UNIVERSITAT POLITÈCNICA DE CATALUNYA

Màster:

AUTOMÀTICA I ROBÒTICA

Tesi de Màster

LPV CONTROL OF A QUADROTOR

Abel Torren Larroya

Director: Vicenç Puig i Damiano Rotondo

Curs Acadèmic 2014/15

March 2015

ABSTRACT

This master thesis addresses the LPV control of a quadrotor system. The quadorotor model is first transformed to a LPV model respresentation starting from the nonlinear physical model. Then, using the LPV gain-scheduling control theory, several LPV controller schemes are designed. They are based on static state feedback control with or without reference model. Stability and performance will be established by means of LMIs. Finally, the implemented control solution will be tested in a quadorotor system in a simulation environment using the non-linear model. To assess the performances of those controllers, several scenarios have been simulated.

Contents

1 Introduction **1**

1.1 Motivation 1

1.2 Objectives 2

1.3 Outline of the Thesis 2

2 Background **5**

2.1 Introduction to a Quadrotor 5

2.2 Works on Quadrotor Control 7

2.3 Introduction to LPV 8

2.3.1 LMI Control and Pole Placement 9

3 LPV Model of a Quadrotor **13**

3.1 Deriving the Nonlinear System 13

3.2 Polytopic LPV Model 17

3.2.1 Transforming the Non linear model into a LPV one 18

3.2.2 Apkarian Filter 21

4 LPV Control of a Quadrotor **25**

4.1 Introduction 25

4.1.1 Feedback Control with feedforward 27

4.1.2 Model Reference Control 27

4.1.3 Model Reference Control with Model Matching 29

5 Results Discussion **31**

5.1 Introduction to the simulation scenarios 31
5.2 Scenario 1 31
5.2.1 Feedback Control 32
5.2.2 Reference Model Control 32
5.2.3 Reference Model Control with Model Matching 33
5.3 Scenario 2 34
5.3.1 Feedback Control 34
5.3.2 Reference Model Control 35
5.3.3 Reference Model Control with Model Matching 36
5.4 Scenario 3 37
5.4.1 Feedback Control 37
5.4.2 Reference Model Control 38
5.4.3 Reference Model Control with Model Matching 39
5.5 Comparison 40

6 CONCLUSIONS AND FURTHER RESEARCH 43

6.1 Conclusions 43
6.2 Ongoing and Further Research 44

CHAPTER 1

INTRODUCTION

The aim of this chapter is to describe the main motivations which have inspired this Master Thesis, whose objective is to design and compare several LPV techniques used to control a quadrotor. Finally, the organization of the thesis is presented and the content of the rest of chapters is summarized.

1.1 Motivation

In the recent years, flying robots have attracted a lot of attention and have gained in popularity. Those robots, such as minihelicopters, have a large range of applications, from filming to visual inspection of elements in a building. Therefore, there is a strong research in those kind of aircraft and its control.

An example of these autonomous flying systems is the quadrotor. This rotorcraft is an unmaned aerial system lifted and propelled by four rotors. The rotor mechanics and its configuration make quadrotors an attractive benchmark in control research.

FIGURE 1.1: Quadrotor

A quadrotor is an underactuated system, while it has potentially six degrees of freedom, it

has only four actuators. Quadrotors present a highly non-linear dynamics with highly coupled variables. Those are the main challenges that must be faced to control it. Thus, quadrotors have been used as a test bench for various control techniques.

In this thesis, a LPV control approach is proposed. This kind of technique is well suited to control dynamic systems with parameter variations due to the non-linear dynamics. The use of this approach allows to control in an unified framework systems that change the dynamics with the operating point. Using this technique, the nonlinearities that the system may have are represented in quasi-linear manner by embedding the non-linearities in the parameters.

The controller is designed by solving a LMI optimisation problem. This is a convex optimization method that can be efficiently solved with the current solvers. Using LMIs, its possible to find the closed-loop poles that satisfy a set of constraints related to disturbance rejection, performance and stability conditions.

1.2 Objectives

This master thesis addresses the LPV control of a quadrotor system using three different control laws. The quadorotor model is first transformed into a LPV model representation from the non-linear physical model. Then, using the LPV gain-scheduling control theory, the LPV controllers are designed in order to implement a static state feedback control with or without a reference model. Stability and performance conditions will be established by means of LMIs. Finally, the implemented control solution will be tested in aquadrotor system in a simulation environment using the non-linear model.

1.3 Outline of the Thesis

This dissertation is organized as follows:

Chapter 2: Background

This chapter provides the background on quadrotors. Then, various techniques used to control this system are reviewed. And finally, the LPV techniques are introduced. In this last section, the theory about the LPV models and control design using LMI pole placement is revised.

Chapter 3: LPV Model of a Quadrotor

In this chapter, the nonlinear model has been derived. This has been done using the physical relationships. Then, the nonlinear model has been turned into a LPV model. In order to avoid time varying parameters in the input system matrix, a prefilter has been used. Finally, the LPV parameters have been bounded in order to transform the LPV model to its polytopic form.

Chapter 4: LPV Control of a Quadrotor

In this chapter, the design of the control law has been assessed. Three different approaches have been used. Then, for each of them, a trajectory planner has been designed.

Chapter 5: Results Discussion

This chapter presents the control results obtained with the proposed approaches. Three different control schemes are implemented and compared in simulation in order to see which performs better.

Chapter 6: Conclusions

This chapter summarizes the main contributions of this thesis and presents possible future research extensions.

Chapter 2

Background

This chapter introduces background on the topics used in this thesis. First of all, an introduction to quadrotors is provided. Then, some interesting works using this kind of systems are reviewed and finally the LPV control theory is introduced.

2.1 Introduction to a Quadrotor

A quadrotor, or quadcopter, is a multirotor helicopter that is lifted and propelled by four rotors. The lift force is generated by a set of rotating airfoils.

The quadrotor will be modelled with its four rotors in cross configuration. The structure that connects them will be assumed to be rigid, so the only variable that can vary is the speed of the propellers. In this case, all the possible movements of the quadcopter will be directly related to the rotor velocities (Ω).

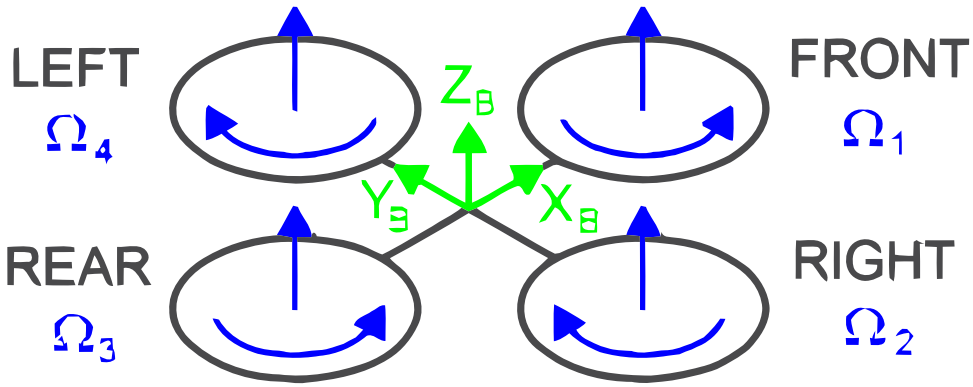

FIGURE 2.1: Simplified Quadrotor

The front and rear rotors rotate counter-clockwise, while the left and right ones turn clockwise. In this way, there is no yaw rotation in hovering and the tail rotor, which is used in standard helicopters, is not required.

Despite the quadrotor potential six degrees of freedom, since it is equipped with four propellers, it can only reach the set-point in four degrees. These are related usually with the basic movements that allow the quadrotor to reach a certain altitude and attitude and correspond the vehicle orientation

- *Throttle*: This movement is provided by varying all the propeller rotations by the same amount. In the case that the quadrotor's pitch and roll orientation are null providing a vertical acceleration in the inertial frame.

- *Roll*: This movement is provided by increasing the rotation of the right propeller while decreasing the rotation of the left one, or the opposite. It produces a torque with respect to the Xb axis (see Figure 2.1).

- *Pitch*: This movement is similar to the roll, but in this case it is produced increasing the velocity of the front propeller and decreasing the velocity of the rear one. In this case, it produces a torque with respect to the Yb axis (see Figure 2.1).

- *Yaw*: Provided by increasing the velocity of the clockwise rotation propellers while decreasing the velocity of the counter-clockwise ones.

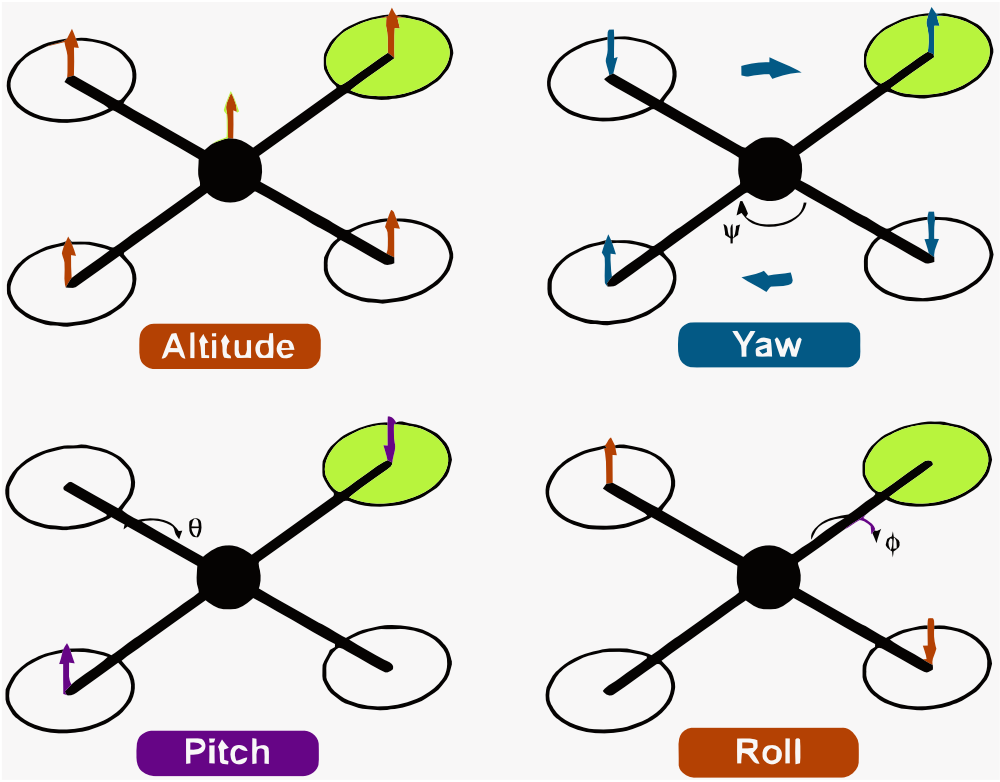

FIGURE 2.2: Basic Attitude and Altitude control

However, there are some techniques that can be used to reach the desired set-point or follow a trajectory in all the coordinates.

2.2 Works on Quadrotor Control

The quadrotor is a classical benchmark for control because of the large number of applications in which it is used and difficulties to control it. Quadrotors have a simple structure with complex dynamics, since it is an under-actuated system. There are a wide variety of algorithms of control proposed in the literature going from PID control to nonlinear techniques.

In the case of the PID control, Tommaso Bresciani et al [4] uses a simplified model to design the control algorithm. In his work, different control structures for the different attitude variables are proposed. As a result, his method proves to be able to stabilize attitude and height. To follow a trajectory, a higher level controller should be implemented. Another PID control work [6] develops a trajectory tracking controller. In order to be able to use planners that do not take into account dynamic feasibility or that provide feedforward inputs, an algorithm that modify the speed profile input paths to guarantee the feasibility of the planned trajectory is proposed. This method has been tested in indoor and outdoor environments with good results.

In the case of nonlinear control, Chowdhary et al [5] compile various nonlinear control techniques that are used for developing flight controllers for UAVs. These techniques are gain scheduling, model predictive control, backstepping, dynamic inversion based control, model reference adaptive control and model based fault tolerant control. In [7], a nonlinear technique based on dynamic feedback is used in order to transform the closed-loop part of the system into a linear, controllable and decoupled subsystem. The system used is divided into two cascade partial parts. Sang-Hyung Lee et al [13] propose to design an attitude controller and a position controller based on feedback linearisation. In this way, the system turns into two fully-controlled subsystems and using the Euler angles it is possible to control the position regardless of the yaw angle. Another nonlinear technique is used by Fouad Yacef et all [15], based on a non-linear state feedback controller which stability and pole placement requirements have been fullfiled by means of the Lyapunov direct method. The modelization has been made using a Takagi-Sugeno model. This control has been applied to the nonlinear system, that has been simulated using the multiple model approach.

In this work, a LPV approach will be used. It is used because allows embedding the system nonlinearities in the varying parameters, such that gain-scheduling control of nonlinear systems can be performed using an extension of linear techniques. A similar approach was proposed in order to control a Twin-Rotor MIMO System (TRMS) [9]. In this case, the model was approximated in a polytopic way and identified using non-linear least squares identification. Then, an LPV state observer and state-space controller were designed using a pole placement method based in LMI regions. Samarathunga and Whidborne's work [12] describes a LPV controller design for a quadrotor vehicle. The controller synthesis requires the LPV plant model which is obtained by linearisation of the modelling equations. This representation is transformed into to a convex polytopic form. The H_∞ self gain scheduling control method has been applied to obtain a LPV controller which is tested using a simplified nonlinear model of the quadrotor. The LPV controller performance was compared with a H_∞ controller and exhibited significant close tracking capabilities with respect to the H_∞ controller.

Rotondo et al [8] proposes a solution that relies on model based control, where a reference model generates the desired trajectory. This approach is used for the design of the Fault Tolerant

Control (FTC) of a quadrotor using the LPV Polytopic framework. The controller is designed solving the LMIs in order to achieve regional pole placement and H_∞. A further paper [10, 11], added a reference model that describes the desired trajectory. Abdullah and Zibri [1] proposed to use this kind of reference as a the principle to the design of a LPV model reference whose space-state matrices depend on a set of parameter varying parameters that are bounded and available online.

2.3 Introduction to LPV

The LPV paradigm was introduced in the PhD Thesis of Shamma [14] for the analysis of the control design practice of "gain-scheduling". The nonlinear system is described as a parameterized linear system, where these parameters depend on some measurable scheduling variables. These systems can be formulated as

$$\dot{x} = A(\rho(t))x + B(\rho(t))u,\ \rho(t) \in S \tag{2.1}$$

where $\rho(t)$ is the parameter varying vector in the region S.

The parameters $\rho(t)$ vary in function of scheduling variables $p(t)$. These scheduling variables should be exogenous to the system. When the scheduling variables are inner variables as the states, then the model (2.1) is known as quasi-LPV. The function that relates the variation of the parameters with the scheduling variables is known as the scheduling function $\rho(t) = \sigma(p(t))$. The parameter scheduling functions should be defined to have a LPV description that is as close as possible to the nonlinear for all the values in the region S.

Finding a LPV description from a nonlinear system is a non-trivial task. One method is to hide the nonlinearities in parameters. So, depending how these parameters are defined, a system can have different LPV descriptions. One of the most used approaches, known as the non-linear embedding approach aims to hide the non-linearities in the parameters such the LPV description and the nonlinear system are equivalent

$$A(\rho(t))x + B(\rho(t))u = f(x,u),\ \rho(t) \in S \tag{2.2}$$

This property ensures that the trajectories of the original nonlinear system are also trajectories of the LPV system.

A LPV system might be seen as an extension of a linear time invariant (LTI) system as they coincide when the parameters are frozen in a given time instant at a given operating point.

The variation of parameter $\rho(t)$ vector can be bounded using a bounding box that bounds the parameter scheduling function in the region of parameter variation

$$\rho(t) \in \Gamma = \{\rho(t) | \bar{\rho}_i(t) \leq \rho_i(t) \leq \bar{\rho}_i(t), i = 1, ..., N\},\ \forall t \geq t_0 \tag{2.3}$$

where N is the number of varying parameters.

The time derivative of the parameter could be assumed to be bounded also in the same manner. By means of the bounding box approximation, the LPV model can be transformed into a interpolation of the vertices of Γ called polytopic representation of the LPV system. In this project, the LPV model of the system will be approximated is such manner

$$A(\rho(t)) = \sum_{i=1}^{N_\rho} \pi_i(p(t)) A_i \tag{2.4}$$

where π_i are polytopic coordinates of each submodel A_i in function of the scheduling variables $p(t)$ and is determined by the weighted distance to the vertices of Γ. N_ρ is the number of submodels that is equal to the number of vertices of Γ, that is, $N_\rho = 2^N$. In a polytropic LPV system, A_i is given by the evaluation of the space-state matrix in the vertices and π_i satisfies

$$\sum_{i=1}^{N_\rho} \pi_i(p(t)) = 1,\ \pi_i(p(t)) > 0 \tag{2.5}$$

The polytopic coordinates π_i are computed using the vertex interpolation using the method proposed by Apkarian [2]. Every A_i corresponds to frozen ρ_i for some given value of the scheduling variables $p(t)$ corresponding to the i-th operating point. The system matrices $A(\rho(t))$ in that operating point are constant. So, a polytopic LPV system is designed by interpolating N_ρ operating functions defined as stated before in (2.4). In the particular case in which $N_\rho = 1$, the polytopic LPV model turns into a LTI one.

2.3.1 LMI Control and Pole Placement

Let us consider that the system will be controlled using a state feedback controller with tracking reference input. Then, the control law can be expressed as

$$u(t) = u_r(t) + K_d(\rho(t))(x(t) - x_r(t)) \tag{2.6}$$

where the state $x(t) \in \mathbf{R}^{n_x}$ and the feedforward control action $u_r(t) \in \mathbf{R}^{n_u}$ corresponds to an equilibrium point in the reference, $x_r(t)$. The control matrix $K_d(\rho(t)) \in \mathbf{R}^{n_x \times n_u}$ is the gain of the LPV controller. This matrix is designed in order to satisfy the specifications of placing the closed-loop poles in a region of the complex plane using a LMI approach.

By constraining the eigenvalues inside a predetermined region, stability is guaranteed and a satisfactory transient response is ensured. A subset $\mathcal{D}$ of a complex plane is an LMI region if there exist a symmetric matrix α and a matrix β such that

$$\mathcal{D} = \{\mathcal{Z} \in \mathbb{C} : f_{\mathcal{D}(\mathcal{Z})} < 0\} \tag{2.7}$$

where $f_{\mathcal{D}(\mathcal{Z})}$ is the characteristic function of $\mathcal{D}(\mathcal{Z})$ and is defined as

$$f_{\mathcal{D}(\mathcal{Z})} = \alpha + \mathcal{Z}\beta + \bar{\mathcal{Z}}\beta^T \tag{2.8}$$

where α and β are real matrices such that $\alpha = \alpha^T$. This characteristic function can be modified in order to define several regions, for instance

- $f_{\mathcal{D}(\mathcal{Z})} < 2a$ ensures that $Re(Z) < a$
- a disk centered at $(-q, 0)$ with a radius R is defined as

$$f_{\mathcal{D}(\mathcal{Z})} = \begin{bmatrix} -R & q + \mathcal{Z} \\ q + \bar{\mathcal{Z}} & -R \end{bmatrix} \tag{2.9}$$

- a conic section with its apex in the origin and a inner angle 2Θ is defined as

$$f_{\mathcal{D}(\mathcal{Z})} = \begin{bmatrix} sin\Theta(\mathcal{Z} + \bar{\mathcal{Z}}) & cos\Theta(\mathcal{Z} - \bar{\mathcal{Z}}) \\ cos\Theta(\bar{\mathcal{Z}} - \mathcal{Z}) & sin\Theta(\mathcal{Z} + \bar{\mathcal{Z}}) \end{bmatrix} \tag{2.10}$$

It is possible to use several LMI regions such that the results LMI region will be the intersection. A real matrix Y, will be $\mathcal{D}(\mathcal{Z})$-stable, all its spectrum belongs to the region $\mathcal{D}(\mathcal{Z})$, if exists a matrix X, that is symmetric and positive definite. In this region, the pole placement is characterized as

$$\mathcal{Z}_{\mathcal{D}}(X, Y) := \alpha \otimes X + \beta \otimes (YX) + \beta^T \otimes (YX)^T < 0 \tag{2.11}$$

where $\otimes$ is the Kronecker product of two matrices.

Let us a LPV system described by

$$\dot{x}(t) = A(\rho(t))x(t) + Bu(t) \tag{2.12a}$$

$$y(t) = Cx(t) \tag{2.12b}$$

where $C = I$, i.e., all the states are measured. If it is not the case, then an observer should be used.

In this system, the feedback control law $u(t) = K_d(\rho(t))(x(t) - x_r(t))$ will be used as an input. In order to design this controller, a state-space gain K_d must scheduled by $\rho(t)$ that places the closed-loop poles of the system (2.12) in some LMI region (2.7) with characteristic function $f_{\mathcal{D}}$ (2.8).

This pole placement constraint is satisfied if exists a symmetrical positive defined matrix X such that

$$\{\alpha X + \beta[A(\rho(t)) + BK_d(\rho(t))]X + \beta X[A(\rho(t)) + BK_d(\rho(t))]^T\} < 0 \tag{2.13}$$

This relationship must be satisfied for all the values of $\rho(t)$. Using the auxiliary variable

$\gamma = K_d(\rho(t))X$, the matrix inequality (2.13) becomes an LMI, that will be solved using convex optimisation techniques

$$\{\alpha X + \beta[A(\rho(t))X + B\gamma] + \beta X[A(\rho(t))X + B\gamma]^T\} < 0 \tag{2.14}$$

In order to reduce the LMI constraints, the system is modelled as a LPV. In this paper, the model will be expressed in a polytopic way as in (2.4) such that the control law (2.6) can be expressed in a polytopic way as well

$$u(t) = u_r(t) + \sum_{i=1}^{N_{\rho(t)}} \pi_i(p(t))K_i(x(t) - x_r(t)) \tag{2.15}$$

where π_i provides the polytopic coordinates in function of the scheduling variables $p(t)$.

Then, this design problem, consists in computing a state-feedback K_d matrix and a single Lyapunov matrix X such that $\mathcal{Z}_{\mathcal{D}}(A_i + BK_{d,i}, X) < 0$.

Chapter 3

LPV Model of a Quadrotor

This chapter introduces the nonlinear equations of the system as shown in the work of Bresciani [3]. Then, a method to transform them into a polytopic LPV model is introduced. Finally, the quadrotor LPV model is obtained.

3.1 Deriving the Nonlinear System

In order to obtain the nonlinear model of this system, two different frames are used as shown in the figure 3.1

- The earth inertial frame
- The body-fixed frame

The kinematics of a generic six degree of freedom body can be defined as

$$\dot{\xi} = J_\Theta \delta \tag{3.1}$$

where $\dot{\xi}$ is the generalized velocity vector with respect to the earth inertial frame, δ is the one with respect to the body-fixed frame and J_Θ is the generalized matrix. ξ is composed of the quadrotor linear and angular position with respect to the earth inertial frame

$$\xi = [X\ Y\ Z\ \phi\ \theta\ \psi]^T \tag{3.2}$$

Similarly, δ is composed of the quadrotor linear and angular velocity with respect the body-fixed frame

$$\delta = [u\ v\ w\ p\ q\ r]^T \tag{3.3}$$

In addition, J_Θ is composed of 4 sub-matrices

$$J_\Theta = \begin{bmatrix} R_\Theta & 0_{3\times3} \\ 0_{3\times3} & T_\Theta \end{bmatrix} \tag{3.4}$$

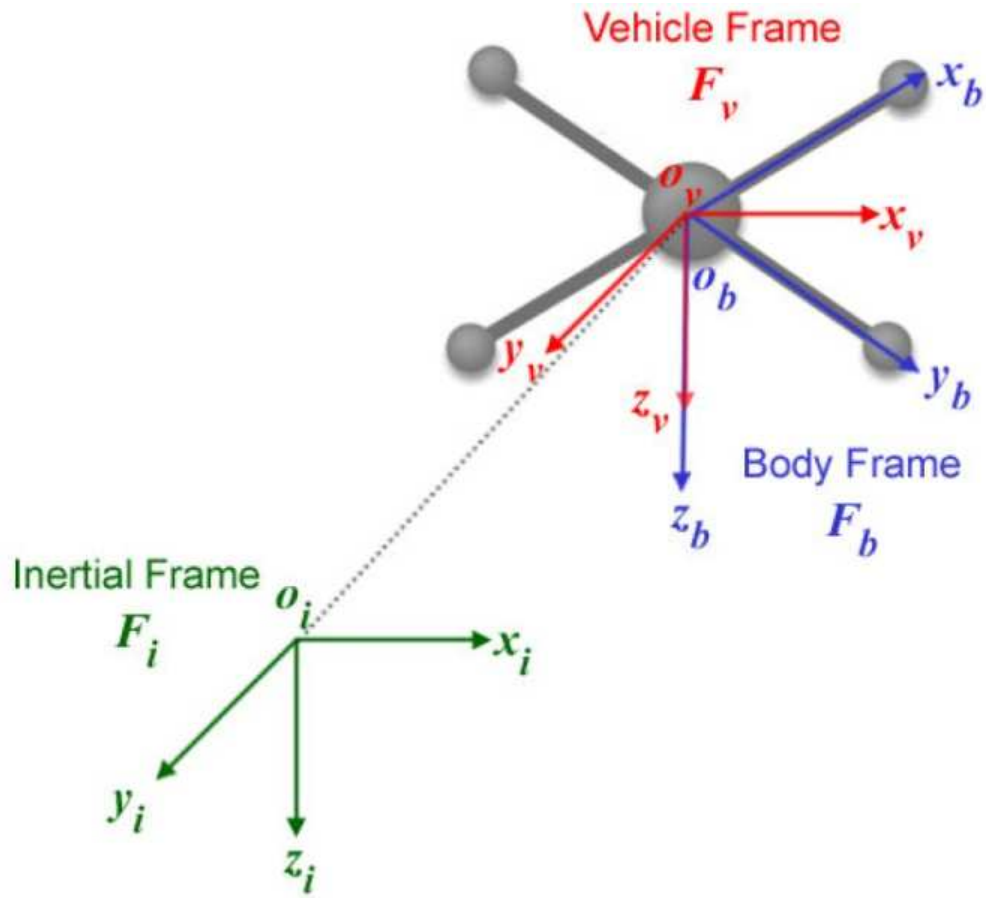

FIGURE 3.1: Basic Attitude and Altitude control

where $0_{3\times3}$ is a 3 by 3 matrix filled with zeros, R_Θ is is the 3 by 3 rotation matrix and T_Θ is the 3 by 3 transfer one.

The rotation and transfer matrices are defined as

$$R_\Theta = \begin{bmatrix} c_\psi c_\theta & -s_\psi c_\theta + c_\psi s_\theta s_\phi & s_\psi s_\theta + c_\psi s_\theta c_\phi \\ c_\psi c_\theta & c_\psi c_\theta + s_\psi s_\theta s_\phi & -c_\psi s_\theta + s_\psi c_\theta c_\phi \\ -s_\theta & c_\theta s_\phi & c_\theta c_\phi \end{bmatrix} \tag{3.5}$$

$$T_\Theta = \begin{bmatrix} 1 & -t_\theta s_\phi & t_\theta s\phi \\ 0 & c_\psi & -s_\psi \\ 0 & s_\psi / c_\theta & c_\psi / c_\theta \end{bmatrix} \tag{3.6}$$

where c_k, s_k and t_k represent the cosinus, sinus and tangent, respectively. The dynamics of the generic degree of freedom rigid-body takes into account the mass of the body,m $[kg]$, and its inertia matrix, I $[N\,m\,s^2]$, and is described by

$$\begin{bmatrix} mI_{3x3} & 0_{3x3} \\ 0_{3x3} & I \end{bmatrix} \begin{bmatrix} \dot{V}^B \\ \dot{\omega}^B \end{bmatrix} + \begin{bmatrix} \omega^b \times mV^B \\ \omega^B \times I\omega^B \end{bmatrix} = \begin{bmatrix} F^B \\ \tau^B \end{bmatrix} \tag{3.7}$$

where I_{3x3} is a 3 by 3 identity matrix, $\dot{V}^B$ is the linear acceleration vector [$m\ s^{-2}$] with respect to the body-fixed frame, $\dot{\omega}^B$ is the angular acceleration vector [rad s^{-2}] with respect to the same frame, F^B is the quadrotor forces vector [N] and τ^B is the quadrotor torques vector [$N\ m$] with respect the body-fixed frame.
Then, it is assumed that the origin of the body-fixed frame is coincident with the center of mass and that the inertial matrix is diagonal.
A generalized force vector can be defined as

$$\Delta = [F_x\ F_y\ F_z\ \tau_x\ \tau_y\ \tau_z]^T \tag{3.8}$$

Using this equation, the dynamics can be rewritten as

$$M_B\dot{v} + C_B(v)v = \Delta \tag{3.9}$$

where $\dot{v}$ is the generalized acceleration vector with respect to the body-fixed frame, M_B is the inertia matrix and $C_B(v)$ is the Coriolis-centripeta matrix, both with respect the body-fixed frame.

$$M_B = \begin{bmatrix} mI_{3x3} & 0_{3x3} \\ 0_{3x3} & I \end{bmatrix} \tag{3.10}$$

Thanks to the assumptions made before, this is diagonal and constant.

$$C_B(v) = \begin{bmatrix} 0_{3x3} & -mS(V^B) \\ 0_{3x3} & -mS(\omega^B) \end{bmatrix} \tag{3.11}$$

where $S(K)$ is defined as a skew-symmetric operator, that given a three dimension vector is defined as follows.

$$S(K) = \begin{bmatrix} 0 & -K_3 & K_2 \\ k_3 & 0 & -K_1 \\ -K_2 & K_3 & 0 \end{bmatrix},\ K = \begin{bmatrix} K_1 & K_2 & K_3 \end{bmatrix}^T \tag{3.12}$$

In the case of a quadrotor, the generalized force vector Δ in (3.8) can be divided in three different vectors according to which of the quadcopter contributions describes.
The first of all is the gravity vector, $G_B(\xi)$ given by the acceleration due to the gravity $g\,[ms^{-2}]$.

$$G_B(\xi) = mg[s_\theta\ -c_\theta s_\phi\ -c_\theta c_\phi\ \tau_x\ \tau_y\ \tau_z]^T \tag{3.13}$$

The second contribution corresponds to gyroscopic effects due to the propeller rotation, since two rotates clockwise and the other two counterclockwise. Thus, there is an imbalance when the sum of the rotation is not zero. There is also a contribution of the pitch and the roll

$$O_B(v)\begin{bmatrix}\Omega_1\\ \Omega_2\\ \Omega_3\\ \Omega_4\end{bmatrix} = J_{TP}\begin{bmatrix}0_{3x1}\\ -q\\ p\\ 0\end{bmatrix}\Omega \tag{3.14}$$

where Ω is defined as the overall speed of the propeller $[rad\,s^{-1}]$, O_B is the gyroscopic propeller matrix, J_{TP} is the total rotational moment of inertia around the propeller axis calculated in the next section $[N\,m\,s^2]$. Additionally Ω_j is the speed of the propeller j as defined in the Figure 1.1.

$$\Omega = \Omega_2 + \Omega_4 - \Omega_1 - \Omega_3 \tag{3.15}$$

The third contribution takes into account the forces and torques produced by the main movement inputs. From aerodynamics consideration, it follows that both torques and forces are proportional to the square of the speed of the propellers. Therefore, the movement vector is defined as follows

$$U_B = [0\ 0\ U_1\ U_2\ U_3\ U_4]^T \tag{3.16}$$

where U_1, U_2, U_3 and U_4 are the throttle, roll, pitch and yaw, respectively

$$U_1 = b(\Omega_1^2 + \Omega_2^2 + \Omega_3^2 + \Omega_4^2) \tag{3.17a}$$
$$U_2 = bl(\Omega_4^2 - \Omega_2^2) \tag{3.17b}$$
$$U_3 = bl(\Omega_3^2 - \Omega_1^2) \tag{3.17c}$$
$$U_4 = d(\Omega_2^2 + \Omega_4^2 - \Omega_1^2 - \Omega_3^2) \tag{3.17d}$$

In these relationships b is defined as the thrust factor $[N\,s^2]$, d is the drag factor $[N\,m\,s^2]$ and l is the distance between the center of the quadcopter and the center of the propeller $[m]$.

These relationships are written in respect to the body-fixed frame. This reference is widely used in six degree of freedom rigid bogy equations. However, in this case the equations will be expressed in terms of a hybrid frame that mixes the frames introduced previously. That reference will be used because it allows to express easily the dynamics combined with the control. In this frame a generalized velocity vector, ζ, would be defined following the next equation

$$\zeta = [\dot{X}\ \dot{Y}\ \dot{Z}\ p\ q\ r]^T \tag{3.18}$$

The dynamics in the hybrid frame can be rewritten as follows

$$M_B\dot{\zeta} + C_H(\zeta)\zeta = G_H + O_H(\zeta)\begin{bmatrix}\Omega_1\\ \Omega_2\\ \Omega_3\\ \Omega_4\end{bmatrix} + U_H \tag{3.19}$$

The system inertia with respect the hybrid frame is equal to the one with respect the fixed-body frame, so $O_B = O_H$. The Coriolis-centripetal matrix is defined as

$$C_H(\zeta) = \begin{bmatrix}0_{3x3} & 0\\ 0_{3x3} & -S(I\omega^B)\end{bmatrix} \tag{3.20}$$

The gravitational contribution with respect the hybrid frame is

$$G_H = -[0\ 0\ mg\ 0\ 0\ 0]^T \tag{3.21}$$

The gyroscopic effects by the propeller rotation are unvaried because it affects only the angular equations referred to the body-fixed frame. Finally, the movement vector is different because the input U_1 affects all the linear equations through the rotation matrix

$$U_H(\Omega) = \begin{bmatrix}R_\Theta & 0_{3x3}\\ 0_{3x3} & T_\Theta\end{bmatrix} U_B(\Omega) \tag{3.22}$$

Isolating the derivate of the generalized velocity in the dynamics equation with respect to the hybrid frame, the system leads to the following equations

$$\ddot{x} = (cos\phi\, sen\theta\, cos\psi + sen\phi\, sen\theta)\frac{U_1}{m} \tag{3.23a}$$

$$\ddot{y} = (cos\phi\, sen\theta\, sen\psi + sen\phi\, cos\theta)\frac{U_1}{m} \tag{3.23b}$$

$$\ddot{z} = -g + (cos\theta\, cos\phi)\frac{U_1}{m} \tag{3.23c}$$

$$\ddot{\phi} = \dot{\theta}\dot{\psi}\frac{I_{yy} - I_{zz}}{I_{xx}} - \frac{J_{TP}}{I_{xx}}\dot{\phi}\Omega + \frac{U_2}{I_{xx}} \tag{3.23d}$$

$$\ddot{\theta} = \dot{\phi}\dot{\psi}\frac{I_{zz} - I_{xx}}{I_{yy}} - \frac{J_{TP}}{I_{yy}}\dot{\theta}\Omega + \frac{U_3}{I_{yy}} \tag{3.23e}$$

$$\ddot{\psi} = \dot{\phi}\dot{\theta}\frac{I_{xx} - I_{yy}}{T_{zz}} + \frac{U_4}{I_{zz}} \tag{3.23f}$$

3.2 Polytopic LPV Model

In this work, it will only be taken into account the attitude and altitude relationships (3.23d)-(3.23f). It will be assumed that all the states are measured.

3.2.1 Transforming the Non linear model into a LPV one

Considering the nonlinear model as

$$\dot{x} = f(x(t), u(t)) \tag{3.24a}$$

$$y = g(x(t), u(t)) \tag{3.24b}$$

in order to automatically transform it into a LPV one, each row of (3.24) is expanded into its summands.

$$\dot{x}_i = \sum_{j=1}^{r_x} f_{ij}(x, u),\ i = 1, .., n_x \tag{3.25a}$$

$$y_i = \sum_{j=1}^{r_y} g_{ij}(x, u),\ i = 1, .., n_y \tag{3.25b}$$

where r_{x_i} and r_{y_i} are the number of summands in each row of (3.24).

Then, those terms are decomposed into its denominator and nominator and a constant factor

$$\dot{x}_i = \sum_{j=1}^{r_{x_i}} k_{ij} \frac{n_{ij}(x, u)}{d_{ij}(x, u)},\ i = 1, .., n_x \tag{3.26}$$

This decomposition is applied also to the input function in a similar way. This number is factored in order to determine the possibility of hiding nonlinearities in the parameters. This factor classifies a term into two possible kinds:

- *Constant or non-factorizable numerator*, K_0. A factor of the state or input cannot be chosen.

- *Arbitrary positive power of factor*, K_p. A factor of the state or the control can be chosen.

Parameters components, according to the classification, can be chosen as

$$\vartheta_{ij}^a = k_{ij} \frac{n_{ij}(x, u)}{d_{ij}(x, u) x_l},\ l = 1, .., n_x \tag{3.27a}$$

$$\vartheta_{ij}^b = k_{ij} \frac{n_{ij}(x, u)}{d_{ij}(x, u) u_l},\ l = 1, .., n_x \tag{3.27b}$$

Then, the parameters are defined depending of the type. If the numerator is K_0, the parameter can be taken using n_x possible assigments of the system matrix A and n_u possible assigments of the system matrix B. In the case of a K_p numerator, the parameter is a factor of the numerator.

Finally, the elements of the matrices A and B and the corresponding parameters $\rho(t)$ need to be derived from the parameter components (3.27). The final state space matrix is given by superposition.

Taking into account the nonlinear relationships of the rotational and altitude coordinates (3.23d)-(3.23f), the LPV parameters can be defined as

$$\rho_1 = \Omega_1,\ \rho_2 = \Omega_2,\ \rho_3 = \Omega_3,\ \rho_4 = \Omega_4 \tag{3.28a}$$

$$\rho_5 = cos(\phi)\, cos(\theta),\ \rho_6 = \dot{\phi},\ \rho_7 = \dot{\theta},\ \rho_8 = \dot{\psi} \tag{3.28b}$$

and considering

$$x_1 = \dot{z},\ x_2 = z \tag{3.29a}$$

$$x_3 = \dot{\phi},\ x_4 = \phi \tag{3.29b}$$

$$x_5 = \dot{\theta},\ x_6 = \theta \tag{3.29c}$$

$$x_7 = \dot{\psi},\ x_8 = \psi \tag{3.29d}$$

$$u_1 = \Omega_1,\ u_2 = \Omega_2,\ u_3 = \Omega_3,\ u_4 = \Omega_4 \tag{3.29e}$$

Using this method, the nonlinear system turns into,

$$\begin{bmatrix} \dot{x_1} \\ \dot{x_2} \\ \dot{x_3} \\ \dot{x_4} \\ \dot{x_5} \\ \dot{x_6} \\ \dot{x_7} \\ \dot{x_8} \end{bmatrix} = A(\rho(t)) \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \\ x_7 \\ x_8 \end{bmatrix} + B(\rho(t)) \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{bmatrix} \tag{3.30}$$

where

$$A(\rho(t)) = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & a_{23} & 0 & a_{25} & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & a_{46} & 0 & a_{48} \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & a_{64} & 0 & 0 & 0 & a_{68} \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & a_{84} & 0 & a_{86} & 0 & 0 \end{bmatrix} \tag{3.31}$$

$$a_{23} = \frac{-g}{2max\{|x_3|, \epsilon\}} \tag{3.32}$$

$$a_{25} = \frac{-g}{2max\{|x_5|, \epsilon\}} \tag{3.33}$$

$$a_{46} = x_8 \frac{I_{yy} - I_{zz}}{2I_{xx}} \tag{3.34}$$

$$a_{48} = x_6 \frac{I_{yy} - I_{zz}}{2I_{xx}} \tag{3.35}$$

$$a_{64} = x_8 \frac{I_{zz} - I_{xx}}{2I_{yy}} \tag{3.36}$$

$$a_{68} = x_4 \frac{I_{zz} - I_{xx}}{2I_{yy}} \tag{3.37}$$

$$a_{84} = x_6 \frac{I_{xx} - I_{yy}}{2I_{zz}} \tag{3.38}$$

$$a_{86} = x_4 \frac{I_{xx} - I_{yy}}{2I_{zz}} \tag{3.39}$$

$$B(\rho(t)) = \begin{bmatrix} 0 & 0 & 0 & 0 \\ b_1 & b_2 & b_3 & b_4 \\ 0 & 0 & 0 & 0 \\ b_5 & b_6 & b_7 & b_8 \\ 0 & 0 & 0 & 0 \\ b_9 & b_10 & b_11 & b_12 \\ 0 & 0 & 0 & 0 \\ b_13 & b_14 & b_15 & b_16 \end{bmatrix} \tag{3.40}$$

in which

$$b_1 = cosx_3 cosx_5 \frac{b}{m} u_1 \tag{3.41a}$$

$$b_2 = cosx_3 osx_5 \frac{b}{m} u_2 \tag{3.41b}$$

$$b_3 = cosx_3 cosx_5 \frac{b}{m} u_3 \tag{3.41c}$$

$$b_4 = cosx_3 cosx_5 \frac{b}{m} u_4 \tag{3.41d}$$

$$b_5 = b_7 = \frac{J_{TP}}{I_{xx}} x_6 \tag{3.41e}$$

$$b_6 = \frac{J_{TP}}{I_{xx}} x_6 - \frac{lb}{I_{xx}} u_2 \tag{3.41f}$$

$$b_8 = \frac{J_{TP}}{I_{xx}} x_6 + \frac{lb}{I_{xx}} u_4 \tag{3.41g}$$

$$b_9 = -\frac{J_{TP}}{I_{yy}} x_4 - \frac{lb}{I_{xx}} u_1 \tag{3.41h}$$

$$b_{10} = b_12 = \frac{J_{TP}}{I_{yy}} x_4 \tag{3.41i}$$

$$b_{11} = -\frac{J_{TP}}{I_{yy}} x_4 + \frac{lb}{I_{xx}} u_3 \tag{3.41j}$$

$$b_{13} = \frac{d}{I_{zz}} u_1 \tag{3.41k}$$

$$b_{14} = \frac{d}{I_{zz}} u_2 \tag{3.41l}$$

$$b_{15} = \frac{d}{I_{zz}} u_3 \tag{3.41m}$$

$$b_{16} = \frac{d}{I_{zz}} u_4 \tag{3.41n}$$

3.2.2 Apkarian Filter

In order to apply the LPV control design methodology described in Chapter 3, the LPV model should be transformed in order to avoid the parameter dependence on B. To this aim, the prefiltering prefiltering proposed in [2] is used. To do that, a new input variable w is defined.

$$\dot{x_u}(t) = A_u x_u(t) + B w(t) \tag{3.42}$$

$$u(t) = C_u x_u(t) \tag{3.43}$$

where A_u is stable and the LPV system will be described by

$$\begin{pmatrix} \dot{x} \\ \dot{x_u} \end{pmatrix} = \begin{pmatrix} A(\rho(t)) & B(\rho(t))C_u \\ 0 & A_u \end{pmatrix} \begin{pmatrix} x \\ x_u \end{pmatrix} + \begin{pmatrix} 0 \\ B_u \end{pmatrix} w(t) \tag{3.44}$$

LPV Model with the Apkarian Filter applied

In the quadrotor case, the matrix A_u will be considered as a diagonal with fast poles compared to the quadrotor ones. Thus,

$$A_u = \begin{pmatrix} -100 & 0 & 0 & 0 \\ 0 & -100 & 0 & 0 \\ 0 & 0 & -100 & 0 \\ 0 & 0 & 0 & -100 \end{pmatrix} \tag{3.45}$$

$$u = x_u \tag{3.46}$$

Then, using (3.44) the system will be expressed as follows

$$\begin{bmatrix} \dot{x_1} \\ \dot{x_2} \\ \dot{x_3} \\ \dot{x_4} \\ \dot{x_5} \\ \dot{x_6} \\ \dot{x_7} \\ \dot{x_8} \\ \dot{x_9} \\ \dot{x_{10}} \\ \dot{x_{11}} \\ \dot{x_{12}} \end{bmatrix} = A(\rho(t)) \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \\ x_7 \\ x_8 \\ x_9 \\ x_{10} \\ x_{11} \\ x_{12} \end{bmatrix} + B(\rho(t)) \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{bmatrix} \tag{3.47}$$

$$A(\rho(t)) = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & a_{23} & 0 & a_{25} & 0 & 0 & 0 & a_{29} & a_{210} & a_{211} & a_{212} \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & a_{46} & 0 & a_{48} & a_{49} & a_{410} & a_{411} & a_{412} \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & a_{64} & 0 & 0 & 0 & a_{68} & a_{69} & a_{610} & a_{611} & a_{612} \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & a_{85} & 0 & a_{86} & 0 & 0 & a_{89} & a_{810} & a_{811} & a_{812} \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -100 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -100 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -100 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -100 \end{bmatrix} \tag{3.48}$$

where,

$$a_{23} = \frac{-g}{2max\{|x_3|, \epsilon\}} \tag{3.49a}$$

$$a_{25} = \frac{-g}{2max\{|x_5|, \epsilon\}} \tag{3.49b}$$

$$a_{29} = cosx_3 cosx_5 \frac{b}{m} u_1 \tag{3.49c}$$

$$a_{210} = cosx_3 cosx_5 \frac{b}{m} u_2 \tag{3.49d}$$

$$a_{211} = cosx_3 cosx_5 \frac{b}{m} u_3 \tag{3.49e}$$

$$a_{212} = cosx_3 cosx_5 \frac{b}{m} u_4 \tag{3.49f}$$

$$a_{46} = x_8 \frac{I_{yy} - I_{zz}}{2I_{xx}} \tag{3.49g}$$

$$a_{48} = x_6 \frac{I_{yy} - I_{zz}}{2I_{xx}} \tag{3.49h}$$

$$a_{49} = a_{411} = \frac{J_{TP}}{I_{xx}} x_6 \tag{3.49i}$$

$$a_{410} = \frac{J_{TP}}{I_{xx}} x_6 - \frac{lb}{I_{xx}} u_2 \tag{3.49j}$$

$$a_{412} = \frac{J_{TP}}{I_{xx}} x_6 + \frac{lb}{I_{xx}} u_4 \tag{3.49k}$$

$$a_{64} = x_8 \frac{I_{zz} - I_{xx}}{2I_{yy}} \tag{3.49l}$$

$$a_{68} = x_4 \frac{I_{zz} - I_{xx}}{2I_{yy}} \tag{3.49m}$$

$$a_{69} = -\frac{J_{TP}}{I_{yy}} x_4 - \frac{lb}{I_{xx}} u_1 \tag{3.49n}$$

$$a_{610} = a_{612} = \frac{J_{TP}}{I_{yy}} x_4 \tag{3.49o}$$

$$a_{611} = -\frac{J_{TP}}{I_{yy}} x_4 + \frac{lb}{I_{xx}} u_3 \tag{3.49p}$$

$$a_{84} = x_6 \frac{I_{xx} - I_{yy}}{2I_{zz}} \tag{3.49q}$$

$$a_{86} = x_4 \frac{I_{xx} - I_{yy}}{2I_{zz}} \tag{3.49r}$$

$$a_{89} = \frac{d}{I_{zz}} u_1 \tag{3.49s}$$

$$a_{810} = \frac{d}{I_{zz}} u_2 \tag{3.49t}$$

$$a_{811} = \frac{d}{I_{zz}} u_3 \tag{3.49u}$$

$$a_{812} = \frac{d}{I_{zz}} u_4 \tag{3.49v}$$

$$B(\rho(t)) = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.50}$$

The LPV parameters are bounded since the states and controls are bounded

$$x_i \in [-0.5, 0.5],\ i = 4, 6 \tag{3.51a}$$

$$x_j \in [-\pi/3, \pi/3],\ j = 3, 5 \tag{3.51b}$$

$$u_k \in [100, 500],\ k = 1, 2, 3, 4 \tag{3.51c}$$

The LPV parameters bounds can be obtained in a straightforward manner using state and control bounds. From these bounds, the LPV polytopic model can be obtained using the bounding box approach proposed by Apkarian [2].

CHAPTER 4

LPV CONTROL OF A QUADROTOR

This chapter introduces the LPV control method selected and then analyzes how to input the trajectory to the controller. In order to control the quadrotor using the LPV model presented in previous chapter and the LPV approach introduced in Chapter 2, the method proposed in the article of Ali Abdullah and Mohamed Zribi will be used [1]. In this chapter, an introduction to this method, various control law design proposals will be presented an applied to the quadrotor.

4.1 Introduction

Considering a LPV model as defined in the previous chapter

$$\dot{x}(t) = A(\rho(t))x(t) + Bu(t) \tag{4.1}$$

$$y(t) = Cx(t) \tag{4.2}$$

expressing the relationships in polytopic form as explained in Section 2.3 and assuming that all the states are measured, $C = I$.

A reference model controller whose output $y(t)$ converges to a desired reference output $\bar{y}(t)$ asymptotically will be designed following the approach proposed in Ali Abdullah and Mohamed Zribi [1]. This reference is obtained following a LPV reference model such as

$$\dot{\hat{x}}(t) = \hat{A}(\rho(t))\hat{x}(t) + \hat{B}\hat{u}(t) \tag{4.3}$$

$$\hat{y}(t) = \hat{C}\hat{x}(t) \tag{4.4}$$

$$\hat{A}(\rho(t)) = \sum_{i=1}^{N_\rho} \pi(t)_i(p(t))\hat{A}_i \tag{4.5}$$

Considering a control law given by,

$$u = K(\rho(t))[x - G\bar{x}] + M(\rho(t))\bar{x} + Q\bar{u} \tag{4.6}$$

where the matrices G, Q, $M(\rho(t))$ and Q are matrices that satisfy the following relations

$$\bar{C} = CG \tag{4.7}$$

$$G\bar{B} = BQ \tag{4.8}$$

$$G\bar{A}(\rho(t)) = BM(\rho(t)) + A(\rho(t))G \tag{4.9}$$

$$K(\rho(t)) = L(\rho(t))L^{-1} \tag{4.10}$$

The matrix $L(\rho(t))$ and the positive definite symmetric matrix P are the solucions of the following inequality

$$H(\rho(t)) := A(\rho(t))P + BL(\rho(t)) + (A(\rho(t))P + BL(\rho(t)))^T < 0 \tag{4.11}$$

The control law applied to a LPV system guarantees that the system output converges to the reference asymptotically. Solving this equations using the LMI method in a region the desired pole placement is obtained (see Figure 4.1), for the considered quadrotor system.

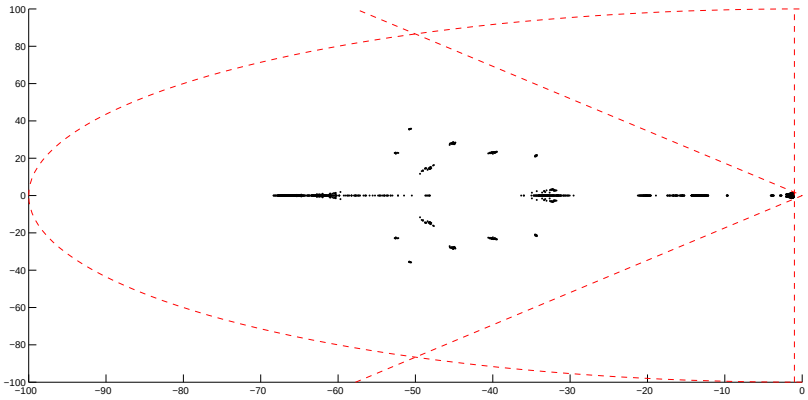

FIGURE 4.1: Pole Placement

In this thesis, three different parametrizations of the control law (4.6) will be used to control the same system and will be compared. According the equation (4.11), the feedback controller will be the same in all the cases.

4.1.1 Feedback Control with feedforward

The first case corresponds to a particular case of (4.6) that only uses the a feedback control. This method is the same explained in the background. The feedforward control problem is solved designing a control law with a feedforward component (4.12)

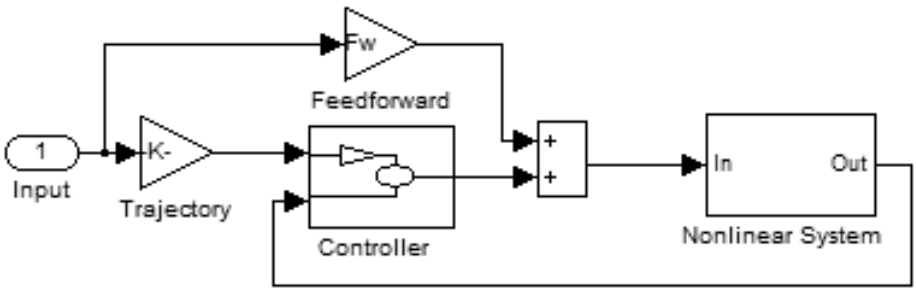

FIGURE 4.2: Feedback Control with Feedforward Component

$$u(t) = \bar{u}(t) + K_d(\rho(t))(x(t) - \bar{x}(t)) \tag{4.12}$$

This control law can be approached as the particular case in which $M = 0$ and $G = \mathcal{I}^{n_x}$ and $Q = \mathcal{I}^{n_u}$, where $\mathcal{I}^n$ is a n times n identity matrix.

In order to implement the control law (4.12), a trajectory $r(t)$, that will be designed such that the angles and the altitude will follow a desired path, is transformed into a state space reference $\bar{x}(t)$ and a feedforward control action $\bar{u}(t)$. Let us consider these desired trajectories as $\phi_r(t)$, $\theta_r(t)$, $\psi_r(t)$ and $z_r(t)$. These desired values and the actual ones will be used to obtain the reference for the velocity. Then, using a similar procedure, the desired acceleration of each angle is computed. These accelerations are used to be able to drive the system from the actual point to a desired one.

The state reference is given by the vector $\bar{x}(t) = [\dot{\phi}_r(t), \ddot{\phi}_r(t), \dot{\theta}_r(t), \ddot{\theta}_r(t), \dot{\psi}_r(t), \ddot{\psi}_r(t), \dot{\phi}_r(t), \ddot{z}_r(t), \dot{z}_r(t)]^T$ and the control feedforward is given by $\bar{u} = [u_1, u_2, u_3, u_4]^T$. These values are computed using the model (3.23d)-(3.23f).

4.1.2 Model Reference Control

In this case, the reference will be computed using a reference model. This model will be exactly the same as the nonlinear one but expressed in LPV form. The control law will be similar as the one explained before.

In order to be able to define the input reference, the model should be converted into the error model $e_{x_i} = \bar{x}_i - x_i$. As the reference model is defined as the real one, the reference relationships

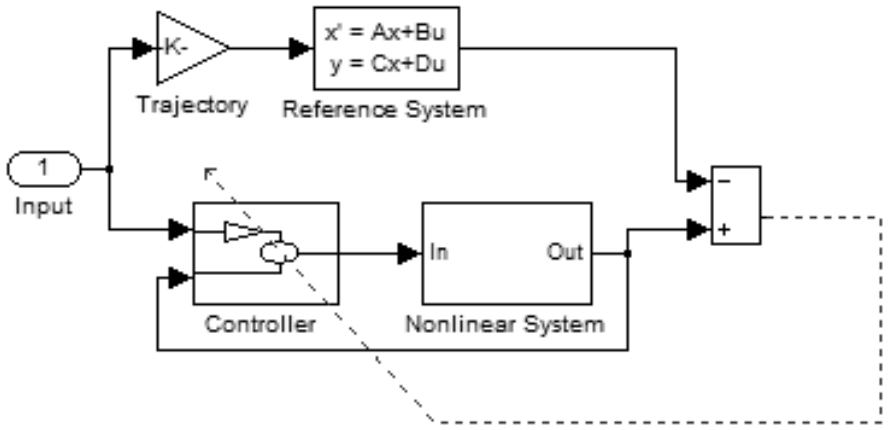

FIGURE 4.3: Reference Model

are defined using the same expressions as the real ones (3.23d)-(3.23f).

$$\ddot{\bar{z}} = -g + (cos\theta\, cos\phi)\frac{V_1}{m} \tag{4.13a}$$

$$\ddot{\bar{\phi}} = (\dot{\bar{\theta}}\dot{\psi} + \dot{\theta}\dot{\bar{\psi}})\frac{I_{yy} - I_{zz}}{2I_{xx}} - \frac{J_{TP}}{I_{xx}}\dot{\phi}\bar{\Omega} + \frac{V_2}{I_{xx}} \tag{4.13b}$$

$$\ddot{\bar{\theta}} = (\dot{\bar{\phi}}\dot{\psi} + \dot{\phi}\dot{\bar{\psi}})\frac{I_{zz} - I_{xx}}{2I_{yy}} - \frac{J_{TP}}{I_{yy}}\dot{\theta}\bar{\Omega} + \frac{V_3}{I_{yy}} \tag{4.13c}$$

$$\ddot{\bar{\psi}} = (\dot{\bar{\phi}}\dot{\theta} + \dot{\phi}\dot{\bar{\theta}})\frac{I_{xx} - I_{yy}}{2I_{zz}} + \frac{V_4}{I_{zz}} \tag{4.13d}$$

where V_i are the reference input control laws. Then, using the relationship $e_{x_i} = \bar{x}_i - x_i$, the error model is computed. It is assumed that the reference model uses the actual pitch and roll angles, so the trajectory will be computed on line

$$\ddot{e}_z = (cos\theta\, cos\phi)\frac{W_1}{m} \tag{4.14a}$$

$$\ddot{e}_\phi = (\dot{\bar{\theta}}\dot{\psi} + \dot{\theta}\dot{\bar{\psi}})\frac{I_{yy} - I_{zz}}{2I_{xx}} - \dot{\theta}\dot{\psi}\frac{I_{yy} - I_{zz}}{I_{xx}} - \frac{J_{TP}}{I_{xx}}\dot{\phi}\bar{\Omega} + \frac{J_{TP}}{I_{xx}}\dot{\phi}\Omega + \frac{W_2}{I_{xx}} \tag{4.14b}$$

$$\ddot{e}_\theta = (\dot{\bar{\phi}}\dot{\psi} + \dot{\phi}\dot{\bar{\psi}})\frac{I_{zz} - I_{xx}}{2I_{yy}} - \dot{\phi}\dot{\psi}\frac{I_{zz} - I_{xx}}{I_{yy}} - \frac{J_{TP}}{I_{yy}}\dot{\theta}\bar{\Omega} + \frac{J_{TP}}{I_{yy}}\dot{\theta}\Omega + \frac{W_3}{I_{yy}} \tag{4.14c}$$

$$\ddot{e}_\psi = (\dot{\bar{\phi}}\dot{\theta} + \dot{\phi}\dot{\bar{\theta}})\frac{I_{xx} - I_{yy}}{2I_{zz}} - \dot{\phi}\dot{\theta}\frac{I_{xx} - I_{yy}}{T_{zz}} + \frac{W_4}{I_{zz}} \tag{4.14d}$$

where W_i is the difference between the desired control input and the desired one.

4.1.3 Model Reference Control with Model Matching

Finally, a reference model that has a simpler dynamic behaviour than the real one has been used. This case needs a model matching controller in order to compare both models and control the system using the error.

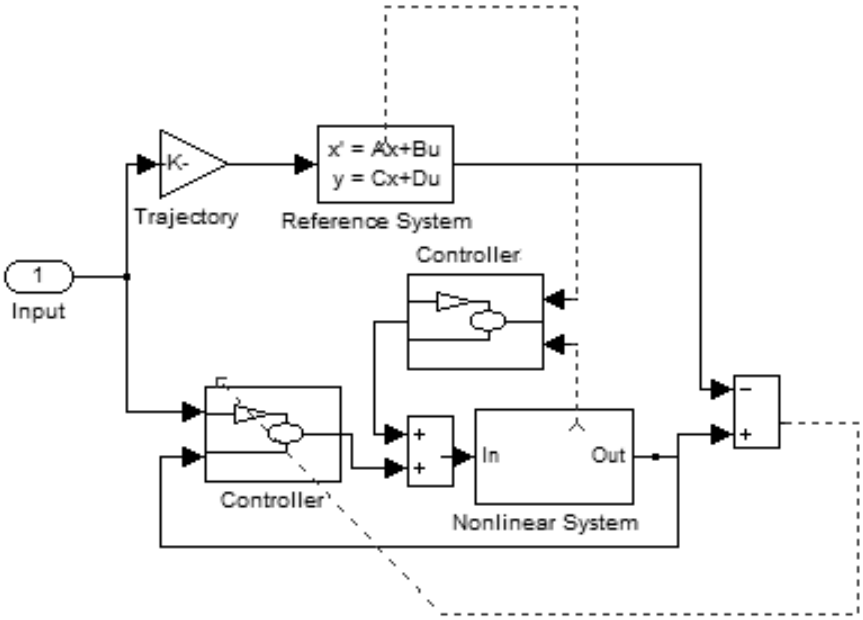

FIGURE 4.4: Reference Model

In this case, the reference model is simplified. In this master thesis, the nonlinear model without the coupled angles and expressed in LPV form will be used

$$\ddot{\bar{z}} = -g + (cos\theta\, cos\phi)\frac{V_1}{m} \tag{4.15a}$$

$$\ddot{\bar{\phi}} = \frac{V_2}{I_{xx}} \tag{4.15b}$$

$$\ddot{\bar{\theta}} = \frac{V_3}{I_{yy}} \tag{4.15c}$$

$$\ddot{\bar{\psi}} = \frac{V_4}{I_{zz}} \tag{4.15d}$$

where V_i are the reference input control laws. Using the relationship $e_{x_i} = \bar{x}_i - x_i$, the error model is computed. It is assumed that the reference model uses the actual pitch and roll angles, so the trajectory will be computed on line

$$\ddot{e}_z = (cos\theta\, cos\phi)\frac{W_1}{m} \tag{4.16a}$$

$$\ddot{e}_\phi = -\dot{\theta}\dot{\psi}\frac{I_{yy} - I_{zz}}{I_{xx}} + \frac{J_{TP}}{I_{xx}}\dot{\phi}\Omega + \frac{W_2}{I_{xx}} \tag{4.16b}$$

$$\ddot{e}_\theta = -\dot{\phi}\dot{\psi}\frac{I_{zz} - I_{xx}}{I_{yy}} + \frac{J_{TP}}{I_{yy}}\dot{\theta}\Omega + \frac{W_3}{I_{yy}} \tag{4.16c}$$

$$\ddot{e}_\psi = -\dot{\phi}\dot{\theta}\frac{I_{xx} - I_{yy}}{T_{zz}} + \frac{W_4}{I_{zz}} \tag{4.16d}$$

where $W_i = V_i - U_i$.

Chapter 5

Results Discussion

This chapter presents the results using the various control methods explained in previous chapter when applied to the quadrotor. Then, those results will be compared.

5.1 Introduction to the simulation scenarios

Various scenarios have been used in order to study the performances of the controllers and compare them, focusing on the error and the control input. All the simulations used the nonlinear model presented in Chapter 3 as real plant and use the same initial states. The initial pitch and roll is defined in a way that is inside the physical limits: $cos(\phi)cos(\theta) \in [\frac{-1}{4}, \frac{1}{4}]$ (See equations (3.51a)-(3.51b)).

5.2 Scenario 1

The first scenario used is an oscillation that is inside the physical bounds, but it does not reach any of them. The references used are the following ones

$$\phi(t) = \frac{\pi}{4} sin(\frac{\pi t}{5}) \tag{5.1a}$$

$$\theta(t) = \frac{\pi}{4} sin(\frac{\pi t}{5}) \tag{5.1b}$$

$$\psi(t) = \frac{\pi}{4} sin(\frac{\pi t}{5}) \tag{5.1c}$$

$$z(t) = 2sin(\frac{\pi t}{5}) \tag{5.1d}$$

5.2.1 Feedback Control

As the Figure 5.1 shows, the attittude and altitude state trajectory follows the desired path. The controller follows perfectly the dynamics of the reference computed using the method explained in Section 4.1.1. It shows no periodical error and the controller is fast. In the worst case, the settling time is around five seconds.

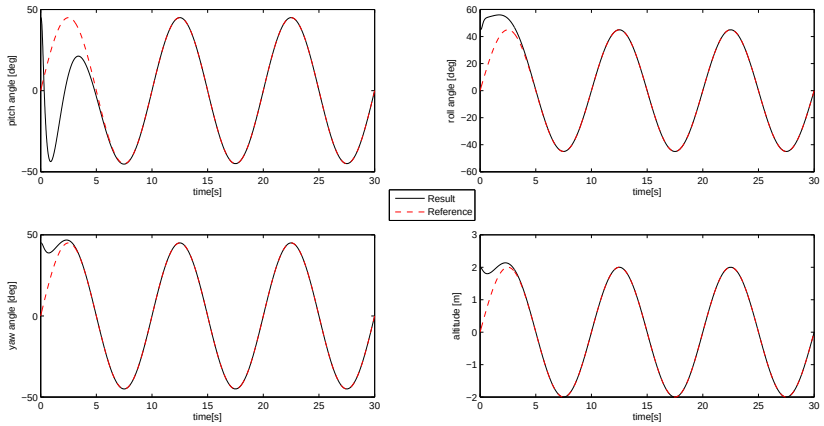

FIGURE 5.1: Scenario 1: Angular position and altitude control using the feedback approach

The error between the reference and the states tends asymptotically to 0, as shown in the Figure 5.2.

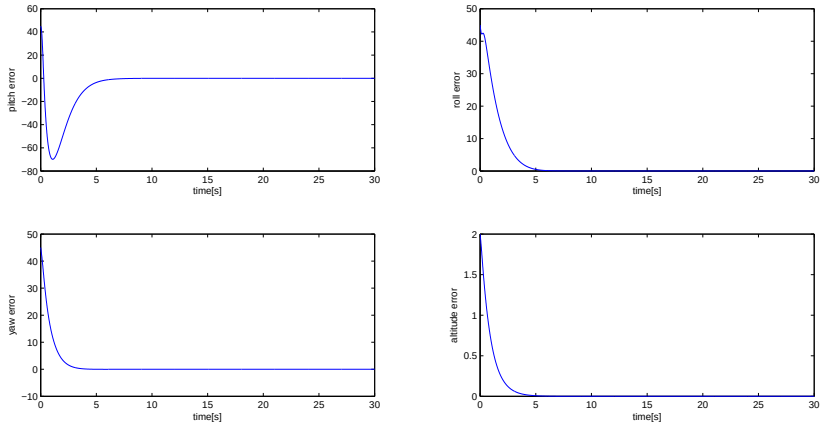

FIGURE 5.2: Scenario 1: Feedback control error

5.2.2 Reference Model Control

The same reference has been used in the reference model presented in the Section 4.1.2.

The Figure 5.4 shows the trajectory of the attitude and altitude states and the desired path. The controller follows perfectly the dynamics of the reference. It shows a little periodical error in the pitch response and its settling point is around six seconds in the worst-case scenario. The error of every variable, as it is shown in the Figure 5.3, tends to 0.

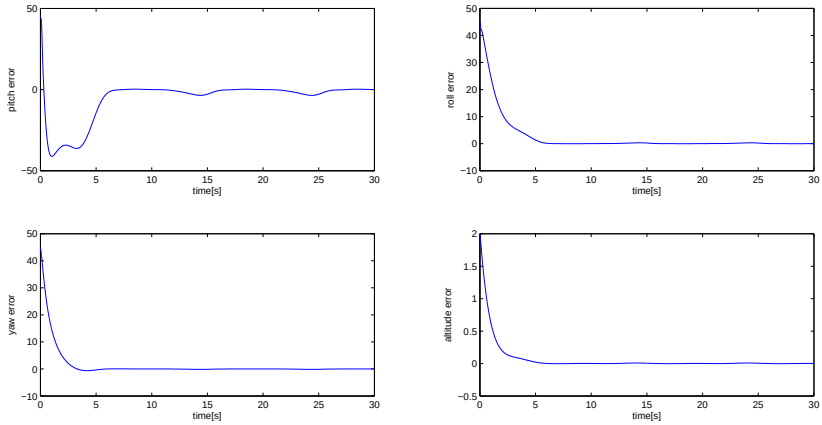

FIGURE 5.3: Scenario 1: Reference model error

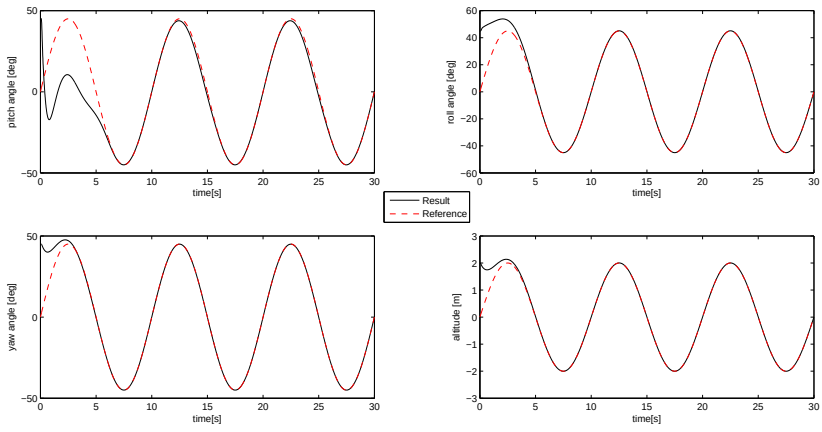

FIGURE 5.4: Scenario 1: Angular position and altitude control using the reference model approach

5.2.3 Reference Model Control with Model Matching

Figure 5.5 presents the path that will follow the attitude and altitude states and the desired trajectory. The controller follows the dynamics of the reference computed using the method explained in the Section 4.1.3. Its settling time is around seven seconds. As the Figure 5.6 shows, the error between the reference and system states tends to 0.

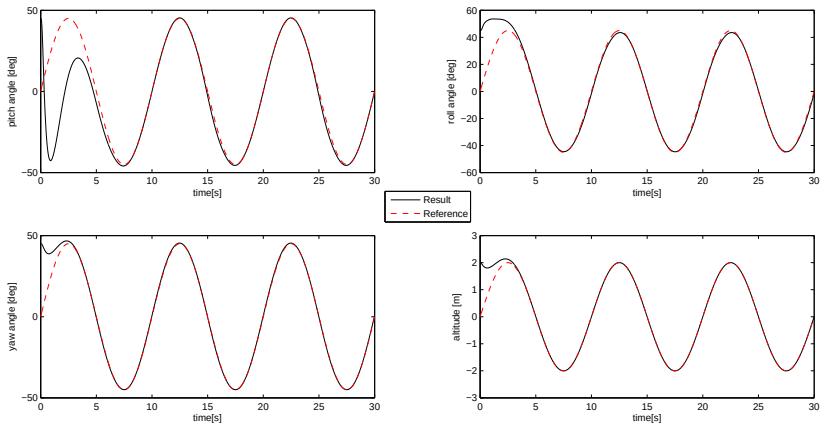

FIGURE 5.5: Scenario 1: Angular position and altitude control using the reference model with model matching approach

5.3 Scenario 2

As a second simulation scenario, a trajectory with a higher oscillation frequency has been used. By doing this, the speed of the response of those controller has been tested.

$$\phi(t) = \frac{\pi}{4} sin(2\frac{\pi t}{5}) \tag{5.2a}$$

$$\theta(t) = \frac{\pi}{4} sin(2\frac{\pi t}{5}) \tag{5.2b}$$

$$\psi(t) = \frac{\pi}{4} sin(2\frac{\pi t}{5}) \tag{5.2c}$$

$$z(t) = 2sin(\frac{2\pi t}{5}) \tag{5.2d}$$

5.3.1 Feedback Control

First, the feedforward approach has been used. The Figure 5.7 shows the attitude and altitude responses to the reference. As it is shown, the results are good. The controller follows perfectly the dynamics of the reference computed using the method explained before. It shows no periodical error and its settling time is about five seconds in the worst case. The error between the desired and system trajectories has a similar pattern as in the Scenario 1 (See Figure 5.2).

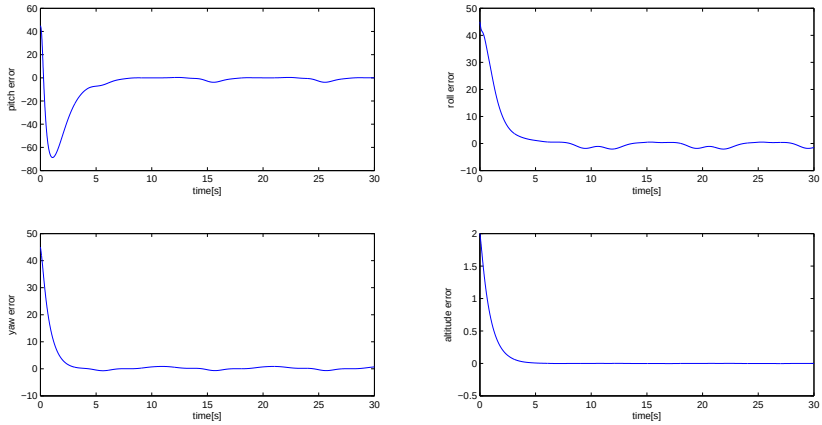

FIGURE 5.6: Scenario 1: Feedback control error

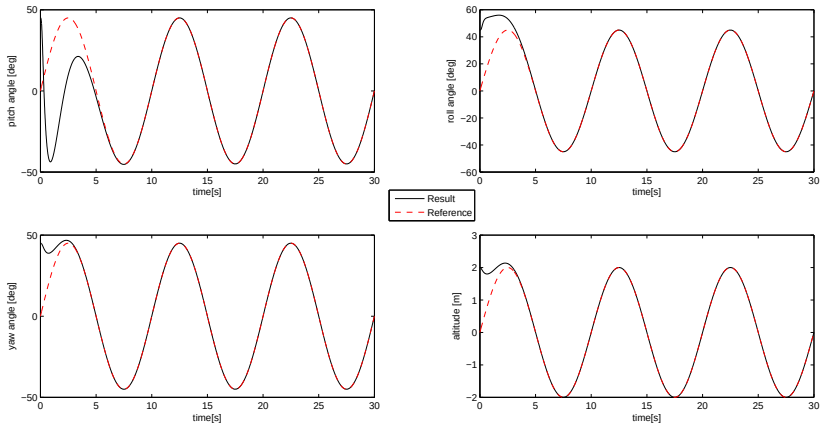

FIGURE 5.7: Scenario 2: Angular position and altitude control using the feedback approach

5.3.2 Reference Model Control

The same reference has been used in the reference model approach described in Section 4.1.2. Figure 5.8 presents the attitude and the altitude variables. It can be seen that the output follows almost perfectly the desired path. The response time is relatively fast, as the worst case settling time is around four seconds.

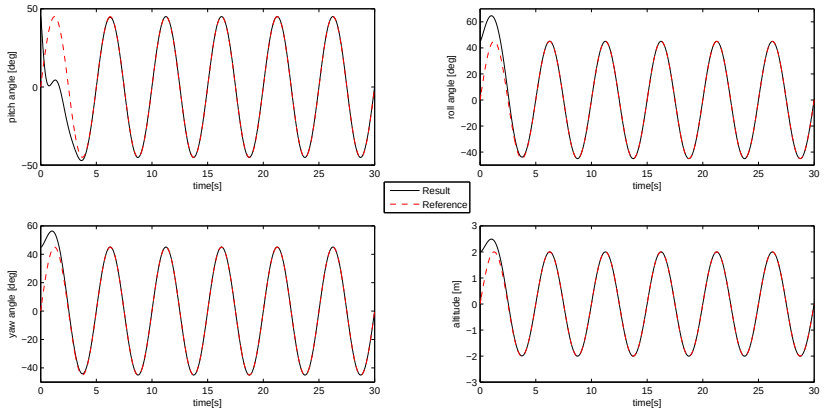

FIGURE 5.8: Scenario 2: Angular position and altitude control reference model approach

As the Figure 5.9 shows, the error between the reference and the output is nearly zero. The settling time in the worst-case is around four seconds.

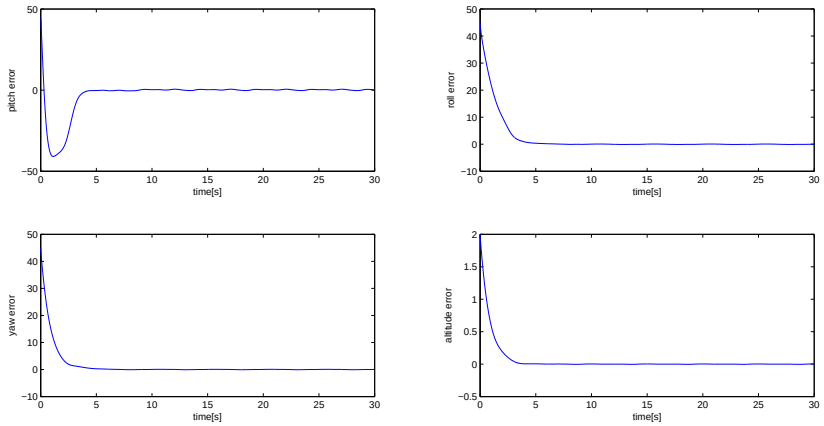

FIGURE 5.9: Scenario 2: Reference mode control error

5.3.3 Reference Model Control with Model Matching

Figure 5.10 presents the attitude and altitude state values in case of the reference model control with model matching. The trajectory shows errors in the peaks and the valleys following the reference path, specially in the roll case. The maximum relative error is around seven percent in the case of the pitch angle and two degrees in the case of the roll. In the case of the yaw and the altitude, the error is nearly zero. The response is fast. In the worst scenario, the settling time is around five seconds.

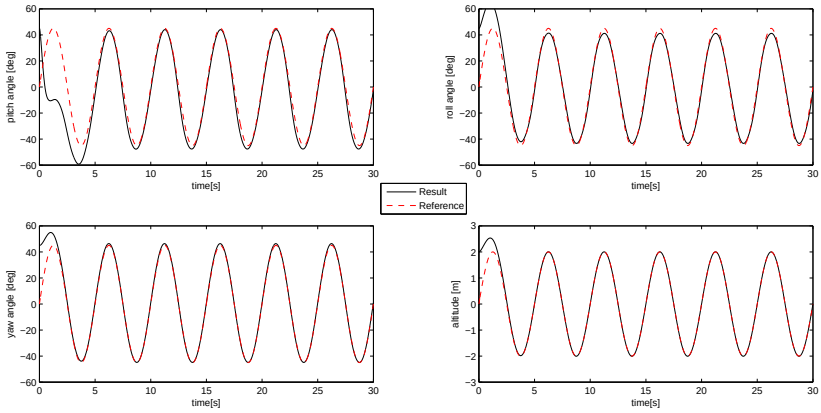

FIGURE 5.10: Scenario 2: Angular position and altitude control using the reference model control with model matching

5.4 Scenario 3

Finally, a trajectory that reaches the bound of the pitch and roll angles is used.

$$\phi(t) = \frac{\pi}{3} sin(\frac{\pi t}{4}) \tag{5.3a}$$

$$\theta(t) = \frac{\pi}{3} sin(\frac{\pi t}{4}) \tag{5.3b}$$

$$\psi(t) = \frac{\pi}{2} sin(\frac{\pi t}{4}) \tag{5.3c}$$

$$z(t) = 2 sin(\frac{\pi t}{5}) \tag{5.3d}$$

5.4.1 Feedback Control

First of all, this scenario has been tested using the feedback control with feedforward approach described in the Section 4.1.1.

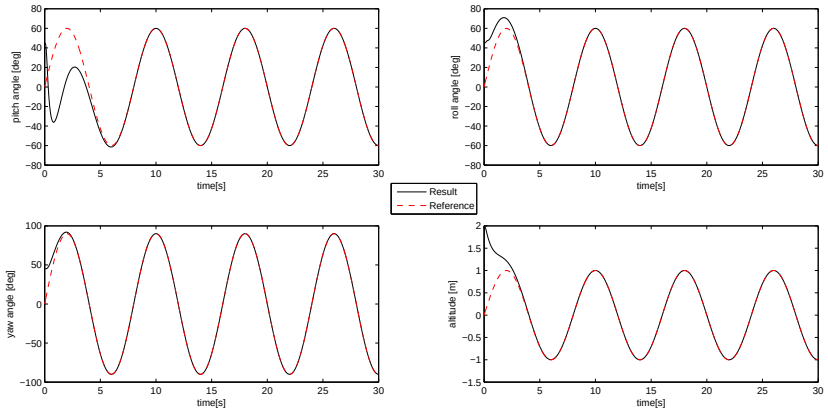

FIGURE 5.11: Scenario 3: Angular position and altitude control using the feedback approach

Figure 5.11 shows the trajectory that the states follow. The controller follows perfectly the dynamics of the reference computed using the method explained before. Its settling time is about five seconds in the worst case. The error of every variable tends asymptotically to 0.

5.4.2 Reference Model Control

The same reference has been used in the second method proposed based on the reference model approach.

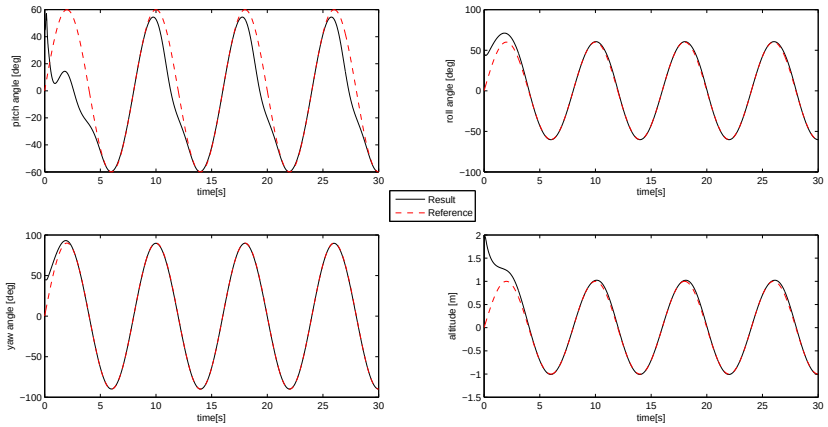

FIGURE 5.12: Scenario 3: Angular position and altitude control using the model reference approach

Figure 5.12 shows the attitude and altitude states and the desired path. The quadrotor reaches the set-point in around four seconds in the worst case scenario. However, an error between the state and the reference of around twenty degrees in the pitch angle response appears. This error, as shown in the Figure 5.13, present a periodical pattern. The other responses have a much better performance.

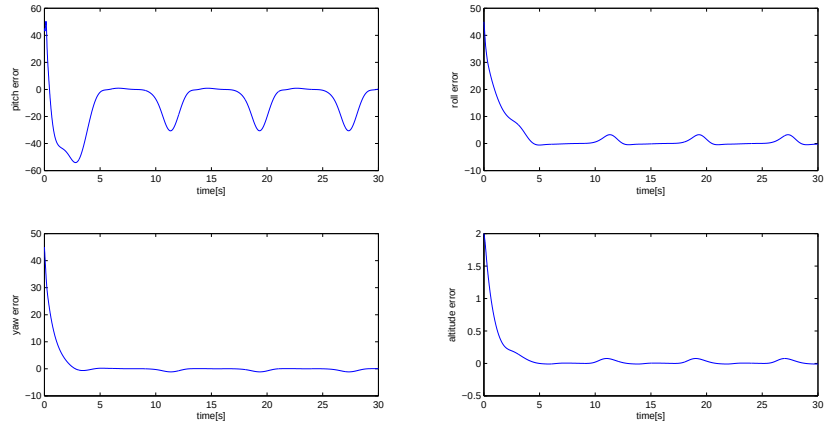

FIGURE 5.13: Error Pattern in the Scenario 3

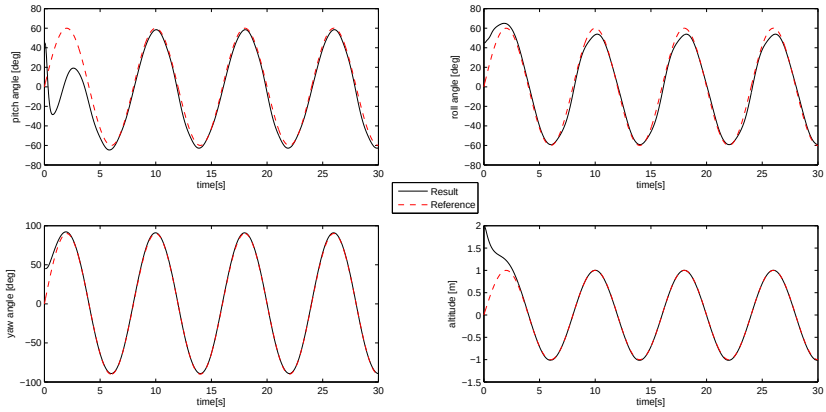

FIGURE 5.14: Scenario 3: Angular position and altitude control

5.4.3 Reference Model Control with Model Matching

Figure 5.14 shows the trajectory followed by the variables and the desired path. The controller can follow the dynamics of the computed reference. However, it has an error between the reference and the system state of around nine degrees in the pitch case and ten degrees in the roll case. It has, as in the previous scenario, a periodical pattern as shown in the Figure 5.15.

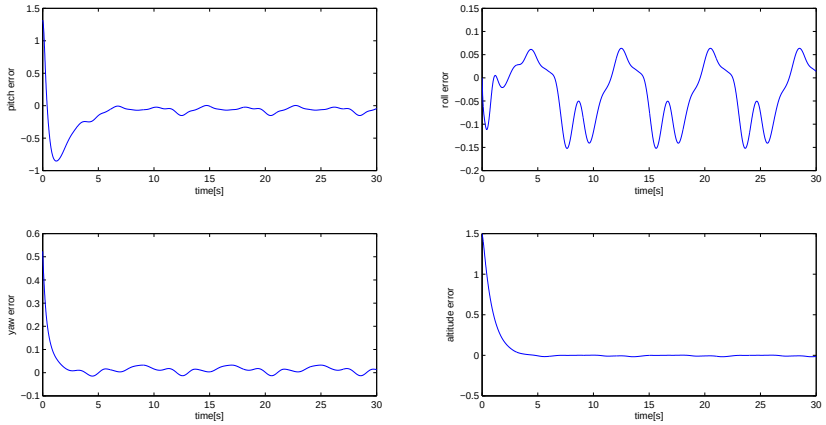

FIGURE 5.15: Error Pattern in the Sceneraio 3 using the model reference with model matching controller

5.5 Comparison

As it have been observed comparing the results obtained in the different simulation scenarios, the feedback controller is the approach that obtained the best performances, as it is able to follow the desired trajectory with the required dynamics. In all the scenarios, the error has a asymptotic tendency to 0.

TABLE 5.1: Comparison of the three schemes results in the first scenario considering the worst case scenario

	Settling Time $[s]$	Overshoot $[\%]$	Maximum Absolute Error	Maximum Relative Error $[\%]$
Scheme 1	5.3	117	7.76	0.6
Scheme 2	6.4	120	0.31	0.9
Scheme 3	6.7	116	2	5.6

As it is shown in the Table 5.1, in the first scenario all the methods have achieved good results. The reference model controller with model matching is the worst one because of a periodical error in the maximum of the desired trajectory. The feedback controller has a higher maximum settling time in the response to the pitch angle reference than the reference model.

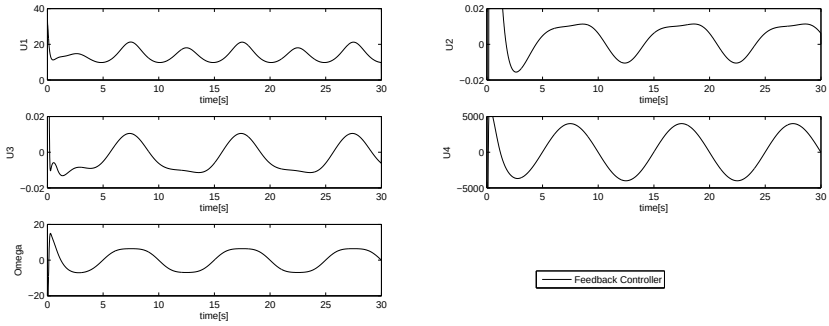

FIGURE 5.16: Control Inputs for the reference approach in the scenario 1

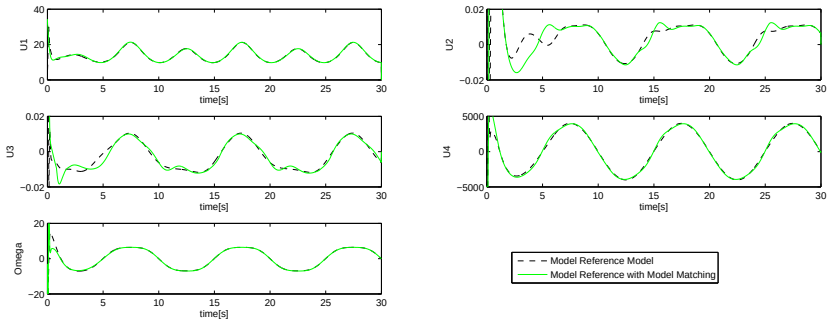

FIGURE 5.17: Control Inputs for the model based approaches in the reference model approaches

As the Figures 5.16-5.17 show, the control inputs (3.15),(3.17) are similar among all the schemes. The only one that has a difference is the reference model with model matching.

TABLE 5.2: Comparison of the three schemes results in the second scenario considering the worst case scenario

	Settling Time $[s]$	Overshoot $[\%]$	Maximum Absolute Error	Maximum Relative Error $[\%]$
Scheme 1	4.6	144	7.76	3.5
Scheme 2	3.9	146	0.96	0.5
Scheme 3	6.3	127	3	6.6

In the case of the second scenario, the results of the responses is simmilar to the first one, as shown in the Table 5.2. The reference model with model matching has a periodical error in the maximum of the reference and the reference model and the feedback controller 's error tends asymptotically to zero.

TABLE 5.3: Comparison of the three schemes results in the third scenario considering the worst case scenario

	Settling Time $[s]$	Overshoot $[\%]$	Absolute Error	Relative Error $[\%]$
Scheme 1	5.9	144	14.41	3.5
Scheme 2	3.9	146	31	26
Scheme 3	6.3	127	3	6.6

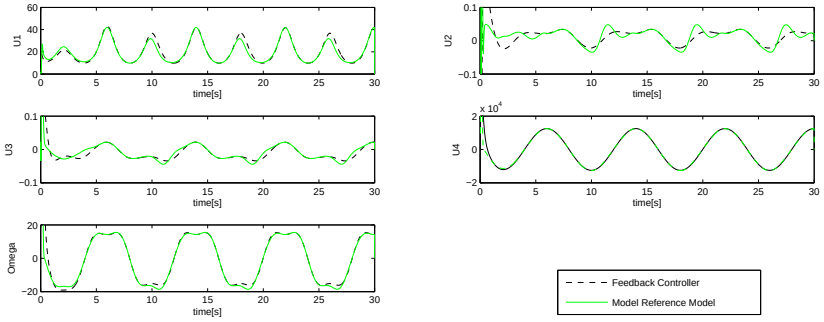

FIGURE 5.18: Control Inputs for the model based approaches in the reference model with model matching case

As shown using the third scenario, the feedforward approach performs better in trajectories near the physical bounds and with higher frequencies. In this scenario, as presented in Figure 5.18, it can be noticed that the control of the roll and pitch is a bit different and appearing peaks in the same time step in which the error in the trajectory occurs. This error is relatively high. However, in the case of the second control scheme, the error is nearly null in all the variables but the pitch, in which is really high.

Chapter 6

Conclusions and Further Research

This chapter summarizes the conclusions and main contributions of the thesis. It also presents some proposals of possible future research.

6.1 Conclusions

This thesis has aimed at developing a controller for a quadrotor using LPV control techniques. To achieve this goal, the quadrotor nonlinear model has been transformed into an LPV one. Then, the system has been prefiltered in order to avoid to have parameter variances in the control matrix.

Several LPV control schemes have been proposed. The first controller needed a feedforward component in order to track a trajectory. This was computed using the model reference equations. The other two were model based controllers. In this case, the control used the error model instead. The second model method used a reference model that was simplified. In order to be able to compare dynamics, a model matching subsystem was used.

To compare the performances achieved with the different control schemes proposes, various simulations have been done. The first simulation scenario corresponds to the case where the desired reference path was inside the physical bounds. In this simulation scenario, all the controllers showed really good results. The second scenario corresponds to a trajectory that showed a higher frequency. In this case, the model reference based controllers showed small periodical errors. Finally, in the last scenario, the reference used forces the quadrotor reached the physical limits. In this simulation, the feedback control performed as before. However, the model reference approaches' error increased.

The feedback controller is the one that has performed better in the tests and the model reference control with model matching had the worse results. The model reference control is the scheme that showed a quicker settling time in all cases. However, its results near the settling time are the worst. Despite all that, the control law performances is similar in all the cases.

6.2 Ongoing and Further Research

The main objective would be to compare the proposed controllers using a real quadrotor and then try it in a more complex helicopter. To do that, various steps must be followed:

- In this Master Thesis, it has been considered that all the states are measured. However, that may not be true in a real quadrotor, so an observed should be designed.
- In order to have a quadrotor that is able to follow a desired trajectory, the x and y should be controlled. To do so, the controller should be modified to control all the coordinates.
- Identification of the real parameters of the quadrotor are required in order the developed controller could work.
- Modelling mismatches with the real quadrotor, errors in the identification of the model and noise in the sensors will introduce disturbances in the system. Those effect should be studied in order to see how they will affect the performance of the designed controllers when applied to a real quadrotor.
- When the controller has been completed and the simulations show the results desired, it should be implemented in the real system.

Bibliography

[1] A. Abdullah and M. Zribi. Model reference control of LPV systems. *Journal of the Franklin Institute*, 346:854–871, 2009.

[2] P. Apkarian, P. Gahinet, and G. Becker. Self-scheduled H_∞ Control of Linear Parameter-Varying Systems: A Design Example. *Automatica*, 31(9):1251 – 1261, 1995.

[3] T. Bresciani. *Modelation, Identification and Control of a Quadrotor Helicopter*. PhD thesis, 2008.

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

[5] G. Chowdhary, E. Frazzoli, J.P. How, and H. Liu. Nonlinear flight control techniques for unmanned aerial vehicles. 2013.

[6] G.M. Hoffmann and S.L. Waslander. Quadrotor helicopter trajectory tracking control. In *Proceedings of the AIAA Guidance, Navigation and Control Conference and Exhibit, Honolulu, HI, USA*, pages 1–14, 2008.

[7] A. Mokhtari and A. Benallegue. Dynamic feedback controller of Euler angles and wind parameter estimation for a quadrotor unmaneed aerial vehicle. In *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA), New Orleans, LA, USA*, pages 2359–2366, 2004.

[8] D. Rotondo, F. Nejjari, and V. Puig. Fault tolerant control design for polytopic uncertain LPV systems. In *Proceedings of the 21st Mediterranean Conference on Control and Automation, Platanias-Chania, Crete, Greece*, pages 66–72, 2013.

[9] D. Rotondo, F. Nejjari, and V. Puig. Quasi-LPV modeling, identification and control of a twin rotor MIMO system. *Control Engineering Practice*, 21, 2013.

[10] D. Rotondo, F. Nejjari, and V. Puig. Model reference quasi-LPV control of a quadrotor UAV. In *IFAC*, 2014.

[11] D. Rotondo, F. Nejjari, and V. Puig. RObust Quasi-LPV Model Reference FTC of a Quadrotor UAV subject to actuator faults. *International Journal of Applied Mathematics and Computer Science*, page 7, 2015.

[12] R. Samarathunga and J. Whidborne. Linear Parameter Varying control of a quadrotor. In *6th International Conference on Industrial and Information Systems*, pages 483–488, 2011.

[13] Seung Hoon Kang Sang-hyun Lee and Youdan Kim. Trajectory tracking control of quadrotor UAV. *Control, Automation and Systems (ICCAS)*, pages 281 – 285, 2011.

[14] J. S. Shamma. An overview of LPV systems. In J. Mohammadpour and C. Scherer, editors, *Control of Linear Parameter Varying Systems with Applications*, pages 3–26. Springer, 2012.

[15] Bouhali O. Khebbache H. Yacef, F. and F. Boudjema. Takagi-Sugeno Model for Quadrotor Modelling and Control using State Feedback Controller. *International Journal of Control Theory and Computer Modeling*, page 9, 2012.