ALMA MATER STUDIORUM · UNIVERSITÀ DI BOLOGNA

Scuola di Ingegneria e Architettura
Corso di Laurea in Ingegneria dell'Automazione

Modeling and Control of a Tilting Rotor UAV

Supervisors:
Chiar.mo Prof.
Lorenzo Marconi

Chiar.mo Prof.
Matteo Fumagalli

Presented by:
Giovanni AMBRUOSO

♯ Sessione I
Anno Accademico 2017/2018

ALMA MATER STUDIORUM · UNIVERSITÀ DI BOLOGNA

Abstract

Scuola di Ingegneria e Architettura
Corso di Laurea in Ingegneria dell'Automazione

Master Degree

Modeling and Control of a Tilting Rotor UAV

by Giovanni AMBRUOSO

This dissertation aims at analyzing the Unmanned Aerial Vehicles (UAVs) involved in aerial manipulation strictly related to the surrounding environment. In order to extend as much as possible the entire operating flying time, the main focus is on energy consumption. The dissertation takes into account the architecture characterized by a quad-rotor drone with two over four tilting propellers. There is a comparison between the energy and power consumption of this new architecture with a traditional quad-rotor structure with fixed-propeller. The comparison mainly focused on two energy aspects: the instant power spent by all four motors in both cases, and the cost function involved during the control of both structures. Cost functions values are extracted by an optimal control problem setting in the two different cases. Advantages and disadvantages were proved concerning the tilting rotor solution in terms of energy consumption and involved dynamics.

Acknowledgements

I would to thank first my supervisors, Prof. Lorenzo Marconi and Prof. Matteo Fumagalli for the time they dedicated to me, and for the opportunity they gave to me to work by their side. For the same reason, I would to thank all their assistants and colleagues who received me when I was stuck on my work: thanks to Alessandro Melis, Nicola Mimmo, and David Wuthier. I would to give big thanks to whom helped and supported my unperfect English, thanks to Paola and Giorgia. This latter is thanked double because she is part of my wonderful family which always supported me. So an enormous thanks goes to my father Luigi, thanks for the *sponsorship* and your infinite support. As last but not least all my friends and colleagues which were constantly at my side also in darkest and most difficult moments. Thank you to all of you!

Contents

List of Figures

Chapter 1

Introduction

1.1 State of the art

The applications of Unmanned Aerial Vehicles (UAVs) widely diversified during the last years. The main task of UAVs is to provide a mobile extension of human perceptions. It gives security to the user (soldiers [1], policeman [2], cameraman [3]) and it also gathers informations such as images or video, locations coordinates, weather conditions [4], etc., for either online or offline analysis. However, there are missions whose task is to go beyond the capabilities of conventional UAVs designs since they require not only flight endurance capabilities but also object manipulating skills through physical interaction with the environment. In [5] a helicopter had success in grasping an unknown object while hovering and then reach flying stability; in [6] a quad-rotor is able to grasp a moving object; in [7] a quad-rotor is equipped with a robotic arm able to exert known pulling forces on the environment.

In particular quad-rotor UAVs have also recently proven to be an effective platform for aerial manipulation and inspection: in [8] it is showed that a quad-rotor can inspect and analyze industrial plants, so basically it is able to be involved in an unknown environment. A good example of transportation and cooperation between a group of UAVs is showed in [9] in which is designed a fleet of drones able to build a cubic architecture grasping and assembling primary elements.

In this wide set of applications some researches introduced a new design for multi-rotor drones to increase maneuverability in manipulation tasks. It is based on tilting rotors, i.e multi-rotor drones with propellers which can tilt their own rotation axis. This new concept of quad-rotor is treated in [10] where is showed the increasing of degrees of freedom in terms of position and orientation introducing tilting propellers. In [11] the same design has been applied to an exa-rotor drone in order to study two different controls in cascade concerning position and attitude. In [12] a great result was obtained in term of decoupling position and attitude problem allowing the drone to hover keeping its body in a vertical position. It could be an useful skill for manipulating tasks.

Despite good results in maneuverability, in evaluating tilting rotors literature it appeared that no one focused enough on power consumption of this solutions. This latter aspect is crucial in development of quad-rotors, or drones in general, and deserves to be analyzed with carefulness. Moreover, the overall flying time is another aspect strictly related to the energy consumption during manipulation tasks. Some solutions in this direction have been introduced in [13] by perching the quad-rotor on a smooth surface in order to switch off the motors increasing the entire operational time from minutes to hours. The same behavior is described in [14] also providing an explicit remaining flight time estimation. With the aim of reducing the amount of energy spent during flight some researchers developed an algorithm that can estimate the best energy saving trajectory of a fixed-propeller quad-rotor [15]. In order to save energy, some researches focalized their attention on strictly mechanical aspects and designs. In [16] and [17] an innovative scheme was tested to improve mechanical efficiency and increase flight time. They realized a quad-rotor with overlapped propellers' areas in order to improve not only the energy saving but also the payload capacity. In [18] and [19] they focused on a general concept of efficiency on a fixed-propellers quad-rotor. They mainly treated mechanical aspects as propellers' aerodynamical shape, blade's pitch angles, and electronics energy consumption.

In conclusions it emerged that very few of analyzed studies and papers focused on evaluating energy consumption of a tilting rotor architecture. To assert appreciable advantages in aerial manipulation of this new design and justify this constructive solution, a study in this direction is needed.

1.2 Main goals

This dissertation aims at study a solution which merges skills required for aerial manipulation with energy saving during flight. The attention was put on the evaluation of effective advantages in terms of efficiency using tilting rotors in aerial manipulation tasks. It was simulated a quad-rotor structure with two over four tilting propellers. In order to determine whether a tilting rotors architecture is an endurance and power conservative solution, the entire study is presented following the perspective of energy savings. According to this purpose it is analyzed within Matlab/Simulink environment the overall power consumption of a model of a tilting propellers quad-rotor drone during some sample situations. In the same environment it was also simulated a fixed-propeller quad-rotor drone. In conclusion both architectures' simulations are compared obtaining considerable results. Furthermore, an optimal control problem was set to compare the total amount of energy spent during control in both cases. As before, cost functions results are compared and discussed. Moreover, mechanical torque power is calculated and compared with cost functions' values.

1.3 Chapters overview

The dissertation is divided into the following chapters:

- **Chapter 1 - Introduction**
  It was presented an overview of related works conducted in other projects and researches. Some papers which treated similar topics was analyzed. The dissertation project was exposed with motivations and goals. In the end a chapters overview is listed.

- **Chapter 2 - Mathematical models and Simulink implementation**
  In this chapter, non linear mathematical models of tilting and fixed propellers quad-rotor drones are exposed. They derived from analysis of Euler's equations. Afterwards, a Matlab/Simulink model implementation derived from equations of mathematical model is presented. Furthermore it is treated the PID control tuning for position and attitude controls.

- **Chapter 3 - Simulations and results**
  Matlab/Simulink simulation results are discussed in this chapter focusing on both simulated architectures. Dynamics, power and energy consumption are compared in order to assert energy saving.

- **Chapter 4 - Cost Function and Optimal Control**
  In this chapter an optimal control problem is set in order to evaluate cost function values in different cases. Both architectures are tested on

this point of view. Even in this chapter, new dynamics and energy consumptions are evaluated and compared. In the end results from analysis are exposed and discussed.

- **Chapter 5 - Conclusions**
  A conclusion about the whole project is reached here; the observations about the overall results and possible future works are listed in this chapter.

Chapter 2

Mathematical models and Simulink implementation

The common architecture for a quad-rotor drone is characterized by fixed propellers, i.e. described by four rotation axes all parallel to each other. In this dissertation this common structure is compared with another structure equipped with two over four tilting propellers. In order to simulate the dynamical behavior of physical architectures, it's necessary to define their mathematical model. In this chapter a mathematical model is developed concerning both simulated architectures. In section 2.1 the fixed propellers structure will be analyzed in all its characteristics starting from physical laws; in section 2.2 will be treated the unusual tilting propeller architecture describing its differences with the fixed propellers structure.

2.1 Fixed propellers structure

The drone sample structure is the quad-rotor. As shown in Fig. 2.1, it is made of four arms at the end of which there is a propulsion unit. Each propulsion unit is composed by a brushless motor and a propeller linked to it (Fig. 2.2). In the model considered in simulations, every propulsion unit is identified by a number from 1 to 4, and it is characterized by a local reference frame centered on each motor.

2.1.1 Propulsion unit working principles

As a result, each propeller produces two main effects: a thrust force F_i expressed by the equation (2.1), and a reaction torque τ_i expressed by the equation (2.2).

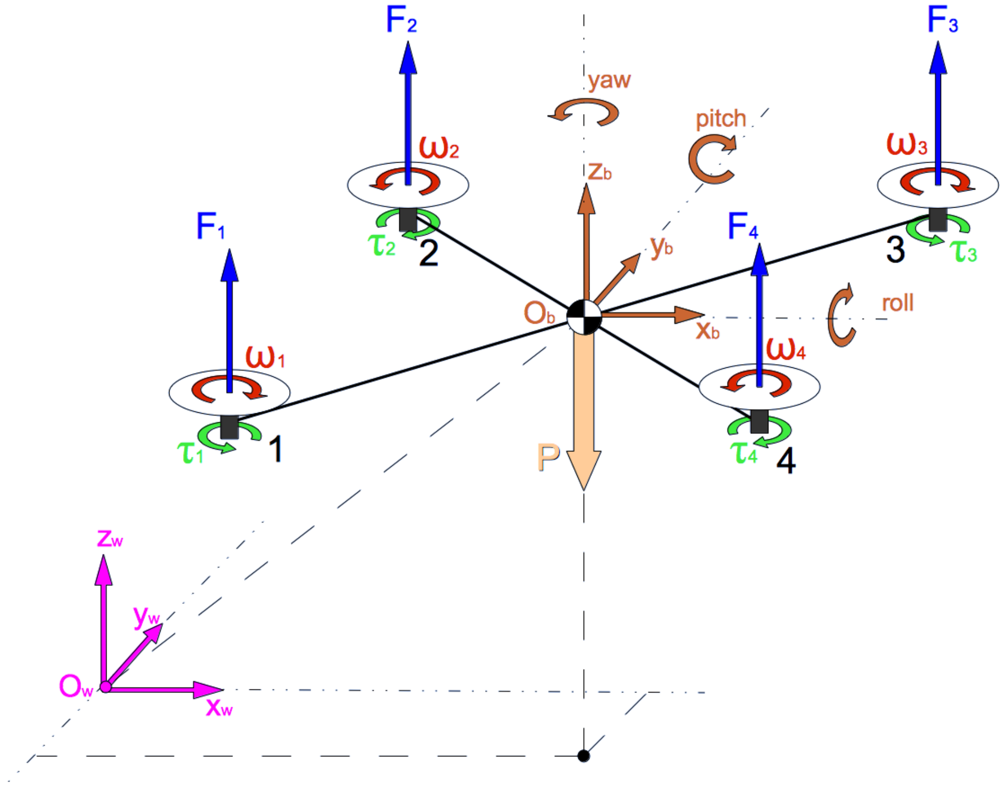


FIGURE 2.1: The simplified structure of a drone analyzed to obtain the model

$$F_i = k \cdot \omega_i^2 \tag{2.1}$$

Where F_i is the thrust force exerted by the i-th propeller, k is an aerodynamic coefficient which depends on propeller diameter, fluid density, flight velocity and others parameters. To simplify the process, any detail about this coefficient is omitted. In the end, ω_i is the angular velocity of the i-th propeller.

The thrust force of each propulsion unit is directed along its own z-axis. In order to hover in a steady point in the space, all the thrust forces applied to the body of the quad-rotor needs to counteract the overall weight P.

The second main effect produced by rotation of propellers is a reaction torque transmitted from the motor to the frame of the drone. It is due to drag forces developed by a propeller rotating in a fluid; its formula is exposed in (2.2).

$$\tau_i = \pm b \cdot \omega_i^2 \tag{2.2}$$

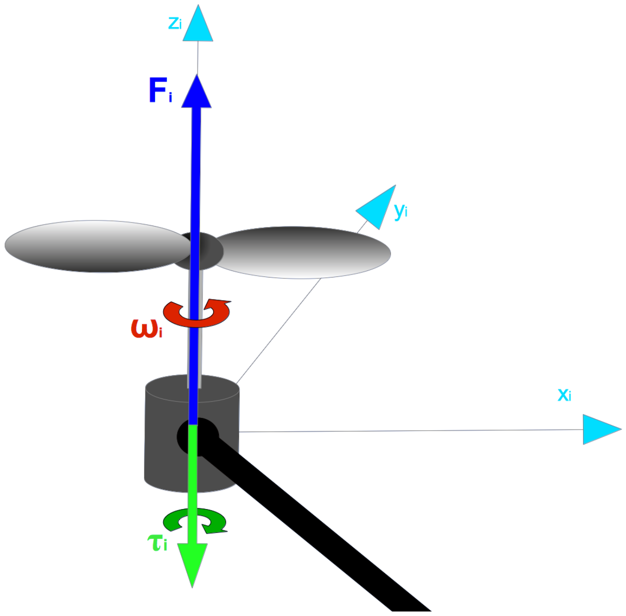


FIGURE 2.2: Representation of forces and torque expressed by a propeller

Where τ_i is the drag torque produced by the i-th propeller; b is a drag coefficient, which depends from propeller's dimensions and angle of attack, air density, and other aerodynamical parameters; ω_i is the angular velocity of the i-th propeller.
Reaction torque τ_i is always counter rotating compared to the angular speed direction. Thus the sign in (2.2) is decided following this rule. Furthermore, to balance all the reaction torques, propellers must rotate alternatively in clockwise and counterclockwise direction (Fig. 2.1).

2.1.2 Roll, Pitch and Yaw

As in Fig. 2.1, two main reference frames were considered. The inertial frame, or world reference frame, is represented in purple and has its origin in O_w. The body fixed frame is represented in orange and its origin O_b is fixed in the center of gravity of the structure. The latter reference frame is used to define body rotations described by the three Euler angles of roll, pitch and yaw. A vector rotation in a three-dimensional space is always represented by 3x3 matrix. In particular, roll (ϕ) is a rotation around x-axis of body fixed frame and it is defined in (2.3); pitch (θ) is a rotation around y-axis in body fixed frame defined in (2.4); and yaw (ψ) is a rotation around z-axis in body fixed frame defined in (2.5). The rotation of each Euler angle follows the right-hand rule and it is based on corresponding axis direction.

$$R(\phi) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\phi) & -\sin(\phi) \\ 0 & \sin(\phi) & \cos(\phi) \end{bmatrix} \tag{2.3}$$

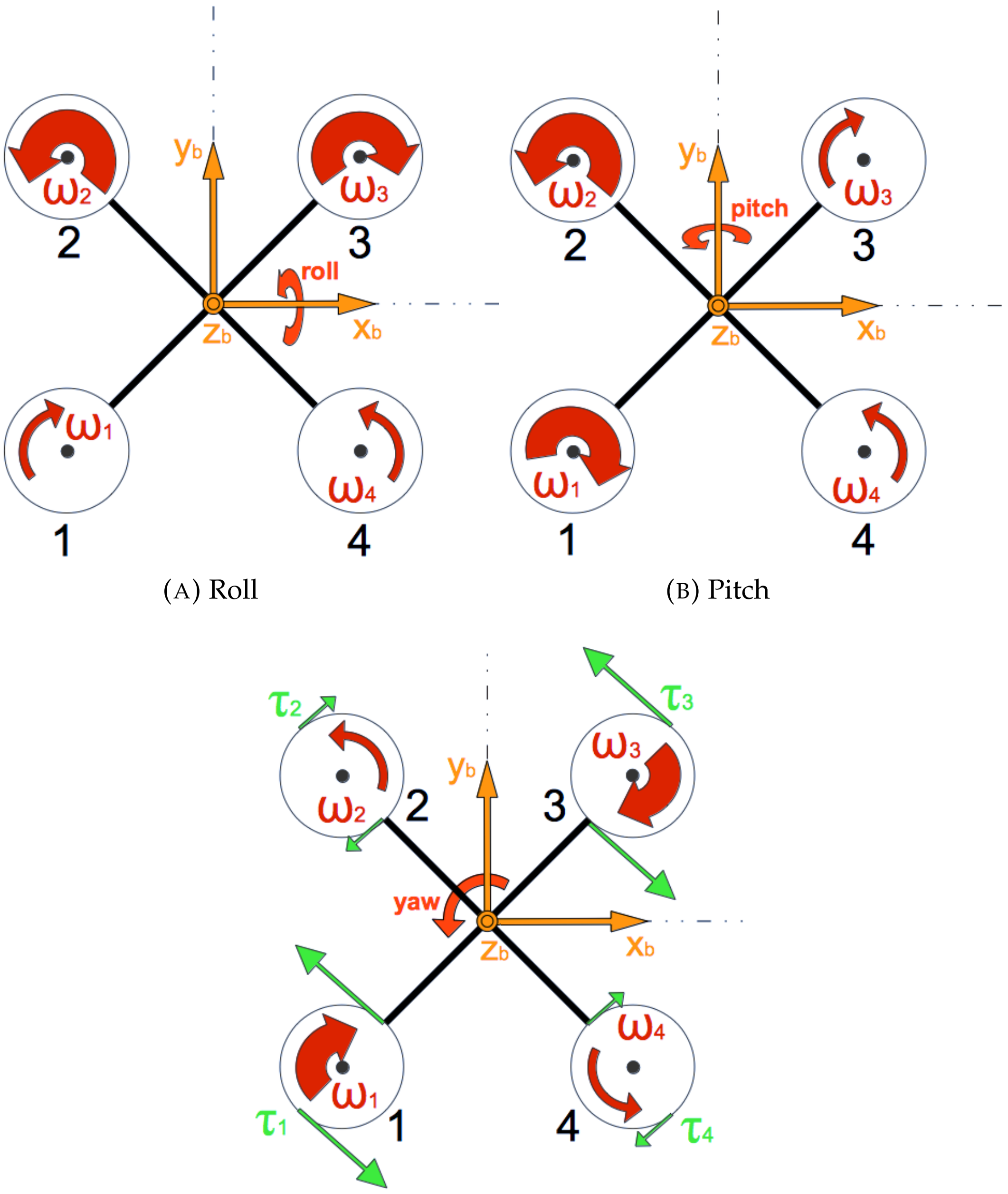
$$R(\theta) = \begin{bmatrix} \cos(\theta) & 0 & \sin(\theta) \\ 0 & 1 & 0 \\ -\sin(\theta) & 0 & \cos(\theta) \end{bmatrix} \tag{2.4}$$

$$R(\psi) = \begin{bmatrix} \cos(\psi) & -\sin(\psi) & 0 \\ \sin(\psi) & \cos(\psi) & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{2.5}$$

The roll, pitch, and yaw rotations can be used to place the body in any orientation in three-dimensional space. Therefore, a single rotation matrix can be obtained by multiplying the roll, pitch, and yaw rotation matrices.

$$\begin{aligned} R(\psi,\theta,\phi) &= R(\psi)R(\theta)R(\phi) \\ &= \begin{bmatrix} C_\psi C_\theta & C_\psi S_\theta S_\phi - S_\psi C_\phi & C_\psi S_\theta C_\phi + S_\psi S_\phi \\ S_\psi C_\theta & S_\psi S_\theta S_\phi + C_\psi C_\phi & S_\psi S_\theta C_\phi - C_\psi S_\phi \\ -S_\theta & C_\theta S_\phi & C_\theta C_\phi \end{bmatrix} \end{aligned} \tag{2.6}$$

Where, for the sake of simplicity, C_x represents cosine of x and S_x represents sine of x. As in any matrices multiplication, the order is important even concerning the roll-pitch-yaw notation. It implies that the first rotation applied to a vector is on the right, the last on the left.
The quad-rotor can perform this three main rotations by properly changing the angular speeds of each propeller. In Fig. 2.3 is represented how each single rotation can be expressed adapting angular speeds and consequently changing forces and torques following (2.1) and (2.2).



(A) Roll

(B) Pitch

(C) Yaw

FIGURE 2.3: A representation of how to perform roll, pitch and yaw changing propellers angular speeds ω_i

To achieve a positive roll rotation (Fig. 2.3a) motors number 2 and 3 needs to increase the angular speed of their own propeller, increasing thrust forces and drag torques. Conversely, motors 1 and 4 act contrariwise, they decrease angular speed to reduce corresponding forces and torques allowing the body to rotate around the x_b-axis. The pitch rotation around y_b-axis (Fig. 2.3b) is performed in a similar way. The motors which produce greater forces are 1 and 2, instead 3 and 4 decrease their thrusts and torques.

To rotate around z_b-axis according to its direction, it is required to increase the angular speeds of those motors which have counterclockwise drag torque direction (Fig. 2.3c). In the considered figures these motors are numbered 1 and 3. At the same time, motors number 2 and 4 decrease their angular speeds and consequently their clockwise drag torques reduce too.

2.1.3 Euler's equations

In this subsection the Euler's equations are exposed in order to analyze dynamics involved in the fixed propeller quad-rotor simulation. Forces and torques dynamics are treated in two different paragraphs. In Fig. 2.4 is represented the considered three-dimensional model.

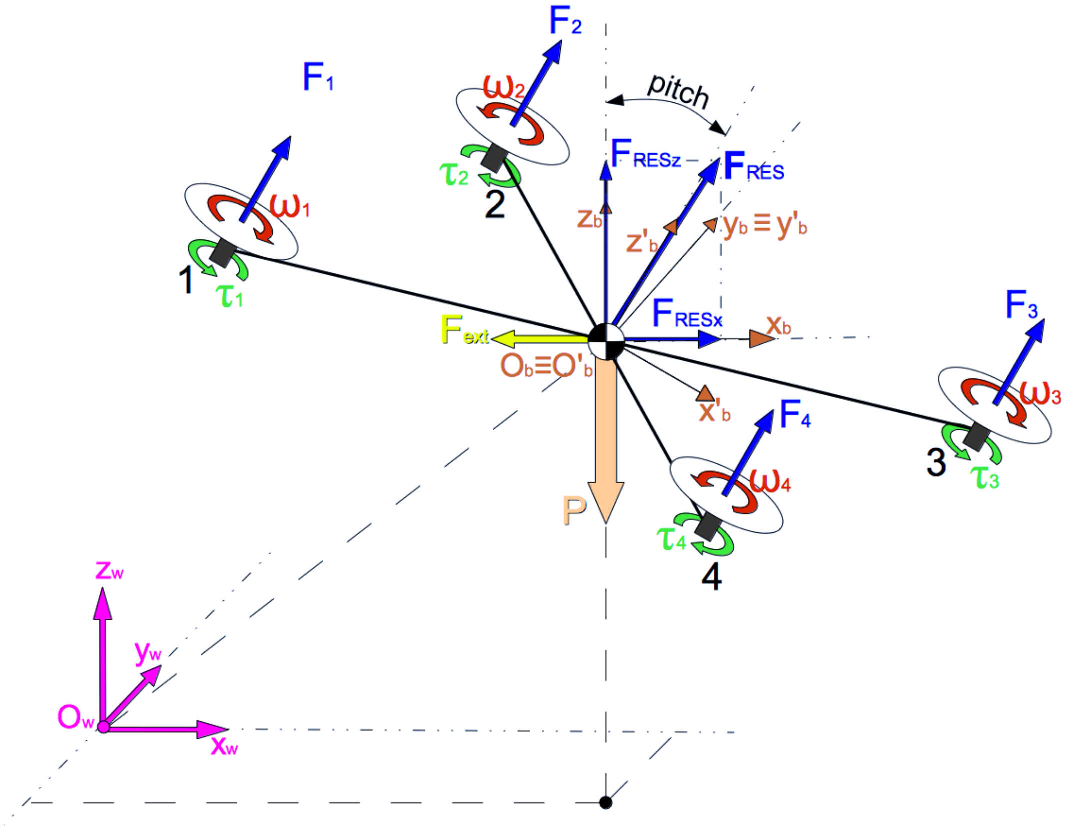


FIGURE 2.4: Representation of simulated translation movement to counteract an external force

Forces

In order to simplify simulations, it is considered only one external force parallel to the x_w-axis, but oriented in opposite direction (F_{ext} in Fig. 2.4). The result is represented in (2.7), where there are all external forces applied to the quad-rotor body.

$$F_{tot} = \begin{bmatrix} -F_{ext} \\ 0 \\ P \end{bmatrix} = \begin{bmatrix} -F_{ext} \\ 0 \\ -mg \end{bmatrix} \tag{2.7}$$

Where m is the overall mass of the quad-rotor and g gravity acceleration. A friction force F_D which behaves as viscous damping is considered within the model. It is represented by a three dimensional vector and it has the same damping coefficient k_d along all three inertial reference frame axes. It is exposed in (2.8).

$$F_D = \begin{bmatrix} -k_d\dot{x} \\ -k_d\dot{y} \\ -k_d\dot{z} \end{bmatrix} \tag{2.8}$$

As dumping forces, F_D is proportional to each velocity along each axis of inertial reference frame. Its negative sign is related to dissipative behavior of this force. To counteract these negative forces, the quad-rotor needs to exert a positive action through its propulsion units.

$$\vec{F_{RES}} = \sum_{i=1}^{4} \vec{F_i} = k \begin{bmatrix} 0 \\ 0 \\ \sum_{i=1}^{4} \omega_i^2 \end{bmatrix} \tag{2.9}$$

F_{RES} is obtained by summing all the single propellers thrust forces described in (2.1). Because of propulsion units are fixed to the quad-rotor body, this resultant force has components only along z_b-axis. To exert linear forces in an inertial frame, a fixed propellers quad-rotor must tilt its entire body following roll and pitch rotation. Hence, to refer F_{RES} to world reference frame, it must be multiplied by roll-pitch-yaw rotation matrix described in (2.6). Therefore $-F_{ext}$ and P are counterbalanced by F_{RESx} and F_{RESz} respectively, as shown in Fig. 2.4. The final Euler′s equation for linear dynamics is written in (2.10).

$$M\ddot{\vec{X}} = R(\psi, \theta, \phi)\vec{F_{RES}} + \vec{F_D} + \vec{F_{tot}} \tag{2.10}$$

Where $\ddot{\vec{X}}$ is the quad-rotor three-dimensional linear acceleration vector referred to inertial frame, and M is the mass matrix defined in (2.11).

$$M = \begin{bmatrix} m & 0 & 0 \\ 0 & m & 0 \\ 0 & 0 & m \end{bmatrix} \tag{2.11}$$

Torques

To properly analyze torques involved in quad-rotor dynamics, it is necessary to convert roll-pitch-yaw from body-referenced rotations to world-referenced rotations. The idea is to consider small changes in each of roll, pitch and yaw angles, and determine the effects. Recalling multiplication order in (2.6), the first angle undergoes two rotations, the second angle only one rotation, and the last angle no additional rotations. Thus (2.12) is obtained.

$$\vec{\Omega} = R(\psi)R(\theta) \begin{bmatrix} 0 \\ 0 \\ d\phi/dt \end{bmatrix} + R(\psi) \begin{bmatrix} 0 \\ d\theta/dt \\ 0 \end{bmatrix} + \begin{bmatrix} d\psi/dt \\ 0 \\ 0 \end{bmatrix} \tag{2.12}$$

Where $\vec{\Omega}$ expresses body angular velocities referred to the inertial frame. Solving (2.12) and simplifying equation form, it becomes (2.13).

$$\vec{\Omega} = \begin{bmatrix} 1 & 0 & \text{-}S_\theta \\ 0 & C_\phi & C_\theta S_\phi \\ 0 & \text{-}S_\phi & C_\theta C_\phi \end{bmatrix} \dot{\vec{\Theta}} \tag{2.13}$$

Where $\dot{\vec{\Theta}}$ represents roll, pitch and yaw velocities vector.
Each propulsion unit, as presented in 2.1.1, delivers a thrust force and a drag torque which produce effects on the overall torque on quad-rotor body.
As can be noticed in Fig. 2.1, all thrust forces exert a torque around x_b-axis. In particular, F_2 and F_3 produce a positive revolution opposed to F_1 and F_4, which apply a negative momentum. Considering thrust forces expression (2.1), torque around x_b-axis is (2.14).

$$\tau_x = L \cdot \cos\left(\frac{\pi}{4}\right) \cdot k(-\omega_1^2 + \omega_2^2 + \omega_3^2 - \omega_4^2) \tag{2.14}$$

Where L is distance between each propulsion unit and O_b. To obtain the distance between propulsion units and x_b-axis, it is sufficient to multiply L by $cos(\frac{\pi}{4})$. Same approach is used to determine torque around y_b-axis.

$$\tau_y = L \cdot \cos\left(\frac{\pi}{4}\right) \cdot k(+\omega_1^2 + \omega_2^2 - \omega_3^2 - \omega_4^2) \tag{2.15}$$

The only differences between (2.14) to (2.15) are forces directions. Around y_b-axis, F_1 and F_2 bring positive contribution while F_3 and F_4 turn in opposite direction. Momentum around z_b-axis is determined only by drag torques described in (2.2). Its expression is (2.16).

$$\tau_z = b \cdot (\omega_1^2 - \omega_2^2 + \omega_3^2 - \omega_4^2) \tag{2.16}$$

Thus, a single resultant vector can be defined for torques exerted on the quad-rotor body.

$$\vec{\tau}_{RES} = \begin{bmatrix} \tau_x \\ \tau_y \\ \tau_z \end{bmatrix} \tag{2.17}$$

At this point the final form of Euler's equation for rotational dynamics can be written.

$$I\dot{\vec{\Omega}} = \vec{\tau}_{RES} - \vec{\Omega} \times (I\vec{\Omega}) \tag{2.18}$$

Where $\dot{\vec{\Omega}}$ is quad-rotor body angular acceleration referred to the inertial frame; I is the matrix of moments of inertia which is defined in (2.19)

$$I = \begin{bmatrix} I_{\mathrm{xx}} & 0 & 0 \\ 0 & I_{\mathrm{yy}} & 0 \\ 0 & 0 & I_{\mathrm{zz}} \end{bmatrix} \tag{2.19}$$

In particular, $\mathrm{I_{xx}}$, $\mathrm{I_{yy}}$ and $\mathrm{I_{zz}}$ are quad-rotor body moments of inertia respectively around x_b, y_b and z_b-axis. Hence, the ultimate set of Euler's equations which models the fixed propellers quad-rotor is given by (2.10) and (2.18).

2.1.4 Control scheme and Simulink power calculation

A control system in a quad-rotor architecture aims at controlling its position and attitude. As shown in Fig. 2.5, the result is achieved performing two feedback loops. The innermost loop controls attitude, i.e. the roll, pitch and yaw reference following; the external loop controls quad-rotor position tracking following its reference.

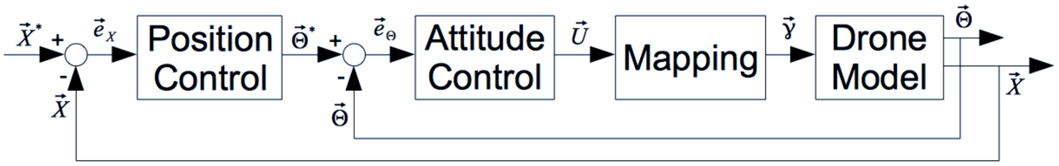


FIGURE 2.5: Common block diagram to control a quad-rotor

Attitude Control

In order to control the quad-rotor, it is necessary to identify input signals to the drone model. In this fixed propeller quad-rotor case, the four propulsion units angular speeds ω_i are considered as inputs. To define them clearly, a mapping block is necessary between the attitude controller and the drone model block (Fig. 2.5). Through the mapping block, inputs ω_i are defined as function of control laws u_i.
Control laws depend on PID controller which produces a regulated torque vector τ_{CTRL} defined in (2.20).

$$\tau_{\text{CTRL}} = \begin{bmatrix} \tau_\phi \\ \tau_\theta \\ \tau_\psi \end{bmatrix} = \begin{bmatrix} I_{\text{xx}}(K_p^{att} e_\phi + K_i^{att} \int_0^T e_\phi \, dt + K_d^{att} \dot{e}_\phi) \\ I_{\text{yy}}(K_p^{att} e_\theta + K_i^{att} \int_0^T e_\theta \, dt + K_d^{att} \dot{e}_\theta) \\ I_{\text{zz}}(K_p^{att} e_\psi + K_i^{att} \int_0^T e_\psi \, dt + K_d^{att} \dot{e}_\psi) \end{bmatrix} = \begin{bmatrix} I_{\text{xx}} u_\phi \\ I_{\text{yy}} u_\theta \\ I_{\text{zz}} u_\psi \end{bmatrix} \tag{2.20}$$

Where K_p^{att}, K_i^{att}, and K_d^{att} are control gains of PID attitude regulator; T is the overall simulation time; e_ϕ, e_θ and e_ψ are respectively error on roll (ϕ), pitch (θ) and yaw (ψ) defined in (2.21).

$$\begin{aligned} e_\phi &= \phi^* - \phi \\ e_\theta &= \theta^* - \theta \\ e_\psi &= \psi^* - \psi \end{aligned} \tag{2.21}$$

Where ϕ^*, θ^* and ψ^* are reference values of roll, pitch and yaw respectively. To map angular speeds from controlled torque, it is required to match τ_{RES} in (2.17) with τ_{CTRL} in (2.20) obtaining (2.22). For the sake of conciseness, we will intend the square of propulsion unit angular speeds as γ_{i}.

$$\begin{bmatrix} L \cdot \cos(\frac{\pi}{4}) \cdot k(-\gamma_1 + \gamma_2 + \gamma_3 - \gamma_4) \\ L \cdot \cos(\frac{\pi}{4}) \cdot k(\gamma_1 + \gamma_2 - \gamma_3 - \gamma_4) \\ b \cdot (\gamma_1 - \gamma_2 + \gamma_3 - \gamma_4) \end{bmatrix} = \begin{bmatrix} I_{\text{xx}} u_\phi \\ I_{\text{yy}} u_\theta \\ I_{\text{zz}} u_\psi \end{bmatrix} \tag{2.22}$$

It is a system with three equation in four unknowns. To make it solvable, it is necessary to introduce a fourth equation, where vertical components of overall thrust forces need to be equal to the quad-rotor weight P.

$$k(\gamma_1 + \gamma_2 + \gamma_3 + \gamma_4) \cdot \cos(\phi) \cdot \cos(\theta) = mg \tag{2.23}$$

Thus the system's final form is described in (2.24).

$$\begin{bmatrix} L \cdot \cos(\frac{\pi}{4}) \cdot k(-\gamma_1 + \gamma_2 + \gamma_3 - \gamma_4) \\ L \cdot \cos(\frac{\pi}{4}) \cdot k(\gamma_1 + \gamma_2 - \gamma_3 - \gamma_4) \\ b \cdot (\gamma_1 - \gamma_2 + \gamma_3 - \gamma_4) \\ \gamma_1 + \gamma_2 + \gamma_3 + \gamma_4 \end{bmatrix} = \begin{bmatrix} I_{\text{xx}} u_\phi \\ I_{\text{yy}} u_\theta \\ I_{\text{zz}} u_\psi \\ \frac{mg}{k \cdot \cos(\phi) \cdot \cos(\theta)} \end{bmatrix} \tag{2.24}$$

Extracting the $\vec{\gamma}$ vector the equation becomes (2.25).

$$\begin{bmatrix} -\frac{kL}{\sqrt{2}} & \frac{kL}{\sqrt{2}} & \frac{kL}{\sqrt{2}} & -\frac{kL}{\sqrt{2}} \\ \frac{kL}{\sqrt{2}} & \frac{kL}{\sqrt{2}} & -\frac{kL}{\sqrt{2}} & -\frac{kL}{\sqrt{2}} \\ b & -b & b & -b \\ 1 & 1 & 1 & 1 \end{bmatrix} \begin{bmatrix} \gamma_1 \\ \gamma_2 \\ \gamma_3 \\ \gamma_4 \end{bmatrix} = \begin{bmatrix} I_{\text{xx}} u_\phi \\ I_{\text{yy}} u_\theta \\ I_{\text{zz}} u_\psi \\ \frac{mg}{k \cdot \cos(\phi) \cdot \cos(\theta)} \end{bmatrix} \tag{2.25}$$

Hence, the controlled inputs $\vec{\gamma}$ given to the model are extracted in (2.26) considering substitutions expressed in (2.27) and (2.28).

$$\vec{\gamma} = A^{-1}\vec{U} \tag{2.26}$$

$$A = \begin{bmatrix} -\frac{kL}{\sqrt{2}} & \frac{kL}{\sqrt{2}} & \frac{kL}{\sqrt{2}} & -\frac{kL}{\sqrt{2}} \\ \frac{kL}{\sqrt{2}} & \frac{kL}{\sqrt{2}} & -\frac{kL}{\sqrt{2}} & -\frac{kL}{\sqrt{2}} \\ b & -b & b & -b \\ 1 & 1 & 1 & 1 \end{bmatrix} \tag{2.27}$$

$$\vec{U} = \begin{bmatrix} I_{\text{xx}} u_\phi \\ I_{\text{yy}} u_\theta \\ I_{\text{zz}} u_\psi \\ \frac{mg}{k \cdot \cos(\phi) \cdot \cos(\theta)} \end{bmatrix} \tag{2.28}$$

Position Control

Position control acts similarly to attitude control. It is a PID controller which defines position control laws as exposed in (2.29).

$$\begin{bmatrix} u_{\mathrm{x}} \\ u_{\mathrm{y}} \\ u_{\mathrm{z}} \end{bmatrix} = \begin{bmatrix} (K_p^{pos} e_x + K_i^{pos} \int_0^T e_x \, dt + K_d^{pos} \dot{e}_x) \\ (K_p^{pos} e_y + K_i^{pos} \int_0^T e_y \, dt + K_d^{pos} \dot{e}_y) \\ (K_p^{pos} e_z + K_i^{pos} \int_0^T e_z \, dt + K_d^{pos} \dot{e}_z) \end{bmatrix} \tag{2.29}$$

Where K_p^{pos}, K_i^{pos}, and K_d^{pos} are control gains of PID position regulator; T is the overall simulation time; e_x, e_y and e_z are respectively errors on x_w, y_w and z_w directions defined in (2.30).

$$\begin{aligned} e_x &= x^* - x \\ e_y &= y^* - y \\ e_z &= z^* - z \end{aligned} \tag{2.30}$$

Where x^*, y^* and z^* are reference values of x, y and z respectively.
In the fixed propeller case, position on x_w/y_w plane depends on roll and pitch. Consequently, in order to move the body in both horizontal directions, position control defines roll and pitch reference angles.
To simplify exposition, in this simulated case it is assumed the body moves mainly along a x_w-axis parallel direction, thus pitch is the most considered reference angle. A geometric approach, shown in Fig. 2.6, is formulated in order to define a reliable pitch reference angle.

Following used geometric approach, reference pitch angle is defined in (2.31).

$$\theta^* = \text{atan}\left(\frac{u_x}{F_{vert}^{fix}} \right) \tag{2.31}$$

Where F_{vert}^{fix} is a constant vertical fictitious force parallel to z_w-axis; u_x is the position control law along a direction parallel to x_w-axis. As for the reference pitch angle calculation, it is required a roll reference angle, in order to preserve overall stability. It is defined in (2.32).

$$\phi^* = \text{atan}\left(\frac{-u_y}{F_{vert}^{fix}} \right) \tag{2.32}$$

The position control law u_y is considered negative because signs of roll angle and y_b-axis direction are opposite. Thus, to move positively along y_b-axis, roll angle must be negative and viceversa.
The altitude control which regulates position along z_w-axis has no relation with roll, pitch and yaw; thus it doesn't act on attitude control. Its action is directly included in angular speeds mapping block. Recalling (2.23), a new fourth mapping equation can be written.

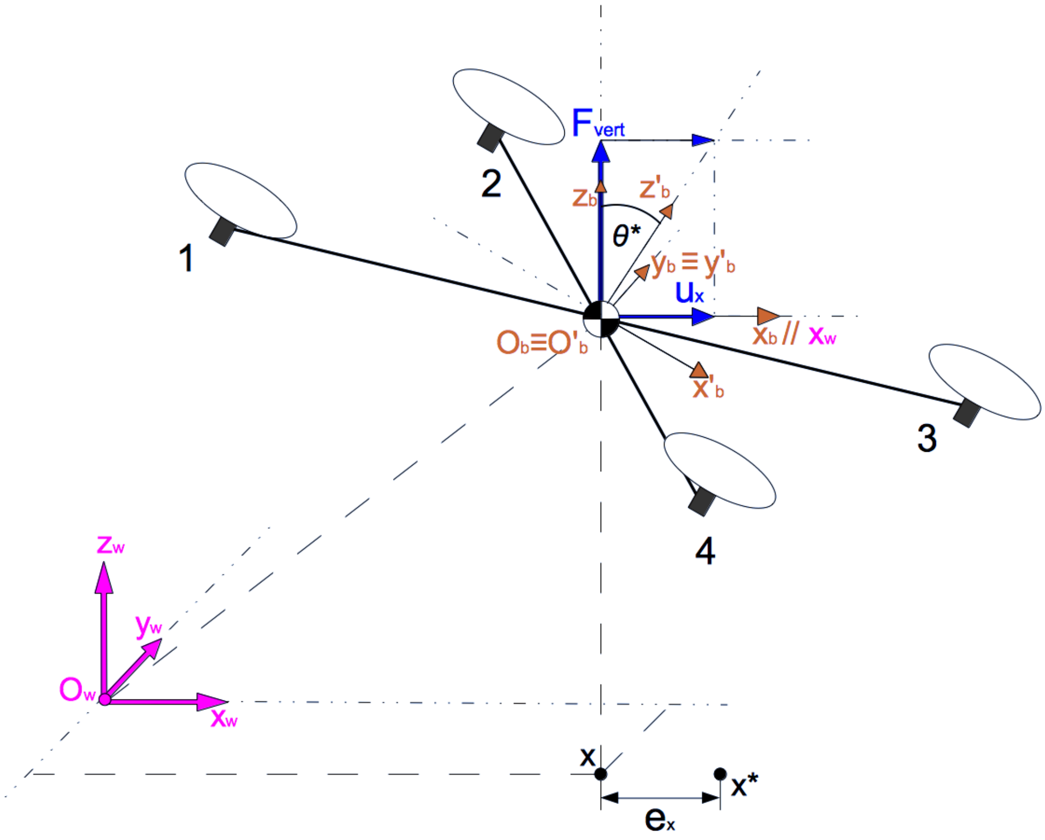


FIGURE 2.6: Geometric scheme to determine reference pitch angle θ^{*}

$$\sum_{i=1}^{4} \gamma_i = \frac{mg}{k \cdot \cos(\phi) \cdot \cos(\theta)} + u_z \tag{2.33}$$

Power Calculation

In order to evaluate the entire power consumption, it is important to focus on which are the main consumptive parts and areas of the drone. In the considered model, parts that absorb the largest amount of energy are the electric motors; hence aerodynamical losses and controllers electric consumption are neglected. Concerning calculation of instant power spent by motors, the mechanical power of each propulsion unit was considered and measured as shown in (2.34).

$$P_{fix} = \sum_{i=1}^{4} P_i = \sum_{i=1}^{4} \tau_i \omega_i = \sum_{i=1}^{4} b\,\gamma_i \omega_i = \sum_{i=1}^{4} b\omega_i^3 \tag{2.34}$$

Where τ_i and ω_i are i-th motor's drag torque (defined in (2.2)) and angular speed respectively. The given angular speeds are considered as mapping block output as shown in Fig. 2.7.

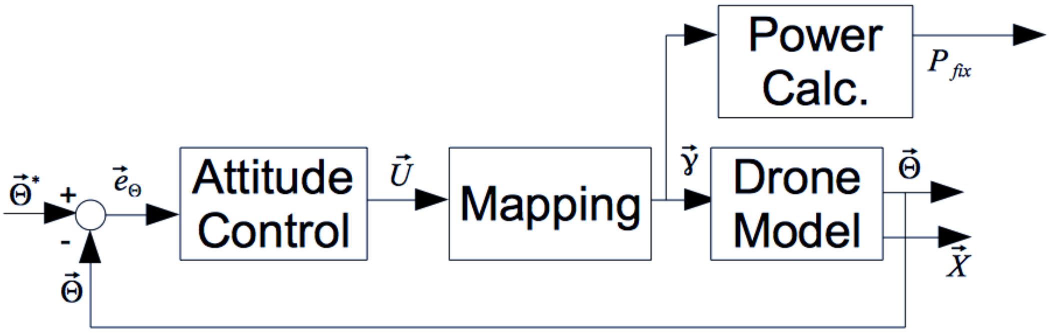


FIGURE 2.7: Power calculation block position

2.2 Tilting propellers structure

In this section it will be described the second type of quad-rotor architecture, in order to compare it in simulations with the fixed propellers structure previously described in section 2.1.
The presented tilting rotor architecture is obtained starting from the fixed propellers structure, allowing propulsion units number 1 and 2 to change direction of their z_i-axis, already shown in Fig. 2.2. Addition of tilting propellers increases the entire amount of reachable positions in space. In particular, introducing two over four tilting propellers, a difference between *under*-actuated and *full*-actuated architectures needs to be defined.

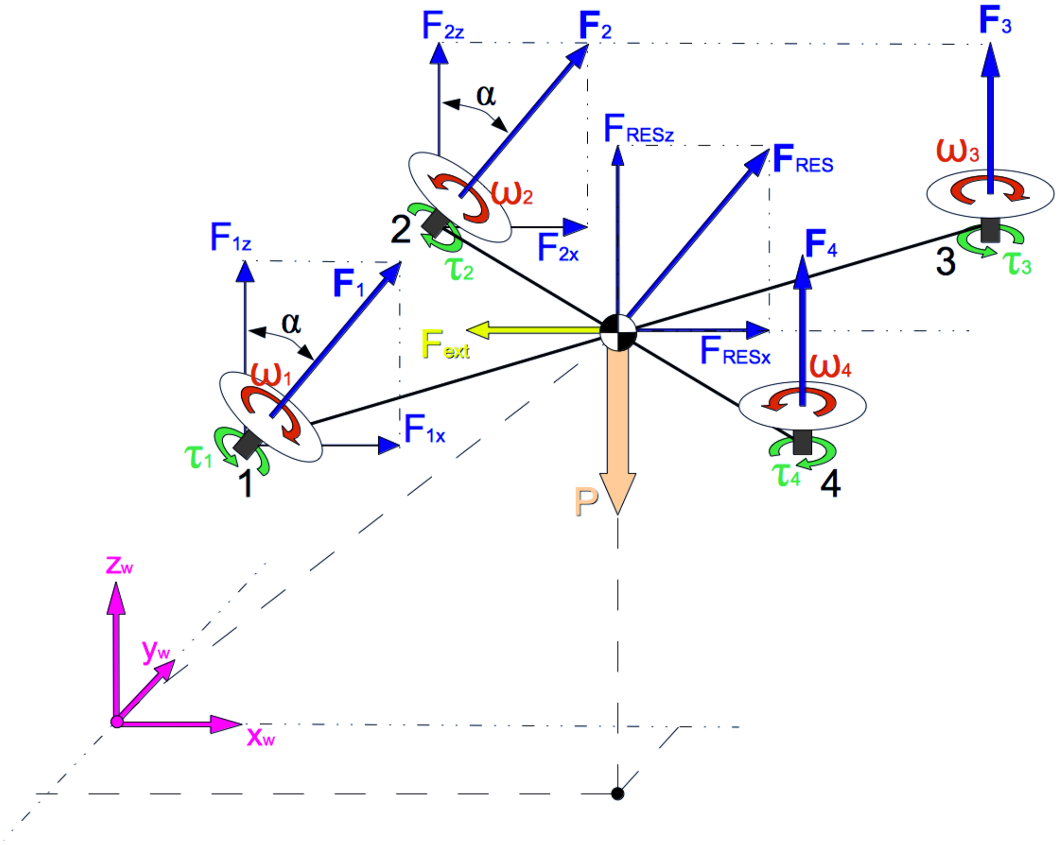


FIGURE 2.8: Representation of simulated translation movement along x_b-axis tilting rotors to counteract an external force F_{ext}

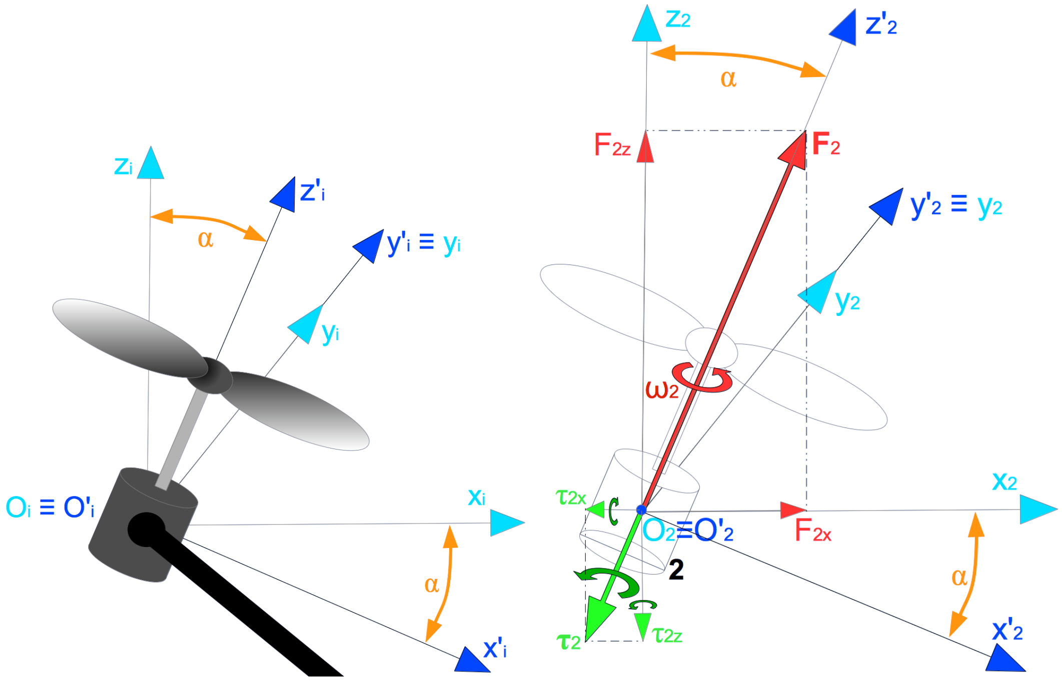
2.2.1 Under/Full actuated systems

Introducing tilting propellers, it is important to give proper definitions for *full*-actuated and *under*-actuated systems. This definitions pass through the concept of degrees of freedom. An *under*-actuated system is defined as a system that cannot be commanded to follow arbitrary trajectories in space. In other words the system has less actuators than degrees of freedom. In fixed propellers drone architectures (described in 2.1) there are 4 actuators (i.e. propulsion units) and 6 degrees of freedom (3 rotations and 3 translations in a three-dimensional space), thus they are classified as *under*-actuated systems. An effect of under-actuation in fixed propeller quad-rotors is the unavoidable binding between pitch angle and translation along x_b-axis.

Different from *under*-actuated systems, a *full*-actuated system has the same number of actuators and degrees of freedom. In simulated tilting rotors architecture it is introduced another actuator in order to tilt two over four propulsion units. Thus the overall architecture with 5 actuators and 6 degrees of freedom is still considered as *under*-actuated, but *full*-actuated along x_b-axis. In this configuration pitch angle is isolated from translation along x_b-axis.

2.2.2 Tilt angle

To represent the tilt angle α introduced in propulsion units number 1 and 2, a local reference frame is considered on each of them as shown in Fig. 2.9. The tilt angle α is defined as a rotation of this local frame around its own y_i-axis. The angle α is forced to be the same for unit 1 and 2, as represented in Fig. 2.8.



(A) Tilt angle α

(B) Forces and Torques

FIGURE 2.9: A representation of reference frames on a generic *i*-th tilting rotor (2.9a); forces and torques scheme on propeller number 2 (2.9b).

Being angle α a rotation around y-axis, it can be defined by a matrix similar to pitch rotation presented in (2.4). This new matrix which represents α rotation is defined in (2.35).

$$R(\alpha) = \begin{bmatrix} \cos(\alpha) & 0 & \sin(\alpha) \\ 0 & 1 & 0 \\ -\sin(\alpha) & 0 & \cos(\alpha) \end{bmatrix} \tag{2.35}$$

The tilt angle doesn't influence roll, pitch and yaw angles, which are still valid because they fully define the quad-rotor body orientation in space. But direction of forces and torques, related to units number 1 and 2, changes with α. This effect is analyzed in Euler's equations.

2.2.3 Euler's Equations

Dynamics of tilting rotors structure are determined by Euler's equations as in the fixed propellers case. The only difference is the introduction of rotation matrix $R(\alpha)$. To apply α rotation to forces and torques of motors number 1 and 2, they will be multiplied by $R(\alpha)$. Forces and torques dynamics represented in Fig. 2.10 will be analyzed separately in the following sections.

Forces

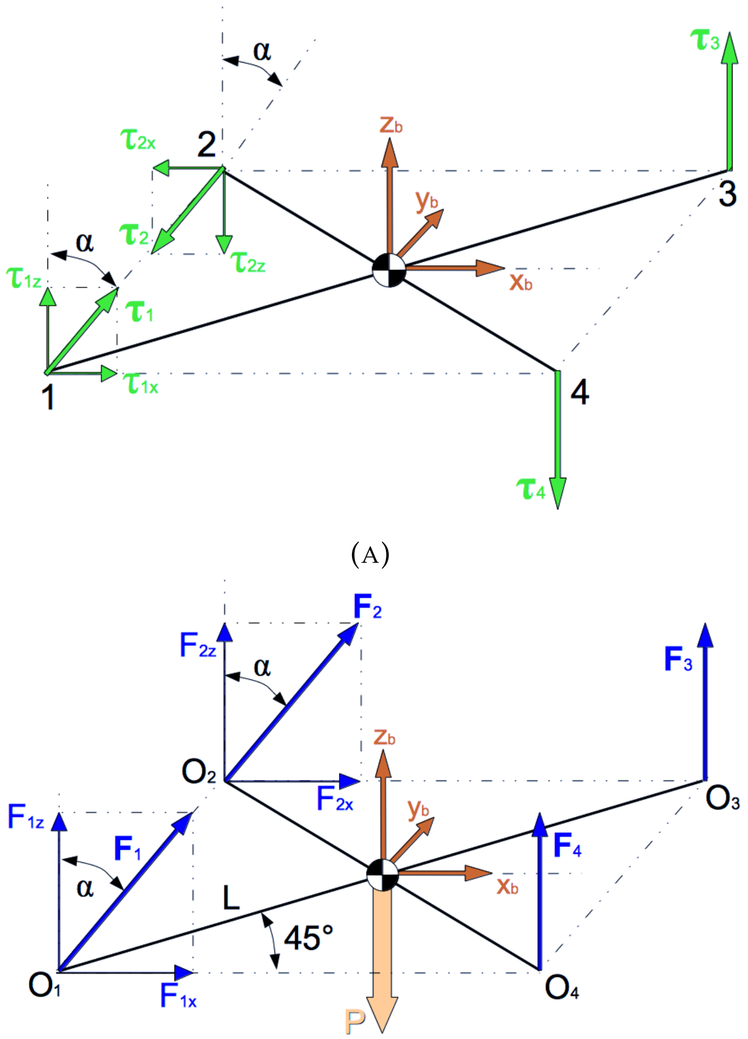
In forces dynamics, keeping (2.7) and (2.8) still valid, the first overall Euler's differential equation is still described in (2.10) but with a different $\vec{F_{RES}}$ defined in (2.36).

$$\vec{F_{RES}} = R(\alpha) \sum_{i=1}^{2} \vec{F_i} + \sum_{i=3}^{4} \vec{F_i} \tag{2.36}$$

Where rotation matrix *R(α)* is applied only to F_1 and F_2, in order to decompose both thrust forces in horizontal and vertical components.

Torques

There is a geometrical approach to define torques applied to drone body in tilting rotors case. Since some forces orientation changes referred with body fixed frame, each forces and drag torques will be analyzed related to their application point position. It is assumed that forces and torques' application points are placed in correspondence of propulsion units. Reminding that L represents the drone's arm length, application points positions can be written as in (2.37).



(A)

(B)

FIGURE 2.10: Representation of torques (2.10a) forces (2.10b) in tilting rotor case.

$$\begin{aligned}
O_1 &= \begin{bmatrix} -L \cdot \cos\frac{\pi}{4} & -L \cdot \sin\frac{\pi}{4} & 0 \end{bmatrix}^T \\
O_2 &= \begin{bmatrix} -L \cdot \cos\frac{\pi}{4} & +L \cdot \sin\frac{\pi}{4} & 0 \end{bmatrix}^T \\
O_3 &= \begin{bmatrix} +L \cdot \cos\frac{\pi}{4} & +L \cdot \sin\frac{\pi}{4} & 0 \end{bmatrix}^T \\
O_4 &= \begin{bmatrix} +L \cdot \cos\frac{\pi}{4} & -L \cdot \sin\frac{\pi}{4} & 0 \end{bmatrix}^T
\end{aligned} \tag{2.37}$$

Each drag torque can be represented in a vector form. Remembering (2.2), a generic *i*-th drag torque can be written as in (2.38).

$$\vec{\tau}_i = \begin{bmatrix} 0 \\ 0 \\ \pm b \cdot \omega_i^2 \end{bmatrix} \tag{2.38}$$

The sign is decided based on rotation direction of corresponding *i*-th propeller. Each propulsion units' thrust force is already defined in (2.9). To calculate momentum produced by each of thrust forces, it is applied a vector product between propulsion unit position O_i, related to body fixed frame O_b, and forces exerted by each propellers. Thus, total torque applied to a tilting rotors body can be calculated as in (2.40).

$$\tau_{\vec{RES}} = \sum_{i=1}^{2}[R(\alpha)\vec{\tau}_i + O_i \times R(\alpha)\vec{F}_i] + \sum_{i=3}^{4}[\vec{\tau}_i + O_i \times \vec{F}_i] \tag{2.39}$$

It is possible to obtain the second overall Euler's differential equation substituting the (2.40) in the (2.18), which remains still valid also for tilting propellers structure.

2.2.4 Control scheme and Simulink power calculation

As in section 2.1.4, the total control scheme in tilting rotors case is composed by two feedback loops. In Fig. 2.11 can be noticed how in this case the external position control loop is slightly different from previous. In the following sections it will be analyzed in details.

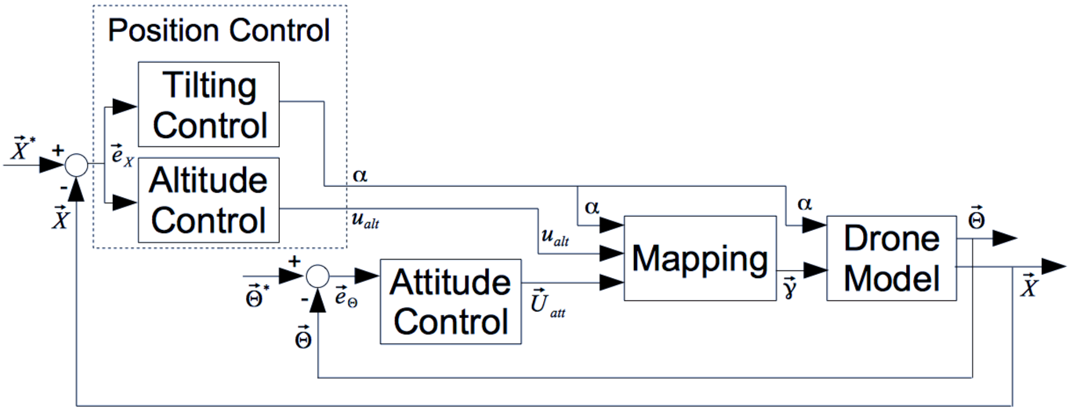


FIGURE 2.11: Total control scheme in tilting rotors case

Attitude Control

Introducing tilting propellers, pitch angle is no longer connected to lateral translation along x_b-axis. The attitude control is largely similar to the previous case. The attitude reference set externally, instead of being decided by position control, is a difference between the two cases. Even in this context, to follow attitude reference, a PID regulator is adopted. Because of this, being a general form of attitude control, (2.20) is still valid. In order to map regulated angular speeds, it is necessary to equal the controlled torques vector to new resultant torques vector obtained in (2.40).

$$\sum_{i=1}^{2}[R(\alpha)\vec{\tau}_i + O_i \times R(\alpha)\vec{F}_i] + \sum_{i=3}^{4}[\vec{\tau}_i + O_i \times \vec{F}_i] = \begin{bmatrix} I_{\mathrm{xx}}u_\phi \\ I_{\mathrm{yy}}u_\theta \\ I_{\mathrm{zz}}u_\psi \end{bmatrix} \tag{2.40}$$

The fourth equation is obtained following the same approach used before. With tilting propellers, the requisite to keep the drone floating, is barely different from (2.23). In addition to the vertical forces F_3 and F_4, it contains only vertical components of F_1 and F_2 as shown in (2.41).

$$k(\gamma_1 \cos(\alpha) + \gamma_2 \cos(\alpha) + \gamma_3 + \gamma_4) \cdot \cos(\phi) \cdot \cos(\theta) = mg \tag{2.41}$$

At this point, we can deduced that the mapping system to extract the propellers' controlled speeds is identical to (2.26), but using a different A matrix defined in (2.42).

$$A = \begin{bmatrix} b\sin(\alpha) - \frac{kL}{\sqrt{2}}\cos(\alpha) & -b\sin(\alpha) + \frac{kL}{\sqrt{2}}\cos(\alpha) & \frac{kL}{\sqrt{2}} & -\frac{kL}{\sqrt{2}} \\ \frac{kL}{\sqrt{2}}\cos(\alpha) & \frac{kL}{\sqrt{2}}\cos(\alpha) & -\frac{kL}{\sqrt{2}} & -\frac{kL}{\sqrt{2}} \\ b\cos(\alpha) + \frac{kL}{\sqrt{2}}\sin(\alpha) & -b\cos(\alpha) - \frac{kL}{\sqrt{2}}\sin(\alpha) & b & -b \\ \cos(\alpha) & \cos(\alpha) & 1 & 1 \end{bmatrix} \tag{2.42}$$

It can be noticed how effects of tilting rotors come out in the first two columns as sum of drag torques and thrust forces components. On the other hand, third and fourth columns are the same represented in (2.27).

Position Control

As figured in Fig. 2.11, position control is divided in two sections: altitude control and tilting angle control.

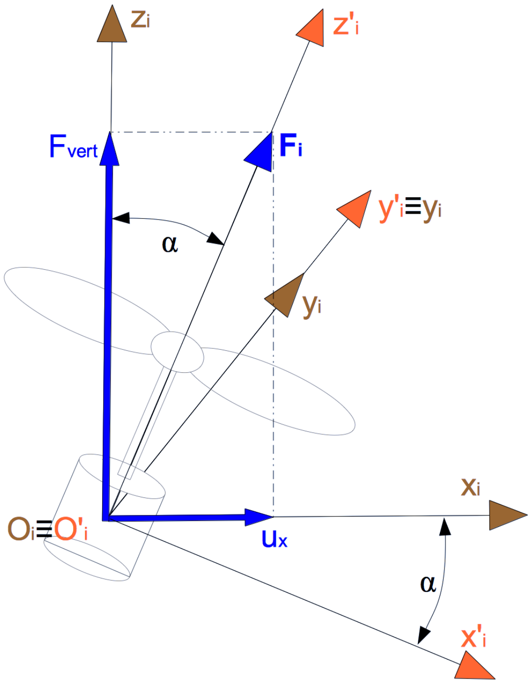


FIGURE 2.12: Geometric scheme to determine α

The altitude control consists in a PID regulator which determines the correct control law u_{alt} to be added in (2.41) obtaining equation (2.43).

$$\sum_{i=1}^{2} \gamma_i \cos(\alpha) + \sum_{i=3}^{4} \gamma_i = \frac{mg}{k \cdot \cos(\phi) \cdot \cos(\theta)} + u_{alt} \tag{2.43}$$

Where u_{alt} is a common PID control law defined in (2.44).

$$u_{alt} = K_p^{pos} e_z + K_i^{pos} \int_0^T e_z \, dt + K_d^{pos} \dot{e}_z \tag{2.44}$$

Where the same position error e_z defined in (2.30) is considered.
The tilting angle control regulates the angle α, which, as shown in Fig. 2.11, is handled in mapping block (eq. (2.41), (2.42)) and drone model block (eq. (2.36), (2.40)). For this control it is adopted a geometrical approach analogous to (2.31). The α angle represented in Fig. 2.12 is determined by (2.45).

$$\alpha = \text{atan}\left(\frac{u_x}{F_{vert}^{tilt}}\right) \tag{2.45}$$

Where u_x is defined even in this case by a PID controller. The tilt angle α is the same for both tilting propulsion units, thus only one controller is adopted and tuned.

Power and efficiency calculation

The hypothesis by which electric motors are the most consumptive parts, here is still valid. To calculate power spent instantly by tilting propellers structure, the same method shown in (2.34) is adopted.

$$P_{tilt} = \sum_{i=1}^{4} b\omega_i^3 \tag{2.46}$$

Block scheme is identical to previous, even in this case to calculate mechanical power, angular speeds are considered as mapping block output. Results from power blocks represented in Fig. 2.7 are compared in a power saving rate estimation block.

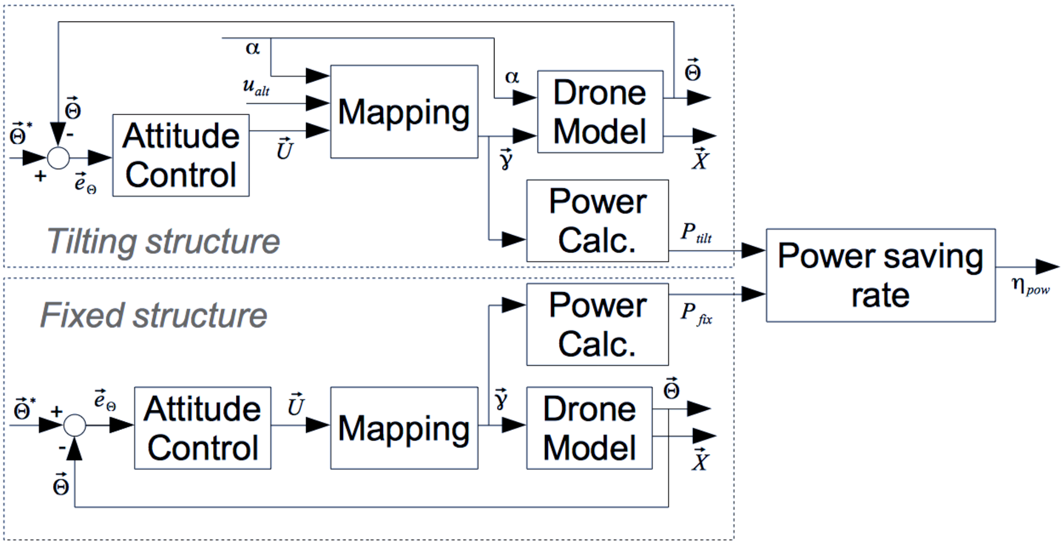


FIGURE 2.13: Power saving rate calculation scheme

In the power saving block shown in Fig. 2.13, an instant power saving rate η_{pow} is defined following (2.47).

$$\eta_{pow} = 1 - \frac{P_{fix}}{P_{tilt}} \tag{2.47}$$

Where P_{fix} is instant power consumption of fixed propellers structure, and P_{tilt} is instant power consumption of tilting propellers structure. Simulation results are analyzed in the next chapter.

Chapter 3

Simulation and results

Matlab/Simulink simulation results are discussed in this chapter focusing on both simulated architectures. Dynamics and power consumption will be compared in order to assert energy saving. To simulate both architecture, it is necessary to assign numerical values to parameters, as in (3.1).

$$\begin{aligned} L &= 0,225\,[\mathrm{m}];\\ m &= 0,55\,[\mathrm{Kg}];\\ I_{xx} &= 5\cdot 10^{-3}\,[\mathrm{Kg\,m^2}];\\ I_{yy} &= 5\cdot 10^{-3}\,[\mathrm{Kg\,m^2}];\\ I_{zz} &= 8,9\cdot 10^{-3}\,[\mathrm{Kg\,m^2}];\\ g &= 9,8\,[\mathrm{m/s^2}];\\ b &= 0,001;\\ k_d &= 0,1;\\ k &= 0,01; \end{aligned} \tag{3.1}$$

Physical parameters (as arm length L, mass m and moments of inertia I_{xx}, I_{yy}, I_{zz}) are defined by dimensions similar to middle range quad-rotor. These parameters have the same value in both simulations of tilting and fixed propellers structure.

Controllers' gains need to be declared too. They are all tuned by a trial-and-error method to obtain a reasonable behavior by the system. Tilting architecture controllers represented in Fig. 2.11 have gains listed in (3.2).

$$\begin{aligned} \text{Tilting angle } \alpha \text{ control}: \; K_p = 83000;\; K_i = 16000;\; K_d = 130000;\\ \text{Altitude control}: \; K_p = 100000;\; K_i = 32000;\; K_d = 130000;\\ \text{Attitude control}: \; K_p = 8500;\; K_i = 2000;\; K_d = 400; \end{aligned} \tag{3.2}$$

Gains of controller in fixed propeller case represented in Fig. 2.5 are defined in (3.3).

$$\begin{aligned}\text{Position control}: \; K_p = 3500000;\, K_i = 1650000;\, K_d = 550000;\\ \text{Attitude control}: \; K_p = 8500;\, K_i = 2000;\, K_d = 400;\end{aligned} \tag{3.3}$$

Recalling (2.31), (2.32) and (2.45) vertical components values are required. As controller's gains, vertical components are carried out by consecutive attempts until a reasonable performance is reached.

$$\begin{aligned}F_{vert}^{tilt} = 10000;\\ F_{vert}^{fix} = 2000000;\end{aligned} \tag{3.4}$$

3.1 Simulation Setting

To analyze simulation results, test settings must be declared. The simulation consists in applying an external force F_{ext} which changes its magnitude during the test. It is applied only along the x_b-axis, as figured in Fig. 2.4 and Fig. 2.8. It changes following a trapezoidal profile assuming values between 0 N to 100 N as represented in Fig. 3.1.

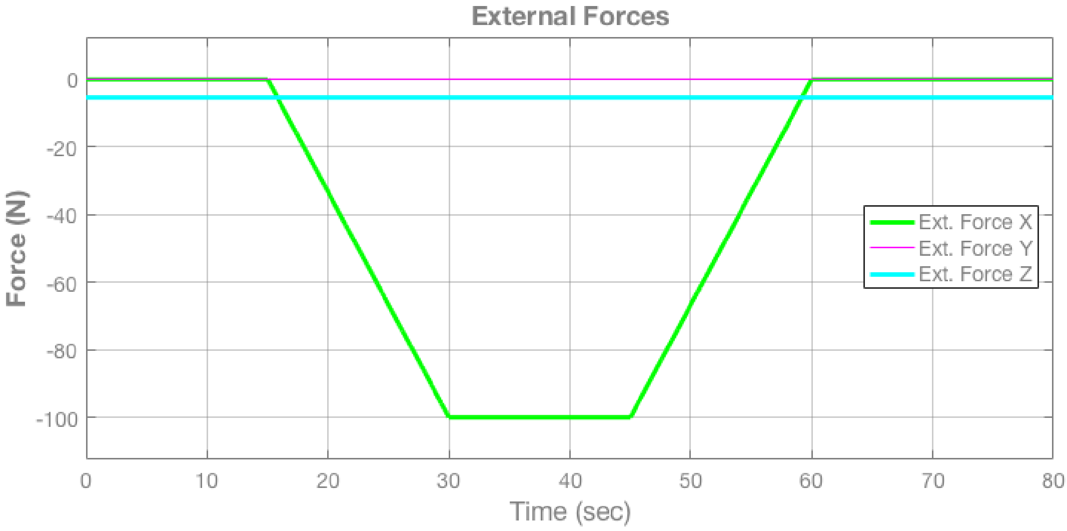


FIGURE 3.1: External forces shape during simulation.

As represented along the O_b, reference for position and attitude is set to zero in both cases.

$$\begin{aligned} \vec{X}^* &= \begin{bmatrix} 0 & 0 & 0 \end{bmatrix}^T \\ \vec{\Theta}^* &= \begin{bmatrix} 0 & 0 & 0 \end{bmatrix}^T \end{aligned} \tag{3.5}$$

It suggests that the body reference frame O_b is set in the same center of world reference frame O_w. This implies that the simulated drone is controlled to hold its position, staying in a floating condition, while the external force is applied.

3.2 Position and attitude results

In Fig. 3.2 results in terms of position, attitude and tilting angle α are represented for both the cases. In the first part of figure, under the action of external forces, both architectures drift $1,5\,m$ away from reference point on x_w-axis . This behavior occurs only when the external force is not constant. As soon as the external force becomes constant, the control action is able to bring quad-rotor back to the reference position. Furthermore, controllers are able to keep the reference position along y_w-axis and z_w-axis. Watching at dynamics of position, the only difference between tilting rotor's and fixed rotor's structures is velocity of drifting from reference point.
Concerning attitude behavior, in the central frame of Fig. 3.2, it can be noticed how in tilting rotor's simulation, the attitude is perfectly maintained on the reference point while angle α reaches the maximum amplitude of 88,6°.

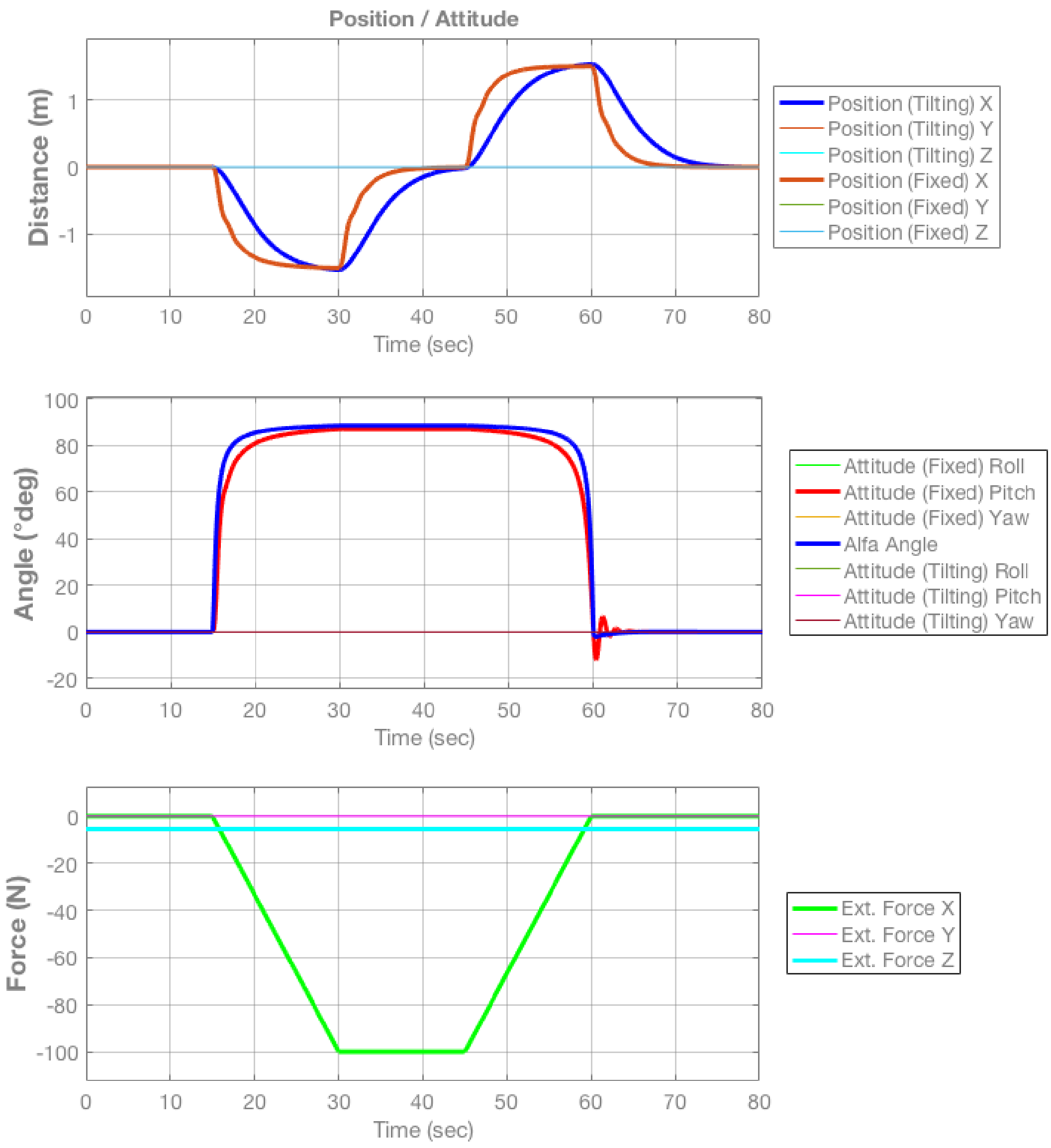


FIGURE 3.2: Position and attitude results during simulation.

As figured in Fig. 2.9, it is important to notice that the angle α is measured with respect to vertical z_b-axis. Differently, to counteract external forces' action in fixed rotor simulation, the control system lets the entire drone pitch around its y_b-axis. This is due to the under-actuated behavior of a fixed propellers structure, where pitch and lateral translation are linked together. In this case the maximum pitch angle reached is $86,7\,°$.

3.3 Power consumption and saving rate results

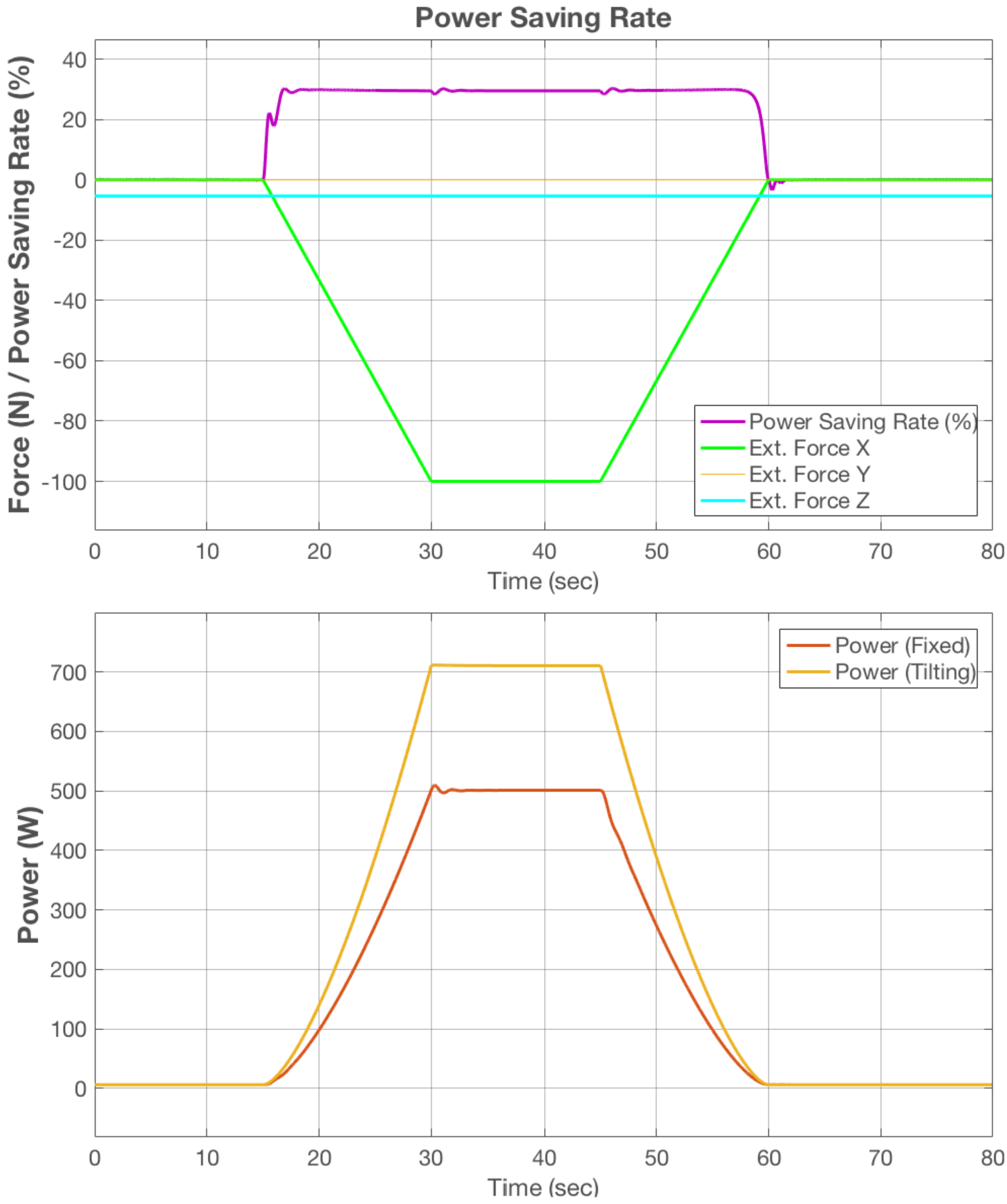


FIGURE 3.3: Representation of power saving rate during simulation with a comparison between powers instantly spent

In Fig. 3.3 it is represented the power instantly spent by both architectures during simulation. Plotted value represents the power spent by the entire structure, i.e by all four propulsion units. Both architectures spend the maximum amount of power during application of force F_{ext}, when it takes the maximum value of 100 N. The architecture of tilting rotors is clearly more

consumptive in this phase. Tilting rotors' solution reaches a maximum constant power spent of 711 W, compared with fixed propellers' structure which absorbs 501 W.
During the entire simulation, a power saving rate is measured in real time following (2.47). When the external force is applied, as in Fig. 3.3, the quad-rotor with fixed propellers saves 29, 7 % of energy compared to fixed rotor structure.

At this point it is possible to claim the overconsumption of the tilting solution. The only advantage which derives from using it, is a more stable attitude completely divided from lateral translation, that is a good attribute in aerial manipulation. But this utility is paid with about 30 % more power than the common quad-rotor structure. The overconsumption of tilting architecture can be easily translated in a shortest amount of flying time which is a bad feature for a drone with manipulating tasks.

Chapter 4

Linear Optimal Control Test

In this chapter a linear optimal control problem is set in order to evaluate cost function values in different cases. Both architectures are tested on this point of view. In the end results from analysis are exposed and discussed. An energy efficiency calculation will be proposed through an optimal control approach. Particular attention will be paid to the minimal possible control effort necessary to maintain the drone balanced on the axis′ origin. As presented in section 3.1 the experiment includes lateral disturbances. A fixed propeller architecture was simulated as a tilting propeller case, therefore hypothesizing a null α.
In order to apply this control algorithm, the system was downgraded from a six degrees of freedom to a three degrees of freedom model, becoming a planar system. In addition, a linearization was applied to obtain the optimal control algorithm. Considering this approach, energy consumption was measured in terms of mechanical power spent by motors in the non-linear model.

4.1 Model simplification and Linearization

The model presented in 2.2.3 was simplified to obtain planar dynamics and for easier calculations. This physical planar model will be analyzed below to obtain the subsequent non-linear model in the state-space representation. As this is a non-linear planar model, it develops three degrees of freedom.

The three dynamics equations expressed in (4.1) are directly derived from the representation of forces shown in Fig. 4.1.

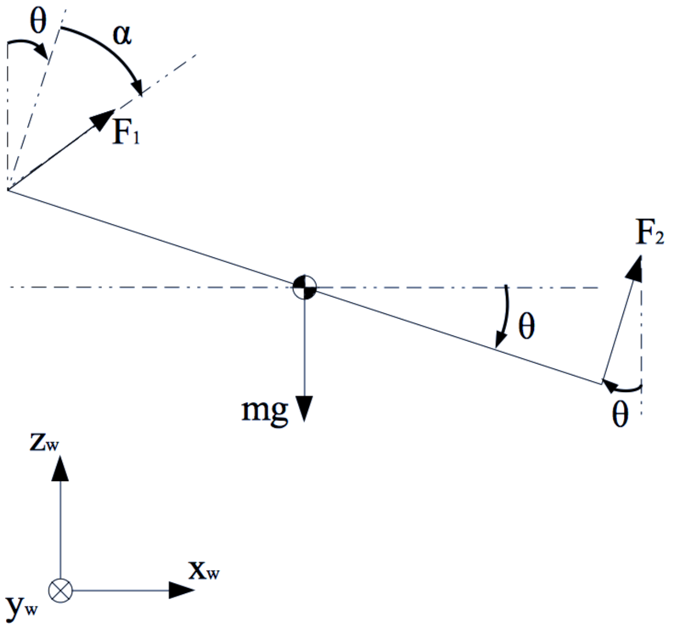


FIGURE 4.1: A simplified planar structure for a new drone model

$$
\begin{aligned}
m\ddot{x} &= F_1 \sin(\alpha + \theta) + F_2 \sin(\theta) - k_d \dot{x} \\
m\ddot{z} &= F_1 \cos(\alpha + \theta) + F_2 \cos(\theta) - k_d \dot{z} - mg \\
I_{yy}\ddot{\theta} &= L \cos\Big(\frac{\pi}{4}\Big) F_1 \cos(\alpha) - L \cos\Big(\frac{\pi}{4}\Big) F_2
\end{aligned}
\tag{4.1}
$$

Considering F_1 and F_2 as the forces produced by propellers along their own axis; and m, g, L, k_d, I_{yy} as the same parameters already described in (3.1). The non-linear model in the state-space representation can be deduced turning (4.1) into an ordinary differential system of first-order. To obtain this result, it is necessary to apply the substitutions described in (4.2), involving state variables and inputs.

$$
x(t) = \begin{bmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \\ x_4(t) \\ x_5(t) \\ x_6(t) \end{bmatrix} = \begin{bmatrix} x(t) \\ \dot{x}(t) \\ z(t) \\ \dot{z}(t) \\ \theta(t) \\ \dot{\theta}(t) \end{bmatrix} \qquad u(t) = \begin{bmatrix} u_1(t) \\ u_2(t) \\ u_3(t) \end{bmatrix} = \begin{bmatrix} F_1(t) \\ F_2(t) \\ \alpha(t) \end{bmatrix}
\tag{4.2}
$$

From this point onward, $x(t)$ will be considered as the state variables vector and not simply a translation along x-axis. Substituting the parameters taken from (3.1), the non-linear state-space representation can be expressed as (4.3).

$$\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= 1.8\,u_1 \sin(u_3 + x_5) + 1.8\,u_2 \sin(x_5) - 0.18\,\dot{x}_2 \\ \dot{x}_3 &= x_4 \\ \dot{x}_4 &= 1.8\,u_1 \cos(u_3 + x_5) + 1.8\,u_2 \cos(x_5) - 0.18\,\dot{x}_4 - 9.8 \\ \dot{x}_5 &= x_6 \\ \dot{x}_6 &= 31.8\,u_1 \cos(u_3) - 31.8\,u_2 \end{aligned} \tag{4.3}$$

The time dependency was omitted for clarity reasons but it remains valid. To obtain an akin linear model to the one described in (4.4), a first-order Taylor linearization process around an equilibrium point (x_e, u_e) is applied.

$$\dot{x}(t) = A\,x(t) + B\,u(t) \tag{4.4}$$

A and B are the Jacobian matrices of the state functions described in (4.3) as derivatives of $x(t)$ and $u(t)$ respectively. Thus, the equation (4.5) is valid.

$$A = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \cdots & \frac{\partial f_1}{\partial x_6} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_6}{\partial x_1} & \cdots & \frac{\partial f_6}{\partial x_6} \end{bmatrix}_{(x_e,u_e)} \qquad B = \begin{bmatrix} \frac{\partial f_1}{\partial u_1} & \cdots & \frac{\partial f_1}{\partial u_3} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_6}{\partial u_1} & \cdots & \frac{\partial f_6}{\partial u_3} \end{bmatrix}_{(x_e,u_e)} \tag{4.5}$$

The linearization equilibrium point (x_e, u_e) is defined in (4.6).

$$x_e = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \qquad u_e = \begin{bmatrix} \frac{m\,g}{2} \\ \frac{m\,g}{2} \\ 0 \end{bmatrix} \tag{4.6}$$

This equilibrium point was selected to have x_2, x_4, x_6 velocities all null. The coordinate x and z can take any value, null in this case. A null value was assigned as well to the α angle, represented as u_3, to obtain an equilibrium point valid for both fixed and tilting architecture. As a consequence, the pitch angle x_5 must be null, and the two remaining inputs, u_1 and u_2, must be sufficiently large as to counteract the overall weight. A fundamental choice

in the linearization process is the equilibrium point setting, because this is the point around which the system is locally asymptotically stable. The simulated disturbances and reference values must be assigned in line with this locality property, so as to stay in the system's attraction neighborhood.
The A and B matrices' numerical value was obtained by substituting the equilibrium point value in the Jacobian matrices. They are represented in (4.7).

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & -0.181 & 0 & 0 & 9.8 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & -0.181 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \qquad B = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 4.9 \\ 0 & 0 & 0 \\ 1.818 & 1.818 & 0 \\ 0 & 0 & 0 \\ 31.82 & -31.82 & 0 \end{bmatrix} \tag{4.7}$$

In order to even the constant disturbance, the linear system in (4.4) is reinforced by an additional term, as shown in (4.8).

$$\dot{x}(t) = A\,x(t) + B\,u(t) + D\,w \tag{4.8}$$

D is a matrix describing the way disturbances w influence the state equation $\dot{x}(t)$. As described in section 3.1, to consider the horizontal disturbances, a constant horizontal force was modeled. On the contrary, disturbances due to modeled mass uncertainties $\tilde{m}$ were considered as a vertical constant force. The D matrix and the disturbances vector w are defined in (4.9).

$$D = \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 0 \\ 0 & 1 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \qquad w = \begin{bmatrix} -F_x \\ -\tilde{m}g \end{bmatrix} \tag{4.9}$$

To remove the effects of constant disturbances acting on the system, making it more robust, it is necessary to introduce an integral action on the state $x(t)$. To produce this result, a C matrix must be defined and, according to its content, it will determine the state vector's component on which integral action is applied. Adding one more equation to the system (4.8), it becomes (4.10).

$$\begin{aligned} \dot{x}(t) &= A\,x(t) + B\,u(t) + D\,w \\ \dot{\eta}(t) &= C\,x(t) - x^* \end{aligned} \tag{4.10}$$

$x^*(t)$ is the reference point where the state $x(t)$ is steered despite the disturbance's action. Disturbances were applied on $\dot{x}_2$ and $\dot{x}_4$, so that the C matrix can be accurately described as (4.11).

$$C = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix} \tag{4.11}$$

The final expression of the linear system is (4.12)

$$\begin{aligned} \dot{x}(t) &= A\,x(t) + B\,u(t) + D\,w \\ \dot{\eta}(t) &= C\,e(t) \end{aligned} \tag{4.12}$$

Where $e(t)$ is the position error defined in (4.13).

$$e(t) = \begin{bmatrix} \tilde{x}_1(t) \\ \tilde{x}_3(t) \end{bmatrix} = \begin{bmatrix} x_1(t) - x_1^* \\ x_3(t) - x_3^* \end{bmatrix} \tag{4.13}$$

4.1.1 Extended Model

To define the optimal control law $u(t)$ for a system like (4.12), an extended form of the system is needed, therefore considering the latter as a single matrix equation system. To derive this extended system, new extended disturbances and state variable vectors are defined as in (4.14). Moreover, new system's matrices are obtained from the previous ones, as in (4.15).

$$x_{ext}(t) = \begin{bmatrix} x(t) \\ \eta(t) \end{bmatrix} \qquad w_{ext} = \begin{bmatrix} w \\ -x^* \end{bmatrix} \tag{4.14}$$

$$A_{ext} = \begin{bmatrix} A & \mathbf{0}_{6\times 2} \\ C & \mathbf{0}_{2\times 2} \end{bmatrix} \qquad B_{ext} = \begin{bmatrix} B \\ \mathbf{0}_{2\times 3} \end{bmatrix} \qquad D_{ext} = \begin{bmatrix} D & \mathbf{0}_{6\times 2} \\ \mathbf{0}_{2\times 2} & \boldsymbol{I}_{2\times 2} \end{bmatrix} \tag{4.15}$$

The matrix equation that characterizes the overall extended system is (4.16).

$$\dot{\tilde{x}}_{ext}(t) = A_{ext}\,\tilde{x}_{ext}(t) + B_{ext}\,\tilde{u}_{ext}(t) + D_{ext}\,w_{ext} \tag{4.16}$$

Where $\tilde{u}_{ext}(t)$ is the optimal control law described in (4.17), and representing the control law governing the error. It is driven by the K_{ext} matrix extracted through an optimal control algorithm that will be defined in the next section.

$$\tilde{u}_{ext}(t) = K_{ext}\,\tilde{x}_{ext}(t) \tag{4.17}$$

Thus, the final control law is indicated by the equation (4.18).

$$u_{ext}(t) = \tilde{u}_{ext}(t) + u^*_{ext} \tag{4.18}$$

u^*_{ext} depends on the equilibrium point (4.6) around which the linearization is centered.

4.2 Optimal control definitions and problem settings

Optimal control deals with the problem of finding a control law for a given system such that a certain optimality criterion is achieved. It is a set of differential equations describing the paths of the control variables that minimize the cost function, e.g what defines the optimality criterion. This optimality criterion is driven by a cost function that can be set depending on the framework. In this specific case, it is defined as in (4.19).

$$J = \int_{t_0}^{\infty} \tilde{x}^T_{ext}(t)\,Q\,\tilde{x}_{ext}(t) + \tilde{u}^T_{ext}(t)\,R\,\tilde{u}_{ext}(t)dt \tag{4.19}$$

Where Q and R matrices are positive-semidefinite and positive-definite respectively. This is a *infinite-horizon* case because the final time value is set at infinite. For this reason, Q and R matrices are also constant. These matrices are used to determine the different relations among all system's components, such as state variables and inputs, that influence its internal dynamics. To minimize the functional cost J, the control law $u(t)$ must be determined as in (4.17), meaning K_{ext} as in (4.20).

$$K_{ext} = -R^{-1}\,B^T_{ext}\,S_\infty \tag{4.20}$$

Where S_∞ is a solution for a continuous-time algebraic Riccati equation (4.21).

$$A^T_{ext}\,S + S\,A_{ext} - S\,B_{ext}\,R^{-1}\,B^T_{ext}\,S + Q = 0 \tag{4.21}$$

The K_{ext} matrix can be split in two sub-matrices as shown in (4.22).

$$K_{ext} = \begin{bmatrix} K_1 & K_2 \end{bmatrix} \tag{4.22}$$

Clarifying (4.17) control law in (4.23), it is possible to highlight that K_1 governs a $x(t)$ state dynamics, while K_2 governs the integral's dynamic on the position error $\eta(t)$, removing all disturbances.

$$u_{ext}(t) = K_1\,x(t) + K_2\,\eta(t) \tag{4.23}$$

Thanks to these hypothesis, the control law in (4.17) is the best possible law. According to this, the feedback matrix $(A_{ext} + B_{ext}K_{ext})$ is a stabilizing Hurwitz matrix, with all negative real part eigenvalues. In line with these features, the linearized extended system is asymptotically stable around the equilibrium point (x_e, u_e) shown in (4.6).

4.2.1 Simulation Settings

Both tilting and fixed rotor architectures' simulation is set in terms of optimal control. Each architecture has a different pair of Q and R matrices, therefore defining different cost functions. The elements of these matrices were tuned according to a precise dynamics expressed by each architecture. Thanks to this tuning, all the simulations developed the same dynamics. It was possible with this result it to better compare each architecture's energy consumption. In both cases, the Q matrix is squared and diagonal, with $(n + p)$ dimension, where n is the state vector $x(t)$ dimension, and p is the disturbances vector $\eta(t)$ dimension. The R matrix is also squared but its dimension is m, with as many inputs as $u(t)$.

In the tilting rotor case, these matrices are expressed in (4.24) and (4.25).

$$Q_\alpha = \boldsymbol{diag}[1,\ 1,\ 1,\ 1,\ 80'000,\ 0.005,\ 1,\ 50] \tag{4.24}$$

$$R_\alpha = \boldsymbol{diag}[1,\ 1,\ 100] \tag{4.25}$$

The values appearing on the matrices' diagonal Q_α e R_α influence the variables to which they are linked. The higher the value, the more the related state variable is penalized in the minimization process of the cost function described in the (4.19). Looking at the Q_α elements, and keeping in mind the state variables' order in (4.2) and (4.14), what can be noticed is how penalized the pitch movement is, when multiplied by a coefficient of 80'000. On the contrary, the coefficient influencing the pitch velocity is relatively small to guarantee a quick response against the disturbances along the x-axis. The last two Q_α diagonal elements are coefficients by which the integral actions along x-axis and z-axis are multiplied. Considering more intense disturbances along the x-axis, $Q_{7,7}$ is smaller than $Q_{8,8}$ in order to prevent the occurrence of an overshoot when reaching the set-point on x. In conclusion, to favor the α angle opening, $R_{3,3}$ is set to 100 as a compromise to obtain a sufficiently fast $\tilde{x}_1(t)$ dynamics without a too high overshoot on α dynamics.

In the fixed propeller architecture, the right approach to define a correct dynamics where the disturbances' effects are removed, is completely the opposite to the tilting case. According to this, Q_θ and R_θ matrices are described in (4.26) and (4.27).

$$Q_\theta = \boldsymbol{diag}[1,\ 1,\ 1,\ 1,\ 100,\ 0.005,\ 0.65,\ 50] \tag{4.26}$$

$$R_\theta = \boldsymbol{diag}[1,\ 1,\ 80'000] \tag{4.27}$$

What can be noticed is that the coefficients for x_5 pitch and the tilting angle u_3 are reversed. In this case, the integral coefficient on the x was slightly reduced to the value of 0.65 to obtain the closest possible value to $\tilde{x}_1$ in the previous case.

A third hybrid case was also analyzed. Its matrices are Q_{hyb} and R_{hyb}, defined in (4.28) and (4.29).

$$Q_{hyb} = \boldsymbol{diag}[1,\ 1,\ 1,\ 1,\ 100,\ 0.005,\ 0.5,\ 50] \tag{4.28}$$

$$R_{hyb} = \boldsymbol{diag}[1,\ 1,\ 100] \tag{4.29}$$

Here, the pitch and α dynamics are equally weighted to allow the optimal control algorithm to use both in order to counter the disturbances. The integral coefficient on the x was further reduced to the value of 0.5 to obtain a dynamics as close as possible to all the previous ones.
The same w_{ext} is adopted in each simulation.

$$w_{ext} = \begin{bmatrix} -10 \\ -0.001 \\ 0 \\ 0 \end{bmatrix} \tag{4.30}$$

In (4.30) can be noticed that the disturbance force along x-axis has a value of $-10\,N$. This model was adopted for being relatively precise despite uncertainties on the mass value considering a disturbance of $-0.001\,N$ on the z-axis. In conclusion, the target position on the (x,z) plane is set to the axis origin $(0,0)$.

4.3 Results

The domains in which results were obtained are the following: the first is a linear domain where a cost function value and a matrix K_{ext} were defined after a linearization process and an optimal control problem setting. A different cost function value and K_{ext} matrix were reached for each simulated architecture. The second domain is a non-linear environment in which the previously obtained K_{ext} matrices are applied to the system. The resulting dynamics are then read and commented. Moreover, in this domain, the mechanical power delivered by motors is determined according to inputs u_1 and u_2.

4.3.1 Linear system

All possible scenarios in the linear domain were simulated: in particular, they are a tilting rotor structure, a fixed rotor structure, and a hybrid one in between the previous two.

First is the tilting rotor case, in which a Riccati equation is resolved, given Q_α and R_α, exposed in (4.24) and (4.25). Thus, following (4.20), an optimal gain matrix is defined as in (4.31).

$$K_\alpha = \begin{bmatrix} -0.1 & -0.1 & -4.1 & -1.6 & -200.0 & -2.5 & -0.0 & -5.0 \\ 0.1 & 0.1 & -4.1 & -1.6 & 200.0 & 2.5 & 0.0 & -5.0 \\ -0.2 & -0.3 & -0.0 & -0.0 & -0.0 & -0.0 & -0.0 & -0.0 \end{bmatrix} \tag{4.31}$$

All K_α eigenvalues have a negative real part, therefore producing a stabilizing action on the system.

$$\lambda_\alpha = \begin{bmatrix} (-79.7903 \pm 79.7586i)^2 \\ (-0.4969 \pm 0.6316i)^2 \\ -0.7605 + 0.0000i \\ (-1.4862 \pm 1.9156i)^2 \\ -3.0929 + 0.0000i \end{bmatrix} \tag{4.32}$$

The functional cost is determined as in (4.19) after substituting Q_α and R_α. To obtain a finite functional cost (4.33), the final time value must be different from infinite, but it must be wide enough to complete the transient dynamics. For this reason it is set to 30 seconds.

$$J_\alpha = 11245.52 \tag{4.33}$$

The same method was applied to find the optimal gains' matrices for the other two architectures. In the fixed propeller case gains matrix was defined as in (4.34) and its eigenvalues are exposed in (4.35).

$$K_\theta = \begin{bmatrix} -1.5 & -1.6 & -4.1 & -1.6 & -8.1 & -0.5 & -0.5 & -5.0 \\ 1.5 & 1.6 & -4.1 & -1.6 & 8.1 & 0.5 & 0.5 & -5.0 \\ -0.0 & -0.0 & 0.0 & -0.0 & -0.0 & -0.0 & -0.0 & 0.0 \end{bmatrix} \tag{4.34}$$

$$\lambda_\theta = \begin{bmatrix} (-15.0760 \pm 14.9233i)^2 \\ (-0.7061 \pm 0.6275i)^2 \\ -0.8855 + 0.0000i \\ (-1.4862 \pm 1.9156i)^2 \\ -3.0929 + 0.0000i \end{bmatrix} \tag{4.35}$$

The same method used to determine J_α was applied to measure the functional cost for a fixed propeller architecture dynamics. The time interval considered was between 0 and 30 seconds.

$$J_\theta = 3940.86 \tag{4.36}$$

The hybrid case is discussed below. In this architecture, both angles, θ and α, have the same penalizing coefficient as shown in (4.28) and (4.29). The same method used in the two previous cases is applied here. The gain matrix K_{hyb}, its eigenvalues λ_{hyb} and the functional cost J_{hyb} are expressed in (4.37), (4.38) and (4.39) respectively.

$$K_{hyb} = \begin{bmatrix} -1.2 & -1.2 & -4.1 & -1.6 & -7.8 & -0.5 & -0.4 & -5.0 \\ 1.2 & 1.2 & -4.1 & -1.6 & 7.8 & 0.5 & 0.4 & -5.0 \\ -0.1 & -0.1 & -0.0 & -0.0 & -0.0 & -0.0 & -0.0 & -0.0 \end{bmatrix} \tag{4.37}$$

$$\lambda_{hyb} = \begin{bmatrix} (-15.0760 \pm 14.9233i)^2 \\ (-0.7640 \pm 0.5727i)^2 \\ -0.8498 + 0.0000i \\ (-1.4862 \pm 1.9156i)^2 \\ -3.0929 + 0.0000i \end{bmatrix} \tag{4.38}$$

$$J_{hyb} = 3242.79 \tag{4.39}$$

Particular attention was paid to the final functional cost J in all three simulated cases. It is possible to notice that the functional cost J_{hyb} is really low compared to all the other costs. This means that largely different control costs are generated by penalizing or easing θ and α angles. In order to turn this result in an explicit energy-saving effect, non-linear dynamics needed to be analyzed when different K_{ext} matrices are applied.

4.3.2 Non-Linear system

In this subsection, what will be discussed are the results derived from simulations in a Matlab/Simulink environment of the non-linear system described in (4.3), revised with the integral action exposed in (4.10). The K_{ext} matrices previously calculated, expressed in (4.31), (4.34) and (4.37), were substituted in the non-linear model in three consecutive simulations. As can be noticed in Fig. 4.2, the control law $\tilde{u}_{ext}$ was added to the term u^* following the (4.18). The total simulation time in this case was 10 seconds. This is a sufficient time to complete the transient dynamics despite disturbances' action.

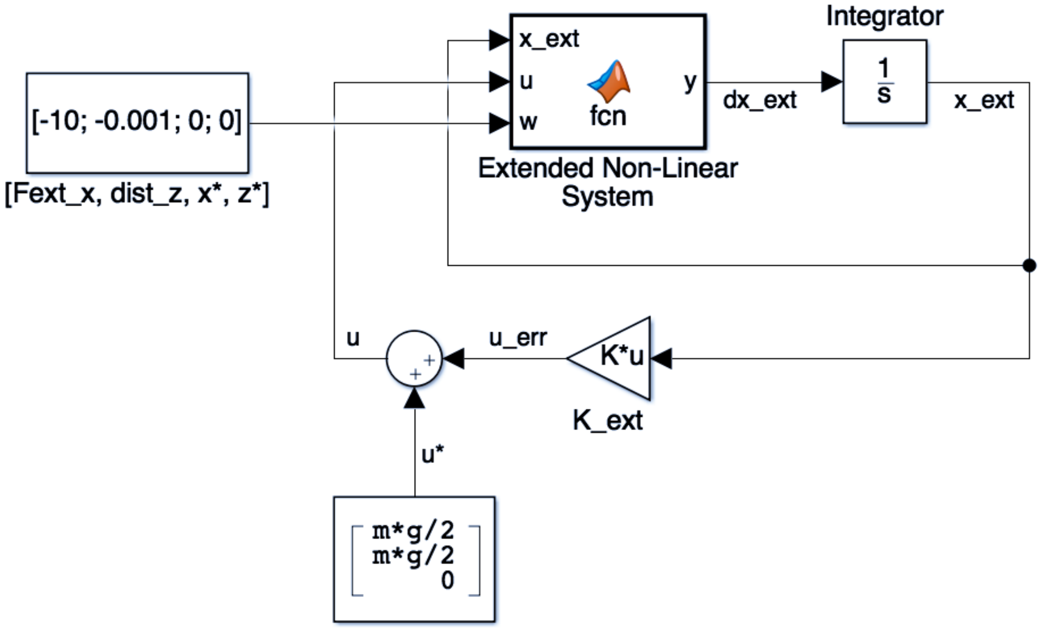


FIGURE 4.2: Scheme of a Simulink model for non-linear system

In the following paragraphs, the dynamics expressed by each of the simulated architectures are presented. The results are grouped according to different architectures' simulations. For each simulation, the x and z translational dynamics, the α and θ evolution, and the total power consumption, are discussed.

Tilting propeller structure

Fig. 4.3 shows the tilting rotor structure's translational dynamics under the action of disturbances. At time instant t_0, disturbances w were applied to the drone which started drifting from the origin where it was initialized. The widest displacement achieved along x-axis was around -2.74 m, while along the z-axis it was -42 cm. From this point on, the integral action effect is distinguishable. The control action was tuned trying to reach a trade-off among overshoot presence, severe evolution dynamics, and reasonable settling time, which can be considered around 10 seconds.

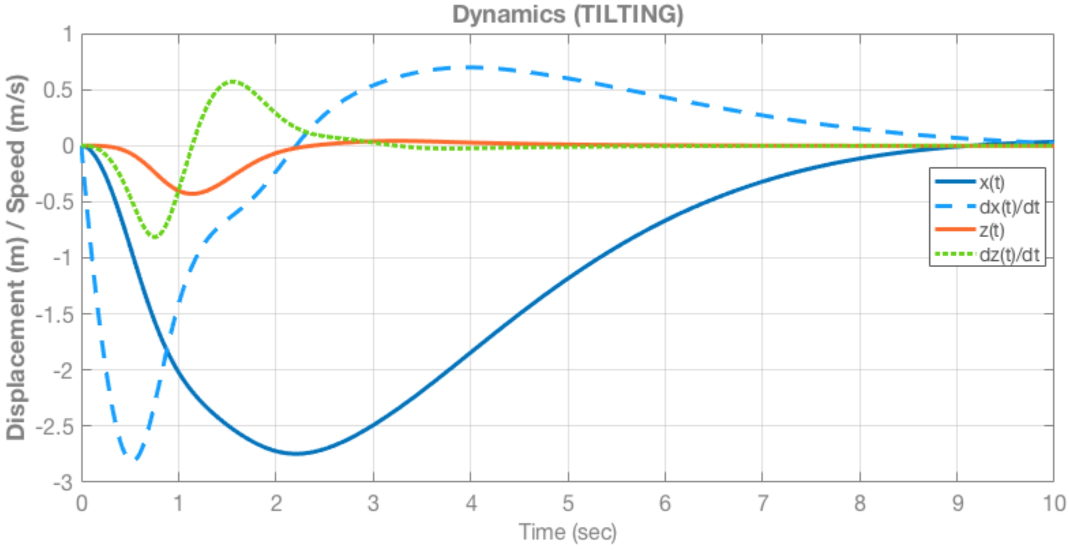


FIGURE 4.3: Translational dynamics in tilting rotor structure

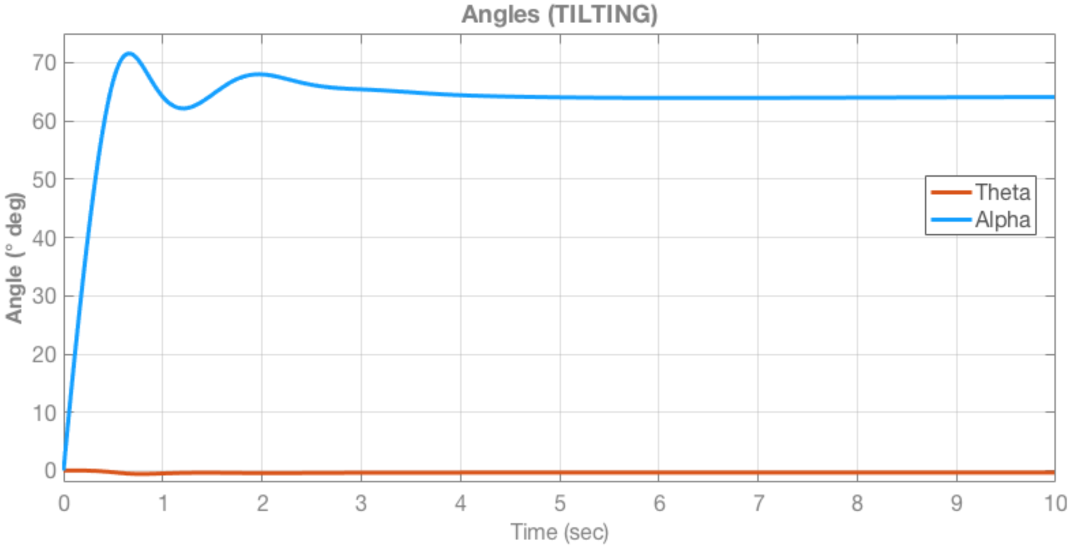


FIGURE 4.4: Angles' dynamics in tilting rotor structure

The dynamics of both angles involved, α and θ, is shown in Fig. 4.4. It is possibile to notice that α settled on a value of $+64.07\,^\circ$, after a maximum value of $+70.76\,^\circ$ was reached. On the contrary, the pitch angle θ stabilized on a value of $-0.34\,^\circ$ after reaching a $-0.56\,^\circ$ peak. This equilibrium point was possible because this tilting rotor architecture was full-actuated along x direction. This means that pitch and translation along x-axis are not linked together. Looking at the configuration in Fig. 4.1, to obtain any admissible equilibrium point, the couple (α , θ) must fulfill the equation in (4.40).

$$2\cos(\alpha)\sin(\theta) + \cos(\theta)\sin(\alpha) = 0 \tag{4.40}$$

This latter equation was obtained resolving the system in (4.3) in an equilibrium condition, i.e. considering null derivatives.
The last plot printed related to this architecture inspects the energy consumption based on the optimal inputs $u1$ and $u2$. As presented in (4.2), these are forces from which it is possible to extract the angular velocity ω following (2.1). Thus, knowing angular velocity, it is possible to calculate mechanical power consumption following the same approach as in (2.46). Taking into consideration the modulus of power consumption, results are printed in Fig. 4.5.

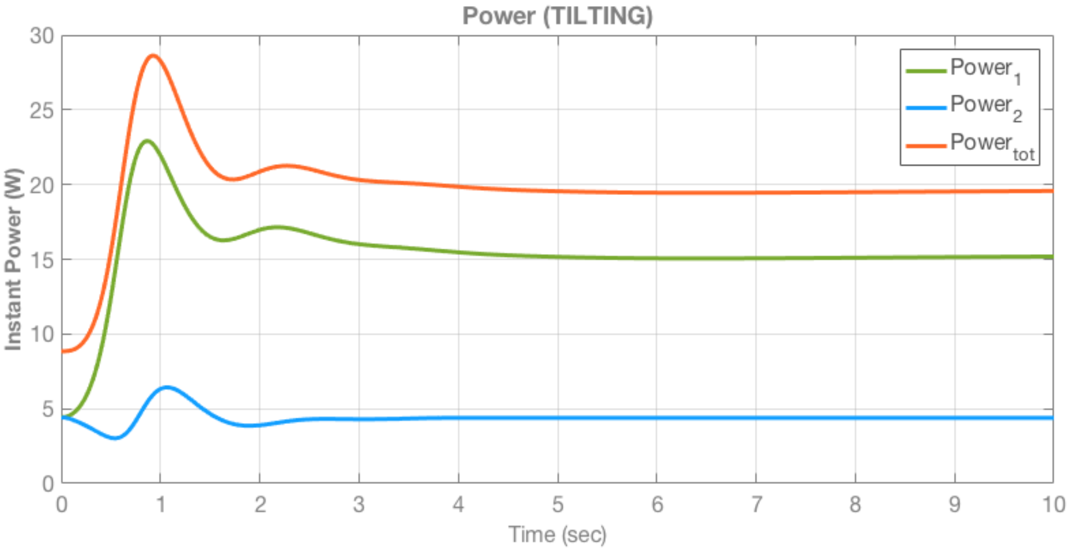


FIGURE 4.5: Instant power consumption evolution in tilting rotor structure

It is evident how tilting motor number 1 consumes 15.16 W in steady condition, as opposed to fixed motor number 2 which needs 4.3 W. In steady condition, they both develop an instant power of 19.55W after a peak of 28.54W.

Fixed propeller structure

An analogous approach can be used to describe the fixed propellers architecture. Fig. 4.6 highlights the similarities between fixed translational dynamics and tilting rotors' ones. As presented in section 4.2.1, this is due to the tuning criterion for optimal Q and R matrices. They were tuned trying to obtain a dynamics as coincident as possible for all the architectures. Because of this, displacements are very close to the previous case. Along the x-axis the maximum displacement was $-2.94\,m$, while on the z was $-33\,cm$. However, a striking difference between these two cases is that in the fixed rotor case dynamics are much more oscillating than in the previously analyzed tilting rotor case.

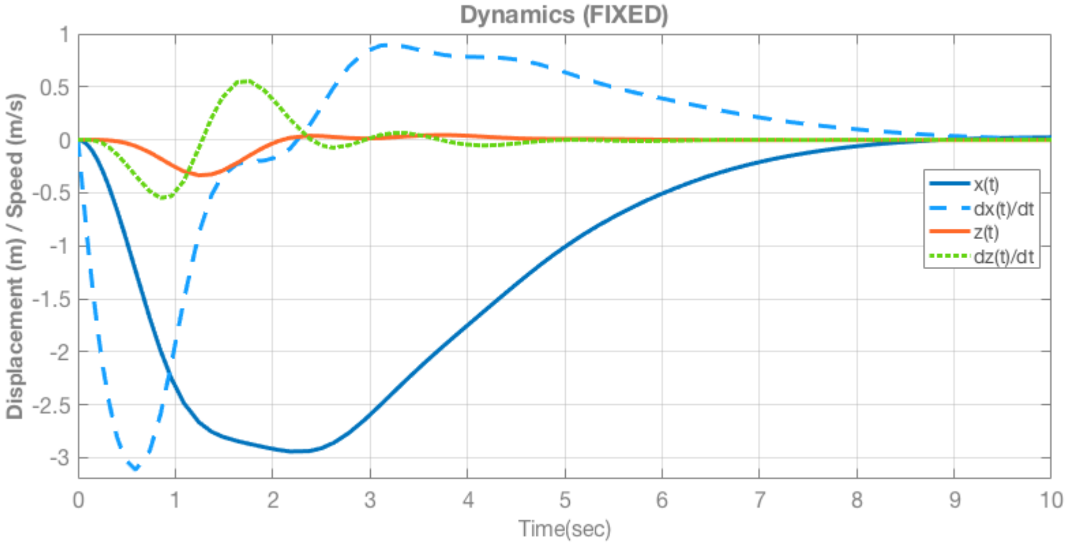


FIGURE 4.6: Translational dynamics in fixed structure

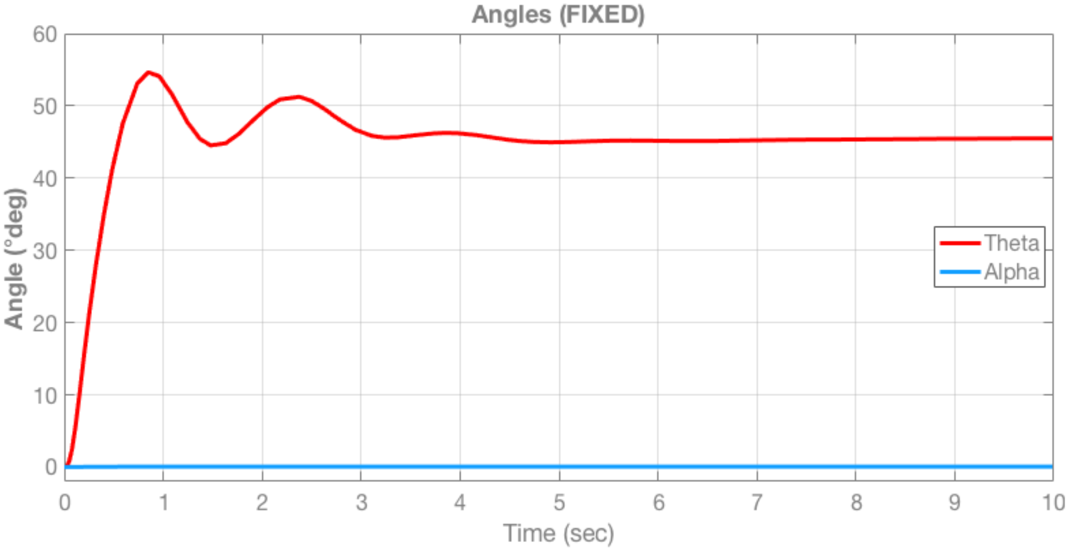


FIGURE 4.7: Angles' evolution in fixed rotor structure

The tilting and pitch angles evolution during elapsing time is presented in Fig. 4.7. The penalization on the α angle opening was rather effective. An evidence of this can be found on the α value steadied at 0.06 °. On the contrary, pitch angle θ reached a steady value of 45.42 °, after passing a peak of 54.65 °. Analogously to the previous power analysis, Fig. 4.8 presents the evolution of instant power consumption for each motor involved.

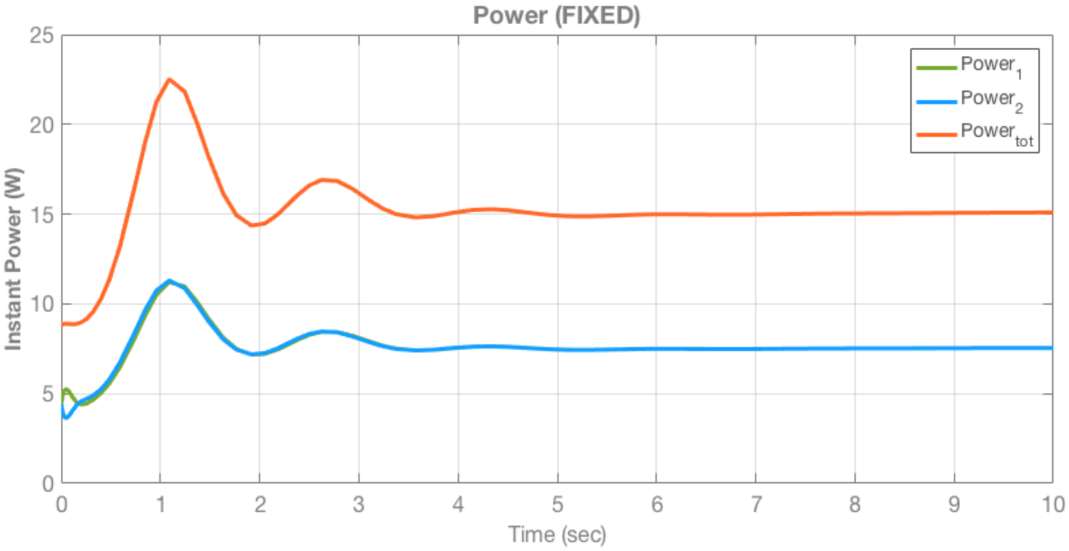


FIGURE 4.8: Instant power consumption evolution in fixed rotor structure

To counteract disturbances' effects, the two fixed motors tend to consume the same amount of energy. Overall, the instant power consumed in steady condition was 15.08 W, reaching an overshoot of 22.52 W. This result was achieved adding both powers, which at steady condition have a value of 7.53 W each.

Hybrid Tilting/Fixed propeller structure

The last simulated scenario was the hybrid case. Angles α and θ were equally penalized by coefficients of the same value in Q_{hyb} and R_{hyb} matrices. In this case, the optimal control algorithm was used to set the best values for the angles in order to minimize the cost function J_{hyb} in (4.39). Translational dynamics along vertical and horizontal axes are shown in Fig. 4.9. The maximum displacement reached $-2.86\,m$ on x-axis and $-33\,cm$ along z-axis.

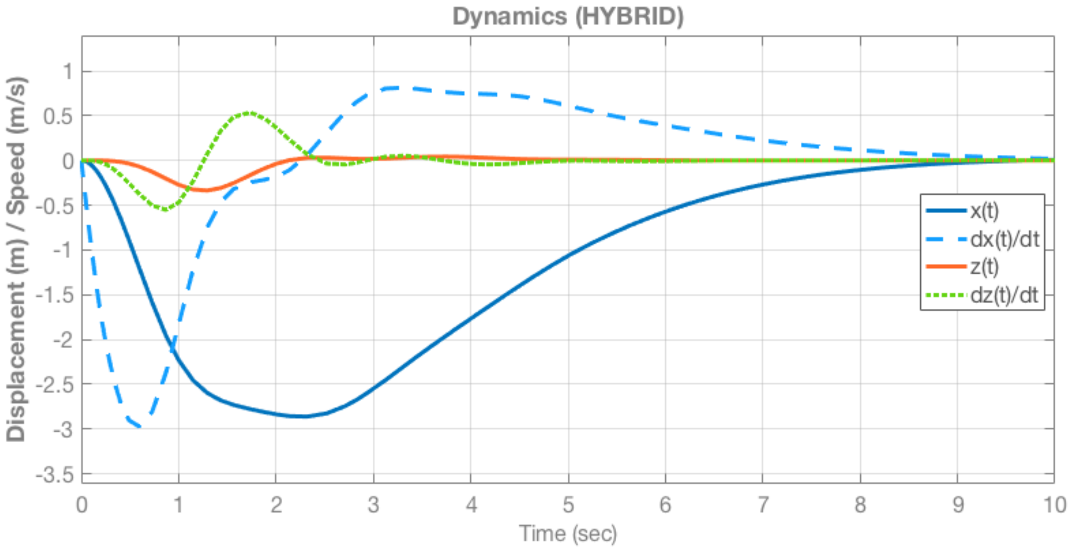


FIGURE 4.9: Translational dynamics in hybrid structure

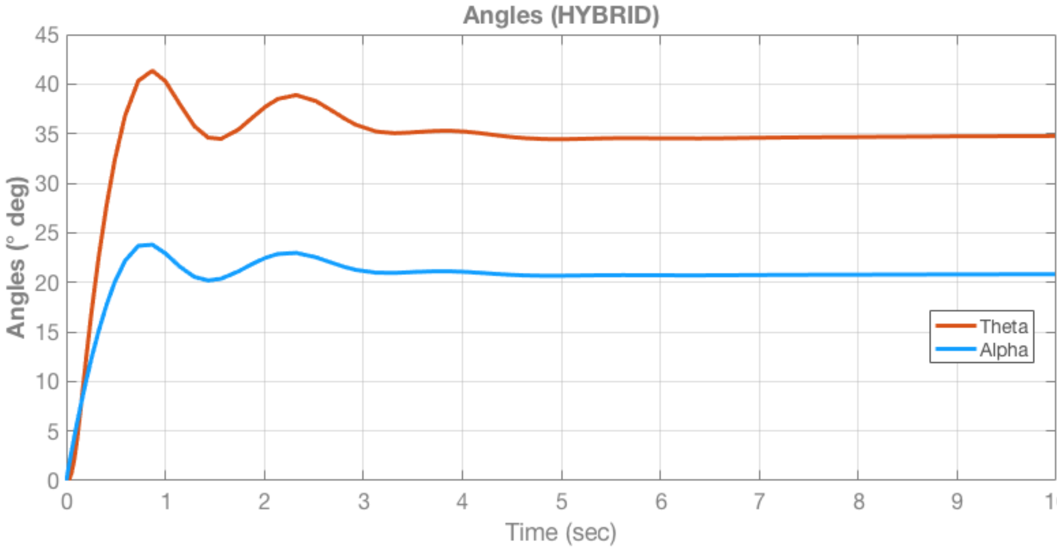


FIGURE 4.10: Angles' evolution in hybrid controlled structure

In this case, both angles assumed a value largely different from zero. The pitch angle had a maximum value of 41.20 ° then stabilizing at 34.73 °. The tilting angle α reached an equilibrium value of 20.81 ° after a peak of 23.77 °. In Fig. 4.11 the evolution of instant power consumption is presented. It can be noticed that, in term of power consumption, the first motor settled to a value of 8.13 W after a peak of 11.8 W. At the equilibrium point, the second motor reached a value of 7.34 W, after a maximum of 10.57 W.

This result is similar to what was seen in the tilting rotor case, in which the tilting motor was more consumptive than the fixed one. On the contrary, the overall consumed power is lower than that in the tilting case. In steady condition it is 15.47 W, passing a maximum of 22.38 W.

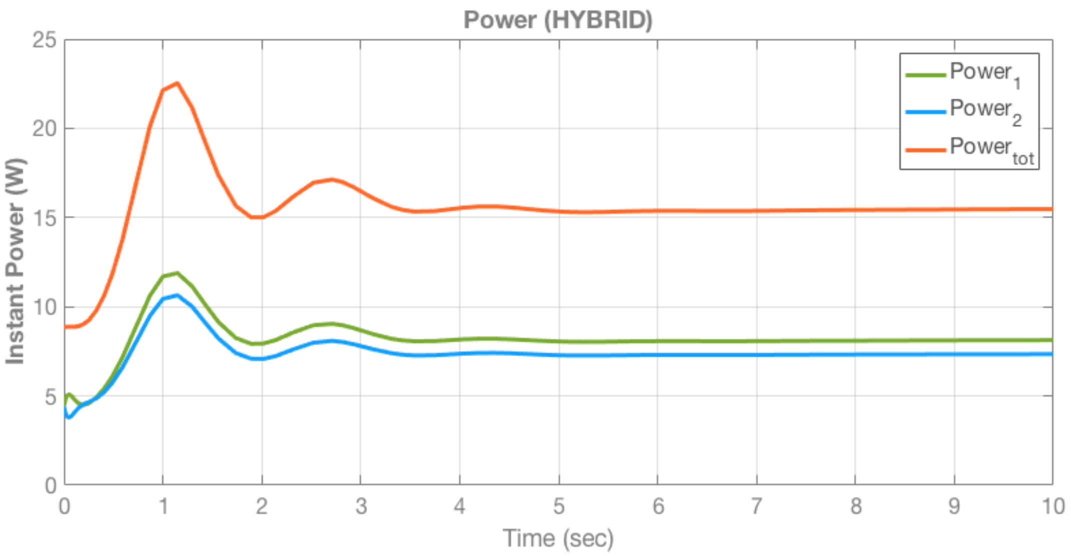


FIGURE 4.11: Instant power consumption evolution in hybrid controlled structure

Noticeably, the lowest power consumption can be achieved with a fixed rotor architecture. This result was obtained despite the fixed case's functional cost value. The lowest functional cost value is reported in the hybrid case, which depends on the content of Q_{hyb} and R_{hyb}. Despite this, the fixed rotors architecture simulation grants the best energy saving outcome. The table 4.1 summarizes these results and the differences among all simulated architectures.

Variable	Tilting	Fixed	Hybrid
Functional Cost J	11245.52	3940.86	3242.79
Peak Power $\hat{P}_{tot}$	28.54 W	22.52 W	22.38 W
Steady Power $\bar{P}_{tot}$	19.55 W	15.08 W	15.47 W

TABLE 4.1: Summary of consumptions

The differences among functional costs can be attributed to the optimal control matrices which represents some of the possible degrees of freedom during the tuning process. For this reason, the functional cost can't be assumed as an objective tool to evaluate energy consumption. In order to do achieve a reliable evaluation of energy consumption, a mechanical power consumption had to be estimated, as presented in section 2.2.4. In the latter

section, the overall powers show widely different values from the ones obtained here. One of the causes is the different magnitude of disturbances. In these last three simulations, the disturbances taken into account were around 10 times lower than in the simulations in Chapter 3. This substantial reduction is caused by a linearization restriction. Moreover, the system dimension's reduction, from 6 to 3 degrees of freedom, led to a lower number of actuators, therefore lowering the total values of power consumption.
Despite the restrictions caused by simplifications, the results obtained in this Chapter corresponds to those obtained in Chapter3. Even through this optimal control approach, the over-consumption of a tilting rotors architecture is confirmed. As already analyzed in the introduction, this is a considerable and remarkable limitation in terms of overall flight time. This is a fundamental variable which needs to be kept in due consideration when dealing with aerial manipulation tasks.

Chapter 5

Conclusions

In previous the chapters of this dissertation, a structure of a quad-rotor equipped with a combination of fixed and tilting propellers was described. A scenario in which a quad rotor is involved has been simulated adopting a variable force application. The results of testing a one-direction full-actuated quad-rotor, prove that this is not a good solution in terms of power and energy saving. Previous works and papers analyzed in my preliminary research, exposed in Chapter 1, didn't show tests in this direction. According to this study's results, tilting rotors consume more energy. In some cases, internal forces can also be produced, which has no sense in terms of energy consumption for aerial manipulating tasks.
A tilting rotor architecture designed as the one in this dissertation, can furnish high performances only during tasks which require low operative time and a very precise attitude specifications. To obtain a more general-purpose solution, a new architecture needs to be analyzed. Standing on what has been described in Chapter 2, in which thrust force is proportional to the second power of angular speed, and to refine the results obtained in this dissertation, a future work should introduce two additional tilting rotors. Furthermore, we can even formulate a new hypothesis: they could all be tilting together following parallel directions (as in the fixed rotor case) to avoid generating internal forces. In order to counteract external forces, a structure with all tilting rotors can deliver more equalized forces from its propulsion units. Acting in this direction, each mechanical power can be lower when compared to what showed in only two tilting rotors structure. An effect of this structural improvement could be to obtain an inferior, or at least equal, overall power consumption compared to fixed propeller structure. This energy advantage can be combined to the second advantage described before, i.e. controlling attitude disjointed by lateral translation during aerial manipulation. Still acting in this direction, operational time could be large enough to complete long time tasks while also saving energy.

Bibliography

[1] Z. Zaheer, A. Usmani, E. Khan, and M. A. Qadeer, *Aerial surveillance system using UAV*, Zakir Husain College of Engineering and Technology Aligarh Muslim University, India, 2016.

[2] A. G. Foina, R. Sengupta, P. Lerchi, Z. Liu, and C. Krainer, *Drones in smart cities: Overcoming barriers through air traffic control research*, 2015.

[3] L. D. P. Pugliese, F. Guerriero, E. Natalizio, and N. R. Zema, *A biobjective formulation for filming sport events problem using drones*, 2017.

[4] L. Wirth, P. Oettershagen, J. Ambühl, and R. Siegwart, *Meteorological path planning using dynamic programming for a solar-powered UAV*, ETH Zurich Autonomous Systems Lab, Zurich, Switzerland, 2013.

[5] P. E. I. Pounds, D. R. Bersak, and A. M. Dollar, *Grasping from the air: Hovering capture and load stability*, 2011.

[6] R. Spica, A. Franchi, G. Oriolo, H. H. Bülthoff, and P. R. Giordano, *Aerial grasping of a moving target with a quadrotor UAV*, 2012.

[7] D. Wuthier, D. Kominiak, E. Fresk, and G. Nikolakopoulos, *A geometric pulling force controller for aerial robotic workers*, Robotics Group at the Control Engineering Division of the Department of Computer, Electrical and Space Engineering, Luleå University of Technology, Sweden, 2016.

[8] M. Fumagalli, R. Naldi, A. Macchelli, R. Carloni, S. Stramigioli, and L. Marconi, *Modeling and control of a flying robot for contact inspection*, 2012.

[9] D. Mellinger, M. Shomin, N. Michael, and V. Kumar, *Cooperative grasping and transport using multiple quadrotors*, GRASP Laboratory - University of Pennsylvania, Philadelphia, USA, 2013.

[10] M. Ryll, H. H. Bülthoff, and P. R. Giordano, *Modeling and control of a quadrotor UAV with tilting propellers*, 2012.

[11] M. Ryll, D. Bicego, and A. Franchi, *Modeling and control of fast-hex: A fully-actuated by synchronized tilting hexarotor*, 2016.

[12] A. Oosedo, S. Abiko, S. Narasaki, A. Kuno, A. Konno, and M. Uchiyama, *Flight control systems of a quad tilt rotor unmanned aerial vehicle for a large attitude change*, 2015.

[13] H. Wopereis, T. van der Molen, T.H.Post, S. Stramigioli, and M. Fumagalli, *Mechanism for perching on smooth surfaces using aerial impacts*, 2016.

[14] J. F. Roberts, J.-C. Zufferey, and D. Floreano, *Energy management for indoor hovering robots*, 2008.

[15] N. Kumbhare, A. Rao, C. Gniady, W. Fink, and J. Rozenblit, *Waypoint-to-waypoint energy-efficient path planning for multi-copters*, Department of Electrical and Computer Engineering, University of Arizona, USA, 2016.

[16] G. Nandakumar, T. Ranganathan, A. B.J., and A. Thondiyath, *Design and analysis of a novel quadrotor system - voops*, 2015.

[17] B. Theys, G. Dimitriadis, P. Hendrick, and J. D. Schutter, *Influence of propeller configuration on propulsion system efficiency of multi-rotor unmanned aerial vehicles*, 2016.

[18] D.-K. Phung and P. Morin, *Modeling and energy evaluation of small convertible uavs*, 2013.

[19] B. Uragun, *Energy efficiency for unmanned aerial vehicles*, 2011.