# A MODULAR SIMULATION ENVIRONMENT FOR THE IMPROVED DYNAMIC SIMULATION OF MULTIROTOR UNMANNED AERIAL VEHICLES

BY

JAN WILLEM VERVOORST

THESIS

Submitted in partial fulfillment of the requirements
for the degree of Master of Science in Systems and Entrepreneurial Engineering
in the Graduate College of the
University of Illinois at Urbana-Champaign, 2016

Urbana, Illinois

Adviser:

Professor Naira Hovakimyan

# Abstract

Multirotor unmanned aerial vehicles (UAVs) have gained immense popularity in both research and commercial applications due to their versatility and mechanical simplicity. However, despite these advantages, multirotor systems still constitute a considerable challenge in the design of powerful control architectures that guarantee safe and reliable flight performance. As is customary today, the design of guidance, control and navigation algorithms (GNC) is mostly performed in simulation. In order to guarantee a seamless transition between control solutions generated in a simulation environment and real-world flight performance, the simulation should reproduce real-world behavior with sufficient fidelity. It is of course not feasible to attempt to model every minute dynamic effect acting on the airframe, but at least the major influences should be modeled so that the simulation provides users with realistic and relevant test data that is comparable with flight test data. This thesis addresses the problem of improved modeling of multirotor UAVs for the design of GNC algorithms. First, a simplified simulation model is derived fully and complete solutions for a number of standard airframe configurations are presented. For this model to be valid, several significant simplifications and assumptions are made about the structure of the airframe. This model may already be sufficient for users who only want to simulate basic multirotor behavior, for example for the design and stability testing of low-level control algorithms. However, the model is not able to properly represent more complicated three-dimensional airframes or realistic environmental effects like wind resistance or dynamic thrust. The thesis then outlines an improved dynamic model that does not require any of the previous simplifying assumptions. This allows the model to be used for just about any imaginable multirotor airframe, regardless of symmetry or specific layout. The included environmental effects also help to make the simulation behave more natural when compared to flying a real UAV outside. The main deliverable is a *MATLAB/Simulink* simulation environment that includes all scenarios described in this thesis. It allows the user the realistic simulation of any arbitrary multirotor airframe.

# Acknowledgments

I would like to express my gratitude to my advisor, Prof. Naira Hovakimyan, for her guidance, support, and encouragement throughout my studies at the University of Illinois. Similarly, I want to thank all the people I interacted with during my time at UIUC. It's hard to pick favorites, so I want to thank all of them for playing a part in building such a fantastic academic environment. Finally, to all my officemates in MEB 349–351, thanks for making this both such a fun and academically stimulating time. I will cherish our friendship and hope to meet you again in the future.

# Table of Contents

# List of Figures

# Notation, Symbols, and Acronyms

| | |
|---|---|
| $\mathbb{I}_{\boldsymbol{n}}$ | Identity matrix of size $n$. |
| $\mathbf{0}$ | Zero matrix of appropriate dimension. |
| $\mathbf{1}_{\boldsymbol{n}}$ | Vector of $\mathbb{R}^n$ whose components are all 1. |
| $\boldsymbol{v}_F$ | Vector $\boldsymbol{v}$ resolved in frame $\{\mathcal{F}\}$. |
| $\boldsymbol{R}_{\boldsymbol{F1}}^{\boldsymbol{F2}}$ | Rotation matrix from frame $\{\mathcal{F}1\}$ to frame $\{\mathcal{F}2\}$. |
| $\boldsymbol{v}^\top$ | Transpose of vector $\boldsymbol{v}$. |
| $\|\boldsymbol{v}\|$ | 2-norm of vector $\boldsymbol{v}$. |
| $\boldsymbol{M}^\top$ | Transpose of matrix $\boldsymbol{M}$. |
| $\det(\boldsymbol{M})$ | Determinant of matrix $\boldsymbol{M}$. |
| $\text{sat}\,(\cdot)$ | The saturation function. |
| DoD | Department of Defense. |
| DoF | Degree of Freedom. |
| GNC | Guidance, Navigation and Control. |
| GPS | Global Positioning System. |
| IMU | Inertial Measurement Unit. |
| UAV | Unmanned Aerial Vehicle. |
| VTOL | Vertical Take-Off and Landing. |

Unless otherwise noted, bold-face, lower-case letters refer to column vectors (e.g. $\boldsymbol{v}$), while bold-face, capital letters refer to matrices (e.g. $\boldsymbol{M}$). In general, the $i$th component of vector $\boldsymbol{v}$ is denoted by $v_i$, and the $(i, j)$ entry of matrix $\boldsymbol{M}$ is represented by $M_{ij}$. The cross product of two vectors $\boldsymbol{v}$ and $\boldsymbol{w}$ is denoted by $\boldsymbol{v} \times \boldsymbol{w}$, while their dot product is denoted by $\boldsymbol{v} \cdot \boldsymbol{w}$.

# Chapter 1

# Introduction

In the last decade, we have observed an enormous increase in the research and use of Unmanned Aerial Vehicles (UAVs). Primarily driven by the emergence of low-cost miniature electronic systems, the number of commercially available UAV systems has skyrocketed, and together with it the number of potential applications. Nowadays, the possibilities to employ UAVs seem endless.

As is common in emerging technologies, the military has played a major role in the wide-spread adoption of unmanned systems. While well-known tactical UAVs like the *General Atomics Predator* or *Reaper* are highly complex aviation systems with expensive sensors and an impressive operational infrastructure built around them, their general capabilities, like providing live video feed of areas of interest, have been matched in the small scale on the hobby side. Today, a motivated hobbyist can easily acquire all the parts to quickly build a camera-carrying UAV that can take aerial photographs or provide a live video feed to a base station screen or video goggles for instant viewing.

One particular class of UAVs has garnered the majority of the attention in the commercial and hobby markets recently. Multirotor UAVs are a special kind of airframe that combine many positive attributes. They were designed to provide a simpler and cheaper solution to a helicopter while providing almost identical flight performance and retaining identical Vertical Take-Off and Landing (VTOL) and hover capabilities. Because of their mechanical simplicity and low cost, they are especially well-suited for miniature applications such as small indoor flyers for research. One of the first references to multirotors can be found in the early 2000s timeframe in [17] and [16], before one of the first commercial UAVs became available in about 2007, as mentioned in [5].

In a helicopter, a swashplate is used to transmit the pilot's commands from the non-rotating fuselage to the rotating rotor hub and rotor blades. This swashplate consists of a number of moving parts with bearings and linkages. They are already complex and maintenance-intensive in full size, and become even more intricate when trying to shrink them down for use in miniature UAVs. Instead of having one main rotor and a swashplate, a multirotor, as the name implies, uses multiple distributed rotors. In most cases, they are distributed equally around the aircraft's center of gravity. There are no other moving parts on the airframe

except these moving propellers. Pilot commands such as pitch, roll, and yaw are generated by varying the motors' speeds relative to each other. This generates torques around the center of gravity that move the airframe in the desired direction. We will explain these concepts in detail in Chapter 2.2. Figure 1.1 shows the most typical implementation of a multirotor, a quadrotor with four symmetrically spaced rotors.

*Source*: `www.asctec.de`

Figure 1.1: Typical example of a quadrotor UAV. Motors and propellers spaced symmetrically around the airframe's center of gravity allow the pilot to influence the aircraft's attitude and direction of travel simply by changing motor speeds between the actuators.

It is fairly obvious that such an airframe is inherently dynamically unstable, and of course it is impossible for the pilot to manually control each motor separately to generate the desired body motions. Therefore, such an aircraft can only be controlled using an onboard digital control system. A general hardware overview for such a control system can be found in [5]. Onboard sensors, consisting of at least angular rate sensors and acceleration sensors in three axes, make up an Inertial Measurement Unit (IMU) that is used to compute the aircraft's attitude or position in space. In order to do proper position control, it is of course necessary to have some kind of Global Positioning System (GPS) sensor or other means of localization on board as well. Depending on the desired control algorithms, the pilot commands angular rates, attitude angles, or position and the on-board control system generates the necessary motor commands that guarantee tracking of these input commands. Various control algorithms have been designed over the years to stabilize a multirotor, a small collection is [5], [1], [3], [2], [8], [9], and many more.

In order to efficiently test and tune control algorithms that can stabilize a multirotor platform, it is useful to have a computer simulation model for prototyping. That way no time and effort is spent on implementing potentially faulty algorithms on a physical testbed that may crash during testing. So the challenge becomes generating a simulation model that is sufficiently close to real-world behavior such that algorithms that perform well in testing will also show similar performance when testing on the real system. Looking at works concerned primarily with low-level control design such as Stability Augmentation Systems (SAS) or

attitude control, many times the basic dynamic model of the multirotor airframe may be good enough for simulation testing. See [3], [13] or [20] for examples. However, when higher-order control tasks such as velocity or trajectory tracking are desired, the discrepancies between simulation and real world become more obvious. [11] and [10] show an example how a multirotor's performance differs between commanded and actual response when tracking a trajectory. Here it will be helpful to have a more detailed simulation model so that shortcomings in the control design can be identified early on in the development.

Goal of this thesis is the implementation of an improved multirotor simulation model for the design and testing of GNC algorithms. The thesis is structured as follows:

Chapter 2 includes the full derivation of a simplified multirotor model. All necessary notation and coordinate systems are defined, necessary simplifications and assumptions are discussed and the full kinematic and dynamic equations are derived. A particular solution example is provided for the quadrotor case. Additional particular solutions for a number of popular airframes are provided in the appendix.

Chapter 3 expands on the work in Chapter 2 and details all the differences that make up the improved dynamic multirotor model. Effects like dynamic thrust, air resistance, blade flapping, actuator misalignment, and others are outlined and their effects on the system explained.

Chapter 4 explains the *MATLAB/Simulink* simulation in full detail. The contents of each block are explained it is shown how the principles outlined previously in Chapter 3 have been implemented in the simulation.

# Chapter 2

# Simplified Multirotor Model

This chapter presents the full derivation of a simplified multirotor model. The chapter first introduces all necessary notation and coordinate systems that will be used throughout the thesis. Then the necessary simplifications and assumptions for the simplified model are discussed and the full kinematic and dynamic equations are derived. A particular solution example is provided for the quadrotor case. Additional particular solutions for a number of popular airframes are provided in the appendix.

## 2.1 Coordinate Systems

As outlined in [19], in order to describe the motion of a rigid body in space, it is helpful to define two primary reference frames:

- Inertial reference frame ($I$-frame)
- Body-fixed reference frame ($B$-frame)

The Inertial frame is a stationary reference frame defined by the right-hand convention. Positions and velocities in this frame all receive the subindex $I$. Its definition is very similar to the North-East-Down (NED) frame with respect to the earth: the $x_I$ axis points North and defines forward, the $y_I$ axis points East and defines right, and the $z_I$ axis points into downward into the earth and defines up. The Body-fixed frame is fixed to the airframe body and moves with it. Positions and velocities in this frame all receive the subindex $B$. It also adheres to the right-hand rule and its axes are defined as: $x_B$ points to the front of the airframe, $y_B$ points to the right side, and $z_B$ points down. Figure 2.1 shows the two reference frames.

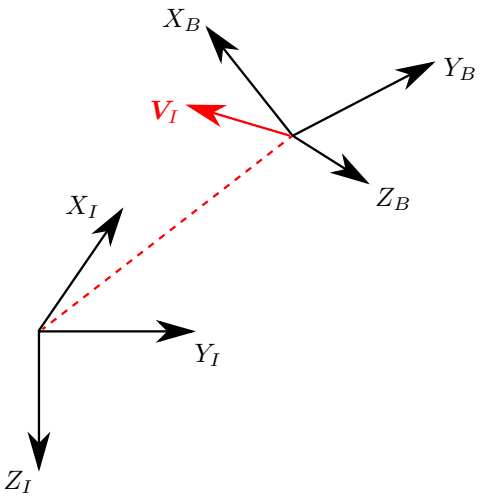
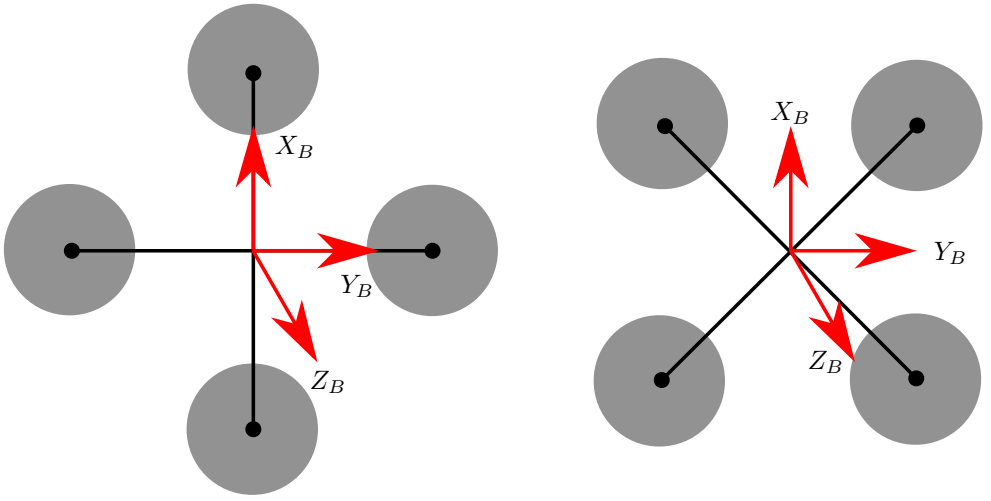



Figure 2.1: Inertial and Body coordinate systems. The axes of the Inertial coordinate system are chosen similar to the North-East-Down (NED) convention. The axes of the Body coordinate system are fixed to the airframe body and the coordinate system moves with velocity $\boldsymbol{V}_I$ relative to the Inertial frame.

Usually the Body-fixed frame is chosen for convenience such that it aligns with the geometric center of the airframe and the motor arms of the multirotor align with the reference axes in a certain way. Ultimately, it is up to the user to decide what they want to be the 'forward' direction of their aircraft. One example for this is the distinction between flying a quadrotor in '+ Mode' versus 'X Mode':

- In '+ Mode', the equally spaced motor arms of the quadrotor are exactly aligned with the Body-fixed frame, so that one arm each points in the positive/negative $x_B$ and $y_B$ directions, respectively.
- In 'X Mode', the axes of the Body-fixed frame are aligned to point exactly between the motor arms, so that the motor arms are pointing at $45°/135°/225°/315°$ in the $x - y$ plane.

The difference in airframe orientation can be seen in Figure 2.2.




(a) '+ Mode' (b) 'X Mode'

Figure 2.2: Two examples of convenient Body-fixed frame definitions for a quadrotor airframe.

## 2.2 Basic Multirotor Motion Concepts

Chapter 1 already provided a very brief overview of the reasoning behind and the mode of operation of multirotors. Multirotors have become so popular because the only moving parts are a number of distributed electric motors that spin fixed-pitch propellers. Assuming a basic multirotor model, these motors/propellers generate thrust strictly in the direction of the negative $z$-axis of the Body-fixed frame. So in order to move the airframe in a certain direction, the system has to point its overall thrust vector in that direction. This is achieved by rotating the airframe in the desired direction of travel by generating a net torque around the airframe's center of gravity. The amount of thrust generated by each of the motors is changed, meaning speeding up or slowing down certain motors as necessary. Which motors need to change their speeds is of course dependent on the physical layout of the multirotor. The basic symmetric quadrotor airframe with four propellers is chosen as a simple example to illustrate these concepts. The quadrotor is chosen to fly in '+ Mode' as described in the previous section. So the motor arms are exactly aligned with the $x_B$ and $y_B$ axes. By convention, the propeller in the front is numbered 1 and spins clockwise, the right propeller is numbered 2 and spins counterclockwise, and so on. Please see Figure 2.3 for further details.

### 2.2.1 Roll motion

In order to generate a sideways motion, the multirotor must rotate around its roll axis. For motion in the direction of the positive $y$-axis, it must rotate a positive angle around its roll axis. To achieve this, motor 4 has to generate more thrust than motor 2 so that a net torque around the Body roll axis is generated.

So motor 4 needs to speed up and motor 2 slow down by amounts $\Delta\omega_{M4}$ and $\Delta\omega_{M2}$, respectively. These changes are chosen such that the overall vertical thrust remains constant. For very small values of the roll angle, they are approximately identical.

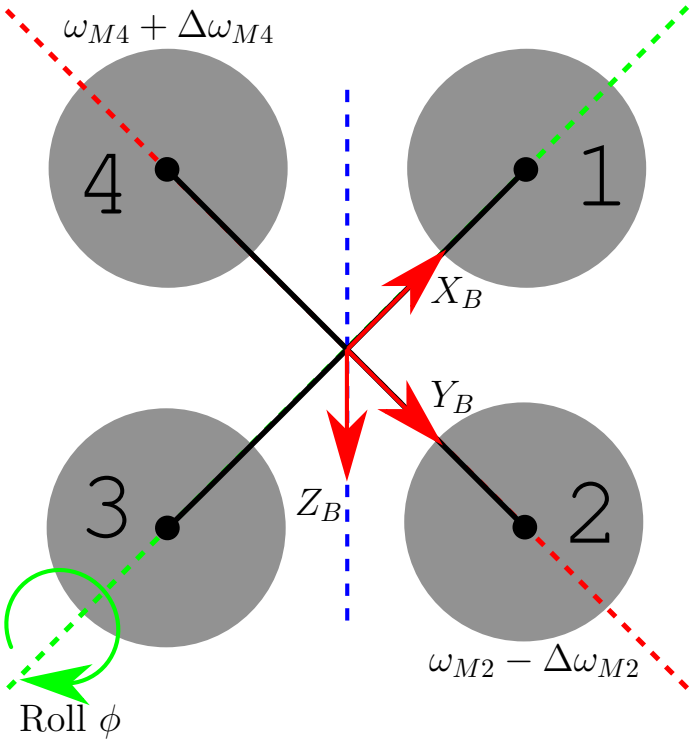



Figure 2.3: Quadrotor roll motion.

### 2.2.2 Pitch motion

In order to generate a forward/back motion, the multirotor must rotate around its pitch axis. For motion in the direction of the negative $x$-axis, it must rotate a positive angle around its pitch axis. To achieve this, motor 1 has to generate more thrust than motor 3 so that a net torque around the Body pitch axis is generated. So motor 1 needs to speed up and motor 3 slow down by amounts $\Delta\omega_{M1}$ and $\Delta\omega_{M3}$, respectively. These changes are chosen such that the overall vertical thrust remains constant. For very small values of the pitch angle, they are approximately identical.

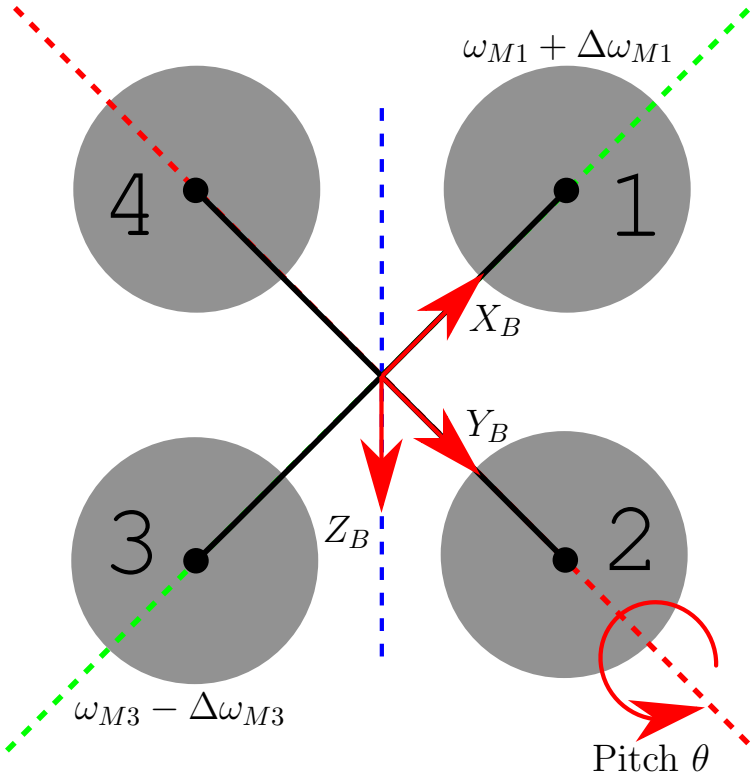



Figure 2.4: Quadrotor pitch motion.

### 2.2.3 Yaw motion

Unlike a regular helicopter, a multirotor does not use a tailrotor to cancel out the yaw torques induced by the spinning main propeller. Instead, the total yaw torques should be cancelled out by the total sum of the motors. That is why usually multirotors are found with an even number of motors and those are split up into equal numbers of clockwise and counterclockwise rotating propellers. The torque induced by the spinning propellers behaves similarly to the generated thrust, so if the sum of all clockwise motor speeds equals the sum of all counterclockwise motor speeds, the overall induced yaw torque equals zero. If a positive yaw motion is desired (positive yaw is a clockwise motion when looking from above), then the sum of counterclockwise motor speeds must be greater than the sum of clockwise motor speeds. That means motors 2 and 4 must speed up and motors 1 and 3 must slow down by amounts $\Delta\omega_{M2/4}$ and $\Delta\omega_{M1/3}$, respectively. These changes are chosen such that the overall vertical thrust remains constant and they are approximately identical for very small changes.

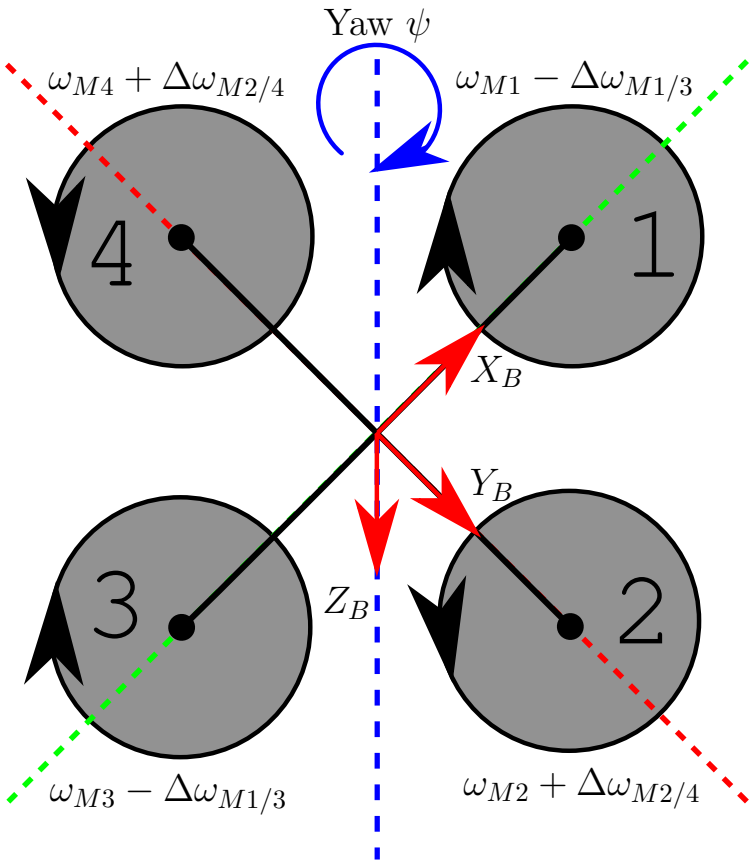



Figure 2.5: Quadrotor yaw motion.

### 2.2.4 Throttle motion

When changing the throttle command of a multirotor, all motor speeds change by the same amount $\Delta\omega_{Mi}$. If we assume the example of a perfectly symmetric and balanced quadrotor without any external influence acting on it, then in a static hover position, all motors spin at exactly the same speed $\omega_M$. The thrust generated by each motor is exactly one fourth of the weight of the airframe. When increasing all motor speeds, the quadrotor will begin to ascend. If the quadrotor was previously in a non-level state and the throttle is increased, the direction of the airframe will remain unchanged, because all motors will speed up by the same amount.

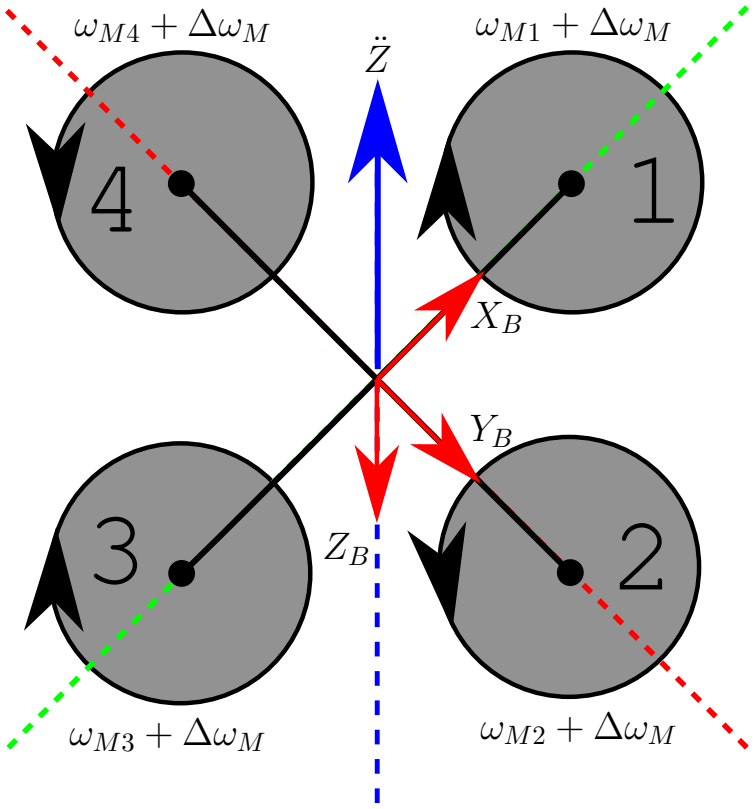



Figure 2.6: Quadrotor throttle motion.

## 2.3 Simplifying assumptions

In order to be able to derive the simplified quadrotor model, we must make a number of assumptions that simplify the equations immensely:

- The overall mass of the system is constant.
- The moment of inertia of the system is constant.
- The origin of the Body-fixed reference frame coincides with the center of gravity of the airframe.
- The axes of the Body-fixed reference frame coincide with the body principal axes of inertia. Then the $(3x3)$ inertia matrix collapses to a strictly diagonal form corresponding to the three main axes of symmetry, meaning $\boldsymbol{J} = \begin{bmatrix} J_{xx} & 0 & 0 \\ 0 & J_{yy} & 0 \\ 0 & 0 & J_{zz} \end{bmatrix}$
- Each motor is perfectly aligned along the Body-fixed $z$-axis and its thrust vector originates in the plane of the center of mass. That means the center of gravity of the airframe is assumed to be in the plane

formed by the propellers. Then the thrust vector acts only in the direction of the Body-fixed $z$-axis and it is perpendicular to the lever arm from center of gravity to the motor. Therefore, the torque generated by each motor around the center of gravity is exactly equal to $\tau_{Motor} = F_{thrust} \cdot l_{arm}$.

## 2.4 Kinematics

In order to describe the position and attitude of the Body-fixed frame with respect to the Inertial frame, we define the position and attitude vectors respectively:

$$\boldsymbol{P}_I = \begin{bmatrix} x & y & z \end{bmatrix}^T \tag{2.1}$$

$$\boldsymbol{\Theta}_I = \begin{bmatrix} \phi & \theta & \psi \end{bmatrix}^T \tag{2.2}$$

where $\boldsymbol{P}_I$ is the linear position and contains the coordinates of the Body-Fixed frame with respect to the axes of the Inertial frame, and $\boldsymbol{\Theta}_I$ is the angular position (or attitude) of the Body-Fixed frame with respect to the Inertial Frame. The angles in $\boldsymbol{\Theta_I}$ are commonly referred to as *Euler angles* in aeronautics, and $\phi$, $\theta$, and $\psi$ are known respectively as *roll*, *pitch*, and *yaw*. We have previously discussed the common definition of the Body-fixed frame as the forward direction being along the positive Body-fixed $x$-axis, the positive Body-fixed $y$-axis pointing to the right, and the Body-fixed $z$-axis pointing downward. Then the attitude configuration $\boldsymbol{\Theta}_I = \begin{bmatrix} \phi & \theta & \psi \end{bmatrix}^T = \begin{bmatrix} 0 & 0 & 0 \end{bmatrix}^T$ describes the airframe as flat and level and pointing forward along the Inertial $x$-axis. For the simplified model, the well-known Euler angle representation for attitude was chosen due to its intuitive and simple nature. However, this representation is not without problems, suffering from singularities in certain configurations. Therefore, a more robust representation using quaternions will be chosen for the model used in Chapter 3. We now define the rotation matrix $\boldsymbol{R}_I^B$ that transforms a vector expressed in the Inertial frame into a vector expressed in the Body-fixed frame.

$$\boldsymbol{z}_B = \boldsymbol{R}_I^B \cdot \boldsymbol{z}_I \tag{2.3}$$

The orientation of one Cartesian coordinate system with respect to another can always be described by three consecutive rotations. Common practice in aeronautics is to describe the Body orientation by the Euler angle sequence (1,2,3) such that the three consecutive rotation angles are the Euler angles as defined above. Starting from the Inertial reference frame, the sequence of rotations is defined as:

1. Right-handed rotation about the $z$-axis (positive yaw $\psi$), denoted by rotation matrix $\boldsymbol{R}(\psi, z)$

2. Right-handed rotation about the new $y$-axis (positive pitch $\theta$), denoted by rotation matrix $\boldsymbol{R}(\theta, y)$

3. Right-handed rotation about the new $x$-axis (positive roll $\phi$), denoted by rotation matrix $\boldsymbol{R}(\phi, x)$

This means:

$$\begin{aligned}
\boldsymbol{z}_B &= \boldsymbol{R}_I^B \cdot \boldsymbol{z}_I \\
&= \boldsymbol{R}(\phi, x) \cdot \boldsymbol{R}(\theta, y) \cdot \boldsymbol{R}(\psi, z) \cdot \boldsymbol{z}_I \\
&= \begin{bmatrix} 1 & 0 & 0 \\ 0 & c\phi & s\phi \\ 0 & -s\phi & c\phi \end{bmatrix} \cdot \begin{bmatrix} c\theta & 0 & -s\theta \\ 0 & 1 & 0 \\ s\theta & 0 & c\theta \end{bmatrix} \cdot \begin{bmatrix} c\psi & s\psi & 0 \\ -s\psi & c\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \boldsymbol{z}_I \\
&= \begin{bmatrix} c\theta c\psi & c\theta s\psi & -s\theta \\ -c\phi\, s\psi + s\phi\, s\theta\, c\psi & c\phi\, c\psi + s\phi\, s\theta\, s\psi & s\phi\, c\theta \\ s\phi\, s\psi + c\phi\, s\theta\, c\psi & -s\phi\, c\psi + c\phi\, s\theta\, s\psi & c\phi\, c\theta \end{bmatrix} \cdot \boldsymbol{z}_I
\end{aligned} \tag{2.4}$$

with the following notation: $sx = \sin(x)$, $cx = \cos(x)$, and $tx = \tan(x)$. When describing a rotating reference frame using a set of time-varying Euler angles, the components of the angular velocity vector in the Body-fixed frame can be determined as follows:

$$\begin{aligned}
\boldsymbol{\omega}_B &= \boldsymbol{H}_I^B \cdot \dot{\boldsymbol{\Theta}}_I \\
&= \begin{bmatrix} \dot{\phi} \\ 0 \\ 0 \end{bmatrix} + \boldsymbol{R}(\phi, x) \cdot \left( \begin{bmatrix} 0 \\ \dot{\theta} \\ 0 \end{bmatrix} + \boldsymbol{R}(\theta, y) \cdot \begin{bmatrix} 0 \\ 0 \\ \dot{\psi} \end{bmatrix} \right) \\
&= \begin{bmatrix} \dot{\phi} \\ 0 \\ 0 \end{bmatrix} + \boldsymbol{R}(\phi, x) \cdot \begin{bmatrix} 0 \\ \dot{\theta} \\ 0 \end{bmatrix} + \boldsymbol{R}(\phi, x) \cdot \boldsymbol{R}(\theta, y) \cdot \begin{bmatrix} 0 \\ 0 \\ \dot{\psi} \end{bmatrix} \\
\begin{bmatrix} p \\ q \\ r \end{bmatrix} &= \begin{bmatrix} 1 & 0 & -s\theta \\ 0 & c\phi & s\phi\, c\theta \\ 0 & -s\phi & c\phi\, c\theta \end{bmatrix} \cdot \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix}
\end{aligned} \tag{2.5}$$

where $\boldsymbol{\omega}_B = \begin{bmatrix} p & q & r \end{bmatrix}^T$ is the angular velocity in Body-fixed frame. It follows:

$$\begin{aligned} \dot{\boldsymbol{\Theta}}_I &= \boldsymbol{H}_B^I \cdot \boldsymbol{\omega}_B \\ &= \left(\boldsymbol{H}_I^B\right)^{-1} \cdot \boldsymbol{\omega}_B = \left(\boldsymbol{H}_I^B\right)^T \cdot \boldsymbol{\omega}_B \\ \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} &= \begin{bmatrix} 1 & t\theta\, s\phi & t\theta\, c\phi \\ 0 & c\phi & -s\phi \\ 0 & s\phi/c\theta & c\phi/c\theta \end{bmatrix} \cdot \begin{bmatrix} p \\ q \\ r \end{bmatrix} \end{aligned} \tag{2.6}$$

See [4] and [19] for additional details.

## 2.5 Dynamics

We remember that all previous assumptions listed in Section 2.3 still hold. Then we can formulate the standard Newton-Euler equations. For the translational component we have:

$$\begin{aligned} \boldsymbol{F}_I &= m \cdot \ddot{\boldsymbol{P}}_I \\ \boldsymbol{R}_B^I \cdot \boldsymbol{F}_B &= m \cdot \frac{d}{dt}\left(\dot{\boldsymbol{P}}_I\right) \\ \boldsymbol{R}_B^I \cdot \boldsymbol{F}_B &= m \cdot \frac{d}{dt}\left(\boldsymbol{R}_B^I \cdot \boldsymbol{V}_B\right) \\ \boldsymbol{R}_B^I \cdot \boldsymbol{F}_B &= m \cdot \left(\boldsymbol{R}_B^I \cdot \dot{\boldsymbol{V}}_B + \dot{\boldsymbol{R}}_B^I \cdot \boldsymbol{V}_B\right) \\ \boldsymbol{R}_B^I \cdot \boldsymbol{F}_B &= m \cdot \boldsymbol{R}_B^I \cdot \left(\dot{\boldsymbol{V}}_B + \boldsymbol{\omega}_B \times \boldsymbol{V}_B\right) \\ \boldsymbol{F}_B &= m \cdot \left(\dot{\boldsymbol{V}}_B + \boldsymbol{\omega}_B \times \boldsymbol{V}_B\right) \\ \boldsymbol{F}_B &= m \cdot \dot{\boldsymbol{V}}_B + \boldsymbol{\omega}_B \times \left(m \cdot \boldsymbol{V}_B\right) \end{aligned} \tag{2.7}$$

where $m$ is the total mass of the aircraft and $\boldsymbol{V}_B = \begin{bmatrix} u & v & w \end{bmatrix}^T$ is the velocity vector in the Body-fixed frame.

Similarly, for the rotational component:

$$\begin{aligned}
\boldsymbol{\tau}_I &= \boldsymbol{J} \cdot \ddot{\boldsymbol{\Theta}}_I \\
\boldsymbol{H}_B^I \cdot \boldsymbol{\tau}_B &= \boldsymbol{J} \cdot \frac{d}{dt}\left(\dot{\boldsymbol{\Theta}}_I\right) \\
\boldsymbol{H}_B^I \cdot \boldsymbol{\tau}_B &= \boldsymbol{J} \cdot \frac{d}{dt}\left(\boldsymbol{H}_B^I \cdot \boldsymbol{\omega}_B\right) \\
\boldsymbol{H}_B^I \cdot \boldsymbol{\tau}_B &= \boldsymbol{J} \cdot \left(\boldsymbol{H}_B^I \cdot \dot{\boldsymbol{\omega}}_B + \dot{\boldsymbol{H}}_B^I \cdot \boldsymbol{\omega}_B\right) \\
\boldsymbol{H}_B^I \cdot \boldsymbol{\tau}_B &= \boldsymbol{J} \cdot \boldsymbol{H}_B^I \cdot (\dot{\boldsymbol{\omega}}_B + \boldsymbol{\omega}_B \times \boldsymbol{\omega}_B) \\
\boldsymbol{\tau}_B &= \boldsymbol{J} \cdot (\dot{\boldsymbol{\omega}}_B + \boldsymbol{\omega}_B \times \boldsymbol{\omega}_B) \\
\boldsymbol{\tau}_B &= \boldsymbol{J} \cdot \dot{\boldsymbol{\omega}}_B + \boldsymbol{\omega}_B \times (\boldsymbol{J} \cdot \boldsymbol{\omega}_B)
\end{aligned} \tag{2.8}$$

where $\boldsymbol{J}$ is the moment of inertia matrix of the aircraft.

## 2.6 Forces and Moments

We recall the translational and rotational model dynamics from Chapter 2.5:

$$\begin{aligned}
\boldsymbol{F}_B &= m \cdot \dot{\boldsymbol{V}}_B + \boldsymbol{\omega}_B \times (m \cdot \boldsymbol{V}_B) \\
\boldsymbol{\tau}_B &= \boldsymbol{J} \cdot \dot{\boldsymbol{\omega}}_B + \boldsymbol{\omega}_B \times (\boldsymbol{J} \cdot \boldsymbol{\omega}_B)
\end{aligned} \tag{2.9}$$

Both the forces and torques acting on the system are three-dimensional vectors in the Body-fixed frame: $\boldsymbol{F}_G = \begin{bmatrix} F_x & F_y & F_z \end{bmatrix}^T$ and $\boldsymbol{\tau}_G = \begin{bmatrix} \tau_x & \tau_y & \tau_z \end{bmatrix}^T$, respectively.
A critical part in modeling is the question what forces and moments will be considered to be acting on the model system. For the simplified multirotor model, we do not consider any environmental effects. Rather, only the sources described in the next subsections are taken into account:

### 2.6.1 Gravity

The acceleration of gravity is always acting on the center of gravity of the airframe. This is a force that is acting strictly on the translational dynamics. The force of gravity in the Inertial frame is a constant vector:

$$\boldsymbol{F}_{I_G} = \begin{bmatrix} 0 & 0 & m \cdot g \end{bmatrix}^T \tag{2.10}$$

We can then use the previously defined rotation matrix $\boldsymbol{R}_I^B$ to transform this vector into the Body-fixed frame:

$$\begin{aligned}
\boldsymbol{F}_{B_G} &= \boldsymbol{R}_I^B \cdot \boldsymbol{F}_{I_G} \\
\boldsymbol{F}_{B_G} &= \boldsymbol{R}_I^B \cdot \begin{bmatrix} 0 & 0 & m \cdot g \end{bmatrix}^T \\
\boldsymbol{F}_{B_G} &= \begin{bmatrix} -m \cdot g \cdot s\theta \\ m \cdot g \cdot s\phi \cdot c\theta \\ m \cdot g \cdot c\phi \cdot c\theta \end{bmatrix}
\end{aligned} \tag{2.11}$$

### 2.6.2 Propeller thrust

The main forces and moments acting on the airframe are of course generated by the motors themselves:

- All motors combined generate an overall thrust vector that acts on the airframe center of gravity as a force.
- Each motor acts on a lever arm around the airframe center of gravity to generate a torque vector due to the motor thrust.
- Each motor induces a torque into the airframe that is due to aerodynamic drag that the spinning propeller experiences.

Since a number of simplifying assumptions were made in Chapter 2.3, these motor forces and moments simplify as follows:

- The overall thrust vector for all motors acts strictly in the negative direction of the Body-fixed $z$-axis.
- Each motor's thrust vector acts on a lever arm of length $l$ around the airframe center of gravity.
- Since all the motor axes are perfectly aligned with the Body-fixed $z$-axis, the induced torque from aerodynamic drag acts strictly around the airframe yaw axis.

So the force generated by thrust is:

$$\boldsymbol{F}_{B_T} = \begin{bmatrix} 0 & 0 & -\sum_{i=1}^{n} F_{thrust_i} \end{bmatrix}^T \tag{2.12}$$

where $F_{thrust_i}$ is the amount of thrust generated by motor $i$.

The torques generated by the motors are:

$$\begin{aligned}
\boldsymbol{\tau}_{B_T} &= \begin{bmatrix} \left(\sum_{i=1}^{n} \boldsymbol{r}_{Motor_i} \times \boldsymbol{F}_{thrust_i}\right)_{roll} \\ \left(\sum_{i=1}^{n} \boldsymbol{r}_{Motor_i} \times \boldsymbol{F}_{thrust_i}\right)_{pitch} \\ \sum_{i=1}^{n} \tau_{drag_i} \end{bmatrix} \\
\boldsymbol{\tau}_{B_T} &= \begin{bmatrix} \left(\sum_{i=1}^{n} l \cdot F_{thrust_i}\right)_{roll} \\ \left(\sum_{i=1}^{n} l \cdot F_{thrust_i}\right)_{pitch} \\ \sum_{i=1}^{n} \tau_{drag_i} \end{bmatrix}
\end{aligned} \tag{2.13}$$

where $\boldsymbol{r}_{Motor_i}$ and $\boldsymbol{F}_{thrust_i}$ are the lever arm and thrust vectors for Motor $i$, both in Body-fixed frame. Due to the simplifying assumptions discussed above, this cross product collapses to just the dot product of thrust and lever arm magnitudes, $F_{thrust_i}$ and $l$. The amounts of torque introduced into the roll and pitch channel depend on the location of each motor with respect to the Body-fixed reference frame. For example, a motor located strictly along the Body-fixed $x$-axis will contribute all of its torque around the airframe pitch axis. Similarly, a motor located strictly along the Body-fixed $y$-axis would contribute all of its torque around the airframe roll axis. Motors in other locations would contribute to both roll and pitch channel torques, subject to simple rules of geometry. $\tau_{drag_i}$ is the torque induced into the airframe by each motor due to aerodynamic drag.

### 2.6.3 Gyroscopic effects

Spinning propellers on an airframe that is itself experiencing an angular velocity generate a torque due to the gyroscopic effect. This torque is the result of the vector product of the airframe angular velocities and each motor's/propeller's angular velocity vector, both in Body-fixed frame. The resulting torque vector will be perpendicular to both of them. When the airframe is at rest, the gyroscopic effect is zero, equally the effect will cancel out when an equal number of clockwise and counterclockwise rotating propellers are spinning at the same motor speeds.

In this simplified model, the motor axes are assumed to be perfectly aligned with the Body-fixed $z$-axis. So the angular velocity vectors in the Body-fixed frame for each motor either point in the positive or negative $z$ direction exclusively: $\begin{bmatrix} 0 & 0 & \omega_{M_i} \end{bmatrix}^T$ for clockwise spinning propellers or $\begin{bmatrix} 0 & 0 & -\omega_{M_i} \end{bmatrix}^T$ for counterclockwise spinning propellers with rotor speed $\omega_{M_i}$ for Motor $i$.

Then the resulting torque can be computed as:

$$\begin{aligned} \boldsymbol{\tau}_{B_{Gyro}} &= \sum_{i=1}^{n} J_{MP} \cdot \left(\boldsymbol{\omega}_B \times \boldsymbol{\omega}_{B_{M_i}}\right) \\ \boldsymbol{\tau}_{B_{Gyro}} &= \sum_{i=1}^{n} J_{MP} \cdot \left( \begin{bmatrix} p \\ q \\ r \end{bmatrix} \times \begin{bmatrix} 0 \\ 0 \\ \pm\omega_{M_i} \end{bmatrix} \right) \end{aligned} \tag{2.14}$$

where $n$ is the total number of motors, $J_{MP}$ is the total moment of inertia of a motor and propeller around their rotation axis, and the motor speed $\omega_{M_i}$ is either positive or negative depending on the direction of rotation, as discussed above.

## 2.7 Summary of Model Parameters

A number of model specific parameters are necessary to properly simulate the behavior of a multirotor. While the assumptions made in Chapter 2.3 get rid of a lot of complexity, several model parameters are still needed. They are discussed in detail in this chapter.

### 2.7.1 Mass

The total mass of a multirotor can easily be found by weighing the physical airframe in the exact configuration the user is trying to simulate. That gives an accurate measurement of the all-up weight (AUW) of the aircraft.

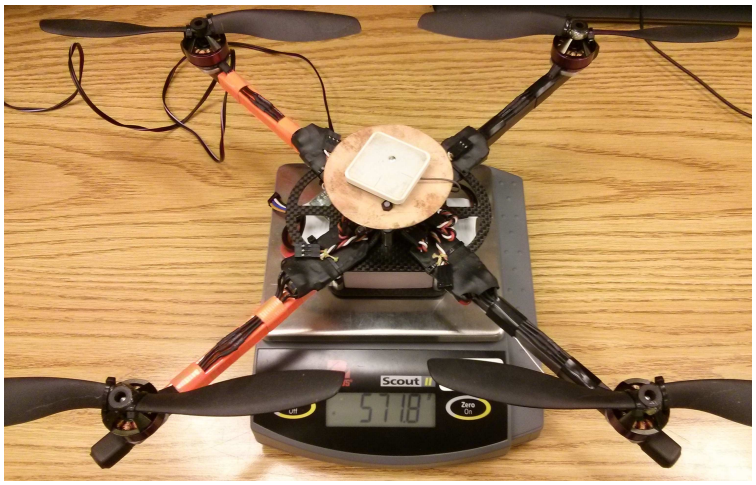



Figure 2.7: Multirotor mass. A regular laboratory scale can be used to determine the all-up weight (AUW) of the airframe to be modeled.

### 2.7.2 Moment of Inertia

There are several ways to determine the moment of inertia for a given airframe, but in most cases it is probably worthwhile to model the entire multirotor as a three-dimensional computer aided design (CAD) model. It does require the user to model each part of the airframe with correct dimensions and mass, so it is an initally time-consuming task. However, the finished CAD model allows the user to quickly and easily display the modeled airframe's moment of inertia matrix as well as its mass and the exact location of the center of gravity. All modern CAD software suites have the functionality to instantly display these parameters. Figure 2.8 shows a screenshot of an experimental multirotor airframe that was modeled in Autodesk Inventor.

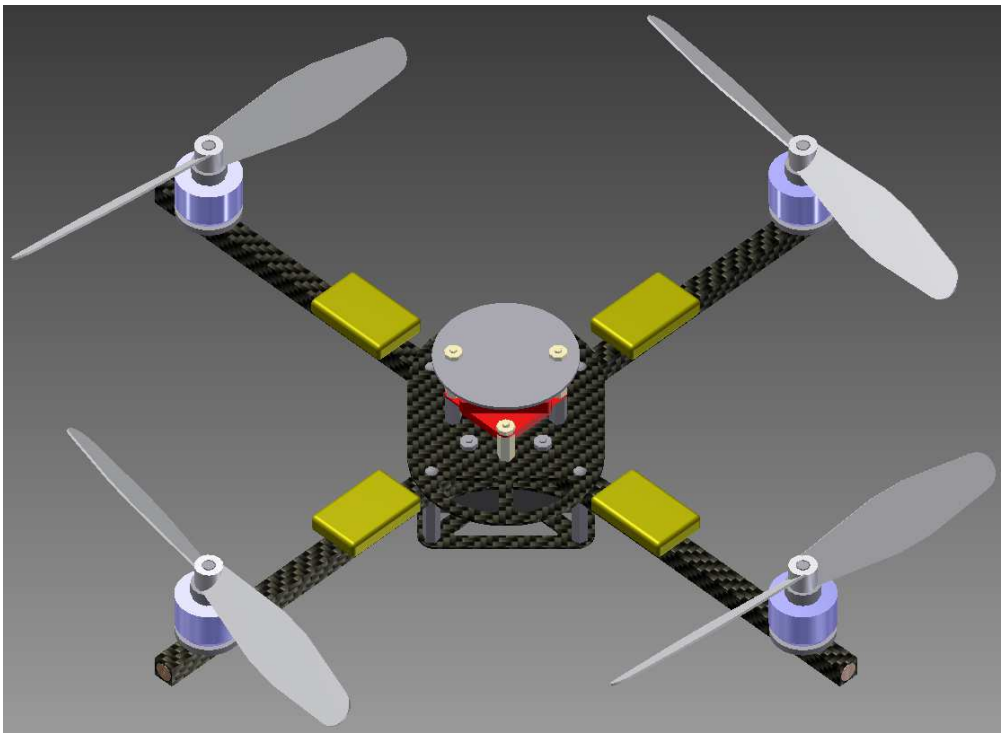


Figure 2.8: Quadrotor 3D CAD model. A fully detailed three-dimensional computer-aided design (CAD) model allows the computation of several physical model parameters.

### 2.7.3 Lever arm

Since the assumptions from Chapter 2.3 hold for the simplified model, it is easy to measure the length of the lever arm the multirotor motors are attached to. Since it is assumed that the airframe center of gravity is located in the plane of the propellers, the lever arm length is simply the horizontal distance from the motor axis to the geometric center of the airframe. Figure 2.9 shows an example for a simple symmetric quadrotor airframe.

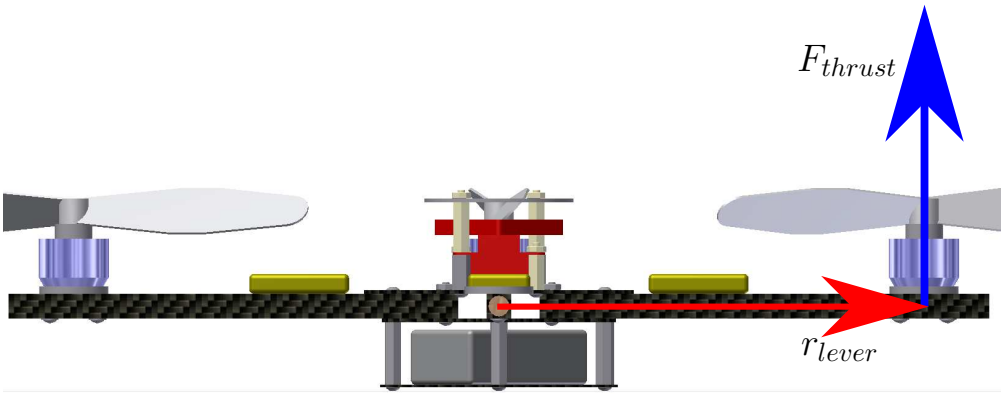
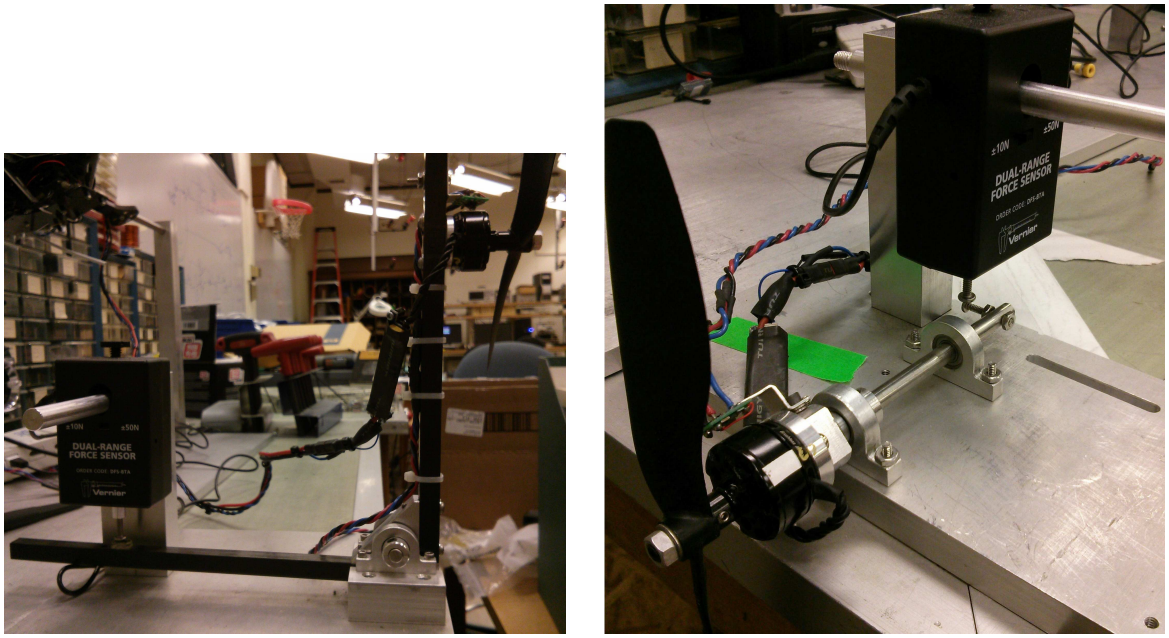



Figure 2.9: Multirotor motor lever arm. For this simple quadrotor airframe, the motor lever arm length is the distance from motor axis to the geometric center of the airframe.

### 2.7.4 Propeller Aerodynamic Coefficients

In order to determine the aerodynamic parameters of a specific motor/propeller combination, a dedicated test setup has to be constructed. Figure 2.10 shows two test rigs that were designed by the author and have been used in the UIUC College of Engineering Mechatronics Laboratory to measure the thrust and induced torque produced by a given motor/propeller combination. Motor angular velocities are given as command inputs, and the thrust and torque generated are measured by a load cell and recorded digitally.

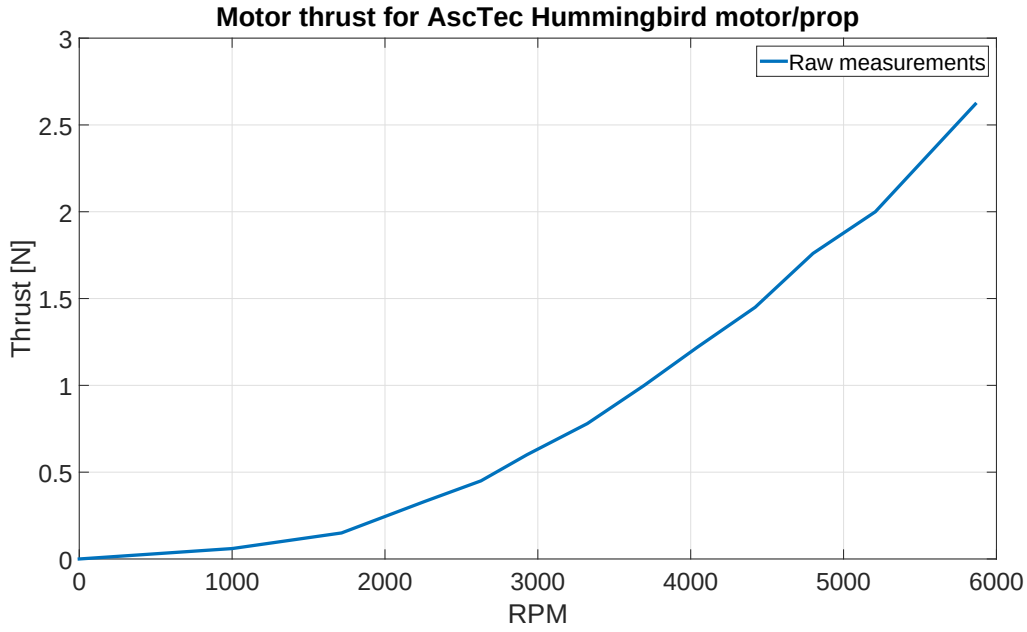



(a) Motor thrust test rig

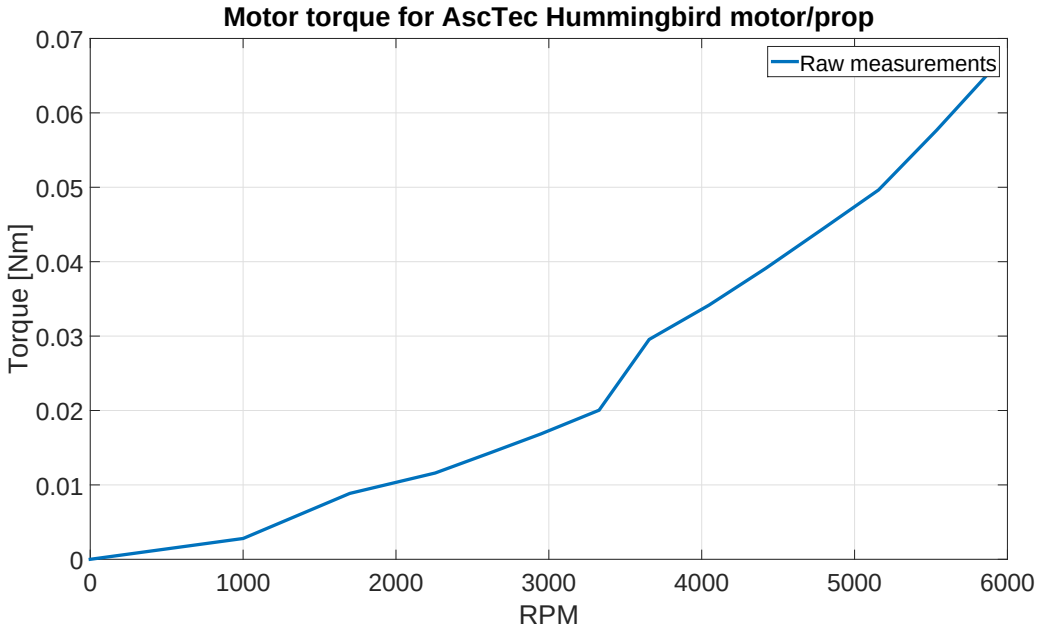
(b) Motor torque test rig

Figure 2.10: Test setup to determine motor/propeller performance data. These two test rigs allow the measurement of motor thrust as well as induced motor torque for a given motor angular velocity.

The motor is driven through its entire range of angular velocities and data is collected at various angular velocities. The density of this data point grid is up to the user; it should be rich enough to allow proper curve fitting for the determination of a thrust and drag functions. A representative result of this testing is shown in Figure 2.11.




(a) Motor thrust test




(b) Motor torque test

Figure 2.11: Test data for motor thrust and torque. Examples of the data collected when stepping a motor/propeller combination through its operating range and recording thrust and torque.

Then a curve fitting can be conducted with the help of *MATLAB/Simulink*. See Figure 2.12 for the fitted thrust data and Figure 2.13 for the fitted torque data. It can be seen that in both cases, the data fits a square function sufficiently well.

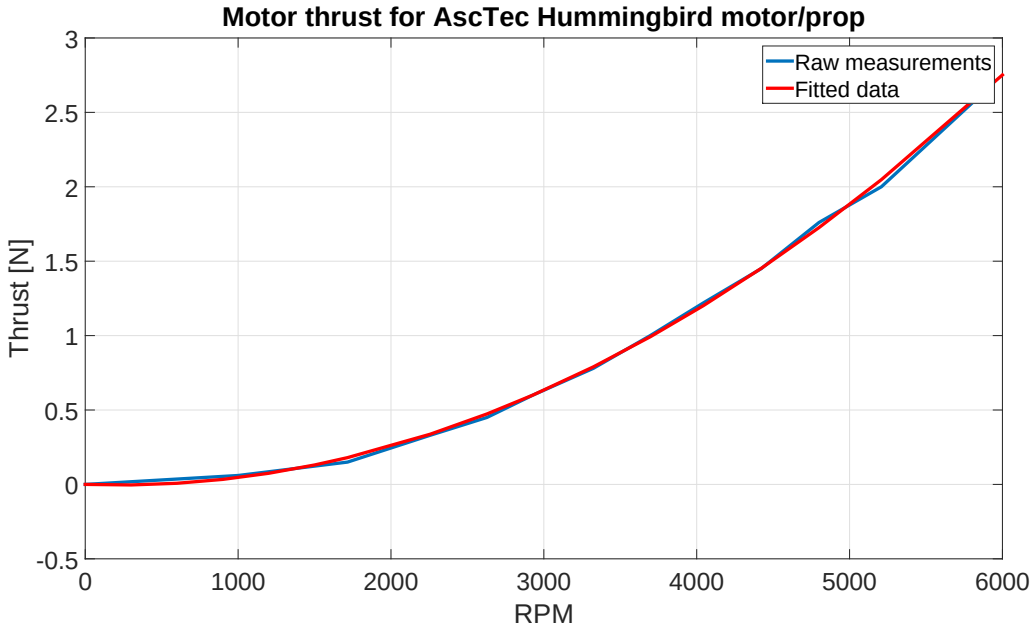



Figure 2.12: Fitted motor/propeller thrust data. Curve fitting the test data shows that motor thrust can be expressed as a square function of motor angular velocity.

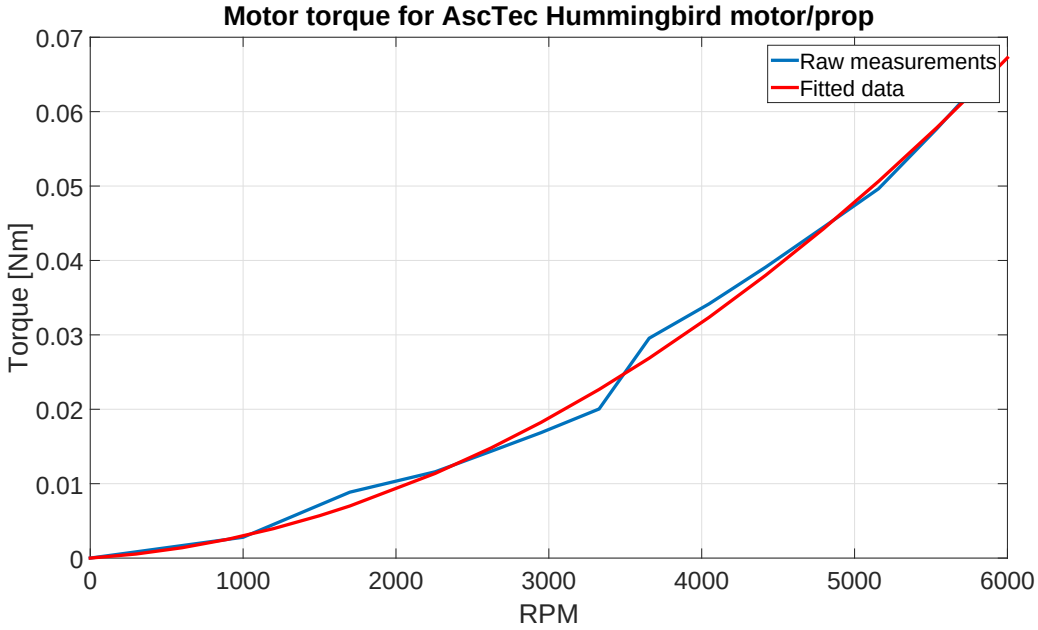



Figure 2.13: Fitted motor/propeller torque data. Curve fitting the test data shows that induced motor torque can be expressed as a square function of motor angular velocity.

The expression for motor thrust as a function of angular velocity can then be formulated as:

$$F_{Motor} = k_T \cdot \omega_{Motor}^2 \tag{2.15}$$

where $k_T$ is the propeller aerodynamic thrust coefficient.
Similarly, the expression for the induced motor torque as a function of angular velocity can be written as:

$$\tau_{Motor} = k_D \cdot \omega_{Motor}^2 \tag{2.16}$$

where $k_D$ is the propeller aerodynamic drag coefficient. All above values are in proper SI units.

## 2.8 Quadrotor Example

The model principles laid out in this chapter can now be applied to a specific airframe example. A standard quadrotor airframe is considered. All simplifying assumptions mentioned in Chapter 2.3 still hold, and we define:

- The airframe's all-up weight (AUW) $m$ is known.
- The airframe's moment of inertia matrix $J$ is known and it is strictly diagonal.
- The lever arm $l$ from center of gravity to each motor is known.
- The propeller aerodynamic thrust and drag coefficients $k_T$ and $k_D$ are known.

The quadrotor is chosen to fly in '+ Mode', so its arms are aligned with the Body-fixed reference frame. See Figure 2.14 for a diagram.

### 2.8.1 Dynamics

Recall the dynamics derived in Chapter 2.5, Equations 2.7 and 2.8:

$$\begin{aligned} \boldsymbol{F}_B &= m \cdot \dot{\boldsymbol{V}}_B + \boldsymbol{\omega}_B \times (m \cdot \boldsymbol{V}_B) \\ \boldsymbol{\tau}_B &= \boldsymbol{J} \cdot \dot{\boldsymbol{\omega}}_B + \boldsymbol{\omega}_B \times (\boldsymbol{J} \cdot \boldsymbol{\omega}_B) \end{aligned} \tag{2.17}$$

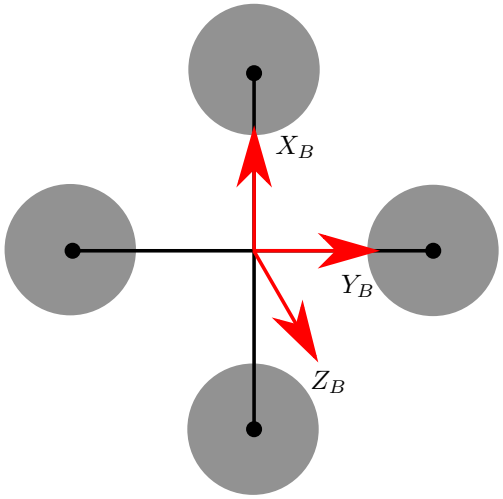



Figure 2.14: Quadrotor airframe example. The symmetric quadrotor airframe flying in '+ Mode' is chosen as an example for the full derivation of the simplified multirotor model.

We will solve for $\dot{\boldsymbol{V}}_B$ and $\dot{\boldsymbol{\omega}}_B$:

$$\begin{aligned} \dot{\boldsymbol{V}}_B &= \frac{\boldsymbol{F}_B}{m} - \boldsymbol{\omega}_B \times \boldsymbol{V}_B \\ \dot{\boldsymbol{\omega}}_B &= \boldsymbol{J}^{-1} \cdot \boldsymbol{\tau}_B - \boldsymbol{J}^{-1} \cdot \boldsymbol{\omega}_B \times (\boldsymbol{J} \cdot \boldsymbol{\omega}_B) \end{aligned} \tag{2.18}$$

We expand:

$$\begin{aligned} \begin{bmatrix} \dot{u} \\ \dot{v} \\ \dot{w} \end{bmatrix} &= \frac{1}{m} \cdot \boldsymbol{F}_B - \begin{bmatrix} p \\ q \\ r \end{bmatrix} \times \begin{bmatrix} u \\ v \\ w \end{bmatrix} \\ \begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} &= \begin{bmatrix} \frac{1}{J_{xx}} & 0 & 0 \\ 0 & \frac{1}{J_{yy}} & 0 \\ 0 & 0 & \frac{1}{J_{zz}} \end{bmatrix} \cdot \boldsymbol{\tau}_B - \begin{bmatrix} \frac{1}{J_{xx}} & 0 & 0 \\ 0 & \frac{1}{J_{yy}} & 0 \\ 0 & 0 & \frac{1}{J_{zz}} \end{bmatrix} \cdot \begin{bmatrix} p \\ q \\ r \end{bmatrix} \times \left( \begin{bmatrix} J_{xx} & 0 & 0 \\ 0 & J_{yy} & 0 \\ 0 & 0 & J_{zz} \end{bmatrix} \cdot \begin{bmatrix} p \\ q \\ r \end{bmatrix} \right) \end{aligned} \tag{2.19}$$

### 2.8.2 Forces and Moments

The force $\boldsymbol{F}_B$ consists of the force of gravity and the overall thrust generated by all motors. It was already derived previously. See Equation 2.11 for the gravity component:

$$\boldsymbol{F}_{B_G} = \begin{bmatrix} -m \cdot g \cdot s\theta \\ m \cdot g \cdot s\phi \cdot c\theta \\ m \cdot g \cdot c\phi \cdot c\theta \end{bmatrix} \tag{2.20}$$

For the specific quadrotor example, the thrust component of the force previously defined in Equation 2.12 is

$$\boldsymbol{F}_{B_T} = \begin{bmatrix} 0 \\ 0 \\ -k_T \cdot \left(\omega_{M_1}^2 + \omega_{M_2}^2 + \omega_{M_3}^2 + \omega_{M_4}^2\right) \end{bmatrix} \tag{2.21}$$

The torque $\boldsymbol{\tau}_B$ consists of three components: the torque due to gyroscopic effects, the portion of torque generated by the motor thrust in the roll and pitch channels, and then the torque generated by the induced torque due to aerodynamic drag of the propellers.

When reviewing the structure of the quadrotor airframe in Figure 2.14, we can see that for the '+ Mode', motors 2 and 4 generate a torque strictly in the roll channel, while motors 1 and 3 generate torque only in the pitch channel. Since the motor axes are perfectly aligned with the Boxy-fixed $z$-axis, the induced torque due to aerodynamic drag acts only in the yaw channel. Then the torque component due to motor thrust and propeller aerodynamic drag can be written as:

$$\boldsymbol{\tau}_{B_T} = \begin{bmatrix} l \cdot k_T \cdot \left(\omega_{M_4}^2 - \omega_{M_2}^2\right) \\ l \cdot k_T \cdot \left(\omega_{M_1}^2 - \omega_{M_3}^2\right) \\ k_D \cdot \left(\omega_{M_2}^2 + \omega_{M_4}^2 - \omega_{M_1}^2 - \omega_{M_3}^2\right) \end{bmatrix} \tag{2.22}$$

The portion of the torque due to gyroscopic effects was previously derived in Figure 2.14 and for a quadrotor with propeller rotations as shown in Figure 2.14, it comes out to be:

$$\begin{aligned}
\boldsymbol{\tau}_{B_{Gyro}} &= \sum_{i=1}^{n} J_{MP} \cdot \left(\boldsymbol{\omega}_B \times \boldsymbol{\omega}_{B_{M_i}}\right) = \sum_{i=1}^{n} J_{MP} \cdot \left( \begin{bmatrix} p \\ q \\ r \end{bmatrix} \times \begin{bmatrix} 0 \\ 0 \\ \pm\omega_{M_i} \end{bmatrix} \right) \\
\boldsymbol{\tau}_{B_{Gyro}} &= J_{MP} \cdot \left( \begin{bmatrix} p \\ q \\ r \end{bmatrix} \times \begin{bmatrix} 0 \\ 0 \\ \omega_{M_1} - \omega_{M_2} + \omega_{M_3} - \omega_{M_4} \end{bmatrix} \right) \\
\boldsymbol{\tau}_{B_{Gyro}} &= J_{MP} \cdot \begin{bmatrix} q \cdot (\omega_{M_1} - \omega_{M_2} + \omega_{M_3} - \omega_{M_4}) \\ p \cdot (-\omega_{M_1} + \omega_{M_2} - \omega_{M_3} + \omega_{M_4}) \\ 0 \end{bmatrix}
\end{aligned} \tag{2.23}$$

Entering the forces $\boldsymbol{F}_B$ and moments $\boldsymbol{\tau}_B$ into the dynamics in Equation 2.19 and simplifying, we get:

$$\begin{aligned}
\dot{u} &= (v \cdot r - w \cdot q) - g \cdot \sin\theta \\
\dot{v} &= (w \cdot p - u \cdot r) - g \cdot \cos\theta \cdot \sin\phi \\
\dot{w} &= (u \cdot q - v \cdot p) - g \cdot \cos\theta \cdot \cos\phi - \frac{1}{m} \cdot k_T \cdot \left(\omega_{M_1}^2 + \omega_{M_2}^2 + \omega_{M_3}^2 + \omega_{M_4}^2\right) \\
\dot{p} &= \frac{J_{yy} - J_{zz}}{J_{xx}} \cdot q \cdot r + \frac{J_{MP}}{J_{xx}} \cdot q \cdot (\omega_{M_1} - \omega_{M_2} + \omega_{M_3} - \omega_{M_4}) + \frac{1}{J_{xx}} \cdot l \cdot k_T \cdot \left(\omega_{M_4}^2 - \omega_{M_2}^2\right) \\
\dot{q} &= \frac{J_{zz} - J_{xx}}{J_{yy}} \cdot p \cdot r - \frac{J_{MP}}{J_{yy}} \cdot p \cdot (\omega_{M_1} - \omega_{M_2} + \omega_{M_3} - \omega_{M_4}) + \frac{1}{J_{yy}} \cdot l \cdot k_T \cdot \left(\omega_{M_1}^2 - \omega_{M_3}^2\right) \\
\dot{r} &= \frac{J_{xx} - J_{yy}}{J_{zz}} \cdot p \cdot q + \frac{1}{J_{zz}} \cdot k_D \cdot \left(\omega_{M_2}^2 + \omega_{M_4}^2 - \omega_{M_1}^2 - \omega_{M_3}^2\right)
\end{aligned} \tag{2.24}$$

This is the complete simplified dynamics model for a quadrotor in '+ Mode' as depicted in Figure 2.14. These equations make up the six degree of freedom simulation of a quadrotor. If a reader has implemented these dynamics in *MATLAB/Simulink* and wants to display positions and velocities in the Inertial frame instead of the Body-fixed frame, it is merely a matter of transforming using the previously defined rotation matrix $\boldsymbol{R}_B^I$ and integrating to get $\boldsymbol{V}_I$ and $\boldsymbol{P}_I$.

Several more particular solutions for very common airframes like hexacopter or octocopter can be found in Appendix A.

# Chapter 3

# Improved Multirotor Model

This chapter presents the full derivation of an improved multirotor model with higher fidelity than the one previously shown in Chapter 2. We build upon the basic concepts developed in the previous chapter and, in addition, consider various external influences that play a role in more precise modeling of multirotor dynamics. The full kinematic and dynamic equations for this improved model are derived and all forces and moments acting on the system are discussed in detail.
All concepts regarding coordinate systems and how a multirotor moves are still valid and have been derived previously, so the reader should consult Chapters 2.1 and 2.2 if they need to refamiliarize themselves with these ideas.

## 3.1 Simplifying assumptions

A number of simplifying assumptions had to be made in Chapter 2.3 to allow the construction of the simplified multirotor model. For the improved model, these assumptions are no longer necessary. This means:

- The overall mass of the system can be time-varying.
- The moment of inertia of the system can be time-varying.
- The origin of the Body-fixed reference frame can be located away from the center of gravity of the airframe, if so desired.
- The multirotor airframe is not required to be symmetric about any of the Body-fixed reference frame axes. Therefore, the $(3 \times 3)$ inertia matrix is not necessarily diagonal nor symmetric:

$$\boldsymbol{J} = \begin{bmatrix} J_{xx} & J_{xy} & J_{xz} \\ J_{yx} & J_{yy} & J_{yz} \\ J_{zx} & J_{zy} & J_{zz} \end{bmatrix}$$

- Motors are no longer considered perfectly aligned along the Body-fixed $z$-axis. Also, the center of gravity of the airframe can be located at any point on the airframe and does not have to lie in the geometric plane formed by the propellers. Then the thrust vector acts in some direction in the Body-fixed frame (due to motor misalignment) and it is no longer strictly perpendicular to the lever arm from the center of gravity to the motor.

## 3.2 Kinematics

Recall that Euler angles were used for the representation of attitude in Chapter 2.4. This was done for convenience because Euler angles present a very intuitive representation of attitude and many readers will already have had previous exposure to the concept. However, the Euler angle representation of attitude is not without problems, namely there are certain attitudes in which the representation reaches a singularity and does not display attitude correctly. This phenomenon is usually referred to as *gimbal lock*. See [4] and [19] for additional details.

In order to avoid any of the issues associated with the Euler attitude representation, a quaternion representation of airframe attitude is chosen instead. A quaternion is a four-element vector that uniquely describes any rotation in the three-dimensional space. It is composed of one real element and three vector elements. This vector describes an axis in the space around which a rotation should be performed, while the scalar part of the quaternion describes the amount of rotation around that axis.

$$\boldsymbol{q}_I = \begin{bmatrix} q_0 & q_1 & q_2 & q_3 \end{bmatrix}^T = \begin{bmatrix} q_0 & \boldsymbol{q} \end{bmatrix}^T \tag{3.1}$$

$\boldsymbol{q}_I$ is defined here as the unit quaternion that represents the rotation from the Inertial frame to the Body-fixed frame of the aircraft. As the name implies, the norm of the unit quaternion is equal to 1:

$$\|\boldsymbol{q}_I\| = \sqrt{q_0^2 + q_1^2 + q_2^2 + q_3^2} = 1 \tag{3.2}$$

Just as previously, we describe the position of the Body-fixed frame with respect to the Inertial frame as:

$$\boldsymbol{P}_I = \begin{bmatrix} x & y & z \end{bmatrix}^T \tag{3.3}$$

where $\boldsymbol{P}_I$ is the linear position and contains the coordinates of the Body-Fixed frame with respect to the axes of the Inertial frame. The Body-fixed frame is still commonly defined as the forward direction being

along the positive Body-fixed $x$-axis, the positive Body-fixed $y$-axis pointing to the right, and the Body-fixed $z$-axis pointing downward. Then the attitude configuration $\boldsymbol{q}_I = \begin{bmatrix} q_0 & q_1 & q_2 & q_3 \end{bmatrix}^T = \begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix}^T$ describes the airframe as flat and level and pointing forward along the Inertial $x$-axis. To rotate a vector from the Inertial frame to the Body-fixed frame, we previously defined a rotation matrix $\boldsymbol{R}_I^B$, see Equation 2.4. However, this rotation matrix was based on the Euler angle attitude representation. Similarly, a quaternion-based definition exists:

$$\begin{aligned} \boldsymbol{z}_B &= \boldsymbol{R}_I^B \cdot \boldsymbol{z}_I \\ \boldsymbol{z}_B &= \begin{bmatrix} q_0^2 + q_1^2 - q_2^2 - q_3^2 & 2q_1q_2 + 2q_0q_3 & 2q_1q_3 - 2q_0q_2 \\ 2q_1q_2 - 2q_0q_3 & q_0^2 - q_1^2 + q_2^2 - q_3^2 & 2q_2q_3 + 2q_0q_1 \\ 2q_1q_3 + 2q_0q_2 & 2q_2q_3 - 2q_0q_1 & q_0^2 - q_1^2 - q_2^2 + q_3^2 \end{bmatrix} \cdot \boldsymbol{z}_I \end{aligned} \tag{3.4}$$

When describing the orientation of an object when the two reference frames are rotating relative to each other, the attitude quaternion has to be updated continuously. The equations describing this relationship will be differential equations for the quaternion, where the coefficients are determined from the angular rates of the rotation. They take the following form, very similar in structure to the Euler kinematical equations previously presented in Equation 2.5:

$$\dot{\boldsymbol{q}} = \begin{bmatrix} \dot{q}_0 \\ \dot{q}_1 \\ \dot{q}_2 \\ \dot{q}_3 \end{bmatrix} = \frac{1}{2} \cdot \begin{bmatrix} 0 & -p & -q & -r \\ p & 0 & r & -q \\ q & -r & 0 & p \\ r & q & -p & 0 \end{bmatrix} \cdot \begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \end{bmatrix} = \frac{1}{2} \cdot \begin{bmatrix} -q_1 & -q_2 & -q_3 \\ q_0 & -q_3 & q_2 \\ q_3 & q_0 & -q_1 \\ -q_2 & q_1 & q_0 \end{bmatrix} \cdot \begin{bmatrix} p \\ q \\ r \end{bmatrix} \tag{3.5}$$

where is the $\boldsymbol{\omega}_B = \begin{bmatrix} p & q & r \end{bmatrix}^T$ is the angular velocity in Body-fixed frame, as previously defined.
The derivation of this relationship is omitted here, but the reader can consult [4] and [19] if additional details are desired. Additionally, should a transformation from unit quaternion to Euler angles ever be necessary or desired, the following relationship holds:

$$\begin{bmatrix} \phi \\ \theta \\ \psi \end{bmatrix} = \begin{bmatrix} \text{atan2}\left(2q_2q_3 + 2q_0q_1, q_3^2 - q_2^2 - q_1^2 + q_0^2\right) \\ \text{-asin}\left(2q_1q_3 - 2q_0q_2\right) \\ \text{atan2}\left(2q_1q_2 + 2q_0q_3, q_1^2 + q_0^2 - q_3^2 - q_2^2\right) \end{bmatrix} \tag{3.6}$$

## 3.3 Dynamics

Since none of the definitions in the dynamical equations have been changed, the equations themselves remain unchanged as well. We recall the translational and rotational model dynamics from Equations 2.7 and 2.8 in Chapter 2.5:

$$\begin{aligned} \boldsymbol{F}_B &= m \cdot \dot{\boldsymbol{V}}_B + \boldsymbol{\omega}_B \times (m \cdot \boldsymbol{V}_B) \\ \boldsymbol{\tau}_B &= \boldsymbol{J} \cdot \dot{\boldsymbol{\omega}}_B + \boldsymbol{\omega}_B \times (\boldsymbol{J} \cdot \boldsymbol{\omega}_B) \end{aligned} \tag{3.7}$$

where $m$ is the total mass of the aircraft, $\boldsymbol{J}$ is the moment of inertia matrix, $\boldsymbol{V}_B = \begin{bmatrix} u & v & w \end{bmatrix}^T$ is the velocity vector in the Body-fixed frame, and $\boldsymbol{\omega}_B = \begin{bmatrix} p & q & r \end{bmatrix}^T$ is the angular velocity vector in the Body-fixed frame.

## 3.4 Forces and Moments

As mentioned above, the translational and rotational dynamical equations remain unchanged. However, for the improved multirotor model it will be investigated in more detail how forces and torques are generated and also more and different factors will be considered, for example environmental effects.

### 3.4.1 Aerodynamic drag of the airframe

One realistic environmental effect not considered in the simplified model is the aerodynamic drag. In realistic flight, the multirotor airframe experiences wind resistance or aerodynamic drag when traveling through the air. Without aerodynamic drag, a multirotor in simulation previously flying sideways would just continue flying in that direction at constant speed, even with no net force moving it that way. So the drag acts as kind of a damping force that slows this motion down to zero over some time.

A very simple wind resistance model is constructed for the airframe. As has been mentioned in Chapter 2.7.2, modeling the entire airframe in a computer-aided design (CAD) software allows the user to easily extract important model parameters. Most software suites provide a measurement of physical surface area from specified perspectives. This can be used to generate a simple surface model of the airframe. The surface area is computed along the axes of the Body-fixed reference frame and assumed to be continuous and smooth inbetween these points of measurement. Then the surface area model of the airframe resembles an ellipsoid

of form

$$\frac{x^2}{A_x^2} + \frac{y^2}{A_y^2} + \frac{z^2}{A_z^2} = 1 \tag{3.8}$$

where the semi-axis lengths $A_x$, $A_y$, $A_z$ are the surface areas when viewing the airframe from along the $x$, $y$, or $z$ axis, respectively.

If wind is considered in the simulation, the wind speed must also be used to compute the true air speed. True air speed (TAS) is the speed of the aircraft relative to the air mass in which it is flying. True airspeed in Body-fixed frame can be computed by:

$$\boldsymbol{V}_{B_{TAS}} = \boldsymbol{V}_B - \boldsymbol{V}_{B_{Wind}} \tag{3.9}$$

$\boldsymbol{V}_{B_{TAS}}$ is the velocity in Body-fixed frame with which the airframe moves through the air. Its direction can be used to determine the surface area of the airframe that is exposed to the air, and its magnitude can be used to determine the overall drag force acting on the airframe against the direction of motion. Overall drag force in the Body-fixed frame is computed by:

$$\boldsymbol{F}_{B_{Drag}} = -\frac{1}{2} \cdot \rho \cdot \boldsymbol{V}_{B_{TAS}} \cdot \|\boldsymbol{V}_{B_{TAS}}\| \cdot C_D \cdot A \tag{3.10}$$

where $\rho$ is the density of air, $C_D$ is the drag coefficient, $A$ is the surface area of the airframe exposed to the wind, and the quantity $(-\boldsymbol{V}_{B_{TAS}} \cdot \|\boldsymbol{V}_{B_{TAS}}\|)$ is the squared magnitude of true air speed pointing in the true air speed's opposite direction. For simplicity, the force $\boldsymbol{F}_{B_{Drag}}$ due to aerodynamic drag of the airframe is considered strictly as a force acting at the center of gravity. It has no torque components and therefore only acts on the translational dynamics.

### 3.4.2 Gravity

The acceleration of gravity is always acting on the center of gravity of the airframe. It is a force acting strictly on the translational dynamics. The force of gravity in the Inertial frame is a constant vector:

$$\boldsymbol{F}_{I_G} = \begin{bmatrix} 0 & 0 & m \cdot g \end{bmatrix}^T \tag{3.11}$$

We can use the previously defined rotation matrix $\boldsymbol{R}_I^B$ to transform this vector into the Body-fixed frame. Since the improved multirotor model uses a quaternion attitude representation, the rotation matrix is equally

expressed in quaternions:

$$
\begin{aligned}
\boldsymbol{F}_{B_G} &= \boldsymbol{R}_I^B \cdot \boldsymbol{F}_{I_G} \\
\boldsymbol{F}_{B_G} &= \boldsymbol{R}_I^B \cdot \begin{bmatrix} 0 & 0 & m \cdot g \end{bmatrix}^T \\
\boldsymbol{F}_{B_G} &= \begin{bmatrix} q_0^2 + q_1^2 - q_2^2 - q_3^2 & 2q_1q_2 + 2q_0q_3 & 2q_1q_3 - 2q_0q_2 \\ 2q_1q_2 - 2q_0q_3 & q_0^2 - q_1^2 + q_2^2 - q_3^2 & 2q_2q_3 + 2q_0q_1 \\ 2q_1q_3 + 2q_0q_2 & 2q_2q_3 - 2q_0q_1 & q_0^2 - q_1^2 - q_2^2 + q_3^2 \end{bmatrix} \cdot \begin{bmatrix} 0 & 0 & m \cdot g \end{bmatrix}^T \\
\boldsymbol{F}_{B_G} &= \begin{bmatrix} m \cdot g \cdot (2q_1q_3 - 2q_0q_2) \\ m \cdot g \cdot (2q_2q_3 + 2q_0q_1) \\ m \cdot g \cdot \left(q_0^2 - q_1^2 - q_2^2 + q_3^2\right) \end{bmatrix}
\end{aligned} \tag{3.12}
$$

### 3.4.3 Gyroscopic effects

The torque generated due to gyroscopic effects was discussed in Chapter 2.6.3 and occurs in the improved model as well. Just like before, it is the result of the vector product of the airframe angular velocities and each motor's/propeller's angular velocity vector, both in Body-fixed frame.

However, in the improved multirotor model, we no longer assume the motor axes to be perfectly aligned with the Body-fixed $z$-axis. So the angular velocity vectors in the Body-fixed frame for each motor point in some direction of the Body reference frame as defined by the misalignment vector of each motor:

$$
\boldsymbol{\omega}_{B_{M_i}} = \begin{bmatrix} \boldsymbol{\omega}_{B_{M_i}}^x \\ \boldsymbol{\omega}_{B_{M_i}}^y \\ \boldsymbol{\omega}_{B_{M_i}}^z \end{bmatrix} \tag{3.13}
$$

Whether the motor angular velocity vector points upward or downward along the motor axis depends on the propeller direction of rotation and is governed by the right-hand rule. The vector has magnitude $\omega_{B_{M_i}}$ which is the speed that Motor $i$ is spinning at. Then the resulting torque can be computed as:

$$
\begin{aligned}
\boldsymbol{\tau}_{B_{Gyro}} &= \sum_{i=1}^{n} J_{MP} \cdot \left(\boldsymbol{\omega}_B \times \boldsymbol{\omega}_{B_{M_i}}\right) \\
\boldsymbol{\tau}_{B_{Gyro}} &= \sum_{i=1}^{n} J_{MP} \cdot \left( \begin{bmatrix} p \\ q \\ r \end{bmatrix} \times \begin{bmatrix} \boldsymbol{\omega}_{B_{M_i}}^x \\ \boldsymbol{\omega}_{B_{M_i}}^y \\ \boldsymbol{\omega}_{B_{M_i}}^z \end{bmatrix} \right)
\end{aligned} \tag{3.14}
$$

where $n$ is the total number of motors, $J_{MP}$ is the total moment of inertia of a motor and propeller around their rotation axis, and the motor angular velocity vector $\boldsymbol{\omega}_{B_{M_i}}$ is defined as mentioned just above.

### 3.4.4 Propeller thrust

Even for the improved multirotor model, the main forces and moments acting on the airframe are generated by the motors themselves:

- All motors combined generate an overall thrust vector that acts on the airframe center of gravity as a force.
- Each motor acts on a lever arm around the airframe center of gravity to generate a torque vector due to the motor thrust.
- Each motor induces a torque into the airframe that is due to aerodynamic drag of the spinning propeller.

However, the simplifying assumptions made previously in Chapter 2.3 no longer hold for the improved model, so the simplified effects of motor thrust that were derived in Chapter 2.6.2 are not valid anymore either. This means:

- Since the motor axes are no longer perfectly aligned with the Body-fixed $z$-axis, the overall thrust vector for all motors is the sum of all the motor thrusts and no longer acts strictly in the negative direction of the Body-fixed $z$-axis.
- Each motor's thrust vector acts on a lever arm around the airframe center of gravity. Since the airframe is not required to be symmetric, the center of gravity may not lie in the geometric center of the airframe and the effective lever arm to each motor may be of a different length.
- Since the airframe center of gravity is no longer assumed to lie in the geometric plane generated by the propellers and due to the motors' misalignment, the effective lever arm and the thrust vector are not at a right angle to each other anymore.
- Since the motor axes are no longer perfectly aligned with the Body-fixed $z$-axis, the induced torque from aerodynamic drag also acts three-dimensionally in the airframe Body-fixed frame.

So the force generated by thrust is:

$$\boldsymbol{F}_{B_T} = \sum_{i=1}^{n} \boldsymbol{F}_{B_{T_i}} = \begin{bmatrix} F_{T_i}^x \\ F_{T_i}^y \\ F_{T_i}^z \end{bmatrix}_B \tag{3.15}$$

where $\boldsymbol{F}_{B_{T_i}}$ is the thrust vector of motor $i$ with proper direction and magnitude and $F_{T_i}^x$, $F_{T_i}^y$, and $F_{T_i}^z$ are its components in the Body-fixed frame.

The torques generated by the motors are:

$$\boldsymbol{\tau}_{B_T} = \sum_{i=1}^{n} \left( \begin{bmatrix} \left(\boldsymbol{r}_{Motor_i} \times \boldsymbol{F}_{B_{T_i}}\right)_{roll} \\ \left(\boldsymbol{r}_{Motor_i} \times \boldsymbol{F}_{B_{T_i}}\right)_{pitch} \\ \left(\boldsymbol{r}_{Motor_i} \times \boldsymbol{F}_{B_{T_i}}\right)_{yaw} \end{bmatrix} + \begin{bmatrix} \left(\boldsymbol{\tau}_{drag_i}\right)_{roll} \\ \left(\boldsymbol{\tau}_{drag_i}\right)_{pitch} \\ \left(\boldsymbol{\tau}_{drag_i}\right)_{yaw} \end{bmatrix} \right) \tag{3.16}$$

where $\boldsymbol{r}_{Motor_i}$ and $\boldsymbol{F}_{B_{T_i}}$ are the lever arm and thrust vectors for Motor $i$, both in Body-fixed frame. Since thrust vector and lever arm are no longer necessarily perpendicular, this cross product may yield torque components in all three Body-fixed axes. $\boldsymbol{\tau}_{drag_i}$ is the torque vector induced into the airframe by each motor due to aerodynamic propeller drag, also in Body-fixed frame.

As explained before, a motor's location in the Body-fixed frame governs the direction of the torque it generates. For example, a motor located strictly along the Body-fixed $x$-axis would contribute all of its torque around the airframe pitch axis. Similarly, a motor located strictly along the Body-fixed $y$-axis would contribute all of its torque around the airframe roll axis. Motors in other locations will contribute to both roll and pitch channel torques, subject to simple rules of geometry.

When previously looking at the thrust generated by the motors/propellers in Chapter 2.7.4, motor thrust was measured on a static test rig. However, this testing yields only a static thrust measurement, because the air surrounding the propeller is not moving. This test setup essentially only simulates the thrust generated when the multirotor is hovering in place without any wind acting on the airframe. In reality, thrust is highly dependent on variables like climb velocity and angle of attack and the thrust produced at a given motor speed can vary greatly throughout the flight envelope of the multirotor. These effects will be discussed in more detail in Chapter 3.5.7.

### 3.4.5 Blade flapping

Another noteworthy realistic effect that influences multirotor behavior is a phenomenon called blade flapping. This effect occurs when a spinning propeller moves through the air translationally, as depicted in Figure 3.1. When moving sideways, the advancing propeller blade (the blade moving in the direction of the body velocity) travels through the air at a higher effective speed, while the retreating propeller blade (moving against the direction of the body velocity) sees a lower effective speed relative to the air.

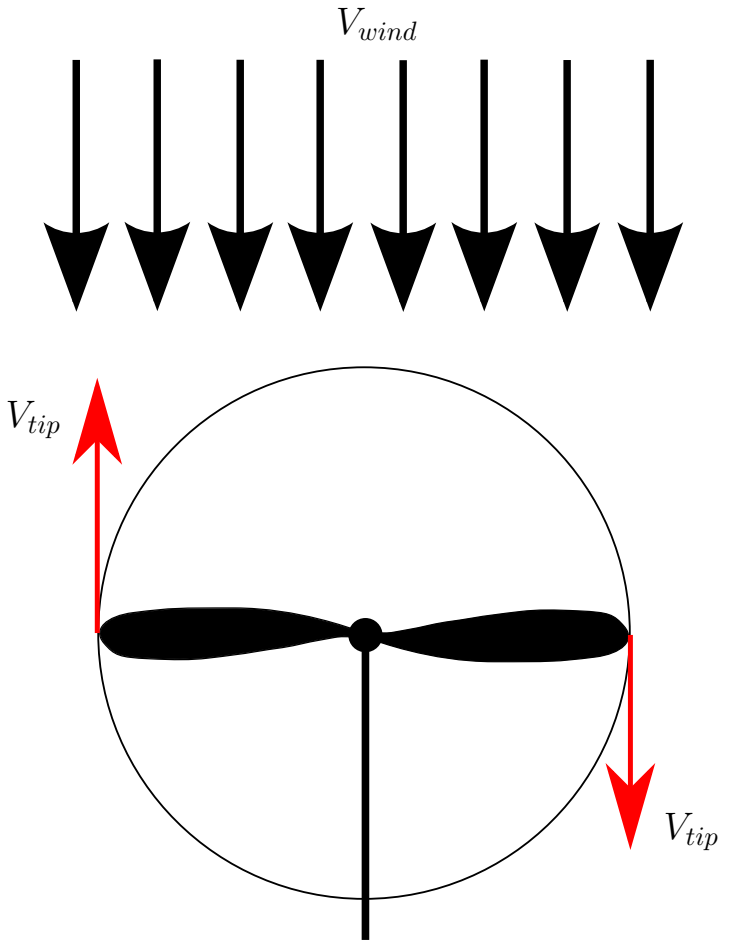



Figure 3.1: Blade flapping geometry. When a spinning propeller moves through the air translationally, the propeller blades will see different effective air speeds.

This causes a dissymmetry of lift, because the advancing blade generates more lift than the retreating blade. In conventional helicopter blades, steps are taken to reduce the effects of blade flapping. There, the rotor blades are flexible and sometimes hinged. Allowing the blade to bend upward or hinge changes its angle of attack and it produces less lift. However, propellers normally used on multirotors are rigid fixed-pitch propellers that do not fully cancel the effects of blade flapping. Therefore, additional forces are introduced into the airframe at the motor hub that need to be considered.

The two main contributions of blade flapping, as proposed in [6], can be described as follows:

- Due to the differences in lift seen between the propeller blades, the blades flap and bend with each revolution. This flapping tilts the plane of the propeller backwards from the direction of motion.
- The bending of the propeller introduces an additional torque into the motor hub that acts on the airframe.

We now discuss these two contributions in detail. Figure 3.2 shows the effects of blade flapping on a propeller.

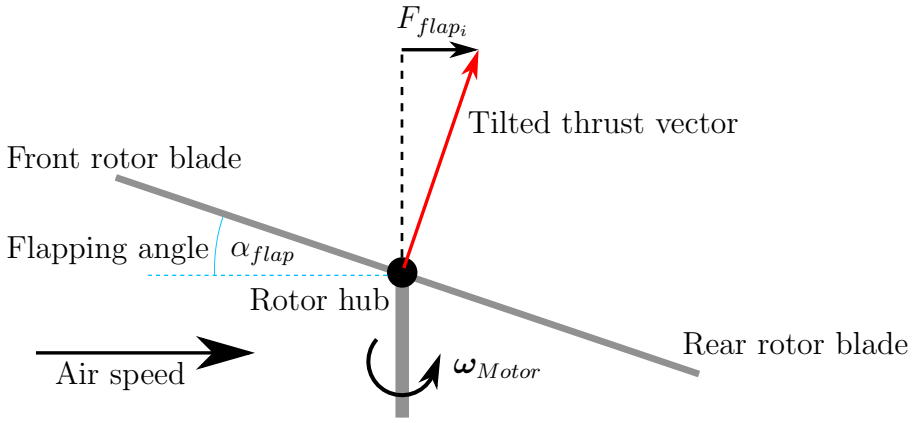



Figure 3.2: Blade flapping effects. The plane of the propeller tilts backwards from the direction of motion, causing a deflection of the thrust vector by the same angle.

The amount of tilt that the propeller plane experiences for a given wind velocity depends on a number of aerodynamic and physical parameters that are not listed here. A good derivation can be found in [7]. For the rigid fixed-pitch propellers used in multirotor aircraft, a sufficiently accurate description of the deflection angle is a linear relationship between translational wind velocity and deflection angle:

$$\alpha_{flap} = k_{flap1} \cdot V_{\perp} \tag{3.17}$$

where $\alpha_{flap}$ is the deflection angle of the propeller plane, $V_{\perp}$ is the true air speed (TAS) component perpendicular to the motor axis (in other words, lying in the original untilted propeller plane), and $k_{flap1}$ is the parameter linking deflection angle and air speed in their linear relationship. Parameter $k_{flap1}$ has to be determined experimentally.

The deflection angle generates a thrust component of magnitude

$$F_{flap_i} = F_{B_{T_i}} \cdot \sin \alpha_{flap_i} \tag{3.18}$$

where $F_{B_{T_i}}$ is the magnitude of the regular motor thrust vector, stated here for Motor $i$. The direction of $F_{flap_i}$ is the airframe's true air speed (TAS) vector projected into the undisturbed propeller plane of Motor $i$. Remember that we no longer assume that the motor axes are perfectly aligned with the Body-fixed $z$-axis, so the propeller plane may not be parallel to the Body-fixed $x-y$ plane.

The force vector $\boldsymbol{F}_{B_{flap_i}}$ in Body-fixed frame acts at the hub of Motor $i$. This force will generate a torque

around the airframe center of gravity:

$$\boldsymbol{\tau}_{B_{F1}} = \boldsymbol{r}_{Motor_i} \times \boldsymbol{F}_{B_{flap_i}} \tag{3.19}$$

where $\boldsymbol{r}_{Motor_i}$ is the lever arm vector for Motor $i$ in Body-fixed frame.
The second contribution of the blade flapping effect is an induced torque introduced into the motor hub due to the flexing of the propeller. Its magnitude is a function of the deflection angle $\alpha_{flap_i}$ and can be expressed as

$$\boldsymbol{\tau}_{B_{F2}} = k_{flap2} \cdot \alpha_{flap_i} \tag{3.20}$$

where $k_{flap2}$ is a parameter that describes the stiffness of the propeller with units $\frac{Nm}{rad}$. The direction of $\boldsymbol{\tau}_{B_{F2}}$ is identical to $\boldsymbol{\tau}_{B_{F1}}$.
Further details may be found in [15] and [12].

## 3.5 Summary of Model Parameters

The improved multirotor model requires additional model specific parameters that describe the added effects acting on the system. The derivation of a number of model specific parameters was already discussed in Chapter 2.7, but more are needed now to describe various environmental effects, for example. They are discussed in detail in this chapter.

### 3.5.1 Mass

The all-up weight (AUW) of a multirotor can easily be found by weighing the physical airframe in the exact configuration the user is trying to simulate using an accurate laboratory scale. See Chapter 2.7.1 for reference.

### 3.5.2 Moment of Inertia

As mentioned previously, the most comprehensive way to model a multirotor to determine physical parameters is to model the entire multirotor as a three-dimensional computer aided design (CAD) model. The finished CAD model allows the user to quickly and easily display the modeled airframe's moment of inertia matrix. Please see Chapter 2.7.2 for additional details.

### 3.5.3 Center of Gravity

In the simplified multirotor model used previously, one of the assumptions was that the airframe center of gravity lies in the geometric plane of the propeller. Also, due to the assumed symmetry of the airframe, the center of gravity coincides with the geometric center of the airframe.

Since these simplifying assumptions no longer hold for the improved model, the true location of the airframe center of gravity has to be determined. As mentioned before, computer aided design software has the functionality to instantly display the exact location of the center of gravity for a three dimensional model.

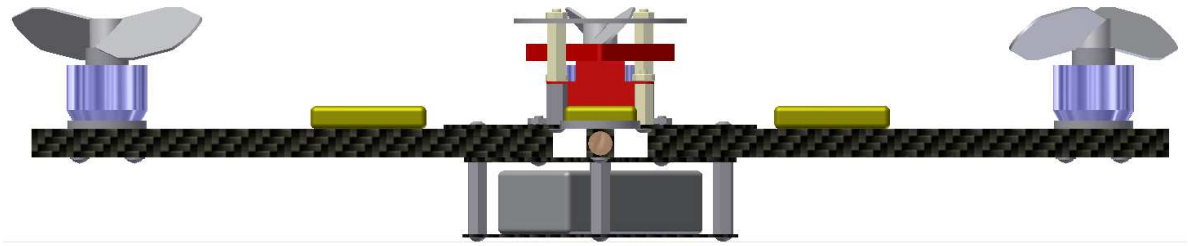
### 3.5.4 Surface area

When the multirotor airframe is modeled in 3D in CAD software, the model can also be used to determine at least an estimated surface area. As mentioned previously in Chapter 3.4.1, it is probably sufficient to compute the surface area along the axes of the Body-fixed reference frame and assume it to be continuous and smooth inbetween these points of measurement. Then the surface area model of the airframe would resemble an ellipsoid of form

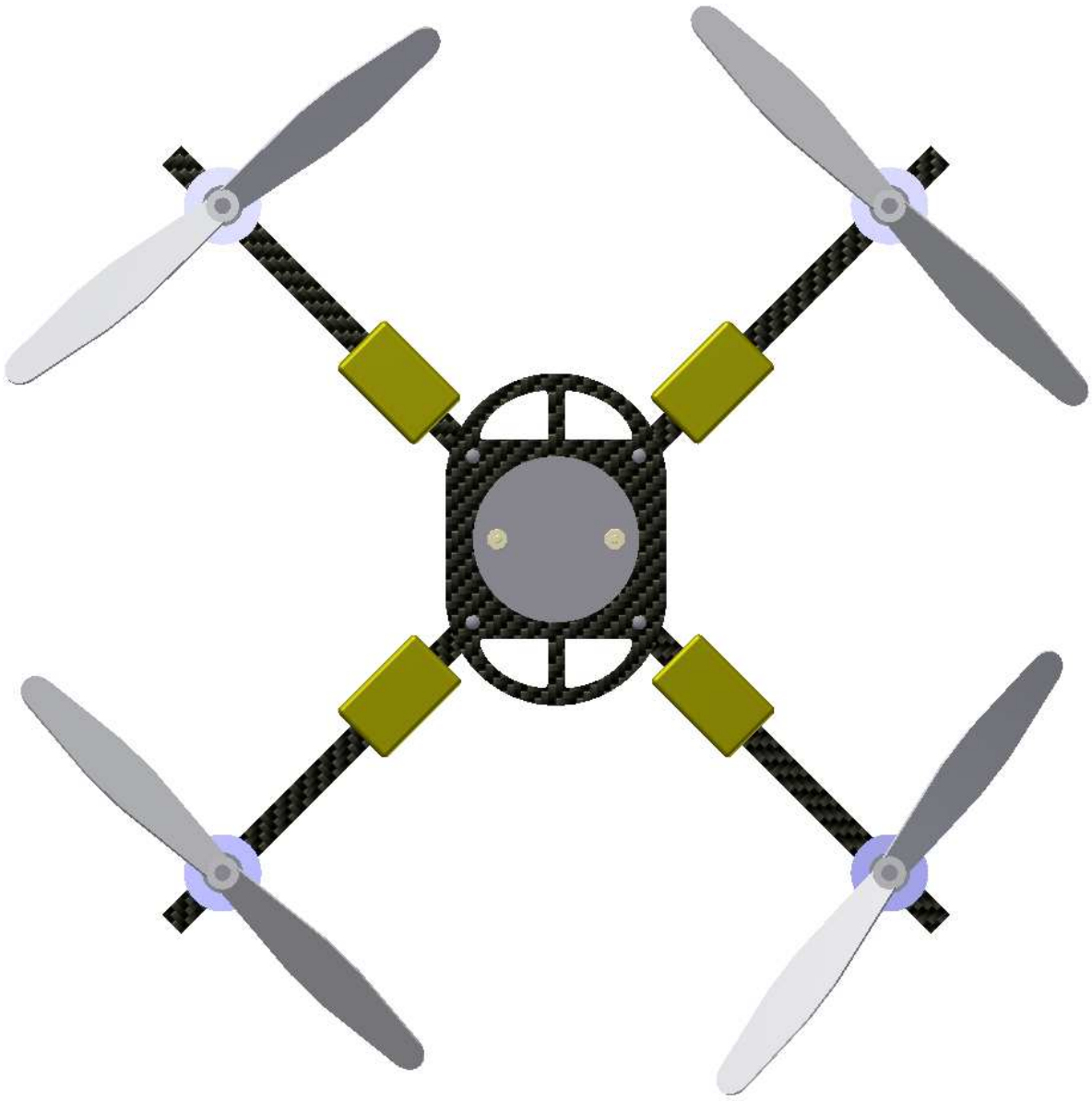
$$\frac{x^2}{A_x^2} + \frac{y^2}{A_y^2} + \frac{z^2}{A_z^2} = 1 \tag{3.21}$$

where the semi-axis lengths $A_x$, $A_y$, $A_z$ are the surface areas when viewing the airframe from along the Body $x$, $y$, or $z$ axis, respectively.

Figure 3.3 shows the views of a quadrotor model from different perspectives. This is the surface area the wind would see when the airframe is moving in those directions.



(a) Airframe along $x$ and $y$ axes



(b) Airframe along $z$-axis

Figure 3.3: Quadrotor 3D CAD perspectives. A fully detailed three-dimensional computer-aided design (CAD) model can be used to estimate the surface area of the airframe.

### 3.5.5 Aerodynamic drag coefficient

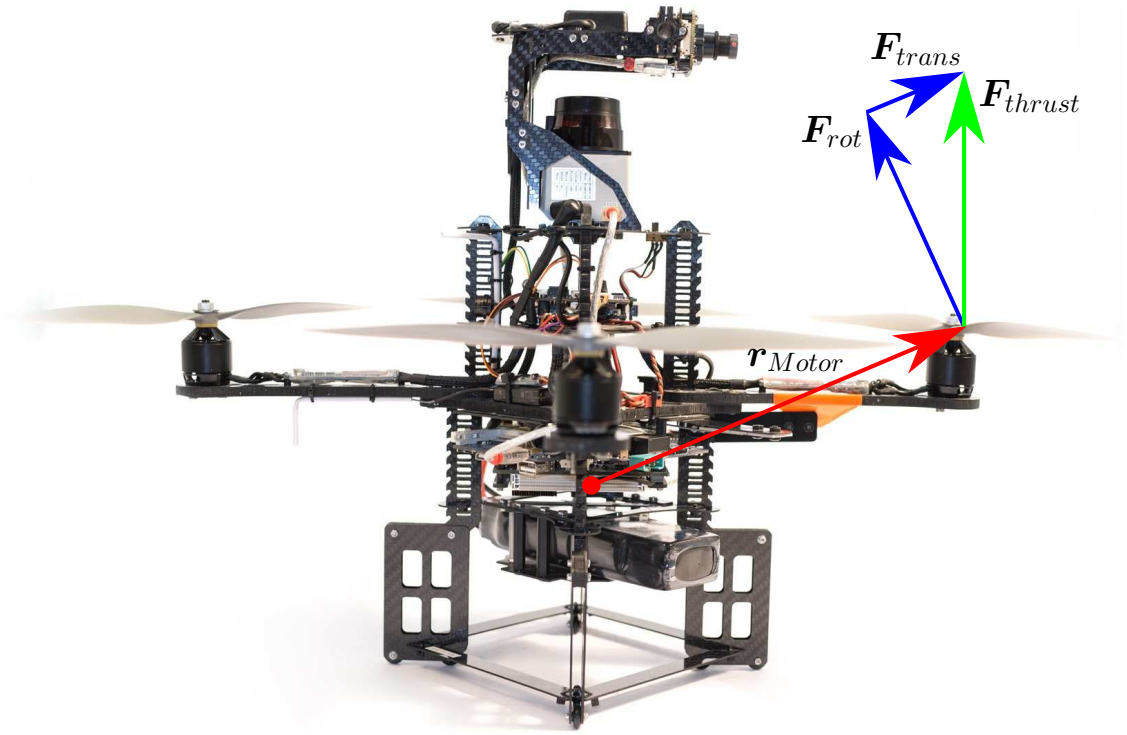
The force acting on the airframe due to aerodynamic drag was previously derived in Chapter 3.4.1. Recall that the equation for the overall drag force in the Body-fixed frame is given by:

$$\boldsymbol{F}_{B_{Drag}} = -\frac{1}{2} \cdot \rho \cdot \boldsymbol{V}_{B_{TAS}} \cdot \|\boldsymbol{V}_{B_{TAS}}\| \cdot C_D \cdot A \tag{3.22}$$

where $\rho$ is the density of air, $C_D$ is the drag coefficient and the quantity $(-\boldsymbol{V}_{B_{TAS}} \cdot \|\boldsymbol{V}_{B_{TAS}}\|)$ is the squared magnitude of true air speed pointing in the true air speed's opposite direction. $A$ is the surface area of the airframe exposed to the wind and the previous section just explained how to find this value.
The remaining unknown parameter is the drag coefficient $C_D$. This parameter can only be identified correctly in a wind tunnel setting, where the actual multirotor is subjected to an air flow and the generated drag force is measured in a controlled testing environment. Of course the drag coefficient can also be approximated. It depends greatly on the shape of the object that is subjected to the air flow. If approximating the multirotor airframe as a shape similar to an angled cube, a drag coefficient of $C_D = 0.80 - 1.00$ can be assumed. See [18] for details.

### 3.5.6 Lever arm

The simplifying assumptions from Chapter 2.3 no longer hold for the improved model, so the airframe center of gravity is not located in the geometric propeller plane. It may lie significantly above or below the propeller plane. However, since the exact location of the center of gravity can be extracted from the multirotor 3D CAD model (see Chapter 2.7.2), the effective lever arm from center of gravity to the motor hub can be computed. Figure 3.4 shows a sideview example for a complicated multirotor airframe. It can be seen that due to the angle of the effective lever arm in this example, only the portion $\boldsymbol{F}_{rot}$ of the thrust vector is used to generate a rotation about the airframe center of gravity. The vector component $\boldsymbol{F}_{trans}$ acts as a translational disturbance force that pulls the airframe sideways with each rotation. This behavior would be especially noticeable when attempting to control position.




*Source*: `www.asctec.de`

Figure 3.4: Multirotor motor lever arm. For this complicated multirotor airframe, the motor lever arm length is the distance from motor axis to the true center of gravity of the airframe.

### 3.5.7 Propeller Aerodynamic Coefficients

#### Dynamic Thrust

As mentioned previously in Chapter 3.4.4, the assumptions for how motor thrust is generated that were used in the simplified multirotor model were overly basic. It was assumed that at a given motor speed, the motor would always produce the same amount of thrust, regardless of how fast the airframe was already moving through the surrounding air. This amount of thrust was the static thrust that was measured in the static test rig shown in Chapter 2.7.4.

However, it is known that the amount of thrust generated for a given motor speed varies greatly with both the speed of the air entering into the propeller and the angle of attack of the propeller plane to the air stream. This is generally called dynamic thrust. For example, if the air is already moving with some speed into the direction the propeller will push it, the generated thrust will decrease. Alternatively, if air is entering the propeller from below (opposing the direction the propeller will push it), the generated thrust produced at the same motor speed will be amplified. See Figure 3.5 for an example from [7]. Here the ratio of dynamic thrust to hover thrust is plotted for a range of climb speeds. Positive climb speed means air is

entering the propeller from above at climb speed and is pushed out the bottom. This reduces the amount of produced thrust. Negative climb speed means the multirotor is descending. In this scenario, air is entering the propeller from below. This leads to an amplification of thrust compared to the hover thrust.

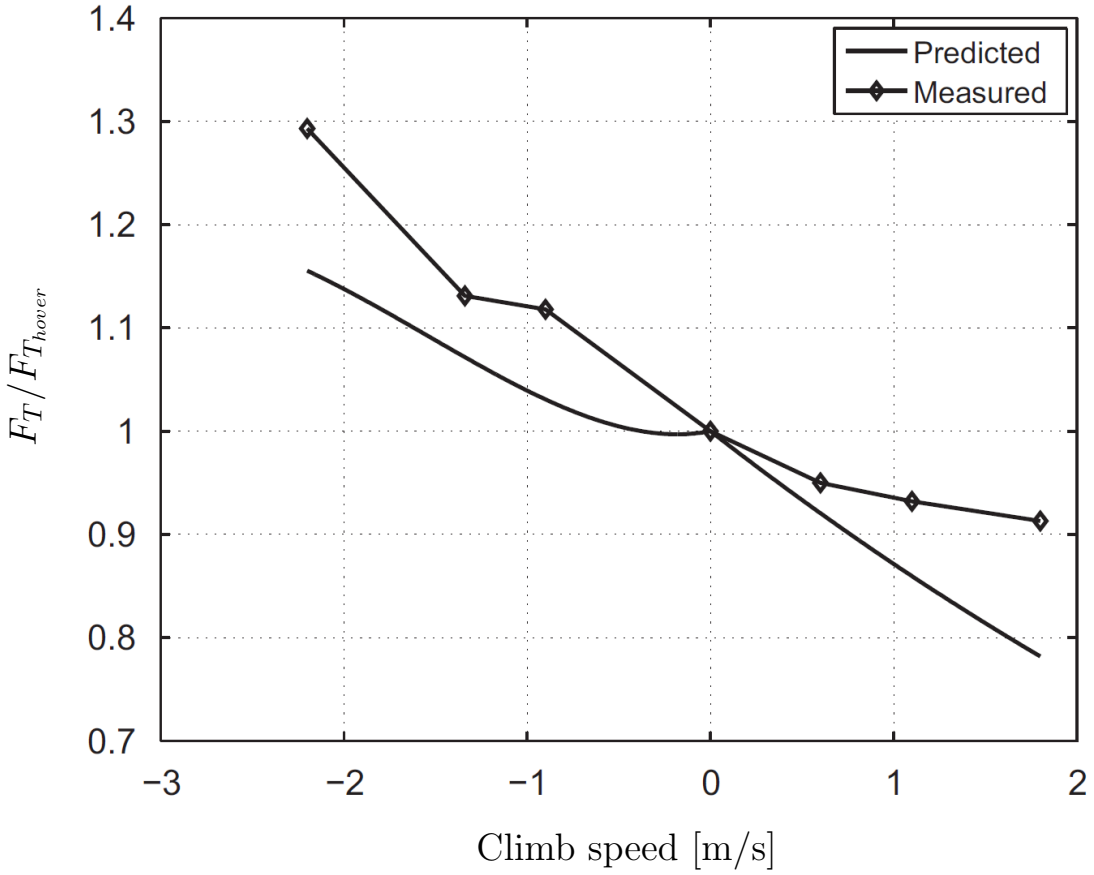



Figure 3.5: Thrust dependence on vertical speed. The amount of thrust that is generated at a given motor speed varies greatly with the speed of the air entering the propeller.

Furthermore, the amount of dynamic thrust also varies with the angle at which the air flows through the spinning propeller. A propeller that has air coming at it from its side at 90 degrees actually generates more lift at a given motor speed than a motor in a completely static environment. Figure 3.6 shows an example for this complicated relationship between air speed and direction of flow taken from [6]. Here an angle of attack of 0 degrees means that air is flowing into the propeller from its side. A positive angle of attack means that the air is flowing into the propeller from the top, and therefore a negative angle of attack describes a scenario in which the air is entering the propeller from below. See [6] for further details.
The fluctuations in generated thrust in comparison to hover thrust are substantial. On the other hand, the displayed range of angle of attack and flight speed is easily reachable during aggressive maneuvers. Therefore, the effects of dynamic thrust are important to consider.

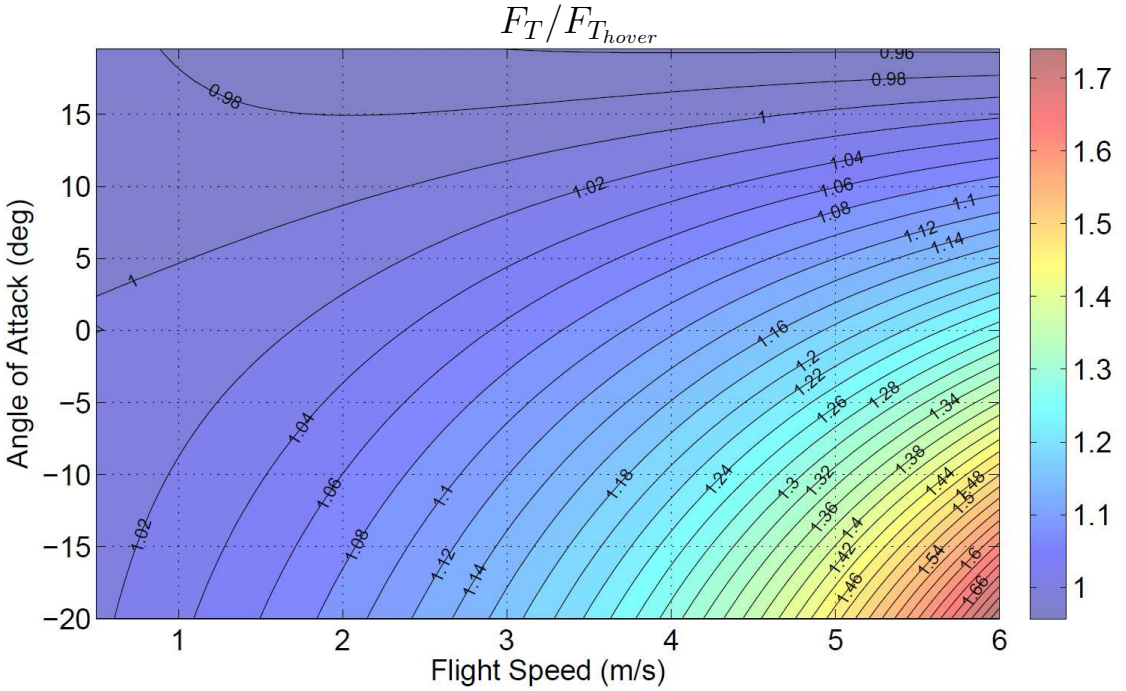



Figure 3.6: Thrust dependence on speed and angle of attack. The amount of thrust generated depends greatly on the direction and speed of the air column entering the propeller.

The relationship of thrust to angle of attack and air speed depends on a large number of aerodynamic parameters. Therefore this relationship is very specific to the propeller used in the multirotor model, and good modeling data can only be obtained through experiments. In this case, that would mean measuring the motor thrust on a test rig that itself is inside a wind tunnel. Then a measurement grid has to be established, adjusting the wind speed throughout the range of possible air speeds experienced by the multirotor, and also adjusting the angle at which the air stream hits the propeller. This generates a lookup table that can then be used for modeling in the simulation.

### Motor Dynamics

Another realistic effect that was not considered in the previous simplified model is motor dynamics. A speed command sent to the motor is not immediately achieved by the motor, as it needs to overcome motor and propeller moments of inertia, among other things. However, when a quality electronic speed controller (ESC) is used, a sufficiently good assumption is that the motor has the behavior of a first order system. The time constant can be obtained by running the motor with the correct propeller in a test rig and measuring motor speed while applying step commands in motor speed. Figure 3.7 shows some data collected during static

thrust measurements for an AscTec Hummingbird motor with the standard 8" propeller. When analyzing the step responses between each speed step, the time constant can be approximated to approximately 50 ms.

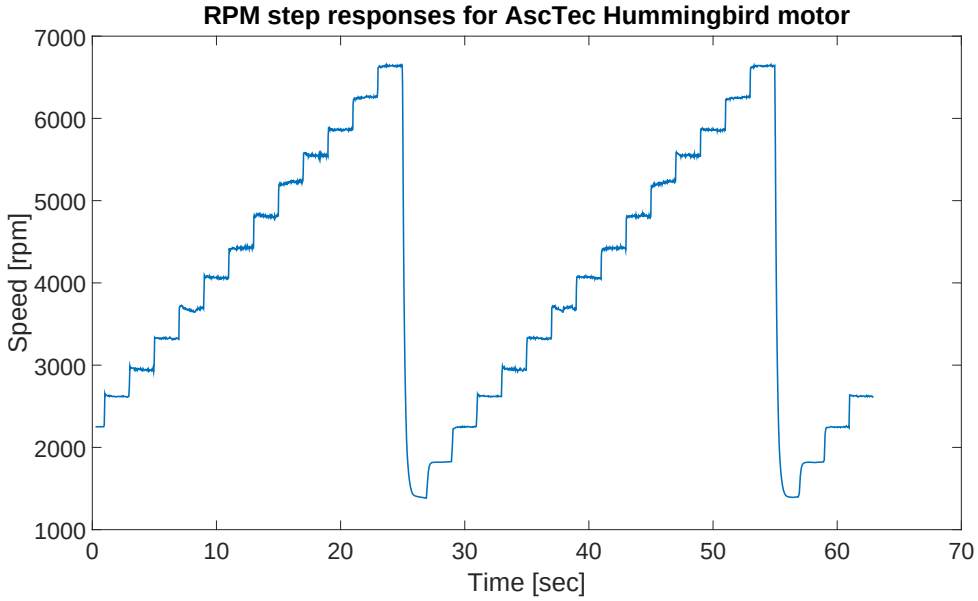



Figure 3.7: Motor dynamics. The motors track speed commands with a first order response.

### 3.5.8 Blade Flapping Coefficients

In Chapter 3.4.5, the forces and moments generated by blade flapping were derived. It was shown that the two components of the effect, the tilting of the rotor plane and the bending of the propeller, can be modeled using two parameters, $k_{flap1}$ and $k_{flap2}$. Here $k_{flap1}$ is a parameter linking the rotor plane deflection angle and air speed in a linear relationship, and $k_{flap2}$ is a parameter that describes the stiffness of the propeller with units $\frac{Nm}{rad}$.

$k_{flap1}$ has to be measured in a wind tunnel, because the spinning propeller has to experience wind coming from its side so that blade flapping actually occurs. With the motor mounted on a load cell, the horizontal force component generated by the tilting propeller plane can be measured and the amount of tilt angle can be computed from that. It may also be helpful to reference existing helicopter literature to deduce a fitting value to approximate this constant. See for example [14].

$k_{flap2}$ can simply be measured by bending the propeller blades by hand to defined angles and recording the torque generated in the motor hub. Naturally, the motor has to be mounted on a load cell for this purpose.

# Chapter 4

# Guide to the *MATLAB/Simulink* Simulation Implementation

In this chapter a complete overview is given of the *MATLAB/Simulink* simulation environment that has been implemented to simulate a multirotor aircraft. First all the elements that make up the simulation are explained block by block and linked with the derivations found in the earlier chapters. Then all the settings and parameters in the initialization file are explained and the procedures necessary to run the simulation are outlined. This should give the user a good overview and make the simulation easy to use.

## 4.1 Overview

Figure 4.1 shows an overview of the complete *Simulink* simulation.

The block in the center contains the full six degree of freedom simulation. Here the full kinematics and dynamics are computed as well as all the forces and moments acting on the airframe. Outputs of this block are the position, velocity, and attitude of the airframe.

The green block on the right side contains all the plotting functionality. That way output data can be visualized conveniently. Below it is a simple built-in *Simulink* animation feature that allows the user to see the attitude of the airframe in 3D during the simulation. This gives a more intuitive understanding than just looking at plots alone.

Sensor output from the 6DOF block enters the red state estimation block. Here the truth data from the equations of motion is treated with noise, bias, and other sensor effects to to turn it into realistic, noisy sensor data. State estimation algorithms would be implemented here to compute estimated aircraft attitude or other values.

The orange block on the left contains a controller to stabilize the system. Inputs are commands from the magenta commands block and actual values from the red state estimation block, and controller outputs are speed commands to each of the motors.

Additionally there is the yellow dynamic parameters block that contains parameters that may be changing during the simulation, like wind, mass, moment of inertia, and disturbance forces.

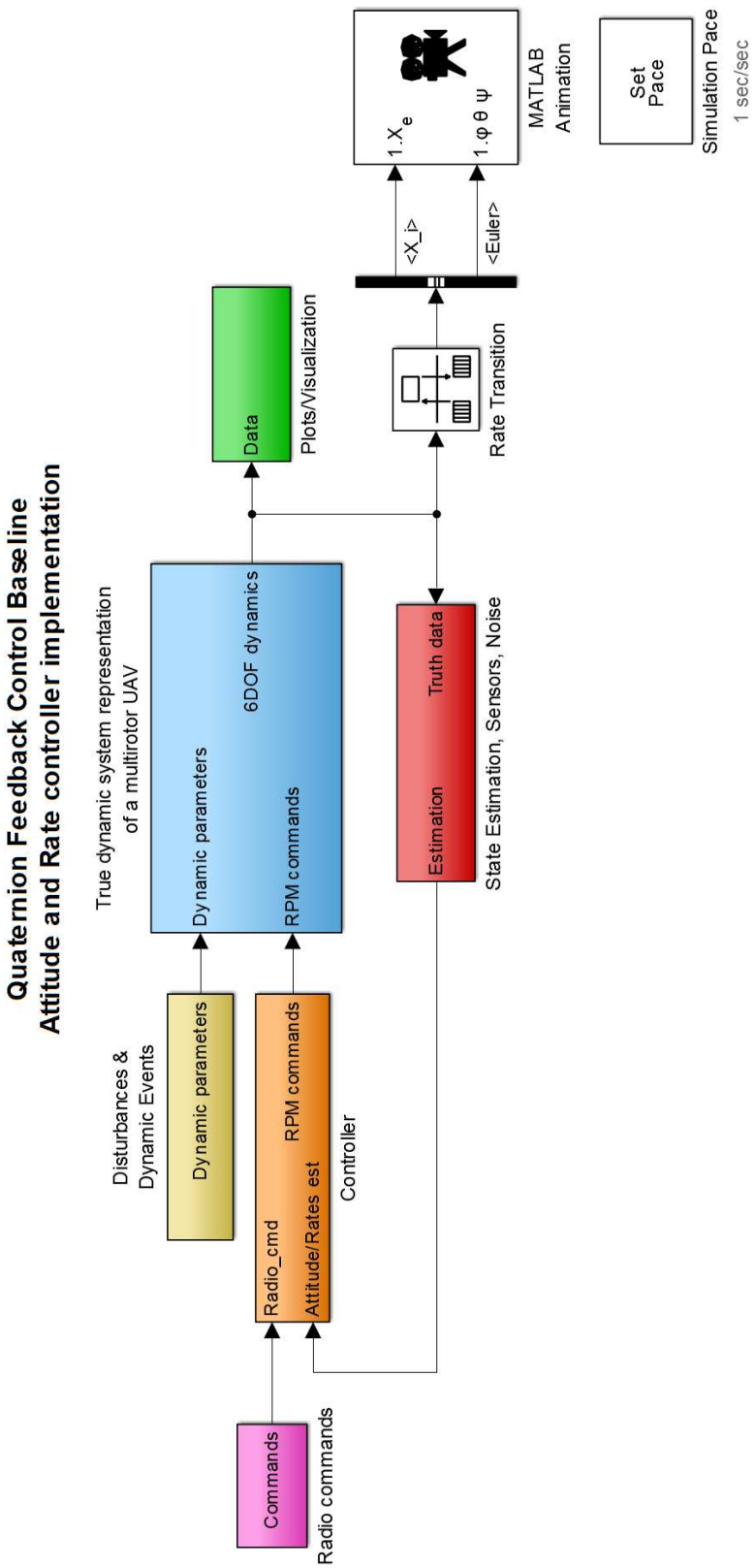



Figure 4.1: Multirotor simulation overview. The complete model to simulate a multirotor aircraft and visualize the output.

## 4.2 Main blocks

We now go through each of the simulation's main blocks in detail.

### 4.2.1 6DOF simulation

Figure 4.2 shows that the 6DOF simulation block itself consists of two blocks. The magenta block contains the six degree of freedom kinematics and dynamics. Inputs are the forces and moments acting on the system and the physical parameters like mass and moment of inertia. The green block contains the computation of all forces and moments acting on the system during flight. This essentially is the implementation of the entire Chapter 3.4.

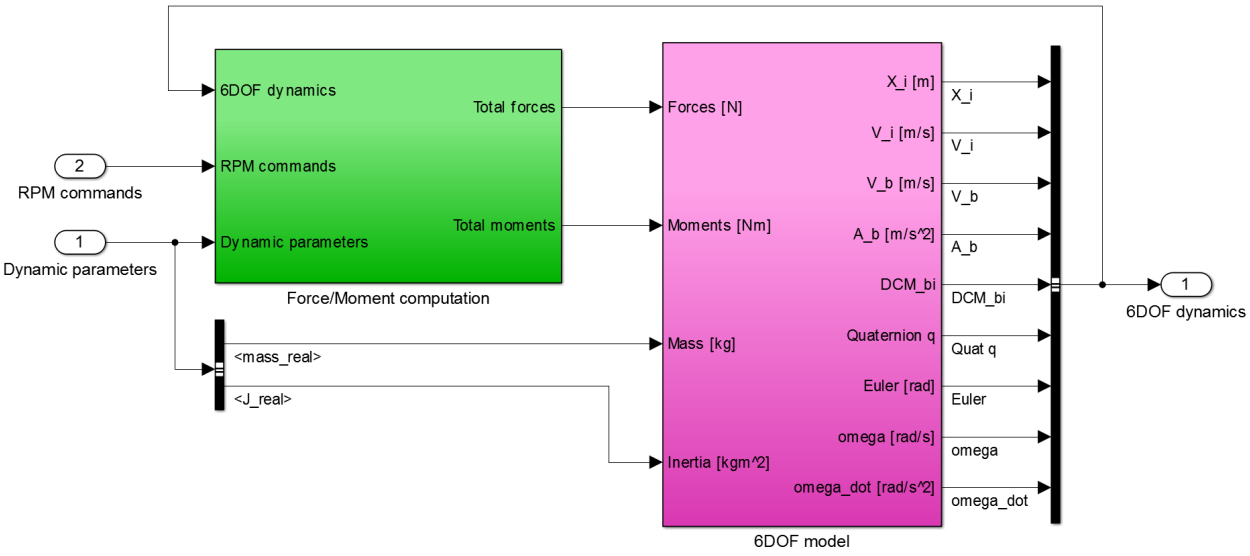



Figure 4.2: 6DOF simulation block. The main 6DOF simulation block contains the kinematics and dynamics computation and a block that computes all forces and moments acting on the system.

As can be expected, the magenta 6DOF model block is the *Simulink* implementation of the kinematics and dynamics as listed in Equations 3.6 and 3.7 in Chapter 3.3. Figure 4.3 shows its contents.

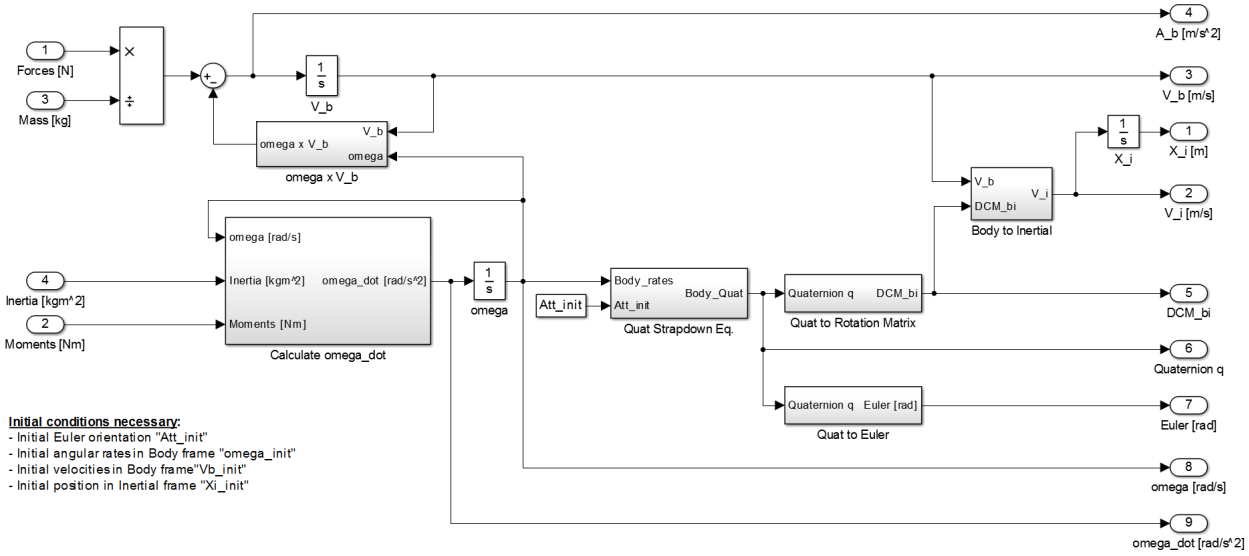



Figure 4.3: 6DOF kinematics and dynamics. This block contains the standard 6DOF kinematics and dynamics equations.

The forces/torques computational block computes the forces acting directly on the airframe (gravity and aerodynamic drag) as well as the forces and torques generated by each motor/propeller, see Figure 4.4. It does so dynamically for as many propellers as are specified in the initialization script.

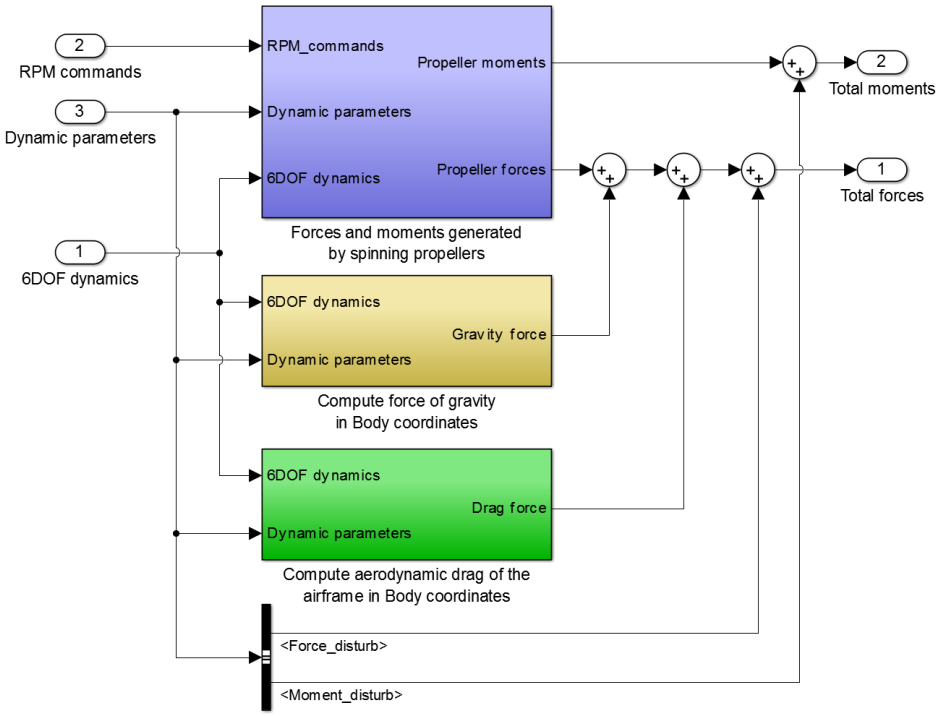



Figure 4.4: Forces and torques. Forces acting directly on the airframe are gravity and drag, all other components are generated due to spinning propellers.

For each motor, the computed forces and torques include dynamic thrust, blade flapping effects, gyroscopic effects, the resulting torques around the airframe center of gravity, meaning all the force and torque components from Chapter 3.4. See Figure 4.5.

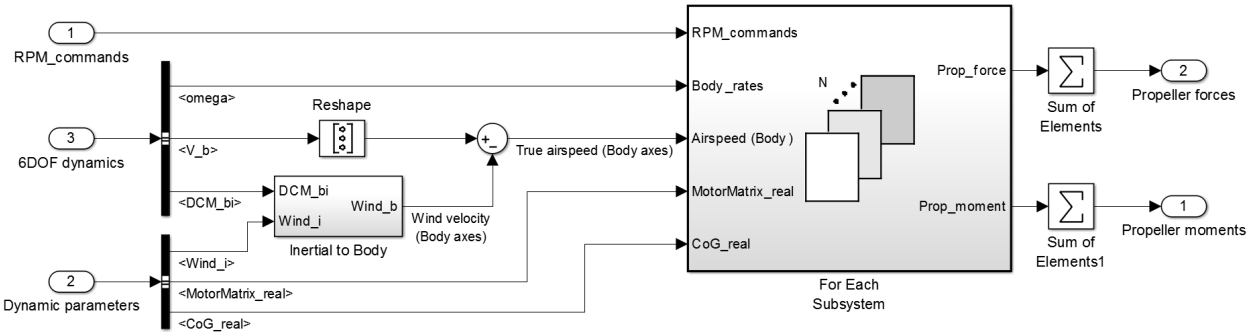



Figure 4.5: Forces and torques per motor. All resulting forces and torques are computed for each motor and summed.

### 4.2.2 Sensors

The sensors block takes the ideal data from the 6DOF output and adds sensor noise and biases to turn it into realistic sensor output data. The user can then use this data directly or pretend not to know the airframe's attitude and instead run a state estimation algorithm to estimate the attitude. See Figure 4.6.

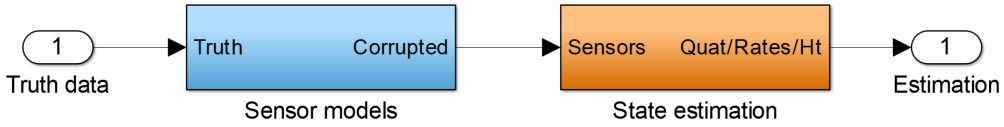



Figure 4.6: Sensors outputs. The ideal, noise-free output from the 6DOF simulations is transformed into realistic sensor data by adding noise and biases.

### 4.2.3 Control

The control block contains a quaternion attitude controller that stabilizes the aircraft in attitude control mode, see Figure 4.7. That means that the user can command an attitude and it will be tracked. This is the normal flight mode for many multirotor systems. The user is invited to explore the details of this controller, as its implementation falls outside the scope of this work.

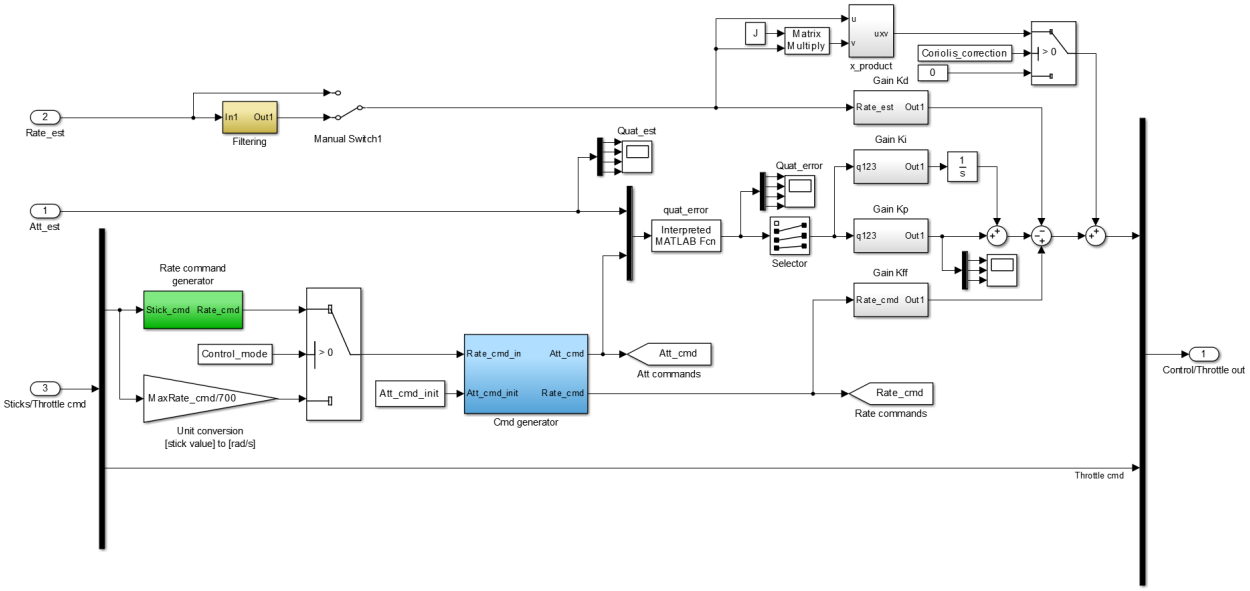



Figure 4.7: Attitude control. This block takes attitude commands and real attitude as inputs and computes the motor speed commands necessary to track the attitude commands. Those motor commands are then fed to the dynamic system model.

### 4.2.4 Input Commands

The input commands block acts very much like the transmitter used by a pilot to control the multirotor aircraft. Here the four channel commands for roll, pitch, yaw, and throttle are generated. For testing purposes, normal signal may include step commands or doublets in any of the channels, for example to check controller performance.

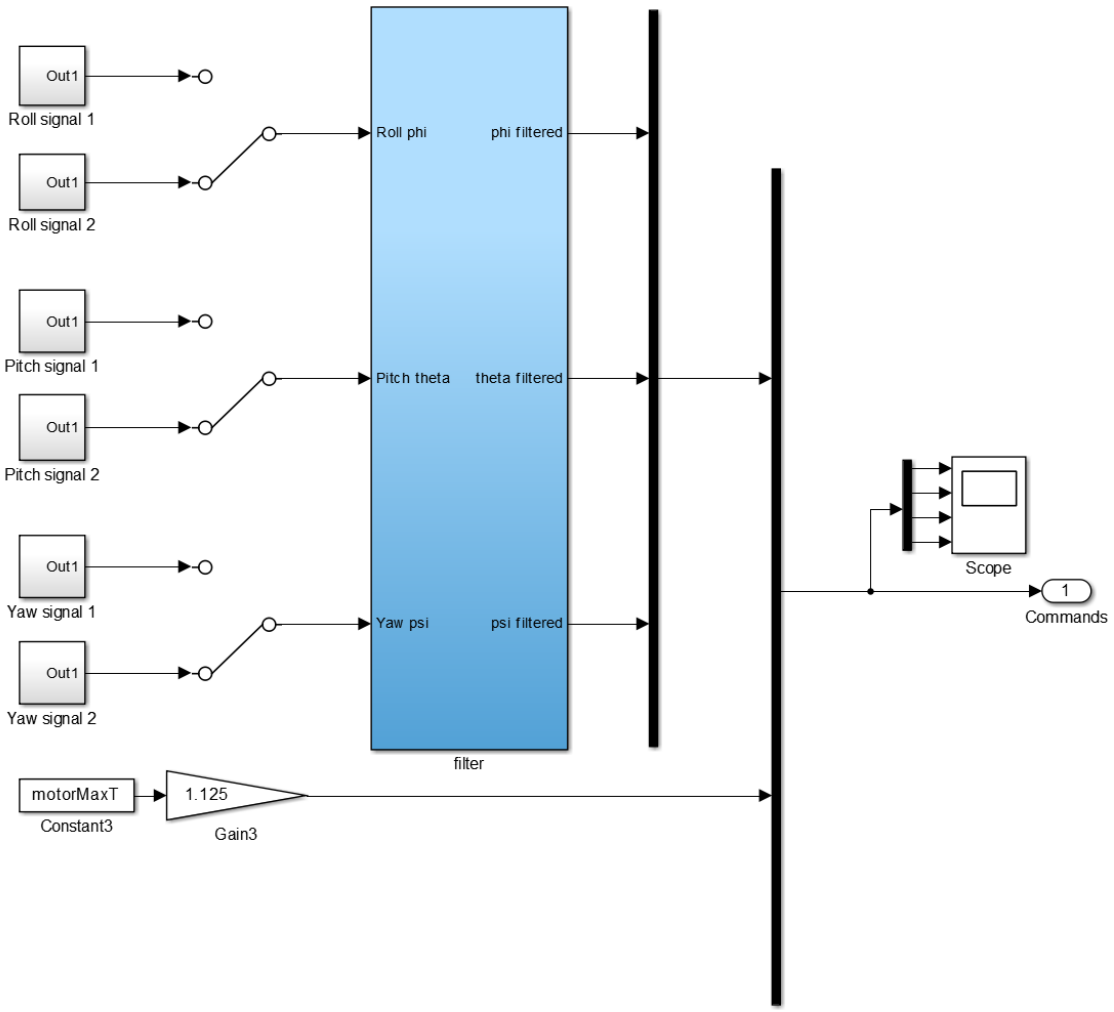



Figure 4.8: Input commands. The four control channels for attitude mode are roll, pitch, yaw, and throttle. Commands for these channels are generated in this block.

### 4.2.5 Plots/Visualization

A number of outputs are already linked into scopes to visualize for debugging or just visualization, see Figure 4.9. The user can add whatever additional data as necessary.

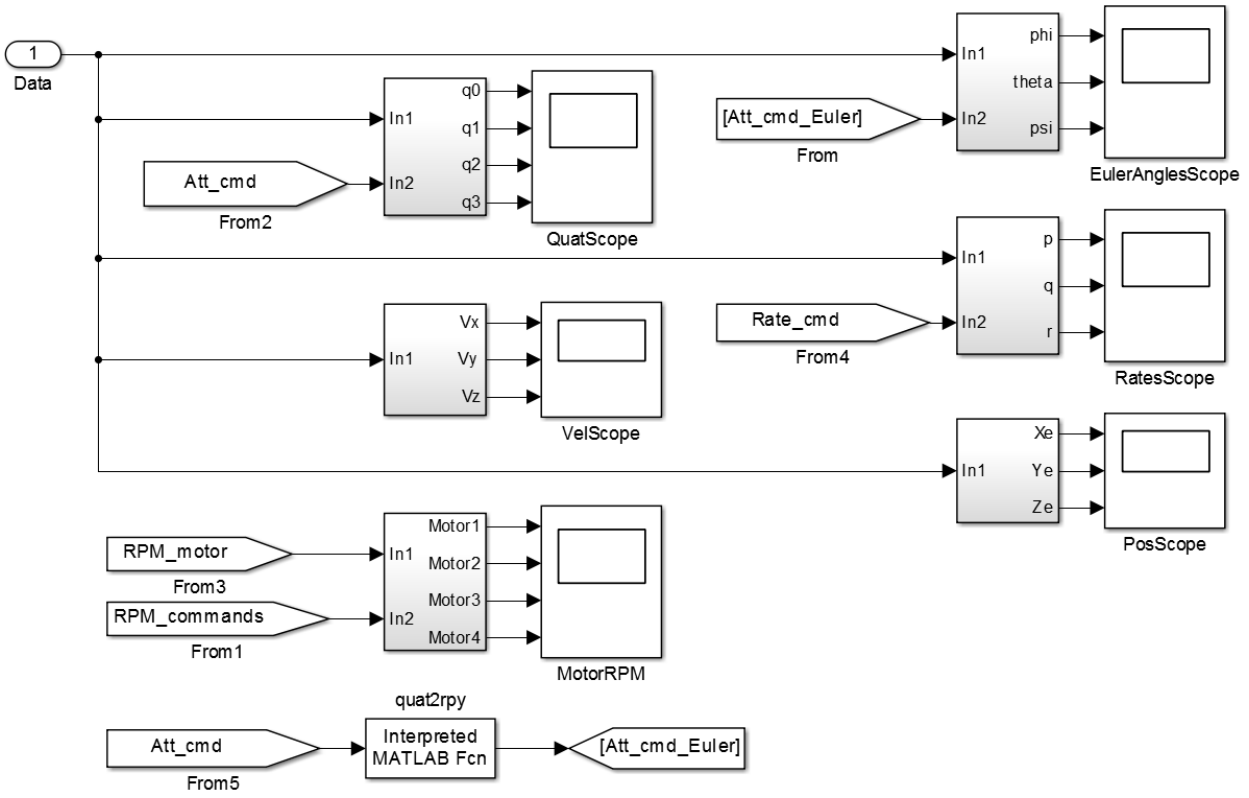



Figure 4.9: Plots/Visualization. A number of scopes are already prepared to visualize many relevant outputs..

## 4.3 Initialization File

The simulation is initialized and executed from the initialization file `Init_Multi.m`. By executing this file, all necessary parameters are initialized and the *Simulink* simulation file `Sim_Multi.mdl` is started. The file sets all necessary parameters for the simulation, for example:

- Simulation frequency
- Various conversion factors
- Initial conditions for angular velocity, linear velocity, attitude, and position
- Location of the airframe center of gravity
- Airframe moment of inertia
- Controller parameters
- Airframe configuration
- Motor parameters

The most critical components of the initialization file are the airframe configuration and motor parameters:

- To describe the physical layout of the multirotor, each motor is defined by its motor arm's angle from the forward Body-fixed $x$-axis and the length of that arm to the geometric center of the airframe, both in the Body-fixed $x - y$ plane.
- Each motor is assigned its own distance in Body-fixed $z$ direction from the geometric center of the airframe.
- Each propeller's direction of rotation is defined as clockwise or counterclockwise.
- Each motor is assigned its own first order transfer function to describe its dynamics.
- Each motor is assigned its own second order RPM to thrust transfer function.
- Each motor is assigned its own second order RPM to induced torque transfer function.
- Each motor is assigned its own minimum and maximum motor speed values.
- Each motor is assigned its own motor misalignment vector that describes the direction of the thrust vector in Body-fixed frame.

- Each propeller is assigned its own diameter and mass.

These definitions allow the user to design any arbitrary multirotor airframe configuration they desire. Of course certain guidelines should still be followed, such as equal number of clockwise and counterclockwise spinning propellers or having the differently rotating propellers arranged interchangeably around the airframe. But the user is not bound to the basic configurations like quadrotor or hexarotor or similar.

# Chapter 5

# Conclusions and Future Work

## 5.1 Conclusions

This thesis addresses the general issue of dynamic modeling for multirotor UAVs. The basic dynamic model usually found in the literature lacks sufficient fidelity and does not adequately represent the real-world behavior of a multirotor airframe in flight. Especially in aggressive flight regimes, unmodeled effects such as dynamic thrust can introduce large errors in the control channels. For example, when the thrust vector and the motor lever arm are not perpendicular, then the motor produces a torque around the airframe center of gravity as well as a sideways force pulling on the airframe. While this kind of disturbance force would not be very noticeable in an attitude control problem, it would definitely represent a problem in a position controller. Therefore, it is good if a simulation model exists that allows the user to address structural or environmental issues such as these effects.

The proposed improved multirotor model addresses this need and should be seen as an attempt to include more realistic behavior into the flight simulation without making the system overly complex. Especially when aerodynamics are involved, complexity can grow very quickly and become burdensome. So while the simulation does introduce a number of new forces and moments into the system, the overall computational burden is still very limited. One advantage of the improved simulation is that it allows the user to design any arbitrary multirotor airframe. Since no assumptions on symmetry are made, the geometry of the airframe can be chosen freely. There are of course physical limits to what design is actually feasible to fly.

However, with this added freedom also comes additional responsibility. The user has to compute or measure more information about his/her UAV than was necessary with the simplified model. However, this tradeoff should be expected. If the user wants the simulation to behave more realistic, more information about the model needs to be provided.

Ultimately, it is futile to attempt to include every minute environmental effect in the simulation. Rather, the solution should be to have control techniques that are robust against these uncertainties and disturbances. But during the process of control design, it is still beneficial to have a simulation that guarantees natural

behavior, for example in order to estimate how large the disturbances from environmental effects can be. This thesis is accompanied by a *MATLAB/Simulink* simulation environment where all the described effects are implemented. It can be used for the early design and testing stages of GNC algorithms for multirotor UAVs.

## 5.2 Future Work

The mathematical groundwork has been laid and the simulation environment has been implemented, but what remains to be done is to generate accurate experimental data for all the parameters necessary for the simulation. A wind tunnel testing setup is needed to measure blade flapping and dynamic thrust dependence. Since this testing is very time-consuming and the parameters are absolutely specific to only that one type of propeller used in testing, the decision what hardware to test is crucial. If it is known that one particular airframe is going to be used as a research vehicle exclusively for a period of time, it would be worthwhile to generate all necessary parameters to get a full dynamic model.

# References


[1] Y. Bai, X. Gong, Z. Hou, and Y. Tian, *Stability Control of Quad-Rotor Based on Explicit Model Following with Inverse Model Feedforward Method*, in IEEE Conference on Mechatronics and Automation, Beijing, China, August 2011, pp. 2189–2194.

[2] S. Bouabdallah, *Design and Control of Quadrotors with Application to Autonomous Flying*, PhD thesis, École Polytechnique Fédérale de Lausanne, Lausanne, Switzerland, February 2007.

[3] T. Bresciani, *Modelling, Identification and Control of a Quadrotor Helicopter*, Master's thesis, Lund University, Lund, Sweden, October 2008.

[4] J. Diebel, *Representing Attitude: Euler Angles, Unit Quaternions, and Rotation Vectors*, Tech. Report, Stanford University, Stanford, CA, USA, October 2006.

[5] D. Gurdan, J. Stumpf, M. Achtelik, K.-M. Doth, G. Hirzinger, and D. Rus, *Energy-efficient Autonomous Four-rotor Flying Robot Controlled at 1 kHz*, in IEEE Conference on Robotics and Automation, Roma, Italy, April 2007, pp. 361–366.

[6] G. M. Hoffmann, H. Huang, S. L. Waslander, and C. J. Tomlin, *Quadrotor Helicopter Flight Dynamics and Control: Theory and Experiment*, in AIAA Guidance, Navigation and Control Conference, Hilton Head, SC, USA, August 2007.

[7] G. M. Hoffmann, H. Huang, S. L. Waslander, and C. J. Tomlin, *Precision flight control for a multi-vehicle quadrotor helicopter testbed*, Journal of Control Engineering Practice, 19 (2011), pp. 1023–1036.

[8] T. Madani and A. Benallegue, *Sliding Mode Observer and Backstepping Control for a Quadrotor Unmanned Aerial Vehicles*, in American Control Conference, New York, NY, USA, July 2007, pp. 5887–5892.

[9] S. Mallikarjunan, B. Nesbitt, E. Kharisov, E. Xargay, N. Hovakimyan, and C. Cao, $\mathcal{L}_1$ *Adaptive Controller for Attitude Control of Multirotors*, in AIAA Guidance, Navigation and Control Conference, Minneapolis, MN, USA, August 2012.

[10] D. Mellinger and V. Kumar, *Minimum Snap Trajectory Generation and Control for Quadrotors*, in IEEE Conference on Robotics and Automation, Shanghai, China, May 2011, pp. 2520–2525.

[11] D. W. Mellinger, *Trajectory Generation and Control for Quadrotors*, PhD thesis, University of Pennsylvania, Philadelphia, PA, USA, January 2012.

[12] M. Mikkelsen, *Development, Modelling and Control of a Multirotor Vehicle*, Master's thesis, Umeå University, Umeå, Sweden, September 2015.

[13] A. Mokhtari and A. Benallegue, *Dynamic Feedback Controller of Euler Angles and Wind parameters estimation for a Quadrotor Unmanned Aerial Vehicle*, in IEEE Conference on Robotics and Automation, New Orleans, LA, USA, April 2004, pp. 2359–2366.

[14] S. Newman, *The Foundations of Helicopter Flight*, Butterworth-Heinemann, Oxford, UK, 1994.

[15] S. Omari, M.-D. Hua, G. Ducard, and T. Hamel, *Nonlinear Control or VTOL UAVs Incorporating Flapping Dynamics*, in IEEE Conference on Intelligent Robots and Systems, Tokyo, Japan, November 2013, pp. 2419–2425.

[16] P. Pounds, R. Mahony, and J. Gresham, *Towards Dynamically-Favourable Quad-Rotor Aerial Robots*, in Australasian Conference on Robotics and Automation, Canberra, Australia, December 2004.

[17] P. Pounds, R. Mahony, P. Hynes, and J. Roberts, *Design of a Four-Rotor Aerial Robot*, in Australasian Conference on Robotics and Automation, Auckland, New Zealand, November 2002, pp. 145–150.

[18] M. Sadraey, *Aircraft Performance Analysis*, VDM Verlag Dr. Müller, Saarbrücken, Germany, 2009.

[19] B. L. Stevens and F. L. Lewis, *Aircraft Control and Simulation, Second Edition*, John Wiley & Sons, Inc., Hoboken, NJ, USA, 2003.

[20] Y. S. Suh, *Robust control of a quad-rotor aerial vehicle*, International Journal of Applied Electromagnetics and Mechanics, 18 (2003), pp. 103–114.

# Appendices

# Appendix A

# Particular solutions for the simplified model

This appendix continues the example from the end of Chapter 2 and includes more particular solutions to the six degree of freedom equations of motion for commonly found airframes. The example in Chapter 2.8 already covered a quadrotor airframe in '+ Mode', so the remaining configurations investigated here are:

- quadrotor in 'X mode'
- hexarotor in '+ Mode'
- hexarotor in 'X Mode'
- octocopter in '+ Mode'
- octocopter in 'X Mode'

Since we are dealing with the simplified model, all simplifying assumptions hold. The dynamics and kinematics are unchanged for all the different airframe configurations; the only thing that changes is how the motors generate the force and torque vector. So the only things changing in Equation 2.19 are $\boldsymbol{F}_B$ and $\boldsymbol{\tau}_B$.

## A.1 Quadrotor in 'X Mode'

### A.1.1 Airframe geometry

Diagram of the airframe geometry:

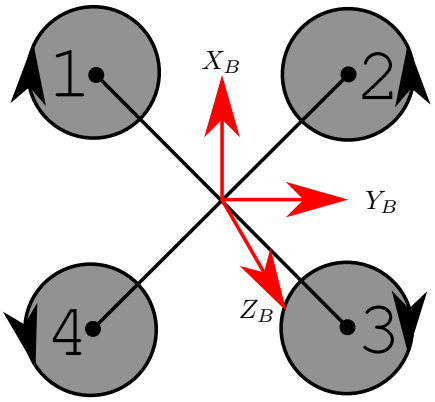



Figure A.1: Quadrotor in 'X Mode'.

### A.1.2 Forces and Moments

Gravity acting on the airframe:

$$\boldsymbol{F}_{B_G} = \begin{bmatrix} -m \cdot g \cdot s\theta \\ m \cdot g \cdot s\phi \cdot c\theta \\ m \cdot g \cdot c\phi \cdot c\theta \end{bmatrix} \tag{A.1}$$

Thrust force generated by all motors:

$$\boldsymbol{F}_{B_T} = \begin{bmatrix} 0 \\ 0 \\ -k_T \cdot \left(\omega_{M_1}^2 + \omega_{M_2}^2 + \omega_{M_3}^2 + \omega_{M_4}^2\right) \end{bmatrix} \tag{A.2}$$

Torque due to motor thrust and propeller aerodynamic drag:

$$\boldsymbol{\tau}_{B_T} = \begin{bmatrix} \frac{1}{\sqrt{2}} \cdot l \cdot k_T \cdot \left(\omega_{M_1}^2 + \omega_{M_4}^2 - \omega_{M_2}^2 - \omega_{M_3}^2\right) \\ \frac{1}{\sqrt{2}} \cdot l \cdot k_T \cdot \left(\omega_{M_1}^2 + \omega_{M_2}^2 - \omega_{M_3}^2 - \omega_{M_4}^2\right) \\ k_D \cdot \left(\omega_{M_2}^2 + \omega_{M_4}^2 - \omega_{M_1}^2 - \omega_{M_3}^2\right) \end{bmatrix} \tag{A.3}$$

Torque due to gyroscopic effects:

$$\begin{aligned}\boldsymbol{\tau}_{B_{Gyro}} &= J_{MP} \cdot \left( \begin{bmatrix} p \\ q \\ r \end{bmatrix} \times \begin{bmatrix} 0 \\ 0 \\ \omega_{M_1} - \omega_{M_2} + \omega_{M_3} - \omega_{M_4} \end{bmatrix} \right) \\ \boldsymbol{\tau}_{B_{Gyro}} &= J_{MP} \cdot \begin{bmatrix} q \cdot (\omega_{M_1} - \omega_{M_2} + \omega_{M_3} - \omega_{M_4}) \\ p \cdot (-\omega_{M_1} + \omega_{M_2} - \omega_{M_3} + \omega_{M_4}) \\ 0 \end{bmatrix}\end{aligned} \tag{A.4}$$

## A.2 Hexarotor in '+ Mode'

### A.2.1 Airframe geometry

Diagram of the airframe geometry:

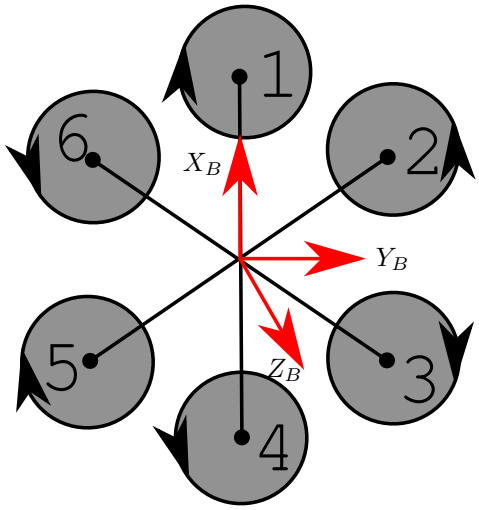



Figure A.2: Hexarotor in '+ Mode'.

### A.2.2 Forces and Moments

Gravity acting on the airframe:

$$\boldsymbol{F}_{B_G} = \begin{bmatrix} -m \cdot g \cdot s\theta \\ m \cdot g \cdot s\phi \cdot c\theta \\ m \cdot g \cdot c\phi \cdot c\theta \end{bmatrix} \tag{A.5}$$

Thrust force generated by all motors:

$$\boldsymbol{F}_{B_T} = \begin{bmatrix} 0 \\ 0 \\ -k_T \cdot \left(\omega_{M_1}^2 + \omega_{M_2}^2 + \omega_{M_3}^2 + \omega_{M_4}^2 + \omega_{M_5}^2 + \omega_{M_6}^2\right) \end{bmatrix} \tag{A.6}$$

Torque due to motor thrust and propeller aerodynamic drag:

$$\boldsymbol{\tau}_{B_T} = \begin{bmatrix} \frac{\sqrt{3}}{2} \cdot l \cdot k_T \cdot \left(\omega_{M_5}^2 + \omega_{M_6}^2 - \omega_{M_2}^2 - \omega_{M_3}^2\right) \\ \frac{1}{2} \cdot l \cdot k_T \cdot \left(2 \cdot \omega_{M_1}^2 + \omega_{M_2}^2 + \omega_{M_6}^2 - 2 \cdot \omega_{M_4}^2 - \omega_{M_3}^2 - \omega_{M_5}^2\right) \\ k_D \cdot \left(\omega_{M_2}^2 + \omega_{M_4}^2 + \omega_{M_6}^2 - \omega_{M_1}^2 - \omega_{M_3}^2 - \omega_{M_5}^2\right) \end{bmatrix} \tag{A.7}$$

Torque due to gyroscopic effects:

$$\begin{aligned}\boldsymbol{\tau}_{B_{Gyro}} &= J_{MP} \cdot \left( \begin{bmatrix} p \\ q \\ r \end{bmatrix} \times \begin{bmatrix} 0 \\ 0 \\ \omega_{M_1} - \omega_{M_2} + \omega_{M_3} - \omega_{M_4} + \omega_{M_5} - \omega_{M_6} \end{bmatrix} \right) \\ \boldsymbol{\tau}_{B_{Gyro}} &= J_{MP} \cdot \begin{bmatrix} q \cdot (\omega_{M_1} - \omega_{M_2} + \omega_{M_3} - \omega_{M_4} + \omega_{M_5} - \omega_{M_6}) \\ p \cdot (-\omega_{M_1} + \omega_{M_2} - \omega_{M_3} + \omega_{M_4} - \omega_{M_5} + \omega_{M_6}) \\ 0 \end{bmatrix} \end{aligned} \tag{A.8}$$

## A.3 Hexarotor in 'X Mode'

### A.3.1 Airframe geometry

Diagram of the airframe geometry:

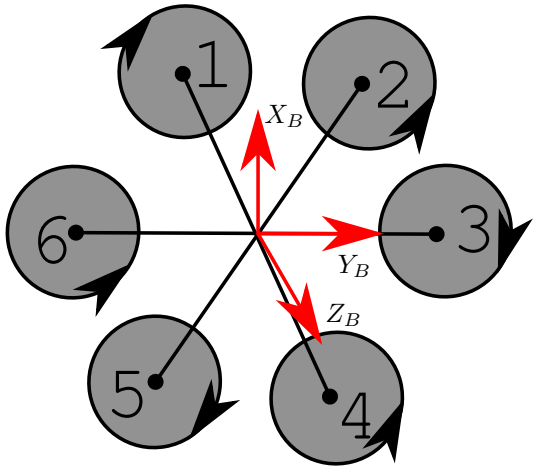



Figure A.3: Hexarotor in 'X Mode'.

### A.3.2 Forces and Moments

Gravity acting on the airframe:

$$\boldsymbol{F}_{B_G} = \begin{bmatrix} -m \cdot g \cdot s\theta \\ m \cdot g \cdot s\phi \cdot c\theta \\ m \cdot g \cdot c\phi \cdot c\theta \end{bmatrix} \tag{A.9}$$

Thrust force generated by all motors:

$$\boldsymbol{F}_{B_T} = \begin{bmatrix} 0 \\ 0 \\ -k_T \cdot \left(\omega_{M_1}^2 + \omega_{M_2}^2 + \omega_{M_3}^2 + \omega_{M_4}^2 + \omega_{M_5}^2 + \omega_{M_6}^2\right) \end{bmatrix} \tag{A.10}$$

Torque due to motor thrust and propeller aerodynamic drag:

$$\boldsymbol{\tau}_{B_T} = \begin{bmatrix} \frac{1}{2} \cdot l \cdot k_T \cdot \left(\omega_{M_1}^2 + \omega_{M_5}^2 + 2 \cdot \omega_{M_6}^2 - \omega_{M_2}^2 - 2 \cdot \omega_{M_3}^2 - \omega_{M_4}^2\right) \\ \frac{\sqrt{3}}{2} \cdot l \cdot k_T \cdot \left(\omega_{M_1}^2 + \omega_{M_2}^2 - \omega_{M_4}^2 - \omega_{M_5}^2\right) \\ k_D \cdot \left(\omega_{M_2}^2 + \omega_{M_4}^2 + \omega_{M_6}^2 - \omega_{M_1}^2 - \omega_{M_3}^2 - \omega_{M_5}^2\right) \end{bmatrix} \tag{A.11}$$

Torque due to gyroscopic effects:

$$\begin{aligned}\boldsymbol{\tau}_{B_{Gyro}} &= J_{MP} \cdot \left( \begin{bmatrix} p \\ q \\ r \end{bmatrix} \times \begin{bmatrix} 0 \\ 0 \\ \omega_{M_1} - \omega_{M_2} + \omega_{M_3} - \omega_{M_4} + \omega_{M_5} - \omega_{M_6} \end{bmatrix} \right) \\ \boldsymbol{\tau}_{B_{Gyro}} &= J_{MP} \cdot \begin{bmatrix} q \cdot (\omega_{M_1} - \omega_{M_2} + \omega_{M_3} - \omega_{M_4} + \omega_{M_5} - \omega_{M_6}) \\ p \cdot (-\omega_{M_1} + \omega_{M_2} - \omega_{M_3} + \omega_{M_4} - \omega_{M_5} + \omega_{M_6}) \\ 0 \end{bmatrix} \end{aligned} \tag{A.12}$$

## A.4 Octocopter in '+ Mode'

### A.4.1 Airframe geometry

Diagram of the airframe geometry:

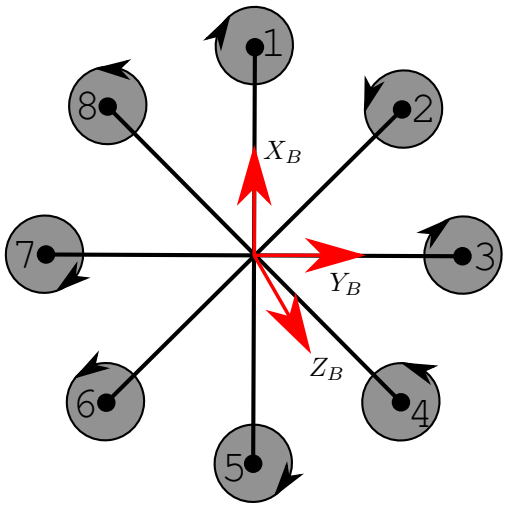



Figure A.4: Octocopter in '+ Mode'.

### A.4.2 Forces and Moments

Gravity acting on the airframe:

$$\boldsymbol{F}_{B_G} = \begin{bmatrix} -m \cdot g \cdot s\theta \\ m \cdot g \cdot s\phi \cdot c\theta \\ m \cdot g \cdot c\phi \cdot c\theta \end{bmatrix} \tag{A.13}$$

Thrust force generated by all motors:

$$\boldsymbol{F}_{B_T} = \begin{bmatrix} 0 \\ 0 \\ -k_T \cdot \left(\omega_{M_1}^2 + \omega_{M_2}^2 + \omega_{M_3}^2 + \omega_{M_4}^2 + \omega_{M_5}^2 + \omega_{M_6}^2 + \omega_{M_7}^2 + \omega_{M_8}^2\right) \end{bmatrix} \tag{A.14}$$

Torque due to motor thrust and propeller aerodynamic drag:

$$\boldsymbol{\tau}_{B_T} = \begin{bmatrix} \frac{1}{\sqrt{2}} \cdot l \cdot k_T \cdot \left(\omega_{M_6}^2 + \sqrt{2} \cdot \omega_{M_7}^2 + \omega_{M_8}^2 - \omega_{M_2}^2 - \sqrt{2} \cdot \omega_{M_3}^2 - \omega_{M_4}^2\right) \\ \frac{1}{\sqrt{2}} \cdot l \cdot k_T \cdot \left(\sqrt{2} \cdot \omega_{M_1}^2 + \omega_{M_2}^2 + \omega_{M_8}^2 - \omega_{M_4}^2 - \sqrt{2} \cdot \omega_{M_5}^2 - \omega_{M_6}^2\right) \\ k_D \cdot \left(\omega_{M_2}^2 + \omega_{M_4}^2 + \omega_{M_6}^2 + \omega_{M_8}^2 - \omega_{M_1}^2 - \omega_{M_3}^2 - \omega_{M_5}^2 - \omega_{M_7}^2\right) \end{bmatrix} \tag{A.15}$$

Torque due to gyroscopic effects:

$$\begin{aligned}
\boldsymbol{\tau}_{B_{Gyro}} &= J_{MP} \cdot \left( \begin{bmatrix} p \\ q \\ r \end{bmatrix} \times \begin{bmatrix} 0 \\ 0 \\ \omega_{M_1} - \omega_{M_2} + \omega_{M_3} - \omega_{M_4} + \omega_{M_5} - \omega_{M_6} + \omega_{M_7} - \omega_{M_8} \end{bmatrix} \right) \\
\boldsymbol{\tau}_{B_{Gyro}} &= J_{MP} \cdot \begin{bmatrix} q \cdot (\omega_{M_1} - \omega_{M_2} + \omega_{M_3} - \omega_{M_4} + \omega_{M_5} - \omega_{M_6} + \omega_{M_7} - \omega_{M_8}) \\ p \cdot (-\omega_{M_1} + \omega_{M_2} - \omega_{M_3} + \omega_{M_4} - \omega_{M_5} + \omega_{M_6} - \omega_{M_7} + \omega_{M_8}) \\ 0 \end{bmatrix}
\end{aligned} \tag{A.16}$$

## A.5 Octocopter in 'X Mode'

### A.5.1 Airframe geometry

Diagram of the airframe geometry:

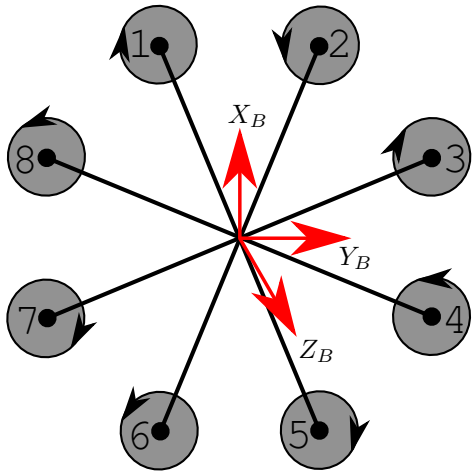



Figure A.5: Octocopter in 'X Mode'.

### A.5.2 Forces and Moments

Gravity acting on the airframe:

$$\boldsymbol{F}_{B_G} = \begin{bmatrix} -m \cdot g \cdot s\theta \\ m \cdot g \cdot s\phi \cdot c\theta \\ m \cdot g \cdot c\phi \cdot c\theta \end{bmatrix} \tag{A.17}$$

Thrust force generated by all motors:

$$\boldsymbol{F}_{B_T} = \begin{bmatrix} 0 \\ 0 \\ -k_T \cdot \left(\omega_{M_1}^2 + \omega_{M_2}^2 + \omega_{M_3}^2 + \omega_{M_4}^2 + \omega_{M_5}^2 + \omega_{M_6}^2 + \omega_{M_7}^2 + \omega_{M_8}^2\right) \end{bmatrix} \tag{A.18}$$

Torque due to motor thrust and propeller aerodynamic drag:

$$\boldsymbol{\tau}_{B_T} = \begin{bmatrix} l \cdot k_T \cdot \left[0.382683 \cdot \left(\omega_{M_1}^2 + \omega_{M_6}^2 - \omega_{M_2}^2 - \omega_{M_5}^2\right) + 0.923880 \cdot \left(\omega_{M_7}^2 + \omega_{M_8}^2 - \omega_{M_3}^2 - \omega_{M_4}^2\right)\right] \\ l \cdot k_T \cdot \left[0.382683 \cdot \left(\omega_{M_3}^2 + \omega_{M_8}^2 - \omega_{M_4}^2 - \omega_{M_7}^2\right) + 0.923880 \cdot \left(\omega_{M_1}^2 + \omega_{M_2}^2 - \omega_{M_5}^2 - \omega_{M_6}^2\right)\right] \\ k_D \cdot \left(\omega_{M_2}^2 + \omega_{M_4}^2 + \omega_{M_6}^2 + \omega_{M_8}^2 - \omega_{M_1}^2 - \omega_{M_3}^2 - \omega_{M_5}^2 - \omega_{M_7}^2\right) \end{bmatrix} \tag{A.19}$$

Torque due to gyroscopic effects:

$$\begin{aligned}
\boldsymbol{\tau}_{B_{Gyro}} &= J_{MP} \cdot \left( \begin{bmatrix} p \\ q \\ r \end{bmatrix} \times \begin{bmatrix} 0 \\ 0 \\ \omega_{M_1} - \omega_{M_2} + \omega_{M_3} - \omega_{M_4} + \omega_{M_5} - \omega_{M_6} + \omega_{M_7} - \omega_{M_8} \end{bmatrix} \right) \\
\boldsymbol{\tau}_{B_{Gyro}} &= J_{MP} \cdot \begin{bmatrix} q \cdot (\omega_{M_1} - \omega_{M_2} + \omega_{M_3} - \omega_{M_4} + \omega_{M_5} - \omega_{M_6} + \omega_{M_7} - \omega_{M_8}) \\ p \cdot (-\omega_{M_1} + \omega_{M_2} - \omega_{M_3} + \omega_{M_4} - \omega_{M_5} + \omega_{M_6} - \omega_{M_7} + \omega_{M_8}) \\ 0 \end{bmatrix}
\end{aligned} \tag{A.20}$$