5-2018

A Propeller Model Based on a Modern Numerical Lifting-Line Algorithm with an IterativeSemi-Free Wake Solver

Zachary S. Montgomery
Utah State University

A PROPELLER MODEL BASED ON A MODERN NUMERICAL LIFTING-LINE ALGORITHM WITH AN ITERATIVE SEMI-FREE WAKE SOLVER

by

Zachary S. Montgomery

A thesis submitted in partial fulfillment
of the requirements for the degree

of

MASTERS OF SCIENCE

in

Mechanical Engineering

Approved:

Douglas Hunsaker, Ph.D. Major Professor	Geordie Richards, Ph.D. Committee Member
Stephen Whitmore, Ph.D. Committee Member	Mark R. McLellan, Ph.D. Vice President for Research and Dean of the School of Graduate Studies

UTAH STATE UNIVERSITY
Logan, Utah

2018

ABSTRACT

A Propeller Model Based on a Modern Numerical Lifting-Line Algorithm with an Iterative Semi-Free Wake Solver

by

Zachary S. Montgomery, Masters of Science
Utah State University, 2018

Major Professor: Douglas Hunsaker, Ph.D.
Department: Mechanical and Aerospace Engineering

Prandtl's classical lifting-line theory applies only to a single wing with no sweep or dihedral. However, the fundamental model upon which the theory is based can be extended to apply to multiple lifting surfaces, each with arbitrary sweep and dihedral. Here, this numerical lifting-line algorithm is extended further to account for the rotating motion of propeller blades as well as the induced velocity from the helical wake. An iterative semi-free wake solver is developed to account for high blade loadings at lower advance ratios. An algorithm capable of predicting the aerodynamics of propellers with arbitrary geometry including sweep and dihedral is presented, and results are compared to propeller blade theory solutions.

(87 pages)

PUBLIC ABSTRACT

A Propeller Model Based on a Modern Numerical Lifting-Line Algorithm with an Iterative Semi-Free Wake Solver

Zachary S. Montgomery

A fundamental aerodynamic analysis technique for a single straight fixed wing has been expounded upon and turned into a modern technique that can analyze multiple wings of more realistic shapes common on aircraft. This modern technique is extended further to apply towards propellers. A method to overcome propeller analysis problems at low airspeeds is presented. This method is compared to more traditional propeller analysis techniques.

ACKNOWLEDGMENTS

I'd like to thank Dr. Hunsaker for his mentoring and guidance that has helped me through this research. I am grateful to have been able to participate in this research and to closely study and build upon the fundamental principles developed by those who have gone before me. I'd also like to thank my family and friends for their continual support and encouragement towards me.

Zachary S. Montgomery

CONTENTS

LIST OF FIGURES

Figure Page

Nomenclature

$\mathrm{d}A_i$	= differential planform area at control point i
A_p	= area of the propeller disk
a	= helix radius
b	= helix pitch length
b_n	= new pitch calculated from the current iteration
b_o	= pitch from the previous iteration
$\widetilde{C}_{Li}$	= local airfoil lift coefficient
$\widetilde{C}_{Li,\alpha}$	= local airfoil lift slope
$\widetilde{C}_{mi}$	= local airfoil moment coefficient
C_P	= coefficient of power
C_T	= coefficient of thrust
c_{i_1}	= chord length for node 1 at control point i
c_{i_2}	= chord length for node 2 at control point i
d_p	= diameter of the propeller
$\hat{\boldsymbol{e}}_{\boldsymbol{T}}$	= tangent unit vector
$\hat{\boldsymbol{e}}_{\boldsymbol{\theta}}$	= unit vector in θ direction
$\vec{\boldsymbol{F}}$	= total aerodynamic force for the propeller
$\mathbf{d}\vec{\boldsymbol{F}}$	= differential aerodynamic force
$\hat{\boldsymbol{\imath}}$	= unit vector in x direction
J	= advance ratio
$[\boldsymbol{J}]$	= Jacobian matrix of partial derivatives
$\hat{\boldsymbol{\jmath}}$	= unit vector in y direction
$\hat{\boldsymbol{k}}$	= unit vector in z direction
k	= total number of steps used in the numerical integration
$\vec{\boldsymbol{l}}$	= vector representing a straight bound vortex filament
$\mathbf{d}\vec{\boldsymbol{l}}$	= directional differential vortex filament section
$\mathbf{d}\vec{\boldsymbol{l}}_{\boldsymbol{i}}$	= spatial vector along bound portion of horseshoe vortex i
ℓ	= aerodynamic torque on the propeller
$\vec{\boldsymbol{M}}$	= total aerodynamic moment on propeller
m	= number of steps/segments in a single loop
$\dot{m}$	= mass flow rate through the propeller disk
N	= total number of control points from every blade
N_b	= number of control points per blade
n	= number of loops in helix
p	= power clustering factor
$\boldsymbol{R}$	= residual vector
$\vec{\boldsymbol{r}}$	= position vector from differential filament section to control point
$\vec{\boldsymbol{r}}_{\boldsymbol{1}}$	= position vector from start of vortex filament to control point
$\vec{\boldsymbol{r}}_{\boldsymbol{2}}$	= position vector from end of vortex filament to control point
r_h	= hub radius
$\vec{\boldsymbol{r}}_{\boldsymbol{i}}$	= spatial vector from the center of the propeller to control point i
r_p	= propeller radius
s	= arc length of helix

s_{i_1} = spanwise coordinate for node 1 at control point i

s_{i_2} = spanwise coordinate for node 2 at control point i

T = thrust produced by the propeller

$\boldsymbol{\hat{u}_{ai}}$ = axial unit vector aligned with local chord line

$\boldsymbol{\hat{u}_{ni}}$ = normal unit vector perpendicular with the local chord line

$\boldsymbol{\hat{u}_{si}}$ = spanwise unit vector orthogonal to the local airfoil plane

V_{ai} = component of local velocity aligned with the chord

V_g = estimate for the axial induced velocity used for the initial condition

$\boldsymbol{\vec{V}_h}$ = induced velocity from a helical vortex filament

$\mathbf{d}\boldsymbol{\vec{V}}$ = differential velocity

$\boldsymbol{\vec{V}_i}$ = total velocity at control point i

$\boldsymbol{\vec{V}_{ji}}$ = induced velocity from horseshoe vortex j at control point i

V_{ni} = component of local velocity normal with the chord but within the local airfoil plane

$\boldsymbol{\vec{V}_{pi}}$ = velocity at control point i on the propeller due to propeller motion

$\boldsymbol{\vec{V}_\infty}$ = freestream velocity

$\boldsymbol{\vec{v}_h}$ = induced velocity from a helical vortex filament divided by the vortex strength

$V_{z_{\text{avg}}}$ = average axial velocity from all control points

$\boldsymbol{\vec{w}_i}$ = intermediate variable to aid in displaying equations

x_p = x coordinate of control point

y_p = y coordinate of control point

z_p = z coordinate of control point

α_i = local angle of attack

α_{L0i} = local zero lift angle of attack

β = local twist angle

$\mathbf{\Gamma}$ = vector of vortex strengths

Γ = vortex filament strength

Γ_i = vortex filament strength of horseshoe i

Δb = predicted change in pitch

$\mathbf{\Delta\Gamma}$ = corrector vector of vortex strengths

$\boldsymbol{\delta\vec{M}_i}$ = local aerodynamic moment

ζ = parameterization variable along vortex filament

$\mathrm{d}\zeta$ = differential element of the parameterization variable

θ = circumferential cylindrical coordinate

Λ_i = local sweep angle

λ = nondimensional helix pitch

λ_c = chord-line pitch length

$\boldsymbol{\vec{\nu}_{ji}}$ = induced velocity from horseshoe vortex j at control point i divided by the vortex strength

ρ = air density

$\boldsymbol{\vec{\sigma}}$ = nondimensional velocity

ϕ = change of variables used for cosine clustering

ψ = helix pitch angle

Ω = relaxation factor for Newton's method

Ω_b = relaxation factor for pitch solver

$\boldsymbol{\vec{\omega}}$ = angular velocity of propeller

CHAPTER 1

Introduction

1.1 Classical Lifting-Line Theory

Prandtl's classical lifting-line theory [1] provides the basis for much of finite wing theory as we understand it today. Prandtl hypothesized that the flow around a finite wing could be approximated by an infinite number of concentric horseshoe-shaped vortices with the bound portions placed on the quarter chord of the wing which form the lifting line. The trailing potions of the horseshoe vortices create a semi-infinite vortex sheet aligned with the freestream, and extend from the wing quarter chord aft to infinity. He then predicted that the lift from a spanwise section of the wing is equivalent to that of an infinite wing of the same airfoil shape and local angle of attack as the original spanwise section. He accomplished this by relating the Kutta-Joukowski theorem at a spanwise section of the wing with the two dimensional lift [2] given by

$$\rho V_\infty \Gamma = \frac{1}{2}\rho V_\infty^2 \widetilde{C}_L(\alpha)c \tag{1.1}$$

where ρ, V_∞, Γ, α, $\widetilde{C}_L$, and c are respectively the density of air, the freestream velocity, the vortex strength at the given spanwise section, the local angle of attack at the given spanwise section, the two dimensional lift coefficient of the airfoil at the given spanwise section which is a function of α, and the chord length at the given spanwise section. In order to obtain a tractable solution, Prandtl limited his analysis to a single wing with no sweep or dihedral. This allows the system of equations to be written in terms of a Fourier series, which can be readily solved [2].

1.2 Modern Numerical Lifting-Line Method

More recently, Phillips and Snyder [3] extended the work of Prandtl to include multiple wings of arbitrary sweep and dihedral, in the development of a modern numerical lifting-line algorithm. Prandtl's vortex sheet is replaced with a finite number of horseshoe vortices placed side by side and extend from the wing quarter chord aft in the direction of the freestream velocity. From the three-dimensional vortex lifting law, the differential force on a spanwise section of the wing can be computed from the bound vorticity by

$$\mathbf{d}\vec{\boldsymbol{F}} = \rho\Gamma\vec{\boldsymbol{V}} \times \mathbf{d}\vec{\boldsymbol{l}} \tag{1.2}$$

where $\mathbf{d}\vec{\boldsymbol{F}}$, $\vec{\boldsymbol{V}}$, and $\mathbf{d}\vec{\boldsymbol{l}}$ are respectively the aerodynamic force acting on the spanwise section, the local velocity at the spanwise section, and the directional vector for the bound vortex filament of the spanwise section. To solve for the bound vortex strength, the local section lift at any spanwise location is assumed to be equal to that of the lift produced by an identical airfoil section of an infinite wing at the same local angle of attack,

$$|\,\mathbf{d}\vec{\boldsymbol{F}}| = \frac{1}{2}\rho V_\infty^2 \widetilde{C}_L(\alpha)\,\mathrm{d}A \tag{1.3}$$

where $\widetilde{C}_L$ is again the two-dimensional lift coefficient of the airfoil at the spanwise section which is a function of the local angle of attack. The local induced angle of attack is given by

$$\alpha = \tan^{-1}\left(\frac{\vec{\boldsymbol{V}} \cdot \hat{\boldsymbol{u}}_{\boldsymbol{n}}}{\vec{\boldsymbol{V}} \cdot \hat{\boldsymbol{u}}_{\boldsymbol{a}}}\right) \tag{1.4}$$

with $\hat{\boldsymbol{u}}_{\boldsymbol{a}}$ and $\hat{\boldsymbol{u}}_{\boldsymbol{n}}$ being the axial and normal unit vectors respectively of the airfoil with respect to the chord line. Combining Eqs. (1.2) and (1.3), the fundamental relation used by Phillips and Snyder can be written as

$$2\left|\Gamma_i\left(\vec{\boldsymbol{V}}_{\boldsymbol{\infty}} + \sum_{j=1}^{N}\Gamma_j\vec{\boldsymbol{\nu}}_{\boldsymbol{ji}}\right) \times \mathbf{d}\vec{\boldsymbol{l}}_{\boldsymbol{i}}\right| - V_\infty^2\widetilde{C}_{Li}(\alpha_i)\,\mathrm{d}A_i = 0 \tag{1.5}$$

where $\Gamma_j\vec{\boldsymbol{\nu}}_{\boldsymbol{ji}}$ is the velocity induced by horseshoe vortex j on control point i, and N is the total number of control points, or spanwise sections, from all lifting surfaces being analyzed. Equation (1.5) is used to create a system of N nonlinear equations, which is readily solved using Newton's method to evaluate the vortex strengths at each spanwise segment.

This algorithm has shown to be accurate for wings of aspect ratio greater than about 4, and includes the effects of arbitrary sweep and dihedral. Furthermore, it can be used to model aerodynamic interactions between any number of lifting surfaces, and a method for approximating viscous effects was presented [4]. For the complete development of the algorithm for fixed-wing aircraft, see Phillips and Snyder [3]. As a side note, this modern numerical lifting-line method is sometimes incorrectly associated with vortex lattice methods, which apply an assumption from flat-plate theory to close the equations [5].

1.3 Blade Element Theory

Blade element theory (BET) is a more common and traditional method than lifting-line theory for analyzing a propeller. BET was first proposed by Drzewiecki [6] in 1900 to analyze propellers. BET has a similar assumption as lifting-line theory, that an airfoil section of the propeller blade has the same section lift as its two-dimensional counterpart. The difference with BET is that the induced velocity at a given radial distance r on the propeller blades is only a function of the airfoil sections at that radial point r. This means that induced velocity at a given section on the propeller blade is independent of propeller geometry farther out towards the tip or closer in towards the center of that given section. The propeller geometries are limited for this model, not allowing for swept blades, raked (dihedral) blades, or asymmetrically spaced blades. BET is presented in various textbooks and articles [6–11].

1.4 Current Propeller Codes

There is a wide range of analysis codes for propellers and wind turbines currently available, some of which are free. Many of these use computational fluid dynamics (CFD) as a means of analyzing the aerodynamics of the propeller. CFD codes are high fidelity models able to take into account almost all aspects of the aerodynamics including viscous effects, vortex roll up, slipstream contraction, and stall. Additionally these are able to handle virtually any propeller geometry. However, CFD codes can be difficult to implement correctly and are extremely computationally expensive compared to other methods.

Others codes use BET, momentum theory, or a combination of the two as their aerodynamic model [12]. BET and momentum theory are low fidelity models that are easier to implement and solve seemingly instantaneously on modern computers. These codes, however, cannot get accurate results for uncommon propeller geometries.

Another option are codes that implement vortex theory techniques which are mid-range fidelity models. There are several propeller codes that use various forms of vortex theory techniques. Some of these claim to use lifting-line theory, but either do so incorrectly [13] by violating the Helmholtz's vortex theorems of ending a vortex filament in the fluid and having a vortex filament of varying strength [2] or implement a vortex lattice technique [5]. The main difference between codes in this category are how they treat the slipstream wake/trailing vortices. The most common approaches are to use a prescribed wake model, or a free wake model.

1.4.1 Prescribed Wake

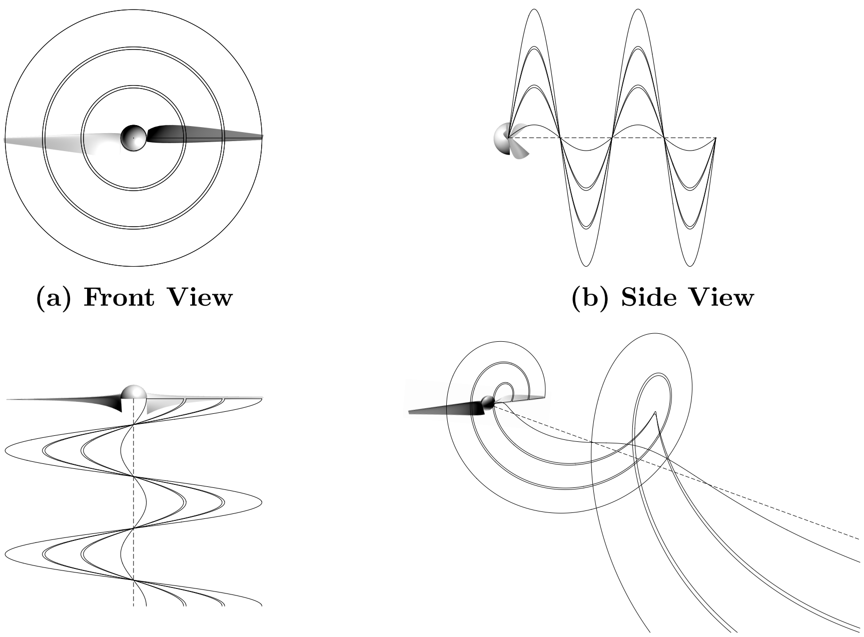
Most codes use the prescribed wake model, because it is simpler to implement and runs relatively quickly. This method forces the trailing vortices to some predetermined shape, usually a helix with constant pitch and radius. These codes also, generally discretize the trailing helical vortex filament into numerous straight segments in order to further simplify the analysis. These codes usually perform very well, giving a good balance between accuracy and code run time, but can run into issues at lower advance ratios.

1.4.2 Free Wake

Codes with free wake solutions generally discretize the trailing vortex filaments into small straight segments that can then be aligned with the local velocity. This is important because a trailing vortex cannot support a force, and if there is no force, then the vortex section must be parallel with the local fluid velocity. This is verified by Eq. (1.2) since in order to have $\mathbf{d}\vec{\boldsymbol{F}}$ equal to zero, then $\vec{\boldsymbol{V}} \times \mathbf{d}\vec{\boldsymbol{l}}$ must equal zero and that only happens when the two vectors are parallel. The benefits of free wake models are that they include the effects of induced velocity on trailing vortex filament geometry, which usually account for the strong axial flow within the wake of the propeller, the slipstream contraction, and vortex roll up of trailing vortices. This, in turn, alters the induced velocity on the propeller, thus turning this into an iterative method. Free wake methods are usually very computationally expensive, having to iterate on trailing vortex filament geometry and wake velocity flow field. However, they are still better than CFD solutions in terms of computational efficiency. An example of a code that implements this type of solution is CACTUS, by Sandia National Laboratories (SNL) [14], which is a wind turbine code but follows the same basic principles as those for a propeller code.

1.5 Applying Lifting-Line Theory to Propellers

The classical lifting-line theory does not directly apply to propellers due to the fact that propellers rotate relative to the freestream, where fixed wings do not. The main challenge of applying the classical lifting-line theory to a propeller flow field is that the flow field behind a propeller follows a helical motion as seen in Fig. 1.1. The fundamental mathematics



(a) Front View

(b) Side View

(c) Top View

(d) Isometric View

Fig. 1.1: Helical horseshoe vortices shown on one blade of a two bladed propeller

used in classical lifting-line theory to predict the induced velocity from the trailing vortex sheet on a finite wing applies the approximation that the trailing vortex sheet is planar and aligned with the free stream. Such an approximation cannot be used for propeller modeling and an alternative means of calculating the induced velocity from a helical vortex filament must be created. Note that Fig. 1.1 shows small gaps between adjacent horseshoe vortices. This is done as a visual aid only and there really should not be a gap between adjacent horseshoe vortices.

Another challenge with applying classical lifting-line theory to propellers, is handling high blade loadings which occur at low advance ratios. Issues arise in accurately describing the trailing vortex geometry at low advance ratios and a means of correcting them needs to be developed. These challenges are addressed in the following algorithm.

CHAPTER 2

Helical Wake

2.1 Helix Equations

2.1.1 Review of Straight Vortex Segments

Crucial to the propeller lifting-line method is the velocity induced by a semi-infinite helical vortex filament. In order to calculate this induced velocity, we must first review the induced velocity produced by a straight finite vortex filament as seen in Fig. 2.1. From Fig. 2.1 and the Biot-Savart Law, we have

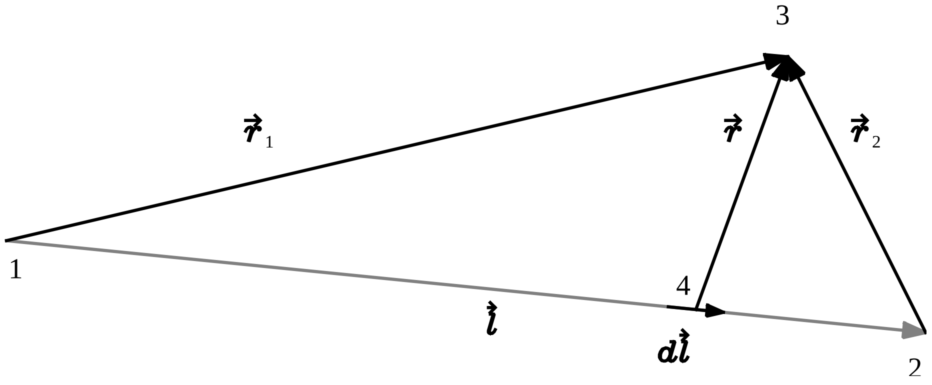


Fig. 2.1: Velocity induced by a straight vortex segment.

$$\mathbf{d}\vec{V} = \frac{\Gamma\,\mathbf{d}\vec{l} \times \vec{r}}{4\pi|\vec{r}|^3} \tag{2.1}$$

where $\mathbf{d}\vec{V}$ is the induced velocity at point 3 by the filament segment $\mathbf{d}\vec{l}$ and $\vec{r}$ is the vector from the filament segment at point 4 to point 3. Now let ζ be the distance from point 1 to point 4 divided by the distance from point 1 to point 2. [4] This gives

$$\mathbf{d}\vec{l} = \vec{l}\,\mathrm{d}\zeta \tag{2.2}$$

2.1.2 Helix Derivation

Equation (2.2) can be considered to be a line integral of some arbitrary curve h. For Fig. 2.1, h is a straight line between points 1 and 2. Using this, Eq. (2.2) can be rewritten as

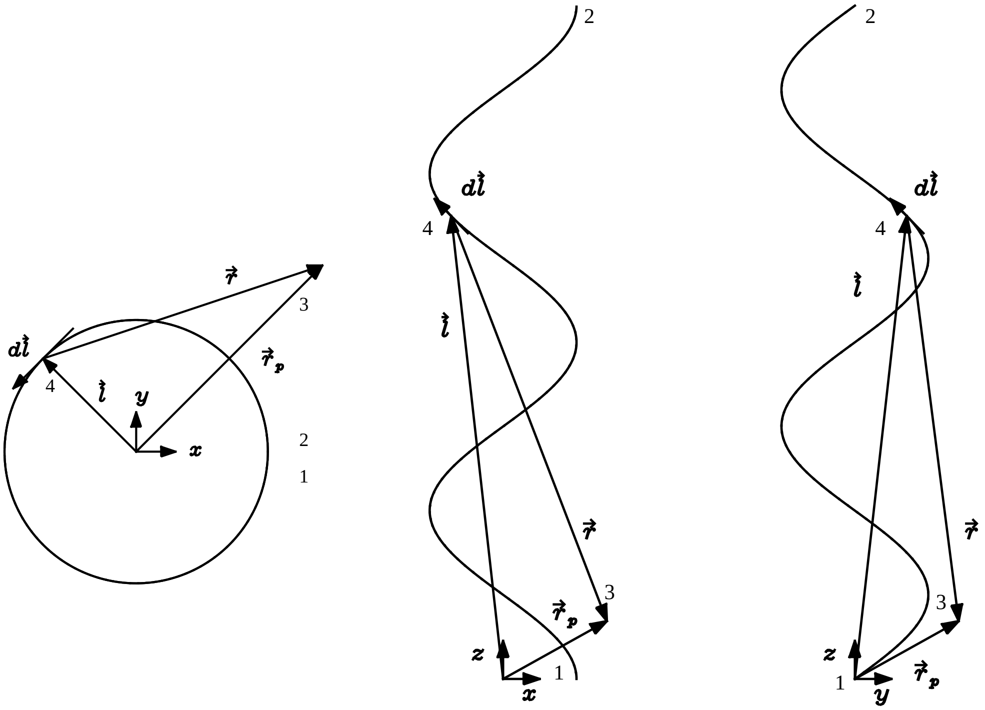
$$\begin{aligned}
\mathbf{d}\vec{\boldsymbol{l}} &= \vec{\boldsymbol{l}}\,\mathrm{d}\zeta \\
\mathbf{d}\vec{\boldsymbol{l}} &= |\vec{\boldsymbol{l}}|\hat{\boldsymbol{e}}_{\boldsymbol{l}}\,\mathrm{d}\zeta \\
\hat{\boldsymbol{e}}_{\boldsymbol{l}} &= \hat{\boldsymbol{e}}_{\boldsymbol{T}} \\
|\vec{\boldsymbol{l}}| &= \text{length of curve } h = s \\
\mathbf{d}\vec{\boldsymbol{l}} &= s\hat{\boldsymbol{e}}_{\boldsymbol{T}}\,\mathrm{d}\zeta
\end{aligned} \tag{2.3}$$

where $\hat{\boldsymbol{e}}_{\boldsymbol{l}}$ is the unit vector of $\vec{\boldsymbol{l}}$, $|\vec{\boldsymbol{l}}|$ is the magnitude of $\vec{\boldsymbol{l}}$, and $\hat{\boldsymbol{e}}_{\boldsymbol{T}}$ is a unit vector tangent at any point along the curve h. Now, lets say h is no longer a straight line, but a helix running from point 1 to point 2 as seen in Fig. 2.2. From Fig. 2.2, we can see that, $\hat{\boldsymbol{e}}_{\boldsymbol{l}} \neq \hat{\boldsymbol{e}}_{\boldsymbol{T}}$, $|\vec{\boldsymbol{l}}| \neq s$, and $\mathbf{d}\vec{\boldsymbol{l}} \neq \vec{\boldsymbol{l}}\,\mathrm{d}\zeta$ thus rendering Eq. (2.2) invalid for a helix and either Eq. (2.3) must be used or an alternative means of calculating $\mathbf{d}\vec{\boldsymbol{l}}$ must be derived.

The helix is parameterized by

$$\begin{aligned}
R(\zeta) &= a \\
\theta(\zeta) &= h_r 2\pi n\zeta \\
z(\zeta) &= bn\zeta
\end{aligned} \tag{2.4}$$

where a is the radius, n is the number of loops in the helix, b is the helix pitch or axial distance of one loop, ζ is the parameterization variable from Eq. (2.2) that can vary from 0 to 1, and h_r can equal 1 or -1 and determines the direction of helix rotation. Rewriting the helix in Cartesian coordinates and noting the symmetry of trigonometric functions, namely



(a) Top View **(b) Front View** **(c) Side View**

Fig. 2.2: Helix Vortex Filament

$\cos(-\phi) = \cos(\phi)$ and $\sin(-\phi) = -\sin(\phi)$, we can rewrite Eq. (2.4) as

$$\begin{aligned} x_h(\zeta) &= a\cos\theta = a\cos\theta_{rh} \\ y_h(\zeta) &= a\sin\theta = h_r a\sin\theta_{rh} \\ z_h(\zeta) &= bn\zeta \end{aligned} \tag{2.5}$$

where $\theta_{rh} = 2\pi n\zeta$ and is the right-handed rotation of the helix. Taking the derivative of Eq. (2.5) gives $\mathbf{d}\vec{\boldsymbol{l}}$ as

$$\mathbf{d}\vec{\boldsymbol{l}} = -2\pi an\sin\theta_{rh}\,\mathrm{d}\zeta\hat{\boldsymbol{\imath}} + h_r 2\pi an\cos\theta_{rh}\,\mathrm{d}\zeta\hat{\boldsymbol{\jmath}} + bn\,\mathrm{d}\zeta\hat{\boldsymbol{k}} \tag{2.6}$$

Having solved for $\mathbf{d}\vec{\boldsymbol{l}}$, we now turn our attention to $\vec{\boldsymbol{r}}$. From Fig. 2.2 we have $\vec{\boldsymbol{r}}_{\boldsymbol{p}} = \vec{\boldsymbol{l}} + \vec{\boldsymbol{r}}$ which leads to

$$\vec{\boldsymbol{r}} = \vec{\boldsymbol{r}}_{\boldsymbol{p}} - \vec{\boldsymbol{l}} \tag{2.7}$$

with $\vec{\boldsymbol{r}}_{\boldsymbol{p}}$ that is known and $\vec{\boldsymbol{l}}$ which is a function of ζ, given by Eq. (2.5). This gives

$$\vec{\boldsymbol{r}} = (x_p - a\cos\theta_{rh})\hat{\boldsymbol{\imath}} + (y_p - h_r a\sin\theta_{rh})\hat{\boldsymbol{\jmath}} + (z_p - bn\zeta)\hat{\boldsymbol{k}} \tag{2.8}$$

where x_p, y_p, and z_p represent point 3 from Fig. 2.2 where we are solving for the induced velocity. It then follows that the magnitude cubed of $\vec{\boldsymbol{r}}$ is

$$|\vec{\boldsymbol{r}}|^3 = \big((x_p - a\cos\theta_{rh})^2 + (y_p - h_r a\sin\theta_{rh})^2 + (z_p - bn\zeta)^2\big)^{\frac{3}{2}} \tag{2.9}$$

Using Eq. (2.6) and Eq. (2.8) and taking the cross product gives

$$\begin{aligned}\mathbf{d}\vec{\boldsymbol{l}} \times \vec{\boldsymbol{r}} = {} & \big(h_r 2\pi an\cos\theta_{rh}(z_p - bn\zeta) - bn(y_p - h_r a\sin\theta_{rh})\big)\hat{\boldsymbol{\imath}}\,\mathrm{d}\zeta \\ & + \big(bn(x_p - a\cos\theta_{rh}) + 2\pi an\sin\theta_{rh}(z_p - bn\zeta)\big)\hat{\boldsymbol{\jmath}}\,\mathrm{d}\zeta \\ & + \big(-2\pi an\sin\theta_{rh}(y_p - h_r a\sin\theta_{rh}) - h_r 2\pi an\cos\theta_{rh}(x_p - a\cos\theta_{rh})\big)\hat{\boldsymbol{k}}\,\mathrm{d}\zeta\end{aligned} \tag{2.10}$$

Simplifying Eq. (2.10) becomes

$$\begin{aligned}\mathbf{d}\vec{\boldsymbol{l}} \times \vec{\boldsymbol{r}} = {} & \big(-bny_p + h_r abn\sin\theta_{rh} + h_r 2\pi anz_p\cos\theta)rh - h_r 2\pi abn^2\zeta\cos\theta_{rh}\big)\hat{\boldsymbol{\imath}}\,\mathrm{d}\zeta \\ & + \big(bnx_p - abn\cos\theta_{rh} + 2\pi anz_p\sin\theta_{rh} - 2\pi abn^2\zeta\sin\theta_{rh}\big)\hat{\boldsymbol{\jmath}}\,\mathrm{d}\zeta \\ & + 2\pi an\big(h_r a - h_r x_p\cos\theta_{rh} - y_p\sin\theta_{rh}\big)\hat{\boldsymbol{k}}\,\mathrm{d}\zeta\end{aligned} \tag{2.11}$$

Having solved everything in terms of ζ, Eq. (2.1) can be found by substituting in Eq. (2.9) and Eq. (2.11).

$$\mathbf{d}\vec{\boldsymbol{V}} = \begin{cases} \dfrac{\Gamma n}{4\pi}\dfrac{-by_p + h_r ab\sin\theta_{rh} + h_r 2\pi az_p\cos\theta_{rh} - h_r 2\pi abn\zeta\cos\theta_{rh}}{\big((x_p - a\cos\theta_{rh})^2 + (y_p - h_r a\sin\theta_{rh})^2 + (z_p - bn\zeta)^2\big)^{\frac{3}{2}}}\,\mathrm{d}\zeta\ \hat{\boldsymbol{\imath}} \\ +\dfrac{\Gamma n}{4\pi}\dfrac{bx_p - ab\cos\theta_{rh} + 2\pi az_p\sin\theta_{rh} - 2\pi abn\zeta\sin\theta_{rh}}{\big((x_p - a\cos\theta_{rh})^2 + (y_p - h_r a\sin\theta_{rh})^2 + (z_p - bn\zeta)^2\big)^{\frac{3}{2}}}\,\mathrm{d}\zeta\ \hat{\boldsymbol{\jmath}} \\ +\dfrac{\Gamma an}{2}\dfrac{h_r a - h_r x_p\cos\theta_{rh} - y_p\sin\theta_{rh}}{\big((x_p - a\cos\theta_{rh})^2 + (y_p - h_r a\sin\theta_{rh})^2 + (z_p - bn\zeta)^2\big)^{\frac{3}{2}}}\,\mathrm{d}\zeta\ \hat{\boldsymbol{k}} \end{cases} \tag{2.12}$$

Then, the induced velocities from the helical finite vortex filament can be found by integrating which yields

$$\vec{\boldsymbol{V}} = \begin{cases} \dfrac{\Gamma n}{4\pi} \displaystyle\int_0^1 \dfrac{-by_p + h_r ab\sin\theta_{rh} + h_r 2\pi a z_p \cos\theta_{rh} - h_r 2\pi abn\zeta\cos\theta_{rh}}{\left((x_p - a\cos\theta_{rh})^2 + (y_p - h_r a\sin\theta_{rh})^2 + (z_p - bn\zeta)^2\right)^{\frac{3}{2}}}\,\mathrm{d}\zeta\ \hat{\boldsymbol{\imath}} \\ + \dfrac{\Gamma n}{4\pi} \displaystyle\int_0^1 \dfrac{bx_p - ab\cos\theta_{rh} + 2\pi a z_p \sin\theta_{rh} - 2\pi abn\zeta\sin\theta_{rh}}{\left((x_p - a\cos\theta_{rh})^2 + (y_p - h_r a\sin\theta_{rh})^2 + (z_p - bn\zeta)^2\right)^{\frac{3}{2}}}\,\mathrm{d}\zeta\ \hat{\boldsymbol{\jmath}} \\ + \dfrac{\Gamma an}{2} \displaystyle\int_0^1 \dfrac{h_r a - h_r x_p\cos\theta_{rh} - y_p\sin\theta_{rh}}{\left((x_p - a\cos\theta_{rh})^2 + (y_p - h_r a\sin\theta_{rh})^2 + (z_p - bn\zeta)^2\right)^{\frac{3}{2}}}\,\mathrm{d}\zeta\ \hat{\boldsymbol{k}} \end{cases} \tag{2.13}$$

remembering that θ_{rh} is a function of ζ. As of yet, an analytical solution to these integrals have not been found and they should be integrated numerically.

Nondimensionalizing Eq. (2.13) can help to generalize the problem and help to see patterns. I chose to nondimensionalize most parameters using the helix radius a. The pitch of the helix is nondimensionalized by $\lambda = \frac{b}{a}$, where λ is the nondimensional pitch. The induced velocity is nondimensionalized by $\vec{\boldsymbol{\sigma}} = \frac{\vec{\boldsymbol{V}}a}{\Gamma}$, where $\vec{\boldsymbol{\sigma}}$ is the nondimensional velocity. Implementing this gives

$$\vec{\boldsymbol{\sigma}} = \begin{cases} \dfrac{n}{4\pi} \displaystyle\int_0^1 \dfrac{-\lambda\hat{y}_p + h_r\lambda\sin\theta_{rh} + h_r 2\pi\hat{z}_p\cos\theta_{rh} - h_r 2\pi\lambda n\zeta\cos\theta_{rh}}{\left((\hat{x}_p - \cos\theta_{rh})^2 + (\hat{y}_p - h_r\sin\theta_{rh})^2 + (\hat{z}_p - \lambda n\zeta)^2\right)^{\frac{3}{2}}}\,\mathrm{d}\zeta\ \hat{\boldsymbol{\imath}} \\ + \dfrac{n}{4\pi} \displaystyle\int_0^1 \dfrac{\lambda\hat{x}_p - \lambda\cos\theta_{rh} + 2\pi\hat{z}_p\sin\theta_{rh} - 2\pi\lambda n\zeta\sin\theta_{rh}}{\left((\hat{x}_p - \cos\theta_{rh})^2 + (\hat{y}_p - h_r\sin\theta_{rh})^2 + (\hat{z}_p - \lambda n\zeta)^2\right)^{\frac{3}{2}}}\,\mathrm{d}\zeta\ \hat{\boldsymbol{\jmath}} \\ + \dfrac{n}{2} \displaystyle\int_0^1 \dfrac{h_r - h_r\hat{x}_p\cos\theta_{rh} - \hat{y}_p\sin\theta_{rh}}{\left((\hat{x}_p - \cos\theta_{rh})^2 + (\hat{y}_p - h_r\sin\theta_{rh})^2 + (\hat{z}_p - \lambda n\zeta)^2\right)^{\frac{3}{2}}}\,\mathrm{d}\zeta\ \hat{\boldsymbol{k}} \end{cases} \tag{2.14}$$

where $\hat{x}_p = \frac{x_p}{a}$, $\hat{y}_p = \frac{y_p}{a}$, and $\hat{z}_p = \frac{z_p}{a}$ are the nondimensional control points for the location of the induced velocity.

An important parameter here is λ, which is a measure the 'helix stretch', where a large λ signifies a helix that is really long and looks as though it has been stretched out, and a low λ looks like a tightly wound coil, or a compressed spring. At the extreme cases where

$\lambda = \infty$, the helix becomes a straight line, and where $\lambda = 0$, the helix becomes n circles all overlaid over one another.

2.1.3 Numerical Integration

Caution should be used with the numerical integration, since several factors will determine the effectiveness of the integration including: the number of loops in the helix n, the general shape of the helix a and b, and the location of the control point x_p, y_p, and z_p. Different methods are shown here to aid the user.

Fixed Step Size

A convenient way to determine the fixed step size $\mathrm{d}\zeta$ is with

$$\mathrm{d}\zeta = \frac{1}{nm} \tag{2.15}$$

where m is the desired number of steps in a single loop. The value 1 in the numerator, is the total length of the ζ variable, which goes from 0 to 1. The denominator contains the number of loops in the helix times the number of steps per loop which equals the total number of steps in the helix. The total number of steps should always be an integer value for the integration to work properly. So $\mathrm{d}\zeta$ is equal to the length of ζ per step, which is exactly what we want. Using Eq. (2.15) one can change the number of loops in the helix without worrying about reducing the precision of the step size. The user can then use whichever numerical integration technique to solve for the induced velocity.

Variable Step Size

Since the induced velocity in Eq. (2.13) is inversely proportional to $|\vec{r}|^2$, it may be advantageous to introduce a variable step size so that there can be high precision near the control point and less so farther away. This becomes more essential when n is particularly large. If the control point is within the propeller plane, $z_p = 0$, a simple power clustering method can be used. The first step of this method is to determine the total number of steps, k, to be used in the integration. This can either be chosen directly by the user, or it can be determined by

$$k = nm \tag{2.16}$$

where m in this case is the average number of steps per loop. When using this equation, the user must ensure that k is an integer value for the method to work properly, just like for the fixed step size case. Then an equivalent fixed step size can be calculated by

$$\mathrm{d}\zeta_f = \frac{1}{k} \tag{2.17}$$

Then, equivalent ζ values can be calculated using the equivalent fixed step size by

$$\zeta_{f_i} = i \ \mathrm{d}\zeta_f \qquad\qquad 0 \leq i \leq k \tag{2.18}$$

The variable step sizes and ζ values can then be calculated by

$$\begin{aligned} \zeta_{v_i} &= (\zeta_{f_i})^p & p > 0 \quad 0 \leq i \leq k \\ \mathrm{d}\zeta_{v_i} &= \zeta_{v_i} - \zeta_{v_{i-1}} & 1 \leq i \leq k \end{aligned} \tag{2.19}$$

where p is a value input by the user to determine the degree of clustering. The greater the p, the more potent the clustering towards the propeller plane. Note, that when $p = 1$ the variable step size case reduces down to the fixed step size case. Also, $p < 1$ will cluster away from the propeller plane, and is almost always bad for the integration. After having solved for ζ_{v_i} and $\mathrm{d}\zeta_{v_i}$ the user can use any numerical integration technique to solve for the induced velocity, as long as that technique works with a variable step size.

2.1.4 Semi-Infinite Helical Vortex Filament

Up to this point the number of loops in the helix has been arbitrary and represented by the variable n. In order to get a semi-infinite helix, we need to take the limit as n approaches ∞. This is difficult to do without an analytical solution for Eq. (2.13). Therefore, one can approximate the semi-infinite helical vortex filament by a finite helix with large n. This should work, since the induced differential velocity from a given differential filament section is inversely proportional to $|\vec{\boldsymbol{r}}|^2$. So the induced differential velocity from a differential filament section will quickly drop to a negligible value as $|\vec{\boldsymbol{r}}|$ increases. An appropriate value for n could then be calculated, where a bigger n won't change the induced velocity past a certain tolerance range.

Number of Loops

In order to approximate a finite helix as a semi-infinite helix, we need to insure that there are enough loops in the helix to the point that each differential velocity element from the numerical integration is small enough to be ignored. To start this we must go back to Eq. (2.1).

$$\mathbf{d}\vec{\boldsymbol{V}} = \frac{\Gamma\,\mathbf{d}\vec{\boldsymbol{l}} \times \vec{\boldsymbol{r}}}{4\pi|\vec{\boldsymbol{r}}|^3} \tag{2.20}$$

Then we analyze the cross product and note that

$$\mathbf{d}\vec{\boldsymbol{l}} \times \vec{\boldsymbol{r}} = |\,\mathbf{d}\vec{\boldsymbol{l}}|\,|\vec{\boldsymbol{r}}|\sin\Lambda\,\hat{\boldsymbol{e}}_{\boldsymbol{v}} \tag{2.21}$$

where Λ is the angle between the two vectors and $\hat{\boldsymbol{e}}_{\boldsymbol{v}}$ is the unit vector of the induced velocity. Substituting this in, and noting that we are only interested in the magnitude of the induced velocity we can leave off the unit vector, resulting in

$$\mathrm{d}V = \frac{\Gamma}{4\pi}\frac{|\,\mathbf{d}\vec{\boldsymbol{l}}|\,\sin\Lambda}{|\vec{\boldsymbol{r}}|^2} \tag{2.22}$$

Now we solve $|\,\mathbf{d}\vec{\boldsymbol{l}}|$ to be

$$|\,\mathbf{d}\vec{\boldsymbol{l}}| = \sqrt{(2\pi an\;d\zeta)^2 + (bn\;d\zeta)^2} = \sqrt{4\pi^2a^2 + b^2}n\;d\zeta \tag{2.23}$$

$$b = \lambda a \tag{2.24}$$

$$|\,\mathbf{d}\vec{\boldsymbol{l}}| = \sqrt{4\pi^2 + \lambda^2}an\;d\zeta \tag{2.25}$$

Substituting this back in we get

$$\mathrm{d}V = \frac{\Gamma}{4\pi}\frac{\sqrt{4\pi^2+\lambda^2}an\sin\Lambda d\zeta}{|\vec{\boldsymbol{r}}|^2} \tag{2.26}$$

Next comes $|\vec{\boldsymbol{r}}|^2$. This is a little bit more difficult because $\vec{\boldsymbol{r}}$ depends on the control point, which is completely arbitrary. To be conservative, we will choose a control point that will minimize $|\vec{\boldsymbol{r}}|$, since that will maximize $\mathrm{d}V$. From Eq. (2.9) and using the fact that $\zeta = 1$ for the far end point of the helix, we can easily see that

$$|\vec{\boldsymbol{r}}|^2 = \big(x_p - a\cos(2\pi n)\big)^2 + \big(y_p - h_r a\sin(2\pi n)\big)^2 + \big(z_p - bn\big)^2 \tag{2.27}$$

Since we are interested in induced velocities on propeller blades, we will restrict our control point to be within the propeller plane, or XY plane as seen from Fig. 2.2, therefore $z_p = 0$. To minimize $|\vec{\boldsymbol{r}}|$ the control point should be such that $\vec{\boldsymbol{r}}$ only has a z component. This implies that $x_p = a\cos(2\pi n)$ and $y_p = h_r a\sin(2\pi n)$. This yields

$$|\vec{\boldsymbol{r}}|^2 = b^2n^2 \tag{2.28}$$

This can easily be verified by choosing a point and helix geometry that meets this criteria, like $(a, 0, 0)$ and n an integer. Now we can substitute this back into our equation to yield

$$\mathrm{d}V = \frac{\Gamma}{4\pi}\frac{\sqrt{4\pi^2+\lambda^2}a\sin\Lambda d\zeta}{nb^2} \tag{2.29}$$

We are now left with solving for Λ. Using Fig. 2.2 as a guide and remembering that $\vec{\boldsymbol{r}}$ will point straight down in the negative z direction, $\Lambda = \pi - \psi$, where ψ is the pitch angle of the helix. This gives $\sin\Lambda = \sin(\pi-\psi) = \sin\psi$ and from the helix geometry we know that

$$\sin\psi = \frac{2\pi an}{s} \tag{2.30}$$

$$s = \sqrt{(2\pi an)^2 + (bn)^2} = \sqrt{4\pi^2+\lambda^2}an \tag{2.31}$$

where s is the arc length of the helix. This gives

$$\sin\psi = \frac{2\pi}{\sqrt{4\pi^2 + \lambda^2}} \tag{2.32}$$

and substituting this back into our velocity equation and remembering the nondimensional terms, yields

$$\mathrm{d}V = \frac{\Gamma a \,\mathrm{d}\zeta}{2b^2 n} \qquad \mathrm{d}\sigma = \frac{\mathrm{d}\zeta}{2\lambda^2 n} \tag{2.33}$$

for the dimensional and nondimensional case. Substituting the value for $\mathrm{d}\zeta$ gives

$$\mathrm{d}V = \frac{\Gamma a}{2b^2 n^2 m} \qquad \mathrm{d}\sigma = \frac{1}{2\lambda^2 n^2 m} \tag{2.34}$$

Then setting a tolerance value as, $\tau = \mathrm{d}V m$ or $\tau = \mathrm{d}\sigma\, m$ which can be thought of as the amount of velocity induced by adding another loop in the helix, and solving for n gives

$$n = \sqrt{\frac{\Gamma a}{2b^2 \tau}} \qquad n = \sqrt{\frac{1}{2\lambda^2 \tau}} \tag{2.35}$$

Equation (2.35) is meant as a guide in determining the number of loops needed and usually over predicts what is needed.

2.2 Verification

2.2.1 Helix Approximation Method

Another way to calculate the induced velocity from a helical vortex filament is to approximate the helix geometry with several small straight vortex segments. From the numerical lifting-line method presented by Phillips and Snyder, [3] we know that the induced velocity from a straight vortex segment is

$$\vec{\boldsymbol{V}} = \frac{\Gamma}{4\pi} \frac{(r_1 + r_2)(\vec{\boldsymbol{r}}_{\boldsymbol{1}} \times \vec{\boldsymbol{r}}_{\boldsymbol{2}})}{r_1 r_2 (r_1 r_2 + \vec{\boldsymbol{r}}_{\boldsymbol{1}} \cdot \vec{\boldsymbol{r}}_{\boldsymbol{2}})} \tag{2.36}$$

where $\vec{\boldsymbol{r}}_{\mathbf{1}}$ and $\vec{\boldsymbol{r}}_{\mathbf{2}}$ are the same as in Fig. 2.1 and r_1 and r_2 are the magnitudes of their respective vectors. If the helix geometry is composed of several small straight segments, then the approximate induced velocity from the entire helix is

$$\vec{\boldsymbol{V}} = \frac{\Gamma}{4\pi} \sum_{i=1}^{k} \frac{(r_{1i} + r_{2i})(\vec{\boldsymbol{r}}_{\boldsymbol{1i}} \times \vec{\boldsymbol{r}}_{\boldsymbol{2i}})}{r_{1i} r_{2i} (r_{1i} r_{2i} + \vec{\boldsymbol{r}}_{\boldsymbol{1i}} \cdot \vec{\boldsymbol{r}}_{\boldsymbol{2i}})} \tag{2.37}$$

where the subscript i represents a corresponding straight segment, and k is from Eq. (2.16) and is the total number of segments in the helix. The position and size of the straight segments can be evaluated using either Eq. (2.18) or Eq. (2.19) and Eq. (2.5). For a given helix geometry and control point location, Eq. (2.13) and Eq. (2.37) converge to the same values as the number of steps/segments are increased. The nondimensional version of Eq. (2.37) is

$$\vec{\boldsymbol{\sigma}} = \frac{1}{4\pi} \sum_{i=1}^{k} \frac{(\hat{r}_{1i} + \hat{r}_{2i})(\vec{\hat{\boldsymbol{r}}}_{\boldsymbol{1i}} \times \vec{\hat{\boldsymbol{r}}}_{\boldsymbol{2i}}}{\hat{r}_{1i} \hat{r}_{2i} (\hat{r}_{1i} \hat{r}_{2i} + \vec{\hat{\boldsymbol{r}}}_{\boldsymbol{1i}} \cdot \vec{\hat{\boldsymbol{r}}}_{\boldsymbol{2i}}} \tag{2.38}$$

where $r_{1i} = \hat{r}_{1i} a$, $r_{2i} = \hat{r}_{2i} a$, $\vec{\boldsymbol{r}}_{\boldsymbol{1i}} = \vec{\hat{\boldsymbol{r}}}_{\boldsymbol{1i}} a$, $\vec{\boldsymbol{r}}_{\boldsymbol{2i}} = \vec{\hat{\boldsymbol{r}}}_{\boldsymbol{2i}} a$, and remembering that $\vec{\boldsymbol{\sigma}} = \vec{\boldsymbol{V}} \frac{a}{\Gamma}$.

2.2.2 Vortex Ring Comparison

In order to see how well these two methods work, it is necessary to compare them to an analytical solution. One analytical solution comes from the vortex ring. If we set the pitch to be, $b = 0$, and the number of loops to be, $n = 1$, then we are left with a circular vortex filament. Now, if we also fix the control point in the center of the ring, $(0, 0, 0)$, then we can derive an analytical solution. Starting with Eq. (2.1)

$$\mathbf{d}\vec{\boldsymbol{V}} = \frac{\Gamma\, \mathbf{d}\vec{\boldsymbol{l}} \times \vec{\boldsymbol{r}}}{4\pi |\vec{\boldsymbol{r}}|^3} \tag{2.39}$$

then noting that $|\vec{\boldsymbol{r}}|$ is constant and is $|\vec{\boldsymbol{r}}| = a$ then the equation becomes

$$\mathbf{d}\vec{\boldsymbol{V}} = \frac{\Gamma\, \mathbf{d}\vec{\boldsymbol{l}} \times \vec{\boldsymbol{r}}}{4\pi a^3} \tag{2.40}$$

Next, computing the cross product gives

$$\mathbf{d}\vec{\boldsymbol{V}} = \frac{\Gamma|\,\mathbf{d}\vec{\boldsymbol{l}}||\vec{\boldsymbol{r}}|\sin\Lambda\hat{\boldsymbol{e}}_{\boldsymbol{v}}}{4\pi a^3} \tag{2.41}$$

where Λ is the angle between the two vectors and $\hat{\boldsymbol{e}}_{\boldsymbol{v}}$ is the unit vector in the direction of the induced velocity. Because $\mathbf{d}\vec{\boldsymbol{l}}$ is always tangent to the circle and $\vec{\boldsymbol{r}}$ is always normal to the circle, pointing towards the center, then $\Lambda = \frac{\pi}{2}$. Also, since both $\mathbf{d}\vec{\boldsymbol{l}}$ and $\vec{\boldsymbol{r}}$ are always in the XY plane, and due to the way $\mathbf{d}\vec{\boldsymbol{l}}$ points along the circle, $\hat{\boldsymbol{e}}_{\boldsymbol{v}} = \hat{\boldsymbol{k}}$, thus for a positive Γ the induced velocity is in the positive z direction. Then, noting, that $|\,\mathbf{d}\vec{\boldsymbol{l}}| = 2\pi a\,\mathrm{d}\zeta$ and implementing all this into the equation gives the analytical solution for the center point of a vortex ring,

$$\begin{aligned}
\mathbf{d}\vec{\boldsymbol{V}} &= \frac{\Gamma\hat{\boldsymbol{k}}d\zeta}{2a}\\
\vec{\boldsymbol{V}} &= \int_0^1 \frac{\Gamma\hat{\boldsymbol{k}}}{2a}d\zeta\\
\vec{\boldsymbol{V}} &= \frac{\Gamma}{2a}\hat{\boldsymbol{k}}
\end{aligned} \tag{2.42}$$

Now, the results of Eq. (2.37) and Eq. (2.13) can be compared against the analytic solution in Eq. (2.42). Figure 2.3 shows these results for the axial component of velocity, where the numeric integration used was the trapezoidal rule. Both methods used a constant step/segment length. The x and y velocities computed from Eq. (2.37) give 0% error and don't show up on a loglog plot, regardless of the number of segments in the ring. The velocities computed from Eq. (2.13) did the same, 0% error regardless of the number of steps used. This is because the helix function, Eq. (2.13), simplifies down to Eq. (2.42) given the conditions for the vortex ring, namely $b = x_p = y_p = z_p = 0$ and $h_r = n = 1$.

2.2.3 Centerline Comparison

In order to verify the accuracy of the two methods further, the analytical solution for the axial velocity on the centerline of the helix is derived. Starting with Eq. (2.13) and

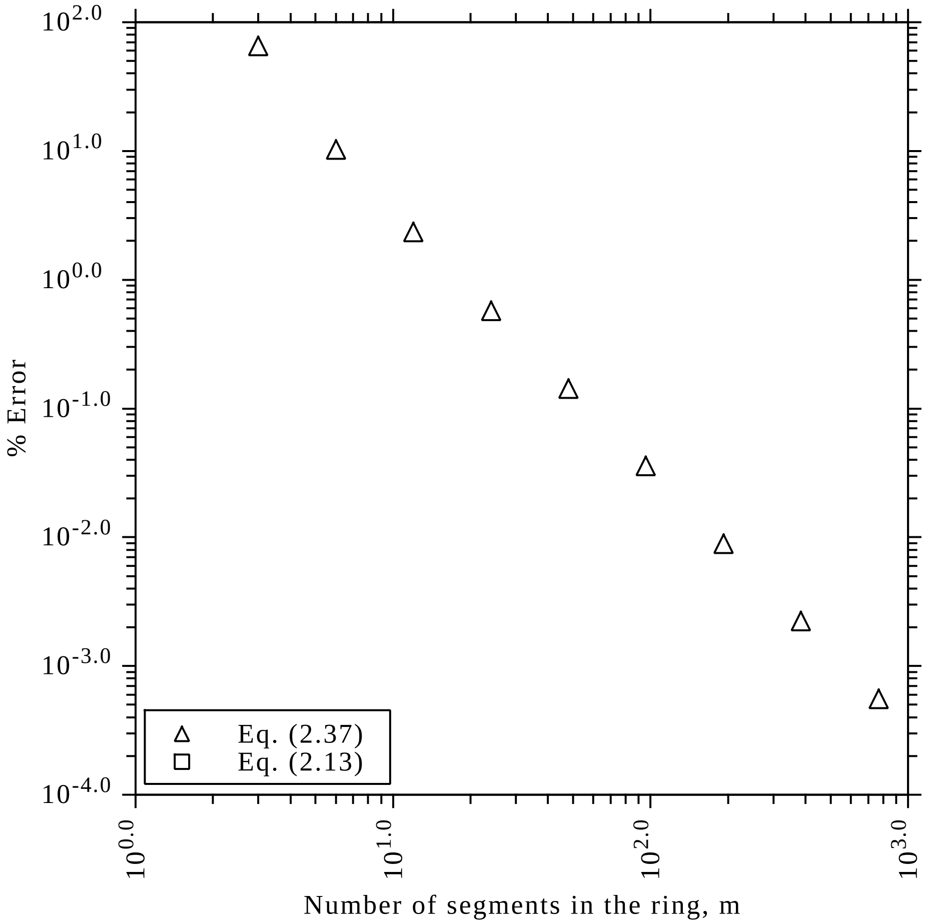


Fig. 2.3: Comparison of Eq. (2.37) and Eq. (2.13) against the analytic solution of the vortex ring, Eq. (2.42). The trapezoidal rule was used for Eq. (2.13).

setting $x_p = y_p = 0$ gives

$$\vec{\boldsymbol{V}} = \begin{cases} \dfrac{\Gamma n}{4\pi} \displaystyle\int_0^1 \dfrac{h_r ab \sin\theta_{rh} + h_r 2\pi a z_p \cos\theta_{rh} - h_r 2\pi abn\zeta \cos\theta_{rh}}{\left(a^2 + (z_p - bn\zeta)^2\right)^{\frac{3}{2}}} \,\mathrm{d}\zeta \; \hat{\boldsymbol{\imath}} \\ + \dfrac{\Gamma n}{4\pi} \displaystyle\int_0^1 \dfrac{ab \cos\theta_{rh} + 2\pi a z_p \sin\theta_{rh} - 2\pi abn\zeta \sin\theta_{rh}}{\left(a^2 + (z_p - bn\zeta)^2\right)^{\frac{3}{2}}} \,\mathrm{d}\zeta \; \hat{\boldsymbol{\jmath}} \\ + \dfrac{h_r \Gamma a^2 n}{2} \displaystyle\int_0^1 \dfrac{1}{\left(a^2 + (z_p - bn\zeta)^2\right)^{\frac{3}{2}}} \,\mathrm{d}\zeta \; \hat{\boldsymbol{k}} \end{cases} \tag{2.43}$$

The x and y components of Eq. (2.43) still look fairly involved and I haven't yet been able to integrate them. The z component on the other hand has lost all ζ terms in the numerator.

Integrating gives

$$V_z = \frac{h_r \Gamma z_p}{2b\sqrt{a^2 + z_p^2}} - \frac{h_r \Gamma (z_p - bn)}{2b\sqrt{a^2 + (z_p - bn)^2}} \tag{2.44}$$

Equation (2.44) is the analytical solution for the axial velocity along the centerline of a finite helix. The nondimensional version of Eq. (2.44) is

$$\sigma_z = \frac{\hat{z}_p}{2\lambda\sqrt{1 + \hat{z}_p^2}} - \frac{\hat{z}_p - \lambda n}{2\lambda\sqrt{1 + (\hat{z}_p - \lambda n)^2}} \tag{2.45}$$

The results from Eq. (2.38) and Eq. (2.14) for the axial component can be compared against the analytic solution Eq. (2.45). The results are given in Figs. 2.4-2.7 with a setup as follows: 5 loops in the helix, a control point of $(0, 0, 0)$, and various values for λ.

Figures 2.4 and 2.5 show results using a uniform spacing for both methods and where Eq. (2.14) uses the trapezoidal rule and 4$^{\text{th}}$ order Runge-Kutta integration techniques respectively. First thing of note about these figures, is that for $\lambda \geq 1$ the % error doesn't decrease linearly on the loglog plot as expected. This happens because the control point is held at $\hat{z}_p = 0$ while λ is large, meaning the helix pitch is large and only a small portion of the helix is close to the control point. Recall from Eq. (2.1) that $\mathbf{d}\vec{\boldsymbol{V}}$ is inversely proportional to $|\vec{\boldsymbol{r}}|^2$, meaning the parts of the helix closer to the control point have greater effect than those farther away. This causes increased error in the two methods if there is insufficient number of segments to capture the beginning parts of the helix properly. The higher the λ, the more segments are required to eliminate this phenomenon.

Although, this can be remedied by clustering the segments closer to the control point. Figures 2.6 and 2.7 repeat the same setup as before except the points are clustered using a power clustering factor of 3 as explained in section 2.1.3.

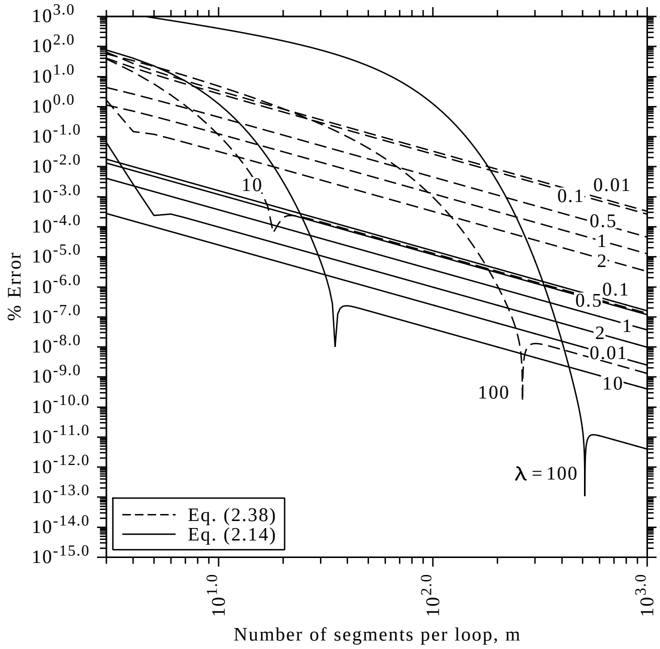


Fig. 2.4: Comparison of Eq. (2.38) and Eq. (2.14) for the z component at location (0,0,0) against the analytical solution, Eq. (2.45) for, 5 loops, various values of λ, using uniform spacing for both methods, and the trapezoidal rule for Eq. (2.14).

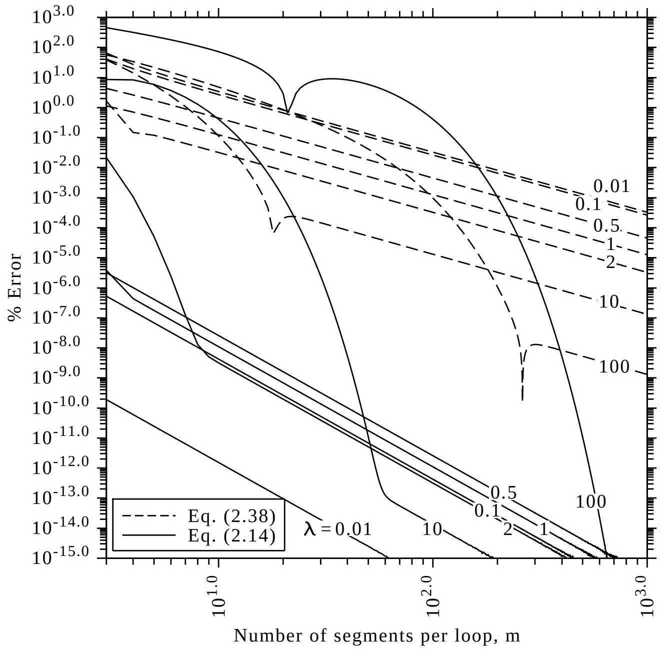


Fig. 2.5: Comparison of Eq. (2.38) and Eq. (2.14) for the z component at location (0,0,0) against the analytical solution, Eq. (2.45) for, 5 loops, various values of λ, using uniform spacing for both methods, and 4th order Runge-Kutta for Eq. (2.14).

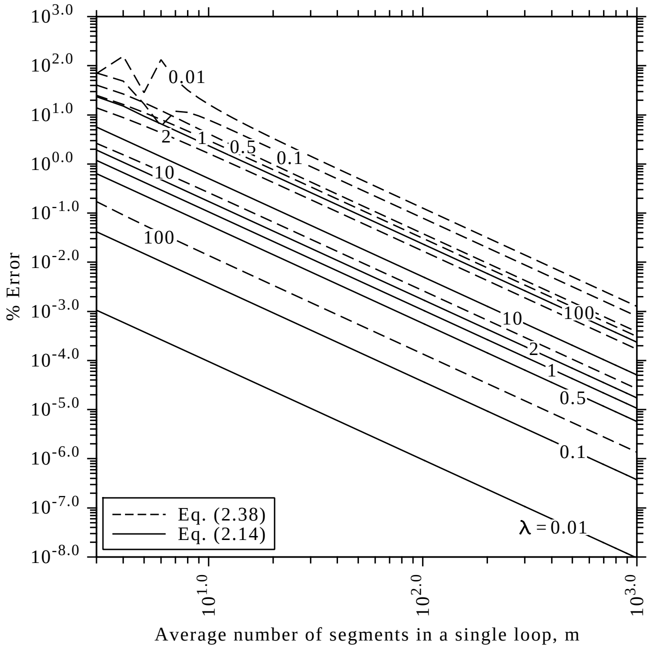


Fig. 2.6: Comparison of Eq. (2.38) and Eq. (2.14) for the z component at location (0,0,0) against the analytical solution, Eq. (2.45) for, 5 loops, various values of λ, using a power clustering factor of 3 for both methods, and the trapezoidal rule for Eq. (2.14).

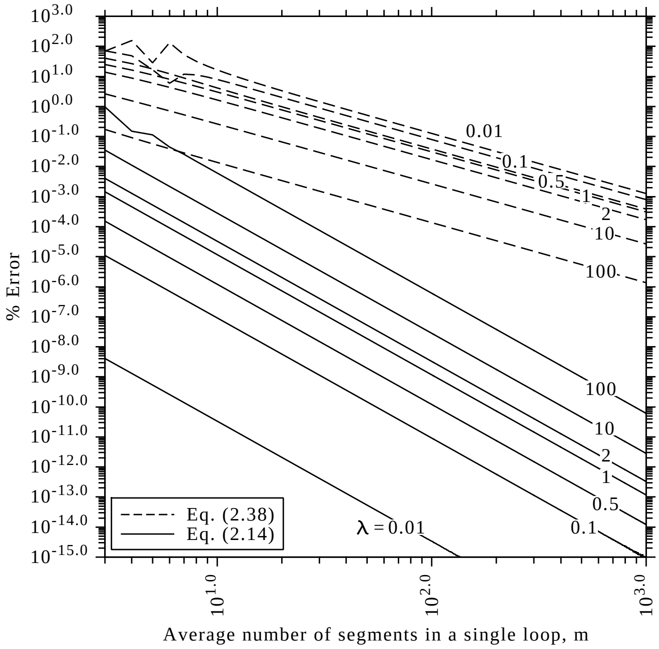


Fig. 2.7: Comparison of Eq. (2.38) and Eq. (2.14) for the z component at location (0,0,0) against the analytical solution, Eq. (2.45) for, 5 loops, various values of λ, using a power clustering factor of 3 for both methods, and 4th order Runge-Kutta for Eq. (2.14).

Interestingly, the derivation of the axial component of velocity along the centerline of the helix can be continued further for those that are curious. Since we have an expression for the analytical solution for Eq. (2.44), we can now take the limit as n approaches ∞ to find the value for a helix that trails off forever as,

$$V_z = \lim_{n \to \infty} \left(\frac{\Gamma z_p}{2b\sqrt{a^2 + z_p^2}} - \frac{\Gamma (z_p - bn)}{2b\sqrt{a^2 + (z_p - bn)^2}} \right) \tag{2.46}$$

$$V_z = \frac{\Gamma}{2b} + \frac{\Gamma z_p}{2b\sqrt{a^2 + z_p^2}} \tag{2.47}$$

with the nondimensional version of Eq. (2.47) as

$$\sigma_z = \frac{1}{2\lambda} + \frac{\hat{z}_p}{2\lambda\sqrt{1 + \hat{z}_p^2}} \tag{2.48}$$

The axial velocity on the centerline of a fully infinite helix, (infinite both directions), can be computed by taking the limit of Eq. (2.47) as z_p approaches ∞. Doing so yields

$$V_{z_\infty} = \lim_{z_p \to \infty} \left(\frac{\Gamma}{2b} + \frac{\Gamma z_p}{2b\sqrt{a^2 + z_p^2}} \right) \tag{2.49}$$

$$V_{z_\infty} = \frac{\Gamma}{b} \tag{2.50}$$

And again the nondimensional form of Eq. (2.50) is

$$\sigma_{z_\infty} = \frac{1}{\lambda} \tag{2.51}$$

Comparing Eq. (2.47) to Eq. (2.13) can be a way to verify that Eq. (2.35) predicts the correct number of loops for the finite helix to approximate a semi-infinite helix.

CHAPTER 3

Numerical Lifting-Line Approach

3.1 Calculating Velocities

3.1.1 Velocity from Propeller Motion

We can start to build a propeller lifting-line method by first analyzing the motion of the propeller blades. A fixed wing's relative motion to the air can be described by simple translational motion, often denoted as $\vec{\boldsymbol{V}}_{\boldsymbol{\infty}}$. The relative motion of a propeller blade to the air is more complex than that for the fixed wing. The propeller blade has the same translational motion as the fixed wing, namely $\vec{\boldsymbol{V}}_{\boldsymbol{\infty}}$, but it also has rotational motion as well. The upstream velocity for a point on the propeller blade is

$$\vec{\boldsymbol{V}}_{\boldsymbol{pi}} = \vec{\boldsymbol{V}}_{\boldsymbol{\infty}} + \vec{\boldsymbol{r}}_{\boldsymbol{i}} \times \vec{\boldsymbol{\omega}} \tag{3.1}$$

where $\vec{\boldsymbol{r}}_{\boldsymbol{i}}$ is a vector pointing from the center of the propeller to the point on the propeller, and $\vec{\boldsymbol{\omega}}$ is the angular velocity of the propeller.

3.1.2 Induced Velocity from the Helical Wake

Now that we know the velocity caused by the motion of the propeller, we need to know the velocity induced by a helical horseshoe vortex. In order to get an expression for the whole horseshoe vortex, we first need the velocity induced by a helical vortex filament. This process is outlined in Appendix A. Using the same coordinate system as shown in Fig. 2.2, and noting that $h_r = -\mathrm{sgn}(\omega_z)$, then Eq. (2.13) can be rewritten as

$$\vec{\boldsymbol{V}}_{\boldsymbol{h}} = \left\{ \begin{aligned} &\frac{\Gamma n}{4\pi} \int_0^1 \frac{-b y_p - \mathrm{sgn}(\omega_z) a b \sin\theta - \mathrm{sgn}(\omega_z) 2\pi a z_p \cos\theta + \mathrm{sgn}(\omega_z) 2\pi a b n \zeta \cos\theta}{\left((x_p - a\cos\theta)^2 + (y_p + \mathrm{sgn}(\omega_z) a \sin\theta)^2 + (z_p - bn\zeta)^2\right)^{\frac{3}{2}}} \,\mathrm{d}\zeta \; \boldsymbol{\hat{\imath}} \\ &+ \frac{\Gamma n}{4\pi} \int_0^1 \frac{b x_p - a b \cos\theta + 2\pi a z_p \sin\theta - 2\pi a b n \zeta \sin\theta}{\left((x_p - a\cos\theta)^2 + (y_p + \mathrm{sgn}(\omega_z) a \sin\theta)^2 + (z_p - bn\zeta)^2\right)^{\frac{3}{2}}} \,\mathrm{d}\zeta \; \boldsymbol{\hat{\jmath}} \\ &+ \frac{\Gamma a n}{2} \int_0^1 \frac{-\mathrm{sgn}(\omega_z) a + \mathrm{sgn}(\omega_z) x_p \cos\theta - y_p \sin\theta}{\left((x_p - a\cos\theta)^2 + (y_p + \mathrm{sgn}(\omega_z) a \sin\theta)^2 + (z_p - bn\zeta)^2\right)^{\frac{3}{2}}} \,\mathrm{d}\zeta \; \boldsymbol{\hat{k}} \end{aligned} \right. \tag{3.2}$$

Knowing the freestream velocity and the angular rate of the propeller, the helix pitch can be calculated as

$$b = \frac{2\pi V_\infty}{\omega} \tag{3.3}$$

where ω and V_∞ are the magnitudes of their respective vectors. Note how the helix pitch is independent of the propeller geometry and is therefore the same for all trailing vortices off of the propeller. Also we can change θ so that it can include a starting angle for the helix by

$$\theta = 2\pi n \zeta + \theta_0 \tag{3.4}$$

where θ_0 is the start angle of the helix. Adding this allows analyzing a helix that doesn't start on the x axis, which is important since not all will. Note that Eq. (3.2) is still valid even with this alternate form of θ.

Now, because of the way we derived Eq. (2.13), when $\mathrm{sgn}(\omega_z) = -1$, $\vec{\boldsymbol{V}}_{\boldsymbol{h}}$ represents the velocity induced by a trailing vortex from a right-handed propeller and a positive Γ value corresponds to a positive axial flow within the helix. When $\mathrm{sgn}(\omega_z) = 1$, the helix is one trailing from a left-handed propeller and a positive Γ value corresponds to a negative axial flow within the helix. This is not desired and we want a positive Γ value to represent a positive axial flow within the helix regardless of propeller rotation direction. In order to fix this, a $-\operatorname{sgn}(\omega_z)$ can be multiplied on to $\vec{\boldsymbol{V}}_{\boldsymbol{h}}$ giving

$$\vec{\boldsymbol{V}}_{\boldsymbol{h}} = \begin{cases} \dfrac{\Gamma n}{4\pi}\displaystyle\int_0^1 \dfrac{\operatorname{sgn}(\omega_z) b y_p + ab\sin\theta + 2\pi a z_p\cos\theta - 2\pi abn\zeta\cos\theta}{\left((x_p - a\cos\theta)^2 + (y_p + \operatorname{sgn}(\omega_z) a\sin\theta)^2 + (z_p - bn\zeta)^2\right)^{\frac{3}{2}}}\,\mathrm{d}\zeta\ \hat{\boldsymbol{\imath}} \\ -\dfrac{\Gamma n}{4\pi}\displaystyle\int_0^1 \dfrac{\operatorname{sgn}(\omega_z)\big(b x_p - ab\cos\theta + 2\pi a z_p\sin\theta - 2\pi abn\zeta\sin\theta\big)}{\left((x_p - a\cos\theta)^2 + (y_p + \operatorname{sgn}(\omega_z) a\sin\theta)^2 + (z_p - bn\zeta)^2\right)^{\frac{3}{2}}}\,\mathrm{d}\zeta\ \hat{\boldsymbol{\jmath}} \\ +\dfrac{\Gamma a n}{2}\displaystyle\int_0^1 \dfrac{a - x_p\cos\theta + \operatorname{sgn}(\omega_z) y_p\sin\theta}{\left((x_p - a\cos\theta)^2 + (y_p + \operatorname{sgn}(\omega_z) a\sin\theta)^2 + (z_p - bn\zeta)^2\right)^{\frac{3}{2}}}\,\mathrm{d}\zeta\ \hat{\boldsymbol{k}} \end{cases} \tag{3.5}$$

Having an expression for the velocity induced by a helical vortex filament, whether for a right or left-handed propeller, we can now write the velocity induced by an entire horseshoe vortex as

$$\vec{\boldsymbol{V}} = \vec{\boldsymbol{V}}_{\boldsymbol{h1}} - \operatorname{sgn}(\omega_z)\frac{\Gamma}{4\pi}\frac{(r_1 + r_2)(\vec{\boldsymbol{r}}_{\boldsymbol{1}} \times \vec{\boldsymbol{r}}_{\boldsymbol{2}})}{r_1 r_2 (r_1 r_2 + \vec{\boldsymbol{r}}_{\boldsymbol{1}} \cdot \vec{\boldsymbol{r}}_{\boldsymbol{2}})} + \vec{\boldsymbol{V}}_{\boldsymbol{h2}} \tag{3.6}$$

where $\vec{\boldsymbol{V}}_{\boldsymbol{h1}}$ corresponds with the smaller radius helix of the two. Note that the since we multiplied a $-\operatorname{sgn}(\omega_z)$ term onto $\vec{\boldsymbol{V}}_{\boldsymbol{h}}$ earlier, we have to do the same for the bound segment section as well. Now we can write

$$\vec{\boldsymbol{V}}_{\boldsymbol{h}} = \Gamma\,\vec{\boldsymbol{v}}_{\boldsymbol{h}} \tag{3.7}$$

where $\vec{\boldsymbol{v}}_{\boldsymbol{h}}$ is calculated the same as Eq. (3.5) except with Γ taken out. Now, the velocity induced at control point i from horseshoe vortex j can be rewritten more precisely as

$$\vec{\boldsymbol{V}}_{\boldsymbol{ji}} = \Gamma_j \vec{\boldsymbol{\nu}}_{\boldsymbol{ji}} \tag{3.8}$$

$$\vec{\boldsymbol{\nu}}_{\boldsymbol{ji}} = \begin{cases} \vec{\boldsymbol{v}}_{\boldsymbol{h}_{\boldsymbol{j_2 i}}} - \vec{\boldsymbol{v}}_{\boldsymbol{h}_{\boldsymbol{j_1 i}}} - \operatorname{sgn}(\omega_z)\dfrac{(r_{j_1 i} + r_{j_2 i})(\vec{\boldsymbol{r}}_{\boldsymbol{j_1 i}} \times \vec{\boldsymbol{r}}_{\boldsymbol{j_2 i}})}{4\pi r_{j_1 i} r_{j_2 i}(r_{j_1 i} r_{j_2 i} + \vec{\boldsymbol{r}}_{\boldsymbol{j_1 i}} \cdot \vec{\boldsymbol{r}}_{\boldsymbol{j_2 i}})} & i \neq j \\ \vec{\boldsymbol{v}}_{\boldsymbol{h}_{\boldsymbol{j_2 i}}} - \vec{\boldsymbol{v}}_{\boldsymbol{h}_{\boldsymbol{j_1 i}}} & i = j \end{cases} \tag{3.9}$$

where Fig. 3.1 can help with understanding the subscripts and geometry involved. The subscripts j_1 and j_2 correspond to the node points on horseshoe vortex j, while i represents

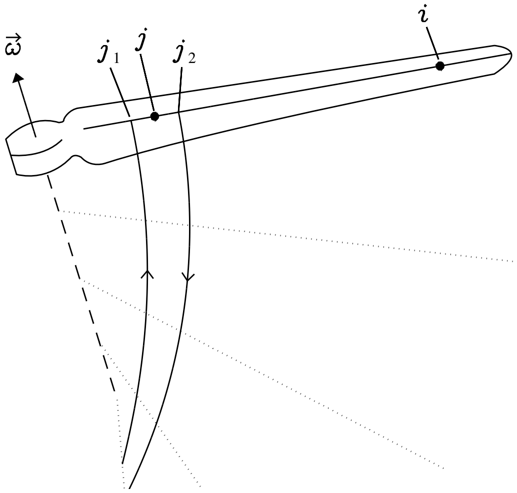


Fig. 3.1: Arbitrary helical horseshoe vortex with control and node points shown on one blade of a right-handed propeller.

the control point. Note that node j_1 is always closer to the axis of rotation of the propeller, shown as the dashed line in Fig. 3.1, than node j_2. This notation is similar to that in Eq. (3.6).

The total velocity at control point i can be found by combining Eq. (3.1) and Eq. (3.8) and accounting for all the horseshoe vortices from all blades of the propeller as

$$\vec{\boldsymbol{V}}_{\boldsymbol{i}} = \vec{\boldsymbol{V}}_{\boldsymbol{\infty}} + \vec{\boldsymbol{r}}_{\boldsymbol{i}} \times \vec{\boldsymbol{\omega}} + \sum_{j=1}^{N} \Gamma_j \vec{\boldsymbol{\nu}}_{\boldsymbol{ji}} \tag{3.10}$$

where N is the total number of control points, accounting for all blades on the propeller.

3.2 Lifting-Line Solution Method

3.2.1 Fundamental Lifting-Line Equation

Now that we have an expression for the total velocity at a given control point, we can turn our focus to the lifting-line method. Classical lifting-line theory suggests that a given spanwise section of the wing has a section lift equal to the lift on a similar section of an infinite wing with the an angle of attack equal to the local angle of attack of the original spanwise section. Using this we can use the three-dimensional vortex lifting law for a differential segment of the wing to give us the section lift, [3] which gives

$$\mathbf{d}\boldsymbol{\vec{F_i}} = \rho \Gamma_i \boldsymbol{\vec{V_i}} \times \mathbf{d}\boldsymbol{\vec{l_i}} \tag{3.11}$$

where ρ is the density of air and $\mathbf{d}\boldsymbol{\vec{l_i}}$ is a spatial vector going from node 1 to node 2 of section i, for a right-handed propeller, or from node 2 to node 1 for a left-handed propeller. Then, from the lifting-line theory, we can relate this as

$$|\mathbf{d}\boldsymbol{\vec{F_i}}| = \frac{1}{2}\rho V_i^2 \widetilde{C}_{Li}(\alpha_i)\,\mathrm{d}A_i \tag{3.12}$$

where $\widetilde{C}_{Li}$ is the two-dimensional section lift coefficient which is a function of angle of attack, $\widetilde{C}_{Li} = \widetilde{C}_{Li}(\alpha_i)$ and $\mathrm{d}A_i$ is the local section planform area. The local angle of attack, α_i, is given by

$$\alpha_i = \tan^{-1}\left(\frac{\boldsymbol{\vec{V_i}} \cdot \boldsymbol{\hat{u}_{ni}}}{\boldsymbol{\vec{V_i}} \cdot \boldsymbol{\hat{u}_{ai}}}\right) \tag{3.13}$$

where $\boldsymbol{\hat{u}_{ni}}$ and $\boldsymbol{\hat{u}_{ai}}$ are unit vectors in the normal and axial directions respectively to the chord line of the local airfoil section. Both vectors lie within the plane of the local airfoil section as shown in Fig. 3.2.

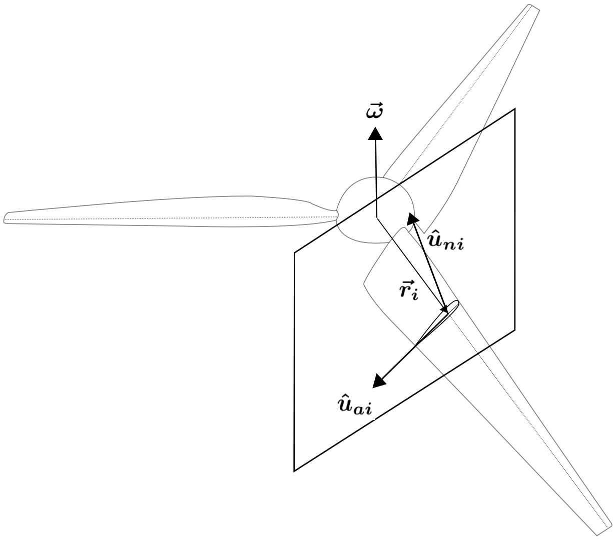


Fig. 3.2: Local unit vectors shown on a propeller.

Using Eqs. (3.11) and (3.12) to set the magnitude of the local section force equal to each other and rearranging gives the lifting-line formula as

$$2\left|\Gamma_i\left(\vec{\boldsymbol{V}}_{\boldsymbol{\infty}}+\vec{\boldsymbol{r}}_{\boldsymbol{i}}\times\vec{\boldsymbol{\omega}}+\sum_{j=1}^{N}\Gamma_j\vec{\boldsymbol{\nu}}_{\boldsymbol{ji}}\right)\times\mathbf{d}\vec{\boldsymbol{l}}_{\boldsymbol{i}}\right|-V_i^2\widetilde{C}_{Li}(\alpha_i)\,\mathrm{d}A_i=0 \tag{3.14}$$

which can be used to write N equations, one for each control point, leaving a system of nonlinear equations with N unknown vortex strengths. The system of equations can then be solved using Newton's method. Note that if a linear variation of chord length over each spanwise section is assumed then

$$\mathrm{d}A_i=\int_{s_1}^{s_2}c\,\mathrm{d}s=\frac{c_{i_1}+c_{i_2}}{2}|\,\mathbf{d}\vec{\boldsymbol{l}}_{\boldsymbol{i}}|\cos\Lambda_i \tag{3.15}$$

where c_{i_1} is the chord length at the node i_1, and Λ_i is the sweep angle at control point i.

3.2.2 Jacobian Derivation

In order to use Newton's method, we need to put the system of equations in vector form as follows

$$f(\boldsymbol{\Gamma})=\boldsymbol{R} \tag{3.16}$$

where

$$f_i(\boldsymbol{\Gamma})=2\left|\Gamma_i\left(\vec{\boldsymbol{V}}_{\boldsymbol{\infty}}+\vec{\boldsymbol{r}}_{\boldsymbol{i}}\times\vec{\boldsymbol{\omega}}+\sum_{j=1}^{N}\Gamma_j\vec{\boldsymbol{\nu}}_{\boldsymbol{ji}}\right)\times\mathbf{d}\vec{\boldsymbol{l}}_{\boldsymbol{i}}\right|-V_i^2\widetilde{C}_{Li}(\alpha_i)\,\mathrm{d}A_i \tag{3.17}$$

We now need to find the vortex strength vector $\boldsymbol{\Gamma}$ that will make the residual vector $\boldsymbol{R}$ go to zero. Starting with an initial guess for $\boldsymbol{\Gamma}$, we will have a corresponding $\boldsymbol{R}$ that is likely non-zero. In order to zero it out, we need to change the residual vector by $-\boldsymbol{R}$. We can then find out how much we should change our initial guess of $\boldsymbol{\Gamma}$ by

$$[\boldsymbol{J}]\boldsymbol{\Delta\Gamma}=-\boldsymbol{R} \tag{3.18}$$

where $[\boldsymbol{J}]$ is the N by N Jacobian matrix of partial derivatives given by [15]

$$J_{ij} = \frac{\partial f_i}{\partial \Gamma_j} = \begin{cases} \left[\begin{aligned} & 2\Gamma_i \frac{\boldsymbol{\vec{w}_i} \cdot (\boldsymbol{\vec{\nu}_{ji}} \times \mathbf{d}\boldsymbol{\vec{l}_i})}{|\boldsymbol{\vec{w}_i}|} - 2\widetilde{C}_{Li} \boldsymbol{\vec{V}_i} \cdot \boldsymbol{\vec{\nu}_{ji}} \, \mathrm{d}A_i \\ & - V_i^2 \frac{\partial \widetilde{C}_{Li}}{\partial \alpha_i} \frac{V_{ai}(\boldsymbol{\vec{\nu}_{ji}} \cdot \boldsymbol{\hat{u}_{ni}}) - V_{ni}(\boldsymbol{\vec{\nu}_{ji}} \cdot \boldsymbol{\hat{u}_{ai}})}{V_{ni}^2 + V_{ai}^2} \, \mathrm{d}A_i \end{aligned} \right], & i \neq j \\ \left[\begin{aligned} & 2|\boldsymbol{\vec{w}_i}| + 2\Gamma_i \frac{\boldsymbol{\vec{w}_i} \cdot (\boldsymbol{\vec{\nu}_{ji}} \times \mathbf{d}\boldsymbol{\vec{l}_i})}{|\boldsymbol{\vec{w}_i}|} - 2\widetilde{C}_{Li} \boldsymbol{\vec{V}_i} \cdot \boldsymbol{\vec{\nu}_{ji}} \, \mathrm{d}A_i \\ & - V_i^2 \frac{\partial \widetilde{C}_{Li}}{\partial \alpha_i} \frac{V_{ai}(\boldsymbol{\vec{\nu}_{ji}} \cdot \boldsymbol{\hat{u}_{ni}}) - V_{ni}(\boldsymbol{\vec{\nu}_{ji}} \cdot \boldsymbol{\hat{u}_{ai}})}{V_{ni}^2 + V_{ai}^2} \, \mathrm{d}A_i \end{aligned} \right], & i = j \end{cases} \tag{3.19}$$

where

$$\boldsymbol{\vec{w}_i} = \boldsymbol{\vec{V}_i} \times \mathbf{d}\boldsymbol{\vec{l}_i} \tag{3.20}$$

$$V_{ni} = \boldsymbol{\vec{V}_i} \cdot \boldsymbol{\hat{u}_{ni}} \tag{3.21}$$

$$V_{ai} = \boldsymbol{\vec{V}_i} \cdot \boldsymbol{\hat{u}_{ai}} \tag{3.22}$$

Solving for the correction vector, $\boldsymbol{\Delta\Gamma}$, we can now find a new guess by

$$\boldsymbol{\Gamma} = \boldsymbol{\Gamma} + \Omega \boldsymbol{\Delta\Gamma} \tag{3.23}$$

where Ω is a relaxation factor. This process can be iterated until the residual vector $\boldsymbol{R}$ meets some tolerance criteria, such as: $\max(\text{abs}(\boldsymbol{R})) < tol$.

3.2.3 Initial Guess

Having a good first guess for $\boldsymbol{\Gamma}$ can be crucial to the effectiveness of Newton's method to converge quickly and accurately. One method to determine an initial guess is to linearize Eq. (3.14) and solve the system of equations for $\boldsymbol{\Gamma}$. If we first assume a small angle of attack we can write

$$\alpha_i \cong \boldsymbol{\vec{V}_i} \cdot \boldsymbol{\hat{u}_{ni}} = \boldsymbol{\vec{V}_{pi}} \cdot \boldsymbol{\hat{u}_{ni}} + \sum_{j=1}^{N} \Gamma_j \boldsymbol{\vec{\nu}_{ji}} \cdot \boldsymbol{\hat{u}_{ni}} \tag{3.24}$$

Then we can approximate the lift coefficient as

$$\widetilde{C}_{Li} \cong \widetilde{C}_{Li,\alpha}(\alpha_i - \alpha_{L0i}) \tag{3.25}$$

$$\widetilde{C}_{Li} \cong \widetilde{C}_{Li,\alpha}(\boldsymbol{\vec{V}_{pi}} \cdot \boldsymbol{\hat{u}_{ni}} + \sum_{j=1}^{N} \Gamma_j \boldsymbol{\vec{\nu}_{ji}} \cdot \boldsymbol{\hat{u}_{ni}} - \alpha_{L0i}) \tag{3.26}$$

Making one more approximation, $\boldsymbol{\vec{V}_i} \cong \boldsymbol{\vec{V}_{pi}}$, we can rewrite Eq. (3.14) that is linearized as

$$2\left|\Gamma_i \boldsymbol{\vec{V}_{pi}} \times \mathbf{d}\boldsymbol{\vec{l}_i}\right| - V_{pi}^2 \widetilde{C}_{Li,\alpha} \sum_{j=1}^{N} \Gamma_j \boldsymbol{\vec{\nu}_{ji}} \cdot \boldsymbol{\hat{u}_{ni}} \, \mathrm{d}A_i \cong V_{pi}^2 \widetilde{C}_{Li,\alpha}(\boldsymbol{\vec{V}_{pi}} \cdot \boldsymbol{\hat{u}_{ni}} - \alpha_{L0i}) \, \mathrm{d}A_i \tag{3.27}$$

Unfortunately, Eq. (3.27) isn't very accurate. Propeller blades are much more likely to have sections that are stalled than regular wings. Therefore, the small-angle approximation for the angle of attack isn't valid. The lift coefficient approximation is also invalid, because it doesn't include stall effects and is only accurate up to a certain angle of attack. The last approximation is also invalid since it assumes that no induced velocity is generated by the propeller.

Another way to determine an initial guess is to analyze the change in velocity through the propeller disk. We can write

$$T = \dot{m} V_g \tag{3.28}$$

where T is the thrust produced by the propeller, $\dot{m}$ is the mass flow rate through the propeller disk, and V_g is the change is velocity through the propeller disk. The thrust can also be written as

$$T = C_T \rho (\omega / 2\pi)^2 d_p^4 \tag{3.29}$$

where C_T is the coefficient of thrust and d_p is the diameter of the propeller. We also know that

$$\dot{m} = \rho A_p (V_\infty + V_g) \tag{3.30}$$

with A_p being the area of the propeller disk. Combining Eqs. (3.29) and (3.30) into Eq. (3.28) yields

$$V_g = \frac{T}{\dot{m}} = \frac{C_T \rho \omega^2 d_p^4}{(2\pi)^2 \rho A_p (V_\infty + V_g)} \tag{3.31}$$

Solving for V_g requires using the quadratic formula and gives

$$V_g = \frac{-V_\infty + \sqrt{V_\infty^2 + 16 C_T \omega^2 r_p^2/\pi^3}}{2} \tag{3.32}$$

where the answer with the plus sign is the valid solution. Now that we have an expression for the axial change in velocity across the propeller disk, we can relate it back to the z component of the induced velocity part of $\vec{\boldsymbol{V}}_{\boldsymbol{i}}$ as

$$\sum_{j=1}^{N} \Gamma_j \nu_{zji} = V_g \tag{3.33}$$

which can be written for each control point i resulting in a linear system of N equations from which $\mathbf{\Gamma}$ can be solved. The only unknown in Eqs. (3.33) and (3.32) other than $\mathbf{\Gamma}$ is C_T.

Using a guess for C_T makes Eq. (3.33) a better first guess for $\mathbf{\Gamma}$ than Eq. (3.27). If the user doesn't have a good feel for a guess on C_T for their propeller geometry and run scenario, then a default value of $C_T = 0.05$ or Eq. (3.27) can be used. Usually using the default value of $C_T = 0.05$ gives a better first guess on $\mathbf{\Gamma}$ than Eq. (3.27). Either method presented here is only good enough as a first estimate for $\mathbf{\Gamma}$ and should be iterated on as explained in the previous section.

3.3 Forces and Moments

Once the vortex strengths have been found, the forces and moments along the propeller blades can be determined. The total aerodynamic force can be found by using Eq. (3.11). If a large number of control points are used on the propeller blades, such that each point corresponds to a relatively small spanwise section of the blade, then we can assume that the aerodynamic force is approximately constant across each spanwise section and the total aerodynamic force is

$$\boldsymbol{\vec{F}} = \rho \sum_{i=1}^{N} \Gamma_i \left(\boldsymbol{\vec{V}_{\infty}} + \boldsymbol{\vec{r}_i} \times \boldsymbol{\vec{\omega}} + \sum_{j=1}^{N} \Gamma_j \boldsymbol{\vec{\nu}_{ji}} \right) \times \mathbf{d}\boldsymbol{\vec{l}_i} \tag{3.34}$$

The aerodynamic moments can be calculated using the same assumptions used for the aerodynamic forces. The total aerodynamic moment about the center of the propeller is

$$\boldsymbol{\vec{M}} = \rho \sum_{i=1}^{N} \boldsymbol{\vec{r}_i} \times \left[\Gamma_i \left(\boldsymbol{\vec{V}_{\infty}} + \boldsymbol{\vec{r}_i} \times \boldsymbol{\vec{\omega}} + \sum_{j=1}^{N} \Gamma_j \boldsymbol{\vec{\nu}_{ji}} \right) \times \mathbf{d}\boldsymbol{\vec{l}_i} \right] + \boldsymbol{\delta \vec{M}_i} \tag{3.35}$$

where $\boldsymbol{\delta \vec{M}_i}$ is an equivalent local pitching moment for a given control point.

If we assume that the propeller has no dihedral, is symmetric, and is faced directly into the freestream velocity, a pitching moment from a control point on one blade will be symmetric to other pitching moments for corresponding control points on those blades. Therefore,

$$\sum_{i=1}^{N} \boldsymbol{\delta \vec{M}_i} = \boldsymbol{\vec{0}} \tag{3.36}$$

and the total aerodynamic moment becomes

$$\boldsymbol{\vec{M}} = \rho \sum_{i=1}^{N} \boldsymbol{\vec{r}_i} \times \left[\Gamma_i \left(\boldsymbol{\vec{V}_{\infty}} + \boldsymbol{\vec{r}_i} \times \boldsymbol{\vec{\omega}} + \sum_{j=1}^{N} \Gamma_j \boldsymbol{\vec{\nu}_{ji}} \right) \times \mathbf{d}\boldsymbol{\vec{l}_i} \right] \tag{3.37}$$

Caution should be used with Eq. (3.36), since it only implies that the sum of the all local aerodynamic moments is zero and not that the local aerodynamic moments are zero. If one wanted to analyze the aerodynamic loading placed on the propeller blades or if one of the conditions stated above is not met, then $\boldsymbol{\delta \vec{M}_i}$ will need to be taken into account. For this purpose $\boldsymbol{\delta \vec{M}_i}$ is given by

$$\boldsymbol{\delta \vec{M}_i} = -\frac{1}{2} \rho V_i^2 \widetilde{C}_{mi} \frac{c_{i_1} + c_{i_2}}{2} \, \mathrm{d}A_i \boldsymbol{\hat{u}_{si}} = -\frac{1}{8} \rho V_i^2 \widetilde{C}_{mi} (c_{i_1} + c_{i_2})^2 (s_{i_2} - s_{i_1}) \boldsymbol{\hat{u}_{si}} \tag{3.38}$$

where $\widetilde{C}_{mi}$ is the local section moment coefficient and $\boldsymbol{\hat{u}_{si}}$ is another unit vector given by the propeller geometry and is given by $\boldsymbol{\hat{u}_{si}} = \boldsymbol{\hat{u}_{ai}} \times \boldsymbol{\hat{u}_{ni}}$. Equation (3.38) assumes that the chord length varies linearly between control points, and that the section has a constant moment coefficient, which are similar assumptions made for the forces.

3.4 Control and Node Point Locations

Up to this point, we haven't mentioned anything about the location of the control and node points along the propeller blade. Figure 1.1 shows the nodes equally spaced along the propeller blade. Typically, lifting-line methods don't place the nodes equally spaced along the lifting surface, opting instead to cluster the points near the edges of the lifting surface. A common technique to place the control and node points is the cosine clustering technique. Cosine clustering can be achieved based on the change of variables

$$s = \frac{r_p - r_h}{2}(1 - \cos\phi) + r_h \tag{3.39}$$

where r_h is the hub radius, r_p is the propeller radius, s is the spanwise variable down the length of the propeller blade, and ϕ is the change of variables used to do the clustering. Placing the nodes and control points equally in ϕ from $0 \leq \phi \leq \pi$ will cluster the points towards the hub at the inside of the propeller blade and towards the outside edges of the propeller blade. This could then be repeated for each blade, or alternatively, the original blade geometry could be rotated around the z axis to get the the node and control points for the other blades.

If the number of control points per blade is known then the spanwise nodal coordinate is given by

$$s_i = \frac{r_p - r_h}{2}\left[1 - \cos\left(\frac{i\pi}{N_b}\right)\right] + r_h \qquad 0 \leq i \leq N_b \tag{3.40}$$

where N_b is the number of control points per blade, and s_i represents the spanwise location of the second node on the i^{th} horseshoe vortex. Recall that the first node on horseshoe i is the same point as the second node on horseshoe $i-1$. Noting this, s_0 is the same as the first node on the first horseshoe vortex. Also, the control points should be placed midway between their node points with respect to ϕ. This is realized by

$$s_i = \frac{r_p - r_h}{2}\left[1 - \cos\left(\frac{i\pi}{N_b} - \frac{\pi}{2N_b}\right)\right] + r_h \qquad 1 \leq i \leq N_b \tag{3.41}$$

where s_i is now the spanwise coordinate for control point i.

CHAPTER 4

Iterative Semi-Free Wake Method

4.1 Using the Given Pitch

Up to this point we have enough developed that we can analyze our method. A brief comparison of the propeller lifting-line method against propeller blade theory is presented. The propeller used for this comparison has a diameter of 2 ft, a constant chord-line pitch length of 1.8 ft, and 2 blades. It has a hub radius of 0.1 ft and a constant airfoil section with a zero-lift angle of attack of -2.1 degrees and a coefficient of lift given by

$$\widetilde{C}_L = \begin{cases} 2\pi\alpha_b, & \alpha_b \leq 0.25 \\ \dfrac{\pi}{2}\dfrac{\cos\alpha_b}{\cos(0.25)}, & \alpha_b > 0.25 \end{cases} \tag{4.1}$$

where α_b is the angle of attack measured with respect to the zero-lift line of the airfoil. The coefficient of drag was set to 0 for this analysis. The twist distribution of the blade is given by

$$\beta = \tan^{-1}\left(\frac{\lambda_c - 2\pi r \tan\alpha_{L0}}{2\pi r + \lambda_c \tan\alpha_{L0}}\right) \tag{4.2}$$

where β is the twist angle measured from the zero-lift line to the propeller plane, λ_c is the chord-line pitch length, α_{L0} is the zero-lift angle of attack, and r is the radial coordinate that varies from the hub to tip of the propeller. The chord length is given by the relation

$$c = 0.075 d_p \sqrt{1 - \left(\frac{2r}{d_p}\right)^2} \tag{4.3}$$

and d_p is the diameter of the propeller. For a more detailed explanation of the propeller geometry and the propeller blade theory used for this comparison, see Phillips. [10]

Coefficients of thrust and power are compared between the two methods, for a range of advance ratios as shown in Figs. 4.1 and 4.2 respectively. The advance ratio is determined

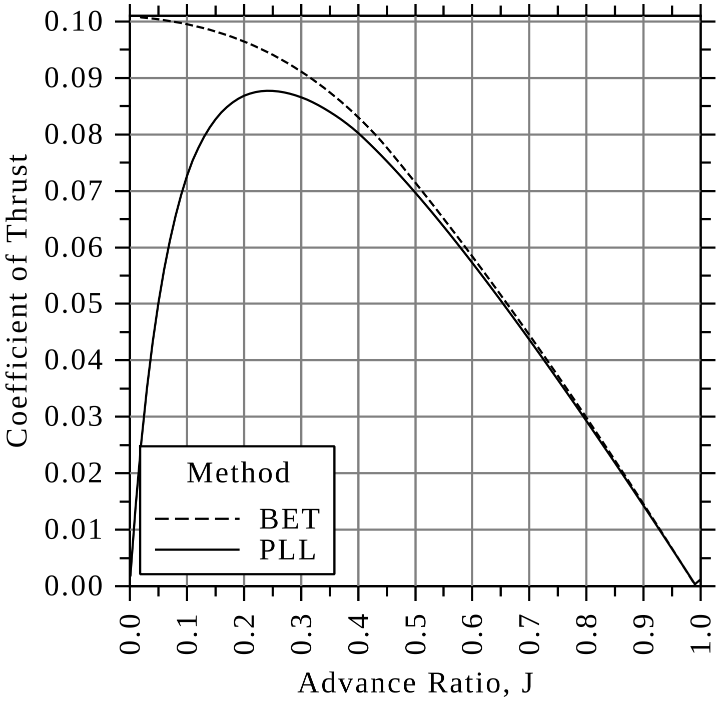


Fig. 4.1: Coefficient of Thrust versus Advance Ratio for the propeller lifting-line method (PLL) and blade element theory (BET).

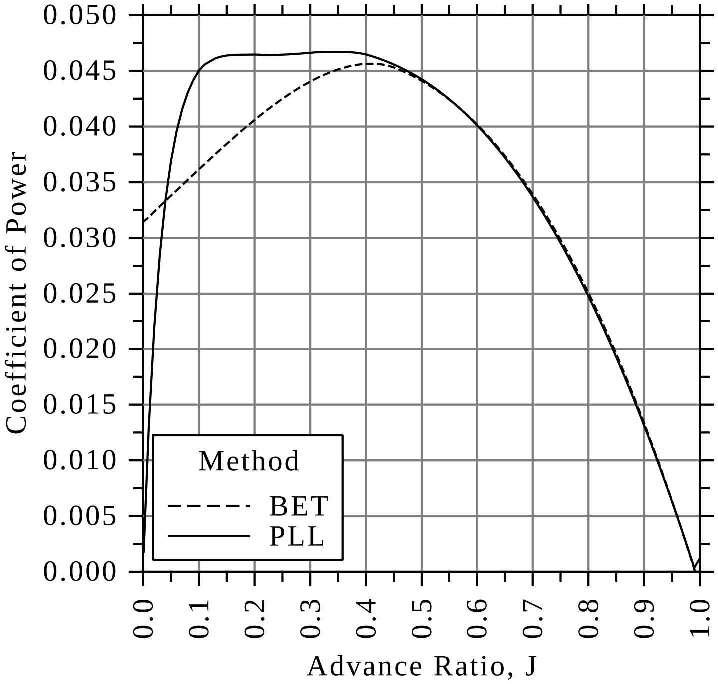


Fig. 4.2: Coefficient of Power versus Advance Ratio for the propeller lifting-line method (PLL) and blade element theory (BET).

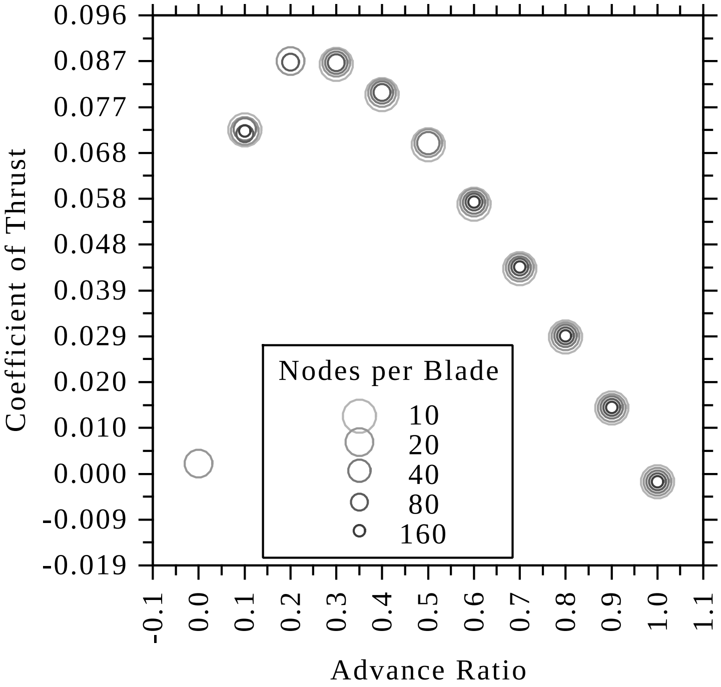


Fig. 4.3: Grid Resolution of the Coefficient of Thrust versus Advance Ratio for the propeller lifting-line method (PLL).

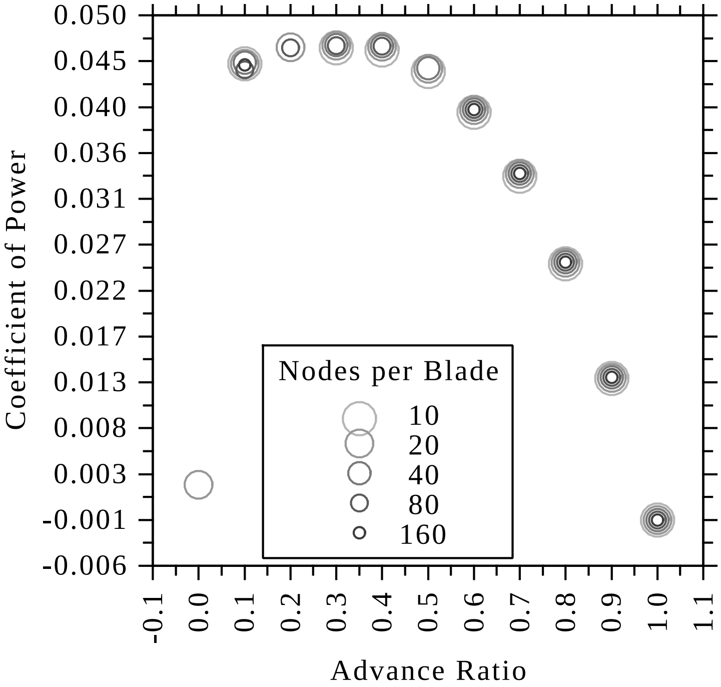


Fig. 4.4: Grid Resolution of the Coefficient of Power versus Advance Ratio for the propeller lifting-line method (PLL).

by freestream velocity, the angular rate of the propeller, and the propeller diameter to give

$$J = \frac{2\pi V_\infty}{\omega d_p} \tag{4.4}$$

and the coefficient of thrust was calculated as

$$C_T = \frac{T}{\rho(\omega/2\pi)^2 d_p^4} \tag{4.5}$$

where T and C_T are respectively the aerodynamic thrust and coefficient of thrust, while the coefficient of power was calculated by

$$C_P = \frac{\ell\omega}{\rho(\omega/2\pi)^3 d_p^5} \tag{4.6}$$

where ℓ is the aerodynamic torque and C_P is the coefficient of power. A grid resolution test is shown in Figs. 4.3-4.4 and is comparable to Figs. 4.1-4.2. Some data points are missing due to stall on inward portions of the propeller blade that can cause convergence issues with the nonlinear solver. The grid resolution tests shown in Figs. 4.3-4.4 are repeated for BET and are shown in the appendix.

Looking at Fig. 4.1, one can quickly determine that the propeller lifting-line method does not do well at lower advance ratios, but has good agreement with blade element theory at higher advance ratios. The reason for this is a lack of incorporating induced velocities in Eq. (3.3) when determining the helical pitch, b, for the trailing vortices. For a lifting-line method on a fixed wing, the induced downwash is relatively small compared to the freestream, and the effect of the downwash on the trailing vortices' path is negligible. The induced downwash within the slipstream of the propeller is large compared to the induced downwash caused by a fixed wing. Therefore, the induced velocity should be taken into account when determining the helical pitch of the trailing vortices.

4.2 Global Pitch Solver

One way to fix this is to iterate on the propeller lifting-line method correcting for pitch until convergence. Doing so changes the wake from a forced helical system, to one that is free with respect to one variable, b, the helical pitch. To begin the iterative pitch solver we first need an initial guess for b. As a first estimate for b we could use Eq. (3.3), but if we are already using Eq. (3.33) for the lifting-line method it is advantageous to use

$$b = \frac{2\pi(V_\infty + V_g)}{\omega} \tag{4.7}$$

which will provide a better first estimate for b.

Once we have an initial guess for b, then Eq. (3.14) should be solved using the Newton's method described previously. After solving for the vortex strengths of the horseshoe vortices we now have accurate values for $\vec{\boldsymbol{V}}_{\boldsymbol{i}}$. From $\vec{\boldsymbol{V}}_{\boldsymbol{i}}$, we can take the average axial component from all control points as

$$V_{z_{\text{avg}}} = \frac{1}{N}\sum_{i=1}^{N} V_{zi} \tag{4.8}$$

A temporary new pitch value is

$$b_t = \frac{2\pi V_{z_{\text{avg}}}}{\omega} \tag{4.9}$$

with b_t being the temporary new pitch value. It is advantageous to have a temporary new pitch value because it is more efficient to have a relaxation factor on the pitch solver. A change in pitch value can then be calculated by

$$\Delta b = b_t - b_o \tag{4.10}$$

where b_o is the previous pitch value. A new pitch value can then be determined by

$$b_n = b_o + \Omega_b \Delta b \tag{4.11}$$

where b_n is the new pitch value and Ω_b is a relaxation factor for the pitch solver. The lifting-line method would then be repeated and another b_t would then be calculated. When repeating the lifting-line method, the $\mathbf{\Gamma}$ values solved from the previous iteration should be used as the first guess to improve accuracy and run time. This should be repeated until some tolerance criteria is met, such as

$$\text{abs}\left(\frac{\Delta b}{b_t}\right) < tol \tag{4.12}$$

This method is called a global pitch solver, since it uses a single pitch value for all helices/node points.

4.3 Local Pitch Solver

A local pitch solver can also be used. This would allow each helix/node point to have their own pitch value. To do so, we can follow the same steps as for the global pitch solver, except instead of using an average axial velocity, we would need to calculate the velocity at the node points. Calculating the velocity at the node points can be a little tricky, but if we do, all of the other steps for the local pitch solver would be the same as that from the global pitch solver, except that each node point would iterate on pitch until the max error from any node point met the tolerance criteria.

We cannot calculate the velocity at the node points the same way we calculated the velocity at the control points. This is because any given node point sits on the tip of the helix for the current and adjacent horseshoe vortices. This causes a singularity in Eq. (3.5) and the velocity cannot be calculated.

A means around this could be to average the axial velocities from the control points on either side of a node point and use that velocity. Then at the ends, just assume that the axial velocity at the node point is the same as that from the closest control point. If cosine clustering is used to place the node and control points as presented earlier, then the control point at the end would be very close to the node point and this would be a reasonably valid approximation.

This technique could be extended further to interpolate the axial velocity at the node point from the two adjacent control points. The axial velocity from the end node points could be extrapolated from the two closest control points. Either of these methods to approximate the axial velocity at the node points works reasonably well, especially if a larger number of control points are used.

The local pitch solver agrees well with the global pitch solver at higher and mid range advance ratios. Although it has issues with convergence at lower advance ratios. Overall, the local pitch solver takes longer to run and doesn't provide much difference from the global pitch solver as far as results. For these reasons the global pitch solver is recommended over the local pitch solver.

4.4 Semi-Free Wake

Implementing either of these pitch solver methods changes the propeller lifting-line method presented from a prescribed wake model to a semi-free wake model. The wake is still prescribed in terms of helix radius and constant pitch, although it is free in determining the value of the constant pitch. This means that the semi-free wake model here still fails to account for slipstream contraction and vortex roll up, but it does account for, at least in part, the strong axial flow within the slipstream. This allows the propeller lifting-line method to work at lower advance ratios when the blades are highly loaded and the wake geometry is highly depended on the induced velocity.

Repeating the analysis that was done for Figs. 4.1 and 4.2, except with the global pitch solver used with the propeller lifting-line method gives better results. The results are shown in Figs. 4.5 and 4.6. The two methods give nearly identical results for the total thrust and power coefficients for the upper range of advance ratios shown. At lower advance ratios the difference in results from the two methods are more noticeable.

A grid resolution test is shown in Figs. 4.7 and 4.8 and is comparable to Figs. 4.5 and 4.6. As before, there are some data points missing on these plots due to stall conditions which can sometimes cause the nonlinear solver to fail. Overall, the propeller lifting-line method grid resolves nicely. Using the data points from Fig. 4.7 for an advance ratio of 0.6 and plotting them against the nodes per blade can help get a better understanding of the grid resolution as seen in Fig. 4.9 where the coefficient of thrust clearly approaches a given value as the nodes per blade are increased. The plots shown in Figs. 4.7-4.9 are repeated using BET and are shown in the appendix.

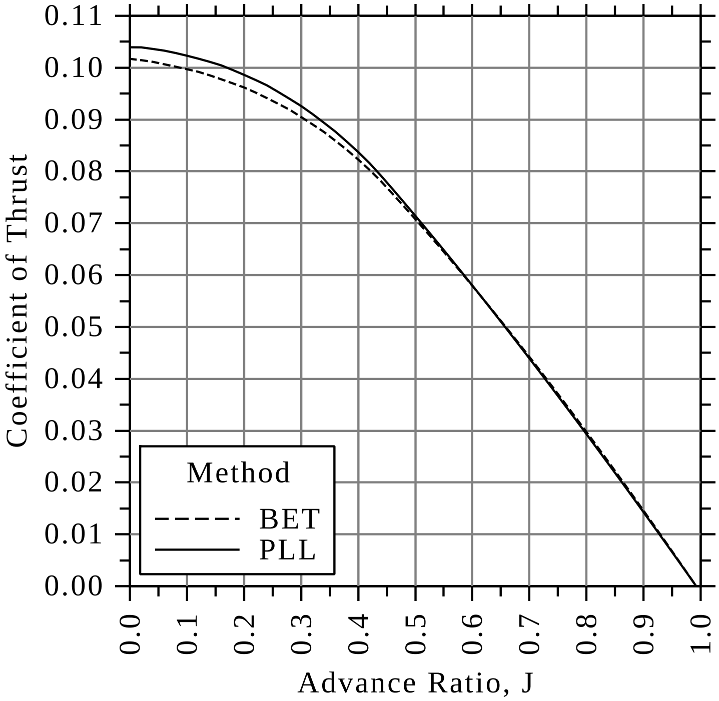


Fig. 4.5: Coefficient of Thrust versus Advance Ratio for the propeller lifting-line method (PLL) with the global pitch solver and blade element theory (BET).

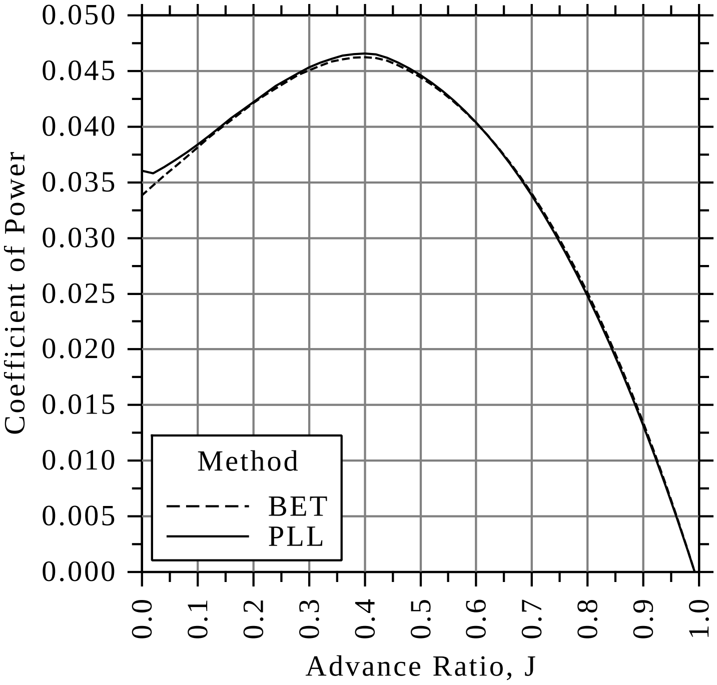


Fig. 4.6: Coefficient of Power versus Advance Ratio for the propeller lifting-line method (PLL) with the global pitch solver and blade element theory (BET).

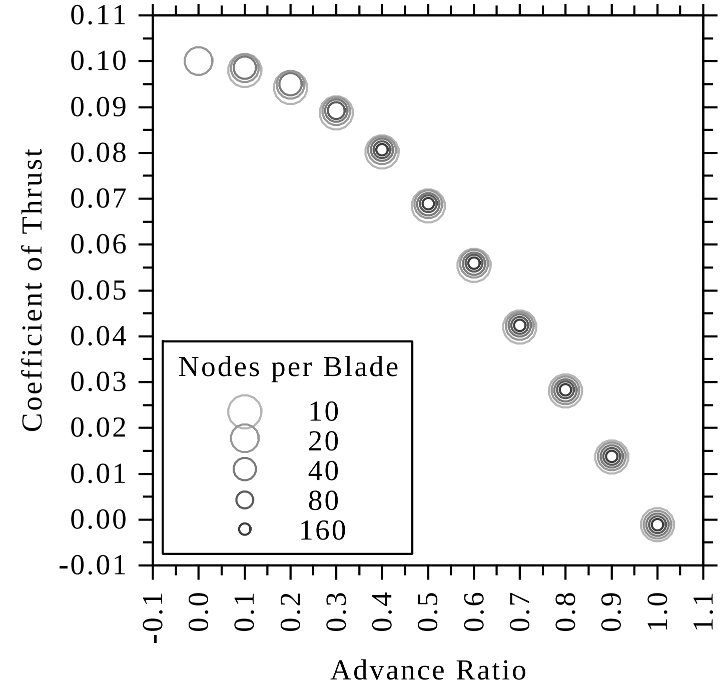


Fig. 4.7: Grid Resolution of the Coefficient of Thrust versus Advance Ratio for the propeller lifting-line method (PLL) with the global pitch solver.

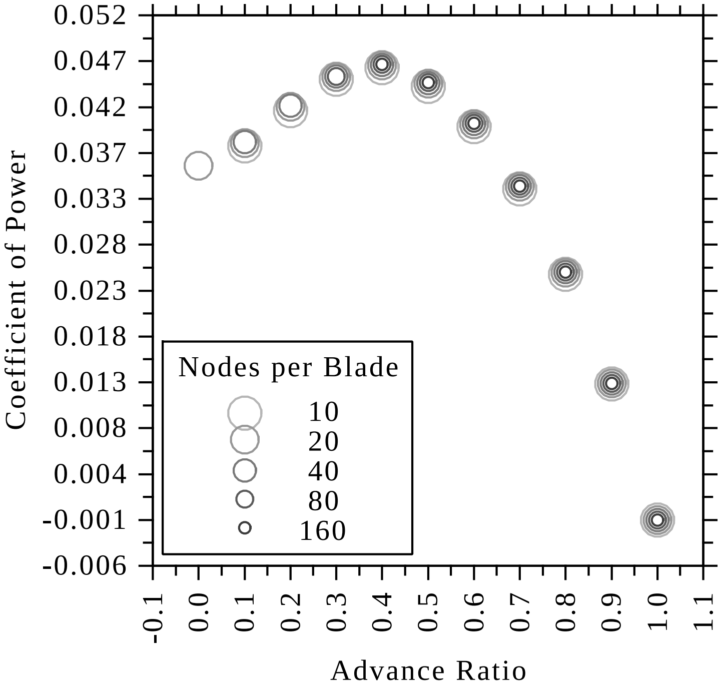


Fig. 4.8: Grid Resolution of the Coefficient of Power versus Advance Ratio for the propeller lifting-line method (PLL) with the global pitch solver.

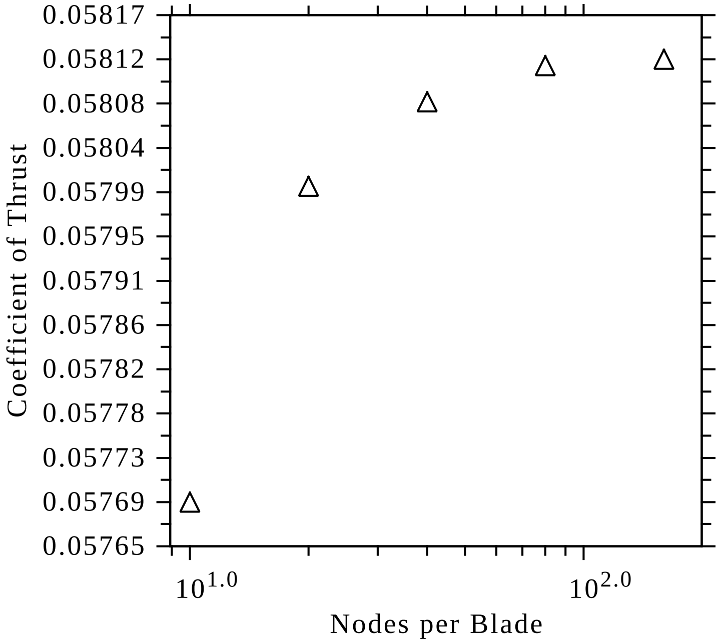


Fig. 4.9: Coefficient of Thrust values vs the nodes per blade calculated by the propeller lifting-line method with the global pitch solver at an advance ratio of 0.6.

4.4.1 Initial Guess Verification

In section 3.2.3, it was mentioned that Eq. (3.33) was a better initial guess than Eq. (3.27). Now that we have the propeller lifting-line method along with the iterative pitch solver, we can compare these two means of determining the initial guess for the nonlinear solver. Along with these two methods we can use BET as a third method to determine an initial guess. In Tables 4.1 and 4.2 we can compare the number of pitch and nonlinear solver iterations for each initial guess method at advance ratios of 0 and 0.6 respectively. All propeller and solver parameters are the same for these tests with a guess of $C_T = 0.05$ used for Eq. (3.33).

Each pitch iteration requires the solver to recalculate Eq. (3.9), which can be a little computationally taxing to run depending on the total number of control points N and the size and fidelity of the helix integrations used. Each nonlinear solver iteration requires to solve Eq. (3.18) which has the N by N matrix which can become computationally taxing if there is a large number of control points.

Table 4.1: Comparison of the number of pitch and nonlinear solver iterations required to meet pitch and nonlinear solver tolerances of 10^{-5} and 10^{-12} respectively with an initial guess using BET, Eq. (3.27), and Eq. (3.33) at an advance ratio of 0 and using a pitch relaxation factor of 0.65.

Pitch Solver	Iterations for Nonlinear Solver		
Iteration	BET	Eq. (3.27)	Eq. (3.33)
1	17	4	6
2	7	6	4
3	6	6	3
4	4	4	2
5	4	4	2
6	3	3	2
7	2	2	
8	2	2	

For the cases ran in Table 4.1 we can see quite a difference in the number of iterations. Using BET to calculate $\mathbf{\Gamma}$ as an initial guess makes the pitch solver iterate 8 times and causes 45 total iterations for the nonlinear solver. Equation (3.27) requires 8 pitch iterations and 31 nonlinear solver iterations while Eq. (3.33) requires 6 pitch iterations and 19 nonlinear solver iterations. Thus, using Eq. (3.33) greatly reduces the number computations required to solve. This is in part due to this methods ability to also apply a first guess for the pitch as given in Eq. (4.7).

For the cases ran in Table 4.2 the methods are more equal. Using BET as the initial guess method requires 4 pitch iterations and 18 nonlinear solver iterations and Eq. (3.27) requires 4 pitch iterations and 13 nonlinear solver iterations with the same results for Eq. (3.33). So BET requires more computations that the other two methods, with no noticeable

Table 4.2: Comparison of the number of pitch iterations and nonlinear solver iterations required to meet a pitch tolerance of 10^{-5} and the nonlinear solver tolerance of 10^{-12} with an initial guess using BET, Eq. (3.27), and Eq. (3.33) at an advance ratio of 0.6 and using a pitch relaxation factor of 1.

Pitch Solver	Iterations for Nonlinear Solver		
Iteration	BET	Eq. (3.27)	Eq. (3.33)
1	9	4	4
2	4	4	4
3	3	3	3
4	2	2	2

difference between the other two. This could be for a couple different reasons. First, the assumptions that went into Eq. (3.27) hold better at this advance ratio since there are smaller angles of attack across the blades. Second, there is less of a difference between Eqs. (3.3) and (4.7) since V_g is small as compared to V_∞, which also helps with the assumptions that went into Eq. (3.27). So at higher advance ratios, Eqs. (3.27) and (3.33) have little difference in the number of computations required, although Eq. (3.33) could be improved if a better guess for C_T was known.

CHAPTER 5

Results

5.1 Compare to MachUp

Another way to verify the propeller lifting-line method is to compare it to a freeware software MachUp. [16] If the propeller geometry is setup to be similar to a fixed wing, then the results should approach the numerical lifting-line method [16] results at high advance ratios. In order to test this, a propeller geometry consisting of one blade, a constant twist of 100 degrees, and a constant chord length of 0.1 ft was used. The lift coefficient for the airfoils of the blade were the same as that for Figs. 4.1 and 4.2, with no drag for the airfoils. The results for this comparison are shown in Figs. 5.1 and 5.2. To get the equivalent

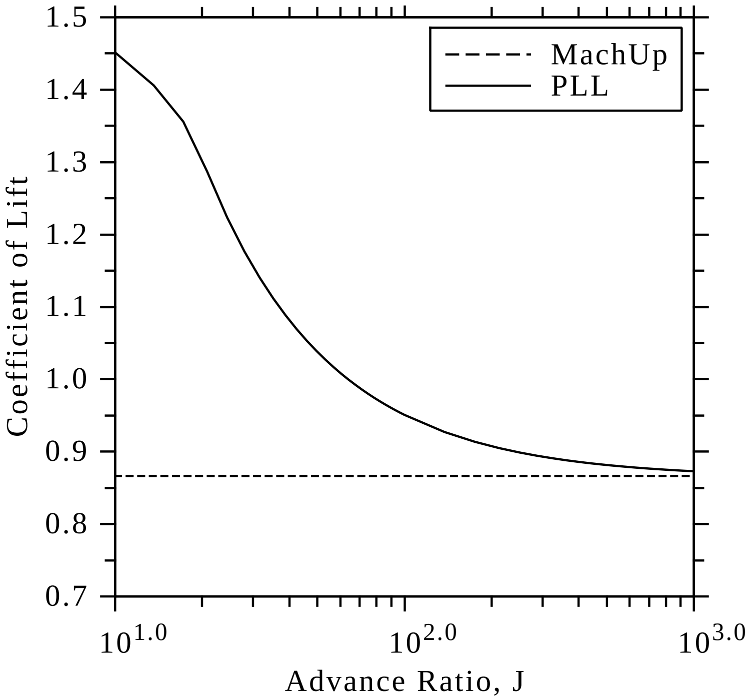


Fig. 5.1: Verification of the propeller lifting-line method at high advance ratios against MachUp. Comparing the coefficient of the vertical force from propeller lifting-line method against the coefficient of lift from MachUp.

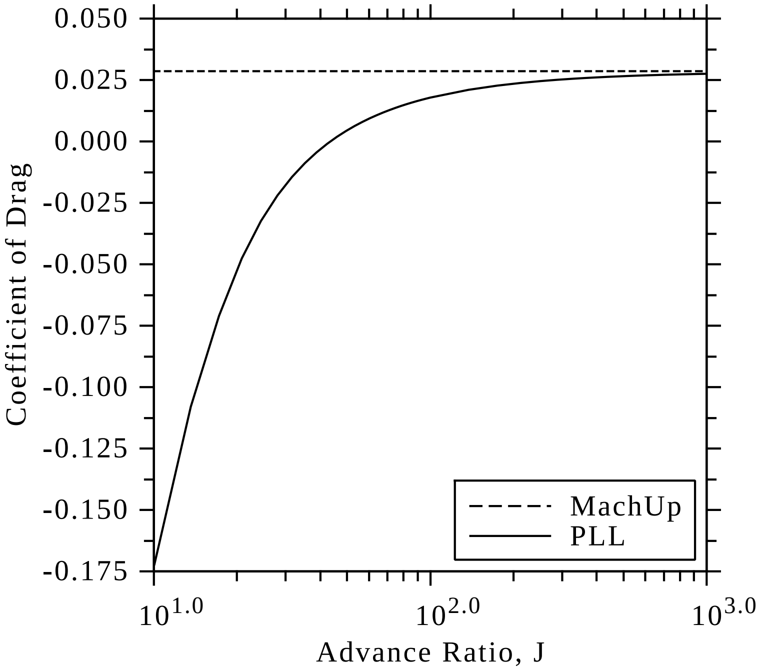


Fig. 5.2: Verification of the propeller lifting-line method at high advance ratios against MachUp. Comparing the coefficient of the axial force from propeller lifting-line method against the coefficient of drag from MachUp.

coefficient of lift and drag from the propeller results, the following expressions were used

$$C_L = \frac{2F_y}{\rho V_\infty^2 (r_p - r_h)c} \tag{5.1}$$

$$C_D = \frac{2F_z}{\rho V_\infty^2 (r_p - r_h)c} \tag{5.2}$$

where c is the chord length of the blade and F_y and F_z are the total aerodynamic forces from Eq. (3.34) in the y and z directions. As can be seen in Figs. 5.1 and 5.2, the values from PLL, converge to those from MachUp as the advance ratio increases.

5.2 Compare Force Loading Distributions

To further evaluate the accuracy and performance of the PLL method with the iterative pitch solver, thrust and drag profiles from this method are compared to results from BET.

For these cases, an airfoil with lift and drag coefficients given by

$$\widetilde{C}_L = -2.5\alpha^2 + \frac{5\pi}{4}\alpha \tag{5.3}$$

$$\widetilde{C}_D = \begin{cases} 0.224\alpha^2 + 0.006 & \alpha \leq 0.25 \\ 16.6944\alpha^2 - 1.0234 & 0.25 < \alpha \leq 0.3 \\ \dfrac{\pi \sin\alpha}{2\cos(0.25)} & 0.3 < \alpha \end{cases} \tag{5.4}$$

were used, where the angle of attack, α, is measured relative to the zero-lift line of the airfoil, which is at -2.1^o. The same twist and chord distributions were used as the case ran in Figs. 4.1 and 4.2 as well as all other propeller parameters. The propeller is right-handed with 2400 revolutions per minute. The results show the thrust and drag distributions across a blade span for various advance ratios in Figs. 5.3-5.24. Drag in this case, refers to the force that opposes the direction of rotation of the propeller and adds to the torque. The results from these two methods agree really well, except at the high and low ends of the propeller advance ratio range.

So far, the results have been for a typical propeller twist distribution as given in Eq. (4.2). It is advantageous to compare the two methods at uncommon twist distributions and advance ratios, since this will test their range of effectiveness in terms of propeller geometries and advance ratios. These cases are shown in Figs. 5.25-5.32 and have the same propeller parameters as the cases in Figs. 5.3-5.24 except where specified differently in Table 5.1. For Table 5.1, r is the spanwise coordinate down the blade with $r = 1$ at the tip and $r = 0$ at the center of the propeller.

It is interesting to note the difference in drag profiles between the two methods at low advance ratios, $J \leq 0.0$. For these cases, there are visually distinct differences in the methods on where the blades are stalled. This is mostly seen in the drag profiles, but it can also be seen in the thrust profiles for the uncommon twist distribution cases, though to a lesser degree. An advance ratio of 0.0 represents the static thrust of an airplane or even

Table 5.1: Summary of the different propeller geometries and run scenarios used for thrust and drag profile comparisons between the propeller lifting-line method and blade element theory.

Figures	Blades	J	Twist Distribution
Figs. 5.25 and 5.26	2	0.0	$\beta = (-50r + 50)\frac{\pi}{180}$
Figs. 5.27 and 5.28	3	0.6	$\beta = (-60r + 70)\frac{\pi}{180}$
Figs. 5.29 and 5.30	4	-0.1	$\beta = (-30r + 40)\frac{\pi}{180}$
Figs. 5.31 and 5.32	8	0.0	$\beta = (-60r + 70)\frac{\pi}{180}$

the hover condition of a rotor craft, while a negative advance ratio represents a rotor craft in descent. These are critical moments of an aircraft's operation. Therefore, a means of accurately predicting the thrust and torque from a propeller at low advance ratios is crucial to the success of the aircraft's mission.

The linear twist distribution also has an effect on the difference between the two methods. From Figs. 5.27 and 5.28 we can see that there are visually significant differences in the thrust and drag profiles, even though this is run at an advance ratio of 0.6. The number of blades can also effect the difference between the two methods, although this isn't really noticed until the number of blades is fairly large and more than what is realistic for an aircraft propeller.

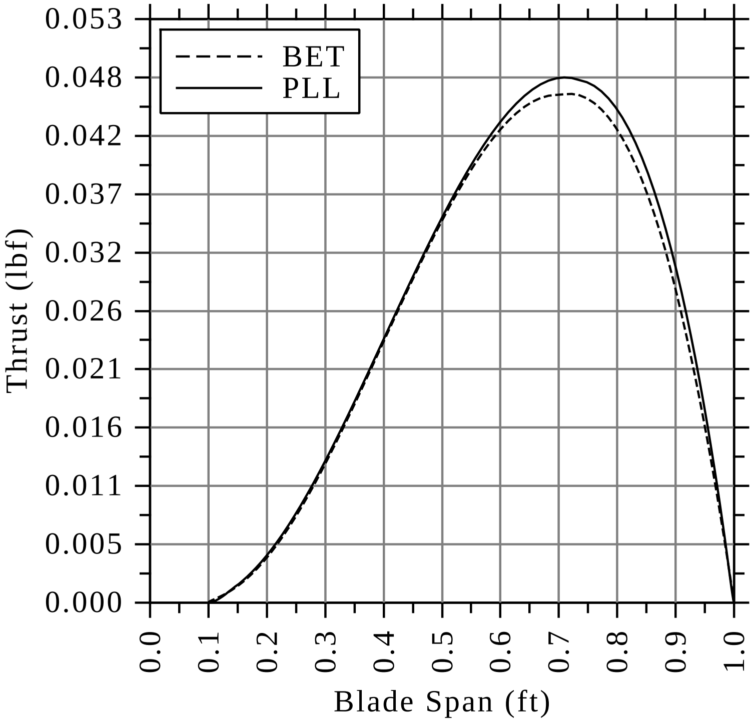


Fig. 5.3: Spanwise thrust loading compared from blade element theory (BET) and propeller lifting-line (PLL) for a 2 bladed propeller at 0.0 advance ratio.

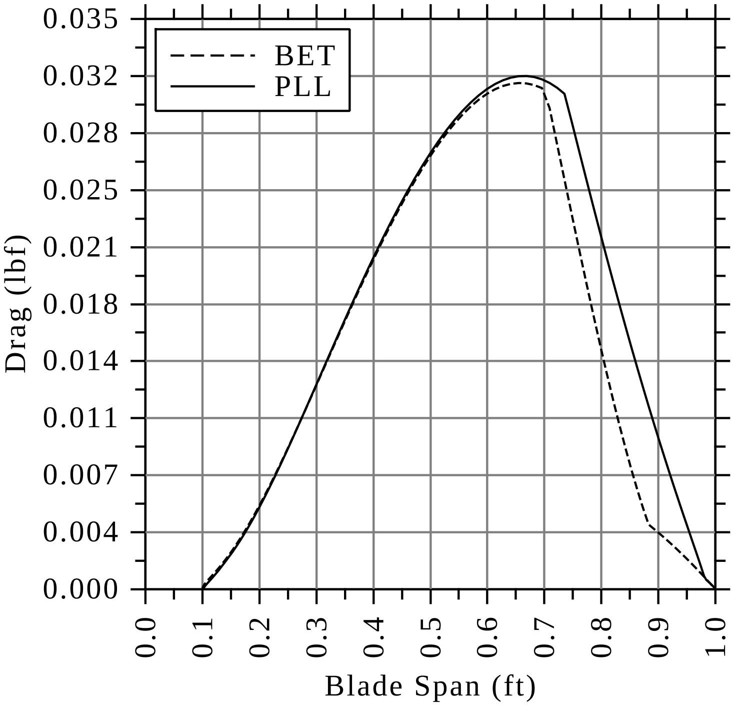


Fig. 5.4: Spanwise drag loading compared from blade element theory (BET) and propeller lifting-line (PLL) for a 2 bladed propeller at 0.0 advance ratio.

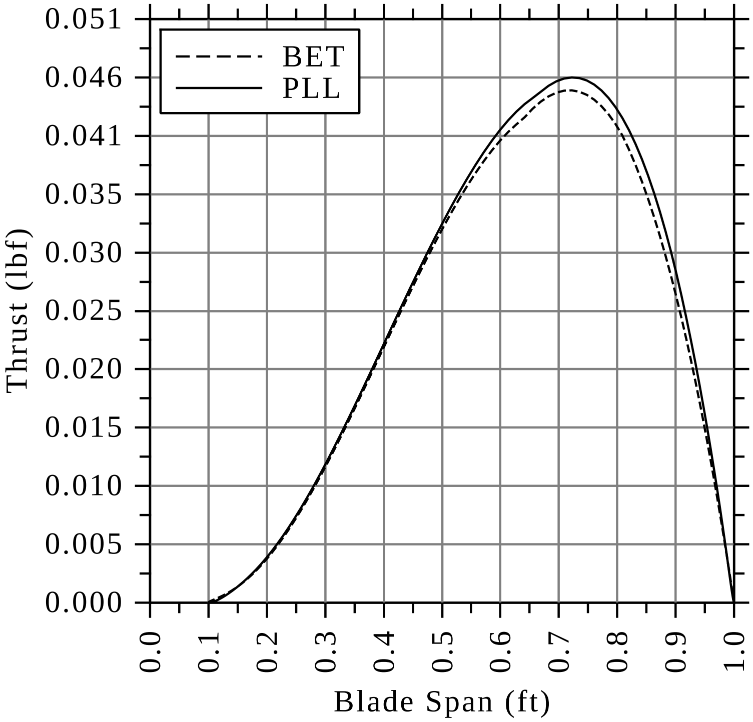


Fig. 5.5: Spanwise thrust loading compared from blade element theory (BET) and propeller lifting-line (PLL) for a 2 bladed propeller at 0.1 advance ratio.

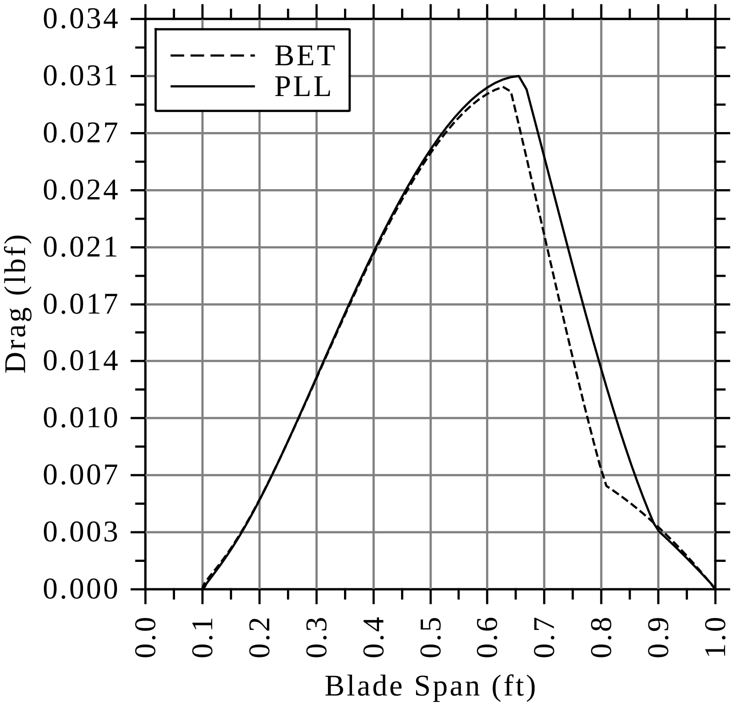


Fig. 5.6: Spanwise drag loading compared from blade element theory (BET) and propeller lifting-line (PLL) for a 2 bladed propeller at 0.1 advance ratio.

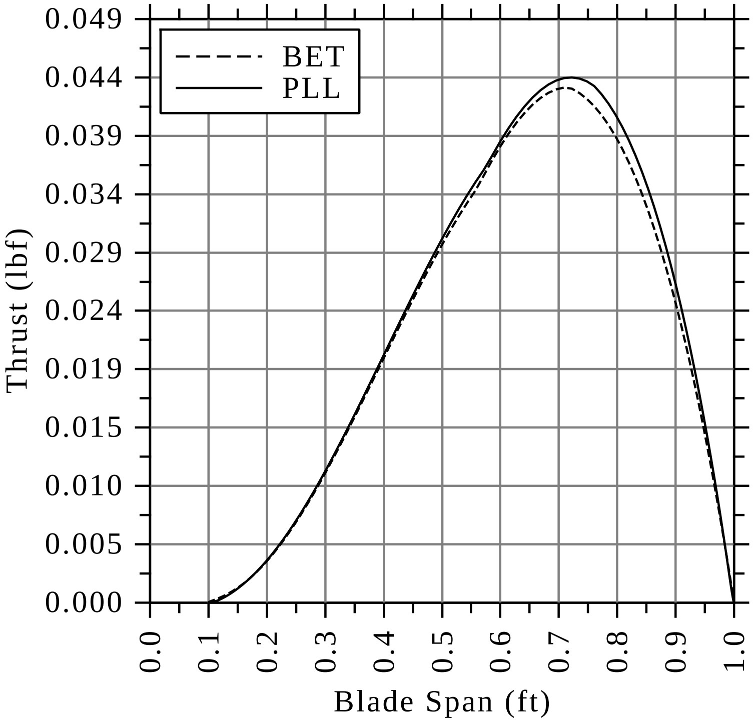


Fig. 5.7: Spanwise thrust loading compared from blade element theory (BET) and propeller lifting-line (PLL) for a 2 bladed propeller at 0.2 advance ratio.

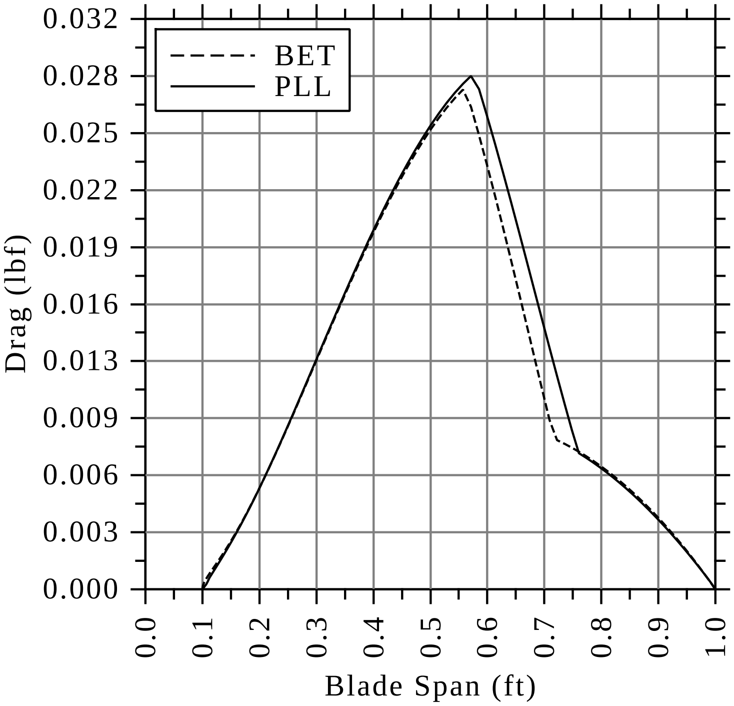


Fig. 5.8: Spanwise drag loading compared from blade element theory (BET) and propeller lifting-line (PLL) for a 2 bladed propeller at 0.2 advance ratio.

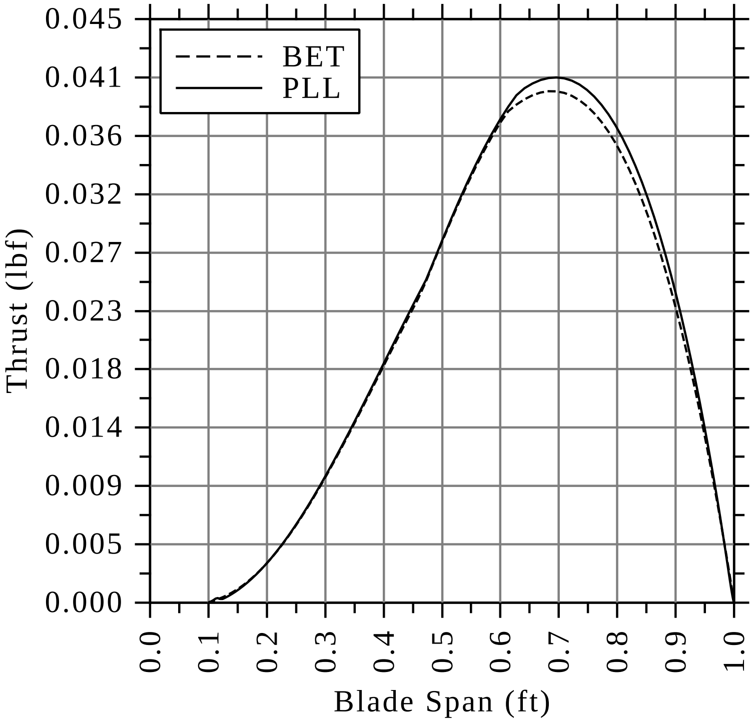


Fig. 5.9: Spanwise thrust loading compared from blade element theory (BET) and propeller lifting-line (PLL) for a 2 bladed propeller at 0.3 advance ratio.

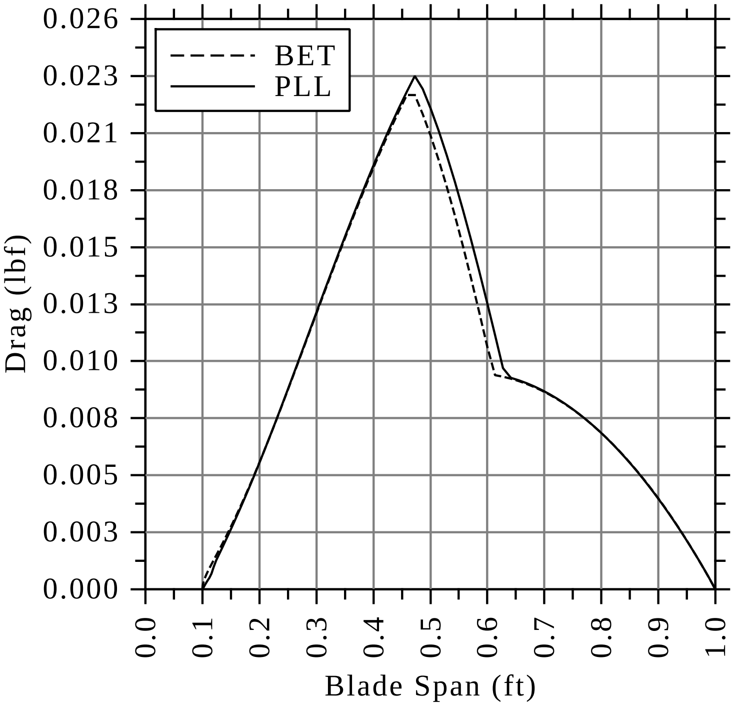


Fig. 5.10: Spanwise drag loading compared from blade element theory (BET) and propeller lifting-line (PLL) for a 2 bladed propeller at 0.3 advance ratio.

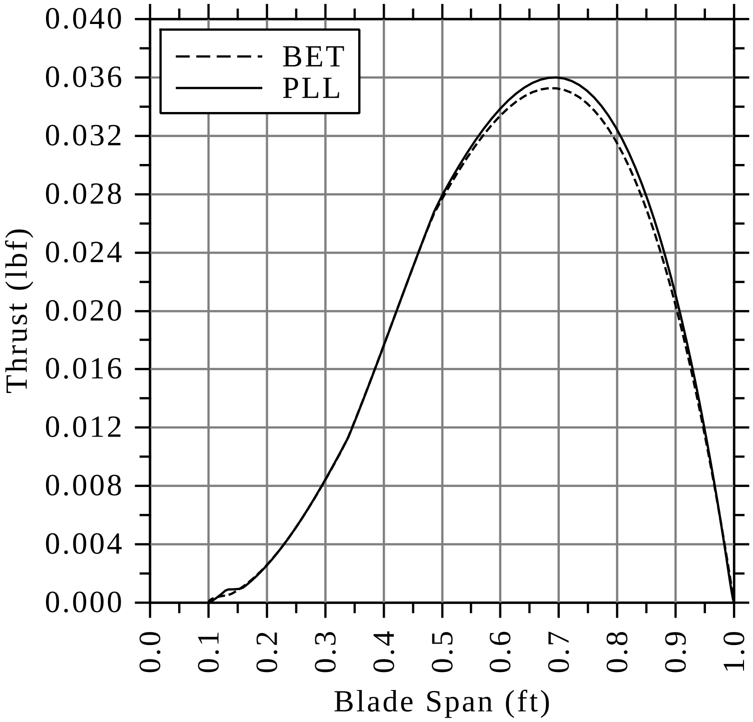


Fig. 5.11: Spanwise thrust loading compared from blade element theory (BET) and propeller lifting-line (PLL) for a 2 bladed propeller at 0.4 advance ratio.

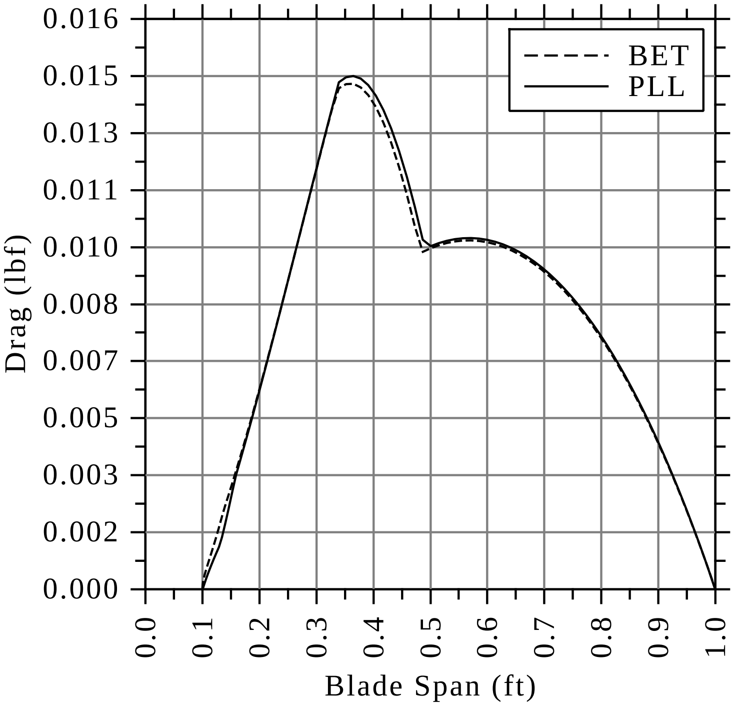


Fig. 5.12: Spanwise drag loading compared from blade element theory (BET) and propeller lifting-line (PLL) for a 2 bladed propeller at 0.4 advance ratio.

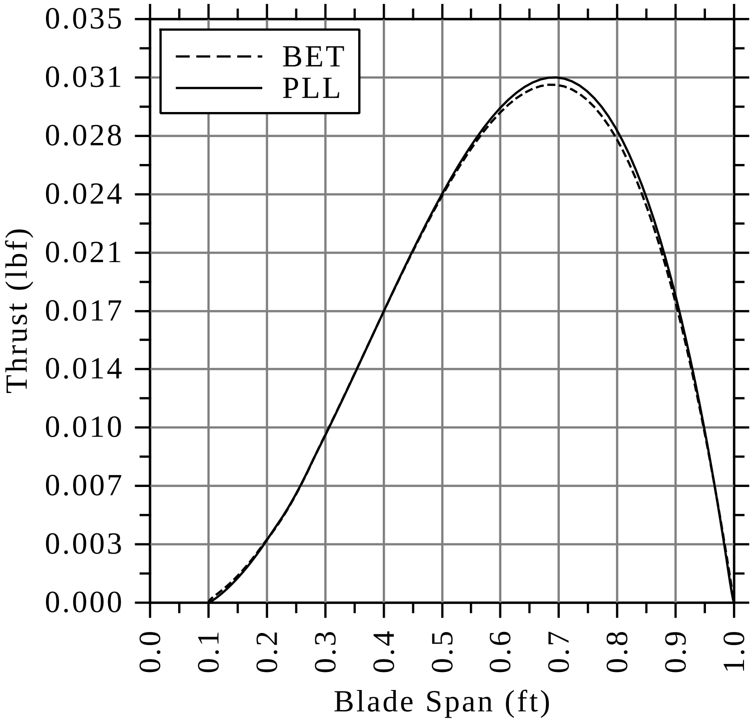


Fig. 5.13: Spanwise thrust loading compared from blade element theory (BET) and propeller lifting-line (PLL) for a 2 bladed propeller at 0.5 advance ratio.

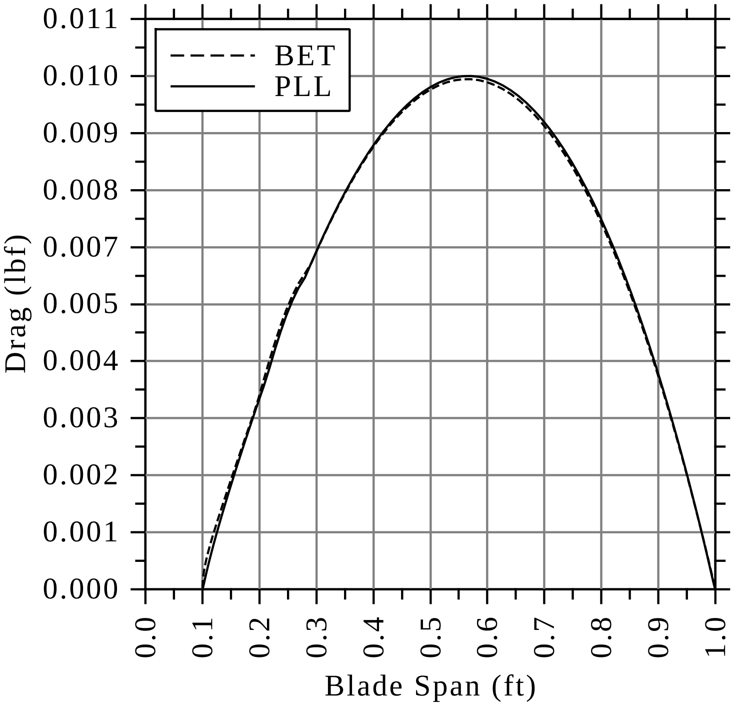


Fig. 5.14: Spanwise drag loading compared from blade element theory (BET) and propeller lifting-line (PLL) for a 2 bladed propeller at 0.5 advance ratio.

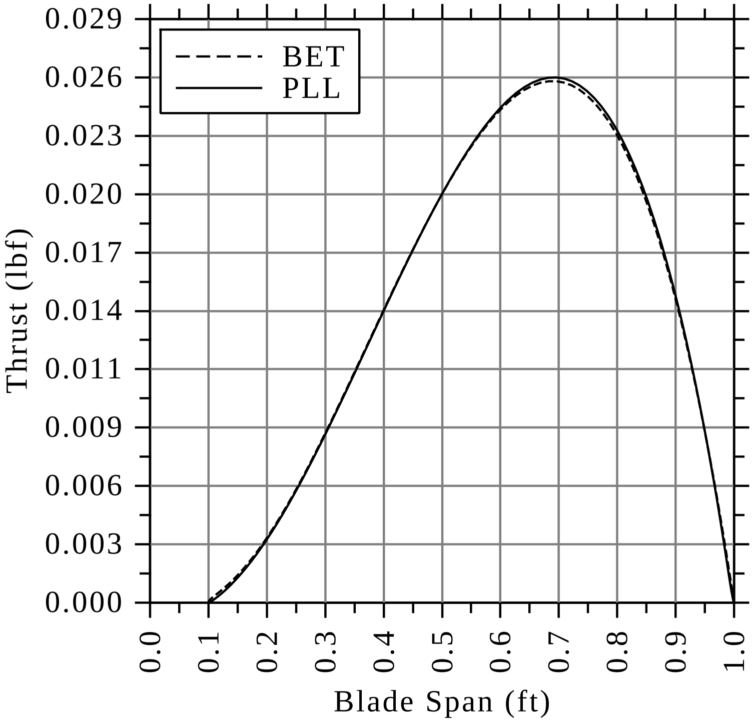


Fig. 5.15: Spanwise thrust loading compared from blade element theory (BET) and propeller lifting-line (PLL) for a 2 bladed propeller at 0.6 advance ratio.

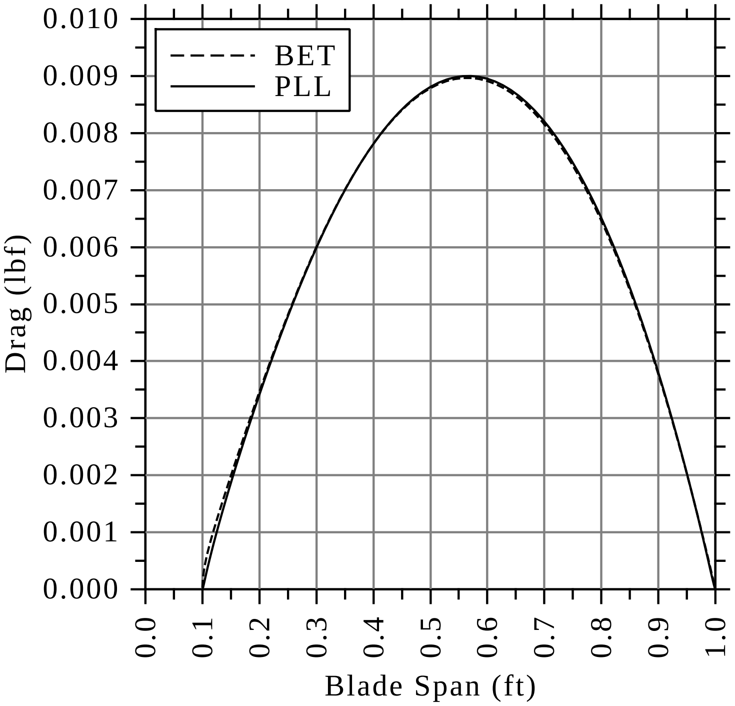


Fig. 5.16: Spanwise drag loading compared from blade element theory (BET) and propeller lifting-line (PLL) for a 2 bladed propeller at 0.6 advance ratio.

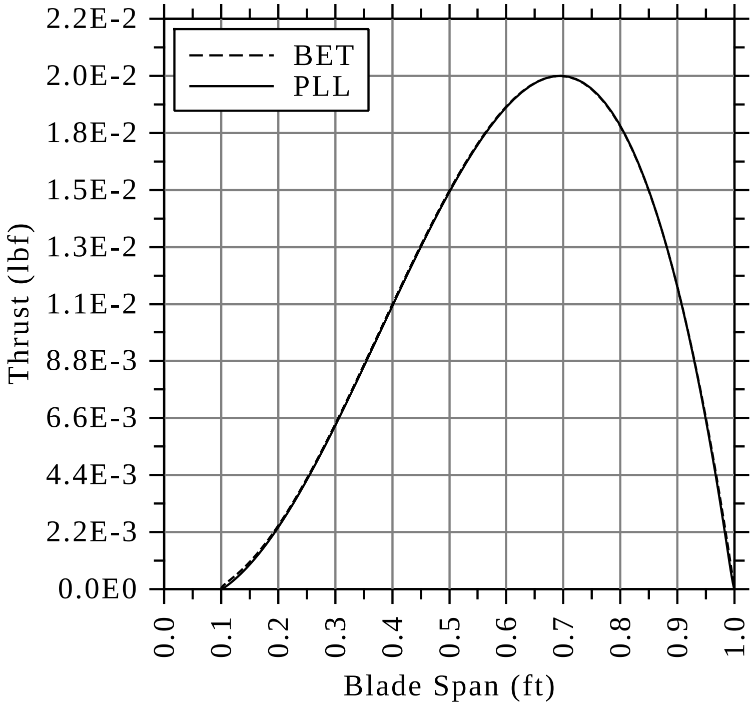


Fig. 5.17: Spanwise thrust loading compared from blade element theory (BET) and propeller lifting-line (PLL) for a 2 bladed propeller at 0.7 advance ratio.

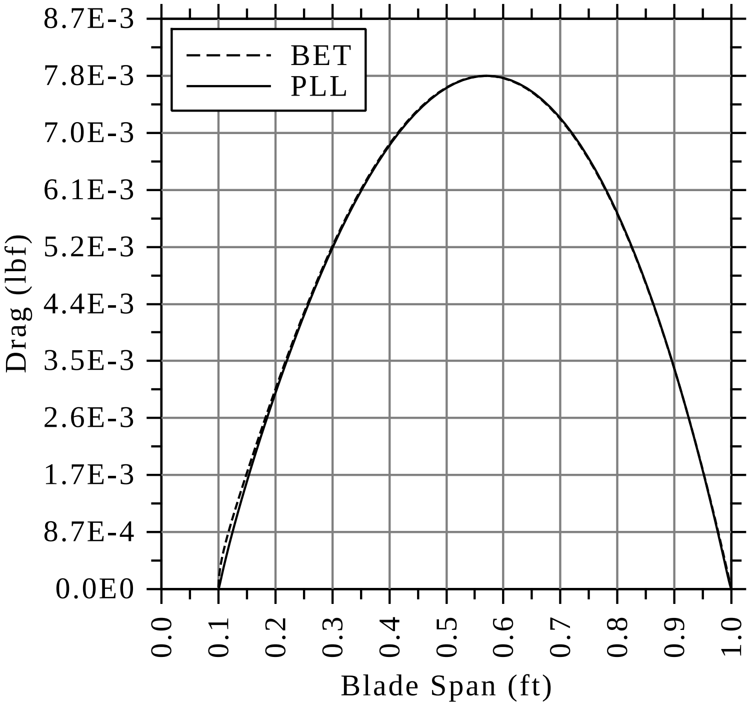


Fig. 5.18: Spanwise drag loading compared from blade element theory (BET) and propeller lifting-line (PLL) for a 2 bladed propeller at 0.7 advance ratio.

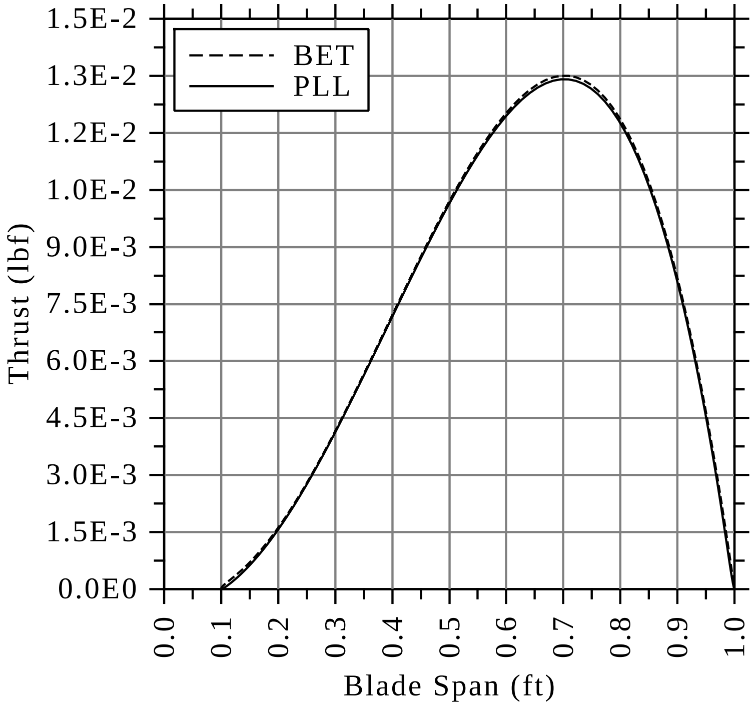


Fig. 5.19: Spanwise thrust loading compared from blade element theory (BET) and propeller lifting-line (PLL) for a 2 bladed propeller at 0.8 advance ratio.

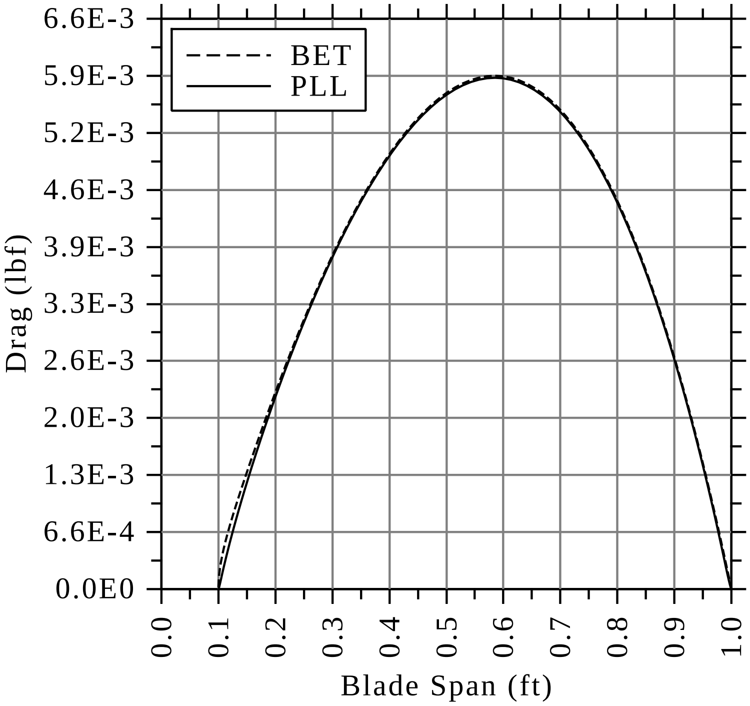


Fig. 5.20: Spanwise drag loading compared from blade element theory (BET) and propeller lifting-line (PLL) for a 2 bladed propeller at 0.8 advance ratio.

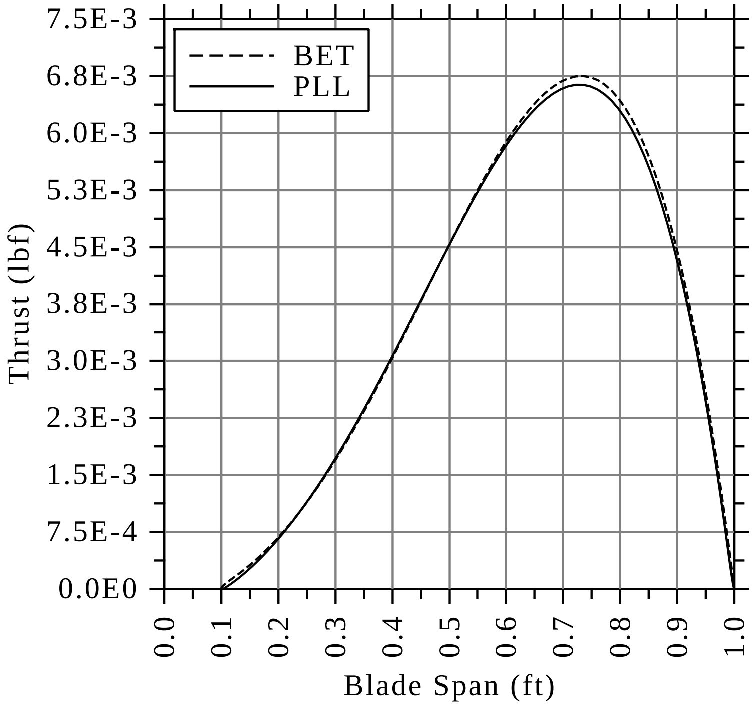


Fig. 5.21: Spanwise thrust loading compared from blade element theory (BET) and propeller lifting-line (PLL) for a 2 bladed propeller at 0.9 advance ratio.

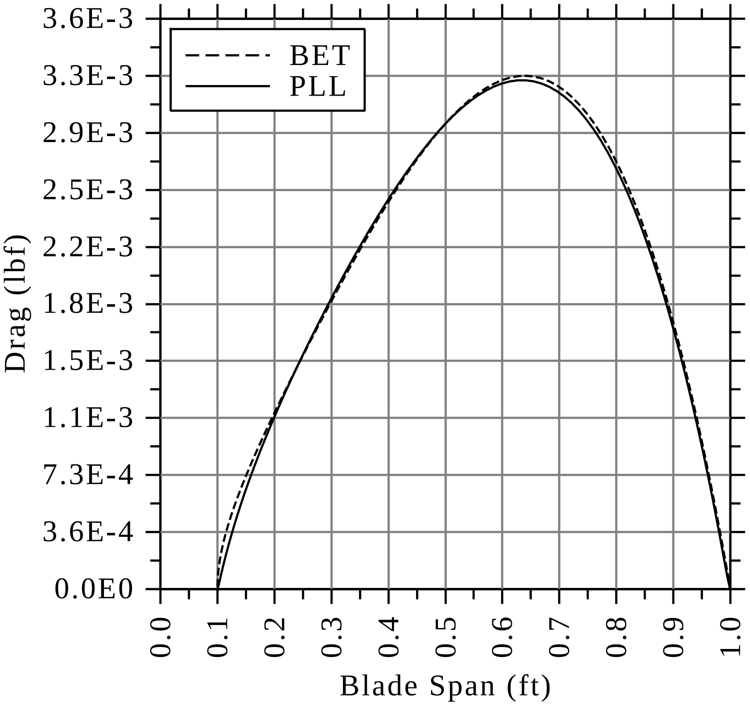


Fig. 5.22: Spanwise drag loading compared from blade element theory (BET) and propeller lifting-line (PLL) for a 2 bladed propeller at 0.9 advance ratio.

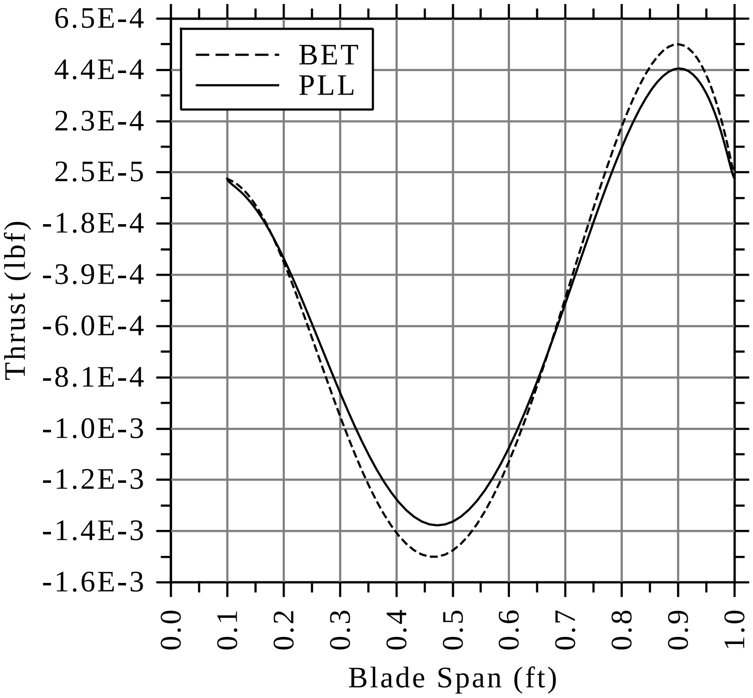


Fig. 5.23: Spanwise thrust loading compared from blade element theory (BET) and propeller lifting-line (PLL) for a 2 bladed propeller at 1.0 advance ratio.

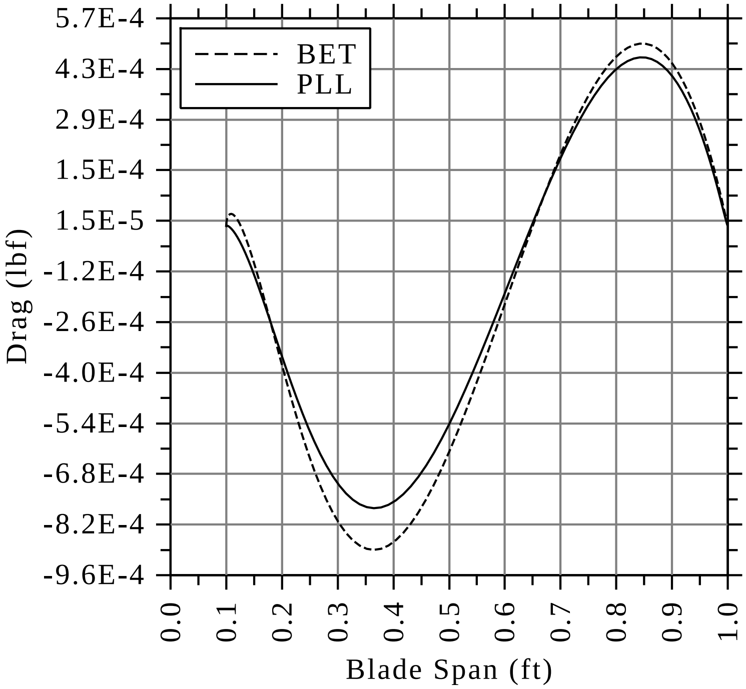


Fig. 5.24: Spanwise drag loading compared from blade element theory (BET) and propeller lifting-line (PLL) for a 2 bladed propeller at 1.0 advance ratio.

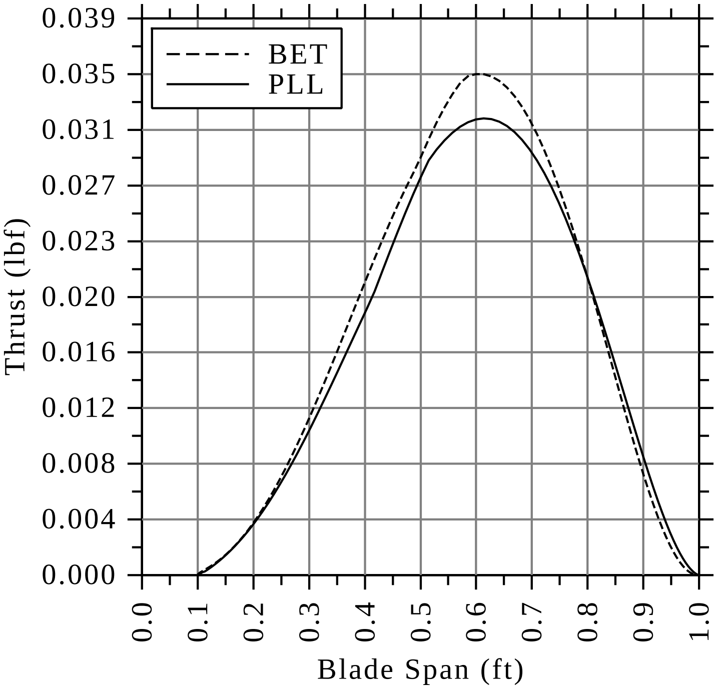


Fig. 5.25: Spanwise thrust loading compared from blade element theory (BET) and propeller lifting-line (PLL) for a 2 bladed propeller at 0.0 advance ratio and a twist distribution given in Table 5.1.

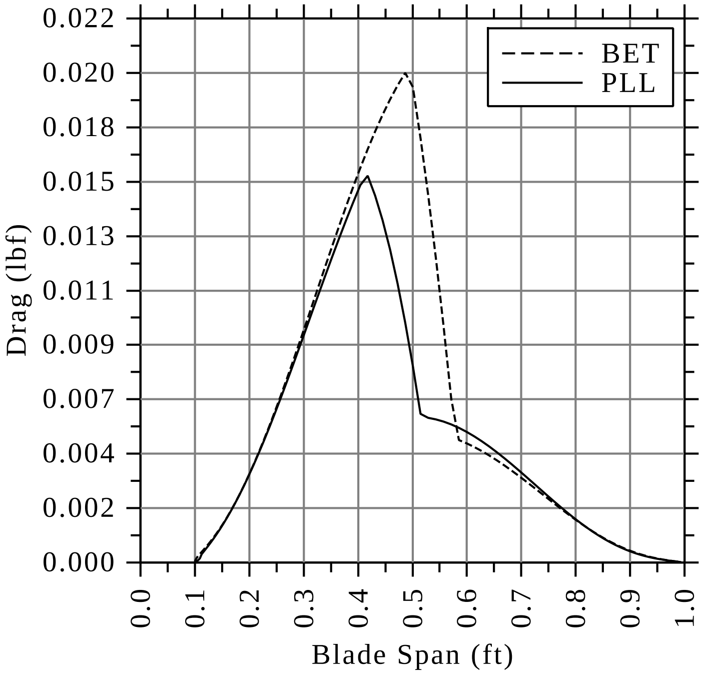


Fig. 5.26: Spanwise drag loading compared from blade element theory (BET) and propeller lifting-line (PLL) for a 2 bladed propeller at 0.0 advance ratio and a twist distribution given in Table 5.1.

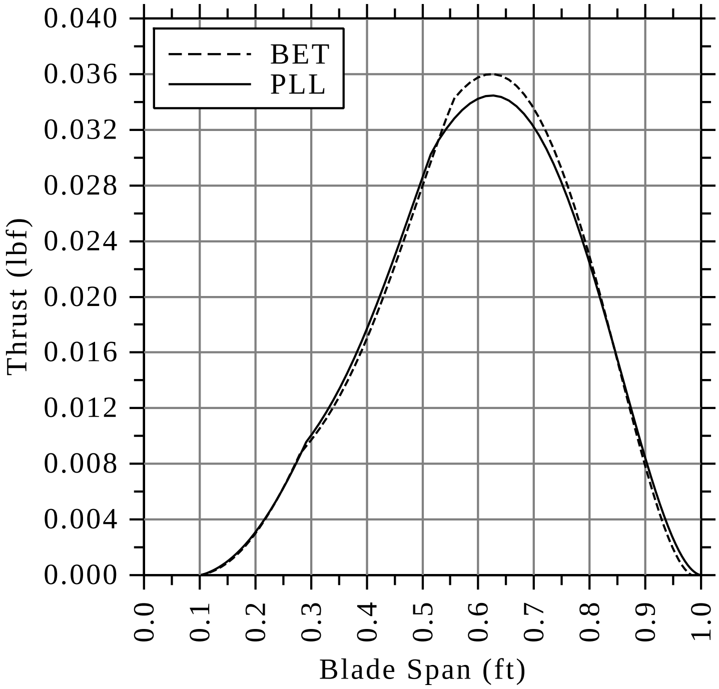


Fig. 5.27: Spanwise thrust loading compared from blade element theory (BET) and propeller lifting-line (PLL) for a 3 bladed propeller at 0.6 advance ratio and a twist distribution given in Table 5.1.

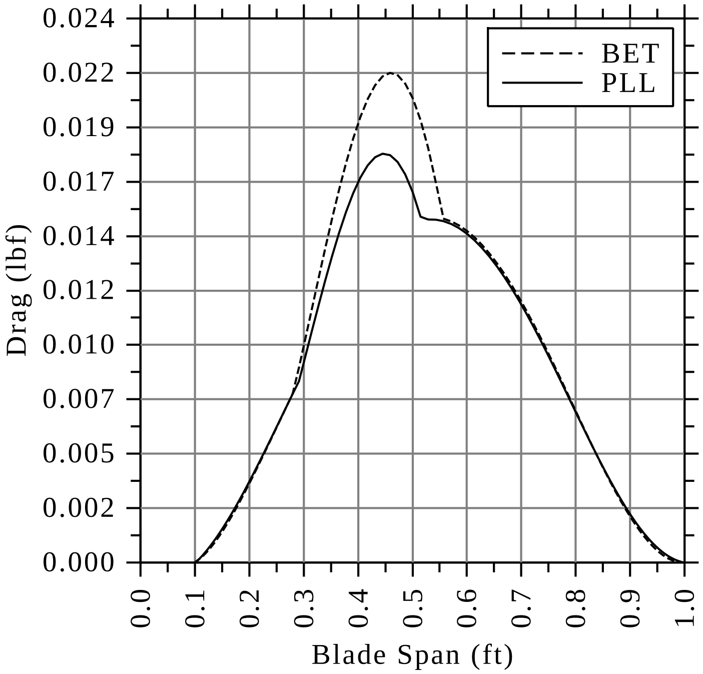


Fig. 5.28: Spanwise drag loading compared from blade element theory (BET) and propeller lifting-line (PLL) for a 3 bladed propeller at 0.6 advance ratio and a twist distribution given in Table 5.1.

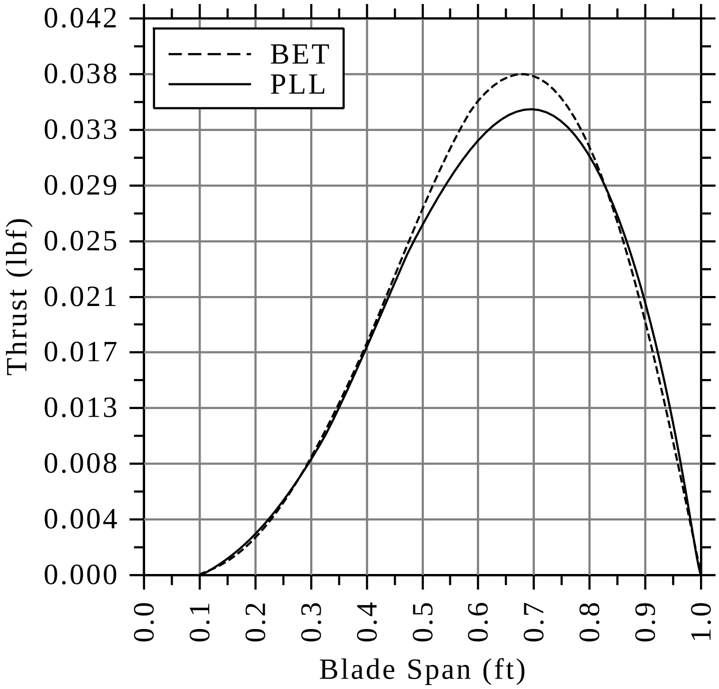


Fig. 5.29: Spanwise thrust loading compared from blade element theory (BET) and propeller lifting-line (PLL) for a 4 bladed propeller at -0.1 advance ratio and a twist distribution given in Table 5.1.

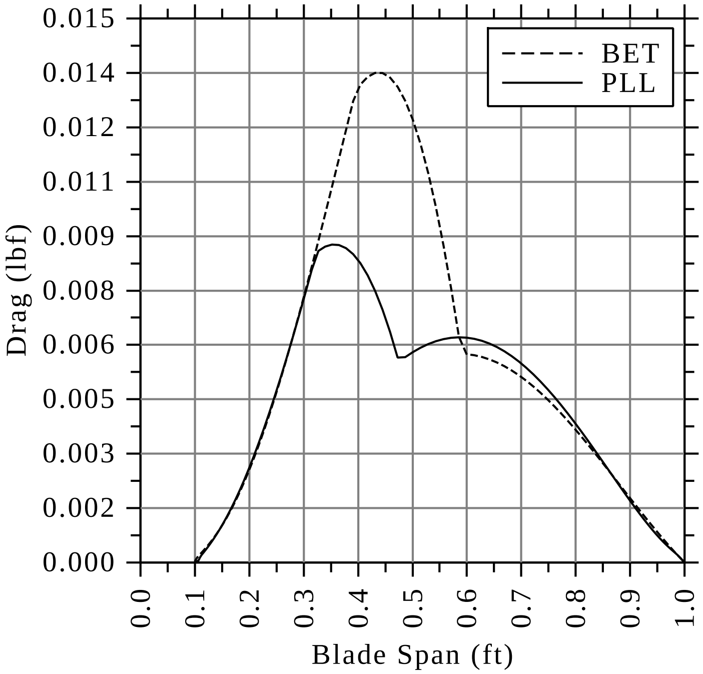


Fig. 5.30: Spanwise drag loading compared from blade element theory (BET) and propeller lifting-line (PLL) for a 4 bladed propeller at -0.1 advance ratio and a twist distribution given in Table 5.1.

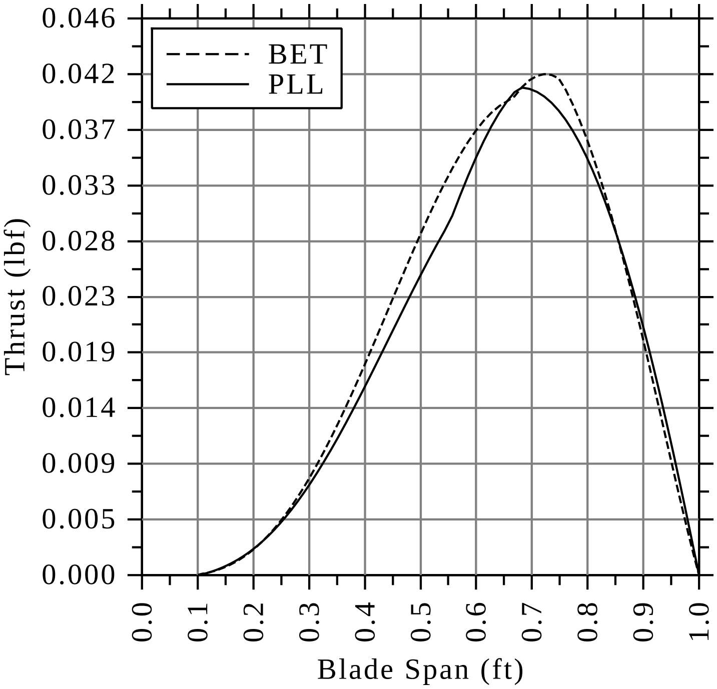


Fig. 5.31: Spanwise thrust loading compared from blade element theory (BET) and propeller lifting-line (PLL) for a 8 bladed propeller at 0.0 advance ratio and a twist distribution given in Table 5.1.

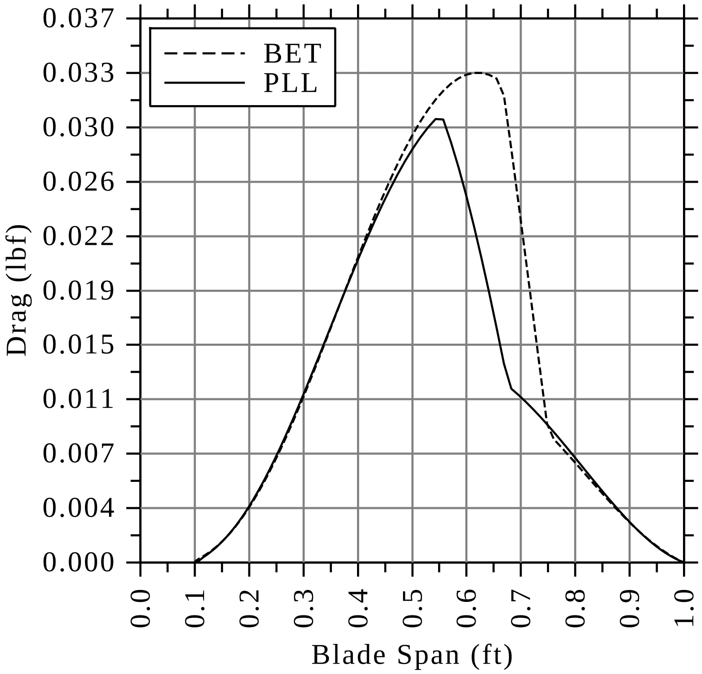


Fig. 5.32: Spanwise drag loading compared from blade element theory (BET) and propeller lifting-line (PLL) for a 8 bladed propeller at 0.0 advance ratio and a twist distribution given in Table 5.1.

CHAPTER 6

Conclusion

Aerodynamic propeller analysis codes come in various forms, from low to high fidelity models. Some allow any propeller geometry, while others are very limiting. Higher fidelity codes are usually very computationally expensive compared to lower fidelity models. The propeller lifting-line method provides a good balance as a mid-range model, allowing for any propeller geometry while not being extremely computationally expensive like CFD models.

Fundamental to the propeller lifting-line model, or any propeller analysis code, is the treatment of the wake behind the propeller. Here, a method is presented of calculating the induced velocity from a helical vortex segment, Eq. (2.13), that is much more accurate than the most common alternative method, Eq. (2.37). This allows for greater accuracy when calculating the lifting-line solution.

There were several challenges in developing the numerical lifting-line approach for propellers that have been addressed. Since propeller analysis codes need to be performed over a wide range of advance ratios, the lifting-line approach needed to be able to handle the static condition of 0 freestream velocity without encountering any mathematical singularities. Because of this, typical nondimensional forms of fixed wing lifting-line techniques were avoided and the lifting-line approach for propellers is dimensional. Another challenge faced was coming up with a reasonable initial guess for the nonlinear Newton's solver. Since propeller blades are more likely to experience stall in normal operating conditions than fixed wings, it is important that the initial guess be able to handle those conditions. It is also crucial for the propeller lifting-line method that the airfoil data be well defined for regions of stall and high angle of attack.

At high blade loadings, which usually occur at lower advance ratios, there are large induced velocities within the slipstream of the propeller. This causes issues in accurately defining the trailing vortex geometry, mainly in terms of pitch. This is corrected by using an iterative pitch solver, which iterates on the propeller lifting-line method until the induced velocities at the propeller disk are in agreement with the pitch length of the trailing helical vortex system. Doing so causes the propeller lifting-line method to change from a prescribed wake model to a semi-free wake model, since the pitch is free to change at the propeller disk. This resolves issues at low advance ratios in defining the trailing vortex geometry.

The propeller lifting-line method combined with the iterative pitch solver is in close agreement with blade element theory over a wide range of advance ratios and propeller geometries. The results from the two methods deviate somewhat from one another at low and high advance ratios for a given propeller. This is distinctly noticeable at lower advance ratios, when the blades are stalled. Overall, the two methods are in real close agreement.

However, the presented work allows for additional propeller geometries that blade element theory does not, including swept and raked blades as well as asymmetrically spaced blades. Additionally, blade element theory [10] as used for the results presented must have a twist angle at the blade tip greater than 0, or the analysis will fail due to singularities. A twist angle less than or equal to 0 at the blade tip would not normally be encountered, but it is an additional restriction on blade element theory.

Potential future work that could be done to the propeller lifting-line method could be to incorporate the ability to have an off axis propeller, which is one that isn't pointed directly into the freestream. Due to the way that this method was derived, it could potentially be combined with the method presented by Phillips and Snyder [3] to have a single lifting-line method that works for propeller and fixed-wing combinations that could account for the effects of the propeller on the fixed wing and vice versa.

A main drawback from this method is that the run time is significantly greater than that for blade element theory. However, the propeller lifting-line method is still significantly better than CFD algorithms with respect to run time. It was expected that the current method would provide more accurate results than blade element theory, and it likely does, especially at low advance ratios. However, for most common geometries and conditions, the differences between the two methods appear to fall within the accuracy to which we can measure propeller performance.

REFERENCES

[1] Prandtl, L., "Tragflugel Theorie," *Nachrichten von der Gesellschaft der Wisseschaften zu Gottingen*, Geschaeftliche Mitteilungen, Klasse, 1918, pp. 451-4773

[2] Anderson, J. D., "Incompressible Flow over Finite Wings," *Fundamentals of Aerodynamics*, 5th ed., McGraw-Hill, New York, 2011, pp. 411-444.

[3] Phillips, W. F. and Snyder, D. O., "Modern Adaptation of Prandtl's Classic Lifting-Line Theory," *Journal of Aircraft*, Vol. 37, No. 4, 2000, pp. 662-670

[4] Phillips, W. F., *Mechanics of Flight*, 2nd ed., Wiley, New Jersey, 2010, pp. 25-107

[5] Katz, J., and Plotkin, A., "Lifting-Line Solution by Horseshoe Elements," *Low-Speed Aerodynamics, from Wing Theory to Panel Methods*, McGraw-Hill, New York, 1991, pp. 379-386

[6] Drzewiecki, S., "Des Helices propulsives," *Congres d'Architecture et de Constructions Navales*, Paris, 1900

[7] Goldstein, S., "On the Vortex Theory of Screw Propellers," *Proceedings of the Royal Society*, Vol. A123, No. 792, 1929, pp. 440-465

[8] Glauert, H., "The Airscrew: Blade Element Theory," *The Elements of Aerofoil and Airscrew Theory*, 2nd ed., Cambridge Univ. Press, Cambridge, England, U.K., 1992, pp. 208-221.

[9] McCormick, B. W., *Aerodynamics, Aeronautics and Flight Mechanics*, 2nd ed., Wiley, New York, 1995,

[10] Phillips, W. F., *Mechanics of Flight*, 2nd ed., Wiley, New Jersey, 2010, pp. 173-187

[11] Seddon, J. and Newman S., *Basic Helicopter Aerodynamics*, 3rd ed., John Wiley & Sons, Chichester, U.K., 2011, pp. 63-78.

[12] Morgado, J., Silvestre, M. A. R., and Pascoa, J. C., "Validation of New Formulations for Propeller Analysis," *Journal of Propulsion and Power*, Vol. 31, No. 1, 2015, pp. 467-477

[13] Tremmel, M., Taulbee, D. B., and Sonnenmeier, J. R., "Numerical Determination of Circulation for a Swept Propeller," *Journal of Aircraft*, Vol. 38, No. 6, November-December 2001, pp. 1085-1092

[14] Murray, J. C. and Barone, M., "The Development of CACTUS, a Wind and Marine Turbine Performance Simulation Code," *AIAA Aerospace Sciences Meeting including the New Horizons Forum and Aerospace Exposition*, Vol 49, No. 147, 2011

[15] Hunsaker, D. F., ”A Numerical Vortex Approach to Aerodynamic Modeling of SUAV/VTOL Aircraft,” Master’s Thesis, Department of Mechanical Engineering, Brigham Young University, Provo, 2007, pp. 66-69.

[16] Hodson, J., Hunsaker, D. F., and Spall, R., ”Wing Optimization Using Dual Number Automatic Differentiation in MachUp,” *55th AIAA Aerospace Sciences Meeting, AIAA SciTech Forum*, 2017

APPENDIX

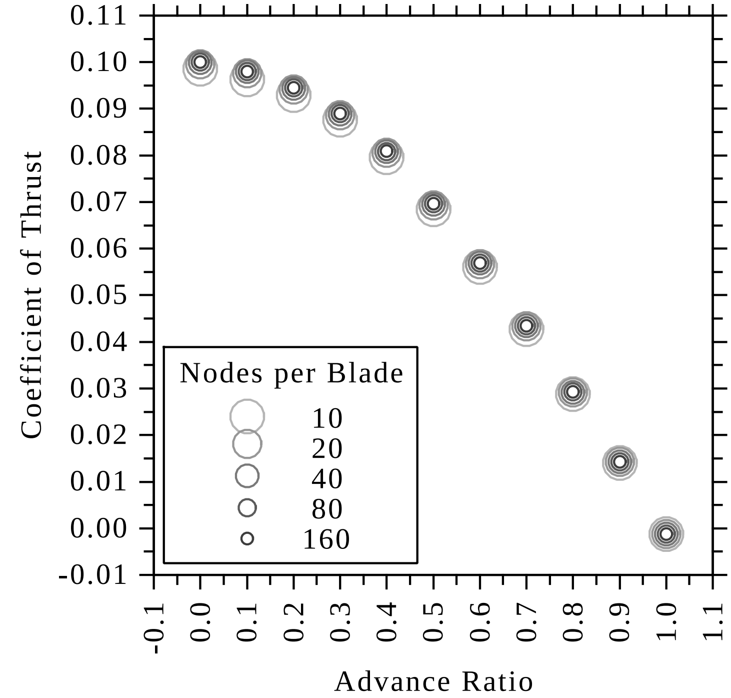


Fig. A.1: Grid Resolution of the Coefficient of Thrust versus Advance Ratio calculated using BET, comparable to the results in Figs. 4.1 and 4.5.

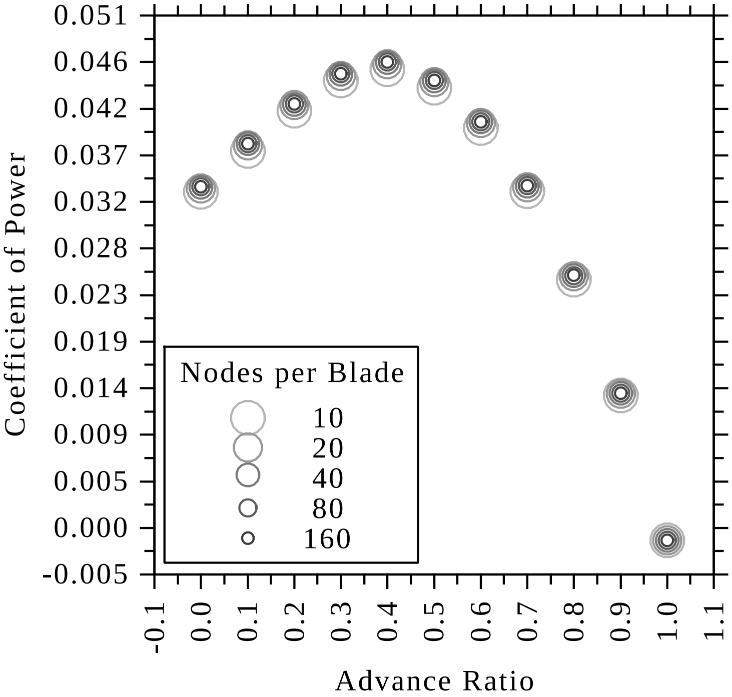


Fig. A.2: Grid Resolution of the Coefficient of Power versus Advance Ratio calculated using BET, comparable to the results in Figs. 4.2 and 4.6.

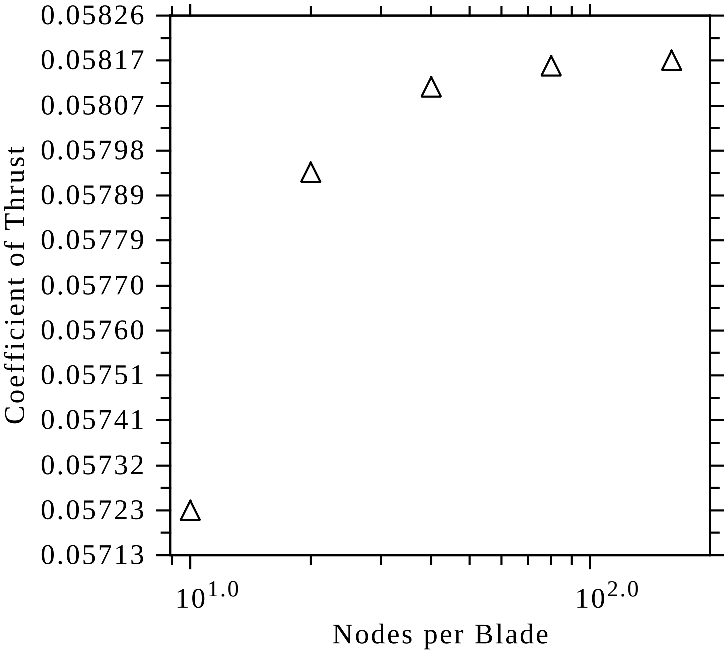


Fig. A.3: Coefficient of Thrust values vs the nodes per blade calculated by BET at an advance ratio of 0.6, comparable to Fig. 4.9.