UNIVERSIDADE FEDERAL DE SANTA CATARINA
PROGRAMA DE PÓS-GRADUAÇÃO EM
ENGENHARIA DE AUTOMAÇÃO E SISTEMAS

Rodrigo Donadel

Modeling and Control of a Tiltrotor Unmanned Aerial Vehicle for Path Tracking

Florianópolis

2015

Rodrigo Donadel

Modeling and Control of a Tiltrotor Unmanned Aerial Vehicle for Path Tracking

Dissertação de mestrado submetida ao Programa de Pós-Graduação em Engenharia de Automação e Sistemas para a obtenção do Grau de Mestre em Engenharia de Automação e Sistemas.
Orientador: Prof. Dr. Leandro Buss Becker
Coorientador: Prof. Dr. Guilherme Vianna Raffo

Florianópolis

2015

Ficha de identificação da obra elaborada pelo autor,
através do Programa de Geração Automática da Biblioteca Universitária da UFSC.

Donadel, Rodrigo
Modeling and control of a tiltrotor unmanned aerial vehicle for path tracking / Rodrigo Donadel ; orientador, Leandro Buss Becker ; coorientador, Guilherme Vianna Raffo. - Florianópolis, SC, 2015.
143 p.

Dissertação (mestrado) - Universidade Federal de Santa Catarina, Centro Tecnológico. Programa de Pós-Graduação em Engenharia de Automação e Sistemas.

Inclui referências

1. Engenharia de Automação e Sistemas. 2. Modelagem dinâmica. 3. Controle ótimo. 4. Tiltrotor. 5. Veículo aéreo não-tripulado. I. Becker, Leandro Buss. II. Raffo, Guilherme Vianna. III. Universidade Federal de Santa Catarina. Programa de Pós-Graduação em Engenharia de Automação e Sistemas. IV. Título.

Rodrigo Donadel

Modeling and Control of a Tiltrotor Unmanned Aerial Vehicle for Path Tracking

Esta Dissertação foi julgada adequada para obtenção do Título de "Mestre em Engenharia de Automação e Sistemas", e aprovada em sua forma final pelo Programa de Pós-Graduação em Engenharia de Automação e Sistemas.

Florianópolis, 22 de Abril de 2015.

Prof. Rômulo Silva de Oliveira
Coordenador do Programa de Pós-Graduação em Engenharia de Automação e Sistemas

---

Prof. Leandro Buss Becker, Dr. - PGEAS UFSC
Orientador

Prof. Guilherme Vianna Raffo, Dr. - PPGEE UFMG
Coorientador

Banca Examinadora:

Prof. Leandro Buss Becker, Dr.
Presidente - PGEAS UFSC

Prof. Eugênio de Bona Castelan Neto, Dr. - PGEAS UFSC

Prof. Hector Bessa Silveira, Dr. - PGEAS UFSC

Prof. Ebrahim Samer El Youssef, Dr. - UFSC - Blumenau

Prof. Julio Elias Normey Rico Dr. - PGEAS UFSC

To my caring parents, Nelso and Lucia.
To my best friends, Marcelo and Marilia.
To my love, Joana.

Acknowledgments

I want first to thank my family: my father Nelso, my mother Lucia, my brother Marcelo and my sister Marilia. Even though most of the time I could not be with them, their support and understanding were essential to my studies. To my love Joana, who was always there for me, I love you with all my heart.

I am gratefull for all the support and hard work from the members of Project ProVANT: Patrick, Martin, Richard, João Paulo, Fernando Rodrigues, Interlandi, Willian, Vinicius, Gabriel and Izabele. A special thank you for Fernando Gonçalves, whose dedication to the project is unvaluable, as is his friendship.

I would also like to thank my advisors, Guilherme Vianna Raffo and Leandro Buss Becker, for all that they tought me and for supporting me with great disposition throughout my thesis, sometimes even during weekends. Thank you very much, know that I appreciate all you have done for me.

And a special thank you for Vinicius Stramosk, Eduardo Hulse, Douglas Bertol, Daniel Martins and all the others who showed genuine interest and supported me even though they were not part of project ProVANT.

“Somewhere, something incredible is waiting to be known.“

Carl Sagan

Abstract

This master thesis deals with the modeling and control of a small scale birotor tiltrotor unmanned aerial vehicle (UAV). A tiltrotor is characterized by a mechanism that tilts the aircraft's rotors in order to control the flight. An UAV with such characteristics is being developed by this author and other researchers in the scope of the project named ProVANT. The developed UAV prototype is used in this work to obtain experimental results.

This kind of system can be characterized by its underactuated, highly nonlinear, and coupled dynamics. Instead of using a dynamic model available in literature, this work proposes a more accurate model considering the UAV as a multibody system. By doing so the tilting angles become generalized coordinates and the tilt mechanism dynamics are naturally added to the model, as well as the coupling between the bodies. The result is an eight degrees of freedom model obtained through Euler-Lagrange formulation.

The path tracking problem is solved here with linear optimal controllers for the full model, instead of the classical approach of cascade control for the translation and rotation subsystems. The developed controllers are linear quadratic regulators, a H_∞ controller and a multiobjective H_2/H_∞ controller, all with LMI formulation. A nonlinear backstepping controller taken from the literature is implemented in order to be compared with the designed controllers.

In addition, controllers for the hovering problem are also designed to be used in experiments with ProVANT's tiltrotor. They reduce the complexity of the initial experimental flights, focusing not only in the validation of the control system, but the complete project, including its electronics, mechanical design, and additional software. Such experiments are presented and discussed in details along this work.

The work also addresses how flight-related information are gathered and processed. This includes the design of a nonlinear complementary filter for the attitude estimation that works with data acquired from the UAV sensors.

Keywords: UAV, tiltrotor, Euler-Lagrange, path tracking, LMI, robust control, LQR, H_∞, H_2, optimal control, attitude estimation.

Resumo

Esta dissertação de mestrado aborda a modelagem e o controle de um veiculo aéreo não-tripulado (VANT) de pequena escala do tipo birotor tiltrotor. Um tiltrotor é caracterizado por um mecanismo que possibilita inclinar (do inglês "tilt") seus rotores. Um VANT com esta configuração está sendo desenvolvido como parte de um projeto chamado ProVANT, o qual é usado neste trabalho para obter resultados experimentais.

Este tipo de sistema é caracterizado por sua dinâmica subatuada, fortemente acoplada e não-linear. Ao invéns de usar um modelo dinâmico disponível na literatura, este trabalho propõe um modelo mais preciso que considera o tiltrotor como um sistema multi-corpos. Com isto os ângulos de inclinação dos rotores tornam-se coordenadas generalizadas do sistema e a dinâmica do mecanismo que inclina os rotores é naturalmente incluída no modelo, assim como o acoplamento entre os corpos. O resultado é um modelo com oito graus de liberdade obtido com a formulação Euler-Lagrange.

O problema de seguimento de trajetórias é resolvido com controladores lineares ótimos para o modelo completo, ao invés da abordagem clássica de controladores em cascata para os subsistemas de translação e rotação. Os controladores desenvolvidos são reguladores lineares quadráticos, um controlador $\mathcal{H}_\infty$ e um controlador multiobjetivo $\mathcal{H}_2/\mathcal{H}_\infty$, todos com uma formulação LMI. Um controlador não-linear do tipo backstepping retirado da literatura foi implementado a fim de se comparar com os controladores projetados.

Ademais, também foram projetados controladores mais simples para o problema de voo pairado. Tais controladores foram desenvolvidos com o intuito de utiliza-los nos testes iniciais com o tiltrotor do projeto ProVANT. Eles reduzem a complexidade dos primeiros voos experimentais, concentrando-se não apenas na validação do sistema de controle, mas também do projeto como um todo, incluindo a eletrônica, o projeto mecânico e software. Tais experimentos são apresentados e discutidos em detalhes ao longo deste trabalho.

Este trabalho também aborda como as informações relativas ao voo são adquiridas e processadas. Isto inclui o projeto de um filtro complementar não-linear para a estimação da atitude, o qual funciona com dados

obtidos de sensores presentes no VANT.

Palavras-chave: VANT, tiltrotor, Euler-Lagrange, seguimento de trajetória, LMI, controle robusto, LQR, H_∞, H_2, controle ótimo, estimação de atitude.

Contents

List of Figures

List of Tables

List of Acronyms

CAD	computer-aided design
DAS	Automation and Systems Department
DELT	Electronic Engineering Department
DOFs	degrees of freedom
ESC	electronic speed controller
FAA	Federal Aviation Administration of the United States
GPS	Global Positioning System
IMU	inertial measurement system
LMI	linear matrix inequality
LQR	linear quadratic regulator
LTV	linear time varying system
UAVs	Unmanned aerial vehicles
UFMG	Federal University of Minas Gerais
UFSC	Federal University of Santa Catarina
VTOL	vertical take-off and landing

List of Symbols

α_L	Angular displacement (tilt) for the left rotor
α_R	Angular displacement (tilt) for the right rotor
β	Fixed lateral tilt angle for the rotors
$\boldsymbol{\delta}(t)$	Disturbance vector
$\boldsymbol{\eta}$	Angular position of the UAV using Euler angles
$\hat{\boldsymbol{b}}_\omega$	Gyroscope bias estimation
$\hat{\boldsymbol{R}}$	Rotation matrix estimation
$\boldsymbol{\omega}_m^{\mathfrak{B}}$	Gyroscopes measurement
$\boldsymbol{\xi}$	Linear position vector, distance from the origin o $\mathfrak{B}$ and $\mathfrak{I}$
$\boldsymbol{a}_{\boldsymbol{m}}^{\mathfrak{B}}$	Accelerometers measurement
$\boldsymbol{b}_\omega$	Gyroscopes bias
$\boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}})$	Coriolis and centrifugal force matrix
$\boldsymbol{d}_1$	Distance between frames $\mathfrak{C}_1$ and $\mathfrak{B}$
$\boldsymbol{d}_2$	Distance between frames $\mathfrak{C}_2$ and $\mathfrak{B}$
$\boldsymbol{d}_3$	Distance between frames $\mathfrak{C}_3$ and $\mathfrak{B}$
$\boldsymbol{F}(\boldsymbol{q})$	External Force Vector
$\boldsymbol{G}(\boldsymbol{q})$	Gravitational force vector
$\boldsymbol{m}_e^{\mathfrak{I}}$	Earth's magnetic field

$\boldsymbol{m}_{\boldsymbol{m}}^{\mathfrak{B}}$	Magnetometers measurement
$\boldsymbol{M}(\boldsymbol{q})$	Inertia Matrix
$\boldsymbol{q}$	Generalized coordinate vector
$\boldsymbol{R}$	Rotation matrix
$\boldsymbol{R}_{\mathfrak{B}}^{\mathfrak{I}}$	Rotation matrix from frame $\mathfrak{B}$ to frame $\mathfrak{I}$
$\boldsymbol{w}_{\mathfrak{B}\mathfrak{I}}^{\mathfrak{B}}$	Angular velocity of frame $\mathfrak{B}$ in relation to frame $\mathfrak{I}$, expressed in frame $\mathfrak{B}$
$\boldsymbol{x}(t)$	State vector
$\mathfrak{B}$	Body frame, rigdly attached to the UAV's center of rotation
$\mathfrak{C}_1$	Frame rigdly attached to the center of mass of the UAV's main body
$\mathfrak{C}_2$	Frame rigdly attached to the center of mass of the UAV's right rotor
$\mathfrak{C}_3$	Frame rigdly attached to the center of mass of the UAV's left rotor
$\mathfrak{I}$	Inertial reference frame
ϕ	Roll angle
ψ	Yaw angle
ρ	Mass density
θ	Pitch angle
$\vec{X}_{\mathfrak{B}}$	X axis for frame $\mathfrak{B}$
$\vec{Y}_{\mathfrak{B}}$	Y axis for frame $\mathfrak{B}$
$\vec{Z}_{\mathfrak{B}}$	Z axis for frame $\mathfrak{B}$
f_L	Thrust force for the left rotor
f_R	Thrust force for the right rotor
g	Earth's gravity
$h^{\mathfrak{S}}$	Sonar measurement

I	Inertia tensor
J	Displaced inertia tensor
m	UAV's total mass
m_1	Main body's mass
m_2	Right rotor mass
m_3	Left rotor mass
$p^{\mathfrak{B}}$	Point expressed on the body frame
R^3	Three-dimensional Euclidean space
t	Time
v	Velocity of a single particle

Chapter 1

Introduction

1.1 Motivation

Unmanned aerial vehicles (UAVs) are flying machines that operate either autonomously or remotely. Until the 90's the use of UAVs has almost exclusively been military. The advance of new technologies brought forth miniaturization, price reduction, and enhanced performance for electronic devices to a level that made the development and commercialization of UAVs affordable for the civil sphere.

Teal Group's 2014 market study estimates that UAV spending will nearly double over the next decade, from $6.4 billion annually to $11.5 billion, totaling almost $91 billion in the next ten years. The study calculates the UAV market at 89% military and 11% civil cumulative for the decade, with the numbers shifting to 86% military and 14% civil by the end of the 10-year forecast [1]. The Federal Aviation Administration of the United States (FAA) estimates roughly 7,500 commercial small UAVs will be operating at the end of five years in the United States alone (FAA, 2014) and shows that the UAV market still has many years of growth to come.

Within this billionaire market, small scale civil UAVs are considered a established technology in some specific fields, with practical

[1] http://www.tealgroup.com/index.php/about-teal-group-corporation/press-releases/118-2014-uav-press-release

applications as pest detection and 3D mapping in precision agriculture, police surveillance and pursuit, border control, cinematographic shoots, and nuclear power plants monitoring. In this scope, helicopter-like, fixed-wing, and quadrotors UAVs are by far the most commercialized and researched. However, another model that is steadily gaining more attention are tiltrotor aircrafts, specially the birotor configuration.

Birotor tiltrotor aircrafts are characterized by two rotors independently coupled with some type of revolute joint, typically servomotors in small scale UAVs, which can tilt the whole rotor to a specified angular position. It has held great interest in the military sphere since the 50's (MARTIN et al., 2000) for its ability to perform vertical takeoff and landing (VTOL) like a helicopter and engage an airplane like flight by tilting its rotors horizontally.

The prime example of military tiltrotor is the Bell Boeing V-22 Osprey shown in figure 1.1. Its advantages are clear: the VTOL enables tactical maneuvers in a battlefield as an helicopter while able to fly as a fixed-wing aircraft to provide higher cruise speed and fuel efficiency (DRWIEGA, 2013).

Figure 1.1 – Bell Boeing V-22 Osprey birotor tiltrotor[2].

Such qualities can no doubt find its uses within the commercial public, leading to the development of the first civil tiltrotors like the AW609 from AugustaWestland, which is shown in figure 1.2. It

[2]Image taken from Boeing's official website http://www.boeing.com/defense/v-22-osprey

is sold as a multi-role aircraft that can be configured for passenger transport, search and rescue, law enforcement, maritime surveillance, training, and other government applications.

Figure 1.2 – AugustaWestland AW609 civil tiltrotor[3].

In respect to unmanned tiltrotors, military project examples are the Bell Eagle Eye and the Lockheed Martin ARES project - figure 1.3 depicts the Eagle Eye. As for the civil market, a commercial UAV modeled after the Boeing V-22 Osprey is being sold by Rotormast (figure 1.4), but its use is still for hobby and cannot perform autonomous flight (an user piloting the UAV via a radio controller is required).

Figure 1.3 – Bell eagle eye UAV tiltrotor[4].

[3]Image taken from http://www.agustawestland.com/products/aw609
[4]Image taken from http://www.naval-technology.com/projects/belleagleeyeuav

Figure 1.4 – Rotormast UAV tiltrotor modeled after the Bell Boeing V-22 Osprey[5]

Nonetheless, tiltrotor UAVs potential is obvious and in the future will substitute other types of UAV in applications that require traveling considerable distances. A birotor tiltrotor when compared to fixed-wing UAVs has the advantage of VTOL; compared to helicopter-like UAVs, it achieves faster cruise speed and has better autonomy, alongside dispensing the use of the complex swashplates; and compared to quadrotors again the cruise speed and autonomy factors appear, but the fact that it is more compact is also crucial for restrict space applications.

A birotor tiltrotor however is not a trivial system and presents control challenges which are not present in a quadrirotor system. It constitutes an underactuated mechanical system and is highly nonlinear and coupled.

Another challenge for the control design is that the tilting actuators that control the rotor angular position have much slower dynamics than the rotors. Also the tilting of a rotor produces a significant counter-torque in the rest of the body as the rotor mass is not negligible.

Besides the specific characteristics of the tiltrotor, a UAV is a complex system that motivates researches not only on control systems, but on several fields, e.g., sensor fusion and state estimation,

[5]Image taken from http://www.rotormast.com

computer vision, embedded systems performance and methodologies, wireless communication, and so on.

1.2 Project ProVANT

ProVANT is a Brazilian project within the Automation and Systems Department (DAS) of the Federal University of Santa Catarina (UFSC) and the Electronic Engineering Department (DELT) of the Federal University of Minas Gerais (UFMG).

Its goal is to develop from scratch a birotor tiltrotor UAV for academic research purposes. In fact ProVANT targets creating a UAV capable to perform autonomous flights according to a predefined trajectory while being remotely monitored. Research targeting using the UAV for load transportation is also being addressed. A computer aided design computer-aided design (CAD) image of the designed prototype is presented in figure 1.5 and a photo of the assembled prototype is shown in figure 1.6.

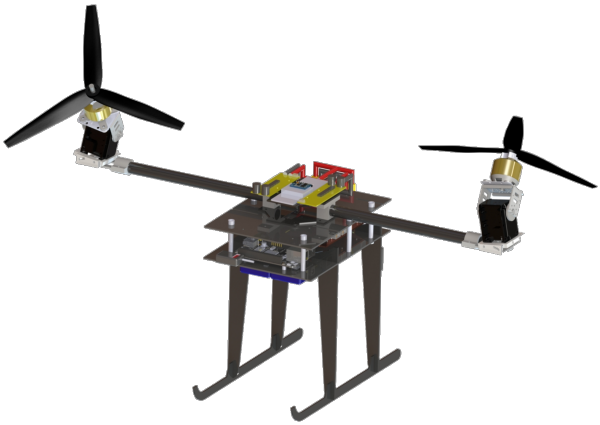


Figure 1.5 – Project ProVANT tiltrotor first prototype CAD.

The reason for creating a UAV comes from the fact that when purchasing ready-for-use aircrafts it is difficult to customize such devices, since several features are not documented or available for the final user to change. Therefore, ProVANT targets the complete aircraft de-

sign, including its electronic components, control software (including adopted algorithms), and most mechanical aspects.

The ProVANT project includes different fields of knowledge and has several members working to build its first prototype. In its context, three master degrees have already been concluded: two with emphasis in the embedded system (GONCALVES, 2014; BODANESE, 2014) and one in control techniques for load transportation (ALMEIDA, 2014), aside from several conference papers published. The study presented in this work is within the project's context and aims to develop control algorithms to perform autonomous flight. It is also the first work within the project's scope to present experimental results using the UAV under design. Nevertheless, such experiments are still preliminary, given that the UAV's Global Positioning System (GPS) is still under development, precluding the path tracking controllers experiments.

Figure 1.6 – Project ProVANT tiltrotor first assemble.

1.3 Objectives

The main objective of this work is to develop control laws that solve the hovering and path tracking problems for a small scale birotor tiltrotor UAV, with experimental flights applying the hovering controls.

The hovering problem consists in stabilizing the UAV's orientation and height, while path tracking is the ability of the aircraft to follow a predefined trajectory in the cartesian space.

This thesis' first objective is to develop a dynamic model for simulations and control design. In the literature, birotor tiltrotors have been analyzed with six degrees of freedom (DOFs) models that do not consider the tilting mechanism dynamics. As commented before, its dynamics are much slower than the other actuators and have a substantial influence on the system. In this work the model includes these dynamics, augmenting the model to eight DOFs.

The control laws design aim is to achieve linear optimal controllers that allows path tracking and hovering while presenting robustness in the presence of parametric uncertainties, external disturbances and unmodeled dynamics.

Finally the hovering control laws are implemented into the real tiltrotor from project ProVANT to perform the experimental flights. The path tracking controllers are not experimentally tested as the UAV had no functional positioning system at the time this work was concluded.

1.4 Outline

The structure of this Master thesis is organized as follows:

- **Chapter 2** presents the dynamic modeling of a birotor tiltrotor UAV using the Euler-Lagrange formulation. The system's generalized coordinates are defined and an eight DOFs system is obtained. The tiltrotor is considered as a multibody system to include the tilting mechanism dynamics and the coupling between the rotors motion and UAV's main body. Also considered is a displacement in the center of mass, a small fixed tilt in the UAVs structural arms and the counter torque produced by the propellers rotation. Lastly, a table with the parameters of project ProVANT's tiltrotor is given, which are the parameters used throughout this work.
- **Chapter 3** brings the control designs developed for this thesis. First a state-space representation and linearization of the non-

linear model presented in chapter 2 is performed. Then several path tracking controls are derived with different linear techniques: linear quadratic regulator (LQR), LQR for linear time varying system (LTV), LQR with a linear matrix inequality (LMI) formulation, $\mathcal{H}_\infty$ control with a LMI formulation, $\mathcal{H}_2/\mathcal{H}_\infty$ mixed control with a LMI formulation. The backstepping design proposed in (CHOWDHURY et al., 2012) is then briefly explained followed by a comparison of the controllers. The last part of the chapter develops a LQR for the hovering problem and compares it to an equivalent backstepping controller.

- **Chapter 4** explains how the system states are measured and estimated with the available sensors in project ProVANT's prototype. Then the experimental results performed with the hovering controllers, developed in chapter 3, are presented. The references used in the experiments are recorded and used in simulation to help in the results assessment.

- **Chapter 5** summarizes the thesis contributions and highlights its main conclusions. It also presents suggestions for future works.

1.5 List of Publications

During the course of the master's degree five conference papers have been published:

1. GONCALVES, F.; DONADEL, R.; BECKER, L.; RAFFO, G. Assessing the use of matlab simulink on the development process of an unmanned aerial vehicle. In: *Third Workshop on Design, Modeling and Evaluation of Cyber Physical Systems.* Philadelphia, USA: CyPhy, 2013.

2. GONCALVES, F.; BODANESE, J.; DONADEL, R.; RAFFO, G.; NORMEY-RICO, J.; BECKER, L. Small scale uav with birotor configuration. In: *Unmanned Aircraft Systems (ICUAS), 2013 International Conference on.* Atlanta, USA: ICUAS, 2013. p. 761–768.

3. DONADEL, R.; RAFFO G.V. BECKER, L. Modeling and control of a tiltrotor uav for path tracking. In: *Proceedings of the*

19th IFAC World Congress, 2014. Cape Town, South Africa: IFAC, 2014. p. 3839–3844.

4. DONADEL, R.; ALMEIDA, M.; RAFFO G.V. BECKER, L. Path tracking control of a small scale tiltrotor unmanned aerial vehicle. In: *Anais do XX Congresso Brasileiro de Automática*. CBA 2014. Belo Horizonte, Brasil: SBA, 2014. p. 1450–1457

5. ALMEIDA, M.; DONADEL, R.; RAFFO G.V. BECKER, L. Full control of a tiltrotor uav for load transportation. In: *Anais do XX Congresso Brasileiro de Automática*, CBA 2014. Belo Horizonte, Brazil: SBA, 2014. p. 2097–2104

Chapter 2

Tiltrotor UAV Modeling

This chapter presents the modeling of a birotor tiltrotor UAV. It aims to create a dynamic model that mathematically describes the relationship between the external forces applied to the UAV and the motion of its center of mass. According to the related works, the modeling of tiltrotors UAV's is typically done either through the Newton-Euler equations or through the Euler-Lagrange formulation.

The first small scale civil birotor tiltrotor is presented in (GRESS, 2002, 2004). It has two tilting rotors and a mechanical structure with a tail, as shown in figure 2.1. This work takes advantage of the gyroscopic effect of the tilting mechanism to help in the aircraft's torques, showing that the effect caused by the tilting of the rotors is significant.

In (KENDOUL et al., 2005) the author expands Gress' structure by deliberately displacing the center of mass in the z axis to achieve a better pitch moment and a six degree of freedom model is found using the Newton-Euler formulation. Here the existence of a multiplication between control inputs is first acknowledged and the author makes some simplifications to work around it.

The work of (SANCHEZ et al., 2008) brings a mechanical structure without the tail from the two previous works. Nevertheless the model obtained using Newton-Euler equations is similar, except for its external force vector.

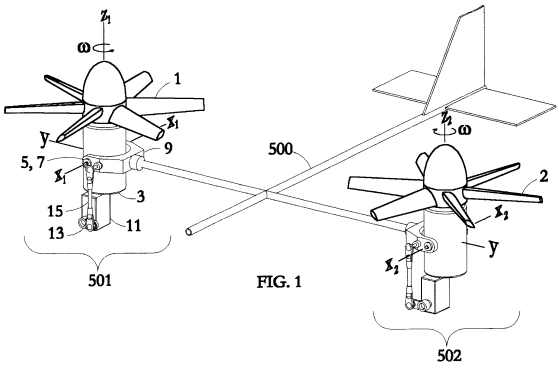


Figure 2.1 – Gress mechanism as presented in US Patent 6,719,244.

Some other subsequent works are (PAPACHRISTOS et al., 2011a, 2011b; JANSEN; RAMIREZ-SERRANO, 2011; CHOWDHURY et al., 2012), which have models obtained by Newton-Euler and Euler-Lagrange formulations. The models vary in structure depending on the formulation used but, aside the external forces, the models found in literature are basically the model of a single rigid body in space and are actually similar to most quadrotors works (e.g., (RAFFO et al., 2008),(CASTILLO et al., 2005)). The models differ with respect to its external forces, but from a control design viewpoint they all suffer from a fundamental drawback: a nonlinear coupling between the control inputs.

The exception would be the work of (NOTARSTEFANO; HAUSER, 2010) that has a mechanical structure composed by a single rotor located in the middle of the aircraft's body with three tilting mechanisms. As the system is highly dependent on the performance of the servomotors that tilt the rotor, its dynamics were included in the model. However this system is quite different from the birotor tiltrotor that we will model and its results cannot be directly used.

In the present work the tiltrotor is modeled as a multibody system. The reasons for such are threefold:

- Includes the servomotors dynamics. This is important because its dynamics is much slower than that of the brushless motors, and slow enough to have a significant impact when controlling the system.

- Includes the coupling between the UAV's main body and its tilting bodies, as well as the gyroscopic effect of the latter. These bodies have around 15% of the mass of the central body, meaning that its rotation has a substantial effect on the system.
- Solves the nonlinear coupling of the control inputs in the external forces equations.

The rigid body kinematics and dynamics theory used throughout this chapter is mainly based on (SPONG et al., 2005), while the theory on multibody systems is based on (SHABANA, 2010).

2.1 Rigid body in R^3 space

A rigid body is a body that does not suffer deformations. Mathematically, it means that the distance between any two given points rigidly attached to the body is constant. For this work the position of rigid bodies, as well as its motion, is considered in the R^3.

The position of a body can only be numerically described when in relation to a reference. For this section a generic rigid body is described by a Cartesian coordinate frame $\mathfrak{B} = \{\vec{X}_{\mathfrak{B}}, \vec{Y}_{\mathfrak{B}}, \vec{Z}_{\mathfrak{B}}\}$, whilst its position is related to a fixed inertial reference frame $\mathfrak{I} = \{\vec{X}_{\mathfrak{I}}, \vec{Y}_{\mathfrak{I}}, \vec{Z}_{\mathfrak{I}}\}$.

For a rigid body in R^3 space, the position can be represented with its linear and angular positions, each with three degrees of freedom. They are henceforth called position and orientation, respectively.

The position is expressed as the distance between the origin of the frames, here defined as:

$$\boldsymbol{\xi}(t) = \begin{bmatrix} x(t) & y(t) & z(t) \end{bmatrix}' \quad , \tag{2.1}$$

where the superscript $'$ denotes the transpose operation.

There are different ways to represent the orientation of a rigid body in space. In this work Euler angles are used. It defines three angles, each one representing a rotation around an axis of a coordinate frame, called elemental rotations. The three elemental rotations executed in sequence express the orientation of the body frame $\mathfrak{B}$ in relation to the reference frame $\mathfrak{I}$. Intrinsic rotations are used so the

rotations take place about a rotating coordinate system, which is initially aligned with the inertial frame, and modifies its orientation after each elemental rotation, i.e., each new rotation takes place around the resulting frame of the last rotation.

Twelve distinct sequences of elemental rotations are possible. In this work the adopted convention is that the first rotation is around the $\vec{Z}$ axis (yaw), followed by a rotation around the $\vec{Y}$ axis (pitch), and finally a rotation around $\vec{X}$ axis (roll). From here on the roll angle will be represented by ϕ, the pitch by θ and the yaw by ψ. These rotations lead frame $\mathfrak{I}$ to $\mathfrak{B}$ and are depicted in figure 2.2.

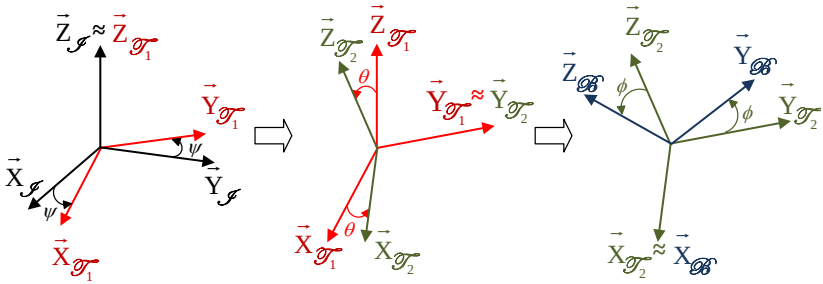


Figure 2.2 – Sequence of rotations convention adopted.

It is important to note that only the first rotation (yaw) is based on the inertial frame $\mathfrak{I}$. Since each new rotation is made considering the current frame, the pitch angle represents a rotation around $\vec{Y}$ axis of the intermediate frame $\mathfrak{T}_1$, and roll a rotation of the $\vec{X}$ axis of the intermediate frame $\mathfrak{T}_2$.

The rigid body orientation is then defined as:

$$\boldsymbol{\eta}(t) = \begin{bmatrix} \phi(t) & \theta(t) & \psi(t) \end{bmatrix}' \quad . \tag{2.2}$$

A rotation in Euclidean space is a linear transformation and can be represented with the so-called rotation matrix, which can be used to describe an elemental rotation. Consider the roll rotation, i.e., rotation around the $\vec{X}$ axis by a ϕ angle. Adopting the right-hand rule, the relation between the frames can be geometrically determined as in

figure 2.3. The rotation matrix is then:

$$\boldsymbol{R}(\vec{X},\phi) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & cos(\phi) & sin(\phi) \\ 0 & -sin(\phi) & cos(\phi) \end{bmatrix} \quad .$$

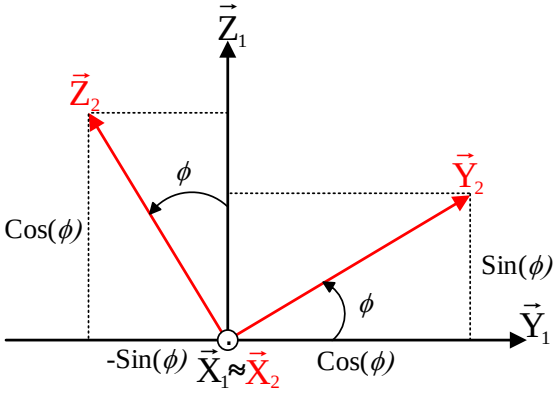


Figure 2.3 – Elemental matrix for a rotation around the $\vec{X}$ axis.

Likewise for the pitch and yaw rotation:

$$\boldsymbol{R}(\vec{Y},\theta) = \begin{bmatrix} cos(\theta) & 0 & -sin(\theta) \\ 0 & 1 & 0 \\ sin(\theta) & 0 & cos(\theta) \end{bmatrix}$$

$$\boldsymbol{R}(\vec{Z},\psi) = \begin{bmatrix} cos(\psi) & sin(\psi) & 0 \\ -sin(\psi) & cos(\psi) & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad .$$

The three elemental rotations can be combined into a single transformation:

$$\boldsymbol{R}_{\mathfrak{I}}^{\mathfrak{B}} = \boldsymbol{R}(\vec{X}_{\mathfrak{T}_2},\phi)\boldsymbol{R}(\vec{Y}_{\mathfrak{T}_1},\theta)\boldsymbol{R}(\vec{Z}_{\mathfrak{I}},\psi) = \boldsymbol{R}(\phi,\theta,\psi) = \begin{bmatrix} C\psi C\theta & S\psi C\theta & -S\theta \\ C\psi S\theta S\phi - S\psi C\phi & S\psi S\theta S\phi + C\psi C\phi & C\theta S\phi \\ C\psi S\theta C\phi + S\psi S\phi & S\psi S\theta C\phi - C\psi S\phi & C\theta C\phi \end{bmatrix} , \tag{2.3}$$

$C(.)$ being the cosine and $S(.)$ the sine.

Equation (2.3) can also be used as a transformation matrix relating the coordinates of a point p between the two frames, as in

$$\boldsymbol{p}^{\mathfrak{B}} = \boldsymbol{R}_{\mathfrak{J}}^{\mathfrak{B}} \boldsymbol{p}^{\mathfrak{J}} \quad . \tag{2.4}$$

Therefore, it is also important to find the inverse operation, that is, the rotation matrix from the body frame $\mathfrak{B}$ to the inertial frame $\mathfrak{J}$. Rotation matrices belong to the special orthogonal group, in our case of order 3, denoted by SO(3). Consequently, the following property applies:

$$\boldsymbol{R}^{-1} = \boldsymbol{R}' \quad . \tag{2.5}$$

Thus

$$\boldsymbol{R}_{\mathfrak{B}}^{\mathfrak{J}} = (\boldsymbol{R}_{\mathfrak{J}}^{\mathfrak{B}})' = \begin{bmatrix} C\psi C\theta & C\psi S\theta S\phi - S\psi C\phi & C\psi S\theta C\phi + S\psi S\phi \\ S\psi C\theta & S\psi S\theta S\phi + C\psi C\phi & S\psi S\theta C\phi - C\psi S\phi \\ -S\theta & C\theta S\phi & C\theta C\phi \end{bmatrix} \quad . \tag{2.6}$$

Equation (2.4) represents a pure rotation between the inertial and the body frame, depicted in figure 2.4a. The general case is a combination of pure rotation and pure translation, also known as rigid motion, as in figure 2.4b. Assume the values for point p are known in relation to the body frame $\mathfrak{B}$. To express it in terms of the inertial frame:

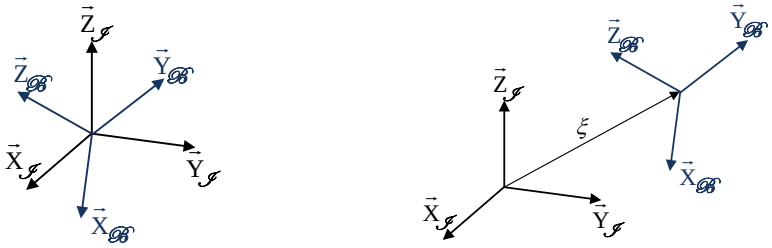
$$\boldsymbol{p}^{\mathfrak{J}} = \boldsymbol{R}_{\mathfrak{B}}^{\mathfrak{J}} \boldsymbol{p}^{\mathfrak{B}} + \boldsymbol{\xi} \quad . \tag{2.7}$$

This kind of relation will be important later in this chapter, when a particle of the body is considered as a point to determine the system's energy.

2.2 Rigid body Kinematics

Kinematics describes motions disregarding the forces causing them. Nevertheless, its definition is essential for the dynamic modeling.

The velocity of a single particle undergoing rotation and translation can be found by differentiating equation (2.7) with respect to



(a) Pure rotation **(b)** Rotation and translation

Figure 2.4 – Representation of a rigid body and its frames $\mathfrak{I}$ and $\mathfrak{B}$ in the R^3 space.

time:

$$\boldsymbol{v}^{\mathfrak{I}} = \dot{\boldsymbol{p}}^{\mathfrak{I}} = \dot{\boldsymbol{R}}^{\mathfrak{I}}_{\mathfrak{B}} \boldsymbol{p}^{\mathfrak{B}} + \boldsymbol{R}^{\mathfrak{I}}_{\mathfrak{B}} \dot{\boldsymbol{p}}^{\mathfrak{B}} + \dot{\boldsymbol{\xi}} \quad . \tag{2.8}$$

As the body is assumed rigid, all particles will have the same velocity and equation (2.8) corresponds to the velocity of the rigid body.

Note however that equation (2.8) requires the time derivative of the rotation matrix. A equivalent, simpler expression for $\dot{R}$ can be found. Let us start with the matrix identity

$$(\boldsymbol{R}^{\mathfrak{I}}_{\mathfrak{B}})^{-1} \boldsymbol{R}^{\mathfrak{I}}_{\mathfrak{B}} = \mathbb{I} \quad .$$

Using equation (2.5) it becomes

$$(\boldsymbol{R}^{\mathfrak{I}}_{\mathfrak{B}})' \boldsymbol{R}^{\mathfrak{I}}_{\mathfrak{B}} = \mathbb{I} \quad .$$

Differentiating this equation in relation to time:

$$(\dot{\boldsymbol{R}}^{\mathfrak{I}}_{\mathfrak{B}})' \boldsymbol{R}^{\mathfrak{I}}_{\mathfrak{B}} + (\boldsymbol{R}^{\mathfrak{I}}_{\mathfrak{B}})' \dot{\boldsymbol{R}}^{\mathfrak{I}}_{\mathfrak{B}} = 0 \quad . \tag{2.9}$$

Defining the matrix

$$\boldsymbol{S} = (\boldsymbol{R}^{\mathfrak{I}}_{\mathfrak{B}})' \dot{\boldsymbol{R}}^{\mathfrak{I}}_{\mathfrak{B}} \tag{2.10}$$

and substituting it in equation (2.9), one obtains:

$$\boldsymbol{S}' + \boldsymbol{S} = 0 \quad .$$

This relation is the definition of skew-symmetric matrices. Therefore,

matrix $\boldsymbol{S}$ is skew-symmetric and, as $\boldsymbol{S} \in R^{3\times3}$, has only three independent elements, which is represented by a three dimensional vector (see Appendix A.1).

Defining $\boldsymbol{w}_{\mathfrak{B}\mathfrak{I}}^{\mathfrak{B}} = \begin{bmatrix} p & q & r \end{bmatrix}'$ as the angular velocity of the body frame in relation to the inertial frame represented in the body frame, $\boldsymbol{S}$ can be defined as (SPONG et al., 2005)

$$\boldsymbol{S}(\boldsymbol{w}_{\mathfrak{B}\mathfrak{I}}^{\mathfrak{B}}) = \begin{bmatrix} 0 & -r & q \\ r & 0 & -p \\ -q & p & 0 \end{bmatrix} \quad . \tag{2.11}$$

From equations (2.10) and (2.11):

$$\dot{\boldsymbol{R}}_{\mathfrak{B}}^{\mathfrak{I}} = \boldsymbol{R}_{\mathfrak{B}}^{\mathfrak{I}} \boldsymbol{S}(\boldsymbol{w}_{\mathfrak{B}\mathfrak{I}}^{\mathfrak{B}}) \quad . \tag{2.12}$$

Considering the angular velocity of a single particle of the body, represented by a point p rigidly attached to the body, and a situation as in figure 2.4a (pure rotation), its position in the inertial frame is:

$$\boldsymbol{p}^{\mathfrak{I}} = \boldsymbol{R}_{\mathfrak{B}}^{\mathfrak{I}} \boldsymbol{p}^{\mathfrak{B}} \quad .$$

Differentiating in relation to time:

$$\dot{\boldsymbol{p}}^{\mathfrak{I}} = \dot{\boldsymbol{R}}_{\mathfrak{B}}^{\mathfrak{I}} \boldsymbol{p}^{\mathfrak{B}} + \boldsymbol{R}_{\mathfrak{B}}^{\mathfrak{I}} \dot{\boldsymbol{p}}^{\mathfrak{B}} \quad .$$

As point p is rigidly attached to the body frame, $\dot{\boldsymbol{p}}^{B} = 0$. Using equation (2.12) yields:

$$\dot{\boldsymbol{p}}^{\mathfrak{I}} = \boldsymbol{R}_{\mathfrak{B}}^{\mathfrak{I}} \boldsymbol{S}(\boldsymbol{w}_{\mathfrak{B}\mathfrak{I}}^{\mathfrak{B}}) \boldsymbol{p}^{\mathfrak{B}} \quad .$$

The rotation matrix can again be used to transform the angular velocity between frames:

$$\boldsymbol{w}_{\mathfrak{B}\mathfrak{I}}^{\mathfrak{I}} = \boldsymbol{R}_{\mathfrak{B}}^{\mathfrak{I}} \boldsymbol{w}_{\mathfrak{B}\mathfrak{I}}^{\mathfrak{B}} \quad .$$

It is important to note that the angular velocity $\boldsymbol{w}_{\mathfrak{B}\mathfrak{I}}^{\mathfrak{B}}$ does not have the same meaning of the Euler angles rate ($\dot{\boldsymbol{\eta}} = \begin{bmatrix} \dot{\phi} & \theta & \dot{\psi} \end{bmatrix}$). As represented in figure 2.2, the Euler angles are defined in relation not only to the inertial and body frames, but also to the intermediary frames $\mathfrak{T}_1$ and $\mathfrak{T}_2$. Therefore the angular velocity is related to the Euler

angular rates as:

$$\begin{bmatrix} p \\ q \\ r \end{bmatrix} = \begin{bmatrix} \dot{\phi} \\ 0 \\ 0 \end{bmatrix} + \boldsymbol{R}_{\mathfrak{B}}^{\mathfrak{T}_2}(\vec{X}, \phi) \begin{bmatrix} 0 \\ \dot{\theta} \\ 0 \end{bmatrix} + \boldsymbol{R}_{\mathfrak{T}_2}^{\mathfrak{T}_1}(\vec{Y}, \theta) \boldsymbol{R}_{\mathfrak{B}}^{\mathfrak{T}_2}(\vec{X}, \phi) \begin{bmatrix} 0 \\ 0 \\ \dot{\psi} \end{bmatrix} \quad . \tag{2.13}$$

Denominating $\boldsymbol{W_\eta}$ as the transformation matrix between the angular velocity and the Euler angular rates as in

$$\boldsymbol{w}_{\mathfrak{B}\mathfrak{I}}^{\mathfrak{B}} = \boldsymbol{W_\eta} \dot{\boldsymbol{\eta}} \quad , \tag{2.14}$$

then by developing equation (2.13) results that

$$\boldsymbol{W_\eta} = \begin{bmatrix} 1 & 0 & -sin(\theta) \\ 0 & cos(\phi) & sin(\phi)cos(\theta) \\ 0 & -sin(\phi) & cos(\phi)cos(\theta) \end{bmatrix}. \tag{2.15}$$

The Euler angular rates can be obtained with $\dot{\boldsymbol{\eta}} = \boldsymbol{W_\eta}^{-1} \boldsymbol{w}_{\mathfrak{B}\mathfrak{I}}^{\mathfrak{B}}$, where:

$$\boldsymbol{W_\eta}^{-1} = \begin{bmatrix} 1 & sin(\phi)tan(\theta) & cos(\phi)tan(\theta) \\ 0 & cos(\phi) & -sin(\phi) \\ 0 & sin(\phi)sec(\theta) & cos(\phi)sec(\theta) \end{bmatrix} \quad . \tag{2.16}$$

This matrix presents a singularity at $\theta = 90°$ and $\theta = 270°$, which is a limitation of our approach. This could be avoided by using other representation for the rigid body orientation such as quaternions. However, in this work Euler angles are chosen in order to avoid the increase in complexity, as quaternions add one more variable to the system.

Finally, equation (2.8) can be used with equation (2.12) and yield the rigid body velocity:

$$\boldsymbol{v}^{\mathfrak{I}} = \boldsymbol{R}_{\mathfrak{B}}^{\mathfrak{I}} \boldsymbol{S}(\boldsymbol{w}_{\mathfrak{B}\mathfrak{I}}^{\mathfrak{B}}) \boldsymbol{p}^{\mathfrak{B}} + \dot{\boldsymbol{\xi}} \quad . \tag{2.17}$$

2.3 Tiltrotor UAV Structure and Generalized Coordinates

The UAV modeled in this work is a birotor with a tiltrotor configuration, like the models discussed in the introduction of this chapter. As the prototype for experimental flights is being built as part of the project, some additions to the aircraft's physical structure could be done in order to enhance the controllability of the model.

The first addition to the UAV's structure, and therefore to the model, is the inclusion of a fixed lateral tilt. By doing so, even if there is no actuator to alter the lateral tilt of the rotors, controllability of the lateral axis is still achieved (RAFFO et al., 2011).

The inclusion of this fixed angle has the counter-effect of decreasing the maximum vertical thrust the UAV can perform, which in turn decreases the total lift force of the aircraft. Even so, it was judged that by using a small angle the gain in controllability is worth the loss.

Also, a displacement of the UAV's center of mass in relation to its center of rotation was intentionally sought during the mechanical assembly. Apart from adding to the composition of a more complete simulation environment, it is actually vital for a tiltrotor to have a displacement of its center of mass along the $Z^{\mathfrak{B}}$ axis to improve the pitch torque (τ_θ). Furthermore, a displacement in $X^{\mathfrak{B}}$ is important for controllability along this axis. The reasons for such are shown later during the derivation of the external forces equations.

In order to include the servomotors dynamics in the model, the UAV is considered as a multibody system composed of three rigid bodies: the two rotors and the main body, the former composed of the brushless motors and the propellers while the latter encompasses the carbon-fiber structure, the landing gear, the servomotors, the batteries, and all other electronic devices. Thus, each of the two rotors is interconnected to the main body by a revolute joint.

Figure 2.5 shows the frames and variables definitions. Frame $\mathfrak{I}$ is the inertial frame; $\mathfrak{C}_1$, $\mathfrak{C}_2$ and $\mathfrak{C}_3$ are frames rigidly attached to the center of mass of each body; and $\mathfrak{B}$ is the moving body frame rigidly attached to the main body center of rotation. Note that it is assumed that the rotors' center of rotation coincides with the center of mass.

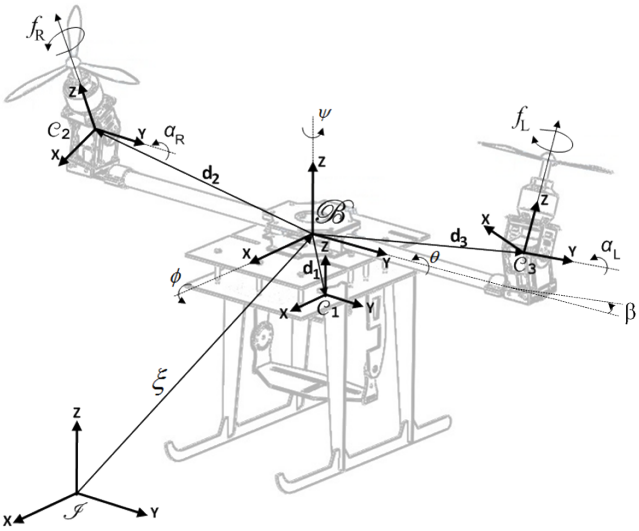


Figure 2.5 – Tiltrotor UAV frames and variables definitions.

Variables f_R and f_L represents the thrust forces and β the fixed lateral tilt angle of the rotors. Vector $\boldsymbol{d_1} = \begin{bmatrix} d_{1x} & d_{1y} & d_{1z} \end{bmatrix}'$ is the distance from frame $\mathfrak{B}$ to frame $\mathfrak{C}_1$ and represents the center of mass displacement of the main body, while $\boldsymbol{d_2} = \begin{bmatrix} d_{2x} & d_{2y} & d_{2z} \end{bmatrix}'$ and $\boldsymbol{d_3} = \begin{bmatrix} d_{3x} & d_{3y} & d_{3z} \end{bmatrix}'$ are the distances between frame $\mathfrak{B}$ and frames $\mathfrak{C}_2$ and $\mathfrak{C}_3$, respectively.

The translation of the body frame in relation to the inertial frame is represented by $\boldsymbol{\xi} = \begin{bmatrix} x & y & z \end{bmatrix}'$ and the attitude by $\boldsymbol{\eta} = \begin{bmatrix} \phi & \theta & \psi \end{bmatrix}'$, as defined in section 2.1.

Even though the main body has six degrees of freedom, each of the rotors have only one. That is because the mechanical assembly (revolute joint) imposes five constraint equations to each of the rotors. The rotation of frame $\mathfrak{C}_2$ in relation to frame $\mathfrak{B}$ is depicted by variable α_R and describes the rotation of the right rotor. Likewise, α_L describes the rotation of the left rotor in relation to the main body.

To find the system's degrees of freedom, we can use the Kutzbach criterion (SHABANA, 2010):

$$DOF = 6.nb - nc = 6.3 - 10 = 8 \quad , \tag{2.18}$$

where nb is the number of bodies and nc is the number of independent constraint equations. A revolute joint, by definition, have a single-axis rotation. Therefore each of them have five constraint equations each as it cannot move in x, y or z and can only rotate around one axis. As we have three bodies and five constraint equations for each revolute joint, the number of degrees of freedom for our multibody system is eight.

Also each brushless motor could be considered as another body with a revolute joint, adding two more degrees of freedom to the system. However, as its dynamics are much faster than the system's, it was decided to discard such assumption and only include the counter torque it produces in the external force equations, avoiding an increase in complexity.

An important definition in mechanical systems modeling is the concept of generalized coordinates. It is a set of variables that can unambiguously specify the position of the system, the minimum number of variables for such being the degrees of freedom of the system (GINSBERG, 2008).

Thereby, the tiltrotor's generalized coordinates are

$$\begin{aligned} \boldsymbol{q} &= \begin{bmatrix} x(t) & y(t) & z(t) & \phi(t) & \theta(t) & \psi(t) & \alpha_R(t) & \alpha_L(t) \end{bmatrix}' \\ &= \begin{bmatrix} \boldsymbol{\xi}'(t) & \boldsymbol{\eta}'(t) & \alpha_R(t) & \alpha_L(t) \end{bmatrix}' \quad . \end{aligned} \tag{2.19}$$

2.4 Euler-Lagrange Formulation

In this section the UAV's equations of motion are derived using the Euler-Lagrange formulation. Classic mechanical systems, such a rigid body system, can be described by the Euler-Lagrange equation

$$\frac{d}{dt}\left(\frac{\partial L(\boldsymbol{q}, \dot{\boldsymbol{q}})}{\partial \dot{\boldsymbol{q}}}\right) - \frac{\partial L(\boldsymbol{q}, \dot{\boldsymbol{q}})}{\partial \boldsymbol{q}} = \boldsymbol{F}(\boldsymbol{q}) \quad , \tag{2.20}$$

where $\boldsymbol{F}(\boldsymbol{q})$ is the external force vector and

$$L(\boldsymbol{q}, \dot{\boldsymbol{q}}) = K(\boldsymbol{q}, \dot{\boldsymbol{q}}) - P(\boldsymbol{q}, \dot{\boldsymbol{q}}) \tag{2.21}$$

is the Lagrangian of the system, K being the kinetic energy and P the potential energy, all in relation to the generalized coordinates.

To find equation (2.20) the following sections derive the system's kinetic and potential energy as well as the external force vector.

For mathematical purposes, the index $i = \{1, 2, 3\}$ is used to refer to different bodies: 1 for the main central body, 2 the left rotor, and the 3 for right rotor.

Also a point $\boldsymbol{p_1}$ rigidly attached to frame $\mathfrak{C}_1$ is defined to represent a particle of body 1; $\boldsymbol{p_2}$ rigidly attached to frame $\mathfrak{C}_2$ to represent a particle of body 2; and $\boldsymbol{p_3}$ rigidly attached to frame $\mathfrak{C}_3$ to represent a particle of body 3.

Lastly, from the frames definitions in figure 2.5, the rotation matrices are:

$$\boldsymbol{R}^{\mathfrak{B}}_{\mathfrak{C}_2} = \begin{bmatrix} C\alpha_R & 0 & S\alpha_R \\ -S\alpha_R S\beta & C\beta & C\alpha_R S\beta \\ -S\alpha_R C\beta & -S\beta & C\alpha_R C\beta \end{bmatrix} \tag{2.22}$$

$$\boldsymbol{R}^{\mathfrak{B}}_{\mathfrak{C}_3} = \begin{bmatrix} C\alpha_L & 0 & S\alpha_L \\ S\alpha_L S\beta & C\beta & -C\alpha_L S\beta \\ -S\alpha_L C\beta & S\beta & C\alpha_L C\beta \end{bmatrix} \quad , \tag{2.23}$$

and $\boldsymbol{R}^{\mathfrak{B}}_{\mathfrak{C}_1} = \mathbb{I}_{3X3}$ because both the body frame and $\mathfrak{C}_1$ are rigidly attached to body 1.

2.4.1 Potential energy

The potential energy of the system can be found as (SHABANA, 2010):

$$P = \sum_{i=1}^{3} P_i \quad , \tag{2.24}$$

where

$$P_i = -\int_{V_i} \rho_i \boldsymbol{g}' \boldsymbol{p}_i^{\mathfrak{I}} dV_i \tag{2.25}$$

is the volume integral with ρ_i as the mass density of body i, V_i the volume of body i and $\boldsymbol{g} = \begin{bmatrix} 0 & 0 & -g \end{bmatrix}'$. In equation (2.25), the negative sign is a consequence of the explicit negative gravity acceleration in vector $\boldsymbol{g}$, as in (LEWIS et al., 2003, p. 123).

Equation (2.7) links $\boldsymbol{p_i^{\mathfrak{I}}}$ to $\boldsymbol{p_i^{\mathfrak{B}}}$. Using the same logic to relate

$\boldsymbol{p_i^{\mathfrak{B}}}$ and $\boldsymbol{p_i^{\mathfrak{C}_i}}$:

$$\boldsymbol{p_i^{\mathfrak{B}}} = \boldsymbol{R_{\mathfrak{C}_i}^{\mathfrak{B}}} \boldsymbol{p_i^{\mathfrak{C}_i}} + \boldsymbol{d_i} \quad . \tag{2.26}$$

Substituting equation (2.26) into (2.7):

$$\boldsymbol{p_i^{\mathfrak{I}}} = \boldsymbol{R_{\mathfrak{B}}^{\mathfrak{I}}}(\boldsymbol{R_{\mathfrak{C}_i}^{\mathfrak{B}}} \boldsymbol{p_i^{\mathfrak{C}_i}} + \boldsymbol{d_i}) + \boldsymbol{\xi} \quad . \tag{2.27}$$

It relates each of the bodies center of mass frame with the inertial frame.

Using equation (2.27) in equation (2.25):

$$P_i = -\boldsymbol{g'} \int_{V_i} \rho_i [\boldsymbol{R_{\mathfrak{C}_i}^{\mathfrak{I}}} \boldsymbol{p_i^{\mathfrak{C}_i}} + \boldsymbol{R_{\mathfrak{B}}^{\mathfrak{I}}} \boldsymbol{d_i} + \boldsymbol{\xi}] dx_{C_i} dy_{C_i} dz_{C_i} \quad . \tag{2.28}$$

The mass of the body is given by

$$m_i = \int_{V_i} \rho_i dx_{C_i} dy_{C_i} dz_{C_i} \quad , \tag{2.29}$$

and assuming that all bodies are symmetric:

$$\int_{V_i} \rho_i \boldsymbol{p_i^{\mathfrak{C}_i}} dx_{C_i} dy_{C_i} dz_{C_i} = 0 \quad . \tag{2.30}$$

Developing equation (2.28) with equations (2.29) and (2.30):

$$P_i = -\boldsymbol{g'}(m_i \boldsymbol{R_{\mathfrak{B}}^{\mathfrak{I}}} \boldsymbol{d_i} + m_i \boldsymbol{\xi}) \quad . \tag{2.31}$$

Thus, using equation (2.24), the potential energy of the system is

$$P = -\boldsymbol{g'} \boldsymbol{R_{\mathfrak{B}}^{\mathfrak{I}}}(m_1 \boldsymbol{d_1} + m_2 \boldsymbol{d_2} + m_3 \boldsymbol{d_3}) - \boldsymbol{g'} m \boldsymbol{\xi} \quad , \tag{2.32}$$

where m is the total mass of the system and is given by

$$m = m_1 + m_2 + m_3 \quad . \tag{2.33}$$

2.4.2 Kinetic energy

The kinetic energy of the whole system can be found with equation (SHABANA, 2010):

$$K = \sum_{i=0}^{n} K_i \quad , \tag{2.34}$$

where

$$K_i = \frac{1}{2} \int_{V_i} \rho_i (\boldsymbol{v_i^{\mathfrak{I}}})' (\boldsymbol{v_i^{\mathfrak{I}}}) dV_i \quad , \tag{2.35}$$

and $v_i^{\mathfrak{I}}$ is the velocity of a single particle of the body in relation to the inertial frame. These velocities are given by the time derivation of the position equations (2.27):

$$\dot{\boldsymbol{p}}_{\boldsymbol{i}}^{\mathfrak{I}} = \dot{\boldsymbol{\xi}} + \dot{\boldsymbol{R}}_{\mathfrak{B}}^{\mathfrak{I}} (\boldsymbol{R}_{\mathfrak{C}_i}^{\mathfrak{B}} \boldsymbol{p}_{\boldsymbol{i}}^{\mathfrak{C}_i} + \boldsymbol{d_i}) + \boldsymbol{R}_{\mathfrak{B}}^{\mathfrak{I}} (\dot{\boldsymbol{R}}_{\mathfrak{C}_i}^{\mathfrak{B}} \boldsymbol{p}_{\boldsymbol{i}}^{\mathfrak{C}_i} + \boldsymbol{R}_{\mathfrak{C}_i}^{\mathfrak{B}} \dot{\boldsymbol{p}}_{\boldsymbol{i}}^{\mathfrak{C}_i} + \dot{\boldsymbol{d}}_{\boldsymbol{i}}) \quad . \tag{2.36}$$

As point $\boldsymbol{p_1}$ is rigidly attached to frame $\mathfrak{C}_1$, there is no relative motion between them and $\dot{\boldsymbol{p}}_{\mathbf{1}}^{\mathfrak{C}_1} = 0$. The same can be deduced for the other two points and we have that $\dot{\boldsymbol{p}}_{\boldsymbol{i}}^{\mathfrak{C}_i} = 0$. Furthermore, the distances $\boldsymbol{d_i}$ are also constant and $\dot{\boldsymbol{d}}_{\boldsymbol{i}} = 0$. With these definitions and using the properties of skew-symmetric matrices, the velocities became:

$$\dot{\boldsymbol{p}}_{\boldsymbol{i}}^{\mathfrak{I}} = \dot{\boldsymbol{\xi}} + \boldsymbol{R}_{\mathfrak{B}}^{\mathfrak{I}} \boldsymbol{w}_{\mathfrak{B}\mathfrak{I}}^{\mathfrak{B}} \times \boldsymbol{R}_{\mathfrak{C}_i}^{\mathfrak{B}} \boldsymbol{p}_{\boldsymbol{i}}^{\mathfrak{C}_i} + \boldsymbol{R}_{\mathfrak{B}}^{\mathfrak{I}} \boldsymbol{w}_{\mathfrak{B}\mathfrak{I}}^{\mathfrak{B}} \times \boldsymbol{d_i} + \boldsymbol{R}_{\mathfrak{C}_i}^{\mathfrak{I}} \boldsymbol{w}_{\mathfrak{C}_i\mathfrak{B}}^{\mathfrak{C}_i} \times \boldsymbol{p}_{\boldsymbol{i}}^{\mathfrak{C}_i} \quad . \tag{2.37}$$

Substituting equation (2.37) into (2.35), using equations (2.29), (2.30) and skew-symmetric matrix properties (appendix A.1), the kinetic energy for body i becomes:

$$\begin{aligned} K_i = & -m_i \dot{\boldsymbol{\xi}}' \boldsymbol{R}_{\mathfrak{B}}^{\mathfrak{I}} \boldsymbol{S}(\boldsymbol{d_i}) \boldsymbol{w}_{\mathfrak{B}\mathfrak{I}}^{\mathfrak{B}} + \frac{1}{2} (\boldsymbol{w}_{\mathfrak{C}_i\mathfrak{B}}^{\mathfrak{C}_i})' \boldsymbol{I_i} \boldsymbol{w}_{\mathfrak{C}_i\mathfrak{B}}^{\mathfrak{C}_i} \\ & + (\boldsymbol{w}_{\mathfrak{C}_i\mathfrak{B}}^{\mathfrak{C}_i})' \boldsymbol{I_i} \boldsymbol{w}_{\mathfrak{B}\mathfrak{I}}^{\mathfrak{B}} + \frac{1}{2} (\boldsymbol{w}_{\mathfrak{B}\mathfrak{I}}^{\mathfrak{B}})' \boldsymbol{J_i} \boldsymbol{w}_{\mathfrak{B}\mathfrak{I}}^{\mathfrak{B}} + \frac{1}{2} m_i \dot{\boldsymbol{\xi}}' \dot{\boldsymbol{\xi}} \quad , \end{aligned} \tag{2.38}$$

where

$$\boldsymbol{I_i} = \int \boldsymbol{S}(\boldsymbol{p_i})' \boldsymbol{S}(\boldsymbol{p_i}) dm = \begin{bmatrix} I_{xx} & I_{xy} & I_{xz} \\ I_{xy} & I_{yy} & I_{yz} \\ I_{xz} & I_{yz} & I_{zz} \end{bmatrix} \tag{2.39}$$

is the inertia tensor and

$$\boldsymbol{J_i} = (\boldsymbol{R}_{\mathfrak{C}_i}^{\mathfrak{B}})' \boldsymbol{I_i} \boldsymbol{R}_{\mathfrak{C}_i}^{\mathfrak{B}} + m_i \boldsymbol{S}(\boldsymbol{d_i})' \boldsymbol{S}(\boldsymbol{d_i}) \tag{2.40}$$

is the displaced inertia tensor, i.e., the inertia tensor for a rotation around an axis displaced by a distance $\boldsymbol{d_i}$ (Steiner's theorem).

Taking into account that there is no relative movement be-

tween frame $\mathfrak{B}$ and frame $\mathfrak{C}_1$, then $\boldsymbol{w}_{\mathfrak{C}_1\mathfrak{B}}^{\mathfrak{C}_1} = 0$. At last the total kinetic energy is obtained:

$$\begin{aligned} K =& \frac{1}{2} m \dot{\boldsymbol{\xi}}' \dot{\boldsymbol{\xi}} - \dot{\boldsymbol{\xi}}' \boldsymbol{R}_{\mathfrak{B}}^{\mathfrak{I}} \boldsymbol{H} \boldsymbol{w}_{\mathfrak{B}\mathfrak{I}}^{\mathfrak{B}} + \frac{1}{2} (\boldsymbol{w}_{\mathfrak{B}\mathfrak{I}}^{\mathfrak{B}})' \boldsymbol{J} \boldsymbol{w}_{\mathfrak{B}\mathfrak{I}}^{\mathfrak{B}} \\ &+ (\boldsymbol{w}_{\mathfrak{C}_2\mathfrak{B}}^{\mathfrak{C}_2})' \boldsymbol{I_2} \boldsymbol{w}_{\mathfrak{B}\mathfrak{I}}^{\mathfrak{B}} + (\boldsymbol{w}_{\mathfrak{C}_2\mathfrak{B}}^{\mathfrak{C}_2})' \boldsymbol{I_2} \boldsymbol{w}_{\mathfrak{C}_2\mathfrak{B}}^{\mathfrak{C}_2} \\ &+ (\boldsymbol{w}_{\mathfrak{C}_3\mathfrak{B}}^{\mathfrak{C}_3})' \boldsymbol{I_3} \boldsymbol{w}_{\mathfrak{B}\mathfrak{I}}^{\mathfrak{B}} + (\boldsymbol{w}_{\mathfrak{C}_3\mathfrak{B}}^{\mathfrak{C}_3})' \boldsymbol{I_3} \boldsymbol{w}_{\mathfrak{C}_3\mathfrak{B}}^{\mathfrak{C}_3} \quad , \end{aligned} \tag{2.41}$$

where

$$\boldsymbol{J} = \boldsymbol{J_1} + \boldsymbol{J_2} + \boldsymbol{J_3} \tag{2.42}$$

$$\boldsymbol{H} = \boldsymbol{S}(m_1 \boldsymbol{d_1} + m_2 \boldsymbol{d_2} + m_3 \boldsymbol{d_3}) \quad . \tag{2.43}$$

To write the kinetic energy in terms of the generalized coordinates we have to express the angular velocities in terms of such. It follows that:

$$\boldsymbol{w}_{\mathfrak{C}_2\mathfrak{B}}^{\mathfrak{C}_2} = \begin{bmatrix} 0 & \dot{\alpha_R} & 0 \end{bmatrix}' \tag{2.44}$$

$$\boldsymbol{w}_{\mathfrak{C}_3\mathfrak{B}}^{\mathfrak{C}_3} = \begin{bmatrix} 0 & \dot{\alpha_L} & 0 \end{bmatrix}' \quad . \tag{2.45}$$

Substituting equations (2.44), (2.45) and (2.14) in equation (2.41) and putting it in matrix form:

$$K = \frac{1}{2} \begin{bmatrix} \dot{\boldsymbol{\xi}} \\ \dot{\boldsymbol{\eta}} \\ \dot{\alpha_R} \\ \dot{\alpha_L} \end{bmatrix}' \underbrace{\begin{bmatrix} m\mathbb{I}_{3X3} & -\boldsymbol{R}_{\mathfrak{B}}^{\mathfrak{I}} \boldsymbol{H} \boldsymbol{W_\eta} & 0 & 0 \\ -\boldsymbol{W_\eta}' \boldsymbol{H} \boldsymbol{R}_{\mathfrak{I}}^{\mathfrak{B}} & \boldsymbol{W_\eta}' \boldsymbol{J} \boldsymbol{W_\eta} & \boldsymbol{W_\eta}' \boldsymbol{R}_{\mathfrak{C}_2}^{\mathfrak{B}} \boldsymbol{I_2} \boldsymbol{a} & \boldsymbol{W_\eta}' \boldsymbol{R}_{\mathfrak{C}_3}^{\mathfrak{B}} \boldsymbol{I_3} \boldsymbol{a} \\ 0 & \boldsymbol{a}' \boldsymbol{I_2} \boldsymbol{R}_{\mathfrak{B}}^{\mathfrak{C}_2} \boldsymbol{W_\eta} & \boldsymbol{a}' \boldsymbol{I_2} \boldsymbol{a} & 0 \\ 0 & \boldsymbol{a}' \boldsymbol{I_3} \boldsymbol{R}_{\mathfrak{B}}^{\mathfrak{C}_3} \boldsymbol{W_\eta} & 0 & \boldsymbol{a}' \boldsymbol{I_3} \boldsymbol{a} \end{bmatrix}}_{\mathbf{M(q)}} \begin{bmatrix} \dot{\boldsymbol{\xi}} \\ \dot{\boldsymbol{\eta}} \\ \dot{\alpha_R} \\ \dot{\alpha_L} \end{bmatrix} , \tag{2.46}$$

where $\boldsymbol{a} = \begin{bmatrix} 0 & 1 & 0 \end{bmatrix}'$ and $\boldsymbol{M}(\boldsymbol{q})$ is the so-called the inertia matrix of the system.

2.4.3 External Force Vector

The external force vector is composed of the generalized forces of the system, i.e, the external forces in relation to the generalized coordinates. It is composed of the translational forces acting on the main body expressed in the inertial frame ($\boldsymbol{T} = \begin{bmatrix} T_x^{\mathfrak{I}} & T_y^{\mathfrak{I}} & T_z^{\mathfrak{I}} \end{bmatrix}$), the rotational torques on the main body expressed on the inertial frame

($\boldsymbol{\tau} = \begin{bmatrix}\tau_x^{\mathfrak{I}} & \tau_y^{\mathfrak{I}} & \tau_z^{\mathfrak{I}}\end{bmatrix}$), the rotational torques on the servomotors of the right rotor (τ_{sR}) and of the left rotor (τ_{sL}):

$$\boldsymbol{F}(\boldsymbol{q}) = \begin{bmatrix} \boldsymbol{R}_{\mathfrak{B}}^{\mathfrak{I}} \boldsymbol{T}^{\mathfrak{B}} \\ \boldsymbol{W}_{\boldsymbol{\eta}}^{-1} \boldsymbol{\tau}^{\mathfrak{B}} \\ \tau_{sR} \\ \tau_{sL} \end{bmatrix} \quad , \tag{2.47}$$

with

$$\boldsymbol{\tau}^{\mathfrak{B}} = \boldsymbol{\tau}_a^{\mathfrak{B}} + \boldsymbol{\tau}_{drag}^{\mathfrak{B}} \quad ,$$

where $\boldsymbol{\tau}_a^{\mathfrak{B}}$ is the torque due to the actuators and $\boldsymbol{\tau}_{drag}^{\mathfrak{B}}$ due to the the air drag in the propellers. The gyroscopic effects are ignored and treated as disturbances.

As frames $\mathfrak{C}_2$ and $\mathfrak{C}_3$ are rigidly attached to its respective bodies then the thrust forces f_R and f_L generated by the rotors will always be the component in the $\vec{Z}$ axis of these frames:

$$\boldsymbol{F}_R^{\mathfrak{C}_2} = \begin{bmatrix} 0 \\ 0 \\ f_R \end{bmatrix} \quad , \quad \boldsymbol{F}_L^{\mathfrak{C}_3} = \begin{bmatrix} 0 \\ 0 \\ f_L \end{bmatrix} \quad . \tag{2.48}$$

Mapping this forces into the body frame:

$$\boldsymbol{F}_R^{\mathfrak{B}} = \boldsymbol{R}_{\mathfrak{C}_2}^{\mathfrak{B}} \boldsymbol{F}_R^{\mathfrak{C}_2} = \begin{bmatrix} f_{Rx}^{\mathfrak{B}} \\ f_{Ry}^{\mathfrak{B}} \\ f_{Rz}^{\mathfrak{B}} \end{bmatrix} = \begin{bmatrix} S(\alpha_R) \\ C(\alpha_R)S(\beta) \\ C(\alpha_R)C(\beta) \end{bmatrix} f_R \tag{2.49}$$

$$\boldsymbol{F}_L^{\mathfrak{B}} = \boldsymbol{R}_{\mathfrak{C}_3}^{\mathfrak{B}} \boldsymbol{F}_L^{\mathfrak{C}_3} = \begin{bmatrix} f_{Lx}^{\mathfrak{B}} \\ f_{Ly}^{\mathfrak{B}} \\ f_{Lz}^{\mathfrak{B}} \end{bmatrix} = \begin{bmatrix} S(\alpha_L) \\ -C(\alpha_L)S(\beta) \\ C(\alpha_L)C(\beta) \end{bmatrix} f_L \quad . \tag{2.50}$$

Note that if the inclination β of the rotors is zero, the $Y_{\mathfrak{B}}$ thrust component is also zero.

The translation forces expressed in the body frame are the sum of thrust forces:

$$\boldsymbol{T}^{\mathfrak{B}} = \boldsymbol{F}_R^{\mathfrak{B}} + \boldsymbol{F}_L^{\mathfrak{B}} \quad . \tag{2.51}$$

The torque due to the actuators around the Z^B axis ($\tau_{a\psi}$)

generates yaw motion and is caused by a difference between the magnitudes of f_{Rx}^{B} and f_{Lx}^{B}, as illustrated in figure 2.6. The resulting torque is:

$$\begin{aligned} \tau_{a\psi}^{\mathfrak{B}} &= f_{Rx}^{\mathfrak{B}} l - f_{Lx}^{\mathfrak{B}} l \\ &= [sin(\alpha_R) f_R - sin(\alpha_L) f_L] l \quad . \end{aligned} \tag{2.52}$$

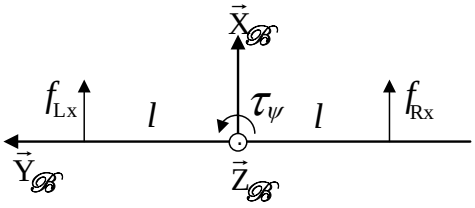


Figure 2.6 – Torque around Z axis.

A pitch rotation is caused by the torque around the Y^B axis ($\tau_{a\theta}$) and it appears if there is a projection of the thrust force in the X^B axis, as can be seen in figure 2.7. The resulting equation for the torque is:

$$\begin{aligned} \tau_{a\theta}^{\mathfrak{B}} &= f_{Rx}^{B} d_z + f_{Lx}^{B} d_z \\ &= [sin(\alpha_R) f_R - sin(\alpha_L) f_L] d_z \quad . \end{aligned} \tag{2.53}$$

The torque around the X^B axis ($\tau_{a\phi}$) causes a roll motion and it appears when there is a difference between the magnitudes of $f_{Rz}^{\mathfrak{B}}$ and $f_{Lz}^{\mathfrak{B}}$, as shown in figure 2.8. Because of the fixed lateral angle β added to the model, the forces $f_{Rz}^{\mathfrak{B}}$ and $f_{Lz}^{\mathfrak{B}}$ are not perpendicular to the lever arm $l^{'}$, with λ as the angle between f_z^B and $l^{'}$. The equation for the torque is:

$$\begin{aligned} \tau_{a\phi}^{\mathfrak{B}} &= (f_{Lz}^{B} - f_{Rz}^{B}) sin(\lambda) l^{'} \\ &= [cos(\alpha_L) f_L - cos(\alpha_R) f_R] cos(\beta) sin(\lambda) l^{'} \quad , \end{aligned} \tag{2.54}$$

with

$$l^{'} = \sqrt{d_z^2 + l^2} \tag{2.55}$$

$$\lambda = \arcsin(\frac{l}{l'}) \quad . \tag{2.56}$$

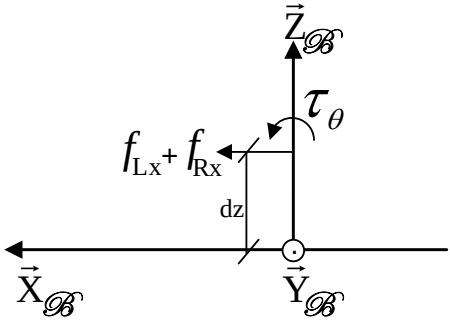


Figure 2.7 – Torque around Y axis.

Besides the torque τ_a, the UAV is also affected by the counter torques τ_M due each brushless motor rotation. Using Newton's second law and neglecting shaft friction, the torque of each motor is (CASTILLO et al., 2005):

$$J_{Mi}\dot{\Omega}_i = -\tau_{drag_i} + \tau_{Mi} \quad , \quad i = R, L \quad ,$$

where J_{Mi} is the motor angular momentum, Ω is the motor angular velocity and τ_{drag} is the torque due to each propeller's air drag and $i = R, L$ refers to the right and left motors, respectively. As the dynamics of the motors are fast when compared to the dynamics of the system, it is assumed that the propellers rotation are always in steady-state and $\dot{\Omega} = 0$, yielding:

$$\tau_{Mi} = \tau_{drag_i} = k_\tau \Omega_i^2 \quad ,$$

where $k_\tau > 0$ is a constant. Having that the force produced by each motor is (CASTILLO et al., 2005):

$$f_i = b\Omega_i^2 \quad ,$$

with b as the rotor's thrust coefficient, then the drag torque can be

expressed as:

$$\tau_{drag_i} = \frac{k_\tau}{b} f_i \quad .$$

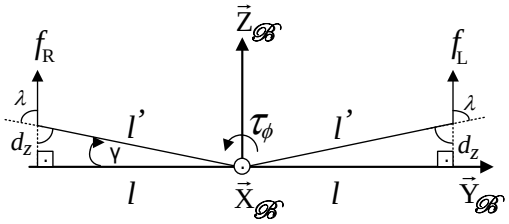


Figure 2.8 – Torque around X axis.

The rotors are set to rotate in opposite directions, the right one clockwise and the left one counter-clockwise, so that the drag torques produced by the propellers rotation cancel each other. However, when the angular velocity of the propellers or the α_R and α_L are not the same, a contribution to the UAV's rotation from the drag torques appears, as the drag torque produced in each axis by the right propeller's rotation will not be the same as its counterpart.

The drag torque equations are:

$$\boldsymbol{\tau}^{\mathfrak{B}}_{drag} = \begin{bmatrix} \boldsymbol{\tau}^{\mathfrak{B}}_{\phi drag} \\ \boldsymbol{\tau}^{\mathfrak{B}}_{\theta drag} \\ \boldsymbol{\tau}^{\mathfrak{B}}_{\psi drag} \end{bmatrix} = \begin{bmatrix} \frac{k_\tau}{b}(f^{\mathfrak{B}}_{Rx} - f^{\mathfrak{B}}_{Lx}) \\ \frac{k_\tau}{b}(f^{\mathfrak{B}}_{Ry} - f^{\mathfrak{B}}_{Ly}) \\ \frac{k_\tau}{b}(f^{\mathfrak{B}}_{Rz} - f^{\mathfrak{B}}_{Lz}) \end{bmatrix} \quad . \tag{2.57}$$

The control inputs in this model are the thrust forces (f_R and f_L) and the servo motors torques (τ_{sR} ans τ_{sL}). Defining a control input vector

$$\boldsymbol{\Gamma} = \begin{bmatrix} f_R \\ f_L \\ \tau_{sR} \\ \tau_{sL} \end{bmatrix} \quad , \tag{2.58}$$

then equation (2.47) can be written as follows:

$$\boldsymbol{F}(\boldsymbol{q}) = \boldsymbol{B}(\boldsymbol{q})\boldsymbol{\Gamma} \quad , \tag{2.59}$$

where $\boldsymbol{B}(\boldsymbol{q})$ is the decoupled external force matrix

$$\boldsymbol{B}(\boldsymbol{q}) = \begin{bmatrix} \boldsymbol{R}_{\mathfrak{B}}^{\mathfrak{I}} & 0 & 0 \\ 0 & \boldsymbol{W}_{\boldsymbol{\eta}}^{-1} & 0 \\ 0 & 0 & \mathbb{I}_{2X2} \end{bmatrix} \tilde{\boldsymbol{B}} \tag{2.60}$$

and

$$\tilde{\boldsymbol{B}} = \begin{bmatrix} S(\alpha_R) & S(\alpha_L) & 0 & 0 \\ C(\alpha_R)S(\beta) & -C(\alpha_L)S(\beta) & 0 & 0 \\ C(\alpha_R)C(\beta) & C(\alpha_L)C(\beta) & 0 & 0 \\ -C(\alpha_R)C(\beta)S(\lambda)l^{'} - \frac{k_\tau}{b}S(\alpha_R) & C(\alpha_L)C(\beta)S(\lambda)l^{'} + \frac{k_\tau}{b}S(\alpha_L) & 0 & 0 \\ S(\alpha_R)d_z + \frac{k_\tau}{b}S(\beta)C(\alpha_R) & S(\alpha_L)d_z - \frac{k_\tau}{b}S(\beta)C(\alpha_L) & 0 & 0 \\ S(\alpha_R)l + \frac{k_\tau}{b}C(\beta)C(\alpha_R) & -S(\alpha_L)l - \frac{k_\tau}{b}C(\beta)C(\alpha_L) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{2.61}$$

This equation is one of the advantages of this model over the others in literature. Note that there still are products between the servo angles and the thrust forces but, due to the fact that α_R and α_L are now a generalized coordinate, the force vector is affine in relation to the control inputs.

2.5 Equations of Motion

Finally the UAV's equations of motion can be derived. Developing equation (2.20) to its matrix form (SPONG et al., 2005):

$$\boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}})\dot{\boldsymbol{q}} + \boldsymbol{G}(\boldsymbol{q}) = \boldsymbol{F}(\boldsymbol{q}) \quad , \tag{2.62}$$

where $\boldsymbol{M}(\boldsymbol{q})$ is the inertia matrix (2.46), $\boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}})$ is the the Coriolis and centrifugal force matrix, $\boldsymbol{G}(\boldsymbol{q})$ the gravitational force vector, and $\boldsymbol{F}(\boldsymbol{q})$ the external force vector of equation (2.47).

The Coriolis and centrifugal force matrix can be calculated with Christoffel symbols of the first kind. Defining c_{kj} as the $(k, j)^{th}$

element of matrix $\boldsymbol{C}(\boldsymbol{q}, \dot{\boldsymbol{q}})$:

$$c_{kj} = \sum_{i=1}^{8} \frac{1}{2} \left\{ \frac{\partial m_{kj}}{\partial q_j} + \frac{\partial m_{ki}}{\partial q_j} - \frac{\partial m_{ij}}{\partial q_k} \right\} \dot{q}_i \quad , \tag{2.63}$$

where m_{ij} the $(i,j)^{th}$ element of matrix $\boldsymbol{M}(\boldsymbol{q})$, $j = 1, ..., 8$ and $k = 1, ..., 8$.

The gravitational force vector $\boldsymbol{G}(\boldsymbol{q})$ is calculated with the system's potential energy:

$$\boldsymbol{G}(\boldsymbol{q}) = \frac{\partial P}{\partial \boldsymbol{q}} \quad . \tag{2.64}$$

From equation (2.32):

$$\boldsymbol{G}(\boldsymbol{q}) = \begin{bmatrix} 0 \\ 0 \\ gm \\ g(m_y C\theta C\phi - m_z C\theta S\phi) \\ g(m_x C\theta - m_y S\theta S\phi - m_z S\theta C\phi) \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad , \tag{2.65}$$

where $m_x = \sum_{i=1}^{3} m_i d_{ix}$, $m_y = \sum_{i=1}^{3} m_i d_{iy}$ and $m_z = \sum_{i=1}^{3} m_i d_{iz}$.

Table 2.1 shows the parameters of the tiltrotor UAV constructed for the Project ProVANT. A computer aided design of the real mechanical assemble was made in the software *SolidWorks*® (figure 1.5), from which the system's moments of inertia were calculated. These parameters are used for all control design and simulations.

Table 2.1 shows the parameters of the system, which are taken from the tiltrotor UAV constructed for the Project ProVANT. A computer aided design of the real mechanical assemble was made in the software *SolidWorks*® (figure 1.5). The software is able to calculate all the system's moments of inertia, which are used for all control design and simulations.

Parameter	Value
β	5°
l	0.247m
g	$9.81\frac{m}{s^2}$
m_1	1.402 Kg
m_2, m_3	0.1566 Kg
d_1	(0.00672, 0.000342, -0.0789)m
d_2	(0, -0.247, 0.0123)m
d_3	(0, 0.247, 0.0123)m
d_z	0.0123m
k_τ	1.7e-7 N.m.s^2
b	9.5e-6 N.s^2
I_{1xx}	0.01902947 Kg.m^2
I_{1yy}	0.00881577 Kg.m^2
I_{1zz}	0.01747731 Kg.m^2
I_{1xy}	0.00002074 Kg.m^2
I_{1xz}	-0.00087669 Kg.m^2
I_{1yz}	0.00000808 Kg.m^2
I_{2xx}, I_{3xx}	0.00004223 Kg.m^2
I_{2yy}, I_{3yy}	0.00004096 Kg.m^2
I_{2zz}, I_{3zz}	0.00002658 Kg.m^2

Table 2.1 – System Parameters

2.6 Summary

This chapter presented the dynamical model for a birotor tiltrotor UAV. Such model was obtained using the Euler-Lagrange formulation in its matrix form. It considers a displacement of mass, which causes a coupling between the translation and rotation; a lateral fixed tilt angle (aside from the longitudinal active tilt); and considered the UAV as a multibody system of three bodies. The latter constitutes the genuine aspect of this proposal, and allowed the inclusion of the tilting mechanism dynamics.

The obtained model is highly nonlinear and coupled. It is used to create a more accurate simulation environment, whereas a linearized model is used for control purposes, as presented in the next chapter.

Chapter 3

Linear Control for Path Tracking

This chapter presents the design of control strategies that solve the hovering and path tracking problems for the birotor tiltrotor UAV. Several controllers are developed and submitted to simulation analysis built with the dynamic model (2.62), external disturbances and parameters uncertainties. The controllers are then compared and assessed.

Observing related works, it is noticed that nonlinear backstepping control strategies are prevailing. They appear in the works of (KENDOUL et al., 2005), (SANCHEZ et al., 2008), and (CHOWDHURY et al., 2012). Tiltrotor UAVs have also been controlled by linear techniques, mostly linear model predictive controllers (PAPACHRISTOS et al., 2011a; JANSEN; RAMIREZ-SERRANO, 2011) and PIDs (PAPACHRISTOS et al., 2011b).

The present work aims to develop linear optimal controllers with LQR, $\mathcal{H}_\infty$ and $\mathcal{H}_2$ techniques. LMI formulations are used in some control designs to account for parameters uncertainties, improving its robustness. Even though different objectives and techniques are used, all controllers are designed assuming all states available.

For comparison purposes, the work in (CHOWDHURY et al., 2012) was also implemented within this thesis. It proposes a nonlinear

backstepping controller for the tiltrotor. Besides, an extension of this work is proposed, as detailed in the upcoming sections.

3.1 Linear State feedback Path Tracking Control

This section presents four different linear optimal full state feedback controllers, each aiming to minimize a different cost function.

The control objective is to track a predefined trajectory $\mathbb{T}$. Initially, the external disturbances variables $\boldsymbol{\delta}(t)$ are included into the nonlinear model (2.62), leading to:

$$\boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{C}(\boldsymbol{q},\dot{\boldsymbol{q}})\dot{\boldsymbol{q}} + \boldsymbol{G}(\boldsymbol{q}) = \boldsymbol{B}(\boldsymbol{q})\boldsymbol{\Gamma} + \boldsymbol{\delta}(t) \quad , \tag{3.1}$$

with $\boldsymbol{q} = \begin{bmatrix} x(t) & y(t) & z(t) & \phi(t) & \theta(t) & \psi(t) & \alpha_R(t) & \alpha_L(t) \end{bmatrix}'$ as in equation (2.19) and $\boldsymbol{\Gamma} = \begin{bmatrix} f_R & f_L & \tau_{sR} & \tau_{sL} \end{bmatrix}'$ as in equation (2.58).

The disturbed nonlinear model (3.1) is then represented in state space and linearized around an operating point. Considering the control input vector $\boldsymbol{u}(t) = \boldsymbol{\Gamma}$ from (2.58), the disturbance $\boldsymbol{w}(t) = \boldsymbol{\delta}(t)$ and defining the state space variables as:

$$\boldsymbol{x}(t) = \begin{bmatrix} x_1 \\ x_2 \\ \cdot \\ \cdot \\ \cdot \\ x_{16} \end{bmatrix} = \begin{bmatrix} \boldsymbol{q} \\ \dot{\boldsymbol{q}} \end{bmatrix} \tag{3.2}$$

results in the nonlinear system:

$$\begin{aligned} \dot{\boldsymbol{x}} &= \begin{bmatrix} \dot{\boldsymbol{q}} \\ \ddot{\boldsymbol{q}} \end{bmatrix} = f(\boldsymbol{x}(t), \boldsymbol{u}(t), \boldsymbol{w}(t)) \\ &= \begin{bmatrix} \dot{\boldsymbol{q}} \\ \boldsymbol{M}^{-1}(\boldsymbol{B}(\boldsymbol{q})\boldsymbol{u}(t) - \boldsymbol{C}(\boldsymbol{q},\dot{\boldsymbol{q}})\dot{\boldsymbol{q}} - \boldsymbol{G}(\boldsymbol{q}) + \boldsymbol{w}(t)) \end{bmatrix} \quad . \end{aligned} \tag{3.3}$$

The equilibrium points $\bar{\boldsymbol{q}}$ for model (3.3) are the states variable values when $f(\boldsymbol{x}(t), \boldsymbol{u}(t), \boldsymbol{w}(t)) = 0$ while considering that no dis-

turbances are being applied to the system ($\boldsymbol{w}(t) = 0$), which leads to:

$$\begin{bmatrix} \dot{\bar{\boldsymbol{q}}} \\ \boldsymbol{B}(\bar{\boldsymbol{q}})\boldsymbol{u}(t) - \boldsymbol{G}(\bar{\boldsymbol{q}}) \end{bmatrix} = 0 \quad . \tag{3.4}$$

The equations in (3.4) are detailed on appendix B.6.

The tiltrotor UAV has four independent control inputs, therefore only four states can be chosen to follow arbitrary trajectories. The rest of the system states are not free to assume any value, but can be stabilized.

The chosen trajectory is composed of the UAV's linear position and heading, defined as $\mathbb{T} = (x_r(t), y_r(t), z_r(t), \psi_r(t))$. To simplify the control design, the trajectory in yaw will always track zero ($\psi_r = 0$). Analyzing (3.4) (its expansion can be seen on appendix B.6) we can note that the states $x(t)$, $y(t)$ and $z(t)$ are not present in the equilibrium equations and therefore do not have any influence in the stability of the system. So, as these states are independent from the equilibrium points, the UAV's trajectory in space can assume any form.

A reference state vector $\boldsymbol{x}_r$, which contains the set points for all system states, is defined as a composition of the trajectory and the equilibrium points:

$$\begin{aligned} \boldsymbol{x_r}(t) &= (x_r, y_r, z_r, \bar{\phi}, \bar{\theta}, 0, \bar{\alpha_R}, \bar{\alpha_L}, \dot{x}_r, \dot{y}_r, \dot{z}_r, 0, 0, 0, 0, 0) \\ &= (\boldsymbol{q_r}, \dot{\boldsymbol{q}}_{\boldsymbol{r}}) \quad . \end{aligned} \tag{3.5}$$

Substituting the parameters from table 2.1 into equation (3.4) and using the software Mathematica 9.0, we find the following the equilibrium points:

$$\begin{aligned} \bar{\phi} = -0.0000969\ rad\ , \quad \bar{\theta} = -0.0736\ rad \\ \bar{\alpha}_R = 0.0733\ rad\ , \quad \bar{\alpha}_L = 0.0733\ rad \end{aligned} \quad , \tag{3.6}$$

and the equilibrium control:

$$\begin{aligned} \bar{f}_R = 8.455\ N\ , \quad \bar{f}_L = 8.436\ N \\ \bar{\tau}_{RS} = 0\ N.m\ , \quad \bar{\tau}_{LS} = 0\ N.m \end{aligned} \quad . \tag{3.7}$$

Note that when in operation the UAV may assume different equilibrium depending on external disturbances or unmodeled dynamics.

Knowing the trajectory beforehand makes it possible to cal-

culate the control reference input $\boldsymbol{u_r}(t)$ by running the trajectory in model (3.1) with no disturbances ($\boldsymbol{\delta}(t) = 0$). This defines the reference control as:

$$\boldsymbol{u_r}(t) = \boldsymbol{\Gamma_r} = \boldsymbol{B}^{+}(\boldsymbol{q_r})(\boldsymbol{M}(\boldsymbol{q_r})\ddot{\boldsymbol{q_r}} + \boldsymbol{C}(\boldsymbol{q_r}, \dot{\boldsymbol{q_r}})\dot{\boldsymbol{q_r}} + \boldsymbol{G}(\boldsymbol{q_r})) \quad . \tag{3.8}$$

Matrix $\boldsymbol{B}(\boldsymbol{q})$ is rectangular and has full column rank, therefore $\boldsymbol{B}^{+}(\boldsymbol{q_r}) = (\boldsymbol{B}'(\boldsymbol{q_r})\boldsymbol{B}(\boldsymbol{q_r}))^{-1}\boldsymbol{B}'(\boldsymbol{q_r})$ is its left pseudoinverse matrix. However, when $\boldsymbol{u_r}(t)$ is applied in equation (3.1) it is used as a right side product in $\boldsymbol{B}(\boldsymbol{q})\boldsymbol{B}^{+}(\boldsymbol{q_r})$, a product which does not yield the identity matrix ($\boldsymbol{B}(\boldsymbol{q_r})\boldsymbol{B}^{+}(\boldsymbol{q_r}) \neq \mathbb{I}$).

The result is a projection into the column space, which is as close as we can get to the identity matrix (STRANG, 2009). This problem stems directly from the fact that the system is underactuated, which in our specific case means that eight equations must be solved with four variables.

Matrix $\boldsymbol{B}^{+}(\boldsymbol{q_r})$ is in fact the Moore-Penrose pseudoinverse of the control matrix $\boldsymbol{B}(\boldsymbol{q_r})$ and its use means that the reference control $\boldsymbol{u_r}(t)$, because $\boldsymbol{B}(\boldsymbol{q_r})\boldsymbol{B}^{+}(\boldsymbol{q_r}) \neq \mathbb{I}$, makes the system follow a slightly different trajectory than desired. This error however is considered as unmodeled system dynamics, which the controllers developed in this chapter have enough robustness to cope with.

Linearizing the model in equation (3.3) around $\boldsymbol{x_r}(t)$ and $\boldsymbol{u_r}(t)$ yields to:

$$\Delta\dot{\boldsymbol{x}}(t) = \boldsymbol{A}(\ddot{x}_r(t), \ddot{y}_r(t), \ddot{z}_r(t))\Delta\boldsymbol{x}(t) + \boldsymbol{B_u}\Delta\boldsymbol{u}(t) + \boldsymbol{B_w}\boldsymbol{w}(t) \quad , \tag{3.9}$$

with

$$\boldsymbol{A} = \frac{\partial f(\boldsymbol{x},\boldsymbol{u})}{\partial \boldsymbol{x}}\Big|^{\boldsymbol{x}=\boldsymbol{x_r}}_{\boldsymbol{u}=\boldsymbol{u_r}} \quad , \quad \boldsymbol{B_u} = \frac{\partial f(\boldsymbol{x},\boldsymbol{u})}{\partial \boldsymbol{u}}\Big|^{\boldsymbol{x}=\boldsymbol{x_r}}_{\boldsymbol{u}=\boldsymbol{u_r}} \quad , \quad \boldsymbol{B_w} = \frac{\partial f(\boldsymbol{x},\boldsymbol{u})}{\partial \boldsymbol{w}}\Big|^{\boldsymbol{x}=\boldsymbol{x_r}}_{\boldsymbol{u}=\boldsymbol{u_r}} \quad .$$

From now on (3.9) is called the error model, as $\Delta\boldsymbol{x}(t) = \boldsymbol{x}(t) - \boldsymbol{x_r}(t)$ and $\Delta\boldsymbol{u}(t) = \boldsymbol{u}(t) - \boldsymbol{u_r}(t)$ can be seen as the error of the system in relation to the desired trajectory.

Note that matrix $\boldsymbol{A}$ in the linearized system is time variant according to the desired trajectory acceleration.

The linearization was determined using the software Mathematica 9.0 and the resulting matrices can be seen in appendix B.1.

The state-space error model is augmented with an integral action for each element of the trajectory, which improves the closed loop system performance and allows the rejection of constant disturbances. The new state space vector is

$$\Delta \boldsymbol{x}(t) = \begin{bmatrix} \Delta x_1 \\ \Delta x_2 \\ . \\ . \\ . \\ \Delta x_{20} \end{bmatrix} = \begin{bmatrix} \Delta \boldsymbol{q} \\ \Delta \dot{\boldsymbol{q}} \\ \int \Delta x \\ \int \Delta y \\ \int \Delta z \\ \int \Delta \psi \end{bmatrix} \quad . \tag{3.10}$$

Considering the nominal system where $\ddot{x}_r(t) = 0, \ddot{y}_r(t) = 0, \ddot{z}_r(t) = 0$, we determine if the system is controllable by computing the rank of the controllability matrix (CHEN, 1998):

$$\begin{bmatrix} B & AB & A^2B & A^3B & ... & A^{19}B \end{bmatrix} \quad . \tag{3.11}$$

Using the software Mathematica 9.0 we find that matrix (3.11) has full rank and consequently the system is controllable, allowing us to design the state feedback controllers in the following sections.

3.1.1 Linear Quadratic Regulator (LQR) for linear time invariant systems

This section is mainly based on (MACKENROTH, 2010) and (NAIDU, 2002).

For such a complex model, classical control techniques may not work as expected. The presence of system zeros and the high coupling between the states makes the system much less intuitive to control with, for example, pole placement techniques.

As a better approach, the aim is to minimize a cost function that represents the closed loop system's energy with the linear-quadratic regulator (LQR), where

$$J = min \int_0^\infty \begin{bmatrix} \Delta \boldsymbol{x}(t) \\ \Delta \boldsymbol{u}(t) \end{bmatrix}' \begin{bmatrix} \boldsymbol{Q} & 0 \\ 0 & \boldsymbol{R} \end{bmatrix} \begin{bmatrix} \Delta \boldsymbol{x}(t) \\ \Delta \boldsymbol{u}(t) \end{bmatrix} dt \tag{3.12}$$

is the cost function and $\boldsymbol{Q}$ and $\boldsymbol{R}$ are matrices that weight the states

energy and the control inputs energy, respectively. The higher the gains in matrix $\boldsymbol{Q}$ are for a certain state, the higher will be the energy to minimize and, consequently, the slower the state will be. As for matrix $\boldsymbol{R}$, higher gains yields less control effort is used to control the system.

The LQR is chosen for its robustness properties, presenting an infinite gain margin and a phase margin of at least sixty degrees (OGATA, 1997).

Assuming that the trajectory's accelerations are zero ($\ddot{x}(t) = 0, \ddot{y}(t) = 0, \ddot{z}(t) = 0$) and no disturbances ($\boldsymbol{w}(t) = 0$) then system (3.10) becomes a linear time invariant (LTI) system with $A = A_0$. The system to be controlled is then:

$$\begin{aligned} \Delta\dot{\boldsymbol{x}}(t) &= \boldsymbol{A}\Delta\boldsymbol{x}(t) + \boldsymbol{B}\Delta\boldsymbol{u}(t) \\ \Delta\boldsymbol{u}(t) &= -\boldsymbol{K}\Delta\boldsymbol{x}(t) \end{aligned} \quad .$$

The time derivative of the Lyapunov function $V(\Delta\boldsymbol{x}(t)) = \Delta\boldsymbol{x}'(t)\boldsymbol{P}(t)\Delta\boldsymbol{x}(t)$ gives:

$$\begin{aligned} \dot{V}(t) &= \Delta\dot{\boldsymbol{x}}'(t)\boldsymbol{P}(t)\Delta\boldsymbol{x}(t) + \Delta\boldsymbol{x}'(t)\boldsymbol{P}(t)\Delta\dot{\boldsymbol{x}} + \Delta\boldsymbol{x}'(t)\dot{\boldsymbol{P}}(t)\Delta\boldsymbol{x}(t) \\ &= \Delta\boldsymbol{x}'(t)(\boldsymbol{A}'\boldsymbol{P}(t) + \boldsymbol{P}(t)\boldsymbol{A} + \dot{\boldsymbol{P}}(t))\Delta\boldsymbol{x}(t) \\ &+ \Delta\boldsymbol{u}'(t)\boldsymbol{B}'\boldsymbol{P}(t)\Delta\boldsymbol{x}(t) + \Delta\boldsymbol{x}(t)'\boldsymbol{P}(t)\boldsymbol{B}\Delta\boldsymbol{u}(t) \quad . \end{aligned} \tag{3.13}$$

As the system is time invariant, then matrix $\boldsymbol{P}(t)$ does not vary in time, leading to $\boldsymbol{P}(t) = \boldsymbol{P}$ and $\dot{\boldsymbol{P}}(t) = 0$. Now assuming that controller $\Delta\boldsymbol{u}(t) = \boldsymbol{K}\Delta\boldsymbol{x}(t)$ stabilizes the closed loop system, then $\lim_{t\to\infty} \Delta\boldsymbol{x}(t) = 0$. This means that

$$\begin{aligned} \int_0^\infty \dot{V}dt &= V(\Delta\boldsymbol{x}_{t\to\infty}) - V(\Delta\boldsymbol{x}(0)) = V(0) - V(\Delta\boldsymbol{x}(0)) \\ &= -\Delta\boldsymbol{x}'(0)\boldsymbol{P}\Delta\boldsymbol{x}(0) \quad , \end{aligned} \tag{3.14}$$

with $\Delta\boldsymbol{x}(0)$ as the initial condition of the system. Now developing equation (3.13):

$$\begin{aligned} \dot{V} &= \Delta\boldsymbol{x}'(t)(\boldsymbol{A}'\boldsymbol{P} + \boldsymbol{P}\boldsymbol{A})\Delta\boldsymbol{x}(t) + (-\boldsymbol{K}\Delta\boldsymbol{x}(t))'\boldsymbol{B}'\boldsymbol{P}\Delta\boldsymbol{x}(t) \\ &\quad + \Delta\boldsymbol{x}(t)'\boldsymbol{P}\boldsymbol{B}(-\boldsymbol{K}\Delta\boldsymbol{x}(t)) \\ &= \Delta\boldsymbol{x}'(t)(\boldsymbol{A}'\boldsymbol{P} + \boldsymbol{P}\boldsymbol{A} - \boldsymbol{K}'\boldsymbol{B}'\boldsymbol{P} - \boldsymbol{P}\boldsymbol{B}\boldsymbol{K})\Delta\boldsymbol{x}(t) \end{aligned} \tag{3.15}$$

Integrating equation (3.15) with respect to time:

$$\int_0^\infty \dot{V}\,dt = \int_0^\infty [\Delta \boldsymbol{x}'(t)(\boldsymbol{A}'\boldsymbol{P} + \boldsymbol{P}\boldsymbol{A} - \boldsymbol{K}'\boldsymbol{B}'\boldsymbol{P} - \boldsymbol{P}\boldsymbol{B}\boldsymbol{K})\Delta \boldsymbol{x}(t)]dt \quad ,$$

and substituting equation (3.14) results in

$$\Delta \boldsymbol{x}'(0)\boldsymbol{P}\Delta \boldsymbol{x}(0) + \int_0^\infty [\Delta \boldsymbol{x}'(t)(\boldsymbol{A}'\boldsymbol{P} + \boldsymbol{P}\boldsymbol{A} - \boldsymbol{K}'\boldsymbol{B}'\boldsymbol{P} - \boldsymbol{P}\boldsymbol{B}\boldsymbol{K})\Delta \boldsymbol{x}(t)]dt = 0. \tag{3.16}$$

Adding (3.16) into (3.12):

$$\begin{aligned} J = min \int_0^\infty [\Delta \boldsymbol{x}'(t)(\boldsymbol{A}'\boldsymbol{P} + \boldsymbol{P}\boldsymbol{A} - \boldsymbol{K}'\boldsymbol{B}'\boldsymbol{P} \\ - \boldsymbol{P}\boldsymbol{B}\boldsymbol{K} + \boldsymbol{Q} + \boldsymbol{K}'\boldsymbol{R}\boldsymbol{K})\Delta \boldsymbol{x}(t)]dt + \Delta \boldsymbol{x}'(0)\boldsymbol{P}\Delta \boldsymbol{x}(0) \quad . \end{aligned} \tag{3.17}$$

From (3.17) comes that the minimum possible value for the cost function is $J = \Delta \boldsymbol{x}'(0)\boldsymbol{P}\Delta \boldsymbol{x}(0)$ when

$$\boldsymbol{A}'\boldsymbol{P} + \boldsymbol{P}\boldsymbol{A} - \boldsymbol{K}'\boldsymbol{B}'\boldsymbol{P} - \boldsymbol{P}\boldsymbol{B}\boldsymbol{K} + \boldsymbol{K}'\boldsymbol{R}\boldsymbol{K} + \boldsymbol{Q} = 0 \quad . \tag{3.18}$$

Defining

$$\boldsymbol{K} = \boldsymbol{R}^{-1}\boldsymbol{B}'\boldsymbol{P} \tag{3.19}$$

transforms equation (3.18) into the algebraic Riccati equation (ARE):

$$\boldsymbol{A}'\boldsymbol{P} + \boldsymbol{P}\boldsymbol{A} - \boldsymbol{P}\boldsymbol{B}\boldsymbol{R}^{-1}\boldsymbol{B}'\boldsymbol{P} + \boldsymbol{Q} = 0 \quad , \tag{3.20}$$

where $\boldsymbol{P} > 0$ is the solution for matrices $\boldsymbol{Q}$ and $\boldsymbol{R}$ chosen positive definite, which in turn assure $\dot{V} < 0$ and that the system is asymptotically stable.

The controller design of the LQR boils down to defining matrices $\boldsymbol{Q}$ and $\boldsymbol{R}$. To assign values to matrix $\boldsymbol{Q}$, it is important to consider that, even though the tiltrotor system is represented through a single matrix, the tiltrotor has different time scales. The servomotors states have the fastest dynamics, followed by the attitude and the position being the slower ones.

As each dynamic has different units and ranges, its values are scaled using Bryson's law (JOHNSON; GRIMBLE, 1987). The results

are:

$$\boldsymbol{R} = \begin{bmatrix} \frac{0.1}{(15-\bar{f}_R)^2} & 0 & 0 & 0 \\ 0 & \frac{0.1}{(15-\bar{f}_L)^2} & 0 & 0 \\ 0 & 0 & \frac{1}{(2-\bar{\tau}_{sR})^2} & 0 \\ 0 & 0 & 0 & \frac{1}{(2-\bar{\tau}_{sL})^2} \end{bmatrix} \tag{3.21}$$

and

$$\begin{aligned} \boldsymbol{Q} = diag[1 \quad 1 \quad 1 \quad & \tfrac{1}{(\frac{\pi}{2})^2} \quad \tfrac{1}{(\frac{\pi}{2})^2} \quad \tfrac{1}{\pi^2} \quad \tfrac{0.1}{(\frac{\pi}{2})^2} \quad \tfrac{0.1}{(\frac{\pi}{2})^2} \\ & \tfrac{1}{2^2} \quad \tfrac{1}{2^2} \quad \tfrac{1}{2^2} \quad \tfrac{1}{(3*\pi)^2} \quad \tfrac{1}{(3*\pi)^2} \quad \tfrac{1}{(3*\pi)^2} \quad \tfrac{1}{(10*\pi)^2} \quad \tfrac{1}{(10*\pi)^2} \\ & 5 \quad 5 \quad 5 \quad 3] \end{aligned} \tag{3.22}$$

where $\boldsymbol{Q}$ is a 20x20 diagonal matrix.

In Matrix $\boldsymbol{R}$ the quotient $(15-\bar{f}_R)^2$ comes from the fact that 15 Newtons is the saturation of the brushless motor and the reason for deducting the equilibrium control is because we are computing $\Delta u(t)$. The same reasoning is used to design the rest of matrix $\boldsymbol{R}$. Also note that the elements in matrix $\boldsymbol{R}$ are designed to avoid an excessive energy in the servomotors control inputs.

The control gain matrix (3.19) is then obtained by solving Riccati's equation (3.20). The computed gain matrix K_{lqrLTI} is presented in appendix B.3, equation (B.9).

Simulation results for this controller are depicted in figures 3.2-3.5. The computer software used for the simulations is Simulink 2013a. The controllers are tested using the tiltrotor nonlinear model (3.1) with the parameters presented in Table 2.1. The simulation scenario has a total of sixty seconds and several aspects are included in the simulation to test the controller robustness:

1. a trajectory that has abrupt changes in its direction every ten seconds. The chosen trajectory is depicted in the appendix B.5.
2. Persistent external disturbances: $\delta_x = 0.5N$ at 5s; $\delta_y = 0.5N$ at 15s; $\delta_z = -1N$ at 25s; $\delta_\phi = 0.5N.m$ at 35s; $\delta_\theta = 0.5N.m$ at 45s; and $\delta_\psi = 0.5N.m$ at 55s, as shown in figure 3.1;
3. An uncertainty of 30% to the inertia moments was added in the simulation model.

4. Saturation on control signals , being the thrust forces constrained in $[0, 15]N$ and the servo torques in $[-2, 2]N.m$, in accordance with the corresponding actuators specifications for Project ProVANT.

The initial condition is zero ($\boldsymbol{x}(0) = 0$), which can easily be achieved in real life experiments.

To help analyze the performance of this and subsequent controllers, the Integral Squared Error (ISE) and Integrated Absolute Variation of the Control signal (IAVU) performance indexes are used. The ISE index is relative to the tracking error of the control, a higher ISE meaning that the control performed a worse path tracking when compared to a lower value. The IAVU index is relative to the control signals with a lower index suggesting a smoother control.

Table 3.1 brings the ISE for the LQR LTI controller and table 3.2 the IAVU. They will be useful for comparison with the controllers developed in the following sections.

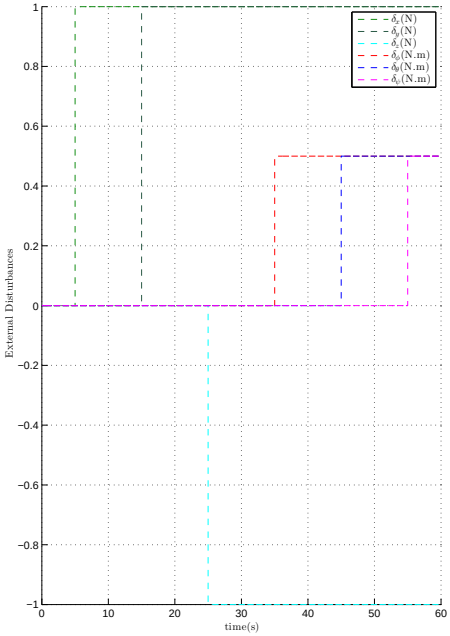


Figure 3.1 – External disturbances applied to the simulations.

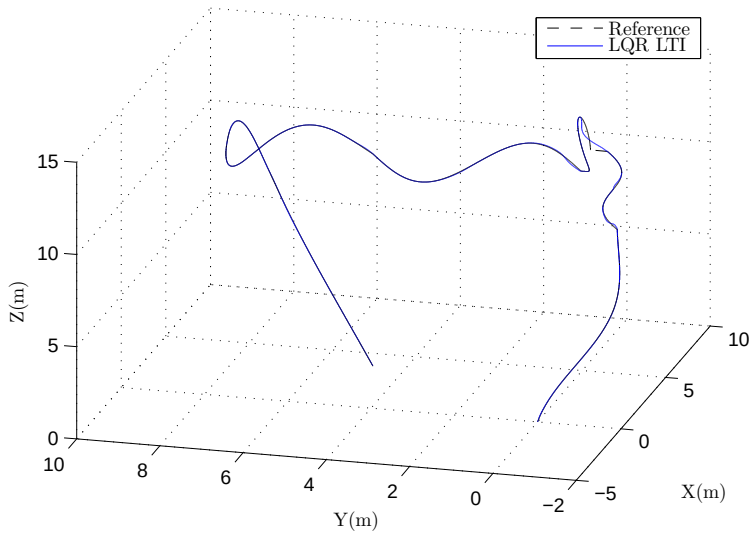


Figure 3.2 – Tiltrotor trajectory in space for the LQR LTI controller.

	LQR LTI
ISE_x	0.0571
ISE_y	0.0656
ISE_z	0.0012
ISE_ψ	0.0250

Table 3.1 – Integral Squared Error for the LQR LTI controller.

	LQR LTI
$IAVU\ f_R$	542.647
$IAVU\ f_L$	506.637
$IAVU\ \tau_{SR}$	0.0136
$IAVU\ \tau_{SL}$	0.0148

Table 3.2 – Integrated Absolute Variation of the Control signal for the LQR LTI controller.

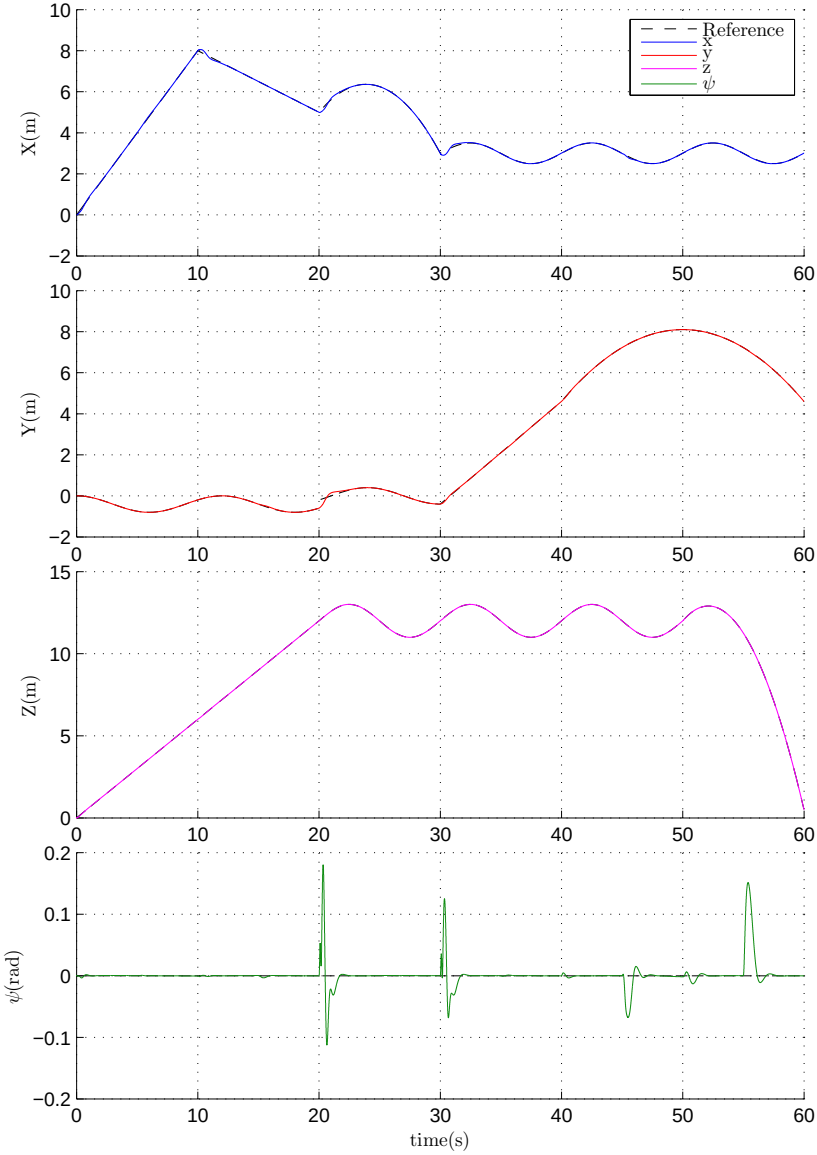


Figure 3.3 – Tiltrotor regulated states for the LQR LTI controller.

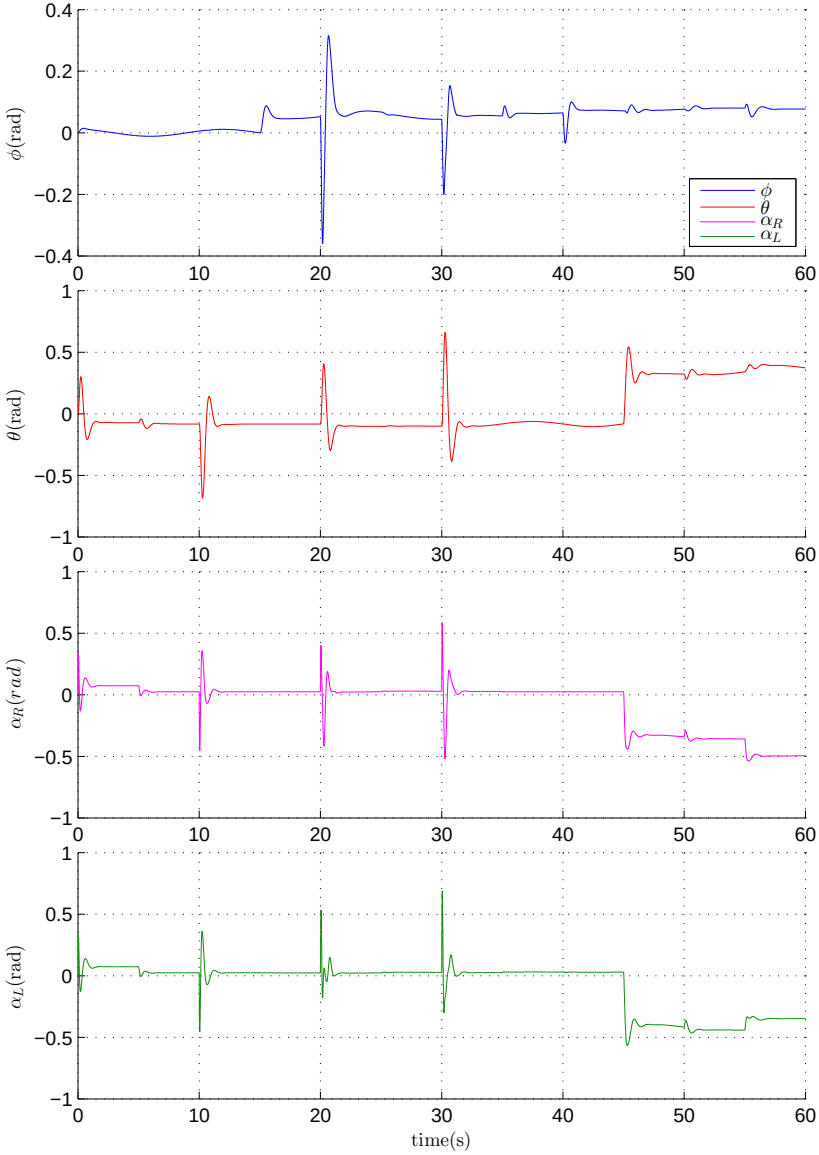


Figure 3.4 – Tiltrotor stabilized states for the LQR LTI controller.

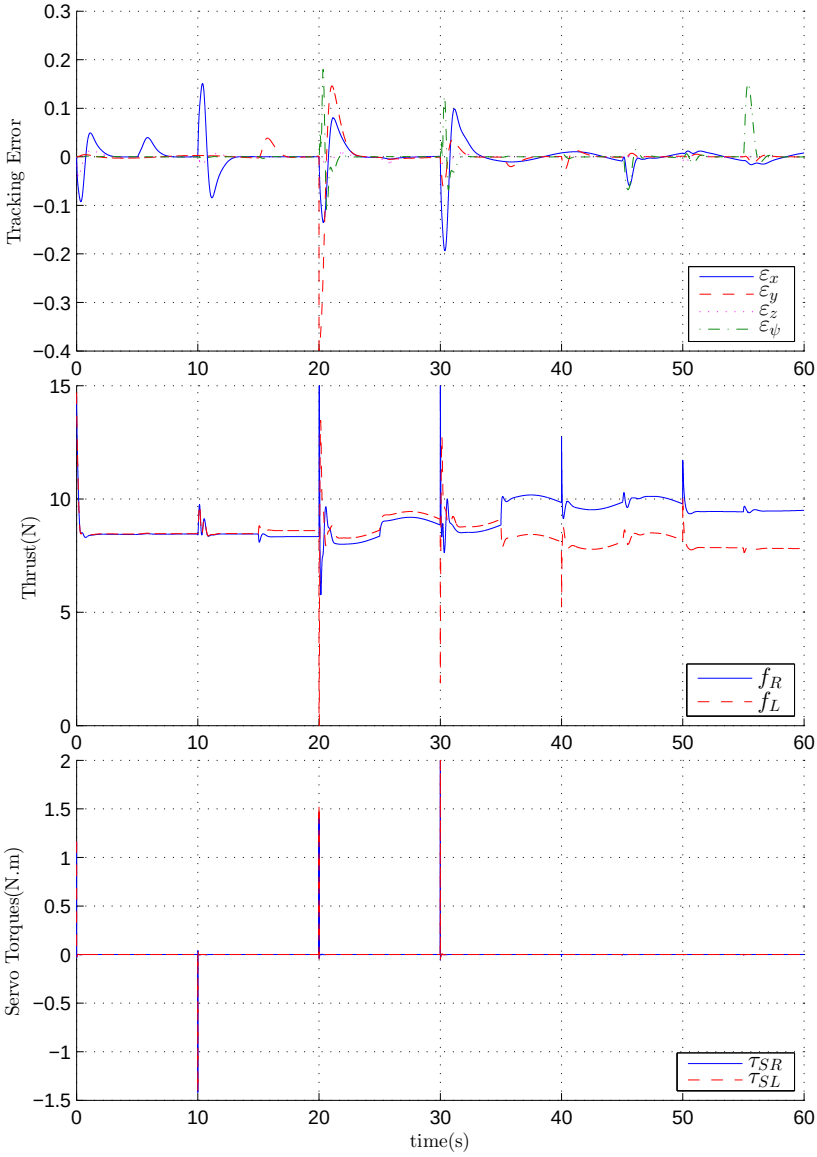


Figure 3.5 – Tiltrotor tracking error and control inputs for the LQR LTI controller.

From the ISE we can see that the tracking error for the $z(t)$ and $\psi(t)$ are significant lower when compared to $x(t)$ and $y(t)$. This falls in place with our understanding of the tiltrotor system, as the former states are less coupled than the latter, meaning they are easier to control. From the IAVU is interesting to note that the f_R control signal is significantly higher then f_L. This is directly related to the center of mass position, which is slightly tilted to the right side, requiring more thrust force from the right rotor.

The LQR LTI controller has a good performance with a low tracking error. The control signal has high peaks at 20 seconds and 30 seconds of the simulation and are due to the abrupt change in the trajectory. These two moments aside, the control is smooth, even when the disturbances are applied. This is a good result as the disturbances values are relatively high.

The drawback to this controller is that it considers only the nominal system, ignoring its time varying nature ($\ddot{x}_r(t) = \ddot{y}_r(t) = \ddot{z}_r(t) = 0$), which can be a factor to destabilize the system. The controller treated in the next section aims to increase robustness related to this aspect, as well as to actuators saturations.

3.1.2 LMI formulation for the LQR control

This section is mainly based on (TROFINO et al., 2003) and (BOYD et al., 1994).

Aiming for a robust control of the system, the time varying parameters of the error model (3.9) are considered as uncertainties and the system is rewritten as follows:

$$\Delta\dot{\boldsymbol{x}}(t) = \boldsymbol{A}(\alpha)\Delta\boldsymbol{x}(t) + \boldsymbol{B_u}\Delta\boldsymbol{u}(t) + \boldsymbol{B_w}\boldsymbol{w}(t) \quad , \tag{3.23}$$

where α represents uncertainties in the system. In this controller, and the others which use a LMI formulation, $\ddot{x}_r(t), \ddot{y}_r(t)$ and $\ddot{z}_r(t)$ are considered as uncertainties. Then by using linear matrices inequalities (LMIs) and considering boundaries for the minimum and maximum values of the trajectory accelerations, linear control techniques can be used while still guaranteeing the stability of the system throughout any given trajectory within the boundaries.

In summary, this controller's purpose is to design a control

law based on a similar cost function to the ideal LQR from last section but taking into consideration the boundaries of time varying parameters for the control design, alongside the uncertainty in the system's mass and inertia moments. These considerations yields more conservative controllers in exchange for robustness.

Considering no disturbances ($w(t) = 0$), a state feedback control and the cost function J associated with the performance output vector $\boldsymbol{z} \in R^{20}$,

$$J = min \int_0^\infty \boldsymbol{z}(t)'\boldsymbol{z}(t)dt \quad , \tag{3.24}$$

then from (3.23):

$$\begin{cases} \Delta\dot{\boldsymbol{x}}(t) = \boldsymbol{A}(\alpha)\Delta\boldsymbol{x}(t) + \boldsymbol{B_u}\Delta\boldsymbol{u}(t) \\ \boldsymbol{z}(t) = \boldsymbol{C_z}\Delta\boldsymbol{x}(t) + \boldsymbol{D_{uz}}\Delta\boldsymbol{u}(t) \\ \Delta\boldsymbol{u}(t) = \boldsymbol{K}\Delta\boldsymbol{x}(t) \qquad \alpha \in \nu \end{cases} , \tag{3.25}$$

where ν is a convex polytope with known vertices v_i.

Using the Lyapunov function $V(\Delta\boldsymbol{x}(t)) = \Delta\boldsymbol{x}'(t)\boldsymbol{P}\Delta\boldsymbol{x}(t)$, for the system to be asymptotically stable then it must obey $\dot{V}(t) < 0$. As the expression $\boldsymbol{z}'(t)\boldsymbol{z}(t)$ will always be positive for any $\boldsymbol{z} \neq 0$, then the Lyapunov inequality can expanded to (BOYD et al., 1994):

$$\dot{V}(t) < -\boldsymbol{z}'(t)\boldsymbol{z}(t) < 0 \quad . \tag{3.26}$$

Integrating equation (3.26) with respect to time and using equation (3.14), yields to:

$$\int_0^\infty \boldsymbol{z}(t)'\boldsymbol{z}(t)dt < \Delta\boldsymbol{x}'(0)P\Delta\boldsymbol{x}(0) \quad . \tag{3.27}$$

By minimizing $\Delta\boldsymbol{x}'(0)P\Delta\boldsymbol{x}(0)$ we are minimizing an upper bound of the cost function J in equation (3.24), which can be done by minimizing a scalar λ where (TROFINO et al., 2003):

$$\lambda - \Delta\boldsymbol{x}'(0)P\Delta\boldsymbol{x}(0) > 0 \quad . \tag{3.28}$$

Now substituting equation (3.15) in equation (3.26) develops

to:

$$\begin{aligned}\dot{V}(t) + \boldsymbol{z}'(t)\boldsymbol{z}(t) =& \Delta\boldsymbol{x}'[(\boldsymbol{A}(\alpha) + \boldsymbol{B_u}\boldsymbol{K})'\boldsymbol{P} + \boldsymbol{P}(\boldsymbol{A}(\alpha) + \boldsymbol{B_u}\boldsymbol{K}) \\ &+ (\boldsymbol{C_z} + \boldsymbol{D_{uz}}\boldsymbol{K})'(\boldsymbol{C_z} + \boldsymbol{D_{uz}}\boldsymbol{K})]\Delta\boldsymbol{x} < 0 \quad .\end{aligned} \tag{3.29}$$

Defining the transformations

$$\boldsymbol{W} = \boldsymbol{P}^{-1} > 0 \quad , \quad \boldsymbol{Y} = \boldsymbol{K}\boldsymbol{W} \quad , \tag{3.30}$$

and multiplying the left-hand side of equation (3.29) by $\boldsymbol{W}'\Delta\boldsymbol{x}'^{-1}(t)$ and the right side by $\Delta\boldsymbol{x}^{-1}(t)\boldsymbol{W}$:

$$\begin{aligned}&(\boldsymbol{A}(\alpha)\boldsymbol{W} + \boldsymbol{B_u}\boldsymbol{Y})' + (\boldsymbol{A}(\alpha)\boldsymbol{W} + \boldsymbol{B_u}\boldsymbol{Y}) \\ &\quad + (\boldsymbol{C_z}\boldsymbol{Q} + \boldsymbol{D_{uz}}\boldsymbol{Y})'(\boldsymbol{C_z}\boldsymbol{W} + \boldsymbol{D_{uz}}\boldsymbol{Y}) < 0 \quad .\end{aligned} \tag{3.31}$$

Finally, applying the Schur complement (see appendix A) in equations (3.31) and (3.28) , the cost function J is subject to the following LMIs:

$$\begin{bmatrix} \lambda & \Delta\boldsymbol{x}'(0) \\ \Delta\boldsymbol{x}'(0) & W \end{bmatrix} > 0 \tag{3.32}$$

$$\begin{bmatrix} (\boldsymbol{A}(\alpha)\boldsymbol{W} + \boldsymbol{B_u}\boldsymbol{Y})' + (\boldsymbol{A}(\alpha)\boldsymbol{W} + \boldsymbol{B_u}\boldsymbol{Y}) & (\boldsymbol{C_z}\boldsymbol{W} + \boldsymbol{D_{uz}}\boldsymbol{Y})' \\ (\boldsymbol{C_z}\boldsymbol{W} + \boldsymbol{D_{uz}}\boldsymbol{Y}) & -I_{nz} \end{bmatrix} < 0, \tag{3.33}$$

with $\boldsymbol{K} = \boldsymbol{Y}\boldsymbol{W}^{-1}$.

Given the convexity of the LMIs, we can solve (3.33) only for the vertices v_i of the uncertainty α, where i ranges from 1 to the total number of vertices. The LMIs are then:

$$\begin{bmatrix} (\boldsymbol{A}(v_i)\boldsymbol{W} + \boldsymbol{B_u}\boldsymbol{Y})' + (\boldsymbol{A}(v_i)\boldsymbol{W} + \boldsymbol{B_u}\boldsymbol{Y}) & (\boldsymbol{C_z}\boldsymbol{W} + \boldsymbol{D_{uz}}\boldsymbol{Y})' \\ (\boldsymbol{C_z}\boldsymbol{W} + \boldsymbol{D_{uz}}\boldsymbol{Y}) & -I_{nz} \end{bmatrix} < 0. \tag{3.34}$$

In our case the vertices are a combination of $\ddot{x}(t) \in [-0.5 \;\; 0.5]$, $\ddot{y}(t) \in [-0.5 \;\; 0.5]$ and $\ddot{z}(t) \in [-0.5 \;\; 0.5]$, comprising $2^3 = 8$ vertices.

To obtain a similar controller as the LQR in section 3.1.1, let

us first expand the cost function (3.24):

$$J = min \int_0^\infty [\Delta \boldsymbol{x}'(t)\boldsymbol{C}_{\boldsymbol{z}}'\boldsymbol{C}_{\boldsymbol{z}}\Delta \boldsymbol{x}(t) + \Delta \boldsymbol{u}(t)'\boldsymbol{D}_{\boldsymbol{uz}}'\boldsymbol{D}_{\boldsymbol{uz}}\Delta \boldsymbol{u}(t) + \Delta \boldsymbol{x}'(t)\boldsymbol{C}_{\boldsymbol{z}}'\boldsymbol{D}_{\boldsymbol{uz}}\Delta \boldsymbol{u}(t) + \Delta \boldsymbol{u}(t)'\boldsymbol{D}_{\boldsymbol{uz}}'\boldsymbol{C}_{\boldsymbol{z}}\Delta \boldsymbol{x}(t)]dt \quad . \tag{3.35}$$

For the cost function (3.35) to be the same as (3.12), the design of matrices $\boldsymbol{C}_{\boldsymbol{z}}$ and $\boldsymbol{D}_{\boldsymbol{uz}}$ will aim to make $\boldsymbol{C}\boldsymbol{z}'\boldsymbol{D}_{\boldsymbol{uz}} = 0$ and $\boldsymbol{C}_{\boldsymbol{z}}'\boldsymbol{C}_{\boldsymbol{z}} = \boldsymbol{Q}$, $\boldsymbol{D}_{\boldsymbol{uz}}'\boldsymbol{D}_{\boldsymbol{uz}} = \boldsymbol{R}$, leading to:

$$\boldsymbol{D}_{\boldsymbol{uz}} = \begin{bmatrix} \sqrt{\frac{0.1}{(15-\bar{f}_R)^2}} & 0 & 0 & 0 \\ 0 & \sqrt{\frac{0.1}{(15-\bar{f}_L)^2}} & 0 & 0 \\ 0 & 0 & \sqrt{\frac{1}{2^2}} & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \sqrt{\frac{1}{2^2}} \\ 0_{12,1} & 0_{12,1} & 0_{12,1} & 0_{12,1} \end{bmatrix} \tag{3.36}$$

$$\begin{aligned} \boldsymbol{C}_{\boldsymbol{z}_{\boldsymbol{diag}}} = diag[1 \quad 1 \quad 1 \quad & \sqrt{\tfrac{1}{(\frac{\pi}{2})^2}} \quad \sqrt{\tfrac{1}{(\frac{\pi}{2})^2}} \quad \sqrt{\tfrac{1}{\pi^2}} \quad \sqrt{\tfrac{0.1}{(\frac{\pi}{2})^2}} \quad \sqrt{\tfrac{0.1}{(\frac{\pi}{2})^2}} \\ & \sqrt{\tfrac{1}{2^2}} \quad \sqrt{\tfrac{1}{2^2}} \quad \sqrt{\tfrac{1}{2^2}} \quad \sqrt{\tfrac{1}{(3*\pi)^2}} \quad \sqrt{\tfrac{1}{(3*\pi)^2}} \quad \sqrt{\tfrac{1}{(3*\pi)^2}} \\ & \sqrt{\tfrac{1}{(10*\pi)^2}} \quad \sqrt{\tfrac{1}{(10*\pi)^2}} \quad \sqrt{5} \quad \sqrt{5} \quad \sqrt{5} \quad \sqrt{3}] \quad . \end{aligned} \tag{3.37}$$

Note that a compromise had to be made to have $\boldsymbol{C}_{\boldsymbol{z}}'\boldsymbol{C}_{\boldsymbol{z}} = \boldsymbol{Q}$ and $\boldsymbol{D}_{\boldsymbol{uz}}'\boldsymbol{D}_{\boldsymbol{uz}} = \boldsymbol{R}$, resulting in the expression $\boldsymbol{C}\boldsymbol{z}'\boldsymbol{D}_{\boldsymbol{uz}}$ not being zero. The values however are small and sparse and have little influence over the results.

Finally, to avoid saturations in the control signals, we include a constraint in the control action, formulated with LMIs developed in (ALAMO et al., 2006). They are:

$$\begin{bmatrix} 1 & \Delta \boldsymbol{x}(0) \\ \Delta \boldsymbol{x}(0)' & \boldsymbol{P}^{-1} \end{bmatrix} > 0 \tag{3.38}$$

$$\begin{bmatrix} \boldsymbol{U}_{max}^2 & \boldsymbol{K} \\ \boldsymbol{K}' & \boldsymbol{P} \end{bmatrix} > 0 \quad , \tag{3.39}$$

where $\boldsymbol{U_{max}} = diag([15, 15, 2, 2] - \boldsymbol{u_r})$ is the desired maximum control effort and $\boldsymbol{x}(0)$ is the initial condition. The transformations (3.30) are once applied in order to maintain the same variables, resulting in:

$$\begin{bmatrix} 1 & \Delta\boldsymbol{x}(0) \\ \Delta\boldsymbol{x}(0)' & \boldsymbol{W} \end{bmatrix} > 0 \quad , \tag{3.40}$$

and

$$\begin{bmatrix} I & 0 \\ 0 & \boldsymbol{P}^{-1} \end{bmatrix} \begin{bmatrix} \boldsymbol{U}_{max}^2 & \boldsymbol{K} \\ \boldsymbol{K}' & \boldsymbol{P} \end{bmatrix} \begin{bmatrix} I & 0 \\ 0 & \boldsymbol{P}^{-1} \end{bmatrix} > 0$$
$$\begin{bmatrix} \boldsymbol{U}_{max}^2 & \boldsymbol{Y} \\ \boldsymbol{Y}' & \boldsymbol{Q} \end{bmatrix} > 0 \quad . \tag{3.41}$$

Equation (3.40) is already represented by equation (3.32) and is not used for the design of this controller.

Finally, minimizing λ subject to equations (3.32), (3.34) and (3.41) solves for the desired controller. Using MATLAB's Robust Control Toolbox version 4.3 alongside an initial condition $\Delta\boldsymbol{x}(0) = 0$, the gain matrix is obtained with $\lambda \approx 0$ (see appendix B, equation (B.10)). Simulations for this controller were made under the same conditions as used in section 3.1.1, the results being depicted in figures 3.6-3.9 and tables 3.3 for the ISE index and 3.4 for the IAVU.

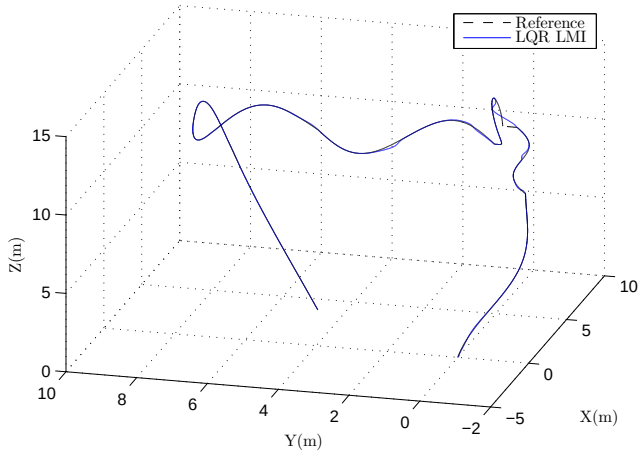


Figure 3.6 – Tiltrotor trajectory in space for the LQR LMI controller.

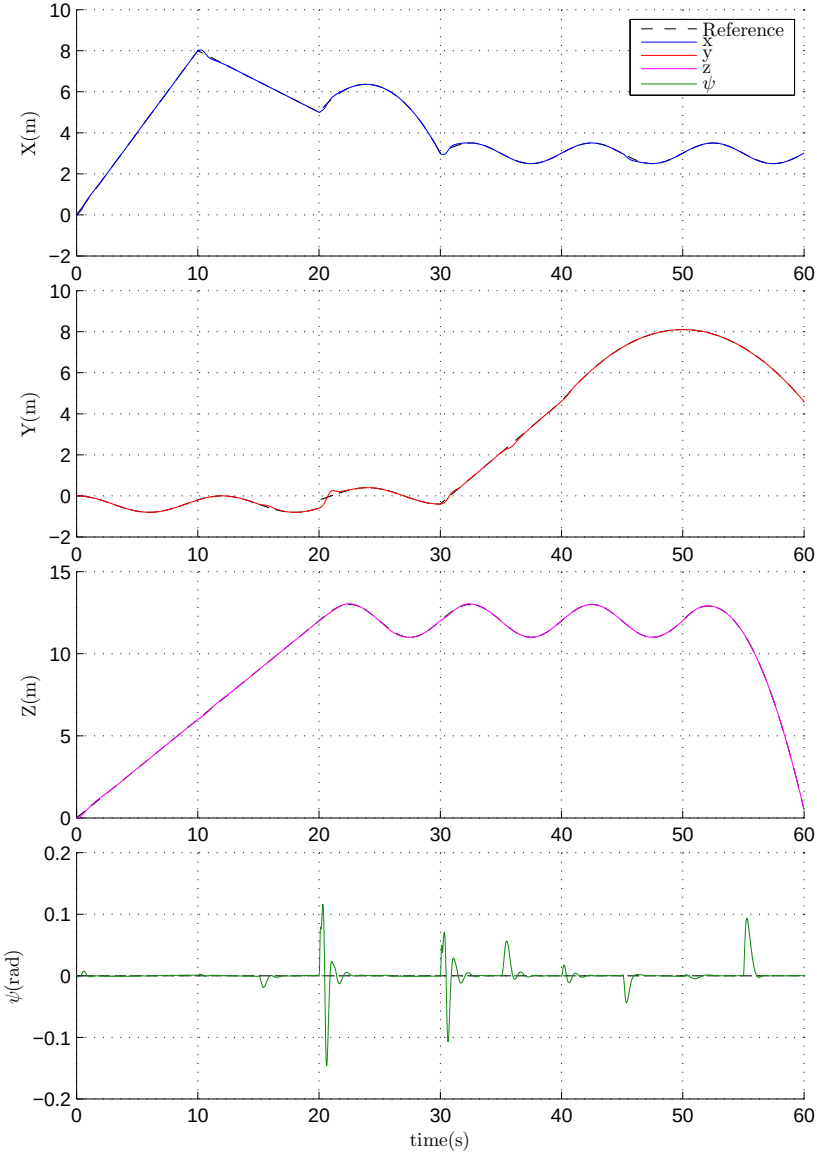


Figure 3.7 – Tiltrotor regulated states for the LQR LMI controller.

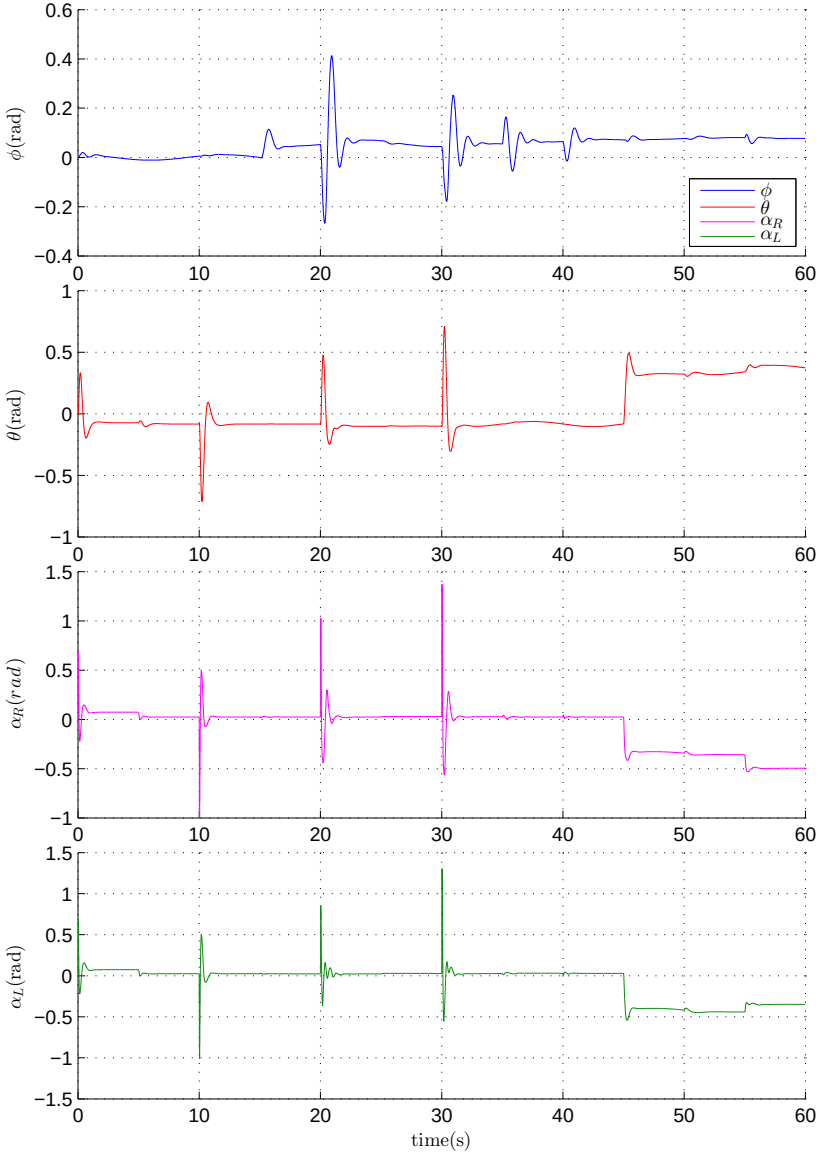


Figure 3.8 – Tiltrotor stabilized states for the LQR LMI controller.

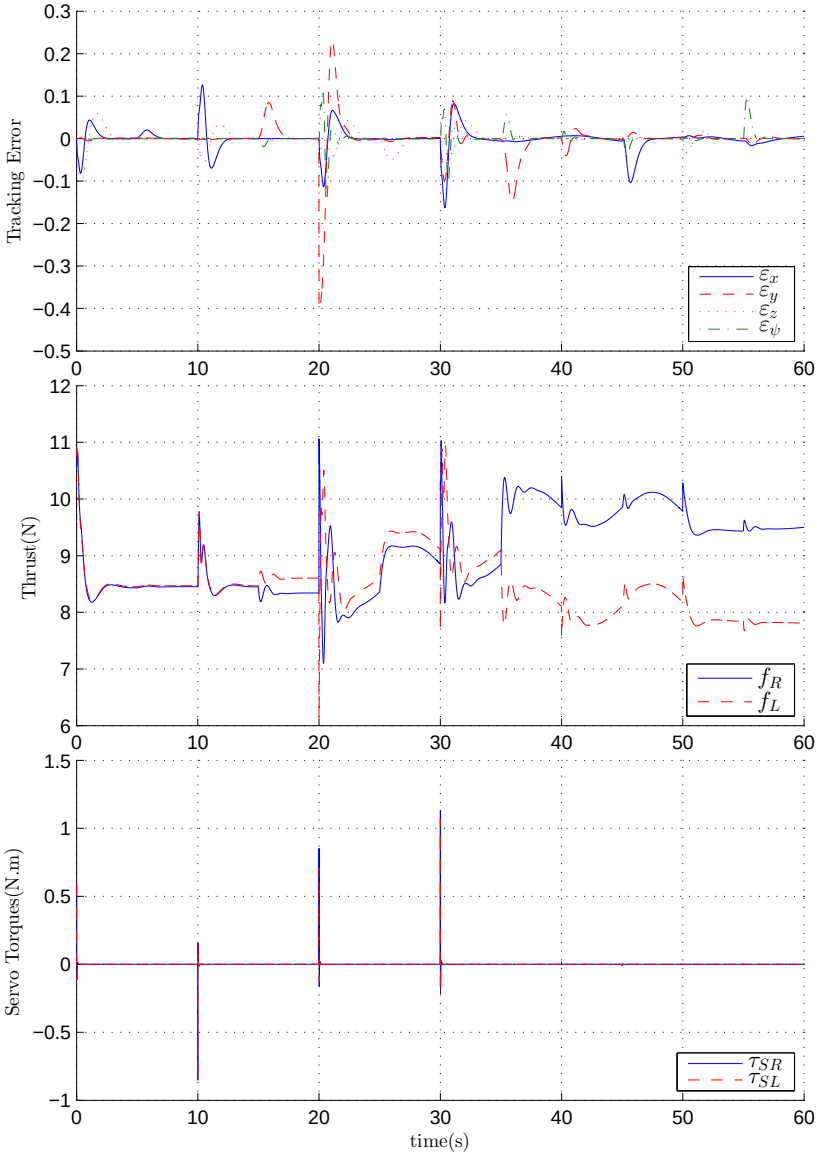


Figure 3.9 – Tiltrotor tracking error and control inputs for the LQR LMI controller.

	LQR LMI
ISE_x	0.0412
ISE_y	0.1163
ISE_z	0.0219
ISE_ψ	0.0163

Table 3.3 – Integral Squared Error for the LQR LMI controller.

	LQR LMI
$IAVU\ f_R$	543.238
$IAVU\ f_L$	507.171
$IAVU\ \tau_{SR}$	0.0342
$IAVU\ \tau_{SL}$	0.0317

Table 3.4 – Integrated Absolute Variation of the Control signal for the LQR LMI controller.

Comparing the ISE index in table 3.3 for this controller to the index from LQR LTI in table 3.2, we can note that this controller has a slightly better performance for the $x(t)$ and $\psi(t)$ states. However the tracking performance in the other two states is significantly worse and the IAVU index shows an overall more aggressive control signal, suggesting that this controller is marginally worse than the previous one.

Note however that this controller uses the same LQR technique with the inclusion of the system's time varying aspects and control saturations into its design, resulting in a more robust controller.

3.1.3 $\mathcal{H}_\infty$ control

This section is mainly based on (DULLERUD; PAGANINI, 2005) and (BOYD et al., 1994).

In this section a linear $\mathcal{H}_\infty$ control is designed with a LMI formulation. This controller was chosen for its properties of disturbance rejection.

Consider the linear system:

$$\begin{cases} \Delta\dot{\boldsymbol{x}}(t) = \boldsymbol{A}(\alpha)\Delta\boldsymbol{x}(t) + \boldsymbol{B_u}\Delta\boldsymbol{u}(t) + \boldsymbol{B_w}\boldsymbol{w}(t) \\ \boldsymbol{z}(t) = \boldsymbol{C_z}\Delta\boldsymbol{x}(t) + \boldsymbol{D_{uz}}\Delta\boldsymbol{u}(t) + \boldsymbol{D_{wz}}\boldsymbol{w}(t) \\ \Delta\boldsymbol{u}(t) = \boldsymbol{K}\Delta\boldsymbol{x}(t) \qquad \alpha \in \nu \end{cases} , \tag{3.42}$$

where $\boldsymbol{w}(t)$ is the disturbance vector, $\boldsymbol{z}(t)$ is the signal that will have its $\mathcal{H}_\infty$ norm minimized, $\boldsymbol{C_z}$, $\boldsymbol{D_{wz}}$, $\boldsymbol{D_{uz}}$ are constant matrices to be determined and ν is a polytope with known vertices v_i.

The $\mathcal{H}_\infty$ feedback controller is an optimal control that exponentially stabilizes the system and minimizes its $\mathcal{H}_\infty$ norm

$$||\boldsymbol{H_{wz}}(s)||_\infty = sup\ \bar{\sigma}\{\boldsymbol{H_{wz}}(jw)\} \quad , \tag{3.43}$$

where $\boldsymbol{H_{wz}}(s)$ is the transfer function between the external disturbance $\boldsymbol{w}(t)$ and $\boldsymbol{z}(t)$.

The $\mathcal{H}_\infty$ norm represents the system's highest frequency response gain. It can also be interpreted as the system's highest energy gain due to an input signal. Applying Parseval's theorem in equation (3.43) results:

$$||\boldsymbol{z}(t)||_2 \leq ||\boldsymbol{H_{wz}}(s)||_\infty ||\boldsymbol{w}(t)||_2$$

or

$$||\boldsymbol{H_{wz}}(s)||_\infty = sup\ \frac{||\boldsymbol{z}(t)||_2}{||\boldsymbol{w}(t)||_2} \quad .$$

This definition makes it clear that minimizing the $\mathcal{H}_\infty$ norm minimizes the effect of external disturbances on the system.

In this work a relaxation of the optimization is made, where the aim is to find an upper bound γ of the norm, thus:

$$||\boldsymbol{H_{wz}}(s)||_\infty < \gamma \quad . \tag{3.44}$$

Considering a Lyapunov function $V(\boldsymbol{x}(t)) = \Delta\boldsymbol{x}'(t)\boldsymbol{P}\Delta\boldsymbol{x}(t)$ and initial conditions $\Delta\boldsymbol{x}(0) = 0$, the following inequation can be used to find the control that minimizes γ (BOYD, 1994):

$$\dot{V}(\Delta\boldsymbol{x}(t)) + \boldsymbol{z}'(t)\boldsymbol{z}(t) - \gamma^2\boldsymbol{w}'(t)\boldsymbol{w}(t) < 0 \quad . \tag{3.45}$$

To verify that by minimizing equation (3.45) we are indeed minimizing the H_∞ norm, first we integrate it with respect to relation

to time and, using (3.14):

$$\int_0^\infty \boldsymbol{z}'(t)\boldsymbol{z}(t)dt < \gamma^2 \int_0^\infty \boldsymbol{w}'(t)\boldsymbol{w}(t)dt \quad , \tag{3.46}$$

which leads to

$$sup\frac{\sqrt{\int_0^\infty \boldsymbol{z}'(t)\boldsymbol{z}(t)dt}}{\sqrt{\int_0^\infty \boldsymbol{w}'(t)\boldsymbol{w}(t)dt}} = sup\ \frac{||\boldsymbol{z}(t)||_2}{||\boldsymbol{w}(t)||_2} < \gamma \quad . \tag{3.47}$$

Developing equation (3.45) with the transformations

$$\boldsymbol{Q} = \gamma \boldsymbol{P}^{-1} \quad , \quad \boldsymbol{Y} = \boldsymbol{K}\boldsymbol{Q} \tag{3.48}$$

leads to the LMIs (DULLERUD; PAGANINI, 2005):

$$\boldsymbol{Q} > 0 \tag{3.49}$$

$$\begin{bmatrix} \boldsymbol{A}(v_i)\boldsymbol{Q} + \boldsymbol{Q}\boldsymbol{A}'(v_i) + \boldsymbol{B}(v_i)\boldsymbol{Y} + \boldsymbol{Y}'\boldsymbol{B}'(v_i) & \boldsymbol{B_w}(v_i) & \boldsymbol{Q}\boldsymbol{C_z}' + \boldsymbol{Y}'\boldsymbol{D_{uz}}' \\ \boldsymbol{B_w}'(v_i) & -\tilde{\gamma}\mathbb{I}_{nz} & \boldsymbol{D_{wz}}' \\ \boldsymbol{C_z}\boldsymbol{Q} + \boldsymbol{D_{uz}}\boldsymbol{Y} & \boldsymbol{D_{wz}} & -\tilde{\gamma}\mathbb{I}_{nz} \end{bmatrix} < 0,$$

where $\tilde{\gamma} = \gamma^2$ and v_i refers to the vertices of polytope ν. In our case the vertices are a combination of $\ddot{x}(t) \in [-0.5 \ \ 0.5]$, $\ddot{y}(t) \in [-0.5 \ \ 0.5]$ and $\ddot{z}(t) \in [-0.5 \ \ 0.5]$, totaling $2^3 = 8$ vertices.

Solving this optimization problem for $\boldsymbol{Q} = \boldsymbol{Q}'$ and $\boldsymbol{Y}$ makes system (3.42) with $\boldsymbol{K} = \boldsymbol{Y}\boldsymbol{Q}^{-1}$ asymptotic stable and $||\boldsymbol{H_{wz}}(s)||_\infty < \gamma$.

The controller design follows the same logic as in section 3.1.2 and reuses matrices C_z and D_z depicted in (3.37) and (3.36). This leaves only matrix D_w to be designed which, after some iterations, became:

$$\boldsymbol{D_{wz}} = \begin{bmatrix} \sqrt{0.1} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \sqrt{0.1} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \sqrt{0.1} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & \sqrt{0.1} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \sqrt{0.1} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \sqrt{0.1} & 0 & 0 \\ 0_{14,1} & 0_{14,1} & 0_{14,1} & 0_{14,1} & 0_{14,1} & 0_{14,1} & 0_{14,1} & 0_{14,1} \end{bmatrix} . \tag{3.50}$$

The saturation in the control action is also considered. Using the LMIs (3.38) and (3.39) with the transformations (3.48) we obtain:

$$\begin{bmatrix} 1 & 0 \\ 0 & \gamma^{\frac{1}{2}} \end{bmatrix} \begin{bmatrix} 1 & \boldsymbol{x}(0) \\ \boldsymbol{x}(0)' & \boldsymbol{P}^{-1} \end{bmatrix} \begin{bmatrix} 1 & 0 \\ 0 & \gamma^{\frac{1}{2}} \end{bmatrix} > 0 \quad , \quad \begin{bmatrix} 1 & \boldsymbol{x}(0) \\ \boldsymbol{x}(0)' & \boldsymbol{Q} \end{bmatrix} > 0 \tag{3.51}$$

and

$$\begin{bmatrix} \gamma^{\frac{1}{2}} & 0 \\ 0 & \gamma^{\frac{1}{2}}\boldsymbol{P}^{-1} \end{bmatrix} \begin{bmatrix} \boldsymbol{U}_{max}^2 & \boldsymbol{K} \\ \boldsymbol{K} & \boldsymbol{P} \end{bmatrix} \begin{bmatrix} \gamma^{\frac{1}{2}} & 0 \\ 0 & \gamma^{\frac{1}{2}}\boldsymbol{P}^{-1} \end{bmatrix} > 0 \quad , \quad \begin{bmatrix} \gamma\boldsymbol{U}_{max}^2 & \boldsymbol{Y} \\ \boldsymbol{Y}' & \boldsymbol{Q} \end{bmatrix} > 0 \tag{3.52}$$

The gain matrix is found by minimizing γ subject to the LMIs (3.49), (3.51) and (3.52) using MATLAB's Robust Control Toolbox version 4.3 (see appendix B, equation (B.11)). As a result we have that $\tilde{\gamma} = 1.8365$, and $||\boldsymbol{H_{wz}}(s)||_\infty < 1.3552$. Simulations for this controller were made under the same conditions as used in section 3.1.1, the results being depicted in figures 3.10-3.13 and tables 3.5 for the ISE index and 3.6 for the IAVU.

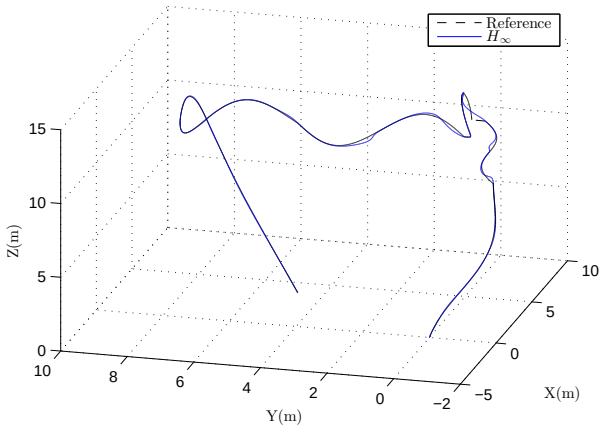


Figure 3.10 – Tiltrotor trajectory in space for the $\mathcal{H}_\infty$ controller.

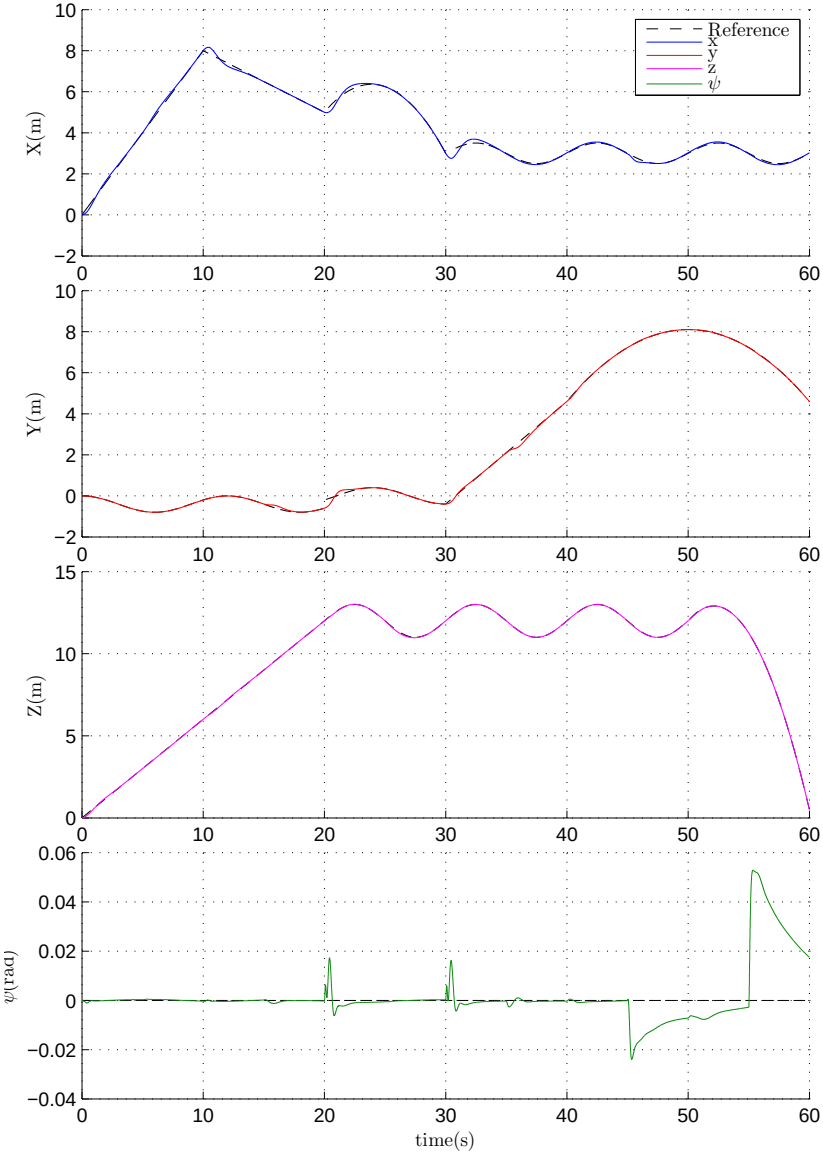


Figure 3.11 – Tiltrotor regulated states for the $\mathcal{H}_\infty$ controller.

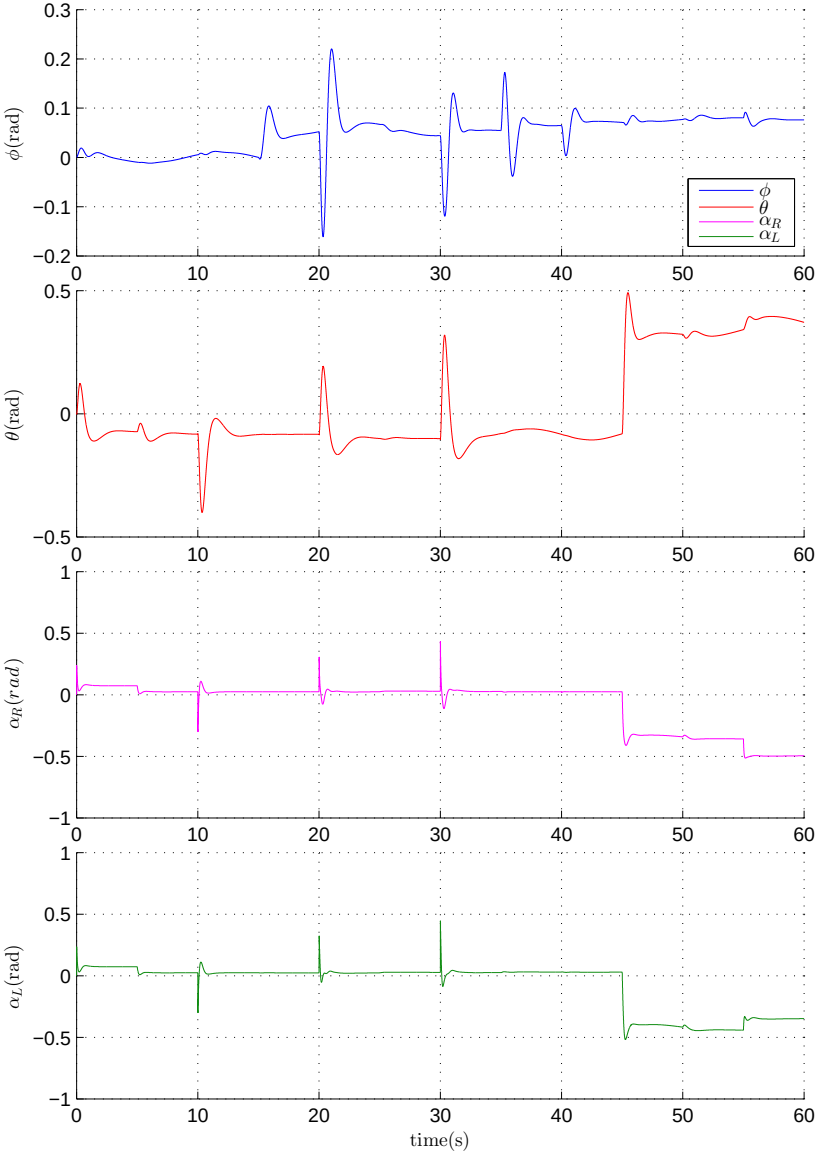


Figure 3.12 – Tiltrotor stabilized states for the $\mathcal{H}_\infty$ controller.

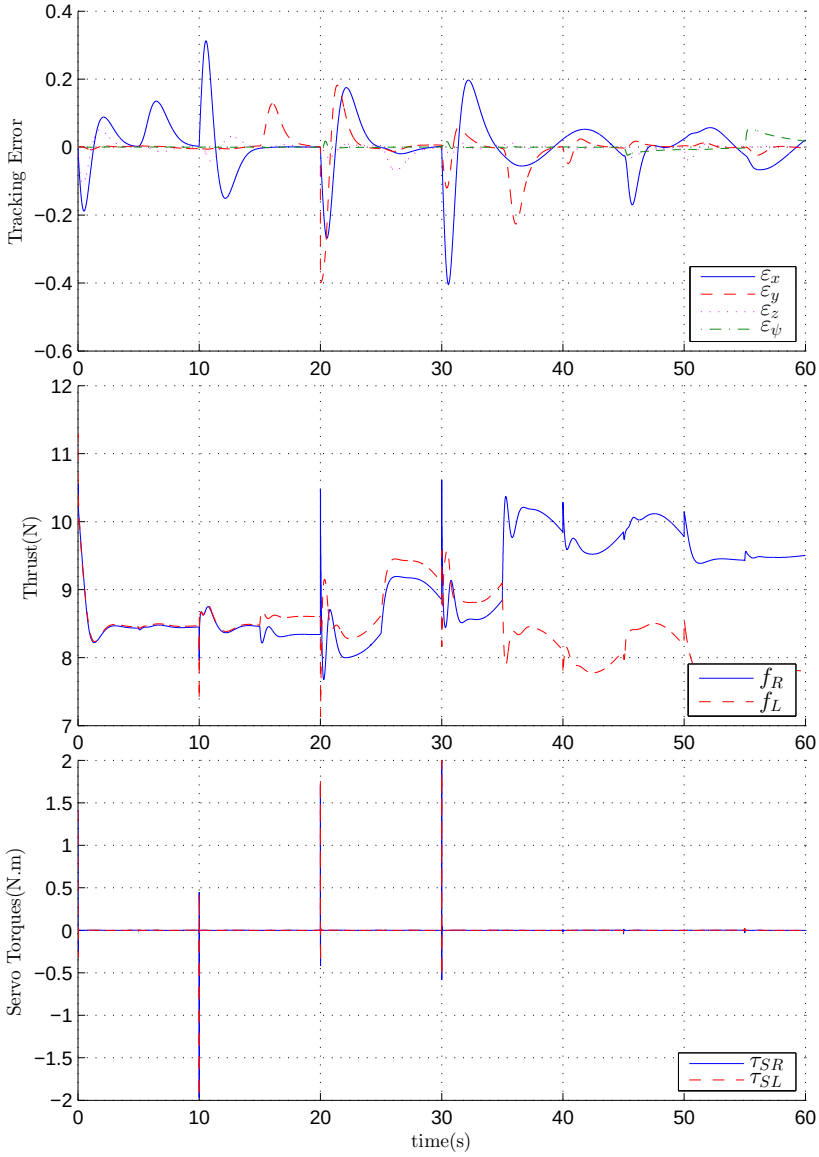


Figure 3.13 – Tiltrotor tracking error and control inputs for the $\mathcal{H}_\infty$ controller.

	$\mathcal{H}_\infty$
ISE_x	0.4305
ISE_y	0.1790
ISE_z	0.0212
ISE_ψ	0.0072

Table 3.5 – Integral Squared Error for the $\mathcal{H}_\infty$ controller.

	$\mathcal{H}_\infty$
$IAVU\ f_R$	542.248
$IAVU\ f_L$	505.951
$IAVU\ \tau_{SR}$	0.0257
$IAVU\ \tau_{SL}$	0.0250

Table 3.6 – Integrated Absolute Variation of the Control signal for the $\mathcal{H}_\infty$ controller.

Despite minimizing the disturbances influence over the system, analyzing the ISE index shows that the $\mathcal{H}_\infty$ controller has a worse performance than the previous controllers. A more exhaustive tuning process of the controller can improve its performance, however the $\mathcal{H}_\infty$ does not present an intuitive adjustment of the parameters. The next controller seeks to maintain its property of disturbance minimization of the $\mathcal{H}_\infty$ and improve its performance by minimizing the system's $\mathcal{H}_2$ norm.

3.1.4 Mixed $\mathcal{H}_2/\mathcal{H}_\infty$ control

This section is mainly based on (TROFINO et al., 2003).

Consider the linear system (3.42) with $\boldsymbol{D_{wz}} = 0$. The $\mathcal{H}_2$ feedback controller is an optimal control that exponentially stabilizes the system and minimizes its $\mathcal{H}_2$ norm:

$$||\boldsymbol{h}_{wz}(t)||_2^2 = \sum_{i,j} \int_0^\infty h_{wz_{ij}}(t)^2 dt = \int_0^\infty tr\{\boldsymbol{h}_{wz}(t)\boldsymbol{h}'_{wz}(t)\}dt\ , \qquad (3.53)$$

where $\boldsymbol{h}_{wz}(t) = \mathfrak{L}^{-1}[\boldsymbol{H}_{wz}(s)]$, $\boldsymbol{H}_{wz}(s)$ being the transfer function between $\boldsymbol{w}(t)$ and $\boldsymbol{z}(t)$, and $tr\{.\}$ is the trace operator. The $\mathcal{H}_2$ norm

can be seen as the energy of the output signal of the system when the system is excited with Dirac delta functions.

The transfer function for the system in (3.42) is:

$$\boldsymbol{H_{wz}}(s) = \boldsymbol{C_z}(s\boldsymbol{I_n} - (\boldsymbol{A}(\alpha) + \boldsymbol{B}(\alpha)\boldsymbol{K}))^{-1}\boldsymbol{B_w} \quad , \tag{3.54}$$

and applying the inverse Laplace transformation:

$$\boldsymbol{h_{wz}}(t) = \mathfrak{L}^{-1}[\boldsymbol{H_{wz}}(s)] = (\boldsymbol{C_z} + \boldsymbol{D_{uz}}\boldsymbol{K})e^{(\boldsymbol{A}(\alpha)+\boldsymbol{B}(\alpha)\boldsymbol{K})t}\boldsymbol{B_w} \quad . \tag{3.55}$$

Using equation (3.55) with (3.53):

$$\begin{aligned} ||\boldsymbol{h_{wz}}(t)||_2^2 &= \int_0^\infty tr\{(\boldsymbol{C_z} + \boldsymbol{D_{uz}}\boldsymbol{K})e^{(\boldsymbol{A}(\alpha)+\boldsymbol{B}(\alpha)\boldsymbol{K})t}\boldsymbol{B_w}\boldsymbol{B_w}' \\ &\quad e^{(\boldsymbol{A}(\alpha)+\boldsymbol{B}\boldsymbol{K})'t}(\boldsymbol{C_z} + \boldsymbol{D_{uz}}\boldsymbol{K})'\}dt \\ &\quad = tr\{(\boldsymbol{C_z} + \boldsymbol{D_{uz}}\boldsymbol{K})(\int_0^\infty e^{(\boldsymbol{A}(\alpha)+\boldsymbol{B}\boldsymbol{K})t}\boldsymbol{B_w}\boldsymbol{B_w}' \\ &\quad e^{(\boldsymbol{A}(\alpha)+\boldsymbol{B}\boldsymbol{K})'t}dt)(\boldsymbol{C_z} + \boldsymbol{D_{uz}}\boldsymbol{K})'\} \\ &\quad = tr\{(\boldsymbol{C_z} + \boldsymbol{D_{uz}}\boldsymbol{K})\boldsymbol{P_c}(\boldsymbol{C_z} + \boldsymbol{D_{uz}}\boldsymbol{K})'\} \quad , \end{aligned} \tag{3.56}$$

where $\boldsymbol{P_c}$ is the controllability Gramian

$$\boldsymbol{P_c} = \int_0^\infty e^{(\boldsymbol{A}(\alpha)+\boldsymbol{B}\boldsymbol{K})t}\boldsymbol{B_w}\boldsymbol{B_w}'e^{(\boldsymbol{A}(\alpha)+\boldsymbol{B}\boldsymbol{K})'t}dt \tag{3.57}$$

and, as all the eigenvalues of $(\boldsymbol{A}(\alpha) + \boldsymbol{B}\boldsymbol{K})$ are negative (we assume the state feedback control stabilizes the system), is the unique solution to (CHEN, 1998)

$$\boldsymbol{A}\boldsymbol{P_c} + \boldsymbol{P_c}\boldsymbol{A}' + \boldsymbol{B_w}\boldsymbol{B_w}' = 0 \quad . \tag{3.58}$$

To work with LMIs and allow uncertainties in the system to be easily incorporated, instead of finding matrix P_c that solves (3.58) we will search for a matrix Q, where $Q > P_c$, that solves:

$$\boldsymbol{A}(\alpha)\boldsymbol{Q} + \boldsymbol{Q}\boldsymbol{A}'(\alpha) + \boldsymbol{B_w}\boldsymbol{B_w}' < 0 \quad . \tag{3.59}$$

Finding Q in the above equation gives the upper bound of the $\mathcal{H}_2$ norm

$$tr\{(\boldsymbol{C_z} + \boldsymbol{D_{uz}}\boldsymbol{K})\boldsymbol{Q}(\boldsymbol{C_z} + \boldsymbol{D_{uz}}\boldsymbol{K})'\} > tr\{(\boldsymbol{C_z} + \boldsymbol{D_{uz}}\boldsymbol{K})\boldsymbol{P_c}(\boldsymbol{C_z} + \boldsymbol{D_{uz}}\boldsymbol{K})'\}. \tag{3.60}$$

To make the problem convex, an auxiliary matrix N is used, where (TROFINO et al., 2003):

$$N - tr\{(\boldsymbol{C_z} + \boldsymbol{D_{uz}}\boldsymbol{K})\boldsymbol{Q}\boldsymbol{Q}^{-1}\boldsymbol{Q}(\boldsymbol{C_z} + \boldsymbol{D_{uz}}\boldsymbol{K})'\} > 0 \quad , \tag{3.61}$$

and the original problem of minimizing $tr\{(\boldsymbol{C_z}+\boldsymbol{D_{uz}}\boldsymbol{K})\boldsymbol{Q}(\boldsymbol{C_z}+\boldsymbol{D_{uz}}\boldsymbol{K})'\}$ is now transformed to minimize matrix N. Applying the Schur complement to equation (3.61) yields:

$$\begin{bmatrix} \boldsymbol{N} & \boldsymbol{C_z} + \boldsymbol{D_{uz}}\boldsymbol{K}\boldsymbol{Q} \\ \boldsymbol{Q}(\boldsymbol{C_z'} + \boldsymbol{K'}\boldsymbol{D_{uz}'}) & \boldsymbol{Q} \end{bmatrix} > 0 \quad . \tag{3.62}$$

Finally, employing the transformation $Y = KQ$ to equations (3.62) and (3.59) gives the LMIs to find a gain matrix $K = YQ^{-1}$ that minimizes the $\mathcal{H}_2$ norm by minimizing $tr\{\boldsymbol{N}\}$ subject to:

$$\begin{bmatrix} \boldsymbol{N} & \boldsymbol{C_z}\boldsymbol{Q} + \boldsymbol{D_{uz}}\boldsymbol{Y} \\ \boldsymbol{Q}\boldsymbol{C_z'} + \boldsymbol{Y'}\boldsymbol{D_{uz}'} & \boldsymbol{Q} \end{bmatrix} > 0 \tag{3.63}$$

$$\boldsymbol{Q}\boldsymbol{A}'(\alpha) + \boldsymbol{A}(\alpha)\boldsymbol{Q} + \boldsymbol{Y'}\boldsymbol{B'} + \boldsymbol{B}\boldsymbol{Y} + \boldsymbol{B_w}\boldsymbol{B_w'} < 0 \quad . \tag{3.64}$$

Minimizing the $\mathcal{H}_2$ norm brings a better transient response for the system. This is used to improve the controller presented in section 3.1.3 by creating a multi-objective controller that minimizes the $\mathcal{H}_2$ norm while using a fixed value for $\tilde{\gamma}$ that is close to the previously found for the pure $\mathcal{H}_\infty$ control. The multi-objective $\mathcal{H}_2/\mathcal{H}_\infty$ controller can be synthesized using (TROFINO et al., 2003):

$$\text{minimize trace}(\boldsymbol{N}) \text{ subject to (3.63) and (3.49).}$$

Note that the LMI (3.64) is already included in (3.49).

The control matrices are designed as it would be for a pure $\mathcal{H}_2$ controller and follows the same logic as in section 3.1.2, reusing matrices C_z and D_z depicted in (3.37) and (3.36).

After some iterations the fixed value $\tilde{\gamma} = 4$ was chosen , which means that the $\mathcal{H}_\infty$ norm for this controller is slightly worse ($||\boldsymbol{H_{wz}}(s)||_\infty < 2$).

Once again the constraint in control action LMIs are used, employing (3.51) and (3.52).

The gain matrix is found by minimizing $tr\{\boldsymbol{N}\}$ subject to (3.63), (3.49), (3.51) and (3.52) using MATLAB's Robust Control Toolbox version 4.3 (see appendix B, equation (B.12)). The result is $tr\{\boldsymbol{Z}\} = 4.0738$ and $||\boldsymbol{H_{wz}}(s)||_2 < 2.0184$.

Simulations for this controller were made under the same conditions as used in section 3.1.1, the results being depicted in figures 3.14-3.17 and tables 3.7 for the ISE index and 3.8 for the IAVU.

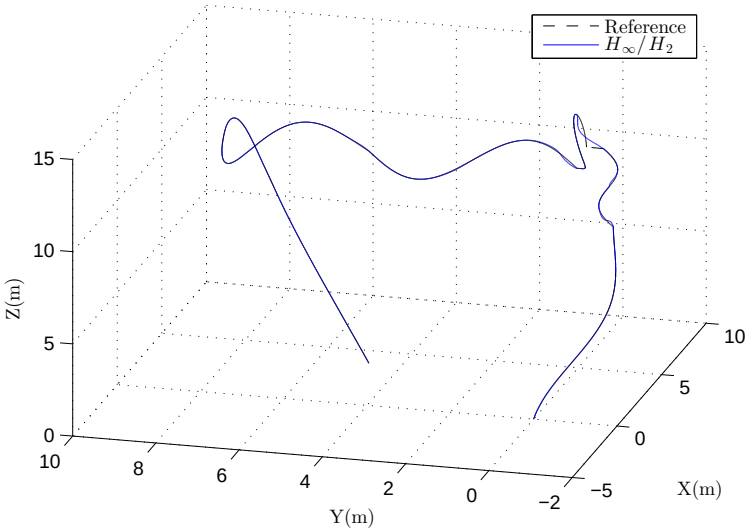


Figure 3.14 – Tiltrotor trajectory in space for the $\mathcal{H}_2/\mathcal{H}_\infty$ controller.

	$\mathcal{H}_2/\mathcal{H}_\infty$
ISE_x	0.0637
ISE_y	0.0664
ISE_z	0.0021
ISE_ψ	0.0198

Table 3.7 – Integral Squared Error for the $\mathcal{H}_2/\mathcal{H}_\infty$ controller.

The $\mathcal{H}_2/\mathcal{H}_\infty$ controller results show an improvement in performance over the $\mathcal{H}_\infty$ controller while maintain a similar $\mathcal{H}_\infty$ norm,

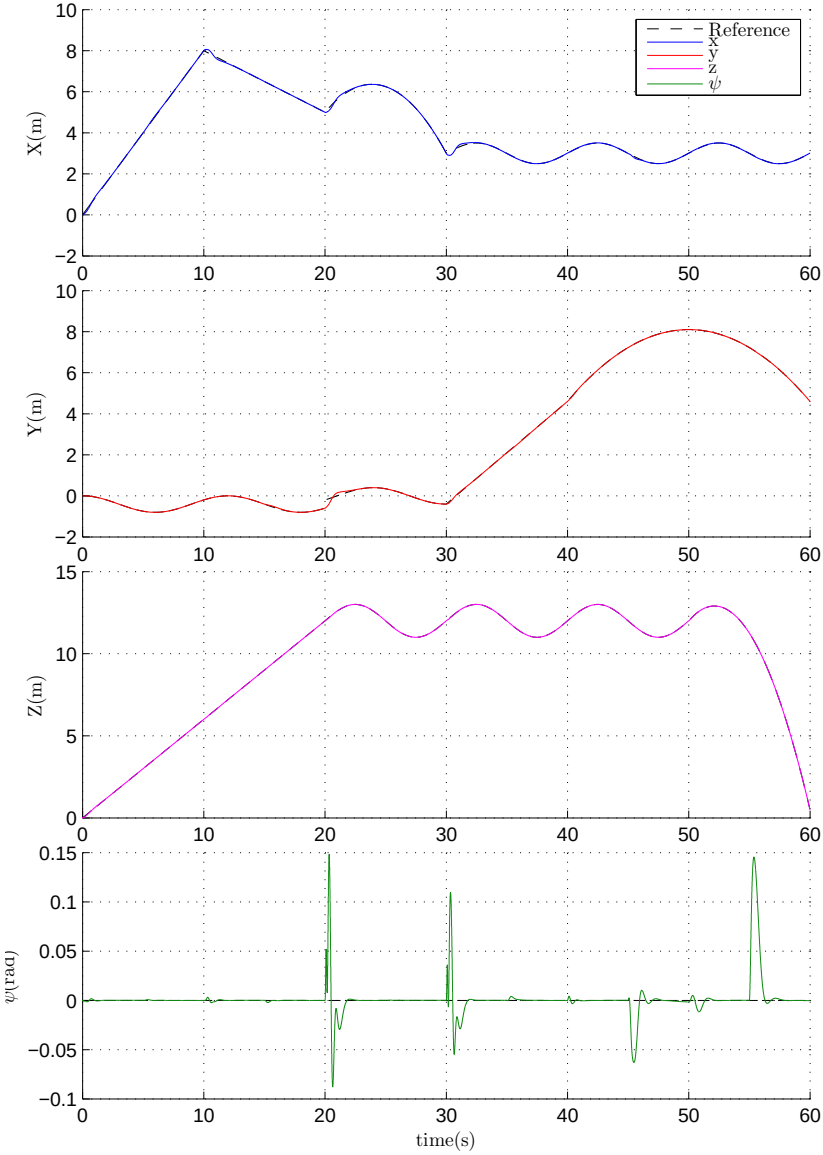


Figure 3.15 – Tiltrotor regulated states for the $\mathcal{H}_2/\mathcal{H}_\infty$ controller.

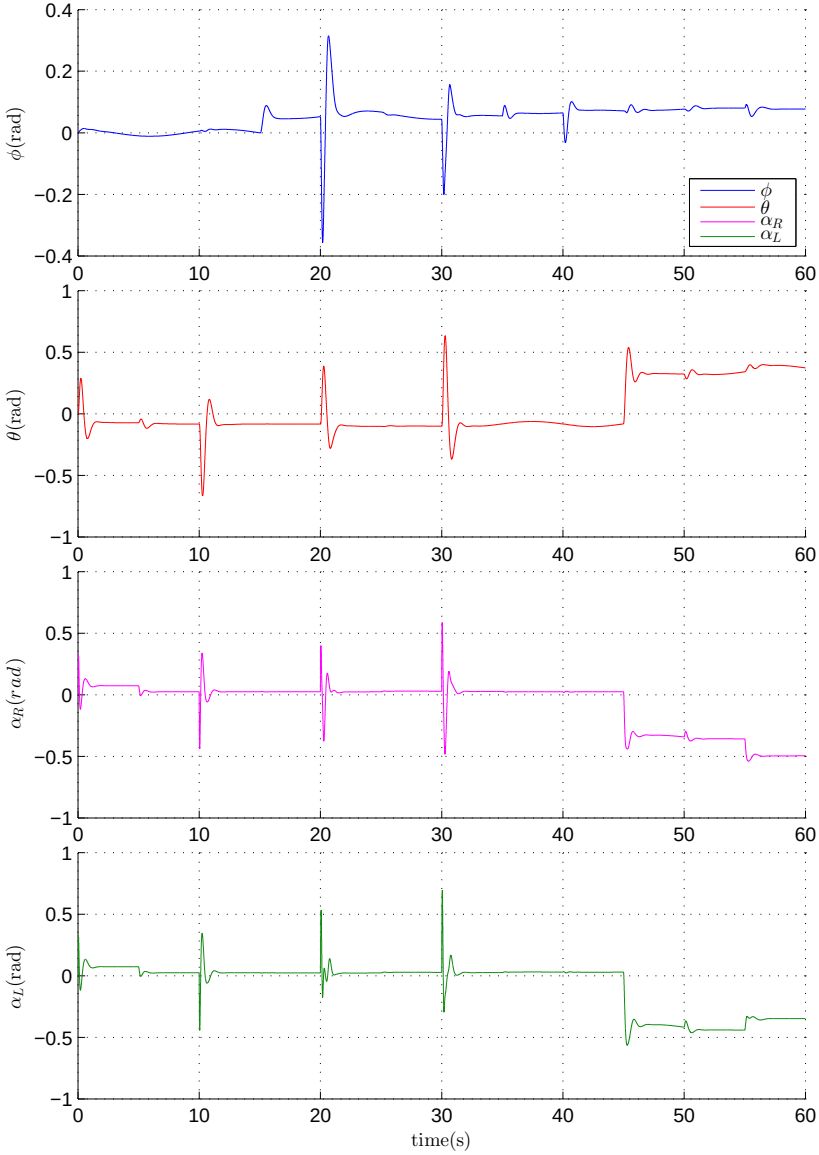


Figure 3.16 – Tiltrotor stabilized states for the $\mathcal{H}_2/\mathcal{H}_\infty$ controller.

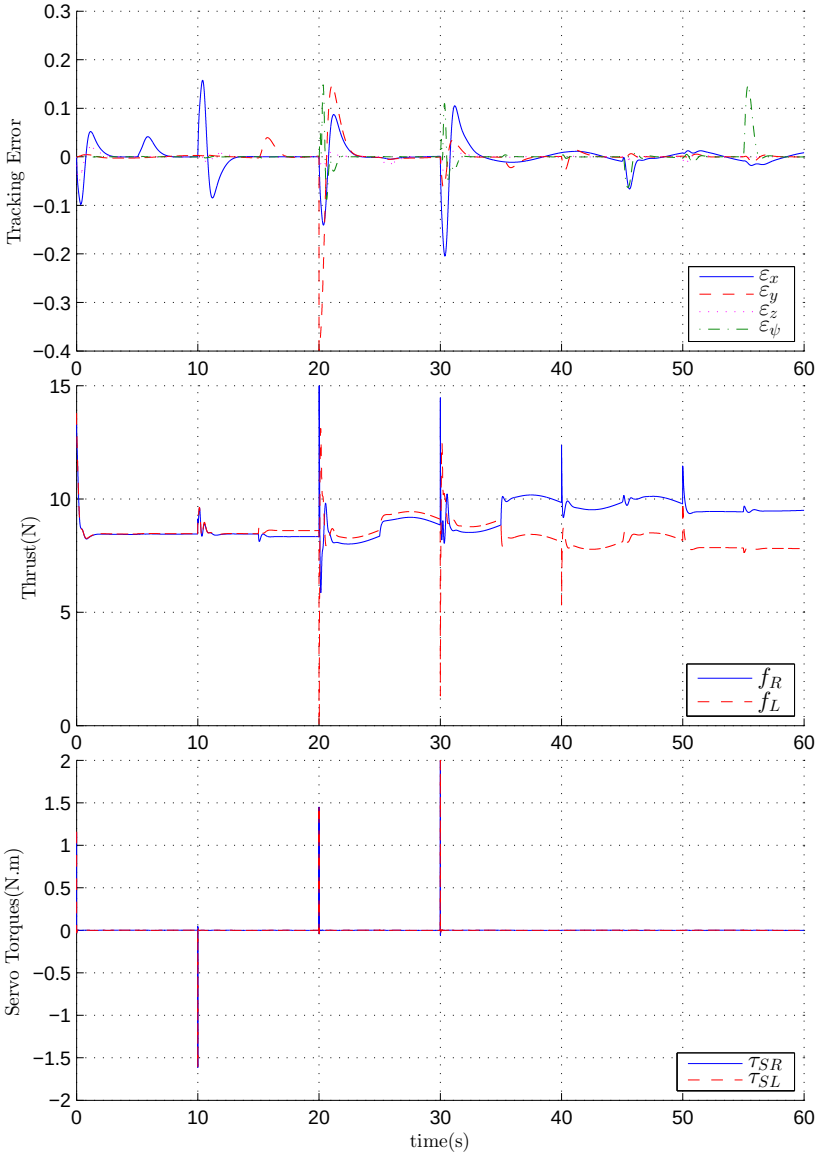


Figure 3.17 – Tiltrotor tracking error and control inputs for the $\mathcal{H}_2/\mathcal{H}_\infty$ controller.

	$\mathcal{H}_2/\mathcal{H}_\infty$
$IAVU\ f_R$	542.660
$IAVU\ f_L$	506.596
$IAVU\ \tau_{SR}$	0.0137
$IAVU\ \tau_{SL}$	0.0147

Table 3.8 – Integrated Absolute Variation of the Control signal for the $\mathcal{H}_2/\mathcal{H}_\infty$ controller.

which was the objective of this controller.

Considering that the controller has an increased robustness than the LQR LTI controller because it includes the time varying nature of the system as well as its control signals saturation into the control design, and that the ISE and IAVU indices for the $\mathcal{H}_2/\mathcal{H}_\infty$ controller are close to the ones found for the LQR LTI controller, we can then conclude that the $\mathcal{H}_2/\mathcal{H}_\infty$ also presents an improvement over the LQR LTI controller.

3.1.5 LQR control for linear time varying systems

The LQR considering the time varying system (3.9) with a state feedback controller can be calculated in different ways, the most common being: (i) through the Ricatti Differential Equation (RDE); (ii) through calculation of the corresponding ARE for each time instant.

The first method can be deduced from realizing that in equation (3.13), $\dot{\boldsymbol{P}}(t)$ is not equal to zero in the LTV case. Using similar manipulations, it is trivial to arrive at the RDE. The use of such method should be made with care as to respect its existence conditions (LEDYAEV, 2011).

For this work the second method is used, which can only be applied because all time variant variables of the system are known at any instant of time, allowing to compute matrix $\boldsymbol{A}(t)$ online. This way, a different set of LTI matrices for each time instant and the instant gain matrix $\boldsymbol{K}$ for each particular set can be computed with its ARE, in the the same manner as in section 3.1.1.

If the computational cost of the ARE is too high for the em-

bedded system used, one can consider to calculate all the gain matrices beforehand. This is possible if all the trajectories are known before the aircraft takes flight, since the trajectory's accelerations are the unique requirement to compute matrix $\boldsymbol{A}(t)$ at a given time. Therefore, the gain matrices can be computed offline for each time interval of the trajectory, transforming the controller into a gain-scheduling controller.

Simulations for this controller were made under the same conditions as used in section 3.1.1, the results being depicted in figures 3.18-3.21 and tables 3.9 for the ISE index and 3.10 for the IAVU.

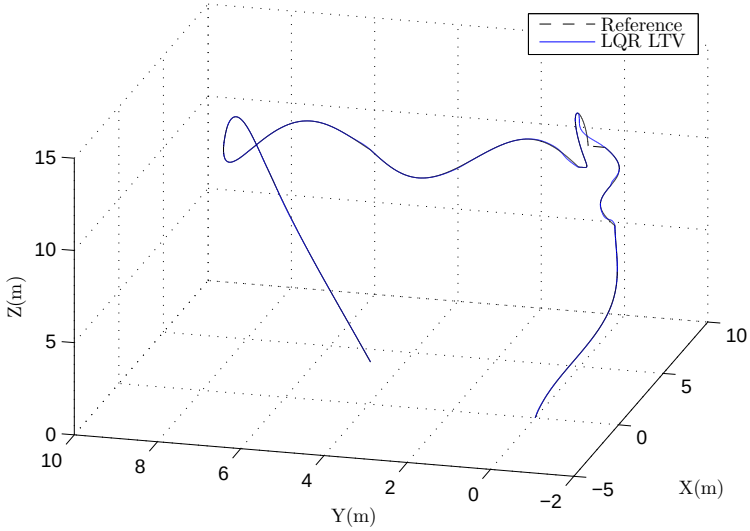


Figure 3.18 – Tiltrotor trajectory in space for the LQR LTV controller.

	LQR LTV
ISE_x	0.0564
ISE_y	0.0651
ISE_z	0.0012
ISE_ψ	0.0226

Table 3.9 – Integral Squared Error for the LQR LTV controller.

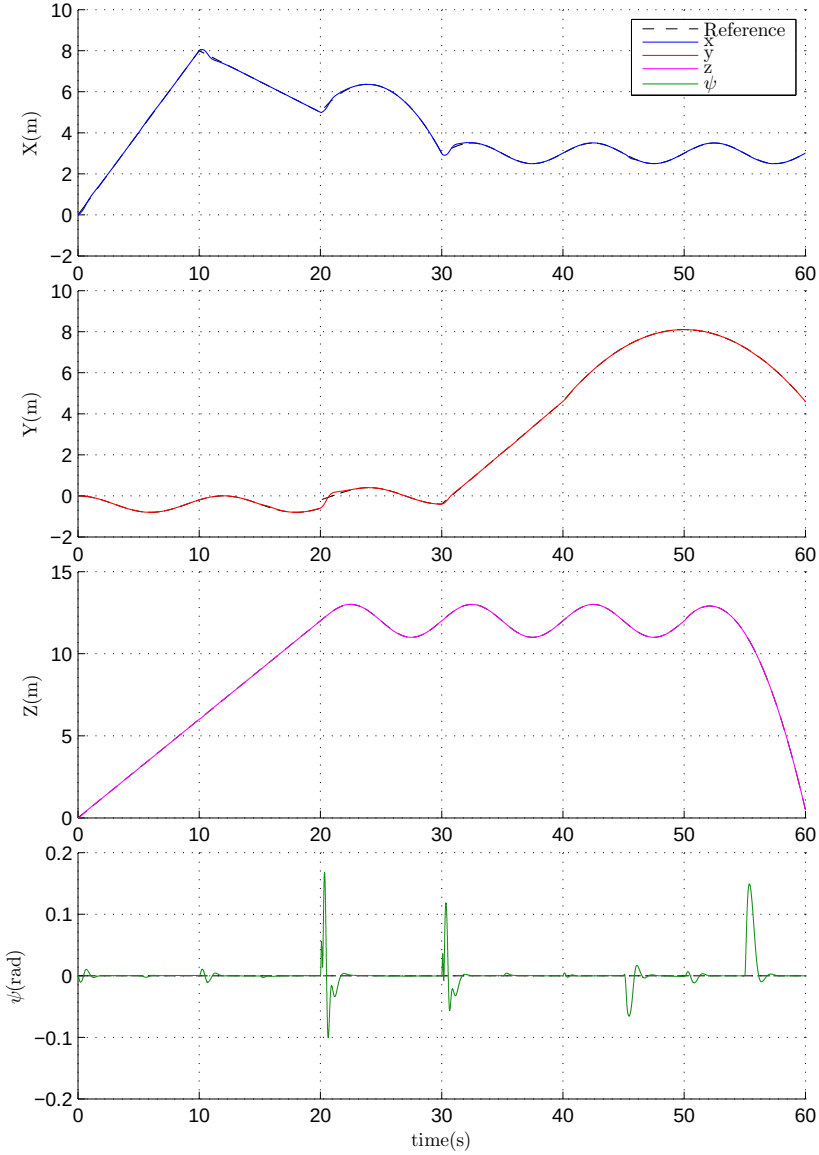


Figure 3.19 – Tiltrotor regulated states for the LQR LTV controller.

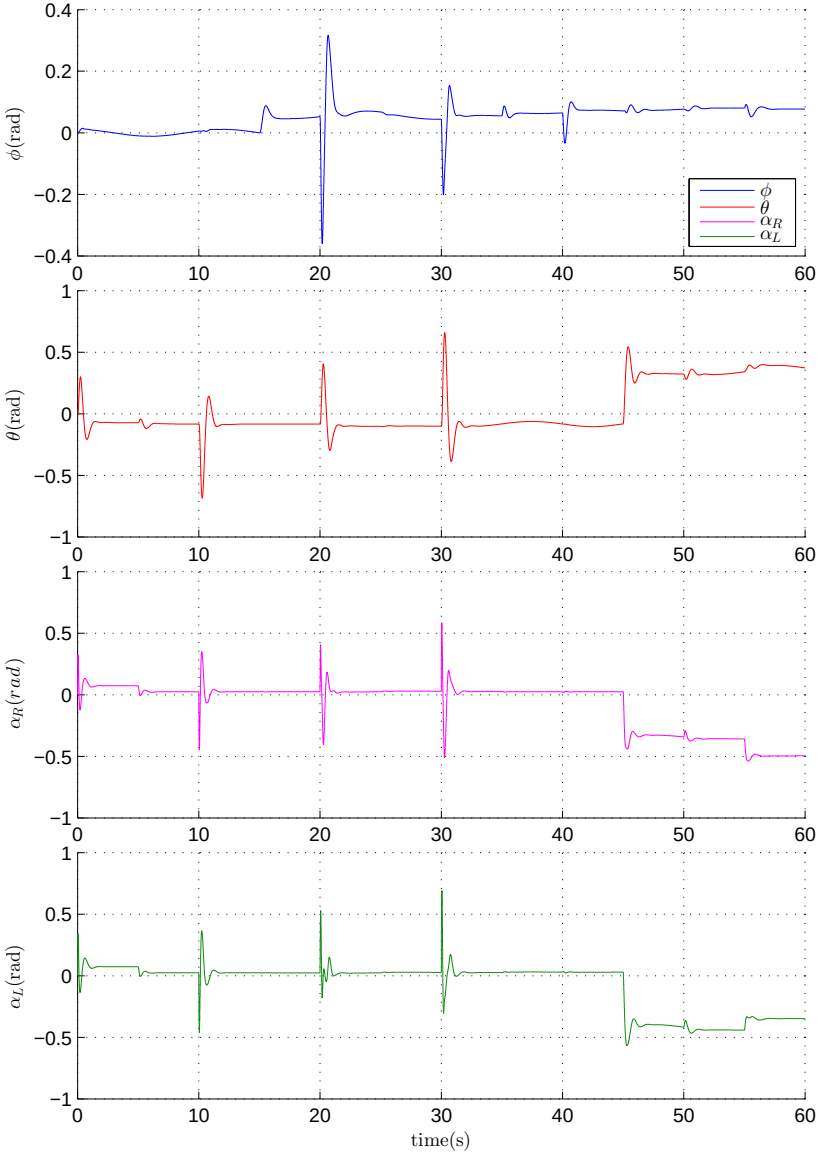


Figure 3.20 – Tiltrotor stabilized states for the LQR LTV controller.

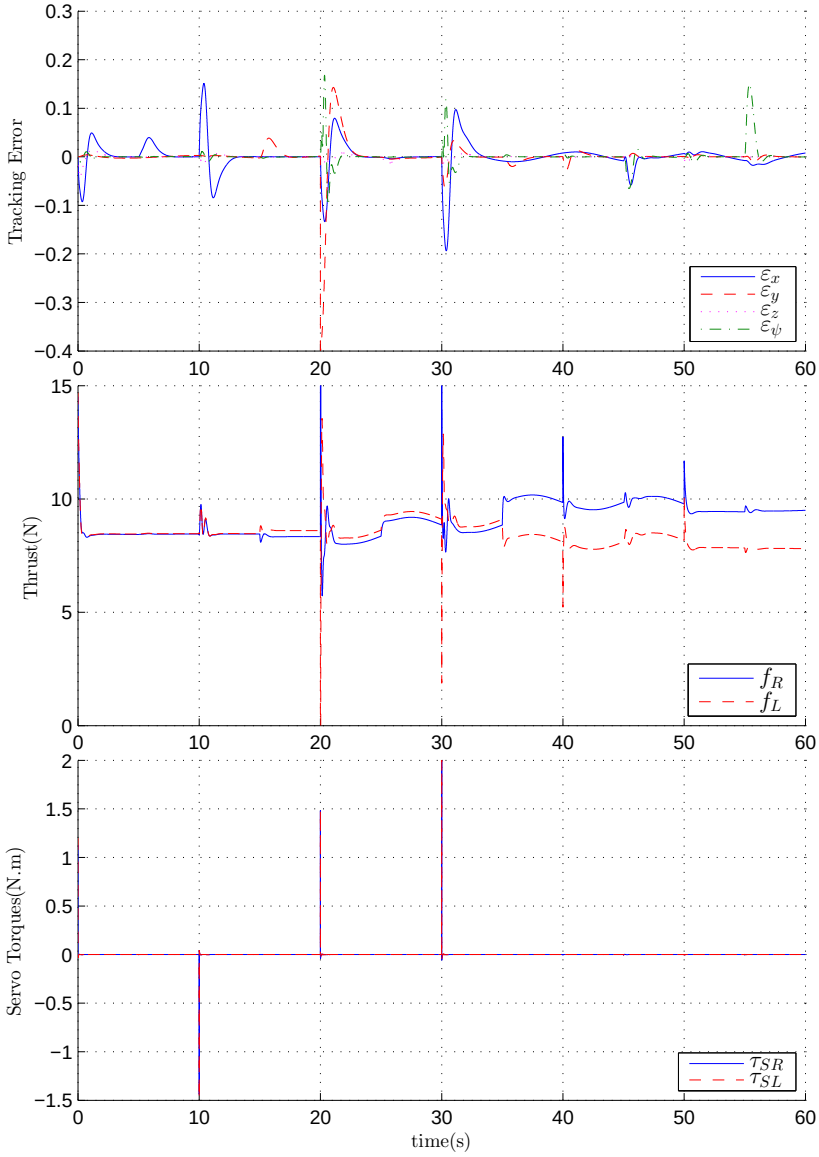


Figure 3.21 – Tiltrotor tracking error and control inputs for the LQR LTV controller.

	LQR LTV
$IAVU\ f_R$	542.667
$IAVU\ f_L$	506.615
$IAVU\ \tau_{SR}$	0.0136
$IAVU\ \tau_{SL}$	0.0147

Table 3.10 – Integrated Absolute Variation of the Control signal for the LQR LTV controller.

While the LMI controllers use the time varying nature of the system to improve the controllers robustness, the LQR LTV controller uses the data in real time to improve its performance. The main idea for the simulation of the LQR LTV controller was to compare and evaluate how much the performance would improve by using the time varying data.

Analyzing the indices shows that the performance are indeed better and that the control is smoother when compared to the other developed controllers, but only by a small margin. This means that, even in a trajectory with so many abrupt changes, the time varying parameters have little influence in the controlled system.

3.2 Backstepping nonlinear control

In order to compare the developed controllers to the literature, an adaptation of the controller published in (CHOWDHURY et al., 2012) is used. This work is chosen because the controller structure is one of the most established for UAVs in general and its definitions are close to the the ones performed in this work, simplifying its study and analysis.

It implements a feedback linearization for a simplified version of the nonlinear model (2.62) with $q = \begin{bmatrix} x & y & z & \phi & \theta & \psi \end{bmatrix}'$ as generalized coordinates. The equations of motion for the translational motion are:

$$m\ddot{x} = (c\psi c\theta)F_{xb} + (c\psi s\theta c\phi + s\psi s\phi)F_{zb} \tag{3.65}$$

$$m\ddot{y} = (s\psi c\theta)F_{xb} + (s\psi s\theta c\phi - c\psi s\phi)F_{zb} \tag{3.66}$$

$$m\ddot{z} = -s\theta F_{xb} + c\theta c\phi F_{zb} - mg \quad , \tag{3.67}$$

and the rotational equations of motion:

$$\boldsymbol{M}(\boldsymbol{\Theta})\ddot{\boldsymbol{\Theta}} + C(\boldsymbol{\Theta}, \dot{\boldsymbol{\Theta}})\dot{\boldsymbol{\Theta}} = \boldsymbol{\tau} \ , \ \boldsymbol{\Theta} = \begin{bmatrix} \phi & \theta & \psi \end{bmatrix} \quad , \tag{3.68}$$

with the rotational inertia matrix $M(\Theta)$ given by:

$$\boldsymbol{M}(\boldsymbol{\Theta}) = \begin{bmatrix} I_{xx} & 0 & -I_{xx}S\theta \\ 0 & I_{yy}C^2\phi + I_{zz}S^2\phi & (I_{yy} - I_{zz})C\phi S\phi C\theta \\ -I_{xx}S\theta & (I_{yy} - I_{zz})C\phi S\phi C\theta & I_{xx}S^2\theta + I_{yy}C^2\theta S^2\phi + I_{zz}C^2\theta C^2\phi \end{bmatrix} \tag{3.69}$$

and the Coriolis and centrifugal forces matrix $\boldsymbol{C}(\boldsymbol{\Theta}, \dot{\boldsymbol{\Theta}})$ found with equation (2.63); This model can be derived from the model found in section 2 from several simplifications:

- consider the system as only one rigid body, that is, omitting the generalized coordinates α_R and α_L.
- Consider the center of mass coincident with the geometrical center, i.e, frame $\mathfrak{C}_1$ is coincident with frame $\mathfrak{B}$.
- Lateral fixed tilt angle β equals zero.

The body forces are:

$$\boldsymbol{F}^{\mathfrak{B}}(\boldsymbol{q}) = \begin{bmatrix} F_x^{\mathfrak{B}} \\ F_y^{\mathfrak{B}} \\ F_z^{\mathfrak{B}} \\ \tau_\phi^{\mathfrak{B}} \\ \tau_\theta^{\mathfrak{B}} \\ \tau_\psi^{\mathfrak{B}} \end{bmatrix} = \begin{bmatrix} f_R sin(\alpha_R) + f_L sin(\alpha_L) \\ 0 \\ f_R cos(\alpha_R) + f_L cos(\alpha_L) \\ [f_L cos(\alpha_L) - f_R cos(\alpha_R)]l \\ [f_L sin(\alpha_L) + f_R sin(\alpha_R)]h \\ [f_R sin(\alpha_R) - f_L sin(\alpha_L)]l \end{bmatrix} \quad . \tag{3.70}$$

The inverse dynamics control objective is to find a nonlinear feedback control law that, when substituted into the model equations (3.65) to (3.68), results in the linear closed-loop system:

$$\ddot{x} = \zeta_1 = -K_{px}(x - x_r) - K_{vx}(\dot{x} - \dot{x_r}) + \ddot{x}_r \tag{3.71}$$

$$\ddot{y} = \zeta_2 = -K_{py}(y - y_r) - K_{vy}(\dot{y} - \dot{y_r}) + \ddot{y}_r \tag{3.72}$$

$$\ddot{z} = \zeta_3 = -K_{pz}(z - z_r) - K_{vz}(\dot{z} - \dot{z}_r) + \ddot{z}_r \tag{3.73}$$

$$\ddot{\phi} = \gamma_1 = -K_{p\phi}(\phi - \phi_r) - K_{v\phi}(\dot{\phi} - \dot{\phi}_r) + \ddot{\phi}_r \tag{3.74}$$

$$\ddot{\theta} = \gamma_2 = -K_{p\theta}(\theta - \theta_r) - K_{v\theta}(\dot{\theta} - \dot{\theta}_r) + \ddot{\theta}_r \tag{3.75}$$

$$\ddot{\psi} = \gamma_3 = -K_{p\phi}(\psi - \psi_r) - K_{v\psi}(\dot{\psi} - \dot{\psi}_r) + \ddot{\psi}_r \quad , \tag{3.76}$$

where ζ_1, ζ_2, ζ_3, γ_1, γ_2 and γ_3 are the new dynamics chosen for the resulting linear system and K_{px}, K_{vx}, K_{py}, K_{vy}, K_{pz}, K_{vz}, $K_{p\phi}$, $K_{v\phi}$, $K_{p\theta}$, $K_{v\theta}$, $K_{p\psi}$, $K_{v\psi}$ are gains to be designed.

The forces and torques expressions chosen to transform the system dynamics into equations (3.71) to (3.76) are:

$$F_{zb} = \frac{mg + m\zeta_3 + S\theta F_{xb}}{C\phi C\theta} \tag{3.77}$$

$$F_{xb} = \frac{m\zeta_1}{C\psi C\theta} \tag{3.78}$$

$$\boldsymbol{\tau} = \begin{bmatrix} \tau_\phi \\ \tau_\theta \\ \tau_\psi \end{bmatrix} = \boldsymbol{M}(\boldsymbol{\Theta}) \begin{bmatrix} \gamma_1 \\ \gamma_2 \\ \gamma_3 \end{bmatrix} + \boldsymbol{C}(\boldsymbol{\Theta}, \dot{\boldsymbol{\Theta}}) \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} \quad , \tag{3.79}$$

with the intermediate references ϕ_d and θ_d:

$$\phi_d = arcsin\left(\frac{m\zeta_1 S\psi - m\zeta_2 C\psi}{F_{zb}}\right) \tag{3.80}$$

$$\theta_d = arcsin\left(-\frac{S\phi S\psi}{C\phi C\psi}\right) \quad . \tag{3.81}$$

The final step transform the computed forces and torques to the applied control signals:

$$\begin{aligned} f_R &= \tfrac{1}{2}\sqrt{(F_{xb} + \tfrac{\tau_\psi}{l})^2 + (-F_{zb} + \tfrac{\tau_\phi}{l})^2} \\ f_L &= \tfrac{1}{2}\sqrt{(-F_{xb} + \tfrac{\tau_\psi}{l})^2 + (F_{zb} + \tfrac{\tau_\phi}{l})^2} \\ \alpha_R &= arctan\left(\frac{F_{xb} + \frac{\tau_\psi}{l}}{F_{zb} - \frac{\tau_\phi}{l}}\right) \\ \alpha_L &= arctan\left(\frac{F_{xb} - \frac{\tau_\psi}{l}}{F_{zb} + \frac{\tau_\phi}{l}}\right) \end{aligned} \quad . \tag{3.82}$$

Analyzing equation (3.82) it can be noted that τ_θ, calculated in (3.79), does not appear anywhere in the equations. This means that the variable θ is not actually being controlled, which is highly detrimental.

For our work we decided to make changes to the original controller in order to include τ_θ into (3.82). First from (3.70) we have that $F_{xb} = \tau_\theta / h$. Substituting in (3.82):

$$\begin{aligned} f_R &= \tfrac{1}{2}\sqrt{(\tfrac{\tau_\theta}{h} + \tfrac{\tau_\psi}{l})^2 + (-F_{zb} + \tfrac{\tau_\phi}{l})^2} \\ f_L &= \tfrac{1}{2}\sqrt{(-\tfrac{\tau_\theta}{h} + \tfrac{\tau_\psi}{l})^2 + (F_{zb} + \tfrac{\tau_\phi}{l})^2} \\ \alpha_R &= arctan\left(\frac{\frac{\tau_\theta}{h} + \frac{\tau_\psi}{l}}{F_{zb} - \frac{\tau_\phi}{l}}\right) \\ \alpha_L &= arctan\left(\frac{\frac{\tau_\theta}{h} - \frac{\tau_\psi}{l}}{F_{zb} + \frac{\tau_\phi}{l}}\right) \end{aligned} \quad . \tag{3.83}$$

Now we included the control τ_θ, namely γ_2, into the equation but removing F_{xb} also excluded the x control ζ_1. To rectify this we take the control out of $F_x b$ in equation (3.78) and include it into θ_d in equation (3.81). The result is:

$$F_{xb} = \frac{1}{C\psi C\theta} \tag{3.84}$$

$$\theta_d = \frac{m\zeta_1 - 1}{F_{zb} C\psi C\phi} - \frac{S\psi S\phi}{C\psi C\phi} \quad . \tag{3.85}$$

Note that the new equations still obey the feedback linearization and equations (3.71) to (3.76) remain unchanged. Regarding these equations, we decided that the addition of an integral action is important to obtain a satisfactory path tracking. The resulting system dynamics are redefined to:

$$\ddot{x} = \bar{\zeta_1} = -K_{px}(x - x_r) - K_{vx}(\dot{x} - \dot{x_r}) - K_{ix}\int (x - x_r) + \ddot{x}_r \tag{3.86}$$

$$\ddot{y} = \bar{\zeta_2} = -K_{py}(y - y_r) - K_{vy}(\dot{y} - \dot{y_r}) - K_{iy}\int (y - y_r) + \ddot{y}_r \tag{3.87}$$

$$\ddot{z} = \bar{\zeta_3} = -K_{pz}(z - z_r) - K_{vz}(\dot{z} - \dot{z_r}) - K_{iz}\int (z - z_r) + \ddot{z}_r \tag{3.88}$$

$$\ddot{\psi} = \bar{\gamma_3} = -K_{p\phi}(\psi - \psi_r) - K_{v\psi}(\dot{\psi} - \dot{\psi}_r) - K_{i\psi}\int(\psi - \psi_r) + \ddot{\psi}_r \quad . \quad (3.89)$$

The original article does not include the gains values used. Its results are only shown in figures but it can be observed that its variables settling time is around six seconds. Unfortunately the rotational system's settling time is too high for it too work in practice.

Albeit the results not being satisfactory, the control strategy was considered suited for our problem. Therefore new gains were designed to comply with the following new system requirements:

- settling time (5%) of 0.3 seconds for the roll and pitch angles (ϕ,θ) and 1.5 seconds for the yaw angle (ψ);
- settling time (5%) of 2.5 seconds for the translational system (x,y,z);
- no overshoot for the rotational system and maximum 5% overshoot for the translational system.

Defining the error in x to be $e_x = x - x_r$, equation (3.86) becomes:

$$\ddot{e}_x + K_{vx}\dot{e}_x + K_{px}e_x + K_{ix}\int e_x = 0 \quad . \quad (3.90)$$

Applying the Laplace transform, the above equation in the frequency domain is:

$$s^2 + K_{vx}s + K_{px} + \frac{K_{ix}}{s} = 0 \quad . \quad (3.91)$$

The control gains are designed to fulfill the system requirements according to the classical equation of a second order closed-loop system with the (ξ, w_n) parametrization (OGATA, 1997)

$$s^2 + 2\zeta w_n s + w_n^2 \quad , \quad (3.92)$$

where ζ is the damping ratio and w_n is the natural frequency. Then, the third pole is chosen to be three times faster in order to have a negligible influence in the closed-loop system's response.

From the system requirements for $x(t)$ and (3.92) we have that $\zeta = 0.69$ and $w_n = 1.74$, resulting in the second degree system $(s + (1.2 \pm 1.26i))^2$. Choosing the third pole as $s = 3.6$, we have that

the desired dynamics is:

$$(s + (1.2 \pm 1.26i))^2(s + 3.6) = s^3 + 6s^2 + 11.67s + 10.90 \quad . \qquad (3.93)$$

Comparing (3.93) with (3.91) yields $K_{vx} = 6$, $K_{px} = 11.67$ and $K_{ix} = 10.9$. As $y(t)$ and $z(t)$ have the same resulting dynamics and requirements as $x(t)$, then their gains are the same. Therefore $K_{vz} = K_{vy} = K_{vx}$ and $K_{pz} = K_{py} = K_{px}$.

For the yaw angle system requirements, a double pole s_d is selected to give a 5% settling time of $1.5s$, resulting in $s_d = 3.2$. Applying the same method as with the translation system, the desired dynamics is $(s + 3.2)^2(s + 9.6)$, yielding $K_{v\psi} = 16$, $K_{p\psi} = 71.68$ and $K_{i\psi} = 98.3$.

The roll and pitch angles desired dynamics do not include an integral action, with resulting second degree systems. According to its requirements, the dynamics $(s + 16)^2$ is desired. Comparing to (3.74) gives us that $K_{v\phi} = K_{v\theta} = 32$ and $K_{p\phi} = K_{p\theta} = 256$.

Simulations for this controller were made under the same conditions as used in section 3.1.1, the results being depicted in figures 3.22-3.25 and tables 3.11 for the ISE index and 3.12 for the IAVU.

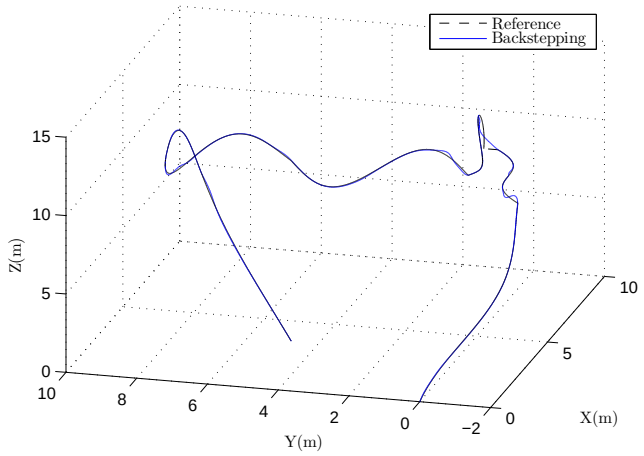


Figure 3.22 – Tiltrotor trajectory in space for the backstepping controller.

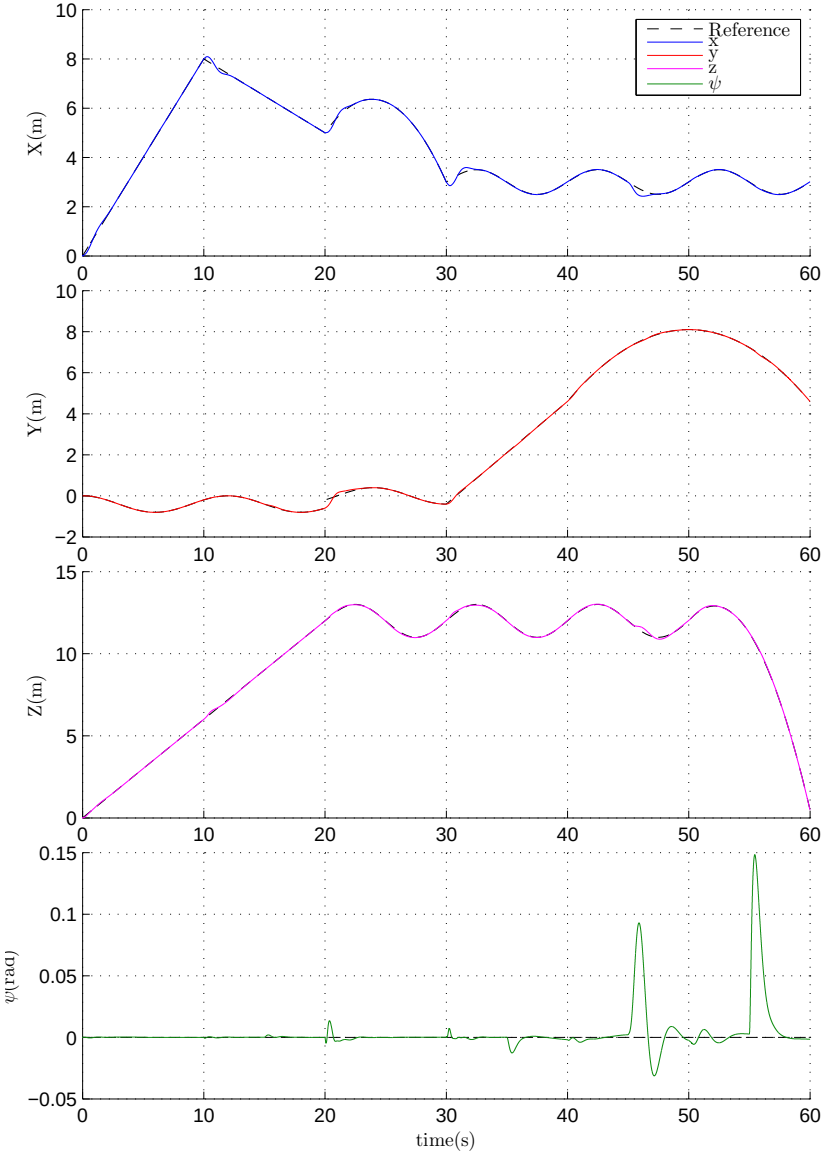


Figure 3.23 – Tiltrotor regulated states for the backstepping controller.

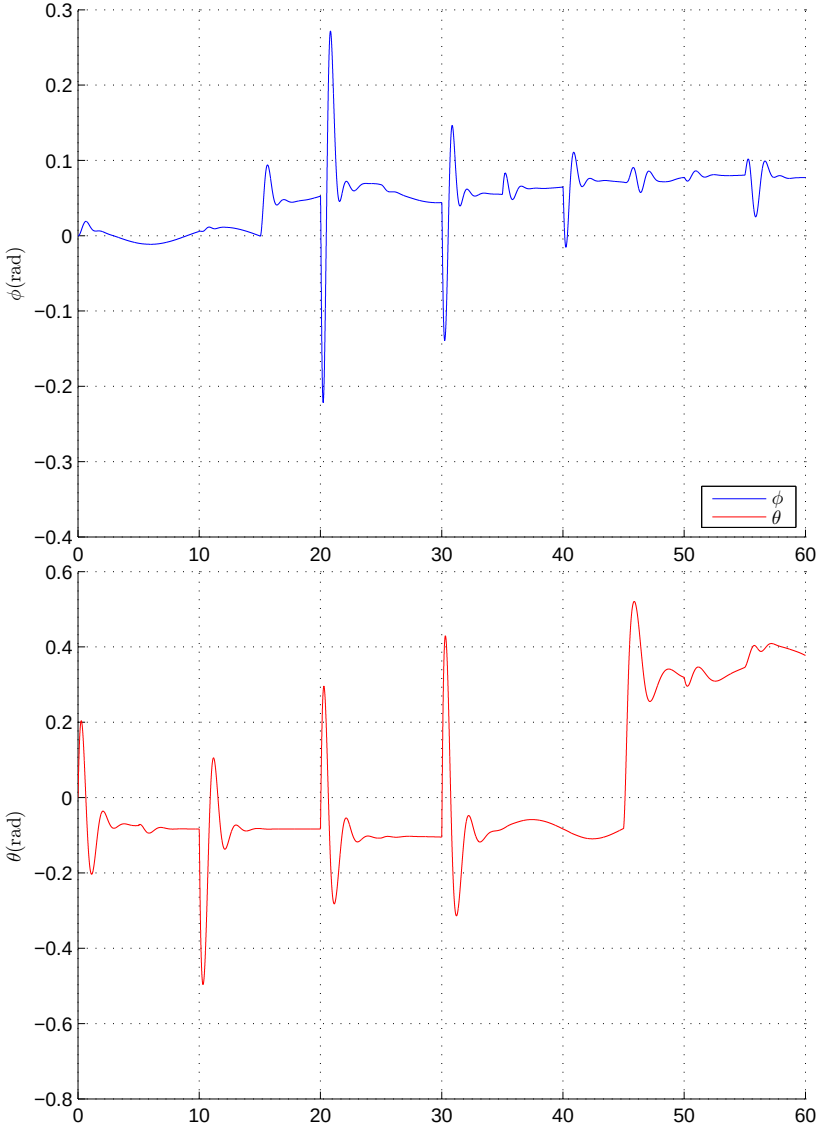


Figure 3.24 – Tiltrotor stabilized states for the backstepping controller.

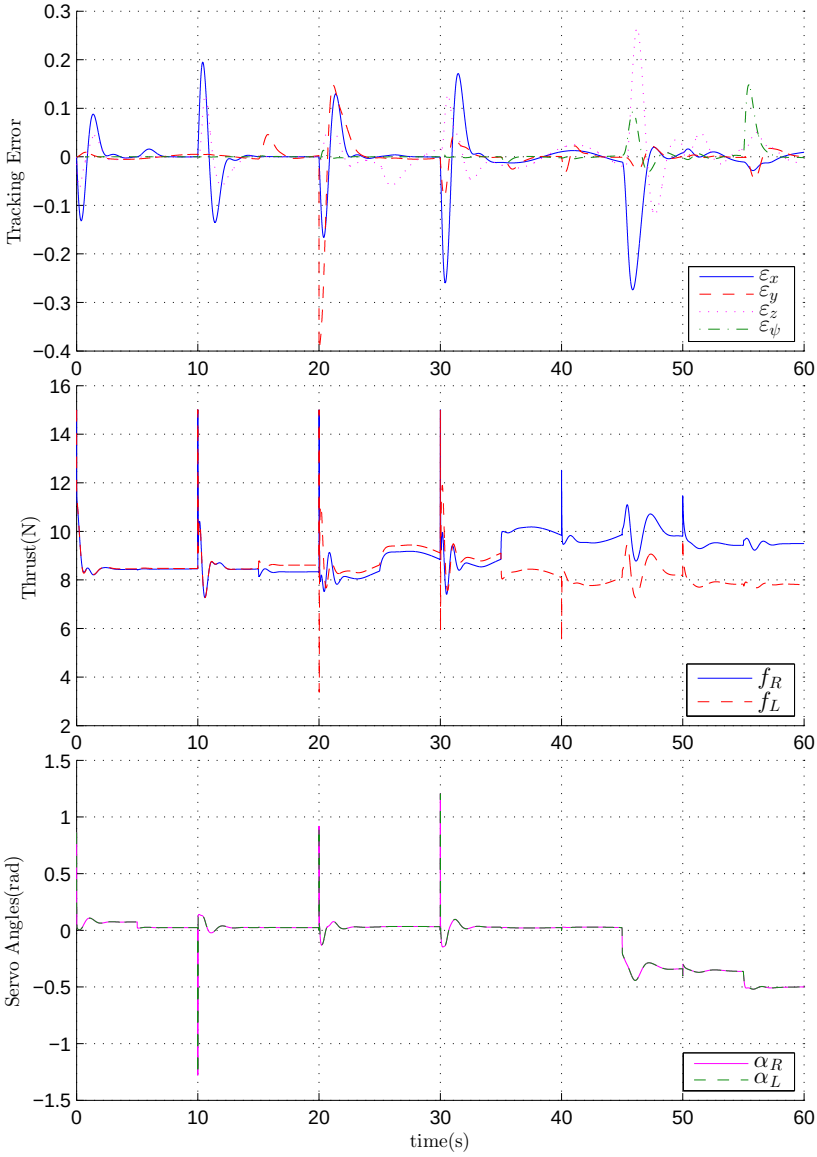


Figure 3.25 – Tiltrotor tracking error and control inputs for the backstepping controller.

	Backstepping
ISE_x	0.1838
ISE_y	0.0821
ISE_z	0.1056
ISE_ψ	0.0205

Table 3.11 – Integral Squared Error for the backstepping controller.

	Backstepping
$IAVU\ f_R$	542.626
$IAVU\ f_L$	506.507
$IAVU\ \alpha_R$	7.603
$IAVU\ \alpha_L$	7.688

Table 3.12 – Integrated Absolute Variation of the Control signal for the backstepping controller.

The backstepping has worse ISE indices than the previous controllers, with the exception of the $\mathcal{H}_\infty$ controller. Also, the other controllers had little to none oscillations, while the backstepping presents a relatively large oscillation when a disturbance in the $\theta(t)$ state occurs.

The $\theta(t)$ is indeed the most difficult state to control for the tiltrotor system, but it becomes more evident when the simplified model is used for the control design. This controller ignores the servomotor dynamics but cannot deny its effects in $\theta(t)$, causing it to oscillate. This effect will amplify for faster controllers.

As the servomotors dynamics are not included in this control, it assumes that the servomotor is fast enough to be neglected real life experiments. Considering the servomotor (inner loop) settling time should ideally be at least five times faster than the rotational system (outer loop), then the servos should settle in about 0.1 seconds. The servomotor used for Project ProVANT can control its speed until a maximum of 100 rpm, that is, 600 degrees per second. This means that in 0.1 seconds it can go through 60 degrees, which may not be enough depending on the trajectory to be followed and can have detrimental effects.

3.3 Comparative Simulation Results

This section aims to bring a side by side comparison for the controllers presented in sections 3.1 and 3.2. Whereas each controller section had simulation results for a abrupt and complex trajectory in order to test its stability and convergence, in this section a circular trajectory was chosen.

It is interesting to simulate the same controllers for a different trajectory, as one controller can be optimal to one trajectory and bad for others. This trajectory is also more adequate to visually compare the different results.

The simulations are executed for the system to track the trajectory $x_r = 1 - cos(\frac{\pi}{20}t)$, $y_r = sin(\frac{\pi}{20}t)$, $z_r = 1 - cos(\frac{\pi}{20}t)$ and $\psi = 0$. Several aspects are included in the simulation to test the controller robustness:

1. constant disturbances: $\delta_x = 0.5N$ at 10s; $\delta_y = 0.5N$ at 15s; $\delta_z = -1N$ at 20s; $\delta_\phi = 0.5N.m$ at 25s; $\delta_\theta = 0.5N.m$ at 30s; and $\delta_\psi = 0.5N.m$ at 35s. The disturbances are depicted in figure 3.26.
2. An uncertainty of 30% to the inertia moments was added.
3. Saturation on control signals were also considered, being the thrust forces constrained in the interval $[0, 15]N$ and the servo torques in $[-2, 2]N.m$, in accordance with the corresponding actuators specifications for Project ProVANT.

The initial condition $\boldsymbol{x}(0) = 0$ is used, as this condition can be easily achieved for real life experiments.

The simulations are once again made in Simulink 2013a, using the tiltrotor nonlinear model (3.1) with the parameters presented in Table 2.1. The same gains previously designed and calculated in each controller section are used.

Initially a comparison of the four linear time invariant controllers is carried out. Figures 3.27-3.32 show the system response. Next a comparison between the $\mathcal{H}_2/\mathcal{H}_\infty$ control, the LQR LTV explained in section 3.1.5 and the nonlinear backstepping controller in section 3.2 is made with the same simulation scenario. Figures 3.33-3.37 displays the results.

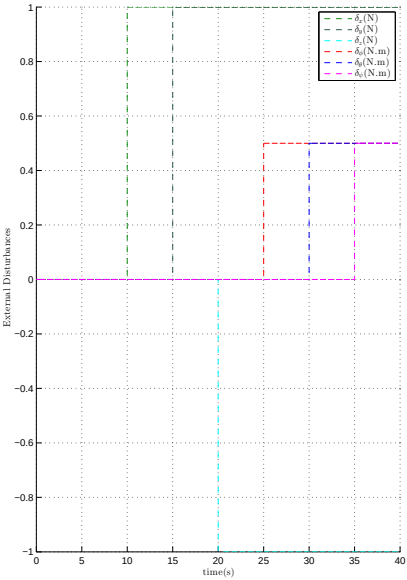


Figure 3.26 – Disturbances applied to the circular trajectory simulation.

In order to facilitate the controllers comparison, table 3.13 has the ISE for every controller and table 3.14 each controller's IAVU. Note however that the IAVU for the backstepping controller is given apart in table 3.15, as it uses different control variables.

	LQR	LQR LMI	$\mathcal{H}_\infty$	$\mathcal{H}_2/\mathcal{H}_\infty$	LQR LTV	BS
ISE_x	0.0023	0.0053	0.0412	0.0028	0.0023	0.0618
ISE_y	0.0021	0.0182	0.0662	0.0022	0.0020	0.0053
ISE_z	0.0001	0.0022	0.0085	0.0003	0.0001	0.0764
ISE_ψ	0.0121	0.0047	0.0058	0.0110	0.0121	0.0196

Table 3.13 – Integral Squared Error comparison for the circular trajectory.

All controllers were able to track the trajectory correctly and show similar response patterns as the simulations with the more abrupt trajectory. The controllers behaved well when taken in account all uncertainties, unmodeled dynamics and disturbances in the system. All three LQR controllers and the $\mathcal{H}_2/\mathcal{H}_\infty$ have a good response with low error and smooth control, which corroborates with the ISE and

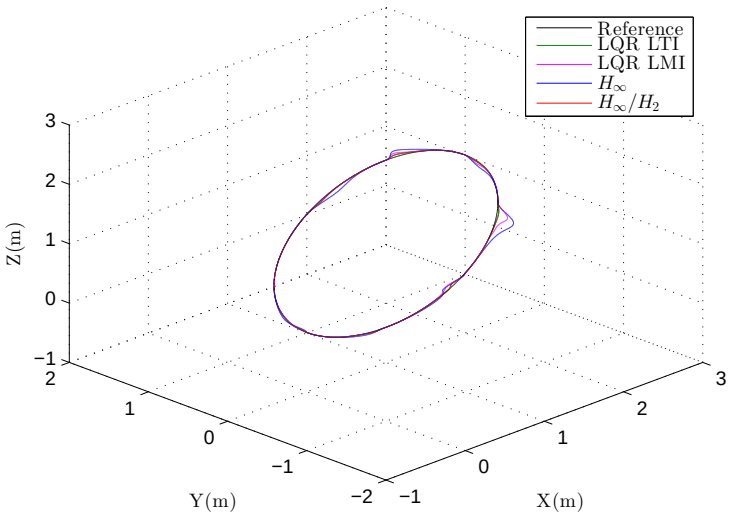


Figure 3.27 – Tiltrotor trajectory in space for the LTI controllers.

	LQR	LQR LMI	$\mathcal{H}_\infty$	$\mathcal{H}_2/\mathcal{H}_\infty$	LQR LTV
$IAVU\ f_R$	360.002	360.034	360.060	360.005	360.002
$IAVU\ f_L$	337.581	337.630	337.645	337.583	337.581
$IAVU\ \tau_{SR}$	0.0009	0.0012	0.0013	0.0009	0.0008
$IAVU\ \tau_{SL}$	0.0007	0.0014	0.0012	0.0007	0.0007

Table 3.14 – Integrated Absolute Variation of the Control signal comparison for the circular trajectory.

	Backstepping
$IAVU\ f_R$	360.0318
$IAVU\ f_L$	360.0318
$IAVU\ \alpha_R$	5.3114
$IAVU\ \alpha_L$	4.9598

Table 3.15 – Integrated Absolute Variation of the Control signal for the backstepping controller tracking the circular trajectory.

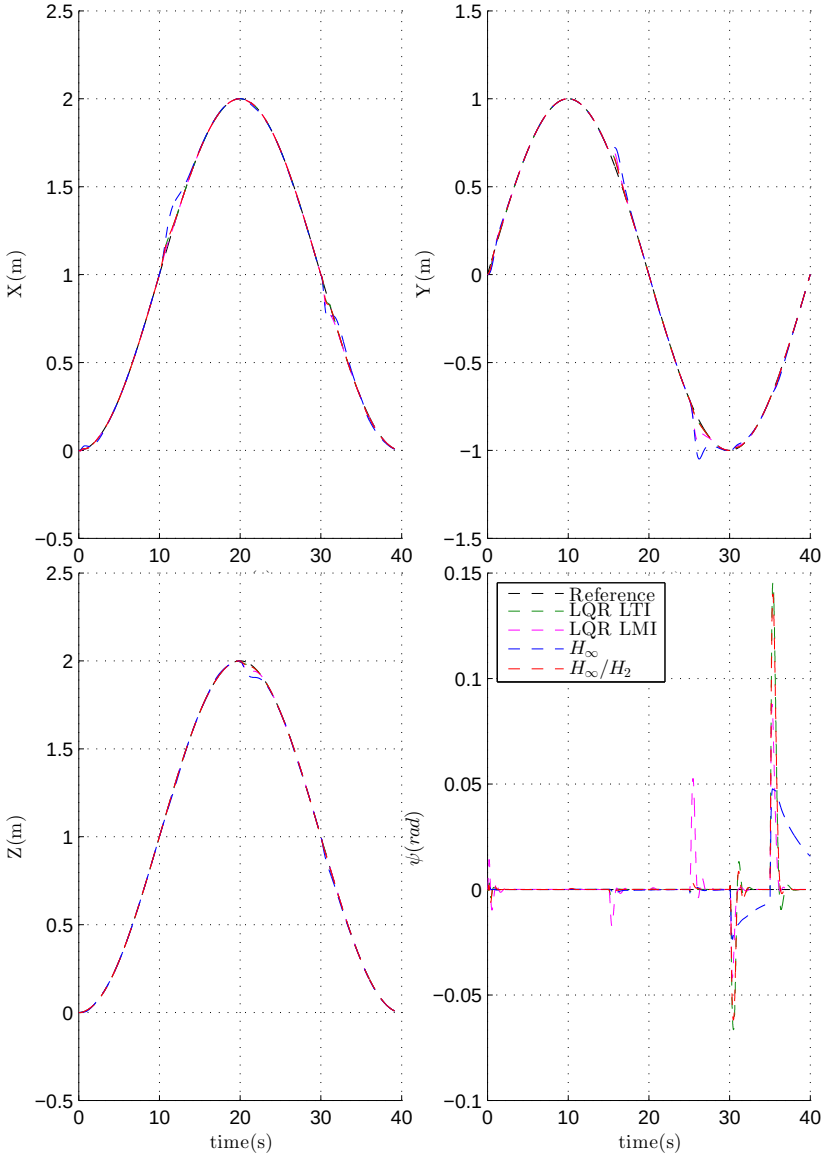


Figure 3.28 – Tiltrotor regulated degrees of freedom for the LTI controllers.

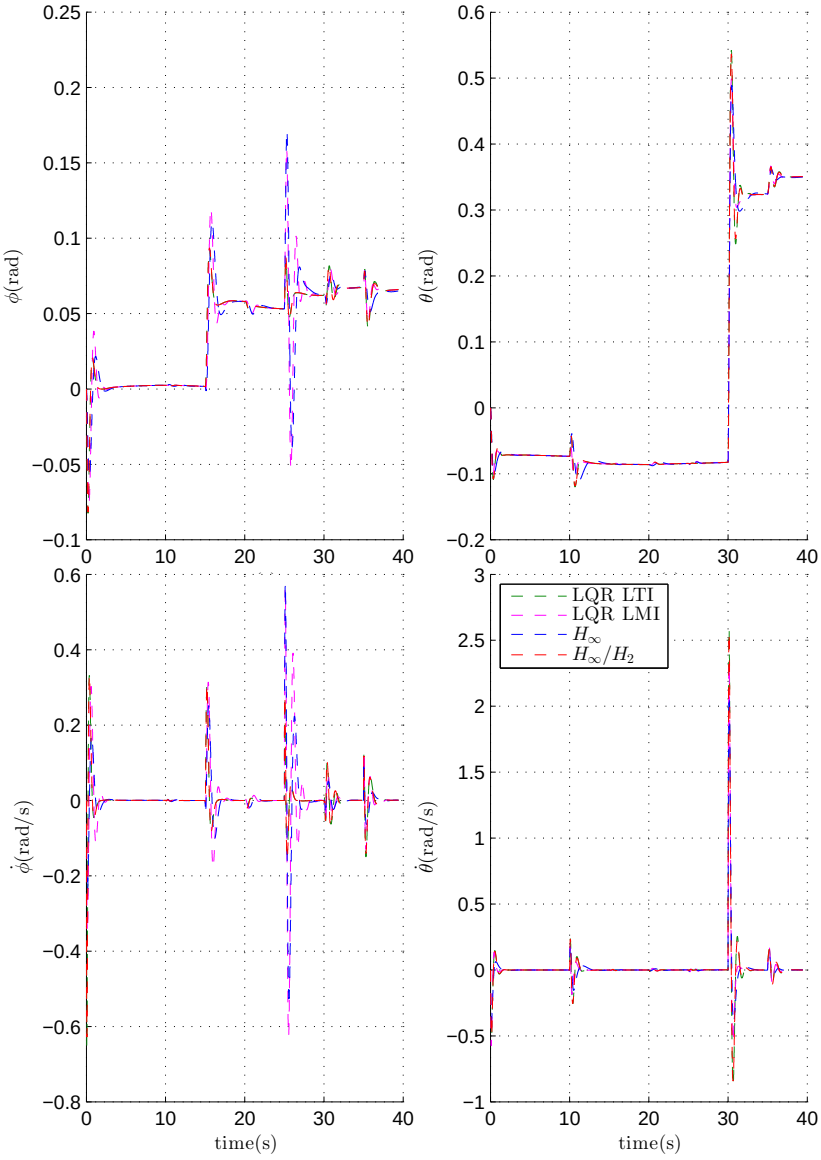


Figure 3.29 – Tiltrotor stabilized body angles and velocities for the LTI controllers.

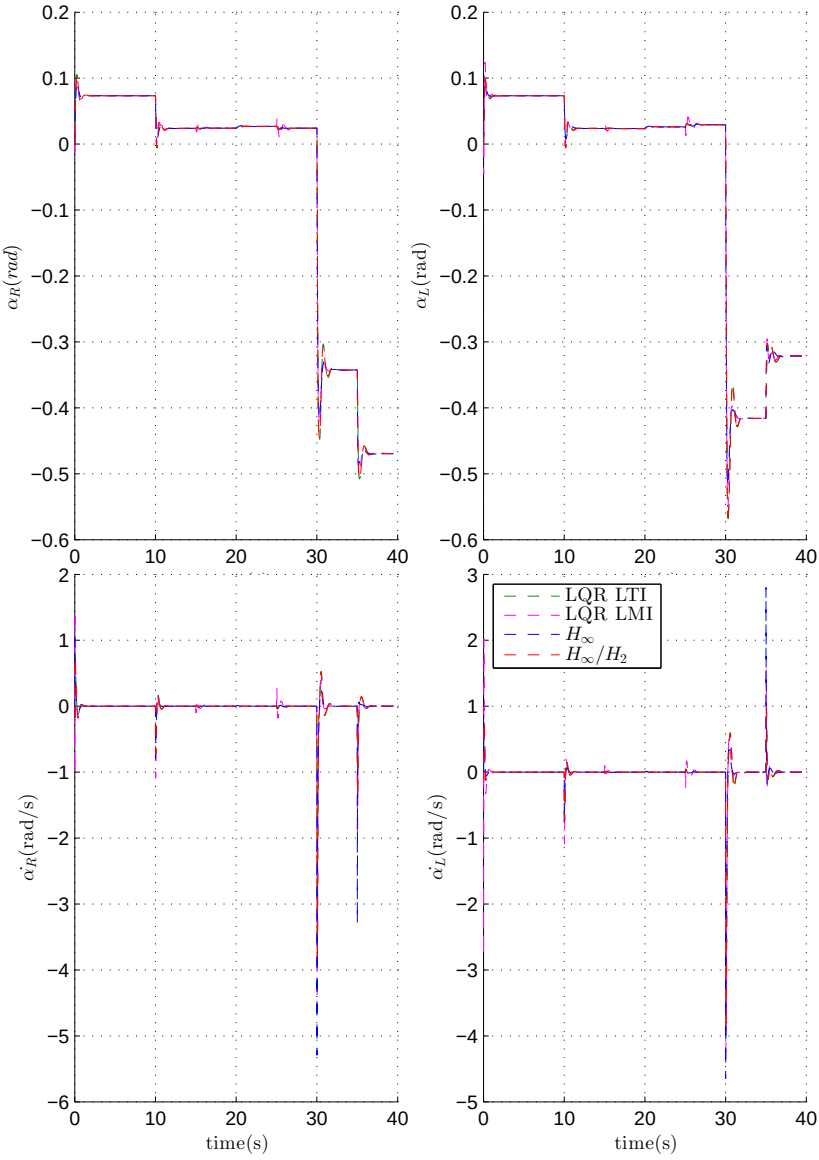


Figure 3.30 – Tiltrotor stabilized servos angles and velocities for the LTI controllers.

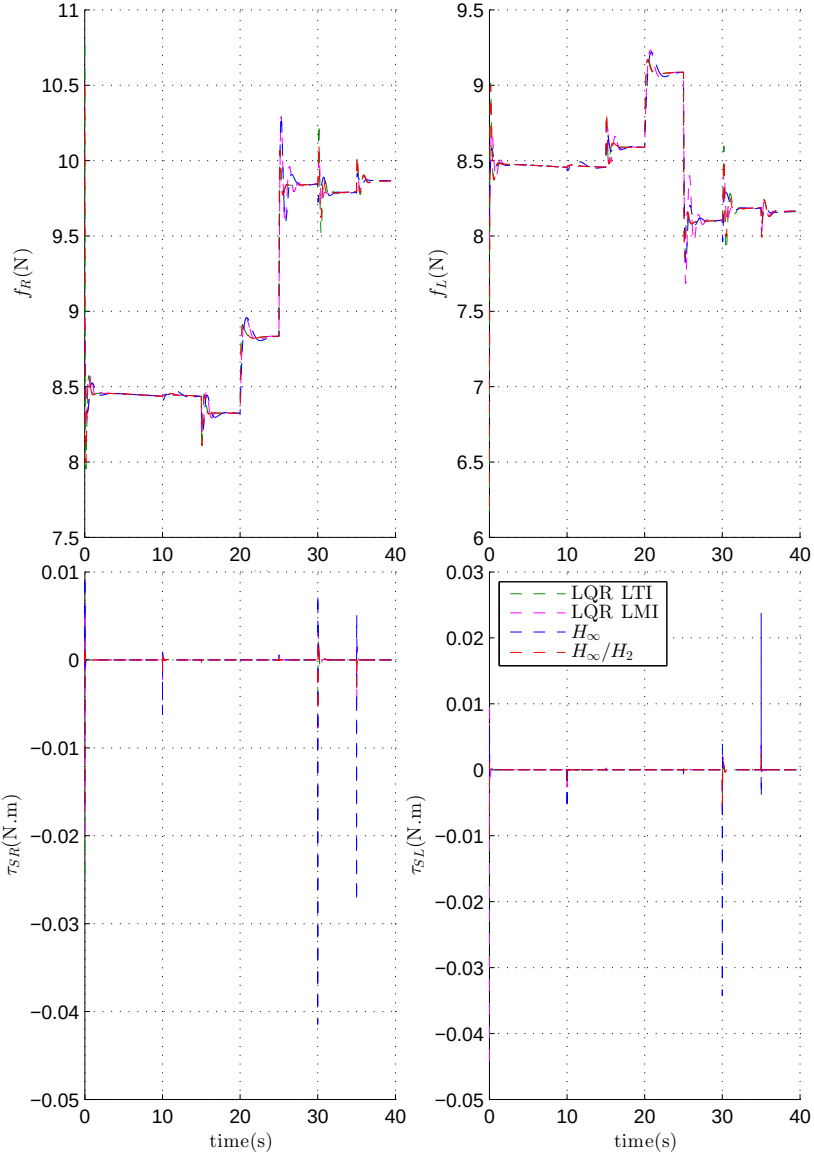


Figure 3.31 – Tiltrotor control inputs for the LTI controllers.

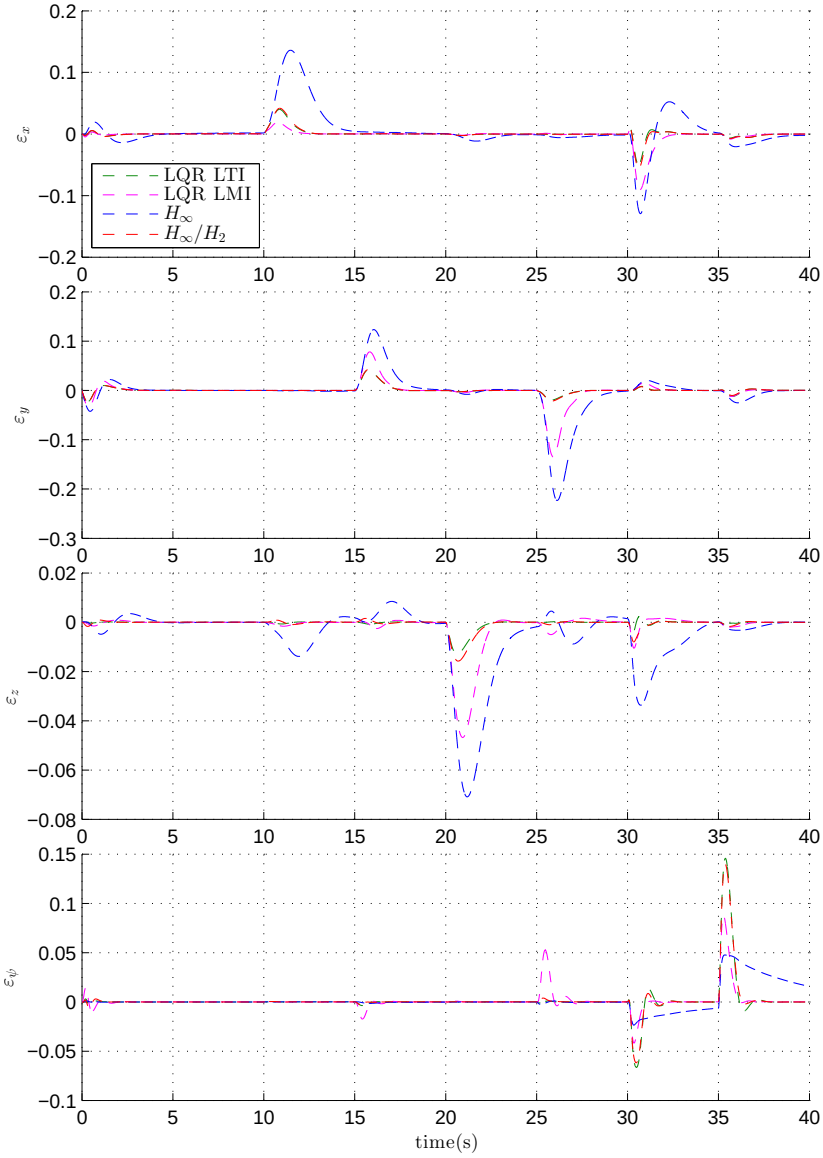


Figure 3.32 – Tracking error comparison for the LTI controllers.

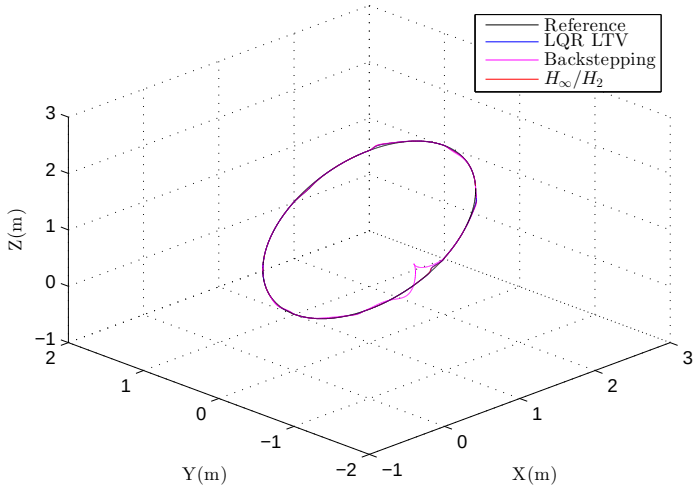


Figure 3.33 – Tiltrotor trajectory in space.

IAVU indices values. The $\mathcal{H}_\infty$ has a higher error, but is still has an overall good response.

The backstepping control presents oscillations in respect to disturbances in the $\theta(t)$ state, which happens at 30 seconds into the simulation. As discussed before, the cause for this are the simplifications assumed in the model used for control design.

The LQR LTV results are once again similar to the LQR LTI and $\mathcal{H}_\infty/\mathcal{H}_2$ controllers, meaning that the time varying parameters have only small effects to the system. The conclusion is that, unless the desired trajectory is very aggressive, the use of the LQR LTI and $\mathcal{H}_\infty/\mathcal{H}_2$ controllers for path tracking is justified, as the computational cost of time varying controllers is higher and more complex to implement.

It is also worth noting that the linear controllers have the limitation of always having to operate nearby the operating point, at risk of destabilizing the system in case of large deviations. However from figures 3.29 and 3.35 we can note that the angles deviate more than forty degrees from their equilibrium point, showing a large region

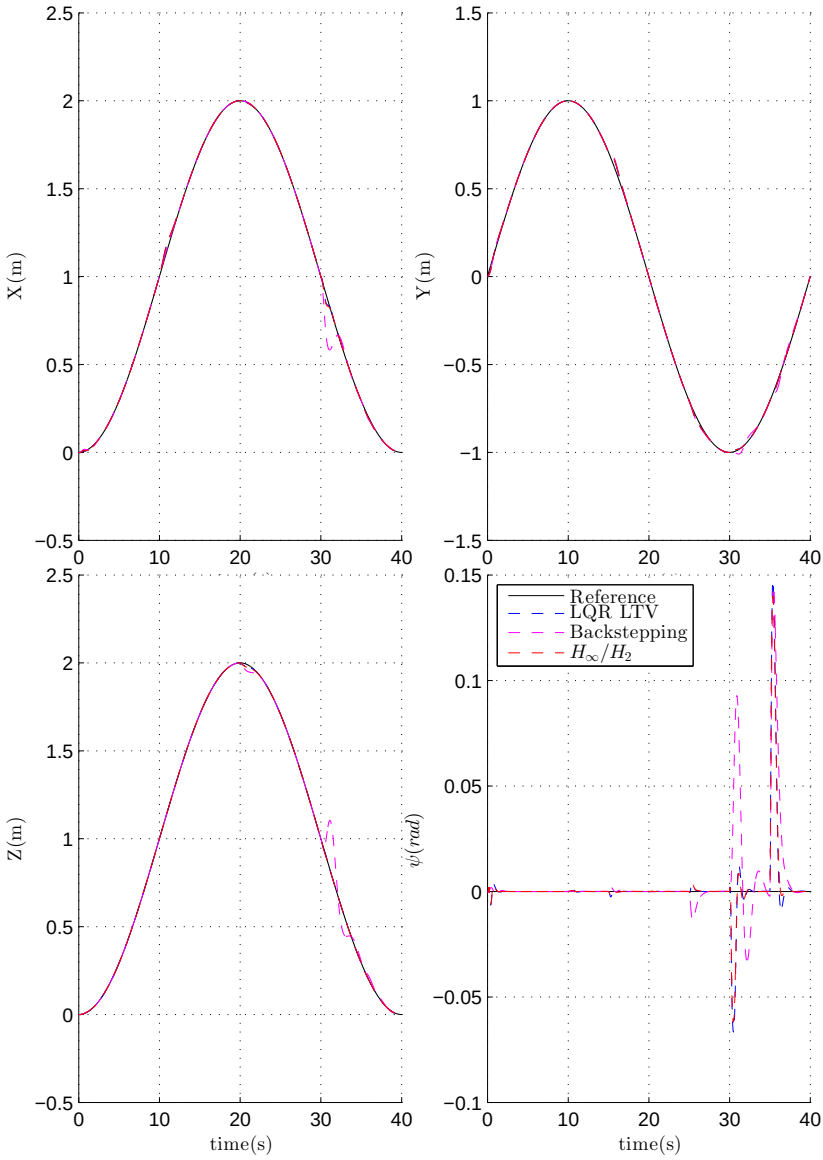


Figure 3.34 – Tiltrotor regulated degrees of freedom.

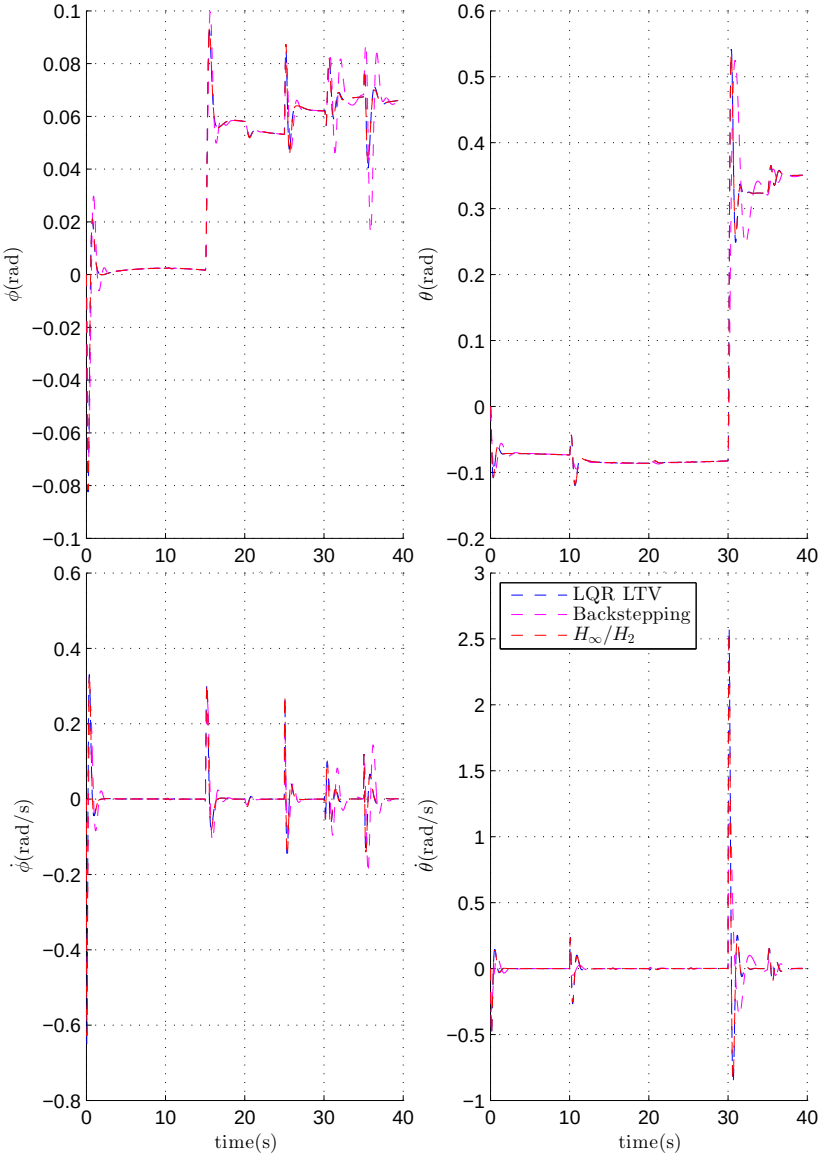


Figure 3.35 – Tiltrotor stabilized body angles and velocities.

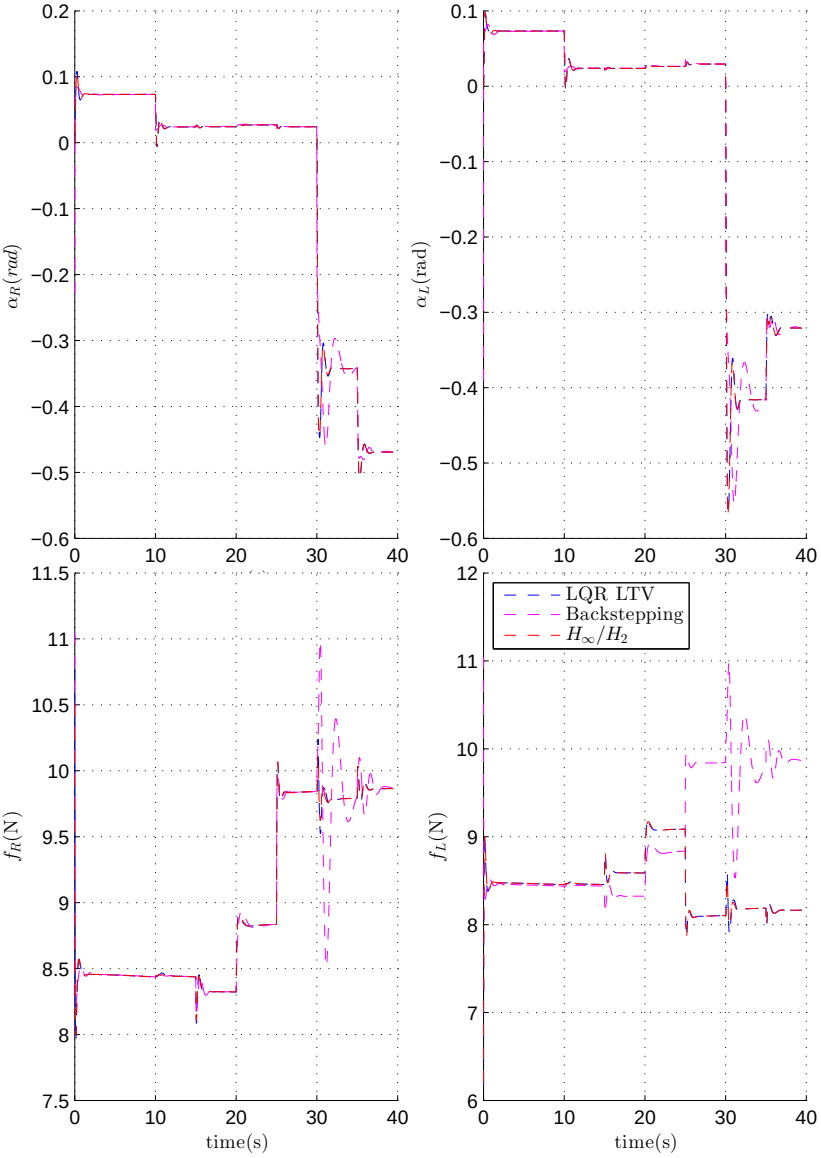


Figure 3.36 – Tiltrotor servos angles and thrust forces.

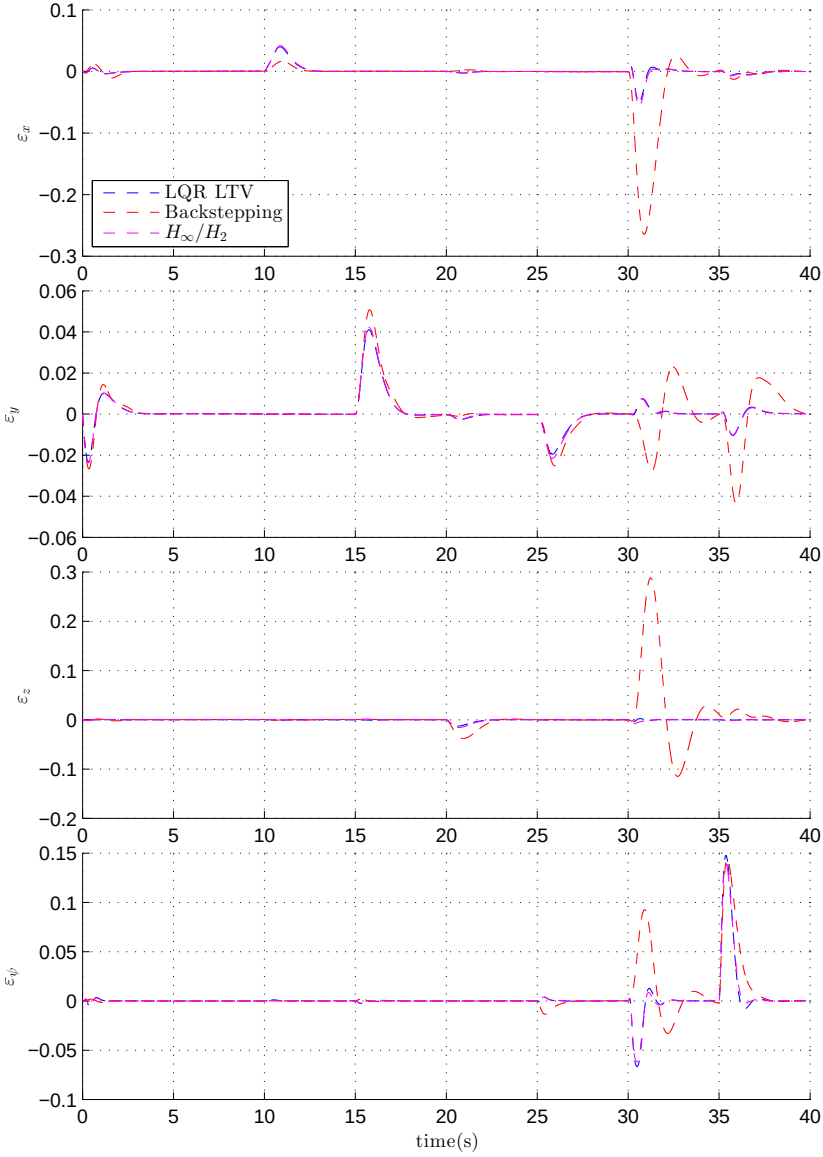


Figure 3.37 – Tracking error comparison.

of attraction.

The backstepping control presented the worse performance, but it is a nonlinear control and does not suffer the limitation of operating near the equilibrium points like the linear controllers and thus is still an interesting controller to test in experimental flights.

3.4 Hovering Control

Anticipating the difficulty of implementing the path tracking controllers in the real UAV, an intermediary control objective which solves the hovering problem is proposed. Hovering is the ability of the aircraft to maintain flight with an stable attitude and height. So, in other words, the hovering control aims stabilize rotational system and the height of the aircraft.

Therefore, the LQR LTI controller developed in section 3.1 and the backstepping controller from section 3.2 are redesigned for a simplified version of their respective models, with the objective of performing hovering, rather than following a trajectory.

Both controllers are aimed to be the first to be implemented and tested in ProVANT's prototype, so a slightly slower dynamic was designed. Also, as the first prototype does not possess a servomotor that is controlled by torque, the servo angle state is dropped and used as a control input. Therefore the simplified generalized coordinates are:

$$\boldsymbol{q} = \begin{bmatrix} z(t) & \phi(t) & \theta(t) & \psi(t) \end{bmatrix}' \quad .$$

Like so, the nonlinear model considered for the hovering control design is a simplification of model (2.62), defined as

$$\boldsymbol{M}_h(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{C}_h(\boldsymbol{q}, \dot{\boldsymbol{q}})\dot{\boldsymbol{q}} + \boldsymbol{G}_h(\boldsymbol{q}) = \boldsymbol{F}_h(\boldsymbol{q}) \quad . \tag{3.94}$$

First the inertia matrix depicted in equation (2.46) is simplified. It is done by eliminating the rows and columns relative to $x(t)$, $y(t)$, $\alpha_R(t)$ and $\alpha_L(t)$, leaving only the elements related to the states to be controlled. It becomes:

$$\boldsymbol{M}_h(\boldsymbol{q}) = \begin{bmatrix} m & \boldsymbol{L} \\ \boldsymbol{L}' & \boldsymbol{W}_{\boldsymbol{\eta}}'\boldsymbol{J}\boldsymbol{W}_{\boldsymbol{\eta}} \end{bmatrix} \quad , \tag{3.95}$$

where $\boldsymbol{L}$ is a 1×3 matrix correspondent to the third row of $-\boldsymbol{R}_{\mathfrak{B}}^{\mathfrak{I}}\boldsymbol{H}\boldsymbol{W}_{\boldsymbol{\eta}}$.

The Coriolis and centrifugal force matrix $\boldsymbol{C}_h(\boldsymbol{q},\dot{\boldsymbol{q}})$ is again calculated with Christoffel symbols of the first kind in equation (2.63). The gravitational force vector comes from the simplification of (2.65):

$$\boldsymbol{G}_h(\boldsymbol{q}) = \begin{bmatrix} gm \\ g(m_yC\theta C\phi - m_zC\theta S\phi) \\ g(m_xC\theta - m_yS\theta S\phi - m_zS\theta C\phi) \\ 0 \end{bmatrix}, \tag{3.96}$$

where $m_x = \sum_{i=1}^{3} m_i d_{ix}$, $m_y = \sum_{i=1}^{3} m_i d_{iy}$ and $m_z = \sum_{i=1}^{3} m_i d_{iz}$.

Finally the external force vector comes from expanding equation (2.59) and selecting only the rows related to states to be controlled:

$$\boldsymbol{F}_h(\boldsymbol{q}) = \begin{bmatrix} F_z \\ \boldsymbol{F}_{\boldsymbol{\eta}} \end{bmatrix} \tag{3.97}$$

with

$$F_z = (C\bar{\psi}S\bar{\theta}C\bar{\phi} + S\bar{\psi}S\bar{\phi})(S(\bar{\alpha}_R)\bar{f}_R + S(\bar{\alpha}_L)\bar{f}_L) + (S\bar{\psi}S\bar{\theta}C\bar{\phi} - C\bar{\psi}S\bar{\phi})(C(\bar{\alpha}_R) S(\beta)\bar{f}_R - C(\bar{\alpha}_L)S(\beta)\bar{f}_L) + C\bar{\theta}C\bar{\phi}(C(\bar{\alpha}_R)C(\beta)\bar{f}_R + C(\bar{\alpha}_L)C(\beta)\bar{f}_L) = 0 \tag{3.98}$$

and

$$\boldsymbol{F}_{\boldsymbol{\eta}} = \boldsymbol{W}_{\boldsymbol{\eta}}^{-1} \begin{bmatrix} (-C(\alpha_R)C(\beta)S(\lambda)l' - \frac{k_\tau}{b}S(\alpha_R))f_R + (C(\alpha_L)C(\beta)S(\lambda)l' + \frac{k_\tau}{b}S(\alpha_L)f_L \\ (S(\alpha_R)d_z + \frac{k_\tau}{b}S(\beta)C(\alpha_R))f_R + (S(\alpha_L)d_z - \frac{k_\tau}{b}S(\beta)C(\alpha_L))f_L \\ (S(\alpha_R)l + \frac{k_\tau}{b}C(\beta)C(\alpha_R))f_R - (S(\alpha_L)l + \frac{k_\tau}{b}C(\beta)C(\alpha_L))f_L \end{bmatrix} \tag{3.99}$$

Note that the simplified model in equation (3.94) is used only for the control design, the simulations still use the model depicted by equation (3.1).

3.4.1 LQR State feedback Control

In this section a LQR, as presented in section 3.1.1, is developed for the simplified height and rotational system. The new system

states are:

$$\boldsymbol{x}(t) = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \\ x_7 \\ x_8 \end{bmatrix} = \begin{bmatrix} z \\ \phi \\ \theta \\ \psi \\ \dot{z} \\ \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} \quad , \tag{3.100}$$

with the control inputs

$$\Gamma = \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{bmatrix} = \begin{bmatrix} f_R \\ f_L \\ \alpha_R \\ \alpha_L \end{bmatrix} \quad . \tag{3.101}$$

Since x and y are not controlled, roll and pitch angles have multiple equilibrium points. However, as we want to minimize the aircraft's displacement in the XY plane, the equilibrium point for the orientation is the same as in (3.6). Once again using equation (3.4) and the parameters in table 2.1, with $\beta = 0°$, gives:

$$\begin{array}{cc} \bar{f}_R = 8.31302\ N\ , & \bar{f}_L = 8.57765\ N \\ \bar{\alpha}_R = 0°\ , & \bar{\alpha}_L = 0° \end{array} \quad . \tag{3.102}$$

The linearized system results in the LTI system:

$$\Delta\dot{\boldsymbol{x}} = \boldsymbol{A}\Delta\boldsymbol{x}(t) + \boldsymbol{B}\Delta\boldsymbol{u} \quad , \tag{3.103}$$

with matrices A and B depicted in the appendix section B.2.

The controller does not follow a trajectory and is stabilized around the equilibrium point, so $\Delta\boldsymbol{x} = \boldsymbol{x} - \bar{\boldsymbol{x}}$. Finally matrices Q and R are designed as:

$$\boldsymbol{R} = \begin{bmatrix} \frac{1}{(15-\bar{f}_R)^2} & 0 & 0 & 0 \\ 0 & \frac{1}{(15-\bar{f}_L)^2} & 0 & 0 \\ 0 & 0 & \frac{1}{(\frac{\pi}{2})^2} & 0 \\ 0 & 0 & 0 & \frac{1}{(\frac{\pi}{2})^2} \end{bmatrix} \quad , \tag{3.104}$$

and

$$\boldsymbol{Q} = diag(\begin{bmatrix} 5 & \frac{5}{(\frac{\pi}{3})^2} & \frac{15}{(\frac{\pi}{3})^2} & \frac{5}{\pi^3} & \frac{1}{2^2} & \frac{1}{(2*\pi)^2} & \frac{1}{(2*\pi)^2} & \frac{1}{(\pi)^2} \end{bmatrix}) \quad . \tag{3.105}$$

The control gain matrix is found by solving the algebraic Riccatti equation (3.20) and the result is presented in appendix B, matrix (B.13).

3.4.2 Backstepping Nonlinear Control

The backstepping control became simpler now, since the roll and pitch angles have their own direct references. The rest of the control steps remain the same, with equations (3.73) to (3.76) being used to design the system dynamics.

Using the same steps as in section 3.2 gives the gains:

$$\begin{array}{rcl} K_{vz} = 2.4 & , & K_{pz} = 3 \\ K_{v\phi} = 19.2 & , & K_{p\phi} = 92.16 \\ K_{v\theta} = 19.2 & , & K_{p\theta} = 92.16 \\ K_{v\psi} = 19.2 & , & K_{p\psi} = 92.16 \end{array} \quad . \tag{3.106}$$

3.4.3 Comparative Simulation Results

The simulation of both controllers are done with the same full model (3.1) used for the simulations in section 3.3. It also uses the same scenario with perturbations and saturation, the difference being that these are stabilization controllers and no path tracking is required from them. Its references now are $\{z_r, \phi_r, \theta_r, \psi_r\} = \{1, \bar{\phi}, \bar{\theta}, 0\}$. The new initial condition is $\boldsymbol{x}(0) = \{0, 0.1, 0.1, 0.1, 0, 0, 0, 0\}$.

The simulation results are depicted in figure 3.38 and figure 3.39. Table 3.16 shows the controllers ISE index and table 3.17 the IAVU index.

Both controllers were able to solve the hovering problem with good results. The LQR controller behaved strictly better than the backstepping as it shows less overshoots, less variation amidst disturbances and a smoother control signal. Note that, for not employing integral ac-

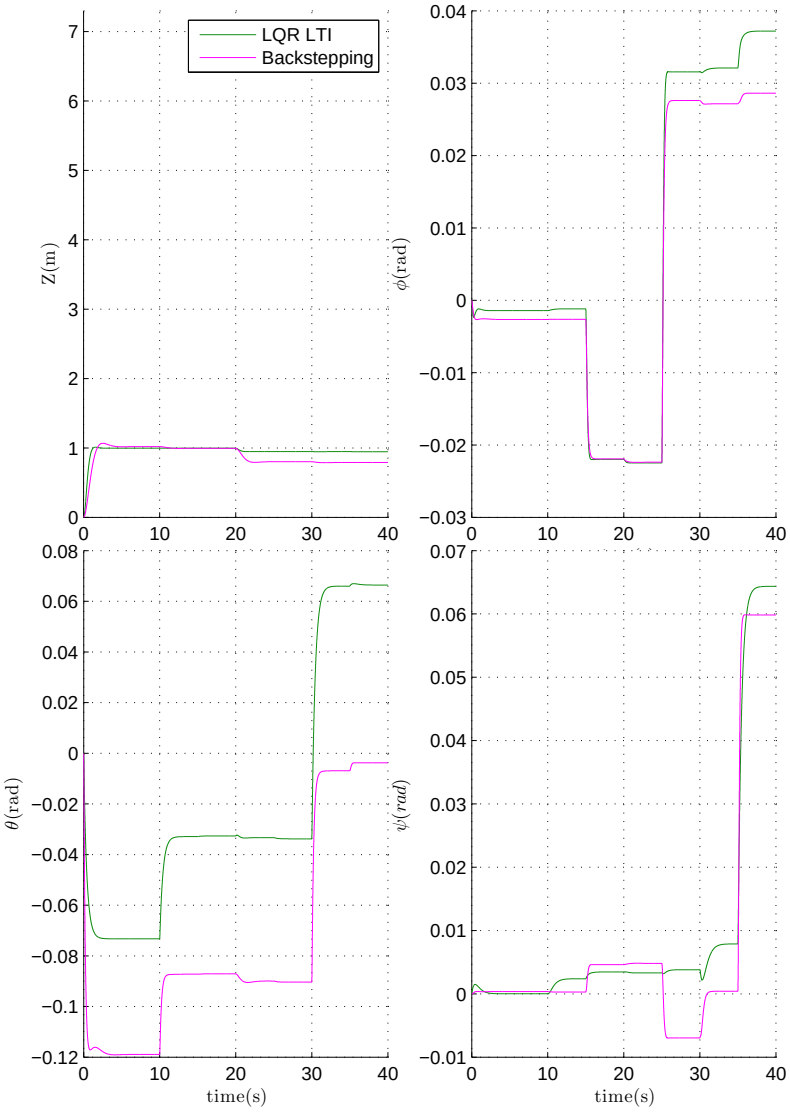


Figure 3.38 – Tiltrotor height and rotational system regulated states stabilization.

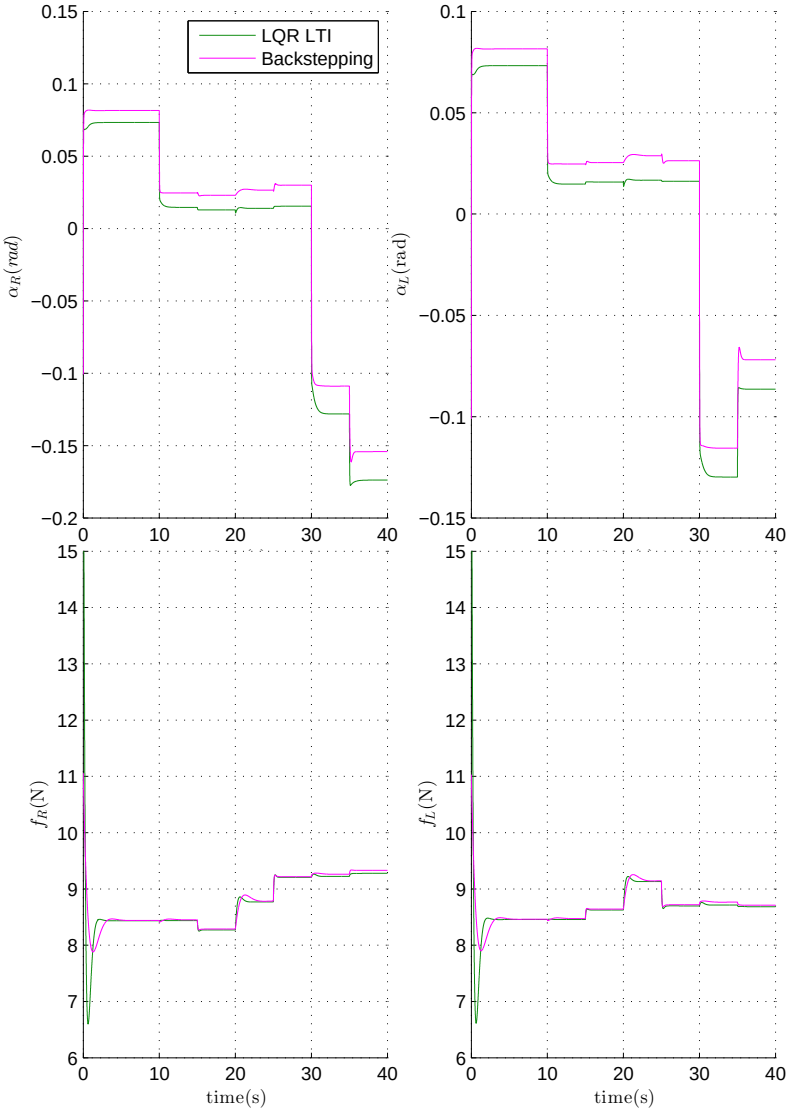


Figure 3.39 – Tiltrotor height and rotational system control inputs.

	LQR LTI	Backstepping
ISE_z	0.218	1.403
ISE_ϕ	0.0185	0.0166
ISE_θ	0.645	0.0693
ISE_ψ	0.0086	0.0174

Table 3.16 – Integral Squared Error comparison for the hovering controllers.

	LQR LTI	Backstepping
$IAVU\ f_R$	350.2976	351.0797
$IAVU\ f_L$	346.1507	346.9324
$IAVU\ \alpha_R$	2.5145	2.6436
$IAVU\ \alpha_L$	2.1221	2.2730

Table 3.17 – Integrated Absolute Variation of the Control signal comparison for the hovering controllers.

tion, the variables stabilize at different values depending on the applied disturbance.

The backstepping controller, even if worse than the LQR controller, performed better than its path tracking counterpart, presenting no oscillations. This is due to the system being simpler, with far less coupling between states and the system is no longer underactuated.

3.5 Summary

This chapter presented the control design to solve the problems of hovering and path tracking for a tiltrotor UAV. Five linear optimal controllers were developed for the path tracking problem, being the robustness and performance the determining factors behind most decisions. A recent work found in literature in (CHOWDHURY et al., 2012) was also implemented for comparison with the developed controllers, with changes in its structure and design being made for the sake of a better response.

Simulations were made with the nonlinear system without any simplifications and in addition to external disturbances and para-

metric uncertainties in order to test the robustness of the controllers.

All controllers managed to follow the predefined trajectory proposed, being the best performances achieved with the LQR LTI, $\mathcal{H}_2/\mathcal{H}_\infty$ and LQR LTV controllers. Taking into account the enhanced robustness of the $\mathcal{H}_2/\mathcal{H}_\infty$ in comparison to the LQR LTI, and that it is a controller which consumes much less computational resources than the LQR LTV, then the conclusion is that the LQR LTI and $\mathcal{H}_2/\mathcal{H}_\infty$ controllers produced the best results.

Two additional controllers were developed for controlling stability of the tiltrotor. The idea was to design simpler controllers as they will be the first to be implemented and tested with the tiltrotor assembled by project ProVANT. It is important to make a smaller step because in the first experiments not only the controllers but also the system as a whole (electronics, software, etc.) needs to be validated. As a result a LQR LTI and a nonlinear backstepping controller were designed, the former being an adaptation from (CHOWDHURY et al., 2012). Both controllers behaved well and were implemented and tested in the UAV, as presented in the following chapter.

Chapter 4

Experimental Results

This chapter starts with a brief description of the tiltrotor UAV assembled within project ProVANT, including the navigation system used. Next it presents and discusses the results obtained by means of the experimental flights performed with the hovering controllers presented in section 3.4.

4.1 Tiltrotor UAV assembly

The birotor tiltrotor UAV designed within ProVANT is assembled with the following electronic components:

- 2x brushless motors AXI 2814/20 GOLD LINE
- 2x servomotors Dynamixel RX-24F
- 1x development board STM32F4DISCOVERY ARM Cortex-M4 32-bit MCU with FPU
- 1x 9-DOF inertial measurement system (IMU) GY-85
- 2x electronic speed controller (ESC) Mikrokopter BL-Ctrl 2.0
- 1x Ultrasonic ranging module HC-SR04
- 1x 2.4 GHz 6 channels radio receiver hobbyking HK-TR6A

- 1x 4-cell LIPO battery (16.8V) Turnigy nanotech 3000mah
- 1x voltage regulator 16.8V/5V
- 1x voltage regulator 16.8V/12V

For the mechanical assembly, carbon fiber is used in the arm that connects the rotors, and glass fiber is used in the body structure that supports the embedded system and the electronic devices. It also contains a protection structure made with glass fiber rods. Smaller parts designed in the software *SolidWorks*® and printed using 3D printers available at the Automation and Systems Department (DAS) at UFSC and at the Electronic Engineering Department (DELT) at UFMG. Figure 4.1 depicts the UAV assembled.

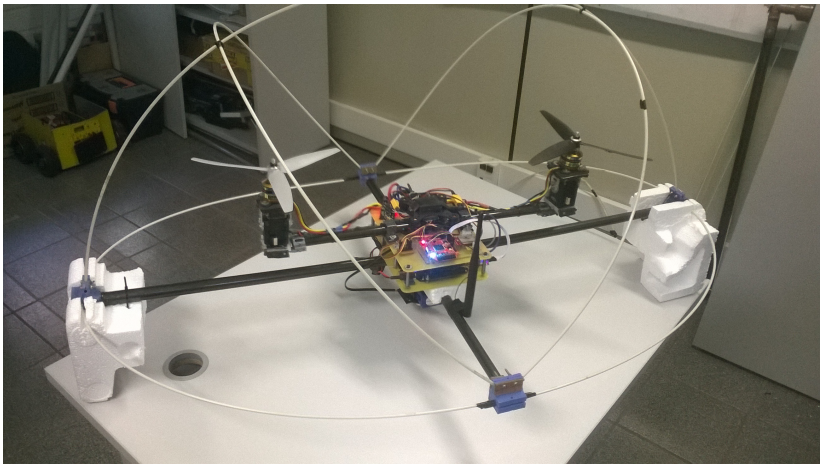


Figure 4.1 – Project ProVANT's first UAV prototype assembly.

4.2 Tiltrotor Navigation System

The role of the navigation system is to estimate the tiltrotor states needed to implement the controllers presented in Chapter 3. Such states are the angular position and velocity ϕ, θ, ψ, $\dot{\phi}$, $\dot{\theta}$, $\dot{\psi}$; the linear position and velocity x, y, z, $\dot{x}$, $\dot{y}$, $\dot{z}$; and the rotor angles characterized by the servomotor angles α_R and α_L, and its velocities $\dot{\alpha}_R$ and $\dot{\alpha}_L$.

ProVANT's tiltrotor prototype has sensors that provide measurements of the linear acceleration, angular velocity, orientation of the magnetic field, and the relative altitude, allowing the estimation of z, $\dot{z}$, ϕ, $\dot{\phi}$, θ, $\dot{\theta}$, ψ, and $\dot{\psi}$. However, the estimation of x and y using only the linear acceleration measurement is not satisfactory, requiring additional sensing to produce a reliable estimation like, for example, the use of a global positioning system (GPS).

4.2.1 Angular position and velocity

To determine the UAV's attitude and angular velocity the following sensors are available in the prototype:

- 3 gyroscopes, which measure the angular velocity for each axis of the body frame;
- 3 accelerometers, that measure the total instantaneous force in each of the three axis of the body frame;
- and 3 magnetometers that measure the local magnetic field components for each axis of the body frame.

All sensors are encapsulated within the 9-DOF IMU and it is assumed that there is no alignment error in the mounting of the sensors. As they are all calibrated, the only errors considered in this analysis are the noises and the gyroscopes bias, which drifts due to the low cost IMU. This sensors typically have a slow drifting bias, which can be significant.

Note that none of these sensors directly measures the system's attitude, however it can be estimated. Two different estimations of the attitude were developed along this work: one makes use of the gyroscopes data and the other of the accelerometers and magnetometers data. Next, a fusion filter that combines both estimations is detailed. Finally, the quaternion equivalent of the filter, which is the one currently implemented in the UAV, is briefly explained.

Firstly, let us define the gyroscopes measurements as:

$$\boldsymbol{\omega}_m^{\mathfrak{B}} = \boldsymbol{w}_{\mathfrak{B}\mathfrak{I}}^{\mathfrak{B}} + \boldsymbol{b}_\omega + \boldsymbol{\mu}_\omega \quad , \tag{4.1}$$

where $\boldsymbol{w}^{\mathfrak{B}}_{\mathfrak{BJ}}$ is the true value of the body frame angular velocity in relation to the inertial frame expressed in the body frame, $\boldsymbol{b}_\omega$ is the gyro bias and $\boldsymbol{\mu}_\omega$ is the measurement noise.

To perform the attitude estimation with the angular velocities, it is possible to use the rotation kinematics presented in equation (2.12). Using $\boldsymbol{\omega}^{\mathfrak{B}}_{\boldsymbol{m}}$ and defining $\hat{\boldsymbol{R}}$ as the estimation of the rotation matrix:

$$\dot{\hat{\boldsymbol{R}}}^{\mathfrak{I}}_{\mathfrak{B}} = \hat{\boldsymbol{R}}^{\mathfrak{I}}_{\mathfrak{B}} \boldsymbol{S}(\boldsymbol{\omega}^{\mathfrak{B}}_{m} - \boldsymbol{b}_\omega) \quad . \tag{4.2}$$

From the estimation of the rotation matrix it is trivial to extract the roll, pitch and yaw angles. Defining $\hat{\boldsymbol{R}}^{\mathfrak{I}}_{\mathfrak{B}(\mathrm{i,j})}$ the element from line i, column j of matrix $\hat{\boldsymbol{R}}^{\mathfrak{I}}_{\mathfrak{B}}$ and analyzing equation (2.6) yields to:

$$\begin{aligned} \phi &= arctan(\hat{R}^{\mathfrak{I}}_{\mathfrak{B}(2,3)} / \hat{R}^{\mathfrak{I}}_{\mathfrak{B}(3,3)}) \\ \theta &= arcsin(-\hat{R}^{\mathfrak{I}}_{\mathfrak{B}(1,3)}) \\ \psi &= arctan(\hat{R}^{\mathfrak{I}}_{\mathfrak{B}(1,2)} / \hat{R}^{\mathfrak{I}}_{\mathfrak{B}(1,1)}) \end{aligned} \quad . \tag{4.3}$$

Accelerometers measure the instantaneous acceleration, i.e., the difference between the acceleration of the vehicle and the gravitational acceleration. Considering a measurement noise μ_a, the accelerometers measures are:

$$\boldsymbol{a}^{\mathfrak{B}}_{\boldsymbol{m}} = \boldsymbol{R}^{\mathfrak{B}}_{\mathfrak{I}}(\dot{\boldsymbol{v}} - \boldsymbol{g}) + \boldsymbol{\mu_a} \quad . \tag{4.4}$$

Usually the gravitational acceleration dominates the measurement, particularly in our case in which the trajectories accelerations are bounded to $\|0.5\| m/s^2$ as stated in chapter 3. Therefore, a common practice is to disregard $\dot{v}$ in equation (4.4) (LEISHMAN et al., 2014), which in practice means we are determining the direction of the gravity vector. This results in:

$$\begin{bmatrix} a_x \\ a_y \\ a_z \end{bmatrix} = \boldsymbol{R}^{\mathfrak{B}}_{\mathfrak{I}} \begin{bmatrix} 0 \\ 0 \\ -g \end{bmatrix} = \begin{bmatrix} S(\theta)g \\ -S(\phi)C(\theta)g \\ -C(\phi)C(\theta)g \end{bmatrix} \quad . \tag{4.5}$$

Then, the roll and pitch angles can be estimated using the linear acceleration measurements as follows:

$$\hat{\phi}_a = arctan(\frac{a_y}{a_z}) \tag{4.6}$$

$$\hat{\theta}_a = arcsin(\frac{a_x}{g}) \quad . \tag{4.7}$$

The drawback of this estimation lies in its reliability, being only appropriate for slow dynamics. Note that the yaw angle ψ does not appear in equation (4.5). To estimate yaw the magnetometers are used.

Defining the magnetometer measurement as

$$\boldsymbol{m}_{\boldsymbol{m}}^{\mathfrak{B}} = \boldsymbol{R}_{\mathfrak{I}}^{\mathfrak{B}} \boldsymbol{m}_{\boldsymbol{e}}^{\mathfrak{I}} + \boldsymbol{\mu}_{\boldsymbol{m}} \quad , \tag{4.8}$$

where $\boldsymbol{m}_{\boldsymbol{e}}^{\mathfrak{I}}$ is Earth's magnetic field and $\boldsymbol{\mu}_{\boldsymbol{m}}$ the measurement noise.

The magnetometers measurements refer to the body frame $\mathfrak{B}$. These measurements can be rotated backwards with the estimated roll and pitch angles obtained from accelerometer measures, leaving the measurements with respect to a frame that is parallel to the inertial frame except for the yaw angle, as frame $\mathfrak{T}_1$ in figure 2.2:

$$\begin{aligned} \boldsymbol{m}_{\boldsymbol{e}}^{\mathfrak{T}_1} = \begin{bmatrix} m_x \\ m_y \\ m_z \end{bmatrix} &= \boldsymbol{R}(\vec{Y}_{\mathfrak{T}_2}, -\theta)\boldsymbol{R}(\vec{X}_{\mathfrak{B}}, -\phi) \begin{bmatrix} m_{mx} \\ m_{my} \\ m_{mz} \end{bmatrix} \\ &= \begin{bmatrix} m_{mx}C\theta + m_{my}S\phi S\theta - m_{mz}C\phi S\theta \\ m_{my}C\phi + m_{mz}S\phi \\ m_{mx}S\theta - m_{my}S\phi C\theta + m_{mz}C\phi C\theta \end{bmatrix} \quad . \end{aligned} \tag{4.9}$$

From figure 2.2 it can be deduced that:

$$\psi = arctan(\frac{m_y}{m_x}) \quad . \tag{4.10}$$

Now we have two estimations: one based on the gyroscopes data, which is suitable for fast movements from the UAV, but based on an integration, which accumulates error; and one from the accelerometers and magnetometers, which makes use of the assumption that $\dot{\boldsymbol{v}} = 0$, being suitable only for slow dynamics. This is based on absolute measurements, avoiding build up error.

To combine the two estimation data, the following complementary filter, developed in (MAHONY et al., 2008), is used:

$$\begin{aligned} \dot{\hat{\boldsymbol{R}}}_{\mathfrak{B}}^{\mathfrak{I}} &= \hat{\boldsymbol{R}}_{\mathfrak{B}}^{\mathfrak{I}} \boldsymbol{S}(\boldsymbol{\omega}_{\boldsymbol{m}}^{\mathfrak{B}} - \hat{\boldsymbol{b}}_{\omega}^{\mathfrak{B}} + K_p \Omega(\tilde{\boldsymbol{R}}_{\mathfrak{B}}^{\mathfrak{I}})) \\ \dot{\hat{\boldsymbol{b}}} &= -K_I \Omega(\tilde{\boldsymbol{R}}_{\mathfrak{B}}^{\mathfrak{I}}) \\ \tilde{\boldsymbol{R}} &= \hat{\boldsymbol{R}}_{\mathfrak{I}}^{\mathfrak{B}} \bar{\boldsymbol{R}}_{\mathfrak{B}am}^{\mathfrak{I}} \end{aligned} \quad , \tag{4.11}$$

where $\hat{\boldsymbol{b}}_{\omega}$ is the estimation of the gyroscope bias, $\tilde{\boldsymbol{R}}$ is the estimation error based on the accelerometer and magnetometer estimation, $\Omega(\tilde{\boldsymbol{R}}_{\mathfrak{B}}^{\mathfrak{I}})$ the correction term of the filter, $\bar{\boldsymbol{R}}_{\mathfrak{B}am}^{\mathfrak{I}}$ is the rotation matrix from frame $\mathfrak{B}$ to $\mathfrak{I}$ that uses the angles found with the estimation of the accelerometers and magnetometers, and K_I, K_p are the filter's feedback gains for the closed loop system in figure 4.2.

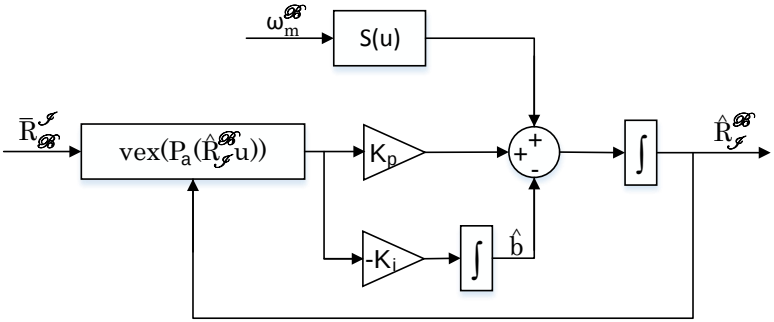


Figure 4.2 – Fusion filter structure.

The filter has an internal loop for the estimation of the gyroscopes bias. For the correction term, in (MAHONY et al., 2008) an expression is chosen based on the filter's Lyapunov stability, in order to guarantee an exponential stability. The chosen expression is:

$$vex(P_a(\tilde{\boldsymbol{R}})) \quad , \tag{4.12}$$

where operator $P_a(.)$ refers to the skew-symmetric projection

$$P_a(\boldsymbol{H}) = \frac{1}{2}(\boldsymbol{H} - \boldsymbol{H}')$$

and $vex(.)$ is the inverse operation of $S(.)$, e.g., $vex(\boldsymbol{S}(\boldsymbol{w})) = \boldsymbol{w}$.

As the fusion filter runs in the embedded system, computational cost is an important factor. Since the estimation algorithm has several matrix operations and a matrix integration - it has a high computational cost - its quaternion equivalent form is much more computational efficient. Its development is also based on (MAHONY et al., 2008).

Defining the set of unit quaternions as $Q = \{\boldsymbol{q} = (s, \boldsymbol{v}) \in R \times R^3 : \|\boldsymbol{q}\| = 1\}$ with the operation

$$\boldsymbol{q_a} \otimes \boldsymbol{q_b} = \begin{bmatrix} s_a s_b - \boldsymbol{v_a'} \boldsymbol{v_b} \\ s_a \boldsymbol{v_b} + s_b \boldsymbol{v_a} + \boldsymbol{v_a} \times \boldsymbol{v_b} \end{bmatrix} \quad . \tag{4.13}$$

The rotation kinematics from (2.12) has its quaternion equivalent form as:

$$\dot{\boldsymbol{q}} = \frac{1}{2}\boldsymbol{q} \otimes \boldsymbol{p}(\boldsymbol{w}) \quad . \tag{4.14}$$

The fusion filter (4.11) becomes:

$$\begin{array}{c} \dot{\boldsymbol{q}} = \frac{1}{2}\boldsymbol{q} \otimes \boldsymbol{p}(\boldsymbol{\omega} - \hat{\boldsymbol{b}} + 2K_p \tilde{s}\tilde{\boldsymbol{v}}) \\ \hat{\boldsymbol{b}} = -2K_I \tilde{s}\tilde{\boldsymbol{v}} \\ 2\tilde{s}\tilde{\boldsymbol{v}} = vex(P_a(\tilde{\boldsymbol{R}})) \end{array} \quad . \tag{4.15}$$

The quaternion fusion filter (4.15) has less matrix operations and only one vector integration, making it more computationally efficient than (4.11). Therefore, this is the estimation algorithm to be implemented into the embedded system.

As the controllers use Euler angles parametrization, a transformation from quaternions is required. The transformation also uses only simple multiplications and still justifies the use of quaternions for the estimation algorithm. Considering $\boldsymbol{v} = \begin{bmatrix} v_1 & v_2 & v_3 \end{bmatrix}$, the transformation is:

$$\phi = arctan\left(\frac{v_2 v_3 + s v_1}{\frac{1}{2} - v_1^2 - v_2^2}\right) \tag{4.16}$$

$$\theta = arcsin(-2v_1v_3 - sv_2) \tag{4.17}$$

$$\psi = arctan\left(\frac{v_1v_2 + sv_3}{\frac{1}{2} - v_2^2 - v_3^2}\right) \quad . \tag{4.18}$$

This concludes the estimation of the attitude. Now from equation (2.16) and the gyroscopes measurements in equation (4.1) the angular velocity of the UAV is obtained as follows:

$$\dot{\boldsymbol{\eta}} = \begin{bmatrix} \dot{\hat{\phi}} \\ \dot{\hat{\theta}} \\ \dot{\hat{\psi}} \end{bmatrix} = \boldsymbol{W}_{\boldsymbol{\eta}}^{-1}\boldsymbol{\omega}_m^{\mathfrak{B}} \quad . \tag{4.19}$$

4.2.2 Linear position and velocity

The sensors used to estimate the x and y states are the accelerometers. However, the IMU used is of low cost (and quality) and has a considerable noise that is propagated throughout the estimation. Also, the estimated attitude developed in section 4.2.1, which would be used in the position estimation, already has a significant noise on its own. To estimate the linear position, a double integration of the acceleration is required, in which the estimation error builds up, providing an unreliable estimation. Therefore, no estimation of the x and y states will be made until the localization system is implemented, e.g., a GPS or a camera vision system.

On the other hand, the sonar provides an estimation of its distance to an obstacle and can be used to estimate the aircraft's height (z) with respect to the ground. The sonar is fixed to the the UAV's structure facing down. In order to obtain an reliable estimation, flight tests are performed in an open area and constraining the maximum altitude to two meters.

The sonar's mounting point in the UAV is at the origin of the sonar frame $\mathfrak{S}$, where the measurements are taken. Note that this measurement does not give the desired height if the UAV has a roll or pitch rotation. The desired value is the distance from the body frame to the ground.

It is assumed that frame $\mathfrak{S}$ is centered in relation to the origin of the body frame and that both posses the same frame alignment,

having only a displacement in the $Z_{\mathfrak{B}}$ axis of d_{SB}. Therefore a measurement $\boldsymbol{h}^{\mathfrak{S}} = \begin{bmatrix} 0 & 0 & s_m \end{bmatrix}'$ taken at the sonar frame can be translated to the body frame with:

$$\boldsymbol{h}^{\mathfrak{B}} = \boldsymbol{h}^{\mathfrak{S}} + \begin{bmatrix} 0 \\ 0 \\ d_{SB} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ s_m + d_{SB} \end{bmatrix} \quad .$$

Now expressing $\boldsymbol{h}^{\mathfrak{B}}$ we obtain:

$$\boldsymbol{h}^{\mathfrak{I}} = \boldsymbol{R}^{\mathfrak{I}}_{\mathfrak{B}} \boldsymbol{h}^{\mathfrak{B}} = \begin{bmatrix} (C\psi S\theta C\phi + S\psi S\phi)(s_m + d_{SB}) \\ (S\psi S\theta C\phi - C\psi S\phi)(s_m + d_{SB}) \\ C\phi C\theta (s_m + d_{SB}) \end{bmatrix} \quad . \tag{4.20}$$

Only the third element is valid, yielding:

$$z = C\phi C\theta (s_m + d_{SB}) \quad . \tag{4.21}$$

There is no direct measurement of the vertical linear velocity, thus an estimation of $\dot{z}$ is made by differentiating z with respect to time.

4.2.3 Rotor Angles

The servomotors used can provide a measurement of their angles. However, they do not have torque control. It is only possible to control their angular position and speed. Hence, states α_R and α_L, even though they are measured, are used as control inputs.

4.3 Tiltrotor Flight Tests

From section 4.2, the available estimated states are:

$$\begin{bmatrix} z & \phi & \theta & \psi & \dot{z} & \dot{\phi} & \dot{\theta} & \dot{\psi} \end{bmatrix}'$$

and the control inputs:

$$\begin{bmatrix} f_R & f_L & \alpha_R & \alpha_L \end{bmatrix}' \quad .$$

These are all the variables needed for the controllers developed in section 3.4.

For security during experiments, a radio receiver is incorporated to the aircraft, enabling a human operator to:

- change the ϕ reference;
- change the θ reference;
- change the z reference;
- enable/disable height control;
- trigger an emergency shutdown (shutdown all actuators).

Since there is no control of the x and y linear positions, the UAV will drift through out the XY plane. To keep these variables under a limited space, the human operator uses the ϕ and θ reference angles.

4.3.1 Backstepping Nonlinear Control for the Height and Rotational System

The controller from section 3.4.2 was implemented in the development board using the C programming language. After some experiments, the controllers gains were iteratively changed to adapt to the unmodeled dynamics and parametric errors, which became:

$$\begin{array}{lll} K_{vz} = 3 & , & K_{pz} = 4.5 \\ K_{v\phi} = 8 & , & K_{p\phi} = 80 \\ K_{v\theta} = 5 & , & K_{p\theta} = 40 \\ K_{v\psi} = 6 & , & K_{p\psi} = 18 \end{array} . \tag{4.22}$$

4.3.2 LQR Control for the Height and Rotational System

The control developed in section 3.4.1 was also implemented using the C programming language. The gain matrices $\boldsymbol{Q}$ and $\boldsymbol{R}$ were

iteratively changed during the flight tests to find a better system response, which were set as follows:

$$\boldsymbol{R} = \begin{bmatrix} 1/(3.69^2) & 0 & 0 & 0 \\ 0 & 1/(3.69^2) & 0 & 0 \\ 0 & 0 & 5/(\frac{\pi}{2})^2 & 0 \\ 0 & 0 & 0 & 5/(\frac{\pi}{2})^2 \end{bmatrix} , \tag{4.23}$$

and

$$\begin{aligned} \boldsymbol{Q} = diag([30/1^2 \quad & 30/(\tfrac{\pi}{3})^2 \quad 30/(\tfrac{\pi}{3})^2 \quad 8/(\pi)^2 \\ & 1/1^2 \quad 1/(\tfrac{\pi}{2})^2 \quad 1/(4\pi)^2 \quad 1/(\tfrac{\pi}{2})^2]) \quad . \end{aligned} \tag{4.24}$$

4.3.3 Comparative Experiment Results

The flight tests were performed in a wide area with no obstacles, as shown in figure 4.3. It is important to notice that the protective dome was not modeled in CAD, which is not incorporated in the parameters from table 2.1. The only parameter that considers the dome is the total mass, while the rest are parametric uncertainties.

Figure 4.3 – Tiltrotor during experimental flight test.

As discussed in section 4.3, a human operator can control the

UAV via a radio controller by changing the roll and pitch reference angles with the intention to limit the aircraft's displacement in the XY plane. The drawback of this procedure is that the human operator acts as an outer loop controller for the x and y motion. Therefore, a direct comparison between the controllers cannot be made.

Figures 4.4 to 4.7 present the experimental flight results obtained for the LQR controller. While in flight, the references used were also recorded and a simulation using them (assuming no disturbances) was carried out for comparison with the experimental results. Note that the states do not track the reference, given that there is no integral action.

Also important to be noticed are the oscillations in the signals. These come from several different reasons that add up to this result: mechanical oscillations in the structure caused by the motors; estimation errors and phase shift caused by the complementary filter in section 4.2.1; unmodeled dynamics; parameters estimation errors; and external disturbances. The oscillations however are small in magnitude, except for the pitch angle during takeoff and it shows that the pitch angle is indeed the most difficult state to control, as it is directly associated with the servomotors that have a much slower dynamic than the brushless motors. However, after the initial stabilization, which requires bigger steps from the servomotors and therefore a bigger delay, the pitch angle behavior is sufficient to achieve a stable flight.

The experiments results while using backstepping control are illustrated in figures 4.8 to 4.11, where the same effects can be noticed.

Table 4.1 gives the ISE of the flight test performed with the LQR controller and table 4.2 for the experiments with the backstepping controller. The height and roll control presented considerable small errors, which was somehow expected due to the fact that they do not depend heavily on the servomotor response. The yaw angle has a considerate error, which is due to its slower dynamics and because the yaw has a slightly worse estimation than the roll and pitch angles, due to magnetic interferences.

As expected, the pitch has the worst performance. Note that it is the only variable that the LQR has a better response than the backstepping. Given the sensitivity of the system in the pitch angle, the fact that the LQR model considers the displacement in the center of mass was enough to present a superior performance than the nonlinear

control in this variable.

	Simulation	Experimental
ISE_z	0.0199	0.0480
ISE_ϕ	0.0001	7.6204
ISE_θ	0.0003	49.7579
ISE_ψ	0.0001	14.2409

Table 4.1 – Integral Squared Error for the experimental LQR controller.

	Simulation	Experimental
ISE_z	0.0085	0.0042
ISE_ϕ	0.0001	0.8245
ISE_θ	0.0194	81.4082
ISE_ψ	0.0001	10.2817

Table 4.2 – Integral Squared Error for the experimental nonlinear backstepping controller.

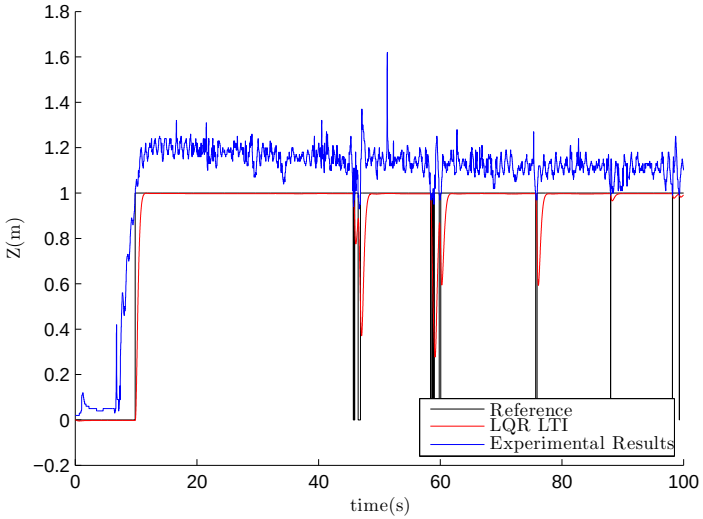


Figure 4.4 – Tiltrotor flight test height with the LQR control.

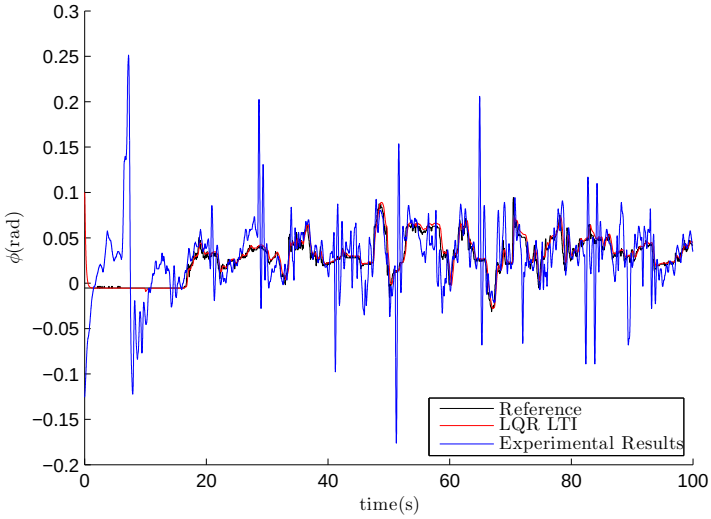


Figure 4.5 – Tiltrotor flight test roll angle with the LQR control.

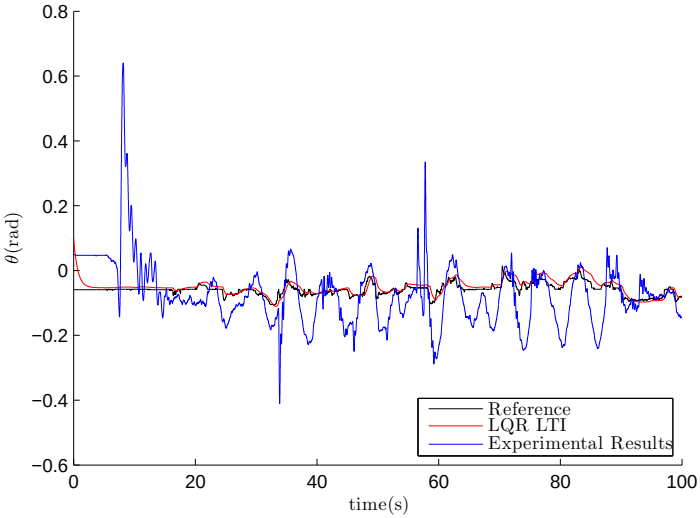


Figure 4.6 – Tiltrotor flight test pitch angle with the LQR control.

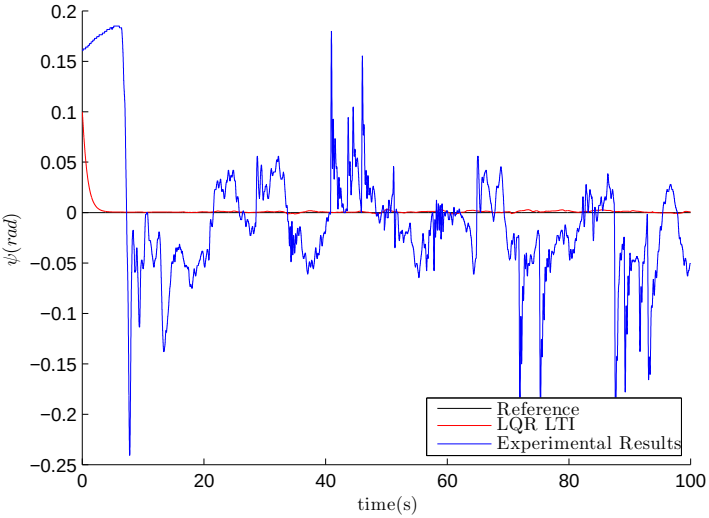


Figure 4.7 – Tiltrotor flight test yaw angle with the LQR control.

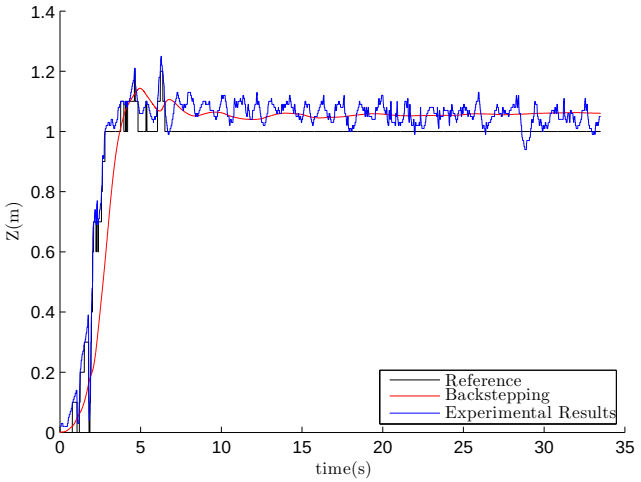


Figure 4.8 – Flight test height with the nonlinear backstepping control.

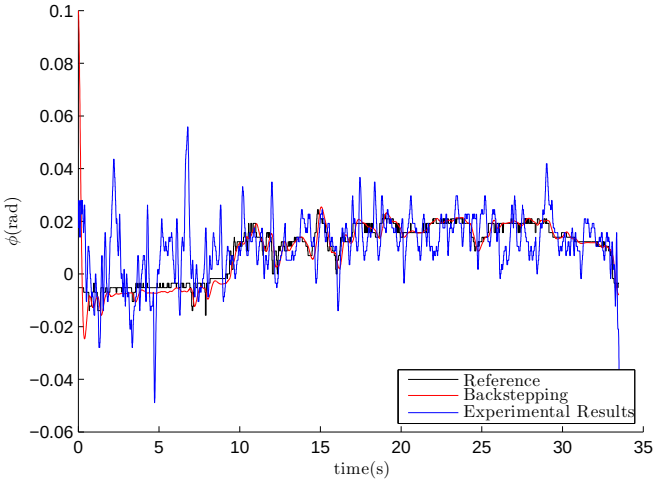


Figure 4.9 – Flight test roll angle with the nonlinear backstepping control.

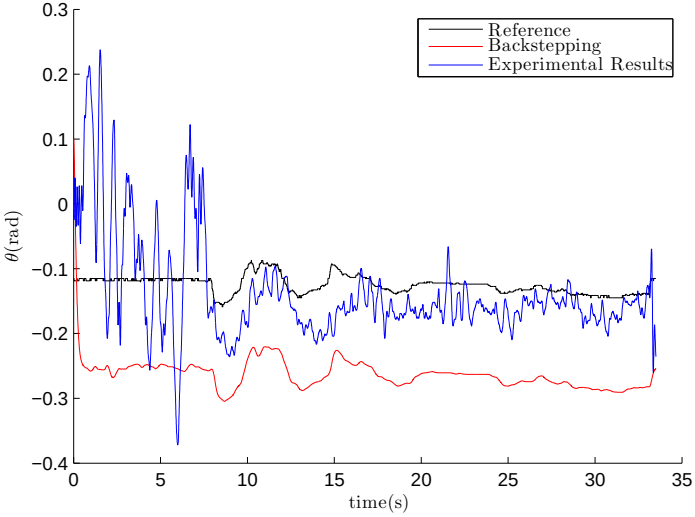


Figure 4.10 – Flight test pitch angle with the nonlinear backstepping control.

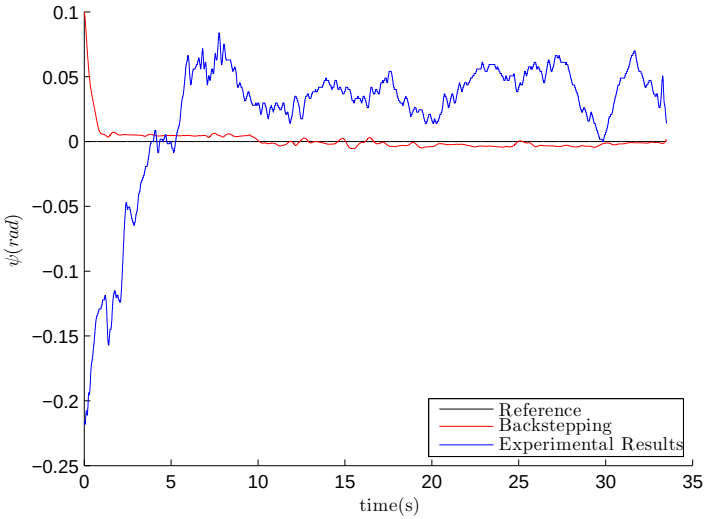


Figure 4.11 – Flight test yaw with the nonlinear backstepping control.

4.4 Summary

This chapter presented the experimental results obtained in this work with the ProVANT tiltrotor and the controllers designed in section 3.4. The states required by the controllers were estimated through a complementary filter making use of a sonar, gyroscopes, accelerometers, and magnetometers measurements. Both control laws were able to stabilize the UAV with satisfactory performances.

Chapter 5

Conclusions

This master thesis tackled the modeling and control of a birotor tiltrotor UAV, aiming at solving the path tracking and hovering problems. The results presented along the thesis were obtained by means of simulations and experimental flights.

The initial step addressed the UAV dynamic modeling. Literature models were expanded in order to consider the tiltrotor as a multibody system. It results in an eight DOF model, opposed to the literature's six DOF model, highly nonlinear and coupled. Furthermore, the developed model considered several other peculiarities: displacement in the UAV's center of mass, a fixed tilt angle in the lateral axis, and the propeller's air drag effect. Therefore, a complex simulation environment closer to the real system could be created.

To solve the path tracking problem, five linear optimal controllers were designed:

- A LQR LTI control was chosen for its properties of infinite gain margin and at least sixty degrees phase margin. The controller matrices were designed according to Bryson's law.
- A new LQR control using LMI formulation was later designed to make the previous controller more robust by including the time varying elements from the linearized system into its formulation, as well as an uncertainty to the mass. Lastly the control inputs saturations were also included into the system in the form of

LMIs, composing a more robust controller than the LQR LTI. The controller matrices were arranged so that the cost function became similar to the one used in the LQR LTI.

- A control that minimizes the $\mathcal{H}_\infty$ norm was designed in order to minimize the disturbance effects in the system. It was also implemented using LMIs considering parametric uncertainties.

- A multi-objective $\mathcal{H}_2/\mathcal{H}_\infty$ control was developed in order to guarantee a $\mathcal{H}_\infty$ norm lesser than a fixed value, which was chosen close to the norm value minimized in the pure $\mathcal{H}_\infty$ controller, while minimizing the $\mathcal{H}_2$ norm. This structure maintains the property of disturbance attenuation of the $\mathcal{H}_\infty$ control and improves the control system performance through the $\mathcal{H}_2$ norm minimization. It was also implemented using LMI formulation.

- Finally, a LQR control for the LTV system was designed . This control has a high computational cost but it was mainly designed to be compared with the other designed controllers. Performed simulation analysis allowed evaluating how much it is lost by ignoring the time varying components of the linear model.

Additionally, a well-known backstepping nonlinear controller (CHOWDHURY et al., 2012) was also implemented for comparison matters. Some adaptations to the controller equations were made in order to improve it, as well as a new gain design.

All developed controllers were simulated with the nonlinear model in the presence of external disturbances and parametric uncertainties, namely a 30% increase in the inertia moments. All controllers were able to track two proposed trajectories: a circular trajectory and a trajectory with abrupt changes. LQR LTI and LQR LTV controllers presented the best performances, but they do not provide the same robustness of the LMI controllers. The inclusion of uncertainties and saturations into the LMI controllers design restricts the solution, meaning a trade-off between robustness and performance had to be made. However, even with such restrictions, the H_2/H_∞ control showed a performance close to the LQRs and consequently was considered as the most suitable controller to be adopted in the UAV.

The nonlinear backstepping control presented a worse performance when compared to the aforementioned designs, specially in

relation to the $\theta(t)$ state. This is due to the several simplifications assumed in the controller design.

The thesis also addressed the design of controllers that only stabilize the height and the rotational system - a simpler problem when compared to the path track controllers. They were designed to serve as intermediate solutions, allowing the team to make the first experimental flights. Such controllers consist of a LQR LTI and an adaptation of the backstepping nonlinear control. Simulations analysis show a better response from the LQR control.

The UAV's attitude was estimated using a complementary filter to fuse the data from gyroscopes, accelerometers, and magnetometers sensors. The filter was based on the work of (MAHONY et al., 2008) and its quaternion equivalent was implemented in the UAV embedded system. The quaternion formulation was chosen for its well-known computational efficiency. In addition, a sonar is also used to measure the UAV height, so that all information required by the stabilization controllers are available.

Experimental flights performed with ProVANT's first prototype showed that the designed controllers can stabilize the tiltrotor in a real scenario, which has unmodeled dynamics, mechanical oscillations, estimation errors, wind gusts, and other disturbances.

The experimental results show a better response from the backstepping control in relation to the height, roll, and yaw, although the LQR still has a satisfactory response. The pitch angle however shows a better performance from the LQR controller. This highlights how sensitive the pitch angle is in this system, as the simplifications made for the design of the backstepping controller appear with higher intensity for this variable.

A drawback from the current UAV prototype comes from the lack of a positioning system and a servomotor controlled by torque. These components are required by the tracking controllers developed in this thesis, making it not possible to make experimental evaluations of them. However, both a GPS and an adequate servomotor should be available in a new UAV prototype in the near future.

5.1 Future Works

This section brings suggestions for future works related to this thesis. They are:

- Experimental flight tests with the developed path tracking controllers, which will be possible in the near future with ProVANT's tiltrotor, after the implementation of the GPS and servomotors with torque input.
- Design and implement other LTV control strategies like Gain-schedule controllers for the time-varying linear model developed in this thesis.
- Include a generic trajectory for the yaw angle instead of a constant value. This includes more time varying parameters in the linearized system.
- Design nonlinear $\mathcal{H}_2$ and $\mathcal{H}_\infty$ control strategies that take advantage of the decoupling in control inputs.
- Development and implementation of a Kalman filter estimator. A comparison with the implemented complementary filter can be made in order to evaluate if the performance gain is worth the additional complexity.
- The use of a fixed camera system, exterior to the UAV, to perform an attitude estimation. A better estimation can be achieved with this system, which can be used as a reference to evaluate and improve the complementary and Kalman filters.

Bibliography

ALAMO, T.; NORMEY-RICO, J.; ARAHAL, M. Introducing linear matrix inequalities in a control course. Proceedings of Advances in Control Education, ACE'06, Madrid, Spain, v. 7, p. 205–210, 2006.

ALMEIDA, M. *Control Strategies of a Tilt-rotor UAV for Load Transportation*. Dissertação (Mestrado) — Federal University of Minas Gerais, Brazil, 2014.

ALMEIDA, M.; DONADEL, R.; RAFFO G.V. BECKER, L. Full control of a tiltrotor uav for load transportation. In: *Anais do XX Congresso Brasileiro de Automática, CBA 2014*. Belo Horizonte, Brazil: SBA, 2014. p. 2097–2104.

BODANESE, J. *Infraestrutura de comunicação sem fio para um veículo aéreo não tripulado de curto alcance*. Dissertação (Mestrado) — Federal University of Santa Catarina, Brazil, 2014.

BOYD, S.; GHAOUI, L. E.; FERON, E.; BALAKRISHNAN, V. *Linear Matrix Inequalities in System & Control Theory (Studies in Applied Mathematics, Volume 15)*. Philadelphia, USA: Society for Industrial & Applied, 1994.

CASTILLO, P.; LOZANO, R.; DZUL, A. Stabilization of a mini rotorcraft with four rotors. *Control Systems, IEEE*, v. 25, n. 6, p. 45–55, 2005. ISSN 1066-033X.

CHEN, C.-T. *Linear System Theory and Design (Oxford Series in Electrical and Computer Engineering)*. New York, USA: Oxford University Press, 1998.

CHOWDHURY, A. B.; KULHARE, A.; RAINA, G. Back-stepping control strategy for stabilization of a tilt-rotor uav. In: *Proc. of the 24th CCDC'2012*. Taiyuan, China: IEEE, 2012. p. 3475–3480.

DONADEL, R.; ALMEIDA, M.; RAFFO G.V. BECKER, L. Path tracking control of a small scale tiltrotor unmanned aerial vehicle. In: *Anais do XX Congresso Brasileiro de Automática, CBA 2014*. Belo Horizonte, Brasil: SBA, 2014. p. 1450–1457.

DONADEL, R.; RAFFO G.V. BECKER, L. Modeling and control of a tiltrotor uav for path tracking. In: *Proceedings of the 19th IFAC World Congress, 2014*. Cape Town, South Africa: IFAC, 2014. p. 3839–3844.

DRWIEGA, A. Future vertical lift: an overview. *Rotor & Wing*, n. 201305, p. 28–33, May 2013.

DULLERUD, G. E.; PAGANINI, F. *A Course in Robust Control Theory: A Convex Approach (Texts in Applied Mathematics)*. New York, USA: Springer-Verlag, 2005. ISBN 0387989455.

FAA, U. F. A. F. *FAA Aerospace Forecast Fiscal Years 2014-2034*. 2014.

GINSBERG, J. *Engineering dynamics*. Cambridge New York: Cambridge University Press, 2008. ISBN 9780521883030.

GONCALVES, F. *Projeto da arquitetura de software embarcado de um veículo aéreo não tripulado*. Dissertação (Mestrado) — Federal University of Santa Catarina, Brazil, 2014.

GONCALVES, F.; BODANESE, J.; DONADEL, R.; RAFFO, G.; NORMEY-RICO, J.; BECKER, L. Small scale uav with birotor configuration. In: *Unmanned Aircraft Systems, 2013 International Conference on*. Atlanta, USA: ICUAS, 2013. p. 761–768.

GONCALVES, F.; DONADEL, R.; BECKER, L.; RAFFO, G. Assessing the use of matlab simulink on the development process of an unmanned aerial vehicle. In: *Third Workshop on Design, Modeling and Evaluation of Cyber Physical Systems*. Philadelphia, USA: CyPhy, 2013.

GRESS, G. Using dual propellers as gyroscopes for tilt-prop hover control. In: *Proc. AIAA Biennial International Powered Lift Conference Exhibit*. Williamsburg, USA: American Institute of Aeronautics and Astronautics, 2002.

GRESS, G. *VTOL aircraft control using opposed tilting of its dual propellers or fans*. Google Patents, abr. 13 2004. US Patent 6,719,244. <http://www.google.im/patents/US6719244>.

JANSEN, F.; RAMIREZ-SERRANO, A. Agile unmanned vehicle navigation in highly confined environments. In: *Proc. of the IEEE International Conference on Systems, Man and Cybernetics*. Anchorage, USA: IEEE, 2011. p. 2381–2386. ISSN 1062-922X.

JOHNSON, M.; GRIMBLE, M. Recent trends in linear optimal quadratic multivariable control system design. *Control Theory and Applications, IEE Proceedings D*, v. 134, n. 1, p. 53–71, 1987.

KENDOUL, F.; FANTONI, I.; LOZANO, R. Modeling and control of a small autonomous aircraft having two tilting rotors. In: *Proc. of the 44th IEEE CDC-ECC '05*. Seville, Spain: IEEE, 2005. p. 8144–8149.

LEDYAEV, Y. S. On analytical solutions of matrix riccati equations. *Proceedings of the Steklov Institute of Mathematics*, v. 273, n. 1, p. 214–228, fev. 2011. ISSN 0081-5438.

LEISHMAN, R.; MACDONALD, J.; BEARD, R.; MCLAIN, T. Quadrotors and accelerometers: State estimation with an improved dynamic model. *Control Systems, IEEE*, v. 34, n. 1, p. 28–41, Feb 2014. ISSN 1066-033X.

LEWIS, F. L.; DAWSON, D. M.; ABDALLAH, C. T. *Robot Manipulator Control: Theory and Practice (Automation and Control Engineering)*. New York, USA: CRC Press, 2003. ISBN 0824740726.

MACKENROTH, U. *Robust Control Systems: Theory and Case Studies*. Berlin: Springer, 2010. ISBN 3642058914.

MAHONY, R.; HAMEL, T.; PFLIMLIN, J.-M. Nonlinear complementary filters on the special orthogonal group. *Automatic Control, IEEE Transactions on*, v. 53, n. 5, p. 1203–1218, June 2008. ISSN 0018-9286.

MARTIN, D.; DEMO, J.; DANIEL, C. *The history of the xv-15 tiltrotor Research aircraft: from concept to flight*. Washington, USA: NASA, 2000.

NAIDU, D. S. *Optimal Control Systems (Electrical Engineering Series)*. USA: CRC Press, 2002.

NOTARSTEFANO, G.; HAUSER, J. Modeling and dynamic exploration of a Tilt-Rotor VTOL aircraft. *8th IFAC Symposium on Nonlinear Control Systems*, p. 119–124, 2010.

OGATA, K. *Modern control engineering*. Upper Saddle River, N.J: Prentice Hall, 1997. ISBN 978-0132273077.

PAPACHRISTOS, C.; ALEXIS, K.; NIKOLAKOPOULOS, G.; TZES, A. Model predictive attitude control of an unmanned tilt-rotor aircraft. In: *Proc. of the 6th International Conference of the International Society for Industrial Ecology*. Berkley, USA: International Society for Industrial Ecology, 2011. p. 922–927.

PAPACHRISTOS, C.; ALEXIS, K.; TZES, A. Design and experimental attitude control of an unmanned tilt-rotor aerial vehicle. In: *Proc. of the 15th international conference on advanced robotics*. Tallinn, Estonia: ICAR, 2011. p. 465–470.

RAFFO, G. V.; ORTEGA, M. G.; RUBIO, F. R. MPC with nonlinear $\mathcal{H}_\infty$ control for path tracking of a quad-rotor helicopter. In: *Proceedings of the 17th IFAC World Congress, 2008*. Seoul, South Korea: IFAC, 2008. p. 8564–8569.

RAFFO, G. V.; ORTEGA, M. G.; RUBIO, F. R. Nonlinear h-infinity controller for the quad-rotor helicopter with input coupling. In: *Proceedings of the 18th IFAC World Congress, 2011*. Milano, Italy: IFAC, 2011. p. 13834–13839.

SANCHEZ, A.; ESCAREÑO, J.; GARCIA, O.; LOZANO, R. Autonomous Hovering of a Noncyclic Tiltrotor UAV: Modeling, Control and Implementaion. In: *Proc. of the 17th IFAC World Congress*. Seoul, South Korea: IFAC, 2008. p. 803–808.

SHABANA, A. A. *Dynamics of Multibody Systems*. New York, USA: Cambridge University Press, 2010.

SPONG, M. W.; HUTCHINSON, S.; VIDYASAGAR, M. *Robot Modeling and Control*. USA: Wiley, 2005. ISBN 0471649902.

STRANG, G. *Introduction to Linear Algebra, Fourth Edition*. USA: Wellesley Cambridge Press, 2009.

TROFINO, A.; COUTINHO, D.; BARBOSA., K. A. *Sistemas multivariaveis: Uma aborgadem via lmis (versao preliminar)*. 2003.

Appendix A

Mathematical Definitions

A.1 Skew-Symmetric Matrix

A square matrix $\boldsymbol{S}$ is skew-symmetric (also known as anti-symmetric matrix) if and only if

$$\boldsymbol{S} + \boldsymbol{S}' = 0 \tag{A.1}$$

A 3-square skew-symmetric matrix S, as a consequence of A.1, only has three independent elements. Therefore, a skew-symmetric matrix on R^3 can be associated with a vector in R^3. Consider the vector $\boldsymbol{a} = (a_x, a_y, a_z)$. The matrix $\boldsymbol{S}(\boldsymbol{a})$ is then defined as

$$\boldsymbol{S}(\boldsymbol{a}) = \begin{bmatrix} 0 & -a_z & a_y \\ a_z & 0 & -a_x \\ -a_y & a_x & 0 \end{bmatrix} \quad , \tag{A.2}$$

where $\boldsymbol{S}(.)$ is the skew-symmetric matrix operator.

Some useful skew-symmetric matrices properties are (SPONG et al., 2005):

1.The operator $\boldsymbol{S}(.)$ is linear, i.e.,

$$\boldsymbol{S}(\alpha\boldsymbol{a}+\beta\boldsymbol{b})=\alpha\boldsymbol{S}(\boldsymbol{a})+\beta\boldsymbol{S}(\boldsymbol{b}) \tag{A.3}$$

for any vectors $\boldsymbol{a}$ and $\boldsymbol{b}$ belonging to R^3 and scalars α and β.

2.For any vector $\boldsymbol{a}$ and $\boldsymbol{p}$ belonging to R^3,

$$\boldsymbol{S}(\boldsymbol{a})\boldsymbol{p}=\boldsymbol{a}\times\boldsymbol{p} \tag{A.4}$$

3.For $\boldsymbol{R}\in \mathrm{SO}(3)$ and $\boldsymbol{a}\in R^3$

$$\boldsymbol{R}\boldsymbol{S}(\boldsymbol{a})\boldsymbol{R}'=\boldsymbol{S}(\boldsymbol{R}\boldsymbol{a}) \tag{A.5}$$

4.For an $n\times n$ skew-symmetric matrix $\boldsymbol{S}$ and any vector $\boldsymbol{X}\in R^n$

$$\boldsymbol{X}'\boldsymbol{S}\boldsymbol{X}=0 \tag{A.6}$$

A.2 Schur's Complement

Defining $\boldsymbol{x}$ a vector with the system variables, $\boldsymbol{A}(\boldsymbol{x})$ and $\boldsymbol{C}(\boldsymbol{x})$ symmetric matrices, $\boldsymbol{B}(\boldsymbol{x})$ a matrix that is affine on $\boldsymbol{x}$, then the Schur complement of the expression:

$$\boldsymbol{A}(\boldsymbol{x})-\boldsymbol{B}(\boldsymbol{x})\boldsymbol{C}(\boldsymbol{x})^{-1}\boldsymbol{B}(\boldsymbol{x})'>0\ \ ,\ \ \boldsymbol{C}(\boldsymbol{x})>0 \tag{A.7}$$

is (BOYD et al., 1994):

$$\begin{bmatrix}\boldsymbol{A}(\boldsymbol{x}) & \boldsymbol{B}(\boldsymbol{x})'\\ \boldsymbol{B}(\boldsymbol{x}) & \boldsymbol{C}(\boldsymbol{x})\end{bmatrix}>0 \tag{A.8}$$

Equations A.7 and A.8 are equivalent, although only the latter is a LMI.

Appendix B

Control Design Matrices

B.1 Full Model Linearization Matrices

Matrices originated from the full model linearization in equation 3.9:

$$\boldsymbol{A} = \begin{bmatrix} 0_{8,8} & I_8 \\ \tilde{\boldsymbol{A}}_{\mathbf{0}} + \tilde{\boldsymbol{A}}_{\mathbf{1}}\ddot{x}_r(t) + \tilde{\boldsymbol{A}}_{\mathbf{2}}\ddot{y}_r(t) + \tilde{\boldsymbol{A}}_{\mathbf{3}}\ddot{z}_r(t) & 0_{8,8} \end{bmatrix} \tag{B.1}$$

where $\ddot{x}_r(t)$, $\ddot{y}_r(t)$ and $\ddot{z}_r(t)$ are accelerations from the desired system's trajectory and

$$\tilde{\boldsymbol{A}}_{\mathbf{0}} = \begin{bmatrix} 0 & 0 & 0 & -1.10E{-}6 & 9.81 & 0.0186 & 8.398 & 8.643 \\ 0 & 0 & 0 & -9.782 & 9.43E{-}7 & 0.0236 & 0.403 & -0.416 \\ 0 & 0 & 0 & 5.40E{-}6 & 3.29E{-}9 & -0.0001.96 & -0.0527 & -0.0460 \\ 0 & 0 & 0 & -0.0278 & -1.16E{-}5 & -0.378 & -6.175 & 6.336 \\ 0 & 0 & 0 & -1.59E{-}5 & -1.15E{-}6 & 0.298 & 56.695 & 58.505 \\ 0 & 0 & 0 & 0.00147 & 0.000243 & 0.0196 & 56.222 & -58.105 \\ 0 & 0 & 0 & 0.000146 & 2.26E{-}5 & -0.295 & -51.508 & -63.405 \\ 0 & 0 & 0 & -0.000110 & -1.96E{-}5 & -0.299 & -61.295 & -53.318 \end{bmatrix} \tag{B.2}$$

$\tilde{\boldsymbol{A}}_{\mathbf{1}} =$

$$\begin{bmatrix} 0 & 0 & 0 & -0.00129 & -4.40E{-}5 & -0.0374 & 0.119 & -0.119 \\ 0 & 0 & 0 & 5.81E{-}5 & -0.00630 & 0.000302 & 0.00570 & 0.00570 \\ 0 & 0 & 0 & 0.0192 & -0.000314 & 0.000262 & -0.000745 & 0.000631 \\ 0 & 0 & 0 & -0.000161 & 0.218 & -9.94E{-}5 & -0.0873 & -0.0868 \\ 0 & 0 & 0 & -0.115 & -0.000117 & -0.290 & 0.802 & -0.802 \\ 0 & 0 & 0 & 0.00196 & -0.0223 & 0.000425 & 0.795 & 0.797 \\ 0 & 0 & 0 & -0.00204 & -0.00109 & 0.289 & -0.728 & 0.870 \\ 0 & 0 & 0 & -0.00204 & 0.00129 & 0.289 & -0.867 & 0.731 \end{bmatrix} \tag{B.3}$$

$\tilde{\boldsymbol{A}}_{\mathbf{2}} =$

$$\begin{bmatrix} 0 & 0 & 0 & -0.101 & 0.000992 & -2.920 & 9.28419 & -9.258 \\ 0 & 0 & 0 & 0.000126 & -0.490 & 0.0236 & 0.445478 & 0.445 \\ 0 & 0 & 0 & 1.499 & -0.0246 & 0.0205 & -0.0583 & 0.0493 \\ 0 & 0 & 0 & -0.0126 & 17.0315 & -0.00794 & -6.823 & -6.784 \\ 0 & 0 & 0 & -8.962 & -0.00912 & -22.673 & 62.676 & -62.666 \\ 0 & 0 & 0 & 0.1532 & -1.744 & 0.0332 & 62.151 & 62.235 \\ 0 & 0 & 0 & -0.160 & -0.0854 & 22.587 & -56.943 & 67.915 \\ 0 & 0 & 0 & -0.160 & 0.101 & 22.587 & -67.762 & 57.111 \end{bmatrix} \tag{B.4}$$

$\tilde{\boldsymbol{A}}_{\mathbf{3}} =$

$$\begin{bmatrix} 0 & 0 & 0 & -1.13E{-}7 & 1. & 0.00190 & 0.856 & 0.881 \\ 0 & 0 & 0 & -0.997 & 9.62E{-}8 & 0.00240 & 0.0411 & -0.0424 \\ 0 & 0 & 0 & 5.506E{-}7 & 3.36E{-}10 & -2.00E{-}5 & -0.00537 & -0.00469 \\ 0 & 0 & 0 & -0.00284 & -1.19E{-}6 & -0.0386 & -0.629462 & 0.646 \\ 0 & 0 & 0 & -1.62E{-}6 & -1.17E{-}7 & 0.0304 & 5.77927 & 5.964 \\ 0 & 0 & 0 & 0.000150 & 2.48E{-}5 & 0.00200 & 5.73109 & -5.923 \\ 0 & 0 & 0 & 1.49E{-}5 & 2.30E{-}6 & -0.0301 & -5.25059 & -6.463 \\ 0 & 0 & 0 & -1.12E{-}5 & -2.00E{-}6 & -0.0304 & -6.248 & -5.435 \end{bmatrix} \tag{B.5}$$

and

$$\boldsymbol{B_u} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0.00127429 & -0.00123498 & -5.6786 & -5.67949 \\ 0.0830988 & -0.0805351 & 0.00713195 & -0.0114203 \\ 0.580813 & 0.580776 & 0.082228 & 0.0821831 \\ -0.501012 & 0.485556 & -0.0774871 & 0.151265 \\ 0.008452 & -0.00819125 & -90.9316 & -90.931 \\ 0.505526 & -0.48993 & 2.55582 & -2.22173 \\ 0.0362134 & -0.0350962 & 24504.9 & 90.3778 \\ -0.0516691 & 0.0500751 & 90.3778 & 24504.8 \end{bmatrix} \tag{B.6}$$

B.2 Height and Rotational System Model Linearization Matrices

Matrices originated from the height and rotational system linearization in equation 3.103:

$$\tilde{\boldsymbol{A}} = \begin{bmatrix} 0 & 0 & 0 & -1.10E{-}6 & 9.81 & 0.0186 & 8.398 & 8.643 \\ 0 & 0 & 0 & -9.782 & 9.43E{-}7 & 0.0236 & 0.403 & -0.416 \\ 0 & 0 & 0 & 5.40E{-}6 & 3.29E{-}9 & -0.0001.96 & -0.0527 & -0.0460 \\ 0 & 0 & 0 & -0.0278 & -1.16E{-}5 & -0.378 & -6.175 & 6.336 \\ 0 & 0 & 0 & -1.59E{-}5 & -1.15E{-}6 & 0.298 & 56.695 & 58.505 \\ 0 & 0 & 0 & 0.00147 & 0.000243 & 0.0196 & 56.222 & -58.105 \\ 0 & 0 & 0 & 0.000146 & 2.26E{-}5 & -0.295 & -51.508 & -63.405 \\ 0 & 0 & 0 & -0.000110 & -1.96E{-}5 & -0.299 & -61.295 & -53.318 \end{bmatrix} \tag{B.7}$$

$$\boldsymbol{B} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0.00127429 & -0.00123498 & -5.6786 & -5.67949 \\ 0.0830988 & -0.0805351 & 0.00713195 & -0.0114203 \\ 0.580813 & 0.580776 & 0.082228 & 0.0821831 \\ -0.501012 & 0.485556 & -0.0774871 & 0.151265 \\ 0.008452 & -0.00819125 & -90.9316 & -90.931 \\ 0.505526 & -0.48993 & 2.55582 & -2.22173 \\ 0.0362134 & -0.0350962 & 24504.9 & 90.3778 \\ -0.0516691 & 0.0500751 & 90.3778 & 24504.8 \end{bmatrix} \tag{B.8}$$

.

B.3 Full Model Gain Matrices

The gain matrices are all size 4×20, but their transpose is used here to better fit the pages.

$$\boldsymbol{K'_{lqrLTI}} = \begin{bmatrix} 0.0718 & -0.0139 & 3.3470 & 3.3743 \\ 34.4361 & -34.2234 & -0.0538 & 0.0446 \\ 29.7587 & 29.7846 & -0.0020 & -0.0018 \\ -24.7622 & 24.6379 & 0.0673 & -0.0596 \\ 0.0601 & 0.0121 & 2.5699 & 2.5877 \\ 0.1271 & -0.1259 & 1.1877 & -1.1816 \\ 0.1219 & -0.1257 & 2.1931 & 0.2497 \\ -0.1182 & 0.1129 & 0.2496 & 2.1982 \\ 0.0321 & -0.0026 & 1.4634 & 1.4757 \\ 14.7479 & -14.6611 & -0.0277 & 0.0235 \\ 10.2394 & 10.2510 & -0.0005 & -0.0005 \\ -1.6177 & 1.6149 & 0.0088 & -0.0083 \\ 0.0051 & 0.0047 & 0.1666 & 0.1662 \\ 0.0092 & -0.0089 & 0.2449 & -0.2453 \\ 0.0001 & -0.0001 & 0.0660 & 0.0011 \\ -0.0001 & 0.0000 & 0.0011 & 0.0660 \\ 0.0668 & -0.0177 & 3.1504 & 3.1741 \\ 32.8750 & -32.6662 & -0.0448 & 0.0364 \\ 32.7622 & 32.7841 & -0.0025 & -0.0023 \\ 0.3279 & -0.3240 & 2.4584 & -2.4401 \end{bmatrix} \tag{B.9}$$

$$\boldsymbol{K'_{lqrLMI}} = \begin{bmatrix} 0.0004 & -0.0016 & -1.8218 & -1.8374 \\ -6.3183 & 6.2684 & -0.1988 & 0.1967 \\ -8.1806 & -8.1998 & -0.0002 & -0.0004 \\ 6.7552 & -6.7126 & 0.1310 & -0.1296 \\ -0.0074 & -0.0104 & -0.9705 & -0.9770 \\ 0.1744 & -0.1733 & -0.5441 & 0.5435 \\ -0.0071 & 0.0071 & -0.5877 & -0.0629 \\ 0.0070 & -0.0069 & -0.0629 & -0.5888 \\ -0.0008 & -0.0017 & -0.6499 & -0.6556 \\ -2.9325 & 2.9106 & -0.0784 & 0.0776 \\ -3.8020 & -3.8110 & -0.0001 & -0.0002 \\ 0.8150 & -0.8130 & 0.0072 & -0.0071 \\ -0.0028 & -0.0034 & -0.0292 & -0.0288 \\ 0.0407 & -0.0405 & -0.0807 & 0.0809 \\ -0.0000 & 0.0000 & -0.0105 & -0.0004 \\ 0.0000 & -0.0000 & -0.0004 & -0.0105 \\ 0.0013 & -0.0008 & -2.0815 & -2.0985 \\ -6.0685 & 6.0197 & -0.2127 & 0.2103 \\ -7.6443 & -7.6620 & -0.0001 & -0.0003 \\ 0.3876 & -0.3853 & -1.2616 & 1.2517 \end{bmatrix} \tag{B.10}$$

$$\boldsymbol{K'_{H\infty}} = \begin{bmatrix} 0.1434 & 0.1412 & -2.3532 & -2.3972 \\ -3.6315 & 3.5925 & -0.2632 & 0.2648 \\ -7.6052 & -7.6679 & -0.0463 & -0.0681 \\ 5.7293 & -5.6698 & 0.2223 & -0.2315 \\ -0.1215 & -0.1289 & -1.8360 & -1.8632 \\ 0.1786 & -0.1677 & -0.4719 & 0.4716 \\ -0.0002 & 0.0010 & -0.7521 & -0.3173 \\ 0.0006 & -0.0005 & -0.3171 & -0.7575 \\ 0.0921 & 0.0910 & -1.0007 & -1.0188 \\ -2.0553 & 2.0343 & -0.1290 & 0.1308 \\ -3.7489 & -3.7816 & -0.0109 & -0.0206 \\ 0.8551 & -0.8456 & 0.0874 & -0.0886 \\ -0.0207 & -0.0214 & -0.0767 & -0.0762 \\ 0.0469 & -0.0458 & -0.0635 & 0.0644 \\ -0.0001 & -0.0001 & -0.0109 & -0.0029 \\ -0.0001 & -0.0001 & -0.0029 & -0.0110 \\ 0.0677 & 0.0653 & -2.4055 & -2.4522 \\ -3.1859 & 3.1591 & -0.1925 & 0.1932 \\ -6.9750 & -7.0328 & -0.0610 & -0.0824 \\ 0.3820 & -0.4301 & -0.9848 & 0.9637 \end{bmatrix} \tag{B.11}$$

$$\boldsymbol{K'_{H_2/H\infty}} = \begin{bmatrix} -0.0489 & 0.0240 & -2.7941 & -2.8124 \\ -28.3236 & 28.1778 & 0.0447 & -0.0363 \\ -20.6820 & -20.7241 & 0.0039 & 0.0038 \\ 20.8514 & -20.7580 & -0.0606 & 0.0539 \\ -0.0456 & -0.0047 & -2.2092 & -2.2216 \\ -0.1493 & 0.1483 & -1.1367 & 1.1282 \\ -0.1252 & 0.1317 & -2.1188 & -0.2613 \\ 0.1242 & -0.1161 & -0.2611 & -2.1229 \\ -0.0238 & 0.0073 & -1.2816 & -1.2895 \\ -12.7384 & 12.6689 & 0.0234 & -0.0196 \\ -8.9009 & -8.9148 & 0.0009 & 0.0008 \\ 1.6151 & -1.6111 & -0.0085 & 0.0081 \\ -0.0045 & -0.0038 & -0.1711 & -0.1710 \\ -0.0150 & 0.0148 & -0.2360 & 0.2362 \\ -0.0000 & 0.0000 & -0.0639 & -0.0027 \\ 0.0000 & 0.0000 & -0.0027 & -0.0640 \\ -0.0657 & 0.0197 & -3.0640 & -3.0837 \\ -31.6306 & 31.4571 & 0.0421 & -0.0328 \\ -26.5708 & -26.6168 & 0.0076 & 0.0074 \\ -0.3367 & 0.3326 & -2.3283 & 2.3125 \end{bmatrix} \tag{B.12}$$

B.4 Height and Rotational Model Gain Matrices

The gain matrices are all size 4×8, but their transpose is used here to better fit the pages.

$$\boldsymbol{K'_{lqrHR}} = \begin{bmatrix} 10.4152 & 10.4135 & -0.0011 & -0.0011 \\ -8.0797 & 8.0197 & -0.0145 & 0.0207 \\ 0.0487 & -0.0242 & 1.4149 & 1.4861 \\ -0.0339 & 0.0337 & 1.5933 & -1.5692 \\ 4.8247 & 4.8265 & -0.0003 & -0.0002 \\ -1.4454 & 1.4396 & 0.0014 & -0.0002 \\ 0.0049 & 0.0029 & 0.5311 & 0.5279 \\ -0.0258 & 0.0257 & 0.3921 & -0.3897 \end{bmatrix} \tag{B.13}$$

B.5 Simulation Trajectory

The trajectory $\mathbb{T} = (x_R(t), y_R(t), z_R(t), \psi_R(t))$ chosen for the simulations of the controllers in chapter 3 is:

- $t < 10s$
  - $x_r(t) = 0.8t$
  - $y_r(t) = 0.4 * (cos(\frac{\pi}{6}t) - 1)$
  - $z_r(t) = 0.6 * t$
- $10s < t < 20s$
  - $x_r(t) = 8 - 0.3(t - 10)$
  - $y_r(t) = 0.4 * (cos(\frac{\pi}{6}t) - 1)$
  - $z_r(t) = 0.6 * t$
- $20s < t < 30s$
  - $x_r(t) = 5 + 0.7(t - 20) - 0.09(t - 20)^2$
  - $y_r(t) = 0.4 * cos(\frac{\pi}{6}t)$
  - $z_r(t) = 12 + sin(0.2\pi(t - 20))$

- $30s < t < 40s$
  - ⋆ $x_r(t) = 11.19 + 0.5sin(0.2\pi(t-30))$
  - ⋆ $y_r(t) = -0.4 + 0.5 * (t-30)$
  - ⋆ $z_r(t) = 12 + sin(0.2\pi(t-20))$
- $40s < t < 50s$
  - ⋆ $x_r(t) = 11.19 + 0.5sin(0.2\pi(t-30))$
  - ⋆ $y_r(t) = 4.6 + 0.7 * (t-40) - 0.035(t-40)^2$
  - ⋆ $z_r(t) = 12 + sin(0.2 * pi(t-20))$
- $t > 50s$
  - ⋆ $x_r(t) = 11.19 + 0.5sin(0.2\pi(t-30))$
  - ⋆ $y_r(t) = 4.6 + 0.7 * (t-40) - 0.035(t-40)^2$
  - ⋆ $z_r(t) = 12 - 0.85 * (t-50) - 0.2 * (t-50)^2$

B.6 Equilibrium Point Equations

This section expands the expression $\boldsymbol{B}(\bar{\boldsymbol{q}})\boldsymbol{u}(t) - \boldsymbol{G}(\bar{\boldsymbol{q}}) = 0$ found in equation (3.4), with $\bar{\boldsymbol{u}}(t) = \begin{bmatrix}\bar{f}_R & \bar{f}_L & \bar{\tau}_{sR} & \bar{\tau}_{sL}\end{bmatrix}$, using equations (2.6), (2.16), (3.97), (2.61) and (2.65). It yields the following eight independent equations:

$$C\bar{\psi}C\bar{\theta}(S(\bar{\alpha}_R)\bar{f}_R + S(\bar{\alpha}_L)\bar{f}_L) + S\bar{\psi}C\bar{\theta}(C(\bar{\alpha}_R)S(\beta)\bar{f}_R - C(\bar{\alpha}_L)S(\beta)\bar{f}_L) - S\bar{\theta}(C(\bar{\alpha}_R)C(\beta)\bar{f}_R + C(\bar{\alpha}_L)C(\beta)\bar{f}_L) = 0 \tag{B.14}$$

$$(C\bar{\psi}S\bar{\theta}S\bar{\phi} - S\bar{\psi}C\bar{\phi})(S(\bar{\alpha}_R)\bar{f}_R + S(\bar{\alpha}_L)\bar{f}_L) + (S\bar{\psi}S\bar{\theta}S\bar{\phi} + C\bar{\psi}C\bar{\phi})(C(\bar{\alpha}_R) S(\beta)\bar{f}_R - C(\bar{\alpha}_L)S(\beta)\bar{f}_L) + C\bar{\theta}S\bar{\phi}(C(\bar{\alpha}_R)C(\beta)\bar{f}_R + C(\bar{\alpha}_L)C(\beta)\bar{f}_L) = 0 \tag{B.15}$$

$$(C\bar{\psi}S\bar{\theta}C\bar{\phi} + S\bar{\psi}S\bar{\phi})(S(\bar{\alpha}_R)\bar{f}_R + S(\bar{\alpha}_L)\bar{f}_L) + (S\bar{\psi}S\bar{\theta}C\bar{\phi} - C\bar{\psi}S\bar{\phi})(C(\bar{\alpha}_R) S(\beta)\bar{f}_R - C(\bar{\alpha}_L)S(\beta)\bar{f}_L) + C\bar{\theta}C\bar{\phi}(C(\bar{\alpha}_R)C(\beta)\bar{f}_R + C(\bar{\alpha}_L)C(\beta)\bar{f}_L) = 0 \tag{B.16}$$

$$
\begin{aligned}
&- C(\bar{\alpha}_R)C(\beta)S(\lambda)l^{'} - \frac{k_\tau}{b}S(\bar{\alpha}_R)\bar{f}_R + C(\bar{\alpha}_L)C(\beta)S(\lambda)l^{'} + \frac{k_\tau}{b}S(\bar{\alpha}_L)\bar{f}_L \\
&+ S(\bar{\phi})T(\bar{\theta})(S(\bar{\alpha}_R)d_z + \frac{k_\tau}{b}S(\beta)C(\bar{\alpha}_R)\bar{f}_R + S(\bar{\alpha}_L)d_z - \frac{k_\tau}{b}S(\beta) \\
&C(\bar{\alpha}_L)\bar{f}_L) + C(\bar{\phi})T(\bar{\theta})(S(\bar{\alpha}_R)l + \frac{k_\tau}{b}C(\beta)C(\bar{\alpha}_R)\bar{f}_R - S(\bar{\alpha}_L)l \\
&- \frac{k_\tau}{b}C(\beta)C(\bar{\alpha}_L)\bar{f}_L) = 0
\end{aligned}
\tag{B.17}
$$

$$
\begin{aligned}
&C(\bar{\phi})(S(\bar{\alpha}_R)d_z + \frac{k_\tau}{b}S(\beta)C(\bar{\alpha}_R)\bar{f}_R + S(\bar{\alpha}_L)d_z - \frac{k_\tau}{b}S(\beta)C(\bar{\alpha}_L)\bar{f}_L) \\
&- S(\bar{\phi})(S(\bar{\alpha}_R)l + \frac{k_\tau}{b}C(\beta)C(\bar{\alpha}_R)\bar{f}_R - S(\bar{\alpha}_L)l - \frac{k_\tau}{b}C(\beta)C(\bar{\alpha}_L)\bar{f}_L) = 0
\end{aligned}
\tag{B.18}
$$

$$
\begin{aligned}
&\frac{S(\bar{\phi})}{C(\bar{\theta})}(S(\bar{\alpha}_R)d_z + \frac{k_\tau}{b}S(\beta)C(\bar{\alpha}_R)\bar{f}_R + S(\bar{\alpha}_L)d_z - \frac{k_\tau}{b}S(\beta)C(\bar{\alpha}_L)\bar{f}_L) \\
&\frac{C(\bar{\phi})}{C(\bar{\theta})}(S(\bar{\alpha}_R)l + \frac{k_\tau}{b}C(\beta)C(\bar{\alpha}_R)\bar{f}_R - S(\bar{\alpha}_L)l - \frac{k_\tau}{b}C(\beta)C(\bar{\alpha}_L)\bar{f}_L) = 0
\end{aligned}
\tag{B.19}
$$

$$
\bar{\tau}_{sR} = 0 \tag{B.20}
$$

$$
\bar{\tau}_{sL} = 0 \tag{B.21}
$$