FACULTY OF ELECTRICAL ENGINEERING, MATHEMATICS
AND COMPUTER SCIENCE

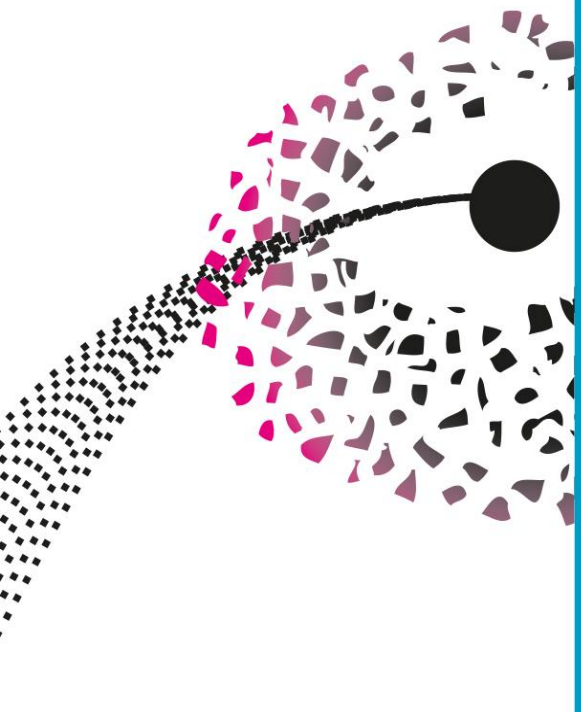
ENERGY-EFFICIENT TRAJECTORY CONTROL FOR VARIABLE-TILT DRONES BY EXPLOITING THE AERODYNAMIC PROXIMITY EFFECT CLOSE TO SURFACES

MASTER THESIS

FABIAN SISAVANH (S2054280)
MSC SYSTEMS & CONTROL

MAI 2022

Committee:
R.G.K.M. Aarts
Prof.dr.ir. A. Franchi
H. Esmaeeli, MSc
Dr.ir. A.Y. Mersha



UNIVERSITY OF TWENTE.

ABSTRACT

In this thesis, an approach has been developed to take advantage of the aerodynamic effects close to surfaces, commonly knows as proximity effect, for a variable tilt multirotor platform. The effect influences the thrust characteristics generated by the propeller, especially when the propeller is in a tilted configuration with respect to the surface. A model describing the proximity effect has been incorporate into the control allocation for an optimal actuation for near ground operation. The same effect can be leverage to minimize the energy consumption by reducing the spinning velocity of the propellers while flying near ground surfaces. To this end, a cost function is designed, and gradient descent optimization is applied to guide the system towards minimum energy consumption. The optimization does not interfere with the primary control object by utilizing the null-space of the control allocation. Several flight scenarios near the ground surface are simulated and compared to a reference scenario without the proximity effect. The results indicate that the controller is capable of rejecting the disturbance cause by the change in thrust and can adapt accordingly. In addition, the energy consumed during the flight period is lower in comparison to the reference. In conclusion, a variable tilt multirotor platform can take advantage of the proximity effect while operating close to ground surfaces to reduce the energy consumption. With the increase of exposer time to surfaces, the energy difference will grow significantly and extend the total flight time. Especially, application in aerial physical interaction have the tendency to fly close to objects or surfaces will benefit from exploiting the proximity effect.

List of Figures

List of Tables

CONTENTS

1 INTRODUCTION

Unmanned aerial vehicles (UAV) have been extensively researched in recent decades, resulting into the development of several types and configurations. In particular, multi rotor aerial vehicles (MRAV) has experienced substantial growth in alternative design approaches to overcome its limitations and open up new fields of application [2]. Up to the present time, they have been equipped with sensing technology in combination with extensive control algorithm to enable flight with minimal human intervention and are utilized in several fields for application such as mapping, localization, surveillance or inspection. However, the majority of those applications are based on sensing the environment by using vision or non-contact sensors and the platform has no physical interaction with the environment.

Meanwhile, the recent trend in research and development shows a growing interest in extending the capabilities of UAV to physically interact and manipulate the environment. This is achieved by integrating robotic manipulators into the platform, and thus establishing the field of aerial robotic manipulators (AEROM) [3]. Consequently, this opens up a variety of possible application such as contact-based inspection, maintenance or object manipulation. This is especially beneficial for operation in high attitude with limited accessibility, for example the inspection and maintenance of a wind turbine. Consequently, AEROM is able minimize time, cost and risk of the people involved.

Typically, in aerial physical interaction (APhI) the target is located on the ground, over a surface, close to a wall or beneath the ceiling. In order to interact or manipulate the target, the drone is required to fly close to the target and perform operations while hovering or moving at low velocity. Under those circumstances, MRAV are a suitable platform with their proven manoeuvrability and agility. Furthermore, they are able to perform vertical takeoff and landing (VTOL) and have the capability to hover in place. Therefore, they are ideal for operation in confined environment.

However, moving towards aerial physical interaction with MRAV several challenges arise and are subject of nowadays research. For an unconstrained interaction between MRAV and environment, it is desired to have the possibility to exert force in arbitrary directions. On the other hand, traditional MRAV with coplanar/collinear propeller design are able to generate thrust in the same direction, thus are only capable to actuate four degree of freedom (DoF) and therefore not able to arbitrary move in three dimension without change of orientation. This type of MRAV is categorized as under-actuated system and constraints the ability to interact with the environment. An unconstrained interaction is possible with the use of fully-actuated platform to control six DoF. This can be achieved with non-coplanar/collinear arrangement or with radial/tangential tilt of the propellers to achieve force-omnidirectionality. But the efficiency of the system will decrease because of counteracting forces to keep the system in equilibrium.

In order to overcome the under-actuated nature while retaining the efficiency of a traditional MRAV, the platform has been extended with additional actuators which are responsible for tangential tilting of the propeller unit. This variable-tilt rotor design enables a more efficient omnidirectional flight and sufficient decoupled force and torque for physical interactions.

1.1 Problem statement

In general, aerial physical interaction requires the platform to fly close to the target for operation, such as touch, grasp or manipulation. Under normal circumstances, the target is located close to the ground, beneath the ceiling or on the wall. Hence, the platform has to fly or hover in proximity to those horizontal or vertical surfaces. In case of a MRAV, the presence of surfaces will obstruct the airflow around the propeller, and thus it changes the aerodynamic characteristic of the thrust generation. For this reason, it is necessary to take the effect into consideration during the control development in order to create a system which is responsive and adaptable to the changes. Especially when it comes to use of variable-tilt MRAV it is essential to have an understanding of the effect and anticipating the changes in order to ensure a safe and stable operation. In addition, it would be beneficial to take advantage of the knowledge to optimize the actuation in order to increase the energy efficiency.

1.2 Related work

Ref.	Author	Year	No. prop	Ctrl	MDT	Var	Prox. effect	Val
[4]	Ryll et al.	2015	4	P	✓	✓	✗	✓
[5]	Kamel et al.	2018	6	P	✓	✓	✗	✓
[6]	Bodie et al.	2020	12	P	✓	✓	✗	✓
[7]	Ryll et al.	2020	6	P	✓	✓	✗	✓
[8]	Allenspach et al.	2020	12	P	✓	✓	✗	✓
[9]	Powers et al.	2013	4	Traj	✗	✗	GE	✓
[10]	Sanchez et al.	2017	4	Traj	✗	✗	GE	✓
[11]	Sanchez et al.	2017	4	Traj	✗	✗	CE	✓
[12]	He et al.	2019	4	Traj	✗	✗	GE	✓
[13]	Gao et al.	2019	4	Traj	✗	✗	GE	✓
[14]	Kan et al.	2019	4	Traj	✗	✗	GE	✓
[15]	Garofano et al.	2021	1	-	✗	✗	WE	CFD
[1]	Matus et al.	2021	4	Traj	✗	✗	GE	✓
[16]	Ding et al.	2022	4	Traj	✓	✗	WE	✓
[-]	This thesis	2022	6	P	✓	✓	✓	Sim

Table 1.1: Literature review on state of the art

Multiple studies have focus on the research and development of multirotor with variable tilt propeller configuration because it combines the advantages of collinear and fix-tilt multirotor configurations. Since traditional collinear multirotor are under actuated

systems, they are only able to perform motion with change in orientation. For example, if the platform wants to perform a horizontal movement in a specific direction, it first needs to roll or pitch in order to redirect the thrust in the appropriate direction. On the order hand, a fully-actuated platform with fix-tilted propeller configuration can make use of the horizontal thrust component to carry out the motion without tilting the system. However, if no motion is required and the platform is hovering, only the vertical thrust components are needed, and the horizontal components are wasted in order to keep the system in equilibrium. This is also referred to as *internal forces* of the systems. In contrast, collinear multirotor designs are utilizing the complete thrust for hovering and consequently are more efficient during hover operation. All things considered, a variable tilt multirotor configuration appears to be an ideal platform to perform interaction operation by varying the tilt angle to be fully actuated for precise motion when needed and revert to a more efficient configuration otherwise.

Variable tilt multirotor is the subject of several studies in recent years. It started with a quadrotor design with limited roll and pitch [4] and was followed by a hexarotor design later on [5]. In order to increase the performance, the hexarotor design has been extended with dual propeller configuration [6][8]. These platforms accomplish force omnidirectionality with high hover efficiency at the expense of added weight, inertia and complex mechanical design. Another design approach is to reduce the actuators responsible for the tilt mechanism. Instead of utilizing up to six additional actuators for individual tilt of each propeller unit, they are coupled by a transmission system to a single actuator [7]. This benefits the mechanical design complexity to some degree, but has a major impact on reducing the total weight of the system. Although this design is capable of generating a variety of force direction efficiently, it sacrifices pose omnidirectionality.

When a multirotor approaches a target for interaction operation, the airflow at the propellers are being influenced by the presence of the surrounding surfaces. The surfaces will constraint the free development of the propeller wake, which will cause changes in propeller thrust. In order to undertake safe operation with MRAV, the effect must be taken into consideration and compensated during the flight. Otherwise, the system stability is at risk. Several studies have been conducted to analyse the characteristic of the effect for standard coplanar multirotor. They have considered surfaces such as ground [9] [10] [1], ceiling [11] and wall [17]. However, the literature on the effect considering tilted propeller configuration is limited [17] [15]. Up to the present time, a model describing the thrust variation with the distance and the inclination angle between propeller and surface has not been presented yet. Nevertheless, the application for platforms with fix- or variable-tilt configuration are increasing, and with it the relevance to analyse the effect in order to ensure stable operation of the platform.

1.3 Thesis contribution

The literature review in Table 1.1 shows that the research on proximity effect for tilted propeller configuration is limited and a control approach for variable-tilt multirotor involving the proximity effect is unexplored. For that reason, this thesis will study the influence of the proximity effect on variable-tilt multirotor control. A control approach for trajectory tracking is proposed, implemented and validated based in a real-time simulation

environment. Furthermore, an optimization is proposed in order to take advantage of the proximity effect and the energy consumption is compared to reference system not under the effect.

1.4 Research question

This thesis aims to address the main question:

- How to exploit the proximity effect for variable tilt-rotor drones in flight operations?

The main question can be separated into two parts:

1. How to modify a controller to take into account the proximity effect to generate optimal thrust close to surfaces?
2. Can the energy consumption of the drone be reduced by optimizing thrust based on the proximity effect?

1.5 Report layout

The rest of the thesis is organized as follows:

Chapter 2 will introduce the relevant theoretical background related to the thesis. It will describe the modelling of a variable tilt multirotor platform together with the actuation principles.

Chapter 3 introduces a proximity effect model for tilted propeller.

Chapter 4 is based on the previous chapter and proposes a trajectory controller.

Chapter 5 shows the implementation of the proposed controller and proximity effect model in Matlab/Simulink environment. In addition, it discusses the simulation results of different flight scenarios and compared it to reference system consider to be in free-flight.

Chapter 6 concludes by summarizing the thesis and recommends future work base on the findings.

2 THEORETICAL BACKGROUND

This chapter provides a brief overview of the theoretical foundation of this thesis.

2.1 System modelling of a variable tilt MRAV

This section introduces a simplified mathematical model of a variable tilt MRAV with coplanar propeller design. The system can be simplified to massless and orientable propeller units P_i connected to a rigid body B. The propeller units are connected to the body by an arm with the length L with a tilting mechanism allowing the propeller units to rotate individually.

2.1.1 Definition and notation

To assure a clarity and consistency throughout the thesis, the following notation are adopted:

- Vectors and matrices will be denoted in bold font, with lower and upper cases, respectively.
- The operator $\bullet^\intercal$ denotes the transpose of vector/matrix $\bullet$
- Unit vectors $\mathbf{e}_1 = \begin{bmatrix}1 & 0 & 0\end{bmatrix}^\intercal$, $\mathbf{e}_2 = \begin{bmatrix}0 & 1 & 0\end{bmatrix}^\intercal$, $\mathbf{e}_3 = \begin{bmatrix}0 & 0 & 1\end{bmatrix}^\intercal \in \mathbb{R}^3$
- Canonical rotation matrix $\mathbf{R}_X(\theta)$, $\mathbf{R}_Y(\theta)$, $\mathbf{R}_Z(\theta) \in \mathbb{R}^{3\times3}$ about **X**, **Y**, **Z** axes with angle θ, respectively.

A world inertial frame is defined as $\mathcal{F}_W : \{O_W; X_W, Y_W, Z_W\}$ and a moving frame $\mathcal{F}_B : \{O_B; X_B, Y_B, Z_B\}$ is rigidly placed on the body of the platform, where O_B coincide at its centre of mass (CoM). In addition, each propeller unit is associated with a frame $\mathcal{F}_{P_i} : \{O_{P_i}; X_{P_i}, Y_{P_i}, Z_{P_i}\}$ where $i = 1 \dots n$ with n referring to the numbers of propeller units. In this particular case of a hexarotor $n = 6$, see Figure 2.1. Furthermore, the tilting axis is associated with X_{P_i} where the tilt angle is described with $\alpha_i \in \mathbb{R}$ and the spinning axis of the propeller is Z_{P_i} generating a thrust force f_{P_i} along the same axis. The relative orientation between frames is described by the rotation matrix $\mathbf{R}^\star_\bullet \in SO(3)$ which denotes the orientation of frame $\bullet$ with respect to (w.r.t) frame $\star$. Therefore, $\mathcal{F}_B$ w.r.t $\mathcal{F}_W$ and $\mathcal{F}_{P_i}$ w.r.t $\mathcal{F}_B$ are represented with $\mathbf{R}^W_B$ and $\mathbf{R}^B_{P_i}$ respectively. Following this, the propeller frame $\mathcal{F}_{P_i}$ is fully described in the body frame with

$$\mathbf{R}^B_{P_i} = \mathbf{R}_Z\left((i-1)\frac{2\pi}{n}\right)\mathbf{R}_X\left(\alpha_i\right),\ i = 1 \dots n \tag{2.1}$$

and

$$ {}^{B}\mathbf{O}_{P_i} = \mathbf{R}_Z\left((i-1)\frac{2\pi}{n}\right)\begin{bmatrix} L \\ 0 \\ 0 \end{bmatrix},\ i = 1\dots n \tag{2.2} $$

where L is the length of each arm from O_B to O_{P_i}. In summery, the system configuration is completely determined given the body position $\mathbf{p} = {}^{W}\mathbf{O}_B \in \mathbb{R}^3$ and the orientation $\mathbf{R}_B^W$ in the world frame and by the tilt angle $\boldsymbol{\alpha} = \begin{bmatrix}\alpha_i \dots \alpha_n\end{bmatrix}^\intercal$ defining the propeller unit orientation $\mathbf{R}_{P_i}^B(\boldsymbol{\alpha})$ in the body frame.

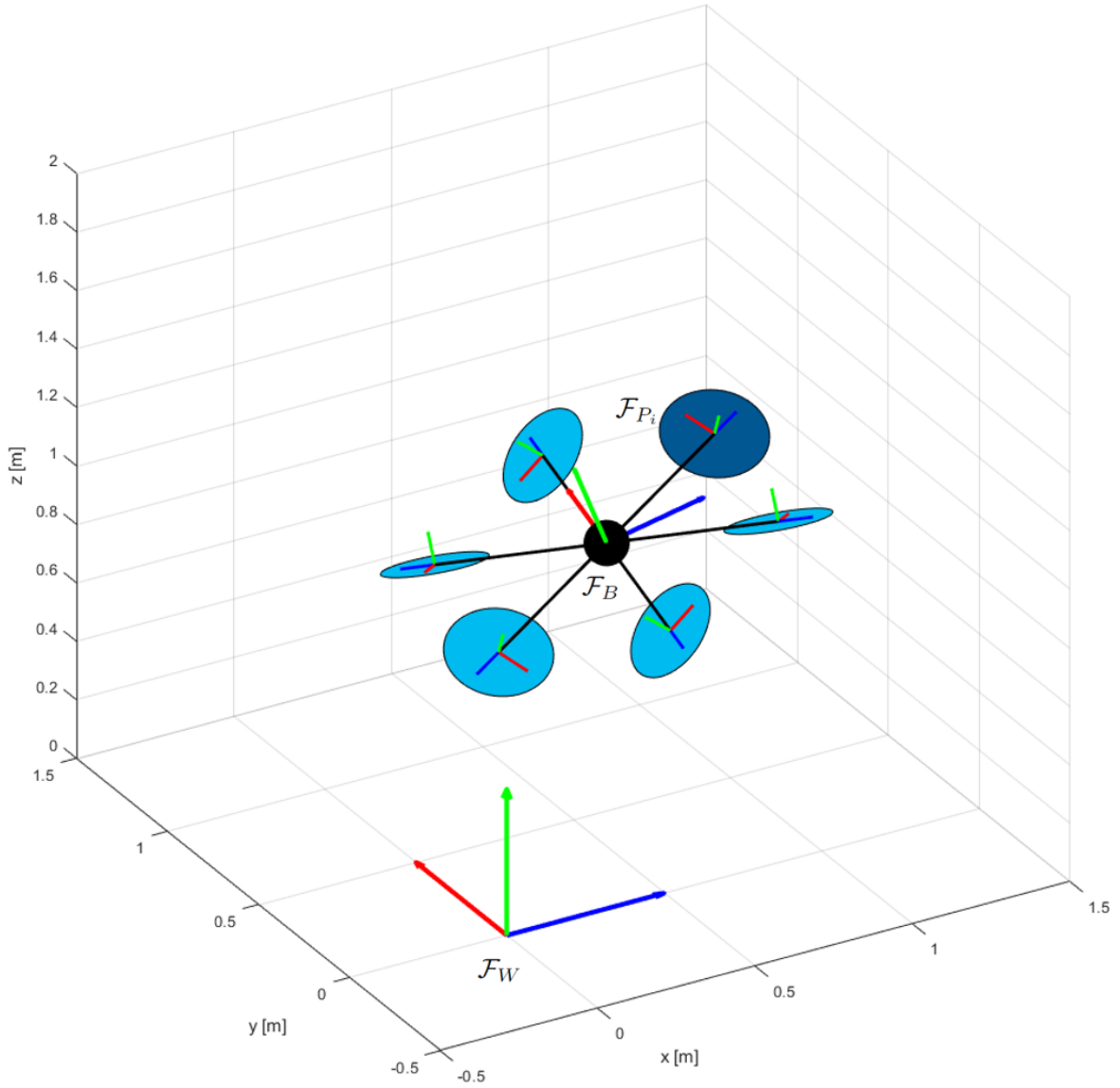


Figure 2.1: Visual representation of the hexarotor with the associated frames

2.1.2 Equation of motion

The dynamic of the system is derived based on the Newton-Euler approach, in which the aerial platform is considered to be a rigid body. The equation of motion of the system

can be compactly expressed as

$$\begin{bmatrix} m\mathbf{I}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbb{J} \end{bmatrix} \begin{bmatrix} \ddot{\boldsymbol{p}} \\ {}^B\dot{\boldsymbol{\omega}} \end{bmatrix} = \begin{bmatrix} -mg\mathbf{e}_3 \\ -\omega \times \mathbb{J}\omega \end{bmatrix} + \begin{bmatrix} \mathbf{R}_B^W & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{I}_3 \end{bmatrix} \begin{bmatrix} {}^B\mathbf{f} \\ {}^B\boldsymbol{\tau} \end{bmatrix} \tag{2.3}$$

where the translational dynamic is described in the world frame and the rotational dynamics in the body frame. m and $\mathbb{J}$ are the mass and inertia of the system, respectively. $\begin{bmatrix} {}^B\mathbf{f} & {}^B\boldsymbol{\tau} \end{bmatrix}^\intercal$ are the forces and moments produced by the actuators in the body frame.

2.1.3 Aerodynamic force and drag model

A rotating propeller generates a thrust $\mathbf{f}_{P_i} \in \mathbb{R}^3$ and a counter-rotating moment $\boldsymbol{\tau}_{P_i} \in \mathbb{R}^3$ caused by air drag. The force and moment are applied on $\mathbf{O}_{P_i}$ along the axis $\mathbf{Z}_{P_i}$ and are modelled as

$$\mathbf{f}_{P_i} = c_{f_i} \cdot \omega_{P_i}^2 \cdot \mathbf{e}_3 \tag{2.4}$$

$$\boldsymbol{\tau}_{P_i} = c_{\tau_i} \cdot \omega_{P_i}^2 \cdot \mathbf{e}_3 \cdot k_{P_i} \tag{2.5}$$

where the thrust coefficient $c_{f_i} \in \mathbb{R}_{>0}$ and drag coefficient $c_{\tau_i} \in \mathbb{R}_{>0}$ are positive constant corresponding to the propeller properties. To account for the direction of the drag moment induced by the spinning direction of the propeller, the factor $k_{P_i} \in \{-1, 1\}$ is introduced. To express the total force and moment generated by the propellers in the body frame, each contribution transformed into the body frame as follows

$${}^B\mathbf{f} = \sum_{i=1}^{n} {}^B\mathbf{f}_i = \sum_{i=1}^{n} \mathbf{R}_{P_i}^B \mathbf{f}_{P_i} \tag{2.6}$$

$${}^B\boldsymbol{\tau} = \sum_{i=1}^{n} {}^B\boldsymbol{\tau}_{f_i} + {}^B\boldsymbol{\tau}_{d_i} = \sum_{i=1}^{n} {}^B\mathbf{O}_{P_i} \times \mathbf{R}_{P_i}^B \mathbf{f}_{P_i} + \mathbf{R}_{P_i}^B \boldsymbol{\tau}_{P_i} \tag{2.7}$$

Because both force and moments are proportional to the squared rotor speed $\omega_{P_i}^2$, it can be defined as a vector

$$\boldsymbol{\Omega} = \begin{bmatrix} \Omega_1 \\ \vdots \\ \Omega_n \end{bmatrix} = \begin{bmatrix} \omega_{P_1}^2 \\ \vdots \\ \omega_{P_n}^2 \end{bmatrix} \tag{2.8}$$

and rewrite Eq.2.6 and Eq.2.7 as a matrix multiplication directly relating the body force and moments to the squared rotor speed as follows:

$$\begin{bmatrix} {}^B\mathbf{f} \\ {}^B\boldsymbol{\tau} \end{bmatrix} = \begin{bmatrix} \mathbf{R}_{P_1}^B \mathbf{e}_3 c_f & \cdots & \mathbf{R}_{P_n}^B \mathbf{e}_3 c_f \\ {}^B\mathbf{O}_{P_1} \times \mathbf{R}_{P_1}^B \mathbf{e}_3 c_f + \mathbf{R}_{P_1}^B \mathbf{e}_3 c_\tau & \cdots & {}^B\mathbf{O}_{P_n} \times \mathbf{R}_{P_n}^B \mathbf{e}_3 c_f + \mathbf{R}_{P_n}^B \mathbf{e}_3 c_\tau \end{bmatrix} \begin{bmatrix} \Omega_1 \\ \vdots \\ \Omega_n \end{bmatrix} \tag{2.9}$$

$$\begin{bmatrix} {}^B\mathbf{f} \\ {}^B\boldsymbol{\tau} \end{bmatrix} = \begin{bmatrix} \mathbf{F}(\alpha) \\ \boldsymbol{\tau}(\alpha) \end{bmatrix} \boldsymbol{\Omega} = \mathbf{A}_\alpha \boldsymbol{\Omega} \tag{2.10}$$

where $\begin{bmatrix}\mathbf{F}(\alpha) & \boldsymbol{\tau}(\alpha)\end{bmatrix}^\intercal = \mathbf{A}_\alpha$ is referred to as the allocation matrix with a dependency on $\boldsymbol{\alpha}$. Substitute Eq.2.10 into the equation of motion results in

$$\begin{bmatrix} m\mathbf{I}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbb{J} \end{bmatrix} \begin{bmatrix} \ddot{\boldsymbol{p}} \\ {}^B\dot{\boldsymbol{\omega}} \end{bmatrix} = \begin{bmatrix} -mg\mathbf{e}_3 \\ -\omega \times \mathbb{J}\omega \end{bmatrix} + \begin{bmatrix} \mathbf{R}_B^W & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{I}_3 \end{bmatrix} \begin{bmatrix} \mathbf{F}(\boldsymbol{\alpha}) \\ \boldsymbol{\tau}(\boldsymbol{\alpha}) \end{bmatrix} \boldsymbol{\Omega} \tag{2.11}$$

which describe the translational and rotational dynamics of the body frame. In summery, the propeller spinning velocity generates thrust forces and moments in order to influence the translation and rotational motion of the body frame B based on the propeller configuration at that instance specified by the tilt angles $\boldsymbol{\alpha}$. It can be simplified to

$$\begin{bmatrix} \ddot{\boldsymbol{p}} \\ {}^B\dot{\boldsymbol{\omega}} \end{bmatrix} = \begin{bmatrix} m\mathbf{I}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbb{J} \end{bmatrix}^{-1} \left(\begin{bmatrix} -mg\mathbf{e}_3 \\ -\omega \times \mathbb{J}\omega \end{bmatrix} + \begin{bmatrix} \mathbf{R}_B^W & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{I}_3 \end{bmatrix} \begin{bmatrix} \mathbf{F}(\boldsymbol{\alpha}) \\ \boldsymbol{\tau}(\boldsymbol{\alpha}) \end{bmatrix} \boldsymbol{\Omega} \right) \tag{2.12}$$

$$= \begin{bmatrix} -g\mathbf{e}_3 \\ \mathbb{J}^{-1}(-\omega \times \mathbb{J}\omega) \end{bmatrix} + \begin{bmatrix} \frac{1}{m}\mathbf{R}_B^W & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbb{J}^{-1} \end{bmatrix} \begin{bmatrix} \mathbf{F}(\boldsymbol{\alpha}) \\ \boldsymbol{\tau}(\boldsymbol{\alpha}) \end{bmatrix} \boldsymbol{\Omega} \tag{2.13}$$

$$= \mathbf{f} + \mathbf{J}\boldsymbol{\Omega} \tag{2.14}$$

where $\mathbf{f} \in \mathbb{R}^{\not 4}$ summarizes the gravitational and gyroscopic effect and $\mathbf{J} \in \mathbb{R}^{6\times n}$ is the output Jacobian. For convenience, an overview of all symbols and definition introduce in this section, is listed in Table 2.1.

Symbols	Set	Definitions
$\mathcal{F}_W$	$\{O_W; X_W, Y_W, Z_W\}$	inertial world frame W
$\mathcal{F}_B$	$\{O_B; X_B, Y_B, Z_B\}$	multirotor body frame B
$\mathcal{F}_{P_i}$	$\{O_{P_i}; X_{P_i}, Y_{P_i}, Z_{P_i}\}$	i-th propeller frame P_i
$\boldsymbol{p}, \dot{\boldsymbol{p}}, \ddot{\boldsymbol{p}}$	$\mathbb{R}^3$	position, velocity, acceleration $\mathcal{F}_B$ w.r.t $\mathcal{F}_W$
${}^B\boldsymbol{\omega}$	$\mathbb{R}^3$	angular velocity $\mathcal{F}_B$ w.r.t $\mathcal{F}_W$ expressed in $\mathcal{F}_B$
${}^B\dot{\boldsymbol{\omega}}$	$\mathbb{R}^3$	angular acceleration $\mathcal{F}_B$ w.r.t $\mathcal{F}_W$ expressed in $\mathcal{F}_B$
$\mathbf{R}_B^W$	$SO(3) := \{\mathbf{R} \in \mathbb{R}^{3\times3}\}$	rotation matrix $\mathcal{F}_B$ w.r.t $\mathcal{F}_W$
$\mathbf{R}_{P_i}^B$	$SO(3) := \{\mathbf{R} \in \mathbb{R}^{3\times3}\}$	rotation matrix $\mathcal{F}_{P_i}$ w.r.t $\mathcal{F}_B$
α_i	$\mathbb{R}$	i-th propeller tilt angle about X_{P_i}
ω_i	$\mathbb{R}$	i-th propeller spinning velocity about Z_{P_i}
$\mathbf{f}_{P_i}$	$\mathbb{R}^3$	i-th propeller thrust along Z_{P_i}
$\boldsymbol{\tau}_{P_i}$	$\mathbb{R}^3$	i-th propeller air drag torque about Z_{P_i}
m	$\mathbb{R}_{>3}$	total mass
$\mathbb{J}$	$\mathbb{R}$	inertia of the multirotor body B
c_f	$\mathbb{R}_{>3}$	propeller thrust coefficient
c_τ	$\mathbb{R}_{>3}$	propeller drag coefficient
L	$\mathbb{R}_{>3}$	distance from $\mathcal{F}_{P_i}$ to $\mathcal{F}_B$
g	$\mathbb{R}$	gravitational acceleration of Earth

Table 2.1: Overview of symbols and definitions

3 PROXIMITY EFFECT

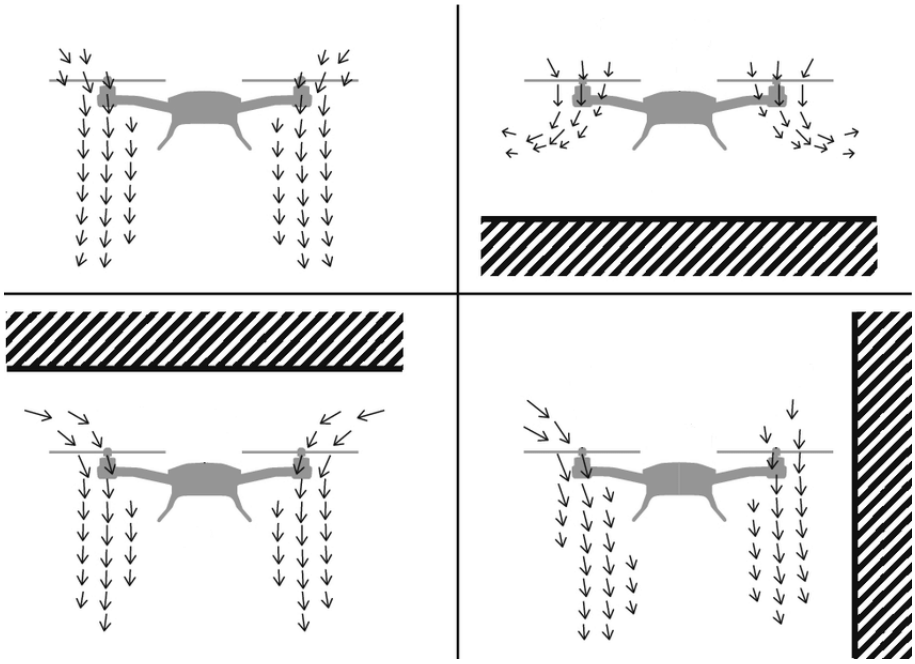


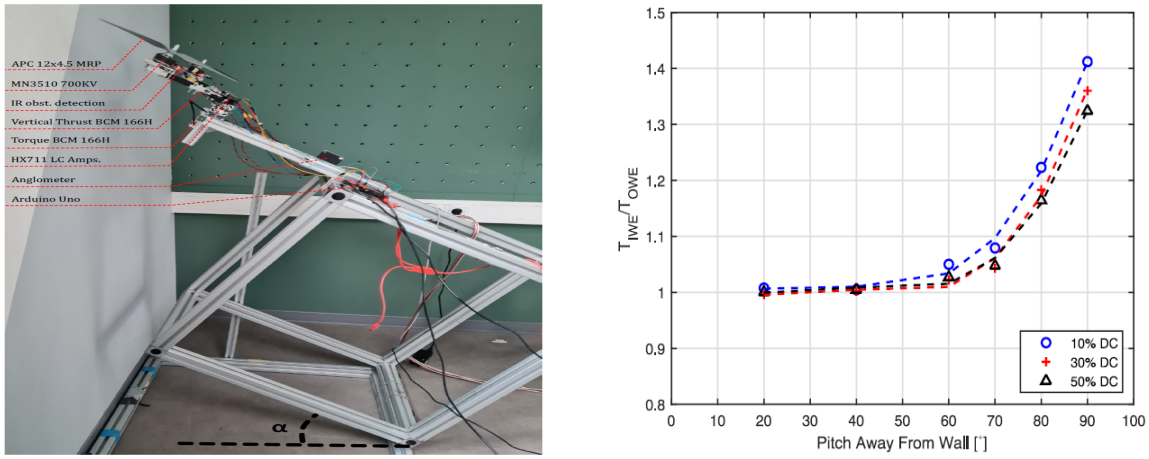
Figure 3.1: An overview on the different types of proximity effect [1]: ground, ceiling and wall effect.

The proximity effect is classified into three types: ground effect (GE), ceiling effect (CE) and wall effect (WE) [1]. The different types of proximity effect are shown in Figure 3.1. In general, surfaces will constraint the airflow before or after passing through the propeller. This will influence the free development of the propeller wake and change the thrust characteristic. In the instance of ground effect, in which the rotorcraft is flying close to the ground, the downwash of the propeller is forced to deflect outwards radially, which in turn increases the pressure beneath the propeller. Thus, this leads to an increase of propeller thrust for the same amount of power [10]. A similar effect has been observed when a horizontal surface is approach from underneath, referred to as ceiling effect. Hereby, the surface will constrain the air intake and create a low pressure zone above the propeller. Consequently, the pressure difference to the surrounding air creates a suction effect and pulls the propeller towards the ceiling, which in turn increases the thrust as well [11]. When it comes to vertical surface or wall, the effect has been considered negligible. Although, the wall causes an asymmetrical propeller wake and introduces some disturbance, the thrust remains unaffected [17]. However, the majority of the studies have analysed the proximity effect for standard coplanar multirotor configuration. For that reason, most analyses are based on a parallel alignment of propeller plane and the horizontal surface, in case of GE and CE, or to be perpendicular to

the vertical surface for the WE. Overall, it indicates that flying in close distance to a surface, such as ground or ceiling, would have a beneficial effect on the thrust generation. Therefore, exploiting the proximity effect by flying close to the surface would increase the energy efficient of the operation [11] [13] [14].

3.1 Proximity effect on tilted propeller configuration

As mention previously, most model of the proximity effect are based on traditional coplanar multirotor design. As a result, the models are based on the propeller radius and the distance to the surface in order to determine the thrust increase. However, they are not applicable on a multirotor tilted propeller configuration. For that reason, an experiment has been design in order to investigate the aerodynamic effect on tilted propeller prior to this thesis, see Figure 3.2a. Hereby, a propeller unit is tilted against a wall while measuring the generated thrust force. Multiple runs are performed at different tilt angle and spinning velocity, while the distance from the propeller tip to the wall remains constant. The collected data have been processed and are shown in Figure 3.2b.



(a) Experiment setup

(b) Experiment data

Figure 3.2: Proximity effect on tilted propeller

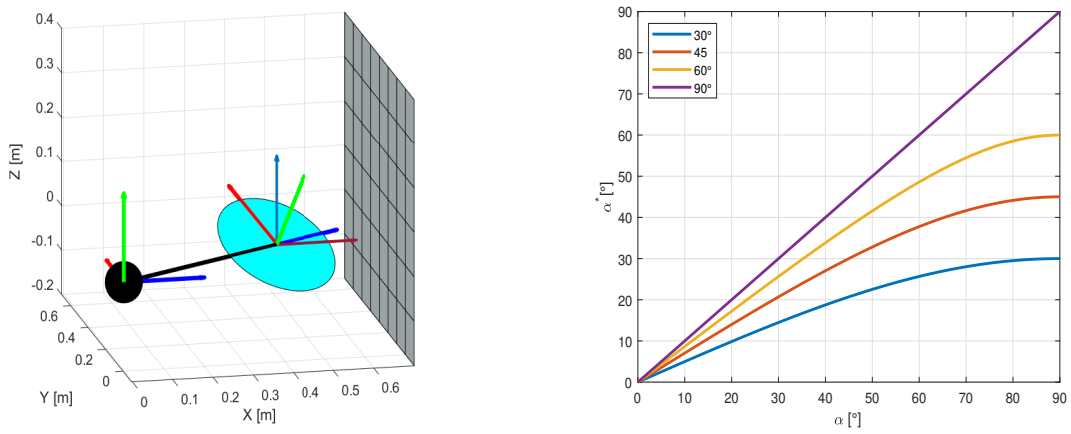
Based on the collected data the following model is formulated:

$$\lambda(\alpha) = \frac{T_{IPE}}{T_{OPE}}(\alpha) = 1 + \frac{r}{K_1 \cdot d} \cdot \sin(\alpha)^{2K_1 - 1} \tag{3.1}$$

where T_{OPE} is the thrust generated by the propeller out of the proximity effect, T_{IPE} is the thrust in proximity effect, α is the inclination angle of the propeller with respect to the surface, d the distance between surface and the closest point of the propeller in centimetre, r the radius of propeller in centimetre, K_2 rounding up the radius to the next integer and K_1 is an experimental factor based on the ground or ceiling effect depending on the tilt direction.

3.2 Proximity effect conjecture

Based on the experiment and the resulting proximity model Eq.3.1, it is established that the effect only has a significant effect when $\alpha > 60°$ in the vicinity of a vertical surface



(a) 3D visualization (b) Angle approximation

Figure 3.3: Angle approximation for different position

such as a wall. In addition, this model only considers a configuration in which the tilt axis of the propeller is parallel to the respective surface, which is not always the case considering traditional MRAV with propeller units spread evenly around the body. To be able to utilize the model, an approximation of the tilt angle α is done by considering the angle between thrust direction of the propeller regarded as $\mathbf{Z}_{P_i}$ and the normal vector to the surface ${}^{P_i}\mathbf{n}_{surf}$ in frame of the propeller, see Figure 3.3a. The angle between these two vectors is defined as,

$$\phi = \arccos\left(\frac{\mathbf{Z}_{P_i} \cdot {}^{P_i}\mathbf{n}_{surf}}{|\mathbf{Z}_{P_i}| \cdot |{}^{P_i}\mathbf{n}_{surf}|}\right) \tag{3.2}$$

which leads to the approximation

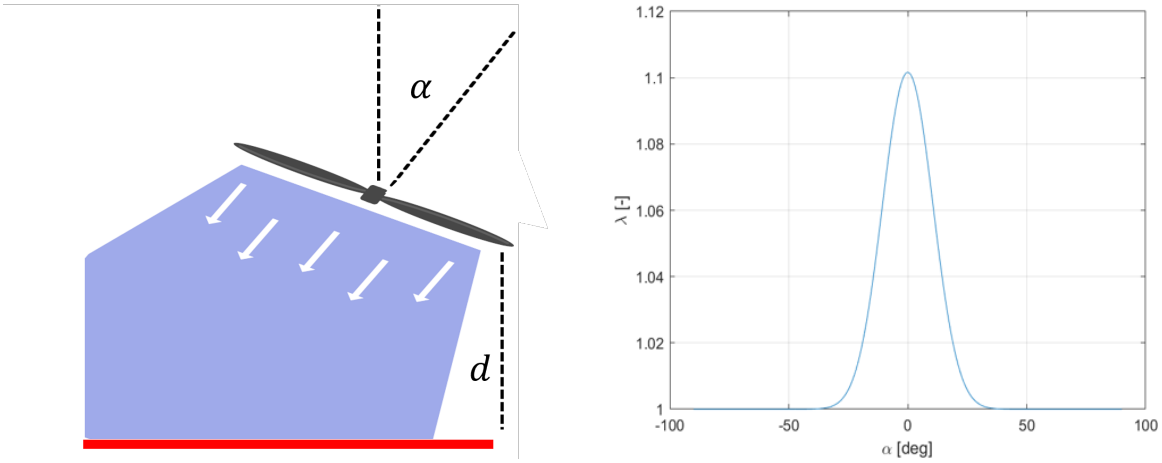
$$\alpha^* = \frac{\pi}{2} - \phi. \tag{3.3}$$

Figure 3.3b shows the approximated angle plotted against propeller tilt angle for different propeller unit around the body. It appears that none of them are able to reach the required $\alpha^* > 60°$ except the unit located at $90°$ from the body, which is equivalent to experimental model.

To circumvent this issue and enable this model to be used for this thesis, it is rotated by 90°, by modifying Eq.3.1 to

$$\lambda(\alpha) = \frac{T_{IPE}}{T_{OPE}}(\alpha) = 1 + \frac{r}{K_1 \cdot d_z} \cdot \cos(\alpha)^{2K_1 - 1} \tag{3.4}$$

which change the surface to represent ground effect as shown in Figure 3.4a. As long the MRAV flies or hover close to a horizontal orientation, the model allows including the effect into the simulation. Figure 3.4b shows the thrust ration λ w.r.t the tilt angle α while the distance d_z remains constant.



(a) Modified model for ground effect (b) Thrust ratio of modified proximity model

Figure 3.4: New model based on horizontal surface

4 VARIABLE TILT MULTIROTOR

Finding a balance between performance and efficiency is a major challenge in the development of MRAV. Especially if it comes to full actuation in six degree of freedom with the goal to independently control the position and orientation for pose-omnidirectionality. In this case, the aim for high efficiency for extended flight periods is countered by the desire for great performance in terms of force and moment generation in all directions. This can either be achieved with high internal forces due to counteracting thrust forces or additional weight due to more components to increase the degree of actuations. One solution to this challenge is an aerial platform with the ability to individually rotate the propeller unit. Based on the traditional layout of a MRAV in which the propeller are evenly spaced around the body, each arm is equipped with an additional actuator to enable a tilt operation of the entire propeller unit. This enables the aerial platform to direct the force more efficiently towards the desired direction and revert to under actuated configuration for efficient hovering.

4.1 Control structure

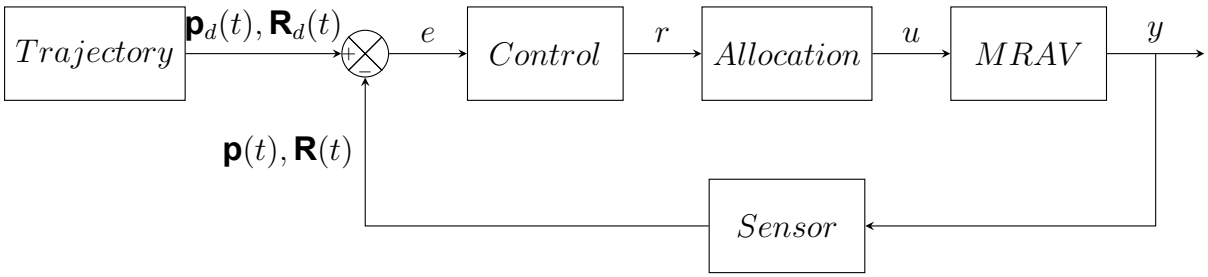


Figure 4.1: Standard control structure

The control problem is considered to be tracking a desired trajectory and orientation, defined as $(\mathbf{p}_d(t), \mathbf{R}_d(t)) \in \mathbb{R}^3 \times SO(3)$ in the world frame. The traditional way of controlling a MRAV is based on the rigid body model describe by Eq.2.3 in conjunction with the aerodynamic force model Eq.2.6 and Eq.2.7 to calculate actuator commands based on thrust vectoring. The desired force and torque are calculated based on the position and attitude error of the feedback loop. The actual actuators command are then computed via a control allocation to find the ideal change in angular velocity and direction of each propeller for generating the desired force and moments.

Control

Previous work have shown different approaches in controlling a variable tilt MRAV. A common approach is to have a separate controller for the position and attitude. Due to the platform pose-omnidirectionality, the position and attitude dynamics are decoupled because of the independency of the actuation forces and moments. Kamel et al. 2018 [5] and Bodie et al. 2020 [6] applied two separate PID controllers for position and attitude control to compute the reference linear and angular acceleration of the system. MRAV controller typically work on the acceleration level but to have access to the additional actuators of the tilt mechanism Ryll et al. 2015 [4] propose to apply dynamic output linearization to the system. By differentiating the system and move the input-output map to a higher differential level, allows to directly involve the propeller tilt angle dynamics. Therefore, the controller outputs linear and angular jerk reference commands. A similar approach is developed by Allenspach et al. 2020 [8] based on LQRI optimal control.

Control allocation

The control allocation is concerned with the determination of the ideal spinning velocuty of the propeller $\boldsymbol{\Omega}$ and angular position of the propeller unit $\boldsymbol{\alpha}$ to ensure the given virtual control reference. Different approaches of solving the control allocation problem has been studied in various fields and are composed in a general survey [18]. The general approach to determine the allocation for MRAV with fix propeller is based on the inversion of the euquation of motion Eq.2.14 as follows:

$$\boldsymbol{\Omega} = \mathbf{J}^{-1} \left(\begin{bmatrix} {}^{B}\mathbf{f}_r \\ {}^{B}\boldsymbol{\tau}_r \end{bmatrix} - \mathbf{f} \right) \quad \mathbf{J} \in \mathbb{R}^{6\times 6} \tag{4.1}$$

A fully actuated platform will have an invertible squared allocation matrix $\mathbf{J} \in \mathbb{R}^{6\times 6}$. But more often the platform is either under- or over-actuated, which result in a non-squared allocation matrix $\mathbf{J} \in \mathbb{R}^{6\times n}$. In this case the Moore-Penrose pseudo inverse is applied

$$\boldsymbol{\Omega} = \mathbf{J}^{\dagger} \left(\begin{bmatrix} {}^{B}\mathbf{f}_r \\ {}^{B}\boldsymbol{\tau}_r \end{bmatrix} - \mathbf{f} \right) \quad \mathbf{J} \in \mathbb{R}^{6\times n} \tag{4.2}$$

to determine the actuation output. A problem arises when the system is a variable tilt MRAV because of it over-actueted nature and the allocation matrix is no longer static, but instead it is depended on the angular position of the propeller which is nonlinear, and it is not possible to solve for angular position $\boldsymbol{\alpha}$ directly. The problem could be resolved by implementing a nonlinear least-squares optimization for the allocation problem or circumvent the allocation by applying a nonlinear model predictive control (NMPC). Both solution would calculate an optimal solution for the actuation of the propeller rotation speed and angular tilt position. Unfortunately, both approaches requires high computational performance to output actuation commands on a high rate to ensure agile and stable performance of the platform. However, MRAV are mostly equipped with small and efficient computer systems which does not have sufficient CPU performance to calculate the solution in an appropriate time step and therefore renders these techniques unsuitable. A new approach is to transform the nonlinear allocation into a linear one by force decomposition [5] [6]. The allocation matrix $\mathbf{J}$ is by closer look a linear combination of $\sin(\alpha_i)\Omega_i$ and $\cos(\alpha_i)\Omega_i$. This allows to simplify the matrix by extracting the

components and create a new static allocation which can be inverted based on Moore-Penrose pseudoinverse. The result of this allocation approach will give the vertical and lateral force components for each propeller unit. Later one, the propeller spinning velocity and angular tilt position can be obtained based on reconstruction of the thrust force. Ryll et al. 2015 [4] follows through with the output linearization approach by differentiating the allocation matrix $\mathbf{J}_\alpha$ with respect to time, which allows accessing directly the dynamics of the tilt angle. In addition, this approach offers a way to access the nullspace for further optimization of the allocation. Allespach et al. 2020 [8] develop an approach based on the linearization method by Ryll et al. 2015 [4], but utilized a decomposed allocation matrix bases on [5] and [6]. This combination creates an efficient control of the platform during hovering and allows a cable unwinding operation.

4.2 Control design

For the purpose of this thesis, several aspects needs to be considered for the choice of a control design. First, the design should allow the incorporation of the proximity effect without extensive modification of the overall principal. And second, there should be an option to involve an optimization to take advantage of increase of thrust near surfaces. Based on the general control structure, see Figure 4.1, there are three modules to consider: trajectory, controller and allocation. Based on the current literature on variable tilt multirotor control [4][5][6][8], the focus was set on the controller and allocation. Involving the trajectory to find the optimal path in order to exploit the proximity effect has been demonstrated for standard quadrotor configuration [13]. But this requires having knowledge or map of the environment. On the other hand, modifying the controller would be best if it would directly output the required actuation. However, the system is over-actuated and nonlinear which requires an optimization based controller, for example quadratic programming or nonlinear model predictive control. Given that, the best option is to adapt the control allocation. In case of an over-actuated system, it is based on Moore-Penrose pseudo inverse, which in its core solves for the minimum-norm least squares solution. Comparing all control design, it can be seen that [8] is a combination of [4] and [5] with some additional steps. To keep things simple, it is the best to revert to the fundamental designs. Therefore, it is decided to move forward with the approach based on dynamic output linearization [4], because it offers a way to include the proximity effect and an optimization via the null-space in order to take advantage of the proximity effect for an efficient flight operation. Henceforth, this section gives a brief overview on the chosen control design.

4.2.1 Simplified model

In the previous chapter, a full dynamic model of the system Eq.2.11 is introduced. It completely describes the motion of the platform in space. However, for the control design a simplified model is considered which neglects the gyroscopic effect by considering it a second order disturbance to be rejected by the controller. The equation of motion is then rewritten to

$$\begin{bmatrix} \ddot{\mathbf{p}} \\ {}^B\dot{\boldsymbol{\omega}} \end{bmatrix} = \begin{bmatrix} -g\mathbf{e}_3 \\ \mathbf{0} \end{bmatrix} + \begin{bmatrix} \frac{1}{m}\mathbf{R}_B^W & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbb{J}^{-1} \end{bmatrix} \begin{bmatrix} \mathbf{F}(\boldsymbol{\alpha}) \\ \boldsymbol{\tau}(\boldsymbol{\alpha}) \end{bmatrix} \boldsymbol{\Omega} \tag{4.3}$$

4.2.2 Control

The simplified model (Eq. 4.3) is first extended with the tilt velocity $\omega_\alpha = \dot{\alpha}$ and is rewritten to

$$\begin{aligned}
\begin{bmatrix} \ddot{\mathbf{p}} \\ {}^B\dot{\boldsymbol{\omega}} \end{bmatrix} &= \begin{bmatrix} -g\mathbf{e}_3 \\ \mathbf{0}_{3\times 1} \end{bmatrix} + \begin{bmatrix} \frac{1}{m}\mathbf{R}_B^W & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbb{J}^{-1} \end{bmatrix} \begin{bmatrix} \mathbf{F}(\boldsymbol{\alpha}) & \mathbf{0}_{3\times n} \\ \boldsymbol{\tau}(\boldsymbol{\alpha}) & \mathbf{0}_{3\times n} \end{bmatrix} \begin{bmatrix} \boldsymbol{\Omega} \\ \boldsymbol{\omega}_\alpha \end{bmatrix} \\
&= \mathbf{f} + \mathbf{J}_R \begin{bmatrix} \mathbf{J}_\alpha(\boldsymbol{\alpha}) & \mathbf{0} \end{bmatrix} \begin{bmatrix} \boldsymbol{\Omega} \\ \boldsymbol{\omega}_\alpha \end{bmatrix} \\
&= \mathbf{f} + \mathbf{J}_R \mathbf{J}_\alpha(\boldsymbol{\alpha}) \begin{bmatrix} \boldsymbol{\Omega} \\ \boldsymbol{\omega}_\alpha \end{bmatrix} \\
&= \mathbf{f} + \mathbf{J}(\boldsymbol{\alpha}) \begin{bmatrix} \boldsymbol{\Omega} \\ \boldsymbol{\omega}_\alpha \end{bmatrix}
\end{aligned} \tag{4.4}$$

where $\mathbf{f} \in \mathbb{R}^6$ is a constant drift vector, $\mathbf{J}_\alpha(\boldsymbol{\alpha}) \in \mathbb{R}^{6\times n}$, $\mathbf{J}_R \in \mathbb{R}^{6\times 6}$ and $\mathbf{J}(\alpha) \in \mathbb{R}^{6\times 2n}$ is the output Jacobian. Naturally, one would apply a static feedback linearization of Eq.4.4 which then follows

$$\begin{bmatrix} \boldsymbol{\Omega} \\ \boldsymbol{\omega}_\alpha \end{bmatrix} = \mathbf{J}(\boldsymbol{\alpha})^\dagger \left(-\mathbf{f} + \begin{bmatrix} \ddot{\mathbf{p}}_r \\ \dot{\boldsymbol{\omega}}_r \end{bmatrix} \right) \tag{4.5}$$

where $\begin{bmatrix} \ddot{\mathbf{p}}_r & \dot{\boldsymbol{\omega}}_r \end{bmatrix}^\mathsf{T} \in \mathbb{R}^6$ is a linear and angular acceleration reference vector to be followed by the system Eq. 4.3. Unfortunately, inverting the matrix $\mathbf{J}(\boldsymbol{\alpha})$ will not give a feasible solution at this stage due to possible rank deficiency. Because of the added null matrix $\mathbf{0} \in \mathbb{R}^{6\times n}$ in $\mathbf{J}_\alpha$ in order to include and weight the inputs $\boldsymbol{\omega}_\alpha$ the matrix rank can only be $rank(\mathbf{J}) = rank(\mathbf{J}_\alpha) \leq 6$. Therefore, inverting the matrix will only give a result related to spinning velocity Ω and depending on the given angular position $\boldsymbol{\alpha}$ at that point it could lead to loss of controllability for the system. Consequently, the linear and angular acceleration of the system are influenced only by the spinning velocity $\boldsymbol{\Omega}$ and angular position $\boldsymbol{\alpha}$ based on $\mathbf{J}_\alpha(\boldsymbol{\alpha})$ and tilt velocity $\boldsymbol{\omega}_\alpha$ does not have any effect. This can bypass by applying dynamic output linearization to move towards a higher differential level to which $\boldsymbol{\omega}_\alpha$ can be utilized. For that reason, the term $\mathbf{J}_\alpha(\boldsymbol{\alpha})\boldsymbol{\Omega}$ is expanded to

$$\mathbf{J}_\alpha(\alpha)\boldsymbol{\Omega} = \sum_{i=1}^{n} \mathbf{j}_i(\boldsymbol{\alpha})\Omega_i \tag{4.6}$$

and note that differentiation with respect to time yields

$$\frac{d}{dt}\mathbf{J}_\alpha(\boldsymbol{\alpha})\boldsymbol{\Omega} = \mathbf{J}_\alpha(\boldsymbol{\alpha})\dot{\boldsymbol{\Omega}} + \sum_{i=1}^{n} \frac{\partial}{\partial \boldsymbol{\alpha}}\mathbf{j}_i(\boldsymbol{\alpha})\boldsymbol{\omega}_\alpha\Omega_i \tag{4.7}$$

Next the equation 4.4 is differentiated with respect to time as follows:

$$\begin{aligned}\begin{bmatrix}\dddot{\mathbf{p}}\\ {}^{B}\dddot{\boldsymbol{\omega}}\end{bmatrix} &= \mathbf{J}_R\mathbf{J}_\alpha(\boldsymbol{\alpha})\dot{\boldsymbol{\Omega}} + \mathbf{J}_R\sum_{i=1}^{n}\frac{\partial}{\partial\boldsymbol{\alpha}}\mathbf{j}_i(\boldsymbol{\alpha})\boldsymbol{\omega}_\alpha\Omega_i + \dot{\mathbf{J}}_R\mathbf{J}_\alpha(\boldsymbol{\alpha})\boldsymbol{\Omega}\\ &= \mathbf{J}_R\begin{bmatrix}\mathbf{J}_\alpha(\boldsymbol{\alpha}) & \sum_{i=1}^{n}\frac{\partial}{\partial\boldsymbol{\alpha}}\mathbf{j}_i(\boldsymbol{\alpha})\Omega_i\end{bmatrix}\begin{bmatrix}\dot{\boldsymbol{\Omega}}\\ \boldsymbol{\omega}_\alpha\end{bmatrix} + \begin{bmatrix}\frac{\dot{\mathbf{R}}_B^W}{m}\mathbf{F}(\boldsymbol{\alpha})\boldsymbol{\Omega}\\ \mathbf{0}\end{bmatrix}\\ &= \mathbf{J}_R\mathbf{J}'_\alpha(\boldsymbol{\alpha},\boldsymbol{\Omega})\begin{bmatrix}\dot{\boldsymbol{\Omega}}\\ \boldsymbol{\omega}_\alpha\end{bmatrix} + \mathbf{b}(\boldsymbol{\alpha},\boldsymbol{\Omega},{}^{B}\boldsymbol{\omega})\\ &= \mathbf{A}(\boldsymbol{\alpha},\boldsymbol{\Omega})\begin{bmatrix}\dot{\boldsymbol{\Omega}}\\ \boldsymbol{\omega}_\alpha\end{bmatrix} + \mathbf{b}(\boldsymbol{\alpha},\boldsymbol{\Omega},{}^{B}\boldsymbol{\omega})\end{aligned} \tag{4.8}$$

where $\dot{\boldsymbol{\Omega}}$ is the change of the propeller spinning velocity. The actual input Ω can then be obtained by integration over time in the simulation. The new allocation matrix $\mathbf{A}(\boldsymbol{\alpha},\boldsymbol{\Omega}) \in \mathbb{R}^{6\times 2n}$ is composed by two blocks, as shown in Eq. 4.8. The first block resembles parts of the former output Jacobian $\mathbf{J}(\boldsymbol{\alpha})$ which consist of $\mathbf{J}_R\mathbf{J}_\alpha$ and a second block of n columns, which is not a null-matrix, to scale the inputs $\boldsymbol{\omega}_\alpha$. In addition, the rank of the matrix $\mathbf{A}(\boldsymbol{\alpha},\boldsymbol{\Omega})$ is $rank(\mathbf{A}) = 6$ if $\Omega_i \neq 0$ for $i = 1\ldots n$. Under those circumstances, full rank of the matrix $\mathbf{A}$ is ensured as long as the propellers are spinning. Provided that $rank(\mathbf{A}) = 6$ the system Eq. 4.8 can be inverted and written as

$$\begin{bmatrix}\dot{\boldsymbol{\Omega}}\\ \boldsymbol{\omega}_\alpha\end{bmatrix} = \mathbf{A}^\dagger\left(\begin{bmatrix}\dddot{\mathbf{p}}_r\\ \dddot{\boldsymbol{\omega}}_r\end{bmatrix} - \mathbf{b}\right) + \left(\mathbf{I}_{2n} - \mathbf{A}^\dagger\mathbf{A}\right)\mathbf{z} \tag{4.9}$$

where

$$\begin{bmatrix}\dddot{\mathbf{p}}\\ {}^{B}\dddot{\boldsymbol{\omega}}\end{bmatrix} = \begin{bmatrix}\dddot{\mathbf{p}}_r\\ \dddot{\boldsymbol{\omega}}_r\end{bmatrix} \tag{4.10}$$

in order to complete input-output linearization. The second term in Eq.4.9 is an extension to utilize the null space of the matrix $\mathbf{A}$. Because of having $2n$ control inputs and only 6 DoF to be controlled, there is an actuation redundancy which can be exploited for optimization purposes. For that case, the vector $\mathbf{z} \in \mathbb{R}^{2n}$ is introduced to be projected onto the null-space of $\mathbf{A}$ and to take advantage of the actuation redundancy.
The control problem is considered to be tracking a desired trajectory and orientation, defined as $(\mathbf{p}_d(t), \mathbf{R}_d(t)) \in \mathbb{R}^3 \times SO(3)$ in the world frame. Suppose the position trajectory is $\mathbf{p}_d(t) \in \mathcal{C}^3$, then the linear jerk reference $\dddot{\mathbf{p}}_r$ can be set as follows:

$$\dddot{\mathbf{p}}_r = \dddot{\mathbf{p}}_d + \mathbf{K}_{p_1}(\ddot{\mathbf{p}}_d - \ddot{\mathbf{p}}) + \mathbf{K}_{p_2}(\dot{\mathbf{p}}_d - \dot{\mathbf{p}}) + \mathbf{K}_{p_3}(\mathbf{p}_d - \mathbf{p}) \tag{4.11}$$

where $\mathbf{K}_{p_1}$,$\mathbf{K}_{p_2}$, $\mathbf{K}_{p_3}$ are positive definite gain matrices which satisfies Hurwitz. This yields an exponential convergence of the position error to $\mathbf{0}$. In order to circumvent the inherent singularity problem with Euler angles, the orientation error is directly defined in $SO(3)$. Suppose the orientation is $\mathbf{R}_d(t) \in \mathcal{C}^3$ and let $\boldsymbol{\omega}_d = [\mathbf{R}_d^\intercal\dot{\mathbf{R}}_d]_\vee$, where $[\bullet]_\vee$ is the inverse map from $so(3)$ to $\mathbb{R}^3$. The orientation error is defined as,

$$\mathbf{e}_R = \frac{1}{2}\left[(\mathbf{R}_B^W)^T\mathbf{R}_d - (\mathbf{R}_d)^T\mathbf{R}_B^W\right]_\vee \tag{4.12}$$

Then the angular jerk reference $\dddot{\boldsymbol{\omega}}_r$ can be set as follows:

$$\dddot{\boldsymbol{\omega}}_r = \dddot{\boldsymbol{\omega}}_d + \mathbf{K}_{\omega_1}(\ddot{\boldsymbol{\omega}}_d - \ddot{\boldsymbol{\omega}}) + \mathbf{K}_{\omega_2}(\dot{\boldsymbol{\omega}}_d - \dot{\boldsymbol{\omega}}) + \mathbf{K}_{\omega_3}\mathbf{e}_R \tag{4.13}$$

where $\mathbf{K}_{\omega_1}$,$\mathbf{K}_{\omega_2}$, $\mathbf{K}_{\omega_3}$ are positive definite gain matrices which satisfies Hurwitz. This yields an exponential convergence of the orientation error to $\mathbf{0}$.

4.2.3 Optimization

The optimization aims to exploit the actuation redundancy and the null-space of the system by utilizing the vector **z** introduced in Eq.4.9. When the vector **z** is projected onto the null-space of matrix **A** it will generate commands which does not interfere with the primary tracking objective. This can be leveraged for additional objective or to maintain certain boundaries. In this case, the dynamic output linearizing in the previous section imposes the requirement that $rank(\mathbf{A}) = 6$ must be maintained during operation. This can be achieved by ensuring that $\boldsymbol{\Omega} \neq \mathbf{0}$. One additional objective is the reduction of energy consumption during flight operation. Therefore, it is beneficial to minimize $\boldsymbol{\Omega}$ as it constantly performing dissipative work while generating thrust. Based on these objectives, a cost function $H(\boldsymbol{\Omega})$ is designed as follows:

$$H(\boldsymbol{\Omega}) = \sum_{i=1}^{n} h(\Omega_i) \tag{4.14}$$

with

$$h(\Omega_i) = \begin{cases} k_{h_1} \tan^2(\gamma_1 \Omega_i + \gamma_2) & \Omega_{min} < \Omega_i \leq \Omega_{rest} \\ k_{h_2}(\Omega_i - \Omega_{rest})^2 & \Omega_i > \Omega_{rest} \end{cases} \tag{4.15}$$

$$\gamma_1 = \frac{\pi}{2(\Omega_{rest} - \Omega_{min})} \tag{4.16}$$

$$\gamma_2 = -\gamma_1 \Omega_{rest} \tag{4.17}$$

where $k_{h_1} > 0$ and $k_{h_2} > 0$ are suitable scalar gains. In order to prevent the spinning velocity to go to zero, a suitable minimum speed $\Omega_{min} > 0$ is defined. In addtion, a resting speed $\Omega_{rest} > \Omega_{min}$ is defined to be

$$\Omega_{rest} = \frac{mg}{nc_t} \tag{4.18}$$

which is the minimum spinning velocity required for the platform to hover or rest in the air. Given all the parameters, the cost function will go to infinity $h_i(\Omega_i) \to \infty$ when $\Omega_i \to \Omega_{min}$ or $\Omega_i \to \infty$ and it has a unique minimum at Ω_{rest}. An example of the cost function is shown in Figure 4.2. The cost function $H(\boldsymbol{\Omega})$ is the minimized by applying gradient descent and substitute it in Eq.4.9 as follows:

$$z = -k_H \begin{bmatrix} \nabla_{\boldsymbol{\Omega}} H(\boldsymbol{\Omega}) \\ \mathbf{0} \end{bmatrix} \tag{4.19}$$

where k_H > 0 is suitable step size.

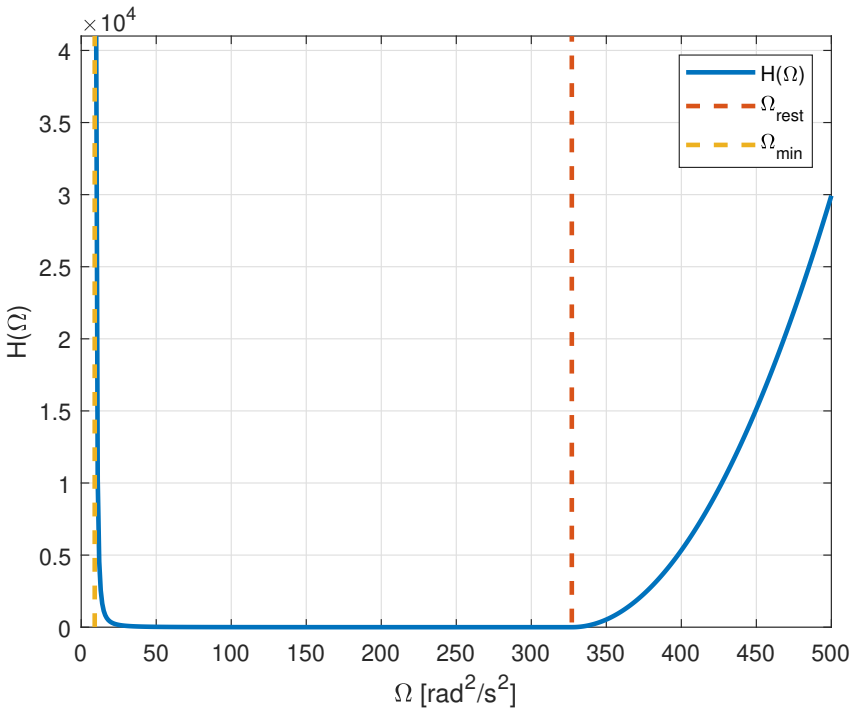


Figure 4.2: Example of a cost function

5 THE EFFECT OF SURFACE PROXIMITY ON VARIABLE TILT MULTIROTOR CONTROL

This section analyses the effect of flying in proximity to surfaces on variable tilt multirotor control.

5.1 Simulation framework in MATLAB/Simulink

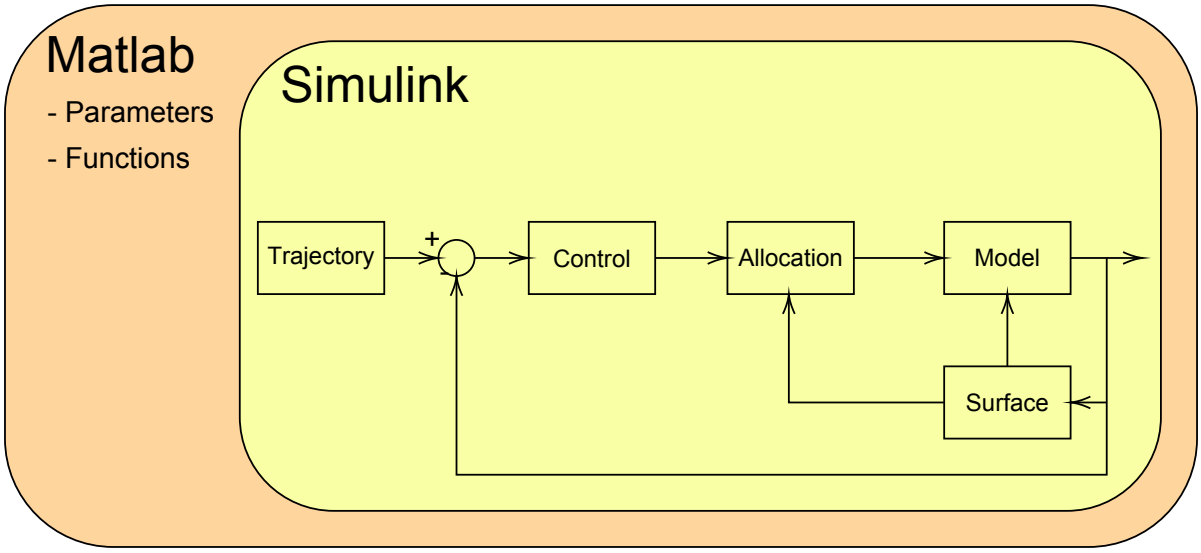


Figure 5.1: Matlab/Simulink framework

A simulation framework is build in the MATLAB/Simulink environment as shown in Figure 5.1. It is based on a parameter script in MATLAB in which all the properties of the simulation are stored, and it generates certain function to be used in Simulink upon execution. The MRAV model consist of the equation of motion described in chapter 2 and the control and allocation are based on the control approach introduced in chapter 4. The controller requires a desired linear and angular jerk as an input, which is provided by a trajectory generator.

5.2 Simulation implementation

Different scenarios have been simulated in order to analyse the proximity effect on variable tilt MRAV control:

1. The MRAV starts in point A and flies over a surface towards point B which is located in the middle and above of the surface (see Figure 5.2). The surface is represented with a light blue cuboid and can be associated with a rooftop or table which the MRAV has to fly over it to execute certain operation. The top of the cuboid is located at $z = 0$. During the movement, the MRAV will keep the y-coordinate constant at $y = 0$ and move on the x-coordinate from $x = 0$ to $x = 4$. The object starts at $x = 2$. The MRAV will maintain a height of $\frac{z}{R} = 1$.

2. The MRAV flies over the entire surface from point A to point B (see Figure 5.3). The platform will keep the y-coordinate constant at $y = 0$ and move on the x-coordinate from $x = 0$ to $x = 8$. The surface starts at $x = 2$ and ends at $x = 6$. The MRAV will maintain a height of $\frac{z}{R} = 1$.

3. The MRAV flies from point A to point B (see Figure 5.4). At point B only the front two propellers are over the surface for partial proximity effect.The platform will keep the y-coordinate constant at $y = 0$ and move on the x-coordinate from $x = 0$ to $x = 1.6952$. The surface starts at $x = 2$ and ends at $x = 6$. The MRAV will maintain a height of $\frac{z}{R} = 1$.

All scenarios are analysed with three different control setup:

- Reference control for free flight.
- Reference control with proximity effect in cooperated into the aerodynamic force model.
- Informed control with extended optimization.

Each control setup utilized the same parameter listed in Table 5.1 and are given the same trajectory. A comparison between these option is made by analysing their behaviour and their exploitation of the proximity effect.

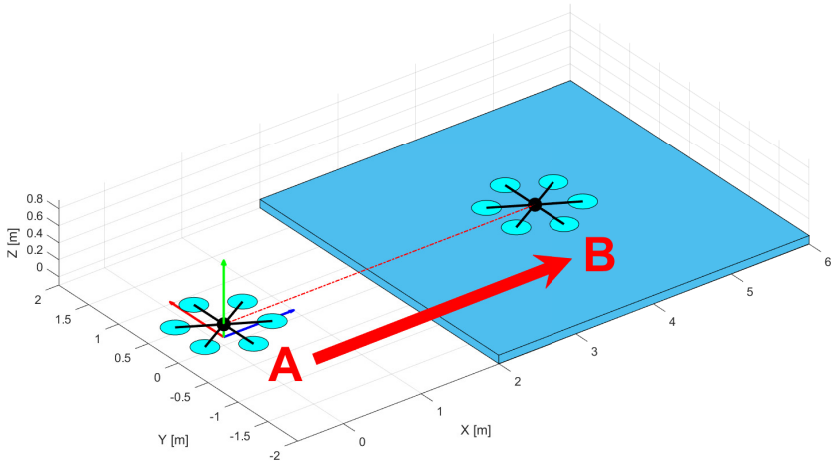


Figure 5.2: Flight scenario 1

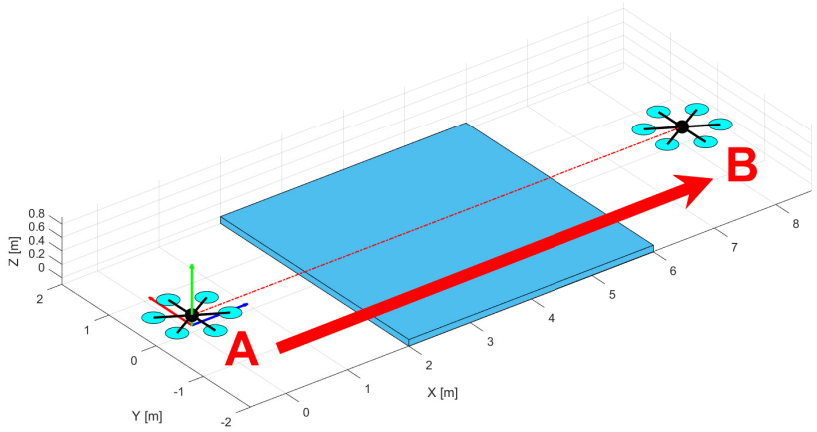


Figure 5.3: Flight scenario 2

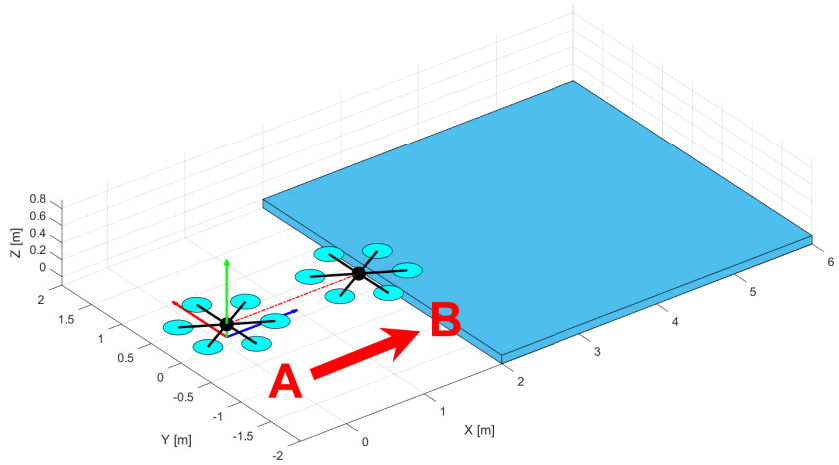


Figure 5.4: Flight scenario 3

5.2.1 Aerodynamic thrust model modification

As illustrated in the previous chapter, when a variable tilt multirotor is flying near to the ground or other horizontal surfaces, the aerodynamic thrust model Eq.2.6 is no longer valid. Consequently, the thrust model needs modification in the simulation to account for the proximity effect as follows,

$$f_{P_i}(\alpha_i, d_i) = c_f \cdot \Omega_{P_i} \cdot \lambda(\alpha_i, d_i) \tag{5.1}$$

where $\lambda(\alpha_i, d_i)$ is the proximity effect factor that accounts for the increase in thrust due to the nearby surfaces. The factor $\lambda(\alpha_i, d_i)$ has a dependency on the relative angle

Symbols	Value	Unit	Definitions
n	6	$-$	number of propellers
m	2	kg	total mass
$\mathbb{J}$	$diag(0.04; 0.04; 0.04)$	$kg \cdot m^2$	inertia of the multirotor body B
L	0.5	m	Arm length
D	12	$inch$	Propeller diameter
c_f	10^{-3}	$-$	Thrust coefficient
c_τ	10^{-6}	$-$	Torque coefficient
g	9.81	m/s^2	Gravitational acceleration of Earth
K_1	10	$-$	Ground effect factor
K_2	15	$-$	Ground effect factor
k_{h_1}	1	$-$	Cost function factor
k_{h_2}	1	$-$	Cost function factor
ω_{min}	3	rad/s	Cost function factor
k_H	3	$-$	Gradient descent step size
K_{p_1}	$diag(10; 10; 10)$	$-$	Position control gain
K_{p_2}	$diag(70; 70; 70)$	$-$	Velocity control gain
K_{p_3}	$diag(100; 100; 100)$	$-$	Acceleration control gain
K_{ω_1}	$diag(10; 10; 10)$	$-$	Orientation control gain
K_{ω_2}	$diag(70; 70; 70)$	$-$	Angular velocity control gain
K_{ω_3}	$diag(100; 100; 100)$	$-$	Angular acceleration control gain

Table 5.1: Overview of all parameter

between the surface and the propeller plane and a relative distance between the surface and the closest point of the propeller. By considering a horizontal surface as shown in the different scenarios, the relative angle between propeller and surface is equal to the tilt angle α used to define the angular position of the propeller unit. The relative distance is simplified by setting the top of the surface on $z = 0$ plane. With this change, the distance can be directly calculated in the world frame. Furthermore, in all scenarios, the MRAV is flying over a surface in which it will transition from no-effect to proximity effect. This transition can be modelled based on the propeller area. The effect increases linear with the propeller area coverage by the surface [10]. Therefore, Eq.3.4 is expanded to

$$\lambda(\alpha, d_x, d_y, d_z) = \frac{T_{IPE}}{T_{OPE}}(\alpha) = 1 + \frac{r}{K_1 \cdot d_z} \cdot \cos(\alpha)^{2K_1 - 1} \cdot \mu(d_x, d_y) \tag{5.2}$$

where d_x and d_y are the distance from the propeller centre to the edge of the surface. Therefore, a surface is defined with a centre position ${}^W O_s = [x_s, y_s, z_s]^\intercal$, length on the x-axis l and width on the y-axis w. Given the position of the ${}^W O_{P_i} = [x_{P_i}, y_{P_i}, z_{P_i}]^\intercal$ transition is described by

$$d_x = x_{P_i} - x_s - \frac{l}{2} - r \tag{5.3}$$

$$d_y = y_{P_i} - y_s - \frac{w}{2} - r \tag{5.4}$$

$$\mu(d_x, d_y) = \begin{cases} 0, & \text{for } d_x < 0,\ d_y < 0 \\ 1, & \text{for } d_y > 2r,\ d_y > 2r \\ \frac{d_x}{2r} \cdot \frac{d_y}{2r}, & \text{otherwise} \end{cases} \tag{5.5}$$

5.2.2 Optimization modification

The optimization propose is chapter 4 aims to reduce energy consumption during free flight. The majority of the energy is consumed by the actuator driving the propellers. The propellers perform dissipative work to generate thrust, which causes a counter-torque due to air drag. In order to reduce the energy consumption, a cost function is designed to minimize the necessary angular velocity of the propeller. For that reason, a minimal spinning velocity for each propeller is defined to keep the MRAV hovering or resting in the air. Hence, a resting propeller spinning velocity is defined as

$$\Omega_{rest} = \frac{mg}{nc_t} \tag{5.6}$$

where Ω_{rest} remains constant as long as the thrust in not influenced by anything. However, in the event of near surface flight, additional thrust is gained due to the proximity effect. Consequently, a lower spinning speed is achievable but is prevented by the current cost function due to higher cost in the lower region. For that reason, a variable resting spinning velocity is proposed which takes into account the additional thrust as follows

$$\Omega_{rest}(d_z) = \frac{mg}{nc_t\lambda_{max}(d_z)} \tag{5.7}$$

where $\lambda_{max}(d_z)$ is the maximal obtainable thrust increase due to the proximity effect. It is determined by the distance d_z between surface and closest point of the propeller. For that reason, the maximum value is determined by

$$\lambda(\alpha, d_z) = 1 + \frac{r}{K_1 \cdot d_z(\alpha)} \cdot \mathsf{cos}(\alpha)^{2K_1-1} \tag{5.8}$$

where $d_z(\alpha)$ can be determined given the actual origin of the propeller frame, so that:

$$d_z(\alpha) = {}^W Z_{P_i} - r\,\mathsf{sin}(\alpha) \tag{5.9}$$

In this case, the maximum value is not necessary at $\alpha = 0$ with decrease of flight height as shown in Figure 5.6. In order to take advantage of the proximity effect, the maximum value needs to be calculated as follows:

$$\lambda_{max} = \int_0^{\frac{\pi}{2}} max\left(1 + \frac{r}{K_1 \cdot ({}^W Z_{P_i} - r\,\mathsf{sin}(\alpha))} \cdot \mathsf{cos}(\alpha)^{2K_1-1}\right) d\alpha \tag{5.10}$$

The newly defined resting spinning velocity Ω_{rest} is then implemented in the cost function Eq.4.15 and is shown in Figure 5.5.

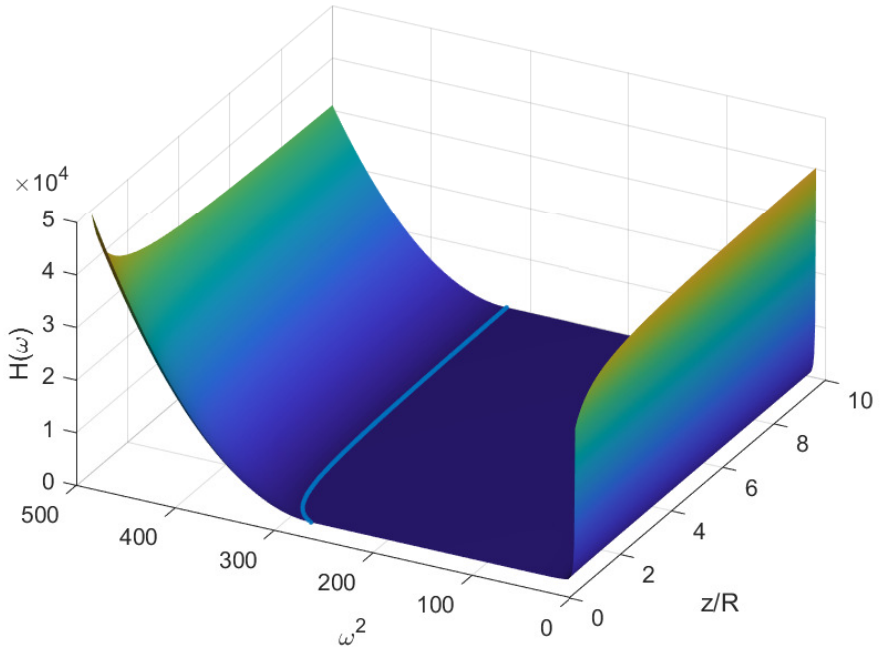


Figure 5.5: New cost function with dependency on the distance to surface

5.3 Simulation results

5.3.1 Flight scenario 1

The result of the first scenario with different control approaches are shown in Figure 5.7, 5.8 and 5.9. The figures show the position $[x, y, z]$, Euler angles orientation $[\phi, \theta, \psi]$, squared angular velocity of the propellers $[\Omega_1 \dots \Omega_6]$ and the tilt angle $[\alpha_1 \dots \alpha_6]$ with respect to time.

Referenc control

The reference control scenario, see Figure 5.7, is considered to fly freely high above any surfaces and does not take into account any effect related to surface proximity. Its response serves a baseline for the comparison with the other methods. The MRAV starts from a steady state hovering position in which the propellers are spinning at the predefined resting speed Ω_{rest} based on the cost function design of the optimization. Herby, all tilt angle are at zero and are pointing upwards working against gravity to hold the height. To move forward, the spinning velocity is increased, and the propellers are tilted to accelerate the platform toward the desired position without changing its orientation. During this phase, the gradient descent optimization wants to decrease the spinning velocity as long it does not interfere with the control objective. Therefore, after an initial peak in propeller speed, it will decrease again while the tilt angles are moving towards the zero position as a result of the reduction. In the deacceleration phase of the movement, the propeller speed reacts the same as before and the propeller orientation moves in the opposite direction. When the platform arrives at the final position, it will have a steady state with the propeller at resting speed and at zero tilt angle. Overall, the platform follows the given trajectory without any deviation.

Extended model

In contrast, the simulation with an extended model which incorporates the proximity effect, see Figure 5.8, displays a different behaviour. The position plot indicates the surface on the x-axis in light green. It can be observed that as soon as the platform is nearby, the surface the behaviour differs from the reference system. It experiences a disturbance due to the change in thrust upon reaching the surface, which results in a torque affecting the system's attitude. The platform manage to fly into the effect at a relative close distance $\frac{z}{r} = 1$ with a small error in pitch while approaching the area, but manage to stabilize upon reaching the final position. A difference is noticeable in comparison to the reference case in the deacceleration phase and hover phase at the final position. The spinning velocity drops and stays at the resting velocity after the initial acceleration. Because of the addition thrust, an increase of spinning velocity is not necessary, instead only tilting the propellers is commanded. Hence, arriving at the final position the propellers remain tilted instead converging back to zero angular position. For this reason, the system reaches its limit set by the cost function, so that lowering the spinning velocity comes at a higher cost. Therefore, a change of the cost function is desired to enable a lower spinning velocity of propellers.

Extended control

In this scenario with an extended controller, see Figure 5.8, the behaviour is similar to the other cases at the beginning, as it is considered to fly freely in mid-air. In the event of reaching the surface, the system experience a pitch movement as result of the disturbance, which is twice as high as in case of the extended model. The reason is that during this time the controller is dropping the spinning velocity, which gives limited range to counter the disturbance. Another interesting behaviour is the actuation during hovering phase at the final destination. Instead of converging to a zero tilt position for the propeller orientation, it maintains a small tilt. As described in the previous section, the highest increase in thrust will shift by a small amount away from zero angular position. In this case of flying one propeller radius above the surface, the thrust increase has its maximum at, $\alpha \approx 2^\circ$ as shown in Figure 5.6. Although this may be true, this will also increase the inner force during this hovering phase and results in a decrease of efficiency. Because of that, the controller does not settle to this particular point, instead it converges to $\alpha \approx \pm 1.5^\circ$ in order to lower spinning velocity.

5.3.2 Flight scenario 2

In this scenario, the platform is flying over the surface to the other side as shown in Figuer 5.3. It can be observed that the simulation with the reference controller, see Figure 5.10, has a similar behaviour as is in scenario 1. On the other hand, the other simulation are experiencing two disturbance. One while flying into the influence of the proximity effect and one while flying out of the effect. The simulation with an extended model, in Figure 5.11, is able to operate with a small error in pitch and roll during the transition phase. During the time flying over the surface, it can be observed that controller commands to open up the tilt angle in order to adjust for the higher thrust. On the other hand, the extended controller is able to handle the disturbance with smaller actuation of the tilt angle while reducing the spinning velocity. In addition, it is able to keep a small tilt angel while gliding over the surface.

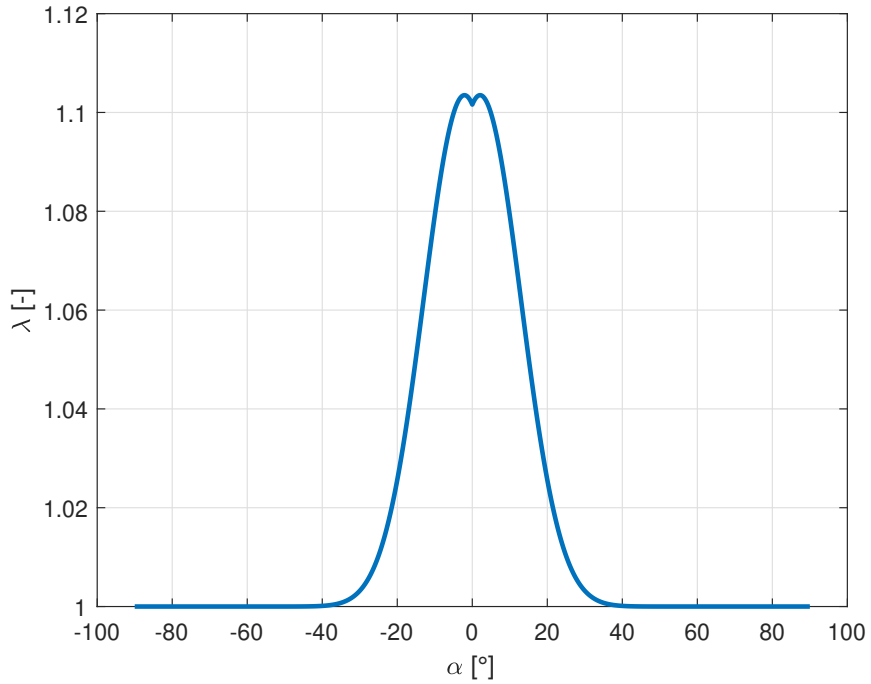


Figure 5.6: Modified proximity effect with only dependency on α

5.3.3 Flight scenario 3

This scenario is based on a task where the platform has to fly close to an object to perform certain operation, such as visual inspection or even manipulation. Hereby, it is not necessary for the platform to fully hover over a surface or object. Therefore, only the two propeller in the front are experiencing proximity effect. In case of the reference controller, there is not any unusual behaviour when the platform move from point A to B, see Figure 5.13. When it comes to the extended model in Figure 5.11, behaviour in the beginning matches the reference case, but as soon the front propellers are over the surface there is pitch disturbance which is handled by actuating the tilt angles. The front propeller open up the wides due to the increase of thrust, where the others are forced to balance out the resulting forces to stay in place. It is noticeable that the controller is going against the optimization by going below the resting spinning velocity set the cost function, which also result in a slow decline of the tilt angle for propeller one and six. The simulation of the extended controller, see Figure 5.15, reacts differently. As it has the ability to reduce the resting velocity of the front propellers immediately, it is not force to request big tilt angle to compensate the thrust. As the platform hovers in position B, the front propellers are settling at $\alpha \approx \pm 1.5°$ similar to scenario one and the other propellers are not as zero in order to compensate for lateral forces.

5.4 Energy consumption

The energy of the system is consumed by three components: propeller motor, tilt motor and onboard electronics. However, the majority of the energy is going towards driving the propeller. Therefore, the energy consumption of the tilt motor and onboard electronic is not taken into consideration. In order to spin the propeller, the motor has to

provide a torque against the air drag cause by the propeller spinning. Given that, the motor torque is defined as follows:

$$|\tau_{m_i}| = |\tau_{P_i}| = c_\tau \omega_{P_i}^2 \tag{5.11}$$

where $\tau_{m_i} \in \mathbb{R}$ is the motor torque.Following that, the total power consumed by the platform is

$$P_{tot} = \sum_i^n P_i = \sum_i^n \tau_{m_i} \omega_{P_i} = \sum_i^n c_\tau \omega_{P_i}^3 \tag{5.12}$$

and the dissipated energy is equal to

$$E = \int P_{tot}\, dt = \int c_\tau \omega_{P_i}^3\, dt. \tag{5.13}$$

Figure 5.16, 5.17 and 5.18 show the total dissipated energy over the simulation time for each scenario using Eq.5.13. The energy has been normalized with respect to the reference control for each case. Overall, it shows a decrease in energy consumption when the proximity effect is exploited by the controller. Keep in mind, the platform is only exposed to the surface for a limited time in the simulation, and a longer flight operation will increase the energy difference to the reference scenario.

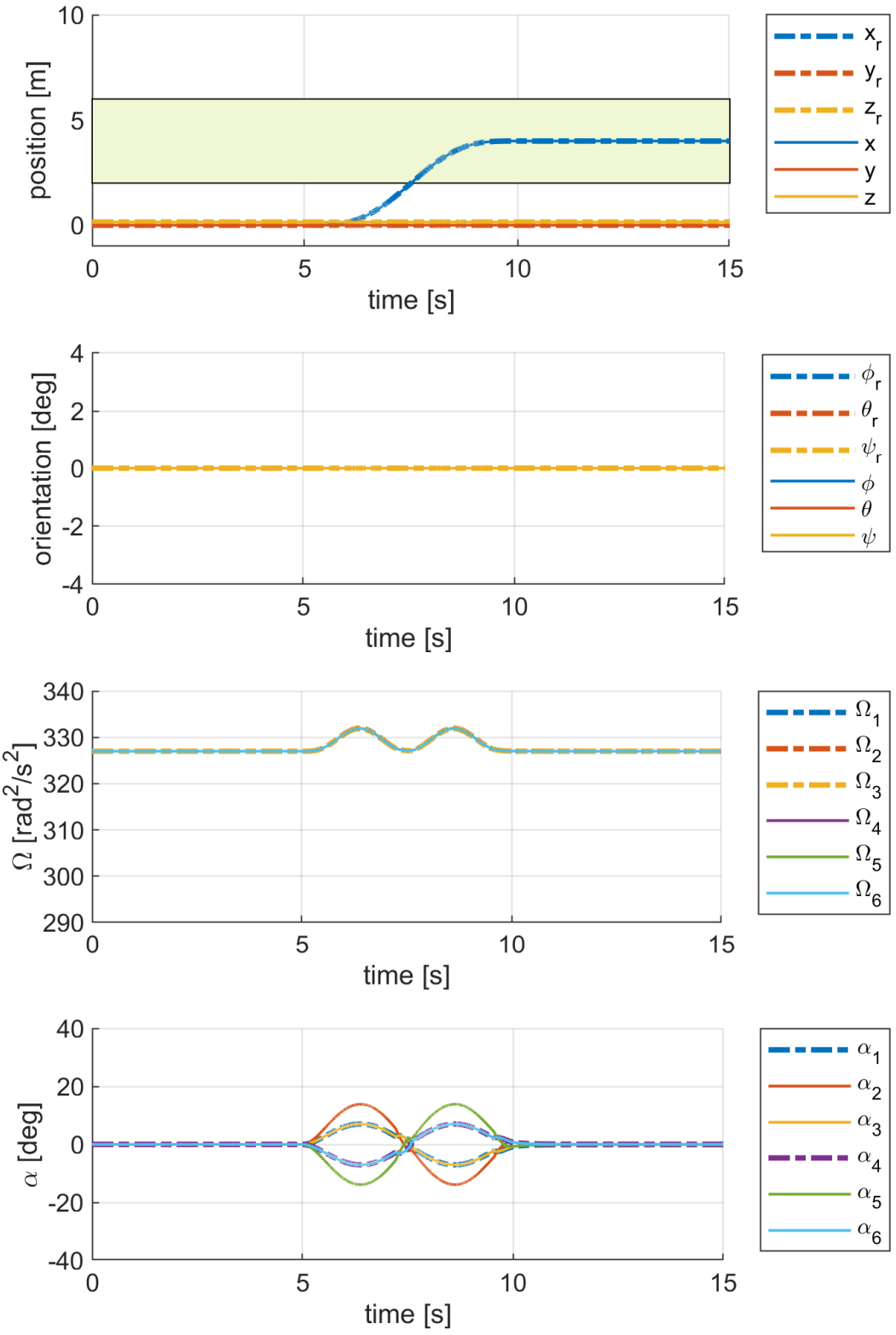


Figure 5.7: Flight 1: reference control

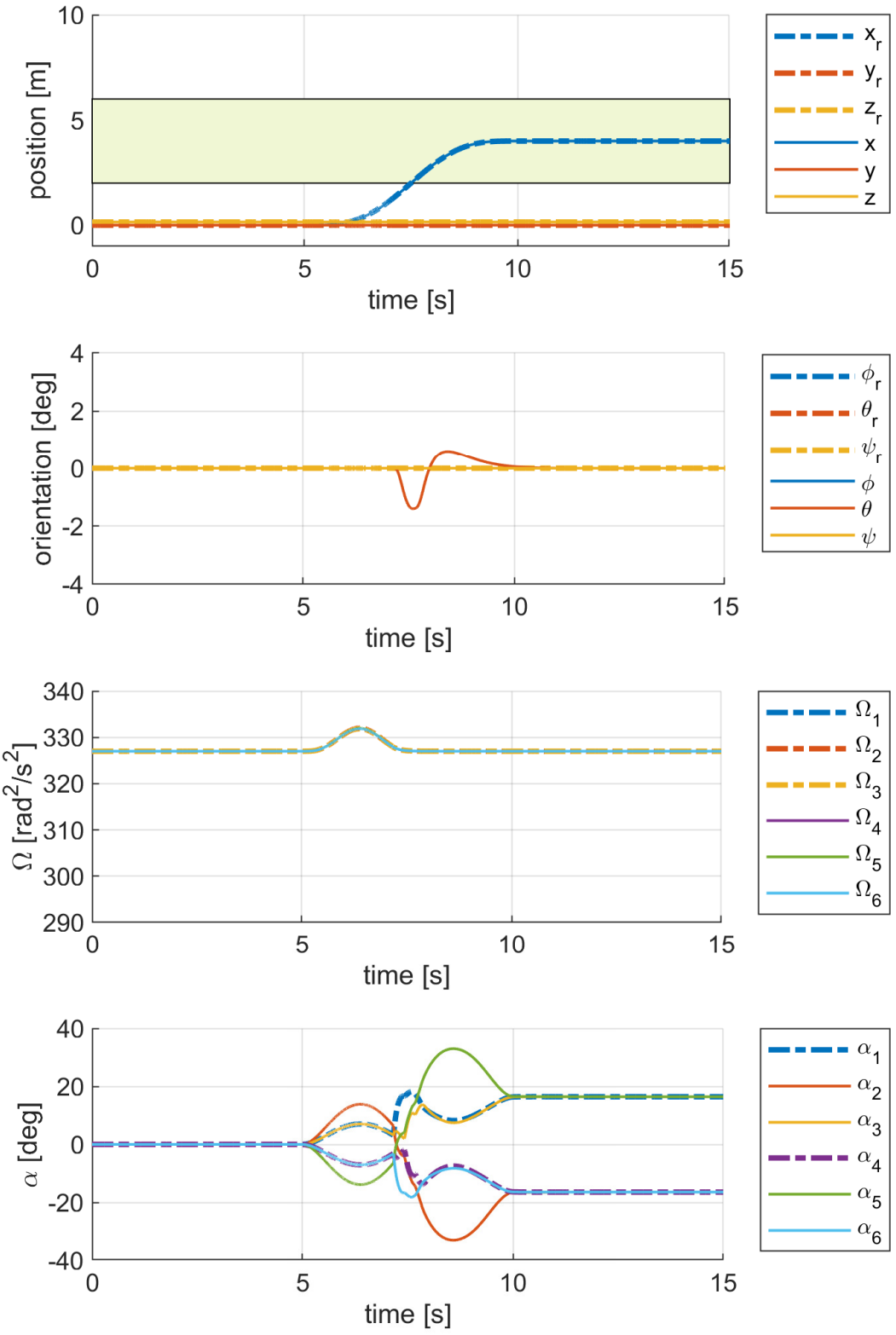


Figure 5.8: Flight 1: extended model

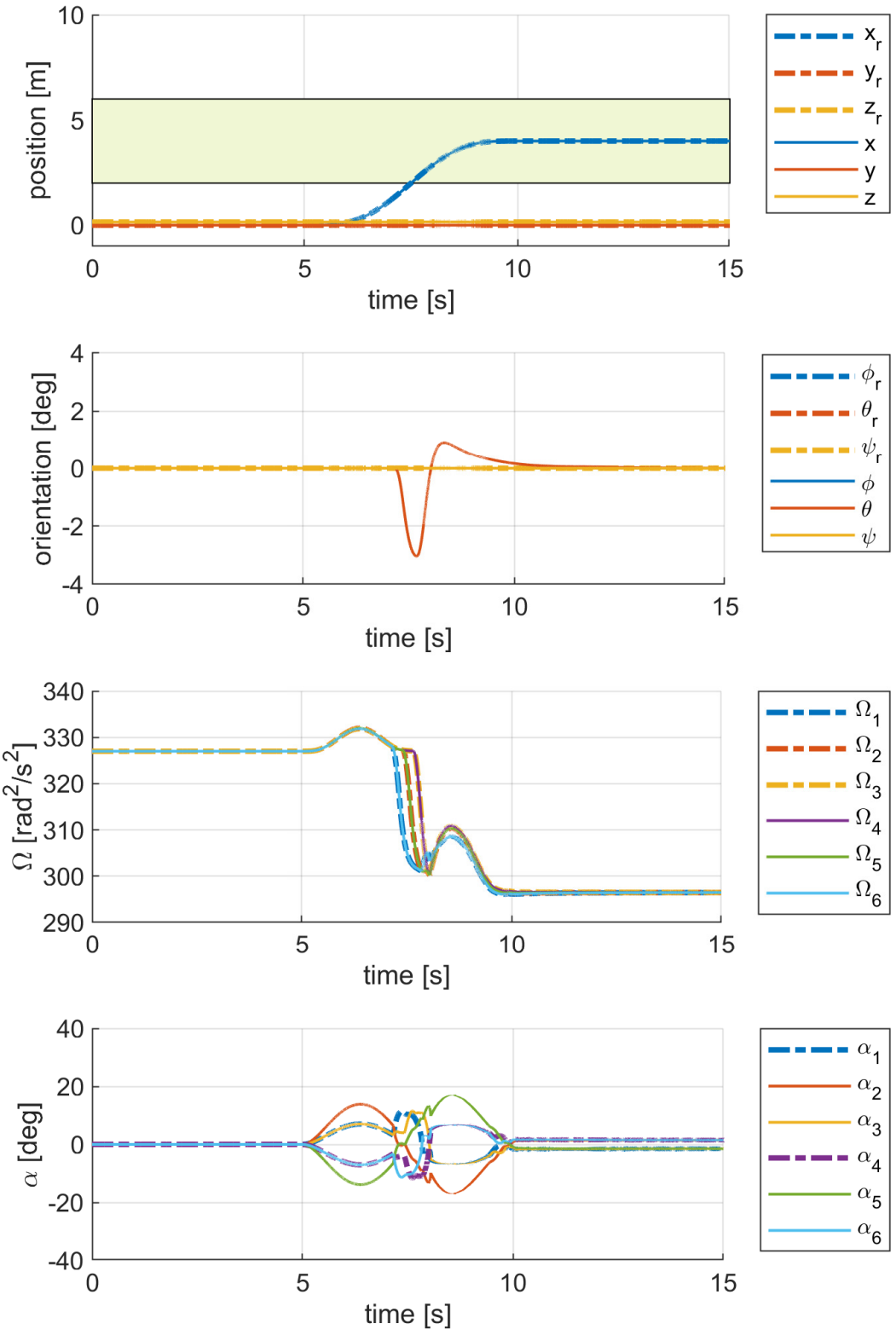


Figure 5.9: Flight 1: extended control

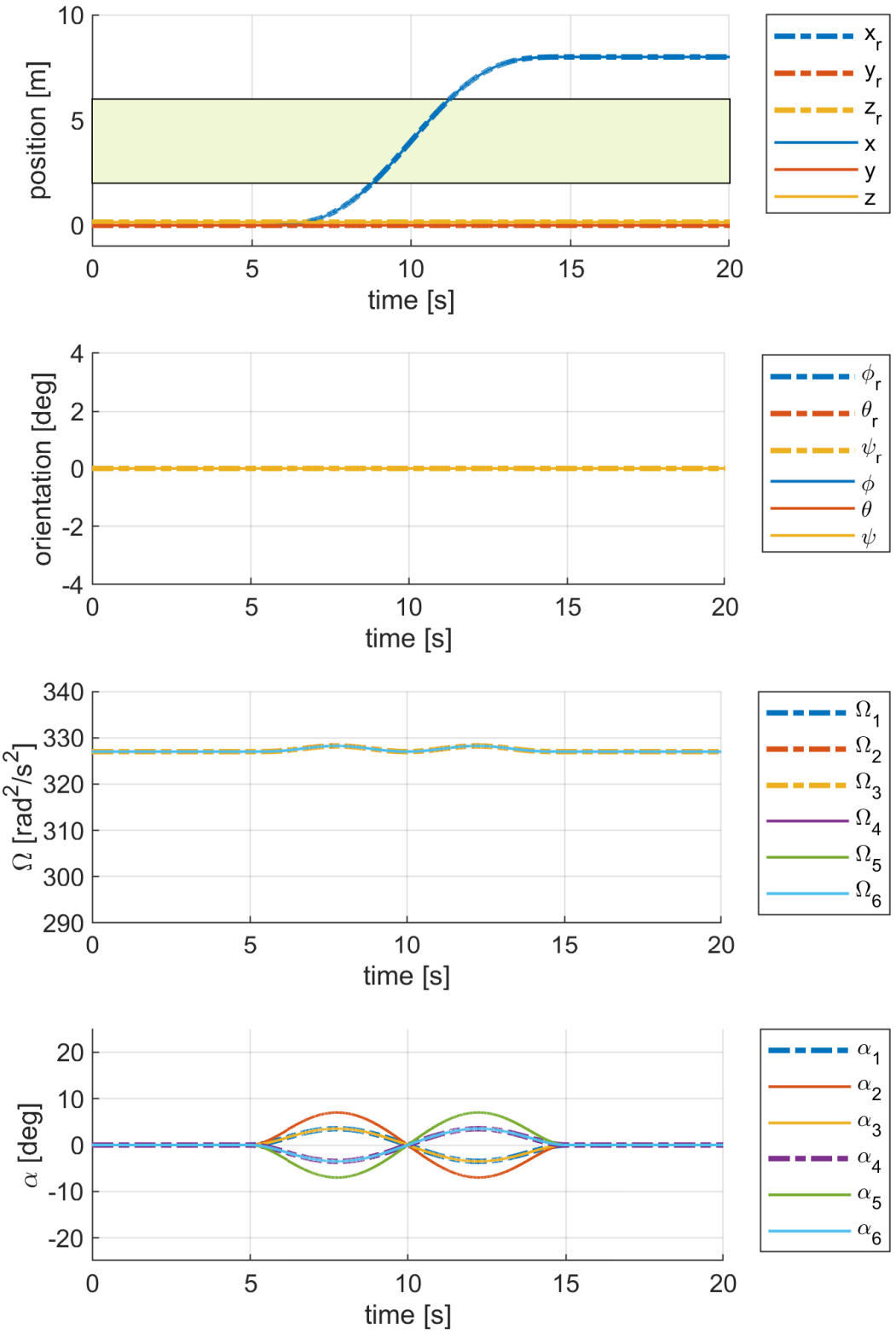


Figure 5.10: Flight 2: reference control

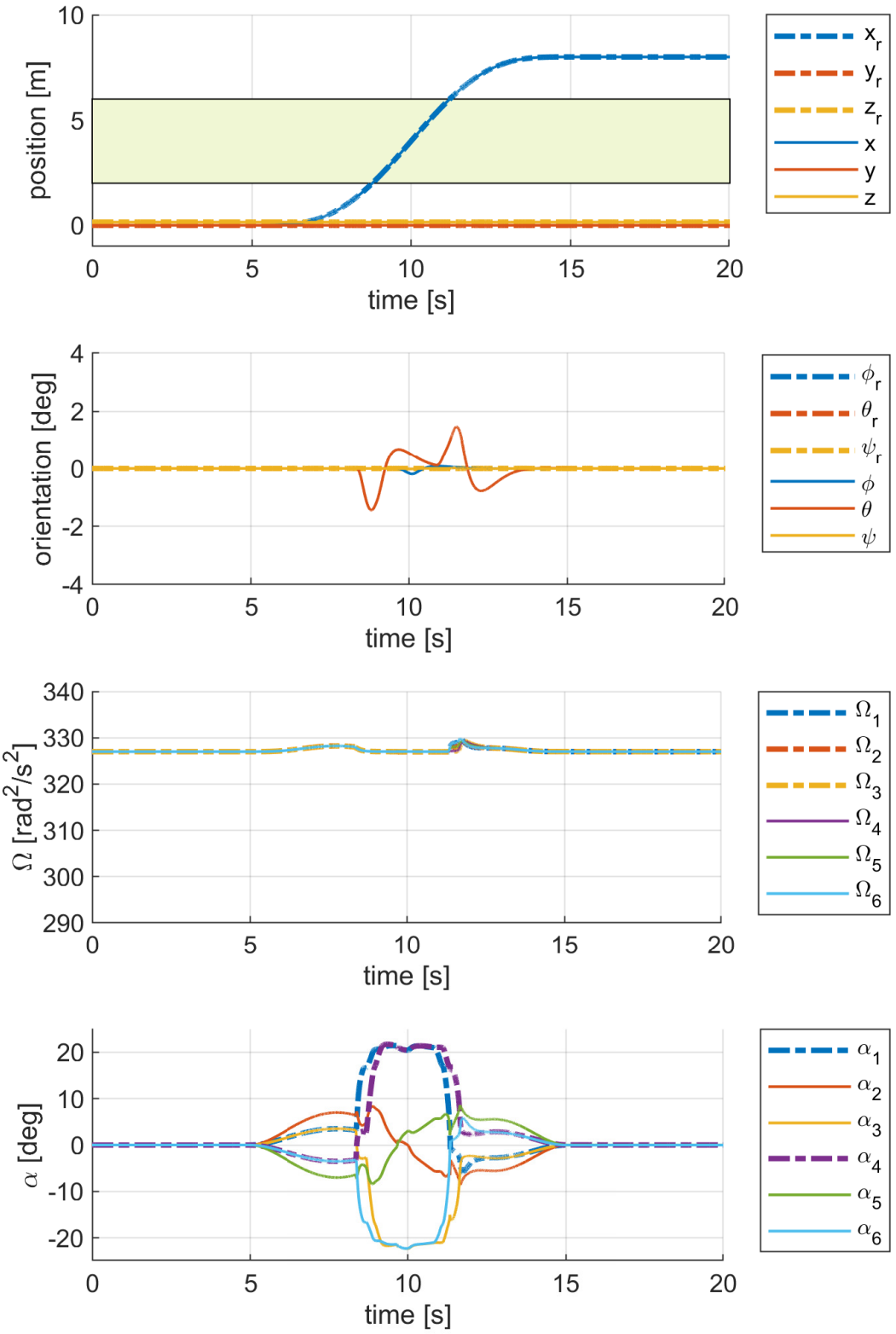


Figure 5.11: Flight 2: extended model

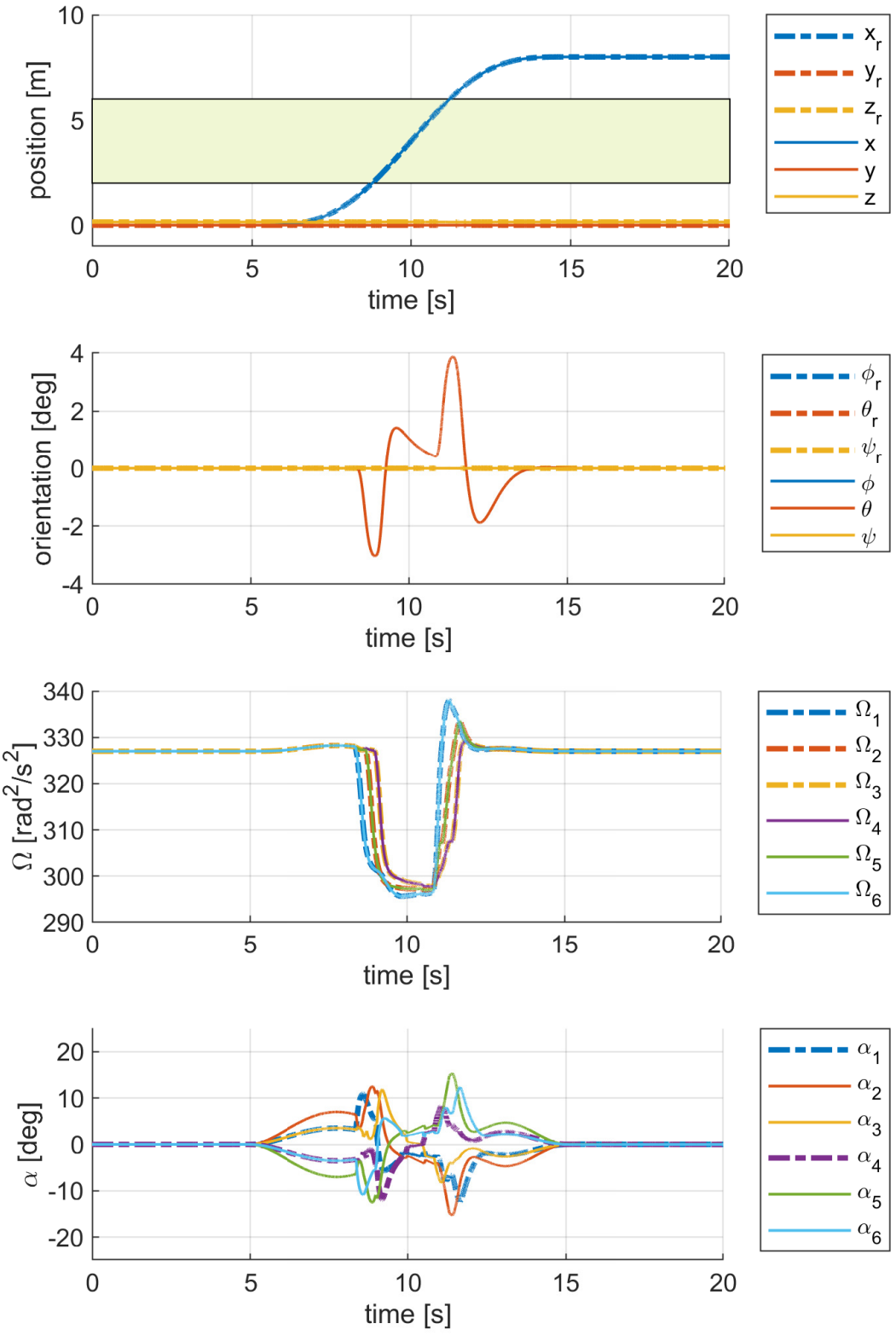


Figure 5.12: Flight 2: extended control

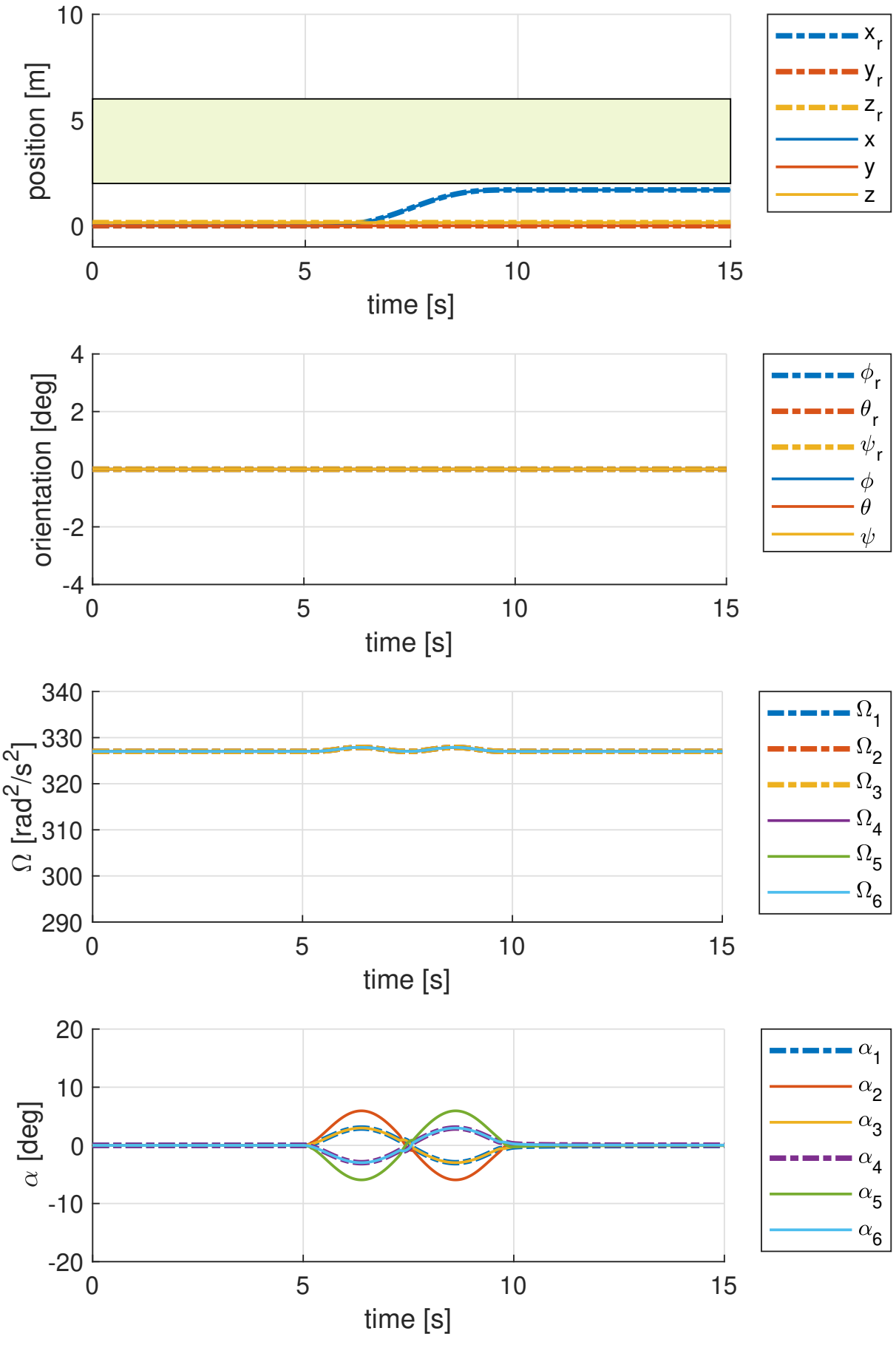


Figure 5.13: Flight 3: reference control

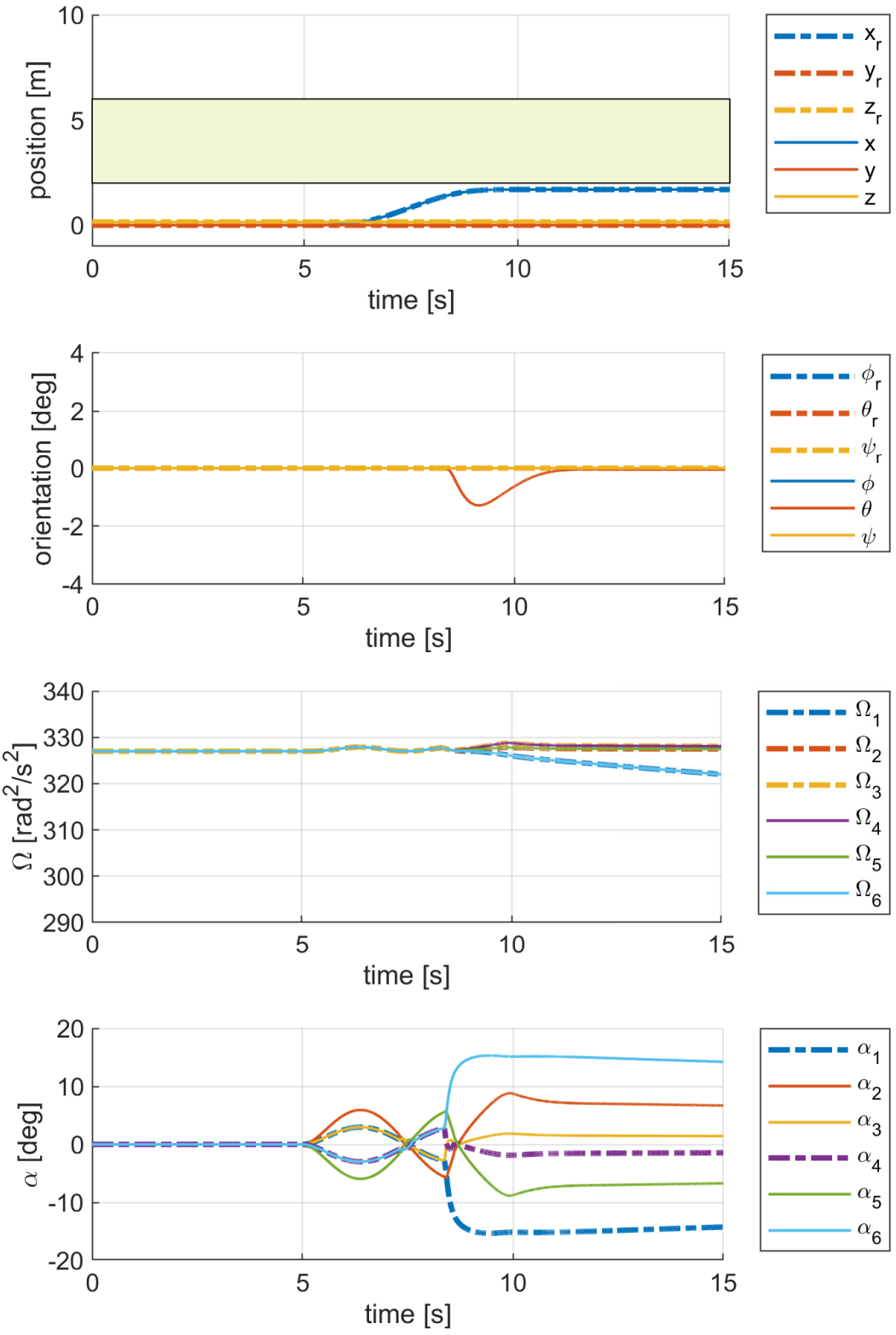


Figure 5.14: Flight 3: extended model

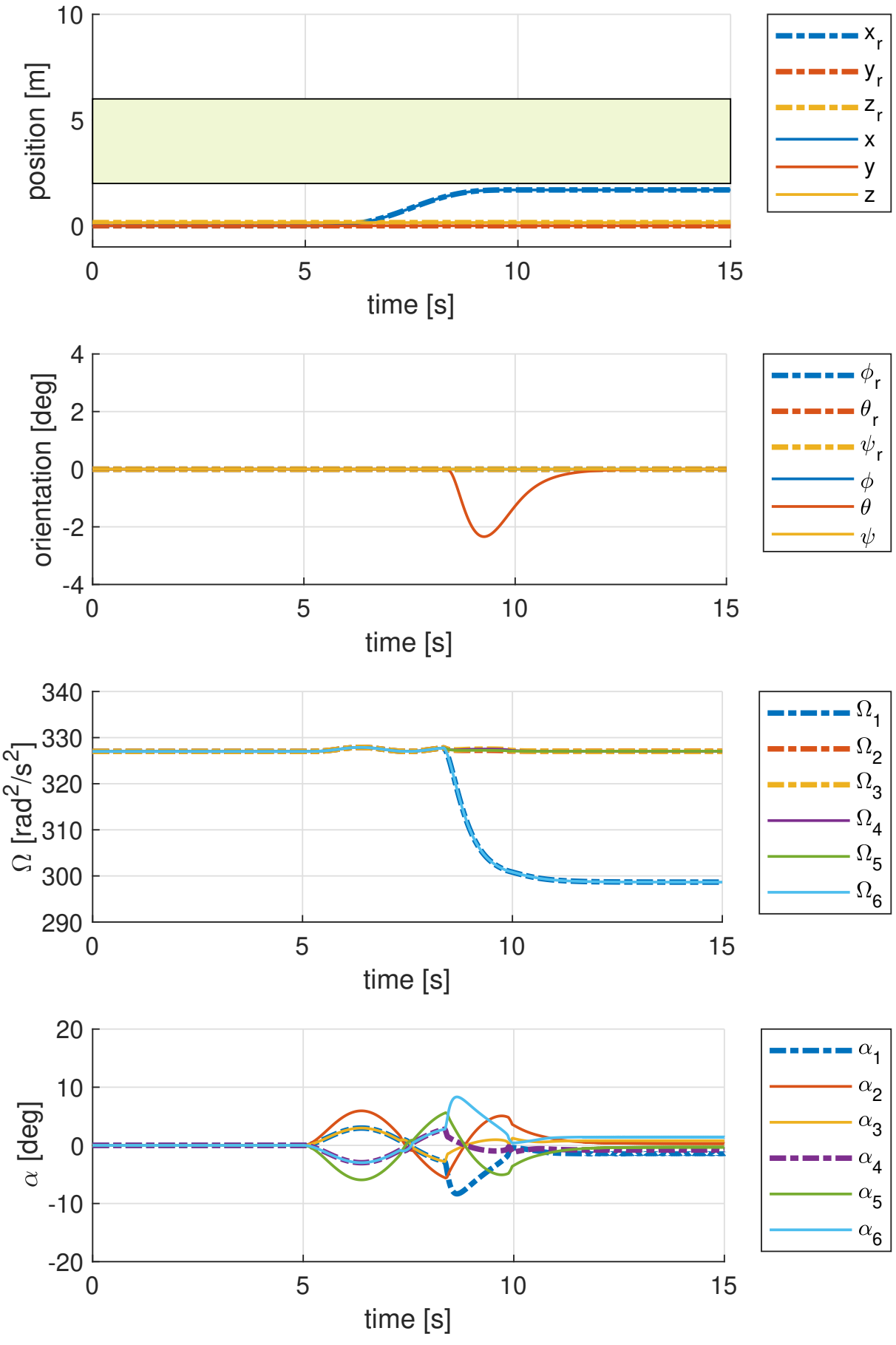


Figure 5.15: Flight 3: extended control

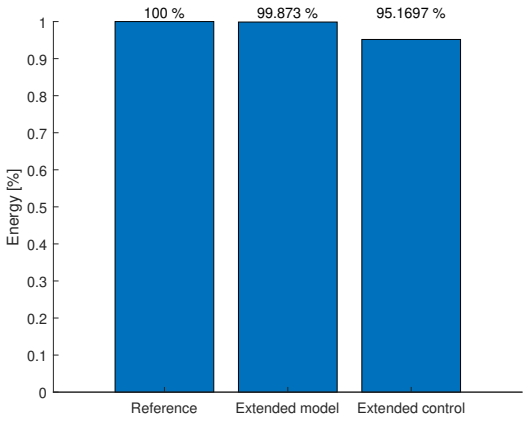


Figure 5.16: Energy flight 1

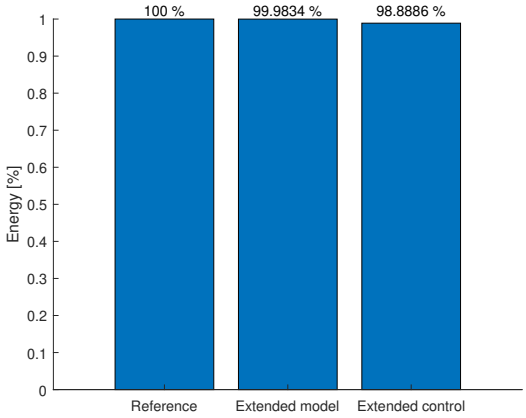


Figure 5.17: Energy flight 2

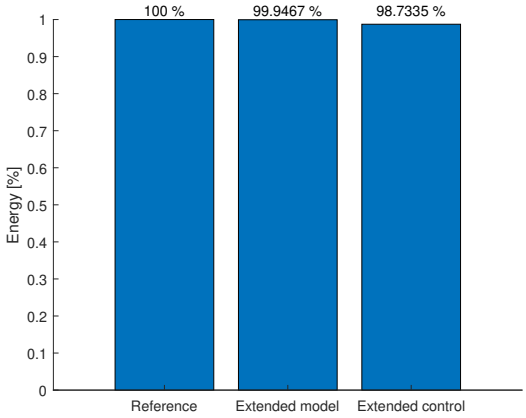


Figure 5.18: Energy flight 3

6 CONCLUSION AND FUTURE WORK

6.1 Conclusion

The following main research question has been posed at the beginner of the thesis:

- How to exploit the proximity effect for variable tilt-rotor drones in flight operations?

Based upon a literature review of the state of the art, the two sub question has been formulated to assist in answering the main question:

1. How to modify a controller to take into account the proximity effect to generate optimal thrust close to surfaces?

2. Can the energy consumption of the drone be reduced by optimizing thrust based on the proximity effect?

The first sub question is aimed towards finding the ideal approach to involve the proximity effect into the controller. The solution has been found to be modifying the control allocation. This circumvents the issue to involve an optimization based controller because of overactuated nature of the variable tilt multirotor and non-linearities in the system. Furthermore, it allows extending existing control approaches without changing the underlying principles. The modification on the control allocation on the chosen controller [4] shows overall good performance in all simulated scenarios. Although, the original controller is able to handle any disturbance due to the proximity effect, the actuation is not optimal. In general, the controller commanded wide tilt angel in order to compensate the additional thrust. Thus, it will reduce the efficiency of the platform. However, informing the controller by involving the proximity effect into the allocation yields smaller tilt actuation and keeps the propeller in an optimal tilt position.

The second sub question focuses on the exploitation of the proximity effect in order to reduce the overall energy consumption. This is achieved by reducing the spinning velocity of the propeller during operation. Hereby, the null-space of the allocation is utilized in order to optimize the actuation without interference on the control objective. With the extended controller, the overall dissipated energy has been reduced by almost 5% in the best flight scenario. It is able to achieve that by reducing the propeller speed and maintaining the optimal tilt angle during the flight. The amount of energy saved, highly depends on the time spend in the presence of the surface. It is interesting to see, that a quick fly by over a surface saves the almost same amount as partially hovering over a surface. All in all, the controller was able to reduce the energy consumption by including the proximity effect in the allocation process.

In summery, an approach has been developed to exploit the proximity effect for a variable-tilt drone. The exploitation leads to reduced energy consumption while flying near to a surface. This achieved by extending the control allocation with a proximity model for tilted propellers and a gradient descent based optimization for the purpose of optimal directed thrust. With the increase of exposer time to surfaces, the energy difference will grow significantly and extend the total flight time. This will be beneficial for application operating in confined environment.

6.2 Future work

Future extension based on this thesis:

- Discretization of the controller and validation of its stability for a real drone implementation. Different from the simulation at almost real time, the real hardware will not always have the performance to execute the control algorithm at high rate and the discrete signal can introduce time delay which could lead to instability of the system.
- Experimental validation on a real drone. Implementing the control approach on real variable tilt multirotor platform to prove feasibility and have realistic feedback on the proximity effect.
- Extend the controller with a surface detection or observer. At the moment, some knowledge of the environment is needed to determine the presence of a surface for simulation purpose. In case of a real drone, a distance sensor or camera will have to perceive the environment. It could be possible to distinguish a surface based on the control behaviour. For example, considering the ground surface case, the propellers remain tilted when further reduction of the spinning velocity is not allowed by the cost function. On the basis of that, an algorithm could develop to switch into a different modus.
- Extend the control approach to interaction control. This could be achieved adding an outer loop to the controller and apply impedance or admittance control. The difficulty is, the proposed controller and allocation operate on the base of linear and angular jerk. A possible approach could be an admittance filter [19]. The pose controller in the inner loop could be replaced by this approach, and the outer loop requires a desired trajectory $(\mathbf{p}_d(t), \mathbf{R}_d(t)) \in \mathcal{C}^2$ for the end effector, which could be extended to $(\mathbf{p}_d(t), \mathbf{R}_d(t)) \in \mathcal{C}^3$.
- Investigate the wall effect for different propeller position around the body.
- Take into account the energy of the tilt motors into the optimization process. With this, the allocation has to determine if a tilt actuation is beneficial or not.

REFERENCES

[1] Antonio Matus-Vargas, Gustavo Gómez, and Jose Martinez-Carranza. Ground effect on rotorcraft unmanned aerial vehicles: a review. *Intelligent Service Robotics*, 14:1–20, 03 2021.

[2] Mahmoud Hamandi, Federico Usai, Quentin Sablé, Nicolas Staub, Marco Tognon, and Antonio Franchi. Design of Multirotor Aerial Vehicles: A Taxonomy Based on Input Allocation. *The International Journal of Robotics Research*, 40(8-9):1015–1044, September 2021.

[3] Anibal Ollero, Marco Tognon, Alejandro Suarez, Dongjun Lee, and Antonio Franchi. Past, present, and future of aerial robotic manipulators. *IEEE Transactions on Robotics*, 38(1):626–645, 2022.

[4] Markus Ryll, Heinrich H. Bülthoff, and Paolo Robuffo Giordano. A novel overactuated quadrotor unmanned aerial vehicle: Modeling, control, and experimental validation. *IEEE Transactions on Control Systems Technology*, 23(2):540–556, 2015.

[5] Mina Kamel, Sebastian Verling, Omar Elkhatib, Christian Sprecher, Paula Wulkop, Zachary Taylor, Roland Siegwart, and Igor Gilitschenski. The voliro omniorientational hexacopter: An agile and maneuverable tiltable-rotor aerial vehicle. *IEEE Robotics Automation Magazine*, 25(4):34–44, 2018.

[6] Karen Bodie, Zachary Taylor, Mina Samir Kamel, and Roland Siegwart. *Towards Efficient Full Pose Omnidirectionality with Overactuated MAVs*, pages 85–95. 01 2020.

[7] Markus Ryll, Davide Bicego, Mattia Giurato, Marco Lovera, and Antonio Franchi. Fast-hex - A morphing hexarotor: Design, mechanical implementation, control and experimental validation. *CoRR*, abs/2004.06612, 2020.

[8] Mike Allenspach, Karen Bodie, Maximilian Brunner, Luca Rinsoz, Zachary Taylor, Mina Kamel, Roland Siegwart, and Juan Nieto. Design and optimal control of a tiltrotor micro-aerial vehicle for efficient omnidirectional flight. *The International Journal of Robotics Research*, 39(10-11):1305–1325, 2020.

[9] Caitlin Powers, Daniel Mellinger, Aleksandr Kushleyev, Bruce Kothmann, and Vijay Kumar. *Influence of Aerodynamics and Proximity Effects in Quadrotor Flight*, pages 289–302. 01 2013.

[10] P.J. Sanchez-Cuevas, Guillermo Heredia, and Anibal Ollero. Characterization of the aerodynamic ground effect and its influence in multirotor control. *International Journal of Aerospace Engineering*, 2017:1–17, 08 2017.

[11] P. J. Sanchez-Cuevas, G. Heredia, and A. Ollero. Multirotor uas for bridge inspection by contact using the ceiling effect. In *2017 International Conference on Unmanned Aircraft Systems (ICUAS)*, pages 767–774, 2017.

[12] Xiang He, Gordon Kou, Marc Calaf, and Kam Leang. In-ground-effect modeling and nonlinear disturbance observer for multi-rotor uav control. *Journal of Dynamic Systems, Measurement, and Control*, 141, 03 2019.

[13] Shijie Gao, Carmelo Di Franco, Darius Carter, Daniel Quinn, and Nicola Bezzo. Exploiting ground and ceiling effects on autonomous uav motion planning. In *2019 International Conference on Unmanned Aircraft Systems (ICUAS)*, pages 768–777, 2019.

[14] Xinyue Kan, Justin Thomas, Hanzhe Teng, Herbert G. Tanner, Vijay Kumar, and Konstantinos Karydis. Analysis of ground effect for small-scale uavs in forward flight. *IEEE Robotics and Automation Letters*, 4(4):3860–3867, 2019.

[15] A. Garofano-Soldado, G. Heredia, and A. Ollero. Aerodynamic interference in confined environments with tilted propellers: Wall effect and corner effect. In *2021 Aerial Robotic Systems Physically Interacting with the Environment (AIRPHARO)*, pages 1–8, 2021.

[16] Runze Ding, Yi-Hsuan Hsiao, Huaiyuan Jia, Songnan Bai, and Pakpong Chirarattananon. Passive wall tracking for a rotorcraft with tilted and ducted propellers using proximity effects. *IEEE Robotics and Automation Letters*, 7(2):1581–1588, 2022.

[17] P.J. Sanchez-Cuevas, Guillermo Heredia, and Anibal Ollero. *Multirotor Aerodynamic Effects in Aerial Manipulation*, pages 67–82. 06 2019.

[18] Tor Johansen and Thor Fossen. Control allocation—a survey. *Automatica*, 49:1087–1103, 05 2013.

[19] Markus Ryll, Giuseppe Muscio, Francesco Pierri, Elisabetta Cataldi, Gianluca Antonelli, Fabrizio Caccavale, Davide Bicego, and Antonio Franchi. 6d interaction control with aerial robots: The flying end-effector paradigm. *The International Journal of Robotics Research*, 38:027836491985669, 06 2019.