National Technical University of Athens
School of Mechanical Engineering
Mechanical Design and Automatic Control
Control Systems Lab

Modeling And System Identification of a multirotor UAV using simulation data

Diploma Thesis

by

Iliopoulou Antouanetta

Advisor:
K.J.Kyriakopoulos, Professor NTUA

Athens, March 2022

I have read and understood the rules on plagiarism and the proper format to cite sources as presented in the Writing Dissertations Guide. I declare that, to the best of my knowledge, the content of this Thesis is the product of my own work and there are references to all the sources I have used.

The opinions and conclusions contained in this Diploma Thesis are those of the author and it should not be assumed that they represent the official positions of the School of Mechanical Engineering or the National Technical University of Athens.

Iliopoulou Antouanetta

Acknowledgments

First of all, I would like to thank my professor K.J.Kyriakopoulos, for the opportunity he gave me to deal with this interesting topic and the support he showed to me throughout the elaboration of this Diploma Thesis.

I would also like to thank Dr. George Karras who instructed me for this project and was willing to help, whenever problems arised. I am also grateful to the members of the Unmanned Aerial Vehicles department of the Control Systems Lab, who gave me useful advice and directions.

Last but not least, I would like to give special thanks to my parents, Lena and Ilias, who have always believed in me and supported me throughout my school and university years.

Modeling And System Identification of a multirotor UAV using simulation data

Diploma Thesis

Iliopoulou Antouanetta

Athens, March 2022

Abstract

Multirotor aerial vehicles have drawn a lot of attention in the recent years and their development has met great growth. The tasks that they are desired to complete are getting more and more complicated alongside, and the development of precise controllers is demanded. The design of such controllers requires accurate dynamic models with determined parameter values. This diploma thesis is concerned with the modeling of a multirotor UAV and the estimation of the model parameters using System Identification methods.
A detailed model is derived, that can describe the kinematic and dynamic behaviour of a multirotor vehicle, regardless of the number of rotors or their arrangement. Two specific types of platforms, an octorotor in X-configuration and a quadrotor in X-configuration, are then examined in more detail, and their simulation environments are presented. The linear least squares method is used for the identification of the parameters appearing in their models. Initially, the octorotor system is identified, using data collected from a simple MATLAB® model, where no controller is implemented. The results are satisfactory, since all parameters are precisely estimated. Following, an attempt to identify the parameters of the ROS model of a quadrotor is made. The process here is more challenging, since an autopilot system interferes with the openloop model, limiting the commands' capabilities, and the collection of data is affected by the simulated sensors' qualities. Sinusoidal signals of varying frequencies and amplitudes are designed and sent to the autopilot, in order to obtain frequency rich data from the simulated flights and estimate the parameters. A validation method is also specified as a way to test the accuracy of the estimated parameters. An acceptable estimation of most of the parameters is achieved, although some of the values may be suitable only for specific operating areas. Overall, the results are promising and provide a good first description of the model. However, there is certainly room for improvement, regarding the procedure and the techniques, aiming for an ever more accurate estimation of the parameters.

Μοντελοποίηση και Αναγνώριση συστήματος μη επανδρωμένου εναέριου οχήματος πολλαπλών ελίκων με χρήση δεδομένων προσομοίωσης

Διπλωματική Εργασία

Ηλιοπούλου Αντουανέττα

Αθήνα, Μάρτιος 2022

Τα μη επανδρωμένα εναέρια οχήματα (UAVs) έχουν προσελκύσει μεγάλο ενδιαφέρον τα τελευταία χρόνια και μεγάλη εξέλιξη έχει σημειωθεί γύρω από τη μελέτη και κατασκευή τους. Ταυτόχρονα, οι αποστολές που καλούνται να φέρουν εις πέρας γίνονται όλο και πιο περίπλοκες και η ενσωμάτωση ελεγκτών μεγαλύτερης ακρίβειας κρίνεται απαραίτητη. Ο σχεδιασμός τέτοιων ελεγκτών απαιτεί την γνώση των δυναμικών μοντέλων και των ακριβών τιμών των παραμέτρων τους. Η παρούσα διπλωματική εργασία ασχολείται με τη μοντελοποίηση ενός UAV πολλαπλών ελίκων και την εκτίμηση των παραμέτρων του μοντέλου με τη χρήση μεθόδων αναγνώρισης συστήματος.
Παρουσιάζεται, αρχικά, ένα λεπτομερές μοντέλο, το οποίο μπορεί να περιγράψει την κινηματική και δυναμική συμπεριφορά ενός οχήματος πολλαπλών ελίκων, ανεξάρτητα από τον αριθμό ή τη διάταξή τους. Στη συνέχεια, εξετάζονται λεπτομερέστερα δύο συγκεκριμένοι τύποι εναέριων οχημάτων, ένα οκτακόπτερο σε διάταξη X και ένα τετρακόπτερο σε διάταξη X, και παρουσιάζονται τα περιβάλλοντα προσομοίωσής τους. Η μέθοδος των γραμμικών ελαχίστων τετραγώνων χρησιμοποιείται για τον προσδιορισμό των παραμέτρων που εμφανίζονται στα μοντέλα των διαφορικών εξισώσεων. Αρχικά, γίνεται αναγνώριση του συστήματος του οκτακοπτέρου, χρησιμοποιώντας δεδομένα που συλλέχθηκαν από ένα απλό μοντέλο υλοποιημένο στην MATLAB®, χωρίς την χρήση ελεγκτή. Τα αποτελέσματα είναι ικανοποιητικά, καθώς όλες οι παράμετροι εκτιμώνται με ακρίβεια. Στη συνέχεια, γίνεται προσπάθεια αναγνώρισης των παραμέτρων του μοντέλου ενός τετρακοπτέρου, όπως αυτό έχει υλοποιηθεί με χρήση του ROS. Η διαδικασία σε αυτή την περίπτωση είναι πιο απαιτητική, καθώς ένα σύστημα αυτόματου πιλότου παρεμβαίνει στο μοντέλο ανοικτού βρόχου, περιορίζοντας τις δυνατότητες των εντολών, και επιπλέον, η συλλογή δεδομένων επηρεάζεται από τις ιδιότητες των προσομοιωμένων αισθητήρων. Ημιτονοειδή σήματα διαφορετικού πλάτους και συχνοτήτων σχεδιάζονται και στέλνονται στον αυτόματο πιλότο, προκειμένου να ληφθούν δεδομένα ποικίλων συχνοτήτων από τις προσομοιωμένες πτήσεις και να εκτιμηθούν οι παράμετροι. Ορίζεται επίσης μια μέθοδος επαλήθευσης και ελέγχου της ακρίβειας των εκτιμώμενων παραμέτρων. Μια καλή εκτίμηση των περισσότερων παραμέτρων πράγματι επιτυγχάνεται, αν και ορισμένες από τις τιμές φαίνεται να αντιστοιχούν σε συγκεκριμένες περιοχές λειτουργίας. Συνολικά, τα αποτελέσματα είναι αισιόδοξα και παρέχουν μια καλή πρώτη περιγραφή του μοντέλου. Ωστόσο, υπάρχουν σίγουρα περιθώρια βελτίωσης, όσον αφορά τη διαδικασία και τις τεχνικές που ακολουθήθηκαν, και μια ακόμη πιο ακριβής εκτίμηση των παραμέτρων μπορεί να τεθεί ως νέος στόχος.

Acronyms

ARX Autoregressive with Extra Input

CCW Counterclockwise

CW Clockwise

EKF Extended Kalman Filter

ESC Electronic Speed Command

FCU Flight Controller Unit

FIR Finite Impulse Response

GPS Global Positioning System

IIR Infinite Impulse Response

IMU Inertial Measurement Unit

LIDAR Light Detection and Ranging

NED North East Down frame

NTUA National Technical University of Athens

PWM Pulse Width Modulator

ROS Robot Operating System

SITL Software In The Loop

System ID System Identification

UAV Unmanned Aerial Vehicle

VTOL Vertical Take-Off and Landing

Contents

Chapter 1

Introduction

1.1 Background

Aerial robotics is a fast-growing field of robotics and UAVs have become widely developed and used in scientific, civilian and military applications[1][2]. An Unmanned Aerial Vehicle (UAV) is a remotely piloted or self-piloted aircraft which is operated through electronic input initiated by the flight controller or by an onboard autonomous flight management control system [3][4]. It can carry payloads and be equipped with multiple sensors, including cameras, Inertial Measurement Unit (IMU), LIDAR, GPS or other communication equipment in order to collect and transmit data in real time[2].
More specifically, a multicopter or multirotor, is a type of rotary-wings UAV [4], that has three or more propellers. It also has the ability of Vertical Take-Off and Landing (VTOL), hence no runway or launcher is required. Multirorots stand out for heir hovering capability and high maneuverability, and their originally small size which make them suitable and efficient for a variety of applications. As stated in [5], in the latest years, research has focused even to modify the classical multirotor designs to overcome their existence limitations.
Although multicopters have gain popularity only in the last decades, their history goes back in early 1900s when the first -although failed- attempts of rotating wing designs were made. The first successful quadcopter, designed by Marc Adman Kaplan, was flown in 1956, but little interest was initially shown due to its low performance abilities even for its time. In the following years the research and development of multicopters was limited for military applications, until the 1990s when the development of microcomputers and IMUs emerged and opened the way to build small multicopters, even for toy market purposes. Since 2005 the attention paid in multicopters has risen and more and more researchers have focused their work around their development. The progress made in the fields of sensors' data acquisition and estimation algorithms , navigation methods, remote control capabilities, fabrication and power systems has boomed the design and usage of UAVs.
UAVs' control is one of the main challenges, since they have an unstable nature, and the control technologies need to follow an accurate approach. A lot of autopilot systems have been developed [6], that can be used with a variety of UAVs' configura-

tions. However, as the applications of UAVs are expanding and the tasks, that they are desired to execute, become more and more complicated, the simplistic approach of the common autopilots becomes inadequate [7][8]. When UAVs need to make aggressive maneuvers or the need of accurate trajectory tracking at higher speed and in uncertain environments appears, more reliable and robust control systems need to be designed. These control systems require dynamic system models, the parameters of which need to be determined. Traditionally, these parameters can be estimated from first principle assumptions or by testing in wind tunnels, but these methods are usually time-consuming and costly. An alternative, to overcome these problems, is the System Identification (ID) procedure, through which a UAV's dynamic model can be determined from flight data. Although in the recent years there has been a little development of System ID techniques for multirotors as Hoffer et al.[9] present in their survey, this process still remains a challenge.

1.2 Purpose

The purpose of this thesis is to obtain a mathematical model of a multirotor UAV and estimate its parameters using system identification methods. The basic idea of this research is to define a procedure of system identification, that can later be applied in a real life octorotor that NTUA Control Systems Lab is equipped with. The initial thought was to study only an octorotor and estimate its parameters, but only a quadrotor simulator could be used under the circumstances. For this reason, a general model that can describe both quadrotors and octorotors is constructed and the identification method is applied in two different platforms; an open-loop MATLAB® octorotor model and an autopilot-equipped ROS/GAZEBO quadrotor model. Since the parameters appearing on the model are independent of the number of rotors, the method followed for the quadrotor simulation and System Identification, can also be employed for any octorotor platform equipped with autopilot.

1.3 Outline

This thesis is organized in 6 chapters, including the introduction.
In Chapter 2 the process of obtaining a model of a multirotor vehicle is described.
In Chapter 3 the basic movements of a multirotor are explained.
In Chapter 4 the configurations of the vehicles that are used in this thesis are presented, along with their simulation environments.
Chapter 5 deals with the problem of System identification. Methods and techniques are stated and then the two different platforms are used for parameter identification.
In Chapter 6 a summary-conclusion of the results is presented, along with suggestion for future work.

Chapter 2

Multirotor modeling

2.1 Coordinate systems

In order to describe the motion of a multirotor system in space, coordinates systems need to be defined. In this case the use of an inertial/earth-fixed frame as well as a body frame are needed.

The earth-fixed frame, represented here with the letter E, is attached in the surface of the earth and is supposed to have the features of an inertial frame. In this thesis the earth-frame is chosen so that the Z_E-axis points up whereas X_E-axis and Y_E-axis are tangential to the earth surface, as shown in Fig. (2.1.1).

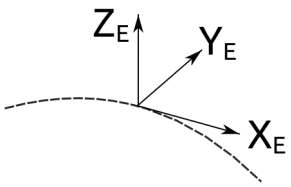
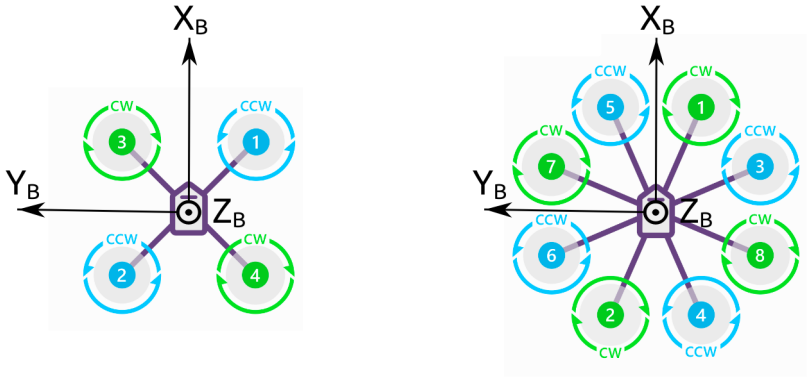


Figure 2.1.1: The earth-fixed coordinate system.

The body-fixed frame, represented here with the letter B, is attached in the center of gravity of the multirotor and it moves as the multirotor changes its location in space. The body-fixed frame is chosen so that the X_B-axis always points in the front of the multirotor, the Y_B-axis always to the left side and the Z_B-axis upwards. This configuration is not affected by the multirotor frame type or number of propellers. For better understanding, an illustration of the body frame is given in Fig. (2.1.2) for the two multirotor frames that will be examined further in this thesis; a quadrotor in X-configuration (2.1.2a) and an octorotor in X-configuration (2.1.2b). Their illustrations are adopted from [10]



(a) Quad-x frame (b) Octo-x frame

Figure 2.1.2: The body-fixed coordinate system in two different multirotor frames.

2.2 Kinematics

2.2.1 Definition of kinematic relations

The location of the UAV in space, as any other rigid body, is fully described by its **linear position** and **orientation**. The linear position $\mathbf{\Gamma}_E$ is defined as the position of the center of gravity of the UAV with respect to an inertial frame -here the earth-fixed frame-, as described in Eq. (2.2.1).

$$\mathbf{\Gamma}_E = [x_E \; y_E \; z_E]^T \tag{2.2.1}$$

The orientation $\mathbf{\Theta}_E$ is defined as the orientation of the body-fixed frame with respect to the earth-fixed frame, as described in Eq. (2.2.2).

$$\mathbf{\Theta}_E = [\phi_E \; \theta_E \; \psi_E]^T \tag{2.2.2}$$

The angles ϕ, θ and ψ are consistent with the ZYX *Euler angles* representation, which is preferred for the depiction of orientation in the aeronautical field, and are called *roll, pitch* and *yaw* respectively [11]. They represent a rotation of the earth-fixed frame about x-axis by angle ϕ ($R_x(\phi)$), about y-axis by angle θ ($R_y(\theta)$) and about z-axis by angle ψ ($R_z(\psi)$) (see Appendix A.1).
With respect to position and orientation, also **linear** and **angular velocities** should be defined. In this thesis the body-fixed frame is preferred for the depiction of the velocities as shown in the following equations.

$$\boldsymbol{\nu}_B = [u_B \; v_B \; w_B]^T \tag{2.2.3}$$

$$\boldsymbol{\omega}_B = [p_B \; q_B \; r_B]^T \tag{2.2.4}$$

2.2.2 Transformation between frames

Since not all terms are expressed in the same frame, a rotation matrix must be defined in order to convert vector coordinates between frames. The complete rotation matrix [1] can be computed as a composition of rotations as explained in Section 2.2.1.

[1] As abbreviations of sinα or cosα, the notations sα and cα are used.

$$R_{B\to E} = R_z(\psi)R_y(\theta)R_x(\phi) = \begin{bmatrix} c\psi c\theta & -s\psi c\phi + c\psi s\theta s\phi & s\psi s\phi + c\psi s\theta c\phi \\ s\psi c\theta & c\psi c\phi + s\psi s\theta s\phi & -c\psi s\phi + s\psi s\theta c\phi \\ -s\theta & c\theta s\phi & c\theta c\phi \end{bmatrix} \tag{2.2.5}$$

The inverse conversion of coordinates (from the body-fixed frame to the earth fixed-frame) is stated as:

$$R_{E\to B} = (R_{B\to E})^{-1} = (R_{B\to E})^T \tag{2.2.6}$$

2.2.3 Differential kinematics

Suffice to say, the orientation angles are not constant in time, so a relation between their derivatives and the angular velocity expessed in the body-fixed frame, $\boldsymbol{\omega}_B$ should be defined. The latter can be expressed as the sum of rotated angular velocities, as shown in Eq. (2.2.7).

$$\begin{aligned} \boldsymbol{\omega}_B &= \begin{bmatrix} \dot{\phi} \\ 0 \\ 0 \end{bmatrix} + R_x(\phi)^T \begin{bmatrix} 0 \\ \dot{\theta} \\ 0 \end{bmatrix} + [R_y(\theta)R_x(\phi)]^T \begin{bmatrix} 0 \\ 0 \\ \dot{\psi} \end{bmatrix} \\ &= \begin{bmatrix} \dot{\phi} \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 1 & 0 & 0 \\ 0 & c\phi & s\phi \\ 0 & -s\phi & c\phi \end{bmatrix} \begin{bmatrix} 0 \\ \dot{\theta} \\ 0 \end{bmatrix} + \begin{bmatrix} c\theta & 0 & -s\theta \\ s\phi s\theta & c\phi & s\psi c\theta \\ c\phi s\theta & -s\phi & c\psi c\theta \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ \dot{\psi} \end{bmatrix} \\ &= \begin{bmatrix} 1 & 0 & -s\phi \\ 0 & c\phi & s\phi c\theta \\ 0 & -s\phi & c\theta c\phi \end{bmatrix} \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} \end{aligned} \tag{2.2.7}$$

Using Eq. (2.2.7), the transfer matrix $T_{E\to B}$ can be defined:

$$T_{E\to B} \triangleq \begin{bmatrix} 1 & 0 & -s\phi \\ 0 & c\phi & s\phi c\theta \\ 0 & -s\phi & c\theta c\phi \end{bmatrix} \tag{2.2.8}$$

The inverse transformation, relating the time derivatives of the Euler angles to the multirotor's angular velocity is defined as:

$$T_{B\to E} = (T_{E\to B})^{-1} = \begin{bmatrix} 1 & s\phi t\theta & c\phi t\theta \\ 0 & c\phi & -s\phi \\ 0 & s\phi/c\theta & c\phi/c\theta \end{bmatrix} \tag{2.2.9}$$

2.2.4 Final kinematics differential equations

Summing all the kinematic relations, two differential equations are derived, that are part of the multirotor model as will be described in Section 2.4.

$$\dot{\boldsymbol{\Gamma}}_E = R_{B\to E}\,\boldsymbol{\nu}_B \tag{2.2.10}$$

$$\dot{\boldsymbol{\Theta}}_E = T_{B\to E}\,\boldsymbol{\omega}_B \tag{2.2.11}$$

2.3 Rigid body Dynamics

In this chapter the motion dynamics of a multirotor are examined, starting from the general rigid body motion equations. Subsequently, all external loads are computed and explained until the full dynamic model is derived.

2.3.1 Rigid body dynamic equations

The dynamic model of the multirotor is based on rigid body mechanics, as expressed in the Newton-Euler's equations.
Before we continue to describe the dynamic equations of a multirotor it is sufficient to explain how the time derivative of a vector $\mathbf{a}$ can be described with respect to a non-inertial frame.

$$\left.\frac{d\mathbf{a}}{dt}\right|_I = \left.\frac{d\mathbf{a}}{dt}\right|_B + \boldsymbol{\omega} \times \mathbf{a}\,, \tag{2.3.1}$$

where ω is the rotational velocity vector of the rotating coordinate system. In this case $\omega = \omega_B$, the angular velocity of the body-fixed frame.
According to Newton's second law, expressed in an inertial frame, the dynamic equations of a rigid body are:

$$\mathbf{F} = \mathbf{m}\left.\frac{d\boldsymbol{\nu}}{dt}\right|_I \tag{2.3.2}$$

$$\boldsymbol{\tau} = \left.\frac{d\boldsymbol{I}\boldsymbol{\omega}}{dt}\right|_I \tag{2.3.3}$$

In Eq. (2.3.2), $\mathbf{F}$ is the sum of external forces applied in the center of gravity of the rigid body, $\mathbf{m}$ is the mass diagonal matrix and $\boldsymbol{\nu}$ is the center of gravity velocity vector. In Eq. (2.3.3), $\boldsymbol{\tau}$ is the sum of external torques, $\mathbf{I}$ is the inertia tensor and $\boldsymbol{\omega}$ is the angular velocity, all with respect to the center of gravity.
According to Eq. (2.3.1) the same equations of motion can be described in the body-fixed frame as shown below.

$$\mathbf{F}_B = \mathbf{m}\dot{\boldsymbol{\nu}}_B + \boldsymbol{\omega}_B \times \mathbf{m}\boldsymbol{\nu}_B \tag{2.3.4}$$

$$\boldsymbol{\tau}_B = \mathbf{I}\dot{\boldsymbol{\omega}}_B + \boldsymbol{\omega}_B \times \mathbf{I}\boldsymbol{\omega}_B \tag{2.3.5}$$

2.3.2 External loads

In order to complete the dynamic model, it is now important to define the external forces and torques applied in the multirotor system, described in the body-fixed frame. The different categories of the external loads are stated below and will be examined separately in order to abduct the desired expressions.

- Thrust and Drag
- Gravity
- Gyroscopic torque
- Aerodynamic effects

Thrust and Drag

The main forces and moments applied on the multirotor during flight are related to the thrust and drag generated by the rotors. When the blades of the rotors rotate they generate an upward thrust aligned with the rotor axis, as well as a torque with respect to the rotor axis, that can be described using a combination of momentum and blade element theory [12]. According to Fay [13], the thrust vector generated by rotor i in a multirotor system can be expressed:

$$\mathbf{T}_i = c_T \rho A_{r_i} r_i^2 \omega_i^2 \mathbf{e}_i^B = K_T \omega_i^2 \mathbf{e}_i^B \tag{2.3.6}$$

In Eq. (2.3.6), c_T is the thrust constant, ρ is the air density, A_{r_i} is the rotor disk area, r is its radius and ω_i is the angular velocity. $\mathbf{e}_i^B$ is the normalized rotor axis vector for rotor i expressed in the body-fixed frame, which for not tilted rotors only has a z-axis component, thus $\mathbf{e}_i^B = [0\ 0\ 1]^T$. For simplicity all the steady parameters can be replaced with $K_T > 0$ which will now be referred as the trust constant.
The torque of rotor i due to the rotor drag is described[13]:

$$\boldsymbol{\tau}_i = c_Q \rho A_{r_i} r_i^3 sgn(\omega_i) \omega_i^2 \mathbf{e}_i^B = K_Q sgn(\omega_i) \omega_i^2 \mathbf{e}_i^B \tag{2.3.7}$$

In Eq. (2.3.7), c_Q is the torque constant and similarly to thrust, all the steady parameters are replaced with $K_Q > 0$, the new torque constant. The sign of each rotor's torque depends on the direction of the motor rotation as described in the sign function:

$$sgn(\omega_i) = \begin{cases} + & \text{if rotor i CW} \\ - & \text{if rotor i CCW} \end{cases} \tag{2.3.8}$$

Since the thrust forces are not applied in the center of gravity, they also generate a torque, which is given by the cross product of the relative with the center of gravity position vector of each rotor and the force vector of Eq. (2.3.6).

$$\boldsymbol{\tau}_{T_i} = \mathbf{r}_i \times \mathbf{T}_i \tag{2.3.9}$$

Let the position vector be denoted as:

$$\mathbf{r}_i = [r_{x_i}\ r_{y_i}\ r_{z_i}]^T, \tag{2.3.10}$$

where r_{x_i}, r_{y_i} and r_{z_i} are the rotor i distances of the body-fixed frame origin B in axes x,y and z respectively.
Summing the above equations, the total force and torques pertained to the rotor movement are:

$$\boldsymbol{F}_{rotors_B} = \sum_{i=1}^{n} \boldsymbol{T}_i \tag{2.3.11}$$

$$\boldsymbol{\tau}_{rotors_B} = \sum_{i=1}^{n} \boldsymbol{\tau}_i + \sum_{i=1}^{n} \boldsymbol{\tau}_{T_i}\ , \tag{2.3.12}$$

where n is the number of rotors, depending on the multirotor type.

Gravity

The force of gravity is always aligned with the Z axis of the earth-fixed frame and is applied in the multirotor center of gravity.

$$\mathbf{F}_{\mathbf{g}_E} = m \begin{bmatrix} 0 \\ 0 \\ -g \end{bmatrix} \tag{2.3.13}$$

Here, g is the gravitational acceleration constant to which the value 9.81 m/s^2 is assigned and will not be mentioned again in this thesis. Since all forces need to be described in the body-fixed frame, the inverse rotation matrix should be used in order to change the reference frame of the gravitational force.

$$\mathbf{F}_{\mathbf{g}_B} = mR_{E\to B} \begin{bmatrix} 0 \\ 0 \\ -g \end{bmatrix} = mg \begin{bmatrix} s\theta \\ -s\phi c\theta \\ -c\phi c\theta \end{bmatrix} \tag{2.3.14}$$

Gyroscopic torques

Since each rotor is rotating, a torque defined according to Newton's law will be applied in the origin of a frame R_i in the center of gravity of each motor. The R_i frame is aligned with the rotor axis and is fixed on the multirotor, so it has the same orientation with the body-fixed frame, implying that $\omega_R = \omega_B$ and $R_{B\to R_i} = R_{R_i\to B} = I$.
Expressed in an inertial frame this torque is:

$$\boldsymbol{\tau}_{gyr_{i_I}} = \frac{dJ_{rot}\omega_i}{dt}, \tag{2.3.15}$$

where J_{rot} is the inertia tensor of each motor with respect to the R_i frame.
Using Eq. (2.3.1) this torque can be now expressed in the R_i frame:

$$\boldsymbol{\tau}_{gyr_{i_R}} = \frac{dJ_{rot}\omega_i}{dt} + \boldsymbol{\omega}_R \times J_{rot}\omega_i = \frac{dJ_{rot}\omega_i}{dt} + \boldsymbol{\omega}_B \times J_{rot}\omega_i, \tag{2.3.16}$$

The term $\dfrac{dJ_{rot}\omega}{dt}$ is insignificant compared with the other quantities and can be assumed as zero, so the gyroscopic torque can be simplified as:

$$\boldsymbol{\tau}_{gyr_{i_R}} = \boldsymbol{\omega}_B \times J_{rot}\omega_i \tag{2.3.17}$$

Considering that all loads must be expressed in the body frame the gyroscopic torque of rotor i should also be converted in body-fixed coordinates using $R_{R_i\to B}$.

$$\boldsymbol{\tau}_{gyr_{B_i}} = R_{R_i\to B}\boldsymbol{\tau}_{gyr_{R_i}} = I\boldsymbol{\tau}_{gyr_{R_i}} = \boldsymbol{\omega}_B \times J_{rot}\omega_i \tag{2.3.18}$$

The total gyroscopic torque is the sum of the torques applied on each rotor:

$$\boldsymbol{\tau}_B = \sum_{i=1}^{n} \boldsymbol{\tau}_{gyr_{B_i}} \tag{2.3.19}$$

Aerodynamic effects

There are a lot aerodynamic effects that influence the motion of a multirotor, most of which are minor compared to the dominant dynamics. However, *blade flapping* and *induced drag* are of more significance, as noted in [14]. The former are caused due to the flexing, whereas the latter are associated with the rigidity of the rotor. These effects induce forces in the x and y-axis of the multirotor, the directions that are genuinely underactuated by the dynamics. Although terms associated with blade flapping are affected by the rotational velocity of the rotors and thus are not constant during flight, in this thesis such variations are considered insignificant. For simplicity, both blade flapping and induced drag effects are expressed as combined constants multiplying the linear x- and y-velocities of the multirotor. That concludes to the following expression for the aerodynamic effects:

$$\mathbf{F}_{aero_B} = -\begin{bmatrix} Kd_u & 0 & 0 \\ 0 & Kd_v & 0 \\ 0 & 0 & 0 \end{bmatrix} \boldsymbol{\nu}_B \tag{2.3.20}$$

$$\boldsymbol{\tau}_{aero_B} = \mathbf{0} \tag{2.3.21}$$

2.3.3 Dynamic model

As soon as all external forces are defined Equations (2.3.4) and (2.3.5) can be rewritten, and given that all vectors are now described in the body-fixed frame from this point no index **B** will be used.

$$\mathbf{m}\dot{\boldsymbol{\nu}} = -\boldsymbol{\omega} \times \mathbf{m}\boldsymbol{\nu} + \mathbf{F_g} + \mathbf{F}_{rotors} + \mathbf{F}_{aero} \tag{2.3.22}$$

$$\mathbf{I}\dot{\boldsymbol{\omega}} = -\boldsymbol{\omega} \times \mathbf{I}\boldsymbol{\omega} + \boldsymbol{\tau}_{rotors} + \boldsymbol{\tau}_{gyr} \tag{2.3.23}$$

2.4 Complete state-space model

The complete model describing the motion of the multirotor in space requires both the dynamic (2.3.22 , 2.3.23) and the kinematic (2.2.10 , 2.2.11) differential equations.
Here, it is necessary to mention that the inertia tensor I is considered diagonal meaning that the mass distribution is symmetric with respect to the body-fixed coordinate system. The expanded matrix form of these equations is stated below. Note that the number of rotors does not yet need to be defined.

$$\begin{bmatrix}\dot{u}\\ \dot{v}\\ \dot{w}\end{bmatrix} = \begin{bmatrix}rv-qw\\ pw-ru\\ qu-pv\end{bmatrix} + \begin{bmatrix}gs\theta\\ -gs\phi c\theta\\ -gc\phi c\theta\end{bmatrix} + \begin{bmatrix}0\\ 0\\ K_T/m\sum_{i=1}^{n}\omega_i^2\end{bmatrix} - \begin{bmatrix}Kd_u/m & 0 & 0\\ 0 & Kd_v/m & 0\\ 0 & 0 & 0\end{bmatrix}\begin{bmatrix}u\\ v\\ w\end{bmatrix} \tag{2.4.1}$$

$$\begin{bmatrix}\dot{p}\\ \dot{q}\\ \dot{r}\end{bmatrix} = \begin{bmatrix}qr(I_{yy}-I_{zz})/I_{xx}\\ rp(I_{zz}-I_{xx})/I_{yy}\\ pq(I_{xx}-I_{yy})/I_{zz}\end{bmatrix} + \begin{bmatrix}K_T/I_{xx}\sum_{i=1}^{n} r_{y_i}\omega_i^2\\ K_T/I_{yy}\sum_{i=1}^{n} r_{x_i}\omega_i^2\\ K_Q/I_{zz}\sum_{i=1}^{n} sgn(\omega_i)\omega_i^2\end{bmatrix} + \begin{bmatrix}-qJ_{rot}/I_{xx}\sum_{i=1}^{n} sgn(\omega_i)\omega_i\\ pJ_{rot}/I_{yy}\sum_{i=1}^{n} sgn(\omega_i)\omega_i\\ 0\end{bmatrix} \tag{2.4.2}$$

$$\begin{bmatrix}\dot{x}\\ \dot{y}\\ \dot{z}\end{bmatrix} = \begin{bmatrix}c\psi c\theta & -s\psi c\phi + c\psi s\theta s\phi & s\psi s\phi + c\psi s\theta c\phi\\ s\psi c\theta & c\psi c\phi + s\psi s\theta s\phi & -c\psi s\phi + s\psi s\theta c\phi\\ -s\theta & c\theta s\phi & c\theta c\phi\end{bmatrix}\begin{bmatrix}u\\ v\\ w\end{bmatrix} \tag{2.4.3}$$

$$\begin{bmatrix}\dot{\phi}\\ \dot{\theta}\\ \dot{\psi}\end{bmatrix} = \begin{bmatrix}1 & s\phi t\theta & c\phi t\theta\\ 0 & c\phi & -s\phi\\ 0 & s\phi/c\theta & c\phi/c\theta\end{bmatrix}\begin{bmatrix}p\\ q\\ r\end{bmatrix} \tag{2.4.4}$$

These four subsystems compose a 12×12 non-linear system of equations, where the state vector is $X = [u\ v\ w\ p\ q\ r\ x\ y\ z\ \phi\ \theta\ \psi]^T$ and the input vector is $U = [\omega_1\ \omega_2\ ...\ \omega_n]^T$.
When an input vector is provided, the states of the the multirotor can be then computed using a differential equation solver.

2.4.1 Linearization in hovering mode

For completeness, a linear approach of the model described in Equations (2.4.1-2.4.4) will now be presented. The linear equations are especially important when an attitude controller needs to be build, controlling the orientation of the multirotor [15].
For a non-linear system in the form $\dot{X} = f(X, U)$, the linearized system using the Taylor series expansion will be:

$$\dot{X}^* = AX^* + BU^*, \tag{2.4.5}$$

where:

$$A = \left.\frac{\partial f}{\partial X}\right|_{Xe} \ , B = \left.\frac{\partial f}{\partial U}\right|_{Ue}$$

$$X^* = X - X_e \ , U^* = U - U_e$$

The equilibrium point will be that of hovering in a random position $\Gamma = (x, y, z)$ with any orientation ψ with respect to the earth-fixed frame. For an equilibrium-stationary point the time derivative of the states is always zero, so $\dot{X}_e = f(X_e, U_e) = 0$. The equilibrium input velocity, can be computed by substituting X_e in the latter mathematical equation, and under the notion that all rotor angular velocities are equal while hovering, it is resulted that:

$$\omega_{i_e} = \sqrt{\frac{mg}{nK_T}}, \tag{2.4.6}$$

where n is the number of rotors.
According to the above, the equilibrium points are:

$$X_e = [0\ 0\ 0\ 0\ 0\ 0\ x\ y\ z\ 0\ 0\ \psi]^T \tag{2.4.7}$$

$$U_e = [\omega_{1_e}, \omega_{2_e}, ..., \omega_{n_e}]^T \tag{2.4.8}$$

Note here, that aerodynamic effects are of minor importance in the linearized system and although they have a linear relation with the states are not taken into consideration. The resulting linear system is:

$$\dot{u} = g\theta \tag{2.4.9}$$

$$\dot{v} = -g\phi \tag{2.4.10}$$

$$\dot{w} = -g + \frac{F}{m} \tag{2.4.11}$$

$$\dot{p} = \frac{\tau_x}{I_{xx}} \tag{2.4.12}$$

$$\dot{q} = \frac{\tau_y}{I_{yy}} \tag{2.4.13}$$

$$\dot{r} = \frac{\tau_z}{I_{zz}} \tag{2.4.14}$$

$$\dot{x} = cos\psi u - sin\psi v \tag{2.4.15}$$

$$\dot{y} = sin\psi u + cos\psi v \tag{2.4.16}$$

$$\dot{z} = w \tag{2.4.17}$$

$$\dot{\phi} = p \tag{2.4.18}$$

$$\dot{\theta} = q \tag{2.4.19}$$

$$\dot{\psi} = r \tag{2.4.20}$$

were $F, \tau_x, \tau_y, \tau_z$ are the force and torques due to rotor movement as explained in Eq. (2.3.11) and (2.3.12).

Chapter 3

Basic movements

In this section the basic control commands of a multirotor will be explained as seen in [16]. Let us begin by noting that the movement of the multirotor is controlled by modifying the angular velocities of the rotors. The variation of the rotation rates can exert forces and torques on the UAV that will affect its states. For better understanding of these commands an octorotor in x-configuration will be used for illustrations. In each case a deviation from a hovering state ($\omega = \omega_H$) will be presented.

3.1 Throttle

The throttle command is related to the vertical movement of the multirotor, with respect to the $x_B y_B$ plane. It is achieved by increasing or decreasing the rotational rate of the rotors by the same amount, in order to exert only vertical thrust-forces, and all correlated torques to be zero. When the multirotor is flying horizontally, throttle command makes the vehicle moving upward or downward with respect to the inertial frame. The speed of the rotors in this case, for a positive throttle command should be $\omega_i = \omega_H + \delta\omega$, as shown in Fig. (3.1.1).

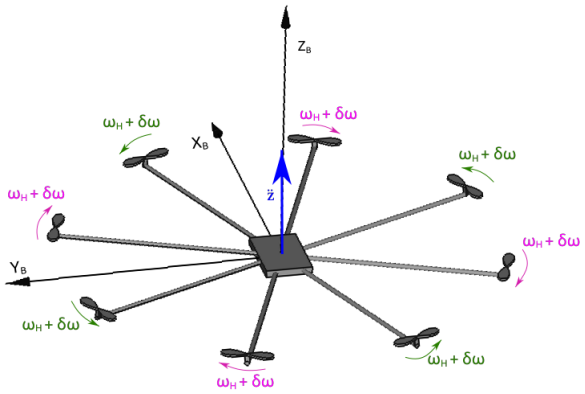


Figure 3.1.1: Illustration of positive throttle command.

3.2 Roll

The roll command is related to the multirotor rotation around the x_B axis, which is accomplished by generating torques with respect to that. More specifically, for a positive command, the angular velocity of the rotors on the left side of x-axis ($y > 0$), should be increased by $\delta\omega$, whereas the angular velocity of the rotors on the right side of x-axis ($y < 0$) should be decreased by the same rate. That way the total vertical thrust is kept constant.

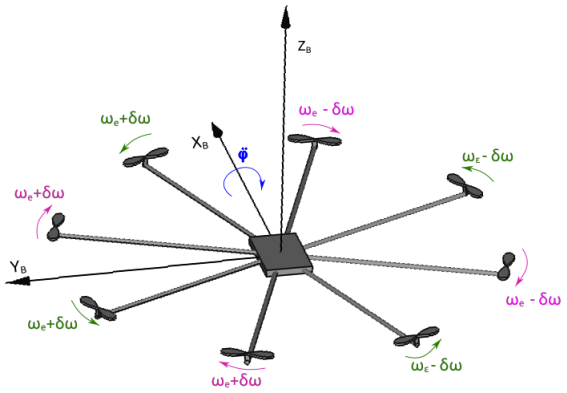


Figure 3.2.1: Illustration of positive roll command.

3.3 Pitch

The pitch command is associated with the rotation around the y_B axis, and is accomplished similarly with the roll command. In this case, for a positive pitch command, the rate of rotors in $x > 0$ is decreased and of those in $x < 0$ is increased, as shown in Fig. (3.3.1).

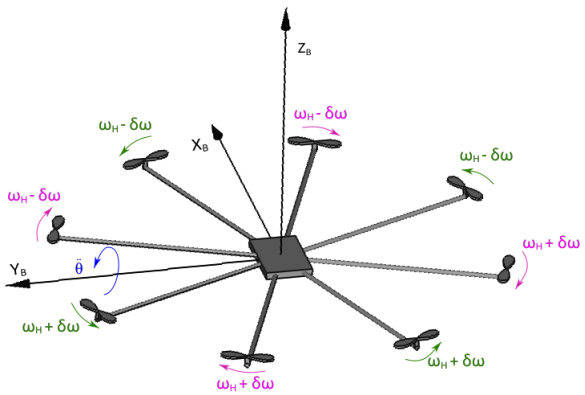


Figure 3.3.1: Illustration of positive pitch command.

For both roll and pitch commands, we may also note, that for any other multirotor configuration, if there are rotors aligned with the x_B (for roll) or y_B (for pitch),

their rates remains unaltered.

3.4 Yaw

This command is related with rotation around the z_B axis. The torques with respect to z-axis are associated with the rotor's drag, as it is already explained in the model, thus yaw command is different from the pitch and roll commands. For a positive yaw rate the rate of Clockwise (CW) rotating rotors must be increased, while the rate of Counterclockwise (CCW) rotating rotors must be decreased. In that way, the overall torque is unbalanced and the vehicle turns on itself around z. As in the above cases, the total vertical thrust is kept constant so that the vehicle maintain its altitude.

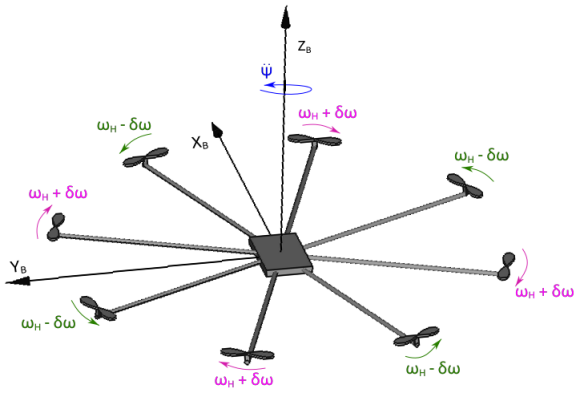


Figure 3.4.1: Illustration of positive yaw command.

Chapter 4

Platforms examined

In this thesis two different configurations of multirotors are examined. They are implemented and simulated in different environments as well. The first application refers to an octorotor using MATLAB® environment while the second deals with a ROS model of a quadrotor simulated within GAZEBO.

4.1 Octorotor using MATLAB environment

An octorotor in X-configuration is used in this case. The arrangement of this vehicle is shown in Fig. (4.1.1). The number for each rotor is assigned following the ArduPilot motor order, found in the official documentation[10]. For the simulations a MATLAB® representation of the model ((2.4.1)-(2.4.4)) is used, where the exact number of rotors ($n = 8$) is now set.

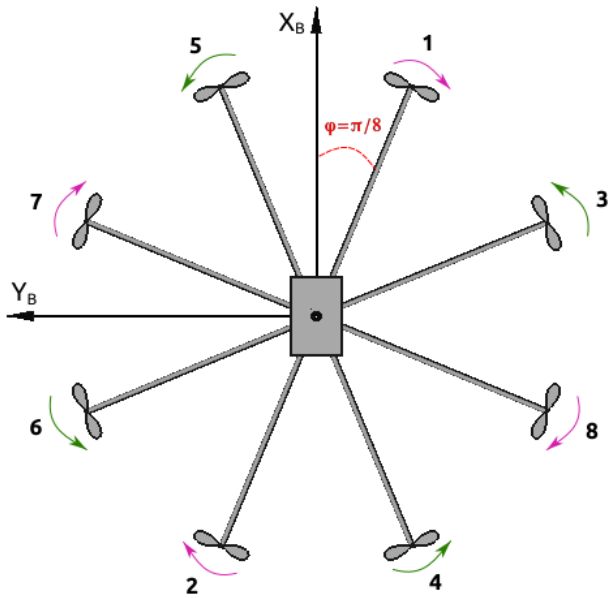


Figure 4.1.1: An octorotor in X-configuration.

Assuming a rotational symmetry for the propellers, meaning that the angle between two consecutive arms is $\pi/4$, the distance vectors r_i can be expressed as:

$$\begin{array}{l} r_1 = d\,[\quad cos\phi \quad -sin\phi \quad 0]^T \\ r_2 = d\,[\quad -cos\phi \quad sin\phi \quad 0]^T \\ r_3 = d\,[\quad sin\phi \quad -cos\phi \quad 0]^T \\ r_4 = d\,[\quad -cos\phi \quad -sin\phi \quad 0]^T \\ r_5 = d\,[\quad cos\phi \quad sin\phi \quad 0]^T, \\ r_6 = d\,[\quad -sin\phi \quad -cos\phi \quad 0]^T \\ r_7 = d\,[\quad -sin\phi \quad cos\phi \quad 0]^T \\ r_8 = d\,[\quad -sin\phi \quad -cos\phi \quad 0]^T \end{array} \tag{4.1.1}$$

where $\phi = pi/8$ and d is the length of each arm.

Suffice to say, the parameters of the model need to be defined for the simulations to run. The unknown parameters for this experiment are assumed as:

$$\begin{array}{rcc} d & = & 0.4\ m \\ m & = & 3.0\ kg \\ K_{d_u} & = & 0.3 \\ K_{d_v} & = & 0.3 \\ K_T & = & \text{2.2e-5}\ Ns^2 \\ K_Q & = & \text{4.5e-7}\ Nms^2 \\ I_{xx} & = & 0.109\ kg\,m^2 \\ I_{yy} & = & 0.108\ kg\,m^2 \\ I_{zz} & = & 0.208\ kg\,m^2 \\ J_{rot} & = & \text{2.0e-5}\ kg\,m^2 \end{array}$$

The above values are similar with ones found in bibliography for octorotor vehicles, ensuring that the order of magnitude is that of real values.
With all the parameters defined the model can be now simulated in MATLAB®. Note that the model is open loop, without any controller, so the user must assign values directly to the input vector, without any means to predict or predefine the states. For that reason it is useful to find the value of the hovering angular velocity and use it as a reference value in the simulations. The relation stated in Eq. (2.4.6) is used, substituting the values for this octorotor and thus:

$$\omega_{i_e} = \sqrt{\frac{mg}{nK_T}} = \sqrt{\frac{3.0 \cdot 9.81}{8 \cdot \text{2.2e-5}}} = 408.9204 \ \text{rad/s} \tag{4.1.2}$$

4.1.1 Basic commands simulation

For simulation requirements, the model is represented in MATLAB® as a function and an `ode45` solver is used to solve the system of differential equations. The vector $U = [\omega_1\ \omega_2\ \omega_3\ \omega_4\ \omega_5\ \omega_6\ \omega_7\ \omega_8]^T$ is an input for the solver, and the state vector X will be derived numerically. As a first test of the model the basic movements are performed. The thrust command is the only one that can be accurately achieved without the use of a controller. When the rest of the commands are implemented, in spite the fact that they concern only the behaviour of the angular terms, they simultaneously exert forces in other axes, that cannot be regulated without a controller. However, the appropriate alterations of the rotor rates will be applied as described, in order to observe the torques produced in the contested axes.
Having the hovering velocity of this vehicle computed, simulations are executed, initializing the UAV stationary at $z = 5$m.

- Throttle: The UAV remains in its original position for 2 seconds and then all rotor rates are increased to 420 rad/s for 3 seconds.

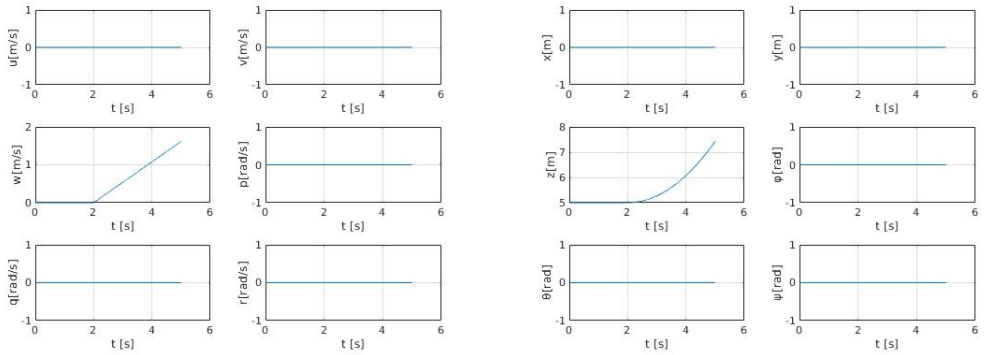


Figure 4.1.2: The state response with a throttle command.

- Roll: The UAV remains in its original position for 2 seconds and then the rates are altered by $\delta\omega = 2$rad/s according to (3.2.1) for 1 second. In this

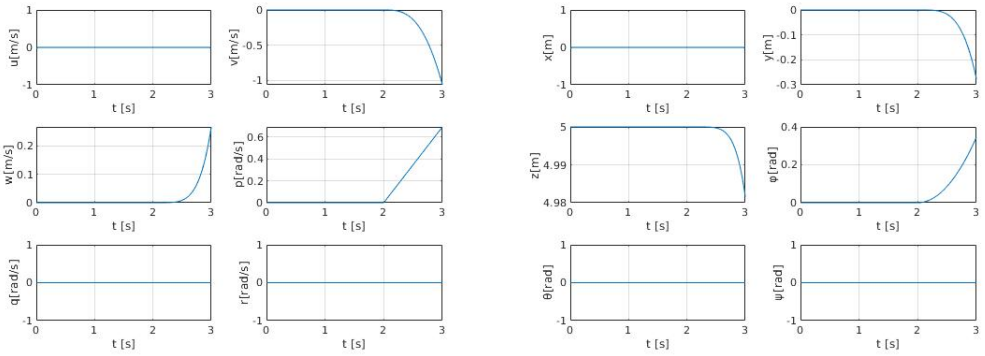


Figure 4.1.3: The state response with a roll command.

case, while a movement in y and z axes is caused as well, it is clear that a torque around x-axis is exerted, as desired.

- Pitch: The UAV remains in its original position for 2 seconds and then the rates are altered by $\delta\omega = 2\text{rad/s}$ according to (3.3.1) for 1 second. The generation of a torque around y axis is present here.

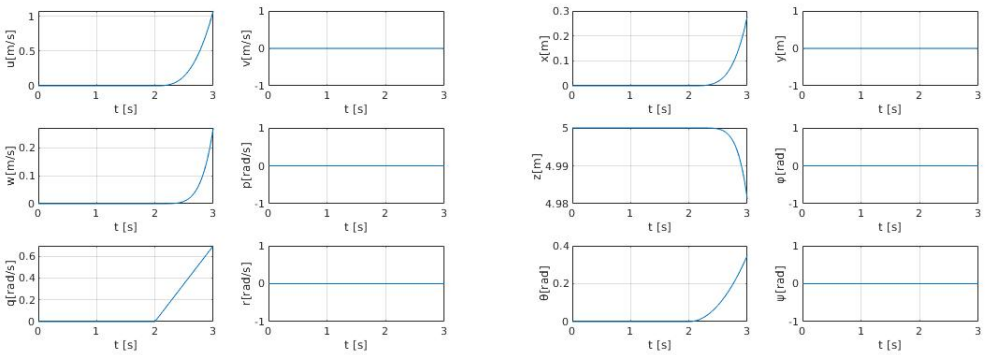


Figure 4.1.4: The state response with a pitch command.

- Yaw: After 2 seconds in its initial position, the CW rotors' rates are increased and the CCW rotors' rates are decreased by 10 rad/s, for the next 1 second.

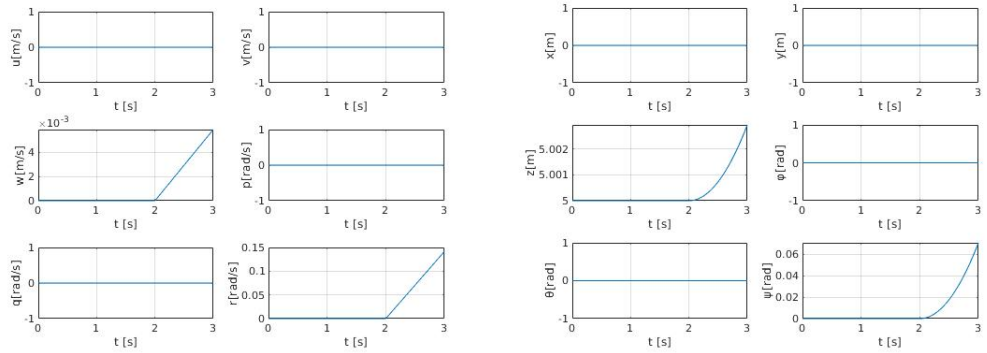


Figure 4.1.5: The state response with a yaw command.

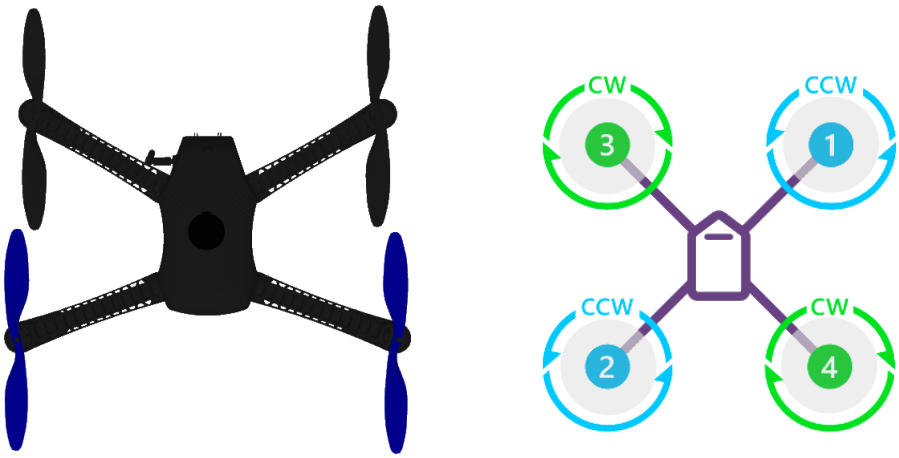
Again, the yaw movement is clear in this simulation, although the vehicle cannot precisely maintain its hovering position.

4.2 Quadrotor in ROS-GAZEBO

In order to overcome the lack of a controller in the MATLAB® model and avoiding building one, a ready ROS model of a quadrotor in X-configuration simulated in GAZEBO environment will be used, as built by NTUA Control Systems Lab [17]. It also integrates ArduCopter and MAVROS communication.

4.2.1 Model and Known Parameters

The ROS model included is an Iris quadrotor. An illustration of the model is presented in Fig. (4.2.1a), along with the rotors' number and direction of rotation (Fig 4.2.1b). Some of its features and parameters can be found in the configuration



(a) Visual model (b) Details about the rotors.

Figure 4.2.1: The iris model used in simulations.

files, but they do not necessarily coincide with the parameters used in the state space model. The position of each rotor with respect to the center of mass is explicitly issued and stated below.

$$\begin{aligned} r_1 &= [\ 0.13 \quad -0.22 \quad 0.023]^T\ m \\ r_2 &= [\ -0.13 \quad 0.2 \quad 0.023]^T\ m \\ r_3 &= [\ 0.13 \quad 0.22 \quad 0.023]^T\ m \\ r_4 &= [\ -0.13 \quad -0.2 \quad 0.230]^T\ m \end{aligned}$$

Since the model is assembled from different components, in order to construct the full entity, the mass of the vehicle is computed as a sum of the components' masses and the inertia tensor is computed by using parallel axes theorem to combine the inertias of the different components. The inertia of each rotor is also given in the model. All these values are stated below.

$$m = 1.9\ Kg$$
$$Jrot = 1.67604\text{e-}4\ Kgm^2$$
$$I = \begin{bmatrix} 0.0127 & 0 & 0 \\ 0 & 0.0178 & 0 \\ 0 & 0 & 0.0242 \end{bmatrix} Kgm^2$$

The calculation of thrust and torques associated with the rotors is accomplished using Lift and Drag Plugins, so the trust and torque constants are not directly inserted in the model. They are actually related to aerodynamic polar diagrams and their values may show fluctuations.

4.2.2 Autopilot

The autopilot integrated in the model is an open-source software [18] provided by Ardupilot. The version used here is Copter 4.1.0, combined with Software In The Loop (SITL) mimicking the Pixhawk hardware [19]. SITL is a build of the autopilot code that allows us to run Copter without any hardware. When running in SITL the sensor data comes from a flight dynamics model in a flight simulator.
A basic overview of Ardupilot's autopilot can be seen in Fig.(4.2.2), which presents the basic functional operation.

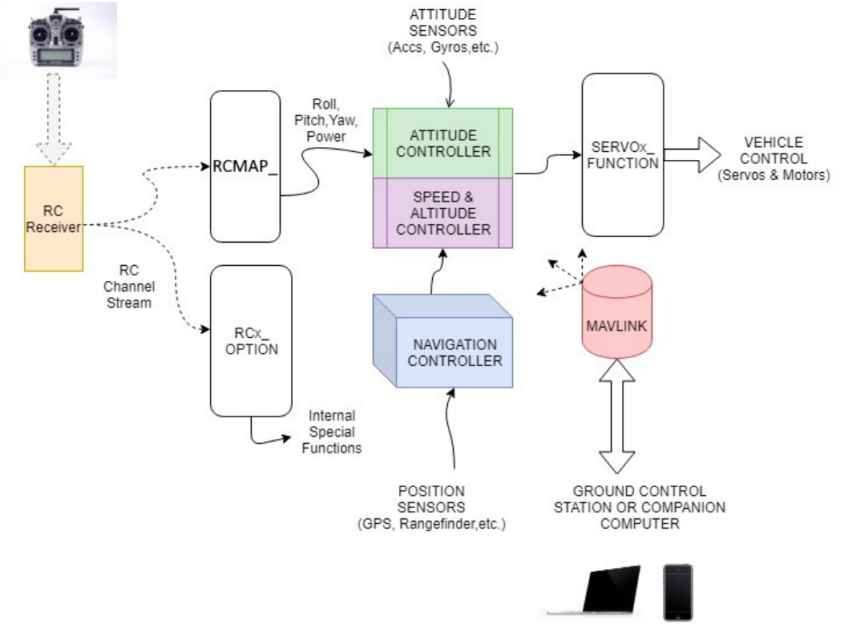


Figure 4.2.2: Simple diagram of the basic functional operation of Ardupilot

As most autopilots used in aeronautical applications, the control scheme is composed of two controller loops. The outer loop includes the position controller, witch calculates position, velocity and acceleration errors, computes the desired throttle and feeds target angles to the inner loop, which then calculates attitude errors and gives the appropriate commands to the motors.

The position controller has separate interfaces for horizontal and vertical control. In Fig. (4.2.3) the block diagram of the xy plane control is presented. A similar scheme is used for the z axis control giving a throttle commands to the motors.

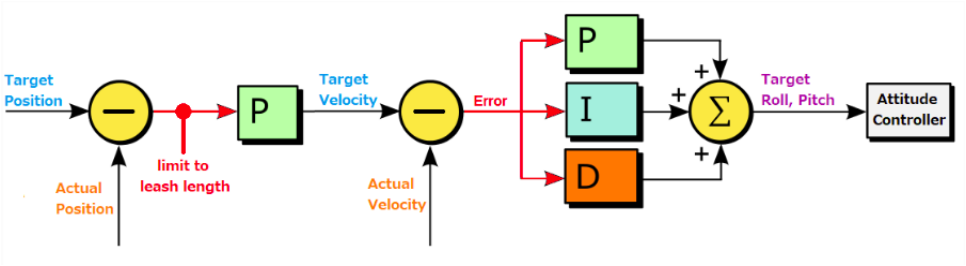


Figure 4.2.3: A block diagram of the xy plane scheme of the position controller.

The attitude controller scheme is described in Fig (4.2.4), where a simplified block diagram with regard to each axis is presented.

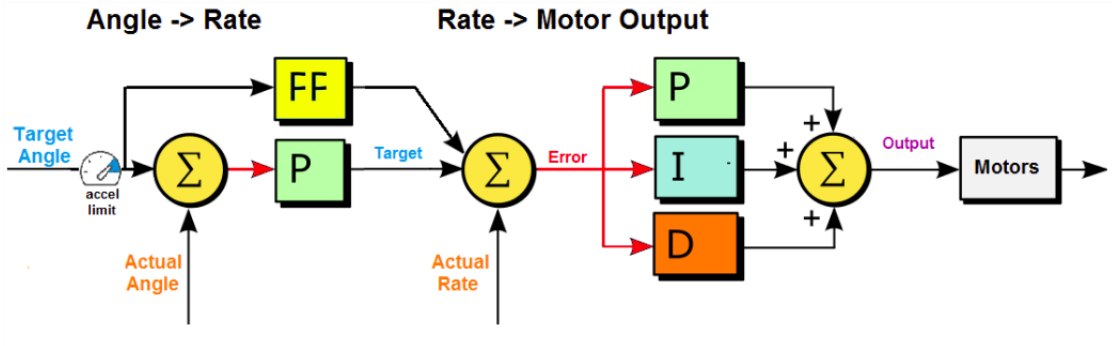


Figure 4.2.4: A block diagram of attitude control scheme for each axis.

The figures of this subsection are adopted from the official Ardupilot documentation [10].

4.2.3 Sensors and data acquisition

There is a variety of sensors simulated during flight that can provide measurements of different physical properties. The basic instrument is the IMU, that uses an accelerometer and a gyroscope.

- Accelerometer
  The accelerometer is a device that measures accelerations in the body-fixed frame relative to free fall. This means that when the accelerometer is stationary will perceive the counterforce of gravity, thus measuring an upward acceleration equal to g.
  The equation describing what an accelerometer mounted in the UAV's center of mass measures is:
  $$a_{acc_B} = a_{ext_B} - R_{E\to B} a_{g_E} \tag{4.2.1}$$

- Gyroscope
  A gyroscope is a device that can measure angles or angular rates. ArduPilot software integrates a rate gyroscope that measures the angular velocities expressed in the body-fixed frame.

The vehicle position, velocity and orientation are estimated based on rate gyroscopes, accelerometer, compass, GPS, airspeed and barometric pressure measurements with the use of an Extended Kalman Filter (EKF) algorithm [20, chapter 8]. The advantage of the EKF is that by fusing all available measurements it is better able to reject measurements with significant errors. This makes the vehicle less susceptible to faults that affect a single sensor.
The angular rates of the rotors are not directly estimated by the autopilot. Actually, Electronic Speed Commands (ESCs) are given to the motors. The protocol used for controlling these ECSs is Pulse Width Modulator (PWM), a periodic input pulse of width between 1000μs and 2000μs for zero to full power, respectively. Therefore, the angular velocities of the rotor cannot be known during the flight or simulation. That raises a problem, since the model needs these velocities as inputs, and not the PWM values. A way to overcome this, is to add an RPM sensor to the vehicle, but the current model does not include one. For this simulated model, there is a way to obtain these values, that are saved as GAZEBO states, but if the need to use RPM values in a real life vehicle without an RPM sensor emerge, then an experiment to derive a relation between PWM and angular velocity, should be conducted prior to identification.
A suitable sampling rate for all sensors and measurements of the specific model is found to be $\omega_s = 30Hz$.

Chapter 5

System Identification

5.1 Introduction to System Identification

System Identification is a method that allows building mathematical models of a dynamic system based on measured data, by adjusting parameters within a given model [21]. It is a field that became to grow after the rise of modern control theory around 1960, although it has its origins in standard statistical techniques. Great development has been made ever since and many techniques are established[22].
There are different identification methods and techniques that depend on the model type, the linearity of the problem or the kind of identification parameters. The most common techniques are applied to models of differential equations descriptions and all types of linear state-space models, but there are also methods that can handle more complex problems. In the latest years techniques for non-linear structures such as Artificial Neural Networks and Fuzzy models have been introduced and widely used as well. Detailed information about non-linear identification methods can be found in [23].
A wide classification of identification techniques is based on the format of the data used. In this context, there is *time-domain*, where measured time-series data are used for parameter estimations based upon least squares, maximum likelihood or recursive techniques, and *frequency-domain* identification, where spectral estimates, identifying frequency response characteristics are used to identify the structural parameters.
In this thesis only linear identification problems in the time-domain that can be applied to parametric model structures will be discussed. Parametric are the models that their class is known up to some set of parameters, which can include coefficients of the variables or other indications about the model structure [24].

5.2 Identification Procedure

The System Identification procedure consist of some basic steps, which according to Ljung [21] are:

- The collection of data set Z^N
  In order for the identification to provide efficient results, the collected data should be as informative as possible with respect to the models. An apparent interpretation of that is that the inputs should not be simple, so that they can evoke all the system properties. In a more precise mathematical approach, an open-loop experiment is informative if the input is persistently exciting. The number of parameters to be estimated in any case, indicates the order of persistent excitation that needs to be accomplished. As Ljung defines, a quasi-stationary signal $u(t)$ is persistently exciting of order n, if its spectrum $\Phi_u(\omega)$ is different from zero on at least n points in the interval $-\pi < \omega \leq \pi$.

- The selection of a model structure or a class of candidate model descriptions
  This is a process that depends on the application and the amount of information available about the system in question. In this multirotor vehicle case, the physical properties pertained to its operation do not allow a variety of model structures to be considered. Actually, only one model structure is contemplated, the gray box model that will be presented in Section 5.7.

- The selection of a fit criterion between the data and the model
  That is actually the identification method used to estimate the parameters. The Least Squares approach is the most common fitting method and that is the one that will be used in this thesis.

- Validation of the resulting model
  When a set of parameters is estimated and a model is selected, its quality needs to be assessed. Whether the model agree with the observed data and describes the true system sufficiently is a question that needs to be answered.

5.3 The archetypical problem

In this section the most common identification problem is presented, that of an Autoregressive with Extra Input (ARX) model. As stated by Ljung [21], its estimation procedure works as an archetypical problem that comprehends the basic identification concepts and the linear Least Square method, helping with the understanding of the techniques used throughout the chapter.

5.3.1 The model

A discrete time relationship between the input $u(t)$ and the output $y(t)$ at time t is the finite-difference model described in the following equation.

$$y(t) + a_1 y(t-1) + ... + a_n y(t-n) = b_1 u(t-1) + ... + b_m u(t-m) \qquad (5.3.1)$$

Eq. (5.3.1) can be rearranged, as a way of determining the next output value given all previous observations:

$$y(t) = -a_1 y(t-1) - ... - a_n y(t-n) + b_1 u(t-1) + ... + b_m u(t-m) \tag{5.3.2}$$

Here, are introduced the vectors:

$$\theta = [a_1, ...a_n, b_1, ..., b_m]^T \tag{5.3.3}$$

$$\phi(t) = [-y(t-1) \cdots -y(t-n)\ u(t-1) \cdots u(t-m)]^T, \tag{5.3.4}$$

and with these, Eq. (5.3.2) can be written as:

$$y(t) = \phi(t)^T \theta \tag{5.3.5}$$

Note that the output $y(t)$ depends on the parameters in θ, thus this calculated value will be called $\hat{y}(t|\theta)$ and Eq. (5.3.5) shall be written:

$$\hat{y}(t|\theta) = \phi(t)^T \theta \tag{5.3.6}$$

5.3.2 The Least squares method

Suppose now that for a given system we do not know the parameters in θ, but that we have a data set of inputs and outputs over a time interval $1 \leq t \leq N$:

$$Z^N = [u(1), y(1), ..., u(N), y(N)] \tag{5.3.7}$$

The goal is to estimate the parameter vector θ that fits the calculated values $\hat{y}(t|\theta)$ to the outputs $y(t)$. The least squares method is described as:

$$min V_N(\theta, Z^N), \tag{5.3.8}$$

where

$$V_N(\theta, Z^N) = \frac{1}{N}\sum_{t=1}^{N}(y(t) - \hat{y}(t|\theta)) = \frac{1}{N}\sum_{t=1}^{N}(y(t) - \phi(t)^T\theta) \tag{5.3.9}$$

The minimum can be found by setting the derivative to zero:

$$0 = \frac{d}{d\theta}V_N(\theta, Z^N) = \frac{2}{N}\sum_{t=1}^{N}\phi(t)(y(t) - \phi(t)^T)\theta \tag{5.3.10}$$

$$\tag{5.3.11}$$

which gives:

$$\sum_{t=1}^{N}\phi(t)y(t) = \sum_{t=1}^{N}\phi(t)\phi(t)^T\theta \tag{5.3.12}$$

So, the value of θ that minimizes (5.3.9) is:

$$\hat{\theta}_N = \Big[\sum_{t=1}^{N}\phi(t)\phi(t)^T\Big]^{-1}\sum_{t=1}^{N}\phi(t)y(t) \tag{5.3.13}$$

5.3.3 Covariance matrix

The estimated covariance matrix of $\hat{\theta}$ is described as:

$$P(\hat{\theta}) = \hat{\lambda}\left[\sum_{t=1}^{N}\phi(t)\phi(t)^T\right]^{-1}, \quad \hat{\lambda} = \frac{1}{N}\sum_{t=1}^{N}(y(t) - \phi(t)^T\hat{\theta})^2 \tag{5.3.14}$$

5.4 Matrix Formulation

All the above expressions can also be written in matrix forms as well as be augmented for multiple inputs and outputs.
Instead of $y(t)$ and $\phi(t)$, we may have $Y_{n\times 1}(t)$ and $\Phi_{n\times d}(t)$ respectively, where n is number of outputs and d is number of parameters.

$$\mathbf{Y}_N = \begin{bmatrix} Y_{n\times 1}(1) \\ Y_{n\times 1}(2) \\ \vdots \\ Y_{n\times 1}(N) \end{bmatrix} \tag{5.4.1}$$

$$\mathbf{\Phi}_N = \begin{bmatrix} \Phi_{n\times d}(1) \\ \Phi_{n\times d}(2) \\ \vdots \\ \Phi_{n\times d}(N) \end{bmatrix} \tag{5.4.2}$$

Then, the model can be written as:

$$\mathbf{Y}_N = \mathbf{\Phi}_N\hat{\theta} \tag{5.4.3}$$

and the criterion (5.3.9) becomes:

$$V_N(\theta) = \frac{1}{N}|\mathbf{Y}_N - \mathbf{\Phi}_N\hat{\theta}|^2 = \frac{1}{N}(\mathbf{Y}_N - \mathbf{\Phi}_N\hat{\theta})^T(\mathbf{Y}_N - \mathbf{\Phi}_N\hat{\theta}) \tag{5.4.4}$$

This concludes to formulation of the parameter estimation:

$$\hat{\theta} = [\mathbf{\Phi}_N^T\mathbf{\Phi}_N]^{-1}\mathbf{\Phi}_N^T\mathbf{Y}_N \tag{5.4.5}$$

In the latter equation the (Moore-Penrose) pseudoinverse of $\mathbf{\Phi}_N$ can be recognized and (5.4.5) can be rewritten as:

$$\hat{\theta} = \mathbf{\Phi}_N^{\dagger}\mathbf{Y}_N \tag{5.4.6}$$

Following these remarks the covariance matrix takes the form:

$$P(\hat{\theta}) = \frac{1}{N}\|\mathbf{Y}_N - \mathbf{\Phi}_N\hat{\theta}\|^2(\mathbf{\Phi}_N^T\mathbf{\Phi}_N)^{-1} \tag{5.4.7}$$

5.5 Merging Experiments

When more than one experiments are conducted or the data set is split into separate segments, a method to merge the parameter estimate needs to be defined. Let $\hat{\theta}_i$ be the estimated parameter vector of experiment i and P_i its estimated covariance matrix. In order to combine these estimates, according to statistics, the best way is to weigh them according to their inverse covariance matrices.

$$\hat{\theta} = P \sum_{i=1}^{n} (P_i)^{-1} \hat{\theta}_i \tag{5.5.1}$$

$$P = \left[\sum_{i=1}^{n} (P_i)^{-1} \right]^{-1} \tag{5.5.2}$$

5.6 Types of inputs

As already explained an informative experiment rely upon the form of the input signal u. There are some guidelines as for what kinds of signal designs result to informative experiments (see [21, chapter 13]), all of which have the distinguishable feature that must contain a variety of distinct frequencies. As Tiscler and Remple [25] introduce, a suitable signal format for aircraft or rotorcraft System ID, is that of a frequency-sweep (or chirp signal). It is actually a sinusoidal signal with amplitude A, as described in Eq.(5.6.1).

$$u_{sweep} = Asin(\omega(t)) \tag{5.6.1}$$

Here, an exponential frequency progression is implemented, according to Eq. (5.6.2).

$$\omega(t) = \omega_{min} + k(\omega_{max} - \omega_{min}), \tag{5.6.2}$$

$$k = c_1(\mathrm{e}^{c_2 t/T_{sweep}} - 1) \tag{5.6.3}$$

where ω_{min} is the minimum frequency, ω_{max} is the maximum frequency and T_{sweep} is the duration of the signal, which can be selected so:

$$T_{sweep} \geq (4 \text{ or } 5) T_{max} \tag{5.6.4}$$

where $T_{max} = 2\pi/\omega_{min}$, the period corresponding to the minimum frequency. The values $c_1 = 4.0$ and $c_2 = 0.0187$ are assigned to the constants, which are found to be suitable for a wide range of applications [25].
Linear or Quadratic sweep signals are also examined, where the frequencies progress according to equations (5.6.5) and (5.6.6) respectively.

$$\omega(t) = \omega_{min} + \frac{\omega_{max} - \omega_{min}}{T_{sweep}} t \tag{5.6.5}$$

$$\omega(t) = \omega_{min} + \frac{\omega_{max} - \omega_{min}}{T_{sweep}^2} t^2 \tag{5.6.6}$$

5.7 Model rearrangement

Although the state-space model does not comply with the ARX model structure as described above, there is a way to be rearranged in order to have a similar form with Eq. (5.4.3). This is only feasible if the initial model is linear with respect to the parameters.
Among the parameters issued in the model the following are assumed as known:

- The vectors r_i that correspond to distances and can easily be measured in any multirotor.
- the mass m of the multirotor

Since only the dynamic differential equations contain parameters in their formation, these are the equations that will be used for identification, and the new model structure is

$$\mathbf{Y} = \mathbf{\Phi}\boldsymbol{\theta}, \tag{5.7.1}$$

where the individual matrices are:

$$\mathbf{Y} = \begin{bmatrix} m(\dot{u} + qw - rv - gs\theta) \\ m(\dot{v} + ru - pw + gs\phi c\theta) \\ m(\dot{w} + pv - qu + gc\phi c\theta \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{5.7.2}$$

$$\mathbf{\Phi} = \begin{bmatrix} -u & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -v & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \sum {\omega_i}^2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -\sum r_{y_i} {\omega_i}^2 & 0 & \dot{p} & -qr & qr & q\sum sgn(\omega_i)\omega_i \\ 0 & 0 & \sum r_{x_i} {\omega_i}^2 & 0 & rp & \dot{q} & -rp & -p\sum sgn(\omega_i)\omega_i \\ 0 & 0 & 0 & -\sum sgn(\omega_i){\omega_i}^2 & -pq & -pq & \dot{r} & 0 \end{bmatrix} \tag{5.7.3}$$

$$\boldsymbol{\theta} = \begin{bmatrix} K_{d_u} & K_{d_v} & K_T & K_Q & I_{xx} & I_{yy} & I_{zz} & J_{rot} \end{bmatrix}^T \tag{5.7.4}$$

It is clear that matrix $\mathbf{Y}$ does not have the physical meaning of an output and some of the outputs are also appearing in matrix $\mathbf{\Phi}$. But that does not raise a problem in the procedure.

5.8 Octorotor Identification

The MATLAB® model can predict the response of the states X, with any given input U. Since there is no controller to guarantee the stability of the octorotor, all obtained results are only useful in a mathematical approach, keeping any physical meaning aside. Although they do not correspond to realistic movements, simulations of the octorotor can be used as a first test of the identification procedure and the selected model (5.7.1) rightness.
Various sets of simulations were conducted in which random values were given as inputs to the model. A fixed time step dt=1e-3 for the solver is selected. The different categories of inputs are explained below.

- Step functions
  In the first group of simulations constant values were given to the input vector U. Of course these values were not identical for every rotor, ensuring that forces and torques are exerted with regard to all axes. Five simulations with duration 5 seconds each, were run, with rotor rates randomly chosen between 400 and 800 rad/s.

- Cosinusoidal functions
  In this category of simulations the rates of the rotors take the form $\omega_i = 400 + W_i cos(f_i t)$, where both the amplitude W_i and the angular frequency f_i are randomly assigned in range 0-400 rad/s for the former and 0-10 rad/s for the latter. Five simulations with 5 seconds duration were run in this case as well.

For every experiment, the set of data $Z = [X(1), U(1), ..., X(N), U(N)]$, is used to construct the matrices $\mathbf{Y}$ and $\mathbf{\Phi}$ of the model (5.7.1). The derivatives of u, v, w and p, q, r are also needed for the procedure. In this case, MATLAB® `gradient` function is used, to obtain the desired terms numerically. When all terms are known Eq.(5.4.5) is used to receive the estimated parameters.

The experiments are merged as explained in Section 5.5 and the results are:

Table 5.8.1: Identification results of the octorotor experiments.

	Step inputs	Cosinusoidal inputs	Both	**Exact values**
K_{d_u}	0.29975	0.29993	0.29992	**0.3**
K_{d_v}	0.29838	0.3	0.29997	**0.3**
K_T	2.2001e-05	2.2001e-05	2.2001e-05	**2.2e-05**
K_Q	4.5002e-07	4.5002e-07	4.5002e-07	**4.5e-07**
I_{xx}	0.10901	0.10901	0.10901	**0.109**
I_{yy}	0.10801	0.10801	0.10801	**0.108**
I_{zz}	0.20801	0.20801	0.20801	**0.208**
J_{rot}	2.0003e-05	2.0001e-05	2.0001e-05	**2e-05**

Comparing these results with the exact values of the parameters (see Section 4.1) it is clear that the identification procedure gives almost perfect results and no validation method is needed here. Additionally, the design of more complex inputs, as described in Section 5.6, is deemed unnecessary, since little or no room for improvement remains.
Of course the same procedure cannot be applied in a real life octorotor and thus the work is not concluded here. It is however a perfect proof that the model works, and a fundament for the following work.

5.9 Quadrotor identification

In this section the identification of the model parameters using the quadrotor model described in (4.2) is conducted. The simulated model runs in a closed loop but since the parameters of the control scheme are considered known and/or there is no interest of their estimation a *Direct Identification* approach can be employed [21, p. 457]. Hence, the identification will be done in a same way as for an open loop operation, ignoring any possible feedback.

5.9.1 Input signals

The version of ardupilot used does not allow any user to give directly PWM commands to the vehicle. Hence, the form of the input cannot be predefined. The different ways to give commands to the multirotor are stated below:

1. A joystick that can give manual commands to the (simulated) vehicle. With this method, we may have the highest freedom of movements, but it is really difficult to guarantee repeatability of the experiments.

2. Position or linear velocity targets through SITL command line. It is important to note here, that the SITL works with a NED frame, thus the axes used in the commands do not coincide with that of the earth frame specified in Section 2.1 .

3. Using a ROS publisher to send raw setpoint messages to Flight Controller Unit (FCU). Here, linear velocity and yaw rate setpoints are sent with respect to a local earth-fixed frame.

For the simulations conducted here, the third option is used, since it is the only way to give signals as functions of time. A rate of 15Hz is used for the sampling of these messages, in order for Shannon's sampling theorem to be verified [26]. Since it is impossible to define the format of the input signal (PWM) in order to preassume the persistent excitation of the model, the format of the setpoint signals needs to be wisely chosen.
A first consideration is that the signals must be so, that they exert forces and torques in multiple axes and thus the quadrotor moves in multiple directions. However, the capabilities are limited, since no signals can be sent in the roll and pitch angles or angle rates. The logic that the inputs should consist of multiple frequencies is also taken into account when the design of the setpoint signals is made, hence the directions of Section 5.6 are followed for the setpoint signals as well.

5.9.2 Data Collection and Preprocessing

For any simulated flight the data set needs to be saved in order to use it for the model fitting. The acquisition of the simulation data is done through a ROS subscriber which saves rostopics data in a file that will later be imported in MATLAB®. The different categories of collected data are:

- A record of the simulation time
- Position and Orientation in quaternion formulation with their time stamps
  Since the model uses Euler angles to represent the orientation, the quaternions need to be transformed to ϕ, θ, ψ angles (see Appendix (A.2)).
- Linear and Angular velocities expressed in the body-fixed frame with their time stamps
- Filtered accelerometer measurements from the IMU with time stamps
  According to Eq. (4.2.1) these measurements are expressed, with respect to terms appearing in the model, and for the different axes, as:

$$\begin{aligned} \dot{u}_{IMU} &= \dot{u} + qw - rv - gsin\theta \\ \dot{v}_{IMU} &= \dot{v} + ru - pw + gsin\phi cos\theta \\ \dot{w}_{IMU} &= \dot{w} + pv - qu + gcos\phi cos\theta \end{aligned}$$

- The rotor velocities in rad/s as GAZEBO states
- The PWM commands as given to the motors

The rate of this procedure is set at $\omega_s = 30$Hz but some of the topics have their own time stamps. Hence, a way to adjust all the data to correspond to the same time values need to be found. Firstly, all duplicate records due to sensors lags or delays are deleted. The time stamp of the position data is used, and all other measurements are interpolated in order to gain data at this time record.
The values of the angular accelerations are also needed for the identification procedure but they are not measured by any sensor. The only way to receive these values is to numerically integrate the time series of the angular velocities. MATLAB® function `gradient` is used here, as well. As any numeric computation, this may induce errors that can affect the identification results. The use of a smoothing filter is investigated in order to reduce these errors. MATLAB® Signal Analyzer App was used to test the effects of different smoothing filters on the data, and a Savitzky-Golay [27] filter with a smoothing factor of 0.5 was finally selected. It is under doubt whether the smoothing will improve the procedure, since it may affect the correlation between the data, thus the application of the filter will be examined by comparing the results of both cases.
As part of the preprocessing, the use of a lowpass filter for the data collected from the EKF is also considered. A suitable lowpass filter is designed, using MATLAB® `lowpass` function, which automatically chooses the best filter option between FIR

and IIR filters [28]. The passband frequency is selected as $\omega_f = 6$Hz in order to satisfy the relation:

$$\omega_s \geq 5\omega_f \tag{5.9.1}$$

Since lowpass filters induce a phase lag in the data, the MATLAB® function `liftlift` was used, which performs zero-phase digital filtering in both forward and reverse directions.

5.9.3 Simulated Experiments

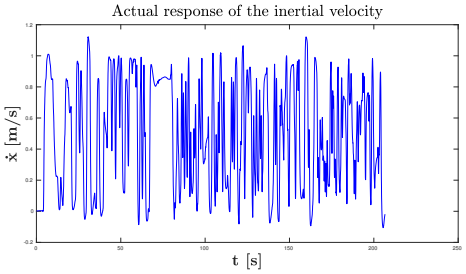
The format of the signals and the desired frequencies demanded a trial and error approach in order to find the best input types (amplitudes, frequencies or mean value of signals). A lot of trial experiments were conducted throughout the procedure, in order to find suitable inputs for the simulation and get "good" predictions of the parameters. The evaluation of the signals, prior to validation of the results, was done by using two main criteria; firstly, by checking how sufficiently the autopilot could follow the setpoint signals and subsequently by examining the estimated parameters. An obvious alert is whether the estimated parameters have negative, non-physical values, or even if they dissent by far from the known values.
Some kinds of signal types stood out, as the most prominent. In all of these cases the setpoint signals take the form $u = Asin(\omega t) + B$, but they differ in the way their frequencies or their amplitudes change during the simulated flight. When the yaw rate is not oscillating around zero, it is verified that produces better estimation values. All these experiments can be divided into three categories.

As for the first group, frequencies that increase stepwise over specified time periods and amplitudes that are fixed during the experiment were applied. The frequencies of the velocity commands and the yaw rate frequency were different in most of these simulations. Here the results of an experiment conducted with setpoint inputs as described in Table (5.9.1) are presented, Experiment N^o 1.

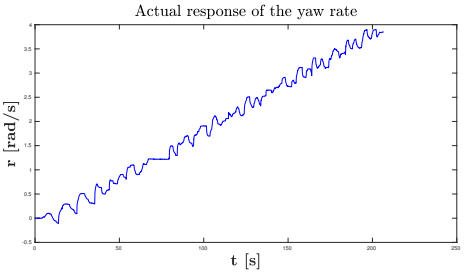
Table 5.9.1: Given input signals for Experiment N^o 1.

Setpoint Velocities	**Values**
$\dot{x}_{des}$	$0.5sin(\omega t) + 0.5$
$\dot{y}_{des}$	$0.5sin(\omega t) + 0.5$
$\dot{z}_{des}$	0.0
$\dot{\psi}_{des}$	$0.1sin(\omega_{yr} t) + b$

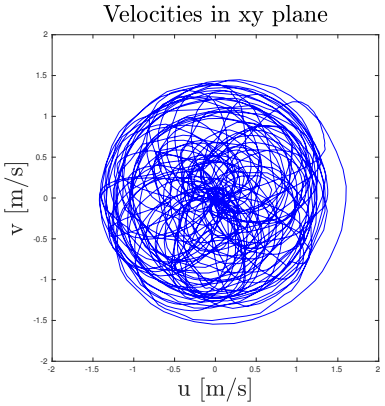
The values of ω, ω_{yr} and b are initially 0.5, 0.5 and 0.0, and are increasing every 10s, by 0.5, 0.02 and 0.2 rad/s respectively.



(a) The actual linear earth-frame velocity of x axis.



(b) The actual yaw rate.



(c) Trajectory of body velocities in xy plane.

Figure 5.9.1: Simulated response of the quadrotor with the input signals of Experiment N^o 1.

In Fig. (5.9.1) some plots of the collected data are presented. Although it is profound that the velocity signal given to the autopilot could not be exactly followed, a generally good estimation of the parameters is provided (Table (5.9.2)). From now on, as $\hat{\theta}_{raw}$ will be referred the vector of parameters estimation, where the numerically computed accelerations are not filtered, while as $\hat{\theta}_{smooth}$ are the estimations using smoothed data. It is profound, that this filter only affects the estimation of the parameters that are involved in the equations describing the angular velocities.

Table 5.9.2: Estimated parameters of Experiment N^o 1.

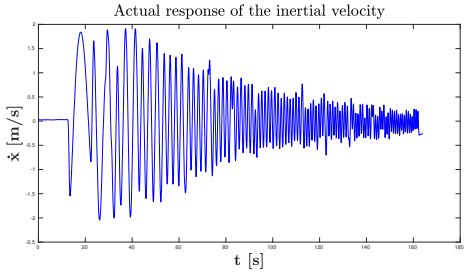
	$\hat{\theta}_{smooth}$	$\hat{\theta}_{raw}$
K_{d_u}	0.0076089	0.0076089
K_{d_v}	0.0091162	0.0091162
K_T	2.169e-05	2.169e-05
K_Q	3.4223e-09	4.4447e-09
I_{xx}	0.014837	0.013958
I_{yy}	0.018459	0.017608
I_{zz}	0.0109	0.0087216
J_{rot}	0.00033652	0.00038228

Noticing that as the frequencies are increasing the quadrotor cannot follow the desired velocity trajectory, signals with decreasing amplitude were designed, composing the second group of experiments. One of the trial inputs is presented in (5.9.3), and will be referred as Experiment N^o 2.

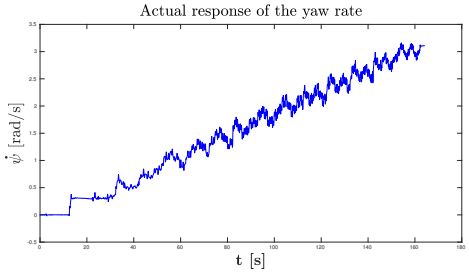
Table 5.9.3: Given input signals for Experiment N^o 2.

Setpoint Velocities	**Values**
$\dot{x}_{des}$	$asin(\omega t)$
$\dot{y}_{des}$	$0.5asin(\omega t) + 0.5$
$\dot{z}_{des}$	0.0
$\dot{\psi}_{des}$	$0.1sin(\omega_{yr} t) + b$

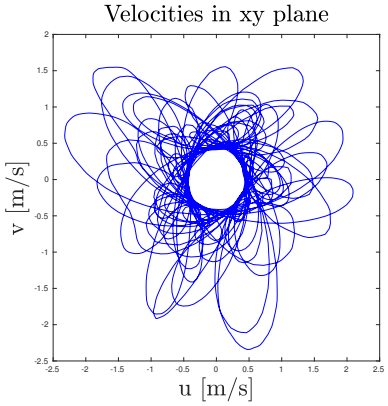
The values of ω, ω_{yr} and b are initially 0.5, 0.1 and 0.0, and are increasing every 10s, by 0.8, 0.02 and 0.1 rad/s respectively. The amplitude a is designed to begin from $a_0 = 0$ and decrease linearly to $a_{final} = 0.05$. Plots of the followed trajectories are presented in (5.9.2).



(a) The actual linear earth-frame velocity of x axis.



(b) The actual yaw rate.



(c) Trajectory of body velocities in xy plane.

Figure 5.9.2: Simulated response of the quadrotor with the input signals of Experiment N^o 2.

A comparison between the PWM signals and the angular velocities of the rotors is also presented here, in order to clarify their relationship. Even from Fig. (5.9.3) it is clear that the signals have similar forms, and both demonstrate oscillating behavior as desired.

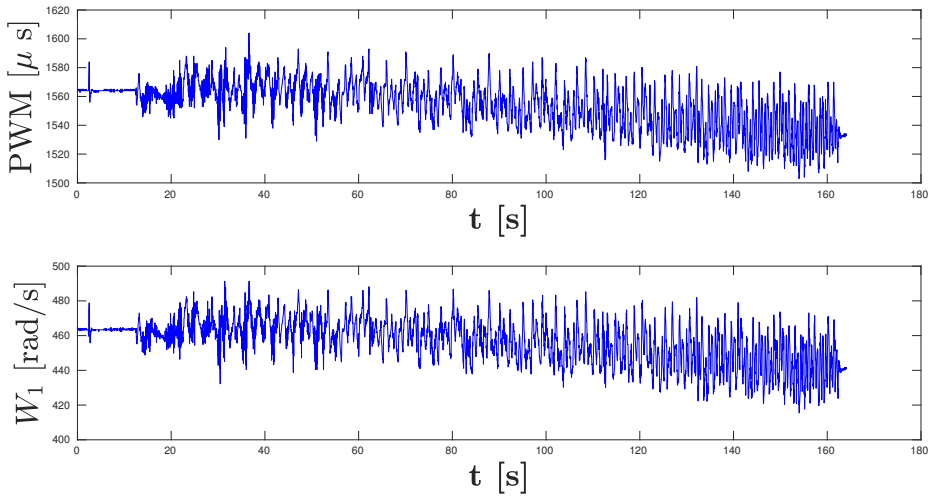


Figure 5.9.3: Plots of input data of rotor n.1 as PWM commands (up) and as angular rate in rad/s (down).

An analytical relation was found between the two, by using MATLAB® Curve Fitting Tool, for all four rotors. The mean values of the coefficients give us a final function description as seen below.

Table 5.9.4: Estimated relation between PWM and angular velocity values with data of Experiment N^o 1.

	Relation	R^2
rotor n.1	$\omega_1 = 0.7298 PWM_1 - 677.4$	0.8471
rotor n.2	$\omega_2 = 0.7328 PWM_2 - 682.3$	0.8685
rotor n.3	$\omega_3 = 0.7318 PWM_3 - 680.9$	0.7318
rotor n.4	$\omega_4 = 0.7393 PWM_4 - 692.6$	0.8711
	$\overline{\omega} = 0.7334 \overline{PWM} - 683.3$	

The estimated parameters from this experiment are presented in Table (5.9.5). Although, the inertia J_{rot} of the rotors takes a negative value, we have a better estimation of I_{zz}, since it is a better approach of the value presented in Section 4.2.1.

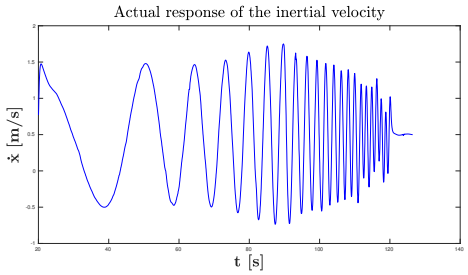
Table 5.9.5: Estimated parameters of Experiment N^o 2.

	$\hat{\theta}_{smooth}$	$\hat{\theta}_{raw}$
K_{d_u}	0.0056729	0.0056729
K_{d_v}	0.0057718	0.0057718
K_T	2.1684e-05	2.1684e-05
K_Q	3.0664e-07	2.7701e-07
I_{xx}	0.013306	0.012733
I_{yy}	0.017525	0.016558
I_{zz}	0.024884	0.022452
J_{rot}	-0.00017544	-0.00012423

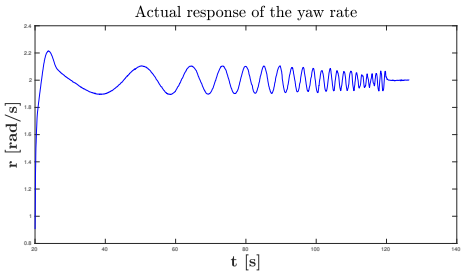
The third group consists of the most prominent category of inputs, the sweep signals. Using as guideline some reference values introduced in [25], and consulting the work of Cho et al. [29] and Gong et al. [30], who performed sweep signals with frequency range 0.6-60rad/s (∼0.1-10Hz) over 60s, a variety of tests were conducted. More suitable for the quadrotor simulations here, were found the frequency sweeps between 0.6-60rad/s over 100s, and an example of these simulations will be presented here as Experiment N^o 3. Eq. (5.6.2), is used to compute the desired frequency at every time step.

Table 5.9.6: Given input signals for Experiment N^o 3.

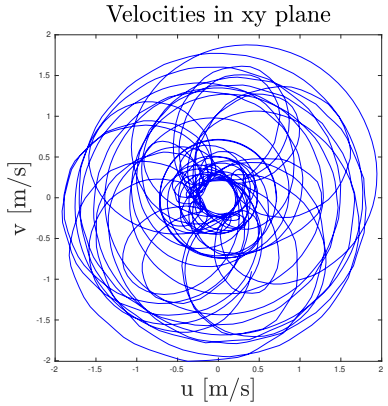
Setpoint Velocities	**Values**
$\dot{x}_{des}$	$sin(\omega t) + 0.5$
$\dot{y}_{des}$	$0.5sin(\omega t) + 0.5$
$\dot{z}_{des}$	0.0
$\dot{\psi}_{des}$	$0.1sin(\omega t) + 2.0$



(a) The actual linear earth-frame velocity of x axis.



(b) The actual yaw rate.



(c) Trajectory of body velocities in xy plane.

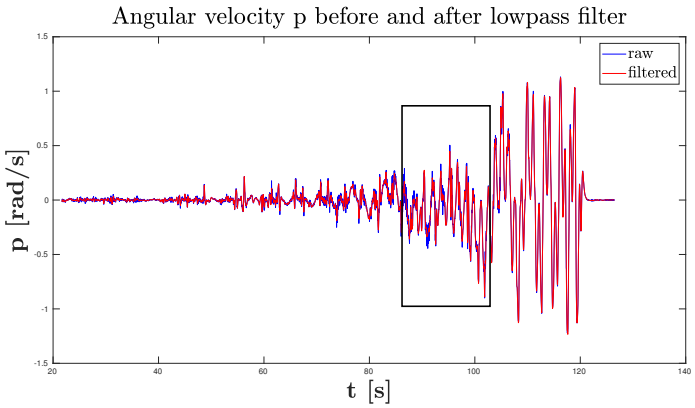
Figure 5.9.4: Simulated response of the quadrotor with the input signals of Experiment N^o 3.

As for the estimated parameters of this experiment, presented in Table (5.9.7), it could be claimed, that a middle solution for the values of I_{zz} and J_{rot} is found. Although I_{zz} takes a lower value than expected, still is bigger than that of Experiment N^o 1, providing a good estimation of J_{rot} simultaneously.

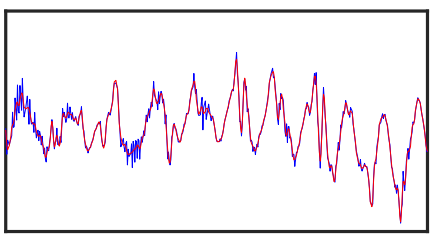
Table 5.9.7: Estimated parameters of Experiment N^o 3

	$\hat{\theta}_{smooth}$	$\hat{\theta}_{raw}$
K_{d_u}	0.0084335	0.0084335
K_{d_v}	0.0089067	0.0089067
K_T	2.169e-05	2.169e-05
K_Q	1.3663e-09	1.2867e-09
I_{xx}	0.014105	0.011815
I_{yy}	0.017953	0.016541
I_{zz}	0.015345	0.014451
J_{rot}	0.00015537	0.00016109

At this point, the effects of the two already discussed filters will be illustrated, using collected data from Experiment N^o 3. In Fig. (5.9.5) the response of the angular velocity p data, is chosen as an example to demonstrate the results after the filter implementation.



(a) The angular velocity p before and after the implementation of the lowpass filter.

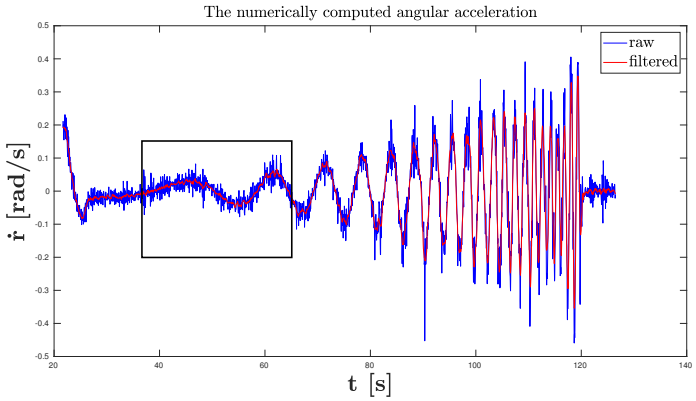


(b) Detail of the box area of plot in (a).

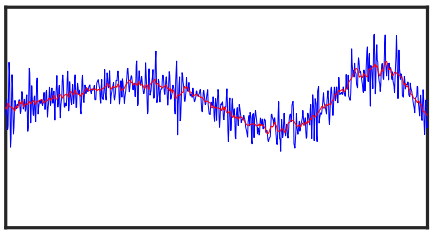
Figure 5.9.5: The response of angular velocity p throughout the Experiment N^o 3 .

The effect of the smoothing filter over the numerically integrated data is illustrated

in Fig. (5.9.6), using the z-axis angular acceleration $\dot{r}$.



(a) The angular acceleration $\dot{r}$ before and after the implementation of the smoothing filter.



(b) Detail of the box area of plot in (a).

Figure 5.9.6: Illustration of the numerically computed $\dot{r}$.

5.9.4 Results and Validation

From the set of conducted experiments, the ones that individually presented the best results, are now combined according to Eq. (5.5). Obviously, their covariance matrices were computed prior to merging. Different combinations of simulations were examined, until the most suitable merger was achieved. Experiments that belong to the first and second group, as described above, could deliver better results as the inertia I_{zz} is concerned.

Table 5.9.8: Estimated parameters from a total of experiments.

	$\hat{\theta}_{smooth}$	$\hat{\theta}_{raw}$
K_{d_u}	0.0067367	0.0067372
K_{d_v}	0.0059741	0.0059741
K_T	2.1685e-05	2.1685e-05
K_Q	1.0666e-08	9.9358e-09
I_{xx}	0.013728	0.01305
I_{yy}	0.017532	0.016772
I_{zz}	0.023258	0.021431
J_{rot}	-0.00010752	-7.4915e-05

Of course this solution, presented in Table (5.9.8), cannot be fully acceptable, due to the unfeasibility of the inertia J_{rot}. However, since this parameter only affects the terms associated with gyroscopic effects, which are of minor significance, this set of parameters will not be completely discarded.
In order to resolve the above issue, more experiments are now included in the procedure of finding a final parameter vector, as seen in Table (5.9.9). Indeed, the value of J_{rot} is now positive, and it well approaches the real one. The "price" is although that the value of I_{zz} is now decreased.

Table 5.9.9: Final parameter estimation.

	$\hat{\theta}_{smooth}$	$\hat{\theta}_{raw}$
K_{d_u}	0.0088314	0.0088328
K_{d_v}	0.0090478	0.0090492
K_T	2.1689e-05	2.1689e-05
K_Q	1.5363e-09	1.421e-09
I_{xx}	0.013845	0.01296
I_{yy}	0.017624	0.016738
I_{zz}	0.014848	0.013394
J_{rot}	0.00017439	0.00019407

As for the smoothing filter, one can notice, that when applied, the values of the affected parameters are slightly increased, but not an outstanding difference is made. We can, thus, conclude that either raw or smoothed integrated data, can be used for the system identification procedure.
The validation of the parameters' estimation here, is a tricking problem. States or their derivatives need to be computed using a model that integrates the estimated

parameters and then be compared with the responses of the initial simulated model. Of course the first thought would be to include the estimated parameters in a MATLAB® model and solve the differential equations, providing input U data collected from a simulated experiment, but the numerical solver propagates any errors that occur and hence the responses show grate deviations. As a middle solution, values of the states and the inputs, along with the estimated parameters, are substituted in the differential equations (2.4.1) and (2.4.2). The derivatives of the states u, v, w, p, q and r, are then compared with the respective values obtained from the simulated sensors. The problem with this procedure is, once again, that the differentiated values have to be used for the angular accelerations, since the actual values cannot be measured. For all the following results and plots, the smoothing filter is applied in the numerically integrated data and hence the parameter vector $\hat{\theta}_{smooth}$ of Table (5.9.9) is assumed as the final estimation.
A first attempt of validation is executed using data collected from an experiment of similar specifications as the ones used for the estimation. The resulting time series of the accelerations, along with data obtained from the simulations are presented in Fig. (5.9.7). The coefficient of determination R^2 [31], is also calculated for every axis as a means to evaluate the selected parameters.

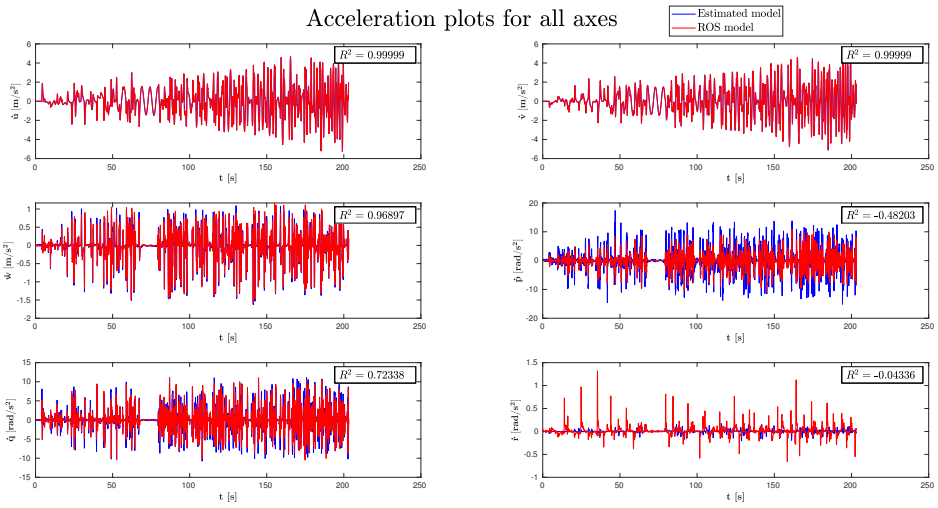


Figure 5.9.7: Plots comparing the actual accelerations and the ones based on the estimated parameters. The coefficients of determination R^2 are also shown in the plots for each axis.

It is clear, that the linear terms, are almost perfectly predicted when the parameters K_{d_u}, K_{d_v} and K_T are implemented, and that can be verified by their R^2 values. The angular terms however seem to have more difficulty to be reproduced from the model. Actually, according to R^2 values, only the response of $\dot{q}$ seems to be well predicted. However, with a closer look in the $\dot{p}$ response (see Fig. (5.9.8)), one can see that despite the negative R^2, the predicted model can follow the response of the original one to a certain extent. The biggest problem arises with the z-axis acceleration. A plausible explanation is the I_{zz} value, which as already stated, diverges from the known one, or even the K_Q, the value of which cannot be verified

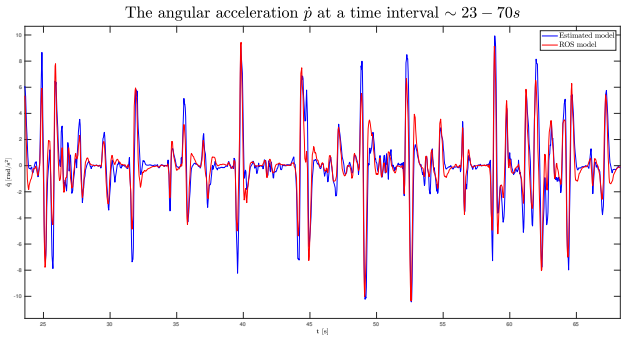


Figure 5.9.8: Detail of the $\dot{p}$ response.

in any other way.
A second experiment, concerning simpler commands, is used for another validation attempt. Desired velocities were given to the vehicle using the SITL command line (2^{nd} method of Subsection (5.9.1)) and the results are presented in Fig. (5.9.9).

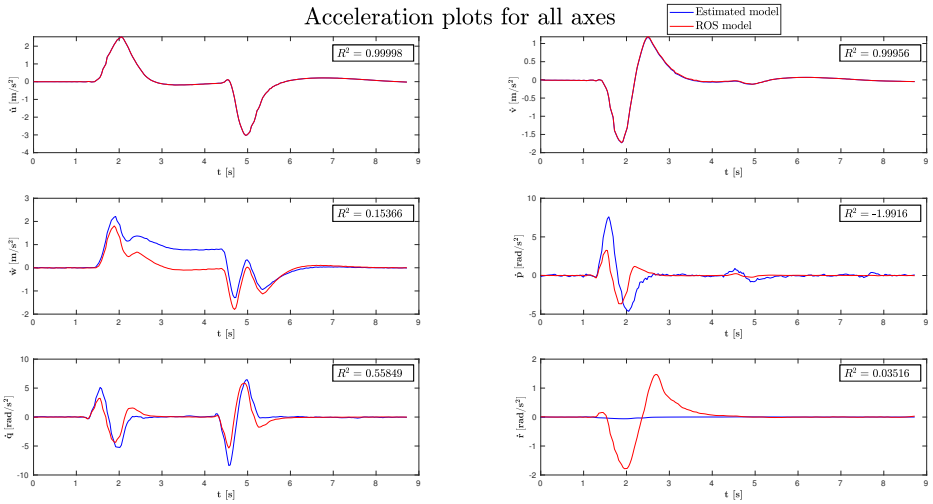
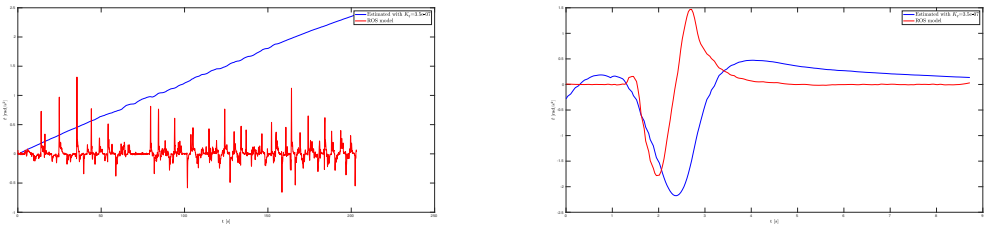


Figure 5.9.9: Plots comparing the actual accelerations and the ones based on the estimated parameters, for the 2^{nd} validation attempt.

In this case, the linear acceleration $\dot{w}$ cannot be predicted as effectively as before, probably because the model was not trained for this kind of movements. It is clearer, though, that the accelerations $\dot{p}$ and $\dot{q}$ are well predicted. What is concerning, is the total failure of the $\dot{r}$ prediction.

5.9.5 Overall discussion of the results

Taking into account the overall System Identification procedure and even more specifically the two validation attempts presented above, some general comments concerning the estimated parameters can now be noted.
The aerodynamic coefficients K_{d_u} and K_{d_v}, although their estimation does not converge in specific values, are always assigned small values of the same order of magnitude. That concludes to the notion that these terms have little contribution in the model, and the responses of $\dot{u}$ and $\dot{v}$ are barely affected by them.
Concerning the rotor coefficients K_T and K_Q, the results are controversial. The Thrust and Drag calculation terms are based on some assumptions, which may not be fully met throughout the simulations. On top of that, as already mentioned, the forces and moments of the propellers are calculated using polar diagrams, without the implementation of constant terms. This indicates that in different operating areas the values may differ. The constant K_T showed a common behaviour in all of the conducted experiments, in which, though, no large accelerations were observed on the z-axis. However, in commands such as that of the second validation attempt, the estimated value does not correspond with the same accuracy in the thrust force calculation. Regarding K_Q, which only appears in the $\dot{r}$ equation, we see that the value found is not able to predict the motion along this axis. In an attempt to verify, at least, the order of magnitude of the predicted K_Q, slight alterations on its value were made, before the comparison of the estimated and the simulated model. The plots produced by applying K_Q = 3.5e-07, prior to the validation procedure, for the two cases examined above are presented in Fig. (5.9.10). In (5.9.10a), it is clearly depicted, that if the value of K_Q was larger than initially predicted, then there would be a large deviation from the simulated model. On the contrary, as observed in (5.9.10b), only with a bigger K_Q value the quadrotor can actually drift from the zero acceleration value. That may work as evidence, that a constant value of K_Q is not sufficient to describe all kinds of motions and maneuvers. Depending on the application, that the parameters are going to be embedded in any future work, more specific experiments could be conducted in order to find a suitable estimation.



(a) Effect of a different K_Q value in the first validation experiment

(b) Effect of a different K_Q value in the second validation experiment

Figure 5.9.10: Effect of a bigger K_Q value ($K_Q = 3.5e{-}07$) in the predicted response of the experiments used for validation.

Regarding the identification of the inertia terms (I_{xx}, I_{yy}, I_{zz}, J_{rot}), although R^2 are not satisfactory for all the rotational equations, in this case their known values can be used to verify the results. Indeed, the values approximate quite well the values of the known parameters for I_{xx}, I_{yy} and J_{rot} and the estimation can be considered as successful. In particular for I_{zz}, that from the overall experiments there is difficulty to find an approximate value, in a parameter identification process of a real-life UAV a different approach might be needed. To begin with, it has been confirmed from the experiments that when large values of angular velocity in z axis are given as inputs, the values of I_{zz} approximate better the known value, so perhaps a separate procedure with only such experiments should be followed, aiming only at finding I_{zz}. In a worst case scenario, an analytical or an experimental procedure could be followed, as Mendes et al. describe [32].

Chapter 6

Conclusions and Future work

The main purposes of this thesis are met, although not in an extent as satisfactory as desired. The results obtained are promising and, while not perfect, they can be a helpful first step for any future work regarding that matter.

An extensive general description of a multirotor model is constructed, that can be used for different types of multirotor configurations, concerning the number of rotors, their tilt angles and their positioning. A specific model is derived for two different platforms, which are explicitly presented along with their characteristics and their simulation environments.
As a following module of this thesis, a system identification procedure for the estimation of a multirotor's model parameters has been described. The Least Squares method has been explained and a way to rearrange the model in order to be suitable for that approach is found. Afterwards, the process has been implemented for the two different model configurations. The identification of the octorotor's parameters provided the desired results but since no control scheme was included, that procedure cannot be applied in any real-life multirotor because the stability of the vehicle cannot be guaranteed. Hence, more weight is given to the case of the quadrotor, which is autopilot equipped. The main focus was directed to the design of the signals that the simulated model could receive, in order to be persistently excited. The sinusoidal signals that were sent as linear velocity and yaw rate commands are found suitable to predict most of the parameters, or at least to provide useful information about their behaviour. With the estimated parameters, the actions of the simulated quadrotor, can be to some extent predicted by the derived model, except from the rotary motion around z axis.

Aiming for better results, this research could be futurely advanced or readjusted. To begin with, the integration of a different autopilot version such as PX4, might be of interest, since it allows the users to send PWM signals directly to the controller. In that way the inputs of the model could be designed based on the guidelines discussed in this thesis, ensuring that they will consist of the desired frequency range. Another future approach could be to use optimization algorithms or other system identification techniques that may be more suitable for this kind of models.
However, even with these parameters, a model-based controller could be designed

and implemented in order to test the behaviour of the system, as another means of validation. It is possible that the predicted values are sufficient and that the controller can follow its tasks, otherwise it may give a good direction in ways to readjust the parameters.

Appendix A

Rotation representation

A.1 Elementary Rotations

Let us define a reference frame $O - xyz$. The rotations of this frame about one of the coordinate axes are called elementary rotations.
Consider that the original frame is rotated by angle α around z axis and a frame $O - x'y'z'$ is obtained. The Rotation matrix of frame $O - x'y'z'$ with respect to $O - xyz$ is:

$$R_z(\alpha) = \begin{bmatrix} cos\alpha & -sin\alpha & 0 \\ sin\alpha & cos\alpha & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{A.1.1}$$

The Rotation matrices of an angle β around y axis and angle γ around x axis are respectively:

$$R_y(\beta) = \begin{bmatrix} cos\beta & 0 & sin\beta \\ 0 & 1 & 0 \\ -sin\beta & 0 & cos\beta \end{bmatrix} \tag{A.1.2}$$

$$R_x(\gamma) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & cos\gamma & -sin\gamma \\ 0 & sin\gamma & cos\gamma \end{bmatrix} \tag{A.1.3}$$

A.2 Quaternions

A different representation of the orientation can be achieved by using unit quaternions. It is a system with 3 complex dimensions that needs four parameters to be described. It can be represented in the two following forms.

$$\mathbf{q} = q_0 + q_1\mathbf{i} + q_2\mathbf{j} + q_3\mathbf{k} \tag{A.2.1}$$

$$\mathbf{q} = \begin{bmatrix} q_0 & q_1 & q_2 & q_3 \end{bmatrix}^T \tag{A.2.2}$$

A rotation of angle α around axis $\mathbf{u}$, can be described by a unit quaternion as Eq. (A.2.3) states.

$$\mathbf{q} = \begin{bmatrix} cos\frac{\alpha}{2} & sin\frac{\alpha}{2}\mathbf{u} \end{bmatrix} \tag{A.2.3}$$

The length of a unit quaternion is always:

$$\|\mathbf{q}\| = \sqrt{q_0^2 + q_1^2 + q_2^2 + q_3^2} = 1 \tag{A.2.4}$$

A rotation matrix corresponding to a given quaternion takes the form:

$$R(\mathbf{q}) = \begin{bmatrix} 2(q_0^2 + q_1^2) - 1 & 2(q_1 q_2 - q_0 q_3) & 2(q_1 q_3 + q_0 q_2) \\ 2(q_1 q_2 + q_0 q_3) & 2(q_0^2 + q_2^2) - 1 & 2(q_2 q_3 - q_0 q_1) \\ 2(q_1 q_3 - q_0 q_2) & 2(q_2 q_3 + q_0 q_1) & 2(q_0^2 + q_3^2) - 1 \end{bmatrix} \tag{A.2.5}$$

Relations between quaternions and Euler angles

According to Eq. (A.2), a quaternion can also be calculated by using the Euler angles as shown below:

$$\mathbf{q}_{B\to E} = \begin{bmatrix} cos(\phi/2)cos(\theta/2)cos(\psi/2) + sin(\phi/2)sin(\theta/2)sin(\psi/2) \\ sin(\phi/2)cos(\theta/2)cos(\psi/2) - cos(\phi/2)sin(\theta/2)sin(\psi/2) \\ cos(\phi/2)sin(\theta/2)cos(\psi/2) + sin(\phi/2)cos(\theta/2)sin(\psi/2) \\ cos(\phi/2)cos(\theta/2)sin(\psi/2) - sin(\phi/2)sin(\theta/2)cos(\psi/2) \end{bmatrix} \tag{A.2.6}$$

An expression of the Euler angles with respect to quaternion parameters can also be derived:

$$\begin{bmatrix} \phi \\ \theta \\ \psi \end{bmatrix} = \begin{bmatrix} tan^{-1}\dfrac{2(q_0 q_1 + q_2 q_3}{1 - 2(q_1^2 + q_2^2} \\ sin^{-1}[2(q_0 q_2 - q_1 q_3)] \\ tan^{-1}\dfrac{2(q_0 q_3 + q_1 q_2}{1 - 2(q_2^2 + q_3^2} \end{bmatrix} \tag{A.2.7}$$

Εκτενής Περίληψη της διπλωματικής εργασίας στα Ελληνικά

Εθνικό Μετσόβειο Πολυτεχνείο
Σχολή Μηχανολόγων Μηχανικών
Τομέας Μηχανολογικού Σχεδιασμού & Αυτομάτου Ελέγχου
Εργαστήριο Αυτομάτου Ελέγχου

Μοντελοποιηση και Αναγνωριση συστηματος μη επανδρωμενου εναεριου οχηματος πολλαπλων ελικων με χρηση δεδομενων προσομοιωσης

Διπλωματική Εργασία

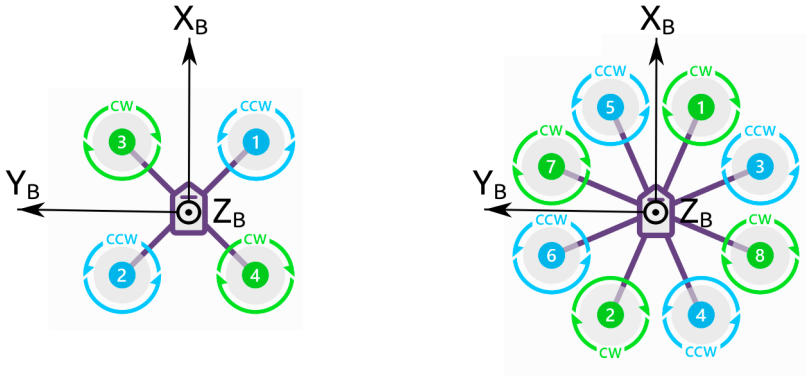
Ηλιοπούλου Αντουανέττα

Επιβλέπων:
Κ.Κυριακόπουλος, Καθηγητής ΕΜΠ

Αθήνα, Μάρτιος 2022

Μοντελοποίηση μη επανδρωμένου εναέριου οχήματος πολλαπλών ελίκων

Πρωτού ξεκινήσει η διαδικασία εύρεσης του μαθηματικού μοντέλου που περιγράφει την κίνηση ενός εναέριου οχήματος πολλαπλών ελίκων, είναι απαραίτητος ο ορισμός συστημάτων συντεταγμένων. Ορίζονται, στην παρούσα περίπτωση, δύο βασικά συστήματα· ένα αδρανειακό γεώδετο σύστημα (E) και ένα σωματόδετο (B), με αρχή στο κέντρο βάρους του οχήματος και τους άξονες όπως φαίνονται στο Σχήμα (1) για τους δύο διαφορετικούς τύπους πολυκοπτέρων που μελετώνται στην παρούσα εργασία.



(α′) Τετρακόπτερο (β′) Οκτακόπτερο

Σχήμα 1: Παρουσίαση του σωματόδετου συστήματος συντεταγμένων για δύο τύπους οχήματος.

Η θέση και ο προσανατολισμός του οχήματος στον χώρο μπορούν να περιγραφούν με βάση το γεώδετο σύστημα:

$$\mathbf{\Gamma}_E = [x_E \; y_E \; z_E]^T \tag{1}$$

$$\mathbf{\Theta}_E = [\phi_E \; \theta_E \; \psi_E]^T \tag{2}$$

Οι γραμμικές και γωνιακές ταχύτητες, από την άλλη, προτιμάται να εκφράζονται στο σωματόδετο:

$$\boldsymbol{\nu}_B = [u_B \; v_B \; w_B]^T \tag{3}$$

$$\boldsymbol{\omega}_B = [p_B \; q_B \; r_B]^T \tag{4}$$

Οι διαφορικές σχέσεις που συνδέουν τα παραπάνω μεγέθη προκύπτουν:

$$\dot{\mathbf{\Gamma}}_E = R_{B\to E}\, \boldsymbol{\nu}_B \tag{5}$$

$$\dot{\mathbf{\Theta}}_E = T_{B\to E}\, \boldsymbol{\omega}_B, \tag{6}$$

όπου

$$R_{B\to E} = \begin{bmatrix} c\psi c\theta & -s\psi c\phi + c\psi s\theta s\phi & s\psi s\phi + c\psi s\theta c\phi \\ s\psi c\theta & c\psi c\phi + s\psi s\theta s\phi & -c\psi s\phi + s\psi s\theta c\phi \\ -s\theta & c\theta s\phi & c\theta c\phi \end{bmatrix} \tag{7}$$

$$T_{B\to E} = \begin{bmatrix} 1 & s\phi t\theta & c\phi t\theta \\ 0 & c\phi & -s\phi \\ 0 & s\phi/c\theta & c\phi/c\theta \end{bmatrix} \tag{8}$$

Το σύνολο των εξωτερικών δυνάμεων και ροπών ($\mathbf{F}_B$, $\boldsymbol{\tau}_B$) που ασκούνται στο σύστημα, εκφράζονται στο σωματόδετο σύστημα ως:

$$\mathbf{F}_B = \mathbf{m}\dot{\boldsymbol{\nu}}_B + \boldsymbol{\omega}_B \times \mathbf{m}\boldsymbol{\nu}_B \tag{9}$$

$$\boldsymbol{\tau}_B = \mathbf{I}\dot{\boldsymbol{\omega}}_B + \boldsymbol{\omega}_B \times \mathbf{I}\boldsymbol{\omega}_B, \tag{10}$$

όπου $\mathbf{m}$ ο διαγώνιος πίνακας μάζας και $\mathbf{I}$ ο τανυστής αδράνειας, που για τις εφαρμογές της παρούσας εργασίας θεωρείται επίσης διαγώνιος. Παρατίθονται στη συνέχεια οι διαφορετικές δυνάμεις και ροπές που ασκούνται στο σύστημα, από τις οποίες οι δύο πρώτες είναι οι κυριότερες και οφείλονται στην περιστροφή των ελίκων.

Ωστική δύναμη:

$$\mathbf{T} = \sum_{i=1}^{n} K_T \omega_i^2 \mathbf{e}_i^B \tag{11}$$

Ροπή αντίστασης:

$$\boldsymbol{\tau} = \sum_{i=1}^{n} K_Q sgn(\omega_i) \omega_i^2 \mathbf{e}_i^B \tag{12}$$

Ροπή λόγω ωστικής δύναμης:

$$\boldsymbol{\tau}_T = \sum_{i=1}^{n} \mathbf{r}_i \times \mathbf{T}_i \tag{13}$$

Στις παραπάνω σχέσεις, ως K_T αναφέρεται η σταθερά ώσης και ως K_Q η σταθερά ροπής, $\mathbf{e}_i^B = [0\ 0\ 1]$ είναι το μοναδιαίο διάνυσμα του άξονα περιστροφής των ελίκων ως προς το σωματόδετο σύστημα, ω_i είναι η ταχύτητα περιστροφής κάθε προπέλας και η συνάρτηση του προσίμου της:

$$sgn(\omega_i) = \begin{cases} + & \text{για ωρολογιακή φορά περιστροφής} \\ - & \text{για αντιωρολογιακή φορά περιστροφής} \end{cases}$$

Το διάνυσμα $\mathbf{r}_i$, εκφράζει την θέση του κέντρου κάθε έλικας ως προς το σωματόδετο σύστημα συντεταγμένων.

Βαρύτητα (εκφρασμένη στο σύστημα B):

$$\mathbf{F_g} = mg \begin{bmatrix} s\theta \\ -s\phi c\theta \\ -c\phi c\theta \end{bmatrix} \tag{14}$$

Γυροσκοπικά φαινόμενα:

$$\boldsymbol{\tau}_{gyr} = \sum_{i=1}^{n} \boldsymbol{\omega}_B \times J_{rot}\omega_i, \tag{15}$$

όπου J_{rot} η ροπή αδράνειας μίας έλικας.

Αεροδυναμικές αντιστάσεις

$$\mathbf{F_{aero}} = - \begin{bmatrix} K_{d_u} & 0 & 0 \\ 0 & K_{d_v} & 0 \\ 0 & 0 & 0 \end{bmatrix} \boldsymbol{\nu}_B, \tag{16}$$

όπου K_{d_u} και K_{d_v} σταθερές.

Λαμβάνοντας, λοιπόν, όλα αυτά υπόψη, και σε συνδυασμό με τις διαφορικές κινηματικές εξισώσεις (5) και (6), το συνολικό μοντέλο, εκφρασμένο σε μητρώα, προκύπτει:

$$\begin{bmatrix} \dot{u} \\ \dot{v} \\ \dot{w} \end{bmatrix} = \begin{bmatrix} rv - qw \\ pw - ru \\ qu - pv \end{bmatrix} + \begin{bmatrix} gs\theta \\ -gs\phi c\theta \\ -gc\phi c\theta \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ K_T/m \sum_{i=1}^{n} \omega_i^2 \end{bmatrix} - \begin{bmatrix} Kd_u/m & 0 & 0 \\ 0 & Kd_v/m & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} u \\ v \\ w \end{bmatrix} \tag{17}$$

$$\begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} = \begin{bmatrix} qr(I_{yy} - I_{zz})/I_{xx} \\ rp(I_{zz} - I_{xx})/I_{yy} \\ pq(I_{xx} - I_{yy})/I_{zz} \end{bmatrix} + \begin{bmatrix} K_T/I_{xx} \sum_{i=1}^{n} r_{y_i}\omega_i^2 \\ K_T/I_{yy} \sum_{i=1}^{n} r_{x_i}\omega_i^2 \\ K_Q/I_{zz} \sum_{i=1}^{n} sgn(\omega_i)\omega_i^2 \end{bmatrix} + \begin{bmatrix} -qJ_{rot}/I_{xx} \sum_{i=1}^{n} sgn(\omega_i)\omega_i \\ pJ_{rot}/I_{yy} \sum_{i=1}^{n} sgn(\omega_i)\omega_i \\ 0 \end{bmatrix} \tag{18}$$

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix} = \begin{bmatrix} c\psi c\theta & -s\psi c\phi + c\psi s\theta s\phi & s\psi s\phi + c\psi s\theta c\phi \\ s\psi c\theta & c\psi c\phi + s\psi s\theta s\phi & -c\psi s\phi + s\psi s\theta c\phi \\ -s\theta & c\theta s\phi & c\theta c\phi \end{bmatrix} \begin{bmatrix} u \\ v \\ w \end{bmatrix} \tag{19}$$

$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} 1 & s\phi t\theta & c\phi t\theta \\ 0 & c\phi & -s\phi \\ 0 & s\phi/c\theta & c\phi/c\theta \end{bmatrix} \begin{bmatrix} p \\ q \\ r \end{bmatrix} \tag{20}$$

Πρόκειται, δηλαδή, για ένα μη γραμμικό σύστημα 12×12, με διάνυσμα καταστάσεων $X = [u\ v\ w\ p\ q\ r\ x\ y\ z\ \phi\ \theta\ \psi]^T$ και διάνυσμα εισόδων $U = [\omega_1\ \omega_2\ ...\ \omega_n]^T$.
Όλα τα αθροίσματα $\sum_{i=1}^{n}$ που εμφανίζονται, αναπτύσονται ανάλογα με τον αριθμό n των ελίκων. Είναι προφανές, ότι το γενικό αυτό μοντέλο μπορεί να εφαρμοστεί ανεξάρτητα του αριθμού αυτού.

Αναγνώριση Συστήματος

Η Αναγνώριση Συστήματος (System Identification) είναι μια μέθοδος εύρεσης μαθηματικών μοντέλων που περιγράφουν ένα δυναμικό σύστημα, που χρησιμοποιεί πειραματικά μετρούμενα δεδομένα για την εκτίμηση των άγνωστων παραμέτρων. Αρχικά, απαιτείται η συλλογή πλήθους δεδομένων, τα οποία πρέπει να φέρουν αξιόλογη πληροφορία για το σύστημα. Ο κατάλληλος σχεδιασμός των εισόδων ενός συστήματος, μπορεί πράγματι να οδηγήσει σε επαρκή διέγερσή όλων των ιδιοτήτων του. Σήματα που καλύπτουν μεγάλο εύρος συχνοτήτων έχουν αποδειχθεί πως είναι τα ιδανικότερα. Επιλέγονται κυρίως ημιτονοειδείς μορφές με διαφορετικές συχνότητες και sweep signals, σήματα δηλαδή που «σαρώνουν» ένα μεγάλο εύρος συχνοτήτων σε καθορισμένο χρόνο.
Σε επόμενο βήμα πρέπει να γίνει η επιλογή μιας υποψήφιας δομής μοντέλου, στο οποίο θα πρέπει να προσαρμοστούν τα δεδομένα. Οι παράμετροι που καλούνται να αναγνωριστούν για το υπο εξέταση μοντέλο είναι οι $K_{d_u}, K_{d_v}, K_T, K_Q, I_{xx}, I_{yy}, I_{zz}$ και J_{rot}. Έτσι, οι εξισώσεις του μοντέλου πρέπει να προσαρμοστούν με τέτοιο τρόπο ώστε να βρεθεί μια μορφή που να είναι γραμμική ως προς τις παραμέτρους αυτές. Εφόσον οι παραπάνω παράμετροι εμφανίζονται μόνο στις δυναμικές και όχι στις κινηματικές διαφορικές εξισώσεις, μόνο αυτές χρησιμοποιούνται για την νέα μορφή του μοντέλου. Προκύπτει, λοιπόν η μορφή $\mathbf{Y} = \mathbf{\Phi}\theta$, όπου οι επιμέρους πίνακες είναι:

$$\mathbf{Y} = \begin{bmatrix} m(\dot{u} + qw - rv - gs\theta) \\ m(\dot{v} + ru - pw + gs\phi c\theta) \\ m(\dot{w} + pv - qu + gc\phi c\theta \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{21}$$

$$\mathbf{\Phi} = \begin{bmatrix} -u & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -v & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \sum {\omega_i}^2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -\sum r_{y_i} {\omega_i}^2 & 0 & \dot{p} & -qr & qr & q\sum sgn(\omega_i)\omega_i \\ 0 & 0 & \sum r_{x_i} {\omega_i}^2 & 0 & rp & \dot{q} & -rp & -p\sum sgn(\omega_i)\omega_i \\ 0 & 0 & 0 & -\sum sgn(\omega_i){\omega_i}^2 & -pq & -pq & \dot{r} & 0 \end{bmatrix} \tag{22}$$

$$\boldsymbol{\theta} = \begin{bmatrix} K_{d_u} & K_{d_v} & K_T & K_Q & I_{xx} & I_{yy} & I_{zz} & J_{rot} \end{bmatrix}^T \tag{23}$$

Ως κριτήριο για την προσαρμογή του παραπάνω μοντέλου στα δεδομένα, επιλέγεται η μέθοδος των Ελαχίστων Τετραγώνων, με βάση την οποία το διάνυσμα θ των παραμέτρων μπορεί να εκτιμηθεί με χρήση της σχέσης:

$$\hat{\boldsymbol{\theta}} = [\mathbf{\Phi}^T\mathbf{\Phi}]^{-1}\mathbf{\Phi}^T\mathbf{Y} = \mathbf{\Phi}^\dagger\mathbf{Y}, \tag{24}$$

στην οποία, ο πίνακας $\mathbf{\Phi}^\dagger$ που εμφανίζεται είναι ο ψευδοαντίστροφος Moore-Penrose. Όταν παραπάνω από ένα πειράματα διεξάγονται για την συλλογή δεδομένων, οι επι-

μέρους εκτιμήσεις συνδυάζονται με χρήση της σχέσης:

$$\hat{\boldsymbol{\theta}} = \boldsymbol{P}\sum_{i=1}^{n}(\boldsymbol{P_i})^{-1}\hat{\boldsymbol{\theta}}_{\boldsymbol{i}} \tag{25}$$

$$\mathbf{P} = \left[\sum_{\mathbf{i=1}}^{\mathbf{n}}(\mathbf{P_i})^{-1}\right]^{-1} \tag{26}$$

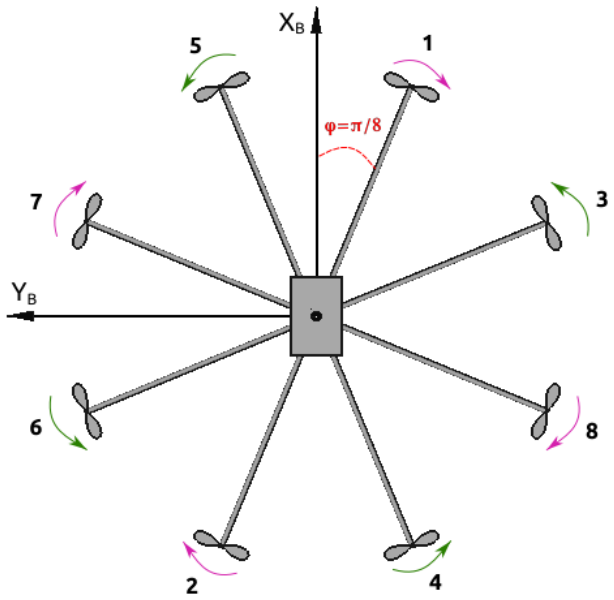
P_i είναι ο πίνακας συμμεταβλητότητας των παραμέτρων του κάθε επιμέρους πειράματος και υπολογίζεται :

$$\boldsymbol{P}(\hat{\boldsymbol{\theta}}) = \frac{\mathbf{1}}{\boldsymbol{N}}\|\mathbf{Y} - \boldsymbol{\Phi}\hat{\boldsymbol{\theta}}\|^{\mathbf{2}}(\boldsymbol{\Phi}^{\boldsymbol{T}}\boldsymbol{\Phi})^{-\mathbf{1}} \tag{27}$$

Στη συνέχεια παρουσιάζεται η εφαρμογή της διαδικασίας αναγνώρισης σε δύο διαφορετικά μοντέλα εναέριων οχημάτων, με διαφορετικό αριθμό ελίκων.

Οκτακόπτερο όχημα σε περιβάλλον MATLAB®

Στην πρώτη αυτή περίπτωση μελετάται ένα οκτακόπτερο (octorotor) σε διάταξη X, όπως φαίνεται στο Σχήμα (2). Θεωρείται πως οι έλικες είναι συμμετρικά κατανεμημένες με μεταξύ τους γωνία $\pi/4$, και έτσι δεδομένου μήκους $d = 0.4m$ για κάθε ράβδο, είναι έυκολο να υπολογιστούν τα διανύσματα r_i της θέσης κάθε έλικας σε σχέση με το κέντρο μάζας.



Σχήμα 2: Οκτακόπτερο σε διάταξη X.

Το μοντέλο προσομειώνεται με χρήση της MATLAB®, χωρίς την ενσωμάτωση ελεγκτή, κι έτσι απευθείας τιμές των είσοδων U μπορούν να δίνονται στο σύστημα. Αυτό φυσικά δεν εξασφαλίζει την ομαλή και ευσταθή «πτήση» του οχήματος, αλλά τουλάχιστον μπορεί να παρέχει δεδομένα για τις αποκρίσεις των καταστάσεων X.

Δύο κατηγορίες εισόδων δοκιμάστηκαν αρχικά για την συλλογή δεδομένων· τυχαίες βηματικές είσοδοι και συνημιτονοειδείς είσοδοι με τυχαία πλάτη και συχνότητες. Τα αποτελέσματα της διαδικασίας αναγνώρισης με αυτά τα δεδομένα παρουσιάζονται στον Πίνακα (1), μαζί με τις πραγματικές τιμές των παραμέτρων που εισήχθησαν για την εκτέλεση των προσομοιώσεων.

Πίνακας 1: Αποτελέσματα διαδικασίας Αναγνώρισης Συστήματος.

	Βηματικές είσοδοι	Συνημιτονειδείς είσοδοι	Συνδυασμός	**Ακριβείς τιμές**
K_{d_u}	0.29975	0.29993	0.29992	**0.3**
K_{d_v}	0.29838	0.3	0.29997	**0.3**
K_T	2.2001e-05	2.2001e-05	2.2001e-05	**2.2e-05**
K_Q	4.5002e-07	4.5002e-07	4.5002e-07	**4.5e-07**
I_{xx}	0.10901	0.10901	0.10901	**0.109**
I_{yy}	0.10801	0.10801	0.10801	**0.108**
I_{zz}	0.20801	0.20801	0.20801	**0.208**
J_{rot}	2.0003e-05	2.0001e-05	2.0001e-05	**2e-05**

Η εκτίμηση των παραμέτρων έγινε, όπως φαίνεται, με μεγάλη ακρίβεια, και δεν χρειάζεται η δοκιμή πιο περίπλοκων εισόδων ή η εξακρίβωση ($Validation$) των αποτελεσμάτων. Φυσικά, η ίδια διαδικασία δεν μπορεί να εφαρμοστεί στην πράξη, αποτελεί, ωστόσο, μια απόδειξη για την αποτελεσματικότητα της μεθόδου και μια βάση για την συνέχεια της μελέτης.

Τετρακόπτερο όχημα σε περιβάλλον ROS

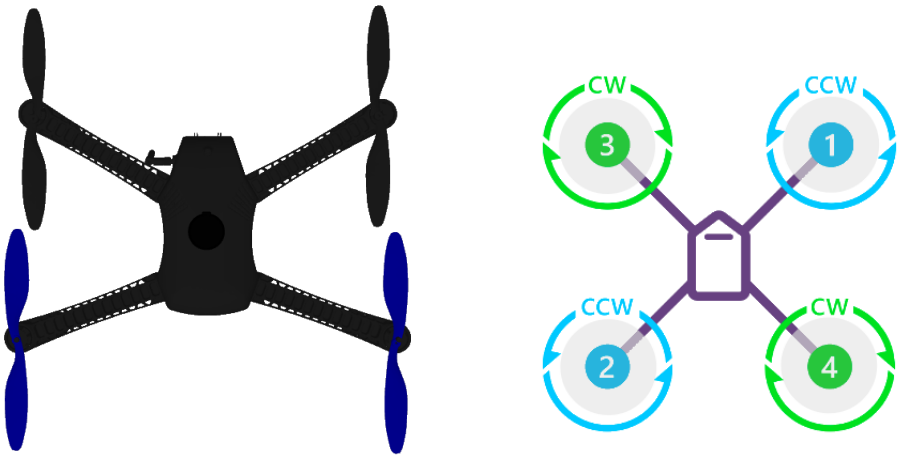
Προσπαθώντας να προσομοιωθεί η πραγματική λειτουργία ενός UAV, χρησιμοποιείται σε δεύτερη φάση, ένα μοντέλο τετρακοπέρου (quadrotor) σε διάταξη X, στο περιβάλλον του ROS. Το μοντέλο αυτό είναι μάλιστα εξοπλισμένο με εικονικό αυτόματο πιλότο ArduPilot, ο οποίος αποτελείται από δύο επιμέρους βρόχους, έναν που αφορά τα γραμμικά και έναν που αφορά τα γωνιακά μεγέθη και τον έλεγχο τους.
Από την διαμόρφωση του μοντέλου στο ROS, μπορούν να γίνουν γνωστά κάποια από τα χαρακτηριστικά του, αυτά που παρουσιάζονται στη συνέχεια. Οι υπόλοιπες παράμετροι είναι άγνωστες.

$$m = 1.9\ Kg$$

$$Jrot = 1.67604e\text{-}4\ Kgm^2$$

$$I = \begin{bmatrix} 0.0127 & 0 & 0 \\ 0 & 0.0178 & 0 \\ 0 & 0 & 0.0242 \end{bmatrix} Kgm^2$$

$$\begin{aligned} r_1 &= [\ 0.13 \quad -0.22 \quad 0.023]^T\ m \\ r_2 &= [\ -0.13 \quad 0.2 \quad 0.023]^T\ m \\ r_3 &= [\ 0.13 \quad 0.22 \quad 0.023]^T\ m \\ r_4 &= [\ -0.13 \quad -0.2 \quad 0.230]^T\ m \end{aligned}$$



(α΄) Απεικόνηση του τετρακοπτέρου. (β΄) Αρίθμηση ελίκων και φορές περιστροφής.

Σχήμα 3: Το μοντέλο του τετρακοπτέρου.

Πλήθος αισθητήρων προσομοιώνονται επίσης, μέσω των οποίων μπορεί να γίνει η εκτίμηση καταστάσεων του οχήματος κατά την διάρκεια της «πτήσης». Οι μετρήσεις από διαφορετικούς αισθητήρες συγχωνεύονται, και με χρήση φίλτρου Kalman, υπολογίζονται τελικά εκτιμήσεις της θέσης, του προσανατολισμού και των ταχυτήτων του τετρακοπτέρου. Η μέτρηση των γραμμικών επιταχύνσεων γίνεται απευθείας, με χρήση επιταχυνσιομέτρου. Η απόκτηση των δεδομένων γίνεται με κατάλληλο «κόμβο» του ROS, όπου ορίζεται συχνότητα δειγματοληψίας $\omega_s = 30Hz$, και πρωτού εισαχθούν στο μοντέλο για την Αναγνώριση, υφίστανται μια προεπεξεργασία. Αρχικά, πρέπει να προσαρμοστούν, ώστε να αντιστοιχούν στις ίδιες ακριβώς χρονικές στιγμές, διαγράφοντας ταυτόχρονα διπλότυπες μετρήσεις που μπορεί να οφείλονται σε καθυστερήσεις των αισθητήρων. Τα δεδομένα από το φίλτρο Kalman, περνούν επιπλέον από χαμηλοπερατό φίλτρο με συχνότητα αποκοπής $6Hz$, για την εξάληψη οποιουδήποτε θορύβου από τις μετρήσεις. Για τις γωνιακές επιταχύνσεις, οι οποίες χρειάζονται για την διαδικασία της Αναγνώρισης, αλλά δεν παρέχονται από τους αισθητήρες, επιβάλεται αριθμητική παραγώγιση των τιμών των γωνιακών ταχυτήτων και εξετάζεται η χρήση ενός smoothing φίλτρου.

Οι εντολές στο μοντέλο, δεν δύναται να δοθούν ως εντολές ταχυτήτων στους έλικες, γι΄ αυτό και πρέπει να βρεθεί άλλος τρόπος είσόδων που να διεγείρουν τις συχνότητες του συστήματος. Αξιοποιείται, λοιπόν, η δυνατότητα να στέλνονται σήματα με επιθυμητές τροχιές για τις γραμμικές ταχύτητες και την ταχύτητα εκτροπής μέσω του ROS, με συχνότητα ανανέωσης τους $15Hz$, ώστε να ισχύει το θέωρημα δειγματοληψίας.

Η μορφή αυτών των σημάτων, επιλέχθηκε μέσω μιας διαδικασίας «δοκιμής και λάθους», ώστε να βρεθούν οι τιμές που επιφέρουν ικανοποιητικά αποτελέσματα. Η αξιολόγηση των σημάτων, πριν από την επικύρωση των αποτελεσμάτων, γινόταν με τη χρήση δύο βασικών κριτηρίων. Πρώτον, βλέποντας πόσο επαρκώς ο αυτόματος πιλότος και κατά συνέπεια το όχημα μπορούσε να πραγματοποιήσει τις τροχιές που δίνονταν και δεύτερον, εξετάζοντας τις εκτιμώμενες παραμέτρους, ελέγχοντας αν αυτές λαμβάνουν αρνητικές μη φυσικές τιμές, ή ακόμη και αν διαφέρουν κατά πολύ από τις γνωστές

τιμές. Για τις περιπτώσεις που ξεχώρισαν, τα σήματα λαμβάνουν την μορφή $u = Asin(\omega t) + B$, αλλά διαφοροποιούνται στον τρόπο που οι συχνότητές και τα πλάτη τους μεταβάλλονται κατά τη διάρκεια της προσομοίωσης. Παρατηρήθηκε επίσης, πως όταν ο ρυθμός εκτροπής δεν ταλαντώνεται γύρω από το μηδέν, παράγει καλύτερη εκτίμηση των τιμών, και ειδικά της ροπής αδράνειας I_{zz}.
Το σύνολο των πειραμάτων, μπορεί να χωριστεί σε τρεις κατηγορίες, ανάλογα με τον τρόπο που διαμορφώνονται οι συχνότητες και τα πλάτη κατά την διάρκεια του κάθε πειράματος. Σε κάθε περίπτωση, οι τιμές των συχνοτήτων κυμαίνονται από $0.6-10Hz$ και τα πλάτη $0-3m$.

1. Συχνότητες που αυξάνονται βηματικά σε συγκεκριμένες χρονικές περιόδους και σταθερά πλάτη κατά την διάρκεια του πειράματος. Η ταχύτητα εκτροπής ταλαντώνεται γύρω από μια επίσης βηματικά αυξανόμενη τιμή.

2. Συχνότητες που αυξάνονται βηματικά σε συγκεκριμένες χρονικές περιόδους και για τις εντολές των γραμμικών ταχυτήτων πλάτη που μειώνονται γραμμικά.

3. Συχνότητες που αυξάνονται γραμμικά ή εκθετικά, δημιουργόντας τα Sweep signals, όπως αναφέρθηκαν, τα οποία σαρώνουν εύρος συχνοτήτων $0.6 - 10Hz$.

Επιλέγονται, τέλος, οι κατάλληλοι συνδιασμοί πειραμάτων ώστε με χρήση της Εξίσωσης (25), να προκύψει η τελική εκτίμηση των παραμέτρων. Με $\hat{\theta}_{smooth}$ συμβολίζεται το διάνυσμα των παραμέτρων που βρέθηκε χρησιμοποιώντας τις φιλτραρισμένες τιμές των γωνιακών επιταχύνσεων, ενώ με $\hat{\theta}_{raw}$ αυτές χωρίς φίλτρο. Πειράματα που ανήκουν στην πρώτη και τη δεύτερη κατηγορία, όπως περιγράφεται παραπάνω, μπορούν να δώσουν καλύτερα αποτελέσματα όσον αφορά στην ροπή αδράνειας I_{zz}, αλλά με «αντάλαγμα» την αρνητική τιμή του J_{rot} (Πίνακας 2). Παρόλα αυτά η λύση δεν μπορεί να θεωρηθεί αποδεκτή, κι έτσι λαμβάνοντας υπόψη περισσότερα πειράματα προκύπτουν οι τελικές εκτιμήσεις των παραμέτρων (Πίνακας 3).

Πίνακας 2: Εκτιμώμενες τιμές των παραμέτρων (καλή εκτίμηση I_{zz}).

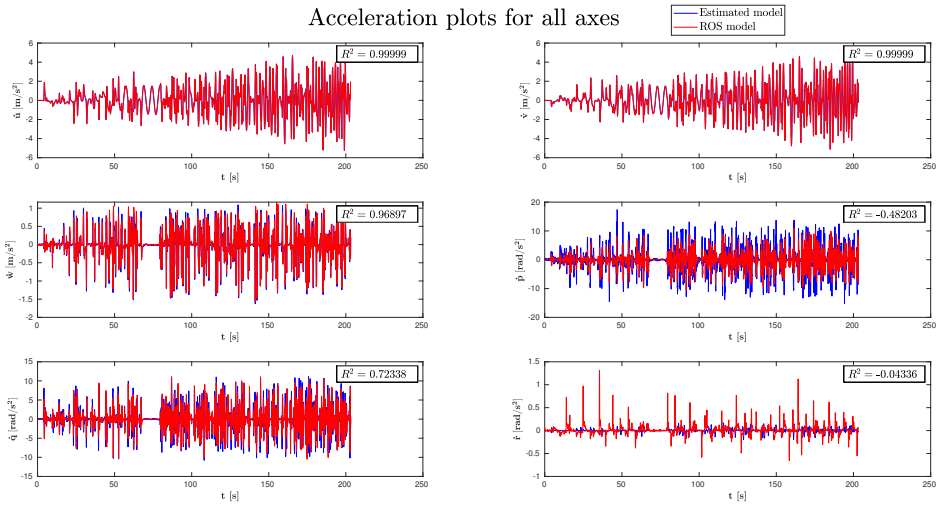
	$\hat{\theta}_{smooth}$	$\hat{\theta}_{raw}$
K_{d_u}	0.0067367	0.0067372
K_{d_v}	0.0059741	0.0059741
K_T	2.1685e-05	2.1685e-05
K_Q	1.0666e-08	9.9358e-09
I_{xx}	0.013728	0.01305
I_{yy}	0.017532	0.016772
I_{zz}	0.023258	0.021431
J_{rot}	-0.00010752	-7.4915e-05

Πίνακας 3: Τελικές τιμές των εκτιμώμενων παραμέτρων.

	$\hat{\theta}_{smooth}$	$\hat{\theta}_{raw}$
K_{d_u}	0.0088314	0.0088328
K_{d_v}	0.0090478	0.0090492
K_T	2.1689ε-05	2.1689ε-05
K_Q	1.5363ε-09	1.421ε-09
I_{xx}	0.013845	0.01296
I_{yy}	0.017624	0.016738
I_{zz}	0.014848	0.013394
J_{rot}	0.00017439	0.00019407

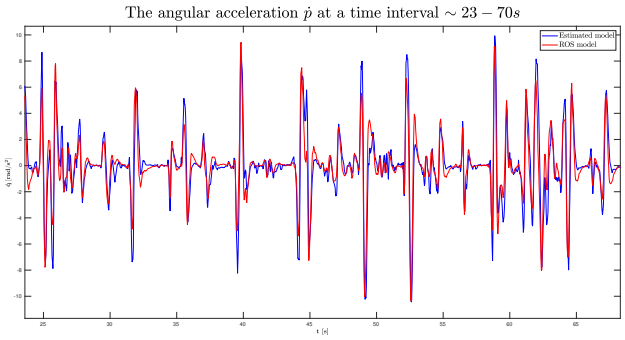
Οι τιμές που βρέθηκαν, περνάνε στην συνέχεια και από μία διαδικασία επικύρωσης (Validation), ώστε να ελεγχθεί η συμπεριφορά του μοντέλου, δεδομένων των παραμέτρων. Για τον σκοπό αυτό, χρησιμοποιούνται οι Εξισώσεις (17) και (18), ώστε να συγκριθούν οι τιμές των παραγώγων που προκύπτουν με αντικατάσταση των σχέσεων, με τις μετρούμενες από τους αισθητήρες τιμές. Φυσικά, για τα γωνιακά μεγέθη

δεν υπάρχουν μετρήσεις από αισθητήρες, και έτσι χρησιμοποιούνται οι φιλτραρισμένες παράγωγοι των γωνιακών ταχυτήτων. Όλα τα υπόλοιπα μεγέθη που εμφανίζονται στις εξισώσεις αυτές, είναι τα μεγέθη που προέκυψαν από τους αισθητήρες.



Σχήμα 4: Σύγκριση των τιμών των επιταχύνσεων από το αρχικό μοντέλο με αυτές που προέκυψαν από τις εκτιμώμενες παραμέτρους. Οι τιμές του συντελεστή προσδιορισμού R^2 έχουν υπολογιστεί και παρουσιάζονται για κάθε άξονα.

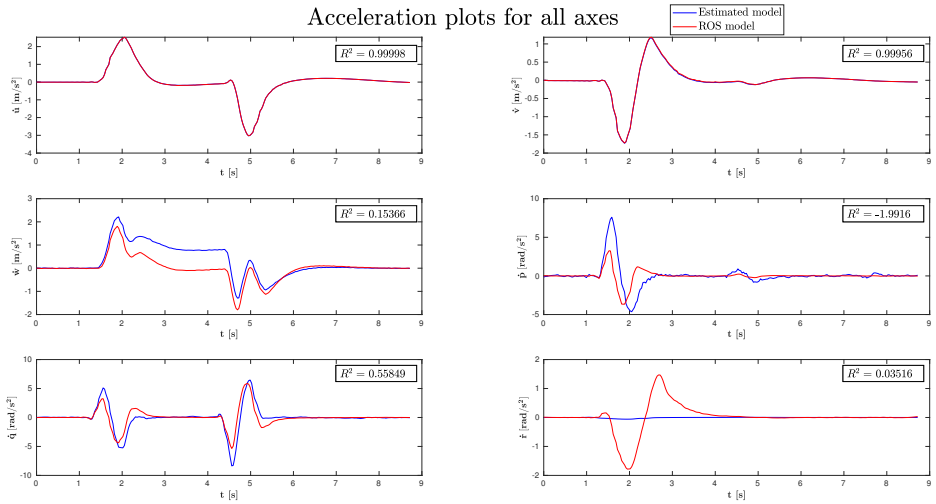
Από τις τιμές του R^2 φαίνεται πως οι παράμετροι που εμπλέκονται στις γραμμικές εξισώσεις (K_{d_u}, K_{d_v} και K_T), έχουν εκτιμηθεί με αρκετά μεγάλη ακρίβεια, ενώ για τα γωνιακά μεγέθη φαίνεται να μην υπάρχει το ίδιο καλή συσχέτιση. Βέβαια, ανεξαρτήτως της τιμής του R^2, η γωνιακή επιτάχυνση $\dot{p}$ που υπολογίστηκε με τις παραμέτρους φαίνεται να πλησιάζει αρκετά την αντίστοιχη απόκριση του μοντέλου, όπως φαίνεται σε λεπτομέρεια του παραπάνω γραφήματος στο Σχήμα (5). Το κυρίως πρόβλημα εντοπίζεται στην γωνιακή επιτάχυνση $\dot{r}$.



Σχήμα 5: Λεπτομέρεια του γραφήματος του $\dot{p}$.

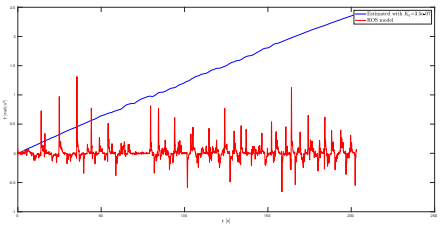
Σε μία ακόμα προσπάθεια επικύρωσης, απλούστερες εντολές εισόδων δόθηκαν στο σύστημα και οι αντίστοιχες αποκρίσεις των επιταχύνσεων φαίνονται στο Σχήμα (6). Στην περίπτωση αυτή, η γραμμική επιτάχυνση $\dot{w}$ δεν μπορεί να προβλεφθεί τόσο αποτελεσματικά όσο προηγουμένως, πιθανότατα επειδή το μοντέλο δεν είχε εκπαιδευτεί

για αυτό το είδος κινήσεων. Φαίνεται κι εδώ, πως παρά τις τιμές του R^2 οι αποκρίσεις των $\dot{p}$ και $\dot{q}$, μπορούν να ακολουθηθούν από το μοντέλο με τις νέες παραμέτρους.

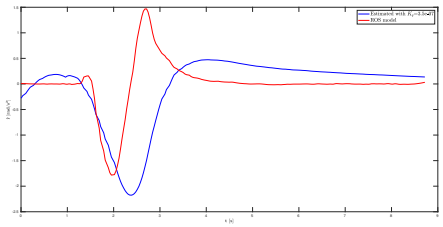


Σχήμα 6: Γραφήματα των τιμών των επιταχύνσεων, συγκριτικά για το αρχικό μοντέλο και τις τιμές που προκέψαν από τις εκτιμώμενες παραμέτρους σε μια δεύτερη προσπάθεια επικύρωσης.

Λαμβάνοντας υπόψη τη συνολική διαδικασία Αναγνώρισης Συστήματος και ειδικότερα τις δύο προσπάθειες επικύρωσης που παρουσιάστηκαν παραπάνω, ορισμένα γενικά σχόλια σχετικά με τις εκτιμώμενες παραμέτρους μπορούν τώρα να σημειωθούν. Οι αεροδυναμικοί συντελεστές K_{d_u} και K_{d_v}, παρόλο που η εκτίμησή τους δεν συγκλίνει σε συγκεκριμένες τιμές στο σύνολο των πειραμάτων, λαμβάνουν πάντοτε μικρές τιμές της ίδιας τάξης μεγέθους. Συμπεραίνεται λοιπον, πως οι όροι αυτοί έχουν μικρή συμβολή στο μοντέλο, και στις αποκρίσεις των $\dot{u}$ και $\dot{v}$. Όσον αφορά στις παραμέτρους K_T και K_Q, οι εκτιμήσεις των τιμών φαίνεται να είναι λιγότερο ακριβείς. Αυτό οφείλεται, αφενός, στο γεγονός πως οι τύποι υπολογισμού της ωστικής δύναμης και τις ροπής αντίστασης βασίζονται σε ορισμένες παραδοχές, οι οποίες μπορεί να μην να πληρούνται πλήρως σε όλες τις προσομοιώσεις, και αφετέρου, στον τρόπο που το ROS, υπολογίζει τα αντίστοιχα μεγέθη μέσω πολικών διαγραμάττων ακολουθώντας διαφορετικό τρόπο μοντελοποίησης. Έτσι, σε διαφορετικές συνθήκες λειτουργίας οι τιμές μπορεί να είναι διαφορετικές. Συγκεκριμένα για τον συντελεστή K_Q, που επηράζει μόνο την απόκριση του $\dot{r}$, παρατηρείται ότι η τιμή που βρέθηκε δεν είναι σε θέση να προβλέψει την κίνηση κατά μήκος αυτού του άξονα. Για να ελεγχθεί αν τουλάχιστον η τάξη μεγέθους της εκτιμώμενης τιμής είναι σωστή, έγινε δοκιμή χρησιμοποιώντας μεγαλύτερο K_Q και επανελέγχοντας τις αποκρίσεις. Συμπεραίνεται, έτσι πως η αδυναμία εύρεσης μιας «σωστής» τιμής για τον συντελεστή αυτό, δεν οφείλεται σε λάθος της διαδικασίας και της συλλογής δεδομένων, αλλά πιθανότατα δεν υπάρχει σταθερή τιμή που να επαρκεί για την περιγραφή όλων των ειδών κινήσεων και ελιγμών.



(α′) Επίδραση μεγαλύτερης τιμής K_Q στο πρώτο πείραμα επικύρωσης.



(β′) Επίδραση μεγαλύτερης τιμής K_Q στο δεύτερο πείραμα επικύρωσης.

Σχήμα 7: Αποκρίσεις με αλλαγή της τιμής του συντελεστή ροπής σε K_Q =3.5e-07, για τα δύο πειράματα επικύρωσης.

Συνολικά για τις εκτιμώμενες τιμές των ροπών αδράνειας, συμπεραίνεται πως οι τιμές βρίσκονται ικανοποιητικά. Παρότι οι τιμές του R^2 δεν είναι ικανοποιητικές για όλες τις εξισώσεις των γωνιακών μεγεθών, οι γνωστές τιμές των παραμέτρων υποδεικνύουν καλή εκτίμηση τους από το μοντέλο, εκτός από την τιμή του I_{zz}. Στην προκειμένη περίπτωση, η τιμή του Πίνακα (2), μπορεί να χρησιμοποιηθεί ως εκτίμηση της συγκεκριμένης παραμέτρου.

List of Figures

List of Tables

References

[1] M. Sivakumar and N. Malleswari, "A literature survey of unmanned aerial vehicle usage for civil applications," *Journal of Aerospace Technology and Management*, vol. 13, 2021.

[2] N. Elmeseiry, N. Alshaer, and T. Ismail, "A detailed survey and future directions of unmanned aerial vehicles (uavs) with potential applications," *Aerospace*, vol. 8, no. 12, 2021.

[3] M. H. Sadarey, *Design of Unmanned Aerial Systems*. John Wiley & Sons Ltd, 2020.

[4] K. Nonami, F. Kendoul, S. Suzuki, W. Wang, and D. Nakazawa, *Autonomous Flying Robots: Unmanned Aerial Vehicles and Micro Aerial Vehicles*. Springer Japan, 2010.

[5] S. Badr, O. Mehrez, and A. E. Kabeel, "A design modification for a quadrotor uav: modeling, control and implementation," *Advanced Robotics*, vol. 33, no. 1, pp. 13–32, 2019.

[6] Q. Quan, *Introduction to Multicopter Design and Control*. Springer,Singapore, 2017.

[7] H. Huang, G. M. Hoffmann, S. L. Waslander, and C. J. Tomlin, "Aerodynamics and control of autonomous quadrotor helicopters in aggressive maneuvering," in *2009 IEEE International Conference on Robotics and Automation*, pp. 3277–3282, 2009.

[8] X. Zhang, X. Li, K. Wang, and Y. Lu, "A survey of modelling and identification of quadrotor robot," *Abstract and Applied Analysis*, vol. 2014, p. 320526, Oct 2014.

[9] N. V. Hoffer, C. Coopmans, A. M. Jensen, and Y. Chen, "A survey and categorization of small low-cost unmanned aerial vehicle system identification," *Journal of Intelligent & Robotic Systems*, vol. 74, pp. 129–145, Apr 2014.

[10] "Arducopter-official documentation." `https://ardupilot.org/copter/index.html/`.

[11] B. Siciliano, L. Sciavicco, L. Villani, and G. Oriolo, *Robotics*. Springer, London, 2009.

[12] J. G. Leishman, *Principles of helicopter aerodynamics*. Cambridge University Press, 2 ed., 2006.

[13] G. Fay, "Derivation of the aerodynamic forces for the mesicopter simulation," 2001.

[14] R. Mahony, V. Kumar, and P. Corke, "Multirotor aerial vehicles: Modeling, estimation, and control of quadrotor," *IEEE Robotics Automation Magazine*, vol. 19, no. 3, pp. 20–32, 2012.

[15] N. Osmić, M. Kurić, and I. Petrović, "Detailed octorotor modeling and pd control," in *2016 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, pp. 002182–002189, 2016.

[16] A. Rodić and G. Mester, "The modeling and simulation of an autonomous quadrotor microcopter in a virtual outdoor scenario," *Acta Polytechnica Hungarica*, vol. 8, pp. 107–122, 01 2011.

[17] "Uav simulator featuring arducopter and mavros integration." `https://github.com/sotomotocross/csl_uav_simulator/`.

[18] "Arduplane, arducopter, ardurover, ardusub source." `https://github.com/ArduPilot/ardupilot/`.

[19] "Pixhawk." `https://pixhawk.org//`.

[20] R. Beard and T. McLain, *Small unmanned aircraft: Theory and practice*. 2012.

[21] L. Ljung, *System identification : theory for the user*. Upper Saddle River, NJ: Prentice Hall PTR, 1999.

[22] L. Ljung, "System identification," in *The Control Systems Handbook: Control System Advanced Methods* (W. S. Levine, ed.), ch. 57, CRC Press, 2010.

[23] O. Nelles, "Nonlinear system identification, from classical approaches to neural networks, fuzzy models, and gaussian processes," 01 2020.

[24] F. Kozin and H. Natke, "System identification techniques," *Structural Safety*, vol. 3, no. 3, pp. 269–316, 1986.

[25] R. Remple K. and M. Tischler B., *Aircraft and Rotorcraft System Identification*. American Institute of Aeronautics and Astronautics, Inc., 2006.

[26] E. Lai, "Converting analog to digital signals and vice versa," in *Practical Digital Signal Processing* (E. Lai, ed.), ch. 2, pp. 14–49, Oxford: Newnes, 2003.

[27] R. W. Schafer, "What is a savitzky-golay filter? [lecture notes]," *IEEE Signal Processing Magazine*, vol. 28, no. 4, pp. 111–117, 2011.

[28] L. Rabiner and B. Gold, *Theory and Application of Digital Signal Processing*. Prentice-Hall, INC.

[29] S. Cho, S. Bhandari, F. Sanders, M. Tischler, and K. Cheung, "System identification and controller optimization of coaxial quadrotor uav in hover," 01 2019.

[30] A. Gong, F. Sanders, R. Hess, and M. Tischler, "System identification and full flight-envelope model stitching of a package-delivery octocopter," 01 2019.

[31] *Coefficient of Determination*, pp. 88–91. New York, NY: Springer New York, 2008.

[32] A. Mendes, E. van Kampen, B. Remes, and Q. Chu, "Determining moments of inertia of small uavs: A comparative analysis of an experimental method versus theoretical approaches," in *AIAA Guidance, Navigation, and Control Conference*, 2012.