9.66679</td>
<td>9.67343</td>
<td>0.0</td>
<td>0.000</td>
<td>0.00066</td>
</tr>
<tr>
<td>22</td>
<td>0.78789</td>
<td>0.78789</td>
<td>9.66745</td>
<td>9.67283</td>
<td>0.0</td>
<td>0.000</td>
<td>0.00066</td>
</tr>
<tr>
<td>23</td>
<td>0.84217</td>
<td>0.84217</td>
<td>9.66811</td>
<td>9.67239</td>
<td>0.0</td>
<td>0.000</td>
<td>0.00050</td>
</tr>
<tr>
<td>24</td>
<td>0.89644</td>
<td>0.89644</td>
<td>9.66878</td>
<td>9.67206</td>
<td>0.0</td>
<td>0.000</td>
<td>0.00040</td>
</tr>
<tr>
<td>25</td>
<td>0.95072</td>
<td>0.95072</td>
<td>9.66944</td>
<td>9.67184</td>
<td>0.0</td>
<td>0.000</td>
<td>0.00030</td>
</tr>
<tr>
<td>26</td>
<td>1.00500</td>
<td>1.00500</td>
<td>9.67010</td>
<td>9.67071</td>
<td>0.0</td>
<td>0.000</td>
<td>0.00000</td>
</tr>
</tbody>
</table>
## J.2 Walls

A table of T-TDES output, extension: wt

<p>| | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>T-TDES output, extension: wt</td>
<td>9.607866</td>
<td>9.930833</td>
<td>0.000000</td>
</tr>
<tr>
<td>2</td>
<td>10.080833</td>
<td>0.033733</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>9.980833</td>
<td>0.033733</td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>9.880834</td>
<td>0.033733</td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>9.830833</td>
<td>0.033733</td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>9.780833</td>
<td>0.033733</td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>9.658350</td>
<td>0.033733</td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>9.657866</td>
<td>0.033733</td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>9.607866</td>
<td>0.034213</td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>999.000000</td>
<td>999.000000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>9.607866</td>
<td>1.005000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>9.657866</td>
<td>1.005000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>9.709368</td>
<td>1.005000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>9.729815</td>
<td>1.005000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>9.779816</td>
<td>1.005000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>9.829816</td>
<td>1.005000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>17</td>
<td>9.929815</td>
<td>1.005000</td>
<td></td>
<td></td>
</tr>
<tr>
<td>18</td>
<td>10.029816</td>
<td>1.005000</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Appendix K

geomturbo_creator Code

K.1 Fortran Program

program geomturbo
!
! program to create the geomturbo file for Wind turbine....Soumitr Dey ←
03/22/2011
!
! inputs: section files, walls file
! output: geomturbo file
implicit none

character*32 fname,fname1,fname2,temp,case
character*32 fname3
integer iap,nspn,nrow,clmn,nblade,np,k,nx,nax,ia,i
integer row_n

parameter(nrow=1,row = 600,clmn = 20)
parameter(nx=500,nax=50)

real bsf
real xb(nx,nax),yb(nx,nax),zb(nx,nax)
real xbl(nx,nax),ybl(nx,nax),zbl(nx,nax)
real xs(nx,nax),ys(nx,nax),zs(nx,nax)! suction side coordinates
real xp(nx,nax),yp(nx,nax),zp(nx,nax)! pressure side coordinates
real zhub(row),rhub(row),tempr1(2)
real ztip(row), rtip(row)
!------- scaled coordinates
real xs_scaled(nx, nax), ys_scaled(nx, nax), zs_scaled(nx, nax)! suction side ←
coordinates
real xp_scaled(nx, nax), yp_scaled(nx, nax), zp_scaled(nx, nax)! pressure side ←
coordinates
real zhub_scaled(row), rhub_scaled(row)
real ztip_scaled(row), rtip_scaled(row)

28 call getarg(1, case)
29
30 !
31 call aski(' Enter the blade row number( 1 or 2 )^', row_n)
32 write(*, *)
33 call aski(' Enter number of blades for this row^', nblade)
34 write(*, *)
35 call aski(' Enter number of sections for each blade^', nspn)
36 write(*, *)
37 !call aski(' Enter number of points in each section^', iap)
38 !write(*, *)
39 call askr(' Enter the blade scaling factor(in meters)^', bsf)
40 write(*, *)
41 write(*, *)'Writing geomturbo file..' 
42 write(*, *)
43 write(temp, *)row_n
44 fname='row./trim(adjstl(temp))/'.geomTurbo'
45 open(1, file=fname, status='unknown')
46 write(1, '*')'GEOMETRY TURBO'
47 write(1, '*')'VERSION ', 5.3
48 write(1, '*')'TOLERANCE', 1e-006
49 write(1, '*')'NI_BEGIN CHANNEL'
50 write(1, '*')' NI_BEGIN basic_curve'
51 write(1, '*')' NAME hub'
52 write(1, '*')' DISCRETISATION', 10
53 write(1, '*')' DATA_REDUCTION', 0
54 write(1, '*')' NI_BEGIN zrcurve'
write(1,*)' ZR'

if (case == 'nrel') then
bsf = 0.500
else if (case == 'wt') then
bsf = 0.5
else if (case == 'booster') then
bsf = 0.5334
else
bsf = 0.5
endif

reading from the walls file——-
fname2=(trim(case)//'twal')
open(3, file=fname2, status='old')
read(3,*)temp
read(3,*)temp
k=0
do while (.true.)
read(3,*)tempr1(1),tempr1(2)
if (tempr1(1).ne.999.000) then
k=k+1
else
exit
endif
enddo
close(3)

open(3, file=fname2, status='old')
read(3,*)temp
read(3,*)temp
do i = 1, k
read(3,*)zhub(i),rhub(i)
zhub_scaled(i)=bsf*zhub(i)
rhub_scaled(i)=bsf*rhub(i)
endo
do i = 1, k
    read(3,*)ztip(i),rtip(i)
    ztip_scaled(i)=bsf*ztip(i)
    rtip_scaled(i)=bsf*rtip(i)
endo
close(3)
write(1,*), k ! no. of hub and tip coordinates
do i=k,1,-1
    write(1,*),zhub_scaled(i),rhub_scaled(i)
endo!
write(1,*)' NI_END zrcurve'
write(1,*)' NI_END basic_curve'
write(1,*)' NI_BEGIN basic_curve'
write(1,*)' NAME shroud'
write(1,*)' DISCRETISATION',10
write(1,*)' DATA_REDUCTION',0
write(1,*)' NI_BEGIN zrcurve'
write(1,*)' ZR'
do i=1,k
    write(1,*),ztip_scaled(i),rtip_scaled(i)
endo'
write(1,*)' NI_END zrcurve'
write(1,*)' NI_END basic_curve'
write(1,*)' NI_BEGIN channel_curve hub'
write(1,*)' NAME hub'
write(1,*)' VERTEX CURVE_P hub',0
write(1,*)' VERTEX CURVE_P hub',1
write(1,*)' NI_END channel_curve hub'
write(1,*)' NI_BEGIN channel_curve shroud'
write(1,*)' NAME shroud'
write(1,*)' VERTEX CURVE_P shroud',0
write(1,*)' VERTEX CURVE_P shroud',1
write(1,*)' NI_END channel_curve shroud'
write(1,*)' NI_BEGIN channel_curve nozzle'
write(1,*)' NAME nozzle'
write(1,*)' NI_END channel_curve nozzle'
write(1,*)'NI_END CHANNEL'
write(1,*)'NI_BEGIN nirow'
write(1,*)' NAME row ',row_n
write(1,*)' TYPE normal'
write(1,*)' PERIODICITY',nblade
write(1,*)'NI_BEGIN NNOnAxiSurfaces hub'
write(1,*)' NAME non axisymmetric hub'
write(1,*)' REPETITION',0
write(1,*)'NI_END NIOnAxiSurfaces hub'
write(1,*)'NI_BEGIN NNOnAxiSurfaces shroud'
write(1,*)' NAME non axisymmetric shroud'
write(1,*)' REPETITION',0
write(1,*)'NI_END NNOnAxiSurfaces shroud'
write(1,*)'NI_BEGIN NIBlade'
write(1,*)' NAME Main Blade'
write(1,*)'NI_BEGIN nibladegeometry'
write(1,*)' TYPE GEOMTURBO'
write(1,*)' GEOMETRY_MODIFIED',0
write(1,*)'GEOMETRY TURBO VERSION 5'
write(1,*)'blade_expansion_factor_hub',0.01! This is just a sample value
write(1,*)'blade_expansion_factor_shroud',0.01
write(1,*)'intersection_npts',10
write(1,*)'intersection_control',1
write(1,*)'data_reduction',0
write(1,*)'data_reduction_spacing_tolerance',0.0001
write(1,*)'data_Reduction_angle_tolerance',90
write(1,*)'control_points_distribution 0 -1 -1 -1 -1 -1 -1'
write(1,*)'units',1
write(1,*)'number_of_blades',nblade
write(1,*)'suction'
write(1,*)'SECTIONAL'
write(1,*)nspn
!
! reading the section files....
do ia = 1, nspn
write(temp,*),ia
fname3='sec'/trim(adjustl(temp))/'.dat'
open(3,file=fname3,status='unknown')
do ia=1,199
read(3,*),xb1(i,ia),yb1(i,ia),zb1(i,ia)
xb(i,ia) = xb1(i,ia)/1000
yb(i,ia) = yb1(i,ia)/1000
zb(i,ia) = zb1(i,ia)/1000
enddo
!
! Assigning the points as suction and pressure side points...
do ia=1,nspn
do i=100,1,-1
xp(i,ia)=xb(i,ia)
yp(i,ia)=yb(i,ia)
zp(i,ia)=zb(i,ia)
enddo
do i=100,199
xs(i,ia)=xb(i,ia)
ys(i,ia)=yb(i,ia)
zs(i,ia)=zb(i,ia)
enddo
!do ia=1,nspn
write(1,*),'
section',ia
write(1,*),'XYZ'
write(1,*),100
do i=100,199
! xs_scaled(i,ia)=bsf*xs(i,ia)
! ys_scaled(i,ia)=bsf*ys(i,ia)
! zs_scaled(i,ia)=bsf*zs(i,ia)
write(1,*),zs(i,ia),ys(i,ia),xs(i,ia)
enddo
!close(3)
enddo
write(l,*),'pressure'
write(l,*),'SECTIONAL'
do ia=1,nspn
do i=100,1,-1
 xp_scaled(i,ia)=bsf*xp(i,ia)
 yp_scaled(i,ia)=bsf*yp(i,ia)
 zp_scaled(i,ia)=bsf*zp(i,ia)
 enddo
enddo
write(l,*),'Ni_END nibladegeometry'
write(l,*),' SOLID_BODY_CONFIGURATION',0
write(l,*),'Ni_END NIBlade'
write(l,*),'Ni_END nirow'
write(l,*),'Ni_END GEOMTURBO'
write(l,*)
write(l,*),'Ni_BEGIN GEOMETRY'
close(l)
end program geomturbo

!---------------------------------------------------------------
SUBROUTINE ASKI(PROMPT,IINPUT)
!
!----- integer input
!
 CHARACTER*(*) PROMPT
 integer IINPUT
!
 NP = INDEX(PROMPT,'^') - 1
 IF(NP.EQ.0) NP = LEN(PROMPT)
235  !
236  10  WRITE(*,1000) PROMPT(1:NP)
237       READ (*,*,ERR=10) IINPUT
238       RETURN
239  !
240  1000  FORMAT(/A, i > ',5)
241       END ! ASKI
242 !-----------------------------------------------------------------------
243  subroutine askr(prompt,rinput)
244  !
245  !---- real input
246  !
247    character(*) prompt
248    real rinput
249  !
250    np = index(prompt,'^') - 1
251    IF(np.eq.0) np = len(prompt)
252  !
253  10  write(*,1000) prompt(1:np)
254    read (*,*,err=10) rinput
255    RETURN
256  !
257  1000  format(/a, r > ',5)
258    END ! askr
## Appendix L

### MISES Input files

#### L.1 Blade File

<p>| | | | | | | | | | | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>10.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>0.08397</td>
<td>0.04300</td>
<td>0.25</td>
<td>0.5</td>
<td>9.0</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>0.90545</td>
<td>0.01694</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>0.88537</td>
<td>0.02335</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>0.84854</td>
<td>0.03017</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>6</td>
<td>0.80756</td>
<td>0.03761</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>0.76314</td>
<td>0.04578</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8</td>
<td>0.71606</td>
<td>0.05462</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9</td>
<td>0.66718</td>
<td>0.06403</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>10</td>
<td>0.61747</td>
<td>0.07379</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>11</td>
<td>0.56801</td>
<td>0.08356</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>0.52005</td>
<td>0.09237</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>13</td>
<td>0.47384</td>
<td>0.09843</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>14</td>
<td>0.42809</td>
<td>0.10101</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>0.38223</td>
<td>0.10109</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>16</td>
<td>0.33689</td>
<td>0.09933</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>17</td>
<td>0.29259</td>
<td>0.09594</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>18</td>
<td>0.24987</td>
<td>0.09113</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>19</td>
<td>0.20920</td>
<td>0.08505</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>0.17103</td>
<td>0.07786</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>21</td>
<td>0.13580</td>
<td>0.06972</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>-------</td>
<td>-------</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>22</td>
<td>0.10390</td>
<td>0.06082</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>23</td>
<td>0.07568</td>
<td>0.05132</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>24</td>
<td>0.05147</td>
<td>0.04143</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>25</td>
<td>0.03158</td>
<td>0.03136</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>26</td>
<td>0.01626</td>
<td>0.02133</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>27</td>
<td>0.00575</td>
<td>0.01166</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>28</td>
<td>0.00037</td>
<td>0.00275</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>29</td>
<td>0.00140</td>
<td>−0.00498</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>30</td>
<td>0.00933</td>
<td>−0.01272</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>31</td>
<td>0.02321</td>
<td>−0.02162</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>32</td>
<td>0.04223</td>
<td>−0.03144</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>33</td>
<td>0.06579</td>
<td>−0.04199</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>34</td>
<td>0.09325</td>
<td>−0.05301</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>35</td>
<td>0.12397</td>
<td>−0.06408</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>36</td>
<td>0.15752</td>
<td>−0.07467</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>37</td>
<td>0.19362</td>
<td>−0.08447</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>38</td>
<td>0.23175</td>
<td>−0.09326</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>39</td>
<td>0.27129</td>
<td>−0.10060</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>40</td>
<td>0.31188</td>
<td>−0.10589</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>41</td>
<td>0.35328</td>
<td>−0.10866</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>42</td>
<td>0.39541</td>
<td>−0.10842</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>43</td>
<td>0.43832</td>
<td>−0.10484</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>44</td>
<td>0.48234</td>
<td>−0.09756</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>45</td>
<td>0.52837</td>
<td>−0.08697</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>46</td>
<td>0.57663</td>
<td>−0.07442</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>47</td>
<td>0.62649</td>
<td>−0.06112</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>48</td>
<td>0.67710</td>
<td>−0.04792</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>49</td>
<td>0.72752</td>
<td>−0.03558</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>50</td>
<td>0.77668</td>
<td>−0.02466</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>51</td>
<td>0.82348</td>
<td>−0.01559</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>52</td>
<td>0.86677</td>
<td>−0.00859</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>53</td>
<td>0.90545</td>
<td>−0.00370</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

153
L.2 Ises File

1 1 2 5 15 6
2 1 3 4 15 6
3 0.05494 0.99765 0.08397 0.00037 0.05473 | M1 S1 P1/Po X1 Vt1/A0
4 0.06217 0.99715 0.15338 1.00088 0.06073 | M2 S2 P2/Po X2 Vt2/A0
5 0.00000 0.00000 | MFRIN HWRAT
6 1.42e6 -3.000 | REYN= 6.7579e6 NCRIT
7 0.0000 0.0000 | XTR1 XTR2 ...
8 4 0.950 1.000 | ISMOM MCRIT MUCON
Index

HAWTs, 6

NREL, 5

T-Axi, 15

VAWTs, 6