ANNUAL PROGRESS SEMINAR REPORT

on

Path Planning of an Unmanned Aerial Vehicle (UAV) with Minimum Energy Consumption

Submitted by

Paraj Ganchaudhuri
(Roll No. 206102023)

Under the guidance of
Dr. Chayan Bhawal

DEPARTMENT OF ELECTRONICS AND ELECTRICAL ENGINEERING
INDIAN INSTITUTE OF TECHNOLOGY GUWAHATI

May 2023

Abstract

Unmanned Aerial Vehicle (UAV) or drone constitutes a new workforce in modern transportation system for delivery and surveillance operations. There are various other use cases for an UAV but a fundamental drawback in its application is its limited flight time. One of the well-known problem in this area is optimal path planning of an UAV with minimal energy and or minimal time of operation. Energy/Power consumption modelling of an UAV is a critical tool for evaluating path planning of a UAV. However, multiple parameters add to the power, which is not easy to identify and track in general. As a result, there are inaccuracies in the calculation of power. In this research, we attempt to map the dynamics/environment and maneuvers of a UAV to the angular speeds of the rotors, as rotors are the end consumers of power in a UAV. Using the speeds, we try to obtain a power model to calculate the energy required for a trip. The power model will also be able to track dynamic uncertainties. We also intend to redesign a path planning algorithm that can handle dynamically evolving environments and provide the operator with optimal routes even when the UAV faces external disturbances, or there is a change in the objective of the UAV in mid-flight.

List of Tables

List of Figures

Contents

Chapter 1

Introduction

Over the years, Unmanned Aerial Vehicles (UAVs) or drones have found widespread application in robotics, viz., surveillance in remote areas, precision farming, logistics, emergency responsiveness, etc. Such applications of UAVs have led to a massive boom in the UAV market worldwide. In India, The Civil Aviation Ministry estimates India's drone sector to achieve a total turnover of Rs. 120-150 billion (US$ 1.63-2.04 billion) by 2026 [1]. The major areas where UAVs are projected to make massive footprints are:
1. Surveillance - India has 15,106.7 km of international land border and a coastline of 7,516.6 km, which are potentially important for surveillance. Present surveillance infrastructure is both fuel-inefficient and workforce intensive. So, an immense space is yet to be filled by UAVs [2].
2. Precision Agriculture - India has arable lands with over 155 million hectares, and it amounts to approximately $265 billion worth of revenue. UAVs can be an essential tool to provide leading-edge digital and precision agriculture technologies to farmers to maximize the efficient use of water, fertilizer and pesticides and improve overall productivity, quality, and yield [3].
Such widespread applications have also led to a bureaucratic push in the country to promote UAVs' use in various sectors. Hence, licensing and commercial use rules and regulations have been preferentially eased out [4].
Considering the importance of this industry, research into UAVs for various applications has been actively pursued by the research community[5]. One of the well-known problems concerning UAV operations is the path planning of a UAV with minimal energy consumption. A Power Consumption Model (PoCM) is used to estimate the endurance of a UAV, like its flying time, range, speed, payload limits, battery capacity, etc. and in most practical scenarios it is very tough to derive a perfect model because there are multiple parameters at play which contribute to power consumption. Also, these parameters are difficult to identify and track in general. So, an accurate PoCM plays a crucial role in planning the best possible path or trajectory in terms of energy being spent for a given operation.

In this research, we aim to address three aspects concerning the path planning of a quadrotor UAV. First, we want to derive a Power Consumption Model (PoCM) that can accurately predict the power consumed by a quadrotor in-flight. Second, using the obtained PoCM, we aim to perform path planning given an operation/objective such that the energy consumed in a trip is minimized. Third, we plan to devise an algorithm for path-planning in a multi-agent setup and dynamically changing environment such that a given objective is achieved with minimum power consumed by each agent.

In this phase of work, we have formulated a power model for a quadrotor in a simulation environment. The model whose proceedings are enumerated in Section 2.6 reflects how the quadrotor consumes power for a predefined trajectory. The model is derived with a motivation that power consumed in a trip is dynamic and depends on the maneuvers a quadrotor undertakes in following a predefined trajectory. The fundamental maneuvers in a quadrotor flight are roll, pitch, yaw, ascend, descend and hover, and all other maneuvers are a combination of these maneuvers. We claim that tracking power based on maneuvers will lead to accurate power estimates. To prove this hypothesis, we have performed a few practical field experiments highlighted in Section 3. Then, to step into path planning formulations, we have used our model and

showed how it can evaluate trajectories based on power.
This report is organized as follows: In Chapter 2, a literature review on Power consumption modelling followed by the possible research gaps are outlined. Then we discuss the principles of quadrotor dynamics and control. Next, we introduce our Power Consumption formulation and show some simulation results of the power calculation of a quadrotor given a predefined trajectory. In Chapter 3, we present the power consumption results using a DJI Air 2 drone and highlight why our approach can be accurate in estimating the power. In Chapter 4, we discuss Path planning, including a brief literature review and possible research gaps and show the power attributes of quadrotors based on a choice of trajectory..

Chapter 2

Power consumption Modelling

The power consumption model is a tool for evaluating how expensive a path or trajectory of an aerial vehicle is in terms of energy or time. It identifies optimum paths and checks their feasibility before the vehicle sets flight. In the following section, we outline notable Power Consumption models available in the literature.

2.1 Literature Survey on Power Consumption Modeling

- D'Andrea in [6] provides an influential contribution to model power consumption of an UAV by translating the fundamental flight principles of manned aircraft to a model for the much smaller scale of unmanned aerial drones. [6] presents a model using an integrated approach that combines aerodynamic and drone design aspects into a single critical parameter: the lift-to-drag ratio r. The energy model also includes a fixed component for avionics power. Using the fact that a high-end lithium-ion battery has a specific power of 0.35 kW/kg, the paper proposes a metric to obtain the required battery weight for a trip. The worst-case energy requirement in kWh is formulated and economics i.e the average energy cost per kilometer is also addressed. Lastly, the average battery cost per km is approximated for a trip given a payload weight and flight velocity.

- Dorling et al. in [7] provides an equation for the power that is consumed by a multirotor helicopter in hover as a function of the battery and payload weight. The approach for this modeling is based on helicopter operations, with the assumption that the power consumed during level flight, takeoff, or landing is approximately equivalent to the power consumed while hovering as the power consumed by the helicopter is often reduced due to translational lift, a phenomenon where air flowing horizontally along the rotor generates additional lift. This power consumption model does not consider the role of UAV speed. These authors also report field experiments on a 3D Robotics ArduCopter Hexa-B hexacopter. and develop regression parameters with small payloads.

- In [8], Liu et al. derive a power consumption model where power is distributed into three components namely induced power, profile power, and parasitic power. The power generated by the propellers to balance the weight of the UAV in flight is represented by induced power. Profile power overcomes the rotational drag encountered by rotating propeller blades. The parasite power resists body drag when there is relative translational motion between the vehicle and wind. These power equations are derived analytically using aerodynamic principles and then encapsulated with fewer parameters. The values of these parameters are obtained by conducting field tests on an IRIS+ UAV. A least-square fit for power vs payload is also obtained. It is shown that ascending takes 9.8% more power than hovering, and descending takes 8.5% less power than hovering.

- Kirchstein in [9] derives component-based model similar to [8] but specifically for a drone delivery application. He describes characteristic power consumption calculations for takeoff and ascent, steady level flight, descent, hovering, and landing. The return trip is considered similar but without the payload. Kirchstein also compares delivery with trucks versus UAVs in Berlin and shows that

drone delivery often requires more energy. The article also shows how wind and drone hovering increase the power consumption of an UAV. It is concluded that in rural settings with long distances between customers, UAV-based parcel delivery infrastructure has comparable energy considerations to a delivery system with electric trucks.

- The authors in [10] studied a truck-drone hybrid delivery system. In this study, they extend vehicle routing models to the hybrid delivery systems by taking into account two important practical issues: the effect of parcel weight on drone energy consumption and restricted flying areas. The power consumption model is derived from [7] and tweaked to obtain the flight time of an UAV for an operation. They obtain a formulation where the variation in flight time is obtained based on the variation in the payload. The model is tested against practical experiments on an MK8-3500 standard UAV and it is claimed that the results are realistic.

- Tseng et al. in [11] uses a black-box modeling approach to obtain the power consumption model of an UAV. Field experiments are performed by the researchers considering the impact of various flight scenarios, payload weight and wind to study how power consumption varies with them, and then a non-linear regression model is presented. They obtain the power model where horizontal and vertical speed and acceleration, as well as payload mass and wind speed, are the variables. It is also mentioned that the error in estimation of power consumption in the field experiments is within 0.4% of the original power consumption obtained from the onboard sensors.

- In [12], Stolaroff et al. develops a two-component model based on the thrust required to balance the UAV weight and the parasite drag force. The power consumption model is derived for hovering from helicopter dynamics like [7] and then extended for steady flight for variation with significant velocity or in significant wind. The minimum power requirement then changes somewhat depending on the airspeed and incident angle at which the UAV traverses. This model is used to assess the energy use and life cycle greenhouse gas (GHG) emissions for small drones with short ranges (4 km) delivering from warehouses. Results indicate that small drones are likely to provide lower lifecycle GHG emissions than conventional delivery trucks, but that benefits depend on the carbon intensity of electricity and the size of the drones.

- Zhang et. al in [13] studies the significant power models available in literature and reviews, classifies the drone energy, consumption models. They document very wide variations in the modeled energy consumption rates resulting from differences in the scopes and features of the models, the specific designs of the drones; and the details of their assumed operations and uses. They also try to obtain the optimum value of payload weight and flight speed that would minimize the energy consumed per unit distance on a few standard energy consumption models.

Based on the literature surveyed, we enlist the following research gaps.

2.2 Research Gap on Power Consumption Modelling

1. An UAV design combines the attributes of airplanes and helicopters. While airplanes are designed to travel long distances efficiently, helicopters are designed to hover efficiently [9]. In most cases, the power consumption modeling is derived from hovering action in Helicopters [14] as evident from the works done in [6], [12], [7], [10]. Some works like [12] have extrapolated the dynamics of hovering to capture the power consumption during leveled flight with constant velocity. An UAV operation involves stages of leveled flight, so deriving power from hovering and then extrapolating it for the

entire flight can be a reason why the power consumption model cannot accurately capture the total power that is expended during an UAV operation.

2. There is a consideration where the power consumed during hover is an upper bound to the power consumed for other instances of the flight i.e take-off/landing/leveled translational motion etc [7]. This is based on the phenomenon of translational lift [15], a phenomenon that occurs when the UAV is in a horizontal motion and air flowing horizontally along the rotor generates additional lift. This intuition is not ideally true as the UAV is not at the same attitude for the entire duration of the flight and hence the rotors does not cross the air horizontally leveled at all instances. This consideration can be a reason for the inaccuracy in a power consumption model of an UAV.

3. In literature, none of the power consumption models that are designed analytically gives an optimum horizontal velocity that would minimize the power consumption for an UAV during leveled flight. The work done in [13] obtains the value of that airspeed intuitively and with rigorous flight tests. Moreover solution for an optimum vertical velocity during take-off/landing, and an optimum angle for ascend/desced can also be looked for. In literature, a standard angle for ascend/descend is considered as 45^o [9] but it is not verified if such an angle minimizes the power consumption.

4. Based on the literature surveyed, there is a wide variation in the power models with divergent power consumption values for essentially the same UAV operation. As seen in [13], the same UAV with the same operating settings shows wide variations in E_{pm}(Energy consumed per unit distance) values and range values with different power consumption models. These differences give a strong implication that we don't have an accurate or a benchmark power consumption model. The differences are because of different scopes and features of different models, different designs of UAVs being modeled and different assumptions in operating conditions. Thus, current research has not reached a consensus on standards for UAV power consumption which is why existing models do not reflect UAV operations very accurately.

5. It is observed that different models use a different set of parameters for the power consumption modeling. Although different models are based on different philosophies but there is no metric to characterize the crucial parameters essential for power consumption modeling. For example model in [8] considers the effect of vertical velocity while the model in [12] does not consider the effect of vertical velocity whereas the model in [6] does not consider velocity at all. The discrepancies in power models maybe because of the lack of essential parameters in modeling. [8] considers a plethora of parameters initially but while deriving the power model, multiple parameters are lumped as a single variable which takes away the essence of considering multiple parameters at the first place.

6. None of the power consumption models that are derived from aerodynamic principles consider acceleration, either vertical or horizontal acceleration as a parameter for power consumption modeling. On the contrary, the UAV is subjected to changes in acceleration at various instants of the flight path like take-off/landing, undertaking turns etc. Such instances involve a change in velocity which introduces a change in acceleration. The velocity is considered static in most models due to a lack of means in obtaining the velocity in real-time [8] or due to oversimplification where leveled flights are considered as straight-line paths with constant velocities. In fact, some models do not consider the role of velocity at all [6].

Our aim is to design an accurate power consumption model for a quadrotor (an UAV with 4 thrusters) and to arrive to it's formulation, we dicuss few prelimenaries required to understand the dynamics and control of a quadrotor.

2.3 Prelimenaries

We dive into the fundamental concepts required to describe three dimensional motion of a quadrotor.

○ Reference Frames

The position and orientation of rigid objects flying through the space are described by rigid body transformations. Here, we highlight the basic tools required for three dimensional displacements. The key concept is that of a reference frame. We associate a reference frame with each position and orientation. In each reference frame, there are three linearly independent basis vectors. In figure 2.1, vectors a_1, a_2, a_3 are attached to frame A and vectors b_1, b_2, b_3 are attached to frame B. Now, any vector in three dimensional space can be written as a linear combination of these independent basis vectors.

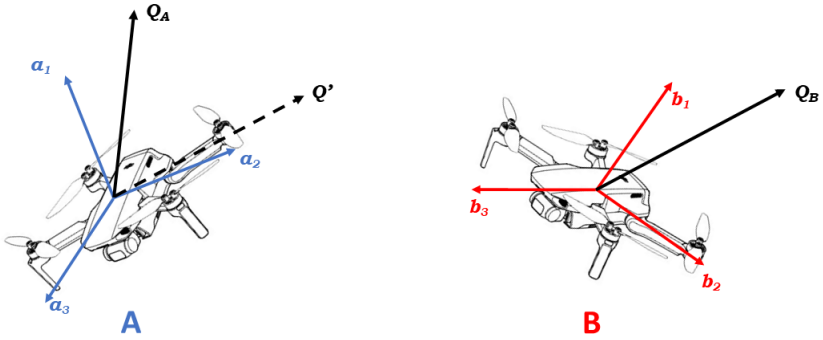

Figure 2.1: Reference frames attached to the quadrotor body

And then we discuss transformations. We use transformations to think about how vectors in one frame can be written in another frame.

○ Transformations and Displacements

Transformations describe the relationship between reference frames attached to different rigid bodies whereas displacements describe the relationships between two positions and orientations of a frame attached to a displaced rigid body.

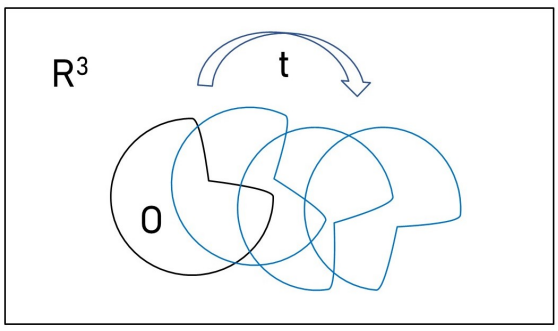

Figure 2.2: Rigid Body Displacement

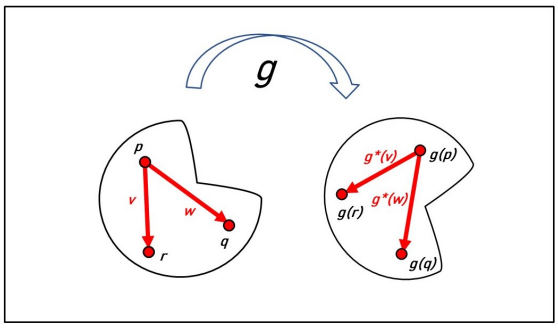

Figure 2.3: Rigid Body Motion

Consider an object $O \in \mathbb{R}^3$(see Figure 2.2).
Rigid body displacement is a map g such that

$$g : O \mapsto \mathbb{R}^3$$

Similarly, rigid body motion (see Figure 2.3) is a continuous family of time parameterised maps $g(t)$ such that

$$g(t) : O \mapsto \mathbb{R}^3$$

The necessary conditions for a rigid body displacement is that the length between any two points within the rigid body and the cross product between any two vectors attached to the rigid body remains same after displacement.
With reference to a rigid body displacement shown in Figure 2.3, we have

$$\|g(p) - g(q)\| = \|p - q\|$$
$$g*(v) \times g*(w) = g*(v \times w)$$

Rotation matrices are used to quantitatively describe the relationship between reference frames in transformations and displacements.

○ Rotation Matrix

We assume that we have mutually orthogonal unit vectors attached to every rigid body.
We can write the mutually orthogonal unit vectors in one frame as a linear combination of the mutually orthogonal unit vectors in the other frame. From Figure 2.1, we have

$$b_1 = R_{11}a_1 + R_{12}a_2 + R_{13}a_3$$
$$b_2 = R_{21}a_1 + R_{22}a_2 + R_{23}a_3$$
$$b_3 = R_{31}a_1 + R_{32}a_2 + R_{33}a_3$$

This collection of these nine coefficients $(R_{11}, R_{12}, .., R_{33})$ can be gathered into a matrix R, and we call this matrix a rotation matrix. Rotation matrices belongs to the special orthogonal group $(SO(3))$ in 3 dimensions which is definced as

$$SO(3) = \{R \in \mathbb{R}^{3\times3} \mid R^TR = RR^T = I \mid det(R) = +1\}$$

A general notation of Rotation matrix is given by ${}^AR_B$, where B is the origin frame and A is the destination frame.
From Figure 2.1, $Q' = {}^AR_BQ_A$. The vector Q'(denoted by dotted line) is the vector Q_B moved from the origin of reference frame B to reference frame A.
Rotation matrices are not intuitively very obvious in describing the extent of rotation so we use co-ordinates for rotation matrices in the form of Euler angles.

○ Euler Angles

The Euler angles are three angles used to describe the orientation of a rigid body with respect to a fixed co-ordinate system. Any rotation in 3D space is composed of three successive rotations around 3 independent axes and 3 characteristic euler angles.

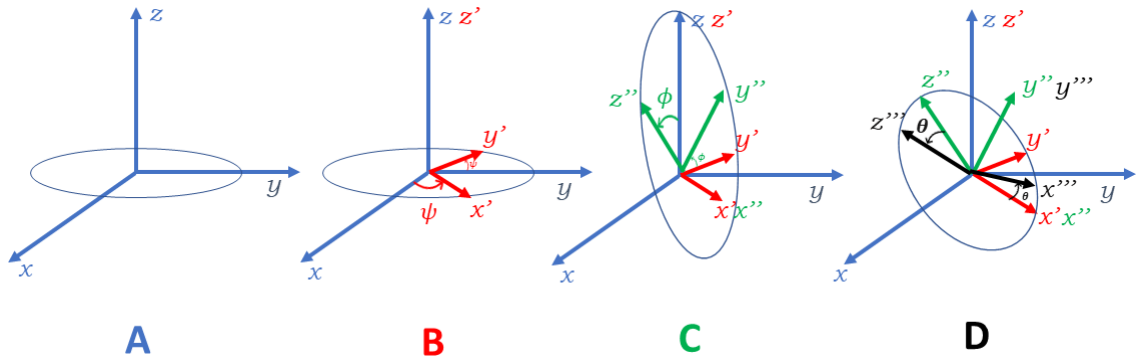

Figure 2.4: Rotations and Euler Angles

We can describe the rotation from reference frame A to reference frame D in Figure 2.4 as rotations from A to B,B to C,C to D. The same rotataion can be described using euler angles as rotation about three independent axes x, y, z with angles ϕ, θ, ψ respectively. In a UAV terminology, the euler angles are called **roll, pitch** and **yaw** angles which are rotations about world x, y, z axis respectively.

$$^{A}R_{D} = {}^{A}R_{B} \times {}^{B}R_{C} \times {}^{C}R_{D} \tag{2.1}$$

$$^{A}R_{D} = rot(z,\psi) \times rot(x,\phi) \times rot(y,\theta) \tag{2.2}$$

$$rot(z,\psi) = \begin{bmatrix} \mathbf{c}\psi & -\mathbf{s}\psi & 0 \\ \mathbf{s}\psi & \mathbf{c}\psi & 0 \\ 0 & 0 & 1 \end{bmatrix},\ rot(x,\phi) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \mathbf{c}\phi & -\mathbf{s}\phi \\ 0 & \mathbf{s}\phi & \mathbf{c}\phi \end{bmatrix},\ rot(y,\theta) = \begin{bmatrix} \mathbf{c}\phi & 0 & \mathbf{s}\phi \\ 0 & 1 & 0 \\ -\mathbf{s}\phi & 0 & \mathbf{c}\phi \end{bmatrix}$$

Here **c** and **s** denotes cos and sine respectively.

◯ Skew-Symmetric Matrices

A matrix is skew-symmetric if $A^T = -A$. A 3×3 skew-symmetric matrix has 3 unique entries. For example, let A be a skew-symmetric matrix. Then, A can be written as

$$A = \begin{bmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{bmatrix} \tag{2.3}$$

We can concisely represent a skew-symmetric matrix as a 3×1 vector denoted by a hat operator. So,

$$\hat{A} = \begin{bmatrix} a_1 \\ a_2 \\ a_3 \end{bmatrix} \tag{2.4}$$

The hat operator is also used to denote the cross product between two vectors.

$$A \times B = \hat{A}B = \begin{bmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{bmatrix} \begin{bmatrix} b_1 \\ b_2 \\ b_3 \end{bmatrix} \tag{2.5}$$

◯ Angular Velocity Vectors

We denote rate of change of rotations using angular velocity vectors.

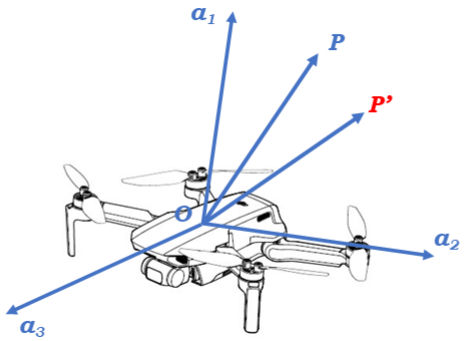

Figure 2.5: Quadrotor with two position vectors

Let $\vec{OP} = p_1a_1 + p_2a_2 + p_3a_3$ and $\vec{OP'} = q_1a_1 + q_2a_2 + q_3a_3$ where $\vec{OP'}$ is translated from a different reference frame X (see Figure 2.5).

$$\begin{bmatrix} q_1 \\ q_2 \\ q_3 \end{bmatrix} = {}^{A}R_X \begin{bmatrix} p_1 \\ p_2 \\ p_3 \end{bmatrix} \tag{2.6}$$

In a continuous motion,

$$q(t) = R(t)p \tag{2.7}$$

where vector $q(t)$ is the changing coordinates of P, matrix $R(t)$ is the time varying rotation matrix as the rigid body rotates and vector p is the coordinates of P in the body-fixed frame A. Note that p is in body fixed frame while q is in inertial frame.
In order to get the velocity of this motion, we differentiate equation (2.7). So, we have

$$\dot{q}(t) = \dot{R}(t)p \tag{2.8}$$

Here $\dot{q}(t)$ is the velocity in inertial frame and $\dot{R}(t)$ is the time derivative of the rotation matrix. There are two representations of the velocity vector, $\dot{q}(t)$, which represents velocity in inertial frame and $R^T(t)\dot{q}(t)$, which represents velocity in body-fixed frame where,

$$\dot{q} = \underbrace{\dot{R}(t)R^T(t)q}_{\hat{\omega}_s} \tag{2.9}$$

$$R^T(t)\dot{q}(t) = \underbrace{R^T(t)\dot{R}(t)}_{\hat{\omega}_b} p \tag{2.10}$$

Equations (2.9) and (2.10) are obtained by replacing p in equation (2.8) with R^Tq and premultiplying (2.8) with $R^T(t)$. The reasons for these tweaks is that $\hat{\omega}_b$ and $\hat{\omega}_s$ are skew-symmetric matrices and as a result they can be written as vectors. These vectors are called the angular velocity vectors.

Let us consider the rotation about the z-axis, $rot(z,\psi) = R = \begin{bmatrix} \mathbf{c}\psi & -\mathbf{s}\psi & 0 \\ \mathbf{s}\psi & \mathbf{c}\psi & 0 \\ 0 & 0 & 1 \end{bmatrix}$

Here, $R^T(t)\dot{R}(t) = \dot{R}(t)R^T(t) = \underbrace{\begin{bmatrix} 0 & -1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}}_{\omega}\dot{\psi}$. So, $\hat{\omega}_b = \hat{\omega}_s = \underbrace{\begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}}_{\hat{\omega}}\dot{\psi}$

Therefore, $\hat{\omega}$ is the angular velocity vector for $rot(z,\psi)$.

○ Rotor Thrust and Torque

The rotors consists of electric motors and are controlled by electronic speed controllers. The quadrotor generates thrust using these rotors which are fitted with propellers. The vertical component of this thrust which is known as lift balances the weight of the UAV while the horizontal component counteracts the drag and helps in translational motion. The parameters that affect thrust are [7],[13] :-

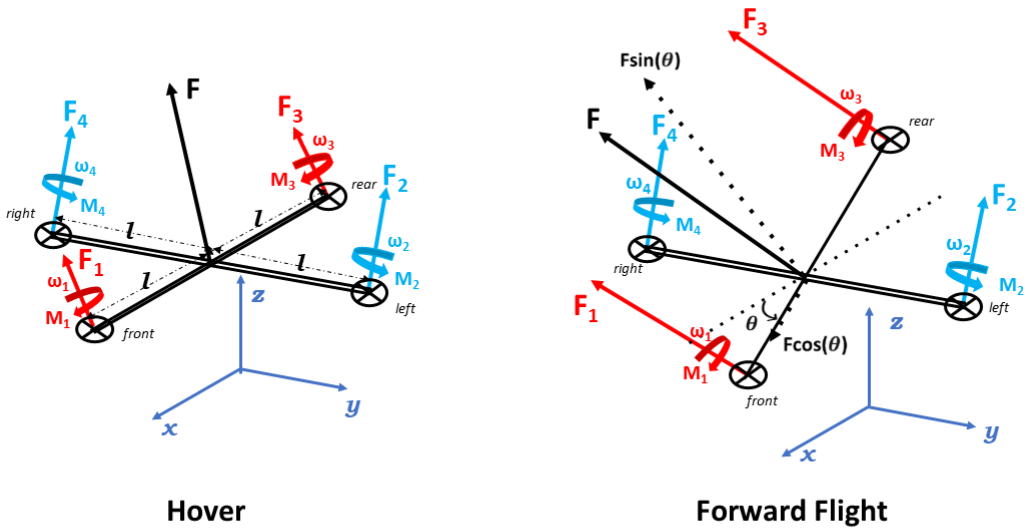

Figure 2.6: Thrust during Hover and Forward flight

Fixed

1. No. of blades in a single propeller
2. Drag coefficient of blade
3. Blade chord length
4. Angle of attack of propeller blade
5. Propeller blade area

Variable

1. Drone Design
   (a) Drone weight (fixed with fixed battery)
   (b) Payload weight
2. Environment
   (a) Wind velocity
   (b) Wind incident angle

(c) Air density

(d) Gravity

3. Quadrotor maneuvers

(a) Climb/Decend (Vertical motion) - All propellers produce same thrust. Climb/sink rate (vertical velocity) decides the required thrust profile.

(b) Hover - All propellers produce same thrust.

(c) Roll - Right side and left side propellers produce unequal thrust producing a rolling torque. Roll angle decides the required thrust in respective propellers.

(d) Pitch - Front and rear propellers produce unequal thrust producig a pitching torque. Pitch angle decides the required thrust in respective propellers.

(e) Yaw - Clockwise and counter-clockwise spinning propellers produce unequal torques resulting a yaw torque. Yaw angle decides the required thrust in respective propellers.

(f) Cruise (Horizontal motion) - Combination of pitch and hover(see Figure 2.6). The thrusts are unequal due to pitch. Due to pitch angle, the vertical component of thrust ($Fsin\theta$) is higher than that required in hover. Also, horizontal velocity depends on the pitch angle(θ).

In 3(c), 3(d) and 3(e) the total thrust produced by the propellers is constant and is equal to the gross weight of the UAV. In 3(f), the vertical component of thrust is equal to the gross weight of the UAV.

Ideally,

$$\text{Thrust } F = k_F\omega^2 \tag{2.11}$$

where, ω is the angular velocity in number of revolutions per minute(RPM) of the propeller/rotor and k_F depends upon the fixed parameters $1-5$ and 2(c). Generally k_F is considered constant for low level flights.

Pairwise differences in rotor thrusts cause the vehicle to rotate. The rolling torque around the x axis, is generated by the moments

$$\tau_{roll} = d(F_4 - F_2) \tag{2.12}$$

where l is the distance from the rotor axis to the center of mass. We can write this in terms of rotor speeds by substituting eq(2.11)

$$\tau_{roll} = lk_F(\omega_4^2 - \omega_2^2) \tag{2.13}$$

Similarly, the pitching torque, around the y axis is given by

$$\tau_{pitch} = lk_F(\omega_3^2 - \omega_1^2) \tag{2.14}$$

Rotation of each propeller is opposed by aerodynamic drag and produces a moment given by

$$M_i = k_M\omega_i^2 \text{ and } i \in \{1, 2, 3, 4\} \tag{2.15}$$

k_M depends on the same factors as k_F.
This exerts a reaction moment on the airframe which acts to rotate the airframe in the opposite direction to the direction of rotation of the propellers. The reaction torque is

$$\begin{aligned}\tau_{yaw} &= M_1 - M_2 + M_3 - M_4 \\ &= k_M(\omega_1^2 - \omega_2^2 + \omega_3^2 - \omega_4^2)\end{aligned} \tag{2.16}$$

So, a yaw torque is generated simply by appropriate coordinated control of all four rotor speeds.
At any given instant, the total torque applied to the airframe according to eq(2.13), eq(2.14) and eq(2.16) is $\tau = (\tau_{roll}, \tau_{pitch}, \tau_{yaw})^T$.
So the rotor thrusts and torques are related to the rotor speed as,

$$\begin{bmatrix} F \\ \tau \end{bmatrix} = \begin{bmatrix} -k_F & -k_F & -k_F & -k_F \\ 0 & -lk_F & 0 & lk_F \\ lk_F & 0 & -lk_F & 0 \\ k_M & -k_M & k_M & -k_M \end{bmatrix} \begin{bmatrix} \omega_1^2 \\ \omega_2^2 \\ \omega_3^2 \\ \omega_4^2 \end{bmatrix} = \mathbb{A} \begin{bmatrix} \omega_1^2 \\ \omega_2^2 \\ \omega_3^2 \\ \omega_4^2 \end{bmatrix} \tag{2.17}$$

The matrix $\mathbb{A}$ is constant and full rank if $k_F, k_M, l > 0$ and we can obtain the required rotor speeds as

$$\begin{bmatrix} \omega_1^2 \\ \omega_2^2 \\ \omega_3^2 \\ \omega_4^2 \end{bmatrix} = \mathbb{A}^{-1} \begin{bmatrix} T \\ \tau_x \\ \tau_y \\ \tau_z \end{bmatrix} \tag{2.18}$$

○ Newton Euler Equations of Motion

In classical mechanics, the Newton–Euler equations describe the combined translational and rotational dynamics of a rigid body. Euler's laws of motion are equations of motion which extend Newton's laws of motion for point particle to rigid body motion. Using Netwon-Euler equations of motion,
For linear motion, the net force(F) on a rigid body is the rate of change of linear momentum(L) in an inertial frame(A)

$$F = \frac{^A dL}{dt} \tag{2.19}$$

For rotational motion, the rate of change of angular momentum (H) of a rigid body B relative to C(the center of mass) in reference frame A is equal to the resultant moment(M) of all external forces acting on the body relative to C.

$$\frac{^A d\, ^A H_C^B}{dt} = M_C^B \tag{2.20}$$

The angular momentum which is a 3×1 vector is given by $^A H_C^B = \mathbb{I}_C . ^A\Omega^B$, where $\mathbb{I}_C$ is the inertia tensor (a 3×3 matrix) and $^A\Omega^B$, the angular velocity (a 3×1 vector). Note that $\mathbb{I}_C$ is measured about the center of mass C and $^A\Omega^B$ is measured in the inertial frame A.

○ Principal Axes and Principal Moments

Principal axis of inertia is that direction in which the angular momentum vector is parallel to the angular velocity vector. The angular momentum along the principal axes are called the principal moments. For a general three-dimensional body, it is always possible to find 3 mutually orthogonal axis for which the products of inertia are zero, and the inertia tensor takes a diagonal form.

In the next section, we describe the equations governing the dynamics of a quadrotor.

2.4 Quadrotor Dynamics

In this section, we derive the equations governing the dynamics of a quadrotor motion in 3-dimensional space. A quadrotor in plus configuration is illustrated in Figure 2.7 which is used to demonstrate the proceedings of this section.

- The quadrotor has a mass m kg.

  We have two reference frames or coordinate systems, one attached to the moving quadrotor (reference frame B with center at C which is also the center of mass of the body) and the other, the inertial reference frame of reference (A) with center at O. $\{b_1, b_2, b_3\}$ constitute the set of unit vectors that describe the body fixed coordinate system, and likewise $\{e_1, e_2, e_3\}$ constitute a coordinate system that's fixed to the inertial frame. Here $\{e_1, e_2, e_3\}$ are aligned with the world x, y, z axes respectively.

  The body frame, B, is attached to the center of mass of the quadrotor with b_1 coinciding with the preferred forward direction and b_3 perpendicular to the plane of the rotors pointing vertically up during perfect hover.
  r_d is the position vector of the center of mass from the inertial reference frame.
  Rotor 1 is a distance l_1 m away along b_1, Rotor 2 is at a distance l_2 m away along b_2, while Rotor 3 and Rotor 4 are similarly l_3 m and l_4 m away along the negative b_1 and b_2 respectively. Here $l_1 = l_2 = l_3 = l_4 = l$.

  The principal axes of the quadrotor coincide with the body reference frame vectors. The moment of inertia along b_1, b_2, b_3 are I_{11} kg.m.m, I_{22} kg.m.m, I_{33} kg.m.m respectively.
  The acceleration due to gravity is g m/s^2.

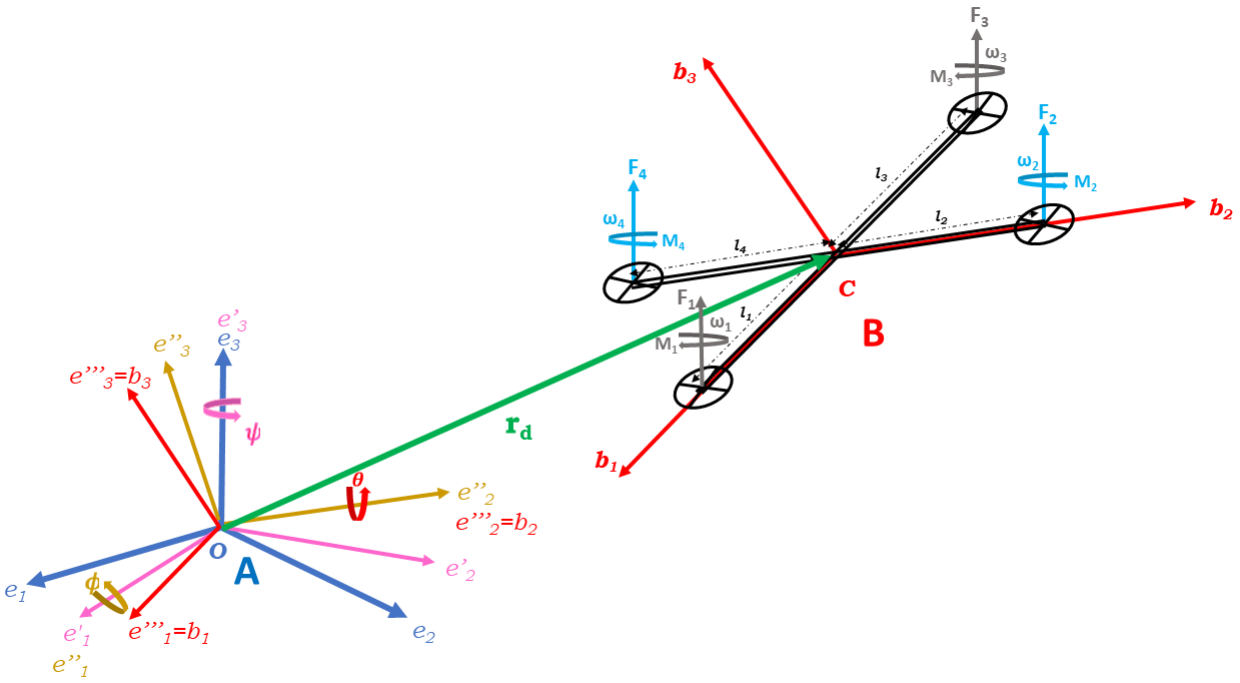

Figure 2.7: Quadrotor Dynamics

- Euler angles are used to describe the rotation of the quadrotor body with reference to the inertial

frame. Here we use ZXY convention that is at first we have rotation about the $z-axis$ with angle ψ which results in an intermediate reference frame $\{e'_1, e'_2, e'_3\}$ followed by rotation about the $x-axis$ with angle ϕ which results in an intermediate reference frame $\{e''_1, e''_2, e''_3\}$ and then rotation about the $y-axis$ with angle θ which results in the reference frame $\{e'''_1, e'''_2, e'''_3\}$ which is also the reference frame $\{b_1, b_2, b_3\}$.
So, the rotation matrix is given by ${}^A R_B = rot(z,\psi) \times rot(x,\phi) \times rot(y,\theta)$.
Here, ϕ, θ, ψ refers to the roll, pitch and yaw angles respectively. The rotation matrix is given by

$$ {}^A R_B = \begin{bmatrix} c\psi c\theta - s\phi s\psi s\theta & -c\phi s\psi & c\psi s\theta + c\theta s\phi s\psi \\ c\theta s\psi + c\psi s\phi s\theta & c\phi c\psi & s\psi s\theta - c\psi c\theta s\phi \\ -c\phi s\theta & s\phi & c\phi c\theta \end{bmatrix} \tag{2.21} $$

where c represents cos and s represents sine.

- The rotors which generate thrust for the quadrotor are independently actuated. The thrust and reaction moment produced by each rotor is given by $F_i = k_F \omega_i^2$ and $M_i = k_M \omega_i^2$, where $k_F(N/rpm^2)$ and $k_M(Nm/rpm^2)$ are rotor constants and $\omega_i(rpm)$ is the angular velocity of each rotor. Here $i \in \{1,2,3,4\}$
The net force acting on the quadrotor frame is given by

$$ F = F_1 + F_2 + F_3 + F_4 - mge_3 \tag{2.22} $$

The net forces act on the body of the quadrotor. To derive the equations of motion, we translate the forces to the inertial frame of reference and for that we use the Roation matrix. Note that in the body reference frame the forces acting are given by the vector $\begin{bmatrix} 0 & 0 & (F_1+F_2+F_3+F_4) \end{bmatrix}^T$. Using Newton's second law of motion the net acceleration, which is the second derivative of r_d in this case can be written by

$$ \ddot{r_d} = \frac{1}{m}\left(\begin{bmatrix} 0 \\ 0 \\ -mg \end{bmatrix} + {}^A R_B \begin{bmatrix} 0 \\ 0 \\ (F_1+F_2+F_3+F_4) \end{bmatrix} \right) \tag{2.23} $$

$r_d = \begin{bmatrix} r_1 \\ r_2 \\ r_3 \end{bmatrix}$ is a position vector where r_1, r_2, r_3 are the co-ordinates in reference frame A.
In addition to forces, each rotor produce a moment perpendicular to the plane of rotation of the blade. Rotors 1 and 3 rotate in the $-b_3$ direction while 2 and 4 rotate in the $+b_3$ direction. Since the moment produced in the quadrotor is opposite to the direction of rotation of the blades, M_1 and M_3 act in the $+b_3$ direction while M_2 and M_4 act in the $-b_3$ direction.
Since l is the distance from the axis of rotation of the rotors to the center of mass of the quadrotor, the net moment acting on the quadrotor frame is given by

$$ M = l \times F_1 + l \times F_2 + l \times F_3 + l \times F_4 + M_1 + M_2 + M_3 + M_4 \tag{2.24} $$

Equation (2.24) denotes a vector sum.

- Since unit vectors of the body reference frames $\{b_1, b_2, b_3\}$ be along the principal axes, we define

$$ {}^A\Omega^B = \Omega_1 b_1 + \Omega_2 b_2 + \Omega_3 b_3 \tag{2.25} $$

where ${}^A\Omega^B$ is the angular velocity of rotation of the quadrotor frame.

Using Euler's equations we have

$$\frac{^{A}d\, H_C}{dt} = M_C \tag{2.26}$$

Here, H_C is the angular momentum of the quadrotor frame and M_C is the net moment acting due to all external forces and torques which is given by equation 2.24.
Equation 2.27 can be written with a correction factor $^{A}\Omega^{B} \times H_C$ that helps us do the differentiation in the body frame.

$$\frac{^{B}d\, H_C}{dt} + {}^{A}\Omega^{B} \times H_C = M_C \tag{2.27}$$

Writing H_C in terms of inertia and angular velocity we have,

$$\frac{^{B}d\, H_C}{dt} = I_{11}\dot{\Omega}_1 b_1 + I_{22}\dot{\Omega}_2 b_2 + I_{33}\dot{\Omega}_3 b_3 \tag{2.28}$$

where I_{11}, I_{22}, I_{33} are the inertia terms along the principal axes.
Using equation 2.28, we can write equation 2.27 as

$$\begin{bmatrix} I_{11} & 0 & 0 \\ 0 & I_{22} & 0 \\ 0 & 0 & I_{33} \end{bmatrix} \begin{bmatrix} \dot{\Omega}_1 \\ \dot{\Omega}_2 \\ \dot{\Omega}_3 \end{bmatrix} + \begin{bmatrix} 0 & -\Omega_3 & \Omega_2 \\ \Omega_3 & 0 & -\Omega_1 \\ -\Omega_2 & \Omega_1 & 0 \end{bmatrix} \begin{bmatrix} I_{11} & 0 & 0 \\ 0 & I_{22} & 0 \\ 0 & 0 & I_{33} \end{bmatrix} \begin{bmatrix} \Omega_1 \\ \Omega_2 \\ \Omega_3 \end{bmatrix} = \begin{bmatrix} M_{C,1} \\ M_{C,2} \\ M_{C,3} \end{bmatrix} \tag{2.29}$$

The components $\Omega_1, \Omega_2, \Omega_3$ are denoted by p, q, r respectively for subsequent references.
So, $^{A}\Omega^{B} = pb_1 + qb_2 + rb_3$.
They are related to the rate of change of roll, pitch and yaw angles as

$$\begin{bmatrix} p \\ q \\ r \end{bmatrix} = \begin{bmatrix} cos(\theta) & 0 & -cos(\phi)sin(\theta) \\ 0 & 1 & sin(\phi) \\ sin(\theta) & 0 & cos(\phi)cos(\theta) \end{bmatrix} \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} \tag{2.30}$$

The moments acting on the body are given by

$$M_{C,1} = l(F_2 - F_4) \tag{2.31}$$
$$M_{C,2} = l(F_3 - F_1) \tag{2.32}$$
$$M_{C,3} = M_1 - M_2 + M_3 - M_4 \tag{2.33}$$

Equation 2.29 can be now written as

$$\mathbb{I} \begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} = \begin{bmatrix} l(F_2 - F_4) \\ l(F_3 - F_1) \\ M_1 - M_2 + M_3 - M_4 \end{bmatrix} - \begin{bmatrix} p \\ q \\ r \end{bmatrix} \times \mathbb{I} \begin{bmatrix} p \\ q \\ r \end{bmatrix} \tag{2.34}$$

where $\mathbb{I} = \begin{bmatrix} I_{11} & 0 & 0 \\ 0 & I_{22} & 0 \\ 0 & 0 & I_{33} \end{bmatrix}$ is the inertia tensor.

We can rewrite (2.34) as

$$\mathbb{I}\begin{bmatrix}\dot{p}\\ \dot{q}\\ \dot{r}\end{bmatrix} = \begin{bmatrix}0 & l & 0 & -l\\ -l & 0 & l & 0\\ \gamma & -\gamma & \gamma & -\gamma\end{bmatrix}\begin{bmatrix}F_1\\ F_2\\ F_3\\ F_4\end{bmatrix} - \begin{bmatrix}p\\ q\\ r\end{bmatrix}\times\mathbb{I}\begin{bmatrix}p\\ q\\ r\end{bmatrix} \tag{2.35}$$

where $\gamma = \frac{k_M}{k_F}$ is the relationship between lift and drag given by Equations (2.11),(2.15).

From equation (2.23) and (2.35), the governing equations of the quadrotor dynamics are given by

$$m\begin{bmatrix}\ddot{r}_1\\ \ddot{r}_2\\ \ddot{r}_3\end{bmatrix} = \begin{bmatrix}0\\ 0\\ -mg\end{bmatrix} + {}^{A}R_{B}\begin{bmatrix}0\\ 0\\ \underbrace{(F_1+F_2+F_3+F_4)}_{u_1}\end{bmatrix} \tag{2.36}$$

$$\mathbb{I}\begin{bmatrix}\dot{p}\\ \dot{q}\\ \dot{r}\end{bmatrix} = \underbrace{\begin{bmatrix}0 & l & 0 & -l\\ -l & 0 & l & 0\\ \gamma & -\gamma & \gamma & -\gamma\end{bmatrix}\begin{bmatrix}F_1\\ F_2\\ F_3\\ F_4\end{bmatrix}}_{u_2} - \begin{bmatrix}p\\ q\\ r\end{bmatrix}\times\mathbb{I}\begin{bmatrix}p\\ q\\ r\end{bmatrix} \tag{2.37}$$

Note that equation (2.36) has components in the inertial frame while equation (2.37) has componenets in the body frame. The quadrotor motion in 3 dimensional space is achievevd by manipulating u_1(scalar) and $u_2(3\times 1$ vector) using controllers and is discussed in the next section. With u_1, the quadrotor stays afloat by balancing its weight and with u_2 the quadrotor performs the rotational maneuvers.

In the following section we introduce a PD controller to drive it's error dynamics.

2.5 Quadrotor Control

We want to make the quadrotor follow a specified trajectory. In this section we discuss the proceedings of how we can control the quadrotor and it's dynamics so that it can follow a predefined trajcetory.
First we show how trajectories are defined in 3 dimensional space

○ Trajectory Tracking in 3 dimensions

Trajectories are defined by time parameterized polynomials. Let us consider a trajectory whose position, velocity and acceleration is given by $r^{des}(t), \dot{r}^{des}(t), \ddot{r}^{des}(t)$. We also define the heading of the quadrotor in terms of time functions of yaw angle given by by $\psi^{des}(t), \dot{\psi}^{des}(t), \ddot{\psi}^{des}(t)$.

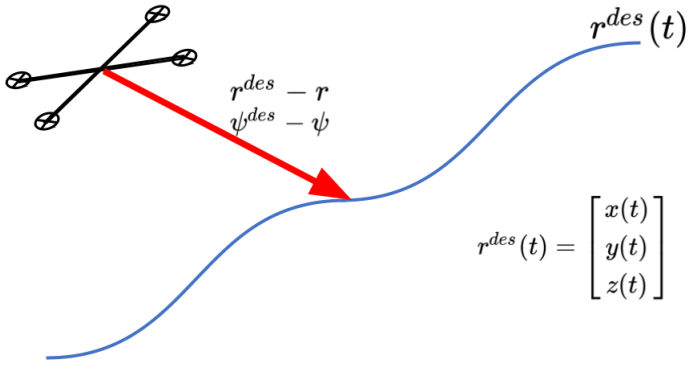

Figure 2.8: Quadrotor following a trajectory

○ Error dynamics

In our pursuit to drive the quadrotor follow a particular trajectory, we define a desired trajectory. Now we command the rotors in a way that the quadrotor follows that particular trajectory. In this regard we define the error dynamics of the quadrotor motion. Error is defined as the difference between the desired and present position and the error dynamics is given by

$$e_p = r^{des}(t) - r(t) \tag{2.38}$$

$$e_v = \dot{r}^{des}(t) - \dot{r}(t) \tag{2.39}$$

where e_p and e_v are the errors in position and velocity. We want

$$\ddot{r}^{des}(t) - \ddot{r}^{c}(t) + K_d e_v + K_p e_p = 0 \tag{2.40}$$

Here $\ddot{r}^{c}(t)$ is the commanded acceleration so that the errors e_p and e_v exponentially decay to 0. Here, constants K_p and K_d defines the rate at which the error decays exponentially.

○ Control Architecture

Quadrotors are underactuated machines. We have 6 degrees of freedom, 3 rotaional (roll, pitch and yaw) and 3 translational (forward/backward, sideward and upward/downward) but only 4 independent controls in the form of it's 4 rotors. So, to control a quadrotor, we employ a nested control structure involving two controller blocks. One block defines the attitude or orientation of the quadrotor and the other block controls it's position. To have a translational motion in 3 dimensional space, the quadrotor first takes an attitude or leans with a roll, pitch and yaw angle and then then moves ahead.

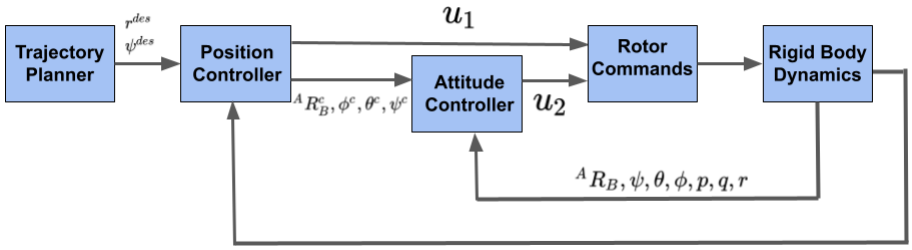

Figure 2.9: Quadrotor Control Architecture

So, for a given trajectory having position velocity and acceleration, the inner loop attitude controller takes command from the outer loop position controller in the form of desired roll, pitch and yaw angle $(\phi^c, \theta^c, \psi^c)$

which in turn calculates the desired rotation matrix ${}^{A}R_{B}$. The desired roll, pitch and yaw angle is dependent on the desired velocity and acceleration. Depending upon the current values of roll, pitch and yaw angles along with thier rates, the attitude controller spits out u_2 which drives the rotors and sets course or attitude for the quadotor motion. The outer loop position controller takes command from the desired trajectory. The primary task of the position controller is to balance the weight of the quadrotor by providing command u_1 to the rotors. It also re-estimates the command for the inner loop attitude controller after taking feedback of the present position of the quadtoror.

○ The Nominal State

Our controllers are derived by linearizing the equations of motion and motors models (2.36),(2.37),(2.11),(2.15) at an operating point that corresponds to the nominal hover state, $\boldsymbol{r^{des}(t)} = \boldsymbol{r} = \boldsymbol{r}_0, \theta = \phi = 0, \psi = \psi_0, \dot{\boldsymbol{r}} = 0$, and $\dot{\phi} = \dot{\theta} = \dot{\psi} = 0$, where the roll and pitch angles are small ($cos\phi \approx 1, cos\theta \approx 1, sin\phi \approx \phi$, and $sin\theta \approx \theta$). At this state, the lift from the propellers is given by:

$$F_{i,0} = \frac{mg}{4} \tag{2.41}$$

The nominal values for the inputs at hover are $u_{1,0} = mg, \boldsymbol{u}_{2,0} = \mathbf{0}$.
Linearizing (2.36), we get:

$$\ddot{r}_1 = g\left(\Delta\theta \cos\psi_0 + \Delta\phi \sin\psi_0\right) \tag{2.42}$$
$$\ddot{r}_2 = g\left(\Delta\theta \sin\psi_0 - \Delta\phi \cos\psi_0\right) \tag{2.43}$$
$$\ddot{r}_3 = \frac{1}{m}u_1 - g \tag{2.44}$$

Linearizing (2.36), we get:

$$\begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} = \mathbb{I}^{-1} \begin{bmatrix} 0 & l & 0 & -l \\ -l & 0 & l & 0 \\ \gamma & -\gamma & \gamma & -\gamma \end{bmatrix} \begin{bmatrix} F_1 \\ F_2 \\ F_3 \\ F_4 \end{bmatrix} \tag{2.45}$$

From equation (2.30), we have $p = \dot{\phi}$ and $q = \dot{\theta}$.
In other words, the equations of motion are decoupled in terms of angular accelerations. Each component of angular acceleration depends only on the appropriate component of $\boldsymbol{u}_2$.

○ Position and Attitude Control

The control problem is to determine the four inputs, $\{u_1, \boldsymbol{u}_2\}$ required to hover or to follow a desired trajectory. As shown in Figure 2.9, we will use errors in the robot's position (equation (2.40)) to drive a position controller from (2.44) which directly determines u_1. The model in equations (2.42),(2.43),(2.44) also allows us to derive a desired orientation. The attitude controller for this orientation is derived from the model in equation (2.45).

· **Attitude Control**

We now present a a proportional plus derivative (PD) attitude controller to track a trajectory in $SO(3)$ specified in terms of a desired roll, pitch and yaw angle. Since our development of the controller will be

based on linearized equations of motion, the attitude must be close to the nominal hover state where the roll and pitch angles are small.
Near the nominal hover state the proportional plus derivative control laws take the form:

$$\boldsymbol{u}_2 = \begin{bmatrix} k_{p,\phi}\left(\phi^c - \phi\right) + k_{d,\phi}\left(p^c - p\right) \\ k_{p,\theta}\left(\theta^c - \theta\right) + k_{d,\theta}\left(q^c - q\right) \\ k_{p,\psi}\left(\psi^c - \psi\right) + k_{d,\psi}\left(r^c - r\right) \end{bmatrix} \tag{2.46}$$

Here the terms with superscript **c** denotes the commanded angles calculated from the position controller. $k_{p,x}$ and $k_{d,x}$ where $x \in \{\phi, \theta, \psi\}$ are the proportional and derivative gains respectively of the PD controller.
· **Position Control**
In the next subsection, we will present two position control methods that use the roll and pitch angles as inputs to drive the position of the quadrotor. In both methods, the position control algorithm will determine the commanded roll and pitch angles, θ^c and ϕ^c, which can be used to compute the desired rotor speeds from equation (2.46).
The first, a hover controller, is used for station-keeping or maintaining the position at a desired position vector, $\boldsymbol{r}_0$. The second tracks a specified trajectory, $\boldsymbol{r}^{des}(t)$, in three dimensions. In both cases, the desired yaw angle, is specified independently. It can either be a constant, ψ_0, or a time varying quantity, $\psi^{des}(t)$. We will assume that the desired trajectory will be provided by a trajectory planner as an input to specify the trajectory of the position vector and the yaw angle we are trying to track.

2.5.1 Hover Controller

For hovering, $\boldsymbol{r}^{des}(t) = \boldsymbol{r}_0$ and $\psi^{des}(t) = \psi_0$. The command accelerations, $\ddot{r}_i^c$, are calculated from a proportional plus derivative (PD) controller. We define the position error in terms of components using the standard reference frame $\boldsymbol{e}_1, \boldsymbol{e}_2, \boldsymbol{e}_3$ by:

$$e_i = \left(r_i^{des} - r_i\right) \tag{2.47}$$

$i \in \{1, 2, 3\}$
In order to guarantee that this error goes exponentially to zero, we require

$$\left(\ddot{r}_i^{des} - \ddot{r}_i^c\right) + k_{d,i}\left(\dot{r}_i^{des} - \dot{r}_i\right) + k_{p,i}\left(r_i^{des} - r_i\right) = 0 \tag{2.48}$$

where $\ddot{r}_i^{des} = \dot{r}_i^{des} = 0$ for hover. $k_{d,i}$ and $k_{p,i}$ where $x \in \{1, 2, 3\}$ are the derivative and proportional gains of the PD controller respectively
From equations (2.42),(2.43),(2.44), we can obtain the relationship between the desired accelerations and roll and pitch angles. Given that $\delta\theta = \theta - \theta_0 = \theta$ and $\delta\psi = \psi - \psi_0 = \psi$, we can write

$$\ddot{r}_1^c = g\left(\theta^c \cos\psi_0 + \phi^c \sin\psi_0\right) \tag{2.49}$$
$$\ddot{r}_2^c = g\left(\theta^c \sin\psi_0 - \phi^c \cos\psi_0\right) \tag{2.50}$$
$$\ddot{r}_3^c = \frac{1}{m}u_1 - g. \tag{2.51}$$

For hover, equation (2.51) yeilds:

$$u_1 = mg + m\ddot{r}_3^c = mg - m\left(k_{d,3}\dot{r}_3 + k_{p,3}\left(r_3 - r_{3,0}\right)\right) \tag{2.52}$$

Equations (2.50),(2.51) can be used to compute the desired roll and pitch angles for the attitude controller:

$$\phi^c = \frac{1}{g}\left(\ddot{r}_1^{\,c}\sin\psi_0 - \ddot{r}_2^{\,c}\cos\psi_0\right) \tag{2.53}$$

$$\theta^c = \frac{1}{g}\left(\ddot{r}_1^{\,c}\cos\psi_0 + \ddot{r}_2^{\,c}\sin\psi_0\right) \tag{2.54}$$

The desired roll and pitch velocities are taken to be zero.

$$p_{\text{des}} = 0$$
$$q_{\text{des}} = 0$$

Equations (2.53),(2.54) provide the setpoints for the attitude controller in (2.46). So,

$$\boldsymbol{u}_2 = \begin{bmatrix} k_{p,\phi}\left(\phi^c - \phi\right) + k_{d,\phi}\left(p^c - p\right) \\ k_{q,\theta}\left(\theta^c - \theta\right) + k_{d,\theta}\left(q^c - q\right) \\ k_{r,\phi}\left(\psi^c - \psi\right) + k_{d,\psi}\left(r^c - r\right) \end{bmatrix} \tag{2.55}$$

So with u_1 and u_2, we can drive the quadrotor attain the desired trajectory.

2.5.2 3D Trajectory Control

The 3-D Trajectory Controller is used to follow three-dimensional trajectories with modest accelerations so the near-hover assumptions hold. To derive this controller we follow the same steps as in hover except that $\dot{r}_i^{des}$ and $\ddot{r}_i^{des}$ are no longer zero in (2.48) but are obtained from the specification of the trajectory. If the near-hover assumption holds and the dynamics are linear with no saturation on the inputs, a controller that generates the acceleration $\ddot{r}^c f_i$ according to equatioin (2.48) is guaranteed to drive the error exponentially to zero. However, it is possible that the commanded trajectory has twists and turns that are too hard to follow perfectly because of errors in the model or limitations on the input thrusts.

Having discussed the dynamics and control of a quadrotor, we step into the proceedings of our power calculation formulation. The idea is to estimate how the thrust and torque demands required to fulfil a particular trajectory vary with time. We have seen that with every attitude the quadrotor takes, the commanded values of roll, pitch and yaw angle change, and to fulfil those commands, the thrust and torque required changes. The thrust and torque changes result from changes in rotor speed, and rotor speed has direct implications with power [16]. In the next section, we show how trajectories are evaluated in terms of power.

2.6 A power model of a quadrotor based on maneuvers

To the best of our knowledge, almost all of the power models present in literature take in variables like velocities, body weight, payload weight, etc. and spit out power as a static number [13], but we claim that power drawn by the rotors varies with time and are not identical in all of the four rotors at all instants.

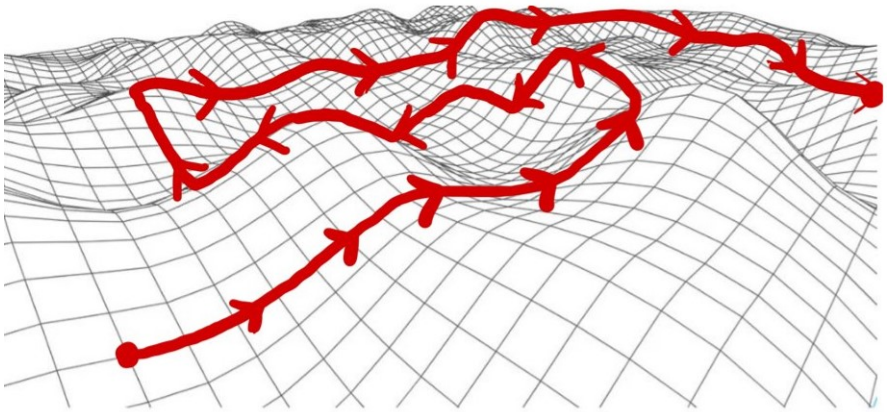

Figure 2.10: Quadrotor moving through a 3D terrrain

A quadrotor takes a sequence and combination of maneuvers (see Fig 2.11) to travel in a predefined trajectory (see Fig 2.10). Each maneuver is performed with a specific set of roll, pitch, and yaw angles, achieved with specific thrust and torque in each rotor. To produce a specific thrust, the rotors need to spin at a designated speed; hence, to execute these maneuvers, the rotor speeds are different in the four rotors. For example, to achieve a left roll maneuver, the right rotors spin faster than the left rotors. So, the power demand is different in all four rotors and hence power becomes a function of time and the trajectory it follows. Motivated by this idea, we have simulated a linearized dynamics of a quadrotor using MATLAB to see how each rotor power varies with time and derived a formulation to track the power in each rotor as the quadrotor moves in a trajectory. The following subsection enumerates our approach.

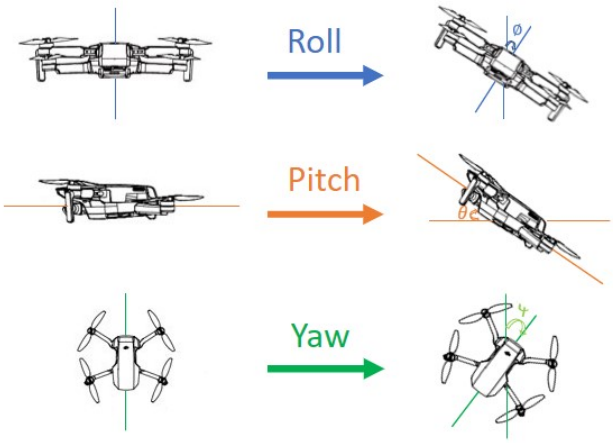

Figure 2.11: Various maneuvers of a quadrotor

2.6.1 Power model design using MATLAB

Section 2.4 summarises the dynamics governing a quadrotor's motion. In Section 2.5, we have shown how to incorporate a Proportional-Derivative (PD) control architecture to drive the error dynamics of the quadrotor motion.

In this formulation, the model takes the quadrotor parameters (mass, arm length, inertia tensor, gravity), predefined trajectory (in the form of a twice differentiable polynomial) and the controller gains as input and outputs the individual rotor powers as a function of time. We achieve it by probing into the controller outputs $u_1(t), u_2(t)$ in order to calculate instantaneous thrust $F_k(t)_{k\in\{1,2,3,4\}}$. Finding the instantaneous thrusts help us in realizing the maneuvers that the quadrotor undertakes. Considering the ideal case where thrust $F \propto \omega^2$, we obtain the corresponding rotor speeds $\omega_k(t)_{k\in\{1,2,3,4\}}$. The goal is to obtain a rotor's current and voltage profile from a given rotor speed data to calculate the power profiles. So, we have evaluated an inverse map from the governing second-order equations of a DC motor (where we consider rotor parameters as R_a, L_a, B, J, K_m) which is used to calculate the current profiles $i_k(t)_{k\in\{1,2,3,4\}}$ from the given rotor speed profiles $\omega_k(t)_{k\in\{1,2,3,4\}}$. Considering voltage supplied to the motors as $v(t)$, we then obtain the instantaneous power profiles as $P_k(t)_{k\in\{1,2,3,4\}} = v.i_K(t)$.
The algorithm designed in MATLAB is outlined using the illustration in Figure 2.12.
Here we define the state vector as $s = \begin{bmatrix} x & y & z & \dot{x} & \dot{y} & \dot{z} & \phi & \theta & \psi & p & q & r \end{bmatrix}$ where x, y, z denotes the quadrotor position in inertial reference frame, ϕ, θ, ψ denotes the roll, pitch and yaw angles and p, q, r is the angular velocities of the quadrotor body.

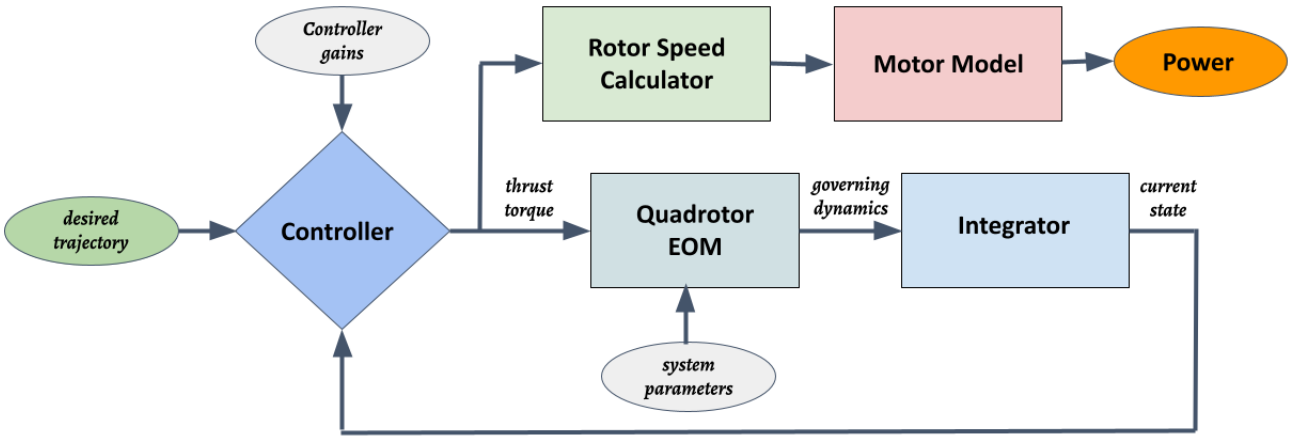

Figure 2.12: Flowchart of Power Model Design

We use seperate scripts for each of the block in Figure 2.12. The scripts used are enumerated below

- **Trajectory Generator** This script outputs the dynamics of the trajectory as time series polynomials for each iteration of the simulation. It takes a time counter as input and outputs the desired state as a vector containing 4 fields. The filds are position, velocity, acceleration, yaw and yaw derivative.

- **Controller** This scripts takes the desired position, velocity, acceleration and yaw at each iteration compares it with the current state which is also an input to this script and outputs decoupled torques and thrusts of each rotors. This script is based on equations (2.52) and (2.53). This script contains the controller gains which help us tune the response of the quadrotor motion. It has 12 tunable gains.

- **Quadrotor EOM** This scripts emulates the dynamics of the quadrotor(equations (2.36), (2.37)) and outputs the governing equations after taking system parameters, torques and thrusts (from the output of the controller script) as inputs.

- **Integrator** This script is basically a solver which integrates the differential equations of the quadrotor dynamics and outputs the state vector. We use $ode45$ as a solver in this case. At each iteration the solver iterates for an interval of $0.05s$ with a step size of $0.01s$.

- **Rotor Speed Calculator** This script calculates the rotor speeds of each rotor from the thrust and torque data derived from the output of the controller script

- **Motor Model** This script derives the current and voltage data that would have been delivered to achieve the rotor speeds obtained in our simulation and then computes the power profile considering a set of motor constants.

Examples showing the power calculation of a quadrotor flight using our formulation is discussed in the next subsection. In this case, we have assumed nominal values of the quadrotor attributes and motors parameters.

2.6.2 Power calculations using the designed model

The power calculations is shown as follows -
We define the following
Quadrotor attributes -

$$m = 0.18kg, L = 0.018m, I = \begin{bmatrix} 0.0025 & 0 & 0 \\ 0 & 0.000232 & 0 \\ 0 & 0 & 0.0003738 \end{bmatrix} kg.m.m$$

Motor Parameters -

$R_a = 1\Omega, L_a = 1H, J = 5Kg.m.m, B = 0.01N.m.s, K = 1.6V/rad/s, K_F = K_M = 1, v(t) = 1V$

Controller Gains -

Attitude Controller - $\begin{bmatrix} k_{p,\phi} \\ k_{p,\theta} \\ k_{p,\psi} \end{bmatrix} = \begin{bmatrix} 100 \\ 100 \\ 100 \end{bmatrix}, \begin{bmatrix} k_{d,\phi} \\ k_{d,\theta} \\ k_{d,\psi} \end{bmatrix} = \begin{bmatrix} 2 \\ 2 \\ 2 \end{bmatrix}$

Position Controller - $\begin{bmatrix} k_{p,x} \\ k_{p,y} \\ k_{p,z} \end{bmatrix} = \begin{bmatrix} 200 \\ 200 \\ 100 \end{bmatrix}, \begin{bmatrix} k_{d,x} \\ k_{d,y} \\ k_{d,z} \end{bmatrix} = \begin{bmatrix} 40 \\ 40 \\ 20 \end{bmatrix}$

Desired Trajectory -

$$\ddot{r}_{des} = \begin{bmatrix} 2 \\ 0 \\ 0 \end{bmatrix} m/s^2, \dot{r}_{des} = \begin{bmatrix} 2t \\ 0 \\ 0 \end{bmatrix} m/s, r_{des} = \begin{bmatrix} t^2 \\ 0 \\ 0 \end{bmatrix} m$$

The simulation uses a runtime of 10s and since each iteration of the solver is of 0.05s, we have 200 iterations. The response of the quadrotor motion is shown in Figure 2.13. The Thrust profile of each rotor is shown in Figure 2.14. The individual rotor speed profiles are shown in Figure 2.15 . To show the difference in rotor speeds of Rotor 1 and Rotor 3, these two rotor speeds are shown in a same plot in Figure 2.16. Finally the rotor power profiles are shown in Figure 2.17 .

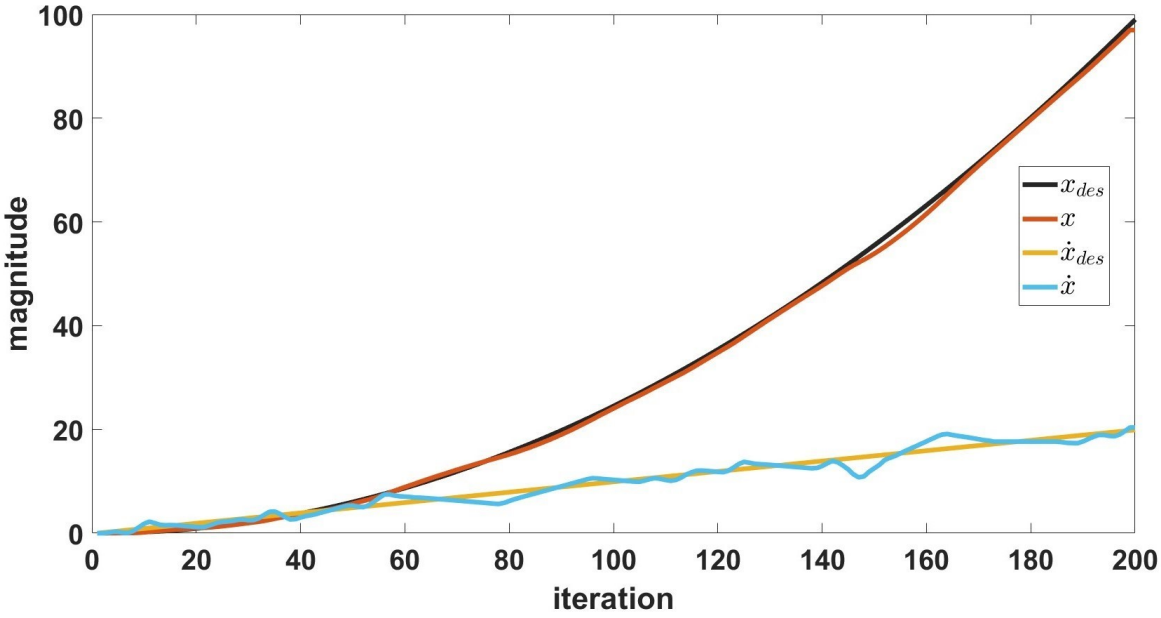

Figure 2.13: Various maneuvers of a quadrotor

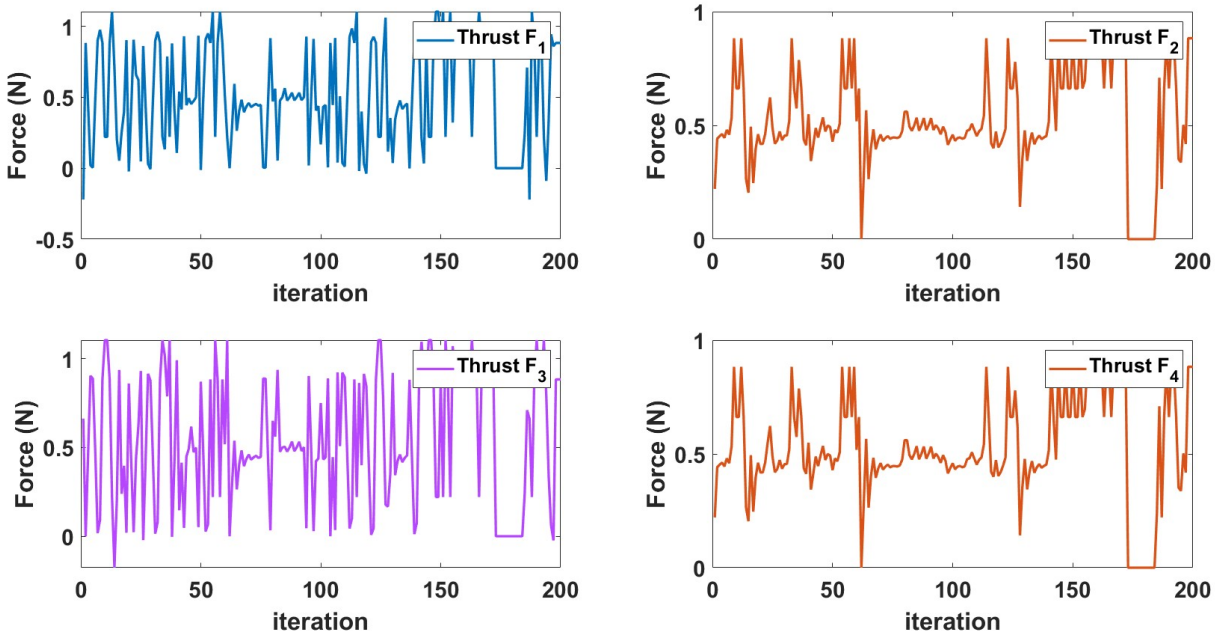

Figure 2.14: Force Profiles

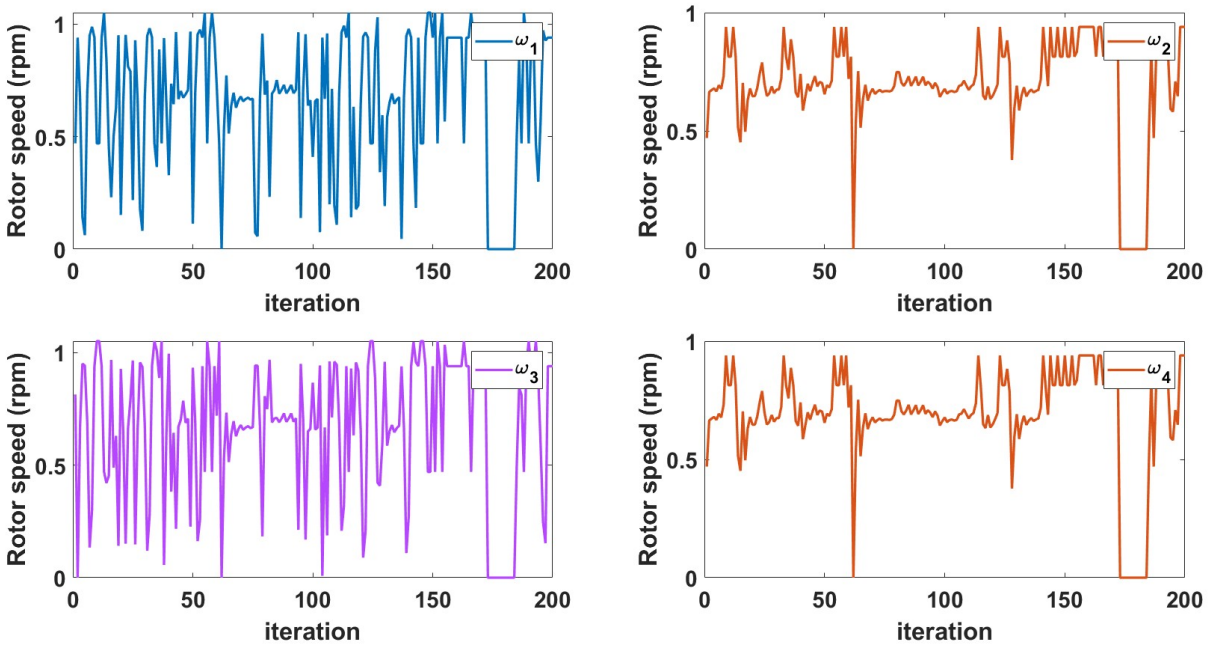

Figure 2.15: Rotor Speed Profiles

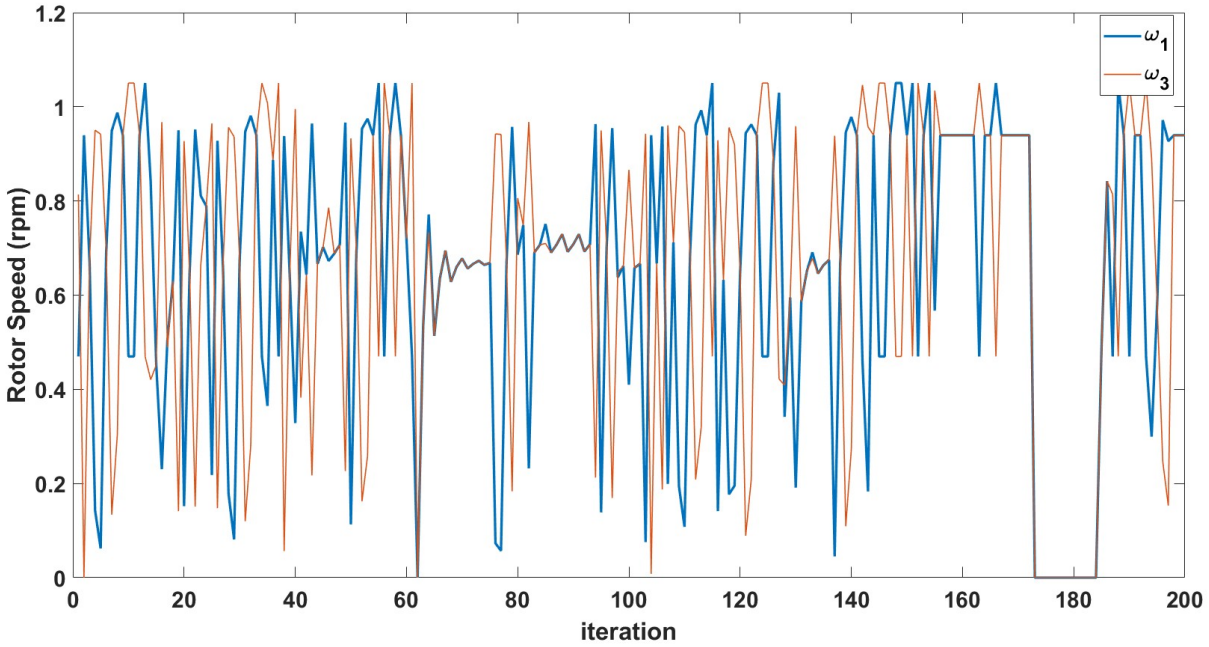

Figure 2.16: Rotor Speed 1 and Rotor Speed 3

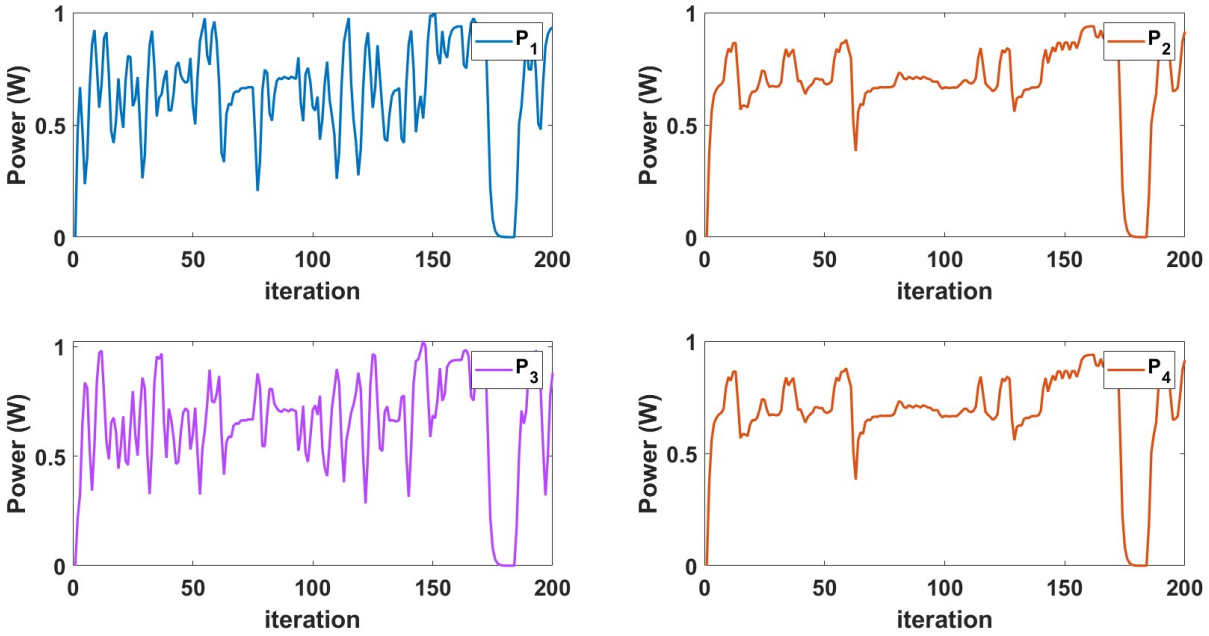

Figure 2.17: Power Profiles

To show how power varies with different maneuvers, we discuss three quadrotor flight simulations to reflect pitch only, roll only and hover maneuver in order to highlight the power implications in each case.
We define the following -

Quadrotor attributes -

$$m = 0.18kg, L = 0.018m, I = \begin{bmatrix} 0.0025 & 0 & 0 \\ 0 & 0.000232 & 0 \\ 0 & 0 & 0.0003738 \end{bmatrix} kg.m.m$$

Motor Parameters -

$R_a = 1\Omega, L_a = 1H, J = 5Kg.m.m, B = 0.01N.m.s, K = 1.6V/rad/s, K_F = K_M = 1, v(t) = 1V$

Controller Gains -

Attitude Controller - $\begin{bmatrix} k_{p,\phi} \\ k_{p,\theta} \\ k_{p,\psi} \end{bmatrix} = \begin{bmatrix} 100 \\ 100 \\ 100 \end{bmatrix}, \begin{bmatrix} k_{d,\phi} \\ k_{d,\theta} \\ k_{d,\psi} \end{bmatrix} = \begin{bmatrix} 2 \\ 2 \\ 2 \end{bmatrix}$

Position Controller - $\begin{bmatrix} k_{p,x} \\ k_{p,y} \\ k_{p,z} \end{bmatrix} = \begin{bmatrix} 200 \\ 200 \\ 100 \end{bmatrix}, \begin{bmatrix} k_{d,x} \\ k_{d,y} \\ k_{d,z} \end{bmatrix} = \begin{bmatrix} 40 \\ 40 \\ 20 \end{bmatrix}$

Desired Trajectory -

We define

$$p = 0.375t^3 - 0.00285t^4 + 0.0001t^5 \tag{2.56}$$

$$v = 0.1125t^2 - 0.0112t^3 + 0.0005t^4 \tag{2.57}$$

$$a = 0.225t - 0.0336t^2 + 0.002t^3 \tag{2.58}$$

Desired trajectory Pitch maneuver -

$$\ddot{r}_{des} = \begin{bmatrix} p \\ 0 \\ 0 \end{bmatrix} m/s^2, \dot{r}_{des} = \begin{bmatrix} v \\ 0 \\ 0 \end{bmatrix} m/s, r_{des} = \begin{bmatrix} a \\ 0 \\ 0 \end{bmatrix} m$$

Desired trajectory Roll maneuver -

$$\ddot{r}_{des} = \begin{bmatrix} 0 \\ p \\ 0 \end{bmatrix} m/s^2, \dot{r}_{des} = \begin{bmatrix} 0 \\ v \\ 0 \end{bmatrix} m/s, r_{des} = \begin{bmatrix} 0 \\ a \\ 0 \end{bmatrix} m$$

Desired trajectory for hover -

$$\ddot{r}_{des} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} m/s^2, \dot{r}_{des} = \begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix} m/s, r_{des} = \begin{bmatrix} 0 \\ 0 \\ 11 \end{bmatrix} m$$

The simulation uses a runtime of 10s and since each iteration of the solver is of 0.05s, we have 200 iterations. For each simulated flight, we get thrust $F_{i\in\{1,2,3,4\}}$, rotor speeds $\omega_{i\in\{1,2,3,4\}}$ and power data $P_{i\in\{1,2,3,4\}}$ as a function of time. To summarize these data, we have calculated the RMS value of each time series data, which has been tabulated in Table 2.1.

Table 2.1: Power estimates for maneuver specific flights

Maneuver	**Thrust (N)**				**Rotor Speed (RPM)**				**Rotor Power(W)**				**Total Power (W)**	**% diff**
	F1	F2	F3	F4	ω_1	ω_2	ω_3	ω_4	P1	P2	P3	P4		
Hover	0.4405	0.4405	0.4405	0.4405	0.6594	0.6594	0.6594	0.6594	0.6551	0.6551	0.6551	0.6551	2.6204	-
Pitch	0.5177	0.5181	0.5571	0.5181	0.6994	0.7151	0.7304	0.7151	0.6846	0.7117	0.7203	0.7117	2.8283	8
Roll	0.5063	0.5720	0.5063	0.5442	0.7048	0.7150	0.7048	0.6984	0.7019	0.6977	0.7019	0.6761	2.7776	6

It is seen that power consumed during pitch and roll is 8% and 6% higher than that during hover. A traditional power model could not have differentiated roll and pitch in terms of power, as the horizontal velocity is the same in both cases. This is also observed from the power data obtained from field experiments (shown in the next chapter). Based on this formulation, we want to outline that in order to estimate the power that the quadrotor will consume when following a predefined trajectory, we must look into the complete power profiles of four rotors. Such a method might lead to accurate power tracking and produce efficient trajectories through path planning.

Chapter 3

Field Experiments

Power consumption models derived in the literature are based on variables like payload weight, velocity etc. However none of these models are based on different instances or maneuvers that quadrotor undergo during a flight [13]. It is unknown if power consumed by a quadrotor cruising with a pitch angle (Figure 3.1) is different from that of a quadrotor cruising with a roll angle (Figure 3.2) in a horizontal flight. The

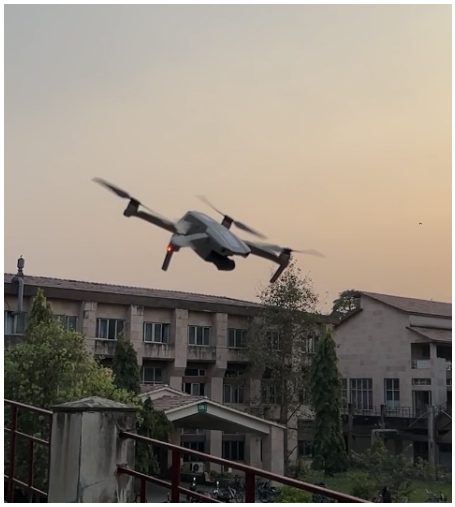

Figure 3.1: Cruise with Pitch

Figure 3.2: Cruise with Roll

prime idea of our research is based on the fact that a quadrotor trajectory is nothing but a combination and sequence of maneuvers and each maneuver is achieved with a specific set of roll, pitch, yaw angles and thrust generated by the rotors. The roll, pitch and yaw angles are achieved as a result of rolling, pitching and yaw torques. If we map each maneuver to a specific set of rolling, pitching, yaw torques and thrust, we can calculate the rotor speeds that will be required to achieve that maneuver. From the rotor speeds, we can calculate the power that will be consumed by each rotor. In this chapter, we study if power consumption computation using maneuvers has an edge over the power consumption models available in literature. In the

Figure 3.3: DJI Mavic Air 2

following sections, we present results of experiments to show how power varies with different maneuvers. We use a DJI Mavic Air 2 drone (see specifications in Table 3.1) to perform different sets of maneuvers. The power profiles (power as a function of time) of the four rotors, vertical velocity (vv), horizontal velocity (hv) and other flight attributes for each characteristic maneuver are extracted from the log files stored in the motherboard of the drone. All the experiments are performed on a time span of 16 seconds. DJI Mavic Air 2 has 3 flight modes viz. *cine*, *normal*, and *sport*. We have performed the experiments in normal mode (moderate speed) and sport mode (high speed) and the average power for each maneuver is tabulated in Table 3.2 and Table 3.3.

Table 3.1: DJI Mavic Air 2 Specifications

Mass	570 gms
Dimension	183x253x77 mm
Max Flight time	34 minutes
Battery Capacity	3500 mAh
Maximum Range	18.5 kms
Max Ascend Velocity	4 m/s
Max Horizontal Velocity	12 m/s (N Mode), 19 m/s (S Mode)

3.1 Experiment 1: Hover maneuver

In this experiment, the quadrotor is made to hover at an altitude of 53m and the power consumed by the four rotors with respect to time is shown in Figure 3.4. It is seen that the front rotors consume a higher power than the back ones. This is possibly due to the asymmetric placements of the front and back arms of the drone. The rear arms are placed low compared to the front arms. (see Figure 3.3)

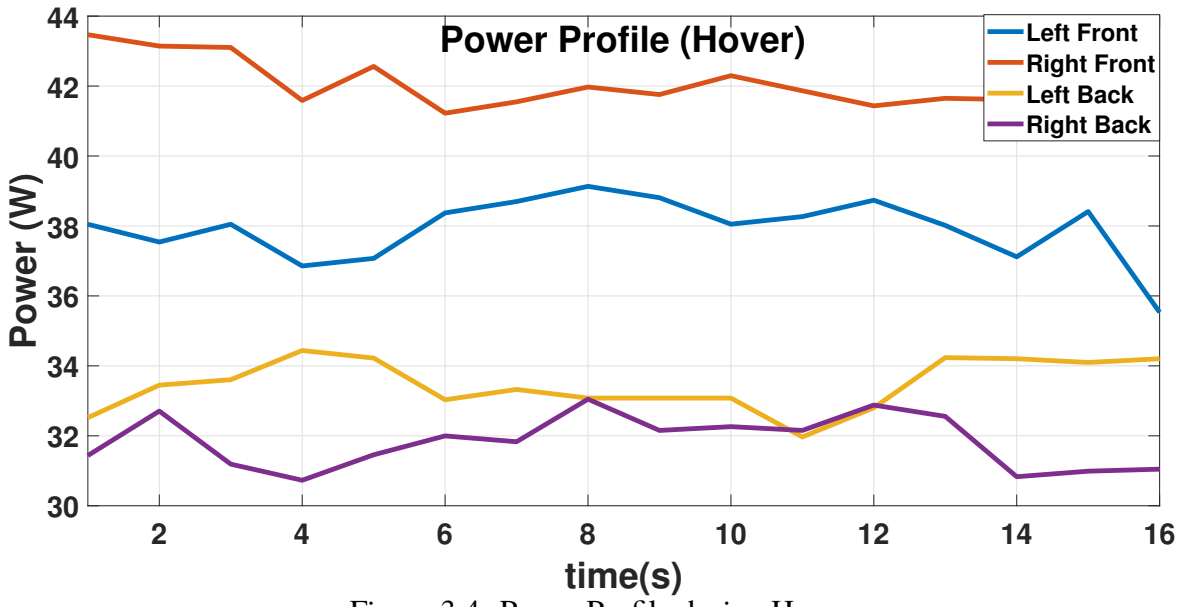

Figure 3.4: Power Profile during Hover

3.2 Experiment 2 - Pitch maneuver

In this experiment, the quadrotor is commanded to pitch Forward(F) and Backward(B) and the power profiles are shown in Figure 3.5 and Figure 3.6 respectively. As expected, the back rotors spin faster during

pitching forward and hence consume more power and similarly the front rotors consume more power during pitching backward. During pitching forward, the horizontal velocity is 11.9 m/s while during pitching backward the horizontal velocity is 11.98 m/s. Although the horizontal velocities are identical in both cases, but there is a difference of 31.4 W in power consumed. A traditional power model can not capture this difference because in both the maneuvers, the significant variable is horizontal velocity which is identical in both cases. A different set of pitch experiments are also performed in sport mode with a horizontal velocities of 16.38 m/s (pitching forward) and 16.82 m/s (pitching backward) and the average power consumed are tabulated in Table 3.3.

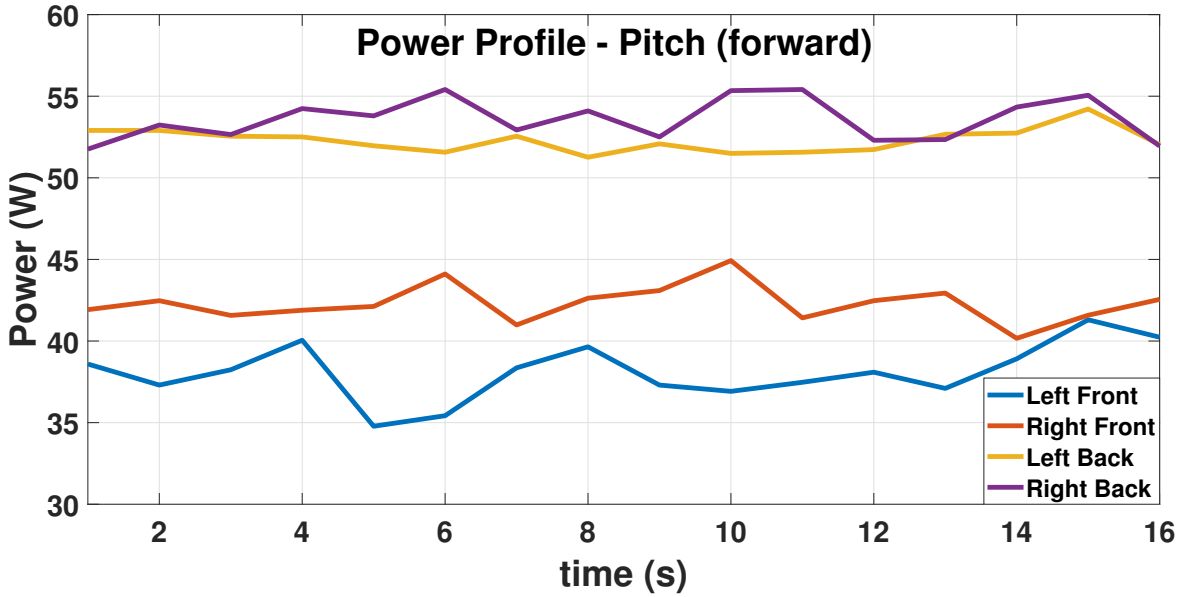

Figure 3.5: Power Profile during Pitch(forward)

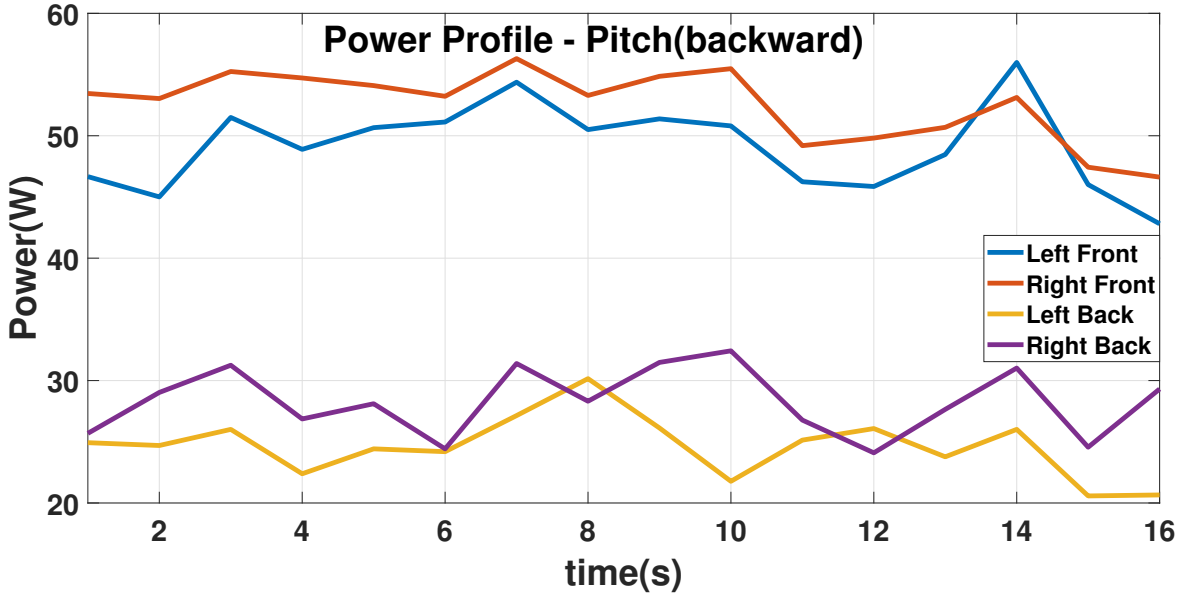

Figure 3.6: Power Profile during Pitch(backward)

3.3 Experiment 3 - Roll maneuver

In this experiment, the quadrotor is made to roll to the Left(L) and undertake a steady flight. The horizontal velocity achieved during this maneuver is 12.01 m/s and the power profile is shown in Figure 3.7. The front right and back right rotors consume the most power, which is obvious for this type of maneuver but there is a difference of 13.37 W between the right side rotors which is not obvious. This can probably be because of the geometric arrangements of the rotors or some headwind and tailwind that the quadrotor is trying to overcome. Similarly another maneuver is tested with a roll to the Right(R), where the left rotors consume most power. The power figures are tabulated in Table 3.2 and the power profile is shown in Figure 3.8. A different set of roll experiments are also performed in sport mode with a horizontal velocities of 18.94 m/s (roll left) and 18.4 m/s (roll right) and the average power consumed are tabulated in Table 3.3.

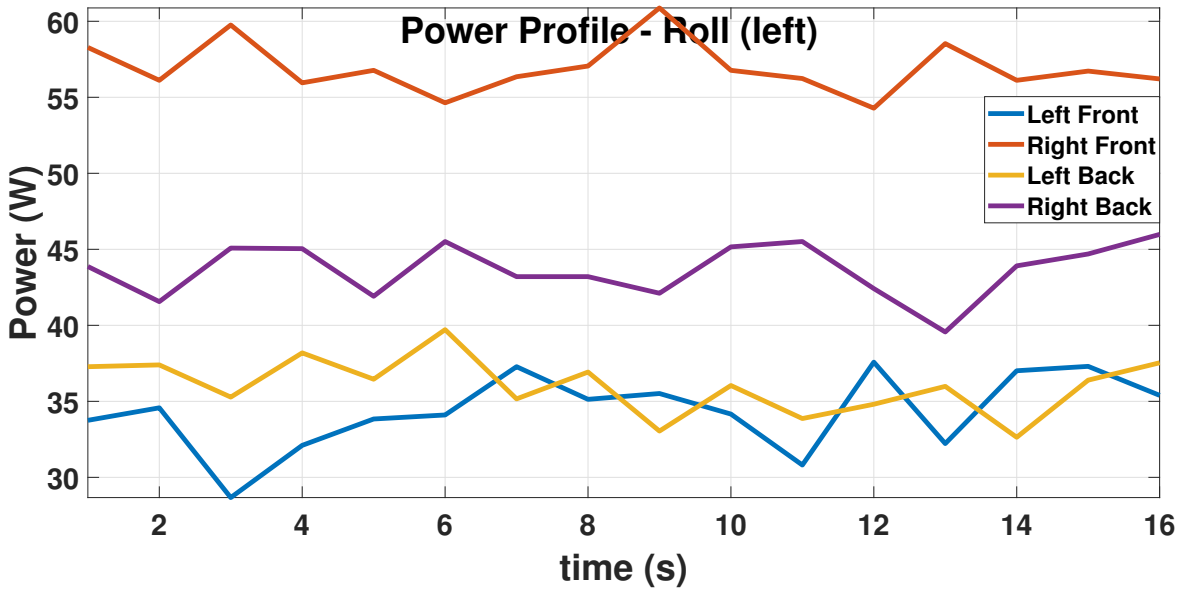

Figure 3.7: Power Profile during Roll(left)

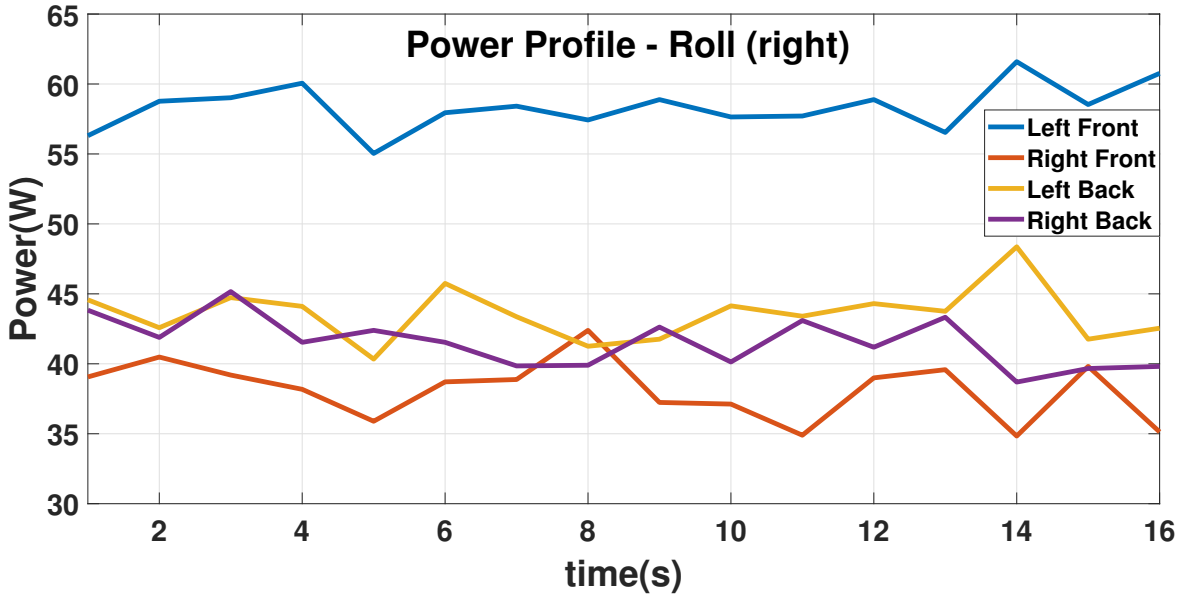

Figure 3.8: Power Profile during Roll(right)

3.4 Experiment 4 - Ascend and Descend maneuver

In this experiment the quadrotor is commanded for steady ascend (vertical velocity being 4.14 m/s) and steady descend (vertical velocity being −3.02 m/s) and the power profiles are plotted in Figure 3.9 and Figure 3.10 respectively. Steady descend consumes the least power in all of the maneuvers.

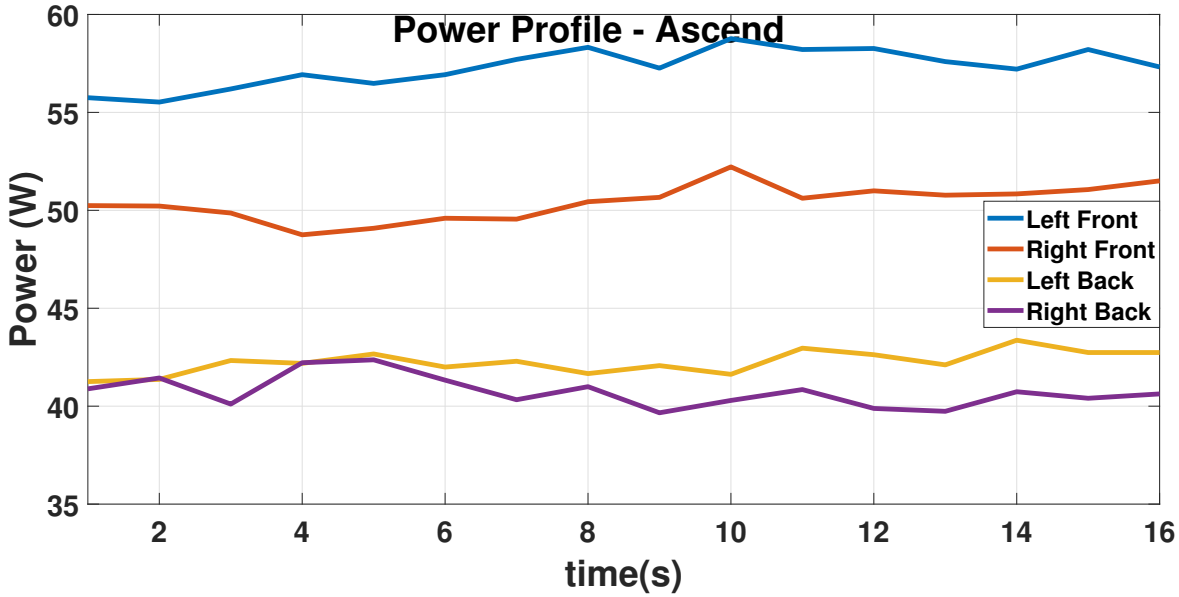

Figure 3.9: Power Profile during ascend

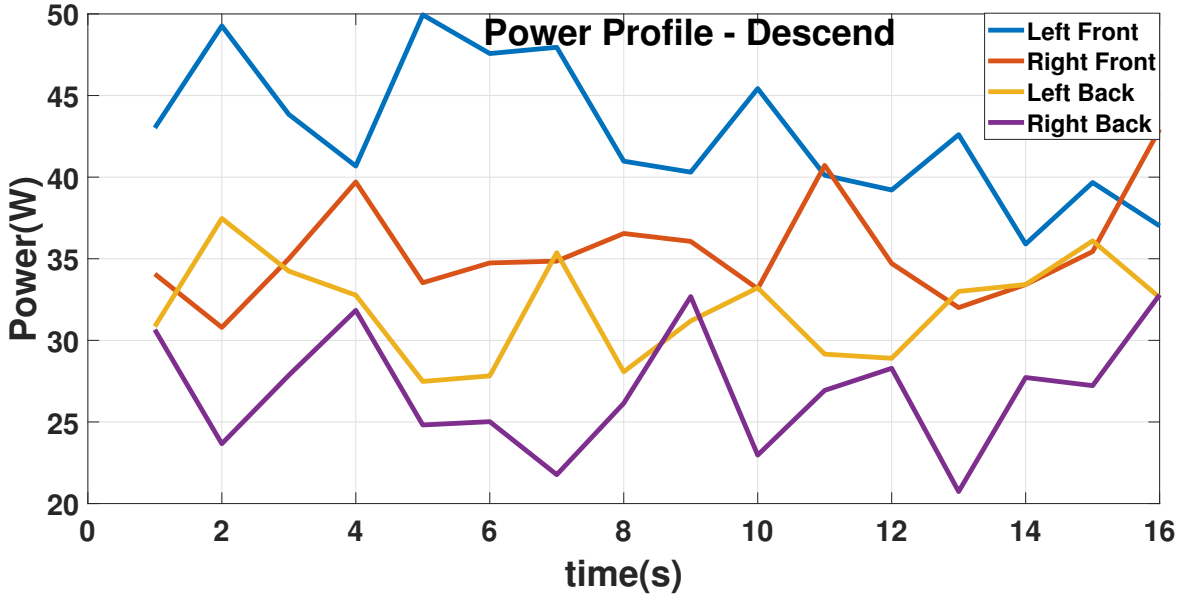

Figure 3.10: Power Profile during descend

3.5 Experiment 5 - Yaw maneuver

In this experiment, the vehicle is commanded to yaw at an altitude of 53 m, first Counter-Clockwise (CCL) and then Clockwise (CL) and the power consumption profiles are shown in Figure 3.11 and Figure 3.12

respectively. Both these maneuvers consume identical power as shown in Table 3.2.

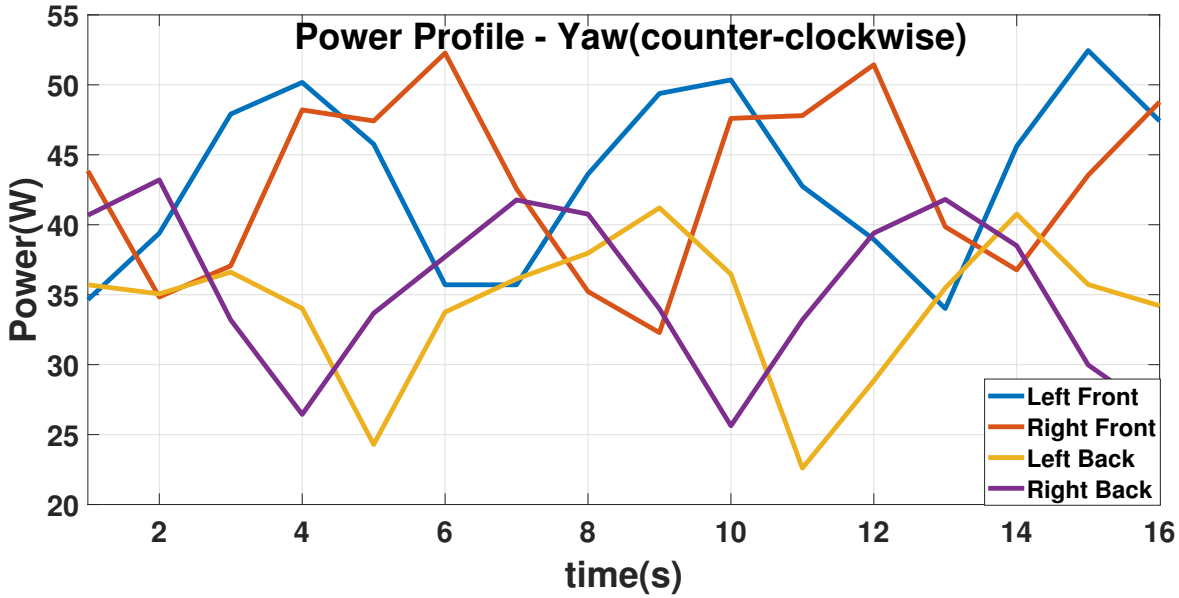

Figure 3.11: Power Profile - Yaw(counter-clockwise)

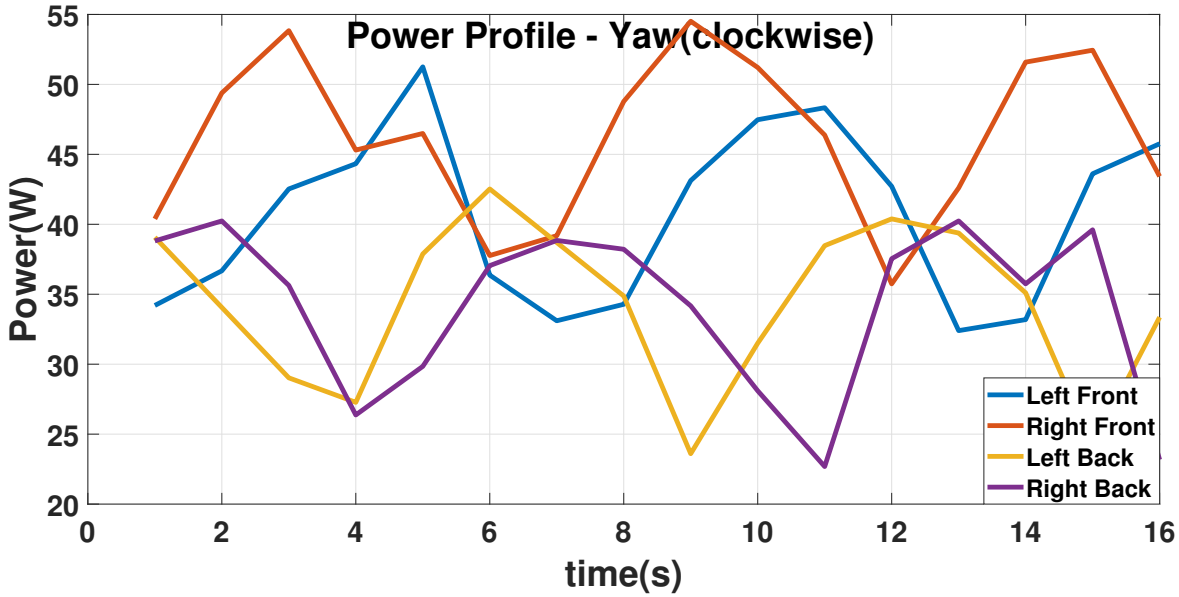

Figure 3.12: Power Profile - Yaw(clockwise)

3.6 Experiment 6 - Ascend/Descend + Pitch maneuver

In this experiment the vehicle is commanded to simultaneously ascend and pitch (backward) in normal mode. The horizontal and vertical velocity was 8.6 m/s and 3.99 m/s respectively. We see approximately 30 W of power difference between the front and back rotors. The power profile is shown in Figure 3.13. Similarly the vehicle is commanded to simultaneously descend and pitch (forward) and the power profile is shown in Figure 3.14. A different set of the same experiment is performed in sport mode where the

horizontal and vertical velocities for ascend and pitch (backward) is 15.14 m/s and 4.02 m/s respectively. The power difference between front and back rotors is approximately 40 W and the total power difference between normal and sport mode is approximately 57 W. For descend and pitch (backward) in sport mode, there is a power difference of approximately 66 W with normal mode.

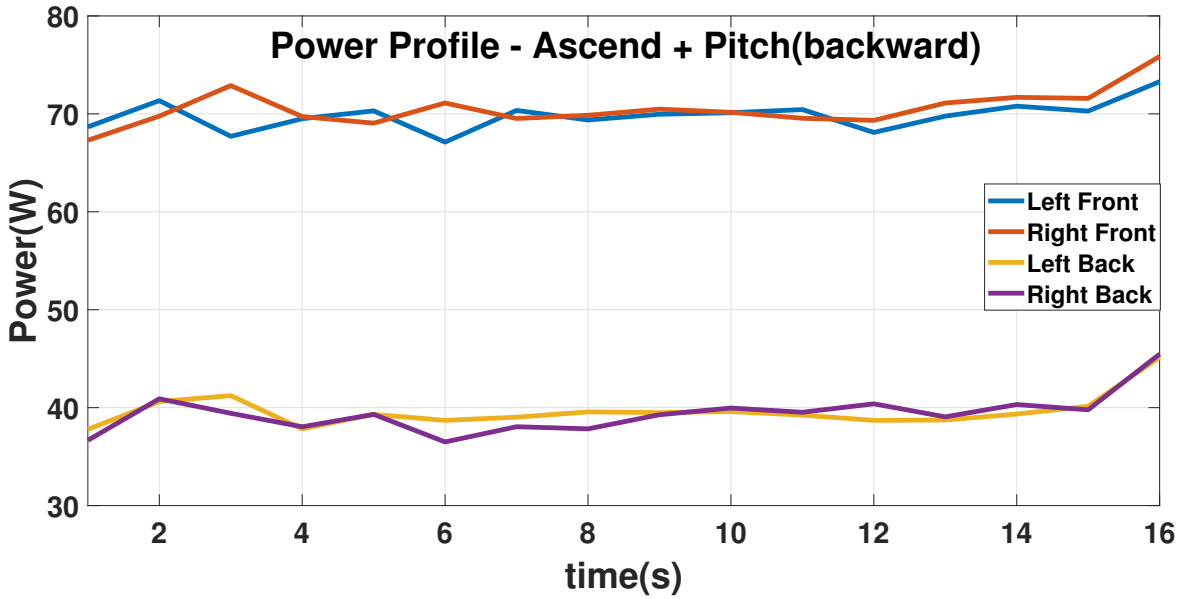

Figure 3.13: Power Profile - Ascend+Pitch(backward)

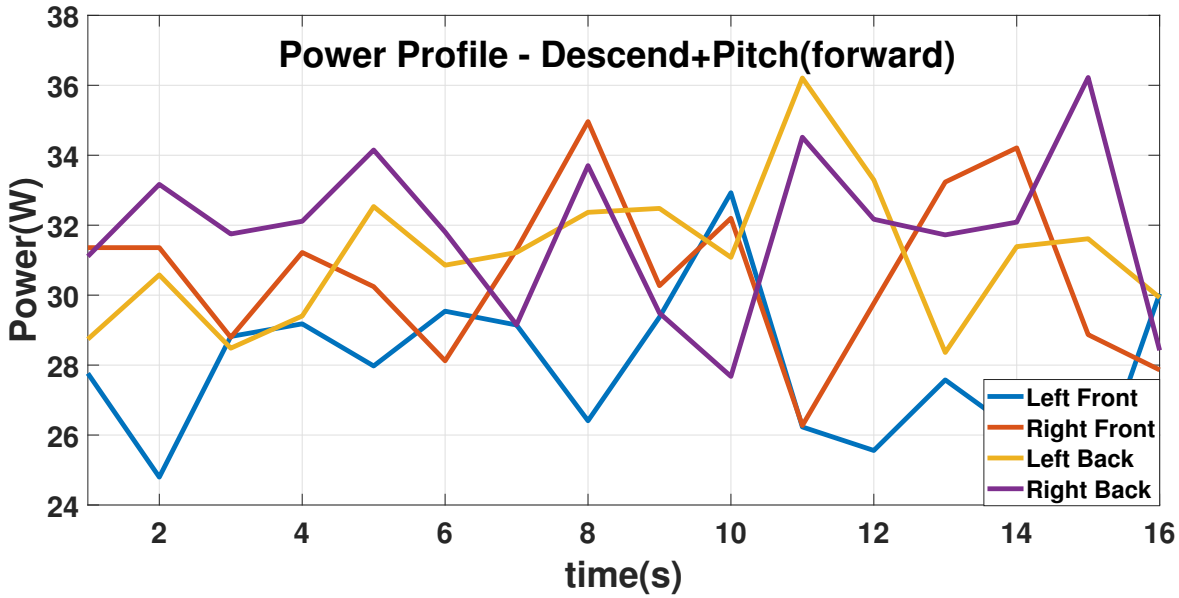

Figure 3.14: Power Profile - Descend+Pitch(forward)

Table 3.2: Average Power - normal mode

Maneuver	**Motor Power (W)**				**Total Power(W)**
	Left Front	Right Front	Left Back	Right Back	
Hover	38.01	42.16	33.37	31.8	145.34
Pitch(F)	38.09	42.26	52.32	53.53	186.2
Pitch(B)	49.31	52.76	24.76	28.05	154.88
Roll(L)	34.26	57.04	36.03	43.67	171
Roll(R)	58.12	38.22	43.67	41.72	181.73
Ascend	57.24	50.36	42.2	40.7	190.5
Descend	42.7	35.3	31.7	27.09	136.79
Ascend+Pitch(B)	69.59	70.21	39.33	39	218.13
Descend+Pitch(F)	27.87	30.73	31.06	31.84	121.5
Yaw(CCL)	42.81	43.22	34.36	35.76	156.15
Yaw(CL)	40.14	45.81	34.66	34.58	155.19

Table 3.3: Average Power - sport mode

Maneuver	**Motor Power (W)**				**Total Power(W)**
	Left Front	Right Front	Left Back	Right Back	
Pitch(F)	57.42	58.66	60.79	120.57	297.44
Pitch(B)	76.95	76.32	39.27	38.36	230.9
Roll(R)	92.8	61.53	55.35	53.28	262.96
Roll(L)	55.89	86.46	49.71	52.61	244.67
Ascend+Pitch(B)	88.31	88.71	49.41	49.21	275.64
Descend+Pitch(F)	37.42	40.37	52.93	56.65	187.37

3.7 Comparison with a Power Model

In this section, we compare the power that is calculated as per the proceedings of the model derived by Lie et al. in [8] with the actual power that is consumed by the DJI Mavic Air 2 for the maneuvers discussed in previous section. The proceedings of the power calculation is discussed in the following subsection

3.7.1 Power calculation using Liu et. al power model

The following symbols will be required to define the different notions of power in component based modeling -
T : Total thrust applied by the UAV
k_1 : Ratio of actual airflow to idealised uniform airflow
ρ : Density of air
A : Total propeller area
V_{vert} : Vertical velocity of the UAV
V_{air} : Horizontal velocity of the UAV
N : Total number of blades in a single propeller
M : Total number of rotors
c_d : Drag coefficient of the blade
c : Blade chord width
C_d : Drag coefficient of vehicle body

R : Radius of the propeller blade
ω_i : Angular speed of i^{th} rotor
μ_i : Advance ratio for propellers in rotor i
α_i : Angle of attack for propeller disks in rotor i
V_{wind} : Velocity of wind head on to the UAV
V_{ground} : Ground velocity of the UAV
A_{quad} : Cross sectional area of the vehicle when against wind
c_l : Lift coefficient
According to [8], the power consumed by a quadrotor is the composed of three components namely induced power (P_i), profile power (P_p) and parasitic power (P_{par})

1. Induced Power - The induced power is the power required to keep the UAV afloat. The modeling of induced power is derived from disk actuator theory [14].

$$P_i = k_1 T \left(\sqrt{\frac{T}{2\rho A} + \left(\frac{V_{vert}}{2}\right)^2} + \frac{V_{vert}}{2} \right) \tag{3.1}$$

2. Profile Power - The profile power is the power required to overcome the rotational drag encountered by rotating propeller blades. The profile power consumed by a rotating rotor blade is derived from blade element theory [17]. The profile power for the i^{th} rotor while the UAV is hovering is given by

$$P_{p,hover,i} = \frac{N \times c \times c_d \times \rho \times R^4}{8} {\omega_i}^3 \tag{3.2}$$

During horizontal flight, the profile power becomes

$$P_{p,i} = P_{p,hover,i} \left(1 + {\mu_i}^2\right) \tag{3.3}$$

where,

$$\mu_i = \frac{V_{air} cos(\alpha_i)}{\omega_i R} \tag{3.4}$$

In addition, all angles of attack are identical. Then the total profile power is

$$P_p = \sum_{i=1}^{M} P_{p,i} = \sum_{i=1}^{M} \left(\frac{N \times c \times c_d \times \rho \times R^4}{8} {\omega_i}^3 \left(1 + {\mu_i}^2\right) \right) \tag{3.5}$$

$$P_p = \sum_{i=1}^{M} \left(\frac{N \times c \times c_d \times \rho \times R^4}{8} \left({\omega_i}^3 + \left(\frac{V_{air} cos(\alpha_i)}{R} \right)^2 \omega_i \right) \right) \tag{3.6}$$

3. Parasitic Power - The parasite power is the power required to resist body drag when there is relative translational motion between the vehicle and wind. The parasite power is obtained by assuming that the body drag (F_{par}) is proportional to airspeed (V_{air}) squared.

$$P_{par} = \frac{1}{2} C_d \times \rho \times A_{quad} \times {V_{air}}^3 \tag{3.7}$$

These power equations are applicable only when the UAV is at steady state, i.e in force equilibrium. Figure 4.1 shows all the external forces, where L is the lift, D is the parasitic drag, α is the angle of attack.

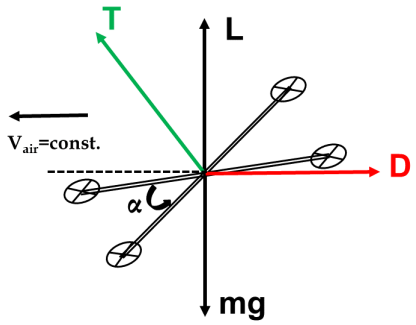

Figure 3.15: The UAV under force equilibrium in horizontal and vertical directions

Therefore, the thrust at steady state is computed from

$$T = \sqrt{(mg - L)^2 + D^2} \tag{3.8}$$

The lift as described in [8] is given by

$$L = c_5(V_{air}cos\alpha)^2 + c_6 T \tag{3.9}$$

where $c_5 = \frac{N \times c \times c_l \times \rho \times R}{4}$ and $c_6 = k_3 \frac{N \times c \times c_l \times \rho \times R^3}{6}$.

$$D = c_4 V_{air}^2 \tag{3.10}$$

where $c_4 = \frac{C_d \times \rho \times A_{quad}}{2}$.
Theoretically, solution for T can be obtained by solving a complicated quadratic equation, by substituting equation (3.9) into equation (3.8). However, we are only interested in modeling lift as a function of horizontal airspeed. It allows us to match the thrust reduction observed in horizontal flight. Therefore, it is further assumed that $c_6 = 0$, to simplify the identification process.

Simplified analytical expressions for Power The power consumed by an UAV needs to be calculated using the powers that are described above. From equations (3.1), (3.6) and eq(3.7), it is evident that there are a lot of parameters at play and it is not convenient to have an exact estimation of all the parameters during every operation. The parameters that are dependent on an UAV operation are Thrust T which is a measure of the weight carried by the UAV, V_{air}, the horizontal speed and V_{vert}, the vertical speed of the UAV.
So a power equation needs to be formulated based on these variables. Accordingly, equations (3.1),(3.6) and (3.7) can be encapsulated as

$$P_i(T, V_{vert}) = P_i = k_1 T \left(\frac{V_{vert}}{2} + \sqrt{\frac{T}{{k_2}^2} + \left(\frac{V_{vert}}{2}\right)^2} \right) \tag{3.11}$$

where, $k_2 = \sqrt{2\rho A}$
For a multi-rotor UAV, it is common to assume that the thrust generated is proportional to the angular speed squared [18], or $T_i = k_3 {\omega_i}^2$ where, k_3 is a scaling factor converting from rotor angular speed ω to thrust T.

$$P_p(T, V_{air}) = c_2 T^{\frac{3}{2}} + c_3 (V_{air}cos\alpha)^2 T^{\frac{1}{2}} \tag{3.12}$$

$$P_{par}(V_{air}) = c_4 {V_{air}}^3 \tag{3.13}$$

$$V_{air} = \|V_{air}\| = \|V_{ground} - V_{wind}\| \tag{3.14}$$

$$T = \sqrt{(mg - (c_5(V_{air}cos\alpha)^2 + c_6T))^2 + (c_4{V_{air}}^2)^2} \tag{3.15}$$

While hovering, $V_{vert} = 0$ and the induced power is reduced to

$$P_{i,hover}(T) = \frac{k_1}{k_2}T^{\frac{3}{2}} = c_1T^{\frac{3}{2}} \tag{3.16}$$

Therefore, $k_1, k_2, c_1, c_2, c_3, c_4, c_5$ are the parameters that are to be identified

Model Identification using experiments From equation (3.11) to (3.15), the power components are almost linear functions of payload and airspeed. To identify the unknown coefficients, three simple experiments are performed, namely hover, steady-state ascend/descend, and cyclic straight-line mission, on a UAV the power modeling of which we are interested in. In each experiment, the total power drawn is computed by multiplying voltage and current measurements from an onboard power module fitted in the UAV.

1. Experiment 1 - Hover - In this experiment, the UAV is loaded with different payloads and made to hover. The total power is given by equation (3.17) from which $(c_1 + c_2)$ can obtained . A least square fit of power with varying payload weight can also be derived.

$$P_{exp1} = P_{i,hover}(mg, 0) + P_p(mg, 0) = (c_1 + c_2)(mg)^{\frac{3}{2}} \tag{3.17}$$

2. Experiment 2 - Steady State ascend/descend - In this experiment, the UAV is commanded to ascend and descend at constant vertical speed between a defined altitude range without payloads. The total power is given by equation (3.18). Together with Experiment 1, there are four equations (hover, ascend, descend, and $c_1 = k_1/k_2$), and four unknown parameters (k_1, k_2, c_1, and c_2).

$$P_{exp2} = P_i(mg, V_{vert}) + P_p(mg, 0) \tag{3.18}$$

3. Experiment 3 - Cyclical straight lines - The goal of this experiment is to quantify the effect of parasite drag. The total power is given by equation (3.19), with the thrust T, and lift L and drag D defined by equation (3.8) and (3.9), respectively. The parameter c_3 is assumed to be 0 to simplify the identification process

$$P_{exp3} = P_i(T, 0) + P_p(T, V_{air}) + P_{par}(V_{air}) \tag{3.19}$$

$$= (c_1 + c_2)T^{\frac{3}{2}} + c_3(V_{air}cos\alpha)^2T^{\frac{1}{2}} + c_4V_{air}^3 \tag{3.20}$$

$$\simeq (c_1 + c_2)T^{\frac{3}{2}} + c_4V_{air}^3 \tag{3.21}$$

The coefficients that supports the power formulae for DJI Mavic Air 2 are obtained by performing Experiment 1,2 and 3 and shown in Table 3.4.

Table 3.4: Power Model co-efficients for DJI Air 2

Parameter	Value	Parameter	Value
m	0.57 kg	c2	9.02 $(m/kg)^{1/2}$
g	9.8 m/s2	c3	~0
k1	2.4795	c4	-0.033611 kg/m
k2	1.2346 $(kg/m)^{1/2}$	c5	-0.0048941 Ns/m
c1	1.99 $(m/kg)^{1/2}$	c6	~0

Table 3.5: Comparison of Power with a power model

Maneuver	**(hv,vv) (m/s,m/s)**	**Total Power (W)**	**Power from [8] (W)**	**Power Difference (W)**	**% Power difference**
Normal mode					
Hover	(0,0)	145.34	145.35	-0.01	-0.00
Pitch(F)	(11.9,0)	186.2	186.86	-0.66	-0.35
Pitch(B)	(11.98,0)	154.88	187.86	-32.98	-21.29
Roll(L)	(12.01,0)	171	188.25	-17.25	-10.08
Roll(R)	(11.1,0)	181.73	177.77	3.96	2.18
Ascend	(0,4.14)	190.5	186.18	4.32	2.27
Descend	(0,-3.02)	136.79	131.87	4.92	3.59
Ascend+ Pitch(B)	(8.61,3.99)	218.13	203.33	14.8	6.78
Descend+ Pitch(F)	(11.1,-4.9)	121.5	152.87	-31.37	-25.81
Yaw(CCL)	(0,0)	156.15	145.35	10.8	6.91
Yaw(CL)	(0,0)	155.19	145.35	9.84	6.34
Sport mode					
Pitch(F)	(16.38,0)	297.44	273.52	23.92	8.04
Pitch(B)	(16.82,0)	230.9	285.78	-54.88	-23.76
Roll(R)	(18.4,0)	262.96	336.06	-73.1	-27.79
Roll(L)	(18.94,0)	244.67	355.58	-110.91	-45.33
Ascend+ Pitch(B)	(15.14,4.02)	275.64	310.55	-34.91	-12.66
Descend+ Pitch(F)	(17.95,-5)	187.37	270.18	-82.81	-44.19

Table 3.5 shows a comparison of the power obtained using the Liu et al. model in [8] with the actual power that is consumed by the quadrotor for various maneuvers. The data for hover, pitch (forward), ascend and descend (marked in blue) are used to obtain the coefficients of the model and hence they do not support any comparison. The pitch (backward) maneuver is considered as the same maneuver as pitch (forward) by the power model in [8] but practically there is a power consumption difference of approximately 21%. The power in roll (left) maneuver has a 10% deviation in power consumption from the model prediction while the roll (right) maneuver has no significant difference. Again descend+pitch (backward) shows a significant difference of approximately 25% while ascend+pitch (forward) shows a difference of approximately 6% in power consumption. Yaw maneuvers stand at a deviation of approximately 6%. There are significant power differences in most maneuvers which indicates that the power model in [8] can not accurately capture the dynamic nature of power consumption. The percentage difference in power increases with higher speed as seen from the sport mode data.

Chapter 4

Path Planning

4.1 What is path planning

Path planning refers to the generation of a set of waypoints between an initial location and a desired destination with an optimal or near-optimal performance under specific constraint conditions. Path planning is performed to ensure the following attributes [19] :

- Stealth: Stealth means safety. The topography of an area through which the UAV travels may include buildings, mountains, forest cover etc, so it is essential that it doesnot crash against any such objects. Moreover, in military applications, UAVs have to ensure that they stay away from the realm of enemy radars or evade when there is a pursuit by an enemy UAV.
- Physical Feasibility: The physical feasibility of a route refers to the physical limitations from the use of UAVs. They include the following constraints.
  - Flying time
  - Payload weight
  - Communication range
  - Maximum translational velocity
  - Maximum climb/sink rate
  - On-board computational power

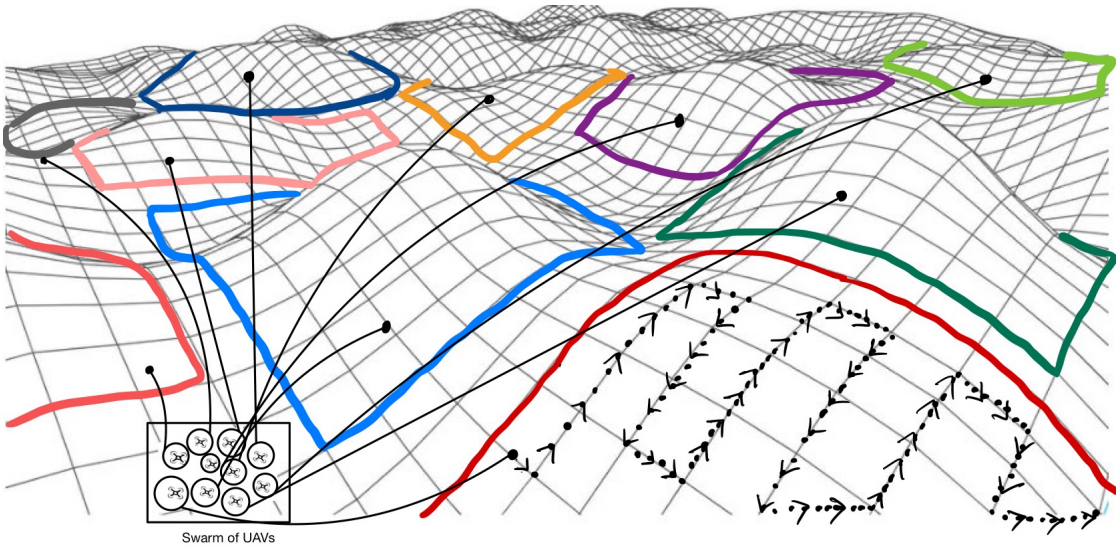

Figure 4.1: A quadrotor swarm surveying a 3D terrain

- Performance of Mission: Each flight has its special mission. This depends on the application. For example in applications like emergency responsiveness, the operator would want to reduce the time of operation while in

parcel delivery, the operator would like to reduce the cost of operation. Certain missions require multiple trips by the same UAV to improve the efficiency of operation.

- Cooperation: The path planning algorithm must be compatible with the cooperative nature envisioned for the use of UAVs. A flight mission might involve multiple UAVs and in such an applications, the ability to coordinate with the routes flown by other UAVs become vital for a smooth and efficient operation.

- Real-Time Implementation: The flight environments of the UAVs are usually constantly changing. Therefore, the route-planning algorithm must be able to adapt to those changes and at the same time be computationally efficient.

In the next section we enumerate a brief literature review on Path Planning of UAVs.

4.2 Literature Review on path planning

The main objective of UAV path planning is to design flight paths with minimum comprehensive costs and maximum safety. The major dimensions of UAV path planning in literature are based on aspects of algorithm (deterministic/non-deterministic), time-domain (online/offline), and space domain (2D/3D)

- Dorling et. al in [7] have solved Vehicle Routing Problems as Multi-trip Vehicle Routing Problem (MTVRP) while considering battery and payload weight as parameters in the energy consumption modeling that governs the cost of a trip. Two MTVRPs are solved, one that minimizes the time of an operation given a budget and other that minimizes the cost of an operation given a time limit. The authors have derived their own energy consumption model and claimed that the energy consumed is irrespective of whether the drone is hovering or flying with constant velocity. They have also concluded that energy consumption varies linearly with weight. The authors have derived MILPs for the VRP and have also used SA approach to obtain suboptimal solutions when the region of application is very large. Given an operation, solutions to the VRP are the number of UAVs, the routes they fly, battery weight and payload weight. It is shown that minimum time has an inverse exponential relationship with minimum cost.

- Authors of [20] solve a multi-modal path planning problem for UAVs under a low altitude dynamic urban environment. A Multi-objective path planning (MOPP) framework concerning travel time and safety level has been proposed. To this end, a static SIM is offline established to indicate the main static obstacles in the geography map, and a dynamic SIM is online constructed to capture unexpected obstacles that are not available in the geography map during flying. Then a joint offline and online search method has been developed to address the MOPP problem. The performance of the MOPP is evaluated using metrics consisting of the average and maximum runtime of the program, the UAV trajectory, travel time, and the total safety index. A travel time and safety index tradeoff curve is provided which can provide the users to select a Pareto optimal path according to their preferences. The perception range used for dynamic SIM decides the travel time in case of online replanning but that is restricted to the sensing infrastructure that is available during practical implementation.

- A Multi-Step A*(MSA*) search algorithm is proposed in [21] for four-dimensional multi-objective path panning of an UAV in a large dynamic environment. Generally, the path planning algorithms outputs a sequence of linear tracks in any grid-based motion planning scheme where track angle and velocity are restricted by assumptions. Here in this work, MSA* employs a variable successor operator and finds a cost-optimal path using variable length, angle and velocity trajectory segments. The multiple objectives addressed are safety, flying rules, delivery time and fuel consumption. The constraints considered in flight are cruise velocity, altitude, rate of climb, turn radius, vehicle separation, storm cell avoidance and population risk criterion. It is concluded that, on average, the computational time of MSA* is four times better than A* while the total cost is only marginally improved. It has been also shown that MSA* is suitable for online replanning as the average computation time is a fraction of the minimum track traversal time.

- Roberge et. al in [22] have used and compared two evolutionary algorithms for solving the real-time path planning problem of an UAV in a complex 3D environment. They have proposed a comprehensive cost function that includes optimal criteria like the length of the path, altitude and danger zones and feasible criteria like power availability, and collision avoidance. The cost function can easily be integrated with any other non-deterministic algorithm. They have concluded that GA produces superior path planning results than PSO. Later they have developed a parallel computing paradigm between these two algorithms in a single program and improved its execution time. It is classified as a multiple-deme parallel GA where sub-populations evolve independently while allowing some level of migration between the demes.

- Sundar and Rathinam in [23] have proposed an algorithm for obtaining a sequence of routes for an UAV in the presence of multiple refueling depots. The task of the UAV is to visit a set of target points ideally for a surveillance application in a region with the flexibility of using multiple refueling stations such that the overall fuel consumption by the UAV is minimized. The UAV in this context is modeled as a Dubins' vehicle with a minimum turning radius and an optimal heading is considered at each target location. This combinatorial difficulty is handled using an approximation algorithm with some added heuristic layers and it is seen that solutions with costs within 1.4% of the optimum are obtained relatively quickly. It is also shown that when heuristics are added, the average deviation of the suboptimality of feasible solutions are comparatively lower as the number of targets increase.

Based on the literature reviewed we enumerate few possible research gaps.

4.3 Research Gap on Path Planning

- On a multi-objective path planning scheme, the objective of minimizing energy consumption is avoided in most cases. The cases that consider energy consumption as an objective are grid-based path planning, where the path is assumed as a straight line from one point to the other[24]. The energy consumption modeling can be tested against various types of trajectories and accordingly an optimal/quasi-optimal trajectory can be calculated that would result in the least energy consumption.

- To the best of our knowledge, online path planning is not formulated in the literature when an UAV gets dissuaded from its path due to environmental factors like wind. Consider a case where a UAV is dissuaded from its preplanned path due to external stimuli. The path planning algorithm should be able to replan a trajectory from that instant such that energy is minimized for the rest of the journey.

- To the best of our knowledge, weather data is not taken into consideration during the path planning of an UAV. In civil aviation, a flight plan is designed based on the current weather which may include waypoints that are placed according to the wind directions of the region[25]. Such flight plans aid the flight with minimum fuel burnt against the flow of wind. Similarly in UAV path planning, wind data can be considered for an efficient flight.

Next we step into a formulation of path planning based on minimum energy consumption. The idea leading into minimum energy trajectories are discussed in the next section.

4.4 Path Planning with minimum energy consumption

Our goal is to minimize the energy that will be consumed in a trip. We achieve it by evaluating how expensive a trajectory is in terms of power using the power model derived in Section 2.6 and then choosing the least expensive trajectory. In this section we test the power attributes of two trajectories using our power model.
Let the objective of a quadrotor is to travel 11 meters along the x-axis in 4 seconds.
We define two trajectories as follows -

Trajectory 1 -

$$p(t) = 0.3125t^3 - 0.058t^4 + 0.0058t^5 \tag{4.1}$$
$$v(t) = 0.9375t^2 - 0.232t^3 + 0.029t^4 \tag{4.2}$$
$$a(t) = 0.9375t - 0.703t^2 + 0.117t^3 \tag{4.3}$$

The magnitude vs time plot of equations (4.1),(4.2),(4.3) is shown in Figure 4.2

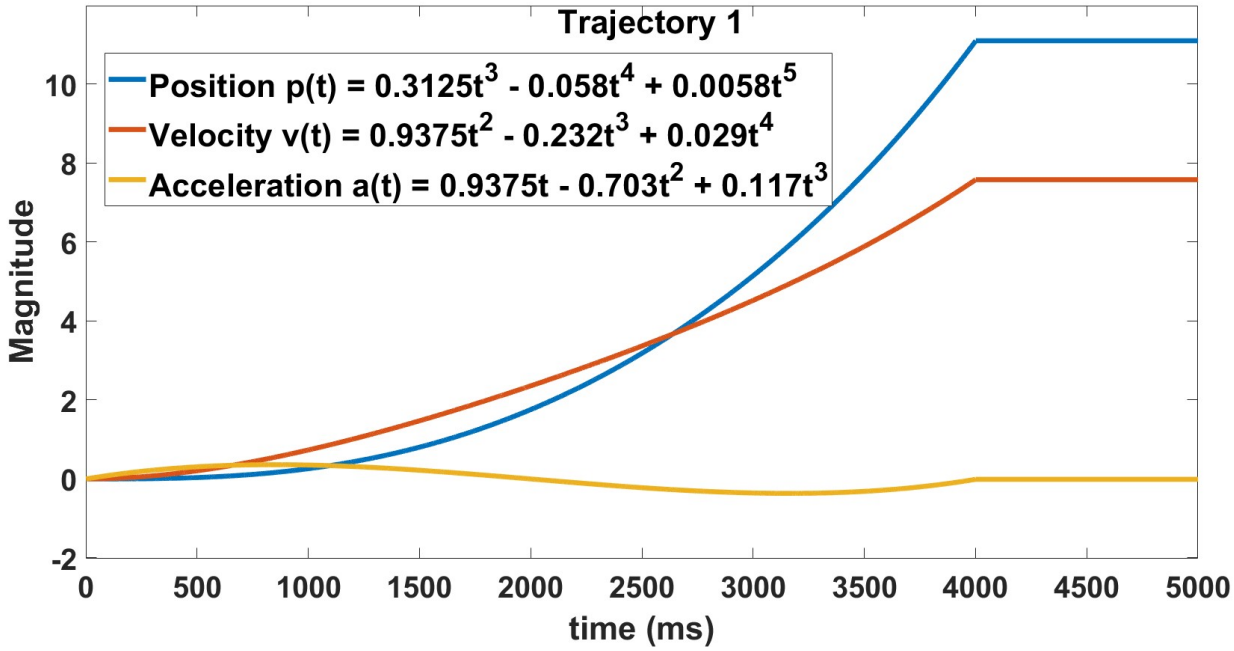

Figure 4.2: Position, Velocity, Acceleration vs time

Trajectory 2 -

$$p(t) = t^2 \tag{4.4}$$
$$v(t) = 2t \tag{4.5}$$
$$a(t) = 2 \tag{4.6}$$

The magnitude vs time plot of equations (4.4),(4.5),(4.6) is shown in Figure 4.3

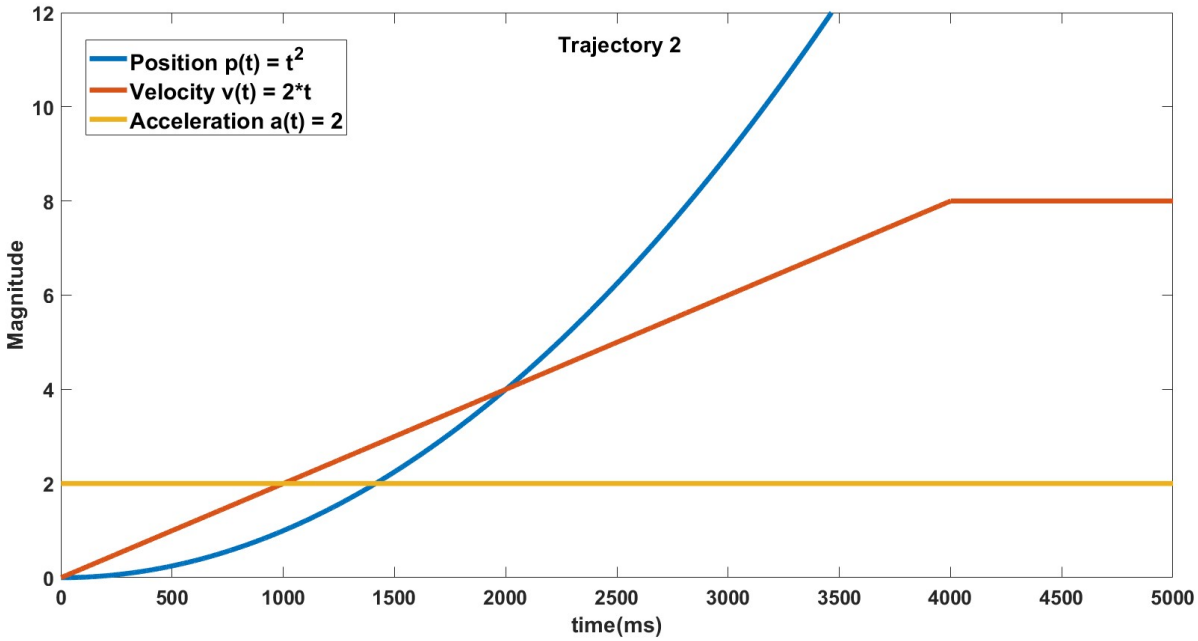

Figure 4.3: Position, Velocity, Acceleration vs time

For estimating power using our power model let us define the following -

Quadrotor attributes -

$$m = 0.18kg, L = 0.018m, I = \begin{bmatrix} 0.0025 & 0 & 0 \\ 0 & 0.000232 & 0 \\ 0 & 0 & 0.0003738 \end{bmatrix} kg.m.m$$

Motor Parameters -

$R_a = 1\Omega, L_a = 1H, J = 5Kg.m.m, B = 0.01N.m.s, K = 1.6V/rad/s, K_F = K_M = 1, v(t) = 1V$

Controller Gains -

Attitude Controller - $\begin{bmatrix} k_{p,\phi} \\ k_{p,\theta} \\ k_{p,\psi} \end{bmatrix} = \begin{bmatrix} 100 \\ 100 \\ 100 \end{bmatrix}, \begin{bmatrix} k_{d,\phi} \\ k_{d,\theta} \\ k_{d,\psi} \end{bmatrix} = \begin{bmatrix} 2 \\ 2 \\ 2 \end{bmatrix}$

Position Controller - $\begin{bmatrix} k_{p,x} \\ k_{p,y} \\ k_{p,z} \end{bmatrix} = \begin{bmatrix} 200 \\ 200 \\ 100 \end{bmatrix}, \begin{bmatrix} k_{d,x} \\ k_{d,y} \\ k_{d,z} \end{bmatrix} = \begin{bmatrix} 40 \\ 40 \\ 20 \end{bmatrix}$

The power aspects for these two trajectories is tablulated in Table 4.1

Table 4.1: Power consumption based on trajectories

Trajectory	**Time of convergence(s)**	**Rotor Power**				**Total Power (W)**
		P1	**P2**	**P3**	**P4**	
Trajectory 1	4	0.4796	0.6602	0.7493	0.6602	2.5439
Trajectory 2	3.3	0.6695	0.7126	0.6889	0.7126	2.7836

We see that Trajectory 1 consumes 9.4% less power compared to Trajectory 2.

Chapter 5

Conclusion

In this phase of work, we have attempted to derive a power model for a quadrotor where given a trajectory, the model can estimate the power profiles for each rotor as a function of time. This model may be unique because unlike models present in literature, which gives power as a static number, it gives instantaneous power as a function of time which might lead to accurate power estimations. This model is based on the linearised dynamics of a quadrotor around hover, whereas deriving the power model on the non-linear dynamics of a quadrotor will make this model more accurate and full-proof, which is a future scope of our work.
Based on our field experiments, we also get a clue that obtaining powers based on maneuvers or obtaining the complete power profiles of all four rotors of the quadrotor might lead to accurate power estimates, as the present models can not differentiate power based on maneuvers. We are also working on measuring the inertia tensor of the DJI Air 2 drone and obtaining the current-speed relationship of its propellers so that we can apply our model in estimating its power and check the validity of our hypothesis.
We have also used our model to evaluate the power aspects when a quadrotor to subjected to follow different trajectories. We will extend our work to form a framework of obtaining energy-optimal trajectories for more realistic UAV scenarios and objectives and finally devise a formulation of minimising energy consumption in a multi-agent UAV setup.

Bibliography

[1] India Brand Equity Foundation(IBEF). Indian drone industry reaching the skies.

[2] Federation of Indian Chambers of Commerce and Industry (FICCI). Smart border management-contributing to a 5 trillion dollar economy. 2019.

[3] The Hindu Business Line. Precision farming and drones can help india make the big leap, May 2021.

[4] Ministry of Civil Aviation, Govt. of India. Drone rules 2021. *The Gazette Of India, Extraordinary, Part II, Section 3, Sub Section(i)*, 2021.

[5] Mohammad Ammad Uddin, Muhammad Ayaz, El-Hadi M Aggoune, Ali Mansour, and Denis Le Jeune. Affordable broad agile farming system for rural and remote area. *IEEE Access*, 7:127098–127116, 2019.

[6] Raffaello D'Andrea. Guest editorial can drones deliver? *IEEE Transactions on Automation Science and Engineering*, 11(3):647–648, 2014.

[7] Kevin Dorling, Jordan Heinrichs, Geoffrey G Messier, and Sebastian Magierowski. Vehicle routing problems for drone delivery. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 47(1):70–85, 2016.

[8] Zhilong Liu, Raja Sengupta, and Alex Kurzhanskiy. A power consumption model for multi-rotor small unmanned aircraft systems. In *2017 International Conference on Unmanned Aircraft Systems (ICUAS)*, pages 310–315. IEEE, 2017.

[9] Thomas Kirschstein. Comparison of energy demands of drone-based and ground-based parcel delivery services. *Transportation Research Part D: Transport and Environment*, 78:102209, 2020.

[10] Ho Young Jeong, Byung Duk Song, and Seokcheon Lee. Truck-drone hybrid delivery routing: Payload-energy dependency and no-fly zones. *International Journal of Production Economics*, 214:220–233, 2019.

[11] Chien-Ming Tseng, Chi-Kin Chau, Khaled M Elbassioni, and Majid Khonji. Flight tour planning with recharging optimization for battery-operated autonomous drones. *CoRR, abs/1703.10049*, 2017.

[12] Joshuah K Stolaroff, Constantine Samaras, Emma R O'Neill, Alia Lubers, Alexandra S Mitchell, and Daniel Ceperley. Energy use and life cycle greenhouse gas emissions of drones for commercial package delivery. *Nature communications*, 9(1):1–13, 2018.

[13] Juan Zhang, James F Campbell, and Andrea C Sweeney II. Energy consumption models for delivery drones: A comparison and assessment. *Transportation Research Part D: Transport and Environment*, 90:102668, 2021.

[14] Wayne Johnson. *Helicopter Theory*. Dover Publications, 1980.

[15] United States Department Of The Army. *Fundamentals of Flight (FM 3-04.203)*. Army Knowledge Online, Washington D.C, US, 2007.

[16] M.A. Jabbar, Hla Nu Phyu, Zhejie Liu, and Chao Bi. Modeling and numerical simulation of a brushless permanent-magnet dc motor in dynamic conditions by time-stepping technique. *IEEE Transactions on Industry Applications*, 40(3):763–770, 2004.

[17] Snorri Gudmundsson. *General Aviation Aircraft Designs - Applied Methods and Procedures*. Butterworth-Heinemann, 2013.

[18] Peter Corke. *Robotics, Vision and Control: Fundamental Algorithms in MATLAB*. Springer, 2011.

[19] Changwen Zheng, Lei Li, Fanjiang Xu, Fuchun Sun, and Mingyue Ding. Evolutionary route planner for unmanned air vehicles. *IEEE Transactions on robotics*, 21(4):609–620, 2005.

[20] Chao Yin, Zhenyu Xiao, Xianbin Cao, Xing Xi, Peng Yang, and Dapeng Wu. Offline and online search: UAV multiobjective path planning under dynamic urban environment. *IEEE Internet of Things Journal*, 5(2):546–558, 2017.

[21] Paul P-Y Wu, Duncan Campbell, and Torsten Merz. Multi-objective four-dimensional vehicle motion planning in large dynamic environments. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, 41(3):621–634, 2010.

[22] Vincent Roberge, Mohammed Tarbouchi, and Gilles Labonté. Comparison of parallel genetic algorithm and particle swarm optimization for real-time UAV path planning. *IEEE Transactions on industrial informatics*, 9(1):132–141, 2012.

[23] Kaarthik Sundar and Sivakumar Rathinam. Algorithms for routing an unmanned aerial vehicle in the presence of refueling depots. *IEEE Transactions on Automation Science and Engineering*, 11(1):287–294, 2013.

[24] Rong-Jong Wai and Alex S Prasetia. Adaptive neural network control and optimal path planning of UAV surveillance system with energy consumption prediction. *IEEE Access*, 7:126137–126153, 2019.

[25] Wen-Xian Lim and Zhao-Wei Zhong. Re-planning of flight routes avoiding convective weather and the “three areas”. *IEEE Transactions on Intelligent Transportation Systems*, 19(3):868–877, 2017.