The Pennsylvania State University

The Graduate School

Department of Aerospace Engineering

COMPUTATIONAL FREE WAKE ANALYSIS OF A HELICOPTER ROTOR

A Thesis in

Aerospace Engineering

by

Christopher J. Szymendera

Submitted in Partial Fulfillment
of the Requirements
for the Degree of

Master of Science

May 2002

We approve the thesis of Christopher J. Szymendera.

Date of Signature

Lyle N. Long
Professor of Aerospace Engineering
Thesis Advisor

Mark D. Maughmer
Professor of Aerospace Engineering

Dennis K. McLaughlin
Professor of Aerospace Engineering
Head of the Department of Aerospace Engineering

ABSTRACT

One of the most important issues in understanding the behavior of rotorcraft is the accurate prediction of the rotor wake. Understanding the complex nature of the wake is necessary for the prediction of such factors as blade loading, acoustics, and vibration. A free wake vortex lattice method was used to predict the wake structure and blade loading for a rotor in arbitrary motion. The blades were modeled as flat plates, and ring vortices were distributed on the surface of the blades. As the blades rotated, vortices were shed into the wake and then moved with a local velocity induced by the effects of the vortices on the blades and in the wake. The wake was allowed to freely deform over time into its natural structure. The lift was determined from the strength of each vortex on the blade, calculated by applying the boundary condition of no flow normal to the blade.

On a helicopter in forward flight, the advancing blades usually experience high Mach numbers at the tip. Therefore, it is important to be able to accurately predict the blade loading in compressible flow. The free wake method, though, calculates induced velocities using the Biot-Savart Law, which is only applicable to incompressible flow. A Prandtl-Glauert correction was applied then to the Biot-Savart Law, which allowed the code to accurately model compressible flow. The code was validated using two experimental hover cases, one in which the flow was entirely incompressible and one in which the flow was compressible at the tips, and then extended to handle any arbitrary motion of the helicopter.

TABLE OF CONTENTS

LIST OF FIGURES

ACKNOWLEDGMENTS

I would like to thank the National Science Foundation for funding the CEMBA program, an IGERT research center. And I would like to thank the students and faculty of the CEMBA program for providing me with the opportunity and the resources to work on this project.

I would also like to acknowledge the students of the IHPCA for their suggestions and technical assistance throughout my work. Many thanks also to Dr. Mark Maughmer for his comments and suggestions regarding my work. Finally, I would like to express my appreciation to Dr. Lyle Long for his guidance throughout this project. His support in this work was invaluable.

Chapter 1

INTRODUCTION

1.1 Overview

Accurate prediction of the rotor wake is one of the biggest challenges facing the rotorcraft industry today. Understanding the detailed structure of the wake is fundamental to the accurate prediction of rotor loads, performance, vibrations, and acoustics. Away from the rotor, understanding the wake is important in the prediction of interference effects with the fuselage, empennage, and tail rotor. The position and strength of the wake are influenced by many factors, including blade geometry, number of blades, rotor thrust, angle of attack of the tip path plane, and operating state of the helicopter. In most forward flight operating conditions, the combination of the forward speed of the helicopter and the rotating speed of the blades results in areas of high-Mach number flow at the blade tips on the advancing side of the rotor disk. Therefore, it is important to consider compressibility effects when attempting to model the flow of a helicopter in any operating state other than the simple hover case and slow-speed forward flight.

The complexity of the wake of a helicopter in forward flight can be seen in *Figure 1.1*. It shows some of the aerodynamic issues that exist in the wake as well as the interaction of the wake with the body of the helicopter. Because of these interaction

effects, the ability to predict the performance of the entire helicopter is highly dependent on the ability to predict the detailed structure of the wake. The details of the wake are hard to visualize by either experimental or computational models, however, due to the complex nature of the wake. Therefore, it is necessary to develop more advanced computational models that will allow engineers to accurately predict the position and strength of the rotor wake and, in turn, accurately predict the performance and aeroacoustics of the entire helicopter.

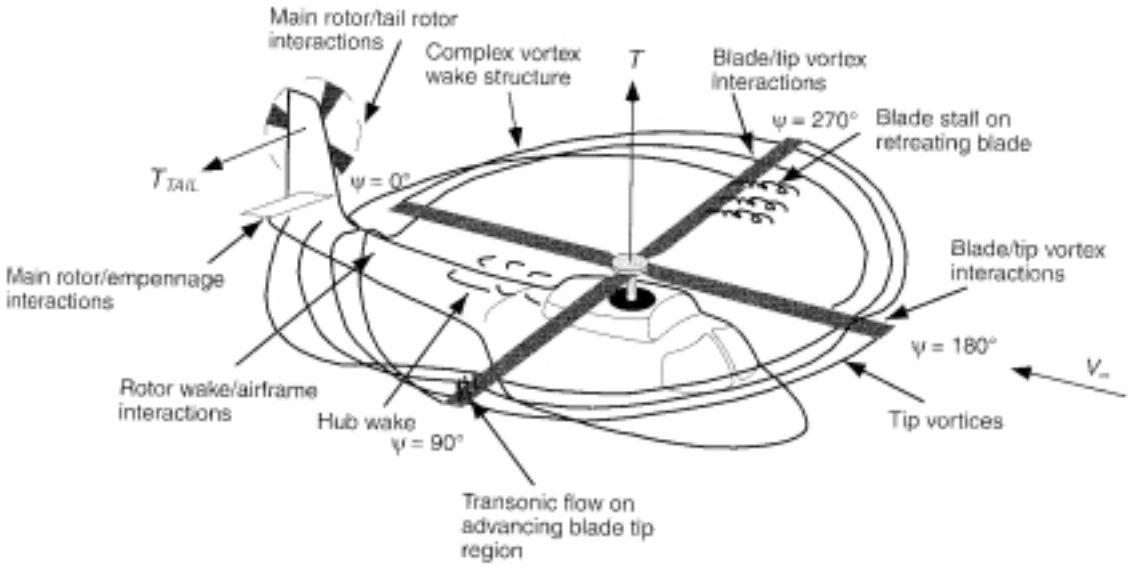


Figure 1.1: Schematic showing the wake and its interaction with the fuselage in forward flight [22]

Because the rotor wake is affected by many different factors, it is important to develop a method that is extremely versatile in order for it to be effective. The method should be able to handle any number of rotating blades with any given pitch and coning angle and should be able to model the wake of a helicopter following any specific flight path. The simplest approach, which will be presented in detail here, is for a two-bladed rotor in hover, but it should be noted that the method described herein is easily expanded to a more general case. Because compressibility effects are considered, the method can handle any typical helicopter operating state where the flow at the blade tips is subsonic.

As will be presented in detail later, the rotor wake can be modeled as a collection of a large number of vortex elements that all affect each other. Understanding how these elements affect each other is fundamental to understanding the wake in general. Problems of this nature are often termed "many-body" problems. Many-body problems are found throughout the scientific and engineering world, and include such fields as molecular dynamics, gas dynamics, charged particles in an electromagnetic field, and protein folding. While the actual physics behind these problems may be different, they are for the most part solved computationally using similar algorithms and procedures.

1.2 Rotor Wake Characteristics

The rotor wake consists of a shed vortex sheet and a concentrated vortex at the tip, as seen in *Figure 1.2*. There is a bound circulation on the rotor blade associated with lift, and conservation of vorticity requires that the circulation be trailed into the wake at the blade tip and root. The strong tip vortices are the dominant feature in the rotor wake.

Vorticity is also shed and trailed into the wake, creating the vortex sheet, as a result of changes in the circulation on the blade. The trailed vorticity is oriented parallel to the local free stream when it leaves the blade, similar to the tip vortex, while the shed vorticity is oriented radially in the wake, perpendicular to the trailed vorticity [16]. Because of the rotation of the blade, lift and circulation are highest near the tip. Both reach a maximum before decreasing rapidly to zero at the tip, which creates a trailing vorticity of high strength at the edge of the wake and causes the wake to roll-up quickly into a concentrated tip vortex. The same occurs at the root, but since the circulation drop-off from the maximum to the root is more gradual, the trailed vortex is weaker and the roll-up is not as pronounced [16]. *Figure 1.3* shows the bound circulation on the blade, the lift distribution on the blade, the vortex sheet, and the roll-up of the tip vortex.

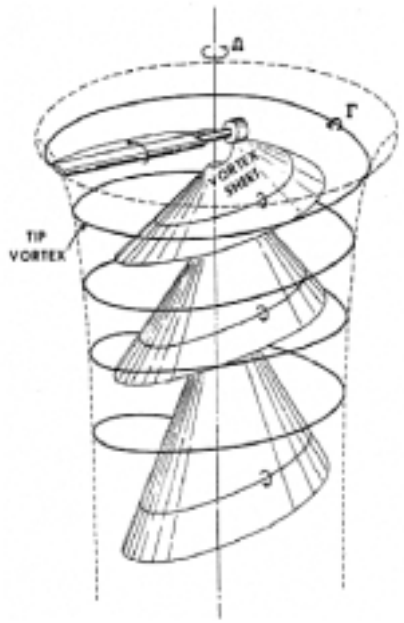


Figure 1.2: Traditional model of a rotor wake, showing a concentrated tip vortex and a trailing vortex sheet [21]

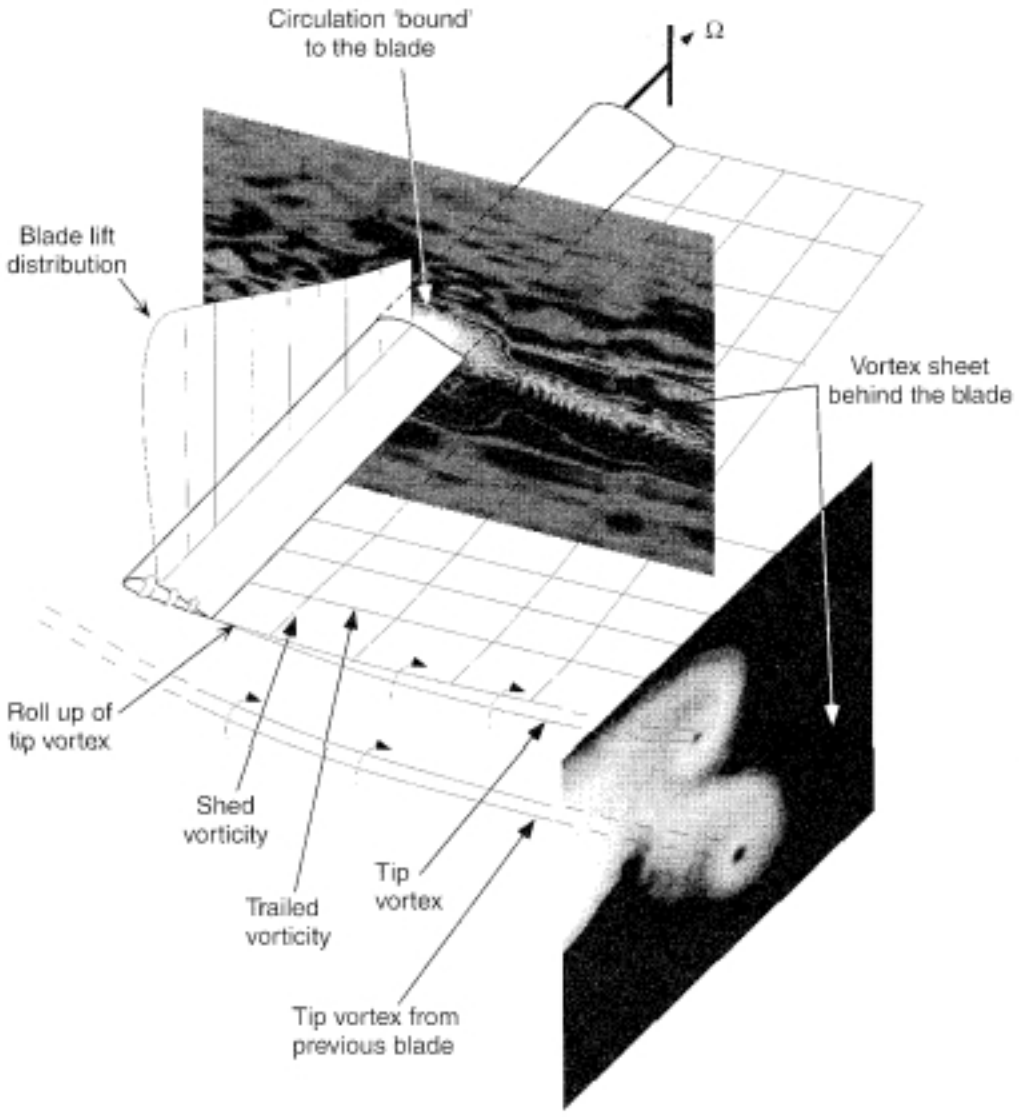


Figure 1.3: Schematic showing the wake and tip vortex roll-up [22]

Although the basic structure of the rotor wake is similar to that of a fixed-wing wake, there is the added complication that each blade might pass through or near the wakes of previous blades. This can be seen in *Figures 1.4* and *1.5*, which show the paths of tip vortices trailed from a fixed-wing and rotary-wing aircraft respectively. The path of the tip vortices, as well as the geometry and the strength of the rest of the wake, depends mainly on the operating state of the helicopter. In hover, the tip vortices follow a helical path as they are convected below the rotor disk. In forward flight, the wake is skewed behind the rotor, as seen in *Figure 1.6*. This results in a more complicated distortion and stronger interactions among the vortices in the wake [22].

Figure 1.4: Tip vortices trailed behind an E/A-6B Prowler [40]

Figure 1.5: Tip vortices trailed behind the blades of an AH-1W Super Cobra [40]

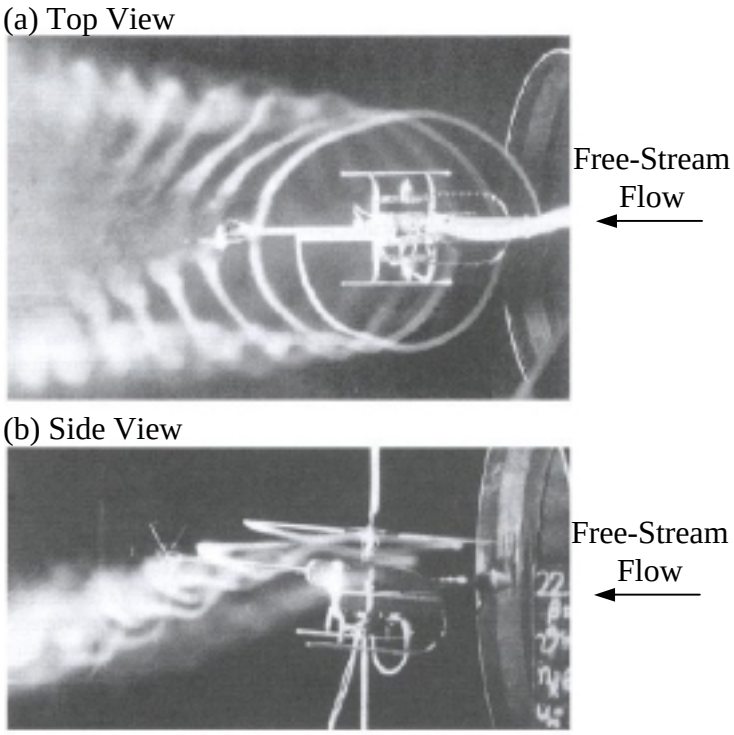


Figure 1.6: Smoke visualization of the tip vortex locations in forward flight [22]

Vorticity is also shed into the wake in the form of a vortex sheet as the blades rotate. The vortex sheet is convected downward, normal to the rotor disk, by the wake induced velocity and a component of the free stream velocity normal to the disk in forward flight and climb. The induced velocity is due to the vortex-vortex interactions in the wake.

Rotating blades encounter tip vortices shed from proceeding blades, which results in a phenomenon known as Blade Vortex Interaction (BVI). The blade experiences a sudden change in the local velocity induced by the approaching vortex. The vortex produces a downwash velocity as it approaches the blade and then a sudden upwash velocity as it passes the leading edge of the blade. These unsteady loads are an important factor in the vibration, noise, and performance of a helicopter. BVIs occur mainly on the advancing and retreating sides of the rotor disk and are most intense in forward flight [16].

1.3 Previous Research in Rotor Wake Simulation

The complicated nature of the rotor wake makes studying it, either experimentally or computationally, a complicated task. Flow visualization on scaled rotor models has been the most common technique to study the wake experimentally. Many tests have been completed using smoke flow visualization, shadowgraphy, hot-wire anemometry, and laser Doppler velocimetry [22]. The rotor wake can even be seen naturally through condensation in the tip vortices under certain conditions, as shown previously in *Figures 1.4* and *1.5*. While recent advances have allowed more accurate data to be obtained

through experimental techniques, advances in computational power have also allowed for more accurate mathematical modeling of the rotor wake. Current computational models usually contain simplifications that limit their usefulness to actual problems of significance, so there is still much work to be done to improve these models.

1.3.1 Vortex Methods

In a vortex method, the wake is modeled by vortex lines that are discretized into a lattice with either straight or curved segments. The circulation strength of each element in the wake is set by the circulation on the blade when the element is shed into the wake. The advantage of vortex models is that the strength and position of all wake elements are known, and thus, the induced velocity field on the blade due to the wake can be calculated. While less computationally intensive than finite-difference methods, vortex methods can still run into a problem of high computational requirements when a large number of elements is used to obtain more accurate results. There are two main classes of vortex methods, distinguished by how they solve for the position of the wake. In prescribed wake models, the location of the wake is assumed based on experimental or computational data. In free wake methods, however, the strength and position of the wake is solved for directly at each time step, which generally results in a more accurate but more time consuming solution.

Landgrebe [20] modeled the rotor wake in 1969 by starting with the classical rigid wake, based on undistorted helical sheets, and calculating the distorted geometry of the wake. He observed significant distortions to the rigid wake, and he saw the wake

contract and the tip vortices roll-up, as expected. Later, Landgrebe [21] used experimental data to create a prescribed wake and observed an improvement in the accuracy of the results compared to previous work. He used experimental data from a number of different rotor configurations to create a generalized wake model for a hovering rotor that could be used as the basis of a prescribed wake model. Egolf and Landgrebe [10] used a similar procedure to create a generalized wake model for a rotor in forward flight. Since not much experimental data is available for forward flight cases, however, the model was based on a combination of experimental data and a mathematical calculation of the distortion of a classical rigid wake. Beddoes [1] created a simple prescribed wake model for forward flight by using an undistorted wake and developing a set of distortion equations to modify the vertical displacement of the wake elements. This model showed good agreement with free wake models available at the time but was considerably less computationally intensive.

As the available computational power increased, the prospects of creating a true free wake model improved. Early work on free wake models was done by Clark and Lieper [7] in 1970 for heavy-lift helicopters in hover. They started with a basic helical wake shape represented by straight-line vortex filaments and allowed the wake to propagate over time, declaring convergence when the wake shape stopped changing between time steps. They observed that the tip vortex remained close to the tip path plane until it interacted with the next blade and was convected downward. The computed wake shape closely matched experimental data for the Sikorsky S-65. At about the same time, Sadler [31] developed a free wake model by impulsively starting the rotor blades

from rest with no wake and allowing the wake to develop behind the blades over time. Vortices were shed into the wake with strengths corresponding to the blade circulation strength at each time step and were allowed to translate with a speed based on the forward speed of the helicopter and the local induced velocity. He was able to obtain good results with a fine grid, but the computational requirements were too high, so certain simplifications had to be made. He used only streamwise segments of the vortices in the far wake to reduce the computational time without significant loss of accuracy and was able to produce a realistic wake geometry for advance ratios greater than 0.1. The model was not able to capture the more severe wake distortion and Blade Vortex Interactions at lower advance ratios. In 1988, however, Egolf and Massar [11] showed that modern advanced computers could handle free wake models without many of the simplifying assumptions previously used.

1.3.2 Computational Fluid Dynamics

Computational Fluid Dynamics (CFD) uses finite difference, finite volume, or finite element methods to solve the Euler and Navier-Stokes equations in the entire rotor flow field. CFD generally provides a more detailed view of the rotor wake than vortex methods but still has some outstanding issues that limit its usefulness. Although available computer power continues to increase, the complex nature of the rotor wake requires enormous computer resources to provide accurate results. CFD methods also exhibit numerical dissipation, an artificial viscosity that tends to cause unrealistic diffusion of the wake. The high grid resolution required to avoid numerical dissipation

and to capture the full unsteady aerodynamics of the rotating blades leads to computational requirements that are several orders of magnitude greater than those of vortex methods.

Chen and McCroskey [6] first captured the rotor wake by solving the Euler equations in 1988. The results showed good agreement with experimental data near the tip but exhibited considerable numerical diffusion. Still, it was a major step in the development of a method that would capture the rotor wake without using any *ad hoc* modeling. Srinivasan et al. [34] developed the TURNS code in 1992, which used the thin-layer Navier-Stokes equations to capture the rotor wake without any wake modeling. It showed fair agreement with experimental data except in the inboard region and tip region of the blade, and it predicted the flow separation seen in the tip region in experiments and the detailed structure of the tip vortex.

There has been much work done to improve the accuracy of these early CFD methods. Strawn and Barth [37] used unstructured grids with Euler equations and adaptive-grid refinement to improve the resolution of interesting flow features. Similarly, Duque and Srinivasan [9] used overset grids to improve the resolution of certain flow features by applying separate optimized grids to different areas of the domain. Despite the improvements of these and other methods, CFD in general is still too computationally intensive to be used in many rotor wake problems. More computational power must be available before CFD simulations with finer grids can be run that could truly capture the physics of the wake and limit the effects of numerical diffusion.

1.3.3 Hybrid Methods

One of the ways around the current limitations of CFD is to use a hybrid method that combines a finite-difference method in the near wake with a vortex method in the far wake. Tung et al. [39] combined two independent codes with only minor modifications to create a hybrid method in 1986. They used a finite-difference code near the blade tip to calculate the loading and passed that information to an integral code, which calculated the downwash effects and passed them back to the finite-difference code. Steinhoff and Ramachandran [36] developed an alternate method where, instead of treating the vortex sheets separately from the region near the blade, they are embedded into a compressible potential flow field. The method, which was implemented in a code, HELIX-I, calculated the free convection of the wake and eliminated the problems of numerical diffusion seen in true CFD codes. Later, Moultan et al [28] combined an overset version of HELIX-I with the TURNS Navier-Stokes code. They used the TURNS code to calculate the viscous effects on the blade and HELIX-I to calculate the free wake convection. Sezer-Uzol and Long [33] developed an approach at Penn State to preserve vortices over longer distances in a coupled Euler/vortex method, which also reduced the effects of numerical diffusion seen in pure CFD codes. It produced a more accurate model of the tip vortex in the far wake than what is seen in most CFD codes, where the tip vortex is quickly dissipated.

A different approach was applied by Berkmen et al. [4] in 1997. They separated the flow field into three different parts. First, the unsteady, compressible Navier-Stokes equations were solved in an inner zone surrounding each blade to capture the wake and

viscous effects. Next, the isentropic potential flow equations were solved in an inviscid outer zone, which was used to carry pressure waves to the far wake. Finally, a Lagrangian wake zone inside the outer zone was used to model the wake. It captured the vorticity leaving the viscous region and convected it to the far field by solving the Biot-Savart Law.

1.3.4 Compressibility

One of the most important and challenging aspects of modeling the rotor wake is accounting for the effects of compressibility, since it is common for the tips of advancing blades to experience high-Mach number flows in forward flight. As was discussed in previous sections, many CFD and hybrid-CFD methods already account for compressibility, but there are other ways to consider its effects as well. A supersonic vortex lattice method was implemented for arbitrary wings in the VORLAX code at Lockheed-California in 1977, in which the induced velocity equations were rederived starting with the governing equations for inviscid compressible flow [28].

A method introduced by Morino [29] in 1974 and later extended to arbitrary motion by Gennaretti and Morino [14] used a new integral equation derived from the velocity potential equation to account for subsonic, compressible flow. In application of the method to a hovering rotor, they used the wake geometry obtained from an incompressible free wake analysis as a prescribed wake for the compressible case. They compared their results with a crude application of the Prandtl-Glauert correction, which consisted only of dividing the spanwise pressures by a correction factor, and concluded

that their results showed good agreement with the experiment but that the Prandtl-Glauert correction over-predicted lift.

Long and Watts [26] used an “acoustic analogy approach” to develop an integral equation based on the Ffowcs Williams-Hawkings equation to solve for compressible arbitrary motion. They accounted for wake effects by storing the time history of the blades and calculating the effects of the pressure jump that lingers on at each previous position of the blade. Because of the finite speed of propagation of the disturbances caused by the pressure jumps, a time lag exists before the effects of each jump are felt on the blade. Epstein and Bliss [12] investigated the effect of compressibility during the initial stages of wake development using a similar technique. New wake elements produce a pressure wave at the instant they are emitted, and these waves propagate both upstream and downstream at the speed of sound. They calculated the time required for newly emitted waves to reach the leading edge of a wing in forward flight and the distance the wing traveled in that time. They used compressible methods for the near wake region within that characteristic length but incompressible methods for the far wake. The combined compressible-incompressible method showed considerable improvement over a fully incompressible method and produced results very similar to those of a fully compressible method. Therefore, it was concluded that the effects of compressibility are related to the wake generation process and are confined to the very near wake.

1.4 Current Approach

The goal of this work is to study vortex-vortex interactions by developing a method that will model the wake of a multi-bladed rotor following an arbitrary flight path. While the code will be able to handle such motion, it will only be validated for a two-bladed hovering rotor, for which extensive experimental data is available. Since compressibility is a factor in forward flight but usually not a factor in hover, an unrealistic hover case will be run where the flow is compressible at the tips to show that the method can account for it. The best way to observe vortex-vortex interactions is through a vortex method, since vortex methods are based on the effects of these interactions. In order to obtain the best results, a free wake method was chosen, since the free wake method provides a good balance between accuracy and computational efficiency. In general, vortex methods cannot account for the effects of compressibility, so a Prandtl-Glauert correction will be added to account for these effects. Unlike the Prandtl-Glauert correction used by Gennaretti and Morino [14] and discussed previously, this correction will be applied directly to the calculation of vortex-induced effects rather than to the final pressure calculations on the blade. A detailed description of the free wake method and the compressibility correction will be discussed, as well as the development of the code to implement these methods and the results for two hover cases. The ability of the code to be used to study more complicated helicopter operating states will also be highlighted through a series of example cases.

Chapter 2

FUNDAMENTALS OF POTENTIAL FLOW

2.1 Overview

Vortex methods are based on the interaction of vortices in the wake and the calculation of vortex-induced velocity fields on the rotor blades and in the wake. It is necessary to understand the fundamentals of the flow field in and around a vortex, however, before considering the vortex-vortex interactions. The Biot-Savart Law, which is the basis for calculating vortex-induced velocities, will be developed in this chapter. Since the Biot-Savart Law is valid only for incompressible flow, the Prandtl-Glauert compressibility correction will also be developed and applied to the Biot-Savart Law. It should be noted that the following work is only valid for regions of inviscid flow. Although the results cannot be used to account for the effects of the thin, viscous boundary layer on the surface of the blades, it will be shown later that they do provide a suitable approximation to the actual flow field.

2.2 Vorticity and Circulation

It is beneficial to start by defining vorticity and circulation, since they will be used extensively in the vortex method. In general, the motion of a fluid particle consists

of translation, rotation, and deformation. The focus of this chapter will be on rotation. The angular velocity of a fluid particle in vector notation is

$$\omega = \frac{1}{2}\nabla \times \mathbf{q} \qquad (\ 2.1\ )$$

where **q** represents the velocity field of the particle. Vorticity is defined as twice the angular velocity,

$$\zeta \equiv 2\omega = \nabla \times \mathbf{q} \qquad (\ 2.2\ )$$

and circulation is defined as

$$\Gamma \equiv \int_C \mathbf{q} \cdot d\mathbf{l} \qquad (\ 2.3\ )$$

From *Figure 2.1*, which shows a surface *S* enclosed by the curve *C*, the circulation can be related to the vorticity on the surface using Stokes' Theorem:

$$\Gamma = \int_C \mathbf{q} \cdot d\mathbf{l} = \int_S \nabla \times \mathbf{q} \cdot \mathbf{n} dS = \int_S \zeta \cdot \mathbf{n} dS \qquad (\ 2.4\ )$$

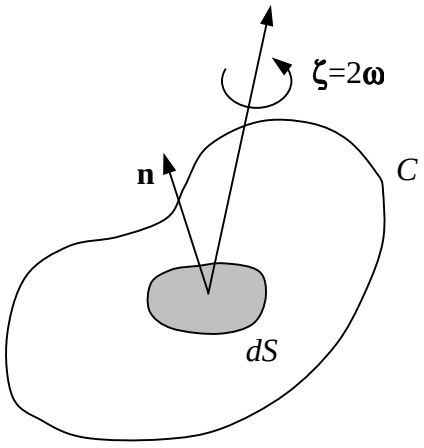


Figure 2.1: Relation between surface integral and line integral [18]

2.3 Two-Dimensional Vortex

To illustrate a two-dimensional vortex, it is useful to consider the solid cylinder shown in *Figure 2.2a*, with radius *R*, rotating in a viscous fluid at an angular velocity of ω. This rotation causes circular streamlines to develop around the cylinder, and the tangential velocity of these streamlines is given by [18]

$$q_\theta = -\frac{\Gamma}{2\pi r} \qquad (2.5)$$

where the circulation is

$$\Gamma = 2\omega\pi R^2 \qquad (2.6)$$

The velocity field will be derived in detail for a three-dimensional vortex in the next section, and it will be shown that the result simplifies to Equation 2.5 for the two-dimensional case. The tangential velocity of a point in the solid body, where *r* is less than *R*, is simply given as

$$q_\theta = r\omega \qquad (2.7)$$

The total velocity distribution given by Equations 2.5 and 2.7 is shown in *Figure 2.2b*. The significance of the relationship between the velocity inside the solid body and outside the body will be discussed later when the concept of the vortex core is introduced.

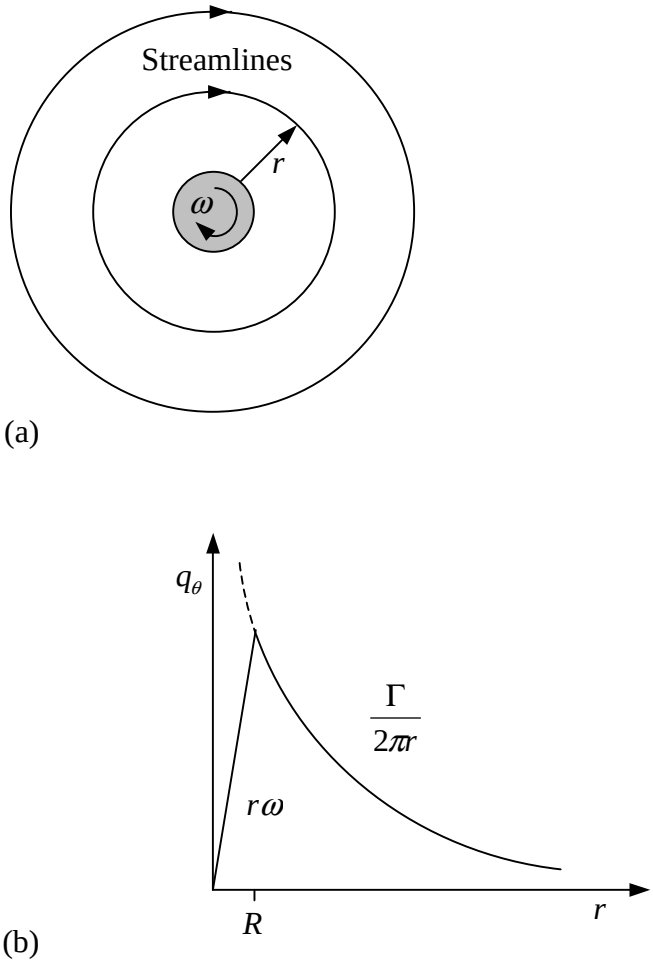


Figure 2.2: Two-dimensional flow field around a solid rotating cylinder showing (a) streamlines and (b) tangential velocity of flow [18]

2.4 Three-Dimensional Vortex

The three-dimensional induced-velocity field can be calculated using the Biot-Savart Law, which is derived next, based on Refs. 3 and 18. Starting with the assumption that the flow is incompressible, the continuity equation is

$$\nabla \cdot \mathbf{q} = 0 \qquad (2.8)$$

The velocity field **q** can be expressed as the curl of a vector field **B**,

$$\mathbf{q} = \nabla \times \mathbf{B} \qquad (2.9)$$

where **B** is selected such that

$$\nabla \cdot \mathbf{B} = 0 \qquad (2.10)$$

The vorticity can then be expressed in terms of **B**:

$$\zeta = \nabla \times \mathbf{q} = \nabla \times (\nabla \times \mathbf{B}) = \nabla(\nabla \cdot \mathbf{B}) - \nabla^2 \mathbf{B}$$

Using the condition set in Equation 2.10, the vorticity equation reduces to Poisson's Equation for the vector field **B**

$$\zeta = -\nabla^2 \mathbf{B} \qquad (2.11)$$

Equation 2.11 can solved using Green's Theorem to evaluate the vector field at a point *P* due to the vorticity in a volume *V* as shown in *Figure 2.3*.

$$\mathbf{B} = \frac{1}{4\pi} \int_V \frac{\zeta}{|\mathbf{r}_0 - \mathbf{r}_1|} dV \qquad (2.12)$$

Substituting this result into Equation 2.9, the velocity field at *P* is

$$\mathbf{q} = \frac{1}{4\pi} \int_V \nabla \times \frac{\zeta}{|\mathbf{r}_0 - \mathbf{r}_1|} dV \qquad (2.13)$$

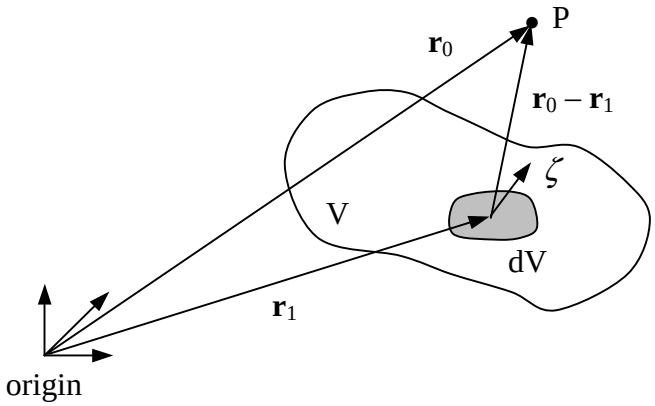


Figure 2.3: Velocity at point P due to a vortex distribution in volume V [18]

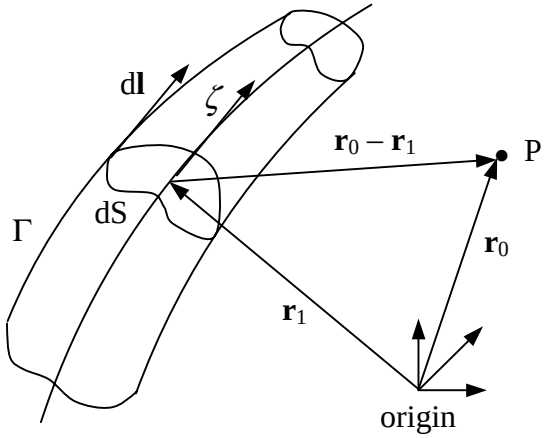


Figure 2.4: Velocity at point P induced by a vortex segment [18]

It is necessary to make a substitution for the integrand before evaluating Equation 2.13. Considering the vorticity filament shown in *Figure 2.4*, where dS and d**l** are normal to the filament, the circulation is

$$\Gamma = \zeta dS$$

and the volume is related to the surface by

$$dV = dSdl$$

Therefore, the integrand in Equation 2.13 becomes

$$\nabla \times \frac{\zeta}{|\mathbf{r}_0 - \mathbf{r}_1|} dV = \nabla \times \Gamma \frac{d\mathbf{l}}{|\mathbf{r}_0 - \mathbf{r}_1|}$$

or, if the curl operation is evaluated while keeping $\mathbf{r}_1$ and d**l** fixed,

$$\nabla \times \Gamma \frac{d\mathbf{l}}{|\mathbf{r}_0 - \mathbf{r}_1|} = \Gamma \frac{d\mathbf{l} \times (\mathbf{r}_0 - \mathbf{r}_1)}{|\mathbf{r}_0 - \mathbf{r}_1|^3} \qquad (\ 2.14\ )$$

Substituting this into Equation 2.13 produces the Biot-Savart Law:

$$\mathbf{q} = \frac{\Gamma}{4\pi} \int \frac{d\mathbf{l} \times (\mathbf{r}_0 - \mathbf{r}_1)}{|\mathbf{r}_0 - \mathbf{r}_1|^3} \qquad (\ 2.15\ )$$

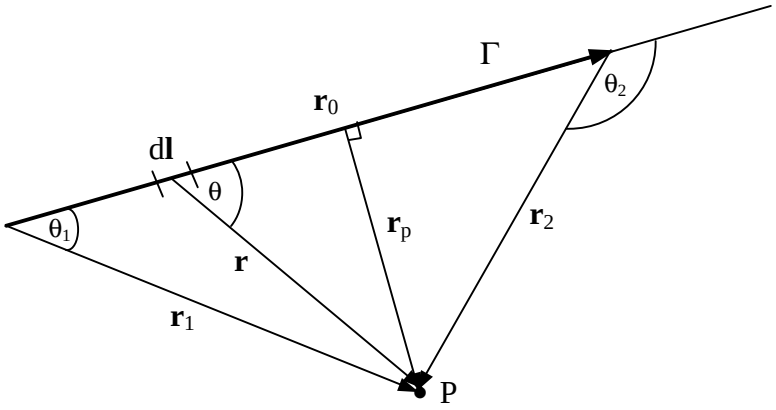


Figure 2.5: Velocity at point P induced by a vortex segment [3]

The general form of the Biot-Savart Law, Equation 2.15, must be integrated before it can be used in the numerical method. It can be written in differential form in terms of the new notation shown in *Figure 2.5*:

$$\Delta\mathbf{q} = \frac{\Gamma}{4\pi}\frac{d\mathbf{l}\times\mathbf{r}}{r^3} \qquad (2.16)$$

The cross product is evaluated,

$$d\mathbf{l}\times\mathbf{r} = r\sin\theta dl$$

and the following substitutions are made, based on the geometry shown:

$$r_p = r\sin\theta, \qquad r = \frac{r_p}{\sin\theta}$$

$$\tan(\pi-\theta) = \frac{r_p}{r_0}, \qquad r_0 = \frac{-r_p}{\tan\theta}, \qquad dl = \frac{r_p}{\sin^2\theta}d\theta$$

Equation 2.16 can then be rewritten in scalar form as

$$\Delta q = \frac{\Gamma}{4\pi}\frac{r\sin\theta dl}{r^3} = \frac{\Gamma}{4\pi}(\sin\theta)\left(\frac{\sin^2\theta}{r_p^2}\right)\left(\frac{r_p}{\sin^2\theta}d\theta\right) = \frac{\Gamma}{4\pi r_p}\sin\theta d\theta \quad (2.17)$$

Equation 2.17 is integrated over the length of the segment:

$$q = \frac{\Gamma}{4\pi r_p}\int_{\theta_1}^{\theta_2}\sin\theta d\theta = \frac{\Gamma}{4\pi r_p}(\cos\theta_1 - \cos\theta_2) \quad (2.18)$$

Converting back to vector notation and making the following geometric substitutions,

$$r_p = \frac{|\mathbf{r}_1 \times \mathbf{r}_2|}{r_0} \qquad \cos\theta_1 = \frac{\mathbf{r}_0 \cdot \mathbf{r}_1}{r_0 r_1} \qquad \cos\theta_2 = \frac{\mathbf{r}_0 \cdot \mathbf{r}_2}{r_0 r_2}$$

the final expression for the induced velocity at a point due to a straight-line vortex is

$$\mathbf{q} = \frac{\Gamma}{4\pi}\frac{\mathbf{r}_1 \times \mathbf{r}_2}{|\mathbf{r}_1 \times \mathbf{r}_2|^2}\left[\frac{\mathbf{r}_0 \cdot \mathbf{r}_1}{r_1} - \frac{\mathbf{r}_0 \cdot \mathbf{r}_2}{r_2}\right] \quad (2.19)$$

Equation 2.19 is the general form of the Biot-Savart Law which will be used extensively in the vortex method to determine the vortex-induced velocities.

To see how this relates to the two-dimensional velocity field presented in the last section, consider the case of an infinite vortex, where $\theta_1 = 0$ and $\theta_2 = \pi$. Equation 2.18 reduces to

$$q_\theta = \frac{\Gamma}{4\pi r}(\cos(0) - \cos(\pi)) = \frac{\Gamma}{2\pi r} \quad (2.20)$$

which matches the result for the solid cylinder. As in the two-dimensional case, though, the Biot-Savart Law is only valid outside of the vortex core.

2.5 Vortex Core

The flow field inside a vortex can be seen in *Figure 2.6*, which shows a vortex sheet and the concentrated roll-up of the tip vortex. The flow follows circular streamlines around the vortex core and is similar to the two-dimensional flow field around the solid cylinder.

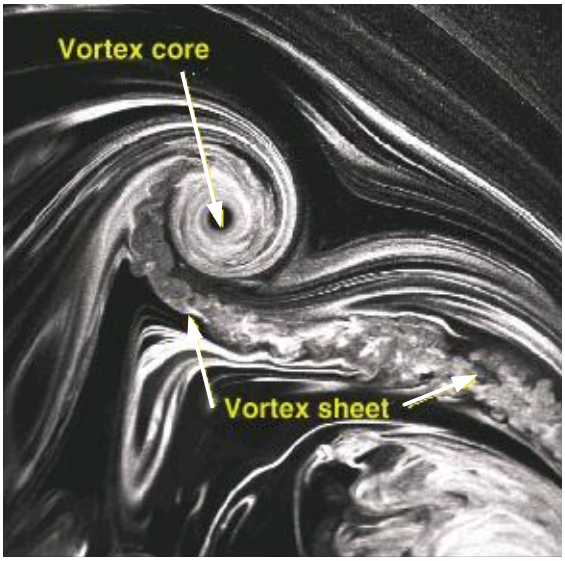


Figure 2.6: Laser light sheet flow visualization tip vortex and vortex sheet [23]

Inside the vortex core, where the flow is viscous, the tangential velocities should tend to zero, as shown in *Figure 2.7a* and by the dotted line in *Figure 2.7b*. The induced velocities calculated with the potential flow model, however, increase towards infinity inside the core, as shown by the dashed line in *Figure 2.7b*. Therefore, it is necessary to model the flow inside the core using a different method. Since the flow field in the vortex core is similar to the flow field around a solid cylinder, the simplest approach is to

use an approximation based on the solid body rotation developed earlier. Inside the core, the velocity field is represented by

$$q_\theta = \left(\frac{\Gamma}{2\pi r_c} \right) \left(\frac{r}{r_c} \right) \qquad (2.21)$$

which is a straight line from $r = 0$ to $r = r_c$. Outside the core, the velocities are calculated using the Biot-Savart Law, Equation 2.19. The solid line in *Figure 2.7b* represents the combination of the solid body rotation approximation and the Biot-Savart Law, which is known as the Rankine vortex model. This is one of the simplest vortex models and the one that will be used in this method. The size of the vortex core is known to grow somewhat as a function of wake age [22], but for simplicity, the core radius was set as a constant in this model.

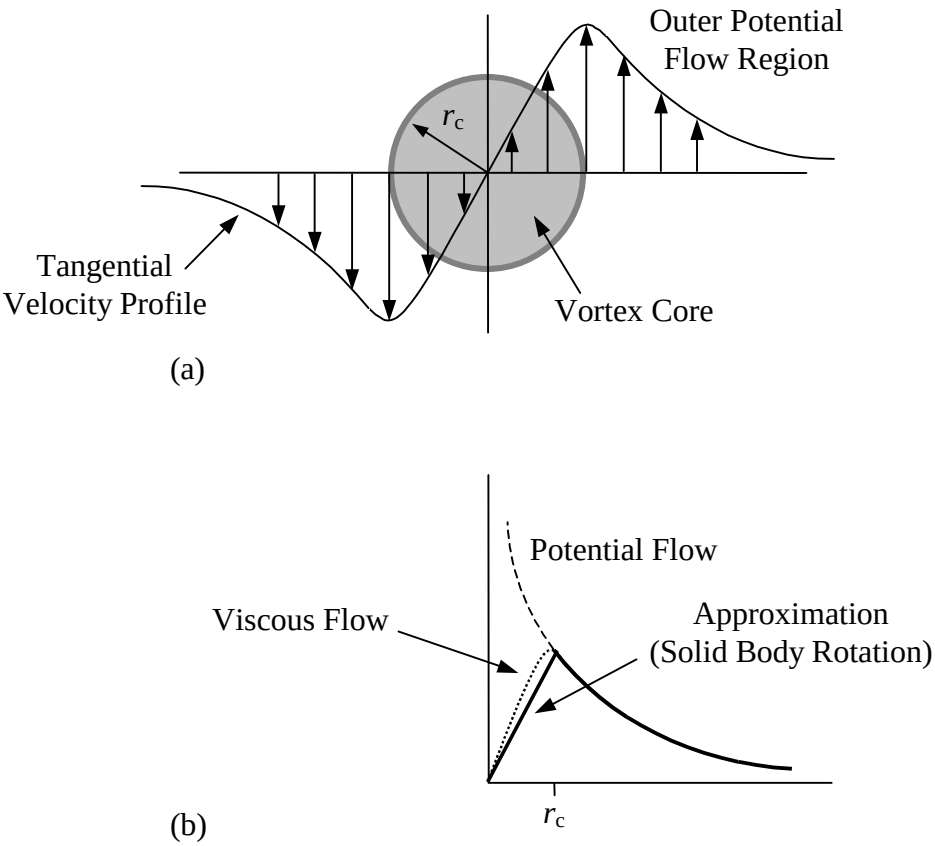


Figure 2.7: Velocity field inside a tip vortex shown by (a) idealized view [22] and (b) model used in vortex method [35]

Alternative vortex models have also been developed [22]. The first, created by Oseen and Lamb, is based on a simplified form of the Navier-Stokes equation:

$$q_\theta(r) = \frac{\Gamma}{2\pi r_c (r/r_c)}\left(1 - e^{-\alpha (r/r_c)^2}\right) \qquad (2.22)$$

where $\alpha = 1.25643$. Other similar models were also developed by Scully,

$$q_\theta(r) = \left(\frac{\Gamma}{2\pi r_c}\right)\frac{r/r_c}{\left(1 + (r/r_c)^2\right)} \qquad (2.23)$$

and Vatistas et al,

$$q_\theta(r) = \left(\frac{\Gamma}{2\pi r_c}\right)\frac{r/r_c}{\sqrt{1 + (r/r_c)^4}} \qquad (2.24)$$

A comparison of these models is shown in *Figure 2.8*. Notice that the velocities calculated by each method are similar away from the core but that there are significant differences inside and just outside of the core.

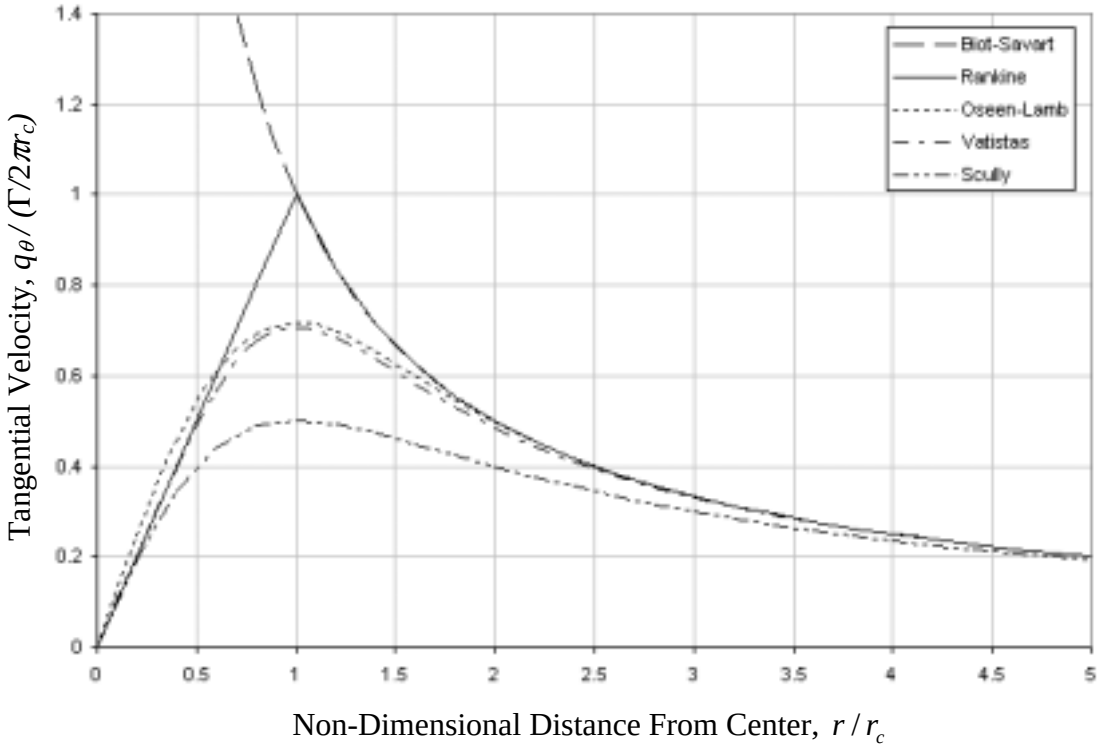


Figure 2.8: Comparison of different 2-D vortex models

2.6 Compressibility Correction

The derivation of the compressibility correction is based on Refs. 3, 8, and 13. It is useful to start with the continuity equation and the Navier-Stokes equation. This time, however, the assumption of incompressible flow is not made, and the continuity equation is presented using tensor notation:

$$\frac{\partial \rho}{\partial t} + \frac{\partial}{\partial x_k}(\rho u_k) = 0 \qquad (2.25)$$

Ignoring external body forces, the Navier-Stokes equation is

$$\rho \frac{\partial u_j}{\partial t} + \rho u_k \frac{\partial u_j}{\partial x_k} = -\frac{\partial p}{\partial x_j} + \frac{\partial}{\partial x_j}\left(\lambda \frac{\partial u_k}{\partial x_k}\right) + \frac{\partial}{\partial x_i}\left[\mu\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right)\right] \qquad (2.26)$$

Assuming that the effects of viscosity are negligible, Equation 2.26 reduces to the Euler equation:

$$\rho \frac{\partial u_j}{\partial t} + \rho u_k \frac{\partial u_j}{\partial x_k} = -\frac{\partial p}{\partial x_j} \qquad (2.27)$$

The continuity equation and Euler equation can be linearized by considering small disturbances in a uniform flow in a coordinate system that is at rest with respect to the flow (i.e. $u_0 = 0$). The change in the velocity component is u_i, where u_i is so small that u_i^2 is negligible compared to u_i; p_0 and ρ_0 are the pressure and density of the undisturbed flow; and p' and ρ' are the small perturbations in the pressure and density, such that

$$p = p_0 + p' \qquad \rho = \rho_0 + \rho'$$

The linear continuity equation then becomes

$$\frac{\partial}{\partial t}(\rho_0 + \rho') + \frac{\partial}{\partial x_k}[(\rho_0 + \rho')(u_k)] = 0$$

or, eliminating higher order terms and assuming ρ_0 is a constant,

$$\frac{\partial \rho'}{\partial t} + \rho_0 \frac{\partial u_k}{\partial x_k} = 0 \qquad (2.28)$$

Likewise, the linear Euler equation becomes

$$(\rho_0 + \rho')\frac{\partial u_k}{\partial t} + (\rho_0 + \rho')(u_k)\frac{\partial u_k}{\partial x_k} = -\frac{\partial}{\partial x_k}(p_0 + p')$$

or, eliminating higher order terms,

$$\rho_0 \frac{\partial u_k}{\partial t} = -\frac{\partial p'}{\partial x_k} \qquad (2.29)$$

Taking the time derivative of Equation 2.28 and the spatial derivative of Equation 2.29 yields

$$\frac{\partial^2 u_k}{\partial x_k \partial t} = -\frac{1}{\rho_0}\frac{\partial^2 \rho'}{\partial t^2}$$

and

$$\frac{\partial^2 u_k}{\partial x_k \partial t} = -\frac{1}{\rho_0}\frac{\partial^2 p'}{\partial x_k \partial x_k}$$

Combining these leads to

$$\frac{\partial^2 p'}{\partial x_k \partial x_k} - \frac{\partial^2 \rho'}{\partial t^2} = 0 \qquad (2.30)$$

If the following substitution is made,

$$a^2 = \frac{dp}{d\rho} = \frac{p'}{\rho'}$$

then Equation 2.30 can be rewritten as

$$\frac{\partial^2 p'}{\partial x_k \partial x_k} - \frac{1}{a^2}\frac{\partial^2 p'}{\partial t^2} = 0 \qquad (2.31)$$

which is the fundamental wave equation in acoustics for a fluid at rest at infinity. If small perturbations are again considered, then

$$\begin{aligned} a^2 = \frac{dp}{d\rho} &= \left(\frac{dp}{d\rho}\right)_{\rho'=0} + \rho'\left(\frac{d^2 p}{d\rho^2}\right)_{\rho'=0} + \cdots \\ &= a_0{}^2 + O(\rho') \end{aligned}$$

The term a_0 is the speed of propagation of a sound wave in the undisturbed fluid and is usually written as

$$a_0 = \sqrt{\left(\frac{\partial p}{\partial \rho}\right)_S} \qquad (2.32)$$

to indicate that the derivative is taken at constant entropy. In a perfect gas, the speed of sound can be written as

$$a_0 = \sqrt{\frac{\gamma p_0}{p_0}} \qquad (2.33)$$

or

$$a_0 = \sqrt{\gamma R T} \qquad (2.34)$$

where γ is the ratio of heat capacities, R is the universal gas constant, and T is temperature.

Using this result and neglecting higher order terms, Equation 2.31 becomes

$$\frac{\partial^2 p'}{\partial x_k \partial x_k} - \frac{1}{a_0^{\,2}} \frac{\partial^2 p'}{\partial t^2} = 0 \qquad (2.35)$$

This is an important result because vortices produce disturbances that propagate from the trailing edge of the blade with the speed of sound when they are shed into the wake. In incompressible flow, the speed of sound is assumed to be infinite, and the disturbances are instantaneously transmitted to infinity in all directions. In compressible flow, however, the speed of sound is finite, and there is a time delay before the disturbances are felt throughout the domain [13].

It is necessary at this point to define the velocity potential and derive the potential equation for incompressible flow. If the flow is irrotational, then

$$\nabla \times \mathbf{u} = 0 \qquad (2.36)$$

The velocity vector can be expressed as the gradient of a scalar function ϕ, since

$$\nabla \times \nabla \phi \equiv 0$$

is an identity for any scalar function. Therefore,

$$\begin{aligned} \mathbf{u} &= \nabla \phi \\ u_k &= \frac{\partial \phi}{\partial x_k} \end{aligned} \qquad (2.37)$$

The function ϕ is referred to as the velocity potential of the flow field. For incompressible flow, the continuity equation reduces to

$$\begin{aligned} \frac{\partial u_k}{\partial x_k} &= 0 \\ \frac{\partial^2 \phi}{\partial x^2} + \frac{\partial^2 \phi}{\partial y^2} + \frac{\partial^2 \phi}{\partial z^2} &= 0 \end{aligned} \qquad (2.38)$$

Equation 2.38 is the potential equation for incompressible flow.

The velocity potential equation for compressible flow can be derived from the wave equation. First, it is necessary to show that the velocity potential also satisfies the wave equation. Recall that

$$u_k = \frac{\partial \phi}{\partial x_k} \qquad (2.37)$$

and

$$\rho_0 \frac{\partial u_k}{\partial t} = -\frac{\partial p'}{\partial x_k} \qquad (2.29)$$

Substituting Equation 2.37 into 2.29 and rearranging, 2.39

$$\rho_0 \frac{\partial}{\partial t}\left(\frac{\partial \phi}{\partial x_k}\right) + \frac{\partial p'}{\partial x_k} = \frac{\partial}{\partial x_k}\left(\rho_0 \frac{\partial \phi}{\partial t} + p'\right) = 0 \qquad (2.39)$$

Therefore,

$$\rho_0 \frac{\partial \phi}{\partial t} + p' = 0$$

and

$$p' = -\rho_0 \frac{\partial \phi}{\partial t} \qquad (2.40)$$

Since the disturbed pressure is proportional to the velocity potential, the velocity potential also satisfies the wave equation, and the wave equation can be rewritten as

$$\frac{\partial^2 \phi}{\partial x_k \partial x_k} - \frac{1}{{a_0}^2} \frac{\partial^2 \phi}{\partial t^2} = 0 \qquad (2.41)$$

The wave equation was derived using a coordinate system that was at rest with respect to the fluid. If the coordinate system is moving with a speed U in the negative x-direction, it is necessary to apply a coordinate transformation:

$$x = x' - Ut', \qquad y = y', \qquad z = z', \qquad t = t'$$
$$x' = x + Ut'$$

Looking at the time derivative term in Equation 2.41,

$$\frac{\partial \phi}{\partial t'} = \frac{\partial \phi}{\partial x'}\left(\frac{\partial x'}{\partial t'}\right) + \frac{\partial \phi}{\partial t'}\left(\frac{\partial t'}{\partial t'}\right)$$
$$= U\frac{\partial \phi}{\partial x'} + \frac{\partial \phi}{\partial t'}$$

and

$$\frac{\partial^2 \phi}{\partial t'^2} = \frac{\partial}{\partial t'}\left(U\frac{\partial \phi}{\partial x'}\right) + \frac{\partial}{\partial t'}\left(\frac{\partial \phi}{\partial t'}\right)$$
$$= U\frac{\partial}{\partial x'}\left(\frac{\partial \phi}{\partial t'}\right) + \frac{\partial}{\partial t'}\left(\frac{\partial \phi}{\partial t'}\right)$$
$$= U\frac{\partial}{\partial x'}\left(U\frac{\partial \phi}{\partial x'} + \frac{\partial \phi}{\partial t'}\right) + \frac{\partial}{\partial t'}\left(U\frac{\partial \phi}{\partial x'} + \frac{\partial \phi}{\partial t'}\right)$$
$$= \left(U^2\frac{\partial^2 \phi}{\partial x'^2} + U\frac{\partial^2 \phi}{\partial x' \partial t'}\right) + \left(U\frac{\partial^2 \phi}{\partial x' \partial t'} + \frac{\partial^2 \phi}{\partial t'^2}\right)$$
$$= \frac{\partial^2 \phi}{\partial t'^2} + 2U\frac{\partial^2 \phi}{\partial x' \partial t'} + U^2\frac{\partial^2 \phi}{\partial x'^2}$$

Therefore, the wave equation can be rearranged into the form

$$\left(\frac{\partial^2 \phi'}{\partial x'^2} + \frac{\partial^2 \phi'}{\partial y'^2} + \frac{\partial^2 \phi'}{\partial z'^2}\right) - \frac{1}{{a_0}^2}\left(\frac{\partial^2 \phi'}{\partial t'^2} + 2U\frac{\partial^2 \phi'}{\partial x' \partial t'} + U^2\frac{\partial^2 \phi'}{\partial x'^2}\right) = 0$$
$$-\frac{1}{{a_0}^2}\frac{\partial^2 \phi'}{\partial t'^2} - \frac{2U}{{a_0}^2}\frac{\partial^2 \phi'}{\partial x' \partial t'} - \frac{1}{{a_0}^2}\left(U^2 - 1\right)\frac{\partial^2 \phi'}{\partial x'^2} + \frac{\partial^2 \phi'}{\partial y'^2} + \frac{\partial^2 \phi'}{\partial z'^2} = 0$$

or, in terms of the Mach number,

$$-\frac{1}{a_0{}^2}\frac{\partial^2\phi}{\partial t'^2}-\frac{2M}{a_0}\frac{\partial^2\phi}{\partial x'\partial t'}+\left(1-M^2\right)\frac{\partial^2\phi}{\partial x'^2}+\frac{\partial^2\phi}{\partial y'^2}+\frac{\partial^2\phi}{\partial z'^2}=0 \qquad (2.42)$$

For steady flow, the time derivative terms can be eliminated, which leaves

$$\left(1-M^2\right)\frac{\partial^2\phi}{\partial x^2}+\frac{\partial^2\phi}{\partial y^2}+\frac{\partial^2\phi}{\partial z^2}=0 \qquad (2.43)$$

where the primes have been omitted for clarity. Equation 2.43 is the well-known Prandtl-Glauert equation, which is a linearized small-perturbation form of the full velocity potential equation for a flow with a mean velocity in the *x*-direction. It is in a similar form as the velocity potential equation for incompressible flow:

$$\frac{\partial^2\phi}{\partial x^2}+\frac{\partial^2\phi}{\partial y^2}+\frac{\partial^2\phi}{\partial z^2}=0 \qquad (2.38)$$

Applying another coordinate transformation to Equation 2.43, where

$$x'=\frac{x}{\sqrt{1-M^2}};\qquad y'=y;\qquad z'=z$$

leads to

$$\frac{\partial^2\phi}{\partial x'^2}+\frac{\partial^2\phi}{\partial y'^2}+\frac{\partial^2\phi}{\partial z'^2}=0 \qquad (2.44)$$

Therefore, applying this coordinate transformation creates an analogy between a compressible flow domain and an incompressible flow domain. This allows for a solution to the subsonic compressible flow field with small disturbances to be found using methods that were developed for incompressible flow. The effect of this transformation is seen in *Figure 2.9*, which shows that the domain can be stretched in the direction of the flow to create the analogous domain that can be solved using

incompressible methods. This stretching effect is known as the Prandtl-Glauert compressibility correction and will be used with the Biot-Savart Law to account for compressibility effects on the blade. The stretching effect on a rotating blade in hover is shown in *Figure 2.10*, where the dashed line represents the original geometry and the solid line represents the stretched geometry of the blade. Since the tangential velocity varies linearly along the span of the blade, the stretching effect is more pronounced at the tip where the flow speed is highest.

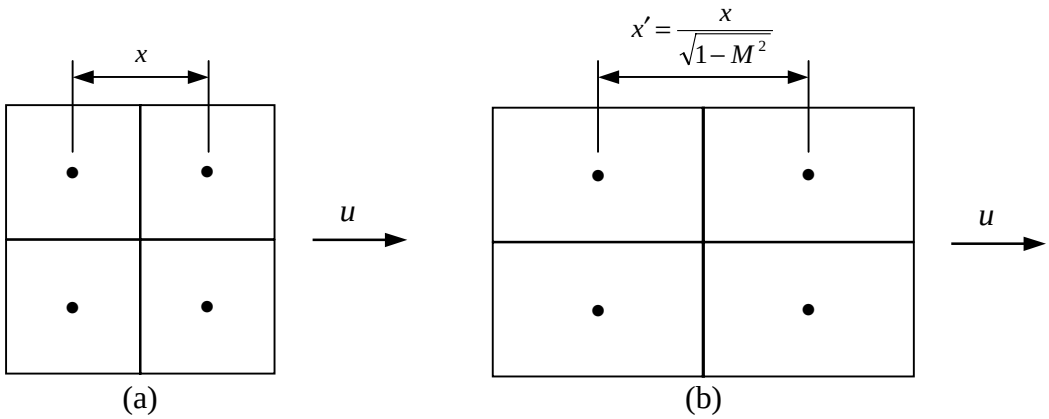


Figure 2.9: Effect of Prandtl-Glauert transformation: (a) actual domain in compressible flow; (b) stretched domain in analogous incompressible flow

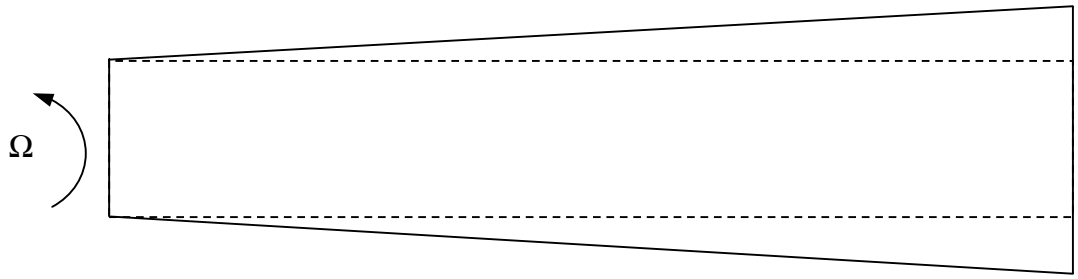


Figure 2.10: Stretching effect on a rotating blade

2.7 Applying the Prandtl-Glauert Transformation to the Biot-Savart Law

Because the Biot-Savart Law was derived for incompressible flow, it is necessary to apply the Prandtl-Glauert transformation before it can be used in problems where the flow is compressible. The Biot-Savart Law was given as Equation 2.19:

$$\mathbf{q} = \frac{\Gamma}{4\pi} \frac{\mathbf{r}_1 \times \mathbf{r}_2}{\left|\mathbf{r}_1 \times \mathbf{r}_2\right|^2} \left[\frac{\mathbf{r}_0 \cdot \mathbf{r}_1}{r_1} - \frac{\mathbf{r}_0 \cdot \mathbf{r}_2}{r_2} \right] \qquad (\ 2.19\ )$$

If a frame of reference is used where the control point is fixed and the vortex is moving, it is possible to find the component of the velocity and Mach number of the vortex with respect to the point *P* along the perpendicular vector between the vortex and the point. Two vectors are identified that represent the vector defined by the vortex segment, $\mathbf{r_0}$, and

the vector from one endpoint of the segment to the control point, $\mathbf{r_1}$. The projection of $\mathbf{r_1}$ on $\mathbf{r_0}$ is

$$\text{proj}_{\mathbf{r_0}}\mathbf{r_1} = \left(\frac{\mathbf{r_0}\cdot\mathbf{r_1}}{\left|\mathbf{r_0}\right|^2}\right)\mathbf{r_0} \qquad (2.45)$$

and the projection of $\mathbf{r_1}$ perpendicular to $\mathbf{r_0}$ is

$$\text{proj}_{\perp\mathbf{r_0}}\mathbf{r_1} = \mathbf{r_1} - \text{proj}_{\mathbf{r_0}}\mathbf{r_1} \qquad (2.46)$$

Equations 2.45 and 2.46 [2] are used to calculate the components of the perpendicular vector $\mathbf{r_p}$ between the vortex segment and the control point. Likewise, Equation 2.47 is used to find the components of the vector $\mathbf{u_p}$, which is the projection of the velocity vector on the perpendicular vector:

$$\text{proj}_{\mathbf{r_p}}\mathbf{u} = \left(\frac{\mathbf{u}\cdot\mathbf{r_p}}{\left|\mathbf{r_p}\right|^2}\right)\mathbf{r_p} \qquad (2.47)$$

The components of the Mach number of the vortex can then be calculated, and the length of the vector $\mathbf{r_p}$ is recalculated using the Prandtl-Glauert correction:

$$r_p' = \frac{r_p}{\sqrt{1-M_p{}^2}} \qquad (2.48)$$

This is seen in *Figure 2.11*, where the vortex is stretched along $\mathbf{r_p}$ to a new position. New values for $\mathbf{r}_1'$, $\mathbf{r}_2'$, r_1', and r_2' can be calculated, and the Biot-Savart Law becomes

$$\mathbf{q} = \frac{\Gamma}{4\pi}\frac{\mathbf{r}_1'\times\mathbf{r}_2'}{\left|\mathbf{r}_1'\times\mathbf{r}_2'\right|^2}\left[\frac{\mathbf{r}_0'\cdot\mathbf{r}_1'}{r_1'} - \frac{\mathbf{r}_0'\cdot\mathbf{r}_2'}{r_2'}\right] \qquad (2.49)$$

Equation 2.49 can now be used to solve for induced velocities in a compressible flow. The translation of the vortex to a new position is only for the purposes of calculating the induced velocity; the vortex is not physically moved during this process.

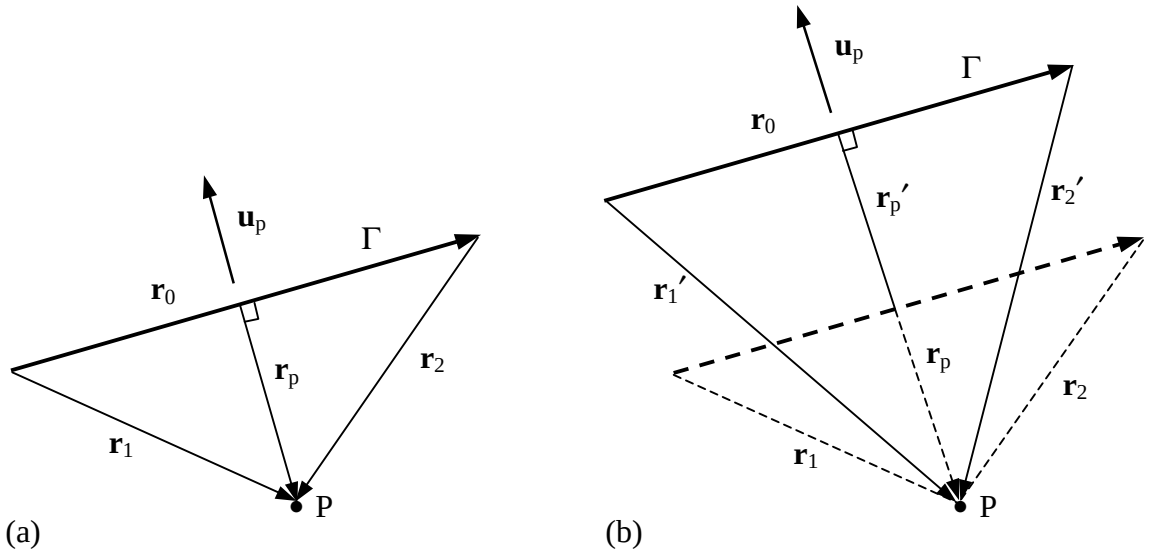


Figure 2.11: Stretching effect of the distance between the vortex and control point based on the Prandtl-Glauert correction: (a) actual distance; (b) stretched distance

Chapter 3

FREE WAKE METHOD

3.1 Overview

The equations developed in the last chapter to calculate the velocity induced at a point by a vortex can be used in a free wake method to determine the lift on the blades and the distortion of the wake. First, the general methodology of the free wake method for rotating blades will be discussed in this chapter. Then, the implementation of the method into a computer program will be described.

3.2 Free Wake Procedure

The free wake method is a method in which the wake is generated behind the blade at each time step and allowed to freely deform. The blade is discretized into panels, and a ring vortex is placed at each panel. As the blade advances, the trailing edge vortices on the wing are shed into the wake, and the wake is allowed to freely deform. Each ring vortex on the blade and in the wake induces a velocity on all the other vortices in the domain, which is calculated using the Biot-Savart Law. The Prandtl-Glauert compressibility correction is applied to the Biot-Savart Law when considering vortices on

the blade. The induced velocities on the blades are used to calculate the circulation and lift on each panel, and the induced velocities in the wake are used to deform the wake.

3.2.1 Discretization of the Blade and Wake

The blade is modeled by a thin wing with no thickness and no camber and is situated such that its feathering axis lies along the quarter-chord line. The blade is divided into panels, and a ring vortex is placed in each panel. Each vortex is placed at the panel's quarter-chord line so that the two-dimensional Kutta condition is satisfied along the chord [18]. *Figure 3.1* shows the ring vortices on the blade, with the circulation of each vortex being defined as positive in the clockwise direction. A collocation point, where the properties of the panel are stored, is also defined at the center of each panel's three-quarter-chord line.

Before the blade starts to rotate, there are no free wake elements, but the aft segments of the vortex rings along the trailing edge of the blade lie in the wake. The end points of these vortex segments make up one set of corner points for the first row of wake vortices. As the blade advances during the first time step, the new endpoints of these aft segments make up the second set of corner points, and the first row of wake elements is created and oriented at the same angle of attack as the blade. This procedure is repeated during each subsequent time step, and the wake grows with time as shown in *Figure 3.2*. The strength of the most recently created wake element is set equal to the strength of the trailing edge vortex from the previous time step. In essence, the trailing edge vortex is shed into the wake as the blade advances from one position to the next. Once a wake

vortex is created, its strength remains unchanged, according to the Helmholtz theorem [18]. Since the wake elements cannot carry aerodynamic loads, they move only with the local velocity induced by other vortices on the blades and in the wake.

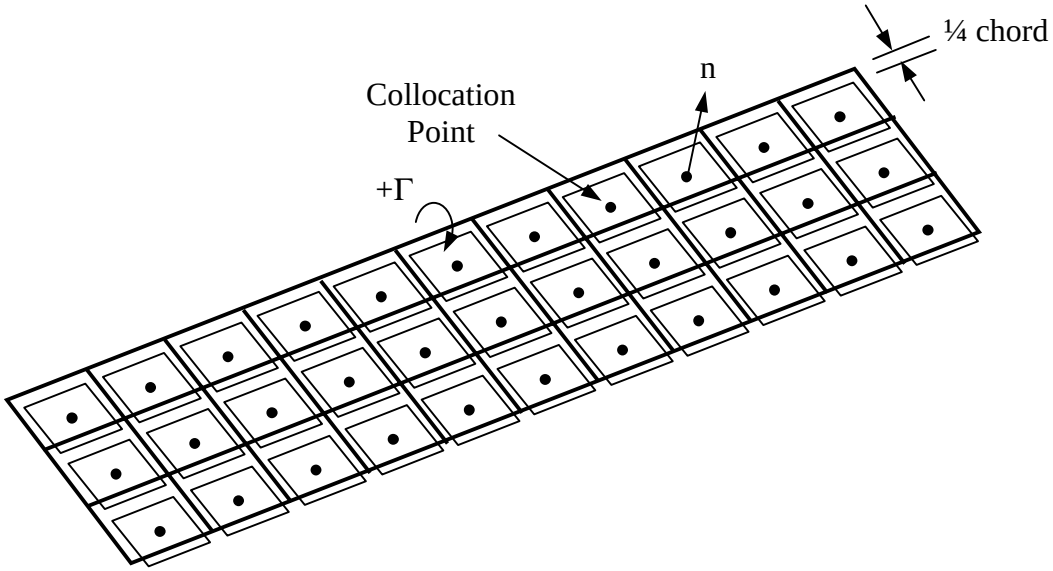


Figure 3.1: Vortex ring model

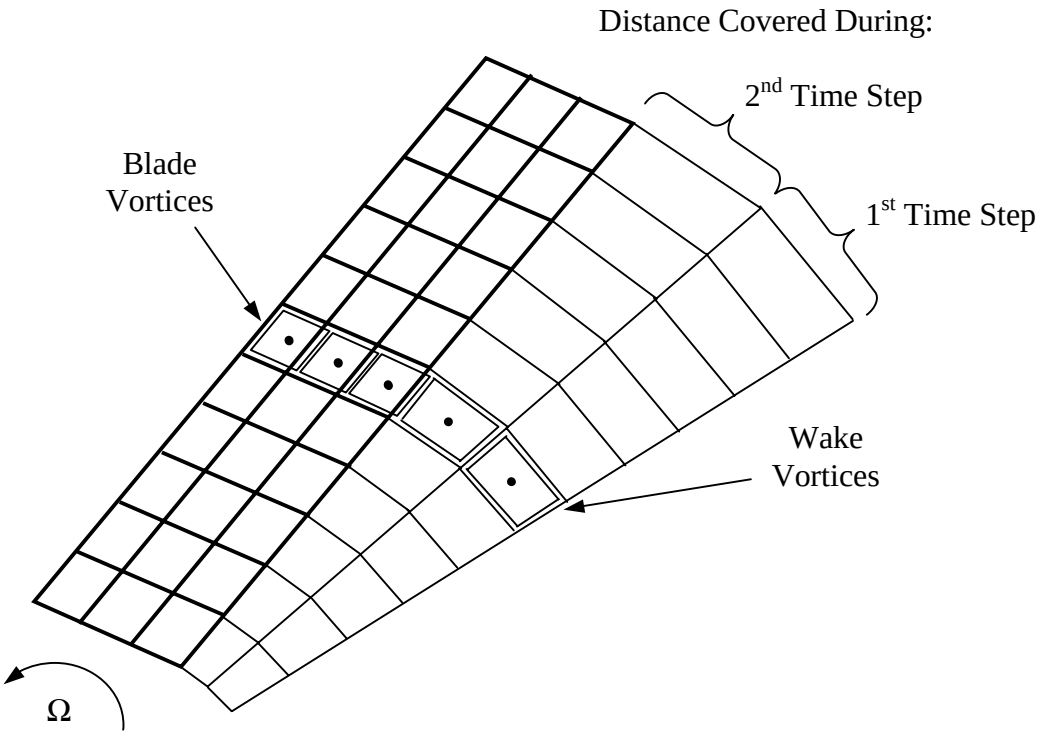


Figure 3.2: Wake shedding procedure

3.2.2 Determine the Lift on the Blades

The strengths of the vortices on the blade can be solved for at each time step by applying the boundary condition that there is no flow normal to the blade. The velocity of each ring vortex on the blade is comprised of the free stream velocity of the blade, the velocity induced by other vortices on the blade, and the velocity induced by the vortices in the wake:

$$Q_{nK} = Q_{\substack{\text{normal velocity}\\ \text{induced by blade}\\ \text{vortices}}} + Q_{\substack{\text{normal velocity}\\ \text{induced by wake}\\ \text{vortices}}} + Q_{\substack{\text{normal component}\\ \text{of free stream}\\ \text{velocity}}} = 0 \qquad (3.1)$$

Since the strength and position of the wake elements are known, the induced velocities due to the wake can be calculated at each time step using the Biot-Savart Law. These velocities, as well as the free stream velocity of the blade, which is also known, can be transferred to the right hand side of Equation 3.1. It remains then to calculate the normal velocity induced by the blade vortices as a function of the strength of each vortex and then to solve for those strengths.

Since Equation 3.1 is used to solve for the strengths of the vortices on the blade, it is necessary to incorporate these strengths into the equation. This is done in the only unknown quantity, the velocity induced by the vortices on the blade. At each blade panel, the velocity induced by every panel is calculated using the Biot-Savart Law. The Biot-Savart Law can be rewritten as

$$\begin{aligned}\mathbf{q} &= \left(\frac{1}{4\pi} \frac{\mathbf{r}_1 \times \mathbf{r}_2}{\left|\mathbf{r}_1 \times \mathbf{r}_2\right|^2} \left[\frac{\mathbf{r}_0 \cdot \mathbf{r}_1}{r_1} - \frac{\mathbf{r}_0 \cdot \mathbf{r}_2}{r_2} \right] \right) (\Gamma) \\ &= a_{KL}\Gamma \end{aligned} \qquad (3.2)$$

where a_{KL} is an influence coefficient accounting for the effects of the L^{th} panel on the K^{th} panel. Equation 3.2 can be summarized in matrix form for all the panels on the blades:

$$Q_{nB} = \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1L} \\ a_{21} & a_{22} & \cdots & a_{2L} \\ a_{31} & a_{32} & \cdots & a_{3L} \\ \vdots & \vdots & \ddots & \vdots \\ a_{K1} & a_{K2} & \cdots & a_{KL} \end{pmatrix} \begin{pmatrix} \Gamma_1 \\ \Gamma_2 \\ \Gamma_3 \\ \vdots \\ \Gamma_L \end{pmatrix} \qquad (\ 3.3\ )$$

The matrix is a square matrix of order *m*, where *m* is the total number of panels on all the blades.

The known quantities in Equation 3.1 can be grouped together and transferred to the right hand side of the equation. The normal velocity induced by vortices in the wake can be calculated using the Biot-Savart Law, since the strength and position of each wake vortex is known at every time step. Summing the effects of all the wake vortices on one blade panel *K*, the normal component of the induced velocity is

$$Q_{nWK} = \sum_{i=1}^{W} \mathbf{q}_{iK} \cdot \mathbf{n}_K = [u_W, v_W, w_w] \cdot \mathbf{n}_K \qquad (\ 3.4\ )$$

where *i* covers all *W* wake vortices. The normal component of the free stream velocity of each panel is also known, since the motion of the blades is predefined:

$$Q_{nFSK} = [U(t), V(t), W(t)]_K \cdot \mathbf{n}_K \qquad (\ 3.5\ )$$

Therefore, transferring the known quantities to the right hand side yields

$$RHS_K = -[U(t) + u_W, V(t) + v_W, W(t) + w_W]_K \cdot \mathbf{n}_K \qquad (\ 3.6\ )$$

Equation 3.1 can then be rearranged and solved for the unknown vortex strengths:

$$\begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1m} \\ a_{21} & a_{22} & \cdots & a_{2m} \\ a_{31} & a_{32} & \cdots & a_{3m} \\ \vdots & \vdots & \ddots & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mm} \end{pmatrix} \begin{pmatrix} \Gamma_1 \\ \Gamma_2 \\ \Gamma_3 \\ \vdots \\ \Gamma_m \end{pmatrix} = \begin{pmatrix} RHS_1 \\ RHS_2 \\ RHS_3 \\ \vdots \\ RHS_m \end{pmatrix} \qquad (\ 3.7\ )$$

Once the circulation of each vortex ring on the blade is known, the lift per unit span of each panel can easily be calculated using the Kutta-Joukowski theorem,

$$l_K = \rho U_K \Gamma_K \qquad (\ 3.8\ )$$

where U_K is the free stream velocity of the panel and Γ_K is the circulation of the panel [3]. For the leading edge panels, Γ_K is just equal to the circulation of the panel, but for all other panels, Γ_K is equal to the difference between the circulation of that panel and the circulation of the panel directly forward of that panel [18]. The actual lift on each panel is

$$L_K = l_K \Delta b_K \qquad (\ 3.9\ )$$

where Δb_K is the spanwise dimension of the panel. Since the blade is modeled by a flat plate, the pressure difference on the panel is simply

$$\Delta p_K = \frac{L_K}{\Delta S_K} \qquad (\ 3.10\ )$$

where ΔS_K is the area of the panel. The sectional lift l_j is found by summing the sectional lift of all the panels at a given spanwise location *j*, and the sectional lift coefficient is

$$C_{l_j} = \frac{l_j}{\frac{1}{2}\rho U_\infty^2 c} \qquad (\ 3.11\ )$$

The thrust coefficient is

$$C_T = \frac{T}{\rho \pi R^2 U_\infty^2} \qquad (3.12)$$

where the thrust T is equal to the total lift produced by all the blades.

3.2.3 Calculate Wake Roll-Up

The vortices in the wake cannot carry aerodynamic loads, so they move only with a local velocity induced by the other vortices in the wake and on the blades. The components of the local induced velocity are calculated by summing the components of the velocity induced by all of the other vortices. The velocities are calculated at the corner points of the wake vortices, which are then allowed to translate in space:

$$(\Delta x, \Delta y, \Delta z) = (u, v, w)_w \Delta t \qquad (3.13)$$

This is what allows the wake to freely deform by rolling-up, contracting, and convecting downward below the rotor.

3.3 Implementation of the Free Wake Procedure in the Code

The free wake method was implemented into a Fortran program. *Figure 3.3* is a flow chart showing the structure of the program. Subroutines are shown in capital letters in brackets, and flow charts for the important subroutines are presented in the Appendix. The major components of the program will be described in detail in the order in which they are performed.

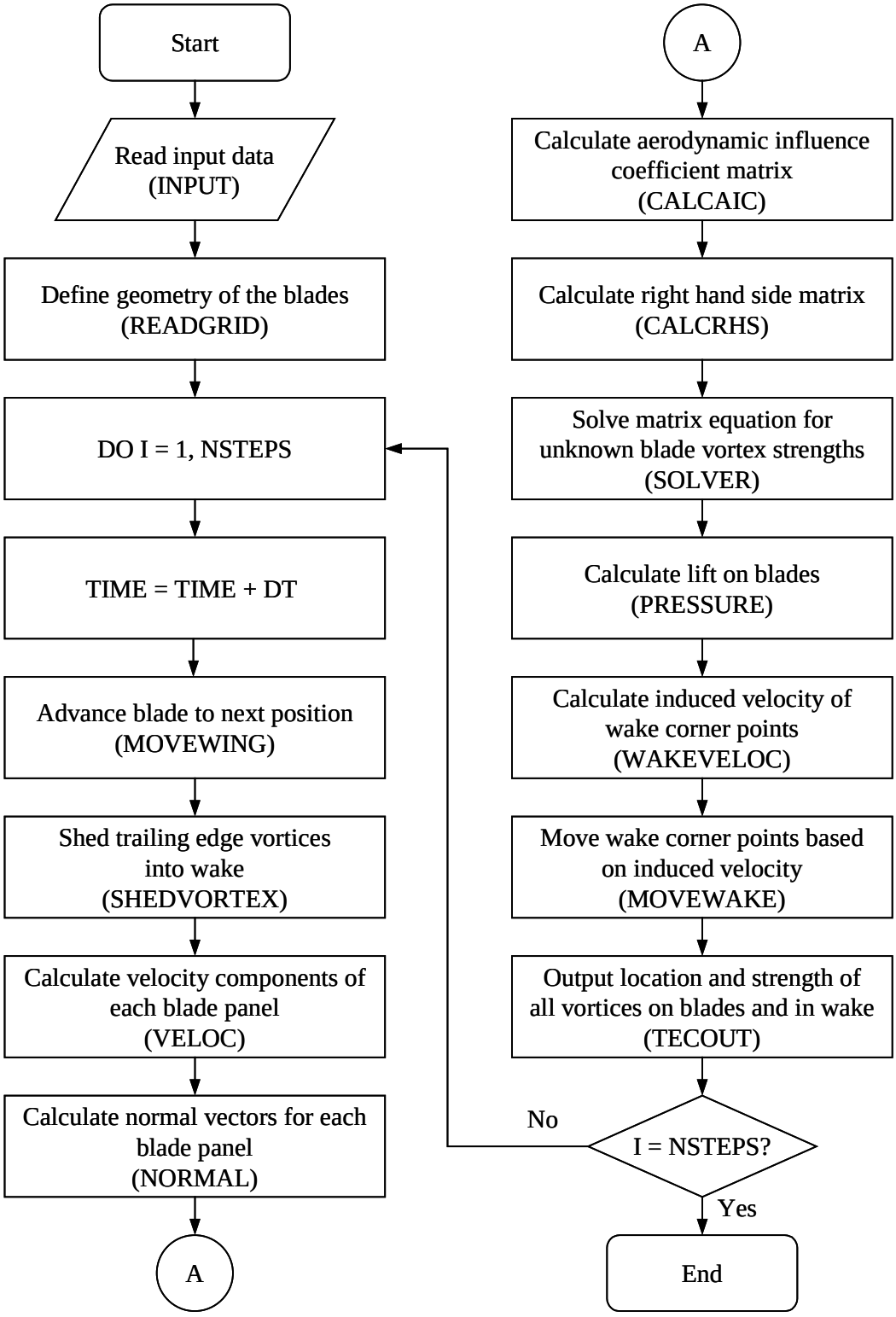


Figure 3.3: Computer program flowchart

The first step in the program is to input the parameters that define the problem in the INPUT subroutine. These include the time step size, number of time steps, number of blades, chord and span of the blades, pitch and coning angle of the blades, number of elements in the chordwise and spanwise directions, vortex core radius, rotational speed of the rotors, and velocity of the helicopter fuselage. All units are defined in the metric system. A sample input file is given in the Appendix. Next, the geometry of the blades is defined in the READGRID subroutine. The position of one blade is defined and panels are applied to that blade. Cosine spacing is used in the chordwise direction so that smaller panels are used near the leading and trailing edges and larger panels are used in the middle of the blade. Similarly, semi-cosine spacing is used in the spanwise direction from the root to the tip so that smaller panels are used near the tip. This type of spacing allows for smaller panels to be used in the areas of interest but saves computational time since larger panels can be used in areas that are less important. After the panels have been assigned, the pitch and precone angles are applied to the blade. The ring vortices are applied at the quarter-chord line of each panel, and the collocation points are applied at the center of the three-quarter-chord line of each panel. Then, the remaining blades are created by copying the first blade, and the blades are evenly spaced around the rotor disk using the blade quarter-chord line, or the feathering axis, as a reference.

After the initial geometry of the blades is defined, the time loop is started that runs over the length of the simulation. The first step in the loop is to advance each blade to the current time step position using the MOVEWING subroutine. First, the location of the center of the rotor disk is subtracted from the coordinates of the panels so that the

blade rotation and rectilinear motion of the helicopter can be considered separately. A transformation matrix is constructed that accounts for the rotation of the blades, and the old blade locations are multiplied by the transformation matrix to produce the new blade locations due to rotation. The center of the rotor disk is then translated based on the rectilinear motion of the helicopter, and the components of the location of the center point due to translation and of the locations of the panels due to rotation are added to produce the new locations of the panels. After the blades have been moved, a new row of vortices is shed into the wake using the SHEDVORTEX subroutine. The new row of vortices occupies the space previously occupied by the trailing-edge vortices of the blades before the blades were moved. They are oriented such that they are at the same angle of attack as the blade when shed. The strengths of the vortices in the first row of the wake are set equal to the strengths of the trailing edge blade vortices from the previous time step.

Next, with the blades in their current positions, the velocity of each panel is calculated in the VELOC subroutine. The velocity is calculated at the collocation point and is the sum of the rotating velocity of the blades and the translation velocity of the helicopter. The normal vectors are then found for all blade panels in the NORMAL subroutine. The normal vectors can easily be calculated using Equation 3.14,

$$\mathbf{n}_K = \frac{\mathbf{A}\times\mathbf{B}}{|\mathbf{A}\times\mathbf{B}|} \qquad (3.14)$$

where **A** and **B** are vectors defined by opposite corner points of the panel [18].

With the velocity and normal vectors calculated for each panel, it is possible to start the process of solving for the strength of the vortex on each panel. The first step is

to calculate the aerodynamic influence coefficient matrix using the CALCAIC subroutine. Recall that the influence matrix is a square matrix of order *m*, where *m* is the total number of panels on all the blades. Each coefficient in the matrix represents the effects of a unit-strength vortex ring at a panel *L* on the panel *K*. A series of nested loops is used to calculate the coefficients. The first series of loops iterates over the blades, the chordwise panels on each blade, and the spanwise panels on each blade. A counter then assigns the value *K* to the current panel. With the panel fixed, a second series of loops iterates over the blades, the chordwise panels on each blade, and the spanwise panels on each blade. Another counter assigns the value *L* to the second panel. The subroutine VORTEX is then used to calculate the induced velocity at *K* due to a vortex of unit strength at *L*. The subroutine is applied to each side of the ring vortex at *L* and the components of the velocities are summed. The components are multiplied by the components of the normal vector at *K* and summed to produce one influence coefficient at *K* due to *L*.

The second step in calculating the strengths of the vortices on the blades is to determine the right hand side of the matrix equation using the CALCRHS subroutine. Recall that the right hand side is made up of the normal component of the free stream velocity and the normal component of the velocity induced by the wake. First, a series of loops iterates over the blades, the chordwise panels on each blade, and the spanwise panels on each blade, and the current panel is assigned the value *K*. Then, the part of the right hand side due to the free stream velocity of the panel is calculated by multiplying the components of the velocity of the panel *K* by the components of the normal vector at

the panel *K* and summing. Next, the part due to the wake is calculated by looping over the number of blades, the number of time steps (where each time step represents one row of panels shed into the wake), and the panels in each row of the wake. Again, the VORTEX subroutine is used to calculate the velocity induced by each side of the ring vortex in the wake, but this time, the known strengths of the wake panels are used in the calculation to get an actual induced velocity rather than a coefficient. The components of the velocity are summed and then multiplied by the components of the normal vector at the panel *K*. Unlike in the CALCAIC subroutine, however, where the influence of each panel was considered separately, now the effects of all the panels in the wake are added to the effect of the free stream velocity to produce one total induced velocity field at the panel *K*. When the matrix is completed, it is transferred to the right hand side of the equation by subtracting it from both sides.

It is necessary at this point to describe in detail the VORTEX subroutine that was used in the calculation of the influence matrix and the right hand side matrix. The VORTEX subroutine uses the Biot-Savart Law and, when dealing with vortices on the blade, the Prandtl-Glauert compressibility correction to calculate induced velocities. The first step is to calculate the compressibility correction, if applicable. The perpendicular vector between the vortex segment and the control point of the panel *K* is found. Then the velocity and Mach number of the vortex along the perpendicular vector can be found and the length of the vector is recalculated using the Prandtl-Glauert correction.

The Biot-Savart calculation begins by calculating the components of the cross product of the vectors $\mathbf{r}_1'$ and $\mathbf{r}_2'$, which define the distance between the endpoints of the

virtual location of the vortex and the control point, and the components of the vector $\mathbf{r_0}$ that defines the direction of the vortex. If the perpendicular distance r_p' between the control point and the virtual vortex is less than the size of the core radius, then the components of the induced velocity are calculated using the solid body approximation:

$$q_i = \left(\frac{r_p'}{r_c}\right)\left(\frac{\Gamma}{2\pi r_c}\right)\left(\frac{(\mathbf{r_1'}\times\mathbf{r_2'})_i}{|\mathbf{r_1'}\times\mathbf{r_2'}|}\right) \qquad (3.15)$$

If the control point is located outside the vortex core, then the components of the induced velocity are calculated using the Biot-Savart Law:

$$q_i = \left(\frac{\Gamma}{4\pi}\right)\left(\frac{(\mathbf{r_1'}\times\mathbf{r_2'})_i}{|\mathbf{r_1'}\times\mathbf{r_2'}|^2}\right)\left(\frac{\mathbf{r_0}\cdot\mathbf{r_1'}}{r_1'} - \frac{\mathbf{r_0}\cdot\mathbf{r_2'}}{r_2'}\right) \qquad (3.16)$$

After the influence coefficient matrix and the right hand side matrix have been established, the unknown vortex strengths are solved for in the SOLVER subroutine. The SOLVER subroutine uses the LINPACK library to solve the matrix equation by *LU* decomposition. LINPACK is a freely available library of Fortran subroutines designed to solve linear systems [24].

Next, the lift on the blades is calculated in the PRESSURE subroutine. Since the strength of each ring vortex on the blades is known, the lift on each panel can be calculated using Equation 3.8. First, a loop iterates over all the blades and the total lift on each blade is set to zero. The spanwise panel locations on each blade are looped over and the spanwise lift coefficient at each location is set to zero. Then the chordwise panels are looped over at each spanwise location. The free stream velocity of each panel is determined, and the sectional lift and total lift of each panel are calculated. The sectional

lift of the panel is added to the cumulative sectional lift of the current spanwise location, and, after the chordwise panel loop is completed, the sectional lift coefficient can be calculated for that spanwise location. Similarly, the total lift for the panel is added to the cumulative lift of the blade, and after the chordwise and spanwise loops are completed, the total lift on the blade is known. Finally, the pressure and pressure coefficient are calculated for each panel. Since the blade is represented by a flat plate, the pressure of each panel represents the difference between the pressure on the upper and lower surfaces of an airfoil. After the loop over the blades is completed, the thrust coefficient of the blade configuration can be calculated.

Next, the wake is allowed to freely deform in the WAKEVELOC and MOVEWAKE subroutines. First, the induced velocity due to the vortices on the blades and in the wake is calculated at the corner points of each wake element. A series of loops iterates over all the blades, the number of time steps (where each time step represents one row of panels shed into the wake), and the panels in each row of the wake. Then, a second series of loops iterates over all the blades and the panels on each blade, and the VORTEX subroutine is used to calculate the induced velocity at the current wake corner point due to all the vortices on the blades. The Prandtl-Glauert compressibility correction is not used in the VORTEX subroutine when calculating the induced effects of one wake vortex on another wake vortex. When the second set of loops is completed, a third series of loops iterates over all the blades and the panels in the wake, and the induced velocity at the current wake point due to all the vortices in the wake is calculated. The induced velocities from the blades and the wake are summed to produce one induced velocity at

each corner point of the wake panels. Then, that induced velocity is used in the MOVEWAKE subroutine, where all the wake corner points are looped over and allowed to move based on their velocity. The first row of wake panels trailing off each blade is not allowed to move to ensure that the flow leaves tangent to the blade and that the Kutta condition is preserved.

Finally, the strength and position of the vortices on the blades and in the wake are outputted to files in the TECOUT subroutine. The files are formatted to be read into Tecplot where they can be viewed and analyzed. The data output is the last step in the time loop, and the program is terminated after it loops over the total number of time steps.

Chapter 4

RESULTS AND DISCUSSION

4.1 Overview

Two experimental cases were chosen as test cases to validate the free wake code. The experiments were conducted by Caradonna and Tung to provide data to be used in the validation of future rotor performance codes [5]. Blade pressure measurements were made for a hovering two-bladed rotor over a wide range of tip Mach numbers from the incompressible to transonic flow regimes. The two cases chosen to validate this code represented an incompressible flow case (M_{tip} = 0.44) and a compressible flow case (M_{tip} = 0.88). The compressible hover case is an unrealistic case from the standpoint of normal helicopter operations, but it is useful as a test case because it introduces flow characteristics normally only seen in high-speed forward flight. After the code was validated, a number of other cases were run to demonstrate the versatility of the code.

The experiments were run in the Army Aeromechanics Laboratory hover facility, a large chamber specially designed to eliminate recirculation. *Figure 4.1* shows the set-up of the rotor blades in the test facility. The blades were NACA 0012 airfoils with no twist or taper and a half degree of precone. Each blade had a radius of 3.75 ft (1.143 m) and an aspect ratio of 6. The root cutout was approximately equal to one chord. Each blade was fitted with pressure taps at five spanwise locations, with more data being taken

near the tip and near the leading edge. Both validation cases presented here use a rotor with 8° of collective pitch, with the first rotating at 1250 rpm, which corresponds to a tip Mach number of 0.44, and the second rotating at 2500 rpm, which corresponds to a tip Mach number of 0.88.

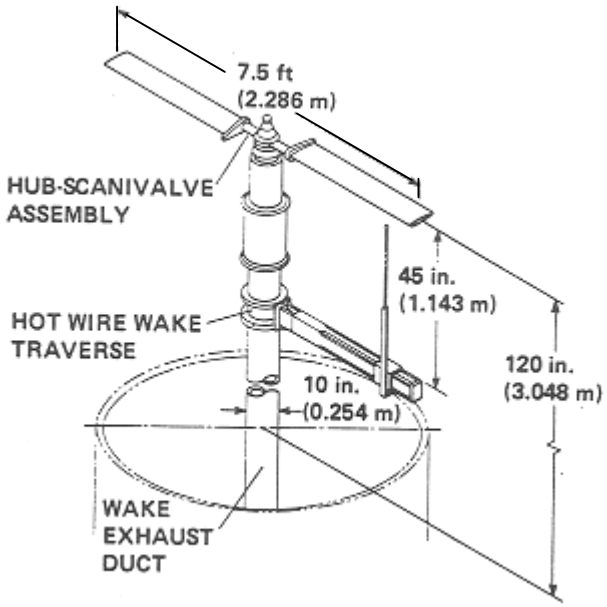


Figure 4.1: Configuration of the experimental rotor [5]

4.2 Code Validation

The computational model of the rotor blades used in both cases was the same. The blades were modeled by a flat plate with 8° of collective pitch and 0.5° of precone. The dimensions of the blades were set exactly equal to the dimensions of the blades used

in the experiment. Each blade had eight panels in the chordwise direction and ten panels in the spanwise direction. Cosine spacing was used in the chordwise direction such that the smallest panels were placed near the leading edge and trailing edge of the blade. Likewise, semi-cosine spacing was used in the spanwise direction from the root such that the smallest panels were near the blade tip. *Figure 4.2* shows the layout of the panels on one blade. Recall that each ring vortex is placed at the quarter-chord line of its corresponding panel. The time step size was set in each case such that the blades advanced 6° during each time step. The number of panels and the time step size were based on simulations run by Katz and Maskew, who closely matched the data from the same incompressible test case used here with their own free wake code [17]. The vortex core radius was set equal to ten percent of the spanwise length of the smallest panels at the tip to ensure proper roll-up of the tip vortex. The number of panels, the panel spacing, and the core size were all changed to see what effect they had on the results. All cases were run on the same machine with one 800 MHz processor and up to 1 GB of RAM available.

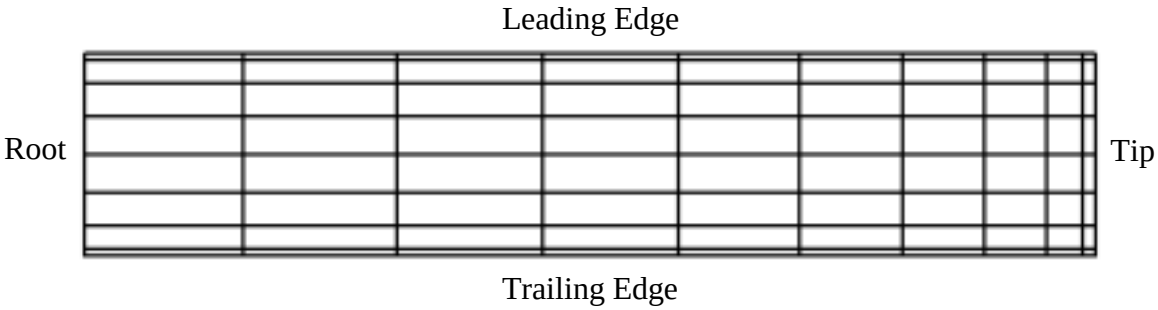


Figure 4.2: Grid spacing on blade

4.2.1 Case 1: Incompressible Hover

The first validation case was a case where the flow could be considered incompressible everywhere, since the Mach number at the tip was only 0.44. Because the flow is incompressible, a traditional Biot-Savart method can be used to evaluate it, but the modified code including the Prandtl-Glauert correction should produce accurate results as well. Therefore, the first case was run using both the incompressible and the compressible versions of the code, and the results for both are presented and compared. Each case ran for eight revolutions, or 480 time steps. In actual time, eight revolutions represents 0.384 s. Convergence was declared after eight revolutions based on the fact that variations in the thrust coefficient leveled off and that there were no significant differences in the pressure readings from the one blade to the other. The incompressible code ran in 33 hrs, while the compressible code ran in 43.67 hrs, with the difference coming from the extra calculations required by the Prandtl-Glauert correction. *Figure 4.3* shows the thrust coefficient over time for each code, as well as the experimentally determined thrust coefficient of 0.00459. The large increase in thrust at the beginning is due to the effects of the strong starting vortex shed when the blades start from rest. As the starting vortex is convected away, its effects diminish and the thrust coefficient eventually decreases to a steady state value.

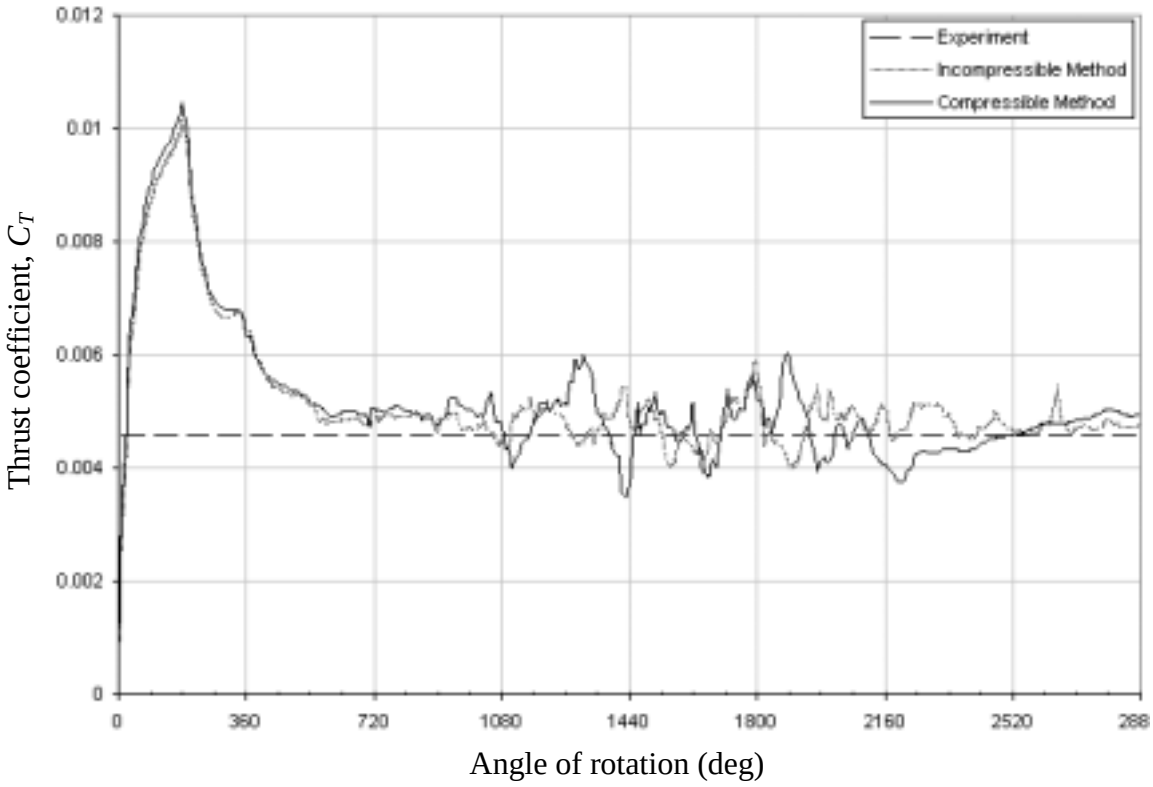


Figure 4.3: Thrust coefficient over time

The most effective way to validate the accuracy of the code is to compare the calculated lift coefficient and pressure coefficients with experimental data. *Figure 4.4* shows the experimental lift coefficient and the lift coefficient computed by each version of the code over the span of the blade. Both versions of the code produce very similar results, as expected, and these results closely match the experimental data. *Figure 4.5* shows a series of pressure coefficient plots at different spanwise locations. Since the blade is modeled by a flat plate, the computed pressure represents the difference between the pressure on the upper surface and the lower surface of an airfoil. Therefore, a delta pressure coefficient is actually plotted for the each version of the code as well as for the experimental data. The vortex lattice method for a flat plate is singular at the leading edge, which is why the pressure the coefficient increases so rapidly there [18]. This singular can be avoided by using a more accurate panel method. Again, both versions of the code produce nearly identical results, although these results differ slightly from the experimental results. The difference is most likely due to the resolution of both the computational data and the experimental data. It is likely that more panels are required near the leading edge in the computational method to fully capture the flow, but more accurate experimental data could also be found using more pressure taps near the leading edge, especially in the inboard region.

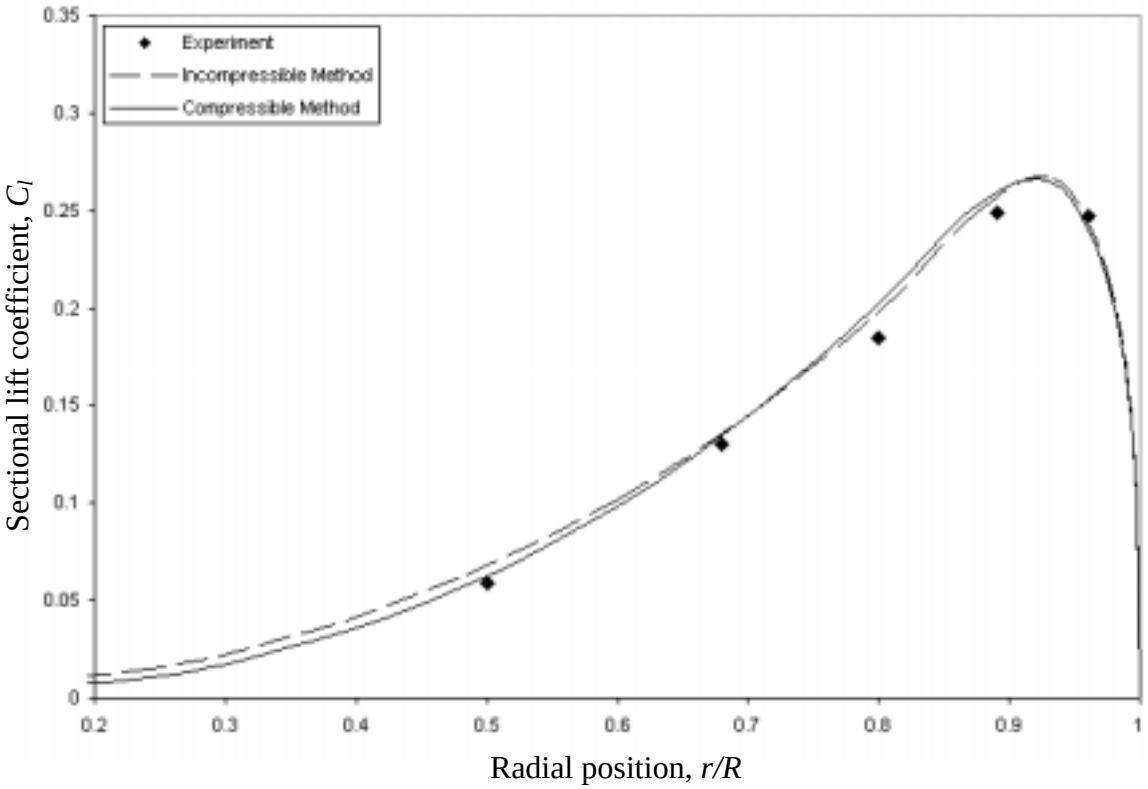


Figure 4.4: Spanwise lift coefficient (normalized by tip speed)

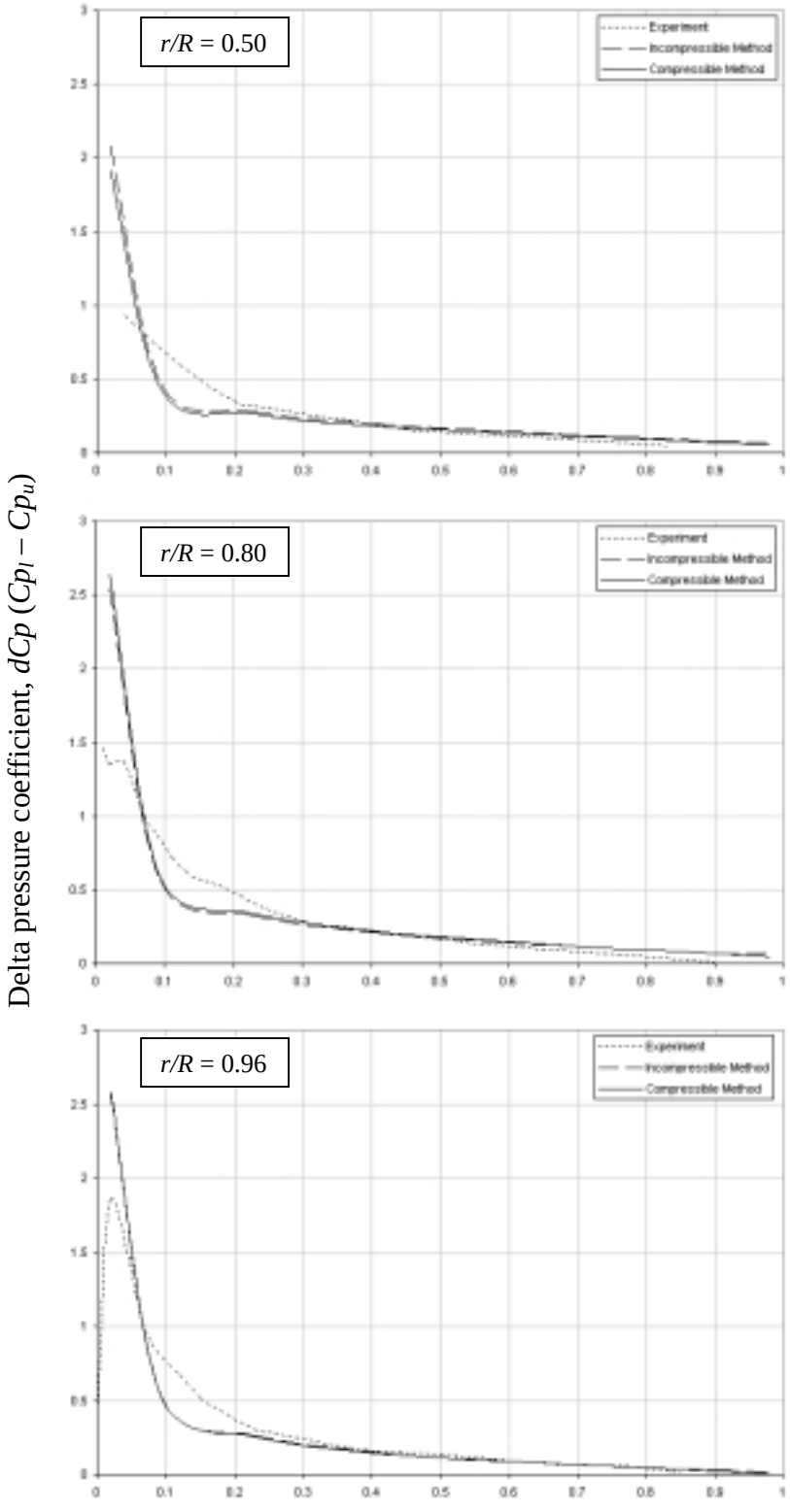


Figure 4.5: Chordwise pressure coefficient at different radial sections

The results from both the incompressible and the compressible versions of the code are almost identical in all cases. This is expected since the compressibility corrections should not have a significant impact on incompressible flow. Even at the tip, where the Mach number is 0.44, the effect of the Prandtl-Glauert correction on the virtual distance between a control point and a vortex is only about 10%. But the fact that the compressible version of the code still works for incompressible flow is an important one when trying to validate it. Still, since the incompressible version ran approximately 30% faster than the compressible version while producing essentially the same results, it makes sense to use it in cases where the flow is known to be incompressible everywhere.

4.2.2 Case 2: Compressible Hover

The second validation case was one where the flow near the tips was compressible, with the tip Mach number being 0.88. This is the most interesting case, since traditional Biot-Savart techniques cannot accurately capture the physics of compressible flow. Several variations of this case were run to determine the best set-up of the code to achieve a balance between accuracy and speed.

4.2.2.1 Case 2a: Incompressible Code vs. Compressible Code

This case was run with the same set-up as Case 1, using both the incompressible and compressible versions of the code. Convergence was declared after eight revolutions (480 time steps) with the incompressible code and ten revolutions with the compressible

code (600 time steps). The incompressible code ran in 33 hrs, while the compressible code ran in 65.5 hrs. The large difference in run time is due to the extra revolutions and the extra calculations required by the Prandtl-Glauert correction. *Figure 4.6* shows the thrust coefficient over time and the experimental thrust coefficient of 0.00473.

Figure 4.7 shows the experimental lift coefficient and the lift coefficient computed by both versions of the code. It also shows the effects of applying the Prandtl-Glauert correction directly to the lift coefficient. There is more lift at the tip in this case, where the flow is compressible, but the incompressible method fails to predict this lift increase. Although the compressible version slightly overpredicts lift inboard of the tip, it does predict the increased lift at the tip. Simply applying the Prandtl-Glauert correction to the lift coefficient is seen to severely overpredict lift, especially at the tip. This agrees with the conclusions of Gennaretti and Morino that, while this technique may be useful for fixed wings, it is not valid for use with rotating blades [14].

Figure 4.8 shows a series of pressure coefficient plots at different radial locations for both the incompressible and compressible versions of the code. In the inboard region, the methods produce nearly identical results, which is expected since the flow in this region is still incompressible. Moving closer to the tip, however, the results begin to differ as the compressible method predicts more lift than the incompressible method. The compressible method, which will be used from this point forward, shows good agreement with the experiment in the inboard region, but the agreement breaks down closer to the tip, especially near the leading edge. This again is a result of the singularity of the method at the leading edge.

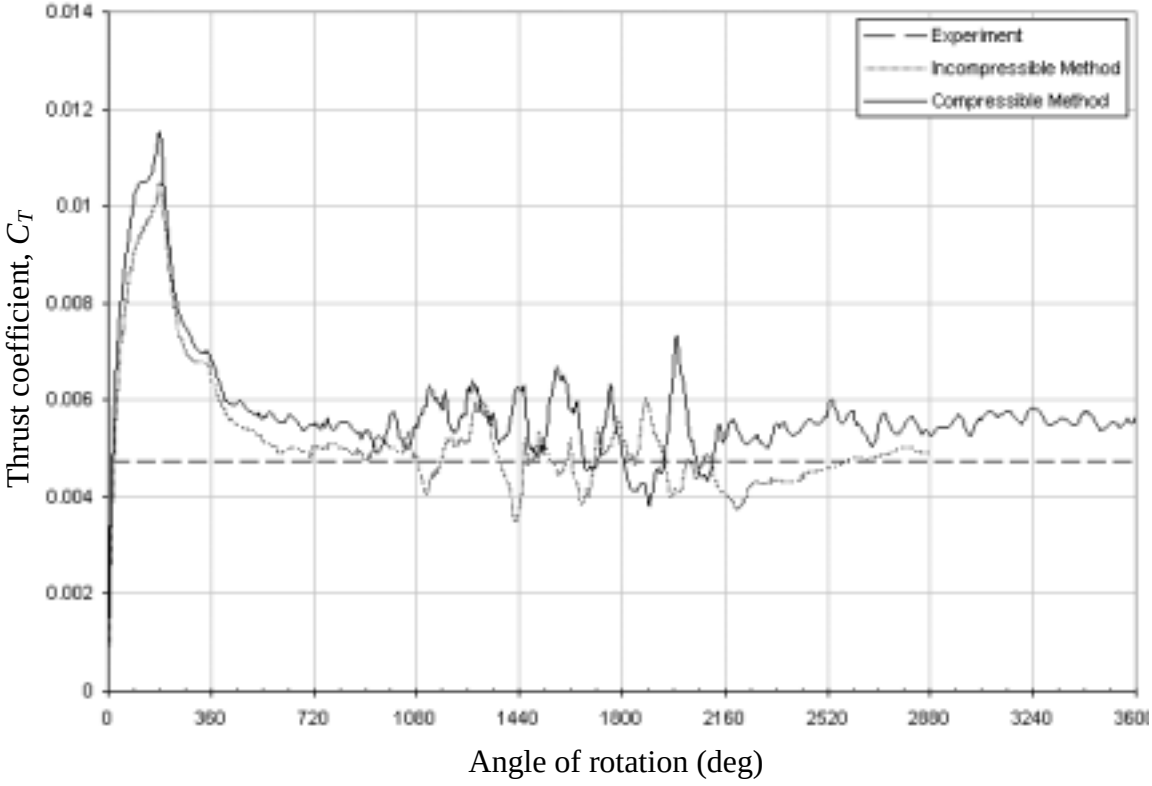


Figure 4.6: Thrust coefficient over time

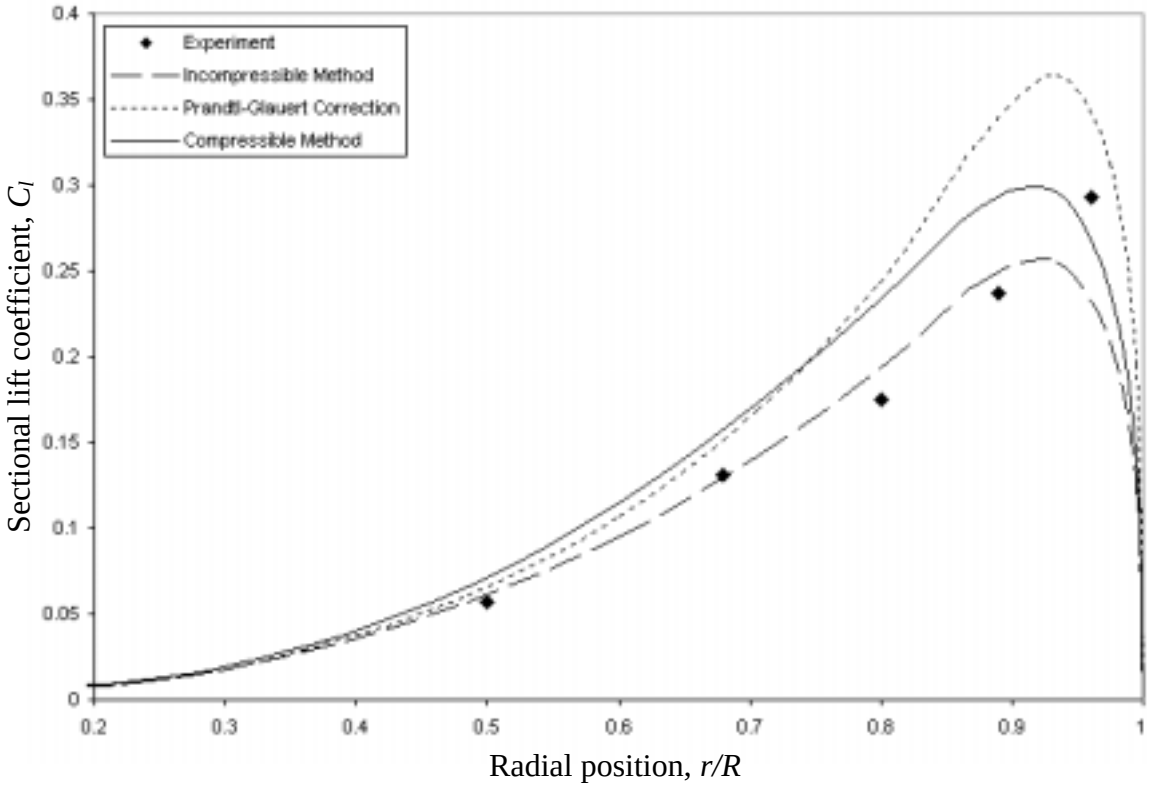


Figure 4.7: Spanwise lift coefficient (normalized by tip speed)

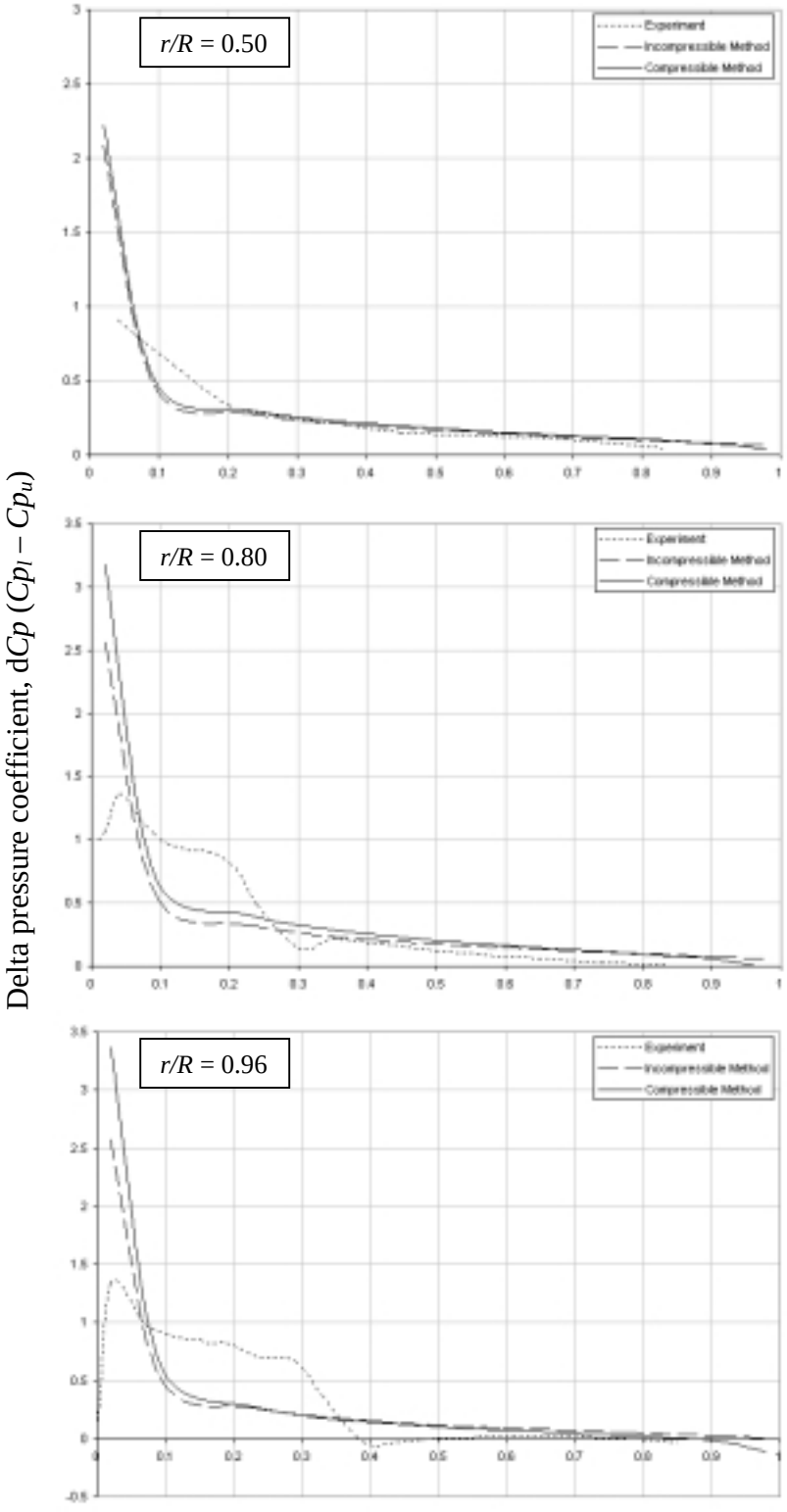


Figure 4.8: Chordwise pressure coefficient at different radial locations

4.2.2.2 Case 2b: Changing the Chordwise Panel Spacing

Another simulation was run to determine the effect of the panel spacing in the chordwise direction. The original case used full-cosine spacing, which concentrated smaller panels and the leading and trailing edges, but the modified case used semi-cosine spacing to concentrate panels at the just the leading edge. Both cases used 8 panels along the chord and ran in same amount of time. *Figure 4.9*, the spanwise lift coefficient, shows that the modified method predicts more lift at the tip than the original method and more closely matches the experimental data inboard of the tip. It is difficult to determine which method is more accurate, however, due to the sparse spacing of experimental data points. *Figure 4.10*, the chordwise pressure coefficient, shows that the modified method does a slightly better job at approaching the experimental data over the forward half-chord. Again, it is difficult to determine which spacing method is better, but because the pressure is changing much more near the leading edge than near the trailing edge, it makes sense to concentrate smaller panels there.

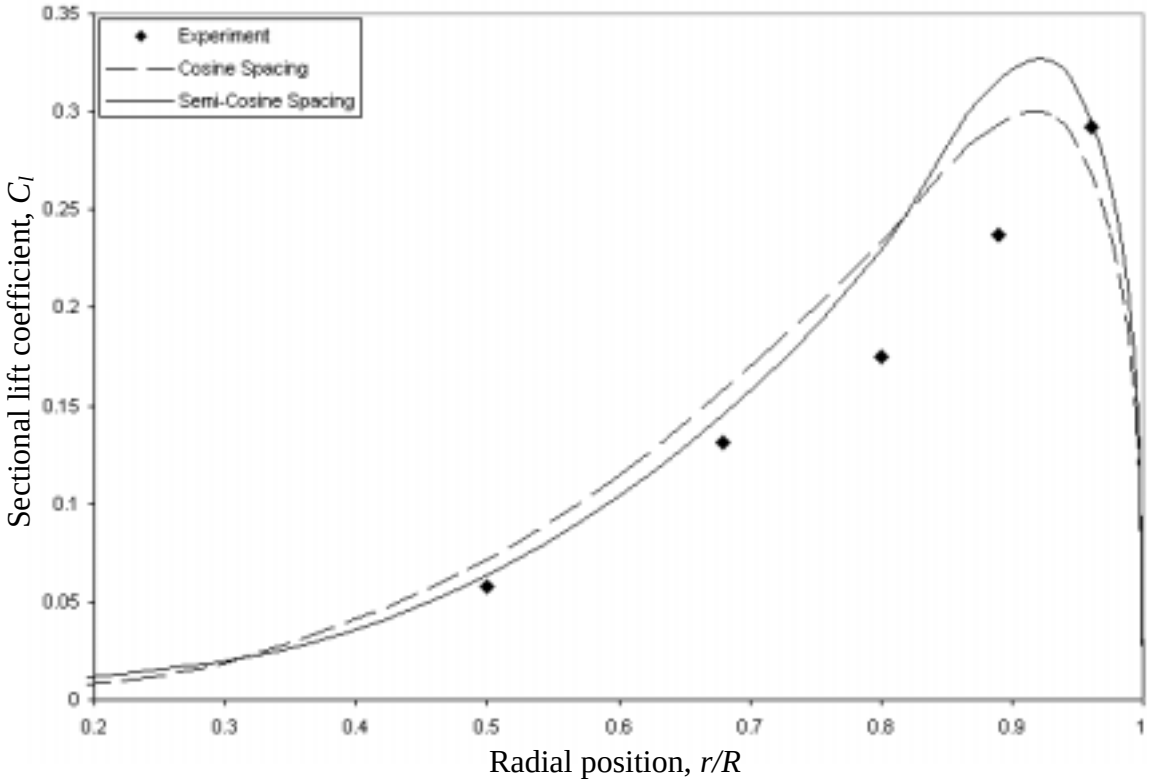


Figure 4.9: Spanwise lift coefficient (normalized by tip speed)

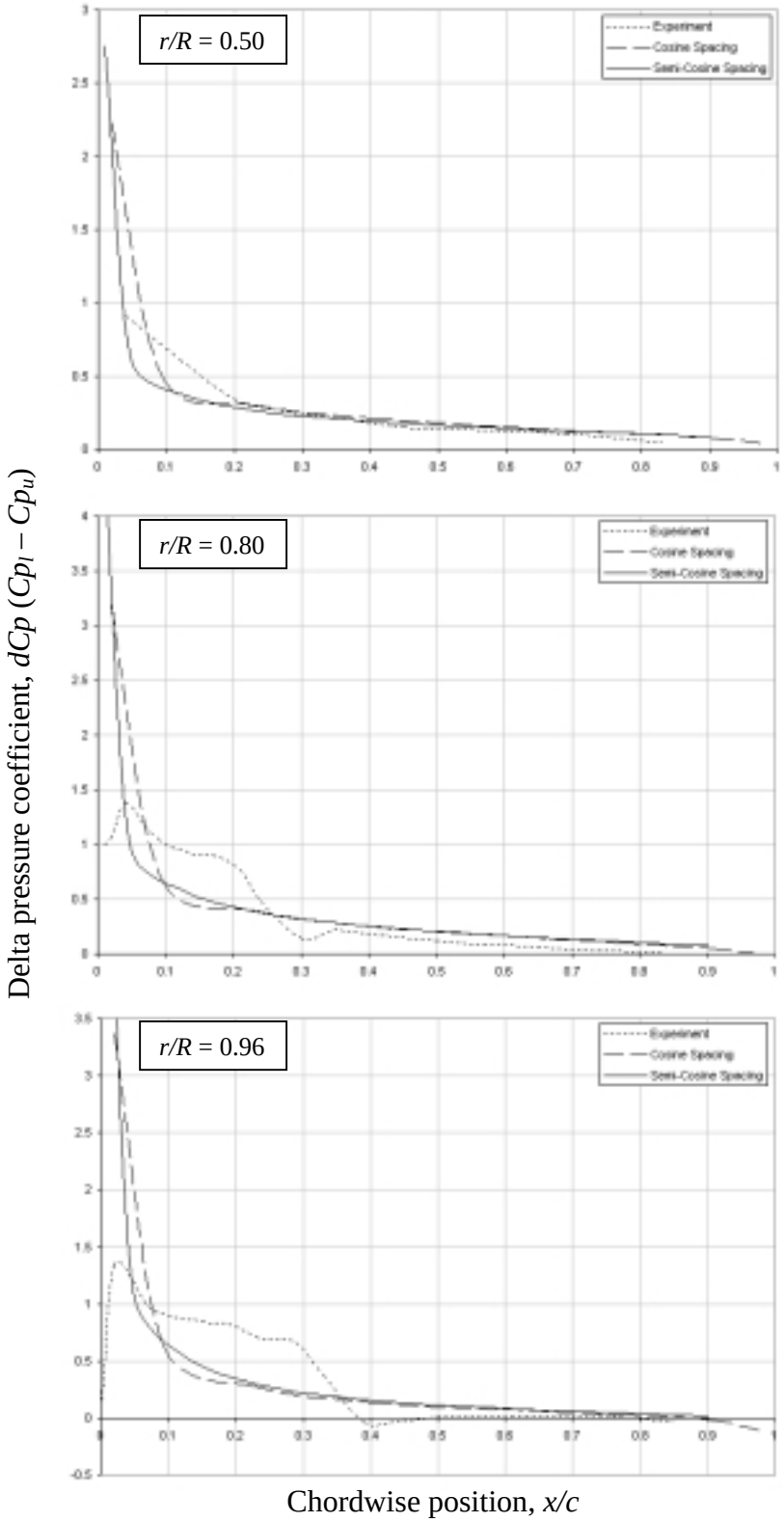


Figure 4.10: Chordwise pressure coefficient at different radial sections

4.2.2.3 Case 2c: Changing the Vortex Core Radius

Two more simulations were run in which the vortex core radius was changed. Recall that the original simulation used a core radius equal to 0.1% of the spanwise dimension of the smallest blade panel. The first modified run used a core radius equal to 0.001% of the smallest panel, and the second run used a core radius equal to 0.5% of the smallest panel. All three cases ran in the approximately same time, with only slight differences coming from the number of calculations performed inside the vortex core using the simpler solid body approximation versus the number performed outside the core using the more complicated Biot-Savart Law for each case. *Figures 4.11* and *4.12* show that the results for all three cases are essentially the same, although the case with the largest core radius predicts slightly more lift than the other two cases. Increasing the core radius to a value greater than half the size of the panels at the tip will mean that the distance along the span between the control point and the vortex ring which it defines will be within the vortex core, which could adversely affect the roll-up of the tip vortex.

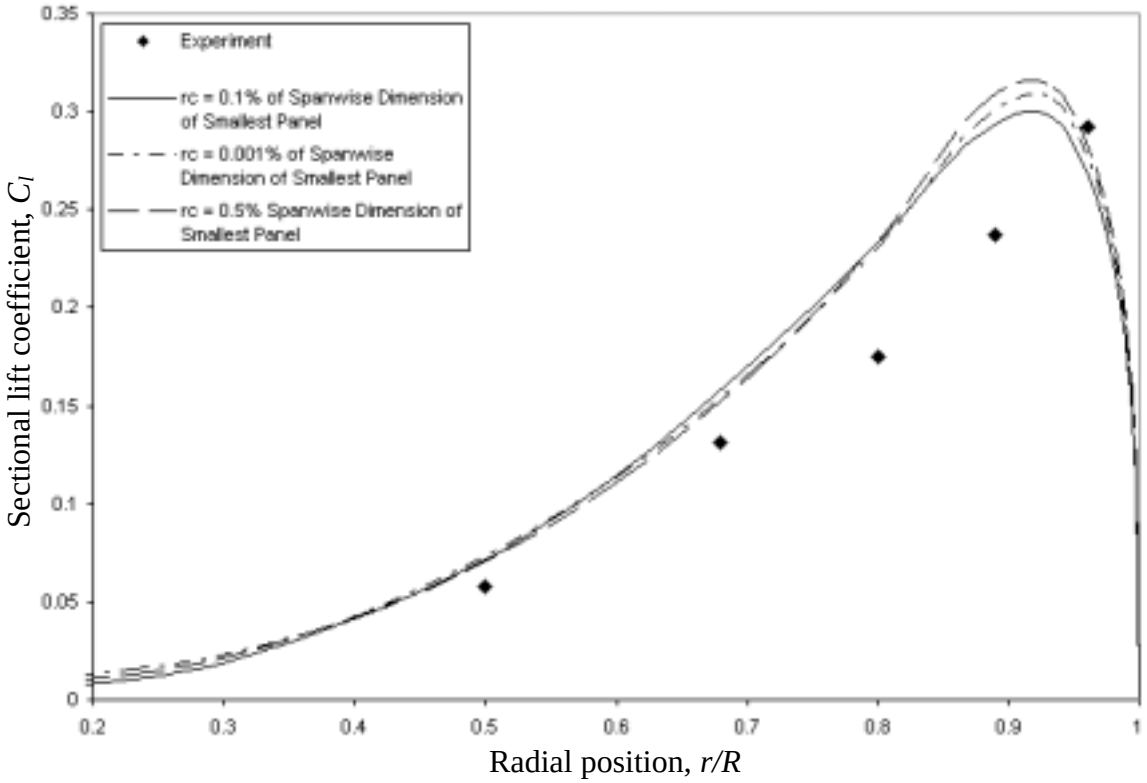


Figure 4.11: Spanwise lift coefficient (normalized by tip speed)

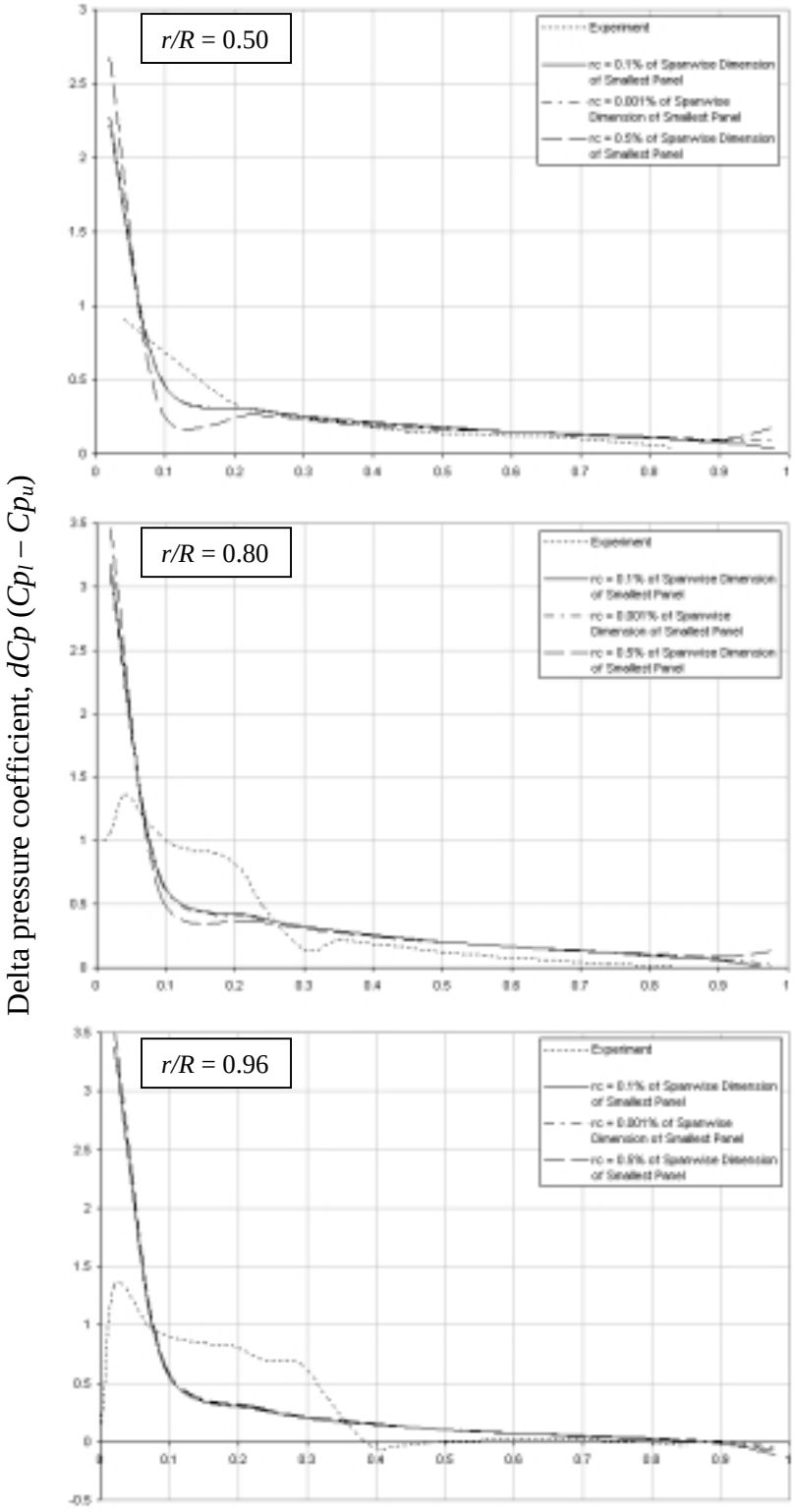


Figure 4.12: Chordwise pressure coefficient at different radial sections

4.2.2.4 Case 2d: Changing the Number of Spanwise Panels

Using semi-cosine spacing in the chordwise direction now and the original core size, two simulations were run to test the effect of changing the number of panels in the spanwise direction. The original case used 10 spanwise panels and ran in 65.5 hrs, while the first modified case used 8 spanwise panels and ran in 43 hrs and the second used 15 spanwise panels but ran in 145 hrs. *Figure 4.13*, the spanwise lift coefficient, shows that there is little difference in the spanwise lift between using 10 panels and 15 panels but that using 8 panels underpredicts the lift. Likewise, *Figure 4.14*, the chordwise pressure coefficient, shows that using more than 10 panels in the spanwise direction has almost no impact on the results but that using less than 10 panels results in a lower prediction of the pressure. Considering the large amount of extra computer time required to run with 15 panels, it is clear that there is no advantage over using 10 panels. And although using 8 panels allows the code to run in less time, the savings in time does not make up for the loss in accuracy.

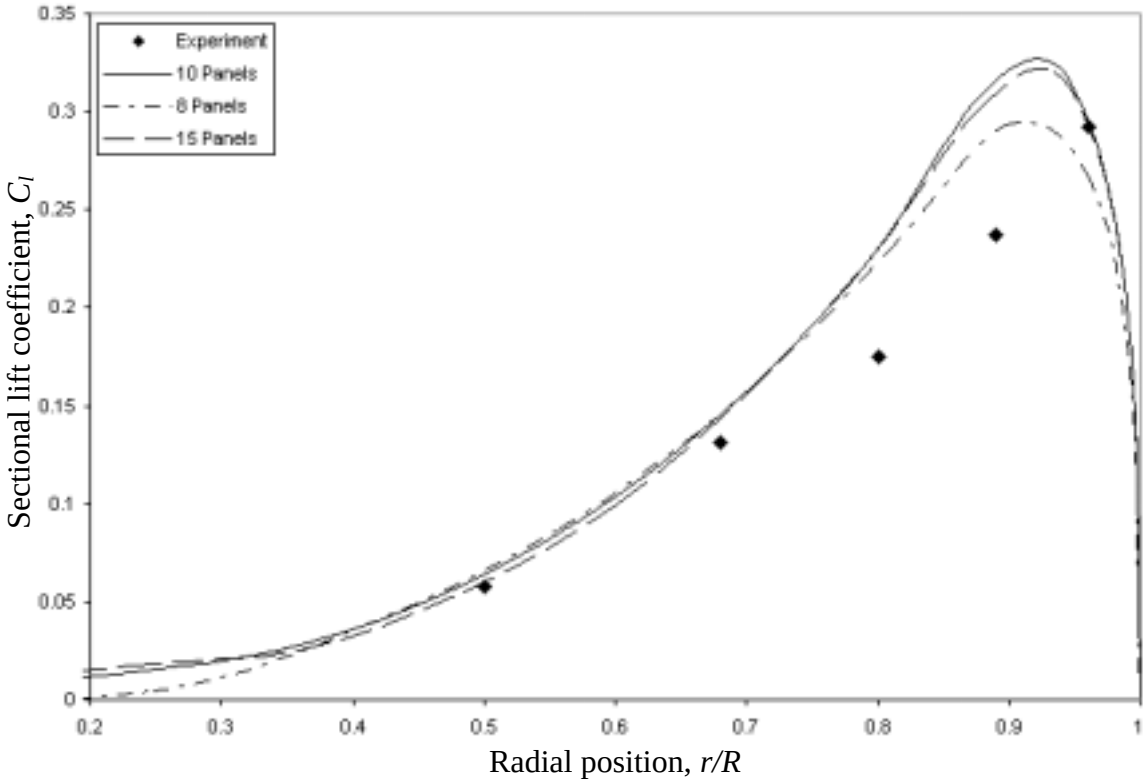


Figure 4.13: Spanwise lift coefficient (normalized by tip speed)

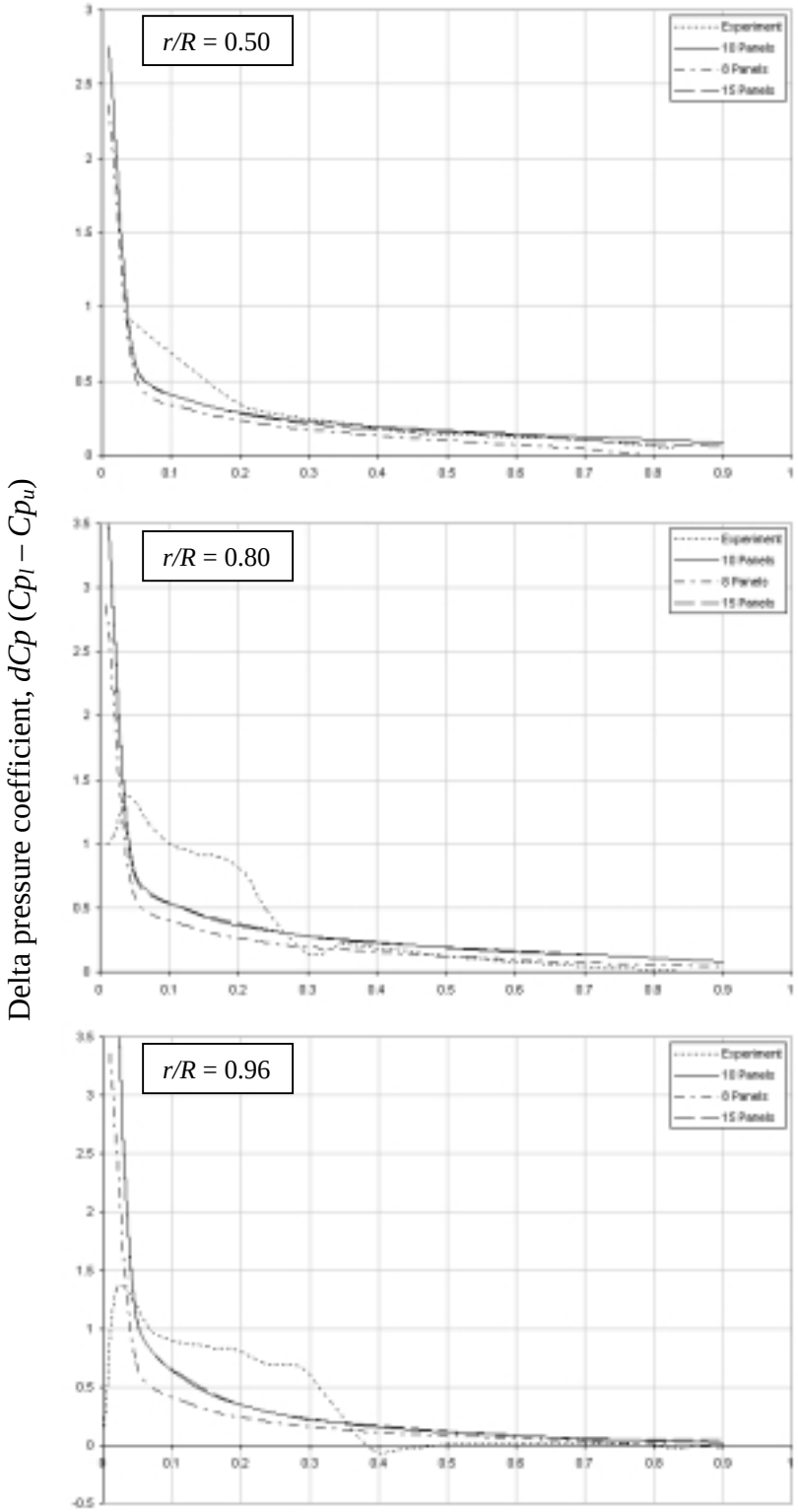


Figure 4.14: Chordwise pressure coefficient at different radial sections

4.2.2.5 Case 2e: Changing the Number of Chordwise Panels

Another case was run to test the effect of using more panels along the chord. The original case used 8 chordwise panels and ran in 65.5 hrs, while the modified case used 10 chordwise panels and ran in 72 hrs. *Figure 4.15* shows that using more panels changes the lift distribution slightly along the span. Likewise, *Figure 4.16* shows a small difference between the results from each case, with the results from the case using 10 panels more closely matching the experimental data. Still, the improvement in the results is small and not worth the extra computational cost at this stage.

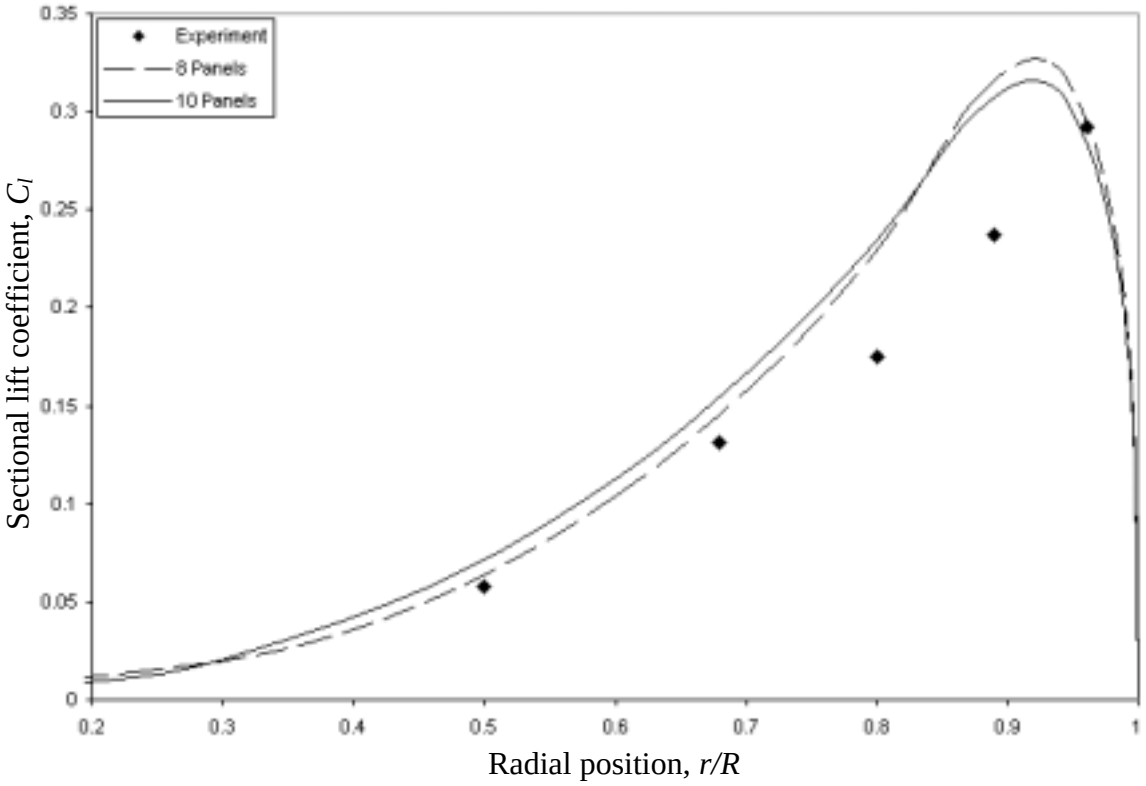


Figure 4.15: Sectional lift coefficient (normalized by tip speed)

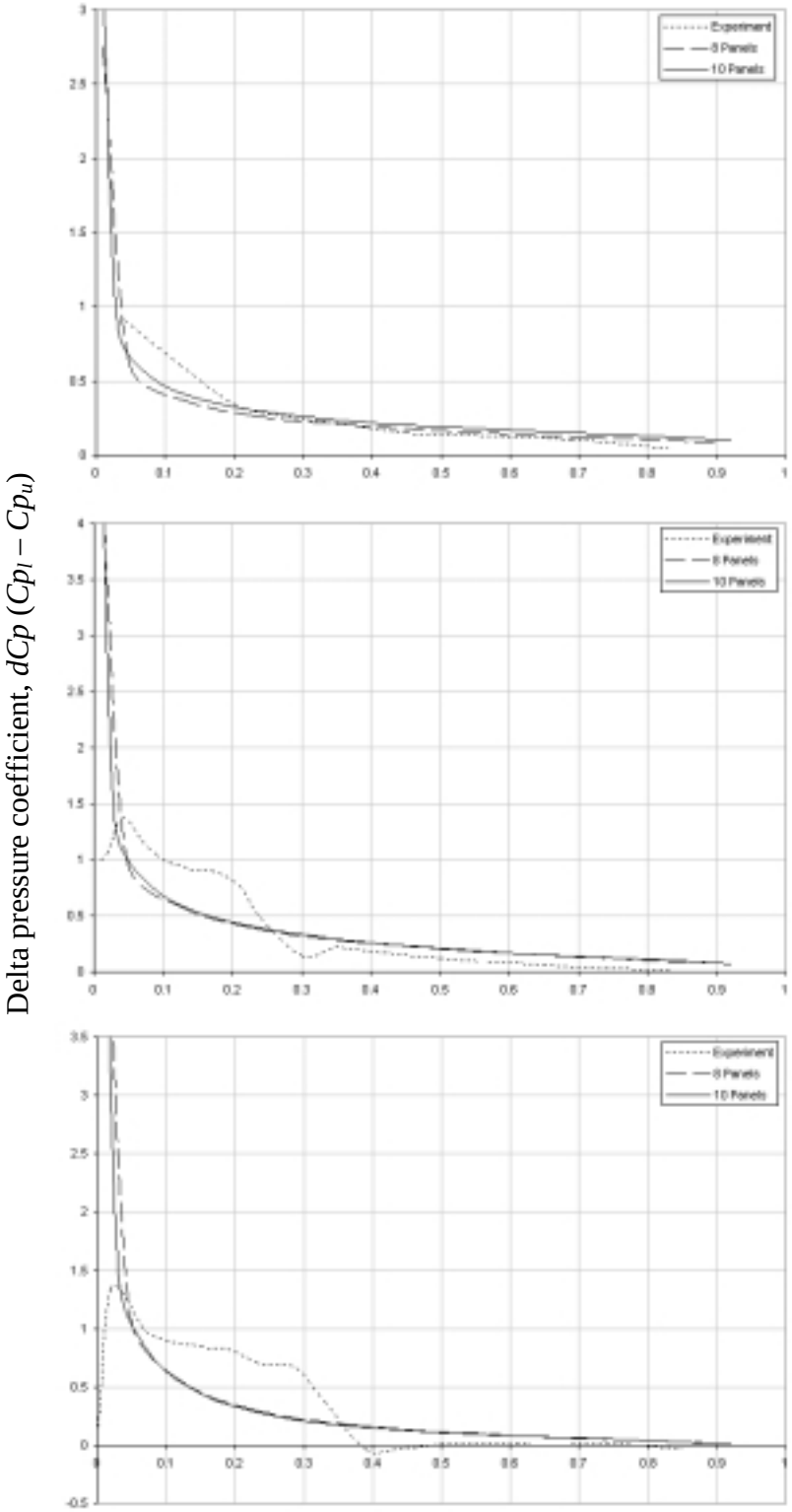


Figure 4.16: Chordwise pressure coefficient at different radial sections

4.3 Wake Visualization

With the code validated using experimental lift and pressure data, it is possible to look at the interesting characteristics of the wake structure. The wake essentially looks the same for all the validation cases run, with a few exceptions, since they were all hover cases. The roll-up of the tip vortex is slightly different depending on the number of spanwise elements on the blade and the size of the vortex core. Still, looking at the wake from the original compressible case (Case 2a) provides a good description of the flow physics seen in the other cases. *Figure 4.17* shows the fully developed wake trailing behind both blades after ten revolutions, where only the near wake region is shown for clarity. It shows the roll-up of the tip vortex and the vortex sheet being shed from the blades. Looking at the wake from just one blade from the side in *Figure 4.18*, it is possible to see the helical pattern of the wake as it is convected below the rotor disk. Notice that as the vortex sheet convects downward below the rotor, it convects more rapidly near the tip than it does inboard of the tip. This is due to the higher induced velocities that exist at the tip and leads to the vortex sheet becoming increasingly sloped from the root to the tip over time.

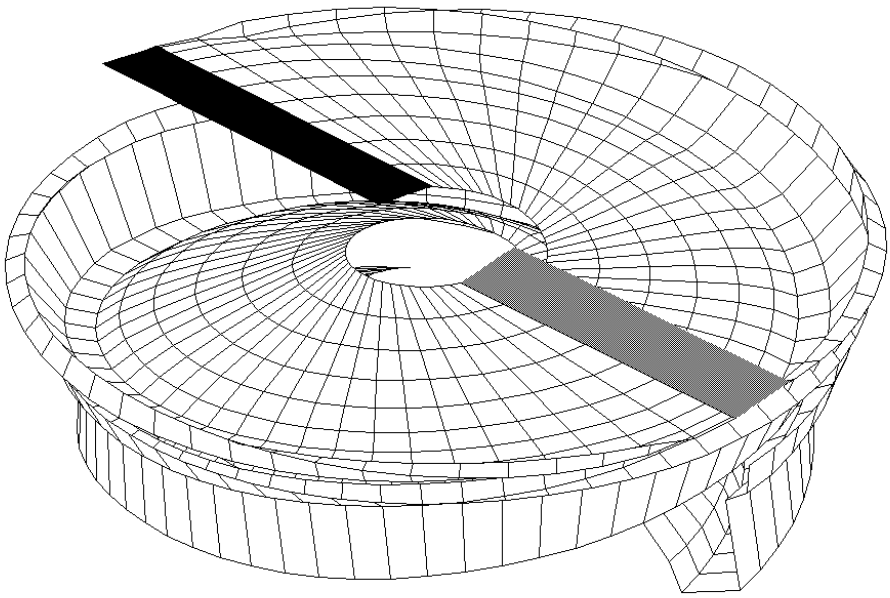


Figure 4.17: Wake trailing behind both blades

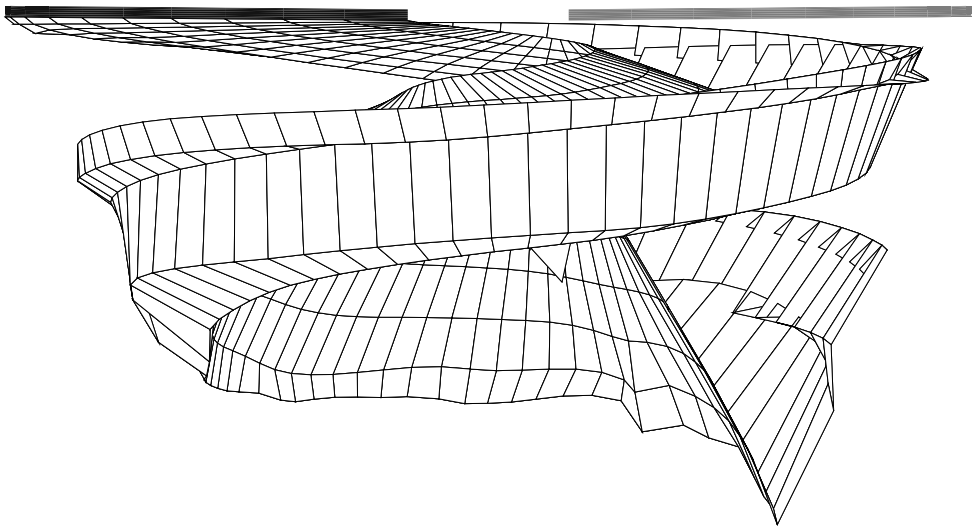


Figure 4.18: Wake trailing behind one blade

The convection of the tip vortices can be seen more clearly in *Figure 4.19*, where the slipstream boundary is added for emphasis. The slipstream boundary separates the

region of turbulent flow in the wake from the region of quiescent flow outside the wake. Notice that the tip vortices convect along the boundary and that the boundary contracts in the far wake. Contraction of the wake is measured by the wake contraction ratio, which is the ratio of the far wake radius to the radius of the blades. In this case, the wake contraction ratio is approximately 0.73, which corresponds favorably to the value of 0.707 obtained using momentum theory and the more practical value of approximately 0.78 seen in experiments [22]. The roll-up of the tip vortex after 180° is shown in *Figure 4.20*. Although this method does not allow for a detailed description of the tip vortex, it does at least show the presence of the roll-up.

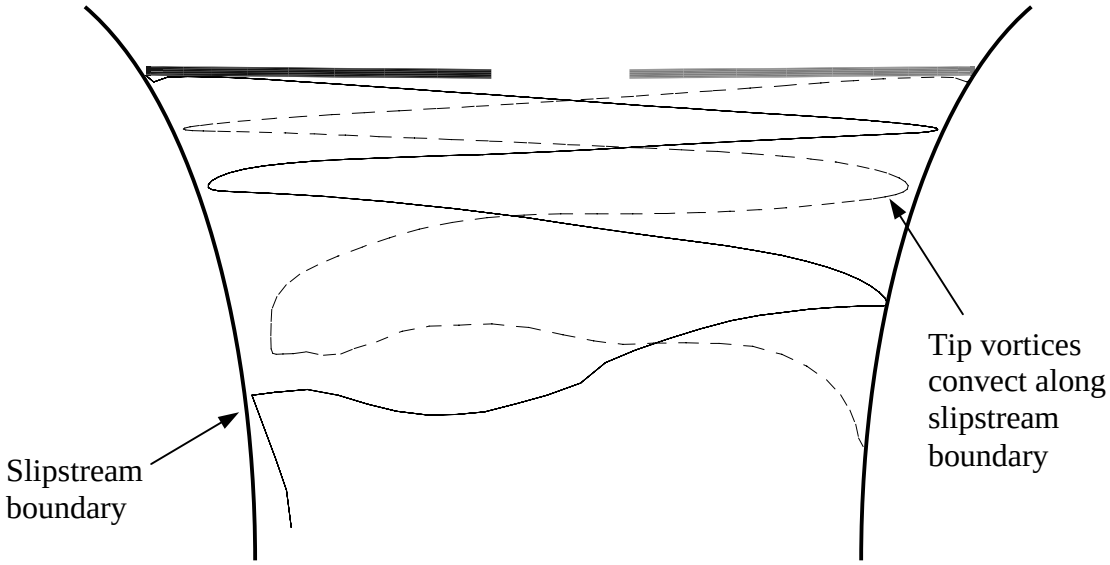


Figure 4.19: Path of tip vortices

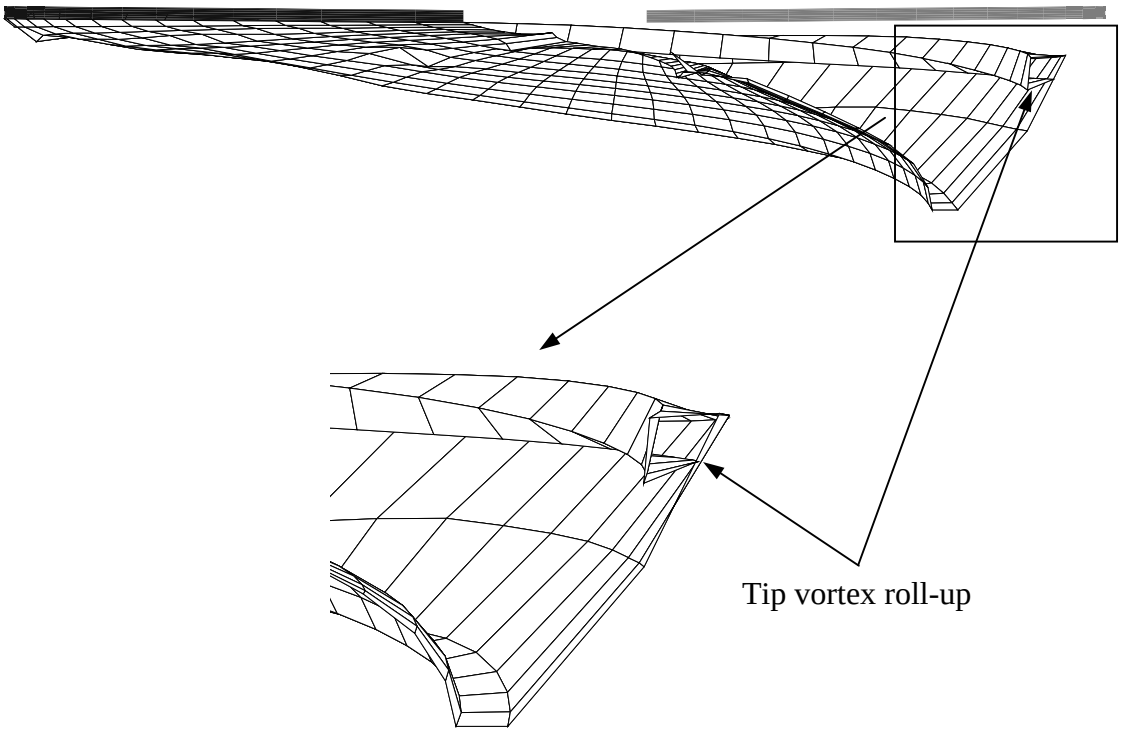


Figure 4.20: Roll-up of tip vortex at 180°

4.4 Application to Other Cases

Now that the code has been validated for hover using experimental data, it is possible to use the code to analyze flow in other operating conditions, such as climb and forward flight. Since the code was shown to be able to handle compressible flow, it can be used in typical forward flight conditions, where the flow on the advancing blade is compressible. The code does not currently trim the helicopter (i.e. account for blade flap, pitch control, or lead-lag), however, so the results in forward flight should only serve as an approximation.

4.4.1 Axial Climb

Using the same set-up as the validation cases, the code was used to simulate axial climb at a rate of approximately 1000 ft/min (5 m/s). In a helicopter, the rotor would need to provide extra thrust to climb, but in this case, since there is no weight to be lifted, the rotor is kept at a constant thrust while being vertically translated at the climb speed. This is equivalent to wind tunnel testing of a climbing rotor where the rotor is kept at constant thrust and an external fan blows air down over the rotor [30]. The vertical flow effectively reduces the angle of attack of the blade, which should result in less lift being produced. Since the climb velocity (5 m/s) is relatively small compared to the free stream velocity of the blades due to rotation (300 m/s at the tip), the lift in this case should only be slightly less than the lift in the hover case. *Figures 4.21* and *4.22* show this is in fact what is predicted by the code.

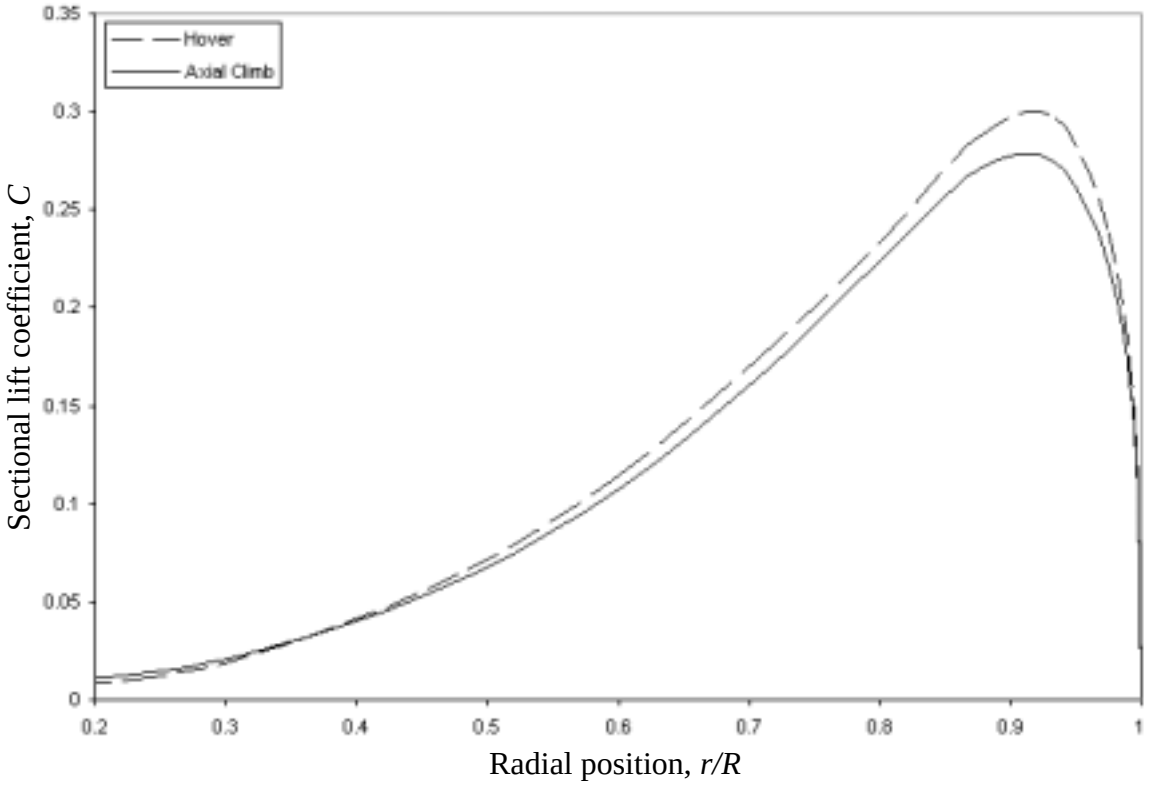


Figure 4.21: Sectional lift coefficient (normalized by tip speed)

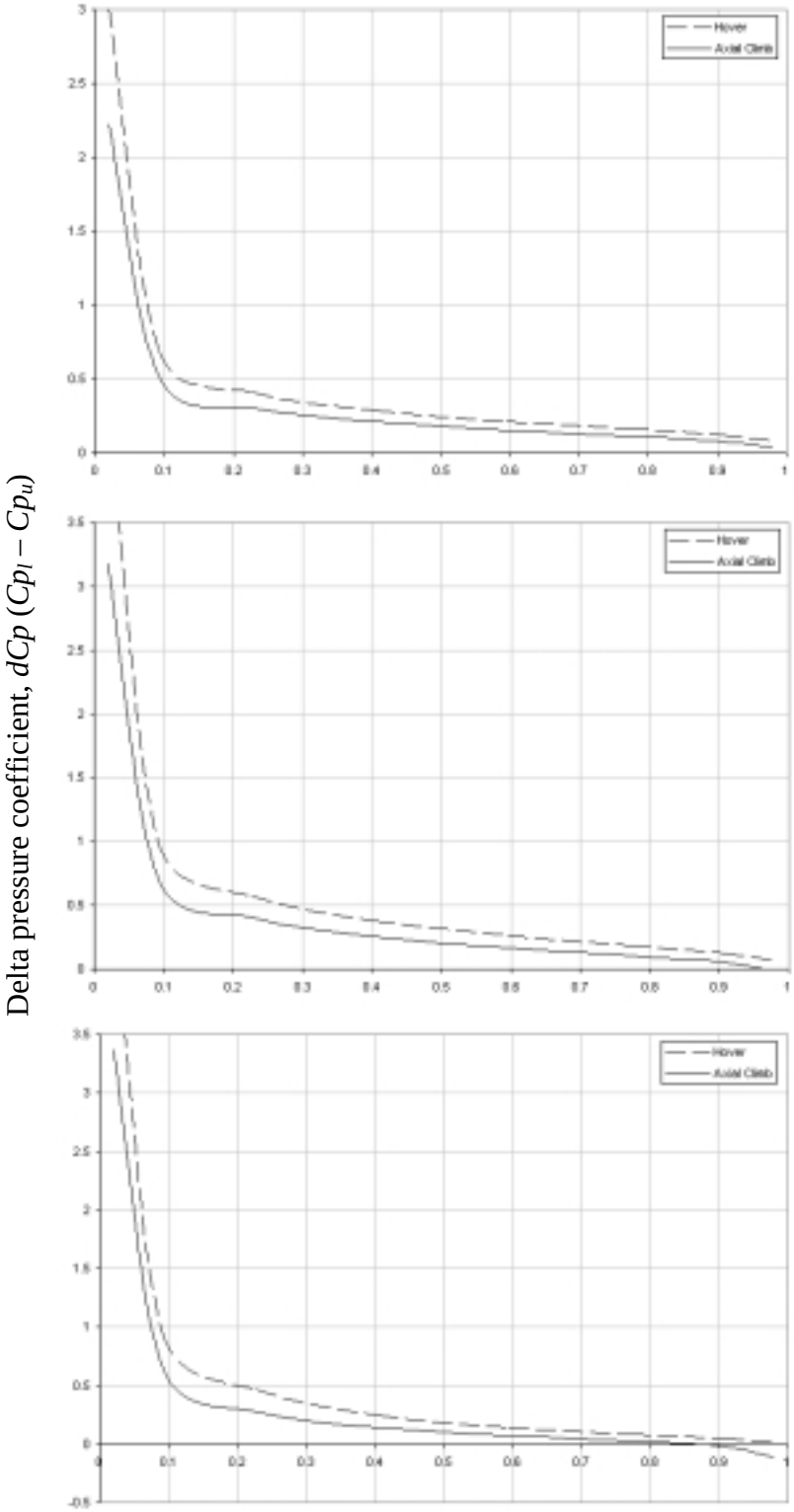
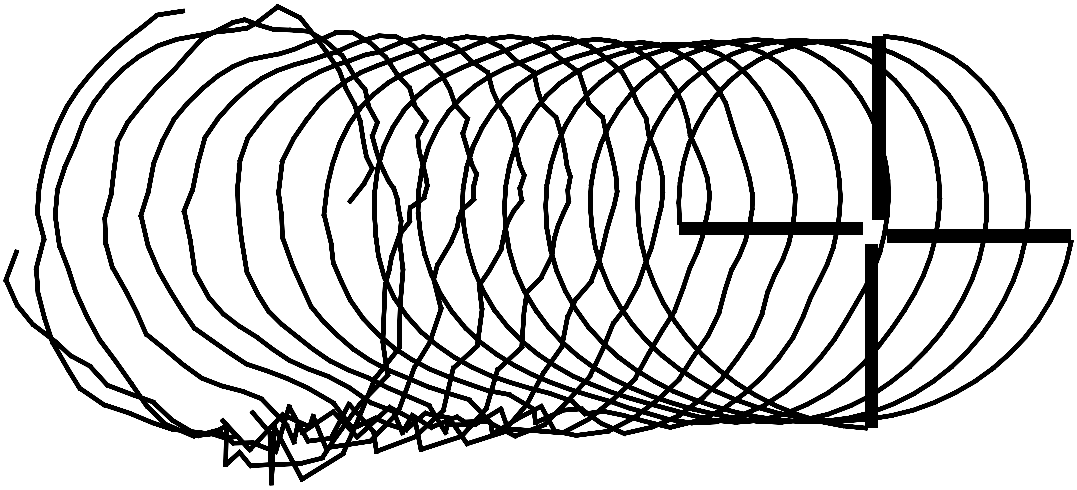


Figure 4.22: Chordwise pressure coefficient at different radial sections

4.4.2 Forward Climb

Another more interesting case is the combination of forward flight and climb, which is more typical for a helicopter than a pure axial climb. Since this code does not account for rotor trim, it cannot be used to accurately predict blade loading in forward flight, but it can still approximate the wake generated in such a case. A 4-bladed rotor with a diameter of 44 ft (13.4 m), aspect ratio of 16, collective pitch of 8°, and coning angle of 3° was rotated at 325.5 rpm (V_{tip} = 750 ft/s, 230 m/s). The rotor had a forward speed of 60 kts (31 m/s) and a climb speed of 3000 ft/min (15 m/s). *Figure 4.23* tracks the tip vortices coming off the blades seen from above and from the side. Recall that this is very similar to the smoke visualization image shown in Chapter 1 (*Figure 1.6*).

(a) Top View



(b) Side View

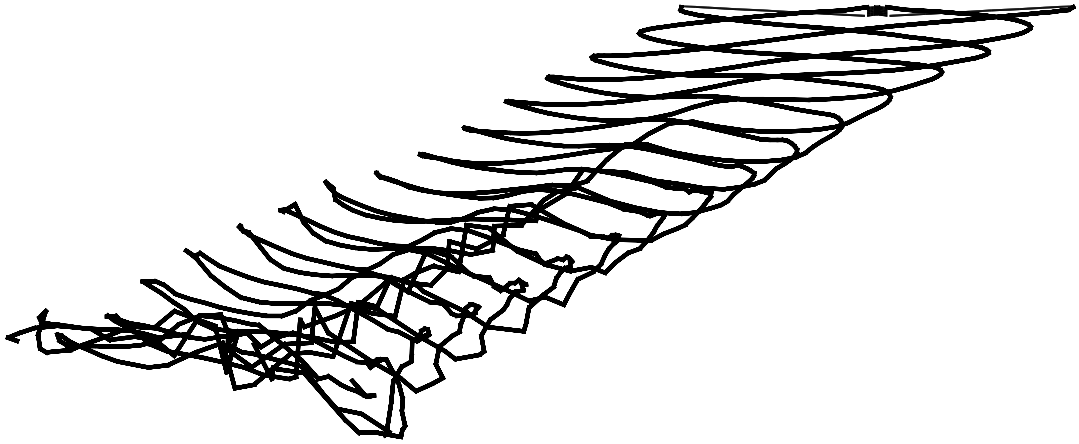


Figure 4.23: Tip vortices from a 4-bladed rotor in forward climb

Chapter 5

CONCLUSIONS

5.1 Summary and Conclusions

A code has been developed that accurately predicts the wake of a hovering rotor using a free wake vortex lattice method. The code solves the Biot-Savart Law to calculate vortex-induced velocities and uses a Prandtl-Glauert based correction to account for compressibility effects on the blade.

The accuracy of the code was determined by comparing the results with experimental data for two hover cases. The code was seen to closely match the experimental data for the first case where the flow was incompressible everywhere. It was also shown to predict the increased lift in the compressible region at the tip in the second case, whereas an incompressible version of the code did not. The results due not agree well near the leading edge of the blade, however, because of the singularity associated with the vortex lattice method and flat plate assumption. Using a panel method would eliminate this singularity, but it would also increase the computational time. Although the overall accuracy of the code was slightly less for the compressible case than the incompressible case, the results are still encouraging considering the complexity of the flow, the coarse grid used on the blades, and the assumptions made in the modeling process.

Several variations of the compressible case were run to test the sensitivity of the code to panel size, panel spacing, and vortex core radius. The first modified run used semi-cosine spacing along the chord instead of full-cosine spacing. Although the results were similar, it was determined that, because of the higher pressure gradients near the leading edge as compared to the trailing edge, semi-cosine spacing provides for the best use of computational resources. The second set of runs used different vortex core radii. The code was seen to be relatively insensitive to the core radius, as long as the radius was not larger than half the spanwise dimension of the smallest panel, which affected the roll-up of the tip vortex. Increasing the number of panels in the spanwise direction did not have any effect on the results, but it did greatly increase the computational time of the simulation. Decreasing the number of panels improved the computational time but did not provide the same level of accuracy. Increasing the number of panels in the chordwise direction, however, did improve the results slightly but with an increase in the computational time. Therefore, it was found that using eight panels along the chord and ten panels along the span with semi-cosine spacing in each direction to cluster smaller panels at the leading edge and the tip provides the best balance between accuracy and computational time.

After the code was validated, it was used to evaluate two other test cases to illustrate the versatility and potential of the code. First, the code was used to evaluate the flow during an axial climb of the rotor. The vertical motion of the rotor decreases the effective angle of attack of the blades, which produces slightly less lift than the hover case. Second, the code was used to evaluate the flow during a forward climb of the rotor,

which is more typically seen in helicopters than pure axial climb. Although the code does not account for blade flapping, pitch control, or lead-lag, it does provide an approximation of the wake structure and position in forward flight.

5.2 Suggestions for Future Work

The code showed good accuracy when compared to experimental data in both validation cases, but the agreement for the compressible case shows room for improvement. The most obvious choice for improving the results would be to use more panels on the blade or a smaller time step. Unfortunately, this does not always increase the accuracy of the code and can, in some cases, actually make the results worse. Using more panels near the leading edge will not eliminate the singularity associated with using the vortex lattice method on a flat plate either, although this could be dealt with by using a panel method. The downside to using more panels, a smaller time step, or a panel method, however, is the increased computational time. So before attempting to improve the results, it is necessary to look at ways to improve the speed of the code. The easiest way to do that would be to allow the code to run on multiple processors in parallel. Another common technique to improve the speed of rotor codes is to truncate the wake after a set number of revolutions, keeping only the tip vortex. Because the effect of the wake on the blades diminishes the as the wake moves farther downstream, truncating the far wake should have little effect on the lift of the blades. Another possible way to improve the speed would be to only use the compressible correction in the near wake region, where the compressibility effects are greatest.

Although the goal in this case was to determine the lift on the rotor blades, another common use of the free wake method is to predict wake interference effects with the fuselage or tail rotor. Unfortunately, in its current state, this method creates unnatural distortions of the far wake over time. So, in order to be able to use the far wake geometry for practical purposes, the code must be modified to eliminate these distortions. This is usually done by allowing the vortex core radius to grow over time, which reduces the induced velocities and damps out the wake, or by some other means of artificial damping.

As was shown by the axial climb and forward climb cases, the code has the potential to evaluate many different types of operating conditions of a helicopter. To truly analyze forward flight, though, the code must first be able to account for rotor trim. Understanding the complex coupling of blade flapping, cyclic pitch, and lead-lag is not an easy task, however, as it requires an advanced knowledge of rotorcraft dynamics. Likewise, implementing such a scheme into the code is not completely straightforward. Solving for the rotor trim is an iterative process, which must then be applied to the blade grids to recalculate the positions of all the panels at each time step. If a trim code is added, though, any prescribed flight path could be input into the code, and the wake could be accurately modeled.

The code could easily be adapted to handle multiple rotors for a tilt-rotor or tandem-rotor helicopter. Likewise, a mirror image of the rotor could be used to simulate ground effect by placing it below the ground plane at a distance equal to the height of the rotor. Combining the use of multiple rotors and ground effect would allow tilt-rotor and tandem-rotor helicopters to be studied in ground effect or, more interestingly, in the

approach to a naval ship or elevated helipad where one rotor is in ground effect and the other is not. Because the code can handle rotation of the blades about more than one axis, it could also be used to analyze the wake during the transition phase of a tilt-rotor as it changes from helicopter to airplane or vice versa.

BIBLIOGRAPHY

1. Beddoes, T.S. "A Wake Model for High Resolution Airloads." *Proceedings of the 2nd International Conference on Basic Rotorcraft Research*. Triangle Park, NC, 1985.

2. Berkey, D.D. and Blanchard, P. *Calculus, 3rd Edition*. Saunders College Publishing, 1992.

3. Bertin, J.L. and Smith, M.L. *Aerodynamics for Engineers, 3rd Edition*. Prentice Hall, Upper Saddle River, NJ, 1998.

4. Berkman, M.E., Sankar, L.N., Berezin, C.R., and Torok, M.S. "A Navier-Stokes/Full Potential/Free Wake Method for Advancing Multi-Blade Rotors." *Proceedings of the 53th Annual Forum of the American Helicopter Society*. Virginia Beach, VA, May 1997.

5. Caradonna, F.X. and Tung, C. "Experimental and Analytical Studies of a Model Helicopter Rotor in Hover." *NASA TM-81232*, 1981.

6. Chen, C.L. and McCroskey, W.J. "Numerical Simulation of Helicopter Multi-Bladed Rotor Flow." *AIAA 26th Aerospace Sciences Meeting*. AIAA-88-0046. Reno, NV, Jan. 1988.

7. Clark, D.R. and Leiper, A.C. "The Free Wake Analysis, A Method for the Prediction of Helicopter Rotor Hovering Performance." *Journal of the American Helicopter Society*. Vol. 15, No. 1, Jan. 1970, pp. 3-11.

8. Currie, I.G. *Fundamental Mechanics of Fluids*. McGraw-Hill, 1993.

9. Duque, E.N. and Srinivasan, G.R. "Numerical Simulation of a Hovering Rotor Using Embedded Grids." *Proceedings of the 48th Annual Forum of the American Helicopter Society*. Washington, DC, June 1992.

10. Egolf, T.A. and Landgrebe, A.J. "Generalized Wake Geometry for a Helicopter Rotor in Forward Flight and Effect of Wake Deformation on Airloads." *Proceedings of the 40th Annual Forum of the American Helicopter Society*. Arlington, VA, May 1984.

11. Egolf, T.A. and Massar, J.P. "Helicopter Free Wake Implementation on Advanced Computer Architectures." *2nd International Conference on Basic Rotorcraft Research*. College Park, MD, Feb. 1988.

12. Epstein, R.J. and Bliss, D.B. "Wake Generation Compressibility Effects in Unsteady Lifting Surface Theory." *15th AIAA Applied Aerodynamics Conference*. Atlanta, GA, June 1997.

13. Fung, Y.C. *An Introduction to the Theory of Aeroelasticity*. John Wiley & Sons, Inc., New York, NY, 1955.

14. Gennaretti, M. and Morino, L. "A Boundary Element Method for the Potential, Compressible Aerodynamics of Bodies in Arbitrary Motion." *Aeronautical Journal*. Vol. 96, Jan. 1992, pp. 15-19.

15. Hall, C.M. "High-Order Accurate Simulations of Wake and Tip Vortex Flowfields." M.S. Thesis, Pennsylvania State University, Department of Aerospace Engineering, Dec. 1998.

16. Johnson, W. *Helicopter Theory*. Dover Publications, Inc., New York, 1994.

17. Katz, J. and Maskew, B. "Unsteady Low-Speed Aerodynamic Model for Complete Aircraft Configurations." *Journal of Aircraft*. Vol. 25, No. 4, Apr.1988, pp. 302-310.

18. Katz, J. and Plotkin, A. *Low-Speed Aerodynamics, 2nd Ed*. Cambridge University Press, 2000.

19. Kinsler, L.E., Frey, A.R., Coppens, A.B., Sanders, J.V. *Fundamentals of Acoustics, 4th Edition*. John Wiley & Sons, Inc., 2000.

20. Landgrebe, A.J. "An Analytical Method for Predicting Rotor Wake Geometry." *Journal of the American Helicopter Society*. Vol. 14, No. 4, Oct. 1969, pp. 20-32.

21. Landgrebe, A.J. "The Wake Geometry of a Hovering Helicopter Rotor and its Influence on Rotor Performance." *Journal of the American Helicopter Society*. Vol. 17, No. 4, Oct. 1972, pp. 3-15.

22. Leishman, J.G. *Principle of Helicopter Aerodynamics*. Cambridge University Press, 2000.

23. Leishman, J.G., Martin, P.B, and Pugliese, G.J. "Surface and Wake Flow Characteristics of a Hovering Helicopter Rotor." *9th International Symposium on Flow Visualization*. Edinburgh, Scotland, UK, Aug. 2000.

24. LINPACK web page. (http://www.netlib.org/linpack/index.html).

25. Long. L.N. "The Compressible Aerodynamics of Rotating Blades Based on an Acoustic Formulation." *NASA TP-2197*, 1983.

26. Long, L.N. and Watts, G.A. "Arbitrary Motion Aerodynamics Using an Aeroacoustic Approach." *AIAA Journal*. Vol. 25, No. 11, Nov. 1987, pp. 1442-1448.

27. Miranda, L.R., Elliot, R.D., and Baker, W.M. "A Generalized Vortex Lattice Method for Subsonic and Supersonic Flow Applications." *NASA CR-2865*, 1977.

28. Moulton, M.A., Wenren, Y., and Caradonna, F.X. "Development of an Overset/Hybrid CFD Method for the Prediction of Hovering Performance." *Proceedings of the 53th Annual Forum of the American Helicopter Society*. Virginia Beach, VA, May 1997.

29. Morino, L. "A General Theory of Unsteady, Compressible, Potential Aerodynamics." *NASA CR-2462*, 1974.

30. Prouty, R.W. "Helicopter Aerodynamics." *Course Notes from A Comprehensive Short Course in Rotary Wing Technology*. University Park, PA, Aug. 2001.

31. Sadler, S.G. "A Method for Predicting Helicopter Wake Geometry, Wake-Induced Flow and Wake Effects on Blade Airloads." *Proceedings of the 27th Annual Forum of the American Helicopter Society*. Washington, DC, May 1971.

32. Schreier, S. *Compressible Flow*. John Wiley & Sons, Inc., 1982.

33. Sezer-Uzol, N. "High Accuracy Wake and Vortex Simulations Using a Hybrid Euler/Discrete Vortex Method." M.S. Thesis, Pennsylvania State University, Department of Aerospace Engineering, May 2001.

34. Srinivasan, G.R., Baeder, J.D., Obayashi, S., and McCroskey, W.J. "Flowfield of a Lifting Rotor in Hover: A Navier-Stokes Simulation." *AIAA Journal*. Vol. 30, No. 10, Oct. 1992, pp. 2371-2378.

35. Stahl-Cucinelli, H. "Vortex-Lattice Free Wake Model For Helicopter Rotor Downwash." *20th European Rotorcraft Forum*. Amsterdam, NL, Oct. 1994.

36. Steinhoff, J. and Ramachandran, K. "Free Wake Analysis of Compressible Rotor Flows." *AIAA Journal*. Vol. 28, No. 3, Mar. 1990, pp. 426-431.

37. Strawn, R.C. and Barth, T.J. "A Finite-Volume Euler Solver for Computing Rotary-Wing Aerodynamics on Unstructured Grids." *Proceedings of the 48th Annual Forum of the American Helicopter Society*. Washington, DC, June 1992.

38. Tauszig, L. "Numerical Detection and Characterization of Blade Vortex Interactions Using a Free Wake Analysis." M.S. Thesis, Pennsylvania State University, Department of Aerospace Engineering, Aug. 1998.

39. Tung, C., Caradonna, F.X., and Johnson, W. "The Prediction of Transonic Flows on Advancing Rotors." *Journal of the American Helicopter Society*. Vol. 31, No. 3, July 1986, pp. 4-9.

40. U.S. Navy Photo Gallery web page. (http://www.chinfo.navy.mil/navpalib/.www/digital.html).

41. Ward, G.N. *Linearized Theory of Steady High-Speed Flow*. Cambridge University Press, 1955.

Appendix A

SAMPLE INPUT FILE

```
! Timestep size, dt (s)
0.0004

! Number of time steps, nsteps
600

! Rotational speeds of rotor, rmpx, rpmy, rpmz (rpm)
0, 0, 2500

! Number of blades, Nb
2

! Number of nodes (panels + 1) along chord and span, nx, ny
9, 11

! Maximum size of wake, nwakemax
600

! Aspect Ratio, AR
6

! Radius, R (m)
1.143

! Root cutout, e_by_c (number of chords)
1.0

! Core size as fraction of smallest spanwise element, rc
0.1

! Fuselage velocity in x, y, and z directions, u, v, w (m/s)
0, 0, 0

! Rotation point of rotor in space, hx, hy, hz
0, 0, 0

! Collective pitch, theta (deg)
8.0

! Precone Angle, beta (deg)
0.5

! Interval for Writing Tecplot Data (60 iterations = 1 revolution)
60
```

Appendix B

COMPUTER PROGRAM SUBROUTINES

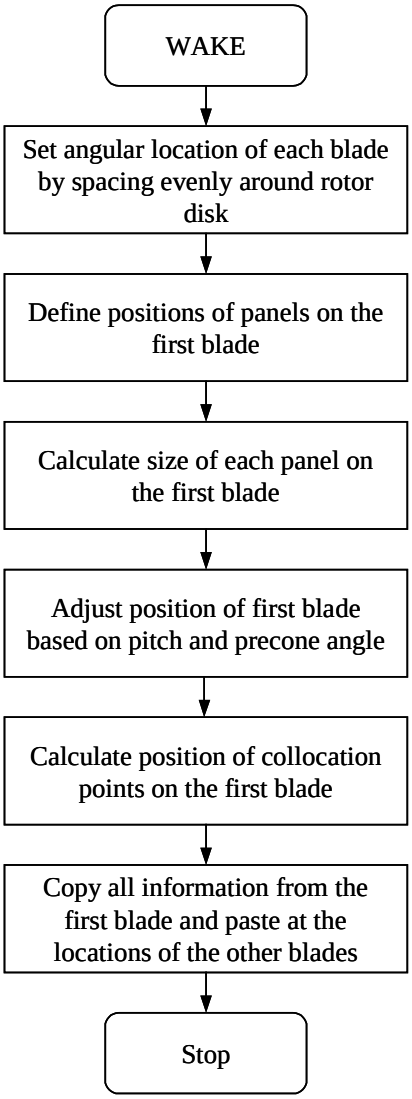


Figure B.1: READGRID subroutine

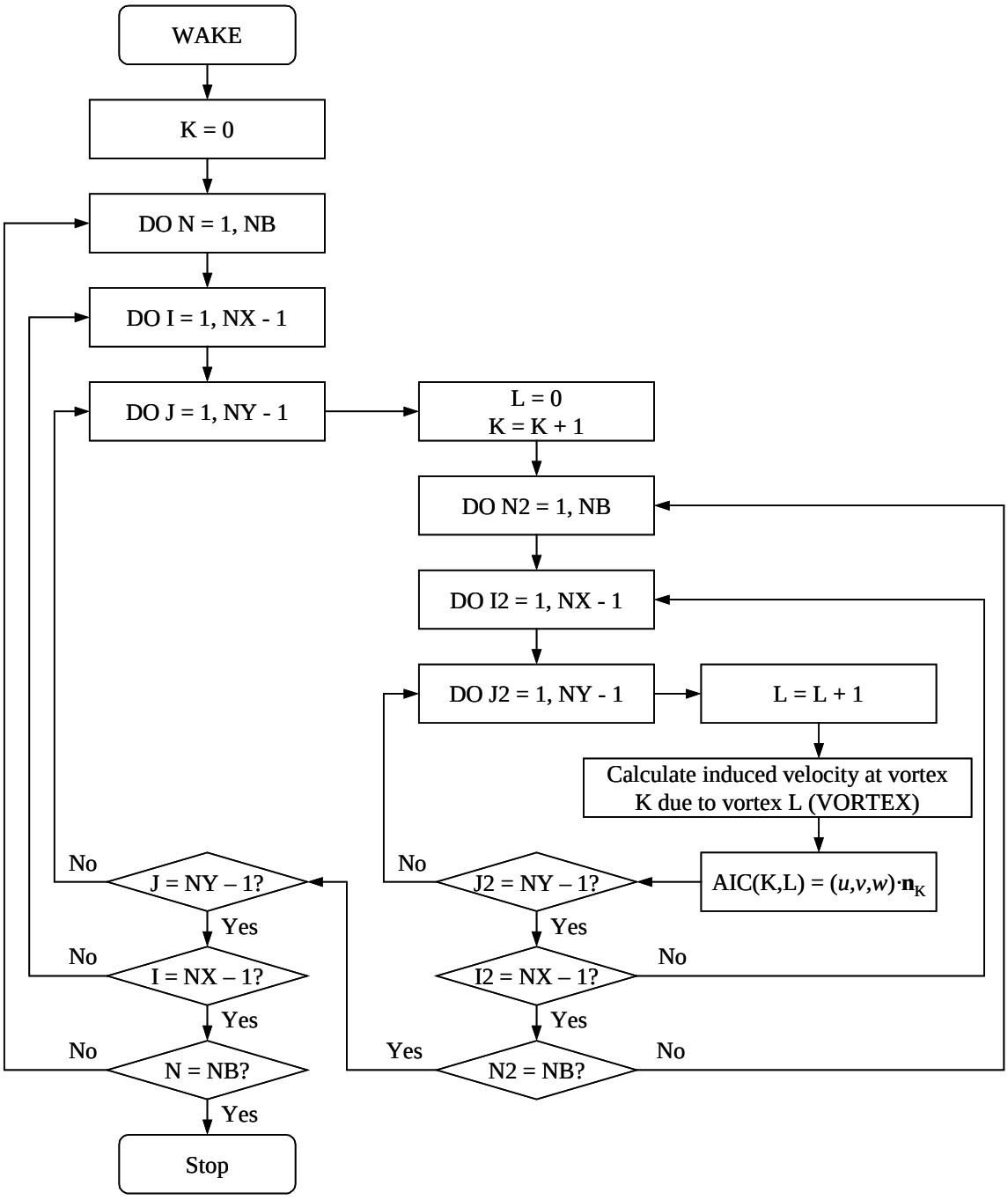


Figure B.2: CALCAIC subroutine

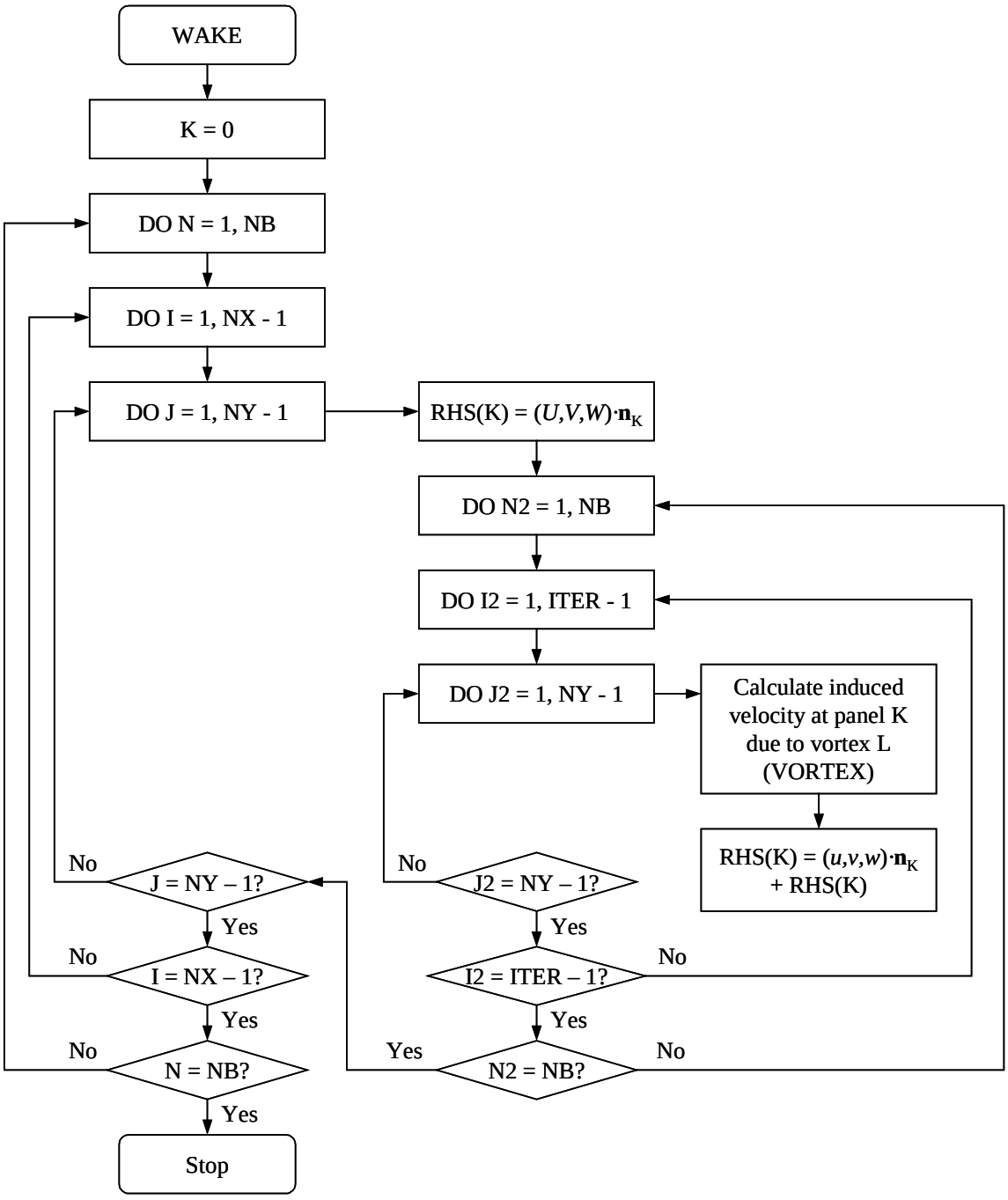


Figure B.3: CALCRHS subroutine

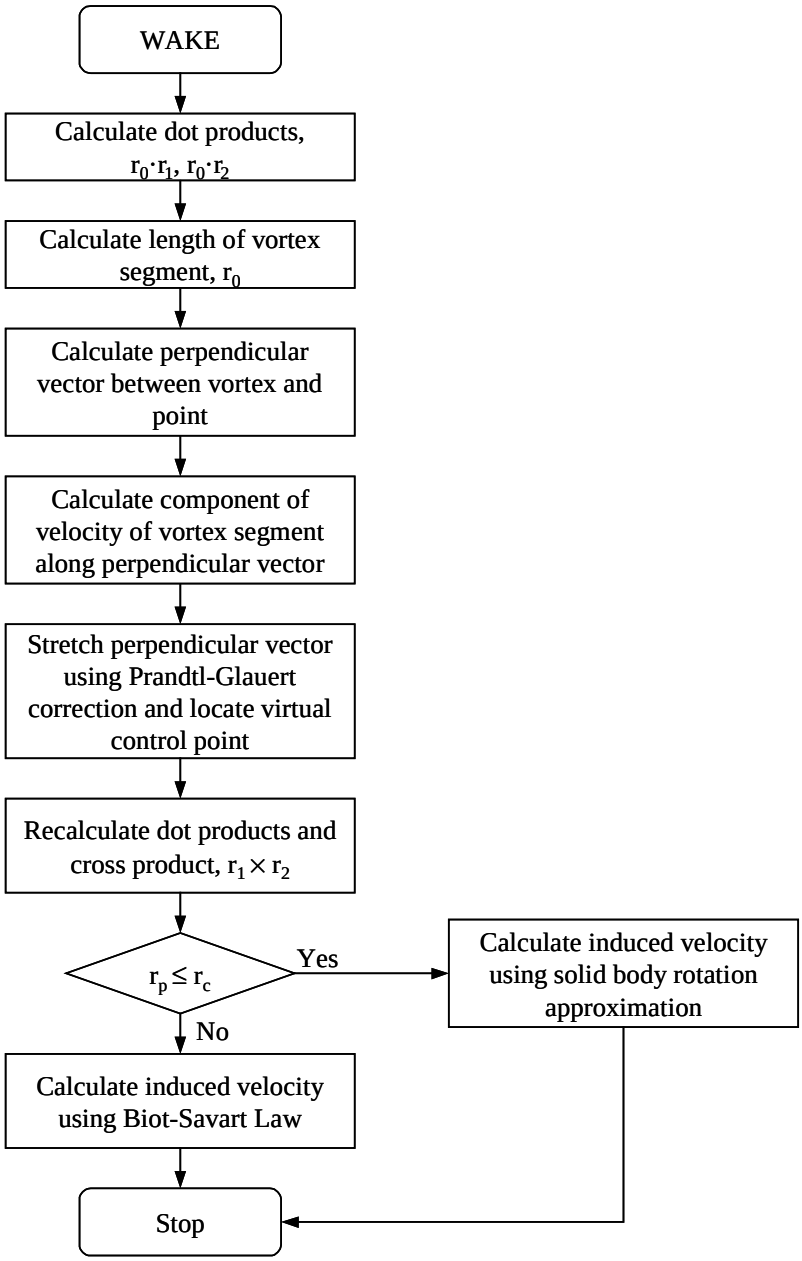


Figure B.4: VORTEX subroutine

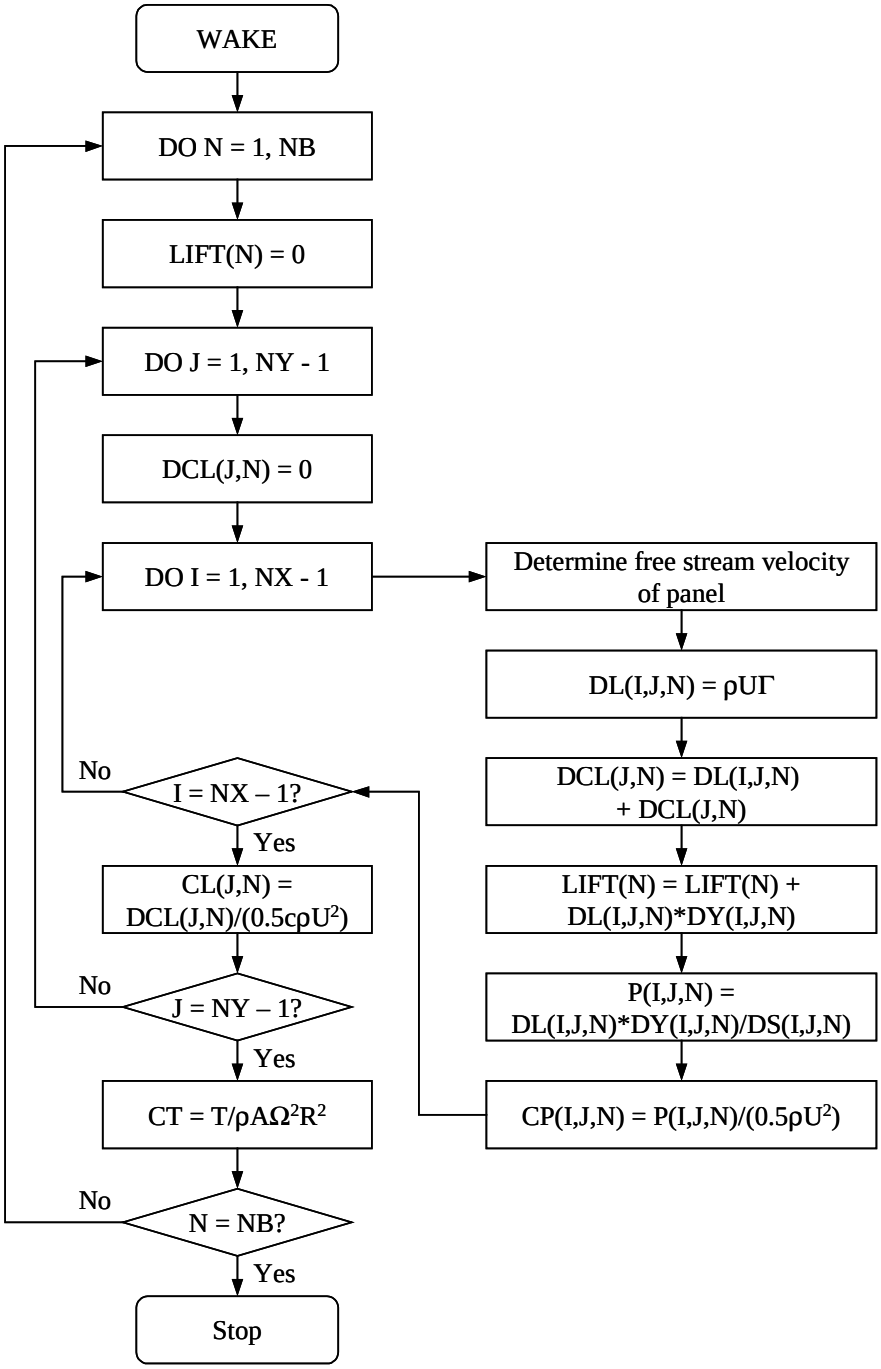


Figure B.5: PRESSURE subroutine

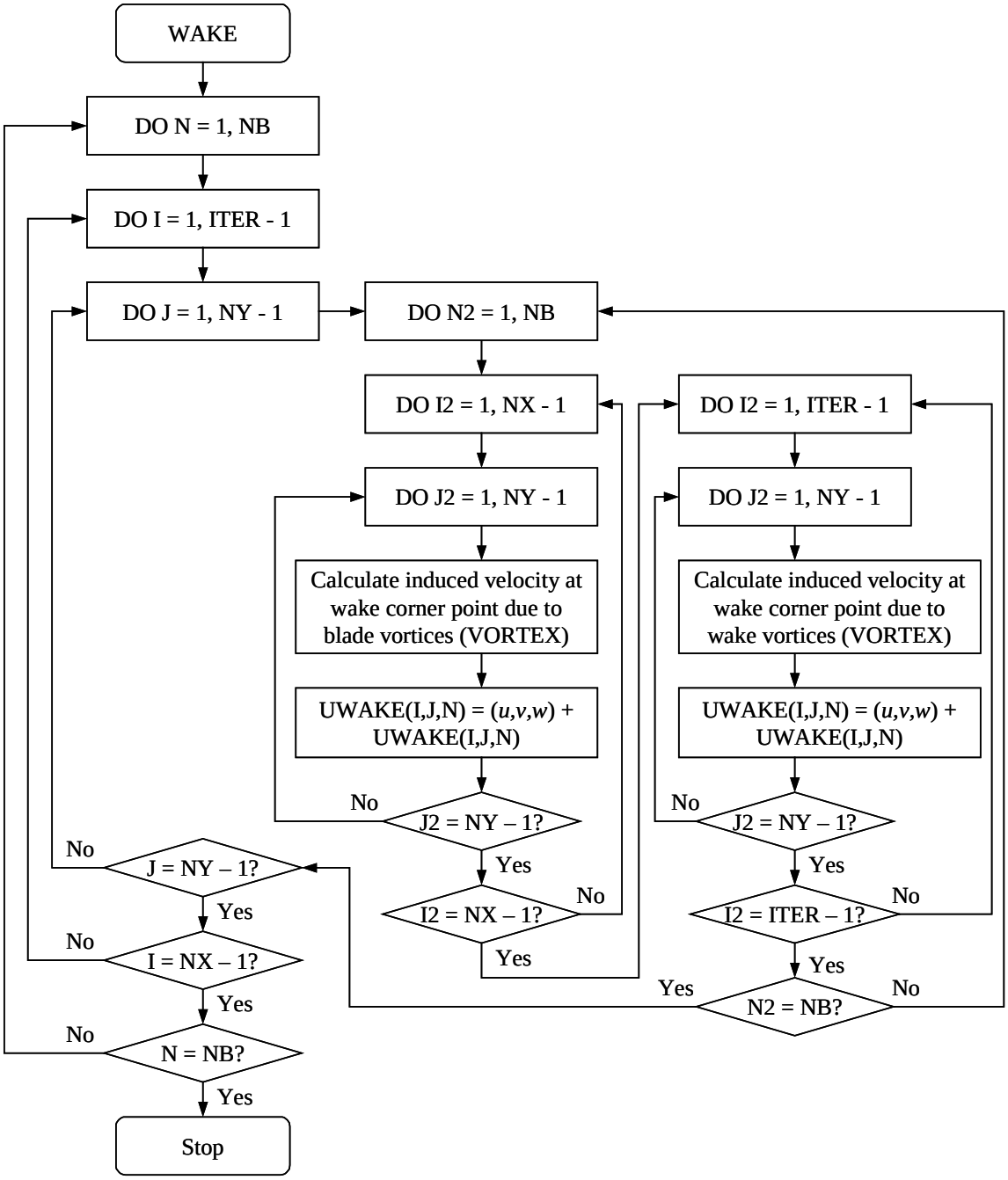


Figure B.6: WAKEVELOC subroutine