

Development of Vortex Box (VBOX) and Vortex Tip (VTIP) Filament Free-Wake Rotor Models

By

Casey Will Matsler

THESIS

Submitted in partial fulfillment of the requirements for the degree of

Master of Science in Aerospace Engineering

at The University of Texas at Arlington

May 2023

Arlington, Texas

Supervising Committee:

Dr. Donald Wilson, Thesis Advisor

Dr. Brian Dennis, Committee Member

Dr. Kamesh Subbarao, Committee Member

Dr. Dudley Smith, External Committee Member

Mr. Mark Dreier, External Committee Member

## Abstract

### Development of Vortex Box (VBOX) and Vortex Tip (VTIP) Filament Free-Wake Rotor Models

Casey Will Matsler, M.S.

The University of Texas at Arlington, 2023

Supervising Professor: Dr. Donald Wilson

Rotor models span from simple static and first order expressions to high fidelity, numerical computations and across the spectrum each model is good for different purposes. The use of models has improved with time and technological capability but choosing the right model for the task is important. A survey of rotor models is performed to identify a rotor model flexible enough for providing high to moderately high-fidelity results without significantly sacrificing computational time. The model chosen is intended to be for general purposes and applicable across a wide range of operating conditions. From the survey, a blade element rotor model is chosen and combined with two vortex-theory free-wake models. The blade element rotor model allows for large range of flexibility to model many physical aspects of rotors. The vortex-theory free-wake model was selected for the ability to simulate a rotor wake's physical characteristics and still provide solid inflow modeling for coupling with the blade motion dynamics. The purpose of having two versions of vortex-theory free-wake models is to quantify the differences of choosing whether or not to satisfy Helmholtz's Law for termination of a vortex filament.

The satisfaction of Helmholtz's Law is accomplished by emitting four vortex filaments with constant vorticity strength each time step that are adjoined to make a closed box representing the shed, bound

and tip vortices. The second vortex-theory free-wake model only emits one filament from the tip of each blade per time step creating a streamer of filaments each with varying vorticity strength.

The development of the rotor blade element model and the vortex-theory free-wake models are described along with the momentum theory inflow model used for comparison and validation. The resulting models are compared against empirical performance calculations, experimental test data, and physical wake characteristics from theory and reality.

Results from the blade element model with both vortex-theory free-wake models show good correlation with rotor power and induced velocity predictions. The vortex-theory free-wake models capture lateral flapping trends that uniform, momentum theory inflow models fail to predict. The velocity field resulting from the vortex-theory free-wake models strongly align with known wake characteristics such as wake contraction and vortex tip roll up during forward flight. The interactions of the vortices also suggest that the free-wake models are capturing transient vibration effects during transitional speeds. The wake dynamics show similar results to accepted dynamic inflow models and show the apparent mass effects described in the development of the dynamic inflow models. Resulting similarity in the two vortex-theory free-wake models indicate that satisfaction of Helmholtz's Law is not strictly necessary for creating good rotor analysis tools.

## Acknowledgements

Special thanks to Mr. Mark Dreier for his time and contributions and to Dr. Dudley Smith for initiating and encouraging the completion of my thesis in a rotorcraft topic.

Particularly special thanks to Samantha, Heidi, and Octavia for the enduring late nights and short weekends. I am extremely proud of their commitment and the sacrifices they made on behalf of my education.

# Table of Contents

Abstract.....	ii
Acknowledgements.....	iv
Table of Contents.....	v
List of Figures .....	vii
List of Symbols .....	viii
CHAPTER 1. Introduction .....	1
CHAPTER 2. Background of Rotor Models .....	3
2.1 Momentum Theory Models.....	3
2.1.1 Basic Momentum Theory Model .....	3
2.1.2 Blade Element Momentum Theory Models.....	4
2.1.3 Improved Momentum Theory Models .....	5
2.2 Algebraic Rotors.....	5
2.3 Blade Element Rotor (BER) Models.....	6
2.4 Prescribed Wakes.....	7
2.4.1 Stream Tube Prescribed Wakes .....	7
2.4.2 Helical Prescribed Wakes .....	7
2.4.3 Ribbon Prescribed Wake .....	8
2.5 Free-Wakes .....	8
2.5.1 Computational Fluid Dynamics .....	8
2.5.2 General Vortex Theory and the Application of Biot-Savart Law in Aerodynamics .....	9
2.5.3 Vortex Ring Models.....	13
2.5.4 Particle Vortex Transport Method Wake.....	14
2.5.5 Free Vortex Wake .....	15
CHAPTER 3. Project Rotor Model.....	17
3.1 Blade Element Rotor (BER) Model .....	17
3.1.1 Description of Model Selection.....	17
3.1.2 Rotor Parameter Setup .....	17
3.1.3 Rotor Axes and Transformations .....	19
3.1.4 Rotor Forces and Moments .....	22
3.1.5 Rotor Controls and State Variables.....	24
3.1.6 Rotor Model Process.....	25
3.2 Wake Models .....	26

3.2.1	Momentum Inflow Model.....	26
3.2.2	VBOX .....	27
3.2.3	VTIP .....	29
3.2.4	VTIP and VBOX Common Method.....	30
3.2.5	Model Trimming.....	31
3.2.5.1	Vortex Life Model.....	35
CHAPTER 4.	Results.....	36
4.1	Description of General Rotor Model Parameters .....	36
4.2	Comparison of Speed Power Polar .....	36
4.3	Comparison of Lateral Flapping vs Advance Ratio.....	40
4.4	Comparison of Induced Power Required Due to Swirl Wake .....	41
4.5	Comparison of Wake Dynamics .....	42
4.6	Comparison of Wake Characterization .....	44
4.7	Comparison of Induced Velocities in High Descent Rates .....	52
4.8	Comparison of VTIP vs VBOX .....	53
CHAPTER 5.	Summary .....	54
CHAPTER 6.	Recommendations/Future Work.....	56
6.1	Expand BER Functionality.....	56
6.2	Increase model performance for Realtime.....	56
6.3	Inclusion of Full Up Air Vehicle Simulation Model.....	56
6.4	V BOX Improvement for Higher Fidelity Near Wake Modelling .....	57
6.5	Further investigation into high descent rates.....	57
CHAPTER 7.	References .....	58
Appendix A.....		60
Appendix B.....		62

## List of Figures

Figure 1. 2D Example of Vortex Induced Velocity with Core Radius.....	12
Figure 2. Vortex Ring Model Display in Y-Z Plane .....	14
Figure 3. Blade Reference Line Illustration .....	19
Figure 4. Formation of VBOX Geometry, Top View .....	28
Figure 5. Formation of VTIP Geometry, Top View .....	29
Figure 6. Hover Induced Velocity During Execution .....	32
Figure 7. Hover Blade Flapping During Execution.....	33
Figure 8. Hover, Blade Rate of Change of Flapping During Execution .....	33
Figure 9. Hover, Rotor Forces During Execution.....	34
Figure 10. Comparison of Total Power Required Results .....	39
Figure 11. Comparison of Induced Power Required Results .....	39
Figure 12. Comparison of Induced Velocity Results .....	40
Figure 13. Lateral Flapping Comparison .....	41
Figure 14. Induced Velocity Response to Collective Step .....	44
Figure 15. VTIP Hover Wake Visualization. ....	46
Figure 16. Velocity Field at Midplane of Rotor .....	47
Figure 17. Tip Drive - Wake Structure in Forward Flight [18] .....	48
Figure 18. 3D View of VTIP Free-wake showing Tip Roll-Up and Classical Horseshoe Vortex .....	49
Figure 19. Prescribed Wake Example [3] .....	49
Figure 20. VTIP Wake in Hover with Dynamic Wake Buildup .....	51
Figure 21. Induced Velocity for Vertical Descent Rates Comparisons.....	52
Figure 22. Program Detailed Flow Chart (1 of 2) .....	60
Figure 23. Program Detailed Flow Chart (2 of 2) .....	61

# List of Symbols

## Variables

Flapping Angle	$\beta$
Vortex Strength	$\Gamma, \gamma(r)$
Blade Element Pitch Angle	$\theta$
Pitch Attitude, Euler	$\theta$
Inflow Ratio	$\lambda$
Advance Ratio	$\mu$
Air Density	$\rho$
Roll Attitude, Euler	$\phi$
Yaw Attitude, Euler	$\psi$
Rotor Rotational Velocity	$\Omega$
Area	A
Blade Count	b
Blade Chord	c
Drag Force	D
Center of Gravity	cg
Lifting Force	L
Pitching Moment	m
Mass, Apparent	M
Position Vector	P
Rotational Velocities	p, q, r
Rotor Radius	R
Rotor Station	r
Thrust of Rotor	T
Translational Velocities	u, v, w, U, V, W
Non-Dimensional Rotor Station	$x=r/R$
Axis Directions	x, y, z



## Subscripts

Aero/Wind Axis	a
Aerodynamic, Mast Axis	am
Body Axis	b
Blade Rotated Axis	br
Blade Flapped Axis	bf
Drag	d
Inertial Axis	e
Hub Rotating Axis	hr
Lift	l
Fixed Mast Axis	m
Pitching Moment	m
Perpendicular Component	p
Rotor Tip	t
Tangential Component	t
Total	tot
Wash Velocity	w
Freestream Value	$\infty$

## CHAPTER 1. Introduction

In pursuit of simulating and analyzing rotorcraft characteristics many engineers and scientists have created mathematical models to replicate the physics demonstrated by rotorcraft. Such models are used to capture the power required, induced velocities, thrust capability, blade motions and other necessary results to aid the development and design of rotorcraft. These models have grown in capability as more advanced methods and technologies have become available for use. Early analytical models derived from the fundamental laws of physics were developed but required making assumptions to make the equations tractable. Advancements on these analytical models have been formed over the years based on new methods and capabilities as well as refinements to the assumptions and inclusion of correction factors learned through test data. These refined analytical models can provide macroscopic level results that are useful for many practical scenarios. The assumptions made to allow the engineer to form solutions from the analytical models also restrict the range of applicability for the model. Later models incorporate higher computing power to solve discrete forms of equations based on the fundamental laws of physics that the analytical models are based off, but without making all the same assumptions previously required. These newer math models can be used for very precise results with high fidelity. The higher fidelity comes at the cost of higher computation power which forces some engineers to resort to intermediate models based on the level of fidelity required.

Still further engineers have pieced together high-fidelity models with more fundamental models to create robust answers at a cost of complexity to the user. The highest fidelity models to date still require more computational power than available to operate them in real time simulation. For this thesis a focus on analysis models is chosen in pursuit of developing a high or moderately high general-purpose model for use in development and analysis efforts. First, we review the models and identification of the assumptions and the range of use or capabilities of each. Then we discuss the development of these models and the options that are available within the confines of their defined

domain. The collection of models is not an exhaustive list but rather capture the primary types as created and used over the course of rotorcraft history. Following the background of rotorcraft models, a description of the model utilized in the efforts of this thesis is used with an emphasis on the rotor and rotor blade wake velocities. This description of the model is used as a premise for the development of the project rotor and wake model with the intent for them to be general purpose models applicable for a wide range of uses and to provide the user with performance predictions, wake characteristics, and the effects on rotor blade motion. This model is compared with results from other accepted models as well with test data to show validity or areas requiring improvement. This leads into the final phase of the project with the future efforts that can and should be performed to advance the rotorcraft simulation and analysis field.

## CHAPTER 2. Background of Rotor Models

### 2.1 Momentum Theory Models

#### 2.1.1 Basic Momentum Theory Model

The most basic rotor model utilizes a stream tube with varying wall diameter but is coincident in diameter with the rotor at the rotor plane. The fluid flow through the stream tube is only quasi-axial with velocity that only changes in the axial direction but is uniform velocity along the radius and around the azimuth of any cross-section of the stream tube. At the rotor plane, an actuator disk exists to represent the rotor and is composed of an infinite number of blades such that the force imparted on the fluid is also uniform. The stream tube diameter varies based on conservation of mass flow such that, assuming constant density, with a change in the velocity of the flow the diameter must expand or contract. The radially and azimuthally uniform force that the actuator disk imparts on the flow creates a change in kinetic energy and is used to derive a relationship between the thrust of the rotor and the induced velocity. Glauert's hypothesis was formed using this principle and creates a relationship of the axial flow to the uniform induced velocity. The expression below shows how thrust can be calculated from the induced velocity, where  $V$  is the axial velocity of the rotor,  $T$  is the thrust, and  $w_w$  is the induced velocity.

$$w_w = \frac{T}{2\rho A_{disk}(V + w_w)} \quad 2-1$$

Furthermore, the power required to create the given thrust can be formed as the product of the thrust and the velocity. Utilizing an efficiency factor for hovering rotors called the Figure of Merit a total power required can be calculated to encompass the non-uniform inflow effects, swirl velocity effects, root cutout, tip loss effects, and profile power. From this basic model the induced velocity and power required can be determined. This model is good for axial flows such as hover and utilizing some

empirical or estimates of efficiency can be used for very simple sizing predications but does not capture the effects of edgewise flight nor does it predict any blade motion. Basic Momentum Theory also does not allow for relating blade pitch, collective or cyclic, to the thrust nor does it consider the blade chord and airfoil properties which are all important details for analysis and design. [1]

### 2.1.2 Blade Element Momentum Theory Models

To improve the fidelity of Momentum Theory rotor models, an alternate approach can be made by looking at elemental sections of the blade and deriving a relationship of the basic airfoil aerodynamics to the thrust and power required. This is done by forming the angle of attack on an airfoil section through the parameters for geometric pitch angle and section inflow angle. From the angle of attack and known airfoil properties a drag and lifting force expression can be derived. Integration along the blade radius and around the rotor azimuth results in a thrust and torque required. These expressions can be formed analytically and are good for incorporating more blade and rotor geometric properties; however, the inflow still must be determined implicitly using the basic Momentum Theory thrust to induced velocity relationship as expressed in equation 2-1.

Inclusion of the components of the rotor inertial velocity allow for derivation of power and thrust expressions in forward flight. Expanding on these expressions, the blade motion can be expressed by integrating the product of blade spanwise forces and radial locations or blade moments as a function of azimuth. Then assuming a harmonic solution, the blade flapping can be derived analytically to include the steady or coning angle and the cyclic motion of blades which is commonly referred to as a quasi-static rotor model. These analytical expressions are good for static analysis based on uniform Momentum Theory inflow but do not account for local, non-uniform inflow angles.

### 2.1.3 Improved Momentum Theory Models

Further improvements to the Momentum Theory rotor inflow model are accomplished by forming a time variant differential equation using an apparent air mass, derived from potential flow theory, upon which the rotor imposes its thrust. Pitt and Peters have developed a widely used version known as the dynamic inflow model [2] which has an axial induced velocity as well as a pitch and roll swirl velocity. The pitch and roll swirls improve the rotor flapping correlation. The Pitt and Peters dynamic inflow model can also be used for real-time simulations and is flexible enough to increase fidelity based on users desired number of states. Other variations of the Pitt-Peters' dynamic inflow model have been developed to better correlate the rotor to and are based on variations in rotor operating conditions. However, these models are still based on Momentum Theory and as such limits the local flow dynamics and is still only applicable to the same regions as basic Momentum Theory.

## 2.2 Algebraic Rotors

For real time simulations a different rotor model can be formed that utilizes a predetermined or correlated data set to return resulting conditions based on inputs. The backbone of this model is similar to the quasi-static model but differs because the algebraic model relies on table lookups of correlated data. This type of model uses other sources to build up the data set before simulation so that it does not have to solve for results but rather just calculate algebraically what the results are for that moment in time. Typical models like this input linear and angular rates at the mast center with the control inputs, collective and two cyclics, to return the forces and moments as well as the first harmonics for lateral and fore/aft flapping. Algebraic models are useful for real time and analysis simulations but are only accurate in the regions where the data set is correlated. Even with high levels of correlation the wakes from these models are either prescribed, rigid stream tubes or altogether ignored. Although these models are highly useful for real-time simulation to engineers and for training purposes, they lack

the fidelity for in depth analysis and are restricted to the range of applicability of the data set that the model is based.

### 2.3 Blade Element Rotor (BER) Models

A blade element method was described previously to calculate analytical expressions for rotor thrust, power required and blade motion. The type of BER discussed here is similar but instead of solving for analytical expressions the elemental segments remain as discrete components to capture dynamic changes in local conditions. This discrete form BER makes it suitable for numerical implementation. This discrete blade element model divides the rotor and the blade into elemental segments and uses two-dimensional airfoil data to determine the forces and moments acting on each segment. Each blade elements' contributions are then summed and resolved to a common rotor hub location to determine the resultant forces and moments of the rotor. BERs can be very useful to analyze and design rotors by modeling the physical effects of mechanical devices of real rotors (flapping hinges, lead-lag hinges, offsets, etc.). Kinematics and dynamic properties can be included to simulate the physical motion of each blade and even each blade segment. BERs can be quite robust in modelling the physical dynamics by applying mechanical properties to the blade sections and allowing the differential forces and moments to influence the blade motion. A prescribed set of normal mode shapes can be implemented and coupled with a dynamic solver to determine the modes which define the final shape of the blade at every instance. Even with all the mechanical modeling of the physical location and shape of the blade, BERs are still limited by the aerodynamics model and must be tied to an inflow or wake model, which will be discussed later. Computational power varies with complexity of the BER, but generally is greater than algebraic or Momentum Theory models making it less ideal for simulation trainers or some real-time applications. Because of the capability to model hinges or flexures, and other mechanical components, blade element models enable the users to model, with greater detail, the mechanism of the rotor and thus enables analysis of the various configurations of many rotor systems making it a

robust model choice. The blade element model is a strong platform for detailed analysis and has a wide range of uses and application. The coupling of the BER with a wake or inflow model further defines its true range of applicability.

## 2.4 Prescribed Wakes

### 2.4.1 Stream Tube Prescribed Wakes

Algebraic and some momentum theory models may use a prescribed stream tube to model the wake structure. The tube may be skewed with advance and inflow angles but remains rigid, extending away from the rotor towards infinity. The velocity along the axis of the stream tube can be varied using the conservation of mass to create better downwash effects but the velocities are dependent on the rotor inflow model. This rigid stream tube can be useful for getting gross effects of wake interferences on other nearfield components but since the wake is rigid it can wrongly portray the wake conditions and structure at larger distances from the rotor plane. The stream tube is analogous to a laser light fixed to the mast of the rotor pointed down and as the rotor/aircraft moves the laser is projected to infinity downstream whereas a real wake is malleable.

### 2.4.2 Helical Prescribed Wakes

An intermediate approach between the rigid stream tubes and the forthcoming free-wakes are helical prescribed wakes. Helical prescribed wakes use a rigid helix extending from the rotor at an angle determined from the ratio of inflow and advance ratio, or skew angle. The prescribed helical structure is similarly rigid like the stream tubes mentioned previously. The prescribed helical path is then used along with vortex theory or other means to determine the wake characteristics. The helix is limited to the geometry prescribed and thus the contraction of the wake is not adaptive to the rotor conditions. Egolf and Landgrebe describe use of a prescribed wake model for use in predicting the inflow at the rotor and for wake flow field analysis [3]. The model they describe uses the advance ratio, coefficient of



thrust and the angle of attack of the tip path plane to determine the prescribed helical shape of vortex filaments that are emitted from the rotor blades.

Harris, utilizing a lifting line blade created an analytical method for the induced velocity using a conventional wing that is flying in a very tight circle [4]. Harris' simplified analytical model builds up the generalized solution for self-induced velocities, but the wake structure is still rigid, it does not create the horseshoe structure in forward flight, and it is restricted to static analysis. The prescribed wake uses primarily trailing vortices emitted along the span of the rotor blade, but a prescribed wake can also have shed vortices emitted parallel to the rotor blade span to increase the fidelity of the wake model.

#### 2.4.3 Ribbon Prescribed Wake

A special version of a prescribed wake utilizes the trailing and/or the shed vortex filaments for high-speed forward flight. With high enough speeds, advance ratios greater than about 0.4, the wake is assumed to lie on a single plane trailing the rotor. The resulting two-dimensional wake, referred to as a 'ribbon' by Johnson [5], can then be used like the other wakes to determine induced velocities on the rotor and surrounding surfaces. The ribbon prescribed wake is limited to range of use and would require a separate wake model to be incorporated for slower airspeed rotor modeling.

## 2.5 Free-Wakes

### 2.5.1 Computational Fluid Dynamics

Computational fluid dynamics (CFD) utilizes numerical methods to calculate the airflow through a discretized volume. The differential form of the Navier-Stokes or a simplified version is used along with a set of initial and boundary conditions to calculate the air properties in each finite volume. CFD iterates along space and time to determine the flow field based on the geometry and motion of the rotor blades within the flow field. The flow field calculations are not necessarily dependent on the blade dynamics and a prescribed motion may be used to determine the flow field and thus wake structure. The

numerical complexity and computational expense are large enough but then to add in the coupled blade motion would require another dimension of iterations. With the expensive computation demand comes the benefit of being able to solve for high fidelity interactions with other components such as other rotors and rotor blades, fuselage, and ground surfaces. There are multiple options within CFD that allow for optimization and refinement based on a multitude of input parameters. Such complexity requires the user to properly calibrate and correlate the inputs to get accurate results. Another drawback of CFD free-wake models is a result of the high computational demand that requires large amounts of computation time and thus obviates real-time simulations. CFD in general is good for determining the flow field properties such as wake structure, including vortices, as well as non-uniform induced velocities; however, unless coupled with a blade motion solver the overall purpose of CFD is simply to analyze the wake and not necessarily the rotor behavior.

## 2.5.2 General Vortex Theory and the Application of Biot-Savart Law in Aerodynamics

Another method to represent the velocity field in and around a rotor uses vortices. This can be done by various implementations but boils down to defining a vortex filament and assigning it a circulation strength and then using the Biot-Savart Law to calculate the induced velocity at given points. One of the early references to using vortices in fluids was from Maxwell [6] who proposed the proportional relationship of the velocities and the ratio of density. From this proposition, he derived a means to determine the tangential velocity along the circumference of a vortex.

Explanation of vortex theory begins with looking at a 2-dimensional flow encircling a point. Assume the flow is inviscid, incompressible and irrotational. The spinning fluid is defined as having no radial velocity and a tangential velocity of a constant radius. The circulation about a circle centered on the point reveals the relation of the tangential velocity to the vortex's strength.

$$\Gamma = \oint_C \mathbf{V} \cdot d\mathbf{s} = V_\theta(2\pi r) \quad \rightarrow \quad V_\theta = \frac{\Gamma}{2\pi r} \quad 2-2$$

(Note: Vectors are represented with bold font, scalars with regular font.) The Kutta-Joukowski theorem states that the lift per unit span is the product of the strength of the vortex with freestream velocity and density. This is derived by summing a series of vortices to form a vortex sheet around an arbitrary shaped airfoil. The Kutta-Joukowski theorem is applicable to thin airfoils and higher Reynold numbers where the viscous effects do not dominate the lifting force of the flow. The practical use of this theorem is valid because the use is focused only on the lifting force and the resulting vorticity that it creates.

$$L' = \rho_{\infty} V_{\infty} \Gamma \quad \rightarrow \quad \Gamma = \frac{L'}{\rho_{\infty} V_{\infty}} \quad 2-3$$

Expand the theory into 3-dimensions where the point filament is now a line in space and carries a constant value for strength of the vortex. The Biot-Savart Law can be applied to determine the velocity at any arbitrary point in space that is created by that vortex. The Biot-Savart Law in differential form is shown below where  $\gamma$  is the vortex strength along the directed differential segment,  $ds$ , and the line from the point of interest to the filament,  $r$ .

$$d\mathbf{V} = \frac{\gamma}{4\pi} \frac{\mathbf{r} \times d\mathbf{s}}{|\mathbf{r}|^3} \quad 2-4$$

For a filament that extends along a straight line to positive and negative infinity, the Biot-Savart Law can be simplified to a scalar equation as a function of the vortex strength and the perpendicular distance  $h$  to the point of interest.

$$V = \frac{\Gamma}{2\pi h} \quad 2-5$$

For a straight-line filament of finite length and known end points, the velocity at a point of interest is modified slightly. Since the filament is a straight line, the velocity produced by the vortex is perpendicular to the filament, simplifying the Biot-Savart expression to a scalar equation.

$$dV = \frac{\Gamma}{4\pi} \frac{ds \sin(\theta)}{h^2} \quad 2-6$$

Using geometry to simplify and parameterize as a function of the interior angles  $\alpha$  and  $\beta$  then integrating across the segmented angles results in an equation for the finite segment of filament. As a check one can take the limit as the ends of the segment extend to positive and negative infinity respectively where the angles  $\alpha$  and  $\beta$  approach zero. This results in the same equation for the infinitely long filament described earlier in equation 2-5.

$$V = \frac{\Gamma}{4\pi h} [\cos(\alpha) - \cos(\pi - \beta)] \quad 2-7$$

Now we can resolve a segment of lift into a vortex strength of a segment of filament and we can use that strength and filament to calculate the velocity at any point of interest in the field. The exception here is when you approach the filament; the velocity will tend to infinity. To overcome this a piecewise equation can be adopted based on the parameter of the filament's core radius. Figure 1 shows an example of a vortex filament pointed out of the page and the induced velocity along the horizontal axis extending from the origin of the vortex. The induced velocity is directed down on the left and up on the right. As the radius approaches the center of the vortex the velocity increases asymptotically and is singular at the center as represented by the solid line outside the core radius and dashed inside the core radius. By assuming some core radius, the singularity can be avoided by connecting the two sides linearly through the center of the vortex as developed by Scully. This linear velocity assumption represents a change from the irrotational flow outside the core to a rotational core that is slowed due to

viscous effect and is zero at the center of the vortex. The reduction of velocity in the core is physically comparable to the eye of a hurricane. The size of the core's radius depends upon the application of the vortex and according to Young is primarily a function of three parameters; rotor thrust, rotor efficiency and advance ratio [7]. Young also creates several parametric equations to incorporate into a model but concludes that the core radius is a subject of trial and error to tune model performance.

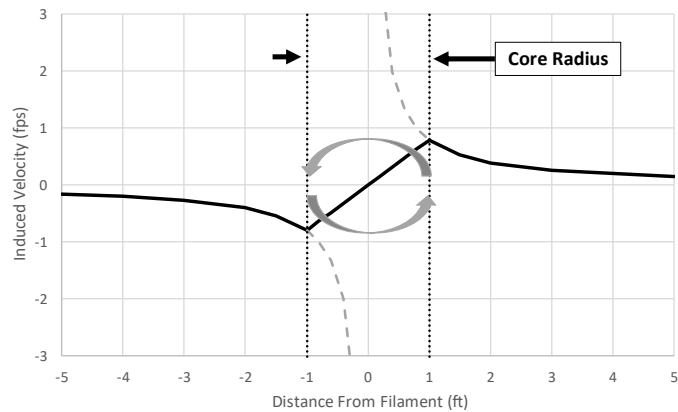


Figure 1. 2D Example of Vortex Induced Velocity with Core Radius

Now applying the vortex theory to a wing, let's look at the types of vortices produced by a lifting surface. The main types of vortices of concern for vortex theory are the bound, shed and tip. A bound vortex represents the rotational flow field around the airfoil and is "bound" to the physical space of the vortex producing structure. The counter to the bound vortex is the shed vortex (sometimes called a starting vortex), which rotates in the opposite direction than the bound vortex and keeps the laws of conservation satisfied. The tip vortices emanate from the ends of the bound vortex and trail away, typically behind the lifting surface, where it connects with the shed vortex. For an isolated wing the vortices would form a rectangular box comprised of the bound vortex, two tip vortices from the wing tips and a shed vortex.

Helmholtz's vortex theorems are required when laying out a vortex theory model. They state that 1) the vortex strength is constant along the length of the filament, 2) a filament cannot end in a fluid and must end at a solid boundary or extend to infinity and 3) the irrotational fluid remains irrotational unless acted upon by an external force.

### 2.5.3 Vortex Ring Models

A simple implementation of vortex theory is the application of a vortex ring that is emitted from the edge of the tip path plane (TPP). The TPP is the theoretical flat disk-shaped plane defined by the sweep of the flapped rotor blade tips. To conceptualize this model, take a perspective view from behind the rotor and only look at the plane bisecting the rotor. As each blade whirls around the azimuth and passes through the plane, it emits a ring that is a closed vortex filament. The filament is given a vorticity strength by means of the Kutta-Joukowski theorem and is allowed to move based on the induced velocities from other vortices by means of the Biot-Savart Law. Figure 2 shows the wake formed by a rotor using a vortex ring model where the dots represent cross-sections of vortex rings, the thin solid lines depict the segmented axially induced velocity over the blade scaled by one-tenth, and the thick dashed curve is the distributed lifting force on the blades scaled by one-fiftieth. The large torus at the bottom of the wake is a consequence of its impulsive start that compiles at the start of the simulation then propagates downward as the wake structure forms. At each time step the summations of all induced velocities from all vortices are applied to each other and without any ad hoc rules the vortex ring model shows the wake contraction as predicted in momentum theory. Brand, et al. used a vortex ring model to layout the fundamentals of vortex ring state (VRS), which is a phenomenon that occurs when the rotor is descending with the induced velocity and the vortex rings [8]. In VRS, as the descent rate increases the vortex rings begin to coalesce and eventually envelope the TPP forming a torus. When in this state the rotor is re-circulating its own downwash, which in turn decreases the effective angle of attack at all blade stations and thus reduces thrust. The result is that the rotor begins

descending more rapidly and can result in significant loss of thrust. Vortex ring models are restricted to axial flow flight due to the simplification of a ring rather than a helical vortex filament but are physically based and thus still able to replicate the characteristics of VRS.

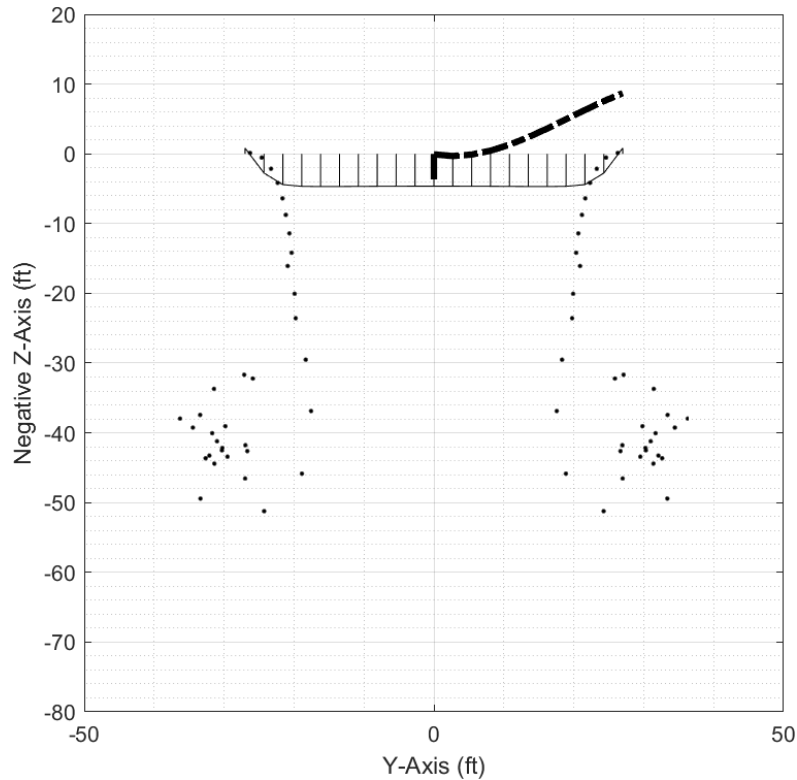


Figure 2. Vortex Ring Model Display in Y-Z Plane

#### 2.5.4 Particle Vortex Transport Method Wake

Particle Vortex Transport Method (PVTM) is a type of vortex-theory free-wake model that utilizes a volume mesh to measure all the deposited vorticity and then calculate the induced velocity from each grid to all the other grids in the mesh. A CFD or similar model solves for the velocity field immediately surrounding the blade and passes the velocity field to the PVTM. The vorticity of each grid in the PVTM is determined by integrating the velocity field over the volume of the grid. Like other vortex models the incremental induced velocity resultant from each grid is calculated using the Biot-Savart Law and then added to the total velocity field. As the wake develops the vorticity is not tied to a filament but rather is

propagated through the evolving velocity field allowing for highly flexible wake structures. Reference [9] uses the PVTM coupled with CFD to reduce the computational time while still achieving high blade loading results. Because PVTM methods uses an organized mesh, far field regions can be combined easily using Fast-Multipole or Barnes-Hut type algorithms. PVTM in general is a good compromise of fidelity and computation speed.

#### 2.5.5 Free Vortex Wake

A Free Vortex Wake (FVW) model is a compilation of vortex filaments that are emitted from a rotor that create a flow field around the rotor and simulates the induced velocity created by the rotor. The filaments are typically discretized into straight line filaments for numerical purposes and are emitted from specified blade span locations. Reference [10] uses a Free Vortex Wake model that emits a vortex filament near the tip of the blade. The bound vortex is assigned a strength determine by using the section lifting force and relating it to the strength using the Kutta-Joukowski theorem. In a similar fashion a blade tip emitted filament is assigned a vortex strength that is a function of the total bound circulation. The bound vortices are not typically emitted but the tip vortex is emitted and allowed to move freely afterwards. At each time step the influence on the velocity field is calculated from each vortex filament. This velocity is also calculated along the blade span which is returned to the blade aerodynamic model for the next iteration or time step. The spatial freedom that the vortex filaments have permit them the ability to simulate natural phenomena such as vortex tip rollup and wake contraction. This increased freedom comes at a computational cost since the velocity field calculations are an N-body problem.

Because FVW models are not dependent on momentum theory they can be used in a wider range of flight conditions without any special rules or empirical corrections like CFD models. FVWs do benefit in reduced computational requirements as compared to CFD solely based on the reduced number of nodes



required for calculation. The tradeoff between CFD and FVWs is the computational time and the fidelity. CFD are as precise as the nodal mesh defined for the problem which can allow it to simulate even small flow characteristics around the blade components. FVWs are restricted to the fidelity of the underlying lifting surface aerodynamic model.

## CHAPTER 3. Project Rotor Model

### 3.1 Blade Element Rotor (BER) Model

#### 3.1.1 Description of Model Selection

Based on the research of the general types of rotor and inflow/wake models, a blade element rotor model (BER) is selected for its capability for inclusion of blade dynamics, discrete or virtual mechanical joints, and calculation of distributed aerodynamic loads on the blade/rotor. A vortex-theory free-wake model is also selected for the ability to simulate wake deformation and dynamic characteristics without extensive computational resource demand. Two versions of the vortex-theory free-wake are developed to evaluate and compare the difference in the assumptions made for each model.

#### 3.1.2 Rotor Parameter Setup

The Blade Element Rotor model built for this project is an isolated rotor with  $n$ -blades composed of  $m$ -radial stations along the blade span. The rotor modelled is an articulated configuration with a flapping hinge at the mast centerline. Each of the blades are modeled as being infinitely rigid except that they are free to flap about the flapping hinge defined at the root of the blade. The rotor rotates with prescribed rotational speed and currently only rotates counterclockwise. The distributed mass and twist are also predefined with the uniform distributed mass,  $dm$ , being used to determine the flapping inertia,  $I_b$ .

$$I_b = \frac{dm R^3}{3} \text{ [slug ft}^2\text{]} \quad 3-1$$

To improve the fidelity of this simple BER model a hub spring is installed with stiffness value that can be adjusted to give a desired non-dimensional flapping frequency,  $P$ .

$$P^2 = 1 + \frac{k_\beta}{I_b \Omega^2} [n/d] \quad 3-2$$

The Blade Reference Line, BRL, is used to represent the lifting line of the blade and is illustrated in Figure 3. The Blade Reference Line has an axis system with the x-direction pointed from the radial position of interest towards the mast, the y-direction points forward parallel and co-linear with the airfoil chord line, and the z-direction completes the right-hand rule pointing down past the lower surface of the airfoil. The BRL is defined with dimensional blade stations that are equally spaced along the span starting from the root moving out to the tip of the blade with values ranging from zero to R, the blade radius. These stations represented as the vector  $\mathbf{sa}$  are the locus of quarter chord lines of the airfoil sections. The rotor is defined with a finite number of radial stations,  $nrs$ , and each is identified with an index,  $irs$ . For an untwisted blade, each station's y-axis points in the same direction; however, due to geometric blade twist each blade section axis is rotated about the x-axis to correspond to its proper geometric positions. Furthermore, the blade is presumed to be rigid in torsion; the blade section pitch is rotated by cyclic and collective inputs through a pitch bearing and each section rotates by an equal amount.

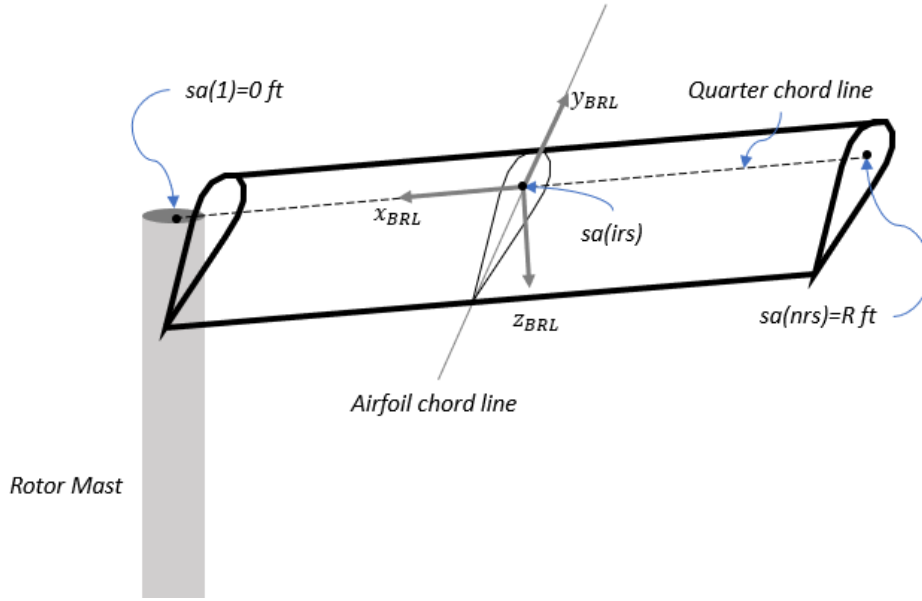


Figure 3. Blade Reference Line Illustration

Within the context of this model the blade reference line can only be rotated out of the plane of rotation through the flapping angle  $\beta$  and we define a position vector,  $P$ , representing the blade section flapped position along the radius. Along with the position, the time derivative of  $P$  or flapping velocity,  $\dot{P}$ , is also required.

$$P(x) = T_{\beta} sa(irs) = \begin{Bmatrix} -sa(irs) \cos(\beta) \\ 0 \\ -sa(irs) \sin(\beta) \end{Bmatrix} \quad 3-3$$

$$\dot{P} = \begin{Bmatrix} sa(irs) \sin(\beta) \dot{\beta} \\ 0 \\ -sa(irs) \cos(\beta) \dot{\beta} \end{Bmatrix} \quad 3-4$$

### 3.1.3 Rotor Axes and Transformations

To model the forces and moments acting on the blades we start from the wide view in the inertial or earth reference frame with the inertial velocities and then orient them through the Euler angles to

arrive at the body axis. The simplification was made to have all the Euler angles remain equal to zero but can easily be added as the model matures.

$$\mathbf{V}_b = T_b \mathbf{V}_e = R_x R_y R_z \mathbf{V}_e \quad 3-5$$

$$\boldsymbol{\omega}_b = \mathbf{E}(\boldsymbol{\Phi}, \boldsymbol{\Theta}) \boldsymbol{\omega}_e \quad 3-6$$

where the  $R_n$  are rotational matrices about their respective axes, x, y, z. The velocities in the Earth x,y,z axes are represented with the subscript “e”. The  $\mathbf{E}$  represents the angular velocities rotation matrix. The resulting body axis velocities, marked with subscript “b”, are then resolved from the aircraft center of gravity (CG) to the hub axis system; again, the simplification was made to have the aircraft CG and hub center be coincident, but for completeness the process is shown below.

$$\mathbf{V}_m = \mathbf{V}_b + \boldsymbol{\omega}_b \times \mathbf{P}_{cg} \quad 3-7$$

$$\boldsymbol{\omega}_m = \boldsymbol{\omega}_b \quad 3-8$$

The resolved inertial velocities at the top of the rotor mast, represented with the subscript “m”, are assumed to be uniform over the whole rotor disk area. Moving along the radius of the blade, the hub fixed mast axis velocities are combined with the induced velocities, denoted with subscript “w”, to resolve the velocities into the aerodynamic or wind axes of the fixed mast. The aerodynamic velocities are then rotated to get into the rotating hub axis system, denoted with subscript “hr”, by rotation through the combined blade rotor azimuth and blade spacing angles. This process is done along the blade span and shown here as x which represents the non-dimensional radius and is equivalent to  $r/R$ .

$$\mathbf{V}_{am}(x) = \mathbf{V}_m - \mathbf{V}_w(x) \quad 3-9$$

$$\boldsymbol{\omega}_{am}(x) = \boldsymbol{\omega}_m - \boldsymbol{\omega}_w(x) \quad 3-10$$

$$\mathbf{V}_{hr}(x) = T_\psi T_\delta \mathbf{V}_{am}(x) \quad 3-11$$

$$\boldsymbol{\omega}_{hr} = T_\psi T_\delta \boldsymbol{\omega}_{am} \quad 3-12$$

Now the rotating hub velocities need to be resolved to the blade rotating axis system, denote with subscript “br”, through the position vector,  $\mathbf{P}$ , and the kinematic velocity,  $\dot{\mathbf{P}}$ , which contains the flapping angle,  $\beta$  and lead-lag angle,  $\gamma$ ; however, for this simple model lead-lag is ignored and assumed to be zero. Along with the kinematic velocity and angular rates crossed with the position vector, the hub velocities are summed to get the blade flapping axis velocities, denoted with subscript “bf”. The rotation matrices,  $R_{bx}, R_{by}, R_{bz}$ , rotate the velocities into the blade flapped axis through the blade flapping angle, lead-lag angle, and the total pitch angle. The flapping angle was used to rotated about the BRL y-axis.

$$\mathbf{V}_{br}(x) = \mathbf{V}_{hr}(x) + \boldsymbol{\omega}_{hr} \times \mathbf{P}(x) + \dot{\mathbf{P}}(x) \quad 3-13$$

$$\mathbf{V}_{bf}(x) = R_{bx}(\theta_{total})R_{by}(-\beta)R_{bz}(x)\mathbf{V}_{br}(x) \quad 3-14$$

The total pitch of the blade section is composed of two components, the geometric twist, and the cyclic control input. The geometric twist is linear in this model and tabulated at each blade segment. The cyclic pitch input was constant along a single blade and is determined using the equation below where  $\psi$  is the azimuth location,  $A_1$  is the lateral control input,  $B_1$  is the longitudinal control input, and  $\theta_0$  is the collective control input.

$$\theta_{total} = -\theta_{twist} - \theta_{cyclic} \quad 3-15$$

$$\theta_{cyclic} = \theta_0 + A_1 \cos(\psi) + B_1 \sin(\psi) \quad 3-16$$

$$\theta_{twist} = \theta_{twist\ rate} \frac{r}{R} \quad 3-17$$

The y-component of the blade flapping axis velocity is the tangential component mentioned previously along with the z-component being the perpendicular velocity. The arctangent of the two components is used to determine the local angle of attack of the blade section.

$$U_t(x) = \mathbf{V}_{bf}(2, x), \quad U_p = \mathbf{V}_{bf}(3, x) \quad 3-18$$

$$\alpha(x) = \tan^{-1} \left( \frac{U_p(x)}{U_t(x)} \right) \quad 3-19$$

$$q_{tot}(x) = \frac{1}{2} \rho (U_t^2(x) + U_p^2(x)) \quad 3-20$$

### 3.1.4 Rotor Forces and Moments

From the tangential and perpendicular velocity combined with the angle of attack the lift, drag and pitching moment are determined for each radial location. Several options are available to the user. One is to use linear aerodynamics where the lift is calculated using a linear lift curve slope and angle of attack and an average drag and pitching moment coefficient value are maintained. A second option uses a table look up for the section properties of lift, drag, and pitching moment providing non-linear aerodynamic capability. A third option uses a compact set of analytic expressions that fit the tabular data. The non-linear option permits the coefficients to be a function of Mach number, whereas the other two options do not currently. This thesis used the tabulated data for a NACA 0012 airfoil. The aerodynamic table is based on high Reynolds number values and is sufficient for full size rotors. Scaling down the rotor to subcritical Reynolds numbers would require an updated aerodynamic table set. The resulting forces and moments are assigned to the appropriate vectors to be transformed to the hub.

$$L' = C_L(\alpha)q_{local}c_e \quad D' = C_D(\alpha)q_{local}c_e \quad m' = C_m(\alpha)q_{local}c_e c_e \quad 3-21$$

$$d\mathbf{F}_a(x) = \begin{pmatrix} 0 \\ -D' \\ -L' \end{pmatrix} \quad 3-22$$

$$d\mathbf{M}_a(x) = \begin{Bmatrix} -m' \\ 0 \\ 0 \end{Bmatrix} \quad 3-23$$

The differential forces and moments are then rotated back through the negative of the angle of attack and into the blade flapping axis using the transpose of the previously used rotation matrices.

$$d\mathbf{F}_{hr}(x) = \left( R_{bx}(x)R_{by}(x)R_{bz}(x) \right)^T R_{(-\alpha(x))} d\mathbf{F}_a(x) \quad 3-24$$

$$d\mathbf{M}_{hr}(x) = \mathbf{P}(x) \times d\mathbf{F}_{hr}(x) + d\mathbf{M}_a(x) \quad 3-25$$

Then the forces and moments are integrated along the BRL to get the total before being rotated back through the azimuth angle into the mast axis system. The integration occurs discretely using a trapezoidal integration as shown below where  $x$  is replaced with the discrete indices,  $irs$ , along the BRL as predefined and the total number of radial positions,  $nrs$ . Recall, the variable  $sa$  represents the dimensional position along the BRL.

$$\mathbf{F}_{hr} = \sum_{irs=2}^{nrs} \frac{d\mathbf{F}_{hr}(irs) + d\mathbf{F}_{hr}(irs - 1)}{2} * (sa(irs) - sa(irs - 1)) \quad 3-26$$

$$\mathbf{M}_{hr} = \sum_{irs=2}^{nrs} \frac{d\mathbf{M}_{hr}(irs) + d\mathbf{M}_{hr}(irs - 1)}{2} * (sa(irs) - sa(irs - 1)) \quad 3-27$$

With the resultant forces and moments transformed and summed into the mast rotating axis, the next step is to resolve them into the mast fixed axis through the blade azimuth and blade separation angles.

$$\mathbf{F}_m = (T_\psi T_\delta)^T \mathbf{F}_{hr} \quad 3-28$$

$$\mathbf{M}_m = (T_\psi T_\delta)^T \mathbf{M}_{hr} \quad 3-29$$



Finally, the forces and moments need to be resolved back to the center of gravity using the following equations.

$$\mathbf{F}_{cg} = \mathbf{F}_m \quad 3-30$$

$$\mathbf{M}_{cg} = \mathbf{M}_m + \mathbf{P}_{cg} \times \mathbf{F}_m \quad 3-31$$

The calculated forces and moments are compared against the force and moments required which for this simplified model are represented by the weight, forward drag, and side drag. The weight,  $W$ , assumes no download or other interferences and is placed into the force required vector with proper sign. The forward and side forces that are required utilize the user defined flat plate drag areas. This simplified model assumes no moments were required and that all forces were acting on the center of gravity.

$$\mathbf{F}_{required} = \begin{Bmatrix} f_{e_x} q_{\infty} \\ f_{e_y} q_{\infty} \\ -W \end{Bmatrix} \quad 3-32$$

$$\mathbf{M}_{required} = \begin{Bmatrix} 0 \\ 0 \\ 0 \end{Bmatrix} \quad 3-33$$

### 3.1.5 Rotor Controls and State Variables

A simple set of control laws were derived to drive the rotor towards force balance using the control input vector. The control vector consists of the three available inputs to conventionally controlled rotors and are the collective,  $\theta$ , lateral cyclic,  $A_1$ , and longitudinal (or fore/aft),  $B_1$ .

$$\mathbf{cont} = [\theta, A_1, B_1] \quad 3-34$$

To track the blade's state a state variable vector,  $sv$ , is used that contains each blade's flapping angle,  $\beta$ , and flapping rate of change,  $\dot{\beta}$ , as well as the momentum equivalent induced velocity,  $w_w$ . As the blade

dynamics are calculated and the resulting change in position and rates are determined they are returned in the derivative state variable vector,  $\mathbf{svd}$ , that holds the same flapping rate of change just advanced to a new index along with each blade's calculated flapping acceleration,  $\ddot{\beta}$ . The rotor induced velocity rate of change,  $\dot{w}_w$ , is not used for the free-wake but does calculate a new induced velocity for the momentum theory option for the model using the equation below.

$$\mathbf{sv} = [\beta_1, \dot{\beta}_1, \beta_2, \dot{\beta}_2, \dots, \beta_{nb}, \dot{\beta}_{nb}, w_w] \quad 3-35$$

$$\mathbf{svd} = [\dot{\beta}_1, \ddot{\beta}_1, \dot{\beta}_2, \ddot{\beta}_2, \dots, \dot{\beta}_{nb}, \ddot{\beta}_{nb}, \dot{w}_w] \quad 3-36$$

$$m_{app} = \frac{4\pi R^3 \rho}{3} \quad 3-37$$

$$\dot{w}_w = \frac{-F_{mast_z} - 2\rho AV_{total} w_w}{m_{app}} \quad 3-38$$

### 3.1.6 Rotor Model Process

Once the initial conditions are established the primary model processing can be performed. The primary process is centralized around a rotor rotation loop that is tied directly to time, under the assumption that rotor rotation rate is constant. Each iteration steps forward a single azimuth step which specifies each blade's rotational position,  $\psi$ . A simple control law is used for root collective, lateral and longitudinal cyclic commands. This control vector along with the state variables are passed to a subroutine that calculates the individual blade forces, moments, flapping rate, and flapping acceleration as previously describe. The newly calculated state variable derivative is used in a Corrected Euler integration scheme that uses the first state variable derivative to determine the next iteration's state variable then repeats the individual blade forces, moments, flapping rate, and flapping acceleration calculation to get a corrected state variable derivative. The two state variable derivatives are averaged and then used to determine a corrected state variable to continue through the process. This integration scheme is often referred to as an Improved Euler or two-step Runge-Kutta (RK2). The momentum

theory wake model calculates an induced velocity derivative using the apparent mass of the air moving through the rotor, current thrust, and induced velocity before it is loaded into the state variable for the next iteration. The free-wake model utilizes the methods described later to form new induced velocities for the blades at the next iteration. Once the specified number of revolutions are completed the loop concludes and posts or plots results as appropriate. Figure 22 (Appendix A) shows a detailed block diagram of this process including the different wake models.

## 3.2 Wake Models

### 3.2.1 Momentum Inflow Model

A momentum theory inflow model was initially incorporated to build the blade element rotor model. This established a baseline for the model and the functionality of direct comparison of the wake models. The induced velocity is initially calculated with the classical hover induced velocity equation shown below where  $T$  is the required thrust,  $\rho$  is the density of air, and  $A_{disk}$  is the rotor disk area.

$$w_w = \sqrt{\frac{T}{2\rho A_{disk}}} \quad 3-39$$

Since momentum theory assumes a uniform downwash across the rotor disk, as each blade element local velocity is calculated the uniform downwash velocity is added vectorially with rest of the velocities to determine local angle of attack. At each time step the rate of change of induced velocity is calculated and then follows the BER integration scheme to determine the new momentum theory uniform induced velocity. The rate of change of induced velocity incorporates the inflow dynamics using an apparent mass,  $M$ , which is adapted from the Pitt-Peters' dynamic inflow model [11].

$$V_{tot} = \sqrt{U_e^2 + V_e^2} \quad 3-40$$

$$M = \frac{8}{3\pi} R^3 \rho \quad 3-41$$

$$\dot{w}_w = \frac{(T - 2\rho A_{disk} V_{tot} w_w)}{M} \quad 3-42$$

### 3.2.2 VBOX

The VBOX free-wake model developed for this project aims to capture the wake characteristics utilizing vortex effects while also maintaining mathematical integrity. This is accomplished by creating a closed loop or box of filaments that satisfy Helmholtz's Law that a vortex filament shall not start or end in a fluid and shall have a constant vortex strength. From this closed loop of filaments, the induced velocities in the flow field can be determined and when combined with a reasonable amount of other VBOX's will create a wake structure. The reasonable number of boxes is to be determined with this study.

The VBOX's are formed as the blade whirls about the mast and at each azimuthal step, points in space are identified and deposited from a reference point at the tip of the blade and another near the root of the blade using the Blade Reference Line (BRL). After the blade has advanced to the next azimuthal location, another set of tip and root points are deposited. Once four points are created, they are connected to form a box starting from the leading root point out to the leading tip point then to the trailing tip point then the trailing root point of the previous azimuth angle and closed off by connecting to the starting point at the leading root point. Figure 4 depicts the geometry of a single VBOX. The line segments between vertices are vortex filaments and all four sides share a single, common value of strength in accordance with Helmholtz's Law.

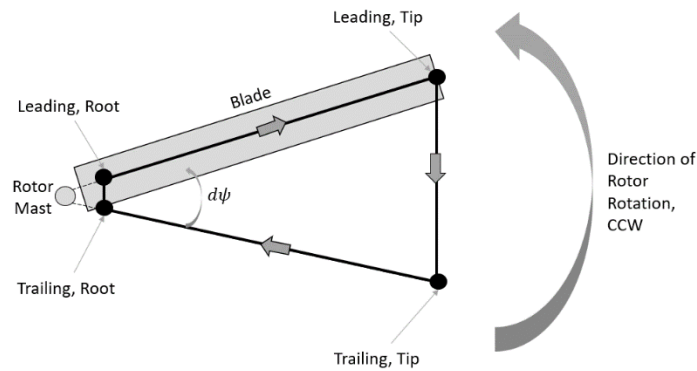


Figure 4. Formation of VBOX Geometry, Top View

After the first azimuth step a VBOX is formed and emitted by each rotor blade and for each azimuth step thereafter. Each VBOX shares both leading edge vertices with the VBOX formed next and similarly, both trailing edge vertices are shared with the VBOX formed just prior. As the VBOX's are formed they are chained together and remain bound together at their vertices as the wake deforms.

The path or direction taken to define the VBOX is important as it imposed a direction of rotation of the vortex. The path outlined above is pertinent such that the tip filament creates a vortex that rolls up on the outboard edge and similarly creates a downwash immediately behind the passing rotor blade.

Looking at a single VBOX such as the one shown in Figure 4 the general direction of induced flow would be into the page inside the box and out of the page outside of the box. This method follows the right-hand rule key for applying the Biot-Savart Law.

The VBOX method captures a discretized free-wake of tip-emitted trailing vortices and shed vortices.

The shed vorticity is physically a continuous flow that varies with the lift produced by the rotor according to the Kutta-Joukowski theorem but is discretized into individual filaments emitted from the rotor blade. Each newly created VBOX is assigned a new value for vorticity strength.

### 3.2.3 VTIP

The VTIP free-wake model developed is a simplified version of the VBOX model described above. Instead of forming a closed loop to satisfy Helmholtz's Law, only a single filament is created using two points. Each point is emitted from the tip of the rotor blade using the BRL, but one comes from the current azimuthal step and the other from the preceding azimuthal step. That single filament is given a unique value for vortex strength. The VTIP filament line segment also follows the right-hand rule, and the direction is identical to the VBOX tip segment.

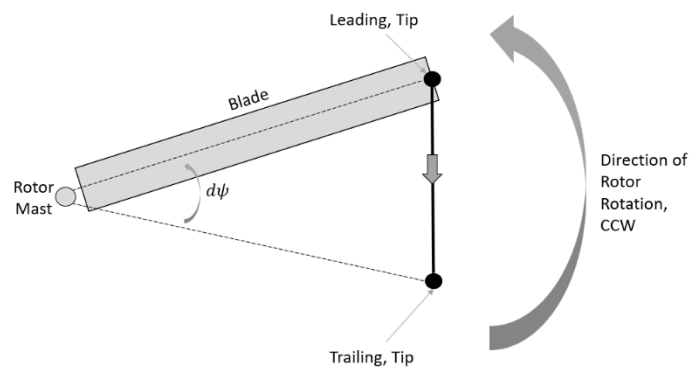


Figure 5. Formation of VTIP Geometry, Top View

One of the benefits to just using the tip vortex is that now only two vertices are tracked. The VTIP violates Helmholtz's Law because the filaments are terminated in a fluid. It is perceivable that the filaments are not actually terminated in a fluid, but they are conjoined with another filament (with the exception to the first filament emitted); however, this still violates Helmholtz's Law in that it has discontinuously varying strength of vorticity along the filament. As part of this study the difference in the two models were investigated but for now this assumption must be taken on faith that the difference is negligible. Egolf and Landgrebe also made this same assumption in their vortex lattice, prescribed wake model [3].

### 3.2.4 VTIP and VBOX Common Method

The vortex strength for each filament is associated with a particular blade and azimuth step. The vortex strength is calculated within the *blade* module of the BER, where the distributed aerodynamics forces are calculated using a lifting line and then summed. Using the total blade lift and assuming a triangular lift distribution the value for vortex strength is calculated using the following equation where L is the summed lift of the blade, R is the rotor radius,  $U_t(x = 1)$  is the tangential velocity at the blade tip and  $\rho$  is the air density. The 2 in the numerator and the R in the denominator are a result of the triangular lift distribution on the blade to determine the strength of the vortex at the top of the blade.

$$\Gamma = -\frac{2L}{R U_t(x = 1)\rho} \left[ \frac{ft^2}{s} \right] \quad 3-43$$

Once each VBOX or VTIP is formed the Biot-Savart Law is applied using the stored vortex strength and the known locations of the vertices. For the mitigation of the singularity at the vortex filament a Scully vortex core is used and is given the value of 5% of the rotor radius. The chosen core radius and linear approximation is sufficient for capturing the gross effects of the vortices being emitted while reducing large spikes in velocity. The induced velocities at all vertices are calculated as well as along the BRL and a series of points on the disk plane. Once all the induced velocities at all the points for a single time step/ azimuthal step are determined, the induced velocities are used to predict all the vertices next location in space utilizing a simple Euler step, shown below where  $P_n$  is the position vector,  $V_n$  is the velocity vector summed from all the induced velocities,  $V_{inertial}$  is the inertial velocity of the rotor and  $dt$  is the differential time step that applies to a single azimuthal step.

$$P_{n+1} = P_n + (V_n + V_{inertial}) * dt \quad 3-44$$

### 3.2.5 Model Trimming

A certain number of iterations must be performed to converge on a steady state or trimmed condition. The model must accumulate the vortex filaments and they must reach a converged structure and strength before the model can be considered trimmed. Even though vortices structures are inherently chaotic the gross effects of the vortices can and do converge.

Several studies during the development of the two vortex-theory free-wake models were performed to determine the number of rotor revolutions that would be required to achieve a trimmed or converged state. The approach divided the problem into two parts; the first part set the total number of revolutions that were executed for a given condition setup and the second part specified the number of revolutions that remained within the flow field. The former was primarily focused on getting the start-up wake to propagate significantly far downstream so that the effects from it were minimal. The latter was accomplished using a 'first in, first out' concept so that after the iteration count exceeded a specified number of revolutions, the first formed vortex filaments were ignored and effectively erased leaving only the converged wake structure.

To determine which method and how many revolutions were required the convergence of blade flapping, rate of change of blade flapping, thrust required, and induced velocity were examined for several airspeed conditions with hover shown as an example. Using the first method, a large revolution count was executed while tracking the aforementioned parameters. The resulting plots showed that between 12 and 14 revolutions provided sufficient time for the model to settle on a result. The following figures show the curves of the parameters during the execution of a hovering rotor. Figure 6 shows the uniform equivalent induced velocity as measured and averaged for each time step along with a Savitzky-Golay [12] filtered curve to distinguish the steady component. Figure 7 shows the trace of blade flapped position for a single blade during the trimming phase. Similarly, Figure 8 shows the trace



of the time rate of change of the blade flapped position. The last of these figures, Figure 9, shows the trace of the forces as summed and oriented with the fixed mast axis; the x- and y- forces are near 0 while the z-force is converging on the desired thrust of 6,000 lbf.

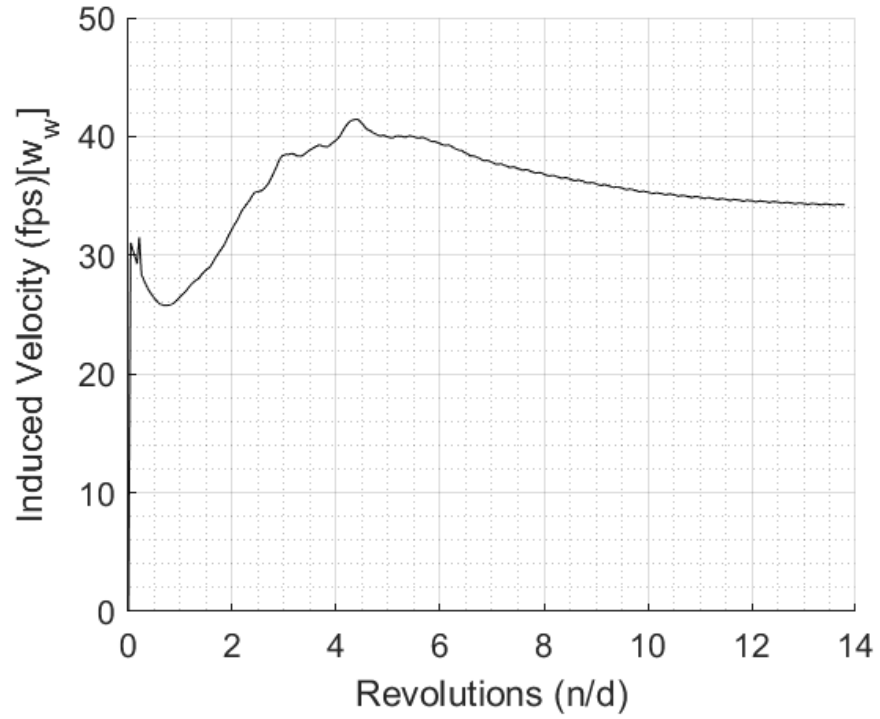


Figure 6. Hover Induced Velocity During Execution

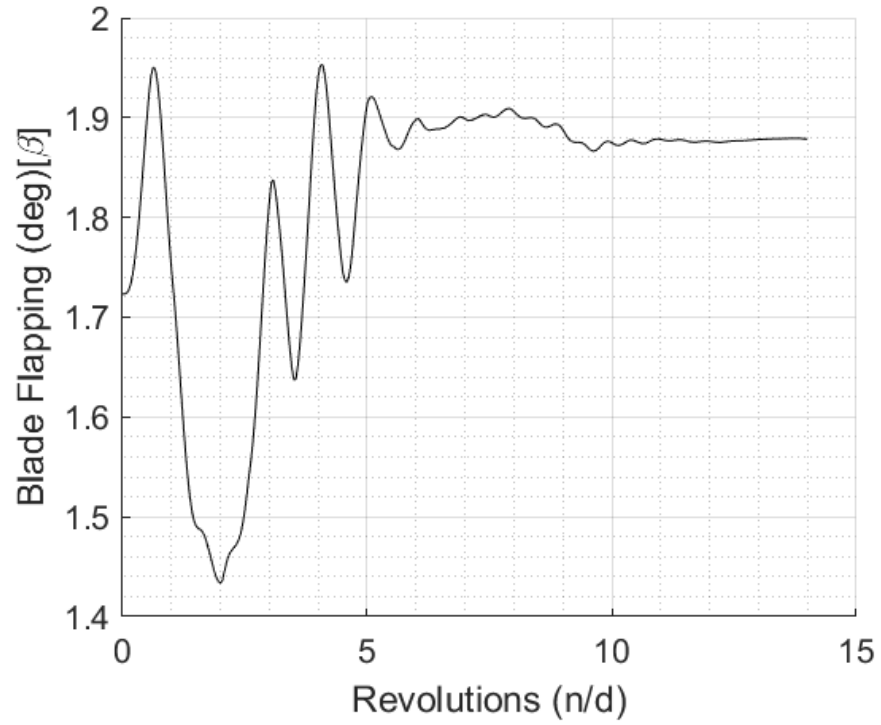


Figure 7. Hover Blade Flapping During Execution

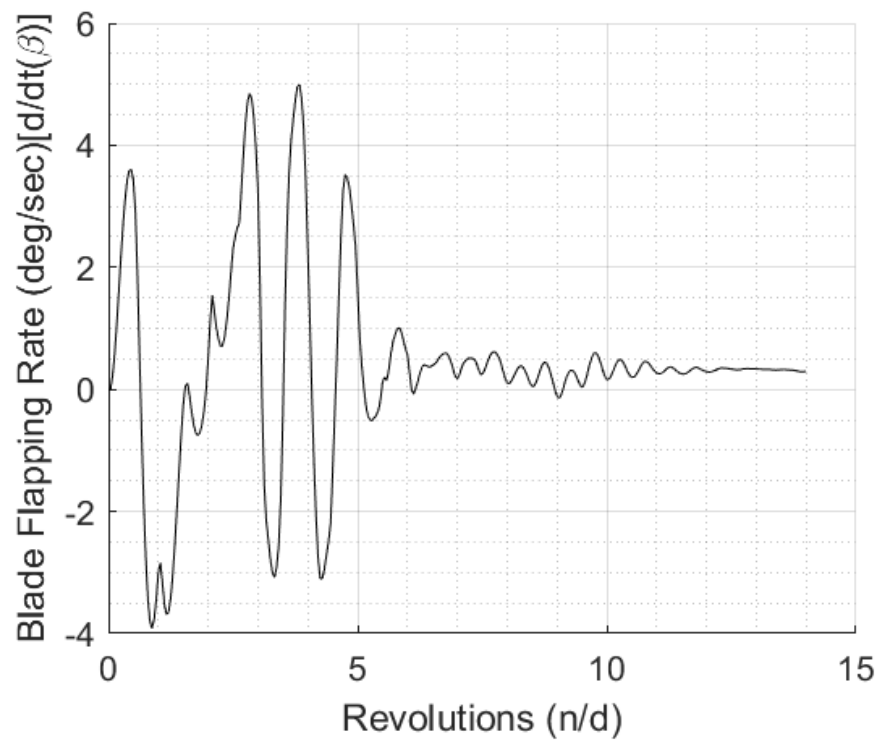


Figure 8. Hover, Blade Rate of Change of Flapping During Execution

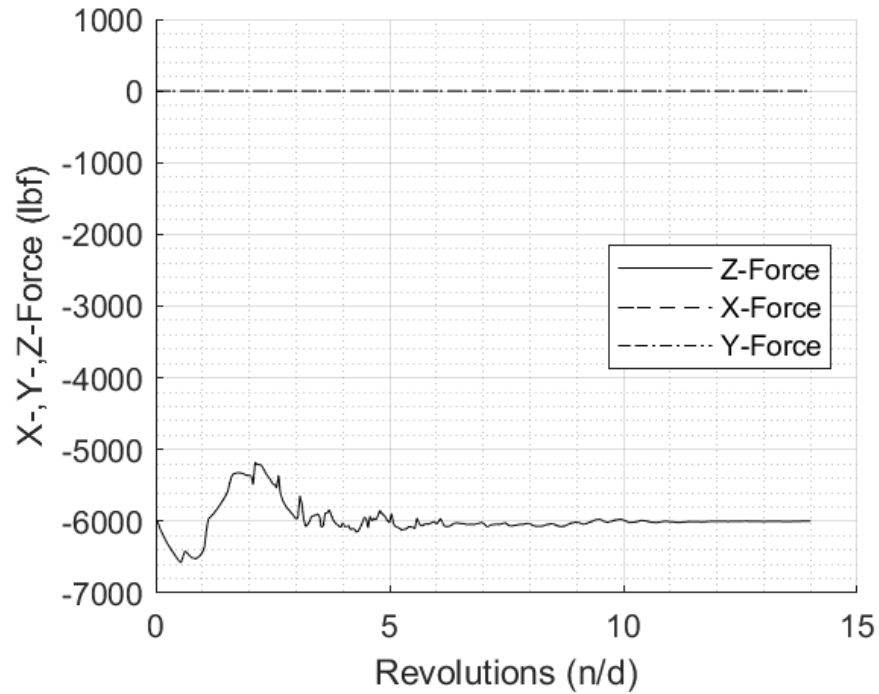


Figure 9. Hover, Rotor Forces During Execution

The second method simply tested and tracked a large number of revolutions. This method had the benefit of fixing the number of calculations per azimuth step whereas the former method continuously grew in computational time per azimuth step. Even with the benefits of the possible reduced computational time, the “first in, first out” method still required the same 12-14 revolutions that the first method required. This is primarily due to the “ $1/r$ ” relation that the Biot-Savart Law has on the induced velocity. In essence, the evolving wake must push the start-up portion sufficiently far away so that it has a negligible effect on the flow field near the rotor and this occurs regardless of the two methods used. The start-up wake is even less of a concern because the rotor is flying away from the start-up wake further reducing its effects on the velocity field near the rotor.

For these reasons a simple revolution count of 14 was used for the analyses. The “first in, “first out” method would be beneficial for extended duration runs for maneuvering or other dynamic simulations

because it only must track a finite number of vertex positions and vortex filaments, whereas the large revolution count computation demand continuously grows.

#### 3.2.5.1 Vortex Life Model

The life of a vortex is finite and as such an investigation was taken within the confines of this model to determine the effects that vortex decay would have on the results. A vortex decay model from Flandro et al [13] that reduces the strength as a function of time was used in a series of tests, but the resulting decay in vortex strength was numerically invisible. A sensitivity test was also accomplished by further decaying the vortex strength and evaluating the results of the rotor model. The results of this sensitivity test showed that the decay of vortex strength had little effect on the rotor behavior within the time frame that the vortices remain within range of the rotor. Thus, the decay of vortex strength was assumed negligible for these free-wake models and removed.

## CHAPTER 4. Results

### 4.1 Description of General Rotor Model Parameters

Unless otherwise stated the following rotor parameters were used for analysis. The rotor is a two bladed, articulated type with a radius of 20 feet. The blades have a rectangular planform with a linear twist of -10 degrees and a constant chord of 1.5 ft. The blade airfoil is a NACA 0012 and either uses a lift curve slope of 5.73 CL/rad or a table look up with Mach effects built in for generally large Reynolds number airfoils. The airfoil drag is determined using either an average coefficient of drag of 0.0090 or uses the table look up. In either aerodynamic method the pitching moment is assumed to be zero. The blades have a distributed mass of 0.4 slug/ft resulting in each blade weighing 257.36 lbf. The blades are assumed infinitely rigid except for the flapping hinge (no lead-lag hinge) that is coincident with the mast axis and has a flapping spring with a flapping stiffness of 133,930 ft-lbf/rad. The rotor whirls around the mast at 35 rad/sec which equates to a blade tip speed of 700 ft/sec. The given parameters result in a blade flapping natural frequency,  $P$ , of 1.05. The required thrust is 6,000 lbf and the forward drag equivalent flat plate area is 16 ft<sup>2</sup>. The coefficient of thrust is 0.0041 at sea level standard.

### 4.2 Comparison of Speed Power Polar

The model developed for this thesis was compared with a physics based, empirical performance model for calculating the power required across a sweep of airspeeds. The blade element model was exercised for this comparison using all three wake/inflow models, that is the momentum theory, VTIP, and VBOX versions. The empirical performance method is taken from reference [14] and typically denoted using "S-K" in respect for the authors. The equations and parameters necessary to determine the power required using the performance model are shown below. It was assumed that the airfoil differences are traceable and the equations for drag due to compressibility and stall are used straight from the text.

$$\mu = \frac{U}{V_{tip}} \quad 4-1$$

$$T = W \quad 4-2$$

$$\sigma = \frac{bc}{\pi R} \quad 4-3$$

$$C_T = \frac{T}{\rho AV_{tip}^2} \quad 4-4$$

$$B_t = 1 - \frac{\sqrt{2C_T}}{b} \quad 4-5$$

$$v_0 = \sqrt{\frac{T}{2\rho A_{disk} B_t^2}} \quad 4-6$$

$$v_{if} = v_0 \sqrt{-\frac{\left(\frac{U}{v_0}\right)^2}{2} + \sqrt{\frac{\left(\frac{U}{v_0}\right)^4}{4} + 1}} \quad 4-7$$

$$M_{90} = \frac{U + V_{tip}}{V_{sound}} \quad 4-8$$

$$C_{d_{compressibility}} = 0.2(M_{90} - M_{dd})^3 + 0.0085(M_{90} - M_{dd}) \quad 4-9$$

$$F = \left[ \frac{\frac{C_T}{\sigma}}{(1 - \mu)^2} \right] \left[ 1 + \frac{f_e q}{W} \right] - 0.1375 \quad 4-10$$

$$C_{d_{stall}} = 18.3(1 - \mu)^2 F^2 \quad 4-11$$

$$C_d = C_{d_0} + C_{d_{compressibility}} + C_{d_{stall}} \quad 4-12$$

$$HP_{total} = \frac{T v_{if}}{550} + \frac{\sigma C_d (1 + 4.7\mu^2) \rho A_{disk} V_{tip}^3}{4400} + \frac{f_e \rho U^3}{1100} \quad 4-13$$

$$HP_{total} = \text{Induced HP} + \text{Profile HP} + \text{Parasite HP} \quad 4-14$$

The resulting curve is shown in Figure 10 alongside the three total power required curves from the project BER using the different wake/inflow models. The “S-K’ curve shows a near constant offset for

total power required which is believed to be a result of the different airfoils used. The resulting power-required curves trend closely with the empirical performance calculation. This validates both the base blade element model with momentum theory inflow as well as the VTIP and VBOX free-wake models. The VBOX curve at 20 knots is out of trend with the other models. This dip in total power required is a result of the method used to extract the power required. A Fourier analysis is used but only the steady and first harmonics are extracted. The analysis plots used to determine convergence over the iterations shows that for this airspeed the forces and moments remain unsteady and have higher harmonic oscillations and irregular growth and decay of these oscillations. Further investigation revealed that as the rotor is flying at 20 knots the wake is starting to transition out from underneath the rotor. At this slower, transitioning airspeed the rotor is recirculating some vortices before being able to “outrun” the wake and into translational lift where the inflow is more horizontal and not being recirculated by the rotor anymore. This interaction is causing the blades to encounter varying vortex influences; the variance in inflow across the rotor is causing the forces and moments to oscillate representing vibration felt by the aircraft as it flies through this transitional phase. This finding is likely the same phenomenon felt in helicopters while transitioning through to translational lift and is being captured in the vortex box free-wake model with the increased number of filaments.

Breaking out the induced velocities and induced power validates the trends of the developed models. The base momentum BER models shows less profile power required due to the generally lower drag coefficient than the airfoil used in the empirical performance curve. The vortex-theory free-wake models over predict the induced power at lower speeds and underpredict at higher speeds as seen in Figure 11. This induced power prediction is tied directly to the induced velocities which are shown in Figure 12. The free-wake induced velocities trend with predicted levels; the variation is due to the non-uniform inflow which is expressed in a single “equivalent” value. This “equivalent” value is calculated by obtaining the velocity through the rotor disk plane at a series of points. Those velocities are then

divided by the area of the disk plane that they represent creating a flux which is the divided by the total disk area. The time history of the “equivalent” induced velocity value is processed with a Fourier script that extracts the steady and first harmonic terms.

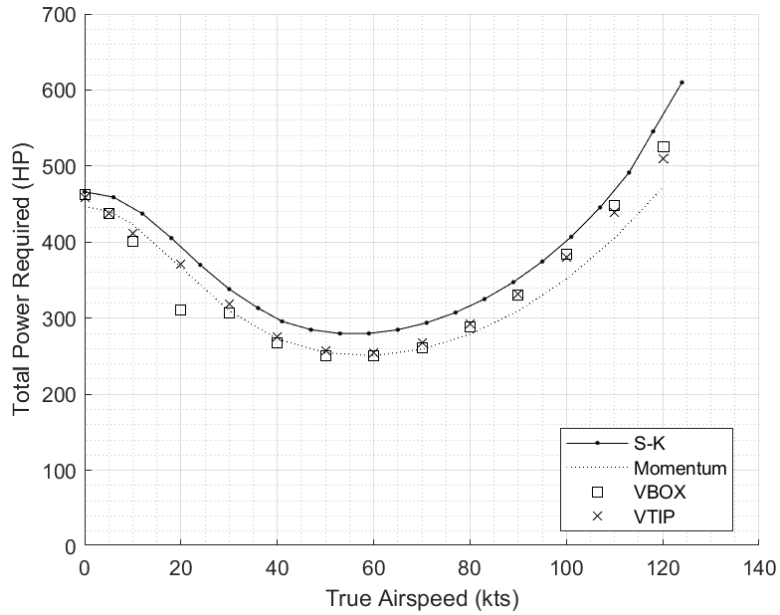


Figure 10. Comparison of Total Power Required Results

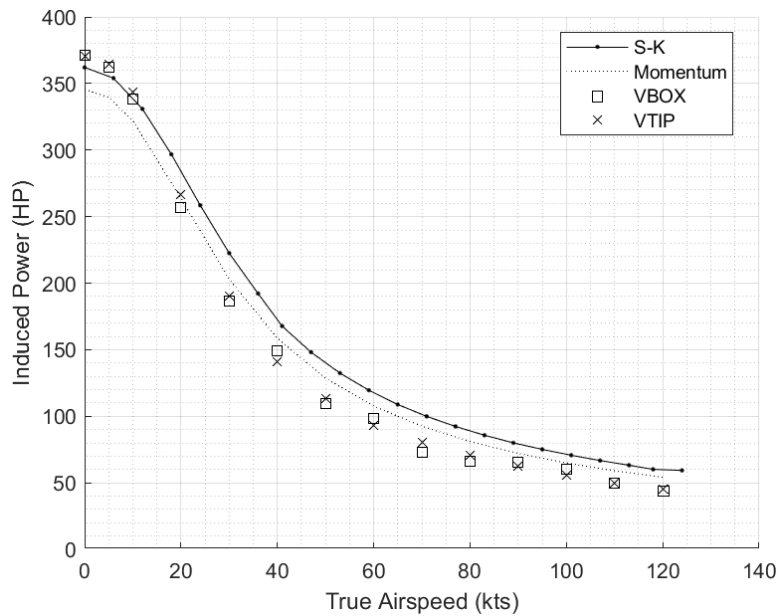


Figure 11. Comparison of Induced Power Required Results



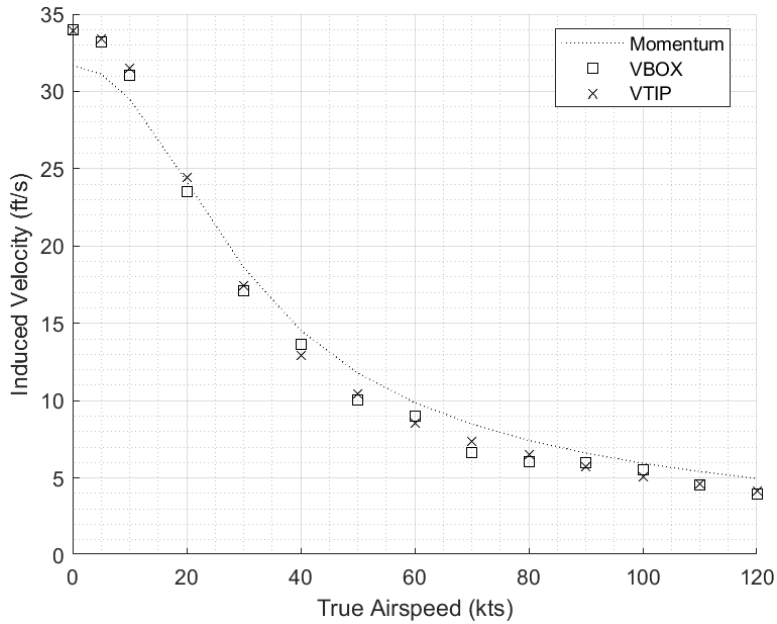


Figure 12. Comparison of Induced Velocity Results

### 4.3 Comparison of Lateral Flapping vs Advance Ratio

Analytical models have historically been challenged in predicting lateral flapping particularly at low advance ratios (less than 0.4) in part because the inflow is not uniform across the rotor disk.

Momentum theory is also challenged in this way because of the uniform inflow and even with a BER model to get local velocities the momentum model still projects uniform inflow across the entire rotor disk. To demonstrate the robust capability of the free-wake models developed here, the rotor properties were altered to best match the powered wind tunnel model described in reference [15]. The Boeing powered wind tunnel test data is shown in Figure 13 alongside the project model results. The BER with momentum theory inflow fails to capture the trend whereas the VTIP and VBOX, free-wake models qualitatively match the trend and magnitude of lateral flapping. As alluded to, the key difference is the ability for the free-wake models to model different velocities over the rotor disk, i.e., a non-uniform inflow. During this phase of flight, the rotor is experiencing a transverse flow effect where the front part of the rotor is experiencing different inflow than the rear of the rotor and with the

gyroscopic precession, the rotor tends to flap laterally. Other than the geometric properties updated for the test rotor, no other inputs or modifications were required, further validating the organic robustness of the VTIP model.

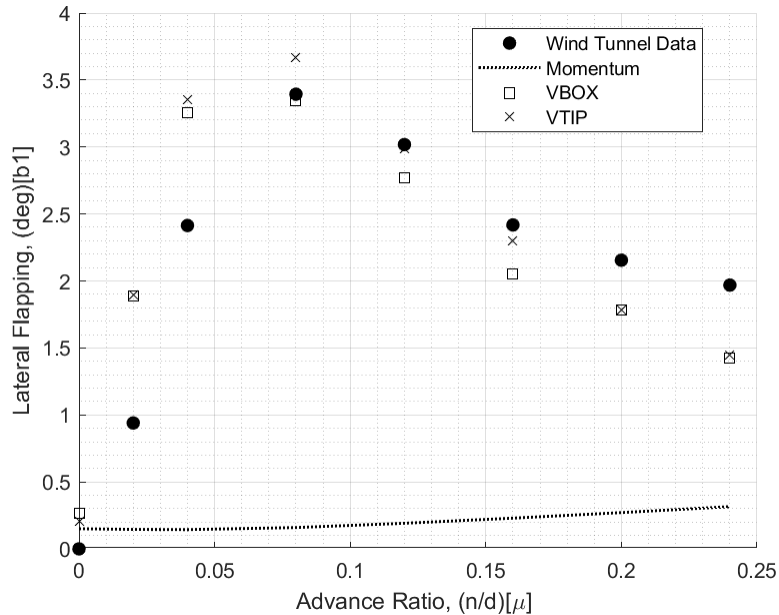


Figure 13. Lateral Flapping Comparison

#### 4.4 Comparison of Induced Power Required Due to Swirl Wake

Using the thrust coefficient for the general rotor of 0.004099 and the equation below from Prouty [16], the ratio of induced power required for the swirl flow in the wake is 1.17% of the static induced power required. For the hovering rotor, the momentum theory returned an induced power required of 346 HP and the free-wake 360 HP. This difference is greater than that predicted by Prouty but due to the small effect it could be masked by other effects. It is worth mentioning that the VTIP and VBOX wake models do not require any ad hoc or special fixes whereas Prouty had to alter the lower limit of integration to keep the radical positive. VTIP and VBOX free-wake models inherently capture the non-axial flow field and thus any power required to swirl the air is also included within the power calculations.

$$\frac{P_{i_{rotation}}}{P_{i_{static}}} = \frac{1}{C_T} \int_{\sqrt{2C_T}}^1 (x^3) \left[ 1 - \sqrt{1 - \frac{2C_T}{x^2}} \right]^2 dx \quad 4-15$$

## 4.5 Comparison of Wake Dynamics

To compare the wake dynamics a look at the change in axial induced velocity due to a collective step input can be used to compare against dynamic inflow theory. This is accomplished by calculating the change in axial induced velocity from the perturbed dynamic inflow equation near a hover with the corresponding rotor parameters as used in the VTIP wake model. The VTIP model was setup in a hover then given a step change in collective control to create a change in induced velocity.

Start off with the dynamic inflow equation and then perturb the equation near a hover. Substituting in the spherical mass of air through the rotor,  $M$ , and incorporating an empirical factor,  $k$ , of 0.86 [17], brings this expression in alignment with the Pitt-Peters dynamic inflow model. The axial velocity,  $V_A$ , is the aerodynamic inflow including the inertial and induced velocities; for the hovering case, the inertial velocity is zero. The tip loss factor,  $B_T$ , is estimated to be 0.97

$$Mw_w + 2\rho A_{disk} B_T^2 |V_A| w_w = T \quad 4-16$$

$$|V_A| = \sqrt{u_i^2 + (w_i - w_w)^2} \quad 4-17$$

$$M = \frac{4}{3} \pi R^3 \rho k^3 \quad 4-18$$

The resulting linearized equation near hover and solved for the time rate of change of induced velocity is

$$\Delta \dot{w}_w = \frac{\Delta T - 4\rho A_{disk} B_T^2 w_{w_o} \Delta w_w}{M} \quad 4-19$$

Using this linearized equation near a hover and for small perturbations, the rate of change of induced velocity can be evaluated at the instant that a collective step input is applied. The starting thrust of the hovering rotor is 6000 lbf and for comparison the same final thrust value of 7250 lbf that the VTIP

achieves is used to make the comparison between the two models. This increase in thrust occurs before the wake or induced velocity has reacted and the change in induced velocity right at the time the input is applied is still zero, thus:

$$\Delta w_w = 0 \quad 4-20$$

$$\Delta \dot{w}_w = \frac{\Delta T}{M} = \frac{7250 - 6000}{50.7} = 29.99 \frac{ft}{s^2} \quad 4.21$$

From this, the rate of change of induced velocity at the instant that a collective step is applied is 29.99 ft/s<sup>2</sup>.

Now use the resulting induced velocity trace from a collective step input from trim at 6000 lbf thrust. Take two points along the curve of induced velocity change just after the step input is applied and determining the linear change results in a time rate of change of induced velocity of 25.72 ft/s<sup>2</sup>. The difference in the two models is about 14% with the momentum/dynamic inflow being greater and reflects the similarity in responses.

$$\Delta \dot{w}_w = \frac{w_{w_2} - w_{w_1}}{(Rev_2 - Rev_1) T_{period}} = \frac{35.0922 - 34.1301}{(14.0417 - 13.8333)0.1795} = 25.72 \frac{ft}{s^2} \quad 4-22$$

Qualitative evaluation of the dynamic response of the models also shows the similarity between the two types of models. Shown in Figure 12 are the two induced velocity traces overlaid and normalized by their respective initial (prior to step input) hovering induced velocity. The general shape of the two curves shows the similarity. It is important to remember that the free-wake induced velocity is not simulated using a single value but rather is the non-uniform inflow determined by the spatially and temporally varying vortex-theory flow field. The single value shown is an average of the induced velocity across the whole rotor, but the tip vortex being at the outer circumference is affecting the outer region more. This averaged

method used to compare the models is limited since the two models are inherently different, but the method permits an approximate comparison. The resulting similarity of the free-wake vortex-theory model with prior accepted dynamic inflow model reflects their capability to be inherently robust.

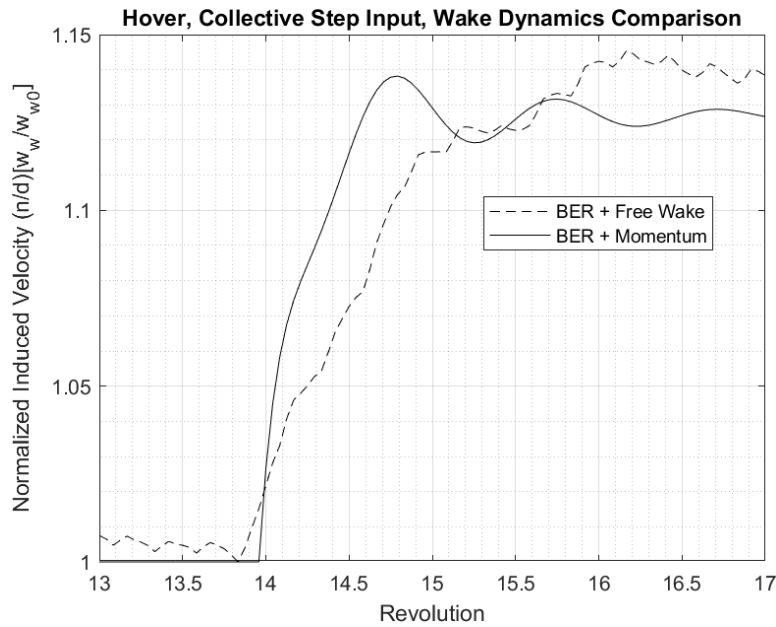


Figure 14. Induced Velocity Response to Collective Step

#### 4.6 Comparison of Wake Characterization

The qualitative characteristics of the wake from the VTIP and VBOX model helps visualize the commonality with wake visualizations. The visualizations show general shape and wake characteristics, but the velocity flow field is the important result from the free-wake models. First looking at the hovering rotor from the side, Figure 15 shows the wake of a hovering rotor where the radius of the rotor is 20 feet and as the wake propagates downstream it contracts to approximately 14 feet. This ratio of wake contraction matches theory which states that the stream tube should contract to 0.707 times the radius which is derived from the continuity equation and the momentum theory result that the induced velocity downstream the disk,  $v_2$ , is twice the induced velocity just below the rotor disk,  $v_1$ .

This is derived in the equations below where the properties just below the rotor are denoted with subscript 1 and downstream is subscript 2.

$$\dot{m} = \rho_1 A_1 v_1 = \rho_2 A_2 v_2 \quad 4-23$$

$$\text{Assume Incompressible: } \rho_1 = \rho_2 \quad 4-24$$

$$A_1 v_1 = A_2 v_2 \quad 4-25$$

$$v_2 = 2v_1 \text{ (from momentum theory)} \quad 4-26$$

$$\frac{v_1}{v_2} = \frac{A_2}{A_1} = \frac{v_1}{2v_1} \quad 4-27$$

$$\frac{\pi R_2^2}{\pi R_1^2} = \frac{v_1}{2v_1} \quad 4-28$$

$$R_2^2 = \frac{R_1^2}{2} \quad 4-29$$

$$R_2 = R_1 \sqrt{\frac{1}{2}} \cong R_1 0.707 \quad 4-30$$

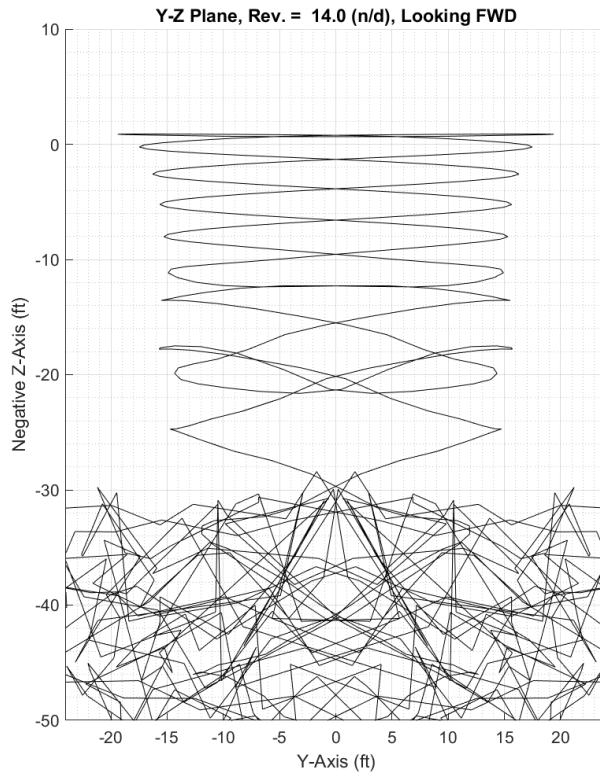


Figure 15. VTIP Hover Wake Visualization.

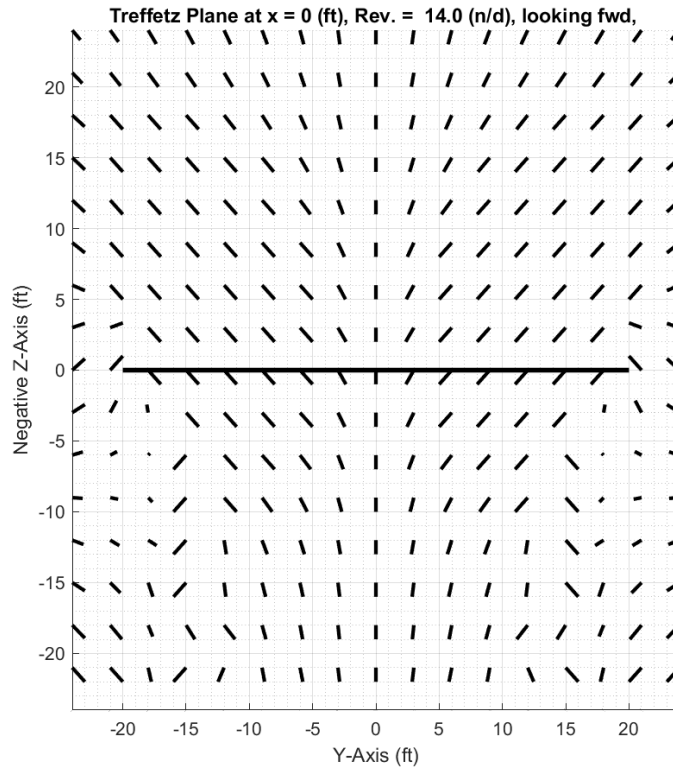


Figure 16. Velocity Field at Midplane of Rotor

Figure 15 also depicts the startup portion of the wake in the lower section of the plot where the large cluster of intersecting lines exists. As the simulation is continued this cluster of filaments will propagate further away from the rotor and was shown during the trimming process to have negligible effects on the rotor once 12-14 revolutions have incurred. Figure 16 is a Trefftz plane that is bisecting the hovering rotor and thus showing the velocity field with each line representing a discrete velocity vector. The thick horizontal line, near the center of the plot is a profile view of the rotor disk. The conglomeration of velocity vectors shows both the general wake contraction downstream from the rotor and the bell mouth above the rotor that aligns with the conservation of mass through a stream tube.

Another very important capability that the vortex-theory free-wake models show is the vortex tip roll up during forward flight. Seen in Figure 17 the wake visualization from a tip drive helicopter shows a



classical horseshoe vortex structure like a fixed wing would produce in forward flight. This phenomenon is also organically created with the VTIP and VBOX free-wake models as demonstrated in Figure 18 showing a three-dimensional view of the VTIP model filaments after a sufficient number of revolutions have passed. The tip roll-up from the vortex-theory free-wake model is inherent and requires no ad hoc or special conditionals. This is a stark difference from prescribed wake models that hold the vortices to a prescribed shape and do not allow the wake to develop more naturally but rather fix them along a skew helix like the one shown in Figure 19. The free-wake models organically develop the skew of the wake which must be predetermined when using a prescribed or stream tube wake model.



*Figure 17. Tip Drive - Wake Structure in Forward Flight [18]*

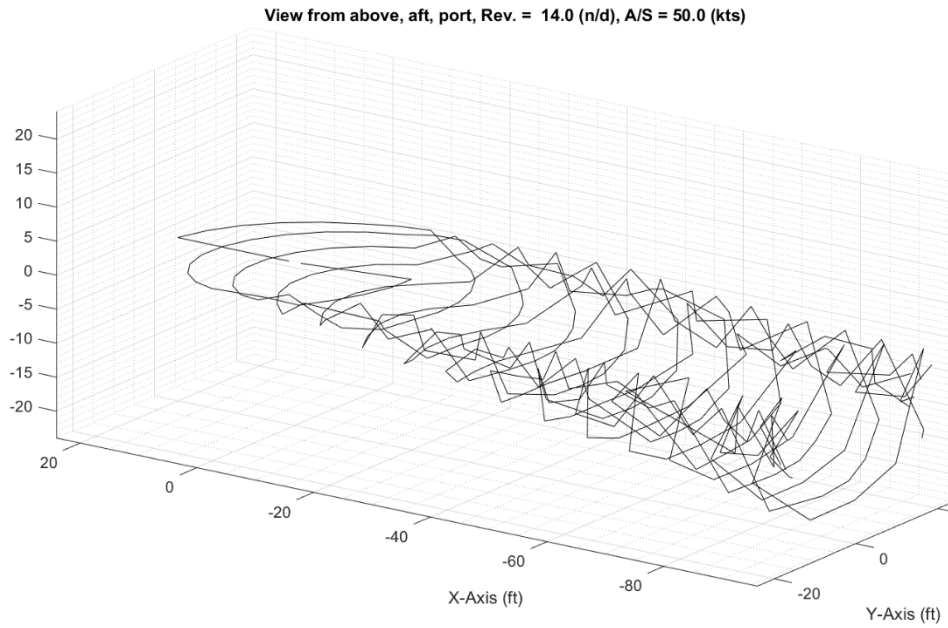


Figure 18. 3D View of VTIP Free-wake showing Tip Roll-Up and Classical Horseshoe Vortex



Figure 19. Prescribed Wake Example [3]

Another inherent phenomenon that the vortex-theory free-wake models create, again with no special rules or routines, is the buildup of the rotor wake formed after control inputs that then exhibits a dynamic time delay before propagating downstream. Figure 20 shows the progress of the wake structure immediately following a collective step input, starting in pane 1, the upper left image, with the classical contracted wake in a hover just before the step input is injected on the beginning of the seventeenth revolution. Pane 2 captures the wake following a brief delay after the step input in which the wake deforms slightly below the disk plane. As that deformation grows the initial and subsequent vortex filaments begin to swirl around one another as depicted with the arrows in pane 3 through 8. This swirling in 3D space is the formation of a torus comprised of filaments that first buildup into the torus and then begin to propagate downstream. This buildup and delay are analogous to the time constant because of the apparent mass effect that the Pitt-Peters dynamic inflow models incorporated to improve the enhanced momentum theory inflow model.

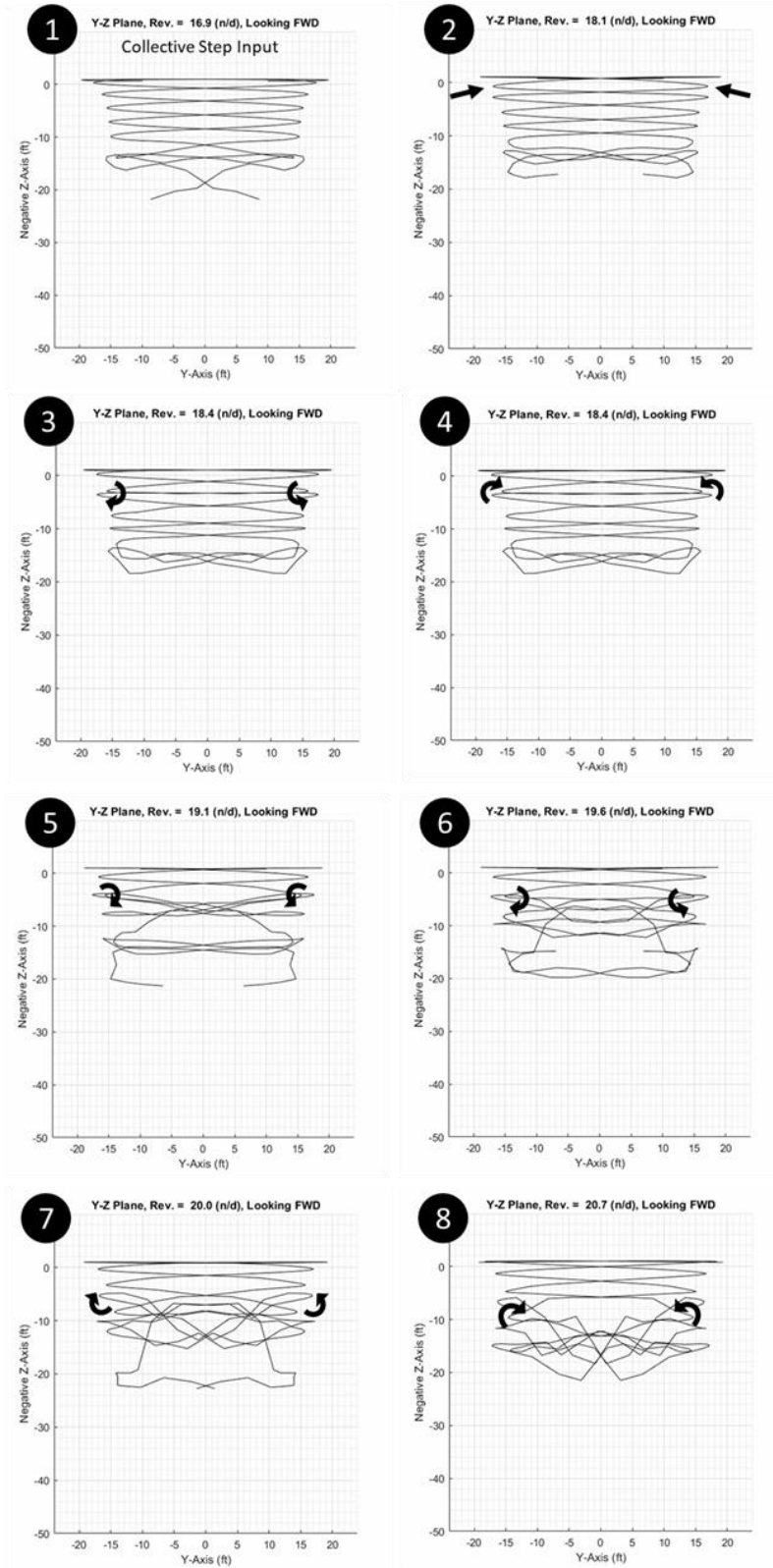


Figure 20. VTIP Wake in Hover with Dynamic Wake Buildup

## 4.7 Comparison of Induced Velocities in High Descent Rates

One aspect of rotor modeling and simulation that is challenging to predict is Vortex Ring State (VRS).

Momentum inflow and prescribed wake models fail to properly capture the very transient and dynamic state of the rotor during high descent rates. High fidelity models such as CFD can also struggle to predict the velocity field in VRS and would require a large amount of computational power. Using the VTIP and the VBOX free-wakes, a sweep of axial descent rates was performed. The resulting induced velocities are compared with Momentum Theory, Momentum Theory with the BER, and powered wind tunnel test data [19]. Although the resulting free-wake induced velocities do not match closely with the experimental data, it does show a similar trend of breaking away from the momentum theory and then falling off more rapidly, similar to the test data's general trend. This provides insight that the model is close to properly predicting VRS and neighboring states, but that more refinement or explanation of the difference is required.

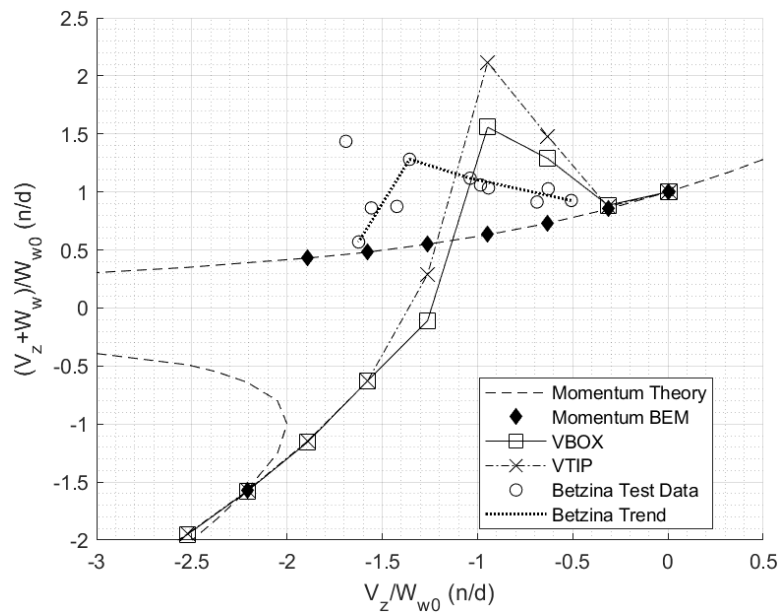


Figure 21. Induced Velocity for Vertical Descent Rates Comparisons

## 4.8 Comparison of VTIP vs VBOX

As it applies to a rotor, one can see the tip vortices emitted from a rotor blade as it whirls around. A student of rotors may also know that the lift is 1) not constant along the radius of the blade (unless ideally twisted) and 2) the lift (even blade span averaged lift) is not constant as it rotates about the hub. This implies that the trailing tip vortex strength varies through time and space, but this is in violation of Helmholtz's first theorem. To rationalize this process there must be a sheet of continuously shed vortices. Another way of thinking through this is small pie shaped closed loops emitted from the rotor blade after a specified time or azimuth step. This is the principle of the VBOX model to satisfy the mathematical law. From the results in Figure 10 and Figure 12 we see the total power and induced velocities of the subject rotor indicate that the tip vortex only model is sufficient for performance prediction and general wake structure characterization. The two models produce similar results, but the VBOX model requires approximately three times the amount of computation time compared to the VTIP model. This is clearly due to not having to calculate all four sides of the box but instead just needing to calculate the tip emitted vortex influence.

## CHAPTER 5. Summary

A survey of the general types of rotor models and associated inflow/wake modeling was completed, and the choice was made conceptually to incorporate a vortex-theory free-wake model into a blade element rotor model. By using a blade element model, high levels of modeling freedom and future expansion can be accomplished. The choice of a vortex-theory free-wake model was a trade of computational efficiency and model fidelity. The vortex-theory free-wakes allow for a wider range of possible uses with no correction factors or conditional rules. Two configurations of vortex-theory free-wakes were implemented, both using a bound vortex, one using just a tip emitted vortex filament, VTIP, and the second emitting a “box” of filaments each azimuthal step, VBOX. Both VTIP and VBOX models used the Kutta-Joukowski Theorem to couple the blade lifting force to the vortex circulation strength. Furthermore, the VBOX was able to satisfy Helmholtz’s Law by forming a closed loop filament with constant strength throughout.

Validation of both models against an empirical performance model for required power showed good results across a large range of airspeeds. The free vortex wakes are also robust in predicting the lateral blade flapping at slow speeds that momentum theory and axial-flow quasi-static models fail to do.

The qualitative characteristics of the vortex-theory free-wake models also align with the physical characteristics of actual rotorcraft wakes. This includes the wake contraction in hover, vortex tip roll-up in forward flight, the toroidal air mass buildup due to thrust (through collective step input) increases. It also suggests an explanation for the apparent mass effects from dynamic inflow models. All these features and results come at a relatively low computational expense.

The wake vortex interactions with the blades near 20 kts suggests that the vortex box free-wake model is also capturing transient vibrations experienced in helicopters as they transition through translational lift.

Using the vortex-theory free-wake models, it was discovered that between 12 and 14 revolutions built up enough wake for it to converge to steady state solution and the wake itself settled into a particular structure for operating conditions. The solutions at steady state were qualitatively and quantitatively consistent with test data, theory, and accepted model comparisons.

The comparison of VTIP and VBOX results show that ignoring the shed vortex and root vortex is inconsequential in predicting power and blade behavior for the conditions tested. The omission of the shed and root vortex further reduces the computational expenses without sacrificing model fidelity, thus for most engineering applications the VTIP model provides an accurate result in a fraction of the time.

The vortex-theory free-wake models were insensitive to using a vortex strength decay model for normal modes of operation. Flight regimes that trap the wake near the rotor for extended periods may benefit from other methods of vortex decay modeling.



## CHAPTER 6. Recommendations/Future Work

### 6.1 Expand BER Functionality

One of the next major steps that should be taken is to expand the functionality of the kinematics and dynamics of the BER so that a more realistic rotor can be analyzed. This expansion would include the offset flapping hinge, lead-lag hinge along with the lead-lag dynamics and the inclusion of delta-3. Other BER improvements would be to include aeroelastic blade dynamics using calculated normal mode shapes or an inclusion of simultaneous blade structural dynamic solver.

### 6.2 Increase model performance for Realtime

The model developed for this thesis is the beginning step of a good analysis tool but improvements to computational power and optimization of the required number of iterations and calculations can lead to a free-wake, real time rotor simulation model. With proper validation such a tool would be invaluable to many rotorcraft developers and manufacturers. As seen in the results for the high rates of descent and VRS conditions, the model can provide sufficient rotor wake behavior and further improvements could align the model output with proper test data.

### 6.3 Inclusion of Full Up Air Vehicle Simulation Model

Analysis of an isolated rotor is excellent for comparison with theory and some experimental data for validation; however, for rotorcraft design the inclusion of the whole, full up air vehicle components should be included. The scope of this effort may be very large due to the nature of also needing to account for wake interferences, ground effect, control law implementation, dynamics, and more to build a high-fidelity aircraft simulation.

#### 6.4 V BOX Improvement for Higher Fidelity Near Wake Modelling

The VBOX model built up in this research uses only one box per blade per azimuth step but increasing the number of boxes along the span of the blade would allow for more flexibility in blade nearfield wake prediction and permit span wise load distribution to provide more insight to the flow field near the rotor plane.

#### 6.5 Further investigation into high descent rates

The models show good trending with limited test data and future work expanding on this region could show better results. Accurate prediction of onset of Vortex Ring State is invaluable to saving lives and aircraft.

## CHAPTER 7.      References

- [1] A. Gessow and G. C. Myers, *Aerodynamics of the Helicopter*, College Park Press, 1985.
- [2] D. M. Pitt and D. A. Peters, "Theoretical Evaluation of Dynamic Inflow Derivatives," *Verica*, vol. 5, pp. 21-34, 1980.
- [3] T. Egolf and A. Landgrebe, "A Prescribed Wake Rotor Inflow and FLOW Field Prediction Analysis - User's Manual and Technical Approach," United Technologies Research Center, East Hartford, CT, 1982.
- [4] F. Harris, "A Note about Self-Induced Velocity Generated by a Lifting-Line Wing or Rotor Blade," NASA CR-2006-213478, NASA Ames Research Center, 2006.
- [5] W. Johnson, *Helicopter Theory*, New York: Dover Publications, Inc., 1994.
- [6] J. Maxwell, "On Physical Lines of Force," *The London, Edinburgh and Dublin Philosophical Magazine and Journal of Science*, vol. Fourth Series, March 1861.
- [7] L. A. Young, "Vortex Core Size in the Rotor Near-Wake," NASA Ames Research Center, Moffett Field, CA, 2003.
- [8] A. Brand, M. Dreier, R. Kisor and T. Wood, "The Nature of Vortex Ring State," *Journal of the American Helicopter Society*, vol. 56, no. 2, 2011.
- [9] P. Anusonti-Inthra, "Development of Rotorcraft Wake Capturing Methodology Using Fully Coupled CFD and Partical Vortex Transport Method," *American Helicopter Society International*, vol. 1, p. 333, 2006.
- [10] J. Leishman, M. J. Bhagwat and S. Ananthan, "Free-Vortex Wake Predictions of the Vortex Ring State For Single-Rotor and Multi-Rotor Configurations," in *58th Annual Forum and Technology Display of the American Helicopter Society International*, Montreal, Canada, 2002.
- [11] D. Pitt and D. Peters, "Rotor Dynamic Inflow Derivatives and Time Constants from Various Inflow Models," in *9th European Rotorcraft Forum*, Stresa, Italy, 1983.
- [12] W. Press, S. A. Reukolsku, W. T. Vetterling and B. P. Flannery, *Numerical Recipes in Fortran*, New York, NY: Cambridge University Press, 1992, pp. 644-649.
- [13] G. Flandro, H. M. McMahon and R. L. Roach, *Basic Aerodynamics, Incompressible Flow*, Cambridge, MA: Cambridge University Press, 2011, pp. 320-321.
- [14] W. Z. Stepniewsky and C. N. Keys, *Rotary-Wing Aerodynamics*, New York: Dover Publications, Inc., 2015.

- [15] F. Harris, "Articulated Rotor Blade Flapping Motion at Low Advance Ratio," *Journal of the American Helicopter Society*, pp. 41-48, 1972.
- [16] R. W. Prouty, *Helicopter Performance, Stability, and Control*, Malabar, FL: Krieger Publishing Company, 2002.
- [17] M. Dreier, *Introduction to Helicopter and Tiltrotor Flight Simulation*, 2nd ed., American Institute of Aeronautics and Astronautics, Inc., 2017.
- [18] "*Rotor Wake Visualization At Full Scale Flight Conditions*". [Film]. Advanced Technologies Incorporated.
- [19] W. Johnson, "Model for Vortex Ring State Influence on Rotorcraft Flight Dynamics," in *AHS 4th Decennial Specialist's Conference on Aeromechanics, American Helicopter Society, International*, Moffet Field, California, 2004.

# Appendix A

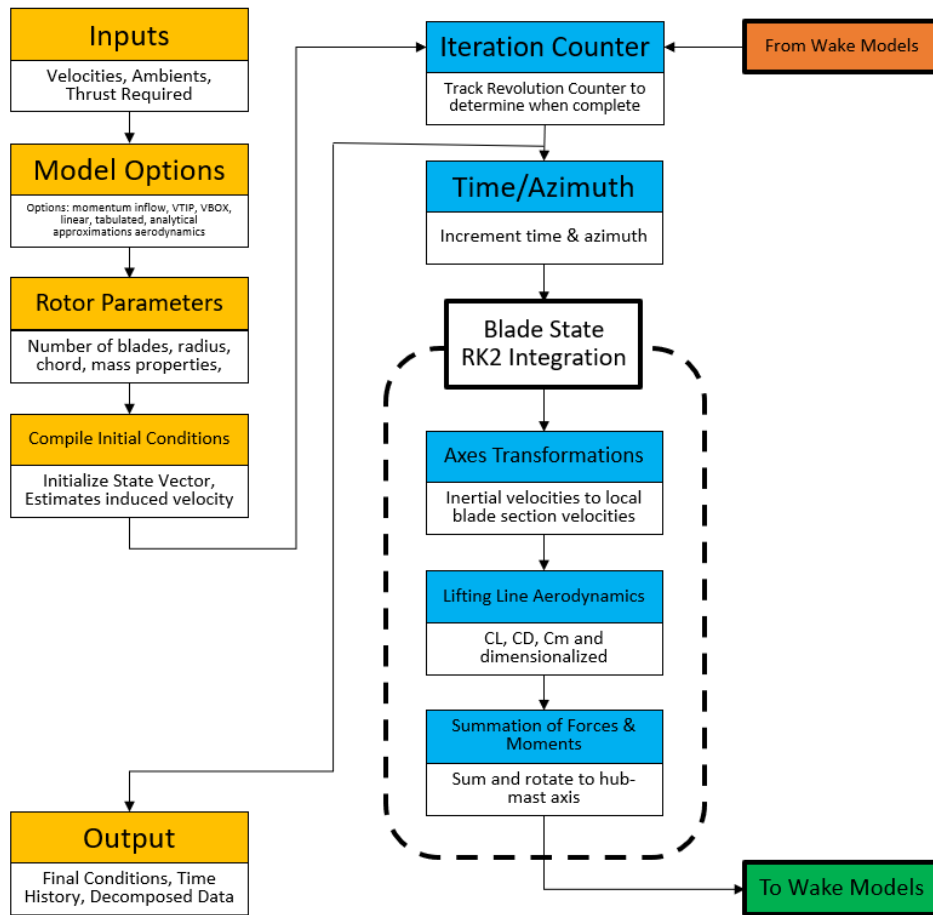


Figure 22. Program Detailed Flow Chart (1 of 2)

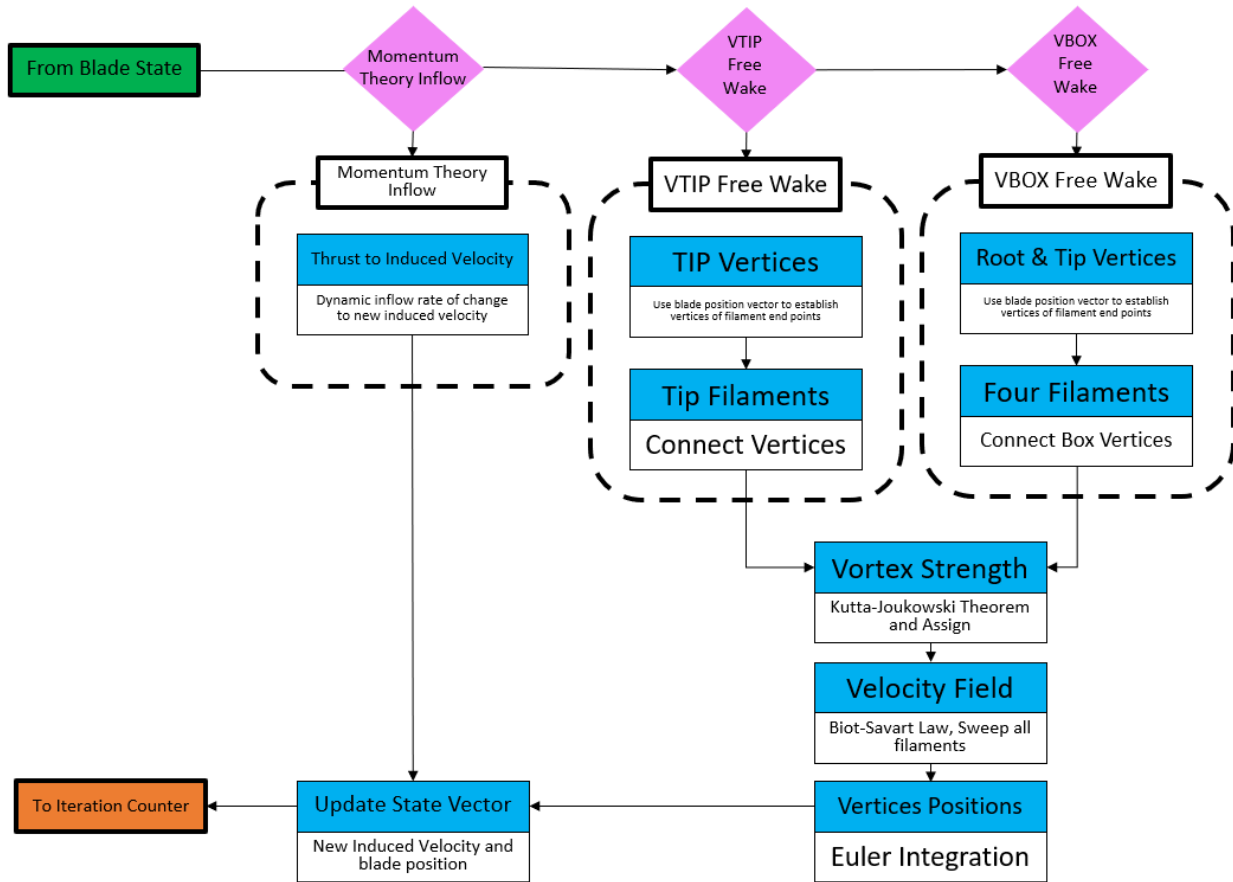


Figure 23. Program Detailed Flow Chart (2 of 2)

## Appendix B

### Main Program File

```
% Driver BER w/ Free-Wake VBOX/VTIP
fclose('all');clearvars; close all; clc;

%% Set up globals

global Conversion_const
global atmos_const
global washID rotor
global col
global aerocoefs
global iLinAero
global VERTEX
global VBOX

%% Load pre-defined constants/conversion factors
col = ['r';'b';'m';'k'];           %Color names for four blades

% Get Conversion Constants
Conversion_factors;               %Initialize conversions
kts_to_fps = Conversion_const(1);
rpm_to_rps = Conversion_const(2);
r2d         = Conversion_const(3);
d2r         = Conversion_const(4);
in2ft       = Conversion_const(5);
TDsf        = Conversion_const(6);
grav        = Conversion_const(7);
lbmin2_to_slugft2 = Conversion_const(8);

%% Flight Condition Inputs

U_input = 50 * kts_to_fps;        % fps = kts * kts_to_fps
V_input = 0 * kts_to_fps;        % fps = kts * kts_to_fps
W_input = 0 * kts_to_fps;        % fps = kts * kts_to_fps ||| (+) is
descent
p_input = 0 * d2r;               % rad/sec = deg/sec * d2r
q_input = 0 * d2r;               % rad/sec = deg/sec * d2r
r_input = 0 * d2r;               % rad/sec = deg/sec * d2r
Thrust_required = 6000;          % lbf
Palt      = 0;                   % ft, Pressure Altitude
Tamb      = 0;                   % deg F
Tdelt     = 0;                   % deg C

% Wake washID
washID = 2;      % 1 = free_VBOX
               % 2 = free_VTIP
if washID == 1
    wake_str = 'free_VBOX';
elseif washID == 2
    wake_str = 'free_VTIP';
end

% Airfoil Aerodynamics
iLinAero = 3;    % 1 = linearer
```

```

        % 2 = non-linear (look-up table)
        % 3 = analytical aero
%% Initialize Atmosphere

[ rho, pressure, tempf, tempr, Vsound, mu_atm, nu_atm, ...
  theta_atm, delta_atm, sigmap_atm ] = Atmosphere( Palt, Tamb, Tdelt );
atmos_const = [rho;Vsound;sigmap_atm;nu_atm];
qdyn = 0.5*rho*(U_input^2+W_input^2);

%% Initialize clcdcm model

[aoamax,clmax,clp,aoacdmx,cd0,cd2,cdmax,cl0,cm0,toc,zoc] = make_0012_a();
[aerocoefs] = init_clcdcm4_a(aoamax,clmax,clp,aoacdmx,cd0,cd2,cdmax,cl0,...
  cm0,toc,zoc);

%% Define Rotor Geometry

[rotor] = get_Rotor_Definiton2(Thrust_required);

%% Define equivalent flat plat drag areas

fex = 16; % equivalent flat plate area, x-direction
fey = 0; % equivalent flat plate area, y-direction
fx_required = qdyn*fex; % desired Xf
fy_required = qdyn*fey; % desired Yf

%% Process Inputs & Establish Initial Conditions

vi_body = [U_input;V_input;W_input;p_input;q_input;r_input];
Oz = [0;0;-rotor.om];
Vi = vi_body(1:3);

cont = [10;0;0]*d2r; % Th0, A1s, B1s
%sv = [beta_1, betad_1, beta_2, betad_2,...beta_nb, betad_nb,
inflow(ww)]]
sv = zeros(2*rotor.nb+1,1); % Initialize state vectors
svd = zeros(2*rotor.nb+1,1); % Init state vector
svdn = zeros(2*rotor.nb+1,1); % Init state vector

Fmast = zeros(3,1); % Zero accumulator
Mmast = zeros(3,1); % Zero accumulator
Mmastind= zeros(3,1); % Zero accumulator
Mmastpro= zeros(3,1); % Zero accumulator
Fmast(3)= -Thrust_required; % Load Mast force guess

ww = sqrt(Thrust_required/(2*rho*rotor.area)); %Hover induced velocity
sv(2*rotor.nb+1) = ww; % Initialize state vector
mu = Vi(1)/rotor.Vtip; % Advance ratio

%% Establish loop variables

naz = 24; % Number of azimuth steps per
rotation
nrev = 14; % Number of total revolutions
nloops = nrev*naz + 1; % Total number of azimuth steps

```



```

nstab = 2; % Number of revs to stabilize
thrust transient
period = 2*pi/rotor.om; % Rotor period, sec
dt = period/naz; % Time step, sec
daz = rotor.om*dt; % Azimuth Step
Vw = [zeros(2,rotor.nars);ones(1,rotor.nars)*ww]; % Init induced
velocity, 1 blade
% [0 0 0 ...0; {u_w_iars}
% [0 0 0 ...0; {v_w_iars} {{Just writing out for a mental
picture!}}
% [ww ww ww ...ww]; {w_w_iars}

% CLAW

lama = (Vi(3)-ww)/rotor.Vtip; %Inflow ratio
Vt = sqrt(mu^2+lama^2); %Total velocity (ND)
Cz_denom = rho*rotor.area*rotor.Vtip^2; %Non-dimensionalizing factor
Ct_des = -Thrust_required/Cz_denom; %Desired thrust coefficient
[dctdth,BA11,BA12,BA21,BA22,mut] = CLAW_init4(Ct_des,rotor,naz); %Initialize
CLAWS
dczdth = interp1(mut,dctdth,mu)/4; %Select CLAW for this speed
BA11mu = interp1(mut,BA11,mu); %d(A1)/d(a1)
BA12mu = interp1(mut,BA12,mu); %d(A1)/d(b1)
BA21mu = interp1(mut,BA21,mu); %d(B1)/d(a1)
BA22mu = interp1(mut,BA22,mu); %d(B1)/d(b1)
chi = atan2(0.2,mu); %Not really chi

% Filter initialization, used to smooth control inputs from the CLAWS
dth0 = 0; %last pass dth0
dth0l = 0; %last pass dth0
dth0ll = 0; %last-last pass dth0
dth0rl = 0; %last pass dth0r
dth0rll = 0; %last-last pass dth0r
dA1 = 0; %last pass lat cyclic
dA1rl = 0; %last pass dA1r
dB1 = 0; %last pass F/A cyclic
dB1rl = 0; %last pass dB1rl
Czel = 0; %last pass Cze
Czell = 0; %last-last pass Cze
forder = 1;
accel = 1.0; %Accelerate the wwd eqn for
trimming

% Filter Properties
global cf11 cf12 cf13
global cf21 cf22 cf23 cf24 cf25
tau = period/2; %Time constant
[cf11,cf12,cf13] = init_filter1(tau,dt); %Initialize tau*s+1 filter
wn = rotor.om;
zeta = 0.71;
[cf21, cf22, cf23, cf24, cf25] = init_filter2(wn,zeta,dt);

%% Time loop for momentum wake model

resmom = zeros(naz*nrev,(2*rotor.nb+17));
for iaz = 1:naz*nrev %Advance naz*nrev az steps
    psi = (iaz-1)*daz; %Rotor azimuth angle

```

```

    resmom(iaz,:) = [(psi*r2d) sv' Fmast' Mmast' Mmastind' Mmastpro'
cont'*r2d]; %Store current values
    Cz = Fmast(3)/Cz_denom; %Actual Cz
    Cze = Cz - Ct_des; %Cz error
    dth0r = -Cze/dczdth;

    Xferr = (Fmast(1) - fx_required);
    Yferr = (Fmast(2) - fy_required);
    da1 = Xferr/Thrust_required;
    db1 = -Yferr/Thrust_required;
    dA1r = BA11mu*da1 + BA12mu*db1;
    dB1r = BA21mu*da1 + BA22mu*db1;

    [dth0,dth0r1] = CLAWS2(dth0,dth0r,dth0r1);
    [dB1,dB1r1] = CLAWS2(dB1,dB1r,dB1r1);
    [dA1,dA1r1] = CLAWS2(dA1,dA1r,dA1r1);

    cont(1) = cont(1) + dth0; %Update collective
    if mu > 0.01
        cont(2) = cont(2) + dA1; %Update lateral cyclic
        cont(3) = cont(3) + dB1; %Update fore/aft cyclic
    end

    ww = sv(2*rotor.nb+1); %Current value of ww
    Vw(3,:) = ones(1,rotor.nars)*ww; %Load uniform wake value
    Fmast = zeros(3,1); %Init Fmast accumulator
    Mmast = zeros(3,1); %Init Fmast accumulator
    Mmastind = zeros(3,1); %Init Fmast accumulator
    Mmastpro = zeros(3,1); %Init Fmast accumulator

    for ib = 1:rotor.nb %Cycle over all blades
        ibeta = (ib-1)*2 + 1; %Ptr to flapping angle
        ibetad = ibeta + 1; %Ptr to flapping rate
        svi = sv(ibeta:ibetad); %Load blade state vector
        [svdi, Fmast_i, Mmast_i, ~, ~, ~, Mmastindi, Mmastproi] = blade2(svi,
cont, psi, Vi, Vw, ib);
        Fmast = Fmast + Fmast_i; %
        Mmast = Mmast + Mmast_i; %
        Mmastind = Mmastind + Mmastindi;
        Mmastpro = Mmastpro + Mmastproi;
        svd(ibeta:ibetad,1) = svdi; %Load overall state deriv
    end
    vtot = sqrt(Vi(1)^2+(Vi(3)-ww)^2); %Total velocity, center of
disk
    wwd = accel*(-Fmast(3)-2*rho*rotor.area*vtot*ww)/rotor.am;
    svd(2*rotor.nb+1) = wwd; %Load wwd into state vector
deriv

    svn = sv + svd*dt; %Take Euler step

    wwn = svn(2*rotor.nb+1);
    Vw(3,:) = ones(1,rotor.nars)*wwn; %Load predictor value of ww
    psin = iaz*daz; %Point to next azimuth
    Fmastn = zeros(3,1);
    Mmastn = zeros(3,1);
    for ib = 1:rotor.nb %Cycle over all blades
        ibeta = (ib-1)*2 + 1; %Ptr to flapping angle

```

```

        ibetad = ibeta + 1; %Ptr to flapping rate
        svni = svn(ibeta:ibetad); %Load blade state vector
        [svdni, Fmastni, Mmastni, ~, ~, ~, ~] = blade2(svni, cont, psin, Vi,
Vw, ib);
        Fmastn = Fmastn + Fmastni; %
        Mmastn = Mmastn + Mmastni; %
        svdn(ibeta:ibetad,1) = svdni; %Load overall state deriv
    end
    vtot = sqrt(Vi(1)^2+(Vi(3)-wwn)^2); %Total vel, center of disk
    wwd = accel*(-Fmastn(3)-2*rho*rotor.area*vtot*wwn)/rotor.am;
    svdn(2*rotor.nb+1) = wwd; %load wwd into svdn.

    sv = sv + (svd+svdn)*dt/2; %Take improved Euler step

end

% Free-wake Implementation

% Clean out variables previously used and define more for free-wake
wwb = zeros(nloops,1); %Equivalent ww time history
sv = zeros(2*rotor.nb+1,1); %Init state vector
svd = zeros(2*rotor.nb+1,1); %Init state vector
svdn = zeros(2*rotor.nb+1,1); %Init state vector
Vws = zeros(3,rotor.nars,rotor.nb); %Storage for induced velocity
Vwsn = zeros(3,rotor.nars,rotor.nb); %Storage for next step
induced velocity
PBRLb = zeros(3,rotor.nars,rotor.nb); %Storage for blades position
flapb = zeros(3,nloops); %Storage for flapping filter
rflap = zeros(3,1); %Storage for flapping filter
rflapl = zeros(3,1); %Storage for flapping filter
psib = zeros(nloops,1); %Storage for flapping filter
em_vel = zeros(3,rotor.nars,4); %Distributed wake values
Tavg = -Fmast(3); %Init moving average on T.
Tavgn = 0; % Moving average T
TP_vel = zeros(3,21,21); % Treffetz plane velocity
TP_pos = zeros(3,21,21); % Treffetz plane position
gambi = zeros(rotor.nb,1); % Storage for circulation
gambib = zeros(nloops,rotor.nb); % Time History Storage for
gambi, vortex strength
res = zeros(nloops,2*rotor.nb+8+6+3); % Time storage for states
+6 for induced and profile moments
nbox = 0; % Init # of VBOX counters
Vw = [zeros(2,rotor.nars);ones(1,rotor.nars)*ww]; % Init induced
velocity, 1 blade
nrvkp = nrev; % Number of rev worth of
filament/VBOXs to use for ww calculations - CWM 20 MAY 2022
nrvplt = nrvkp;%5; % Number of revs to
plot, now separate from the kept revs - CWM 22 AUG 2022

%% Initialize Wake Variables

[VERTEX,VBOX,maxVRTX,maxVBOX] = init_vertices_VBOX3(rotor.nb,nrev,naz);
rcore = rotor.rcore; %Vortex core radius

%% Initialize the state vector with the results from the momentum model
% above. Calculate beta and beta_dot for each blade when psi0 = 0.
% Also, place ww0 into the last sv position.

```

```

for ib = 1:rotor.nb
    psib = 2*pi/rotor.nb*(ib-1);
    csy = cos(psib);
    ssy = sin(psib);
    ibeta = 2*(ib-1)+1;
    ibetad = ibeta + 1;
    sv(ibeta) = a01 + csy*acoef1(1) + ssy*bcoef1(1);
    sv(ibetad) = -rotor.om*ssy*acoef1(1) + rotor.om*csy*bcoef1(1);
end
sv(rotor.nb*2+1) = ww0;
%% Whirl the Rotor

for iaz = 1:nloops

    TFS = (nstab*naz-iaaz+1)/(nstab*naz);
    TFS = max([min([TFS 1]) 0]);
    TFSm1 = 1-TFS;

    psi = (iaaz-1)*daz; %Rotor azimuth angle
    Tpsi = make_rz(-psi); %Tpsi matrix

% Calculate Cz error and collective correction. Smooth with a filter.

    Cz = Fmast(3)/Cz_denom; %Actual Cz
    Cze = Cz - Ct_des; %Cz error
    dth0r = -Cze/dczdth;

    Xferr = (Fmast(1) - fx_required);
    Yferr = (Fmast(2) - fy_required);
    da1 = Xferr/Thrust_required;
    db1 = -Yferr/Thrust_required;
    dA1r = BA11mu*da1 + BA12mu*db1;
    dB1r = BA21mu*da1 + BA22mu*db1;

    [dth0,dth0rl] = CLAWS2(dth0,dth0r,dth0rl);
    [dB1,dB1rl] = CLAWS2(dB1,dB1r,dB1rl);
    [dA1,dA1rl] = CLAWS2(dA1,dA1r,dA1rl);

    cont(1) = cont(1) + dth0; %Update collective
    if mu > 0.01
        cont(2) = cont(2) + dA1; %Update lateral cyclic
        cont(3) = cont(3) + dB1; %Update fore/aft cyclic
    end

% Save flapping for time domain presentation.

    res(iaaz,:) = [(psi/2/pi) sv' Fmast' Mmast' Mmastind' Mmastpro'
cont'*r2d]; %Store states and psi/2/pi

    Fmast = zeros(3,1); %Zero accumulator
    Mmast = zeros(3,1); %Zero accumulator
    Mmastind = zeros(3,1); %Zero accumulator
    Mmastpro = zeros(3,1); %Zero accumulator

    for ib = 1:rotor.nb %Cycle over all blades

```

```

        ibeta = (ib-1)*2 + 1;           %Ptr to flapping angle
        ibetad = ibeta + 1;           %Ptr to flapping rate
        svi = sv(ibeta:ibetad);       %Load blade state vector

        Vw = [zeros(2,rotor.nars);ones(1,rotor.nars)*ww0]*TFS +
Vws(:, :, ib)*TFSm1;

        [svdi, Fmasti, Mmasti, PBRL, gamb, dFar, Mmastindi, Mmastproi] =
blade2(svi, cont, psi, Vi, Vw, ib);
        gambi(ib) = gamb;
        gambib(iaz,ib) = gamb;       %Storage for Time History

        Fmast = Fmast + Fmasti;
        Mmast = Mmast + Mmasti;
        Mmastind = Mmastind + Mmastindi;
        Mmastpro = Mmastpro + Mmastproi;
        PBRLb(:, :, ib) = PBRL;      %BRL in rotating axes.
        svd(ibeta:ibetad,1) = svdi;  %Load global state deriv
    end

    svn = sv + svd*dt;               %Euler step, part 1 of RK2

    psin = iaz*daz;                 %Point to next azimuth
    for ib = 1:rotor.nb
        Vw = [zeros(2,rotor.nars);ones(1,rotor.nars)*ww0]*TFS +
Vwsn(:, :, ib)*TFSm1;
        ibeta = (ib-1)*2 + 1;       %Ptr to flapping angle
        ibetad = ibeta + 1;         %Ptr to flapping rate
        svi = svn(ibeta:ibetad);    %Load blade state vector
        [svdi, ~, ~, ~, ~, ~, ~, ~] = blade2(svi, cont, psin, Vi, Vw, ib);
        svdn(ibeta:ibetad,1) = svdi; %Load global state deriv
    end

    sv = sv + (svd+svdn)*dt/2;      %Part 2 of RK2

% Get vertices for vortex boxes. Use the blade tip and inboard location
% on the leading edge to define the leading edge of the vortex box.

    for ib = 1:rotor.nb
        del_local = (ib-1)*rotor.dels; %Blade spacing
        Tdel = make_rz(-del_local);    %Blade spacing matrix
        P_outboard = PBRLb(:, rotor.nars, ib); %Outboard vertex position
        P_inboard = PBRLb(:, 2, ib);   %Inboard vertex position
        vrtxp2 = (Tdel*Tpsi)'*P_outboard; %v2 in fixed system
        vrtxp1 = (Tdel*Tpsi)'*P_inboard; %v1 in fixed system
        VERTEX(iaz, ib, 1).P = vrtxp1; %Store inboard position
        VERTEX(iaz, ib, 2).P = vrtxp2; %Store outboard position
        VBOX(iaz, ib).gam = gambi(ib);
    end

    if mod(iaz, naz) == 0
        Tavgn = Tavgn;
        Tavgn = 0;
    else
        Tavgn = Tavgn - Fmast(3)/naz;
    end
end

```

```

if iaz > 1

% Determine index of VBOX to start calculating induced velocities based
% on nrvkp*naz.

    if iaz > nrvkp*naz
        nstart = iaz - nrvkp*naz;
    else
        nstart = 1;
    end
    if iaz > nrvplt*naz
        nstart_plot = iaz - nrvplt*naz;
    else
        nstart_plot = 1;
    end

% Calculate the velocity field at all vertices not on the blade.

    nbox = iaz;
    for jb = 1:rotor.nb
        for ivrtx = 1:(iaz-1)
            for irs = 1:2
                Ptarget = VERTEX(ivrtx,jb,irs).P;
                VERTEX(ivrtx,jb,irs).V =
get_target_vel5(Ptarget,nbox,rotor.nb,rcore,nstart);
            end
        end
    end

% Calculate the velocity field at all stations on all blades.

    Vws = zeros(3,rotor.nars,rotor.nb);
    for jb = 1:rotor.nb
        del_local = (jb-1)*rotor.dels;           %Blade spacing
        Tdel = make_rz(-del_local);             %Blade spacing matrix
        for irs = 1:rotor.nars
            Pb = PBRLb(:,irs,jb);
            Ptarget = (Tdel*Tpsi)'*Pb;          %v2 in fixed system
            Vws(:,irs,jb) =
get_target_vel5(Ptarget,nbox,rotor.nb,rcore,nstart);
        end
    end

    Vwsn = zeros(3,rotor.nars,rotor.nb);
    for jb = 1:rotor.nb
        del_local = (jb-1)*rotor.dels;           %Blade spacing
        Tdel = make_rz(-del_local-daz);         %Blade spacing matrix +
one step ahead for Euler Correction
        for irs = 1:rotor.nars
            Pb = PBRLb(:,irs,jb);
            Ptarget = (Tdel*Tpsi)'*Pb;          %v2 in fixed system
            Vwsn(:,irs,jb) =
get_target_vel5(Ptarget,nbox,rotor.nb,rcore,nstart);
        end
    end

% Calculate the velocity field at cardinal points to calculate the

```

```

% equivalent momentum theory wake values.

    for jpsi = 1:4
        jps = -(jpsi-1)*90*d2r;
        cj = cos(jps);
        sj = sin(jps);
        Tjps = [cj sj 0;-sj cj 0;0 0 1];
        for irs = 1:rotor.nars
            BRL = [-rotor.sa(irs);0;0];
            Ptarget = Tjps*BRL;
            em_vel(:,irs,jpsi) =
get_target_vel5(Ptarget,nbox,rotor.nb,rcore,nstart);
            end
        end
        wwb(iaz) = make_mom_equiv1(rotor.sa,em_vel)*TFSm1 + ww0*TFS;

    end

% Take an Euler step with the V vectors.

    for jb = 1:rotor.nb
        for ivp = 1:iaz
            for irs = 1:2
                fc = VERTEX(ivp,jb,irs).V-Vi;
                fcl = VERTEX(ivp,jb,irs).Vl-Vi;
                VERTEX(ivp,jb,irs).P = VERTEX(ivp,jb,irs).P + (fc+fcl)*dt/2;
                VERTEX(ivp,jb,irs).Vl = VERTEX(ivp,jb,irs).V;
            end
        end
    end
end
end
end

```

## Blade Dynamic and Aerodynamic Script

```
function [svd, Fmast, Mmast, PBRL, gamb, dFar, Mmast_ind, Mmast_pro] =
blade2(sv, cont, psi, Vmm, Vw, ib)
global rotor
global atmos_const
global Conversion_const
global aerocoefs
global iLinAero

om      = rotor.om;
sa      = rotor.sa;
R       = rotor.R;
nars    = rotor.nars;
kbeta  = rotor.kbeta;
c       = rotor.c;
Iblade = rotor.Iblade;
Clmax   = rotor.Clmax;
Cdavg   = rotor.Cdavg;
a0      = rotor.a0 ;
tht     = rotor.tht;
Btip    = rotor.Btip;
cutout  = rotor.cutout;
evk     = rotor.evk;
rho     = atmos_const(1);
Vsound  = atmos_const(2);
nu      = atmos_const(4);
r2d     = Conversion_const(3);
d2r     = Conversion_const(4);

th0 = cont(1);           %Collective
A1  = cont(2);           %Lateral cyclic
B1  = cont(3);           %Longitudinal cyclic

beta = sv(1);            %Flapping angle
betad = sv(2);           %Flapping rate

%   Get the azimuth angle, Tpsi matrix and blade spacing matrix

del  = (ib-1)*rotor.dels;
Tdel = make_rz(-del);
Tpsi = Tdel*make_rz(-psi);

%   Resolve inertial velocity to rotor axes

cb = cos(beta);
sb = sin(beta);

omr = [0;0;-om];
Oahh = Tpsi*omr;

%   Initialize distributed force, moment, flapping moment and blade BRL

dFar = zeros(3,nars);
```



```

dMar = zeros(3,nars);
dMyar = zeros(nars,1);
PBRL = zeros(3,nars);
Utb = zeros(nars,1);
dFarind = zeros(3,nars);
dFarpro = zeros(3,nars);
dMarind = zeros(3,nars);
dMarpro = zeros(3,nars);
tht_b = zeros(nars,1);
alpha_degb = zeros(nars,1);

% Evaluate distributed forces and moments along the blade

dth = th0 + A1*cos(psi+del) + B1*sin(psi+del);
for irs = 1:nars

    P0 = [-sa(irs);0;0];
    P = make_ry(beta)*P0;
    PBRL(:,irs) = P; %Store PBRL
    Pdot = [sa(irs)*sb;0;-sa(irs)*cb]*betad; %d(BRL)/dt

    phix0 = -tht*sa(irs)/R - dth;
    phiy0 = -beta;
    phiz0 = 0;
    rx0 = make_rx(phix0);
    ry0 = make_ry(phiy0);
    rz0 = make_rz(phiz0);
    rbr = rx0*ry0*rz0;

    Vamm = Vmm - [0;0;Vw(3,irs)];
    Vahb = Tpsi*Vamm;
    Vahbf = Vahb + cross(Oahh,P) + Pdot;
    Vafbf = rbr*Vahbf;

    Ut = Vafbf(2); %Tangential velocity
    Up = Vafbf(3); %Perpendicular velocity
    vt2 = Ut*Ut + Up*Up;
    machno = sqrt(vt2)/Vsound;
    qtotc = 0.5*rho*vt2*c;
    alfa = atan2(Up,Ut);
    alpha_deg = alfa*r2d;
    alpha_degb(irs) = min([max([alpha_deg -180]) 180]);

    if iLinAero
        Cl = min([max([a0*alfa; -Clmax]) Clmax]);
        Cd = Cdavg;
        Cm = 0;
    elseif iLinAero == 2
        [Cl,Cd] = clcd0012(machno,alpha_deg);
        Cm = 0;
    elseif iLinAero == 3
        [Cl, Cd, Cm] = clcdcm8_a(alpha_deg, aerocoefs, d2r);
    end

    if sa(irs)<=cutout
        Cl = 0;
        Cd = 0.04;
    end
end

```

```

end

if sa(irs) > Btip*R
    Cl = 0;
end

dlift = qtotc*Cl;
ddrag = qtotc*Cd;
dmom  = qtotc*c*Cm;
dfa   = [0;-ddrag;-dlift];
dfaind= [0;0;-dlift];
dfapro= [0;-ddrag;0];
dma   = [dmom;0;0];
rxma  = make_rx(-alfa);

dFar(:,irs) = rbr'*rxma*dfa;
dMar(:,irs) = cross(P0,dFar(:,irs)) + dma;
dMyar(irs)  = dMar(2,irs);

dFarind(:,irs) = rbr'*rxma*dfaind;
dFarpro(:,irs) = rbr'*rxma*dfapro;
dMarind(:,irs) = cross(P0,dFarind(:,irs));
dMarpro(:,irs) = cross(P0,dFarpro(:,irs));

Utb(irs) = Ut; %Store for interpolation later

end

% Integration dFar and dMar(y). Resolve the forces into mast axes.

Far = [0;0;0];
Mar = [0;0;0];
Mya = 0;
Far_ind = [0;0;0];
Mar_ind = [0;0;0];
Mar_pro = [0;0;0];

for irs = 2:nars
    ds = sa(irs)-sa(irs-1);
    Far = Far + (dFar(:,irs)+dFar(:,irs-1))*ds/2;
    Mar = Mar + (dMar(:,irs)+dMar(:,irs-1))*ds/2;
    Mya = Mya + (dMyar(irs)+dMyar(irs-1))*ds/2;

    Far_ind = Far_ind + (dFarind(:,irs)+dFarind(:,irs-1))*ds/2;
    Mar_ind = Mar_ind + (dMarind(:,irs)+dMarind(:,irs-1))*ds/2;
    Mar_pro = Mar_pro + (dMarpro(:,irs)+dMarpro(:,irs-1))*ds/2;
end
Fmast = Tpsi'*Far;
Mmast = Tpsi'*Mar;

Mmast_ind = Tpsi'*Mar_ind;
Mmast_pro = Tpsi'*Mar_pro;

% Load the derivative of the state vector.

svd(1) = betad;

```

```
svd(2) = -(om*om+kbeta/Iblade)*beta - Mya/Iblade;
```

```
d1r = 2*Far(3)/R;
```

```
gamb = -d1r/(rho*Ut);
```

```
end
```

### Wake Vertex Structure

```
function [VERTEX,VBOX,maxVRTX,maxVBOX] = init_vertices_VBOX3(nb,nrev,naz)
% Initialize the vertex structure - VERTEX is a structure that contains
% two vectors, V and P (velocity and position).

maxVRTX      = (naz*nrev+1);
vrtxs.V      = [0;0;0];           % Velocity
vrtxs.V1     = [0;0;0];           % Velocity
vrtxs.P      = [0;0;0];           % Position
VERTEX       = repmat(vrtxs,maxVRTX,nb,2);
maxVBOX      = naz*nrev;
vboxs.gam    = 1;
VBOX         = repmat(vboxs,maxVBOX,nb);

gamma        = 1;
for iaz = 2:(naz*nrev+1)
    ibox = iaz-1;
    for ib = 1:nb
        VBOX(ibox,ib).gam = gamma;
    end
end
end
end
```

### Wake Velocities Calculation

```
function [vtar] = get_target_vel5(Ptarget,nBOX,nb,rcore,nstart)
global VBOX
global VERTEX
global washID

vtar = 0;
startat = nstart + 1;
for ib = 1:nb
    for iBOX = startat:nBOX
        gam = VBOX(iBOX,ib).gam;
        vp1 = VERTEX(iBOX, ib,1).P;
        vp2 = VERTEX(iBOX, ib,2).P;
        vp3 = VERTEX(iBOX-1,ib,2).P;
        vp4 = VERTEX(iBOX-1,ib,1).P;
        vtar = vtar + BiotSavart(vp2, vp3, Ptarget, gam, rcore);
        if washID == 1
            vtar = vtar + BiotSavart(vp1, vp2, Ptarget, gam, rcore);
            vtar = vtar + BiotSavart(vp3, vp4, Ptarget, gam, rcore);
            vtar = vtar + BiotSavart(vp4, vp1, Ptarget, gam, rcore);
        end
    end
end
end
end
```