Modélisation et contrôle des véhicules VTOL avec manipulateurs rigides

Jonatan Álvarez-Muñoz

Communauté
UNIVERSITÉ Grenoble Alpes

THÈSE

Pour obtenir le grade de

DOCTEUR DE LA COMMUNAUTÉ UNIVERSITÉ GRENOBLE ALPES

Spécialité : **Automatique - Productique**

Arrêté ministériel : 7 Août 2006

Présentée par

Jonatan Uziel ALVAREZ MUÑOZ

Thèse dirigée par **Nicolas MARCHAND,**
codirigée par **José Fermi GUERRERO CASTELLANOS**
et par **Sylvain DURAND**

préparée au sein **du GIPSA-Lab de Grenoble**
dans **l'école doctorale EEATS**

Modeling and control of VTOL vehicles with rigid manipulators

Thèse soutenue publiquement le **7 Novembre 2017**,
devant le jury composé de :

Antonio FRANCHI
Chargé de recherche, LAAS-CNRS, Toulouse, Rapporteur
Arturo ZAVALA RIO
Chargé de recherche, Instituto Potosino de Investigación Científica y Tecnológica, A.C., S.L.P, Mexique, Rapporteur
Stéphane VIOLLET
Directeur de recherche, ISM-CNRS, Marseille, Président
Edouard LAROCHE
Professeur, ICube - AVR, Université de Strasbourg, Examinateur
Ahmad HABLY
Maître de conférences, GIPSA-Lab-CNRS, Grenoble, Examinateur
Nicolas MARCHAND
Directeur de Recherche, GIPSA-Lab-CNRS, Grenoble, Directeur de thèse
José Fermi GUERRERO CASTELLANOS
Professeur, BUAP, Puebla, Mexique, Co-Directeur de thèse

A dissertation submitted for the degree of Doctor of Philosophy of the Université de Grenoble

❧

Électronique, Électrotechnique, Automatique et Traitement du Signal (EEATS)

Field : Automatique-Productique

Modeling and control of VTOL vehicles with rigid manipulators

By : **Jonatan Uziel Alvarez Muñoz**

Under the direction of Nicolas Marchand, José Fermi GUERRERO CASTELLANOS **and** Sylvain DURAND

Doctoral Committe:

Rapporteur: Antonio FRANCHI, Chargé de recherche, LAAS-CNRS, Toulouse,

Rapporteur: Arturo ZAVALA RIO, Chargé de recherche, Instituto Potosino de Investigación Científica y Tecnológica, A.C., S.L.P, Mexique,

Examinateur: Stéphane VIOLLET, Directeur de recherche, ISM-CNRS, Marseille,

Examinateur: Edouard LAROCHE, Professeur, ICube - AVR, Université de Strasbourg,

Examinateur: Ahmad HABLY, Maître de conférences, GIPSA-Lab-CNRS, Grenoble.

Date: December 4, 2017

Aknowledgments

First of all, I want to express my gratitude to my advisors, professors Nicolas MARCHAND, Fermi GUERRERO-CASTELLANOS and Sylvain DYRAND for their counseling during the development of my Ph.D studies, their kindness and friendship. Thank you for always have your doors open for questions or any other kind of support.

I want to thank the jury members, Antonio FRANCHI, Arturo ZAVALA RIO, Stéphane VIOLLET, Edouard LAROCHE and Ahmad HABLY for accepting to review this work and for their suggestions to improve the quality of this work.

My gratitude also to the engineering team of GIPSA-lab, especially to Jonathan DUMON, for his support during the experimental development of the project.

My gratitude to the National Council of Science and Technology of Mexico (CONACYT), for the granted scholarship. Without this financial support, this work would not have been possible.

I want to thank also to my colleagues and friends in GIPSA, for creating an environment of friendship and for always giving me their support whenever I was in need of it. Special thanks to my mexican friends, José Juan TELLEZ GUZMAN, Josué COLMENARES VAZQUEZ, Jorge VELAZQUEZ RODRIGUEZ and Raúl MELENDEZ, for the numerous discussions, support and camaraderie during my stay in Grenoble.

My thoughts go out to my mom and my aunt, who have always supported me. Thank you from the deep of my heart.

Then, my wife Nancy. Thank you for always being there for me, showing me your love and care.

And last, but not least, my biggest motivations, my daughters Samantha and Daniela, I dedicate this work to you.

Contents

List of Figures

Résumé

La manipulation aérienne a été un domaine de recherche actif au cours des dernières années, principalement parce que le travail actif des véhicules aériens sans pilote (UAV en anglais) augmente l'employabilité de ces véhicules pour diverses applications. Avec la manipulation aérienne, un nouveau chapitre s'ouvre dans les domaines de l'aéronautique, de l'instrumentation, de la robotique et du contrôle, car ce ne sont pas seulement des machines télécommandées, mais aussi de véritables systèmes autonomes pouvant interagir avec l'environnement. Le développement récent de la manipulation aérienne a trouvé des applications potentielles dans les domaines militaires et civils.

Les recherches sur la robotique aérienne ont principalement porté sur les hélicoptères et les architectures de décollage et atterrissage verticaux (VTOL). Le principal avantage de ces plates-formes est leur manoeuvrabilité et leur capacité à effectuer des survols, ce qui est essentiel pour les applications.

Cette thèse traite des avions VTOL, où l'hélicoptère à quatre rotors, quadcopter ou quadrotor est principalement étudié. Un tel véhicule est un système mécanique sous-actionné qui a six degrés de liberté (DoF) mais seulement quatre entrées de contrôle, à savoir le roulis, le tangage, le lacet et la poussée et qui est un système dynamiquement instable. Le véhicule a la capacité de décoller et d'atterrir en position verticale. En raison de sa symétrie, le modèle mathématique et la construction du véhicule sont assez simples.

En ce qui concerne le problème de la manipulation aérienne, la quantité d'applications est augmentée, mais en même temps la complexité de la modélisation et du contrôle d'un tel système est également plus grande. L'un des plus grands défis provient de leur charge utile limitée. Certaines approches ont tenté de résoudre le problème en utilisant des robots multiples pour transporter des charges utiles avec des préhenseurs ou avec des câbles, où leurs effecteurs et les préhenseurs doivent être eux-mêmes légers et capables de saisir des formes complexes. Un autre défi est que la dynamique du robot est significativement modifiée par l'ajout de charges utiles. Cependant, pour le transport de la charge utile, il est nécessaire que les robots puissent estimer l'inertie de la charge utile et s'y adapter pour améliorer les performances de suivi. Avec cela, le contrôle du système devient un

défi dû à la présence de non-linéarités, les forces aérodynamiques, la charge utile limitée des UAV et maintenant le moment d'inertie encore plus complexe qui change en raison du mouvement du bras manipulateur.

Selon les antécédents et les défis des véhicules VTOL transportant des charges utiles ou des manipulateurs, la contribution du présent travail est centrée sur la modélisation et la conception d'un contrôle non linéaire et une analyse de stabilité formelle pour la stabilisation asymptotique d'un véhicule VTOL portant un bras manipulateur.

Le travail est structuré comme suit. Le premier chapitre est consacré à la présentation de quelques préliminaires et concepts mathématiques, abordant les problèmes de l'attitude du corps rigide, de la représentation de la position et des manipulateurs rigides, tous ces concepts sont utilisés dans le présent travail.

Puis, puisque ce travail est étroitement lié à la manipulation, une brève revue des travaux réalisés dans le domaine des manipulateurs robotiques est présentée. Les principales configurations et leurs applications sont abordées, par conséquent, une brève recherche des différentes approches de contrôle et un modèle dynamique de robot manipulateur classique est montré. Pour ce faire, la configuration du coude avec 3 degrés de liberté et des liaisons pivots est introduite. La modélisation est réalisée avec l'approche d'Euler-Lagrange qui prend en compte l'énergie cinétique et potentielle d'un objet.

Le deuxième chapitre se déplace vers le domaine des systèmes aériens. Une présentation des différentes configurations pour les véhicules aériens est présentée, soulignant l'importance des avions VTOL pour certaines applications spécifiques. Ces véhicules (VTOL) sont rapidement revus, présentant certaines des configurations les plus courantes et leurs principales caractéristiques. Avec cette classification, nous nous concentrons sur les systèmes multi-rotors et nous effectuons une recherche concise sur les stratégies de contrôle pour la stabilisation de ce type de véhicules.

Pour finir avec l'état de l'art, une revue des différentes configurations de systèmes VTOL transportant des charges utiles ou des manipulateurs robotiques est effectuée, leurs avantages et inconvénients sont analysés en fonction des applications et de l'environnement où ils interagissent. En outre, une étude des différentes stratégies de contrôle est effectuée, analysant la stabilité, la robustesse, le suivi des performances, etc.

Le chapitre trois est axé sur la modélisation du véhicule aérien portant le bras manipulateur. Pour cela, nous présentons le modèle dynamique du véhicule aérien utilisé dans ce projet. Le comportement du quadcoptère peut être considéré comme un corps rigide flottant sur l'espace sous l'action de divers couples. Cependant, comme certaines singularités peuvent apparaître avec la représentation des angles d'Euler pour l'attitude, le modèle est conduit avec la représentation du quaternion. Ensuite, l'interaction entre les couples et les forces générées par les actionneurs est présentée.

Après cela, étant donné que le mouvement du bras manipulateur situé sous le fuselage du véhicule aérien exerce des couples inconnus provoquant des perturbations, un nouveau modèle pour le quadcoptère portant le bras manipulateur est introduit. En général, le modèle du bras manipulateur est conduit de deux façons: un modèle statique (seuls les effets de la gravité sont pris en compte), et un modèle avec dynamique partiel (une partie de la dynamique du moteur est prise en compte).

Pour le premier, la dynamique du bras manipulateur est négligée. Cependant, grâce à certaines équations proposées, il est possible d'estimer les couples statiques par la connaissance de la position angulaire des liens et de certains paramètres physiques du manipulateur, comme la masse, la longueur et le moment d'inertie de chaque liaison. Ce couple calculé est envoyé au contrôle d'attitude pour compenser les forces indésirables.

Le deuxième modèle prend en compte une partie de la dynamique du bras manipulateur. Comme le système physique utilise des servomoteurs, il est possible de les modéliser comme des systèmes de premier ordre. Avec ceci, il est possible de calculer la position et la vitesse sur chaque lien et d'effectuer quelques changements sur les équations précédentes afin d'obtenir un modèle plus complet. Par conséquent, les couples calculés tiennent compte de la dynamique du système. Comme nous l'avons déjà fait, ces couples sont envoyés à la loi de contrôle d'attitude avec le même objectif.

Le chapitre quatre traite des lois de contrôle d'attitude et de position pour le système aérien. Pour le premier, l'objectif est de concevoir une loi de commande qui entraîne le quadrotor à la stabilisation d'attitude sous les couples et les moments exercés sur le robot à partir du mouvement du bras manipulateur attaché à sa partie inférieure. Puis, puisque les actionneurs du manipulateur fonctionnent en boucle ouverte, un observateur non linéaire est conçu pour la connaissance de la position angulaire des liens. En raison de l'utilisation de servomoteurs dans ce projet, des non-linéarités et des comportements inconnus peuvent être présents. Dans le cadre de l'amélioration de la précision, l'observateur non linéaire fusionne les données provenant d'un modèle du premier ordre des moteurs et de la position angulaire réelle des liaisons. Avec ceci, une estimation de la position angulaire des liens réels est effectuée. Afin de calculer la position angulaire des liens dans le robot manipulateur, le problème cinématique inverse est adressé. En bref, le problème cinématique inverse permet de déterminer les différentes positions angulaires sur chaque lien par la connaissance de la position de l'effecteur. L'idée principale derrière cette méthode est d'avoir un calcul plus précis des couples exercés par le manipulateur et de fournir plus de robustesse à la loi de contrôle.

Ensuite, un modèle, qui représente la vitesse linéaire et la position et sa dépendance avec le modèle d'attitude est présenté. La dynamique de l'ensemble du système est obtenue avec le formalisme de Newton-Euler. Après cela, la conception d'une commande non

linéaire qui garantit la stabilisation de la position du système est présentée. Plus précisément, une fois que le système a été stabilisé par la loi de contrôle d'attitude, la loi de contrôle de position doit assurer la stabilisation à une position désirée même sous les perturbations provenant du mouvement du bras manipulateur.

Une fois le problème de stabilisation du quadrotor est résolu, une première approche de manipulation aérienne est effectuée par la stabilisation de la position de l'effecteur terminal. L'objectif est d'amener l'effecteur terminal du robot à la position désirée. Et ceci est possible grâce au calcul de position angulaire des liens, effectué par la cinématique inverse.

Enfin, quelques résultats de simulation sont présentés pour les différentes méthodes: la stabilisation sans tenir compte des couples manipulateur, la stabilisation avec le modèle de couple statique, la stabilisation du système avec le modèle dynamique et la stabilisation à l'aide du modèle dynamique et l'observateur Luenberger. Les résultats de simulation de la stabilisation de la position de l'effecteur sont également présentés.

Le cinquième chapitre est consacré à la présentation du système physique utilisé dans ce projet, ainsi que les résultats expérimentaux.

Premièrement, afin de tester les différentes méthodes proposées, l'attitude et la position réelles du système doivent être connues. Pour cela, la salle MOCA (motion capture en anglais) est brièvement présentée.

Ensuite, au cours du développement de ce travail, une variété de véhicules VTOL ont été utilisés. Le premier, un micro-quadrotor FLEXBOT, est introduit. Certains détails, comme son matériel sont présentés. Ce quadrotor a permis de valider les lois de contrôle d'attitude et de position. Cependant, en raison de sa petite taille et par conséquent de sa faible capacité de charge utile, il lui était impossible de décoller lorsque le manipulateur était monté.

Ensuite, il a été décidé de passer à une configuration hexacopter, restant sur les prototypes FLEXBOT, où deux modèles ont été testés. Le premier consistait en un micro-hexacoptère FLEXBOT totalement conçu, mais la faible autonomie a permis quelques tests avec seulement un bras manipulateur à 2 degrés de liberté. Face à ce nouveau problème, le deuxième micro-hexacopter consistait en un prototype FLEXBOT à l'écoute, où certains éléments matériels ont été modifiés, comme la structure et les moteurs. Ce prototype a été utilisé pour valider la méthode d'estimation du couple statique à l'aide du bras manipulateur à 2 degrés de liberté et nous a donné l'opportunité de présenter les résultats lors de la conférence IROS.

Enfin, deux autres prototypes ont été testés, dont les structures ont été entièrement conçues et imprimées en 3D au laboratoire. L'élection des différentes parties de ces prototypes a été faite spécifiquement pour ce projet, permettant d'avoir assez de puissance et

d'autonomie pour réaliser les expériences et prouver l'efficacité des algorithmes proposés. Le choix final pour le prototype réel est également détaillé.

La conception du bras manipulateur consiste en une configuration de coude avec 3 degrés de liberté à articulations pivots, et la structure était également faite maison.

Par la suite, des séries d'expériences ont été réalisées pour tester l'efficacité des méthodes proposées. Les résultats sont présentés dans la dernière partie de ce chapitre, ainsi qu'une étude statistique qui montre la robustesse des algorithmes proposés.

En général, la manipulation aérienne établit une nouvelle norme pour la robotique mobile, puisque les véhicules aériens peuvent non seulement accéder aux zones indisponibles, mais aussi interagir avec l'environnement, faisant de ces systèmes de grandes aides pour une variété d'applications. Ensuite, la communauté de la robotique a proposé plusieurs façons de faire face au problème en donnant des solutions à certaines tâches et défis, mais en laissant d'autres encore sans réponse. De cette manière, le présent travail peut être étendu dans plusieurs directions.

- Concernant le prototype expérimental actuel, le matériau utilisé pour la construction du bras manipulateur, la résine, est très fragile. Ensuite, un nouveau prototype à base de fibre de carbone ou de titane devrait être envisagé, afin d'avoir un bras manipulateur plus résistant et plus rigide.

- En général, trois méthodes ont été proposées pour la stabilisation du véhicule aérien portant le bras manipulateur: le modèle à couple statique, le modèle à couple dynamique et le modèle à couple dynamique aidé par l'observateur Luenberger. Cependant, toutes les dynamiques de l'ensemble du système ne sont pas prises en compte. Ensuite, la proposition d'un modèle dynamique est envisagée et l'idée de l'utilisation d'un contrôleur saturé reste présente, puisqu'il a été montré dans l'état de la technique la nouveauté de la méthode proposée.

- Enfin, étant donné que la capacité de charge des véhicules VTOL est réduite, le concept de contrôle de formation de multi-UAV pour la préhension et le transport peut également être considéré comme une extension du présent travail. A cet effet, de nouvelles lois de contrôle, de détection d'objets et de stratégies de navigation, comme la théorie des graphes seront objets d'étude.

Introduction

A brief discussion on aerial manipulation.

Aerial manipulation has been an active area of research in recent years, mainly because the active tasking of Unmanned Aerial Vehicles (UAV) increases the employability of these vehicles for various applications. With aerial manipulation, a new chapter in the areas of aeronautics, instrumentation, robotics and control is opened, since they are not only remote controlled machines, but also real autonomous systems which can interact with the environment. The recent development of the aerial manipulation has found potential applications in both military and civilian domains. Military applications include border patrolling, mine detection, reconnaissance, etc., while civilian applications are in disaster management, bridge inspection, construction, material delivery, search and rescue, etc. Fig. 1 shows some examples of the cited applications.

Figure 1 – Some examples of applications.

The research on aerial robotics has mainly involved helicopters and Vertical Take-off and Landing (VTOL) architectures. The main advantage of these platforms is their maneuverability and the capacity to perform hovers, which is essential for the applications.

This thesis deals with VTOL aircrafts, where the four rotor helicopter, quadcopter or quadrotor is mainly studied. Such a vehicle is an underactuated mechanical system that has six Degrees of Freedom (DoF) but only four control inputs namely roll, pitch, yaw and thrust and is a dynamically unstable system. The vehicle has the capacity to take-off and land in vertical position. Because of its symmetry, the mathematical model and construction of the vehicle are quite simple.

Regarding the problem of aerial manipulation, the amount of applications are increased, but at the same time the complexity of modeling and control of such a system are equally bigger. One of the biggest challenges arise from their limited payload. Some approaches have tried to solve the problem using multiple robots to carry payloads with grippers or with cables, where their end effectors and grippers have to be lightweight themselves and capable of grasping complex shapes. Another challenge is that the dynamics of the robot are significantly altered by the addition of payloads. However, for payload transport, it is necessary that the robots are able to estimate the inertia of the payload and adapt to it to improve tracking performance. With this, the control of the system becomes into a challenge due to the presence of non-linearities, the aerodynamic forces, the limited payload of the UAV's and now the even more complex moment of inertia which changes due to the movement of the arm manipulator.

Quadrotors are preferred due to their simple mechanical structure and symmetry, as opposed to the traditional helicopters that need complex mechanical controls due to their linkages for the actuation. In fact, it is much simpler to perform a hover with four pushing forces acting at one similar distance from the center of mass instead of one unique pushing force acting on the entire center of mass. All this leads to robustness and modularity, which reduce the maintenance costs. These characteristics also allow the obtention of a simple dynamical model, which enables the possibility for the design of precise control laws, making the system safer if it is used indoor. However, as in most of systems, there are some disadvantages in the use of these kind of vehicles, like their high energy consumption, the limited payload and the poor survivability if one of the propellers is damaged.

Objectives

According to the background and challenges on VTOL vehicles carrying payloads or manipulators, the contribution of the present work is centered on the modeling and the design of a nonlinear control and a formal stability analysis for the asymptotical stabilization of a VTOL vehicle carrying a manipulator arm. More specifically, the different stages for the proposition of a solution of the problem are:

- Propose a general model of a quadcopter carrying a manipulator arm.

- Design of a smooth almost globally asymptotically stable control law for attitude stabilization which takes into account the arm motion effects.
- Design of a globally asymptotical nonlinear controller for the translational dynamics based in the usage of nested and sum of saturation functions in order to take into account the actuators limitations.
- Carry out some experiments in order to validate the proposed control laws.
- Design and implementation of an algorithm which drives the arm manipulator to a desired point, taking into account the movement coming from the quadcopter.

Outline of the dissertation

The first chapter is devoted to the presentation of some mathematical preliminaries and concepts, addressing the problems of rigid body attitude, position representation and open chain manipulators, all these concepts will be used along the present work.

Then, since this work is closely related to the manipulation, we present a brief review of the works carried out in the domain of robot manipulators. The main configurations and their applications are addressed, consequently, a short research of the different control approaches and a classical robot manipulator dynamic model is shown. To do this, we introduce the elbow configuration with 3 degrees of freedom (DOF) and revolute joints. The modeling is conducted with the Euler-Lagrange approach which takes into account the kinetic and potential energy of an object.

The second chapter moves to the domain of aerial systems. A presentation of the different configurations for aerial vehicles is shown, highlighting the importance of the VTOL aircrafts for some specific applications. These vehicles (VTOL) are quickly reviewed, presenting some of the most common configurations and their main characteristics. With this classification, we focus on the multi-rotors systems and we perform a concise research on the control strategies for the stabilization of this kind of vehicles.

To finish with the state of the art, a review of the different configurations of VTOL systems carrying payloads or robot manipulators is conducted, their advantages and disadvantages are analysed depending on the applications and the environment where they interact. Also, a study of the different control strategies is performed, analysing stability, robustness, tracking performance, etc.

Chapter three is focused on modeling of the aerial vehicle carrying the manipulator arm. For this, we present the dynamic model of the aerial vehicle used in this project. The behaviour of the quadcopter can be considered as a rigid body floating on the space under the action of various torques. However, since some singularities can arise with the

Euler angles representation for the attitude, the model is conducted with the quaternion representation. Afterwards, the interaction between the torques and the forces generated by the actuators is presented.

After that, since the movement of the manipulator arm located under the fuselage of the aerial vehicle exerts unknown torques causing disturbances, a novel model for the quadcopter carrying the arm manipulator is introduced. In general, the model of the manipulator arm is conducted in two ways: a static model (only the effects of the gravity are taken into account), and a partial dynamic model (a part of the motor dynamics is taken into account).

For the first one, the dynamics of the arm manipulator are neglected. However, through some proposed equations it is possible to estimate the static torques by the knowledge of the links angular position and some physical parameters of the manipulator, like the mass, length and inertial moment of each link. This computed torque is sent to the attitude control to compensate the undesired forces.

The second model takes into account a part of the dynamics of the arm manipulator. Since the physical system makes the use of servomotors, it is possible to model these ones as first order systems. With this, it is possible to compute the position and velocity on each link and perform some changes on the previous equations in order to obtain a more complete model. Consequently, the computed torques take into account the dynamics of the system. As we did before, these torques are sent to the attitude control law with the same objective.

Chapter four deals with the attitude and position control laws for the aerial system. For the first one, the objective is to design a control law which drives the quadrotor to attitude stabilization under the torques and moments exerted to the robot from the movement of the manipulator arm attached to its lower part. Then, since the manipulator actuators work in open-loop, a Luenberger observer is designed for the knowledge of the links angular position. Due to the usage of servomotors in this project, non-linearities and unknown behaviours can be present. In pursuance of precision improvement, the Luenberger observer fuses the data coming from a first order model of the motors and the actual links angular position. With this, an estimation of the real links angular position is performed. In order to compute the angular position of the links in the robot manipulator the inverse kinematic problem is addressed. Briefly speaking, the inverse kinematic problem allows us to determine the different angular positions on each link by the knowledge of the end-effector position. The main idea behind this method is to have a more precise computation of the torques exerted by the manipulator and to provide more robustness to the control law.

Then, a model, which represents the linear velocity and position and its dependence

with the attitude model is presented. The dynamics of the whole system is obtained with the Newton-Euler formalism. After that, the design of a nonlinear control which guarantees the position stabilization of the system is presented. More specifically, once the system has been stabilized by the attitude control law, the position control law should ensure the stabilization to a desired position even under the disturbances coming from the movement of the manipulator arm.

Once the quadrotor stabilization problem is solved, a first approach of aerial manipulation is conducted by the end-effector position stabilization. The objective is to drive the robot manipulator end-effector to a desired position. And this is possible through the links angular position computation, carried out by the inverse kinematics.

Finally, some simulation results are presented for the different methods: the stabilization without taking into account the manipulator torques, the stabilization with the static torque model, the stabilization of the system with the dynamic torque model and the stabilization using the dynamic torque model and the Luenberger observer. Simulation results of the end-effector position stabilization are presented as well.

The fifth chapter is devoted to the presentation of the physical system used in this project, as well as the experimental results.

First, in order to test the different proposed methods, the actual attitude and position of the system must be known. For this, the MOCA (motion capture) room is briefly presented.

Then, during the development of this work a variety of VTOL vehicles were used. The first one, a FLEXBOT micro-quadrotor, is introduced. Some details, like its hardware are presented. This quadrotor allowed to validate the attitude and position control laws. However, due to its small size and consequently its poor payload capacity, it was impossible for it to take off when the manipulator was mounted.

Then, it was decided to move to a hexacopter configuration, resting on the FLEXBOT prototypes, where two models were tested. The first one consisted on a totally designed FLEXBOT micro-hexacopter, however the poor autonomy allowed some tests with only a 2 DOF manipulator arm. Facing this new problem, the second micro-hexacopter consisted on a tuned FLEXBOT prototype, where some hardware elements where changed, like the frame and the motors. This tuned prototype was used to validate the static torque estimation method using the 2 DOF manipulator arm and it gave us the opportunity to present the results at the IROS conference.

Finally, two more prototypes were tested, whose structures were totally designed and 3d printed at the laboratory. The election of the different parts of these prototypes were specifically made for this project, allowing to have enough power and autonomy to carry out the experiments and prove the effectiveness of the proposed algorithms. The final

choice for the actual prototype is detailed as well.

The design of the arm manipulator consists on an elbow configuration with 3 DOF and revolute joints, and the structure was equally home-made.

Afterwards, series of experiments were carried out to test the effectiveness of the proposed methods. The results are shown in the final part of this chapter, as well as a statistical study which shows the robustness of the proposed algorithms.

The sixth and final chapter contains some conclusions about the work, also some possible future works are barely presented.

Chapter 1
Preliminaries

In this chapter, some mathematical preliminaries, that will be used throughout the document are presented. The position and attitude representation are introduced. Then, the quaternion representation is emphasized, since this one is used along the work. After that, we present some essential concepts for the modeling and control of the arm manipulators, like the forward and inverse kinematics, the opened chain manipulator concept and the equations of motion of Lagrange to obtain the general dynamic model.

Once the preliminaries have been addressed, we present the state of the art of the robot manipulators, focusing on some of the classical configurations and the most used actuation for their operation. We introduce the problem of control and manipulation with robotic arms, presenting at the same time some of the approaches used to tackle the topic. To finish with the state of the art of the robot manipulators, the general model of a three degrees of freedom arm manipulator, composed by three links, one moving along the z axe, and the rest moving along the y axe is presented.

1.1 Position representation

Robot tasks are often defined by the use of Cartesian coordinates. Let consider the scheme in Fig. 1.1. It is possible to specify the coordinates of the point p with respect to either frame $o_0x_0y_0$ or frame $o_1x_1y_1$. So that the reference frame will always be clear, a notation in which a superscript is used to denote the reference frame is adopted

Geometrically, a point corresponds to a specific location in space, and a vector specifies a direction and a magnitude. Vectors can be used, for example, to represent displacements or forces. Therefore, while the point p is not equivalent to the vector v_1, the displacement from the origin o_0 to the point p is given by the vector v_1. Under this convention, it is clear that points and vectors are not equivalent, since points refer to specific locations in space, but a vector can be moved to any location in space. Then, two vectors are said to

be equal if they have the same direction and the same magnitude.

When assigning coordinates to vectors, the same notational convention is used as when assigning coordinates to points. Thus, v_1 and v_2 are geometric entities that are invariant with respect to the choice of coordinate systems, but the representation by coordinates of these vectors depend directly on the choice of reference coordinate frame.
Using this convention, an expression of the form $v_1^1 + v_2^2$ where v_1^1 and v_2^2 are as in Fig. 1.1, is not defined since the frames $o_0x_0y_0$ and $o_1x_1y_1$ are not parallel. Thus, a clear need appears, not only for a representation system that allows points to be expressed with respect to various coordinate systems, but also for a mechanism that allows to transform the coordinates of points that are expressed in one coordinate system into the appropriate coordinates with respect to some other coordinate frame.

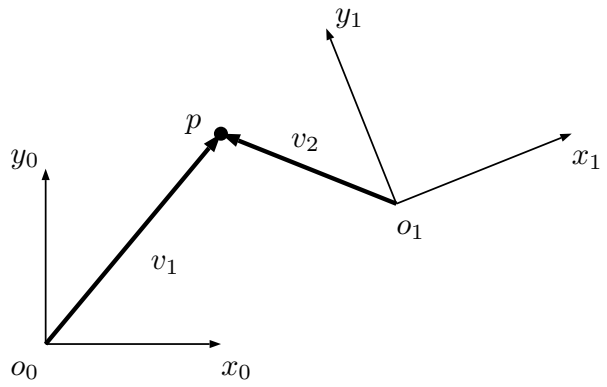


Figure 1.1 – Two coordinate frames, a point p, and two vectors v_1 and v_2, (Spong et al. [2004]).

1.2 Rigid body attitude representation

Consider two orthogonal right-handed coordinate frames: the body coordinate frame, $B(x_b, y_b, z_b)$ located at the center of mass of the rigid body (CG), and the inertial coordinate frame, $N(x_n, y_n, z_n)$, located at some point in the space (for instance, the earth NED frame). This coordinate system is showed in Fig. 1.2.

The body attitude in the space can be represented in many ways, each one with their advantages and disadvantages, depending mainly on the application.

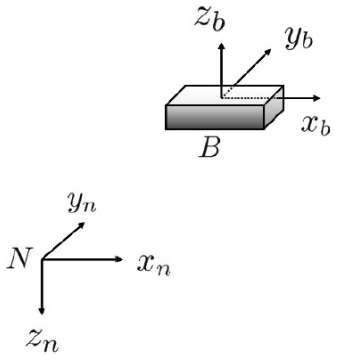


Figure 1.2 – Inertial and mobile frame of a rigid body.

Definition 1.2.1 *The movement of a rigid body with reference frame B relative to a rigid body or reference frame N is called a simple rotation of B in N, if there is a line L, called rotation axis, where the orientation relative to B and N keeps the same between the start and the end of the movement.*

Theorem 1.2.2 *(Rotation Euler Theorem). Every relative change in orientation between two rigid bodies with the two coordinate systems B and N can be produced by one simple rotation of B over N.*

Let $\vec{b}$ and $\vec{r}$ be the coordinates of a vector $\vec{X}$ in B and N respectively. Vector $\vec{b}$ can be written in terms of vector $\vec{r}$.

Let $\vec{e} = [e_1\ e_2\ e_3]$ be a unit vector collinear to the rotation axis L around which B is rotated by an angle β. Consequently, $\vec{b}$ is obtained by

$$\vec{b} = \cos\beta\vec{r} + (1 - \cos\beta)\vec{e}\,\vec{e}^{\,T}\vec{r} - \sin\beta\vec{e} \times \vec{r} \tag{1.1}$$

In fact, the coordinates $\vec{b}$ and $\vec{r}$ are linked by means of the following linear transformation:

$$\vec{b} = C\vec{r} \tag{1.2}$$

Matrix C can be token as an operator which takes a fixed vector $\vec{r}$ expressed in N and is expressed in B. From (1.1)

$$C = \cos\beta I_3 + (1 - \cos\beta)\vec{e}\,\vec{e}^{\,T} - \sin\beta[\vec{e}^{\,\times}] \tag{1.3}$$

where I_3 represents the identity matrix of dimension three and $[\xi^\times]$ represents the skew-symmetric matrix, given by:

$$[\xi^\times] = \begin{pmatrix} \xi_1 \\ \xi_2 \\ \xi_3 \end{pmatrix}^\times = \begin{pmatrix} 0 & \xi_3 & \xi_2 \\ -\xi_3 & 0 & \xi_1 \\ \xi_2 & -\xi_1 & 0 \end{pmatrix} \tag{1.4}$$

Matrix $C \in \mathbb{R}^{3\times3}$ identifies the orientation of the moving frame B with respect to the inertial frame N and it allows the coordinate transformation of a vector system into another one. This matrix is known as the direction cosine matrix (DCM), rotation matrix or attitude matrix.

Rotation matrix

Rotation matrix C belongs to the subspace of orthogonal matrices of dimension three, called special orthogonal group, denoted by $S0(3)$, and defined by

$$S0(3) = C|C \in \mathbb{R}^{3\times3}, C^TC = I_3, \det(C) = 1 \tag{1.5}$$

In a rotation matrix C, each element c_{ij} is a direct cosine, given by:

$$C = \begin{pmatrix} c_{11} & c_{12} & c_{13} \\ c_{21} & c_{22} & c_{23} \\ c_{31} & c_{32} & c_{33} \end{pmatrix} \tag{1.6}$$

where

$$c_1 = \begin{pmatrix} c_{11} \\ c_{21} \\ c_{31} \end{pmatrix} \quad c_1 = \begin{pmatrix} c_{12} \\ c_{22} \\ c_{32} \end{pmatrix} \quad c_1 = \begin{pmatrix} c_{13} \\ c_{23} \\ c_{33} \end{pmatrix} \tag{1.7}$$

Consequently,

$$C = \begin{pmatrix} c_1 & c_2 & c_3 \end{pmatrix} \tag{1.8}$$

where

$$c_i^T c_i = 1 \;\text{ and }\; c_i^T c_j = 0 \;\; \forall i \neq j \tag{1.9}$$

Rotational velocity

Suppose that a rotation matrix C is time varying, so that $C = C(t) \in S0(3)$ for every $t \in \mathbb{R}$. Assuming that $C(t)$ is continuously differentiable as a function of t. An argument identical to the one in the previous section shows that the time derivative $\dot{C}(t)$ of $C(t)$ is given by

$$\dot{C}(t) = [\vec{\omega}\,^{\times}]C(t) \tag{1.10}$$

where $\vec{\omega}$ is the angular velocity of the rotating frame with respect to the fixed frame at time t.

Euler angles and Roll, Pitch and Yaw angles

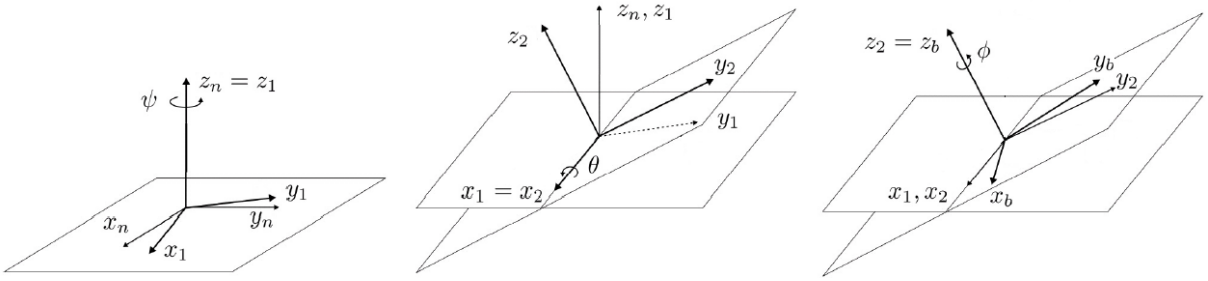


Figure 1.3 – Euler angles.

A common method of specifying a rotation matrix in terms of three independent quantities is to use the so-called Euler Angles (ψ, θ, φ). Consider again the fixed coordinate frame $N(x_n, y_n, z_n)$ and the rotated frame (x_b, y_b, z_b), shown in Fig. 1.3.

It is possible to specify the orientation of the frame (x_1, y_1, z_1) relative to the frame $N(x_n, y_n, z_n)$ by the three angles, and it can be obtained by three successive rotations as follows: first rotate about z-axis by the angle ϕ, next, rotate about the current y-axis by the angle θ. Finally, rotate about the current z-axis by the angle ψ. In terms of the basic rotation matrices the resulting rotational transformation C_1^0 can be generated as the product

$$\begin{aligned} C_1^0 = C_{z,\phi}, C_{y,\theta}, C_{z,\psi} &= \begin{bmatrix} c_\phi & -s_\phi & 0 \\ s_\phi & c_\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} c_\theta & 0 & s_\theta \\ 0 & 1 & 0 \\ -s_\theta & 0 & c_\theta \end{bmatrix} \begin{bmatrix} c_\psi & -s_\psi & 0 \\ s_\psi & c_\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \\ &= \begin{bmatrix} c_\phi c_\theta c_\psi - s_\phi s_\psi & -c_\phi c_\theta s_\psi - s_\phi c_\psi & c_\phi s_\theta \\ s_\phi c_\theta c_\psi + c_\phi s_\psi & -s_\phi c_\theta s_\psi + c_\phi c_\psi & s_\phi s_\theta \\ -s_\theta c_\psi & s_\theta s_\psi & c_\theta \end{bmatrix} \end{aligned} \tag{1.11}$$

A rotation matrix C can also be described as a product of successive rotations about the principal coordinate axes x_n, y_n and z_n taken in a specific order. These rotations define the roll, pitch and yaw angles. The order of rotations is specified as $z - y - x$, in other words, first a yaw about z_n by an angle ϕ, then pitch about the y_n by an angle θ, and finally roll about the x_n by an angle ψ. Since the successive rotations are relative to

the fixed frame, the resulting transformation matrix is given by

$$\begin{aligned} C_1^0 = C_{z,\phi}C_{y,\theta}C_{x,\psi} &= \begin{bmatrix} c_\phi & -s_\phi & 0 \\ s_\phi & c_\phi & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} c_\theta & 0 & s_\theta \\ 0 & 1 & 0 \\ -s_\theta & 0 & c_\theta \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & c_\psi & -s_\psi \\ 0 & s_\psi & c_\psi \end{bmatrix} \\ &= \begin{bmatrix} c_\phi c_\theta & -s_\phi c_\psi + c_\phi s_\theta s_\psi & s_\phi s_\psi + c_\phi s_\theta c_\psi \\ s_\phi c_\theta & c_\phi c_\psi + s_\phi s_\theta s_\psi & -c_\phi s_\psi + s_\phi s_\theta c_\psi \\ -s_\theta & c_\theta s_\psi & c_\theta c_\psi \end{bmatrix} \end{aligned} \tag{1.12}$$

Now, let $\vec{\omega} = [\ \omega_x \quad \omega_y \quad \omega_z\ ]$ be the angular velocity of the body in the reference frame B with respect to the reference frame N. Then, the kinematic equation is given by Fossen [1994]:

$$\begin{pmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{pmatrix} = \begin{pmatrix} 1 & \tan\theta\sin\phi & \tan\theta\sin\phi \\ 0 & \cos\phi & -\sin\phi \\ 0 & \frac{\sin\phi}{\cos\theta} & \frac{\cos\phi}{\cos\theta} \end{pmatrix} \begin{pmatrix} \omega_x \\ \omega_y \\ \omega_z \end{pmatrix} \tag{1.13}$$

Quaternions

There is another representation for the attitude, which is the unit *quaternion* or Euler parameters. The quaternion is an alternative solution to the Euler theorem, which states that one rotation in the space can be performed by a rotation β over a rotation axis $\vec{e}$. The unit quaternion is composed by a unit vector $\vec{e}$ called Euler axis, and a rotation angle β around this axis. It is defined by:

$$q := \begin{pmatrix} \cos\frac{\beta}{2} \\ \vec{e}\sin\frac{\beta}{2} \end{pmatrix} = \begin{pmatrix} q_0 \\ \vec{q} \end{pmatrix} \in \mathbb{H} \tag{1.14}$$

where

$$\mathbb{H} = \{q | q_0^2 + \vec{q}^{\,T}\vec{q} = 1,\ q = \begin{pmatrix} q_0 \\ \vec{q} \end{pmatrix},\ q_0 \in \mathbb{R},\ \vec{q} \in \mathbb{R}^3\} \tag{1.15}$$

where $\vec{q} = (q_1 \quad q_2 \quad q_3)^T \in \mathbb{R}^3$ and $q_0 \in \mathbb{R}$ are known as the vector and scalar parts of the quaternion respectively. The identity quaternion and the conjugated quaternion are given by:

$$q_{id} = [1 \quad \vec{0}_{3\times1}^{\,T}]^{\,T} \quad \bar{q} = [q_0 \quad -\vec{q}^{\,T}]^{\,T} \tag{1.16}$$

Since the quaternion is unitary, $q^{-1} = \bar{q}$. The product of two quaternions $q_1 = [q_{1_0} \quad \vec{q_1}^T]^T$ and $q_2 = [q_{2_0} \quad \vec{q_2}^T]^T$ is defined by:

$$q_1 \otimes q_2 = \begin{pmatrix} q_{1_0} & -\vec{q}_1^{\,T} \\ \vec{q}_1 & I_3 q_{1_0} + [\vec{q}^{\,\times}] \end{pmatrix} \begin{pmatrix} q_{2_0} \\ \vec{q}_2 \end{pmatrix} \tag{1.17}$$

Now, let b_q and r_q be the quaternions associated to vectors $\vec{b}$ and $\vec{r}$, defined by:

$$b_q = [0 \;\; \vec{b}^{\,T}]^{\,T} \quad r_q = [0 \;\; \vec{r}^{\,T}]^{\,T} \tag{1.18}$$

These two quaternions are linked by the next relation:

$$b_q = q \otimes r_q \otimes q^{-1} = q \otimes r_q \otimes \bar{q} \tag{1.19}$$

Rotation matrix C can be expressed in terms of quaternions by:

$$C = C(q) = (q_0^2 - \vec{q}^{\,T}\vec{q})I_3 + 2(\vec{q}\,\vec{q}^{\,T} - q_0[\vec{q}^{\,\times}]) \tag{1.20}$$

from where

$$C(q) = \begin{pmatrix} 2(q_0^2+q_1^2)-1 & 2(q_1q_2+q_0q_3) & 2(q_1q_3-q_0q_2) \\ 2(q_1q_2-q_0q_3) & 2(q_0^2+q_2^2)-1 & 2(q_0q_1+q_2q_3) \\ 2(q_0q_2+q_1q_3) & 2(q_2q_3-q_0q_1) & 2(q_0^2+q_3^2)-1 \end{pmatrix} \tag{1.21}$$

The relation between the vectors $\vec{b}$ and $\vec{r}$ is:

$$\vec{b} = C(q)\vec{r} \tag{1.22}$$

Denoting by $w = (\omega_1 \;\; \omega_2 \;\; \omega_3)^T$ the angular velocity vector of the body coordinate frame, B relative to the inertial coordinate frame N expressed in B, the kinematics equation is given by

$$\begin{pmatrix} \dot{q}_0 \\ \dot{\vec{q}} \end{pmatrix} = \frac{1}{2}\begin{pmatrix} -\dot{q}^T \\ I_3 q_0 + [\vec{q}^{\times}] \end{pmatrix} w = \frac{1}{2}\Xi(q)w \tag{1.23}$$

Rigid motions

Definition 1.2.3 *A rigid motion is an ordered pair (d,C) where* $d \in \mathbb{R}^3$ *and* $C \in SO(3)$. *The group of all rigid motions is known as the* ***Special Euclidean Group*** *and is denoted by* $SO(3)$. *It is denoted that* $SO(3) = \mathbb{R}^3 \times SO(3)$.

A rigid motion is a pure translation together with a pure rotation.

Homogeneous transformations

The combination of position and orientation representations allows the formulation of homogeneous transformations, used in the modeling of arm manipulators. Then, the trans-

formation matrices with the form:

$$H = \begin{bmatrix} C & d \\ 0 & 1 \end{bmatrix} \tag{1.24}$$

are called **homogeneous transformations**. Then, a homogeneous transformation is a representation of a rigid motion where the system $S0(3)$ is used interchangeably to represent the set of rigid motions and the set of all 4×4 matrices given in (1.24).

A set of basic homogeneous transformations generating $S0(3)$ is given by

$$Trans_{x,a} = \begin{bmatrix} 1 & 0 & 0 & a \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}; \quad Rot_{x,a} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & c_a & -s_a & 0 \\ 0 & s_a & c_a & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{1.25}$$

$$Trans_{y,b} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & b \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}; \quad Rot_{y,b} \begin{bmatrix} c_\beta & 0 & s_\beta & 0 \\ 0 & 1 & 0 & 0 \\ -s_\beta & 0 & c_\beta & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{1.26}$$

$$Trans_{z,c} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & c \\ 0 & 0 & 0 & 1 \end{bmatrix}; \quad Rot_{x,a} \begin{bmatrix} c_\gamma & -s_\gamma & 0 & 0 \\ s_\gamma & c_\gamma & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{1.27}$$

for translation and rotation about the x, y, z axis respectively.
The most general homogeneous transformation that is considered is given by

$$H_1^0 = \begin{bmatrix} n_x & s_x & a_x & d_x \\ n_y & s_y & a_y & d_y \\ n_z & s_z & a_z & d_z \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} n & s & a & d \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{1.28}$$

In the above equation $n = (n_x, n_y, n_z)^T$ is a vector representing the direction of X_1 in the $o_0x_0y_0$ system, $s = (s_x, s_y, s_z)^T$ represents the direction of y_1, and $a = (a_x, a_y, a_z)$ represents the direction of z_1. $d = (d_x, d_y, d_z)^T$ represents the vector from the origin o_0 to the origin o_1 expressed in the frame $o_0x_0y_0z_0$.

Attitude error

Two orientations of a rigid body are considered, described by rotation matrices C_1 and C_2 respectively. Then, the relative attitude between these two orientations is computed by:

$$C_r = C_1^{-1} C_2 \tag{1.29}$$

In fact, C_r represents an operator of orientation which rotates C_2 about C_1. From here, the relative orientation is used in the estimation and in the orientation control as *attitude error*. With this, $C_d = C_1$ is the desired attitude of a rigid body and $C = C_2$ is the actual attitude of the body. Consequently, the attitude error is computed by:

$$C_e = C_d^{-1} C \tag{1.30}$$

If the attitude error is zero, then, $C_e = I_3$. When the unitary quaternion is used to represent the attitude of the body, the relative orientation between q_1 and q_2 is expressed by:

$$q_r = q_1^{-1} \otimes q_2 = \begin{pmatrix} q_{1_0} & \vec{q}_1^{\ T} \\ -\vec{q}_1 & I_3 q_{1_0} - [\vec{q}^{\ \times}] \end{pmatrix} \begin{pmatrix} q_{2_0} \\ \vec{q}_2 \end{pmatrix} = \bar{q}_1 \otimes q_2 \tag{1.31}$$

The geodesic metric is given by the next expression:

$$\beta_r = 2|\arccos(q_{r_0})| \tag{1.32}$$

This metric represents the smallest angle of rotation between attitude q_1 and the attitude q_2. The Euclidean distance between the two unit quaternions gives an approximation of the geodesic metric:

$$\frac{2}{\pi}\beta_r \leq 2 \parallel q_1 - q_2 \parallel_2 \leq \beta_r \tag{1.33}$$

Also, when β_r is enough small, it is possible to do the next approximation:

$$\beta_r \approx 2 \parallel q_1 - q_2 \parallel_2 \tag{1.34}$$

For the case of the attitude control law, if $q_d = q_1$ is the desired attitude of the body and $q = q_2$ is the real attitude of the body, the attitude error is given by:

$$q_e = q_d^{-1} \otimes q = \bar{q}_d \otimes q \tag{1.35}$$

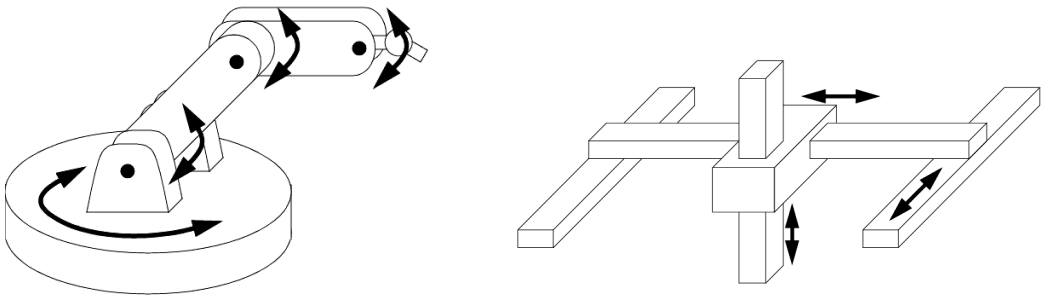
When the attitude error is zero, the error quaternion has two possible values:

$$q_e = [\pm 1 \quad 0]^T \tag{1.36}$$

This is due to the non bijection quaternion with the group $SO(3)$.

1.3 Forward kinematics of robot manipulators

This part of the chapter is devoted to the **forward** kinematic equations for rigid manipulators. The forward kinematics is the relationship between each joint in the robot manipulator and the position and orientation of the end-effector. In other words, the problem of forward kinematics is to determine the position and orientation of the end-effector, if the values of the joint variables are available. The joint variables are the angles between the links in the case of revolute or rotational, and the link extension in the case of prismatic or sliding joints, see Fig. 1.4. Since that, this work is centered on the robot manipulators with prismatic joints.



(a) (b)

Figure 1.4 – (a)Revolute joints and (b)Prismatic joints

A robot with n joints will have $n+1$ links, since each joint connects two links. With the i^{th} joint, a joint variable is associated, denoted by q_i. For a revolute joint, q_i is the angle of rotation, and in order to perform a kinematic analysis, a coordinate frame is attached to each link, in this case $O_i(x_i, y_i, z_i)$ to link i. So, whatever motion the robot executes, the coordinates of each point of link i are constant when expressed in the i^{th} frame. The frame $O_0(x_0, y_0, z_0)$ represents the coordinate frame attached to the base of the robot, and at the same time is taken as the inertial frame. Fig. 1.5, illustrates the idea of attaching coordinates frames to links in a robot manipulator.

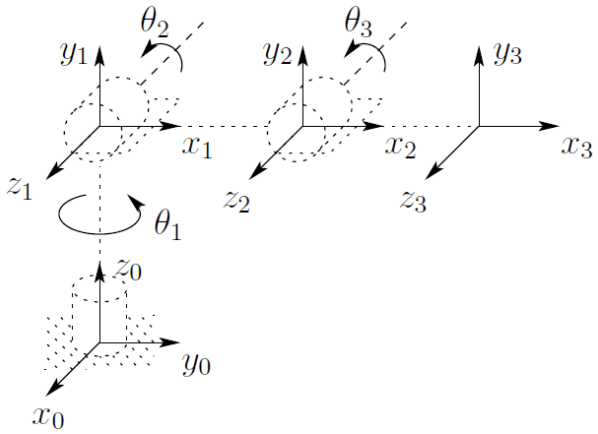


Figure 1.5 – Coordinate frames attached to robot manipulator, (Spong et al. [2004]).

Now, A_i is the homogeneous transformation matrix and it represents the position and orientation of $O_i(x_i, y_i, z_i)$ with respect to $O_{i-1}(x_{i-1}, y_{i-1}, z_{i-1})$. Since the configuration of the robot is changing, A_i is not constant, consequently A_i is a function of only a single joint variable, namely q_i.

$$A_i = A_i(q_i) \tag{1.37}$$

The homogeneous transformation that expresses the position and orientation of $O_j(x_j, y_j, z_j)$ with respect to $O_i(x_i, y_i, z_i)$ is called **transformation matrix**, and is denoted by T_j^i. Then:

$$\begin{aligned} T_j^i &= A_{i+1}A_{i+2}\dots A_{j-1}A_j \text{ if } i < j \\ T_j^i &= I \text{ if } i = j \\ T_j^i &= (T_i^j)^{-1} \text{ if } j > i \end{aligned} \tag{1.38}$$

Now, for the problem of the position of the end-effector of the robot manipulator, this is independent of the configuration of the robot. Denote the position and orientation of the end-effector with respect to the inertial frame or the base frame by a three-vector o_n^0 (which gives the coordinates of the origin of the end-effector frame with respect to the base frame) and the 3 rotation matrix R_n^0 and define the homogeneous transformation matrix

$$H = \begin{bmatrix} R_n^0 & o_n^0 \\ 0 & 1 \end{bmatrix} \tag{1.39}$$

Then, the position and orientation of the end-effector in the inertial frame are given by

$$H = T_n^0 = A_1(q_1)\dots A_n(q_n) \tag{1.40}$$

Each homogeneous transformation A_i is of the form

$$A_i = \begin{bmatrix} R_i^{i-1} & o_i^{i-1} \\ 0 & 1 \end{bmatrix} \tag{1.41}$$

Hence

$$T_j^i = A_{i+1} \dots A_j = \begin{bmatrix} R_j^i & o_j^i \\ 0 & 1 \end{bmatrix} \tag{1.42}$$

Denavit Hartenberg representation

A commonly used convention for selecting frames of reference in robotic applications is the Denavit-Hartenberg, or D-H convention. In this convention, each homogeneous transformation A_i is represented as a product of four basic transformations

$$\begin{aligned} A_i =& Rot_{z,\theta_i} Trans_{z,d_i} Trans_{x,a_i} Rot_{x,a_i} \\ =& \begin{bmatrix} c_{\theta_i} & -s_{\theta_i} & 0 & 0 \\ s_{\theta_i} & c_{\theta_i} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & a_i \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & c_{a_i} & -s_{a_i} & 0 \\ 0 & s_{a_i} & c_{a_i} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\ =& \begin{bmatrix} c_{\theta_i} & -s_{\theta_i} c_{a_i} & s_{\theta_i} s_{a_i} & a_i c_{\theta_i} \\ s_{\theta_i} & s_{\theta_i} c_{a_i} & -c_{\theta_i} s_{a_i} & a_i s_{\theta_i} \\ 0 & s_{a_i} & c_{a_i} & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \end{aligned} \tag{1.43}$$

where the four quantities $\theta_i, a_i, d_i, \alpha_i$ are parameters associated with the link i and joint i. These parameters in (1.43) are generally known as **link length**, **link twist**, **link offset** and **joint angle**, respectively. Since the matrix A_i is a function of a single variable, three of the above four quantities are constant for a given link and the fourth parameter θ_i is the joint variable.

1.3.1 The product of exponentials formula

Another geometric description of the kinematics can be obtained by using the fact that motion of the individual joints is generated by a twist associated with the joint axis. If ξ is a twist, then the rigid motion associated with rotating and translating along the axis of the twist is given by

$$g_{ab}(\theta) = e^{\hat{\xi}\theta} g_{ab}(0) \tag{1.44}$$

If ξ corresponds to a prismatic joint, then $\theta \in \mathbb{R}$ is the amount of translation; otherwise,

$\theta \in \mathbb{S}^1$ measures the angle of rotation about the axis.

In order to find the forward kinematics map for any open chain manipulator with n degrees of freedom, let S be a frame attached to the base of the manipulator and T be a frame attached to the last link of the manipulator. Define the *reference configuration* of the manipulator to be the configuration of the manipulator corresponding to $\theta = 0$ and let $g_{st}(0)$ represent the rigid body transformation between T and S when the manipulator is in its reference configuration. For each joint construct a twist ξ_i which corresponds to the screw motion of the ith joint with all other joint angles held fixed at $\theta_j = 0$. For a revolute joint, the twist ξ_i has the form

$$\xi_i = \begin{bmatrix} -w_i \times q_i \\ w_i \end{bmatrix} \tag{1.45}$$

where $w_i \in \mathbb{R}^3$ is a unit vector in the direction of the twist axis and $q_i \in \mathbb{R}^3$ is any point on the axis. For a prismatic joint,

$$\xi_i = \begin{bmatrix} v_i \\ 0 \end{bmatrix} \tag{1.46}$$

where $v_i \in \mathbb{R}^3$ is a unit vector pointing in the direction of translation. All vectors and points are specified related to the base coordinate frame S.

Combining the individual joint motions, the forward kinematics map, $g_{st} : Q \to SE(3)$, is given by:

$$g_{st}(\theta) = e^{\hat{\xi}_1\theta_1} e^{\hat{\xi}_2\theta_2} \cdots e^{\hat{\xi}_n\theta_n} g_{st}(0) \tag{1.47}$$

The ξ_i's must be numbered sequentially starting form the base, but $g_{st}(\theta)$ gives the configuration of the tool frame independently of the order in which the rotations and translations are actually performed. Equation 1.47 is called the *product of exponentials formula* Murray et al. [1994], for the manipulator forward kinematics.

1.4 Inverse kinematics of robot manipulators

The problem of the inverse kinematics is to find the joint variables in terms of the end-effector position and orientation. For this, given 4 homogeneous transformation

$$H = \begin{bmatrix} R & o \\ 0 & 1 \end{bmatrix} \tag{1.48}$$

with $R \in SO(3)$, find (one or all) solutions of the equation

$$T_n^0(q_1, \ldots, q_n)) = H \tag{1.49}$$

where

$$T_n^0(q_1, \ldots q_n) = A_1(q_1) \cdots A_n(q_n)) \tag{1.50}$$

Here, H represents the desired position and orientation of the end-effector, and the objective is to find the values for the joint variables $q_1, \ldots, q_n$ so that $T_n^0(q_1, \ldots, q_n) = H$.

Equation (1.49) results in twelve nonlinear equations with n unknown variables, which can be written as

$$T_{ij}(q_1, \ldots, q_n) = h_{ij}, \quad i = 1, 2, 3, \quad j = 1, \ldots, 4 \tag{1.51}$$

where T_{ij} refers to the twelve nontrivial entries of T_n^0 and H, respectively.

1.5 Statics vs Dynamics

In a discussion of static vs. dynamic torque, it is often easier to start with an understanding of the difference between a static and dynamic force. To put it simply, a dynamic force involves acceleration, where a static force does not. The relationship between dynamic force and acceleration is described by Newton's second law: $F = ma$ (force equals mass times acceleration). The force required to stop a car with its substantial mass would be a dynamic force, as the car must be decelerated. The force exerted by the brake caliper in order to stop that car would be a static force because there is no acceleration of the brake pads involved.

Torque is just a rotational force, or a force through a distance. From the previous discussion, it is considered static if it has no angular accelerations. The torque exerted by a clock spring would be a static torque, since there is no rotation and hence no angular acceleration. The torque transmitted through the drive axle of a car as it cruises down the highway (at a constant speed) would be an example of a rotating static torque, because even though there is rotation, at a constant speed there is no acceleration. The torque produced by the cars engine will be both, static and dynamic, depending on where it is measured.

The torque required to crank up the windows in a car would be an example of a static torque, even though there is a rotational acceleration involved, because both the acceleration and the rotational inertia of the crank are very small and the resulting dynamic torque will be negligible when compared to the frictional forces involved in the window

movement.

1.6 Dynamic modeling of robot manipulators

Robot manipulators can be described mathematically in different ways. The problem of kinematics is to describe the motion of the manipulator without the consideration of forces and torques causing the motion. These equations determine the position and orientation of the end-effector given the values for the join variables (forward kinematics), and as the opposite the values of the joint variables given the position and orientation of the end-effector (inverse kinematics).

Dynamics modelling means deriving equations that explicitly describes the relationship between force and motion. These equations are important to consider in the design of robots, in simulation and animation of robot motion, and in the design of control algorithms.

Computing the dynamics of robot manipulators can be challenging. Researchers have discovered different approaches, where in general there are two methods available: the Euler-Lagrange formulation and the Newton-Euler formulation. In the standard Euler-Lagrange formulation the manipulator is treated as a whole system, and the system is analyzed based on its kinetic and potential energy. The Newton-Euler formulation is quite different because each link of the manipulator is treated in turn. The resulting dynamic model is the same for both methods and can be written in matrix form as:

$$M(\theta)\ddot{\theta} + C(\theta, \dot{\theta})\dot{\theta} + G(\theta, \dot{\theta}) = \tau \tag{1.52}$$

where θ is the vector of joint variables, τ is the vector of torques, M is the inertia matrix, C are the centrifugal and Coriolis terms and G includes the gravity effects.

In this work the Euler-Lagrange approach is used in order to obtain the manipulator arm dynamic model.

1.6.1 Equations of motion for an open-chain manipulator

As described before, a robot manipulator is composed by a set of links connected together by various joints to form a kinematic chain, where the joints are revolute or prismatic. Then, let $\theta \in \mathbb{R}^n$ be the joint angles for an open-chain manipulator. The Lagrangian is of the form

$$\mathcal{L}(\theta, \dot{\theta}) = \frac{1}{2}\dot{\theta}^T M(\theta)\dot{\theta} - U(\theta) \tag{1.53}$$

where $M(\theta)$ is the manipulator inertia matrix and $U(\theta)$ is the potential energy due to gravity. It will be convenient to express the kinetic energy as a sum

$$\mathcal{L}(\theta, \dot{\theta}) = \frac{1}{2} \sum_{i,j=1}^{n} M_{i,j}(\theta)\dot{\theta}_i\dot{\theta}_j - U(\theta) \tag{1.54}$$

The equations of motion are given by substituting into Lagrange's equations,

$$\frac{d}{dt}\frac{\partial \mathcal{L}}{\partial \dot{\theta}_i} - \frac{\partial \mathcal{L}}{\partial \theta_i} = \tau_i \tag{1.55}$$

where τ_i represents the actuator torque and other nonconservative, generalized forces acting on the *ith* joint. Using the previous equation, it can be obtained

$$\begin{aligned} \frac{\partial \mathcal{L}}{\partial \dot{\theta}_i} =& \frac{d}{dt}\left(\sum_{j=1}^{n}(M_{ij}\dot{\theta}_j)\right) = \sum_{j=1}^{n}(M_{ij}\ddot{\theta}_j + \dot{M}_{ij}\dot{\theta}_j) \\ \frac{\partial \mathcal{L}}{\partial \theta_i} =& \frac{1}{2}\sum_{j,k=1}^{n}\frac{\partial M_{kj}}{\partial \theta_i}\dot{\theta}_k\dot{\theta}_j - \frac{\partial U}{\partial \theta_i} \end{aligned} \tag{1.56}$$

The $\dot{M}_{ij}$ term can now be expanded in terms of partial derivatives, which yields

$$\sum_{j=1}^{n} M_{ij}(\theta)\ddot{\theta}_j + \sum_{j,k=1}^{n}\left(\frac{\partial M_{ij}}{\partial \theta_k}\dot{\theta}_j\dot{\theta}_k - \frac{1}{2}\frac{\partial M_{kj}}{\partial \theta_i}\dot{\theta}_k\dot{\theta}_j\right) + \frac{\partial U}{\partial \theta_i}(\theta) = \Upsilon_i \qquad i = 1, ..., n \tag{1.57}$$

Rearranging terms, it is possible to write

$$\sum_{j=1}^{n} M_{ij}(\theta)\ddot{\theta}_j + \sum_{j,k=1}^{n}\Gamma_{ijk}\dot{\theta}_j\dot{\theta}_k + \frac{\partial U}{\partial \theta_i}(\theta) = \tau_i \qquad i = 1, ..., n \tag{1.58}$$

where Γ_{ijk} is given by

$$\Gamma_{ijk} = \frac{1}{2}\left(\frac{\partial M_{ij}(\theta)}{\partial \theta_k} + \frac{\partial M_{ik}(\theta)}{\partial \theta_j} - \frac{\partial M_{kj}(\theta)}{\partial \theta_i}\right) \tag{1.59}$$

Equation (1.58) is a second-order differential equation in terms of the manipulator joint variables. It consists of four pieces: inertial forces, which depend on the acceleration of the joints; centrifugal and Coriolis forces, which are quadratic in the joint velocities; potential forces, of the form $\frac{\partial U}{\partial \theta_i}$; and external forces τ_i.

The functions Γ_{ijk} are called the Christoffel symbols corresponding to the inertia matrix $M(\theta)$.

In order to put the equations of motion back into vector form, the matrix $C(\theta, \dot{\theta}) \in$

$\mathbb{R}^{n\times n}$ is defined as

$$C_{ij}(\theta,\dot{\theta}) = \sum_{k=1}^{n} \Gamma_{ijk}\dot{\theta}_k = \frac{1}{2}\sum_{k=1}^{n}\left(\frac{\partial M_{ij}}{\partial \theta_k} + \frac{\partial M_{ik}}{\partial \theta_j} - \frac{\partial M_{kj}}{\partial \theta_i}\dot{\theta_k}\right) \tag{1.60}$$

The matrix C is called the Coriolis matrix for the manipulator; the vector $C(\theta,\dot{\theta})\dot{\theta}$ gives the Coriolis and centrifugal force terms in the equations of motion.
Then, as described by the Equation (1.52), (1.58) can now be written as

$$M(\theta)\ddot{\theta} + C(\theta,\dot{\theta})\dot{\theta} + N(\theta,\dot{\theta}) = \tau \tag{1.61}$$

1.7 Robot manipulators: an overview

Due to the exigences and rapidity of the production systems, a variety of technological advances has started in the industry. Robotics is occupying an important place in the modernization of industry and this is directly related with a variety of fields, like electrical engineering, mechanical engineering, electronics, computer sciences and mathematics.

The term robot is usually used to refer to automatic machines. These machines can be categorized as follows:

- Robot manipulators
- Mobile robots

Mobile robots also comprise terrestrial robots, aquatic robots and aerial robots. And as it can be seen, robot manipulators and mobile robots are considered key elements in the field of robotics. Both are concerned in this thesis.

The robot manipulators are commonly defined as manipulator machines with several degrees of freedom automatically controlled and programmable. They can be used in many activities, being able to stay in a fixed or mobile place to accomplish industrial applications. For the industry, the robot manipulators normally execute repetitive and accurate tasks, as well as dangerous tasks for human operators. Some of the main advantages in the usage of robot manipulators are reducing production costs and the increase of production and precision. Additionally, some applications are totally absorbed by the robot manipulators, like the manipulation in radioactive or explosive zones and also in aquatic and space applications.

Short term projections show that the main activity for robot manipulators is assembly. Moreover, according to the International Federation on Robotics, for the year 2000 there were more than 749 000 operational units around the world and the annual robotic population growth is about 20 000.

Consequently, a lot of works have been addressed on different aspects of the field of robot manipulators, like kinematics, dynamics and the design of control laws. The control of robot manipulators has been an active area of research, focusing mainly on the design of nonlinear controllers. However, since the amount of applications for this kind of systems has increased, the control laws have been studied from different points of view, like position control and trajectory tracking, as in Dixon et al. [1999], Parra-Vega et al. [2003] and Slotine and Li [1987], torque and force control, like in Khatib [1987] and Raibert and Craig [1981].

There are two primary types of motion that a robot manipulator links can produce: *revolute* or *prismatic* movements. Revolute joints act like the human arm, and are known as anthropomorphic joints. Prismatic joints are able to stretch and retract the link, like a car radio antenna. The majority of robots are classified with respect to the configuration and type of the three first joints or axes, where it is common to find the Articulated configuration, Type I Scara, Type II Scara and Cartesian. There are also two extra configurations, but nowadays they are much less common used, these are the Cylindrical and Spherical configurations. All these configurations are briefly presented in the next sections.

1.7.1 Articulated arms

The variety of commercial robots with this configuration is very large. All of these robots have revolute joints. Normally, the second and third axes are co-planar and work together to produce a motion in a vertical plane. The first axis is in the base, and it turns in order to sweep out the work area volume. The limitation performance of this arm configuration is that the second joint has to lift the subsequent arm link and the payload, causing the loss of accuracy. In fact, articulated arms have less accuracy compared to other robot arm configurations, since all joint angles have position errors and are accumulated along the arm, for any serial manipulator. An example is presented in Fig. 1.6

1.7.2 Type I SCARA

The SCARA acronym stands for **Selective Compliance Assembly Robot Arm**, and it is shown in Fig. 1.7. The arm is rigid in the Z axis and pliable in the X-Y axes, which allows it to adapt to holes in the X-Y axes. In general, the arm structure allows to carry weight, but the first and second axes do not lift. The third axis provides work area volume by the addition of a vertical or Z axis. Then, the fourth axis adds rotation about the Z axis to control orientation in the horizontal plane, Lewis et al. [2004]. This is advantageous to transfer parts from one cell to another one.

Figure 1.6 – Articulated robot arm configuration, courtesy of (KUKA [2016]).

Figure 1.7 – Type I SCARA configuration, courtesy of (ABB [2016]).

1.7.3 Type II SCARA

It has almost the same configuration as the Type I SCARA. The difference is that the first axis is a long vertical prismatic Z link, which lifts the two parallel revolute joints and their links. In general, this configuration is used to move heavy objects (over approximately $30kg$) over long distances at high speed. In this sense, the Type II SCARA configuration is more efficient than the Type I. Fig. 1.8 shows an example of Type II SCARA.

Figure 1.8 – Type II SCARA configuration, courtesy of (Toshiba [2016]).

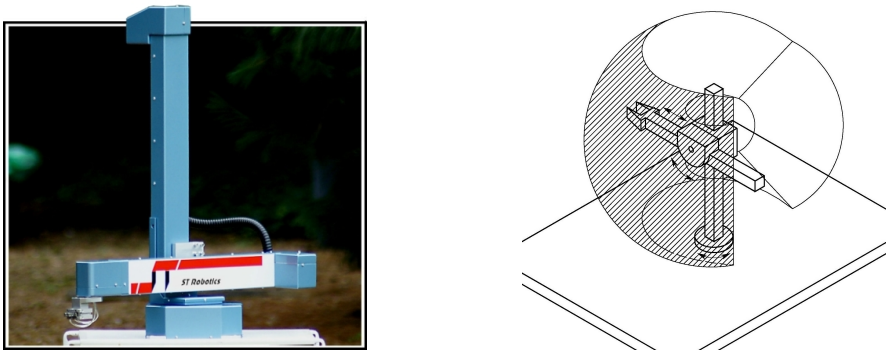
1.7.4 Cartesian coordinate robots

A cartesian coordinate robot is usually an industrial robot whose three principal axis are linear (they have prismatic joints). These three sliding links correspond to moving the wrist up-down, in-out and back-forth. One, two or three revolute joints can be added to allow control orientation. Popular applications for this kind of robots are the computer numerical control machine (CNC machine) and 3D printing. Pick and place machines and plotters are also based on the principal of the cartesian coordinate robot. Fig. 1.9 shows an example of Cartesian Coordinate Robot and its application on a 3D printing machine.

Figure 1.9 – 3D printing machine, courtesy of (PRUSA [2016]).

1.7.5 Spherical and cylindrical coordinate robots

The spherical robot is a robot with two revolute joints and one prismatic joint. This configuration allows it to form a spherical coordinate system with a spherical work volume. On the other hand, cylindrical coordinate robots are robots whose axes form a cylindrical coordinate system, the first axis is a revolute base rotation and the second and third axes are prismatic. Commercial robots with these configurations were mainly used in machine tending and material handling applications. Some of them are still in use, but the decline in use of these two configurations was attributed to problems arising from the use of the prismatic link for radial extension/retraction motion, Lewis et al. [2004]. Fig. 1.10 shows cylindrical and spherical configuration robots.



(a) (b)

Figure 1.10 – (a) Cylindrical arm, courtesy of (ST-Robotics [2016]), (b) Spherical robot configuration.

1.7.6 Parallel link manipulators

A parallel-link manipulator is a mechanical system that uses several computer-controlled serial chains to support a single platform. With a correct design, a six-link parallel-link manipulator can have until six degrees of freedom for the working platform, see Fig. 1.11. Compared to serial-link robots, parallel-link configuration offers greater stiffness and precision. Some industrial applications of these devices are flight simulators, automobile simulators, in manufacturing processes and optical fiber alignment.

Figure 1.11 – Parallel-link robot, courtesy of (Codian-Robotics [2016]).

1.7.7 Robotic systems

A robot manipulator can not be considered just as a series of mechanical linkage. In fact, the mechanical system is only a component of a Robotic System, which consists on the arm manipulator, power source, end-effector, a set of internal and external sensors, a computer interface (also programmed software) and the control computer.

All these tools allow the robot to accomplish a task or a sequence of tasks, where accuracy and repeatability are key points to carry out the final objective. The accuracy

of a manipulator can be considered as the measurement of how close the manipulator can come to a given point within its workspace. Repeatability is a measure of how close the manipulator can return to a previously taught point, Spong et al. [2004]. The most used devices for sensing position error are the position encoders, located at the joints, on the shaft of the motor that actuates the joint or the joint itself. For the position and orientation measurement of the end-effector, one relies on the knowledge of the geometry and rigidity of the system to estimate through some geometrical tools and the measured joint angles, the position of the end-effector. However, accuracy can be affected by some factors, like computational errors, errors during the construction of the manipulator, flexibility effects (if the manipulator materials are not rigid enough) such as the bending of the links due to gravity or other loads and due to static and dynamical effects. Then, without proper rigidity, the accuracy of the arm manipulator can only be improved by direct sensing of the end-effector.

On the other hand, repeatability can be affected by the controller resolution. Controller resolution means the smallest increment of motion that the controller can sense, where the resolution is considered as the total distance traveled by the tip and divided by 2^n, where n is the number of bits of encoder accuracy, Spong et al. [2004]. With this, it is possible to conclude that robots with prismatic joints have better resolution, since the distance traveled by the tip between two points is smaller than the arc angle traced by the tip of a revolute joint.

Even so, manipulator arms with revolute joints have some advantages in design compared to prismatic configurations, like the increased dexterity and compactness of the revolute joint designs, which propitiates a smaller working volume and as consequence, a robot can be working with other robots and people. Also, revolute joint manipulators have the ability to move around obstacles and have a wider range of applications.

1.7.8 Dynamics of a three-link manipulator arm

The mechanism shown in Fig. 1.12, consists of two intersecting axes at the shoulder, an elbow, and a spherical wrist (modeled as three intersecting axes). The reference configuration ($\theta = 0$) is fully extended.

The forward kinematics is computed by calculating the individual twist motions for each joint, as in section 1.3.1. The transformation between the tool and base frames at $\theta = 0$ is given by

$$j_{st}(0) = \begin{bmatrix} & & \begin{pmatrix} 0 \\ l_1 + l_2 \\ 1 \end{pmatrix} \\ I & & \\ 0 & & 1 \end{bmatrix} \tag{1.62}$$

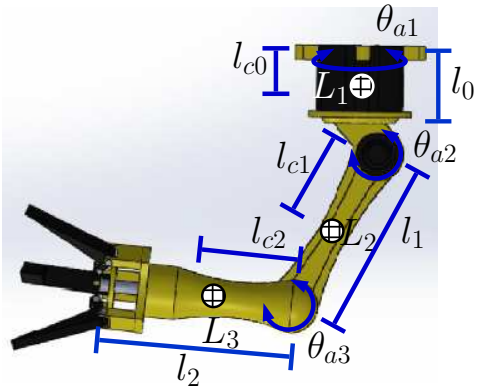


Figure 1.12 – Three-link open-chain manipulator, θ_{ai} represent the joint angles.

Now, the joint twists are given by

$$\xi_1 = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} \qquad \xi_2 = \begin{bmatrix} 0 \\ -l_0 \\ 0 \\ -1 \\ 0 \\ 0 \end{bmatrix} \qquad \xi_3 = \begin{bmatrix} 0 \\ -l_0 \\ l_1 \\ -1 \\ 0 \\ 0 \end{bmatrix} \tag{1.63}$$

A frame L_i is attached to each link at the center of mass and aligned with the principal inertia axe of the link

$$j_{sl_1(0)} = \begin{bmatrix} I & \begin{pmatrix} 0 \\ 0 \\ l_{c0} \end{pmatrix} \\ 0 & 1 \end{bmatrix} \qquad j_{sl_2(0)} = \begin{bmatrix} I & \begin{pmatrix} 0 \\ l_{c1} \\ l_0 \end{pmatrix} \\ 0 & 1 \end{bmatrix} \qquad j_{sl_3(0)} = \begin{bmatrix} I & \begin{pmatrix} 0 \\ l_1 + l_{c2} \\ l_0 \end{pmatrix} \\ 0 & 1 \end{bmatrix} \tag{1.64}$$

With this choice of links frames, the inertia matrices have the general form

$$\mathcal{M}_i = \begin{bmatrix} m_i & 0 & 0 & & & \\ 0 & m_i & 0 & & 0 & \\ 0 & 0 & m_i & & & \\ & & & I_{xi} & 0 & 0 \\ & 0 & & 0 & I_{yi} & 0 \\ & & & 0 & 0 & I_{zi} \end{bmatrix} \tag{1.65}$$

where $\mathcal{M}_i$ is the generalized inertia matrix for the ith link, m_i is the mass of the object and I_{xi}, I_{yi} and I_{zi} are the moments of inertia about x, y and z-axes of the ith frame. To

compute the manipulator inertia matrix, the body Jacobians corresponding to each link frame are computed, the calculation yields

$$\begin{aligned} J_1 = J^b_{sl_1(0)} &= \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix} \qquad J_2 = J^b_{sl_2(0)} = \begin{bmatrix} -l_{c1}c_2 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & -l_{c1} & 0 \\ 0 & -1 & 0 \\ -s_2 & 0 & 0 \\ c_2 & 0 & 0 \end{bmatrix} \\ J_3 = J^b_{sl_3(0)} &= \begin{bmatrix} -l_2c_2 - l_{c2} - c_{23} & 0 & 0 \\ 0 & l_1s_3 & 0 \\ 0 & -l_{c2} - l_1c_3 & -l_{c2} \\ 0 & -1 & -1 \\ -s_{23} & 0 & 0 \\ c_{23} & 0 & 0 \end{bmatrix} \end{aligned} \tag{1.66}$$

The inertia matrix for the system is given by

$$M(\theta) = \begin{bmatrix} M_{11} & M_{12} & M_{13} \\ M_{21} & M_{22} & M_{23} \\ M_{31} & M_{32} & M_{33} \end{bmatrix} = J_1^T \mathcal{M}_1 J_1 + J_2^T \mathcal{M}_2 J_2 + J_3^T \mathcal{M}_3 J_3 \tag{1.67}$$

where the components of M are given by

$$\begin{aligned} M_{11} &= I_{y2}s_2^2 + I_{y3}s_{23}^2 + I_{z1} + I_{z2}c_2^2 + I_{z3}c_{23}^2 + m_2 l_{c1}^2 c_2^2 + m_3(l_1c_2 + l_{c2}c_{23})^2 \\ M_{22} &= I_{x2} + I_{x3} + m_3 l_1^2 + m_2 l_{c1}^2 + m_3 l_{c2}^2 + 2m_3 l_1 l_{c2} c_3 \\ M_{23} &= I_{x3} + m_3 l_{c2}^2 + m_3 l_1 l_{c2} c_3 \\ M_{32} &= I_{x3} + m_3 l_{c2}^2 + m_3 l_1 l_{c2} c_3 \\ M_{33} &= I_{x3} + m_3 l_{c2}^2 \\ M_{12} &= M_{13} = M_{21} = M_{31} = 0 \end{aligned} \tag{1.68}$$

Note that several of the moments of inertia of the different links do not appear in this expression. This is because the limited degrees of freedom of the manipulator do not allow arbitrary rotations of each joint around each axis. The Coriolis and centrifugal forces are computed directly from the inertia matrix through the Christoffel symbols, and it yields

$$C_{ij}(\theta, \dot{\theta}) = \sum_{k=1}^{n} \Gamma_{ijk} \dot{\theta}_k = \frac{1}{2} \sum_{k=1}^{n} \left(\frac{\partial M_{ij}}{\partial \theta_k} + \frac{\partial M_{ik}}{\partial \theta_j} - \frac{\partial M_{kj}}{\partial \theta_i} \right) \tag{1.69}$$

A calculation shows that the nonzero values of Γ_{ijk} are given by

$$\begin{aligned}
\Gamma_{112} &= \Gamma_{121} = (I_{y2} - I_{z2} - m_2 l_{c1}^2)c_2 s_2 + (I_{y3} - I_{z3})c_{23}s_{23} - m_3(l_1 c_2 + l_{c2}c_{23})(l_1 s_2 + l_{c2}s_{23}) \\
\Gamma_{113} &= \Gamma_{131} = (I_{y3} - I_{z3})c_{23}s_{23} - m_3 l_{c2}s_{23}(l_1 c_2 + l_{c2}c_{23}) \\
\Gamma_{211} &= (I_{z2} - I_{y2} - m_2 l_{c1}^2)c_2 s_2 + (I_{y3} - I_{z3})c_{23}s_{23} + m_3(l_1 c_2 + l_{c2}c_{23})(l_1 s_2 + l_{c2}s_{23}) \\
\Gamma_{223} &= \Gamma_{232} = \Gamma_{233} = -l_1 m_3 l_{c2} s_3 \\
\Gamma_{311} &= (I_{z3} - I_{y3})c_{23}s_{23} + m_3 l_{c2}s_{23}(l_1 c_2 + l_{c2}c_{23}) \\
\Gamma_{322} &= l_1 m_3 l_{c2} s_3
\end{aligned} \tag{1.70}$$

Finally, the gravitational forces on the manipulator are computed. These forces are written as

$$N(\theta, \dot{\theta}) = \frac{\partial U}{\partial \theta} \tag{1.71}$$

where $U : \mathbb{R}^n \to \mathbb{R}$ is the potential energy of the manipulator. For the three link-manipulator under consideration here, the potential energy is given by

$$U(\theta) = m_1 g h_1(\theta) + m_2 g h_2(\theta) + m_3 g h_3(\theta) \tag{1.72}$$

where h_i is the height of the center of mass for the ith link. These can be found using the forward kinematic map.

$$j_{sl_i(\theta)} = e^{\widehat{\xi}_{1\theta_1}} ... e^{\widehat{\xi}_{i\theta_i}}(0) \tag{1.73}$$

which gives

$$\begin{aligned}
h_1(\theta) &= l_{c0} \\
h_2(\theta) &= l_0 - l_{c1}\cos\theta_2 \\
h_3(\theta) &= l_0 - l_1\cos\theta_2 - l_{c2}\cos(\theta_2 + \theta_3)
\end{aligned} \tag{1.74}$$

The computational of the gravitational forces is given by

$$N(\theta, \dot{\theta}) = \frac{\partial U}{\partial \theta} = \begin{bmatrix} 0 \\ (m_2 l_{c1} + m_3 l_1)g\sin\theta_2 + m_3 g l_{c2}\sin(\theta_2 + \theta_3) \\ m_3 g l_{c2}\sin(\theta_2 + \theta_3) \end{bmatrix} \tag{1.75}$$

This completes the derivation of the dynamics and, finally, the equation of motion (1.52) can also be obtained.

Chapter 2
State of the art of aerial manipulation

This chapter is devoted to the study of the state of the art of the topic addressed in this thesis. This study is divided in two parts.

For the first part, the field of Unmanned Aerial Vehicles is contemplated. Here, a brief introduction of these aerial systems is shown, introducing the different configurations but, highlighting the importance of the VTOL vehicles and their importance for some kind of applications. Then, analyzing the VTOLs in a deeper way, a brief history of these vehicles is shown, together with the classification of the most common configurations. From here, we focus our attention on the multirotors, where we perform a concise state of the art, since this topic has been widely discussed.

Finally, we introduce the problem of UAV's carrying payloads or robot manipulators. Since the movement of the payload or the arm manipulator exerts unknown moments and torques on the aerial vehicle, there will be a loss of stability and the tasks activities will be barely reached. To solve this problem, many approaches have been presented, and the corresponding studies are addressed and discussed.

2.1 Unmanned Aerial Vehicles (UAV)

An Unmanned Aerial Vehicle (UAV) is an autonomous flying machine, this means that the vehicle can perform flights without a human pilot by the usage of an autonomous pilot algorithm. While the concept of UAV seems a novelty, its usage is not. In fact, the interest in the study of these type of vehicles has increased, and there are many works published especially related to control laws applied on the topic. The research on this subject is also possible due to the evolution of information and communication technologies and the power computation of the different devices.

It should be noted that the ballistic or semi-ballistic missiles, the cruise missiles and the artillery projectiles are not considered UAV systems.

An UAV system is composed mainly by two elements:

- Flight component: formed by the aerial vehicle.

- Ground component: formed by the ground station, which allows the communication between this and the aerial system, data processing, monitoring, control, etc.

In general the drones can be classified according with the size, the application or a combination of both, however, the size is the most used criterion to make the classification. The classification, depicted in Fig. 2.1 is resumed in the next items, Guerrero-Castellanos [2008]:

- Micro-UAV: This kind of vehicles can be operated by just one person. The propulsion is electrically made and due to the low cost of the materials, the vehicles are primarily used for civilian applications. The average autonomy is about 30 minutes.

- Mini-UAV: The equipment necessary for these vehicles is more specialized. In general, these drones fly at speeds of $70km/h$ and medium altitudes of $3,5km$. Their size allows them to fly around 4 hours.

- MALE/HALE: The Medium Altitude Long Endurance (MALE) drones are used to perform higher endurance flights and also at higher altitudes ($10 - 15km$). The High Altitude Long Endurance (HALE) drones can fly at about $20km$ of altitude. These drones are able to perform missions with medium endurance of two days. The Predator (MALE) and the Global Hawk (HALE) are the most popular drones of this category and are mainly applied for military missions.

- UCAV: The Unmanned Combat Air Vehicle is mainly used for offensive missions. They are under research and only some prototypes are available, like the RQ-170 Centinel.

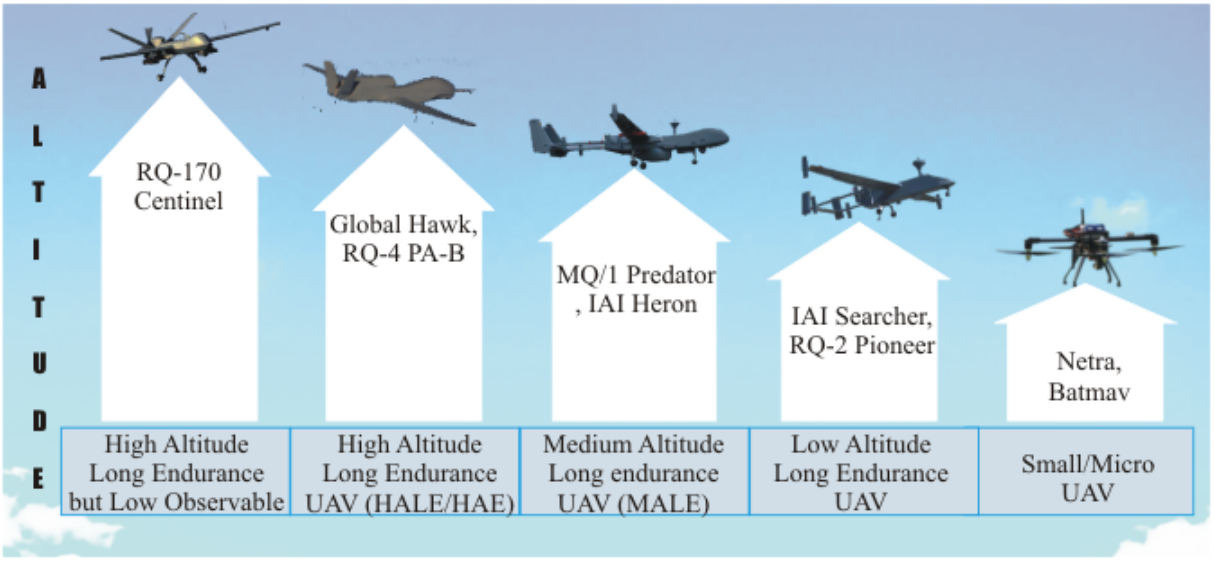


Figure 2.1 – The different categories of drones.

Due to their aerodynamical function, there is an alternative classification, where it is possible to find three families: fixed wings aircrafts, rotating wings aircrafts and the swing-wing aircrafts. A study shows that the research has mainly focused on the rotating wings aircrafts with VTOL (Vertical Take Off and Landing) configuration. These vehicles are capable of take off and land in a vertical way and due to this advantage, the range of applications is very extensive, being used on military and civilian applications, like surveillance, rescue, aerial mapping, etc.

2.2 Vertical Take Off and Landing (VTOL) vehicles

A Vertical Take Off and Landing (VTOL) vehicle is a special aerial system, designed and utilized for highly specialized missions because of their advantages comparing to traditional UAVs. VTOLs can take off and land vertically, having the ability to land in difficult and restricted size areas. Besides, this kind of systems can hover over specific areas of interest.

The idea of vertical flight has been present for about a century. The first four rotor helicopter was designed and built in 1922 by George de Bothezat for military applications. Even when the aircraft had a successful flight, the performance did not accomplish the expectations and the project was forgotten. Then, the research on manned VTOL aircrafts continued, but it was until the years 1950-1970 that there were some advances on the topic.

In general, the classification of the VTOL vehicles includes two kinds of systems: the **rotorcraft** systems and the **powered lift** systems, where many subsystems can be found.

2.2.1 Rotorcraft

- Helicopter: is a type of rotorcraft in which lift and thrust are supplied by rotors. This allows the helicopter to take off and land vertically, to hover, and to fly forward and laterally. These attributes allow helicopters to be used in congested or isolated areas where fixed-wing aircrafts would usually not be able to take off or land. Fig. 2.2 shows an example of a helicopter.

Figure 2.2 – An Apache attack helicopter

- Multirotor: also called multicopter, is a rotorcraft with more than two rotors. An advantage of multirotor aircraft is the simpler rotor mechanics required for flight control. Unlike single and double-rotor helicopters, multirotors often use fixed-pitch and yaw blades; control of vehicle motion is achieved by varying the relative speed of each rotor to change the thrust and torque produced by each one. Due to their ease of both construction and control, multirotor aircraft are frequently used in radio control aircraft and UAV projects, in which the names **tricopter**, **quadcopter** see Fig. 2.3a, **hexacopter** and **octocopter** see Fig. 2.3b, are frequently used to refer to 3, 4, 6 and 8-rotor helicopters, respectively.

(a)

(b)

Figure 2.3 – (a)Quadcopter and (b)octocopter configurations, courtesy of (DJI [2017])

- Autogyro: is also known as gyroplane, gyrocopter or rotaplane. Is a type of rotorcraft that uses an unpowered rotor in autorotation to develop lift, and an engine-powered propeller, similar to that of a fixed-wing aircraft, to provide thrust. While similar to a helicopter rotor in appearance, the autogyro's rotor must have air flowing through the rotor disc to generate rotation. Fig. 2.4 shows an example of an autogyro.

Figure 2.4 – Autogyro MT-03 in flight

- Gyrodyne: is a type of VTOL aircraft with a helicopter rotor-like system that is driven by its engine for take off and landing and also includes one or more conventional propellers to provide forward thrust during cruising flight. Lift during forward flight is provided by a combination of the rotor, like the autogyro, as well as conventional wings. Fig. 2.5 shows an example of gyrodyne.

Figure 2.5 – Gyrodyne AM-X3 in flight

2.2.2 Powered lift

- Convertiplane: is an aircraft which uses rotor power for vertical take off and landing and converts to fixed-wing lift in normal flight. These vehicles may be divided into two broad classes, based on wether the rotor is fixed as in a helicopter or tilts to provide thrust in forward flight. Fig. 2.6 shows an example of a convertiplane.

Figure 2.6 – Convertiplane

- Tail-sitter: called also tailsitter, is a type of VTOL vehicle that takes off and lands on its tail, then tilts horizontally for forward flight. Fig. 2.7 shows an example of tail-sitter UAV's.

Figure 2.7 – Tail-sitter prototypes, as part of the Google's project Wing for delivery

- Lift jets: is an auxiliary jet engine used to provide lift for VTOL operation, but may be shut down for normal wing-borne flight.

- Lift fans: is an aircraft configuration in which lifting fans are located in large holes in an otherwise conventional fixed wing or fuselage. It is used for V/STOL [1] operation. The aircraft takes off using the fans to provide lift, then transitions to fixed.wing lift in forward flight. Several experimental craft have been flown, but only the F-35 Lighting II entered into production, see Fig. 2.8

The present work is centered on the analysis and study of multirotors, particularly on the modelling and control of a quadcopter carrying a manipulator arm. However, during the development of the project, the hexacopter configuration was also used. It allowed the obtention of some interesting results.

1. Vertical and/or Short Take-Off and Landing

Figure 2.8 – F-35 Lighting II combat aircraft during take off

2.3 Multi-rotors: state of the art

The most common vehicle with the capacity of take off and landing in vertical way is the standard helicopter, which is composed of a principal rotor and a rear rotor. However, the multirotors, where we can find the four rotor helicopter, quadrotor or quadcopter, the hexacopter or hexarotor and the octocopter or octorotor have been the center of interest for many works in the last years, see Guerrero-Castellanos et al. [2011], Alaimo et al. [2013] and Fogelberg [2013]. Some of the advantages offered by these vehicles are its symmetry, which makes it easy to design and build. The usage of four, six or eight rotors provides improved stability on hover, because the distributed pushing forces are acting at the same distance from the center of mass instead of just one pushing force acting on the whole center of mass. Also, the propellers can be protected by the frame of the prototype, which makes its usage indoor safer, face to the non-protected propellers of the standard helicopter.

The modeling of this kind of vehicles has done possible the design of many control laws which allow the attitude and position stabilization with good results. Some of the most used control laws have been: backstepping, found on Bouabdallah and Sierwart [2005], Madani and Benallegue [2006] and Yali et al. [2010]; sliding mode control has also been applied to these vehicles, and it can be found on Bouabdallah and Sierwart [2005], Zheng et al. [2014] and Arellano-Muro et al. [2013]; linear control laws (PD and PID), they can be seen at Bora and Erdinc [2007], Jun and Yuntang [2011] and Hoffmann et al. [2007].

Other interesting projects developed with multi-rotors can be found on the domains of: navigation, where embedded cameras and/or different sensors are used to know the relative position of the aerial system, see Courbon et al. [2009] and Sebesta and Baillieul [2012]; fault-tolerant control, where fault detection on the actuators is implemented and a control law is proposed for the aerial vehicle, see Sharifi et al. [2010] and Li et al. [2013].

2.4 Some examples of Unmanned Aerial Vehicles carrying manipulators or payloads

Aerial manipulation has been an active area of research in recent times, mainly because the active tasking of Unmanned Aerial Vehicles (UAVs) increases the employability of these vehicles for various applications. For active tasking one would consider manipulation, grasping and transporting, etc.

Unlike fixed wings UAVs that are incapable of driving their velocity to zero, VTOL (Vertical Take-Off and Landing) vehicles such as helicopters with four rotors are ideally suited to the task of aerial manipulation or grasping. However, there are many challenges in aerial grasping for quadrotors. The biggest challenge arises from their limited payload. While multiple robots can carry payloads with grippers (Mellinger et al. [2010]) or with cables (Michael et al. [2011] and Palunko et al. [2012]), see Fig. 2.9, their end effectors and grippers have to be light weight themselves and capable of grasping complex shapes. Secondly, the dynamics of the robot are significantly altered by the addition of payloads. Indeed this is also an attraction in assembly because aerial robots can use this to sense disturbance forces and moments, like in (Mohammadi et al. [2016]), where the tele-operation of a team of VTOL vehicles for cooperative aerial manipulation uses haptic feedback to transport an object. Besides, for payload transport, it is necessary that the robots are able to estimate the inertia of the payload and adapt to it to improve precise positioning and tracking performance.

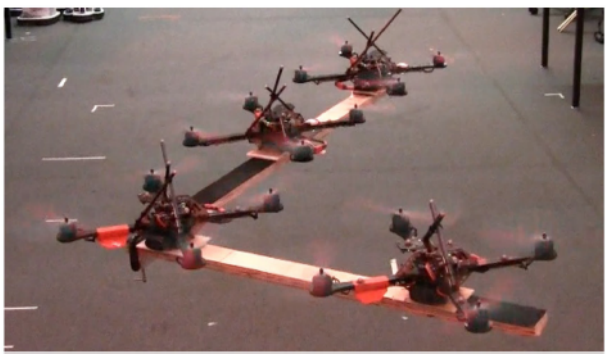


Figure 2.9 – Cooperative transportation with quadrotors.

Numerous approaches have been proposed to deal with such a problem. In Orsag et al. [2013b] and Khalifa et al. [2012], a Newton-Euler approach is used to model and control a manipulator based quadrotor. However, for the first one the control over the manipulator is partial, since the operator controls the extension of the manipulator by visual contact. In Orsag et al. [2013a], a Lyapunov based model Reference Adaptive Control is used to stabilize a quadrotor with multi degree of freedom (DOF) manipulator, see Fig. 2.10.

However, the stability analysis is carried out with a linear approach and only rigid body dynamics of the quadrotor were considered due to the complexity of the system.

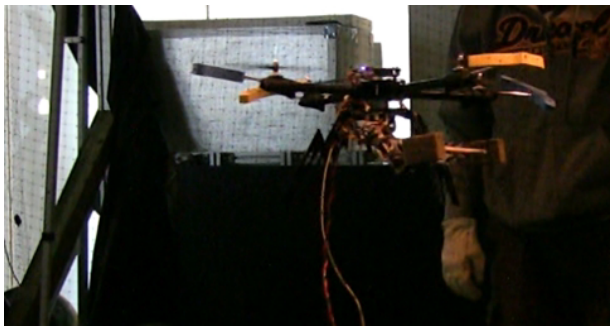


Figure 2.10 – MM-UAV carrying a long rod.

In Ghadiok et al. [2011], a gripping of an object is presented. For this, indoor experiments are performed with a quadrotor equipped with a gripper, where a monocular camera is used to locate and grip an object without any prior information of the environment by the usage of a model-based controller. In Ghadiok et al. [2012], the experiments are extended to outdoor, using a GPS system and a Kalman filter to improve the navigation to the position of the object. Pounds et al. [2011] reports the grasping of an object with a helicopter and a gripper, by means of a PID controller. The work presents a stability analysis, where some stability bounds are determined in which the changing mass-inertia parameters of the system due to the grasped object does not destabilize this flight controller. But the last contributions are limited to the use of a 1-DOF gripper, which reduces the precision of manipulation and the scenarios where there are obstacles to reach a target, see Fig. 2.11.

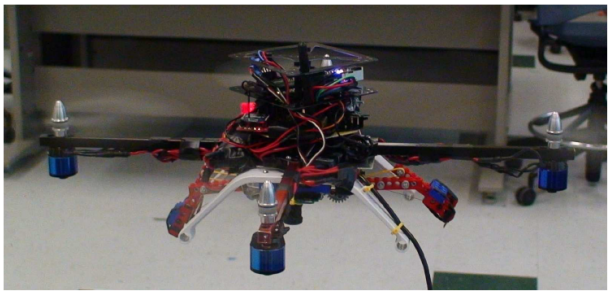


Figure 2.11 – Quadrotor in flight to perform aerial gripping.

In Lipiello and Ruggiero [2012a,b], a dynamic model of an UAV with an attached robotic arm was derived by the Euler-Lagrange formalism. With this, a Cartesian impedance control, which provides a relationship between external forces and the system is designed. Jimenez-Cano et al. [2013] equally deals with aerial manipulators consisting on a UAV with a robotic multi-link arm. The work presents a Newton-Euler approach to model and

control a quadrotor under the perturbations caused by the motion of an arm, through a Variable Parameter Integral Backstepping (VPIB), outperforming the results obtained by using PID controllers. However, the parametrization of the system is made through Euler angles, which present attitude estimation singularities. Kim et al. [2013] presents aerial manipulation using a quadrotor and a 2-DOF robot arm. By considering the quadrotor and the arm manipulator as a combined system the kinematic and dynamics models are obtained by the Euler-Lagrange formulation. After that, an adaptive sliding mode controller is designed, the effectiveness of the proposed method is experimentally showed by picking up and delivering and object.

Heredia et al. [2014] presents the development of a large payload multirotor-based aerial manipulator for outdoor operation. The full dynamic model of the system is derived through the Euler-Lagrange formalism. Then, a control law for the multirotor was proposed to dampen the disturbances caused by the movement of the manipulator. Moreover, a control law for the 7-DoF manipulator was proposed. The validation of the proposed solution is made through some experiments, where the multirotor performs a hover while the manipulator is moving. Then, in Kondak et al. [2014], a system for aerial manipulation composed of a helicopter and an industrial manipulator is presented. An analysis is performed when the controllers for the helicopter and the manipulator are independent, leading to unstable helicopter behaviour. To face this, a scheme control for kinematical and dynamical coupling between the two controllers is proposed. The validation of the proposed scheme is performed through some experiments, which show that at least simple manipulation tasks can be achieved.

Yüksel et al. [2016a] present the dynamic modeling and control design of a Planar Vertical Take Off and Landing Vehicle (PVTOL) equipped with either rigid or elastic joints manipulator arm. The system is linearized with a dynamic feedback and is proven to be differentially flat. The performance of the two cases of manipulator arms are compared with simulations and preliminary experimental results are presented using an actuated joint arm on a quadrotor. Then, in Yüksel et al. [2016a], a new class of aerial manipulator is presented, the protocentric. It is formed by a PVTOL equipped with any number of different parallel manipulator arms, all of them attached to the Center Of Mass (COM) of the aerial vehicle. The whole system is proven to be differentially flat, regardless the mixture between rigid and elastic joints. A control law has been proposed for the case of rigid joints only and validated through simulations. Finally, in Tognon et al. [2017] a control methodology for aerial manipulators able to achieve highly dynamic behaviours is presented. The method comprises two steps: a nominal input-state trajectory generator and a decentralized feedback controller acting on the degrees of freedom providing robustness to the closed-loop system. Experimental results using a quadrotor and a 2-DOF rigid ma-

nipulator demonstrate that the proposed controller is able to track dynamic trajectories and also the benefit of the usage of differential flatness of aerial manipulators.

As we can see from TABLE 2.1, numerous approaches have tried to solve the problem of attitude and position stabilization of an aerial system carrying a manipulator arm. Then, it is possible to propose a classification based on the formalisms and proposed strategies in order to accomplish such an objective.

Formalism/ Approach	Author	Limitations
Newton-Euler/ Euler-Lagrange formalisms	Lipiello and Ruggiero [2012a] Lipiello and Ruggiero [2012b] Khalifa et al. [2012] Orsag et al. [2013b] Jimenez-Cano et al. [2013] Heredia et al. [2014] Kim et al. [2013]	There is a singularity that occurs in the rotation matrix when the second Euler rotation is 90° (or 270°), resulting in an infinite number of solutions to the Euler sequence.
Linearized approach	Orsag et al. [2013a] Yüksel et al. [2016b]	Since the nonlinear system is linearized to work in a limited range of operation, the operation of the control techniques is also limited to that operating region.
Planar-vertical approach	Yüksel et al. [2016a] Yüksel et al. [2016b]	These approaches do not take into account the full 3d space, then, the proposed solutions are limited to the planar case.

Table 2.1 – State of the art of aerial systems carrying manipulators.

Then, the main contribution of the present work is centered on the modeling and the asymptotical stabilization of a mini-quadrotor carrying a rigid manipulator arm, see Fig. 2.12. For this, a mathematical model which takes into account the coupling between the two systems and makes use of the quaternion parametrization is presented. Besides, the proposed model allows, through a simple algorithm, the computation of the moments and torques exerted by the movement of the manipulator arm.

Then, an attitude nonlinear control law based on bounded functions is designed. This takes into account the torques produced by the motion of the manipulator arm and ensures the attitude stabilization of the system. Contrary to the above mentioned research, the design of the attitude control law uses also the quaternion parametrization, which avoids the presence of singularities. Finally, the quadcopter is driven to a desired position by a nonlinear control, also based on nested saturation functions.

The proof of stability and experimental results validate the proposed method control strategy and allow a comparison of the results when the motion of the arm is taken into

account or not.

Figure 2.12 – Mini-quadcopter with its manipulator arm.

Chapter 3
Modeling of a quadcopter carrying a manipulator arm

This chapter is devoted to the presentation of the model of a quadcopter carrying a manipulator arm.

In order to obtain the full model of the combined systems, quadrotor and manipulator arm, a general description of the aerial system and its model are addressed. In general, it is possible to extend the proposed model to any multirotor, since the dynamics of the system are considered for a rigid body in the space and the difference relies on the distribution of the forces coming from the actuators velocity. Initially the perturbations, like wind, are neglected

Basically, the model of the quadcopter takes into account the torques coming from the movement of the arm manipulator. The model was driven in two ways: a static model and a dynamic model. For the static model, the total center of mass of the arm manipulator is computed in every moment, and used together with a formulation taken from a previous work to compute the torque that the arm manipulator is producing; this estimation is directly used by the attitude control law, presented in the next chapter. For the dynamic model, a parameter identification study and the first order model formulation for the real robot manipulator actuators were performed in order to know the behavior of this one; after that, using the equations of the manipulator, it is possible to estimate the different torques and send them to the attitude control law.

3.1 Quadrotor operation

This part of the chapter is devoted to the mathematical modeling of the quadrotor. First, a general description of the function of the system is presented. After that, the mathematical modeling is treated, and equally the relation between motors, propellers and the dynamics

of the system.

The quadrotor or four rotor helicopter is a mechatronic system composed of a cross structure. At each end of the cross we find a propeller coupled to a motor, and at the center of the configuration all the electronic elements are found (power source, computer, etc). Compared to the classical helicopter, this system does not have main rotor and the control is performed by the angular velocity change on each rotor, Nelson [1998].

The four rotors are composed of the propellers coupled to DC motors or DC Brushless motors (BLDC). Such a platform is represented in Fig. 3.1, where the front and rear motors (1 and 2) rotate clockwise, while the other two (3 and 4) rotate counterclockwise. In this way, gyroscopic effects and aerodynamic torques tend to cancel each other out in trimmed flight.

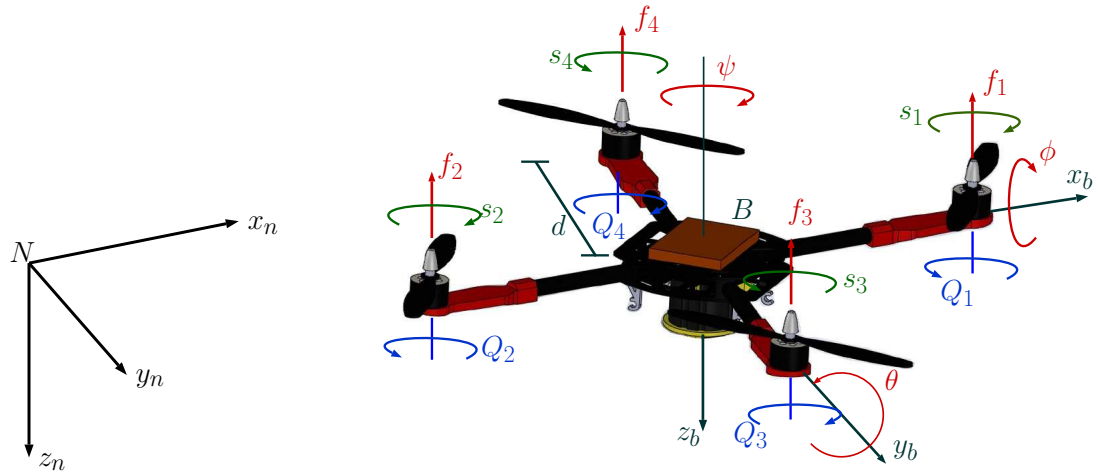


Figure 3.1 – Scheme of the quadrotor configuration: inertial reference frame $N(x_n, y_n, z_n)$, the reference body fixed frame $B(x_b, y_b, z_b)$, the f_i forces on each motor, angular velocity of the motors s_i and the reaction torques Q_i.

Each rotor produces a force f_i parallel to its rotation axe, as well as a drag torque Q_i, opposite to the direction of rotation. The total force or total thrust acting on the helicopter (parallel to the z_b axis) is the addition of the four forces generated by each rotor ($F_T = f_1 + f_2 + f_3 + f_4$). The combination of these forces and the drag torques allow the angular motions over the main axes of the helicopter. Consequently, three movements for the position are produced, see Fig. 3.2.

- **Roll** (ϕ): It is produced by the difference $f_3 - f_4$. To obtain this, the velocity of the right motor m_3 is increased/reduced, while the velocity of the motor m_4 is equally decreased/increased. This difference of forces produces a torque Γ_ϕ around the axis x_b.

- **Pitch** (θ): It is produced by the difference $f_1 - f_2$. It is obtained similarly using

the front and rear motors m_1 and m_3. This difference of forces produces a torque Γ_θ around the axis y_b.

- **Yaw** (ψ): It is the combination of all the reactive torques, $Q_1 + Q_2 - Q_3 - Q_4$. It is obtained by decreasing/increasing the speed of the front and rear motors while decreasing/increasing the speed of the lateral motors. In other words, if a difference of speed between the motors turning in the opposite direction is produced, the reactive torques produce a torque Γ_ψ around the axis z_n.
- **Vertical displacement on the x_n axis**: To go forward or back, the rotational speed of motor m_2 must be decreased/increased, while decreasing/increasing the rotational speed of the motor m_1.
- **Lateral displacement on the y_n axis**: To go to the right or left, the rotational speed of the lateral motors m_4 and m_3 must be decreased/increased.
- **Displacement on the z_n axis**: To go up or down, the torque of all the rotors m_i must be decreased/increased. In the absence of disturbances, the aerial system can perform a hover at a certain height by having a zero translation speed. Then, the total thrust F_T must balance the weight mg of the aerial system by pointing its direction in the axe z_b.

The first three movements are considered in the body fixed frame, and the last three in the inertial reference frame.

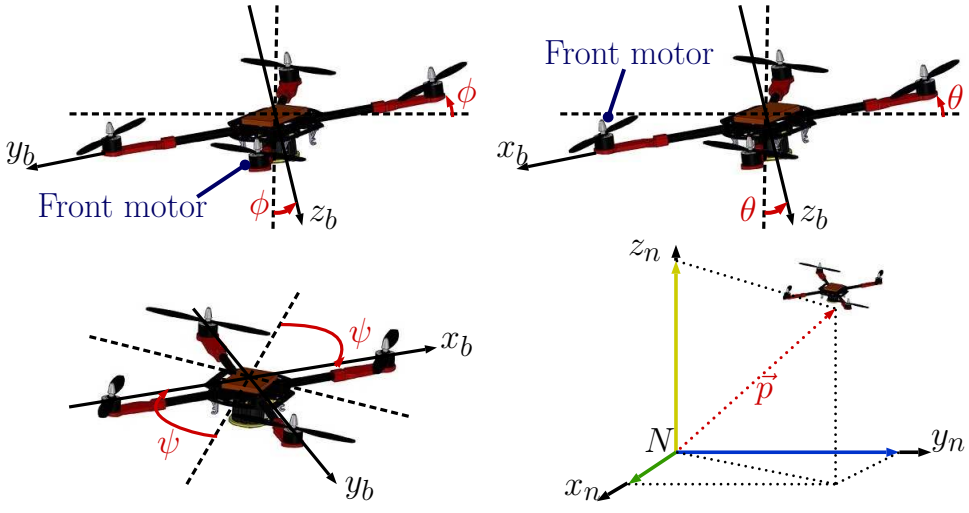


Figure 3.2 – Roll (ϕ), pitch (θ), yaw (ψ) and space displacement

Since the angular movements produce the vertical and horizontal displacements, it is possible to conclude that the linear position of the system depends on its attitude. In this way, the displacement of the quadcopter can be controlled from the attitude control law.

3.2 Mathematical modeling of a quadrotor

There are different approaches to describe the dynamics of the system. For example, the Newton-Euler approach which takes into account the concepts of forces and torques. There is also the Euler-Lagrange approach which considers the concepts of kinetic and potential energy. In this work, the unit quaternion to represent the attitude of the system and the Newton-Euler approach are used in order to obtain the dynamical equations of the system. The dynamical rotation model of the quadrotor can be considered similar as that one of the rigid body.

In order to obtain the equations of motion of the aerial system, some assumptions will be taken into account:

- The cross-shaped structure is supposed to be rigid,
- the quadrotor has a perfectly symmetrical structure, which allows to consider a diagonal inertia matrix,
- the pushing force f_i and the reactive torque Q_i produced by each rotor are supposed proportional to the square of the speed of rotation of the blade,
- the helicopter flight is assumed to be performed under conditions of a standard atmosphere,
- the disturbances produced by the air are neglected.

To obtain the equations of motion of the quadcopter first two coordinate systems are considered (N and B). $B(x_b, y_b, z_b)$ is the coordinate system fixed to the body and is located at the center of mass of this one. $N(x_n, y_n, z_n)$ is the coordinate located at some point in the space (for instance, the earth NED (North-East-Down) frame). The modeling of the system has been reviewed in many works, and is given by the next expressions:

$$\begin{aligned} \dot{\vec{p}} &= \vec{v} \\ \dot{\vec{v}} &= \vec{g} - \frac{1}{m} C^T(q)\vec{F} - \vec{F}_H \\ \dot{q} &= \frac{1}{2}\Xi(q)\vec{\omega} \\ J_h\dot{\vec{\omega}} &= -\vec{\omega} \times J_h\vec{\omega} - \Gamma_G + \Gamma_{\phi,\theta,\psi} \end{aligned} \tag{3.1}$$

where $\vec{p} = (x, y, z)^T$ and $\vec{v} = (\upsilon_x, \upsilon_y, \upsilon_z)^T$ are the linear position and velocity of the center of mass of the body with respect to the inertial frame B. $J_h \in \mathbb{R}^{3\times3}$ is the inertial matrix and m is the mass of the body. $\vec{\omega}$ is the rotational speed and C is the rotational matrix which allows the passage between the inertial and fixed frames. $\vec{F} = [0\ 0\ F_T]^T$, where F_T

is the total force produced by the rotors and is defined in B. $\vec{F}_H = [F_{Hx}\ F_{Hy}\ 0]^T$ is the vector of forces$-H$ in the inertial frame N. Γ_G and $\Gamma_{\phi,\theta,\psi}$ are the gyroscopic forces and the control torques applied to the vehicle. $\vec{g}$ is the gravity vector and is pointing downwards.

The force f_i delivered by the rotor n with a blade rotational speed s_n is modeled by:

$$f_i = bs_i^2 \ \text{ with } \ i \in 1, 2, 3, 4 \tag{3.2}$$

where $b > 0$ is the thrust force parameter. If the four rotors are identical, the expression F_T is given by:

$$F_T = \sum_{i=1}^{4} f_i = b \sum_{i=1}^{4} s_i^2 \tag{3.3}$$

where $k > 0$ is the motor drag constant. Both b and k are parameters that depend on the density of air, the radius, the shape, the pitch angle of the blade and other factors, see (Castillo et al. [2004]).

The components of the forces$-H$ vector are:

$$\begin{aligned} F_{Hx} &= \sum_{i=1}^{4} F_{Hx_i} \\ F_{Hy} &= \sum_{i=1}^{4} F_{hy_i} \end{aligned} \tag{3.4}$$

The reactive couple Q_i generated in the free air by the rotor i due to the motor drag and the total thrust Γ_T produced by the four rotors can be, respectively, approximated by (Kendoul et al. [2005])

$$Q_i = ks_i^2 \tag{3.5}$$

The vector of gyroscopic couples Γ_G is a consequence of the simultaneous rotation of the structure of the quadrotor and the high-speed rotation of the actuators, and it is given by:

$$\Gamma_G = \sum_{i=1}^{4} J_r(\vec{\omega} \times \vec{z}_b)(-1)^{i+1} s_i \tag{3.6}$$

where J_r is the inertia of the so-called rotor (composed of the motor rotor itself with the gears). The components of the control torque $\Gamma \in \mathbb{R}^3$ generated by the rotors are given by $\Gamma = [\Gamma_\phi\ \Gamma_\theta\ \Gamma_\psi]^T$, with

$$\Gamma_\phi = d(f_3 - f_4) = db(s_3^2 - s_4^2) \tag{3.7}$$

$$\Gamma_\theta = d(f_1 - f_2) = db(s_1^2 - s_2^2) \tag{3.8}$$

$$\Gamma_\psi = -Q_1 - Q_2 + Q_3 + Q_4 = k(-s_1^2 - s_2^2 + s_3^2 + s_4^2) \tag{3.9}$$

where d represents the distance from one rotor to the center of mass of the quadrotor. Combining the equation (3.3) with (3.7)-(3.9), the torques and forces applied to the helicopter are written in vector form as

$$\begin{bmatrix} \Gamma_{\phi,\theta,\psi} \\ \Gamma_T \end{bmatrix} = \begin{bmatrix} 0 & 0 & db & -db \\ db & -db & 0 & 0 \\ -k & -k & k & k \\ b & b & b & b \end{bmatrix} \begin{bmatrix} s_1^2 \\ s_2^2 \\ s_3^2 \\ s_4^2 \end{bmatrix} = NS \tag{3.10}$$

with $S = [s_1^2\ s_2^2\ s_3^2\ s_4^2]^T$ the square rotor speeds of the four motors. If we neglect the friction, it should be noted that the relations between the rotation velocities and the resistive torque of the actuator axis and those of the motor are given by

$$\begin{cases} z_m = K_g z \\ Q_m = \frac{Q}{K_g} \end{cases} \tag{3.11}$$

where z_m and Q_m represent the rotation speed and the resistive torque of the motor, respectively. K_g is the gear ratio.

One of the main characteristics of the quadcopter is its symmetry. Consequently, the distribution of the mass can be considered uniform. Considering that the structure is similar at that one of the Fig. 3.3, the classical definitions of inertial moments and inertia product are used, Fossen [1994]. The inertia matrix J_h is then computed in the fixed

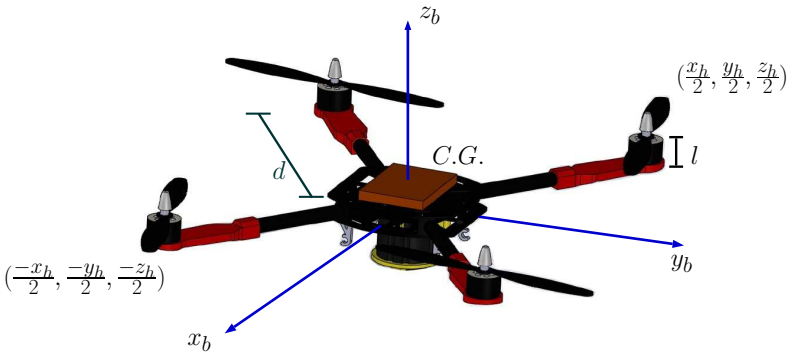


Figure 3.3 – Quadcopter's body.

frame, and is given by

$$J_h = \frac{m}{12} \begin{bmatrix} y_h^2 + z_h^2 & 0 & 0 \\ 0 & x_h^2 + z_h^2 & 0 \\ 0 & 0 & x_h^2 + y_h^2 \end{bmatrix} \tag{3.12}$$

3.3 Model of a quadrotor carrying a manipulator arm

The interest for the Unmanned Aerial Vehicles (UAV) and the Micro-Aerial Vehicles (MAV) has been mainly focused on military applications. However, the civilian applications have also arisen in the last years, for example for the environmental surveillance, traffic monitoring and high voltage lines, cinematography or anthropology. Due to the capacity of performing hovers and the manoeuvrability, the quadcopter or quadrotor has attracted the attention of the community of robotics and aeronautics. This vehicle has the capacity of take off in vertical line, and because of its symmetry it is quite simple to model and build up.

However, one disadvantage present in the quadrotor is its small moment of inertia. Consequently, the system is vulnerable to angular accelerations caused by perturbations coming from the environment or aerodynamical effects due to the angular speed of the rotors.

The attitude dynamics and kinematics for the quadrotor have been reported in many works *e.g.* Castillo et al. [2004], Guerrero-Castellanos et al. [2008]. In these works the model considers that the quadrotor mass distribution is symmetric. However, the mass distribution of a quadrotor with a manipulator is no longer symmetrical and varies with the movement of the arm. Therefore, there exists a displacement of the center of mass, which in general will be off the z body axis of the aircraft. Consider a quadrotor with a manipulator arm with n links attached to its lower part. If the dynamics of the arm is neglected, the attitude kinematics and dynamics are given by

$$\begin{pmatrix} \dot{q}_0 \\ \dot{q}_v \end{pmatrix} = \frac{1}{2}\Xi(q)\vec{\omega} \tag{3.13}$$

$$J\dot{\vec{\omega}} = -\vec{\omega}^{\times} J\vec{\omega} + \Gamma_T \tag{3.14}$$

where q_0 and $q_v \in \mathbb{R}^3$ are the scalar and vector parts of the quaternion, respectively, $\omega = (\omega_1 \ \omega_2 \ \omega_3)^T$ is the angular velocity, $J \in \mathbb{R}^{3\times3}$ is the symmetric positive definite constant inertial matrix of the rigid body expressed in the body frame B and $\Gamma_T \in \mathbb{R}^3$ is the vector of applied torques. Γ_T depends on the couples generated by the actuators (control couples), aerodynamic couples such as gyroscopic couples, gravity gradient or, as in the case of the present work, the couple generated by the movement of a robot manipulator placed under the body. In the present work only the control couples, gyroscopic couples and the one generated by the manipulator is considered in the control design. Consequently,

$$\Gamma_T = \Gamma_{\phi,\theta,\psi} + \Gamma_{arm} + \Gamma_G \tag{3.15}$$

where $\Gamma_{\phi,\theta,\psi}$ and Γ_G have been described in the section 3.1. On the other hand, the vector Γ_{arm} is the torque generated by the total propulsive force being applied at the quadrotor geometric center which is displaced from the center of mass.

3.3.1 Aerial manipulator arm modeling

The movement of the robot manipulator can be seen as a physical pendulum attached to the fuselage of the aerial vehicle for the formulation of a mathematical model. First, reviewing this system, the physical pendulum consists on a rigid body that undergoes fixed axis rotation about a fixed point P, see Fig. 3.4.

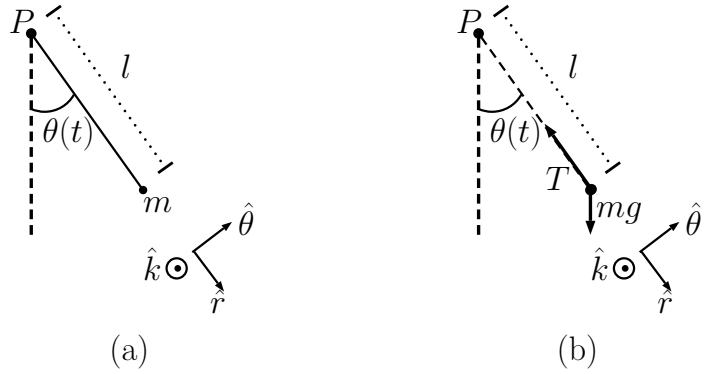


Figure 3.4 – (a) Coordinate system and (b) torque diagram of the pendulum.

In general, the torque about a pivot point is given by

$$\tau = \vec{r} \times \vec{F} \tag{3.16}$$

where τ is the torque vector, $\vec{r}$ is the position vector from the origin of the coordinate system to the point where the force is applied and $\vec{F}$ is the vector force. Then, the torque about the pivot point is given by:

$$\tau_p = \vec{r} \times m\vec{g} = l\hat{r} \times mg(\cos\theta\hat{r} - \sin\theta\hat{\theta}) = -lmg\sin\theta\hat{k} \tag{3.17}$$

where m is the mass of the hanging rigid body, g is the acceleration due to gravity, l is the length of the pendulum, θ is the angle that the pendulum swings away from the vertical and θ, $\hat{r}$ and $\hat{k}$ are the components of the torque acting on the pendulum.

The z component of the torque about point P is

$$(\tau_p)_z = -mgl\sin\theta \tag{3.18}$$

When $\theta > 0$, $(\tau_p)_z < 0$ and the torque about P is directed in the negative $\hat{k}$-direction (into

the plane of Fig). When $\theta < 0$, $(\tau_p)_z$ and the torque about P is directed in the positive $\hat{k}$-direction (out of the plane of Fig. 3.4).

Then, it is possible to extend the physical pendulum to a spherical pendulum, see Fig. 3.5. However, the previous process considers that the pivot point is located on a fixed axis and the mass is hanging around this fixed pivot point. In our case, the aerial vehicle is carrying the manipulator arm and the fixed pivot point becomes into an aerial pivot point.

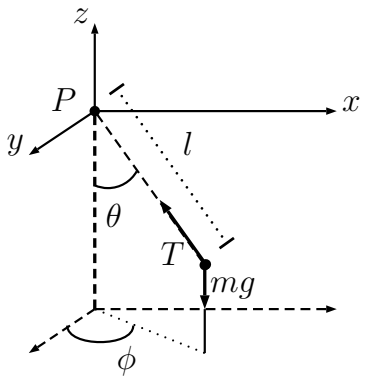


Figure 3.5 – Torque diagram of the spherical pendulum

Taking this into account and neglecting the dynamics for the robot manipulator, it is possible to compute the torques and forces acting on the manipulator due to the gravity through

$$\frac{\partial U_G(C, \theta_{ai})}{\partial C} \tag{3.19}$$

where $U_G : \mathbb{R}^n \to \mathbb{R}$ is the potential energy of the system, which is a function of the rotation matrix of the quadrotor C, given by the equation (1.20) in section 1.2, and the joint angles in the robot manipulator θ_{ai}. The potential energy is given by

$$U_G(C, \theta_{ai}) = -\sum_{i=1}^{n} m_{ai} g e_3^T C \zeta_i \tag{3.20}$$

where m_{ai} is the mass of the i-th link of the manipulmator; g represents the effects of the gravity, $e_3 = (\ 0 \quad 0 \quad 1\ )^T$ and $\zeta_i = (\zeta_{ix}\ \zeta_{iy}\ \zeta_{iz})^T \in \mathbb{R}^3$ is the position of the center of mass of the i-th link of the manipulator with respect to the pivot point.

The total center of mass for the system can be computed through

$$\zeta_c = \frac{1}{m_T}\left[\sum_{i=1}^{n} m_{ai}\varrho_i + m_l \varrho_l\right] \tag{3.21}$$

where $m_T = m_{ai} + m_l$ is the total mass of the manipulator plus a load; m_l is the mass of the load; ϱ_i and ϱ_l are the position vectors of each link of the manipulator and the load,

respectively, both with respect to the reference body frame given by the quadrotor.

Then, applying the derivative given in (3.19), and according to Sangbum [2002], the manipulator torque can be computed by:

$$\Gamma_{arm} = m_T g \zeta_c \times C^T e_3 \tag{3.22}$$

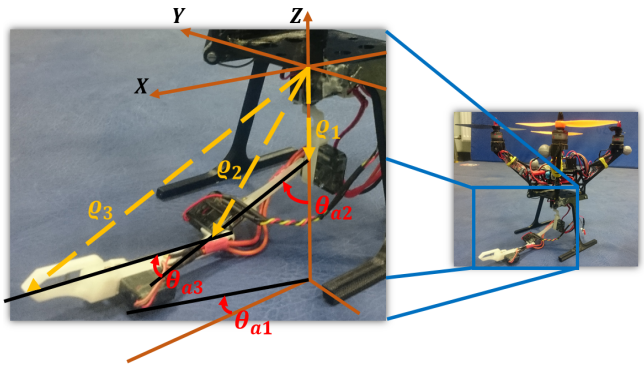


Figure 3.6 – Manipulator arm with three degrees of freedom.

In our case, let consider the scheme in Fig. 3.6, which shows an anthropomorphic arm manipulator. This system has three degrees of freedom and then, the corresponding ϱ_i, where $i = \{1, 2, 3\}$, is given by

$$\begin{aligned}
\varrho_1 &= [\ 0 \quad 0 \quad -l_{c1}\ ]^T \\
\varrho_2 &= [l_{c2}\sin\theta_{a2}\cos\theta_{a1} \quad l_{c2}\sin\theta_{a2}\sin\theta_{a1} \quad -(l_1 + l_{c2})\cos\theta_{a2}]^T \\
\varrho_3 &= [(l_2\sin\theta_{a2} + l_{c3}\sin(\theta_{a2}+\theta_{a3}))\cos\theta_{a1} \quad (l_2\sin\theta_{a2} + l_{c3}\sin(\theta_{a2}+\theta_{a3}))\sin\theta_{a1} \\
&\quad -(l_1 + l_2)\cos\theta_{a2} - l_{c3}\cos(\theta_{a2}+\theta_{a3})]^T
\end{aligned} \tag{3.23}$$

where l_{c1}, l_{c2} and l_{c3} are the distances from the respective joint axes to the center of mass of each link, l_1, l_2 and l_3 are the total length of the links, and θ_{ai} measures the angular displacement from z and x axes.

From equation (3.22), it is possible to estimate the torque generated by the manipulator arm. However, the estimated value is static, since we are not taking into account the velocities and accelerations and the torque computation considers that the angular positions are instantaneous on each link in the manipulator.

Then, since servomotors are used to move the arm manipulator, these can be easily considered as first order systems. For this, a parameter identification based in the technique of closed loop operation, is performed in order to know the different constant values

of the motors. In general, the found system has the form:

$$\theta_{ai} = \frac{Ku(t) - \dot{\theta}_{ai}}{a} \tag{3.24}$$

where θ_{ai} is the angular position of the servo output shaft, $\dot{\theta}_{ai}$ is the angular velocity of the servo, a is a time constant linked to the time response of the servo, K is the gain of the system and $u(t)$ is the input.

Now, using equation (3.24) into equation (3.23) this one becomes:

$$\begin{aligned}
\varrho_{1d} &= [\ 0 \quad 0 \quad -l_{c1}\ ]^T \\
\varrho_{2d} &= [l_{c2}\sin(\tfrac{Ku(t)-\dot{\theta}_{a2}}{a})\cos(\tfrac{Ku(t)-\dot{\theta}_{a1}}{a}) \quad l_{c2}\sin(\tfrac{Ku(t)-\dot{\theta}_{a2}}{a})\sin(\tfrac{Ku(t)-\dot{\theta}_{a1}}{a}) \\
&\quad -(l_1 + l_{c2})\cos(\tfrac{Ku(t)-\dot{\theta}_{a2}}{a})]^T \\
\varrho_{3d} &= [(l_2\sin(\tfrac{Ku(t)-\dot{\theta}_{a2}}{a}) + l_{c3}\sin((\tfrac{Ku(t)-\dot{\theta}_{a2}}{a}) + \tfrac{Ku(t)-\dot{\theta}_{a3}}{a}))\cos(\tfrac{Ku(t)-\dot{\theta}_{a1}}{a}) \\
&\quad (l_2\sin(\tfrac{Ku(t)-\dot{\theta}_{a2}}{a}) + l_{c3}\sin((\tfrac{Ku(t)-\dot{\theta}_{a2}}{a}) + (\tfrac{Ku(t)-\dot{\theta}_{a3}}{a})))\sin(\tfrac{Ku(t)-\dot{\theta}_{a1}}{a}) \\
&\quad -(l_1 + l_2)\cos(\tfrac{Ku(t)-\dot{\theta}_{a2}}{a}) - l_{c3}\cos((\tfrac{Ku(t)-\dot{\theta}_{a2}}{a}) + (\tfrac{Ku(t)-\dot{\theta}_{a3}}{a}))]^T
\end{aligned} \tag{3.25}$$

where ϱ_{id} now are the position vectors of each link of the manipulator but considering also the dynamics linked to the servomotors.

Consequently, equation (3.21) changes to:

$$\zeta_{cd} = \frac{1}{m_T}\left[\sum_{i=1}^{n} m_{mi}\varrho_{id} + m_l\varrho_{ld}\right] \tag{3.26}$$

Finally, equation (3.22) becomes into:

$$\Gamma_{arm_{dyn}} = m_T g\zeta_{cd} \times C^T e_3 \tag{3.27}$$

where $\Gamma_{arm_{dyn}}$ is the manipulator torque taking into account the dynamics of each servomotor present in the robot manipulator.

3.4 Extension to the model of a multi-rotor carrying a n-DOF manipulator arm

The model presented in section 3.3 can be extended to any multi-rotor system with n rotors, where n is an even number. A summary of the entire model of the multi-rotor carrying an articulated arm with $n-$DOF is presented. In general, the model can be divided in three sub-systems: the attitude and position dynamics of the aerial system and

the aerial manipulator modeling. The expressions are given by:

$$\Sigma_A : \begin{cases} \dot{q} = \frac{1}{2}\Xi(q)\vec{\omega} \\ J_h\dot{\vec{\omega}} = -\vec{\omega} \times J_h\vec{\omega} - \Gamma_G + \Gamma_{arm} + \Gamma_{\phi,\theta,\psi} \end{cases} \tag{3.28}$$

$$\Sigma_P : \begin{cases} \dot{\vec{p}} = \vec{v} \\ \dot{\vec{v}} = \vec{g} - \frac{1}{m}C^T(q) \begin{pmatrix} 0 \\ 0 \\ b\ \Sigma_{i=1}^n s_i^2 \end{pmatrix} + \begin{pmatrix} \Sigma_{i=1}^n F_{H_x i} \\ \Sigma_{i=1}^n F_{H_y i} \\ 0 \end{pmatrix} \end{cases} \tag{3.29}$$

$$\Sigma_{\Gamma_{arm}} : \begin{cases} Static \begin{cases} \Gamma_{arm} = m_T g \zeta_c \times C^T e_3 \\ \zeta_c = \frac{1}{m_a}\left[\sum_{i=1}^n m_{mi}\varrho_i + m_l\varrho_l\right] \end{cases} \\ Dynamic \begin{cases} \Gamma_{arm_{dyn}} = m_T g \zeta_{cd} \times C^T e_3 \\ \zeta_{cd} = \frac{1}{m_a}\left[\sum_{i=1}^n m_{mi}\varrho_{id} + m_l\varrho_{ld}\right] \end{cases} \end{cases} \tag{3.30}$$

$$\begin{bmatrix} \Gamma_{\phi,\theta,\psi} \\ \Gamma_T \end{bmatrix} = \begin{bmatrix} o_{11} & o_{12} & \dots & o_{1n} \\ o_{21} & o_{22} & \dots & o_{2n} \\ o_{31} & o_{32} & \dots & o_{3n} \\ b & b & \dots & b \end{bmatrix} \begin{bmatrix} f_1 \\ f_2 \\ \vdots \\ f_n \end{bmatrix} \tag{3.31}$$

where the coefficients o_{ij}, $i = 1, 2$, $j = 1, ..., n$ are derived from the geometry of the multi-rotor and its flight configuration, $o_{3j} = \pm k$, $j = 1, ..., n$, where k is the motor drag constant and the sign depends on the propeller blade direction of rotation. Finally, b is the thrust coefficient of each propeller blade.

During the development of the present project, two multi-rotor configurations were used, the quadcopter and the hexacopter. However, our work was mainly concerned with the use of a quadrotor. With this in mind, we developed a simulator on MATLAB/Simulink, called "Virtual Aerial Carrying Simulator", VACS. The main objective of the VACS is to simulate a real behaviour of the aerial system carrying the manipulator arm, and to test and improve the different control strategies developed for the system.

Through this simulator it is possible to know the behaviour of some of the main system variables, like the quadcopter attitude and linear positions and the manipulator torques and links angular position. The VACS has a virtual reality module, which gives another perspective of the behaviour of the system in the 3d space, see Fig. 3.7.

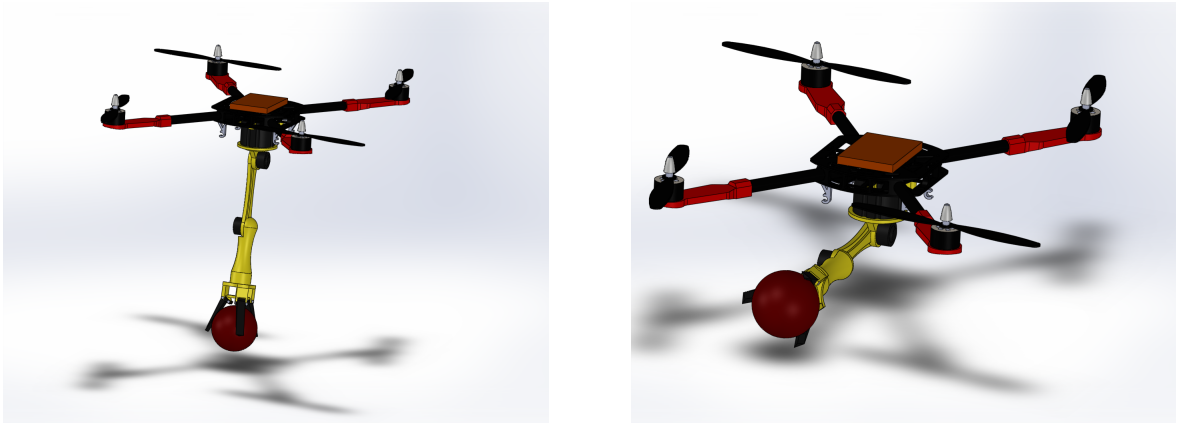


Figure 3.7 – Virtual Aerial Carrying Simulator.

3.5 Conclusions

This chapter was devoted to the presentation of the model of a quadrotor carrying a manipulator arm. To do this, the movement of the arm manipulator is considered as a disturbance, however it was showed that it is possible to estimate such a disturbance by means of two methods. The first one takes into account the instantaneous position of each manipulator link. With this, it is possible to compute the total center of mass of the manipulator and consequently the torque exerted on the quadrotor. The second method improves the estimation of the torque taking into account the arm manipulator dynamics. Since this project makes the use of servomotors, these were modeled as first order systems, from here it was possible to obtain the necessary data to compute the different torques through the dynamic model of the arm manipulator.

Once the mathematical model of the system was obtained, it was possible to develop a virtual simulator. Through this simulator we can see the behaviour of the main variables in the system and also allows the test of the developed algorithms.

Chapter 4
Nonlinear control of a quadcopter carrying a manipulator arm

Once the full model of the system has been developped, the formulation of the attitude and position stabilization problems and their corresponding control laws are presented in this chapter.

First, the problem of the attitude stabilization of a quadcopter carrying a manipulator arm is addressed. Since here, and having the proposed torque estimations, it is possible to formulate an attitude control law which is capable to reduce or avoid the presence of perturbations due to the movement of the manipulator, taking into account the estimated torques.

After that, in order to improve the proposed torque estimations, a nonlinear observer is designed. Its objective is the knowledge of the joints angular position. Due to the usage of servomotors as the manipulator actuators and their operation in open-loop, malfunctions or non-modeled behaviours can be present, deriving into an incorrect torque estimation. The nonlinear observer consists on a Luenberger observer, which fuses the data coming from the servomotors first order model and the computation of the manipulator inverse kinematics in order to know the joint angles in the manipulator arm. The resulting estimation is then used by the manipulator dynamic model, and again, the torque estimation is sent to the attitude control law.

Once the attitude problem is solved, the position stabilization problem is formulated. The objective is to drive the quadcopter to a desired position and keep it there even under the disturbances exerted by the manipulator arm. For this, a nonlinear position control law based on nested saturation functions is proposed.

As a first approach to aerial manipulation, the end-effector position stabilization problem is presented. In this part of the chapter, an approach for the end-effector position stabilization is presented, taking into account concepts like the manipulator workspace.

In the last part of the chapter, some simulation results are presented for the proposed torque estimation methods, as well as the end-effector position stabilization.

4.1 Bounded control

In general, the approach proposed for the stabilization of the aerial vehicle consists on the development of a dynamic system control law with bounded inputs. Since large amplitude disturbances can push the system actuators into saturation, forcing the system to operate in a mode for which it was not designed and even more, from which it may not be able to recover, the usage of this kind of algorithms becomes into a good option of control design. Recently, in Bernstein and Michel [1995] many works are presented giving a good perspective of the control of systems with bounded inputs.

Mainly, the works that consider this kind of approach are focused on the stabilization of linear systems, known as chains of integrators. These systems are represented by the next form:

$$\dot{x} = Ax + Bu \tag{4.1}$$

where the matrix $A \in \mathbb{R}^{n\times n}$ and $B \in \mathbb{R}^n$.

The control amplitude is limited by the physical constraints in the system; in this case the physical constraints in the actuators, given by:

$$-\bar{u} \leq u \leq \bar{u} \tag{4.2}$$

where u is a positive number representing the known limit for the control law.

Then, as it was shown, many approaches have been proposed. The most representatives are:

- Optimal control
- Pseudo-optimal control based on Ricatti equation
- Nonlinear control

Among the precedent approaches, the most noticeable is the nonlinear control. This, due to its simplicity and the regularity properties presented by the approach. This field of research has been initialized by Teel [1992]. Then, this result was generalized in Sussmann et al. [1993]. However, with the approaches previously presented, the system performance is affected as the system dimension increases. Then, these approaches were studied and improved by Johnson and Kannan [2003], Marchand [2003] and Marchand and Hably [2005].

The control laws presented in these works improved the performance in closed loop for systems with multiple dimensions ($n \geq 3$). Then, in Tarbouriech et al. [2006] the authors presented a work where nested saturations were considered. This, because of the limits presented in sensors and actuators.

These results show the interest on systems stabilization by bounded inputs of the community of automatic control. In this way, the nonlinear approach is highly implemented due to its simplicity and high performance.

Finally, and as we expressed before, the problem of the stabilization of an aerial vehicle carrying a rigid manipulator using a bounded control law has not been explored or used in these type of systems, where most of the cited approaches present high complexity.

4.2 Attitude control design

Considering the attitude dynamics of the quadcopter carrying the manipulator arm, where Σ_A and $\Sigma_{\Gamma_{arm}}$ were defined in the section 3.4 and assuming that the links angular position are known, we have the next attitude control problem statement.

4.2.1 Problem statement

The objective is to design a control law which drives the quadrotor to attitude stabilization under the torques and moments exerted to it from the movement of a manipulator arm attached to its lower part. In other words, let q_d denote the constant quadrotor stabilization orientation, then the control objective is described by the following asymptotic conditions

$$q \to q_d,\ \omega \to 0 \text{ as } t \to \infty \tag{4.3}$$

The quaternion error that represents the attitude error between the current orientation and the desired one is given in (1.35). To drive the quadrotor to attitude stabilization, $q_d = [\pm 1\ 0\ 0\ 0]^T$ (inertial coordinate frame), the quaternion error coincides with the current attitude quaternion, that is, $q_e = q$. The control objective is then

$$q \to [\pm 1\ 0\ 0\ 0]^T,\ \omega \to 0 \text{ as } t \to \infty \tag{4.4}$$

Furthermore, it is known that actuator saturation reduces the benefits of the feedback. When the controller continuously outputs infeasible control signals that saturate the actuators, system instability may follow. Then, besides the asymptotic stability, the control law also takes into account the physical constraints of the control system, in order to apply only feasible control signals to the actuators.

4.2.2 Attitude control with manipulator arm

In this subsection, a control law that stabilizes the system described by (3.13) and (3.14) is proposed. The goal is to design a control torque that is bounded.

Definition 4.2.1 *Given a positive constant M, a continuous, nondecreasing function $\sigma_M : \mathbb{R} \to \mathbb{R}$ is defined by*

$$\begin{aligned} &(1)\sigma_M = s \text{ if } |s| < M; \\ &(2)\sigma_M = \text{sign}(s)M \text{ elsewhere;} \end{aligned} \tag{4.5}$$

Note that the components of Γ_{arm_i} are always bounded, *i.e.* $\mid \Gamma_{arm_i} \mid < \delta_i$. Then, one has the following result.

Theorem 4.2.2 *Consider a rigid body rotational dynamics described by (3.13) and (3.14) with the following bounded control inputs $\Gamma = (\Gamma_1 \ \Gamma_2 \ \Gamma_3)^T$ such that*

$$\Gamma_i = -\sigma_{M_{i2}} \left(\Gamma_{arm_i} + \sigma_{M_{i1}}(\lambda[\omega_i + \rho_i q_i])\right) \tag{4.6}$$

with $i \in \{1, 2, 3\}$ and where $\sigma_{M_{i1}}$ and $\sigma_{M_{i2}}$ are saturation functions. Assuming $\delta_i < M_{i2} - M_{i1}$ and $M_{i1} \geq 3\lambda_i\rho_i$. λ_i and ρ_i are positive parameters. Then the inputs (4.6) asymptotically stabilize the rigid body to the origin $(1 \ 0^T \ 0^T)^T$ (i.e. $q_0 = 1, q_v = 0$ and $\omega = 0$) with a domain of attraction equal to $\mathbb{S}^3 \times \mathbb{R}^3 \setminus (-1 \ 0^T \ 0^T)^T$.

Proof 4.2.3 *Consider the candidate Lyapunov function V, which is positive definite.*

$$V = \frac{1}{2}\vec{\omega}^T J\vec{\omega} + \kappa((1-q_0)^2 + \vec{q}^T\vec{q}) = \frac{1}{2}\vec{\omega}^T J\vec{\omega} + 2\kappa(1-q_0) \tag{4.7}$$

where J is defined as before, and $\kappa > 0$ must be determined. The derivative of (4.7) after using (3.13) and (3.14) is given by

$$\begin{aligned} \dot{V} &= \vec{\omega}^T J\dot{\vec{\omega}} - 2\kappa\dot{q}_0 \\ &= \vec{\omega}^T(-\vec{\omega} \times J\vec{\omega} + \Gamma + \Gamma_{arm} + \Gamma_G) + \kappa\vec{q}^T\vec{\omega} \\ &= \underbrace{\omega_1(\Gamma_1 + \Gamma_{arm_1}) + \kappa q_1\omega_1}_{\dot{V}_1} + \underbrace{\omega_2(\Gamma_2 + \Gamma_{arm_2}) + \kappa q_2\omega_2}_{\dot{V}_2} + \underbrace{\omega_3(\Gamma_3 + \Gamma_{arm_3}) + \kappa q_3\omega_3}_{\dot{V}_3} \end{aligned} \tag{4.8}$$

$\dot{V}$ is the sum of the three terms $(\dot{V}_1, \dot{V}_2, \dot{V}_3)$. First $\dot{V}_1$ is analyzed. From Γ_1 in (4.6) and equation (4.8), one gets

$$\dot{V}_1 = \omega_1(-\sigma_{M_{12}} \left(\Gamma_{arm_1} + \sigma_{M_{11}}(\lambda[\omega_1 + \rho_1 q_1])\right) + \Gamma_{arm_1}) + \kappa q_1\omega_1 \tag{4.9}$$

if we choose $\delta_1 < M_{12} - M_{11}$, $\sigma_{M_{12}}$ *is always operating in its linear region so the* $\dot{V}_1$ *becomes*

$$\dot{V}_1 = -\omega_1 \sigma_{M_{11}}(\lambda_1[\omega_1 + \rho_1 q_1]) + \kappa q_1 \omega_1 \tag{4.10}$$

Assume that $|\omega_1| > 2\rho_1$, *that is* $\omega_1 \in]2\rho_1, +\infty[$. *Since* $|q_1| \leq 1$, *it follows that* $|\omega_1 + \rho_1 q_1| \geq \rho_1 + \epsilon$ *for any* $\epsilon > 0$ *sufficiently small. Therefore,* $\omega_1 + \rho_1 q_1$ *has the same sign as* ω_1. *From equation (4.10) and the norm condition on the quaternion,* $\dot{V}_1$ *takes the following form*

$$\dot{V}_1 = -\omega_1 \sigma_{M_{11}}(\lambda_1[\omega_1 + \rho_1 q_1]) + \kappa \omega_1 q_1 \leq -|\omega_1| \sigma_{M_{11}}(\lambda_1(\rho_1 + \epsilon)) + \kappa |\omega_1| \tag{4.11}$$

Taking

$$\kappa < \min(M_{11}, \lambda_1 \rho_1 + \epsilon) \tag{4.12}$$

one can assure the decrease of V_1, *i.e.* $\dot{V}_1 < 0$. *Consequently,* ω_1 *enters* $\Phi_1 = \{\omega_1 : |\omega_1| \leq 2\rho_1\}$ *in finite time* t_1 *and remains in it thereafter. In this case,* $(\omega_1 + \rho_1 q_1) \in [-3\rho_1, 3\rho_1]$. *Let* M_{11} *verify the next inequality* $M_{11} \geq 3\lambda_1 \rho_1$, *equation (4.12) then becomes:*

$$\kappa < \lambda_1 \rho_1 + \epsilon \tag{4.13}$$

For $t_2 > t_1$, *the argument of* $\sigma_{M_{11}}$ *will be bounded as follows*

$$|\lambda_1(\omega_1 + \rho_1 q_1)| \leq 3\lambda_1 \rho_1 \leq M_{11} \tag{4.14}$$

Consequently, σ_{M1} *operates in a linear region*

$$\Gamma_1 = -\alpha_1 \lambda_1 [\omega_1 + \rho_1 q_1] \tag{4.15}$$

As a result, (4.10) becomes

$$\dot{V}_1 = -\lambda_1 \omega_1^2 - \lambda_1 \rho_1 \omega_1 q_1 + \kappa \omega_1 q_1 \tag{4.16}$$

Choosing $\kappa = \lambda_1 \rho_1$ *which satisfies inequality (4.13), one obtains*

$$\dot{V}_1 = -\lambda_1 \omega_1^2 \leq 0 \tag{4.17}$$

The same argument is applied to $\dot{V}_2$ *and* $\dot{V}_3$, *(4.8) becomes*

$$\dot{V} = \dot{V}_1 + \dot{V}_2 + \dot{V}_3 = -(\lambda_1 \omega_1^2 + \lambda_2 \omega_2^2 + \lambda_3 \omega_3^2) \leq 0 \tag{4.18}$$

In order to complete the proof, the LaSalle Invariance Principle is invoked. All the trajectories converge to the largest invariant set $\bar{\Omega}$ *in* $\Omega = \{(\vec{q}, \vec{\omega}) : \dot{V} = 0\} = \{(\vec{q}, \vec{\omega}) : \vec{\omega} = 0\}$. *In the invariant set,* $J\dot{\vec{\omega}} = -[\lambda_1\rho_1 q_1 \ \lambda_2\rho_2 q_2 \ \lambda_3\rho_3 q_3]^T = 0$ *that is,* $\bar{\Omega}$ *is reduced to the origin. This ends the proof of the asymptotic stability of the closed loop system.*

Remark 4.2.4 *Since a quaternion and its negative represent the same physical angular position, there exist two equilibrium points:* $(q_0 = \pm 1, \vec{q} = 0, \vec{\omega} = 0)$, *where* $(q_0 = -1, \vec{q} = 0, \vec{\omega} = 0)$ *can be considered a repeller point (see Joshi et al. [1995]). However, it can be reached using the control law* $\Gamma_i = -\sigma_{M_{i2}}\left(\Gamma_{arm_1} + \sigma_{M_{i1}}(\lambda[\omega_i - \rho_i q_i])\right)$ *instead of the one in (4.6). Therefore, applying*

$$\Gamma_i = -\sigma_{M_{i2}}\left(\Gamma_{arm_i} + \sigma_{M_{i1}}(\lambda[\omega_i + sign(q_0)\rho_i q_i])\right) \tag{4.19}$$

ensures that, of the two rotations of angle β *and* $2\pi - \beta$, *the one of smaller angle is chosen. The demonstration is trivial by adapting the previous proof.*

4.3 Sensorless control for the arm manipulator

4.3.1 Problem statement

The objective of this part of the project is to design a nonlinear observer for the estimation of the joints angles in the manipulator arm, combining the data coming from the first order model of the system (seen in section 3.3.1) and the data coming from the end effector position tracked by a motion capture (MoCa) system. In other words, since the first order model does not fully describes the behaviour of the arm manipulator (non-modeled dynamics, actuators malfunction, etc.), there will be a loss of precision for the torque estimation. However, if we fuse these data with those coming from the arm position tracking by means of a nonlinear observer, the links angular position estimation is improved and consequently the precision of the arm computed torques improves equally. In conclusion, the general performance of the aerial vehicle face to the arm torques is enhanced.

In order to know the manipulator links angular position, we present in a briefly way the inverse kinematics of a three degrees of freedom manipulator arm.

4.3.2 Inverse kinematics of the 3-DOF arm manipulator

According to section 1.4, where the problem of the inverse kinematics was presented, the objective is to find the joint variables in terms of the end-effector position and orientation. In our case, the motion capture system provides the information about the linear position of the end-effector manipulator arm.

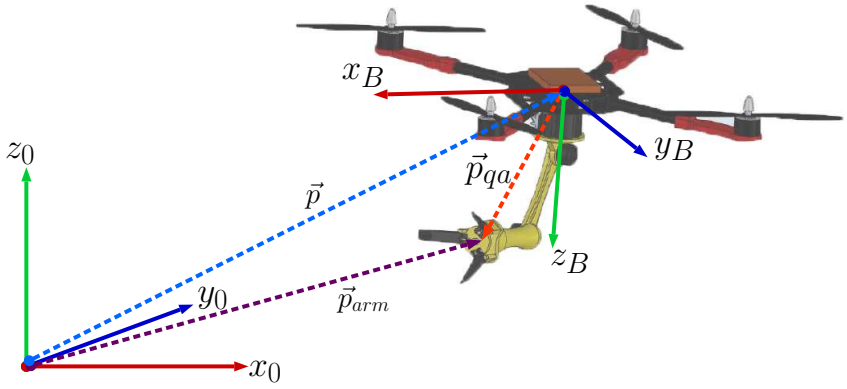


Figure 4.1 – 3d projection of the quadrotor and the manipulator arm.

Then, considering that the quadrotor carries the 3-DOF arm manipulator, the procedure to compute the inverse kinematics is performed with respect to the quadrotor's frame, see Fig. 4.1. Besides

$$\vec{p}_{qa} = R(q)(\vec{p}_{arm} - \vec{p}) = (x_{qa} \quad y_{qa} \quad z_{qa})^T \tag{4.20}$$

where $\vec{p}_{qa}$ is the end-effector position with respect to the quadrotor center of mass, $R(q)$ is the rotation matrix of the quadrotor, $\vec{p}_{arm}$ is the end-effector position in the 3d space, $\vec{p}$ is the position of the quadrotor in the 3d space and x_{qa}, y_{qa} and z_{qa} are the vector components of the position.

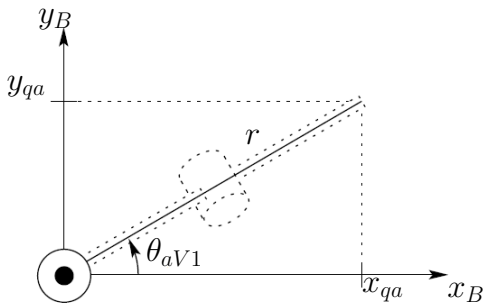


Figure 4.2 – Projection onto the plane formed by the first link.

It is desirable to find the joint variables θ_{aV1}, θ_{aV2} and θ_{aV3}, corresponding to a given end effector position p_{qa} and which represent the estimated angles through the motion capture system and the inverse kinematics. From Fig. 4.2, where the plane $x_B - y_B$ is projected, the first joint is given by

$$\theta_{aV1} = A\tan(x_{qa}, y_{qa}) \tag{4.21}$$

where $A\tan(x_{qa}, y_{qa})$ denotes the two arguments arctangent function, x_{qa} is the position

of the end-effector on the x_B-axis and y_{qa} is the end-effector position on the y_B-axis. $A\tan(x,y)$ is defined for all $(x,y) \neq (0,0)$ and equals the unique angle θ such that

$$\cos\theta = \frac{x}{(x^2+y^2)^{\frac{1}{2}}}; \quad \sin\theta = \frac{y}{(x^2+y^2)^{\frac{1}{2}}} \tag{4.22}$$

This solution for θ_{aV1}, is valid unless $x_{qa} = y_{qa} = 0$.

To find the angles θ_{aV2} and θ_{aV3} for the arm manipulator, given θ_{aV1}, the plane formed by the second and third links is considered, as shown in Fig. 4.3, since the motion of links 2 and 3 is planar it is possible to apply the law of cosines to obtain

$$\begin{aligned} \cos(\theta_{aV3}) &= \frac{r^2+s^2-l_2^2-l_3^2}{2l_2l_3} \\ &= \frac{x_{qa}^2+y_{qa}^2+(z_{qa}-l_1)^2-l_2^2-l_3^2}{2l_2l_3} := C_3 \end{aligned} \tag{4.23}$$

where z_{qa} is the manipulator end-effector position on the z_B-axis.

Since $r^2 = x_{qa}^2 + y_{qa}^2$ and $s = z_{qa} - l_1$. Hence, θ_{aV3} is given by

$$\theta_{aV3} = A\tan(C_3, \pm\sqrt{1-C_3^2}) \tag{4.24}$$

Similarly θ_{aV2} is given by

$$\begin{aligned} \theta_{aV2} &= A\tan(r,s) - A\tan(l_2 - l_3\cos\theta_{aV3}, l_3\sin\theta_{aV3}) \\ &= A\tan(\sqrt{x_{qa}^2+y_{qa}^2}, z_{qa}-l_1) - A\tan(l_2 + l_3\cos\theta_{aV3}, l_3\sin\theta_{aV3}) \end{aligned} \tag{4.25}$$

The two solutions for θ_{aV3} correspond for the elbow-up position and elbow-down position, respectively.

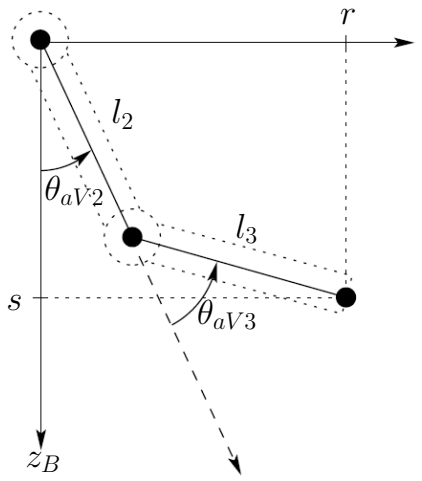


Figure 4.3 – Projection onto the plane formed by links 2 and 3.

Finally, $\theta_{aV1}, \theta_{aV2}$ and θ_{aV3} will be used by the nonlinear observer in order to estimate the real angular positions on each link in the manipulator.

4.3.3 Manipulator links angular position estimation

The objective of the observer is to estimate the present angles on the arm manipulator. For this, the expression in (3.24) is used. The expression allows the modeling of the motors as first order systems, however, after a sufficient long time the real angles can not be identical to those ones in the model. Even more, since the dynamic method makes the use of the velocities and accelerations on each link, the computation of the torque could not be correct.
To face this problem, a Luenberger observer was designed to estimate the angles on each link in the manipulator, which uses the first order system data in conjunction with the data coming from the MoCa system.

Due to the size of the prototype, MoCa system is used to obtain the end-effector position information and then, the inverse kinematics is applied to know the angle on each link. With this in consideration, the expression that describes a link angle is given by:

$$\theta_{aV} = \theta_a + \mu_V \tag{4.26}$$

where θ_{aV} is the measured angle computed with the MoCa system and the inverse kinematics, θ_a is the real angle and μ_V is a noise of minimal value.

On the other hand, the observer allows the computation of the angular velocity, since the parameters of the first order model are used to design the observer. The expressions that represent the observer are given by:

$$\dot{\hat{\theta}}_\delta = a\hat{\theta}_\delta + Ku_\delta + L(\theta_{aV} - \hat{\theta}) \tag{4.27}$$

$$\hat{\theta} = \hat{\theta}_V \tag{4.28}$$

where $\hat{\theta}_\delta$ is the estimated angle on a link in the manipulator, a and K are parameters of the first order system previously presented and L is a positive tuning parameter.

Now, given the expression (4.27), where $\hat{\theta}_\delta$ and $\dot{\hat{\theta}}_\delta$ were estimated, it is possible to compute the manipulator torque from equations (3.23) or (3.24) and use this new term as Γ_{arm} into the attitude control law.

4.4 Position control design

4.4.1 Problem statement

The objective is to design a control law with the inner-outer loop configuration, which stabilizes the quadcopter to a desired position, having the attitude stabilization problem solved. In other words, once the control law has stabilized the attitude of the system, $\lim_{t\to\infty}(R, \vec{\omega}) = (R_d, \vec{0})$, this could be able to stabilize the quadrotor in a desired position, $\lim_{t\to\infty}(\vec{p}, \vec{v}) = (\vec{p}_d, \vec{0})$, and this stabilization must be kept even under the disturbances from the manipulator arm.

4.4.2 Position control with the arm manipulator

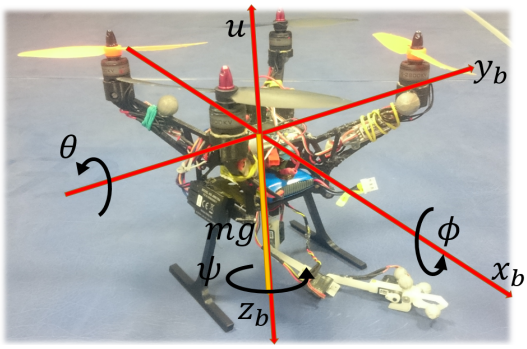


Figure 4.4 – Schematic configuration of a quadrotor carrying a manipulator arm.

The schematic representation of a quadrotor carrying a manipulator arm can be seen in Fig. 4.4, where the inertial reference frame $N(x_n, y_n, z_n)$, the body reference frame $B(x_b, y_b, z_b)$, the force u (thrust) and the weight vector $m\vec{g}$ are depicted. The dynamics of the whole system is obtained with the Newton-Euler formalism and the kinematics is represented using the quaternions formalism, and is given by

$$\Sigma_T : \begin{cases} \dot{\vec{p}} = \vec{v} \\ m_T\dot{\vec{v}} = -m_T\vec{g} + R\begin{pmatrix} 0 \\ 0 \\ u \end{pmatrix} \end{cases} \tag{4.29}$$

$$\Sigma_O : \begin{cases} \dot{q} = \frac{1}{2}\Xi(q)\vec{\omega} \\ J\dot{\vec{\omega}} = -\vec{\omega}^{\times}J\vec{\omega} + \Gamma_T \end{cases} \tag{4.30}$$

where $\vec{p}$ and $\vec{v}$ are linear position and velocity vectors, m_T is the total mass of the system, the quadrotor (m_q), the manipulator and the load (already seen before), $\vec{g}$ the acceleration

due to gravity and R is the rotation matrix.

Note that the rotation matrix R can be given in function of Euler angles, that is

$$R(\phi,\theta,\psi) = \begin{pmatrix} \mathrm{C}\psi\,\mathrm{C}\theta & \mathrm{S}\psi\,\mathrm{C}\theta & -\mathrm{S}\theta \\ \mathrm{C}\psi\,\mathrm{S}\theta\,\mathrm{S}\phi - \mathrm{S}\psi\,\mathrm{C}\theta & \mathrm{S}\phi\,\mathrm{S}\theta\,\mathrm{S}\psi + \mathrm{C}\psi\,\mathrm{C}\phi & \mathrm{C}\theta\,\mathrm{S}\phi \\ \mathrm{C}\psi\,\mathrm{C}\phi\,\mathrm{S}\theta + \mathrm{S}\psi\,\mathrm{S}\phi & \mathrm{S}\theta\,\mathrm{S}\psi\,\mathrm{C}\phi - \mathrm{C}\psi\,\mathrm{S}\phi & \mathrm{C}\theta\,\mathrm{C}\phi \end{pmatrix} \tag{4.31}$$

Taking into account the equations (4.29) and (4.30), this system can be seen as a cascade system, where the translational dynamics (4.29), depends on the attitude (4.30), but the attitude dynamics does not depend on the translational one. This property will be used to design the control law. Now, assume that using the control law (4.6) one can stabilize the yaw dynamics, that is $\psi = 0$. Then, in order to simplify the calculation, system (4.29) becomes:

$$\begin{pmatrix} \dot{p}_x \\ \dot{p}_y \\ \dot{p}_z \end{pmatrix} = \begin{pmatrix} v_x \\ v_y \\ v_z \end{pmatrix} \tag{4.32}$$

$$\begin{pmatrix} \dot{v}_x \\ \dot{v}_y \\ \dot{v}_z \end{pmatrix} = \begin{pmatrix} -\frac{u}{m_T}\sin\theta \\ \frac{u}{m_T}\sin\phi\,\cos\theta \\ \frac{u}{m_T}\cos\phi\,\cos\theta - g \end{pmatrix} \tag{4.33}$$

With an appropriate choice of these target configuration, it will be possible to transform (4.32)-(4.33) into three independent linear triple integrators. For this, take

$$\begin{aligned} \phi_d &:= \arctan\left(\frac{r_2}{r_3+g}\right), \\ \theta_d &:= \arcsin\left(\frac{-r_1}{\sqrt{r_1^2+r_2^2+(r_3+g)^2}}\right) \end{aligned} \tag{4.34}$$

where r_1, r_2 and r_3 will be defined after. Then, choose as positive thrust the input control

$$u = m_s\sqrt{r_1^2+r_2^2+(r_3+g)^2} \tag{4.35}$$

where m_s is the known mass of the system: the mass of the quadrotor and the mass of the manipulator ($m_s = m_q + m_m$). Let be the state $p = (p_1,p_2,p_3,p_4,p_5,p_6,p_7,p_8,p_9) = (\int p_x, p_x, v_x, \int p_y, p_y, v_y, \int p_z, p_z, v_z)$, then (4.32)-(4.33) becomes:

$$\Sigma_x : \begin{cases} \dot{p}_1 = p_2 \\ \dot{p}_2 = p_3 \\ \dot{p}_3 = r_1 \end{cases} \tag{4.36}$$

$$\Sigma_y : \begin{cases} \dot{p}_4 = p_5 \\ \dot{p}_5 = p_6 \\ \dot{p}_6 = r_2 \end{cases} \tag{4.37}$$

$$\Sigma_z : \begin{cases} \dot{p}_7 = p_8 \\ \dot{p}_8 = p_9 \\ \dot{p}_9 = r_3 \end{cases} \tag{4.38}$$

Note that u will be always positive, and $u \geq mg$, in order to compensate the system's weight. Since the chains of integrators given in (4.36)-(4.38) have the same form, a control law can be proposed as in Cruz-José et al. [2012], and can be established by the next theorem:

Theorem 4.4.1 *Consider the quadrotor translational dynamics expressed in (4.32-4.33). Then, the thrust input u given by (3.3) with r_1, r_2, r_3 as in (4.39), where $\sigma_{M_1}(\cdot)$ is defined in (4.5) with $M_1 = 1$ and ς_i are given by (4.40), $a_{(1,2,3)}, b_{(1,2,3)}, c_{(1,2,3)} > 0$ tuning parameters such that $(a,b,c)_1 > (a,b,c)_2 + (a,b,c)_3$, $(a,b,c)_2 > (a,b,c)_3$, stabilizes globally and asymptotically the quadcopter translational dynamics at the origin. Furthermore, if none of the σ_{M_1} are saturated, the poles of the linearized closed-loop for the subsystems (4.36)-(4.38) reside at $-(a,b,c)_1, -(a,b,c)_2, -(a,b,c)_3$, respectively.*

$$\begin{aligned} r_1 &:= -\varsigma_1\{a_3\sigma_{M1}[\frac{1}{\varsigma_1}(a_2p_1 + p_2 + p_3)] + a_2\sigma_{M1}[\frac{1}{\varsigma_1}(a_1p_2 + p_3)] + a_1\sigma_{M1}[\frac{1}{\varsigma_1}(p_3)]\} \\ r_2 &:= -\varsigma_2\{b_3\sigma_{M1}[\frac{1}{\varsigma_1}(b_2p_4 + p_5 + p_6)] + b_2\sigma_{M1}[\frac{1}{\varsigma_2}(b_1p_5 + p_6)] + b_1\sigma_{M1}[\frac{1}{\varsigma_2}(p_6)]\} \\ r_3 &:= -\varsigma_3\{c_3\sigma_{M1}[\frac{1}{\varsigma_1}(c_2p_7 + p_8 + p_9)] + c_2\sigma_{M1}[\frac{1}{\varsigma_3}(c_1p_8 + p_9)] + c_1\sigma_{M1}[\frac{1}{\varsigma_3}(p_9)]\} \end{aligned} \tag{4.39}$$

$$\begin{aligned} \varsigma_1 &= \bar{r}_1/(a_1 + a_2 + a_3), \\ \varsigma_2 &= \bar{r}_2/(b_1 + b_2 + b_3), \\ \varsigma_3 &= \bar{r}_3/(c_1 + c_2 + c_3) \end{aligned} \tag{4.40}$$

Then, the control laws in (4.39) exponentially stabilize the systems (4.36)-(4.38) to the desired position $(p_1, p_2) = (p_{dx}, 0)$, $(p_3, p_4) = (p_{dy}, 0)$ and $(p_5, p_6) = (p_{dz}, 0)$.

Remark 4.4.2 *In the above Theorem, the stabilization goal is the origin. In the case where the asymptotic condition is different from the origin, the variables p_2, p_5, p_8 should be replaced in the control law (4.39) by $e_1 = p_2 - p_x^d$, $e_2 = p_5 - p_y^d$, $e_3 = p_8 - p_z^d$, respectively. In this case p_x^d, p_y^d, p_z^d represent the desired position in the space.*

4.5 Aerial system stabilization strategy: summary

Overviews of the stabilization of the system are depicted in Fig. 4.5, Fig. 4.6 and Fig. 4.7, where all the torque manipulator models are showed together with the control law.

The three methods include the inner-outer loop configuration in order to achieve the quadrotor stabilization under the torques coming from the manipulator. In other words, once the attitude control law has stabilized the orientation of the aerial system, the position control law is able to stabilize the quadrotor in a desired position.

The first stabilization method, shown in Fig. 4.5, takes into account the model of the system, given in section 3.3, the manipulator static torque model, given by (3.22) and the corresponding control laws.
The second stabilization method, shown in Fig. 4.6, takes into account the same model for the aerial system, the computation of the manipulator actuators dynamics, given by (3.27) and the attitude and position control laws presented in the previous sections.
Finally, the third strategy, represented by the overview in Fig. 4.7, takes into account the model for the aerial system, the links angular position estimation from the data coming from the manipulator end-effector position, dynamic manipulator torque model and control laws for the aerial vehicle, introduced in sections 4.2.2, 4.4.2 and 4.3.3

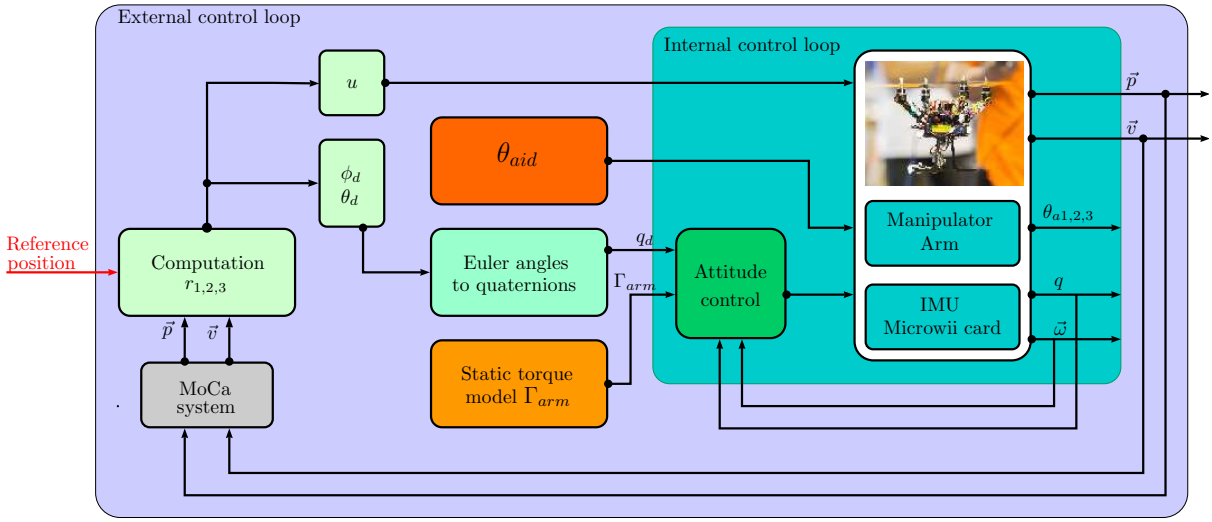


Figure 4.5 – Block diagram of the system with static estimation method.

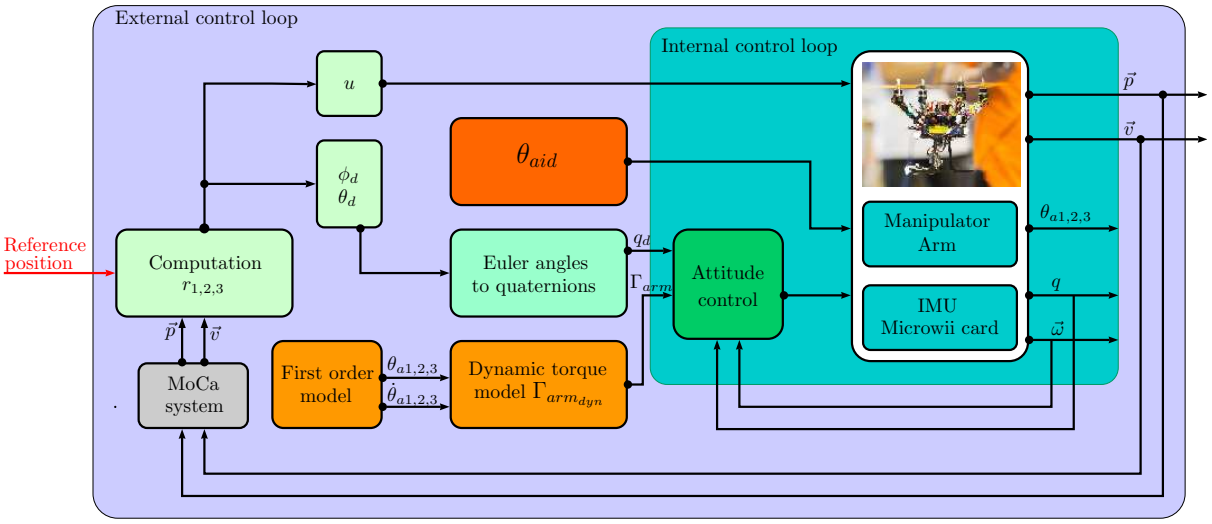


Figure 4.6 – Block diagram of the system with dynamic estimation method.

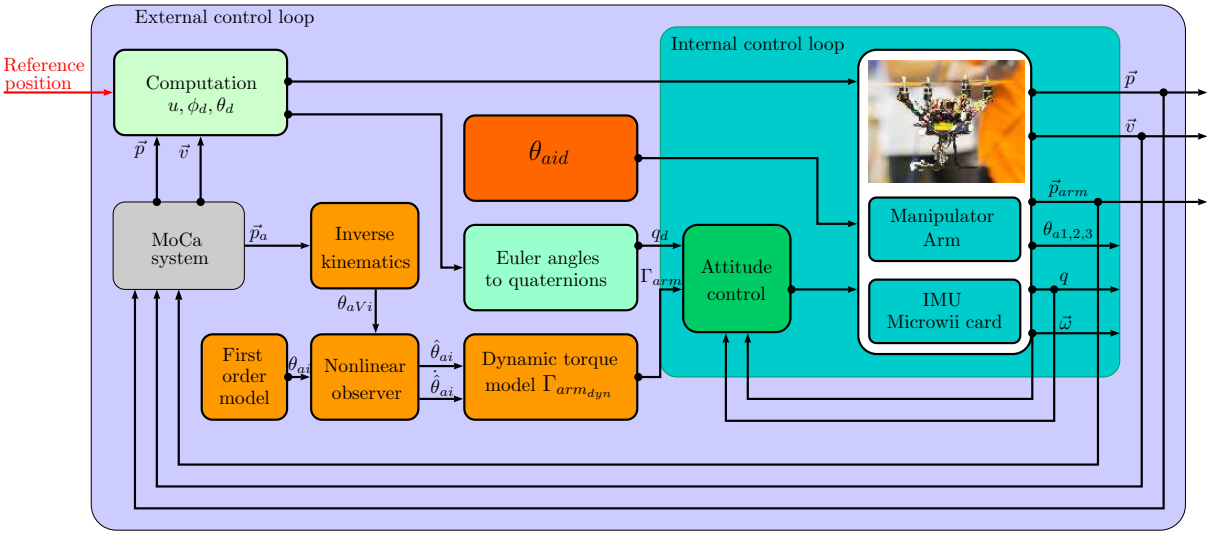


Figure 4.7 – Block diagram of the system with the nonlinear observer and dynamic estimation method.

4.6 Manipulator end-effector position stabilization

Once the quadrotor attitude and linear position problems have been solved through the presented strategies, now it is possible to drive the end effector of the arm manipulator to a desired linear position instead of the quadcopter, and keep it there even under the presence of disturbances coming from the mobile platform, i.e. the quadrotor. For this, we make use of the robot manipulator workspace or operational space, which is the region described by the origin of the end effector frame when all the manipulator joints execute all possible motions. In our case, since we use a 3-DOF configuration arm manipulator,

the corresponding workspace should be a hemisphere pointing down, however, due to the design of the final prototype, the workspace corresponds to the half of a hemisphere. A graphical representation of the arm manipulator workspace and its interaction with the aerial system is shown in Fig. 4.8.

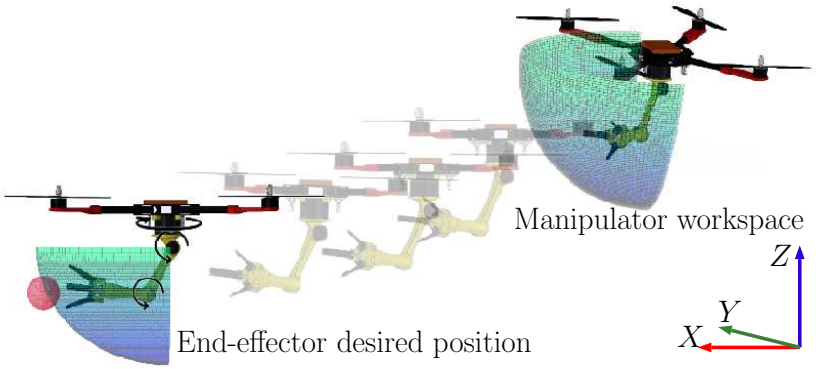


Figure 4.8 – Graphical representation of the aerial manipulation system workspace.

To automatically drive the end effector to a desired position, it is necessary to ensure that the mobile platform is enough close to that position, in other words, the target position must be inside the manipulator workspace. To do this, we take into account the quadrotor and manipulator actual positions. The expressions that allow the stabilization of the end effector are given by:

$$\vec{p}_{armd} = R(q)(\vec{p}_r - \vec{p}) \tag{4.41}$$

where $\vec{p}_{armd}$ is the desired linear position for the end effector, p_r is the reference position and $\vec{p}$ is the actual position of the quadrotor. Then, for the quadrotor:

$$\begin{aligned} \vec{p}_{qd} &= \vec{p}_d - \vec{p} \\ \vec{p}_d &= \vec{p} - \vec{p}_{arm} \end{aligned} \tag{4.42}$$

where $\vec{p}_{qd}$ is the desired linear position for the quadrotor, $\vec{p}_d$ is the position of the quadrotor with respect to the manipulator and $\vec{p}_{arm}$ is the actual position of the end effector. Besides, some conditions and system constraints must be taken into account to accomplish the position stabilization task:

- To prevent the presence of singular configurations, the next conditions are imposed:

$$x_{qa} \quad \text{and} \quad y_{qa} \neq 0 \tag{4.43}$$

$$\theta_{a2} \quad \text{and} \quad \theta_{a3} > 0 \tag{4.44}$$

- To avoid any collision between the manipulator and the quadrotor structure, the manipulator joints have limited rotation range:

$$z_{qa_{min}} \leq z_a \leq z_{qa_{max}} \tag{4.45}$$

$$\theta_{a_{min}} \leq \theta_a \leq \theta_{a_{max}} \tag{4.46}$$

Besides, due to the redundancy of the system, the position equations given in (4.42) must be equally constrained in order to reduce the number of possible solutions for the quadrotor position. These constraints are given by:

$$\begin{aligned} p_{qdy} &= p_{ady} \\ p_{qdz} &\geq p_{adz} \end{aligned} \tag{4.47}$$

where p_{qdy} and p_{qdz} are the quadrotor desired positions in y and z axes respectively and p_{ady} and p_{adz} are the components in y z axes for the desired end effector position. In general, the aerial system must be behind the target point in order to accomplish the task. After that, according to the inverse kinematics equations, given in section (4.3.2), two configurations for links 2 and 3 are possible: elbow up and elbow down. In this case, elbow down configuration is chosen and automatically each joint of the arm manipulator takes an angular position in such a way that the end effector arrives to the target position. Fig. 4.9 shows an overview of the end effector stabilization proposed solution.

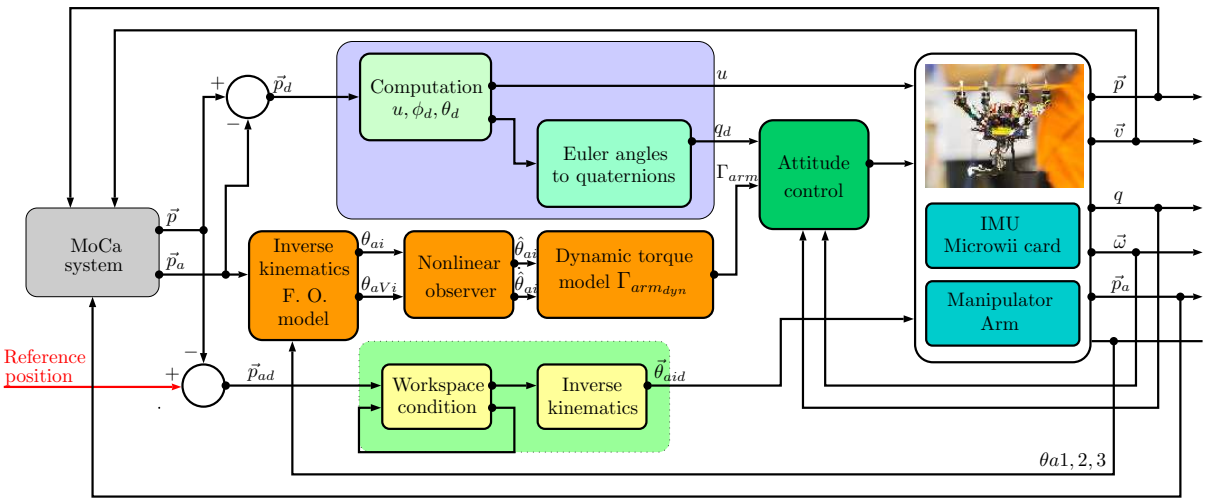


Figure 4.9 – Block diagram of the end effector stabilization.

4.7 Simulation results

In order to test the effectiveness of the control law proposed for the system, a set of simulations were performed using MATLAB/Simulink. The parameters of the system used for the simulation are as follows: $m_T = 316g$, $\max|\Gamma_{1,2}| = 1.5Nm$, $\max|\Gamma_3| = 0.9Nm$ and $\max|u| = 5.5N$, which are values close to the physical prototype parameters.

4.7.1 Setup

The scenario for the simulation is divided in four parts. We consider that the system is driven to $\vec{p}_d = (0\ 0\ 1)^T$ to perform the position stabilization. Then, between time 10s and 25s three movements are performed in the manipulator:

- At time 10s, $\theta_{a1,a3}$ are positioned to 0° and θ_{a2} is positioned at 90°,
- at time 15s, θ_{a1} is positioned to 50° and $\theta_{a2,a3}$ change to 45°,
- at time 20s θ_{a1} changes to $-45°$ and $\theta_{a2,a3}$ remain at 45°,
- finally, at 25s, a disturbance, consisting on a 1s and $0.2N$ amplitude pulse signal, is exerted directly to the manipulator.

4.7.2 Stabilization with manipulator static model

In Fig. 4.10, angular positions angular and linear position and velocity are depicted, where attitude stabilization is achieved. Note that even when we consider the quaternion parametrization, Euler angles, given in (4.31), are used in order to have a better perspective of the behavior of the system. The plots in Fig. 4.11 show the force u and the control torques $\Gamma_{1,2,3}$ which stabilize the quadrotor. It is shown that the control law ensures the stabilization of the quadrotor to the desired position even with the disturbances exerted from the manipulator.

4.7.3 Stabilization with manipulator dynamic model

The same simulation was performed taking into account the dynamic method for the computation of the torques coming from the arm manipulator in order to compare the general performance of the aerial system. As we did in the last subsection, Fig. 4.12 shows the angular position on each link in the arm manipulator and the quadrotor angular and linear positions and velocities, while Fig. 4.13 shows the necessary force to take the quadrotor to a desired altitude as well as the control torques.

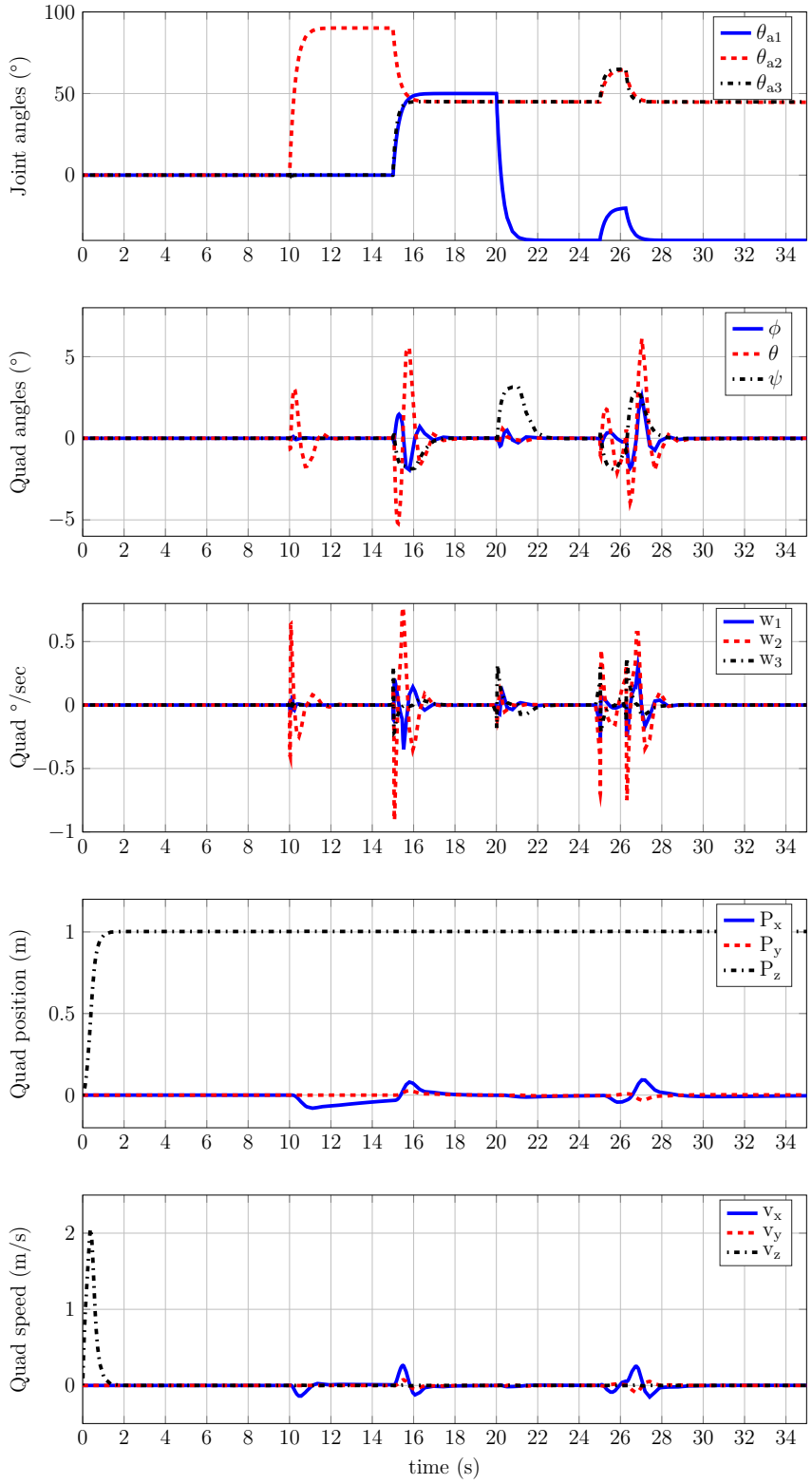


Figure 4.10 – Manipulator links angular position and quadrotor angular and linear position and velocities during the simulation with the static method estimation.

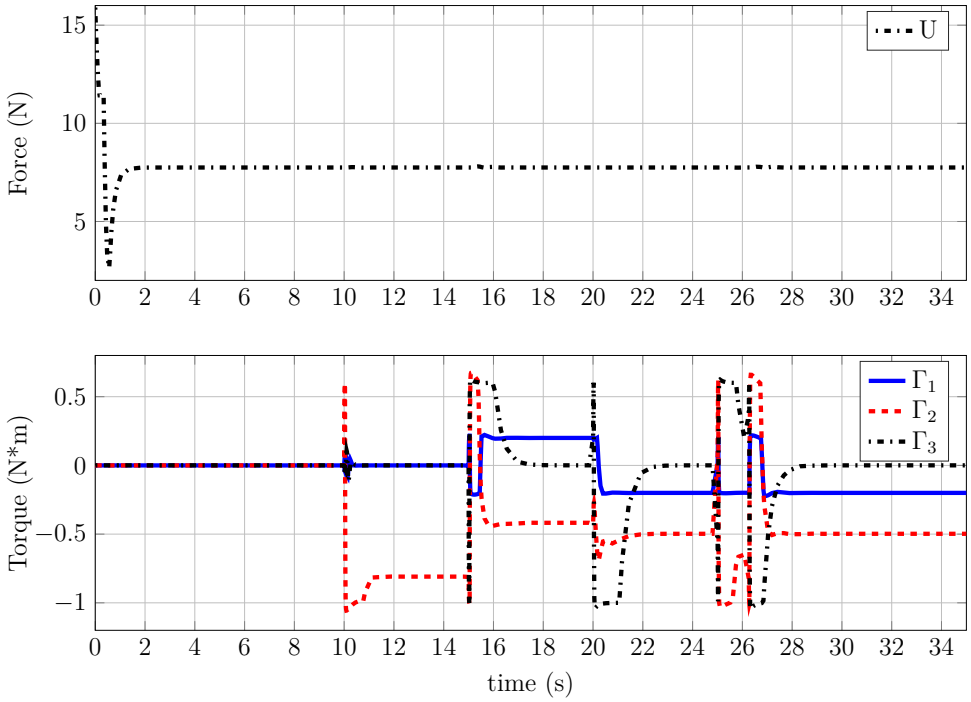


Figure 4.11 – Thrust and control torques signals during the simulation with the static method estimation.

We can see clearly that attitude and position stabilizations are achieved using the dynamic method for the computation of the torques coming from the arm manipulator, and compared to the static method, it is visible that when the dynamics of the servomotors in the arm manipulator is considered, the performance of the system during the simulation is improved.

4.7.4 Nonlinear observer for the arm manipulator

In order to validate the advantages of the usage of the nonlinear observer to improve the angular position estimation of the links in the arm manipulator, a simulation was implemented. The scenario is exactly the same as presented in section 4.7.1.

As before Fig. 4.14 shows the angular and linear position and velocity of the aerial vehicle during the simulation. Besides Fig. 4.15 shows the estimation of the links angles by the first order model proposed before, a virtual measurement for the links angular position, and finally, the output of the nonlinear observer. These curves are specially interesting since they show the result of the fusion coming from the data of the first order model and the virtual measurement for the links angular position. With this, the precision of the links angular position estimation is improved during the transition of the movements of the manipulator arm. Also, since the design of the observer makes the use of the

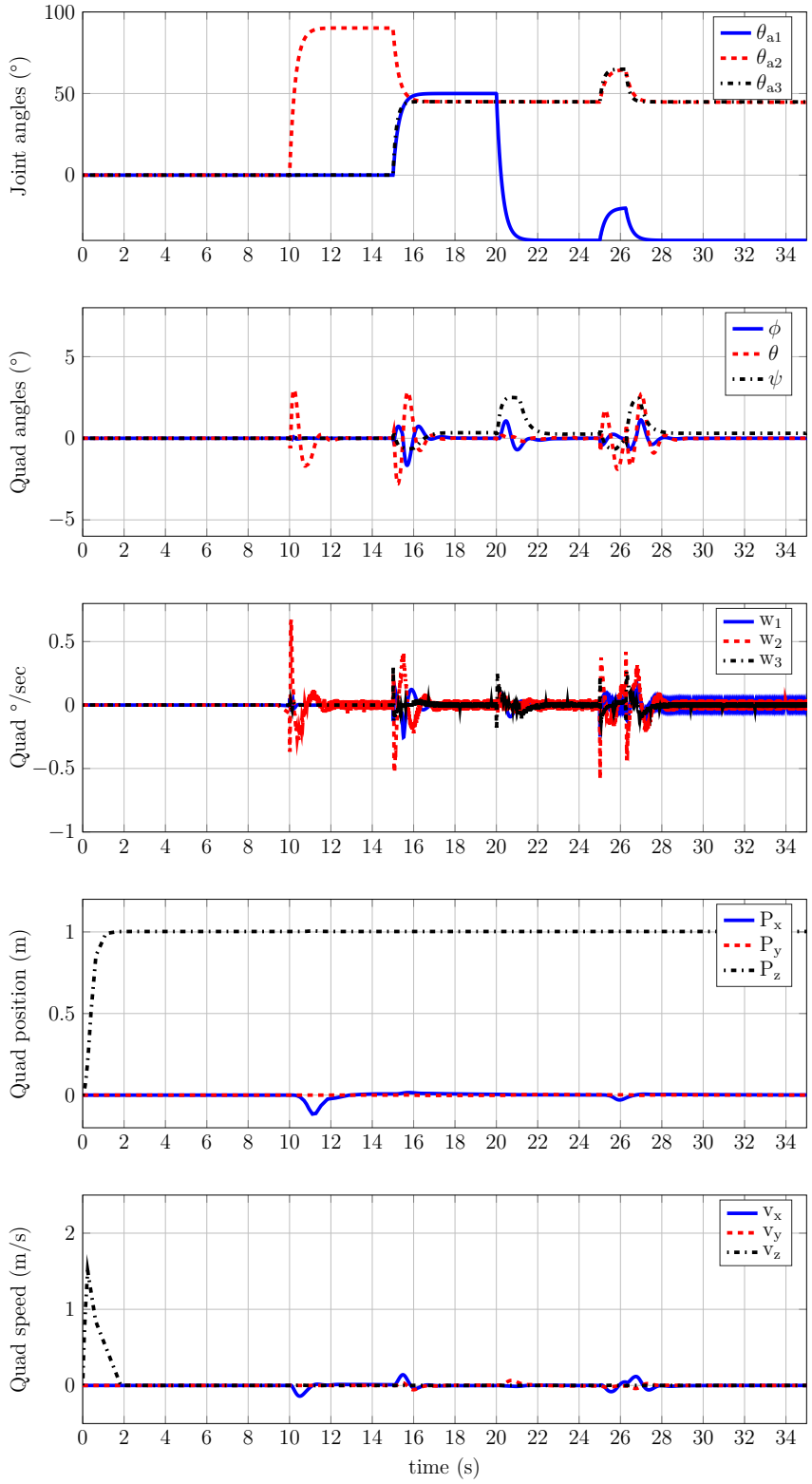


Figure 4.12 – Links angular positions and quadrotor angular and linear positions and velocities during the simulation with the dynamic method estimation.

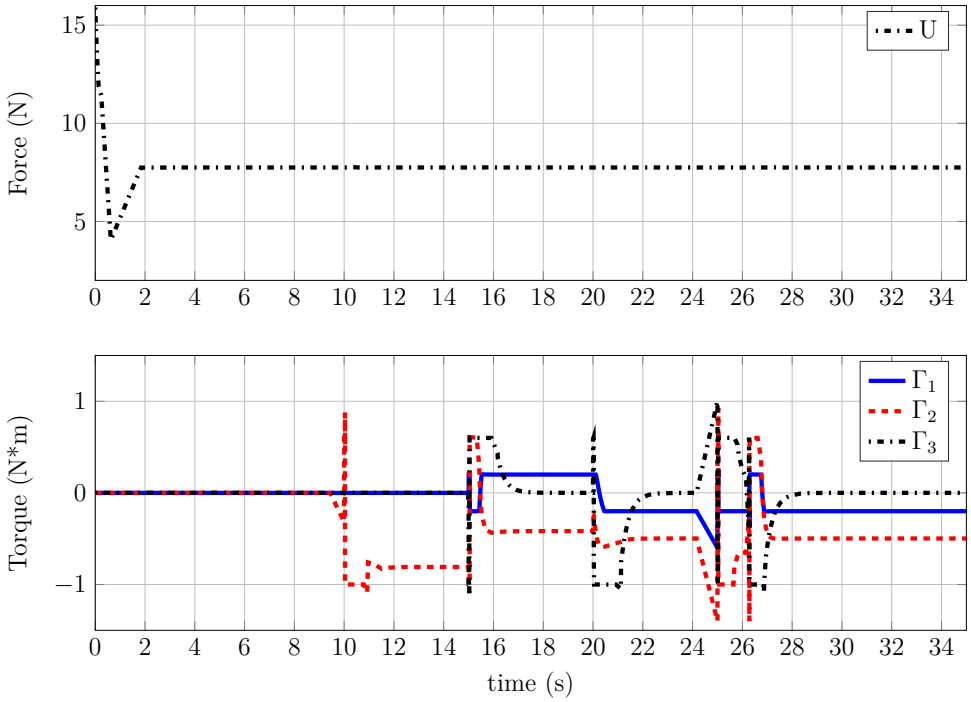


Figure 4.13 – Thrust and control torques signals during the simulation with the dynamic method estimation.

first order model, it allows the retrieve of the angular velocity and acceleration, necessaries for the computation of the manipulator dynamics. Related to attitude and position stabilization performance, there is not a big difference compared to the dynamic method torque estimation, however, since some non-modeled dynamics or actuators malfunctions could be present, this method improves the dynamic method estimation.

4.7.5 Actuator saturations handling

The objective of this simulation is to show the benefits of the usage of saturation functions into the control law. Since the control law takes into account the actuator saturations, only feasible control torques will be produced by the controller, guaranteeing the correct operation of the actuators and consequently the stabilization of the system.

To show this, the simulation scenario consists on add a virtual weight to the end-effector every amount of time in order to see the effects on the stabilization process. More precisely:

- The quadrotor is sent to the position $(0\ 0\ 1)^T$,
- at time 10s, the joint angles in the manipulator change from $(0°\ 0°\ 0°)^T$ to $(0°\ 90°\ 0°)^T$
- at time 15s, a virtual 15g weight is added to the manipulator end-effector,

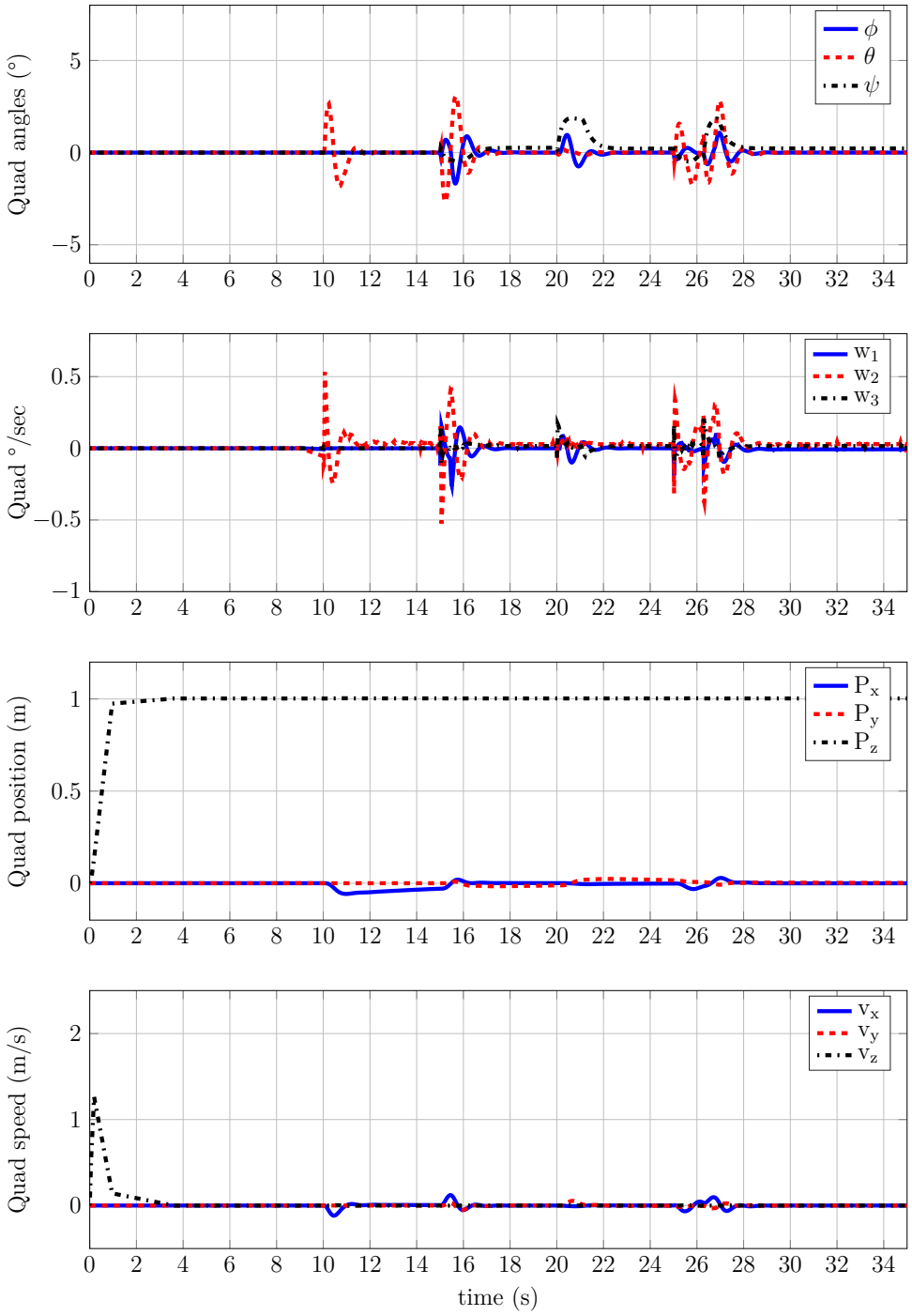


Figure 4.14 – Angular and linear position and velocity of the quadrotor during the simulation with the dynamic method estimation and the nonlinear observer.

- at time 20s, the virtual weight increases to 30g,
- finally, at 25s, the virtual weight is increased to 40g, and the control torques ($\Gamma_{\phi,\theta,\psi}$), are not able to stabilize the system, since the weight produces a torque greater than saturation function values.

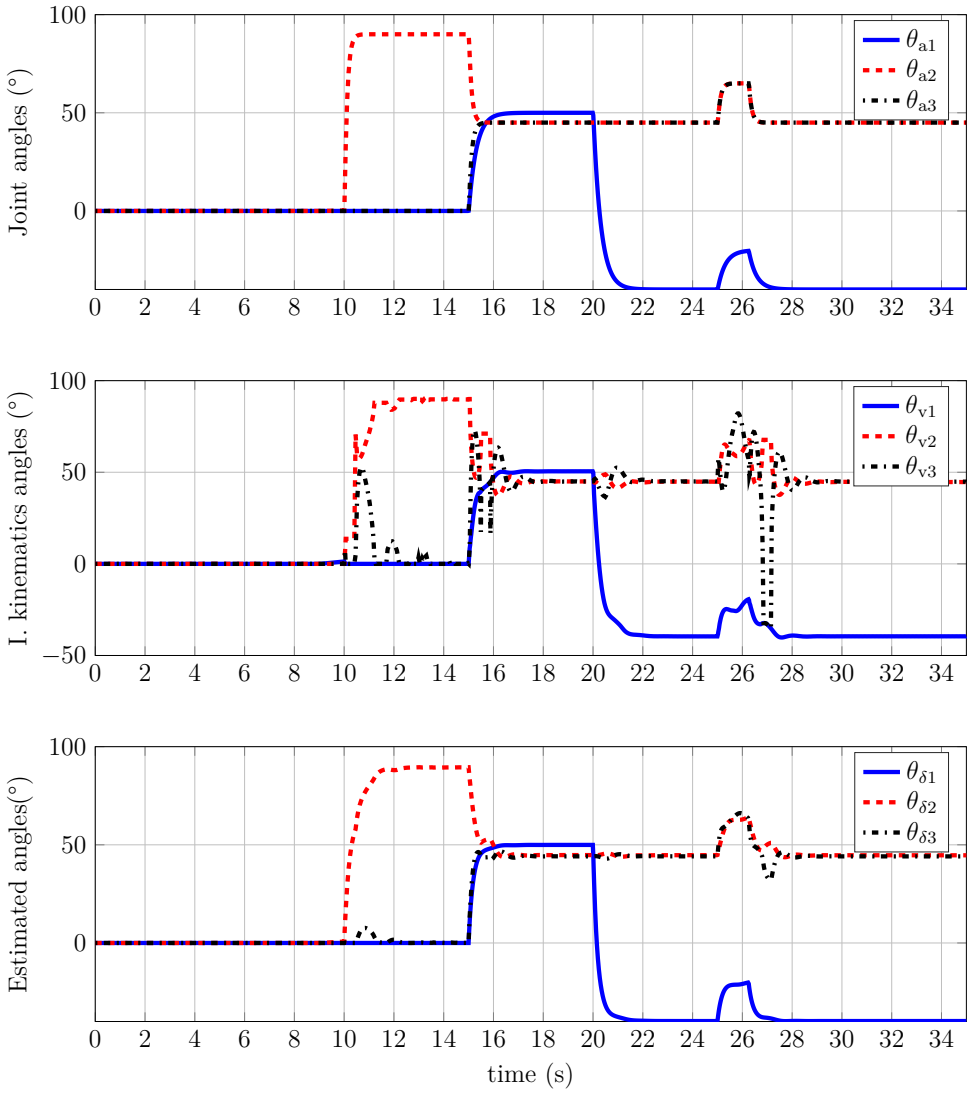


Figure 4.15 – Links angular positions with the first order model, angular positions with inverse kinematics and estimated links angular positions during the simulation with the dynamic method estimation and the nonlinear observer

Fig. 4.16 shows from top to bottom the angular and linear positions and velocities of the quadrotor during the simulation. Then, Fig. 4.17 shows the control torques as well as the total thrust produced by the aerial system. The simulation shows that the stabilization can be guaranteed while the torque coming from the manipulator does not exceed the torque control, which is already bounded according to the system parameters. Once the control torque is surpassed, the stabilization can not be longer guaranteed, as showed in the simulation at time 20s.

Since the manipulator torque estimation is also bounded for the control law, a parameter study for both, aerial system and manipulator arm, must be performed in order to

design and implement a physical prototype.

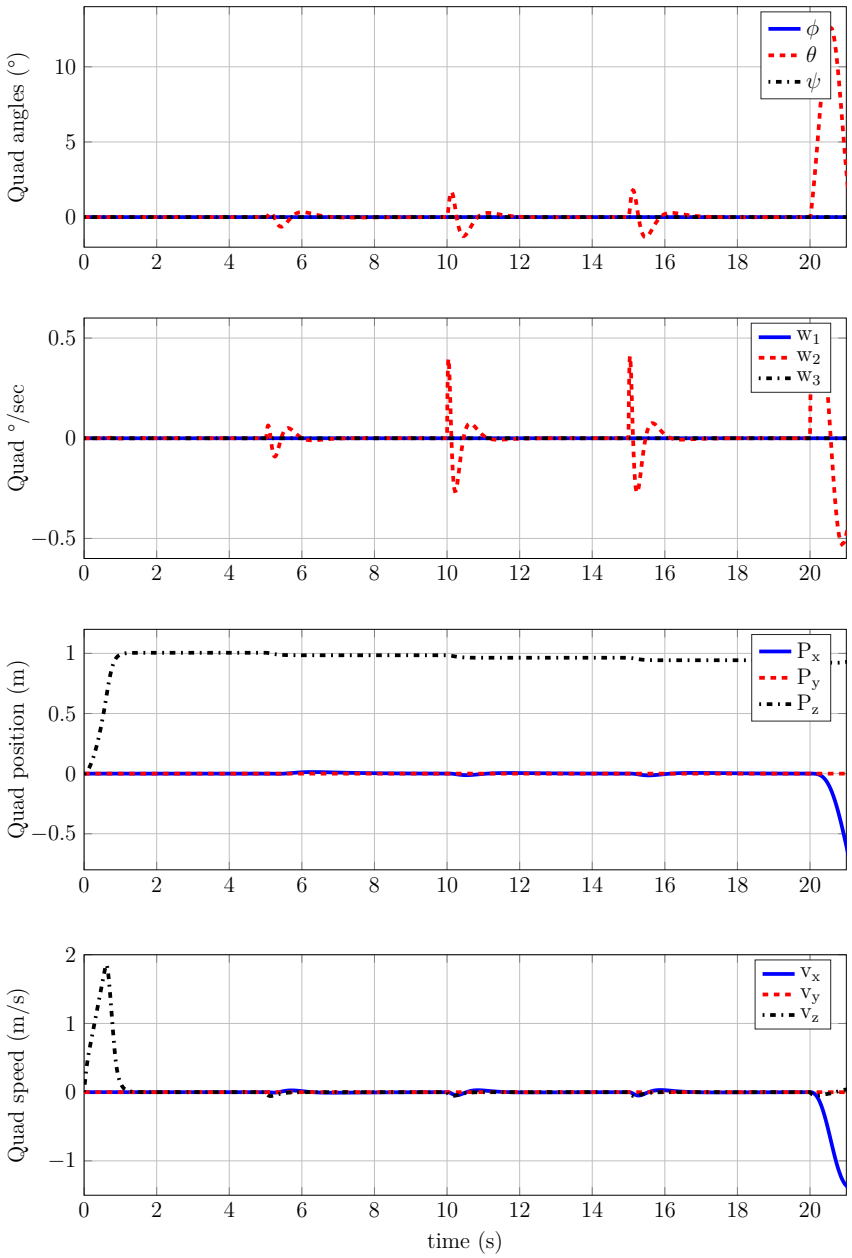


Figure 4.16 – Quadrotor angular and linear positions and velocities during the simulation.

4.7.6 End effector position stabilization

The aim of this simulation is to show the stabilization of the end-effector arm manipulator to a desired linear position. In other words, once the aerial vehicle problem stabilization under the disturbances coming from the arm manipulator has been solved, it is possible to stabilize the end effector arm manipulator to a desired position, and keep it there, even

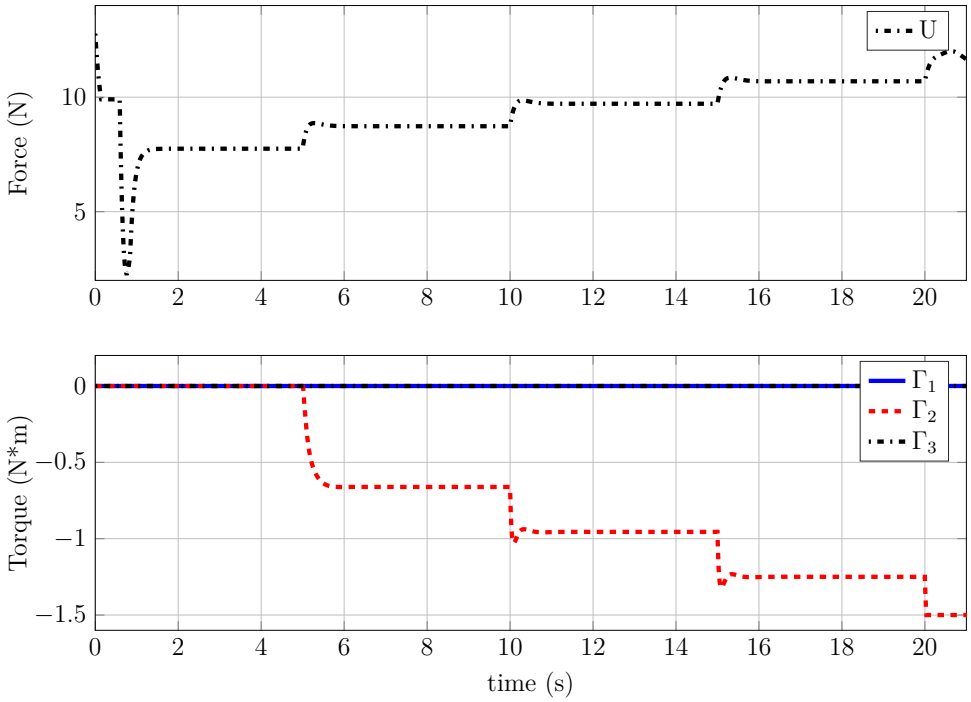


Figure 4.17 – Thrust and control torques signals during the simulation.

if the quadcopter suffers some external disturbances. However, this can be done only if the quadrotor remains inside the robot manipulator workspace.

The simulation scenario is as follows: since the objective is that the gripper reaches the position $\vec{p}_{ad} = (0\ 0\ 1)^T$, the quadcopter is sent to the position $\vec{p}_d = (-0.08\ 0\ 1.05)^T$. In this way, the desired position is inside the arm workspace and this can start the position tracking. Once both systems are stabilized to their respective positions, at time 6s a disturbance is exerted directly on the attitude position of the aerial vehicle, causing also some disturbances on the final linear position. The disturbance consists on two sine functions acting on the roll ϕ and pitch θ quadrotor angles. The one acting on ϕ is a $0.5Nm$ amplitude sine function at $0.6rad/sec$, and the other one is a $0.06Nm$ amplitude sine wave at $2rad/sec$. The simulation runs for 20s.

Fig. 4.18 shows the linear position of the gripper and the links angles of the arm manipulator during the stabilization and Fig. 4.19 shows the linear and angular position and velocity of the quadrotor during the simulation. We can see the effects of the perturbations on both, attitude and linear position on the quadcopter and how they make the aerial vehicle displace on x and y axes. Even under the disturbances, the final effector keeps its position, which validates the effectiveness of the developed algorithm in simulation.

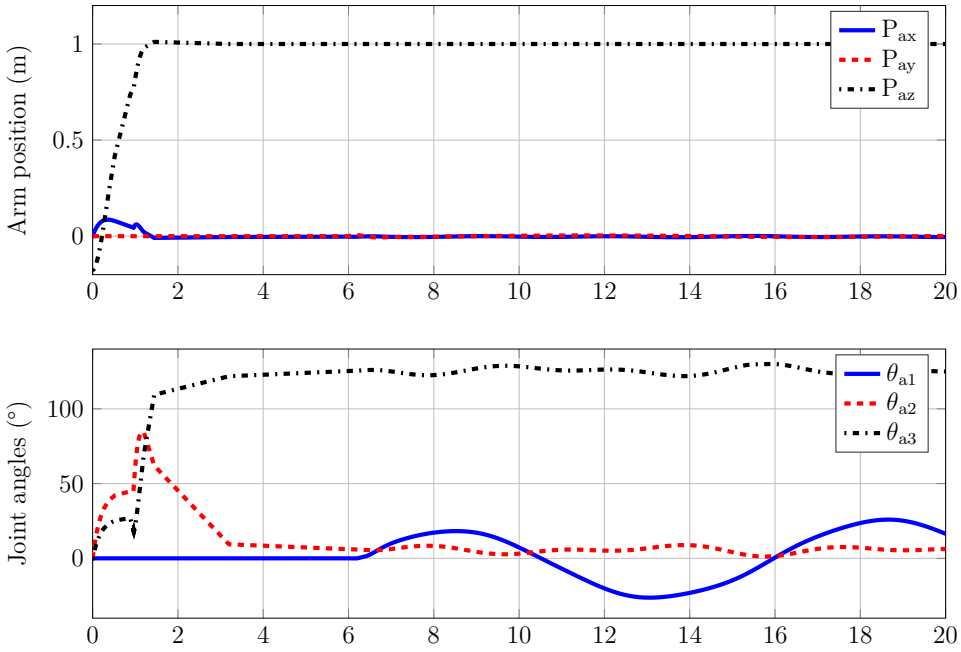


Figure 4.18 – Position of the gripper and links angular position during the manipulator stabilization.

4.8 Conclusions

The control laws for the quadrotor stabilization carrying the manipulator arm were presented in this chapter. First, the attitude controller, which consists on a saturated control law, takes into account the torques coming from the arm manipulator. Since these ones are directly injected on the controller, this guarantees the stabilization of the aerial system. One of the benefits of this controller is that it considers the mechanical constraints of the system, with this, only feasible control signals are applied to the actuators.

After that, considering that the arm manipulator makes the use of servomotors as actuators and these ones work in open loop, only assumptions of the actual position of each link can be made. Consequently, minimal errors can be present on the torque computation. In order to improve the precision of the method, a nonlinear observer was designed, this one uses the data coming from the first order model of the actuators in the arm manipulator and the data obtained from the inverse kinematics computation. The result of this data fusion increases the link position estimation precision and consequently the torque estimation is enhanced.

Then, once the attitude problem with the arm disturbances is solved, the position control law was designed. It consists also on a saturated controller with the objective of stabilizing the quadrotor to a desired position.

After that, the end-effector position stabilization and a solution approach were pre-

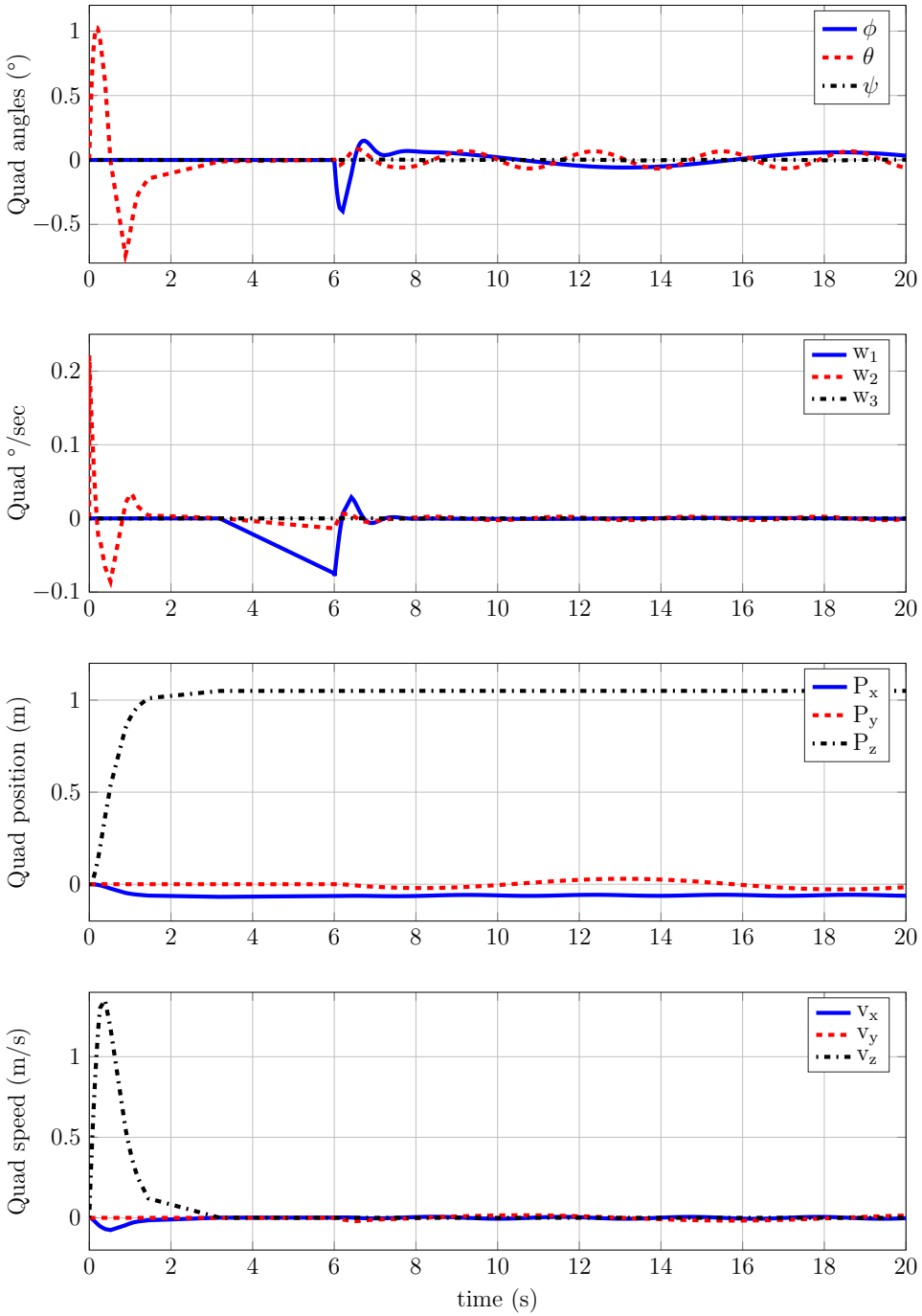


Figure 4.19 – Linear and angular position and velocity of the quadrotor during the arm position stabilization.

sented. Considering that the quadrotor global stabilization problem was solved through the different methods previously presented, it was possible to drive the end-effector manipulator to a desired position using the corresponding inverse kinematics computation.

Finally, some simulation results were presented in order to test the effectiveness of the proposed methods. First, the simulation results for the static method estimation

torque were presented; it was shown that the proposed approach reduces the effect coming from the disturbance. After that, the simulation results for the dynamic method torque estimation were shown, this time the flight performance was improved compared to that one using the static method. Then, the simulation results of the torque estimation through the nonlinear observer were presented. There was not a visible improvement of the flight performance compared to that one of the dynamic method, however, the links angular position estimation is improved aided by the inverse kinematics. Finally, the simulation results of the end-effector position stabilization were presented. It was demonstrated that, even when the quadrotor was under an unknown disturbance, the arm manipulator was able to rest in the desired position.

Chapter 5
Experimental validation

This chapter is devoted to the presentation of the experimental platform, including the MOCA room at GIPSA-lab, ground station, developed aerial platforms and manipulator arm, as well as the presentation of the experimental results and their consequent analysis. First, to implement the position control law, the linear position of the system must be known, for this, the MOCA room was used. This one is composed by 12 cameras and a ground station, which allows the computation of the control inputs through MATLAB/Simulink and sends them to the system through radio signals. More details about this system are also given in this chapter.

During the development of the work, different platforms were used in order to prove the algorithms comprising the different stages of the project.

First, a Flexbot micro quadcopter was used in order to test the attitude and position control laws. The general performance of this model was good, however, when the arm manipulator was added to test the modified attitude control law, the platform was not able to take off due to its reduced specifications. As a result of this, it was decided to move to a bigger platform.

The second platform was the Flexbot micro-hexacopter. The technical specifications of flight controller board and motors were similar to the first one, but this model also offered two extra actuators and a bigger battery, in order to obtain more carrying capacity and autonomy. This platform gave the opportunity to test the proposed method controller, however, due to the dimensions of the arm manipulator, the flight autonomy was not enough to perform longer experimental tests. Face to this problem, some elements of the platform were tuned in order to gain flight time. For the tuned prototype, the frame was enlarged to host larger propellers, motors and battery and also these elements were changed to increase the autonomy, the power and consequently the carrying capacity. Since the new motors were feeded at different voltages, a voltage regulator was added to feed the flight controller board. With this new model, new experimental tests were

performed and the behaviour of the system was improved. The motors used until this point of the project were DC motors and their operational life is not so long, therefore, the performance of these ones were reduced drastically with each test.

Finally, and due to the precedent problems, all the system was changed. Two different frames were totally designed and 3D printed. The flight controller board was changed, with better specifications in terms of processor and general performance. The DC motors were changed to brushless motors, to increase the power and carrying capacity. Consequently, the use of speed controllers for the motors was needed. The propellers were enlarged according to the specifications of the motors as well as the size of the battery.

All the used platforms are described in the next subsections, some characteristics of each one are given, but only the last two, will be deeply detailed.

After that, the experimental results are presented. Basically, the experimental scenarios are the same as the ones presented in the simulation results. First, the experimental results of the behaviour of the system are presented when the orientation control law does not take into account the estimated torque. Then, the static method estimation was used to carry out the second experiment. After that, and in order to see the improvement on the flight performance, the dynamic method torque estimation is used. As we did with the simulation results, we present also the experimental implementation of the nonlinear observer and the obtained results. After that, we move to the experimental arm position stabilization using the dynamic method estimation together with the nonlinear observer.

5.1 MOCA room and ground station

In order to test the developed algorithms, it is necessary to know the attitude and linear position of the system in real time. For this, GIPSA-lab has the MOCA (motion capture) room.

5.1.1 MOCA room

The motion acquisition is made through infrared cameras with emitters and receivers of infrared light and also through reflecting markers attached to the moving objects or individuals. The MOCA room is composed of 12 VICON© cameras (T40 series), attached to a metal structure in high and pointing their vision towards a common area. There are also 8 digital cameras pointing to the same area, but these ones are used for objects reconstruction or motion capture by image processing. With this system it is possible to compute the position and attitude up to 100Hz. Fig. 5.1 shows an image of the MOCA room and the reflecting markers.

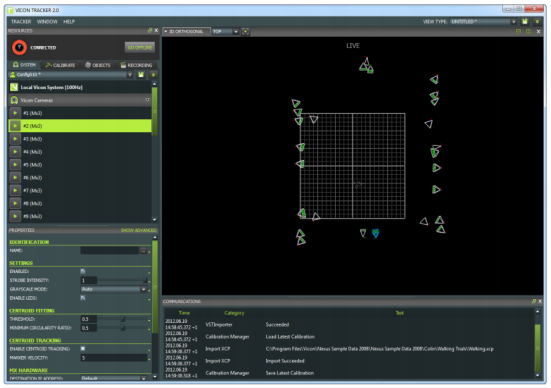
(a)

(b)

Figure 5.1 – (a)MOCA room and (b)reflecting markers

A VICON camera is an infrared camera, which emits and receives infrared rays. A set of cameras pointing towards a common area is able to detect a reflective marker. The markers are little balls of retro-reflecting materials going from 0.5 to 2cm of diameter. The cameras emit an infrared light which makes the receivers sensitive only to this one, when a marker is placed in the area covered by the cameras, it creates a single point in the plane of each one of the cameras (if the area is well covered). Then, the information is collected in a computer running the VICON© tracker software. Fig. 5.2 shows an image of the used VICON cameras and the VICON tracker environment.

(a)



(b)

Figure 5.2 – (a)VICON cameras and (b)VICON tracker environment

5.1.2 Ground station

The ground station is composed by two computers: the first one is under the real time MATLAB/Simulink© environment and a target PC, which is under the xPC target tool-

box, as well as a radio-frequency emitter.

The estimated states (attitude and position) are sent to MATLAB/Simulink through a UDP frame every 2ms. From these data, the position control algorithm is computed and implemented in real-time at 200Hz on the target PC, which uses the xPC target toolbox. xPC Target also manages communications between the host and target PC, as well as the different inputs/outputs of the real-time application.

The control variables are finally sent back to the system through a GIPSA-lab's built-in bridge that converts UDP frames to DSM2 protocol. For this, the radio-frequency emitter is used. Fig. 5.3 shows an overview of the computing process.

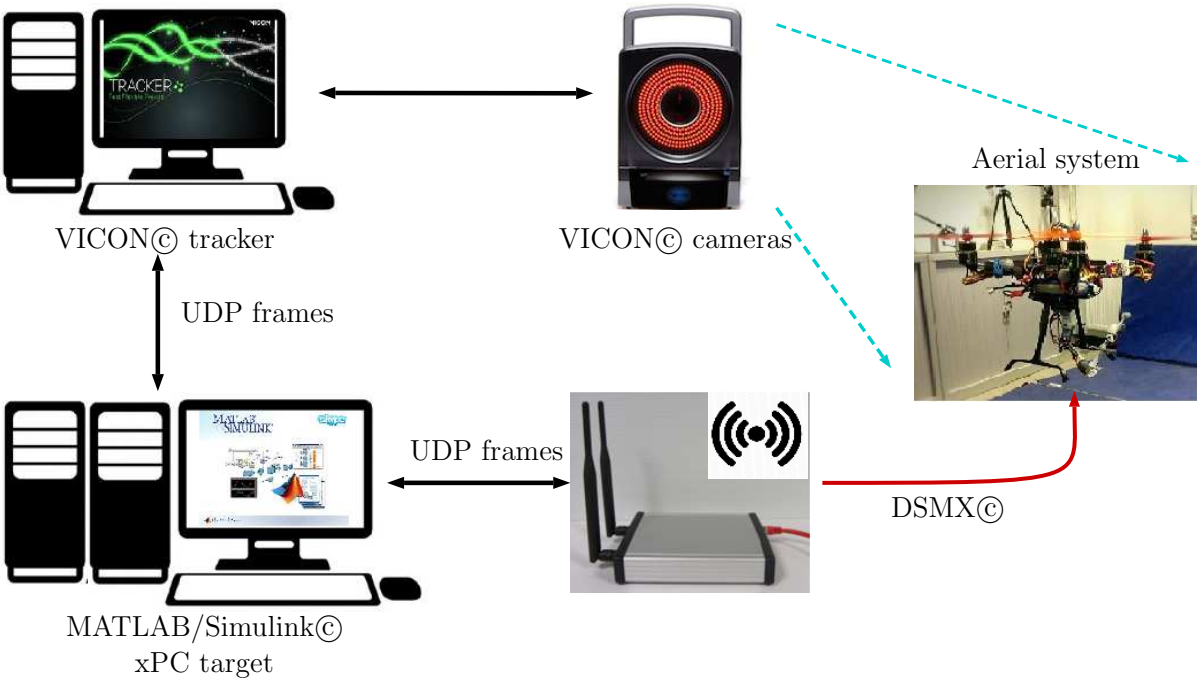


Figure 5.3 – Quadrotor control system process at MOCA room.

5.2 Experimental platforms

Many experimental platforms were built and used for different projects. For the development of this work, 5 experimental platforms were used: 3 FLEXBOT protoypes and 2 home-made platforms. Here, we will present all of them, but our attention will be mainly focused on the home-made prototypes, since these ones allowed to perform most of the tests and to obtain the final results.

For more information about the the FLEXBOT prototypes and the other developed platforms, please refer to B.

5.2.1 Flexbot on the project

Two original flexbot prototypes and a tuned flexbot prototype were used along the present project. First, the flexbot quadrotor was used in order to test the orientation and position control laws but, due to its small size it was not able to take off carrying the manipulator arm. From here, it was decided to move to the Flexbot hexacopter, which has two extra actuators and propellers. However, this time we had some battery constraints and consequently a reduced flight time autonomy. In order to alleviate this problem a 2-DOF arm manipulator was used and it gave the opportunity to test the static estimation torque method.

After that, a tuned Flexbot hexacopter was used. The structure was modified in order to enlarge the carrying capacity (frame, motors and battery were modified) and the same 2-DOF arm manipulator was mounted on the prototype. This configuration allowed the validation of the static estimation torque method for longer tests. TABLE 5.1 shows some physical characteristics of the tuned Flexbot nano-hexacopter and Fig. 5.4 shows the prototype with the arm manipulator in flight.

System	Description	Value	Units
	Mass (m)	70	g
	Distance (d)	7.1	cm
	Battery	3.7	V
	Carrying capacity	32	g
Tuned Flexbot	Inertial moment x (J_ϕ)	0.0016	$Kg \cdot m^2$
hexacopter	Inertial moment y (J_θ)	0.0016	$Kg \cdot m^2$
	Inertial moment z (J_ψ)	0.0037	$Kg \cdot m^2$
	Proportionality constant (b)	615.23	N/s
	Proportionality constant (k)	153.80	N/s

Table 5.1 – Characteristics and parameters of the tuned Flexbot nano-hexacopter.

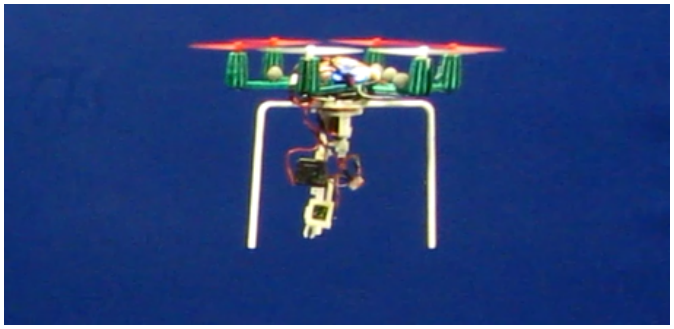


Figure 5.4 – The nano-hexacopter with its 2-DOF arm manipulator in flight.

5.2.2 Home-made prototype: AeCa robot

Hardware

To improve the robustness of the proposed control laws, a new experimental platform was developed. The aerial system consists on a home-made quadcopter and a 3-DOF arm manipulator, named "Aerial Carrying robot" (Aeca). Both structures were specially designed and built for this project. The characteristics and parameters of the mini-quadcopter are described in the TABLE 5.2. The total weight of the quadcopter is about 280g and its carrying capacity is about 80g. The system is depicted in Fig. 5.5.

System	Description	Value	Units
	Mass (m)	280	g
	Distance (d)	10.7	cm
	Battery	7.4	V
	Carrying capacity	80	g
Quadcopter	Inertial moment x (J_ϕ)	0.0056	$Kg \cdot m^2$
	Inertial moment y (J_θ)	0.0056	$Kg \cdot m^2$
	Inertial moment z (J_ψ)	0.0097	$Kg \cdot m^2$
	Proportionality constant (b)	4.136×10^{-6}	$N/(rad/s^2)$
	Proportionality constant (k)	3.153×10^{-7}	$Nm/(rad/s^2)$

Table 5.2 – Characteristics and parameters of the mini-quadcopter.

Figure 5.5 – The mini-quadcopter with its arm manipulator.

Flight controller board

The attitude control law (4.6) for the quadcopter was programmed on this new Copter board, which features an IMU sensor (MPU6050), integrated by gyros and accelerometers, the HMC5883L as 3-axis digital magnetometer, a MS5611-01BA03 high precision

altimeter and the ATMega 2560-16AU as processor. The processor consists on a high-performance, low-power Atmel 8-bit AVR RISC-based microcontroller and it combines 256KB ISP flash memory, 8KB SRAM, 4KB EEPROM, 86 general purpose I/O lines, 32 general purpose working registers, real time counter, six flexible timer/counters with compare modes, PWM, 4 USARTs [1], byte oriented 2-wire serial interface, 16-channel 10-bit A/D converter, and a JTAG [2] interface for on-chip debugging. The device achieves a throughput of 16 MIPS at 16 MHz and operates between 4.5-5.5 volts.

The board dimensions are of $50 \times 50 \times 11.66mm$ and the weight of $14.5g$. The motors are connected to the card though the speed controllers (ESC), since these are brushless motors. Fig. 5.6 shows the CRIUS flight controller board.

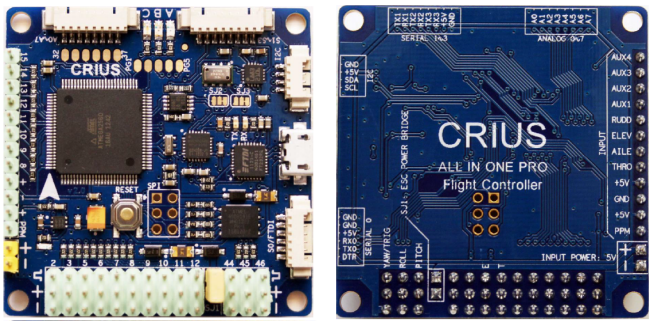


Figure 5.6 – CRIUS flight controller board.

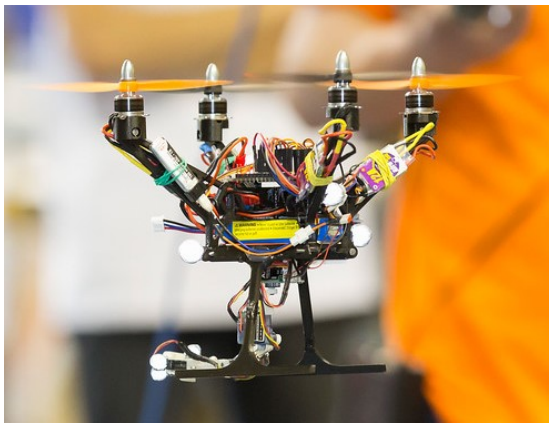
Communication

In order to communicate between the ground station and the platform, a Spektrum© AR6115e module was used. This one features a 2.4GHz 6-channel park flyer receiver with red led indicator under the DSM2/DSMX© protocol.

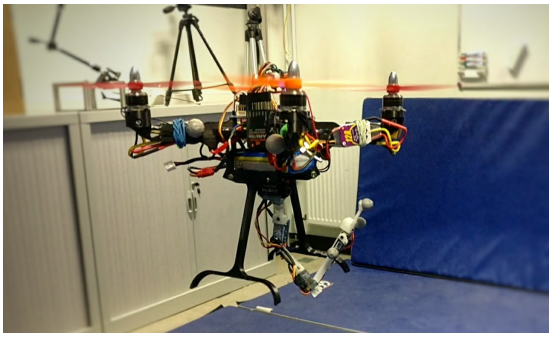
Frame

Two different frames were designed and 3D printed. Since these designs have the actuators at different positions with respect to the quadrotor center of mass, the objective of their construction was to test and validate a better flight performance with the manipulator arm. Both systems are shown in Fig. 5.7a and Fig. 5.7b. After some experiments, it was concluded that the AeCa robot version 2.0, with the medium height located actuators had a better performance in flight, since the actuators are closer to the center of mass and the pendulum effect, present at the top located actuators structure, is reduced.

1. Universal Synchronous/Asynchronous Receiver/Transmitter
2. Joint Test Action Group: Electronics industry association responsible for the IEEE Standard 1149.1-1990, “Standard Test Access Port and Boundary-Scan Architecture”



(a) AeCa robot version 1.0 (Top located actuators).



(b) AeCa robot version 2.0 (Medium height located actuators).

Figure 5.7 – 3D printed frames of the final prototypes.

5.2.3 Battery effects and motor speed control

The prototype uses brushless motors as actuators, connected to the control board card through the speed controllers. However a common problem experienced with quadcopters is the time-variant thrust response due to a drop of the battery voltage. In other words, for a same command received by the flight controller, the resulting thrust given by the motors will depend on the battery state of charge. This time-variant response of the quadcopter requires a high integrator gain in the position controller which results in adding too much phase. For the flexbot, the DC motors are controlled directly by the flight controller card through a power stage converter, but for the actual quadrotor, as power is required for the motors, Electronic Speed Controllers (ESC) are used in order to provide the necessary amount of power and to handle the 3-phases of the motors. However, most of the ESCs do not achieve closed-loop control of the motor speed, and are then sensitive to the voltage drop of the battery. BLHeli is an open source project intended for replacing the official firmware of different ESCs. The main advantage is that it provides a sensorless closed-loop control mode of the motor speed. Therefore, the rotation speed should not be impacted by the battery's state of charge. Several adjustable parameters are available, however finding the good values for our setup is quite difficult without any objective measurement. For this reason, a test bench for the couple motor/ESC has been build in order to quantify the effect of the different tuning parameters for a PI controller.

Motor test bench setup

The only data we want to acquire is the speed of the motor in order to measure the response time and to check if the closed-loop control is able to reject a voltage drop.

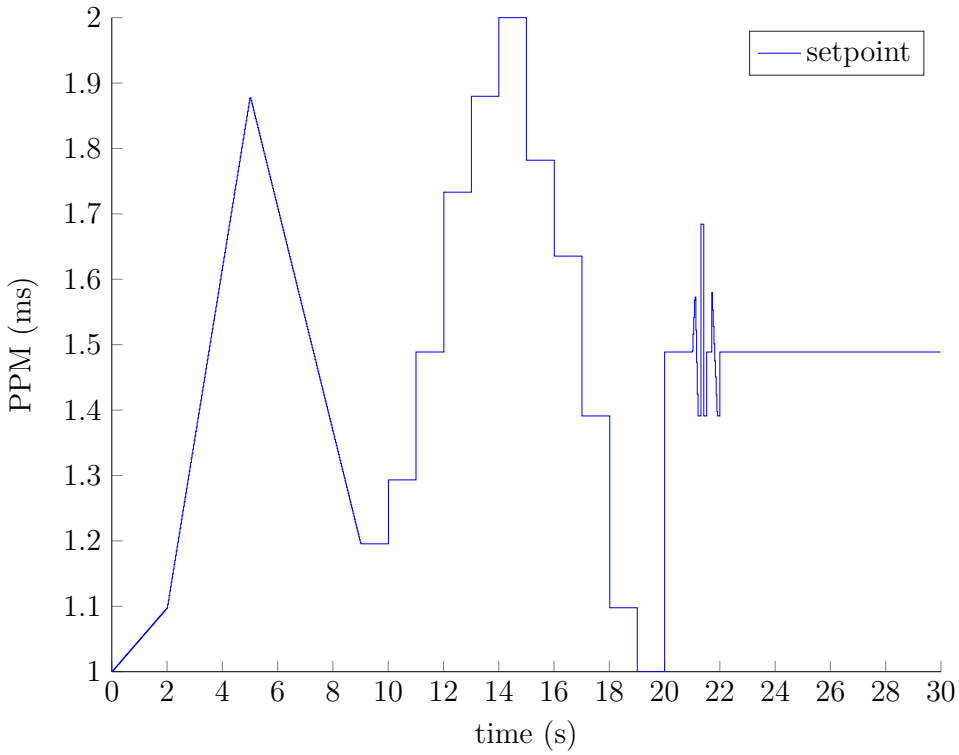


Figure 5.8 – Profile input for the tuning of the motor control loop.

A hall sensor effect is used to measure the angular speed, for the considered model of the motor, the poles are directly visible by the sensor. If it would not be the case some magnets can be glued around the motor to trigger the sensor. An arduino Uno has been programmed to compute the frequency given by the hall sensor effect, this frequency is then sent over USB to a PC periodically. The PC sends several setpoints through USB to another arduino card which generates the PPM signal for the ESC: 1ms pulses translate to zero throttle, 2ms pulses are full throttle, Fig. 5.8 shows the input profile sent to the ESC, the first part (from 0 to 10 seconds) consists on an ascending and descending ramp, the second part (from 10 to 20 seconds) tests several step responses, the third part (from 20 to 22 seconds) consists on a high frequency reference, and during the last part, the voltage of the power supply is dropped by 1 Volt. A Labview real-time software has been built to send, receive and save the different data. Fig. 5.9 shows an overview of the motor test bench.

Results

18 configurations of PI control parameters have been tested. Figure 5.10 shows the measured speed of the motor for 4 different tuning parameters. It can be seen that the response to the ramp input is the same for all the closed-loop parameters. To determine the best values for the proportional gain (Kp) and the integral gain (Ki), we considered two impor-

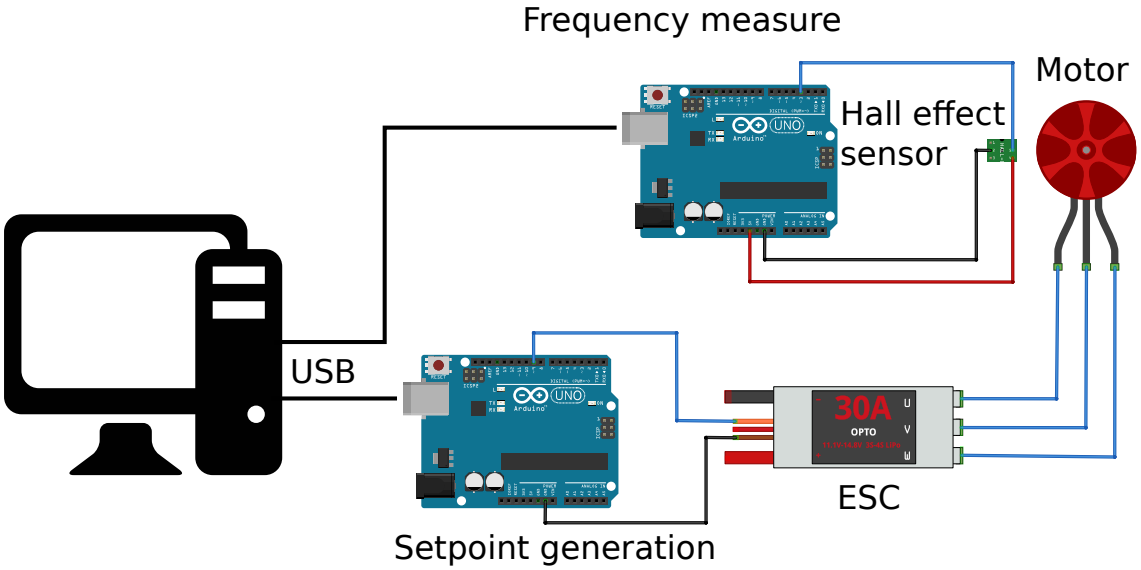


Figure 5.9 – Test bench ESC + motor.

tant behaviors: the rejection of a voltage drop and the step response. Figure 5.11 shows a detailed zoom around two regions of interest:

- It can be seen clearly on Figure 5.11a that a voltage drop of 1 Volt (near 24 s) results in a speed drop of about 15 Hz for the open-loop control, whereas all the closed-loop response are less impacted and tend to recover the speed drop. As the drop was manually applied to the power supply, it is not synchronized between all the experiments.

- Figure 5.11b shows the response to a step input, at $t = 11$s the PPM signal input of the ESC goes from 1.29 ms (29 % full speed) to 1.49 ms (49 % full speed). The open-loop response has not been plotted because it does not converge to the same value.

The parameters which have been found to give the best performances in terms of both disturbance rejection and step response are Kp = 3, and Ki = 3. A high Ki tends to provide faster disturbance rejection but it leads to an important overshoot of the step response if the Kp is not high enough.

Firmware

The firmware is based on Multiwii, however the code has been written to run on numerous platforms and flight systems. In order to edit the code, first it is necessary to specify which type of multirotor is used. For this, two software packages are needed:

- **Arduino**: the development environment which allows to edit and upload the code;

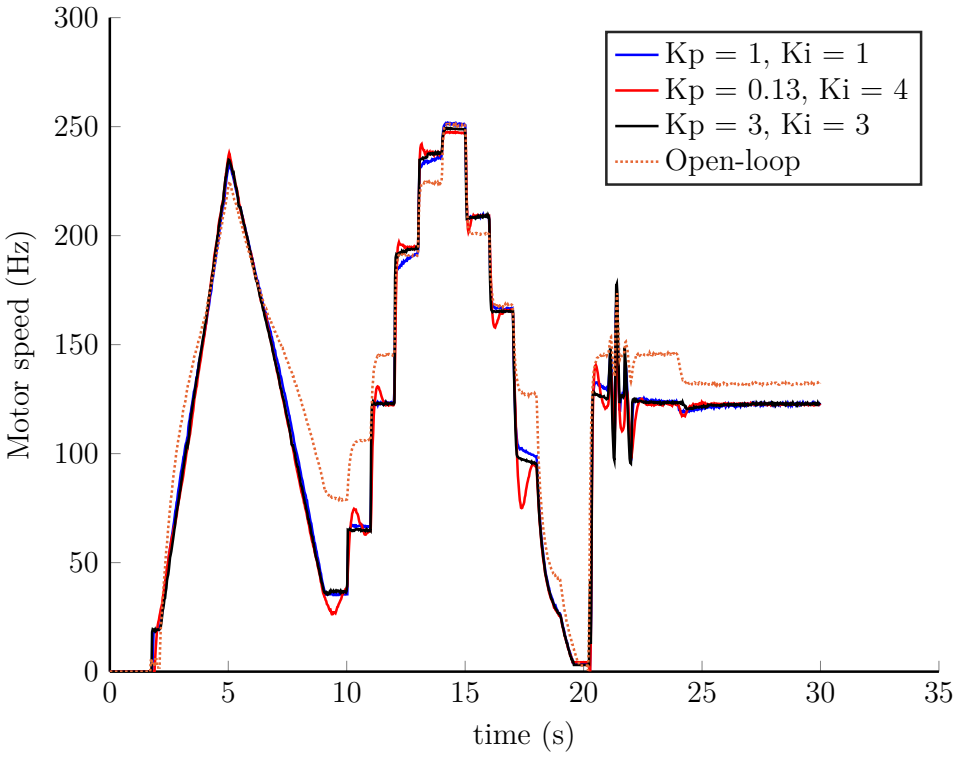
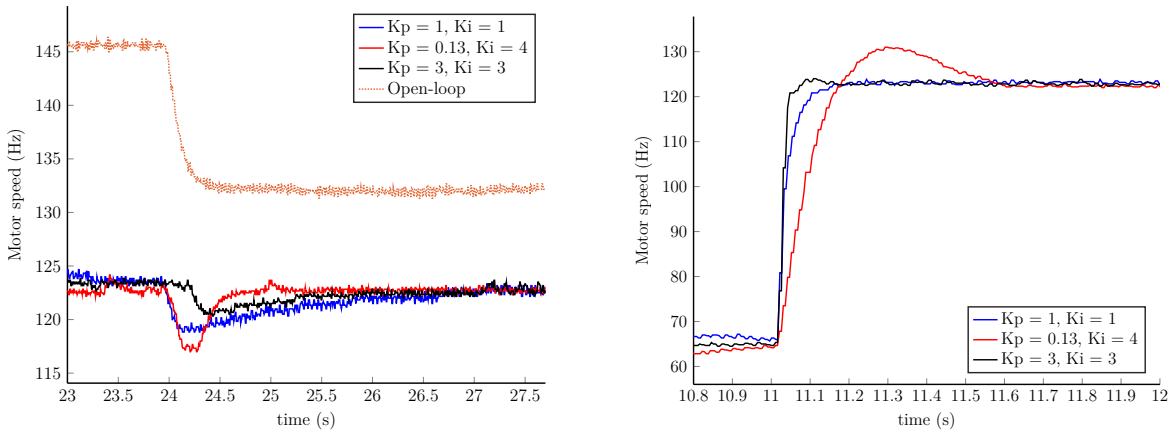


Figure 5.10 – Measured speed for different tuning parameters.



(a) Disturbance rejection. A 1 Volt drop is applied to the power supply, please note that the drop is not applied at the same time between the different experiments.

(b) Step response. At $t = 11$s the PPM signal input of the ESC goes from 1.29 ms (29 % full speed) to 1.49 ms (49 % full speed).

Figure 5.11 – Tuning of the ESC's gain.

- **Multiwii**: It includes both the open source code and the graphic user interface (GUI), necessary for the configuration of the board parameters

Multiwii GUI

To change the card's flight parameters we need the MultiWii graphical interface, which is activated running the file MULTIWIIConf.exe. Once the board is connected to a COM port of the computer, it is possible to change the parameters of the different sensors in the board (accelerometers, gyros, magnetometer, altimeter). In general, the main window of the Multiwii GUI, can be seen in Fig. 5.12

Figure 5.12 – Multiwii GUI.

5.2.4 Manipulator arm design and implementation

The manipulator arm consists on an elbow configuration with 3 degrees of freedom, all of them with revolute joints. The structure was totally designed and 3D printed at GIPSA-lab. The parameters and characteristics are presented in TABLE 5.3.

System	Description	Value	Units
Manipulator	Mass manipulator m_a	55	g
	Length 1st link l_1	5	cm
	Length 2nd link l_2	5	cm
	Length 3rd link l_3	8.4	cm

Table 5.3 – Characteristics and parameters of the manipulator.

The manipulator is controlled by two types of servomotors as actuators: the DUALSKY DS3101 digital micro servo for the first and second links and the E-flite EFLRDS35 digital super sub-micro servo for the third link and the final effector.

The DUALSKY DS3101 is a digital coreless micro servomotor, which exerts until $0.45kg/cm$ and reaches a speed of $0.10sec/60°$. Its dimensions are of $19.8 \times 24 \times 8.5mm$ and it weights $5g$. The device operates between 4.8-6 volts.

The E-flite EFLRDS35 is a digital high speed servomotor, which can exert until $0.29kg/cm$ and reaches an error precision of less than 1°. Its dimensions are of $21.8 \times$

$17.5 \times 6.4mm$ and the weight of $3.5g$. The device operates in a range of 3-5 volts. Finally, the communication between the robot manipulator and the ground station is trough a Spektrum AR6300 module receiver, which features an extremely thin and compact 2.4GHz nanolite 6-Channel Receiver with DSM2 protocol communication. The detailed design of the arm manipulator as well as the final prototype can be seen in Fig. 5.13.

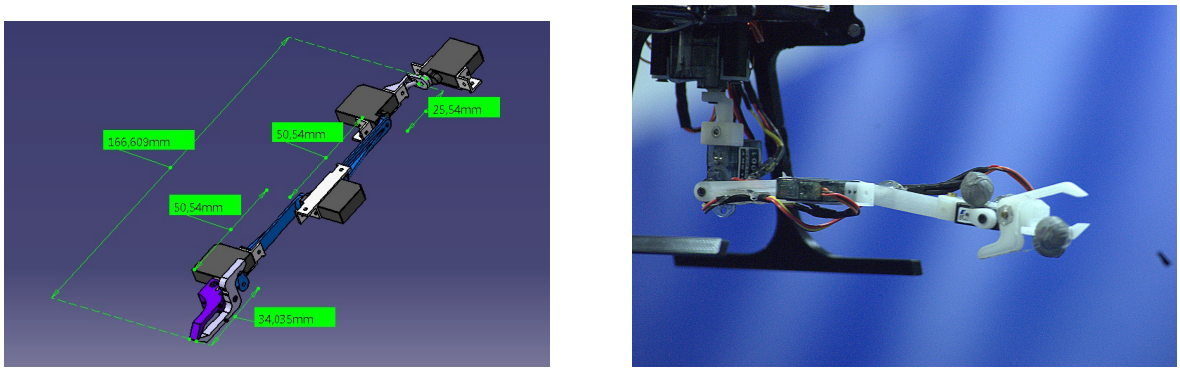


Figure 5.13 – 3D design and final prototype for the 3-DOF arm manipulator

5.3 Hardware and experimental implementation: summary

Fig. 5.14 shows a general scheme of the entire control process for the aerial system. We see the interaction between the different elements and how they operate.

In general there are three main component blocks: the aerial system, the VICON system and the ground station. The aerial system consists on the quadcopter structure, the manipulator arm, the flight controller board, the ESC's and the actuators. Here, the flight controller board computes the attitude control law from the obtained data (IMU sensor, manipulator torque values and desired Euler angles), it computes the control torques Γ_i and then they are changed to PWM signals. The ESC's receive the PWM signals and they reinterpret these data as inputs for the brushless motors and the propellers.

The VICON system performs the motion capture of the aerial system: quadrotor and end-effector of the manipulator arm. It sends the data through an UDP frame to MATLAB/Simulink.

Finally, the ground station is composed by MATLAB/Simulink and the xPC target. MATLAB/Simulink receives the position data and computes the position control for the quadrotor as well as the different elements of the manipulator torque computation. Then, the control inputs are sent through DSM2 protocol to the aerial system.

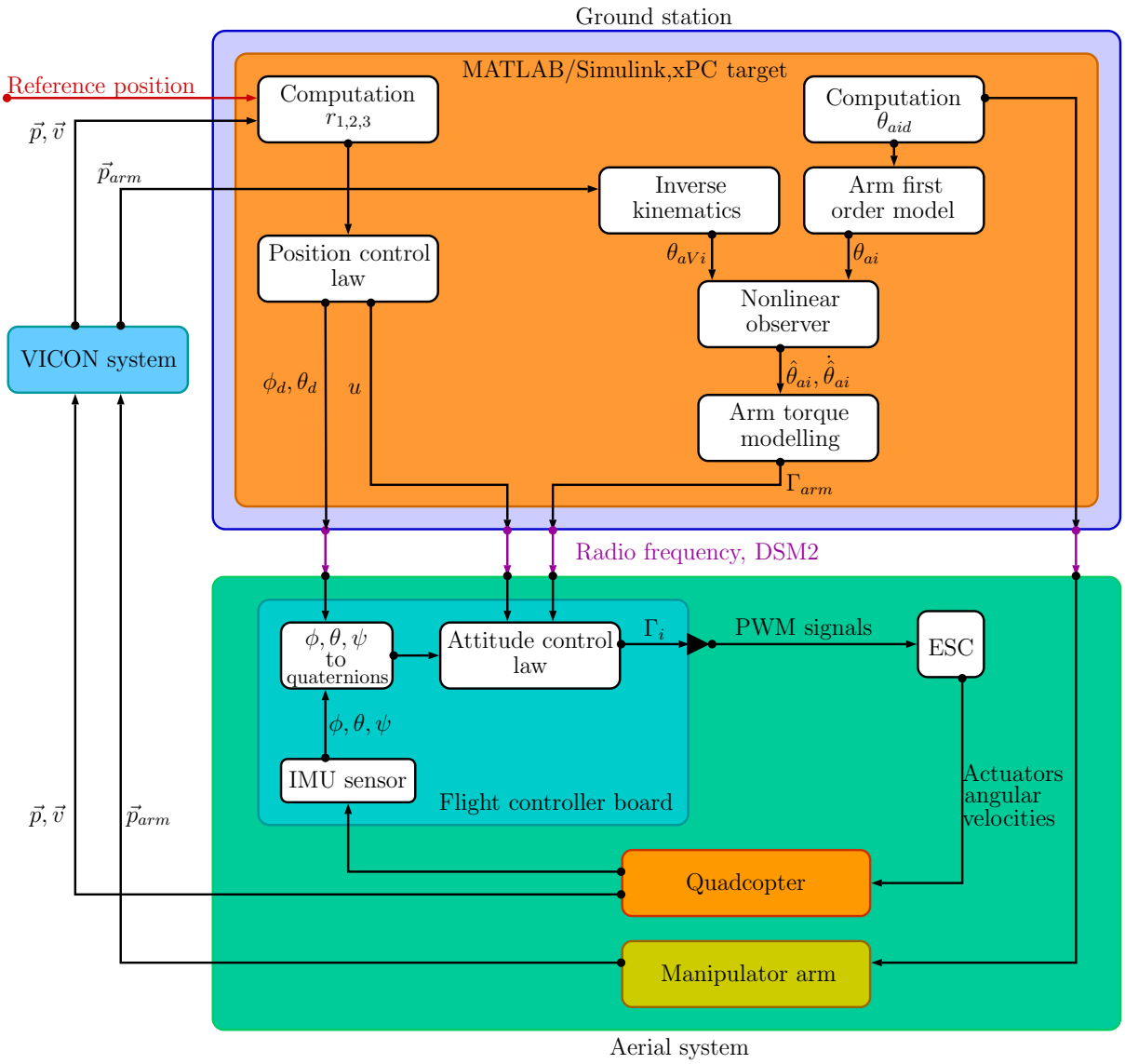


Figure 5.14 – General scheme of the hardware implementation.

5.4 Experimental results

This part of the chapter is devoted to the presentation of the experimental results and their analysis. As we know, the main objective of the project is centered on the global stabilization of a quadrotor carrying a manipulator arm and for this, a set of methods were developed and previously simulated in order to test their effectiveness. From here, it was possible to implement and test the developed approaches on the aerial platforms previously presented.

First, the experimental scenario is presented and it is based on the taking off of the aerial platform, once this one is stabilized, the arm manipulator performs a sequence of movements, after that the quadrotor lands. Then, since various methods were developed

in favor of the flight stabilization improvement, a statistical study is addressed, showing the advantages of the use of certain method.

5.4.1 Experimental scenario

An experiment was performed for each stabilization method in order to compare the performance of these ones and perform a statistical study.

In order to compute the tuning parameters for the attitude control law given in (4.6), the actuators parameters must be taken into account. For this, since the actuators are controlled by the PPM, a 1.95 ms (95 % full speed) signal input of the ESC is applied, resulting on a maximum speed of $s_{max} = 240Hz = 1400rad/s$ on the actuator. With this and using the expressions (3.7), (3.8) and (3.9) from section 3.2 and together with the actuators parameters b and k given in TABLE 5.4, the resulting maximum torques around every axis in the quadrotor frame are given by:

$$\bar{\Gamma}_\phi = 0.86 \qquad \bar{\Gamma}_\theta = 0.86 \qquad \bar{\Gamma}_\psi = 0.614 \tag{5.1}$$

Then, having $\max|\Gamma_{armi}| = 0.08Nm$ and $\delta_i = 0.1$, TABLE 5.4 presents the chosen parameters for the attitude and position control laws.

Controller	Parameter	Value
Attitude	$M_{11,21,31}$	0.1
	$M_{12,22,32}$	0.5
	$\lambda_{1,2}$	0.015
	λ_3	0.013
	$\rho_{1,2}$	10.5
	ρ_3	11
Position	$a_1 = b_1$	2.3
	c_1	1.65
	$a_2 = b_2$	1.2
	c_2	0.55
	$a_3 = b_3$	0.1
	c_3	0.015
	$\bar{r}_{1,2,3}$	5

Table 5.4 – Parameter values for the control laws.

The experimental scenario consists on two parts. First, the links of the manipulator arm are initialized at $\theta_{ai} = (0° \ 90° \ 0°)^T$ and the quadrotor is driven to the position $\vec{p}_d = (0 \ 0 \ 1)^T$, then:

- at time 20s the manipulator arm performs a first movement to reach $\theta_{ai} = (40° \ 30° \ 0°)^T$,

- at time 25s the angular positions change to $\theta_{ai} = (-30^\circ\ 70^\circ\ 25^\circ)^T$,
- at time 30s, the links angular positions are changed again to $\theta_{ai} = (10^\circ\ 20^\circ\ 35^\circ)^T$,
- finally, at time 35s, the manipulator arm performs a last movement to $\theta_{ai} = (0^\circ\ 90^\circ\ 60^\circ)^T$ and the quadrotor lands.

5.4.2 Experimental results without taking into account the manipulator torque computation

Fig. 5.15 shows the general performance of the aerial system under the disturbances coming from the manipulator arm when the arm torque estimation is not taken into account.

Specifically, the figure shows the angular positions on the manipulator and angular and linear positions of the quadcopter during the experiment. Note that even when the quaternion parametrization is considered, Euler angles given in (4.31) are used in order to have a better perspective of the behavior of the system. In this case, attitude stabilization is reached but it is visible how the movement of the manipulator causes disturbances for both, attitude and linear position, causing a non-desired displacement on the linear position.

5.4.3 Experimental results using the static manipulator torque model

Fig. 5.16 shows the angular positions of the links in the manipulator, the angular and linear positions of the aerial vehicle as well as the computed torques using the static torque model. Compared to the previous results, the manipulator torque estimation allows the compensation of the disturbance generated by the manipulator, reducing the attitude disturbances and consequently the non-desired displacement on the linear position, improving the general stabilization of the quadrotor.

5.4.4 Experimental results using the dynamic manipulator torque model

The same experiment was carried out when the control law takes into account the torque coming from the manipulator arm. However, this time Fig. 5.17 shows the behaviour of the quadcopter as well as the computed torque using the dynamic method estimation during the experiment. The angular positions of the links in the manipulator, angular and linear positions of the quadrotor and the computed torques are depicted. The precision

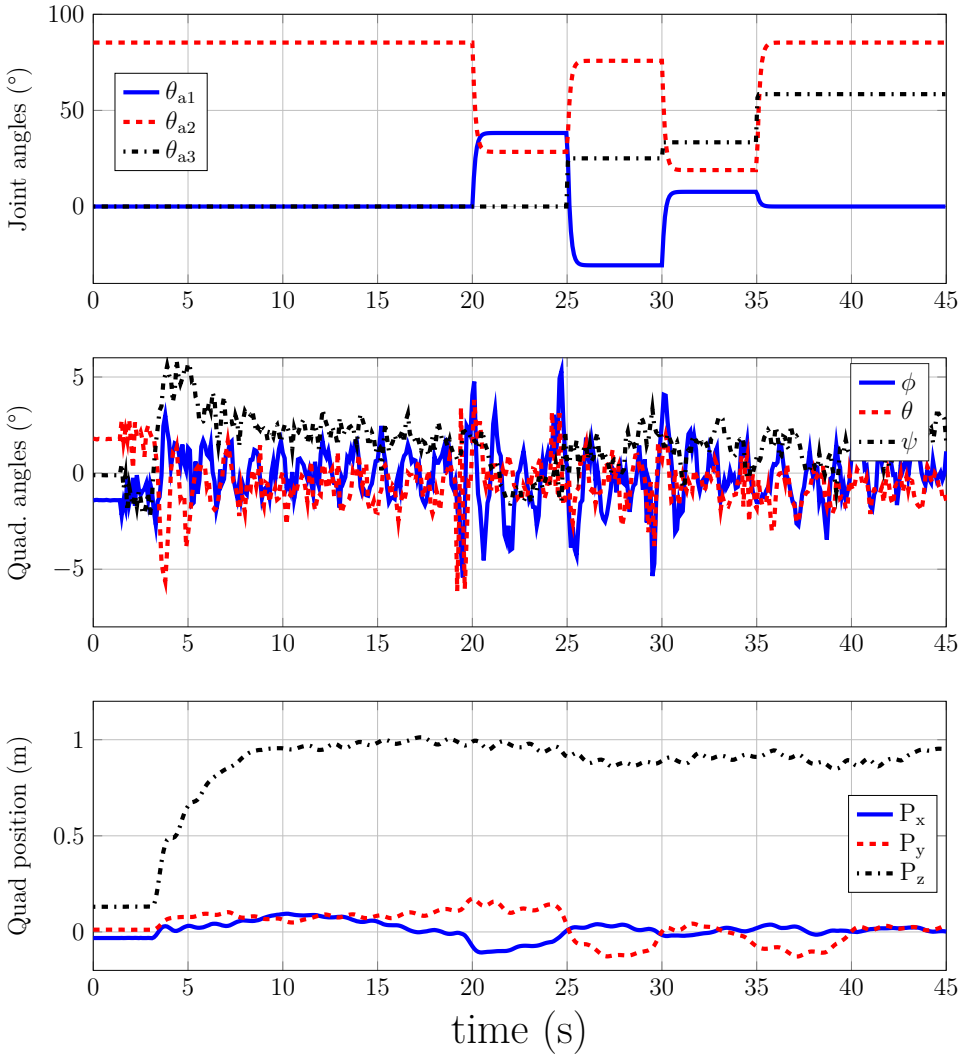


Figure 5.15 – General behavior of the system during the experiment without compensation.

improvement of the estimated torque taking into account the manipulator arm dynamics results also in general stabilization improvement of the quadrotor compared to the static estimation torque method.

5.4.5 Experimental results with dynamic manipulator torque model and the nonlinear observer

The design of the nonlinear observer for the links angular positions estimation of the manipulator arm, allows the fusion of the data coming from the first order model, given by (3.24) and the data coming from the inverse kinematics computation. The objective of the nonlinear observer is to estimate the links angular position in the arm manipulator, since

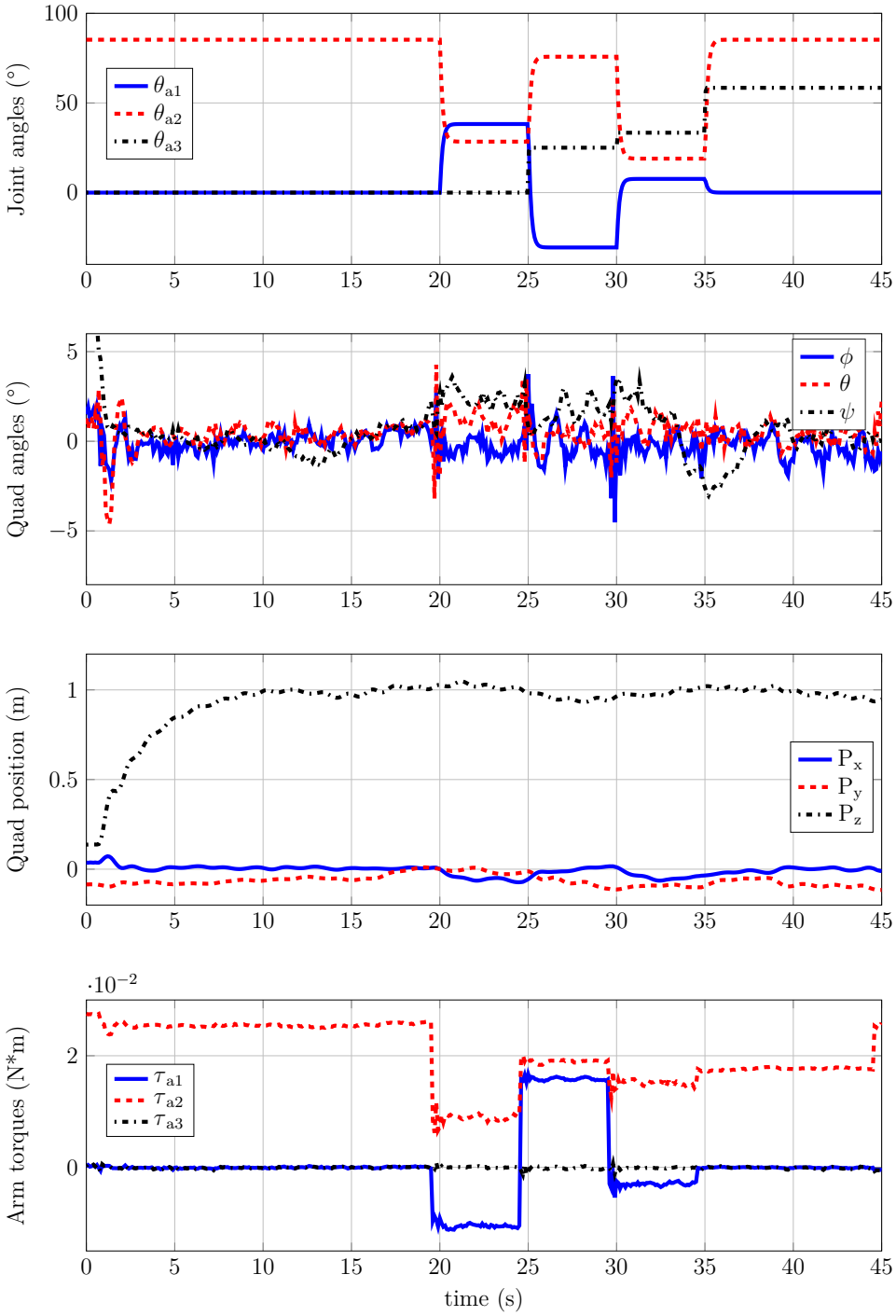


Figure 5.16 – General behavior of the system during the experiment with static estimation torque compensation.

the first order model does not fully describes the behaviour of the manipulator. Fig. 5.18 shows the angular position of the links computed by the inverse kinematics, the angular position of the links computed by the nonlinear observer, angular and linear positions of the quadrotor as well as the computed manipulator torques. In general, the results show a

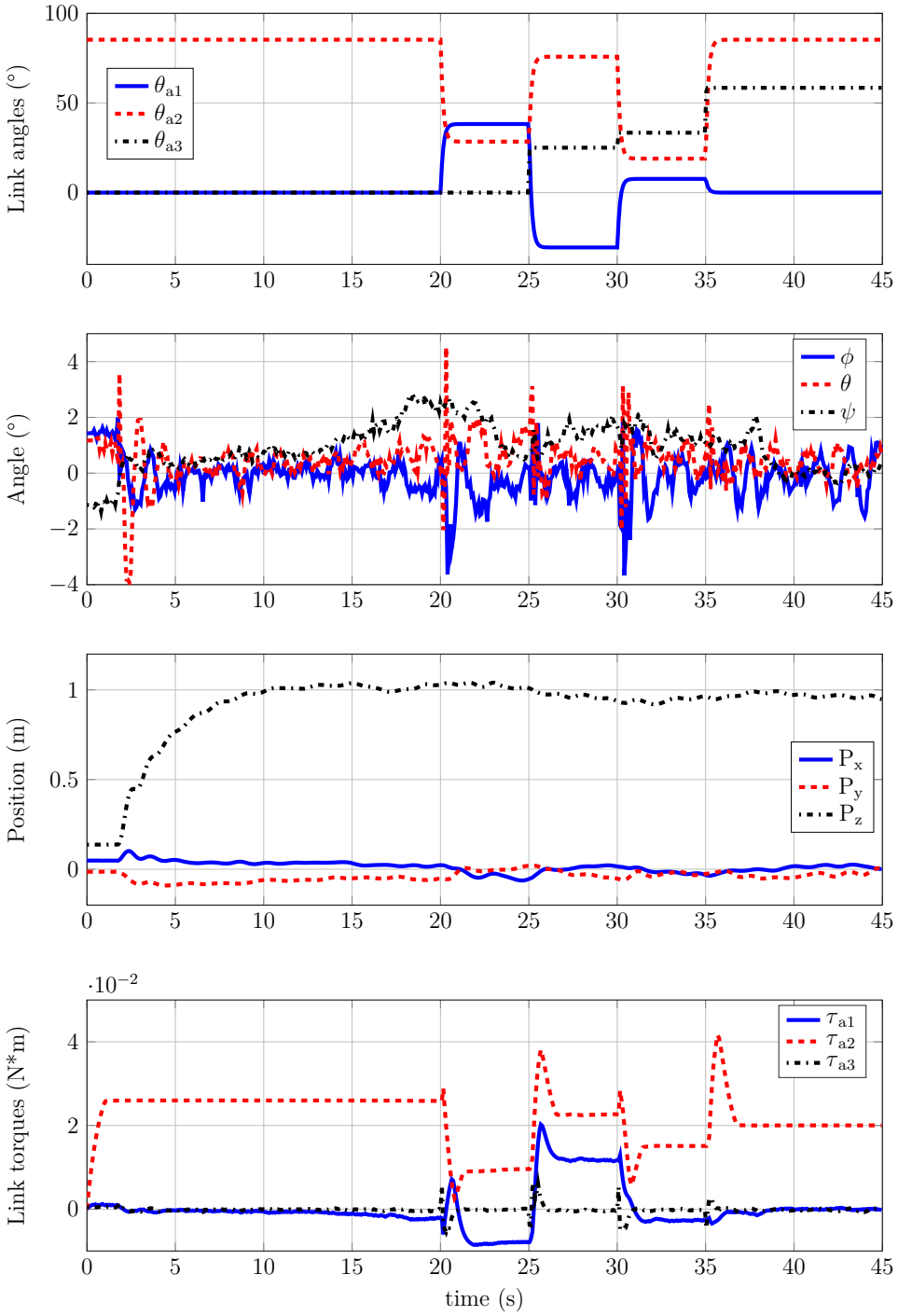


Figure 5.17 – General behaviour of the system during the experiment using dynamic method estimation torque compensation.

similar performance as that one which uses the dynamic torque estimation, however, the importance of this approach is that we have a better angular position knowledge, which guarantees a more precise torque estimation.

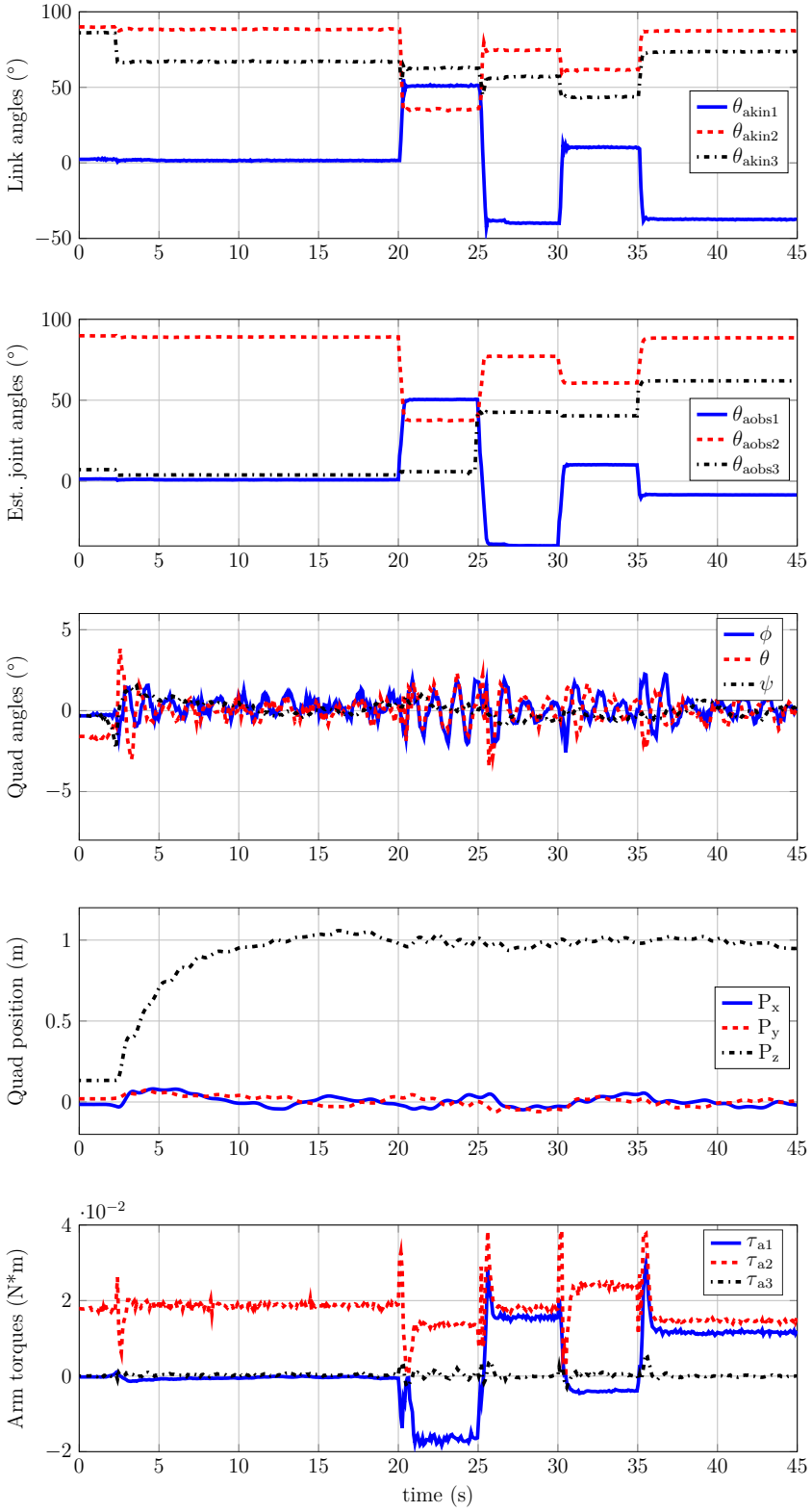


Figure 5.18 – General behaviour of the system during the experiment using dynamic method estimation torque compensation and the nonlinear observer.

5.4.6 Results analysis and statistical study

In order to test the robustness of the proposed methods and how these ones improve the flight performance of the aerial vehicle under the constant disturbances coming from the movement of the manipulator arm, a statistical study of the errors of linear position, attitude errors and the computation of the average value for the attitude error is carried out. For this, the experiments described before were performed 8 times. More experiments would improve this study, but they were limited due to time constraints.

Fig. 5.19 shows the linear position errors for the quadrotor during the different experiments. From top to bottom we see the position errors when the manipulator torque is not taken into account, the position errors when the static torque model is used, the position errors when the dynamic torque model is used and finally, the position errors when the nonlinear observer is implemented.

Equally, from top to bottom, Fig. 5.20 shows the attitude errors and the average value error, when the torque computation is not taken into account, using the static torque computation, the dynamic torque computation and that one which uses also the nonlinear observer.

In order to calculate the attitude error, $\|2 \arccos q_0\|$ was used, where $\|\cdot\|$ represents the norm and q_0 was defined before. The different error values computation allows to have another perspective of the improvement reached by each torque method estimation. Comparing the results obtained when the torque is not taken into account with those ones where the static torque is computed and applied to the system, we see that the error is reduced. Then, comparing the results of the static torque estimation with those ones which use the dynamic torque estimation and also with those ones which add the nonlinear observer yields another improvement in both, attitude and linear positions. It is clear that there is not a significant difference between the performances when the last two torque estimations are used, however, as we said before, the nonlinear observer allows a better knowledge of the manipulator arm's behaviour, if a malfunction or non-modeled behaviour is present, this estimation results in a closer real position and consequently a more precise torque estimation.

Fig. 5.21 presents the integral square error (ISE), of the linear positions for the set of experiments. From top to bottom a comparison for the x, y and z-axes is made for the different methods. Since the ISE penalizes large errors more than the small ones, the benefits due to the usage of the proposed method are more visible during the movement of the manipulator arm during the experiments.

TABLE 5.5 shows the different average error values for the set of experiments. The first column shows the attitude average value error, the second column shows the average error value for the $x - axis$, the third column for the $y - axis$ and the last column shows

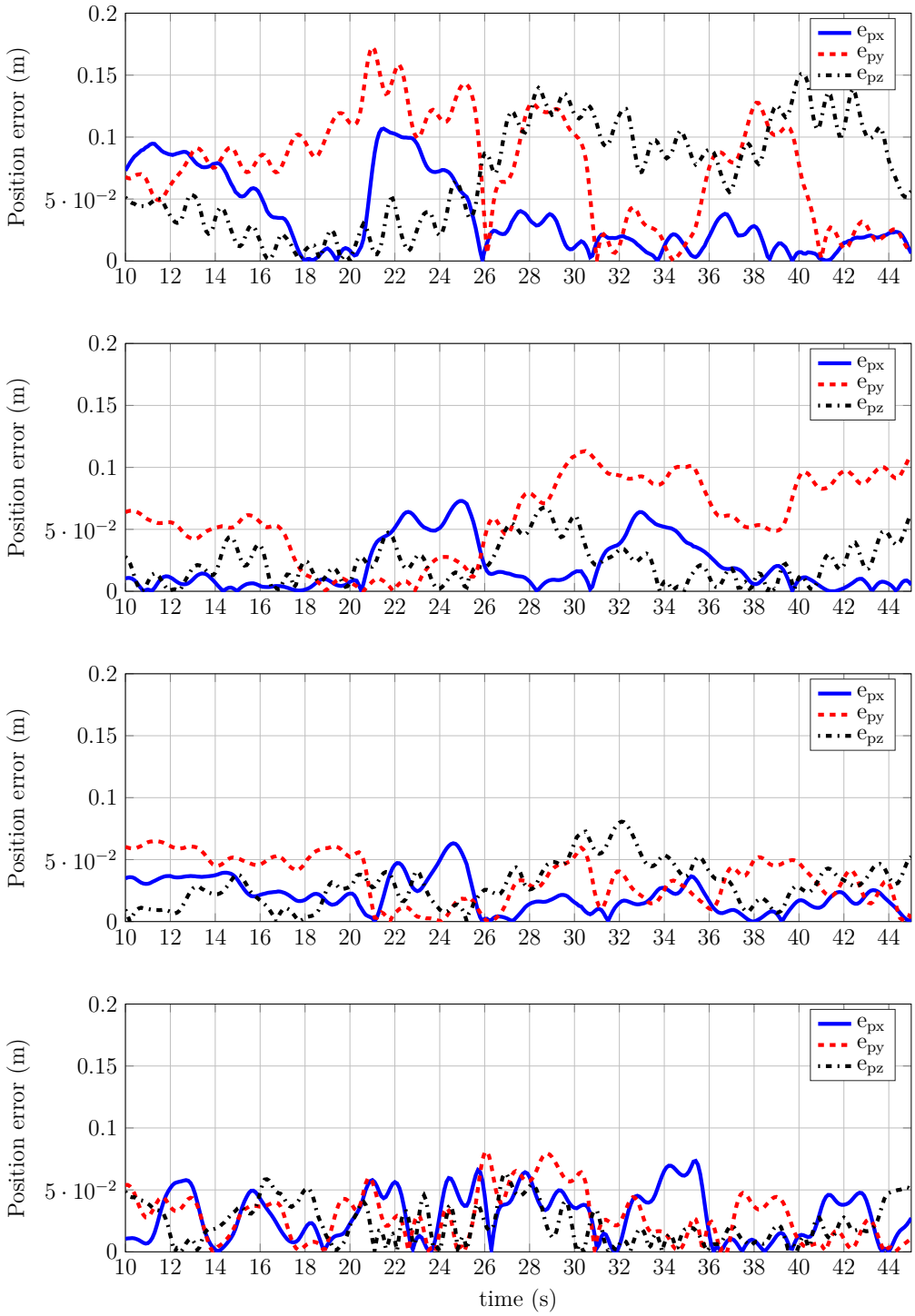


Figure 5.19 – Position error during the different experiments.

the average error value for the $z-axis$. Since the experiment was repeated 8 times using the different approaches, this TABLE gives us a better perspective of the stabilization improvement for both, quadrotor attitude and linear positions. For the first part, the torque computation was not taken into account, consequently the average error values

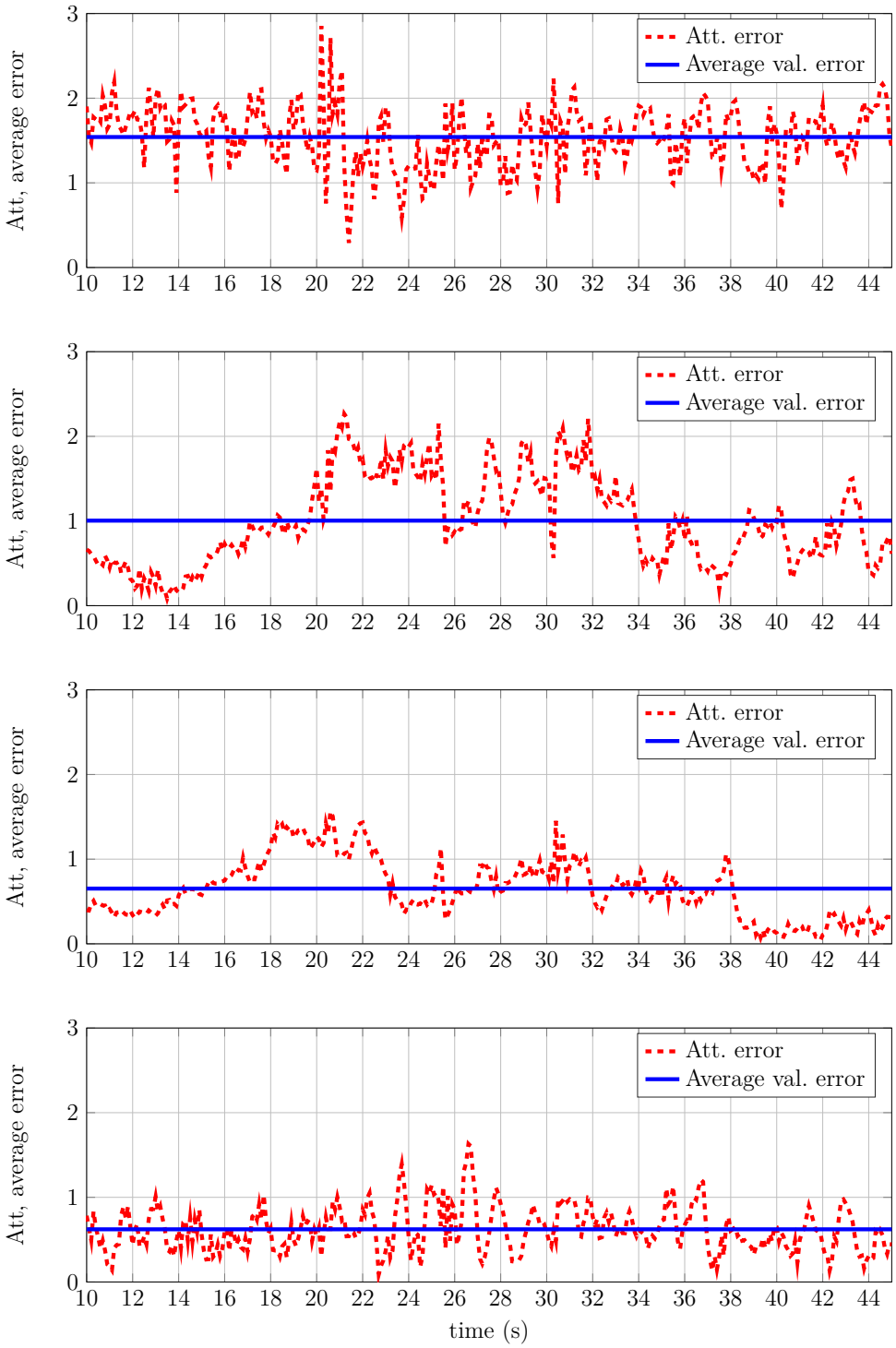


Figure 5.20 – Attitude error and attitude average error value during the different experiments.

are quite important. Then, the static torque computation is used and the average error values are reduced, showing that the stabilization has been improved. For the third and fourth parts, dynamic torque model and the nonlinear observer are implemented and the

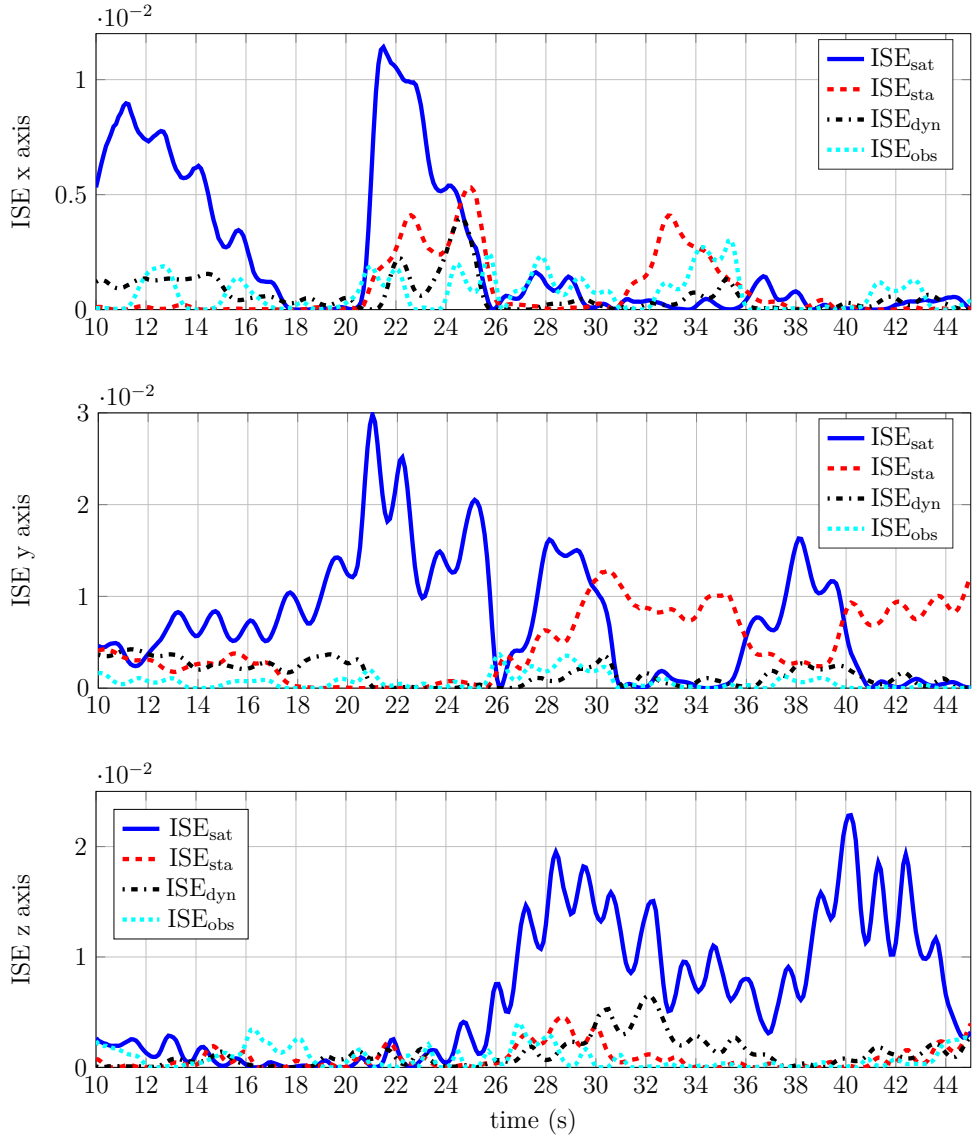


Figure 5.21 – Linear position integral square errors (ISE) for each axis during the experiments.

average value errors decrease again, showing the robustness of the proposed approach.

Fig. 5.22 presents a sequence of images with the behaviour of the system during the different experiments, where the lines intersection represents the desired position for the quadcopter. This one shows another perspective of the system stabilization.

5.4.7 Experimental results of the manipulator arm position stabilization

The experimental scenario for the manipulator arm stabilization is as follows. According to (4.41), the objective is to send the end-effector to a desired position, taking this into

Average error value	Orientation (°)	Pos x (m)	Pos y (m)	Pos z (m)
	1.5312	0.0375	0.0777	0.0468
	1.5411	0.0361	0.0762	0.0690
	1.4901	0.0341	0.0681	0.0335
Saturated control	1.4320	0.0683	0.0728	0.0552
	1.3941	0.0525	0.0679	0.0592
	1.4480	0.0507	0.0538	0.0347
	1.4175	0.0489	0.0674	0.0404
	1.4520	0.0542	0.0702	0.0532
Total average error value	1.4632	0.0477	0.0692	0.049
	1.0043	0.0201	0.0592	0.0226
	0.9621	0.0308	0.0534	0.0315
Static torque modeling	1.010	0.0248	0.0458	0.0268
	0.9033	0.0282	0.0484	0.0250
	0.9516	0.0283	0.0471	0.0379
	1.0428	0.0356	0.0404	0.0345
	0.9347	0.0378	0.0422	0.0356
	0.9457	0.0315	0.0520	0.0335
Total average error value	0.9693	0.0296	0.0485	0.0309
	0.6525	0.0209	0.0343	0.0302
	0.5777	0.0301	0.0334	0.0272
Dynamic torque modeling	0.6714	0.0284	0.0365	0.0279
	0.490	0.0297	0.0378	0.0331
	0.6584	0.0284	0.0372	0.0265
	0.7005	0.0300	0.0375	0.0273
	0.6475	0.0160	0.0333	0.0253
	0.6727	0.0251	0.0323	0.0283
Total average error value	0.6338	0.0260	0.0352	0.0282
	0.6223	0.0294	0.0283	0.0230
Dynamic torque modeling	0.6034	0.0231	0.0344	0.0239
and nonlinear observer	0.6197	0.0251	0.0323	0.0278
	0.5714	0.0214	0.0234	0.0229
	0.6524	0.0269	0.0289	0.0189
	0.6277	0.0247	0.0277	0.0245
	0.5998	0.0225	0.0239	0.0239
	0.6308	0.0279	0.0302	0.0246
Total average error value	0.6159	0.0251	0.0286	0.0236

Table 5.5 – Average error values for the experiments.

account $\vec{p}_{ad} = (0\ 0\ 1)^T$. The quadrotor searches for the best linear position in order to stay into the manipulator workspace. The experiment is carried out for 30s.

Fig. 5.23 shows the behaviour of the system during the manipulator arm position stabilization test. First, the end-effector linear position is depicted. After that, angular and linear quadrotor positions are shown. Finally, the manipulator arm torque computation

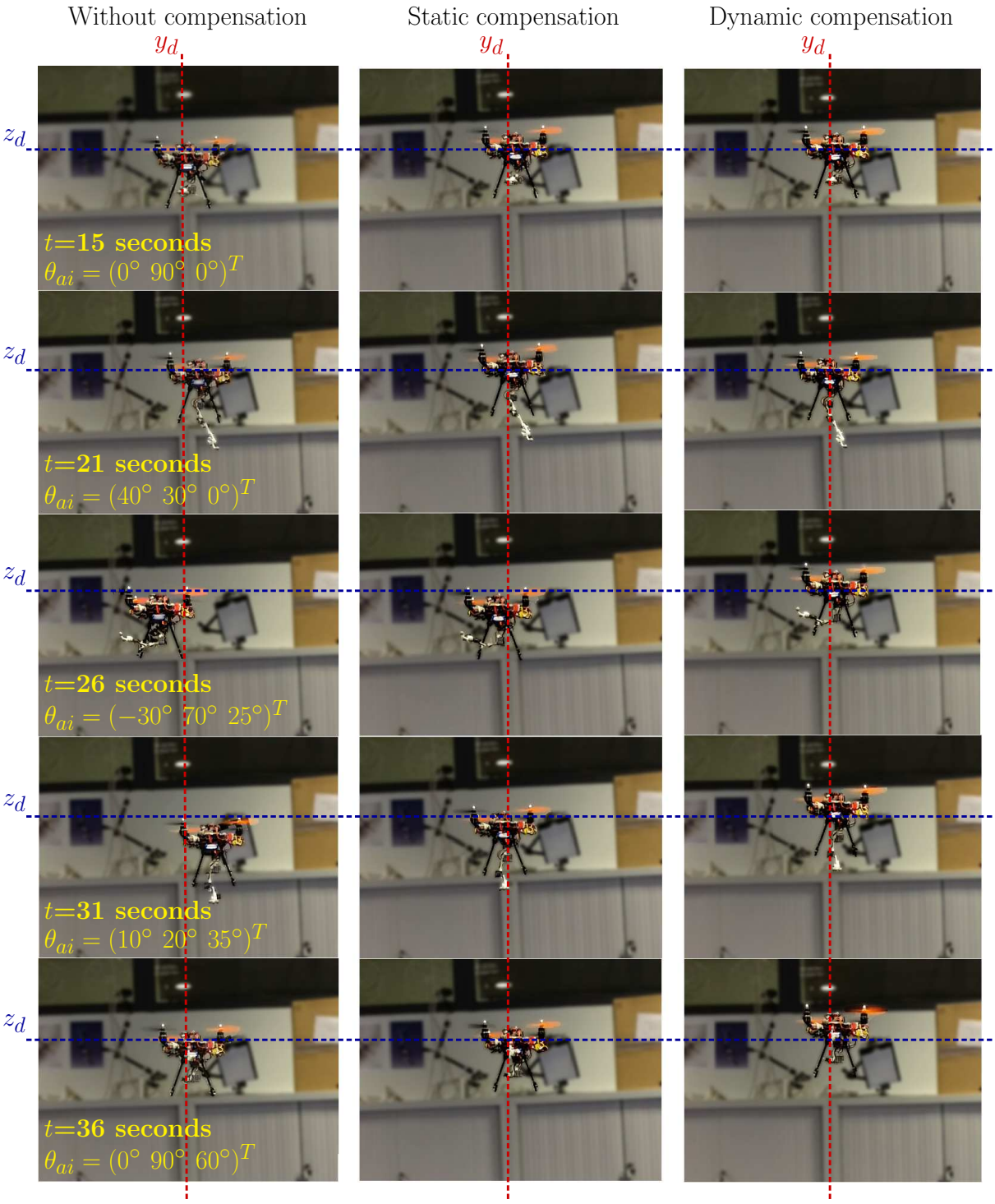


Figure 5.22 – Behaviour of the system during the experiments.

is illustrated. The inverse kinematics computation allows the stabilization of the end-effector manipulator even if the linear position of the quadrotor is changing. However, the workspace condition must be accomplished in order to generate the desired angular positions, otherwise the manipulator arm keeps the actual angular positions. Fig. 5.24

presents a sequence of images showing the arm position stabilization.

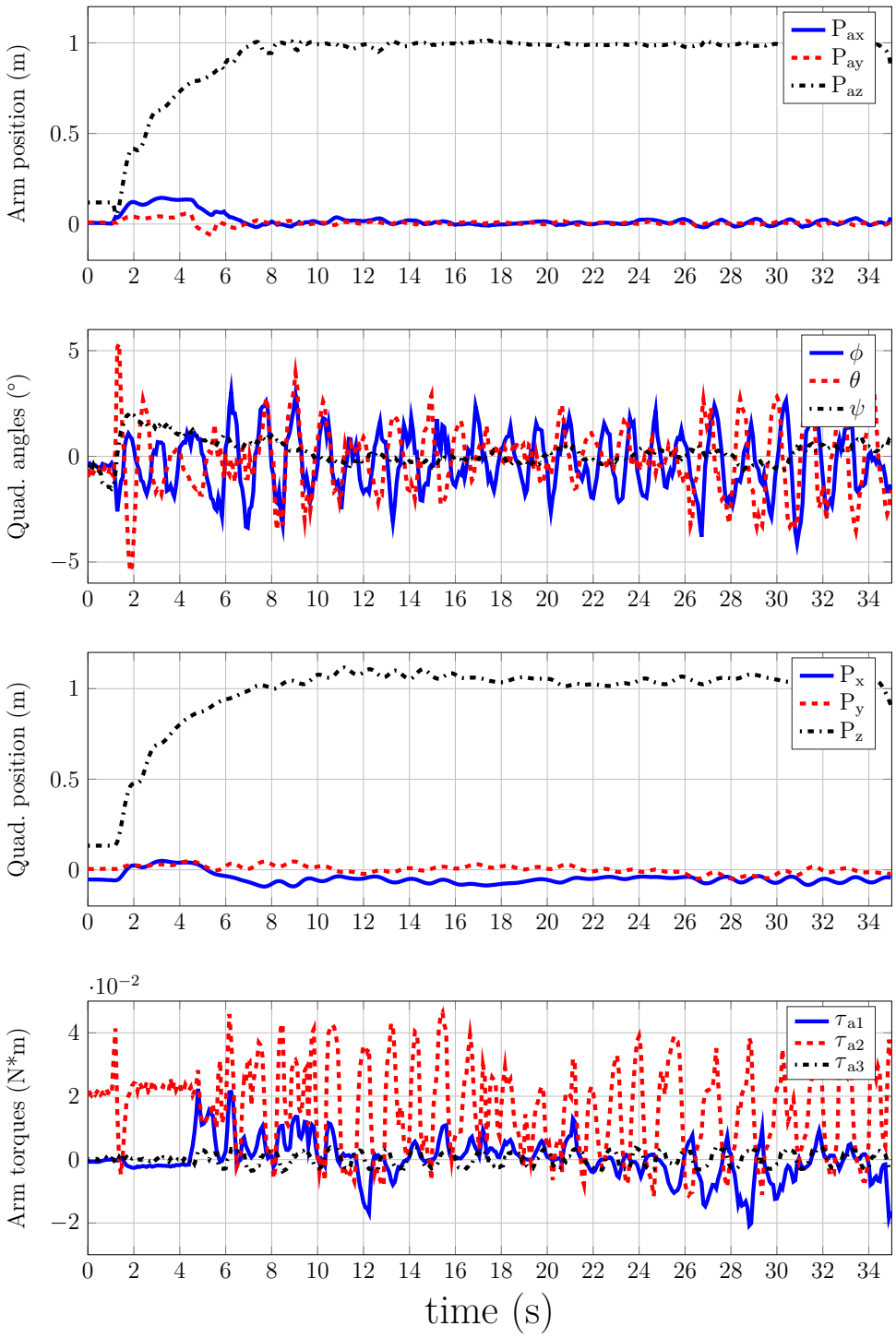


Figure 5.23 – General behaviour of the system during the end-effector manipulator position stabilization.

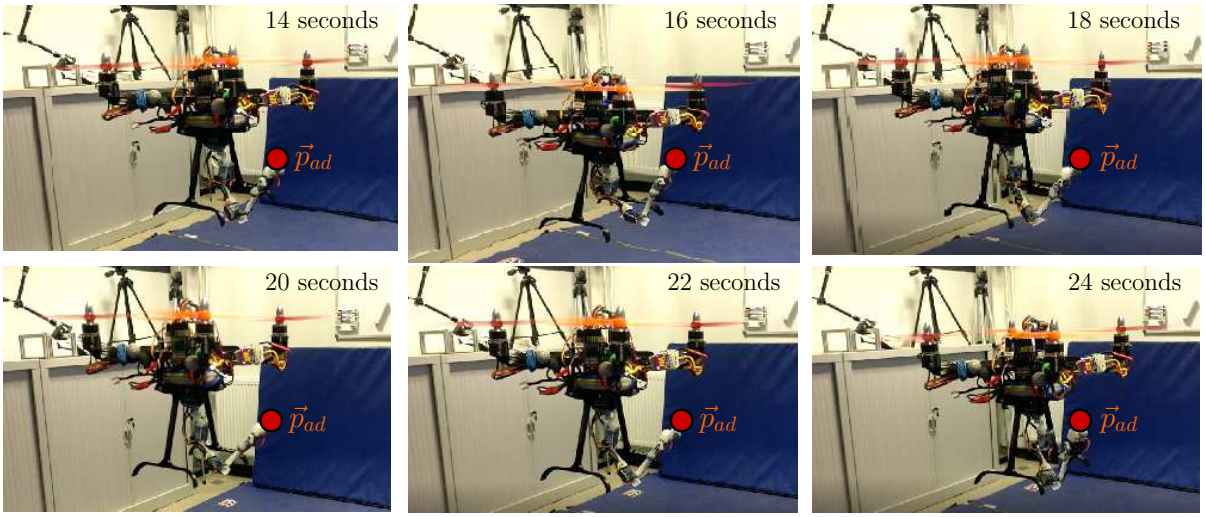


Figure 5.24 – Picture sequence of the manipulator position stabilization.

5.5 Conclusions

This chapter was devoted to the different components of the hardware setup for the project and the experimental results.

First, the MOCA room was introduced, this one is composed by a set of cameras which allow the position tracking of the aerial vehicle and the manipulator arm.

Then, the different used platforms along the project were detailed. The FLEXBOT prototypes (nano-quadcopter and hexacopter) allowed to carry out some experiments and the partial validation of the proposed algorithms. However, their reduced size and carrying capacity led us to move to other platforms.

These platforms consisted on home-made quadrotors. The structures were totally designed and 3D printed and the hardware was totally chosen for the purposes of the project. Since two frames were designed, some tests were performed in order to verify the best performance carrying the robot manipulator. The chosen frame was the one which has the medium height located actuators, because the pendulum effect with the manipulator arm is reduced. This prototype allowed the experimental validation of the proposed control laws and the manipulator arm position stabilization.

This chapter also presented the experimental results for the stabilization of the quadrotor carrying the manipulator arm. In general, an experiment was carried out many times using the different approaches. First, the case where the attitude control law does not take into account the torques coming from the robot manipulator. With this, attitude stabilization was reached, but the presence of disturbances is evident. Then, the results when the static torque is used by the attitude control law was presented. Here, there was an improvement in terms of stabilization, the disturbances were reduced and the non-

desired linear displacement was decreased. After that, the case where the dynamic torque estimation used by the attitude control law was shown. The usage of the manipulator dynamics improved the knowledge of the links angular position estimation, consequently, the manipulator torque estimation is improved and this leads to attitude and position enhancement for the quadrotor. Finally, the results using the nonlinear observer for the links angular position estimation are presented. This approach equally shows robustness and stabilization improvement compared to the precedent methods.

Then, since the experiment was carried 8 times, this allowed to perform a statistical study, where the robustness of the different methods was compared through the computation of the average error values. As a conclusion, the methods with the best performance are those ones which use the dynamic model of the arm manipulator.

In the last part of the chapter, the manipulator end-effector position stabilization was experimentally tested. The results showed that, even if the quadrotor suffered some disturbances, the end-effector remained in the same place if the aerial system stays inside the workspace manipulator.

Chapter 6
Conclusions and Future work

6.1 Conclusions

This thesis project dealt with the problem of modelling and control of a VTOL vehicle carrying a rigid manipulator, where theoretical and practical work has been developed. More specifically, the thesis project was focused on the attitude and position stabilization of a quadrotor carrying a 3-DOF manipulator arm.

The first chapter was devoted to the presentation of some mathematical preliminaries, needed for the development of the present project, as well as the state of the art concerning the robot manipulators.

Then, the second chapter presents a study of the state of the art concerning the Unmanned Aerial Vehicles, highlighting the characteristics of the VTOL vehicles and their applications. Then, the interaction between VTOLs and payloads or robot manipulators was addressed through a bibliographical study.

This study for the VTOL vehicles carrying payloads or manipulators showed that the research on this topic has increased due to the variety of possible applications, passing from passive tasks, like surveillance or monitoring, to active tasks, like construction, rescue, delivering, etc. As we did before, some approaches for the control of VTOL vehicles carrying payloads or manipulators were presented and discussed. Most of the cited works take the two systems as a unity, and they apply different methods, like the Euler-Lagrange formulation to get a mathematical model. However this adds complexity to the modelling and design of control laws.

Thanks to this bibliographical research and the analysis of the proposed methods, we realized that there is not research carried out for the stabilization of an aerial vehicle carrying a manipulator arm through the use of bounded inputs. This allowed us to position our work on the state of the art of the topic.

The third chapter introduced in a formal way the problem of attitude and position

stabilization of the aerial system under the disturbances coming from the movement of the manipulator arm. With this, it was possible to propose a novel mathematical model, where the torques exerted by the arm manipulator are considered unknown, but through some simple modelling methods it is possible to estimate the value of these variables.

In general, two methods were addressed. The first one takes into account the instantaneous links angular positions and via a proposed algorithm it is possible to compute the generated torques. Then, the second case takes into account the dynamics of the manipulator arm; to do this, the manipulator actuators were modeled as first order systems and their correspondent response was used by the manipulator dynamic model. In other words, from the knowledge of the position and velocity of the different links, the manipulator torque is modeled. Contrary to the cited works, our estimation methods are quite simple and the general model can be extended to any multi-rotor carrying a n-DOF articulated manipulator.

Chapter four presents the proposed attitude control law, which consists on nested saturated functions that take into account the estimation of the manipulator torques. The objective of the control law is to drive the VTOL vehicle to attitude stabilization and even more, guarantee this one under the disturbances coming from the robot manipulator. Another big benefit of the proposed control law is the usage of saturation functions, which provide only feasible control signals to the vehicle actuators. However, this characteristic is present if the condition related the manipulator design and the control law is preserved.

Once the attitude problem was solved, a position control law was proposed in order to send the quadrotor to a desired linear position and keep it there avoiding the effects of the movement of the robot manipulator. The main characteristics of our approach are its generic nature and its simplicity (it can be implemented on any platform, in a simple way).

After that, in order to give more robustness to the proposed approach, a nonlinear observer was designed. The objective of this algorithm was to estimate the links angular position in the robot manipulator, since these ones operate in open-loop. To do this the problem of inverse kinematics was tackled in order to obtain more information about the links angular position. Then, to implement the nonlinear observer, the data coming from the first order model actuators and the inverse kinematics were fused. Finally, the output of the nonlinear observer is sent to the manipulator dynamic model.

A first approach for aerial manipulation was also proposed. The objective was to send the manipulator end-effector to a desired position. For this, considering that the quadrotor problem stabilization was solved, it was possible to send the aerial vehicle into the manipulator workspace; there, the inverse kinematics computes the desired links angular position to send the end-effector to the target point.

All these approaches were validated via simulation, showing good attitude and position performance.

The fifth chapter was devoted to the presentation of the hardware setup, used experimental platforms and MOCA room, necessary to know the actual attitude and position of the systems and to do the correct implementation of the proposed methods. Then, a set of experiments show the performance and robustness of our algorithms, where we can remark that each method outperforms the precedent. In conclusion, the torque estimation methods which use the dynamics of the manipulator arm generate the best attitude and position stabilization for the aerial system.

To know more about the robustness of the different approaches, a statistical study was performed. This one allowed the comparison of the obtained results. In general, the asymptotical stabilization of the aerial vehicle carrying the manipulator arm was accomplished and improved with each torque estimation.

In general, the aerial manipulation sets a new standard for mobile robotics, since aerial vehicles can not only access to unavailable areas but also interact with the environment, making these systems big aids for a variety of applications. Then, the robotics community has proposed many ways to face the problem giving solutions to some tasks and challenges, but leaving some others still without answer. In this way, the present work can be extended in many directions.

6.2 Future work

- Concerning the actual experimental prototype, the material used for the construction of the manipulator arm, the resin, is very fragile and it brakes easily. Then, a new prototype based on carbon fiber or titanium should be envisaged, in order to have a more resistant and rigid manipulator arm.

- In general, three methods were proposed for the stabilization of the aerial vehicle carrying the manipulator arm: the static torque model, the dynamic torque model and the dynamic torque model aided by the Luenberger observer. However, not all the dynamics for the entire system are taken into account. Then, the proposal of a dynamic model is envisaged and the idea of the usage of a saturated controller keeps present, since it was shown in the state of the art the novelty of the proposed method. For this, the active disturbance rejection concept can be studied. The basic idea is to estimate the disturbances coming from the manipulator arm by an extended state observer, and then introduce the saturated nonlinear controller to actively compensate the uncertainties in real-time.

- Regarding the aerial manipulation, many tasks can be achieved, like retrieve and delivery, objects manipulation, etc., where a first approach using the manipulator inverse kinematics for position stabilization has been addressed. However, the general performance of the end-effector can be improved by the design of another algorithm, like in Kim et al. [2013], Korpela et al. [2014] or Thomas et al. [2014], where different approaches are proposed in order to perform end-effector position tracking, collision avoidance or stabilization. Also, the dependence of the VICON system will disappear, since this one will not be used to know the end-effector position.

- Finally, since the carrying capacity of the VTOL vehicles is reduced, the concept of multi-UAV formation control for grasping and transportation can also be considered as an extension of the present work. For this purpose, novel control laws, objects detection and navigation strategies, like the graph theory will be objects of study.

Appendix A
Generated publications

1. J.U. Alvarez-Muñoz, N. Marchand, F. Guerrero-Castellanos, S. Durand and A.E. López-Luna, "Improving control of quadrotors carrying a manipulator arm", Proc. of the XVI Congreso Latinoamericano de Control Automático (CLCA 2014), 2014.

2. J.U. Alvarez-Muñoz, N. Marchand, J.F. Guerrero Castellanos, A.E. López-Luna, J.J. Téllez-Guzmán, J. Colmenares-Vazquez, S. Durand, J. Dumon and G. Hasan, "Non-linear control of a nano-hexacopter carrying a manipulator arm", Proc. IEEE/RJS Int. Conf. on Intelligent Robots and Systems (IROS), 2015.

3. J. U. Alvarez-Muñoz, J. F. Guerrero-Castellanos, N. Marchand1, J. J. Téllez-Guzmán and S. Durand, "Modelling and control of a VTOL vehicle carrying a rigid manipulator", RA-L and ICRA Special Issue on Aerial Manipulation for the IEEE Robotics and Automation Letters (RA-L), 2017 (submitted).

Appendix B
Experimental platforms

B.1 Flexbot

Flexbot comes from a crowdfunding project dedicated to offer open source hardware multicopters controllable with a smartphone. Fig. B.1 shows the flexbot hexarotor and quadrotor platforms.

Figure B.1 – Flexbot platforms, hexarotor and quadrotor.

Hardware

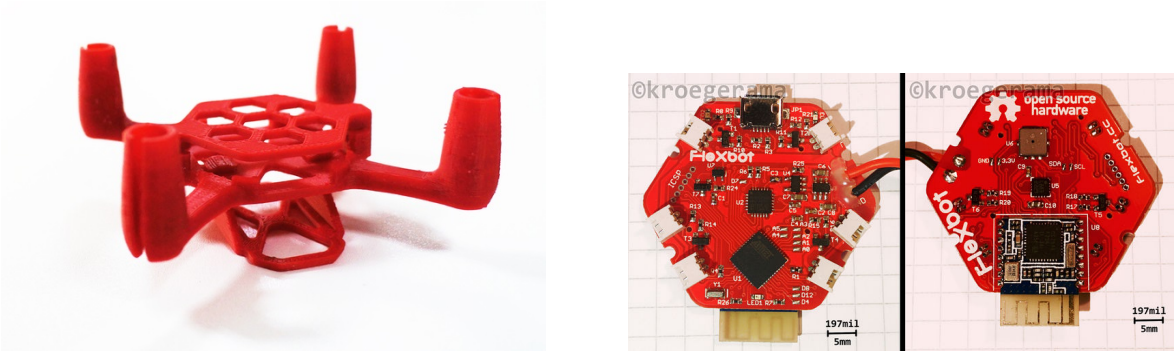
Some of the main elements of these platforms are presented below. We present a brief introduction of the frame, flight controller board and communication protocol.

Frame

The frame is 3D printed and the CAD model is open source. The frame of the quadcopter is shown on Fig. B.2a

Flight controller board

The flight controller board allows the implementation of the attitude control law, since the on-board software is equally open source. The board features an ATMega32u4 processor, an IMU sensor (MPU6050), a magnetometer (HMCL5883L) and a barometer (BMP085). The motors are directly connected to the card as some transistors are mounted to run the DC motors. The board can be programmed with arduino, and the default firmware is based on MultiWii. A bluetooth low energy module is also mounted and communicate through a serial bus with the processor. Fig. B.2b shows the flight controller board.



(a) 3D printed frame.

(b) Flight controller board.

Figure B.2 – 3D printed frame of the flexbot quadrotor and the flight controller board.

Communication

An application for smartphone is provided to control the multicopter with bluetooth protocol. The data received by the flight controller board is organized with *MultiWii Serial Protocol* (MSP). In order to communicate between the ground station and the platform, a specific board had to be developed. It consists on an arduino MEGA, an Ethernet shield and a bluetooth shield. The arduino board is connected to the same local network as the ground station which packs the commands to fulfill the MSP and sends it through UDP to the IP adress of the arduino board at a specific port. The arduino board is running an UDP server which listens to the communication, pairs the bluetooth device and relays the data received by UDP to the Flexbot's bluetooth module.

This solution has been found to be reliable, but was only possible because the bluetooth module of the flexbot and the arduino shield were both the same. However, Flexbot changed the bluetooth module after the first batch, the bluetooth shield was then not able to be paired with the flight controller board. To overcome this issue, a Raspberry Pi and a bluetooth dongle have been used. The communication protocol had to be reverse engined, a python script has been developed to run an UDP server, and to send the data through bluetooth protocol.

B.2 Other experimental platforms at GIPSA

Currently there are three other projects that gave way to the usage of other experimental platforms.

The main objective of the first project was the development of a control law for a hybrid vehicle, which allows to the system fly and move on the floor automatically. It was developed by Josue Colmenares. Two prototypes were used during the development of this project. The first one consisted on the FLEXBOT nano-hexacopter, see section B.1. The second one consisted on a hybrid vehicle called the "Wagon" hybrid vehicle, see Fig. B.3. As in our case, the frame was designed and 3D printed in GIPSA-lab.

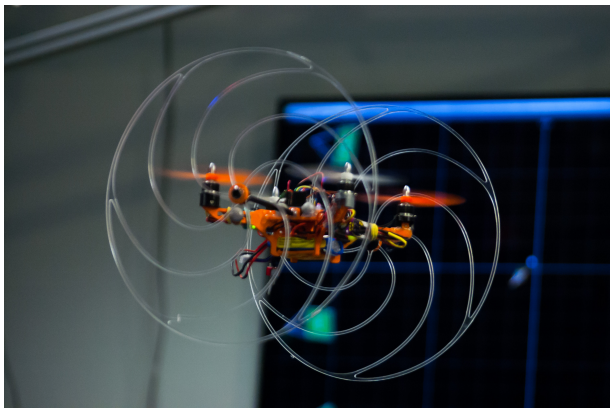
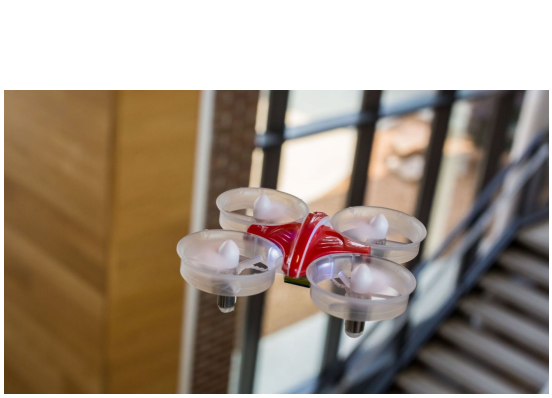


Figure B.3 – Wagon hybrid vehicle in flight

The objective of the second project was the design of an event-based control for micro biomimetic robots. The development of this work was in charge of Bruno Boisseau. Many experimental prototypes were tested during the development of the project, including two nano-quadrotors called "Inductrix" and "Nano-QX" both from BLADE, see Fig. B.4. BLADE is a company subsidiary of Horizon Hobby, dedicated to the design and construction of mini UAVs.

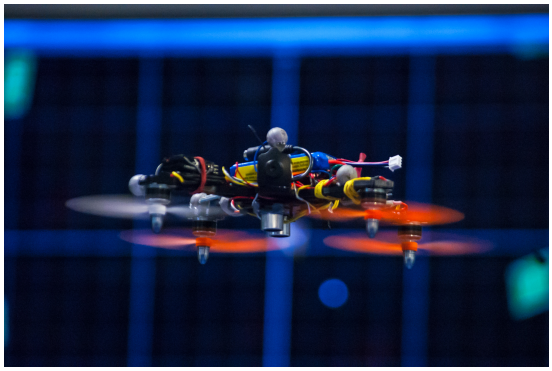
The third project consists on the tele-operation of an aerial vehicle aided by a vision algorithm in a corridor. The development of this project is in charge of Jose Juan Tellez. Two prototypes have been mainly used: the first one consists on a mini quadrotor, called the "Mosca". The frame was designed and 3D printed in GIPSA-lab. The prototype mounts a camera and has its rotors in reverse. The second one consists also on a BLADE mini-quadcopter, the "Inductrix 200", see Fig. B.5.



(a)

(b)

Figure B.4 – (a)Inductrix nano-quadcopter and (b)Nano-QX nano-quadrotor in flight



(a)

(b)

Figure B.5 – (a)Mosca mini-quadcopter and (b)Inductrix 200 mini-quadrotor in flight

Bibliography

ABB. Abb group - leading digital technologies for industry, November 2016. URL `http://new.abb.com/products/robotics/industrial-robots/irb-910sc`.

A. Alaimo, V. Artale, C. Milazzo, A. Ricciardello, and L. Trefiletti. Mathematical modeling and control of a hexacopter. In *2013 International Conference on Unmanned Aircraft Systems (ICUAS)*, pages 1043–1050, May 2013.

C. A. Arellano-Muro, L. F. Luque-Vega, B. Castillo-Toledo, and A. G. Loukianov. Backstepping control with sliding mode estimation for a hexacopter. In *2013 10th International Conference on Electrical Engineering, Computing Science and Automatic Control (CCE)*, pages 31–36, Sept 2013.

D. S. Bernstein and A. N. Michel. A chronological bibliography on saturating actuators. *International Journal of robust and nonlinear control*, 5(5):375–380, 1995.

E. Bora and A. Erdinc. Modeling and pd control of a quadrotor vtol vehicle. In *2007 IEEE Intelligent Vehicles Symposium*, pages 894–899. IEEE, 2007.

S. Bouabdallah and R. Sierwart. Backstepping and sliding-mode techniques applied to an indoor micro quadrotor. In *IEEE International Conference on Intelligent Robot and Automation*, 2005.

P. Castillo, A. Dzul, and R. Lozano. Real-time stabilization and tracking of a four rotor mini rotorcraft. In *IEEE Transactions on Control Systems Technology*, volume 12, pages 510–516, 2004.

Codian-Robotics. Td pick and place robots, November 2016. URL `http://www.codian-robotics.com/en/td-robots/`.

J. Courbon, Y. Mezouar, N. Guenard, and P. Martinet. Visual navigation of a quadrotor aerial vehicle. In *Intelligent Robots and Systems, 2009. IROS 2009. IEEE/RSJ International Conference on*, pages 5315–5320. IEEE, 2009.

R. Cruz-José, J. F. Guerrero-Castellanos, W. F. Guerrero-Sánchez, and J. J. Oliveros-Oliveros. Estabilización global de mini naves aéreas tipo vtol. In *Congreso Nacional de Control Automático*, Campeche, México, 2012.

W. E. Dixon, M. S. Queiroz, Zhang F., and D. M. Dawson. Tracking control of robot manipulators with bounded torque inputs. *Robotica*, 17:121–129, 1999.

DJI. The future of possible, February 2017. URL `http://www.dji.com`.

J. Fogelberg. *Navigation and Autonomous Control of a Hexacopter in Indoor Environments*. PhD thesis, Sweden Lund University, 2013.

T. I. Fossen. *Guidance and Control of Ocean Vehicles*. John Wiley and Sons, New York, 1994.

V. Ghadiok, J. Goldin, and W. Ren. Autonomous indoor aerial gripping using a quadrotor. In *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 4645–4651, 2011.

V. Ghadiok, J. Goldin, and W. Ren. On the design and development of attitude stabilization, vision-based navigation, and aerial gripping for a low-cost quadrotor. *Autonomous Robots*, 33:41–68, 2012.

J. F. Guerrero-Castellanos. *Estimation de l'attitude et commande en attitude d'un corps rigide : application à un mini hélicoptère à quatre rotors*. PhD thesis, Université Joseph Fourier-Grenoble I, January 2008.

J. F. Guerrero-Castellanos, N. Marchand, S. Lesecq, and J. Delamare. Bounded attitude stabilization: Real-time application on four-rotor mini-helicopter. In *17th IFAC World Congress*, Seoul, Korea, 2008.

J. F. Guerrero-Castellanos, N. Marchand, A. Hably, S. Lesecq, and J. Delamare. Bounded attitude control of rigid bodies: Real-time experimentation to a quadrotor mini-helicopter. *Control Engineering Practice*, 19:790–797, 2011.

G. Heredia, A. E. Jimenez-Cano, I. Sanchez, D. Llorente, V. Vega, J. Braga, J. A. Acosta, and A. Ollero. Control of a multirotor outdoor aerial manipulator. In *Intelligent Robots and Systems (IROS 2014), 2014 IEEE/RSJ International Conference on*, pages 3417–3422. IEEE, 2014.

G. M. Hoffmann, H. Huang, S. L. Waslander, and C. J. Tomlin. Quadrotor helicopter flight dynamics and control: Theory and experiment. In *Proc. of the AIAA Guidance, Navigation, and Control Conference*, volume 2, page 4, 2007.

A.E. Jimenez-Cano, J. Martin, G. Heredia, A. Ollero, and R. Cano. Control of an aerial robot with multi-link arm for assembly tasks. In *2013 IEEE International Conference on Robotics and Automation*, pages 4916–4921, 2013.

E. N. Johnson and S. K. Kannan. Nested saturation with guaranteed real poles. In *American Control Conference, 2003. Proceedings of the 2003*, volume 1, pages 497–502. IEEE, 2003.

S. M. Joshi, A. G. Kelkar, and J. T. Wen. Robust attitude stabilization of spacecraft using nonlinear quaternion feedback. In *IEEE Transactions on Automatic Control*, volume 40, pages 1800–1803, 1995.

L. Jun and L. Yuntang. Dynamic analysis and pid control for a quadrotor. In *2011 IEEE International Conference on Mechatronics and Automation*, pages 573–578. IEEE, 2011.

F. Kendoul, I. Fantoni, and R. Lozano. Modeling and control of a small aircraft having two tilting rotors. In *44th IEEE conference on decision and control and European control conference*, 2005.

A. Khalifa, M. Fanni, A. Ramadan, and A. Abo-Ismail. Modeling and control of a new quadrotor manipulation system. In *2012 First International Conference on Innovative Enginerring Systems (ICIES)*, pages 109–114, 2012.

O. Khatib. A unified approach for motion and force control of robot manipulators: The operational space formulation. *IEEE Journal on Robotics and Automation*, 3(1):43–53, February 1987. ISSN 0882-4967. doi: 10.1109/JRA.1987.1087068.

S. Kim, S. Choi, and H. J. Kim. Aerial manipulation using a quadrotor with a two dof robotic arm. In *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 4990–4995, Nov 2013. doi: 10.1109/IROS.2013.6697077.

K. Kondak, F. Huber, M. Schwarzbach, M. Laiacker, D. Sommer, M. Bejar, and A. Ollero. Aerial manipulation robot composed of an autonomous helicopter and a 7 degrees of freedom industrial manipulator. In *Robotics and Automation (ICRA), 2014 IEEE International Conference on*, pages 2107–2112. IEEE, 2014.

C. Korpela, M. Orsag, and P. Oh. Towards valve turning using a dual-arm aerial manipulator. In *Intelligent Robots and Systems (IROS 2014), 2014 IEEE/RSJ International Conference on*, pages 3411–3416. IEEE, 2014.

KUKA. Robotics and Automation, November 2016. URL `http://www.kuka-robotics.com/en/`.

F. L. Lewis, D. M. Dawson, and Abdallah C. T. *Robot Manipulator Control: Theory and Practice*. Marcel Dekker, Inc., 2004.

T. Li, Y. Zhang, and B. W. Gordon. Passive and active nonlinear fault-tolerant control of a quadrotor unmanned aerial vehicle based on the sliding mode control technique. *Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering*, 227(1):12–23, 2013.

V. Lipiello and F Ruggiero. Cartesian impedance control of a uav with a robotic arm. In *10th IFAC Symposium on Robot Control*, 2012a.

V. Lipiello and F 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*, pages 3768–3773, 2012b.

T. Madani and A. Benallegue. Control of a quadrotor mini-helicopter via full state backstepping technique. In *45th IEEE Conference on Decision and Control*, pages 1515–1520, 2006.

N. Marchand. Further results on global stabilization for multiple integrators with bounded controls. In *Decision and Control, 2003. Proceedings. 42nd IEEE Conference on*, volume 5, pages 4440–4444. IEEE, 2003.

N. Marchand and A. Hably. Global stabilization of multiple integrators with bounded controls. *Automatica*, 41(12):2147–2152, 2005.

D. Mellinger, M. Shomin, and V. Kumar. Cooperative grasping and transport using multiple quadrotors. In *International Symposium on Distributed Autonomous Systems*, Lausanne, Switzerland, 2010.

N. Michael, J. Fink, and Kumar V. Cooperative manipulation and transportation with aerial robots. *Autonomous Robots*, 30:73–86, 2011.

M. Mohammadi, A. Franchi, D. Barcelli, and D. Prattichizzo. Cooperative aerial telemanipulation with haptic feedback. In *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 5092–5098, Oct 2016.

R.M. Murray, Z. Li, S.S. Sastry, and S.S. Sastry. *A Mathematical Introduction to Robotic Manipulation*. Taylor & Francis, 1994. ISBN 9780849379819. URL `https://books.google.fr/books?id=D_PqGKRo7oIC`.

R. C. Nelson. *Flight Stability and Automatic Control*, volume second edition. McGraw-Hill, Boston, 1998.

M. Orsag, C. Korpela, S. Bogdan, and P. Oh. Lyapunov based model reference adaptive control for aerial manipulation. In *2013 International Conference on Unmanned Aircraft Systems (ICUAS)*, pages 966–973, 2013a.

M. Orsag, C. Korpela, and P. Oh. Modeling and control of mm-uav:mobile manipulating unmanned aerial vehicle. *Journal of Intel Robot Sys*, 69:227–240, 2013b.

I. Palunko, R. Fierro, and P. Cruz. Trajectory generation for swing-free maneuvers of a quadrotor with suspended payload: A dynamic programming approach. In *2012 IEEE International Conference on Robotics and Automation*, pages 2691–2697, May 2012.

V. Parra-Vega, S. Arimoto, Y. Liu, G. Hirzinger, and P. Akella. Dynamic sliding pid control for tracking of robot manipulators: theory and experiments. *IEEE Transactions on Robotics and Automation*, 19(6):967–976, 2003.

P. Pounds, D. Bersak, and Dollar A. Grasping from the air: hovering capture and load stability. In *2011 IEEE International Conference on Robotics and Automation*, pages 2491–2498, May 2011.

PRUSA. Prusa research, November 2016. URL `http://www.prusa3d.com/`.

M. H. Raibert and J. J. Craig. Hybrid position/force control of manipulators. *Journal of Dynamic Systems, Measurement, and Control*, 103(2):126–133, 1981.

C. Sangbum. *Dynamics and control of a underactuated multibody spacecraft*. PhD. Thesis Report, 2002.

K. Sebesta and J. Baillieul. Animal-inspired agile flight using optical flow sensing. In *Decision and Control (CDC), 2012 IEEE 51st Annual Conference on*, pages 3727–3734. IEEE, 2012.

F. Sharifi, M. Mirzaei, B. W Gordon, and Y. Zhang. Fault tolerant control of a quadrotor uav using sliding mode control. In *Control and Fault-Tolerant Systems (SysTol), 2010 Conference on*, pages 239–244. IEEE, 2010.

J. E. Slotine and W. Li. On the adaptive control of robot manipulators. *The international journal of robotics research*, 6(3):49–59, 1987.

M. W. Spong, S. Hutchinson, and Vidyasagar M. *Robot Dynamics and Control*. JOHN WILEY and SONS, INC., 2004.

ST-Robotics. Robotics within reach, November 2016. URL `http://www.strobotics.com/cylindrical-format-robot.htm`.

H. Sussmann, E. Sontag, and Y. Yang. A general result on the stabilization of linear systems using bounded controls. In *Decision and Control, 1993., Proceedings of the 32nd IEEE Conference on*, pages 1802–1807. IEEE, 1993.

S. Tarbouriech, Ch. Prieur, and JM. G. Da Silva. Stability analysis and stabilization of systems presenting nested saturations. *IEEE Transactions on Automatic Control*, 51 (8):1364–1371, 2006.

A. R. Teel. Global stabilization and restricted tracking for multiple integrators with bounded controls. *Systems & control letters*, 18(3):165–171, 1992.

J. Thomas, G. Loianno, K. Sreenath, and V. Kumar. Toward image based visual servoing for aerial grasping and perching. In *Robotics and Automation (ICRA), 2014 IEEE International Conference on*, pages 2113–2118. IEEE, 2014.

M. Tognon, B. Yüksel, G. Buondonno, and A. Franchi. Dynamic decentralized control for protocentric aerial manipulators. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*, May 2017.

Toshiba. Reach for the impossible, achieve the incredible, November 2016. URL `http://www.toshiba-machine.com/departmentlanding.aspx?dept=4`.

Y. Yali, J. Changhong, and W. Haiwei. Backstepping control of each channel for a quadrotor aerial robot. In *2010 International Conference on Computer, Mechatronics, Control and Electronic Engineering*, volume 3, Aug 2010.

B. Yüksel, G. Buondonno, and A. Franchi. Differential flatness and control of protocentric aerial manipulators with any number of arms and mixed rigid-/elastic-joints. In *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 561–566, Oct 2016a.

B. Yüksel, N. Staub, and A. Franchi. Aerial robots with rigid/elastic-joint arms: Single-joint controllability study and preliminary experiments. In *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 1667–1672, Oct 2016b.

E. H. Zheng, J. J. Xiong, and J. L. Luo. Second order sliding mode control for a quadrotor uav. *ISA Transactions*, 53:1350 – 1356, 2014. ISSN 0019-0578.