

# A MULTIROTOR VEHICLE PERFORMANCE PREDICTION METHOD

by

Issi-Rae Emanuelle George

Bachelor of Engineering, The University of Western Ontario, 2014

A thesis

presented to Ryerson University

in partial fulfillment of the  
requirements for the degree of  
Master of Applied Science  
in the Program of  
Aerospace Engineering

Toronto, Ontario, Canada, 2017

©Issi-Rae Emanuelle George 2017

## **AUTHOR'S DECLARATION FOR ELECTRONIC SUBMISSION OF A THESIS**

I hereby declare that I am the sole author of this thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I authorize Ryerson University to lend this thesis to other institutions or individuals for the purpose of scholarly research.

I further authorize Ryerson University to reproduce this thesis by photocopying or by other means, in total or in part, at the request of other institutions or individuals for the purpose of scholarly research.

I understand that my dissertation may be made electronically available to the public.

# **A MULTIROTOR VEHICLE PERFORMANCE PREDICTION METHOD**

**Issi-Rae Emanuelle George**

Master of Applied Science, Aerospace Engineering, Ryerson University (2017)

## **Abstract**

A method has been developed to predict the performance of small multirotor vehicles. Using the vehicle geometry, rotor geometry and flight conditions as inputs, the aerodynamic performance is found through an interpolation method using tabulated rotor performance data. The model is able to predict performance in hover and forward flight, and can produce results quickly and easily, making the prediction model a powerful tool. The vehicle performance prediction model also includes a wake interference model that captures the effect of rotors and their shed wakes on others rotors in the vicinity. When compared to flight test data, the method shows good agreement when predicting the angle of attack, rotational velocity and power requirements of the vehicle. The effect of the vehicle orientation on the performance of the vehicle was investigated showing that during fast forward flight, the vehicle requires about 5% less power in a diamond configuration than in a square configuration.

## Acknowledgements

I would like to thank my parents, siblings and friends for their love and support. Without your unending encouragement and faith in my ability I would not have been able to complete this work. Thank you to the professors and staff in the engineering department at Ryerson University, especially Jerry Karpynczyk. Without your hardwork, the wind tunnel and test stand project would not have moved forward as swiftly as it had.

To the FSAE team at my alma mater, the University of Western Ontario: I would not have had the work ethic and perseverance to complete this project without the long nights spent at the shop. To Jeff Mock, thank you for encouraging me to take a leap and start this degree. To my colleagues at the Ryerson Applied Aerodynamics Laboratory of Flight (RAALF), thank you for lending an ear when I needed one, for the invaluable advice and the laughs. Thank you to Nafisa, who believed in me.

I would like to acknowledge Aeryon Labs Inc. for their financial and expert support, especially Brandon Pinkey, Michael Tribou and Mike Peasgood. Your experience in the field greatly aided this project. Thank you to Tim Carroll for allowing me to use your work in this project. Thank you to the Ontario Centres of Excellence and NSERC which made this research possible.

Finally, I would like to acknowledge my supervisor, Dr. Goetz Bramesfeld for organizing this work and putting your trust in me to take it on. Thank you for introducing me to the field of rotorcraft, for your technical guidance, professional development and constant support through the program.

# Table of Contents

<i>Declaration</i> . . . . .	ii
<i>Abstract</i> . . . . .	iii
<i>Acknowledgements</i> . . . . .	iv
<i>List of Tables</i> . . . . .	vii
<i>List of Figures</i> . . . . .	viii
<i>List of Appendices</i> . . . . .	x
<i>List of Symbols</i> . . . . .	xi
<b>1 Introduction</b>	<b>1</b>
1.1 Multirotor Small Unmanned Aerial Systems . . . . .	1
1.2 Multirotor sUAS Aerodynamics . . . . .	3
1.3 Aerodynamic Performance Analysis Methods . . . . .	5
1.4 Contribution . . . . .	6
<b>2 Multirotor Vehicle Performance</b>	<b>8</b>
2.1 Equations of Motion . . . . .	9
2.2 Multirotor Vehicle Thrust . . . . .	10
2.3 Multirotor Vehicle Power . . . . .	11
2.3.1 Parasitic Power . . . . .	12
2.3.2 Rotor Power . . . . .	13
2.3.3 Interference Power . . . . .	14
2.4 Rotor Thrust and Power . . . . .	16
2.5 Advancing/Retreating Blade . . . . .	19

<b>3</b>	<b>Multirotor Vehicle Performance Model</b>	<b>22</b>
3.1	Methodology . . . . .	22
3.2	Parasitic Drag Module . . . . .	25
3.3	Rotor Module . . . . .	31
3.3.1	Table Lookup Method . . . . .	31
3.3.2	Rotor Aerodynamic Performance Prediction . . . . .	34
3.3.3	Experimental Rotor Performance . . . . .	39
3.3.4	Comparisons with Other Experimental Facilities . . . . .	41
3.4	Wake Interference Module . . . . .	42
<b>4</b>	<b>Multirotor Vehicle Performance Model Validation</b>	<b>48</b>
<b>5</b>	<b>Case Studies</b>	<b>52</b>
5.1	Effect of Altitude . . . . .	52
5.2	Effect of Weight . . . . .	56
5.3	Effect of Orientation in Forward Flight . . . . .	58
<b>6</b>	<b>Conclusions</b>	<b>62</b>
	<b>Appendices</b>	<b>64</b>
	<b>Bibliography</b>	<b>105</b>

# List of Tables

3.1	Dimensions of SkyRanger components. . . . .	30
3.2	Values for determining Reynolds number for components of Aeryon Skyranger. . . . .	30
3.3	Predicted Reynolds number, parasitic drag coefficient and parasitic drag for Aeryon-Skyranger components. . . . .	31
3.4	Solution time for studies in Figs.3.18 b-d . . . . .	45

# List of Figures

1.1	Two types of small unmanned aerial systems. . . . .	1
1.2	Depiction of standard quadrotor configuration[4]. . . . .	2
1.3	Wake geometry of rotor blade in hover [14]. . . . .	5
2.1	Free-body diagram of sUAS in forward flight. . . . .	9
2.2	Total power required and power decomposition for the lead rotor of a quadcopter in forward flight. . . . .	12
2.3	Effect of induced velocity on rotor airfoil section[23]. . . . .	14
2.4	Induced velocity vectors produced by rotor in hover. . . . .	15
2.5	Power required for coaxial and single rotor configuration[17]. . . . .	15
2.6	Power required for coaxial and single rotor configuration [17]. . . . .	16
2.7	Experimental performance coefficients for ASP130 rotor taken in the Ryerson University Large Subsonic Wind Tunnel. . . . .	18
2.8	Velocities acting on element of rotating blade at azimuth angle $\psi$ [19]. . . . .	19
2.9	Flow diagrams for advancing and retreating blade conditions. . . . .	20
2.10	Normal force coefficient versus angle of attack [24]. . . . .	21
3.1	Performance model flow chart . . . . .	23
3.2	Vehicle orientation and rotor numbering convention. . . . .	24
3.3	Aeryon Skyranger with labeled components: A) centre body B) rotor arms C) landing gear[1]. . . . .	25
3.4	Drag coefficient versus Reynolds number for sphere and cylinder[19]. . . . .	26
3.5	Aeryon Scout body mounted on balance. . . . .	28



3.6	Lift and drag coefficients measured in wind tunnel for Aeryon Scout body. . . . .	29
3.7	Dynamic pressure versus thrust per density for a rotor at 0 degrees angle of attack. . . . .	32
3.8	Algorithm for rotor performance interpolation method. . . . .	33
3.9	Rotor sectional element with acting velocities and forces[19]. . . . .	35
3.10	Predicted performance coefficients for MA11x7 rotor $\alpha = 0$ deg. . . . .	37
3.11	Predicted performance coefficients for MA11x7 rotor $\alpha = 30$ deg. . . . .	38
3.12	Ryerson University Large Subsonic Wind Tunnel drawing[26]. . . . .	39
3.13	Test stand angle convention for propeller testing. . . . .	40
3.14	Propeller performance testing apparatus with labeled components. . . . .	41
3.15	Comparison of experimental and predicted MA11x7 rotor performance at 90 degrees. . . . .	42
3.16	Vortex rings representing a series of wakes shed by rotor. . . . .	42
3.17	Geometry of skewed wake. . . . .	44
3.18	Wake interference model results comparison with analytical results. . . . .	46
4.1	Comparison of predicted and measured vehicle angle of attack versus forward velocity. . . . .	49
4.2	Comparison of predicted and measured rotational velocity versus angle of attack. . . . .	50
4.3	Comparison of predicted and measured power versus angle of attack. . . . .	51
5.1	Experimental performance prediction for Aeryon SkyRanger using T-Motor rotors at 1000 ft. . . . .	54
5.2	Rotor power predictions versus forward flight speed at various altitudes for T-Motor rotor. . . . .	55
5.3	Thrust per rotor versus forward flight speed for sUAS with three different flight masses. . . . .	56
5.4	Angle of attack versus forward flight speed for sUAS with three different flight masses. . . . .	57
5.5	Force triangle depicting effect of weight on vehicle angle of attack. . . . .	57
5.6	Rotor power versus forward flight speed for sUAS with payloads and empty weight. . . . .	58
5.7	Total power required for diamond and square configuration using T-Motor rotor experi- mental data. . . . .	59
5.8	Rotor power required and interference power for diamond and square configuration using T-Motor rotor experimental data. . . . .	61

# List of Appendices

<b>A</b>	<b>Wind Tunnel Data</b>	<b>64</b>
A.1	Uncertainty Analysis . . . . .	64
A.2	Experimental Procedure . . . . .	65
A.3	Data Reduction . . . . .	66
<b>B</b>	<b>Rotor Performance Tables</b>	<b>69</b>
<b>C</b>	<b>MATLAB Scripts</b>	<b>74</b>

# List of Symbols

$\alpha$	Vehicle angle of attack (deg)
$\Gamma$	Blade circulation
$\Omega$	Rotational velocity (rpm)
$\psi$	Azimuth angle (deg)
$\rho$	Air density (kg/m <sup>3</sup> )
$B$	Number of rotor blades
$C_{D_o}$	Parasitic drag coefficient
$C_M = \frac{M}{\rho n^2 D^5}$	Pitching moment coefficient
$C_{P_x} = \frac{P_x}{\rho n^2 D^4}$	Normal force coefficient
$C_P = \frac{P}{\rho n^3 D^5}$	Power coefficient
$C_T = \frac{T}{\rho n^2 D^4}$	Thrust coefficient
$D$	Rotor diameter (m)
$D_L$	Characteristic length (m)
$D_{par}$	Parasitic drag (N)
$J = \frac{V_\infty}{nD}$	Advance ratio
$\dot{m}$	Mass flow rate (kg/s)

$n$	Rotational velocity (rps)
$P_x$	Normal force (N)
$P_{ind,int}$	Interference induced power (W)
$P_{ind}$	Induced power (W)
$P_{par}$	Parasite power (W)
$P_{pro}$	Profile power (W)
$P_{rot}$	Rotor power (W)
$P_{tot}$	Total power required (W)
$Re$	Reynolds number
$T$	Required thrust (N)
$v_i$	Induced velocity (m/s)
$V_\infty$	Freestream velocity 9m/s)
$W$	Vehicle weight (N)
$w$	Downwash produced by rotor (m/s)
$cc$	subscript denotes basedrag values
$c$	subscript denotes corrected values
BEMT	Blade Element Momentum Theory
CFD	Computational Fluid Dynamics
RAALF	Ryerson Applied Aerodynamics Laboratory of Flight
SUAS	Small Unmanned Aerial System
UAV	Unmanned Aerial Vehicle
VTOL	Vertical Take-Off and Landing

# Chapter 1

## Introduction

### 1.1 Multirotor Small Unmanned Aerial Systems

Multirotor small unmanned aerial systems (sUAS) have become increasingly relevant in recent years. Multirotor sUAS, like the Aeryon SkyRanger quadrotor in Fig.1.1a, are versatile systems used for reconnaissance missions, for example inspection of power lines, oil rigs and crop surveillance. These vehicles fall into the vertical take-off and landing (VTOL) category of unmanned aerial systems. Unlike fixed wing UAVs, similar to the one pictured in Fig. 1.1b, multirotor sUAS can operate in both forward flight and hover.



(a) Aeryon SkyRanger quadrotor sUAS[1].



(b) Small fixed wing UAV[2].

Figure 1.1: Two types of small unmanned aerial systems.

Their ability to fly steadily at low speeds and altitudes is useful for surveillance, while the ability to take off and land vertically proves useful for operation in areas with rugged or uneven terrain. Small

unmanned aerial systems are desirable as they are also quieter and cheaper to operate than larger rotorcraft such as helicopters.

The quadrotor configuration offers a popular and relatively simple approach to stability and control. Quadrotor vehicles operate by having four, in-plane rotors in a cross configuration. Figure 1.2 illustrates the operation of quadrotor vehicles. Rotors opposite to each other rotate the same direction, with the pairs of rotors operating in opposite directions. The rotational velocity of the rotors in pairs or as a whole affects the movement of the vehicle. For example, to increase the altitude of the vehicle, the rotational velocity of all four rotors is increased together, increasing the thrust. The relative ease of controlling these vehicles at hover and slow flight have made them very popular for many applications. For example, the ability to operate an sUAS easily appeals to everyday consumers who would use the vehicles recreationally, while their slow flight and ability to hover has made them popular for military purposes, and for use in the film industry [3]. The controls, however, become significantly more complicated at higher speeds and when performing maneuvers. The challenges at higher speeds are primarily due to the highly nonlinear aerodynamics of these vehicles.

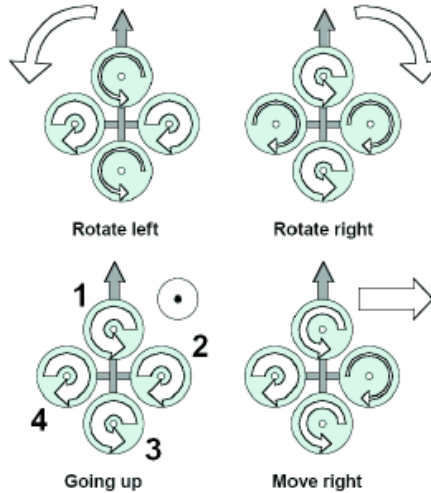


Figure 1.2: Depiction of standard quadrotor configuration[4].

The growing interest in the further development of multirotor sUAS has been spurred by the advancement in the technology used to operate these vehicles and the miniaturization of sensors that use sUAS as platforms. Multirotor sUAS are operated by controllers that gather information from various sensors on board the vehicle, such as accelerometers and directional sensors. The control systems have

seen major development over the past years through the miniturization of sensors and actuators, which aid in the reduction of power requirements [5]. Improvements to the motors that power sUAS have provided a decrease in cost, weight and packaging of the vehicles, allowing them to be further decreased in size [5]. As these advancements continue to decrease the cost and size of the vehicles, sUAS become more popular for different markets and their possible areas of application increase.

## 1.2 Multicopter sUAS Aerodynamics

The need for stable and controlled flight has lead to a large research focus on control systems of multicopter systems and less on aerodynamics [6]. Although control systems are a key component to controlled flight, it is essential to understand the aerodynamics of the vehicle as well. Bouabdallah has shown that rotor aerodynamics greatly affect motor control, as the forces and moments that are used in stability and control models can be directly attributed to the rotor[4]. In order to further expand the flight envelope of these systems, for example to fly at increasingly higher speeds, better aerodynamic models need to be included in controller designs.

Several factors complicate the analysis of the aerodynamics of multicopter sUAS. To begin, the size of the rotors used to produce thrust is much smaller than the conventional helicopter rotor with diameters usually ranging from 6-22 inches [7]. Small diameter rotors operate at lower Reynolds numbers typically from 30,000-300,000 based on the chord length at the  $\frac{3}{4}$  blade station [8], making scaling data from large rotors unreliable due to missing data at the low Reynolds number range. Airfoil data is often unreliable and difficult to predict at low Reynolds numbers, as flow behaves highly nonlinear in this region[9]. Airfoil performance decreases as the Reynolds number decreases. As the Reynolds number decreases, there is a drop in the maximum section lift coefficient,  $c_{l_{max}}$ , and an increase in the minimum drag coefficient,  $c_{d_{min}}$  [10]. A decrease in the  $c_{l_{max}}$  results in the blade section to stall earlier.

Another challenge arises from the wakes shed during the rotation of the blade. Young compared wake analysis methods and showed that the improper representation of the wake geometry causes the failure of most prediction models[11]. However, modelling these wakes in hover and forward flight proves to be a difficult task due to the complexity of the wakes, as there are many effects that occur simultaneously. For example, a multicopter vehicle requires rotors to operate in close proximity to one another. In Ref. [12], the effects of multicopter vehicles operating in close proximity to one another are explored. Micro

quadrotor vehicles were flown in close formation in multiple scenarios, having them travel in either the same direction or in opposite directions. The scenarios showed neighbouring rotors can have positive and negative affects on each other, depending on whether they are in the downwash or upwash of another vehicle.

Another issue arises in forward flight, as the wakes shed from the rotor plane are highly skewed. Both the rotors and their shed wakes induce a velocity in the flow field and thus, may impact the performance of other rotors. In Ref.[13] a practical method for computing the velocities induced by a rotor is presented. Results show that the velocity induced by the rotor at a specific point is affected by the skew angle of the rotor wake. The skew angle of a wake, discussed in Sec. 3.4, is a function of the angle of attack, the thrust, forward flight velocity and the advance ratio,  $J$ , which is the ratio of the forward velocity,  $V_\infty$ , to the rotor tip speed,  $nD$ :

$$J = \frac{V_\infty}{nD} \quad (1.1)$$

where  $D$  is the diameter of the blade in meters and  $n$  is the rotational velocity.

Uneven blade loading also affects the shape of the wake, increasing the difficulty of modelling. Blade loading increases near the tip of the rotor blade, increasing the vortex strength at the tip. As the wake propagates downward, the wake contracts and the tip vortex moves inward and close to the rotor plane. This results in an upwash field that the next blade passes over [14]. Due to the uneven blade loading, the outside of the wake propagates downward faster than the inside, further complicating the geometry of the wake, as it becomes inclined to the rotor plane. This effect is depicted in Fig. 1.3. The assumption that the rotors used to propel these sUAS experience light loading allow for the assumption of a uniform helix shaped wake[15].



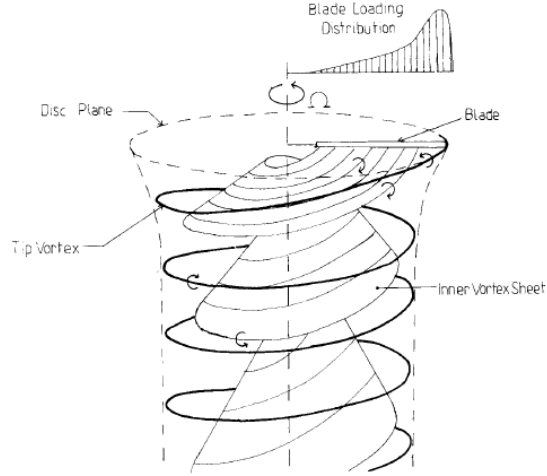


Figure 1.3: Wake geometry of rotor blade in hover [14].

### 1.3 Aerodynamic Performance Analysis Methods

A major design challenge of sUAS is to minimize the power required for flight in order to maximize endurance for missions. Assessing the power requirements of a multirotor vehicle is not straight forward, as there are many variables that affect the overall power required for flight. Examples include the weight of the vehicle, flight velocity, altitude and the type of rotor used. Tradeoffs between variables must be made when designing these vehicles. For example, the addition of battery packs will increase the total power available. The additional weight of the battery, however, also increases the power required. Payloads such as cameras also add weight and drag to the vehicle, also affecting the overall power requirements of the vehicle.

Currently there are different methods used to analyze the aerodynamic performance of multirotor sUAS. Computational fluid dynamics (CFD) methods, which solve the Navier-Stokes equations for flow around the rotor blades, can be used to predict vehicle and rotor performance. In Ref.[16], a CFD analysis was conducted on a helicopter scale model with focus on the main-rotor blades. The purpose of this study was to investigate the viability of CFD simulations with sUAS. The simulation showed good agreement with experimental data for hovering flight, with some difficulties predicting performance in forward flight as moments in pitch and roll began to emerge. While CFD simulations may produce good predictions, the method tends to be computationally expensive and time consuming, and is not feasible

for the early stages of conceptual design, where it is necessary to solve multiple design configurations quickly. Furthermore, CFD only provides limited insight since it only generates integrated loads. The integrated loads developed using CFD make it difficult to identify the source of performance penalties, for example whether they are related to wake effects or due to configuration choices.

As a rotor performance prediction method, the blade element momentum theory (BEMT) combines the blade element and momentum theory models of rotors and offers an approach that is computationally efficient, while still able to produce good predictions [17, 18, 19]. As the blade element momentum theory is only a rotor analysis tool, it must be used in conjunction with another model to produce vehicle performance predictions.

Wind tunnel testing is another method used for analyzing the performance of multirotor sUAS. A project performed at the wind tunnel at NASA Ames Research Center tested a number of multirotor sUAS, measuring the full body vehicle forces and moments at various angles of attack and forward flight velocities[7]. The project was performed to address the lack of published data for the performance of sUAS, and showed that wind tunnel testing can be used to produce data for validating analytical and numerical models. While wind tunnel testing is a good method for measuring performance, it is not the most versatile performance prediction method as tunnel access is not readily available to most designers. In addition, this method would be expensive and time consuming if used in the preliminary design phase, as an existing prototype would need to be made. While wind tunnel testing can give an overall measure of the vehicle performance, it does not provide a detailed breakdown of the sources of the performance cost.

## 1.4 Contribution

In order to aid the design process of small multirotor vehicles and their rotors, a multirotor vehicle performance prediction method was developed. For known flight conditions, for example, hover or vehicle orientation, and vehicle configuration parameters, the prediction model can be used to predict certain performance characteristics, such as the power and thrust required over the flight range. This thesis discusses in detail the vehicle performance prediction model and the components of the model. The prediction model is validated using flight-test data provided by Aeryon Labs Inc, a designer and manufacturer a sUAS based in Waterloo, Ontario. Finally, several design studies are presented in which

the performance impact of various flight conditions and aircraft configurations are explored.

## Chapter 2

# Multicopter Vehicle Performance

The term "multicopter vehicle performance" used in this thesis refers to the power required,  $P_{req}$ , of a multicopter vehicle for a given flight condition. The power required for flight is an important metric, as the vehicle battery can supply only a limited amount of power. The power required limits the endurance of the vehicle, how long the vehicle can stay airborne. Identifying the factors that affect the power required and including aerodynamic effects in the control system models that operate the vehicle will enable designers to improve the endurance and range of the vehicle, making them more appealing to consumers [20].

Understanding the contribution of the factors that affect the power required over a flight range is essential to making an accurate prediction of the performance, though it is a difficult task as there are different effects occurring simultaneously. Low speed Reynolds number effects increase the difficulty of rotor performance prediction due to highly nonlinear behaviour of airflow in this region. In addition, rotors operating in close proximity to each other complicates the problem of performance prediction, as rotors and their wakes influence the performance of neighboring rotors.

The forces that act on a multicopter vehicle during steady, horizontal flight are shown in Fig. 2.1. The angle of attack,  $\alpha$ , is the angle at which the vehicle is angled to the freestream velocity,  $V_\infty$ . In the work presented here, the angle of attack is considered positive when the freestream velocity impinges the reference line, here defined by the rotor plane, from above. The freestream velocity,  $V_\infty$ , is parallel to the flight path. Subsequently, the vehicle drag is in opposite to the vehicle motion. The weight,  $W$ , acts negatively along the z-axis. The thrust,  $T$ , is the force produced by the rotors to overcome the

weight and drag of the vehicle and acts perpendicular to the rotor plane. Finally the in-plane normal force produced by the rotor,  $P_x$ , acts perpendicular to the axis of rotation.

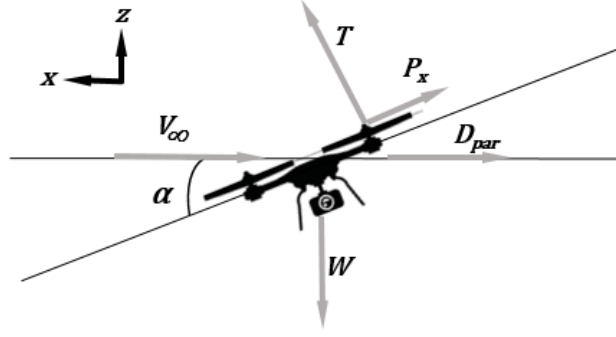


Figure 2.1: Free-body diagram of sUAS in forward flight.

In this chapter, the equations of motion that govern the flight of multirotor sUAS are introduced and important performance metrics are discussed, in particular, exploring their relation to each other and the overall vehicle performance. The concept of the power required is also discussed in greater detail, decomposing the power required into its components and explain the factors that affect them.

## 2.1 Equations of Motion

Using the free-body diagram presented in Fig. 2.1, the equations of motion in the longitudinal plane can be derived by totaling the forces acting on the vehicle:

$$0 = D_{par} + P_x \cos \alpha - T \sin \alpha \quad (2.1)$$

$$0 = T \cos \alpha + P_x \sin \alpha - W \quad (2.2)$$

Rearranging Eq. 2.2 for  $\cos \alpha$ :

$$\cos \alpha = \frac{W - P_x \sin \alpha}{T} \quad (2.3)$$

Substituting in Eq. 2.1 and multiplying both sides by thrust,  $T$ , yields:

$$TD_{par} + P_x(W - P_x \sin \alpha) - T^2 \sin \alpha = 0 \quad (2.4)$$

Rearranging for  $\sin \alpha$ :

$$\sin \alpha = \left( \frac{P_x W + D_{par} T}{T^2 + P_x^2} \right) \quad (2.5)$$

Rearranging Eq. 2.2 and using the angle of attack, thrust is:

$$T = \frac{W - P_x \sin \alpha}{\cos \alpha} \quad (2.6)$$

The normal force,  $P_x$ , is a function of the angle of attack, though the value of the normal force is required to find the angle of attack in Eq. 2.5. This dependency results in the need for an iterative solution to evaluate the angle of attack and the thrust. This process is discussed in Section 3.1.

## 2.2 Multicopter Vehicle Thrust

Using the equations of motion, the variables that effect the overall vehicle thrust can be explored. Referring to Eq. 2.6, the dependency on the weight of the vehicle is shown. As the weight is increased, the required thrust is increased as well. Equation 2.1 can also be rearranged to describe the thrust, showing the dependency on the parasitic drag:

$$T = \frac{D_{par} + P_x \cos \alpha}{\sin \alpha} \quad (2.7)$$

For an increase in the parasitic drag, there is an increase in the required thrust. By looking at what constitutes the parasitic drag, other variables that affect the required thrust become more apparent. The parasitic drag of the vehicle body can be expanded using the non-dimensional variable  $C_{D_o}$ , as:

$$D_{par} = \frac{1}{2} C_{D_o} \rho V_\infty^2 S_w \quad (2.8)$$

where  $C_{D_o}$  is the parasitic drag coefficient and  $S_w$  is a reference area. Parasitic drag depends on form drag and the size of object matters with regards to that drag value. Furthermore, altitude, at which the

vehicle operates, also affects the drag of the vehicle. With an increase in the altitude, the air density,  $\rho$ , decreases. This will result in a decrease in the parasitic drag if the other variables are kept constant. Finally, an increase in the forward velocity of the sUAS will once again result in an increase in drag. It is important to understand the contribution of these variables as they not only affect the production of thrust, but the production of drag and thus, the power required. This is further explored in the following section.

## 2.3 Multicopter Vehicle Power

The power required of a multicopter vehicle can be expressed using a power decomposition:

$$P_{tot} = P_{rot} + P_{par} + P_{ind,int} \quad (2.9)$$

where  $P_{rot}$  is the power that is required to operate the rotor,  $P_{par}$  is the parasitic power, which is related to the drag of the fuselage and other extremities, and  $P_{ind,int}$ , which is the induced interference power. Each of the components is discussed in detail in this section. To better predict the performance of a multicopter sUAS, it is important to understand how the components are affected by factors such as physical attributes of the vehicle and flight conditions. A power decomposition, where the total power required and its components are plotted over a range of velocities, can be performed to understand the performance of the vehicle and identify areas that can be improved.

A power decomposition is shown in Fig. 2.2 for the example of the lead rotor of a sUAS with a quadcopter configuration during steady and horizontal flight for standard sea level flight. The figure shows the total power required with separate lines for the contributing components. Parasitic power is zero during hover, but increases approximately with the cube of the velocity (see Eq. 2.10.) The parasitic power, of which only one-fourth is shown, is directly attributed to the physical geometry of the vehicle as it is related to the drag of the body, landing gear and support arms. Therefore, variables such as the number of rotors used, and therefore, arm components required for mounting the rotors, affect the parasitic power.

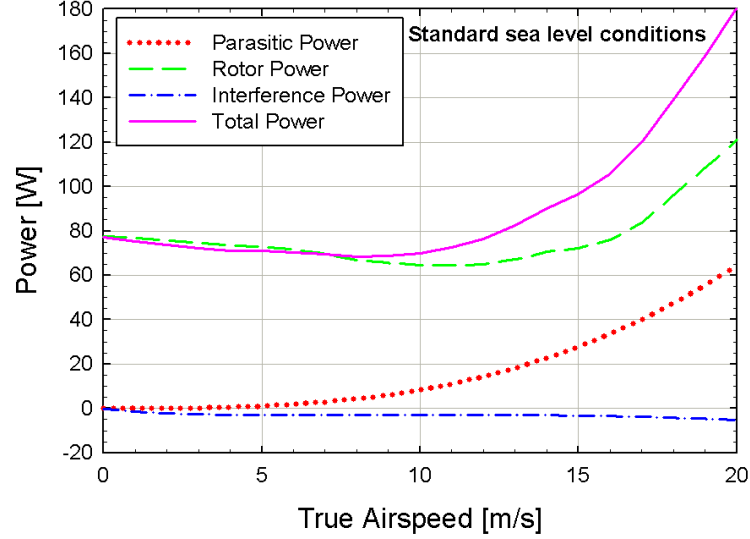


Figure 2.2: Total power required and power decomposition for the lead rotor of a quadcopter in forward flight.

The interference power is due to several rotors being in close proximity, which actually reduces the total power required, because of the induced upwash field. This is especially evident from 0 m/s to 5 m/s where the pure rotor power that is required to produce a desired thrust, is larger than the total power that is actually required. The negative interference power decreases the total power required, showing an assistance from the other rotors. The interference power is discussed more in detail in the following sections. At high speeds, the penalty of parasitic power results in a relatively steep increase in power required. This can be used to find the optimal flight speed that maximizes flight range and endurance. With an evident dependency on the components of the power required across the flight regimes, the performance of multirotor vehicles can be improved with a better understanding of the contributions of each of the components. The following sections will introduce the components in detail.

### 2.3.1 Parasitic Power

Parasitic power is directly attributed to the aerodynamic form drag produced by the vehicle components, which are the body, rotor support arms, landing gear and any components not directly related to the rotors. Thus, the parasitic drag is the total drag of the sUAS, with the exception of any drag related to the rotors. The parasitic power can be expressed as:



$$P_{par} = D_{par}V_{\infty} \quad (2.10)$$

Referring to Eq. 2.8, the parasitic drag is proportional to the square of the freestream velocity. Therefore, the parasitic power, in a first approximation, is proportional to the velocity cubed. The parasitic drag, composed of the skin friction drag and the pressure drag, is affected by the shape of the components and the surface quality. For example, more streamlined objects produce less pressure drag as flow separation is less likely to occur. Rough surfaces can cause flow to transition from laminar flow to turbulent flow, which may increase drag.

### 2.3.2 Rotor Power

The power that is required for a rotor to produce thrust, can be separated in profile power and the induced power:

$$P_{rot} = P_{pro} + P_{ind} \quad (2.11)$$

The profile power,  $P_{pro}$  is the power associated with the profile drag of the blade. The induced power,  $P_{ind}$ , is related to the velocities that the rotor induces. It is related to the self induced flow field and is in contrast with the induced interference power, which is a result of the rotor interacting with rotors and their wakes in its vicinity. Both the induced power and profile power are directly attributed to the rotor and its geometry and are therefore affected by the type of rotor used for propulsion. In Ref.[21], Gustafson compares the effect of the profile drag of helicopter rotors on the total power required, evaluating the performance of different rotor blades. The profile drag is affected by the sectional airfoil used for the rotor blade and its geometric characteristics. For example, Gustafson showed an increase in the thrust produced and a decrease in the power required for hovering flight when using a low-drag NACA 3-H-13.5 airfoil section, instead of the thicker profiled NACA 23015 [21, 22]. Gustafson's studies show the importance of the rotor geometry as it can impact the production of thrust and the overall vehicle performance.

The induced power is directly related to the production of thrust and the induced velocity,  $v_i$ :

$$P_{ind} = Tv_i \quad (2.12)$$

In Fig. 2.3 a section of a rotor airfoil is shown during hover. The presence of the induced velocity, named  $w$  in this figure, reduces the sectional angle of attack,  $\alpha$ , by  $\alpha_i$ , tilting the lift vector backwards. The resulting component of the lift in the horizontal direction is the induced drag.

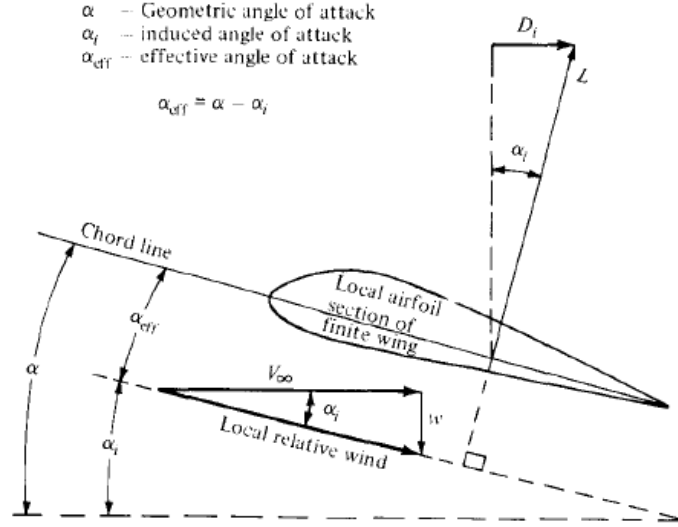


Figure 2.3: Effect of induced velocity on rotor airfoil section[23].

### 2.3.3 Interference Power

The interference power is the result of rotors and their wakes operating in close proximity and influencing each other. The ring in Fig. 2.4 represents a rotor in hover. The prediction of the induced flow field using the wake interference module is discussed in detail in Section 3.4. Note, the vectors are scaled to make the flow directions more easily noticeable. The direction and strength of the induced velocity vectors change with proximity to the rotor plane. These induced velocities have an influence on neighbouring rotors. Rotors caught in a downwash require more power in order to produce a certain amount of thrust. In contrast, rotors located in a region of upwash require less power for the same amount of thrust. This component of the power required caused by the induced velocity of rotors in proximity is called the interference induced power,  $P_{\text{ind,int}}$ , and can be expressed as:

$$P_{\text{ind,int}} = T v_{i,\text{int}} \quad (2.13)$$

where  $T$  is the thrust and  $v_{i,\text{int}}$  is the induced interference velocity, the velocity induced on a particular

rotor by its neighbouring rotors.

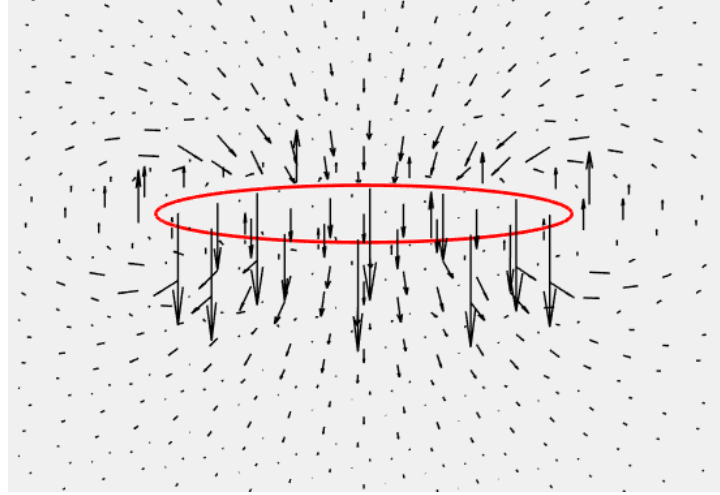


Figure 2.4: Induced velocity vectors produced by rotor in hover.

This effect can be seen when comparing rotors in a coaxial configuration, where the rotors are stacked one atop the other, to the single rotor configuration. Figure 2.5 shows the rotor power that the two configurations require during forward flight versus the advance ratio as it is defined in Eq. 1.1 and referred to as  $\mu$  in this figure. For the same advance ratio, the coaxial configuration requires a greater rotor power. In addition to the added drag due to additional components needed for the coaxial configuration, the interference between the two rotors and their wakes increases the power required by a factor of nearly three during hover.

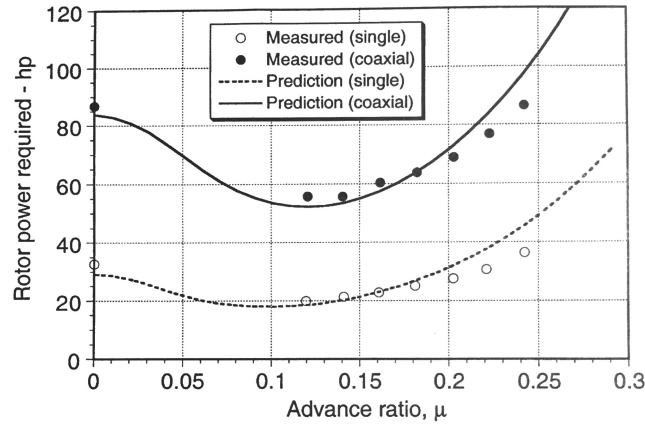


Figure 2.5: Power required for coaxial and single rotor configuration[17].

## 2.4 Rotor Thrust and Power

The production of thrust, the upward force produced by the rotor that is responsible for the propulsion of the vehicle, can be explained using classical momentum theory analysis. Shown in Fig. 2.6 is a helicopter in axial flight [17]. In classical momentum theory, it is assumed that over the rotor disc, the velocity is constant and pressure is uniform. The flow is assumed to be incompressible.

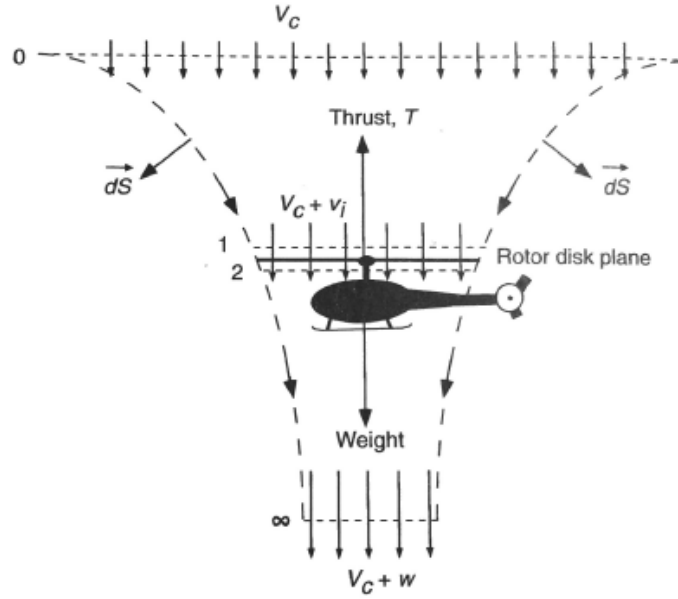


Figure 2.6: Power required for coaxial and single rotor configuration [17].

From momentum theory:

$$T = \dot{m}w \quad (2.14)$$

where  $w$  is the downwash, the velocity of the far wake and  $\dot{m}$  is the mass flow rate through the rotor blade. For a constant mass flow rate:

$$\dot{m} = \rho A v_i \quad (2.15)$$

The downwash is related to the induced velocity by:

$$w = 2v_i \quad (2.16)$$

Substituting Eq. 2.15 and Eq. 2.16 in Eq. 2.14, the thrust in hover is:

$$T = 2\rho A v_i^2 \quad (2.17)$$

The rotor power is related to the thrust by the induced velocity:

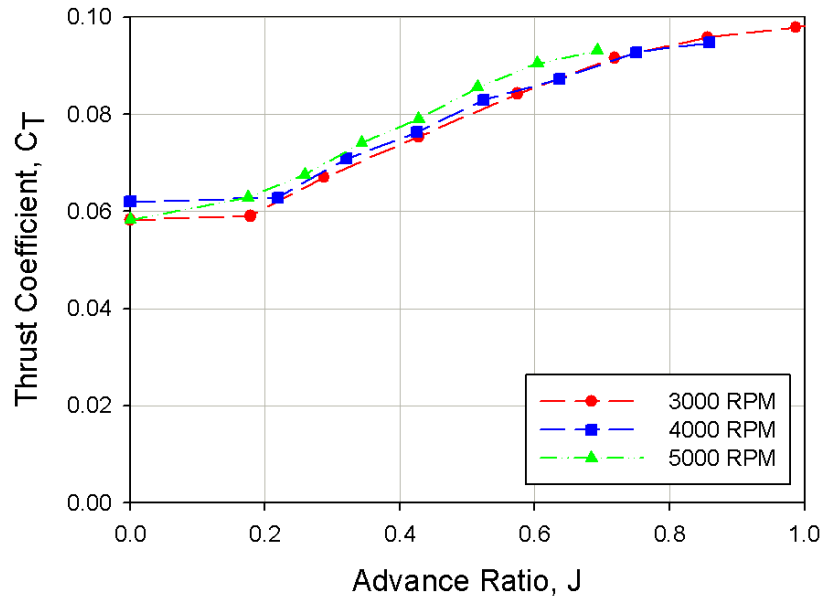
$$P = T v_i = 2\rho A v_i^3 \quad (2.18)$$

The thrust and rotor power are in coefficient form:

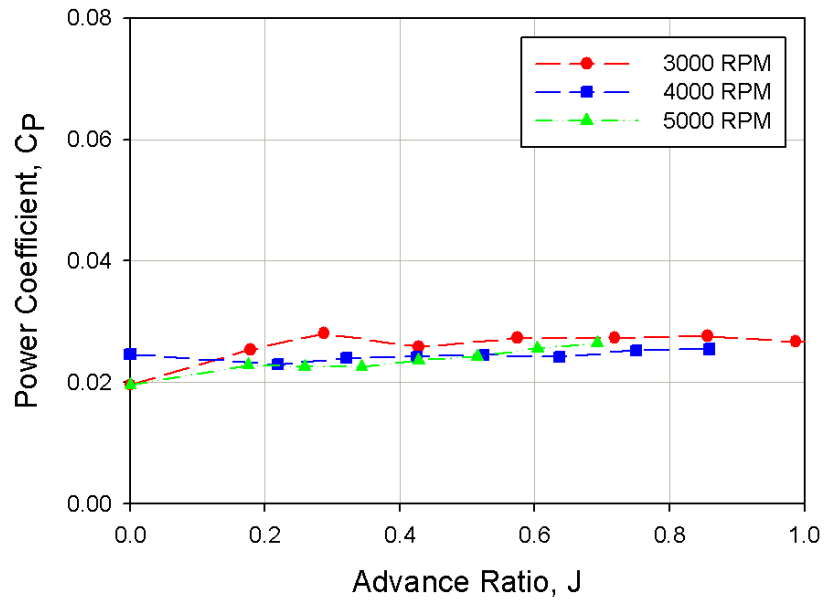
$$C_T = \frac{T}{\rho n^2 D^4} \quad (2.19)$$

$$C_P = \frac{P}{\rho n^3 D^5} \quad (2.20)$$

where  $n$  is the rotational velocity of the rotor blade and  $D$  is the diameter of the blade. The performance coefficients are used to more easily compare rotors of different sizes or performance of rotors operating at different altitudes or rotational velocities. To more easily assess the performance of rotors, the performance coefficients are commonly plotted versus the advance ratio for a range of angle of attacks and rotational velocities. An example is shown in Fig. 2.7, which shows experimental data taken in the Ryerson University Large Subsonic Wind Tunnel. Two sets of data at rotational velocities of 3000 rpm, 4000 rpm and 5000 rpm are presented, for an angle of attack of 0 degrees. Referring to Fig. 2.7a, the thrust coefficient increases as the advance ratio increases. Comparing the data sets between rotational velocities, it is shown that for the same advance ratio, the thrust coefficient increases with the rotational velocity. In both Figs. 2.7a and 2.7b, we see that between rotational velocities, there is only a slight increase and decrease in the  $C_T$  and  $C_P$  respectively for the same angle of attack.



(a) Effect of advance ratio on thrust coefficient.



(b) Effect of advance ratio on power coefficient

Figure 2.7: Experimental performance coefficients for ASP130 rotor taken in the Ryerson University Large Subsonic Wind Tunnel.

## 2.5 Advancing/Retreating Blade

During forward flight, the rotor blade of an sUAS experiences a nonuniform inflow field and, thus, uneven blade loading. This uneven velocity field causes effects known as advancing and retreating blade effects. Figure 2.8, shows a rotor blade during forward flight. The azimuth angle,  $\psi$ , is the angular position of the rotor blade which is measured positively in the direction of rotation. The blade is advancing when traveling in the direction of flight and retreating when traveling opposite the direction of flight.

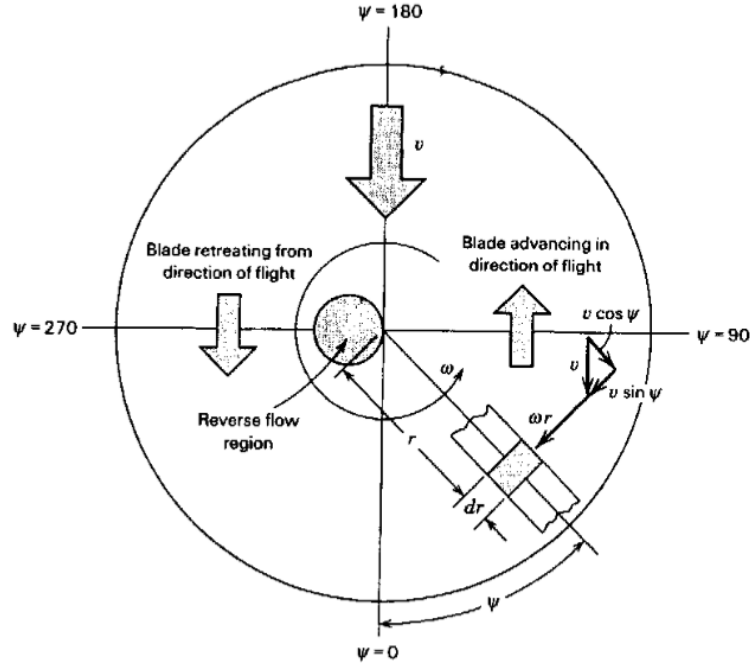
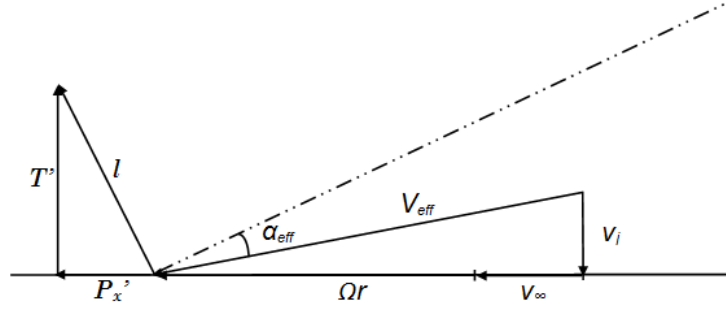
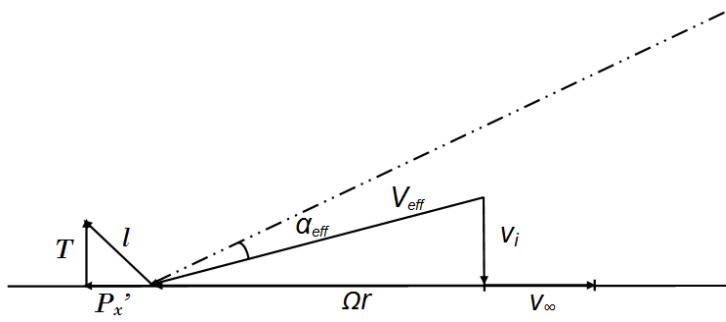


Figure 2.8: Velocities acting on element of rotating blade at azimuth angle  $\psi$ [19].

Diagrams of the velocity vectors of the advancing and retreating blade are shown in Fig. 2.9. On the advancing side, blade tip speed,  $\omega R$ , and the forward flight speed,  $V_\infty$ , are summed, while on the retreating side, the blade tip speed is opposite to the forward flight speed, which results in a decrease in the horizontal component of the effective velocity,  $V_{eff}$ . On the retreating side, the effective velocity vector is angled closer to the zero lift line (the dotted line in Fig. 2.9) than in the case of the advancing blade. Subsequently, the effective angle of attack,  $\alpha_{eff}$ , is reduced on the retreating blade. This in turn reduces the section lift component,  $l$ , on the retreating side, causing an asymmetric production of lift. The resulting horizontal component results in a net force in the plane of rotation,  $P_x$ .



(a) Advancing blade.



(b) Retreating blade.

Figure 2.9: Flow diagrams for advancing and retreating blade conditions.

Due to the larger horizontal component when the blade is advancing than retreating, for the same induced velocity the angle of attack,  $\alpha$ , is larger for an advancing blade than a retreating blade. The asymmetric loading creates a roll moment at  $90^\circ$  and  $270^\circ$ . While some helicopter blades are articulated, allowing them to flap, multirotor sUAS blades do not flap, though they do experience small plastic deformations. Performance measurements taken in the Ryerson University Large Subsonic Wind Tunnel measure the roll moment as a pitching moment since the rotors are mounted as propellers in the tunnel. The pitching (roll) moment,  $M$ , can be expressed in coefficient form as:

$$C_M = \frac{M}{\rho n^2 D^5} \quad (2.21)$$

Although the proposed model is capable of predicting the pitching moment, a moment trim solution is not the focus of the herein discussed research. Therefore, the pitching moment is disregarded.

The horizontal normal force present in the axis of rotation can be expressed in coefficient form [18]:



$$C_{P_x} = \frac{P_x}{\rho n^2 D^4} \quad (2.22)$$

As the normal force is dependent on the angle of attack and advance ratio, the normal force is not present during hover. Randall provides a study on the aerodynamics of micro air vehicles where the forces acting on the propulsion system were investigated [24]. Figure 2.10 shows the affect on the angle of attack on the normal force coefficient (referred to as  $C_N$  in the provided example.) As the angle of attack increases, the normal force steadily increases as well.

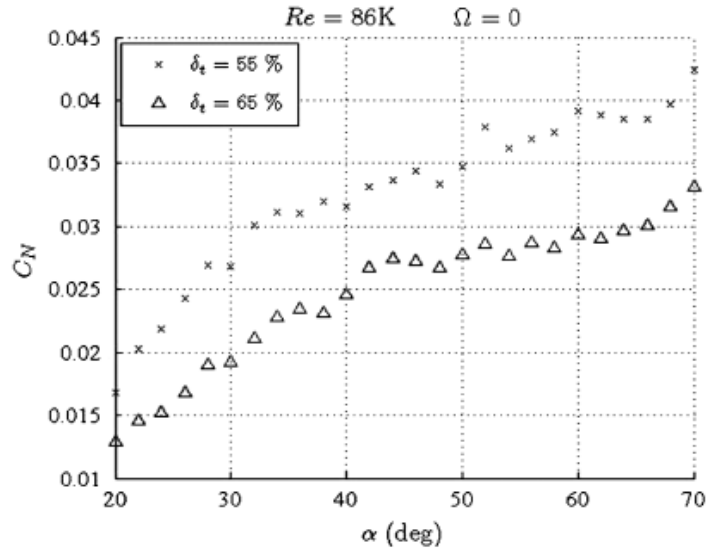


Figure 2.10: Normal force coefficient versus angle of attack [24].

## Chapter 3

# Multirotor Vehicle Performance Model

As part of the herein discussed research, a multirotor vehicle prediction method was developed using MATLAB. For known flight conditions, angle of attack and required thrust, the multirotor vehicle performance prediction model predicts performance metrics over the flight envelope. For example, for a known angle of attack and forward velocity of the vehicle, the power required and parasitic drag can be predicted. The vehicle performance model is a useful tool to aid in conceptual design as the geometrical aspects of the vehicle and physical conditions can be easily changed to study the effect on performance. In this chapter, the vehicle performance prediction model is discussed in detail.

### 3.1 Methodology

The vehicle performance model is composed of three subfunctions that evaluate the components of the total power required: the parasitic drag module, rotor performance module and the wake interference module. An overview of the workflow of the model is shown in Fig. 3.1. Each of the subfunctions is discussed in detail in the following sections. The model inputs are the freestream flow conditions and the rotor and vehicle geometry, while the air density is calculated as a function of the vehicle's altitude. The outputs include the power required per rotor, the vehicle angle of attack and the rotational velocity and thrust required by each rotor. Overall, the program is organized in modules that are easily modified or exchanged in order to be able to respond to configuration changes.

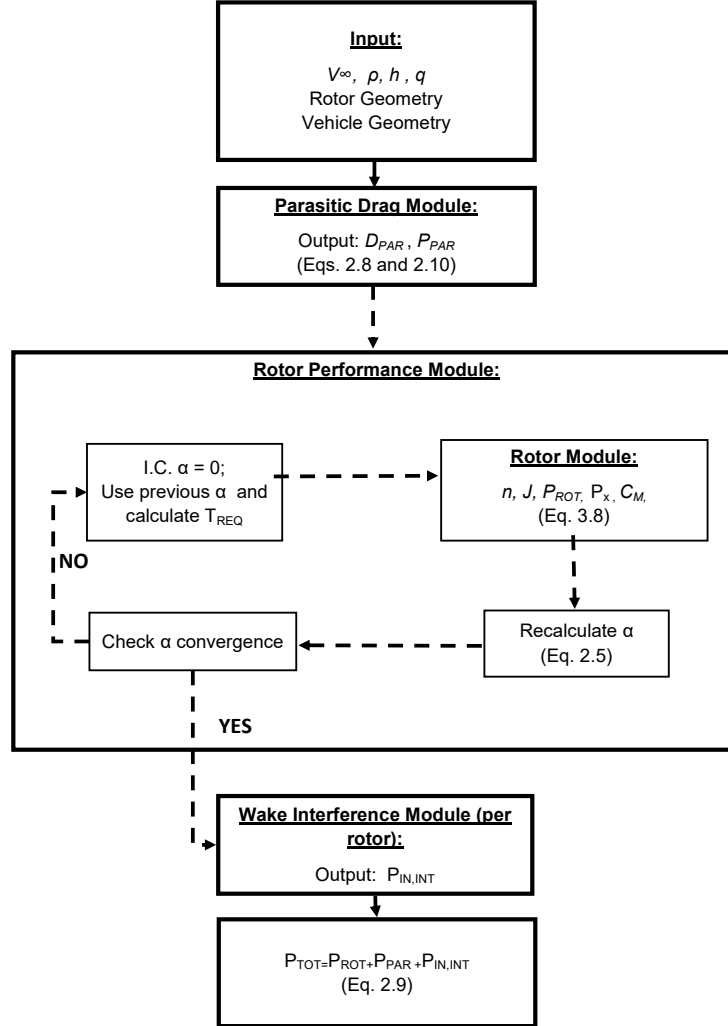


Figure 3.1: Performance model flow chart

Over a flight speed range the angle of attack, thrust, rotational velocity and power required per rotor are computed. For each velocity point, the vehicle performance model begins with the prediction of the parasitic drag and power using the flow conditions and the geometry of the vehicle components. In the rotor performance module, an iterative method which converges on the angle of attack is performed to evaluate the thrust, normal force, roll moment, rotational velocity and rotor power. In each iteration,

the thrust and angle of attack are used in an interpolation scheme of tabulated performance data to evaluate the performance variables. This process is discussed in Section 3.3.1. After a thrust has been determined during the previous iteration, the angle of attack is recomputed using Eq. 2.5 and compared with the value of the previous iteration step. If convergence is not achieved, a new thrust value is calculated using Eq. 2.7.

The next velocity point starting angle of attack and thrust are based on the values of the previous case. The special case is hover, where the angle of attack is zero and the thrust is equivalent to the weight of the vehicle. With the rotor power evaluated, the next step in the workflow model is the wake interference module. In this module, the induced interference power, the power produced by the interaction between a rotor and the wake it sheds as well as rotors in proximity, is evaluated for each rotor.

Finally, the total power per rotor is then calculated using Eq. 2.9. Note, the total power is a summation of the rotor power, parasitic power and the induced interference power, which may differ between rotors. Thus, the total power required is calculated per rotor. The effect of the interference power and the vehicle orientation on the vehicle performance is investigated in Sec. 5.3. The convention used for the vehicle orientation and rotor numbering system in the model is shown in Fig. 3.2, where the dotted line in the figures represents the line of symmetry.

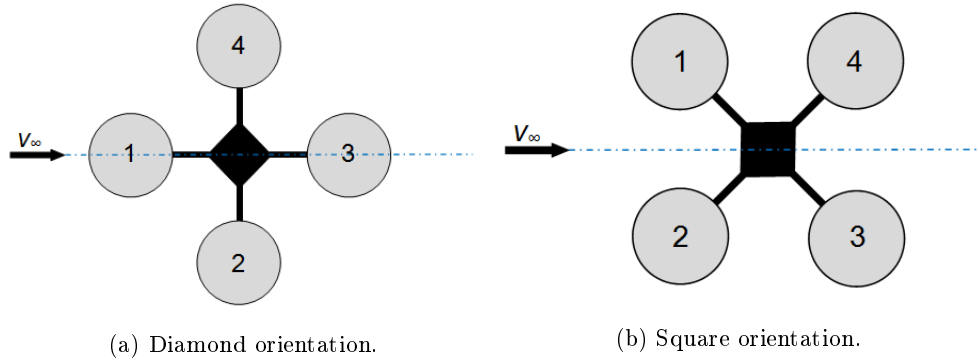


Figure 3.2: Vehicle orientation and rotor numbering convention.

## 3.2 Parasitic Drag Module

In this section, the parasitic drag prediction module is introduced in detail, explaining the methods and any assumptions. As mentioned, parasitic drag is due to the drag of the fuselage, rotor support arms, landing gear, and any other elements not directly related to the rotor. Results produced using the method are presented and compared to experimental wind tunnel data. Finally, an example is presented where the total drag of the Aeryon SkyRanger sUAS is predicted.

### Drag Prediction Method

The parasitic drag and subsequent parasitic power are determined using an estimated approach by totaling the drag each component of the vehicle contributes. The complex shape of the components are decomposed into common shapes. For example for the Aeryon Skyranger shown in Fig. 3.3, the parasitic drag is estimated by adding the drag of the center body (A), rotor arms (B), and landing gear (C). The body is assumed as a sphere whereas the other components are approximated using cylinders.



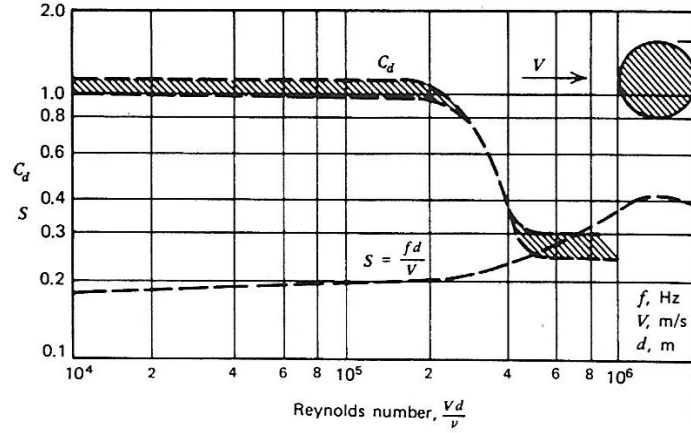
Figure 3.3: Aeryon Skyranger with labeled components: A) centre body B) rotor arms C) landing gear[1].

The drag coefficient versus the Reynolds number for smooth spheres and cylinders is shown in Fig. 3.4. The characteristic length,  $D_L$ , which is the diameter of the sphere or cylinder, is used to find the Reynolds number:

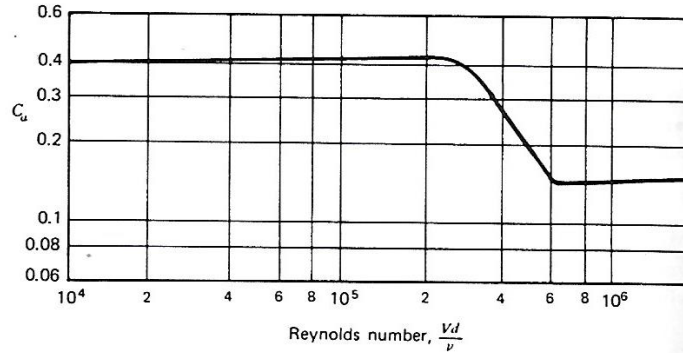
$$Re = \frac{\rho V_{\infty} D_L}{\mu} \quad (3.1)$$

At supercritical Reynolds numbers, transition from laminar to turbulent flow occurs prior to flow

separation, resulting in longer attached flow and a decrease in the pressure drag[19]. This leads to significant reductions in drag coefficients for cylinders (Fig. 3.4a) and spheres Fig. 3.4b) at Reynolds numbers greater about 300,000. Quantitatively, the drag coefficient of a sphere is smaller than that of a cylinder. This is due to three dimensional relieving effects where flow can travel in additional directions over a sphere as compared to a cylinder where it can only move over or under the cylinder[23]. This decreases the stress on the flow and thus the drag.



(a) Circular cylinder.



(b) Sphere.

Figure 3.4: Drag coefficient versus Reynolds number for sphere and cylinder[19].

To predict the parasitic drag, the flow conditions and characteristic length for each component are first input into the model. Based on the geometry, the Reynolds number is calculated for the component. Then using the Reynolds number and interpolating with experimental drag coefficient and Reynolds number data for cylinders and spheres, the parasitic drag coefficient is determined.

Although components may not be designed to be lifting surfaces, some may produce a lift force. In

turn, this production of lift results in an increase in the drag coefficient of the component [19]. This increase in the drag coefficient is the induced drag coefficient and is expressed as:

$$C_{D_i} = \frac{(\alpha C_{L_\alpha})^2}{\pi AR e} \quad (3.2)$$

It is important to note that as the angle of attack increases the induced drag does as well. In addition, the induced drag is inversely related to the aspect ratio. An elliptical lift distribution is assumed thus, the span efficiency is 1. The lift curve slope,  $C_{L_\alpha}$ , is evaluated as [19]:

$$C_{L_\alpha} = \frac{C_{l_\alpha} AR}{2 + \sqrt{AR^2 + 4}} \quad (3.3)$$

Based on thin airfoil theory, the section lift curve slope,  $C_{l_\alpha}$ , is assumed to be  $2\pi$ . The total drag coefficient is the sum of the induced drag coefficient and the parasitic drag coefficient:

$$C_D = C_{D_o} + C_{D_i} \quad (3.4)$$

where the zero-lift drag coefficient,  $C_{D_o}$ , is based on the drag of a sphere according to Fig. 3.4 and the induced drag coefficient,  $C_{D_i}$ , on Eq. 3.2.

Finally, the total drag and total power are found using Eqs. 2.8 and 2.10, respectively. As the lift force, and thus the induced drag, is a second order effect, the lift and induced drag are not included in the model. Furthermore, a trim solution is not included in the current model.

### Experimental Drag

Experimental performance data was collected for the body of the Aeryon Scout sUAS in the Ryerson University Large Subsonic Wind Tunnel as shown in Fig. 3.5. The body was mounted on a three degrees of freedom external balance used to measure lift, drag and the pitching moment of the body. Clearly visible in Fig. 3.5 are the two vertical pylons that connect to the external balance above the test section. Further visible in that figure are the rods that support the test object in the test section.



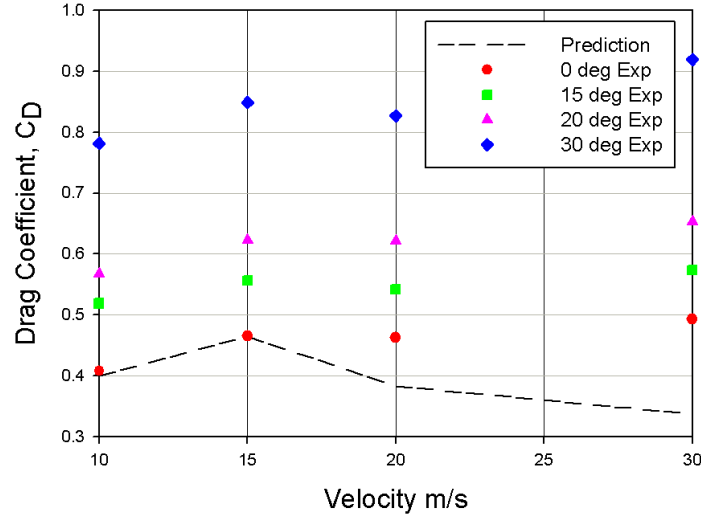
Figure 3.5: Aeryon Scout body mounted on balance.

Prior to testing, the balance was calibrated and the influence of the supports was determined. The pitching moment is considered positive in the clockwise direction while drag is positive in the direction of flow. Lift is measured positive downward. Tests were conducted at 10, 15, 20 and 30 m/s and for angles from 0 to -30 degrees in 5 degree increments. The measured data was post processed using a MATLAB script. Figure 3.6a shows the lift and drag coefficient of the body measured in the tunnel tests. As expected, Fig. 3.6a shows the drag coefficient increasing with both angle of attack and velocity. From Fig. 3.6b we see that the body acts as a lifting surface at some velocities at an angle of attack of 0 degrees, although only an insignificant amount of lift is produced. The body produces a downforce as the negative angle of attack and velocity increase.

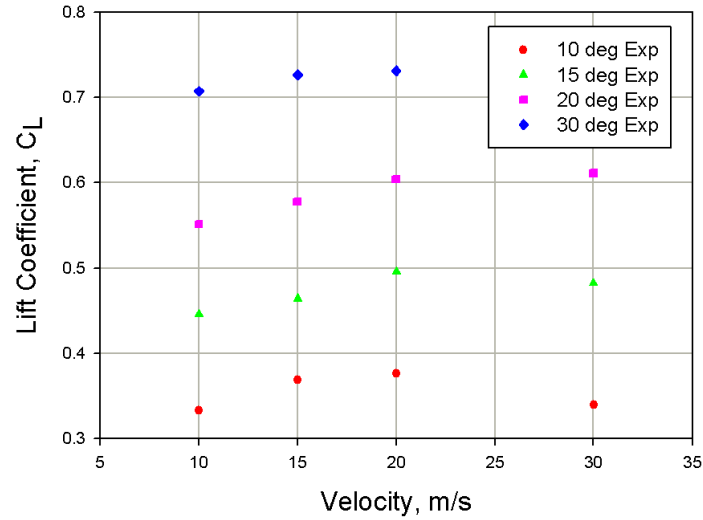
### Drag Model Validation

Figure 3.6a shows the comparison between the experimental and prediction result of the drag coefficient with respect to angle of attack. The drag The Reynolds numbers are based on the body diameter of 18cm. For lower velocities and angles of attack, the prediction model fits the experimental results quite well. As the velocity increases, the drag model strays from the experimental results. This is a result of the zero-lift drag,  $C_{D_o}$  of Eq.3.4, suddenly getting smaller, as the sphere that represents the fuselage changes from sub- to super-critical Reynolds numbers as indicated in Fig.3.4b. As the maximum velocity at which the Aeryon Scout operates is around 20m/s, this dropoff of the  $C_D$  is not an issue, since this is beyond the expected operational range.





(a) Drag coefficient versus angle of attack.



(b) Lift coefficient versus angle of attack.

Figure 3.6: Lift and drag coefficients measured in wind tunnel for Aeryon Scout body.

### Drag Prediction Example

An example of the parasitic drag estimation is presented using the Aeryon Skyranger body, landing gear and arms. The dimensions of the components of the sUAS are presented in Table 3.1. The flow

conditions, for sea level, are presented in Table 3.2.

Table 3.1: Dimensions of SkyRanger components.

Dimension	Value (m)
Landing gear length	0.295
Landing gear width	0.02
Rotor arm length	0.255
Rotor arm width	0.018
Centre body diameter	0.1

Table 3.2: Values for determining Reynolds number for components of Aeryon S kyranger.

Variable	Value
Velocity (V)	5 m/s
Density ( $\rho$ )	1.225 kg/m <sup>3</sup>
Dynamic viscosity ( $\mu$ )	1.846x10 <sup>-5</sup> kg/ms

First, the Reynolds number for each component is calculated using their respective characteristic length,  $D_L$ . For the landing gear and rotor arms, the characteristic length is the width of the component. The Reynolds number for the centre body,  $Re_B$  is:

$$Re_B = \frac{\rho V_{\infty} D_L}{\mu} = \frac{(1.225)(5)(0.18)}{1.846 \times 10^{-5}} = 59723.73 \quad (3.5)$$

Using the plots for drag coefficient versus Reynolds number for cylinders and spheres presented in Fig. 3.4, the parasitic drag coefficient for each component are determined. Finally, the parasitic drag can be calculated using Eq. 2.8. The wetted area,  $S_w$ , for the components assumed to be cylindrical is the length multiplied by the diameter of the component. As there are four rotor arms and four landing gear components, the total parasitic drag is:

$$D_o = 4D_{o_L} + 4D_{o_A} + D_{o_B} \quad (3.6)$$

Subscript  $B$  refers to the centre body, subscript  $L$  refers to the landing gear and subscript  $A$  refers

to the rotor arm. The parasitic power is then calculated using Eq. 2.10. Table 3.3 shows the Reynolds number, parasitic drag coefficient and parasitic drag for each component.

Table 3.3: Predicted Reynolds number, parasitic drag coefficient and parasitic drag for AeryonSkyranger components.

	Re	$C_{Do}$	$D_o$ (N)
Landing gear	16590	1.00	0.045
Rotor arm	33180	1.04	0.082
Centre body	59724	0.457	0.736

The parasitic drag is proportional to the square of the airspeed, while the parasitic power has a cubic relationship to the airspeed. Therefore, parasitic drag must be considered when designing sUAS as its influence increases with the number of components added to the vehicle.

### 3.3 Rotor Module

As indicated in Fig. 3.1, the power that is needed to produce a certain amount of thrust with a rotor is determined using the rotor performance module. Tabulated rotor performance data is used in a table lookup routine. The tabulated data can be based on experimental results, for example from wind-tunnel tests, or on theoretical predictions, for example as done by Carroll [25]. The table lookup method for evaluating the rotor power is introduced in this section, while the tables used to produce results in this thesis are presented in Appendix B. The prediction and experimental methods used to populate the rotor performance tables, as well as the wind tunnel and testing apparatus are discussed in detail in Sections 3.3.2 and 3.3.3.

#### 3.3.1 Table Lookup Method

The rotor power is found through interpolation of table rotor performance data that is derived either experimentally or using a rotor-performance prediction code, for example as done by Carroll [25]. For a given inflow angle, the data is grouped by rotational speeds of the rotor. For each rotational speed, the rotor loads are listed for increasing dynamic pressures. The rotor loads include thrust per density,  $\frac{T}{\rho}$ ,

thrust coefficient,  $C_T$ , power coefficient,  $C_P$ , normal force per density,  $\frac{P_x}{\rho}$ , and roll moment coefficient,  $C_M$ . The roll moment coefficient is included for future research into the stability of the vehicle. Figure 3.7 shows an example of the tabulated thrust per density data for an inflow angle of 0 degrees (freestream parallel to rotor disc) as it is used for the thrust interpolation. In place of thrust and velocity as querying variables for the interpolation scheme, the thrust per density and the dynamic pressure are used in order to easier accommodate for simulations of different altitudes.

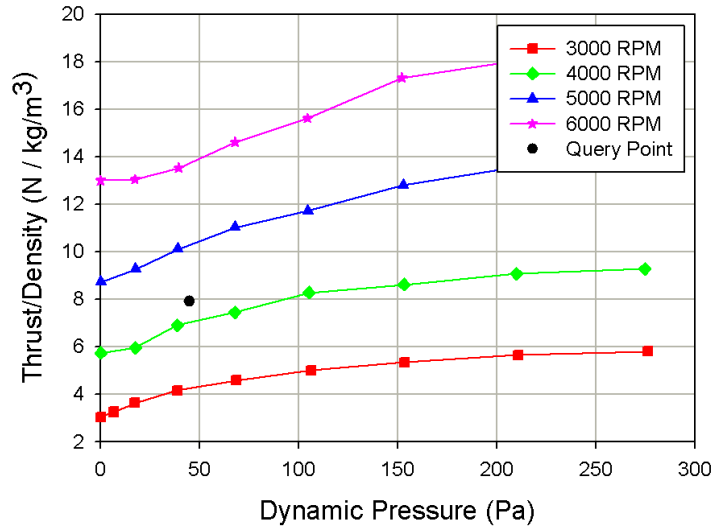


Figure 3.7: Dynamic pressure versus thrust per density for a rotor at 0 degrees angle of attack.

For a known angle of attack, the dynamic pressure, thrust per density, power coefficient, normal force and rotational velocity are found through a three step interpolation method. The algorithm for the multiple interpolation method is shown in Fig. 3.8. The method begins with the required thrust, angle of attack and density as inputs. As the tables are specific to the angle of attack at which the data was collected, the complete interpolation method is performed on the tables bounding the known angle of attack, resulting in two values for each of the performance variables; one set for the upper bounding table and one for the lower.

An example of rotational velocity curves for a rotor at 0 degrees angle of attack is shown in Fig. 3.7. In a first approximation, thrust and rotational speed of the rotor have a quadratic relationship (see Eq. 2.19). Therefore, three data points are used to make a quadratic fit. For that purpose, the curves of the three rotational velocity curves that are closest to the query point are selected for the interpolation

method. In the example of Fig. 3.7, the values of 3000, 4000, and 5000 RPM are used for the quadratic interpolation. Using the known thrust per density, the rotational velocity is found using the following:

$$\Omega = A\left(\frac{T}{\rho}\right)^2 + B\frac{T}{\rho} + C \quad (3.7)$$

With the rotational velocity known, the advance ratio, power coefficient and normal force per density can be found through interpolation. Using the example presented in Fig. 3.7, three power coefficients that correspond to the thrust per density and dynamic pressure at 3000, 4000 and 5000 RPM are selected from the table. Using the known rotational velocity, the three query rotational velocities and three query power coefficients, an interpolation is performed to find the resultant power coefficient. As there is a cubic relationship between  $C_P$  and  $n$  as seen in Eq. 2.20, a cubic interpolation is performed. This method is repeated for the advance ratio and the normal force per density.

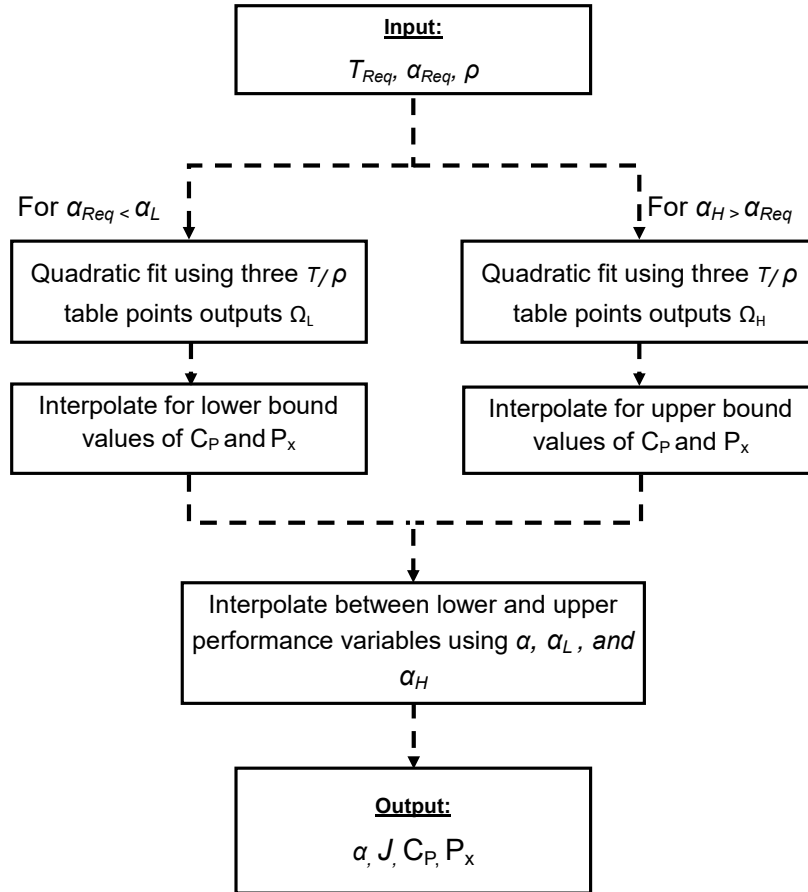


Figure 3.8: Algorithm for rotor performance interpolation method.

A final interpolation is performed between the two data sets using the known angle of attack resulting in the final  $C_P$ ,  $J$  and  $\frac{P_x}{\rho}$ . The normal force per density is multiplied with the density appropriate for that altitude and then using Eq. 2.5, a new angle of attack is found. The table lookup method is repeated, using updated inputs, until the angle of attack converges. The rotor power is found, using the resulting  $C_P$  from the interpolation method, by rearranging Eq. 2.20 for power:

$$P_{rot} = \rho n^3 D^5 C_P \quad (3.8)$$

At the state of this writing, the experimental apparatus used to measure the performance data is not equipped to measure the normal force. Thus, the normal force is calculated using a rotor prediction code as part of the post processing procedure. The two sources will be discussed in the following sections.

### 3.3.2 Rotor Aerodynamic Performance Prediction

An aerodynamic performance prediction model was used to populate the performance tables. The prediction model used combines the blade element momentum theory with two dimensional lookup scheme, allowing for the prediction of performance for both forward flight and hover conditions [25]. The rotor performance prediction method is based on blade element momentum theory (BEMT) and uses a rotor blade that is discretized into elements across its radius, analyzed at each element and integrated to predict overall rotor performance.

In Fig. 3.9, the sectional forces and angles of a two dimensional rotor cross section are shown. The dashed line is the zero lift line, where  $\beta$  is the geometric pitch,  $\alpha$  is the angle of attack and  $\phi$  is the helix angle.  $V_R$  is the resultant velocity vector of the forward flight speed,  $V$  and rotational velocity  $\omega r$ . The rotor rotates about the axis of rotation through azimuth angles  $\Psi$ . The performance of the blade is computed at each azimuth station and integrated over a full rotation of the blade.

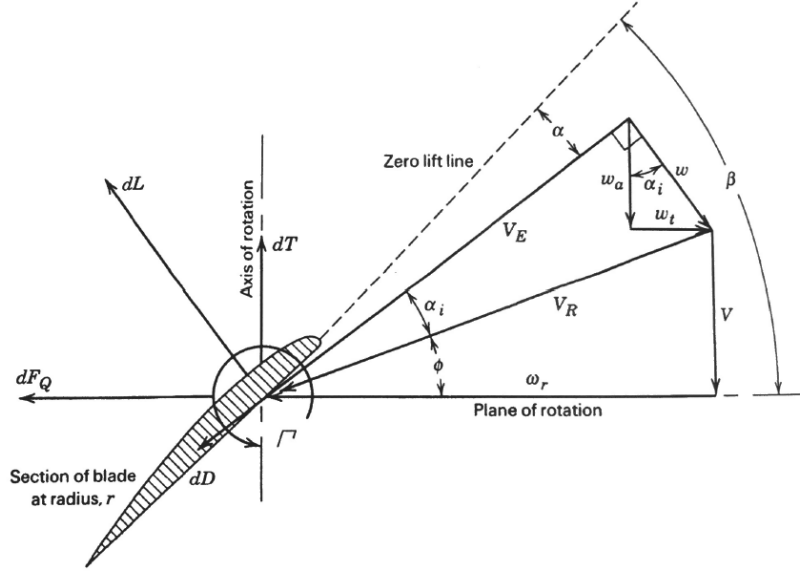


Figure 3.9: Rotor sectional element with acting velocities and forces[19].

The sectional thrust or the sectional lift and drag components, perpendicular to the rotor plane:

$$dT = dL \cos(\phi + \alpha_i) - dD \sin(\phi + \alpha_i) \quad (3.9)$$

The sectional lift and drag forces in the rotor plane result in a torque about the rotor:

$$dQ = r[dL \sin(\phi + \alpha_i) + dD \cos(\phi + \alpha_i)] \quad (3.10)$$

where the sectional lift and drag are:

$$dL = \frac{1}{2} \rho V_E^2 c C_l dr \quad (3.11)$$

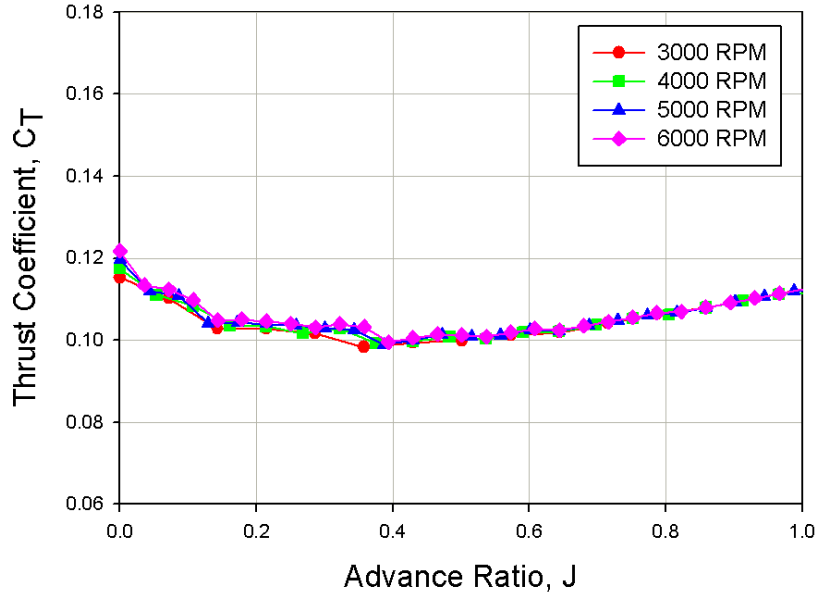
$$dD = \frac{1}{2} \rho V_E^2 c C_d dr \quad (3.12)$$

An interpolation scheme using two-dimensional airfoil data is used to find the sectional lift and drag coefficients required in Eqs. 3.11 and 3.12 respectively [25]. The prediction model uses the wake interference model presented in Section 3.4 to predict interference induced velocities created by shed wakes and rotors in the vicinity. Finally, integrating Eqs. 3.9 and 3.10 and reducing to coefficient forms results in the standard performance equations. The normal force is calculated by integrating the

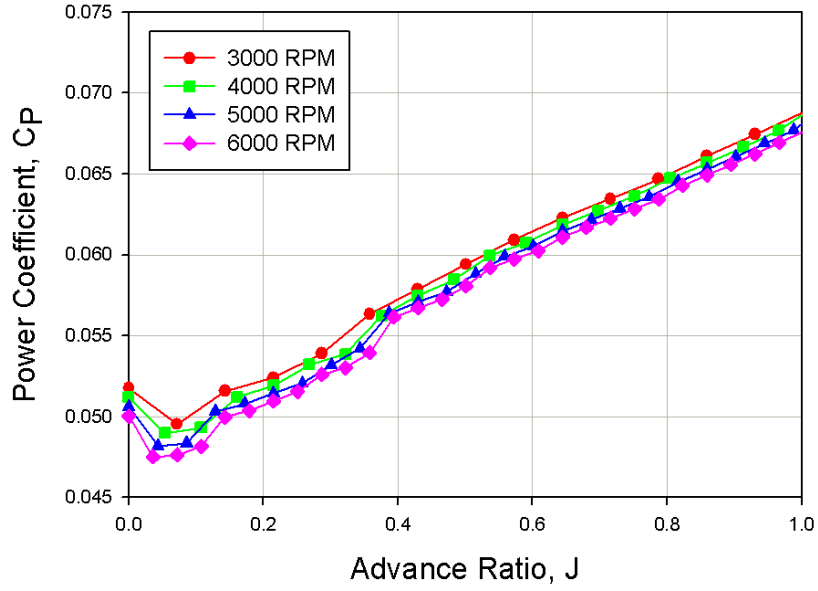
sectional forces calculated at each azimuth station and integrating across the rotor.

Using the blade element momentum theory prediction model, performance data was produced for a Master Airscrew 11x7 rotor. Figures 3.10 and 3.11 show performance data for the rotor at 0 degrees and 30 degrees respectively. Looking at the thrust coefficient versus the advance ratio in Fig. 3.10a, the curves for each rotational velocity lay close to each other showing low dependency on the rotational velocity. As the angle of attack increases, the thrust coefficient decreases for the same advance ratio and rotational velocity. Small performance differences can be observed between the different rotational speeds. These differences are related to Reynolds number effects [25].



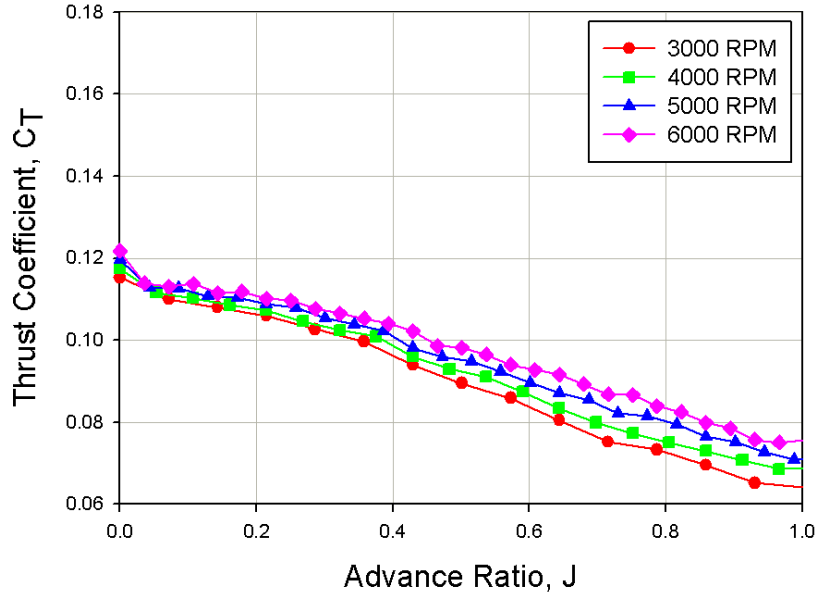


(a) Effect of advance ratio on thrust coefficient.

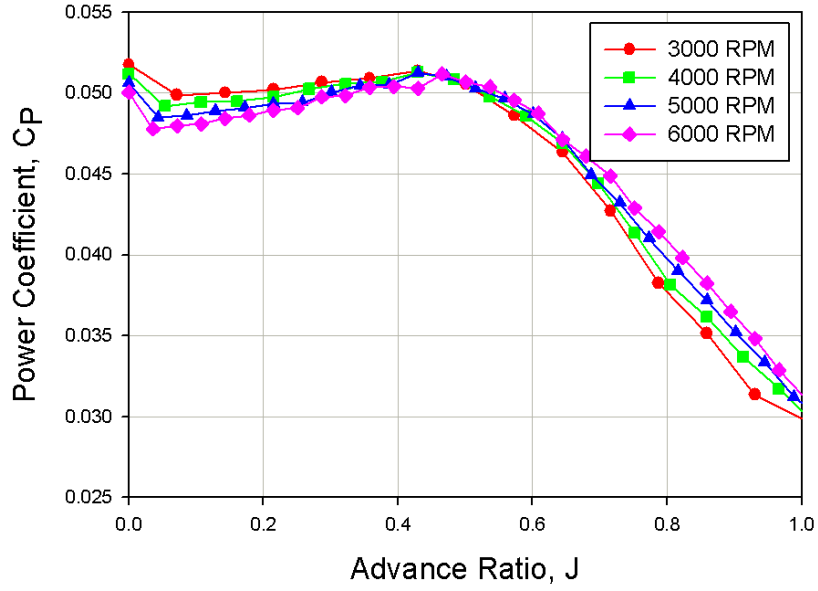


(b) Effect of advance ratio on power coefficient.

Figure 3.10: Predicted performance coefficients for MA11x7 rotor  $\alpha = 0$  deg.



(a) Effect of advance ratio on thrust coefficient.



(b) Effect of advance ratio on power coefficient.

Figure 3.11: Predicted performance coefficients for MA11x7 rotor  $\alpha = 30$  deg.

### 3.3.3 Experimental Rotor Performance

In addition to the theoretical predictions, propeller performance experiments were conducted using the Ryerson University Large Subsonic Wind Tunnel. This section discusses the wind tunnel experiments. The data reduction method is discussed, including any sources of error.

#### Wind Tunnel

Figure 3.12 presents a drawing of the closed circuit wind tunnel. The tunnel has a test section that measures 0.9 m by 0.9 m, 1.5 m long. Screens are placed ahead of the contraction zone of the tunnel to assist flow quality control. Turning vanes assist the flow at each corner. The freestream velocity can reach a maximum of approximately 67 m/s. The performance tables used in the herein presented study consists of data taken at speeds from 0 m/s to approximately 20 m/s.

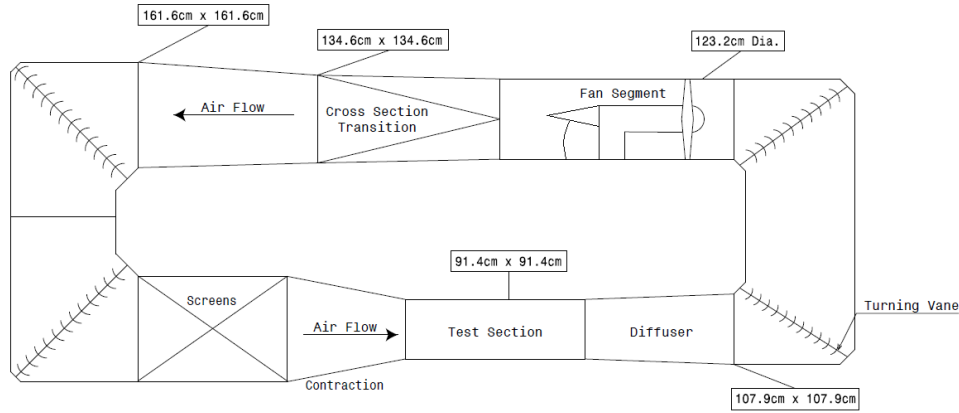


Figure 3.12: Ryerson University Large Subsonic Wind Tunnel drawing[26].

Recent improvements to the tunnel included the replacement of the fan assembly as well as an investigation of the flow quality of the tunnel. A fan nacelle and streamline device seen in the fan segment section of Fig. 3.12 is used to cover the motor belt shaft and was added in efforts to reduce the turbulence intensity. A hot wire anemometry test was conducted to measure the turbulence of the airflow at the centre of the tunnel test section. The addition of the streamline devices reduced the turbulence intensity from 1.2% to 0.29%[26].

### Rotor Test Stand

A propeller test stand was designed, manufactured and mounted in the wind tunnel to collect rotor performance data, such as thrust, torque, and pitching moment (due to advancing/retreating blade effects). The test stand is mounted on a turn table in order to test with oblique inflow angles. The chosen angle of attack convention is shown in Fig. 3.13. At 90 degrees the rotor axis is facing into the free stream, thus the rotor acts as a conventional propeller. An orientation of 0 degrees is fully edgewise flow with the free stream parallel to the rotor plane. The test stand, displayed in Fig. 3.14, is equipped with three load cells to measure thrust, torque and pitching loads developed by the rotor. Note, the pitching moment manifests itself in a roll moment for a multi-rotor vehicle. The rotational speed is measured using an infrared sensor that measures the time between blade passes.

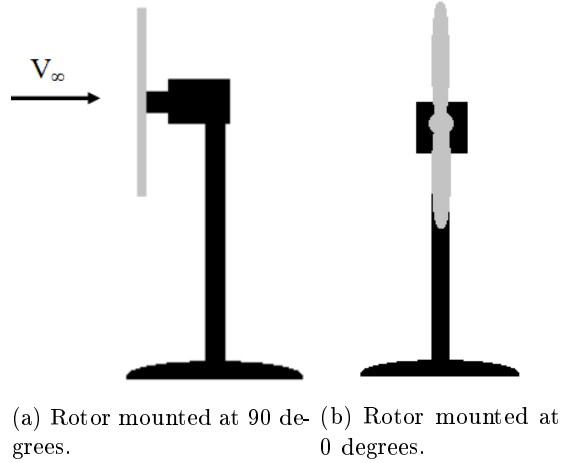


Figure 3.13: Test stand angle convention for propeller testing.

The dynamic pressure in the test section is measured using the static pressure drop across the contraction zone upstream of the test section. Air density is calculated for each data point using the present conditions in the test section. Atmospheric pressure is measured using a laboratory barometer while the wind tunnel air temperature is measured using a thermocouple. The data acquisition is performed using a LabJack T7 data acquisition device equipped with a LabJack CB37 terminal board. The terminal board is used to connect the sensors to the data acquisition system. A detailed description of the experimental procedure can be found in Appendix A.2.

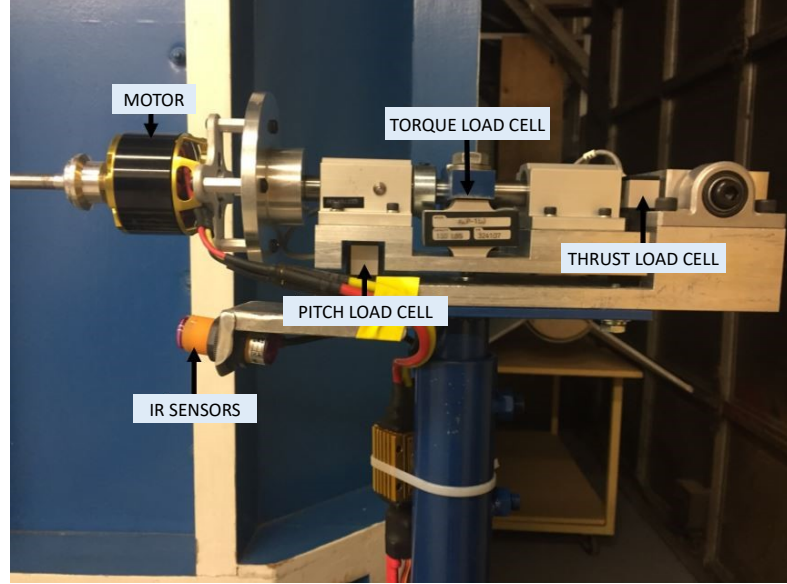


Figure 3.14: Propeller performance testing apparatus with labeled components.

### 3.3.4 Comparisons with Other Experimental Facilities

A comparison study was performed, presented in Fig. 3.15, for a Master Airscrew 11x7 rotor using predicted results from the blade element momentum theory model[25], experimental results from the Ryerson University Large Subsonic Wind Tunnel and published experimental results in the low-speed wind tunnel at the University of Illinois [27]. The coefficient of thrust and power at 5000 rpm for a range of advance ratios are plotted for the experimental and prediction methods. The experimental results published by Brandt are for a rotational velocity of 5006 rpm [27]. The experimental and prediction method show good agreement for the thrust coefficient. The experimental and prediction results trend towards the same static point for the thrust coefficient. The experimental performance results for the power coefficient agree well with the published results, whereas the predicted values capture the same trend, but consistently under predicts the power. This is likely due to the under prediction of section drag coefficients.

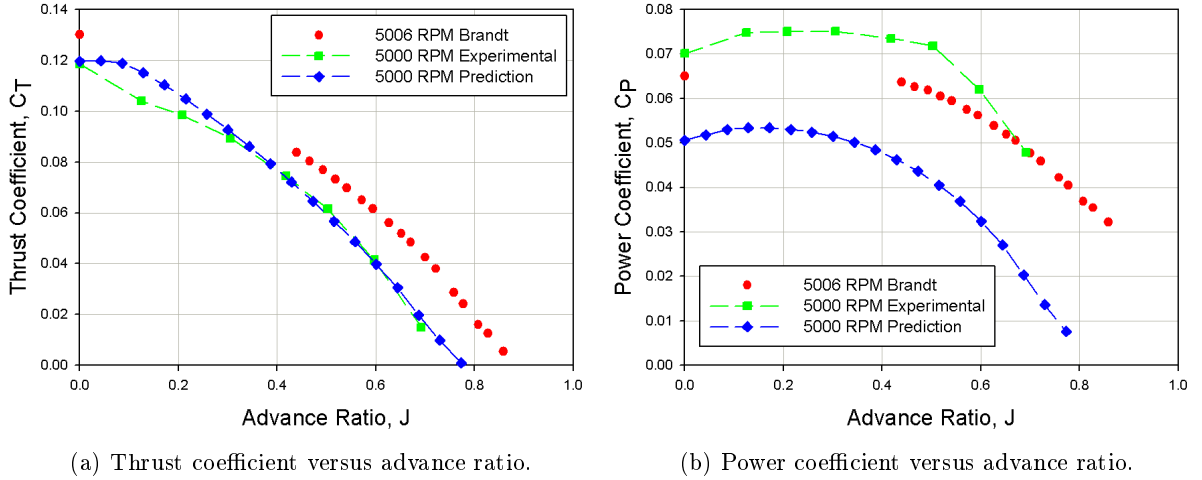


Figure 3.15: Comparison of experimental and predicted MA11x7 rotor performance at 90 degrees.

### 3.4 Wake Interference Module

The wake interference module captures the effect of the wakes shed by the rotating blades in single and multiple rotor configurations. The rotor and its wakes are modelled using a series of vortex rings. An example of this model can be seen in Fig. 3.16 where the ring at the top of the series of rings represents the rotor, while the subsequent rings represent the shed wakes.

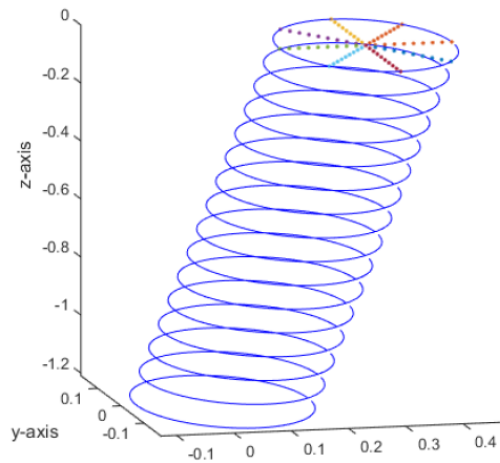


Figure 3.16: Vortex rings representing a series of wakes shed by rotor.

For an arbitrary point, the model is able to calculate the velocity induced by the rotor blades and the velocity induced by the wakes shed from the rotor. The total induced velocity at that point, is a summation of the velocity induced by the rotor and the velocity induced by the shed wakes.

The rotor and rotor wakes are composed of multiple vortex line segments. Each vortex line segment has a circulation which is assumed to be constant across the rotor. Using the Kutta-Joukowski theorem and assuming constant circulation across the blade, the circulation per rotor blade is [28]:

$$\Gamma = \frac{T}{\rho R^2 n \pi B} \quad (3.13)$$

As this is the circulation of one rotor, the thrust used to evaluate the circulation is the thrust produced by one rotor. Using the Biot-Savart rule, the velocity induced at point P by the segment is found as a summation of the velocities calculated for each segment in the vortex ring[29].

As the rotor blades advance, wakes are produced and shed from the rotor plane. These wakes are modeled as vortex rings in planes offset from and parallel to the rotor plane. A wake is shed when a rotor blade completes on full revolution. The time it takes for a rotation to be completed can be expressed as the circumference of the disc area divided by the rotational velocity:

$$t = \frac{2\pi r}{\omega r} \quad (3.14)$$

Simplifying:

$$t = \frac{1}{n} \quad (3.15)$$

The velocity in the z direction at which the wake moves away from the rotor plane is the downwash This can be found as [19]:

$$w = \frac{1}{2} \left( -V_{\infty} \sin \alpha + \sqrt{(V_{\infty} \sin \alpha)^2 + \frac{2T}{\rho A}} \right) \quad (3.16)$$

Together, Eqs. 3.15 and 3.16 express the z coordinate of the vortex ring:

$$z = \frac{1}{n2B} \left( -V_{\infty} \sin \alpha + \sqrt{(V_{\infty} \sin \alpha)^2 + \frac{2T}{\rho A}} \right) \quad (3.17)$$

In Eq. 3.17,  $B$  is the number of blades on the rotor,  $T$  is the required thrust per rotor found in the rotor module,  $A$  is the disc area of the rotor and  $n$  is the rotational velocity.

When a forward velocity is present, the wakes are skewed at angle  $\chi$  as they are shed from the rotor plane. The skew angle, pictured in Fig. 3.17, is measured from the vertical,  $z$  axis, to the center line of the skewed wake and can be calculated as a function of the forward velocity and the vehicle angle of attack as [13]:

$$\chi = \tan^{-1} \left( \frac{-V_{\infty} \cos \alpha}{V_{\infty} \sin \alpha - w} \right) \quad (3.18)$$

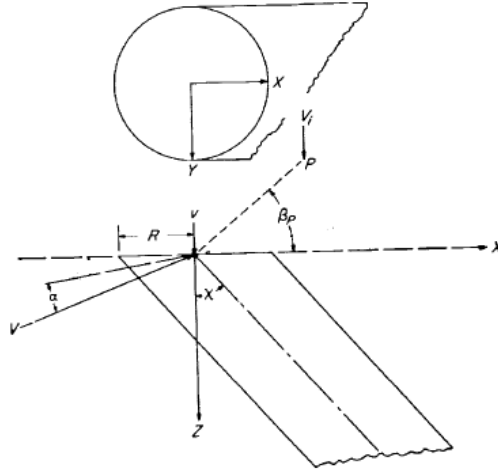


Figure 3.17: Geometry of skewed wake.

To validate the results of the interference model, the induced velocities calculated were compared to analytical results published by Castles and De Leeuw, who found the induced velocity in the flow field is found through numerical integration of complete elliptic integrals[13]. For example, Fig.3.18 shows the normalized induced velocity along the rotor plane for skew angles of 0, 26.56 and 45 degrees comparing the results of the wake interference model and Ref.[13]. For both cases, the rotor radius was set to unity. The induced velocity at each radial section is normalized by the induced velocity at the center of the rotor. The number of rotor segments used per wake segment,  $j$ , and the number of wake rings used,  $k$ , are varied to evaluate the least number of vortex segments per vortex ring and vortex rings are needed to result in a valid solution. This is an important study, as there is a tradeoff between computational speed and accuracy. As  $j$  and  $k$  are increased, the accuracy of the solution increases, however, the



computational effort increases, resulting in a longer solution time.

In order to find a computationally efficient and accurate combination of the number of segments per wake ring and the number of wake rings, a sensitivity study was performed and is summarized in Fig. 3.18. Referring to Fig. 3.18a the number of rotor segments used is set to 100 segments and 200 segments, while the number of wake rings used is set to 0 in order to evaluate the effect of the number of segments for each vortex ring. It can be seen that without the inclusion of wakes, the solution does not fit the published results. In addition, there is little difference in the solution between the two cases in Fig. 3.18a, as compared to the plots in Figs. 3.18b-d, where the number of wake rings that were used is compared. In order to achieve good agreement with Castles and De Leeuw[13], a large number of vortex rings has a greater impact than the number of vortex segments that constitute a vortex ring.

Figures 3.18 b-d show that as the number of wake rings that were used is increased from 100 to 400, the solution converges. When the number of wakes is set to 400, good agreement exists with the analytical results of Ref. [13]. There is a loss in resolution, especially in regions just left and right of the tip of the rotor, as the skew angle increases, although other regions along the rotor plane show good agreement. Referring to Table 3.4, the solution time in seconds for studies shown in Figs. 3.18 b-d are shown. For each angle of attack, the difference in solution time between using 300 wakes and 400 wakes is only about one second, meaning that the penalty for better accuracy is minimal. The times are based on predictions found on a personal computer using an Intel core i7 processor. As a result of the validation study, the number of wakes used in the model is set to 400, while the number of vortex segments per ring is set to 100.

Table 3.4: Solution time for studies in Figs.3.18 b-d

	Solution Time (s)		
	(b)	(c)	(d)
j=100, k=100	0.919	0.934	0.908
j=100, k=200	1.772	1.771	1.757
j=100, k=300	2.637	2.691	2.664
j=100, k=400	3.513	3.565	3.576

The wake interference model is used to investigate the effect of vehicle orientation during forward

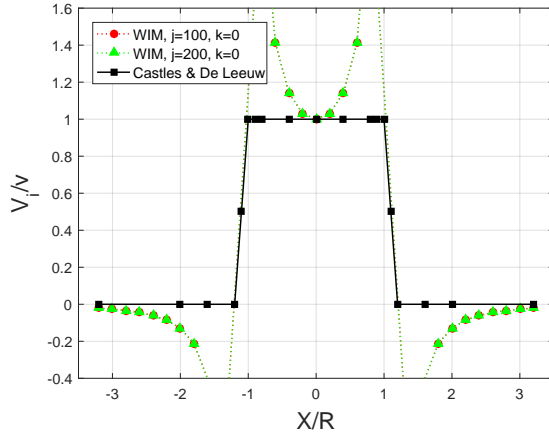
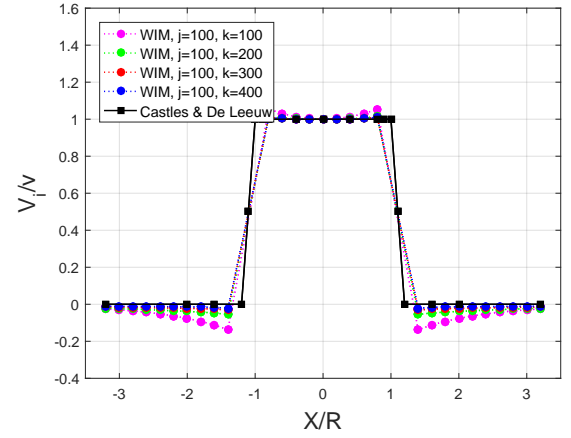
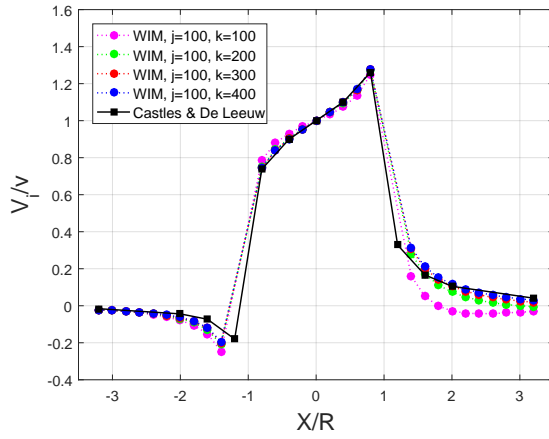
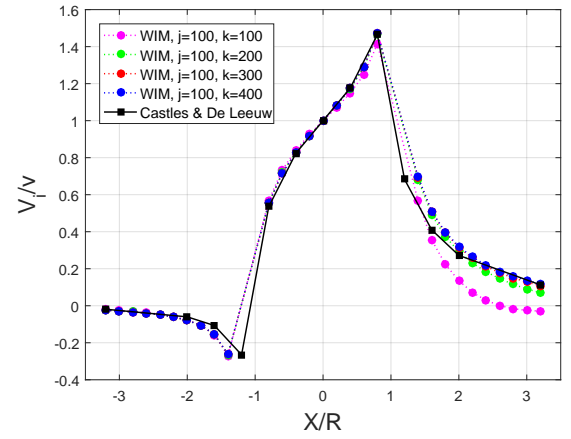
(a)  $\chi = 0^\circ$ , no wakes(b)  $\chi = 0^\circ$ (c)  $\chi = 26.56^\circ$ (d)  $\chi = 45^\circ$ 

Figure 3.18: Wake interference model results comparison with analytical results.

flight on the power required. Interference power is dependent on the relative proximity and location of the rotors to each other, therefore, the orientation of the rotors in the system with respect to each other and with respect to the direction of flight is important. These effects are investigated in the case studies presented in Chapter 5.

## Chapter 4

# Multicopter Vehicle Performance Model Validation

To determine the proper functioning of the model, predicted performance data was compared to flight-test data provided by Aeryon Labs Inc. in Waterloo, Ontario. Aeryon provided flight-test data for their SkyRanger model. The data were collected over a series of flights by the onboard sensors and processed in order to compare it to the predicted results. The flight-test data were provided without environmental data, for example wind speed and direction, and therefore required filtering to achieve a better comparison.

As a method of determining the proper functioning of the model, predicted values of the rotational velocity, power required per rotor and vehicle angle of attack over a flight-speed range were compared to flight-test data. The flight-test data position, pitch angle, acceleration and airspeed are measured as a combination of GPS data and accelerometer data. The flight-test data power per rotor is a measure of how much power is drawn from the battery, while the rotational velocity per rotor is provided by the electronic speed controller.

The flight-test data was filtered with the purpose of removing errors and effects that are not present in the prediction model. For example, the prediction model assumes there is no acceleration present in the positive x-direction. To replicate steady state motion in the flight-test data, data points with an acceleration of more than  $0.15 \text{ m/s}^2$  were excluded. As the model makes predictions for specific

velocities, a range of  $\pm 0.2$  m/s was allowed for the flight-test data airspeed.

Predicted results were obtained for a model of the Aeryon SkyRanger using experimental ASP130 rotor performance data at an altitude of 346 m. As discussed, lowering the allowable forward flight acceleration trimmed the flight-test data, removing non steady state data points from comparison. A trend in the flight-test data is present in Fig. 4.1 where a comparison of the predicted and measured vehicle angle of attack in degrees is shown. The model captures the trend of the flight-test data as the pitch angle increases with the velocity. An under prediction in the parasitic drag, as the components geometry was simplified to common shapes, can account for the low under prediction in the angle of attack.

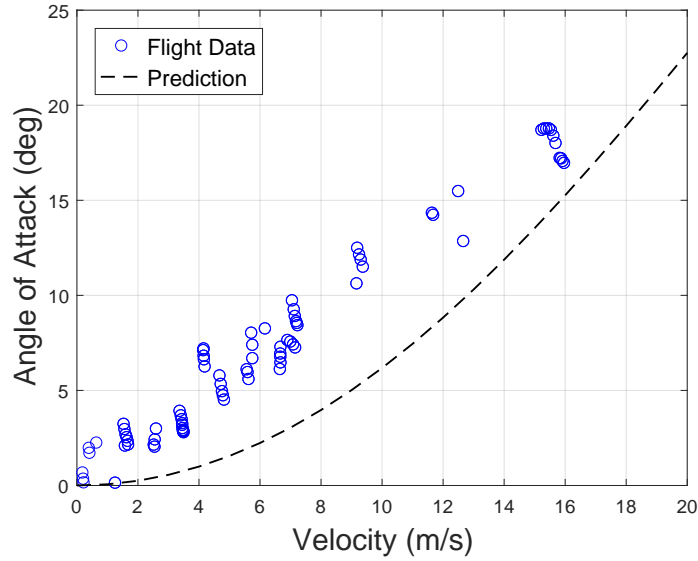


Figure 4.1: Comparison of predicted and measured vehicle angle of attack versus forward velocity.

The flight-test data rotational velocity and power are measured per rotor, though only the total power required is predicted per rotor. For each data point, an average of the measured rotational velocity and power is found. Standard error bars are calculated to mark the standard deviation of the mean, where a smaller error bar indicates a smaller deviation from the mean [30]. The predicted and measured rotational velocity and power per rotor are plotted versus the angle of attack of the vehicle and are presented in Fig.4.2 and Fig. 4.3 respectively. The values of the rotational velocity and rotor power are expected to be under predicted as errors in both values are introduced through a lack of a

motor model. The motor efficiency, which accounts for mechanical losses in the motor, is not included in the model and thus, the prediction for the power and rotational velocity will be less than the recorded flight-test data. The experimental data measured propeller torque, not engine torque, and therefore no engine losses are included.

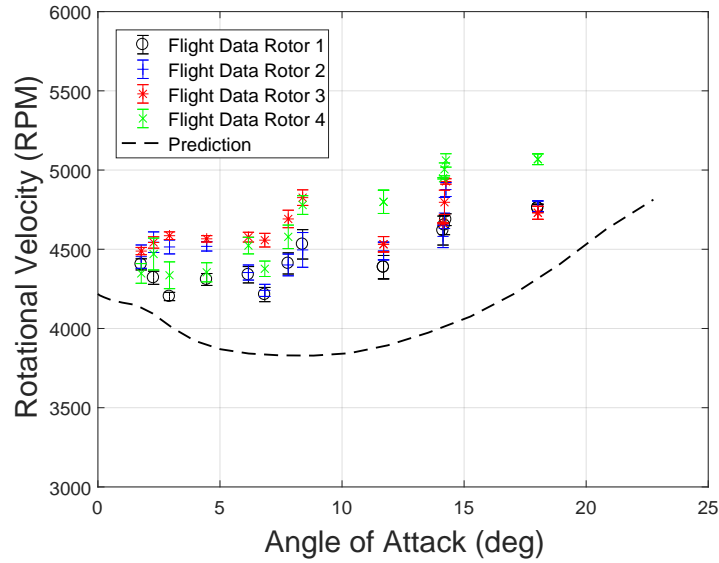


Figure 4.2: Comparison of predicted and measured rotational velocity versus angle of attack.

Referring to Fig. 4.3, as the vehicle is in the diamond configuration (Fig. 3.2), the predictions for Rotor 2 and Rotor 4 are equivalent. The prediction and flight-test data both show higher power requirements over the range for the rear rotor, Rotor 1. Many of the points with large error bars, which indicates less confidence in its value, lay off of the predicted lines. In hover, low speeds and around 8 m/s, there is good agreement with the flight-test data.

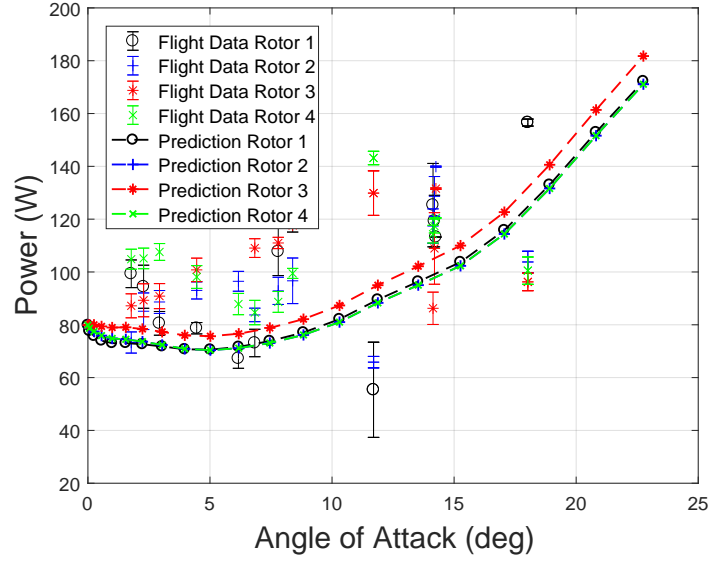


Figure 4.3: Comparison of predicted and measured power versus angle of attack.

Overall, the predicted trends in the angle of attack, rotor power and rotational velocity agree well with measured flight-test data, showing the model is capable of predicting the performance of multirotor sUAS. In the following section, case studies consisting of performance predictions using the model are presented.

## Chapter 5

# Case Studies

A number of case studies are presented in the following subsections consisting of performance predictions created using the program described in Chapter 3. Each of the studies highlight a component of the power required and explores its contribution to the overall vehicle performance. In turn, the functionality and flexibility of the vehicle performance model as a design tool is demonstrated, as well as the utility of power decomposition methods.

### 5.1 Effect of Altitude

The performance of an Aeryon SkyRanger quadrotor equipped with T-Motor 18 by 6.1 inch rotor was predicted using the aforementioned prediction model. The power decomposition of a four rotor vehicle based on the Aeryon SkyRanger is shown in Fig. 5.1. The predictions are for flight at an altitude of 1000 ft above sea level in the diamond configuration and are shown per rotor. The total power required is represented by the solid line and is a summation of its components; the parasitic power, interference power and rotor power. Noting the rotor number convention presented in Fig. 3.2, as Rotor 2 and Rotor 4 are opposite each other about the symmetry line, they both have the same power requirements and thus, identical power decomposition plots.

All rotors are assumed to have the same power requirements and thus, the rotor power is identical between plots. The rotor power is the main contributor to the total power in hover, but decreases over the flight-speed range as the induced power decreases (Eq. 2.11). As the rotor tilts forward during



increasing forward flight, the rotor must work harder to meet power requirements and therefore, the rotor power increases. The parasitic power is equal across rotors, as the total drag is shared between the rotors. The parasitic power is zero in hover (Eq. 2.10), but becomes more influential at high speeds as it is proportional to the velocity cubed.

The interference power for Rotor 2 and Rotor 4 is negative, resulting in a decrease in the total power required. This is a result of the neighbouring rotors and their wakes having an upwash effect on Rotors 2 and 4. The rear rotor, Rotor 3, has a positive interference power that increases over the flight-speed range. As the vehicle increases its forward velocity, the angle of attack increases as well as the skew angle of the wakes shed from the rotor plane. The wakes shed from Rotors 1, 2 and 4 are positioned close to the rear rotor plane at high velocities, causing a downwash effect for the rear rotor. This interference increases the total power required for the rear rotor.

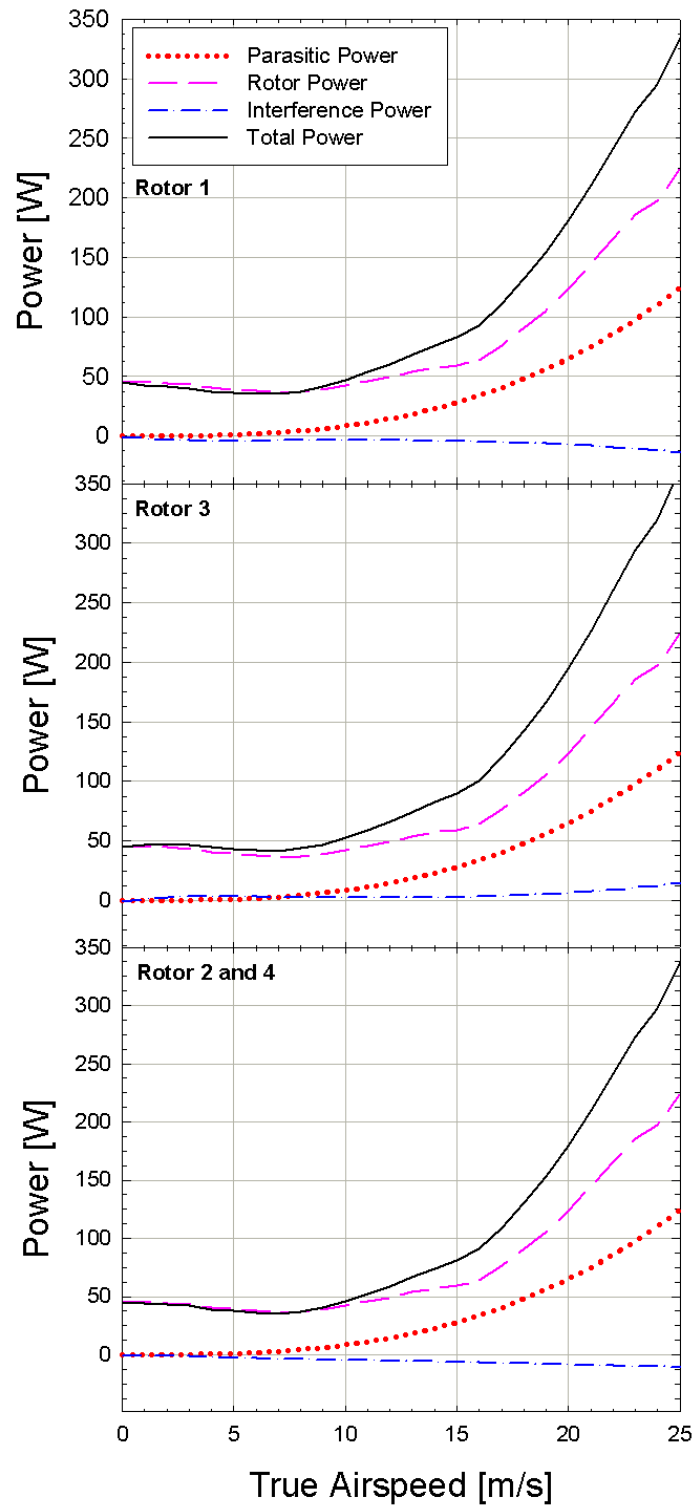


Figure 5.1: Experimental performance prediction for Aeryon SkyRanger using T-Motor rotors at 1000 ft.

The effect of altitude on the overall vehicle performance was investigated using the prediction model and experimental performance data. For the same vehicle and rotor geometry, performance predictions were calculated at 1000 feet, 2500 feet and 5000 feet. The rotor power at each altitude is shown versus the forward flight speed in Fig. 5.2. In hover and at low airspeeds, the vehicle requires more power to operate as the altitude increases. This is due to the reduction of air density as the altitude increases. For higher airspeeds, the lower air density results in a lower parasitic drag, and thus power, lowering the power requirements at high altitudes.

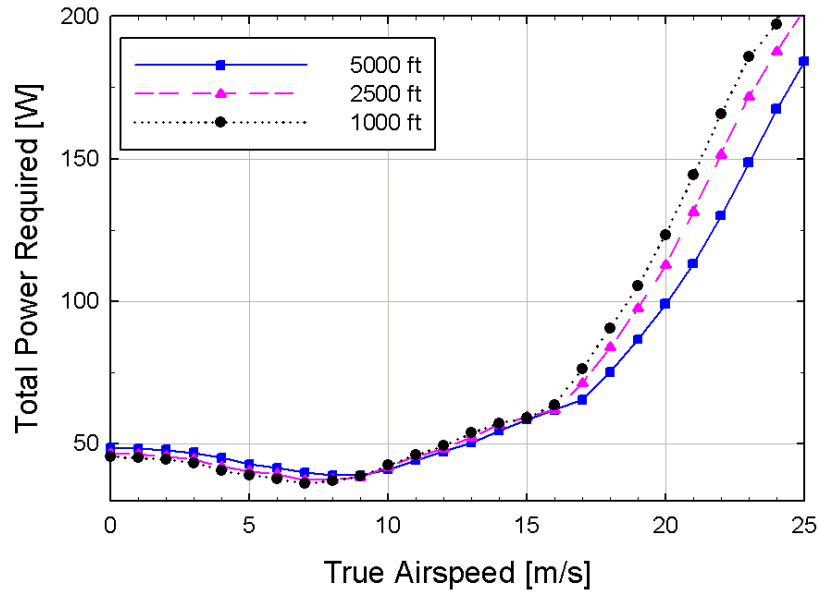


Figure 5.2: Rotor power predictions versus forward flight speed at various altitudes for T-Motor rotor.

This study demonstrates that the model is capable of providing a means of effectively analyzing the performance of an sUAS over a flight-speed range and for different flight altitudes. By predicting the performance of the vehicle at various altitude, it is shown that low speed and hovering flight should be performed at low altitudes and fast forward flight performed at high altitudes to achieve the best power requirements. In addition, the sUAS in this configuration performs better in low speed and hovering flight, than during fast forward flight.

## 5.2 Effect of Weight

In this study, the power required is investigated as a function of the weight. A T-Motor 18 inch by 6.1 inch rotor is used in this example on the Aeryon SkyRanger body. The predictions are for an altitude of 1000 ft and the sUAS is in the standard diamond configuration, that is Rotor 1 is in the lead and Rotor 3 in the rear. Rotors 2 and 4 are to left and right of the symmetry plane of the flight vehicle.

To decouple the effect of the parasitic drag from the weight, the physical geometry of the sUAS is not changed, leaving the parasitic drag equivalent across data sets. The first weight is the weight of the sUAS with no payload added and is about 3 kg. The second weight represents a camera mounted on the sUAS and is a total weight of 3.62 kg. Finally, the third weight represents an additional battery added to the system for a total weight of 4.2 kg.

Quite obviously and supported by Eq. 2.6, a higher thrust is required to operate if the weight of the sUAS is increased. This relationship is also shown in Fig. 5.3 that shows the thrust per rotor versus the forward flight speed for the aforementioned sUAS with and without payload and an extra battery is plotted. In all cases, as the forward velocity of the quadcopter is increased, the thrust increases as well.

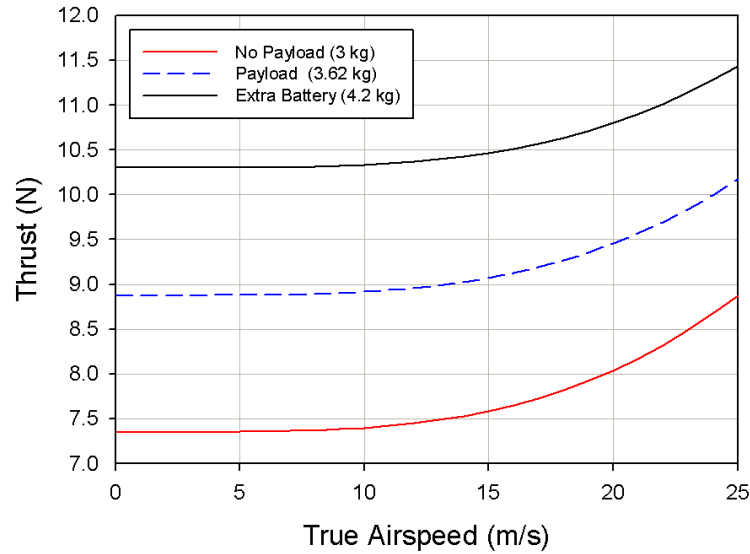


Figure 5.3: Thrust per rotor versus forward flight speed for sUAS with three different flight masses.

The change in the weight of the vehicle also affects the angle of attack as seen in Fig. 5.4, which indicates lower angles of attack over the entire flight range for the vehicle with a payload added. This

relationship is supported by rearranging Eq. 2.6 for angle of attack, showing an increase in the weight, and thus the thrust, results in a decrease in the angle of attack. This is further shown in Fig. 5.5, that shows force diagrams for two sUAS with the same parasitic drag, but different flight masses. To compensate for the increase in weight, the thrust vector must be increased, resulting in a decrease in the angle of attack.

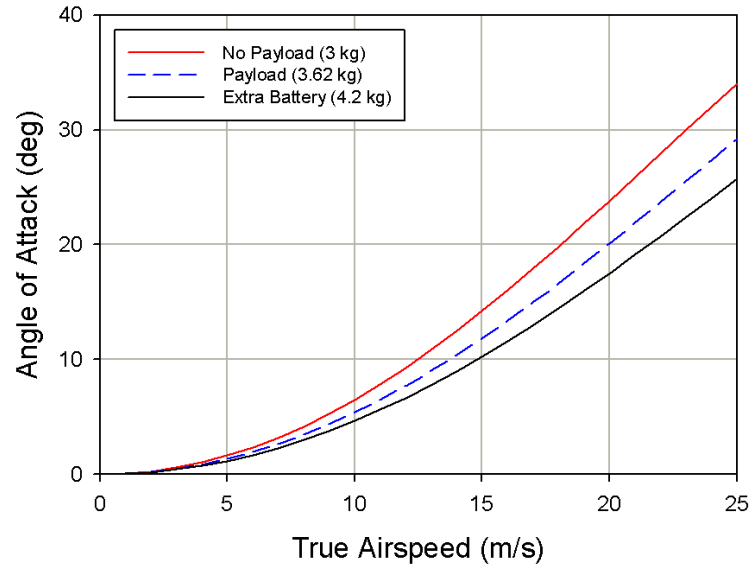


Figure 5.4: Angle of attack versus forward flight speed for sUAS with three different flight masses.

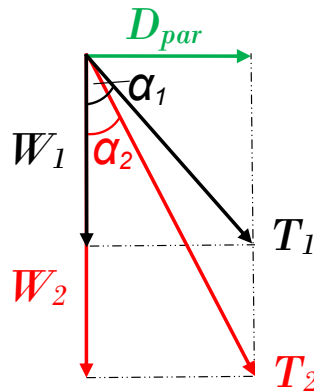


Figure 5.5: Force triangle depicting effect of weight on vehicle angle of attack.

Finally, Fig. 5.6 shows the effect of the weight on the rotor power required. As discussed in Sec. 2.3, the rotor power consists of the profile power and the induced power. In hover, the portion of the power required that is attributed to the induced power is at its greatest. As the forward flight speed increases, the induced velocity and thus the induced power decreases up to a certain speed [17]. The crossover that occurs around 24 m/s is due to incomplete airfoil data in the rotor performance table. Above a certain speed, thrust loading, on which induced power depends, increases as the rotor must increasingly compensate for the parasitic drag that grows with the velocity squared.

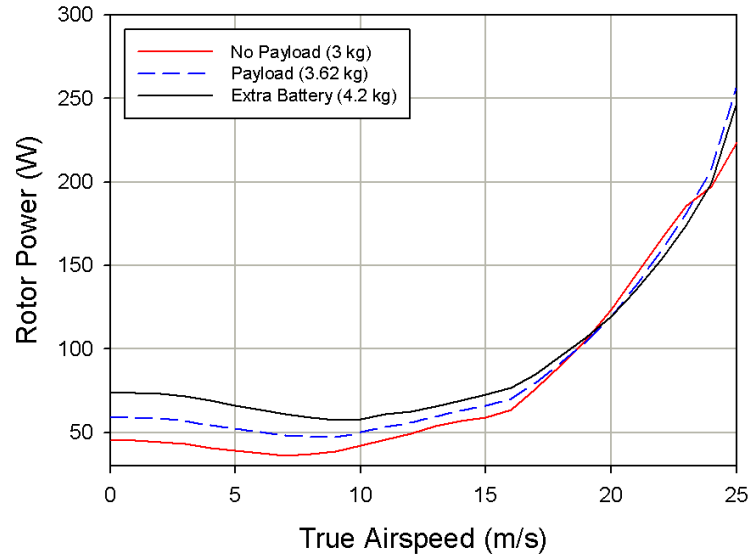


Figure 5.6: Rotor power versus forward flight speed for sUAS with payloads and empty weight.

In conclusion, the effect of weight on the performance of an sUAS was investigated. Through this study, the capability of the model as a design tool was shown. For example, based on the battery capacity and the motor efficiency, designers can determine if the additional power available provided by an extra battery in the system is worth the added weight.

### 5.3 Effect of Orientation in Forward Flight

Another characteristic of the vehicle performance that can be explored using the aforementioned analysis tool is the effect of the orientation of the vehicle in forward flight. Referring to Fig. 3.2, two orientations

are presented, the square configuration where two rotors are leading and the diamond configuration, where one rotor leads. In this case study, performance predictions over a flight range for the geometry of the Aeryon Skyranger using experimental T-Motor rotor data were obtained for both square and diamond configurations for an altitude of 1000ft. The total power required for the vehicle in each configuration is presented in Fig. 5.7. Comparing the performance in the two configurations, more power is required to operate in the square configuration than in the diamond configuration over the flight-speed range.

As the inflow from the neighbouring rotors is not included in the rotor performance prediction, the value of the rotor power does not change between orientations. The parasitic power will also be constant between the two configurations. Therefore any difference in the power required is a result of the change in the interference power. In hover, the two orientations require the same amount of power since the induced interference power is zero during hovering flight (Eq. 3.16). As the forward velocity is increased, the interference power increases more in the square configuration, than in the diamond configuration, resulting in a higher total power over the flight-speed range in the square configuration. This can be explained by investigating the interference power per rotor.

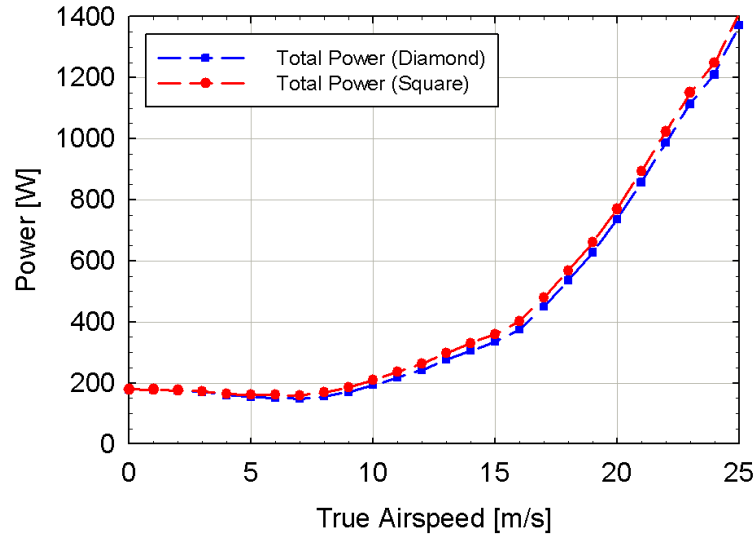


Figure 5.7: Total power required for diamond and square configuration using T-Motor rotor experimental data.

The total power required and interference power for each rotor in both configurations are presented in Fig. 5.8 to illustrate the difference in rotor power between the two configurations. Noting the line of

symmetry in each configuration in Fig. 3.2, Rotors 2 and 4 have the same power requirements in the diamond configuration. In the square configuration, Rotors 1 and 2 and Rotors 3 and 4 have the same power requirements.

The orientation of the vehicle in forward flight is important, as the wakes shed from the rotor plane are in different locations in the two orientations. Therefore, their interactions with neighbouring rotors differ in the two cases since position effects the velocity induced by the wakes. For example, wakes positioned farther away from a given point will induce a smaller velocity at that point.

In the square configuration, Rotors 3 and 4 act as the rear rotors. As supported by Eq. 3.17, as the vehicle speed increases, the skew angles of the shed wake increases as well. The shed wakes of the front rotors, Rotors 1 and 2, pass underneath the rear rotors, increasing the power required as they are caught in the downwash of the wakes. This is supported by Fig. 5.8 which shows a positive and equal interference power for Rotors 3 and 4 in the square configuration, resulting in an increase in the power required. Note the negative interference power for the front rotors, Rotor 1 and 2, in the upwash of the neighbouring rear rotors.

In the diamond configuration, Rotor 3 acts as the rear rotor. As there is only one rotor in line with the rear rotor in the diamond configuration, a positive interference power that is smaller than in the square configuration is present. The remaining rotors in the diamond configuration all experience a negative interference power, resulting in a decrease in the total power required for each rotor. This combination of effects results in a smaller total power required in the diamond configuration as compared to the square configuration as presented in Fig. 5.7.

From this study, the effect of the vehicle orientation on the performance of the vehicle was investigated. While in hover and low speeds, the power requirements for the two configurations are about the same, during fast forward flight, the sUAS requires about 5% less power in the diamond configuration than in the square configuration. Therefore, fast forward flight should be performed in the diamond configuration.



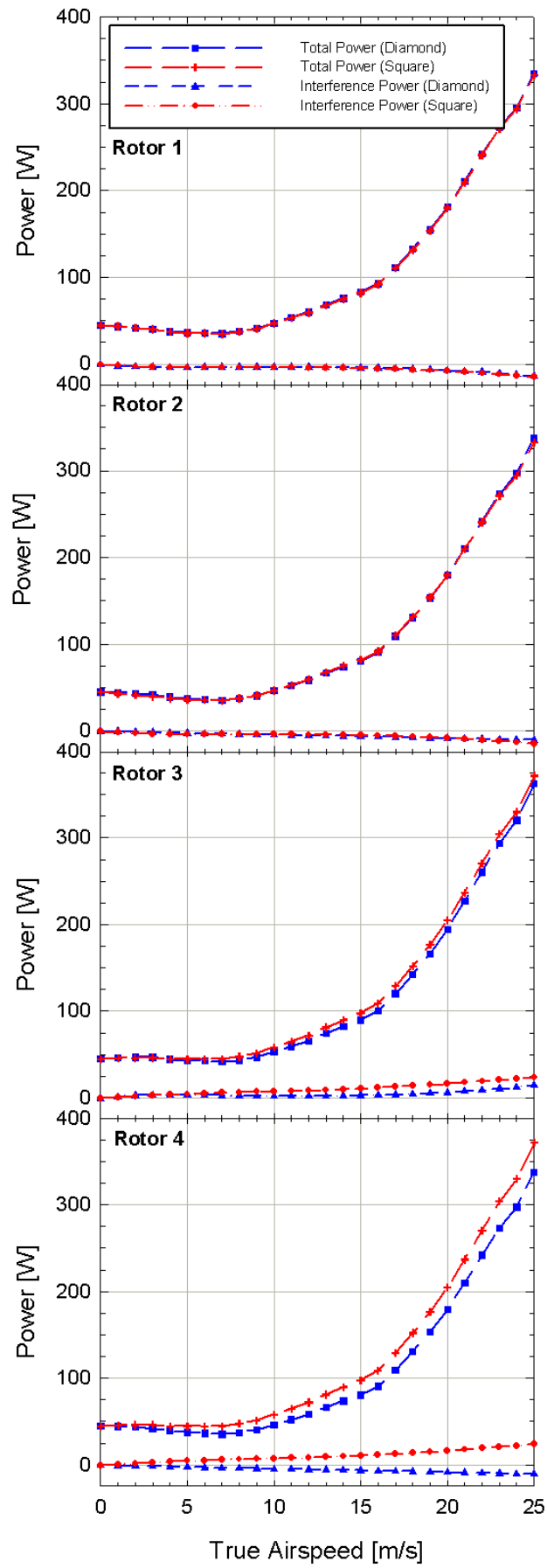


Figure 5.8: Rotor power required and interference power for diamond and square configuration using T-Motor rotor experimental data.

## Chapter 6

# Conclusions

This thesis presents a method for predicting the performance of multirotor sUAS in hover and forward flight, with the purpose of aiding the design process of multirotor sUAS by allowing designers to assess the contributions of the vehicle and rotor geometry, as well as atmospheric conditions on the performance of the vehicle. The use of an interpolation scheme with tabulated rotor performance data allows for flexibility, as the data can be provided using experimental or predicted rotor data. The method uses a wake interference model, that captures the effect of rotors and the wakes. The modular approach allows designers to easily assess the contributions of the vehicle configuration, choice of rotor system and atmospheric conditions on the performance of the vehicle. The study shows that the model is suited for the iterative conceptual design stage of product development.

When comparing predictions produced by the method with flight data measured from an Aeryon SkyRanger sUAS the predictions show good agreement in the trends for the rotational velocity, angle of attack and power required. The model captures the trend in the angle of attack during forward flight, though under predicts the value due to the under prediction of the parasitic drag. While the trends in the rotational velocity and the power required are captured, the values are under predicted due to the lack of a motor model.

Case studies have been presented in this thesis that explore the capabilities and functionality of the method. The method is capable of evaluating the effect of altitude on the performance of the vehicle. This feature of the model allows designers to assess the performance of a design in different conditions, as well as select the right rotor for the altitude. The effect of weight on the vehicle is also discussed;

results of the case study show the model can be used as a design tool to assess the effect of additional components to the vehicle. Finally, the method is used to assess the performance of a quadrotor sUAS in two different orientations, showing that the diamond configuration results in the least power required during forward flight. This study can be extended to evaluate the performance of co-axial rotors, as well as more complicated rotor configurations.

It is suggested from the findings of these studies that fast forward flight be performed at high altitudes, while hover and slow flight be performed at low altitudes. In addition, forward flight should be performed in the diamond configuration since it was shown that 5% less power is required than in the square configuration.

Although the current method for predicting the parasitic drag of the vehicle is proven to produce a good estimate, the accuracy of the model can be improved by using measured wind tunnel drag data. A wider range of scenarios, for example climbing and descending flight, can be analyzed with the addition of a full trim solution and an inflow model. These improvements to the current tool will make it more robust and more applicable to a broader range of problems.

# Appendix A

## Wind Tunnel Data

### A.1 Uncertainty Analysis

The uncertainties examined include those present in the measurement devices for wind tunnel velocity, propeller thrust, torque, bending moment and power. The uncertainties of the sensors used were determined based upon data and calibration sheets provided by the manufacturers of each device and is presented in Table A.1.

Table A.1: Sensor Uncertainties

Sensor	Uncertainty
Static Pressure	0.217%
Differential Pressure	1.503%
Load Cells	0.688%
RPM Sensor	0.491%
Thermocouple	1.301%

A general uncertainty analysis was performed using the Taylor Series Method (TSM) as shown by Coleman and Steele[31]. This includes the uncertainties present in the measurement devices used as well as that of the data acquisition system, all of which contribute to the uncertainty of measurements. All uncertainty estimates presented were determined using the Taylor Series Method and result from the

reduction equations used for the analysis of collected data. The uncertainties in the variables are listed in Table A.2. The uncertainties observed are within an acceptable range, all less than 5 percent, although there is room for improvement. For example, the uncertainty in both the thrust and the power coefficient can be decreased by decreasing the uncertainty in the infrared sensor used to measure the rotational velocity of the propeller. Future work is expected to include an upgrade to the testing apparatus through the use of higher accuracy sensors.

Table A.2: Uncertainties in Variables

Variable	Uncertainty
Temperature (T)	1.302%
Air Density ( $\rho$ )	1.320%
Freestream Velocity ( $V_\infty$ )	1.000%
Propeller Rotational Speed (n)	0.491%
Advance Ratio (J)	1.114%
Coefficient of Torque, Thrust & Pitch ( $C_Q, C_T, C_M$ )	1.783%
Torque, Thrust & Pitch	0.688%
Power (P)	0.845%
Power Coefficient ( $C_P$ )	2.151%
Efficiency	3.008%

## A.2 Experimental Procedure

The experimental procedure begins with the positioning of the propeller test stand. The desired angle of attack must be set before zeroing the system. It is selected by rotating the turn table base until the angle dial is in the correct position. The bolts securing the turn table must be carefully tightened to reduce vibration in the stand. Once all sensors are connected to the LabJack data acquisition system, a software tare is performed to zero the load cells. The ambient pressure is recorded prior to testing. Before forward flight testing is conducted, static tests are performed and compared to existing data to ensure the system is working correctly. For forward flight testing, the propeller motor throttle, and thus the

rotational velocity, is kept constant while the wind tunnel velocity is swept through a range, collecting a variety of data points. Time is given to allow for the tunnel velocity to reach steady state before data points are collected. The raw data is saved, with the tunnel freestream and rotational velocity for each data point recorded, and is then post-processed using a MATLAB script.

### A.3 Data Reduction

It is important that the raw data collected during wind tunnel experimentation is properly post processed with all system and external errors accounted for. Raw data is taken for each angle of attack, holding the wind tunnel velocity constant while changing the rotational velocity of the propeller to sweep through advance ratios. The advance ratio (Eq. 1.1) is used as the independent variable when plotting performance coefficients and is the ratio of forward flight speed to the speed of the rotor tip. Blockage corrections are not as the diameter of the rotor is relatively small compared to the diameter of the test section.

For thrust, pitching moment and torque a basedrag correction is applied to the measured value after conversion to metric units. The basedrag is defined as the drag created by the propeller test stand. This drag is picked up by the load cells and therefore, must be removed from the measurements taken with the propeller in use. For each angle of attack, static data was collected across a range of wind tunnel velocities to be used in a basedrag lookup table. A linear relationship between the dynamic pressure and the force/moment is assumed. Figure A.1 shows an example of the measured basedrag versus the dynamic pressure for an angle of attack of 90 degrees.

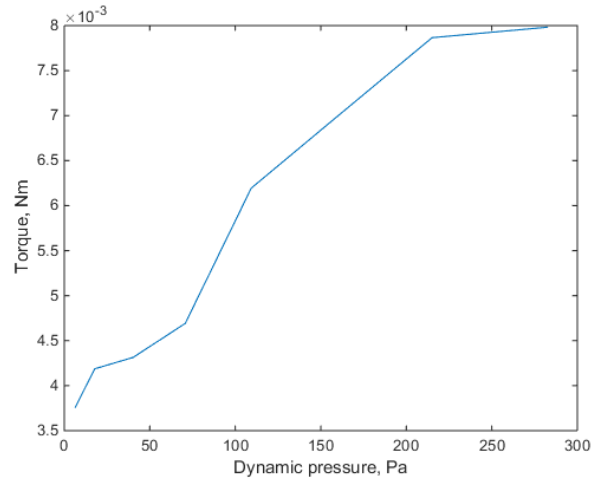
For each dynamic pressure measured, a corrected basedrag for thrust, pitching moment and torque can be found using linear interpolation. The basedrag values are denoted by subscript "cc" while the corrected values are denoted by subscript "c".

$$T_c = T - T_{cc} \tag{A.1}$$

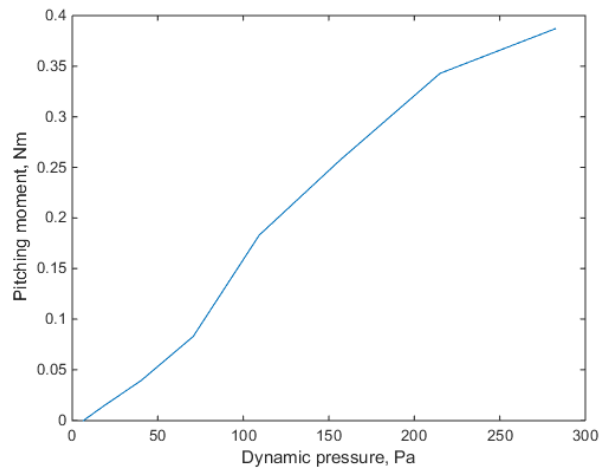
$$M_c = M - M_{cc} \tag{A.2}$$

$$Q_c = Q - Q_{cc} \tag{A.3}$$

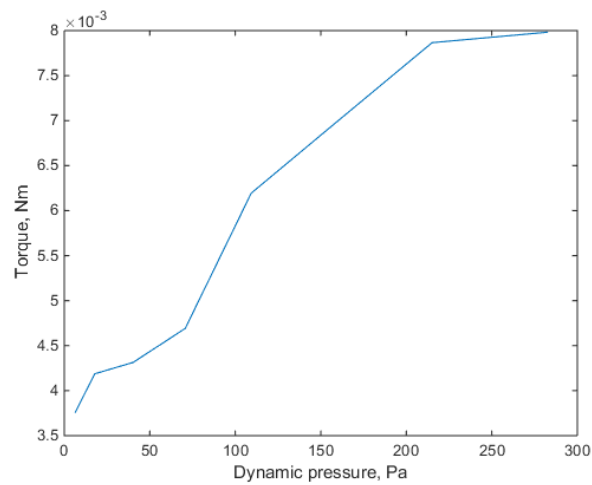
In the future, the testing apparatus will be modified to allow for the measurement of the normal force produced by the rotor,  $P_x$ .



(a)



(b)



(c)

Figure A.1: Basedrag measurements for a) thrust, b) pitching moment and c) torque versus dynamic pressure at an angle of attack of 90 degrees.



## Appendix B

### Rotor Performance Tables

---

APPENDIX B. ROTOR PERFORMANCE TABLES

---

Table B.1: Rotor performance table for T-Motor Rotor at  $\alpha = 0^\circ$ .

q	Thrust_rho	RPM	Jinf	CT	CP	CM	Px_rho
0	8.54349309	3000	0	0.078211493	0.024272791	-0.002627075	0
8.210870197	8.859720582	3000	0.16424924	0.081106401	0.024278662	-0.0055131	0
20.05205347	9.215707967	3000	0.255471372	0.083527924	0.023433798	-0.00711574	0
38.88100951	9.978108509	3000	0.359007871	0.09207986	0.025078327	-0.008375046	0
68.31143892	11.51111627	3000	0.471312686	0.104125376	0.025094862	-0.009333661	0
105.643329	12.02951109	3000	0.590850439	0.110492281	0.026026067	-0.011758124	0
147.8772919	12.1	3000	0.699586727	0.112	0.026817055	-0.01522984	0
197.0698002	12.38135026	3000	0.809506545	0.114104625	0.025823989	-0.019117718	0
252.7711647	13.25258066	3000	0.91404584	0.121159283	0.026191555	-0.022513539	0
314.1924952	13.47156094	3000	1.026358933	0.124568107	0.026178301	-0.025937143	0
380.8685323	14.02468126	3000	1.125202819	0.128047398	0.023950175	-0.029638149	0
0	14.44388487	4000	0	0.073639279	0.023232157	-0.00178737	0
7.745051977	14.5	4000	0.119990734	0.074	0.02463407	-0.002827206	0
18.96196968	15.13511357	4000	0.187463044	0.077317159	0.023269547	-0.004878235	0
38.88812104	15.82266037	4000	0.268377494	0.080829466	0.023659802	-0.005796609	0
67.71579168	16.58338563	4000	0.355215741	0.08522415	0.024505184	-0.0057884	0
103.998803	17.27324895	4000	0.441090443	0.089080644	0.024412163	-0.007391994	0
146.8539982	18.07280853	4000	0.525282879	0.093531408	0.025815081	-0.009600317	0
194.8548393	19.73315999	4000	0.604006948	0.101614169	0.025421156	-0.012114115	0
249.68926	19.92017971	4000	0.678384181	0.100805398	0.02511969	-0.01308125	0
310.1612641	21.0389332	4000	0.763921558	0.108338133	0.024865818	-0.014259343	0
378.9611214	20.89884532	4000	0.840428477	0.10628406	0.023368421	-0.017149882	0
0	23.56127033	5000	0	0.077339412	0.024432122	-0.002041516	0
7.210345935	19.78073782	5000	0.093460574	0.065714531	0.024368661	-0.001930956	0
18.19282127	21.45754656	5000	0.147324322	0.070153645	0.023168421	-0.004109106	0
37.81746653	21.71667629	5000	0.211940879	0.070718819	0.021866723	-0.004494026	0
67.78988631	24.97040713	5000	0.284838273	0.081964876	0.02178371	-0.004838126	0
103.0972121	25.1427311	5000	0.352740215	0.083193419	0.021537934	-0.003800364	0
145.0948114	23.41482128	5000	0.418616789	0.077476032	0.016477944	0.000262616	0
193.0932432	24.50781255	5000	0.484065933	0.081385883	0.017923167	-0.002016348	0
247.0978419	26.32825477	5000	0.547096195	0.087116134	0.017522855	-0.003956197	0
308.245662	28.48533195	5000	0.610688642	0.093876932	0.011572872	-0.003402971	0

APPENDIX B. ROTOR PERFORMANCE TABLES

---

Table B.2: Rotor performance table for T-Motor Rotor at  $\alpha = 5^\circ$ .

q	Thrust_rho	RPM	Jinf	CT	CP	CM	Px_rho
0	8.064312251	3000	0	0.072609625	0.026517368	-0.006887378	0
8.802232693	8.74572109	3000	0.169892763	0.078588952	0.022625931	-0.004016875	0
19.9168254	8.765812907	3000	0.258318719	0.080514881	0.021807336	-0.003794061	0
38.24053111	9.568511966	3000	0.352630692	0.085305071	0.022592338	-0.004661069	0
67.45095005	9.972406448	3000	0.474504795	0.091231664	0.023950134	-0.006689884	0
103.7291856	11.2874386	3000	0.581166251	0.100629623	0.025359994	-0.008070923	0
144.9922543	11.89274504	3000	0.696941529	0.108944871	0.025745956	-0.009422482	0
193.1693478	12.80759767	3000	0.801082832	0.116160586	0.026412298	-0.012193503	0
247.0021871	13.1739854	3000	0.904680854	0.118773211	0.027443682	-0.015214208	0
306.6377332	13.45874611	3000	1.015598436	0.122880327	0.028531279	-0.018424691	0
372.4636575	13.93495861	3000	1.124436226	0.127823237	0.028544324	-0.023633838	0
0	14.86232757	4000	0	0.076473079	0.025990017	-0.001929986	0
8.236994432	13.55121288	4000	0.124408367	0.069777842	0.023740396	-0.004218264	0
19.85162503	14.69706505	4000	0.191433905	0.074381442	0.022487638	-0.004193003	0
38.63406869	15.60671099	4000	0.267980544	0.079535609	0.021308202	-0.003835431	0
67.32648058	16.48417164	4000	0.354192225	0.084180547	0.020981498	-0.004198927	0
102.8706436	17.561252	4000	0.438456723	0.089856216	0.019880811	-0.004783742	0
143.7702696	17.95459202	4000	0.519604865	0.092199734	0.02004175	-0.004892564	0
191.0103806	19.93002459	4000	0.601227905	0.102968315	0.023180022	-0.007434889	0
245.4357994	21.27910271	4000	0.68279066	0.109976999	0.024557448	-0.010717478	0
304.5928852	21.18094032	4000	0.762747454	0.10981112	0.025330282	-0.01173564	0
371.06128	18.53188719	4000	0.840989279	0.095449422	0.020382755	-0.013331176	0
0	24.80065243	5000	0	0.081733616	0.024754419	-0.001028291	0
7.790621367	23.67174599	5000	0.095752282	0.078013165	0.025623599	-0.000467342	0
19.67928593	24.75931164	5000	0.151285739	0.080626946	0.025019205	0.000192923	0
42.5637706	26.18480888	5000	0.223841325	0.086295274	0.024907008	-0.001061519	0
71.98931147	27.36264972	5000	0.291645021	0.090466255	0.024653912	-0.002263286	0
107.3595724	28.33522087	5000	0.356802289	0.093945048	0.024960224	-0.003317849	0
149.8170657	30.16539028	5000	0.417792266	0.098036458	0.02450667	-0.004503307	0
199.4118399	31.3249217	5000	0.4843168	0.102618517	0.024766854	-0.006353653	0
254.1339211	32.18988931	5000	0.549045658	0.106085759	0.024601401	-0.008014271	0
316.1514288	32.37849069	5000	0.614479847	0.107135432	0.025163702	-0.009295981	0

APPENDIX B. ROTOR PERFORMANCE TABLES

Table B.3: Rotor performance table for T-Motor Rotor at  $\alpha = 15^\circ$ .

q	Thrust_rho	RPM	Jinf	CT	CP	CM	Px_rho
0	7.848512156	3000	0	0.071610383	0.026484769	-0.001372249	0
9.228617439	7.893694313	3000	0.1748988	0.071498222	0.026468366	0.002752186	0
20.40373815	8.095794669	3000	0.261181895	0.074014315	0.027948487	0.005224193	0
38.88215631	8.446488516	3000	0.356604407	0.075550339	0.02867998	0.004316581	0
67.75210393	8.244427613	3000	0.47474274	0.074973042	0.028903063	0.003598188	0
103.7049153	8.185347706	3000	0.588147434	0.074584381	0.030212877	0.003099673	0
145.2043684	8.71148516	3000	0.696871125	0.07948421	0.030529527	0.002590287	0
192.7774539	8.426657823	3000	0.806196325	0.077348047	0.031121139	0.002529579	0
245.9244665	8.564840711	3000	0.903958972	0.077116293	0.031526243	-0.000573719	0
305.3398881	8.510265918	3000	1.007382356	0.076372233	0.03198915	-0.001878471	0
370.8505423	8.607652923	3000	1.111898004	0.077144322	0.032293525	-0.004321616	0
0	14.64246287	4000	0	0.073549834	0.026404037	-0.001106342	0
9.46763437	13.5512877	4000	0.133434486	0.070061126	0.024044251	-0.000333544	0
21.71736384	15.66941387	4000	0.200186781	0.079491423	0.024028902	-0.00089541	0
40.91221323	15.24132922	4000	0.276628782	0.078366288	0.024322444	-0.001464267	0
69.91269819	15.10010475	4000	0.360196519	0.076984885	0.02489524	-0.001656281	0
105.5052734	14.61932688	4000	0.443158877	0.074719489	0.025173715	-0.002007737	0
147.1741731	14.0836675	4000	0.523507773	0.07191007	0.025097863	-0.002630416	0
194.9566802	13.81455538	4000	0.603689482	0.070676691	0.025575393	-0.004080481	0
249.2685155	13.51789421	4000	0.682722205	0.068986926	0.025911158	-0.005393459	0
308.8841856	13.16936345	4000	0.761418012	0.067241708	0.026306106	-0.006112608	0
375.1546957	12.8763786	4000	0.839964542	0.065647659	0.025390629	-0.007945107	0
0	22.68533111	5000	0	0.074315749	0.025643012	-0.001133965	0
9.43009413	23.2897598	5000	0.027433238	0.075841454	0.023809161	-0.000642398	0
21.63670407	23.76058244	5000	0.041811821	0.078337268	0.025613595	-0.002466133	0
44.16705508	24.05302002	5000	0.059883158	0.079747468	0.024931565	-0.00241786	0
70.93847485	25.04534732	5000	0.075712078	0.082639175	0.024924287	-0.002959268	0
105.1618678	25.49071834	5000	0.091433057	0.082679647	0.025998232	-0.003610847	0
146.282182	25.92276519	5000	0.108335309	0.084752302	0.026015197	-0.004096966	0
194.3364564	25.38691179	5000	0.125334567	0.083498625	0.026669869	-0.005017478	0
247.6437982	25.63542192	5000	0.141859328	0.08455233	0.026654621	-0.005892607	0
307.7277117	26.20834792	5000	0.158485174	0.086545855	0.026917154	-0.007208047	0
373.7504903	23.21962284	5000	0.17494709	0.076584392	0.025735396	-0.011624524	0

APPENDIX B. ROTOR PERFORMANCE TABLES

---

Table B.4: Rotor performance table for T-Motor Rotor at  $\alpha = 30^\circ$ .

<b>q</b>	<b>Thrust_rho</b>	<b>RPM</b>	<b>Jinf</b>	<b>CT</b>	<b>CP</b>	<b>CM</b>	<b>Px_rho</b>
0	8.522799669	3000	0	0.077247647	0.027160748	-0.00196197	0
10.08217162	6.144169756	3000	0.180400366	0.054957052	0.025360011	0.003400664	0
21.09422348	5.3478641	3000	0.262995893	0.048567756	0.025326715	0.003683714	0
40.70600528	5.023204845	3000	0.362962698	0.045019442	0.025211113	0.003457865	0
70.89000015	4.317980955	3000	0.480735909	0.038955557	0.023709265	0.002769156	0
105.6463051	3.438803391	3000	0.591500005	0.03148056	0.021524517	0.002859947	0
147.7408169	2.808360404	3000	0.694172499	0.025285975	0.018605802	0.002453248	0
195.0620272	1.582459911	3000	0.801311731	0.014352377	0.015242598	0.002288723	0
248.4918726	0.276514228	3000	0.901256418	0.002478202	0.012787213	0.002154482	0
309.1720722	-0.968347536	3000	1.016766395	-0.008852939	0.007932426	0.002815768	0
374.396566	-2.187468139	3000	1.114369169	-0.019747763	0.004519268	0.002216306	0
0	14.54222283	4000	0	0.073956549	0.026474879	-0.001984538	0
10.88303834	11.50618743	4000	0.141604368	0.058226674	0.024605469	0.000825796	0
23.62710198	12.80355473	4000	0.208399327	0.06463168	0.024861752	0.000301548	0
41.85166835	12.25580801	4000	0.277206175	0.061805474	0.025101051	0.000150695	0
71.05019572	10.36963705	4000	0.366109932	0.053719365	0.024863424	-0.000120253	0
105.952132	9.006915276	4000	0.44561653	0.046333971	0.024177601	-0.000533362	0
146.9295864	8.396048472	4000	0.522322806	0.042699303	0.023240676	-0.000873812	0
194.8552083	7.023889574	4000	0.602033775	0.035720993	0.020807189	-0.000923653	0
248.1085502	5.362338265	4000	0.68614934	0.027751472	0.018254659	-0.001142026	0
307.9255036	3.833842919	4000	0.76269105	0.019692774	0.015277614	-0.000846209	0
373.3264161	2.685932403	4000	0.843652777	0.013865615	0.012174478	-0.000190927	0
0	24.45057927	5000	0	0.079274474	0.026135019	-0.001782557	0
12.29582634	22.0621725	5000	0.122216429	0.073000326	0.024280894	-0.001535802	0
25.57894872	22.34846544	5000	0.174982252	0.072921096	0.02504956	-0.001852178	0
45.81477127	21.74661149	5000	0.234286039	0.071070401	0.025574394	-0.002221932	0
72.18032068	21.28805651	5000	0.293770124	0.069433436	0.025360343	-0.002420265	0
106.4596905	19.02276369	5000	0.356271876	0.061847878	0.025429186	-0.002502402	0
147.2090633	17.87186155	5000	0.422240521	0.058969671	0.025575457	-0.002500062	0
195.1227391	17.45213487	5000	0.484028485	0.057012851	0.024589753	-0.002327311	0
248.7353983	14.36432244	5000	0.548442534	0.047150609	0.023613445	-0.002802502	0
308.1665626	12.1659312	5000	0.606902627	0.039382259	0.021062462	-0.00366951	0
373.0640095	8.406833753	5000	0.68216704	0.028268299	0.019511269	-0.003663859	0

## Appendix C

# MATLAB Scripts

```
1 % -----
2 % MULTI-ROTOR VEHICLE PERFORMANCE MODEL: MAIN SCRIPT_v3
3 % BY: ISSI GEORGE
4 % Most Recent Revision: June 21st 2016
5 % -----
6 % Multirotor prediction model to calculate overall flyer power required ,
   thrust
7 % and moment using blade element momentum theory and wake interaction model
8 % V2 Update: plots data per rotor , read geometry from .txt file , calculates
   induced velocity for each
9 % rotor
10 % V3 Update: proper angle convention , Pforce , interaction , Vinf+Vint inflow ,
11 % acceleration terms
12 % blade: blade geometry structure .
13 % oper: operating conditions and wake parameters structure .
14 % flow: flowfield and operating conditions .
15 % perf: performance structure output.1
16 % fgeom: flyer geometry
```

```
17     % rotor: holds data pertaining to each rotor on flyer
18 close all;
19
20 clc; clear;
21
22 cd('C:\Users\Issi\Documents\Academia\Ryerson\Performance Codes\Thesis\
    Performance Tables\T-Rotor Experimental');
23
24
25 %% Flow field Information
26 flow.temp          =    298.15;                % [ Kelvin]
27 flow.altitude      =    0.3048*1000;           % [m]
28 flow.mu            =    1.846*10−5;          % Dynamic
    viscosity [kg/ms]
29 rho_o              =    1.225;                % density at
    sea level
30 flow.rho            =    density_calc(flow, rho_o); % Calculate
    density at altitude
31 flow.inflow_angle  =    0;
32 %% Import Flyer Geometry and Set Coordinates
33 folder = 'C:\Users\Issi\Documents\Academia\Ryerson\Performance Codes\Thesis
    \Geometry';
34
35 fgeom = geomimport(sprintf('%s%s', folder, '\trotor.txt'));
36
37 fgeom.armang=360/fgeom.norotors;                %angle between
    arms
38 fgeom.Swl=2*fgeom.legr*fgeom.legl;              %wetted area of
    leg, m^2
```

```

39 fgeom.Swa=2*fgeom.armr*fgeom.arml;           %wetted area of
        arm, m^2
40 fgeom.Swb=pi*(fgeom.bodyr)^2;               %wetted area of
        body
41 fgeom.payloadr = 0.045;           %payload radius
42 fgeom.payloadl = 0.165;           %payload length
43 fgeom.Swp=pi*(fgeom.payloadr)^2;
44 R = fgeom.diameter/2;
45 cd = 0.08;
46 cl=0.3;
47 sigma = 2*(0.03)/(pi*R);
48 area = pi*(R)^2;
49 a=1;
50
51 %set coordinates
52 [ fgeom ] = coord_setup(fgeom,flow);
53
54 %% Initial Values
55
56 %initialize power values
57 Ppar = 0;
58 Pind = 0;
59 Ppro = 0;
60 Prot = 0;
61 T2 = 0;
62 alpha2 = 0;
63 %velocity loop counter
64 j=1;
65 count=1;

```



```
66 %% Load Performance Data for Rotor
67
68 D = dir([ 'C:\Users\Issi\Documents\Academia\Ryerson\Performance Codes\Thesis
           \Performance Tables\T-Rotor Experimental', '*.*mat' ]);
69 Num = length(D(not([D.isdir])));
70 for k=1:Num
71     file{k} = D(k).name;
72 end
73
74 file=char(file);
75 for k=1:Num
76     filename(k,:)=strsplit(file(k,:),{' ','_'});
77 end
78
79 angle_list=filename(:,1);
80 angle_list=str2double(angle_list);
81 count=1;
82 %% BEMT Performance Prediction Function w/ Wake Interaction Model
83 %for V_TAS=10:10
84 for V_TAS=0:25
85     %% Calculate True Airspeed
86
87     flow.V=V_TAS;
88     q = 0.5*flow.rho*V_TAS^2;           %dynamic pressure
89
90 %% Parasite Drag Calculation
91 [Ppar,Dpar_tot] = par_power(flow,fgeom);
92 perf.Dpar_tot = Dpar_tot;
93 perf.Ppar = Ppar/4;
```

```
94     alpha1=flow.inflow_angle;
95     % alpha1=10;
96 %% Performance
97     %Initial Condition Statement
98     if V_TAS==0
99         flow.inflow_angle=atand(perf.Dpar_tot/fgeom.weight); %alpha 1
100        %starting thrust
101        perf.Treq=sqrt(fgeom.weight^2 + Dpar_tot^2)*1/4; %starting thrust
102        T1=perf.Treq;
103        perf.Treq_rho=perf.Treq/flow.rho;
104
105        % Aerodynamic Performance Using Lookup Table
106        [ rpm,CP,CM,Px,Jint ] = tablelookup(file,flow, angle_list, q, perf);
107        alpha2=0;
108        alpha1=0;
109    else
110        alpha1=alpha1+0.1;
111        flow.inflow_angle=alpha1;
112        T1=sqrt(fgeom.weight^2 + Dpar_tot^2)*1/4;
113        perf.Treq_rho=T1/flow.rho;
114
115        while alpha1~=alpha2
116            flow.inflow_angle=alpha1;
117            % Aerodynamic Performance Using Lookup Table
118            [ rpm,CP,CM,Px,Jint ] = tablelookup(file,flow, angle_list, q, perf
119                );
119            Px=0;
120            %find alpha2 and T2
121            alpha2=asind((Dpar_tot/4*T1+Px*fgeom.weight/4)/(T1^2+Px^2));
```

```
122         T2=(Px*cosd(alpha2)+Dpar_tot/4)/sind(alpha2);
123
124         %check convergence
125         if (alpha1-alpha2)*(alpha1-alpha2)<0.001
126             alpha1=alpha2;
127             perf.Treq=T2;
128             flow.inflow_angle=alpha2;
129             alpha2=0;
130             break;
131         else
132             T1=T2;
133             perf.Treq=T2;
134             perf.Treq_rho=T2/flow.rho;
135             clear T2
136             alpha1=alpha2;
137             alpha2=0;
138         end
139     end
140 end
141 clear count;
142
143 %     %starting thrust
144 %     perf.Treq=; %starting thrust
145 %     perf.Treq_rho=perf.Treq/flow.rho;
146 %
147
148 %% Wake Interference Model
149 for roi=1:fgeom.norotors
150     [vix,viy,viz,vi_self,X] = WIM_v10(flow,rpm,perf,fgeom,R,roi);
```

```
151     rotor(j,roi).vix=vix;
152     rotor(j,roi).viy=viy;
153     rotor(j,roi).viz=viz;
154     rotor(j,roi).vi_self=vi_self;
155
156     %     for count=1:fgeom.norotors
157     %         if count==roi
158     %             continue;
159     %         else
160     %             count2=1;
161     %             rotor(j,roi).w_int(1,count2)=wi(count,3);
162     %             count2=count2+1;
163     %         end
164     %     end
165     %     clear count2
166
167     %calculate self induced and interference power
168     rotor(j,roi).Pind_int=1.15*perf.Treq*rotor(j,roi).viz;
169     rotor(j,roi).Pind_self=1.15*perf.Treq*rotor(j,roi).vi_self;
170
171 end
172
173 %% Calculate Performance Values
174 J=V_TAS/(rpm/60*fgeom.diameter);
175
176
177 Prot=(CP*flow.rho*(rpm/60)^3*fgeom.diameter^5);
178 Ppro=(flow.rho*area*((rpm/60)*2*pi*(R))^3*sigma*cd/8*(1+3*J^2))/4;
179 % Ppro_test= Prot - Pind_self;
```

```

180
181     %With Interference
182     for roi=1:fgeom.norotors
183         rotor(j,roi).Ptot=Prot+perf.Ppar-rotor(j,roi).Pind_int;
184     end
185
186     %Without Interference
187     %    results(j,1).Ptot=Prot;%+perf.Ppar;
188
189     %% Store Values
190     %save values
191     results(j,1).Dpar=perf.Dpar_tot;
192     results(j,1).Vrange=V_TAS;
193     results(j,1).Treq=perf.Treq;
194     results(j,1).Alpha=alpha1;
195     results(j,1).rpm_h=rpm;
196     results(j,1).J = J;
197     results(j,1).Jint = Jint;
198     results(j,1).T=perf.Treq;
199     results(j,1).q=q;
200     results(j,1).Px=Px;
201     results(j,1).CM=CM;
202     results(j,1).X=X;
203
204     %results(j,1).Ptot=rotor(j,:).Ptot;           %total power holder
205     results(j,1).Ppr=Ppro;           %profile power holder
206     results(j,1).Pp=perf.Ppar;       %parasitic power holder
207     results(j,1).Pav=perf.Treq*V_TAS;
208     results(j,1).RotP=Prot;           %rotor power holder

```

```

209
210 %
211     j=j+1;
212
213 end

1 function [ fgeom ] = coord_setup( fgeom, flow )
2 % Coordinate setup for flyer
3 po=ones(fgeom.norotors,3);
4 ab_radius=fgeom.arml+fgeom.bodyr; %total distance from center
    of rotor to center of body, in m
5
6 %set rotor midpoints,po
7 check = 'diamond';
8 tf = strcmp(fgeom.orientation,check);
9
10 %diamond
11 if tf == 1
12     for i=1:fgeom.norotors
13         po(i,1:3) =[ab_radius*cosd((i-1)*fgeom.armang),ab_radius*sind((i-1)
            *fgeom.armang),0]; %rotors labelled 1-4 ccw
14     end
15 %square
16 else
17
18     po(1,1:3) =[ab_radius*(1/sqrt(2)),ab_radius*(1/sqrt(2)),0]; %rotors
        labelled 1-4 ccw
19     po(2,1:3) =[ab_radius*(1/sqrt(2)),-ab_radius*(1/sqrt(2)),0];
20     po(3,1:3) =[-ab_radius*(1/sqrt(2)),-ab_radius*(1/sqrt(2)),0];
21     po(4,1:3) =[-ab_radius*(1/sqrt(2)),ab_radius*(1/sqrt(2)),0];

```

```

22 end
23
24 % figure;
25 % scatter3(po(:,1),po(:,2),po(:,3));
26 % xlabel('X-axis');
27 % ylabel('Y-axis');
28 % zlabel('Z-axis');
29
30 fgeom.po=po;
31
32 end

1 %Function to calculate parasitic drag and power of flyer body
2
3 function [Ppar,Dpar_tot,Cd_body,Re] = par_power(flow,fgeom)
4 %calculate Sw for each component
5 %fgeom.Swb=fgeom.bodyl*cosd(flow.inflow_angle)*2*fgeom.bodyr;
6
7
8 %%Calculations for Parasite Drag
9 [Dpar_arm,Cd_arm,ReA] = CylinderDrag(flow,fgeom.armr,fgeom.Swa); %arm
10 [Dpar_leg,Cd_leg,ReL] = CylinderDrag(flow,fgeom.legr,fgeom.Swl); %arm
11 [Dpar_body,Cd_body,ReB] = SphereDrag(flow,fgeom.bodyr,fgeom.Swb);%body
12 [Dpar_payload,Cd_body,ReP] = SphereDrag(flow,fgeom.payloadr,fgeom.Swp);%
    body
13
14 Dpar_tot=(4*Dpar_leg+4*Dpar_arm+Dpar_body+Dpar_payload);
15 Ppar = Dpar_tot*flow.V; %parasitic power due to fuselage
    etc
16 end

```

```
1 % Modified Script for Cd determination for cylinder and sphere (Unmodified)
2
3 function [Dpar,Cd_cyl,Re] = CylinderDrag(flow,dx,Sw)
4
5 [Rex_in, Cdy_in, Rex_out, Cdy_out] = generateReynoldsCurveCylinder();
6 dx=2*dx;
7 Re=flow.V*flow.rho*dx/flow.mu;
8
9 Cd_cyl = interp1(Rex_out,Cdy_out,Re, 'pchip', 'extrap'); %interpolate Cd
    at Re
10
11 Dpar=0.5*Cd_cyl*flow.rho*flow.V^2*Sw;
12 end
13
14 function [Rex_in, Cdy_in, Rex_out, Cdy_out] = generateReynoldsCurveCylinder
    ()
15     Rex_in=[0.1 0.3 1 3 10 40 100 300 1000 10000 100000 300000 480000
        1000000];
16     % Cdy_in=[60 25 10 4 3 2 1.8 1.5 1 1.3 1.4 1 0.28 0.45];
17     Cdy_in=[60 25 10 4 3 2 1.8 1 1 1 1 1 0.28 0.45];
18     [Rex_out, Cdy_out] = generateReynoldsCurve(Rex_in, Cdy_in, 100);
19 end
20
21 function [Rex_out, Cdy_out] = generateReynoldsCurve(Rex, Cdy, points)
22     decade_i = ceil(log10(Rex(1))) - 1;
23     decade_f = ceil(log10(Rex(end)));
24     Rex_out = logspace(decade_i, decade_f, points);
25     Cdy_out = interp1(Rex, Cdy, Rex_out, 'pchip', 'extrap');
26 end
```



```

1  function [Dpar,Cd_sph,Re] = SphereDrag(flow,dx,Sw) %test function
2
3      [Rex_in, Cdy_in, Rex_out, Cdy_out] = generateReynoldsCurveSphere();
4      dx=2*dx;
5      Re=flow.V*flow.rho*dx/flow.mu;
6
7
8      Cd_sph = interp1(Rex_out,Cdy_out,Re, 'pchip', 'extrap'); %interpolate
          Cd at Re
9
10     % Cd_sph=0.41;
11     if flow.V==0
12         Cd_sph=0;
13     end
14     Dpar=0.5*Cd_sph*flow.rho*flow.V^2*Sw;
15 end
16
17 function [Rex_in, Cdy_in, Rex_out, Cdy_out] = generateReynoldsCurveSphere(
    Rex, Cdy)
18     Rex_in=[0.2 0.4 1 2 6 10 30 10^2 400 2000 10000 100000 400000 1000000
          6000000];
19     % Cdy_in=[10^2 60 25 15 6 4 2 1 0.6 0.4 0.4 0.5 0.08 0.1 0.2];
20     Cdy_in=[10^2 60 25 15 6 4 2 1 0.6 0.4 0.4 0.4 0.27 0.16 0.2];
21     [Rex_out, Cdy_out] = generateReynoldsCurve(Rex_in, Cdy_in, 10000);
22 end
23
24 function [Rex_out, Cdy_out] = generateReynoldsCurve(Rex, Cdy, points)
25     decade_i = ceil(log10(Rex(1))) - 1;
26     decade_f = ceil(log10(Rex(end)));

```

```
27     Rex_out = logspace(decade_i, decade_f, points);
28     %Cdy_out = spline(log(Rex), Cdy, log(Rex_out)); % kinda messy for
        sphere...
29
30     Cdy_out = interp1(Rex, Cdy, Rex_out, 'pchip', 'extrap');
31 end

1 function [ low_ang, high_ang, ArrAng ] = tablebound( flow, angle_list)
2
3
4 end

1 function [rpm, CP] = rpm_lookup_v3(Vmps, perf, angle_table)
2 %Performance Lookup Table
3 % Using RAALF Experimental Data and a series of interpolation...
4 % finds the RPM, CT, and CP for a required Thrust and known dynamic
5 % pressure
6 count=1;
7 angle_table = cell2struct(angle_table, {'angle_table'});
8 %% Interpolation A: Find the lower RPM curve
9 %lowest curve would be in static data
10 %find RPM group
11 for i=1:size(angle_table, 'angle_table.RPM')
12     if angle_table('angle_table.VT')(i)<Vmps && angle_table('angle_table.Thrust')(
        i)<perf.Treq || Vmps==0
13         T_h(count)= angle_table('angle_table.Thrust')(i);
14         indices(count)=i;
15         count=count+1;
16     end
17 end
18 %find RPM curve or range
```

```
19 [A,B]=min(abs(T_h-perf.Treq));
20 rpm_A=angle_table.angle_table.RPM(indices(B)); %%RPM curve
21
22 T_Ar = angle_table.angle_table{angle_table.angle_table.RPM == rpm_A, {'
    Thrust'}}};
23 VT_A = angle_table.angle_table{angle_table.angle_table.RPM == rpm_A, {'VT'
    }};
24 CP_Ar = angle_table.angle_table{angle_table.angle_table.RPM == rpm_A, {'CP'
    }};
25
26 T_A=interp1(VT_A,T_Ar,Vmps);
27 CP_A=interp1(VT_A,CP_Ar,Vmps);
28 %% Interpolation B: Find the upper RPM curve
29 %Search through RPMs to find at which RPM are both the VT and the Thrust
30 %higher than the required thrust and velocity
31
32 %reset counter
33 count=1;
34 for i=1:size(angle_table.angle_table.RPM)
35     if Vmps<angle_table.angle_table.VT(i) && perf.Treq<angle_table.
        angle_table.Thrust(i) && angle_table.angle_table.RPM(i)~=rpm_A
36         T_h(count)= angle_table.angle_table.Thrust(i);
37         indices(count)=i;
38         count=count+1;
39     end
40 end
41
42 %find RPM curve or range
43 count=1;
```

```

44 [A,B]=min(abs(T_h-perf.Treq));
45 rpm_B=angle_table.angle_table.RPM(indices(B)); %RPM curve
46
47 T_Br = angle_table.angle_table{angle_table.angle_table.RPM == rpm_B, {'
    Thrust'}}};
48 VT_B = angle_table.angle_table{angle_table.angle_table.RPM == rpm_B, {'VT'
    }};
49 CP_Br = angle_table.angle_table{angle_table.angle_table.RPM == rpm_B, {'CP'
    }};
50
51 %find upper velocity
52 T_B=interp1(VT_B,T_Br,Vmps);
53 CP_B=interp1(VT_B,CP_Br,Vmps);
54
55 %% Value Check
56 %stop interpolation if bounds are equal
57 if rpm_B==rpm_A
58     rpm=rpm_B;
59     CP=CP_B;
60     return;
61 end
62
63 %% Interpolation C: Interpolate between curves to locate RPM
64 ArrT=[T_A,T_B];
65 ArrR=[rpm_A,rpm_B];
66 ArrCP=[CP_A,CP_B];
67 rpm=interp1(ArrT,ArrR,perf.Treq);
68 CP=interp1(ArrT,ArrCP,perf.Treq);
69

```

```

70
71 end
72
73
74 %reset counter
75 % count=1;
76 % for i=1:size(ASP130_15.RPM)
77 %     if Vmps>ASP130_15.VT(i) && perf.Treq>ASP130_15.Thrust(i)
78 %         T_hl(count)= ASP130_15.Thrust(i);
79 %         indices(count)=i;
80 %         count=count+1;
81 %     end
82 % end
83 %
84 %
85 % %find RPM curve or range
86 % count=1;
87 % [A,B]=max(abs(T_hl-perf.Treq));
88 % rpm_A=ASP130_15.RPM(indices(B)); %RPM curve
89 %
90 % for i=1:size(ASP130_15.RPM)
91 %     if ASP130_15.RPM(i)==rpm_A
92 %         range_A(count)=i;
93 %         count=count+1;
94 %         i=i+1;
95 %     end
96 % end
97 %
98 % %find upper velocity

```

```

99 % V_A=interp1(ASP130_15.Thrust(range_A),ASP130_15.VT(range_A),perf.Treq);
100 % CP_A=interp1(ASP130_15.Thrust(range_A),ASP130_15.CP(range_A),perf.Treq);
101
102
103 % for i=1:size(ASP130_15.RPM)
104 %     if ASP130_15.RPM(i)==rpm_B && Vmps<ASP130_15.VT(i) && perf.Treq<
        ASP130_15.Thrust(i)
105 %         range_B(count)=i;
106 %         count=count+1;
107 %         i=i+1;
108 %     end
109 % end
110
111 % %find lower boundary
112 % for i=1:size(range_B,2)
113 %     Range_B(i)=range_B(i)-1;
114 % end
115 % Range_B((size(range_B,2))+1)=range_B(size(range_B,2));

1 function [rpm,CP,CM,Px_rho, Jint] = rpm_lookup_fix(q,perf,angle_table)
2 j=1;
3 h=1;
4
5 for k=2:length(angle_table.angle_table.RPM)
6     if angle_table.angle_table.q(j)==q
7         low(1,h)=j;
8         up(1,h)=k;
9         h=h+1;
10    else
11        %lower and upper indices

```

```
12         if (angle_table.angle_table.q(j)< q && q < angle_table.angle_table.  
            q(k)) %correction , less than or equal  
13             low(1,h)=j;  
14             up(1,h)=k;  
15             h=h+1;  
16         end  
17     end  
18     j=j+1;  
19 end  
20  
21 %determine which curves to be interpolated on. The three curves with the  
    smallest  
22 % difference between the required T/rho and the tabled T/rho will be used  
23 %If there are 3 then the interpolation scheme  
24 %can move on without the need to sort for which curves are closer  
25 for i=1:length(low)  
26     A(i)=abs(angle_table.angle_table.Thrust_rho(low(i))-perf.Treq_rho);  
27 end  
28  
29 [B,I]=sort(A,'ascend');  
30 %pick the first diverse three which are going to be the three closest.  
31 %The program can exploit by choosing the upper bound as an indice if q is  
32 %listed on the table. Make sure the rpms will be diverse by making sure  
33 %this does not happen  
34 I=I(1:3);  
35  
36  
37 rpm_LL=angle_table.angle_table.RPM(low(I(1)));  
38 q_LL=[angle_table.angle_table.q(low(I(1))),angle_table.angle_table.q(up(I
```

```

(1)))];
39 T_rho_LL =[angle_table.angle_table.Thrust_rho(low(I(1))),angle_table.
    angle_table.Thrust_rho(up(I(1)))];
40 CM_LL =[angle_table.angle_table.CM(low(I(1))),angle_table.angle_table.CM(up
    (I(1)))];
41 CP_LL =[angle_table.angle_table.CP(low(I(1))),angle_table.angle_table.CP(up
    (I(1)))];
42 Px_LL =[angle_table.angle_table.Px_rho(low(I(1))),angle_table.angle_table.
    Px_rho(up(I(1)))];
43 J_LL =[angle_table.angle_table.Jinf(low(I(1))),angle_table.angle_table.
    Jinf(up(I(1)))];
44
45 T_LL=interp1(q_LL,T_rho_LL,q);
46 CP_LL=interp1(q_LL,CP_LL ,q);
47 CM_LL=interp1(q_LL,CM_LL,q);
48 Px_LL=interp1(q_LL,Px_LL,q);
49 J_LL=interp1(q_LL,J_LL,q);
50
51
52 %interpolate lower rpm
53 %create mini arrays
54 rpm_L=angle_table.angle_table.RPM(low(I(2)));
55 q_L=[angle_table.angle_table.q(low(I(2))),angle_table.angle_table.q(up(I(2)
    ))];
56 T_rho_L =[angle_table.angle_table.Thrust_rho(low(I(2))),angle_table.
    angle_table.Thrust_rho(up(I(2)))];
57 CM_L =[angle_table.angle_table.CM(low(I(2))),angle_table.angle_table.CM(up(
    I(2)))];
58 CP_L =[angle_table.angle_table.CP(low(I(2))),angle_table.angle_table.CP(up(

```



```
        I(2)))];  
59 Px_L =[angle_table.angle_table.Px_rho(low(I(2))),angle_table.angle_table.  
        Px_rho(up(I(2)))];  
60 J_L =[angle_table.angle_table.Jinf(low(I(2))),angle_table.angle_table.Jinf  
        (up(I(2)))];  
61  
62  
63 T_L=interp1(q_L,T_rho_L,q);  
64 CP_L=interp1(q_L,CP_L ,q);  
65 CM_L=interp1(q_L,CM_L,q);  
66 Px_L=interp1(q_L,Px_L,q);  
67 J_L=interp1(q_LL,J_L,q);  
68  
69 %interpolate upper  
70 rpm_U=angle_table.angle_table.RPM(low(I(3)));  
71 q_U =[angle_table.angle_table.q(low(I(3))),angle_table.angle_table.q(up(I  
        (3)))];  
72 T_rho_U =[angle_table.angle_table.Thrust_rho(low(I(3))),angle_table.  
        angle_table.Thrust_rho(up(I(3)))];  
73 CM_U =[angle_table.angle_table.CM(low(I(3))),angle_table.angle_table.CM(up(  
        I(3)))];  
74 CP_U =[angle_table.angle_table.CP(low(I(3))),angle_table.angle_table.CP(up(  
        I(3)))];  
75 Px_U =[angle_table.angle_table.Px_rho(low(I(3))),angle_table.angle_table.  
        Px_rho(up(I(3)))];  
76 J_U =[angle_table.angle_table.Jinf(low(I(3))),angle_table.angle_table.Jinf  
        (up(I(3)))];  
77  
78
```

```
79 T_U=interp1(q_U,T_rho_U,q);
80 CP_U=interp1(q_U,CP_U ,q);
81 CM_U=interp1(q_U,CM_U,q);
82 Px_U=interp1(q_U,Px_U,q);
83 J_U=interp1(q_LL,J_U,q);
84
85 %solution
86 %make array to interpolate
87 ArrT=[T_LL,T_L,T_U];
88 ArrR=[rpm_LL,rpm_L,rpm_U];
89 ArrCP=[CP_LL,CP_L,CP_U];
90 ArrCM=[CM_LL,CM_L,CM_U];
91 ArrPx=[Px_LL,Px_L,Px_U];
92 ArrJ=[J_LL,J_L,J_U];
93
94 %check to see if all elements of ArrT are distinct , MUST be for polyfit
95 % for i=1:2
96 %     for j=2:3
97 %         if j==i
98 %             continue;
99 %         else if ArrT(i)==ArrT(j)
100 %             ArrT(i)=ArrT(i)+0.1;
101 %         end
102 %     end
103 % end
104 % end
105
106 coeff=polyfit(ArrT,ArrR,2);
107 rpm=coeff(1)*perf.Treq_rho^2+coeff(2)*perf.Treq_rho+coeff(3);
```

```

108 %rpm=interp1 (ArrT,ArrR,perf.Treq_rho);
109
110 %find Cp through cubic interpolation since n and Cp are cubically related
111 CP=interp1 (ArrR,ArrCP,rpm,'pchip','extrap');
112 CM=interp1 (ArrR,ArrCM,rpm,'linear','extrap');
113 Px_rho=interp1 (ArrR,ArrPx,rpm,'linear','extrap');
114 Jint=interp1 (ArrR,ArrJ,rpm,'linear','extrap');
115
116
117
118 end

1 %function [ Vtin,Wtin] = WM(blade,flow,oper,gamma,alpha)
2 %Wake Interference Model V.10 Uses updated vortring and vortsegment
3 %Calculate the induced velocity caused by rotor/wake/body interference in a
4 %rotorcraft/UAV at the center of the rotor. To be used with TABLE LOOKUP
5 %thrust(N), azimuth(rad), V(m/s),
6 %updated structures, geometry
7
8 function [ vix,viy,viz,vi_self,X] = WIM_v10(flow,omega,perf,fgeom,R,roi)
9 %% General Parameters
10 % hold ;
11 %axis equal;
12 nose=100; %number of segments for VortRing calculation
13 nowakes=400; %number of wakes for VortRing calculation
14
15
16 r=R;
17 alpha=deg2rad(flow.inflow_angle); %angle of attack
18 rotors=fgeom.norotors; %number of rotors

```

```
19  Blades=2; %number of blades
20  %thrust for each rotor
21  po=fgeom.po;
22
23  for i=1:rotors
24      thrust(:,i)=perf.Treq;
25      R(:,i)=r;
26      A(:,i)=pi.*R(:,i)^2;
27      n(:,i)=omega/60; %rev/s
28      circ(:,i)=thrust(:,i)/(flow.rho*pi()*n(:,i).*(R(:,i)).^2*Blades);
29  end
30
31
32  %% Calculate induced Velocity
33
34  P=[po(roi,1),po(roi,2),po(roi,3)]; %point of interest
35
36
37  %figure;
38  for m=1:rotors %m denotes which rotor is influencing
39      % if m==roi %CHANGE FOR DIFFERENT ROTOR
40      % wi(m,1:3)=0;
41      % elseif m==1
42      % wi(m,1:3)=0;
43      % elseif m==3
44      % wi(m,1:3)=0;
45      % else
46      mu=(flow.V*cos(alpha));
47      v=0.5*(-flow.V*sin(alpha)+sqrt((flow.V*sin(alpha))^2+2*thrust(m)/(flow.
```

```
        rho*R(m)^2*pi));
48     %v=sqrt(thrust(m)/(2*flow.rho*A(m)));
49     lambda=flow.V*sin(alpha);
50     X=atan2(mu,lambda+v); %calculate skew angle of wakes for influencing
        rotor
51     %X=atan((-flow.V*cos(alpha))/(flow.V*sin(alpha)-sqrt(thrust(m)/(2*flow.
        rho*R(m)^2*pi))));
52     X=radtodeg(X);%+180;
53
54     %     if X>90
55     %         X= 1/(atan2((-flow.V*cos(alpha)),(flow.V*sin(alpha)-0.5*(-flow.V*
        sin(alpha)+sqrt((flow.V*sin(alpha))^2+(2*thrust(m)/(flow.rho*A(m))))))
        ));
56     %     end
57
58     wih=0;
59     zoff=0;
60     for k=0:1:nowakes
61         zoff=k*1/n(m)*0.5*(-flow.V*sin(alpha)+sqrt((flow.V*sin(alpha))
            ^2+(2*thrust(m)/(flow.rho*A(m)))))*(1/Blades);
62         wih=wih+VortRing(noseg,zoff,X,R(m),circ(m),P,po(m,1:3));
63     end
64     wi(m,1:3)=wih; %induced velocity from each rotor
65
66 end
67 %add induced velocities and subtract rotor of interest to get interference
68 vi_int=(sum(wi)-wi(roi,1:3));
69 %self induced velocity
70 vi_self=wi(roi,3);
```

```
71 %
72 vix=vi_int(1,1);
73 viy=vi_int(1,2);
74 viz=vi_int(1,3);
75
76 %vi_self=wi(1,3);
77
78 % vix=wi(1,3);
79 % viy=wi(2,3);
80 % viz=wi(3,3);
81
82
83
84
85 %clear wi;
86
87
88 %%
89 %plot quiverplot for specific rotor
90 % poi_x=vi_int(1,1);
91 % poi_y=vi_int(1,2);
92 % z2=vi_int(1,3);
93 % plot3(poi_x,poi_y,z2);
94 % quiver3(P(1),P(2),P(3),poi_x,poi_y,z2);
95
96 % xlabel('x-axis');
97 % ylabel('y-axis');
98 % zlabel('z-axis');
99 end
```

```

1  function [wi,xo] = VortRing (i,z,X,R,gamma,P,po)
2  %September 8th: added for statement to plot rings only when z=0, only rotor
3  %November 23rd: vectorized vortsegment and vortring, cut 20 seconds
4  %
5
6  ang=360/i; %angle between vortex segment
7
8  %set midpoint based on skew angle
9  %xo = xofn(X,po,z);
10  xo=[po(1)+z*tand(X), po(2), po(3)-z];
11
12  %set coordinates
13  k=0; %counter for coordinates loop
14  for j=1:i
15      s(j,:)=[xo(1)+R*cosd(k*ang),xo(2)+R*sind(k*ang),xo(3)];
16      k=k+1;
17  end
18
19
20  %    hold on;
21  %    %scatter3(xo(:,1),xo(:,2),xo(:,3));
22  %    plot3(s(:,1),s(:,2),s(:,3));
23  %    scatter3(P(1),P(2),P(3));
24
25
26
27  wi=sum(VortSegment(P,s,gamma)); %call vortex segment code for first i-1
    segments
28

```

```

29
30 % wi=0;
31 % for j=1:1:i
32 %     if j==i
33 %         wi=wi+VortSegment(P,s(i,:),s(1,:),gamma);
34 %         break
35 %     end
36 %     wi=wi+VortSegment(P,s(j,:),s(j+1,:),gamma); %call vortex segment code
           for first i-1 segments
37 % end
38
39
40 end

1 %Vortex Segment ———— calculates the induced velocity on a point caused
           by vortex segment filament
2 %September 8th: added if statement for d distance to avoid singularities
3
4 function [q12]= VortSegment(P,s,gamma)
5 A=length(s(:,1));
6
7 s1=s;
8
9 s2=[s([2:end],:);s(1,:)];
10 Pe=ones(A,1)*P;
11
12
13 r1=Pe-s1;%Point P - vortex segment starting point gives length of r1
14 r2=Pe-s2;%Point P - vortex segment ending point gives length of r2
15

```



```

16  r0=r1-r2;
17
18  product = cross(r1,r2,2);
19  product_norm = (product(:,1).^2+product(:,2).^2+product(:,3).^2).^0.5;
20
21  norm_r0 = (r0(:,1).^2+r0(:,2).^2+r0(:,3).^2).^0.5;
22  norm_r1 = (r1(:,1).^2+r1(:,2).^2+r1(:,3).^2).^0.5;
23  norm_r2 = (r2(:,1).^2+r2(:,2).^2+r2(:,3).^2).^0.5;
24
25  d = product_norm./norm_r0;
26
27
28  cosB1 = dot(r0,r1,2)./(norm_r0.*norm_r1); %where B1 is angle between r1 and
      line segment
29  cosB2 = dot(r0,r2,2)./(norm_r0.*norm_r2); %where B2 is angle between r2 and
      line segment
30
31  qtheta = -gamma./(4*pi.*d).*(cosB1-cosB2); %velocity induced in P by vortex
      segment
32
33  q12=product.*[qtheta qtheta qtheta] ./ [ product_norm product_norm
      product_norm]; %velocity in in P by vortex segment directionalized
34
35  d2 = d.*d;
36  idx = (d2>=10^(-10));
37
38  q12 = q12.*[idx idx idx];
39  %q12(isnan(q12)) = 0;
40  % if (d.*d)<=10^(-10)

```

```
41 %      q12=[0,0,0];  
42 % else  
43  
44  
45 end
```

# Bibliography

- [1] Aeryon, “Aeryon SkyRanger the benchmark for VTOL sUAS,” <https://www.aeryon.com/aeryon-skyranger>, 2016, [Online; retrieved 16-September-2016].
- [2] Calvo, K., “So You Want to Fly Drones?” <http://voices.nationalgeographic.com/2013/11/30/so-you-want-to-fly-drones/>, 2013, [Online; retrieved 16-September-2016].
- [3] Gupte, S., Mohandas, P. I. T., and Conrad, J. M., “A survey of quadrotor unmanned aerial vehicles,” *2012 Proceedings of IEEE*, Orland, Fl, March 2012, pp. 1–6.
- [4] Bouabdallah, S., *Design and control of quadrotors with application to autonomous flying*, Master’s thesis, Ecole Technique Federale de Lausanne, 2007.
- [5] Logan, M., Chu, J., Motter, M. A., Carter, D. L., Ol, M., and Zeune, C., “Small UAV Research and Evolution in Long Endurance Electric Powered Vehicles,” *AIAA Infotech Aerospace 2007 Conference and Exhibit*, 2007.
- [6] Logan, M., Chu, J., Motter, M. A., Carter, D. L., Ol, M., and Zeune, C., “Toward an Accurate Physics-Based UAV Thruster Model,” *IEEE/ASME Transactions on Mechatronics*, 2013.
- [7] Russell, C., Jung, J., Willinuk, G., and Glasner, B., “Wind Tunnel and Hover Performance Test Results for Multicopter UAS Vehicles,” *American Helicopter Society 72nd Annual Forum*, West Palm Beach Fl., 2016.
- [8] Merchant, M. P. and Miller, L. S., “Propeller Performance Measurement for Low Reynolds Number UAV Applications,” *AIAA Aerospace Sciences Meeting and Exhibit*, Vol. 2006-1127, Reno, NV, Jan 2006.

## BIBLIOGRAPHY

---

- [9] Deters, R. W., Ananda, G. K., and Selig, M. S., “Reynolds Number Effects on the Performance of Small-Scale Propellers,” *32nd AIAA Applied Aerodynamics Conference*, Vol. 2014-2151, Atlanta, GA, June 2014.
- [10] Thomas, F., *Fundamentals of Sailplane Design*, College Park Press, College Park Maryland, USA, 1999.
- [11] Young, C., “The Prediction of Helicopter Rotor Hover Performance using a Prescribed Wake Analysis,” Tech. Rep. No. 1341, Aeronautical Research Council, 1974.
- [12] Powers, C., Mellinger, D., Kushleyev, A., Kothmann, B., and Kumar, V., “Influence of Aerodynamics and Proximity Effects in Quadrotor Flight,” *The 13th International Symposium on Experimental Robotics*, Vol. 88, Quebec City, CA, June 2012, pp. 289–302.
- [13] De Leeuw, J. and Castles, W., “The Normal Component of the Induced Velocity in the Vicinity of a Lifting Rotor and Some Examples of its Application,” Tech. Rep. No.1184, NACA, 1955.
- [14] Seddon, J., *Basic Helicopter Aerodynamics*, BSP Professional Books, Oxford, Eng., 1990.
- [15] Bramwell, A., *Bramwell’s Helicopter Dynamics*, Butterworth-Heinemann, Oxford, Eng., 2001.
- [16] Rodriguez, C., *CFD Analysis on the Main-Rotor Blade of a Scale Helicopter Model using Overset Meshing*, Master’s thesis, KTH Royal Institute of Technology, 2012.
- [17] Leishman, J. G., *Principles of Helicopter Aerodynamics*, Cambridge University Press, New York, US, 2006.
- [18] Borst, H. V., “Summary of Propeller Design Procedures and Data Vol. 1,” Tech. Rep. No. 73-34A, Army Air Mobility Research and Development Laboratory, 1973.
- [19] McCormick, B. W., *Aerodynamics, Aeronautics, and Flight Mechanics*, Vol. 2, Wiley New York, New York, US, 1995.
- [20] Ghazbi, S. N., Aghli, Y., Alimohammadi, M., and Akbari, A., “Quadrotors Unmanned Aerial Vehicles: A Review,” *International Journal on Smart Sensing and Intelligent Systems*, 2016.
- [21] Gustafson, F., “Effect on Helicopter Performance of Modifications in Profile-Drag Characteristics of Rotor-Blade Airfoil Sections,” Wartime Report No. 6447, NACA, 1944.

## BIBLIOGRAPHY

---

- [22] Tetervin, N., “Tests in the NACA Two-Dimensional Low-Turbulence Tunnel of Airfoil Sections Design to Have Small Pitching Moments and High Lift-Drag Ratios,” Wartime Report No.3113, NACA, 1943.
- [23] Anderson Jr., J., *Fundamentals of Aerodynamics*, Tata McGraw-Hill, USA, 5th ed., 2012.
- [24] Randall, R., Shkarayev, S., Abate, G., and Babcock, J., “Longitudinal Aerodynamics of Rapidly Pitching Fixed-Wing Micro Air Vehicles,” *Journal of Aircraft*, 2012.
- [25] Carroll, T., *The Aerodynamic Simulation, Design and Optimization of Small Rotors in Configuration for Autonomous Aerial Vehicles*, Master’s thesis, Ryerson University, 2016.
- [26] Barcelos, D., “Flow Quality Testing and Improvement of the Ryerson University Low Speed Wind Tunnel,” Undergraduate Thesis, Ryerson University, 2016.
- [27] Brandt, J., *Small Scale Propeller Performance at Low Speeds*, Master’s thesis, University of Illinois, 2003.
- [28] Rwigema, M., “Propeller Blade Element Momentum Theory with Vortex Wake Deflection,” *27th International Congress of the Aeronautical Sciences*, Vol. 2010-2.3.3, Nice, France, 2010.
- [29] Katz, J. and Plotkin, A., *Low-Speed Aerodynamics*, Cambridge University Press, New York, US, 2001.
- [30] Wallace, R., “Graphing Resources: Using Error Bars in your Graph,” <https://www.ncsu.edu/labwrite/res/gt/gt-stat-home.html>, 2005, [Online; retrieved 17-October-2016].
- [31] Coleman, H. W. and Glenn Steele, W., *Experimentation, Validation, and Uncertainty Analysis for Engineers*, Vol. 1, John Wiley and Sons, 2009.