NATIONAL TECHNICAL UNIVERSITY OF ATHENS

DIPLOMA THESIS

Control Oriented Aerodynamic Design Optimization for an Aerial Manipulator

Author:

Gavridis Georgios

Supervisor:

Prof. Kostas J. Kyriakopoulos

A thesis submitted in fulfilment of the requirements
for the degree of Mechanical Engineer

in the

Control System Laboratory
Mechanical Engineering Department

June 11, 2014

Εθνικο Μετσοβειο Πολυτεχνειο

Διπλωματικη Εργασια

Βελτιστοποίηση Αεροδυναμικού Σχεδιασμού

Ιπτάμενου Βραχίονα

Προσανατολισμένη στο Πρόβλημα Ελέγχου

Συγγραφέας:
Γαβρίδης Γεώργιος

Επιβλέπων:
Καθηγητής Κωνσταντίνος Ι. Κυριακόπουλος

Μια διατριβή που υποβάλλεται για την εκπλήρωση των απαιτήσεων
του διπλώματος των Μηχανολόγων Μηχανικών

στο

Εργαστήριο Αυτομάτου Ελέγχου και Ρυθμίσεως Μηχανών
Τμήμα Μηχανολόγων Μηχανικών

11 Ιουνίου 2014

Abstract

Historically, aerial vehicles with more than two rotors have not been very commonly used, primarily because most of the preferable payloads could be lifted by using only one or two rotors. However, these aerial vehicles possess several characteristics which make them attractive. Naturally, the first one is the superior payload capacity and the second is the simplicity of the necessary control system. In fact, the control of such a vehicle can be achieved by independently tuning the speed (rpm) of each rotor. This control system is particularly suitable for small UAVs, because it reduces the mechanical complexity of the rotors reducing in that way the volume and the weight of the body-structure respectively.

Therefore, taking the above into consideration, still remains challenging to structure a completely new aerial vehicle that will be classified also in rotary-wing VTOL air crafts. This air craft will be called Aerial Manipulator and could be considered as a small UAV that interacts with the environment via an end-effector by applying actuating force and torque. As this is an aerial vehicle, it is essential to have a complete understanding of the rotary-wing VTOL flight principles that will be applied partially in this analysis.

This Diploma thesis presents a detailed Static Design Model of this new UAV that will define the optimal number of rotors, their locations as well as their directions on the body structure, with respect to the design limitations. These locations and directions are products of several optimization problems with general goal the low body volume, for which solving methodology is proposed. In addition to that, with the aim of introducing possible aerodynamic interaction between the rotors appropriate experimental results were considered.

Moreover, a search was made on the market so as to select appropriate flying components such as propellers and motors that will match the Design requirements.

To conclude, the overall mathematical equations implied from the Static Design problem accompanied by the characteristics and the limitations incorporated from the commercial components, were programmed in Matlab environment. In this manner, can be found the maximum actuating force and torque the Aerial Manipulator can apply via the end-effector when using specific motors and propellers.

Περίληψη

Κατά την διάρκεια των τελευταίων χρόνων, τα εναέρια οχήματα με παραπάνω των δυο στροφείων δεν είναι ευρέως χρησιμοποιούμενα, κυρίως γιατί τα απαιτούμενα φορτία μπορούσαν να ανυψωθούν μόνο με ένα ή δύο στροφεία. Ωστόσο, αυτά τα εναέρια οχήματα διαθέτουν διάφορα χαρακτηριστικά που τα καθιστούν ελκυστικά. Το πρώτο είναι το υψηλό ωφελούμενο φορτίο ανύψωσης και το δεύτερο είναι η απλότητα του απαιτούμενου συστήματος ελέγχου. Για την ακρίβεια, ο έλεγχος αυτού του είδους του οχημάτων πραγματοποιείται ρυθμίζοντας ανεξάρτητα την ταχύτητα κάθε στροφείου. Αυτού του είδους το σύστημα ελέγχου είναι κατάλληλο για μικρού μεγέθους Μη Επανδρωμένων Εναέριων Οχημάτων (UAV), γιατί μειώνει την πολυπλοκότητα κατασκευής μειώνοντας ταυτόχρονα τον όγκο και το βάρος της.

Έτσι, λαμβάνοντας τα παραπάνω, δημιουργήθηκε η ανάγκη κατασκευής ενός καινούργιου εναέριου οχήματος το οποίο κατατάσσεται επίσης στα οχήματα περιστρεφόμενων πτερυγίων που είναι ικανά για Κάθετη Απογείωση και Προσγείωση (VTOL air crafts) . Αυτό το αεροσκάφος με ονομασία *Ιπτάμενος Βραχίονας*, μπορεί να θεωρηθεί ως ένα μικρού μεγέθους Μη Επανδρωμένο Εναέριο Όχημα το οποίο θα αλληλεπιδρά με το περιβάλλον μέσω ενός τελεστή δράσης εφαρμόζοντας ωθούμενη δύναμη και ροπή. Καθώς πρόκειται για ένα εναέριο όχημα, είναι σημαντικό να γίνουν πλήρως κατανοητές οι αρχές πτήσης - ενός οχήματος με περιστρεφόμενα πτερύγια που είναι ικανό για Κάθετη Απογείωση και Προσγείωση - οι οποίες θα εφαρμοστούν μερικώς εδώ.

Σε αυτήν τη Διπλωματική Εργασία παρουσιάζεται ενδελεχώς ένα Στατικό Μοντέλο Σχεδίασης, αυτού του Μη Επανδρωμένου Οχήματος, το οποίο θα καθορίζει τον βέλτιστο αριθμό στροφείων, τις θέσεις τους καθώς επίσης και τον προσανατολισμό τους πάνω στο σώμα της κατασκευής, με γνώμονα τους σχεδιαστικούς περιορισμούς. Αυτές οι θέσεις και οι προσανατολισμοί είναι απότοκα διαφόρων προβλημάτων βελτιστοποίησης που έχουν ως γενικό στόχο τον χαμηλό όγκο του οχήματος. Για τα προβλήματα βελτιστοποίησης προτάθηκε μεθοδολογία επίλυσής τους, αφού παρουσιάζουν ποικιλία δυσκολιών. Πέραν αυτών στην παρούσα διατριβή, με στόχο την αποφυγή των αεροδυναμικών αλληλεπιδράσεων μεταξύ των στροφείων εξετάστηκαν κατάλληλα πειραματικά αποτελέσματα που πραγματοποιήθηκαν στον Εργαστήριο Αυτομάτου Ελέγχου και Ρυθμίσεως Μηχανών. Επιπρόσθετα, έμφαση δόθηκε στη αγορά/εμπόριο ερευνώντας πιθανά εξαρτήματα, όπως προπέλες και κινητήρες, που θα ανταποκρίνονται στις σχεδιαστικές απαιτήσεις του Ιπτάμενου Βραχίονα.

Καταλήγοντας, όλες οι μαθηματικές εξισώσεις και σχεδιαστικοί περιορισμοί που εισάγονται τόσο από το Στατικό Μοντέλο όσο και από τα εμπορικά εξαρτήματα, προγραμματιστήκαν σε περιβάλλον Matlab . Με αυτόν τον τρόπο, λοιπόν, βρίσκεται στην εκάστοτε περίπτωση η ωθούμενη δύναμη και ροπή που ο Ιπτάμενος Βραχίονας μπορεί να εφαρμόσει μέσω του τελεστή δράσης, όταν χρησιμοποιείται συγκεκριμένος συνδυασμός προπέλας - κινητήρα.

Acknowledgements

Ευχαριστίες

Contents

List of Figures

List of Tables

Abbreviations

AM	**A**erial **M**anipulator
BET	**B**lade **E**lement **T**heory
HP	**H**ub **P**lane
LP	**L**inear **P**rogramming
MMT	**M**odified **M**omentum **T**heory
MT	**M**omentum **T**heory
NLP	**N**on **L**inear **P**rogramming
NSO	**N**on **S**mooth **O**ptimization
SVD	**S**ingular **V**alue **D**ecomposition
TPP	**T**ip **P**ath **P**lane
TWS	**T**urbulent **W**ake **S**tate
VRS	**V**ortex **R**ing **S**tate
UAV	**U**nmanned **A**erial **V**ehicle
VTOL	**V**ertical **T**ake- **O**ff and **L**anding
WBS	**W**indmill **B**rake **S**tate

Dedicated to

my father Christos, my mother Spyridoula
and to my brother Alexandros

Στον πατέρα μου Χρήστο, στην μητέρα μου Σπυριδούλα
και στον αδερφό μου Αλέξανδρο

Chapter 1

Introduction

1.1 General

It is true that, over the last years much progress has been made in the field of the UAVs and emphasis was given, among others, on developing vehicles of the form of quadrotors or hexarotors, that their lift power is produced from rotors positioned in the same level. These vehicles are capable of hover, forward flight and vertical take off and landing, therefore it should be classified as rotary-wing VTOL air crafts.

Traditionally the configuration of these vehicles has not been used in the aerospace industry, mainly because most of the usual payloads could be lifted using one or two rotors. However, they have some special characteristics that make them attractive. One, of course, is the superior payload capacity. The other is the simplicity of their control system.

Considering these, it is challenging to structure a completely new aerial vehicle that will be classified also in rotary-wing VTOL air crafts. This air craft will be called **_Aerial Manipulator_** and could be considered as a small UAV that interacts with the environment via an end-effector by applying actuating force and torque. In contrast to quadrotors/hexarotors the rotors of the Aerial Manipulator will not necessarily be positioned in the same level.

1.2 Problem statement

The aim of this Diploma Thesis is to produce a Static Design Model for the Aerial Manipulator that defines the optimal locations and directions of a number of rotors on the body structure, with respect to design limitations. These locations and directions are

products of several optimization problems for which solving methodology is proposed here in terms of minimizing the structure volume. Notice also at this point that the aerodynamics of each rotor are taken into consideration. The Design Model will incorporate the mathematical equations that define the structure and will be programmed in Matlab environment.

Naturally, in order to be successful in that task it is essential to have a complete understanding of the rotary-wing VTOL flight principles that will be applied partially in Aerial Manipulator. Moreover emphasis was given on gathering appropriate commercial flying components (such as motors or propellers) so as the model to take a materialized form. Ideally, the proposed Model should have enough flexibility to be adapted in the future simulations.

1.3 Overview of the work undertaken

A complete description of the work that made will be done in the following Sections but in here it would be enough to briefly describe the tasks that were performed.

Firstly, an extensive work of search was made so as to acquire a basic understanding about the characteristics of the rotary-wing air crafts in general. This research was mainly based on the resources available on the Internet, since the traditional sources (such as books, articles) contained information that did not match entirely. More specifically, emphasis was given on searching physical models for quadrotors since the characteristic of these vehicles are closer to the Aerial Manipulator. It was seen that many models use a convenient simplification in mathematics which can lead to the omission of important effects, such as the blade flexibility, that may significantly affect flight states. In additions to that, there are researchers that use equally principles of the helicopter theory in order to produce models. This though is not accurate as there are many differences in helicopter's flight states. For instance, conventional helicopters keep rotor speed constant which is not happening in Aerial Manipulator as will be controlled by independently modifying the speeds of the rotors.

This analysis will try to incorporate a model driven from the quadrotor field (see Chapter 3) into the Aerial Manipulator without omitting crucial details and without making arbitrary simplifications. Underline in here that this Thesis will not focus on dynamic analysis since the Static Model is the goal but as it seen in retrospect, the Design Model requires some flying principles so as to be completed. This issue occurs afterwards when calculating thrust force and torque for the Simulation. In that way the analysis will

be only in a more static flight state, this of hovering, but the proposed model will be capable to be extended also in other flight states.

The next phase is to introduce the Static Design Model of the Aerial Manipulator (Chapter 4). A Model that will incorporate all the appropriate design factors and limitations of such structure. In this way, the optimal location and direction of each rotor are defined on the structure with respect always to a low body-volume. Important design specifications concerning the fully ranked matrices, the influence of fluctuations in actuating force and torque values on the structure allocation or the aerodynamic effects of each rotor, are taken seriously into consideration. Thus, effort was made so as the fluctuations in the output values (actuating thrust/torque) deliver a small change in the input arguments (position/orientation and thrust force of each rotor) of the Aerial Manipulator. In addition to that, with the aim of introducing possible aerodynamic interaction between the rotors appropriate experimental results were considered.

All the specifications of the Static Design Model are gathered into an optimization problem which must be solved. Once the above models were ready, it is possible to build the Matlab simulation and propose a methodology that will solve this problem (Chapter 5). A problem that has dis-joint feasible regions and is characterized from its non-smooth and non-linear constraints. So there was made a major search on how to handle this type of problem, which requires high computational power, and were tested different procedures and packages. The proposed methodology of solving the optimization problem aims on producing sufficiently accurate results in possible future simulations.

After this, a search was made on the market so as to select appropriate flying components such as propellers and motors that will match the requirements of the Aerial Manipulator. Here was made a combination between the manufacturer performance data and the model used to approximate the flight principles. That happened because a gap was observed in some critical information the manufacturer offers to the consumer when calculating the thrust force and torque. Apart from this, it is critical to provide a model that approximates sufficiently the distribution of thrust and torque as the control of the Aerial Manipulator will be achieved by tuning each rotor's rpm.

Finally, the Matlab environment was used so as to incorporate the characteristics of the commercial components into the analysis and produce more realistic simulation results concerning the Static form (Chapter 6). With that way, it was found the maximum actuating force and torque the Aerial Manipulator can apply via the end-effector when using specific motors and propellers.

Chapter 2

Background and general information

2.1 Introduction

As it was stated this diploma thesis focuses on analysing the fundamentals of the *Aerial Manipulator*. But before commenting in detail the characteristics of this structure, it is considered to be necessary to have a general knowledge about the so-called "Unmanned aerial vehicle", such as operating principles, potential applications/uses, potential operational environment and advantages-disadvantages.

2.2 Unmanned aerial vehicle

From the definition *unmanned aerial vehicle* (UAV), commonly known as drone, someone can easily understand that is this aircraft without any human pilot on board. Its flight is controlled either autonomously by computers in the vehicle or under the remote control of a pilot on the ground or in another vehicle. The earliest attempt at a powered unmanned aerial vehicle was A. M. Low's "Aerial Target" of 1916. Nikola Tesla described a fleet of unmanned aerial combat vehicles in 1915. After these, numerous attempts were made during the World War I, and till now these aerial vehicles exist in different shapes, sizes, are structured for different applications and based on different principles (with or without rotor blades). This diversity is shown by the figures that follow.

FIGURE 2.1: Aeryon Scout In Flight.

FIGURE 2.2: In December 2013, the DHL tested a "microdrones md4-1000" for delivery
of medicine.

The *Aerial Manipulator* is a rotor-craft, composed of a number of rotors, which will
be revealed later on this thesis. It should be capable -above all - of hovering, forward
flight and vertical take off and landing. Also, due to its own end-effector should be
capable of interacting with the environment. So taking also into consideration the above,
AerialManipulator can be classified as UAV. This happens because of its own nature of
controlling the flight, which is without an on board human pilot.

FIGURE 2.3: Predator launching a Hellfire missile.

FIGURE 2.4: The RQ-7 Shadow is capable of delivering a 20 lb (9.1 kg) "Quick-MEDS" canister to front-line troops.

The ability for air vehicles to manipulate a target or carry objects they encounter could greatly expand the types of missions achievable by unmanned systems. Flying robots with dexterous arms could lead to transformative applications in near-Earth environments. Such applications could be infrastructure inspection and repair, agricultural care and possibly even construction and assembly. Although work has been made in this area with ground-based vehicles, little work has been done in aerial vehicles where arm or manipulator motions.

(A) Yale Aerial Manipulator carrying toolcase in hover [1]

(B) A quadrotor with a gripper

(C) Four quadrotors carrying a payload [2]
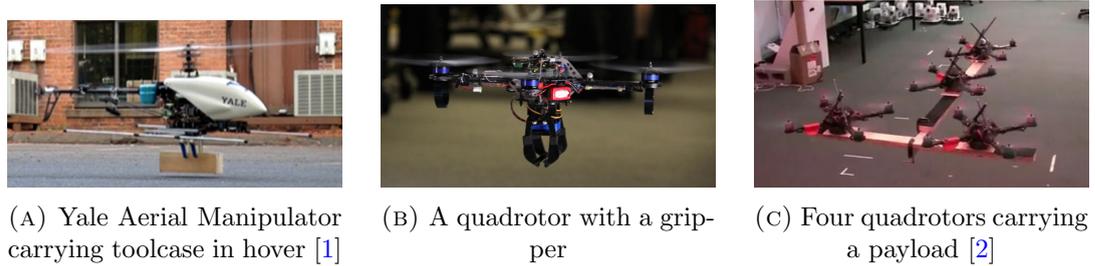
FIGURE 2.5: A figure that depicts different Aerial Manipulators

2.2.1 Mission of UAV

It is natural that determining the general Mission of an UAV is crucial when defining the flight requirements of such vehicle. In other words, according to the desirable role and to the desirable operation, there are changes into the Dynamic and Static Models of the UAV and of course there are changes into the whole Design and Control Analysis. Thus, defining the Mission of such vehicle has important role to each Analysis.

As it was referred, the major development and the begging of the research on the UAVs was made because of War necessities. Yet beyond the military applications of UAVs with which "drones" became most associated with, numerous civil aviation uses have been developed, including aerial surveying of crops, acrobatic aerial footage in film making, search and rescue operations, inspecting power lines and pipelines, and counting wildlife, delivering medical supplies to remote or otherwise inaccessible regions and others.

FIGURE 2.6: Fulmar UAV, developed by Aerovision for civilian applications.

FIGURE 2.7: A thermal imaging gimbal pod camera mounted on the side of a Huey
UH-1.

2.2.2   The operational environment

As it shown from the previews Section, according to the General Mission that the UAV
is designed to accomplish depend the environment that is obliged to operate. A typi-
cal operational requirement will include a definition of the environmental conditions in
which the UAV needs to work in terms of temperature, density altitude, wind strength
and visibility. These things will then be reflected in the air-craft's design. In general the
requirements wording may take the form: this UAV must be able to operate (i.e., con-
duct its intended mission, including start-up and shut-down) in a variety of conditions
depending on the mission. So defining the appropriate operational condition environ-
ment is critical also for the Analysis of the *Aerial Manipulator*. *Aerial Manipulator* at
this thesis is supposed to be a *low altitude* flying vehicle, operating *low speed* flights and
with aerodynamic effects having *low impact* on it. Also as it mentioned already *Aerial*

Manipulator is a vehicle that interacts with its environment via actuating Torque and Force.

Thus, ideal condition for that vehicle would be with: wind speeds of approximately (or less than) $10knots$, $60ft$ approximately of altitude and conditions of the typical day with $T = 15^oC$ temperature, $p = 101325N/m^2$ pressure and $RH = 60\%$ of relevant moisture. Also that body-structure should be capable of taking off from any direction and landing in day or night. This description defines the limits to the operational capability in the form of a multidimensional envelope. As this thesis would be the first attempt of designing, these conditions are satisfying at this point.

2.2.3   Advantages and Disadvantages

The advantages and the disadvantages respectively generally in UAVs differ for each category, for each type, size and structure. In here emphasize will be given on the small size types of vehicles (for example quadrotors, hexarotors) that their movement based on the rotor thrust force. That happens because also *Aerial Manipulator* could be categorized in such vehicles with relatively "small" rotors and for "small" range distances operations. So here are mentioned the advantages and the disadvantages that correspond also to the *Aerial Manipulator*.

Advantages

In that way the major advantages are:

- Higher payload capacity

- Reduced gyroscopic effects

- Simplicity of the control system

Higher payload capacity

The more obvious advantage is the first one. The thrust developed by a rotor increases with its diameter. Thus, by increasing the diameter it is possible to increase the thrust and therefore the payload which can be lifted. However, there is a limit as to how much the diameter can be increased, which is imposed by the compressibility effects that occur at the tip of the blade when it is moving so fast that it approaches the transonic region. Even then, it is possible to augment the thrust by adding more blades to the rotor, but this also has a limit, imposed both by the increasing mechanical complexity and by the

interaction between the wakes of the blades. So if the thrust has to be raised even more, it is necessary to add more rotors to the structure but also for economic and technical reasons the number of rotors should be kept as low as possible.

Reduced gyroscopic effects

Gyroscopic effects can affect every rotating body, including the rotor of a helicopter. However, depending on the type of rotor, these gyroscopic effects will be different. An articulated rotor will tend to behave like an ideal gyroscope, which means that its angular momentum vector will tend to keep the same orientation when the helicopter changes its attitude. On the other hand, a perfectly rigid rotor will introduce a gyroscopic moment on the airframe when there is a change in attitude. Other types of rotors will behave in an intermediate way. In all the cases, the gyroscopic effects will depend on the rpm and the direction of rotation of the rotor.

Simplicity of the control system

In general the simplicity of the control system of such a vehicles based on the fact that it is possible to control the altitude just by adjusting separately the rpm of each rotor. And of course there are also other control methods that do not consist of varying the rpm. But in this diploma thesis emphasis will not be given in the control Analysis. In stead Static analysis and overcoming certain problems will be the main prospect.

Disadvantages

And on the other hand the major drawbacks of these type of UAVs are:

- Higher weight. Lower payload/weight ratio

- Bigger power consumption

- Coupling between controllability and motor dynamics

- Technology in its infancy

Higher weight. Lower payload/weight ratio

As for the higher take off weight, it is an obvious conclusion of the fact that, instead of one or two main rotors, there are four. Regarding the low payload/take off weight ratio, it is not so obvious. On the one hand, the take off weight is larger, as it has been explained. But on the other hand, the payload is also larger, because the thrust available is bigger.

Bigger power consumption

This is another consequence of having more rotors. Bigger power consumption implies bigger power plants and bigger energy reserves (either batteries or fuel tanks), and this in turn implies higher take off weight, which was already high because of the increased number of rotors.

Coupling between controllability and motor dynamics

However, it should not be forgotten that the speed of the rotors depends strongly on the dynamics of the motor(s) driving them. Any motor or engine, no matter of what type (electric, internal combustion, gas turbine, steam powered) has a certain inertia to changes in its regime (i.e., speed). The larger the inertia, the larger the time lag. Depending on the type of motor/engine and its size, the time lag may differ in several orders of magnitude, but it will never be zero (no engine has an instantaneous response). Because of this, whenever a change in the speed of the rotor (motor) is demanded, there will be a time lag until this change is fully implemented, its length being of the same order of magnitude as the time constant of the motor.

Technology in its infancy

Unlike the rest of the disadvantages, this will disappear as soon as *Aerial Manipulator* is designed and operated. Until then it remains a primary concern, especially for manned vehicles, where reliability is essential.

Chapter 3

Modelling

3.1 Introduction

In this Chapter emphasis will be given on the models that will be used in order to approach the different parts of the *Aerial Manipulator*. More specifically effort was made on modelling each rotor, the induced flow throughout the rotation of the blade and on modelling the aerodynamics of the airframe. Apart from simply stating the models that were used, reasoning these selections will be a priority.

3.2 Selecting rotor model

Obtaining a physical model of the rotor is essential for the study of the Aerial Manipulator. Models will be restricted to the blades and more precisely, it will be a model of the physical system formed by the b blades of the rotor. The other element of the rotor, the hub, will not be part of this model. Since the selection of this model will greatly influence the outcome of the whole analysis, it is worth taking some time to examine the different choices available, its advantages and its limitations.

3.2.1 Available Models

There are both empirical as well as theoretical models concerning rotor/blade approach.

Empirical models

Empirical models are all the models that are almost exclusively based on empirical data, typically obtained in the wind tunnel. In order to obtain the large amounts of

data which are required, many wind tunnel tests have to be carried out. Statistical software packages have then to be used to identify trends and find correlations. That is the reason why this analysis will not be based on these type of approach due to its own necessity of test procedure.

<u>Theoretical Models</u>

There are models that are based on a variety of theories and their purpose is to explain the physical behaviour of a rotor. Some of them are those that are follow [3]

- Blade Element Theory (BET)+ Momentum Theory (MT)

- Prescribed Wake Methods

- Free Wake Methods

- Solving methods for the Navier-Stokes equations

As a matter of fact, from all these methods only the first one is affordable in terms of effort and resources considering the scope of this Project and the time assigned to it. The main disadvantage of these models is that they all simplify the real system to a certain extent. More complex models will make fewer assumptions and thus be more accurate. But there will always be some error due to the inevitable simplifications that have to be made. This is an important but will not be handled on this thesis. On the other hand, these models are much more flexible than their empirical counterparts.

Considering that Navier-Stokes equations are completely out of the scope of this analysis, everything will be limited to the models based on the Momentum and Blade Element Theories.

Blade Element-Momentum Theory is a theory simultaneously calculates the induced velocity and the coefficients of forces and moments. It was initially considered but was quickly rejected because the formulation that was found in the published works was not detailed enough. For example, we were interested in modelling the blade flapping and the influence on it of the blade stiffness, but no book or article was found describing how to do so with this theory. Therefore it was decided to use the Momentum Theory (or a variation of it) for the modelling of the induced flow and Blade Element Theory (BET) for the coefficients of forces and moments. Even then, there were still many possibilities depending on the assumptions made for BET:

<u>1st order flapping</u> This means that the flapping angle is assumed to be a first order series of sines and cosines of the azimuthal angle of the blade. Models that assume first order

flapping are by far the most extended, at least for non-commercial applications. There are several particularly well-known:

Newman [4] developed a simple model where real blade flapping dynamics are simulated by means of an equivalent system composed by a rigid blade with a zerooffset hinge. To simplify the formulation, other hypotheses are made. Because of its simplicity, this model is very widespread. This model was used by Hoffmann et al for their quadrotor, with satisfactory results.

Prouty [5] developed a very detailed mathematical model of the rotor of a conventional helicopter. A very simplified version of this model was later used by Pounds et al for their quadrotor. This model uses an offset hinge with no spring to simulate the stiffness of the blade. It considers the effect of the angular rates on the flapping angles, but not on the coefficients of forces and moments. It can produce very satisfactory results, as shown by Pounds et al [6], [7].

Padfield's model [8] includes the effect of the angular rates on the coefficients and gives even more attention to the problem of determining the flapping angles. It uses a hinge with spring and no offset to simulate the dynamics of the blade. Overall, this model is more detailed that the previous one, yet lacks its popularity, possibly because it is more complex.

Bramwell's model [9],[10] is similar to that of Prouty, with not so many details .

2nd order flapping One well-known model of this type is the one developed by Wheatley [11] and Bailey [11].

3.2.2 Summarizing

It was decided to reject second order flapping models because the benefits of including second order harmonics probably did not compensate for the additional complexity. As for the first order flapping models, Padfield's was the most detailed, but perhaps excessively complex. On the other hand, Newman's was too simple. Between Bramwell's model and that of Prouty, the latter was more comprehensive and more detailed in some aspects. Besides, Prouty's model had the advantage of having already been used for quadrotor modelling [6],[7] and that would be a great benefit for that analysis. Therefore, it was decide to follow Prouty to develop the rotor model for the Aerial Manipulator.

3.3 Rotor modelling

This section incorporates to an extent all the elements and components that are necessary for "constructing" the Model of each rotor of the Aerial Manipulator.

3.3.1 Describing the motion of the blades

Rotation The motion of the blade around the axis of the rotor or, in other words, the motion of the hub around its axis. This is the most obvious type of motion. The speed of rotation is represented by Ω

Feathering With this term is defined the motion of the blade around its longitudinal axis. In conventional helicopters, where control is achieved by collective and cyclic variations of the pitch, the feathering motion is controlled by the position of the swashplate. The *Aerial Manipulator* on the contrary, is not capable of controlling the pitch angle of the blade, since this feature is not required to control the vehicle. Therefore, the blade will be free to feather and the feathering motion will be the result of the moments acting around the longitudinal axis of the blade. It should be noted, however, that the feathering motion is constrained by the stiffness of the blade around its longitudinal axis. If the blade was completely rigid, there would be no feathering motion. Although feathering is important, specially because of its influence on the flapping motion.

Flapping The motion of the blade on a plane which contains the axis of rotation of the rotor. It will be seen that, as in any other rotor-craft, blade flapping has a decisive influence. The origin of this motion lies on the cyclic variations of lift seen by the blade. The blade of a rotor in hover produces the same lift as it turns. However, under other flight conditions of the Aerial Manipulator, the lift varies as the blade turns around the axis of the rotor. This makes the blade move up or down as the lift changes.

So flapping is constrained because the blade is attached to the hub and because it is rigid. In fact, a perfectly rigid blade attached to the hub by a perfectly rigid joint would not flap. However, in real life the blade will not be perfectly rigid. Moreover, in order to alleviate stresses, the joint between the blade and the hub will Modelling of the rotor either be designed to be flexible or just be replaced by a hinge. The consequence of all this is that the blade will be able to flap. This leads to the issue of how to model the flapping and that will be handled later on this thesis.

The flapping angle β [8] is difficult to define in a real rotor. However, in simplified models β can be easily defined as the angle between the rigid blade and the plane of the hub.

Lead-Lag (or Lagging) This is the motion of the blade on its plane. It is a consequence of the flapping motion. When the blade rotates and flaps in the same time, each blade element is subjected to the Coriolis force. This force is perpendicular to the plane of the flapping motion. One of the main consequences of lagging is that the centre of mass of the rotor will no longer be on the axis of rotation. Instead, it will move around it. This will induce vibrations on the vehicle, which can be very intense under certain circumstances[8]. However, this phenomenon is of no importance for the determination of the attitude and trajectory of the vehicle. Therefore the lead-lag motion will be ignored in here.
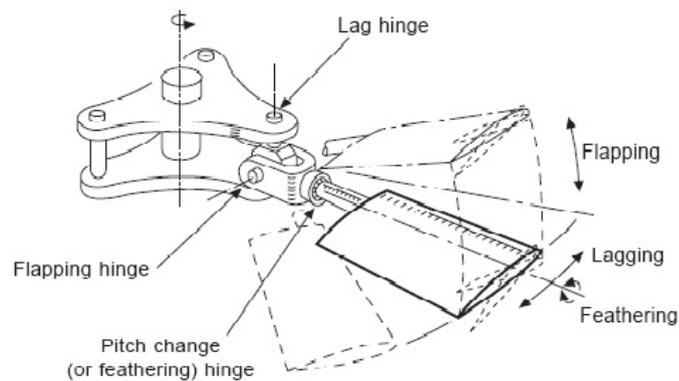
FIGURE 3.1: The different motions of blades.

3.3.2 The simplified models

It has been mentioned that, an exact model which considers the flexibility of the blade is too complicated for this analysis. It is therefore necessary to find a simplified model. Thus, in literature are described the following:

- Centre hinge with spring

- Offset-hinge and spring

- Offset-hinge with no spring

Centre hinge with spring This is an alternative model proposed by Padfield. In this case, though, it is difficult to use. The root of the real blade has an offset, and therefore between the hub axis and the blade root there is no lift generation.

Offset-hinge with no spring In this model the spring is simulated by increasing the offset of the hinge. An "effective hinge offset" is thus defined. The advantage of eliminating the spring is that the mathematical expressions become simpler. This type of model is very common and it has been developed in great detail by Prouty. However, an extra offset is not a perfect replacement for a spring, for several reasons.

Offset-hinge and spring In this model the hinge offset and the spring strength are chosen so that they match one or several of the physical characteristics of the blade. Young suggested selecting the spring strength so that the resulting non-rotating flap frequency matched with the real one. Then, the hinge offset should be chosen so to match the rotating flap frequency.

The figure that follows are showing the three models.
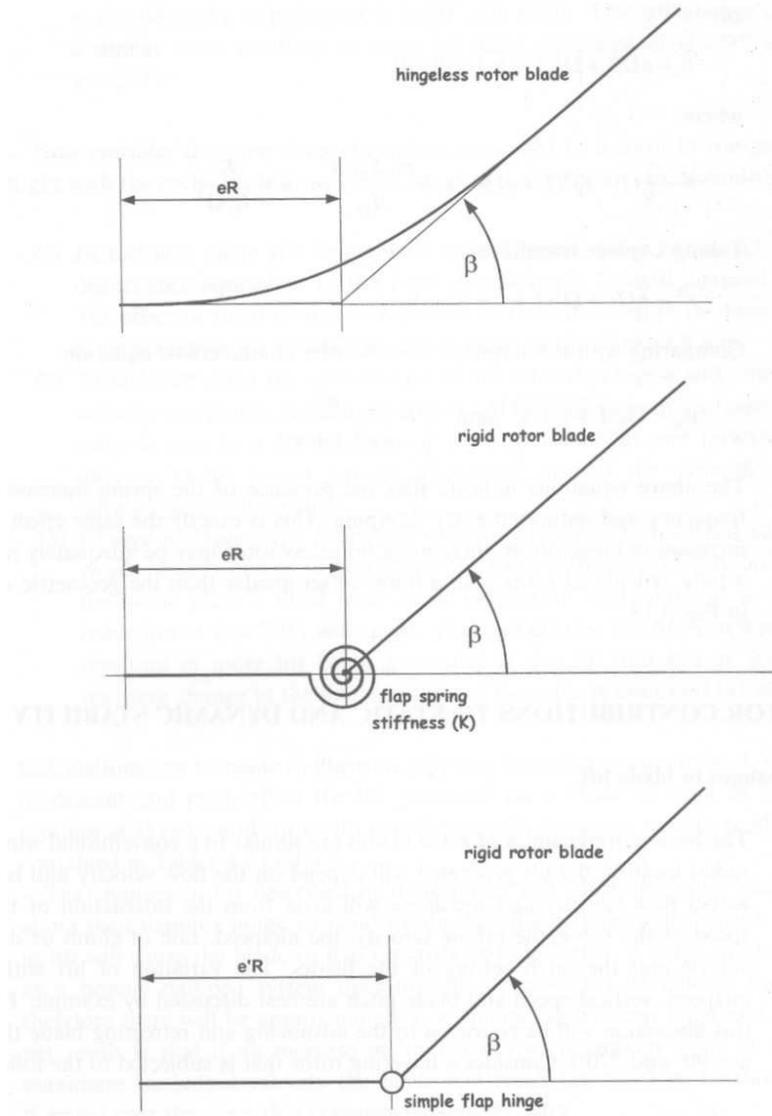
FIGURE 3.2: Real blade (up), Offset-hinge and spring (middle), Offset-hinge with no spring (down) [12]

3.3.3 Tip Path Plane. Flapping angle

The flapping angle β will be a periodic function of the azimuthal angle ψ. When the rotor is in forward flight, ψ is arbitrarily defined in such a way that it is zero when the blade is at the rear (with the front facing the direction of flight). This is shown in the figure below. ψ is positive, increasing as the blade turns. ψ should not be confused with the yaw angle of the airframe. When the rotor is in hover or in axial flight, then

the above definition of ψ fails. However, this will not affect our theoretical results since since they will be averaged for a complete turn, thus enabling us to choose an arbitrary origin of y in hover or axial flight.
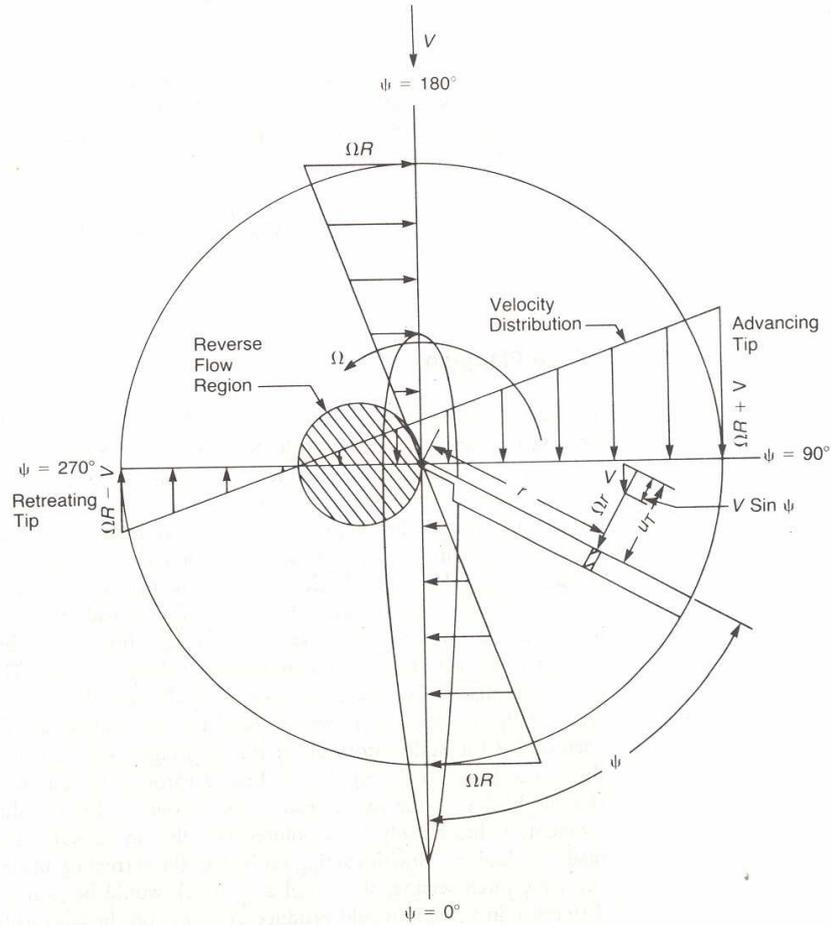
FIGURE 3.3: Azimuthal angle y in a conventional helicopter. Note that the front of the helicopter is aligned with the velocity vector [5].

Since β is a periodic function of ψ. With the usual notation:

$$\beta = a_0 - a_{1s} \cos \psi - b_{1s} \sin \psi \tag{3.1}$$

a_0 is called the "coning angle" while a_{1s}, b_{1s} , are called "flapping angles" (although strictly speaking the flapping angle is β). From now onwards will be used the term "flapping angle" to refer to a_{1s}, b_{1s}, except if stated otherwise. One very important consequence of choosing a first order flapping is that the path followed by the tip of the

blade is contained in a plane. This plane will be called "Tip Path Plane" or TPP. This plane should not be confused with the "Hub Plane" or HP. Also in when the rotor is hovering or in axial flight ($a = \frac{\pi}{2}$) the TPP and the HP will be parallel.

Also should be noticed that in hover or in axial flight it is $a_{1s} = b_{1s} = 0$ is the logical consequence of the symmetry of that flight condition. However, it does not occur if the cyclic pitch is non-zero. Nevertheless, since the AM is not capable of cyclic pitch control (it lacks the swashplate as HC have) this will be zero, and so $a_{1s} = b_{1s} = 0$ in hover and axial flight.
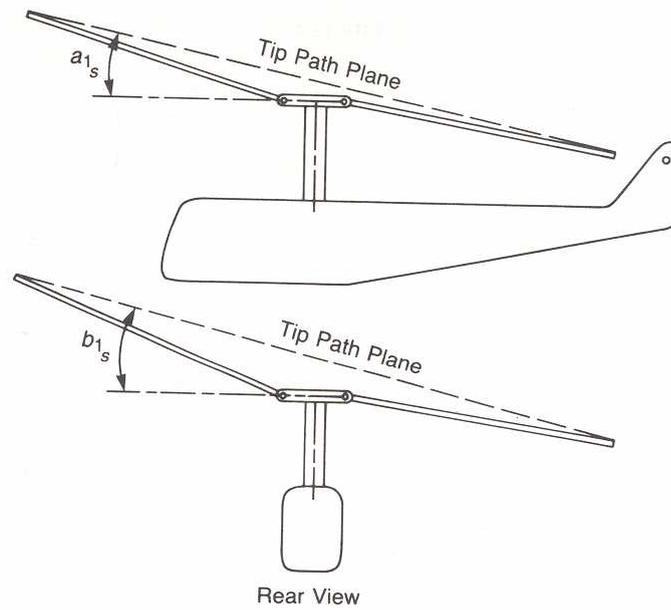
FIGURE 3.4: Flapping angles in a conventional helicopter where the rotor turns counter clockwise [5].

3.3.4 Pitch angle

Pitch angle in literature is defined by the Greek letter θ and is the angle between the zero-lift line of a blade element and the HP. Since the blade has usually a non-zero flapping angle, it is more rigorous to define it as the angle between the zero-lift line and the intersection of the HP with a plane that contains the blade element. Generally it is written as:

$$\theta(r, \psi) = \theta_{root}(\psi) + \theta_1 \frac{r}{R} \tag{3.2}$$

Where r is the distance from the blade root (not from the center of the hub) to the blade element and should not be confused with the yaw rate of the airframe. θ_{root} is the pitch angle at the blade root and it will be a periodic function of the form:

$$\theta_{root} = \theta_0 - A_1 \cos \psi - B_1 \sin \psi \qquad (3.3)$$

Notice that in a conventional helicopter θ_{root} is controlled by the position of the swashplate.
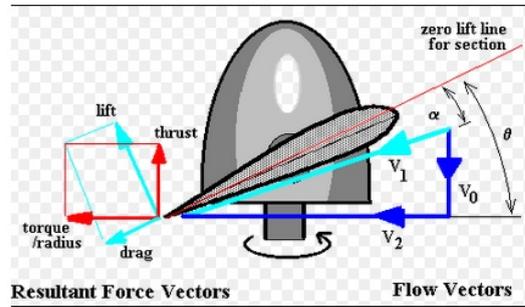
FIGURE 3.5: Blade view. V_0 is the axial flow at propeller disk, V_1 section local flow velocity vector, v_2 is the angular flow velocity vector. θ is the pitch angle

However considering the fact that - as it mentioned - AM there is no swashplate to control pitch, since there is no need to do so in order to control the vehicle (contrary to what happens in conventional helicopters). The blades are attached to the hub and there is no articulated joint that enables the blade to turn around its longitudinal axis, as in conventional helicopters. But this does not mean that the pitch angle is going to be constant, because the blade is flexible around its longitudinal axis. This also happens in conventional helicopters. However, this phenomenon (feathering) in this analysis will be neglected for several reasons that will be stated. Therefore:

$$A_1 = B_1 = 0$$

and

$$\theta = \theta_0 + \theta_1 \frac{r}{R}$$

. But later on this diploma thesis will be mentioned some things concerning the feathering and its elimination.

Notice that as it was stated in hover and axial flight it is $A_1 = B_1 = 0$ (zero cyclic pitch) and $a_{1s} = b_{1s} = 0$.

3.3.5 Blade Element Theory

As it mentioned previously, in this analysis in order to model the aerodynamic forces and moments over the blade, Blade Element Theory (BET) will be followed. The blade is assumed to be made of several infinitesimal strips of width dr . This theory considers, as its own name suggests, a blade element subjected to forces and moments that are below:

Lift(perpendicular to the velocity)

$$dL = \frac{1}{2}\rho c U_R^2 c_1 dr \tag{3.4}$$

Drag (a force opposite the velocity)

$$dD = \frac{1}{2}\rho c U_R^2 c_d dr \tag{3.5}$$

Moment at the aerodynamic center

$$dM_{ac} = \frac{1}{2}\rho c^2 U_R^2 c_{mac} dr \tag{3.6}$$

It is usually instead of dL, dD used dT,dF_T where

$$dT \approx dL$$
$$dF_T \approx dD - \phi dL$$

Where here ϕ is the angle between the air velocity vector an and the HP and is assumed to be small. Also ϕ should not be confused with the roll angle of the airframe. Where U_R is the modulus of the air velocity vector seen by the blade element. This velocity can be separated into two components, U_P, and U_T.

The idea behind BET is that the total force and torque produced by the rotor can be calculated by integrating dT,dF_T,dM_{ac} across r for each blade.

It is true that BET has some limitations compared to some other theories but BET continues to be the most widely accepted method to calculate the total force and torque in a fast and easy way. That is the reason why also it is used in here too. Notice that as it mentioned above that it has been decided to ignore the torque around the longitudinal axis of the blade. Therefore dM_{ac} will not be used.

From Prouty [5]

$$U_T = \Omega R \left(\frac{(r+r)}{R} + \mu \sin \psi \right) \tag{3.7}$$

$$U_P = \Omega R \begin{pmatrix} -\lambda_z - \lambda_{ind}(1 + K_c \frac{r}{R} \cos \psi) - \frac{r}{R}(a_{1s} \sin \psi - b_{1s} \cos \psi) - \\ \\ -\mu(a_0 - a_{1s} \cos \psi - b_{1s} \sin \psi) \cos \psi \end{pmatrix} + (r+e)(q_w \cos \psi + p_w \sin \psi) \tag{3.8}$$

and

$$U_R \approx U_T \tag{3.9}$$

Where in here:

r is the longitudinal coordinate of the blade element (not to be confused with the yaw angular rate of the airframe)

e is the hinge offset

R is the rotor radius, from the axis of the hub to the tip of the blade

$\mu = \frac{V_x}{|\Omega|R}$ is the horizontal speed to tip speed ratio

$\lambda_z = \frac{V_z}{|\Omega|R}$ is the vertical speed to tip speed ratio

$\lambda_{ind} = \frac{V_{ind}}{|\Omega|R}$, where v_{ind} is the induced velocity

Notice also that Ω is positive when the rotor turns counter clockwise and negative in the other case. It should be noted that the (3.8) takes into account the blade flapping and the effect of the angular rates of the hub, p_w, q_w. The induced flow is represented using Glauert's model and will be stated later on this thesis. In this model will be made the appropriate reference for the v_{ind} (the induced velocity). That will happen also for the λ_{ind}.

As already has been mentioned the above model produces accurate results when the rotor is in forward flight ($V_x \neq 0$) but its application in hover or axial flight (that is, when $V_x = 0$) is problematic, because ψ is not defined in that case.

3.3.6   The aerodynamic forces and moments

In order to define the aerodynamic forces and moments again the analysis will be based on the Prouty and:

- The thrust **T**, is assumed to be perpendicular to the Tip Path Plane (TPP)

- The torque vector **Q** is also perpendicular to the TPP.

- A horizontal force **H**, which is contained in the TPP

The representation of these in TPP is in the next figure.
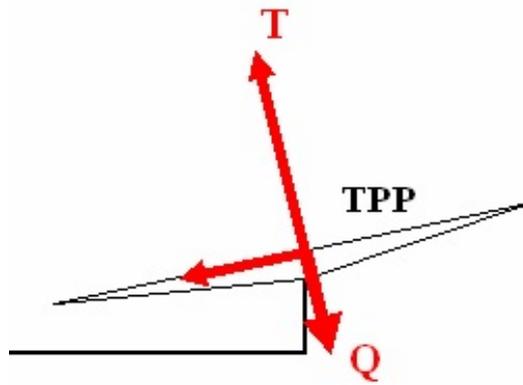
FIGURE 3.6: The thrust T and the torque vector Q are perpendicular to Tip Path Plane (TPP). The force H is parallel to it.

Coefficients

Moreover at this point it is useful to define the following coefficients, as the standard notation in the literature:

$$C_T = \frac{T}{\rho \pi R^4 \Omega^2}$$

$$C_H = \frac{H}{\rho \pi R^4 \Omega^2}$$

$$C_Q = \frac{Q}{\rho \pi R^5 \Omega^2}$$

(3.10)

3.3.7 Calculation of the aerodynamic forces and moments

Through the BET,as it has been anticipated, someone can calculate T,H,Q . This has already been done by Prouty [5]. But before using his results it is important to consider an issue. Being rigorous, the integration of dT and dF_T has to start at the root of the blade, that is, at the first point which generates lift and drag. In the real rotor, this point can be assumed to be the joint between the blade and the hub. But the hinge, in the idealised model, is not going to be coincident with the joint. Therefore it is needed to consider two different offsets, the one of the joint, e_j, and the one of the hinge, e . The problem is that the expressions obtained by Prouty [5] above assume that both offsets are equal.

So that in order to avoid re-formulating all the expressions will be assumed that $e_j \approx e$. Although this introduces some error, this will be much smaller than the one which is going to be introduced by replacing the real blade by an untapered one.

$$\frac{4C_T}{a\sigma} = \left(1 - \frac{e}{R}\right)\left[\left(\frac{2}{3} + \mu^2\right)\theta_0 + \frac{1}{2}(1+\mu^2)\theta_1 - \mu B_1 - \lambda_z - \lambda_{ind}\right] \tag{3.11}$$

$$\frac{4C_H}{a\sigma} = \frac{c_d}{a}\mu - \mu\frac{-\lambda_z - \lambda_{ind} + \mu a_{is}}{1 + \frac{3}{2}\mu^2}\left[\theta_0\left(-\frac{1}{3} + \frac{3}{2}\mu^2\right) + \frac{\theta_1}{2}\left(-1 + \frac{3}{2}\mu^2\right) + \lambda_z + \lambda_{ind} - \mu a_{1s}\right]$$
$$+ \frac{\mu}{1 + \frac{1}{2}\mu^2}\left[\frac{a_0^2}{2}\left(\frac{1}{9} + \frac{\mu^2}{2}\right) + \frac{1}{3}\mu a_0\lambda_{ind} + \frac{1}{8}\lambda_{ind}^2\right] \tag{3.12}$$

$$\frac{4C_Q}{a\sigma} = \frac{1}{2}\frac{c_d}{a}(1+\mu^2)$$
$$- \frac{-\lambda_z - \lambda_{ind} + \mu a_{1s}}{1 + \frac{3}{2}\mu^2}\left[\frac{\theta_0}{3}(2-\mu^2) + \frac{\theta_1}{2}\left(1 - \frac{\mu^2}{2}\right) + \left(1 + \frac{\mu^2}{2}\right)(-\lambda_z - \lambda_{ind} + \mu a_{is})\right] \tag{3.13}$$
$$- \frac{\mu^2}{1 + \frac{\mu^2}{2}}\left[\frac{a_0^2}{2}\left(\frac{1}{9} + \frac{\mu^2}{2}\right) + \frac{1}{3}\mu a_0\lambda_{ind} + \frac{1}{8}\lambda_{ind}^2\right]$$

Where σ is the rotor solidity that is

$$\sigma = \frac{bc}{\pi R}$$

and c is the "mean" chord. It has also been considered that $e << R$, in order to make the expressions above more simple.

As it was previously explained in hover or in axial flight ($\mu = 0$) it is $a_{1s} = b_{1s} = 0$. Besides, examining (3.12) when $\mu = 0$ then $C_H = 0$ also (and $H = 0$)

It was made the hypothesis that drag coefficients are constant, which yet leads to inaccurate results especially for C_Q when $\lambda_z < 0$. Because of this, it is convenient to eliminate this hypothesis and assume instead that the drag polar of the aerofoil can be approximated by a quadratic curve (" three term drag polar "):

$$c_d = c_{d0} + c_{d1}a + c_{d2}a^2 \tag{3.14}$$

Which incorporating the (3.14) into the Prouty's equations (3.12), (3.13) will be transformed into a more accurate form.

$$
\begin{aligned}
\frac{4C_Q}{a\sigma} = {} & \frac{1}{2}\frac{c_{d0}}{a}(1+\mu^2) \\[4pt]
& +\frac{1}{1+\frac{3}{2}\mu^2}\frac{c_{d1}}{a}\left[\theta_0\left(\frac{1}{2}-\frac{19}{36}\mu^2+\frac{3}{4}\mu^4\right)+\theta_1\left(\frac{2}{5}-\frac{2}{5}\mu^2+\frac{\mu^4}{2}\right)+(-\lambda_z+\mu a_{1s}-\lambda_{ind})\left(\frac{2}{3}-\frac{\mu^2}{3}\right)\right] \\[4pt]
& +\frac{1}{\left(1+\frac{3}{2}\mu^2\right)^2}\frac{c_{d2}}{a}
\begin{bmatrix}
\theta_0^2\left(\frac{1}{2}+\frac{2}{9}\mu^2-\frac{\mu^4}{24}+\frac{9}{8}\mu^6\right)+ \\[4pt]
(-\lambda_z+\mu a_{1s}-\lambda_{ind})^2\left(1+2\mu^2+\frac{3}{4}\mu^4\right)+ \\[4pt]
\theta_0\theta_1\left(\frac{4}{5}+\frac{2}{5}\mu^2-\frac{\mu^4}{5}+\frac{3}{2}\mu^6\right)+ \\[4pt]
\theta_0(-\lambda_z+\mu a_{1s}-\lambda_{ind})\left(\frac{4}{3}+\frac{4}{3}\mu^2-\mu^4\right)+ \\[4pt]
\theta_1(-\lambda_z+\mu a_{1s}-\lambda_{ind})\left(1+\mu^2-\frac{3}{4}\mu^4\right)
\end{bmatrix} \\[4pt]
& +\frac{\mu^2}{\left(1+\frac{1}{2}\mu^2\right)^2}\frac{c_{d2}}{a}\left[a_0^2\left(\frac{1}{18}+\frac{\mu^2}{6}-\frac{\mu^4}{8}\right)+\lambda_{ind}^2\left(\frac{1}{8}+\frac{\mu^2}{16}\right)+a_0\lambda_{ind}\left(\frac{\mu}{3}+\frac{\mu^3}{6}\right)\right]
\end{aligned}
\tag{3.15}
$$

And of course,

$$
\begin{aligned}
\frac{4C_H}{a\sigma} = {} & \frac{c_{d0}}{a}\mu+\frac{\mu}{1+\frac{3}{2}\mu}\frac{c_{d1}}{a}\left[\theta_0\left(\frac{1}{9}-\frac{\mu^2}{2}\right)-\frac{1}{2}\theta_1\mu^2+\frac{1}{3}(-\lambda_z+\mu a_{1s}-\lambda_{ind})\right]+ \\[4pt]
& \frac{\mu}{\left(1+\frac{3}{2}\mu^2\right)^2}\frac{c_{d2}}{a}
\begin{bmatrix}
\theta_0^2\left(-\frac{7}{9}+\frac{5}{3}\mu^2-\frac{15}{4}\mu^4\right)+\theta_1^2\left(-\frac{1}{2}+\frac{3}{2}\mu^2-\frac{9}{8}\mu^4\right)- \\[4pt]
2(-\lambda_z+\mu a_{1s}-\lambda_{ind})^2+\theta_0\theta_1\left(-\frac{4}{3}+3\mu^2-\frac{9}{2}\mu^4\right)+ \\[4pt]
\theta_0(-\lambda_z+\mu a_{1s}-\lambda_{ind})(-2+5\mu^2)+ \\[4pt]
\theta_1(-\lambda_z+\mu a_{1s}-\lambda_{ind})(-2+3\mu^2)
\end{bmatrix}- \\[4pt]
& \frac{\mu^3}{\left(1+\frac{1}{2}\mu^2\right)^2}\frac{c_{d2}}{a}\left[\left(\frac{2}{9}\mu+\frac{\mu^3}{3}\right)a_0^2+\frac{\mu}{8}\lambda_{ind}^2+\left(\frac{1}{6}+\frac{5}{12}\mu^2\right)a_0\lambda_{ind}\right]
\end{aligned}
\tag{3.16}
$$

As it mentioned again previously it has been considered that $e << R$ so as to simplify the expressions.

3.3.8 Feathering

As it was described earlier in this section the model has considered that the pitch angle is constant. However, this does not occur in reality. Due to the blade torsional flexibility, the pitch angle will be able to vary cyclically. These cyclic variations are produced by the torques acting around the elastic axis of the blade (the elastic axis needs not be perfectly parallel to the longitudinal axis of the blade). The importance of these cyclic variations lies in the fact that they are going to modify the coning and flapping angles, thus altering the system of forces and moments exerted on the hub. Although it is not completely right, these cyclic variations can be assumed to be governed by the same first order trigonometric expression valid for conventional helicopters with swashplate [8],[10], that is:

$$\theta = \theta_0 + \theta_1 \frac{r}{R} - A_1 \cos \psi - B_1 \sin \psi \qquad (3.17)$$

On the other hand, the analysis on that extent concerning the Feathering is considered to be beyond the main goals of this diploma thesis.

However it is worth mentioning that the procedure in order to solve and determining the Feathering motion has two main problems difficult to overcome. First it might not be possible to uncouple the problem of finding θ_0, A_1, B_1 from finding the already mentioned quantities a_0, a_{1s}, b_{1s} . And secondly a much more serious difficulty is how to determine the torsional stiffness of the blade. So, in that way, considering also the fact there were no experimental tests to be based on, the feathering will have to be ignored in this model.

3.3.9 Reverse flow region

The reverse flow region is the part of the rotor disc where $U_T < 0$. In order for this to happen, V_x has to be non-zero (see equation (3.7)). Inside this region, the lift force over the blade element is entirely different from outside.

Including the effect of the reverse flow region in a simplified way, as Prouty [5] recommends, is certainly possible, although it complicates significantly the expressions.

In any case, the reverse flow region will be kept out of the model.

3.3.10 Blade Taper and Tip Losses

When a blade rotates, each point on it travels at a different speed. The further away from the root, the higher the velocity. This means that the contribution to lift and drag of every point on the blade differs, with each aspect getting larger when moving closer to the rotor tip. Clearly, the lift distribution over the blade is not constant. This is not a desirable situation, because the contribution diminishes when getting closer to the root. To change this distribution, blades are twisted and, sometimes, also tapered. The twist is such that the angle of attack increases when travelling towards the root, producing more lift. Tapering the blade also contributes to achieving a more evenly spaced lift distribution. With blade tapering, the blade's surface gets larger when travelling towards its root. Both tapering and twisting can be observed when looking carefully at rotorblades at rest (Figure 3.9). Note that blade tapering is not always used (especially on metal blades because of a more complicated fabrication process).
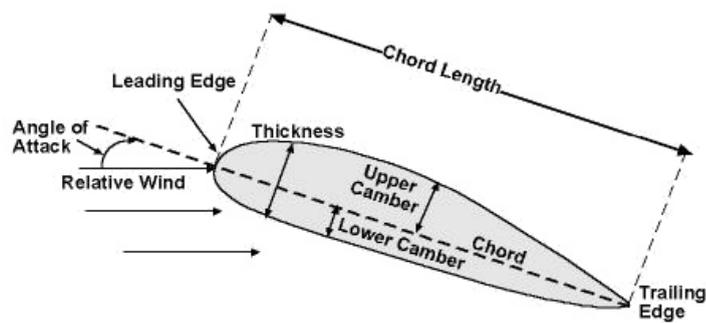
FIGURE 3.7: Angle of attack and characteristics of the blade.
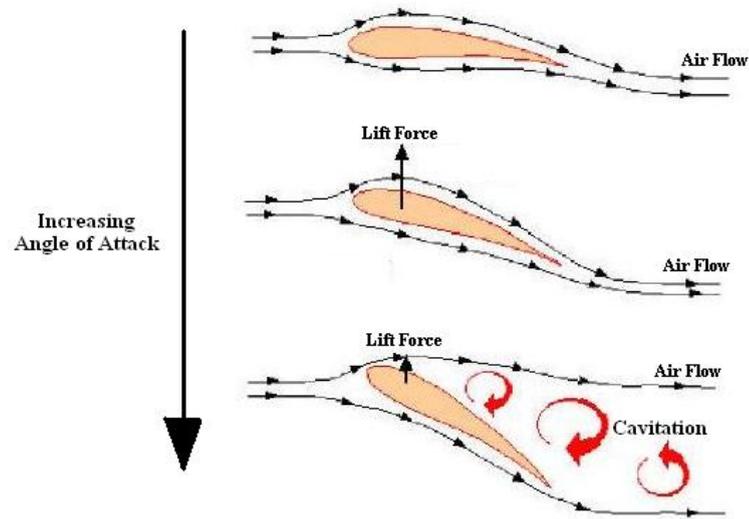
FIGURE 3.8: Different increases in Angle of attack, due to the increase of the angle which is created between the relative flow and the chord of the blade.

There are two types of physical *blade taper*, distal and profile.

Distal tapering refers to a blade's cross-section thinning from its base to its tip. This is used to create the handling characteristics of individual blades and the amount of distal taper varies depending upon the intended purpose of the blade. Many modern replica blades are not made with any distal taper, resulting in a blade that, when wielded, will feel unresponsive and heavy.

Profile taper refers to narrowing upon the edges of the flat of the blade. Blades with a more gradual taper are meant for cutting, whereas blades with an acute taper are usually meant for thrusting.
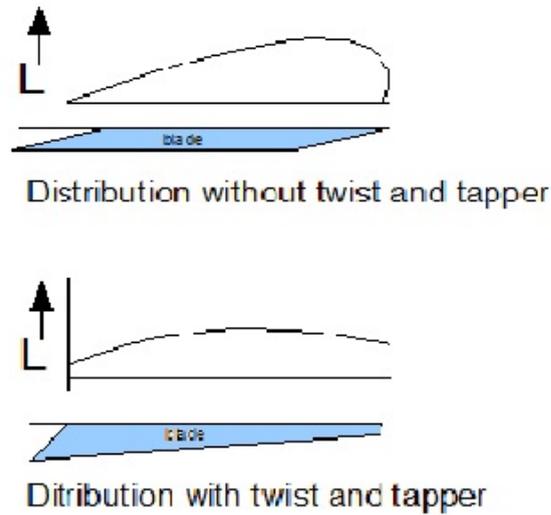
FIGURE 3.9: Distribution of the Lift throughout the entire blade. (Above) Figure shows the Lift distribution without twist and taper and (Low) Figure indicates the re-shaping in Lift distribution due to the existence of twist and taper.

Moreover at the *tip of the rotor blade losses* are introduced but will not be incorporated in this model. On the other hand, it is likely that the effect of blade taper will be of the same order of magnitude as the effect of the tip losses. Thus, both of them will be neglected in here.

3.4 Modelling the Induced Flow

In order to complete the model of the rotor it is necessary to determine the induced velocity, because it is needed for the expressions derived in the previous Section. These expressions were obtained using Blade Element Theory (BET). However, BET does not give any information about the induced velocity, it is therefore necessary to turn into other theories to determine it.

Here will use the model developed by Glauert .

3.4.1 Inflow model by Glauert

Since Glauert's Model is reasonably accurate it was decided to keep this on this diploma thesis.

According to this model:

$$(v_{ind})_{local} = v_{ind}\left(1 + \frac{r}{R}K_c\cos\psi\right) \tag{3.18}$$

Where in here ψ is the azimuthal angle and r is the distance from the blade root to the blade element. v_{ind} is the " average " or " reference " induced velocity, but for the sake of simplicity will be just called it as " induced velocity ". K_c is a constant which remains unspecified in the original work by Glauert. According to Coleman et al [13] , K_c will depend on the wake skew angle. Other authors have proposed different methods of determining it [14] . Still for relatively high values of μ, above 0.2 (around 100 knots in a conventional, manned helicopter), $K_c = 1$ is a good approximation, according to Prouty [5]. But for lower values K_c might even reach 2. Finally, when $\mu = 0$, K_c has to be zero in order to obtain $b_{1s} = 0$.

Generally, v_{ind} will usually be assumed to be equal to the induced velocity predicted by one of the several versions of the Momentum Theory.

3.4.2 Momentum Theory

This theory has the advantage of its simplicity and produces reasonably accurate results when combined with Glauert's model and BET. It is therefore very suitable for real-time simulations. But here emphasis will be given only to the "structuring" of the model for the AM.

Also this subsection will refer only to the "Classical" Momentum Theory (MT)

"Classical" Momentum Theory

The Momentum Theory (MT) based on the following expression for thrust [3]:

$$T = 2\rho\pi R^2 v_{ind}\sqrt{V_x^2 + (V_z + v_{ind})^2} \tag{3.19}$$

Where R is, as usual, the rotor radius (from the axis of the hub to the tip of the blade). V_x is the component of the free stream velocity parallel to rotor disk and V_z is the component perpendicular to the rotor disk. Finally, T is the thrust, perpendicular to the rotor disk. Following Leishman [3] will be assumed the rotor disk to be parallel to the Tip Path Plane (TTP), so that the thrust T of the previous Section is the same as the one here. In the previous Section V_x, V_z where respectively parallel and perpendicular to the Hub Plane (HP). However, since the flapping angles are small, we can also assume that the V_x, V_z of the previous Section are the same as the ones here.

In that way also,

$$C_T = 2\lambda_{ind}\sqrt{\lambda_x^2 + (\lambda_z + \lambda_{ind})^2} \qquad (3.20)$$

In the previous Section using BET (3.11), an expression for C_T was derived. According to it $C_T = C_T(\lambda_z, \mu, \lambda_{ind})$.

Thus, entering (3.11) in (3.20) a new equation will be produced from which yields λ_{ind} and with it v_{ind}. That is the methodology of finding v_{ind}, with solving the new-produced equation numerically.

Once λ_{ind} is know, someone can re-enter in (3.11) to obtain C_T.

So it is possible then, for example, for someone to plot C_T against several discrete values of λ_z.

As it can be seen, MT provides a simple way of calculating v_{ind}. However, MT as it has been presented here has a very serious flaw: it is only valid when the rotor is: (a) at hover, (b) climbing or (c) descending at high speed. In order to understand this better it is thought to be important to presented briefly and explained the different working regimes of a rotor. These are, after Leishman [3]:

- "Normal" or "Helicopter" working state, when $\bar{V}_z \geq 0$ (it should be remembered that V_z was positive when the free stream was blowing from above or, in other words, when the rotor was climbing)

- Vortex Ring State (VRS) when $\bar{V}_z$ negative

- Turbulent Wake State (TWS) when $\bar{V}_z$ more negative than in VRS

- Windmill Brake State (WBS) , approximately when $\bar{V}_z < -2\bar{v}_{ind}$

These States can also be seen in literature be Padfield [8].

The hat symbol means that the variables that have been non-dimensionalized with

$$v_0 = \sqrt{\frac{T}{2\rho\pi R^2}}$$

where T is the thrust delivered by the rotor in each working point.

Non - dimensionalise the equation (3.19) it turns into:

$$1 = \bar{v}_{ind}\sqrt{\bar{V}_x^2 + (\bar{V}_z + \bar{v}_{ind})^2} \qquad (3.21)$$

and in Axial flight ($\bar{V}_x = 0$):

$$\pm 1 = \bar{v}_{ind}(\bar{V}_z + \bar{v}_{ind}) \tag{3.22}$$

Where here the $\pm$ symbol indicates that the equation has "branches".

The figure on next page shows the approximate boundaries of the different working states, in axial flight.
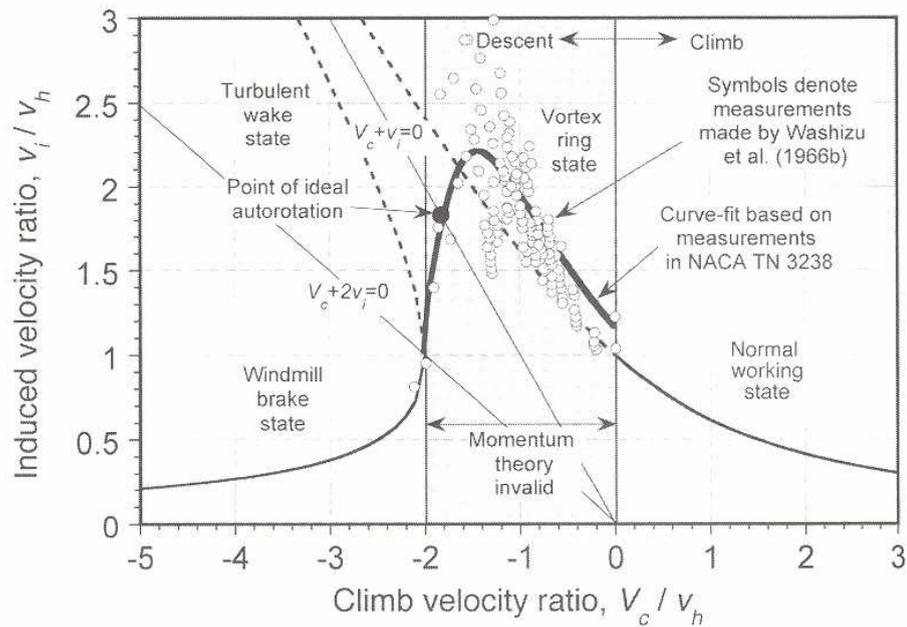
FIGURE 3.10: Induced Velocity Ratio vs Climb Velocity Ratio in axial flight. The two "branches" of the MT solution are shown. Solid lines have been used where those "branches" are valid.

MT will be valid for the "Normal" state and for WBS, but not for VRS and TWS, according to Leishman. This is because the hypotheses in which MT is based are not applicable in the VRS and TWS regions. However, determining the boundaries of these regions, which is equivalent to determining where MT ceases to be valid, is a difficult task. Wolkovitch [15], and later Peters and Chen [15], used a *dynamic* inflow model to estimate the upper limit $\bar{V}_z = -\eta$ of the VRS. According to Wolkovitch:

$$\eta \approx 0.7\bar{v}_{ind}$$

In this point it is important to refer that there is possibility the rotors enter the VRS region under normal operation. That is clear and tested in various of UAV (for example quadrotors). In that way arises the necessity of finding an other model that is valid also in the VRS and TWS region respectively. This is not an easy task.

C.Chen provides a comprehensive list of the different attempts that have been made to obtain a method to predict the induced velocity in the VRS and TWS. Basically, there are two types of methods available:

- those based on a parametric extension of the Momentum Theory

- and those based on wake models (prescribed or free wake)

The former have the advantage of their simplicity, although they lack a solid theoretical background. Wake models represent a radical departure from MT, basing on entirely different hypotheses. They are much more accurate, too, but this comes at the price of a higher complexity (which in turn will lead to the need of larger computational resources).

Thus, in this analysis will be chosen a parametric extension method due to its own simplicity that someone can fine it in Reference [16].

"Modified version of the Momentum Theory"

Having a closer look on the equation (3.23), which for completeness reasons is written again:

$$\pm 1 = \bar{v}_{ind}(\bar{V}_z + \bar{v}_{ind}) \tag{3.23}$$

As it was mentioned this equation will have two solutions. If someone plot $\bar{V}_z$ against $\bar{v}_{ind}$ will be produced two "branches". Parametric Parametric extension methods consist in finding a curve that smoothly joins the two branches and that is reasonably close to the empirical curve. In figure it is shown a possible curve fit, but there are several, for example, that suggested by Johnson [17] or the approximation suggested by Young [18] (which is linear see figure Figure 3.11 below)
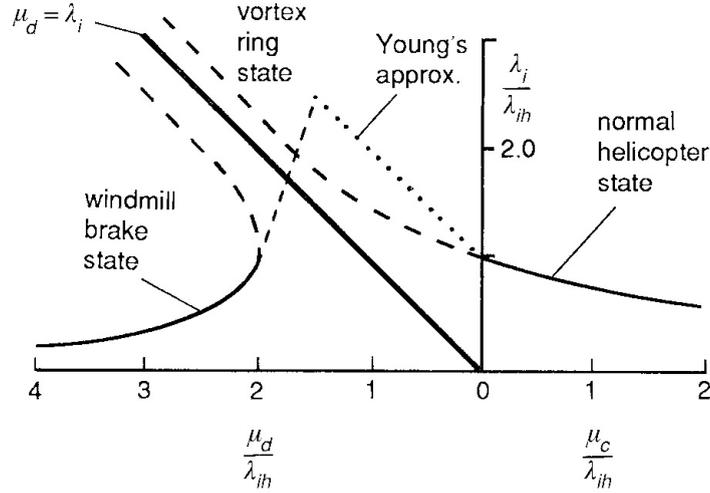
FIGURE 3.11: Momentum theory solutions for rotor inflow in axial flight that incorporates Young approximation [18].

But in here will be used the solution proposed by López Ruiz ([19], [20]), as used in [16], according to which:

$$1 = \frac{\bar{v}_{ind}}{k_1} \sqrt{\left(\frac{\bar{V}_z + \bar{v}_{ind}}{k_1}\right)^2 + \left(\frac{1}{k_2^2 - k_1^2}\right)\bar{V}_z^2 + \left(\frac{\bar{V}_x}{k_2}\right)^2} \qquad (3.24)$$

Where based on the literature $k_1 = \left(\frac{9}{5}\right)^{\frac{1}{4}}$ and $k_2 = \left(\frac{5}{4}\right)^{\frac{1}{4}}$

And with dimensions

$$T = 2\rho\pi R^2 \frac{v_{ind}}{k_1} \sqrt{\left(\frac{V_z + v_{ind}}{k_1}\right)^2 + \left(\frac{1}{k_2^2 - k_1^2}\right)V_z^2 + \left(\frac{V_x}{k_2}\right)^2} \qquad (3.25)$$

and in that way C_T:

$$C_T = 2\frac{\lambda_{ind}}{k_1} \sqrt{\left(\frac{\lambda_z + \lambda_{ind}}{k_1}\right)^2 + \left(\frac{1}{k_2^2 - k_1^2}\right)\lambda_z^2 + \left(\frac{\mu}{k_2}\right)^2} \qquad (3.26)$$

And this will be the equation that will be used in this model. This model will be referred as the "Modified Momentum Theory" (MMT) as also happens in [16].

Thus, entering again (3.11) in (6.5) - this time - a new equation will be produced from which yields λ_{ind} and with it v_{ind}. That is the methodology of finding v_{ind}, with solving the new-produced equation numerically.

In this way, by combining BET and MMT, the rotor model completed.

3.4.3   Correction

Here will be mentioned an empirical correction in λ_{ind} chosen by [16].

When multiplying the value of λ_{ind} obtained with BET/MMT by a certain corrective function, the resulting correlation would be better when λ_z is negative.

It was tested from the literature [16] that :

$$\kappa = 1 - \kappa_0 \lambda_z \tag{3.27}$$

Where $\kappa_0 \approx 1.2$

Notice that after all that the above is completely empirical and there is no physical justification for it. Details will be stated in the Section that follows.

3.5   Analysing the different flight states in Rotor Model

This section will provide the appropriate information about the uses that where made in order to incorporate the above *Rotor Model* into this analysis.

It is true also that apart from modelling the Rotor and its own operations with different proposed methods/models, it is indeed crucial to an extent, specific experimental tests to take place. That happens because "testing" the theory in which an Analysis is based on can after all might result in the avoidance of specific errors and in the avoidance of different arbitrarinesses. Either way considering the time that was given and the main purposes of this diploma thesis, there were held no experimental tests and the models are taken from the literature as referred its time.

3.5.1   Aerodynamic properties of the blade

In order to apply the model that was developed in the previous Section knowing several parameters, which are related to the geometric and aerodynamic properties of the blade, is critical.

These mentioned parameters are listed on the table that follows.

Symbol	Name
θ_0	Collective Pitch
θ_1	Twist
a	Slope of the lift coefficient of the aerofoil
c_{d0}	1st term of the drag polar
c_{d1}	2nd term of the drag polar
c_{d2}	3rd term of the drag polar
c	Mean chord

TABLE 3.1: Required parameters by the Rotor Model

From the parameters listed above, θ_0 assumed to be equal to the static pitch angle of the root and θ_1 is the blade twist.

As it was not shown previously on this analysis the figure that follows shows the twist throughout the blade from the root to the tip
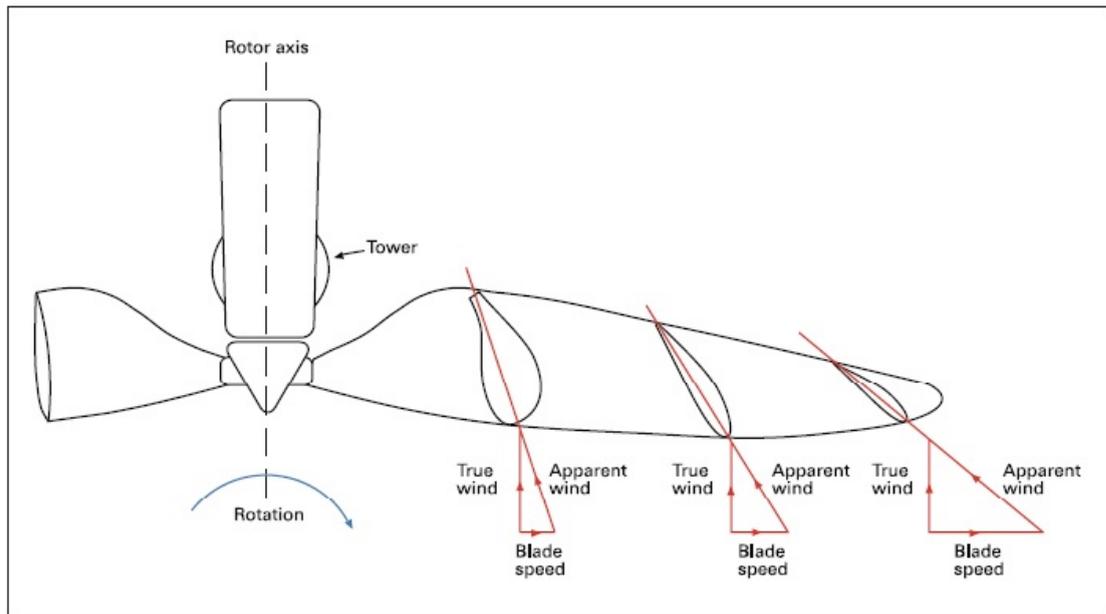
FIGURE 3.12: A representation of the "twist" throughout the length of the blade.

It is very important to bear in mind that, due to the high degree of simplification introduced by our theoretical model, the values of θ_0,θ_1 that provide a better correlation with the experimental results need not be the real ones, that is, the ones measured on the blade.

Therefore a sample blade could have the following values in θ_0,θ_1

$$\theta_0 = 0.37rad(21^o) \quad \theta_1 = -0.09rad(\approx -5^o)$$

Now as far as the $a, c_{d0}, c_{d1}, c_{d2}$ is concern, they can not be easily estimated with accuracy, unless of course wind tunnel tests are carried out on each blade and its own aerofoil. It also assumed earlier on the model that $a, c_{d0}, c_{d1}, c_{d2}$ are identical for every blade element. For this to be true, all the aerofoils must have an identical shape (identical camber and relative thickness). The blades of the AM also must have constant thickness, and the radius of curvature is constant both chordwise and spanwise. Since they are tapered, the relative thickness will not be identical for every blade element. However, since the thickness is very small ($\approx 1mm$), this can be ignored.

Hence, the hypothesis of identical $a, c_{d0}, c_{d1}, c_{d2}$ throughout the blade seems plausible.

Also it is true that most of the books about helicopter theory [10], [8] provide estimates for $a, c_{d0}, c_{d1}, c_{d2}$. Unfortunately, most of these estimates refer to blades of conventional, full-sized helicopters, which are quite different from those that will be used on the AM. The best solution to estimate $a, c_{d0}, c_{d1}, c_{d2}$ is probably as it was referred through experimental tests which could not take place for several reasons.

As far as the mean chord c is concern , its value will depend greatly on how will be defined. Again in conventional helicopters the blades are usually untapered and thus the mean chord is simply the chord of every section of the blade. However, as it was mentioned before, the blades of the AM are tapered. When the blades are tapered, a usual practice is to assume that the mean chord is the one of the blade element situated at 70 % of the blade span. The reason why this blade element is chosen is because it usually leads to accurate results, since it is quite representative of the whole blade. Following this convention, the mean chord will be $c \approx 0.02m$.

And that will be the way to obtain the value of the mean chord of the blade.

3.5.2 Empirical λ_{ind}

Is was stated in previous Section that it was found that an empirical correction of λ_{ind} predicted by the MMT is required. So from [16],

$$(\lambda_{ind})_{corrected} = \kappa\lambda_{ind} \tag{3.28}$$

3.5.3 Axial Flight

Introduction

At this point of the analysis focus will be given to the different states of Flights.

The begging will be with the case of the *Axial Flight* of the Aerial Manipulator ($a = \pm\frac{\pi}{2}$). The reason of starting the analysis with that state of the Flight is because it is simpler to study and that happens due to the existence of one variable λ_z rather than two (λ_z, μ).

As it was stated before in Axial Flight $a_{1s} = b_{1s} = 0$.

Therefore, follows the formulation of the Thrust force and the Torque at this state of Flight.

Thrust

In that way, taking the general expression of Thrust coefficient of the equation (3.11) and particularize it for the Axial Flight it results:

$$\frac{4C_T}{a\sigma} = \left(1 - \frac{e}{R}\right)\left(\frac{2}{3}\theta_0 + \frac{1}{2}\theta_1 - \lambda_z - \lambda_{ind}\right) \tag{3.29}$$

Notice here that in Axial Flight there is no horizontal speed vector thus,

$$\left.\begin{array}{l} V_x = 0 \\ \mu = \frac{V_x}{|\Omega|R} \end{array}\right\} \Rightarrow \mu = 0$$

As it is shown the thrust coefficient depends on a. It should remembered that the effective hinge offset e is estimated from the ω_{nr} but since in this analysis are were held no experimental tests to define it will be assumed that the joint offset (this time which is approximately equal to e) is 10% of the R. Which R is the radius of the rotor. Caution should be given so as not to confuse the radius of the rotor with the length of the blade as that length is approximately equal to 90% of R. That means that the Rotor disk radius (R) includes the hinge offset ($\approx$ joint offset).

Also taking the typical value for $a = 5.5$ given by Bramewll [10] (in hover) is a well sized approach for the computations.

Apart from these it should be remembered that the true meaning of λ_{ind} was given by Glauert's formula. According to this formula, λ_{ind} was the local induced velocity at

$\psi = \pm \frac{\pi}{2}$, non-dimensionalized with ΩR . But Glauert's formula is not valid in axial flight. Therefore it is much more convenient to choose κ for the correct results in C_T.

It was decided to use [16] the following:

$$\kappa = 1 - \kappa_0 \lambda_z \qquad (3.30)$$

With $\kappa_0 = 1.2$

The reason why also κ was introduced was to try to represent the so-called "pit" when $\lambda_z < 0$ and in that way in this region λ_{ind} to be increased. More about this note will also be mentioned later.

Taking these into consideration someone can conclude to the fact that in here is described a model that is valid in all Axial flight states (hovering, climb and descent). That model is based on Blade Element Theory (BET) and Modified Momentum Theory (MMT). According to experimental results this model fails when the spinning speed of the rotor is too low, probably due to the fact the blades are producing insufficient (or not the predicted one) induced flow. That maybe could be characterized as a limitation of this model.

Besides it is true that the rotors of the AM will be rarely operating in low speeds (might be in the take off or in landing states respectively) there will be no further approach and change at the present analysis.

Torque

Again for the Axial Flight according to the equation (3.37), when $\mu = 0$ it is produced:

$$\frac{4C_Q}{a\sigma} = \frac{1}{2}\frac{c_{d0}}{a} + \frac{c_{d1}}{a}\left[\frac{1}{2}\theta_0 + \frac{2}{5}\theta_1 + \frac{2}{3}(-\lambda_z - \lambda_{ind})\right] +$$
$$\frac{c_{d2}}{a}\left[\frac{1}{2}\theta_0^2 + \frac{1}{3}\theta_1^2 + (-\lambda_z - \lambda_{ind})^2 + \frac{4}{5}\theta_0\theta_1 + \frac{4}{3}\theta_0(-\lambda_z - \lambda_{ind}) + \theta_1(-\lambda_z - \lambda_{ind})\right] \qquad (3.31)$$

So in this phase of the analysis it is necessary to propose a method in order to approximately obtain the the coefficients c_{d1} and c_{d2} that are part of the above equation (3.31).

A way for someone to deal with this [16] is first to neglect the third term of the equation (3.31) that includes θ_0, θ_1, λ_z, λ_{ind}. This is not as arbitrary as it may seems since θ_0, θ_1, λ_z, λ_{ind} have values smaller than 1.

Then assuming that for $\lambda_z > 0.15$ it is $\frac{d\lambda_{ind}}{d\lambda_z} \approx 0$ so as to be obtained that:

$$\frac{dC_Q}{d\lambda_z} \approx -\frac{2}{3}\frac{\sigma}{4}c_{d1} \tag{3.32}$$

Based also on the experimental tests of the Reference [16] and using square method in order to approximate the curve of C_Q against λ_z, it is derived that $c_{d1} \approx 0.7$.

On the other hand, in order to estimate c_{d2} the equation (3.31) will be used again but without neglecting any terms this time. So deriving C_Q, it is produced:

$$\frac{dC_Q}{d\lambda_z} = \frac{\sigma}{4}\left\{-c_{d1}\frac{2}{3}\left(1 + \frac{d\lambda_{ind}}{d\lambda_z}\right) + c_{d2}\left[2(\lambda_z + \lambda_{ind})\left(1 + \frac{d\lambda_{ind}}{d\lambda_z}\right) - \left(\frac{4}{3}\theta_0 + \theta_1\right)\left(1 + \frac{d\lambda_{ind}}{d\lambda_z}\right)\right]\right\} \tag{3.33}$$

And deriving second time:

$$\frac{d^2C_Q}{d\lambda_z{}^2} = \frac{\sigma}{4}\left(-c_{d1}\frac{2}{3}\frac{d^2\lambda_{ind}}{d\lambda_z{}^2} + c_{d2}\left[2\left(1 + \frac{d\lambda_{ind}}{d\lambda_z}\right)^2 - \left(\frac{4}{3}\theta_0 + \theta_1\right)1 + \frac{d^2\lambda_{ind}}{d\lambda_z{}^2}\right]\right) \tag{3.34}$$

So after these considering in $\lambda_z = 0$ the $\frac{d\lambda_{ind}}{d\lambda_z} \approx -\frac{1}{2}$ and of course $\frac{d^2\lambda_{ind}}{d\lambda_z{}^2} \approx 0$. Notice that λ_{ind} can be calculated, when $\lambda_z = 0$, using BET-MMT. Also not to forget $\frac{d^2C_Q}{d\lambda_z{}^2} \approx 0$ when again $\lambda_z \approx 0$.

In that way entering all these inside the equations (3.34) it is obtained $c_{d2} \approx 0$

Also a common value for the coefficient c_{d0} is approximately again close to 0.05.

So the Table 3.1 summarize the values that were found above as follows:

Symbol	Value
θ_0	$0.37 rad(21^o)$
θ_1	$-0.09 rad(\approx -5^o)$
a	5.5 (for Hover state)
c_{d0}	0.05
c_{d1}	≈ 0.7
c_{d2}	≈ 0
c	Mean chord

TABLE 3.2: Values of the required parameters by the Rotor Model

Also, it was measured and tested that when a constant c_d is used correlation is good for positive λ_z but the trend is wrong when λ_z is negative. This is the reason why the drag polar was introduced. When used $c_d = 0.05 + 0.7a$ the correlation for negative λ_z is much better.

If a constant c_d is being used in the negative λ_z - it was tested that - the relative results in C_Q will not be accurate and in fact as the λ_z decreases the C_Q will decrease instead of rising. That is the reason why a more complex form c_d was needed. The three-term polar provides a good approximation, according to most of the literature ([8], [3]) and so it is the one used here.

Ground effect

Generally is commonly stated in the specialized literature ([8], [3]) that the ground effect in hover becomes noticeable when the altitude of the rotor above the ground is less than one rotor diameter. This also applies for axial flight, although in this case the problem is purely academic since a rotor moving vertically will soon be far from the ground.

That mentioned altitude, for the AM, of one rotor diameter is very low in absolute terms. It seems probable that the AM will rarely fly so close to the ground, except when taking off or landing. This is the main reason why the ground effect will not be incorporated in here.

3.5.4   Conclusion

To conclude, in this Section *Analysing the different states in Rotor Model* it seems that emphasis was given so as to illustrate the methodology of calculating the properties of a blade which corresponds to correct and justifiable selections. Also a reference was made to the empirical/experimental necessary "factor" κ and of course major time spend in analysing the rotor condition of the *Aerial Manipulator* during the Axial Flight. A state that as already was mentioned is characterized by its own simplicity and it is defined from the phases of hover, climb and descent.

Caution was given to the limitations of the chosen Model and solutions were proposed so as to overcome these problems. In that way, the reason of this Section was mainly to clarify to the reader the "path" of applying the proposed Model to the Analysis of the Flaying Manipulator.

Moreover it is important to underline that the proposed Model it is capable to be applied apart from the *Axial Flight* also to the *Forward Flight* state and with that

to produce a more "all-around" and complete analysis that incorporates almost every Flight phase. Naturally, based on the Model proposed above, with some changes and hypothesis Forward Flight can be stated.

Nevertheless, stating also the procedure of including the Forward Flight phase will require more time spent on the project and considering the time limitations will impel this diploma thesis out of the timetable restrictions. Besides more equipment and experimental tests would be necessary to accomplish this. Also, should be mentioned that proceeding to that incorporation will not reflect the main goals of this thesis. Thus, taking these into account Forward Flight will not be considered in here. That could be an option for some further and future research but these will be summarized later on this diploma.

Finally, as it was referred previously in this thesis the lack of experimental results and tests are a fact. From this position it would be difficult to identify vertical speeds V_z (see Blade Element Theory subsection 3.3.5) and tip speed ratio ΩR as these values vary from one position of the AM to another. Yet a whole wide range of values would be the proper so as to identify fully these quantities at a vast of AM's Flight positions. At this point apart from the fact that experimental tests could not take place, the identification of the quantities would be impossible also because AM does not have (yet) a "material subsistence" and so no real-time measures can be taken on a rotor.

These are the main reasons why the Analysis of the Flaying Manipulator will be limited only to the Axial Flight and more specifically to the hover state. That happens because in hover state there is no need of identifying the vertical speed V_z as in fact there the air-craft remains still, so $V_z = 0$ as it was mentioned earlier on the Model. Moreover as the Analysis of the AM is limited to the Axial Flight there will not exist horizontal movement and thus $V_x = 0$

It should remembered here that:

$\mu = \frac{V_x}{|\Omega|R}$ is the horizontal speed to tip speed ratio

$\lambda_z = \frac{V_z}{|\Omega|R}$ is the vertical speed to tip speed ratio

$\lambda_{ind} = \frac{V_{ind}}{|\Omega|R}$, where v_{ind} is the induced velocity

Therefore when $V_x = 0$ and $V_z = 0$ it is clearly also that $\mu = 0$ and $\lambda_z = 0$.

As referred for simplification reasons these two will be the flying conditions of the AM that will be studied in here.

Also these conditions will automatically affect the Thrust and Torque coefficients.

Thrust

Considering the equation (3.29) that is an expression of the Thrust coefficient, will be transformed in hover state as:

$$\frac{4C_T}{a\sigma} = \left(1 - \frac{e}{R}\right)\left(\frac{2}{3}\theta_0 + \frac{1}{2}\theta_1 - \lambda_{ind}\right) \tag{3.35}$$

which the term of λ_z was removed.

Torque

With the same way the equation (3.31) of the Torque coefficient, in hover state, will be:

$$\frac{4C_Q}{a\sigma} = \frac{1}{2}\frac{c_{d0}}{a} + \frac{c_{d1}}{a}\left[\frac{1}{2}\theta_0 + \frac{2}{5}\theta_1 + \frac{2}{3}(-\lambda_{ind})\right] +$$
$$\frac{c_{d2}}{a}\left[\frac{1}{2}\theta_0^2 + \frac{1}{3}\theta_1^2 + (-\lambda_{ind})^2 + \frac{4}{5}\theta_0\theta_1 + \frac{4}{3}\theta_0(-\lambda_{ind}) + \theta_1(-\lambda_{ind})\right] \tag{3.36}$$

where again here the term of λ_z has been removed.

Notice that all the coefficients of these two equations (6.4), (6.6) are being calculated with the same way as it was referred earlier in this Section. In fact someone should see the Table 3.2 that has the needed values for the quantities θ_0, θ_1, a, c_{d0}, c_{d1}, c_{d2}, c. Of course at the end the quantities of λ_{ind} and σ will be calculated with the methodology and with the equation that proposed respectively at the Model.

Finally, it should written that as the present Analysis focuses only in the Hover state of the AM there will no emphasis on the so-called "conning angle" (remember the definition in the Subsection 3.3.3). That happens due to the fact that in Hover State - as mentioned - μ and λ_z are equal to zero. So the equation (3.37), which for completeness reasons is stated again,

$$\frac{4C_Q}{a\sigma} = \frac{1}{2}\frac{c_{d0}}{a}\left(1+\mu^2\right)$$

$$+\frac{1}{1+\frac{3}{2}\mu^2}\frac{c_{d1}}{a}\left[\theta_0\left(\frac{1}{2}-\frac{19}{36}\mu^2+\frac{3}{4}\mu^4\right)+\theta_1\left(\frac{2}{5}-\frac{2}{5}\mu^2+\frac{\mu^4}{2}\right)+\left(-\lambda_z+\mu a_{1s}-\lambda_{ind}\right)\left(\frac{2}{3}-\frac{\mu^2}{3}\right)\right]$$

$$+\frac{1}{\left(1+\frac{3}{2}\mu^2\right)^2}\frac{c_{d2}}{a}\begin{bmatrix}\theta_0^2\left(\frac{1}{2}+\frac{2}{9}\mu^2-\frac{\mu^4}{24}+\frac{9}{8}\mu^6\right)+\\[4pt](-\lambda_z+\mu a_{1s}-\lambda_{ind})^2\left(1+2\mu^2+\frac{3}{4}\mu^4\right)+\\[4pt]\theta_0\theta_1\left(\frac{4}{5}+\frac{2}{5}\mu^2-\frac{\mu^4}{5}+\frac{3}{2}\mu^6\right)+\\[4pt]\theta_0(-\lambda_z+\mu a_{1s}-\lambda_{ind})\left(\frac{4}{3}+\frac{4}{3}\mu^2-\mu^4\right)+\\[4pt]\theta_1(-\lambda_z+\mu a_{1s}-\lambda_{ind})\left(1+\mu^2-\frac{3}{4}\mu^4\right)\end{bmatrix}$$

$$+\frac{\mu^2}{\left(1+\frac{1}{2}\mu^2\right)^2}\frac{c_{d2}}{a}\left[a_0^2\left(\frac{1}{18}+\frac{\mu^2}{6}-\frac{\mu^4}{8}\right)+\lambda_{ind}^2\left(\frac{1}{8}+\frac{\mu^2}{16}\right)+a_0\lambda_{ind}\left(\frac{\mu}{3}+\frac{\mu^3}{6}\right)\right]$$

$$(3.37)$$

when these conditions are taken into this equation there will be no term that incorporates the "conning angle" a_0.

Not to mention also that Bramwell [10], a_0 has very little importance in comparison to the values of a_1, b_1 that where defined to model (see again the Subsection 3.3.3). Besides the fact that also these angles in Axial Flight are also equal to zero.

3.6 The aerodynamic effect on the airframe

Generally, here in Flaying Manipulator obtaining a model that encapsulates the aerodynamic forces and moments that affect on the airframe of the structure exerted by the airflow around it could be a difficult task.

Not to mention that the strict formation of the frame of the AM is not defined yet as the position and the direction of each "thruster" it is - above all - between the main targets of this diploma thesis. So, after the definition of those two follows the exact formation of the frame that could be also complicated.

Thus, estimating theoretically the forces and moments that affect the airframe is impractical unless CFD software is used. The airflow over the airframe is very complicated, with boundary layer separation in many areas. This could be even worsen by the complex geometry the airframe of the AM might have.

Hence, besides CFD, the only solution is to carry out wind tunnel tests. But this is also problematic and apart from that, this procedure it is not strictly included to the goals of this analysis.

To conclude, for simplification reasons that probably are not so arbitrary (see the Reference in the quadrotor [16]), the forces and the moments exerted be the airflow around the airframe will be considered approximately equal to zero with no further reference upon this throughout the analysis.

Chapter 4

Technical Problem Statement

4.1   General

At this particular part of the thesis will be held an extensive description on the problem that it is tried to be solved, its own applicable equations and requirements as well as its own restrictions.

4.2   Description of the object

As it mentioned previously the overall description of this "flying object" held on the idea of considering a structure composed of a set number n thrusters and an end-effector with reference to which the whole analysis is made on. The strict shape of the structure is not defined yet and clearly is not the main goal of this diploma as it is the product of the analysis. In that way the whole thesis will be based on the static equations and on the static model that will be composed. This structure also is assumed to have an interaction with the environment throughout the end-effector, producing corresponding actuating force (F_{act}) and torque (M_{act}).

FIGURE 4.1: This figure shows an approximation of the Aerial Manipulator's structure as resulted from a solution in a Chapter that follow.

4.3 Principles of the problem

To begin with, think of vectors $\mathbf{r_i} \in \Re^3$ with $i = 1, \ldots, n$ that are showing the positions of thrusters with reference to the "exodus" of the structure, called "end-effector". Their orientations are defined in that way throughout a unit vector $\hat{\mathbf{F}}_i \in \Re^3$ with again $i = 1, \ldots, n$.

Taking these into consideration, someone can designate the "propulsion effort" as scalar λ_i of each of these thrusters, through the propulsion vector which is given by $\lambda_i \cdot \hat{\mathbf{F}}_i$. In order to understand the meaning of propulsion effort could think it as a kind of percentage of the entire Thrust force of each thruster that is necessary to be produced.

At this point is important to state that every one of these n thrusters are exactly the same. Having that way same length of blades (chord), same and equal design parameters and generally having the same air foil. Also it is crucial all the aerodynamic parameters of each thruster to be the same so as equal and same type of thrust,lift and drag force to be produced. Of course the motors that will rotate each blade of the structure (n in number) must have exactly the same characteristics with each other.

4.3.1 Forces

As far as the forces is concern, there are two kinds of them:

- the thrust force $\hat{\mathbf{F}}_\mathbf{i}$ and more specifically $\lambda_i \cdot \hat{\mathbf{F}}_\mathbf{i}$ accompanied-as it is shown- by the propulsion effort λ_i.

- and the force that is produced from the gravitational field of the earth to the structure (weight of structure).

So the whole force that is transmitted essentially through the end-effector is

$$\sum_{i=1}^{n}(\lambda_i \cdot \hat{\mathbf{F}}_\mathbf{i}) + \mathbf{W} = \mathbf{F}_\mathbf{act} \tag{4.1}$$

where $\mathbf{F}_\mathbf{act} \in \Re^3$ is the corresponding actuation force vector of the end-effector and $\mathbf{W} \in \Re^3$ is the vector of the weight of the Aerial manipulator. Keep in mind that the overall weight of the structure is a result of the weights w of the individual thrusters - assumed equal since they have same characteristics as it mentioned above - and the result of the weight w_s of the rest of the structure. In that weight w_s should be incorporated some of the below:

1. The weight of the possible material that will be used for the frame (i.e. arms)

2. Battery

3. CFRP plate

4. Arm brackets

5. Electronics Plate

6. Receiver

7. Gears

8. Blades

9. Motors

10. Other possible elements (Blade joints,screws, end-effector weight)

Consequently the weight of the whole body of the Aerial manipulator is written as:

$$\mathbf{W} = n \cdot \mathbf{w} + \mathbf{w}_\mathbf{s} \tag{4.2}$$

where here the $\mathbf{w}$ and $\mathbf{w}_\mathbf{s}$ are the vectors that correspond to the weights of each thruster and the weight of the rest structure. Both vectors belong to $\Re^3$

Thus with simple replacement using both (4.1) and (4.2) it is exported:

$$\sum_{i=1}^{n} (\lambda_i \cdot \hat{\mathbf{F}}_{\mathbf{i}}) + n \cdot \mathbf{w} + \mathbf{w}_{\mathbf{s}} = \mathbf{F}_{\mathbf{act}} \qquad (4.3)$$

Then, with the aim of modifying the 4.3 equation, the "effort" can be written as:

$$\lambda = \begin{bmatrix} \lambda_1 & \lambda_2 & \dots & \lambda_n \end{bmatrix}^T$$

And apart from this, the orientation unit vectors can also be have the form of a matrix:

$$\hat{F} \triangleq \begin{bmatrix} \hat{F}_1 & \hat{F}_2 & \dots & \hat{F}_n \end{bmatrix} \in \Re^{3xn}$$

which matrix will be called "thruster direction allocation matrix".

So the 4.3 is changed into:

$$\hat{F} \cdot \lambda + n \cdot \mathbf{w} + \mathbf{w}_{\mathbf{s}} = \mathbf{F}_{\mathbf{act}} \qquad (4.4)$$

where $F_{act} \in \Re^3$.

4.3.2 Torques

Torques as well as forces can also be separated into two types:

- those torques that are defined as the cross product of a specific distance and force

- and those torque that are known in helicopter's bibliography as "reaction-type torques" or "anti- torques".

The first type of torque is the well known cross product of a lever-arm distance and a force, which tend to produce rotation. In symbolic way torque is $\tau = \mathbf{r} \times \mathbf{F}$ and in this analysis as it is written there are two types of forces the thrust forces and the general weights of the structure. So there are two cross products:

- Cross product concerning thrust forces accompanied by the "propulsion effort" λ_i and in symbolic way written:
$$\mathbf{r}_{\mathbf{i}} \times (\lambda_i \cdot \hat{\mathbf{F}}_{\mathbf{i}})$$

- Cross product concerning the weights of the whole body. Since it not determined yet it is assumed as R_G the position of the the center of gravity G of the Aerial Manipulator with reference again to the end-effector. In symbolic way:

$$\mathbf{R_G} \times \mathbf{W}$$

Somebody should notice that since the whole weight of the structure is divided into two (weight of the individual thruster/weight of the rest structure) the vector that corresponds to the position of the center of gravity ($\mathbf{R_G}$) should do the same. Thus by the center gravity definition:

$$\mathbf{R_G} \times \mathbf{W} = -\mathbf{W} \times \mathbf{R_G} \triangleq - \left(\sum_{i=1}^{n} \mathbf{w} \times \mathbf{r_i} \right) - \mathbf{w_s} \times \mathbf{r_s}$$

Concerning the second type of torque think of when a rotor turns into one direction, the body of the structure (Aerial manipulator) tends to rotate in the opposite direction according to Newton's third law. This is known as *reaction torque*. This torque symbolic will be shown with $\tau_\mathbf{i} \in \Re^3$, which is co-linear to $\hat{\mathbf{F}}_\mathbf{i}$ vector of every thruster of the system. So, it is reasonable to assume that:

$$\tau_\mathbf{i} = \mu \cdot (\lambda \cdot \hat{\mathbf{F}}_\mathbf{i})$$

where μ is assumed to be a coefficient that represents the relationship between the force from the thruster and the reaction torque or anti-torque.

Therefore the whole torque that is transmitted essentially through the end-effector comes from summarizing these three torques and its equation is as follows:

$$\sum_{i=1}^{n} \left[\mathbf{r_i} \times (\lambda_i \cdot \hat{\mathbf{F}}_\mathbf{i}) + \mu \cdot (\lambda_i \cdot \hat{\mathbf{F}}_\mathbf{i}) \right] + \mathbf{R_G} \times \mathbf{W} = \mathbf{M_{act}} \qquad (4.5)$$

Recalling the fact that both vector of the position of the gravity center and vector of the weights are divided into two, the equation (4.5) can also be transformed into:

$$\sum_{i=1}^{n} \left[\mathbf{r_i} \times (\lambda_i \cdot \hat{\mathbf{F}}_\mathbf{i}) + \mu \cdot (\lambda_i \cdot \hat{\mathbf{F}}_\mathbf{i}) \right] + \left(\sum_{i=1}^{n} \mathbf{r_i} \right) \times \mathbf{w} + \mathbf{r_s} \times \mathbf{w_s} = \mathbf{M_{act}} \qquad (4.6)$$

4.3.3 End-Effector

At this point is essential to be reminded that the cross product of two vectors ($\mathbf{A}$ and $\mathbf{B}$) can be written in matrix form as follows:

$$\mathbf{A} \times \mathbf{B} = S(A) \cdot B = -S(B) \cdot A$$

where here S is the *skew-symmetric* matrix which is a anti-symmetric and singular (non invertible) matrix. This matrix S for a random vector $\mathbf{A}$ written in a matrix form $A = \begin{bmatrix} A_x & A_y & A_z \end{bmatrix}^T$ has the form:

$$S(A) = \begin{bmatrix} 0 & -A_z & A_y \\ A_z & 0 & -A_x \\ -A_z & A_x & 0 \end{bmatrix}$$

So the cross product of the position of each thruster ($\mathbf{r_i}$) with the thrust force ($\hat{\mathbf{F}}_\mathbf{i}$) accompanied by the propulsion effort λ_i is :

$$\mathbf{r_i} \times (\lambda_i \cdot \hat{\mathbf{F}}_\mathbf{i}) = \lambda_i \cdot (\mathbf{r_i} \times \hat{\mathbf{F}}_\mathbf{i}) = \lambda \cdot S(r_i) \cdot \hat{F}_i$$

transforming then the equation (4.6) to

$$\sum_{i=1}^{n} \left[\lambda_i \cdot S(r_i) \cdot \hat{F}_i + \mu \cdot (\lambda_i \cdot \hat{\mathbf{F}}_\mathbf{i}) \right] + \left(\sum_{i=1}^{n} \mathbf{r_i} \right) \times \mathbf{w} + \mathbf{r_s} \times \mathbf{w_s} = \mathbf{M_{act}}$$

and finally having the equation

$$\sum_{i=1}^{n} \left[\lambda_i \cdot S(r_i) \cdot \hat{F}_i \right] + \mu \cdot \sum_{i=1}^{n} \left(\lambda_i \cdot \hat{\mathbf{F}}_\mathbf{i} \right) + \left(\sum_{i=1}^{n} \mathbf{r_i} \right) \times \mathbf{w} + \mathbf{r_s} \times \mathbf{w_s} = \mathbf{M_{act}} \qquad (4.7)$$

Moreover combining the (4.4) with the last equation of the actuating torque (4.7) it is produced:

$$\sum_{i=1}^{n} \left[\lambda_i \cdot S(r_i) \cdot \hat{F}_i \right] + \mu \cdot (\mathbf{F_{act}} - n \cdot \mathbf{w} - \mathbf{w_s}) + \left(\sum_{i=1}^{n} \mathbf{r_i} \right) \times \mathbf{w} + \mathbf{r_s} \times \mathbf{w_s} = \mathbf{M_{act}}$$

and as it is already mentioned

$$\lambda = \begin{bmatrix} \lambda_1 & \lambda_2 & \dots & \lambda_n \end{bmatrix}^T$$

then it is produced:

$$E(r, \hat{F}) \cdot \lambda = \mathbf{M_{act}} - \mu \cdot (\mathbf{F_{act}} - n \cdot \mathbf{w} - \mathbf{w_s}) - \left(\sum_{i=1}^{n} \mathbf{r_i} \right) \times \mathbf{w} - \mathbf{r_s} \times \mathbf{w_s}$$

where here

$$r \triangleq \begin{bmatrix} r_1 & r_2 & \dots & r_n \end{bmatrix}^T \in \Re^{3 \times n}$$

$$\hat{F} \triangleq \begin{bmatrix} \hat{F}_1 & \hat{F}_2 & \dots & \hat{F}_n \end{bmatrix} \in \Re^{3 \times n} \qquad (4.8)$$

$$E(r, \hat{F}) \triangleq \begin{bmatrix} S(r_1) \cdot \hat{F}_1 & S(r_2) \cdot \hat{F}_2 & \dots & S(r_n) \cdot \hat{F}_n \end{bmatrix}^T \in \Re^{3 \times n}$$

Suppose now that the number n of thrusters and their associated positions r_i with $i = 1, \dots, n$ are known quantities the equations (4.4) and (4.7) can be re-written:

$$\left. \begin{aligned} \hat{F} \cdot \lambda + n \cdot \mathbf{w} + \mathbf{w_s} &= \mathbf{F_{act}} \\ E(r, \hat{F}) \cdot \lambda &= \mathbf{M_{act}} - \mu \cdot (\mathbf{F_{act}} - n \cdot \mathbf{w} - \mathbf{w_s}) - (\sum_{i=1}^{n} \mathbf{r_i}) \times \mathbf{w} - \mathbf{r_s} \times \mathbf{w_s} \end{aligned} \right\} \qquad (4.9)$$

And from the system (4.9) it is implied

$$D(r, \hat{F}) \cdot \lambda = \mathbf{W_R} \qquad (4.10)$$

where

$$\mathbf{W_R} \hat{=} \begin{bmatrix} \mathbf{F_{act}} - n \cdot \mathbf{w} - \mathbf{w_s} \\ \\ \mathbf{M_{act}} - \mu \cdot (\mathbf{F_{act}} - n \cdot \mathbf{w} - \mathbf{w_s}) - (\sum_{i=1}^{n} \mathbf{r_i}) \times \mathbf{w} - \mathbf{r_s} \times \mathbf{w_s} \end{bmatrix} \in \Re^6 \qquad (4.11)$$

will be called "augmented wrench vector" on account of the need to represent for a example a force-torque vector that incorporates the effects of the gravity. Also the

$$D(r, \hat{F}) \triangleq \begin{bmatrix} \hat{F} \\ E(r, \hat{F}) \end{bmatrix} \in \Re^{6 \times n} \qquad (4.12)$$

will be called the "Thruster-to-End Effector " matrix.

4.4 Unidirectional fans

First and foremost it is crucial at this section to refer to some of the assumptions that were made.

As it is referred to the previous Chapter, this diploma analysis will be based on the assumption that each rotor of the structure works on the so-called "normal state". There is also another "unrealistic" solutions of the equations which include descent rates from hover through to the windmill brake condition, thus encompassing the so-called ideal autorotation condition when the inflow equals the descent rate. This region includes the vortex-ring condition where the wake beneath the rotor becomes entrained in the air moving upwards relative to the rotor outside the wake and, in turn, becoming part of the inflow above the rotor again. This circulating flow forms a toroidal-shaped vortex which has a very non-uniform and unsteady character, leading to large areas of high inflow in the centre of the disc and stall outboard. The vortex-ring condition is not amenable to modelling via momentum considerations alone. However, there is evidence that the mean inflow at the rotor can be approximated by a semi-empirical shaping function that will be taken into consideration at this analysis.

Nevertheless somebody should not forget the *Inertia phenomenon* - the resistance of any physical object to any change in its state of motion, including a change in direction - that one rotor has during its own operation. In that way it is practically impossible for the rotor in t^- time to rotate in one direction and in t^+ time expected to rotate the exactly opposite one.

In this manner in order to reconnect the analysis that made concerning the *Forces* and the *Torques* somebody should notice that when solving the equation (4.10), the matrix that corresponds to the propulsion effort λ can obtain any value in $\Re^6$. Including thus both negative and positive values for its component λ_i of the matrix. But as it referred fans are somehow "optimally" designed to rotate on a certain direction and there is no swashplate to change the angle of attack. So the thrust force, which is the product of this rotation, will have a certain direction as well, which will assume to correspond to positive values for λ_i.

Special attention in that way should be given on the appearance of such a negative λ_i which will oblige the ith rotor to operate the opposite direction from the optimal one. In order to alleviate this problem one could think of an "engineering" solution such as adopting appropriate fans of variable geometry.

On the other hand in this analysis it is adopted a conservative solution and a solution more likely to respond on any general case of this negative λ_i. This solution based on the idea of introducing *one and only one* additional fan to the whole structure of body.

Beginning with reporting the approach that has been followed, first equation (4.10) should be re-written as:

$$\sum_{i=1}^{n} \lambda_i \cdot t_i = \mathbf{W_R} \tag{4.13}$$

So supposing that $D(r, \hat{F}) = \begin{bmatrix} t_1 & t_2 & \dots & t_n \end{bmatrix}, t_i \in \Re^6$ defined in (4.12), then it is observed that

$$t_i = \begin{bmatrix} \hat{F}_i \\ S(r_i) \cdot \hat{F}_i \end{bmatrix}$$

At this point will be introduced - in matrix form - the "assistive vector" that corresponds to the one and only one additional fan of this approach:

$$t_a \triangleq -\sum_{i=1}^{n} t_i = -\sum_{i=1}^{n} \begin{bmatrix} \hat{F}_i \\ S(r_i) \cdot \hat{F}_i \end{bmatrix} = \begin{bmatrix} -\sum_{i=1}^{n} \hat{F}_i \\ -\sum_{i=1}^{n} (S(r_i) \cdot \hat{F}_i) \end{bmatrix} \triangleq \begin{bmatrix} \hat{F}_a \\ S(r_a) \cdot \hat{F}_a \end{bmatrix} \tag{4.14}$$

Having these equalities and these definitions that are shown above (4.14) the corresponding direction of the one extra fan is:

$$\mathbf{\hat{F}_a} = -\sum_{i=1}^{n} \mathbf{\hat{F}_i} \tag{4.15}$$

Also as it shown from (4.14) the position vector of the assistive fan should satisfy the following matrix equation:

$$-\sum_{i=1}^{n} S(r_i) \cdot \hat{F}_i = S(r_a) \cdot \hat{F}_a \Rightarrow +\sum_{i=1}^{n} S(\hat{F}_i) \cdot r_i = -S(\hat{F}_a) \cdot r_a \tag{4.16}$$

Which leads to:

$$S(\hat{F}_a) \cdot r_a = -\sum_{i=1}^{n} S(\hat{F}_i) \cdot r_i \tag{4.17}$$

As it is mentioned already skew-symmetric matrix S is singular matrix, so its determinant equals to zero. Thus in the first part of the (4.17) left to the equality

$$det \left[S(\hat{F}_a) \right] = 0$$

So r_a is obliged to satisfy the (4.17), which is an equation of the type

$$A \cdot x = b$$

where A , b are matrix that can be calculated and more specifically are

$$A = S(\hat{F}_a)$$

and

$$b = - \sum_{i=1}^{n} S(\hat{F}_i) \cdot r_i$$

.

In fact as it is exported from this procedure someone has to find the solution of the system $A \cdot x = b$ in order to find the position of the added assistive fan. In this system also A is an anti-symmetric and singular matrix , which leads to the fact the there two possibilities:

- feasible solution to exist

- or this specific system will not have any solutions

In order the system to have a solution must:

$$rank \left[A \right] = rank \left[A \quad b \right] \tag{4.18}$$

otherwise if this equality will not be verified there will not exist any solution to the system. However this subject it remains to be handled later on this diploma thesis. Notice that the part right to the equality sign defines the rank of the augmented matrix consisting of A, b matrices.

4.4.1   A single negative "effort coefficient"

Reconnecting the structure of the analysis concerning the method of alleviating the appearance of negative "effort coefficient".

The first case that will be introduced will handle the existence of only one negative "effort coefficient" λ_i.

So assume at this point that, solving the equation (4.10) results to $\lambda_k < 0$, for some k. Thus (4.13) can be re-written as:

$$\sum_{i=1,i\neq k}^{n} (\lambda_i \cdot t_i) + \lambda_k \cdot t_k = \sum_{i=1,i\neq k}^{n} (\lambda_i \cdot t_i) + (-\lambda_k) \cdot (-t_k) = W_R \qquad (4.19)$$

where W_R is in matrix form.

Notice that although in this case that is studied all coefficients are non-negative a new issue is arise and that is how to "generate" $-t_k$.

From the (4.14) can be exported:

$$t_a \triangleq -\sum_{i=1}^{n} t_i = -\sum_{i=1,i\neq k}^{n} (t_i) - t_k \Rightarrow -t_k = t_a + \sum_{i=1,i\neq k}^{n} (t_i) \qquad (4.20)$$

In that way by using (4.20) the equation (4.19) can be expressed as:

$$W_R = \sum_{i=1,i\neq k}^{n} \lambda_i \cdot t_i + (-\lambda_k) \cdot \left(t_a + \sum_{i=1,i\neq k}^{n} (t_i) \right) = \sum_{i=1,i\neq k}^{n} (\lambda_i - \lambda_k) \cdot t_i + (-\lambda_k) \cdot t_a \quad (4.21)$$

Concluding, the last equation (4.21) exposes the way to incorporate the effect of the negative "effort-coefficient" and transforming the beginning equation (6.17), into a new one that can handle this negativity.

4.4.2 Multiple negative "effort coefficients"

On the other side here will be revealed a way to alleviate this time the appearance of multiple negative "effort coefficients".

Again at this second case assume that solving the equation (4.10) results to $\lambda_k < 0$ for $k \in \sigma_N$. Which now σ_N is the set of negative coefficients.

Therefore having the same approach with the one that was followed in *A single negative "effort coefficient"* section, someone can re-write (4.10) as:

$$\sum_{i\notin\sigma_N}\lambda_i\cdot t_i + \sum_{j\in\sigma_N}(-\lambda_j)\cdot(-t_j) = W_R \tag{4.22}$$

Where in this equation (4.22) it is tried to separate the negative coefficients from the non negative.

At this point of the thesis by assuming that $t_p = \left[t_{i\notin\sigma_N}\right] \in \Re^{6\times p}$ is a matrix that its columns consist of the non-negative coefficients and that explains the p indicator which corresponds to *positive* and assuming that $t_n = \left[t_{i\in\sigma_N}\right] \in \Re^{6\times n}$ is a matrix respectively that its columns correspond to the negative coefficients, then (4.22) is:

$$t_p\cdot\lambda_p + t_n\cdot\lambda_n = W_R \tag{4.23}$$

where respectively as shown above:

$$\lambda_p = \left[\lambda_{i\notin\sigma_N}\right]$$

is the matrix that corresponds to *positive effort coefficients* and of course

$$\lambda_n = \left[\lambda_{\in\sigma_N}\right]$$

is the matrix that corresponds to *negative effort coefficients*.

Again in order to alleviate this negative signs the equation of W_R can be re-written as:

$$W_R = t_p\cdot\lambda_p + (-t_n)\cdot(-\lambda_n) \tag{4.24}$$

From the definition of the equation (4.14) someone can easily deduce - in matrix form - that:

$$t_a \triangleq -\sum_{i=1}^{n} t_i \Rightarrow t_a = -(t_n + t_p) \tag{4.25}$$

Which clearly denotes that summarizing t_n and t_p corresponds to the whole number of the thrusters $(i = 1\ldots n)$.

Notice that arising issue of generating $-t_n$ in (4.24) will confronted by re-forming the (4.25) as follows:

$$-t_n = t_p + t_a \tag{4.26}$$

Finally incorporating that equation (4.26) into the (4.24) will give the total matrix form of the W_R as:

$$W_R = t_p \cdot \lambda_p + (-t_n) \cdot (-\lambda_n) = t_p \cdot \lambda_p + (t_p + t_a) \cdot (-\lambda_n) = t_p \cdot (\lambda_p - \lambda_n) + t_a \cdot (-\lambda_n) \tag{4.27}$$

4.4.3 Summarize

Closing the section of the *Unidirectional Fans* someone should realise that emphasis was given on some difficulties that arise during the effort of building the whole analysis and writing the equations that define the "Aerial Manipulator" . The manufacture limitations is one of them and these exist due to the "normal state" that rotors/fans have to work on and the *Lows of Physics* which oblige each fan to work on a specific way with aiming always the optimal outcome. Which in this case is translated into the necessity of each fan to rotate with *one* and specific direction, the "the optimal". An other issue that has been confronted is the appearance of negative "effort coefficient(s)" λ, when solving the equation (4.10), which issue handled for two cases with One and with Multiple negative coefficient(s) respectively.

So in this point is important to notice that the above proposed implementation of F_{act}, M_{act}:

- uses only *unidirectional* fans since all "effort coefficients" in equations (4.21) and (4.27) are positive

- results in operating fans in (considerably) higher efforts compared to if they were omnidirectional, and

- makes necessary to incorporate the additional fan, the *assistive* one, which symbolic is written by a

Rewriting then equation (6.17):

$$\mathbf{W_R} \hat{=} \left[\begin{array}{c} \mathbf{F_{act}} - (n+1) \cdot \mathbf{w} - \mathbf{w_s} \\ \\ \mathbf{M_{act}} - \mu \cdot (\mathbf{F_{act}} - (n+1) \cdot \mathbf{w} - \mathbf{w_s}) - \left(\sum_{i=1}^{n} \mathbf{r_i} \right) \times \mathbf{w} - \mathbf{r_s} \times \mathbf{w_s} \end{array} \right] \in \Re^6$$

$$(4.28)$$

where here after n is added the "assistive fan".

4.5 Aerodynamic fan interaction

At this point it is crucial to mention the major impact that aerodynamic interaction has between operating rotors/fans of the Aerial Manipulator. If the thrusters are located close to each other then the referred interaction could complicate and result to bad quality operational rotors. Thus, considering information referred to previous chapter, for every fan, assumed a "tube" around it so as to encapsulate a volume that should not intersect with the corresponding volumes of other fans, avoiding that way aerodynamic interactions.

The description of that "tube" at this phase of the analysis will be assumed cylindrical. In fact that assumption is made for simplification reasons. Notice that this thesis will incorporate the aerodynamic effects into the analysis. Besides incorporating other type of "tubes" or models that reflect the aerodynamic effects of the flow throughout the rotor plate relies on the approach (or accuracy) that someone willing to introduce according of its own uses.

So that description of such a cylindrical "tube", when expressed to its own, body-attached, coordinates frame $O'x'y'z'$, is given by

$$\begin{aligned} -x_r \leq x' \leq x_f \\ y'^2 + z'^2 \leq \rho^2 \end{aligned}$$

$$(4.29)$$

where ρ is related to (approximately at the order of, but certainly greater than) the fan radius and x_f, x_r are parameters depicting the aerodynamic effects *front* and *rear* respectively to the rotor plate.

Notice that when this description is required in term of the base (world) coordinate frame $Oxyz$ which is related to the body-attached, coordinate frame $O'x'y'z'$ via the translation and orientation vectors $(r_i, \hat{F}_i)$. That world frame should corresponds to the position of the "end - effector" of that structure. In other words, $O'x'y'z'$ is the frame

attached to the ith fan which is directed towards the direction pointed by the vector $\mathbf{F}_i$ and is located at $\mathbf{r}_i$. To achieve this, providing the description of an arbitrary point

$$p = \begin{bmatrix} x & y & z \end{bmatrix}^T$$

expressed in $Oxyz$ in terms of its description

$$p' = \begin{bmatrix} x' & y' & z' \end{bmatrix}^T$$

expressed in $O'x'y'z'$, related by an equation of the form that follows:

$$p = T_O^{O'}(\mathbf{r_i}, \hat{\mathbf{F}_i}) \cdot p' \tag{4.30}$$

where $T_O^{O'}(\mathbf{r_i}, \hat{\mathbf{F}_i})$ is the appropriate frame-transformation corresponding to the translation and orientation vectors $(\mathbf{r_i}, \hat{\mathbf{F}_i})$.
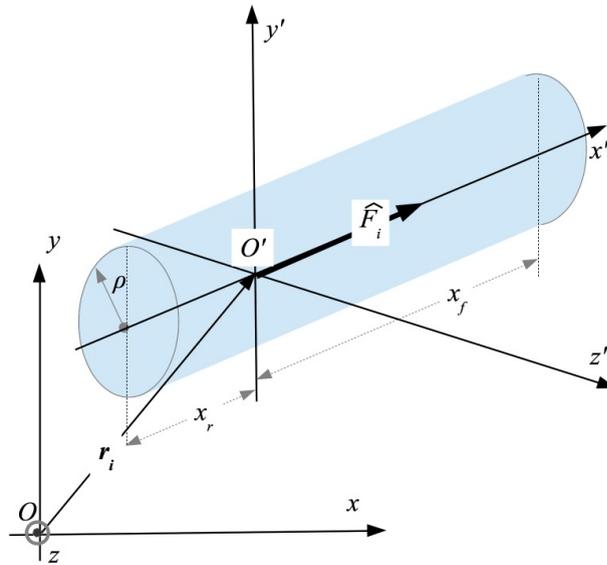
FIGURE 4.2: Transformation from world frame to cylinder coordinates.

In that way the constraints (5.4) that depict the volume of the cylindrical tube in space, if combined with the (4.30), are recast to a set of constraints of the matrix form:

$$G(r_i, \hat{F}_i, p) \leq 0 \tag{4.31}$$

One can define and evaluate the distance between two such tubes i, j via an optimization problem of the form:

$$d_{ij}(r_i, \hat{F}_i, r_j, \hat{F}_j) = \min_{p_i, p_j} \|p_i - p_j\|$$

$$s.t. \qquad\qquad G(r_i, \hat{F}_i, p_i) \leq 0 \qquad\qquad (4.32)$$

$$G(r_i, \hat{F}_i, p_j) \leq 0$$

4.6 Design problem

This section will introduce to the reader the *Design problem* that incorporates the overall analysis of this diploma thesis.

In that way given a particular structure defined be the matrices $(r, \hat{F})$, for a set of required actuation force and torque (F_{act}, M_{act}) its obvious that is necessary, the associates levels of effort λ_i, to be found. The solution that is used based on the following approach:

Since $W_R \in \Re^6$, to be able to have a solution of the equation (4.10) for

$$\lambda = \begin{bmatrix} \lambda_1 & \lambda_2 & \dots & \lambda_n \end{bmatrix}^T$$

, it is needed:

- $n \geq 6$ and

- the matrix $D(r, \hat{F})$ be of an appropriate rank, $rank\left[D(r, \hat{F})\right] = 6$

The rank condition of the matrix $D(r, \hat{F})$ is adequate from a strict mathematical perspective but, from a practical point of view, since the equation (4.10) leads to the "thruster efforts" values

$$\lambda = \begin{bmatrix} \lambda_1 & \lambda_2 & \dots & \lambda_n \end{bmatrix}^T$$

the sought solutions should not be "very sensitive" to small parametric uncertainties and deviations. This is partially achieved if, instead of using the rank condition, it is used the "condition number" concept:

$$\kappa(D) = \frac{\sigma_{max}(D)}{\sigma_{min}(D)}$$

where here $\sigma_{max}(D)$, $\sigma_{min}(D)$ are the maximum and minimum singular values of $D(r, \hat{F})$ and require that the condition number is not large (ideally should be close to one "1").

Thus it is required:

$$\kappa(D) \le K$$

where K is appropriate bound to be tuned.

Notice in here that if $rank\left[D(r, \hat{F})\right] = 6$ automatically means that the singular values of the matrix will not be equal to zero, so the condition number will not take undefined values such as zero and infinity, but this is a prerequisite so as the equation (4.10) to have *one and only one solution.* On other hand though, a subject arises when singular values are close to zero because then although the condition number will take feasible values (not equal to zero/infinity), the $D(r, \hat{F})$ might be badly ranked and the the determinate close to zero $(det\left[D(r, \hat{F})\right] \approx 0)$. Here, as shown, emphasis is given to the condition number because this will be the constraint incorporated the Design problem. Thus, apart from this an other constraint should be introduced that concerns the singular values.

More specifically, $\sigma(D) \ge \epsilon_1$, where ϵ_1 depends on the user but a well defined value would be $\epsilon_1 \approx 0.01$

Furthermore, as it is referred, to avoid fan interaction an other constraint is:

$$d_{ij}(\mathbf{r_i}, \hat{\mathbf{F}_i}, \mathbf{r_j}, \hat{\mathbf{F}_j} \ge \epsilon_2 > 0 \quad \forall i, j = 1, 2, \ldots, n, \alpha$$

which shows that all fans, including the "assistive" one, are non-interacting with each other. Notice though, that for the case where on from the "tubes" corresponds to the "assistive fan" then the corresponding position and direction vectors are not free (decision variables) since they depend on, through the equations (4.17),(4.15) (showing respectively the position and the direction vectors of the "assistive fan"), the position and direction vectors of all other fans.

We also need some form of optimization to allow the system to be as small (volume-wise) as possible:

$$\min_{r, \hat{F}} J(r)$$

where for r defined earlier as the matrix encapsulating the position vectors, is chosen to be a simple form of the norm:

$$J(r) = \|r\|_2$$

Taking all these into consideration the design problem is essentially recast to the following optimization problem:

$$\min_{r,\hat{F}} J(r)$$

$$s.t. \quad \sigma(D) \geq \epsilon_1 > 0$$

$$d_{ij}(\mathbf{r_i}, \hat{\mathbf{F}_i}, \mathbf{r_j}, \hat{\mathbf{F}_j}) \geq \epsilon_2 > 0 \quad \forall i, j = 1, 2, \ldots, n, \alpha$$

$$\hat{\mathbf{F}}_{\mathbf{a}} = -\sum_{i=1}^{n} \hat{\mathbf{F}_i} \tag{4.33}$$

$$S(\hat{F}_a) \cdot r_a = -\sum_{i=1}^{n} S(\hat{F}_i) \cdot r_i$$

$$\kappa(D(r, \hat{F})) \leq K$$

If a solution is found throughout the above optimization problem (4.33) that defines the Design problem then "general optimal" values that correspond to matrix form $(r_i^*, \hat{F}_i^*)$ for $i = 1, 2, \ldots, 6$ are derived, describing the optimal position and orientation of the fans, including of course those of the "assistive one". In that way throughout the equations (4.8), (4.12), (4.28) all terms $D(r^*, \hat{F}^*), W_R(r^*)$ of the (4.10) are obtained and thus someone can proceed towards obtaining the effort coefficient λ, with each time an actuation force and torque (F_{act}, M_{act}) is required.

Notice moreover that if the total number of fans, including the assistive fan, is $(n+1)$ then the total number of distances $d_{ij}(\mathbf{r_i}, \hat{\mathbf{F}_i}, \mathbf{r_j}, \hat{\mathbf{F}_j})$ to be evaluated and therefore the associated inequality constraints, are $\binom{n+1}{2}$ and for example $n + 1 = 7$, the total number of the distances to be evaluated are 21. A matter that requires special attention and will be mentioned again later.

Chapter 5

Solution to the Design problem

5.1 Introduction

The general purpose of this Chapter is to propose a solution to the *Design problem* of the Chapter 4, which is besides an *optimization problem*. In that way, for completeness reasons reference will be made on the classification and on the background of the optimization methods/problems and according to these the procedure to the solution will be stated.

5.2 Classifying optimizations

Optimization might be defined as the science of determining the "best" solutions to certain mathematical defined problems, which are often models of physical reality as the one in here (4.33). It involves the study of optimal criteria for problems, the determination of algorithm methods of solution, the study of the structure of such methods and computer experimentation with methods both under trial conditions and on real life problems.

The Classification of the problem into different categories is crucial because according this, the methods that will be used in order to solve it diverse.

Continuous and Discrete optimization

A basic differentiation of the optimization problems is between the *continuous* and *discrete* optimizations.

Continuous problems, involve real variables as free variables or free parameters or design variables or optimization variables, that might have upper and lower boundaries that in fact define the "searching space" of each variable. It is obvious that in continuous problems the optimum solution is being searched between an infinity number of "candidate solutions"

In discrete optimization problems integer solutions are being searched from a finite (usually large) number of desirable solutions. These optimization problems have objective functions and constraint functions that are usually differentiable.

Bare in mind that solving discrete problems with continuous optimization methods will produce uncertain results[21].

Global and Local optimization

The terms of *Global* and *Local* optimization are used so as to determine if the search concerns the global extrema or it is merely a local extrema, a solution that simply outmatch its own "neighbours". Of course the search for global solution is always the desirable one [21].

A simple way (not always the optimum) to define and find the global optimum solution using a algorithm of local optimization is to use it several times with different starting points.

Deterministic and Stochastic optimization problems

A other distinction of the optimization problems is between the *deterministic* and the *stochastic*. Attention this distinction refer to the problem not to the method of the optimization. The most economical and commercial optimization problems are stochastic, in the matter that the issues of "demand and offer" of products and of services are subject to randomness of the trade market and can not be pre-determined with clarity.

Taking the above into consideration, the Design problem (4.33) is clearly a deterministic optimization problem[21].

Deterministic and Stochastic optimization methods

On the other side of the methods not at this time of the problems, basic distinction is between the *deterministic* and the *stochastic* method. A deterministic optimization method uses the generalised meaning of the derivative of the objective function, values

of which has to calculate or to approach. In contrast to this, stochastic optimization methods is that they use random elements or organized random search of the optimum solution.

Many optimization methods nowadays use simultaneously elements from both categories[21].

Unconstrained and constrained problem

Since, as it was mentioned, focus will be given on continuous optimization problems, unconstrained problems usually have the form:

$$minimize \quad f(x), \quad x \in \Re^n$$

where $f(x)$ is referred as the objective function and the minimizing point or *minimizer* is denoted by x^*[22] .

On the other hand the structure of most constrained optimization problems is essentially contained in the following form:

$$
\begin{aligned}
minimize \quad & f(x) \quad && x \in \Re^n \\
s.t. \quad & c_i(x) = 0, \quad && i \in E \\
& c_i(x) \geq 0 \quad && i \in I
\end{aligned}
\tag{5.1}
$$

where also in here $f(x)$ is the objective function, but there are additional constraints functions $c_i(x), i = 1, 2, \ldots, p$. E is the index set of equations or equality constraints in the problem, I is the set of inequality constraints, and both these sets are finite. More general constraints can usually be put into the form, for example $c_i(x) \leq b$ becomes $b - c_i(x) \geq 0$. If any point x' satisfies all the constraints of the above equation, then it said to be a *feasible point* and the set of all such points is referred as a *feasible region* R.

The definition of a constrained local minimizer x^* is that $f(x^*) \leq f(x)$ for all feasible x sufficiently close to x^* and the x^* which has the smallest $f(x^*)$ compare to x in all feasible regions is called global.

Linear and Non-Linear programming

It is natural that when the (5.1) has linear objective function and constraints then has the title of Linear programming.

Non-liner programming is the general case of the equation (5.1) in which both the objective and constraint functions may be non-linear. Indeed there is no general agreement on the best approach and much research is still to be done. Historically the earliest developments were *sequential minimization methods* based on the use of penalty and barrier functions. These methods suffer from some computational disadvantages. Another apparently attractive idea is to define an exact penalty function in which the minimizer of the penalty function and solution of the non-linear programming problem coincide. This avoids the inefficiency inherent in sequential techniques.

Notice in here that penalty and barriers functions constitute a global approach to non-linear programming and an alternative way to proceed is to consider local methods which perform well in an "neighbourhood" of the solution [22].

Non-smooth and smooth optimization

Non-smooth (NSO) or non-differentiable optimization problems are those ones that either the objective function of the equation (5.1) is not differentiable or either the constraints[22]. Examples of NSO problems also occur when solving non linear equations $c_i = 0$, $i = 1, 2, \ldots, m$ by minimizing $||c(x)||_1$ or $||c(x)||_\infty$. Also an other similar problem arises when solving the system of non-linear inequalities $c_i(x) \leq 0$, $i = 1, 2, \ldots, m$ by minimizing this time the $||c(x)||_1^+$ or $||c(x)||_\infty^+$.

5.3 Optimization problem of the Aerial Manipulator

In this section special reference will be made on the optimization problem that mentioned (4.33) of the Chapter 4. It should be remembered here that the optimization problem, which is also the Design Problem of the AM is the following:

$$
\begin{aligned}
\min_{r,\hat{F}} \quad & J(r) \\
s.t. \quad & \sigma(D) \geq \epsilon_1 > 0 \\
& d_{ij}(\mathbf{r_i}, \hat{\mathbf{F}_i}, \mathbf{r_j}, \hat{\mathbf{F}_j}) \geq \epsilon_2 > 0 \quad \forall i, j = 1, 2, \ldots, n, \alpha \\
& \hat{\mathbf{F}_a} = -\sum_{i=1}^{n} \hat{\mathbf{F}_i} \\
& S(\hat{F}_a) \cdot r_a = -\sum_{i=1}^{n} S(\hat{F}_i) \cdot r_i \\
& \kappa(D(r, \hat{F})) \leq K
\end{aligned}
\tag{5.2}
$$

5.3.1 Objective function

As it was stated, $J(r)$ comes from the major need the AM to be as small as possible and therefore $J(r)$ represents an expression of the structure volume. An definition of:

$$J(r) = ||\mathbf{r}||_2$$

could be an excellent simple/first choice since minimizing it will entail the minimization of the structure volume. $\mathbf{r_{i,j}} \in \Re^3$ are showing the positions of thrusters with reference to the "end-effector" (world frame).

This will be the objective function of the problem, a smooth and continuous one. If someone neglect the constraints of the problem and simple focus on the minimization of the $J(r)$ will agree that this function has only one minimum, which is also global and that is when every $\mathbf{r_i}, \mathbf{r_j}$ for $i, j = 1, 2, \ldots, n, \alpha$ are equal to zero. And then $J(r) = 0$ This of course does not happen when the constraints are introduced to the optimization problem.

5.3.2 Constraints

As someone can see from the equation (5.2) the optimization problem has both inequality and equality constraints and the lower/upper boundaries of the design variables will be defined later in this diploma thesis.

Inequality constraints

The first constraint refers to the singular values of the matrix $D(r, \hat{F})$ which depicts the necessity of solving the system (4.10) so as to have one and only one solution. So, the constraint is $\sigma(D) \geq \epsilon_1 > 0$, which is clearly non-smooth and non-linear and can be rewritten so as to take the typical used form, $-\sigma(D) + \epsilon_1 \leq 0$.

The other constraint of the Design problem is the one that refer to the avoidance of the fan interaction:

$$d_{ij}(\mathbf{r_i}, \hat{\mathbf{F}}_i, \mathbf{r_j}, \hat{\mathbf{F}}_j) \geq \epsilon > 0 \quad \forall i, j = 1, 2, \ldots, n, \alpha$$

as it was mentioned in 4.5 Section. In that Section was shown also that this constraint results from an other optimization problem, that has form :

$$d_{ij}(\mathbf{r_i}, \hat{\mathbf{F}_i}, \mathbf{r_j}, \hat{\mathbf{F}_j}) = \min_{p_i, p_j} \|p_i - p_j\|$$

$$s.t. \qquad\qquad G(\mathbf{r_i}, \hat{\mathbf{F}_i}, p_i) \leq 0 \qquad\qquad (5.3)$$

$$G(\mathbf{r_i}, \hat{\mathbf{F}_i}, p_j) \leq 0$$

which reflects the necessary distance between every thruster in order to avoid the aerodynamic interaction.

That interaction could be approximated with a cylinder that has radius the radius of the thruster and appropriate length or a more complex shape, more realistic, that is commonly defined in Aerodynamic literature with the "cone" approximation shape.
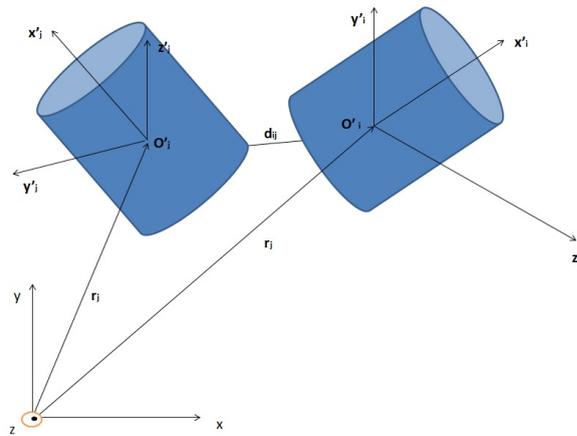
FIGURE 5.1: This figure shows the distance d_{ij} between two cylindrical aerodynamic shapes and their own coordinates in terms of world frame.

So, in every case of the aerodynamic shape, it is like having an "smaller" optimization inside a greater optimization which will define the appropriate positions and orientations of the thrusters. Notice in here that this inequality constraint is a non-smooth constraint and of course a non-linear one in terms of the outside optimization problem. But it is smooth as an identical optimization that has smooth objective function and constraints with one global always minimum. This remark for the (5.2) generally increases the complexity of finding a solution to the Design problem. It should be underlined in here that there are ways to transform the "smaller" (inside) optimization from non-smooth to a approximately smooth one using theories like the *Perturbation Approach to Sensitivity Analysis* (see [23]) but this will be out of the scope of this thesis as there are methods and optimization "packages" that deal with non-smooth equation-constraints. In any case, that will be an interesting point for further search and compare.

The other aspect of the time consumption of this "complex" inequality constraint in term of approximating-finding a solution, will be mentioned later on this thesis as this constraint has each time to compute the distances between seven (7) thrusters (cylinder/cube shapes).

This constraint can also be rewritten as:

$$-d_{ij}(\mathbf{r_i}, \hat{\mathbf{F}}_\mathbf{i}, \mathbf{r_j}, \hat{\mathbf{F}}_\mathbf{j}) + \epsilon \leq 0$$

so as to have the typical form of $c(\mathbf{x}) \leq 0$.

The other just as "hard" inequality constraint as the distance between the thrusters is the "rank condition" of the matrix $D(r, \hat{F})$ which was introduced in Section 4.6 and furthermore from a practical point of view, the usage of "condition number" instead.

$$\kappa(D) = \frac{\sigma_{max}(D)}{\sigma_{min}(D)}$$

where here

$$\sigma_{max}(D) = \sqrt{max\left\{eig(D^T D)\right\}}$$

,

$$\sigma_{min}(D) = \sqrt{mim\left\{eig(D^T D)\right\}}$$

are the maximum and minimum singular values of $D(r, \hat{F})$ and require that the condition number is not large (ideally should be close to one "1").

Thus it is required:

$$\kappa(D) \leq K$$

where K is appropriate bound to be tuned (usually low values).

Condition number was introduced because of the $rank(D) = 6$, which results from definition that there are no singular values of matrix D equal to zero. Attention should given on the fact that the $rank(D) = 6$ comes from the necessity the equation (4.10) to have one and only one solution with the fewer possible number of thrusters.

This inequality constraint from its own definition is a non-linear and a non-smooth one, although exist "methods" that approximate the Jacobian matrix of the Singular Value Decomposition (SVD) [24]. That also could be a matter of further research in order to incorporate the derivatives and transform an non-smooth constraint to approximately a smooth one and underline possible variations between the results. But this will also be mentioned later on an other Chapter.

For this analysis also this inequality constraint will be thought to be non-smooth and non-linear and will transformed also as:

$$\kappa(D) - K \leq 0$$

and now has the typical form $c(\mathbf{x}) \leq 0$.

Equality constraints

In the optimization problem (5.2) apart from the inequality constraints there are also two equality constraints. Both of them correlate the "assistive" fan/thruster with the other thrusters of the Aerial Manipulator.

The first equation that was shown in the previous Chapter at 4.4 Section indicates the direction of the "assistive" thruster:

$$\hat{\mathbf{F}}_{\mathbf{a}} = -\sum_{i=1}^{n} \hat{\mathbf{F}}_{\mathbf{i}}$$

That is a simpler constraint compare to the others as it is a sum of unit-directional vectors which corresponds to each thruster and to the "assistive" one.

The second one that follows shows the position of this extra thruster and as it was referred has the equation:

$$S(\hat{F}_a) \cdot r_a = -\sum_{i=1}^{n} S(\hat{F}_i) \cdot r_i$$

and that is not as trivial as it seems because it is like solving a system that does not always has solutions. Also remember in here that the symbol S refers to the skew-symmetric matrix as was stated in 4.4.

5.3.3 Summarize

Summarizing the above someone clearly can see that this thesis focuses on a optimization problem that although has a simple objective function (calculation of the structure volume), its own constraints on the other side are demanding as some of them are non-smooth and non-linear increasing that way the complexity. Not to mention also that the "condition number" constraint is a highly "hard" condition that determines the "behaviour" of a matrix especially when this number is bounded close to one (1). Besides that is the reason why special attention was given to the classification of the

optimization problems and methods so as the reader to understand the difficulties and the procedure of solving each optimization category.

Notice also that during the procedure of solving the optimization problem (5.2) each time should be solved a "smaller" problem of the constraint d_{ij} that contains the necessary distance of two thrusters (aerodynamic interference). But the number of thrusters is seven (7) and in that way the total number of distances to be calculated are twenty one (21). That entails to "call" twenty one times the constraint function or in other words to solve twenty one optimization problems for each "call" of the outside problem (5.2). Thus, automatically the estimated computer CPU (central processing unit) cost is increased and of course the time needed to produce a solution.

5.4 Approximation of the design problem in Matlab environment

This Section will incorporate the main goal of the entire Chapter, namely to transform all the equations of the Design problem into a program. The whole work will be made in Matlab environment with plenty of useful Toolboxes one of which is the *Global Optimization*. In here also will be analysed the functions that were made.

5.4.1 Volume

First of all it is needed to create the Matlab function that has the objective function of the Design problem (5.2).

```matlab
function f = volume(~)
%structure volume
%...
f=norm(R);
end
```

As it is shown that is a simple function that calculates the norm of the matrix/vector that contains all the distances of the thrusters. The dimension of the matrix R is 1×21 as there are seven position vectors for the seven thrusters of the Aerial Manipulator.

As it was said, a first approximation for the volume with the norm of the positions vectors indicates the general minimization of the Structure volume.

So, this function takes as inputs the positions of the thrusters and produce an approximation of the volume of the structure. These position vectors are expressed in terms of the world frame, so as the function to be user-friendly.

5.4.2 Distance between two thrusters

As it was underlined earlier the calculation of the distance between two thrusters is a result of a "smaller" (compare to the (5.2)) optimization problem that has smooth objective function and constraints. Also that minimization problem has each time one and only one global solution. That is being emphasised because this problem with a simple for example deterministic *Steepest Descent Methodology* good produce the solution [21],[22].

More specifically the objective function of this optimization problem is a simple distance of two points in space and has the equation:

$$d = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2 + (z_i - z_j)^2}$$

where x_i, y_i, z_i is a point that belongs to the ith cylinder (or any other chosen aerodynamic shape) and x_j, y_j, z_j is respectively a point that belongs to the jth cylinder (or again any other chosen aerodynamic shape). As it said a smooth and simple function has the form in Matlab code:

```
1  function f = distance(x)
2  %...
3  %distance between two points in the world coordinate
4  f = d(x);
5  end
```

Caution should be given again on the fact that the inputs of this functions are two points expressed in the world coordinates, also that was chosen in order the whole structure of program to be friendly to a future (possible) use. The output of course is the value of the distance.

Apart from the objective function of this optimization problem also the constraints are smooth. One has to know the exact aerodynamic shape that is produced above and below the thruster/fan in order to construct the appropriate equation. It is commonly known that any shape is symmetric to the rotation axe of the propel.

In here for simplicity reason will be stated the cylindrical shape that has equations (see Section 4.5):

$$-x_r \leq x' \leq x_f$$
$$y'^2 + z'^2 \leq \rho^2$$

(5.4)

where here the x' axe defines the length of the cylinder. Remember here that $O'x'y'z'$ are the coordinates of the the cylinder not the world coordinates. So there are two equation "packages" of the form (5.4) one for the ith and one for the jth cylinder.

So it has the form:

```matlab
function [c, ceq] = shape (x,~,~,~,~)

% constrains concerning the first and the second cylinder

%x to Cyc1
%...

%x to Cyc2
%...
c=[shapei(x);
    shapej(x)];
%...
ceq =[] ;


end
```

In here it should be noticed that constraint function has more than one input. In fact it has four (4) inputs, two rotation matrices for the two group of points that constitute two cylindrical shapes and the two position vectors of the cylinders.

Here arises the need of producing the Rotation and Transformation matrices from the cylinder to the world coordinate and backwards.

It is known that (see reference [25]) a basic rotation (also called elemental rotation) is a rotation about one of the axes of a Coordinate system. It is given by the matrices that follow, using right hand rule.

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

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

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

The above are three basic rotation matrices rotate vectors by an angle ϕ, θ, ψ about the x, y and z axis, in three dimensions. The angles ϕ, θ, ψ with the Greek letters are usually used.

Also a commonly used sequence of rotation in order to produce the overall Rotation matrix is rotation about the z, y and finally x axis. As a result:

$$R = R_z(\psi) \cdot R_y(\theta) \cdot R_x(\phi) =$$

$$\begin{bmatrix} \cos(\psi) \cdot \cos(\theta) & \cos(\psi) \cdot \sin(\phi) \cdot \sin(\theta) - \cos(\phi) \cdot \sin(\psi) & \sin(\phi) \cdot \sin(\psi) + \cos(\phi) \cdot \cos(\psi) \cdot \sin(\theta) & 0 \\ \cos(\theta) \cdot \sin(\psi) & \cos(\phi) \cdot \cos(\psi) + \sin(\phi) \cdot \sin(\psi) \cdot \sin(\theta) & \cos(\phi) \cdot \sin(\psi) \cdot \sin(\theta) - \cos(\psi) \cdot \sin(\phi) & 0 \\ -\sin(\theta) & \cos(\theta) \cdot \sin(\phi) & \cos(\phi) \cdot \cos(\theta) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$(5.5)$$

So that is the necessary Rotation matrix, but also the Greek letters that correspond to the rotation angles should be found. In fact, in order for someone to understand the rotation needed should recall the figure in 5.1 which shows that body coordinates can be transformed to world coordinates when the x' axis of the body frame matches the x axis of the world frame.

In here it should be remembered that as it was seen in the above figure, each cylinder is axial symmetric to each own x' axis. Notice also that the direction vector of each thruster is collinear and uni-directional with the axis x'.

In order to find the angles a directional vector should have so as to rotate and match the axis x', must:

$$F = R \cdot BF \tag{5.6}$$

Remember that directional vectors can be parallel transported. Thus, in order to match a vector with an axis, only the appropriate rotation is needed. F is a matrix with dimension 4×1 that has the unitary directional vector $\hat{F}_i$ of the ith cylinder

$$F = \begin{bmatrix} \hat{F}_{ix} \\ \hat{F}_{iy} \\ \hat{F}_{iz} \\ 1 \end{bmatrix}$$

R is the Rotation matrix that was stated and BF is a matrix that incorporates the axis x' of the body frame, in other words:

$$BF = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 1 \end{bmatrix}$$

Notice that all are written in homogeneous coordinates form.

Moreover the second part of the equation (5.6) that has the Rotation matrix R and the BF matrix can be rewritten as:

$$R \cdot BF = \begin{bmatrix} \cos(\psi) \cdot \cos(\theta) \\ \cos(\theta) \cdot \sin(\psi) \\ -\sin(\theta) \\ 1 \end{bmatrix}$$

So the Matlab function that calculates these angles for each directional vector has the form of the figure that follows and called *ftoeuler.m*.

```matlab
function y=ftoeuler(x,f)

%contains the necessary part of the Euler Rotation Matrix
%unit direactional vector
f=f/norm(f);
%...
%rotation matrix
Rz=[cos(psi) -sin(psi) 0 0;sin(psi) cos(psi) 0 0; 0 0 1 0;0 0 0 1];
Ry=[cos(theta) 0 sin(theta) 0; 0 1 0 0; -sin(theta) 0 cos(theta) 0; 0 0 0 1];
Rx=[1 0 0 0; 0 cos(phi) -sin(phi) 0; 0 sin(phi) cos(phi) 0; 0 0 0 1];
R=Rz*Ry*Rx;

%...
y=F(R,x,f);
```

As shown, that function takes as inputs the unitary directional vector of a cylinder and produces the rotation angles.

Above stated the procedure of finding the rotation angles but still remains the subject of structuring the transformation of one point on a frame to an other point on the second frame.

In order to achieve this there is a transformation equation (see reference [26]) of the form:

$$WF = R \cdot BF + r \tag{5.7}$$

where here R is the rotation matrix whose angles ϕ, θ, ψ were found from the equation (5.6), WF (world frame) is a matrix that consists of a point expressed in world

coordinates and has the form:

$$WF = \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

and BF (body frame) is a matrix that consists of a point expressed in body coordinates (cylinder) and has the form:

$$BF = \begin{bmatrix} x' \\ y' \\ z' \\ 1 \end{bmatrix}$$

Finally r is a matrix that shows the "transportation" of a cylinder and basically has the position vector of each thruster:

$$r = \begin{bmatrix} r_x \\ r_y \\ r_z \\ 1 \end{bmatrix}$$

So, taking into consideration all these, a function that simultaneously handles the above was made. That function with that name *domi2.m*, "calls" for every pair of thrusters -that has to calculate their distance- the *ftoeuler.m* so as to find the rotational angles throughout the equation (5.6). Afterwards produces the Rotational matrices from the equation (5.5) for this pair of thrusters and introduces the results into the *solver.m*.

```matlab
1  function [x, fval] =solver(~,~,~,~)
2
3  %options = ...
4  [x, fval] = fmincon(@distance,[0 0 0 0 0 0],[],[],[],[],[],[], @(x) shape(x
        ,~,~,~,~), options);
5
6  end
7
8  function f = distance(x)
9
10 %distance between two points in the world coordinate
11
12 f = sqrt((x(1) - x(4))^2 + (x(2) - x(5))^2 + (x(3) - x(6))^2);
13
```

```
14  end
15
16  function [c, ceq] = shape (x,~,~,~,~)
17
18  % constrains concerning the first and the second cylinder
19
20  %x to Cyc1
21  %...
22
23  %x to Cyc2
24  %...
25  c=[ shapei(x);  % the cylinder i or the other aerodynamic shape
26      shapej(x)];   % the cylinder j or the other aerodynamic shape
27  %...
28  ceq =[] ;
29
30
31  end
```

where someone can see that this function uses the functions *shape.m* and *distance.m* so as to solve the "smaller" optimization problem which of course constitute a constraint to the general optimization with the equation (5.2). But *solver.m* takes as inputs points in world coordinates and must transform them to body coordinates. That happens because in body coordinates the constraints of the equation (5.4) of each cylinder are valid. But the equation (5.7) transforms a point in body coordinates to a point in world coordinates. So, *solver.m* uses the inverse Transformation:

$$BF = R^{-1} \cdot (WF - r) \tag{5.8}$$

as R^{-1} exist.

To connect the natural float of the script, it was stated that there is a function that introduces date into the *solver.m*. That function called *domi2.m* and has the following form:

```
1   function[x, fval]=domi2 (ri,rj,fi,fj)
2   % % define the translation and orientation vectors for both cylinder 1 and 2
3   % clear all;
4   % close all;
5   % clc;
6   %
7   % tic
8   %%%%%%%%%%%%%%%%%%%%%%%%%%%
9   %Find tranformation matrices
10  %%%%%%%%%%%%%%%%%%%%%%%%%%%%
11  opt.Display='off';
12  %...
13
14      angles= fsolve(@(x)ftoeuler(x,fi),[0 0 0]',opt); %finds the rotation angles
```

```
15  %...
16      phi=angles(1);
17      theta=angles(2);
18      psi=angles(3);
19  %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
20      angles=fsolve(@(x)ftoeuler(x,fj),[0 0 0]',opt); %finds the rotation angles
21      phi=angles(1);
22      theta=angles(2);
23      psi=angles(3);
24  %...
25  RR2=Rz*Ry*Rx;
26  %...
27  %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
28  %follows the solving of optimazation problem
29  %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
30  [x, fval] =solver(RR1,RR2,ri,rj);
31
32  end
```

To conclude *domi2.m* is the function that takes as inputs the direction and position vectors of two thrusters and produces, through solving an optimization problem (*solver.m*), the distance between them d_{ij}.

5.4.3 Constraints of the design problem

In this subsection emphasis will be given on analysing the formation of the constraints of the Design Problem (5.2) in Matlab environment.

In the previous subsection was stated the way to produce the distance between two thrusters. In fact this now should be transformed into a constraint of the Design Problem of the Aerial Manipulator. That can happen when obliging that distance to be greater than an ϵ as was referred earlier.

The function that will form all the appropriate constraints in Matlab must also incorporate the equations from where these constraints produced. In other words this function must have the equations of the previous Chapter *Technical Problem Statement* 4 that will not be mentioned again in here.

```
1  function [c, ceq]= cyl_d(x)
2
3  %where x is the both r and f
4          %...
5          R=x(1:21); %contains all r (the last one is the assistive)
6          F=x(22:42); %contains all f (the last one is the assistive)
7          %%%%
8          %producing the unit vectors
9          %%%%
10         f=[];
```

```matlab
11          %%%%
12          %producing the SF matrices
13          %%%%
14          %...
15          SF
16          %%%%
17          %producing the SF,r product
18          %%%%
19          SF1r1=SF1*R(1,1:3)';
20          SF2r2=SF2*R(2,1:3)';
21          SF3r3=SF3*R(3,1:3)';
22          SF4r4=SF4*R(4,1:3)';
23          SF5r5=SF5*R(5,1:3)';
24          SF6r6=SF6*R(6,1:3)';
25          %%%%
26          %producing the E matrix
27          %%%%
28          E_negative=[SF1r1 SF2r2 SF3r3 SF4r4 SF5r5 SF6r6];
29          F1=[f1' f2' f3' f4' f5' f6'];
30          %%%%
31          %producing the D matrix
32          %%%%
33          D=[F1;E_negative];
34          %%%%
35          %Singular value of D matrix
36          %%%%
37            s=svd(D);
38          %%%%
39          %Calculation of the 21 distances
40          %%%%
41  for i=1:6
42      for j=i+1:7
43
44          ri= [R(i,1:3)]';
45          rj= [R(j,1:3)]';
46          fi= [F(i,1:3)]';
47          fj= [F(j,1:3)]';
48          [x, fval]=domi2(ri,rj,fi,fj);
49          dij(i,j)=[fval];
50      end
51  end
52  % Nonlinear inequality constraints
53  c=[-dij +e; %epsilon is defined fromm the user choise
54      %the condition number constraint
55          cond(D)-K; %K is user defined also
56            -s+ e1];
57  % Nonlinear equality constraints
58  %...
59  %...
60  ceq=[abs(ft);
61      abs(SFR)];
62  end
```

As it is shown *cyl_d.m* - in structural form - uses all the equations of the *Technical*

Problem Statement Chapter so as to produce the constraints of the Design problem (with the equation (5.2)). Naturally inequality constraints contain also and the "condition number" , whose upper boundaries can be tuned from the user. Apart from these, for one group of design variables with the symbol x, *domi2.m* function is being called twenty one (21) times and solves twenty one smaller optimization problems. That should be considered a major drawback that influences negatively the computational cost in approximating a solution.

The design variables are both the position and the orientation vectors of each thruster. As there are seven thrusters, the design variables are included in a matrix with dimension 42×1, twenty one of which are the position and twenty one the orientation vectors.

5.4.4 Conclusion

Above were formed functions that construct the procedure of solving the Design Problem of the equation (5.2). More especially these functions, in Matlab environment, form the objective function as well as the constraints that define the minimization. Attention was given, so as to clarify to the reader the inputs and the outputs of these functions and to identify the design variables. Last but not least, was underlined the disadvantage that arises in computational terms.

5.5 Searching algorithms

This section will present the Algorithms and the Methods that will be used in order to solve the optimizations. As it was said there are two completely different optimization problems. The Design problem of the equation (5.2) and the "smaller" optimization of the equation (5.3).

5.5.1 "Smaller" optimization

This is a relevantly trivial optimization with smooth objective function and constraints. Also it should be underlined here that this problem has only one global minimum, so there is no arising issue of finding a local minimum by using deterministic method. In that way it should be chosen a deterministic algorithm from the Matlab environment. Not to forget again that this algorithm should have the lowest computational cost because of the fact that this small optimization will be "called" twenty one times for every iteration of the Design problem.

In here will be used the *fmincon* function of the Matlab 7.12.0(R2011a) environment. This is a function that uses deterministic algorithms and was designed to work on problems where the objective and constraint functions are both continuous and have continuous first derivatives. Limitations that this optimization problem satisfies.

The available algorithms of the *fmincon* are four:

- interior-point

- sqp

- active-set

- trust-region-reflective

interior-point handles large, sparse problems, as well as small dense problems. The algorithm satisfies bounds at all iterations, and can recover from NaN or Inf results.

sqp satisfies bounds at all iterations and can recover as well from NaN or Inf results.

active-set can take large steps, which adds speed. The algorithm is effective on some problems with nonsmooth constraints.

trust-region-reflective requires from the user to provide a gradient, and allows only bounds or linear equality constraints, but not both.

The fastest of those four is the algorithm *active-set* and that is a major advantage for the reasons stated above. Also *trust-region-reflective* requires from the user to provide the gradient. That at first is not a problem if there was a fixed form of aerodynamic shape (see equations (5.3),(5.4)), but from an other point of view here it is proposed a method that will apply to any preference the user applies with only requiring from him the fewest. So, in here *active-set* algorithm will be chosen.

As it was said a couple of times earlier this optimization problem has *one and only one* minimum according to the inputs and that minimum is global. Also both the objective function and the constraints of the problem are continuous. Thus, with any starting point the result will not differ. That may not happen if there were multiple minimum and there *fmincon* could possible stop to a local minimum.

	fmincon
Options	Display: off, Algorithm: active-set, MaxFunEvals: 1000, MaxIter: 1000
Input test values	$f_i = \begin{bmatrix} -7.6474 \\ -2.1869 \\ 2.7023 \end{bmatrix}$, $f_j = \begin{bmatrix} -5.0523 \\ -3.3981 \\ -5.6170 \end{bmatrix}$, $r_i = \begin{bmatrix} -2.0511 \\ -0.9187 \\ 0.0363 \end{bmatrix}$, $r_j = \begin{bmatrix} -0.5943 \\ -2.8504 \\ -1.6031 \end{bmatrix}$
Starting value of the optimization	$s.p. = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$
Aerodynamic effect	Cylindrical
Parameters of the Cylinder	xf=1 (front length), xr=1 (rear length), rho=1 (radius)
PC properties	Intel(R) Core(TM) i5 CPU M460 @2.53GHz, RAM:4.00GB
Called functions	*domi2.m, fteuler.m, solver.m, distance.m, shape.m*
Elapsed time	0.085858 seconds

TABLE 5.1: This table gathers all the required features/options/data of a "run test" in *fmincon*.

As it is shown again in the above Table 5.1 *domi2.m* is the "main" function that handles all the others. Is the function that takes four inputs -two position vectors and two direction vectors in matrix form- calls *fteuler.m* and solves the equation (5.6) and produces afterwords the Rotation matrix for each of the two thrusters. After this, calls and introduces the two Rotation matrices and the two position vectors inside *solver.m*. *solver.m* with its own turn calls both the *distance.m* and the *shape.m* so as to solve the optimization problem of the equation (5.3). All these are achieved in 0.085858 seconds.

Notice in here that, as it was said, the *domi2.m* function calls *fteuler.m* and solves the equation (5.6). In order to solve this in Matlab environment, the function *fsolve.m* was chosen. *fsolve.m* is a function that solves a system of non-linear equations, as it is in the equation (5.6). This system again has *one and only one* solution. Baring this in mind and knowing the fact that *fsolve.m* requires from the user a starting point, the choice is not limited. Thus, with any starting point the result will not differ.

The Table 5.3 that follows has the needed data so as to evaluate someone the computational cost of the *fsolve.m* for a particular "run-test".

These "test" cases, were mainly formed so as to present firstly the computational costs and secondly the formation of the inputs/outputs.

Apart from these, for completeness reason, it is crucial to present also the results from this optimization. As it was said again earlier in this diploma thesis *domi2.m* produces two results. The two points in the world frame that define the distance of the two (in this case) cylinders and the distance of these two aerodynamic shapes. So using the inputs of the table 5.1 the results that follow are produced.

Results		
The two points in world frame	$\begin{bmatrix} -0.8766 \\ -1.6253 \\ -0.3846 \end{bmatrix}$,	$\begin{bmatrix} -0.7358 \\ -1.8691 \\ -0.5947 \end{bmatrix}$
Distance between the two cylinders	0.3512	

TABLE 5.2: Results from the "test" case of the optimization problem with the equation(5.3)

fsolve			
Options	Display: off		
Input test values	$f_i = \begin{bmatrix} -7.6474 \\ -2.1869 \\ 2.7023 \end{bmatrix}$,		
Starting value	$s.p. = \begin{bmatrix} 0 & 0 & 0 \end{bmatrix}$		
PC properties	Intel(R) Core(TM) i5 CPU M460 @2.53GHz, RAM:4.00GB		
Called function	*fteuler*		
Elapsed time	0.034759 seconds		

TABLE 5.3: This table gathers all the required features/options/date of a "run test" in *fsolve*.

Finally, at this point it is needed to refer that also an another function was made with the name *plotstructure.m* that plots every thruster in Matlab environment. That function uses as inputs the position and the directional vectors of each thruster so as to plot the seven of them in world coordinates. More specially, this function incorporates the aerodynamic effects in cylindrical shape (at this "test" case) with dimensions that are shown in the table 5.1. *plotstructure.m* also calls an other function with the name *Cylinder.m* which plots a cylinder in space with a given direction, position and length. Apart from these *plotstructure.m* plots also the directional vectors, so as the user to identify the orientation of each thruster.

The "test" case above refers to only two thrusters and their location in world space is shown below.
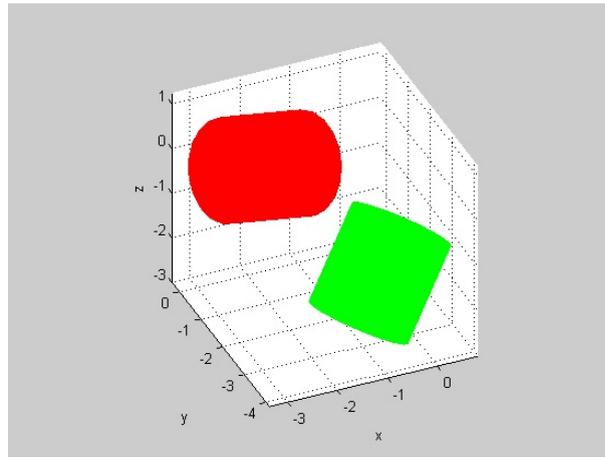
FIGURE 5.2: This figure shows the location of two thrusters in the world frame. The red cylinder is the i and the green is j of the "test case".

5.5.2 Design problem optimization

This subsection will refer to the searching algorithm that used so as to solve the Design problem of the equation (5.2).

As it was stated this is an optimization problem with some non-linear and non-smooth constraints which are also discontinuous functions. Taking these into consideration it should be chosen an algorithm "package" that can handle these type of peculiars because when a problem has them it is difficult to find a feasible point in a NLP. Notice that it may not be a *priori* whether a model is feasible, or if it is, how many dis-contiguous feasible regions there are and where are located, as in this case. In addition to that, there are some non-linear optimization algorithms that require a feasible starting point and many others are more efficient and effective if started at a point that is either feasible

or close to feasibility. So the problem of finding a feasible point might be just as hard as the global optimization problem, which is the one of the main goals of this diploma thesis.

The scope of this analysis is to propose a method that will produce at any case results for the optimization problem, that satisfy the constraints. So, there are methods that use stochastic searching criteria and introduce different randomness ways of detecting optimal solutions. This stochastic way of scouting the solution space is a basic procedure that prevents from trapping into local extrema [21]. Above all that is crucial for the problem of the equation (5.2) which as it is shown later has disjoint feasible regions. That increases the difficulty of finding an optimal solution and therefore the global minimum.

Apart from these there are methods that do not follow the principle of evaluating and optimizing, at each repetition, only one solution but a group of them. This pluralism increases the computational cost but on the other side provides the certainty to the solution to be a global minimum. The methods that handle many solutions at each time (or in one sense a population of solutions) are called *population-based optimization methods*. In the same category someone can find the methods that observe the evolution of a population of solutions and use effectors that remind the biological evolution of populations. These are the *evolutionary algorithms*.

Previously are mentioned two particular procedures of confronting the difficulty of trapping into a local optimum. In here must be underlined again that this difficulty (in the terms of the optimization problem that this analysis is dealing with) arises due to the dis-contiguous feasible regions and not to the existence of multiple local optimum solutions. Remind that the objective function of the design problem (see section 5.4.1) has only one optimum which is also global. Firstly, because of the constraints, this optimum is not the same as if there were no constraints and secondly these constraints produce dis-joint feasible regions which result to the creation of local extrema (non optimum). So, the goal is to identify the extrema that is closest to the global solution or even better the global solution itself.
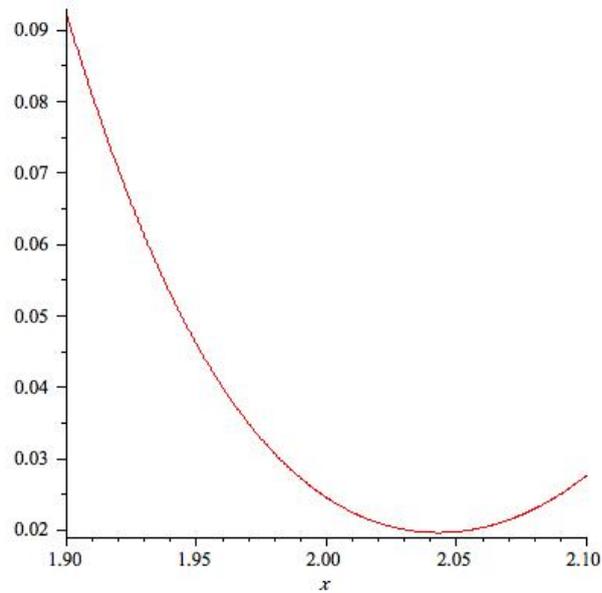
FIGURE 5.3: This figure shows a function $f(x)$ defined in a spectrum that has one global minimum for $x \approx 2.04$

The figure 5.3 shows a function that is defined in a spectrum that has only one optimum solution which is global minimum. Introducing some constraints to this function $f(x)$, which define three feasible regions from 1.90 to 1.94, from 1.96 to 1.98 and from 2.07 to 2.10, will automatically change the global minimum from $x \approx 2.04$ to $x \approx 2.07$ as this point satisfies the constraints. Apart from this, it is shown that these constraints produce some local feasible extrema (minima) as for example the point at $x = 1.94$ or $x = 1.98$ that are not the optimum solution. These type of minima might occur and "trap" a possible solution of the Design problem.

To clarify to reader the difference the figure that follows shows a function that has multiple minima in comparison to the previous one that has only one.
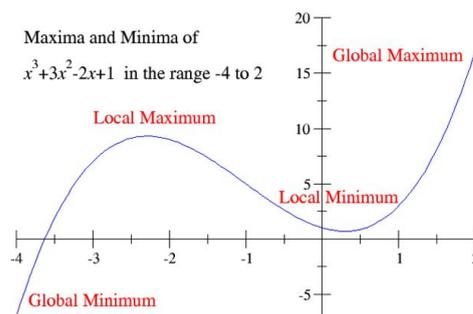
FIGURE 5.4: This figure shows a function $f(x)$ with multiple optimum solutions.

Thus, selecting the appropriate algorithm for the Design problem of the equation (5.2) is not simple as this algorithm or package should handle the above limitations and complexities. It is worth mentioning in here that were made a lot attempts and were used a lot of packages that deal with this type of dis-joint/non-smooth and non-linear optimization. Most of them are gathered in an Appendix which shows in every case the reasons for not choosing them. These methods were both stochastic based, such the *Genetic algorithm* or *Hybrids* as well deterministic based. Remember in here that is being searched a package that will always produce correct results to any preference the user applies and will be easy to use.

The "package" that finally produced results in every case, is from the *Global Toolbox* of the Matlab 7.12.0(R2011a) version and called *pattern search*. Is important to mention in here the way this algorithm works.

Pattern search finds a local minimum of an objective function by the following method, called polling. The search starts at an initial point, which is taken as the current point in the first step:

1. Generate a pattern of points, typically plus and minus the coordinate directions, times a mesh size, and center this pattern on the current point.

2. Evaluate the objective function at every point in the pattern.

3. If the minimum objective in the pattern is lower than the value at the current point, then the poll is successful, and the following happens:

    (a) The minimum point found becomes the current point.

    (b) The mesh size is doubled.

    (c) The algorithm proceeds to Step 1.

4. If the poll is not successful, then the following happens:

    (a) The mesh size is halved.

    (b) If the mesh size is below a threshold, the iterations stop.

    (c) Otherwise, the current point is retained, and the algorithm proceeds at Step 1.

Notice in here that Direct search is a method for solving optimization problems that does not require any information about the gradient of the objective function. Unlike more traditional optimization methods that use information about the gradient or higher derivatives to search for an optimal point, a direct search algorithm searches a set of

points around the current point, looking for one where the value of the objective function is lower than the value at the current point. You can use direct search to solve problems for which the objective function is not differentiable, or is not even continuous.

Global Optimization Toolbox functions include three direct search algorithms called the generalized pattern search (GPS) algorithm, the generating set search (GSS) algorithm, and the mesh adaptive search (MADS) algorithm. All are pattern search algorithms that compute a sequence of points that approach an optimal point. At each step, the algorithm searches a set of points, called a mesh, around the current point—the point computed at the previous step of the algorithm. The mesh is formed by adding the current point to a scalar multiple of a set of vectors called a pattern. If the pattern search algorithm finds a point in the mesh that improves the objective function at the current point, the new point becomes the current point at the next step of the algorithm.

The GPS algorithm uses fixed direction vectors. The GSS algorithm is identical to the GPS algorithm, except when there are linear constraints, and when the current point is near a linear constraint boundary. The MADS algorithm uses a random selection of vectors to define the mesh. So, as it was seen and tested the more appropriate of these three algorithms, for the Design problem, is the GPS algorithm.

This is a simple algorithm, with some minor modifications, provides a robust and straightforward method for optimization. It requires no gradients for the objective function and for the constraints and therefore is an algorithm that deals with non-smooth and non-linear constraints.

At this point will follow some basic options that were used to the *pattern search* package.

The basic option tablets that the user is called to fill and change when using *pattern search* of the Global Optimization Toolbox are:

- poll

- search

- mesh

- constraints parameters

- cache

It must be written again here that pattern search requires from the user a starting point that is not necessary satisfies every constraint.

Poll

As it was said, it was chosen the GPS algorithm. More specifically was chosen GPS positive basins 2N, because this algorithm searches more points at each iteration from the GPS Positive basis Np1. Here emphasis was given to the need of searching the most possible search-space, so as to find the closest solution to the global. Again in order to search a big spectrum of points *Complete poll* was changed to "on". That option specifies whether all the points in the current mesh are polled at each iteration. Because of the fact *Complete poll* is "on" there is no need to define the *Polling order*.

Search

Search options specify an optional search that the algorithm can perform at each iteration prior to the polling. If the search returns a point that improves the objective function, the algorithm uses that point at the next iteration and omits the polling. So the option *Complete Search* is turned to "on" so as to use this search pattern that is offered. As far the *Search method* option is concern it was tested and works perfectly the method *Latin hypercube*.

In here it should made a parenthesis that refer to the *Latin hypercube sampling*, which is used for the initial sample within the variable space defined by the variables bounds. This ensures that the points are distributed throughout the search space, and Latin hypercube sampling it known to provide better coverage than simple random sampling (reference [27]).

The other options in *Search* tablet kept the default values.

Mesh

The other option tablet handles the mesh of the pattern search. The mesh size was kept in default values and there was made no use in *Accelerator* because it is proposed in Matlab's Help to use the mesh accelerator for problems in which the objective function is not too steep near the optimal point, which is not known from the begging. That is the reason that *Accelerator* is "off". Apart from this *Scale* is not used when there are equality constraints and therefore turned "off". The other factors kept the default values.

Constraints parameters

The options in *Constraints parameters*, defining penalty factors, kept the default values.

Cache

The pattern search algorithm can keep a record of the points it has already polled, so that it does not have to poll the same point more than once. If the objective function requires a relatively long time to compute, the cache option can speed up the algorithm.

The memory allocated for recording the points is called the cache. This option should only be used for deterministic objective functions, but not for stochastic ones (source: Matlab's help). Also *Cache* was kept "off" because otherwise, pattern search might fail to identify a point in the current mesh that improves the objective function because it is within the specified tolerance of a point in the cache. That is not preferable for the optimization problem of the equation (5.2).

As far as the other options of the *Optimization Toolbox* is concern there will be no notation as these are options that define the stopping criteria, the outputs and the display in general. Options that their definition differ and vary depending on the accuracy and on the users requirements.

Chapter 6

Simulation and Results

6.1 Introduction

This Chapter will incorporate the theoretical Model of the Chapter 3 into the Simulations and the produced results will be stated.

6.2 Experiment

In this point it worth mentioning that there has been made also, from a fellow student, a experiment that held in Control System Laboratory (CSL) with supervisor Prof. Kostas J. Kyriakopoulos. This experiment was made to the *Asctec Firefly*, a hexarotor with $8 \times 4.5 inches$ size of propellers and with *Hacker* motor.

The basic goal of this was to identify firstly the shape of aerodynamic effects in terms of different rotation speeds of the blade (rpm), and secondly represent the distribution of Thrust and Torque against the rpm. The basic results are summarized in an Appendix **??** that follows. So, in every case needed in this diploma proper reference to this experiment will be made.

6.3 Propellers

In here was chosen to use Propellers from the company with brand-name *APC Model Air plane Propellers*. That happens because this is a company with a lot of experience in the field of manufacturing air-foils in general, has a big variety of shapes/sizes for the propellers and offers to the possible buyer specific performance data information to a

logical extent. This company also it is known to the market for the reliable and quality products.

Here follow some specific properties of the air foils used in *APC* so as to produce each blade.

Generally, the air foils may have arbitrary shapes defined with either tabular data (splined cubic fits) or analytical functions typically used for NACA airfoils. The airfoil shapes may vary with span. The dominant basis for the primary airfoil shape used in most APC propellers is similar to the NACA 4412 and Clark-Y airfoils, except the leading edge is somewhat lower. Also, the aft region is somewhat thicker. This alters the zero-lift angle by approximately one degree and provides greater lift without having to twist the blade even more. All blades of this manufacture have some washout near the tip.

This company do not offer any further information about the other characteristics of the blade, such as the collective pitch (θ_0) or the blade twist (θ_1).

Finally, as it seen at the performance data, for each propeller rpm there is no fix value on the produced Thrust force and Torque. In fact the Thrust/Torque and their coefficients are not constant in every rpm but their values vary with the change of UAV speed. Also in some cases a gap of information is observed between several rpm. On the other hand, this is not useful in terms of producing Simulation results and this constitutes one of the reasons of stating - to that extent - the Model of the Chapter 3 but this wields later in this thesis.

6.4 Motors

As it was said the *Asctec Firefly* used in the CSL experiment has 8 inches propeller. So it was found that *Scorpion* motors cooperate well with the 8 inches *APC* propellers and offers to the buyer performance data and a vast variety of products depending on every preference. Notice in here that are companies that considered to have lower quality products such as *Cobra* motors, or some others for which there is little information about their products and these reasons prevent from choosing them in here.

Apart from this, it is very important for every Simulation attempt to choose specific categories of propellers and motors with specific performance data that cooperate correctly. More specifically, as it was mentioned earlier in this thesis, Flaying Manipulator is a multi-rotor UAV for slow flying operations. So, the constant

$$K_v = \frac{RPM}{Volt}$$

should take values 800-1200 approximately. In here it should mentioned that K_v as used refers to the rpm constant of a motor - it is the number of revolutions per minute that the motor will turn when 1V (one Volt) is applied with no load attached to the motor. So it is called as the revs per volt. It is related to the power out from a motor, or more usefully the torque level of a motor. It is determined by the number of winds on the armature (or turns as we sometimes call it) and the strength of the magnets, there are so many variables with electric motors. So KV allows to get a handle on the torque expected from a particular motor.

In summary, a low KV motor has more winds of thinner wire - it will carry more volts at less amperes, produce higher torque and swing a bigger prop. That may sound confusing, but compare it with a high KV motor which has less winds of thicker wire which will carry more amperes at less volts and spin a smaller prop at high revs.

As it was seen an other factor that should be considered, before any effort of matching the appropriate motor with a propeller, is the maximum amperes and and voltage a motor can handle depending on the propeller rpm and size so as to produce the preferable amount of Thrust force.

6.5 First simulation

Above was made a general reference to the companies and to their products, this section will gather more accurate data so as to perform a simulation.

6.5.1 Choosing propeller and motor

The Design of such a UAV can be characterized with the terms of originality and innovation and with that in mind *Asctec Firefly* will be used as a benchmark. More especially, will be used propellers that have approximately the same dimensions with those used in *Asctec Firefly* and as it mentioned the motor will be from the same company. Searching for a propeller close to these dimensions in *APC* list products someone can see that for *slow fly* UAV is the $8 \times 3.8SF$ with the product code LP08038SF (where SF means *slow fly*).

APC propeller	Motor Scorpion	Thrust (grams)	i(A)	RPM
	SII-2212-1070 i(max)=15A			
	11 Volts			
$8 \times 3.8SF$		658.2	10.17	9443
$8 \times 4E$		600.7	9.03	9753
$7 \times 6SF$		456.6	9.33	9673
	7.4 Volts			
$9 \times 6SF$		483	10.3	5528
$10 \times 3.8SF$		538	9.59	5730
$11 \times 3.8SF$		593	10.51	5508
	SII-2212-960 i(max)=13A			
	11 Volts			
$9 \times 4.5E$		704	9.81	8480
$9 \times 3.8SF$		733	10.65	8250
$8 \times 3.8SF$		580	8.18	8924
	7.4 Volts			
$10 \times 4.7SF$		515.2	8.61	5271
$11 \times 3.8SF$		540.5	8.64	5254
$11 \times 4.7SF$		598	10.28	4873
	SII-2205-1585 i(max)=12A			
	11 Volts			
$6 \times 4E$		372	7.83	13344
$5.5 \times 4.5E$		288	7.27	13735
	7.4 Volts			
$8 \times 3.8SF$		381	8.79	7395
$7 \times 6SF$		278	8.4	7677
$7 \times 5SF$		301	7.48	8244
	SII-2205-1900 i(max)=12A			
	11 Volts			
$4.7 \times 4.2E$		251	7.15	17029
$4.7 \times 5.5E$		225.1	8.9	15965
	7.4 Volts			
$6 \times 4E$		253	6.41	11160
$7 \times 4SF$		369	8.95	9865
$7 \times 5SF$		371	10.36	9118

TABLE 6.1: A table that gathers a variety of information concerning four types of Scorpion motors

The above is a table that has different information about four Scorpion motor and emphasis was given so to select APC propeller sizes close to the dimensions $8 \times 4.5 inches$ of the *Asctec Firefly*. So, when these propellers are found from the product list of the company, effort was made so as to find as well the appropriate motors to cooperate.

It is known that limitations are introduced from each motor power which bounds automatically the ability of propeller to produce Thrust force. In contrast to this, the table 6.1 proposes matches between propellers and motors. Most matches of the table above refer to a very well motor-propeller sizing, but there are cases as for example in SII-2212-960 with $9 \times 3.8SF$ and $9 \times 4.7SF$ where the operation corresponds to "burst" (which means operation for short period of time). The data so as to produce this table were driven from the information these two companies offer and are summarized in Appendix.

Moreover it should be mentioned that the motors with the code name SII-2212-X is a more "quality" category in comparison to the SII-2205-X which besides someone can see it from the price and the weight.

For all these reasons, of perfect matching between propeller-motor, of quality products and of produced force it was chosen SII-2212-1070. That happens because although $8 \times 3.8SF$ propeller can be used with SII-2205-1585 and with SII-2212-960, the first motor of the two has lower quality characteristics and the second produces lower maximum force in comparison to the SII-2212-1070 motor. Thus, for the blade $8 \times 3.8SF$, according to the table above the maximum producing Thrust force is 658.2 grams at 9443 rpm. Notice that this combination of propeller-motor is a "well-sized" match and corresponds to smooth operation and also K_v is between the values for slow flight UAV's and is equal to $K_v = 1070$

6.5.2 Aerodynamic effects

As it was mentioned earlier the experiment in CSL produced the aerodynamic effects for one propeller/motor of the *Asctec Firefly*. The propeller has dimensions $8 \times 4.5 inches$ and the measurements where made each time for different rpm of the motor and are shown in Appendix.

It was chosen not to consider the cylindrical shape (see Chapter 5) for the aerodynamic effects between the thrusters but to incorporate the results from the experiment into this analysis. Using them will transform the Simulation to a more accurate and realistic version.

Here becomes clear the reason of choosing a propeller that has approximately equal dimensions to those used in *Asctec Firefly*. Although as it seen the propeller from the APC does not have exactly the same dimension, these two propellers have the same diameter (8 inches) which is crucial. So, it terms of strict aerodynamics these two propellers might not produce the same fluid flow shape around a thruster. In any case though, incorporating the aerodynamic effects of the experiment will imply a more realistic Simulation results from just selecting an arbitrary cylindrical shape.

From the above subsection 6.5.1 it was found that the APC propeller with sizes $8 \times 3.8 SF$ when cooperating with the Scorpion motor SII-2212-1070 produces at 9443 rpm, 658.2 grams of force.

It is needed to refer at this point a figure that depicts the air flow throughout the rotor blades of a Helicopter in hovering state (out of the ground effects) and in ground effects.
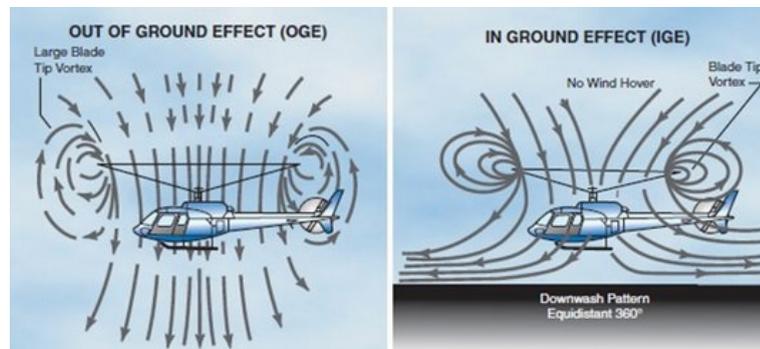
FIGURE 6.1: A figure that shows the aerodynamic approximation shape of the air flow around a Helicopter out of the ground and in the ground effects.

Of course as said the ground effect will not be considered in this analysis but as the figure shows the flow of the air changes dramatically due to the ground effect and was stated so as to see the difference in every case. So, as someone can see, the flow of the air is being affected both at the entrance and exit of the rotor. In fact as also the experiments showed, the flow at the exit of the rotor is being affected more than the flow at the entrance. As a result the aerodynamic shape reminds a One-sheet Hyperboloid plotted in Matlab environment.
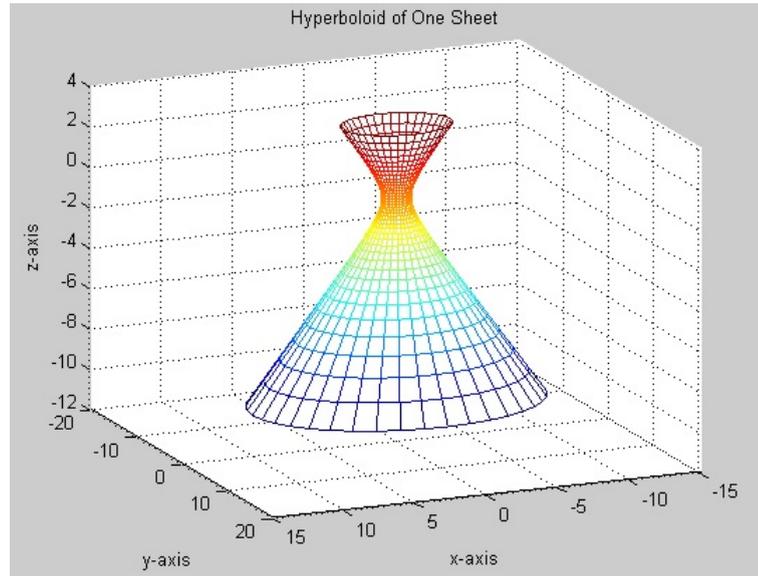
FIGURE 6.2: One-sheet hyperboloid rotation shape plotted in Matlab environment with the function *Hyperboloid_of_One_Sheet.m*.

As said because the flow at the exit of the rotor is more affected, the One-Sheet Hyperboloid is being extended more in the z-axis.

The measurements from the experiment that are closer to the 9443 rpm, in which the maximum thrust force of the chosen APC propeller is observed, are those that follow.
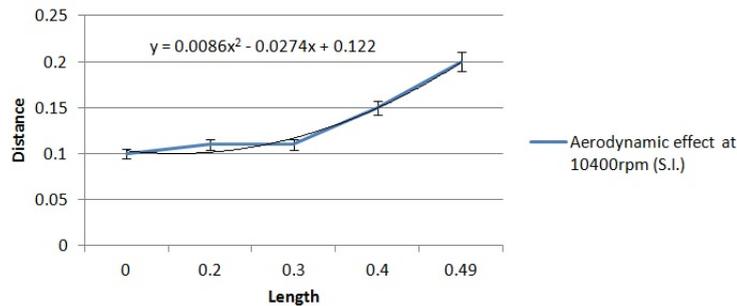
FIGURE 6.3: This figure shows the aerodynamic effect at 10400 rpm of *Asctec Firefly* experiment. The blue line shows the measurements and on top of this is the 2nd order curve that approximates this line. The x axis shows the length of the aerodynamic effect of the exit flow and the y axis shows the distance of the effect measured from the rotation axis of the rotor. The values are expressed in S.I.

This figure, as written, shows with the blue line the measurements that gathered from the experiment of the *Asctec Firefly* in 10400rpm and these are the closest measurements that can be compared to the APC propeller, which in 9443 rpm produces the thrust force. The x axis shows the length of the aerodynamic effect of the exit flow and in order to understand this someone should think as if the rotor/blade is at $x = 0$. On the

other side, y axis shows the distance of the effect measured from the rotation axis of the blade, where in $x = 0$, $y = \frac{8''}{2} = \frac{20.32cm}{2} \approx 0.10m(S.I.)$ which is approximately the blade radius (because also an offset exist).

Furthermore, must be shown in here the aerodynamic effect that is being created at the entrance of the air flow to the rotor. The only information given from the experiment in 10400rpm, is that this length extents $0.04m$ above the rotor and there were made no measurements so as to approximate the curve. For simplification reasons will be thought as if the curve of the figure 6.1 starts from $x = -0.04m$ and stops again at the $x = 0.49m$ so as to incorporate the aerodynamic effects at the entrance of the rotor. Again it should be remembered that in $x = 0$ is the rotor/blade and this is the reason why $y = 0.1m(\approx$ blade radius$)$.

Summarizing the above, it is clear that the aerodynamic effects of the air flow throughout the rotor (from the entrance to the exit) are extended from $x = -0.04m$ to $x = 0.49m$ and the curve that approximates these effects is the 2nd order equation

$$y = 0.0086 \cdot x^2 - 0.0274 \cdot x + 0.122 \quad (S.I.) \tag{6.1}$$

This second order curve approximates the blue line, that is produced from the experiment results, with accuracy of the order 5%.

Based on the above equation here must generate the three dimensional (3D) shape that will reflect the aerodynamic effects of the chosen APC propeller.

In order to generate the 3D shape must rotate the equation (6.1) from a fixed axis and follow the rules of producing symmetrical rotational shapes.

Thus, the aerodynamic effects will be introduced to this analysis with the following equation:

$$\begin{aligned} -0.04 \leq x' \leq 0.49 \\ y'^2 + z'^2 \leq (0.0086 \cdot x'^2 - 0.0274 \cdot x' + 0.122)^2 \end{aligned} \quad (S.I.) \tag{6.2}$$

The equation (6.2) is written in correspondence to the equation (5.4) and so it is expressed to the body coordinates of the shape and hence the tuna (') to the variables of the equations. Also, the length of the aerodynamic shape is specified in x' axis as was followed with the equations of the cylinder.

It must be mentioned that the above equation was a result of the parametric version bellow.

$$z = \sin(v) \cdot (0.0086 \cdot x'^2 - 0.0274 \cdot x' + 0.122)$$
$$y = \cos(v) \cdot (0.0086 \cdot x'^2 - 0.0274 \cdot x' + 0.122) \tag{6.3}$$
$$x = u$$

where

$$-0.04 \le u \le 0.49$$
$$0 \le v \le 2 \cdot \pi$$

This aerodynamic shape in Matlab environment is shown in the next figure.
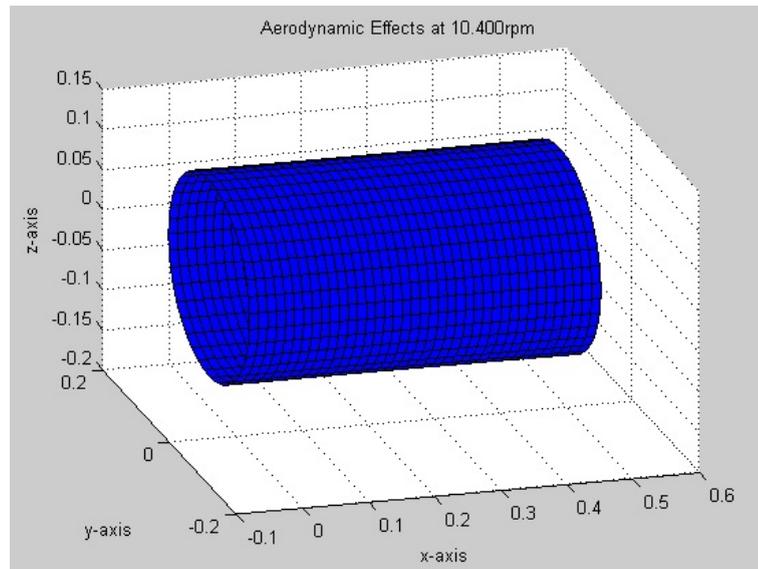
FIGURE 6.4: This figure shows the shape of the 3D aerodynamic effect in Matlab environment, based on *Asctec Firefly* experiment at 10400 rpm.

At first the shape of this figure might seem to be identical to cylindrical but from a closer point this changes when someone uses the Matlab function with the name *aerodynamic_effect_surface_exper.m*, that creates and plots the form based on the equation (6.3).

6.5.3 Determine thrust/torque and their coefficients

The subject of determining thrust force, torque and their coefficients arises again as new manufacturing values introduced into the analysis.

As mentioned in the previous subsection, *APC Model Air plane Propellers* offers to the buyer a list of performance data for each size of propeller. However, in this list for every rpm the values of thrust, torque and their coefficients differ according to the changes in

UAV speed (see Appendix C, B). Then, as it seen from the Appendix C, APC offers data for specific rpm and for example there is no information about 9443 rpm (that produce maximum thrust force in SII-2212-1070 for the $8 \times 3.8SF$ propeller) or anything between the 8000rpm and 12000rpm.

It should be remembered in here the main characteristic of the Aerial Manipulator, which is the ability to interact with the environment throughout the end-effector and produce corresponding actuating force (F_{act}) and torque (M_{act}). That means that with every particular demand for actuating force (F_{act}) and torque (M_{act}), every thruster will be obliged to rotate in specific rpm that might not be in the performance data APC offers and produce specific thrust force (respecting always the manufacturing limitations). And of course the same happens with the anti-torque and their coefficients, since these are interdependent (see Chapter 3).

The above will be an impetus to seek a relationship between thrust force and rpm or torque and rpm, that corresponds to a more simplified formation and more useful for this analysis. A relationship that gives clear information about the distribution of thrust and torque in every rpm.

The only information found for the $8 \times 3.8SF$ APC propeller concerns the relationship between thrust force and rpm is the one that follows.
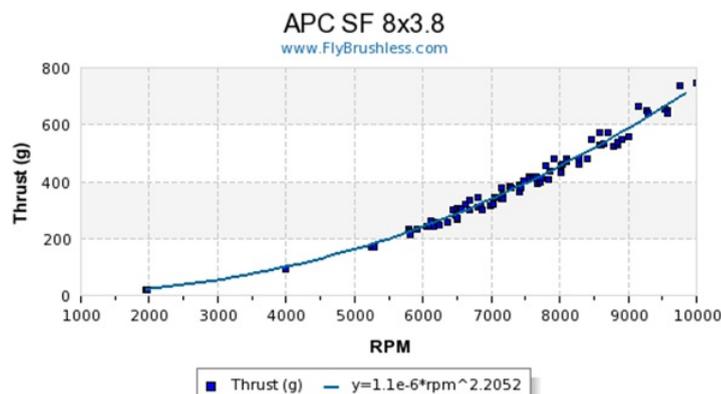
FIGURE 6.5: This figure shows the relationship between thrust force and rpm of the $8 \times 3.8SF$ APC propeller. (source:Fly Brushless)

This figure was alleged from a site and the plotted points are results from different experiments that incurred by various (sometimes anonymous) users which used motor models that correspond to the $8 \times 3.8SF$ APC propeller. Also it is represented a curve that approximates all these experiment measurements.

Still this curve will not solve the arisen problem as no other information was found concerning the distribution of the torque against rpm.

Taking into consideration the above, in order to overcome the obstacle the following procedure was chosen.

It should be remembered in here the equations introduced in the Model of the Chapter 3 that produce results for the Thrust, Torque and for the two coefficients respectively.

The Proposed Methodology starts by calculating first λ_{ind}, for which will be used the following two equations:

$$\frac{4C_T}{a\sigma} = \left(1 - \frac{e}{R}\right)\left(\frac{2}{3}\theta_0 + \frac{1}{2}\theta_1 - \lambda_{ind}\right) \tag{6.4}$$

and the equation by Lopez Ruiz

$$C_T = 2\frac{\lambda_{ind}}{k_1}\sqrt{\left(\frac{\lambda_{ind}}{k_1}\right)^2 + \left(\frac{\mu}{k_2}\right)^2} \tag{6.5}$$

In order to determine the λ_{ind} someone should introduce the second equation inside the first one or backwards.

Notice in here that these two equations are a simplified version that correspond only to the Axial Flight and more especially in Hover State.

It should remembered that:

$\mu = \frac{V_x}{|\Omega|R}$ is the horizontal speed to tip speed ratio

$\lambda_z = \frac{V_z}{|\Omega|R}$ is the vertical speed to tip speed ratio

$\lambda_{ind} = \frac{V_{ind}}{|\Omega|R}$, where v_{ind} is the induced velocity

Therefore when $V_x = 0$ and $V_z = 0$ it is clearly also that $\mu = 0$ and $\lambda_z = 0$.

When λ_{ind} is found then by using either the (6.5) or (6.4) the coefficient C_T can be calculated. After this, also by using the value of the λ_{ind}, the C_Q can be found from the equation,

$$\frac{4C_Q}{a\sigma} = \frac{1}{2}\frac{c_{d0}}{a} + \frac{c_{d1}}{a}\left[\frac{1}{2}\theta_0 + \frac{2}{5}\theta_1 + \frac{2}{3}(-\lambda_{ind})\right] +$$
$$\frac{c_{d2}}{a}\left[\frac{1}{2}\theta_0^2 + \frac{1}{3}\theta_1^2 + (-\lambda_{ind})^2 + \frac{4}{5}\theta_0\theta_1 + \frac{4}{3}\theta_0(-\lambda_{ind}) + \theta_1(-\lambda_{ind})\right] \tag{6.6}$$

The a, c_{d0}, c_{d1}, c_{d2}, θ_0, θ_1, c, K_1, K_2, e, R are parameters that were mentioned in Chapter 3. Notice in here that will be no correction to the value λ_{ind} as stated in subsection 3.4.3 because the analysis is based on the Hover state equations.

Thus, using the C_T and C_Q, from the following equations,

$$C_T = \frac{T}{\rho \pi R^4 \Omega^2}$$

$$C_Q = \frac{Q}{\rho \pi R^5 \Omega^2}$$

(6.7)

the Thrust force T and the Torque Q can be calculated in every variation of Ω.

In this point it should be noticed that was created in Matlab environment a function that uses the above methodology and produces the C_T and C_Q and called *culc_lamda_ind_and_CT_CQ.m* and takes as inputs the a, b, c_{d0}, c_{d1}, c_{d2}, θ_0, θ_1, c, K_1, K_2, e, R (radius of the rotor). The values of these inputs are gathered in the table that follows.

Symbol	Value
θ_0	$0.049 rad (\approx 3^o)$
θ_1	$-0.09 rad (\approx -5^o)$
a	5.5 (for Hover state)
c_{d0}	0.05
c_{d1}	≈ 0.7
c_{d2}	≈ 0
c	$\approx 0.02m$
R	0.124 m (radius of the rotor)
e	$0.1 \cdot R$
K_1	$\left(\frac{9}{5}\right)^{\frac{1}{4}}$
b	2

TABLE 6.2: Values of the required parameters for the Simulation.

Some of the values of the table are already been mentioned in the previous Chapters and some others such as the θ_0, R, c are changed due to the new manufacturing data introduced from the APC propeller.

In this point, will be made a compare between the thrust force of the Model and the thrust force with the equation of the figure 6.5 and the results will be plotted.
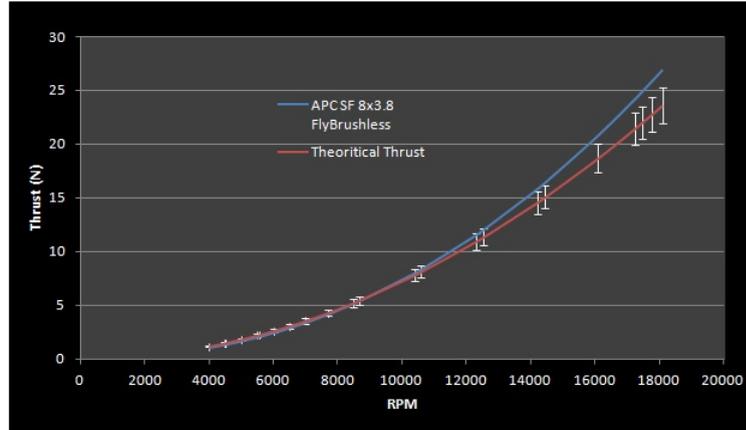
FIGURE 6.6: Comparison between the theoretical Thrust force and the one proposed in Fly Brushless against rpm, for the $8 \times 3.8 SF$ APC propeller.

It is clear that the thrust force proposed from the Model of the Chapter 3 approximates with high accuracy the equation of thrust, that is given from the measurements referred in Fly Brushless. In fact this approximation is made with less than 5% of error until 12000 rpm. After these rpm the deviation between the two curves increases, but still the maximum Thrust force for the chosen combination of propeller and motor is achieved in 9443 rpm, as was stated in the subsection 6.5.1. Moreover since the approximation is being made with accuracy in Thrust, the Model will produce same accuracy results for the Torque as well. That happens because, as shown in the Chapter 3, these two are inextricably linked. An accurate evaluation of C_T will automatically produce accurate Thrust, C_Q and Torque.

Notice that since values of specific characteristics of the blade are not given from the APC company, there were made different changes in θ_0, θ_1 so as to approximate correctly the curve of the figure 6.5. The final values of the θ_0, θ_1 are shown in the table6.2 and the *culc_lamda_ind_and_CT_CQ.m* produced a value $C_T = 0.0074$. So, as someone can see, the results from the figure 6.6 are promising.

Considering the above, from now on the evaluation of the Thrust, Torque and their coefficients will be made using the equations of the Proposed Model.

The values that are used so as to create these two plotted curves 6.6 are in the following table.

RPM	Fly Brushless equation Thrust(N)	Theoretical Thrust(N)
4000	0.965284601	1.152352785
4500	1.251575118	1.458446494
5000	1.578924409	1.843066446
5500	1.948231173	2.230110399
5550	1.987501987	2.270842168
6000	2.360325088	2.654015682
6500	2.815977752	3.114782293
7000	3.315911134	3.612410233
7700	4.091495071	4.371016382
8500	5.088002314	5.326462028
8670	5.315111737	5.541651094
10400	7.93879532	7.97384267
10600	8.279367862	8.283477833
12300	11.49347715	11.1535009
12530	11.97276303	11.57452363
14200	15.7768008	14.86543672
14450	16.39582452	15.39347526
16100	20.81062421	19.10965013
17240	24.19940312	21.91167142
17470	24.91707155	22.50022152
17770	25.87041532	23.27961746
18100	26.94172994	24.15227993

TABLE 6.3: Values so as to produce the plotted curves of thrust force in figure 6.6

For completeness reasons follows the Theoretical Torque produced from the above methodology accompanied with the appropriate table (with $C_Q = 0.009$).
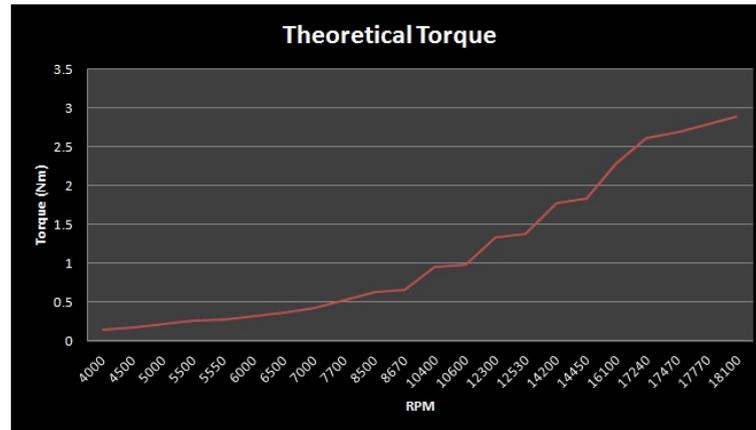
FIGURE 6.7: This is a figure that shows the Theoretical Torque produced from the Model against rpm, for the $8 \times 3.8SF$ APC propeller

RPM	Theoretical Torque$(\mathrm{N \cdot m})$
4000	0.177890781
4500	0.225143019
5000	0.277954345
5500	0.336324757
5550	0.342467549
6000	0.400254257
6500	0.469742843
7000	0.544790516
7700	0.659196525
8500	0.803288057
8670	0.835740895
10400	1.202541678
10600	1.249238008
12300	1.682068514
12530	1.745563293
14200	2.241868565
14450	2.321502485
16100	2.881941831
17240	3.304516934
17470	3.393276651
17770	3.510817984
18100	3.642424919

TABLE 6.4: Values so as to produce the plotted curve of torque in figure 6.14

6.5.4 Results

This subsection will present the results from solving the optimization problem of the equation (4.33) through *pattern search*, using simultaneously all the previous information stated above and introducing them into the Matlab functions described in Chapter 5.

Firstly as was written above the design variables of this optimization problem are forty two (42), twenty one for the seven position vectors and twenty one for the seven orientation vectors. Seven is the number of thrusters. The *pattern search* requires from the user to set the boundaries in which every design variable is obliged to range. In other words, the user must define in *pattern search* the searching space.

Apart from this it is required a starting point, which it would be preferable to result a value in objective function close to the minimum. That of course it is not so trivial to achieve for various reasons and mainly because that value is not known from the beginning. Such point though will decrease the computational costs of a future possible search with different parameters.

In order to handle the matter of the starting point it was chosen to search the design variable space with a "test" run. In this run was chosen to use the parameters of the "test" case described in Chapter 5.

fsolve		
Options	Display: off	
Starting value	$s.p. = \begin{bmatrix} 0 & 0 & 0 \end{bmatrix}$	
Called function	*fteuler*	
fmincon		
Options	Display: off, Algorithm: active-set, MaxFunEvals: 1000, MaxIter: 1000	
Starting value of the optimization	$s.p. = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$	
Aerodynamic effect	Cylindrical	
Parameters of the Cylinder	xf=1 (front length), xr=1 (rear length), rho=1 (radius)	
Called functions	*domi2.m, fteuler.m, solver.m, distance.m, shape.m*	
patternsearch		
Basic options	Poll method: GPS Positive basins 2N, Complete poll: on, Polling order: Random, MaxIter: 1000	
	Complete search: on, Search method: Latin hypercube	
	Default Mesh size options, Accelerator: off, Scale: off	
	Constraint parameters default	
	Cache: off	
	Mesh tolerance: $1e^{-6}$, Max iterations: Default, Max function evaluations: $12 \cdot 10^6$	
	Nonlinear constraint tolerance: $1e^{-15}$	
Called functions	*cyl_d.m, volume.m*	

TABLE 6.5: This table gathers the parameters of the run "test" in *pattern search*.

	s.p.= [-2.05109799923668,-0.91865746591622,-0.0363271233776378
	-0.59431171268305 5,-2.85040884729230,-1.603110991904 53
	0.30575229159237 8,0.44833096447283 5,-2.70758919725553
	0.49926282301941 5,-2.50743426978193,1.80515353511907
	1.36784322485729,1.01609081203704,2.16090941853865
	0.98098116472425 6,2.88938400434531,-0.94981018260036 3
	-3.32543774643142,2.45371549353136,0.34392151873686 7
Starting value of the optimization a 1×42 matrix	-7.64743002119190,-2.18690630613150,2.70226873536000
	-5.05233297343551,-3.39810441196491,-5.61696467002869
	5.74080674467992,4.49633672475667
	26.4861292254579,4.41704691325084,-6.72272413249351
	2.09349303657401,7.91662992676383,3.68824517544570,3.12357814026146
	-2.98225541687803,8.74143929720683,-2.14134300677311,1.17623777754876 0
	-0.01931217559811 36,-4.73885810673221]
Elapsed time	30 days
PC properties	Intel(R) Core(TM)2 Quad CPU 99550 @2.83GHz, RAM: 2.00GB

TABLE 6.6: This is a sequel table of the previous with extra parameters and information

Design variables after the optimization, a 1×42 matrix	[-1.1156943627868698 -0.9175989960313365 -0.0360989903873962 -0.5941008718732 -2.8409867275782 -1.5955845920106 0.3057605219875616 0.4471574428035519 -2.3948154680205 0.4993472235113314 -1.8165549651515098 1.7727770272914 0.97850845422084 0.9910590544420901 1.8354311849864202 0.980573842870689 1.9825574861706 -0.5731363593446 -1.357572253070202 1.402703159507120 -0.0878118591825460 -8.7316534217086 -2.706223977781183 2.153200879812 -4.544706432067 -3.773227294597 -5.2286823694784 13.037946041793 1.7202103282163197 19.7681095150772 5.40206151427628 -6.348151222247558 3.438287772795761 5.441749584497 4.499349461162 4.350144717674730 -4.2139513732969 8.442178323792 -3.107887286757 1.0790939930321 0.0492704594651732 -8.5208100838250]
Objective Function	6.33451
f-counts	1.267.327
Mesh size	$8.128e^{-13}$
Iterations	4
Maximum constraint	0.0001242

TABLE 6.7: This table shows the results of the of the ”test” case in *pattern search*.

Then, follows the table that introduces to the reader the accuracy of the optimization results, in terms of approximating the constraints boundaries.

Constraints	
Distance between cylinders was chosen to be $d_{i,j} \geq 0.1$	$d_{1,2} = 2.2871$
	$d_{1,3} = 2.7152$
	$d_{1,4} = 2.3672$
	$d_{1,5} = 3.2322$
	$d_{1,6} = 3.3878$
	$d_{1,7} = 2.0836$
	$d_{2,3} = 3.2515$
	$d_{2,4} = 3.4722$
	$d_{2,5} = 5.2142$
	$d_{2,6} = 4.9439$
	$d_{2,7} = 4.1800$
	$d_{3,4} = 4.2006$
	$d_{3,5} = 3.7051$
	$d_{3,6} = 1.8889$
	$d_{3,7} = 2.3057$
	$d_{4,5} = 2.6081$
	$d_{4,6} = 4.2803$
	$d_{4,7} = 3.9682$
	$d_{5,6} = 2.3556$
	$d_{5,7} = 2.8201$
	$d_{6,7} = 2.0569$
Condition number was chosen to be $\kappa(D(r,\hat{F})) \leq 10$	$\kappa(D(r,\hat{F})) = 3.9784$
Singular values were chosen to be $\sigma(D) \geq 0.01$	2.6587
	2.2869
	1.6154
	1.3686
	0.9007
	0.6683

TABLE 6.8: This table shows how accurately *pattern search* approximated the constraints that were inserted in "test" case.

So, the $D(r,\hat{F})$ of this case is:

$$D(r, \hat{F}) = \begin{bmatrix} -0.9297 & -0.5761 & 0.5491 & 0.5991 & 0.6562 & -0.4242 \\ -0.2882 & -0.4783 & 0.0725 & -0.7040 & 0.5425 & 0.8498 \\ 0.2293 & -0.6628 & 0.8326 & 0.3813 & 0.5245 & -0.3128 \\ 0.2208 & -1.1199 & -0.5458 & -0.5554 & 0.4759 & 0.1332 \\ -0.2894 & -0.5255 & 1.5696 & -0.8717 & -0.6911 & -0.5499 \\ 0.5316 & 1.3525 & 0.2234 & -0.7368 & 0.1194 & -1.6743 \end{bmatrix}$$

A full ranked matrix of $rank(D(r, \hat{F})) = 6$.

The results of this "run-test" approximately identifies a "tighter" searching space for the *Simulation run* that follows, despite the fact that these two use several different parameters. The searching space of this "test" run was defined for the position vectors from -10 to 10 (1×21 matrix) and for the direction vectors from -30 to 30 (1×21 matrix). Every value is stated in S.I.

Underline in here that the result above might not satisfies the boundaries of the constraints, as modified for the *Simulation run*, but this does not form a problem since the *pattern search* does not require a starting point that satisfies all the constraints. In every case, the results of the "run-test" are giving a starting value that will impel the algorithm to converge faster from simple choosing an arbitrary.

First Simulation Results

At this point of the thesis will be mentioned the results from the first simulation concerning the APC $8 \times 3.8SF$ propeller and the Scorpion SII-2212-1070 motor. The results of this first Simulation will incorporate the aerodynamic effects from the experiment. Also as said the starting point of this run will be the design variables that produced from the "test" case. The searching space in the Simulation will be defined from -5 to 5 for every design variable.

	fsolve
Options	Display: off
Starting value	$s.p. = \begin{bmatrix} 0 & 0 & 0 \end{bmatrix}$
Called function	*fteuler*

	fmincon
Options	Display: off, Algorithm: active-set, MaxFunEvals: 1000, MaxIter: 1000
Starting value of the optimization	$s.p. = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \end{bmatrix}$
Aerodynamic effect	Based on the experiment
Called functions	*domi2.m, fteuler.m, solver.m, distance.m, shape.m*

	patternsearch
Basic options	Poll method: GPS Positive basins 2N, Complete poll: on, Polling order: Random, MaxIter: 1000
	Complete search: on, Search method: Latin hypercube
	Default Mesh size options, Accelerator: off, Scale: off
	Constraint parameters default
	Cache: off
	Mesh tolerance: $1e^{-6}$, Max iterations: Default, Max function evaluations: $12 \cdot 10^6$
	Nonlinear constraint tolerance: $1e^{-15}$
Called functions	*cyl_d.m, volume.m*

TABLE 6.9: This table gathers the necessary options/parameters chosen for the First Simulation.

Starting value of the optimization a 1×42 matrix	s.p.= [-1.1156943627868698 -0.9175989960313365 -0.0360989038773962 -0.5941008871873029 -2.84098672757822 -1.59558459201063 0.305760521987616 0.447157442803519 -2.394815466880205 0.499347223511314 -1.8165549651515098 1.772770272729144 0.9785845422084 0.9910590544420901 1.835431184984202 0.980573842870689 1.98255748617061 -0.573136359344668 -1.3575722530707202 1.4027031595071202 -0.08781185918254603 -8.73165342170886 -2.70622397781183 2.15320087981281 -4.54470643206723 -3.77322729459796 -5.22868236947841 13.03794604179358 1.7202103282163197 19.7681095150772 5.40206151427628 -6.34815122247558 3.43828772795761 5.44174958449786 4.49934946116257 4.3501447176747305 -4.21395137329692 8.44217832379226 -3.10788728675723 1.07909399303021 0.0492704594651732 -8.520810083825051]
Elapsed time	2 days
PC properties	Intel(R) Core(TM)2 Quad CPU 99550 @2.83GHz, RAM: 2.00GB

TABLE 6.10: This is a sequel table of the previous with extra parameters and information.

Design variables after the optimization, a 1×42 matrix	[-1.1000196879968853 -0.9256537162085784 -0.036098903877396 -0.5941008871873329 -2.8409867275782 -1.595584592010063 0.3057605219877616 0.4471574428035119 -2.3948154688020205 0.4993472235113144 -1.816554965151509 1.772777027729144 0.9785084542084 0.99105905444420901 1.8354311849864202 0.98057384287061689 1.9825574861706061 -0.5731363593442668 -1.35757225307072020 1.40270315950712002 3.9120659512956646 -3.762617187032848 -4.807888598718491 0.6378202366283992 -4.4197046186952552 -2.27322729457979606 -4.2286823647841 3.0457585417358 1.7202103282163197 2.7544986771188325 1.5270615142628 -3.363776222247558 3.4382882047984062 3.4461512646863023 4.4759090905770706 4.4753316415660365 -2.89779475260000758 4.81717587085649 -2.6078891941058626 1.0790939930321 0.04927045946511732 -3.520809845406472]
Objective Function	7.44329
f-counts	89970
Mesh size	$7.244e^{-22}$
Iterations	7
Maximum constraint	$3.048e^{-9}$

TABLE 6.11: This is a table that shows the results of the First Simulation

The above tables show that the chosen starting point produced results, with the same computational power, for this Simulation with *pattern search* in only 2 days in comparison to the "test" case where 30 days were needed. Also someone can understand that the accuracy of the results in approximating the boundaries inserted from the user is very high, as seen in maximum constraint $3.048e^{-9}$. However, a matter of further search arises due to the high value of the objective function. That is depicted also to the increased values of the distances between the thrusters which implies automatically high body volume. An other drawback of this solution is the condition number which is also far form the value the Design problem requires.
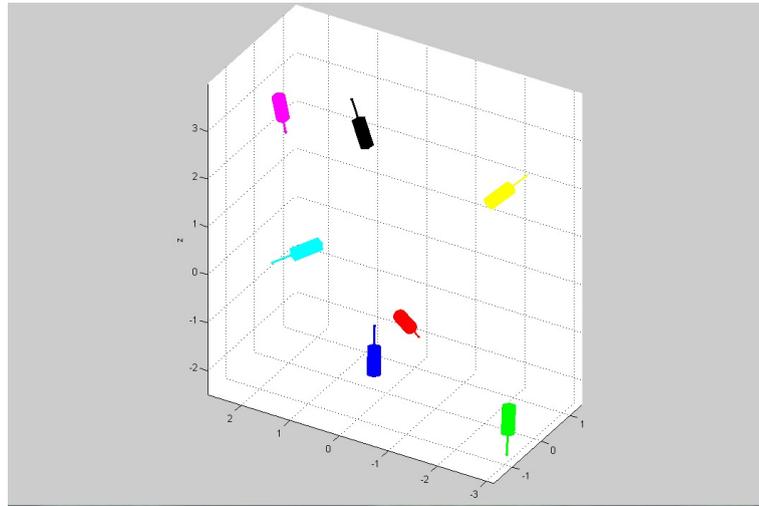
FIGURE 6.8: A figure that shows the allocation of thrusters as resulted from the first Simulation, using *plotstructure.m* function.

The figure 6.8 verifies that the volume of the Aerial Manipulator as resulted from the first Simulation is high. The function *plotstructure.m* apart from depicting the position of each thruster, shows also the direction using a vector.

The table 6.12 that follows has the values of the constraints. Notice that was chosen $d_{i,j} \geq 0.01$ because of the smallest aerodynamic shape introduced from the experiment and of course for accuracy reasons.

Constraints	
Distance between cylinders was chosen to be $d_{i,j} \geq 0.01$	$d_{1,2} = 2.1178$
	$d_{1,3} = 2.8331$
	$d_{1,4} = 2.3636$
	$d_{1,5} = 3.2236$
	$d_{1,6} = 3.4424$
	$d_{1,7} = 3.9902$
	$d_{2,3} = 3.2452$
	$d_{2,4} = 3.4715$
	$d_{2,5} = 5.2210$
	$d_{2,6} = 4.9321$
	$d_{2,7} = 6.4219$
	$d_{3,4} = 4.3797$
	$d_{3,5} = 3.7593$
	$d_{3,6} = 1.8653$
	$d_{3,7} = 5.7606$
	$d_{4,5} = 2.6121$
	$d_{4,6} = 4.3004$
	$d_{4,7} = 3.7835$
	$d_{5,6} = 2.3604$
	$d_{5,7} = 2.5421$
	$d_{6,7} = 4.4583$
Condition number was chosen to be $\kappa(D(r,\hat{F})) \leq 10$	$\kappa(D(r,\hat{F})) = 9.4091$
Singular values were chosen to be $\sigma(D) \geq 0.01$	2.9845
	2.5819
	1.5026
	1.2303
	0.7082
	0.3172
Equality constraints	$1.0e^{-8}*$
	0.3048
	0.0000
	0.0000
	0.0000
	0.0000
	0.0011

TABLE 6.12: This table shows the accuracy in approximating the constraints of the first Simulation in *pattern search*

Reducing the volume of the structure

As said the previous solution produces results with high body volume. In order to overcome this, was chosen to start a new optimization with the same options and parameters set in *pattern search* but with starting point the design variables resulted above.

Starting value of the optimization a 1×42 matrix	s.p.= [-1.1000196879968853 -0.9256537162085784 -0.036098903773962
	-0.5941008871873229 -2.8409867275722 -1.5955845920163
	0.3057605219876616 0.4471574428035195 -2.39481546880205
	0.4993472235113114 -1.8165549651515098 1.772777027729144
	0.9785084542208408 0.9910590544420901 1.835431184986402
	0.9805738428706889 1.9825574861706061 -0.5731363593444668
	-1.357572253070702 1.4027031595071202 3.9120659512956646
	-3.762617187032845 -4.807888598718491 0.6378202366283992
	-4.4197064618695552 -2.27322729459796606 -4.228682369478411
	3.04575854179358 1.7202103282163197 2.7544986771188325
	1.5270615142762808 -3.3637762224755808 3.4382882047984062
	3.4461512648630237 4.4759090905707006 4.475331641566036505
	-2.8977947526007587 4.817173587085649 -2.607889194105862606
	1.0790939930321 0.0492704594651732 -3.520809845406472]

Elapsed time	20 days
PC properties	Intel(R) Core(TM)2 Quad CPU 9550 @2.83GHz, RAM: 2.00GB

TABLE 6.13: This table incorporates the starting value, the elapsed time and the PC properties that used for the optimization of reducing the volume.

Design variables after the optimization, a 1×42 matrix	[-0.11458787156775818 -0.048557294191957756 -0.020698971018000075 0.030410831562670992 -0.21558904538882785 -0.15808459201062997 0.24206774351140925 0.197155531729959588 -0.08201155774593882 0.123610957136722243 -0.0665549651515150983 0.21027702729144004 0.193850060398822296 0.11510919482294946 0.27168644069866601 0.23032970224568905 0.107557478720029 -0.26057532418848418 -0.23127048337491996 0.21425975866096625 -0.046296828553583 -2.9402369584080626 -4.9915516632432929 1.8142050094900375 -4.98980341684755055 -1.74369039169345 -1.9965022681232703 1.92370322223101007 2.50162685800588867 2.909197031934824 1.15182724194452806 -3.08108000091919455 3.0024176386654579 3.09093843762808086 3.75989101020085874 2.894427936187377 -0.63760732229225793 3.070855428625777774 -4.764723758757557 1.16401932382362005 0.193050266226418 -4.99998365218305]

Objective Function	0.784301
f-counts	1132107
Mesh size	$7.586e^{-16}$
Iterations	19
Maximum constraint	$1.224e^{-8}$

TABLE 6.14: This table shows the results from the optimization which aims to reduce the volume of the Aerial Manipulator.

Constraints	
Distance between cylinders was chosen to be $-d_{i,j} + 0.01 \leq 0$	-0.0259
	- 0.2142
	- 0.1172
	- 0.2473
	- 0.2110
	- 0.0239
	- 0.2511
	- 0.1417
	- 0.3721
	- 0.1300
	- 0.2843
	- 0.1990
	- 0.0000
	- 0.0000
	- 5.7606
	- 2.6121
	- 4.3004
	- 3.7835
	- 2.3604
	- 2.5421
	- 4.4583
Condition number was chosen to be $\kappa(D(r, \hat{F})) \leq 10$	$\kappa(D(r, \hat{F})) = 9.4091$
Singular values were chosen to be $\sigma(D) \geq 0.01$	1.8114
	1.5955
	0.5203
	0.2795
	0.2149
	0.1811
Equality constraints	$1.0e^{-7}*$
	0.0004
	0.0000
	0.1224
	0.0000
	0.0000
	0.0000

TABLE 6.15: This table shows the accuracy in approximating the constraints of the stated low volume-structure optimization.
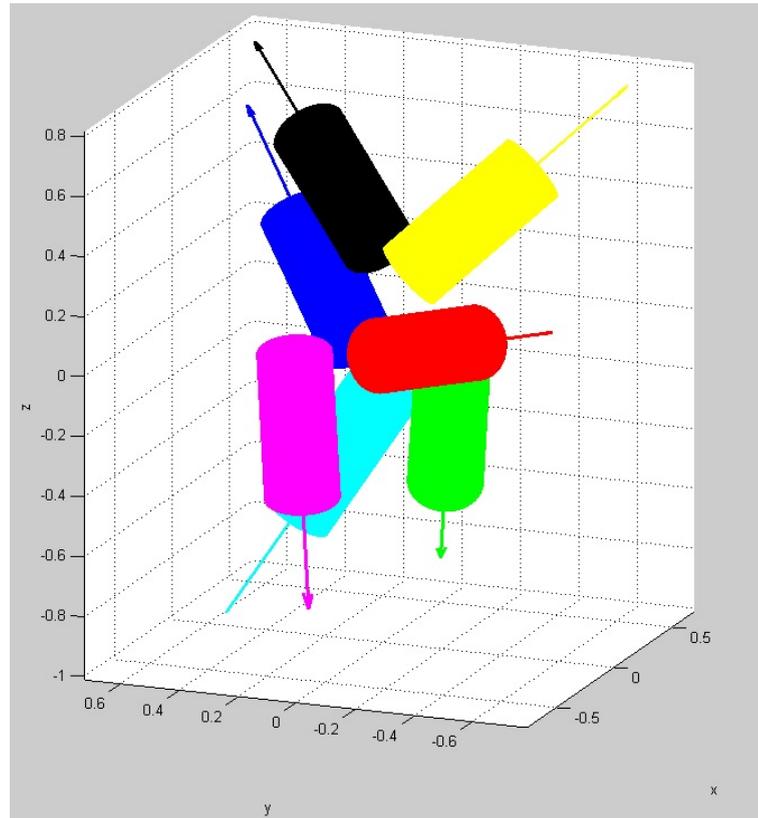
FIGURE 6.9: A figure that shows the allocation of thrusters as resulted from the first Simulation in terms of reducing the volume, using *plotstructure.m* function.

From comparing the results of the first Simulation (both with high and with low volume structure), someone can understand that the optimization problem described in Chapter 4 with the equation (4.33) has dis-continuous feasible regions. That was confirmed from the large deviations of the objective function values, which values in the last results (of reducing the volume) are ten and more times less than the first results. Thus, here one can see the difficulty that arises when solving this optimization problem, which is the best possible approximation of the global optimum.

Reducing the condition number

Previously said that the first solution with the high volume has condition number with value far from the requirements of the Design problem.

The starting value to this optimization run will be the result to the design variables of the reduced volume above

Starting value of the optimization a 1×42 matrix	s.p.= [-0.11458787156775818 -0.04855729419195775 -0.020698971018000075
	0.030410831562670992 -0.21558904538827 -0.15808459201062997
	0.242067743514092 0.197155531729595 -0.082011557745938
	0.12361095713672243 -0.06655496515150983 0.21027702729144004
	0.193850060398229 0.11510919482294946 0.271686440698660
	0.23032970224568905 0.107557478720029 -0.260575324188418
	-0.23127048337491996 0.2142597586609662 -0.0462968285535883
	-2.940236958480626 -4.9915516632432 1.8142050094900375
	-4.98980341684755 -1.74369039169345 -1.99650262812327
	1.92370322231010 2.50162685800588 2.90919703193482
	1.151827419452806 -3.08108000919194 3.002147638665479
	3.090938437628086 3.759891010208587 2.894427936187377
	-0.637607322292579 3.07085542862577 -4.76472375875795
	1.16401932382362 0.19305026622641 -4.99998365218230]

Elapsed time	12 days
PC properties	Intel(R) Core(TM)2 Quad CPU 99550 @2.83GHz, RAM: 2.00GB

TABLE 6.16: This table incorporates the starting value, the elapsed time and the PC properties that used for the optimization of reducing the condition number.

Design variables after the optimization, a 1×42 matrix	[0.50963813662279417 -0.17834282417040104 0.034874788331633155 0.099356420709979434 -0.48001361282229946 -0.16863671308329486 0.109959289943114297 0.029370168046639833 -0.23570903683832345 -0.39545561025216003 0.29087261153403029 0.007841940933571934 0.21727336457715918 0.42337517242032874 -0.30034668121062325 -0.25597207603648986 -0.51918904799908583 -0.04493476271121288 0.60209502008867539 $1.204960419318726e^{-4}$ -0.07240298526924427 0.45218343598779868 1.76995290125664685 3.835724056783005 -2.21614707031427783 -2.88757428418393225 1.90918958063440654 0.42356148241044345 -4.99999526614569694 -1.58605755453621134 0.52582298333222474 -2.48520139227737377 4.251852976510117 1.84427246245360009 1.385295906310704 -1.15165570690088504 -0.38478043135878526 0.7915948905878674 -1.52141115633528711 -0.54593734646382066 1.63571960181919197 -1.33482735256662808]
Objective Function	1.3757
f-counts	852154
Mesh size	$8.71e^{-10}$
Iterations	17
Maximum constraint	0.000148

TABLE 6.17: This table shows the results from the optimization which aims to reduce the condition number.

Constraints	
Distance between cylinders was chosen to be $-d_{i,j} + 0.01 \leq 0$	- 0.2993
	- 0.2508
	- 0.6429
	- 0.4846
	- 0.5784
	- 0.0003
	- 0.0003
	- 0.6335
	- 0.7042
	- 0.0001
	- 0.4631
	- 0.3660
	- 0.1952
	- 0.2119
	- 0.1978
	- 0.4870
	- 0.4773
	- 0.6942
	- 0.7043
	- 0.0000
	- 0.7429
Condition number was chosen to be $\kappa(D(r,\hat{F})) \leq 6$	$\kappa(D(r,\hat{F})) = 6$
Singular values were chosen to be $\sigma(D) \geq 0.01$	1.9071
	1.4437
	0.8486
	0.7042
	0.3490
	0.3179
Equality constraints	$1.0e^{-9}*$
	0.0937
	0.1196
	0.0675
	0.0225
	0.1532
	0.0935

TABLE 6.18: This table shows the accuracy in approximating the constraints of the above optimization.
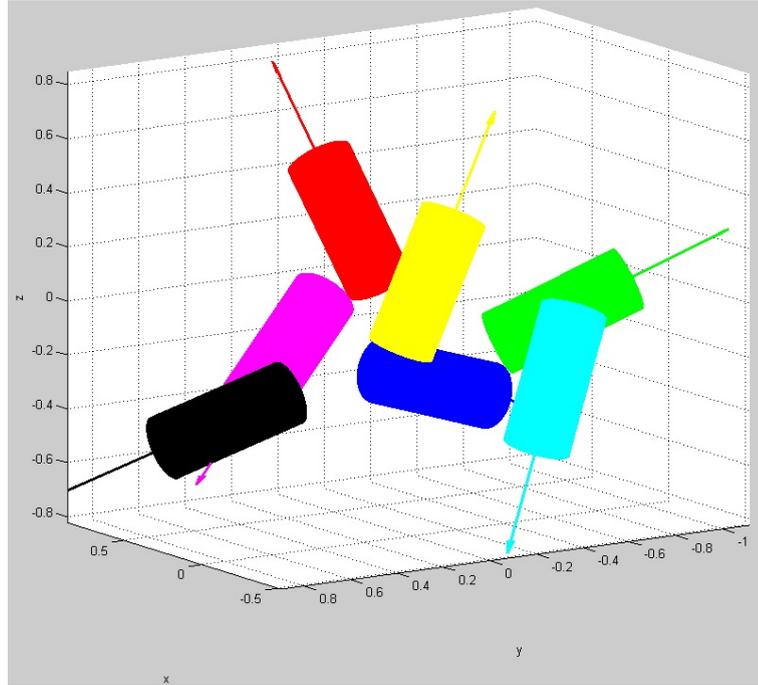
FIGURE 6.10: A figure that shows the allocation of thrusters as resulted from the above optimization in reducing the condition number, using *plotstructure.m* function.

In here, the $D(r, \hat{F})$ is:

$$D(r, \hat{F}) = \begin{bmatrix} 0.1064 & -0.5392 & 0.0805 & 0.1062 & 0.7154 & -0.2189 \\ 0.4166 & -0.7025 & -0.9501 & -0.5018 & 0.5373 & 0.4504 \\ 0.9028 & 0.4645 & -0.3014 & 0.8585 & -0.4467 & -0.8656 \\ 0.1755 & 0.3414 & 0.2328 & -0.2536 & 0.0277 & -0.4696 \\ 0.4564 & -0.0448 & -0.0142 & -0.3403 & 0.1178 & 0.2117 \\ -0.2313 & 0.3286 & 0.1068 & -0.1675 & 0.1861 & 0.2289 \end{bmatrix}$$

a full ranked matrix.

This optimization that elapsed in only 12 days produced several interesting results which approximate sufficiently the boundaries introduced from the user. Of course, all the constraints were satisfied and produced results that correspond to a low volume body structure with a low also value to the condition number. Furthermore, the solution raised from this optimization will be the one with which will be evaluated the actuating force and torque the Aerial Manipulator is capable to apply for the specific match of motor and propeller.

In this point, it should be recalled the equation (4.10) which translates the preferable actuating force and torque (right part of the equation) into the necessary thrust each

rotor must produce (left part of the equation). The $D(r, \hat{F})$ matrix was found above and the W_R matrix has the positions of the rotors, the weights of the body structure and the values of the actuating force/torque with the form:

$$\mathbf{W_R} \hat{=} \begin{bmatrix} \mathbf{F_{act}} - n \cdot \mathbf{w} - \mathbf{w_s} \\ \\ \mathbf{M_{act}} - \mu \cdot (\mathbf{F_{act}} - n \cdot \mathbf{w} - \mathbf{w_s}) - \left(\sum_{i=1}^{n} \mathbf{r_i} \right) \times \mathbf{w} - \mathbf{r_s} \times \mathbf{w_s} \end{bmatrix} \in \Re^6 \qquad (6.8)$$

Generally, from the Model made in Chapter 3, it is known that:

$$Q = C_Q \cdot \rho \cdot \pi \cdot R^5 \cdot \Omega^2 \qquad (6.9)$$

and

$$T = C_T \cdot \rho \cdot \pi \cdot R^4 \cdot \Omega^2 \qquad (6.10)$$

So, by dividing these two equations it is obtained:

$$\frac{Q}{T} = \frac{C_Q}{C_T} \cdot R \implies Q = \frac{C_Q}{C_T} \cdot R \cdot T \qquad (6.11)$$

which practically, in here, the term $\frac{C_Q}{C_T} \cdot R$ correspond to the μ symbol introduced in Chapter 4 that refers to the relationship between the thrust force and the reaction-type torque. In that way,

$$\mu = \frac{C_Q}{C_T} \cdot R \qquad (6.12)$$

It was calculated earlier in this Chapter that

$$C_T = 0.0074, \quad C_Q = 0.009$$

Thus, with $R \approx 0.124m$,

$$\mu = 0.1508$$

Then follows a Table that gathers estimated values of the weights to the components that structure the Aerial Manipulator. These values are typical values that were driven mainly from the information each company offers to a possible buyer. The frame weight corresponds to an approximation structured in *Solidworks* environment and the battery is a *NEU Energy* 2100 mAh capacity.

Weights (kg)	
Rotor (motor/propeller)	≈ 0.280
Frame	≈ 0.260
Battery	≈ 0.250
Several electronic components	≈ 0.150

TABLE 6.19: This table shows estimated values to the components of the Aerial Manipulator.

Therefore, by using the above values and inserting the preferable actuating force and torque ($\mathbf{F_{act}}$, $\mathbf{M_{act}}$) someone can determine the matrix W_R. Also by using the equation

$$D(r, \hat{F}) \cdot \lambda = \mathbf{W_R} \tag{6.13}$$

can identify the propulsion effort (see Chapter 4) λ since $D(r, \hat{F})$ matrix is invertible. In that way, the maximum thrust force and torque the Aerial Manipulator can apply is approximately $5N$ (without applying any torque simultaneously) and $4Nm$ respectively (without applying any thrust simultaneously). Which is translated in approximately 0.5 kg lifting ability beyond its own weight, which is close to 1 kg.

6.6 Second Simulation

Clearly the need of producing more lifting ability via the end-effector implies the search for different matches between motors and propellers.

6.6.1 Choosing propeller and motor

As mentioned, the Aerial Manipulator is being characterized by the terms of originality and innovation. Using again as a benchmark the *Asctec Firefly*, the propellers that will be chosen in this Second Simulation will have approximately the same dimensions with those used above. More specifically, again in here the propeller will be 8 inches, designed

from the *APC* company. Searching for a propeller close to these dimensions in *APC* list products someone can see that for *slow fly* UAV is the $8 \times 4.7SF$ with the product code LP08047SF (where SF means *slow fly*). In here was chosen a propeller again with 8 inches diameter but 4.7 pitch. That happens because, according to the information given from the *APC*, this propeller produces higher values of thrust force than the $8 \times 3.8SF$ used in the First Simulation. Not to forget also that, the 8 inches diameter was kept constant due to the limited experimental data concerning the aerodynamic effects which are bounded in these dimensions.

As far as the motor is concern there were few information available in choosing the ideal match with this particular *APC* propeller. In this Simulation will be used a motor powered by the *Neu Motors*, a well known company with extensive experience in this field, offering a variety of components.

In the following table are gathered some of the motors that match with the $8 \times 4.7SF$ *APC* propeller, introducing to the reader the efficiency percentages, the maximum produced thrust force in specific rpm and the model of the *Nue* motor.

Neu Motors model	Efficiency percentage(%)	Produced thrust force (kg)	RPM	Weight (kg)
		8 × 4.7SF APC propeller		
NEU 1107/6D -1900	76.7	1.5	15053	0.09
NEU 1105/2.5Y -3800	58	1.8	18560	0.06
NEU 1105/3Y -3300	59.3	2.075	17625	0.06
NEU 1107/2.5Y -2750	73.7	2.3	18567	0.09
NEU 1110/2.5Y -1814	83	1.62	15593	0.11
NEU 1110/2Y -2250	83.9 (max eff. 89.8)	2.2	18297	0.11
NEU 1115/1.5Y -1900	86.9 (max eff. 89.1)	1.9	16952	0.16
NEU 1415/1.5Y -1500	91.8 (max eff. 92.2)	1.5	15041	0.29
NEU 1509/2Y -1820	87.7 (max eff. 89.4)	1.8	16571	0.21
NEU 1512/1.5Y -1900	91.7 (max eff. 92.1)	2.0	17686	0.28
NEU 1512/2.5D -2000	90.9 (max eff. 91.5)	2.2	18200	0.28
NEU 1512/3D -1700	92.3 (max eff. 92.5)	1.8	16440	0.28
NEU 1902/2Y -2035	86.3 (max eff. 90.2)	2.0	17550	0.12

TABLE 6.20: In this table are gathered the *Neu Motors* that match with the 8 × 4.7SF APC propeller. Still here referred, the efficiency percentage of each match (in comparison to the maximum efficiency), the produced maximum thrust force in specific rpm and the motor weight.

Oriented to increasing the overall produced thrust force from a rotor/thruster, in this Simulation will be used the NEU 1902/2Y - 2035 motor. That is logical also, if someone consider the efficiency percentage and the motor weight, since there is no other motor from this company that cooperates with such an efficiency and produces that value of thrust force while having low weight. Notice in here that the motors gathered in the

table above, respect the inserted - by the propeller - limitations related to the maximum rpm, which in $8 \times 4.7SF$ are the 19000 rpm.

6.6.2 Aerodynamic effects

Acting with the same way as previously in First Simulation, the aerodynamic effects will be driven from the experiment made in the CSL upon *Asctec Firefly*. Here becomes clear the reason of choosing a propeller that has approximately equal dimensions to those used in *Asctec Firefly*. Although as it was seen the propeller from the APC does not have exactly the same dimension, these two propellers have the same diameter (8 inches) which is important. In any case though, incorporating the aerodynamic effects of the experiment will imply a more realistic Simulation results from just selecting an arbitrary cylindrical shape.

In the above subsection, it was shown that the $8 \times 4.7SF$ APC propeller accompanied with the Neu Motor NEU 1902/2Y - 2035 produces at 17550 rpm, 2.0 kilograms of force.

The measurements from the experiment that are closer to the 17550 rpm, in which the maximum thrust force of the chosen APC propeller is observed, are those that follow.
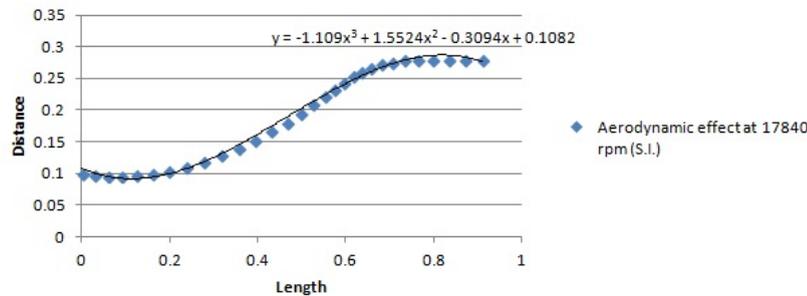
FIGURE 6.11: This figure shows the aerodynamic effect at 17840 rpm of *Asctec Firefly* experiment. The blue line shows the measurements and on top of this is the 3nd order curve that approximates this line. The x axis shows the length of the aerodynamic effect of the exit flow and the y axis shows the distance of the effect measured from the rotation axis of the rotor. The values are expressed in S.I.

This figure shows with the blue line the measurements that gathered from the experiment of the *Asctec Firefly* in 17840rpm and these are the closest measurements that can be compared to the APC propeller, which in 17550 rpm produces the thrust force. The x axis shows the length of the aerodynamic effect of the exit flow and in order to understand this someone should think as if the rotor/blade is at $x = 0$. On the other side, y axis shows the distance of the effect measured from the rotation axis of the blade, where in

$x = 0$, $y = \frac{8''}{2} = \frac{20.32cm}{2} \approx 0.10m(S.I.)$ which is approximately the blade radius (because also an offset exist).

Moreover, as in First Simulation, here must be shown that the aerodynamic effect which is being created at the entrance of the air flow to the rotor. The only information given from the experiment in 17840 rpm, is that this length extents $0.06m$ above the rotor and there were made no other measurements so as to approximate the curve. For simplification reasons will be thought as if the curve of the figure 6.11 starts from $x = -0.06m$ and stops again at the $x = 0.91m$ so as to incorporate the aerodynamic effects at the entrance of the rotor.

In that way, the the aerodynamic effects of the air flow throughout the rotor are extended from $x = -0.06m$ to $x = 0.91m$ and the curve that approximates these effects is the 3nd order equation

$$y = -1.109 \cdot x^3 + 1.5524 \cdot x^2 - 0.3094 \cdot x + 0.1082 \quad (S.I.) \tag{6.14}$$

Based on the above equation here must generate again the three dimensional (3D) shape that will reflect the aerodynamic effects of the chosen APC propeller. In order to do achieve this must rotate the equation (6.14) from a fixed axis and follow the rules of producing symmetrical rotational shapes.

Thus, the aerodynamic effects will be introduced to this analysis with the following equation:

$$-0.06 \leq x' \leq 0.91$$
$$y'^2 + z'^2 \leq (-1.109 \cdot x'^3 + 1.5524 \cdot x'^2 - 0.3094 \cdot x' + 0.1082)^2 \quad (S.I.) \tag{6.15}$$

The equation is written in correspondence to the equation (5.4) and so it is expressed to the body coordinates of the shape and hence the tuna (') to the variables of the equations. Also, the length of the aerodynamic shape is specified in x' axis as was followed with the equations of the cylinder.

It must be mentioned that the above equation was a result of the parametric version bellow.

$$z = \sin(v) \cdot (-1.109 \cdot x^3 + 1.5524 \cdot x^2 - 0.3094 \cdot x + 0.1082)$$
$$y = \cos(v) \cdot (-1.109 \cdot x^3 + 1.5524 \cdot x^2 - 0.3094 \cdot x + 0.1082) \tag{6.16}$$
$$x = u$$

where

$$-0.06 \leq u \leq 0.91$$
$$0 \leq v \leq 2 \cdot \pi$$

This aerodynamic shape in Matlab environment is shown in the next figure.
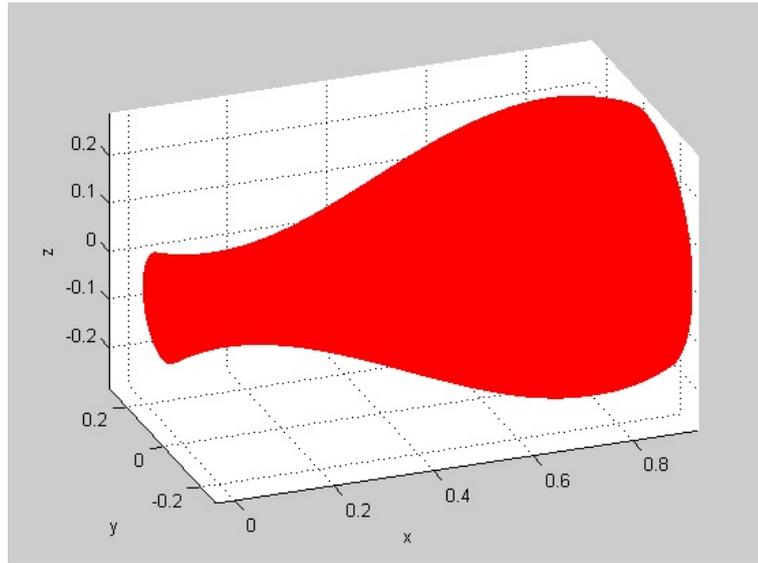
FIGURE 6.12: This figure shows the shape of the 3D aerodynamic effect in Matlab environment, based on *Asctec Firefly* experiment at 17840 rpm.

6.6.3 Determine thrust/torque and their coefficients

In comparison to the First Simulation, the information given from the APC concerning the $8 \times 4.7SF$ propeller is better distributed in the rpm spectrum. Therefore, values for the thrust force and torque are given in every one thousand rpm (see Appendix C).

Remember again in here the characteristic of the Aerial Manipulator, which is the ability to interact with the environment throughout the end-effector and produce corresponding actuating force (F_{act}) and torque (M_{act}). That means that with every particular demand for actuating force (F_{act}) and torque (M_{act}), every thruster will be obliged to rotate in specific rpm that might not be in the performance data APC offers and produce specific thrust force (respecting always the manufacturing limitations). And of course the same happens with the anti-torque and their coefficients, since these are interdependent (see Chapter 3).

However, for this particular $8 \times 4.7SF$ propeller there was found no relationship between the thrust force and the rpm based on experimental data in relation to motors, as in First Simulation. Practically this is not a matter for further search since the APC offers the necessary information needed but this is an opportunity to re-confirm the accuracy

of the Model in Chapter 3. Notice one more time the arising issue associated with the lack of experiments concerning measurements of thrust force and torque of specific propellers - motors.

As was stated several times before this thesis focuses only in Hover Flight state, which implies that during the calculation of thrust/torque the speed ratios of Chapter 3 will be equal to zero. In addition to that, in order to confirm the accuracy of the previously proposed Model, the methodology that calculates the thrust and torque of the First Simulation, will be applied also in here.

In this point it should be noticed that was used *culc_lamda_ind_and_CT_CQ.m*, a function in Matlab environment that takes as inputs the a, b, c_{d0}, c_{d1}, c_{d2}, θ_0, θ_1, c, K_1, K_2, e, R (radius of the rotor) and produces the C_T and C_Q coefficients. The values of these inputs are gathered in the table that follows.

Symbol	Value
θ_0	$0.0489 rad (\approx 2.80^o)$
θ_1	$-0.086 rad (\approx -4.93^o)$
a	5.5 (for Hover state)
c_{d0}	0.05
c_{d1}	≈ 0.7
c_{d2}	≈ 0
c	$\approx 0.02m$
R	0.124 m (radius of the rotor)
e	$0.1 \cdot R$
K_1	$\left(\frac{9}{5}\right)^{\frac{1}{4}}$
b	2

TABLE 6.21: Values of the required parameters.

Then follows a table that gathers the thrust force as calculated from APC in hover state (without model speed) for the $8 \times 4.7SF$ propeller, accompanied with the thrust force produced from the methodology of the Chapter 3.

RPM	APC Thrust(N)	Theoretical Thrust(N)
4000	1.2	1.286360754
5000	1.9	2.009938678
6000	2.7	2.894311696
7000	3.7	3.939479809
8000	4.89	5.145443015
9000	6.2	6.512201316
10000	7.7	8.039754711
11000	9.3	9.728103201
12000	11.12	11.57724678
13000	13.18	13.58718546
14000	15.43	15.75791923
15000	17.8	18.0894481
16000	20.45	20.58177206
17000	22.9	23.23489112
18000	25.85	26.04880526
19000	28.9	29.02351451

TABLE 6.22: Values so as to produce the plotted curves of thrust force in figure 6.13

The elements of the above table will be plotted in the figure that follows, where it is clear that the thrust force proposed from the Model of the Chapter 3 approximates with high accuracy the thrust as given from the *APC*. In fact this approximation is made with less than 5% error ($C_T = 0.008$).
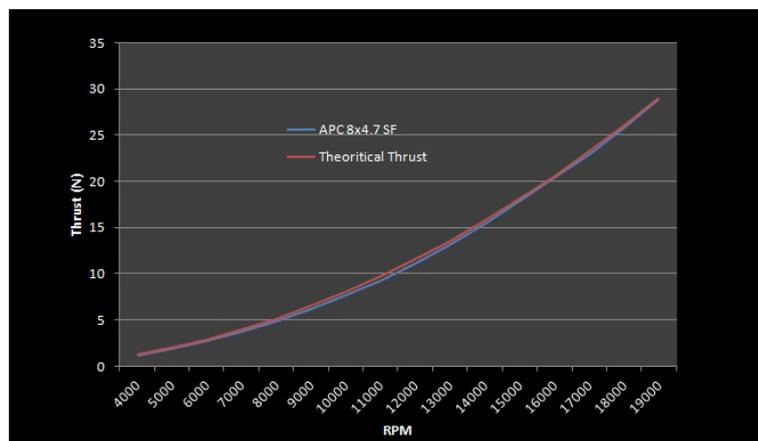
FIGURE 6.13: Comparison between the theoretical Thrust force and the one given from the *APC* against rpm, for the $8 \times 4.7SF$ propeller.

For completeness reasons follows the Theoretical Torque produced from the above methodology accompanied with the appropriate table (with $C_Q = 0.0095$).
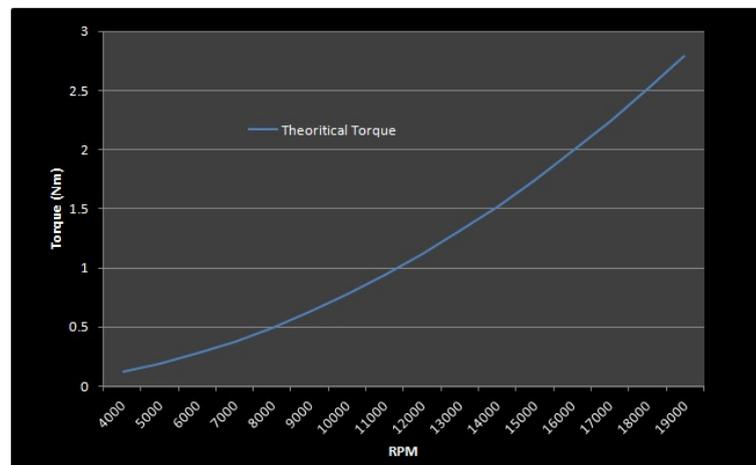
FIGURE 6.14: This is a figure that shows the Theoretical Torque produced from the Model against rpm, for the $8 \times 3.8SF$ APC propeller

RPM	Theoretical Torque($\mathbf{N \cdot m}$)
4000	0.187773602
5000	0.293396253
6000	0.422490604
7000	0.575056656
8000	0.751094408
9000	0.95060386
10000	1.173585012
11000	1.420037865
12000	1.689962418
13000	1.983358671
14000	2.300226624
15000	2.640566278
16000	3.004377632
17000	3.391660686
18000	3.80241544
19000	4.236641895

TABLE 6.23: Values so as to produce the plotted curve of torque.

6.6.4 Results

In correspondence with the previous Section in here will be presented the results from solving the optimization problem of the equation (4.33) through *pattern search*, based on the limitations introduced from the Second Simulation.

The starting value to this optimization will be the design variables of the reduced condition number, as resulted from the First Simulation (see the subsection)

Starting value of the optimization a 1×42 matrix	s.p.= [0.509638136627941 -0.178342824170401 0.034747883163315 5 0.099356420709794 34 -0.480013612822994 6 -0.168636713083294 86 0.109959289431429 7 0.029370168046639833 -0.235709036383234 5 -0.395455610252160 03 0.290872611534029 0.007841940933571934 0.217273364577159 18 0.423375172420328 74 -0.300346681210623 25 -0.255972076036489 86 -0.519189047990858 3 -0.044934762711212 88 0.602095020086753 9 1.204960419318723 6e^{-4} -0.072402985269244 27 0.452183435987986 8 1.769952901256468 5 3.835724056783500 5 -2.216147070314278 3 -2.887574284183922 5 1.909189580634465 4 0.423561482410443 45 -4.999995266145694 -1.586057554536211 0.525822983332247 4 -2.485201392277377 4.251852976510117 1.844272462453600 9 1.385295906310704 -1.151655706908850 4 -0.384780431358785 26 0.791594890587867 4 -1.521411115633528 7 -0.545937346463820 6 1.635719601819197 -1.334827352566280 8]
Elapsed time	16 days
PC properties	Intel(R) Core(TM)2 Quad CPU 99550 @2.83GHz, RAM: 2.00GB

TABLE 6.24: This table incorporates the starting value, the elapsed time and the PC properties that used for the Second Simulation.

Design variables after the optimization, a 1×42 matrix	[0.5096378982093626 -0.1783428241704104 -0.5204533423904127 0.09935642070979434 -0.2592481413645594 -0.1686367130832948 0.1099592894311429 -1.063131146453157 -0.2357090368382345 -0.3954091186292352 0.2908723731154499 0.007902499925266373 0.2170405483346665 0.4233751724203287 -0.298392960164755 -0.2559720760364898 -0.5151905300000746 -0.04493523954837086 0.6020785692047959 0.9369069092346809 -0.07240298526924427 0.3509878621812241 1.80123771036901 4.224997461858574 -1.724058752443184 -4.625979511845055 1.471266381295826 0.4235652971077090 -2.907558952516543 -1.590088974290239 3.390858040401185 0.4909605645097326 4.99999974889476 3.003544656362292 0.3853009131008651 -2.00267747357144 -4.931536537098595 4.228504804604591 -3.667535996118002 -1.558312051842186 1.635751311490217 -1.334762741131344]
Objective Function	2.00018
f-counts	1135069
Mesh size	$1e^{-9}$
Iterations	4
Maximum constraint	$9.953e^{-7}$

TABLE 6.25: This table shows the results from the optimization for Second Simulation.

Constraints	
Distance between cylinders was chosen to be $-d_{i,j} + 0.01 \leq 0$	-0.2260
	-0.7398
	-0.2566
	-0.2424
	-0.6300
	-0.4451
	0.0000
	-0.4899
	-0.4938
	0.0000
	-1.0627
	-1.2280
	-1.2339
	-0.4478
	-1.8085
	-0.5094
	-0.2829
	-0.6635
	-0.6950
	-0.3591
	-1.2126
Condition number was chosen to be $\kappa(D(r,\hat{F})) \leq 5$	$\kappa(D(r,\hat{F})) = 3.3615$
Singular values were chosen to be $\sigma(D) \geq 0.01$	1.6151
	1.4298
	1.2272
	0.7017
	0.5371
	0.4805
Equality constraints	$1.0e^{-6}*$
	0.0084
	0.0342
	0.0100
	0.0871
	0.0102
	0.1229

TABLE 6.26: This table shows the accuracy in approximating the constraints for the Second Simulation.

After the deposition of the optimization results, follows a figure that depicts the positions and the directions of the thrusters upon the Aerial Manipulator.
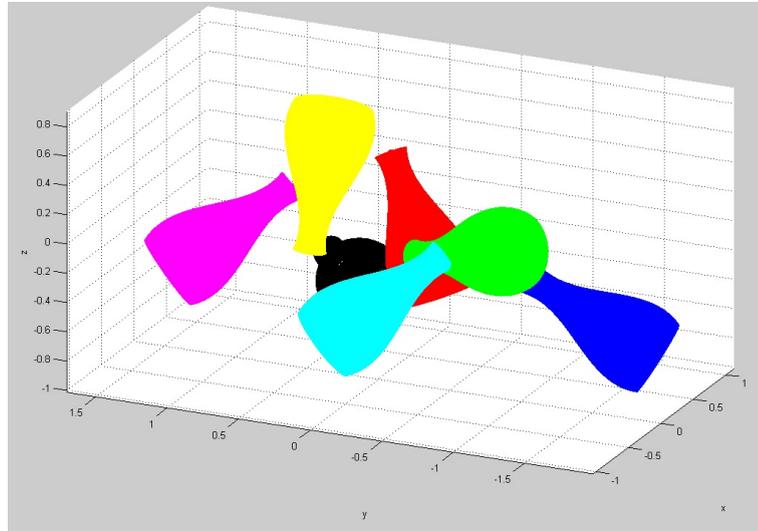
FIGURE 6.15: A figure that shows the allocation of thrusters as resulted for the Second Simulation, using *plotstructure.m* function.

So for this Simulation $D(r, \hat{F})$ is:

$$D(r, \hat{F}) = \begin{bmatrix} 0.0762 & -0.3347 & 0.1268 & 0.5594 & 0.8273 & -0.6612 \\ 0.3910 & -0.8980 & -0.8703 & 0.0810 & 0.1061 & 0.5667 \\ 0.9172 & 0.2857 & -0.4759 & 0.8249 & -0.5516 & -0.4916 \\ -0.0399 & 0.2255 & -0.3009 & -0.2393 & 0.2019 & -0.2787 \\ 0.5071 & -0.0281 & -0.0225 & -0.3306 & 0.1271 & 0.0961 \\ -0.2129 & 0.1760 & -0.0391 & 0.1948 & 0.3272 & 0.4857 \end{bmatrix}$$

a full ranked matrix.

In this optimization, of course, all the constraints were satisfied and produced results that correspond to a low volume body structure with a low also value to the condition number. The above raised solution will be the one with which will be evaluated the actuating force and torque the Aerial Manipulator is capable to apply for the specific match of motor and propeller.

Again, it should be remembered the equation (4.10) which translates the preferable actuating force and torque (right part of the equation) into the necessary thrust each rotor must produce (left part of the equation). The $D(r, \hat{F})$ matrix was found above and the W_R matrix has the positions of the rotors, the weights of the body structure and the values of the actuating force/torque with the form:

$$\mathbf{W_R} \hat{=} \begin{bmatrix} \mathbf{F_{act}} - n \cdot \mathbf{w} - \mathbf{w_s} \\ \\ \mathbf{M_{act}} - \mu \cdot (\mathbf{F_{act}} - n \cdot \mathbf{w} - \mathbf{w_s}) - (\sum_{i=1}^{n} \mathbf{r_i}) \times \mathbf{w} - \mathbf{r_s} \times \mathbf{w_s} \end{bmatrix} \in \Re^6 \quad (6.17)$$

From the Model made in Chapter 3:

$$Q = C_Q \cdot \rho \cdot \pi \cdot R^5 \cdot \Omega^2 \qquad (6.18)$$

and

$$T = C_T \cdot \rho \cdot \pi \cdot R^4 \cdot \Omega^2 \qquad (6.19)$$

So, by dividing these two equations it is obtained:

$$\frac{Q}{T} = \frac{C_Q}{C_T} \cdot R \implies Q = \frac{C_Q}{C_T} \cdot R \cdot T \qquad (6.20)$$

which practically, in here, the term $\frac{C_Q}{C_T} \cdot R$ correspond to the μ symbol introduced in Chapter 4 that refers to the relationship between the thrust force and the reaction-type torque. In that way,

$$\mu = \frac{C_Q}{C_T} \cdot R \qquad (6.21)$$

It was calculated earlier in this Chapter that

$$C_T = 0.008, \quad C_Q = 0.0095$$

Thus, with $R \approx 0.124m$,

$$\mu = 0.1473$$

Then follows again the Table that gathers estimated values of the weights to the components that structure the Aerial Manipulator.

Weights (kg)	
Rotor (motor/propeller)	≈ 0.12
Frame	≈ 0.260
Battery	≈ 0.250
Several electronic components	≈ 0.150

TABLE 6.27: This table shows estimated values to the components of the Aerial Manipulator for the Second Simulation.

Therefore, by using the above values and inserting the preferable actuating force and torque ($\mathbf{F_{act}}$, $\mathbf{M_{act}}$) someone can determine the matrix W_R. Also by using the equation

$$D(r, \hat{F}) \cdot \lambda = \mathbf{W_R} \tag{6.22}$$

can identify the propulsion effort (see Chapter 4) λ since $D(r, \hat{F})$ matrix is invertible. In that way, the maximum thrust force and torque the Aerial Manipulator can apply is approximately $30N$ and simultaneously $3Nm$. Which is translated in approximately 3 kg lifting ability beyond its own weight, which is approximately 1,500 kg.

6.7 Electronics box

An other crucial subject for investigation is the box that will contain the necessary electronic parts of the Aerial Manipulator. Two are the major points for consideration:

- the size/volume of this box and its own location upon the body structure

- the possible aerodynamic interaction with the rotors

Clearly dealing with the first obligation is quite trivial since the this box must contain the electronic parts. Commonly these parts are the following:

- battery

- electronics plate

- receiver

- sensors

- computer processor

from which the battery occupies the highest volume and has the biggest weight. It was measured in the CSL (Control System Laboratory) that a box with 6×12 (mm) length and width with 10mm height is sufficiently for containing all the appropriate electronic parts. Mainly these dimensions are result of the restrictions introduced from the battery due to its own volume. Moreover, it is important to consider the location of this box upon the body structure of the Aerial Manipulator. Optimally it should be located close to the center of mass which from definition is close to the center of the world coordinates. That happens in order to avoid the moments of inertia and to achieve better weight distribution in the structure.

On the other hand, as far the aerodynamic interaction is concern, the box must be considered to be placed correctly away from the air flow of each rotor as found in the previous two Sections.

Conclusively, for both solutions - of the two Simulations mentioned before (see Section 6.5,6.6) - exist sufficient volume almost in the center of mass of the structure so as to place the box ($6 \times 12 \times 10$ mm) with the electronics avoiding simultaneously the aerodynamic interaction with the rotors/thrusters. Nevertheless, the best approach to the matter of locating this box would be by incorporating it to the optimization problem, finding this way the optimal location and direction upon the structure in terms always to the aerodynamic interaction with the rotors. That also could be a matter of further research in the future.

Chapter 7

Conclusions

In this final Chapter it is necessary to make a synopsis of the Diploma thesis, propose certain extensions and future directions for further research and development on the Aerial Manipulator.

7.1 Reviewing the aims of the Diploma thesis

Although much work has been made in the field of the UAVs, here was proposed a completely new vehicle that has the characteristics of a rotary-wing air craft with the name *Aerial Manipulator*. An Aerial Unmanned Vehicle that will be able among others to apply to the environment actuating force and torque throughout an end-effector. Since it is an aerial vehicle any study or analysis before it begins will require accurate models of the flight characteristics which, as said, are based on the quadrotors. That was relevant because there is still not much published information about quadrotors, and those available are usually scattered and incomplete with several arbitrariness. Thus, special research was made in order to identify the appropriate flight model and incorporate - to an extent- the principles into the analysis.

The aim of this Diploma Thesis, as explained in the Introduction, was to produce a Static Design Model for the Aerial Manipulator by defining the optimal locations and directions of a number of rotors on the body structure, with respect to design limitations. These locations and directions were produced from a general optimization problem which has all the necessary design specifications and factors. This optimization problem is characterized from its dis-joint feasible regions, non-linear and non-smooth constraints which increase the complexity of solving process. Nevertheless, after a research in approaching this type of problem was made, a methodology was stated that produces sufficiently

accurate results in possible future simulations. In addition to that, with the goal of introducing possible aerodynamic interaction between the rotors appropriate experimental results were considered.

The next phase of this thesis was to search and collect specific flying components such as propellers and motors that will match the requirements of the Aerial Manipulator. Here was made a combination between the manufacturer performance data and the model used to approximate the flight principles. Two were the reasons of this combination: a gap observed in some critical information the manufacturer offers to the consumer when calculating the thrust force and torque, and the necessity to provide a model that approximates the distribution of thrust and torque as the control of the Aerial Manipulator will be achieved by tuning each rotor's rpm.

Finally, the overall Static Design Model accompanied with the mathematical equations, the constraints, the flight principles, the aerodynamic effects and the performance data of each component (motors/propellers) where programmed in Matlab environment. Thus, by solving several optimizations problems each time was found the maximum actuating force and torque the Aerial Manipulator can apply via the end-effector when using specific motors and propellers.

7.2   Suggestions for future work

Additions to the model concerning the flight principles

It is commonly known that every using model has its own limitations and simplifications, which happened also in this analysis. Several effects have been omitted such as:

- Reverse flow region

- Blade tip losses

- Feathering and Lead-Lag

- Aerodynamic forces and moments into the airframe

In order to include these in the future, special research should be made. As said in previous Chapter these effects can not simply introduced into the model because they might require lot of experimental data and rich theories. In addition to that, since the main goal of this thesis was to produce a Static Design model the studied flight state was Hovering. So, a work could be done in incorporating the other states of Axial Flight

and Forward Flight into the analysis. These might need a more materialized form of the Aerial Manipulator so as to gather experimental flight data such as speed ratios, volts and rpm.

Possible approximations into the optimization problem

It was mentioned that one difficulty of the optimization problem lies in the existence of non-smooth constraints. More specifically the two non-smooth constraints are:

- the inside "smaller" optimization problem that defines the distance, concerning the aerodynamic effects, between the thrusters

- the condition number

A possible future work could be made in transforming, with various reasons and procedures, these non-smooth constraints into smooth approximations.

So, as referred, there are ways to transform the "smaller" optimization from non-smooth to a approximately smooth one using theories like the *Perturbation Approach to Sensitivity Analysis* (see [23]) and the second constraint can be transformed using the approximation of the Jacobian matrix with the Singular Value Decomposition (SVD) [24]. These transformations would be really interesting points in terms of comparing possible variations to the values of the objective function or to the values of the design variables. Moreover, by transforming the non-smooth constraints to smooth might result also in reducing the computational costs, since some deterministic algorithms would be also applicable to the Design optimization problem. Emphasis is given to both the computational costs and to the selected algorithm because, as it was seen, are crucial factors in determining accurate results to the problem.

Aerodynamic effects

It is clear that the aerodynamic effects of each thruster have special role to the analysis and to the results produced from the optimization problems. Apart from this though, it is also perceptible the lack of experimental data concerning the accuracy of the aerodynamic interaction between thrusters.

Notice that the effects chosen in this thesis were the only available from experiments which were made at *Asctec Firefly*, a hexarotor with 8×4.5 inches size of propellers and *Hacker* motor. As stated in here were used also 8 inches propellers from the APC company but this do not mean that the aerodynamic effects will be the same. That happens because the shape, the angle of attack, the thickness and the distribution of the produced lift force from the root to the tip of each blade differs. That hopefully

became clear from the previous Chapters and someone can conceive it by comparing the produced thrust force in specific rpm between the propellers. In fact, the produced thrust force, of the *Asctec Firefly* propellers, in specific rpm is considerably less than the force produced from the APC propellers with the same length size. Therefore, an aspect for further approach would be to evaluate correctly the aerodynamic effects and defining more accurately the air flow throughout the propeller. That could possibly happen by using wind-tunnel measurements in the specific combinations of propellers and motors used in here. Afterwards, running again the optimization problems so as to produce better results would be necessary.

An other point that requires focus lies in the fact of considering the box that contains the electronics into the optimization problem. In that way the optimal locations and directions the box must have will be found, so as to avoid the aerodynamic interaction with the rotors.

Areas for further research

The whole analysis in this Diploma thesis is based on the Static Design Model of the Aerial Manipulator. As it is an unmanned vehicle analysis concerning the Dynamic and Kinematic behaviour would be very interesting areas for further research, since only then could be determined the movement of the Aerial Manipulator in space. Of course, an other crucial part for future work would be the Control Analysis that will enable the correct handling of the vehicle.

Appendix A

Appendix Matlab Files

The main files required by the Simulation are included in this Appendix.

volume.m

```
function f = volume(x)
% volume
F=[x(22:42)];
R=[x(1:21)];
f=norm(R);
end
```

cyl_d.m

```
function [c, ceq ]= cyl_d(x)
%where x is the both r and f
R=[x(1:3);x(4:6);x(7:9);x(10:12);x(13:15);x(16:18);x(19:21)]; %contains all r (the last one is the assistive)
F=[x(22:24);x(25:27);x(28:30);x(31:33);x(34:36);x(37:39);x(40:42)]; %contains all f (the last one is the assistive)
%%%%
%producing the unit vectors
%%%%
f1=(x(22:24));
f1=f1/norm(f1);
f2=(x(25:27));
```

f2=f2/norm(f2);

f3=(x(28:30));

f3=f3/norm(f3);

f4=(x(31:33));

f4=f4/norm(f4);

f5=(x(34:36));

f5=f5/norm(f5);

f6=(x(37:39));

f6=f6/norm(f6);

f7=(x(40:42));

f7=f7/norm(f7); %the assistive one

%%%%

%producing the SF matrices

%%%%

SF1=[0 -f1(1,3) f1(1,2);f1(1,3) 0 -f1(1,1);-f1(1,2) f1(1,1) 0];

SF2=[0 -f2(1,3) f2(1,2);f2(1,3) 0 -f2(1,1);-f2(1,2) f2(1,1) 0];

SF3=[0 -f3(1,3) f3(1,2);f3(1,3) 0 -f3(1,1);-f3(1,2) f3(1,1) 0];

SF4=[0 -f4(1,3) f4(1,2);f4(1,3) 0 -f4(1,1);-f4(1,2) f4(1,1) 0];

SF5=[0 -f5(1,3) f5(1,2);f5(1,3) 0 -f5(1,1);-f5(1,2) f5(1,1) 0];

SF6=[0 -f6(1,3) f6(1,2);f6(1,3) 0 -f6(1,1);-f6(1,2) f6(1,1) 0];

SFa=[0 -f7(1,3) f7(1,2);f7(1,3) 0 -f7(1,1);-f7(1,2) f7(1,1) 0]; %the assistive one

%%%%

%producing the SF,r product

%%%%

SF1r1=SF1*R(1,1:3)';

SF2r2=SF2*R(2,1:3)';

SF3r3=SF3*R(3,1:3)';

SF4r4=SF4*R(4,1:3)';

SF5r5=SF5*R(5,1:3)';

SF6r6=SF6*R(6,1:3)';

%%%%

%producing the E matrix

%%%%

E_negative=[SF1r1 SF2r2 SF3r3 SF4r4 SF5r5 SF6r6];

F1=[f1' f2' f3' f4' f5' f6'];

%%%%

%producing the D matrix

%%%%

D=[F1;E_negative];

```
%%%%
%introducing the singular values
%%%%
s=svd(D);
%calculating all distances
for i=1:6
for j=i+1:7
ri= [R(i,1:3)]';
rj= [R(j,1:3)]';
fi= [F(i,1:3)]';
fj= [F(j,1:3)]';


[x, fval]=domi2(ri,rj,fi,fj);


dij(i,j)=[fval];
end
end
% Nonlinear inequality constraints c(x)≤ 0
% dij ≥ ε where ε is user defined
c=[-dij(1,2)+0.001;
-dij(1,3)+0.001;
-dij(1,4)+0.001;
-dij(1,5)+0.001;
-dij(1,6)+0.001;
-dij(1,7)+0.001;
-dij(2,3)+0.001;
-dij(2,4)+0.001;
-dij(2,5)+0.001;
-dij(2,6)+0.001;
-dij(2,7)+0.001;
-dij(3,4)+0.001;
-dij(3,5)+0.001;
-dij(3,6)+0.001;
-dij(3,7)+0.001;
-dij(4,5)+0.001;
-dij(4,6)+0.001;
-dij(4,7)+0.001;
-dij(5,6)+0.001;
```

-dij(5,7)+0.001;

-dij(6,7)+0.001;

%the condition number constraint

% k(D) $\leq K$ where K is user defined

cond(D)-10;

%introducing the singular values $s \geq e$, where e is user defined

-s+0.01]

% Nonlinear equality constraints ceq(x)=0

ceq=[f1'+f2'+f3'+f4'+f5'+f6'+f7'

SF1*R(1,1:3)'+SF2*R(2,1:3)'+SF3*R(3,1:3)'+SF4*R(4,1:3)'+SF5*R(5,1:3)'

+SF6*R(6,1:3)'+SFa*R(7,1:3)'];

end

domi2.m

function[x, fval]=domi2 (ri,rj,fi,fj)

% define the translation and orientation vectors for both cylinder 1 and 2

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%Find tranformation matrices

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

opt.Display='off';

angles=fsolve(@(x)ftoeuler(x,fi),[0 0 0]',opt); phi=angles(1);

theta=angles(2);

psi=angles(3);

Rz=[cos(psi) -sin(psi) 0 0;sin(psi) cos(psi) 0 0; 0 0 1 0;0 0 0 1];

Ry=[cos(theta) 0 sin(theta) 0; 0 1 0 0; -sin(theta) 0 cos(theta) 0; 0 0 0 1];

Rx=[1 0 0 0; 0 cos(phi) -sin(phi) 0; 0 sin(phi) cos(phi) 0; 0 0 0 1];

RR1=Rz*Ry*Rx;

RR1=[RR1(1,1:3);

RR1(2,1:3);

RR1(3,1:3)];

Trans1=[inv(RR1) -ri;

0 0 0 1];

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

angles=fsolve(@(x)ftoeuler(x,fj),[0 0 0]',opt);

phi=angles(1);

```
theta=angles(2);
psi=angles(3);
Rz=[cos(psi) -sin(psi) 0 0;sin(psi) cos(psi) 0 0; 0 0 1 0;0 0 0 1];
Ry=[cos(theta) 0 sin(theta) 0; 0 1 0 0; -sin(theta) 0 cos(theta) 0; 0 0 0 1];
Rx=[1 0 0 0; 0 cos(phi) -sin(phi) 0; 0 sin(phi) cos(phi) 0; 0 0 0 1];
RR2=Rz*Ry*Rx;
RR2=[RR2(1,1:3);
RR2(2,1:3);
RR2(3,1:3)];
Trans2=[inv(RR2) -rj;
0 0 0 1];
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%follows the solving of optimization problem
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%


[x, fval] =solver(RR1,RR2,ri,rj);


end
```

ftoeuler.m

```
function y=ftoeuler(x,f)
%contains the necessary part of the Euler Rotation Matrix
%unit directional vector
f=f/norm(f);
phi=x(1);
theta=x(2);
psi=x(3);
%rotation matrix
Rz=[cos(psi) -sin(psi) 0 0;sin(psi) cos(psi) 0 0; 0 0 1 0;0 0 0 1];
Ry=[cos(theta) 0 sin(theta) 0; 0 1 0 0; -sin(theta) 0 cos(theta) 0; 0 0 0 1];
Rx=[1 0 0 0; 0 cos(phi) -sin(phi) 0; 0 sin(phi) cos(phi) 0; 0 0 0 1];
R=Rz*Ry*Rx;
ff= R*[1 0 0 1]';
ff=[ff(1) ff(2) ff(3)]';
y=f-ff;
```

solver.m

This is the *solver.m* for cylindrical aerodynamic effect shape of the "test-case"

```
function [x, fval] =solver(RR1,RR2,ri,rj)
options = optimset('Display','off','Algorithm','active-set','MaxFunEvals',30000000,'MaxIter',1000);


[x, fval] = fmincon(@rosenbrock,[0 0 0 0 0 0],[],[],[],[],[],[], @(x) unitdisk(x,RR1,RR2,ri,rj),
options);


end
function f = rosenbrock(x)
%distance between two points in the world coordinate
f = sqrt((x(1) − x(4))^2 + (x(2) − x(5))^2 + (x(3) − x(6))^2);
end


function [c, ceq] = unitdisk (x,RR1,RR2,ri,rj)


% constrains concerning the first and the second cylinder
% transform x from world frame to Cyc1
xyz1=inv(RR1)*([x(1:3)]'-ri); %xyz
% transform x from world frame to Cyc2
xyz2=inv(RR2)*([x(4:6)]'-rj); %xyz
c=[ -xyz1(1)-1;
xyz1(1)-1;
xyz1(2)^2+xyz1(3)^2-1;
-xyz2(1)-1;
xyz2(1)-1;
xyz2(2)^2+xyz2(3)^2-1];
ceq =[] ;
end
```

This is the *solver.m* for experimental aerodynamic effect shape

```
function [x, fval] =solver(RR1,RR2,ri,rj)
options = optimset('Display','off','Algorithm','active-set','MaxFunEvals',30000000,'MaxIter',1000);


[x, fval] = fmincon(@rosenbrock,[0 0 0 0 0 0],[],[],[],[],[],[], @(x) unitdisk(x,RR1,RR2,ri,rj),
options);


end
function f = rosenbrock(x)
%distance between two points in the world coordinate
f = sqrt((x(1) − x(4))^2 + (x(2) − x(5))^2 + (x(3) − x(6))^2);
end


function [c, ceq] = unitdisk (x,RR1,RR2,ri,rj)


% constrains concerning the first and the second cylinder
% transform x from world frame to Cyc1
xyz1=inv(RR1)*([x(1:3)]'-ri); %xyz
% transform x from world frame to Cyc2
xyz2=inv(RR2)*([x(4:6)]'-rj); %xyz
c=[ -xyz1(1)-1;
xyz1(1)-1;
xyz1(2)^2+xyz1(3)^2-(0.0086 ∗ xyz1(1)^2 − 0.003 ∗ xyz1(1) + 0.122)^2;
-xyz2(1)-1;
xyz2(1)-1;
xyz2(2)^2+xyz2(3)^2-(0.0086 ∗ xyz2(1)^2 − 0.003 ∗ xyz2(1) + 0.122)^2];
ceq =[] ;
end
```

plotstructure.m

```
function plotstructure(R,F,nc)
k=1;
pcolor=['r'; 'g' ;'b' ;'y' ;'k' ;'c' ;'m'];
close all
```

```
hold off
opt.Display='off';
for i=1:nc


ri=R(k:k+2)';
fi=F(k:k+2)';
fi=fi/norm(fi);


angles=fsolve(@(x)ftoeuler(x,fi),[0 0 0]',opt);


phi=angles(1);
theta=angles(2);
psi=angles(3);


Rz=[cos(psi) -sin(psi) 0 0;sin(psi) cos(psi) 0 0; 0 0 1 0;0 0 0 1];
Ry=[cos(theta) 0 sin(theta) 0; 0 1 0 0; -sin(theta) 0 cos(theta) 0; 0 0 0 1];
Rx=[1 0 0 0; 0 cos(phi) -sin(phi) 0; 0 sin(phi) cos(phi) 0; 0 0 0 1];
RR=Rz*Ry*Rx;
%inverse transformation
xs=(RR)*([-1 0 0 1]')+[ri;1];
xs=[xs(1);xs(2);xs(3)];
xf=(RR)*([1 0 0 1]')+[ri;1];
xf=[xf(1);xf(2);xf(3)];
if (norm(xf)¿norm(xs))
Cylinder(xs,xf,1,20,pcolor(i,:),0,0);
else
Cylinder(xf,xs,1,20,pcolor(i,:),0,0);
end
hold on
plot3(ri(1),ri(2),ri(3),[pcolor(i,:) '.'],'MarkerSize',30);
%this is the vector showing the directon of the cylinder
quiver3(ri(1), ri(2), ri(3), fi(1)/norm(fi),fi(2)/norm(fi),fi(3)/norm(fi),pcolor(i,:),'LineWidth',2);
plot3(xs(1),xs(2),xs(3),[pcolor(i,:) '.'],'MarkerSize',30);
plot3(xf(1),xf(2),xf(3),[pcolor(i,:) '.'],'MarkerSize',30);
k=k+3;
end
grid on
```

axis equal

xlabel('x');

ylabel('y');

zlabel('z');

Cylinder.m

function [Cylinder EndPlate1 EndPlate2] = Cylinder(X1,X2,r,n,cyl_color,closed,lines)

% Calculating the length of the cylinder

length_cyl=norm(X2-X1);

% Creating a circle in the YZ plane

t=linspace(0,2*pi,n)';

x2=r*cos(t);

x3=r*sin(t);

% Creating the points in the X-Direction

x1=[0 length_cyl];

% Creating (Extruding) the cylinder points in the X-Directions

xx1=repmat(x1,length(x2),1);

xx2=repmat(x2,1,2);

xx3=repmat(x3,1,2);

% Drawing two filled cirlces to close the cylinder

if closed==1

hold on

EndPlate1=fill3(xx1(:,1),xx2(:,1),xx3(:,1),'r');

EndPlate2=fill3(xx1(:,2),xx2(:,2),xx3(:,2),'r');

end

% Plotting the cylinder along the X-Direction with required length starting

% from Origin

```
Cylinder=mesh(xx1,xx2,xx3);
% Defining Unit vector along the X-direction
unit_Vx=[1 0 0];


% Calulating the angle between the x direction and the required direction
% of cylinder through dot product
angle_X1X2=acos( dot( unit_Vx,(X2-X1) )/( norm(unit_Vx)*norm(X2-X1)) )*180/pi;


% Finding the axis of rotation (single rotation) to rotate the cylinder in
% X-direction to the required arbitrary direction through cross product
axis_rot=cross([1 0 0],(X2-X1) );


% Rotating the plotted cylinder and the end plate circles to the required
% angles
if angle_X1X2 =0 % Rotation is not needed if required direction is along X
rotate(Cylinder,axis_rot,angle_X1X2,[0 0 0])
if closed==1
rotate(EndPlate1,axis_rot,angle_X1X2,[0 0 0])
rotate(EndPlate2,axis_rot,angle_X1X2,[0 0 0])
end
end


% Till now cylinder has only been aligned with the required direction, but
% position starts from the origin. so it will now be shifted to the right
% position
if closed==1
set(EndPlate1,'XData',get(EndPlate1,'XData')+X1(1))
set(EndPlate1,'YData',get(EndPlate1,'YData')+X1(2))
set(EndPlate1,'ZData',get(EndPlate1,'ZData')+X1(3))


set(EndPlate2,'XData',get(EndPlate2,'XData')+X1(1))
set(EndPlate2,'YData',get(EndPlate2,'YData')+X1(2))
set(EndPlate2,'ZData',get(EndPlate2,'ZData')+X1(3))
end
set(Cylinder,'XData',get(Cylinder,'XData')+X1(1))
set(Cylinder,'YData',get(Cylinder,'YData')+X1(2))
```

set(Cylinder,'ZData',get(Cylinder,'ZData')+X1(3))

% Setting the color to the cylinder and the end plates
set(Cylinder,'FaceColor',cyl_color)
if closed==1
set([EndPlate1 EndPlate2],'FaceColor',cyl_color)
else
EndPlate1=[];
EndPlate2=[];
end

% If lines are not needed making it disapear
if lines==0
set(Cylinder,'EdgeAlpha',0)
end

aerodynamic_effect_position_orientation.m

function [Cylinder EndPlate1 EndPlate2] =
aerodynamic_effect_position_orientation(X1,X2,r,n,cyl_color,closed,lines)

%aerodynamic effect equations
u=linspace(0,0.49,40);
v=linspace(0,2*pi,40)';
$a = 0.008571 * u.^2 - 0.003 * u + 0.122$;
a=a(1,1:end);
z=sin(v)*a;
y=cos(v)*a;
x=u;

xx1=repmat(x,length(y),1);

% Defining Unit vector along the X-direction
unit_Vx=[1 0 0];

% Calulating the angle between the x direction and the required direction

angle_X1X2=acos(dot(unit_Vx,(X2-X1))/(norm(unit_Vx)*norm(X2-X1)))*180/pi;

% Finding the axis of rotation (single rotation) to rotate in

% X-direction to the required arbitrary direction through cross product

axis_rot=cross([1 0 0],(X2-X1));

% Plotting along the X-Direction with required length starting

% from Origin

hSurface =surf(xx1,y,z);

if angle_X1X2 =0 % Rotation is not needed if required direction is along X

rotate(hSurface,axis_rot,angle_X1X2,[0 0 0])

end

set(hSurface,'XData',get(hSurface,'XData')+X1(1))

set(hSurface,'YData',get(hSurface,'YData')+X1(2))

set(hSurface,'ZData',get(hSurface,'ZData')+X1(3))

% Setting the color

set(hSurface,'FaceColor',cyl_color)

if lines==0

set(hSurface,'EdgeAlpha',0)

end

Appendix B

Appendix Motor Performance Data

This Appendix includes data of the motor used in the Simulation.

Scorpion SII-2212-1070 Motor Propeller Data

Motor Wind	Motor Kv	No-Load Current	Motor Resistance	I Max	P Max (3S)
18-Turn Delta	1070 RPM/Volt	Io = 0.59 Amps @ 10v	Rm = 0.091 Ohms	15 Amps	160 W

Outside Diameter	Body Length	Total Shaft Length	Shaft Diameter	Motor Weight
27.9 mm, 1.098in.	23.0 mm, 1.181 in.	49.0 mm, 1.929 in.	2.98 mm, 0.117 in.	58.0 gm, 2.05 oz

Prop Manf.	Prop Size	Input Voltage	Motor Amps	Watts Input	Prop RPM	Pitch Speed	Thrust Grams	Thrust Ounces	Thrust Eff. Grams/W
APC	8x8-E	7.4	8.67	64.2	5,979	45.3	306.5	10.81	4.78
APC	9x4.7-SF	7.4	7.20	53.3	6,345	28.2	438	15.45	8.22
APC	9x6-E	7.4	7.41	54.8	6,289	35.7	404.3	14.26	7.37
APC	9x6-SF	7.4	10.37	76.7	5,528	31.4	483.5	17.05	6.30
APC	9x7.5-E	7.4	9.93	73.4	5,653	40.1	393.6	13.88	5.36
APC	9x7.5-SF	7.4	11.75	87.0	5,197	36.9	450.8	15.90	5.18
APC	9x9-E	7.4	11.60	85.8	5,210	44.4	356.3	12.57	4.15
APC	10x3.8-SF	7.4	9.59	71.0	5,730	20.6	538.8	19.01	7.59
APC	10x4.7-SF	7.4	10.40	77.0	5,522	24.6	563.9	19.89	7.33
APC	10x5-E	7.4	8.33	61.6	6,050	28.6	493.3	17.40	8.00
APC	10x6-E	7.4	9.24	68.4	5,814	33.0	511.4	18.04	7.48
APC	10x7-E	7.4	10.41	77.1	5,538	36.7	503.9	17.77	6.54
APC	10x7-SF	7.4	12.89	95.4	4,911	32.6	565.3	19.94	5.93
APC	10x10-E	7.4	13.64	100.9	4,726	44.8	381.5	13.46	3.78
APC	11x3.8-SF	7.4	10.51	77.8	5,508	19.8	593.9	20.95	7.64
APC	11x4.7-SF	7.4	12.16	90.0	5,070	22.6	638.6	22.53	7.10
APC	11x7-E	7.4	12.03	89.0	5,119	33.9	608.2	21.45	6.83
APC	11x7-SF	7.4	14.69	108.7	4,401	29.2	645.8	22.78	5.94
APC	11x8-E	7.4	12.83	94.9	4,910	37.2	521.6	18.40	5.49
APC	11x8.5-E	7.4	13.22	97.8	4,825	38.8	577.8	20.38	5.91
APC	11x10-E	7.4	14.99	110.9	4,346	41.2	432.6	15.26	3.90
APC	12x3.8-SF	7.4	13.35	98.8	4,763	17.1	683.6	24.11	6.92
APC	12x6-E	7.4	12.36	91.5	5,015	28.5	669.6	23.62	7.32
APC	12x6-SF	7.4	15.81	117.0	4,086	23.2	711	25.08	6.08
APC	12x8-E	7.4	14.37	106.3	4,503	34.1	570.2	20.11	5.36
APC	12x10-E	7.4	15.92	117.8	4,084	38.7	493.3	17.40	4.19
APC	13x4-E	7.4	11.56	85.5	5,217	19.8	680.9	24.02	7.96
GEM	8x4.5	7.4	6.72	49.8	6,476	27.6	383.3	13.52	7.70
GEM	8x4.5-C	7.4	6.53	48.4	6,374	27.2	379.7	13.39	7.85
GEM	9x4.7	7.4	7.71	57.0	6,208	27.6	451.7	15.93	7.92
GEM	9x4.7-C	7.4	7.56	56.0	6,131	27.3	453	15.98	8.10
GEM	10x4.5	7.4	10.17	75.2	5,582	23.8	545.6	19.25	7.25
GEM	11x4.7-C	7.4	12.19	90.2	5,097	22.7	644.6	22.74	7.15
GEM	12x4.5-C	7.4	13.63	100.9	4,708	20.1	637.2	22.48	6.32
GWS	8x4x3-DD	7.4	4.55	33.6	7,029	26.6	298.6	10.53	8.88
GWS	8x4.3-SF	7.4	5.12	37.9	6,873	28.0	323.6	11.41	8.54
GWS	8x6-SF	7.4	6.72	49.8	6,474	36.8	367.4	12.96	7.38
GWS	9x4.7-SF	7.4	7.61	56.3	6,233	27.7	453.9	16.01	8.06
GWS	9x5-DD	7.4	6.34	46.9	6,575	31.1	413.4	14.58	8.81
GWS	9x5x3-DD	7.4	7.70	57.0	6,219	29.4	449.9	15.87	7.89
GWS	9x7-SF	7.4	10.33	76.4	5,551	36.8	451.4	15.92	5.91
GWS	9x7.5-HD	7.4	9.41	69.6	5,803	41.2	401.6	14.17	5.77
GWS	10x4.7-SF	7.4	10.63	78.7	5,466	24.3	579.2	20.43	7.36
GWS	10x6-DD	7.4	8.22	60.8	6,078	34.5	492.7	17.38	8.10
GWS	10x6x3-DD	7.4	10.28	76.1	5,587	31.7	564.2	19.90	7.42
GWS	10x8-HD	7.4	11.81	87.4	5,210	39.5	479.6	16.92	5.49
GWS	10x8-SF	7.4	13.07	96.7	4,862	36.8	539.2	19.02	5.57
GWS	11x4.7-SF	7.4	12.25	90.7	5,087	22.6	648.1	22.86	7.15
GWS	11x7-DD	7.4	11.23	83.1	5,346	35.4	624.6	22.03	7.52
GWS	12x8-DD	7.4	14.10	104.3	4,608	34.9	675.9	23.84	6.48
MAS	8x6x3	7.4	7.17	53.0	6,351	36.1	312.1	11.01	5.88
MAS	9x7x3	7.4	9.95	73.6	5,636	37.4	419.3	14.79	5.69
MAS	10x5x3	7.4	9.01	66.6	5,883	27.9	501.2	17.68	7.52
MAS	10x7x3	7.4	11.77	87.1	5,193	34.4	530.6	18.72	6.09
MAS	11x7x3	7.4	12.98	96.0	4,840	32.1	612.7	21.61	6.38
MAS	11x8x3	7.4	13.65	101.0	4,651	35.2	610.7	21.54	6.04
MAS	12x6x3	7.4	13.88	102.7	4,596	26.1	658.2	23.22	6.41
MAS	12x8x3	7.4	16.21	120.0	3,940	29.8	674.2	23.78	5.62
Prop Manf.	**Prop Size**	**Input Voltage**	**Motor Amps**	**Watts Input**	**Prop RPM**	**Pitch Speed**	**Thrust Grams**	**Thrust Ounces**	**Thrust Eff. Grams/W**
APC	7x4-E	11.1	6.28	69.8	10,535	39.9	442.3	15.60	6.34
APC	7x4-SF	11.1	6.34	70.4	10,520	39.8	423	14.92	6.01
APC	7x5-E	11.1	7.92	88.0	10,088	47.8	454	16.01	5.16
APC	7x5-SF	11.1	7.74	85.9	10,138	48.0	462.2	16.30	5.38
APC	7x6-E	11.1	8.33	92.4	9,969	56.6	489.7	17.27	5.30
APC	7x6-SF	11.1	9.33	103.5	9,673	55.0	456.6	16.11	4.41
APC	8x3.8-SF	11.1	10.17	112.9	9,443	34.0	658.2	23.22	5.83
APC	8x4-E	11.1	9.03	100.2	9,753	36.9	600.7	21.19	5.99
APC	8x6-E	11.1	12.11	134.4	8,904	50.6	661.5	23.33	4.92
APC	8x6-SF	11.1	14.97	166.1	8,024	45.6	674.5	23.79	4.06
APC	8x8-E	11.1	15.11	167.8	7,974	60.4	555.5	19.59	3.31
APC	9x3.8-SF	11.1	12.96	143.9	8,619	31.0	798.5	28.17	5.55
APC	9x4.5-E	11.1	11.97	132.8	8,933	38.1	793.2	27.98	5.97
APC	9x4.7-SF	11.1	12.84	142.5	8,675	38.6	817.8	28.85	5.74
APC	9x6-E	11.1	13.66	151.6	8,440	48.0	759.3	26.78	5.01
APC	9x7.5-E	11.1	17.05	189.2	7,344	52.2	709.1	25.01	3.75
GEM	8x4.5	11.1	12.35	137.1	8,816	37.6	730.6	25.77	5.33
GEM	8x4.5-C	11.1	11.64	129.2	8,641	36.8	708.3	24.98	5.48
GEM	9x4.7-C	11.1	13.02	144.5	8,252	36.7	809.8	28.56	5.61
GEM	10x4.5	11.1	16.67	185.0	7,128	30.4	944.8	33.33	5.11
GWS	8x4-DD	11.1	7.23	80.2	10,274	38.9	552.1	19.47	6.88
GWS	8x4x3-DD	11.1	8.73	96.9	8,611	32.6	607.8	21.44	6.28
GWS	9x5-DD	11.1	11.70	129.9	9,006	42.6	794.5	28.02	6.12
GWS	9x5x3-DD	11.1	14.04	155.8	7,303	34.6	838.9	29.59	5.38
GWS	9x7.5-DD	11.1	16.38	181.8	7,520	53.4	703.8	24.83	3.87
GWS	10x6x3-DD	11.1	17.23	191.2	6,394	36.3	981.8	34.63	5.13
MAS	7x4x3	11.1	7.38	81.9	10,217	38.7	423	14.92	5.17
MAS	8x6x3	11.1	12.80	142.1	8,706	49.5	641.8	22.64	4.52
MAS	9x7x3	11.1	16.78	186.2	7,412	49.1	809.1	28.54	4.34

Propeller Chart Color Code Explanation

The prop is to small to get good performance from the motor. (Less than 50% power)

The prop is sized right to get good power from the motor. (50 to 80% power)

The prop can be used, but full throttle should be kept to short bursts. (80 to 100% power)

The prop is too big for the motor and should not be used. (Over 100% power)

PLEASE NOTE:
The data contained in this prop chart is based on actual measurements taken in a controlled test environment. The test voltages used are based on a properly sized Li-Po battery for the current draw of the motor being tested. If you are using a larger than normal capacity battery, or a very high C-Rated battery, your actual voltages will be higher than those shown in this chart, and this will result in higer current draw for each prop used. You should always test your power system with a watt meter whenever a prop is used to ensure that you are not exceeding the recommended rating of the motor being used. The prop recommendations in this chart are based on the motor receiving adequate cooling throughout its operation. If your motor is being used inside a cowl, you must provide adequate cooling to the motor and make sure that the motor is not getting too hot during operation.

Appendix C

Appendix Propeller Data

This Appendix includes data of the propeller used in the Simulation.

8x38SF.dat

06/08/13

====== PERFORMANCE DATA APC (versus advance ratio and MPH) ======

DEFINITIONS:

J=V/nD (advance ratio)

Ct=T/(rho * n**2 * D**4) (thrust coef.)

Cp=P/(rho * n**3 * D**5) (power coef.)

Pe=Ct*J/Cp (efficiency)

V (model speed in MPH)

PROP RPM = 4000

V (mph)	J (Adv Ratio)	Pe (Hp)	Ct (In-Lbf)	Cp (Lbf)	PWR	Torque	Thrust
0.0	0.00	0.0000	0.1200	0.0779	0.013	0.207	0.250
0.8	0.03	0.0407	0.1105	0.0729	0.012	0.194	0.231
1.6	0.05	0.0802	0.1011	0.0677	0.011	0.180	0.211
2.4	0.08	0.1184	0.0920	0.0626	0.011	0.166	0.192
3.3	0.11	0.1553	0.0832	0.0576	0.010	0.153	0.174
4.1	0.13	0.1909	0.0746	0.0524	0.009	0.139	0.156
4.9	0.16	0.2252	0.0709	0.0507	0.009	0.135	0.148
5.7	0.19	0.2582	0.0686	0.0500	0.008	0.133	0.143
6.5	0.21	0.2899	0.0663	0.0491	0.008	0.131	0.138
7.3	0.24	0.3203	0.0639	0.0482	0.008	0.128	0.133
8.1	0.27	0.3495	0.0615	0.0472	0.008	0.126	0.128
9.0	0.30	0.3773	0.0590	0.0462	0.008	0.123	0.123
9.8	0.32	0.4038	0.0563	0.0450	0.008	0.119	0.118
10.6	0.35	0.4291	0.0536	0.0436	0.007	0.116	0.112
11.4	0.38	0.4531	0.0508	0.0421	0.007	0.112	0.106
12.2	0.40	0.4756	0.0479	0.0406	0.007	0.108	0.100
13.0	0.43	0.4965	0.0449	0.0389	0.007	0.103	0.094
13.8	0.46	0.5158	0.0419	0.0371	0.006	0.099	0.087
14.6	0.48	0.5331	0.0387	0.0351	0.006	0.093	0.081
15.5	0.51	0.5481	0.0356	0.0331	0.006	0.088	0.074
16.3	0.54	0.5599	0.0323	0.0310	0.005	0.082	0.067
17.1	0.56	0.5674	0.0290	0.0288	0.005	0.077	0.060
17.9	0.59	0.5693	0.0256	0.0266	0.004	0.071	0.053
18.7	0.62	0.5637	0.0221	0.0242	0.004	0.064	0.046
19.5	0.64	0.5480	0.0186	0.0219	0.004	0.058	0.039
20.3	0.67	0.5183	0.0150	0.0195	0.003	0.052	0.031
21.2	0.70	0.4673	0.0114	0.0170	0.003	0.045	0.024
22.0	0.73	0.3820	0.0076	0.0145	0.002	0.038	0.016
22.8	0.75	0.2397	0.0038	0.0120	0.002	0.032	0.008
23.6	0.78	-0.0029	0.0000	0.0095	0.002	0.025	0.000

PROP RPM = 5000

V (mph)	J (Adv Ratio)	Pe	Ct	Cp	PWR (Hp)	Torque (In-Lbf)	Thrust (Lbf)
0.0	0.00	0.0000	0.1184	0.0766	0.025	0.318	0.386
1.0	0.03	0.0398	0.1091	0.0716	0.024	0.298	0.356
2.0	0.05	0.0784	0.1000	0.0667	0.022	0.277	0.326
3.0	0.08	0.1158	0.0912	0.0617	0.020	0.256	0.297
4.0	0.10	0.1520	0.0826	0.0568	0.019	0.236	0.269
5.0	0.13	0.1870	0.0743	0.0519	0.017	0.216	0.242
5.9	0.16	0.2207	0.0697	0.0496	0.016	0.206	0.228
6.9	0.18	0.2532	0.0675	0.0488	0.016	0.203	0.220
7.9	0.21	0.2844	0.0652	0.0479	0.016	0.199	0.213
8.9	0.24	0.3145	0.0628	0.0470	0.015	0.195	0.205
9.9	0.26	0.3433	0.0604	0.0460	0.015	0.191	0.197
10.9	0.29	0.3708	0.0579	0.0449	0.015	0.187	0.189
11.9	0.31	0.3971	0.0554	0.0438	0.014	0.182	0.181
12.9	0.34	0.4220	0.0527	0.0424	0.014	0.176	0.172
13.9	0.37	0.4457	0.0499	0.0410	0.014	0.170	0.163
14.9	0.39	0.4681	0.0471	0.0394	0.013	0.164	0.154
15.8	0.42	0.4888	0.0442	0.0378	0.012	0.157	0.144
16.8	0.44	0.5080	0.0412	0.0360	0.012	0.150	0.134
17.8	0.47	0.5251	0.0381	0.0342	0.011	0.142	0.124
18.8	0.50	0.5400	0.0350	0.0322	0.011	0.134	0.114
19.8	0.52	0.5518	0.0318	0.0301	0.010	0.125	0.104
20.8	0.55	0.5593	0.0286	0.0280	0.009	0.116	0.093
21.8	0.58	0.5614	0.0252	0.0258	0.009	0.107	0.082
22.8	0.60	0.5555	0.0218	0.0236	0.008	0.098	0.071
23.8	0.63	0.5400	0.0183	0.0213	0.007	0.088	0.060
24.8	0.65	0.5099	0.0148	0.0189	0.006	0.079	0.048
25.7	0.68	0.4588	0.0111	0.0165	0.005	0.069	0.036
26.7	0.71	0.3739	0.0075	0.0141	0.005	0.059	0.024
27.7	0.73	0.2322	0.0037	0.0117	0.004	0.049	0.012
28.7	0.76	-0.0036	0.0000	0.0094	0.003	0.039	0.000

PROP RPM = 6000

V (mph)	J (Adv Ratio)	Pe	Ct	Cp	PWR (Hp)	Torque (In-Lbf)	Thrust (Lbf)
0.0	0.00	0.0000	0.1177	0.0751	0.043	0.449	0.553
1.2	0.03	0.0399	0.1085	0.0703	0.040	0.420	0.510
2.4	0.05	0.0787	0.0996	0.0654	0.037	0.391	0.468
3.5	0.08	0.1163	0.0908	0.0606	0.035	0.362	0.427
4.7	0.10	0.1527	0.0823	0.0558	0.032	0.334	0.387
5.9	0.13	0.1880	0.0741	0.0510	0.029	0.305	0.348
7.1	0.16	0.2220	0.0693	0.0484	0.028	0.290	0.325
8.2	0.18	0.2548	0.0670	0.0476	0.027	0.285	0.315
9.4	0.21	0.2864	0.0647	0.0467	0.027	0.280	0.304
10.6	0.23	0.3168	0.0623	0.0458	0.026	0.274	0.293
11.8	0.26	0.3460	0.0598	0.0447	0.025	0.268	0.281
12.9	0.28	0.3739	0.0573	0.0436	0.025	0.261	0.269
14.1	0.31	0.4006	0.0548	0.0424	0.024	0.254	0.257
15.3	0.34	0.4260	0.0521	0.0412	0.023	0.246	0.245
16.5	0.36	0.4501	0.0494	0.0398	0.023	0.238	0.232
17.6	0.39	0.4727	0.0466	0.0383	0.022	0.229	0.219
18.8	0.41	0.4938	0.0437	0.0366	0.021	0.219	0.205
20.0	0.44	0.5132	0.0407	0.0349	0.020	0.209	0.191
21.2	0.47	0.5307	0.0376	0.0330	0.019	0.197	0.177
22.3	0.49	0.5457	0.0345	0.0311	0.018	0.186	0.162
23.5	0.52	0.5572	0.0313	0.0291	0.017	0.174	0.147
24.7	0.54	0.5644	0.0280	0.0270	0.015	0.161	0.132
25.9	0.57	0.5660	0.0248	0.0249	0.014	0.149	0.116
27.0	0.60	0.5601	0.0214	0.0228	0.013	0.136	0.101
28.2	0.62	0.5440	0.0180	0.0205	0.012	0.123	0.085
29.4	0.65	0.5134	0.0145	0.0183	0.010	0.109	0.068
30.6	0.67	0.4613	0.0110	0.0160	0.009	0.096	0.052
31.8	0.70	0.3752	0.0074	0.0137	0.008	0.082	0.035
32.9	0.72	0.2333	0.0037	0.0115	0.007	0.068	0.017
34.1	0.75	-0.0012	0.0000	0.0093	0.005	0.056	0.000

PROP RPM = 7000

V (mph)	J (Adv Ratio)	Pe	Ct	Cp	PWR (Hp)	Torque (In-Lbf)	Thrust (Lbf)
0.0	0.00	0.0000	0.1169	0.0731	0.066	0.595	0.748
1.4	0.03	0.0412	0.1077	0.0683	0.062	0.556	0.689
2.8	0.05	0.0812	0.0987	0.0635	0.057	0.517	0.631
4.2	0.08	0.1200	0.0899	0.0587	0.053	0.478	0.575
5.5	0.10	0.1576	0.0814	0.0539	0.049	0.439	0.520
6.9	0.13	0.1940	0.0731	0.0492	0.044	0.401	0.467
8.3	0.16	0.2292	0.0685	0.0469	0.042	0.381	0.438
9.7	0.18	0.2632	0.0662	0.0460	0.042	0.374	0.423
11.1	0.21	0.2959	0.0639	0.0451	0.041	0.367	0.408
12.5	0.24	0.3274	0.0614	0.0441	0.040	0.359	0.393
13.9	0.26	0.3575	0.0589	0.0430	0.039	0.350	0.377
15.2	0.29	0.3864	0.0564	0.0419	0.038	0.341	0.360
16.6	0.31	0.4139	0.0537	0.0407	0.037	0.331	0.344
18.0	0.34	0.4401	0.0510	0.0394	0.036	0.321	0.326
19.4	0.37	0.4648	0.0483	0.0380	0.034	0.309	0.309
20.8	0.39	0.4878	0.0455	0.0365	0.033	0.297	0.291
22.2	0.42	0.5090	0.0426	0.0349	0.032	0.284	0.272
23.5	0.44	0.5286	0.0396	0.0333	0.030	0.271	0.253
24.9	0.47	0.5459	0.0366	0.0315	0.028	0.257	0.234
26.3	0.50	0.5602	0.0335	0.0297	0.027	0.242	0.214
27.7	0.52	0.5708	0.0304	0.0278	0.025	0.226	0.194
29.1	0.55	0.5768	0.0271	0.0258	0.023	0.210	0.174
30.5	0.57	0.5769	0.0240	0.0239	0.022	0.194	0.153
31.9	0.60	0.5693	0.0207	0.0219	0.020	0.178	0.133
33.2	0.63	0.5512	0.0175	0.0198	0.018	0.162	0.112
34.6	0.65	0.5182	0.0141	0.0178	0.016	0.145	0.090
36.0	0.68	0.4632	0.0107	0.0156	0.014	0.127	0.068
37.4	0.71	0.3741	0.0072	0.0135	0.012	0.110	0.046
38.8	0.73	0.2299	0.0036	0.0115	0.010	0.093	0.023
40.2	0.76	-0.0026	0.0000	0.0095	0.009	0.078	0.000

PROP RPM = 8000

V (mph)	J (Adv Ratio)	Pe	Ct	Cp	PWR (Hp)	Torque (In-Lbf)	Thrust (Lbf)
0.0	0.00	0.0000	0.1163	0.0715	0.097	0.760	0.972
1.6	0.03	0.0425	0.1070	0.0667	0.090	0.709	0.893
3.2	0.05	0.0837	0.0979	0.0619	0.084	0.658	0.817
4.8	0.08	0.1238	0.0890	0.0571	0.077	0.607	0.743
6.4	0.11	0.1626	0.0804	0.0524	0.071	0.557	0.672
8.0	0.13	0.2002	0.0721	0.0476	0.064	0.507	0.602
9.6	0.16	0.2366	0.0679	0.0456	0.062	0.485	0.567
11.2	0.19	0.2717	0.0656	0.0447	0.060	0.475	0.548
12.8	0.21	0.3055	0.0632	0.0438	0.059	0.465	0.527
14.4	0.24	0.3379	0.0607	0.0428	0.058	0.455	0.507
16.0	0.26	0.3690	0.0582	0.0417	0.056	0.443	0.486
17.6	0.29	0.3988	0.0556	0.0406	0.055	0.431	0.464
19.2	0.32	0.4271	0.0529	0.0393	0.053	0.418	0.442
20.9	0.34	0.4539	0.0502	0.0380	0.051	0.404	0.419
22.5	0.37	0.4790	0.0474	0.0367	0.050	0.390	0.396
24.1	0.40	0.5023	0.0447	0.0353	0.048	0.375	0.373
25.7	0.42	0.5239	0.0419	0.0338	0.046	0.360	0.350
27.3	0.45	0.5434	0.0390	0.0323	0.044	0.343	0.325
28.9	0.48	0.5606	0.0360	0.0306	0.041	0.325	0.301
30.5	0.50	0.5745	0.0330	0.0289	0.039	0.307	0.276
32.1	0.53	0.5845	0.0299	0.0271	0.037	0.288	0.250
33.7	0.56	0.5896	0.0268	0.0253	0.034	0.269	0.224
35.3	0.58	0.5882	0.0236	0.0234	0.032	0.248	0.197
36.9	0.61	0.5789	0.0204	0.0214	0.029	0.228	0.170
38.5	0.64	0.5594	0.0172	0.0195	0.026	0.207	0.143
40.1	0.66	0.5243	0.0139	0.0175	0.024	0.186	0.116
41.7	0.69	0.4684	0.0106	0.0156	0.021	0.166	0.089
43.3	0.71	0.3785	0.0072	0.0136	0.018	0.145	0.060
44.9	0.74	0.2267	0.0036	0.0116	0.016	0.124	0.030
46.5	0.77	-0.0008	0.0000	0.0099	0.013	0.105	0.000

PROP RPM = 12000

V	J	Pe	Ct	Cp	PWR	Torque	Thrust
(mph)	(Adv Ratio)				(Hp)	(In-Lbf)	(Lbf)
0.0	0.00	0.0000	0.1185	0.0705	0.321	1.686	2.226
2.5	0.03	0.0457	0.1085	0.0653	0.298	1.563	2.039
5.0	0.06	0.0902	0.0989	0.0603	0.275	1.443	1.859
7.5	0.08	0.1335	0.0896	0.0554	0.252	1.325	1.683
10.0	0.11	0.1754	0.0805	0.0505	0.230	1.208	1.513
12.5	0.14	0.2160	0.0719	0.0457	0.208	1.094	1.350
15.0	0.17	0.2553	0.0687	0.0444	0.202	1.063	1.292
17.5	0.19	0.2931	0.0664	0.0436	0.199	1.043	1.247
20.0	0.22	0.3295	0.0639	0.0427	0.194	1.021	1.201
22.5	0.25	0.3643	0.0614	0.0417	0.190	0.998	1.154
25.0	0.28	0.3978	0.0588	0.0407	0.185	0.973	1.105
27.5	0.30	0.4297	0.0561	0.0395	0.180	0.946	1.055
30.0	0.33	0.4601	0.0534	0.0383	0.175	0.917	1.004
32.5	0.36	0.4889	0.0507	0.0371	0.169	0.886	0.952
35.0	0.39	0.5160	0.0478	0.0357	0.162	0.853	0.898
37.5	0.41	0.5410	0.0450	0.0343	0.156	0.821	0.846
40.0	0.44	0.5640	0.0422	0.0330	0.150	0.788	0.794
42.5	0.47	0.5832	0.0395	0.0316	0.144	0.757	0.742
45.0	0.50	0.5996	0.0366	0.0302	0.138	0.723	0.688
47.5	0.52	0.6078	0.0338	0.0291	0.133	0.696	0.636
50.0	0.55	0.6121	0.0305	0.0274	0.125	0.655	0.572
52.5	0.58	0.6280	0.0273	0.0251	0.114	0.600	0.512
55.0	0.61	0.6069	0.0237	0.0236	0.108	0.566	0.446
57.5	0.63	0.6127	0.0207	0.0214	0.097	0.512	0.389
60.0	0.66	0.5871	0.0173	0.0194	0.089	0.465	0.325
62.5	0.69	0.5425	0.0139	0.0176	0.080	0.421	0.261
65.0	0.72	0.4736	0.0106	0.0161	0.073	0.384	0.200
67.5	0.74	0.3364	0.0066	0.0145	0.066	0.347	0.123
70.0	0.77	0.1842	0.0030	0.0126	0.058	0.303	0.057
72.5	0.80	-0.0005	0.0000	0.0108	0.049	0.259	0.000

8x47SF.dat

06/08/13

====== PERFORMANCE DATA (versus advance ratio and MPH) ======

DEFINITIONS:

J=V/nD (advance ratio)

Ct=T/(rho * n**2 * D**4) (thrust coef.)

Cp=P/(rho * n**3 * D**5) (power coef.)

Pe=Ct*J/Cp (efficiency)

V (model speed in MPH)

PROP RPM = 4000

V (mph)	J (Adv Ratio)	Pe	Ct	Cp	PWR (Hp)	Torque (In-Lbf)	Thrust (Lbf)
0.0	0.00	0.0000	0.1338	0.0897	0.015	0.238	0.279
0.9	0.03	0.0454	0.1259	0.0858	0.014	0.228	0.263
1.9	0.06	0.0894	0.1181	0.0818	0.014	0.217	0.247
2.8	0.09	0.1318	0.1104	0.0777	0.013	0.207	0.230
3.8	0.12	0.1727	0.1026	0.0735	0.012	0.195	0.214
4.7	0.15	0.2118	0.0950	0.0694	0.012	0.184	0.198
5.6	0.19	0.2496	0.0881	0.0656	0.011	0.174	0.184
6.6	0.22	0.2855	0.0856	0.0649	0.011	0.173	0.179
7.5	0.25	0.3201	0.0828	0.0640	0.011	0.170	0.173
8.4	0.28	0.3531	0.0798	0.0629	0.011	0.167	0.167
9.4	0.31	0.3845	0.0768	0.0618	0.010	0.164	0.160
10.3	0.34	0.4143	0.0737	0.0606	0.010	0.161	0.154
11.3	0.37	0.4425	0.0705	0.0592	0.010	0.157	0.147
12.2	0.40	0.4692	0.0672	0.0576	0.010	0.153	0.140
13.1	0.43	0.4944	0.0639	0.0560	0.009	0.149	0.133
14.1	0.46	0.5179	0.0604	0.0542	0.009	0.144	0.126
15.0	0.50	0.5398	0.0568	0.0521	0.009	0.138	0.119
15.9	0.53	0.5598	0.0530	0.0498	0.008	0.132	0.111
16.9	0.56	0.5779	0.0491	0.0473	0.008	0.126	0.102
17.8	0.59	0.5935	0.0451	0.0446	0.008	0.119	0.094
18.8	0.62	0.6062	0.0409	0.0418	0.007	0.111	0.085
19.7	0.65	0.6153	0.0367	0.0388	0.007	0.103	0.077
20.6	0.68	0.6191	0.0324	0.0356	0.006	0.095	0.068
21.6	0.71	0.6157	0.0280	0.0324	0.005	0.086	0.058
22.5	0.74	0.6025	0.0236	0.0291	0.005	0.077	0.049
23.4	0.77	0.5743	0.0190	0.0256	0.004	0.068	0.040
24.4	0.80	0.5238	0.0143	0.0220	0.004	0.059	0.030
25.3	0.84	0.4367	0.0096	0.0184	0.003	0.049	0.020
26.3	0.87	0.2831	0.0048	0.0147	0.002	0.039	0.010
27.2	0.90	-0.0031	0.0000	0.0111	0.002	0.029	0.000

PROP RPM = 5000

V	J	Pe	Ct	Cp	PWR	Torque	Thrust
(mph)	(Adv Ratio)				(Hp)	(In-Lbf)	(Lbf)
0.0	0.00	0.0000	0.1330	0.0889	0.029	0.369	0.434
1.1	0.03	0.0446	0.1252	0.0851	0.028	0.353	0.408
2.3	0.06	0.0879	0.1175	0.0811	0.027	0.337	0.383
3.4	0.09	0.1296	0.1099	0.0771	0.025	0.320	0.359
4.6	0.12	0.1699	0.1022	0.0729	0.024	0.303	0.333
5.7	0.15	0.2087	0.0946	0.0687	0.023	0.285	0.309
6.9	0.18	0.2459	0.0874	0.0646	0.021	0.268	0.285
8.0	0.21	0.2816	0.0847	0.0638	0.021	0.265	0.276
9.2	0.24	0.3158	0.0820	0.0629	0.021	0.261	0.267
10.3	0.27	0.3484	0.0791	0.0619	0.020	0.257	0.258
11.5	0.30	0.3796	0.0761	0.0608	0.020	0.252	0.248
12.6	0.33	0.4092	0.0730	0.0595	0.020	0.247	0.238
13.8	0.36	0.4373	0.0699	0.0581	0.019	0.241	0.228
14.9	0.39	0.4639	0.0666	0.0566	0.019	0.235	0.217
16.1	0.42	0.4891	0.0632	0.0548	0.018	0.228	0.206
17.2	0.45	0.5126	0.0598	0.0530	0.017	0.220	0.195
18.4	0.48	0.5346	0.0562	0.0510	0.017	0.212	0.183
19.5	0.52	0.5547	0.0525	0.0488	0.016	0.203	0.171
20.7	0.55	0.5729	0.0486	0.0463	0.015	0.192	0.159
21.8	0.58	0.5887	0.0446	0.0436	0.014	0.181	0.146
23.0	0.61	0.6017	0.0406	0.0409	0.013	0.170	0.132
24.1	0.64	0.6112	0.0364	0.0379	0.012	0.157	0.119
25.3	0.67	0.6155	0.0322	0.0349	0.011	0.145	0.105
26.4	0.70	0.6129	0.0278	0.0316	0.010	0.131	0.091
27.6	0.73	0.5995	0.0233	0.0283	0.009	0.117	0.076
28.7	0.76	0.5722	0.0188	0.0249	0.008	0.103	0.061
29.8	0.79	0.5236	0.0142	0.0214	0.007	0.089	0.046
31.0	0.82	0.4387	0.0096	0.0179	0.006	0.074	0.031
32.1	0.85	0.2857	0.0048	0.0142	0.005	0.059	0.016
33.3	0.88	-0.0046	-0.0001	0.0106	0.004	0.044	0.000

PROP RPM = 6000

V	J	Pe	Ct	Cp	PWR	Torque	Thrust
(mph)	(Adv Ratio)				(Hp)	(In-Lbf)	(Lbf)
0.0	0.00	0.0000	0.1326	0.0880	0.050	0.526	0.623
1.4	0.03	0.0445	0.1249	0.0841	0.048	0.503	0.587
2.7	0.06	0.0877	0.1172	0.0802	0.046	0.480	0.551
4.1	0.09	0.1295	0.1096	0.0762	0.043	0.456	0.515
5.5	0.12	0.1698	0.1020	0.0720	0.041	0.431	0.479
6.8	0.15	0.2087	0.0945	0.0679	0.039	0.406	0.444
8.2	0.18	0.2460	0.0871	0.0637	0.036	0.381	0.409
9.5	0.21	0.2820	0.0843	0.0627	0.036	0.375	0.396
10.9	0.24	0.3164	0.0815	0.0618	0.035	0.370	0.383
12.3	0.27	0.3494	0.0786	0.0607	0.035	0.363	0.369
13.6	0.30	0.3809	0.0757	0.0596	0.034	0.356	0.355
15.0	0.33	0.4110	0.0726	0.0582	0.033	0.348	0.341
16.4	0.36	0.4396	0.0694	0.0568	0.032	0.340	0.326
17.7	0.39	0.4668	0.0661	0.0552	0.031	0.330	0.311
19.1	0.42	0.4924	0.0627	0.0534	0.030	0.320	0.295
20.4	0.45	0.5166	0.0592	0.0516	0.029	0.308	0.278
21.8	0.48	0.5391	0.0557	0.0495	0.028	0.296	0.262
23.2	0.51	0.5598	0.0520	0.0474	0.027	0.283	0.244
24.5	0.54	0.5785	0.0482	0.0450	0.026	0.269	0.226
25.9	0.57	0.5946	0.0442	0.0423	0.024	0.253	0.208
27.3	0.60	0.6080	0.0402	0.0396	0.023	0.237	0.189
28.6	0.63	0.6178	0.0360	0.0367	0.021	0.219	0.169
30.0	0.66	0.6219	0.0318	0.0337	0.019	0.202	0.149
31.3	0.69	0.6188	0.0275	0.0307	0.017	0.183	0.129
32.7	0.72	0.6058	0.0231	0.0275	0.016	0.164	0.109
34.1	0.75	0.5796	0.0189	0.0244	0.014	0.146	0.089
35.4	0.78	0.5319	0.0144	0.0211	0.012	0.126	0.068
36.8	0.81	0.4460	0.0097	0.0175	0.010	0.105	0.045
38.2	0.84	0.2918	0.0049	0.0140	0.008	0.084	0.023
39.5	0.87	-0.0022	0.0000	0.0106	0.006	0.063	0.000

PROP RPM = 7000

V (mph)	J (Adv Ratio)	Pe	Ct	Cp	PWR (Hp)	Torque (In-Lbf)	Thrust (Lbf)
0.0	0.00	0.0000	0.1323	0.0866	0.078	0.705	0.846
1.6	0.03	0.0455	0.1244	0.0827	0.075	0.673	0.795
3.2	0.06	0.0897	0.1167	0.0788	0.071	0.641	0.746
4.8	0.09	0.1324	0.1090	0.0747	0.068	0.608	0.697
6.4	0.12	0.1738	0.1012	0.0705	0.064	0.574	0.647
8.0	0.15	0.2137	0.0936	0.0663	0.060	0.540	0.599
9.6	0.18	0.2521	0.0863	0.0622	0.056	0.506	0.552
11.2	0.21	0.2891	0.0835	0.0612	0.055	0.498	0.534
12.8	0.24	0.3246	0.0806	0.0601	0.054	0.489	0.515
14.4	0.27	0.3586	0.0777	0.0590	0.053	0.480	0.497
16.0	0.30	0.3911	0.0747	0.0578	0.052	0.470	0.477
17.7	0.33	0.4222	0.0716	0.0564	0.051	0.459	0.457
19.3	0.36	0.4518	0.0683	0.0549	0.050	0.447	0.437
20.9	0.39	0.4799	0.0650	0.0533	0.048	0.434	0.415
22.5	0.42	0.5063	0.0615	0.0515	0.047	0.419	0.393
24.1	0.45	0.5311	0.0580	0.0495	0.045	0.403	0.371
25.7	0.48	0.5540	0.0543	0.0475	0.043	0.386	0.347
27.3	0.51	0.5750	0.0506	0.0452	0.041	0.368	0.323
28.9	0.54	0.5938	0.0467	0.0429	0.039	0.349	0.299
30.5	0.58	0.6098	0.0428	0.0404	0.037	0.329	0.274
32.1	0.61	0.6225	0.0388	0.0378	0.034	0.307	0.248
33.7	0.64	0.6309	0.0348	0.0350	0.032	0.285	0.222
35.3	0.67	0.6329	0.0306	0.0322	0.029	0.262	0.195
36.9	0.70	0.6267	0.0263	0.0292	0.026	0.238	0.168
38.5	0.73	0.6104	0.0220	0.0262	0.024	0.214	0.141
40.1	0.76	0.5792	0.0178	0.0233	0.021	0.189	0.114
41.7	0.79	0.5253	0.0135	0.0202	0.018	0.164	0.086
43.3	0.82	0.4327	0.0090	0.0169	0.015	0.138	0.057
44.9	0.85	0.2737	0.0045	0.0140	0.013	0.114	0.029
46.5	0.88	-0.0024	0.0000	0.0113	0.010	0.092	0.000

PROP RPM = 8000

V (mph)	J (Adv Ratio)	Pe	Ct	Cp	PWR (Hp)	Torque (In-Lbf)	Thrust (Lbf)
0.0	0.00	0.0000	0.1319	0.0855	0.115	0.909	1.101
1.9	0.03	0.0467	0.1239	0.0815	0.110	0.867	1.035
3.7	0.06	0.0920	0.1161	0.0775	0.105	0.824	0.969
5.6	0.09	0.1358	0.1082	0.0734	0.099	0.781	0.904
7.4	0.12	0.1783	0.1004	0.0692	0.093	0.735	0.838
9.3	0.15	0.2193	0.0927	0.0649	0.088	0.690	0.774
11.2	0.18	0.2588	0.0856	0.0610	0.082	0.648	0.715
13.0	0.21	0.2969	0.0828	0.0599	0.081	0.637	0.691
14.9	0.25	0.3335	0.0798	0.0588	0.079	0.625	0.667
16.8	0.28	0.3686	0.0768	0.0576	0.078	0.613	0.642
18.6	0.31	0.4021	0.0738	0.0563	0.076	0.599	0.616
20.5	0.34	0.4342	0.0706	0.0549	0.074	0.584	0.589
22.3	0.37	0.4647	0.0672	0.0533	0.072	0.567	0.562
24.2	0.40	0.4936	0.0639	0.0517	0.070	0.549	0.533
26.1	0.43	0.5208	0.0603	0.0498	0.067	0.530	0.504
27.9	0.46	0.5459	0.0568	0.0479	0.065	0.509	0.474
29.8	0.49	0.5692	0.0531	0.0458	0.062	0.487	0.443
31.6	0.52	0.5903	0.0493	0.0436	0.059	0.464	0.412
33.5	0.55	0.6087	0.0455	0.0413	0.056	0.439	0.380
35.4	0.58	0.6243	0.0416	0.0389	0.053	0.414	0.348
37.2	0.61	0.6362	0.0377	0.0364	0.049	0.387	0.315
39.1	0.64	0.6432	0.0338	0.0338	0.046	0.360	0.282
40.9	0.68	0.6434	0.0297	0.0312	0.042	0.332	0.248
42.8	0.71	0.6358	0.0256	0.0285	0.038	0.303	0.214
44.7	0.74	0.6172	0.0215	0.0257	0.035	0.273	0.180
46.5	0.77	0.5829	0.0173	0.0228	0.031	0.242	0.144
48.4	0.80	0.5246	0.0130	0.0198	0.027	0.211	0.109
50.3	0.83	0.4291	0.0087	0.0169	0.023	0.179	0.073
52.1	0.86	0.2670	0.0044	0.0141	0.019	0.150	0.037
54.0	0.89	-0.0002	0.0000	0.0116	0.016	0.123	0.000

PROP RPM = 9000

V	J	Pe	Ct	Cp	PWR	Torque	Thrust
(mph)	(Adv Ratio)				(Hp)	(In-Lbf)	(Lbf)
0.0	0.00	0.0000	0.1320	0.0848	0.163	1.141	1.395
2.1	0.03	0.0478	0.1239	0.0808	0.155	1.087	1.309
4.3	0.06	0.0942	0.1159	0.0767	0.147	1.032	1.224
6.4	0.09	0.1392	0.1079	0.0725	0.139	0.976	1.140
8.5	0.12	0.1828	0.0999	0.0682	0.131	0.917	1.056
10.6	0.16	0.2249	0.0921	0.0638	0.123	0.859	0.973
12.8	0.19	0.2655	0.0853	0.0601	0.116	0.809	0.902
14.9	0.22	0.3047	0.0824	0.0590	0.113	0.794	0.871
17.0	0.25	0.3423	0.0794	0.0578	0.111	0.778	0.839
19.1	0.28	0.3784	0.0763	0.0566	0.109	0.761	0.807
21.3	0.31	0.4129	0.0732	0.0553	0.106	0.744	0.774
23.4	0.34	0.4459	0.0700	0.0538	0.103	0.724	0.740
25.5	0.37	0.4772	0.0667	0.0523	0.100	0.703	0.705
27.6	0.41	0.5068	0.0632	0.0506	0.097	0.680	0.668
29.8	0.44	0.5345	0.0597	0.0487	0.094	0.656	0.631
31.9	0.47	0.5601	0.0560	0.0468	0.090	0.630	0.592
34.0	0.50	0.5836	0.0523	0.0447	0.086	0.602	0.553
36.1	0.53	0.6046	0.0485	0.0425	0.082	0.572	0.513
38.3	0.56	0.6228	0.0447	0.0403	0.077	0.542	0.473
40.4	0.59	0.6379	0.0409	0.0380	0.073	0.511	0.433
42.5	0.62	0.6491	0.0371	0.0356	0.068	0.479	0.392
44.6	0.65	0.6547	0.0332	0.0332	0.064	0.446	0.351
46.8	0.69	0.6536	0.0292	0.0306	0.059	0.412	0.308
48.9	0.72	0.6441	0.0251	0.0279	0.054	0.376	0.265
51.0	0.75	0.6238	0.0210	0.0252	0.048	0.339	0.222
53.1	0.78	0.5846	0.0167	0.0223	0.043	0.300	0.177
55.3	0.81	0.5226	0.0126	0.0195	0.038	0.263	0.133
57.4	0.84	0.4182	0.0083	0.0167	0.032	0.224	0.088
59.5	0.87	0.2477	0.0040	0.0140	0.027	0.189	0.042
61.7	0.90	-0.0002	0.0000	0.0119	0.023	0.160	0.000

PROP RPM = 10000

V (mph)	J (Adv Ratio)	Pe	Ct	Cp	PWR (Hp)	Torque (In-Lbf)	Thrust (Lbf)
0.0	0.00	0.0000	0.1326	0.0852	0.224	1.415	1.730
2.4	0.03	0.0482	0.1243	0.0809	0.213	1.344	1.622
4.8	0.06	0.0952	0.1162	0.0766	0.202	1.273	1.516
7.1	0.09	0.1408	0.1081	0.0723	0.191	1.201	1.410
9.5	0.13	0.1850	0.1000	0.0679	0.179	1.128	1.305
11.9	0.16	0.2277	0.0920	0.0635	0.167	1.054	1.201
14.3	0.19	0.2688	0.0854	0.0599	0.158	0.995	1.115
16.7	0.22	0.3086	0.0824	0.0587	0.155	0.976	1.076
19.0	0.25	0.3468	0.0794	0.0575	0.152	0.955	1.036
21.4	0.28	0.3834	0.0763	0.0562	0.148	0.934	0.995
23.8	0.31	0.4184	0.0731	0.0548	0.145	0.911	0.954
26.2	0.35	0.4519	0.0698	0.0534	0.141	0.887	0.911
28.5	0.38	0.4836	0.0665	0.0518	0.137	0.861	0.868
30.9	0.41	0.5135	0.0631	0.0501	0.132	0.833	0.823
33.3	0.44	0.5415	0.0595	0.0483	0.127	0.802	0.776
35.7	0.47	0.5673	0.0559	0.0464	0.122	0.771	0.729
38.1	0.50	0.5909	0.0522	0.0444	0.117	0.738	0.681
40.4	0.53	0.6118	0.0485	0.0423	0.112	0.703	0.633
42.8	0.57	0.6301	0.0448	0.0402	0.106	0.668	0.585
45.2	0.60	0.6455	0.0411	0.0380	0.100	0.631	0.536
47.6	0.63	0.6570	0.0373	0.0357	0.094	0.593	0.487
50.0	0.66	0.6628	0.0334	0.0333	0.088	0.553	0.436
52.3	0.69	0.6622	0.0294	0.0307	0.081	0.510	0.384
54.7	0.72	0.6532	0.0255	0.0282	0.074	0.468	0.333
57.1	0.75	0.6328	0.0214	0.0255	0.067	0.423	0.279
59.5	0.78	0.5964	0.0173	0.0227	0.060	0.378	0.225
61.8	0.82	0.5337	0.0130	0.0200	0.053	0.332	0.170
64.2	0.85	0.4317	0.0087	0.0171	0.045	0.284	0.114
66.6	0.88	0.2529	0.0042	0.0146	0.038	0.242	0.055
69.0	0.91	-0.0011	0.0000	0.0121	0.032	0.201	0.000

PROP RPM = 11000

V	J	Pe	Ct	Cp	PWR	Torque	Thrust
(mph)	(Adv Ratio)				(Hp)	(In-Lbf)	(Lbf)
0.0	0.00	0.0000	0.1337	0.0860	0.302	1.729	2.110
2.6	0.03	0.0485	0.1253	0.0815	0.286	1.639	1.978
5.3	0.06	0.0959	0.1170	0.0770	0.270	1.549	1.847
7.9	0.09	0.1420	0.1088	0.0725	0.255	1.458	1.717
10.5	0.13	0.1867	0.1006	0.0680	0.239	1.367	1.588
13.2	0.16	0.2299	0.0925	0.0635	0.223	1.277	1.461
15.8	0.19	0.2715	0.0860	0.0600	0.210	1.206	1.358
18.4	0.22	0.3116	0.0830	0.0588	0.206	1.183	1.310
21.0	0.25	0.3502	0.0799	0.0576	0.202	1.158	1.261
23.7	0.28	0.3872	0.0767	0.0563	0.198	1.132	1.212
26.3	0.32	0.4226	0.0736	0.0549	0.193	1.105	1.161
28.9	0.35	0.4563	0.0703	0.0535	0.188	1.075	1.110
31.6	0.38	0.4884	0.0670	0.0520	0.182	1.045	1.058
34.2	0.41	0.5186	0.0636	0.0503	0.177	1.011	1.004
36.8	0.44	0.5468	0.0601	0.0485	0.170	0.976	0.948
39.5	0.47	0.5729	0.0565	0.0467	0.164	0.938	0.891
42.1	0.51	0.5969	0.0528	0.0446	0.157	0.897	0.833
44.7	0.54	0.6182	0.0490	0.0426	0.149	0.856	0.774
47.3	0.57	0.6372	0.0452	0.0403	0.141	0.810	0.714
50.0	0.60	0.6530	0.0414	0.0380	0.133	0.765	0.654
52.6	0.63	0.6645	0.0377	0.0358	0.126	0.720	0.595
55.2	0.66	0.6702	0.0337	0.0334	0.117	0.671	0.533
57.9	0.69	0.6667	0.0296	0.0308	0.108	0.619	0.467
60.5	0.73	0.6573	0.0257	0.0284	0.100	0.571	0.406
63.1	0.76	0.6414	0.0215	0.0253	0.089	0.509	0.339
65.8	0.79	0.6036	0.0172	0.0225	0.079	0.453	0.272
68.4	0.82	0.5405	0.0130	0.0197	0.069	0.397	0.205
71.0	0.85	0.4365	0.0087	0.0170	0.060	0.342	0.138
73.7	0.88	0.2861	0.0051	0.0159	0.056	0.319	0.081
76.3	0.92	0.0544	0.0008	0.0135	0.047	0.272	0.013

PROP RPM = 12000

V	J	Pe	Ct	Cp	PWR	Torque	Thrust
(mph)	(Adv Ratio)				(Hp)	(In-Lbf)	(Lbf)
0.0	0.00	0.0000	0.1336	0.0865	0.394	2.069	2.510
2.9	0.03	0.0482	0.1252	0.0818	0.373	1.957	2.352
5.7	0.06	0.0955	0.1168	0.0772	0.351	1.846	2.195
8.6	0.09	0.1416	0.1086	0.0725	0.330	1.735	2.041
11.5	0.13	0.1864	0.1004	0.0679	0.309	1.625	1.887
14.3	0.16	0.2296	0.0924	0.0634	0.289	1.517	1.736
17.2	0.19	0.2712	0.0858	0.0599	0.273	1.432	1.613
20.1	0.22	0.3114	0.0829	0.0587	0.267	1.405	1.557
22.9	0.25	0.3500	0.0798	0.0575	0.262	1.375	1.499
25.8	0.28	0.3872	0.0766	0.0561	0.256	1.343	1.439
28.7	0.32	0.4228	0.0734	0.0547	0.249	1.309	1.379
31.5	0.35	0.4578	0.0708	0.0536	0.244	1.283	1.331
34.4	0.38	0.4901	0.0675	0.0521	0.237	1.246	1.268
37.3	0.41	0.5194	0.0635	0.0501	0.228	1.199	1.194
40.1	0.44	0.5479	0.0601	0.0484	0.220	1.158	1.129
43.0	0.47	0.5743	0.0566	0.0466	0.212	1.114	1.063
45.8	0.50	0.5986	0.0529	0.0446	0.203	1.067	0.995
48.7	0.54	0.6207	0.0492	0.0425	0.194	1.016	0.925
51.6	0.57	0.6399	0.0456	0.0404	0.184	0.967	0.856
54.4	0.60	0.6563	0.0419	0.0382	0.174	0.914	0.787
57.3	0.63	0.6698	0.0377	0.0355	0.162	0.850	0.709
60.2	0.66	0.6765	0.0341	0.0334	0.152	0.799	0.641
63.0	0.69	0.6754	0.0301	0.0309	0.141	0.740	0.566
65.9	0.72	0.6649	0.0257	0.0281	0.128	0.672	0.484
68.8	0.76	0.6468	0.0220	0.0257	0.117	0.615	0.413
71.6	0.79	0.6098	0.0178	0.0230	0.105	0.549	0.334
74.5	0.82	0.5390	0.0136	0.0206	0.094	0.494	0.255
77.4	0.85	0.4495	0.0095	0.0180	0.082	0.431	0.179
80.2	0.88	0.2983	0.0053	0.0156	0.071	0.372	0.099
83.1	0.91	0.0461	0.0007	0.0132	0.060	0.316	0.013

PROP RPM = 13000

V (mph)	J (Adv Ratio)	Pe	Ct	Cp	PWR (Hp)	Torque (In-Lbf)	Thrust (Lbf)
0.0	0.00	0.0000	0.1344	0.0873	0.505	2.450	2.964
3.1	0.03	0.0485	0.1259	0.0823	0.477	2.311	2.775
6.2	0.06	0.0961	0.1175	0.0775	0.449	2.176	2.591
9.4	0.10	0.1427	0.1090	0.0727	0.421	2.040	2.405
12.5	0.13	0.1880	0.1008	0.0680	0.394	1.908	2.222
15.6	0.16	0.2317	0.0927	0.0634	0.367	1.779	2.043
18.7	0.19	0.2736	0.0863	0.0600	0.347	1.684	1.902
21.9	0.22	0.3148	0.0839	0.0591	0.342	1.660	1.850
25.0	0.25	0.3530	0.0802	0.0576	0.334	1.618	1.769
28.1	0.29	0.3904	0.0770	0.0562	0.326	1.579	1.697
31.2	0.32	0.4261	0.0739	0.0550	0.318	1.544	1.630
34.3	0.35	0.4602	0.0706	0.0535	0.310	1.501	1.556
37.5	0.38	0.4928	0.0673	0.0519	0.301	1.458	1.483
40.6	0.41	0.5247	0.0644	0.0506	0.293	1.420	1.420
43.7	0.44	0.5523	0.0604	0.0486	0.281	1.363	1.333
46.8	0.48	0.5803	0.0573	0.0470	0.272	1.319	1.264
49.9	0.51	0.6052	0.0536	0.0449	0.260	1.261	1.182
53.1	0.54	0.6277	0.0498	0.0427	0.248	1.200	1.098
56.2	0.57	0.6464	0.0456	0.0402	0.233	1.130	1.005
59.3	0.60	0.6613	0.0422	0.0385	0.223	1.080	0.931
62.4	0.63	0.6764	0.0379	0.0355	0.206	0.996	0.835
65.6	0.67	0.6815	0.0341	0.0333	0.193	0.934	0.751
68.7	0.70	0.6836	0.0298	0.0304	0.176	0.853	0.657
71.8	0.73	0.6741	0.0256	0.0277	0.161	0.778	0.565
74.9	0.76	0.6518	0.0214	0.0250	0.145	0.702	0.472
78.0	0.79	0.6120	0.0172	0.0223	0.129	0.625	0.379
81.2	0.82	0.5445	0.0128	0.0194	0.112	0.544	0.283
84.3	0.86	0.4432	0.0089	0.0171	0.099	0.480	0.195
87.4	0.89	0.2767	0.0044	0.0142	0.083	0.400	0.098
90.5	0.92	-0.0025	0.0000	0.0115	0.067	0.323	-0.001

PROP RPM = 14000

V (mph)	J (Adv Ratio)	Pe	Ct	Cp	PWR (Hp)	Torque (In-Lbf)	Thrust (Lbf)
0.0	0.00	0.0000	0.1357	0.0885	0.640	2.880	3.470
3.4	0.03	0.0485	0.1270	0.0833	0.602	2.711	3.248
6.7	0.06	0.0962	0.1185	0.0782	0.566	2.547	3.029
10.1	0.10	0.1431	0.1099	0.0732	0.529	2.383	2.810
13.5	0.13	0.1887	0.1015	0.0683	0.494	2.225	2.595
16.8	0.16	0.2328	0.0933	0.0637	0.460	2.073	2.385
20.2	0.19	0.2749	0.0868	0.0602	0.436	1.961	2.221
23.6	0.22	0.3155	0.0838	0.0591	0.427	1.924	2.144
27.0	0.25	0.3547	0.0808	0.0579	0.419	1.884	2.065
30.3	0.29	0.3921	0.0777	0.0567	0.410	1.845	1.988
33.7	0.32	0.4287	0.0746	0.0553	0.400	1.801	1.909
37.1	0.35	0.4631	0.0714	0.0539	0.390	1.755	1.826
40.4	0.38	0.4960	0.0680	0.0523	0.378	1.703	1.740
43.8	0.41	0.5269	0.0646	0.0507	0.367	1.650	1.653
47.2	0.44	0.5562	0.0612	0.0489	0.354	1.593	1.565
50.5	0.48	0.5834	0.0576	0.0470	0.340	1.531	1.472
53.9	0.51	0.6086	0.0538	0.0450	0.325	1.464	1.376
57.3	0.54	0.6317	0.0499	0.0427	0.309	1.390	1.276
60.7	0.57	0.6515	0.0462	0.0406	0.293	1.321	1.182
64.0	0.60	0.6687	0.0423	0.0382	0.276	1.243	1.082
67.4	0.64	0.6809	0.0385	0.0359	0.260	1.169	0.984
70.8	0.67	0.6811	0.0342	0.0335	0.242	1.090	0.874
74.1	0.70	0.6762	0.0301	0.0311	0.225	1.012	0.769
77.5	0.73	0.6758	0.0261	0.0282	0.204	0.919	0.668
80.9	0.76	0.6086	0.0220	0.0275	0.199	0.896	0.562
84.2	0.79	0.6146	0.0173	0.0224	0.162	0.730	0.444
87.6	0.83	0.5356	0.0138	0.0213	0.154	0.693	0.353
91.0	0.86	0.4378	0.0092	0.0181	0.131	0.590	0.236
94.3	0.89	0.2305	0.0040	0.0156	0.113	0.508	0.103
97.7	0.92	0.0083	0.0001	0.0125	0.091	0.408	0.003

PROP RPM = 15000

V (mph)	J (Adv Ratio)	Pe	Ct	Cp	PWR (Hp)	Torque (In-Lbf)	Thrust (Lbf)
0.0	0.00	0.0000	0.1368	0.0896	0.797	3.348	4.016
3.6	0.03	0.0477	0.1281	0.0842	0.749	3.148	3.761
7.1	0.06	0.0950	0.1196	0.0790	0.703	2.955	3.510
10.7	0.09	0.1414	0.1111	0.0740	0.658	2.767	3.263
14.3	0.13	0.1866	0.1028	0.0691	0.615	2.584	3.017
17.8	0.16	0.2303	0.0946	0.0645	0.574	2.410	2.778
21.4	0.19	0.2722	0.0879	0.0608	0.541	2.272	2.580
25.0	0.22	0.3126	0.0849	0.0597	0.531	2.231	2.493
28.5	0.25	0.3516	0.0818	0.0584	0.520	2.184	2.401
32.1	0.28	0.3890	0.0788	0.0572	0.509	2.138	2.312
35.7	0.31	0.4250	0.0756	0.0558	0.497	2.087	2.219
39.2	0.35	0.4595	0.0723	0.0543	0.483	2.031	2.122
42.8	0.38	0.4923	0.0690	0.0528	0.470	1.973	2.025
46.4	0.41	0.5235	0.0656	0.0512	0.455	1.912	1.927
49.9	0.44	0.5528	0.0622	0.0495	0.440	1.849	1.827
53.5	0.47	0.5805	0.0586	0.0476	0.423	1.777	1.721
57.1	0.50	0.6063	0.0549	0.0455	0.405	1.700	1.612
60.6	0.53	0.6301	0.0510	0.0432	0.384	1.613	1.496
64.2	0.56	0.6512	0.0471	0.0409	0.364	1.528	1.383
67.8	0.60	0.6694	0.0432	0.0385	0.343	1.439	1.269
71.3	0.63	0.6796	0.0397	0.0367	0.326	1.371	1.165
74.9	0.66	0.6935	0.0352	0.0334	0.298	1.250	1.033
78.5	0.69	0.6958	0.0310	0.0308	0.274	1.150	0.910
82.0	0.72	0.6889	0.0268	0.0281	0.250	1.049	0.786
85.6	0.75	0.6695	0.0225	0.0253	0.225	0.945	0.660
89.2	0.78	0.6331	0.0181	0.0225	0.200	0.839	0.532
92.7	0.82	0.5716	0.0137	0.0195	0.174	0.730	0.402
96.3	0.85	0.4664	0.0093	0.0168	0.150	0.630	0.272
99.9	0.88	0.3029	0.0050	0.0144	0.128	0.538	0.146
103.4	0.91	0.0004	0.0000	0.0108	0.096	0.403	0.000

PROP RPM = 16000

V (mph)	J (Adv Ratio)	Pe	Ct	Cp	PWR (Hp)	Torque (In-Lbf)	Thrust (Lbf)
0.0	0.00	0.0000	0.1377	0.0907	0.979	3.856	4.599
3.8	0.03	0.0478	0.1289	0.0851	0.918	3.618	4.304
7.7	0.06	0.0953	0.1201	0.0796	0.860	3.387	4.013
11.5	0.09	0.1421	0.1116	0.0744	0.803	3.164	3.727
15.3	0.13	0.1877	0.1031	0.0694	0.749	2.952	3.444
19.1	0.16	0.2317	0.0948	0.0646	0.698	2.748	3.168
23.0	0.19	0.2739	0.0882	0.0610	0.659	2.595	2.946
26.8	0.22	0.3145	0.0853	0.0600	0.647	2.550	2.850
30.6	0.25	0.3537	0.0822	0.0587	0.634	2.497	2.745
34.5	0.28	0.3915	0.0790	0.0574	0.620	2.441	2.640
38.3	0.32	0.4275	0.0758	0.0560	0.605	2.381	2.532
42.1	0.35	0.4621	0.0726	0.0546	0.589	2.321	2.425
45.9	0.38	0.4955	0.0692	0.0530	0.572	2.252	2.313
49.8	0.41	0.5272	0.0658	0.0512	0.553	2.179	2.198
53.6	0.44	0.5560	0.0625	0.0497	0.537	2.115	2.089
57.4	0.47	0.5838	0.0590	0.0478	0.517	2.035	1.970
61.2	0.51	0.6098	0.0550	0.0456	0.492	1.939	1.838
65.1	0.54	0.6346	0.0511	0.0432	0.467	1.838	1.707
68.9	0.57	0.6559	0.0471	0.0408	0.441	1.736	1.574
72.7	0.60	0.6741	0.0433	0.0385	0.416	1.638	1.445
76.6	0.63	0.6885	0.0391	0.0359	0.388	1.527	1.308
80.4	0.66	0.6958	0.0350	0.0333	0.360	1.418	1.169
84.2	0.69	0.6952	0.0307	0.0306	0.331	1.303	1.024
88.0	0.73	0.6850	0.0264	0.0280	0.302	1.190	0.882
91.9	0.76	0.6623	0.0220	0.0252	0.272	1.073	0.736
95.7	0.79	0.6243	0.0175	0.0222	0.239	0.943	0.585
99.5	0.82	0.5581	0.0124	0.0182	0.197	0.776	0.414
103.4	0.85	0.4407	0.0085	0.0165	0.178	0.700	0.284
107.2	0.88	0.2545	0.0039	0.0135	0.146	0.575	0.130
111.0	0.92	-0.0548	-0.0007	0.0109	0.117	0.462	-0.022

PROP RPM = 17000

V (mph)	J (Adv Ratio)	Pe	Ct	Cp	PWR (Hp)	Torque (In-Lbf)	Thrust (Lbf)
0.0	0.00	0.0000	0.1368	0.0908	1.176	4.359	5.158
4.0	0.03	0.0471	0.1281	0.0851	1.102	4.087	4.829
8.1	0.06	0.0940	0.1195	0.0797	1.032	3.825	4.505
12.1	0.09	0.1402	0.1110	0.0744	0.964	3.574	4.186
16.1	0.13	0.1852	0.1026	0.0694	0.899	3.334	3.870
20.2	0.16	0.2287	0.0945	0.0647	0.838	3.107	3.563
24.2	0.19	0.2705	0.0878	0.0610	0.790	2.928	3.309
28.2	0.22	0.3109	0.0848	0.0598	0.775	2.873	3.198
32.3	0.25	0.3491	0.0818	0.0587	0.761	2.821	3.085
36.3	0.28	0.3875	0.0786	0.0572	0.741	2.747	2.965
40.4	0.31	0.4236	0.0755	0.0558	0.723	2.680	2.846
44.4	0.34	0.4581	0.0723	0.0544	0.704	2.612	2.726
48.4	0.38	0.4911	0.0690	0.0528	0.684	2.535	2.601
52.5	0.41	0.5229	0.0656	0.0511	0.661	2.452	2.472
56.5	0.44	0.5526	0.0622	0.0494	0.639	2.370	2.344
60.5	0.47	0.5810	0.0586	0.0474	0.614	2.275	2.209
64.6	0.50	0.6071	0.0550	0.0454	0.588	2.180	2.073
68.6	0.53	0.6319	0.0511	0.0430	0.557	2.066	1.925
72.6	0.56	0.6541	0.0471	0.0406	0.526	1.950	1.776
76.7	0.60	0.6714	0.0433	0.0384	0.497	1.842	1.632
80.7	0.63	0.6899	0.0388	0.0352	0.456	1.691	1.462
84.7	0.66	0.6903	0.0353	0.0336	0.435	1.614	1.330
88.8	0.69	0.6893	0.0311	0.0311	0.403	1.495	1.174
92.8	0.72	0.6813	0.0269	0.0284	0.368	1.364	1.013
96.8	0.75	0.6540	0.0230	0.0265	0.343	1.272	0.869
100.9	0.78	0.6386	0.0172	0.0210	0.273	1.010	0.647
104.9	0.81	0.5723	0.0126	0.0179	0.232	0.861	0.475
108.9	0.85	0.4562	0.0080	0.0148	0.192	0.710	0.301
113.0	0.88	0.2496	0.0033	0.0117	0.152	0.563	0.126
117.0	0.91	0.0018	0.0000	0.0128	0.166	0.617	0.001

PROP RPM = 18000

V (mph)	J (Adv Ratio)	Pe	Ct	Cp	PWR (Hp)	Torque (In-Lbf)	Thrust (Lbf)
0.0	0.00	0.0000	0.1375	0.0917	1.410	4.938	5.812
4.2	0.03	0.0464	0.1287	0.0859	1.320	4.623	5.442
8.5	0.06	0.0927	0.1201	0.0803	1.234	4.321	5.077
12.7	0.09	0.1385	0.1116	0.0749	1.152	4.034	4.720
16.9	0.12	0.1832	0.1033	0.0699	1.074	3.762	4.367
21.1	0.15	0.2265	0.0952	0.0651	1.001	3.505	4.024
25.4	0.19	0.2679	0.0882	0.0612	0.941	3.293	3.727
29.6	0.22	0.3080	0.0852	0.0600	0.923	3.232	3.604
33.8	0.25	0.3467	0.0823	0.0588	0.904	3.167	3.478
38.0	0.28	0.3840	0.0792	0.0575	0.884	3.097	3.349
42.3	0.31	0.4201	0.0762	0.0562	0.864	3.025	3.221
46.5	0.34	0.4546	0.0730	0.0548	0.842	2.947	3.087
50.7	0.37	0.4876	0.0698	0.0532	0.818	2.865	2.950
54.9	0.40	0.5155	0.0668	0.0522	0.802	2.808	2.822
59.2	0.43	0.5484	0.0631	0.0499	0.768	2.688	2.668
63.4	0.46	0.5774	0.0596	0.0480	0.737	2.581	2.518
67.6	0.50	0.6048	0.0558	0.0457	0.703	2.462	2.358
71.8	0.53	0.6301	0.0519	0.0434	0.667	2.336	2.194
76.1	0.56	0.6517	0.0482	0.0412	0.634	2.219	2.036
80.3	0.59	0.6687	0.0442	0.0389	0.598	2.095	1.869
84.5	0.62	0.6749	0.0402	0.0369	0.568	1.987	1.700
88.7	0.65	0.6880	0.0363	0.0343	0.528	1.849	1.535
93.0	0.68	0.6872	0.0325	0.0322	0.496	1.735	1.374
97.2	0.71	0.6830	0.0281	0.0293	0.450	1.576	1.186
101.4	0.74	0.6698	0.0226	0.0251	0.386	1.353	0.957
105.6	0.77	0.6329	0.0192	0.0235	0.362	1.266	0.813
109.9	0.81	0.5669	0.0148	0.0211	0.324	1.134	0.627
114.1	0.84	0.4748	0.0107	0.0189	0.291	1.019	0.454
118.3	0.87	0.3353	0.0063	0.0162	0.249	0.871	0.264
122.5	0.90	0.0004	0.0000	0.0099	0.152	0.533	0.000

PROP RPM = 19000

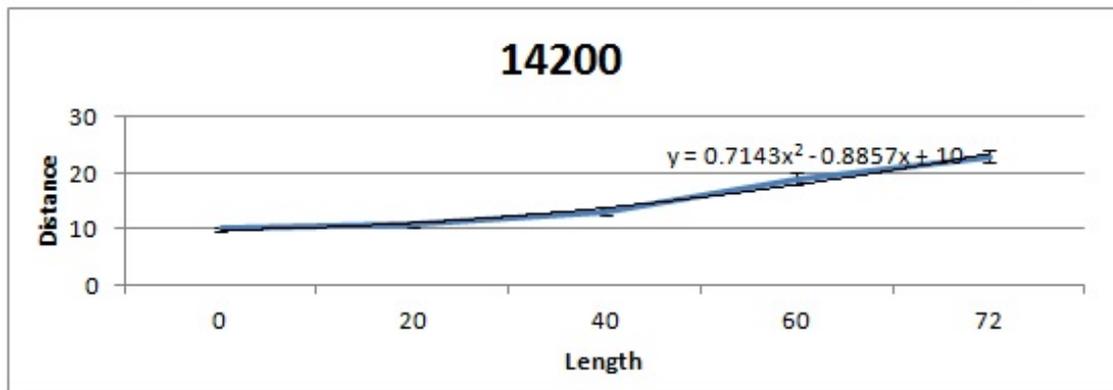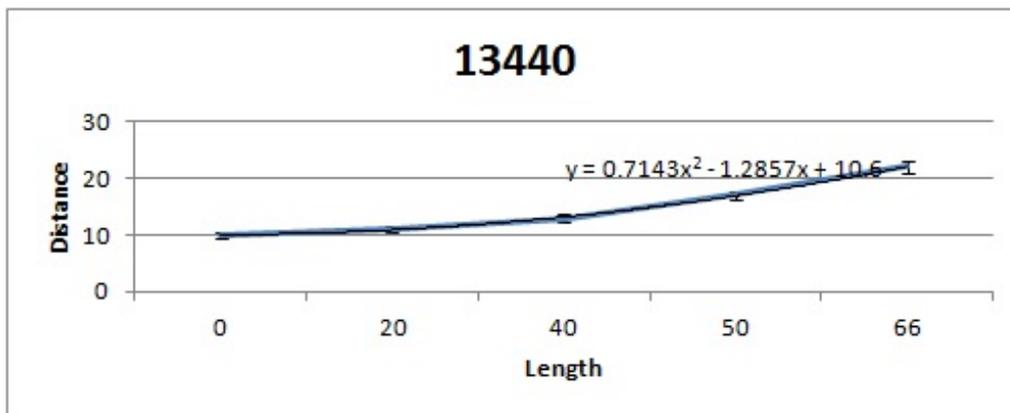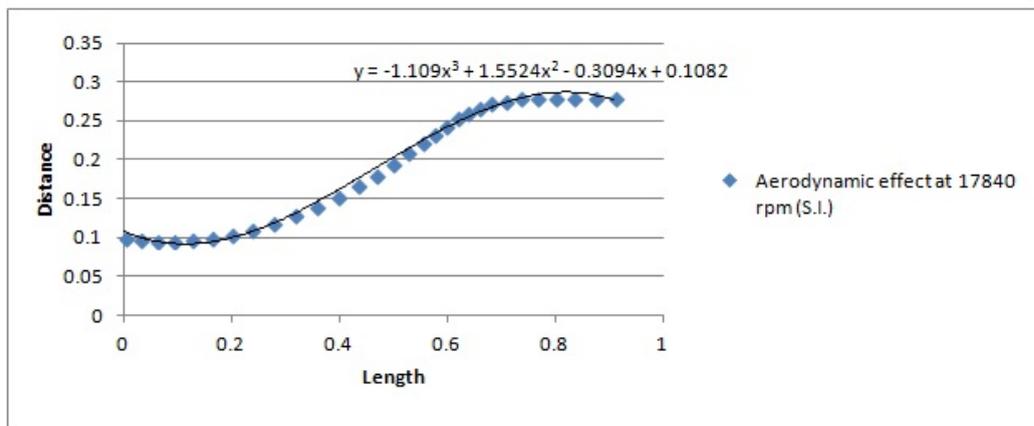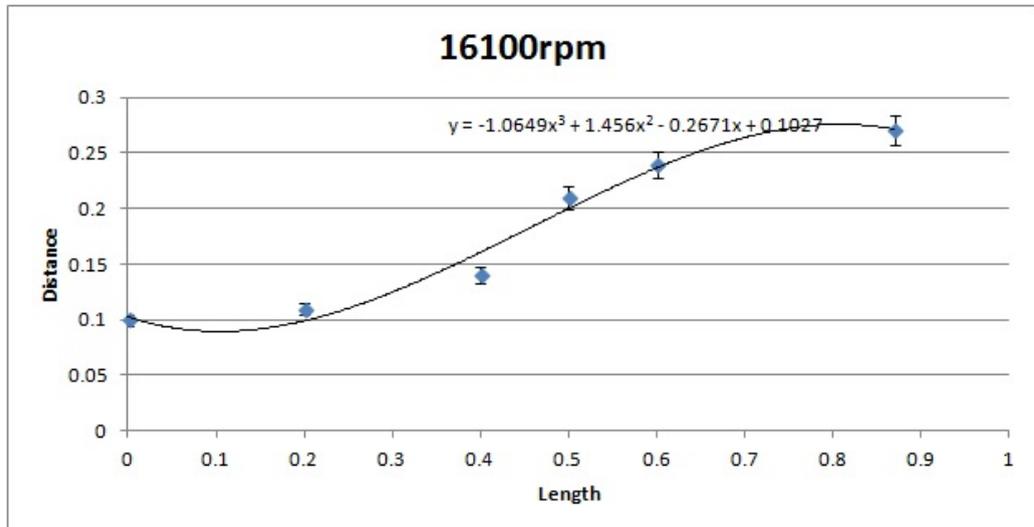| V	J	Pe	Ct	Cp	PWR	Torque	Thrust
(mph)	(Adv Ratio)				(Hp)	(In-Lbf)	(Lbf)
0.0	0.00	0.0000	0.1383	0.0924	1.671	5.544	6.515
4.5	0.03	0.0465	0.1295	0.0866	1.565	5.191	6.098
9.0	0.06	0.0928	0.1208	0.0809	1.463	4.853	5.689
13.4	0.09	0.1386	0.1122	0.0755	1.365	4.529	5.286
17.9	0.12	0.1834	0.1037	0.0704	1.272	4.220	4.887
22.4	0.16	0.2268	0.0955	0.0655	1.184	3.927	4.500
26.9	0.19	0.2684	0.0885	0.0615	1.112	3.690	4.169
31.3	0.22	0.3086	0.0855	0.0603	1.091	3.619	4.029
35.8	0.25	0.3472	0.0826	0.0592	1.070	3.550	3.891
40.3	0.28	0.3846	0.0795	0.0579	1.047	3.472	3.747
44.8	0.31	0.4206	0.0764	0.0565	1.021	3.388	3.599
49.2	0.34	0.4551	0.0732	0.0551	0.995	3.302	3.450
53.7	0.37	0.4880	0.0700	0.0535	0.967	3.209	3.296
58.2	0.40	0.5187	0.0667	0.0520	0.940	3.117	3.141
62.7	0.44	0.5478	0.0633	0.0503	0.910	3.019	2.984
67.1	0.47	0.5773	0.0596	0.0482	0.871	2.890	2.810
71.6	0.50	0.6053	0.0559	0.0459	0.830	2.755	2.632
76.1	0.53	0.6312	0.0518	0.0434	0.785	2.604	2.442
80.6	0.56	0.6541	0.0480	0.0411	0.743	2.463	2.261
85.0	0.59	0.6734	0.0441	0.0387	0.700	2.323	2.080
89.5	0.62	0.6887	0.0401	0.0362	0.655	2.171	1.888
94.0	0.65	0.6987	0.0359	0.0335	0.606	2.012	1.690
98.5	0.68	0.7000	0.0317	0.0310	0.561	1.861	1.495
102.9	0.72	0.6939	0.0273	0.0282	0.510	1.690	1.288
107.4	0.75	0.6755	0.0226	0.0250	0.452	1.498	1.065
111.9	0.78	0.6397	0.0186	0.0226	0.409	1.356	0.876
116.4	0.81	0.5792	0.0140	0.0196	0.354	1.174	0.661
120.9	0.84	0.4802	0.0095	0.0166	0.300	0.994	0.447
125.3	0.87	0.3144	0.0049	0.0137	0.247	0.819	0.232
129.8	0.90	0.0018	0.0000	0.0105	0.190	0.630	0.001

Appendix D

Appendix Experimental Data

This Appendix includes the information given from the experiment made from the fellow college Zoi Trachana, in CSL under the supervision of Prof. Kostas J. Kyriakopoulos. The following curves (S.I.) represent the aerodynamic effects of the air flow at the exodus of the *Asctec Firefly* rotor.

6600rpm

$y = -0.5x^2 + 4.1x + 6.5$

7360rpm

$y = 0.75x^2 - 1.85x + 11.25$

8500rpm

$y = 0.75x^2 - 1.05x + 10.25$

Aerodynamic effect at 10400rpm (S.I.)

$y = 0.0086x^2 - 0.0274x + 0.122$

11160rpm

$y = 0.4286x^2 - 0.1714x + 9.6$

12300rpm

$y = 0.2857x^2 + 1.0857x + 8.6$

13440

$y = 0.7143x^2 - 1.2857x + 10.6$

14200

$y = 0.7143x^2 - 0.8857x + 10$

14960rpm

$y = 1E\text{-}14x^2 + 3.1429x + 6$

16100rpm

$y = -1.0649x^3 + 1.456x^2 - 0.2671x + 0.1027$

$y = -1.109x^3 + 1.5524x^2 - 0.3094x + 0.1082$

♦ Aerodynamic effect at 17840 rpm (S.I.)

Bibliography

[1] Stability of small-scale uav helicopters and quadrotors with added payload mass under pid control, autonomous robots, vol. 33, pp. 129–142, 2012, 10.1007/s10514-012-9280-5. [online]. available: http://dx.doi.org/10.1007/s10514-012-9280-5. Technical report.

[2] N. Michael D. Mellinger, M. Shomin and V. Kumar. "cooperative grasping and transport using multiple quadrotors," in proceedings of the international symposium on distributed autonomous robotic systems, nov 2010. Technical report.

[3] J.G. Leishman. Principles of helicopter aerodynamics, 2nd ed. cambridge university press, cambridge. 2006.

[4] S. Newman. *The Foundations of Helicopter Flight. Edward Arnold, London.* 1994.

[5] R. Prouty. *Helicopter Performance, Stability and Control, reprint edition with corrections. Krieger, Florida.* 1990.

[6] Mahony R. Gresham J. Corke P. Pounds, P. and J. Roberts. Towards dynamically-favourable quad-rotor aerial robots. in: Proc. australasian conference on robotics and automation, canberra. December 2004.

[7] Mahony R. Pounds, P. and P. Corke. Modelling and control of a quadrotor robot. in: Proc. australasian conference on robotics and automation. 2006.

[8] Gareth D. Padfield. *Helicopter Flight Dynamics.Blackwell, Oxford.* 1996.

[9] A.R.S. Bramwell. *Helicopter Dynamics. Arnold, London.* 1976.

[10] Bramwell A.R.S. Balmford D. and Done G. *Bramwell Helicopter Dynamics, 2nd ed. Butterworth-Heinemann, Oxford.* 2001.

[11] J.B. Wheatley. Technical report 487: An aerodynamic analysis of the autogiro rotor with a comparison between calculated and experimental results. Technical report, NACA.

[12] Alastair K. Cooke and Eric W.H. Fitzpatrick. Helicopter test and evaluation. black-well and qinetiq, oxford. 2002.

[13] Feingold A. M. Coleman, R. P. and C. W. Stempin. *ARR L5E10: Evaluation of the Induced-Velocity Field of an Idealized Helicopter Rotor, NACA.* 1945.

[14] R.T.N. Chen. A survey of nonuniform inflow models for rotorcraft flight dynamics and control applications. nasa. 1989.

[15] J. Wolkovitch. Analytical prediction of vortex-ring boundaries for helicopters in steep descents. journal of the american helicopter society, 17, no. 3. July 1972.

[16] VICENTE MARTÍNEZ MARTÍNEZ. Modelling of the flight dynamics of a quadro-tor helicopter. Master's thesis, CRANFIELD UNIVERSITY SCHOOL OF ENGI-NEERING Departement of aerospace sciences.

[17] W. Johnson. *Helicopter Theory. Princeton University Press, Princeton.* 1980.

[18] C. Young. A note on the velocity induced by a helicopter rotor in the vortex ring state, rae technical report 78125. Technical report.

[19] J.L. López Ruiz. *Helicópteros. Teoría y Diseño Conceptual. Escuela Técnica Supe-rior de Ingenieros Aeronáuticos, Madrid.* 1993.

[20] J.L. Espino, A. Cuerva, and J. Meseguer. *Helicópteros y Aeronaves Diversas II. Escuela Técnica Superior de Ingenieros Aeronáuticos, Madrid.* 2004.

[21] K.X. Giannakoglou. *Optimization Methods for Aerodynamics. National Technical University of Athens. Mechanical Engineer Department.* 2006.

[22] R. Fletcher. *Practical Methods of Optimization. Second edition. A Wiley-Interscience Publication. University of Dundee,Scotland,UK.* May 2000.

[23] C. Castillo R. Minguez D. Ortigosa E. Castillo, A.J. Conejo. Perturbation approach to sensitivity analysis in mathimatical programming. Technical report.

[24] Manolis I.A. Lourakis Thedore Papadopoulo. Estimating the jacobian of the sin-gular value decomposition:theory and applications. Technical report, INRIa Sophia Antipolis 2004 Route des Lucioles, BP 93 06902 SOPHIA ANTIPOLIS Cedex, FRANCE.

[25] James Arvo. "fast random rotation matrices", in david kirk, graphics gems iii, san diego: Academic press professional, pp. 117–120, isbn 978-0-12-409671-4. Technical report.

[26] Timothy W. McLain Randal W. Beard. *Small Unmanned Aircraft, Theory and Practice. PRINCETON UNIVERSITY PRESS Princeton and Oxford. ISBN 978-0-691-14921-9.* 2012.

[27] R.J.Beckman M.D.McKay and W.J.Conover. *A comparison of three methods for selecting values of input variables in the analysis of output from a computer code, Technometrics 42(1),pp. 55-61.*