Università degli Studi di Padova
Dipartimento di Ingegneria dell'Informazione
Master Degree in Control Systems Engineering

Academic Year 2023-2024

COMPARATIVE ANALYSIS OF TRACKING CONTROLLERS FOR AERIAL MANIPULATORS WITH DIFFERENT ACTUATION AND DEXTERITY CAPABILITIES

Supervisor:
Prof. Angelo Cenedese

Co-supervisor:
Prof. Giulia Michieletto

Master Candidate:
Marcello Sorge

Student ID:
2054136

July 3, 2024

To my parents Cristina and Paolo, my sister Laura,
to my wonderful family,
to Giulia, who has always been supporting me,
to my dearest friend Jacopo for always being there for me,
and to all the amazing people I've met along this journey.

ABSTRACT

Aerial manipulation is a trending topic in robotics research, as it allows mobile robots to perform active tasks such as grasping, transporting and manipulating objects. In particular, UAVs are great candidates to be used as aerial manipulators due to their ability to take off and land vertically.

In this work, two different kinds of aerial manipulators, namely a flying hand and an unmanned aerial manipulator, are analyzed. The goal is to compare the two paradigms, to investigate wether a particular configuration of the degrees of freedom presents some advantages from the control point of view.

SOMMARIO

La manipolazione aerea è un argomento di grande interesse nella ricerca robotica, in quanto permette ad agenti mobili di svolgere compiti quali la presa, il trasporto e la manipolazione di oggetti. In particolare, gli UAV risultano ottimi candidati ad essere impiegati come manipolatori aerei grazie alla loro capacità di decollare e atterrare verticalmente.

In questa tesi, due diversi tipi di manipolatori aerei, una flying hand ed un UAM, vengono analizzati. L'obiettivo è confrontare i due paradigmi, per indagare se una particolare configurazione dei gradi di libertà presenti dei vantaggi dal punto di vista del controllo.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

1

INTRODUCTION

1.1 AERIAL MANIPULATION OVERVIEW

The interest for Unmanned Aerial Vehicles (UAV) has been growing in the last decade, both as a research topic in academic environments and also within industrial companies and for civilian uses. The scenarios in which UAVs are utilized are several, those include: monitoring and inspection of physical infrastractures; smart responses to disasters [1], such as in areas hit by hearthquakes; patrolling specific areas for safety purposes; digital reconstruction of archaeological sites; agricultural and meteorological domains [2]. In particular, multirotors, i.e. aircrafts with two or more spinning propellers, are well suited for those kind of tasks with respect to other aerial vehicles, due to their agility and high manouverability, and their ability to hover at a certain height and to take off and land vertically. However, all the tasks listed above are passive tasks. Instead, as established by the European Strategic Research Agenda (eSRA), aerial (and space) robots are intended to be employed as robotic workers and co-workers, and for contact-aware exploration and inspection [3]. In order to perform those kind of active tasks, UAVs need to be endowed with manipulation skills. For aerial manipulation it is intended the grasping, transporting, positioning, assembly or disassembly of physical objects, performed with UAVs [4]. Aerial manipulators are useful in several applications, such as bridge inspection or high voltage electric lines inspection and fixing up. Those activities are unsafe for the human operator, and the aerial system would be able to provide assistance in critical and hazardous situations.
Clearly mounting a grasping tool on the drone, such as a gripper or a multi-fingered hand, reduces its payload and creates coupling effects in the system dynamics. Depending on how the interacting tool can be moved with respect to the floating base, contact-aware aerial platforms can be devided into two main categories:

- **Flying Hands (FH)**: the interacting tool cannot be moved independently from the floating base. This is the case when the tool has zero controllable DoFs and when the grasped object is connected to the multirotor through cables or tether mechanism.

In both cases, the actuation properties are entirely provided by the UAV;

- **Unmanned Aerial Manipulators (UAM)**: the interacting tool connected to the UAV base presents one or more controllable DoFs, and can therefore be moved independently from the drone.

The presence of a grasping tool attached to the multirotor, and of an object to be manipulated, generate coupling effects in the system dynamics. In particular, the masses and inertia tensors are increased, resulting in slower robot responses to control inputs. Those effects are even more critical in the case of UAMs, since the robotic manipulator dynamics depends on the actual configuration state of the whole system.
Considering the control problem for flying hands, adaptive control schemes have been developed to make the flight more agile when lifting and trasporting an object [5]. An hybrid motion-force control is designed in [6] for a rigid tool attached to the UAV. In order to compensate the payload loss due to the weight of the gripping tool, a scenario where multiple cooperative UAVs need to transport an object has also been investigated in [7].
Concerning the control of UAMs, two main strategies are adopted, namely a decentralized approach and a centralized approach. In the former, the UAV and the manipulator are treated as separate entities, and are controlled independently from each other. This requires robust control strategies for both the components, since the movements of the manipulator are treated as disturbances to be rejected by the UAV, and vice versa. In [8], a multilayer architecture is employed to control a multirotor equipped with a servo robot arm, where a momentum-based observer is employed to compensated unmodeled aerodynamic effects and the arm dynamics. For the UAV control, a variable parameter integral backstepping controller is designed in [9] to improve the results of a simple PID-based controller. On the other hand, in a centralized approach, the UAV and robotic manipulator are treated as a paired entity. The derivation of a complete dynamical model for a UAM is typically tackled either exploiting the Euler-Lagrange formalism, or the Newton-Euler formalism. Regardless of the method chosen to derive the robot dynamical model, several model-based control strategies have been developed to cope with the issues described above. A full state feedback LQR is designed in [10], while an adaptive sliding mode controller is studied in [11]. A moving mass control (MMC) for the UAV attitude dynamics is presented in [12]. Equipping the multirotor with an articulated arm increases the total DoFs of the system, and may lead to a fully and possibly over actuated system. A cartesian impedance control is developed in [13], where redundant DoFs are exploited to improve the UAV position control. An alternative approach, usually referred to as a

mixed approach, consists of developing separate controllers for the multirotor and robotic arm, and allowing the UAV to access feedback information of the manipulator generalized coordinate and vice versa, in order to compensate the dynamic coupling effects and improve performances.

1.2 OBJECTIVES

In this work, two robots are analysed, namely a FH composed of a fully actuated hexarotor and a zero DoFs interacting tool (end effector), and a UAM composed by an underactuated hexarotor, to which a 1 DoF robotic arm is attached. Schematic representations of the two robots are reported in figures 1.1 and 1.2. The complete and coupled dynamic models are derived for both systems, and two trajectories, one for each robot, are designed in order to yield similar trajectories for the end effectors. The goal of this thesis is to compare the FH and UAM paradigms, to investigate wether a system presents some advantages in terms of control performances and effort.

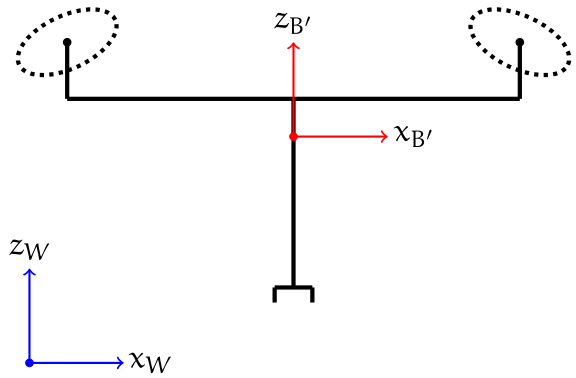

Figure 1.1: Schematic representation of the Flying Hand

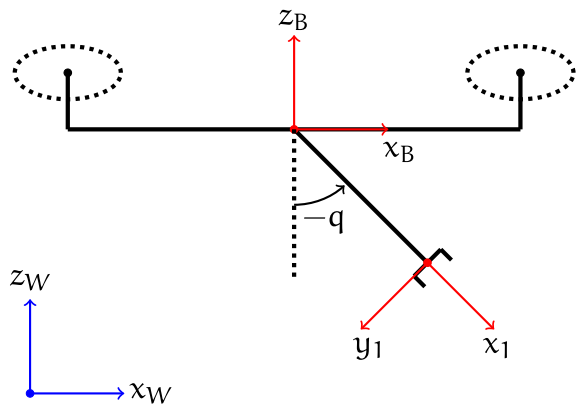

Figure 1.2: Schematic representation of the UAM

1.3 THESIS STRUCTURE

The thesis is divided into three main chapters, dealing with robot dynamic modeling, control design and improvements for each system, and assessing the control performances, respectively. Finally, some conclusive remarks and possible future developments in the context of aerial manipulation are discussed. The main topics of this work can be summarised as follows:

1. **Modeling**: in chapter 2, first the dynamic modeling of a generally tilted multirotot is derived, and the actuation properties are discussed. Then, the coupled dynamic model of the UAM studied is derived, through the Euler-Lagrange formalism. The actuation properties of the UAM are investigated with respect to the operational space variables, and the model is validated through some simple tests performed by applying forces and torques in open-loop. Finally, the FH model is derived exploiting the Newton-Euler formalism. It is established that the actuation properties of the full robot coincide with those of the UAV base, and also this model is validated through simulations in open loop.

2. **Control Design**: in chapter 3, tracking algorithms are designed for the two systems. For the UAM, a decentralized approach is considered, and two control laws for the drone and the link, respectively, are derived. For the FH, only the UAV controller is designed, exploiting the system's full actuation. Due to the poor performance achieved by the UAM compared with the FH, different improvement strategies are developed.

3. **Simulations and Results**: in chapter 4, the robots responses to the control algorithms developed in chapter 3 are tested. A qualitative analysis is performed in order to assess the convergence properties of the control laws. Afterwards, the performances and control effort required by the FH and UAM are analyzed through the introduction of some performance indexes, in order to compare the two paradigms.

4. **Conclusions**: in chapter 5, the main points of this work are summarized and briefly discussed. Some insights on possible related future works are provided.

2

MODELING

2.1 MULTIROTOR MODEL

2.1.1 *Dynamic Equations*

Let $\mathbb{W} = \{x_{\mathbb{W}}, y_{\mathbb{W}}, z_{\mathbb{W}}\}$ be the world (inertial) frame, and let $\mathbb{B} = \{x_{\mathbb{B}}, y_{\mathbb{B}}, z_{\mathbb{B}}\}$ be the body fixed frame with origin in the center of mass of the vehicle. Moreover, let $\mathbb{O}^i = \{x_{\mathbb{O}^i}, y_{\mathbb{O}^i}, z_{\mathbb{O}^i}\}$ be the reference frame attached to each propeller, where $i = 1, ..., n$, n is the number of propellers and $z_{\mathbb{O}^i}$ defines the spinning axis of each rotor. Most UAVs are characterized by alternately spinning propellers. In a generally tilted multirotor, the spinning axis of each propeller can be rotated by an angle α_i about axis $x_{\mathbb{O}^i}$ and by an angle β_i about axis $y_{\mathbb{O}^i}$. Then, considering a composition of rotations with respect to the current frame, the orientation of the spinning axis of each motor with respect to $\mathbb{B}$ can be expressed as

$$z_{Pi} = R_z(\gamma_i)R_y(\beta_i)R_x(\alpha_i)\hat{e}_3 \tag{2.1}$$

where R_x, R_y, R_z are the canonical $\mathbb{SO}(3)$ matrices representing rotations around axes x, y and z respectively, γ_i is the angle between two consecutive propeller arms and $\hat{e}_3 = (0, 0, 1)^T$. Considering ω_i the spinning velocity of each rotor, the force generated by each propeller is given by

$$f_i = c_{f_i}\omega_i|\omega_i|z_{Pi} \tag{2.2}$$

where $c_{f_i} > 0$ is a force coefficient that depends on the propeller structure, and $u_i = \omega_i|\omega_i|$ is the motor input. Then, the total force applied on the vehicle cenetr of mass, expressed in body frame, is given by

$$f_c = \sum_{i=1}^{n} c_{f_i}u_i z_{Pi} = Fu \tag{2.3}$$

where $u = (u_1, ..., u_n)^T$, $F \in \mathbb{R}^{3\times n}$ and column i of matrix F corresponds to the vector $c_{f_i}z_{Pi}$.
Since the propellers are spinning in a viscous medium, the vehicle

is also subject to drag torques, which can be expressed in the body frame as

$$\tau_i^d = c_{\tau_i} u_i z_{Pi} \tag{2.4}$$

where c_{τ_i} is a drag coefficient that depends on the propeller structure, and $c_{\tau_i} > 0$ if the natural spinning direction of motor i is clockwise, or $c_{\tau_i} < 0$ if the natural spinning direction is counterclockwise. Moreover, the force generated by each propeller causes the emergence of a torque in the center of mass of the vehicle. For the i-th propeller, this is

$$\tau_i^f = r_i \times f_i \tag{2.5}$$

where r_i is the vector connecting the origin of reference frame $\mathbb{B}$ and the origin of reference frame $\mathbb{O}^i$. Then, the total torque acting on the center of mass of the multirotor can be expressed as

$$\tau_c = \sum_{i=i}^{n} (c_{\tau_i} u_i z_{Pi} + r_i \times c_{f_i} u_i z_{Pi}) = Mu \tag{2.6}$$

where $M \in \mathbb{R}^{3\times n}$ and column i of matrix M corresponds to the vector $c_{\tau_i} z_{P_i} + r_i \times c_{f_i} z_{Pi}$.
Considering the Newton Euler formulation, the dynamic equations of a generally tilted multirotor can be expressed in matrix form as

$$\begin{pmatrix} m_B \ddot{p}_B \\ I_B \dot{\omega}_B^B \end{pmatrix} = \begin{pmatrix} R_{WB}(\Phi) f_c \\ \tau_c \end{pmatrix} - \begin{pmatrix} m_B g \hat{e}_3 \\ \omega_B^B \times I_B \omega_B^B \end{pmatrix} \tag{2.7}$$

where $\ddot{p}_B \in \mathbb{R}^3$ is the linear acceleration of the vehicle's center of mass with respect to frame $\mathbb{W}$, $\omega_B^B \in \mathbb{R}^3$ is the angular velocity of the drone expressed in body frame $\mathbb{B}$, m_B and $I_B \in \mathbb{R}^{3\times 3}$ are the drone mass and body frame inertia tensor, respectively, $R_{WB} \in \mathbb{SO}(3)$ is the body to world rotation matrix, obtained considering a triplet of yaw-pitch-roll Euler angles $\Phi = (\rho, \theta, \psi)^T \in \mathbb{R}^3$ in the current frame, g is the gravity acceleration and $\hat{e}_3 = (0, 0, 1)^T \in \mathbb{R}^3$ is the third canonical vector of reference frame $\mathbb{W}$.
It is useful to understand what is the relation between the UAV angular velocity ω and the time derivative of the Euler angles. The time derivative of $R_{WB}(\Phi)$ can be computed as

$$\dot{R}_{WB}(\Phi) = R_{WB}(\Phi)[\omega_B^B]_\times \tag{2.8}$$

with $[\omega_B^B]_\times$ being the skew symmetric matrix built on vector ω_B^B. This should be consistent with

$$\dot{R}_{WB}(\Phi) = \frac{\partial R_{WB}(\Phi)}{\partial \rho}\dot{\rho} + \frac{\partial R_{WB}(\Phi)}{\partial \theta}\dot{\theta} + \frac{\partial R_{WB}(\Phi)}{\partial \psi}\dot{\psi} \tag{2.9}$$

By imposing the eqaulity between 2.8 and 2.9, it holds $\omega_B^B = T(\Phi)\dot{\Phi}$, which can be explicitly written as

$$\omega_B^B = \begin{pmatrix} 1 & 0 & -\sin\theta \\ 0 & \cos\rho & \cos\theta\sin\rho \\ 0 & -\sin\rho & \cos\theta\sin\rho \end{pmatrix} \begin{pmatrix} \dot{\rho} \\ \dot{\theta} \\ \dot{\psi} \end{pmatrix} \tag{2.10}$$

It is important to notice that the matrix $T(\Phi)$ has determinant $\cos\theta$, therefore it is invertible for all angles except $\theta = \frac{\pi}{2} + k\pi$.

2.1.2 *Actuation and Decoupling*

A second order differential equation dynamical system described by $\ddot{\eta} = f(\eta, \dot{\eta}, u, t)$, where η is the generalized position, u the input vector and t the time, is said to be fully actuated if the map $f : u \to \ddot{\eta}$ is surjective. Usually, the dynamic equations describing the system can be expressed in input affine form:

$$\ddot{\eta} = f_1(\eta, \dot{\eta}, t)u + f_2(\eta, \dot{\eta}, t) \tag{2.11}$$

from which it is evident that the actuation properties are defined by $f_1(\eta, \dot{\eta}, t)$. Following this definition, the drone model derived in 2.7 can be rewritten in input affine form as:

$$\begin{pmatrix} \ddot{p}_B \\ \dot{\omega}_B^B \end{pmatrix} = \begin{pmatrix} \frac{1}{m_B} R_{WB}(\Phi) & 0_{3\times3} \\ 0_{3\times3} & I_B^{-1} \end{pmatrix} \begin{pmatrix} F \\ M \end{pmatrix} u - \begin{pmatrix} g\hat{e}_3 \\ I_B^{-1}(\omega_B^B \times I_B \omega_B^B) \end{pmatrix} \tag{2.12}$$

where it is highlited

$$f_1 = \begin{pmatrix} \frac{1}{m_B} R_{WB} & 0_{3\times3} \\ 0_{3\times3} & I_B^{-1} \end{pmatrix} \begin{pmatrix} F \\ M \end{pmatrix} \tag{2.13}$$

Given that $\det(R_{WB}(\Phi)) = 1$ and the inertia tensor is positive definite, the actuation properties only depend on the image space of matrix $(F^T, M^T)^T$. Considering the full pose of a three dimensional rigid object, where $\eta = (p_B^T, \Phi^T)^T \in \mathbb{R}^6$, it holds:

- if $\mathrm{rank}((F^T, M^T)^T) < 6$, the system is under actuated;
- if $\mathrm{rank}((F^T, M^T)^T) = 6$, the system is fully actuated.

Usually, it holds $\mathrm{rank}(M) \geqslant \mathrm{rank}(F)$. Considering matrix M, there exist matrices $A \in \mathbb{R}^{n\times3}$, $B \in \mathbb{R}^{n\times(n-3)}$, such that $\mathbb{I}m(A) = \mathrm{Ker}(M)^\perp$, $\mathbb{I}m(B) = \mathrm{Ker}(M)$. Then, the matrix $(A, B) \in \mathbb{R}^{n\times n}$ can be considered as a change of basis matrix for $\mathbb{R}^n$. It is now possible to rewrite the input vector as $u = (A, B)\tilde{u}$, where $\tilde{u} = (\tilde{u}_A^T, \tilde{u}_B^T)^T$. The input forces and torques for the platform can now be expressed in terms of the input vector $\tilde{u}$:

$$\begin{aligned} f_c &= FA\tilde{u}_A + FB\tilde{u}_B = f_{cA} + f_{cB} \\ \tau_c &= MA\tilde{u}_A = \tau_{cA} \end{aligned} \tag{2.14}$$

where $MB\tilde{u}_B = 0$ since $\mathbb{I}m(B) = Ker(M)$. f_{cA} represents the forces that are coupled with moments acting on the vehicle. Given this decomposition, it holds:

- if $\dim(\mathbb{I}m(F)) = \dim(\mathbb{I}m(FB))$, the system is fully decoupled;
- if $\dim(\mathbb{I}m(F)) > \dim(\mathbb{I}m(FB)) > 0$, the system is partially coupled;
- if $\dim(\mathbb{I}m(F)) > \dim(\mathbb{I}m(FB)) = 0$, the system is fully coupled.

Considering a fully decoupled platform with $\dim(\mathbb{I}m(FB)) = 3$, the control force f_c can always be chosen in such a way that it is decoupled from the control torque τ_c. Therefore, a fully decoupled platform with $\dim(\mathbb{I}m(FB)) = 3$ is fully actuated.

2.2 UAM MODEL

The UAM analyzed in this work consists of a coplanar hexarotor, with a 1 DoF robotic arm attached to its center of mass. The arm is actuated by a revolute joint that allows rotations about body frame axis $y_{\mathcal{B}}$. Since the UAV is coplanar, the propeller tilt angles are null, i.e. $\alpha_i = \beta_i = 0° \forall i$. Table 2.1 contains the physical paramters describing the case of study robot.

Name	Symbol [unit]	Value
UAV mass	m_B [Kg]	3.5
γ angle	γ [°]	60
Arm length	r [m]	0.358
Force coefficient	c_f [N/(rad/s)^2]	0.0015
Drag coefficient	c_τ [Nm/(rad/s)^2]	4.59×10^{-5}
UAV inertia tensor	I_B [kgm^2]	$\begin{pmatrix} 0.155 & 0 & 0 \\ 0 & 0.147 & 0 \\ 0 & 0 & 0.251 \end{pmatrix}$
Link mass	m_l [Kg]	0.29
Link length	l [m]	0.14
Link inertia tensor	I_l [kgm^2]	$\begin{pmatrix} 3.62 & 0 & 0 \\ 0 & 6.55 & 0 \\ 0 & 0 & 6.55 \end{pmatrix} \times 10^{-4}$

Table 2.1: UAM parameters

2.2.1 *Euler-Lagrange Dynamic Equations*

Consider a body fixed frame $\mathbb{B} = \{x_{\mathbb{B}}, y_{\mathbb{B}}, z_{\mathbb{B}}\}$ with origin located at the center of mass of the vehicle. The drone position in world frame is described by $p_B = (x, y, z)^T$, while its orientation is represented by the ZYX set of yaw-pitch-roll angles in the current frame, $\Phi = (\rho, \theta, \psi)^T$. Then, considering 2.10, the angular velocity of the drone in world frame can be written as

$$\omega_B = R_{WB}(\Phi)T(\Phi)\dot{\Phi} \tag{2.15}$$

Consider now a redundant body fixed frame $\mathbb{O} = \{x_{\mathbb{O}}, y_{\mathbb{O}}, z_{\mathbb{O}}\}$ with the same origin as $\mathbb{B}$, which is the base frame for the manipulator. The frame is assigned in such a way that it holds $z_{\mathbb{O}} = y_{\mathbb{B}}$. Finally, consider a link fixed frame $\mathbb{1} = \{x_{\mathbb{1}}, y_{\mathbb{1}}, z_{\mathbb{1}}\}$, where $z_{\mathbb{O}} \parallel z_{\mathbb{1}}$, according to the Denavit-Hartenberg convention, as it can be seen in figure 1.2. Then, the world frame position of the link's center of mass can be expressed as

$$p_l = p_B + R_{WB}(\Phi)R_{BO}p_l^O \tag{2.16}$$

where p_l^O denotes the position of the center of mass of the link with respect to frame $\mathbb{O}$ and $R_{BO} \in \mathbb{SO}(3)$ is the constant transformation matrix from frame $\mathbb{O}$ to frame $\mathbb{B}$. The velocity of the center of mass of the link in frame $\mathbb{O}$ can be obtained as

$$\dot{p}_l^O = J_p(q)\dot{q} \tag{2.17}$$

where q is the link generalized coordinate and $J_p(q) \in \mathbb{R}^{3x1}$ is the positional geometric jacobian of the link. Similarly, for the link's angular velocity it holds

$$\omega_l^O = J_o(q)\dot{q} \tag{2.18}$$

where $J_o(q) \in \mathbb{R}^{3x1}$ is the orientation geometric jacobian of the link. The dynamic equations of the full robot can be derived exploiting the Euler-Lagrange formalism [14], in which the mechanical structure is characterized by the lagrangian function $\mathcal{L} = \mathcal{K} - \mathcal{U}$, where $\mathcal{K}$ is the total kinetic energy of the system and $\mathcal{U}$ is the total potential energy of the system. Considering a vector of generalized coordinates $\xi = (p_B^T, \Phi^T, q)^T$, the lagrange equations can be expressed as

$$\frac{d}{dt}\frac{\partial \mathcal{L}}{\partial \dot{\xi}} - \frac{\partial \mathcal{L}}{\partial \xi} = \tau \tag{2.19}$$

where $\tau \in \mathbb{R}^{7x1}$ represents the generalized forces acting on the system due to actuation inputs. The kinetic energy contribution due to the UAV can be expressed as

$$\mathcal{K}_B = \frac{1}{2}m_B\dot{p}_B^T\dot{p}_B + \frac{1}{2}\omega_B^T R_{WB}(\Phi)I_B R_{WB}(\Phi)^T\omega_B \tag{2.20}$$

Considering 2.15, the drone kinetic energy can be rewritten in terms of the vehicle's generalized coordinates as

$$\mathcal{K}_B = \frac{1}{2} m_B \dot{p}_B^T \dot{p}_B + \frac{1}{2} \dot{\Phi}^T T(\Phi)^T I_B T(\Phi) \dot{\Phi} \tag{2.21}$$

The kinetic energy contribution due to the link can be written as

$$\mathcal{K}_l = \frac{1}{2} m_l \dot{p}_l^T \dot{p}_l + \frac{1}{2} \omega_l^T R_{WB}(\Phi) R_{BO} R_{O1}(q) I_l R_{O1}(q)^T R_{BO}^T R_{WB}(\Phi)^T \omega_l \tag{2.22}$$

where $R_{O1}(q) \in \mathbb{SO}(3)$ is the rotation matrix expressing the orientation of frame $\mathbb{1}$ with respect to frame $\mathbb{O}$, and $\omega_l = R_{WB}(\Phi) R_{BO} \omega_l^O$ represents the world frame link angular velocity. The total kinetic energy of the robot $\mathcal{K} = \mathcal{K}_\mathcal{B} + \mathcal{K}_l$ can be rewritten exploiting 2.16, 2.17 and 2.18 as

$$\mathcal{K} = \frac{1}{2} \dot{\xi}^T B(\xi) \dot{\xi} \tag{2.23}$$

where $B(\xi) \in \mathbb{R}^{7x7}$ is a positive definite symmetric inertia matrix, whose block components are:

$$\begin{aligned}
B_{11} =& (m_B + m_l) I_3 \\
B_{12} =& - m_l S(p_l) R_{WB}(\Phi) T(\Phi) \\
B_{13} =& m_l R_{WB}(\Phi) R_{BO} J_p(q) \\
B_{22} =& T(\Phi)^T I_B T(\Phi) + m_l T(\Phi)^T R_{WB}(\Phi)^T S(p_l)^T S(p_l) R_{WB}(\Phi) T(\Phi) \\
& + T(\Phi)^T R_{BO} R_{O1}(q) I_l R_{O1}(q)^T R_{BO}^T T(\Phi) \\
B_{23} =& - m_l T(\Phi)^T R_{WB}(\Phi)^T S(p_l)^T R_{WB}(\Phi) R_{BO} J_p(q) \\
& + T(\Phi)^T R_{BO} R_{O1}(q) I_l R_{O1}(q)^T J_o(q) \\
B_{33} =& m_l J_p(q)^T J_p(q) + J_o(q)^T R_{O1}(q) I_l R_{O1}(q)^T J_o(q)
\end{aligned}$$

where $S(p_l) = [R_{WB}(\Phi) R_{BO} p_l^O]_\times$. Concerning the potential energy of the system, the UAV contribution can be computed as

$$\mathcal{U}_B = m_B g \hat{e}^T p_B \tag{2.24}$$

where $\hat{e}$ is a unit vector selecting the axis along which the gravity acts, while the link contribution can be computed as

$$\mathcal{U}_l = m_l g \hat{e}^T (p_B + R_{WB}(\Phi) R_{BO} p_l^O) \tag{2.25}$$

Finally, the lagrange equations of the system can be rewritten as

$$B(\xi) \ddot{\xi} + C(\xi, \dot{\xi}) \dot{\xi} + g(\xi) = \tau \tag{2.26}$$

where

$$g(\xi) = \frac{\partial \mathcal{U}}{\partial \xi}^T \tag{2.27}$$

and $C(\xi, \dot{\xi})$ is a matrix calculated exploiting the Christoffel symbols of the first type, whose generic element is given by [15]

$$C(i,j) = \sum_{k=1}^{7} \frac{1}{2}\left(\frac{\partial B(i,j)}{\partial \xi_k} + \frac{\partial B(i,k)}{\partial \xi_j} - \frac{\partial B(j,k)}{\partial \xi_i}\right)\dot{\xi}_k \tag{2.28}$$

2.2.2 *Actuation and Decoupling*

The UAV base of the aerial manipulator has coplanar propellers, meaning that the tilt angles are $\alpha_i = \beta_i = 0° \forall i$. For this kind of platform, considering the procedure discussed in 2.1.2, it can be verified that $\mathrm{rank}((F^T, M^T)^T) = 4$, and therefore the drone is underactuated. In fact, it is not possible to perform motion on the xy-plane without changing the UAV's attitude. The UAM then presents five independent control inputs, which are the thrust force f_z, the UAV torques $\tau_c = (\tau_\rho, \tau_\theta, \tau_\psi)^T$ and the link torque τ_l. However, for aerial manipulation tasks, it is useful to study the actuation properties of the robot at the end effector level. For this purpose it is necessary to define the operational space of the manipulator. Such space can be defined by a vector containing the end effector position and orientation with respect to $\mathbb{W}$:

$$x_e = (p_e^T, \Phi_e^T)^T \tag{2.29}$$

where p_e represents the end effector position and Φ_e represents the end effector orientation through a minimal representation. In particular, for the robot considered in this work it holds

$$p_e = p_B + R_{WB}(\Phi) R_{BO} R_{O1}(q)(l, 0, 0)^T \tag{2.30}$$

$$\Phi_e = \begin{pmatrix} \rho_e \\ \theta_e \\ \psi_e \end{pmatrix} = \begin{pmatrix} \rho \\ \frac{\pi}{2} + \theta + q \\ \psi \end{pmatrix} \tag{2.31}$$

By rewriting the operational space variables as functions of the generalized coordinates as $x_e = k(\xi)$, a relation can be established between the end effector velocities (i.e., the time derivatives of the operational space variables) and the joint velocities, according to

$$\dot{x}_e = J_A(\xi)\dot{\xi} \tag{2.32}$$

where $J_A(\xi)$ is the analytical jacobian, obtained through time differentiation of the map $k(\xi)$. The analytical jacobian can be expressed as

$$J_A(\xi) = \begin{pmatrix} E_{3\times 3} & (\frac{\partial p_e}{\partial \rho})^T & (\frac{\partial p_e}{\partial \theta})^T & (\frac{\partial p_e}{\partial \psi})^T & (\frac{\partial p_e}{\partial q})^T \\ 0_{3\times 3} & \hat{e}_1 & \hat{e}_2 & \hat{e}_3 & \hat{e}_2 \end{pmatrix} \tag{2.33}$$

where $E_{3\times3}$ is the identity matrix $\in \mathbb{R}^{3\times3}$, $\hat{e}_i$ is the i-th canonical vector $\in \mathbb{R}^3$. It can be noticed that, with this parametrization of the operational space, the jacobian $J_A(\xi)$ always has row rank $= 6$. By taking the time derivative of 2.32, it holds

$$\ddot{x}_e = J_A(\xi)\ddot{\xi} + \dot{J}_A(\xi)\dot{\xi} \tag{2.34}$$

Considering the dynamic model in 2.26, the generalized forces τ can be rewritten in terms of the equivalent end effector generalized forces γ_e as

$$\tau = J(\xi)^T \gamma_e \tag{2.35}$$

where $J(\xi)$ is the robot geometric jacobian. The analytical and geometric jacobian are related by a transformation matrix $T_A(\xi)$, according to $J(\xi) = T_A(\xi)J_A(\xi)$. Then, considering 2.32 and 2.34, the dynamical model can be rewritten in terms of the operational space variables as

$$B_x(\xi)\ddot{x}_e + C_x(\xi,\dot{\xi})\dot{x}_e + g_x(\xi) = \gamma_A \tag{2.36}$$

where

$$B_x(\xi) = J_A(\xi)^{-T} B(\xi) J_A(\xi)^{-1} \tag{2.37}$$

$$C_x(\xi,\dot{\xi}) = J_A(\xi)^{-T}(C(\xi,\dot{\xi}) - B(\xi)J_A(\xi)^{-1}\dot{J}_A(\xi))J_A(\xi)^{-1} \tag{2.38}$$

$$g_x(\xi) = J_A(\xi)^{-T} g(\xi) \tag{2.39}$$

$$\gamma_A = T_A(\xi)^T \gamma_e \tag{2.40}$$

Moreover, the input torques τ can be expressed in terms of the UAV and link input variables $u_{UAM} = (u^T, \tau_l)^T$ [16] [17] as

$$\tau = \begin{pmatrix} R_{WB}(\Phi) & 0_{3x3} & 0_{3x1} \\ 0_{3x3} & T(\Phi)^T R_{WB}(\Phi) & 0_{3x1} \\ 0_{1x3} & 0_{1x3} & 1 \end{pmatrix} \begin{pmatrix} F & 0_{3x1} \\ M & 0_{3x1} \\ 0_{1x6} & 1 \end{pmatrix} u_{UAM} \tag{2.41}$$

Finally, considering $\tau = J_A(\xi)^T \gamma_A$ and isolating the operational space acceleration in 2.36, the dynamic equations describing the system can be expressed in input affine form, with

$$f_1 = J_A(\xi)^{-T} \begin{pmatrix} R_{WB}(\Phi) & 0_{3x3} & 0_{3x1} \\ 0_{3x3} & T(\Phi)^T R_{WB}(\Phi) & 0_{3x1} \\ 0_{1x3} & 0_{1x3} & 1 \end{pmatrix} \begin{pmatrix} F & 0_{3x1} \\ M & 0_{3x1} \\ 0_{1x6} & 1 \end{pmatrix} \tag{2.42}$$

where the premultiplication by $B_x(\xi)^{-1}$ has been omitted since the inertia matrix is positive definite, and therefore does not influence the rank of f_1. Assuming $\theta \neq \frac{\pi}{2}$, it can be verified that $rank(f_1) \leqslant 5$ and the system is underactuated.

2.2.3 *UAM Model Validation*

To assess the validity of the UAM model previously derived, some tests are performed by applying constant forces and torques to the robot in open loop, while the UAM initial conditions are set as $\xi_0 = 0^T$. First, the ability of the system to maintain hovering is tested by applying a body frame thrust force $f_z = (m_B + m_l)g$ N, while no torque input is provided. As expected, the robot does not move from its initial pose, and no relative motion between the UAV and the end effector happens. This behaviour can be observed from figures 2.1, 2.2 and 2.3. Moreover, from figure 2.1, it can be noticed that the z coordinate of the end effector is negative. This is coherent with the given initial condition for the system, since the UAV center of gravity is located at $p_{B,0} = 0^T$. Consequently, given that $q = 0°$, the end effector points downwards. It can be noticed that the z coordinate of the gap between the UAV and end effector z coordinates is exactly equal to the link's length.

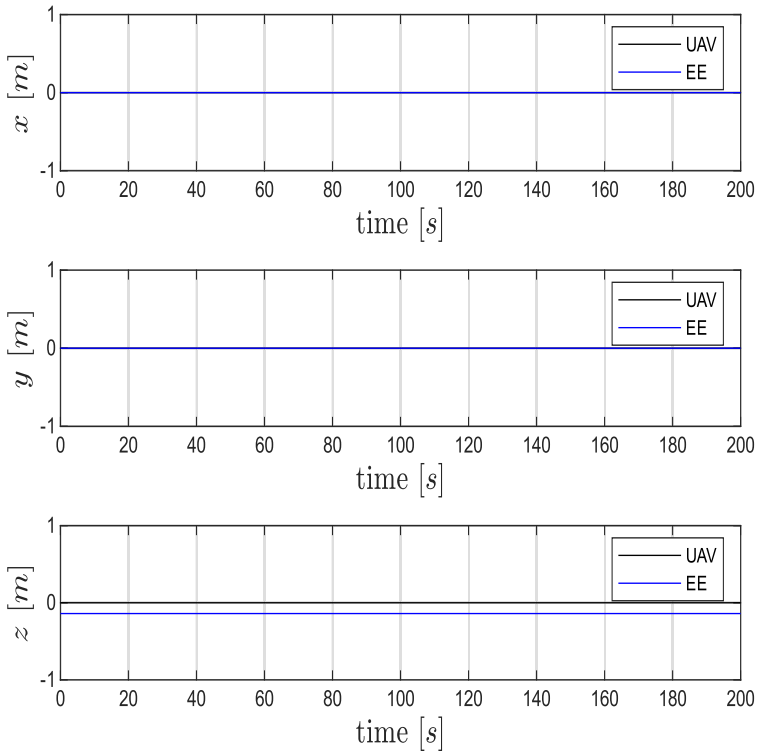

Figure 2.1: UAV CoM and End Effector position at hovering

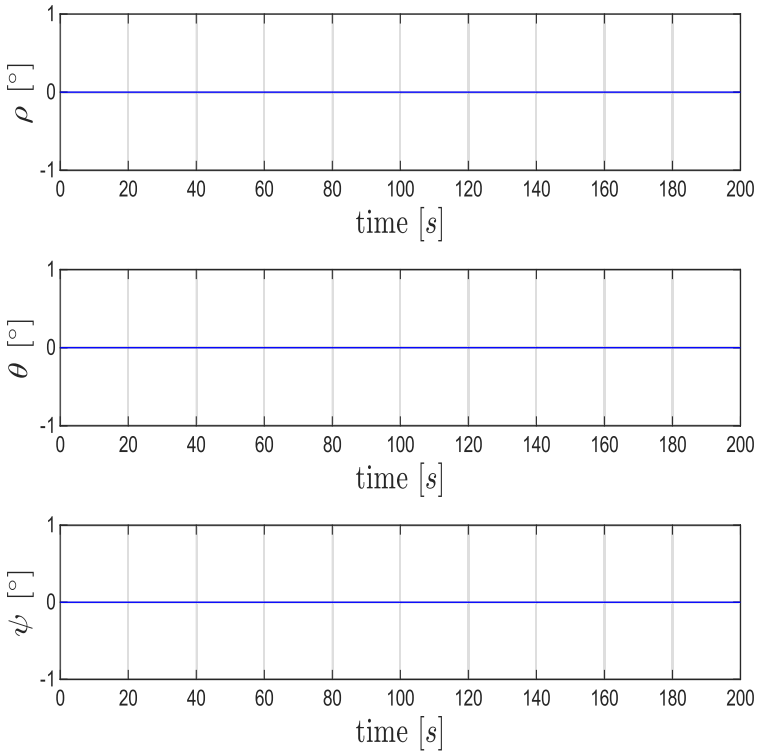

Figure 2.2: UAV Euler angles at hovering

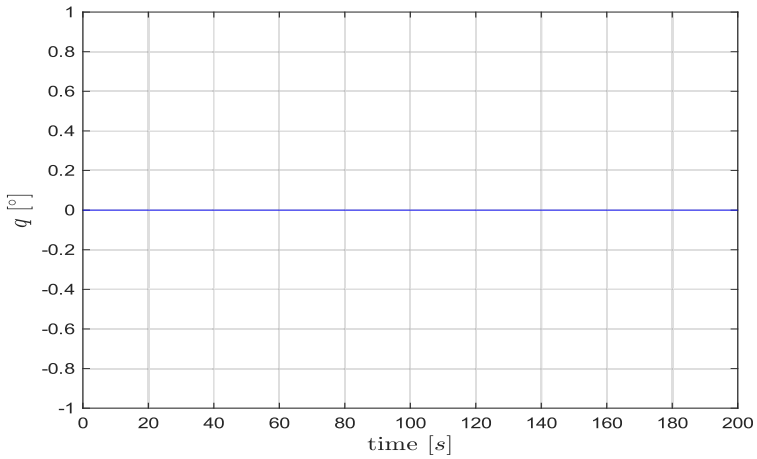

Figure 2.3: q coordinate at hovering

Then, a torque $\tau_\psi = 0.001$ Nm is applied to the system, as well as a thrust force equal to the one in the previous test. No additional torque is provided. With this choice of input, the robot is expected to only change its heading direction. Indeed, from figure 2.5, it can be noticed that the ψ angle increases as a quadratic function, due to the constant input torque, while the ρ and θ angles remain constant. The robot also maintain its current position, and no relative motion between the UAV and link is observed, as showed in figures 2.5 and 2.6.

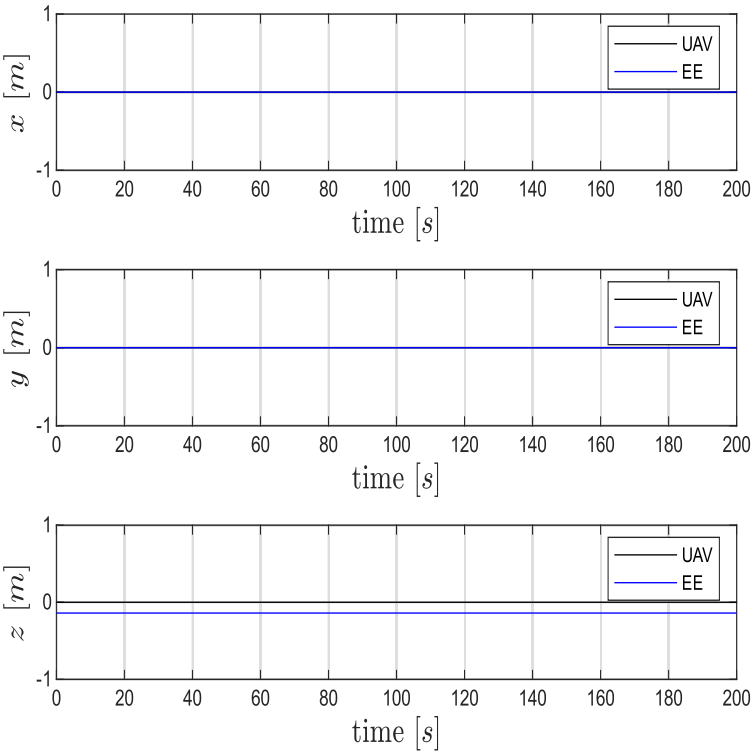

Figure 2.4: UAV and End Effector position with constant input torque τ_ψ

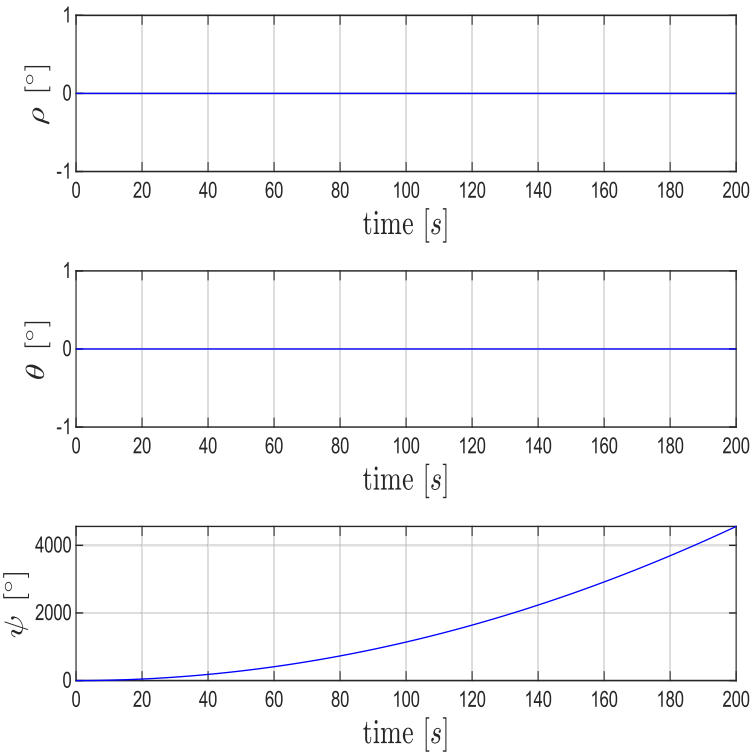

Figure 2.5: UAV Euler angles subject with constant input torque τ_ψ

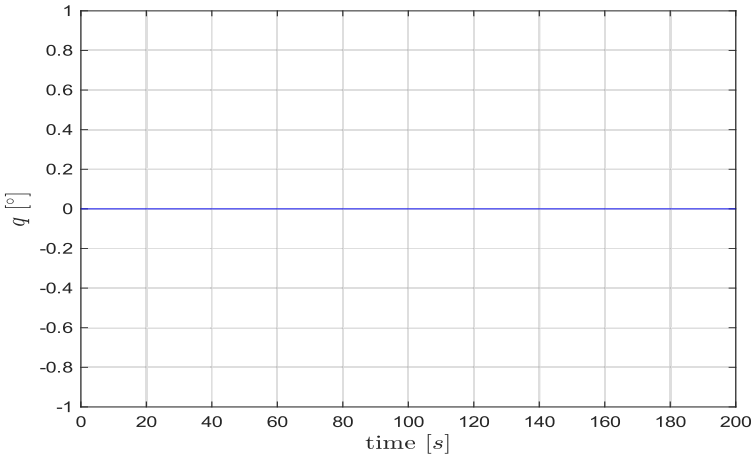

Figure 2.6: q coordinate with constant input torque τ_ψ

Then, a torque $\tau_\rho = 0.001$ Nm is fed as input to the robot, alongside the thrust force f_z of the previous tests. The results can be seen in figures 2.7, 2.8 and 2.9. Due to the drone being underactuated, the ρ rotation causes a translation along the y direction. Moreover, due to the body frame rotation, the thrust force component along the vertical direction reduces, causing the robot's altitude to decrease. Since the link can only rotate about axis z_O, the input torque provided to the UAV does not generate end effector motion relative to the drone.

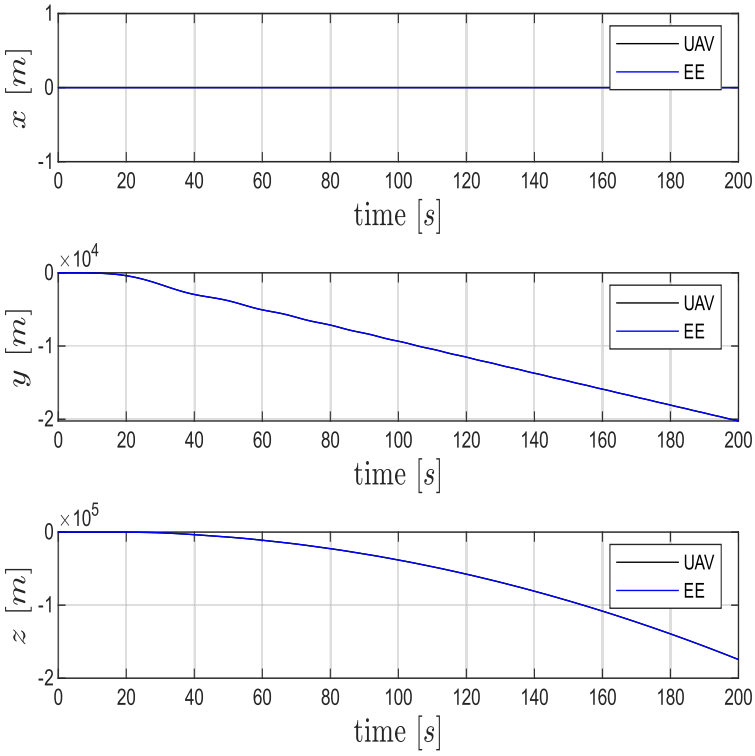

Figure 2.7: UAV and End Effector position with constant input torque τ_ρ

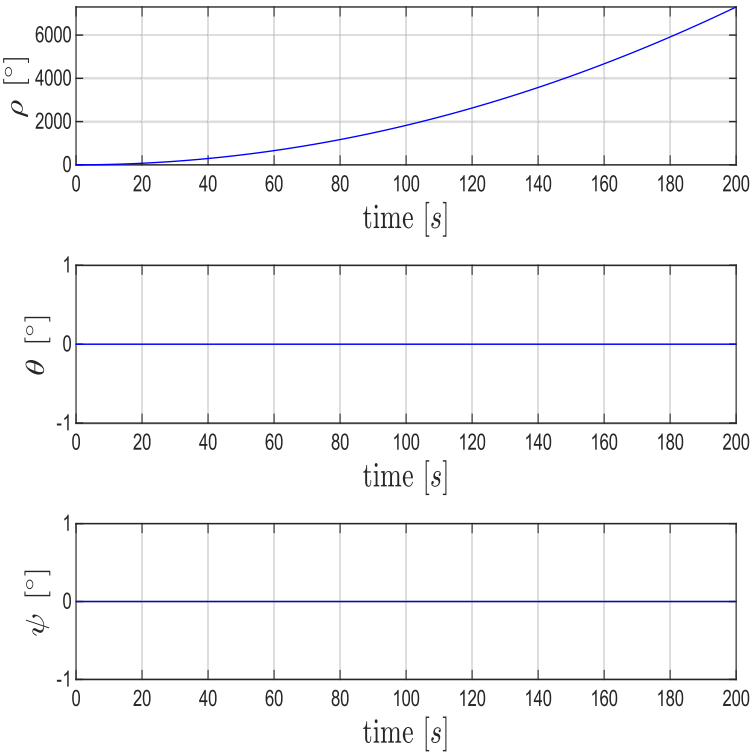

Figure 2.8: UAV Euler angles with constant input torque τ_ψ

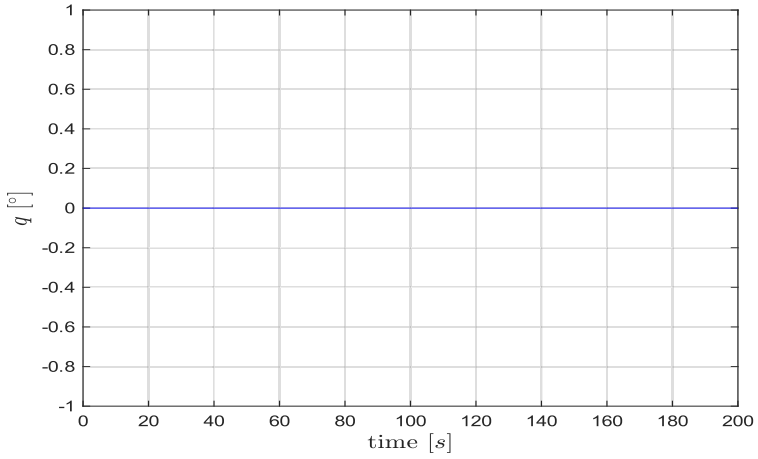

Figure 2.9: q coordinate with constant input torque τ_ψ

Then, the system is fed with the body frame thrust force f_z, and a torque $\tau_\theta = 0.001$ Nm. The input torque acts on the same axis of rotation of the link revolute joint. Due to the coupled UAV and link dynamics, the link experiences a constant reaction torque, which causes oscillations around an equilibrium point characterised by $\bar{q} \neq 0$, as can be seen from figure 2.12. The cartesian UAV motion is analogous to that of the previous test, with the motion happening along the x direction, as it can be seen from figures 2.10 and 2.11.

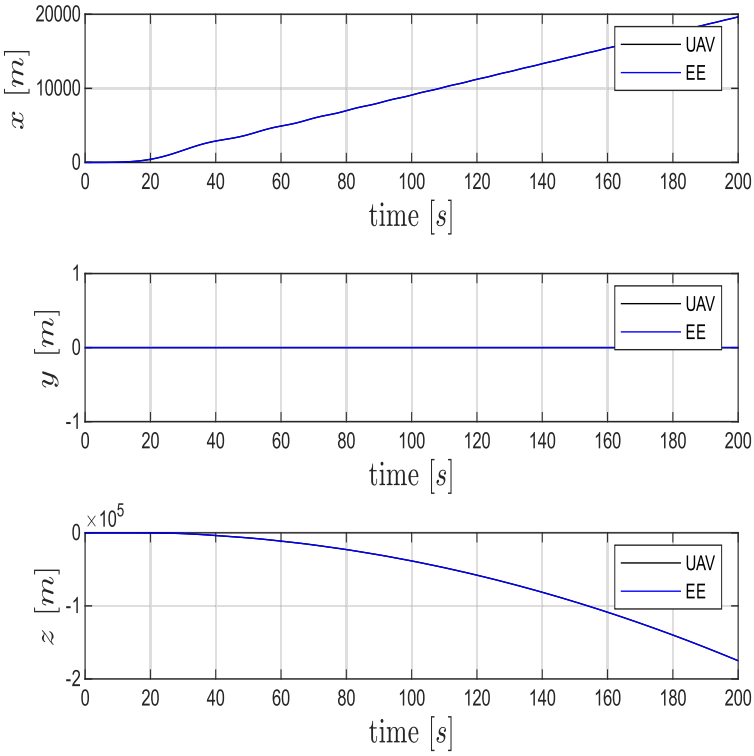

Figure 2.10: UAV and End Effector position with constant input torque τ_θ

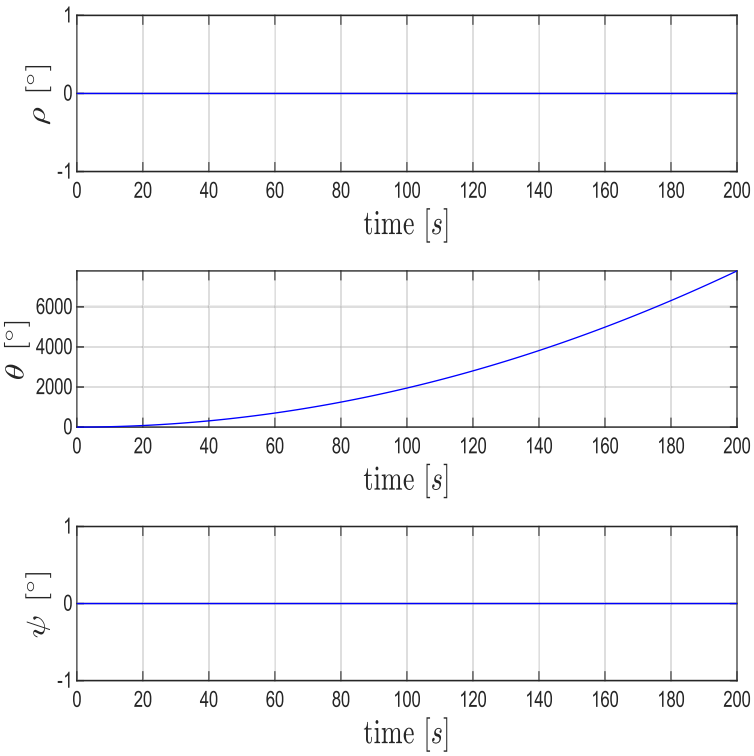

Figure 2.11: UAV Euler angles with constant input torque τ_θ

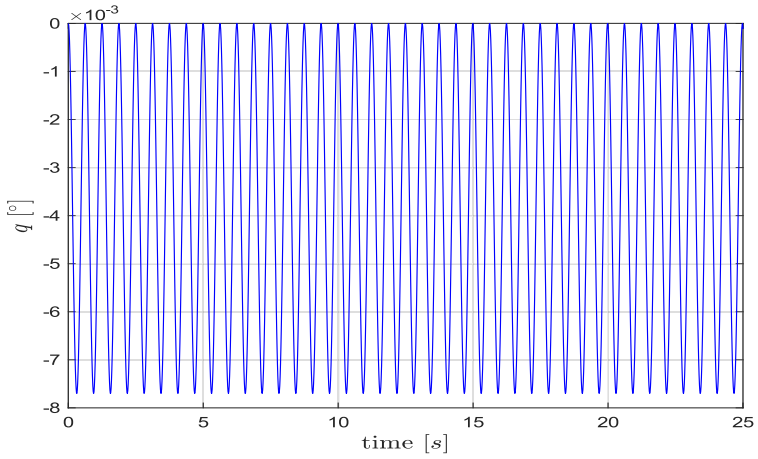

Figure 2.12: q coordinate with constant input torque τ_θ

Finally, the robot is provided with an input force f_z and a link torque $\tau_l = 0.001$ Nm. As previously mentioned, the rotation axis of the revolute joint coincides with axis $y_{\mathbb{B}}$. Therefore, the link rotation induced by the input torque produces a reaction torque τ_θ. Moreover, as the end effectormoves, the center of mass of the whole system changes, and the forces generated by the UAV propellers cause additional force torques in the UAV center of gravity. The UAV cartesian position and Euler angles can be observed in figures 2.13 and 2.14, respectively.

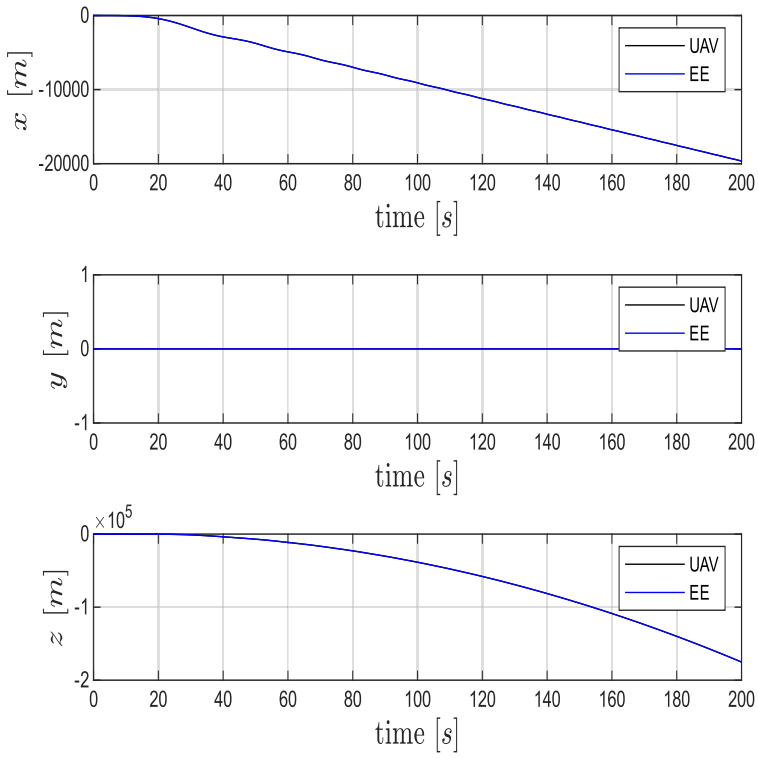

Figure 2.13: UAV and End Effector position with constant input torque τ_l

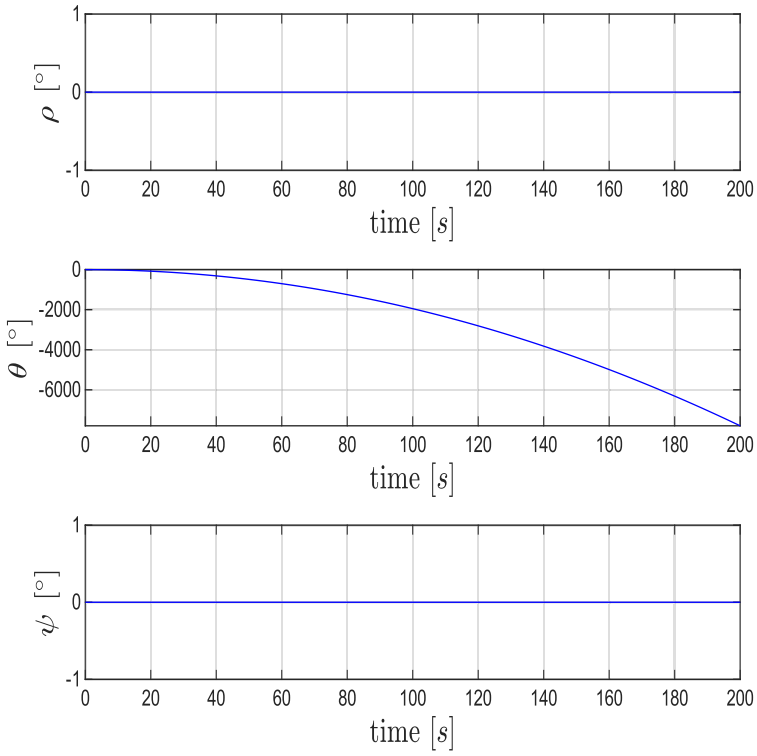

Figure 2.14: UAV Euler angles with constant input torque τ_l

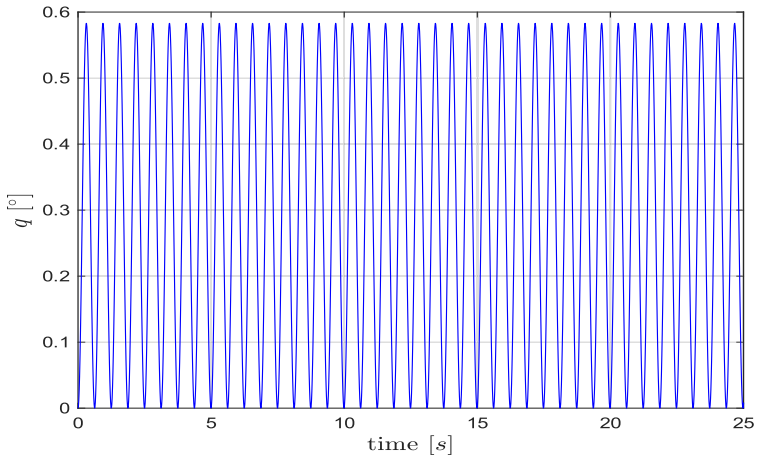

Figure 2.15: q coordinate with constant input torque τ_l

It can be noticed that the θ angle decreases quadratically, due to the constant reaction torque experienced by the UAV. The body frame rotation then causes a translation along the $-x$ axis, according to the model derived in 2.7. The input torque τ_l causes the link to oscillate around an equilibrium different from its rest position, as it can be seen from figure 2.15. Given the high frequency of the oscillations and their periodicity, only a window of 25s is showed for the sake of visibility in the last two tests.

2.3 FH MODEL

The flying hand studied in this work consists of a star-shaped, $\alpha\beta$ tilted hexarotor, with a passive manipulator consisting of a single link (the same as in the UAM), rigidly attached to the UAV center of gravity in such a way that the link is perpendicular to the floating base, with the end effector pointing downwards (see figure 1.1). As in most tilted multirotors, the β angle is the same for every propeller, and it is such that each rotor's z axis points outwards with respect to the drone's center of mass. On the other hand, the α_i angle for each propeller can be computed as $\alpha_i = (-1)^{i-1}\alpha$, meaning that consecutive rotors are facing each other in couples. Differently from the previou system, the link cannot move with respect to the aerial vehicle, however tilting the propellers provides greater actuation capabilities for the mobile platform, as it will be discussed later. Table 2.2 contains the physical parameters describing the case of study robot.

Name	**Symbol [unit]**	**Value**
UAV mass	m_B [Kg]	3.5
α tilt angle	α [°]	25
β tilt angle	β [°]	10
γ angle	γ [°]	60
Arm length	r [m]	0.358
Force coefficient	c_f [N/(rad/s)^2]	0.0015
Drag coefficient	c_τ [Nm/(rad/s)^2]	4.59×10^{-5}
UAV inertia tensor	I_B [kgm^2]	$\begin{pmatrix} 0.155 & 0 & 0 \\ 0 & 0.147 & 0 \\ 0 & 0 & 0.251 \end{pmatrix}$
Link mass	m_l [Kg]	0.29
Link length	l [m]	0.14
Link inertia tensor	I_l [kgm^2]	$\begin{pmatrix} 6.55 & 0 & 0 \\ 0 & 6.55 & 0 \\ 0 & 0 & 3.62 \end{pmatrix} \times 10^{-4}$

Table 2.2: FH parameters

It can be noticed that the physical parameters describing the FH are equal to those characterising the UAM, except for the propeller tilt angles. In fact, the two case of study robots differ only by their actuation properties, while their physical characteristics are the same.

2.3.1 *Dynamic Equations*

Since the link is rigidly connected to the drone and no relative motion between the two bodies is allowed, it is useful to model the system as a single rigid body. Consider then a new body fixed frame $\mathbb{B}' = \{x_{\mathbb{B}'}, y_{\mathbb{B}'}, z_{\mathbb{B}'}\}$, whith origin located at the center of mass of the overall system. Due to the symmetry of the structure, the x and y coordinates of the baricenter coincide with the x and y coordinate of the drone's CoM. The z coordinate in frame $\mathbb{B}$ can be computed as

$$z_{CoM} = \frac{m_B l + m_l \frac{l}{2}}{m_B + m_l} \tag{2.43}$$

The total inertia of the system can be computed exploiting the Huygens-Steiner theorem, by expressing the inertia tensors of both the UAV and the passive manipulator with respect to the baricenter of the entire robot, according to

$$I'_B = I_B + m_B(|d_B|^2 I_3 - d_{B,i} d_{B,j}) \tag{2.44}$$

$$I'_l = I_l + m_l(|d_l|^2 I_3 - d_{l,i} d_{l,j}) \tag{2.45}$$

where d_B and d_l represent the vectors connecting the full system center of mass and the drone and link center of mass, respectively. Then, the total inertia of the system is given by $I_{tot} = I'_B + I'_l$. By representing the robot orientation with the triplet of ZYX Euler angles in the current frame $\Phi = (\rho, \theta, \psi)^T$, the flying hand dynamic equations can be derived through the Newton-Euler formalism:

$$\begin{pmatrix} m_{tot} \ddot{p}_{B'} \\ I_{tot} \dot{\omega}^{B'}_{B'} \end{pmatrix} = \begin{pmatrix} R_{WB}(\Phi) f_c \\ \tau_c \end{pmatrix} - \begin{pmatrix} m_{tot} g \hat{e}_3 \\ \omega^{B'}_{B'} \times I_{tot} \omega^{B'}_{B'} \end{pmatrix} \tag{2.46}$$

where $p_{B'}$ denotes the position of the robot's center of mass in world coordinates, while $\omega^{B'}_{B'}$ represents the robot angular velocity with respect to body frame $\mathbb{B}'$ and f_c and τ_c are the UAV input forces and torques. Since the link is not actuated, the only inputs to the system are provided by the drone.
The end effector position can be retrieved from the robot's center of gravity position through a rigid trasformation:

$$p_e = p_{B'} - R_{WB}(\Phi) v \tag{2.47}$$

where v is the vector connecting the robot's CoM and the end effector. On the other hand, the end effector orientation coincides with the drone orientation, due to the fixed joint connecting the UAV base and the link.

2.3.2 *Actuation and Decoupling*

The hexarotor composing the flying hand is characterized by tilt angles $\alpha = 25°$, $\beta = 10°$. According to the decomposition presented in 2.1.2, it is possible to verify that such system is fully actuated. However, the presence of the link attached to the floating base generates coupling effects in the robot dynamics. In particular, the center of gravity of the system no longer coincides with the center of mass of the UAV, and the inertia of the robot is modified. Those changes do not affect the forces or drag torques applied by the drone, however, they influence the force torques exerted by the UAV. In fact, the new force torques can be computed as

$$\tau_i^{f\prime} = p_i' \times f_i \tag{2.48}$$

where f_i is the force generated by propeller i, and p_i' is the vector connecting the baricenter of the robot (origin of reference frame $\mathbb{B}'$) to the origin of the propeller reference frame $\mathbb{O}_i$. Then, the total input torque of the robot can be computed as

$$\tau_c = \sum_{i=1}^{6} \tau_i^d + \tau_i^{f\prime} = M'u \tag{2.49}$$

It can be verified that it still holds $rank((F^T, M'^T)^T) = 6$ and the robot maintains full actuation properties.
It is once again of interest to study the degrees of freedom of the robot in the operational space, i.e., the degrees of freedom of the end effector. By defining the operational space variables as $x_{e,FH} = (p_e^T, \Phi^T)^T$, it is possible to notice that the robot and end effector generalized coordinates coincide, up to a rigid transformation. Hence, the robot is fully actuated even with respect to the operational space variables.

2.3.3 *FH Model Validation*

To assess the validity of the flying hand model derived in the previous section, some simulations are performed. As for the UAM model, those tests consists of applying to the system some input forces and torques in open loop, for the duration of 200s, and observing the robot behaviour. The initial conditions for the robot are set as follows: the initial position for the robot center of gravity is $p_{B',0} = 0^T$, while its initial orientation is defined by $\Phi_0 = 0^T$. The simulations performed for the flying hand are similar to the ones considered for the UAM. First, the ability of the system to maintain hovering is tested by applying a body frame thrust force $f_z = (m_{tot} g)$ N. From figures 2.16 and 2.17, it is evident that the robot hovers at its initial condition. It can be noticed that the end effector z coordinate is negative, and the distance from the system center of gravity is consistent with 2.43.

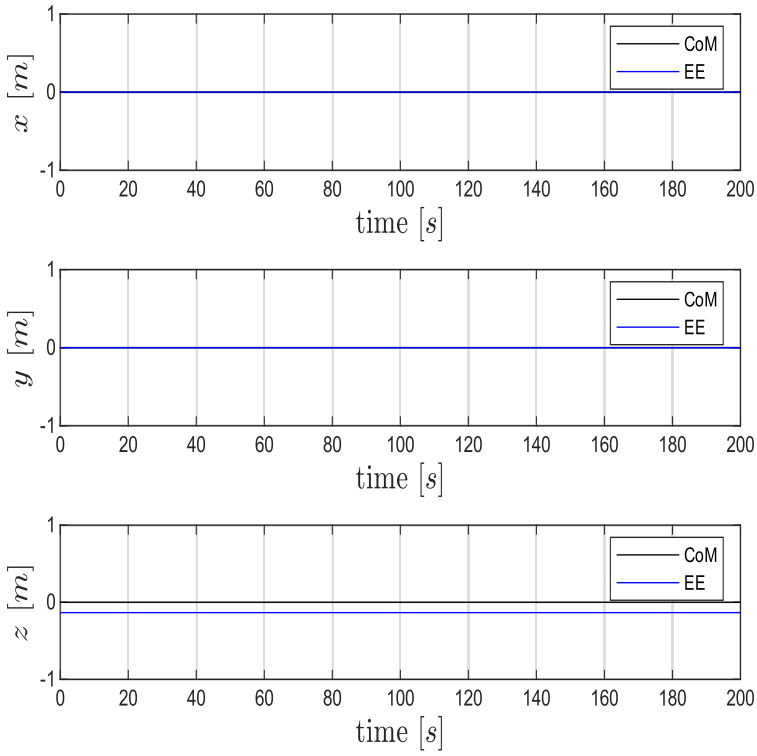

Figure 2.16: FH CoM and End Effector position at hovering

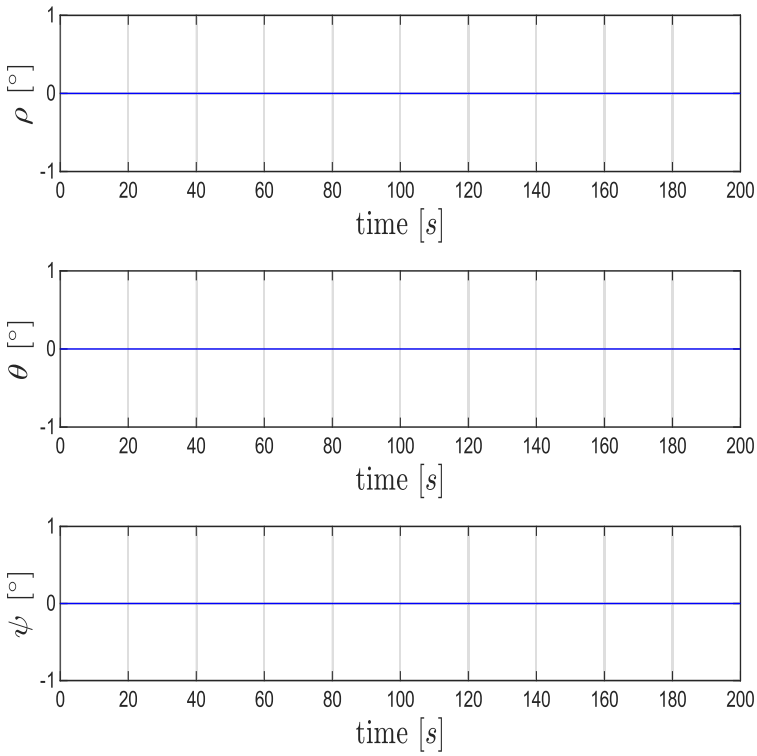

Figure 2.17: FH Euler angles at hovering

Then, given that the flying hand is fully actuated, the robot is able to move freely in the three dimensional space without changing its orientation. Two separate tests are performed, to observe motion along the x and y direction. To obtain cartesian motion along the desired direction, body frame forces $f_x = 1$ N and $f_y = 1$ N are exerted on the system, respectively. Moreover, in order to maintain constant altitude, the thrust force f_z is applied to the system to compensate the robot weight force. The input torque is set to $\tau_c = 0^T$.
Figures 2.18 and 2.19 show the robot response to the control input force $f_c = (f_x, 0, f_z)^T$. It can be seen that the robot's cente of gravity shows constant acceleration along the x direction due to the constant force generated by the propellers, while the y and z coordinates remain constant. The Euler angles remain at their initial value throughout the simulation. Analogous results can be observed in figures 2.20 and 2.21, which describe the flying hand response to the input force $f_c = (0, f_y, f_z)^T$. With such choice of inputs, a uniformly accelerated motion along the y direction can be observed, while the orientation remains unchanged.
For the next test, a torque $\tau_\psi = 0.001$ Nm is applied, as well as the thrust force f_z. The flying hand is expected to hover at its initial position while changing its heading direction. Indeed, from figure 2.22 and 2.23, it can be seen that the baricenter coordinates are constant in time, the ρ and θ are fixed while the ψ angle increases quadratically due to the constant torque generated by the propellers.

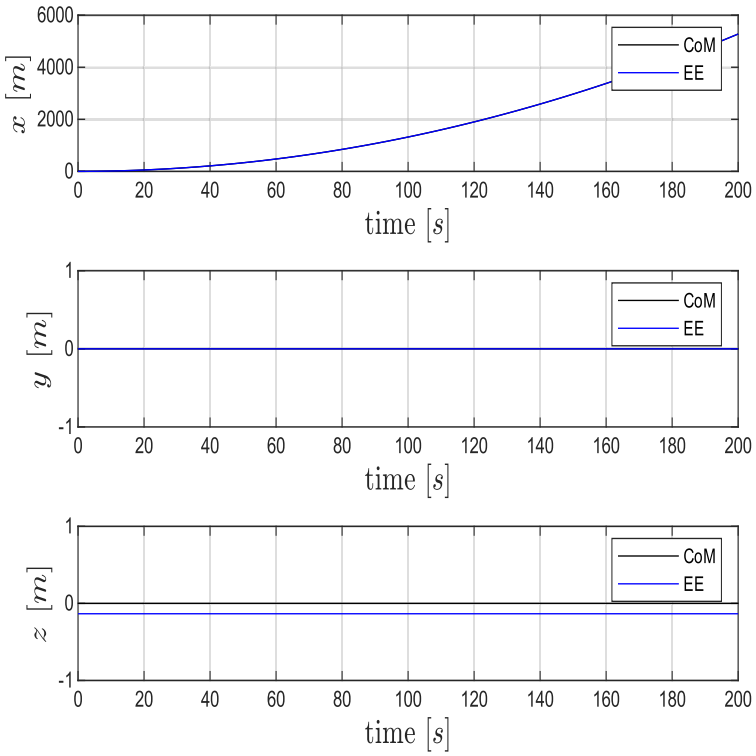

Figure 2.18: FH and End Effector position with constant input force f_x

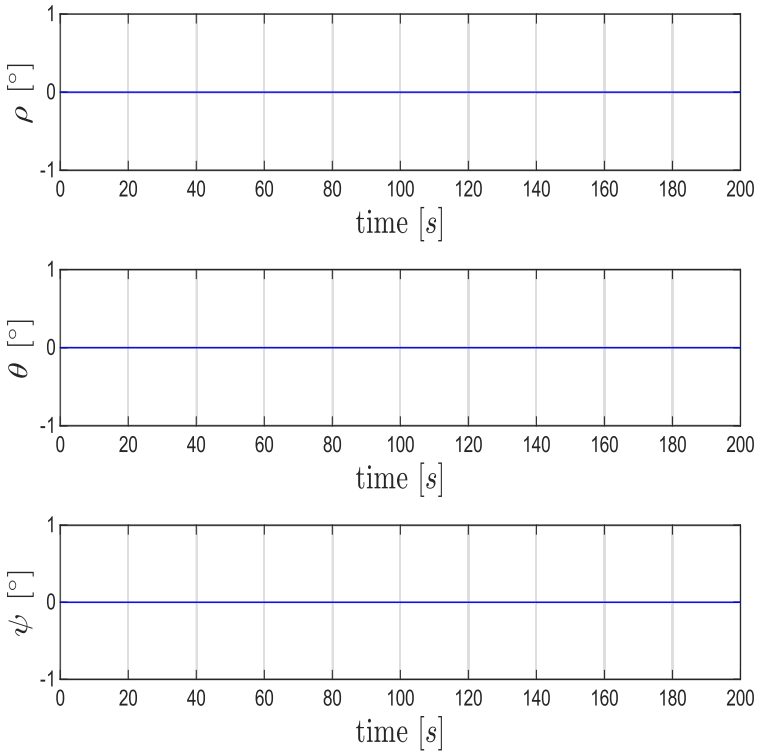

Figure 2.19: UAV Euler angles with constant input force f_x

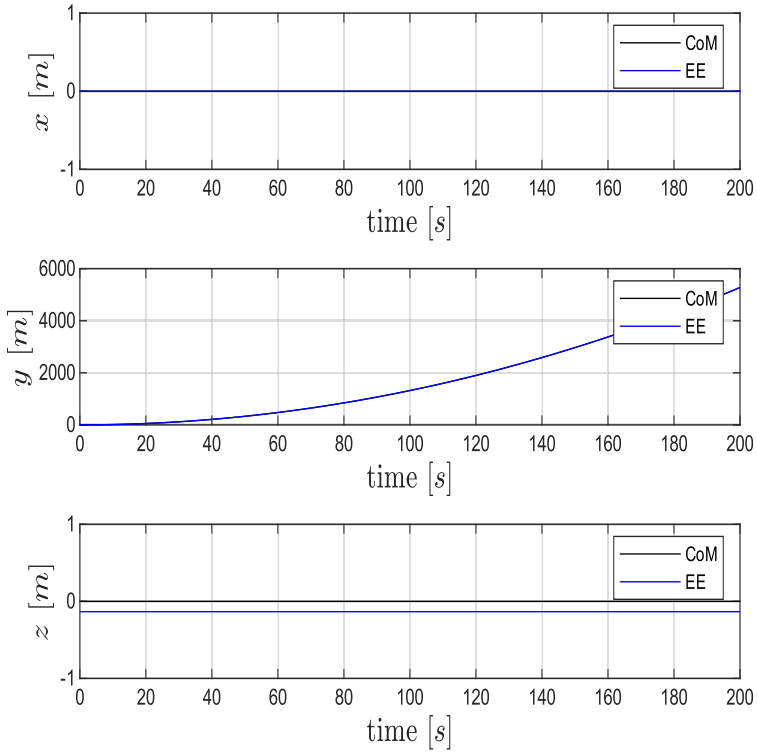

Figure 2.20: FH and End Effector position with constant input force f_y

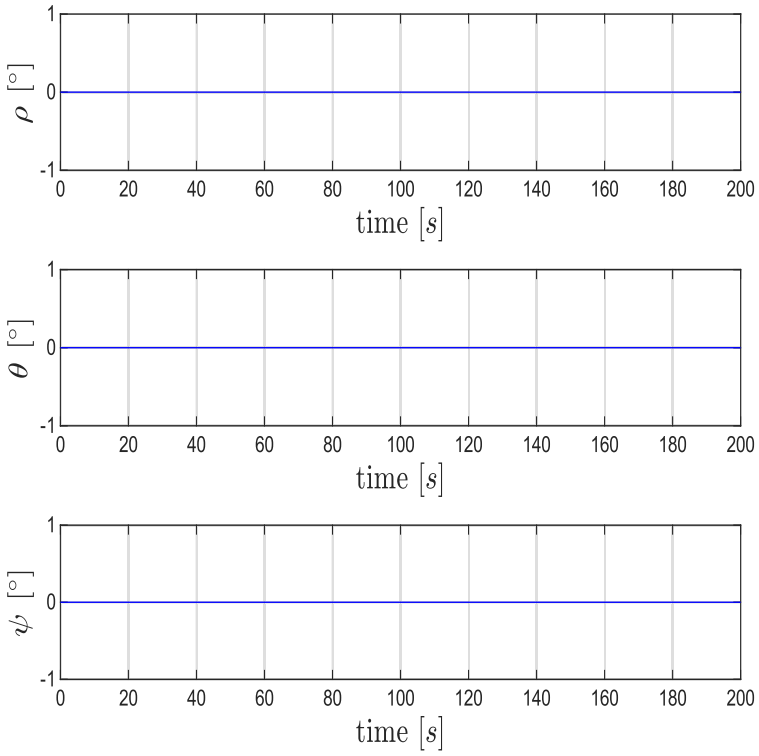

Figure 2.21: UAV Euler angles with constant input force f_y

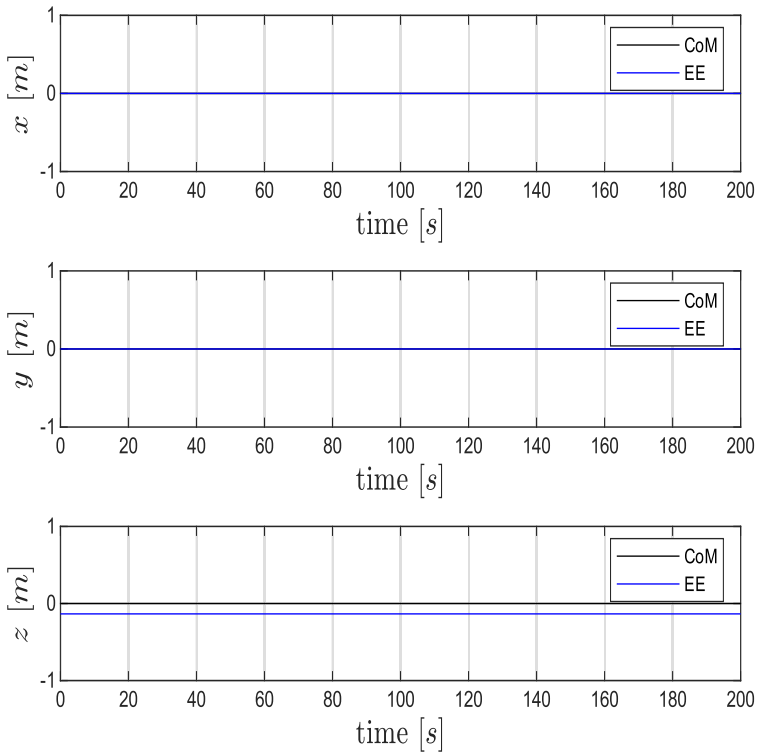

Figure 2.22: FH and End Effector position with constant input torque τ_ψ

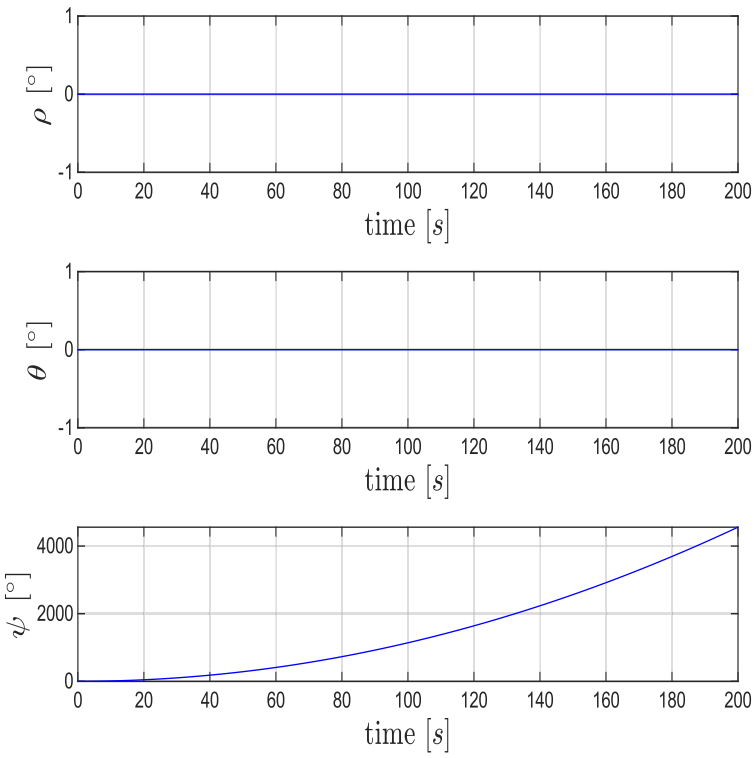

Figure 2.23: UAV Euler angles with constant input torque τ_ψ

Finally, the flying hand responses to ρ and θ rotations are observed. In order to achive such rotations, input torques $\tau_\rho = 0.001$ Nm and $\tau_\theta = 0.001$ Nm are applied to the robot, respectively, as well as the body frame thrust force f_z. Since the input signals are applied in open loop, the system is expected to move along the x and y direction, based on the torque considered, and lose height due to the body frame rotation, according to the model derived in 2.46.
Figures 2.24 and 2.25 show the system response to input torque τ_ρ and force f_z. It can be notice that such response is identical to that of the UAM reported in figures 2.7 and 2.8. In fact, since the link revolute joint allows relative motion between the two bodies only about the $y_{\mathcal{B}}$ frame, when the motion does not involve θ rotations the two systems are equivalent. It can be noticed that, due to the constant torque τ_ρ generated by the propellers, the ρ angle grows as a quadratic function.
Figures 2.26 and 2.27 show the robot response to the input torque τ_θ and thrust force f_z. The constant torque produced by the motors causes a body frame θ rotation. Since the input force and torque are applied in open loop, the body frame rotation is not compensated and causes a reduction in the vertical component of the thrust force, resulting in the system losing height. Moreover, a force component along the x direction arises, causing motion along this axis.

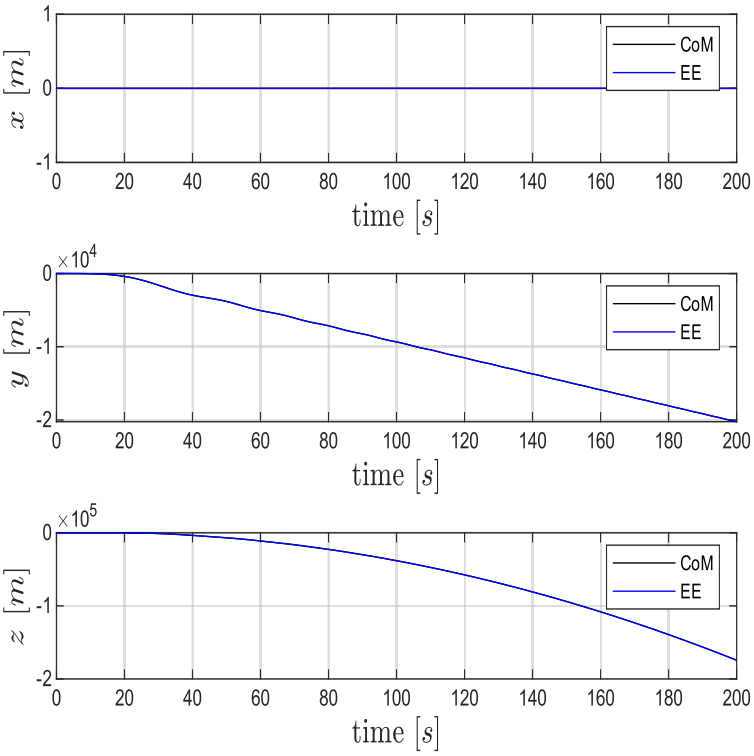

Figure 2.24: FH and End Effector position with constant input torque τ_ρ

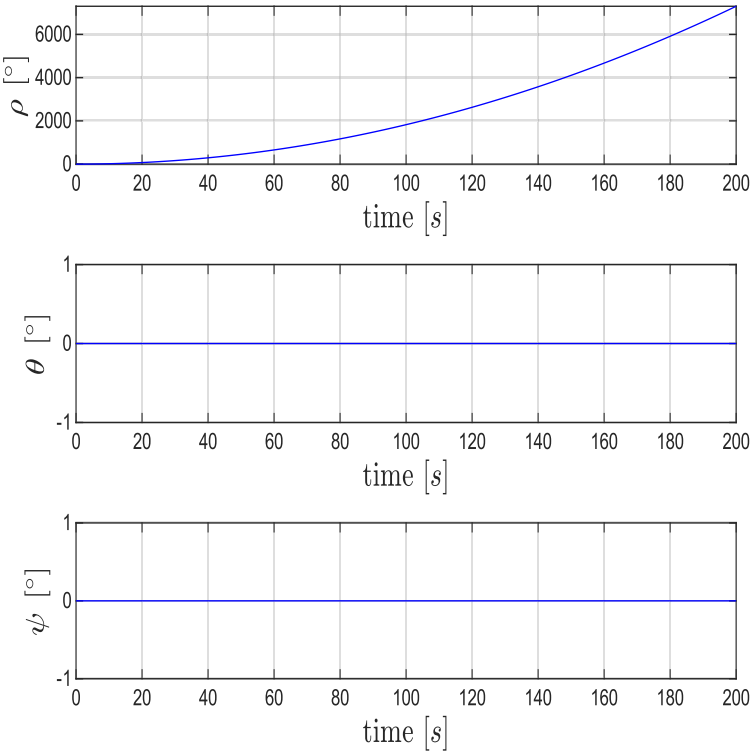

Figure 2.25: UAV Euler angles with constant input torque τ_ρ

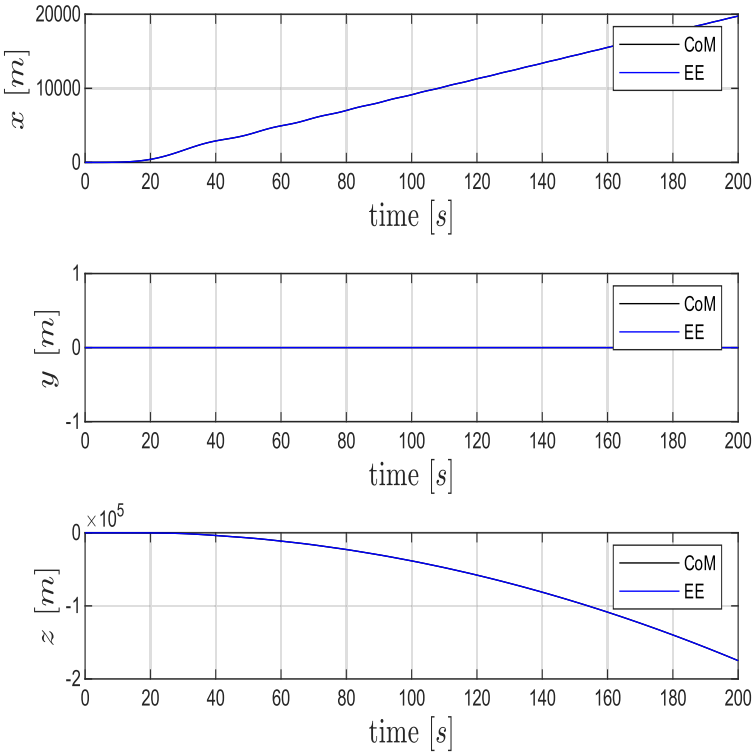

Figure 2.26: FH and End Effector position with constant input torque τ_θ

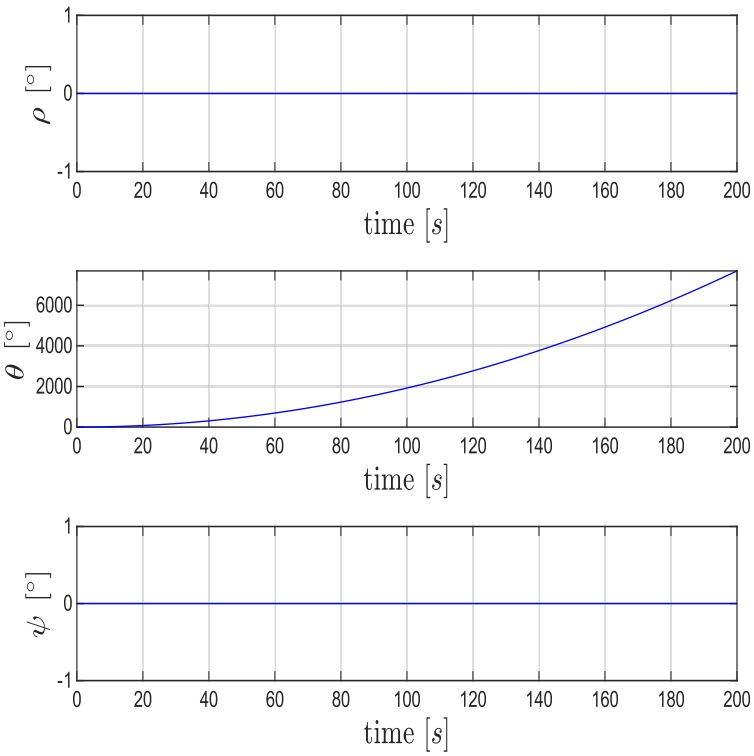

Figure 2.27: UAV Euler angles with constant input torque τ_θ

3

CONTROL DESIGN

In this chapter, control schemes are designed for the two case of study systems (UAM and FH). The goal of those controllers is to achieve asymptotic tracking of a simple joint space trajectory, similar for both robots. The trajectory is defined as follows:

- **vertical rising**: the robot vertically ascends up to $2m$ of height;
- x **translation**: the robot maintains a constant height of $2m$ while moving along the x direction, for a total length of $2m$. For the UAM, the q coordinate remains at $0°$;
- **end effector positioning**: for the UAM robot, the drone hovers at its current position, while the link rotates of $-30°$, while for the FH robot the full system rotates of $-30°$;
- y **translation**: maintaining the current end effector orientation, the robot slides along the y direction for a length of $2m$.

The translations are defined for the UAV's center of mass for the UAM, while for the FH they refer to the full robot's baricenter. Each phase of the trajectory takes $7s$ to complete. Those joint space trajectories yield similar end effector trajectories for both the UAM and the FH. For the latter, it has already been established that the end effector variables coincides with the full robot variables, up to a rigid transformation, therefore it is only necessary to design a controller for the flying base. On the other hand, for the UAM two controllers are designed, one for the UAV and the other for the robotic arm.

3.1 UAM TRAJECTORY TRACKING

3.1.1 *Underactuated Platform Control*

The drone which composes the UAM is underactuated. In fact, as previously stated, it holds $rank((F^T, M^T)^T) = 4$. Therefore, it is possible to independently control 4 UAV generalized coordinates. In this work, the choice adopted is to control the cartesian position $p_B = (x, y, z)^T$ and the robot heading direction ψ. The thrust force f_z induced by the propellers is used to control the altitude, i.e. the z coordinate, while

the torques need to be designed in order to control the x, y and ψ coordinates. In order to do so, a map between the cartesian position and attitude information needs to be derived, assuming a desired trajectory $(x_d, y_d, z_d, \psi_d)^T$ is given. Considering the translational dynamics and linearizing around hovering condition ($\rho \approx 0$, $\theta \approx 0$), it holds

$$\ddot{p}_B = \begin{pmatrix} (\sin\psi_d \rho + \cos\psi_d \theta)\frac{f_z}{m_B} \\ (-\cos\psi_d \rho + \sin\psi_d \theta)\frac{f_z}{m_B} \\ \frac{f_z}{m_B} - g \end{pmatrix} \tag{3.1}$$

By defining the position error $e_p = p_{B,d} - p_B$, where $p_{B,d} = (x_d, y_d, z_d)^T$ is the desired drone cartesian trajectory, and imposing some desired error dynamics according to

$$\ddot{e}_p + K_D \dot{e}_p + K_P e_p = 0 \tag{3.2}$$

the reference position can be obtained as

$$\ddot{p}_{B,r} = \ddot{p}_{B,d} + K_D \dot{e}_p + K_P e_p \tag{3.3}$$

which yields asymptotic convergence to $e_p = 0$ as long as K_D, $K_P \in \mathbb{R}^{3\times3}$ are diagonal positive definite control matrices. Then, the Euler angles reference trajectory $\Phi_r = (\rho_r, \theta_r, \psi_r)^T$ can be obtained through the map 3.1, by substituting the cartesian acceleration $\ddot{p}_B$ with the reference acceleration $\ddot{p}_{B,r}$ derived in 3.3:

$$\begin{cases} \rho_r = (\sin\psi_d \ddot{x}_{B,r} - \cos\psi_d \ddot{y}_{B,r}) g^{-1} \\ \theta_r = (\cos\psi_d \ddot{x}_{B,r} + \sin\psi_d \ddot{y}_{B,r}) g^{-1} \\ \psi_r = \psi_d \end{cases} \tag{3.4}$$

The altitude control law can then be written as

$$f_z = \frac{m_B + m_l}{\cos\rho\cos\theta}(\ddot{z}_{B,r} + g) \tag{3.5}$$

for the full UAM. Considering 2.10, the rotation dynamics in 2.7 can be expressed in terms of the Euler angles as

$$\ddot{\Phi} = T(\Phi)^{-1}(I_B^{-1}\tau_c - I_B^{-1}(T(\Phi)\dot{\Phi} \times I_B T(\Phi)\dot{\Phi}) - \dot{T}(\Phi)\dot{\Phi}) \tag{3.6}$$

and the attitude control law can be derived as

$$\tau_c = I_B T(\Phi)(\ddot{\Phi}_r + K_{D,\Phi}\dot{e}_\Phi + K_{P,\Phi} e_\Phi) + T(\Phi)\dot{\Phi} \times I_B T(\Phi)\dot{\Phi} + I_B \dot{T}(\Phi)\dot{\Phi} \tag{3.7}$$

where $e_\Phi = \Phi_r - \Phi$ is the attitude error, $K_{D,\Phi}$ and $K_{P,\Phi} \in \mathbb{R}^{3\times3}$ are positive definite control matrices. Then, the attitude error dynamics can be expressed as

$$\ddot{e}_\Phi + K_{D,\Phi}\dot{e}_\Phi + K_{P,\Phi}e_\Phi = 0 \tag{3.8}$$

which yields asymptotic convergence to $e_\Phi = 0$ if $K_{D,\Phi} \succ 0$ and $K_{P,\Phi} \succ 0$.

3.1.2 *Robotic Arm Control*

Given a desired link coordinate trajectory in terms of q_d, $\dot{q}_d$ and $\ddot{q}_d$, the control law for the robotic manipulator consists of a feedback linearization term to compensate the nonlinear arm dynamics, plus a PID and feedforward actions. This strategy is usually referred to as computed torque control. Considering the Euler-Lagrange dynamic model 2.26, the link input can therefore be written as

$$\tau_l = e_7^T B(\xi)e_7(\ddot{q}_d + K_{D,q}\dot{e}_q + K_{P,q}e_q) + e_7^T(C(\xi,\dot{\xi}))\dot{\xi} + g(\xi)) \tag{3.9}$$

where $e_q = q_d - q$, $K_{D,q} > 0$ and $K_{P,q} > 0$ are control gains, and $e_7 \in \mathbb{R}^7$ is the 7-th canonical vector $\in \mathbb{R}^7$. The vector e_7 is used to select the components of the $B(\xi)$ and $C(\xi,\dot{\xi})$ matrices and of the vector $g(\xi)$ of the dynamical model 2.26 relative to the link coordinate q. Moreover, the scalar contribution $e_7^T C(\xi,\dot{\xi})\dot{\xi}$ contains the Coriolis and Centrifugal terms due to the UAV velocities, and therefore it is a form of compensation of the coupling effects between the flying base and the robotic manipulator.

3.2 UAM DYNAMIC COMPENSATION

As it will be seen in the next chapter, the UAM controllers achieve poor tracking performances. In particular, during the second phase of the trajectory (translation in the x direction), the system presents a large tracking error e_q, with a peak of $e_q = 3.96°$. Moreover, in the last phase of the trajectory, a steady state cartesian tracking error x component equal to $e_{p,x} = 3.7cm$ is present. Finally, when the system is moving along the y direction, the ψ generalized coordinate exhibits a divergent behaviour. In this section, some compensation strategies are designed in order to improve the case of study robot's tracking performances.

3.2.1 *Slower Trajectory*

The cause of the large generalized q coordinate tracking error is attributed to the link dynamics not being fast enough to cope with the reaction torques generated by the UAV. Then, an improvement can be obtained by allowing more time for the robot to complete this phase of the trajectory. The new desired translation requires 20s to

complete. By increasing the required time, the desired velocities and acceleration are reduced, resulting in smaller reaction torques. The tracking error e_q is then reduced. The errors observed in the last phase of the trajectory are attributed to the center of gravity of the robot not remaining aligned with the UAV's baricenter. Then, the forces applied by the drone's propeller generate additional force torques, which the controller is not able to compensate. Those effects are not related to the time requirements imposed by the desired trajectory. Nevertheless, this phase of the trajectory is also slowed down to 20s. With this change, the ψ error is reduced, although its behaviour is still divergent. However, with the slower trajectory, the control effort required is significantly decreased.

3.2.2 *Integral Action*

Although the slower trajectory results in lower tracking errors and smaller control signals, it does not solve the convergence issues previously discussed. A first attempt to solve those problems consists of adding integral actions in the control laws, in order to reject constant disturbances.
The error dynamics for the cartesian position of the drone's center of gravity can then be expressed as

$$\ddot{e}_p + K_D \dot{e}_p + K_P e_p + K_I \int e_p \, dt = 0 \tag{3.10}$$

where asymptotic stability to e_p is guaranteed if $0 < K_{I,i} < K_{P,i} K_{D,i}$, according to the Routh Criterion. The position reference is then retrieved as

$$\ddot{p}_{B,r} = \ddot{p}_{B,d} + K_D \dot{e}_p + K_P e_p + K_I \int e_p \, dt \tag{3.11}$$

The integral control gain matrix is chosen as $K_I = diag(K_{I,x}, K_{I,y}, 0)$.
Concerning the attitude dynamics, the ρ and θ angles inherit the control action from $\ddot{p}_r$ through the map 3.1, while the ψ dynamics is modified to include the integral term. Then, the attitude control law can be rewritten as

$$\begin{aligned} \tau_c = I_B T(\Phi)(\ddot{\Phi}_r + K_{D,\Phi} \dot{e}_\Phi + K_{I,\Phi} \int e_\Phi \, dt) \\ + T(\Phi)\dot{\Phi} \times I_B T(\Phi)\dot{\Phi} + I_B \dot{T}(\Phi)\dot{\Phi} \end{aligned} \tag{3.12}$$

where $K_{I,\Phi} = diag(0, 0, K_{I,\psi})$.
Finally, the integral action is also added in the robotic arm control law 3.9, which is modified according to

$$\tau_l = e_7^T B(\xi) e_7 (\ddot{q}_d + K_{D,q} \dot{e}_q + K_{P,q} e_q + K_{I,q} \int e_q \, dt) + e_7^T (C(\xi, \dot{\xi})\dot{\xi} + g(\xi)) \tag{3.13}$$

where $K_{I,q}$, needs to be chosen to satisfy the Routh criterion to guarantee asymptotic convergence to $e_q = 0$.

3.2.3 *Model-Based Dynamic Compensation*

So far, performance improvements have been achieved by means of relaxations in the trajectory requirements, and the addition of the integral action in the control laws. None of this approaches exploits the coupled dynamical model derived in 2.26. By inspecting the structures of matrices $B(\xi)$ and $C(\xi, \dot{\xi})$ and of the vector $g(\xi)$, it can be observed how the robotic manipulator dynamics influences the UAV dynamics and vice versa. Since the coupling equations are known, it is possible to compensate them by allowing the drone access to the current q coordinate information. It is important to notice that this approach has already been implemented in the link control 3.9. The same strategy is applied to the attitude controller, introducing the information on the angular position and velocity of the link and maintaining the integral action in the ψ dynamics. The attitude control law can then be rewritten as

$$\begin{aligned}\tau_c = I_B T(\Phi)(\ddot{\Phi}_r + K_{D,\Phi}\dot{e}_\Phi + K_{P,\Phi}e_\Phi + K_{I,\Phi}\int e_\Phi dt) + T(\Phi)\dot{\Phi} \times I_B T(\Phi)\dot{\Phi} \\ + I_B \dot{T}(\Phi)\dot{\Phi} + R_{WB}(\Phi)^T T(\Phi)^{-T} S_\Phi (C(\xi, \dot{\xi})e_7\dot{q} + g(\xi))\end{aligned} \tag{3.14}$$

where $S_\Phi = (0_{3\times3}, I_3, 0_{3\times1})$ is a selection matrix isolating the fourth to sixth rows of $C(\xi, \dot{\xi})$ and $g(\xi)$. The contributions $S_\Phi C(\xi, \dot{\xi})e_7\dot{q}$ and $S_\Phi g(\xi)$ in 3.14 aim at exactly compensating the UAV torques due to the manipulator velocity and the gravity contribution. No integral action is needed in the UAV cartesian error dynamics equation.

3.3 FH TRAJECTORY TRACKING

3.3.1 *Fully Actuated Platform Control*

Concerning the flying hand robot, the actuation of the system is entirely provided by the UAV platform, since the manipulator is rigidly connected to it. The hexarotor which composes the FH is fully actuated, so it is possible to independently control the full 6 DoFs robot pose. Assuming a reference trajectory $(p_{B',d}, \Phi_d)^T$ is given, the FH dynamical model 2.46 can be rewritten as

$$\begin{pmatrix} \ddot{p}_{B'} \\ \dot{\omega}^{\mathbb{B}'}_{\mathbb{B}'} \end{pmatrix} = \begin{pmatrix} \frac{1}{m_{tot}} R_{WB}(\Phi) f_c \\ I_{tot}^{-1} \tau_c \end{pmatrix} - \begin{pmatrix} g\hat{e}_3 \\ \omega^{\mathbb{B}'}_{\mathbb{B}'} \times I_{tot} \omega^{\mathbb{B}'}_{\mathbb{B}'} \end{pmatrix} \tag{3.15}$$

Then, by choosing as input force

$$f_c = m_{tot} R_{WB}(\Phi)^T (\ddot{p}_{B',d} + K_{D,\alpha\beta}\dot{e}_p + K_{P,\alpha\beta} e_p + g\hat{e}_3) \tag{3.16}$$

where $e_p = p_{B',d} - p_{B'}$ is the position error, $K_{P,\alpha\beta}$ and $K_{D,\alpha\beta}$ are control matrices, the position error dynamics can be expressed as

$$\ddot{e}_p + K_{D,\alpha\beta}\dot{e}_p + K_{P,\alpha\beta} e_p = 0 \tag{3.17}$$

Assuming $K_{D,\alpha\beta}$ and $K_{P,\alpha\beta}$ to be positive definite, asymptotic convergence to $e_p = 0$ is guaranteed. Concerning the attitude dynamics, recalling 2.10, the equations can be rewritten in terms of the Euler angles as in 3.6, substituting I_B with I_{tot}. The control law can therefore be expressed as

$$\begin{aligned}\tau_c = {} & I_{tot} T(\Phi)(K_{D,\Phi\alpha\beta}\dot{e}_\Phi + K_{P,\Phi\alpha\beta} e_\Phi) + T(\Phi)\dot{\Phi} \times I_{tot} T(\Phi)\dot{\Phi} \\ & + I_{tot}\dot{T}(\Phi)\dot{\Phi}\end{aligned} \tag{3.18}$$

and, assuming $\ddot{\Phi}_d = 0$, the attitude error dynamics is described by

$$\ddot{e}_\Phi + K_{D,\Phi\alpha\beta}\dot{e}_\Phi + K_{P,\Phi\alpha\beta} e_\Phi \tag{3.19}$$

therefore asymptotic convergence to $e_\Phi = 0$ is guaranteed as long as $K_{D,\Phi\alpha\beta} \succ 0_{3\times3}$ and $K_{P,\Phi\alpha\beta} \succ 0_{3\times3}$. The condition $\ddot{\Phi}_d = 0$ can be ensured while designing the robot trajectory, e.g. by deriving the reference Euler angles through linear interpolation between a starting configuration and a desired final configuration.

4

SIMULATIONS AND RESULTS

In this chapter, the control parameters tuning is described, then the responses to the desired trajectories obtained by employing the control strategies previously described are showed, and some performance indexes are considered to analyze the controllers efficiency.

4.1 PLATFORMS TUNING

The control strategies described in chapter 3 are implemented on the single UAV platforms at first. So, the total masses and inertias are substituted with the drone masses and inertias. The goal is to tune the contol parameters so that the two aerial vehicles have similar responses to the same set point. Table 4.1 contains the control gains which yield similar responses for the two systems.

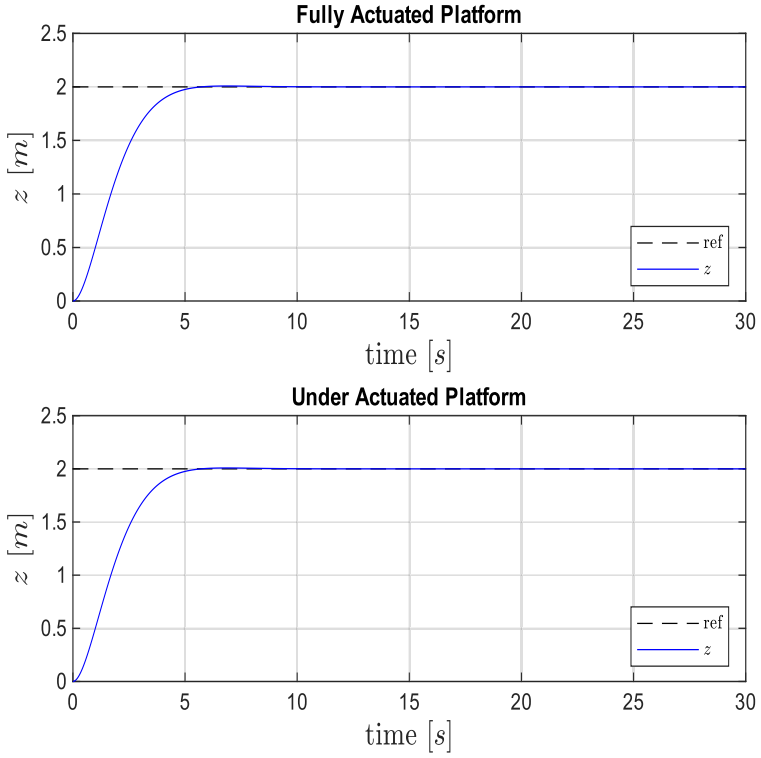

Figure 4.1: UAVs responses to a 2m step reference along z

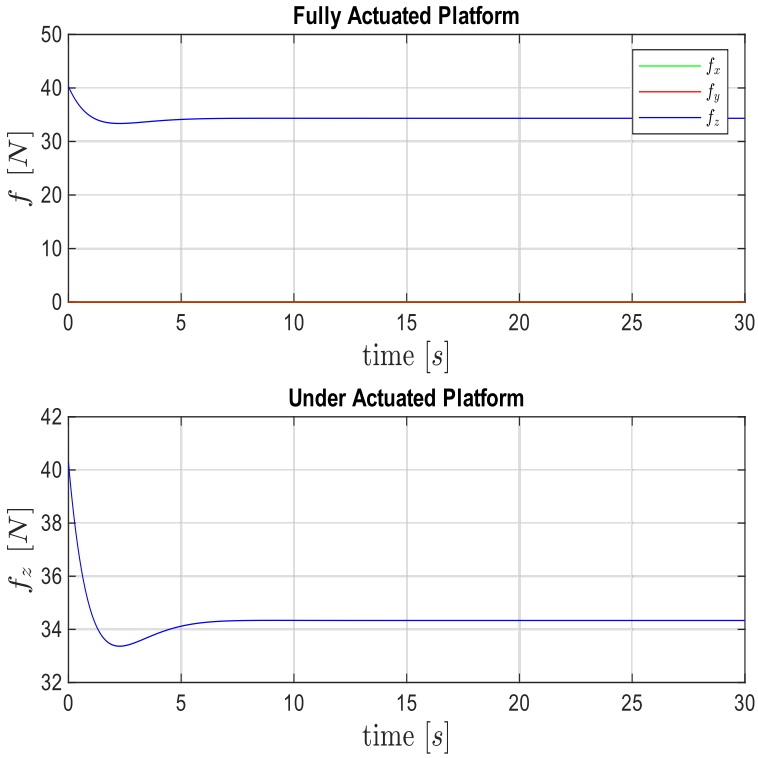

Figure 4.2: UAVs force inputs for a 2m step reference along z

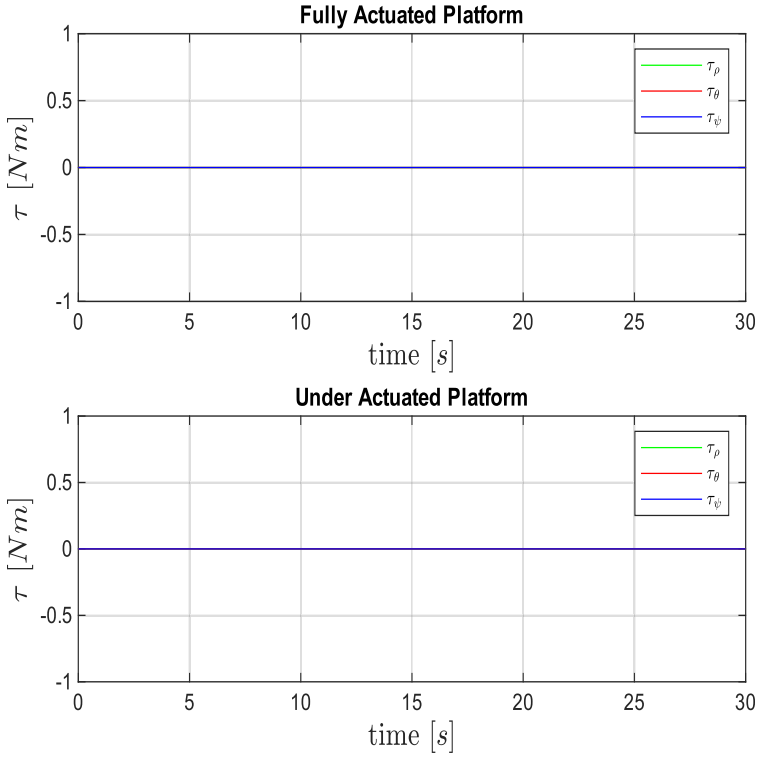

Figure 4.3: UAVs torque inputs for a 2m step reference along z

For the altitude dynamics, a step reference z_{ref} of 2m of height is considered, since both the UAM and FH trajectory require a 2m rise. From figure 4.1 it can be noticed that the responses of the fully actuated drone and of the under actuated one are identical. This is to be expected, since for the altitude dynamics the control laws for the two platforms are identical, and the two systems are charachterized by the same mass and inertia tensor. The responses are charachterized by a rise time $t_{r,1\%} = 5.08s$ and an overshoot of 0.415%. From figures 4.2 and 4.3, the input forces and torques required to reach the desired set point can be observed.
For the ψ dynamics, a step reference of 180° is considered, while the cartesian position reference is set at $\mathbf{o}$. As in the previous case, since the tilted and coplanar drones' closed loop ψ dynamics are equivalent, the responses are expected to be equal. From figure 4.4, it can be seen that the two systems' responses are in fact identical. With the control gains reported in 4.1, a rise time $t_{r,1\%} = 4.84s$ and an overshoot of 0.79% are achieved. The corresponding control signals can be observed in figures 4.5 and 4.6. A constant thrust force along the z direction is required to maintain constant altitude, while the torque τ_ψ produces the desired rotation.

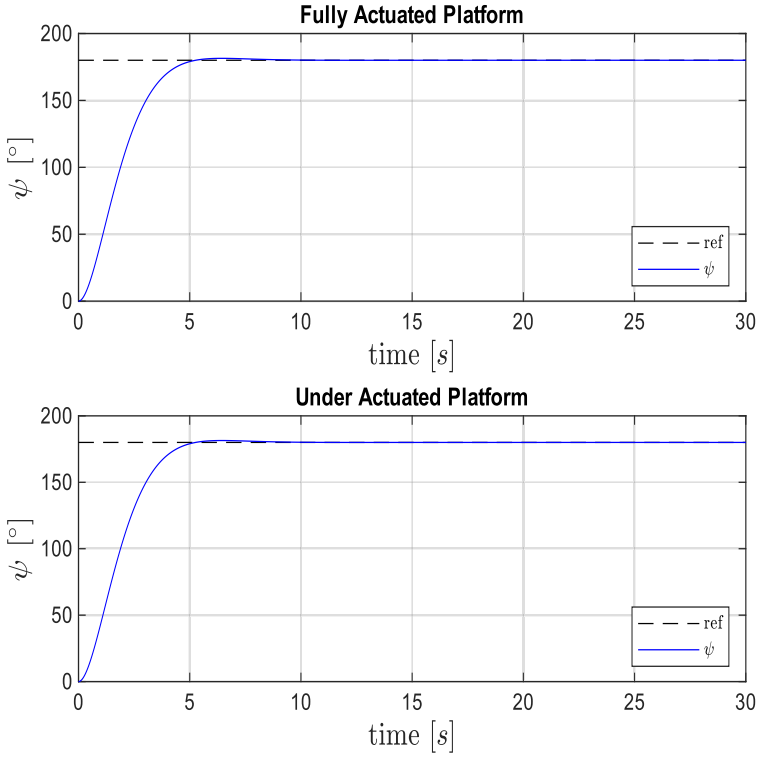

Figure 4.4: UAVs responses to a 180° ψ step reference

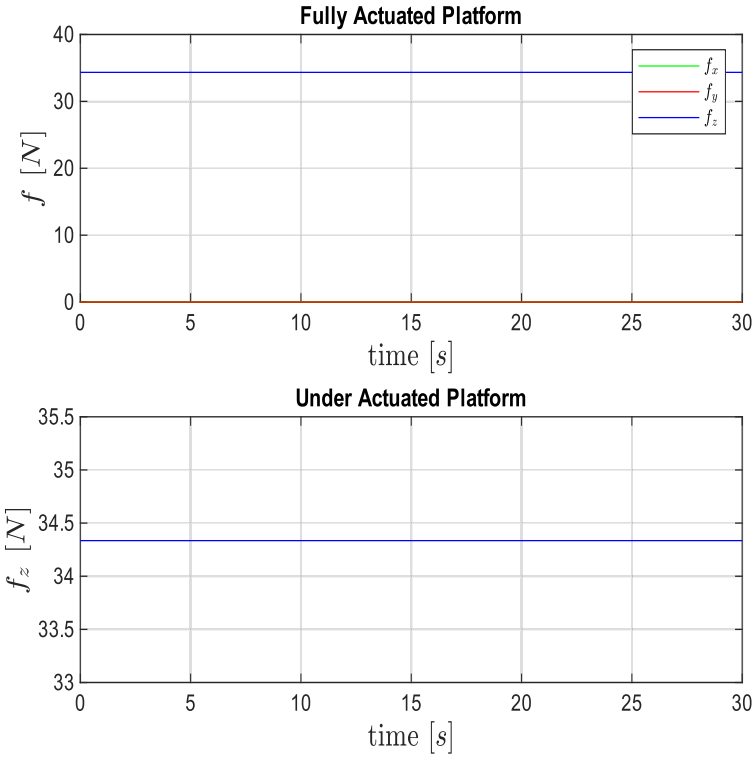

Figure 4.5: UAVs force inputs for a 180° ψ step reference

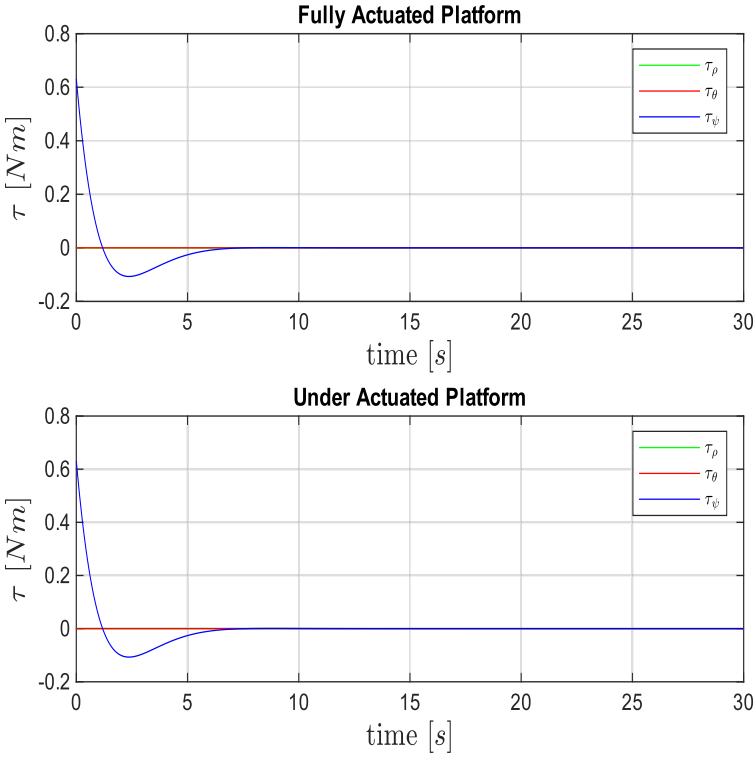

Figure 4.6: UAVs torque inputs for a 180° ψ step reference

A step reference of 2m is considered for the x dynamics, while the other controlled variables are set to **0**. Differently from the altitude and ψ dynamics, the closed loop behaviour for the x coordinate will in general be different between the two systems. The control parameters are tuned to make the robots responses as similar as possible. The step responses are showed in figure 4.7. The fully actuated platform shows a rise time $t_{r,1\%} = 5.56s$ and an overshoot of 0.12%, while the underactuated platform is charachterised by $t_{r,1\%} = 5.71s$ and no overshoot. From gigures 4.8 and 4.9, the required control signals can be observed. Due to its full actuation properties, the tilted hexacopter is able to maintain constant height and move in the x direction without applying any torque. On the other hand, the coplanar hexacopter must perform a θ rotation to move along the x direction. Due to this rotation, the vertical component of the body frame thrust force is reduced, and the robot needs to exert a higher force during the transient in order to maintain constant height.
Similar results can be observed in figures 4.10, where a y coordinate step reference of 2m is considered, and the other controlled variables are set to 0^T. The fully actuated platform's response is charachterized by $t_{r,1\%} = 5.56s$ and overshoot of 0.12%, whereas the coplanar drone's response shows $t_{r,1\%} = 5.7s$ and no overshoot. The corresponding input signals can be observed in figures 4.11 and 4.12.

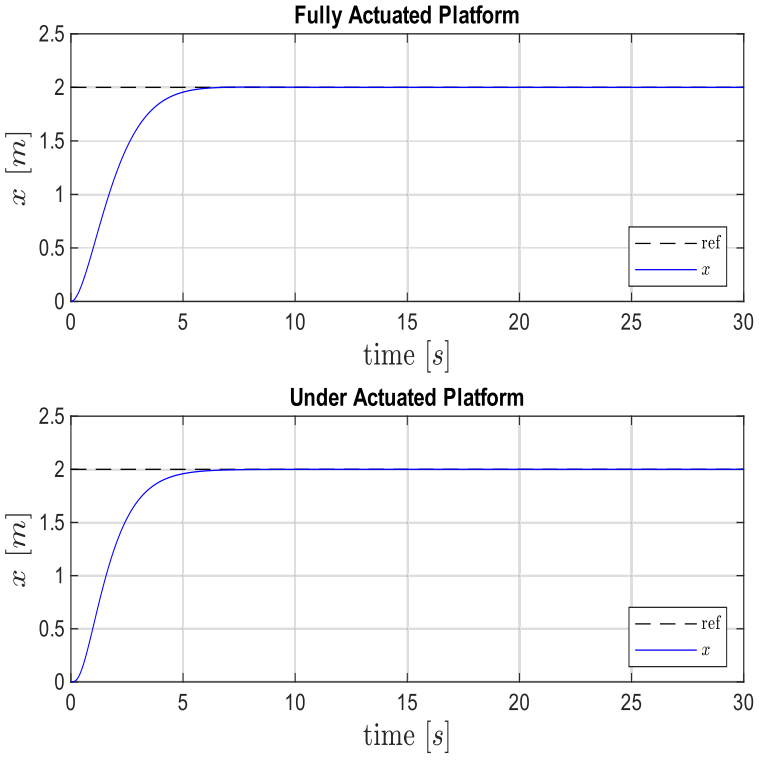

Figure 4.7: UAVs responses to a 2m step reference along x

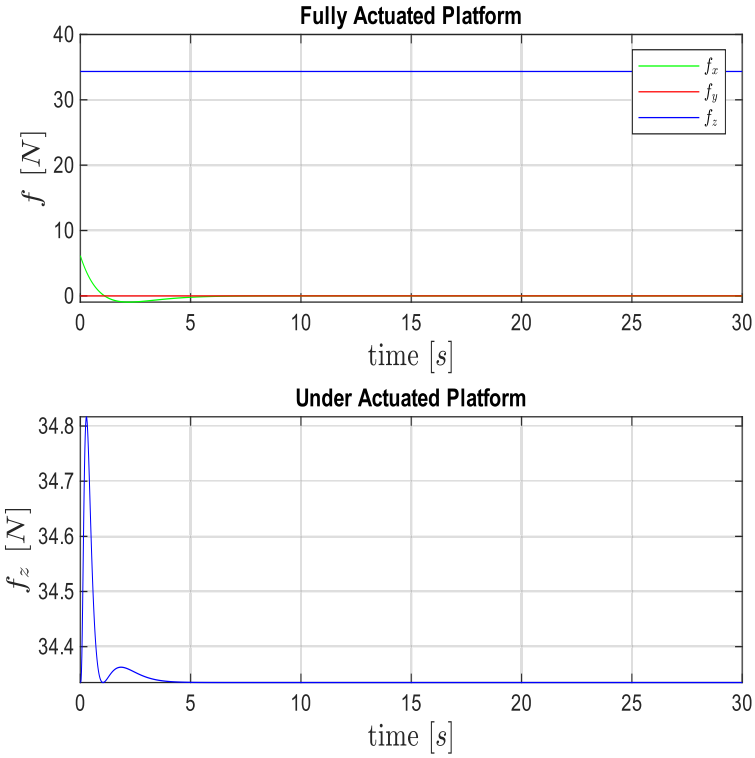

Figure 4.8: UAVs force inputs for a 2m step reference along x

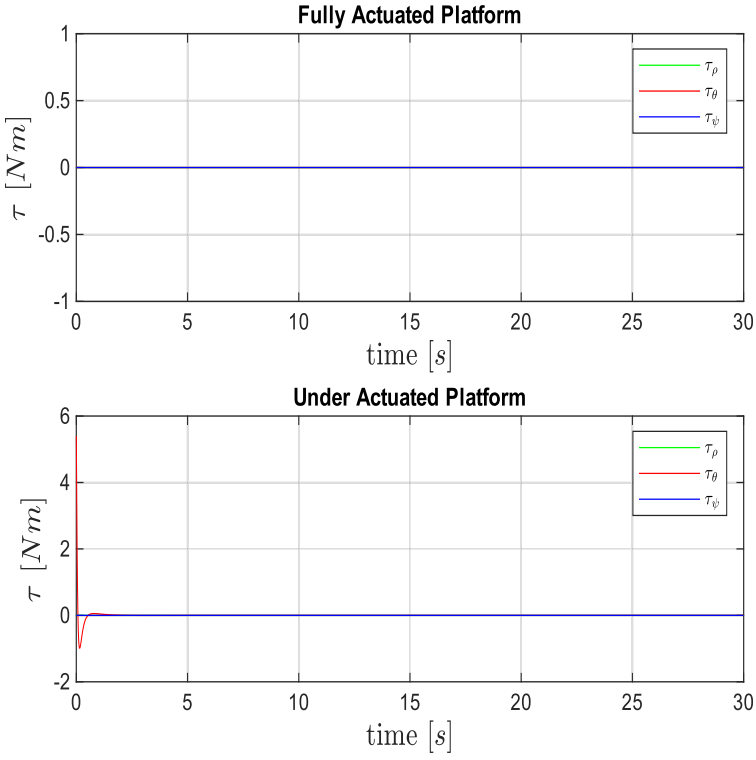

Figure 4.9: UAVs torque inputs for a 2m step reference along x

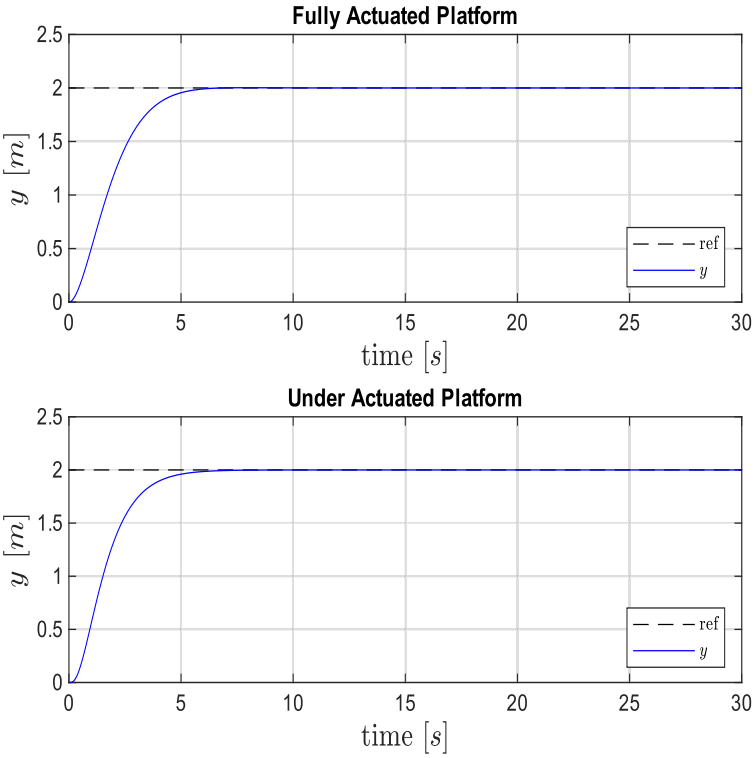

Figure 4.10: UAVs responses to a 2m step reference along y

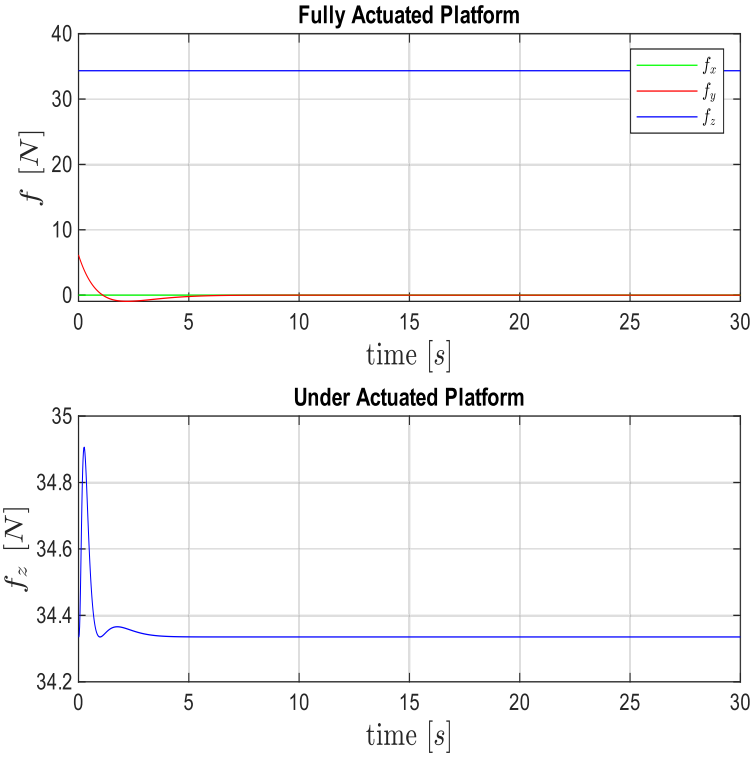

Figure 4.11: UAVs force inputs for a 2m step reference along y

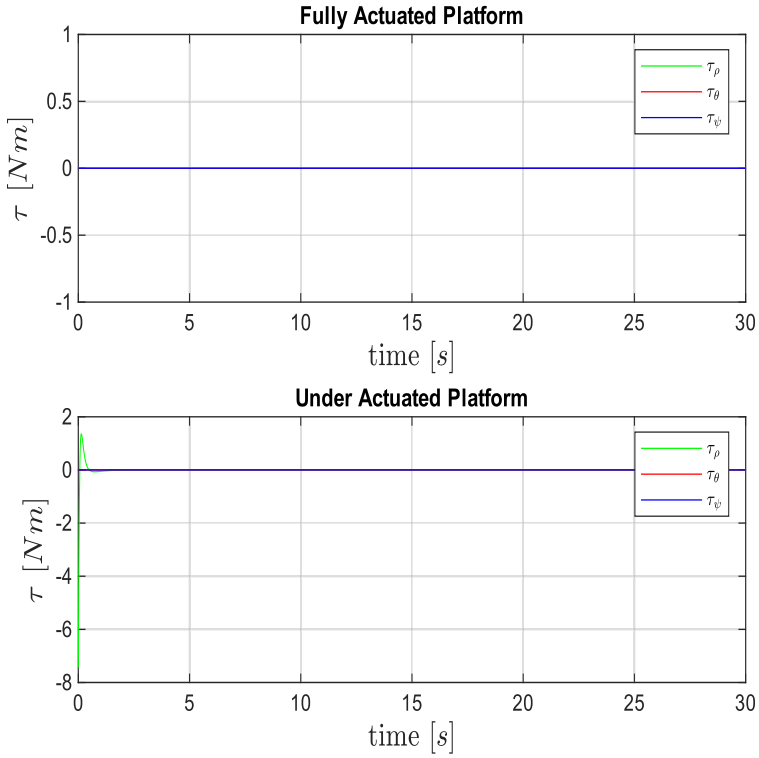

Figure 4.12: UAVs torque inputs for a 2m step reference along y

Up to now, the cartesian and ψ responses of the two mutirotors have been compared. However, the fully actuated UAV is able to change its orientation with respect to the ρ and θ angles independently from its cartesian position. Therefore, the control parameters relative to those two additional degrees of freedom need to be tuned. The goal is to obtain ρ and θ responses comparable with those achieved for the cartesian and ψ coordinates. For the ρ and θ dynamics of the tilted multirotor, step references of 45° are considered. The responses can be observed in figures 4.13 and 4.15, respectively. The ρ response is charachterised by a rise time $t_{r,1\%} = 5.54s$ and an overshoot of 0.051%, while the θ response shows a rise time $t_{r,1\%} = 5.32s$ and overshoot 0.042%. As previously established, those rotations cause motion along the y and x direction, respectively, if the correponding force component is not compensated. The control inputs needed to perform a ρ rotation on the spot are showed in figures 4.14, while the inputs required for a θ rotation maintaining constant position are showed in figure 4.16. Given that the cartesian setpoint has been set to $p_{B'} = 0^T$, the UAV nedds to apply nonzero forces to maintain the desired position.

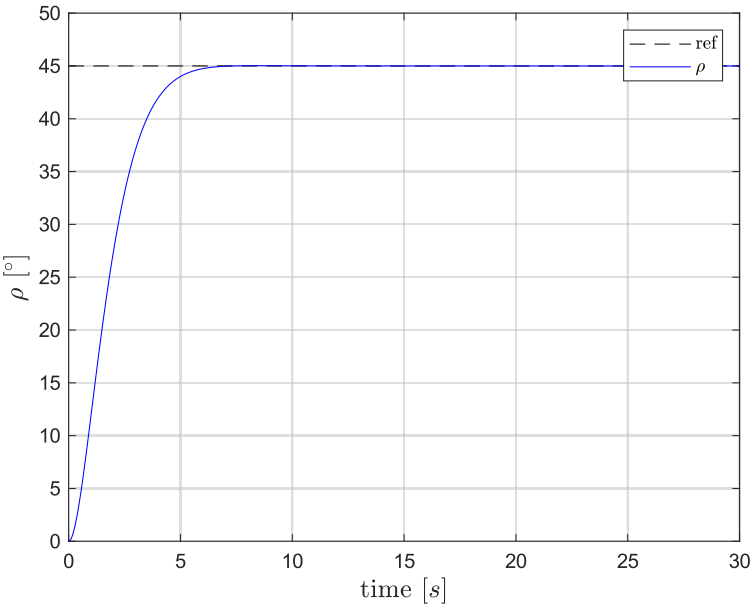

Figure 4.13: UAVs responses to a 45° ρ step reference

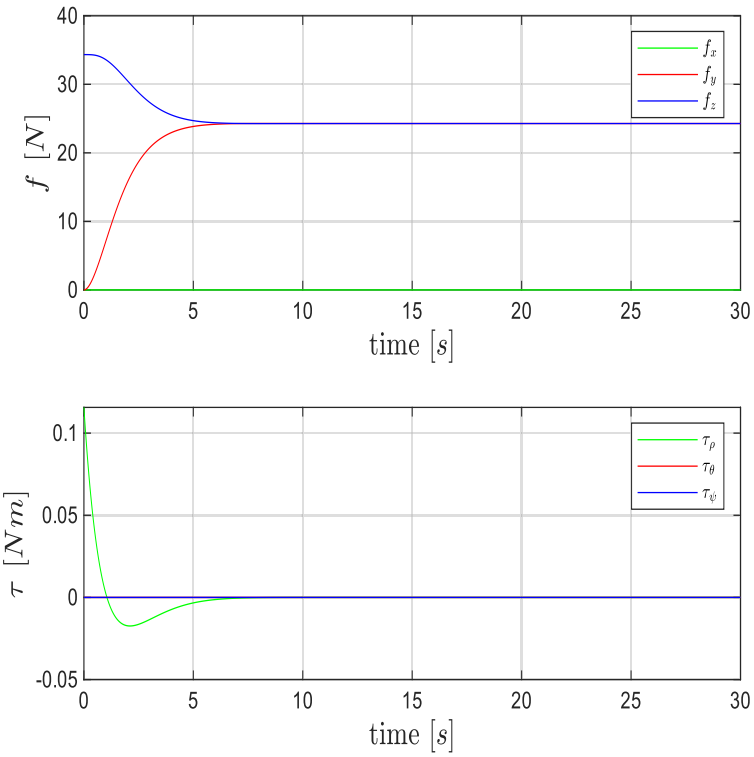

Figure 4.14: UAV force and torque inputs for a 180° ρ step reference

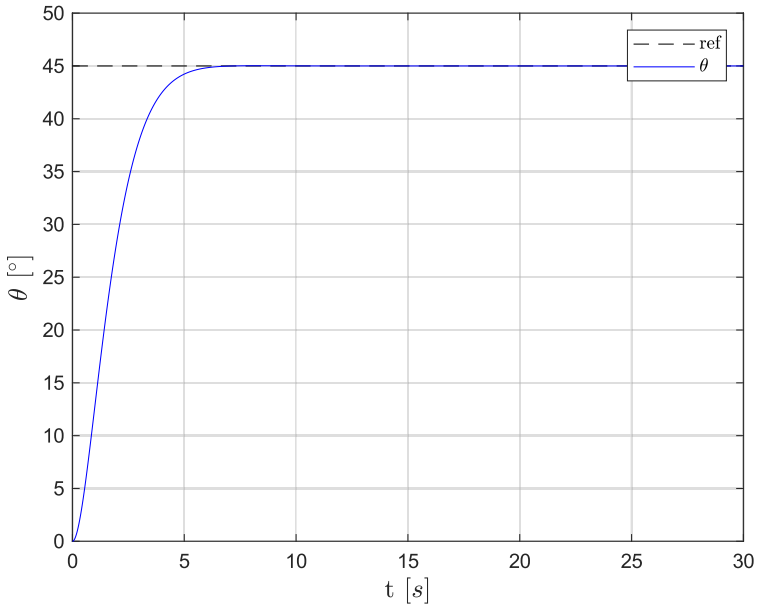

Figure 4.15: UAVs responses to a 45° θ step reference

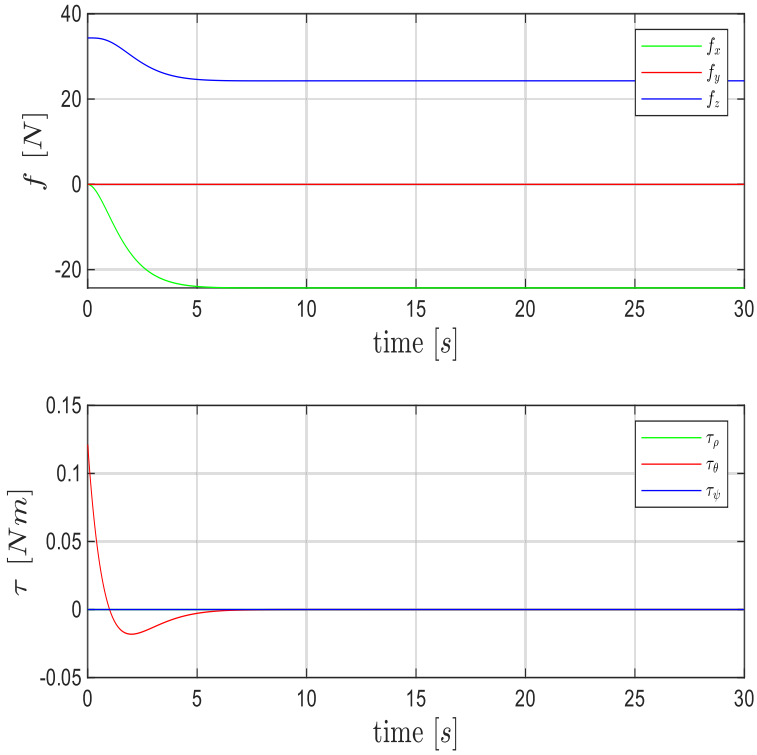

Figure 4.16: UAV force and torque inputs for a 45° θ step reference

Fully Actuated Platform		Under Actuated Platform	
$K_{P,\alpha\beta}$	$\begin{pmatrix} 0.88 & 0 & 0 \\ 0 & 0.88 & 0 \\ 0 & 0 & 0.85 \end{pmatrix}$	K_P	$\begin{pmatrix} 12 & 0 & 0 \\ 0 & 14.8 & 0 \\ 0 & 0 & 0.85 \end{pmatrix}$
$K_{D,\alpha\beta}$	$\begin{pmatrix} 1.7 & 0 & 0 \\ 0 & 1.7 & 0 \\ 0 & 0 & 1.6 \end{pmatrix}$	K_D	$\begin{pmatrix} 10 & 0 & 0 \\ 0 & 10.9 & 0 \\ 0 & 0 & 1.6 \end{pmatrix}$
$K_{P,\Phi\alpha\beta}$	$\begin{pmatrix} 0.95 & 0 & 0 \\ 0 & 1.05 & 0 \\ 0 & 0 & 0.8 \end{pmatrix}$	$K_{P,\Phi}$	$\begin{pmatrix} 15.9 & 0 & 0 \\ 0 & 15 & 0 \\ 0 & 0 & 0.8 \end{pmatrix}$
$K_{D,\Phi\alpha\beta}$	$\begin{pmatrix} 1.8 & 0 & 0 \\ 0 & 1.9 & 0 \\ 0 & 0 & 1.5 \end{pmatrix}$	$K_{D,\Phi}$	$\begin{pmatrix} 16.7 & 0 & 0 \\ 0 & 15 & 0 \\ 0 & 0 & 1.5 \end{pmatrix}$

Table 4.1: UAVs Control Parameters

4.2 FH SIMULATIONS

After the single UAV platforms are tuned, the tracking controllers are implemented and tested on the flying hand and the UAM. For the flying hand, the drone's mass and inertia are substituted with the full robot mass m_{tot} and the total inertia tensor $\mathbf{I}_{tot}$, according to the control laws described in 3.16 and 3.18. Since the actuation properties of this robot are entirely provided by the UAV, the control problem for the system reduces to a control problem for the drone. Given the full actuation of the system, a feedback linearization with PD action is considered.
The flying hand cartesian trajectory tracking response can be seen from figure 4.17, while the attitude dynamics can be observed in figure 4.18. The cartesian response is characterized by negligible tracking error, while in the attitude response only the θ angle exhibits a transient error. It can be noticed that the robot achieves asymptotic tracking of the desired trajectory. The input forces and torques generated to follow the desired trajectory are showed in figures 4.19 and 4.20, respectively. The only nonzero torque component of the input vector τ_c is τ_θ. This is to be expected, since the only rotation required by the trajectory is a θ rotations in order to properly position the end effector. On the other hand, it can be noticed that the force input vector has a nonzero x component even when translation in the x direction are not performed. This force is needed to maintain a constant baricenter's x coordinate when the body frame rotates.

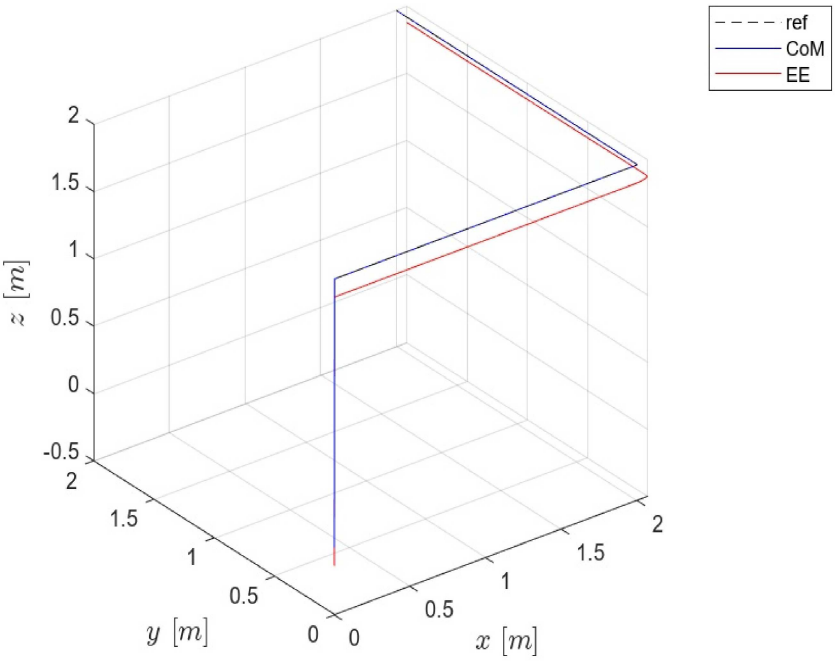

Figure 4.17: FH cartesian trajectory tracking

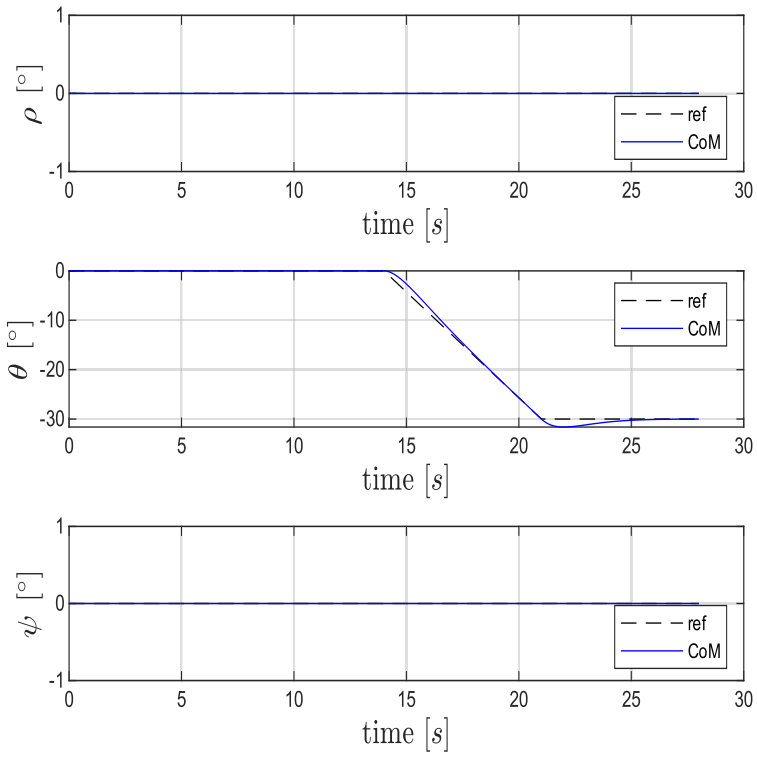

Figure 4.18: FH Euler angles tracking

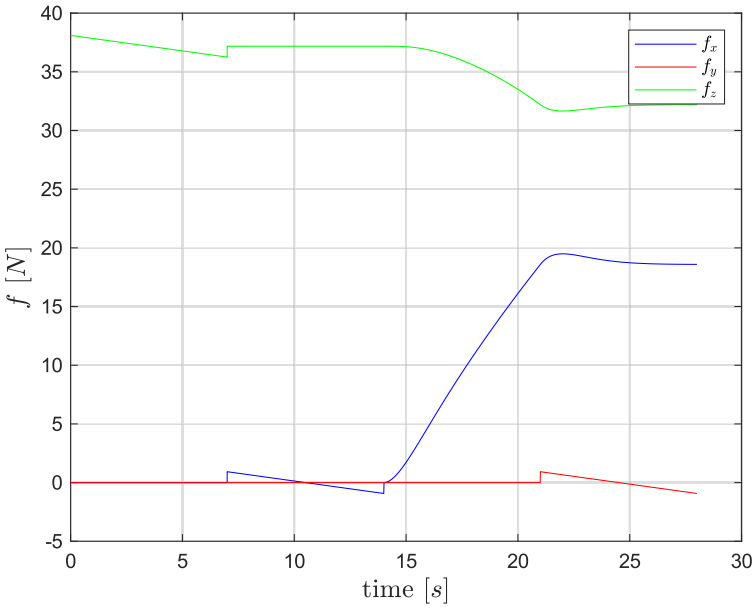

Figure 4.19: FH control forces

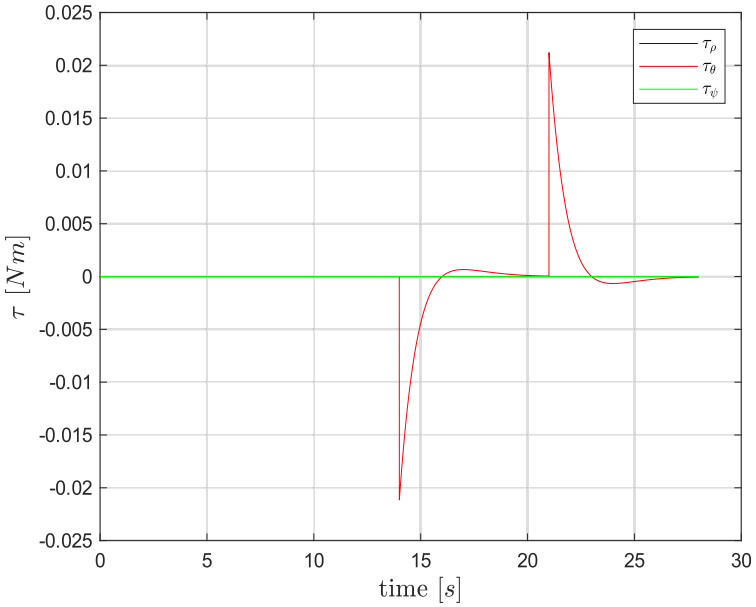

Figure 4.20: FH control torques

4.3 UAM SIMULATIONS

The UAM response to the decentralized controller is showed in figures 4.21 and 4.22, respectively. From 4.21, a constant offset in the last segment of the UAV trajectory with respect to the desired one can be observed. Moreover, the ψ angle exhibits a divergent behaviour. Finally, it can be noticed that the link trajectory presents large errors when the platform is moving along the x axis, only being able to achieve tracking in the last two phases of the trajectory.

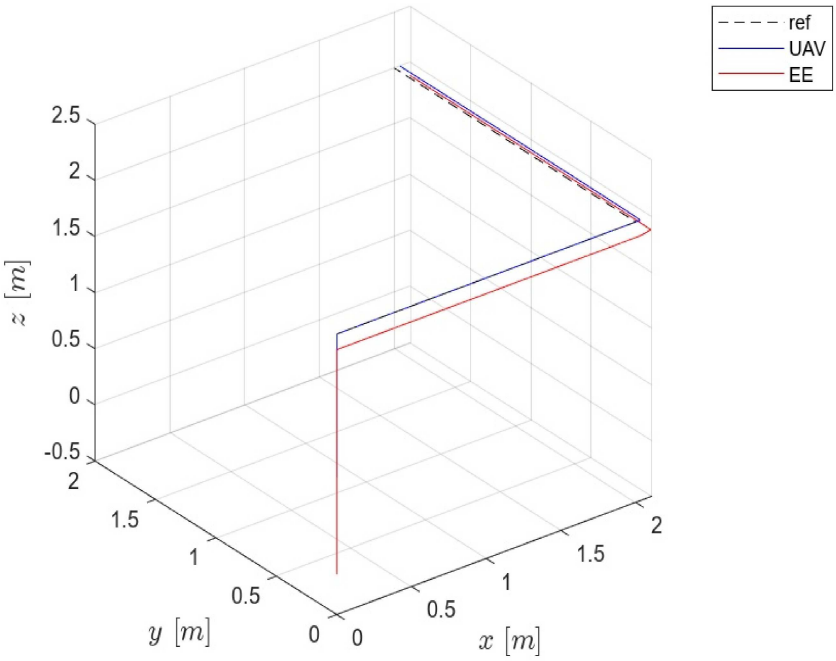

Figure 4.21: UAM cartesian trajectory tracking

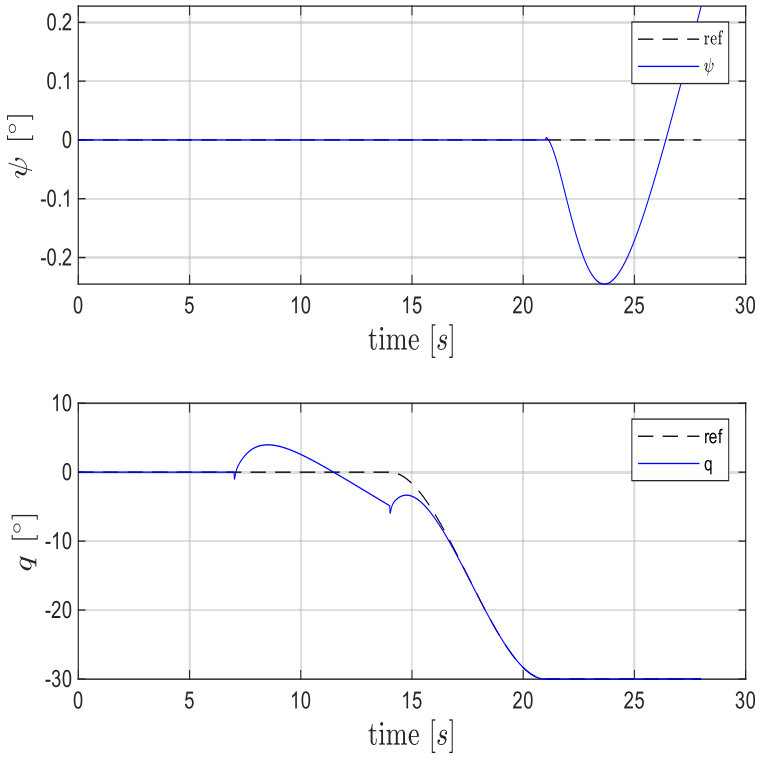

Figure 4.22: UAM ψ and q coordinates

The tracking errors are due to several causes. First, the link closed loop dynamics might be too slow to properly compensate the reaction torques induced by the UAV motion, causing a large tracking error e_q when the robot moves in the x direction. Moreover, the link motion generates reaction torques on the multirotor, which are responsible for the constant offset observed in the cartesian trajectory. Finally, when the system slides along the y direction with the link at a fixed angle, the center of gravity of the full system no longer lies on axis $z_{\mathcal{B}}$. Therefore, the forces applied by the drone propellers generate additional force torques, which cause the ψ angle to diverge from its desired value. The input UAV forces and torques, and the input link torque can be observed in figures 4.23, 4.24 and 4.25, respectively. It is immediate to notice that the multirotor torques are close to an impulsive signal of extremely large height. This is due to the trajectory having tight time requirements, and resulting in expensive control signals. Those requirements also affect the manipulator behaviour. In fact, by observing the control signal τ_l in figure 4.25, two peaks can be noticed. The time instants at which those peaks appear coincide with the time instants at which the τ_θ impulses occur, and they correspond to the time instant at which the robot starts to move along the x axis, and the time instant at which the robot starts to position its end effector.

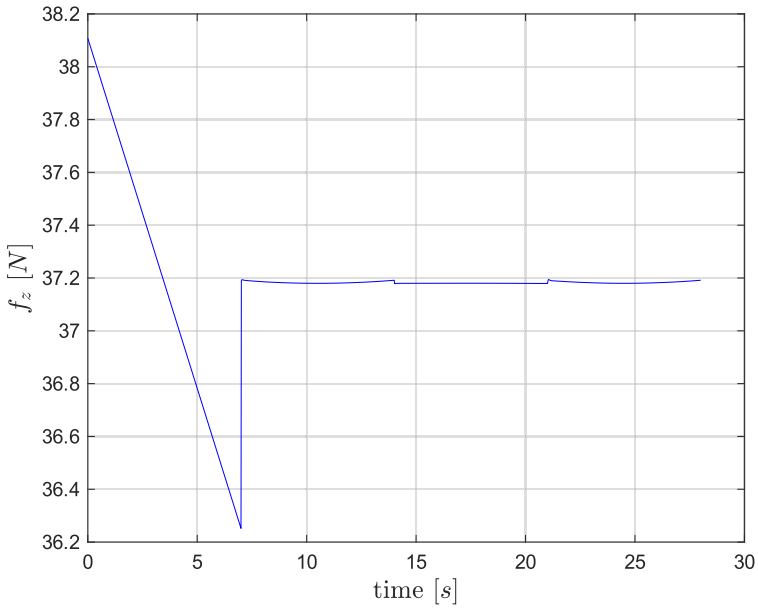

Figure 4.23: UAM thrust force

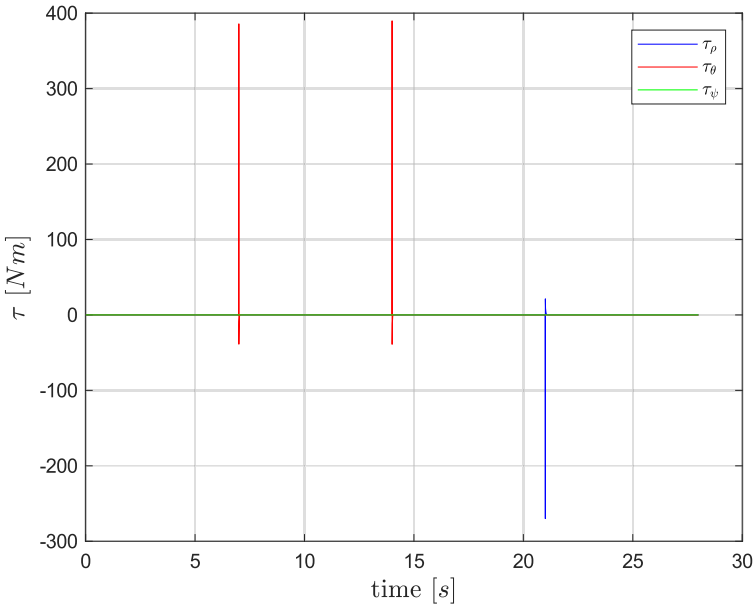

Figure 4.24: UAM input torques

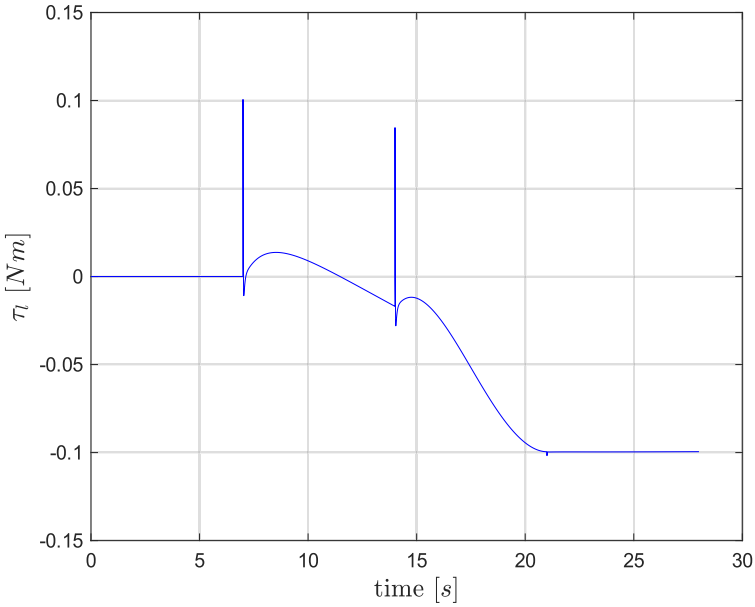

Figure 4.25: UAM link input torque

4.3.1 *UAM Tracking with Slower Trajectory*

The main reason for the poor performances obtained by the UAM are the tight time requirements imposed by the desired trajectory, a first improvement consists of relaxing those constraints. In particular, the two critical phases of the trajectory, i.e. the x and y translation phases, now require 20s to complete. The cartesian and ψ responses to the new, slower desired trajectory are reported in figures 4.26 and 4.27, respectively.

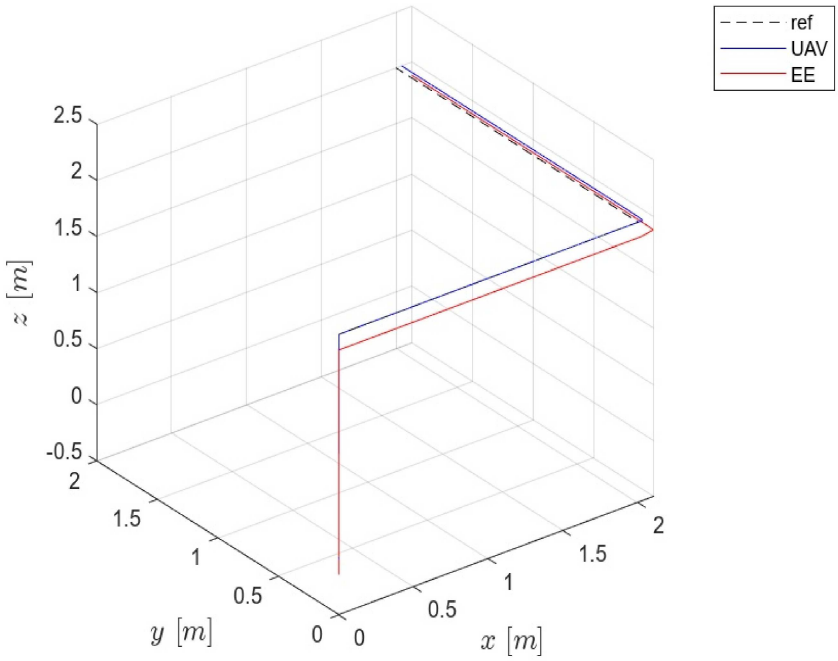

Figure 4.26: UAM cartesian trajectory tracking considering slower trajectory

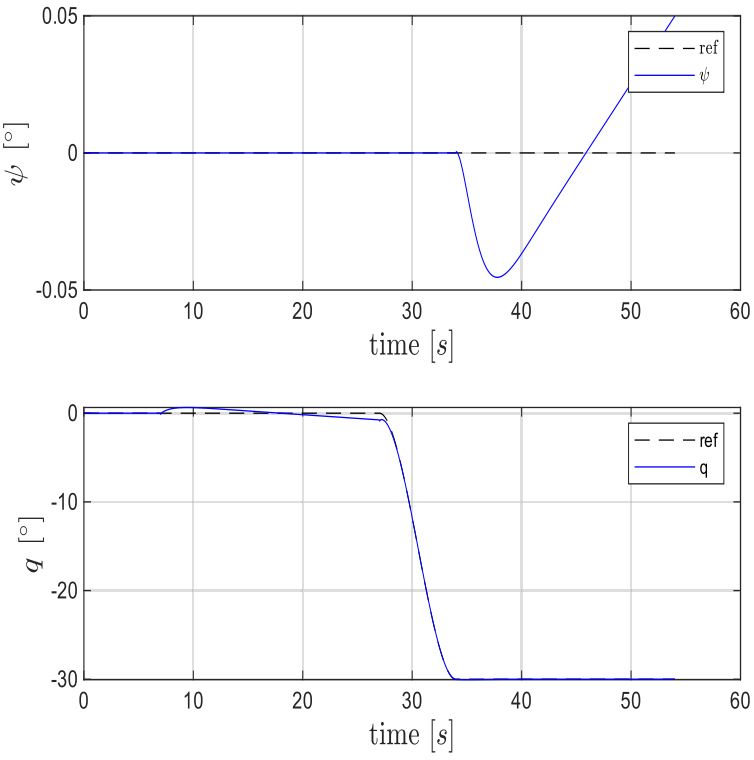

Figure 4.27: UAM ψ and q coordinates considering slower trajectory

The constant cartesian offset is not reduced, and the ψ angle response still exhibits a divergent behaviour. The only improvement provided by the relaxation of the time constraint is a reduction in the tracking error e_q when the UAV moves in the x direction. However, by looking at the new control signals in figures 4.28, 4.29 and 4.30, it can be noticed that the UAV input torques are reduced by an order of magnitude, thus resulting in a much lower control effort for the robot to complete the trajectory.

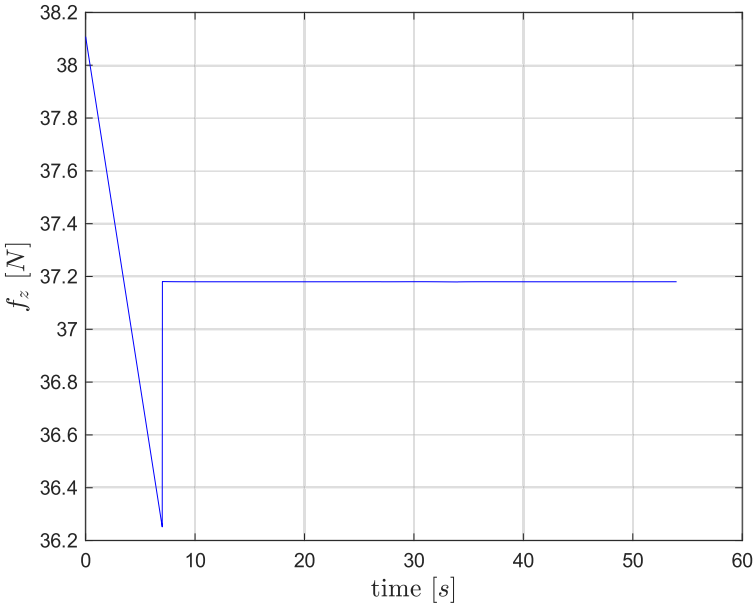

Figure 4.28: UAM thrust force considering slower trajectory

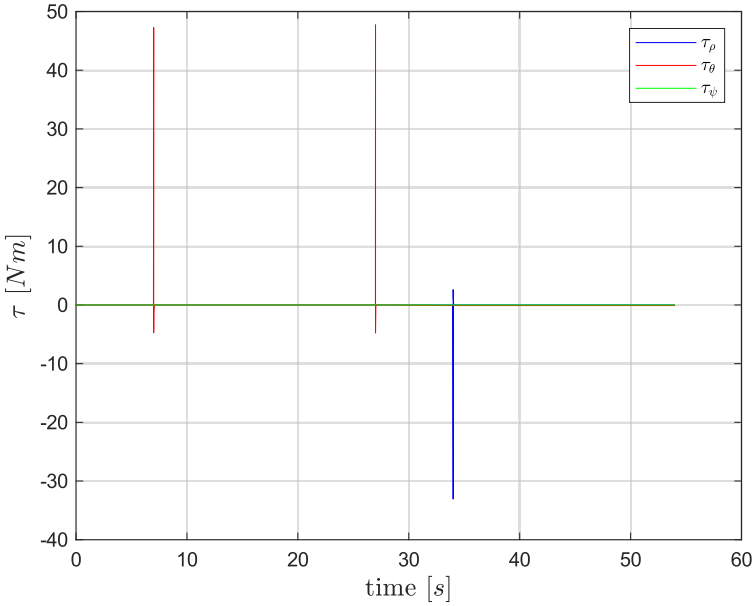

Figure 4.29: UAM input torques considering slower trajectory

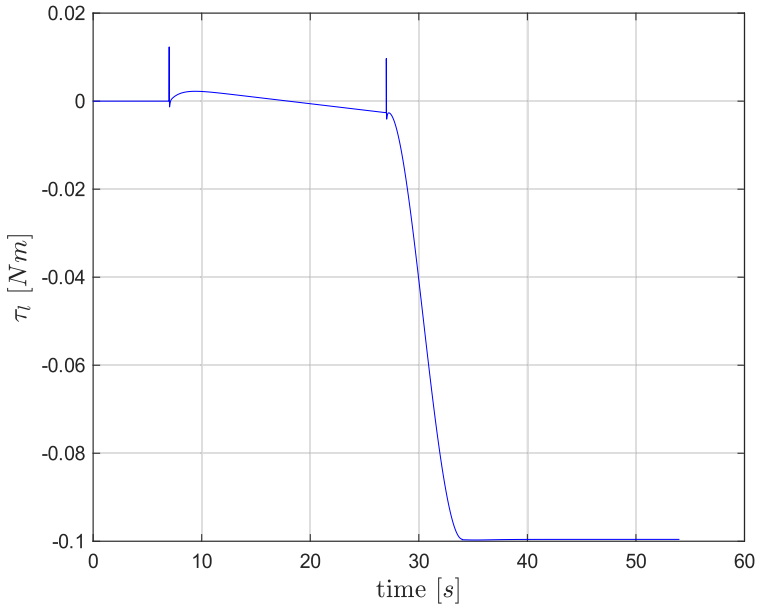

Figure 4.30: UAM link input torque considering slower trajectory

4.3.2 *UAM Tracking with Integral Action*

It is evident that slowing the critical phases of the desired trajectory results in less expensive control actions, however it does not solve the convergence issues previously discussed. A possible solution consists of implementing the integral action in the controller. In particular, the integral contribution has been added to the cartesian error dynamics, to the attitude ψ controller and and in the q generalized coordinate control law. The integral gains have been chosen as $K_I = diag(8, 8, 0)$, $K_{I,\psi} = 3$ and $k_{I,q} = 15$. The UAM response to the controllers implementing the integral action is showed in figures 4.31 and 4.32. For this simulation, the slower trajectory has been considered as reference for the system. It can be noticed that, in the last phase of the trajectory, after a small transient the robot's cartesian trajectory converges to the desired one, thanks to the integral action in the cartesian error dynamics which rejects constant disturbances. Moreover, the ψ tracking error presents a bounded and negligible constant steady state error instead of the divergent behaviour. The generalized q coordinate tracking error is also bounded when the UAV platform moves in the x direction, with $e_q = 0.11°$. However, since in this stage of the trajectory the goal is to correctly position the UAV, it is reasonable to tolerate larger end effector errors.
The control inputs produced by the controllers exploiting the integral contribution can be observed in figures 4.33, 4.34 and 4.35. No significant difference can be observed in the UAV inputs with respect to those in figures 4.28 and 4.29, relative to the response to the slower trajectory.

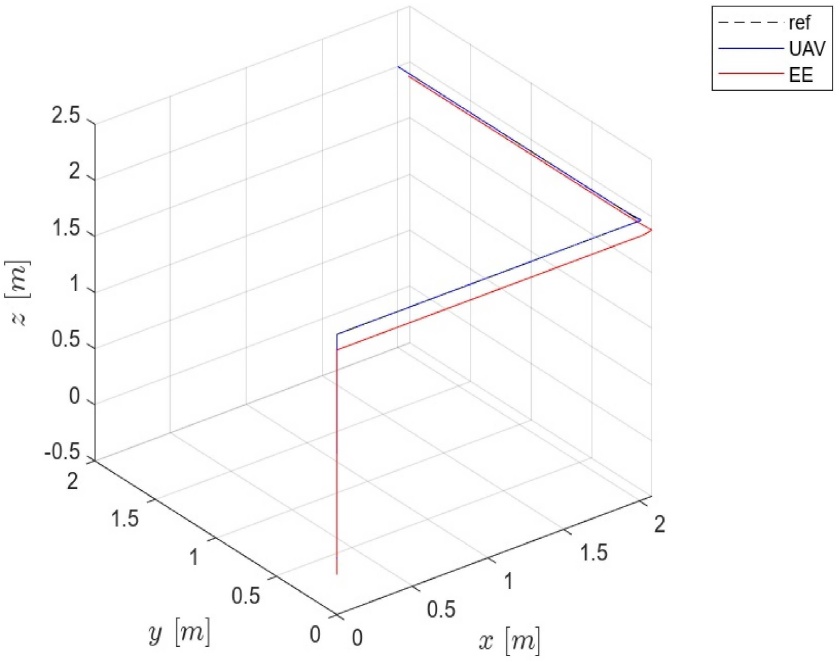

Figure 4.31: UAM cartesian trajectory tracking with integral action

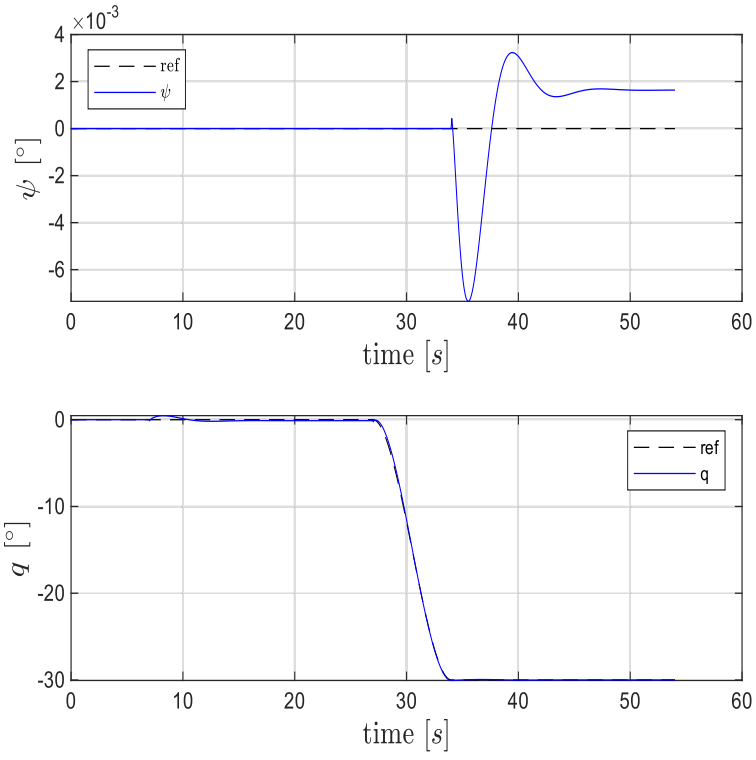

Figure 4.32: UAM ψ and q coordinates with integral action

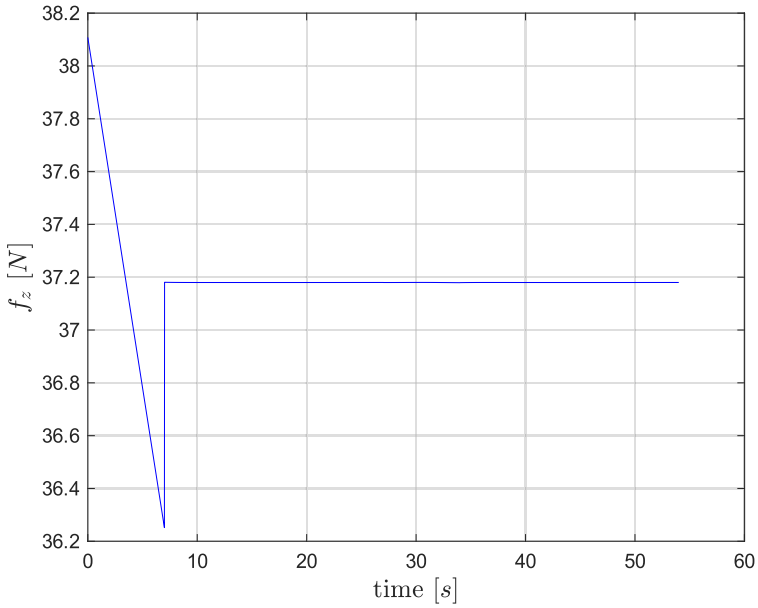

Figure 4.33: UAM thrust force with integral action

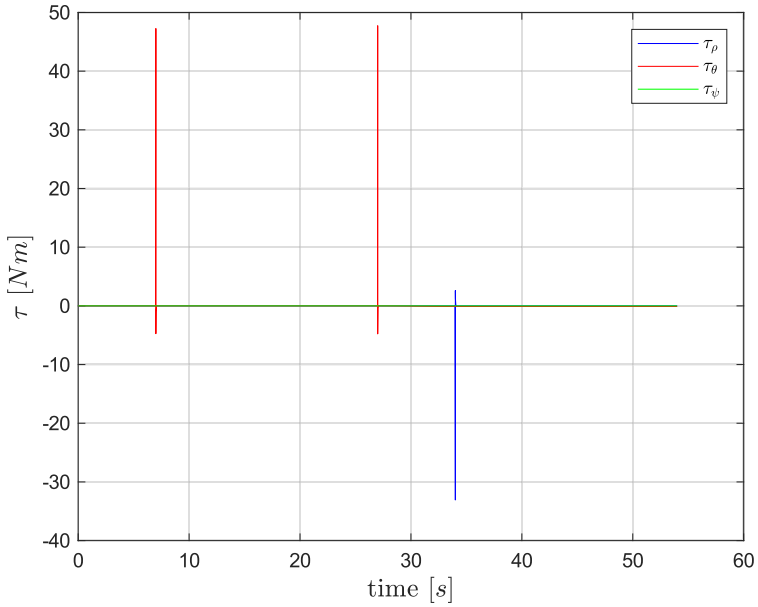

Figure 4.34: UAM input torques with integral action

Considering the results just presented, it is possible to claim that adding the integral contribution in the control laws makes the system more robust, and perfect cartesian asymptotic tracking can be achieved, while the ψ angle presents a negligible steady state error. Convergence for the q generalized coordinate is also obtained. To understand these phenomena, it is important to notice that the controllers for the UAV and the link manipulator are developed separately, in fact this is an instance of decentralized control, where the link motion is treated as a disturbance to be rejected by the multirotor and vice versa.

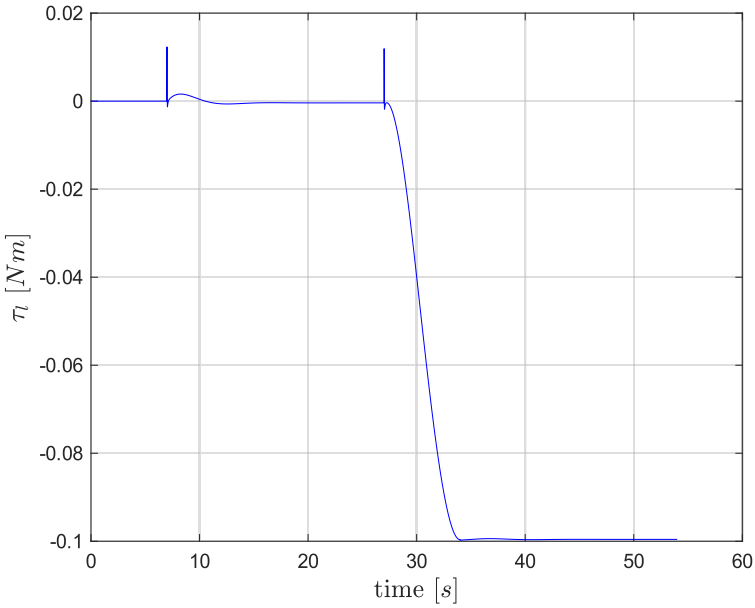

Figure 4.35: UAM link input torque with integral action

4.3.3 *UAM Tracking with Model-Based Compensation*

The addition of the integral contribution in the controllers improves both the convergence properties of the system and the required control effort. However, this strategy does not exploit full knowledge of the dynamical model 2.26. The dynamic coupling between the two bodies are considered to adjust the attitude control law, as in 3.14. The integral contribution is therefore removed from the error dynamics equation. The UAM response to the model-based controller to the slow trajectory are reported in figures 4.36 and 4.37. It is immediate to notice that the model-based control law yields perfect cartesian tracking of the desired trajectory. This is achieved thanks to the exact compensation of the disturbances, up to the first order, on the UAV generated by the link motion. This can be accomplished by adding to the UAV controller feedback on the link angular position and velocity. The ψ response presents a behaviour similar to that of the controller implementing the integral action, where the steady state error is bounded and negligible. The generalized q coordinate also presents closed loop dynamics similar to that of the previous simulation, with a bounded error e_q in the x translation phase of the trajectory, and convergence to the desired value afterwards.
The control inputs generated by the model-based controller are showed in figures 4.38, 4.39 and 4.40. Those are comparable with the signals generated considering the controller with integral action of the previous section, reported in figures 4.33, 4.34 and 4.35.

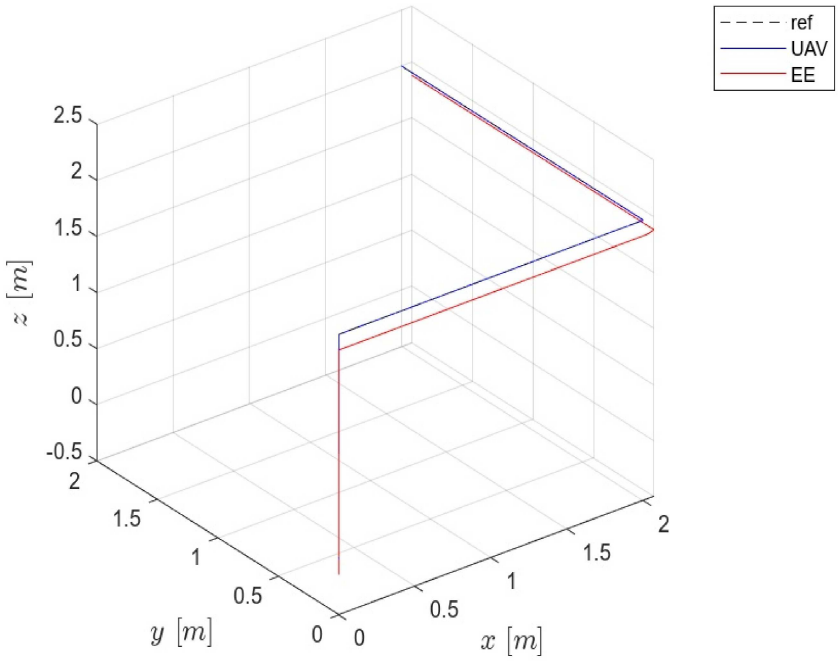

Figure 4.36: UAM cartesian trajectory tracking with model-based compensation

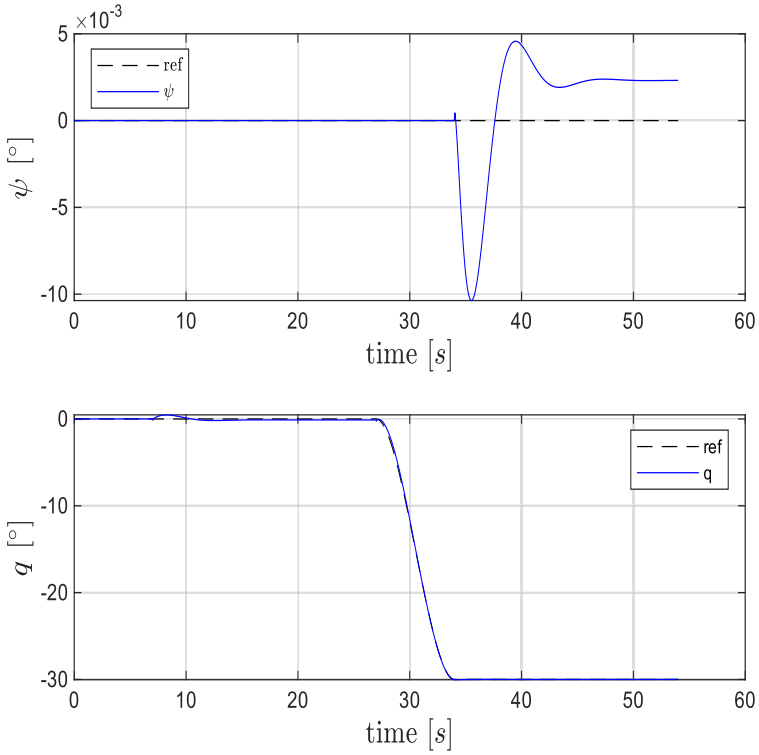

Figure 4.37: UAM ψ and q coordinates with model-based compensation

The implementation of the model-based attitude compensation in the UAV controller solves the cartesian convergence problem and yields perfect tracking of the desired trajectory. The ψ tracking error is bounded and negligible, and the link generalized coordinate error e_q is bounded in the x translation phase and convergent to zero for the rest of the trajectory. Since the multirotor uses feedback information on the link's current angular position and velocity, and the manipulator has access to the drone's current pose and velocity, this can no longer be considered as a decentralized control scheme.

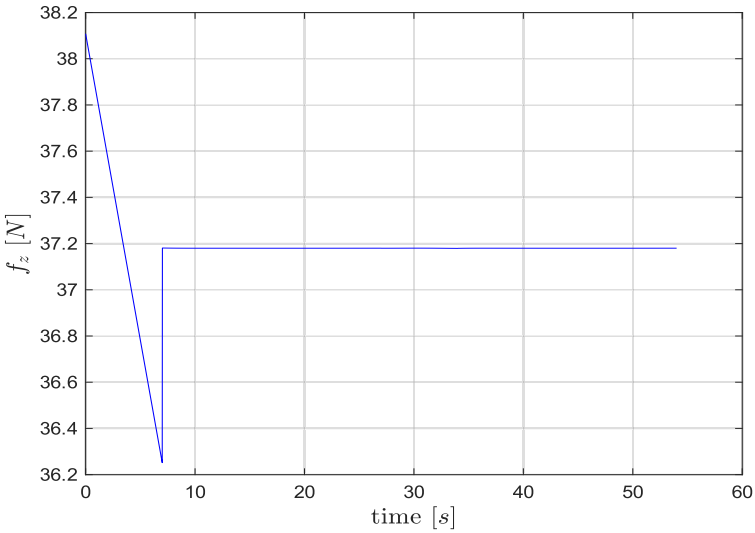

Figure 4.38: UAM thrust force with model-based compensation

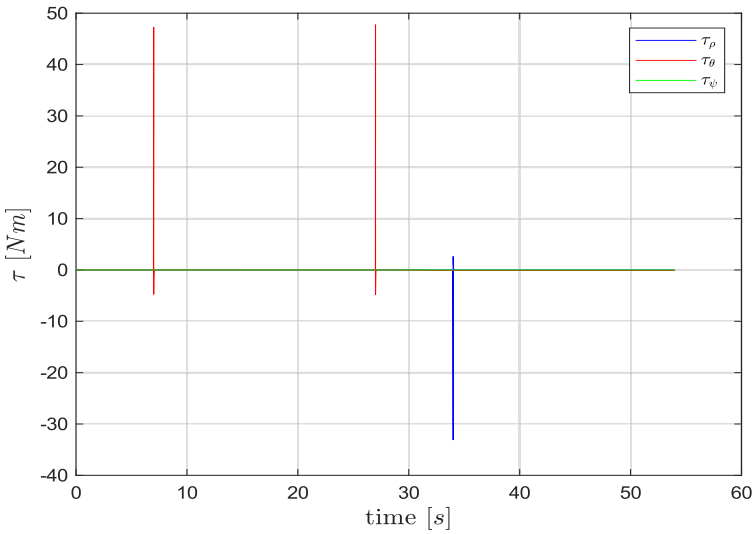

Figure 4.39: UAM input torques with model-based compensation

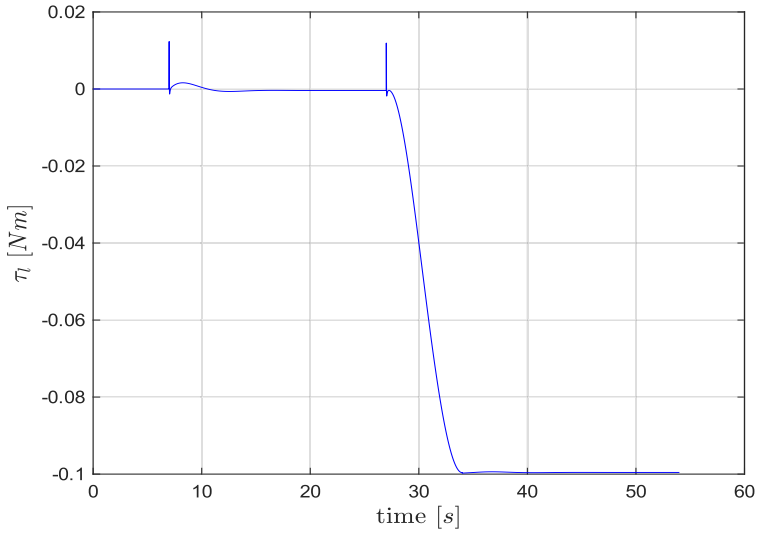

Figure 4.40: UAM link input torque with model-based compensation

4.4 QUANTITATIVE ANALYSIS

So far, only a qualitative analysis of the controllers' performances has been discussed. In this section, a quantitative analysis of the robots responses is carried out. Concerning the UAM robot, only the controller implementing the integral action anf the model-based attitude dynamics compensation are discussed, since those are the two approaches for which stability has been achieved.

4.4.1 *Performance Indexes*

The flying hand robot is fully actuated, in fact the control problem for this system has considered the full robot pose defined in $\mathbb{R}^6$. On the other hand, the UAM is underactuated, and its five controlled variables are the cartesian UAV position, the ψ angle and the link generalized coordinate q. Since the error and input vectors are defined over vector spaces of different dimensions, a direct comparison of the two is impossible. Instead, the magnitude of the error and input vectors are compared. As performance indexes, the maximum, minimum and mean value of those vectors are selected. However, those puntcual indexes do not account for the fact that the UAM is given a slower trajectory, and therefore takes more time to complete the task. If time is a critical factor to determine the robot performance, different indexes need to be defined. The choice adopted in this work consists of computing the total control effort for the trajectory. By definint as t_{FH} the total duration of the task for the flying hand, and defining t_{UAM} the total duration of the task for the UAM, the total control effort can be computed as

$$u_{tot} = \int_0^{t_k} |u_k| dt \tag{4.1}$$

where $k = \{FH, UAM\}$. This performance index penalizes longer execution times, since the input vector norm is a positive quantity integrated over a larger time interval.

4.4.2 *Performance Comparison*

The error and input vector norm for the flying hand with respect to time are showed in figure 4.41. Comparing the error norm over time with the cartesian and Euler angles responses in figures 4.17 and 4.18, it is immediate to understand that the only contribution to the error is due to the transient response to the desired θ rotation. On the other hand, the shape of the input norm signal can be approximated by the sum of the inputs f_z and f_x in figure 4.19. In fact, the most difficult action for the flying hand consist of maintaining constant height and a constant position when the body frame is rotated.

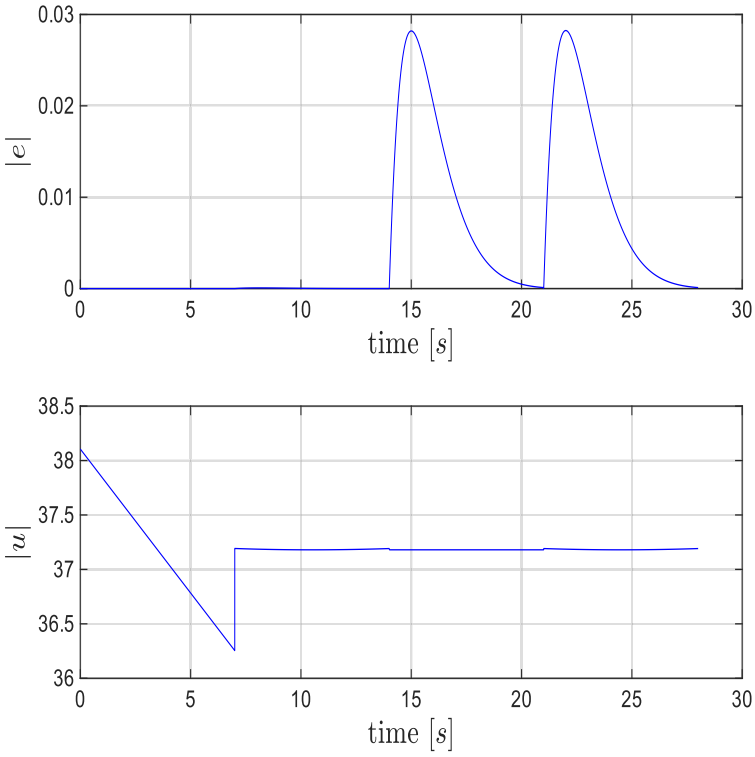

Figure 4.41: Flying Hand error and input vector norm

The error and input norm for the UAM implementing the integral action are showed in figure 4.42. Considering the error norm, it can be noticed that the largest errors correspond to the end effector positioning phase and the translation along the y axis. By looking at the input vector norm, it is immediate to notice that the most expensive actions

correspond to the body frame rotations of the UAV.
In figure 4.43, the error and input vector norm for the UAM implementing the model-based attitude dynamics compensation are reported. Considering the error norm, it can be noticed that the first portion of the signal (up to the robot's motion along x) is identical to the error norom for the system implementing the integral action. The model-based dynamic compensation is in fact relevant when the link dynamics influences the multirotor behaviour, that is, in the last two phases of the trajectory, for which the error norm is smaller. Considering instead the input vector norm, no significant difference can be noticed with respect to that reported in figure 4.42. The local maxima of the input norm correspond with the UAV command torques for the θ and ρ rotations.
In table 4.2, the performance indexes for the flying and the UAM considering the integral action and the model-based attitude compensation are reported. It can be noticed that, in terms of tracking performances, the UAM has lower error norm compared with the flying hand. The model-based compensation is the best approach according to those performance indexes. Considering the input vector norm, the two considered UAM control strategies yield identical mean, maximum and minimum value. The minimum and mean input norm are comparable with those achieved by the flying hand. On the other hand, the flying hand presents a smaller maximum, and it is therefore the better choice considering the required control effort. As previously stated, those performance indexes to not take into account the time needed by each robot to perform the desired task. To penalize the slower system, the integral of the input vector norm over the time interval needed to complete the task is considered. The flying hand presents a total required control effort which is almost halved with respect to that obtained by the UAM. The higher value associated with both UAM control strategies is due to the longer time needed to track the trajectory, and due to the integration of the impulsive components of the torque input vector.

	FH	**Integral UAM**	**Model-Based UAM**
$max(\|e\|)$	0.0282	0.0114	0.0082
$mean(\|e\|)$	0.0051	0.0024	0.0014
$max(\|u\|)$	38.1056	60.5121	60.5121
$min(\|u\|)$	36.2542	36.2517	36.2517
$mean(\|u\|)$	37.1818	37.1812	37.1812
$\int_0^t \|u\|dt$	1.04×10^3	2.01×10^3	2.01×10^3

Table 4.2: FH and UAM performance indexes

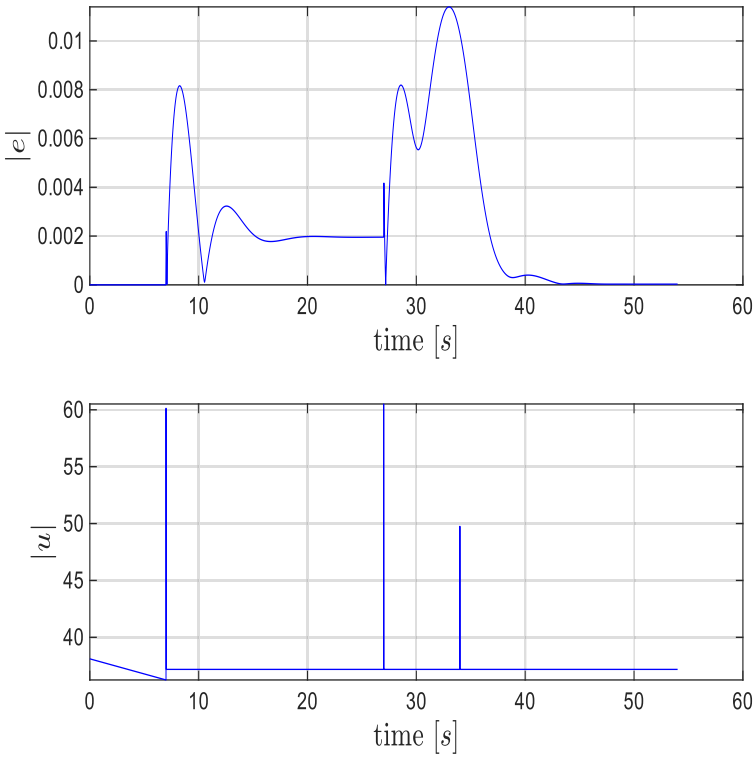

Figure 4.42: UAM error and input vector norm with integral action

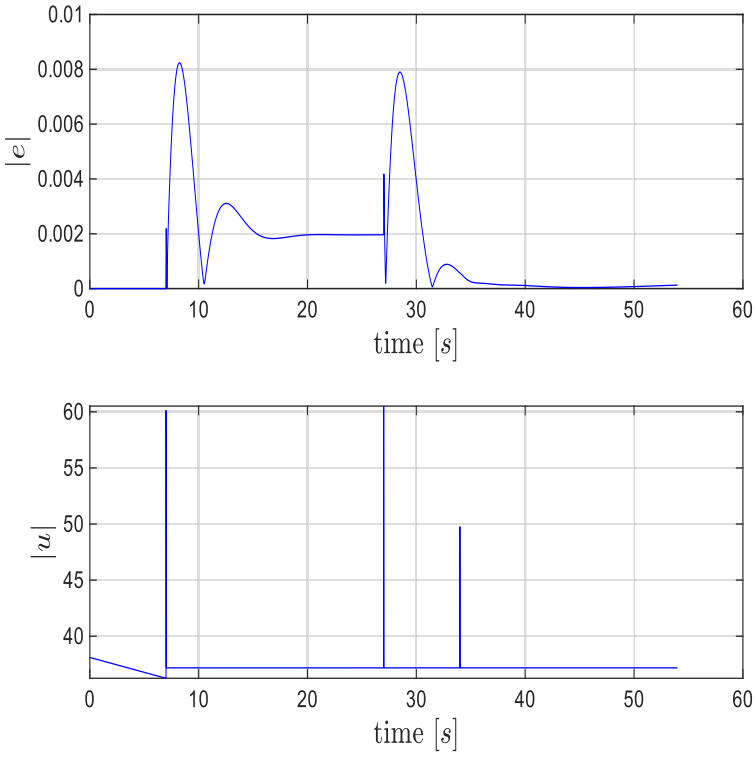

Figure 4.43: UAM error and input vector norm with model-based compensation

In conclusion, from the results obtained it can be noticed that the UAM is able to track the desired trajectory with lower errors with respect to the FH, due to the additional DoF provided by the actuated manipulator which allows higher precision. However, the UAM also requires higher control effort and takes more time to track the desired trajectory.

5

CONCLUSIONS

The scope of this thesis was to analyze two different aerial manipulator robots, with different actuation properties, and to design different control strategies to explore wether a particular configuration presents some advantages in terms of tracking performances and required control effort. To overcome the disturbances due to the manipulator motion, compensation algorithms have also been developed. The control approaches presented in this work are tested in a simulation environment exploiting MATLAB and Simulink. The main key points and final results of the thesis are summarized and discussed in the following, and some ideas on possible related future works are expressed.

5.1 FINAL REMARKS

Aerial manipulation has been a topic of great interest among academic institutions and industrial companies. Indeed, equipping a multirotor with a manipulator exploits the drone's agility and maneuverability and extends the range of possible applications, making it able to perform manipulation tasks in flight. However, the control problem for those kind of systems presents several issues. In particular, the single UAV model is highly nonlinear, and it is usually linearized or simplified by neglecting some components. Furthermore, the presence of a manipulator attached to the drone generates coupling effects in the dynamics of the system. In this work, two aerial manipulator with different actuation capabilities, namely a flying hand and an unmanned aerial manipulator, have been studied. Considering the flying hand, the actuation properties are defined entirely by the multirotor base, which consists of an $\alpha\beta$ tilted hexacopter. The tilt angles for the propellers are chosen in order to guarantee full actuation for the drone. Consequently, the flying hand is also fully actuated. A feedback linearization scheme has been developed to design a tracking controller for this robot. On the other hand, the unmanned aerial manipulator is an underactuated system, despite the robotic arm providing an additional degree of freedom to the underactuated platform. A decentralized UAV control scheme combined with a feedback linearization arm controller has been developed, however it shows poor

tracking performances due to the arm dynamics influencing the UAV dynamics, and large input signals. To overcome those issues, different strategies are explored. First, a more robust strategies includes the integral action on the UAV cartesian and attitude dynamics, and in the link control law, to reject the disturbances caused by the manipulator motion. Afterwards, the fully coupled dynamical model of the multirotor plus drone is exploited to exactly compensate the arm disturbances affecting the UAV attitude dynamics. Both strategies allowthe robot to complete the desired task with lower tracking errors and control effort. In terms of tracking performances, the UAM presents lower error vector norm with respect to the flying hand. However, it is the most expensive system in terms of time and control effort.

5.2 FUTURE WORK

In this work, control startegies yielding stability for both the fully actuated flying hand and the underactuated UAM have been developed. Those approaches have been tested on a simulation environment (MATLAB and Simulink). Possible development of this work include:

- **Testing on more accurate and experimental setups**: in this thesis, each control design was tested in a simplified environment. In order to better asses the performances of the robots and to better understand the systems behaviour and responses, validation test could be performed on a more physicaly accurate simulator such as Gazebo [18]. Followin the same reasoning, the control schemes could be tested in real world experimental setups.

- **Even actuation properties**: in this work, the flying hand was a fully actuated system, while the UAM has beeen revealed to be an underactuated system. Under actuated systems are usually more difficult to control and are limited in their ability to solve a desired task, due to their dynamic constraints resulting in less degrees of freedom to exploit. Adding an independent degree of freedom to the UAM would allow a wider range of possible solutions to complete a desired task, and would allow the implementation of more efficient control strategies.

- **Implementing different control strategies**: for the flying hand, a feedback linearization scheme with PID action has been examined, while for the UAM a position and ψ controller has been developed neglecting the presence of the robotic link, and later adjusted to improve the robot's performances. However, many different control architechtures could be designed and tested to try to further improve the system efficiency.

- **Considering different manipulation task**: in this document, only the task of tracking a desired trajectory has been consid-

ered. Aerial manipulator are able to perform a wide range of tasks, such as trasporting or manipulating an object during flight. Different scenarios, where the robot interacts with the environment, could be considered to carry out a similar analysis of the tracking performances and input effort requirements of aerial manipulators similar to those examined in this work.

BIBLIOGRAPHY

[1] Geo Social-Science and Media. 2014. URL: http://www.geo-social.net/?p=829.

[2] The White House. 2016. URL: https://obamawhitehouse.archives.gov/the-press-office/2016/08/02/fact-sheet-new-commitments-accelerate-safe-integration-unmanned-aircraft.

[3] *The Partnership for Robotics in Europe. The Strategic Research Agenda.* 2013. URL: http://roboproject.h2214467.%20stratoserver.net/cms/upload/PPP/SRA2020_SPARC.pdf.

[4] Fabio Ruggiero, Vincenzo Lippiello, and Anibal Ollero. "Aerial Manipulation: A Literature Review". In: *IEEE Robotics and Automation Letters* 3.3 (2018).

[5] Ivana Palunko, Patricio Cruz, and Rafael Fierro. "Agile Load Transportation : Safe and Efficient Load Manipulation with Aerial Robots". In: *IEEE Robotics & Automation Magazine* 19 (2012), pp. 69–79. URL: https://api.semanticscholar.org/CorpusID:1387495.

[6] H. Nguyen and D. Lee. "Hybrid force/motion control and internal dynamics of quadrotors for tool operation". In: *Proc. 2013 IEEE/RSJ Int. Conf. Intell. Robots Syst., Tokyo, Japan* (2015), pp. 3458–3464.

[7] M. Bernard et al. "Autonomous transportation and deployment with aerial robots for search and rescue mission". In: *Journal of Field Robotics* 28.6 (2011), pp. 914–931.

[8] Fabio Ruggiero et al. "A multilayer control for multirotor UAVs equipped with a servo robot arm". In: *IEEE International Conference on Robotics and Automation, ICRA 2015, Seattle, WA, USA, 26-30 May, 2015*. IEEE, 2015, pp. 4014–4020. DOI: 10.1109/ICRA.2015.7139760. URL: http://dx.doi.org/10.1109/ICRA.2015.7139760.

[9] A. E. Jimenez-Cano et al. "Control of an aerial robot with multi-link arm for assembly tasks". In: *2013 IEEE International Conference on Robotics and Automation, Karlsruhe, Germany, May 6-10, 2013*. IEEE, 2013, pp. 4916–4921. ISBN: 978-1-4673-5641-1. DOI: 10.1109/ICRA.2013.6631279. URL: http://dx.doi.org/10.1109/ICRA.2013.6631279.

[10] B. Yang Y. He J. Han and G. Liu. "Rotor-flying manipulator: Modeling, analysis and control". In: *Math. Problems Eng* (2016), pp. 732–737.

[11] S. Kim, S. Choi, and H. Kim. "Aerial manipulation using a quadrotor with a two DOF robotic arm". In: *Proc. 2013 IEEE/RSJ Int. Conf. Intell. Robots Syst., Tokyo, Japan* (2016), pp. 4990–4995.

[12] M. Orsag et al. "Hybrid adaptive control for aerial manipulation". In: *J. Intell. Robot. Syst.* 73.1-4 (2014), pp. 1446–1451.

[13] Vincenzo Lippiello and Fabio Ruggiero. "Exploiting redundancy in Cartesian impedance control of UAVs equipped with a robotic arm". In: *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2012, Vilamoura, Algarve, Portugal, October 7-12, 2012*. IEEE, 2012, pp. 3768–3773. ISBN: 978-1-4673-1737-5. DOI: 10.1109/IROS.2012.6386021. URL: http://dx.doi.org/10.1109/IROS.2012.6386021.

[14] Vincenzo Lippiello and Fabio Ruggiero. "Cartesian Impedance Control of a UAV with a Robotic Arm". In: *IFAC Proceedings Volumes* 45.22 (2012). 10th IFAC Symposium on Robot Control, pp. 704–709. ISSN: 1474-6670. DOI: https://doi.org/10.3182/20120905-3-HR-2030.00158. URL: https://www.sciencedirect.com/science/article/pii/S1474667016336928.

[15] B. Siciliano et al. *Robotics: Modelling, Planning and Control*. London, UK: Springer, 2008.

[16] K. Nonami et al. *Autonomous Flying Robots. Unmanned Aerial Vehicles and Micro Aerial Vehicles*. Springer, 2010.

[17] S. Martini et al. "Correction to the Euler Lagrange Multirotor Model with Euler Angles Generalized Coordinates". In: *Journal of Intelligent and Robotic Systems* (2024).

[18] Kenta Takaya et al. "Simulation environment for mobile robots testing using ROS and Gazebo". In: *2016 20th International Conference on System Theory, Control and Computing (ICSTCC)* (2016), pp. 96–101. URL: https://api.semanticscholar.org/CorpusID:14692957.