ROTOR PERFORMANCE MODEL (RPM)

By

Chadwick Scott Balfantz, Jr

THESIS

Submitted in partial fulfillment of the requirements for the degree of

Master of Science in Mechanical Engineering

at The University of Texas at Arlington

August 2021

Arlington, Texas

Supervising Committee:
Dr. Dudley E. Smith, Advisor
Mr. Mark E. Dreier, Co-Advisor
Dr. Robert Mullins
Dr. Donald R. Wilson

ABSTRACT

ROTOR PERFORMANCE MODEL

Chadwick Scott Balfantz, Jr

The University of Texas at Arlington, 2021

Advising Professor: Dr. Dudley E. Smith

Co-Advisor: Mr. Mark E. Dreier

Rotorcraft performance modeling is of high interest with multiple industry codes in existence, many of which are high-fidelity and are comprehensive in the analyses that are achievable. Along with the high-fidelity and the comprehensive nature of the industry codes comes an equally high level of complexity in the inputs required to extract meaningful results. The ever-increasing fidelity in the existing codes creates a need for an analysis code that is built to be highly flexible, designed from the top down and implemented from the bottom up in a structured format. The Rotor Performance Model (RPM) was developed to provide high-fidelity performance analysis options without the cost of highly complex input parameters for the rotor. To date, the code was developed with the goal of being well documented, modular, and physics-based to provide the use of macroscopic rotor input parameters and the framework to build upon with future work – all without the need for a complete re-work of the code infrastructure. With the implementation of the dynamic inflow models of Pitt-Peters, and Peters et.al., combined with a hybrid periodic shooting/Newton-Raphson technique, the blade motion trim time and closure of the thrust/induced velocity has been accelerated in the current analysis program without the necessity of a wealth of rotor input data. Significant results include nearly identical results from theory and the RPM code for the inflow velocity in hover and forward airspeed and good qualitative and quantitative power

required calculated values between analytical three-term equations, the RPM code, and test data. Additionally, the RPM code allows for modeling of a trimmed rotor using control other than lateral cyclic.

ACKNOWLEDGEMENTS

I would like to thank Dr. Dudley E. Smith and Mr. Mark Dreier for the constant support and numerous hours spent guiding me along this journey. The value added along the way is something books and articles can never replace and for this I am forever grateful. I would also like to thank my committee members for their support and recommendations.

I would like to thank my parents for the patience and willingness to support my endeavor to attend Graduate school and the never-ending belief in me to complete the degree while hosting a myriad of other responsibilities outside of academics.

Table of Contents

List of Tables

List of Figures

List of Symbols

Advance Ratio	μ
Azimuth Angle	ψ
Blade Coning, Anhedral, Sweep	ζ, τ, ν
Blade Precone, Pre-anhedral, Pre-sweep	ζ_0, τ_0, v_0
Blade Collective Angle	θ_0
Blade Manufactured Twist	$\theta(x)_T$ or θ_{twist}
Blade Pitch Angle	θ
Blade Section Angle of Attack	α
Blade Section Inflow Angle	ϕ
Coefficient of Drag, Average	$C_D, \overline{C_D}$
Coefficient of Lift	C_L
Coefficient of Pitch Moment	C_M
Coefficient of Thrust	C_T
Delta-3	δ_3
Euler Angles (Roll, Pitch, Yaw)	ϕ, θ, ψ
Flapping Angle, Rate, Acceleration	$\beta, \dot{\beta}, \ddot{\beta}$
Flapping Hinge Offset	e_f
Flapping Inertia	I_b
Flapping Inertia – Spin Stiffening	I_e
Hub Spring	K_β
Inflow Ratio	λ
Inflow Skew Angle	χ
Lateral Cyclic	A_1
Lateral Flapping Angle	b_1
Lead-Lag Angle, Rate, Acceleration	$\gamma, \dot{\gamma}, \ddot{\gamma}$
Lead-Lag Hinge Offset	e_L
Linear Lift-Curve Slope,	a_0
Longitudinal Cyclic	B_1
Longitudinal Flapping	a_1

Normalized Radial Position	x
Radial Position, Rate, Acceleration	$r, \dot{r}, \ddot{r}$
Tip Path Plane	TPP

Subscripts

Hub	h
Rotating Frame	r
Flapping Frame	f
Lead-Lag Frame	ll
Inertial	i
Body	b
Aerodynamic	a
Wash	w
Section	s

CHAPTER 1.BACKGROUND

1.1 What Does Blade Element Rotor Model Mean?

Gessow and Myers, and McCormick [1, 2] describe the "blade-element" theory where each blade section, or element on a blade, is treated as a two-dimensional airfoil which follows a helical path. Similar to the aerodynamic strip theory used in the analysis of three-dimensional wings, blade-element theory uses an infinite number of radial blade segments that have finite chords and are infinitesimally narrow in width normal to the blade reference line. These elements allow for the calculation of the distributed lift, drag, and torque contributions along a blade which are then integrated analytically. Drzewiecki first successfully presented this and the progression to numerical integration of a finite number of macroscopic blade segments. Unfortunately, the use of the phrase "blade-element" has come to mean the numerical method of using a finite number of discrete blade segments and has lost the original meaning of a continuous analytical model of a blade. Perhaps a better description of the analysis process is "finite-element method," since the name references a widely used numerical method to solve many of the partial differential equations in engineering. However, since the rotorcraft industry uses "blade-element" to describe the analysis method as well as imply the numerical method employed, the use of "blade-element" will be used in this report as context will make clear which of the two definitions apply at any given point.

1.2 Overview of Industry Rotor Model

Many rotor models exist within the rotorcraft industry, each of which have commendable strengths and important limitations. Several of the existing industry codes are listed below with brief explanations of the model capabilities.

1.2.1 2GCHAS – RCAS

2GCHAS, Second Generation Comprehensive Helicopter Analysis System, (often referred to as 2G Charlie), originated from a failure to predict rotor system aerodynamics and dynamics resulting in the US Army necessitating a tool for comprehensive analysis capabilities [3]. During the mid-1970's, the objectives of the Army were to overcome technical limitations of first-generation rotorcraft analyses, provide multi-disciplinary analysis using "modern" software development practices and provide tested, validated, and well documented software for continued development and support. This was continued through the last few decades of the 20th century before an U.S. Army DEVCOM, Aviation & Missile Center (AvMC) Phase II SBIR contract was awarded to Advanced Rotorcraft Technology, Inc (ART) as well as a DEVCOM, AvMC/ART CRADA which was established in August of 2000 providing that ART would support and maintain the Army code and an official release of RCAS – the successor to 2GCHAS, in 2002.

The Rotorcraft Comprehensive Analysis System, RCAS, is a comprehensive, multi-disciplinary system which allows for the analysis of arbitrary physical models, vehicle configurations, and rotor types. Fully coupled rotor-body, multi-rotor – multi-body analyses are possible using finite elements and multi-body dynamics. The program allows for modular configurations ranging from a single, isolated rotor with single blade analysis to a fully elastic, geometrically exact non-linear beam fuselage coupled with multiple rotors. RCAS is considered a computational structural dynamics (CSD) code which can be coupled with computational fluid dynamics (CFD) codes as well as aero-acoustic analysis codes to provide high-fidelity analysis in many key interest areas within the rotorcraft industry.

1.2.2 C-81/COPTER

This Bell Helicopter/Textron product was one of the first aeroelastic rotor models. It permits the user to model a rotor with any number of blades; each blade has a general planform that includes twist, sweep and anhedral through large angles [4]. A finite number of blade segments comprise each blade. Blade section pitch motion is prescribed by the pilot through the collective and cyclic inputs. Blade section reaction motion is calculated by combining prescribed motion with modal motion calculated using the normal modes method. The section aerodynamics use tables for the lift, drag and moment coefficients. The coefficients are functions of angle of attack and Mach number, but may also use simpler and faster linear aerodynamics or analytic functions. The wake models include an empirically modified momentum-theory, or a vortex based free-wake model. The normal modes are calculated using a fully coupled Myklestad/Holzer method presenting the modes in five degrees of freedom, the sixth degree being along the span of the blade which is constrained to be inextensible. Other features include the hub transfer matrix which combines the root boundary conditions of each blade with all others so that all possible hub configurations such as teetering, gimballed, hinge-less cantilevered, etc. are correctly modeled.

1.2.3 REXOR

This analysis tool is a Lockheed product developed in the 1970s. It is also an aeroelastic rotor model, but it limits the modes to the first in-plane bending, the first flapwise bending and the second flapwise bending using a lumped parameter approach. Most of the derivations of the rotor and airframe motions and loads use formal large angle transformations and linear algebra techniques. This method reduces the number of hand-expansions (and attendant typographical errors) but at the expense of making in-depth analysis more difficult. In fact, the documentation warns that REXOR, an acronym that means Revised and EXtended rotOR model should not be used as a parametric design analysis tool because of the high computer time and cost associated with an execution [5].

1.2.4 MOSTAB

The MOdular STABility derivative program was a product of Paragon Pacific, Inc., also in the 1970s. Its most important feature was the rotor model, which also used the normal modes method to calculate blade motion and loads. A Myklestad program calculated the normal modes and the bending moment coefficients off-line. The MOSTAB program accommodated any number of modes, though the usual selection included the first and second flapwise modes, the first in-plane mode and the first torsional mode. As the frequencies of the modes sometimes exceeded two per rev, MOSTAB employed several numerical techniques to accommodate the higher frequencies. MOSTAB employed the Newton-Raphson method for the overall rotor/fuselage trim problem and the periodic shooting technique to accelerate the trim problem for the rotor. This technique reduced the time to trim when lightly damped modes such as the in-plane and torsional modes were part of an analysis. MOSTAB also generated a detailed time-varying linear model of the rotor with a post-processor called ROLIM [6].

1.2.5 Summary

Many rotor-analysis programs, of which, MOSTAB, REXOR, and COPTER listed above are a small subset, use algebraic methods to close the thrust/induced velocity loop, which is an implicit loop that links the rotor forces and moments to the axial and rotational induced velocity. This is a limitation because the blade motion must be solved for separately from the rotor forces and moments. This increases computational time and decreases the computational efficiency of the program. With the advent of the dynamic inflow models of Pitt-Peters, and Peters et.al., when combined with the periodic shooting technique, the blade motion trim time and closure of the thrust/induced velocity has been accelerated in the current analysis program.

CHAPTER 2. DYNAMICS

Figure 2-1, below, presents a simplified, articulated rotor hub with discrete hinges for flapping and lead-lag motion [1]. The flapping hinge for a rotor hub may also be offset a distance from the axis of rotation.

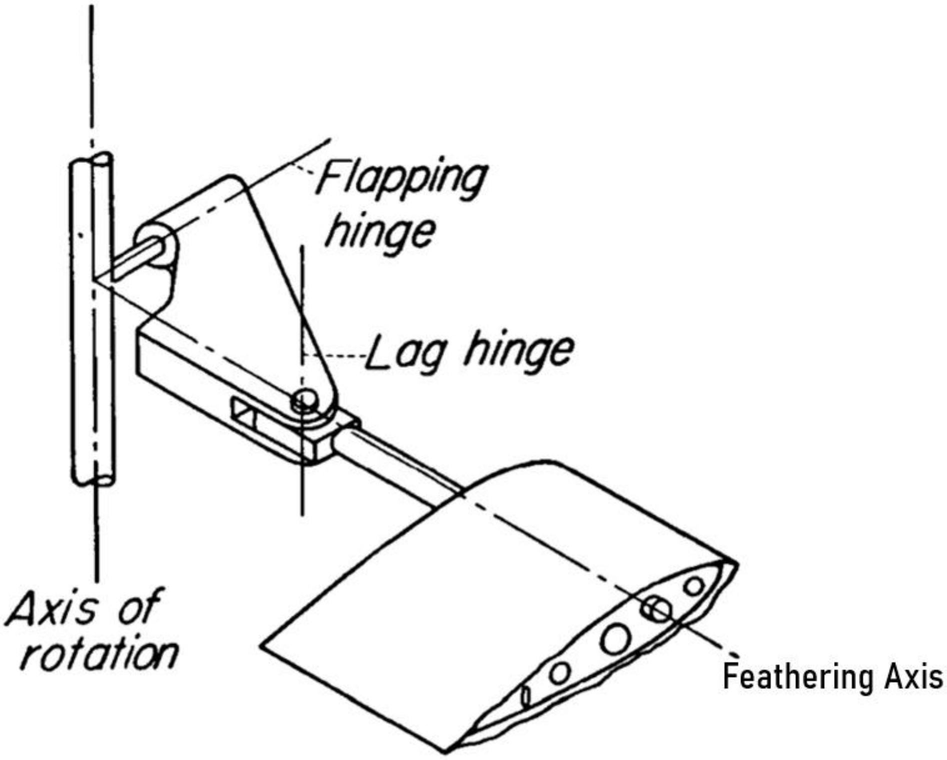


Figure 2-1 Simplified Drawing of an Articulated Rotor Hub.

2.1 Development of the Flapping Equation

2.1.1 Blade Kinematics

Flapping motion will be examined in this section including the additional hub restraints of δ_3 (described later) and hinge offset. The following section works with the assumptions listed below.

Assumptions:

1) The blade is presumed to be infinitely stiff.

2) *The blade is allowed to have discrete hinges for both the flapping and lead-lag degrees of freedom.*
3) *Each hinge is offset an arbitrary distance from the center of rotation (COR).*
4) *The origin of the blade reference line (BRL) is the center of rotation; however, the first station may be offset to account for a torque offset.*
5) *The flapping hinge may have a δ_3 hub restraint.*
6) *The prescribed or manufactured shape of the blade may include a schedule for twist, sweep, anhedral, and (or) precone.*

Torque offset is used to offload or "relieve" in-plane moments developed from chord shears as the blade is whirled around the azimuth and drag is formed. Delta-3, δ_3, is a physical or virtual hub restraint which couples pitch with flap motion of the blade. This is either from the physical cant of the flapping hinge or due to the geometrical attributes of the relationship between the flapping hinge and the pitch horn or the use of electronic δ_3.

The traditional method of determining the position vector, r_p, of a blade particle and its time derivatives, $\dot{r_p}$ and $\ddot{r_p}$ for the rate and acceleration, is demonstrated with the following expressions and figure 2-2. In figure 2-2 below, the hinge is offset a distance, e, the hinge has a physical cant by a value of δ_3, and the blade element is a distance, s, from the center of rotation flapped up by a value of β.

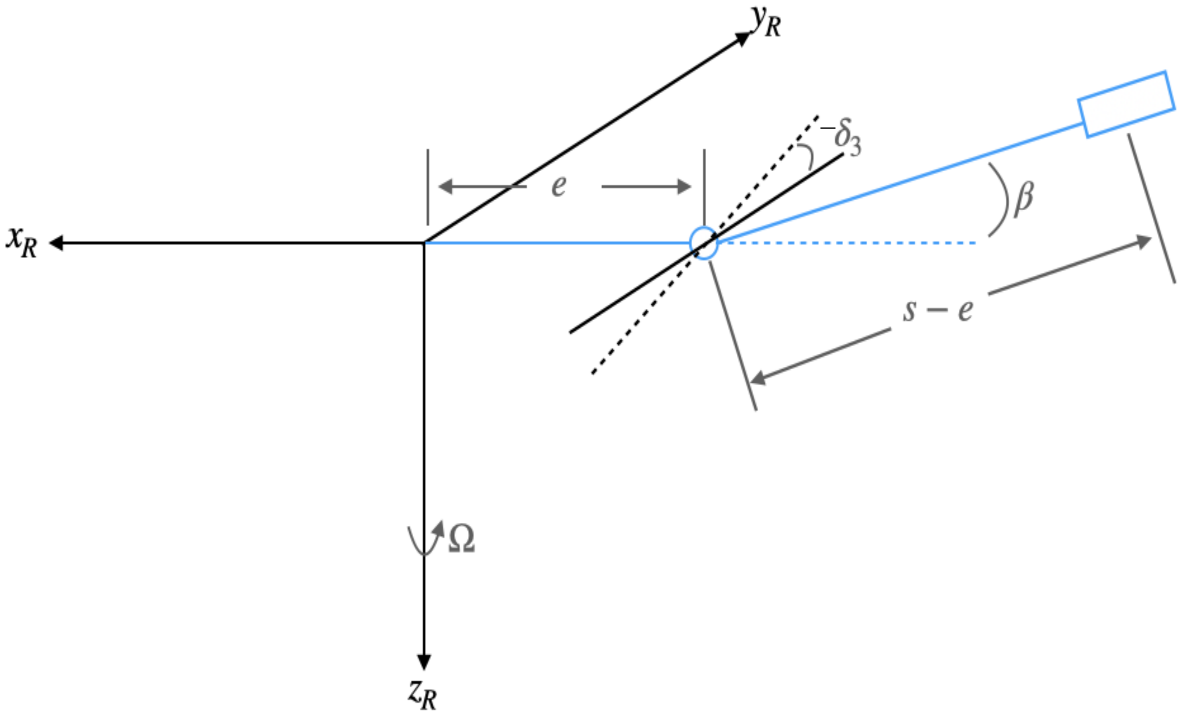


Figure 2-2 Flapped blade model with negative delta-3 and hinge offset in rotating frame.

2.1.1.1 Traditional Kinematics

If the blade station is inboard of the flapping hinge, the position, rate, and acceleration are as follows.

$$r = \begin{bmatrix} x_p \\ y_p \\ z_p \end{bmatrix} = \begin{bmatrix} -s \\ 0 \\ 0 \end{bmatrix}, \dot{r} = \begin{bmatrix} \dot{x}_p \\ \dot{y}_p \\ \dot{z}_p \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}, \ddot{r} = \begin{bmatrix} \ddot{x}_p \\ \ddot{y}_p \\ \ddot{z}_p \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$$

If the blade particle is outboard of the flapping hinge, the position, rate, and acceleration are determined through this sequence of transformations.

1) The first step is to translate along the blade to set the origin at the flapping hinge.
2) The second step is to rotate the coordinate system such that the δ_3 angle is accounted for.
3) The third step is to transform the coordinate system to be aligned with the flapped blade.

4) Next, the coordinate system is transformed back through the flapped axes into the rotating system aligned with the cant from δ_3.

5) Then, the coordinate system is transformed back through the cant from δ_3 and translated such that the origin is back at the center of rotation.

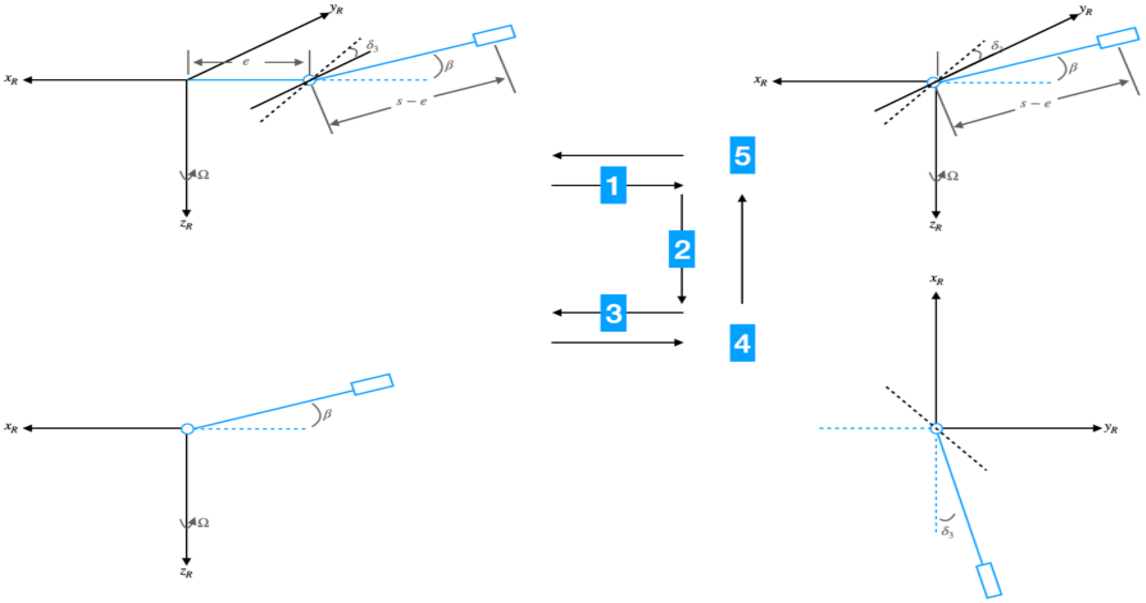


Figure 2-3 Pictorial order of operation for blade kinematics.

$$r = \begin{bmatrix} x_p \\ y_p \\ z_p \end{bmatrix} = \begin{bmatrix} -e - (s-e)\cos(\beta) \\ 0 \\ -(s-e)\sin(\beta) \end{bmatrix}$$

$$\dot{r} = \begin{bmatrix} \dot{x}_p \\ \dot{y}_p \\ \dot{z}_p \end{bmatrix} = \begin{bmatrix} (s-e)\sin(\beta)\,\dot{\beta} \\ 0 \\ -(s-e)\cos(\beta)\,\dot{\beta} \end{bmatrix}$$

$$\ddot{r} = \begin{bmatrix} \ddot{x}_p \\ \ddot{y}_p \\ \ddot{z}_p \end{bmatrix} = \begin{bmatrix} (s-e)\sin(\beta)\,\ddot{\beta} + (s-e)\cos(\beta)\,\dot{\beta}^2 \\ 0 \\ -(s-e)\cos(\beta)\,\ddot{\beta} + (s-e)\sin(\beta)\,\dot{\beta}^2 \end{bmatrix}$$

For the traditional methodology to work properly, bookkeeping must be precise and account for the signs of each component of positioning - which as demonstrated can become quite lengthy.

2.1.1.2 Matrix Transformation Method

The blade kinematics are established through rotation matrices that account for the prescribed angular motion, the flapping rate and acceleration, the lead-lag rate and acceleration. Additional blade configurations such as torque offset or δ_3 are accounted for, too. Figure 2-3 shows a pictorial representation of the process that is required to ensure the motion of the blade is captured. To begin, the kinematics portion of the program establishes the blade reference line vector and compares the location the blade elements to the location specified for the flapping hinge and the lead-lag hinge if they are present and if that degree of freedom is enabled. These vectors are named “efb” and “elb” representing the binary vector of either 0’s or 1’s- a zero if the position is at or before the hinge and a one if the blade element is after the hinge. In general, if the infinitely stiff blade assumption is in place, the transformation matrix only needs to be calculated once for every radial station beyond the flapping or lead-lag hinge. The blade particle rates and accelerations are also required for each station beyond the flapping or lead-lag hinge. After inspection of the terms in the transformation matrix, the only term that changes with respect to time is the flapping term producing a flapping rate and acceleration. The transformation matrices are constructed in a similar fashion remembering that the chain rule must be applied to the rotation matrix for the flapping rate and flapping acceleration as well as the lead-lag rate and acceleration as shown below. Also, keep in mind the positively flapped blade is about the negative y-axis direction and therefore a “β use” variable is specified. These calculations are produced within a radial integration loop at each time step within a call to the rotor model.

$$\beta_{use} = -\beta$$

The rotational matrices for flapping are:

$$R_y(\beta) = \begin{bmatrix} \cos(\beta) & 0 & -\sin(\beta) \\ 0 & 1 & 0 \\ \sin(\beta) & 0 & \cos(\beta) \end{bmatrix}$$

$$R_y{}'(\beta) = \begin{bmatrix} -\sin(\beta) & 0 & -\cos(\beta) \\ 0 & 0 & 0 \\ \cos(\beta) & 0 & -\sin(\beta) \end{bmatrix}$$

$$R_y{}''(\beta) = \begin{bmatrix} -\cos(\beta) & 0 & \sin(\beta) \\ 0 & 0 & 0 \\ -\sin(\beta) & 0 & -\cos(\beta) \end{bmatrix}$$

The rotational matrices for in-plane lead-lag motion are:

$$R_z(\gamma) = \begin{bmatrix} \cos(\gamma) & \sin(\gamma) & 0 \\ -\sin(\gamma) & \cos(\gamma) & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$R_z{}'(\gamma) = \begin{bmatrix} -\sin(\gamma) & \cos(\gamma) & 0 \\ -\cos(\gamma) & -\sin(\gamma) & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

$$R_z{}''(\gamma) = \begin{bmatrix} -\cos(\gamma) & -\sin(\gamma) & 0 \\ \sin(\gamma) & -\cos(\gamma) & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

Next, the transformation matrices are put together using the chain rule as described previously.

$$R_b = R_y(\beta)$$

$$\dot{R_b} = R_y{}'(\beta) * \dot{\beta}$$

$$\ddot{R_b} = R_y{}''(\beta) * \dot{\beta}^2 + R_y{}'(\beta) * \ddot{\beta}$$

The transformation matrices for the lead-lag motion are:

$$R_g = R_z(\gamma)$$

$$\dot{R_g} = R_z{}'(\gamma) * \dot{\gamma}$$

$$\ddot{R_g} = R_z{}''(\gamma) * \dot{\gamma}^2 + R_z{}'(\gamma) * \ddot{\gamma}$$

Following the construction of the transformation matrices, the position vectors of the blade elements with respect to the discrete hinges are built.

$$r_{flap} = r_{flap} + R_g^T * R_b^T \begin{bmatrix} -d_{sf} \\ 0 \\ 0 \end{bmatrix}$$

$$r_{lag} = r_{lag} + R_g^T * \begin{bmatrix} -d_{sl} \\ 0 \\ 0 \end{bmatrix}$$

Where $d_{sf}$and $d_{sl}$are the distances travelled from one radial station to the next, starting from the respective flap or lag hinge.

Then the blade position vector is built using the updated flapping and lead-lag angle inputs.

$$r_s = r_s + R_g^T * R_b^T * \begin{bmatrix} -d_s \\ 0 \\ 0 \end{bmatrix}$$

$$\dot{r}_s = \dot{r}_s + \left(\dot{R}_g^T * R_b^T + R_g^T * \dot{R}_b^T \right) * \begin{bmatrix} -d_s \\ 0 \\ 0 \end{bmatrix}$$

$$\ddot{r}_s = \ddot{r}_s + \left(\ddot{R}_g^T * R_b^T + 2\, \dot{R}_g^T\, \dot{R}_b^T + R_g^T * \ddot{R}_b^T \right) * \begin{bmatrix} -d_s \\ 0 \\ 0 \end{bmatrix}$$

We acknowledge that the expressions above imply radial integration from one radial station to the next.

Then we account for the one-time step lag in loads due to the acceleration terms.

$$\ddot{r}_b = \ddot{R}_b(irs - 1) + [-\sin(\beta), 0, -\cos(\beta)] * \dot{R}_g^T * d_{sf}$$

$$\ddot{r}_g = \ddot{R}_g(irs - 1) + [-\sin(\gamma), \cos(\gamma), 0] * d_{sl}$$

Along with the translational and rotational displacements along the blade due to lead-lag motion, flapping motion, and motion inboard of any discrete hinges placed along the blade, there is prescribed precone, twist, and sweep that can be manufactured into the blade. These are presented in the following way. For the region outboard of the flapping hinge we have:

$$r_a = \begin{bmatrix} \zeta = \zeta_0 + tan(\delta_3)\beta \\ \tau = \tau_0 - \beta \\ \nu = \nu_0 \end{bmatrix}$$

And for the region inboard of the hinge we have:

$$r_a = \begin{bmatrix} \zeta_0 \\ \tau_0 \\ \nu_0 \end{bmatrix}$$

In the figure below, the sign convention for twist for a counterclockwise rotating rotor is such that the twist is negative closer to the root and becomes more positive with increasing radial distance.

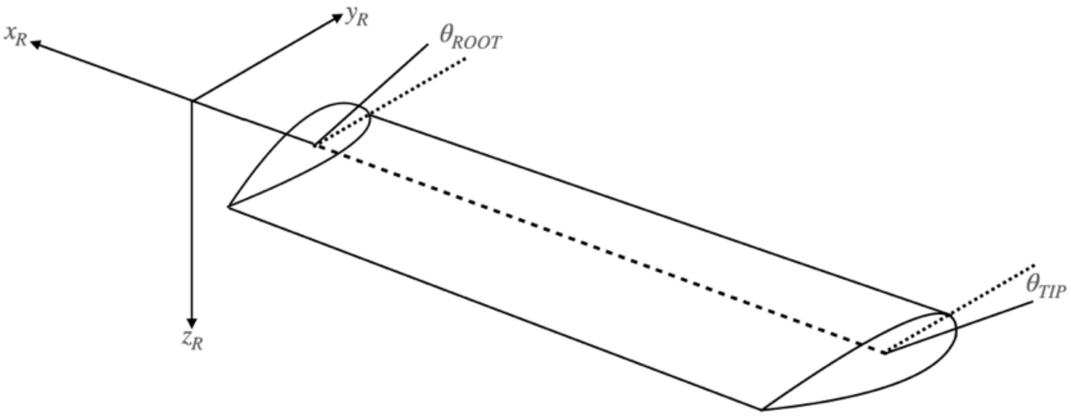


Figure 2-4 Twist distribution for blade structure in rotating system.

2.1.2 Blade Dynamics

This section presents the development of the equations of motion for a dynamic rotor model. Even in a vacuum, a rotating blade will develop both forces and moments in the x-, y-, and z- axes due primarily to the centrifugal field. The development of these loads is presented through the derivation of the acceleration equation and then applied to the distributed forces and moments along with a brief example with several assumptions.

First, the derivation of the acceleration equation is given with the recollection that the rotor hub may translate forming a rectilinear velocity vector in addition to the rotational velocity as shown in the equations below. Referring to figure 2-5, the velocity vector is:

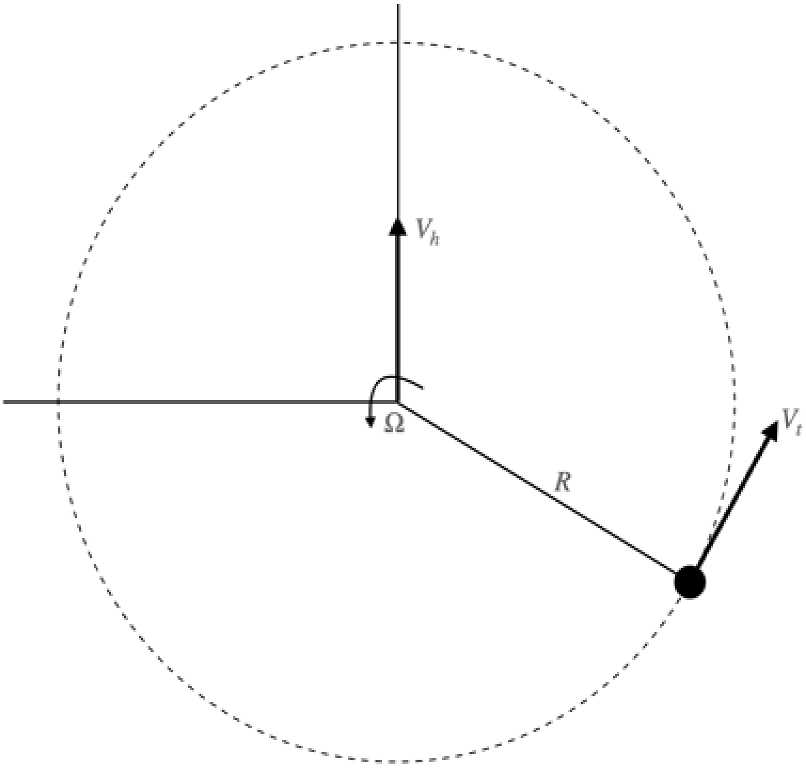


Figure 2-5 Velocity at the hub and at the tip seen from above.

$$v = \dot{r} + \omega \times r$$

It should be noted.

$$\dot{r} = \frac{\partial r}{\partial t}$$

Here, $r = \begin{Bmatrix} x \\ y \\ z \end{Bmatrix}$ and $\omega = \begin{Bmatrix} 0 \\ 0 \\ -\Omega \end{Bmatrix}$, though we note that the angular rate vector in general will also include rotational rates around all three axes. Then the acceleration is the time derivative of the velocity vector.

$$a = \frac{d}{dt}(v) + \omega \times v$$

The time derivative of velocity is given as:

$$\frac{d}{dt}(v) = \ddot{r} + \dot{\omega} \times r + \omega \times \dot{r}$$

Then the rotational contribution becomes:

$$\omega \times v = \omega \times (\dot{r} + \omega \times r)$$

Finally, the acceleration equation is given in the following form – the Coriolis acceleration equation.

$$a = \ddot{r} + \dot{\omega} \times r + 2(\omega \times \dot{r}) + \omega \times (\omega \times r)$$

ω and $\dot{\omega}$ are given below with the subscript b denoting the body reference system.

$$\omega = \begin{bmatrix} p_b cos(\Omega t) - q_b sin(\Omega t) \\ p_b sin(\Omega t) + q_b cos(\Omega t) \\ r_b - \Omega \end{bmatrix}$$

$$\dot{\omega} = \begin{bmatrix} \dot{p}_b cos(\Omega t) - p_b sin(\Omega t)\Omega - \dot{q}_b sin(\Omega t) - q_b cos(\Omega t)\Omega \\ \dot{p}_b sin(\Omega t) + p_b cos(\Omega t)\Omega + \dot{q}_b cos(\Omega t) - q_b sin(\Omega t)\Omega \\ \dot{r}_b - \dot{\Omega} \end{bmatrix}$$

In the blade kinematics section, the development of the position vector and its time derivatives is presented and as such is not developed in this section again; however, recall that r is the position vector of the blade particle and ω is the rate at which the system rotates with respect to the fixed system.

To develop the loads experienced by the blade particle, first the distributed loads are calculated at each blade particle radial station and then numerically integrated to collect the loads at the hub before transforming into the fixed system.

$$dF_i = -\overline{m}a$$

Then the distributed moments are calculated as the cross product of the blade particle position and distributed forces at the blade particle.

$$dM_i = r \times dF_i$$

The distributed forces and moments, dF_i and dM_i, include the forces and moments respectively in each of the x-, y-, and z-axes.

$$\begin{bmatrix} dF_i \\ dM_i \end{bmatrix} = \begin{bmatrix} dFx_i \\ dFy_i \\ dFz_i \\ dMx_i \\ dMy_i \\ dMz_i \end{bmatrix}$$

The transformations required to get the distributed loads back into the fixed system to apply at the rotor hub for this model or to the center of gravity for the full aircraft model are dependent upon the hub restraints, geometry, and motion degrees of freedom for the blade.

2.1.3 Distributed Inertial Loads

The distributed inertial loads accumulate to produce the forces and moments at the hub in the rotating reference axes once an integration is performed from the root to the tip of the blade. The inertial contribution to the flapping equation is demonstrated for example throughout this section, however, the same process is followed to produce the other five degrees of freedom.

Recall the acceleration equation. presented now with the addition of the translational acceleration of the hub resolved to the rotating reference system.

$$a = \ddot{r} + \dot{\omega}_r \times r + 2(\omega_r \times \dot{r}) + \omega_r \times (\omega_r \times r) + T_\psi a_{hub}$$

The equation above has the following definitions:

r is the position vector at the blade particle from the center of rotation (COR).

$\dot{r}$ is the velocity of the blade particle due to in-plane and out-of-plane motion.

$\ddot{r}$ is the acceleration of the blade particle due to in-plane and out-of-plane motion.

ω_r is the angular velocity of the rotor hub accounting for the rotor spin rate resolved to the rotating system.

$\dot{\omega}$ is the angular acceleration of the hub in the rotating system.

a_{hub} is the translational acceleration of the hub (in the hub reference frame).

T_ψ is the transformation matrix to move from the fixed system to the rotating system; it is defined below using the traditional rotation matrix around the z-axis, and accounts for the positive sense of azimuth angle being a negative rotation around the z-axis.

$$T_\psi = \begin{bmatrix} cos(\psi) & -sin(\psi) & 0 \\ sin(\psi) & cos(\psi) & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

The azimuth angle, ψ, is defined below as the product of rotor speed and time.

$$\psi = \int_0^t \Omega dt = \Omega t$$

In a moment, we will use the time derivative of the T_ψ matrix. It is defined below:

$$\dot{T}_\psi = \Omega \begin{bmatrix} -\sin\psi & -\cos\psi & 0 \\ \cos\psi & -\sin\psi & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

In the figure below, the blade reference line (BRL) is demonstrated where r is a function of the arc length along the BRL.

$$r = \begin{bmatrix} x(s) \\ y(s) \\ z(s) \end{bmatrix}$$

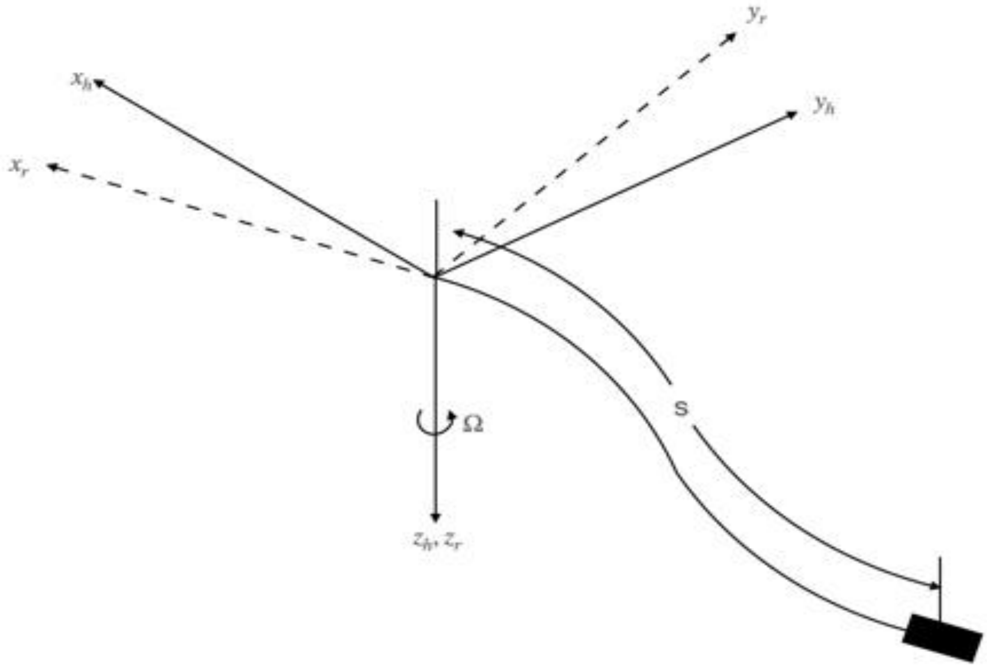


Figure 2-6 Blade reference line development.

This model assumes an infinitely stiff blade except for the discrete hinges for flapping motion and lead-lag motion. The displacements shown in Figure 2-7, exaggerated for clarity, present the path of the BRL in the rotating system.

Next, with the addition of a discrete flapping hinge, the BRL is shown as a rigid blade model as to not confuse the reader with a fully aero-elastic blade formation. This formulation still assumes an infinitely rigid blade.

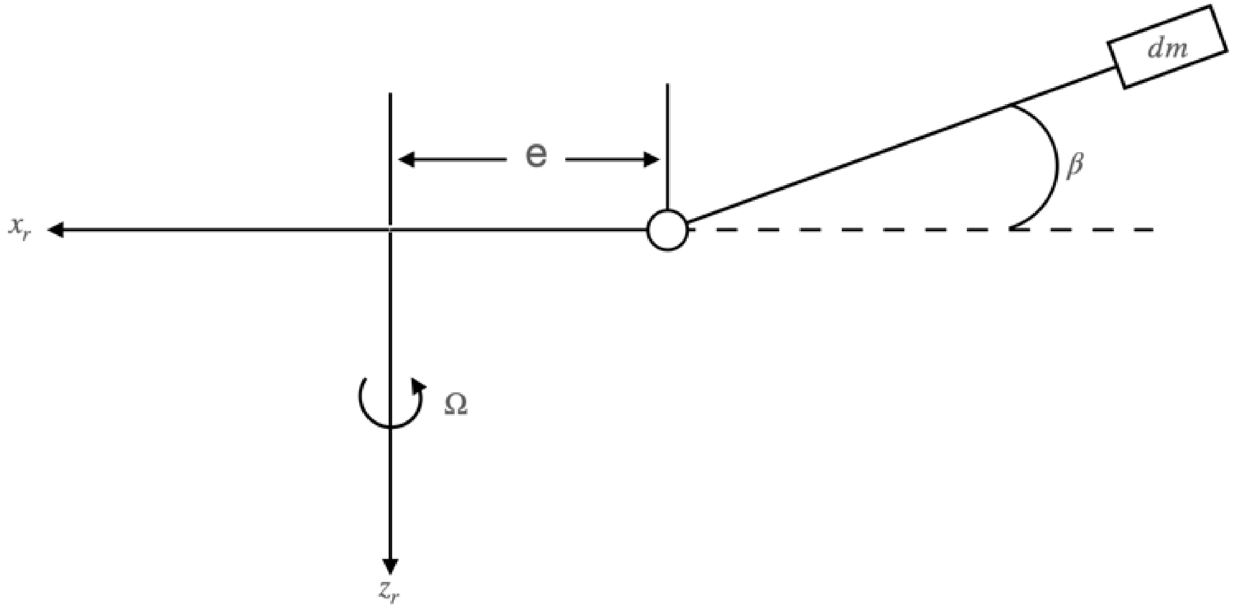


Figure 2-7 Flapped blade sign convention for hinge offset and flapped blade.

As shown in Figure 2-7, there is an inboard region and an outboard region of the blade with respect to the flapping hinge that is offset a distance of e. For the inboard region of the blade, the subscript i is used and for the outboard region the subscript o is used.

$$r_{pi} = \begin{bmatrix} -s \\ 0 \\ 0 \end{bmatrix} \qquad r_{po} = \begin{bmatrix} -e - (s-e)cos(\beta) \\ 0 \\ -(s-e)sin(\beta) \end{bmatrix}$$

$$\dot{r}_{pi} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \qquad \dot{r}_{po} = \begin{bmatrix} (s-e)sin(\beta)\dot{\beta} \\ 0 \\ -(s-e)cos(\beta)\dot{\beta} \end{bmatrix}$$

$$\ddot{r}_{pi} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} \qquad \ddot{\mathrm{r}}_{\mathrm{po}} = \begin{bmatrix} (s-e)sin(\beta)\ddot{\beta} + (s-e)cos(\beta)\dot{\beta}^2 \\ 0 \\ -(s-e)cos(\beta)\ddot{\beta} + (s-e)sin(\beta)\dot{\beta}^2 \end{bmatrix}$$

The angular rates in the hub (fixed) frame of reference come from the body roll, pitch, and yaw rates.

$$\omega_h = \begin{bmatrix} p \\ q \\ r \end{bmatrix}_b$$

Then to resolve the hub rates to the rotating system, the vector is pre-multiplied by the transformation matrix around the z-axis through the azimuth angle and the spin rate of the rotor is accounted for through the addition of Ω.

$$\omega_r = T_\psi \begin{bmatrix} p \\ q \\ r - \Omega \end{bmatrix}$$

And from this the time derivative, or angular acceleration is given as such.

$$\dot{\omega}_r = \dot{T}_\psi \begin{bmatrix} p \\ q \\ r - \Omega \end{bmatrix} + T_\psi \begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} - \dot{\Omega} \end{bmatrix}$$

Next, with the definition of all components of the acceleration equation, the acceleration is calculated at each radial station along the blade. This is then used to calculate the distributed forces followed by the distributed moments.

$$df_i = -\overline{m}a$$

$$dm_i = r_{po} \times df_i$$

The moment at the hinge is then calculated through the numerical integration of the distributed moments from the hinge out to the tip of the blade.

$$M_h = \int_e^R dm_i ds$$

To look specifically at the flapping equation of motion at the hinge, the moment in the y-axis formulation is set to zero - since a hinge cannot support a moment.

$$M_{hinge} = 0 = \int_e^R dm_i ds = I_B\ddot{\beta} + I_B\Omega^2\beta + I_e\Omega^2\beta$$

After algebraic reorganization of the above equation, the equation of motion for flapping becomes:

$$I_B\ddot{\beta} + (I_B + I_e)\Omega^2\beta = 0$$

I_B is the blade flapping inertia and I_e is the flapping inertia that accounts for spin stiffening due to hinge offset. The flapping and lead-lag acceleration, if enabled, lag the constant states through numerical integration by one frame. Thus, the forces and moments must be corrected by multiplying influence coefficients by the second time derivative with respect to the flapping or lead-lag acceleration. For instance, the force and moment correction factors are shown for the flapping acceleration.

$$\frac{dF}{d\ddot{\beta}} = \begin{bmatrix} \frac{1}{2}m(R-e)^2 sin(\beta) \\ 0 \\ -\frac{1}{2}m(R-e)^2 cos(\beta) \end{bmatrix}$$

$$\frac{dM}{d\ddot{\beta}} = \begin{bmatrix} 0 \\ I_B + I_e cos(\beta) \\ 0 \end{bmatrix}$$

Finally, the development of the flapping equation comes with evaluating the hinge moment with no flapping acceleration and dividing by the negative of the flapping Inertia.

$$\ddot{\beta} = \frac{M_{hinge|\ddot{\beta}=0}}{I_B}$$

The addition of the lead-lag degree of freedom only lengthens the process but is straightforward like the process described above where the position vector is shifted to the lag hinge location rather than the flapping hinge and the weighted integration coefficient factors are taken with respect to the lead-lag acceleration. The weighted integration coefficient factors are formed such that the dot product of these factors and the distributed loads, either aerodynamic or inertial, produces the radially integrated forces or moments as a means of increasing the computational speed of the RPM code.

2.2 Hub Restraints

Hub restraints are mechanical devices that provide the hub with the ability to support a moment. They influence the magnitude and/or the phase of the flapping of a rotor blade. Some hub restraints work to increase the hub moment while others are used to move the flapping frequency as to prevent any destructive resonance issues with the fixed system where damping could be low.

There are several forms of hub restraint used in special cases or in specific rotor hub configurations. Those forms are listed Table 2-1 along with their effects.

Table 2-1 Hub Restraints

Hub Restraint	**Effect**
Flapping Hinge Offset	Raises the flapping frequency.
Flapping Hinge Spring	Raises the flapping frequency.
Delta-3 (δ_3)	Can raise or lower the flapping frequency and adjust the phasing of the flapping (discussed in detail in the aerodynamics section).
Lead-Lag Hinge Offset	Relieves in-plane (chordwise) bending moment
Lead-Lag Hinge Spring and Damper	Used to tune chordwise bending moment and provide in-plane motion stability,
Undersling	Used with precone to mitigate Coriolis loads,
Flex Beam Yoke Grips	Virtual hinge in flapping and lead-lag directions – no mechanical hinge to wear out.

2.2.1 Rotor Configurations

Typical rotor hub designs include articulated, semi-articulated, rigid, teetering, and gimballed rotor configurations [7, 8]. Images of each type are presented below for clarity of the styles of rotor hubs that may be modelled through the RPM code – though some are not currently enabled.

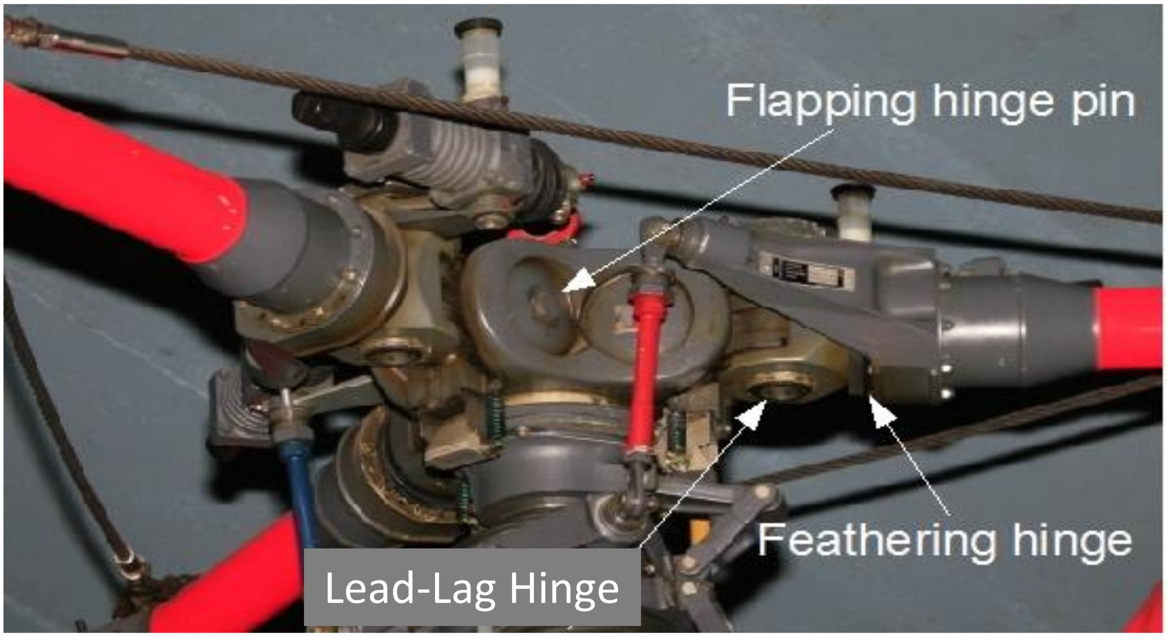


Figure 2-8 Simplified Articulated Rotor Hub.

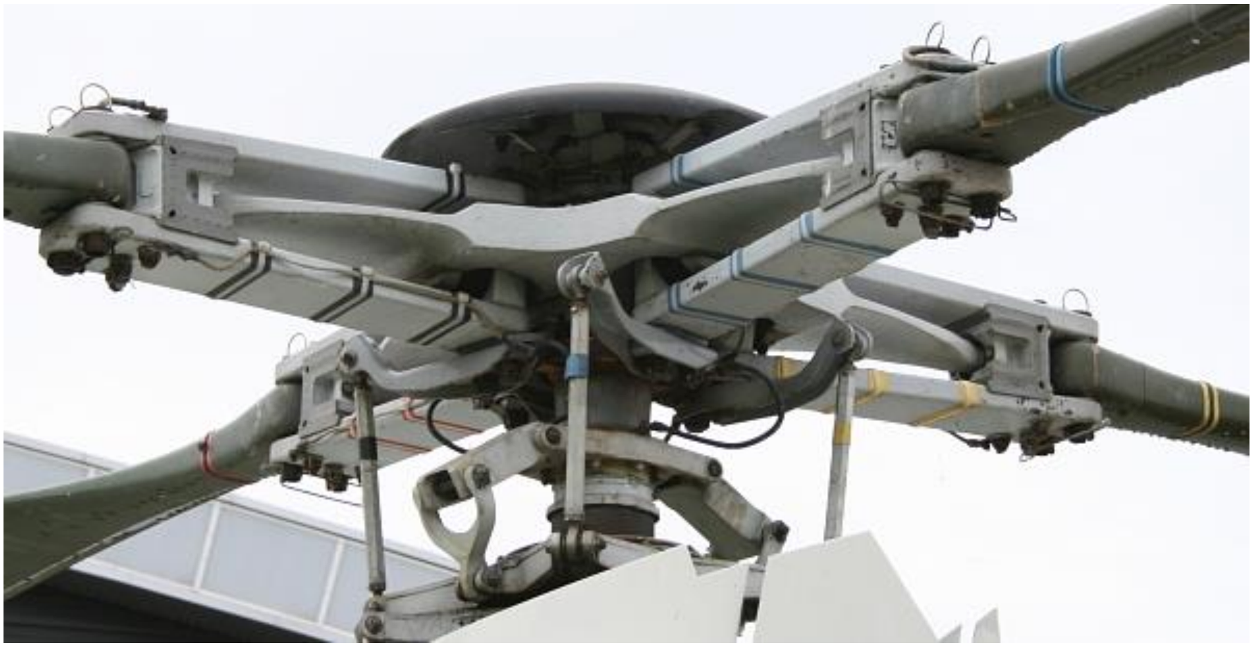


Figure 2-9 Semi-articulated rotor hub.

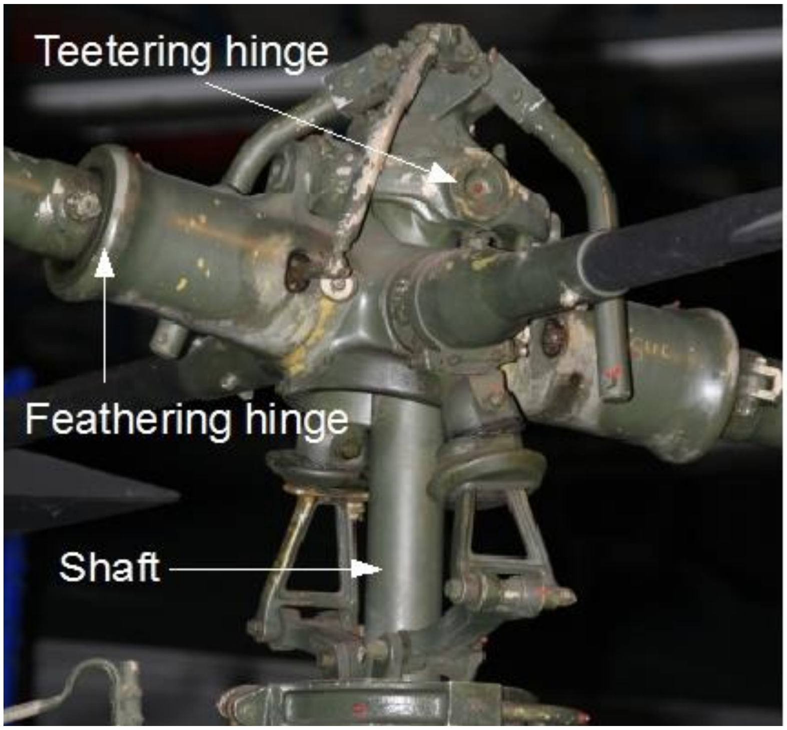


Figure 2-10 Teetering rotor hub.

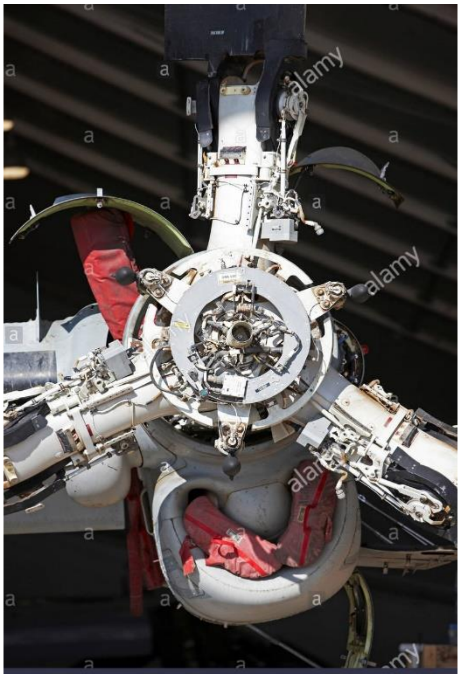


Figure 2-11 Tiltrotor gimballed rotor hub.

2.2.2 Hinge Offset

The hinge offset works to direct the centrifugal force vector and adjust the moment that is passed to the fixed system. The formulation is analyzed as a piece-wise function of the blade particle position vector either being inboard or outboard of the hinge location.

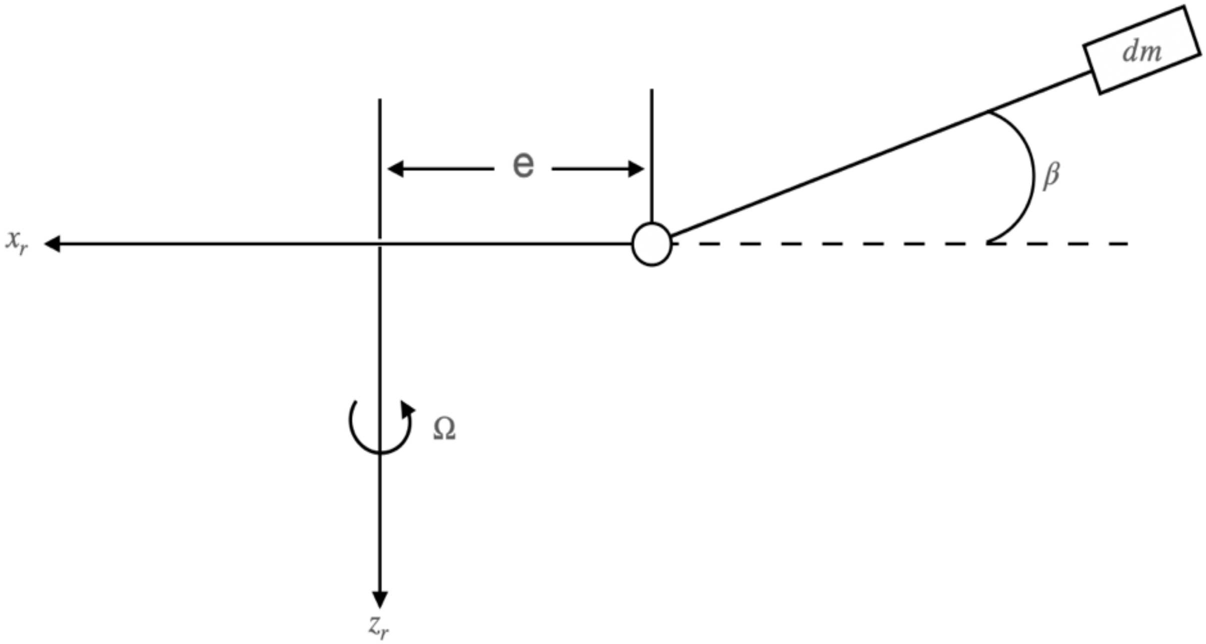


Figure 2-12 Hinge offset, e.

Although not done in the RPM code, for this simple analysis we will make the small angles assumption and state the following.

$$sin(\beta) = \beta$$

$$cos(\beta) = 1$$

$$r_p = \begin{bmatrix} -s \\ 0 \\ -(s-e)\beta \end{bmatrix}$$

$$\dot{r_p} = \begin{bmatrix} 0 \\ 0 \\ -(s-e)\dot{\beta} \end{bmatrix}$$

$$\ddot{r_p} = \begin{bmatrix} 0 \\ 0 \\ -(s-e)\ddot{\beta} \end{bmatrix}$$

The rotational vector is now developed as just the spin vector when setting the body rates to zero - a reasonable assumption for a performance analysis code.

$$\omega = \begin{bmatrix} 0 \\ 0 \\ -\Omega \end{bmatrix}$$

The acceleration then is developed omitting the hub translational acceleration.

$$a = \ddot{r} + 2(\omega \times \dot{r}) + \dot{\omega} \times r + \omega \times (\omega \times r)$$

From this, the piecewise acceleration can be developed for the inboard and outboard region of the rotor with respect to the hinge offset.

$$a = \begin{bmatrix} s\Omega^2 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} s\Omega^2 \\ 0 \\ (s-e)\ddot{\beta} \end{bmatrix}$$

Then the differential forces are the product of the distributed mass and the acceleration.

$$df_i = -\overline{m}a ds$$

From this, the moment around the hinge can be developed from the cross product of the position vector outboard of the hinge and the distributed forces.

$$dM_{HINGE} = \overline{m}(s-e)^2\ddot{\beta}ds + \overline{m}s(s-e)\Omega^2\beta ds$$

After some algebra, substitution for simplification and rearranging, the two mass integrals become the following.

$$I_b = \int_e^R \overline{m}(s-e)^2 ds = \frac{\overline{m}(R-e)^3}{3}$$

$$i_b = \int_e^R \overline{m}s(s-e)ds = \overline{m}\left[\frac{(R-e)^3}{3} + \frac{(R-e)^2}{2}e\right]$$

In this illustrative example, we assume distributed mass. In general, the RPM code presumes that the distributed mass is a tabulated, non-linear function of radial station. The flapping equation can be rewritten as the following expression: the non-dimensional flapping frequency follows immediately afterwards.

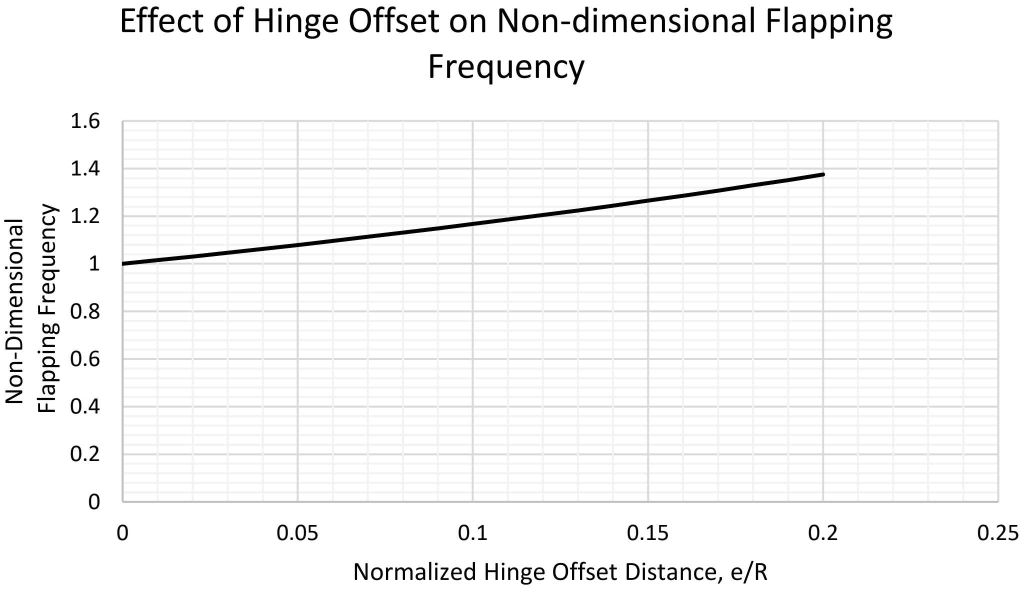


Figure 2-13 Hinge offset effect on non-dimensional flapping frequency.

$$I_b\ddot{\beta} + \Omega^2 i_b \beta = 0$$

Alternatively, the equation above can be rewritten as such below.

$$\ddot{\beta} + \omega_n^2 \beta = 0$$

The non-dimensional flapping frequency, P, is the ratio of the natural frequency and the rotor spin rate.

$$\omega^2 = (P\Omega)^2 = \Omega^2 \left[1 + \frac{3}{2}\frac{e/R}{1 - e/R}\right]$$

The effect of the flapping hinge offset on the non-dimensional flapping frequency can be seen below in figure 2-13.

2.2.3 Hub Spring

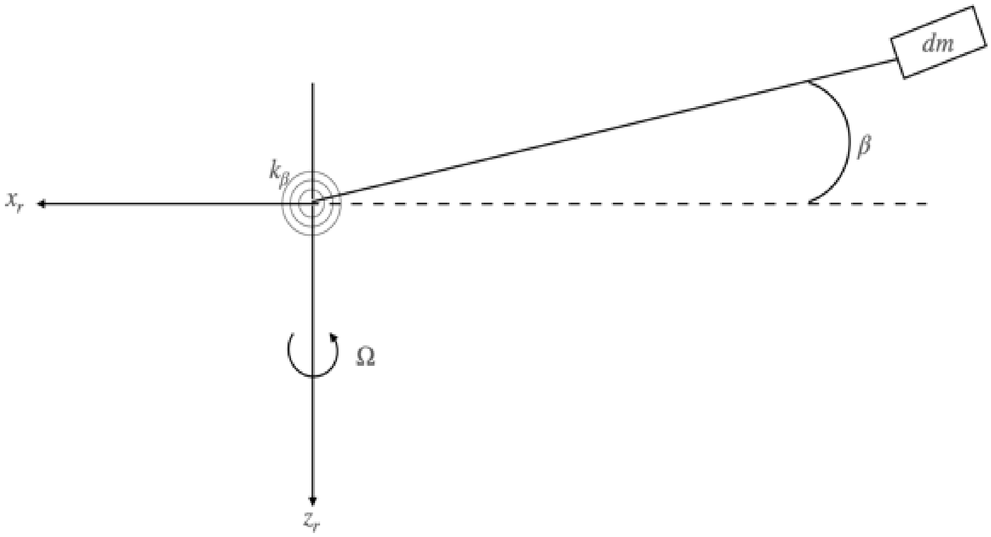


Figure 2-14 Hub spring.

The hub spring acts to resist flapping motion and does so through rotation or translation deformation at the hinge. Hub springs are typically made from metal or an elastomeric material. Figure 2-14 depicts a rigid rotor with a hub spring acting at the center of rotation about the flapping axis. The contribution the flapping spring makes to the flapping frequency is shown in the following expressions.

$$I_b\ddot{\beta} + I_b\Omega^2\beta = -k_\beta\beta + \cdots$$

Alternatively, the equation above can be rewritten as follows.

$$\ddot{\beta} + \left(\Omega^2 + \frac{k_\beta}{I_b}\right)\beta = \cdots$$

The natural flapping frequency is given in the expression below.

$$\omega_n^2 = (P\Omega)^2 = \Omega^2 \left[1 + \frac{k_\beta}{I_b\Omega^2}\right]$$

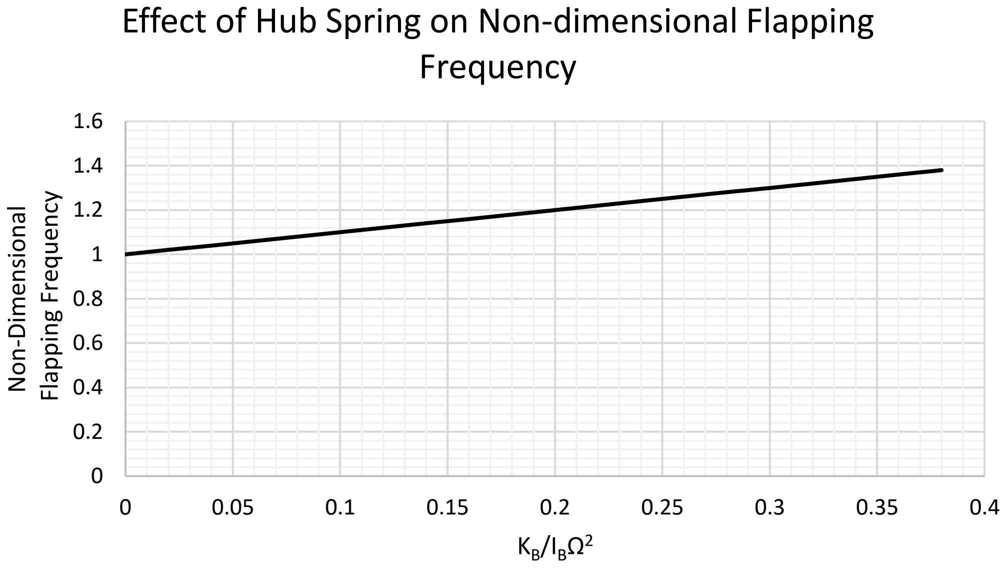


Figure 2-15 Effect of hub spring on non-dimensional flapping frequency.

2.2.4 Summary

In conclusion, hub restraints work to resist flapping motion and can be used to modify the flapping frequency to prevent resonance (often destructive) between the rotor and the fixed system to which it is attached. The hub restraint models can be algebraically added to form the following expression describing the combined or individual effects of a hinge offset and hub spring. Later, in the aerodynamic section delta 3 will be discussed.

$$\frac{\omega_n^2}{\Omega_n^2} = \left[1 + \frac{3}{2}\frac{e/R}{1-e/R} + \frac{k_\beta}{I_b\Omega^2}\right]$$

2.3 Large Angles and Non-linear

2.3.1 Small Angle Approximation Assertion

The RPM analysis utilizes large angle calculations throughout; however, ‘first principle’ analyses used Small Angle Assumptions (SAA) to make certain arithmetic numerically tractable. These earlier analyses argued that the errors introduced by large angles while using SAA were offset by other assumptions. Figure 2-16 supports the legitimacy of SAA and exposes the possible errors of the Small Angle Assumptions. However, as will be seen in the aerodynamic model, large twist angles and reverse flow regions lead to large angles of attack. This leads to unrealistic aerodynamic loads. The RPM model avoids such difficulties by foregoing the SAA.

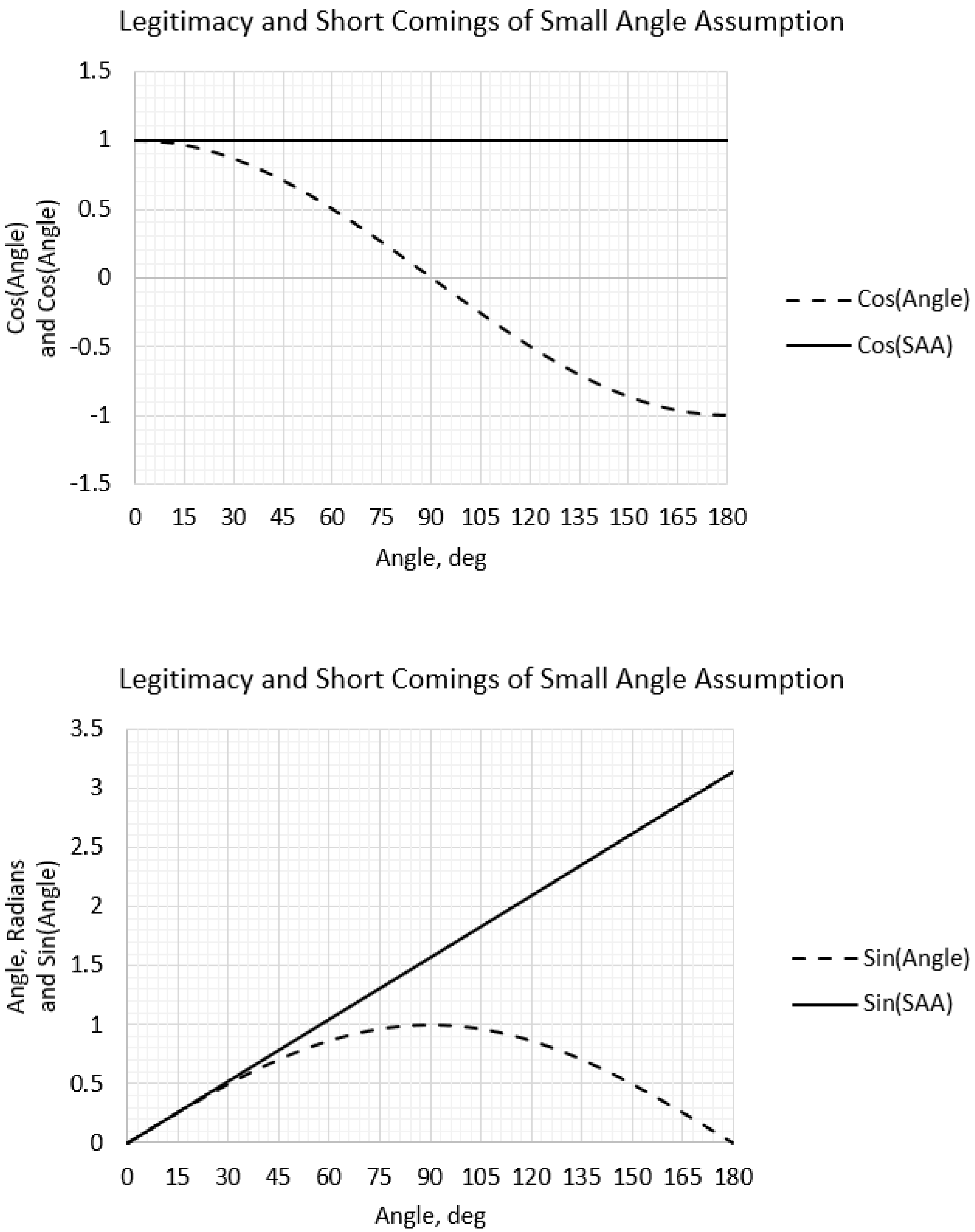


Figure 2-16 Small Angle Assumption.

2.3.2 Comparison to Other Model Methods

Other modeling methods such as modal methods currently work very well in matching frequencies for the defined system but may fail to fully encompass the capability of using large angle analysis within the programs.

CHAPTER 3. AERODYNAMICS

3.1 Aerodynamic Model Basics

Strip theory is used to calculate the aerodynamic loads distributed along the blade. Small angle assumptions are not presumed or used in the development of the distributed loads or the transformations from the flapped-rotating system to the fixed hub reference frame. The blade reference line (BRL) identifies stations along the blade resolved to the rotating system reference frame as shown in Figure 3-1 below.

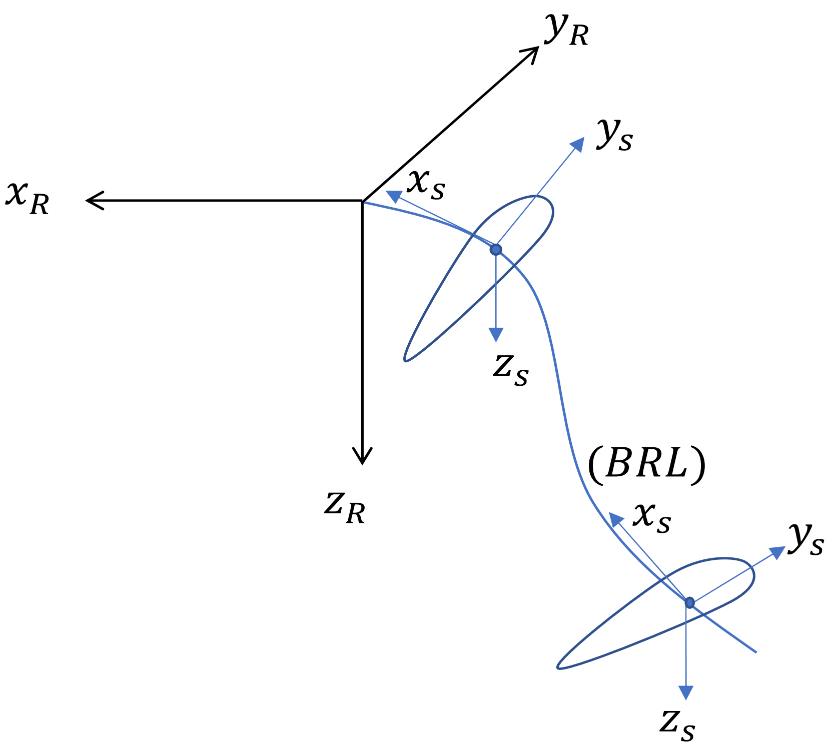


Figure 3-1 Attachment of blade section axes to the Blade Reference Line (BRL).

The origin of the blade section axes is coincident with the quarter chord of the airfoil. The x_s axis is tangent to the BRL, the y_s axis points towards the leading edge of the airfoil section, and the z_s axis is in the direction complying with the right-hand rule. Also, the y_s axis is colinear with the section chord line.

Now in figure 3-2, the section is presented to demonstrate the non-flapped, but cyclic and collective commanded blade section as seen from the tip toward the root.

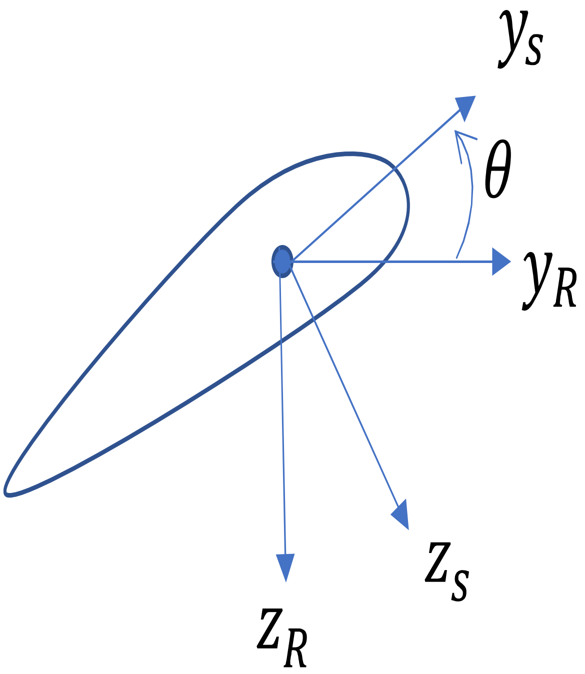


Figure 3-2 Pitched blade section at an arbitrary radial location.

It is important to note that θ grows positively in the counterclockwise direction which is negative about the x-axis since a right-hand rule is used. The equation for the blade pitch angle is given below.

$$\theta = \theta_0 + \theta_T(x) + A_1 cos(\psi) + B_1 sin(\psi)$$

where θ is the blade section pitch angle, θ_0 is the commanded collective input, $\theta_T(x)$ is the section twist, A1 is the commanded lateral cyclic input, and B1 is the commanded longitudinal cyclic input. Generally, the twist is negative, meaning the blade section near the root has a greater leading-edge up angle than the section angle towards the tip of the blade. Figure 3-3 showcases a single blade in the rotating reference frame in which the blade is “flapped up.”

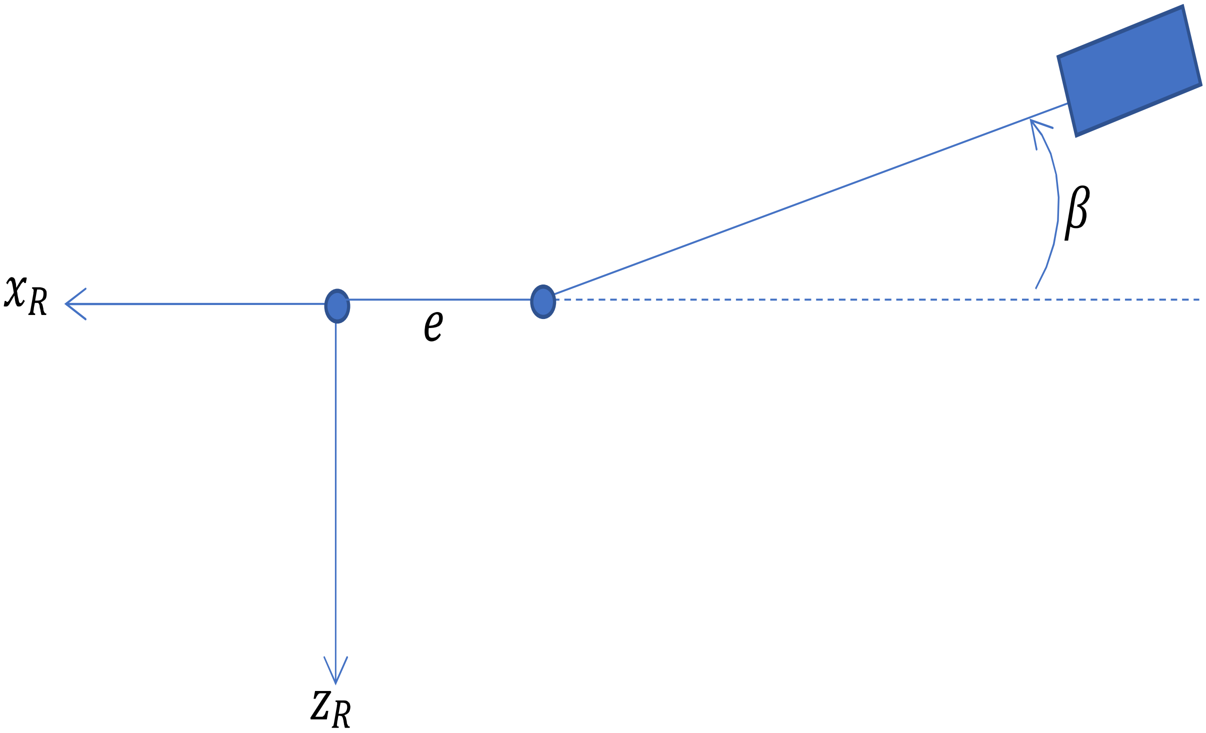


Figure 3-3 A flapped blade element viewed in the rotating axis system.

It is important once again to note that while the blade is physically flapped upwards away from the x-y plane of the fixed system, the blade is flapped negatively about the rotating y-axis through the angle β. Mathematically, a positive β is downward flapping; however, for mental sanity purposes (and generally an easier mental acceptance) a positive β is defined as being upward flapping – we just have to be sure to remember the mathematical signage in the formulation and numerical solution to the flapping equations of motion.

Then looking down on the rotor from above, figure 3-4 shows the lead-lag motion and frame of reference for the blade section. It is important to note the positive angle for lead-lag motion is in the lagging direction – the blade moving opposite the direction of rotation of the rotor. This is simply due to following the right-hand rule for coordinate system. For a brief review thus far, positive flapping is in the negative direction and positive lag motion is positive for the given coordinate systems. It is also important to point out the direction of spin of the rotor is opposite of the positive right-hand rule orientation, as demonstrated in Figure 3-4, presenting the fixed rotor hub frame and the rotating frame of reference in three dimensions.

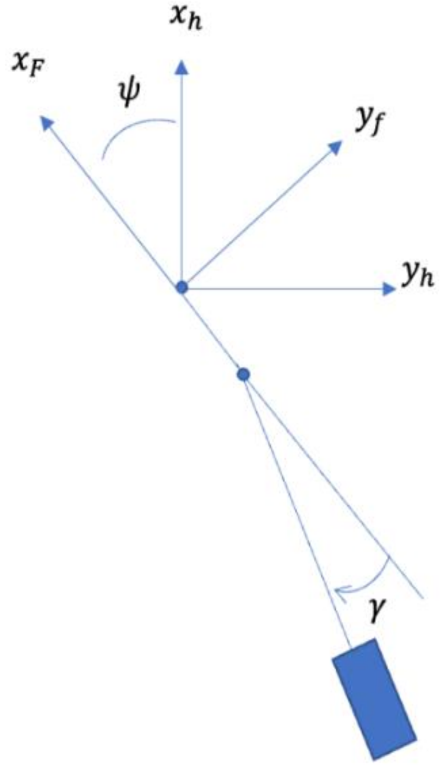


Figure 3-4 A lagging blade element viewed in the rotated and flapped axis system.

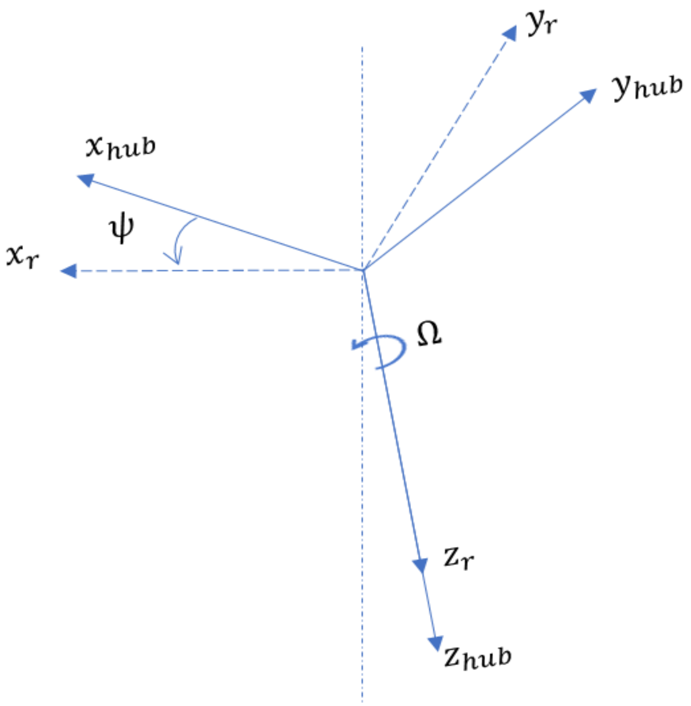


Figure 3-5 Rotating and hub axes have coincident origins and z-axes.

Normally, for most American manufactured rotorcraft, zero azimuth is presented as over the tail boom of the aircraft with 90 degrees over the right wing, 180 over the nose, and so forth. The hub reference frame presented in Figure 3-5 has the positive x direction originating at the center of rotation (center of the mast in shaft normal orientation) directed towards the nose of the aircraft. Following suit, the fixed y-direction is in the direction of the right wing and a positive z-direction completes the right-hand rule. As previously stated, the rotor fixed frame is in the shaft-normal plane in which the z-axis of the rotor fixed frame is colinear with the center of the shaft. This allows for mast tilt and lean for the rotor shaft as depicted below in Figure 3-6 [9].

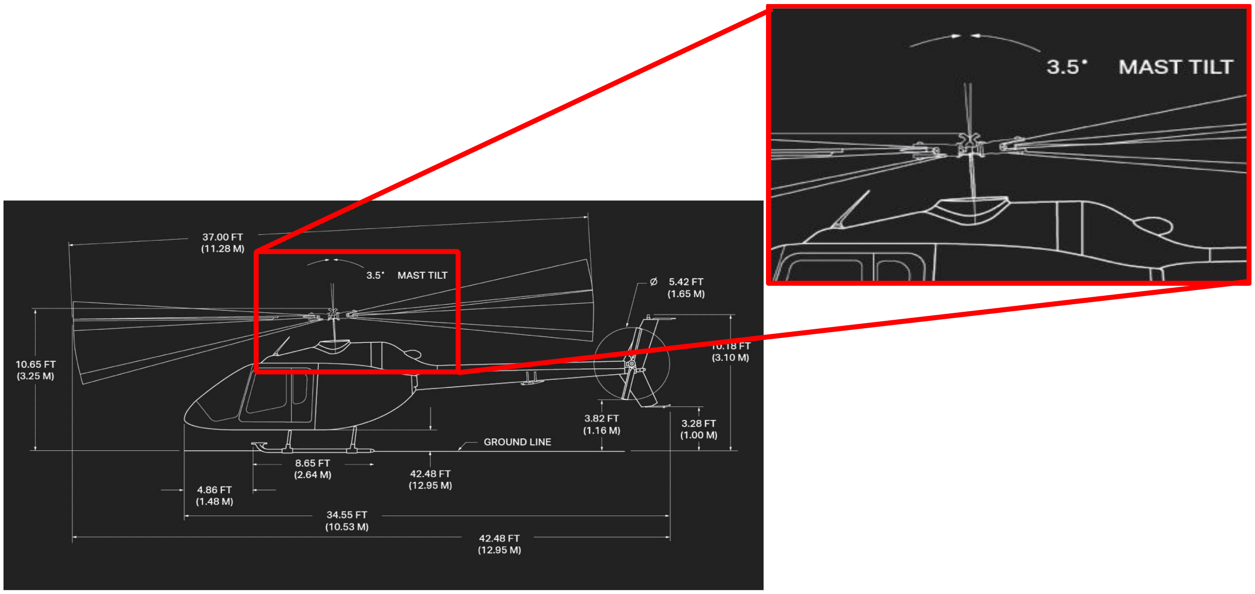


Figure 3-6 Bell 505 displaying Mast Tilt.

The transformation from the fuselage to the hub is described through the following rotations, first through the y-axis and then through the x-axis as below.

$$T_{hb} = R_x(\phi)R_y(\theta)$$

The x- and y- rotation matrices are given below.

$$R_x(\phi_M) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & cos(\phi_M) & sin(\phi_M) \\ 0 & -sin(\phi_M) & cos(\phi_M) \end{bmatrix}$$

$$R_y(\theta_M) = \begin{bmatrix} cos(\theta_M) & 0 & -sin(\theta_M) \\ 0 & 1 & 0 \\ sin(\theta_M) & 0 & cos(\theta_M) \end{bmatrix}$$

The ϕ_M and θ_M angles are the mast lean and tilt respectively. For completeness, the z- rotation matrix is given below which completes the necessary transformation from the fuselage reference frame to the rotating system in the shaft-normal plane.

$$R_z(\psi) = \begin{bmatrix} cos(\psi) & sin(\psi) & 0 \\ -sin(\psi) & cos(\psi) & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

It is important to note that the azimuth angle ψ must use a negative value for counterclockwise spinning rotors. In the next section the development of the aerodynamic velocities, forces, and moments is described in detail building off the basic coordinate systems developed in this section.

3.2 Aerodynamic Model Forces and Moments Development

The aerodynamic velocities, both translational and rotational, are developed by first transforming the earth velocity to the body axis velocity using the T_{be} transformation matrix.

$$T_{be} = R_x(\phi)R_y(\theta)R_z(\psi)$$

Then the velocities are found by pre-multiplying the earth velocities by the T_{BE} transformation matrix.

$$V_b = T_{be}(\phi,\theta,\psi)\dot{P}_e = \ T_{be}V_e$$

The position with respect to Earth and the body velocity are given below.

$$P_e = \begin{Bmatrix} x \\ y \\ z \end{Bmatrix}$$

$$V_b = \begin{Bmatrix} u_b \\ v_b \\ w_b \end{Bmatrix}$$

$$\dot{P}_e = V_e$$

Note the implied definition $\dot{P}_e = V_e$. The Euler angles are:

$$Euler\ Angles = \ \alpha_e = \begin{Bmatrix} \phi \\ \theta \\ \psi \end{Bmatrix}$$

Where ϕ is the roll angle, θ is the pitch angle, and ψ is the yaw angle. The Euler angles orient the fuselage with respect to the earth.

The transformation of Euler rates to body rates is not quite as straightforward. It develops this way:

$$\omega_b = \begin{Bmatrix} \dot{\phi} \\ 0 \\ 0 \end{Bmatrix} + R_x(\phi)\left[\begin{Bmatrix} 0 \\ \dot{\theta} \\ 0 \end{Bmatrix} + R_y(\theta)\begin{Bmatrix} 0 \\ 0 \\ \dot{\psi} \end{Bmatrix}\right] = E(\phi,\theta)\dot{\alpha}_e = \begin{bmatrix} 1 & 0 & -\sin\theta \\ 0 & \cos\phi & \sin\phi\cos\theta \\ 0 & -\sin\phi & \cos\phi\cos\theta \end{bmatrix}\begin{Bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{Bmatrix}$$

It is important to note that the T_{be} matrix is orthonormal; its inverse is its transpose, and it is valid for all angles of any magnitude. The E matrix is not orthonormal, and its inverse is singular at $\theta = \pm\frac{\pi}{2}$, but for this performance analysis code, we can neglect this fact and move forward. The reason it is safe to neglect the singularity is because the aircraft will not be maneuvering or trimming with pitch angles of ± 90 degrees.

To summarize the first step, find velocity and rates at the body axes (center of gravity is the origin) this way:

$$V_b = T_{be}(\phi,\theta,\psi)\dot{P}_e = T_{be}V_e$$

$$\omega_b = E(\phi,\theta)\dot{\alpha}_e$$

Define the rotor spin rate vector as:

$$\omega_z = \begin{Bmatrix} 0 \\ 0 \\ -\Omega \end{Bmatrix}$$

Now, we resolve the velocities to the top of the mast in the shaft-normal axes with the following transformations. It is worth noting that for typical single main rotor configurations it is not uncommon to see a forward mast tilt of up to 5 degrees.

$$T_{mast} = R_x(\varphi_{mast})R_y(\theta_{mast})$$

$$V_{mast:IELA} = T_{mast}V_{mast:IERA}$$

$$\omega_{mast:IELA} = T_{mast}\omega_{mast:IERA}$$

Where IELA refers to the individual element local axis and IERA refers to individual element reference axes respectively. IELA locates an axis system purely through translation from some named base axis system while IERA orients an IELA axis system through three orientation angles. For conciseness, we represent the terms on the left-hand side as V_h and ω_h. While the steps up to this point are essential for further development of the code and integration into a full vehicle simulation model, the isolated rotor performance code can essentially begin at the hub as noted in the above equations. From this point, we now look at the rotor in the rotating axes in which the angle ψ is the time integral of the rotor spin rate with the assumption that the rotor is spinning at constant speed counterclockwise:

$$\psi = \int_0^t \Omega dt$$

A special transformation matrix is built to handle the fact the rotor is spinning about the negative z-axis and this resolves vectors in the fixed system to vectors in the rotating system.

$$T_\psi = R_z(-\psi)$$

Next, the rotor spin rate is accounted for in the rotational velocities vector as:

$$\omega_{h:r} = \omega_h + \omega_z$$

Next the rotating reference frame is considered by first calculating the aerodynamic velocity as follows for both the translational and rotational velocities.

$$V_a = V_h - V_w$$

$$\omega_a = \omega_h - \omega_w$$

The subscript w denotes the wash velocity and the swirl wake velocities.

The rotating coordinate systems are formulated as shown below for the translational and rotational velocities.

$$\omega_r = T_\psi \omega_a$$

$$V_r = T_\psi V_a$$

Next, the individual blade section elements are examined considering manufactured twist, anhedral, and sweep after formulating the blade element section pitch.

$$\theta = \theta_0 + A_1 \cos(\psi) + B_1 \sin(\psi)$$

Accordingly, the prescribed blade angular motion is built below through a series of transformation matrices for each of the x-, y-, and z- angular transformations (twist, anhedral, and sweep).

$$prx = R_x(\phi_{x0})$$

$$pry = R_y(\phi_{y0})$$

$$prz = R_z(\phi_{z0})$$

ϕ_{x0} is defined as the prescribed blade twist at a specified radial station accounting for the blade section pitch:

$$\phi_{x0} = \theta_{twist} - \theta$$

Similarly, ϕ_{y0} and ϕ_{z0} are the prescribed or manufactured anhedral and sweep. The prescribed angular orientation of the blade is finally given as the full transformation through each of the defined angles.

$$prr = prx * pry * prz$$

After accounting for the prescribed motion of the blade, the dynamic flapping transformation matrix is built as the y-transformation matrix presented previously with respect to the flapping angle β.

$$frmtx = \begin{bmatrix} \cos(\beta) & 0 & \sin(\beta) \\ 0 & 1 & 0 \\ -\sin(\beta) & 0 & \cos(\beta) \end{bmatrix}$$

Note, this transformation matrix accounts for the sign convention of negative beta around the y-axis, which we have defined as positive flapping.

Then the final transformation from the shaft normal plane to the blade section axes is given below.

$$prrf = frmtx * prr$$

Now that the final transformation to the blade section axes is taken care of, the forces and moments can be derived. First, the flapped rate is developed below.

$$V_{arb} = V_{ar} + \omega_{ar} \times P_{rs} + \dot{P_{rs}}$$

$$V_{abf} = prrf * V_{arb}$$

The tangential velocity and the perpendicular velocity components are used to calculate the local blade section angle of attack as follows:

$$U_T = V_{abf}(2)$$

$$U_P = V_{abf}(3)$$

$$\alpha = \tan^{-1}\left(\frac{U_p}{U_T}\right)$$

Next, the local blade section dynamic pressure is calculated and then an analytic approximation of a NACA0012 airfoil is used to generate the section CL, CD, and CM coefficients. After the coefficients are calculated, the differential lift, drag, and pitching moments are determined and the aerodynamic forces and moments are temporarily stored in two 3x1 vectors.

$$dF_a = \begin{bmatrix} 0 \\ -dDrag \\ -dLift \end{bmatrix}$$

$$dM_a = \begin{bmatrix} dMoment \\ 0 \\ 0 \end{bmatrix}$$

The negative of the angle of attack returns the force and moment vectors from the local blade section axes to the fixed hub:

$$r\alpha = R_x(-\alpha)$$

The distributed aerodynamic forces and moments in the rotating frame are given by the following transformation:

$$dF_{ar} = prrf^T * r\alpha * dF_a$$

$$dM_{ar} = dM_a + P_{rs} \times dF_{ar}$$

Additionally, the flapping moment and lead-lag moment about the respective discrete hinges are calculated as the cross product of the distributed forces and moments and the distance along the blade reference line from the local blade section element to the hinge.

$$dM_f = P_{flap} \times dF_{ar}$$

$$dM_l = P_{lag} \times dF_{ar}$$

Traditionally, a radial integration is then performed, and the forces are summed in the rotating reference frame before finally being transformed back to the fixed frame; however, for this program, weighted integration factors are calculated during the initialization portion of the program and used here to succinctly sum the forces and moments in the rotating reference frame. Then the forces and moments are transformed through the transpose of the T_ψ matrix and the transpose of the T_{mast} matrix to have the forces and moments in the fixed shaft-normal plane reference frame.

3.3 Effect of Inertial and Aerodynamic Rates on Flapping

3.3.1 Inertial Rate Sanity Check and Integration Technique Study

From theory, for a rigid rotor with zero pilot inputs, no hinge offset and a natural flapping frequency of 1.00, we can use this quasi-static rotor flapping sanity check.

$$a_1 = \frac{p_a}{\mathit{\Omega}} - \frac{16}{\gamma \mathit{\Omega}} q_i$$

$$b_1 = -\frac{q_a}{\mathit{\Omega}} - \frac{16}{\gamma \mathit{\Omega}} p_i$$

An input of inertial pitch rate would give flapping magnitudes related to the equations below.

$$a_1 = -\frac{16}{\gamma \mathit{\Omega}} q_i$$

$$b_1 = -\frac{q_a}{\mathit{\Omega}} = -\frac{(q_i - q_w)}{\Omega} = -\frac{q_i}{\Omega}$$

For an input of inertial roll rate, the expected flapping can be estimated as the following.

$$a_1 = \frac{p_a}{\mathit{\Omega}} = \frac{(p_i - p_w)}{\Omega} = \frac{p_i}{\Omega}$$

$$b_1 = -\frac{16}{\gamma \mathit{\Omega}} p_i$$

To perform this model performance check, the trim routine is muted within the program and the rotor properties are set such that the rotor has a unity non-dimensional flapping frequency. This allows for a more appropriate “apples-to-apples” comparison of the performance analysis program to the quasi-static (QS) rotor results. The results are presented in two fashions, inertial rate checks with an investigation in time integration techniques and a time history comparison of the integration techniques to QS analytical results for pure cyclic inputs to a hovering rotor with uniform inflow. Note, in the figures below, a1 BER and b1 BER denoted in the legend reflect the performance analysis program results and the additional two legend entries, a1 QSR and b1 QSR reflect quasi-static analysis results. Figure 3-7 provides a look at the comparison of the RPM code estimation of the flapping response to a sweep of roll rate inputs to a rotor in hover. Figure 3-8 shows a comparison of the RPM code estimation of the flapping response to a sweep of pitch rate inputs to a rotor in hover. Both results are done with no lead-lag degree of freedom and the Pitt-Peters three state dynamic inflow model – to be discussed in a later section.

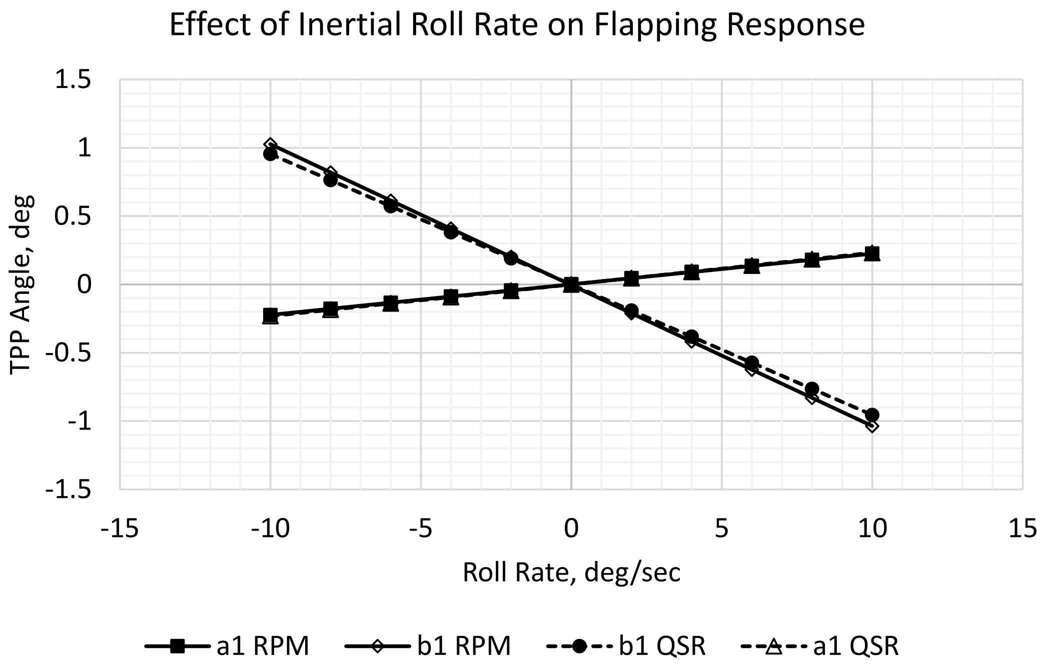


Figure 3-7 Flapping response to roll rate.

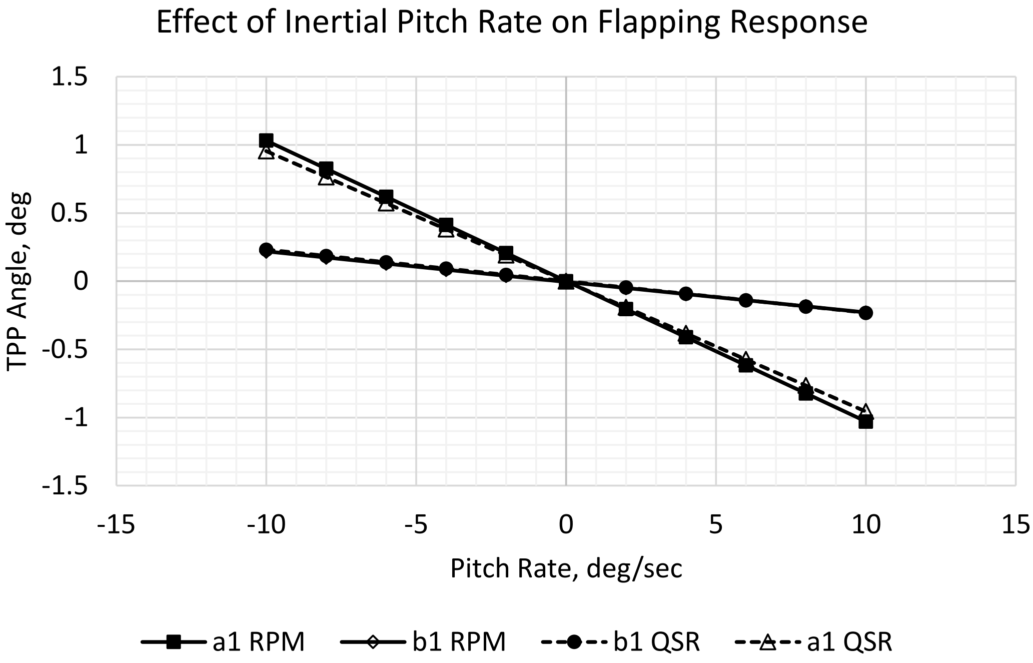


Figure 3-8 Flapping response to pitch rate.

Additionally, TPP is a common abbreviation to denote the tip path plane of the rotor in which the blade tips of the rotor form a “plane” which shows the directionality of the thrust vector as shown in the image of the V-22 tiltrotor below [10].

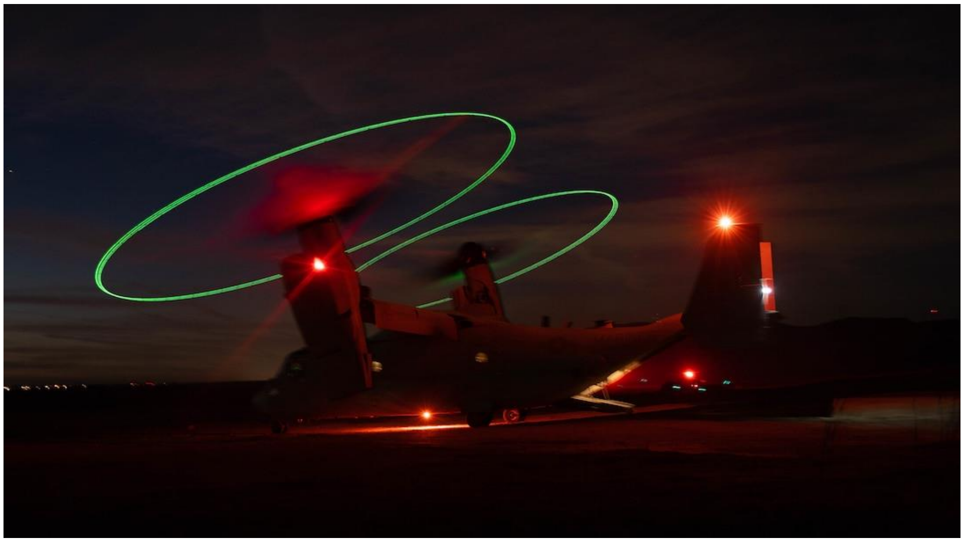


Figure 3-9 Visual representation of the rotor tip path planes for a V-22 Osprey.

From the time histories study, it is not easily discernible that a specific integration technique reigns supreme; however, after further analysis it can be shown that the RK-4 methods (both classic and using Gill’s coefficients) prove to have the smallest amplitude error in comparison to what is suggested by QS analysis. It should be noted that this was only found as an improvement in the third decimal place and at the cost of computational time (even as small and insignificant as this may seem with today’s computing power). From this finding, RK-2 or RK-3 prove to be the best options regarding computational time and are the recommended integration schemes for the performance analysis code. Further discussion will be given for each of the integration schemes in Section 6.

CHAPTER 4. INFLOW MODELING

In the section below, the three inflow models currently implemented in the program are discussed, demonstrated, and compared to Glauert's Momentum Theory hypothesis as a baseline.

4.1 Momentum Theory and Glauert's Hypothesis

Momentum theory, in general, deals with the application of Newton's second law to the field of fluid dynamics. Specifically, for rotorcraft this fluid is air and is extended to the ideal actuator disk through the work of Rankine, Froude, and Greenhill. The ideal actuator disk is modeled with an infinite number of infinitesimally thin blades positioned normal to the flow of the fluid. The control volume is then drawn rectangularly, just large enough to capture the air flow into the disk and long enough for the wake to contract in the far downstream. Figure 4-1 graphically demonstrates this.

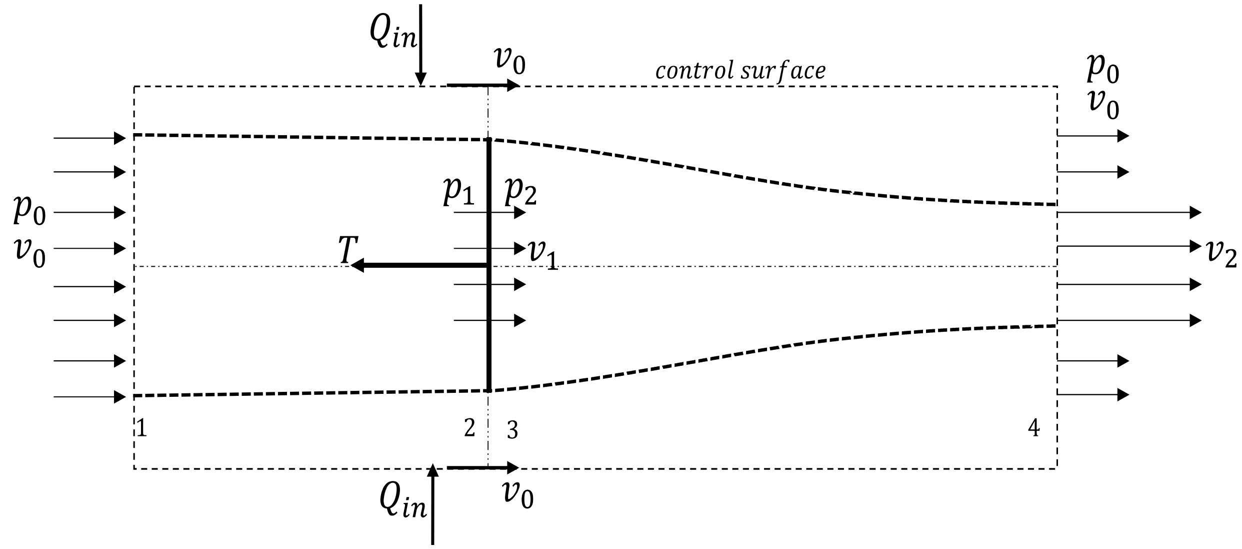


Figure 4-1 Actuator disk representation of momentum theory.

From this the wash velocity can be developed as Glauert's momentum model in the following equation [8].

$$w = \frac{T}{2\rho A\sqrt{(v_o + w)^2}}$$

The interested reader can look through several different derivations and methods to arrive at Glauert's model including Prouty [11], Dreier [12], and McCormick [2]. The comparison of the Glauert momentum model and the 1-state uniform inflow model in the blade element rotor model is seen in Figure 4-2.

4.2 1 State Inflow Model

4.2.1 Pitt-Peters Development

The 1-state, uniform inflow model that is implemented in the performance analysis code is an adaptation of the Pitt-Peters static portion of the 3-state dynamic inflow model. The development begins with creating a first-order, non-linear differential equation building from the Glauert momentum model [12].

$$M\dot{w}_w + (2\rho A|V_a|)w_w = T$$

Where the apparent mass is the amount of air that is accelerated by the induced velocity within a sphere with a diameter equal to the rotor diameter at the current atmospheric density.

$$M = \frac{4}{3}\pi R^3 \rho$$

However, from potential flow theory, the container volume is the following.

$$volume = \frac{8}{3\pi} R^3$$

From the expression above, the mass is given as:

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

Then solving the original non-linear differential equation for the wash velocity derivative, we find:

$$\dot{w_w} = \frac{(T - 2\rho A|V_a|w_w)}{\left(\frac{8}{3\pi}R^3\rho\right)}$$

This derivative is appended to the blade state variable vector where it is solved for and updated at the same time as the blade states.

4.2.2 Comparison to Momentum Theory

Using a twisted blade with uniform distributed mass and NACA 0012 airfoils for the blade element rotor model, the comparison below was completed using a transient solution method in which the rotor was allowed to rotate 5 times per airspeed and the wash velocity was then calculated as the mean during the last revolution. It is clear from Figure 4-2 the 1-state model agrees exceedingly well with the analytical method present through Glauert's momentum hypothesis.

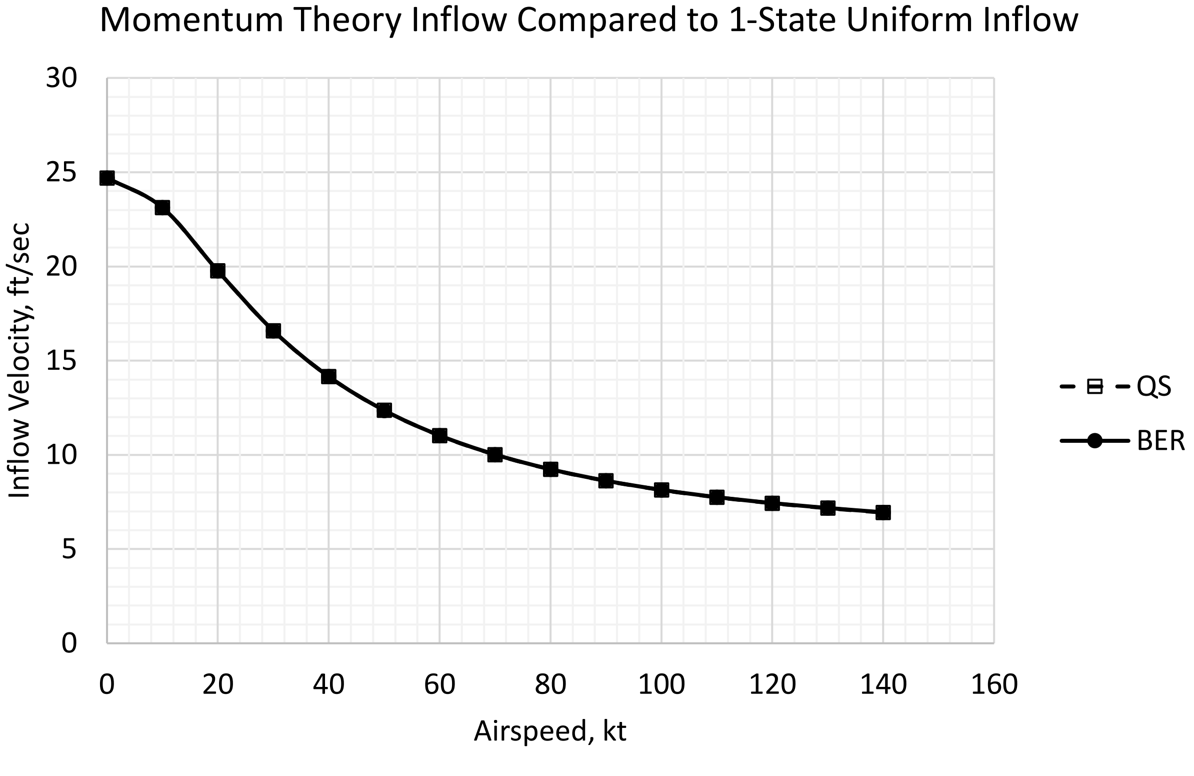


Figure 4-2 RPM code comparison to the quasi-static analysis estimation of inflow velocity for uniform inflow.

4.3 3 State Dynamic Inflow Model

4.3.1 Pitt-Peters Development

The three-state dynamic inflow model by Pitt and Peters is a computationally efficient method to produce higher fidelity simulation and modelling of the rotor downwash that correlated well with test data as described in reference [13-16]. The development follows below.

First, several definitions are put in place. The thrust is denoted, T, and is the negative of the aerodynamic force in the z-direction, $\mathcal{L}$ denotes the aerodynamic roll moment, and M denotes the aerodynamic pitching moment for the rotor. These values are then made non-dimensional using the following expressions:

For forces:

$$\frac{1}{\rho A_d V_t^2}$$

For moments:

$$\frac{1}{\rho A_d R V_t^2}$$

where $A_d$is the area of the rotor disk, $V_t = \Omega R$ is the rotor tip speed in hover, R is the rotor radius and ρ is the density of the air.

The non-dimensional thrust, roll, and pitch coefficients are stored in a vector for usage in the following steps. Several more non-dimensional values are defined – they are the advance ratio, aerodynamic inflow ratio, inertial inflow ratio, and the total velocity along with the wake skew angle.

$$\mu = \frac{u_b}{V_t} \equiv advance\ ratio$$

$$\lambda_a = \frac{-w_a}{V_t} \equiv aerodynamic\ inflow\ ratio$$

$$\chi = \tan^{-1}\left(\frac{\mu}{\lambda_a}\right) \equiv wake\ skew\ angle$$

$$V_{tot} = \sqrt{\mu^2 + \lambda_a^2} \equiv total\ velocity$$

Then, the non-dimensional wash velocity, swirl wake velocities, and mean velocity are defined.

$$v_0 = \frac{w_w}{V_t}$$

$$v_s = \frac{p_w}{\Omega}$$

$$v_c = \frac{q_w}{\Omega}$$

$$v_w = \begin{bmatrix} v_0 \\ v_s \\ v_c \end{bmatrix}$$

$$v_m = \frac{\mu^2 + \lambda_a(\lambda_a + v_0)}{V_{tot}}$$

Next, the apparent mass matrix of the airflow through the plane of the rotor disk is defined.

$$Mass = \begin{bmatrix} \frac{8}{3\pi} & 0 & 0 \\ 0 & -\frac{16}{45\pi} & 0 \\ 0 & 0 & -\frac{16}{45\pi} \end{bmatrix}$$

Now the L matrix, which relates the loads to the induced velocity and induced rates is defined:

$$L = \begin{bmatrix} \frac{1}{2V_{tot}} & 0 & \frac{15\pi}{64v_m}\tan\left(\frac{\chi}{2}\right) \\ 0 & -\frac{4}{\mathrm{v_m}\left(1+\cos(\chi)\right)} & 0 \\ \frac{15\pi}{64\mathrm{V_{tot}}}\tan\left(\frac{\chi}{2}\right) & 0 & -\frac{4\cos(\chi)}{\mathrm{v}_m\left(1+\cos(\chi)\right)} \end{bmatrix}$$

The wake velocity differential equation is solved to find the derivative states of the vertical wash velocity, the roll swirl wake and pitch swirl wake as follows:

$$vwd = Inv(Mass) * \left[\begin{bmatrix} C_T \\ C_L \\ C_M \end{bmatrix} - Inv(L) * \begin{bmatrix} v_0 \\ v_s \\ v_c \end{bmatrix}\right]$$

Where $C_T = \frac{T}{\rho A_d V_t^2}, C_L = \frac{\mathcal{L}}{\rho A_d R V_t^2}, and\ C_M = \frac{M}{\rho A_d R V_t^2}$. Finally, the results are brought back to the dimensional world as the following.

$$wwd = vwd(1) * V_t * \Omega$$

$$pwd = vwd(2) * \Omega^2$$

$$qwd = vwd(3) * \Omega^2$$

These states are included in the blade state vector (BSV) within the rotor performance model code which is detailed in a later section – but it is important to note that storing these values in the BSV allows RPM to drive the blade to a periodic solution and close the thrust-induced velocity loop simultaneously.

4.3.2 Comparison to White and Blake Approximation

The White and Blake enhancement was developed in response to data that Frank Harris [16] measured in wind tunnel testing that showed lateral flapping data that did not agree with theory for low advance ratios. The following seven charts showcase the current code comparison with the uniform inflow model, the 1-state Blake/White model, and the 3-state dynamic inflow model adapted from Pitt and Peters. The Blake and White enhancement will not show up in the power or axial induced velocity charts below.

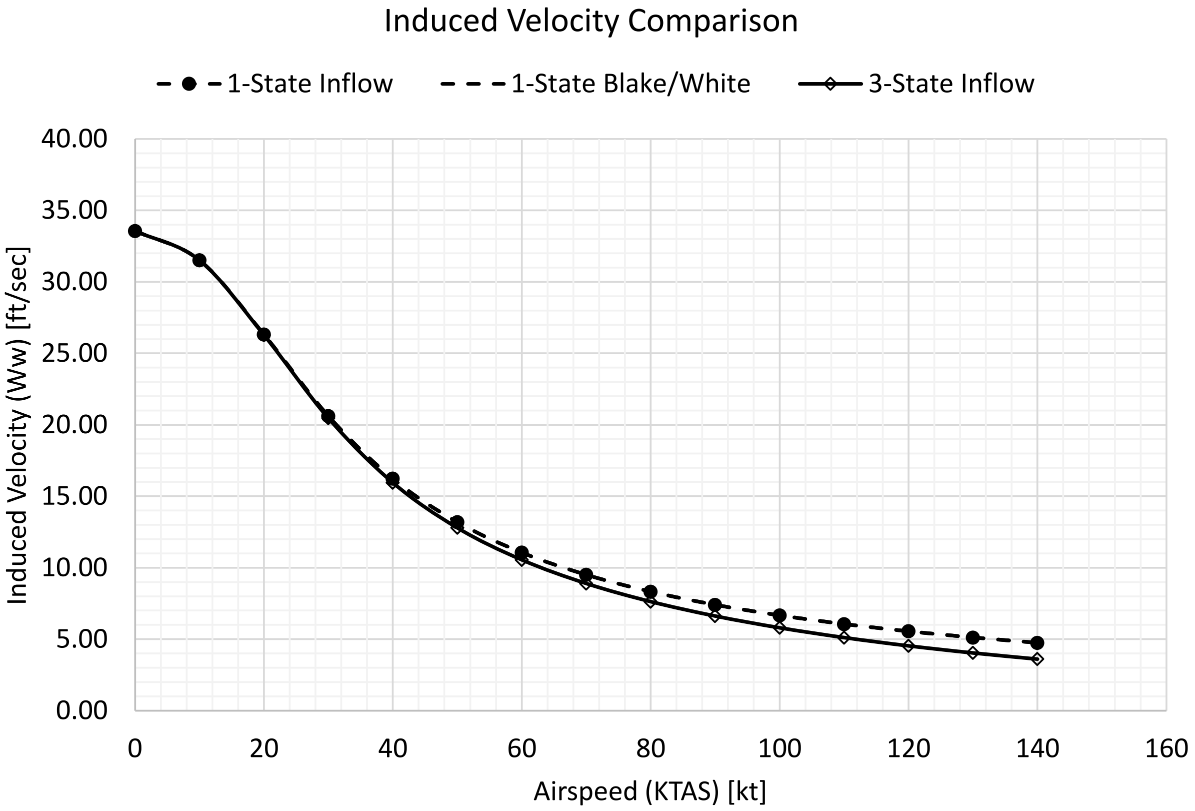


Figure 4-3 Induced velocity comparison for the three inflow models in the RPM code.

It is worth noting the 1-state inflow model and the 1-state model with the Blake/White enhancement for swirl wake applied produce the same results, as shown in the chart above.

The power required for trimmed level flight is demonstrated below for the current three inflow models held within the program. The difference in the power required is approximately 10 Hp at the highest airspeed captured at 140 knots.

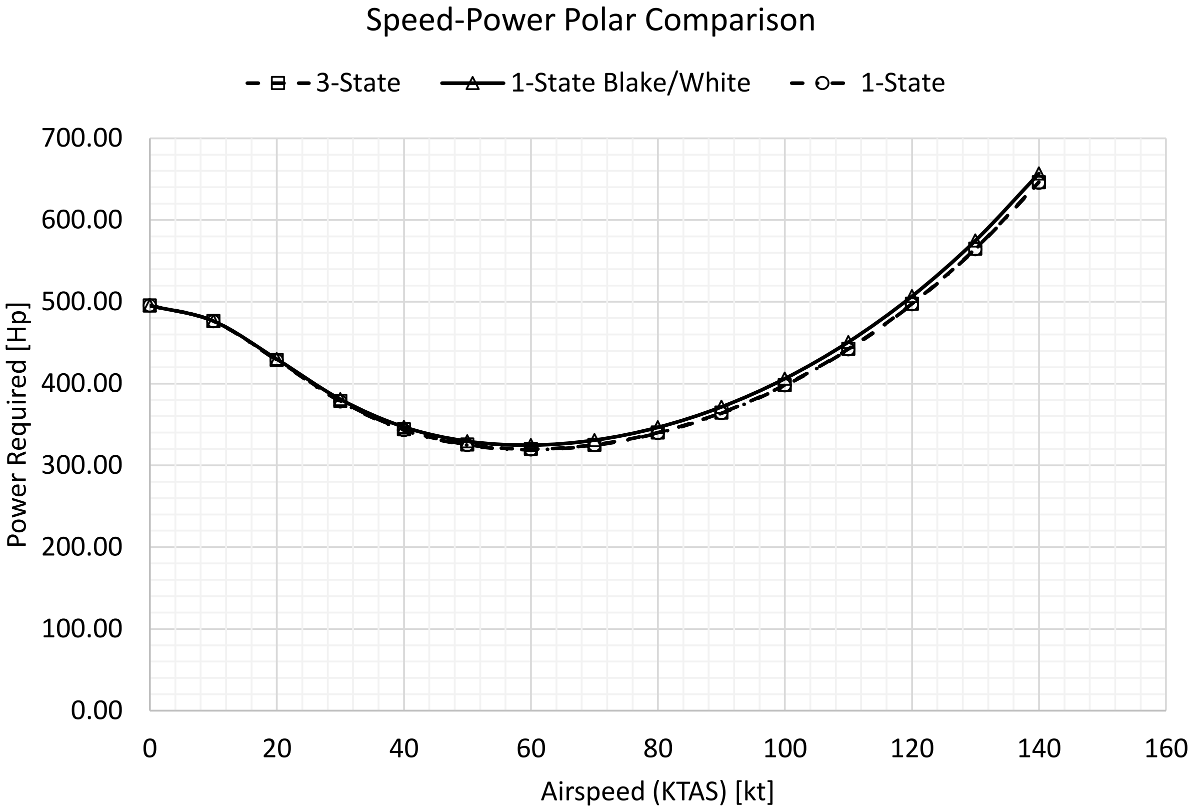


Figure 4-4 Power required using the three different inflow models in the RPM code.

In Figure 4-5 and 4-6, the swirl wake rate and lateral flapping magnitudes are demonstrated showing an amplitude difference but a qualitative similarity between the 3-state dynamic model and 1-state model with the Blake/White correction for swirl wake. Figure 4-6 also shows the importance of getting the inflow model correct – the difference between the 3-state model and the 1-state model is a difference of between 0.2 and 0.5 degrees of lateral flapping. This difference in lateral flapping can mean roughly 45 to 50 pounds non-trimmed lateral force and several hundred foot-pounds of unbalanced moment for a rotor delivering 5000 pounds of thrust.

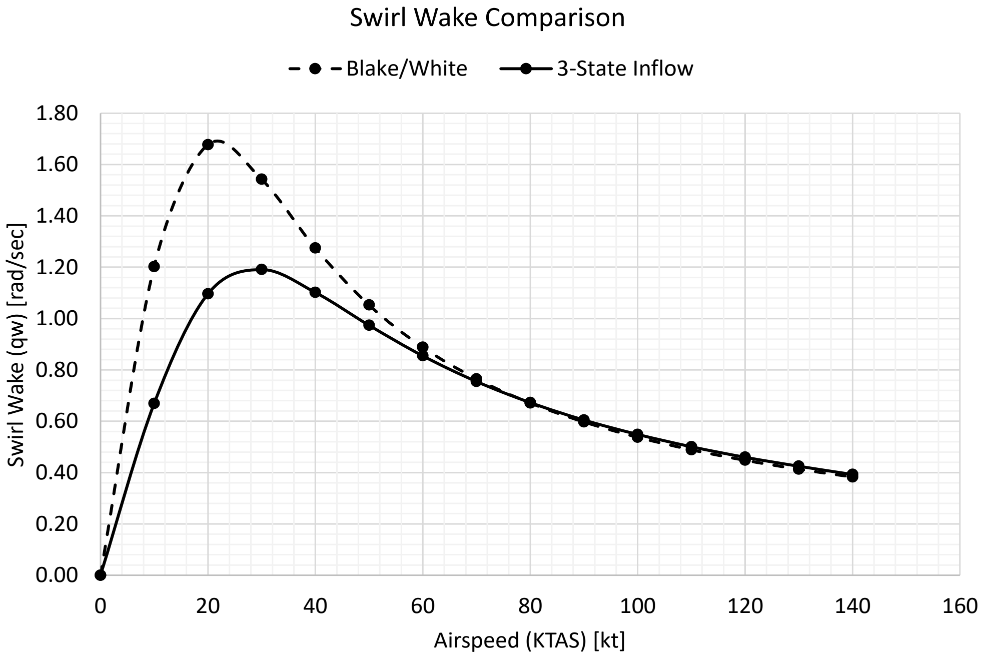


Figure 4-5 Swirl wake magnitude comparison for Blake/White enhancement and Pitt-Peters dynamic 3-state inflow model.

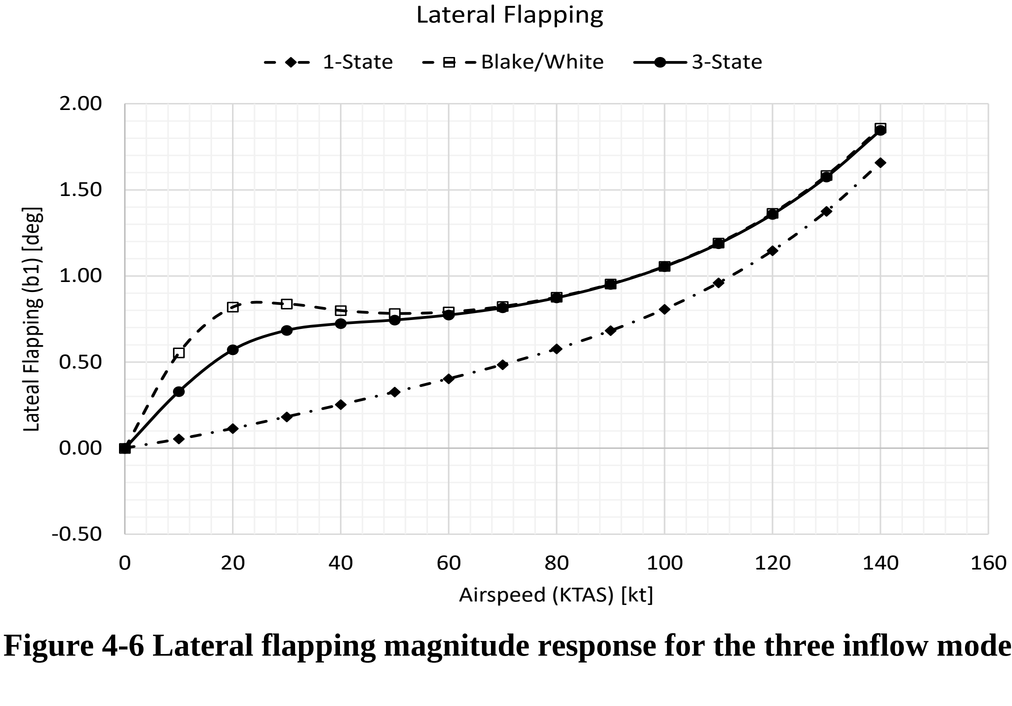


Figure 4-6 Lateral flapping magnitude response for the three inflow models.

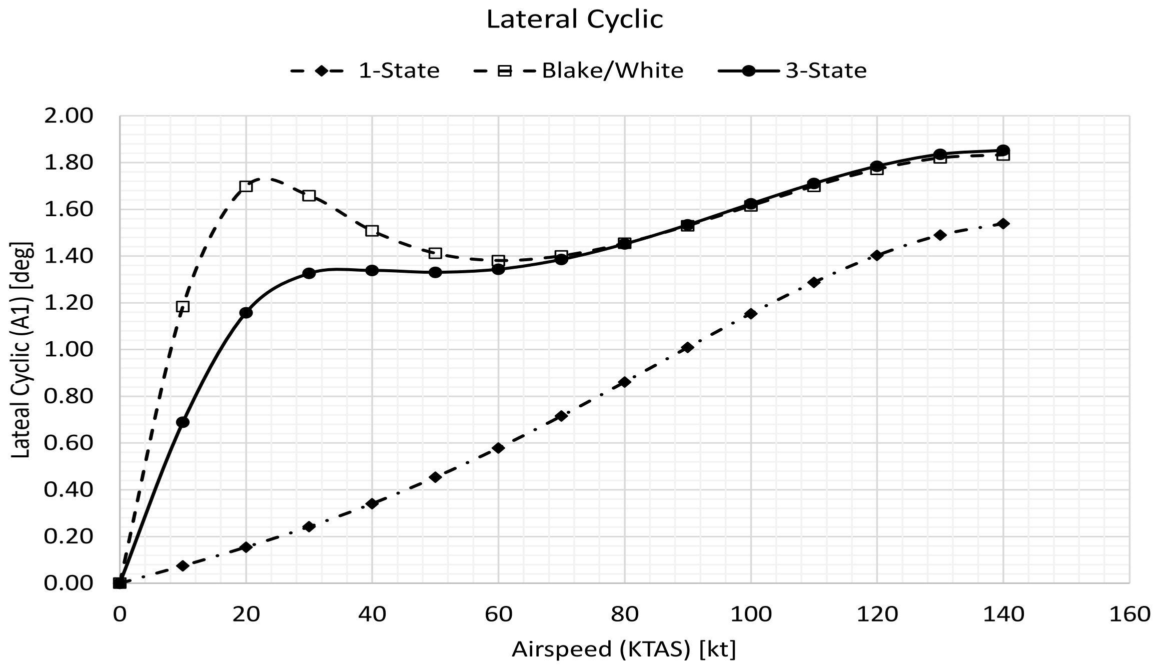


Figure 4-7 Lateral cyclic comparison for uniform inflow model, Blake/White enhanced inflow model, and Pitt-Peters 3-state dynamic inflow model.

This same amplitude difference is reflected in figure 4-7 in the lateral cyclic required for trimming the aircraft in level flight.

In Figure 4-8, it is seen that the longitudinal cyclic required to trim the rotor is relatively unchanged by the inflow model.

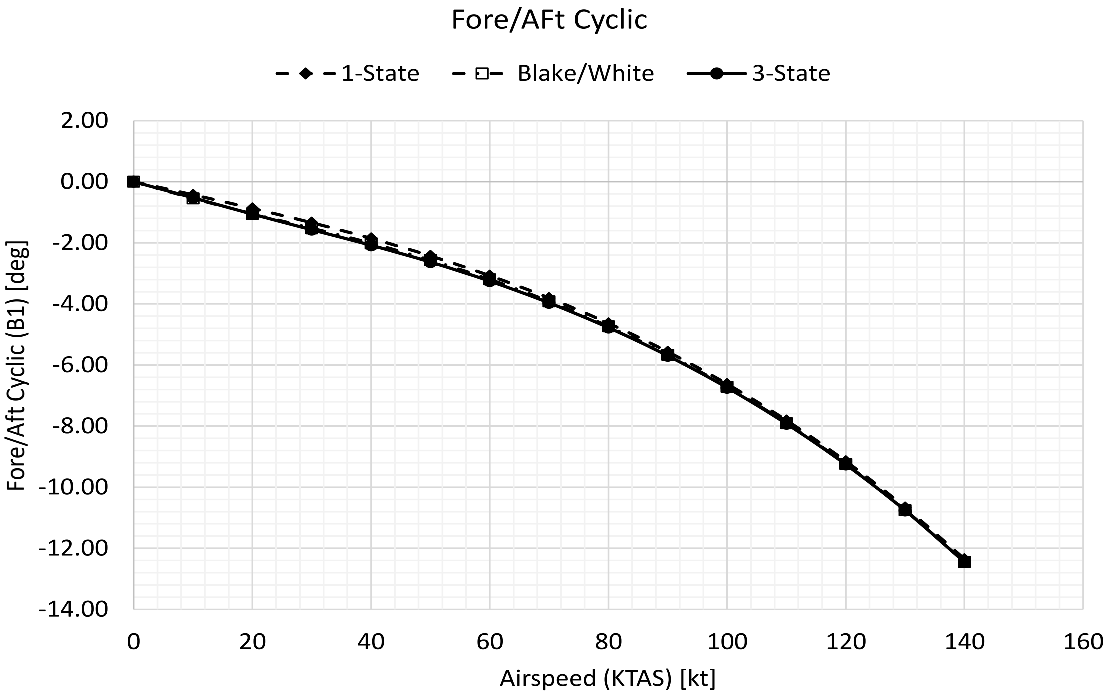


Figure 4-8 Longitudinal cyclic magnitude for uniform inflow model, Blake/White enhanced inflow model, and 3-state dynamic inflow model

Figure 4-9 shows the relative indifference of the longitudinal flapping to the inflow model.

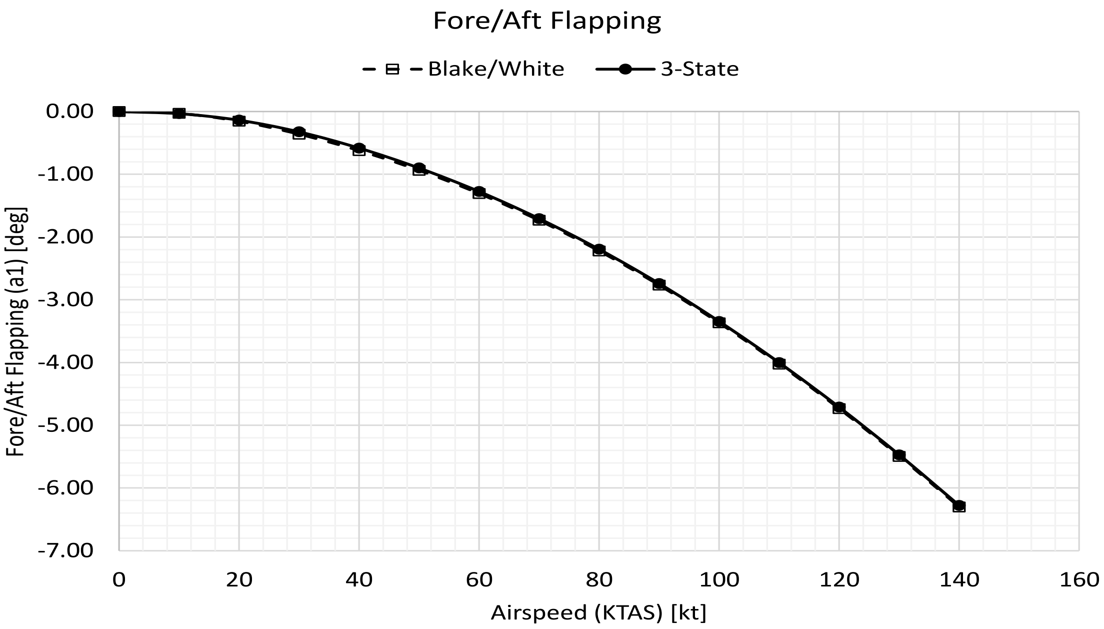


Figure 4-9 Longitudinal flapping response for the Blake/White enhanced and 3-state dynamic inflow models.

CHAPTER 5. TRIM

In this section, the various trim methods used in the development of the performance analysis code are discussed. These methods are the fly-to-trim or transient-decay solution, the Newton-Raphson method, the periodic solution method, and the Hybrid Newton-Raphson periodic shooting technique. For each method, the general procedure, benefits, and usage limitations are described.

5.1 Meaning of Trim

A trimmed aircraft is an aircraft that can maintain an attitude or state without additional inputs to the control surfaces, i.e. the aircraft is unaccelerated. The goals for a trimmed rotor are to drive the rotor to desired forces and moments, achieve a periodic solution for the blade motion, and close the thrust induced velocity loop.

5.2 Transient Solution Method

The transient solution is a natural, non-forced trim method in which the rotor will trim due to natural damping properties. The implementation of this “trim” model is very straight forward in that the rotor is given initial states and frozen control settings and then allowed to rotate for a specified number of revolutions. Given sufficient damping, the transient response decays and then the rotor system can be analyzed for the specific control inputs and body axis states over a simulated period. However, no corrections to control inputs are made to drive the rotor forces and/or moments to desired values, so this method is not very useful in a performance analysis.

5.3 Newton-Raphson Method

The Newton-Raphson method is a very commonly used routine to trim an aircraft’s forces and moments using input controls and system responses in an iterative fashion. Appendix B has the pseudo-code flowchart of the process. The Newton-Raphson method can be summarized with the following process:

1) Specify the problem (number of degrees of freedom will define the number of trim variables and trim targets).
2) Perform the sub-zero call to the rotor model to establish an initialization point for the rotor forces, moments, and blade states.
3) Build the Jacobian matrix which is the sensitivity matrix for the system responses with respect to the trim variables.
4) Calculate corrections to the trim variables using the inverse of the Jacobian and the difference between the sub-zero rotor results and the desired results.
5) Calculate corrections to the initial blade states by using the final values of the previous sub-zero pass, i.e., the blade states at azimuth of 360 degrees, as the initial values for the next sub-zero call.
6) Update the blade states and the trim variables.
7) Check if the rotor is trimmed by comparing the trim targets to the current rotor forces and moments.
    a. If the rotor is trimmed, exit the trim loop and let the rotor rotate for a specified number of revolutions to produce results.
    b. If the rotor is not trimmed, repeat the steps 3 through 5 with updates to the trim variables and targets each iteration.

The mathematical formulation of the process is described below for vector functions [12]. Application to a scalar problem is a trivial subset of this process. Assuming a function is specified as f(x) and the continuous derivative is given by $\boldsymbol{F} = \frac{\partial f(x)}{\partial x}$ the first goal is to find a root of the function such that f$(x) = 0$. Expanding f(x) into a Taylor series expansion gives the following.

$$\mathrm{f}(x_i + \Delta x) = f(x_i) + F(x_i)\Delta x + H.O.T.$$

Where H.O.T are higher order terms that we are not interested in and $x_{i+1} = x_i + \Delta x$. From this, the next step is to set f$(x_{i+1}) = 0$ and solve for Δx.

$$\Delta x = -\left(F(x_i)\right)^{-1} f(x_i)$$

The derivative of the vector function forms what is called the Jacobian matrix or shortened as the "Jacobian." The Jacobian is an M $\times N$ matrix which houses the partial derivatives of the i'th element of f(x) with respect to the j'th element of ∂x.

$$\boldsymbol{f}(x + \partial x) = \begin{Bmatrix} f_1(x) \\ \vdots \\ f_M(x) \end{Bmatrix} + \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \cdots & \frac{\partial f_1}{\partial x_N} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_M}{\partial x_1} & \cdots & \frac{\partial f_M}{\partial x_N} \end{bmatrix} \begin{Bmatrix} \partial x_1 \\ \vdots \\ \partial x_N \end{Bmatrix}$$

$$\mathrm{J}(x) = \frac{d}{dx}\boldsymbol{f}(x) = \begin{Bmatrix} \frac{\partial}{\partial x_1} \\ \vdots \\ \frac{\partial}{\partial x_N} \end{Bmatrix} \{f_1 \quad \cdots \quad f_M\} = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \cdots & \frac{\partial f_1}{\partial x_N} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_M}{\partial x_1} & \cdots & \frac{\partial f_M}{\partial x_N} \end{bmatrix}$$

In practice, M = N and the Jacobian is square and invertible. After the Jacobian is formulated, Δx can be found by taking the inverse of the Jacobian and pre-multiplying the vector function f(x) by it. Therefore, the Newton-Raphson method for a vector function is given by the following.

$$\Delta x = -\mathrm{J}(x)^{-1\boldsymbol{f}}(x)$$

From the Δx calculations, the trim variables are updated as mentioned before and the process is repeated until roots of the vector function are found producing a trimmed state for the aircraft.

5.4 Periodic Solution

5.4.1 Quasi-static Method

The quasi-static method assumes a periodic solution such as a Fourier series to represent the time domain solution for the dependent variable. The derivatives of the series are easily found. Then, all of the expansions are substituted into the differential equation and like terms of the sines and cosines and steady are gathered into equations, effectively reducing the differential equation to a set of simultaneous algebraic equations which are solved with linear algebra techniques.

5.4.2 Periodic Shooting

The general periodic shooting technique is described by taking two equations and finding a value of the independent variable such that the two equations are equal. While this technique is not specifically used within the performance analysis code, the technique builds the foundation for the hybrid Newton-Raphson Periodic Shooting method described in the next section. The basic formulation for both scalar and non-scalar equations for the periodic shooting method is described in the following paragraph.

The goal of the periodic shooting technique is to drive the value of x for two independent functions $\mathrm{f}(x)$ and $\mathrm{g}(x)$ such that $\mathrm{f}(x) = \mathrm{g}(x)$ over a given period. Expanding both equations using a Taylor series through the first derivative and truncating higher order terms gives the following:

$$\mathrm{f}(x_{i+1}) = \mathrm{f}(x_i) + \frac{\partial f}{\partial x}\Delta x$$

$$\mathrm{g}(x_{i+1}) = \mathrm{g}(x_i) + \frac{\partial g}{\partial x}\Delta \mathrm{x}$$

Setting the equations equal and solving for Δx gives the following.

$$\Delta x = \frac{f(x_0) - g(x_0)}{\frac{\partial g}{\partial x} - \frac{\partial f}{\partial x}}$$

Note the expression above is appropriate for scalar functions only.

The formulation for a vector equation follows a similar process as was presented for scalar equations above and continued from the presentation of the vector equations from the Newton-Raphson method. Additionally, the process can work such that the two equations represent the state of a periodic forced function such that f(x) is equal to the state at zero and g(x) is the state at 2π. Then,

$$\mathrm{F}(x) = \frac{\partial f}{\partial x}$$

And then the derivative of the second equation g(x) is given such that the change in the final value gives the following.

$$\mathrm{G}(\mathrm{x}_0) = \frac{\partial \mathrm{x}_\mathrm{f}}{\partial \mathrm{x}_\mathrm{f}} = 1$$

Then following the same process as above for the linear equations we arrive at the following expressions that drive $x_f = x_0$.

$$x_f = 1 \cdot \Delta x = \mathrm{f}(x_0) + \mathrm{F}(x_0)\Delta x$$

$$x_0 + \Delta x = \mathrm{f}(x_0) + \mathrm{F}(x_0)\Delta x$$

Then after solving for Δx the following is the resulting equation summarizing the process for the periodic shooting technique.

$$\Delta x = -\big(I - F(x_0)\big)^{-1}\big(x_0 - f(x_0)\big)$$

The identity matrix, $I,$ is used to demonstrate that for vector equations, the unity matrix is used in place of 1 as shown in the scalar equations.

5.5 Hybrid Newton-Raphson Periodic Shooting Method

The significant advantage of the hybrid trim technique is the ability to perturb the system for forces and moments sensitivity as well as blade states and wake states. The first part of the process is to build the loads portion of the matrix and the second part is to build the motion portion of the matrix. This is the hybrid scheme that is used as the loads and motions sections of the trim matrix are built separately within the same iteration as shown below. The end trim matrix is built towards the following final form.

$$\begin{bmatrix} NR & HL \\ HM & PS \end{bmatrix}$$

Where NR is the forces and moments sensitivities to control inputs as described in the Newton-Raphson method above and PS is the periodic shooting method for the motions portion including blade states and wake states. HL and HM are the hybrid sections for the loads and motions. respectively. This is expanded upon below from the top-level overview matrix above.

$$\begin{bmatrix} \frac{\partial F_1}{\partial C_1} & \cdots & \frac{\partial F_1}{\partial C_n} & \frac{\partial F_1}{\partial BSV0(1)} & \cdots & \frac{\partial F_1}{\partial BSV0(n)} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ \frac{\partial F_n}{\partial C_1} & \cdots & \frac{\partial F_n}{\partial C_n} & \frac{\partial F_n}{\partial BSV0(1)} & \cdots & \frac{\partial F_n}{\partial BSV0(n)} \\ \frac{\partial BSV2PI(1)}{\partial C_1} & \cdots & \frac{\partial BSV2PI(1)}{\partial C_n} & \frac{\partial BSV2PI(1)}{\partial BSV0(1)} & \cdots & \frac{\partial BSV2PI(1)}{\partial BSV0(n)} \\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots \\ \frac{\partial BSV2PI(n)}{\partial C_1} & \cdots & \frac{\partial BSV2PI(n)}{\partial C_n} & \frac{\partial BSV2PI(n)}{\partial BSV0(1)} & \cdots & \frac{\partial BSV2PI(1)}{\partial BSV0(n)} \end{bmatrix}$$

Where the partitioning is such that the top left and bottom right are the traditional Newton Raphson and Periodic Shooting Jacobian matrices respectively and the bottom left, and top right are the hybrid partitions. The notation BSV2PI represents the blade state variable vector values at two pi, or after one complete revolution. The notation BSV0 is the same vector but at the initial state of the revolution. The process to build the above matrix is described below.

1) Specify the problem (number of degrees of freedom will define the number of trim variables and trim targets).
2) Perform the sub-zero call to the rotor model to establish an initialization point for the rotor forces, moments, and blade states hybrid section of the trim matrix.
3) Form the Newton-Raphson and loads hybrid portion of the trim matrix.
4) Perform the motions sub-zero call to the rotor model to establish an initialization point for the blade state and wake state sensitivities by perturbing each state and then returning the state to the initial point.
5) Form the periodic shooting and hybrid motions portion of the trim matrix.
6) Next the trim matrix is multiplied by negative one and the PS section is extracted, and the identity matrix of the same size is added to the trim matrix to facilitate moving the section to the right-hand side of the equations.
7) Next, update the blade states and the trim variables (controls).
8) Check if the rotor is trimmed by comparing the trim targets to the current rotor forces and moments.
    a. If the rotor is trimmed, exit the trim loop and let the rotor rotate for a specified number of revolutions to produce results.
    b. If the rotor is not trimmed, repeat the steps 2 through 8 with updates to the trim variables and targets each iteration. For this step, the central difference derivatives are used and is explained in a later section.

The important distinction between the regular Newton-Raphson method and the hybrid Periodic Shooting and Newton-Raphson technique is the ability to perturb blade states and wake states all within the same trim iteration as the loads. For clarity, the blade states vector (BSV) for an N-bladed rotor which is allowed both lead-lag motion and out of plane flapping motion and a three-state wake structure is constructed below.

$$\mathrm{BSV} = [\beta_1 \quad \dot{\beta}_1 \quad \gamma_1 \quad \dot{\gamma}_1 \quad \cdots \beta_n \quad \dot{\beta}_n \quad \gamma_n \quad \dot{\gamma}_n \quad p_w \quad q_w \quad w_w]$$

Where n represents the blade number, such that for a two-bladed rotor the block of blade states would be repeated once more or for a four bladed rotor there would be four blocks of blade states.

5.6 Trim Study

When executing the code with the Newton-Raphson technique, flapping trim was achieved in 3-5 iterations. However, the Newton-Raphson technique performs poorly for very lightly damped systems such as the lead-lag degree of freedom. Therefore, the NR-PS hybrid technique is introduced and performs especially well for either a single degree of freedom or multiple degree of freedom systems.

5.6.1 Numerical Differentiation Schemes

Additionally, the implementation of the NR-PS hybrid technique presented difficulty initially due to gross estimation of a derivative by large finite differences using a forward differencing scheme. At times, the coarseness of the forward differencing scheme led to a numerical instability that obviated any chance at a solution. A central difference derivative scheme stabilized the trim iteration; this is presented below for clarity.

The forward differencing scheme allows for the estimation of the derivative between an initial point and a point one step forward.

Forward Difference Scheme

$$f'(x) \cong \frac{\mathrm{f(x + \Delta h) - f(x)}}{\mathrm{\Delta h}}$$

Where Δh acts as a perturbation value in which the system will be excited and then allowed to return to during analysis. Similarly, a backward scheme can be constructed using the previous value of f(x) and while these schemes provide some level of confidence in a well-behaved first order system, the central difference scheme is most appropriate for a twice differentiable system giving the best approximation for a derivative with little worry for the solution going unstable.

Backward Difference Scheme

$$f'(x) = \frac{f(x) - f(x - \Delta h)}{\Delta h}$$

Central Difference Scheme

$$f'(x) = \frac{\mathrm{f(x + \Delta h) - f(x - \Delta h)}}{\mathrm{2\Delta h}}$$

Figure 5-1 below presents the three differentiation schemes graphically. Suppose we want the derivative of the function at "x." The forward difference derivative and backward difference derivative would clearly provide disparate estimates of the derivative at "x". The central difference derivative provides a better approximation for the function derivative. Obviously, reliance on forward or backward difference derivatives can lead to larger errors in estimates of system sensitivity to x.

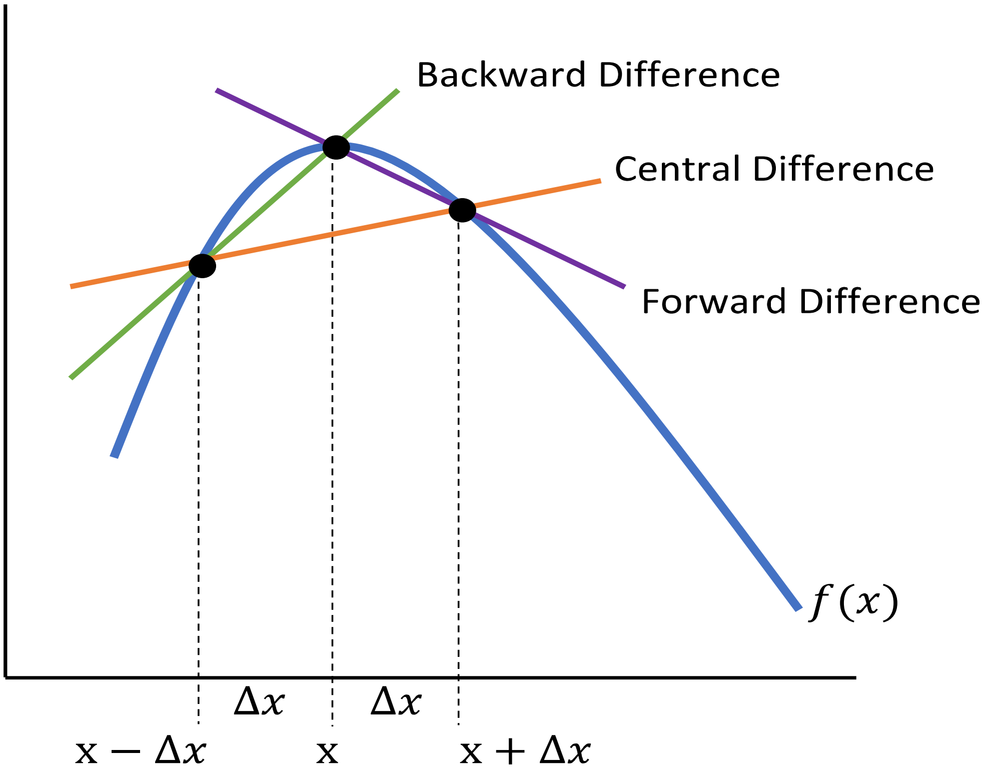


Figure 5-1 Numerical Differentiation Schemes

CHAPTER 6. INTEGRATION METHODS

6.1 Open and Closed Form Development

The integration techniques available within the program include the series of Runge-Kutta algorithms: RK-2, RK-3, classical RK-4, and RK-4 with Gill coefficients [17]. This study was conducted get an understanding of how the flapping amplitude was affected in comparison to the expected results from the quasi-static analysis. The test implemented produced the results in the following table below.

Table 6-1 Amplitude agreement for four integration techniques when compared to QS results

Amplitude Agreement		
Integration Technique	Lateral Response, deg	Longitudinal Response, deg
QS Model	5.0000	5.0000
RK-2	4.9851	4.9858
RK-3	4.9856	4.9865
RK-4 (classic)	4.9863	4.9871
RK-4 (Gill)	4.9863	4.9871

As would be expected, the higher order Runge-Kutta algorithm does produce the best agreement to the quasi-static model results; however, it should be noted that the computational time required was greater (smaller for the single case but could easily provide a delay for extended performance in the program such as trim speed sweep). The uniform inflow model was used in this study along with the transient response analysis option allowing the rotor to let the transients naturally damp out to a periodic solution.

For the RK-2 method, also known as the Improved Euler method, an initial value problem is presented below along with the pseudo code implementation.

$$y_{i+1} = y_i + k_2$$

$$k_1 = hf(x_i, y_i)$$

$$k_2 = hf\left(x_i + \frac{1}{2}h, y_i + \frac{1}{2}k_1\right)$$

$$x_i = x_o + ih$$

The derivative function is f(x,y) and h is the step size for the independent variable. These definitions hold for the other methods described below.

For the RK-3 implementation, there are three integrations per step in the program as described below.

$$y_{i+1} = y_i + \frac{1}{6}(k_1 + 4k_2 + k_3)$$

The three steps are given below.

$$k_1 = hf(x_i, y_i)$$

$$k_2 = hf\left(x_i + \frac{1}{2}h, y_i + \frac{1}{2}k_1\right)$$

$$k_3 = hf(x_i + h, y_i - k_1 + 2k_2)$$

$$x_i = x_0 + ih$$

The RK-4 implementation extends the RK-3 algorithm to include a fourth integration per step.

$$y_{i+1} = y_i + \frac{1}{6}(k_1 + 2k_2 + 3k_3 + k_4)$$

$$k_1 = hf(x_i, y_i)$$

$$k2 = hf\left(x_i + \frac{1}{2}h, y_i + \frac{1}{2}k_1\right)$$

$$k_3 = hf\left(x_i + \frac{1}{2}h, y_i + \frac{1}{2}k_2\right)$$

$$k_4 = hf(x_i + h, y_i + k_3)$$

$$x_i = x_o + h$$

The same number of integrations are used in the RK-4 with Gill coefficients, but the coefficients are adjusted as presented below.

$$y_{i+1} = y_i + \frac{1}{6}\left(k_1 + \left(2 - \sqrt{2}\right)k_2 + \left(2 + \sqrt{2}\right)k_2 k_3 + k_4\right)$$

$$k_1 = hf(x_i, y_i)$$

$$k2 = hf\left(x_i + \frac{1}{2}h, y_i + \frac{1}{2}k_1\right)$$

$$k_3 = hf\left(x_i + \frac{1}{2}h, y_i + \frac{-1 + \sqrt{2}}{2}k_1 + \frac{2 - \sqrt{2}}{2}k_2\right)$$

$$k_4 = hf\left(x_i + h, y_i - \frac{\sqrt{2}}{2}k_2 + \frac{2 + \sqrt{2}}{2}k_3\right)$$

$$x_i = x_0 + h$$

It is worth noting there are higher order methods available to be used, but they come with a computational cost as this is implemented within the trim loop of the program and with each additional call to the rotor model to find a better amplitude agreement, the computational time increases. From the study performed, it is recommended that the Improved Euler method be used for most cases as the differences between the results using the RK-2 and the RK-4 method are not appreciable and therefore do not play a major role in the performance calculations and results.

CHAPTER 7. RPM CAPABILITY SAMPLE

To test the performance estimation of the RPM code, namely the ability to estimate power required, a single main rotor-tail rotor configuration helicopter was modeled. Only the main rotor was modeled, and Table 7-1 gives the main rotor parameters that were used, Appendix C has the rotor definition file and the output summary file for conciseness.

Table 7-1 Rotor Comparison Input Parameters

Item	Value	Unit
Blade Twist	0.0000	rad
Tip Loss Factor	1.0000	ND
Tip Speed	756.0000	ft/sec
Air Density	2.378E-03	slug/ft^3
Radius	17.5000	ft
Disk Area	962.1128	ft^2
Average Drag Coefficient	0.0150	ND
Linear Lift-Curve Slope	5.7300	CL/rad
Number of Blades	4	ND
Solidity	0.0631	ND
Thrust Equivalent Chord	0.8667	ft

In figure 7-1, the performance calculation from the RPM code is compared to two sets of information: 1) the speed power polar build up from Rotary Aerodynamics Volume II [18] and 2) the calibrated performance data of the modeled aircraft for just the main rotor.

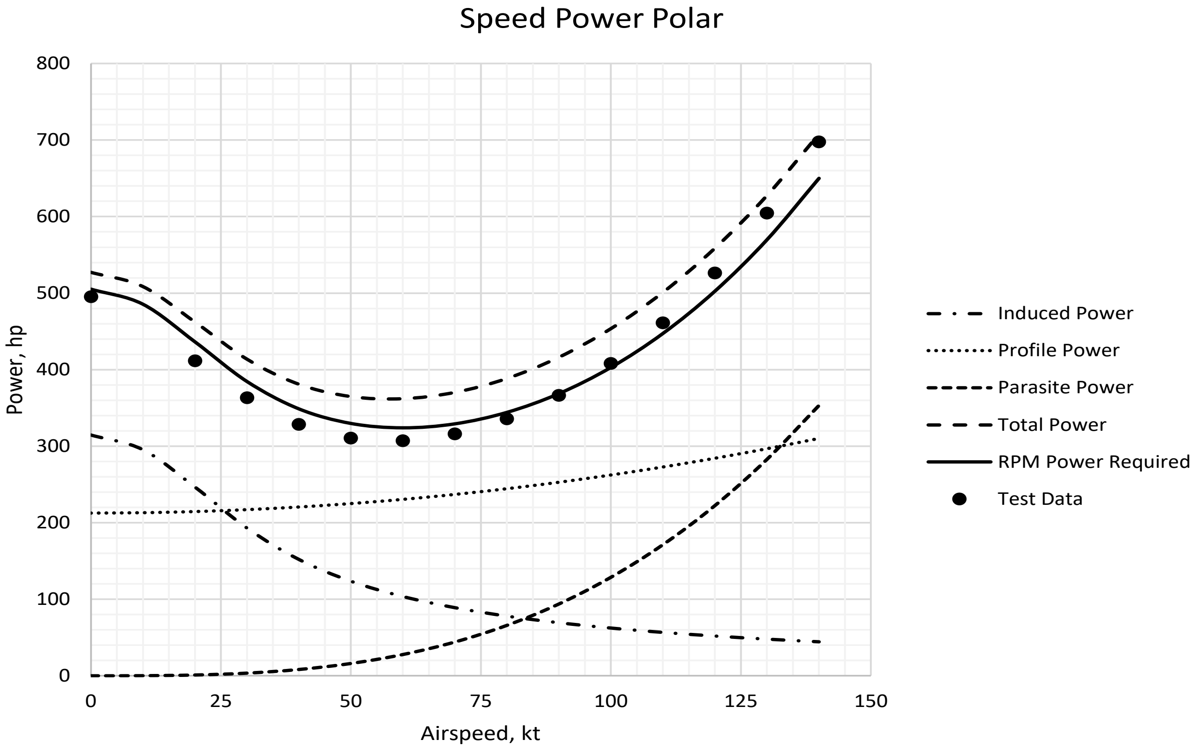


Figure 7-1 5000 LB SMR Main Rotor Performance Estimation Validation

The three main components of the power curve are the induced power, the profile power, and the parasite power with the total power required as the summation of these components. First, the induced power captures the effect of inflow velocity turning the thrust vector slightly in the drag direction; the induced power is given in the equation below.

$$P_{induced} = \frac{T v_{ind}}{550}$$

In the expression above, T is the rotor thrust, v_{ind} is the induced axial or "inflow/wash" velocity, and the 550 constant is used to produce units of horsepower. Secondly, the profile power captures the effect of the blade whirling around the azimuth and drag being created in doing so – profile power is an effect of the drag of the airfoil, the tip speed of the rotor blade, and the forward velocity of the aircraft.

$$P_{profile} = \frac{\sigma\,\overline{C_d}(1 + 4.7\mu)\rho\pi R^2 V_t^3}{4400}$$

In the equation above, σ is the rotor solidity, $\overline{C_d}$ is the average drag coefficient for the airfoil, μ is the advance ratio of the aircraft (the ratio of the forward velocity to the tip speed of the rotor blade), ρ is the air density, R is the rotor radius, and V_t is the tip speed of the rotor blade. Lastly, for the simple power performance calculation, the parasite power is calculated and is dependent upon the forward velocity of the aircraft and the flat plate equivalent drag, f_e, of the aircraft.

$$P_{parasite} = \frac{f_e \rho V^3}{1100}$$

Thus, the full power required is the sum of the individually calculated components of the rotor power and parasite power.

$$P_{required} = \frac{T v_{ind}}{550} + \frac{\sigma\,\overline{C_d}(1 + 4.7\mu)\rho\pi R^2 V_t^3}{4400} + \frac{f_e \rho V^3}{1100}$$

Figure 7-1 shows the ease of estimating the power required using the three-term power equation from Stepniewski and Keys [18]. It is also clear the RPM code matches the measured data well. The key importance differentiating the RPM code and the three-term equation is the that three-term equation made certain assumptions including constant chord, no anhedral or sweep effects and constant drag, among other things, while the RPM code allows the user to assert rotor design points with different aerodynamic tables, geometries and fuselage drag estimates.

CHAPTER 8. OTHER FINDINGS

8.1 Trimming with Active Twist

An interesting finding while using the rotor performance model is the rotor can trim through an airspeed sweep while using variable twist blade control rather than the traditional cyclic input. This is similar to the control implementation in Kaman helicopters using servo-flaps on the blade to actively alter the "twist" of the blade as shown in Figure 8-1 [19].

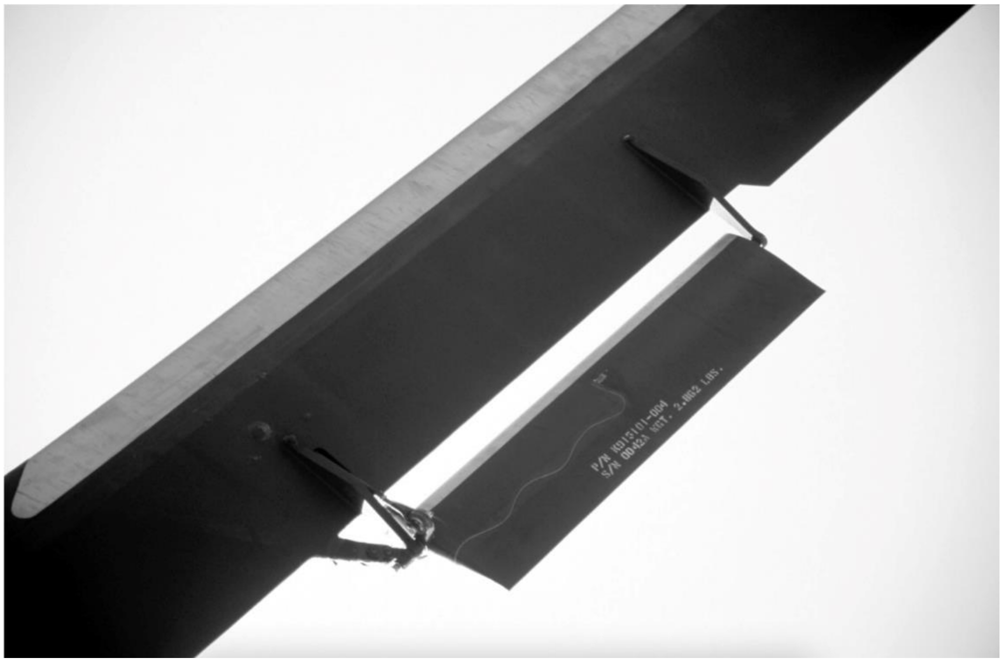


Figure 8-1 Kaman rotor blade servo-flap.

This is accomplished within the model by allowing the user to select three control inputs from a candidate list of controls. If the rotor does not have cyclic control, the user must select control variables $\theta_{TC}(x)$ and $\theta_{TS}(x)$ which provide spanwise cosine and sine harmonic variations in the blade section pitch. Additionally, the user can specify if the blade shape factor is linear, quadratic, or even higher order if more controllability in the blade section pitch is preferred.

CHAPTER 9. SUMMARY

The main difference in the RPM code in comparison to several other codes is the angle of attack is not calculated by simply adding the inflow angle to the blade section pitch. Rather, this calculation is performed through formal large angle rotations of the inertial and wash velocity vectors to generate velocity components resolved to the pitched blade section. These components are beamwise and chordwise components and the arctangent of the beamwise to chordwise ratio provides the angle of attack. This technique preserves fidelity. Using large angle rotations in the blade geometry ensures better modeling of stall sections of the blade when in the retreating flow region.

In the blade kinematics module of the code, the large angles preserve key dynamic factors in the development of loads, specifically the Coriolis loads, which are blade loads developed in-plane. The RPM code does not use normal modes – essentially modelling coning as a prescribed value augmented with small out of plane modal motion. The normal modes method weakens the 1 per rev and 2 per rev character of the in-plane loads; the RPM code calculates the accelerations accounting for the non-linear product of β and $\dot{\beta}$ preserving the non-weakened 1 per rev and 2 per rev character.

The hybrid Newton-Raphson Periodic-Shooting technique uses two well-known techniques but combines them into one overall Jacobian matrix. This allows the RPM code to join blade motion, the wash model, and control calculations to satisfy the fuselage trim problem simultaneously through iteration. This technique allows for a computationally efficient trim routine because the three trim problems (periodic blade motion, desired forces and moments, and induced velocity thrust loop closure) are solvable simultaneously rather than as separate solutions.

The RPM code has provided a capability sample in comparing power required for a 5000 LB single main rotor – tail rotor configuration. Additionally, the RPM code inertial and aerodynamic models have been validated through inertial and aerodynamic flapping response characteristics, and inflow model validation using Momentum Theory, Blake and White’s one-state enhanced model, and the Pitt-Peters three-state dynamic inflow model. The RPM code was successful in matching values exceedingly well when comparing the enhanced one-state model and the three-state dynamic inflow model to the estimated values from Momentum Theory.

CHAPTER 10. FUTURE WORK

Further work should be completed to enhance the comprehensiveness of the performance modeling code, including vortex wake theory, an additional blade state degree of freedom for torsion, and implementation of a loads structure to allow for the coupling of multiple rotor systems. The use of vortex theory will allow for the wake to be examined using wake inflow models such as fully free wake models as well as vortex particle transport methods. This will allow the user to analyze rotor states such as vortex ring state and further visualize the wake contraction and interaction with other components of the rotorcraft model. The use of a torsion degree of freedom would allow for further analysis of the pitch-flap coupling phenomenon as well as build a structure for loosely estimating pitch link loads. The use of a loads structure for multiple rotor systems within the performance model would allow for further analysis of the interactions between main rotors and tail rotors, coaxial rotor systems, tandem rotor systems, or more complicated rotor systems such as those proposed by many eVTOL companies whose systems currently have as many as 16 rotors. Additionally, a hub transfer matrix should be built to enable the modeling of different hub arrangements such as teetering, gimballed, underslung, etc.

CHAPTER 11. **REFERENCES**

[1] Gessow, A. and Myers, G. C., Aerodynamics of the Helicopter, 1999, College Park Press, College Park, Md.

[2] McCormick, B. W., Aerodynamics of V/STOL Flight, 1967, Academic Press, New York

[3] Saberi, H., Hasbun, M., Hong, J., Yeo, H., and Ormiston, R. A., “Overview of RCAS Capabilities, Validations, and Rotorcraft Applications,” American Helicopter Society 71st Annual Forum, Virginia Beach, VA, May 2015

[4] “Comprehensive Program for Theoretical Evaluation of Rotorcraft, Volume III (Copter-2000) Engineer’s Manuel.” Bell Helicopter Textron Report No. 299-099-376, March 2005.

[5] USAAMRDL-TR-76-28A, REXOR Rotorcraft Simulation Model, Volume 1 – Engineering Documentation, July 1976, Lockheed California Co., Burbank, Ca.

[6] Hoffman, J. A., Analysis Methods Incorporated in the MOSTAB-HFA Computer Code, PPI-1013-2, 1975, Paragon Pacific, Inc., El Segundo, Ca.

[7] "Rotorhead Types", *Helistart.com*. [Online]. Available: http://www.helistart.com/RotorheadTypes.aspx. [Accessed: 20- Mar- 2021].

[8] "Detail of hub of proprotor assembly of a United States Marine Corps MV-22 Osprey during maintenance, Camp Bastion, Helmand Province, Afghanistan Stock Photo 4316-5631 : Superstock", *SuperStock*. [Online]. Available: https://rgbprodweb.azurewebsites.net/stock-photos-images/4316-5631.

[9] "Bell 505 Jet Ranger X", *Bell Flight*. [Online]. Available: https://www.bellflight.com/products/bell-505. [Accessed: 20- Mar- 2021].

[10] "Night TRAP", *Marines.mil*. [Online]. Available: https://www.marines.mil/photos/igphoto/2002560119/.

[11] R. Prouty, *Helicopter performance, stability, and control,* 1st ed. Malabar, Fla.: Krieger Pub., 2005.

[12] M. Dreier, *Introduction to helicopter and tiltrotor flight simulation*, 2nd ed. Reston, VA: AIAA, 2018.

[13] Pitt, D. M. and Peters, D. A., Theoretical Prediction of Dynamic Inflow Derivatives, Vertica, Vol. 5, pp. 21-34, 1981.

[14] Chen, R. T. N., *A survey of non-uniform inflow models for rotorcraft flight dynamics and control application*, NASA Ames Research Center, Moffett Field, CA, 1989

[15] White, F., Blake, B., Improved method of predicting helicopter control response and gust sensitivity, American Helicopter Society 35th Annual Forum, Washington, D.C, May 1979

[16] Harris, F. D., Articulated Rotor Blade Flapping Motion at Low Advance Ratio, J. AHS, Vol. 17, No. 4, Jan. 1972.

[17] "Differential Equations", *Mymathlib.com*, 2004. [Online]. Available: http://www.mymathlib.com/diffeq/. [Accessed: 18- Feb- 2021].

[18] W. Stepniewski and C. Keys, *Rotary-wing aerodynamics*. New York, NY: Dover Publ., 1984.

[19] "Kaman K-MAX (K-1200) main rotor blade servo flap", *Vertical Flight Photo Gallery*. [Online]. Available: https://gallery.vtol.org/image/cvKC.

APPENDIX A. JUSTIFICATION FOR THE HYBRID PERIODIC SHOOTING – NEWTON RAPHSON TECHNIQUE

We want the periodic solution to the blade motion problem to determine the blade harmonic loads in the rotating system and the harmonic loads in the fixed system. Ideally, this solution should be completed as expediently as possible and without the imposition of a presumed harmonic solution.

Several methods have been employed in the literature and in practice. The most common are:

- The quasi-static (QS) solution method presumes the solution to the blade motion problem is harmonic. This method works by substituting a harmonic series and its time derivatives for the blade states. This reduces the differential equation to a set of simultaneous algebraic equations which can then be used to reconstruct a time history of the blade motion.
- The quasi-static solution is used as an estimate of the initial condition for a time-domain integration of the differential equation.
- A time-domain integration uses the averages of the blade states at zero and at two pi (the beginning of the revolution and the end) to serve as the initial condition of the next integration around the azimuth.
- A time-domain integration simply executes the differential equation of blade motion presumably until the transients have decayed forming a periodic (and often harmonic) solution.

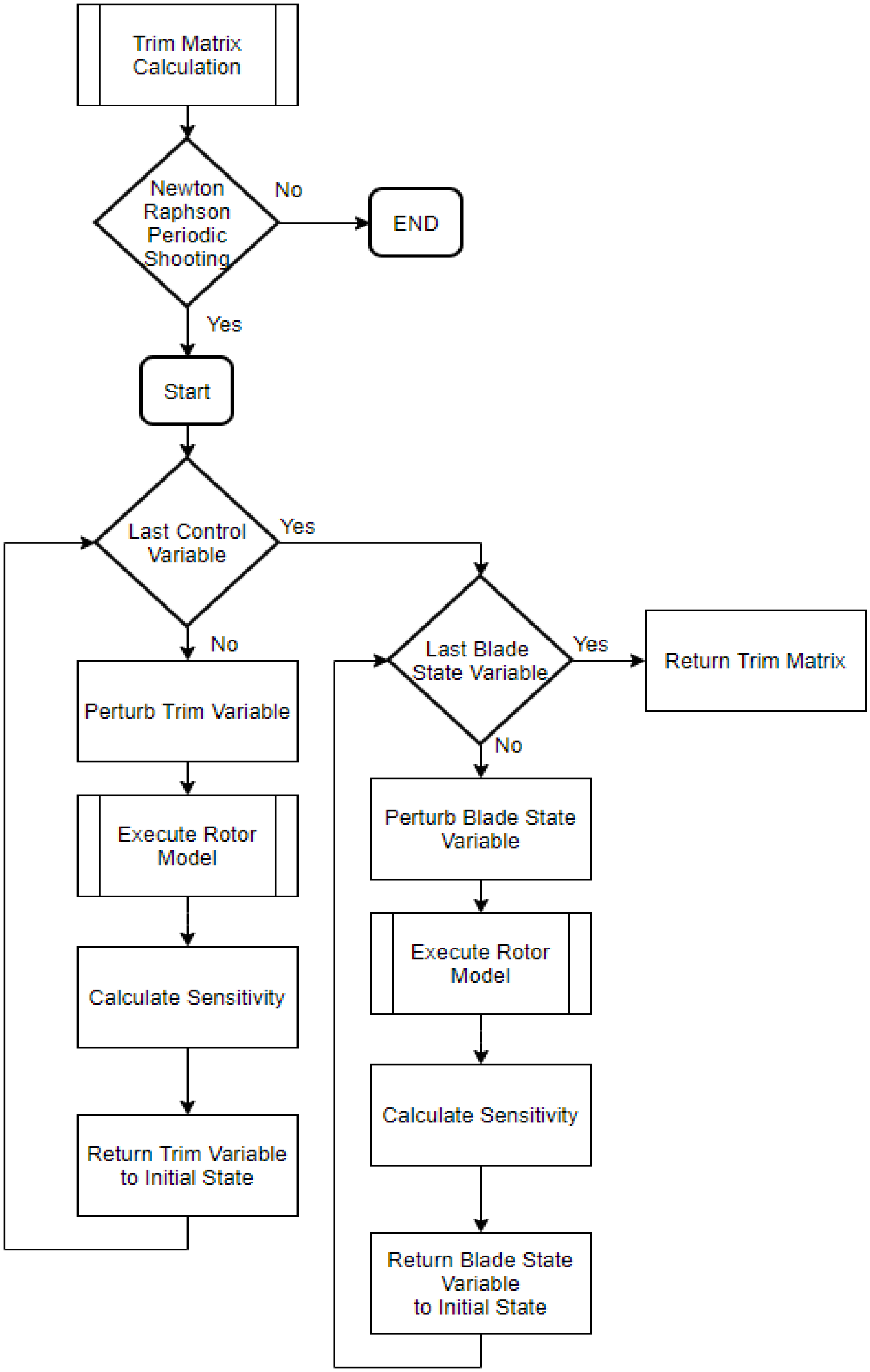
Each of these methods have drawbacks.

- The quasi-static method presumes the solution is harmonic, but often uses only the steady and first harmonic for the solution. This presumption does not adequately model the asymmetric motion of the blades around the azimuth in forward flight and obviously is forcing a harmonic solution onto what may be a rotor experiencing periodic but not harmonic excitation. In addition, the QS method requires the algebraic expansion of the differential equations into their algebraic form.

- Using the QS method as a means of estimating the initial conditions for a time-domain integration is a reasonable approach to finding the periodic solution, but it still requires the differential equation to be executed in time with enough periods to ensure the decay of the transient solution.
- The time-domain integration method that averages the blade states at 0 and 2 pi is a correction scheme which requires many iterations over a rotor period, even for linear differential equations. This is not an efficient use of computational resources.
- The method of executing the differential equation over many periods until the transient solution has decayed is problematic because it presumes the user knows how many periods are required for acceptable transient decay and because certain blade motion models are notorious for their light damping – specifically torsional and lead-lag (in-plane) motion degrees of freedom. Execution until acceptable convergence is expensive in time and computer resources.

The periodic shooting (PS) technique circumvents the problems above by employing any desired time-domain integration method along with a modified Newton-Raphson method to find rapidly the initial conditions of the blade motion states. The PS technique has the advantage of finding the correct initial conditions in one iteration for a linear differential equation. Since the blade equations of motion are only weakly non-linear, the PS method is a good candidate. In addition, it can be combined with the usual Newton-Raphson technique when trimming the rotor model for desired forces and/or moments, and it accelerates the closure of the thrust-induced velocity loop when using the algebraic form of the induced velocity model or the one-state or three-state dynamic inflow model.

APPENDIX B. FLOW CHART FOR TRIM SCHEME INTEGRATION



APPENDIX C. RPM CAPABILITY SAMPLE INPUTS AND OUTPUTS

The following information is the user input data and the calculated input data the RPM code utilized for a capability sample for a main rotor performing in hover.

```
Rotor Data
Hub-CG X-disp:      0.000  ft
Hub-CG Y-disp:      0.000  ft
Hub-CG Z-disp:      0.000  ft
Rot.Direction:      1.000  ND
Rotor Radius:      17.500  ft
Disk Area:        962.113  ft2
Number Blades:      4.000  ND
TE Chord:           0.868  ft
Solidity:           0.063  ND
Lock Number:        4.843  ND
Rotor RPM:        412.530  RPM
Tip Speed:        756.000  ft/s
Rotor Speed:       43.200  rad/sec
Natural freq:      45.621  rad/sec
Precone, B0:        0.000  deg
Delta-3, d3:        0.000  deg
Lag Hinge, el:      0.000  ft
Flap Hinge,ef:      1.250  ft
Blade Mass:         2.797  slug
First Mass M.:     24.476  slug-ft
Lead-Lag Iner:    285.914  slug-ft2
Flapp Inertia:    228.957  slug-ft2
Blade Inertia:    285.914  slug-ft2
Flap Iner.Def:     26.381  slug-ft2
LLag Iner.Def:      0.000  slug-ft2
Tip Loss Fact:      0.975  ND
Control Phase:      1.383  ??
Flap Hinge K:       0.000  ft-lb/rad
Lag Hinge K:        0.000  ft-lb/rad
ND flap freq:       1.056  ND
ND lag freq:        0.000  ND
```

```
Distributed Rotor Data
          s          dm       chord
      0.000       0.160       0.867
      0.875       0.160       0.867
      1.750       0.160       0.867
      2.625       0.160       0.867
      3.500       0.160       0.867
      4.375       0.160       0.867
      5.250       0.160       0.867
      6.125       0.160       0.867
      7.000       0.160       0.867
      7.875       0.160       0.867
      8.750       0.160       0.867
      9.625       0.160       0.867
     10.500       0.160       0.867
     11.375       0.160       0.867
     12.250       0.160       0.867
     13.125       0.160       0.867
     14.000       0.160       0.867
     14.875       0.160       0.867
     15.750       0.160       0.867
     16.625       0.160       0.867
     17.500       0.160       0.867
Manufactured Blade Shape
          s       Twist     Precone       Sweep
      0.000     -12.000       0.000       0.000
      0.875     -11.400       0.000       0.000
      1.750     -10.800      -0.000       0.000
      2.625     -10.200      -0.000       0.000
      3.500      -9.600      -0.000       0.000
      4.375      -9.000      -0.000       0.000
      5.250      -8.400      -0.000       0.000
      6.125      -7.800      -0.000       0.000
      7.000      -7.200      -0.000       0.000
      7.875      -6.600      -0.000       0.000
      8.750      -6.000      -0.000       0.000
      9.625      -5.400      -0.000       0.000
     10.500      -4.800      -0.000       0.000
     11.375      -4.200      -0.000       0.000
     12.250      -3.600      -0.000       0.000
     13.125      -3.000      -0.000       0.000
     14.000      -2.400      -0.000       0.000
     14.875      -1.800      -0.000       0.000
     15.750      -1.200      -0.000       0.000
     16.625      -0.600      -0.000       0.000
     17.500       0.000      -0.000       0.000
```

Prescribed Rotor Flapping Shape

```
        s      ----------- RPF -----------
    0.000      0.000      0.000      0.000
    0.875      0.000      0.000      0.000
    1.750     -0.500      0.000      0.000
    2.625     -1.375      0.000      0.000
    3.500     -2.250      0.000      0.000
    4.375     -3.125      0.000      0.000
    5.250     -4.000      0.000      0.000
    6.125     -4.875      0.000      0.000
    7.000     -5.750      0.000      0.000
    7.875     -6.625      0.000      0.000
    8.750     -7.500      0.000      0.000
    9.625     -8.375      0.000      0.000
   10.500     -9.250      0.000      0.000
   11.375    -10.125      0.000      0.000
   12.250    -11.000      0.000      0.000
   13.125    -11.875      0.000      0.000
   14.000    -12.750      0.000      0.000
   14.875    -13.625      0.000      0.000
   15.750    -14.500      0.000      0.000
   16.625    -15.375      0.000      0.000
   17.500    -16.250      0.000      0.000
```

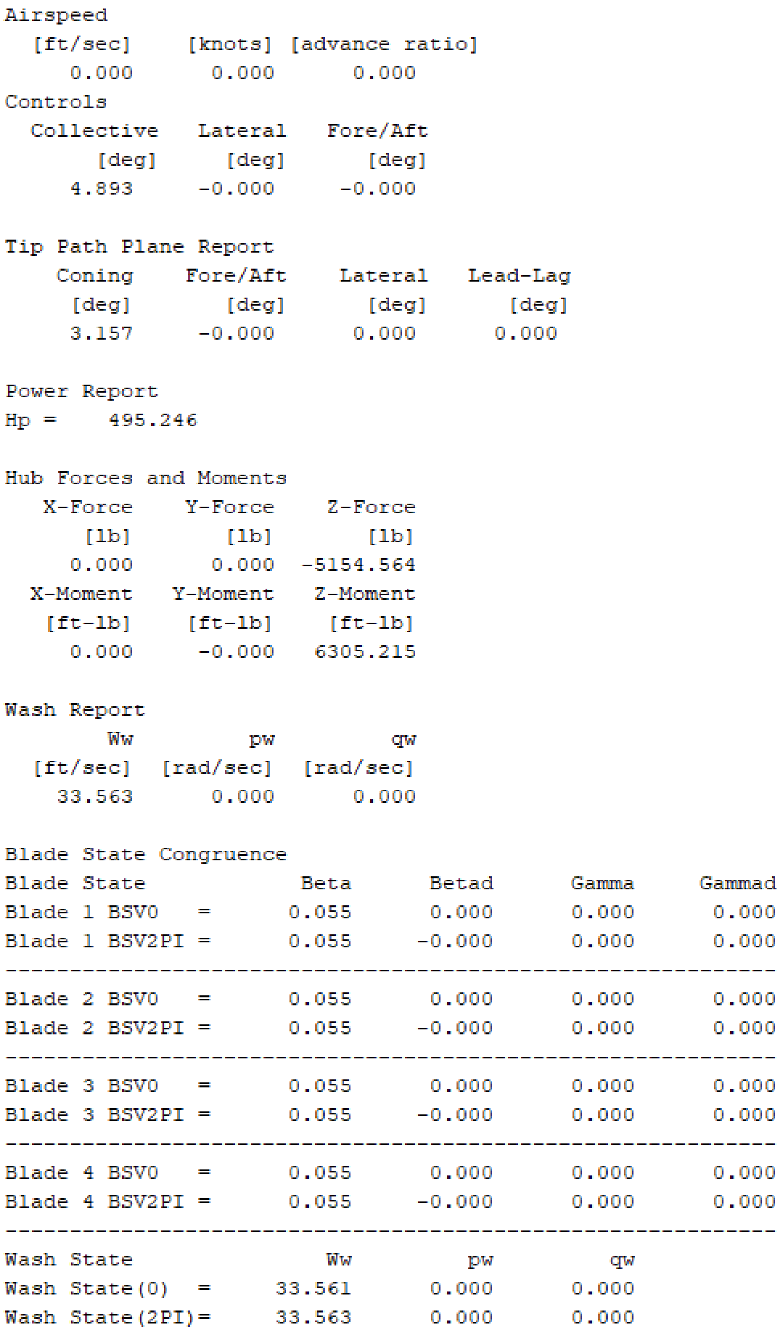
Prescribed Rotor Lead-Lag Shape

```
        s      ----------- RPL -----------
    0.000      0.000      0.000      0.000
    0.875     -0.875      0.000      0.000
    1.750     -1.750      0.000      0.000
    2.625     -2.625      0.000      0.000
    3.500     -3.500      0.000      0.000
    4.375     -4.375      0.000      0.000
    5.250     -5.250      0.000      0.000
    6.125     -6.125      0.000      0.000
    7.000     -7.000      0.000      0.000
    7.875     -7.875      0.000      0.000
    8.750     -8.750      0.000      0.000
    9.625     -9.625      0.000      0.000
   10.500    -10.500      0.000      0.000
   11.375    -11.375      0.000      0.000
   12.250    -12.250      0.000      0.000
   13.125    -13.125      0.000      0.000
   14.000    -14.000      0.000      0.000
   14.875    -14.875      0.000      0.000
   15.750    -15.750      0.000      0.000
   16.625    -16.625      0.000      0.000
   17.500    -17.500      0.000      0.000
```

Trapezoidal Integration Coefficients

s	CWTS	CWTSF
0.000	0.438	0.000
0.875	0.875	0.250
1.750	0.875	0.688
2.625	0.875	0.875
3.500	0.875	0.875
4.375	0.875	0.875
5.250	0.875	0.875
6.125	0.875	0.875
7.000	0.875	0.875
7.875	0.875	0.875
8.750	0.875	0.875
9.625	0.875	0.875
10.500	0.875	0.875
11.375	0.875	0.875
12.250	0.875	0.875
13.125	0.875	0.875
14.000	0.875	0.875
14.875	0.875	0.875
15.750	0.875	0.875
16.625	0.875	0.875
17.500	0.438	0.438

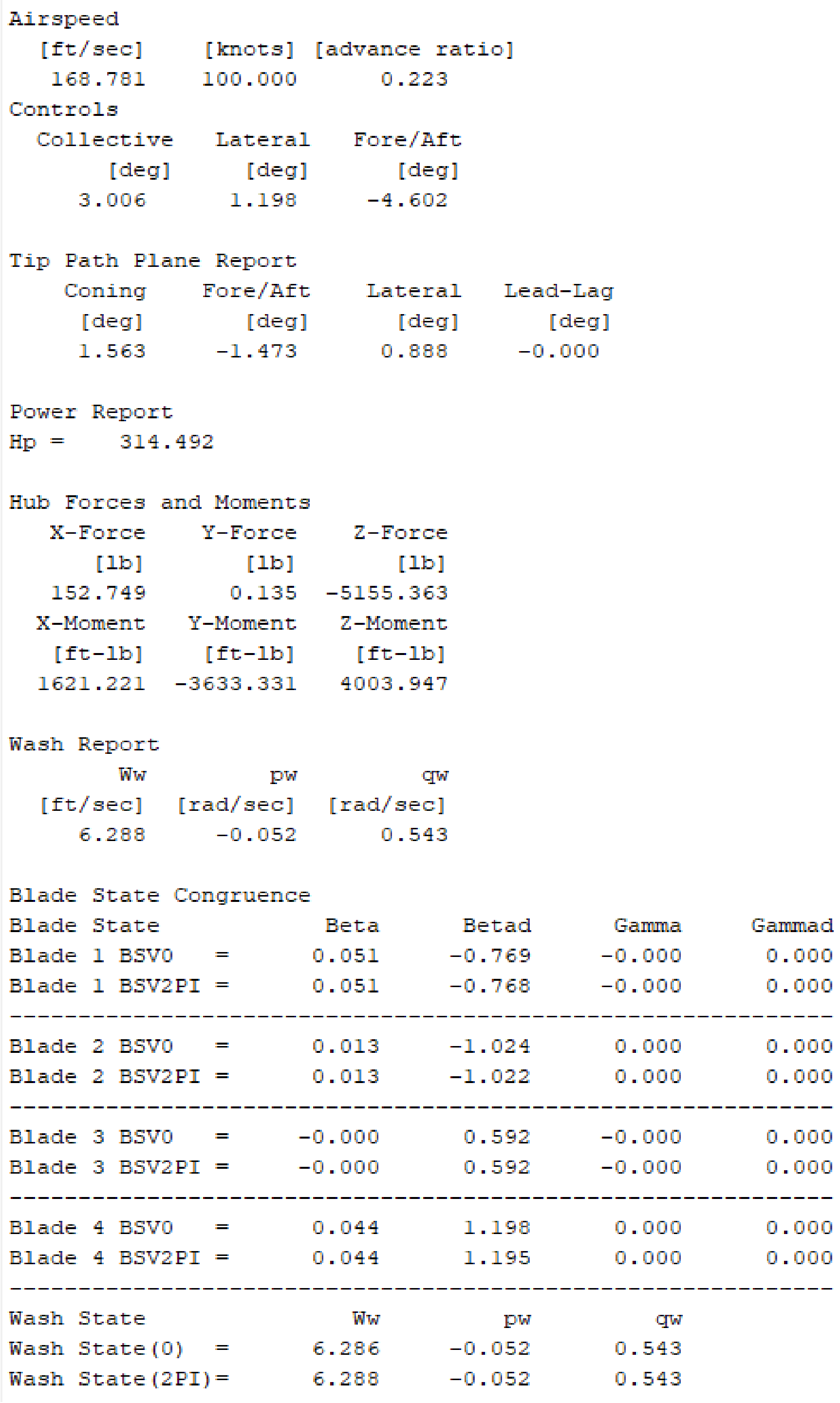
Output Screen from RPM Capability Sample for Hover



```
Airspeed
  [ft/sec]    [knots] [advance ratio]
     0.000      0.000      0.000
Controls
  Collective   Lateral   Fore/Aft
       [deg]     [deg]      [deg]
     4.893     -0.000     -0.000

Tip Path Plane Report
    Coning     Fore/Aft    Lateral   Lead-Lag
     [deg]        [deg]      [deg]      [deg]
     3.157     -0.000      0.000      0.000

Power Report
Hp =    495.246

Hub Forces and Moments
   X-Force    Y-Force    Z-Force
      [lb]       [lb]       [lb]
     0.000      0.000  -5154.564
  X-Moment   Y-Moment   Z-Moment
   [ft-lb]    [ft-lb]    [ft-lb]
     0.000     -0.000   6305.215

Wash Report
        Ww         pw         qw
  [ft/sec]  [rad/sec]  [rad/sec]
    33.563      0.000      0.000

Blade State Congruence
Blade State            Beta       Betad      Gamma      Gammad
Blade 1 BSV0   =       0.055      0.000      0.000      0.000
Blade 1 BSV2PI =       0.055     -0.000      0.000      0.000
------------------------------------------------------------
Blade 2 BSV0   =       0.055      0.000      0.000      0.000
Blade 2 BSV2PI =       0.055     -0.000      0.000      0.000
------------------------------------------------------------
Blade 3 BSV0   =       0.055      0.000      0.000      0.000
Blade 3 BSV2PI =       0.055     -0.000      0.000      0.000
------------------------------------------------------------
Blade 4 BSV0   =       0.055      0.000      0.000      0.000
Blade 4 BSV2PI =       0.055     -0.000      0.000      0.000
------------------------------------------------------------
Wash State               Ww         pw         qw
Wash State(0)   =     33.561      0.000      0.000
Wash State(2PI)=      33.563      0.000      0.000
```

The following is the output for a trimmed rotor at 100 kt airspeed with only flapping motion enabled using the 3-state inflow model.

Output Screen from RPM Capability Sample for 100 kt Airspeed



```
Airspeed
  [ft/sec]    [knots] [advance ratio]
   168.781    100.000      0.223
Controls
  Collective   Lateral   Fore/Aft
       [deg]     [deg]      [deg]
     3.006      1.198     -4.602

Tip Path Plane Report
    Coning    Fore/Aft    Lateral   Lead-Lag
     [deg]       [deg]      [deg]      [deg]
     1.563     -1.473      0.888     -0.000

Power Report
Hp =    314.492

Hub Forces and Moments
   X-Force    Y-Force    Z-Force
      [lb]       [lb]       [lb]
   152.749      0.135  -5155.363
  X-Moment   Y-Moment   Z-Moment
   [ft-lb]    [ft-lb]    [ft-lb]
  1621.221  -3633.331   4003.947

Wash Report
        Ww         pw         qw
  [ft/sec]  [rad/sec]  [rad/sec]
     6.288     -0.052      0.543

Blade State Congruence
Blade State            Beta      Betad      Gamma     Gammad
Blade 1 BSV0   =      0.051     -0.769     -0.000      0.000
Blade 1 BSV2PI =      0.051     -0.768     -0.000      0.000
------------------------------------------------------------
Blade 2 BSV0   =      0.013     -1.024      0.000      0.000
Blade 2 BSV2PI =      0.013     -1.022      0.000      0.000
------------------------------------------------------------
Blade 3 BSV0   =     -0.000      0.592     -0.000      0.000
Blade 3 BSV2PI =     -0.000      0.592     -0.000      0.000
------------------------------------------------------------
Blade 4 BSV0   =      0.044      1.198      0.000      0.000
Blade 4 BSV2PI =      0.044      1.195      0.000      0.000
------------------------------------------------------------
Wash State               Ww         pw         qw
Wash State(0)  =      6.286     -0.052      0.543
Wash State(2PI)=      6.288     -0.052      0.543
```