Design and Control of Multi-Directional Thrust Multi-Rotor Aerial Vehicles with applications to Aerial Physical Interaction Tasks

Davide Bicego

THÈSE

En vue de l'obtention du

DOCTORAT DE L'UNIVERSITÉ DE TOULOUSE

Délivré par :

l'Institut National des Sciences Appliquées de Toulouse (INSA de Toulouse)

Présentée et soutenue le *13/09/2019* **par :**

DAVIDE BICEGO

Design and Control of Multi-Directional Thrust Multi-Rotor Aerial Vehicles with applications to Aerial Physical Interaction Tasks

---

JURY

TAREK HAMEL	Professeur Unsiversité Nice Sophia Antipolis, Nice	Rapporteur
ISABELLE FANTONI	Directrice de Recherche CNRS, École Centrale de Nantes, Nantes	Rapporteur
PAOLO ROBUFFO GIORDANO	Directeur de Recherche CNRS, IRISA, INRIA, Rennes	Membre du Jury
MARC GOUTTEFARDE	Directeur de Recherche CNRS, Université de Montpellier, Montpellier	Membre du Jury
SIMON LACROIX	Directeur de Recherche CNRS, LAAS-CNRS, Toulouse	Membre du Jury
ANTONIO FRANCHI	Chercheur CNRS, LAAS-CNRS, Toulouse	Directeur de thèse

---

École doctorale et spécialité :

EDSYS : Robotique 4200046

Unité de Recherche :

Laboratoire d'Analyse et d'Architecture des Systèmes (LAAS–CNRS)

Directeur de Thèse :

Antonio FRANCHI

Rapporteurs :

Tarek HAMEL et *Isabelle FANTONI*

Abstract

This thesis addresses the study of autonomous **Aerial Vehicles (AVs)** actively interacting with the surrounding environment, with particular attention to the development of **modeling** and **design** techniques, and suitable **control strategies** for these systems. In the last decade, the interest towards Unmanned Aerial Vehicles (UAVs) in the scientific community has seen an overwhelming growth, which has led to the employment of such systems in a wide range of civil applications, ranging from remote monitoring/patrolling and aerial photography/video-shooting to search and rescue missions. The similarities shared by most of these applications are that they require the aerial vehicles to act as remote sensors which autonomously move in order to sense the environment without any contact. On the other hand, in the more recent research field of **aerial physical interaction**, the goal is to go beyond applications of pure and sole sensing, in order to accomplish challenging tasks that involve the exchange of forces and moments with the external environment like, e.g., pushing/pulling/sliding and manipulating an object. This enables aerial vehicles to become aerial robots, in the stricter meaning of the term. Due to the intrinsic difficulty and the novelty associated with the study of these systems, new techniques are needed to: *i)* better describe the aerial vehicle dynamics and its **actuation limits**; *ii)* effectively design new aerial prototypes with particular properties of dexterity and resilience; *iii)* guarantee a stable control during contact-less operations despite the actuation limits; and *iv)* preserve the system stability also during the contact phase with the environment while guaranteeing the fulfillment of the sought manipulation task.

Within the vast class of AVs, multi-rotor Vertical Takeoff and Landing (VTOL) vehicles of medium-small size are the ones which are inherently more suitable to perform this kind of operations, given their proficiency in performing agile maneuvers in confined spaces and their capability to hover maintaining a constant position, which substantially eases the manipulation tasks. Nowadays, most of VTOL Multi-Rotor Aerial Vehicles (MRAVs) are designed with all the propellers oriented towards a common direction in body frame, which makes them under-actuated systems. Despite maximizing the system efficiency, this solution permits these vehicles to generate only a Uni-Directional Thrust (UDT) force. In order to follow a generic position trajectory, the total force direction in world frame is changed by steering the whole vehicle chassis. Maneuvers in which rotation and translation are completely independent are precluded to these platforms. Presence of such an under-actuation does not only limit the set of maneuvers that the aerial vehicle can carry out but even deteriorates its potentiality to interact with the environment by rapidly exerting forces in an arbitrarily-chosen direction of the space while keeping a pre-specified orientation. This could be a serious problem in the case that, e.g., the platform has to move through a hostile and cluttered ambient or resist a wind gust while maintaining the desired attitude.

This thesis explores new strategies to overcome, to a certain extent, the aforementioned issues. At the design level, the simple proposed solution is to orient the aerial vehicle actuators in a different fashion, in order to take advantage of the **Multi-Directional Thrust (MDT)** force capability. This inceptive idea, already introduced by some works in the literature of aerial robotics and control, opens the door to many challenges, many of which were still not addressed or unsolved at the time of the beginning of this thesis. In particular, relevant room for research is still left in the field of aerial physical interaction exploiting such vehicles. The goal of this thesis is to contribute to a wise growth of the preliminary theoretical results on MDT MRAVs laid by the state of the art and, furthermore, to the development of more suitable real aerial robotic systems with enhanced manipulation means, tailored for aerial physical interaction tasks. More specifically, this is achieved by simultaneously addressing all the following phases:

i) modeling: new dynamical representations are formulated in order to better describe the new system capabilities, while explicitly taking into account the actuation limits. As will be shown, neglecting the physical limitations of the system can jeopardize the task fulfillment and, in the worst case, the vehicle stability;

ii) design: alternative configurations for the actuators are investigated, in order to bestow particular dynamic properties upon the aerial vehicle like, e.g., the capability to partially decouple the evolution of the translational dynamics from the rotational one. Ultimately, this leads to the realization of four different real prototypes which are unique in their kind;

iii) motion control: novel control laws are proposed for the accomplishment of precise trajectory tracking, which is the first requirement for an unmanned vehicle. Such methods guarantee compliance with the physical constraints synthesized in the model description; and

iv) interaction control: the transition between motion control to interaction control is managed in an accurate and safe way, thanks to the development of a rational framework which exploits techniques already known in the field of robotic ground manipulators.

This thesis takes place inside the context of the European H2020 AeroArms project, whose goal is to develop aerial robotic systems with advanced manipulation capabilities to be applied in industrial inspection and maintenance. Hence, also the technology transfer and the impact on the industry plays here an important role.

Keywords

– Multidirectional-thrust aerial robots – Aerial physical interaction
– Aerial manipulation – Robotics – Automatic control

Résumé

Cette thèse aborde l'étude de véhicules aériens autonomes interagissant d'une façon active avec l'environnement, en portant une attention particulière au développement des techniques de **modélisation**,de **conception**, et de stratégies de **commande** appropriées pour ces systèmes. Au cours de la dernière décennie, les véhicules aériens ont suscité un intérêt croissant de la part de la communauté scientifique, ce qui a conduit à l'utilisation de tels systèmes dans un large éventail d'applications civiles, allant de la surveillance/patrouille à distance à la photographie/vidéo aérienne pour des missions de recherche et de sauvetage. La plupart de ces applications ont pour similitude d'exiger que les véhicules aériens agissent comme des capteurs à distance qui se déplacent de manière autonome afin d'acquérir des données sur l'environnement sans aucun contact. D'autre part, dans le domaine plus récent de **interaction physique aérienne**, le but est d'aller au-delà des applications de détection pure et simple, afin d'accomplir des tâches difficiles impliquant un échange de forces et de moments avec l'environnement (comme par exemple pousser/tirer/glisser et manipuler un objet). Cela permet aux véhicules aériens de devenir de véritables robots aériens au sens strict du terme. L'étude de ces systèmes étant intrinsèquement complexe et relativement récente, de nouvelles techniques sont nécessaires pour : *i)* mieux décrire la dynamique du véhicule aérien et ses contraintes d'actionnement ; *ii)* concevoir efficacement de nouveaux prototypes aériens dotés de propriétés particulières de dextérité et de résilience ; *iii)* garantir un contrôle stable pendant les opérations sans contact malgré les contraintes d'actionnement ; et *iv)* préserver la stabilité du système pendant la phase de contact avec l'environnement tout en garantissant l'accomplissement de la tâche de manipulation.

Dans la vaste catégorie des véhicules aériens, les véhicules multi-rotors à décollage/atterrissage vertical (VTOL en anglais), de petite à moyenne taille, sont ceux qui conviennent le mieux pour effectuer ce type d'opération. En effet, leurs capacités de survol en position constante et de manœuvre en espaces confinés facilitent considérablement les tâches de manipulation. De nos jours, la plupart des ces véhicules sont conçus avec les hélices orientées dans une même direction, ce qui en fait des systèmes sous-actionnés. Malgré l'optimisation de l'efficacité du système, cette solution permet à ces véhicules de générer uniquement une force de poussée unidirectionnelle. Afin de suivre une trajectoire de position générique, la direction de la force totale (dans un repère fixé de l'environnement) est modifiée en dirigeant l'ensemble du châssis du véhicule. La présence d'un tel sous-actionnement ne limite pas seulement l'ensemble des manœuvres que le véhicule aérien peut effectuer, mais détériore même son potentiel d'interaction avec l'environnement. En particulier, la plateforme ne peut pas à la fois exercer rapidement des forces dans une direction donnée et garder une orientation spécifique. Cela pourrait poser un problème dans le cas où, par exemple, la plate-forme doit se déplacer dans un environnement hostile et encombré ou résister à une rafale de vent tout en conservant l'attitude souhaitée.

Cette thèse explore de nouvelles stratégies pour surmonter, dans une certaine mesure, les problèmes susmentionnés. Au niveau de la conception, la solution simple qui est proposée consiste à orienter les actionneurs des véhicules aériens d'une manière différente afin de tirer parti de la **capacité de poussée multidirectionnelle**. Cette idée, initialement introduite par certains travaux de robotique et de contrôle aériens, ouvre la porte à de nombreux défis non abordés ou non résolus au moment du début de cette thèse. En particulier, il reste une marge de recherche pertinente dans le domaine des interactions physiques aériennes en exploitant de tels véhicules. L'objectif de cette thèse est d'enrichir les résultats théoriques préliminaires sur ces plateformes et, en outre, de contribuer au développement de systèmes robotiques aériens réels plus appropriés aux moyens de manipulation améliorés et adaptés aux tâches d'interaction physique aérienne. Plus précisément, ceci est réalisé en abordant simultanément les phases suivantes :

i) modélisation : de nouvelles représentations dynamiques sont formulées afin de mieux décrire les nouvelles capacités du système, tout en tenant explicitement compte des contraintes d'actionnement. Comme nous le montrerons, négliger les limitations physiques du système peut compromettre l'exécution des tâches et, dans le pire des cas, la stabilité du véhicule ;

ii) conception : des configurations alternatives pour les actionneurs sont étudiées, afin de donner aux véhicules une dynamique particulière, comme par exemple la possibilité de découpler partiellement l'évolution de la dynamique de translation de celle de rotation. Cela conduit finalement à la réalisation de quatre prototypes réels différents, uniques en leur genre ;

iii) contrôle du mouvement : de nouvelles lois de contrôle sont proposées pour la réalisation d'un suivi précis de trajectoire, première exigence pour un véhicule sans pilote. De telles méthodes garantissent le respect de contraintes physiques synthétisées dans la description du modèle ; et

iv) contrôle d'interaction : la transition entre le contrôle du mouvement et le contrôle d'interaction est gérée de manière précise et sûre, grâce à la mise au point d'un cadre rationnel exploitant des techniques déjà connues dans le domaine des manipulateurs robotiques au sol.

Cette thèse s'inscrit dans le cadre du projet européen H2020 AeroArms, dont le but est de développer des systèmes robotiques aériens dotés de capacités de manipulation avancées à appliquer dans les domaines de l'inspection et de la maintenance industrielles. Par conséquent, l'impact sur l'industrie joue ici un rôle important.

Mots Clés

– Robots aériens à poussée multidirectionnelle – Interaction physique aérienne
– Manipulation aérienne – Robotique – Contrôle automatique

Acknowledgements

These few lines of due personal acknowledgments represent the completion of this PhD thesis manuscript, which itself epitomizes the finalization of the thrilling and stimulating experience which I lived during the period of my life related to the PhD studies. Despite my conviction that personal motivation, effort, and determination are the keys to reach all important goals in life, I believe some words should be spent here to sincerely thank all the people that followed me through this adventure until this spatial/temporal 'configuration' of my life, which I honestly feel very special.

During these years of work, there has been a considerable number of colleagues who I met, with whom I discussed, collaborated, and shared a lot of ideas. First of all, I would like to express my most sincere gratitude to my supervisor Dr. Antonio Franchi, who has always guided me and my research in a very active, motivated, and present way. His determination, intuition, and insatiable desire for new research have often been for me a source of inspiration which partially changed my way to see and perceive many things. In particular, thanks for having implicitly taught me that sometimes one can definitively learn much more from a failure rather than from a success, and for having pushed me beyond (what I believed to be) my limits. A special acknowledgments goes to Dr. Markus Ryll, from whom I learned a lot of technical things and with whom I spent a lot of enjoyable time during the preparation and the first part of my PhD studies. In particular, thanks for his availability and his kind support, which I profoundly appreciated. I would like also to thank to my colleagues Dr. Marco Tognon and Dr. Nicolas Staub for the nice moments spent together inside and outside LAAS–CNRS. Sharing with you the intense moments which go together with the accomplishment of strict deadlines have (hopefully) taught me to better deal with stressful situations.

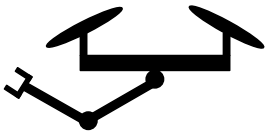
Adventures can be truly better when faced together with good companions. For this reason, I would like to express my most sincere gratitude to Dario Sanalitro and Andrea De Maio, who I consider much more than just colleagues. Thank you for the (un?)-fruitful discussions around the the sense of the universe, for your constant support during some of my wrong days at work, and especially for your funny yet effective motivation during the writing of this thesis. I will never forget those moments. With this regard, a special acknowledgment goes also to Enrica Rossi, Giulia Michieletto, Chiara Gabellieri, and Michele Furci. I would like to additionally thank the students that I have supervised. Teaching is learning twice.

I would like to express my finest gratitude to the whole LAAS–CNRS, directed by Dr. Liviu Nicu, and especially to the Robotics and InteractionS research group, led by Dr. Rachid Alami. This laboratory has been a second home for me, while its people almost a big family. *Un remerciement particulaire est direct à Toulouse, la ville rose qui restera toujours dans mon coeur, et aux gens magnifiques que j'ai connus là-bas. Merci!* Another acknowledgment goes also to the colleagues with whom I collaborated outside LAAS–CNRS, in particular to Dr. Mostafa Mohammadi, Dr. Quentin Delamare, Dr. Min Jun Kim, and Iurii Sarkisov.

One of the most important due personal acknowledgments is directed to my family, who has always been a light for my path and who has always and unconditionally supported during the important moments of my life. To my mother Celestina, my father Paolo, my brother Massimo, my sister Elisa, my grandmother Sabina (who has been like a second mother to me) and to my grandparents Elsa and Giampietro. This achievement was obtained also thanks to all of you. Furthermore, I would like to additionally thank my close friends Francesco Martini, Filippo Astegno, Luca Abbondandolo, Fabio Melison, and Marco Franceschi for their 'remote' support I could rely on when needed, despite the geographical distance.

Last but definitely not the least, my most sincere gratitude goes to Yasmine, my missing half apple. Thanks you for having followed me through this amazing experience and for the energy that you constantly give to me without even realizing it. Staying with is an adventure by itself, which I enjoy every day.

Davide



Contents

List of Figures

List of Tables

List of Multimedia

[video01–2017] **video01**. *Full-Pose Tracking Control for Aerial Robotic Systems With Laterally Bounded Input Force*. https://www.youtube.com/watch?v=pCKfqRSSH4g&list=UUFoSjtoDdbt4G4mL7p1jciA&index=10. 2017 (cited on page 115).

[video02–2017] **video02**. *Full-Pose Tracking Control for Aerial Robotic Systems With Laterally Bounded Input Force*. https://www.youtube.com/watch?v=8TQFzNtjuSE&list=UUFoSjtoDdbt4G4mL7p1jciA&index=11. 2017 (cited on page 118).

[video03–2017] **video03**. *Full-Pose Tracking Control for Aerial Robotic Systems With Laterally Bounded Input Force*. https://www.youtube.com/watch?v=irjKUMOyttc&list=UUFoSjtoDdbt4G4mL7p1jciA&index=12. 2017 (cited on page 120).

[video04–2019] **video04**. *Nonlinear Model Predictive Control with Actuator Constraints for Multi-Rotor Aerial Vehicles*. https://www.youtube.com/watch?v=sNOD3_MTHDc. 2019 (cited on pages 136, 142 and 145).

[video05–2019] **video05**. *6D Interaction Control with Aerial Robots: The Flying End-Effector Paradigm*. https://www.youtube.com/watch?v=SDe-2bl_X0k&t=11s. 2019 (cited on pages 171 and 239).

[video06–2019] **video06**. *6D Interaction Control with Aerial Robots: The Flying End-Effector Paradigm*. https://www.youtube.com/watch?v=jQ6qr38Lulg. 2019 (cited on pages 193 and 239).

[video07–2019] **video07**. *6D Interaction Control with Aerial Robots: The Flying End-Effector Paradigm*. https://www.youtube.com/watch?v=5lsje-QtMf0&t=7s. 2019 (cited on page 239).

[video08–2018] **video08**. *Towards a Flying Assistant Paradigm: the OTHex*. https://www.youtube.com/watch?v=AikN3_PgYU4&list=UUFoSjtoDdbt4G4mL7p1jciA&index=15. 2018 (cited on page 181).

[video09–2018] **video09**. *Tele-MAGMaS: an Aerial-Ground Co-Manipulator System*. https://www.youtube.com/watch?v=TrrPEP3CNlY&list=UUFoSjtoDdbt4G4mL7p1jciA&index=8. 2018 (cited on page 183).

[video10–2018] **video10**. *A Truly Redundant Aerial Manipulator System with Application to Push-and-Slide Inspection in Industrial Plants*. https://www.youtube.com/watch?v=_n_f617b-Ho&list=UUFoSjtoDdbt4G4mL7p1jciA&index=7. 2018 (cited on pages 189 and 199).

[video11–2017] **video11**. *Tele-MAGMaS at Hanover Fair 2017: KUKA video*. https://vimeo.com/217252361. 2017 (cited on page 211).

[video12–2017] **video12**. *Tele-MAGMaS at Hanover Fair 2017: interview 1.* `https://www.youtube.com/watch?v=o9xmPYNPfRE&list=UUFoSjtoDdbt4G4mL7p1jciA&index=19`. 2017 (cited on page 211).

[video13–2017] **video13**. *Tele-MAGMaS at Hanover Fair 2017: interview 2.* `https://youtu.be/GRnGSvJGUKk`. 2017 (cited on page 211).

List of Acronyms

AM Aerial Manipulation

APhI Aerial Physical Interaction

API Application Programming Interface

AR Aerial Robot

AUV Autonomous Underwater Vehicle

AV Aerial Vehicle

CAD Computer Aided Design

CoM Center of Mass

CPU Central Processing Unit

DoF Degree of Freedom

EC Eddy Current

EE End-Effector

ESC Electronic Speed Controller

FRA Fully-Ranked Allocation Matrix

FSM Finite-State Machine

GNSS Global Navigation Satellite System

GPS Global Positioning System

IMU Inertial Measurement Unit

LBF Laterally-Bounded input Force

LMPC Linear Model Predictive Control

LQR Linear Quadratic Regulator

MAGMaS Multiple Aerial-Ground Manipulator System

MDT Multi-Directional Thrust

MoCap Motion Capture System

MPC Model Predictive Control

MRAV Multi-Rotor Aerial Vehicle

NDT Non-Destructive Testing

NMPC Nonlinear Model Predictive Control

OCP Optimal Control Problem

OD Omni-Directional

PCA Principal Component Analysis

PID Proportional, Integrative and Derivative

PWM Pulse Width Modulation

QP Quadratic Programming

RMSE Root Mean Square Error

ROS Robot Operating System

RPV Remotely Piloted Vehicle

RTI Real Time Iteration

RTK Real-Time Tinematic

SQP Sequential Quadratic Programming

TRL Technology Readiness Level

UAR Unmanned Aerial Robot

UAV Unmanned Aerial Vehicle

UDT Uni-Directional Thrust

VTOL Vertical Takeoff and Landing

Part I

Preliminaries

Chapter 1

Introduction

"I would rather have questions that can't be answered than answers that can't be questioned."
Richard Feynman

This first chapter serves as an introduction to this manuscript and has a threefold goal. First of all, it shall clarify the research topic addressed by this thesis and formalize the problems it shall solve. Secondly, it intends to contextualize this work within the broad literature panorama. Last but not least, it aims to highlight the contribution brought to the state of the art.

Motivation

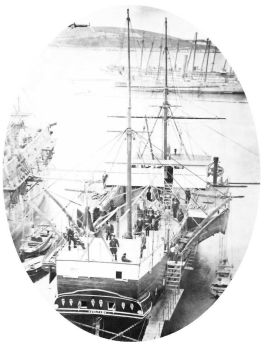
In order to meaningfully motivate the research directions explored in the presented work, this section shall start by introducing the general field of aerial robotics to naturally converge to the well-identified open problems we aim to work out.

Introduction on aerial robotics research and taxonomy

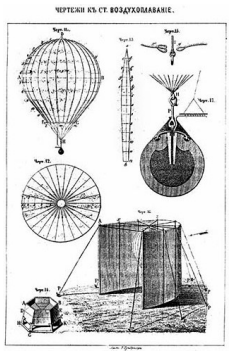
Nowadays, we continuously witness a relentless intensification of the entanglement between robotics and our everyday life. From the research point of view, the field of robotics has stimulated an increasing number of studies and works which produced a flourishing scientific literature. Robots are making a considerable impact on many aspects of modern life, from industrial manufacturing to health-care, transportation [Siciliano–2016], and in many other fields. In a wide set of scenarios, these robots operate from a fixed base on the ground, which earned them the appellation of *ground robots*. Canonical examples of this class are the well-known fixed-base industrial manipulators thoroughly studied in many works on robotics like, e.g., [Paul–1981; Siciliano–2010; Murray–2017]. Certainly, an active control of the base position comes with the significant benefit of an increased workspace for the robot. The base locomotion can be achieved with different means like e.g., wheels, thus generating wheeled robots (see [Campion–1993]), and legs of different kinds, hence producing bipeds, quadrupeds, humanoids, etc. (see [Raibert–1986]). Despite the capability of these robots to span big environments, they are still constrained to move and perform operations on the ground.

It is easy to be convinced that robots which manage to freely fly in the 3D space, namely *aerial robots*, can achieve a wider workspace than those that do not have

[weaponsandwarfare.com]



[en.wikipedia.org]



[weaponsandwarfare.com]

Figure 1.1 – *Left:* the first aerial bombardment in history, performed with UAV balloons by the Austrians to the city of Venice, July 1849. *Center:* SMS Vulcano ship, acting as the balloon carrier. *Right:* sketches of the balloons (F. von Uchatius).

these means. This simple ascertainment, together with many other motivations discussed in the following, contributed to the constant growth of the field of *aerial robotics*, occurred in the last decades. The main goal of aerial robotics is to study, conceive and, possibly, bring to light aerial systems capable to perform a given task fully or partially autonomously, while maintaining a stable flight. In the related literature, such systems are often referred to as Unmanned Aerial Vehicles (UAVs) or, less frequently, as Unmanned Aerial Robots (UARs). More in detail, the subtle difference between an Aerial Vehicle (AV) and an Aerial Robot (AR), which is not the object of strict definition in this thesis, will be further clarified.

Although it is only relatively recently that UAVs gained the interest of a broad and still expanding community, preliminary study and design of such systems were performed even before the advent of the 20^{th} century. The first known use in history of this technology occurred, as happens in many cases, for military purpose. On July 12, 1849, the Austrian forces used UAV balloons containing bombs which were to be ignited by means of electromagnetism to target the besieged Venice, in the Republic of San Marco. The design of UAVs was carried on during the two World Wars. However, the lack of an appropriate technology level, in particular in relation to sensors, prevented an autonomous and reliable navigation. Throughout the following years, the related study remained heavily fostered by military interests, exploited for training, remote surveillance and combat engagement. From this military background came the denomination "drone", which could be related to the resemblance of the sounds of early UAV motors with the hum produced by the male bee (called drone). Other sources suggest that the term was coined to underline that the platform could not function on its own and had to be remotely controlled, probably in relation to the limited natural capabilities of the flying insect with the same name. This term, today almost exclusively used by the press, left the place to Remotely Piloted Vehicle (RPV) to designate AVs with military purpose. In this thesis, we address only civil applications. Therefore, we believe UAV and UAR to be more suitable appellations.

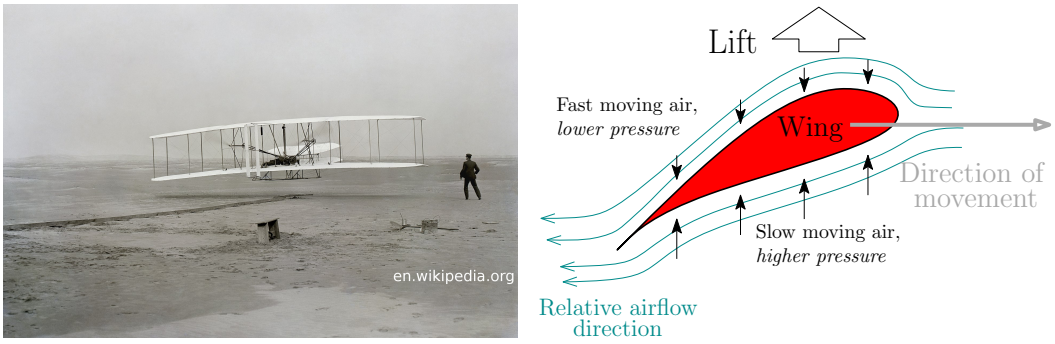


Figure 1.2 – *Left:* The first controlled fixed-wing airplane known in history, realized by the Wright brothers, 1903. *Right:* a simplification of the theory on lift generation (several aerodynamic effects are omitted for simplicity), based on Bernoulli's theory.

It was only from the early 1980s, with the development of accurate sensors and the appearance of Global Navigation Satellite Systems (GNSSs) and of powerful and lightweight processors, that UAVs started to acquire better sensing and navigation capabilities. This technology enhancement, together with the concurrent price drops in small consumer electronics, contributed to spreading the interest towards UAVs, mainly of small size, to a very wide research community and industries outside the scope of heavily funded military projects. This generated a sudden boost of money investments related to aerial robotics, which still endures in the present days. The low cost, compared to other robotic solutions, the theoretically unlimited workspace and the considerable versatility of UAVs allow their employment in a very large spectrum of applications. A yet incomplete list of such applications should record agriculture monitoring, remote patrolling of areas, search and rescue operations, wildfire supervision, archaeological and geographical mapping of sites, wildfire monitoring, and professional video-shooting. Considering also utilization from the general public, we should also add racing challenges, airborne photography, and general entertainment. For a more complete overview of the current status of remote sensing applications based on UAVs, we refer the interested reader to [Pajares–2015]. Other employments that are currently under study and might become possible in the near-future include personal and goods transportation (see, e.g., Amazon[1], Airbus[2] and Volocopter[3]).

Throughout the years, several types of UAVs have been implemented. A preliminary taxonomy could be done w.r.t. the way the platform can physically overcome the constantly-pulling gravitational force. Compared to other mobile vehicles like, e.g., legged robots, automobiles and Autonomous Underwater Vehicles (AUVs), UAVs have to face different physical challenges, given the fact that they have to always counterbalance this force. In a non-vacuum environment, e.g., in a planet

[1] `https://www.amazon.com/Amazon-Prime-Air/b?ie=UTF8&node=8037720011`
[2] `https://www.airbus.com/innovation/urban-air-mobility/vehicle-demonstrators.html`
[3] `https://www.volocopter.com/en/`

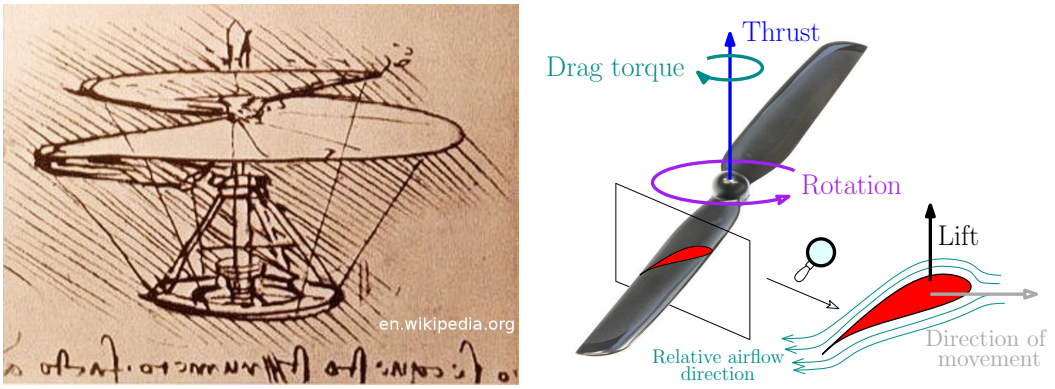


Figure 1.3 – *Left:* Leonardo's design of the "aerial screw", the forerunner of modern helicopters, early 1480s. *Right:* a simplification of the blade element theory.

atmosphere like the Earth one, this might be accomplished by generating difference in the air pressure on strategical points of the floating platform. This can be fulfilled with a *fixed-wing* design, which represents the key component for the operation of airplanes, as simplistically shown in Fig. 1.2. A formal description of fundamentals of aerodynamics can be found in [Anderson Jr–2010]. Alternatively, a lift force can can also be generated with *flapping wings*, inspired by the flight of birds and insects (see, e.g., [De Croon–2009]). However, such designs have some shortcomings. Indeed, the former require high cruise velocities for the robot, which could limits its application in cluttered environments, in particular precluding the accomplishment of tasks which require to maintain a constant and stable position in the air, i.e., to *hover*. On the other hand, the latter often require complex mechanisms and hardly achieve precise maneuverings. These drawbacks can be overcome by using rotating propellers, in the same way helicopters operate, as shown in Fig. 1.3. Such actuation means might provide the UAVs with the useful Vertical Takeoff and Landing (VTOL) capability, i.e., the skill to hover, take-off and land vertically, without the need for open and smooth areas like runways, or particular launching mechanisms. These facts enable the employment of VTOL rotor-crafts in indoor and cluttered environments such as forests and urban/industrial surroundings. This dynamic effectiveness comes with the cost of a typically increased power consumption. Mixed-solution which combine cruising flight and VTOL capabilities for small vehicles have been explored [Morin–2015; Anglade–2019]. Apart from helicopters, whose model has been thoroughly studied in [Ren–2012], other examples of *rotor-craft* designs are ducted-fan platforms, which result more effective for applications where the size must be small and static thrusts are needed [Naldi–2010; Hofer–2016]. Finally, particularly interesting designs are obtained embedding the floating platforms with multiple rotors, normally more than two. Vehicles obtained in this way earn the name Multi-Rotor Aerial Vehicles (MRAVs). An advantage of MRAVs is the simpler rotor mechanics required for flight control. Differently from helicopters, which often use variable-pitch propellers, these vehicles often use fix-pitch blades. The control

of such vehicles is achieved by regulating the speed of the different rotors in order to change the thrust and the torque produced by each of them and, consequently, the wrench applied to the vehicle Center of Mass (CoM). The most representative platform of this sub-class is undoubtedly the quadrotor [Hamel–2002; Mahony–2012a], well-known for its simple but effective design. According to the particular task to be accomplished, one can choose the most suitable vehicle that better fits the sought behavior, finding the best *trade-off* between flight endurance/energy consumption and maneuverability (to this purpose, see [Filippone–2006]). Finally, another simple classification of UAVs could be delineated w.r.t. their size. The interested reader in a very detailed review of these vehicles is addressed to [Cai–2014]. Moreover, [Liew–2017] presents a survey on recent prototypes development. Aiming at the employment of such vehicles for civil applications, in this thesis we restrain the attention to small-scale UAVs. More in detail, driven by the previous considerations and by the ones which will follow, we focus our study on VTOL MRAVs.

Introduction on Aerial Physical Interaction

It is interesting to remarks that in most of the applications listed so-far, UAVs are used as remote sensors to accomplish *contact-less* tasks, i.e., they are assigned to gather data with different sensors, e.g., cameras, without interacting with the environment. Although this class of sole-sensing applications offers already fascinating and arduous challenges to researchers, it still represents a proper sub-set of the ensemble of directions that can be explored. Furthermore, its scope is actually limited w.r.t. the real potential of aerial systems. In the very recent years, UAVs started to be employed for preliminary *in-contact* operations which involve an exchange of forces and torques with the environment, with the goal of performing physical works. This achievement allows to suitably refer to such AVs as ARs in the stricter meaning of the term. Indeed, the word "robot" draws its origins from the Czech word *robota*, which means "forced labor". This step forward paved the way for the occurrence of flourishing studies of the topics of Aerial Physical Interaction (APhI) and Aerial Manipulation (AM), with the latter being a special subject of the former, in which the flying robot is designed and controlled with the additional goal of *manipulating* the environment. A good overview of research works on these topics can be found in [Staub–2018c]. The domain of interest of APhI and AM are extremely variegate and far-reaching. Examples of real-life employments are inspection and maintenance by contact of sensible sites (see Fig. 1.4), assembly/construction and decommissioning of structures, assistance robotics in industrial/urban/domestic surroundings, removal of debris after natural catastrophes, delivery and transportation, and many others can be appended. One of the most remarkable use of ARs is possibly found in risky and dangerous operations, in order to prevent humans from being hurt or injured. Moreover, from a more practical point of view, the use of robots allows to reduce the cost associated with many operations.

Figure 1.4 – An example of aerial physical interaction task: the inspection with contact of oil/gas pipelines. *Left:* the addressed scenario. *Center:* the solution today, achieved by human operators. *Right:* the solution tomorrow in the vision of the AeroArms project, with the task performed by aerial robots.

Considered the relevance of the problem, many laboratories and also companies have steered their related research towards it. As an outcome, many international synergies aimed to advance in this topic were established. In the scope of the European Union, in the last ten years, this gave birth to diverse collaborative projects, a partial list of which is given in the following, together with the related goal.

- ARCAS[4]: conceive ARs for assembly and construction of structures;
- AEROARMS[5]: design and build ARs with high manipulation capabilities for industrial inspection and maintenance, further integrating tele-manipulation;
- AEROWORKS[6]: provide heterogeneous and collaborative aerial robotic workers for inspection and maintenance tasks in infrastructure environments;
- AIROBOTS[7]: design ARs for remote inspection by contact and to support human beings in applications which require interaction capabilities;
- AEROBI[8]: conceive ARs with a specialized multi-joint arm for in-depth structural inspection of concrete bridges and with short term marketing scope;
- ARCOW[9]: introduce aerial co-workers collaborating with humans in manufacturing processes in order to reduce their costs and increase the efficiency;
- HYFLIERS[10]: conceive a robot with hybrid air and ground mobility with a long-reach hyper-redundant manipulator to work in hardly-accessible sites;

[4] http://www.arcas-project.eu/
[5] https://aeroarms-project.eu/
[6] http://www.aeroworks2020.eu/
[7] http://airobots.dei.unibo.it/
[8] http://www.aerobi.eu/
[9] http://www.euroc-project.eu/index.php?id=grvc-catec
[10] http://www.oulu.fi/hyfliers/

- SPECTORS[11]: produce advancements in the fields of UAV and sensing technology, big data cloud computing, applications in precision agriculture, as well as nature conservation and environmental protection;
- PRO-ACT[12]: develop and demonstrate a cooperation and manipulation capabilities between different robots for in-situ resource utilization, in the context of space exploration;
- AERIAL-CORE[13]: develop core technology modules and an integrated aerial cognitive robotic system with high capabilities on the operational range and safety in the interaction with people, or aerial co-workers for applications such as the inspection and maintenance of large linear infrastructures.

This new field of complex tasks including grasping and manipulation results in new challenges in the mechanical design of aerial vehicles and in the control of their dynamics. This thesis raises its contributions on top of this clear observation.

Exploiting a floating platform, an aerial robot has to react to forces/torques arising form the interaction with the environment in an *active* fashion. This is quite different from the typical working condition of a *ground* robot, since the latter can also exploit the ground reaction provided by its constrained base in a *passive* way. Additionally, it should be noticed that the hardware available for ground robots manipulators allows, in most of the cases, to accurately control the torque that each motor applies to the corresponding joint. On the other hand, the control input for MRAVs is typically (in first approximation) the velocity of rotating propellers, which in turn generate, thanks to the air interaction, the thrust forces and moment. Due to the complexity of the aerodynamic effects, the precise control of such forces turns out to be a quite complex task. To overcome substantial actuation errors, a closed-loop of the rotor velocity is advisable [Franchi–2017].

A critical issue for aerial robot interaction control is the measurement of the interaction wrench. A reliable solution is the adoption of force/torque sensors, such as in [Antonelli–2016] where the wrench measurement of a wrist mounted sensor of an aerial manipulator is fed to an admittance filter. In order to avoid additional payload, the sensor could be placed on the interaction surface [Gioioso–2014b], even if such solution might be not always viable. Use of force/torque sensors increases the cost and the weight of the aerial platform, thus alternative solutions based on wrench estimators have been proposed in the last years. In [Yüksel–2014a] a Lyapunov-based nonlinear observer is proposed for estimating the external wrenches applied on a quadrotor, while in [Tomic–2014] a hybrid estimation is proposed, using the linear acceleration for directly computing the interaction forces and a momentum based observer for estimating the interaction torques [De Luca–2005]. In [Tomic–2017], the same authors propose a more refined hybrid estimation, where the estimated forces are not simply computed by the model but are obtained via a first-order stable filter, similarly to the solution proposed in [Yüksel–2014a]. In [Rajappa–2017], the authors, by exploiting both a wrench estimation and a ring of

[11] `https://spectors.eu/wordpress/`

[12] `https://www.h2020-pro-act.eu/about/`

[13] `http://mrs.felk.cvut.cz/projects/aerial-core` (unofficial temp. web-page)

eight contact sensors, proposed a control able to separate human interaction forces from additional disturbances as wind and parameter uncertainties. Furthermore, in [Augugliaro–2013] Kalman filters are used to estimate the external forces. Finally, the external wrenches acting on quadrotor are also estimated by an algorithm based on the unscented quaternion estimator [McKinnon–2016].

In order to enhance the manipulation capability of ARs and to improve their dexterity during the task accomplishment, they are frequently endowed with interactive tools. The simplest tool which could be envisioned is a *rigid tool*. In the first case, the tool is stably fixed to the air-frame, see, e.g., [Nguyen–2013; Gioioso–2014a; Gioioso–2014b; Augugliaro–2014; Yüksel–2014b; Staub–2017]. This enables the exchange of forces/torques with the environment, e.g., pushing/poking surfaces or objects [Yüksel–2014a; Nguyen–2015]. Interesting studies and experimental results on such systems can be found in [Yüksel–2017; Mohammadi–2017]. A drawback of this solution, as better clarified in the next paragraph, is that typical VTOL MRAVs are *under-actuated* and therefore it is impossible to independently control the 6D (position plus orientation) dynamics of the end-effector. This limits the potential applications of the robots and could also create stability issues. In fact, it has been shown that in the presence of interaction with points of the air-frame different from the vehicle's CoM, the internal dynamics of under-actuated multi-rotors is not guaranteed to be stable, and it is, in general, neither easy to stabilize nor practical for real applications [Nguyen–2013]. Another simple yet useful interaction tool is represented by one or more *cables* [Sreenath–2013; Tagliabue–2016]. Indeed, they allow to partially decouple the rotational dynamics of the vehicle w.r.t. the one of the load. However, the control authority of the load pose might result limited and particular attention has to be devoted to the control in order to prevent undesired load oscillations that could make the system unstable. A thorough analysis of tethered UAVs (MRAVs in particular) has been performed in [Tognon–2018c]. Finally, another possibility is to attach an n-Degree of Freedom (DoF) *articulated robotic arm* to the aerial platform ([Fumagalli–2012; Kim–2013; Kondak–2014; Suarez–2015; Baizid–2016; Muscio–2016; Tognon–2017; Muscio–2017]), a solution which aims at overcoming the under-actuation of the end-effector dynamics by exploiting the increased number of actuators provided by the arm. In this way, an improved dexterity at the end-effector side becomes possible ([Yang–2014]). A detailed survey on the topic of aerial manipulation has been recently published [Ruggiero–2018]. Depending on the number of the DoFs, the load can be manipulated independently from the motion of the platform. Furthermore, if the former is higher w.r.t. the dimension of the load configuration space, the robot redundancy can be exploited in order to better compensate external disturbances or to fulfill other tasks [Lippiello–2012; Ryll–2018]. However, this solution comes with some drawbacks as well. In particular, a robotic arm strongly decreases the payload and flight time due to its own weight. These problems, which create though challenges in aerial robotics, can be partially overcome by adopting lightweight manipulators [Suarez–2015]. Secondly, the system results much more complex from a mechanical point of view than a single air-frame with a rigid tool and, thus, it is more expensive to build and

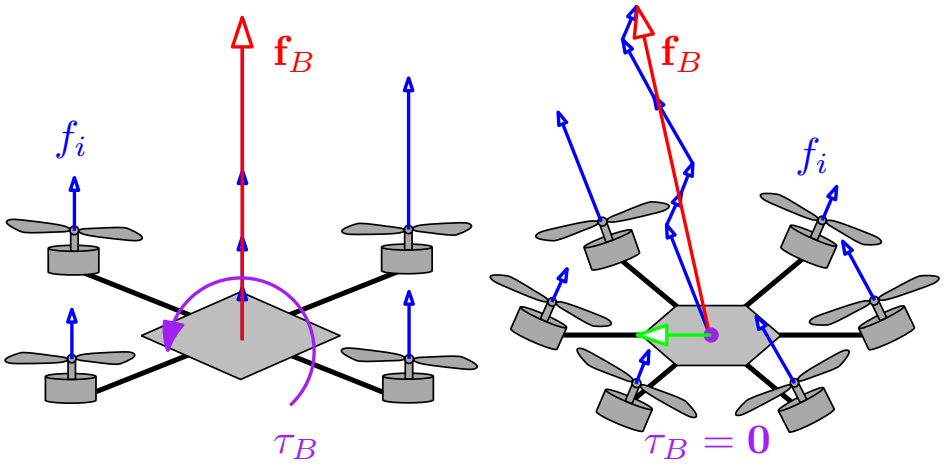


Figure 1.5 – Comparison of the thrust generation for a UDT (*left*) and a MDT (*right*) MRAV. In particular, the MDT platform can steer the thrust without re-orient itself. The UDT platform needs instead to first apply a moment and then wait until the whole platform reorients toward the desired total thrust direction in order to be able to apply the same force in world frame. This maneuver introduces an unavoidable actuation lag in UDT platforms when compared to the MDT ones.

also requires more maintenance across its operational life. Finally, lateral forces in body frame, which cannot be provided by typical under-actuated aerial platforms themselves, have to be generated through the dynamical/inertial coupling between the arm and the aerial robot: the proper mastering of the dynamical coupling is something that has to be necessarily exploited in order to get the sought benefits in terms of 6D force control. This, in turn, requires the knowledge of the precise dynamical model and a very accurate measurement of the system inputs and states (position, orientation, linear and angular velocities). As a matter of fact, these requirements are extremely hard to achieve in real world conditions (especially the former). For this reason, kinematic-only approaches have been preferred for real world validations, see e.g., [Muscio–2016; Muscio–2017], at the expense of losing the main benefits for which the manipulator was introduced.

Comparison between UDT and MDT MRAVs

In this thesis, special attention is devoted to VTOL MRAVs, given the aforementioned nice features of such vehicles. Nevertheless, it is fundamental to point out that traditional designs suffer from a major drawback. Indeed, in typical VTOL platforms all the propellers, responsible for the robot actuation, spin about parallel directions, i.e., they are collinear. The thrust force generated by each propeller is directed along the same line and therefore the total force is exerted along that unique fixed direction in body frame, see Fig. 1.5 (left). We refer to these MRAVs as Uni-Directional Thrust (UDT). Despite this configuration being the most efficient

in terms of energy consumption, since all the thrust forces are completely exploited to compensate gravity and generate motion, it comes with some pitfalls. Maneuvers in which rotation and translation are completely independent are precluded to such vehicles, which constitutes a serious problem in the case that, e.g., they are tasked to move through a hostile and cluttered ambient or to resist a wind gust while keeping a desired attitude. Indeed, in order to follow an arbitrary $3D$ position trajectory, these robots have to modify their orientation in order to steer the thrust direction at every instant in accordance with the needed linear acceleration. In other words, their longitudinal and latitudinal motion are a by-product of the orientation control. This fact makes UDT MRAVs under-actuated systems, in the sense that they can not be commanded to follow arbitrary trajectories in their full configurations space. This is a challenging problem from a control theory point of view, which needs to be handled properly [Spong–1998; Fantoni–2001]. Such an under-actuation even deteriorates the potentiality to interact with the environment by rapidly exerting forces in an arbitrarily-chosen direction of the space while keeping a pre-specified orientation. As a consequence, physical interaction with such platforms is challenging, see e.g., [Yüksel–2014b] and references therein, but sub-optimal.

In the literature, the major solution to overcome these identified issues has been to mount the rotors in a tilted way such that the thrusts of the propellers are not collinear anymore, see Fig. 1.5 (right). In this way, the direction of the total force can be changed by selecting the intensity of the force produced by each propeller, without the need of reorienting the whole vehicle. We identify these vehicles with the denomination of Multi-Directional Thrust (MDT). MDT MRAVs can resist external disturbances while completing a manipulation task without the need to change their orientation, which could jeopardize the manipulation task itself. In the recent years, this idea has been implemented in some real platforms and is becoming more and more popular in the aerial vehicles and robotics communities. At the best of our knowledge, the first contributions proposing to use a MRAV configuration with the actuators *fixed* in a non-collinear way have been [Romero–2007; Salazar–2009]. In these papers, the authors designed a special octo-rotor platform, i.e., a MRAV with eight rotors, where the four co-planar propellers traditionally used to stabilize the vehicle are supported by four perpendicular ones, employed to drive the lateral displacements. The main advantage of such configuration is that the attitude dynamics is decoupled, to a certain extent, from the translational one. The authors exploited this fact to be able to accurately measure the horizontal speed of the rotor-craft, exploiting optical flow with a down-facing camera. Despite the novel concept introduced by this work, such octorotor results limited in the set of body attitudes that it can attain. Observing this fact, and further aiming to reduce the number of actuators, the authors in [Crowther–2011; Langkamp–2011] came up with an alternative arrangement. The proposed design features six propellers arranged in three distinct rotor planes. The capability to vector thrust in a wide set independently from the body attitude enables maneuvering in confined spaces and provides the ability to land and take-off from different attitudes.

In particular, the authors envisaged configurations of this kind to be beneficial for operations where the requirements for translational control authority design outweigh the reduced hovering efficiency compared to standard collinear MRAVs, such as quadrotors. A similar arrangements for the actuators has been chosen also in [Voyles–2012], where the authors considered the interesting force closure grasp property previously defined for fixed-base ground manipulators, and envisaged, still in a preliminary way, the possibility of making the platform resilient to one rotor failure. The dynamic capabilities of aerial vehicles with MDT designs can be further improved, as also realized by the author in [Crowther–2011], by substituting typical uni-directional thrust actuators with bi-directional ones (see Chapt. 3). In this way, it is possible to independently control the body force and torque inside a 6D ball, within the operational conditions of the actuators. We refer to MRAVs belonging to this particular sub-class of MDT platform as Omni-Directional (OD). These vehicles are able to accomplish tasks that are not possible for other aerial systems like, e.g., exert a force /torque in *all* (not only in *many*) directions, in particular from the top, and hovering with every orientation in $\mathrm{SO}(3)$. Other examples of OD MRAVs that have been implemented are given in [Brescianini–2016; Park–2018]. Both platforms developed in these works are octorotors equipped with bi-directional thrusters. In both designs, the change in direction of the force produced by each actuator is generated by inverting the spinning velocity of the rotor, while in [Crowther–2011] this is accomplished by keeping the spinning velocity constant and using variable-pitch propellers to revert the rotor geometry. Both solutions present a main disadvantage. In particular, the former require some delay when crossing a null angular velocity for the rotors, which could induce serious problem to the vehicle stability, if not properly managed. On the other hand, the latter requires additional hardware, which implies a more complicated and expensive design and an additional mass that reduces the available payload. Such problems can be solved by using fixed-pitch uni-directional thrust propellers, at the cost of adding one actuator [Nikou–2015]. Indeed, it has been recently proved that the minimum number of uni-directional actuators needed to generate an OD design is seven [Tognon–2018b].

Another solution to cope with the under-actuation of typical VTOL MRAVs consists in adopting *actively tilting* mechanisms for the actuators, instead of arranging them in a fixed way. This technique was initially proposed in the literature mainly to simplify the mechanical design of helicopters by removing swash-plates, to reduce the energy consumption, and to provide improved rolling/pitching moments [Gress–2002; Salazar Cruz–2005; Kendoul–2006]. In [Long–2013], the authors exploit a fixed ducted-fan for static compensation of the gravity, enhanced with a system of three smaller adjustable-angle ducted-fans to control the roll/pitch moments and provide lateral forces, showing an improvement in the trajectory tracking w.r.t. an under-actuated UDT configuration. A very interesting and distinct work in this panorama has been offered in [Ryll–2012a; Ryll–2015], where the authors design a quadrotor with the actuators independently tilting around their leverage arms. Considering the angle of each arm as an additional control input, the

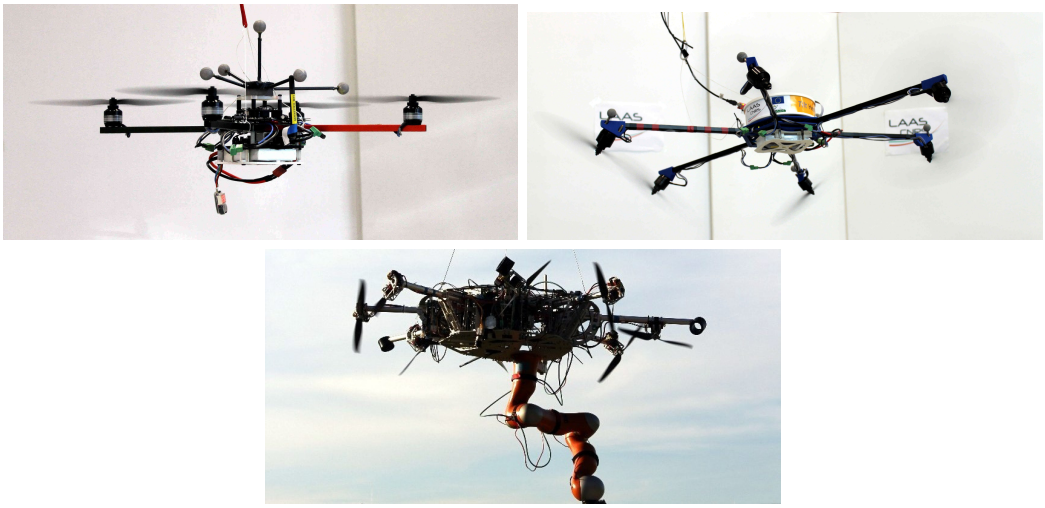


Figure 1.6 – Comparison of different UAVs. *Top left:* a state-of-the-art UDT quadrotor. *Top right:* a MDT hexarotor designed at LAAS-CNRS. *Bottom:* a OD (suspended) octorotor resulting from a collaboration between DLR and LAAS-CNRS.

MRAV achieves *redundancy* in the actuation. Similar works have been proposed also in [Segui Gasco–2014; Oosedo–2015] and in [Rajappa–2015]. In the latter, the theoretical concept of a hexarotor with dual-tilting propellers is outlined, while a inverse-dynamics control algorithm is validated only by means of simulations. Despite the remarkable capabilities shown by these MRAVs, their designs also suffer from some pitfalls. More in detail, they require extra actuation means, mechanical complexity, and weight. Furthermore, they can not, in general, guarantee instantaneous disturbance rejection or fast force exertion since the propellers might have to be re-oriented, which again takes some non-negligible time.

This brief overview of works related to MDT MRAV designs suggests that, in the recent years, several research groups have started to investigate new different solutions to be applied in the field of aerial robotics. Nevertheless, we believe significant gaps were still to be filled at the time the work presented in this thesis was started (and still there is good margin for improvements), in particular in relation to the implementation of aerial robots tailored for APhI tasks. With the goal of bringing the aforementioned concepts and ideas to a more mature level, we addressed the design of both *fixed* and *actively tilting* actuator arrangements for MDT MRAVs, with a special attention to hexarotors, given the good trade-off between maneuverability and payload/energy consumption offered by such platforms. To conclude the comparison between UDT and MDT MRAVs, in Fig. 1.6 we show some of the real robots used for the experimental validation of this work.

Main contribution of the thesis

In this section, we offer a global synopsis of the scientific contribution provided by this thesis. In order to ease and lighten the reading of the manuscript, in Tab. 1.1 and in Tab. 1.2 we reported a schematic description of the main publications and the related achievements that compose the skeleton around which the presented work is built. While the reader interested only in specific parts of this thesis can skip this section and directly refer to the selected chapters, we still highly suggest the reading of the full manuscript for the sake of completeness.

From the previous section, it should be clear that the research fields of aerial robotics and, more in detail, of aerial physical interaction, are remarkably vast. Although today there is a good set of works addressing these topics, numerous problems are still open, that call for additional detailed investigation. Among those challenges, we identified the design and the development of MDT MRAVs tailored for interaction tasks, which involve the exchange of forces and torques between the aerial robots and the surrounding environment. Building on top of the the preliminary bases laid by the state of the art, we conceived and built three MDT hexarotors, each one with specific and unique features. The first one, named "Tilt-Hex", is designed to achieve a partially decoupled tracking of a 6D trajectory in position and orientation, and to perform APhI tasks with a simple rigid tool. Furthermore, the tilt angles of its actuators make the platform resilient to the failure/loss of one propeller, studied in other parallel works. Thanks to a deep theoretical study of the allocation matrix for a meaningful configuration, we conceived and built the FAST-Hex, a MDT platform which allows to actively tilt the actuators in a synchronized fashion with only one additional servo-motor, thus out-performing previous designs in terms of additional complexity and energy consumption. Moreover, we envisioned the concept of the "OT-Hex", a *flying assistant* for cooperative tasks. Ultimately, the actuator arrangements of these aerial robots further influenced the design of the "AeroX", an industrial aerial manipulator from the Center for Advanced Aerospace Technologies (CATEC, Spain). Finally, we took an active part in the design of the "SAM", a suspended octorotor platform with OD capabilities built in the German Aerospace Center (DLR, Germany) with the aim to perform inspection and maintenance tasks. Besides, it is interesting to remark that the design of OD aerial platforms does not only assume relevance in the field of APhI but also for space applications, see e.g., [Roque–2016].

Assessing the need to precisely control the motion of these new aerial robots, exploiting their additional capabilities while accounting for their physical limitations, we made an effort towards the implementation of two different control algorithms. Both strategies take into account, with different level of detail, the fact that unidirectional actuators are used. It is worthwhile to point out that this problem shares some similarities with the control of *cable-driven* robots, where the load can be pulled but not pushed [Gouttefarde–2006; Gouttefarde–2010]. While the first control strategy is Lyapunov-based and exploits a state-feedback to stabilize the vehicle, the second one takes advantage from model predictions of the state evolution

Table 1.1 – Summary of publications in Part II and corresponding contribution list.

Part II - MDT-MRAVs with actuation constraints	
Publication	**Contribution**
[Franchi–2018] **Journal** IEEE T-RO	Full-pose tracking control of Multi-Directional Thrust aerial vehicles with Laterally Bounded Input Force constraints • design of a strategy with prioritized control of the position over the orientation, with theoretical proof of the closed-loop system stability • experimental validation conducted with MDT-MRAVs having fixed-tilted propellers and (addition of this thesis) actively tilting propellers
[Bicego–2019] **Journal** *Submitted to* JINT	Nonlinear Model Predictive Control with actuator constraints for Multi-Rotor Aerial Vehicles • design of a numerical optimization-based motion controller which complies with constraints of min/max force derivatives • definition of a simple but effective identification procedure • extensive experimental validation conducted with different robots
[Ryll–2016] **Conference** IEEE/RSJ IROS 2016	Modeling and control of a convertible MDT MRAV prototype • completely novel design allowing the FAST-Hex to re-orient its actuators in a synchronized way, thanks to one additional servo-motor • numerical simulations of the transition between UDT and MDT
[Morbidi–2018] **Conference** IEEE/RSJ IROS 2018	Study of the actuation properties of a Multi-Directional Thrust hexarotor with dual-tilting propellers • deep analysis of the allocation matrix for a meaningful configuration, allowing for the determination on non-trivial singularities
[Sarkisov–2019] **Conference** IEEE/RSJ IROS 2019	Development of a cable-Suspended Aerial Manipulator • design of a novel Omni-Directional MRAV prototype • integration of industrial manipulator to perform interaction tasks • preliminary experimental and numerical results assessing stability

in a nonlinear Model Predictive Control (MPC) fashion. Both methods guarantee general applicability to the class of MDT MRAVs, which encompasses also UDT vehicles, with actuation constraints. An extensive experimental and numerical validation campaign was conducted to validate the control approaches.

In the view of addressing APhI applications, we further extended the first motion control strategy in order to manage the robot interaction with the environment. Exploiting results available from the literature on ground manipulators, we enhanced the framework with a model-based wrench estimator to retrieve an estimation of the contact wrench, and an admittance filter to shape the compliant behavior of the robot at will. Furthermore, the hard work on the system integration of these tools allowed to set up challenging demonstrations, e.g., the lifting of long loads like beams, and the contact-based inspection of a real metallic pipe using an Eddy Current (EC) sensor. The first task was accomplished in two configurations, i.e., with a single aerial robot and also with two heterogeneous robots. In the second case, the aerial robot interacts with a ground one. On the other hand, the second task was performed by an aerial robot with a simple rigid tool first, and with a 2-DoF lightweight arm after. This let us demonstrate the practicability through

Table 1.2 – Summary of publications in Part III and corresponding contribution list.

Part III - Aerial Physical Interaction applications	
Publication	**Contribution**
[Staub–2018b] **Journal** IEEE RAM	Aerial-Ground Manipulator System with tele-operation capabilities • paradigm of heterogeneous robots cooperation with human in the loop • integration of motion and interaction control, use of force-feedback • experimental validation with an industrial manipulator
[Ryll–2019] **Journal** IJRR	6D interaction control taking advantage of a MDT aerial robot • novel paradigm for physical interactive tasks in aerial robotics • integration of motion and interaction control, use of wrench estimator • experimental validation through various challenging tests
[Tognon–2019] **Journal** IEEE RA-L	Aerial manipulator system for push-and-slide contact operations • hybrid position/force controller for a redundant aerial manipulator • integration of design, control and motion planning • experimental validation with integration of contact inspection sensors
[Staub–2018a] **Conference** IEEE ICRA 2018	Aerial manipulator system for assembly/decommissioning operations • novel conception of a flying assistant for cooperative manipulation • tailored design for bar-lifting tasks and control of the interaction forces • experimental validation integrating a passive arm with grippers

real experiments, of a completely new aerial physical interaction paradigm, called the 6D flying end-effector. We believe that this paradigm will pave the way to novel aerial system concepts which outperform currently adopted solutions for aerial manipulation and physical interaction in terms of capability, reliability, complexity and costs, as also suggested by the subsequent industrial validation.

In conclusion, in order to visually appreciate the experimental results achieved in this work, we highly suggest the reader referring to the related multimedia contents, whose link list is drawn up at the beginning of the document.

Organization of the thesis

The goal of this section is to serve as a guide for the reader, illustrating the organization of the thesis and summarizing the content of each chapter. The written description proposed in the following is supported by a self-descriptive graphical overview of the manuscript, portrayed in Fig. 1.7. With the aim of facilitating the reading of this thesis, we divided it into four parts.

Part I is meant to provide the reader with some preliminary concepts we believe to be convenient for a clear comprehension of the problems this thesis addressed and tried to solve, and to fruitfully contextualize this work in the wide panorama offered by aerial robotics and aerial physical interaction.

Part II deals with the description of modeling techniques and motion control strategies for MDT MRAVs. In particular, significant stress is put towards the characterization and the compliance with the physical actuation limits, which are of utmost importance for the preservation of the system stability during the accomplishment of challenging tasks. Furthermore, it offers some hints on the novel robot designs that we proposed to address in a meaningful way particular tasks of aerial robotics. Finally, it offers the detailed results of the extensive validations of the control algorithms mainly in the free-flight case, i.e., in relation to applications where no contact between the aerial vehicle and the surrounding environment is expected.

Part III presents an extension of the methodologies developed in Part II in order to achieve a stable control of the system also during the interaction phase, i.e., when the contact between the aerial robot and the environment is not only expected, but also actively pursued. The conceptual jump between Part II and Part III mirrors the transition from AVs to ARs. Also in this case, the theoretical analysis is supported with real experimental results. In particular, the accomplishment of aerial physical interaction tasks motivated by use-case applications of real industrial interest is demonstrated. Notably, the lifting and the manipulation of long objects, both with one and multiple robot agents, and the contact inspection of a metallic pipe are fulfilled. It is worth mentioning that these achievements were the outcome of successful collaborations that promoted the enlargement of the spectrum of topics and methods addressed and investigated in this thesis.

Part IV highlights the development of high Technology Readiness Level (TRL) aerial manipulators, thus demonstrating the impact on the industrial sector of the achieved results. Final discussions on the presented work and future ones conclude the manuscript.

In the following, we shall break down the former partition and provide a concise description of the content of each chapter. As far as Part I is concerned:

Chapt. 2 provides a detailed and wide-spectrum overview of the state of the art on MPC strategies for MRAVs. At the best of our knowledge, this is the first time such a classification is proposed.

Chapt. 3 recalls in a synthetic way the mathematical methodologies used as the groundwork for the theoretical analysis of MDT MRAVs with actuation constraints provided in Part II. In particular, we revise the two most used modeling formalisms for the description of the dynamics of a rigid body, i.e., the Newton-Euler and the Lagrangian ones, with a detailed analysis of the wrench generation of a MRAV. Furthermore, hints on the Inverse Dynamics control are given, highlighting the shortcomings of such technique in the presence of actuator constraints.

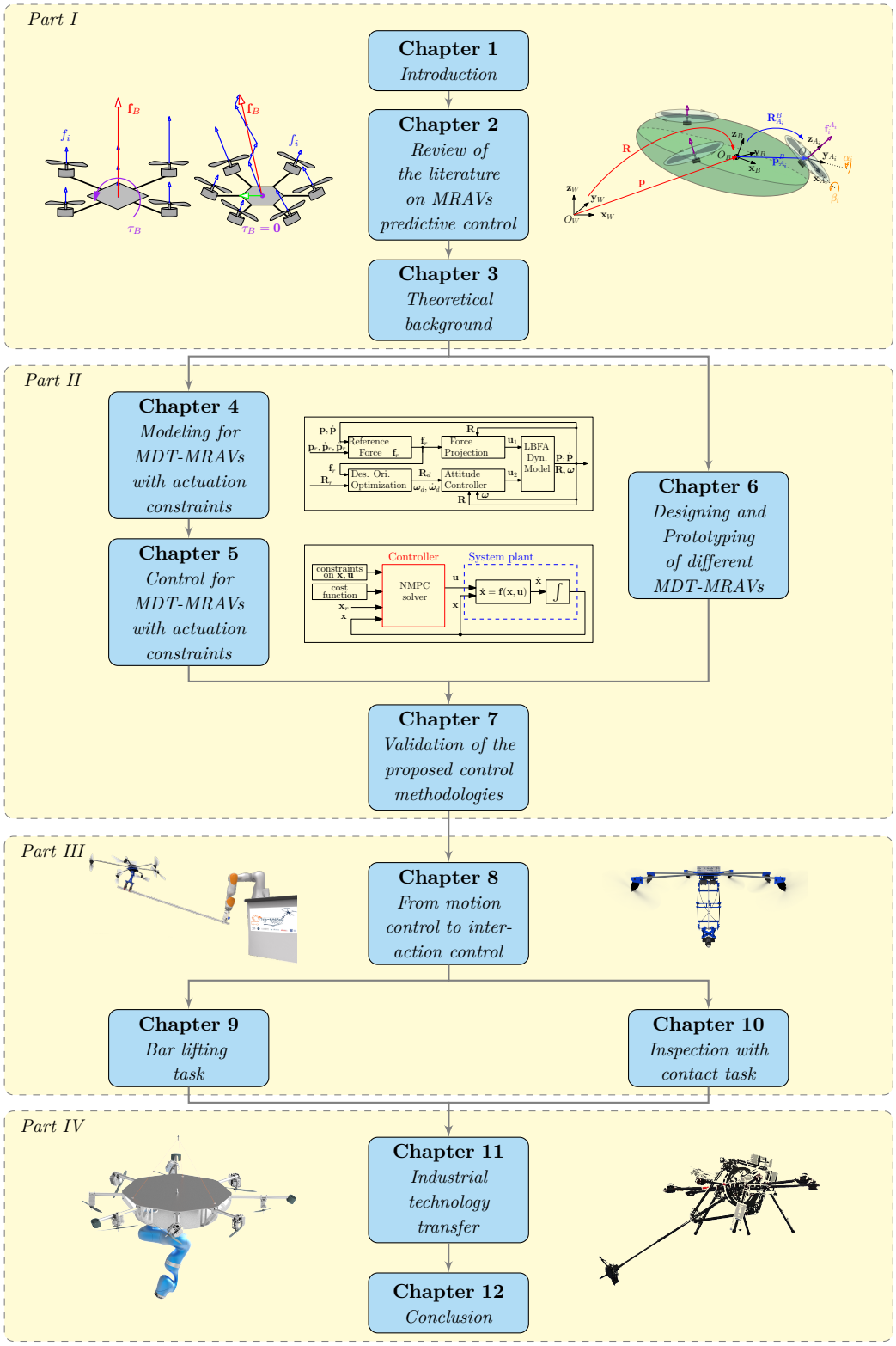


Figure 1.7 – Graphical representation of the thesis organization.

Regarding Part II:

Chapt. 4 proposes two physical models, both based on the Newton-Euler formalism, of a MDT MRAV with actuation constraints. While the first perform a high-level abstraction of the vehicle dynamics, the second aims at providing a detailed description of more platform-specific limitations.

Chapt. 5 presents two control algorithms for the tracking of 6D pose trajectories, one based on each of the models defined in Chapt. 4. While the former consists in a static feedback Lyapunov-based control strategy, the latter implements an optimization-based method which exploits the prediction of the system behaviors along a future time windows, obtained thanks to the detailed model.

Chapt. 6 gives some hints on the design of three MDT hexarotor prototypes conceived and built in-house at LAAS–CNRS, each aimed to accomplish a well-defined task. Moreover, it concisely describes the design of an octorotor aerial manipulator capable of exerting an OD wrench, developed in collaboration with partners from the German Aerospace Center (DLR).

Chapt. 7 offers the results of the extensive experimental validation of the control strategies devised in Chapt. 5, carried out exploiting the hexarotors outlined in Chapt. 6. Furthermore, additional realistic simulations are performed in order to show the valuable capabilities of the controllers in different scenarios.

Regarding Part III:

Chapt. 8 contains the theoretical foundations to pass from motion to interaction control. In particular, an extension of the static feedback control strategy detailed in Chapt. 5 is obtained thanks to an admittance-filter paradigm.

Chapt. 9 shows the accomplishment of the challenging task of lifting long objects like, e.g., metallic beams, by means of a tailored MDT hexarotor, whose design is shown in Chapt. 6. Furthermore, an extension of the application is shown, thanks to the development of a system composed of heterogeneous robots.

Chapt. 10 analyzes the attractive application of contact inspection of surfaces by means of an aerial manipulator. After the fulfillment of this task with a simple rigid tool, the operation is enhanced with the integration of a real EC sensor for the detection of the welding on a metallic pipe.

Regarding Part IV:

Chapt. 11 focuses the attention on higher TRL applications directly and indirectly originated from the study presented in the previous chapters. In particular, it highlights the impact on the industry of the work done in this thesis.

Chapt. 12 completes the thesis by briefly summarizing the presented scientific content and considering applications and extensions for future work.

Although presented with an increasing level of detail, with a bottom-up approach, the four parts are quite independent to each other. Therefore, in the case of lack of time, the reader can focus her/his attention to one specific part. In particular, the aerial physical interaction applications of Chapt. 9 and Chapt. 10 can be read independently. The goal of this chapter is to give a concise yet exhaustive overview of this thesis. Possibly, by now we succeeded in getting the reader's interest towards a particular part or chapter of the manuscript.

Ultimately, we highlight that the work presented in this thesis has been completely carried out within the scope of the H2020 European project AeroArms[14], see [Ollero–2018], which gathers several partners from universities, scientific laboratories and also companies. The goal of AeroArms is to develop aerial robotic systems with advanced manipulation capabilities to be applied in industrial inspection and maintenance. Examples of such operations are the installation of sensors in sensible sites, the deployment of smaller crawler robots (developed by one of the AeroArms partners), or the direct contact inspection by means of an End-Effector (EE) endowed with a specific sensor. These tasks to be accomplished require particular aerial robots capable of acquiring measurements in contact with the surface of interest, exchanging forces with it while maintaining a stable flight. The work shown in this thesis, conducted during the last three years, perfectly fits with the aforementioned needs and was developed with this goal clear in mind. In particular, the presented MDT MRAV designs reply to the call for aerial robots capable to exert forces and torques in a decoupled way and to robustly compensate for external disturbances in different directions. In a complementary fashion, the discussed motion and interaction control strategies aim to reliably drive the robot both in the contact-less and in the in-contact phase, in a seamless way. Furthermore, the bar lifting task described in Chapt. 9 represents a proof-of-concept to demonstrate in a simple yet effective way the robotic system capability to perform assembly and decommissioning operations. Additionally, the contact inspection of sloped surfaces and of a real metallic pipe with both a fixed tool and a two DoF lightweight arm, as shown in Chapt. 10, proved the effectiveness of the proposed solutions. Finally, the implementation of high TRL aerial manipulators in the AeroArms project suggests the industrial interest towards the addressed problems and the directions for future improvements that can be achieved building on top of the presented contributions.

Publication note

This thesis grounds on five journal publications (one of which is still under review) and four conference papers published on major international congresses on robotics research. The list of publication is reported in Tab. 1.1 and in Tab. 1.2, divided w.r.t. the most influenced part of the manuscript. In particular, regarding [Staub–2018a] my colleague N. Staub and I participated in an equal way to the development of that research work. Furthermore, few poster contributions were published in

[14] `https://aeroarms-project.eu/`

different workshops at IEEE/RSJ IROS 2018 and IEEE ICRA 2019. Additional conference papers [Staub–2017; Ryll–2018; Furci–2018] have been co-authored, but not reported in this thesis. As far as [Staub–2017] is concerned, it represented the starting point of the study on the MAGMaS system presented in Sec. 9.2 but exploited a UDT MRAV. For this reason, we decided to leave it out from the discussion. On the other hand, [Ryll–2018; Furci–2018] investigated fascinating research directions in relation to MDT MRAVs. Nevertheless, we believe the level of maturity of these works to have substantial margin of improvement.

Chapter 2

Review of the literature on MRAVs predictive control

"As I review the events of my past life I realize how subtle are the influences that shape our destinies."
Nikola Tesla

The literature of aerial robotics encompasses many different control strategies for Multi-Rotor Aerial Vehicles (MRAVs) designed by researchers and engineers in order to obtain an accurate trajectory tracking for these vehicles, which is the first requirement to perform any operation. Considered the vast panorama, in this chapter we do not aim to present a detailed taxonomy on the control of aerial robots, but rather to introduce the reader to the topic of *predictive* control. Indeed, we make some effort towards a meaningful classification of the sub-class of Model Predictive Control (MPC) methods proposed for these systems.

The most common controllers implemented for MRAVs are the ones performing the well-known Proportional, Integrative and Derivative (PID) actions w.r.t. the state error, either linearizing the system around the hovering condition, as in [Michael–2010], or performing feedback linearization, as in [Lee–2010; Goodarzi–2013]. Other MRAV control methods exploit different/additional techniques which comprehend, but are not limited to, geometric control [Bullo–2005], adaptive control [Dai–2014; Zhao–2014], back-stepping and sliding-mode [Bouabdallah–2005; Lee–2009], and reinforcement learning [BouAmmar–2010; Schoellig–2012]. The interested reader is addressed to [Hua–2013] to a good overview of several control strategies for under-actuated Uni-Directional Thrust (UDT) MRAV, e.g., [Fantoni–2002; Hamel–2002] both linear and nonlinear, while an extension of [Lee–2010] for the Multi-Directional Thrust (MDT) case is proposed in the first part of Chapt. 5, in a similar way to what also done in [Hua–2015b]. Other works addressing this problem have been [Kendoul–2006; Romero–2007; Pounds–2010a; Convens–2017]. Further discussions related to this topic will be offered throughout the manuscript.

The similarity shared by all these controllers is the fact of being *reactive*, meaning that the control input to the system at a certain time instant is computed in relation to the state error at the same time or, at most, in a past interval. Such reactive nature generates an intrinsic delay. Furthermore, there is typically no guarantee that the input or the state will satisfy the physical constraints imposed by the real system unless this compliance being enforced in a very conservative way.

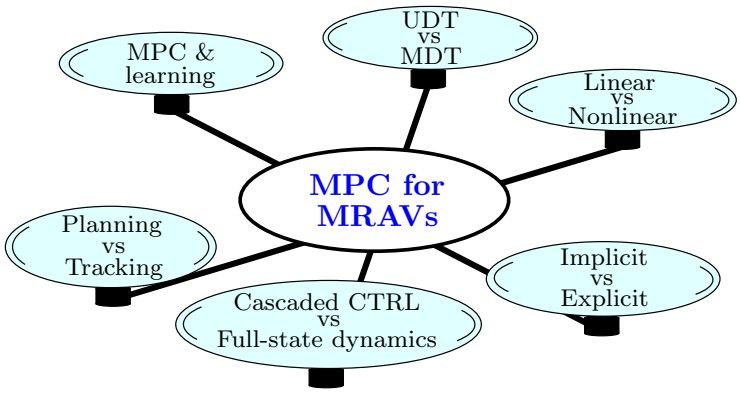


Figure 2.1 – Concept map outlining the fields used for the classification of the presented MPC methods for Multi-Rotor Aerial Vehicles.

In the last few years, several controllers have been designed based on the prediction horizon optimization. In this case, the control input is chosen in accordance to the solution of an Optimal Control Problem (OCP) inside a future time window, along which the evolution of the system can be simulated thanks to a model of the plant. Referring to the Linear Quadratic Regulator (LQR), the OCP can be solved offline in an explicit way but the model of the system must be linear, the problem quadratic and the physical limits are not yet ensured to be satisfied. A very effective technique to deal with such problems is the MPC, a model-based control strategy that originates from the industrial process control. Currently, the method is used for a variety of applications in the field of control engineering, ranging from chemical to automotive ones. The strength of MPC, apart from its knowledge of the dynamic model of the system and of a reference trajectory or a way-point sequence along the prediction horizon, is found in the iterative resolution of the OCP, which creates the possibility to find the optimal control solution for the system, i.e., the one that minimizes a certain user-defined cost function, while handling dynamic constraints related to the state and to the inputs imposed by the physical system. Furthermore, the MPC is constantly updated with new state measurements available from the system, in order to mitigate possible deviations of the model evolution from the real dynamics. Despite the fact that these benefits come with an increased computational demand compared to reactive control algorithms, the significant increase of computational power of Central Processing Units (CPUs) of recent years together with the development of new optimization techniques have made *predictive* controllers, in particular MPC, a concrete and appealing possibility to control also systems with fast dynamics, such as MRAVs.

In the following, we propose a classification of some of the latest and most influential research works related to MPC for MRAVs. The taxonomy is built around six main axes, as it can be appreciated from Fig. 2.1. In order to meaningfully highlight the peculiarities of our approach, introduced in Chapt. 5, at the end of each discussion we present a brief explanation of our objectives. To conclude the chapter, we present the contributions brought to state of the art by our work.

Classification of MPC methods for MRAVs

The flourishing scientific research of recent years on MPC in the context of aerial robotics has led to a rich set of contributions on the problem perspective, the numerical solution methods and the applications. The goal of this section is to make a taxonomy of some of the most representative breakthroughs in such domain, in order to properly place our work inside this rich panorama.

Let us first briefly introduce the mathematical form of a classical MPC problem. Given a continuous-time system, we can model its dynamics in the standard form

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}, \mathbf{u}), \tag{2.1}$$

where $\mathbf{x} \in \mathbb{R}^{n_x}$ and $\mathbf{u} \in \mathbb{R}^{n_u}$ are the state and the input vectors of the model, respectively. At time $t = t_0$, the input to the system computed by the predictive controller is the one that solves the following OCP along the future time window $[t_0\, t_f]$, called the *prediction horizon*

$$\mathbf{u}^* = \arg\min_{\mathbf{x},\mathbf{u}} \quad \mathbf{l}_f(\mathbf{x}(t_f)) + \int_{t_0}^{t_f} \mathbf{l}(\mathbf{x}(t), \mathbf{u}(t))\ dt \tag{2.2}$$

$$\text{s.t.} \quad \mathbf{r}(\mathbf{x}, \mathbf{u}) = \mathbf{0}, \tag{2.3}$$

$$\mathbf{s}(\mathbf{x}, \mathbf{u}) \leq \mathbf{0}. \tag{2.4}$$

In other words, the MPC solver computes the input that minimizes the cost represented by the sum of the integral of the function $\mathbf{l}(\mathbf{x}, \mathbf{u})$ along the horizon and a terminal cost $\mathbf{l}_f(\mathbf{x}_f)$, subject to the respect of some constraints, embodied by the functions $\mathbf{r}(\mathbf{x}, \mathbf{u})$ and $\mathbf{s}(\mathbf{x}, \mathbf{u})$. In particular, $\mathbf{r}$ must contain the constraint of the model dynamics, which has to evolve following (2.1) and according to the measurements available from the real system. On the other hand, $\mathbf{s}$ is used to restrain the set of the possible values that the state and the input can take. Such problem is iteratively solved at each control sample time.

In order to properly classify the set of contributions that will be outlined in the following, we made an effort towards the definition of some objective criteria, which are summarized in the concept map of Fig. 2.1. In particular, we focused our attention on the problem formulations and their resolution approaches, the control methodologies, the modeling techniques and the specific aerial robot platforms employed for the validation of the different MPC algorithms. The main results for each of these fields have been gathered in Tab. 2.1, to which the interested reader is referred for a condensed picture of the detailed discussion presented below.

Planning vs tracking

As far as the perspective is concerned, a first classification can be made regarding the kind of problem that researchers have been addressing. In particular, two main classes can be highlighted: the *trajectory planning/generation*, i.e., the problem

of generating a feasible trajectory w.r.t. the physical limits of the system, and the *trajectory tracking*, which deals with chasing the trajectory exploiting a certain control law, still complying with the system constraints.

Regarding the former, an interesting work combining the Rapidly-exploring Random Tree (RRT) technique with MPC has been presented in [Lin–2016]. The authors used the RRT algorithm for the path searching, that is the quest of waypoint connections between two specified points (the start and the end), while they employed the MPC for the trajectory fitting, namely the fit of the obtained succession of points with a smooth curve parametrized in time. Thanks to its capability of simulating the future evolution of the system model along the prediction horizon, such MPC framework is able to generate constraint-respecting trajectories for quadrotors, which can be tracked by an independent controller. Another relevant contribution in the planning is given by the authors in [Baca–2017], who propose to use MPC for generating a set of desired states of the MRAV in position, velocity, and acceleration that are handed out to a SO(3) state feedback controller as a reference. In this way, the controller is provided with the necessary feed-forward action to follow the known future path. In this case, such framework is deployed on a hexarotor in order to perform an autonomous landing on a moving platform, using only onboard sensing and computation. The approach is validated with experimental results carried out at the 2017 MBZIRC[1] competition. A distinguished contribution regarding the trajectory generation is provided by [Mueller–2013a], which is validated with the relevant experiment of a quadrotor hitting a tossed ball with an embedded racket. The system dynamics is assumed decoupled along three orthogonal axes and each sub-system is modeled independently from the others as a triple integrator with jerk as input. Thanks to this simplified model, the constraints on the total thrust and angular velocity are translated into limitations on the linear velocity and acceleration. In the subsequent work [Mueller–2013b], the authors improved the computational time of the algorithm by removing the constraints from the optimization and deriving sufficient criteria in order to test online the trajectories feasibility with respect to total thrust and the angular velocity. An extension of these two contributions is finally offered by the same authors in [Mueller–2015], where this time a quadrotor, installed with a small net, is assigned to catch a ball thrown by an operator. Results on trajectory generation for quadrotors are also presented in [Liu–2015], where the approach is validated both with MATLAB and ROS/Gazebo physical simulations and also with real experiments, using MPC in combination with a robust controller for systems subject to bounded disturbances. An interesting application exploiting MPC to compute feasible trajectories for a quadrotor, while maximizing the visibility of a point of interest inside the field of view (FOV) of an onboard camera is presented in [Falanga–2018]. In this case, the MPC instance can deal with both action and perception objectives.

Although most of MPC applications fall in the scope of trajectory planning, this predictive control strategy has also been used like a tracker. An example of

[1] `https://www.mbzirc.com/competition/2017`

Table 2.1 – Condensed table of the presented works on MPC applied to AVs.

Symbol	Description
Problem Formulation	
l	linear
nl	nonlinear
lqr	linear quadratic regulator
h	hybrid
ad	axes-decoupled

Problem Resolution	
i	implicit
e	explicit
Validation	
sim	simulations
exp	experiments

Problem targeted	
p.c.	position control
a.c.	attitude control
f.c.	force control
ps.c.	partial-state control
fs.c.	full-state control
t.p.	trajectory planning

Implementation details	
on	on-board the vehicle
off	off-board the vehicle
m.b.	move block technique
t_H	prediction horizon length
N	prediction horizon steps
d.h.	diminishing horizon
T_{MPC}	sample time

MPC paper	state & input cost function & constraints	l / nl / lqr i / e	sim / exp use	platform horizon
[Alexis–2014] [Alexis–2016b]	$\mathbf{x} = [\mathbf{p}^\top \dot{\mathbf{p}}^\top \phi\ \theta]^\top$ & $\mathbf{u} = [\phi^r\ \theta^r\ T]^\top$ $\max_w \max_j \|\|\mathbf{C}\mathbf{x}_{k+j\|k}\|\|_\infty$ & $\mathbf{u}_{k+j\|k} \in \mathcal{U}, \mathbf{x}_{k+j\|k} \in \mathcal{X}, \mathbf{w}_{k+j\|k} \in \mathcal{W}$	l (ad) e	exp p.c. (on)	quadrotor, tilting trirotor $t_H = 0.48\text{s}, 0.6\text{s}$ / $N = 6$
[Baca–2016]	$\mathbf{x}_k = [\mathbf{p}^\top \dot{\mathbf{p}}^\top \ddot{\mathbf{p}}^\top \ddot{\mathbf{p}}_u^\top \ddot{\mathbf{p}}_d^\top]_{x,y}^\top$ & $\mathbf{u}_k = [\phi^r\ \theta^r]^\top$ $\frac{1}{2}\sum_{i=1}^{m-1}(\mathbf{e}_{x,i}^T\mathbf{Q}\mathbf{e}_{x,i} + \mathbf{u}_i^T\mathbf{R}\mathbf{u}_i) + \frac{1}{2}\mathbf{e}_m^T\mathbf{S}\mathbf{e}_m$ & unconstrained	l (ad) e	exp p.c. (on)	quadrotor $t_H = 2.2\text{s}$ / $N = 200$ (m.b.)
[Baca–2017]	$\mathbf{x}_k = [\mathbf{p}^\top \dot{\mathbf{p}}^\top \ddot{\mathbf{p}}^\top \ddot{\mathbf{p}}_u^\top\ , \ddot{\mathbf{p}}_d^\top]_{x,y}^\top$ & $\mathbf{u}_k = [\phi^r\ \theta^r]^\top$ $\frac{1}{2}\sum_{i=1}^{m-1}(\mathbf{e}_{x,i}^T\mathbf{Q}\mathbf{e}_{x,i} + \mathbf{u}_i^T\mathbf{R}\mathbf{u}_i) + \frac{1}{2}\mathbf{e}_m^T\mathbf{S}\mathbf{e}_m$ & $\mathbf{x}_i \in \mathcal{X}, \mathbf{u}_i \in \mathcal{U}$	l (ad) i	exp t.p. (on)	hexarotor $t_H = 8\text{s}$ / $N = 800$ (m.b.)
[Bangura–2014a]	$\mathbf{x} = [\mathbf{p}^\top \dot{\mathbf{p}}^\top F]^\top$ & $\mathbf{u} = [\boldsymbol{\omega}^\top \dot{T}]^\top$ $\sum_{i=0}^{N-1}(\mathbf{e}_{x,k+i\|k}^T\mathbf{Q}\mathbf{e}_{x,k+i\|k} + \mathbf{u}_{k+i\|k}^T\mathbf{R}\mathbf{u}_{k+i\|k})$ & unconstrained	l e	exp p.c. (on)	quadrotor $t_H \approx 0.31\text{s}$ / $N = 5$
[Bemporad–2009]	$\mathbf{x} = [\mathbf{p}^\top \dot{\mathbf{p}}^\top \boldsymbol{\Theta}^\top \dot{\boldsymbol{\Theta}}^\top \int e_z dt]^\top$ & $\mathbf{u} = [\boldsymbol{\tau}^\top T]^\top$ $\sum_{i=0}^{N-1}\sum_{j=1}^{n_y}(\mathbf{e}_{x_j,i}^T\mathbf{Q}\mathbf{e}_{x_j,i} + \boldsymbol{\Delta}\mathbf{u}_{x_j,i}^T\mathbf{R}\boldsymbol{\Delta}\mathbf{u}_{x_j,i})$ & $\mathbf{u} \in \mathcal{U}, \mathbf{x} \in \mathcal{X}$	l / h i	sim p.c. / t.p. (off)	quadrotor $t_H \approx 1.5\text{s}$ / $N = 20$
[Bouffard–2012]	$\mathbf{x} = [x_1\ , \dot{x}_1\ \theta_1\ \dot{\theta}_1\ x_2\ \dot{x}_2\ \theta_1\ \dot{\theta}_1\ x_3\ \dot{x}_3]^\top$ & $\mathbf{u} = [\theta_1^r\ \theta_2^r\ T^r]^\top$ $\sum_{j=0}^{N-1}(\mathbf{e}_{x,j}^T\mathbf{Q}\mathbf{e}_{x,j} + \mathbf{e}_{u,j}^T\mathbf{R}\mathbf{e}_{u,j}) + \mathbf{e}_{x,N}^T\mathbf{P}\mathbf{e}_{x,N}$ & $\mathbf{u} \in \mathcal{U}, \mathbf{x} \in \mathcal{X}$	l i	exp p.c. (on)	quadrotor (+ net) $t_H \approx 0.38\text{s}$ / $N = 15$
[Darivianakis–2014]	$\mathbf{x} = [\mathbf{p}^\top \dot{\mathbf{p}}^\top \phi\ \theta]^\top$ & $\mathbf{u} = [\phi^r\ \theta^r\ T]^\top$ $\sum_{k=0}^{N-1}(\mathbf{e}_{x,k}^T\mathbf{Q}\mathbf{e}_{x,k} + \mathbf{u}_k^T\mathbf{R}\mathbf{u}_k) + \frac{1}{2}\mathbf{e}_N^T\mathbf{P}\mathbf{e}_N$ & $\mathbf{x}_k \in \mathcal{X}, \mathbf{u}_k \in \mathcal{U}$	l (ad) e	exp p.c. / f.c. (on)	quadrotor (+ writing tool) $t_H = 0.2\text{s}$ / $N = 10$
[Falanga–2018]	$\mathbf{x} = [\mathbf{p}^\top \dot{\mathbf{p}}^\top \mathbf{q}^\top]^\top$ & $\mathbf{u} = [\boldsymbol{\omega}^\top T]^\top$ $\sum_{i=1}^{N}([\mathbf{e}_{x,i}^T, \mathbf{e}_{z,i}^T] diag(\mathbf{Q}_{x,i}, \mathbf{Q}_{p,i})[\mathbf{e}_{x,i}^T \mathbf{e}_{z,i}^T]^T)$ & $\mathbf{x} \in \mathcal{X}, \mathbf{u} \in \mathcal{U}$	nl i	exp t.p. (on)	quadrotor $t_H = 2\text{s}$ / $N = 20$
[Foehn–2018]	$\mathbf{x} = [\mathbf{p}^\top \dot{\mathbf{p}}^\top \mathbf{q}^\top]^\top$ & $\mathbf{u} = [\boldsymbol{\omega}^\top T]^\top$ $\int_0^\infty \mathbf{e}_x^T(t)\mathbf{Q}\mathbf{e}_x(t) + \mathbf{e}_u^T(t)\mathbf{R}\mathbf{e}_u(t)\ dt$ & unconstrained	lqr e	exp p.c. (on)	quadrotor $t_H = \infty$ / $T_{\text{MPC}} = 0.1\text{s}$
[Geisert–2016]	$\mathbf{x} = [\mathbf{p}^\top \boldsymbol{\Theta}^\top \dot{\mathbf{p}}^\top \boldsymbol{\omega}^\top \bar{w}_i|_{i=1}^4]^\top$ & $\mathbf{u} = [\dot{\bar{w}}_i|_{i=1}^4]^\top$ $\sum_{i=1}^{N-1}(\mathbf{e}_{p,i}^T\mathbf{Q}\mathbf{e}_{p,i}) + \boldsymbol{\omega}_N^T C \boldsymbol{\omega}_N$ & $\mathbf{u} \in \mathcal{U}$	nl i	sim fs.c. (off)	quadrotor (+ slung load) $t_H = 8\text{s}$ / $N = 60$
[Hofer–2016]	$\mathbf{x} = [\mathbf{p}^\top \boldsymbol{\Theta}^\top \dot{\mathbf{p}}^\top \boldsymbol{\omega}^\top]^\top$ & $\mathbf{u} = [T_{y,i}|_{i=1}^3, T_z]^\top$ $\frac{1}{2}\int_0^\infty \mathbf{x}^T(t)\mathbf{Q}\mathbf{x}(t) + \mathbf{u}^T(t)\mathbf{R}\mathbf{u}(t)\ dt$ & $\mathbf{u} \in \mathcal{U}$	l i	sim & exp fs.c. (on)	ducted-fans + flaps $t_H = \infty$ / $T_{\text{MPC}} = 0.02\text{s}$
[Kamel–2015]	$\mathbf{x} = [\{\mathbf{R}_B^W\}_{ij}, \boldsymbol{\omega}^\top]^\top$ & $\mathbf{u} = [f_i|_{i=1}^6]^\top$ $\int_t^{t+T_h} \mathbf{e}_x^T(t)\mathbf{Q}_x\mathbf{e}_x(t) + \mathbf{e}_u^T(t)\mathbf{Q}_u\mathbf{e}_u(t)dt$ & $\mathbf{x} \in \mathcal{X}, \mathbf{u} \in \mathcal{U}$	nl i	sim & exp a.c. (on)	hexarotor $t_H = 50\text{ms}$ / $N = 10$
[Kamel–2017a]	$\mathbf{x} = [\mathbf{p}^\top \dot{\mathbf{p}}^\top\ {}_I\phi\ {}_I\theta]^\top$ & $\mathbf{u} = [{}_I\phi^r\ {}_I\theta^r\ T]^\top$ $\sum_{i=1}^{N-1}(\mathbf{e}_{x,i}^T\mathbf{Q}\mathbf{e}_{x,i} + \mathbf{e}_{u,i}^T\mathbf{R}\mathbf{e}_{u,i}) + \mathbf{e}_{x,N}^T\mathbf{Q}\mathbf{e}_{x,N}$ & $\mathbf{u} \in \mathcal{U}$	l vs nl i	exp p.c. (on)	hexarotor $t_H = 2\text{s}$ / $N = 20$
[Kocer–2018]	$\mathbf{x} = [\mathbf{p}_W^\top \dot{\mathbf{p}}_B^\top]^\top$ & $\mathbf{u} = [\phi, \theta, \psi, F_z]$ $\int_t^{t+N} \mathbf{e}_x^T(t)\mathbf{Q}\mathbf{e}_x(t) + \mathbf{e}_u^T(t)\mathbf{R}\mathbf{e}_u(t)\ dt$ & $\mathbf{u} \in \mathcal{U}$	nl i	exp p.c. (on)	quadrotor (close to ceiling) $t_H = 0.4\text{s}$ / $N = 40$
[Ligthart–2017]	$\mathbf{x} = [\phi\ \theta\ \dot{\phi}\ \dot{\theta}\ \dot{\psi}]^\top$ & $\mathbf{u} = [f_i|_{i=1}^6]^\top$ $\sum_{i=1}^{N}(\mathbf{e}_{x,i}^T\mathbf{Q}\mathbf{e}_{x,i} + \boldsymbol{\Delta}_{u,i}^T\mathbf{R}_1\boldsymbol{\Delta}_{u,i} + \mathbf{e}_{u,i}^T\mathbf{R}_2\mathbf{e}_{u,i})$ & unconstrained	l e	exp a.c. (on)	hexarotor $t_H = 4\text{s}$ / $N = 500$
[Lin–2016]	$\mathbf{x} = [\phi\ \theta\ \psi\ \dot{\phi}\ \dot{\theta}\ \dot{\psi}]^\top$ & $\mathbf{u} = [f_i|_{i=1}^4]^\top$ $\sum_{i=1}^{N-1}(\mathbf{e}_{x,i}^T\mathbf{Q}\mathbf{e}_{x,i} + \boldsymbol{\Delta}_{u,i}^T\mathbf{R}\boldsymbol{\Delta}_{u,i}) + \mathbf{e}_{x,N}^T\mathbf{Q}\mathbf{e}_{x,N}$ & $\mathbf{x} \in \mathcal{X}, \mathbf{u} \in \mathcal{U}$	l i	sim t.p. (off)	quadrotor $t_H = 0.4\text{s}$ / $N = 20$
[Liu–2012]	$\mathbf{x} = [\mathbf{p}_W^\top \dot{\mathbf{p}}_B^\top \boldsymbol{\Theta}^\top \boldsymbol{\omega}^\top]^\top$ & $\mathbf{u} = [\delta_{lat,lon,ped,col}]^\top$ $\frac{1}{2}\int_0^T \mathbf{e}_x^T(t+\tau)\mathbf{Q}\mathbf{e}_x(t+\tau)\ d\tau$ & unconstrained	nl e	sim / exp fs.c (on)	helicopter $t_H = 4\text{s}$ / $T_{\text{MPC}} = 0.2\text{s}$
[Liu–2015]	$\mathbf{x} = [\mathbf{p}^\top \dot{\mathbf{p}}^\top \ddot{\mathbf{p}}^\top]^\top$ & $\mathbf{u} = [\dddot{\mathbf{p}}^\top]$ $\sum_{k=1}^{N}(\mathbf{u}[k]^T\mathbf{P}\mathbf{u}[k] + \mathbf{e}_x[k]^T\mathbf{L}_s\mathbf{e}_x[k])$ & $\mathbf{x} \in \mathcal{X}, \mathbf{u} \in \mathcal{U}$	l (ad) i	sim / exp t.p. (on)	quadrotor $t_H = 1\text{s}$ / $N = 50$
[Mueller–2013a]	$\mathbf{x} = [\mathbf{p}^\top \dot{\mathbf{p}}^\top \ddot{\mathbf{p}}^\top]^\top$ & $\mathbf{u} = \dddot{\mathbf{p}}$ $\int_0^T (\dddot{x}(t))_j^2 dt,\ j = 1,2,3$ & $\mathbf{x} \in \mathcal{X}, \mathbf{u} \in \mathcal{U}$	l (ad) i	exp t.p. (off)	quadrotor (+ racket) d.h. / N from 10 to 200
[Mueller–2013b]	$\mathbf{x} = [\mathbf{p}^\top \dot{\mathbf{p}}^\top \ddot{\mathbf{p}}^\top]^\top$ & $\mathbf{u} = \dddot{\mathbf{p}}$ $\int_0^T (\dddot{x}(t))_j^2 dt,\ j = 1,2,3$ & unconstrained	l (ad) e	exp t.p. (off)	quadrotor (+ racket) d.h. / $T_{\text{MPC}} = 0.02\text{s}$
[Neunert–2016]	$\mathbf{x} = [\mathbf{p}^\top \boldsymbol{\Theta}^\top \dot{\mathbf{p}}^\top \boldsymbol{\omega}^\top]^\top$ & $\mathbf{u} = [f_i|_{i=1}^6]^\top$ $\int_{t=0}^{t_f} \mathbf{e}_x^T(t)\mathbf{Q}\mathbf{e}_x(t) + \mathbf{u}_x^T(t)\mathbf{R}\mathbf{e}_u(t) + W(x,t)\ dt$ & unconstrained	nl i	exp fs.c. (on)	ballbot, hexarotor d.h. / $T_{\text{MPC}} \leq 25\text{ms}$
[Papachristos–2013a]	$\mathbf{x} = [x\ \dot{x}\ y\ \dot{y}\ z\ \dot{z}]^\top$ & $\mathbf{u} = [\phi_r\ \gamma_x\ T_z]^\top$ $\sum_{i=1}^{N_p}\tilde{\mathbf{e}}_{x,i}^T\mathbf{Q}\tilde{\mathbf{e}}_{x,i} + \sum_{i=1}^{N_c-1}\delta\mathbf{u}_i^T\mathbf{R}\delta\mathbf{u}_i + \sum_{i=1}^{N_p}\mathbf{e}_{u,i}^T\mathbf{N}\mathbf{e}_{u,i}$ & $\mathbf{u} \in \mathcal{U}$	l (ad) e	exp p.c. (on)	tri-tiltrotor $t_H = 0.8\text{s}$ / $N = 8$
[Papachristos–2013b]	$\mathbf{x} = [\theta\ q\ u\ w\ \alpha]^\top$ & $\mathbf{u} = [\Omega_F\ \Omega_B\ \delta_{\alpha_F}\ \delta_{\alpha_B}\ \alpha_r]^\top$ $\sum_{k=0}^{N-1}(\mathbf{e}_{x,k}^T\mathbf{Q}\mathbf{e}_{x,k} + \mathbf{u}_k^T\mathbf{R}\mathbf{u}_k) + \mathbf{e}_{x,N}^T\mathbf{P}\mathbf{e}_{x,N}$ & $\mathbf{x} \in \mathcal{X}, \mathbf{u} \in \mathcal{U}$	l e	sim ps.c.	quad-tiltrotor $t_H = 0.04\text{s}$ / $N = 4$

such application can be found in [Hofer–2016], which proposes to use MPC for the control of a small rotor-craft with limited computational resources, where the control and the state trajectory are approximated over an infinite horizon, while the dynamics and the constraints are simplified using a variational formulation and constraint sampling, respectively. Simulation results comparing the proposed MPC with an LQR-based approach are presented, showing an improvement in the tracking performance and in the stability, thanks to the fulfillment of the system constraints. Furthermore, an experimental validation is carried out performing trajectory tracking with a ducted-fan flying platform with flaps, in order to show the reliable disturbance rejection capabilities. A compelling contribution to the state of the art is given by [Darivianakis–2014], who tackle the inspection through contact of infrastructure facilities with a quadrotor, exploiting a hybrid MPC capable of stable and accurate trajectory tracking on environmental surfaces as well as force control. The position control of a quadrotor flying close to surfaces, in particular to perform ceiling inspections, is considered in [Kocer–2018]. In this work, an optimization-based algorithm leveraging the identified nonlinear model is employed to suppress the vibrations along the z-axis in an acceptable range for a real-time application. As an online controller, the nonlinear MPC tracks the position references by generating the reference attitude trajectories and the vertical force for another local controller acting in an inner control loop.

As can be recognized from an analysis of these and many other papers in the literature, the two problems of planning and tracking are generally addressed in an independent fashion. The issue with this separate methodology is twofold. On the one hand the planner might have no awareness about the system constraints and so it might not be able to re-plan the trajectory accordingly. On the other one, the controller might be asked to track a fixed trajectory without the possibility to adapt it to additional objectives emerging during the plan execution e.g., a new target, and/or constraints, e.g., a moving obstacle. Another example of such separated structure is given in [Bemporad–2009], where the two problems are tackled by two instances of MPC, hybrid and linear, respectively. The authors present a hierarchical scheme where the linear MPC is in charge of stabilizing the vehicle around commanded set-points, generated by the hybrid MPC controller at the higher level.

An exception to this commonly-used architecture is suggested in [Neunert–2016], where the authors recommend to consider the trajectory optimization and the tracking as a single problem, to which MPC can find a solution. In this work, a Sequential Linear Quadratic (SQL) algorithm is used to solve the optimal problem, simultaneously deriving the optimal feedforward and the feedback terms. An experimental validation on two different hardware platforms, namely a ball-balancing ground robot and a hexarotor aerial vehicle, is also presented. The method solves an unconstrained problem disregarding the physical constraints of the system, in order to decrease the computational burden associated with a constrained optimization. Therefore, the satisfaction of input and state limitations is not ensured, in this case.

Motivated by such breakthrough, we aim to solve both the trajectory generation and tracking in a unified approach, additionally taking into account state and input

constraints, exploiting state-of-the-art strategies to deal with strict requirements in the algorithm computing time. More in detail, we assume to have available a reference trajectory or a way-point sequence for the system evolution which does not take into account its physical limits. The goal of the proposed MPC framework is then to locally modify/generate the former in order to comply with high-priority constraints, and also to provide the correct control input to the system.

Cascaded vs full-state dynamics control

The separation between planner and tracker mirrors the typical the hierarchical structure of several flight control laws proposed in the literature. The common strategy adopted by most of the papers in the literature of aerial robotics is to break-down the model of the system in sub-levels and control each of them in a cascaded fashion. A first separation is classically identified between the translational and the rotational dynamics since, as observed, e.g., in [Kocer–2018], the loop cycle frequency from perception to execution is quite high in the latter channel compared to the former. This is a consequence of the fact that the rotational dynamics of the rigid body of MRAVs (in its simplest model) does not depend on the translational dynamics, while the contrary is true. Therefore, the common strategy in many contributions is to stabilize the rotational dynamics in an inner loop and to use the rotation configuration as a virtual input in the outer position-control loop. In order for such control framework to work properly, the virtual input needs to be precisely and quickly regulated, and this implicitly requires the rotation loop to be controlled at a higher frequency. On top of that, another partition is often created between the attitude loop and the motor-rotor (from now on referred to as *actuator*) dynamics. A time-scale separation between the translational, rotational and actuator dynamics is discussed in [Bangura–2014a], where the typical sampling frequencies and response times for each sub-level are presented. The authors motivate the choice of a nested control architecture with the significant computational cost that would be required to apply the feedback transformation to the full-state dynamics of the system. For these reasons, most of the contributions in the field of predictive controllers applied to aerial robotic systems in the last years have focused the attention towards the implementation of a predictive controller for only one of the first two layers (the positional or the rotational one).

In [Bangura–2014a], the authors use MPC for the position tracking of a quadrotor, while the inner-loop attitude control is performed by a high-gain Lyapunov-based controller (made up of a proportional and a feedforward term). The closed-loop dynamics of such sub-level is identified with a system of zero relative degree, meaning that the robot orientation is supposed to be instantaneously controlled. The inputs to the attitude controller, given by the MPC at higher level, are the angular velocity and total thrust derivative, while the angular acceleration is used as the feedforward term. A similar approach is also fostered by [Baca–2016] and by [Darivianakis–2014], where the control of the orientation relies on a PID regula-

tor and the rotational closed-loop dynamics is identified with a 1st- and a 2nd-order system, respectively. In both cases, the control problem is approximated as decoupled on each axis and therefore three distinct problems are solved independently. The input to the orientation control is, in such case, a reference value for the roll and pitch angles, together with the total thrust. The same cascaded formalism is found also in [Kamel–2015], where the authors employ MPC for the control of the inner rotational loop, while the outer one is entrusted to an LQR controller with an integral action. The proposed approach has been implemented onboard a hexarotor and evaluated both in simulations and experiments, showing the system capability of recovering from inverted attitude configurations and maintaining the control authority on the position in the case of one propeller loss. Another occurrence of MPC employed for the control of the attitude dynamics of a hexarotor is found in [Ligthart–2017], where an unconstrained MPC is validated with indoor and outdoor experiments. In this work, the input vector is approximated with Laguerre functions in order to deal with the computationally intensity induced by the large dimension of the control matrix. On the other hand, the input constraints are enforced with an anti-windup saturation solution, since they are not explicitly considered in the formulation of the predictive controller.

In all the works mentioned so far, MPC has been mainly used to solve specific parts of the overall control problem related to MRAVs, i.e., addressing either position tracking/regulation or attitude stabilization but never providing a full-state controller unifying the different sub-levels, as also pointed pointed out by [Foehn–2018]. The problem with this decoupled approach is that the control of each level is limited by the over-imposed references from the previous blocks at a higher level. In particular, the direct control over the orientation is lost since the platform rotation is over-imposed from a reference in linear acceleration, velocity and position that disregards the attitude dynamics and, even more, the dynamics of the actuators, which are the real means for the motion generation of the robot. Therefore, the components of the *full state* that are approximately treated as *partial inputs* – like the linear acceleration and velocity, the angular velocity or the Euler angles, and the total thrust or its dynamic extension – are constrained between heuristic limitations which are conceptually far from the physical constraints of the real system. Furthermore, even though such limits are state-dependent, a conservative constant (state independent) approximation is frequently assumed, to the detriment of attaining the most agile maneuvers possible. It is easy to be convinced that this control structure prevents the MRAV from completely exploiting the full operation envelope of its dynamics.

On the other hand, a different and farsighted perspective against this trend is given another time by [Neunert–2016], who showed how control performance can be significantly improved by allowing the MPC to directly act on the actuation system. In this case, the – yet unconstrained – inputs provided by the MPC are the forces produced by the spinning of the propellers.

Driven by the aim to develop a framework that fuses together the functionalities of a local trajectory planner/generator and a full-state feedback control which

additionally manage the low-level inputs of the system (related to the actuator dynamics), we further extend this approach by selecting control inputs that comply with a more detailed and representative model of the real physical limits of the system. Meaningful state-dependent limitations to such inputs are identified from the particular robot actuators thanks to an offline experimental procedure, and then managed online by the controller in order to fully exploit the dynamic capabilities of the vehicle. The increase of the computational burden which comes at the price of an augmented space-state dimension and the satisfaction of the input constraints, is kept bounded thanks to effectiveness of the OCP solvers and, above all, the use of an enhanced version of Real Time Iteration (RTI) scheme, as discussed in Sec. 7.2. The shift of the abstraction layer for the input down to the actuators level represents a key point of our contribution, done with the aim to provide an MPC strategy that can seamlessly deal with virtually any MRAV design.

Implicit vs explicit

While the previous classifications deal with *what* problem was targeted, another distinction can be done about *how* such problem is solved. In the literature panorama, the solution to the OCP is traditionally obtained in two ways. The first one, referred to as *implicit* MPC, consists in iterating *online* a numerical procedure to solve the problem over a limited-time future window, the so-called "prediction horizon". The outcomes of this procedure are the optimal control inputs which minimize a cost-function defined by the user, and the theoretical behaviors of the studied plant w.r.t. an identified model. Based on the measurements of the plant state at current time, the MPC predicts the state evolution along the horizon thanks to the model, and applies the input to the system at each sampling time. The whole procedure is continuously repeated based on new available measurements, *shifting* forward the considered window in the case of a *receding* horizon, as done in most of the cases, or *shrinking* it until a planned event in the case of a *diminishing* horizon as done, e.g., by [Mueller–2013a]. The main limitation related with the implicit technique is that solving a constrained OCP usually requires a consistent computational time, which has to be necessarily bounded by the sampling time of the system, in order to solve the problem "on-time". In the light of transferring such controller on a cheap process hardware, as could be a Programmable Logic Controller (PLC) in an industrial scenario, this could result prohibitive in the case of very high-bandwidth plants. This has been the reason why, for many years, MPC has been applied only to slowly-varying systems as, e.g., chemical processes.

This fact opened the door to the development of the second method, known as *explicit* MPC. Such approach involves the evaluation of the optimal control action *offline*, computed as an explicit function of the state and input vectors, exploiting multi-parametric programming techniques. Explicit MPC allows to solve the optimization problem off-line for a given range of operating conditions of interest, so that on-line operations reduce to a simple function evaluation. In most cases, such a function is piece-wise affine (PWA) of the state, so that the MPC controller maps

into a lookup table of linear gains. The advantage related with this version of MPC is that of reducing the on-line computational burden to a mere association of values that can be deployed also on cheap hardware. Such approach is fostered by [Alexis–2014], that employ an explicit approach in order to provide a robust MPC framework which ensures the minimum possible deviation from the reference for the worst-case disturbance, considering the Minimum Peak Performance Measure (MPPM) as the metric of optimality. In this case, the use of the explicit formulation is motivated by the high computational cost associated with the optimization, which has to be performed w.r.t. a considerable number of possible evolutions of the disturbance. Interesting experimental results are presented therein and also in [Alexis–2016b], where the robust MPC algorithm performance are evaluated on two different multi-rotor configurations, performing trajectory tracking in the challenging conditions of external disturbances and slung load operations. Another relevant contribution in the field of explicit MPC is given by [Liu–2012]. By approximating the tracking error and control efforts in the receding horizon using Taylor expansions up to a specific order, the authors derive the analytic solution to the MPC problem offline and present the control performance by means of numerical simulations and real flights on an indoor testbed using a small helicopter. Unfortunately, the particular explicit formulation disregards the input and state constraints, and so the computed solution could result unfeasible. The development of an explicit MPC framework tailored to deal with *constrained* Linear Time Invariant (LTI) system is thoroughly discussed in [Bemporad–2002]. In general, for an exhaustive review of other significant contributions on explicit MPC, the interested reader is referred to [Alessio–2009].

Despite being an attractive alternative to the implicit methodology in the case of small problem size, i.e., up to around 5 states, as explained in [Hofer–2016], the explicit strategy suffers the so-called *curse of dimensionality*, i.e., the exponential growth of the total number of the control regions with respect to some key parameters of the controlled system, e.g., the number of states. This causes a dramatic increase in the controller memory requirements and makes the first step of PWA evaluation, i.e., searching for the current control region, computationally intractable. In the case of a generic MRAV, the dimension of the full state-space vector is a linearly increasing function of the number of actuators. Therefore, we decided to avoid the explicit formulation mainly for a matter of scalability.

Motivated by the intent of modeling the full-state dynamics, as stated in the previous paragraph, we decided to develop the work presented in Sec. 5.2 based on the on the implicit formulation, exploiting the state-of-the-art RTI technique outlined in [Chen–2017] in order to reduce the computational time required to solve the OCP. In preparatory experiments to this work, we have ascertained that the computational time with this approach is small enough to guarantee the satisfaction of the typical control frequencies for different multi-rotor systems with up to 18 states and 6 inputs and a prediction horizon of 1s, already in a sub-optimal software implementation.

Linear vs nonlinear

Again in the scope of the method, another classification can be made w.r.t. the kind of model which is employed to simulate the dynamics of the system to be controlled, in particular *linear* or *nonlinear*. While most of real processes are nonlinear, they can often be reasonably approximated as linear over a small operating range. Linear Model Predictive Control (LMPC) approaches are used in the majority of applications thanks to the feedback mechanism of the MPC which compensates for prediction errors due to structural mismatch between the model and the process. In this scope fall the contributions that, as done for example by [Bangura–2014a], identify the closed-loop attitude dynamics and focus on the control of the translational dynamics, which is naturally linear, and the ones linearizing the system around the hovering condition, as done in [Baca–2016], in the hypothesis of small attitude angle deviations.

If it is true that such an approach allowed to develop controllers capable of running onboard simple and computationally limited hardware, by benefiting from a simplified problem, it is well known in the literature that it results insufficient to exploit the full envelope of the dynamics, especially when dealing with agile trajectories that are far from the stable hovering point. An excellent and clarifying comparison between LMPC and Nonlinear Model Predictive Control (NMPC) for MRAVs trajectory tracking is presented in [Kamel–2017a]. In this paper, the authors evaluate the performance of the two kinds of controller, regarding both the tracking performance and the computational time to solve the problem, by testing them on the same hexarotor platform. Concerning the first point, the NMPC reduces the Root Mean Square Error (RMSE) w.r.t. the LMPC in hovering conditions under wind disturbances and in particular outperforms the linear approach in the step response, with faster reaction and no remarkable overshoot. The difference in the performance is even sharper in the case of agile maneuvers, thanks to the exploitation of the full dynamics in the NMPC case. Furthermore, the problem resolution turns out to be five times faster in the nonlinear case compared to the linear one. The explanation for this effectiveness is found in the use of the RTI scheme implemented in the NMPC. The basic idea behind this strategy is to initialize each new iterative problem with the most current solution guess from the previous problem resolution, exploiting the fact that consecutive OCPs are likely to be similar to each other. This allows to initialize the Newton-type method efficiently and so to consistently improve the performance in terms of computation speed. A more thorough explanation of the linear and nonlinear models, the controller design and an implementation in the Robot Operating System (ROS) framework, for both multi-rotor systems and fixed-wing aerial vehicles, is presented in [Kamel–2017b]. The reader interested in a perspicuous overview on the control methods for NMPC and a detailed explanation of the RTI scheme is addressed in [Diehl–2006].

Driven by the intent of precisely describing the full envelope of the dynamics of a generic multi-rotor system, while keeping the computation time small enough to control high-bandwidth processes, we decided to adopt the nonlinear formulation.

UDT vs MDT MRAVs

As far as the application is concerned, almost all the works in the literature of aerial robotics, fairly represented by the consistent amount of previously cited references, have employed MPC to deal only with UDT MRAVs. An interesting exception to this trend is represented by the work of [Papachristos–2013a], which deals with the translational hovering control of a tri-rotor aerial vehicle that has the capability to tilt two of its actuators about one common axis, allowing to exploit the *thrust-vectoring* capability along the longitudinal direction of the body frame. Such a MRAV is capable to perform flight mode conversions between VTOL and fixed-wing, partially inheriting the benefits of both configurations. In this work, the translational controller is provided by three linear MPC instances, one for each axis. Indoor experimental results for simple translational motions are presented therein, while the tracking of more complex trajectories has been addressed in [Papachristos–2016]. Furthermore, the very same aerial robot has been exploited for the experimental validations in other correlated works [Alexis–2014; Alexis–2016b]. Finally, a similar concept has been outlined by the same authors in [Papachristos–2013b], employing a hybrid and explicit MPC, approximating the nonlinear dynamics with a PWA modeling, for the mode transition control of a quad-tiltrotor platform. In this case, only simulation results have been shown.

The baseline motivation of the work presented in Chapt. 5 is to provide the scientific community with a full-state, implicit, nonlinear and constrained MPC framework for the seamless control of arbitrarily-designed MRAVs, in particular for both under-actuated UDT platforms and, in general, MDT ones. As we have seen in Chapt. 1, the latter are designed to produce a moment that can be partially decoupled from the total thrust, which now can assume different directions in body frame, as shown in the comparison of Fig. 1.5. The way these platforms move is intrinsically different from classical UDT vehicles, since MDT ones do not necessarily need to re-orient themselves to express a thrust force in different directions. This capability is achieved at the cost of spinning the propellers in a wider dynamic range, meaning that the spinning rate of the actuators need to span a larger set w.r.t. UDT platforms in order to move their position in the same way while remaining horizontal. The validity of such claim can be assessed by comparing the last plot of Fig. 7.22 and Fig. 7.27, which depict the profile of the actuator velocities of a UDT quadrotor and a MDT hexarotor for the tracking of the very same position trajectory. It is easy to be convinced that for such vehicles, the precise control of the motor spinning velocities – that generates in turn the actuation wrench – is crucial for an accurate tracking.

In view of these considerations (and the others previously highlighted), in order to define meaningful input constraints compatible with a general MRAV system, it makes little sense to consider the orientation of the platform as a virtual input, because the total thrust orientation is not anymore necessarily constrained to be aligned with the vertical z axis of the body frame. Furthermore, it results also inconvenient to consider the total thrust and moment as decoupled virtual inputs,

because they could be coupled by the under-actuation, as it is always the case for UDT-MRAVs, or more in general, by the limits of the propeller forces. This further motivates our claim to move the input abstraction layer down to the actuators level.

MPC and learning

An interesting and recently-developed research line suggests to inquire how statistical learning methods might be integrated with control techniques, in particular with predictive ones. An example of this work is the paper from [Bouffard–2012], where the authors propose a new model-based strategy that further allows for online updates to the model in order to improve the tracking performance, while still guaranteeing some robustness in the control. The control architecture uses a modified version of the extended Kalman filter (EKF) to perform state estimation and model parameters learning. Experimental results for the position control of a quadrotor show a decreased overshoot in the step response compared to a classic LMPC approach. The framework is further validated with the demonstrative task of catching a ball thrown by a human operator, showing a rate of successful catches of over 90%. With a similar goal but different tools, [Berkenkamp–2015] combine machine learning with robust control using Gaussian Process (GP) regression to update an initial model of the system and decrease the uncertainty associated to this model. Thanks to the GPs, such an approach can deal with nonparametric and nonlinear models. An improvement in the step response time of a quadrotor, together with a decreased overshoot, is shown at the increase of the cardinality of the set of points used to train the GP.

Even if the combination of learning techniques with MPC falls outside the scope of this thesis, it is worthwhile to mention the breakthroughs related to such an active topic in the literature. Nevertheless, the employment of such methods in our setup is left for future investigation. For a schematic summary of the key-points regarding the MPC contributions related to MRAVs presented so far in the state of the art, we address again the reader to Tab. 2.1.

Contribution brought by the work in this thesis

Despite the field of MPC-based control for MRAVs is already deeply studied, we believe there is still a considerable margin for interesting research investigation, in particular in relation to the employment of more precise models which take into account more representative constraints for the actuators, and of techniques that can be applied to arbitrarily-designed MRAVs and are demonstrated to run onboard the platform in real experiments.

The aforementioned conservative modeling and control design choices have been often adopted so far by the community to mitigate for possible problems deriving from the commonly computationally burdensome online solution to the OCP; however, they may significantly compromise the closed-loop system performances. In the work related to MPC developed in this thesis, the challenge is to bring these

Table 2.2 – Overview of the contributions brought to the literature by this thesis w.r.t. recent relevant works in the state of the art of MPC applied to MRAVs. A: capability to steer platforms that can independently control their position and orientation, B: full nonlinear model and control (non-cascaded) for the system dynamics, C: extended model for the actuators dynamics including low level constraints, D: controller validated through real experiments with online computation, E: framework suitable to control arbitrarily-designed MRAVs. ✓: implemented, ✗: not implemented.

	[Alexis–2014]	[Alexis–2016b]	[Baca–2016]	[Bangura–2014a]	[Bouffard–2012]	[Darivianakis–2014]	[Foehn–2018]	[Geisert–2016]	[Hofer–2016]	[Kamel–2015]	[Kamel–2017a]	[Ligthart–2017]	[Lin–2016]	[Liu–2012]	[Liu–2015]	[Mueller–2013a]	[Mueller–2015]	[Neunert–2016]	[Papachristos–2013a]	[Papachristos–2016]	WORK IN THIS THESIS
A	✗	✗	✗	✗	✗	✗	✗	✓	✓	✗	✗	✗	✗	✓	✗	✗	✗	✓	✗	✗	✓
B	✗	✗	✗	✗	✗	✗	✗	✓	✗	✓	✓	✗	✗	✓	✗	✗	✗	✓	✗	✗	✓
C	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✓
D	✓	✓	✓	✓	✓	✓	✓	✗	✓	✓	✓	✓	✗	✓	✓	✓	✓	✓	✓	✓	✓
E	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✓

modeling and design choices into question. In particular, we show that MPC, its numerical implementations, and the available computing platforms can currently fully support high-performance, real-time, constrained, and predictive control of MRAVs. The tested MPC scheme for local trajectory generation and tracking uses a full-order nonlinear model. Another important novelty of the approach, presented in Sec. 4.2, is the take-advantage of a novel actuator model that considers more representative actuation constraints compared to existing solutions, thus leveraging the vehicle dynamic capabilities in a better way. It should be noted that this model and control framework is suitable to seamlessly describe UDT and MDT MRAVs, differently from previous contributions. To the best of our knowledge, this is the first time that a framework with all such characteristics is successfully tested online to control non-specific aerial vehicles with an arbitrary propeller arrangement. Following the discussion above, Table 2.2 provides a summary of the contributions related to MPC for MRAVs which are brought to the literature by this work, compared to some of the most relevant works in the recent state-of-the-art. The parts of the thesis related to this topic are Sec. 4.2, Sec. 5.2, and Sec. 7.2.

Chapter 3

Theoretical background

"He who loves practice without theory is like the sailor who boards ship without a rudder and compass and never knows where he may cast."
Leonardo da Vinci

This chapter is devoted to a brief review of the theoretical methodologies employed in this thesis, in particular in Part II, that are presented here in order to introduce some of the mathematical tools which are needed to properly tackle the analytic study of Multi-Rotor Aerial Vehicles (MRAVs). In particular, this review covers general fundamental methods to: *i) model* a mechanical system and, in particular, a MRAV; *ii)* analyze its *dynamic properties*; *iii)* assess its (physical) *limits*; *iv)* design one possible *control method* to accomplish a certain task; and *v)* understand how stability problems could arise due to system constraints. In particular, at the end of Sec. 3.1 a particular attention is devoted to the class of Multi-Directional Thrust (MDT) hexarotors Aerial Vehicles (AVs), i.e., MRAVs with six arbitrarily oriented rotors. A brief outline of the advantageous properties of this specific class of platforms will motivate its relevance and key role in this thesis.

In order to ease the reader understanding and to introduce her/him to the contributions of this thesis in a *crescendo* fashion, i.e., gradually introducing the central topics from more general ones, the mentioned concepts shall be introduced in the following, after a short summary of the adopted notation.

Notation. In this thesis, we denote (column) vectors and matrices in bold font, with lower and upper cases, respectively. The transpose operator is indicated with the superscript $\bullet^\top$. Letter superscripts of vectors represent the reference frame w.r.t. which these vectors are expressed[1]. The notation $\mathbb{R}^{m\times n}$ indicates the set of real matrices with m rows and n columns. In particular, $\mathbb{R}^{n\times n}$ represents a square matrix. Furthermore, $\mathbb{R}^{n\times n}_{>0}$ denotes the set of positive-definite real matrices, i.e., $\mathbb{R}^{n\times n}_{>0} = \{\mathbf{M} \in \mathbb{R}^{n\times n} \,|\, \mathbf{x}^\top\mathbf{M}\mathbf{x} > 0, \forall \mathbf{x} \in \mathbb{R}^n\}$. $\mathbf{0_{m,n}}$ and $\mathbf{1_{m,n}}$ denote the matrices with m rows and n columns with all the elements equal to 0 and 1, respectively. $\mathbf{A}\otimes\mathbf{B}$ denotes the Kronecker product between the matrices $\mathbf{A}$ and $\mathbf{B}$. The notations $\mathrm{SO}(3)$ and $\mathrm{SE}(3)$ refer to special orthogonal group of 3D rotations and the special Euclidean group of 3D rotations and 3D translations, respectively. The symbol $so(3)$ describes the Lie-algebra of $\mathrm{SO}(3)$. The operator $[\bullet]_\times \in so(3)$ represents the skew symmetric matrix associated to any vector $\bullet \in \mathbb{R}^3$, while its inverse operator $\bullet^\vee \in \mathbb{R}^3$ indicates the vector associated with any skew-symmetric matrix $\bullet$.

[1] If not specified, the inertial world frame should be considered as reference.

Modeling of a physical system

In the context of Classical Mechanics, being the branch of Physics that studies the motion of macroscopic objects whose speed is not approaching the one of light, three main formalisms can be identified, i.e., the *Newtonian*, the *Lagrangian* and the *Hamiltonian*. The first published versions of such methodologies in their original languages can be found in [Newton–1687; Lagrange–1811; Hamilton–1834], respectively. More in detail, the Newtonian formalism for point-particle was extended to the rigid body case by Leonhard Euler in [Euler–1736], about 50 years after Newton's laws first publication. For this reason, in the literature (and also in this thesis) the related formalism is commonly referred to as *Newton-Euler* formalism. While the Newton-Euler and the Lagrangian methodologies are proper of Classical Mechanics, the Hamiltonian one contributed also to the formulation of Statistical Mechanics and Quantum Mechanics.

In the following, we restrain our domain of interest to the branch of Dynamics, that is the study of motion and its relation to forces, always in the case of Classical Mechanics applied to a rigid body. Throughout the thesis, we make use of the first two methodologies, which are the most conveniently used in this field, in order to compute dynamical models that we use for control objectives. The interested reader in the vast topic of Classical Mechanics is addressed to dedicated textbooks, e.g., [Goldstein–2002]. It is important to underline that, while the mentioned approaches lead to the same outcome, as it is expected, their practical procedures are quite dissimilar. Furthermore, they could provide different perspectives and insights about the system and its property.

As far as the Newton-Euler approach is concerned, it looks at the Dynamics through forces and torques, expressed in particular Cartesian coordinates. This method is particularly suited for multi-body systems as manipulators with an open kinematic chain, given its *recursive* and efficient nature. Indeed, it treats each joint as an independent entity, and then computes all the coupling by using the so called *forward-backward* recursive algorithm. Thanks to its property of *scalability*, it can dexterously be employed to deal with other complex systems. However, a particular attention should devoted in the case of constrained systems, since the reaction forces induced by the constraints has to be explicitly taken into account.

On the other hand, Lagrangian Mechanics is independent of any particular coordinate system. This elegant and systematic formalism allows to *analytically* derive the dynamic equations describing the evolution of the system model. Instead of writing the equations of motion by starting from the analysis of the force and torque vectors, as in the previous methodology, a proper set of *generalized coordinates* is chosen, that allows to naturally and automatically take into account all the system constraints. Then, the kinematic and the potential energies should be computed. This allows to facilitate the identification of conserved quantities in the model, which could be an interesting property in different situations. Finally, the equations of motion are obtained applying the Lagrange equations. It should be remarked that, despite its smart form, this method reveals unpractical for complex

systems with a high number of degrees of freedom.

In this thesis, most of the models are obtained using the first formalism due to its ease of application for the considered aerial systems. In some cases, the combination of the two methods allowed us to obtain the best representation of the dynamics for our control objectives. In the following, the implementation basis of the two formalisms will be outlined in order to ease the understanding throughout the thesis. For more exhaustive theoretical explanations and practical examples we address the interested reader to [Spong–2006; Siciliano–2010; Siciliano–2016; Lynch–2017].

Newton-Euler formalism

The basis of the Newton-Euler formulation of the Dynamics is the conservation of two fundamental quantities of Physics i.e., , the linear and the angular momenta, in the case of absence of external wrench (forces and torques) applied to the considered rigid-body. If some wrench is applied to the system, then the derivative of the previously mentioned quantities, taken w.r.t. an inertial reference frame, is equal to the total contribution of forces and torques applied to the Center of Mass (CoM). In mathematical terms, that can be written as

$$\left.\frac{d(m\mathbf{v})}{dt}\right|_I = \mathbf{f} \tag{3.1}$$

$$\left.\frac{d(\mathbf{J}^I\boldsymbol{\omega})}{dt}\right|_I = \boldsymbol{\tau}, \tag{3.2}$$

with $m \in \mathbb{R}_{>0}$ and $\mathbf{J}^I \in \mathbb{R}^{3\times3}_{>0}$ being the mass and the inertia matrix of the rigid body about an inertial frame whose origin is at the center of mass and $\mathbf{v}, \boldsymbol{\omega} \in \mathbb{R}^3$ being its linear and angular velocity, respectively. The symbol $|_I$ stresses the fact that the derivation is performed w.r.t. an inertial frame. Finally, $\mathbf{f}, \boldsymbol{\tau} \in \mathbb{R}^3$ represent the total contribution of the forces and the torques applied to the CoM. All these quantities will be deeply discussed further in this thesis.

In the case of a multi-link body system, the Newton-Euler formalism foresees to treat each link in turn, writing down the equations describing its linear and angular motion, considering the coupling interactions coming from neighboring links. At this point, the methodology is based on the application of an algorithm made up of two recursive steps: *i)* forward recursion, and *ii)* backward recursion. This allows to compute all the coupling terms and eventually to arrive at a description of the multi-link body as a whole.

The *forward recursion* is done to propagate the links velocities and accelerations from the first link to the final one. After having defined proper reference frames on each augmented link (incorporating the link itself and the actuating motor) and standard conventions, the translational and rotational velocities and accelerations of the i-th link are computed based on the ones of the previous $(i-1)$-th link and

of the i-th joint, according to its type (either prismatic or revolute). Repeating such method for all the links starting from the base link allows computing the velocities and accelerations of the End Effector (EE), being the terminating part of the manipulator chain which is used to perform some tasks like, e.g., interacting with the environment.

On the other hand, the *backward recursion* propagates forces and moments from the last link to the first one. Knowing the total force and moment applied to the $(i+1)$-th link, denoted by $\mathbf{f}_{i+1} \in \mathbb{R}^3$ and $\boldsymbol{\tau}_{i+1} \in \mathbb{R}^3$, respectively, the ones applied to the i-th link, i.e., $\mathbf{f}_i \in \mathbb{R}^3$ and $\boldsymbol{\tau}_i \in \mathbb{R}^3$, are computed resolving the Newton-Euler equations. The method is repeated for each link starting from the EE one, whose external wrench is known from the interaction, back to the base link.

It is worth remarking that the equations resulting from the application of the Newton-Euler formalism are *not* in *closed-form* , given the coupling of each link with the neighboring ones. Nevertheless, solving the presented *recursive algorithm* allows to efficiently solve both the direct (and also the inverse) dynamic problem.

Lagrange formalism

The basis of the Lagrangian formulation is the choice of a set of independent coordinates $\mathbf{q} = [q_1 \ \dots \ q_{\bar{n}}]^\top \in \mathbb{R}^{\bar{n}}$, called *generalized coordinates*. These quantities shall completely describe the configuration of the system and its $\bar{n} \in \mathbb{N}_{>0}$ degrees of freedom (DoFs). A proper choice for the generalized coordinates allows to automatically take into account the system constraints. Accordingly to the chosen generalized coordinates, we can then compute the *generalized forces* acting on the system. Consider a set of forces $\mathbf{f} = [\mathbf{f}_1^\top \ \dots \ \mathbf{f}_{\bar{m}}^\top]^\top \in \mathbb{R}^{3\bar{m}}$, where the generic force $\mathbf{f}_i \in \mathbb{R}^3$ is applied on the system at point $\mathbf{r}_i \in \mathbb{R}^3$, with $i = 1, \dots, \bar{m}$ and $\bar{m} \in \mathbb{N}_{\geq 0}$. It is possible to compute the generalized force $\xi_j(\mathbf{f}, \mathbf{q}) \in \mathbb{R}$ w.r.t. the j-th generalized coordinate q_j as:

$$\xi_j(\mathbf{f}, \mathbf{q}) = \sum_{i=1}^{m} \mathbf{f}_i^\top \frac{\partial \mathbf{r}_i}{\partial q_j}, \quad j = 1, \dots, \bar{n}. \tag{3.3}$$

At this point, the *Lagrangian* function $\mathcal{L}(\mathbf{q}, \dot{\mathbf{q}})$ can be defined, which is equal to the difference of total kinetic energy, $\mathcal{K}(\mathbf{q}, \dot{\mathbf{q}})$, and potential energy, $\mathcal{U}(\mathbf{q}, \dot{\mathbf{q}})$, i.e., $\mathcal{L}(\mathbf{q}, \dot{\mathbf{q}}) = \mathcal{K}(\mathbf{q}, \dot{\mathbf{q}}) - \mathcal{U}(\mathbf{q}, \dot{\mathbf{q}})$. Finally, the equation of motions of the system are obtained applying the following Lagrange equations:

$$\frac{d}{dt} \frac{\partial \mathcal{L}(\mathbf{q}, \dot{\mathbf{q}})}{\partial \dot{q}_j} - \frac{\partial \mathcal{L}(\mathbf{q}, \dot{\mathbf{q}})}{\partial q_j} = \xi_j(\mathbf{f}, \mathbf{q}), \quad j = 1, \dots, \bar{n}. \tag{3.4}$$

For the type of mechanical systems under exam in this thesis, the potential energy usually corresponds to the sole gravitational potential energy, and the kinetic energy can be computed as a quadratic form, $\mathcal{K}(\mathbf{q}, \dot{\mathbf{q}}) = \frac{1}{2} \dot{\mathbf{q}}^\top \mathbf{M}(\mathbf{q}) \dot{\mathbf{q}}$, where $\mathbf{M}(\mathbf{q}) \in \mathbb{R}^{\bar{n} \times \bar{n}}$ is the inertia matrix of the system. Consequently, the equations of motion

in (3.4) can be then rewritten in the more usual form:

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{g}(\mathbf{q}) = \boldsymbol{\xi}(\mathbf{f}, \mathbf{q}), \tag{3.5}$$

where $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}}$ contains the centrifugal and Coriolis terms, while $\mathbf{g}(\mathbf{q})$ contains the gravitational terms, and $\boldsymbol{\xi}(\mathbf{f}, \mathbf{q}) = [\xi_1(\mathbf{f}, \mathbf{q}) \ \dots \ \xi_{\bar{n}}(\mathbf{f}, \mathbf{q})]^\top \in \mathbb{R}^{\bar{n}}$. With this formulation, the contributions to the generalized forces in $\boldsymbol{\xi}(\mathbf{f}, \mathbf{q})$ are given by the non-conservative forces, e.g., for the case of a robotic manipulator, the joint actuator and friction wrenches, as well as the joint wrenches induced by EE wrenches due to interaction with the environment.

Remark (direct dynamics). The *direct dynamics* problem consists into evaluating the system motion, expressed in terms of the generalized coordinates and their derivatives, i.e., $\ddot{\mathbf{q}}, \dot{\mathbf{q}}$ and $\mathbf{q}$, given the generalized forces $\boldsymbol{\xi}(\mathbf{f}, \mathbf{q})$ and the system initial conditions. Considering the generalized forces as inputs and the motion as output of the system representation, this problem is equivalent to the *simulation problem*, i.e., determining the outputs given certain inputs.

Remark (inverse dynamics). The *inverse dynamics* problem consists into computing the generalized forces $\boldsymbol{\xi}(\mathbf{f}, \mathbf{q})$ given a certain motion expressed in terms of $\ddot{\mathbf{q}}, \dot{\mathbf{q}}$ and $\mathbf{q}$. Considering the generalized forces as inputs and the motion as output of the system representation, this problem is equivalent to the (feedforward) *control problem*, i.e., computing possible inputs to obtain (nominally) certain desired outputs.

Remark (comparison of the two formalisms). Given the analytic expression of the dynamic model (3.5), the Lagrangian formalism is often used to solve the inverse dynamics problem, and thus the control problem. On the other hand, the efficient recursive algorithm provided by the Newton-Euler formalism makes this method advisable to efficiently solve both the direct dynamics problem and the inverse dynamics problem. Nevertheless, the latter formalism presents a good set of advantages compared to the former one. In particular, *i)* it is systematic and of straightforward comprehension; *ii)* it facilitates the identification of conserved quantities in dynamical systems; and *iii)* it is effective to include in the modeling more complex mechanical effects like, e.g., flexible link deformations.
Throughout this thesis, most of the times we will make use of the Newton-Euler formalism, given its natural applicability to floating rigid bodies like the one considered in this study. When convenient, we will also combine this method with the Lagrangian one, e.g., for the modeling of aerial vehicles with manipulators, as briefly outlined in Sec. 10.2.

Modeling of a MRAV

In this thesis, MRAVs are modeled as rigid bodies[2] having mass m, actuated by $n \in \mathbb{N}_{>0}$ spinning motors coupled with propellers, e.g., $n = 4$ and $n = 6$ in the

[2]A rigid body a solid object which is not subject to deformation (or, more realistically, for which the deformation effect is so small to be safely neglected).

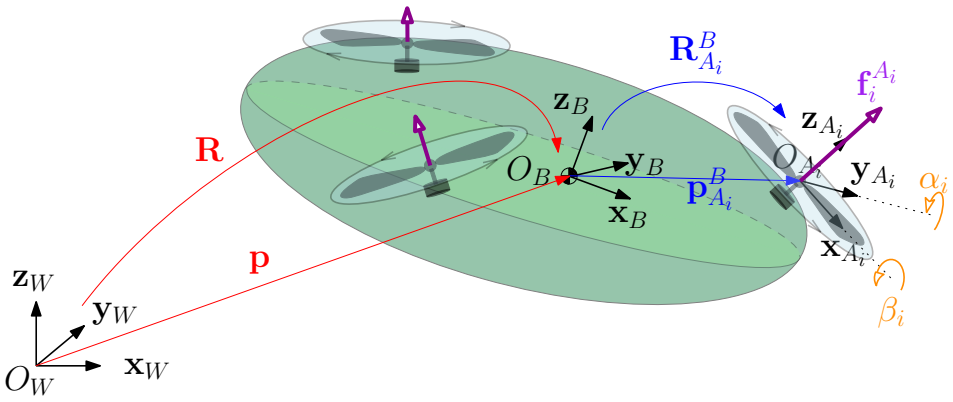


Figure 3.1 – Schematic representation of a generic MRAV with its reference frames.

particular quadrotor and hexarotor models, respectively. Keeping n generic allows to express the model in a non-specific form. With reference to Fig. 3.1, we denote with $\mathcal{F}_W = O_W, \{\mathbf{x}_W, \mathbf{y}_W, \mathbf{z}_W\}$ and $\mathcal{F}_B = O_B, \{\mathbf{x}_B, \mathbf{y}_B, \mathbf{z}_B\}$ the world inertial frame, and the body frame attached to the MRAV, respectively. The origin of $\mathcal{F}_B$, i.e., O_B, is chosen coincident with CoM of the aerial platform and its position w.r.t. O_W, expressed in $\mathcal{F}_W$, is denoted with $\mathbf{p}_B^W \in \mathbb{R}^3$, shortly indicated with $\mathbf{p}$ in the following[3]. The orientation of $\mathcal{F}_B$ with respect to $\mathcal{F}_W$ is represented by the rotation matrix $\mathbf{R}_B^W \in \mathrm{SO}(3)$, denoted with $\mathbf{R}$ for ease of notation[4]. We also define with $\mathcal{F}_{A_i} = O_{A_i}, \{\mathbf{x}_{A_i}, \mathbf{y}_{A_i}, \mathbf{z}_{A_i}\}$ the reference frame related to the i-th actuator[5], $i \in \{1, \dots, n\}$, with O_{A_i} attached to the thrust generation point and $\mathbf{z}_{A_i}$ aligned with the thrust direction. Thanks to this convention, the actuator force expressed in its frame is $\mathbf{f}_i^{A_i} = \gamma_i \mathbf{e}_3$, where $\gamma_i \in \mathbb{R}$ is the scalar value of the produced force and $\mathbf{e}_i$, $i{=}1,2,3$ represents the i-th vector of the canonical basis of $\mathbb{R}^3$. The position of O_{A_i} w.r.t. O_B, expressed in $\mathcal{F}_B$, is indicated with $\mathbf{p}_{A_i}^B$, while the orientation of $\mathcal{F}_{A_i}$ with respect to $\mathcal{F}_B$ is represented with $\mathbf{R}_{A_i}^B$.

Remark (generic MRAV). As already mentioned, in this thesis we consider generic MDT MRAVs whose designs can have arbitrarily positioned and oriented actuators.

The positive definite matrix $\mathbf{J} \in \mathbb{R}_{>0}^{3\times3}$ denotes the vehicle inertia matrix with respect to O_B, expressed in $\mathcal{F}_B$. The angular velocity of $\mathcal{F}_B$ with respect to $\mathcal{F}_W$, expressed in $\mathcal{F}_B$, is indicated with $\boldsymbol{\omega}_B^B \in \mathbb{R}^3$, and compactly denoted as $\boldsymbol{\omega}$ in the following. The vehicle orientation kinematics, accounting for the evolution of the rotation matrix $\mathbf{R}$, is described by the well-known equation

$$\dot{\mathbf{R}} = \mathbf{R}\,[\boldsymbol{\omega}]_\times\,. \tag{3.6}$$

The main symbols related to the modeling of a MRAV have been collected in Tab. 3.1.

[3] When the superscript is not present, $\mathcal{F}_W$ shall be intended as the reference frame, if not explicitly specified otherwise.

[4] $\mathrm{SO}(3) = \{\mathbf{R} \in \mathbb{R}^{3\times3} \,|\, \mathbf{R}\mathbf{R}^\top = \mathbf{R}^\top\mathbf{R} = \mathbf{I}_3, \det(\mathbf{R}) = 1\}$.

[5] Using the word *actuator*, we refer to the motor coupled with the propeller.

Table 3.1 – Overview of the main symbols used in the modeling of MRAV.

Definition	Symbol
World Inertial Frame	$\mathcal{F}_W$
Multi-rotor Body Frame	$\mathcal{F}_B$
Actuator frame (i-th)	$\mathcal{F}_{A_i}$
Position, velocity, acceleration of O_B w.r.t. O_W, in $\mathcal{F}_W$	$\mathbf{p}$, $\dot{\mathbf{p}}$, $\ddot{\mathbf{p}}$
Rotation matrix representing $\mathcal{F}_B$ w.r.t. $\mathcal{F}_W$	$\mathbf{R}$
Angular velocity of $\mathcal{F}_B$ w.r.t. $\mathcal{F}_W$, expressed in $\mathcal{F}_B$	$\boldsymbol{\omega}$
Angular acceleration of $\mathcal{F}_B$ w.r.t. $\mathcal{F}_W$, expressed in $\mathcal{F}_B$	$\dot{\boldsymbol{\omega}}$
Position of O_{A_i} w.r.t. O_B, expressed in $\mathcal{F}_B$	$\mathbf{p}_{A_i}^B$
Rotation matrix representing $\mathcal{F}_{A_i}$ w.r.t. $\mathcal{F}_B$	$\mathbf{R}_{A_i}^B$
Mass of the vehicle	m
Vehicle's inertia matrix w.r.t. to O_B, expressed in $\mathcal{F}_B$	$\mathbf{J}$
Gravity acceleration	g
Total control force acting on the CoM	$\mathbf{f}_B$
Total control moment acting on the CoM	$\boldsymbol{\tau}_B$
External force acting on the CoM	$\mathbf{f}_{B\text{-ext}}$
External moment acting on the CoM	$\boldsymbol{\tau}_{B\text{-ext}}$

Remark (orientation representations). A free rigid body, i.e., not under the action of constraints, has six degrees of freedom. Indeed, its associated body frame $\mathcal{F}_B$ can translate along and rotate around three main axes of the world inertial frame $\mathcal{F}_W$. The three translational Degrees of Freedom (DoFs) are represented by the position of O_B in $\mathcal{F}_W$, which in turn is described by the vector $\mathbf{p}$. On the other hand, the description of the three rotational DoFs is more delicate. Indeed, there exists different possible representations that can be employed [Spong–2006; Siciliano–2010; Siciliano–2016; Corke–2017]. The most popular and used by the robotic community are the following ones.

- The *rotation matrix*: $\mathbf{R}_B^W \in \mathrm{SO}(3)$, $\mathbf{R}_B^W \mathbf{R}_B^{W\top} = \mathbf{R}_B^{W\top} \mathbf{R}_B^W = \mathbf{I}_3$, unequivocally describes the rotation of $\mathcal{F}_B$ w.r.t. $\mathcal{F}_W$. Thanks to Euler's rotation theorem, $\mathbf{R}_B^W$ can be seen as the composition of at most three rotations about some coordinate axis, without two successive rotations about the same axes. The rotation matrix has intrinsically a redundant information content, as nine parameters are used to represent the three rotational DoFs of the rigid body. Nevertheless, this representation eases many operations like, e.g., vector rotations and compositions of rotations. These facts, together with the absence of singularities (see in the following), make this description one of the preferable for control design.
- The *Euler angles* (more in general, all minimum representations based on three angles): exploiting Euler's rotation theorem, it is possible to describe the rotation of a rigid body with only three parameters, which represent the angles of the ordered succession of rotations around three chosen axes. This

allows to use a minimal set of three parameters. Depending on the choice of the axes sequence and on the order of the rotations, different names are used to refer to this representation. One of the most popular convention in the aeronautic field consists in successive rotations about the moving axes $\mathbf{z}_B$, $\mathbf{y}_B$, and $\mathbf{x}_B$ with the angles ψ, θ, and φ, knows as *intrinsic yaw-pitch-roll* sequence[6]. Despite the ease of using only three parameters, this description has a singularity. This occurs when the rotational axis of the middle term in the sequence becomes parallel to the rotation axis of the first or the third term. Therefore, this representation has to be used with awareness.

- The *exponential* or *axis-angle* representation: this description allows to represent the relative orientation between two frames by using a single rotation about a specific axis $\mathbf{v}$ of an angle θ. It should be remarked that this description is not unique, since a rotation of $-\theta$ about $-\mathbf{v}$ results in the same orientation. This representation is parametrized by four parameters: three for the rotation axis, and one for the angle of rotation. However, one of the three parameters of the unit vector can be computed from the other two. Thus, this is another minimal description. Nevertheless, combinations of rotations are not straightforward and the axis of rotation is undetermined when the rotation angle goes to zero.
- The *unit-quaternions*: by using a normalized four-dimensional vector, i.e., four parameters subjected to one constraint, it is possible to get rid of the singularity of other representations. Furthermore, operations with this elegant parametrization are very efficient in terms of computational cost.

Throughout this thesis, we will make use of the rotation matrix description in most of the cases due to its simplicity. This additionally motivate the use of the Newton-Euler methodology for the modeling of the MRAV dynamics. On the other hand, the Lagrangian formalism would have required the use of minimal representations for the orientation.

Using the Newton-Euler formalism, we can derive the dynamics of the aerial platform in order to relate the motion of its CoM, in particular its linear and angular accelerations ($\ddot{\mathbf{p}}$ and $\dot{\boldsymbol{\omega}}$, respectively), to the sum of the forces $\mathbf{f}_B$ and the torques $\boldsymbol{\tau}_B$ acting on this particular point of the rigid body. As traditionally done in the literature of aerial robotics, we express the translational dynamics in world frame, while keeping the rotational one in body frame. This allows to slightly simplify the form of the equations[7]. Combining them in a compact form, we obtain

$$\begin{bmatrix} m\mathbf{I}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{J} \end{bmatrix} \begin{bmatrix} \ddot{\mathbf{p}} \\ \dot{\boldsymbol{\omega}} \end{bmatrix} = \begin{bmatrix} -mg\mathbf{e}_3 \\ -\boldsymbol{\omega} \times \mathbf{J}\boldsymbol{\omega} \end{bmatrix} + \begin{bmatrix} \mathbf{R} & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{I}_3 \end{bmatrix} \left(\begin{bmatrix} \mathbf{f}_B^B \\ \boldsymbol{\tau}_B^B \end{bmatrix} + \begin{bmatrix} \mathbf{f}_{B\text{-ext}}^B \\ \boldsymbol{\tau}_{B\text{-ext}}^B \end{bmatrix} \right), \qquad (3.7)$$

[6]Note that this convention is equivalent to the *extrinsic roll-pitch-yaw*, which consist in successive rotations about the fixed axes $\mathbf{x}_W$, $\mathbf{y}_W$, and $\mathbf{z}_W$ with the angles φ, θ, and ψ.

[7]One of the resulting advantages is the fact that the inertia $\mathbf{J}$, which is expressed in $\mathcal{F}_B$, does not depend on the particular orientation of the vehicle. Conversely, it holds that $\mathbf{J}^W = \mathbf{R}\mathbf{J}\mathbf{R}^\top$.

where g is the gravitational acceleration and $\mathbf{I}_3 \in \mathbb{R}^{3\times3}$ is the identity matrix. The presence of the term $\boldsymbol{\omega} \times \mathbf{J}\boldsymbol{\omega}$, which represents the centrifugal and Coriolis term, is due to the fact that the rotational part of the dynamics has been expressed in body frame, as previously stated. In general, the time-derivative of a vector $\bullet \in \mathbb{R}^3$ as seen from a frame $\mathcal{F}_B$ rotating at angular velocity $\boldsymbol{\omega}$ w.r.t. the inertial frame $\mathcal{F}_W$ is [Goldstein–2002]

$$\left.\frac{d\bullet}{dt}\right|_{\mathcal{F}_B} = \left.\frac{d\bullet}{dt}\right|_{\mathcal{F}_W} + \boldsymbol{\omega} \times \bullet. \tag{3.8}$$

Remark (model generality). It might be interesting to note that model (3.7) could be used to represent the dynamics of a general AV, not only of a MRAV, as no assumption is made around the way the body wrench is generated. This will be done in the following, in order to focus the attention on MRAVs.

Remark (external wrench). The term $[{\mathbf{f}^B_{B\text{-ext}}}^\top \ {\mathbf{f}^B_{B\text{-ext}}}^\top]^\top$ accounts for the external wrench acting of the MRAV CoM, due to interaction with the environment (or other robots). As it can be seen, in (3.7) this term has been put to zero. Such approximation might be legitimated by the lack of contact between the aerial vehicle and the environment. Therefore, for *contact-less* operations, we can safely neglect this term. On the other hand, when targeting operation *in contact*, it will be crucial to properly measure or estimate this contribution, as we will see in Chapt. 8.

The body wrench generation for a MRAV

The model developed until (3.7) is general, in the sense that it properly describes the dynamics of an aerial vehicle and, more in general, of a floating rigid body. Thanks to its generality, this mathematical description will be used as the basis for the modeling of all the MRAVs presented in this thesis. In this representation, the motion of the platform is described in relation to the forces and the torques applied to its CoM. In order to focus our attention to multi-rotor designs, it is important to understand how their body wrench is physically generated. To do so, we shall explicit its dependence w.r.t. the forces generated by the actuators, which are the true responsible for the active motion of the platform.
As far as the force is concerned, it is the sum of the actuator forces $\mathbf{f}_i^{A_i} \in \mathbb{R}^3$, properly rotated in body frame, i.e.,

$$\mathbf{f}_B^B = \sum_{i=1}^{n} \mathbf{f}_i^B = \sum_{i=1}^{n} \mathbf{R}_{A_i}^B \mathbf{f}_i^{A_i} = \sum_{i=1}^{n} \mathbf{R}_{A_i}^B \mathbf{e}_3 \gamma_i. \tag{3.9}$$

On the other hand, the body torque is the result of the moments $\boldsymbol{\tau}_{\gamma_i}^B$ created by the actuator forces due to their leverage arms and the drag torques $\boldsymbol{\tau}_{d_i}^B$ which are a byproduct of the counteracting reactions of the air to the rotation of the propeller

blades. This can be mathematically written as

$$\begin{aligned}\boldsymbol{\tau}_B^B &= \sum_{i=1}^{n} \boldsymbol{\tau}_{\gamma_i}^B + \boldsymbol{\tau}_{d_i}^B = \sum_{i=1}^{n} \mathbf{p}_{A_i}^B \times \mathbf{f}_i^B + \bar{k}_i c_{f_i}^{\tau} \mathbf{f}_i^B \\ &= \sum_{i=1}^{n} \left([\mathbf{p}_{A_i}^B]_{\times} + \bar{k}_i c_{f_i}^{\tau} \mathbf{I}_3 \right) \mathbf{R}_{A_i}^B \mathbf{e}_3 \gamma_i. \end{aligned} \tag{3.10}$$

The constant parameter $c_{f_i}^{\tau} \in \mathbb{R}_{>0}$ is characteristic of the type of propeller and is defined as the intensity ratio (in absolute value) between the thrust produced by the propeller rotation and the generated drag torque. Furthermore, $\bar{k}_i$ is a variable whose value is equal to -1 (respectively, $+1$) in the case the shape of the $i-th$ propeller is meant for a counter-clockwise (respectively, clockwise) rotation w.r.t. $\mathbf{z}_{A_i}$. More precisely, it depends on the angle of attack of the propeller blade, see Fig. 3.2. Finally, the signed scalar value of the produced force $\gamma_i \in \mathbb{R}$ can be modeled in relation to the controllable spinning rate[8] $\mathrm{w}_i \in \mathbb{R}$ of motor i by means of the following equation:

$$\gamma_i = k_i c_{f_i} |\mathrm{w}_i| \mathrm{w}_i, \tag{3.11}$$

where $k_i = -\bar{k}_i$ and $c_{f_i} \in \mathbb{R}_{>0}$, called *lift factor*, is another propeller-dependent constant parameter to be experimentally identified. With reference to the force generation model (3.11), when a propeller is spun a at certain rate, it produces a force $\gamma_i \mathbf{z}_{A_i}$ called the *lift force*, which depends on the aerodynamic properties of the propeller blade. Furthermore, the resistance of the air generates also some tangential forces on the blade. These drag forces, properly integrated over the rotor, generate a *drag moment*, whose scalar value τ_{d_i} can be modeled as follows

$$\tau_{d_i} = \bar{k}_i c_{f_i}^{\tau} \gamma_i. \tag{3.12}$$

In particular, (3.12) models the fact that the drag torque always resists the actuator rotation ($k_i \bar{k}_i = -(\bar{k}_i)^2 = -1$). Finally, each rotor drag moment in turn generates the torque $\boldsymbol{\tau}_{d_i}^B = \tau_{d_i} \mathbf{R}_{A_i}^B \mathbf{e}_3$ on the vehicle CoM.

Note that this is a well-established model [Hamel–2002; Pounds–2010b; Mellinger–2011; Mahony–2012b; Ryll–2012b], that has been validated experimentally, e.g., in [Ryll–2015]. In particular, the model here, given by (3.9)-(3.12), is presented with more generality since it allows for both positive (CCW) and negative (CW) rotations w_i of the rotors w.r.t. $\mathbf{z}_{A_i}$, independently from the value $\bar{k}_i$ of the associated propeller, as illustrated in Fig. 3.2. Nevertheless, it should be highlighted that this model assumes that the propeller can exert the same force (in absolute value) when spun in the two senses. More realistically, the propeller blade shape is optimized to rotate in one particular sense, thus this assumption might result too

[8]The reader should not confuse the scalar symbol $\mathrm{w}_i \in \mathbb{R}$ ('double u') with vector $\boldsymbol{\omega} \in \mathbb{R}^3$ ('bold omega'). The former denotes the spinning rate of motor i, while the latter represents the aerial vehicle angular velocity (of $\mathcal{F}_B$ w.r.t. $\mathcal{F}_W$), expressed in $\mathcal{F}_B$.

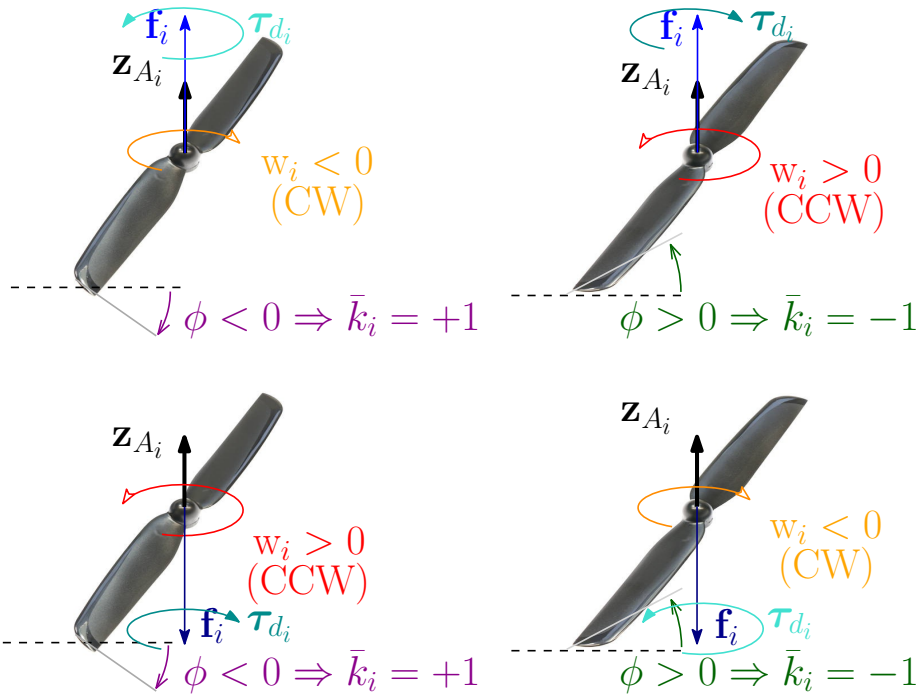


Figure 3.2 – View of two kinds of propellers and the possible rotation combinations.

simplistic. In that case, the value of the parameters c_{f_i} and $c^{\tau}_{f_i}$ might be considered as configuration-dependent. Another generality is represented by the fact that we allow also for different thrust generation models. For this reason in eqs. (3.9)-(3.10) the dependence is left on γ_i. Therefore, different models w.r.t. the one give by (3.11)-(3.12) might be used, just keeping the only assumption that the absolute value of the rotor drag moment produced by the propeller is proportional to the one of the thrust force through a constant coefficient, i.e., $|\frac{\tau_{d_i}}{\gamma_i}| = |\bar{k}_i c^{\tau}_{f_i}| = c^{\tau}_{f_i}$.

Once defined the vector $\boldsymbol{\gamma} = \begin{bmatrix}\gamma_1, \dots, \gamma_n\end{bmatrix}^{\top}$, we can compactly write

$$\underbrace{\begin{bmatrix}\mathbf{f}^B_B \\ \boldsymbol{\tau}^B_B\end{bmatrix}}_{\boldsymbol{w}^B_B} = \underbrace{\begin{bmatrix}\mathbf{G}_1 \\ \mathbf{G}_2\end{bmatrix}}_{\mathbf{G}} \boldsymbol{\gamma} \tag{3.13}$$

where $\mathbf{G} \in \mathbb{R}^{6\times n}$ is called the *allocation matrix*. The blocks $\mathbf{G}_1$ and $\mathbf{G}_2$ are used to define the effect of the actuator forces on the body force and torque, respectively. In particular, the j-th column of $\mathbf{G}$, $j \in \{1, \dots, n\}$, refers to the contribution of the j-th actuator force to the total body wrench, being

$$\mathbf{G}(:, j) = \begin{bmatrix}\mathbf{R}^B_{A_j}\mathbf{e}_3 \\ \left([\mathbf{p}^B_{A_j}]_{\times} + \bar{k}_j c^{\tau}_{f_j}\mathbf{I}_3\right)\mathbf{R}_{A_j}\mathbf{e}_3\end{bmatrix}. \tag{3.14}$$

In order to explicit the dependence of the system dynamics on the actuator forces, we shall plug (3.13) inside (3.7), thus obtaining

$$\underbrace{\begin{bmatrix} m\mathbf{I}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{J} \end{bmatrix}}_{\mathbf{M}_R} \underbrace{\begin{bmatrix} \ddot{\mathbf{p}} \\ \dot{\boldsymbol{\omega}} \end{bmatrix}}_{\mathbf{a}} = \underbrace{\begin{bmatrix} -mg\mathbf{e}_3 \\ -\boldsymbol{\omega} \times \mathbf{J}\boldsymbol{\omega} \end{bmatrix}}_{\mathbf{b}_R} + \underbrace{\begin{bmatrix} \mathbf{R} & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{I}_3 \end{bmatrix}}_{\mathbf{G}_R} \mathbf{G}\boldsymbol{\gamma} \tag{3.15}$$

Remark (actuators arrangement). In general, the actuators could be arranged in order to span a 3-dimensional volume, represented in Fig. 3.1 by the dark-green ellipsoid. On the other hand, it might be convenient, from a mechanical implementation point of view, to displace all the actuators on a common plane, as in the light-green ellipsoid in the same figure. However, it is easy to see that any shift in the position of each motor along the direction identified by the vector $\mathbf{f}_i^B$ does not affect the result of (3.10). In mathematical terms, this can be written as:

$$(\mathbf{p}_{A_i}^B + \delta \frac{\mathbf{f}_i^B}{||\mathbf{f}_i^B||}) \times \mathbf{f}_i^B = \mathbf{p}_{A_i}^B \times \mathbf{f}_i^B, \tag{3.16}$$

with $\delta \in \mathbb{R}$. This means that virtually any multi-rotor geometrie can produce the same body wrench of a configuration where all the actuator positions lay on the same plane. This property turns out to be very significant in the design phase, since it allows to obtain a MRAV with the same control authority and a simpler actuators disposition. The only difference with respect to the overall model will be in the form of the inertia matrix $\mathbf{J}$.

Wrench generation limits

The actuation model (3.13) considers the vector $\boldsymbol{\gamma}$ of the actuator force as the input of the system. This input is then mapped, by means of the linear map embodied by the allocation matrix $\mathbf{G}$, to the generated body wrench $\boldsymbol{w}_B^B$ that is actively used to steer the body CoM frame in the desired way. In this context, we analyze the limitations that might occur in the generation of $\boldsymbol{w}_B^B$. Such limitations are of crucial importance, since they can constrain the dynamic capability of the MRAV by preventing it to reach some state configurations of the six-dimensional space state. Therefore, the study of the allocation matrix is of dramatically important in the design of a MDT MRAV, as we will see in Sec. 6.1. Throughout this thesis, we distinguish between two kinds of wrench generation limits: *i) geometric* limits; and *ii) actuator* limits.

The *geometric* limits are the ones due to loss of rank in the allocation matrix $\mathbf{G}$. We can think about the allocation matrix as a function which maps vectors of actuator force intensities, living in a subset of an n dimensional space, to vectors of body wrenches, laying inside a subset of a six dimensional space. If we assume for a moment that the set of actuator forces can span the whole space $\mathbb{R}^n$ (which is already not realistic, as we will see in the following), we can identify two cases.

- The allocation matrix does not have full rank, i.e., $\text{rank}(\mathbf{G}) < 6$. In this case, we say that the MRAV has geometric limits, as it can not exert a body wrench in some direction, even in the case the actuator forces could span the whole space $\mathbb{R}^n$. In the literature of aerial robotics, a MRAV of this kind is often referred to as *under-actuated* [Hua–2013]. It is interesting to remark the fact that for an under-actuated MRAV it has a number of rank deficiencies equal to its under-actuation degree. In the particular case of a Uni-Directional Thrust (UDT) platform, we have that $\text{rank}(\mathbf{G}) = 4$, with $\text{rank}(\mathbf{G}_2) = 3$ and $\text{rank}(\mathbf{G}_1) = 1$. This reflects the robot capability to exert a body torque in all the directions (again, disregarding the actuator limits) but a body force along only one direction, i.e., the one identified by the $\mathbf{z}_B$ axis.
- The allocation matrix has full rank, i.e., $\text{rank}(\mathbf{G}) = 6$ In this case, the MRAV might exert a body wrench which can span the whole $\mathbb{R}^6$ space. In the literature of aerial robotics, a MRAV of this kind are often referred to as *fully-actuated* if $n = 6$, and as *redundantly-* (or *over-*) *actuated* if $n > 6$, cf. [Rajappa–2015; Ryll–2015].

It should be remarked that a robot is {under, fully, redundantly}-actuated w.r.t. a specific task, that in this case is the tracking of a 6-dimensional trajectory.

On the other hand, the actuator limits are the ones that restrain the set of feasible body wrenches independently from the rank deficiencies of the allocation matrix. These limits are a consequence of the fact that the vector of the feasible actuator forces realistically lives in proper subset of the n-dimensional space, i.e., $\boldsymbol{\gamma} \in \mathcal{A} \subsetneq \mathbb{R}^n$, with $\mathcal{A}$ being the Cartesian product of the scalar subsets $\mathcal{A}_i \subsetneq \mathbb{R}$ containing the feasible forces that each actuator can exert, i.e., $\mathcal{A} = \{\boldsymbol{\gamma} \in \mathbb{R}^n \,|\, \gamma_i \in \mathcal{A}_i, \mathcal{A}_i \subsetneq \mathbb{R}, \forall i = 1, \dots, n\}$. In particular, it is reasonable to assume that each actuator can exert a force which is limited between a lower bound $\underline{\gamma}$ and an upper bound $\overline{\gamma}$[9], i.e., $\mathcal{A}_i = \{\gamma_i \in \mathbb{R} \,|\, \underline{\gamma} \leq \gamma_i \leq \overline{\gamma}\}$. Thus, $\mathcal{A}$ results an hyper rectangle of dimension n, see [Furci–2018]. If $\underline{\gamma} < 0 < \overline{\gamma}$, the actuator is said to be *bidirectional*. On the other hand, if $0 < \underline{\gamma} < \overline{\gamma}$, the actuator is said to be *unidirectional*[10]. Bidirectional actuators are able to invert the direction of the lift force by inverting either the motor rotation w or the propeller angle of attack. However, such actuators have several issues: i) scarceness of reversible Electronic Speed Controllers (ESC) for brushless motors, ii) lower energetic efficiency compared to unidirectional rotors, iii) lower controllability of the exerted force at low speeds, and iv) extra mechanical complexity and increased weight and thus energy consumption (in case of variable pitch propellers).

For these reasons, all the MRAVs modeled in this thesis are assumed to have *unidirectional* actuators. In particular, we assume that the sense of rotation of each actuator is fixed and can not be reversed. Moreover, the collective pitch of the propeller blades is supposed constant and so the generated thrust can not be flipped. Thus, swash-plate designs are left out of the scope of this work. For each pro-

[9] In particular, it is legitimate to assume that such bounds are the same for all the actuators.

[10] The case $\underline{\gamma} < \overline{\gamma} < 0$ is also admissible, but can be obtained just by flipping the actuator.

peller, only the correspondent optimized sense of rotation will be employed. In other words, a propeller optimized to spin CCW ($k_i = +1$) will be spun only CCW ($\mathrm{w}_i > 0$). Conversely, a propeller optimized to spin CW ($k_i = -1$) will be spun only CW ($\mathrm{w}_i < 0$). Therefore, only the two combinations on the top of Fig. 3.2 are allowed. As a consequence, we always get $\gamma_i > 0$, which simplifies the notation and the force generation model. With this shrewdness, model (3.11) becomes

$$\gamma_i = c_{f_i} \mathrm{w}_i^2. \tag{3.17}$$

Finally, for simplicity all the actuators are assumed to be the same. Consequently, the dependence of the parameters $c_{f_i}, c_{f_i}^{\tau}$ from the subscript 'i' will be dropped.

Remark (terminology). Given the fact that $\boldsymbol{\gamma} \in \mathcal{A} \subsetneq \mathbb{R}^n$, even if a MRAV has a fully-ranked allocation matrix, it could not be able to exert a body wrench along all the directions of the six-dimensional space (or, in any case, it could not be able to exert *any* body wrench). For this reason, we believe that the nomenclature *fully-actuated* is misleading and for this reason we will not use it throughout this thesis. In order to preserve clarity and precision, we will use the acronym Fully-Ranked Allocation Matrix (FRA) to refer to those MRAVs which has a fully-ranked allocation matrix but, in general, limited actuator forces.

Remark (particular FRA MDT MRAVs). The theoretical problem of designing an Omni-Directional (OD) MRAV, that is a particular FRA MRAV that can produce an omni-directional wrench inside a limited six-dimensional shell centered in the origin, has been investigated in both in the case of bidirectional actuators [Brescianini–2016; Park–2018] and of unidirectional actuators [Tognon–2018b]. The design of this kind of MRAV, in the specific case of an octorotor platform, will be discussed in Sec. 6.2.

Model validity and limitations

The model defined by equations (3.14),(3.15) and (3.17) describes the dynamics of a generic MRAV with arbitrarily positioned and rotated actuators. Nevertheless, it contains, like all models, a certain degree of simplification w.r.t. the real system. First of all, it neglects the contribution of the *gyroscopic effect* induced by the conservation of the angular momentum of the propellers [Hamel–2002]. Furthermore, it does not take into account the *blade flapping* and the *rotor induced drag* reactions, whose effect arise in particular conditions as strong wind or at very high translational motion for the MRAV [Hoffmann–2007; Mahony–2012b; Bangura–2014b; Hua–2015b; Faessler–2018]. Moreover, it discards additional aerodynamic effects like the *ground* and the *ceiling effects*, which induce additional disturbance to the aerial vehicle when it flies close to a surface [Sanchez Cuevas–2017]. Finally, in the lack of additional constraints, the actuation model assumes that the actuators can generate the desired force (analogously, the rotor spinning velocity) instantaneously.

As far as the gyroscopic effect is concerned, its contribution could be taken into account by adding to the right-side part of the rotational dynamics in (3.7)

a modified version of (3) in [Bangura–2014b] that takes into account the fact that the actuators may have different orientations with respect to $\mathcal{F}_B$. As one can easily figure out, each term in such equation is scaled with $\mathbf{J}_{A_i}$, that is the inertia tensor of the rotating part of the i-th actuator (composed of the propeller and the rotor). For MRAVs with actuators of small-medium size, that are the ones on which this thesis focuses its attention, the entries of this matrix are typically 2-3 orders of magnitude smaller than the ones of $\mathbf{J}$. Therefore, the contribution of the gyroscopic effect can be safely neglected in (3.7). On the other hand, when dealing with vehicles with massive rotors and large propellers such an effect should be taken into account.

Regarding the blade flapping and the rotor induced drag effects, they are mainly associated with the flexibility and the rigidity of the rotors, respectively [Mahony–2012b], and are generated by the interaction of the air with the translating propellers. The results of these aerodynamic effects can be typically observed in UDT MRAVs as exogenous lateral forces in the x-y plane of the rotors. In the case of a MRAV with arbitrarily tilted propellers, this analysis would be complex to be precisely evaluated and would require to model also the possible interactions between the airflow of different propellers, which is outside the scope of this thesis. For these reasons, in the line of [Mueller–2013a; Kamel–2015] and many other relevant works, further motivated by the results presented in [Ryll–2015], we decided to neglect the first-order contribution of these two reactions and all other second-order effects arising at very high speed and highly dynamic MRAV maneuvers. As far as the ground and the ceiling effects are concerned, their contribution is assumed to be negligible in the domain of interest.

The hypothesis of instantaneous control for actuation force (analogously, for the rotor velocity) would require an infinite motor torque, which is practically unfeasible. In order to take into account more realistic actuation limits, one should extend the model with the dynamics of the motor, considering both mechanical and electrical effects [Bangura–2017], together with the dynamics of the Electronic Speed Controller (ESC). Such topic is discussed also in Sec. 4.2. Nevertheless, for our control design purposes, we can assume that the actuator force variations are limited in the domain of interest. Under this assumption, and thanks to the use of a particular ESC control algorithm [Franchi–2017], which guarantees minimal response time, the actuation model results a decent approximation of the real behavior. In this thesis, the previous hypothesis will be implicitly assumed in the simplified model presented in Sec. 4.1. However, it will be abandoned in Sec. 4.2, thanks to the use of a more detailed MRAV model, in the view of addressing the stable tracking control of dynamic maneuvers.

All the previous choices are in accordance with the typical *trade-off* between accuracy and simplicity that the control designer faces in modeling phase. Indeed, a good model for control should be able to catch the most important dynamics of the system, while avoiding to add unnecessary over-complexities. This is particularly true in the case one wishes to use a predictive controller (see Sec. 4.2, Sec. 5.2 and Sec. 7.2) to automatically drive the aerial vehicle, given the fact that the optimization problem has to be solved as fast as possible.

The importance of MDT hexarotors

The goal here is to steer the reader's attention towards the important class of MDT hexarotors, i.e., MRAVs with six arbitrarily oriented (and positioned) rotors. These platforms, around whose design most of the contributions presented in this thesis are built, have particular properties that aroused growing interest in the scientific community and motivated a good number of recent research work in the literature of aerial robotics, among which [Voyles–2012; Voyles–2014; Rajappa–2015; Park–2016; Kamel–2018].

As already mentioned in the previous introductory chapters, the most common multi-rotor design is epitomized by the UDT quadrotor, which presents four collinear actuators. Restraining the attention to UDT MRAVs, the addition of more actuators w.r.t. the quadrotor case is useful only in terms of an *increased payload*, since all the other dynamic properties remain the same, as it can be easily assessed by an analysis of the achievable body wrench. In particular, the addition of two (or more) columns in the allocation matrix does not modify its rank, in this case.

On the other hand, when considering more general designs which bestows on the MRAV the capability of exerting a body frame force in multiple directions, the addition of more actuators involves at least two relevant properties: *i)* the partial *decoupling* between position and orientation tracking; and *ii)* the *robustness* towards a rotor failure. As far as *i)* is concerned, it is possible to design the actuator arrangement in order to partially get rid of the strong coupling that characterizes the dynamics of any UDT platforms. Furthermore, for certain actuator arrangements it is also possible to obtain an allocation matrix with full rank (FRA design), as we will show in Sec. 6.1. In particular, this is achievable, in the case of a MDT hexarotor, with the minimum number of actuators, i.e., six. As already mentioned, the possibility to decouple the force and the torque actuation becomes crucial in view of performing aerial physical interaction tasks, as it will presented and discussed in Part III. In a complementary way, with reference to property *ii)*, it is desirable that the MRAV maintains some degree of controllability in the case of the failure of one actuator, which is likely to happen in a real scenario. In relation to this property, recent theory in the literature of aerial robotics has been developed around this topic [Du–2015; Giribet–2016a; Michieletto–2017]. It is known that both a UDT quadrotor and a hexarotor are uncontrollable in any case of a rotor failure [Achtelik–2012]. On the other hand, in [Michieletto–2018] it has been theoretically proven and experimentally validated that, under some design assumptions, a FRA hexarotor is capable of performing static-hovering, i.e., to control the linear and the angular velocities to the origin and the pose to a constant configuration, after the loss of one actuator.

In view of these considerations, we decided to focus our attention mainly in the design of FRA hexarotor platforms to attain the final goal of performing safe and robust aerial physical interaction tasks. Nevertheless, it should be underlined that the modeling and the control techniques presented in this thesis are non-specific and are applicable to MRAVs with any number of actuators.

Inverse Dynamics control of a MRAV

After having presented in Sec. 3.1 the MRAV model used as a starting base in this thesis, we now briefly outline one possible strategy to control its dynamics. Such strategy, called *Inverse Dynamics* control or *Feedback Linearization*, is very well-known in the literature of non-linear control theory [Oriolo–2002; Martin–1996; De Luca–1998] and has been presented in the quite recent literature of aerial robotics and applied to FRA MRAVs [Rajappa–2015]. In the following, we briefly illustrate through an example the controller idea in this specific case, applying a *static* feedback linearization of the model dynamics, in order to show its limitations. This example will be used to motivate the development of more suitable model descriptions and control strategies, which will be presented in Part II of this thesis. For more thorough insights on the topic of the feedback linearization, the interested reader is referred to [Slotine–1991; Khalil–2002; Isidori–2013].

Solving (3.15) for the vector $\mathbf{a}$ of linear and angular accelerations, we obtain

$$\mathbf{a} = {\mathbf{M}_R}^{-1}(\mathbf{b}_R + \mathbf{G}_R\mathbf{G}\boldsymbol{\gamma}). \tag{3.18}$$

According to the fact that, for a FRA MRAV, the allocation matrix $\mathbf{G}$ is full-rank, it is possible to choose the following inverse dynamics control law:

$$\boldsymbol{\gamma} = \mathbf{G}^{\dagger}\mathbf{G}_R^{-1}(\mathbf{M}_R\mathbf{v} - \mathbf{b}_R), \tag{3.19}$$

which brings the system (3.18) in the following linear and decoupled form:

$$\mathbf{a} = \mathbf{v}, \tag{3.20}$$

where $\mathbf{v} \in \mathbb{R}^6$ is a six-dimensional virtual input which can be assigned in order to steer the six degrees of freedom of the platform along the desired trajectory, and to ensure robustness of the control scheme in a certain extent. A classic way to compute this term is by means of a linear controller with Proportional, Integrative and Derivative (PID) terms on the state errors, computed thanks to the feedback, plus a feed-forward action. Therefore, we see that the inverse dynamics approach requires two phases. First, a virtual control wrench $\mathbf{v}$ is computed in order to track the desired trajectory by canceling the nonlinear dynamical effects and trying to zero the position and orientation errors. Secondly, the thrust forces for each actuator are computed from the control wrench by simply pseudo-inverting the control allocation matrix, which is supposed to be full-ranked. Such strategy has been fostered, e.g., by [Rajappa–2015; Brescianini–2016; Park–2016].

Fig. 3.3 shows two realistic simulations of a FRA MDT hexarotor (the Tilt-Hex, which will be presented in Sec. 6.1) performing the tracking of two trajectories using the inverse dynamics approach. The first one, whose plots of position, orientation and rotor spinning velocities are depicted on the left column, is characterized by a desired constant position and a desired pitch angle which varies sinusoidally. The second one, represented on the right column, is defined by a desired orientation

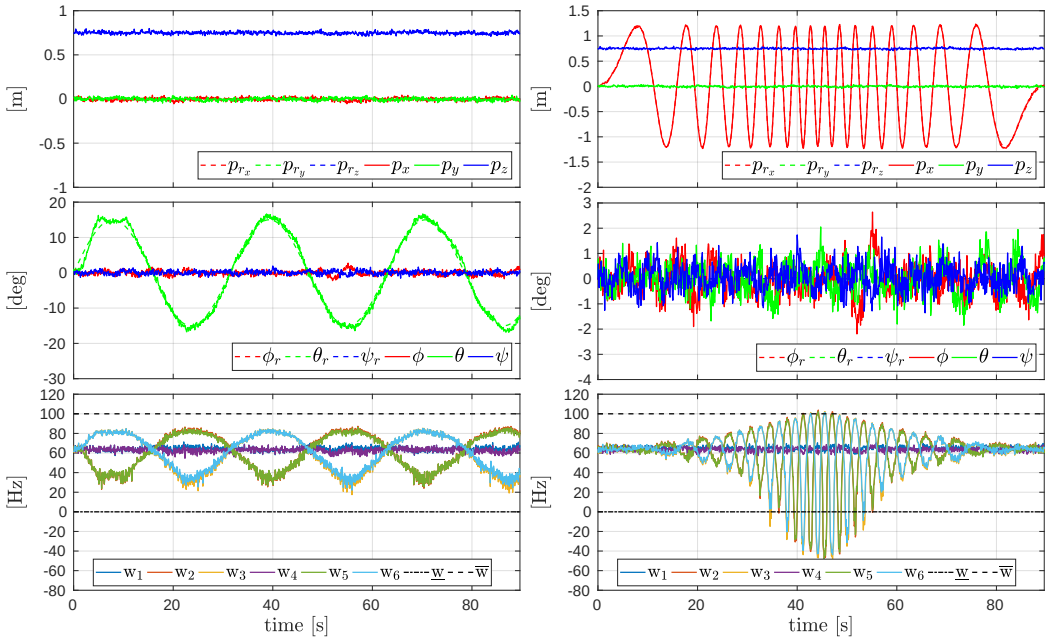


Figure 3.3 – Two realistic simulations (noise and model uncertainty included) of a FRA MDT hexarotor (the Tilt-Hex) controlled via the inverse dynamics approach. *From top to bottom:* platform position, orientation (shown as roll-pitch-yaw just for ease of understanding) and rotor spinning velocities. *Left column:* the Tilt-Hex tracks a trajectory in which the position is constant and the roll varies sinusoidally. *Right column:* the Tilt-Hex tracks a trajectory in which the position follows a chirp signal along the first axis, while the orientation keeps being constant. Both trajectories would be unfeasible for any UDT MRAV due to their geometric limits.

which is constantly flat and a desired position which follows a sine signal with time-varying frequency on the first axis. In particular, it should be remarked that such 6-dimensional motions are nicely tracked by the MDT hexarotor, while they would be unfeasible for UDT MRAVs, due to the geometric limits previously described.

The problem with the actuator limits

The inverse dynamics control assumes that the virtual input $\mathbf{v}$ could be assigned *at will* in order to *independently* steer the six DoFs of the platform along *any* trajectory. Nevertheless, this would be true only if the vector of the controlled forces could span the whole n-dimensional space, i.e., $\boldsymbol{\gamma} \in \mathcal{A} = \mathbb{R}^n$, while, as we have seen in the previous section, such vector can realistically take values only inside a proper subset of that space, i.e., $\boldsymbol{\gamma} \in \mathcal{A} \subsetneqq \mathbb{R}^n$.
Therefore, it is straightforward to figure out that the main limitation of the inverse dynamics approach is that, as it is, it does not account for the input saturations. This fact can easily generate instability in the MRAV motion when unfeasible inputs are required in order to perfectly track a given trajectory. This downside can be appreciated in the simulation plots of Fig. 3.4, where the same reference motion

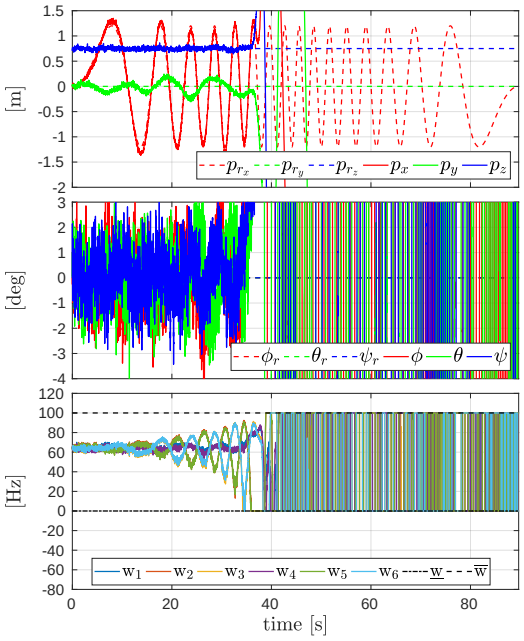


Figure 3.4 – A realistic simulation (noise and model uncertainty included) of a FRA MDT hexarotor (the Tilt-Hex) controlled via the inverse dynamics approach, with the addition of input saturation. While the trajectory is the same of Fig. 3.3-*right*, i.e., with a chirp signal along the first axis of the position and a constant orientation, the sole inverse dynamics control makes the system unstable. Therefore, this trajectory is unfeasible also for this FRA MDT hexarotor, due to its actuator limits.

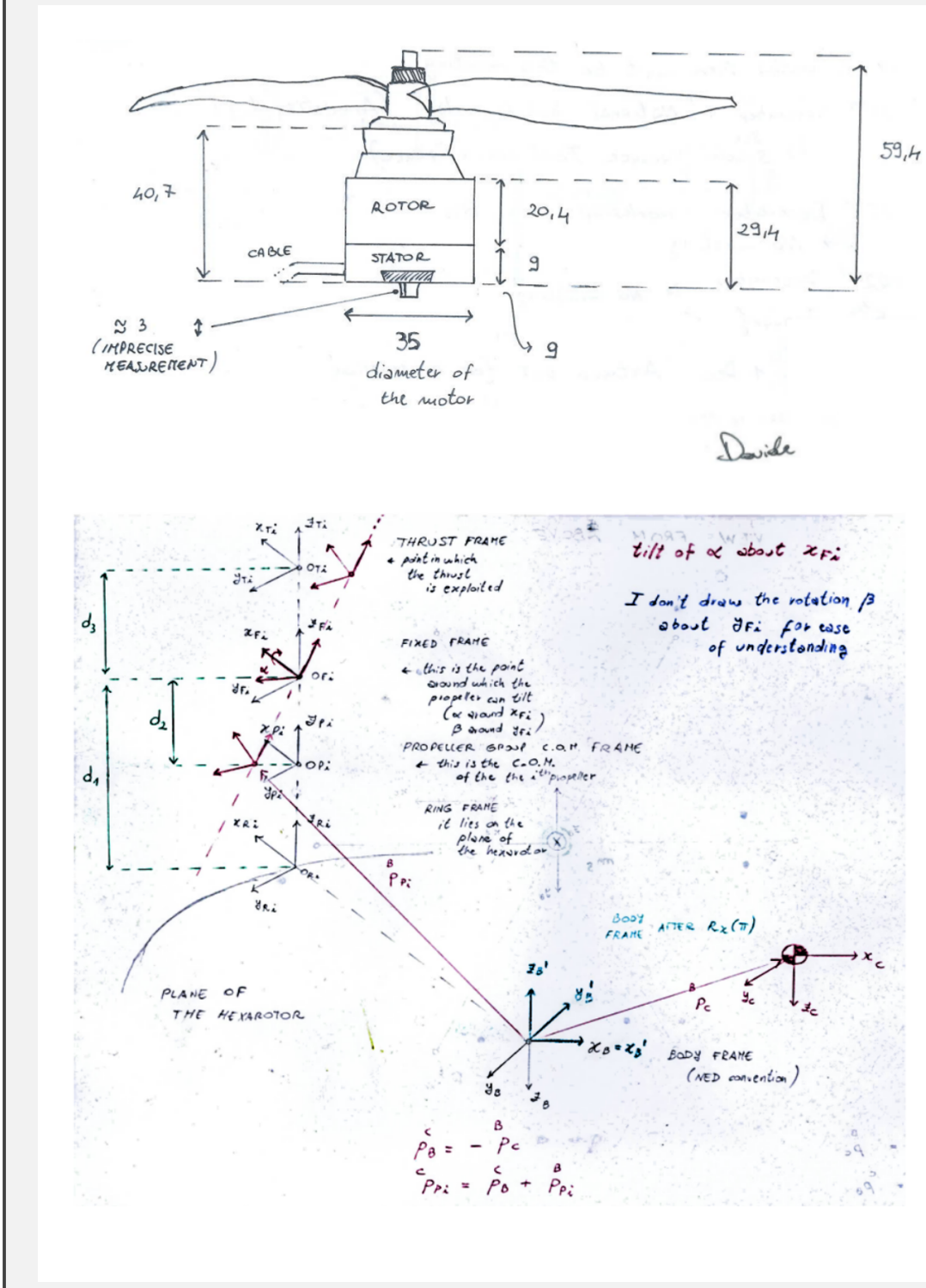
profile of the one shown in Fig. 3.3-right is used, but with the introduction of input saturations. Since the controller is not aware about the system limitations, instability is produced. While the trajectory is unfeasible for a UDT MRAV due to its geometric limits, it results also unfeasible for a FRA MDT, due to its actuator force constraints. This suggests that both kinds of wrench generation limits are equally important and should be considered both in the design and in the control phases of a MRAV.

Remark (Additional rotor acceleration constraints). It should be clear from the previous example, that the sole inverse dynamics technique is not suitable for the control of MRAVs with rotor velocity constraints (equivalently, actuator force constraints). As a consequence, such strategy is *a fortiori* not suitable to deal with additional rotor acceleration constraints (equivalently, actuator force constraints) that characterize a real vehicle. This issue will be tackled in Sec. 4.2 and Sec. 5.2.

A preliminary attempt to solve the issues of the inverse dynamics approach in a special case has been done in [Ducard–2011], where a weighted pseudo-inversion of the allocation matrix is used to allocate the 6 inputs of a UDT hexarotor in order to obtain a lower dimensional task (4D) output (the total thrust and the 3 moment components). The design of a control strategy for the tracking of 4D trajectories with MDT MRAVs has instead been addressed in [Kendoul–2006]. Despite the interesting solution to deal with the *over-actuation* w.r.t. the task to be fulfilled, the problem of actuator saturations is not taken into account in this case. Another control approach for a MDT MRAV is presented in [Romero–2007], which is however specific to the octorotor platform designed therein, it does not consider input bounds either, and is based on a particular Euler angle representation. In [Convens–2017], the authors present a solution that also uses a Euler representation and is based on

the reference governor. The method is tested only in simulation and for constant position and attitude references. Finally, in [Hua–2015b] the authors addressed the nonlinear control of a MDT MRAV for the tracking of 6D trajectory, validating the approach by means of numerical simulations of a quadrotor with synchronized tilting actuators. In this case, the actuation limits are taken into account by modeling the existence of maximum angle between the body-force vector and $\mathbf{z}_B$. In such work, the authors propose a control method that involves a primary objective associated with the linear velocity tracking, and a secondary objective associated with the angular velocity tracking.

Starting from an idea similar to [Hua–2015b], in Sec. 5.1 we will present a nonlinear Lyapunov-based control method for MRAVs with actuation constraints, which will be then validated in Sec. 7.1 with the real aerial prototypes that we have conceived and built at LAAS–CNRS. The design of these platforms will be described in Sec. 6.1. Throughout Part II, the issue of the actuation constraints for MRAVs will be addressed with different levels of abstraction. Indeed, in Sec. 4.2 we will develop a model that takes into account also limits on the rotor accelerations, thus introducing a more detailed description of aerial vehicles. A numerical controller that is based on the prediction horizon strategy and relies on the detailed MRAV model will be presented in Sec. 5.2. Experimental results with real MRAV prototypes will be presented also in this case, in Sec. 7.2. Once presented the theory for the control of MDT MRAVs during contact-less operations and its experimental validation, in Part III we will target the topic of aerial physical interaction, which involves conditions of contact between the aerial vehicle and the environment. Finally, the impact of the presented results will be demonstrated and underlined in Part IV, which presents the industrial technology transfer and concludes the thesis.


59,4
40,7
ROTOR
20,4
29,4
CABLE
STATOR
9
≃ 3
(IMPRECISE
MEASUREMENT)
35
diameter of
the motor
9
Davide
THRUST FRAME
← point in which
the thrust
is exploited
FIXED FRAME
← this is the point
around which the
propeller can tilt
(α around x_{Fi}
β around y_{Fi})
PROPELLER GROUP C.O.M FRAME
← this is the C.O.M.
of the the i-th propeller
RING FRAME
it lies on the
plane of
the hexarotor
tilt of α about x_{Fi}
I don't draw the rotation β
about y_{Fi} for ease
of understanding
d_3
d_2
d_1
PLANE OF
THE HEXAROTOR
BODY
FRAME
AFTER $R_x(\pi)$
BODY FRAME
(NED convention)
P_{Pi}
P_c
$P_B = -P_c$
$P_{Pi} = P_B + P_{Pi}$

Drawing I

Part II

MDT-MRAVs with actuation constraints

Chapter 4

Modeling approaches for MDT-MRAVs with actuation constraints

"All models are wrong, but some are useful."
George E.P. Box

In this chapter we tackle the modeling of the wide class of Multi-Directional Thrust (MDT) Multi-Rotor Aerial Vehicles (MRAVs), in the realistic case in which the actuation capabilities of the vehicles are limited, due to either geometric or actuator force limits. In the following, two approaches are identified in order to do that. The first one allows to define a model with a very general and simple form, thus granting a consistent level of applicability to different Aerial Vehicles (AVs), not only MRAVs. On the contrary, the second one allows for a more precise and realistic description of MRAVs, at the cost of a more in-depth analysis of the platform actuator constraints. In the following, we present the two methods in detail and, finally, compare their benefits and drawbacks.

Simplified model of MRAVs with LBF constraints

The goal of this section is to define a simple model yet capable to describe well enough the class of MDT MRAV with actuation constraints. In order to do this, we propose to start from model (3.7) and perform an *abstraction* on the input abstraction level, as indicated in the following.

As we have seen in the first chapters of Part I, arranging the MRAV actuators in a general way allows the robot to exploit the MDT capability, i.e., the possibility to deviate the orientation of the body force from its typical direction $\mathbf{z}_B$. However, in order to minimize the waste of energy caused by the appearance of internal forces, the maximum component of the body force along the lateral directions, i.e., the ones which lays on the $\{\mathbf{xy}\}_B$ plane, is typically kept (by design) much smaller than the maximum allowed component along the $\mathbf{z}_B$ axis, where the platform can exert most of its force control authority. In this context, we define the class of Laterally-Bounded input Force (LBF) aerial vehicles as the set of aerial platforms characterized by the aforementioned body wrench limitations. As it is evident from the definition, the actuation constraints related of a LBF MRAV are not expressed

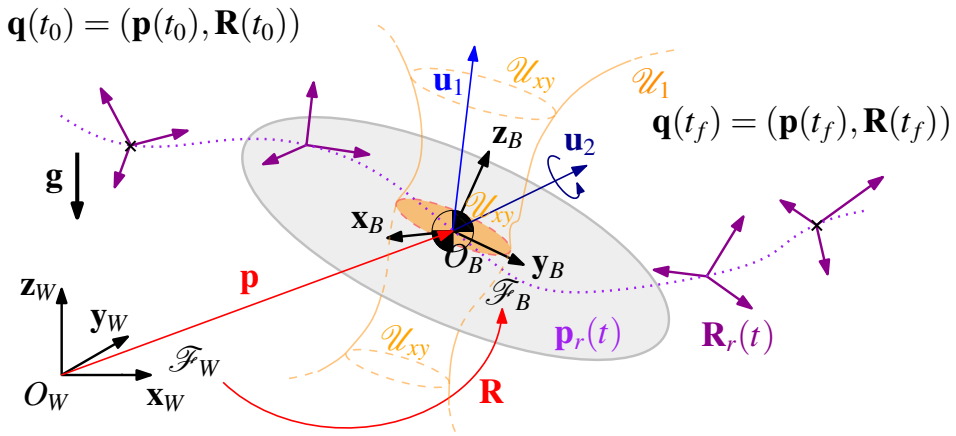


Figure 4.1 – A drawing illustrating the main quantities of an LBF AV, the main frames involved, the laterally bounded input sets and the full-pose 6D reference trajectory. The shape of $\mathcal{U}_{xy}$ could depend, in general, from u_3.

w.r.t. the geometric limits, neither w.r.t. the forces generated by the actuators, but rather w.r.t. the generated body wrench. This constitutes an abstraction of the real system, which is done to make the model general but still effective. Thanks to this shrewdness, there is no need to explicitly model *how* the body wrench is generated. Therefore, the concept of allocation matrix does not need to be introduced for the model of LBF MRAV. This is way the class of LBF platforms could legitimately encompass the one of general AVs, not only of MRAVs.

Once defined the control inputs $\mathbf{u}_1 = [u_1 \ u_2 \ u_3]^\top \in \mathbb{R}^3$ and $\mathbf{u}_2 = [u_4 \ u_5 \ u_6]^\top \in \mathbb{R}^3$ as the input force and moment applied to the vehicle expressed in $\mathcal{F}_B$, for a LBF MRAV the following constraint applies:

$$[u_1 \ u_2]^\top \in \mathcal{U}_{xy} \subset \mathbb{R}^2, \tag{4.1}$$

where the *laterally bounding* set $\mathcal{U}_{xy}$ is a set that contains the origin. We define

$$\mathcal{U}_1 = \{\mathbf{u}_1 \in \mathbb{R}^3 \mid [u_1 \ u_2]^\top \in \mathcal{U}_{xy}\}. \tag{4.2}$$

The control inputs $\mathbf{u}_1$ and $\mathbf{u}_2$ correspond, respectively, to the quantities $\mathbf{f}_B^B$ and $\boldsymbol{\tau}_B^B$ defined in (3.7). The use of a different symbols in this context has been adopted to make explicit that this model consider as control input the body wrench, that is at a higher level of abstraction w.r.t. the actuator forces. Note that $\mathcal{U}_{xy}$ can be constant or even be changing depending of u_3, as shown in Figure 4.1. Also remark that (4.1) is the only explicit constraint for a LBF MRAV. In general, the other components of the wrench, e.g., the force component along $\mathbf{z}_B$, could be even assumed unbounded, as done in most of the works in the literature.

Remark (LBF class of MRAVs). The class of LBF MRAVs is a subclass of the MDT MRAVs one with particular actuation limits, i.e., with a constrained set of lateral body forces. In particular, such class encompasses the one of standard Uni-

Table 4.1 – Main Symbols used in the modeling of a LBF MRAV.

Definition	Symbol
Control force applied at the Center of Mass (CoM) expressed in $\mathcal{F}_B$	$\mathbf{u}_1$
Control moment applied at the CoM expressed in $\mathcal{F}_B$	$\mathbf{u}_2$
Feasible set of the control force $\mathbf{u}_1$	$\mathcal{U}_1$
Feasible set of the projection of $\mathbf{u}_1$ on the xy plane in $\mathcal{F}_B$	$\mathcal{U}_{xy}$
i-th vector of the canonical basis of $\mathbb{R}_3$ with $i = 1, 2, 3$	$\mathbf{e}_i$
Reference position for $\mathbf{p}$ at time t	$\mathbf{p}_r(t)$
Reference rotation matrix for $\mathbf{R}$ at time t	$\mathbf{R}_r(t)$

Directional Thrust (UDT) MRAVs, for which the body force is possible *only* along the principal direction. Indeed, this should be considered as a degenerate case for which the set of lateral forces coincides with the empty set.

A UDT LBF platform can not track a generic full-pose trajectory, i.e., with independent position and orientation in SE (3). Indeed, the rotation about any axis that is orthogonal to the principal fixed total thrust direction must follow the evolution over time of the position trajectory, according to the well-known differential flatness property [Mistler–2001; Mellinger–2011; Faessler–2018]. Therefore, a UDT MRAV can only track a 4D-pose trajectory (i.e., position plus the rotation about the main body force direction).
On the contrary, a (not degenerate) MDT MRAV platform can exert some force in the lateral direction thus allowing the tracking of *some* full-pose (6D) trajectories. However, due to the bounded thrust along the lateral directions, it is not possible to track *any* full-pose trajectory. The larger the bounds the higher the ability of the platform to track a bigger set of full-pose trajectories, the lower the bounds the more the platform resembles a UDT MRAV and thus it becomes almost unable to track a full-pose trajectory, allowing only the tracking of a 4D-pose one.

In view of these considerations, an LBF MRAV is modeled as a rigid body to which gravity and control generalized forces are applied. A sketch of the most relevant quantities in Fig. 4.1. For the sake of completeness, all the main symbols used in this section, which have not been already defined, are summarized in Tab. 4.1. Consequently, the dynamics of a LBF platform can be modeled by the following Newton-Euler equations:

$$m\ddot{\mathbf{p}} = -mg\mathbf{e}_3 + \mathbf{R}\mathbf{u}_1 \tag{4.3}$$

$$\mathbf{J}\dot{\boldsymbol{\omega}} = -\boldsymbol{\omega} \times \mathbf{J}\boldsymbol{\omega} + \mathbf{u}_2 \tag{4.4}$$

$$\mathbf{u}_1 \in \mathcal{U}_1. \tag{4.5}$$

which are the same of the ones given by (3.7), with an additional force constraints.

Detailed model of MRAVs with actuator constraints

Differently from the previous section, the goal in this context is to define a more detailed MRAV model, moving the input abstraction level one step beyond the level of the actuator forces, i.e., towards the opposite direction w.r.t. the one followed in (4.1)-(4.4). Despite having the (same) goal of describing the dynamics of MDT MRAV with actuation constraints, this methodology aims to produce a more particular and detailed model of the platform dynamic constraints w.r.t. the previous one. This is achieved by focusing the attention of the actuators, the real responsible means for the MRAV motion. In particular, we drop here the simplistic assumption, already discussed in Part 3, that the actuator forces (equivalently, the rotor spinning velocities) can be instantaneously controlled. In order to do to this, this model consider as control input the derivatives of the actuator forces, i.e., $\mathbf{u} = \dot{\boldsymbol{\gamma}}$, as clarified in the following.

On the input abstraction level and the related control problem

Driven by the aforementioned argumentation, in order to define the detailed model we make here an effort towards the investigation of the most well-fitting placement for the input abstraction level. More precisely, we try to answer the questions: *i) Which physical effects should be included in the model? ii) What part of the model should be considered as the control input?* In the ideal case, the control inputs should be the voltage applied to each motor and the model should take into account the dynamics of the actuators. The state of such model should incorporate the spinning velocities of the propellers and the currents of the motors (such sub-model would be of the second order) with the corresponding saturations. In the view of using the model for control purposes, which corresponds to most cases in robotics, in this case the actuator dynamics and saturations would be taken into account by an omniscient nonlinear controller that could completely exploit the robot dynamics until the bounds of the physical limits (not some “fictitious" ones).

As also claimed in [Bemporad–2009], further improvements with respect to the current state of the art can be attained by extending the nonlinear model in order to include the motor/blade dynamics and treating the motor voltages as the commanded inputs. The problem associated with such highly detailed model lies in the actual implementation of the related controller. As a matter of fact, it should be able to: i) command the voltage to each motor, i.e., by Pulse Width Modulation (PWM) control or similar techniques; ii) measure their spinning velocity (not always possible); iii) measure their current[1] (even less possible and reliable); iv) control the motors at a very high rate ($\geq 1\,\mathrm{KHz}$) and at a very low latency ($\leq 1\,\mathrm{ms}$); v) have a model of the motor/blade dynamics (and not only of the multi-rotor) precise enough to encase its main effects; vi) accurately model the big nonlinearities introduced by the *active braking*, that is a technique to improve the motor deceleration time by

[1] In particular, the controller should precisely enforce the current saturation, since this is of paramount importance in order not to burn the motors.

producing a counteractive torque, since many Electronic Speed Controller (ESCs) use it. So far, at the best of our knowledge, nobody in the literature has ever implemented such a controller, mainly because of point iv), which is not attainable for the actual technology of the solvers and for the computational capacity of the processing units, despite the impressive advancement they both underwent. Nevertheless, we will show that we can approach such an implementation thanks to a proper model of the control input.

In [Geisert–2016] a trade-off solution is proposed, considering as control input the motor accelerations, while the spinning velocities are included in the state. This allows to put constraints on both the motor velocities and their derivatives. Doing so, the simplistic hypothesis that the spinning velocities of the rotors (and the generated forces, by consequence) can be changed instantaneously, implicitly done by most of other works in the literature, is here abandoned. Constraints on the motor accelerations between a minimum and a maximum constant value are then enforced. However, as corroborated by experimental data, the capability of the motor to accelerate depends on the motor current, the blade dynamics and, indirectly, on the actual velocity set-point. Furthermore, the use of the active-braking or other effects hidden in the electrical level could induce an asymmetry between the acceleration and the deceleration constraints of the motors.

Motivated by the preliminary foundations laid by these two contributions, but with the intent of defining more realistic and suitable constraints for better incorporating the electrical and aerodynamic constraints of the actuators, still without explicitly modeling them due to all the aforementioned difficulties, we propose the following strategy:

a) Consider as control input for the detailed model the derivative of the forces generated by each actuator of the MRAV. Choosing as input the force derivatives easily allows to model the fact that the force command can not be discontinuous or arbitrarily changed. Furthermore, by modeling such input as a function of the spinning velocity and the acceleration of the motors, we can we can keep an explicit dependence of the model w.r.t. the produced forces, which are the true responsible for the robot motion, and their derivatives. This has the clear advantage to let the user specify the desired model for the force generation, without imposing the use of a particular one.

b) Once integrated the control input and transformed into a desired velocity command, thanks to the chosen force generation model, e.g., the one given by (3.17), endow it as reference to a separate robust controller for the actuator spinning velocities, e.g., [Franchi–2017], which runs on-board the brush-less controllers, implements a closed loop control (differently from most regulators commercially available) and possibly exploits the active-braking. By doing so, we can delegate the reading of the motor current and velocity to the ESCs, that can achieve it with low latency ($\approx 1\,\text{ms}$) and safely, i.e., without the risk to burn the motors.

c) Finally, connect the two worlds just described (pose controller and low-level rotor velocity controller) through an interface represented by the modeling of the maximum force changing rate producible by the actuators (function of their maximum accelerations) which allows to have a safe and good tracking, once mapped on the generated thrust using the force generation model. In other words, we identify the input constraints which guarantee to safely comply with the current limitations and to bound the error on the generated force below a given threshold. The identification of such constraints is performed empirically on the real robot actuators, for different working conditions. Through this procedure, we identify the state-dependent input region which allows to safely drive the MRAV in an effective way.

The advantage with the proposed modeling strategy is that the related controller does not have to read the motor current and spinning velocity, and can assume that the desired actuator forces (converted into spinning velocity commands) are properly tracked within a certain precision inside the input feasibility region, thanks to the model of the 'good tracking'. It should be remarked that this allows to consider the low-level actuator control operated by the ESCs as a *black-box*. Ultimately, the reasoning described by points a) and b) corresponds to identify the rotor dynamics with a closed-loop model of first order with respect to the forces generated by the actuator (equivalently, the rotor spinning velocities), while estimating the state-dependent subset of inputs that guarantee the respect of such model. So far, a similar technique has been applied only at higher levels, i.e., for the rotational dynamics, as discussed in Chapt. 2, and besides, imposing constraints that were conceptually far from the real robot limits.

Moreover, it should be appreciated that the proposed approach allows to use a single-stage controller for the tracking of both the translational and the rotational dynamics. Avoiding the typical cascaded paradigm made by many works in the literature allows to exploit in a better way the dynamic capabilities of any arbitrarily-designed MRAV. Indeed, the platform flight envelope is not heavily constrained any more by artificial limits like, e.g., a maximum angular velocity or any maximum rotational angles for the body frame.

State-dependent actuator bounds

In this context, we outline the proposed identification procedure to define state-dependent limits of the motor force derivatives from available data. More specifically, first we experimentally assess how the spinning rate w_i and the acceleration $\dot{\mathrm{w}}_i$ of the motors (each of which is regulated by an independent embedded low-level controller), should be properly constrained in order to prevent the risk of damaging the motors and to guarantee an accurate force tracking. Secondly, we derive the state and input constraints, for $\boldsymbol{\gamma}$ and $\dot{\boldsymbol{\gamma}}$, respectively, exploiting an extension of the force generation model. In the following, we expose how the identification procedure should be applied to a generic actuator, while in Chapt. 7 we present its application to a particular hardware setup. From now on in this section, in order

to prevent the notation to become too cumbersome, we will omit the dependence of the equations from the subscript 'i', since all actuators are assumed to have the same model. The extension to the case of $n_d^\star \leq n$ dissimilar actuators is easily found by applying the procedure on each actuator sequentially.

Experimental assessment of the limitations on the spinning rate w and the acceleration $\dot{\mathrm{w}}$ of the motors

As far as the constraints on the minimum and maximum motor spinning velocities i.e., $\underline{\mathrm{w}}$ and $\overline{\mathrm{w}}$, are concerned, they can be experimentally identified by producing velocity commands that cause the current to be at the safety limits (information available from the motor and ESC data-sheets), with a certain security margin. Such limits should be combined with the ones imposed by the ESC, if any[2].
In order to proceed with the estimation of the constraints on the motor accelerations, i.e., , $\underline{\dot{\mathrm{w}}}$ and $\overline{\dot{\mathrm{w}}}$, let us first model the error e_f in the generated force w.r.t. a given desired value f_d by exploiting the force generation model, e.g., the one given by (3.17). The velocity error e_{w} due to wrong velocity estimation and other non-ideal effects in the low-level controller can be modeled as the difference between the desired value w_d and the actual velocity w, that is

$$e_{\mathrm{w}} = \mathrm{w}_d - \mathrm{w}. \tag{4.6}$$

Plugging (4.6) into (3.17), we model the actual generated force as $\gamma = c_f(\mathrm{w}_d - e_{\mathrm{w}})^2$ and its associated force error as

$$\begin{aligned} e_f(e_{\mathrm{w}}, \mathrm{w}_d) &= \gamma_d - \gamma \\ &= c_f\ (2\mathrm{w}_d e_{\mathrm{w}} - {e_{\mathrm{w}}}^2). \end{aligned} \tag{4.7}$$

In particular, we stress the fact that another model for the force generation may be used in place of (3.17). This would result just on a different form for (4.7).

The idea of the experiment we run in order to identify $\underline{\dot{\mathrm{w}}}$ and $\overline{\dot{\mathrm{w}}}$ is to provide the actuator with a control signal that produces increasing accelerations of the rotor at different velocity set-points and to evaluate the quality of the force tracking. By inspection of the error forces corresponding to the acceleration intervals associated with each velocity set-point, the user can define the velocity-dependent acceleration constraints w.r.t. the force tracking accuracy (s)he is willing to obtain. This is done in line with our claim that the identified limits should not be constant.

Remark (velocity dependent acceleration limits). An intuitive reasoning motivating this intuition comes from the scalar equation of a motor rotating about its shaft, that is

$$J_{A_i,z}\dot{\mathrm{w}} = -b|\mathrm{w}|\mathrm{w} + \tau_m, \tag{4.8}$$

[2]Additional limits could be due to limited arithmetic capabilities of the ESC micro-controller like, e.g., a maximum number of bits to perform operations and/or the lack of a floating point unit.

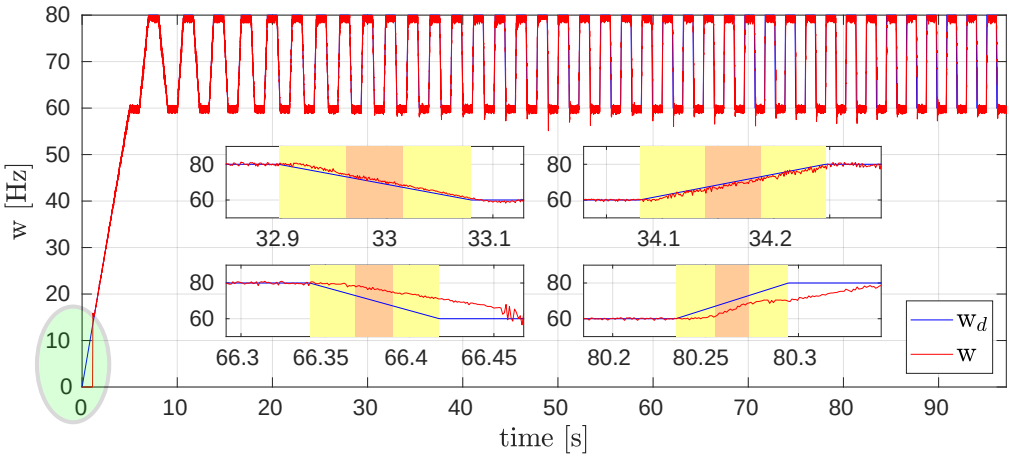


Figure 4.2 – Trajectory for the identification of the input limits at $w^* = 70\,\text{Hz}$ (that is the average spinning rotor velocities while the platform hovers) for the hexarotor setup. A series of ramps with increasing slope, which corresponds to growing acceleration commands, is sent to one actuator. The *top* and *bottom* sub-plots outline intervals where the tracking of the velocity command is good and bad, respectively. In particular, remark on the green ellipse that once the motor is activated, it has a minimum spinning velocity under which it can't physically rotate, due to the encoder-less method with which the velocity is measured. This has to be kept into account by the controller.

where $J_{A_i,z}$ is the rotor moment of inertia around the spinning axis, τ_m is the torque given by the motor, and b is a positive coefficient. As we see, $\dot{w}$ depends on the square of w. Besides, (4.8) neglects all the nonlinear effects introduced by the low-level controller, in particular the active-breaking. Therefore, we believe that the most realistic way to identify $\underline{\dot{w}}$ and $\overline{\dot{w}}$ is by supposing them to be velocity-dependent. It should be appreciated that this solution allows to model also particular MRAV configurations, like the one in [Park–2018], where the bi-directional motor accelerations highly depend on the actual rotor velocity ranges, especially in the proximity of the zero-crossing.

The profile of the desired trajectory, an example of which is given in the plot of Fig. 4.2, is a sequence of ramps (highlighted with yellow rectangles) centered at given set-points w_h^*, $h \in [1, H]$ that are chosen in order to equally span the set $[\underline{w}, \overline{w}]$. The ramp segments are designed with increasing slopes (both positive and negative) over time and separated by rest-intervals where $\dot{w}_d = 0$. In the data analysis, we select a limited part of the ramps (selectable from the user and highlighted with orange rectangles in the same plot), in order to remove possible outliers due to the discontinuity of the signal and to focus our attention in the proximity of w_h^*.
After a standard post processing of the data, mostly consisting in a low-pass filtering of the measured velocity in order to reduce high-frequency noise, by visual inspection of the force error associated with the acceleration intervals centered at each w_h^*, we determine the limits that guarantee an average force inaccuracy below a chosen threshold ϵ_f. Connecting these values using a linear interpolation, we can have an approximation of $\underline{\dot{w}}$ and $\overline{\dot{w}}$ as a function of w.

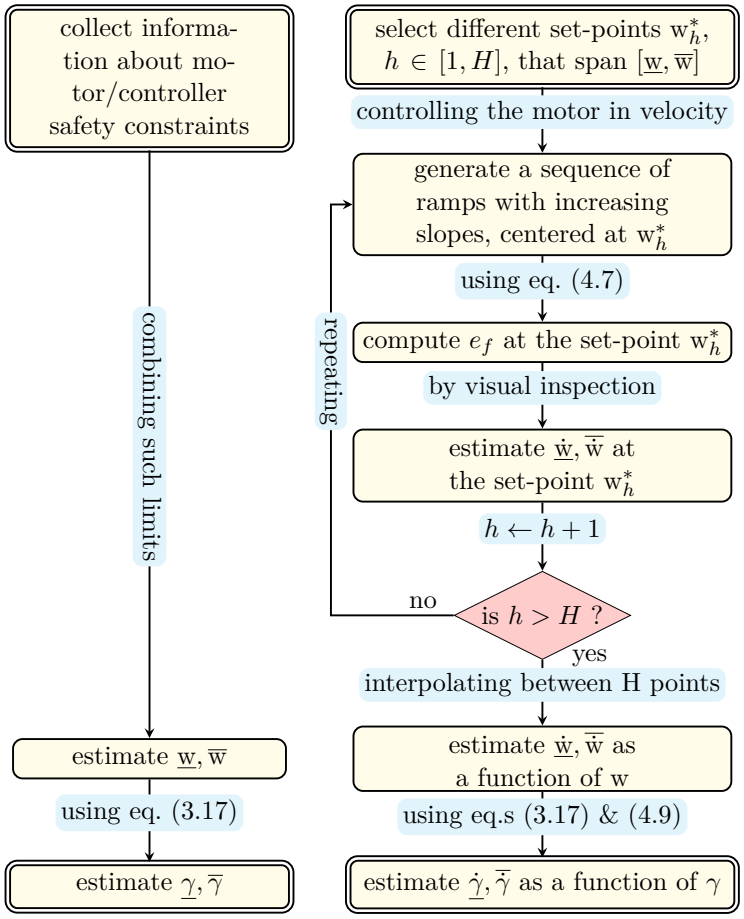


Figure 4.3 – Block diagram of the proposed procedure for the state (*left*) and input (*right*) constraints identification. Yellow blocks represent process states, cyan ones represent conditions to state change and red ones represent condition assessments.

Definition of the constraints on γ and $\dot{\gamma}$

First of all, the values $\underline{w}$ and $\overline{w}$ can be translated into the force constraints $\underline{\gamma}$ and $\overline{\gamma}$ by using the force generation model, e.g., the one given by (3.17).
Secondly, once the functions $\underline{\dot{w}}(w)$ and $\overline{\dot{w}}(w)$ are available, in order to convert them into constraints of a minimum and maximum force derivatives, we compute from (3.17) that

$$\dot{\gamma} = \frac{\partial f}{\partial w}\frac{\partial w}{\partial t} = 2c_f w\dot{w}. \tag{4.9}$$

The expression of the state dependent input constraints $\overline{\dot{\gamma}}(\gamma)$ and $\underline{\dot{\gamma}}(\gamma)$ are finally obtained from (3.17) and (4.9). We remark that, if another model for the thrust generation is used, then (4.7), and consequently (4.9) should be re-computed accordingly. The adopted overall strategy is schematically represented in the block diagram of Fig. 4.3.

State-space model for discrete-time control

As we will see in Sec. 5.2, the detailed model will be used by a receding-horizon numerical controller to produce (pseudo-) optimal inputs for the system. Combining the model for a MDT MRAV already defined in (3.15) with the constraints on the actuator forces and their derivatives, we obtain the following equations for the detailed model:

$$\begin{bmatrix} m\mathbf{I}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{J} \end{bmatrix} \begin{bmatrix} \ddot{\mathbf{p}} \\ \dot{\boldsymbol{\omega}} \end{bmatrix} = \begin{bmatrix} -mg\mathbf{e}_3 \\ -\boldsymbol{\omega} \times \mathbf{J}\boldsymbol{\omega} \end{bmatrix} + \begin{bmatrix} \mathbf{R} & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{I}_3 \end{bmatrix} \mathbf{G}\boldsymbol{\gamma} \tag{4.10}$$

$$\mathbf{u} = \dot{\boldsymbol{\gamma}} \tag{4.11}$$

$$\underline{\boldsymbol{\gamma}} \leq \boldsymbol{\gamma} \leq \overline{\boldsymbol{\gamma}} \tag{4.12}$$

$$\underline{\dot{\boldsymbol{\gamma}}} \leq \dot{\boldsymbol{\gamma}} \leq \overline{\dot{\boldsymbol{\gamma}}} \tag{4.13}$$

At this point, let us define the state vector $\mathbf{x}$ and the input vector $\mathbf{u}$ as

$$\mathbf{x} := \begin{bmatrix} \mathbf{p}^\top & \dot{\mathbf{p}}^\top & \boldsymbol{\eta}^\top & \boldsymbol{\omega}^\top & \boldsymbol{\gamma}^\top \end{bmatrix}^\top \tag{4.14}$$

$$\mathbf{u} := \dot{\boldsymbol{\gamma}} \tag{4.15}$$

with $\boldsymbol{\eta} \in \mathbb{R}^{n_\eta}$ being the vector used for concisely represent the platform orientation, which can be chosen between the ones outlined in Sec. 3.2. The expression of the state-space map $\mathbf{f}(\bullet)$ relating $\dot{\mathbf{x}}$ to $\mathbf{x}$ and $\mathbf{u}$, i.e.,

$$\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t), \mathbf{u}(t)), \tag{4.16}$$

can be obtained from (4.10)-(4.11), according to the definitions of $\mathbf{x}$ and $\mathbf{u}$ in (4.14)-(4.15) and also to the specific convention used to represent the vehicle orientation.

For digital control purposes, the continuous-time model in (4.16) can be discretized using different techniques, e.g., with a fixed step 4^{th} order explicit *Runge-Kutta integrator*, yielding the following discrete-time model:

$$\mathbf{x}_{k+1} = \boldsymbol{\phi}(\mathbf{x}_k, \mathbf{u}_k), \quad k = 0, 1, \ldots, N-1, \tag{4.17}$$

where, for simplicity of notation, $\mathbf{x}_k = \mathbf{x}(kT)$ being T the sampling time and $\mathbf{u}(t) = \mathbf{u}_k$ for $t \in [kT, (k+1)T)$.

Comparison between the two models of MRAVs

The goal of this section is to instantiate a brief comparison between the model presented in Sec. 4.2, referred to as "first model" and the one detailed in Sec. 4.1, indicated with "second model".

With reference to the input abstraction level, it is interesting to remark that the first model has a higher level of abstraction w.r.t. the standard model (3.7), since it does not describe *how* the body wrench is generated. On the other hand,

Table 4.2 – Brief comparison of the two MRAV models presented in Chapt. 4.

\	**Simplified model**	**Detailed model**
Control input	$\mathbf{f}_B^B, \boldsymbol{\tau}_B^B$	$\dot{\boldsymbol{\gamma}}$
Considered constraints	$\mathbf{f}_B^B \in \mathcal{U}_1$	$\underline{\boldsymbol{\gamma}} \leq \boldsymbol{\gamma} \leq \overline{\boldsymbol{\gamma}}$ $\underline{\dot{\boldsymbol{\gamma}}} \leq \dot{\boldsymbol{\gamma}} \leq \overline{\dot{\boldsymbol{\gamma}}}$
Parameters to be identified	$\mathcal{U}_1$	$\underline{\boldsymbol{\gamma}},\ \overline{\boldsymbol{\gamma}}$ $\underline{\dot{\boldsymbol{\gamma}}},\ \overline{\dot{\boldsymbol{\gamma}}}$
Naturally developed for	Lyapunov-based control	Numerical optimization-based control

the second model has a lower level of abstraction, since it considers the derivatives of the actuator forces as the control input to the system. A schematic summary of the characteristics of the two models is presented in Tab. 4.2.

The first model considers very generic constraints for the platform, i.e., body forces with a bounded projection on the $\{\mathbf{xy}\}_B$ plane. In particular, their projection along $\mathbf{z}_B$ is assumed unbounded. Moreover, the analysis of the body force $\mathbf{f}_B^B$ disregards the body torque $\boldsymbol{\tau}_B^B$. It is easy to be convinced that this a simplification w.r.t. the real actuation constraints. On the other hand, the second model goes deeper inside the physical description of the system, considering constraints on the actuator forces $\boldsymbol{\gamma}$ and their derivatives $\dot{\boldsymbol{\gamma}}$, which describe the actuation limits in a more realistic way. The two approaches are complementary in this sense. While the first is generic and applicable to a broad class of MDT MRAVs, the second one is specific to the particular platform under analysis. The higher level of detail comes at the price of a bigger number to parameters to be identified. In particular, a separate identification procedure must be performed for each different aerial vehicle. Finally, it is interesting to underline that the development of the model of a physical system is also mutually influenced by the design of the related controller. In particular, it is likely that a simple mathematical model might be exploited by a simple controller. In general, the higher the level of detail of a model the bigger the number of effects to be taken into account by the control strategy. This point will be better clarified in the next chapter, where we will develop a Lyapunov-based controller based on the simplified model, and a numerical optimal controller which exploits the more detailed one.

Chapter 5

Control strategies for MDT-MRAVs with actuation constraints

"Life is and will ever remain an equation incapable of solution, but it contains certain known factors."
Nikola Tesla

In the previous chapter, we have derived two possible models capable to describe, with different levels of abstraction, a Multi-Directional Thrust (MDT) Multi-Rotor Aerial Vehicle (MRAV) with actuation constraints. In a complementary fashion, the goal of this chapter is to present two suitable controllers, one for each mathematical representation of the physical system. The first proposed controller exploits the simplified model for a Laterally-Bounded input Force (LBF) MRAV to perform static feedback control. On the other hand, the second one takes advantage of the receding horizon paradigm in an Model Predictive Control (MPC) fashion to compute a (pseudo-) optimal control input for the detailed model.

Static feedback controller for LBF MRAVs

This section is devoted to the description of the control strategy that we designed in order to perform the tracking of a 6D pose trajectory for an aerial platform that can be meaningfully modeled as a LBF vehicle, as shown in Sec. 4.1. In the following, the general form of the controller will be discussed. On the other hand, for the particular implementation choices used for the experimental validation, the reader is addressed to Sec. 7.1.

Let be given a full-pose trajectory $\mathbf{q}_r(t) = (\mathbf{p}_r(t), \mathbf{R}_r(t)) : [t_0, t_f] \rightarrow \mathrm{SE}(3)$, where $\mathbf{p}_r(t) \in \mathbb{R}^3$ is the reference position trajectory and $\mathbf{R}_r(t) \in \mathrm{SO}(3)$ is the reference attitude trajectory. Inverting (4.3)-(4.4), the nominal inputs to track $\mathbf{q}_r(t)$ are obtained (in a similar way to what done in (3.18)) as: $\mathbf{u}_1^r = \mathbf{R}_r^\top (mg\mathbf{e}_3 + m\ddot{\mathbf{p}}_r)$ and $\mathbf{u}_2^r = \boldsymbol{\omega}_r \times \mathbf{J}\boldsymbol{\omega}_r + \mathbf{J}\dot{\boldsymbol{\omega}}_r$, where $\boldsymbol{\omega}_r$ is defined by $[\boldsymbol{\omega}_r]_\times = \mathbf{R}_r^\top \dot{\mathbf{R}}_r$.

Definition 5.1.1. $\mathbf{q}_r(t)$ is *feasible* if $\mathbf{u}_1^r(t) \in \mathcal{U}_1 \ \forall t \in [t_0, t_f]$.

Remark (Feasibility of the reference trajectory). The model of a LBF platform assumes no constraints are applied to $\mathbf{u}_2^r$, since the feasibility of $\mathbf{q}_r(t)$ is given only

in terms of $\mathbf{u}_1^r$.

The exact full-pose (6D) tracking for the aerial vehicle is possible only if $\mathbf{q}_r(t)$ is feasible. However, in a general realistic situation it is not granted that $\mathbf{q}_r(t)$ will be such for the particular LBF platform in use. For this reason, we propose here a controller that guarantees *at least* the tracking of $\mathbf{p}_r(t)$, even if $\mathbf{q}_r(t)$ is not feasible. Consider the following position and linear velocity errors

$$\mathbf{e}_p = \mathbf{p} - \mathbf{p}_r \tag{5.1}$$

$$\mathbf{e}_v = \dot{\mathbf{p}} - \dot{\mathbf{p}}_r, \tag{5.2}$$

and two positive definite gain matrices $\mathbf{K}_p, \mathbf{K}_v \in \mathbb{R}^{3\times3}_{>0}$. Then consider

$$\mathbf{f}_r = m\ddot{\mathbf{p}}_r + mg\mathbf{e}_3 - \mathbf{K}_p\mathbf{e}_p - \mathbf{K}_v\mathbf{e}_v, \tag{5.3}$$

which represents the reference total control force (expressed in world frame) that ideally one would like to apply to the aerial vehicle Center of Mass (CoM) if its body wrench was not subject to constraints, i.e., if $\mathcal{U}_1 = \mathbb{R}^3$. The set of orientations that allow to apply $\mathbf{f}_r$ to the CoM of the LBF aerial vehicle is defined as $\mathcal{R}(\mathbf{f}_r) = \{\mathbf{R} \in \mathrm{SO}(3) \mid \mathbf{R}^\top \mathbf{f}_r \in \mathcal{U}_1\}$. For a Uni-Directional Thrust (UDT) MRAV, the set $\mathcal{R}(\mathbf{f}_r)$ is formed by any $\mathbf{R}$ such that $\mathbf{R}\mathbf{e}_3$ and $\mathbf{f}_r$ are parallel, i.e., $\mathbf{R}\mathbf{e}_3 \times \mathbf{f}_r = 0$. For a generic LBF MRAV, the set $\mathcal{R}(\mathbf{f}_r)$ may contain also $\mathbf{R}$'s for which $\mathbf{R}\mathbf{e}_3 \times \mathbf{f}_r \neq 0$. Therefore we have the following.

Proposition 5.1.2. The set $\mathcal{R}(\mathbf{f}_r)$ is always nonempty $\forall\ \mathbf{f}_r \in \mathbb{R}^3$.

Proof. If $\mathbf{f}_r \neq \mathbf{0}$ then, by definition of $\mathcal{R}(\mathbf{f}_r)$ and $\mathcal{U}_1$,
$\mathcal{R}(\mathbf{f}_r) \supset \left\{\mathbf{R} \in \mathrm{SO}(3) \mid \mathbf{R}\mathbf{e}_3 = \frac{\mathbf{f}_r}{\|\mathbf{f}_r\|}\right\} \neq \emptyset$. If $\mathbf{f}_r = \mathbf{0}$ then $\mathcal{R}(\mathbf{f}_r) = \mathrm{SO}(3)$. □

The proposed controller exploits a cascaded structure by choosing, at each time t, a desired orientation $\mathbf{R}_d \in \mathrm{SO}(3)$ that belongs to $\mathcal{R}(\mathbf{f}_r)$ and also minimizes a given cost function w.r.t. $\mathbf{R}_r$. At this point, one can use the rotational dynamics, which is assumed (by definition) unconstrained for a LBF platform, to track $\mathbf{R}_d$ and, in turn, track the reference position $\mathbf{p}_r$. If $\mathbf{q}_r$ is feasible then $\mathbf{R}_d$ will exponentially converge to $\mathbf{R}_r$. Otherwise, only the best feasible orientation will be obtained. Therefore, the controller implicitly prioritizes the position trajectory tracking over the orientation one.
Define $\overline{\mathcal{R}}(\mathbf{f}_r, \mathbf{R}_r) \subset \mathcal{R}(\mathbf{f}_r)$ as the set of rotation matrices that solve the minimization problem

$$\min_{\mathbf{R}' \in \mathcal{R}(\mathbf{f}_r)} c_R(\mathbf{R}_r, \mathbf{R}'), \tag{5.4}$$

where $c_R : \mathrm{SO}(3) \times \mathrm{SO}(3) \to \mathbb{R}_{\geq 0}$ is an arbitrarily chosen cost function that represents the degree of similarity between $\mathbf{R}_r$ and $\mathbf{R}'$ one is interested in. The elements in $\overline{\mathcal{R}}(\mathbf{f}_r, \mathbf{R}_r)$ represent orientations of the LBF platform that allow to apply $\mathbf{f}_r$ and minimize the function c_R w.r.t. $\mathbf{R}_r$. Consider that, at each time t a desired

orientation $\mathbf{R}_d \in \overline{\mathcal{R}}(\mathbf{f}_r, \mathbf{R}_r)$ is chosen. Furthermore, whenever $\mathbf{R}_r \in \overline{\mathcal{R}}(\mathbf{f}_r, \mathbf{R}_r)$ then $\mathbf{R}_d$ must be chosen equal to $\mathbf{R}_r$. Then define the following rotation and angular velocity errors

$$\mathbf{e}_R = \frac{1}{2}(\mathbf{R}_d^\top \mathbf{R} - \mathbf{R}^\top \mathbf{R}_d)^\vee \tag{5.5}$$

$$\mathbf{e}_\omega = \boldsymbol{\omega} - \mathbf{R}^\top \mathbf{R}_d \boldsymbol{\omega}_d, \tag{5.6}$$

where $\boldsymbol{\omega}_d$ is the angular velocity associated to $\mathbf{R}_d$.
Consider then the following control law:

$$\mathbf{u}_1 = \mathrm{sat}_{\mathcal{U}_{xy}}((\mathbf{f}_r^\top \mathbf{R}\mathbf{e}_1)\mathbf{e}_1 + (\mathbf{f}_r^\top \mathbf{R}\mathbf{e}_2)\mathbf{e}_2) + (\mathbf{f}_r^\top \mathbf{R}\mathbf{e}_3)\mathbf{e}_3 \tag{5.7}$$

$$\mathbf{u}_2 = \boldsymbol{\omega} \times \mathbf{J}\boldsymbol{\omega} - \mathbf{K}_R \mathbf{e}_R - \mathbf{K}_\omega \mathbf{e}_\omega - \mathbf{J}\big([\boldsymbol{\omega}]_\times \mathbf{R}^\top \mathbf{R}_d \boldsymbol{\omega}_d - \mathbf{R}^\top \mathbf{R}_d \dot{\boldsymbol{\omega}}_d\big), \tag{5.8}$$

where $\mathrm{sat}_{\mathcal{U}_{xy}}(\bullet)$ is a vector in $\mathcal{U}_{xy}$, with the same direction of $\bullet$, that minimizes the distance from $\bullet$, while $\mathbf{K}_R, \mathbf{K}_\omega \in \mathbb{R}^{3\times 3}_{>0}$ are positive-definite gain matrix.

Remark (comparison with other works in the state of the art). Notice that only in the very special case in which $\mathcal{U}_{xy} = \{\mathbf{0}\}$ the control in (5.7)-(5.8) always resembles the one in [Lee–2010] while its form is in general very different from it. Another difference with [Lee–2010] is that the computation of $\mathbf{R}_d$ in [Lee–2010] is performed in relation to the position reference using the differential flatness property [Mistler–2001; Mellinger–2011; Faessler–2018], while here it is computed ensuring input feasibility and minimizing c_R w.r.t. $\mathbf{R}_r$.
Remark also that the idea of proposed approach to prioritize the control of the position over the one of the orientation has been fostered also by [Hua–2015a]. In this work, it is assumed that the thrust-tilting angle between the body frame force and the $\mathbf{z}_B$ axis can not exceed a certain threshold. This is equivalent to assume that the set of feasible body forces for a specific body torque has a particular shape (which is an inverted pseudo-cone, in that case), as also done in [Invernizzi–2018]. In the presentation of our control strategy, we keep instead the shape very general, leaving the details of the particular choice for the explicit definition of $\mathcal{U}_1$ in (4.2) and the solution of the minimization problem (5.4) in the validation of Sec. 7.1. It should be noticed that the desired feasible orientation in [Hua–2015a] is computed in relation to a reference angular velocity for the force in body frame, which describes the rotational evolution of the body force vector in $\mathcal{F}_B$, while here $\mathbf{R}_d$ is found as the solution of a minimization problem. Therefore, despite sharing the common cascade paradigm and the prioritized control of the position over the orientation, the two control strategies differs, in general, in the accomplishment of the task with lower priority, i.e., the control of the body frame orientation.

In order to prove the convergence properties of the proposed controller let us consider as error function between two rotation matrices $\mathbf{R}_1$ and $\mathbf{R}_2$ to be the following one:

$$d(\mathbf{R}_1, \mathbf{R}_2) = \frac{1}{2}\mathrm{tr}\left(\mathbf{I} - \mathbf{R}_2^\top \mathbf{R}_1\right). \tag{5.9}$$

Theorem 5.1.3. Assume that $\mathbf{R}_d(t) \in \mathcal{R}(\mathbf{f}_r(t))$ for any t and that $\boldsymbol{\omega}_d(t)$ and $\dot{\boldsymbol{\omega}}_d(t)$ are well defined for any t. Consider the control $\mathbf{u}_1$ and $\mathbf{u}_2$ defined in (5.7) and in (5.8). Assume that the initial condition satisfies

$$d\left(\mathbf{R}(0), \mathbf{R}_d(0)\right) < 2 \tag{5.10}$$

$$\|\mathbf{e}_\omega(0)\|^2 < \frac{2}{\lambda_{\min}(\mathbf{J})} k_R \left(1 - d\left(\mathbf{R}(0), \mathbf{R}_d(0)\right)\right). \tag{5.11}$$

Then, the zero equilibrium of the tracking errors $\mathbf{e}_R$, $\mathbf{e}_\omega$, $\mathbf{e}_p$ and $\mathbf{e}_v$ is exponentially stable. The region of attraction is characterized by (5.10) and (5.11).

Proof. The proof is divided into two parts. We first show that, if the $\mathbf{R}(0)$ and $\mathbf{e}_\omega(0)$ satisfy, respectively, (5.10) and (5.11), then $\mathbf{R}(t)$ converges exponentially to $\mathbf{R}_d(t)$, in the sense that the function $d\left(\mathbf{R}(t), \mathbf{R}_d(t)\right)$ goes exponentially to zero. Secondly, we characterize the translational error dynamics and, based on the fact that $\mathbf{R}(t)$ converges exponentially to $\mathbf{R}_d(t)$, we show that also $\mathbf{e}_p$ and $\mathbf{e}_v$ goes exponentially to zero.
As far as the first part is concerned, we start by noticing that the time derivative of $\mathbf{e}_\omega$ is

$$\mathbf{J}\dot{\mathbf{e}}_\omega = \mathbf{J}\dot{\boldsymbol{\omega}} + \mathbf{J}\left([\boldsymbol{\omega}]_\times \mathbf{R}^\top \mathbf{R}_d \boldsymbol{\omega}_d - \mathbf{R}^\top \mathbf{R}_d \dot{\boldsymbol{\omega}}_d\right). \tag{5.12}$$

Plugging the rotational dynamics (4.4) into (5.12) and substituting $\mathbf{u}_2$ from (5.8), we get

$$\mathbf{J}\dot{\mathbf{e}}_\omega = -k_R \mathbf{e}_R - k_\omega \mathbf{e}_\omega. \tag{5.13}$$

In [Lee–2010], it is shown, by exhibiting a suitable Lyapunov function, that, under conditions in (5.10) and in (5.11), the zero equilibrium of the attitude tracking error $\mathbf{e}_R$, $\mathbf{e}_\omega$ is exponentially stable and that there exist two positive constants α, β such that

$$d\left(\mathbf{R}(t), \mathbf{R}_d(t)\right) < \alpha e^{-\beta t} d\left(\mathbf{R}(0), \mathbf{R}_d(0)\right). \tag{5.14}$$

We determine now the error dynamics of the translational dynamics. Substituting $\mathbf{u}_1$ from (5.7) in the translational dynamics (4.3), we obtain

$$m\ddot{\mathbf{p}} = -mg\mathbf{e}_3 + \mathbf{f}_r + \mathbf{R}\mathbf{u}_1 - \mathbf{f}_r = m\ddot{\mathbf{p}}_r - \mathbf{K}_p\mathbf{e}_p - \mathbf{K}_v\mathbf{e}_v + \mathbf{s}, \tag{5.15}$$

where $\mathbf{s} = \mathbf{R}\mathbf{u}_1 - \mathbf{f}_r$. It easily follows that

$$m\dot{\mathbf{e}}_v = -\mathbf{K}_p\mathbf{e}_p - \mathbf{K}_v\mathbf{e}_v + \mathbf{s}. \tag{5.16}$$

Consider (5.14) and observe that, since $\mathbf{R}_d \in \mathcal{R}(\mathbf{f}_r)$ for any t, we have that there exist two positive constants C, ρ such that

$$\|\mathbf{s}(t)\| \leq Ce^{-\rho t}\|\mathbf{s}(0)\|. \tag{5.17}$$

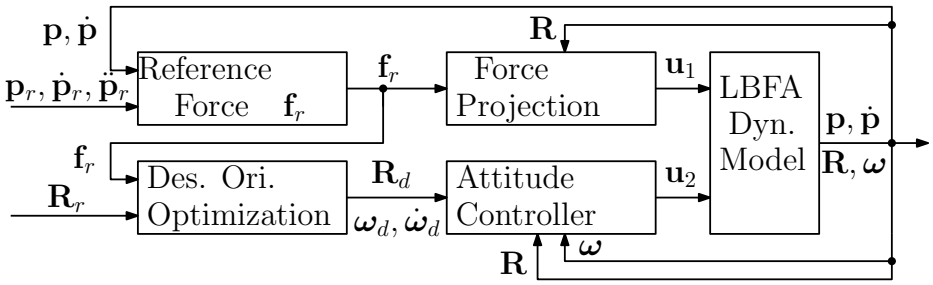


Figure 5.1 – Block diagram of the proposed static feedback controller.

Let $\tilde{\mathbf{e}} = [\mathbf{e}_v \;\; \mathbf{e}_p]^\top$. Consequently, (5.16) can be written in vector form as

$$\dot{\tilde{\mathbf{e}}} = \mathbf{A}\tilde{\mathbf{e}} + \mathbf{B}\mathbf{s}, \tag{5.18}$$

where

$$\mathbf{A} = \frac{1}{m}\begin{bmatrix} -\mathbf{K}_v & -\mathbf{K}_p \\ \mathbf{I} & \mathbf{0} \end{bmatrix}, \qquad \mathbf{B} = \frac{1}{m}\begin{bmatrix} \mathbf{I} \\ \mathbf{0} \end{bmatrix}. \tag{5.19}$$

Since $\mathbf{K}_v, \mathbf{K}_p$ are both positive definite matrices, we have that $\mathbf{A}$ is a Hurwitz matrix. Observe that (5.18) is the cascade of a linear stable system and an exponential stable signal. Then, the statement of the Theorem follows from a technical refinement of Lemma 4.7 in [Khalil–2001], i.e., the cascade of an asymptotically stable linear system and of an exponentially stable system is exponentially stable. □

A block diagram that shows the main subsystems of the proposed control architecture is provided in Fig. 5.1. Theorem 5.1.3 ensures, under mild conditions, the exponential stability of $\mathbf{e}_p$, $\mathbf{e}_v$, $\mathbf{e}_R$, and $\mathbf{e}_\omega$.

Notice that this results holds regardless of the feasibility of $\mathbf{q}_r$. If $\mathbf{q}_r$ is also feasible then exponential tracking of $\mathbf{q}_r$ by $\mathbf{q}$ is also guaranteed. In order to formally state this fact, let us first define the following rotational errors:

$$\mathbf{e}_{R_r} = \frac{1}{2}(\mathbf{R}_r^\top \mathbf{R}_d - \mathbf{R}_d^\top \mathbf{R}_r)^\vee \tag{5.20}$$

$$\mathbf{e}_{\omega_r} = \boldsymbol{\omega}_d - \mathbf{R}_d \mathbf{R}_r^\top \boldsymbol{\omega}_r. \tag{5.21}$$

In next result we characterize the convergence of the above rotational errors to zero provided that the reference trajectory $\mathbf{q}_r(t)$ is *feasible* and satisfies the additional property that $\mathbf{u}_1^r$ is *sufficiently inside* $\mathcal{U}_1$, meaning that there exists a time instant $\bar{t}$ and a positive number ϵ such that the distance of $\mathbf{u}_1^r$ from the boundary of $\mathcal{U}_1$ is greater than $\epsilon > 0$ for all $t > \bar{t}$, i.e.,

$$\operatorname{dist}(\mathbf{u}_1^r(t), \partial\mathcal{U}_1) > \epsilon, \qquad \forall\, t > \bar{t}. \tag{5.22}$$

Corollary 5.1.4. Assume $\mathbf{q}_r(t)$ is a feasible trajectory and that it satisfies the additional property in (5.22). Assume that $\mathbf{R}_d(t) \in \mathcal{R}(\mathbf{f}_r(t))$ for any t and that $\boldsymbol{\omega}_d(t)$ and $\dot{\boldsymbol{\omega}}_d(t)$ are well defined for any t. Consider the control $\mathbf{u}_1$ and $\mathbf{u}_2$ defined at (5.7) and (5.8). Assume that the initial condition satisfies (5.10) and (5.11). Then the zero equilibrium of the tracking errors $\mathbf{e}_R$, $\mathbf{e}_\omega$, $\mathbf{e}_p$ and $\mathbf{e}_v$ is exponentially stable and there exists a time instant $\bar{t} \geq t_0$ such that $\mathbf{e}_{R_r}(t) = \mathbf{e}_{\omega_r}(t) = 0$ for all $t > \bar{t}$. The region of attraction is characterized by (5.10) and (5.11).

Proof. From Theorem 5.1.3 we can write that

$$\mathbf{f}_r = m\ddot{\mathbf{p}}_r + mg\mathbf{e}_3 + \boldsymbol{\xi} \tag{5.23}$$

where

$$\|\boldsymbol{\xi}(t)\| \leq Le^{-\lambda t}\|\boldsymbol{\xi}(0)\| \tag{5.24}$$

for some positive constants L and λ. This implies that the vector $\mathbf{f}_r - (m\ddot{\mathbf{p}}_r + mg\mathbf{e}_3)$ and, in turn, also the vector $\mathbf{R}_r^\top \mathbf{f}_r - \mathbf{R}_r^\top (m\ddot{\mathbf{p}}_r + mg\mathbf{e}_3)$, tend exponentially to zero.

Hence, since $\mathbf{q}_r(t)$ is feasible and satisfies (5.22), it follows, from continuity arguments, that there exists t' such that $\mathbf{R}_r^\top \mathbf{f}_r \in \mathcal{U}_1$ for all $t \geq t'$. Therefore $\mathbf{R}_d(t) = \mathbf{R}_r(t)$ for all $t > t'$. □

Remark. The proposed controller (in particular the attitude controller (5.8)) relies on the availability of $\boldsymbol{\omega}_d$, and $\dot{\boldsymbol{\omega}}_d$. These quantities depend in turn on $\mathbf{R}_d$, which is the output of an optimization algorithm executed at each control step. In order for $\boldsymbol{\omega}_d$ and $\dot{\boldsymbol{\omega}}_d$ to be well defined and available, the optimization must ensure a sufficient smoothness of $\mathbf{R}_d$. This could be enforced by adding, e.g., a regularization term in the cost function c_R. If in the real case at hand this is not possible (or not implementable), then at each time instant in which $\mathbf{R}_d$ is not smooth the attitude controller will undergo a new transient phase. In practice, as we will show in Chapt. 7, we have experimentally ascertain that the presence of a few isolated non-smooth instants does not constitute at all a real problem for the stability of the implementation and that regularization is actually not needed for practical stabilization.

Remark (Extension of stability proof). It should be mentioned that the asymptotic stability discussed and proved in Theorem 5.1.3 and in Corollary 5.1.4 can be ensured almost globally under the evoked assumptions. The simple yet interesting formalization of this extension is left for future work.

NMPC controller for MRAVs with actuator constraints

In the previous section, we have presented the design of a static feedback pose controller capable to comply with the wrench limitations defined by the abstract and general class of LBF vehicles. In a complementary way, the goal of this section is to exploit the more detailed model description outlined in Sec. 4.2 in order to implement a controller that can take into account the more precise and representative limitations of the MRAV actuators. More in detail, we address this problem by making use of numerical optimal control techniques based on the receding-horizon paradigm. This choice is motivated by the will to devise a nonlinear model predictive controller able to optimally compute input commands that can better exploit the actuation capabilities of the platform.

As already discussed in Chapt. 2, the proposed framework should simultaneously address the problem of local reference trajectory generation and that of stabilizing the vehicle dynamics. Specifically, we aim at tracking a reference trajectory denoted $(\mathbf{p}_r(t), \boldsymbol{\eta}_r(t))$ given by a generic global planner. We assume $(\mathbf{p}_r(t), \boldsymbol{\eta}_r(t))$ to be twice continuously differentiable. Whenever the trajectory is not compliant w.r.t. the imposed input and state limitations, then a new feasible one should be computed and tracked. This new trajectory should minimize a user-defined cost function computed alongside the prediction horizon. In this way, the controller can exploit the prediction of the model state evolution to compute a feasible and effective input action. In order to guarantee smoothness properties of the generated trajectory, we force our algorithm to be able to drive also the derivatives of the state variables toward the corresponding ones of the reference trajectory. Therefore, we introduce the following enlarged reference signal

$$\mathbf{y}_r(t) = \left[\mathbf{p}_r^\top(t)\ \dot{\mathbf{p}}_r^\top(t)\ \ddot{\mathbf{p}}_r^\top(t)\ \boldsymbol{\eta}_r^\top(t)\ \boldsymbol{\omega}_r^\top(t)\ \dot{\boldsymbol{\omega}}_r^\top(t)\right]^\top \tag{5.25}$$

and, accordingly, we define the output map as

$$\mathbf{y}(t) = \mathbf{h}\left(\mathbf{x}(t), \mathbf{u}(t)\right) = \begin{bmatrix} \mathbf{p}(t) \\ \dot{\mathbf{p}}(t) \\ \ddot{\mathbf{p}}\left(\mathbf{x}(t), \mathbf{u}(t)\right) \\ \boldsymbol{\eta}(t) \\ \boldsymbol{\omega}(t) \\ \dot{\boldsymbol{\omega}}\left(\mathbf{x}(t), \mathbf{u}(t)\right) \end{bmatrix}. \tag{5.26}$$

For clarity observe that $\mathbf{p}(t), \dot{\mathbf{p}}(t), \boldsymbol{\eta}(t), \boldsymbol{\omega}(t)$ are sub-vectors of the state, while $\ddot{\mathbf{p}}, \dot{\boldsymbol{\omega}}$ are functions of $\mathbf{x}(t), \mathbf{u}(t)$, and, in particular, sub-components of the map $\mathbf{f}$ in (4.16). Furthermore, we define $\mathbf{y}_{r,k}$, $\mathbf{y}_k$ as the discretized version of $\mathbf{y}_r(t)$ and $\mathbf{y}(t)$, respectively, i.e., $\mathbf{y}_{r,k} = \mathbf{y}_r(kT)$, $\mathbf{y}_k = \mathbf{y}(kT)$.

At each discretized control time kT, the Nonlinear Model Predictive Control (NMPC) strategy exploits the model dynamics and the most recent measure $\mathbf{y}_k$

to solve the Optimal Control Problem (OCP) of determining the control input sequence that minimizes a given cost function, which depends on the reference signal $\mathbf{y}_r$ and on the output predictions $\hat{\mathbf{y}}$, along the horizon and also satisfy the model constraints. The OCP to be solved at time kT, given the current state $\mathbf{x}_k$, is formulated as:

$$\min_{\hat{\mathbf{x}}_0,\ldots,\hat{\mathbf{x}}_N,\hat{\mathbf{u}}_0,\ldots,\hat{\mathbf{u}}_{N-1}} \sum_{h=0}^{N-1} \left\{ \|\hat{\mathbf{y}}_h - \mathbf{y}_{r,k+h}\|^2_{\mathbf{Q}_h} + \|\hat{\mathbf{u}}_h\|^2_{\mathbf{R}_h} \right\} + \\ + \|\hat{\mathbf{y}}_N - \mathbf{y}_{r,k+N}\|^2_{\mathbf{Q}_N} \tag{5.27}$$

$$\text{s.t.} \quad \hat{\mathbf{x}}_0 = \mathbf{x}_k \tag{5.28}$$

$$\hat{\mathbf{x}}_{h+1} = \boldsymbol{\phi}(\hat{\mathbf{x}}_h, \hat{\mathbf{u}}_h), \quad h{=}0,1,\ldots,N{-}1, \tag{5.29}$$

$$\hat{\mathbf{y}}_h = \mathbf{h}(\hat{\mathbf{x}}_h, \hat{\mathbf{u}}_h), \quad h{=}0,1,\ldots,N, \tag{5.30}$$

$$\underline{\boldsymbol{\gamma}} \leq \mathbf{M}\hat{\mathbf{x}}_h \leq \overline{\boldsymbol{\gamma}}, \quad h{=}0,1,\ldots,N, \tag{5.31}$$

$$\underline{\dot{\boldsymbol{\gamma}}}_{k+h} \leq \hat{\mathbf{u}}_h \leq \overline{\dot{\boldsymbol{\gamma}}}_{k+h}, \quad h{=}0,1,\ldots,N{-}1, \tag{5.32}$$

where $\mathbf{Q}_h$, $\mathbf{R}_h$ are semidefinite positive matrices and matrix $\mathbf{M}$ is defined in order to select only the last n elements of the state $\mathbf{x}$, that is,

$$\mathbf{M} = \begin{bmatrix} \mathbf{0}_{n\times(9+n_\eta)} & \mathbf{I}_n \end{bmatrix}. \tag{5.33}$$

The bounds $\overline{\boldsymbol{\gamma}}, \underline{\boldsymbol{\gamma}}$, depend on the quantities $\overline{f}$, $\underline{f}$ defined in Sec. 4.2.2 and, compactly, they are defined as

$$\overline{\boldsymbol{\gamma}} = \mathbf{1}_{6\times1} \otimes \overline{\gamma} \tag{5.34}$$

$$\underline{\boldsymbol{\gamma}} = \mathbf{1}_{6\times1} \otimes \underline{\gamma}. \tag{5.35}$$

According to the discussion in Section Sec. 4.2.2, the bounds $\underline{\dot{\boldsymbol{\gamma}}}_{k+h}, \overline{\dot{\boldsymbol{\gamma}}}_{k+h}$, $h = 0, 1, \ldots, N-1$, depend on the time-varying and state dependent quantities $\overline{\dot{\gamma}}(\gamma)$, $\underline{\dot{\gamma}}(\gamma)$ and, precisely, they should be defined as

$$\overline{\dot{\boldsymbol{\gamma}}}_{k+h} = \left[\overline{\dot{\gamma}}(\gamma_{1,k+h}), \ldots, \overline{\dot{\gamma}}(\gamma_{n,k+h})\right]^\top \tag{5.36}$$

$$\underline{\dot{\boldsymbol{\gamma}}}_{k+h} = \left[\underline{\dot{\gamma}}(\gamma_{1,k+h}), \ldots, \underline{\dot{\gamma}}(\gamma_{n,k+h})\right]^\top. \tag{5.37}$$

Observe that constraints in (5.36), with $\overline{\dot{\boldsymbol{\gamma}}}_{k+h}$ and $\underline{\dot{\boldsymbol{\gamma}}}_{k+h}$ defined as above are highly non-linear. In order to retain linearity of these constraints in the OCP, we consider the following alternative definition for $\overline{\dot{\boldsymbol{\gamma}}}_{k+h}$ and $\underline{\dot{\boldsymbol{\gamma}}}_{k+h}$:

$$\overline{\dot{\boldsymbol{\gamma}}}_{k+h} = \left[\overline{\dot{\gamma}}(\tilde{\gamma}_{1,k+h}), \ldots, \overline{\dot{\gamma}}(\tilde{\gamma}_{n,k+h})\right]^\top \tag{5.38}$$

$$\underline{\dot{\boldsymbol{\gamma}}}_{k+h} = \left[\underline{\dot{\gamma}}(\tilde{\gamma}_{1,k+h}), \ldots, \underline{\dot{\gamma}}(\tilde{\gamma}_{n,k+h})\right]^\top. \tag{5.39}$$

where $\tilde{\gamma}_{i,k+h}$ are independent of the decision variables of the OCP at time $t = kT$.

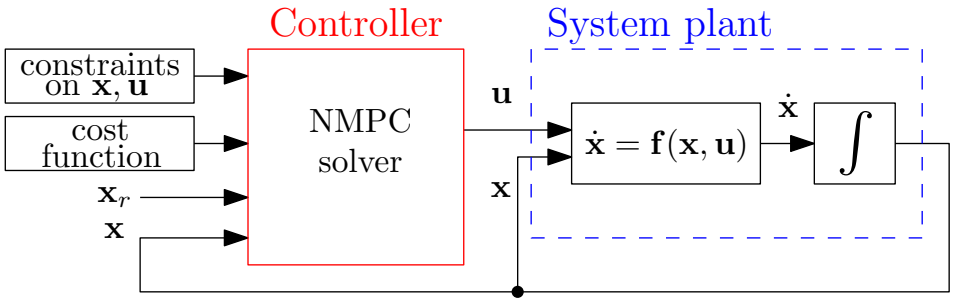


Figure 5.2 – Block diagram of the proposed receding-horizon controller.

Different choices can be taken, for example one possible choice is

$$\tilde{\gamma}_{i,k+h} = \gamma_{i,k}, \quad h = 0, \ldots, N-1. \tag{5.40}$$

Alternatively, $\tilde{\gamma}_{i,k+h}$ can be selected in a time-varying fashion based on the solution of the OCP obtained at instant $t = (k-1)T$. The particular definition of the constraints chosen for the validation is discussed in Sec. 7.2.

The solution to the OCP, at a given time step k consists of the optimal values $\hat{\mathbf{x}}_{0|k}$, …, $\hat{\mathbf{x}}_{N|k}$, $\hat{\mathbf{u}}_{0|k}$, …, $\hat{\mathbf{u}}_{N-1|k}$. According to the receding horizon principle [Mayne–2000], the input value $\mathbf{u}_k = \hat{\mathbf{u}}_{0|k}$ is applied, and the procedure is repeated again at the subsequent time step $k+1$. A block diagram of the controller is presented in Fig. 5.2.

Remark (controller stability). Regarding the stability-related properties of our control scheme, first of all note that the problem addressed here consists of tracking a trajectory generated, possibly without any regard of the vehicle model, by a generic global planner; under this general assumption, stability (in a strict sense) of the reference trajectory cannot be guaranteed, since the possibility to track the given set-point is allowed only provided that the latter is generated compatibly with the system dynamics and the actuator constraints. Under the assumption that the trajectory is consistent with the vehicle dynamics (e.g., as in [Alessandretti–2013]), stability guarantees could be provided by selecting a sufficiently large prediction horizon length N relying on [Grüne–2010; Alamir–1995]. Such approach has been applied for path following in a robotic scenario, e.g., in [Mehrez–2017].

Remark (existence of the solution). Another important feature of MPC-based algorithms is recursive feasibility, i.e., the guarantee that the OCP always admits a solution. In our practical implementation we have adopted the widespread solution of guaranteeing it by enforcing the (slightly tightened) constraints in a soft way using slack variables. All the other implementation details are thoroughly discussed in Sec. 7.2.

Table 5.1 – Brief comparison of the two MRAV controllers presented in Chapt. 5.

\	**Static feedback controller**	**NMPC controller**
Comply with $\underline{\gamma}$, $\overline{\gamma}$	yes (conservatively)	yes
Comply with $\underline{\dot{\gamma}}$, $\overline{\dot{\gamma}}$	no	yes
Stability proof	yes	no
Exploit model predictions	no	yes

Comparison between the two controllers for MRAVs

This section is devoted to a brief comparison between the two control strategies proposed in this chapter. A schematic summary of the characteristics of the two models is presented in Tab. 5.1.

The first one, presented in Sec. 5.1, exploits the simplified model of LBF MRAVs to compute a static feedback input which ensures the tracking of the position reference (providing some smoothness property) and the tracking of the "closest" orientation profile to the reference one that respects the LBF constraint. In order to guarantee the system stability, the set $\mathcal{U}_1$ defined in (4.2) must be chosen in a conservative way w.r.t. the complete set of feasible body force, as we will see in Sec. 7.1. This means that the static feedback controller is proved to stabilize the system, but with reduced performance w.r.t. the real actuation capability of the platform. Furthermore, it can not guarantee the compliance with constraints on the actuator force derivatives. Finally, the input at a certain time is computed only in relation to the state error at the same time, given the reactive nature of the first controller. On the other hand, the NMPC controller is not theoretically proven, due to the difficulties associated with the definition of the stability proof in the general case. Nevertheless, in a consistent set of real experiments, presented in Sec. 7.2, it stabilizes the MRAV and ensures high performance in the trajectory tracking, also in the case of rapidly-changing (even discontinuous) reference profiles. Furthermore, the NMPC can guarantee the compliance with constraints in the actuator forces and their derivatives. Ultimately, it can exploit the model to compute predictions of the system state and regulate the input accordingly.

After the presentation of the design of different MRAV prototypes, to which Chapt. 6 is devoted, the experimental and numerical validation of the two control approaches is addressed in Chapt. 7.

Chapter 6

Design and Prototyping of different MDT-MRAVs

"Invention, it must be humbly admitted, does not consist in creating out of void, but out of chaos."
Mary Shelley

The goal of this chapter is to emphasize the importance of the design phase for the conception of Multi-Directional Thrust (MDT) Multi-Rotor Aerial Vehicle (MRAV) prototypes with the Fully-Ranked Allocation Matrix (FRA) capability, that is the full-rankness of the allocation matrix. In the first section, we will present design considerations related to three hexarotor platforms, while in the second one the study of an octorotor configuration with omni-directional thrust will be addressed.

Multi-Directional Thrust hexarotors

In this section, we first present the study of the allocation matrix for a particular hexarotor configuration with synchronized dual-tilting propellers. This will underline the necessity of a formal analysis for the design on MDT MRAV, in order to prevent the rising of unexpected singularities in the allocation matrix, which would dramatically affect the wrench generation capability of the platform. Secondly, we briefly show the design of three different hexarotor prototypes that we conceived and built in LAAS–CNRS.

Analysis of the allocation matrix

Let us consider here a *star-shaped*[1] hexarotor with arbitrarily oriented actuators.

$$\mathbf{p}_{A_i}^B = \ell\, \mathbf{R}_z((i-1)\,\pi/3)\, \boldsymbol{e}_1, \quad i = 1, \ldots, 6. \tag{6.1}$$

We assume that actuator i can be tilted of an angle α about the $\mathbf{x}_{A_i}$-axis and of an angle β about the $\mathbf{y}_{A_i}$-axis (see Fig. 6.1). In particular, we assume the following

[1] The configuration with all the actuators laying on the same plane and equally spaced, adopted for all the presented platforms, has been chosen for simplicity of mechanic implementation.

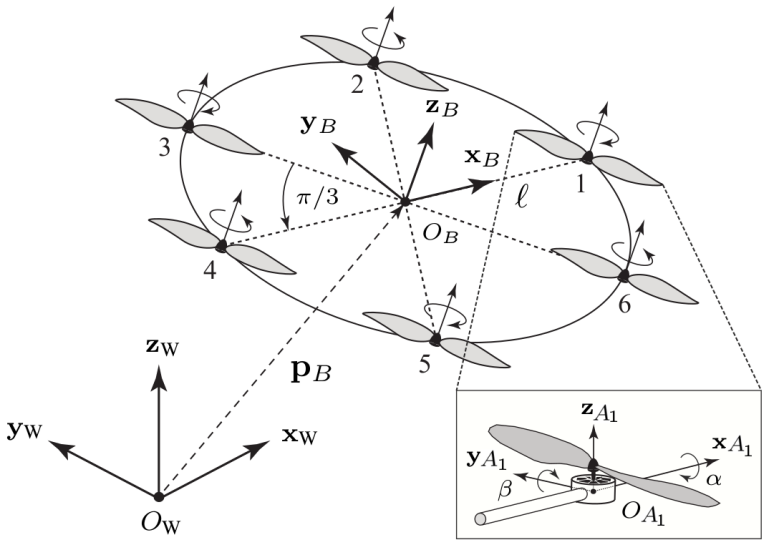


Figure 6.1 – Schematic representation of the hexarotor with dual-tilting propellers. The odd-numbered propellers spin clockwise and the even-numbered propellers spin counter-clockwise. The inset shows the tilting angles α and β of propeller 1.

orientation pattern for the actuators:

$$\mathbf{R}_{A_i}^{B}(\alpha,\beta) = \mathbf{R}_z((i-1)\,\pi/3)\,\mathbf{R}_x((-1)^{i-1}\alpha)\,\mathbf{R}_y(\beta) \quad i = 1,\dots,6. \tag{6.2}$$

With this configuration, we model the fact that the actuators with adjacent indices tilt (inward/outward) in *opposite directions* of an angle α about the $\mathbf{x}_{S_i}$-axis, while the rotation of an angle β about the $\mathbf{y}_{S_i}$-axis is the same. Moreover, we assume that odd actuators are endowed with CW propellers, while CCW propellers are mounted on even rotors (the importance of this fact will be clarified in the following), i.e.,

$$c_i = (-1)^{i-1} \quad i = 1,\dots,6, \tag{6.3}$$

where parameter c_i correspond to $\bar{k}_i$ in (3.10). Note that the alternating sign in (6.3) models the fact that propellers with adjacent indices are designed to spin in *opposite directions* and thus generate opposite drags.

Let us now analyze the allocation matrix $\mathbf{G}$ of this platform in order to find all the values of α and β for which it becomes rank-deficient. Using (3.14), we can obtain explicit expressions for $\mathbf{G}(\alpha,\beta)$ reported in (6.6), where

$$p_y = \ell\cos\alpha + c_f^{\tau}\sin\alpha, \quad p_z = \ell\sin\alpha - c_f^{\tau}\cos\alpha, \tag{6.4}$$

and $s\beta$ and $c\beta$ are shorthands for $\sin\beta$ and $\cos\beta$, respectively. Note that p_y is the scalar projection of vector $[\ell,\, c_f^{\tau}]^{\top}$ onto $[\cos\alpha,\, \sin\alpha]^{\top}$ (i.e. onto the $\mathbf{y}_{A_i}$-axis), and that p_z is the scalar projection of $[\ell,\, c_f^{\tau}]^{\top}$ onto the orthogonal $[\sin\alpha,\, -\cos\alpha]^{\top}$ (i.e.

$$\mathbf{G}(\alpha,\beta) = \begin{bmatrix} s\beta & \frac{1}{2}s\beta - \frac{\sqrt{3}}{2}c\beta s\alpha & -\frac{1}{2}s\beta + \frac{\sqrt{3}}{2}c\beta s\alpha & -s\beta & -\frac{1}{2}s\beta - \frac{\sqrt{3}}{2}c\beta s\alpha & \frac{1}{2}s\beta + \frac{\sqrt{3}}{2}c\beta s\alpha \\ -c\beta s\alpha & \frac{\sqrt{3}}{2}s\beta + \frac{1}{2}c\beta s\alpha & \frac{\sqrt{3}}{2}s\beta + \frac{1}{2}c\beta s\alpha & -c\beta s\alpha & -\frac{\sqrt{3}}{2}s\beta + \frac{1}{2}c\beta s\alpha & -\frac{\sqrt{3}}{2}s\beta + \frac{1}{2}c\beta s\alpha \\ c\beta c\alpha & c\beta c\alpha & c\beta c\alpha & c\beta c\alpha & c\beta c\alpha & c\beta c\alpha \\ c_f^\tau s\beta & -\frac{1}{2}c_f^\tau s\beta + \frac{\sqrt{3}}{2}p_y c\beta & -\frac{1}{2}c_f^\tau s\beta + \frac{\sqrt{3}}{2}p_y c\beta & c_f^\tau s\beta & -\frac{1}{2}c_f^\tau s\beta - \frac{\sqrt{3}}{2}p_y c\beta & -\frac{1}{2}c_f^\tau s\beta - \frac{\sqrt{3}}{2}p_y c\beta \\ -p_y c\beta & -\frac{\sqrt{3}}{2}c_f^\tau s\beta - \frac{1}{2}p_y c\beta & \frac{\sqrt{3}}{2}c_f^\tau s\beta + \frac{1}{2}p_y c\beta & p_y c\beta & -\frac{\sqrt{3}}{2}c_f^\tau s\beta + \frac{1}{2}p_y c\beta & \frac{\sqrt{3}}{2}c_f^\tau s\beta - \frac{1}{2}p_y c\beta \\ -p_z c\beta & p_z c\beta & -p_z c\beta & p_z c\beta & -p_z c\beta & p_z c\beta \end{bmatrix} \tag{6.6}$$

onto the $\mathbf{z}_{A_i}$-axis). We have that,

$$\det(\mathbf{G}(\alpha,\beta)) = -54\cos\alpha\cos^2\beta\,(p_y\sin\alpha\cos^2\beta + c_f^\tau\sin^2\beta)^2\,p_z, \tag{6.5}$$

from which the singularities of the allocation matrix $\mathbf{G}(\alpha,\beta)$ can be easily identified by zeroing its determinant. In fact, $\det(\mathbf{G}(\alpha,\beta)) = 0$ if:

- **Case 1**: $\cos\alpha = 0$, i.e. $\alpha = (2k+1)\pi/2$, $k \in \mathbb{Z}$. In this case, $\text{rank}(\mathbf{G}(\alpha,\beta)) = 5$ and its *underactuation degree*[2] $\delta_A = 1$. The spinning axes $\mathbf{z}_{A_1},\ldots,\mathbf{z}_{A_6}$ of the six motors are aligned. If $\beta = 0$, they are tangent to the circular airframe (see Fig. 6.2-(a)), where $\alpha = \pi/2$): if $f_1 = f_2 = \ldots = f_6$, the hexarotor nominally stands still, balancing the gravity force.

- **Case 2**: $\cos\beta = 0$, i.e. $\beta = (2k+1)\pi/2$, $k \in \mathbb{Z}$. In this case, $\text{rank}(\mathbf{G}(\alpha,\beta)) = 4$ and $\delta_A = 2$ (see Fig. 6.2-(b)), where $\alpha = 0$ and $\beta = -\pi/2$). If $f_1 = f_2 = \ldots = f_6$, the hexarotor nominally stands still.

- **Case 3**:

$$p_y\sin\alpha\cos^2\beta + c_f^\tau\sin^2\beta = 0. \tag{6.7}$$

  As it easy to see, (6.7) is transcendental and its zeros cannot be determined in closed form. However, for a *fixed* $\alpha \in [k\pi - \arctan(\ell/c_f^\tau),\, k\pi]$, $k \in \mathbb{Z}$, we have $\beta = \pm\arctan\big(\sqrt{-\frac{p_y}{c_f^\tau}\sin\alpha}\,\big)$. For these pairs of α and β, $\text{rank}(\mathbf{G}(\alpha,\beta)) = 4$ and $\delta_A = 2$. Two values of α are of special interest:
  - **Case 3.1**: $p_y = 0$, i.e. $\alpha = -\arctan(\ell/c_f^\tau)$ (cf. [Giribet–2016a]). Note that $\ell/c_f^\tau > 0$, therefore $\arctan(\ell/c_f^\tau) > 0$.
  - **Case 3.2**: $\sin\alpha = 0$, i.e. $\alpha = k\pi$, $k \in \mathbb{Z}$. If $\alpha = 0$, the spinning axes $\mathbf{z}_{A_1},\ldots,\mathbf{z}_{A_6}$ of the six motors are parallel to $\mathbf{z}_B$ and the platform behaves as a *conventional hexarotor*.

[2] We recall that the *underactuation degree* δ_A of a mechanical system modeled as $\ddot{\mathbf{q}} = \boldsymbol{f}(\mathbf{q},\dot{\mathbf{q}},\boldsymbol{u})$, where $\mathbf{q}$ is the vector of generalized coordinates and $\boldsymbol{u}$ is the input vector, is the difference between the dimension of $\mathbf{q}$ and the rank of the Jacobian matrix $\partial\boldsymbol{f}/\partial\boldsymbol{u}$, which, in the case under examination, is equal to the rank of $\mathbf{G}$. If $\delta_A = 0$, the system is said *fully actuated*, disregarding the actuator constraints (as already mentioned in Chapt. 3, in the case of a MRAV we prefer the nomenclature FRA).

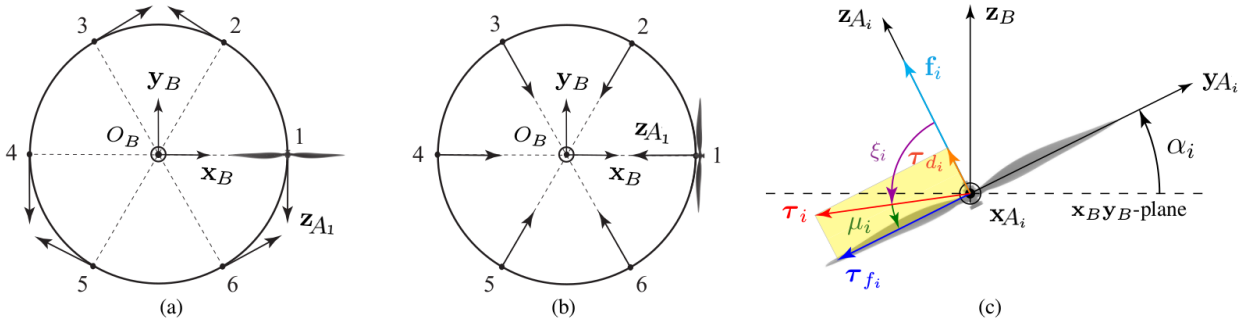


Figure 6.2 – Three of the four singular configurations of the hexarotor with dual-tilting propellers: (a) *Case 1* for $\alpha = \pi/2$ and $\beta = 0$ (top view); (b) *Case 2* for $\alpha = 0$ and $\beta = -\pi/2$ (top view); (c) *Case 4*: illustration of the forces and moments generated by propeller i. The propeller spins clockwise ($w_i < 0$) with a tilting angle $\alpha_i > 0$ about the $\mathbf{x}_{A_i}$-axis. The $\mathbf{x}_B\mathbf{y}_B$-plane is dashed.

- **Case 4**: $p_z = 0$, i.e.
$$\alpha = \arctan(c_f^\tau/\ell). \tag{6.8}$$
In this case, $\text{rank}(\mathbf{G}(\alpha,\beta)) = 5$ and $\delta_A = 1$.

Remark (Force and moment underactuation). Note that the singularities in Case 1 and Case 3.2 are sources of *force underactuation* for the hexarotor, whereas those in Case 3.1 and Case 4 of *moment underactuation*. On the other hand, the singularities in Case 2 cause both *force* and *moment underactuation* (cf. the 3rd and 6th row of $\mathbf{G}(\alpha,\beta)$ in (6.6)).

Remark (Special cases). With reference to (6.5), for $\beta = 0$, we have

$$\det \mathbf{G}(\alpha, 0) = -54\, p_y^2\, p_z \cos\alpha\, \sin^2\alpha,$$

while for $\alpha = 0$ it is

$$\det \mathbf{G}(0, \beta) = 54\, (c_f^\tau)^3 \cos^2\beta\, \sin^4\beta.$$

Geometric interpretation of moment singularities

Note that while the singularities of Case 1 and 2 can be easily avoided, since they correspond to orientations of the propellers of scarce practical interest, the singularities of Case 3.1 and Case 4 are subtler. In what follows, we will only focus on Case 4, since Case 3.1 is complementary. Fig. 6.2-(c) reports the forces and moments generated by propeller i which is tilted of an angle $\alpha_i = (-1)^{i-1}\,\alpha > 0$ about the $\mathbf{x}_{A_i}$-axis, $i \in \{1,3,5\}$ (the axis is directed out of the page at the reader). When propeller i rotates (in particular $\mathrm{w}_i < 0$ for $i \in \{1,3,5\}$) about its spinning axis $\mathbf{z}_{A_i}$, it generates a thrust force γ_i which, in turn, produces a moment $\boldsymbol{\tau}_{\gamma_i}$ at the CoM of the platform, and a drag moment $\boldsymbol{\tau}_{d_i}$. The magnitude of these vectors can be deduced from (3.10), (3.12), (3.17) and are recalled in the following:

$$\gamma_i = c_f\, \mathrm{w}_i^2, \tag{6.9}$$

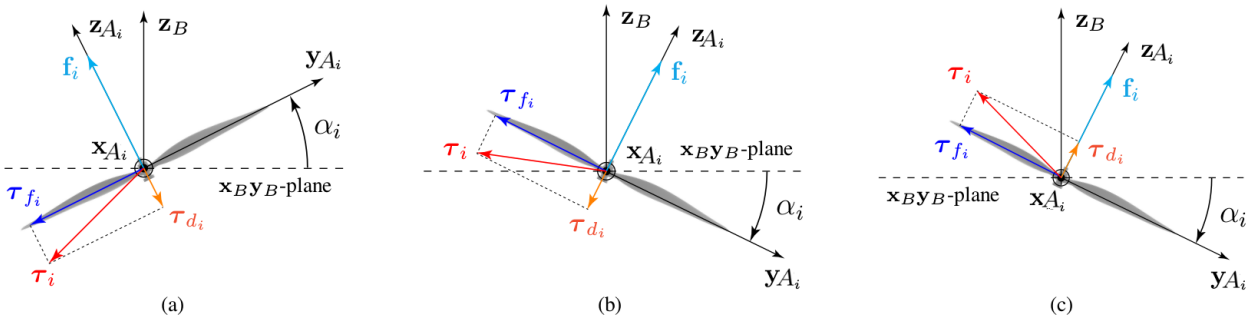


Figure 6.3 – *Case 4*: illustration of the other possible combinations of forces and moments that propeller i can generate (cf. Fig. 6.2. (a) $\alpha_i > 0$ and $\mathrm{w}_i > 0$; (b) $\alpha_i < 0$ and $\mathrm{w}_i > 0$; (c) $\alpha_i < 0$ and $w_i < 0$. The $\mathbf{x}_B\mathbf{y}_B$-plane is dashed.

$$\tau_{\gamma_i} = \ell\gamma_i = \ell\, c_f\, \mathrm{w}_i^2, \tag{6.10}$$
$$\tau_{d_i} = c_\tau\, \mathrm{w}_i^2, \tag{6.11}$$

where w_i is the angular velocity of propeller i about the $\mathbf{z}_{A_i}$-axis, and c_f and c_τ are the estimated thrust and aerodynamic drag factors of the propeller, respectively. To gain some physical insight into condition (6.8), it is convenient to study the total moment $\boldsymbol{\tau}_i$ applied at the Center of Mass (CoM) of the hexarotor by propeller i, which is given by $\boldsymbol{\tau}_i = \boldsymbol{\tau}_{\gamma_i} + \boldsymbol{\tau}_{d_i}$, where the two moment components are always orthogonal, by construction. From Fig. 6.2-(c) (see the yellow-shaded rectangle), we deduce that:

$$\xi_i + \mu_i = \pi/2, \tag{6.12}$$
$$\tan\xi_i = \tau_{\gamma_i}/\tau_{d_i}. \tag{6.13}$$

By plugging (6.12) into (6.13), we find that $\tan\mu_i = \tau_{d_i}/\tau_{f_i}$. Setting $c_f^\tau = c_\tau/c_f$, substitution of (6.10)-(6.11) into the previous equation yields $\tan\mu_i = c_f^\tau/\ell$, which tells us that the direction of the total moment generated by propeller i at the CoM of the hexarotor only depends on two physical parameters, c_f^τ and ℓ, and on the tilting angle α. Notably, the direction of $\boldsymbol{\tau}_i$ does *not* depend on the angular velocity of the propeller itself. If $\mu_i = \alpha_i$, then $\boldsymbol{\tau}_i$ belongs to the $\mathbf{x}_B\mathbf{y}_B$-plane and propeller i cannot generate moments along $\mathbf{z}_B$ anymore (see Fig. 6.2-(c)). If this condition holds for all propellers, i.e. $\mu_i = |\alpha_i|$, $\forall\, i$, then the entries of the 6th row of $\mathbf{G}(\alpha,\beta)$ (which maps the vector of the actuator forces onto moments about the $\mathbf{z}_B$-axis) are zero and the matrix becomes rank-one deficient. This corresponds to condition (6.8), previously obtained in a purely algebraic fashion. Note that Case 3.1 is even more critical than Case 4, since this time $\boldsymbol{\tau}_i$ is aligned with $\mathbf{z}_B$, yielding a rank-two deficient $\mathbf{G}(\alpha,\beta)$ (the 4th and 5th row become zero). Fig. 6.3 complements Fig. 6.8-(c) by reporting the other possible combinations of forces and moments that propeller i can generate according to the signs of α_i and w_i.

Remark (FAST-Hex case). Note that $c_f^\tau = c_\tau/c_f$ is "small", since the aerodynamic drag factor c_τ is typically one order of magnitude smaller than c_f. As a consequence,

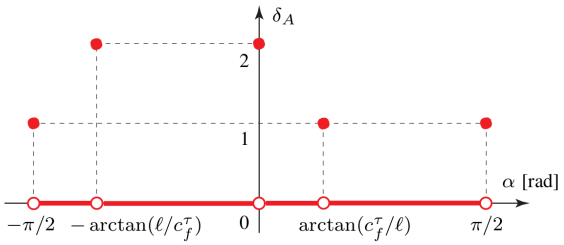


Figure 6.4 – Values assumed by δ_A for $\alpha \in [-\pi/2, \pi/2]$ and $\beta = 0$.

$\alpha = \arctan(c_f^\tau/\ell)$ is "small" as well. For instance, if we take $c_f^\tau = 1.7 \times 10^{-2}$ m and $\ell = 0.315$ m as in [Ryll–2016] (which describes the convertible hexarotor that will be presented in this chapter) we obtain a critical $\alpha \approx 4\,\text{deg}$. Hence, if one decides to increase α starting from zero in order to transform the standard Uni-Directional Thrust (UDT) star-shaped hexarotor into a MDT platform, a singularity will be crossed if the propeller which spins CCW are tilted of a negative α angle and the ones rotating CW are tilted of a positive α angle. This could result in a temporary loss of maneuverability. In order to avoid this behavior, the CCW propellers must be tilted of a positive angle α_i, and the CW ones of a negative angle α_i, which the simplest and most effective solution if the platform is supposed to often cross this transient phase. Fig. 6.5 depicts the nominal actuator forces (calculated in open-loop by inverting (3.13), as done in (3.19)) to let the FAST-Hex MRAV [Ryll–2016] hover, as a function of the tilting angle α. On the left plot, related to a rotor configuration like the one of *Case 3* (c.f. Fig. 6.2-(c)), the allocation matrix singularity for $\alpha \approx 4$ causes some forces to diverge. On the other hand, it should be appreciated that in the second plot this does not happen, thanks to the adoption of the aforementioned solution, which removes the singularity from that interval.

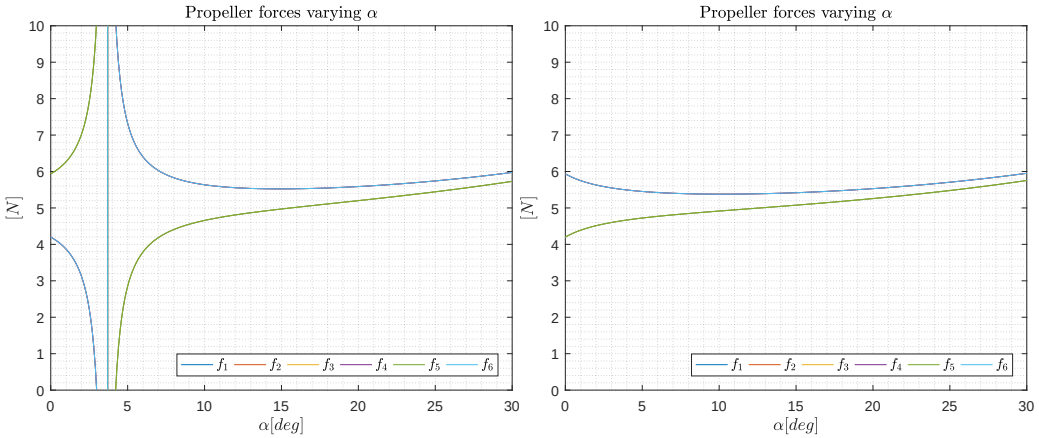


Figure 6.5 – Nominal hovering actuator forces for the FAST-Hex MRAV as a function of the tilting angle α. *Left:* actuator arrangement as in *Case 3* (c.f. Fig. 6.2-(c)), which causes the allocation matrix singularity. *Right:* rotor arrangement as proposed in the previous remark, which solves the singularity. Some signals are superimposed.

Figure 6.6 – CAD model of the Tilt-Hex prototype.

The Tilt-Hex prototype

The tilted hexarotor, Tilt-Hex for short, is the first LAAS–CNRS in house developed MDT MRAV. The design of the actuators arrangement follows the line described in the previous section, i.e., the actuators are arranged in star-shape and are tilted of a fixed quantity w.r.t. the body frame as shown in Fig. 6.1. The particular choice of α and β is a heuristic compromise between maximum lateral forces and a minimization wasted internal forces [Rajappa–2015; Franchi–2018]. All used structural components are either off-the-shelf available or 3D printable by a standard fused deposition modeling printer. The CAD model of the robot is depicted in Fig. 6.6.

The flying platform is composed of a rigid structure made by six aluminum bars and a central plate, for mechanical robustness and ease of maintenance. The electronics is composed of six Electronic Speed Controller (ESCs) BL-Ctrl-2.0 from MikroKopter, running an in-house developed firmware that performs closed-loop spinning frequency control [Franchi–2017] and accepts desired spinning frequency at 1 kHz. This allows a fine control of the propeller spinning velocity, coupled with a static map of the force and moment produced at a given spinning velocity, it results in a precise force control of the platform. The physical parameters of the platform are condensed in Tab. 6.1 In particular, the mass has been measured directly, while the inertia tensor has been estimated by a high detailed CAD model.

The actuation units, located at the end of each bar, are composed of a MK3638 brushless motor from MikroKopter and 12" propellers with 4.5" pitch, each developing a maximum thrust of about 10 N. They are fixed via 3D-printed adapters in order to provide the necessary tilting angles, see Fig. 6.6 and Fig. 6.7, for the multi-directional thrust capability. The propeller tilting angles are $|\alpha| = 35°$, distributed in an alternated fashion, and $\beta = -25°$, which guarantees a well-balanced choice between maximum lateral forces and losses due to internal forces in contact-less hovering. The consistent value of α is motivated by the fact that this angle is the one

Table 6.1 – Physical parameters of the Tilt-Hex prototype.

Tilt-Hex		
Parameter	Value	Unit
m	1.86	Kg
$\mathbf{J}(:,1)$	$[0.11\ 0\ 0]^\top$	$\mathrm{Kg\,m^2}$
$\mathbf{J}(:,2)$	$[0\ 0.11\ 0]^\top$	$\mathrm{Kg\,m^2}$
$\mathbf{J}(:,3)$	$[0\ 0\ 0.19]^\top$	$\mathrm{Kg\,m^2}$
c_i	$(-1)^{i-1}$	[ ]
c_f^τ	1.9×10^{-2}	m
c_f	9.9×10^{-4}	$\mathrm{N/Hz^2}$
$\mathbf{R}_{A_i}^B$	$\mathbf{R}_z((i-1)\frac{\pi}{3}))\mathbf{R}_x(\alpha_i)\mathbf{R}_y(\beta)$	[ ]
$\mathbf{p}_{A_i}^B$	$\mathbf{R}_z((i-1)\frac{\pi}{3}))[\ell\ 0\ 0]^\top$	[ ]
α_i	$(-1)^i\ 35$	deg
β	-25	deg
ℓ	0.368	m

that mostly contributes to the well conditioning of the allocation matrix [Giribet–2016a; Michieletto–2017; Michieletto–2018]. On the other hand, β could have been chosen much smaller (we will see in the following that this angle is much smaller for the other prototypes and it has even not been used for some of them). In this case, the main motivation towards using a quite big value for this second angle is that this platform has been also used to study the problem of the robustness towards the loss of a propeller [Michieletto–2018]. In this work, the authors formalized the study of the rotor failure robustness for different MDT MRAV, showing that if β is big enough, the platform moment actuation capabilities do not degenerate in the case of one propeller loss.

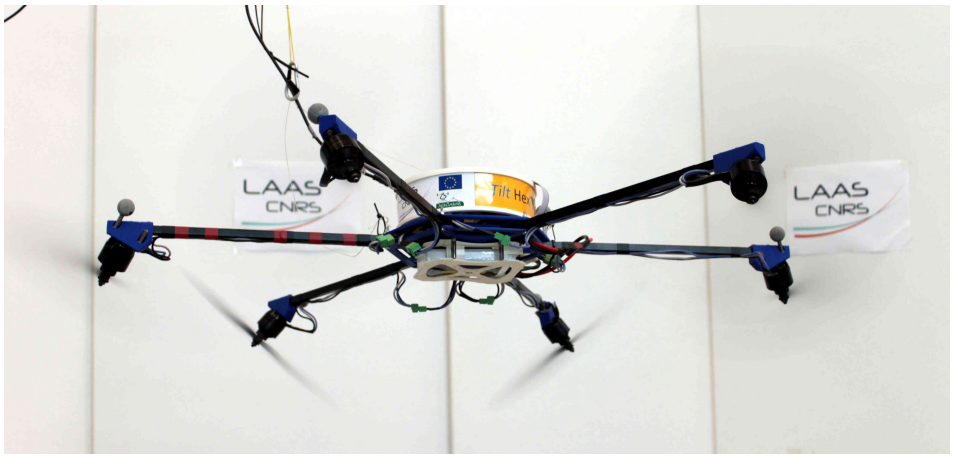


Figure 6.7 – Photo of the real Tilt-Hex prototype.

Figure 6.8 – CAD model of the FAST-Hex prototype.

The FAST-Hex prototype

The fully-actuated by synchronized tilting propellers hexarotor, FAST-Hex for short, is a novel convertible MRAV concept which is able to smoothly change its configuration from UDT to MDT by using only one additional motor that tilts all propellers at the same time (in a similar alternated fashion to the one of the Tilt-Hex). The FAST-Hex can adapt to the task at hand by finely tuning its configuration from the under-actuated (but efficient, in term of energy consumption) trajectory tracking, typical of collinear multi-rotor platforms, to the partial dynamic decoupling given by the FRA capability (yet less efficient), which is attainable by non-collinear multi-rotors. The CAD model of the prototype is shown in Fig. 6.8. Its main feature is the ability to tilt the six propellers of an angle α synchronously by using a transmission system connected to a single servomotor. Such system is made up of cardanic[3] joints and worm-shafts coupled with gears. For this platform, we have chosen $\beta = 0$ for all the actuators, both for an ease of implementation and to be able to attain also the UDT configuration.

The FAST-Hex has two structurally different configurations:

i) $\alpha = 0 \quad \Rightarrow \quad \text{rank}\,(\mathbf{G}(\alpha = 0)) = 4$

ii) $\alpha \in \mathcal{A}\backslash\{0\} \quad \Rightarrow \quad \text{rank}\,(\mathbf{G}(\alpha)) = 6.$

In configuration *i)* all propellers of the FAST-Hex are co-planar or, equivalently, their spinning axes are all collinear. The system degenerates to an ordinary UDT hexarotor platform. In this configuration, the internal forces during hovering are zero. Only internal moments due to the drag moment appear. These are typically

[3]Of, pertaining to, or devised by Cardan (Girolamo Cardano, 1501–1576), a noted Italian physician, mathematician, and astrologer.

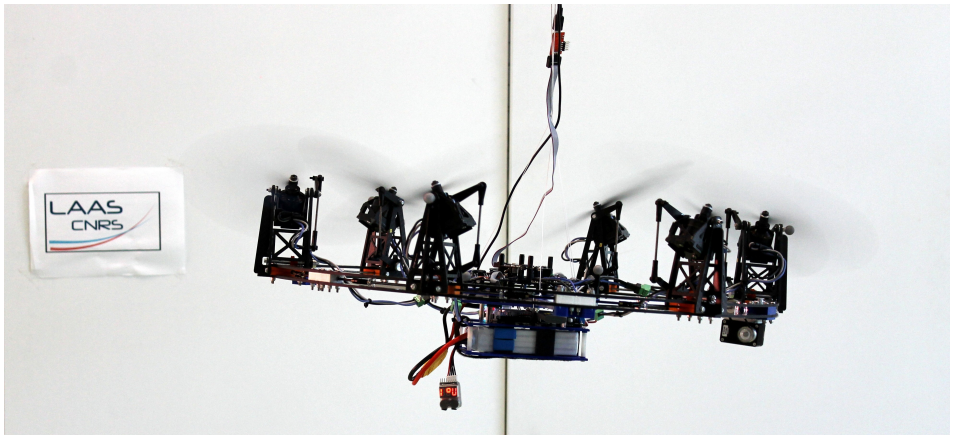


Figure 6.9 – Photo of the real FAST-Hex prototype.

one order of magnitude less strong than the moments generated by the thrust moments and therefore neglected in the following efficiency considerations. We model the wasted force using the following index:

$$\eta_f(\alpha, \boldsymbol{\gamma}) = \frac{\| \sum_{i=1}^{6} \mathbf{f}_i^B(\gamma_i, \alpha) \|}{\sum_{i=1}^{6} \| \mathbf{f}_i^B(\gamma_i, \alpha) \|} = \frac{\| \sum_{i=1}^{6} \mathbf{f}_i^B(\gamma_i, \alpha) \|}{\sum_{i=1}^{6} \gamma_i} \in [0, 1], \tag{6.14}$$

that we call the *force efficiency* index. It is easy to check that $\eta_f(\alpha = 0, \boldsymbol{\gamma}) = 1$ for any input $\boldsymbol{\gamma}$, which corresponds to maximum efficiency. Hence the configuration *i)* is energetically very efficient. This comes with the drawback that the platform is under-actuated and a simultaneous tracking of fully independent $\mathbf{p}_r(t)$ and $\mathbf{R}_r(t)$ is impossible. In the configurations of type *ii)* the internal forces in hovering are greater than zero, which means that the system is wasting more energy than in configuration *i)*. The larger $|\alpha|$, the larger the internal forces. This is clearly visible from the fact that $\eta_f(\alpha \in \mathcal{A} \backslash \{0\}, \boldsymbol{\gamma}) < 1$. In particular, during nominal horizontal hovering, when all the propellers are spinning at the same speed producing the same force f, we have that $\eta_f(\alpha, f\mathbf{1}_{6\times 1}) = \cos\alpha$. If the platform is following a non-hovering trajectory then $\eta_f(\alpha, \boldsymbol{\gamma})$ is in general different from $\cos\alpha$ and one has to use (6.14) to exactly compute it. On the other hand, in configurations of type *ii)* the platform gains the MDT (in particular the FRA) capability, and the larger $|\alpha|$, the larger the volume of admissible total forces $\mathbf{f}_B^B$.

Remark ("configuration" vs. "instantaneous input"). Due to the fact that α is a slowly changeable parameter, it should not be considered as an *instantaneous input*, but rather as a *configuration* of the system [Furci–2018] which can be modified by a high-level slow-rate controller/planner. The high-level controller can gently tune α while flying, thus continuously changing the platform between configuration *i)* and any of the configurations of type *ii)* in order to adapt to the particular task being executed. For example, configuration *i)* can be chosen when a pure horizontal hovering is requested, while a type *ii)* configuration can be selected when hovering with non-zero roll and pitch is needed. Alternatively, $\dot{\alpha}$ can be considered as an

Table 6.2 – Physical parameters of the FAST-Hex prototype.

FAST-Hex		
Parameter	Value	Unit
m	2.4	Kg
$\mathbf{J}(:,1)$	$[0.042\ 0\ 0]^\top$	$\mathrm{Kg\,m^2}$
$\mathbf{J}(:,2)$	$[0\ 0.042\ 0]^\top$	$\mathrm{Kg\,m^2}$
$\mathbf{J}(:,3)$	$[0\ 0\ 0.083]^\top$	$\mathrm{Kg\,m^2}$
c_i	$(-1)^i$	[ ]
c_f^τ	1.9×10^{-2}	m
c_f	9.9×10^{-4}	$\mathrm{N/Hz^2}$
$\mathbf{R}_{A_i}^B$	$\mathbf{R}_z((i-1)\frac{\pi}{3}))\mathbf{R}_x(\alpha_i)\mathbf{R}_y(\beta)$	[ ]
$\mathbf{p}_{A_i}^B$	$\mathbf{R}_z((i-1)\frac{\pi}{3}))[\ell\ 0\ 0]^\top$	[ ]
α_i	$(-1)^{i-1}\ \lvert\alpha\rvert$	deg
β	0	deg
ℓ	0.315	m

additional control input subjected to some limitations to which the controller has to comply with, as done in Sec. 7.2.

Inside the structural ring there are eleven carbon fiber axles, forming a polygon inscribed in the ring, all connected by universal cardanic joints: these allow the propagation of the rotation of the bars throughout the ring. The central axle is attached to a motor, responsible for the actuation of the tilting mechanism. The motor box, a Dynamixel XM430, comprising a Maxon DC motor, a ST CORTEX-M3 MCU and a 12 bit contact-less encoder, divides the axle chain in two parts, forming two sub-chains which are left "open" from the other side. From experimental tests, we have assessed that splitting the whole chain in two sub-parts greatly minimizes friction phenomena and torsion effects of the carbon fiber parts, which, in the case of longer chains, could induce jerky movements on the parts located far from the motor box. Every second axle is endowed with a worm-shaft, coupled with a worm gear, that is responsible for the transmission of the tilting rotation to the corresponding actuator. This solution prevents back-drivability and backlash. Opposed tilting of neighboring actuators, i.e., $\alpha_i = (-1)^{i-1},\ i = 1, \ldots, 6$, is achieved by alternating left and right turning worm drives. The worm shafts and the gears are realized with a high-precision 3D printer, given that the quality of these parts affects the accuracy of the tilting angles. The maximum absolute value of the tilting angle is mechanically limited to $\approx 35\,\mathrm{deg}$ to prevent collision of the propellers with the ring frame. Due to the cardanic joint transmission, the absolute values of the tilting angles of each actuators cannot be exactly kept the same while α changes. In particular, we have that $\alpha_1 = -\alpha_6 = \alpha$, while $\alpha_2 = -\alpha_5 =: -\alpha'$ and $\alpha_3 = -\alpha_4 =: \alpha''$. These discrepancies are mitigated by the reduction rate $\tilde{n} = 20$

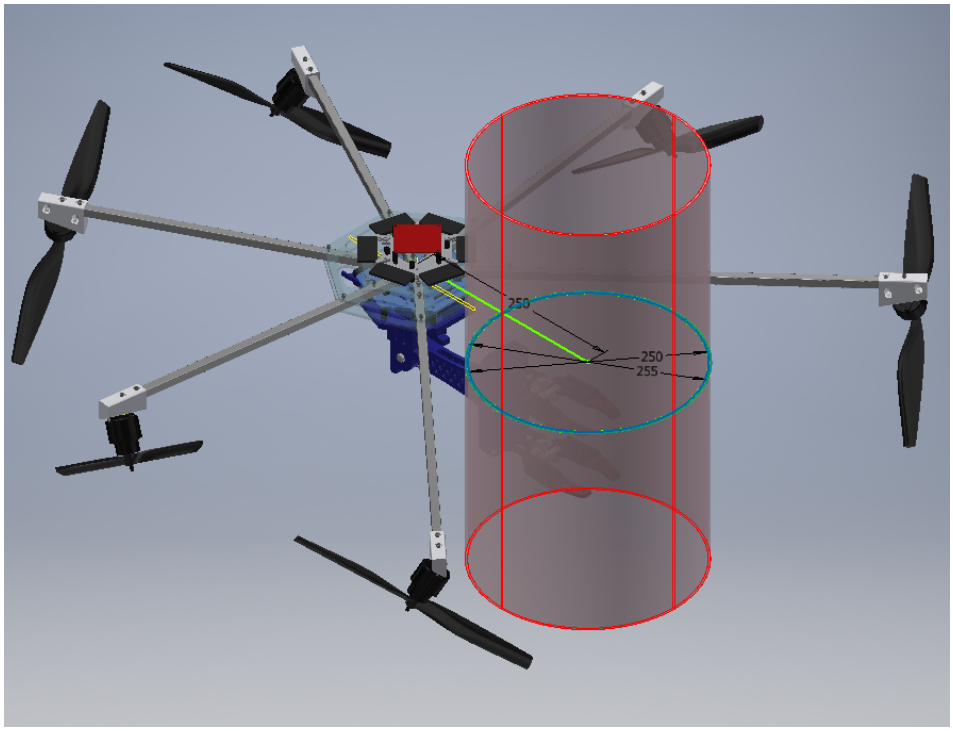


Figure 6.10 – CAD model of the OT-Hex prototype, with detail of the aperture.

of the worm shafts, thus producing a maximum discrepancy below 0.8 deg. The physical parameters of the FAST-Hex are condensed in Tab. 6.2, while a photo of the real prototype is depicted in Fig. 6.9.

The OT-Hex prototype

The open and tilted hexarotor, OT-Hex for short, is a MDT MRAV tailored to perform particular physical interaction tasks, i.e., lifting long objects like, e.g., metallic beams. One mechanical challenge during aerial manipulation is to prevent collision between the manipulated load and the spinning propellers. This issue is easily handled when manipulated objects can remain at a safe distance under the volume occupied by the spinning propellers, but it can not be managed for more complex tasks such as bar lifting. An approach to handle long bars is to use a robotic arm with a workspace large enough to handle the bar outside of the platform perimeter. This approach costs an increased weight and a decreased bar payload. Furthermore, when the arm extends, the force disturbances from the load side may generate large destabilizing moments on the aerial vehicle. To tackle this issue, the OT-Hex is designed with a non-regular frame arrangement, introducing a consistent aperture between the two front propellers (see Fig. 6.10 and Fig. 6.11), hence the denomination *Open*. In particular, the frontal aperture of $\delta_A = 85\,\text{deg}$ guarantees the possibility to safely manipulate a cylindrical bar with a maximum diameter of $d_{B,\,\text{max}} = 255\,\text{mm}$, as shown in Fig. 6.10. This configuration has been

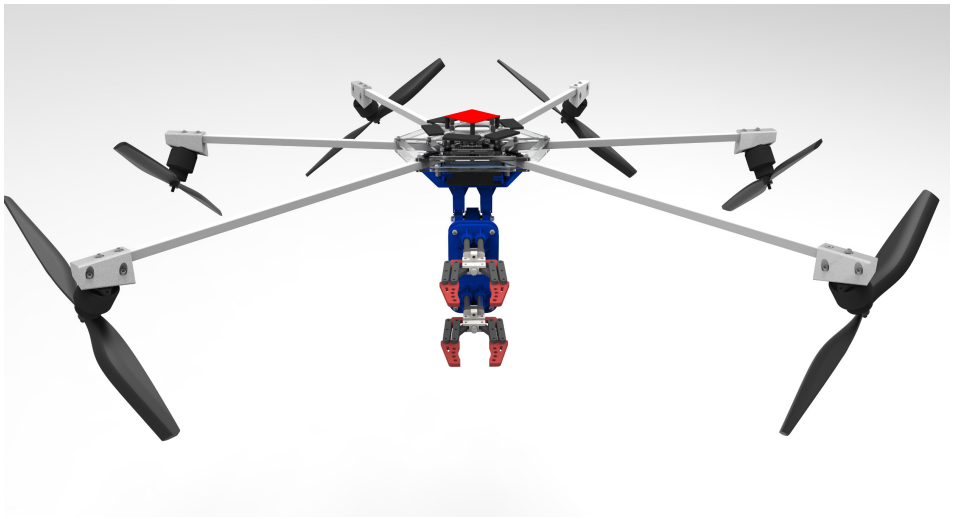


Figure 6.11 – CAD model of the OT-Hex prototype.

preferred over the regular hexagonal positioning of the actuation units (like the one of the Tilt-Hex) also in order keep a safety distance between the spinning propellers and the manipulated load and so to avoid dangerous collisions that could jeopardize the fulfillment of the task. Finally, it should be appreciated that this solution allows for a wider variety of beam manipulation tasks and eases the interaction, since the load is kept close to the platform CoM, thus minimizing disturbance moments. In order to compensate for the non-regular location of the propellers and the shift of CoM of the total platform w.r.t. a regular arrangement, the two frame elements delimiting the aperture are longer than the others. This ensures that in contact-free hovering the control effort is equally distributed between the six propellers.

Figure 6.12 – Photo of the real OT-Hex prototype.

Table 6.3 – Physical parameters of the OT-Hex prototype.

OT-Hex		
Parameter	Value	Unit
m	2.48	Kg
$\mathbf{J}(:,1)$	$[0.124\ 0\ 0]^\top$	$\mathrm{Kg\,m^2}$
$\mathbf{J}(:,2)$	$[0\ 0.110\ 0]^\top$	$\mathrm{Kg\,m^2}$
$\mathbf{J}(:,3)$	$[0\ 0\ 0.192]^\top$	$\mathrm{Kg\,m^2}$
c_i	$(-1)^{i-1}$	[ ]
c_f^τ	1.9×10^{-2}	m
c_f	9.9×10^{-4}	$\mathrm{N/Hz^2}$
$\mathbf{R}_{A_i}^B$	$\mathbf{R}_z(\zeta_i)\mathbf{R}_x(\alpha_i)\mathbf{R}_y(\beta)$	[ ]
$\mathbf{p}_{A_i}^B$	$\mathbf{R}_z(\zeta_i)[\ell\ 0\ 0]^\top$	[ ]
ζ_{1-3}	$42.5, 97.5, 152.5$	deg
ζ_{4-6}	$207.5, 262.5, 317.5$	deg
α_i	$(-1)^{i-1}\ 35$	deg
β_i	-10	deg
ℓ	0.368	m
δ_A	85	deg
$d_{B,\max}$	255	mm

Thanks to the ability to exert a multi-directional thrust, the need for an actuated arm might be relieved for a good set of tasks. Indeed, thanks to the additional Degrees of Freedom (DoFs) of the MRAV, the End-Effector (EE) might reach the desired configurations without the need of an arm with actuated joints, as it will be shown in Sec. 10.1. Thus, a mechanically simpler 1-DoF passive arm can be used to perform complex and dexterous manipulation tasks. This is a clear example of how the MDT capability allows to reduce the system complexity and consequently, to decrease the robot weight, thus increasing the available net payload.

The OT-Hex chassis is similar to the one of the Tilt-Hex, with a central 3D-printed plate and six aluminum rods, for mechanical robustness and ease of maintenance. The actuators are also the same, i.e., MK3638 brushless motors from MikroKopter, coupled with 12" propellers with 4.5" fixed pitch. Apart from the frontal aperture, another difference w.r.t. the Tilt-Hex is found in the second tilting angle of its actuators. In this case, the angle β has been reduced in order to save energy, thus optimizing the consumption and giving the robot more available payload. Below the flying platform, a 1-DoF passive manipulator is fixed. It is

composed of a passive revolute joint, a 3D-printed arm and a set of two Makeblock[4] robotic grippers. In particular, two grippers were used to minimize mechanical stress on those lightweight grippers. The base of the manipulator is a 3D-printed structure, also serving as electronics and battery case, and is mounted directly on the central plate of the flying platform. The grippers have been chosen for their lightweight and easy integration characteristics. The gripper electronics is composed of an Arduino Nano board and some power converters, encased in the 3D-printed structure. The main physical parameters of the OT-Hex are condensed in Tab. 6.3, while a photo of the real prototype is depicted in Fig. 6.12.

Omni-Directional Thrust octorotor

In this section, we briefly introduce the design of an omni-directional thrust octorotor, i.e., a MDT MRAV with eight actuators that can produce any body force inside a limited spherical shell independently from the body moment. Such aerial robot has been conceived inside the [AeRoArms–] H2020 European Project and is the result of a collaboration between the two laboratories LAAS–CNRS, mostly involved in the design part, and DLR[5], in charge of the mechanical implementation and exploitation of the aerial prototype. The suspended aerial manipulator, SAM for short, is part of a complex system conceived in order to perform complex aerial manipulation tasks like, e.g., inspection with contact of pipelines in a refinery scenario, deployment of sensors and/or smaller robots, tele-manipulation tasks involving polishing, drilling, cutting an object of interest, etc.

To address these and other aerial manipulation tasks, in [Kondak–2014] the authors designed and built a system composed of a small scale autonomous helicopter to which a 7 DoF manipulator is attached. The authors exploited the null-space of the manipulator in order to decrease the moment disturbance induced on the helicopter by constraining the arm CoM to stay close to the projection of the helicopter CoM in the horizontal plane. They showed this precaution to be beneficial for the system stability, since the vibrations produced on the helicopter were highly decreased. Related problems have been investigated also in [Kim–2018a]. In [Kim–2018b], a similar system concept was proposed, but with additional moving masses to damp out the oscillations caused by the long reach configuration. However, the use of moving masses does not allow to generate a MDT wrench, which is beneficial in interaction problems. Additionally, moving masses unnecessarily increase the total weight. In [Suarez–2018a; Suarez–2018b] MRAVs with long reach manipulators have recently been developed, where the arm manipulator is attached to the MRAV using a long flexible link instead of being mounted directly. This allows to perform the aerial manipulation in narrow and complex environments while keeping the MRAV at a safe distance from the obstacles. However, the use of flexible links drastically complicates the control problem.

[4] `https://www.makeblock.com/`

[5] German Aerospace Center, Oberpfaffenhofen, Germany.

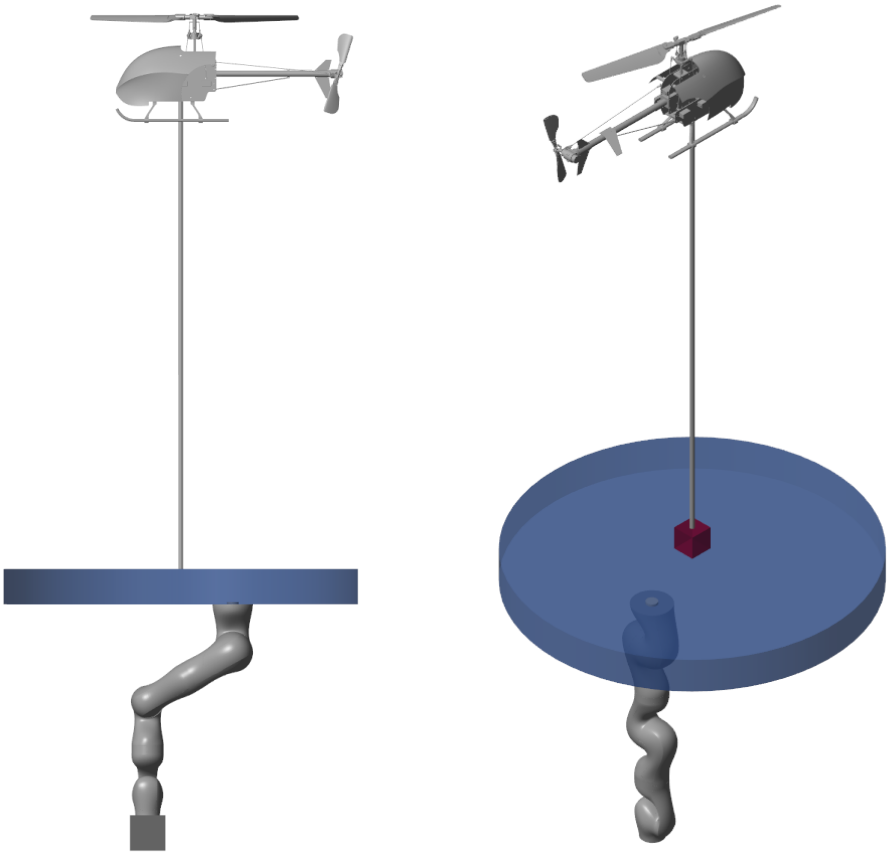


Figure 6.13 – CAD model of the complete system for aerial manipulation.

Despite different interesting aspects investigated in these works, there are still important limitations on the operational space of the manipulator that depend on the structure of the system. Indeed, with these solutions the operational space of the arm end-effector results dramatically reduced, due to the landing skids of the helicopter. Moreover, one has also to consider that, for security reasons, the helicopter blades must be kept at a certain distance from any object, in order to avoid collisions which could prevent the fulfillment of the task or, in the worst case, de-stabilize the system. Finally, the big blades could also induce a non-negligible turbulence on the manipulated object and on the ground, thus serving as an additional source of danger for the helicopter.

The novel proposed solution, portrayed in the sketch of Fig. 6.13, foresees to mount the robotic arm on an active multi-rotor platform, which is itself suspended on the carrier (an helicopter or a much bigger MRAV) by means of a cable. As a result, higher safety can be achieved because the aerial carrier can keep a distance from the obstacles and because the smaller rotor blades should generate less disturbance on the load. Finally, a proper design of the multi-rotor platform should allow for a considerable enlargement of the arm EE workspace.

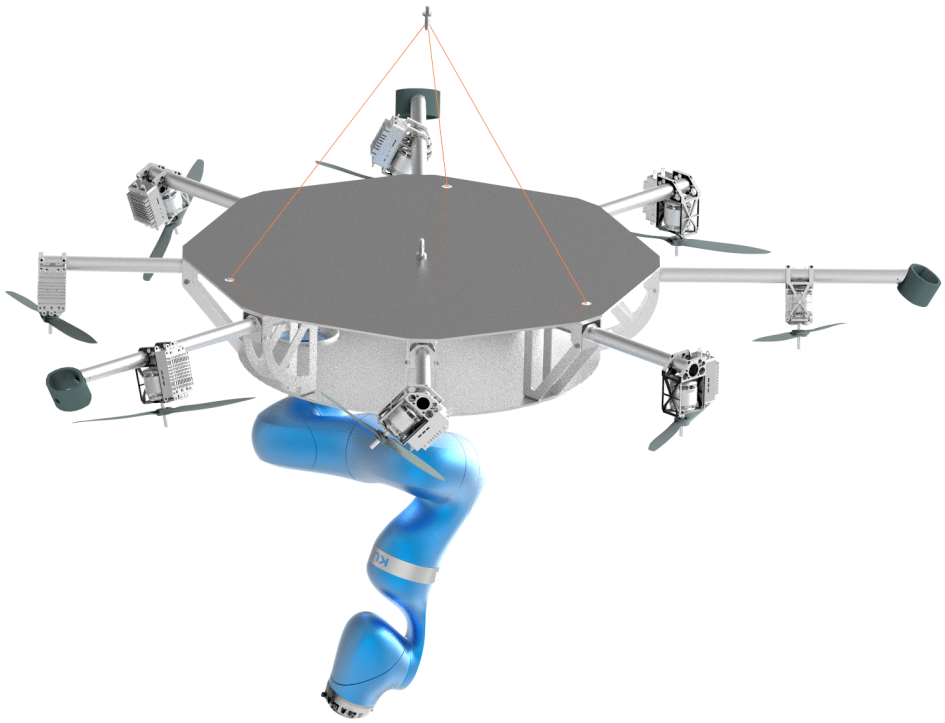


Figure 6.14 – CAD model of the SAM prototype.

The SAM prototype

In this subsection, we will describe the main functional components of the developed platform, whose CAD model is depicted in Fig. 6.14, and present its features.

Landing gear mechanism

In order to reduce the total weight and to accommodate large workspace of the manipulator, the traditional landing gear is not installed in the cable-suspended platform. To land, three out of eight frame propeller arms can be folded and converted to the legs of the landing gear, see Fig. 6.15. Thus, the legs have a dual use: landing legs and propeller arms. In order to switch from the landing leg to the propeller arm, each leg has to be risen from the lower to the upper position. In this transition, the leg rotates in the bearing through about 60 deg. To lift up the leg, the thrust force of the propulsion unit is used. In order to compensate a non-zero wrench caused by mentioned thrust during this procedure, propulsion units of the remaining five propeller arms are used. The downward movement (transition from bar frame to the landing leg) is accomplished by the gravity force, properly damped by the thrust force. To fix the position of the leg/arm in the bottom and top points, a locker actuated mechanism is used. By virtue of the transformable landing gear, the SAM has a larger workspace for manipulation without any restrictions created by the common landing gear, e.g., skids.

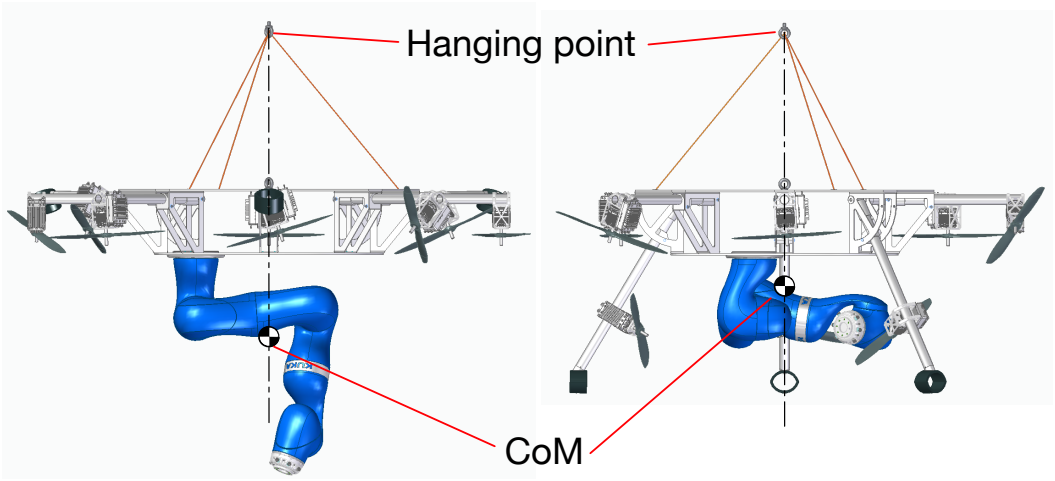


Figure 6.15 – Operation and parking configurations of the SAM.

Robotic manipulator arm

To perform arbitrary manipulation tasks, a 7-DoF KUKA LWR 4 is mounted on the bottom side of the platform. Two main postures of the robotic arm should be defined: operation and parking (see Fig. 6.15). During the transportation and landing, the manipulator should be in the parking position. Before initiating any manipulation task such as pick and place or peg-in-hole, the manipulator arm should be placed in the operation configuration as fewer movements are required to perform any typical manipulation task from this position. Additionally, it is worth mentioning that the manipulator mounting point is shifted from the center of the platform. It allows to choose the parking and operation configurations in such a way that the resulting overall CoM is the same for both (in the horizontal plane), see Fig. 6.15.

Propulsion units (actuators)

The design of the propulsion unit represents a key point in the development of the SAM, since it directly affects the dynamic behavior of the robot and its manipulation capability. Motivated by the fact that the suspended platform should be able to stabilize the pendulum-like system, while resisting dynamic disturbances induced by the motion of the arm and of the manipulated load, we chose the design in order to obtain an omni-directional platform, capable to exert a 6D wrench of forces and moments in all the directions. Planning to use fixed-pitch propeller, to avoid over-complexity in the system, this property should be achieved with unidirectional-thrust actuators. To this purpose, we referred to the theory developed in [Tognon–2018b] and therein. In this work, an algorithm that generates such design a is proposed. The solution guarantees the equal distribution between the actuators of the effort required to generate wrenches in any direction, while keeping the smallest component of the input lower-bounded by a minimum feasible value.

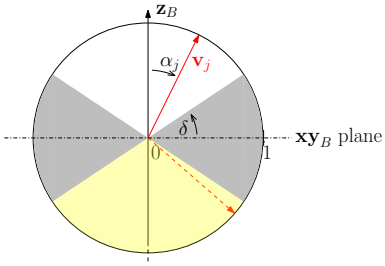


Figure 6.16 – Illustration of constraint *iii)* in the design optimization.

Remark (on the actuator number). It is worthwhile to mention that, the minimum required number of propulsion units to realize 6 DoF body wrench with unidirectional thrust actuator is equals to seven. We decided to use eight units in order to address the exploitation of redundancy in future studies.

For the design of this platform, the following constraints were considered: *i)* the imposition of a particular (the unitary) eigenvector for the allocation matrix in order to obtain a balanced design; *ii)* the normalization constraints for the unit vector defining the directions of the actuators; *iii)* the imposition of a minimal installation α angle of the propulsion units $\{2, 5, 8\}$, which guarantees the lifting of the landing legs (needed only during pre-operation phase) with attainable motor thrust; *iv)* the perpendicularity between the thrust directions and the frame arm axes, i.e., $\beta = 0$, for an ease of mechanical implementation. Mathematical treatment of *i)* and *ii)* can be found in [Tognon–2018b]. As far as *iii)* in concerned, it has been added to provide a minimum projection of the motor thrust vector along the z-axis of the body frame. This constraint can be mathematically expressed as $|\mathbf{z}_B^\top \mathbf{v}_j| \geq sin(\delta_p)$, where $\mathbf{v}_j = \mathbf{R}_{A_j}^B \mathbf{e}_3$, $j \in \{2, 5, 8\}$, are the unit vectors indicating the direction of the propulsion unit thrusts associated with the landing legs, expressed in body frame. In particular, $\delta_p = \pi/6$ is the angle selected to allow the lifting of a leg of 1.5 kg with a leverage arm of 0.75 m, given the maximum thrust force attainable by the chosen actuators in their worst operational mode (see reasonings in the following). Such constraint prevents allocation of the thrust vectors in the gray area of Fig. 6.16. In case the result of the optimization for some of the legs is a vector pointing downward (i.e., lying in the yellow area in Fig. 6.16), in order to generate the upward component of the force needed to initially lift that leg, it will be necessary to spin the propeller in the opposite sense w.r.t. the one it was designed to spin, before the manipulation task begins. However, it is worthwhile to mention that the thrust coefficient of a fixed-pitch propeller which driven in the 'opposite' sense can be very different (typically much smaller) from the nominal one (the one obtained by spinning in the 'correct' sense), as mentioned in Sec. 3.2. For this reason, the value $\delta_p = \pi/6$ was found after performing worst-case experimental tests with the selected hardware components, as shown in the left photo of Fig. 6.17. Finally, constraint *iv)* ensures $\beta_i = 0$ and can be written as ${\mathbf{p}_{A_i i}^B}^\top \mathbf{v}_k = 0$, $\forall i, k \in \{1, \ldots, 8\}$. This constraint contributes to limit the mechanical complexity of the structure and also to avoid possible collisions of the propellers with the frame arms.

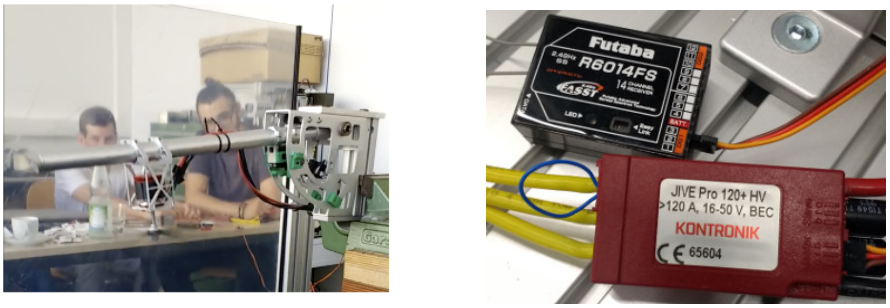


Figure 6.17 – *Left:* leg lifting test in the worst operational mode (spinning in the 'opposite' sense). *Right:* ESC and receiver used for the identification of the propeller parameters.

Ultimately, the remaining design parameter, which is not considered in the optimization problem, is the required thrust value per motor. To calculate this value, a desirable set of wrenches $\boldsymbol{w}^\star$ was estimated in preliminary simulations to compensate disturbances caused by a sequence of typical manipulator postures. Consequently, the thrust force values has been computed, i.e., $\boldsymbol{u} = \boldsymbol{A}^\dagger \boldsymbol{w}$. At this point, we chose from data-sheet the actuators (both the motors and the propellers) in order to guarantee the disturbance compensation with some margin. From the simulations, we observed that the maximum continuous thrust value required per motor was around 20 N. Selecting a maximum value of 40 N for each motor guarantees robust margin toward additional payload and disturbance. To provide the necessary thrust value, Kontronik[6] Pyro 650-65 was chosen as the DC motor, in pair with 16X6" propellers made in carbon fiber. In our experimental setup, this pair could exert continuous 50 N thrust during a typical working condition.To retrieve this, we mounted one actuator on top of a force/torque sensor and commanded different values of Pulse Width Modulation (PWM) by sending to the ESC some commands from a remote controller, see Fig. 6.18, while measuring the wrench. This allowed us to estimate the static parameters of the propeller, see Fig. 6.19.

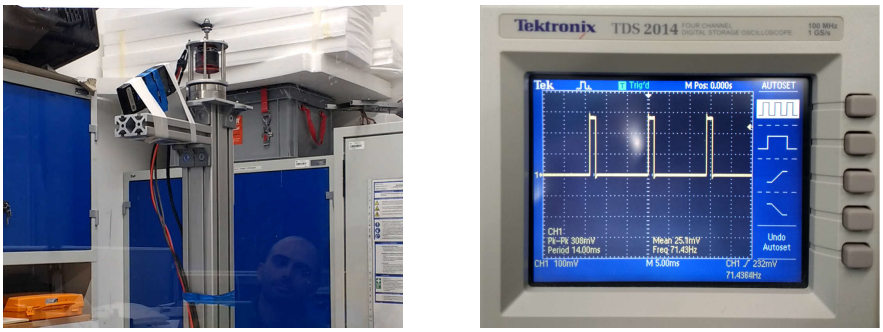


Figure 6.18 – Setup used for the identification of the propeller parameters. *Left:* actuator mounted on top of a force/torque sensor. *Right:* oscilloscope to measure the PWM pulse width of the commands. Relating these data allows to determine the propeller parameters.

[6] `https://www.kontronik.com/en.html`

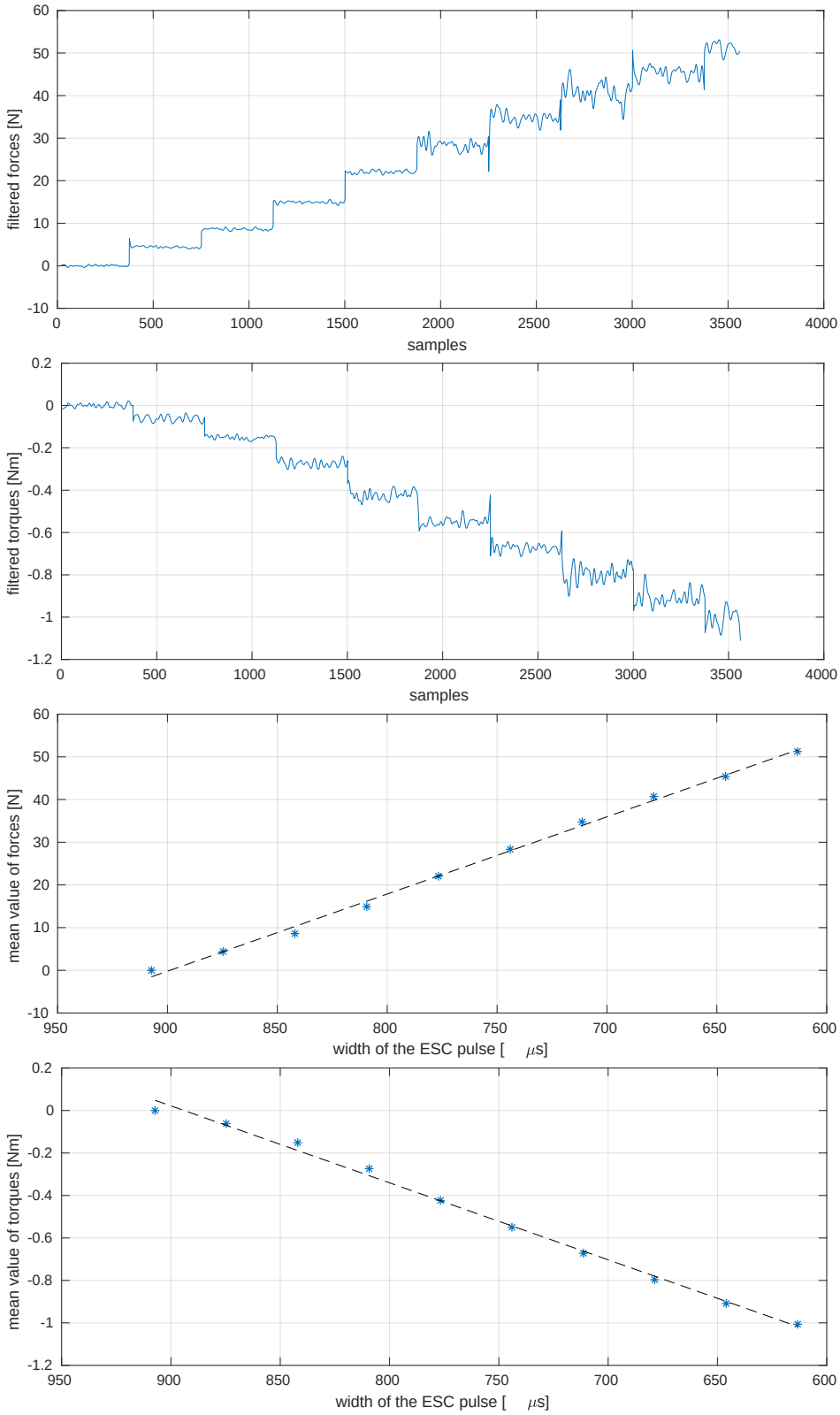


Figure 6.19 – Plots related to the preliminary estimation of the actuator parameters. *Top plots:* measured forces and torques (both low-pass filtered) for different constant values of PWM commands. The signals are represented over discrete time (PWM samples). *Bottom plots:* linear interpolation of mean actuators forces and torques for each constant value of PWM commands. The values are represented over the width of the ESC pulse.

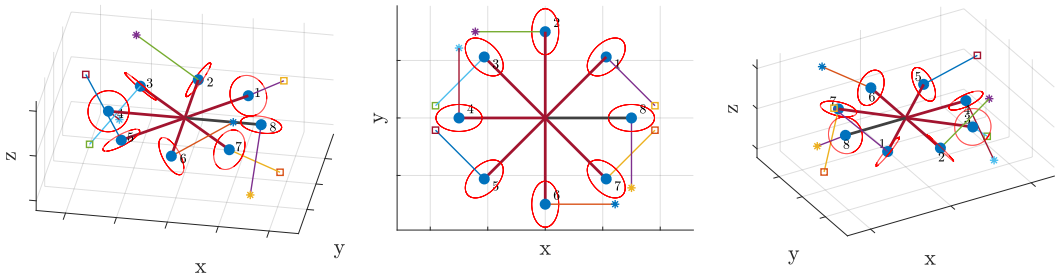


Figure 6.20 – Optimized omni-directional design for the propulsion units of the SAM. The blue spheres represent the placement of the propulsion unit motors. The colored lines point the thrust direction of each motor. The star symbol denotes the counterclockwise propeller, and the square indicates clockwise propeller.

The result of the optimization, w.r.t. the $\alpha_i,\ i = 1, \ldots, 8$ is:

$$\alpha_i^\star = 53.1, -54.1, -126.9, 125.9, 53.1, -54.1, -126.9, 125.9, \quad [\text{deg}]. \tag{6.15}$$

The obtained design for the propulsion units is represented in Fig. 6.20. With this configuration, the SAM is able to generate a set of independent forces and moments, thus allowing to decouple the control of the position and the orientation in a large extent. Fig. 6.21 displays the set of body-frame admissible forces with zero moment (*left*) and the set of admissible moments with zero force (*right*). The first set describes the set of forces that can be exerted in order to apply pure translations to the MRAV CoM, while the second one delineates the set of moments that can be applied in order to produce pure rotations. It is interesting to remark the high control authority around the yaw axis, which is sought in order to guarantee high moment manipulability for the fulfillment of different manipulation tasks (cf. Sec. 11.2) like, e.g., turning a valve or attaching a sensor by rotating it inside a thread. A photo of the real SAM prototype is portrayed in Fig. 6.22, while its physical parameters are condensed in Tab. 6.4.

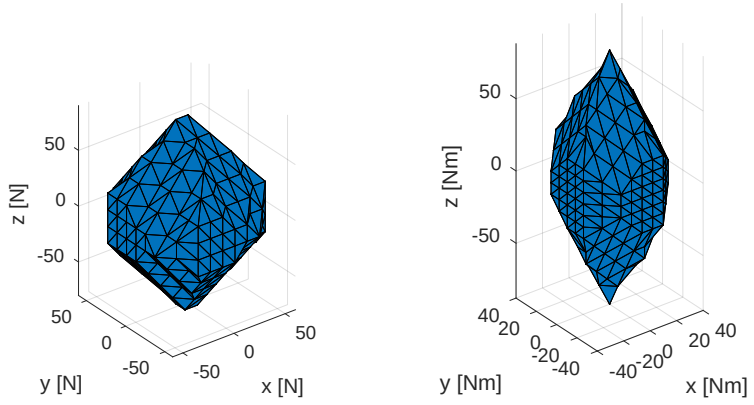


Figure 6.21 – Set of admissible forces with zero moment (*left*), used to produce a pure translation of the vehicle CoM, and set of admissible moments with zero force (*right*), used to generate a pure rotation of the body frame around the vehicle CoM.

Figure 6.22 – Photo of the real SAM prototype.

Table 6.4 – Physical parameters of the SAM prototype.

SAM		
Parameter	Value	Unit
m	45	Kg
$\mathbf{J}(:,1)$	4.275	$\mathrm{Kg\,m^2}$
$\mathbf{J}(:,2)$	4.275	$\mathrm{Kg\,m^2}$
$\mathbf{J}(:,3)$	8.438	$\mathrm{Kg\,m^2}$
c_i	–	[ ]
c_f^τ	–	m
c_f	–	$\mathrm{N/Hz^2}$
$\mathbf{R}_{A_i}^B$	$\mathbf{R}_z((i-1)\frac{\pi}{4}))\mathbf{R}_x(\alpha_i)\mathbf{R}_y(\beta)$	[ ]
$\mathbf{p}_{A_i}^B$	$\mathbf{R}_z((i-1)\frac{\pi}{4}))[\ell\ 0\ 0]^\top$	[ ]
$\alpha_{1-4} = \alpha_{5-8}$	$53.1, -54.2, -126.9, 125.9$	deg
β	0	deg
ℓ	0.75	m

Chapter 7

Validation of the proposed control methodologies

"I think that in the discussion of natural problems we ought to begin not with the Scriptures, but with experiments, and demonstrations."
Galileo Galilei

This chapter is devoted to the validation of the control methodologies presented in Chapt. 5, which in turn exploit the Multi-Rotor Aerial Vehicle (MRAV) dynamical representations outlined in Chapt. 4. In order to validate the two controllers, we conduced different experiments with the real prototypes described in Chapt. 6. In particular, to improve the trajectory tracking in the experiments, we performed an additional identification of the allocation matrix parameters, instead of using the nominal ones, given by (3.14). This is done with the goal of mitigating the deviations of the model from the real system. A brief outline of the state-of-the-art identification procedure and the results obtained with one particular MRAV are presented for completeness in App. A. Moreover, we present additional realistic simulations performed with MRAV designs which are of particular interest like, e.g., a Multi-Directional Thrust (MDT) hexarotor subject to a failure of one actuator. We will show that both controllers succeed in stabilizing the different aerial platforms and achieve good performance in the tracking of challenging reference trajectories. For the reader's ease, we outline the structure of the chapter in Fig. 7.1, presenting the MRAVs employed for each experimental/numerical validation.

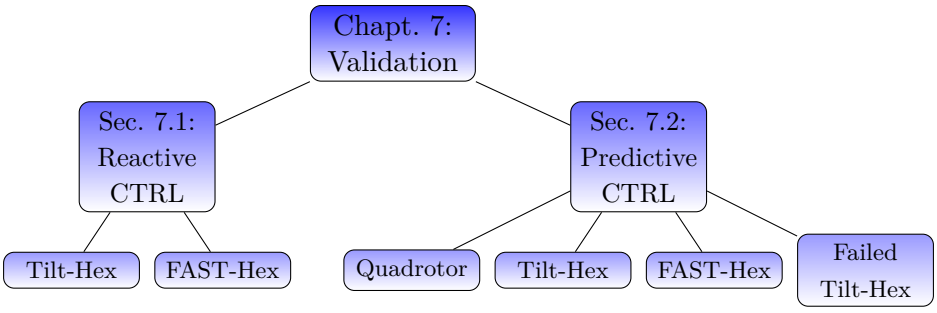


Figure 7.1 – Tree diagram of the exp./num. validation presented in this chapter.

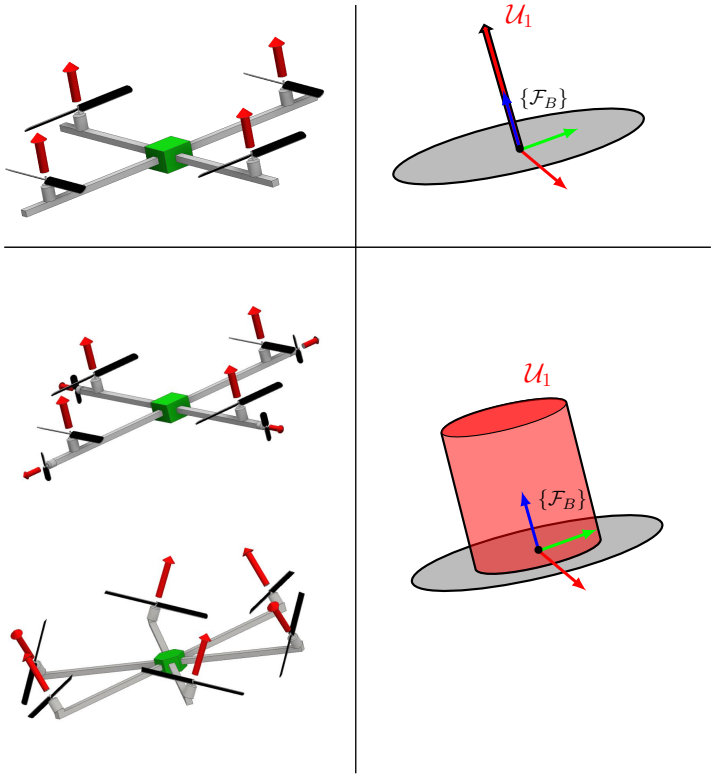


Figure 7.2 – Examples of MDT MRAV designs (*left*) and possible Laterally-Bounded input Force (LBF) approximations (*right*). In particular, the set of body forces obtained with a zero net torque are represented.

Validation of the static feedback controller

In this section, we first outline the practical implementations of the general optimization problem (5.4) in the case where the set $\mathcal{U}_1$ defined in (4.2) can be approximated with two particular sets. These specific approximations are the ones used then to perform the experiments that will follow. Nevertheless, many other different approximations could be employed, thanks to the generality of the approach.

Computation of the feasible rotation for two important cases

The control method proposed in Sec. 5.1 is kept on purpose general regarding two main features: the choice of $\mathcal{U}_{xy}$ in (4.1) and the choice of the cost function c_R. The former allows the method to be used for a large set of aerial vehicles with different actuation capabilities. The latter allows the user to customize the definition of similarity between two orientations in order to comply with the particular task at hand. In this section, we illustrate how these general features are particularized for a couple of specific meaningful cases, depicted in Fig. 7.2.

Case 1: (Uni-Directional Thrust (UDT) constraint). When

$$\mathcal{U}_{xy} = \{\mathbf{0}\}, \tag{7.1}$$

model (4.1)-(4.4) becomes the UDT quadrotor model considered in [Mistler–2001; Hamel–2002; Bouabdallah–2005; Lee–2010]. In this case, as already mentioned, the resulting under-actuation constrain the vehicle to re-orient its body frame in order to steer the control input force $\mathbf{f}_r$ as indicated by the control law. In particular, it must hold:

$$\mathbf{b}_3^d = \mathbf{R}_d \mathbf{e}_3 = \frac{\mathbf{f}_r}{\|\mathbf{f}_r\|}, \tag{7.2}$$

where $\mathbf{b}_i^d = \mathbf{R}_d \mathbf{e}_i, \;\; i = 1, \ldots, 3$ denote the three unit vectors of the body frame. In this case, the MRAV the rotational dynamics is unavoidably coupled with the translational one. Nevertheless, the platform maintains the control authority on the component of the rotation around $\mathbf{b}_3^d$, i.e., the *yaw*. This corresponds to the vehicle heading direction in the plane normal $\mathbf{b}_3^d$ to and it is determined by the direction of $\mathbf{b}_1^d$. In view of this, in order to compute $\mathbf{R}_d$ from $\mathbf{b}_{3d}$ we suggest to employ the following formula, as done in [Lee–2010]:

$$\mathbf{R}_d = \big[\underbrace{(\mathbf{b}_{3d} \times \mathbf{b}_{1r}) \times \mathbf{b}_{3d}}_{\mathbf{b}_{1d}} \quad \underbrace{\mathbf{b}_{3d} \times \mathbf{b}_{1r}}_{\mathbf{b}_{2d}} \quad \mathbf{b}_{3d} \big]. \tag{7.3}$$

Therefore, in the case of UDT MRAV, the wrench constraint is given by (7.1) and the feasible rotation matrix can be computed using (7.3). The most representative design of this class is the very well-known quadrotor platform [Hamel–2002], represented at the top left of Fig. 7.2

Case 2: (Cylindrical LBF constraint). When

$$\mathcal{U}_{xy} = \{[u_1 \;\; u_2]^\top \in \mathbb{R}^2 \mid u_1^2 + u_2^2 \leq r_{xy}^2\}, \tag{7.4}$$

model (4.1)-(4.4) approximates the case of a MRAV whose set of allowable forces, defined in (4.2), can be approximated by a pseudo-cylindrical shape. Examples of such designs are given in the literature in the case of a MDT hexarotor [Voyles–2012] and of a MDT octorotor [Romero–2007]. A sketch of both designs is shown in the bottom left of Fig. 7.2.
In this case, the following choice of cost function c_R can be used:

$$c_R(\mathbf{R}_r, \mathbf{R}') = 1 - \mathbf{b}_{3r}^\top \mathbf{b}_3', \tag{7.5}$$

where $\mathbf{R}_r = [\mathbf{b}_{1r}\, \mathbf{b}_{2r}\, \mathbf{b}_{3r}]$ and $\mathbf{R}' = [\mathbf{b}_1'\, \mathbf{b}_2'\, \mathbf{b}_3']$. The cost function c_R in (7.5) is minimized whenever $\mathbf{b}_{3r} = \mathbf{b}_3'$ and maximized whenever $\mathbf{b}_{3r} = -\mathbf{b}_3'$. In the following, we show how it is possible to efficiently compute an $\mathbf{R}_d$ that belongs to $\overline{\mathcal{R}}(\mathbf{f}_r, \mathbf{R}_r)$ and is also equal to $\mathbf{R}_r$ if $\mathbf{R}_r \in \overline{\mathcal{R}}(\mathbf{f}_r, \mathbf{R}_r)$. These are in fact the requirements needed for $\mathbf{R}_d$ in order for Theorem 5.1.3 and Corollary 5.1.4 to be valid.

Let us start by instantiating $\mathcal{R}(\mathbf{f}_r)$ for this particular case. From simple geometrical considerations on the cylindrical shape of the set $\mathcal{U}_1$ it is easy to see that

$$\mathcal{R}(\mathbf{f}_r) = \left\{ \mathbf{R}' \in SO(3) \mid \mathbf{f}_r^\top \mathbf{b}'_3 \geq \sqrt{\|\mathbf{f}_r\|^2 - r_{xy}^2} \right\}, \tag{7.6}$$

which states that the vector $\mathbf{f}_r$ must lie within the cylinder of radius r_{xy} generated about the axis $\mathbf{b}'_3$.
Using (7.5) and (7.6) we can rewrite the minimization problem in terms of the only variable $\mathbf{b}'_3$, instead of the whole matrix $\mathbf{R}'$, as

$$\min_{\substack{\mathbf{f}_r^\top \mathbf{b}'_3 \geq \sqrt{\|\mathbf{f}_r\|^2 - r_{xy}^2}, \\ \|\mathbf{b}'_3\|^2 = 1}} -\mathbf{b}_{3r}^\top \mathbf{b}'_3, \tag{7.7}$$

where r_{xy}^2, $\mathbf{f}_r^\top$ and $\mathbf{b}_{3r}^\top$ are the givens of the problem.
In the case where $\mathbf{f}_r^\top \mathbf{b}_{3r} \geq \sqrt{\|\mathbf{f}_r\|^2 - r_{xy}^2}$ then $\mathbf{b}'_3 = \mathbf{b}_{3r}$ is the solution to (7.7). Otherwise, let us write $\mathbf{b}'_3$ as the sum of two components $\mathbf{b}'_3 = \mathbf{b}'_{3\parallel} + \mathbf{b}'_{3\perp}$, where $\mathbf{b}'_{3\parallel}$ is parallel to the plane spanned by $\mathbf{b}_{3r}$ and $\mathbf{f}_r$, while $\mathbf{b}'_{3\perp}$ is perpendicular to it, i.e., is parallel to $\mathbf{b}_{3r} \times \mathbf{f}_r$. It is easy to see that the cost function in (7.7), i.e., $\mathbf{b}_{3r}^\top \mathbf{b}'_3$, is not affected by $\mathbf{b}'_{3\perp}$, in fact $\mathbf{b}_{3r}^\top \mathbf{b}'_3 = \mathbf{b}_{3r}^\top \mathbf{b}'_{3\perp} + \mathbf{b}_{3r}^\top \mathbf{b}'_{3\parallel} = 0 + \mathbf{b}_{3r}^\top \mathbf{b}'_{3\parallel}$. The vector $\mathbf{b}'_{3\parallel}$ can be written using the Rodrigues' rotation formula as $\mathbf{b}'_{3\parallel}(\theta) = \mathbf{b}_{3r}\cos\theta + (\mathbf{k} \times \mathbf{b}_{3r})\sin\theta + \mathbf{k}(\mathbf{k} \cdot \mathbf{b}_{3r})(1 - \cos\theta)$, where $\mathbf{k} = \frac{\mathbf{b}_{3r} \times \mathbf{f}_r}{\|\mathbf{b}_{3r} \times \mathbf{f}_r\|}$ and θ is the rotation angle that univocally defines $\mathbf{b}'_{3\parallel}$. Noting that the constraint $\|\mathbf{b}'_3\|^2 = 1$ is automatically verified by $\mathbf{b}'_{3\parallel}(\theta)$ for any θ, we further simplify (7.7) in terms of the only scalar variable θ as

$$\min_{\mathbf{f}_r^\top \, \mathbf{b}'_{3\parallel}(\theta) \geq \sqrt{\|\mathbf{f}_r\|^2 - r_{xy}^2}} -\mathbf{b}_{3r}^\top \, \mathbf{b}'_{3\parallel}(\theta). \tag{7.8}$$

Given that the feasible set is connected, the minimization problem (7.8) can be efficiently solved numerically using a bisection method (as the one shown in Algorithm 1). In order to finally compute $\mathbf{R}_d$, we can plug $\mathbf{b}_3^d$ obtained by the solution of (7.8) in the usual formula (7.3). Ultimately, we note that if $\mathbf{R}_r \in \overline{\mathcal{R}}(\mathbf{f}_r, \mathbf{R}_r)$ then $\mathbf{f}_r^\top \mathbf{b}_{3r} \geq \sqrt{\|\mathbf{f}_r\|^2 - r_{xy}^2}$ which, as we previously said, implies that $\mathbf{b}_{3d} = \mathbf{b}_{3r}$. Then, it results $\mathbf{R}_d = \mathbf{R}_r$, as asked by the second requirement on the computation of $\mathbf{R}_d$.

Remark (Approximation of $\mathcal{U}_1$). It is worthwhile to underline that the approximated sets $\mathcal{U}_1$ depicted in Fig. 7.2 and in Fig. 7.3 correspond to the particular set of body forces which generate, at the same time, a zero net body torque. This corresponds to a subset of the whole set of body forces that can be generated without fixing the corresponding torque. In general, the shape of the feasible force set is affected by the particular value of the required body torque. The motivation towards the choice of this specific sub-set is the fact that the required average torque for a platform to hover is typically zero. If this is not the case, the set $\mathcal{U}_1$ can be recomputed by keeping into account the particular average torque that should be produced.

Algorithm 1: *Bisection used to solve problem* (7.8)

Data: n (number of iterations $\propto$ solution accuracy)
Data: $\mathbf{b}_{3r}$, $\mathbf{f}_r$, and r_{xy} (givens of the problem)
$\mathbf{k} \leftarrow \frac{\mathbf{b}_{3r} \times \mathbf{f}_r}{\|\mathbf{b}_{3r}\| \|\mathbf{f}_r\|}$, $\theta_{max} \leftarrow \arcsin(\|\mathbf{k}\|)$, $\theta \leftarrow \theta_{max}/2$
for $i = 1$ *to* n **do**
  if $\mathbf{f}_r^\top \mathbf{b}'_{3\|}(\theta) \geq \sqrt{\|\mathbf{f}_r\|^2 - r_{xy}^2}$ **then** $\theta \leftarrow \theta - \frac{1}{2}\frac{\theta_{max}}{2^i}$
  else $\theta \leftarrow \theta + \frac{1}{2}\frac{\theta_{max}}{2^i}$
return θ

To conclude this part, we would like to stress that, in case of different sets $\mathcal{U}_{xy}$ and different cost functions c_R, either similar efficient approaches can be used or the second method (cylindrical approximation with a certain radius r_{xy}) presented here can be used as a conservative approximation.

Implementation details

The goal of this part is to clearly state the implementation details that have been chosen for the experimental validation that will be shown in the remaining part of the section. As far as the explicit LBF constraint definition for the aerial platforms, we chose to employ the one given by *case 2*, i.e., , the approximation with a cylindrical set. Consequently, Algorithm 1 will be used, together with (7.3), in order to compute the desired feasible rotation $\mathbf{R}_d$ for the pose controller. Despite its simplicity, the cylindrical set is a good approximation for a wide set of real MRAVs, as suggested by the bottom part of Fig. 7.2. This can be assessed also in Fig. 7.3, which depicts the set of feasible hovering forces, i.e., the ones obtained with zero net body moment, for the Tilt-Hex robot presented in Chapt. 6. It should be appreciated that this represents, in general, a conservative approximation of the real hovering force set. In particular, the cylinder is built around the typical wrench working point of the platform, assumed to be $\boldsymbol{w}_B^{B\star} = [0\ 0\ mg\ 0\ 0\ 0]^\top$.

In the real implementation of (5.8) we have omitted, on purpose, all the terms in which ω_d and $\dot{\omega}_d$ appear in the control (or equivalently, we have considered as if, virtually, $\dot{\omega}_d = \omega_d = 0$). We have done so in order to assess the level of practical importance of those terms and how the controller is in practice robust to the non-correct evaluation of these terms. As it will be shown in the following, it turned out that the tracking performance is still very good even without those terms, therefore one can still obtain acceptable result by neglecting them. However, as requested from the theory and also seen in preliminary simulations, a perfect tracking is guaranteed only if those terms are considered.

Regarding the controller algorithm, it has been implemented in Matlab/Simulink® and runs at 500 Hz on a stationary base PC connected to the aerial vehicles through a serial cable. As the computational effort of the controller is very low (considerably below 1 ms per control loop) it could be ported easily to an on-board system.

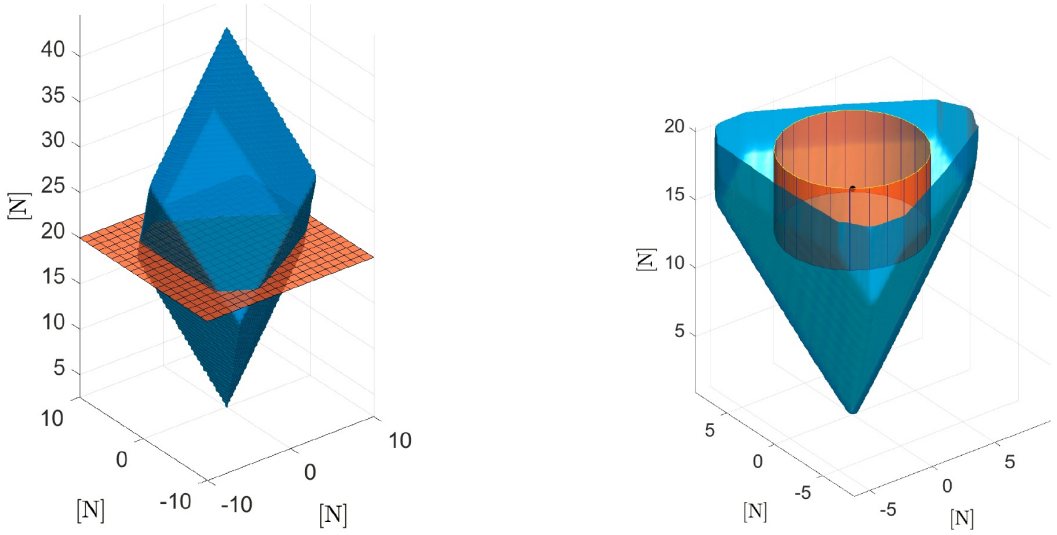


Figure 7.3 – *Left:* The blue volume encloses the set of feasible forces at hovering, obeying the constraints of minimal and maximum rotor spinning velocity for the Tilt-Hex. The red plane visualizes the cut of the sectional view of the plot on the right. *Right:* lower part of the cut of the left figure. The red cylinder visualizes the volume of the imposed cylindric force constraint, whose radius can be shrunk at will to have $r_{xy} \in [0, r_{xy_{max}}]$. Notice that the cylinder is fully inside the volume of feasible forces, thus it is a *conservative* approximation. The black dot in the center visualizes the force needed to hover horizontally, i.e., the nominal working point.

Based on our experience with a similar porting, we expect the performances of an onboard implementation to be much better than the Matlab/Simulink® implementation, thanks to the possibility of reaching a faster control frequency (larger than 1 kHz) and almost real-time capabilities (latency below 1 ms). The presence of the cable is clearly disturbing for the tracking purpose since it produces an unmodeled wrench. Furthermore, the oscillations of the cable are rather erratic and therefore hard to model and compensate. Therefore, all considered, the experiments proposed in in the following represent a worst case scenario from this point of view, in the sense that an onboard implementation should perform better.

In order to design the software architecture, we relied on the GenoM3[1] abstraction level, which allows to encapsulate software functions inside independent components. More in detail, it has been used as a wrapper for the robot low-level controller and the sensors. This allows to obtain high flexibility in the development and in the use of the components. With reference to our architecture, the software in MATLAB/Simulink® communicates with the GenoM3 modules using the Robot Operating System (ROS) middleware. Since MATLAB/Simulink® is not meant for a hard real-time execution, the hardware is commanded via the GenoM3 components, which essentially behave like drivers.

[1] `https://git.openrobots.org/projects/genom3/wiki`

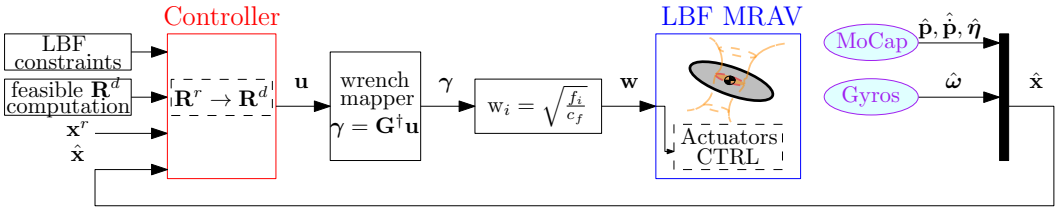


Figure 7.4 – Block diagram of the experimental setup architecture. The main components are highlighted with different colors. The symbol $\boldsymbol{\eta}$ represents, in this case, the vector containing the entries of the rotation matrix $\mathbf{R}$.

Experimental setup

The experimental setup architecture, whose block diagram is portrayed in Fig. 7.4, can be conceptually divided into three main components: the LBF controller for the pose tracking, the physical aerial robots to be controlled, and the sensors, used to retrieve the information about the MRAVs state that is employed as feedback in the closed-loop control strategy. Each block exchanges information with the others thanks to a properly-designed software architecture.

The control algorithm runs on a ground-station PC equipped with an Intel® 2.60GHz Core™ i7-6700HQ CPU (x8) and 32 GB RAM which runs the Linux Ubuntu 16.04 LTS operating system. As it can be observed from Fig. 7.4, the control input $\mathbf{u}$, which provides the needed body wrench to track the trajectory, is mapped to a reference control force for the actuators and then converted into a rotor velocity command $\mathbf{w}$, thanks to the inversion of the force generation model (3.17). The resulting velocity set-points are finally transferred to the module of the low-level controllers on-board the aerial platform, by means of a serial cable.

The real MRAVs used for this experimental validation are the Tilt-Hex and the FAST-Hex platforms described in Chapt. 6. The vehicles are endowed with an electronic board with six independent Electronic Speed Controller (ESCs) for the control of the rotor velocities. In particular, each of them implements the closed-loop control algorithm [Franchi–2017] at a variable frequency (e.g., when the propeller rotation speed is 70 Hz the control frequency is 3.29 kHz).

The main sensors integrated in our experimental framework are the onboard gyroscope, used to measure the rotational velocity of the vehicle around each of the body frame axis, and the Motion Capture System (MoCap) system, which provides the information regarding the robot position and orientation with respect to the inertial reference frame, whose origin is fixed in a particular point of the robots workspace. The platform linear velocity is numerically computed online from the position measurements, using multi-sample least squares model fitting.

Experimental validation

We present here the experimental results obtained with the Tilt-Hex MRAV. For the more recent (and yet unpublished) results related to the FAST-Hex robot, we invite the reader to refer to the next subsection. We conduced three experimental

validation campaigns, in which the tasks has been always to follow a given reference $\mathbf{q}_r(t) = (\mathbf{p}_r(t) = [p_{r_x}(t)\, p_{r_y}(t)\, p_{r_z}(t)]^\top, \mathbf{R}_r(t))$. In *Tilt-Hex experimental batch 1*, the value of r_{xy} in (7.4) and of $\mathbf{R}_r$ is kept constant and three experiments are performed: Exp. 1.1, Exp. 1.2 and Exp. 1.3, which are detailed in the following. In *Tilt-Hex experiment 2*, $\mathbf{R}_r$ varies over time. Finally, in *Tilt-Hex experiment 3*, r_{xy} is artificially modified over time.
For the reader's visual convenience the rotation matrices used internally by the controller, have been converted in the plots to *yaw-pitch-roll* angles, with the convention $\mathbf{R} = \mathbf{R_z}(\psi)\mathbf{R_y}(\theta)\mathbf{R_x}(\phi)$, where $\mathbf{R}_\bullet(\alpha)$ denotes a rotation around one of the main body frame axes $\{\mathbf{x}, \mathbf{y}, \mathbf{z}\}_B$ of an angle α. In the plots where data are very noisy a filtered version (darker color) is presented together with the original data (lighter color in background). We shall refer to the plots of a figure directly by the name of plotted signal, which is easily understandable from the relative legend.

The interested reader is referred to the multimedia attachment of this thesis to fully enjoy the videos all the experiments. The links to the videos are given in the description of each experimental set. In particular, for all the videos presenting the research on aerial robotics at LAAS–CNRS, the reader is referred to the following link `https://www.youtube.com/channel/UCFoSjtoDdbt4G4mL7p1jciA`. Furthermore, the experimental data related to the next three experimental batches (with suitable scripts to plot them) are provided for download at the following link `http://homepages.laas.fr/afranchi/files/2017/dataset1.zip`.

Tilt-Hex experimental batch 1

In this batch we impose $p_{r_y}(t) = 0\,\mathrm{m}$ and $p_{r_z}(t) = 1\,\mathrm{m}$, while $p_{r_x}(t)$ oscillates sinusoidally between $-1.2\,\mathrm{m}$ and $1.2\,\mathrm{m}$ with time-varying frequency, chosen such that the envelope of $\ddot{p}_{r_x}(t)$ is first quasi-linearly increasing from $0\,\mathrm{m\,s^{-2}}$ up to $5.9\,\mathrm{m\,s^{-2}}$ and then quasi-linearly decreasing down to $0\,\mathrm{m\,s^{-2}}$ – see the corresponding signals in Fig. 7.5. On the other side we set $\mathbf{R}_r(t) = \mathbf{I}_{3\times3}$.
In Exp. 1.1 a value of $r_{xy} = 3\,\mathrm{N}$ has been selected, which fits well inside the actual maximum lateral force of the Tilt-Hex given its mass of $m = 1.8\,\mathrm{kg}$. This means that in the parts of the trajectory in which $|\ddot{p}_{r_x}| > \frac{3\,\mathrm{N}}{1.8\,\mathrm{kg}} = 1.66\,\mathrm{m\,s^{-2}}$ we expect the controller to let the platform deviate from $\mathbf{R}_r$ in order to track the high lateral acceleration. On the other side we expect a good independent tracking of position and orientation when $\ddot{p}_{r_x} \leq 1.66\,\mathrm{m\,s^{-2}}$. In fact, Exp.1.1 is meant to illustrate the canonical behavior of proposed controller when controlling a LBF platform with MDT capabilities.
In Exp. 1.2 we tell the controller that $r_{xy} = 0\,\mathrm{N}$ (corresponding to the case of a UDT MRAV). Therefore we expect the controller to let the platform deviate almost always from $\mathbf{R}_r$. This experiment is meant to show that the proposed controller can handle the (classic) under-actuated case, thus not requiring the user to switch between different controllers.
In Exp. 1.3 we replace our controller with the state-of-the-art controller presented in [Rajappa–2015], a controller that does not take into account the input satura-

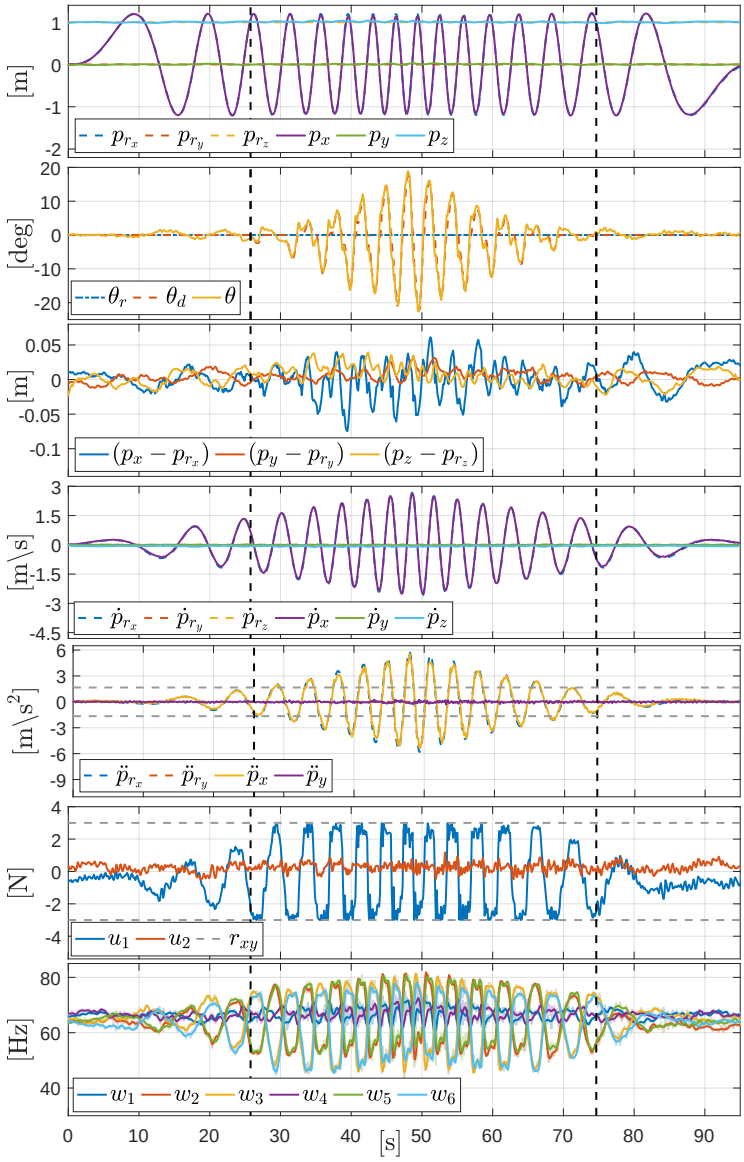


Figure 7.5 – Exp. 1.1: Desired position: sinusoidal motion along the $\mathbf{x}_W$ axis with constant amplitude and triangular (first increasing then decreasing) frequency. Desired orientation: constantly horizontal. Lateral force bound: constant $r_{xy} = 3\,\text{N}$. *From top to bottom:* position and orientation tracking, position error, linear velocity and acceleration tracking, lateral force control input, and measured rotor velocities.

tions. This experiment is meant to show how the proposed controller outperforms other state of the art controllers in terms of robustness and stability.
The video of these three experiments is available in [video01–2017].

Exp. 1.1

Fig. 7.5 visualizes the main results of Exp 1.1. There are three clearly distinct temporal phases separated by the vertical dashed lines in the plots and defined by $t \in T_1 = [0\,\text{s}, 25.8\,\text{s}]$, $t \in T_2 = [25.8\,\text{s}, 74\,\text{s}]$, and $t \in T_3 = [74\,\text{s}, 95\,\text{s}]$, respectively.

Figure 7.6 – The Tilt-Hex performing Exp. 1.1 at different time instances: *left:* $t = 15.4\,\mathrm{s}$; *center:* $t = 37.7\,\mathrm{s}$; *right:* $t = 47.9\,\mathrm{s}$. Although the reference orientation is constant and horizontal, the Tilt-Hex adapts it to allow the prioritized tracking of the reference position.

In the first and third phases $\mathbf{q}_r(t)$ is always feasible. In fact $|\ddot{p}_{r_x}|$ is always below $1.66\,\mathrm{m\,s^{-2}}$, see discussion above. In the second phase, instead, $\mathbf{q}_r(t)$ is not always feasible and in the middle of the second phase, in the neighborhood of $t = 49\,\mathrm{s}$, $\mathbf{q}_r(t)$ is mostly unfeasible, since $|\ddot{p}_{r_x}|$ has peaks of $5.9\,\mathrm{m\,s^{-2}}$ (3.5 times the maximum lateral acceleration attainable while keeping a horizontal orientation). Accordingly to what expected, in the 'feasible' phases (T_1 and T_3) both the orientation and position tracking errors w.r.t. $\mathbf{q}_r(t)$, are relatively low. In particular we have $\|\mathbf{p}(t) - \mathbf{p}_r(t)\| < 0.02\,\mathrm{m}$ and $|\theta(t) - \theta_r(t)| < 1.7°$ and zero average for the two errors. In the 'unfeasible' phase (T_2), the position tracking is still good ($\|\mathbf{p}(t) - \mathbf{p}_r(t)\| < 0.06\,\mathrm{m}$) while $\mathbf{R}_d(t)$ sensibly deviates from $\mathbf{R}_r(t)$ with a peak overshooting 20° for $|\theta_d(t) - \theta_r(t)|$. In fact, tilting is the only way by which the platform can track the desired position, given the lateral force bounds. It is interesting to note that not only the reference position is well tracked along the whole experiment, but also the translational velocity and acceleration are. One can also appreciate how the controller keeps always the lateral force u_1,u_2 within the requested bounds and at the same time touches and stays on the bounds several times for several seconds. This is a clear index that the controller exploits at best the platform capabilities. Comparing the plots one can notice also how when the lateral force is saturated, the controller exploits the platform tilting in order to compensate for the partial loss of control authority and attain the force required to produce the needed acceleration. Fig. 7.6 shows a time-lapse of the Tilt-Hex performing Exp. 1.1 in three different time instants. Furthermore, we encourage the reader to watch the multimedia attachment showing this and the other experiments. Finally, for the sake of completeness, we present also the actual six rotor spinning velocities.

Exp. 1.2

To test the behavior of the controller with UDT MRAV, in Exp. 1.2 we set $r_{xy} = 0\,\mathrm{N}$ and let the controller track the same trajectory of Exp 1.1. In this way the Tilt-Hex should behave like an under-actuated multi-rotor. Figure 7.7 shows the main plots, while the plots that are similar to the ones in Fig. 7.5 are omitted. Contrarily to Exp. 1.1, in Exp. 1.2, phases T_1 and T_2 do not exists. The whole experiment is a long unfeasible phase due to the constraint $r_{xy} = 0\,\mathrm{N}$, which makes impossible, at any time, to track the constant $\mathbf{R}_r(t) = \mathbf{I}_{3\times3}$ while following the sinusoidal reference position trajectory. The orientation tracking of Exp. 1.2 is compared to the one

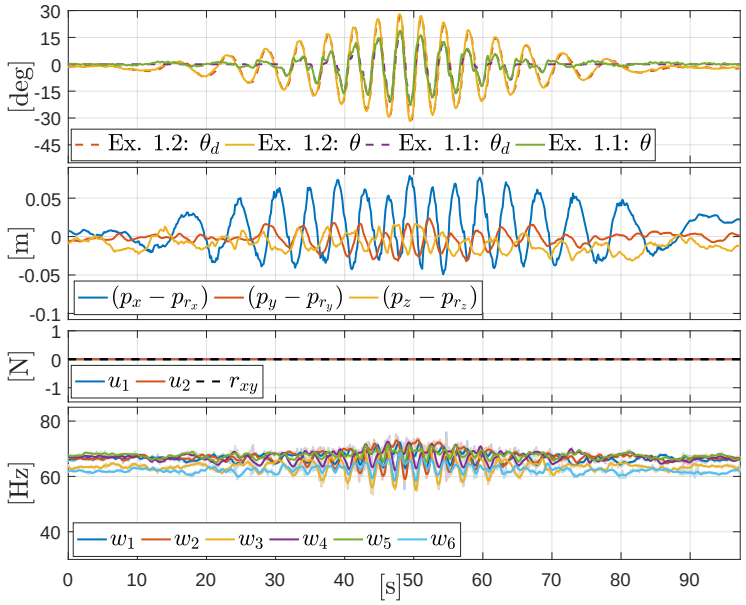


Figure 7.7 – Exp. 1.2: Same desired trajectory as in Exp.1.1 but with $r_{xy} = 0$. *From top to bottom:* orientation and position error, lateral force control input, and measured rotor velocities.

Exp. 1.1. In the period of time T_1 defined for Exp. 1.1, $|\theta(t) - \theta_r(t)|$ in Exp. 1.2, reaches 11°, i.e., 6.5 times the peak of Exp. 1.1 in the same period. In the period of time T_2, $|\theta(t) - \theta_r(t)|$ in Exp. 1.2, reaches 31° i.e., about 1.5 times the peak of Exp. 1.1 in the same period. Regarding the translational behavior, the peak of the position tracking error is about 3 times larger (in the period T_1) and 1.4 times larger (in the period T_2), when compared to the error peak of Exp. 1.1 in the same periods. This is due to the fact that full actuation helps in minimizing the position tracking error too. Furthermore, we can see that that the inputs u_1 and u_2 remain zero as expected during the full trajectory tracking, as required. Finally, for completeness, we present also in this case the six rotor spinning velocities.

Exp. 1.3

In order to compare with the state-of-the-art methods such as [Rajappa–2015], in Exp. 1.3 we tested the controller with a saturated rotor spinning velocity with the minimum and maximum values in Exp. 1.1 ($43\,\text{Hz} \leq \text{w}_i \leq 83\,\text{Hz}$, $i \in [1 \dots 6]$). The results are depicted in Fig. 7.8. The platform tracks well the reference trajectory till the input reaches its limit ($t = 34\,\text{s}$) (see the dashed horizontal lines in the last plot). After that time, the controller asks for spinning velocities outside the limits, which are hardly saturated. The trajectory tracking performance decreases rapidly, until the system becomes completely unstable diverging from the reference position ($||\mathbf{p} - \mathbf{p}_r|| > 0.5\,\text{m}$) and reference velocity ($||\dot{\mathbf{p}} - \dot{\mathbf{p}}_r|| > 1.2\,\text{m}\,\text{s}^{-1}$) such that we had to abort the experiment. This experiment clearly shows how the proposed controller outperforms a state-of-the-art controller in terms of performances and, most important of all, stability and safety.

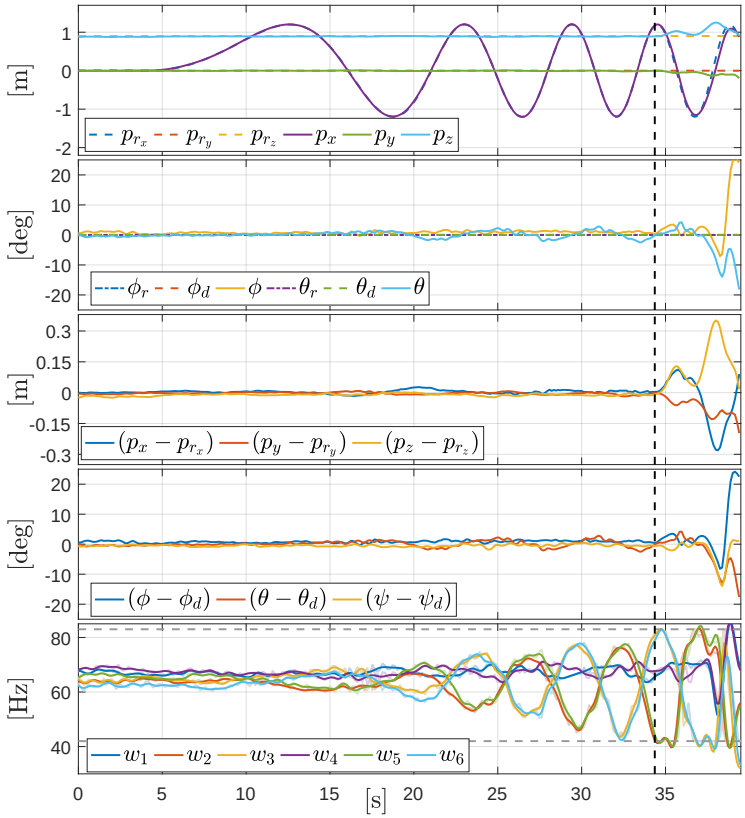


Figure 7.8 – Exp. 1.3: Same desired trajectory as in Exp.1.1 but with saturated rotor spinning velocity $\mathbf{w}$ (saturation indicated by dashed grey lines in plot five). *From top to bottom:* position and orientation tracking, position and orientation error, and measured rotor velocities. The experiment is automatically stopped after about 39 s because the system becomes visibly unstable.

Tilt-Hex experiment 2

To present the full capabilities of the full pose controller on LBF vehicles, in Exp. 2 we set $\mathbf{p}_r(t)$ as in Exp. 1.1, but we additionally ask the platform to follow a $\mathbf{R}_r(t)$ generated applying to $\mathbf{I}_{3\times3}$ a sinusoidal rotation about the $\mathbf{y}_W$ axis (with an amplitude of 10°). This rotational motion is particularly chosen such that the orientation of the Tilt-Hex is in opposition of phase with respect to the orientation that an under-actuated vehicle would need in order to track $\mathbf{p}_r(t)$ (i.e., the top part of the platform facing outwards at the two ends of the position trajectory, while for, e.g., a quadrotor the top would face always toward the center of the position trajectory). Also in this case, see Fig. 7.9, the reference-to-actual position error and the desired-to-actual orientation error remain bounded and small. The maximum lateral thrust is reached sooner than Exp. 1.1 (at $t = 10\,\mathrm{s}$), due to the special inclination required. This results in an earlier adaptation of $\mathbf{R}_d$. As expected, at the time of highest accelerations ($45\,\mathrm{s} \leq t \leq 55\,\mathrm{s}$) θ_d is almost inverted with respect to θ_r.
To appreciate the quality of the trajectory tracking in this challenging case, the reader is referred to [video02–2017].

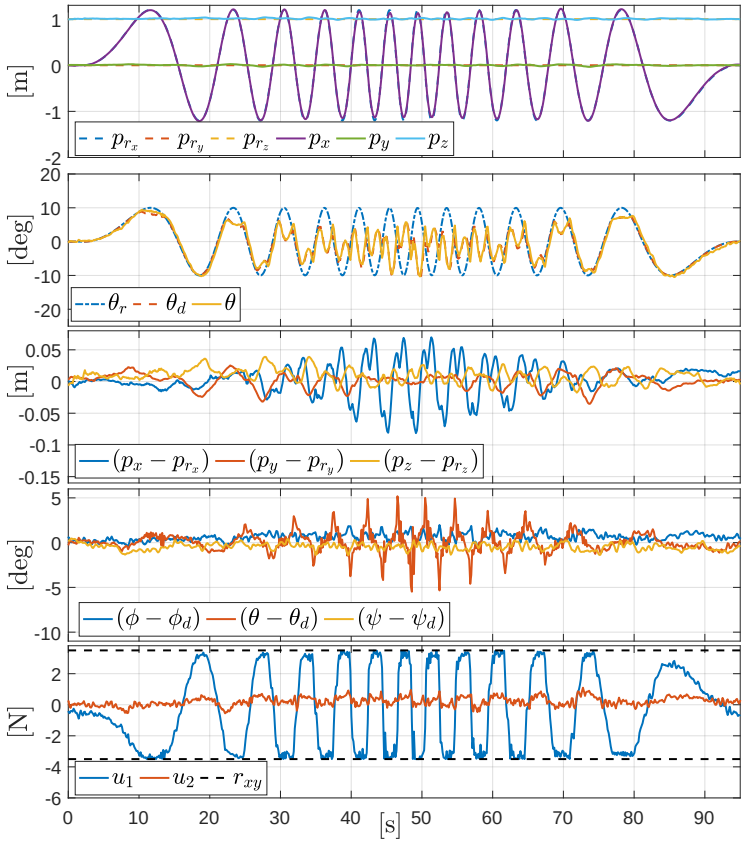


Figure 7.9 – Exp. 2: Desired position: as in Exp. 1.1. Desired orientation: sinusoidal rotation about the $\mathbf{y}_W$ axis in opposition of phase w.r.t. a hypothetical quadrotor following the desired position. Lateral force bound: constant $r_{xy} = 3\,\mathrm{N}$. *From top to bottom:* position and orientation tracking, position and orientation error, and measured rotor velocities.

Tilt-Hex experiment 3

The conclusive Exp. 3 has been designed to stress the fact that the presented controller can seamlessly work with UDT and properly MDT platforms and moreover with platforms that can actively change between these two configurations during flight, as the FAST-Hex, presented in Sec. 6.1. The plots of are reported in Fig. 7.10. The steady state $\mathbf{p}_r(t)$ consists out of two regular sinusoidal motions along the $\mathbf{x}_W$ and $\mathbf{y}_W$ axes with an amplitude of 1.3 m and 0.5 m, respectively, and constant frequencies, while $\mathbf{R}_r(t) = \mathbf{I}_{3\times3}$ is constant. The lateral force bound r_{xy} is virtually changed[2] over time, in particular, it is $r_{xy}(t) = 0\,\mathrm{N}$ for $t \in [0\,\mathrm{s}, 18\,\mathrm{s}]$, $r_{xy}(t) = 10\,\mathrm{N}$ for $t \in [38\,\mathrm{s}, 56\,\mathrm{s}]$, and it is linearly increasing from 0 N to 10 N for $t \in [18\,\mathrm{s}, 38\,\mathrm{s}]$. As it should be, the position tracking is always good. However, until $t = 34\,\mathrm{s}$, the system cannot track at the same time the reference position and the reference orientation. The orientation tracking gradually improves. At $t = 34\,\mathrm{s}$, r_{xy} is large

[2]The real dynamic capabilities of the Tilt-Hex do not change throughout the experiment. What factually changes is the information about the maximum lateral force at the control level.

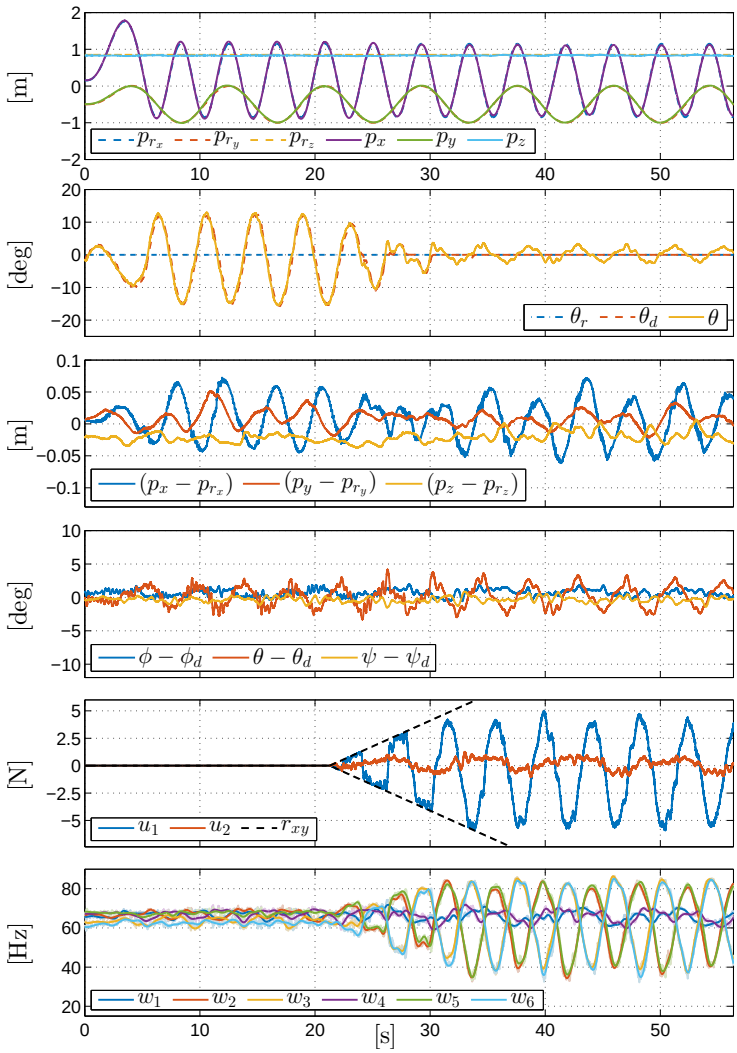


Figure 7.10 – Exp. 3: Desired position: composition of two sinusoidal motions along $\mathbf{x}_W$ and $\mathbf{y}_W$, with constant amplitudes and frequencies. Desired orientation: constant and horizontal. Lateral force bound: r_{xy} linearly increasing from 0 N to 10 N. *From top to bottom:* position and orientation tracking, position and orientation error, lateral force control input, and measured rotor velocities.

enough to track the reference orientation at any time. The behavior of r_{xy} is visualized with the dashed lines in the plot of u_1, u_2, which are always kept within the bounds. The fact that the lateral force bound changes over time does not deteriorate the behavior of the controller, which is instead able to cope with the time-varying constraint exploiting the platform capability always at its best. Finally, notice how the ranges of the propeller spinning velocities utilized by the controller naturally increases with the increase of $r_{xy}(t)$ at the benefit of completely tracking the full pose reference trajectory.
The video related to this experiment is available in [video03–2017].

Figure 7.11 – The FAST-Hex morphing its configuration from UDT to MDT by changing the synchronized alpha angle from 0 deg (*left*) to 30 deg (*right*).

Additional experimental validation

We present here the experimental results obtained with the FAST-Hex MRAV, already described in Chapt. 6 and depicted here in Fig. 7.11 during the morphing phase. The content of this subsection is recent and yet unpublished, and constitute a valuable addition to the experimental results presented so far in this section, which have been added to further demonstrate the generality and the effectiveness of the control algorithm presented in Sec. 5.1. We conduced three additional experimental validation campaigns, in which the tasks has been always to follow a given full-pose reference trajectory $\mathbf{q}_r(t) = (\mathbf{p}_r(t) = [p_{r_x}(t)\, p_{r_y}(t)\, p_{r_z}(t)]^\top, \mathbf{R}_r(t))$. Throughout all the three experiments, detailed in the following, the synchronized tilting angle α is modified over time, i.e., linearly increased from the minimum physical limit to the maximum one. In order to compute the evolution of the max lateral body force needed by the controller, we computed the value of r_{xy} for a discrete number of different α angles, and we interpolated them in the interval of interest with a curve.

FAST-Hex experiment 1

For this experiment we impose $p_{r_y}(t) = 0\,\text{m}$ and $p_{r_z}(t) = 1.1\,\text{m}$, while $p_{r_x}(t)$ oscillates sinusoidally with an amplitude of 1 m – see the corresponding signals in the top-left block of Fig. 7.12 – while the reference orientation is set to be constantly flat, i.e., $\mathbf{R}_r(t) = \mathbf{I}_{3\times3}$. The situation is similar to that of the last experiment performed with the Tilt-Hex, apart from the fact that in this context the lateral body force capability of the platform, epitomized by the parameters r_{xy}, physically (and not only virtually on the controller side) increases throughout the experiment from $r_{xy} = 0\,\text{N}$ to $r_{xy} = 6\,\text{N}$, see the bottom-left block of plots of Fig. 7.13.

Remark (Tikhonov regularization of allocation matrix pseudo-inversion). Since the structural properties of the FAST-Hex allocation matrix $\mathbf{G}(\alpha)$ change with the tilting angle α (i.e., with $\alpha = 0$ the allocation map becomes singular or for values of $\alpha \approx 0$ it is ill-conditioned) the computation of the wrench mapper is not trivial and the use of a simple inversion is not possible. The adopted solution here is the *Tikhonov regularisation*, which computes the pseudo-inverse in a closed form as

$$\boldsymbol{\gamma} = \mathbf{G}(\alpha)^{\ddagger}\boldsymbol{w}_B^B\,, \qquad \mathbf{G}(\alpha)^{\ddagger} = \left(\mathbf{G}(\alpha)^\top\mathbf{G}(\alpha) + \epsilon I_6\right)^{-1}\mathbf{G}(\alpha)^\top \tag{7.9}$$

where we chose $\epsilon = \epsilon(\alpha)$ in order to make its contribution significant for $\alpha \approx 0$ and negligible for $\alpha \gg 0$, e.g., $\epsilon(\alpha) = \frac{k_1}{\alpha + k_2}$, with $k_1, k_2 \in \mathbb{R}_{>0}$ properly tuned.

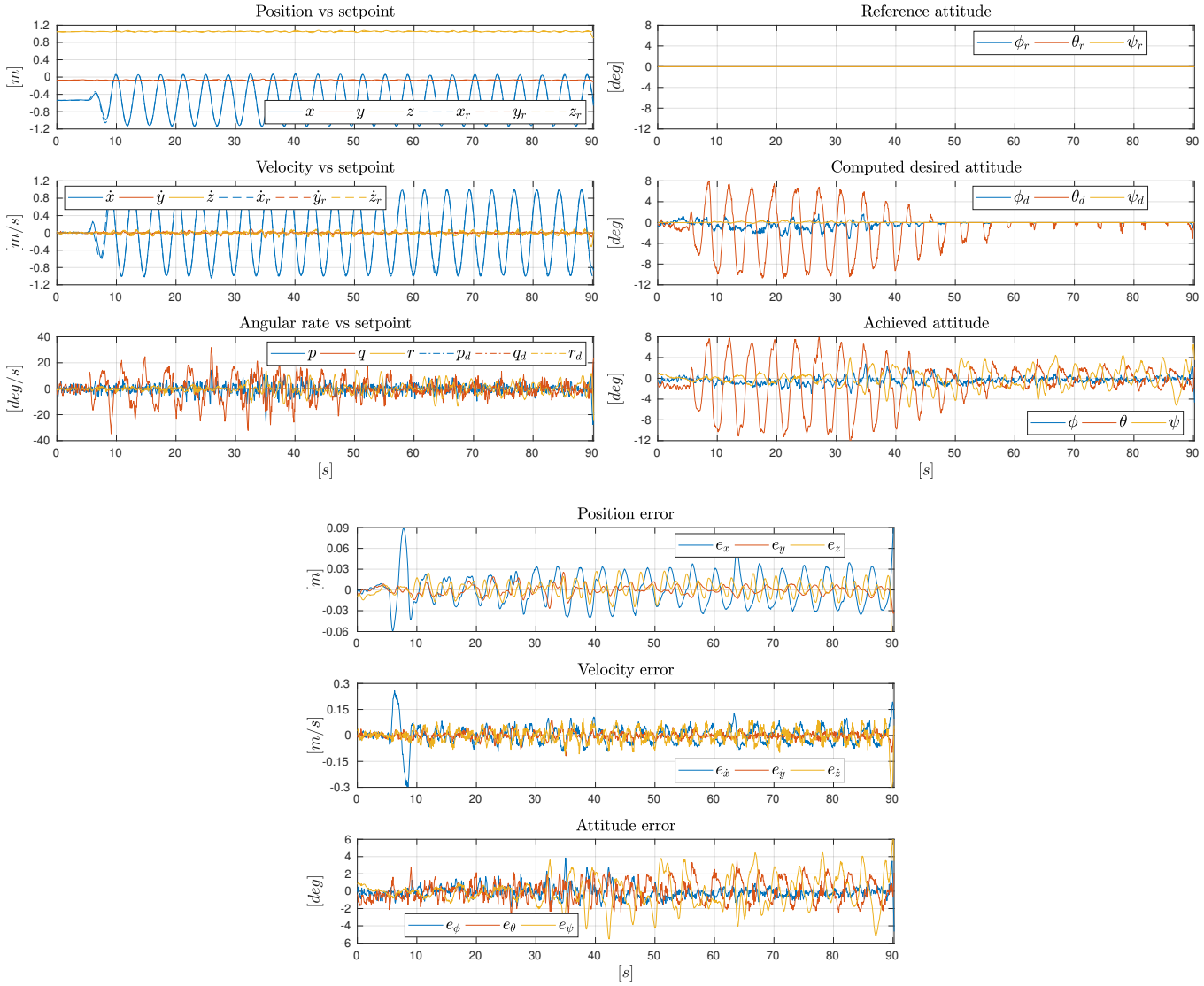


Figure 7.12 – Plots of the experimental results associated with FAST-Hex experiment 1. The name of each quantity and proper legends are reported on each plot.

As it is easy to assess from the plots of the top-left and bottom blocks of Fig. 7.12, the tracking of translational reference is always very good, as all the measured signals keep extremely close to their reference profiles. The plots of the top-right block in Fig. 7.12 deserve a particular attention. While the first one depicts the reference orientation (represented with a minimal representation with three angles for an ease of understanding) given as set-point to the controller, the second one shows the new desired profile computed by the control algorithm in order to comply with the body force constraint and to prioritize the translational tracking over the rotational one. This will be effectively the new controller set-point for the platform attitude. With reference to the this plot and the first one of Fig. 7.13, it is interesting to remark that as soon as the tilting angle is big enough to allow the tracking of the full-pose trajectory (i.e., for $t \geq 50\,\mathrm{s}$), the reference orientation

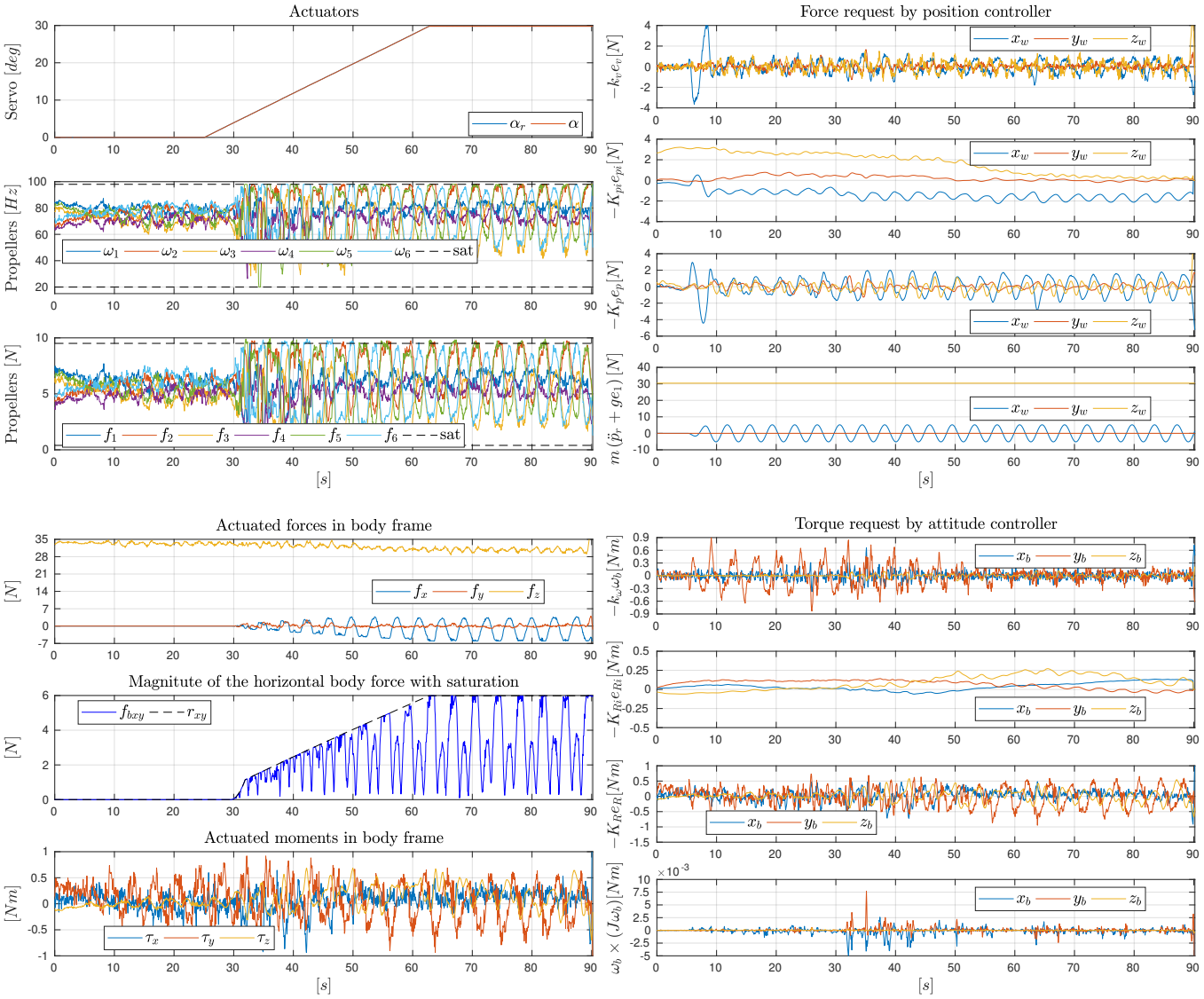


Figure 7.13 – Plots of the experimental results associated with FAST-Hex experiment 1. The name of each quantity and proper legends are reported on each plot.

almost does not need to be changed any more. Moreover, from the third plot of the same block, we can see that the desired orientation is properly followed.

The evolution of the controller actions related to each state errors has been separated in order to provide a clear understanding of the contribution of each term and is presented in the right blocks of Fig. 7.13. On the other hand, the data related to the actuators and the total body wrench which is generated by MRAV are portrayed in the plots of the left blocks of the same figure. In particular, we can appreciate how the measured tilting angle α follows the reference one α_r given by the high-level planner and how the actuator velocities and thrusts stay within the physical bounds. More in detail, it should be appreciated how the transition from UDT tpo MDT goes together with a wider span for the actuator commands.

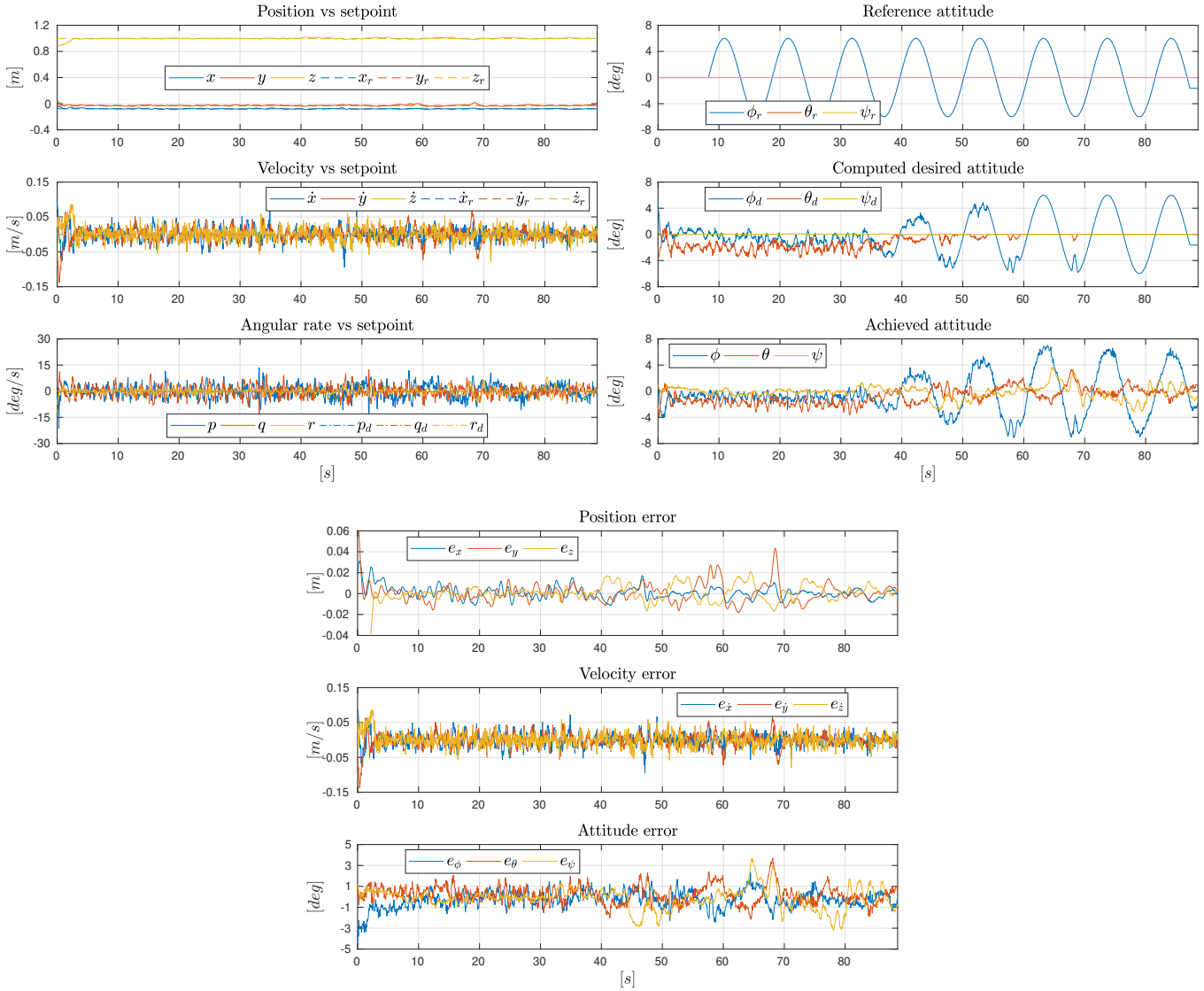


Figure 7.14 – Plots of the experimental results associated with FAST-Hex experiment 2. The name of each quantity and proper legends are reported on each plot.

FAST-Hex experiment 2

For this experiment we impose $p_{r_x}(t) = p_{r_y}(t) = 0\,\text{m}$ and $p_{r_z}(t) = 1.1\,\text{m}$, which corresponds to a static hovering position – see the corresponding signals in the top-left block of Fig. 7.14 – while the reference orientation is set to evolve sinusoidally along the first axis of the body frame, i.e., with an oscillating roll angle, as shown in the top-right plot of Fig. 7.14. it should be clear that this pose trajectory would not be feasible for a UDT MRAV, as it would start to move laterally. Also in this experiment, the FAST-Hex morphs its configuration from UDT to MDT by increasing the lateral body force capability from $r_{xy} = 0\,\text{N}$ to $r_{xy} = 6\,\text{N}$ throughout the experiment, see the bottom-left block of plots of Fig. 7.15.

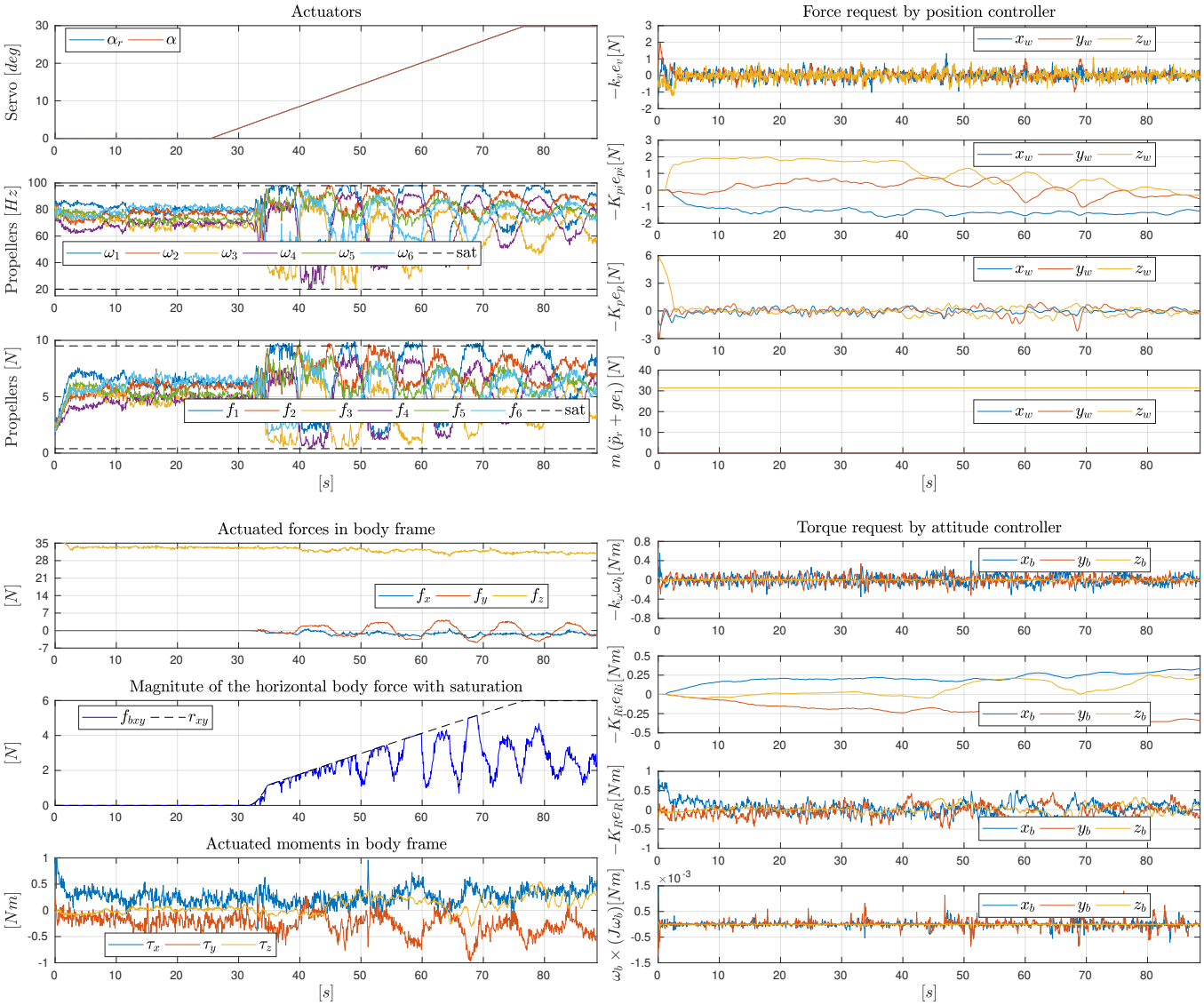


Figure 7.15 – Plots of the experimental results associated with FAST-Hex experiment 2. The name of each quantity and proper legends are reported on each plot.

With reference to the top-right block of plots of Fig. 7.14, we see how the controller 'cuts' the reference orientation as long as its lateral force control authority is not big enough (i.e., for $t \leq 40\,\mathrm{s}$). On the left blocks of Fig. 7.15 we present again the control actions related to each component of the state error, while on the right blocks we depict the data related to the actuators and to the generated body wrench. In particular, it should be appreciated how the constraints related to the lateral body force and also to the actuator velocities (forces, equivalently) are always satisfied throughout the experiments. This demonstrate the good capability of the controller to deal with the system constraints (yet in a conservative way) and to seamlessly drive the platform in both the UDT and MDT configurations.

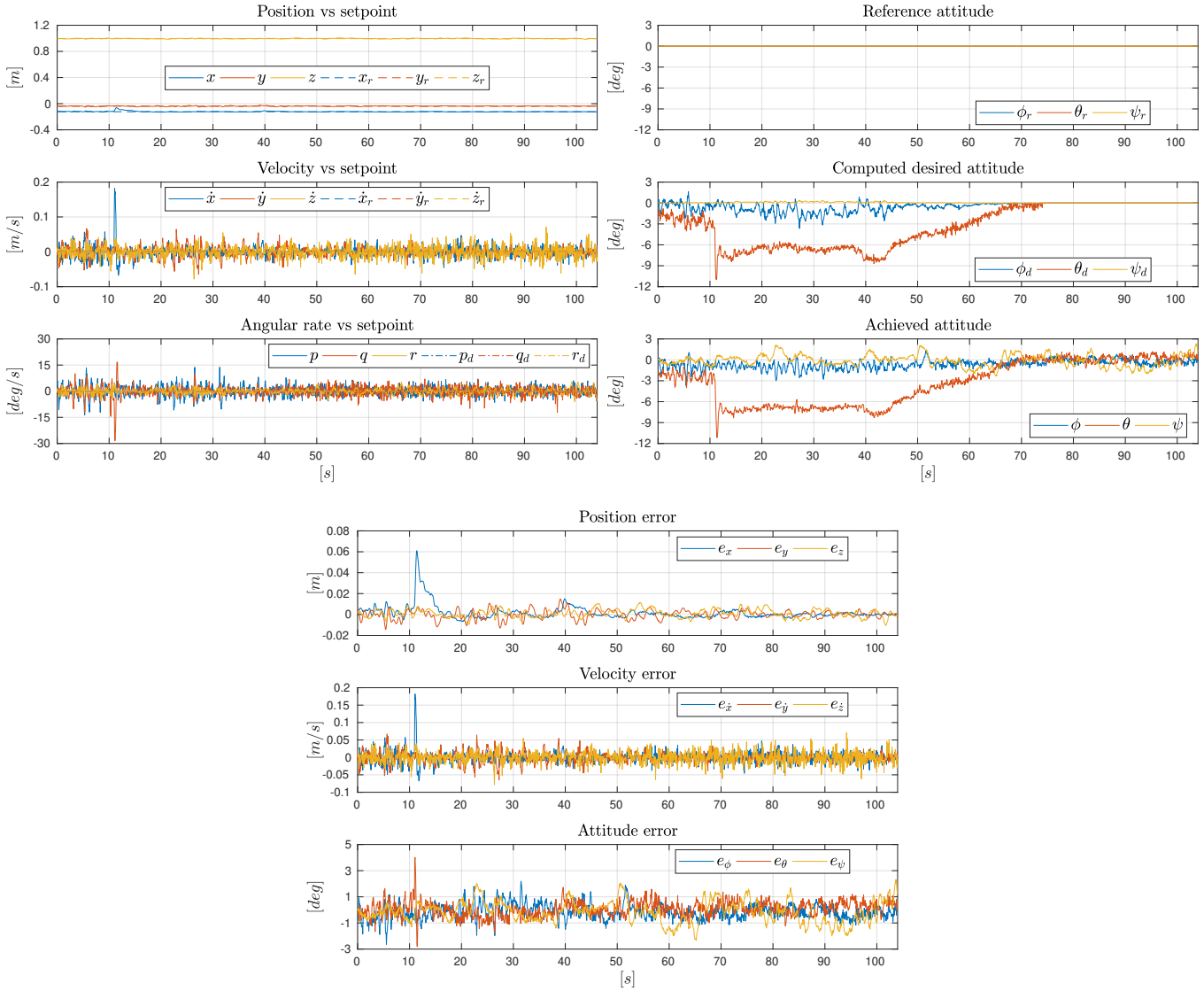


Figure 7.16 – Plots of the experimental results associated with FAST-Hex experiment 3. The name of each quantity and proper legends are reported on each plot.

FAST-Hex experiment 3

This last experiment is designed to highlight, apart from the good performance of our control algorithm, the extremely interesting advantages of MDT MRAVs (and in particular of the FAST-Hex) compared to UDT ones. The reference trajectory for this experiment is a static hovering with flat orientation, that is $p_{r_x}(t) = p_{r_y}(t) = 0\,\text{m}$, $p_{r_z}(t) = 1.1\,\text{m}$, and $\mathbf{R}_r(t) = \mathbf{I}_{3\times3}$, $\forall t \leq 0\,\text{s}$. At $t = 12\,\text{s}$, the platform is perturbed with a constant lateral force in world frame along the positive direction of the x axis, as it is visible from the impulsive behavior of the plots of Fig. 7.16 and of Fig. 7.17 at that time instant. As a consequence, three separated conditions can be recognized in this experiment. For $t \in [0\,\text{s}, 12\,\text{s}]$, the platform is in its

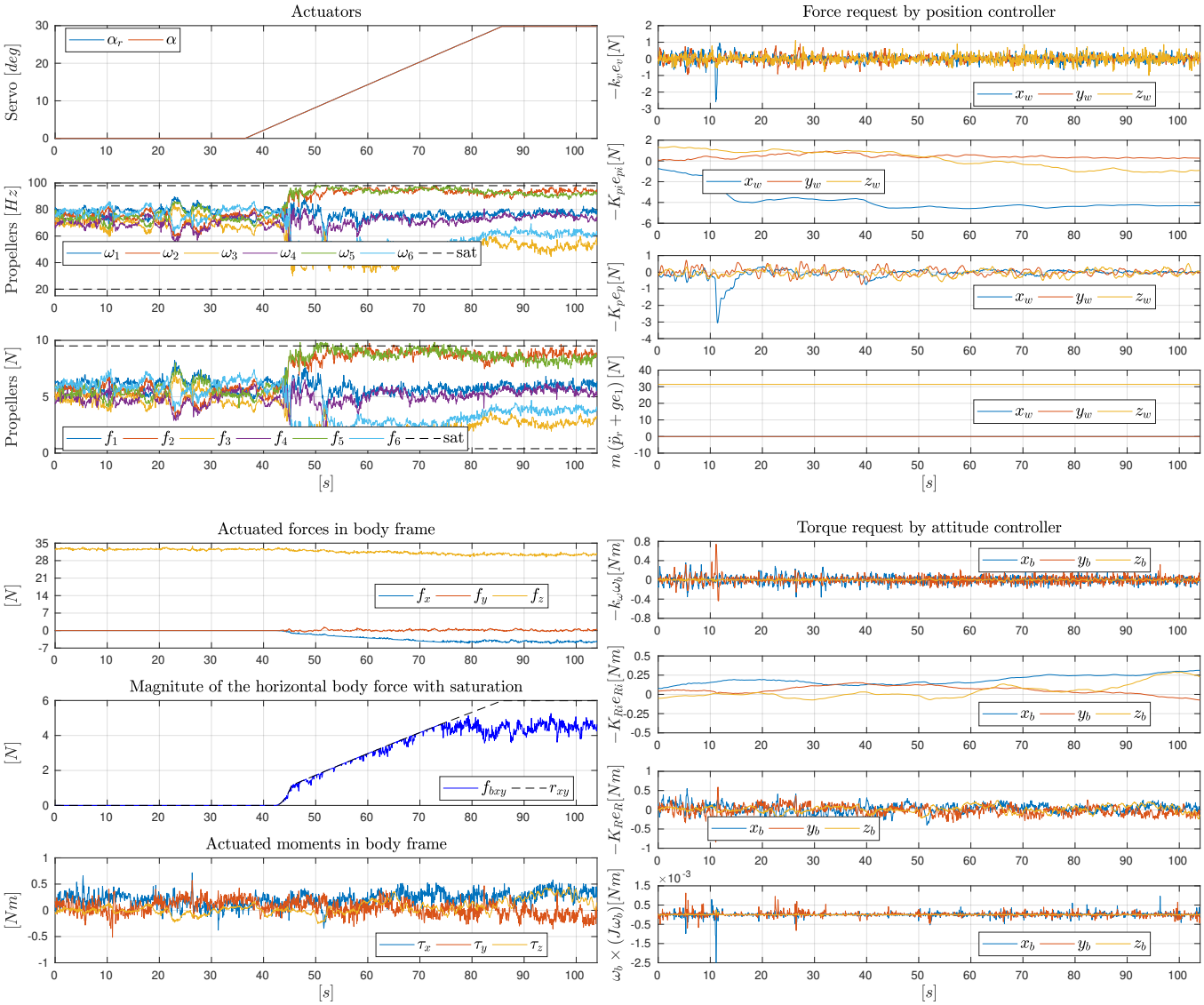


Figure 7.17 – Plots of the experimental results associated with FAST-Hex experiment 3l. The name of each quantity and proper legends are reported on each plot.

UDT configuration and can track well the hovering trajectory, since no nominal disturbance is acting on it. However, once the lateral force is activated, the vehicle cannot resist the disturbance while maintaining a flat orientation, as it would be required by the reference motion. Therefore, for $t \in [12\,\mathrm{s}, 70\,\mathrm{s}]$ the controller is required to reshape the desired orientation, as depicted in the top-left block of plots of Fig. 7.16. Finally, once the tilting angle is big enough for the platform to generate an adequate lateral force compensation (i.e., for $t \leq 70\,\mathrm{s}$), it becomes possible to track the reference pose without the need to change it. As shown by the right block plots of Fig. 7.17, the system constraints are (again) always respected.

This very meaningful assessment concludes the big set of experiments performed and presented to properly validate the LBF controller performance.

Validation of the NMPC controller

In this section, we first apply the procedure outlined in Sec. 4.2 in order to identify the state and input limitations in a real case of interest. This operation is performed for two different actuator models. Then, we show and thoroughly discuss the trajectory tracking results obtained from the application of the proposed Nonlinear Model Predictive Control (NMPC) algorithm to different MRAV prototypes. The performance of the controller are evaluated by means of realistic simulations and in real experiments, with an UDT quadrotor, a MDT hexarotor, a convertible hexarotor with orientable actuators, and a MDT MRAV recovering from a rotor failure.

Application of the identification to the hardware setup

In this context, we explicitly show how to apply the identification procedure related to the detailed model in a practical case. To do this, we use a dedicated testbed, composed of a single motor endowed with a propeller and controlled by a flight control board embedding one ESC controlling a motor. The setup is connected to a ground station via a serial cable. The motor is fixed on a mechanical structure, thus preventing its motion due to the exerted force. On a ground station we run a software which allows to command the motor with a desired velocity profile while collecting the measured values at the same time. The hardware setup under examination is the one of the custom-made MDT hexarotors presented in Sec. 6.1, and it will denoted with the subscript $\bullet_{\mathrm{H}}$. It is composed of a MikroKopter[3] electric motor MK3638 coupled with a 12X4.5' propeller and controlled by a BL-Ctrl V2.0 ESC. The low-level control of the rotor velocity is performed in closed-loop by the ESC using the Adaptive Bias Adaptive Gain (ABAG) algorithm [Franchi–2017].

In this specific case, the constraints on the minimum and maximum velocities are due to the ESCs. Specifically, the actual rotor velocity is estimated by each ESC without any additional sensor and the quality of such estimation is proportional to the spinning rate. This causes the velocity to have a lower bound, in order to be properly estimated by the controller with a certain precision. On the other hand, the limited arithmetic capabilities of the ESC micro-controller (which allows only 8-bit additions and has no floating point unit) translates into a velocity upper bound [Franchi–2017]. For this setup, we identify $\underline{\mathrm{w}} = 16\,\mathrm{Hz}$ and $\overline{\mathrm{w}} = 102\,\mathrm{Hz}$. In particular, the upper limit reveals to satisfy the maximum current limitation of 20 A reported in the motor data-sheet. Finally, using (3.17) we obtain the limits $\underline{f} \approx 0.25\,\mathrm{N}$ and $\overline{f} \approx 10.3\,\mathrm{N}$ used to constrain the NMPC solver.

As far as the identification of the acceleration limits is concerned, we generate a set of increasing $\dot{\mathrm{w}}$ spanning the range $\pm[20, 300]\,\mathrm{Hz/s}$ with a step of $10\,\mathrm{Hz/s}$, centered at a given average velocity level w_h^*. Each ramp fragment takes values in the set $[\mathrm{w}_h^* - \delta_h, \mathrm{w}_h^* + \delta_h]$, with $\delta_h = 10\,\mathrm{Hz}$. With reference to Fig. 4.2, for each ramp we take 30% of the samples centered in the middle of the interval (highlighted

[3] `http://www.mikrokopter.de/en/home`

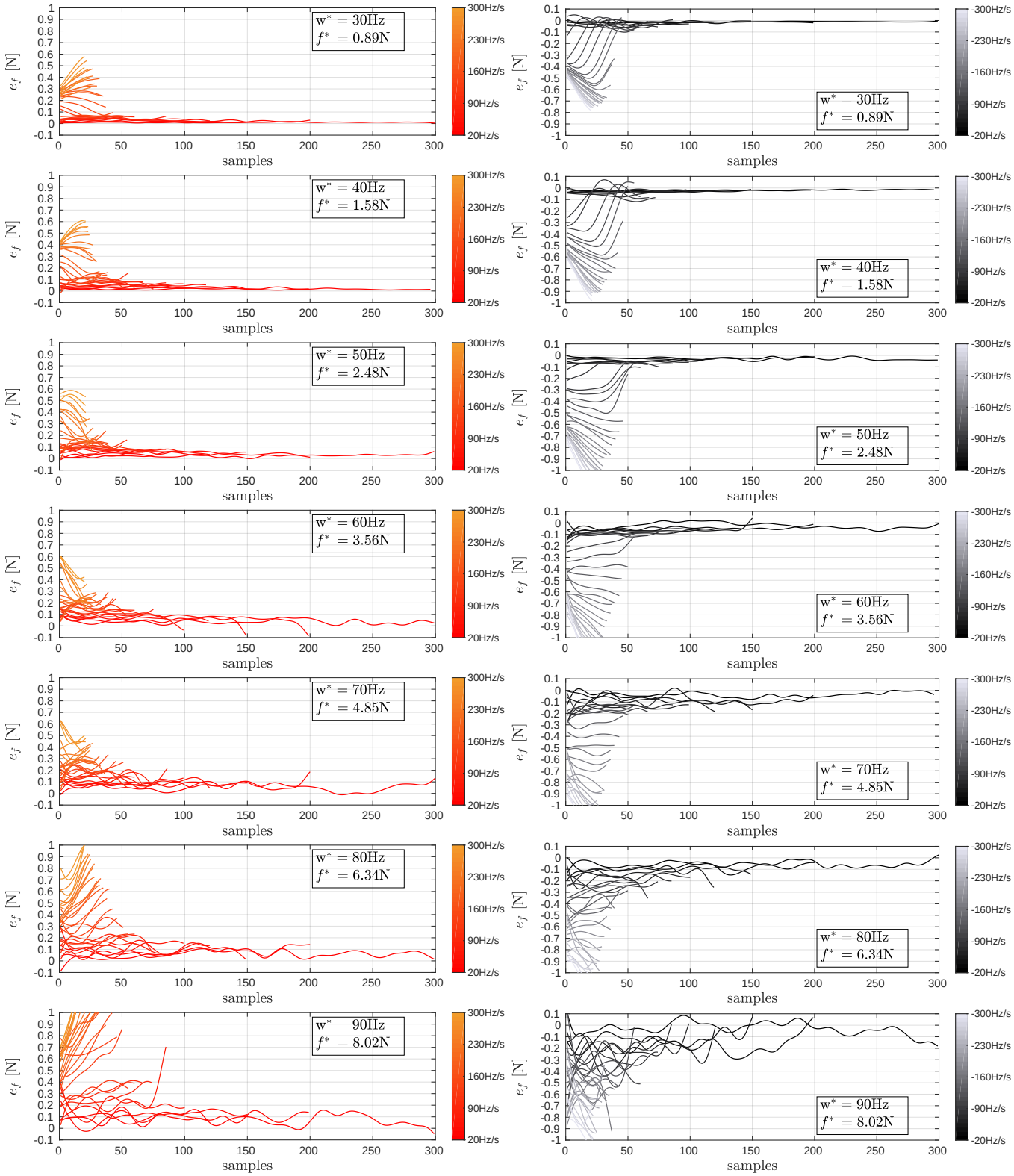


Figure 7.18 – Plots of the force error trends associated to the acceleration intervals (positive on the *left*, negative on the *right*) at the different set-point velocities w_h^* for the hardware of the MDT hexarotors. The error associated with increasing accelerations is represented with different color shades.

Table 7.1 – Identified acceleration limits for the hexarotor actuators.

w [Hz]	30	40	50	60	70	80	90
$\underline{\dot{\mathrm{w}}}_{\mathrm{H}}$ [Hz/s]	−120	−160	−200	−140	−160	−160	−140
$\overline{\dot{\mathrm{w}}}_{\mathrm{H}}$ [Hz/s]	200	200	200	160	180	180	180

with orange rectangles) and compute the correspondent force error using (4.7). The operation is repeated at different set-points w_h^* in the set $[30, 90]$ Hz with a step of 10 Hz, in order to span the set of admissible velocities previously estimated. The plots of the force error trends related to the hardware of our MDT hexarotor platforms is shown in Fig. 7.18. In each subplot, notice that the number of samples related to increasing values of $|\dot{\mathrm{w}}|$ is gradually decreasing. This happens because all the ramps at a given w_h^* take values in the interval $[\mathrm{w}_h^* - \delta_h, \mathrm{w}_h^* + \delta_h]$. Therefore, an increase in the ramp slopes causes a decrease in the time duration associated with the segments.

Remark three facts: (i) At the same velocity set-points, increasing force errors are associated with increasing acceleration values, on average. This suggests that high acceleration references (of both signs) are difficult to be tracked and fosters the idea to constrain them with lower and upper bounds. (ii) For different set-point velocities, the profile of the force error at corresponding acceleration intervals is different. This confirms the claim that the limits are velocity-dependent. In particular, we observe that while increasing values of set-points seem to cause increasing force error for positive accelerations, such trend is not pursued by negative accelerations. A reasonable explanation for such effect could be the fact that the active braking, which intervenes only for negative accelerations, is not behaving in the same way for different velocity levels. (iii) At the same velocity set-points, the force error associated with negative accelerations is larger, on average, with respect to the one associated with positive accelerations. This reveals that, despite the use of the active-braking strategy, the deceleration of a rotor produces a worse force tracking than the corresponding acceleration.

In order to identify the acceleration limits $\underline{\dot{\mathrm{w}}}$ and $\overline{\dot{\mathrm{w}}}$, we define $\epsilon_f \approx 0.2\,\mathrm{N}$ as the force error threshold, admitting slightly bigger values at high velocity set-points. As we will see in the experimental validation plots, such value allows to obtain conservative limits that preserve the platform stability also during agile trajectory tracking. As a general rule, such threshold shall depend on the particular task for the vehicle. The identified acceleration limits for the hardware of the MDT hexarotors are collected in Tab. 7.1, where velocity data are expressed in [Hz], while acceleration ones in [Hz/s]. These values are interpolated with continuous functions of the motor velocity. Ultimately, by using (3.17) and (4.9), we easily obtain $\underline{\dot{f}}(f)$ and $\overline{\dot{f}}(f)$.

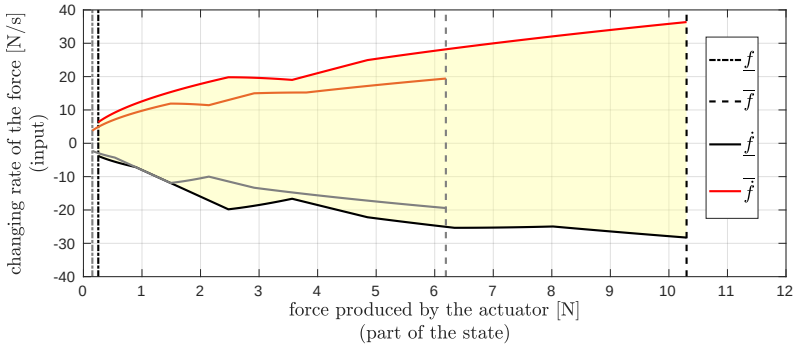


Figure 7.19 – State and input constraints given to the NMPC for the hardware of the hexarotors (*darker colors*) and the quadrotors (*lighter colors*).

A second hardware setup is analyzed, for identifying the constraints in case of the available under-actuated UDT quadrotor, denoted with the subscript $\bullet_{\mathrm{Q}}$. This setup combines a motor MK2832/35 with a 10X4.5' propeller from mikrokopter, controlled by the same model of ESC of the previous case. Since the procedure is the same, we avoid presenting intermediate results for this case and just outline the final results. For both hardware setups, the profile of the constraints for the actuator forces and their derivatives are depicted in the plot of Fig. 7.19, where the admissible set of values is represented with the yellow area. Consistently with the previous results, the limits on $\underline{\dot{f}}(f)$ and $\overline{\dot{f}}(f)$ are not perfectly symmetric.

Implementation details

The predictive control algorithm is implemented using MATMPC[4], a recently developed MATLAB®-based nonlinear NMPC toolbox [Chen–2018b] which embeds the multiple shooting method [Bock–1984] and uses the state-of-the-art Real Time Iteration (RTI) scheme [Diehl–2002]. The RTI scheme performs a single Sequential Quadratic Programming (SQP) iteration to solve the Optimal Control Problem (OCP). To do this, a linearization of the system constraints (5.29) and (5.30) is performed to obtain a quadratic programming problem, to be solved at each sampling time. To reduce the computational time, in [Chen–2017; Chen–2018a], a procedure called partial sensitivity update is proposed, where the constraint linearization is updated only if the dynamics around the generated trajectory exhibits a certain degree of nonlinearity. To reduce the computational complexity, the Quadratic Programming (QP) problem is condensed using the algorithms discussed in [Andersson–2013]. The required linear algebra routines are implemented using OpenBLAS [–]. The resulting dense QP is solved by qpOASES[5] [Ferreau–2014], which employs on-line active-set method with warm-start strategy. The algorithmic routines of MATMPC are written using MATLAB® C Application Programming Interface (API) and available as MEX functions. The tool supports fixed

[4] `https://github.com/chenyutao36/MATMPC`
[5] `https://projects.coin-or.org/qpOASES/wiki`

step Runge-Kutta (RK) integrator for multiple shooting and obtains the derivatives that are needed to perform the optimization from the toolbox CasADi[6]. Such an implementation has been chosen mainly due to the particular ease of test and development of MATLAB/Simulink® compared to pure C/C++.

In order to design the software architecture, we relied again on the GenoM3 abstraction level, which allows to encapsulate software functions inside independent components. More in detail, it has been used as a wrapper for the robot low-level controller and the sensors. This allows to obtain high flexibility in the development and in the use of the components. With reference to our architecture, the software in MATLAB/Simulink® communicates with the GenoM3 modules using the Robot Operating System (ROS) middleware, which is compliant with the soft real-time constraints required for our experiments, i.e., a control bandwidth larger than 200Hz and a latency smaller than 10ms. Since MATLAB/Simulink® is not meant for a hard real-time execution, the hardware is commanded via the GenoM3 components, which essentially behave like drivers.

For all the experiments presented in this section, we chose a prediction horizon of $t_H = 1$s, sampled at $N + 1 = 11$ shooting points. Therefore, the discretization time of the nonlinear Model Predictive Control (MPC) algorithm, being the length of one of the N intervals, results $T_{\text{MPC}} = 0.1$s. Even though the internal MPC prediction is performed at 10Hz, the controller runs at a frequency always larger than 200Hz. Such technique, employed by many state-of-the-art contributions, e.g., [Kamel–2017a], allows the predictive algorithm to simulate the model along a wider prediction horizon with less computational effort. Indeed, as observed in [Falanga–2018], the number of discretization nodes roughly increases the computational time t_{solv} by $O(N^2)$. Basically, one should guarantees a control sample time T_{ctrl} at least equal to the average time t_{solv} needed for the algorithm to solve the OCP. On the other hand, the prediction horizon should be long enough to cover at least the time of one controller iteration. In mathematical terms, this translates in the following chain of inequalities

$$t_{\text{solv}} \leq T_{\text{ctrl}} \leq T_{\text{MPC}} \leq t_H \tag{7.10}$$

As soon as the solver has found the solution $\tilde{\mathbf{u}}$ to the OCP for all the N future sampling points, only the first input $\mathbf{u}_1^*$ of the sequence is actually sent to the robot. Then, the actual measure of the state $\hat{\mathbf{x}}$ coming from the sensors is used as feedback to re-initialize the MPC problem. At this point, the algorithm is ready for another iteration. It should be noted that, despite (7.10), there is no guarantee that a solution to the OCP is always available in due time at each time step. If, at a given instant (say at step k), the time required to compute the solution is occasionally larger than a given threshold, a back-up solution must be taken to guarantee reliability of the control system. In this thesis, this solution consists in taking the possibly sub-optimal but admissible value $\mathbf{u}_k = \hat{\mathbf{u}}_{1|k-1}$, computed as part of the solution to the OCP at time $k - 1$.

[6] `https://github.com/casadi/casadi/wiki`

As far as the representation of the robot orientation is concerned, for the particular experiments presented in this section we decided to use a minimal parametrization with three angles, in particular the $3-2-1$ one (yaw-pitch-roll), i.e.,

$$\boldsymbol{\eta} = \begin{bmatrix} \phi & \theta & \psi \end{bmatrix}^\top \tag{7.11}$$

With reference to this ordered sequence, we have that

$$\begin{aligned} \mathbf{R} &= \mathbf{R}_{\mathbf{z}}(\psi)\mathbf{R}_{\mathbf{y}}(\theta)\mathbf{R}_{\mathbf{x}}(\phi) \\ &= \begin{bmatrix} c_\theta c_\psi & s_\phi s_\theta c_\psi - c_\phi s_\psi & s_\phi s_\psi + c_\phi s_\theta c_\psi \\ c_\theta s_\psi & c_\phi c_\psi + s_\phi s_\theta s_\psi & c_\phi s_\theta s_\psi - s_\phi c_\psi \\ -s_\theta & s_\phi c_\theta & c_\phi c_\theta \end{bmatrix} \end{aligned} \tag{7.12}$$

where $\mathbf{R}_\bullet(\alpha)$ denotes a rotation around one of the main body frame axes $\{\mathbf{x}, \mathbf{y}, \mathbf{z}\}_B$ of an angle α, while s_α, c_α indicate $\sin(\alpha)$ and $\cos(\alpha)$, respectively. Using this convention, we can express the body frame angular velocity as a function of the vector $\dot{\boldsymbol{\eta}}$, that contains the so-called *Euler rates*

$$\boldsymbol{\omega} = \mathbf{T}\dot{\boldsymbol{\eta}} \tag{7.13}$$

In particular, with respect to the specific parametrization of (7.12), we have

$$\mathbf{T} = \begin{bmatrix} 1 & 0 & -s_\theta \\ 0 & c_\phi & s_\phi c_\theta \\ 0 & -s_\phi & c_\phi c_\theta \end{bmatrix}. \tag{7.14}$$

Inverting (7.13) allows to write explicitly the Euler rates as a function of the body angular velocity (expressed in body frame) in the model dynamics. This representation, like all the minimal parametrizations given by three angles, has a singularity, which in the specific case occurs when $\theta = \pi/2$. In general, all these conventions should be avoided if the robot orientation is supposed to evolve in the complete $\mathrm{SO}(3)$ manifold. However, in the particular case of the trajectories that we have tested, we safely used this representation by explicitly avoiding singular configurations for the platform pose. We chose to not use the re-arranged elements of $\mathbf{R}$ or a unit quaternion for a simple matter of convenience. Indeed, in such cases a larger state vector would have been needed. Furthermore, additional constraints, e.g., the orthogonality of the rotation matrix or the unitary-norm for the quaternion, should have been added in the resolution of the OCP, thus increasing the solver computational time and, by consequence, slowing down the available bandwidth of the controller. This can easily be dealt with, of course, by using a slightly more powerful computation unit. To conclude, we would like to stress that the proposed NMPC framework does not depend on the particular orientation representation and easily adapts to the others without the need to deal with additional theoretical issues.

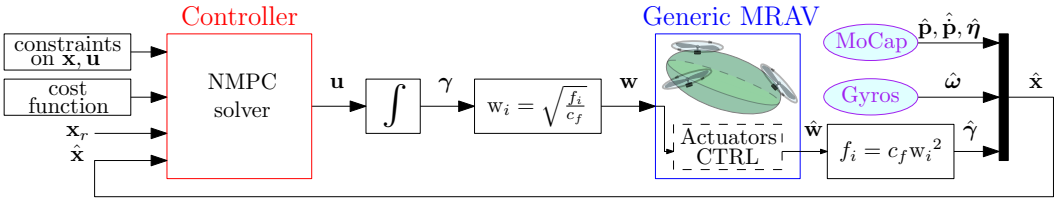


Figure 7.20 – Block diagram of the experimental setup architecture. The main components are highlighted with different colors.

The cost function weights in (5.27) are specified at the beginning of the description of each experiment and simulation. In general, they have been chosen on a case-dependent basis taking into account heuristic considerations and often following a *trial-and-error* procedure. The automatic tuning of such weights is an important topic which is left for future work. Throughout all experiments and simulations presented in this section, the input terms in the cost function have not been considered, i.e., the entries of the weights $\mathbf{R}_h$ related to the input are equal to zero. This has been done with the goal to exploit the MRAVs potentialities until their limits by taking advantage of the actuator dynamics up to their bounds. Therefore, we decided to test the NMPC algorithm by discarding these regularization terms. In all the performed tests, including the most "aggressive" ones, we never encountered problems in the regularity of the input evolution. Furthermore, despite the strong aggressiveness of some of the reference state trajectories to the NMPC algorithm, we never triggered the activation of the slack variables.

Finally, regarding the choice of the input bounds along the prediction horizon, we selected

$$\tilde{f}_{i,k+h} = f_{i,k}, \quad h = 0, \dots, N-1 \tag{7.15}$$

i.e., the limits are kept constant along the future window. This choice has been motivated by a matter of simplicity of implementation. A more rigorous choice could be to select the time-varying $\tilde{f}_{i,k+h}$ in relation to the predicted state evolution at the previous control step for $t = (k-1)T$. The comparison within the results produced by these two configurations is left as future investigation.

Experimental setup

The experimental setup architecture, whose block diagram is portrayed in Fig. 7.20, can be conceptually divided into three main components: the NMPC controller, which periodically computes the input of the actuator controllers, the physical aerial robot to be controlled, and the sensors, used to retrieve the information about the MRAV state that is employed as feedback in the closed-loop control strategy. Each block exchanges information with the others thanks to a properly-designed software architecture.

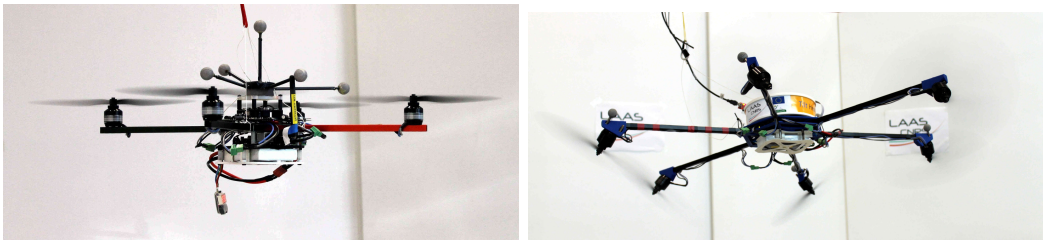


Figure 7.21 – Photos of the quadrotor (*left*) and the Tilt-Hex (*right*).

Table 7.2 – Physical parameters of the quadrotor in the experiments.

Quadrotor		
Parameter	Value	Unit
m	1.04	Kg
$\mathbf{J}(:,1)$	$[0.015\ 0\ 0]^\top$	$\mathrm{Kg\,m^2}$
$\mathbf{J}(:,2)$	$[0\ 0.015\ 0]^\top$	$\mathrm{Kg\,m^2}$
$\mathbf{J}(:,3)$	$[0\ 0\ 0.015]^\top$	$\mathrm{Kg\,m^2}$
c_i	$(-1)^{i-1}$	[ ]
c_f^τ	1.69×10^{-2}	m
c_f	5.95×10^{-4}	$\mathrm{N/Hz^2}$
$\mathbf{R}_{A_i}^B$	$\mathbf{R}_z((i-1)\frac{\pi}{2}))\mathbf{R}_x(\alpha)\mathbf{R}_y(\beta)$	[ ]
$\mathbf{p}_{A_i}^B$	$\mathbf{R}_z((i-1)\frac{\pi}{2}))[l\ 0\ 0]^\top$	[ ]
α	0 35	deg
β	0	deg
l	0.23	m

The presented NMPC algorithm has been implemented on a ground-station PC equipped with an Intel® 2.60GHz Core™ i7-6700HQ CPU (x8) and 32 GB RAM which runs the Linux Ubuntu 16.04 LTS operating system. As it can be observed from Fig. 7.20, the control input $\mathbf{u}$, which provides the actuators' force derivatives references, is integrated and then converted into a rotor velocity command $\mathbf{w}$, thanks to the inversion of the force generation model. The resulting velocity set-points are finally transferred to the module of the low-level controllers on-board the aerial platform, by means of a serial cable.

As far as the aerial vehicles are concerned, we tested the control algorithm with both a UDT and a MDT multi-rotor platform. The first one is a classic under-actuated quadrotor built by assembling off-the-shelf components from MikroKopter and by adding some custom-made features realized in-house with 3D printed components, like the battery support. The other MRAV is the Tilt-Hex, already presented in Sec. 6.1. The two platforms are depicted in Fig. 7.21, while the physical parameters of the quadrotor are condensed in Tab. 7.2 (for the ones of the Tilt-Hex we refer the reader to Tab. 6.1). As already clarified, both vehicles are endowed with an electronic board with independent ESCs for the control of the rotor velocities.

The main sensors integrated in our experimental framework are the onboard gyroscope, the MoCap system, and speedometers of each propeller rotational speed:

- the Gyroscope measures the rotational velocity of the vehicle around each of the body frame axis;
- the MoCap system provides the information regarding the robot position and orientation with respect to the inertial reference frame, whose origin is fixed in a particular point of the robots workspace. The platform linear velocity is numerically computed online from the position measurements, using multi-sample least squares model fitting;
- the rotor spinning velocities are measured by the low-level ESC controller by computing the time elapsed between two phase switches (which depends on the motor number of poles) and reducing the measurement noise with an exponential moving average filter. Ultimately, the rotor velocities are converted into the actuator forces, thanks to the force generation model, and used to complete the information of the measured full-state $\hat{\mathbf{x}}$ of the MRAV.

Note that the accelerometers have been disregarded from the sensor fusion, since we assessed that the noise in their measurements was causing an offset in the estimation of the linear velocity, which motivated the numerical computation of the latter. In general, the effect of such velocity offset on the tracking performance is quite more evident on predictive controllers with respect to reactive ones, given the fact that a wrong state estimation generates an erroneous evolution of the model internally simulated and, in turn, a misleading control input that finally produces an inaccurate trajectory tracking.

Experimental validation

After having introduced the experimental setup and described the implementation details of the algorithm, here we presents the results of the experiments obtained with aforementioned MRAVs. First of all, we shall introduce the experimental outcomes obtained with the UDT quadrotor and with the MDT Tilt-Hex platforms. A video of the experiments is available online [video04–2019]. Secondly, we show the potentiality of the control approach with some realistic simulations of a convertible hexarotor (the FAST-Hex) and of the Tilt-Hex after the failure of one actuator.

Quadrotor experiments

According to the choices we made for the state and input vectors, defined by (4.14)-(4.15), and for the orientation description associated to (7.11), in the case of the quadrotor model we have $\mathbf{x} \in \mathbb{R}^{16}$ and $\mathbf{u} \in \mathbb{R}^4$. With this configuration, the average NMPC solver time is $t_{\text{solv}} = 3.5$ms. In the following, we present the tracking results obtained by the quadrotor with two different trajectories. The first one combines a sinusoidal chirp motion along one component of the position with a steadily horizontal and constant-heading desired orientation. Such dynamic and decoupled motion will be also given as reference to the Tilt-Hex in order to compare the performances of the two platforms. This allows to highlight the different behaviors of UDT and

Table 7.3 – Parameters used in the quadrotor experiments.

Parameter	Value	Unit
ν	1.2	m
ξ	0.025	$\frac{\text{rad}}{\text{s}^2}$
$\bar{t}$	44.84	s
$\boldsymbol{\epsilon}_{\mathbf{p}}$	$[-0.5\,0.2\,0.2]^\top$	m
ρ	$0.125 : 0.125 : 1.75$	[ ]
$\mathbf{Q}_{\mathbf{p}}(j,j)\vert_{j=1,2,3}$	$500, 300, 300$	[ ]
$\mathbf{Q}_{\dot{\mathbf{p}}}(j,j)\vert_{j=1,2,3}$	$1.9, 1.9, 1.9$	[ ]
$\mathbf{Q}_{\boldsymbol{\eta}}(j,j)\vert_{j=1,2,3}$	$0.1, 0.1, 40$	[ ]
$\mathbf{Q}_{\boldsymbol{\omega}}(j,j)\vert_{j=1,2,3}$	$0.15, 0.15, 0.15$	[ ]
$\mathbf{Q}_{\ddot{\mathbf{p}}}(j,j)\vert_{j=1,2,3}$	$0, 0, 0$	[ ]
$\mathbf{Q}_{\dot{\boldsymbol{\omega}}}(j,j)\vert_{j=1,2,3}$	$0, 0, 0$	[ ]

MDT aerial robots. On the other hand, we designed the second reference motion in order to test the controller compliance with the actuator bounds, when dealing with a discontinuous trajectory. In particular, these tests highlight the importance of the control compliance with the input constraints for the preservation of the system stability.

Position chirp trajectory

In the first experiment, the quadrotor is required to track a position reference $\mathbf{p}_r = [c(t)\,0\,0]^\top$, where the chirp signal $c(t)$ is a sine with varying frequency, with amplitude $\nu = 1.2\,\text{m}$, with a triangular frequency that linearly increases from $\xi_0 = 0\,\text{rad/s}$ to $\xi_{\bar{t}} = 1.12\,\text{rad/s}$ with a slope $\xi = 0.025\,\text{rad/s}^2$ in the interval $[0, \bar{t}[$, $\bar{t} = 44.84\,\text{s}$, and then decreases with a slope $-\xi$ in the interval $[\bar{t}, 2\bar{t}[$. In mathematical terms, this translates into:

$$c(t) = \nu \sin\left(\xi(t)\ t\right),\ \ \xi(t) = \begin{cases} \xi t & \text{if } t \in [0, \bar{t}[\\ \xi(\bar{t} - t) & \text{if } t \in [\bar{t}, 2\bar{t}[\end{cases} \tag{7.16}$$

On the other hand, the attitude reference is constantly $\boldsymbol{\eta}_r = [0\,0\,0]^\top$. Moreover, the desired position derivatives $\dot{\mathbf{p}}_r$, $\ddot{\mathbf{p}}_r$ and the rotational derivatives $\boldsymbol{\omega}_r$, $\dot{\boldsymbol{\omega}}_r$ are consistent with the definitions of $\mathbf{p}_r$ and $\boldsymbol{\eta}_r$, respectively. Regarding the form of the diagonal matrices employed to weight the different error terms inside the cost function, we used $\mathbf{Q}_k = diag(\mathbf{Q}_{\mathbf{p}}, \mathbf{Q}_{\dot{\mathbf{p}}}, \mathbf{Q}_{\boldsymbol{\eta}}, \mathbf{Q}_{\boldsymbol{\omega}}, \mathbf{Q}_{\ddot{\mathbf{p}}}, \mathbf{Q}_{\dot{\boldsymbol{\omega}}})$, $\forall k \in \{0, \ldots, N\}$. The values of the trajectory parameters and the diagonal sub-blocks $\mathbf{Q}_\bullet$ chosen for the quadrotor experiments are displayed in Tab. 7.3. •The latter ones are the result of a trial-and-error procedure that we performed, in compliance with some heuristic guidelines, in order to obtain satisfactory tracking performance. In particular, the weights associated with the orientation error have been selected much smaller than the ones related to the position error, given the impossibility for the particular

platform to track the roll and pitch references. On the other hand, the yaw error has a larger impact with respect to the other two angular components, as the authority around this axis is still present despite the under-actuation. Finally, the feed-forward terms related to $\ddot{\mathbf{p}}_d$ and $\dot{\boldsymbol{\omega}}_d$ turned out to be not very relevant in these experiments. This explains why their entries are weighted with null gains.

With reference to the desired trajectory, the platform is required to keep a flat orientation while moving laterally along the x-axis. This motion is unfeasible for a UDT aerial vehicle, since the only way it has to steer the thrust force is by re-orienting its body chassis, with no possibility to keep it horizontal. We provided an unfeasible rotational profile on purpose with the intent of showing that the proposed NMPC scheme can manage the re-generation and tracking of a generic trajectory, subject to the limitations imposed by the particular MRAV under analysis, without the need to resort, e.g., to differential flatness. In this way, the user does not need to explicitly compute the particular platform-dependent feasible trajectory, but can delegate this task to the predictive controller, which automatically adapts the reference profile according to the robot constraints. It is worthwhile to underline that the position errors would have been smaller if a feasible reference trajectory was endowed to the controller.

The plots related to the trajectory tracking are depicted in Fig. 7.22. As it is visible from the first one, related to the position tracking, the trajectory is symmetric with respect to the time instant $t = \bar{t}$. While the position and the linear velocity are globally well tracked, the second components of the orientation and the angular velocity deviate consistently from their reference signals. This is a natural consequence of the platform inability to produce any lateral force in body frame, which makes it under-actuated. More in detail, the peaks in the measured robot pitch θ in the third plot are synchronized with the ones of the position p_x in the first plot. Indeed, the edge points on the sine corresponds to the moments of maximum lateral acceleration, which can be achieved only by a re-orientation of the platform frame. With regard to the position error, illustrated in the fifth plot from the top, it is possible to observe that the negative peaks are more pronounced with respect to the positive ones. This asymmetry is caused by the lateral force disturbance acting on the platform due to the presence of the serial data cable, which pulls the robot in a more severe way towards the positive direction of the x-axis. The very same outcome can be consistently recognized also in the corresponding plot of Fig. 7.27, since the cable configuration remains unchanged throughout the experiments. Apart from the contribution of the external disturbance, the inexact position tracking is also a side effect of the unfeasible flat orientation given as reference to the predictive controller.

The velocities of the MRAV rotors, whose plot is illustrated in the bottom of Fig. 7.22, are centered on the mean value needed to compensate the gravity force while the aerial vehicle is hovering. The small offset between the velocity of rotors 1-3 and 2-4 suggests that the serial cable also generates a small clockwise torque around the z-axis, which is balanced in order to keep the platform aligned with the yaw reference.

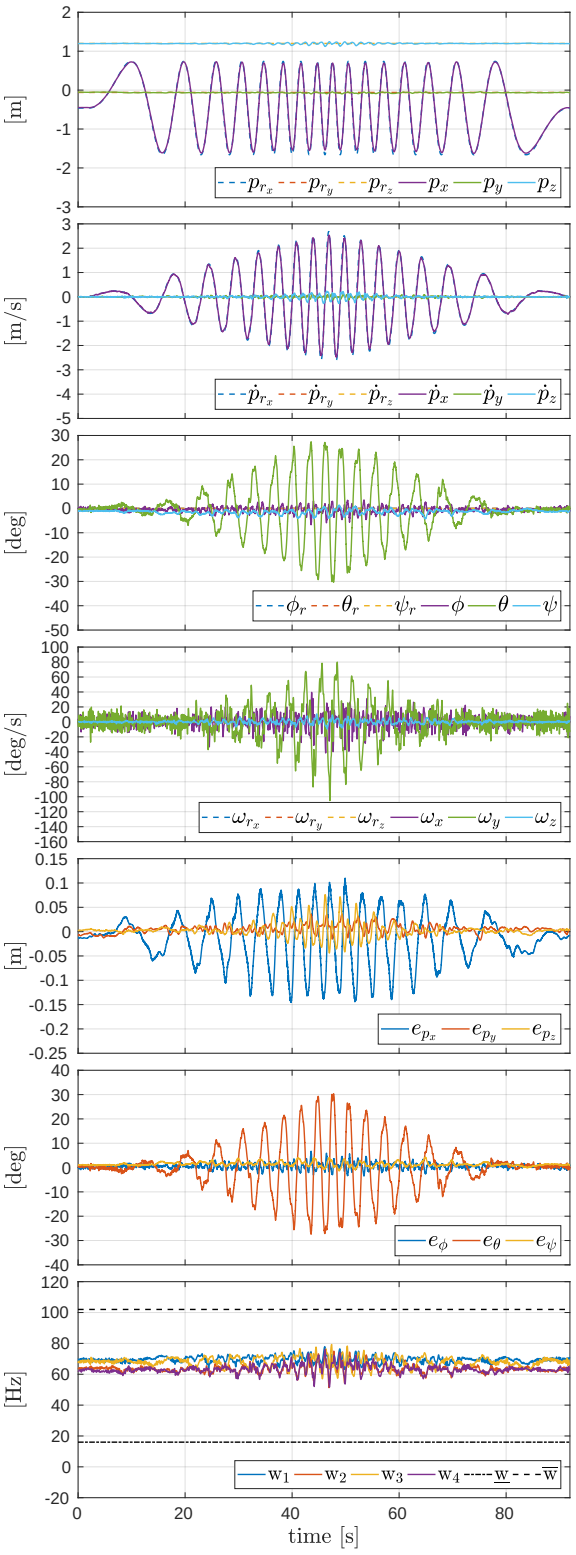


Figure 7.22 – Plots of the quadrotor performing a chirp trajectory on the x-axis. *From top to bottom:* the position, linear velocity, orientation and angular velocity tracking, the position and orientation errors, and the actuator spinning velocities.

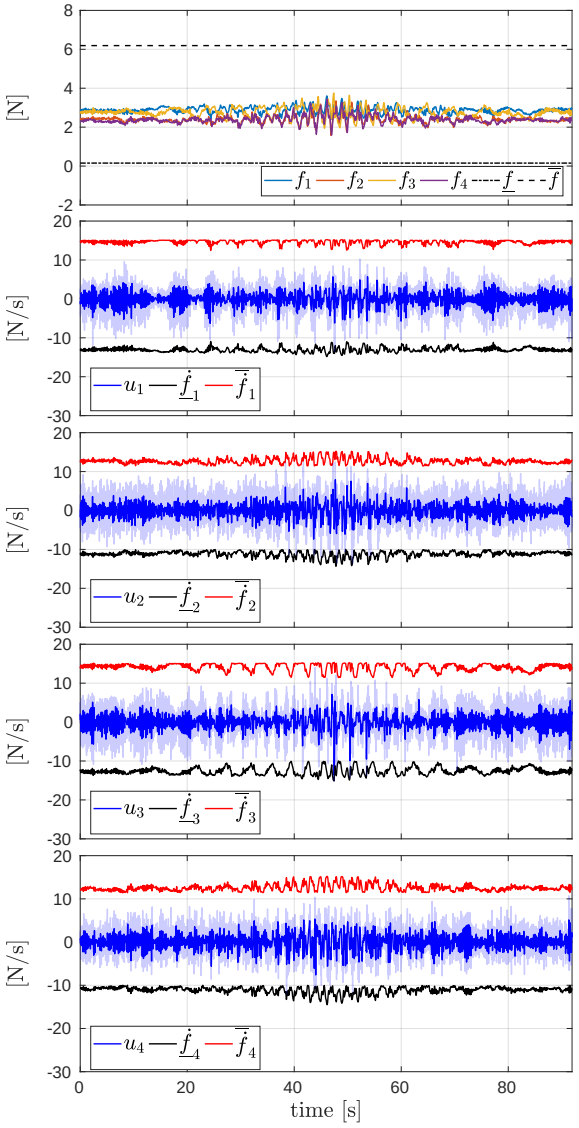


Figure 7.23 – Plots of the quadrotor performing a chirp trajectory on the x-axis. *From top to bottom:* the actuator forces and their derivatives. In particular, all the signals remain inside the feasible region delimited by the identified constraints. Notably, the noisy references u_i are overlapped by their filtered profiles.

In particular remark the fact that, even if the trajectory is rapidly-varying (with a linear acceleration peak of $5.85\,\mathrm{m/s^2}$), the rotor velocities (equivalently their produced forces, presented in the first of plot of Fig. 7.23) take values close to the hovering set-point, without the need to span a large set of values. As already mentioned, this happens because the body torque needed to re-orient the aerial vehicle requires just small differences between the rotor spinning rates. As a consequence, in this experiment the actuator force derivatives do not need to assume large values. This intuition is confirmed by the plots 2-5 of Fig. 7.23, which show that the input components u_i, represented in blue, remain distinctively far from their lower and upper bounds, drawn in black and red, respectively. This evidence suggests that in the case of UDT MRAVs, the limits on the input and on the state components related to the actuator forces can be reached only with rapidly-varying trajectories, designed in order to produce sudden changes in the rotor commands. This motivated the next experiment.

Discontinuous trajectory

Since in the chirp experiment the input limits were far from being approached, we designed a discontinuous trajectory to test the controller stability and its compliance to the actuator constraints in a critical case. For this purpose, we generated as position reference signal a sequence of steps from an initial position $\mathbf{p}_1$ to a final one $\mathbf{p}_2 = \mathbf{p}_1 + \boldsymbol{\epsilon}_\mathbf{p}$, with $\boldsymbol{\epsilon}_\mathbf{p} = [-0.5\,0.2\,0.2]^\top$. On the other hand, all the other reference profiles have been set to zero. In this way, the vehicle was always required to reach the next hovering configuration, with an horizontal attitude and zero translational and rotational velocities, in a short time. Moreover, in order to make the experiment even more challenging, we limited on purpose the predictive capability of the controller, i.e., the NMPC algorithm was made aware about the transitions in the position reference only at the time in which such changes effectively occurred. This strategy emulated an unforeseen event against which the algorithm had to promptly and safely react. In this way, the instantaneous appearance of a consistent error in the controller easily pushed the actuator commands towards their limitations. In this experiment, the identified input constraints on the actuator force derivative have been re-scaled with gains ρ, taking values in $[0.125, 1.75]$, spanning from very conservative - obtained with $\rho = 0.125$ - to larger than the identified ones - obtained with $\rho = 1.75$. The input limits in the controller have been manually increased by the operator by means of a joystick connected to the ground station. This allowed us to empirically assess the validity of the bounds resulting from our identification.

The tracking results related to the trajectory of this specific experiment are shown in Fig. 7.24, where the yellow region highlights the time interval in which the enforced limits correspond exactly to the ones previously identified. The position tracking, depicted in the first plot from the top, shows that very conservative bounds for the actuators, i.e., $\rho \in [\frac{1}{8}\ \frac{1}{4}]$, cause step responses with a remarkable settling time and extended oscillations. Furthermore, the reduced capability to produce a change in the actuator forces seems to affect the tracking of the yaw, that has a non-negligible error for low values of ρ. As already ascertained in the previous

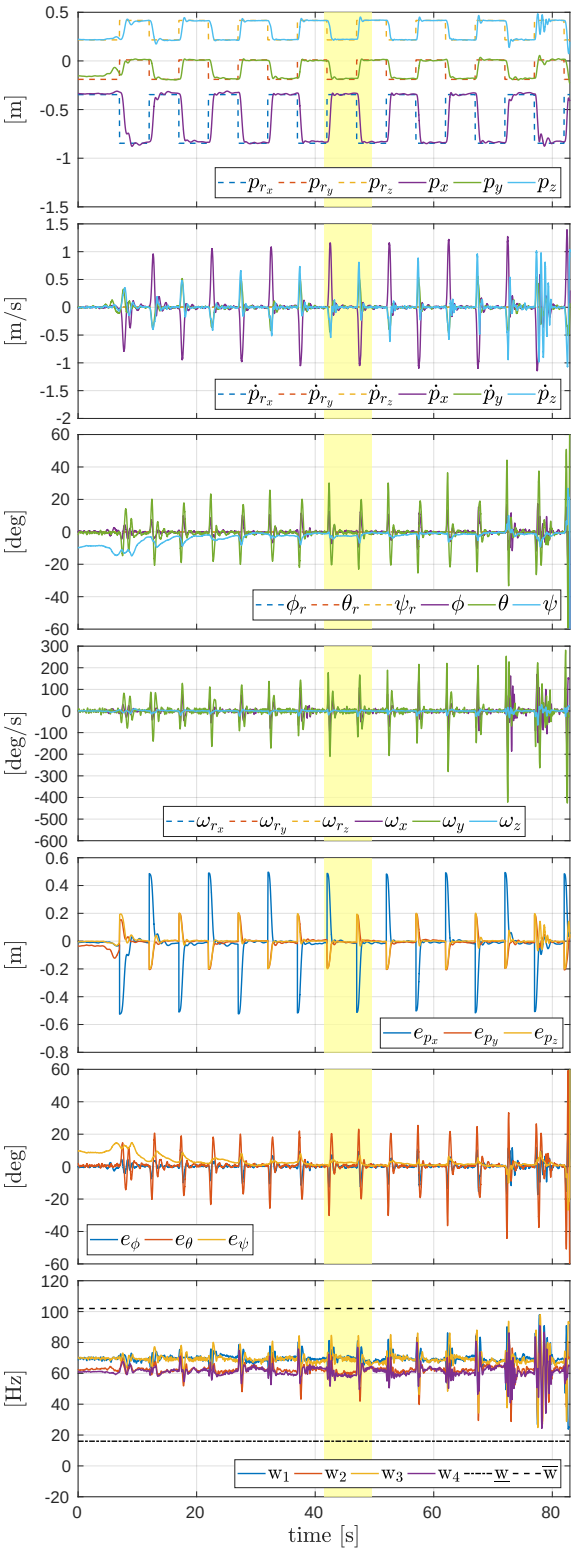


Figure 7.24 – Plots of the quadrotor tracking a discontinuous trajectory, while the controller limits are increased (the yellow region highlights the use of the identified ones). *From top to bottom:* the position, linear velocity, orientation and angular velocity tracking, the position and orientation errors, and the actuator spinning velocities.

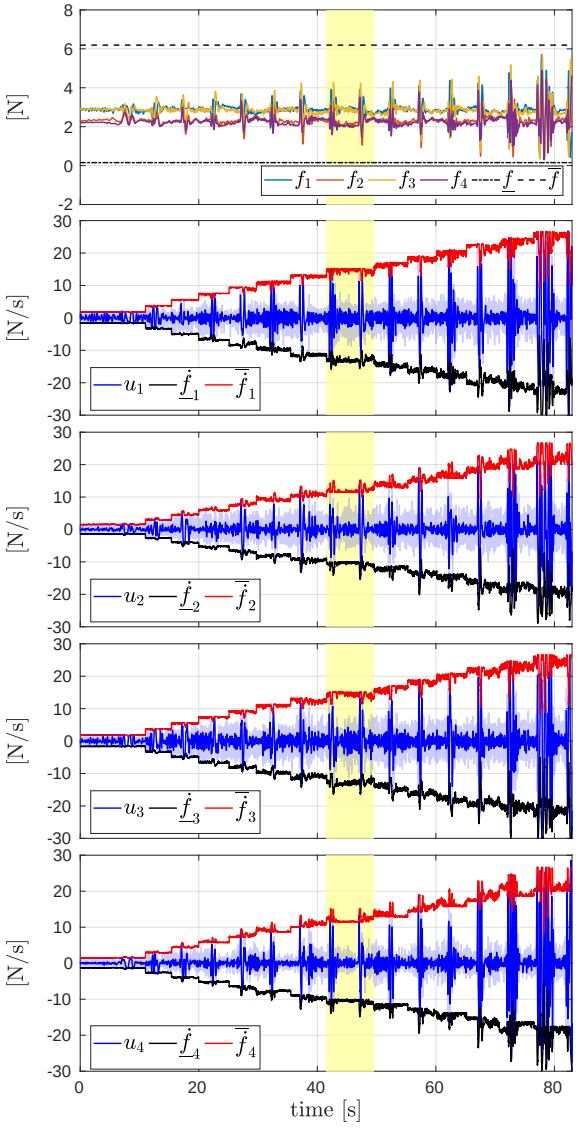


Figure 7.25 – Plots of the quadrotor tracking a discontinuous trajectory with steps in the position, while the controller limits are increased (the yellow region highlights the use of the identified ones). *From top to bottom:* the actuator forces and their derivatives. In particular, all the signals remain inside the feasible region delimited by the constraints.

Table 7.4 – Parameters used in the Tilt-Hex experiment.

Parameter	Value	Unit
ν	1.2	m
ξ	0.025	$\frac{\text{rad}}{\text{s}^2}$
$\bar{t}$	44.84	s
$\mathbf{Q}_{\mathbf{p}}(j,j)\vert_{j=1,2,3}$	$500, 200, 200$	[ ]
$\mathbf{Q}_{\dot{\mathbf{p}}}(j,j)\vert_{j=1,2,3}$	$25, 20, 20$	[ ]
$\mathbf{Q}_{\boldsymbol{\eta}}(j,j)\vert_{j=1,2,3}$	$10, 6, 10$	[ ]
$\mathbf{Q}_{\boldsymbol{\omega}}(j,j)\vert_{j=1,2,3}$	$0.5, 0.5, 0.5$	[ ]
$\mathbf{Q}_{\ddot{\mathbf{p}}}(j,j)\vert_{j=1,2,3}$	$0.01, 0.01, 0.01$	[ ]
$\mathbf{Q}_{\dot{\boldsymbol{\omega}}}(j,j)\vert_{j=1,2,3}$	$0, 0, 0$	[ ]

experiment, this disturbance is induced by the communication cable. On the other hand, the oscillations in the step responses result much more restrained as soon as the control saturations approach the identified ones. Nevertheless, an additional increase in the control bounds imply growing overshoots, especially on the z-axis. Ultimately, the instability is reached at $t \approx 84\,\text{s}$, when $\rho = \frac{7}{4}$. In this moment, the associated limits become almost the double of the identified ones and they induce the platform to reach a configuration from which it was not able to recover. This is confirmed by the plot of the orientation error, where the pitch error reaches almost $e_\theta = 60\,\text{deg}$. The MRAV instability, which causes the experiment to abort, can be particularly well appreciated from the multimedia attachment. Regarding this point, it is worthwhile to make some considerations. First, the tracking results suggest the identified limits to be suitable to ensure the platform stability, also in such a critical experiment. Moreover, this is true within some robustness margin, which was sought in order to avoid an excessive stress for the motor currents. Finally, the plots of Fig. 7.25 deserve a particular attention. With reference to the first one, we can observe that the aerial vehicle becomes unstable even if the actuator forces never reach their limitations, even when instability finally happens. On the other hand, we see from the other plots that their derivatives closely approach the lower and upper bounds. This fact suggests that, neglecting the constraints on the force derivatives, as done in other works, may jeopardize, not only the system performances, but also its stability properties.
The video related to these two experiments is available in [video04–2019]. The same video presents a comparison between the motion of the quadrotor motion and the one of the Tilt-Hex, performing the same experiments, as discussed in the following.

Tilt-Hex experiments

Compared to the quadrotor model, the one of the Tilt-Hex is characterized by two more state and input components to describe the dynamics related to the presence

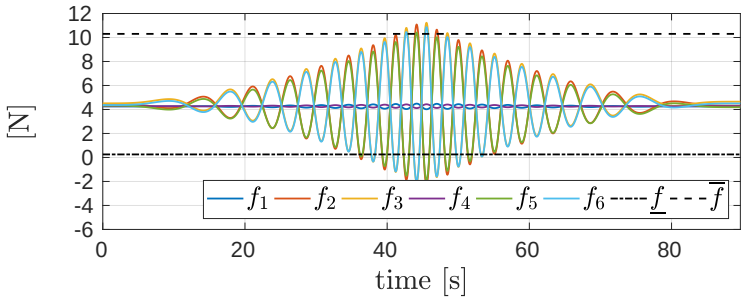


Figure 7.26 – Desired profile for the actuators forces obtained inverting the model dynamics. Such trajectory results unfeasible also w.r.t. the Tilt-Hex limitations.

of the additional actuators. Therefore $\mathbf{x} \in \mathbb{R}^{18}$ and $\mathbf{u} \in \mathbb{R}^6$. With this configuration, the average NMPC solver time is $t_{\text{solv}} = 4.1\,\text{ms}$. In the validation campaign, we made the Tilt-Hex track both the trajectories presented in the previous experiments. The values for the cost function diagonal matrices used in this experiment are reported in Tab. 7.4.

Position chirp trajectory

Thanks to the tilting of its actuators, the Tilt-Hex can exert a 3D set of forces which is not anymore restrained to the body-frame z-axis. In particular, the polytope of forces with zero moment, computed in compliance with all the admissible actuator forces, can be appreciated in the left plot of Fig. 7.3. Thanks to this feature, the vehicle can track decoupled references in position and orientation. However, despite this additional capability, the Tilt-Hex cannot track any decoupled trajectory, due to the limitations still present in the actuators. In particular, he previously defined chirp trajectory has been generated with the goal to be unfeasible also with respect to the Tilt-Hex actuation capabilities. The evolution of the actuator forces, in the ideal case in which no noise or disturbance is present, can be found by plugging the desired trajectory and the physical parameters in (3.19). As shown in Fig. 7.26, the desired actuator force trajectories, obtained via such dynamic inversion, are not compliant with respect to the lower and upper bounds. This means that, also in this case, a new feasible trajectory has to be re-computed by the NMPC strategy. Nevertheless, we expect to obtain improved tracking performances compared to the quadrotor experiment.

The plots related to the trajectory tracking for this experiment are shown in Fig. 7.27. As shown on the two top sub-figures, the translational references are followed in a more precise way compared to Fig. 7.22. In particular, this is true also around the central peaks, which correspond to the most rapidly-varying part of the trajectory, i.e., where the lateral acceleration takes the largest values. From the third and the fourth plots it can be observed that the deviations from the orientation and the angular velocity references are significantly reduced with respect to the ones produced by the quadrotor with the very same trajectory. Such remarkable improvement is a direct consequence of the benefits induced by the multi-directionality of the thrust. On the other hand, also the position error is consistently reduced, with a maximum peak of 6.4 cm (in absolute value) against the

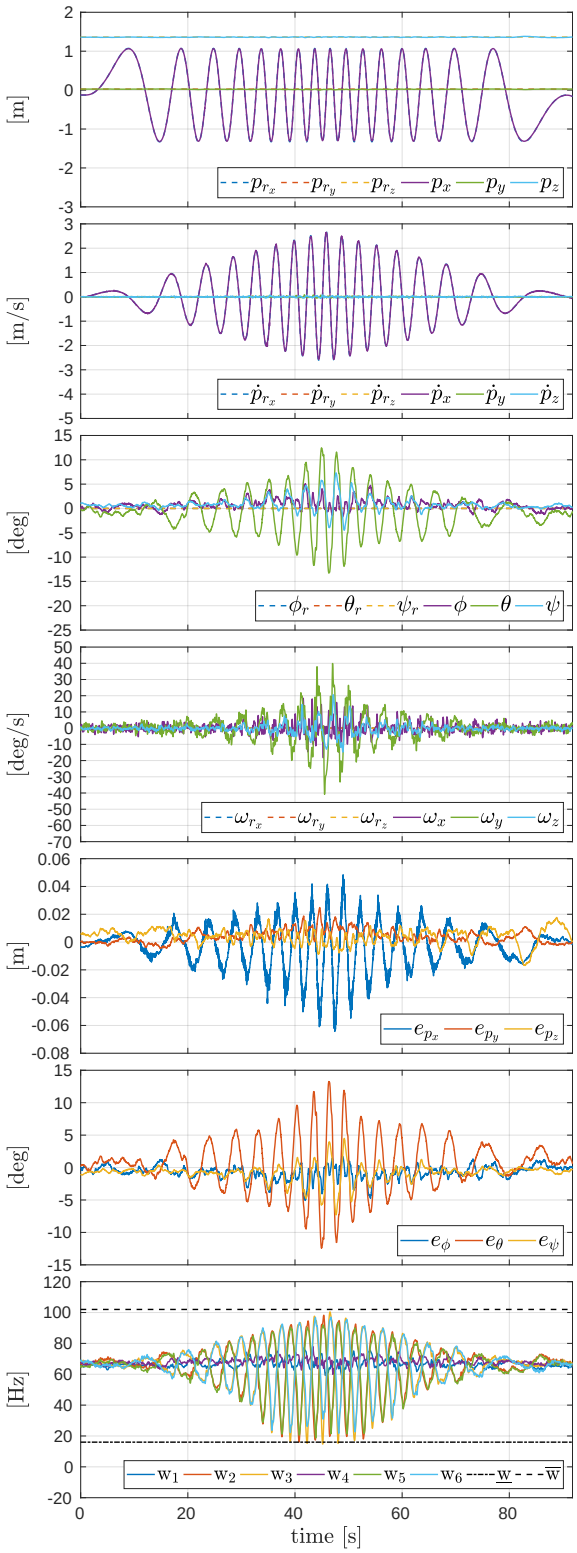


Figure 7.27 – Plots of the Tilt-Hex performing a chirp trajectory on the x-axis. *From top to bottom:* the position, linear velocity, orientation and angular velocity tracking, the position and orientation errors, and the rotor velocities.

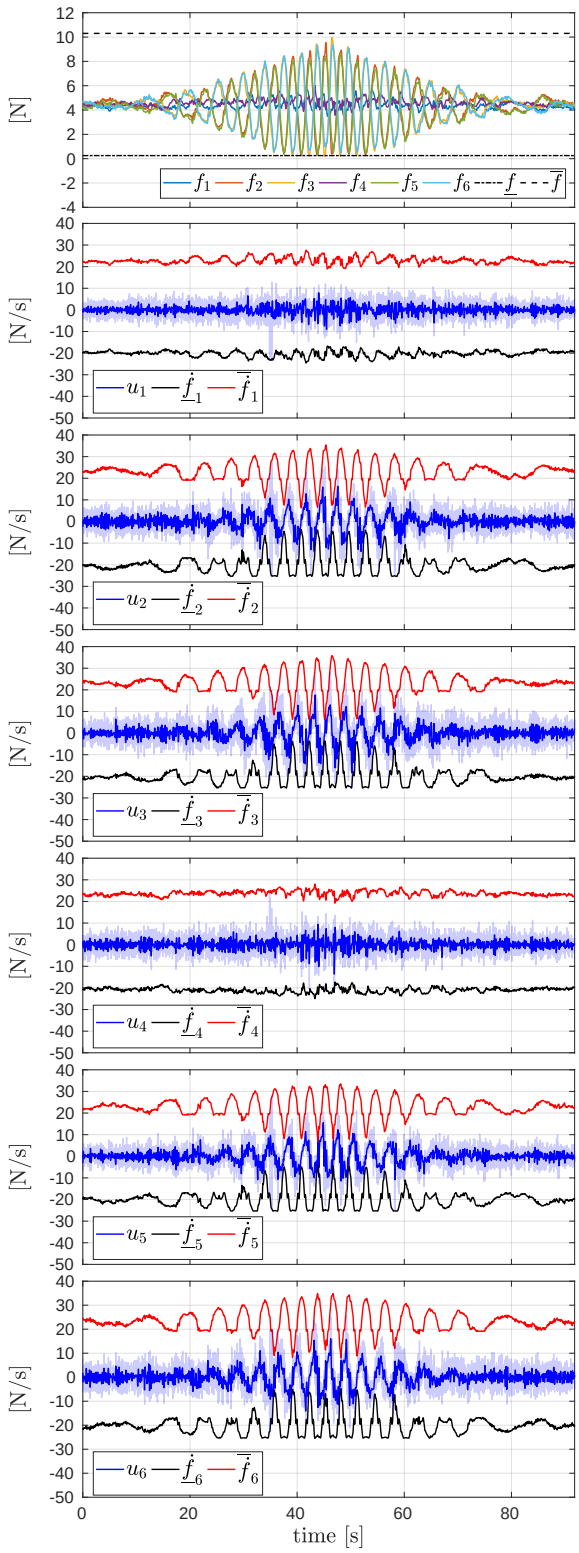


Figure 7.28 – Plots of the Tilt-Hex performing a chirp trajectory on the x-axis. *From top to bottom:* the actuator forces and their derivatives. In particular, all the signals remain inside the feasible region delimited by the constraints.

14.5 cm of the quadrotor experiment. This suggests that the full actuation also helps improving the position tracking. With reference to the fifth plot, the systematic small asymmetry in the position error is caused again by the cable disturbance.

As far as the actuator data are concerned, consider the last plot of Fig. 7.24 and the bottom one in Fig. 7.27. The rotor velocities in the second case span the feasible set in a wider way. While in the quadrotor experiment the velocity constraints are not even approached, in the Tilt-Hex case they become frequently active. In the specific case, the fact that the lower bounds are reached more often than the upper ones is simply due to the platform mass. Indeed, from the first plot of Fig. 7.28 we can see that the mean hovering value per actuator is approximately 4N, which is closer to the lower saturation level. On the other hand, the velocities of a more massive vehicle would have approached more easily the upper part of the plot.

We now shortly compare the results achieved by this NMPC algorithm with the ones obtained by the reactive LBF controller. In particular, Fig. 7.5 presents the tracking results related to the same trajectory and the same MRAV. Regarding the position error, we obtained slightly better performance with the NMPC regulator, mostly in the two lateral tails of the trajectory (where the error is always bounded within 4 cm). Furthermore, while the error in the that experiment was more or less uniformly distributed along the trajectory, in the present case its trend seems to be proportional to the chirp frequency, which also has a triangular envelope. This effect could be explained by the predictive nature of the NMPC algorithm. Indeed, while the reactive regulator always acts in relation to the instantaneous value of the desired trajectory, the NMPC response is affected by the future evolution of the former, which depends on the chirp frequency. As far as the orientation tracking is concerned, a relevant improvement is achieved. Indeed, the maximum pitch error is reduced from 23 deg to 13 deg, i.e., a decrease of more than 43%. Furthermore, analyzing the plot of the rotor velocities we see that now they evolve in a larger range, meaning that the NMPC regulator is exploiting the actuator capabilities in a more efficient way. This is a consequence of the fact that the previous controller deals with a less precise – and more conservative – model of the platform.

Discontinuous trajectory

To assess the effectiveness of our procedure in identifying meaningful actuator limitations for a non-specific hardware setup, we replicated the second experiment described in (a) using the Tilt-Hex robot. The plots related to this test are depicted in Fig. 7.29 and in Fig. 7.30. For this experiment, the limits to the NMPC were scaled by the user after two consecutive jumps of the MRAV. The experiment outcomes show that the best step responses are achieved when the actuator limits are closer to the identified ones. This confirms again the validity of our approach. Furthermore, also in this case, the instability is reached when $\rho = \frac{7}{4}$, i.e., when the force derivative bounds are almost the double of the identified ones, thus guaranteeing an adequate safety margin.

As already mentioned, the video presenting both experiments for the Tilt-Hex, comparing its motion to the one of the quadrotor, is available in [video04–2019].

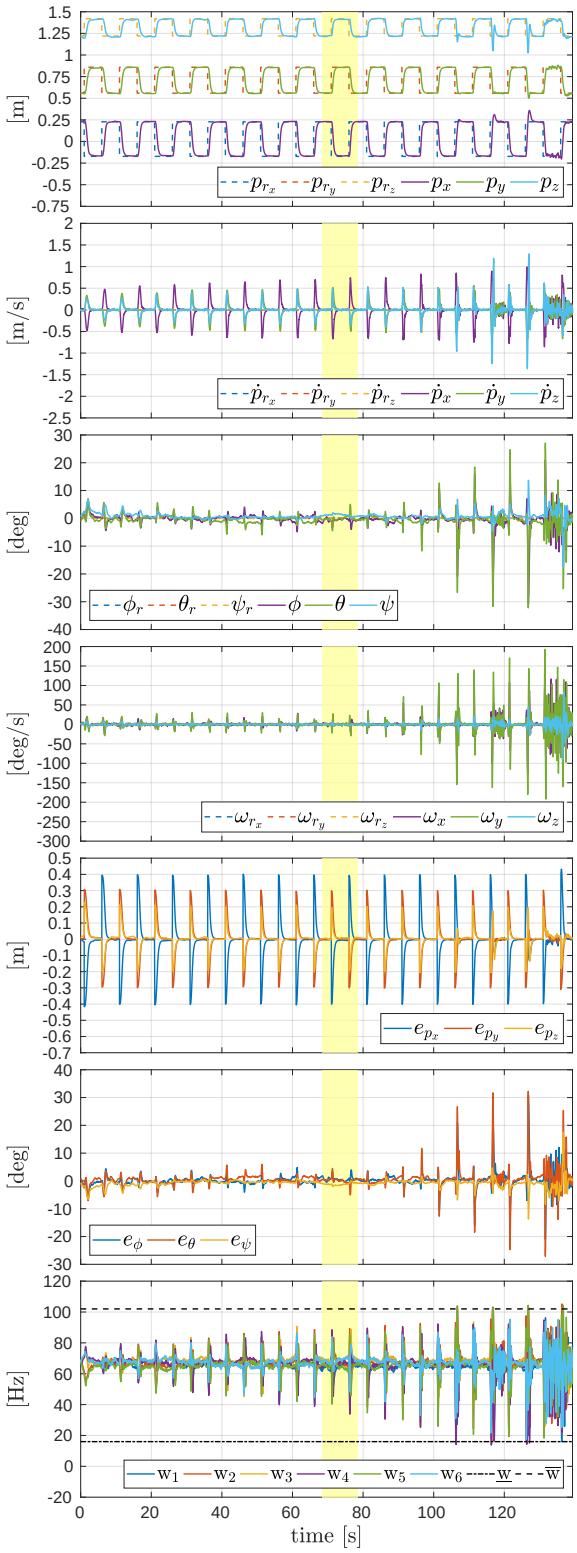


Figure 7.29 – Plots of the Tilt-Hex tracking a discontinuous trajectory, while the controller limits are increased (the yellow region highlights the use of the identified ones). *From top to bottom:* the position, linear velocity, orientation and angular velocity tracking, the position and orientation errors, and the actuator spinning velocities.

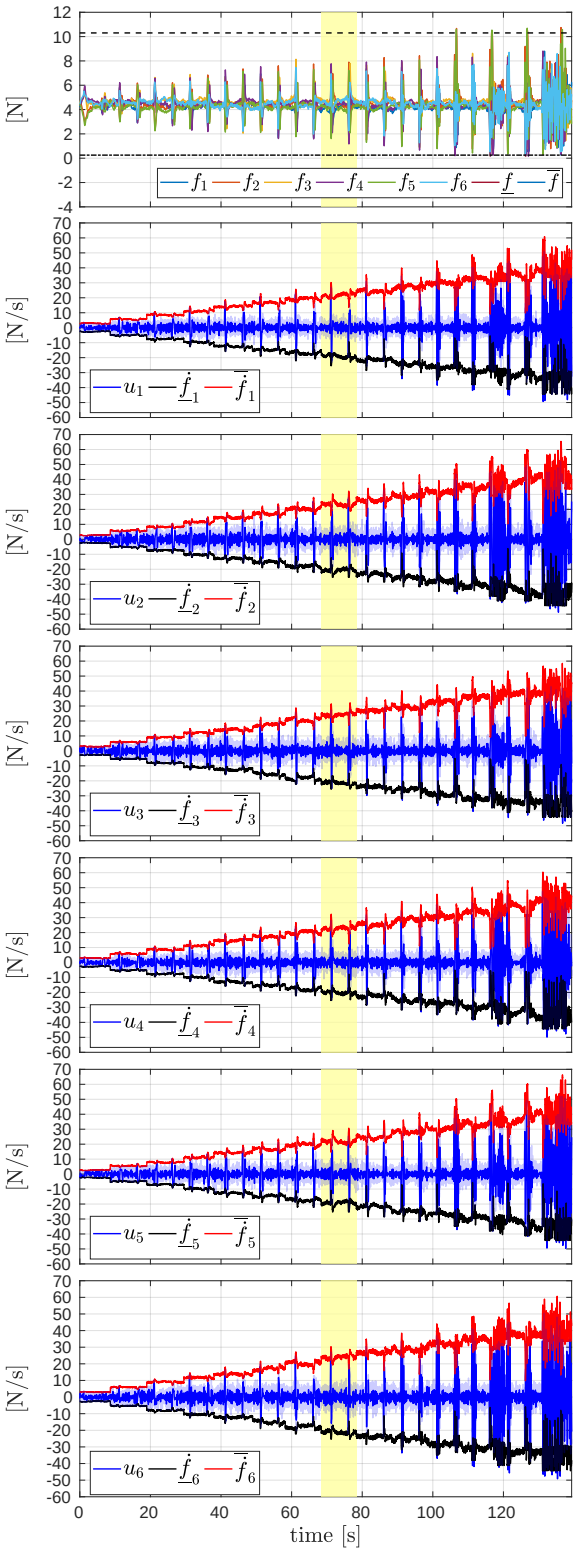


Figure 7.30 – Plots of the Tilt-Hex tracking a discontinuous trajectory with steps in the position, while the controller limits are increased (the yellow region highlights the use of the identified ones). *From top to bottom:* the actuator forces and their derivatives. In particular, all the signals remain inside the feasible region delimited by the constraints.

Figure 7.31 – Pictures of the FAST-Hex (*left*) and the Tilt-Hex with rotor failure (*right*) used in the numerical validation.

Additional numerical validation

To demonstrate that the proposed NMPC framework can deal with a very generic MRAV design, we provide additional numerical validations with two other different vehicles, shown in Fig. 7.31. The first one, shown on the left, is the FAST-Hex platform described in Sec. 6.1. The second vehicle, shown on the right, is a pentarotor (a multi-rotor with five propellers) obtained as a failed Tilt-Hex MRAV, i.e., the platform already described in the experimental validation, but after a rotor failure. In particular the $6-th$ rotor is not allowed to spin, due to, e.g., a technical problem, and cannot exert a thrust force and generate a drag torque. For this reason, from a control point of view, we will consider that such actuator is not present. The rotor failure essentially modifies the available set of body forces and torques. As already pointed out in previous contributions, in case $\alpha = \beta = 0$ it is not possible with five uni-directional actuators to generate torques in pitch and roll without generating a residual disturbing torque in the yaw axis [Giribet–2016b; Achtelik–2012]. In the more general case in which $(\alpha, \beta) \neq (0, 0)$, however, the platform maintains the ability to hover [Giribet–2016b]. Nevertheless, the hovering orientation can not be flat any more, and depends of the actuator tilting angles [Michieletto–2017]. We will show that the NMPC controller can satisfactorily deal with the problem of static hovering, without the need to a-priori compute the steady-state orientation. For the value of the physical parameters of the two robots, the reader is referred to Tab. 6.1 and Tab. 6.2, respectively.

FAST-Hex simulations

In order to take into account the evolution of the angle α and, in particular, to let the NMPC algorithm manage its automatic regulation, we expanded the state and the input vector, defined in (4.14) and in (4.15), in the following way

$$\mathbf{x} := \left[\mathbf{p}^\top \dot{\mathbf{p}}^\top \boldsymbol{\eta}^\top \boldsymbol{\omega}^\top \boldsymbol{\gamma}^\top, \alpha\right]^\top \tag{7.17}$$

$$\mathbf{u} := \left[\dot{\boldsymbol{\gamma}}, \dot{\alpha}\right]. \tag{7.18}$$

The angle α is now a component of the state vector, while $\dot{\alpha}$ is regarded as an additional control input. This allows to constrain the synchronous tilting angle and its derivative within their feasible sets, computed accordingly to the data of the real MRAV prototype designed in [Ryll–2016] and presented in Sec. 6.1. According to this choice, for the FAST-Hex model we have $\mathbf{x} \in \mathbb{R}^{19}$ and $\mathbf{u} \in \mathbb{R}^{7}$.

As it can be appreciated from Fig. 3 in [Ryll–2016], the larger the angle α, the larger the set of body-frame lateral forces. This translates also into the possibility of decoupling the control of the body force and moment in a larger extent, which becomes particularly useful in many realistic scenarios, ranging from 6D trajectory tracking [Franchi–2018] to aerial physical interaction tasks [Staub–2018a; Ryll–2017] and disturbance rejection in general. On the other hand, the increase of the tilting angle implies also an increment in the energy consumption. In fact, the progressive decrease in the projection of the thrust vector along $\mathbf{z}_W$ must be compensated by an increase in the thrust intensity. In view of these considerations, it might be beneficial to regulate the angle α with respect to the particular task to be accomplished, while trying to minimize the energy consumption. In order to fulfill this requirement, we expanded also the output vector defined in 5.26 as follows

$$\mathbf{y}(t) = \mathbf{h}\left(\mathbf{x}(t), \mathbf{u}(t)\right) = \begin{bmatrix} \mathbf{p}(t) \\ \dot{\mathbf{p}}(t) \\ \ddot{\mathbf{p}}\left(\mathbf{x}(t), \mathbf{u}(t)\right) \\ \boldsymbol{\eta}(t) \\ \boldsymbol{\omega}(t) \\ \dot{\boldsymbol{\omega}}\left(\mathbf{x}(t), \mathbf{u}(t)\right) \\ c_e\left(\mathbf{x}(t), \mathbf{u}(t)\right) \end{bmatrix} \tag{7.19}$$

where the cost related to power consumption is taken into account using the following additional cost

$$c_e\left(\mathbf{x}(t), \mathbf{u}(t)\right) = \sum_{i=1}^{n} f_i^2 \tag{7.20}$$

which is integrated along the prediction horizon. Such model has been chosen mainly due to its simple form with respect to the state components f_i.

In this context, we target the classical problem of trajectory regulation to a certain 6D configuration, i.e., the flat hovering, adding the effect of an external unknown disturbance from the environment, which emulates, in a simplified but meaningful way, the scenario of a physical interaction task or an external wind. In the first simulation, we exploit the possibility of regulating α. In this way we show how the NMPC algorithm can automatically and actively manipulate the additional control input $\dot{\alpha}$. Furthermore, in order to demonstrate the usefulness of this supplementary degree of freedom, we present the results of the same simulation, where the tilting angle α is instead forced to assume different fixed values.

Table 7.5 – Parameters used in the FAST-Hex simulation.

Parameter	Value	Unit
$\boldsymbol{\sigma}_{\mathbf{p}}$	$[\sqrt{0.005}\ \sqrt{0.005}\ \sqrt{0.005}]^\top$	m
$\boldsymbol{\sigma}_{\dot{\mathbf{p}}}$	$[\sqrt{0.02}\ \sqrt{0.02}\ \sqrt{0.02}]^\top$	m/s
$\boldsymbol{\sigma}_{\boldsymbol{\eta}}$	$[\sqrt{1}\ \sqrt{1}\ \sqrt{1}]^\top$	deg
$\boldsymbol{\sigma}_{\boldsymbol{\omega}}$	$[\sqrt{0.15}\ \sqrt{0.15}\ \sqrt{0.05}]^\top$	deg/s
Ω_{filt}	25	rad/s
t_1	10	s
t_2	20	s
$\mathbf{f}_{\text{dist}}(t_2)$	$3\ [cos(\frac{\pi}{3})\ sin(\frac{\pi}{3})\ 0]^\top$	N
$\underline{\alpha}$, $\overline{\alpha}$	-35 , 35	deg
$\underline{\dot{\alpha}}$, $\overline{\dot{\alpha}}$	-8.75 , 8.75	deg/s
$\mathbf{Q}_{\mathbf{p}}(j,j)\vert_{j=1,2,3}$	$50, 50, 50$	[ ]
$\mathbf{Q}_{\dot{\mathbf{p}}}(j,j)\vert_{j=1,2,3}$	$0.5, 0.5, 0.5$	[ ]
$\mathbf{Q}_{\boldsymbol{\eta}}(j,j)\vert_{j=1,2,3}$	$15, 15, 15$	[ ]
$\mathbf{Q}_{\boldsymbol{\omega}}(j,j)\vert_{j=1,2,3}$	$0.01, 0.01, 0.01$	[ ]
$\mathbf{Q}_{\bar{\mathbf{p}}}(j,j)\vert_{j=1,2,3}$	$0.0001, 0.0001, 0.0001$	[ ]
$\mathbf{Q}_{\dot{\boldsymbol{\omega}}}(j,j)\vert_{j=1,2,3}$	$0, 0, 0$	[ ]
Q_{e_c}	0.0005	[ ]

To make the simulations more realistic, we added to the measured state a noise, obtained filtering a zero-mean white Gaussian noise with a first-order causal low-pass filter having a cut-off frequency Ω_{filt}, whose value has been estimated analyzing real experimental data. The noise standard deviation values $\boldsymbol{\sigma}_{\bullet}$ are collected in Tab. 7.5, together with the other trajectory parameters, state/input bounds and cost function weights. In particular, the values of $\boldsymbol{\sigma}_{\bullet}$ are related to very unfavorable conditions compared to the use of typical sensors such as MoCap and gyros.

Hovering trajectory with lateral force disturbance

Throughout the presented simulations, the FAST-Hex is required to hover maintaining a flat orientation, i.e.,

$$\mathbf{p}_r = [0.6\ 0.6\ 0.75]^\top,\ \dot{\mathbf{p}}_r = \ddot{\mathbf{p}}_r = [0\ 0\ 0]^\top \tag{7.21}$$

$$\boldsymbol{\eta}_r = \boldsymbol{\omega}_r = \dot{\boldsymbol{\omega}}_r = [0\ 0\ 0]^\top \tag{7.22}$$

under the effect of a lateral force disturbance $\mathbf{f}_{\text{dist}}$ with a triangular profile. Such force, unknown to the controller, has a triangular shape from t_1 to $t_1 + t_2$, with a peak of 3 N at t_2, while it is $\mathbf{f}_{\text{dist}} = \mathbf{0}$ N elsewhere. The reference of the energetic term $c_{e,r}$ is constantly equal to the one needed for hovering horizontally with $\alpha = 0$, i.e., $c_{e,r} = \sum_{i=1}^{n}(\frac{mg}{6})^2$.

As long as the disturbance is not active, the NMPC algorithm should try to maintain α small, ideally equal to zero. This claim is motivated by the fact that this trajectory does not need the MDT capability in order to be tracked. On the other hand, as soon as the lateral force is activated, the platform can react to it either tilting its actuators or re-orienting its chassis. In this choice, the relative

values of the cost function weights play a fundamental role. Intuitively, if the energy cost is weighted consistently (with respect to the tracking error terms on the states), the control algorithm should try to produce an input with low energy consumption, giving less priority to the trajectory tracking. In particular, the task of maintaining a flat orientation should be somehow discouraged by the controller, since the generation of a lateral force in this configuration would require a consistent increase of some of the actuator forces, thus raising up the energy consumption. Conversely, if the weight related to the energy cost is small, the controller would always privilege the trajectory tracking, acting on input α.

In the first simulation, related to the case in which α is actively regulated, we try to achieve a good *trade-off* between the two tasks. In the other simulations, corresponding to different fixed configurations for α, all the parameters are left untouched, in order to fairly compare the resulting performance, in terms of the overall cost function, with respect to the variable case.

The plots related to the trajectory tracking in the variable case are depicted in Fig. 7.32. The first four plots, exhibiting the trajectory tracking of the state components, outline the good performance of the controller. Indeed, the measured linear velocity, orientation, and angular velocity tracking keep very close to their reference profile, which are constantly equal to zero on all components. On the other hand, the measured position visibly deviates from the reference one when the disturbance is acting on the robot. Nevertheless, the position error keeps bounded, with a peak of less than 9 cm on its second component, which corresponds to the direction mostly affected by the lateral force, as shown in the last plot of Fig. 7.34. This error could be considerably reduced by increasing the relative weights inside the NMPC algorithm cost function: this is confirmed by the sixth plot of Fig. 7.32, where the MRAV maintains the orientation error below 1 deg. We were able to achieve such result by properly weighting the attitude term in relation to the others, in particular with respect to the energy cost. Moreover, the last plot of the same figure shows that the external disturbance can be counteracted without an excessive effort of the rotors, since their spinning velocities (and so the generated forces) safely remain with the bounds. The plots related to the force derivatives, which are presented in Fig. 7.33, confirm that a static trajectory, combined with a slowly-varying disturbance, does not produce large values for the inputs.

Consider the first plot of Fig. 7.34, which depicts the trajectory of α. During the middle phase, the tilting angle is increased up to ≈ 21 deg in order to counteract the lateral force and to keep the platform flat at the same time. On the other hand, the reason why α is regulated to a constant value of ≈ 7 deg and not exactly to zero, is due to the noise introduced in the simulation, in particular to the one related to the translational part of the state $[p_x \; p_y \; \dot{p}_x \; \dot{p}_y]^\top$. Indeed, the control algorithm is informed about a non-zero error in these components, and continuously tries to annihilate it by selecting a small tilting angle, in order to be able to exert a lateral force and stay horizontal at the same time. In other words, the predictive controller considers more beneficial to keep a small $\alpha > 0$, in order to minimize the cost function across the horizon. This confirms that the tracking of the orientation

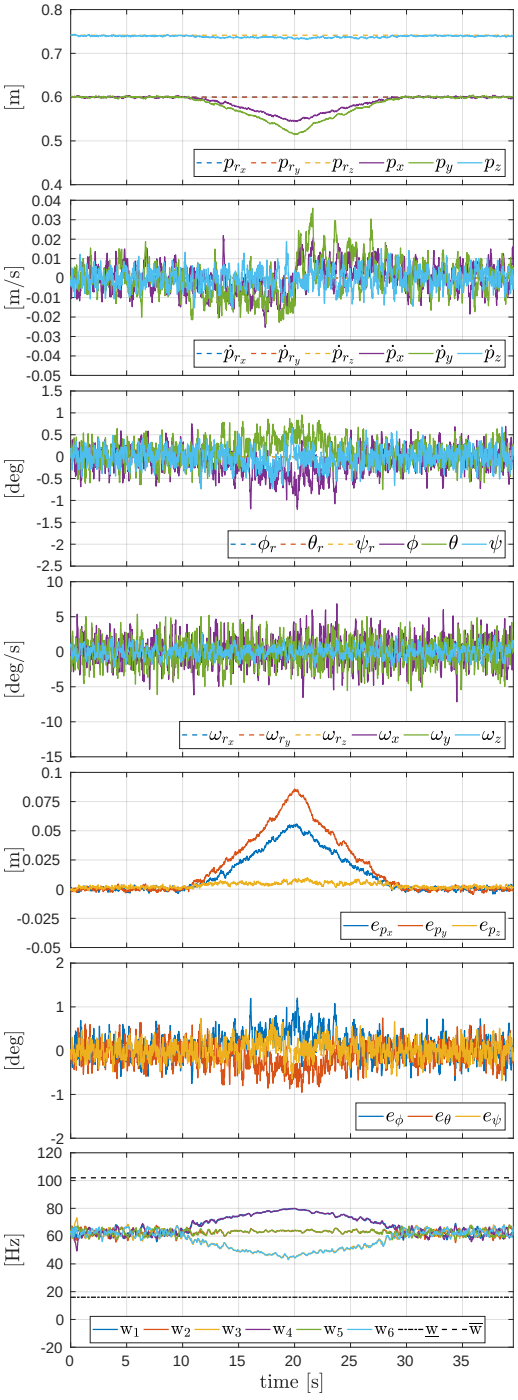


Figure 7.32 – Plots of the FAST-Hex (with variable α regulated from the MPC algorithm) while hovering. The robot is disturbed with a lateral force with a triangular profile. *From top to bottom:* the position, linear velocity, orientation and angular velocity tracking, the position and orientation errors, and the actuator spinning velocities.

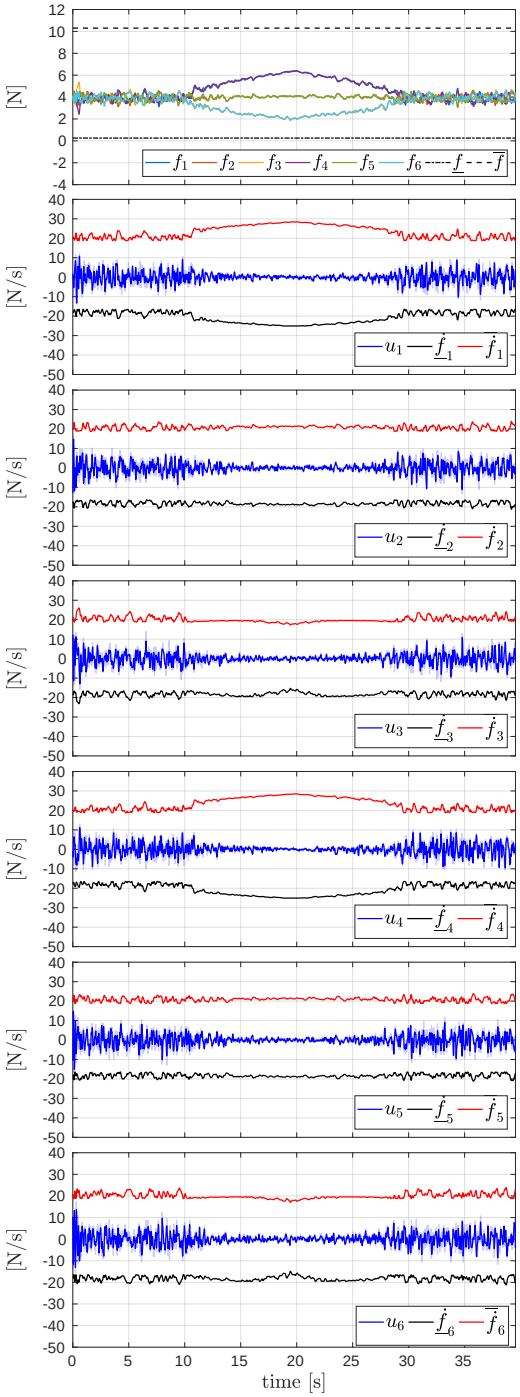


Figure 7.33 – Plots of the FAST-Hex (with variable α regulated from the MPC algorithm) while hovering. The robot is disturbed with an external lateral force with a triangular profile. *From top to bottom:* the actuator forces and their derivatives. In particular, all the signals remain inside the feasible region delimited by the constraints.

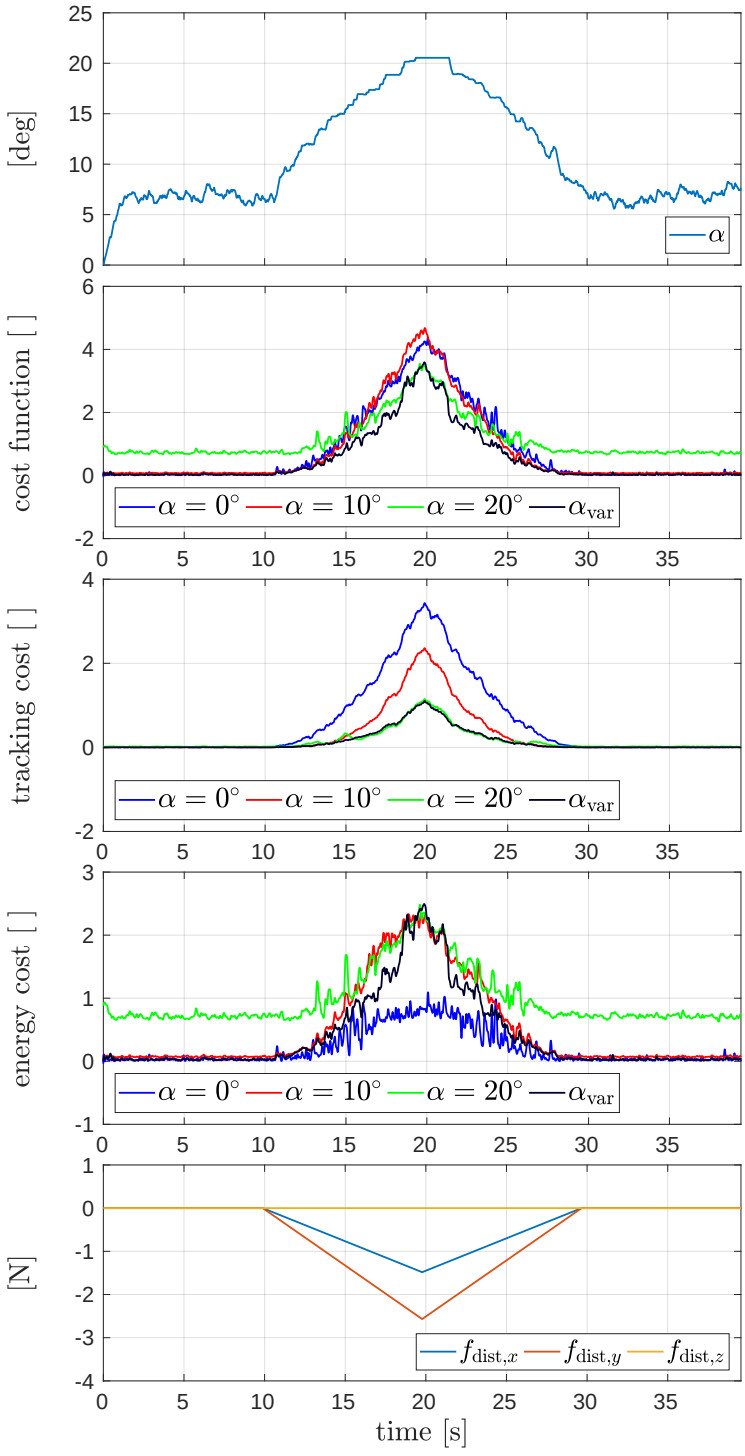


Figure 7.34 – Plots of the FAST-Hex while hovering. In the first two plots, the evolution of α in the variable case and the comparison of the total cost function for different cases of constant α and the variable α (regulated from the NMPC algorithm). Then, the comparison of the partial costs related to the tracking and the energy terms. Finally, the profile of the external force disturbance $\mathcal{F}_W$.

has a high priority with this set of weights.

In order to demonstrate the benefit of the active regulation of the tilting angle, we additionally performed other three simulations (with the same parameters) imposing $\alpha = 0, 10, 20$ deg, respectively. The comparison of the overall NMPC cost functions for the different fixed cases and the variable one is displayed in the second plot of Fig. 7.34. As it can be appreciated, the regulated case, denoted with α_{var}, gives the best trade-off between tracking performance and consumed energy. In the unperturbed hovering phases (lateral parts of the plots), α is regulated to a small value in order to avoid unnecessary energy waste, while in the middle phase, when the disturbance force is activated, α is increased, in order to improve the trajectory tracking, in particular that related to the orientation one. The third and the fourth plots of the same figure outline the partial costs related to the tracking errors and the energy cost. Among the fixed configurations, the one with the largest tilting angle, i.e. $\alpha = 20\,\text{deg}$, generates the smallest tracking cost along all the simulation. This confirms that the MDT capability drastically improves the MRAV tracking performance. On the other hand, it unavoidably causes a larger energy cost, as the angle takes larger values. This is why the additional degree of freedom on α might be very convenient in many applications. Finally, it is worthwhile to point out that, in this case, the active regulation of α globally minimize the overall cost function, even if its optimization is performed locally with an horizon of $t_H = 1\,\text{s}$.

Tilt-Hex with rotor failure simulations

As already mentioned in Sec. 3.2 and in Sec. 6.1, the problem of the robustness of a MRAV in case of a rotor failure is not new in the literature. Indeed, the analysis and the design of a tilted-rotor hexarotor for fault tolerance has been considered in [Giribet–2016b], while formal definitions as well as the design of an analytic controller based on the identification of a direction in the force space, along which the intensity of the control force can be assigned independently from the torque, can be found in [Michieletto–2017; Michieletto–2018]. Given the importance of such topic in the aerial robotics panorama, we decided to target this problem, showing that the presented NMPC algorithm can deal with this problem in a very efficient way.

As described, the failure of one rotor in the Tilt-Hex is modeled removing one state and one input, i.e., the ones related to the $6-th$ actuator. Therefore $\mathbf{x} \in \mathbb{R}^{17}$ and $\mathbf{u} \in \mathbb{R}^5$. In the following, we present the hovering performance results in two different configurations of the angle β (cf. Fig. 6.1), in order to underline the importance of such angle in relation to the fault tolerance system capabilities [Michieletto–2018]. This represents a motivation towards the design of Tilt-Hex angles, discussed in Sec. 6.1. The parameters related to these simulations are reported in Tab. 7.6.

As already pointed out by [Michieletto–2017], given the particular arrangement of the Tilt-Hex actuators, which are symmetrically disposed in a star-configuration with alternated α and equal β angles, it is convenient to switch off the actuator located in the mirrored position with respect to the broken one, when the failure is

Table 7.6 – Parameters used in the Tilt-Hex rotor failure simulation.

Parameter	Value	Unit
$\boldsymbol{\sigma_{\mathbf{p}}}$	$[\sqrt{0.005}\ \sqrt{0.005}\ \sqrt{0.005}]^\top$	m
$\boldsymbol{\sigma_{\dot{\mathbf{p}}}}$	$[\sqrt{0.02}\ \sqrt{0.02}\ \sqrt{0.02}]^\top$	m/s
$\boldsymbol{\sigma_{\eta}}$	$[\sqrt{1}\ \sqrt{1}\ \sqrt{1}]^\top$	deg
$\boldsymbol{\sigma_{\omega}}$	$[\sqrt{0.15}\ \sqrt{0.15}\ \sqrt{0.05}]^\top$	deg/s
Ω_{filt}	25	rad/s
t_1	5	s
$\boldsymbol{\tau}_{\text{dist}}$	$\frac{1}{250}\ [0.68\ 0.39\ 0.62]^\top$	Nm
$\mathbf{Q}_{\mathbf{p}}(j,j)\vert_{j=1,2,3}$	$10, 10, 10$	[ ]
$\mathbf{Q}_{\dot{\mathbf{p}}}(j,j)\vert_{j=1,2,3}$	$0.5, 0.5, 0.5$	[ ]
$\mathbf{Q}_{\boldsymbol{\eta}}(j,j)\vert_{j=1,2,3}$	$1.5, 1.5, 1.5$	[ ]
$\mathbf{Q}_{\boldsymbol{\omega}}(j,j)\vert_{j=1,2,3}$	$0.0005, 0.0005, 0.0005$	[ ]
$\mathbf{Q}_{\ddot{\mathbf{p}}}(j,j)\vert_{j=1,2,3}$	$0, 0, 0$	[ ]
$\mathbf{Q}_{\dot{\boldsymbol{\omega}}}(j,j)\vert_{j=1,2,3}$	$0, 0, 0$	[ ]

detected. In this case, this corresponds to the $3-rd$ one. This represents the best solution in order to balance the control effort (c.f. [Michieletto–2017], Fig. (3)). In the following simulations, this behavior is emulated by setting $\underline{f} = 0$N, i.e., letting the controller the possibility to completely switch off the actuators.

Hovering trajectory with torque disturbance

In this case, the reference trajectory is again a static hovering, i.e.,

$$\mathbf{p}_r = [0\ 0\ 0.75]^\top,\ \dot{\mathbf{p}}_r = \ddot{\mathbf{p}}_r = [0\ 0\ 0]^\top \tag{7.23}$$

$$\boldsymbol{\eta}_r = \boldsymbol{\omega}_r = \dot{\boldsymbol{\omega}}_r = [0\ 0\ 0]^\top \tag{7.24}$$

In order to make the simulations even more realistic, in addition to the already introduced measurement noise, we add a torque disturbance to the platform, whose magnitude can be compared to typical values that one could experience in a real experiment due to parameter mismatches and/or external perturbations. The way this torque $\boldsymbol{\tau}_{\text{dist}}$ is computed deserves some explanations. In the case $\beta = 0$, when both the $6-th$ and the $3-rd$ actuators are switched off, the moments generated by the other four propellers lie all on a 2-dimensional plane (c.f. [Michieletto–2018], Fig. (3)). This can be verified by analyzing the rank of the allocation sub-matrix ${}^3\mathbf{G}_2^6 = \mathbf{G}_2(:, 1, 2, 4, 5)$, i.e., the sub-part related to the torque actuation deprived of the columns related to the actuators which are broken ($6-th$) and off ($3-rd$), respectively. At this point, we select the normal to such plane by finding an orthonormal base $\{\mathbf{v}_1\ \mathbf{v}_2\}$ for the column span of ${}^3\mathbf{G}_2^6$ and operate the cross product $\mathbf{v}_3 = \mathbf{v}_1 \times \mathbf{v}_2$. This unit vector indicates the direction of the most unfavorable

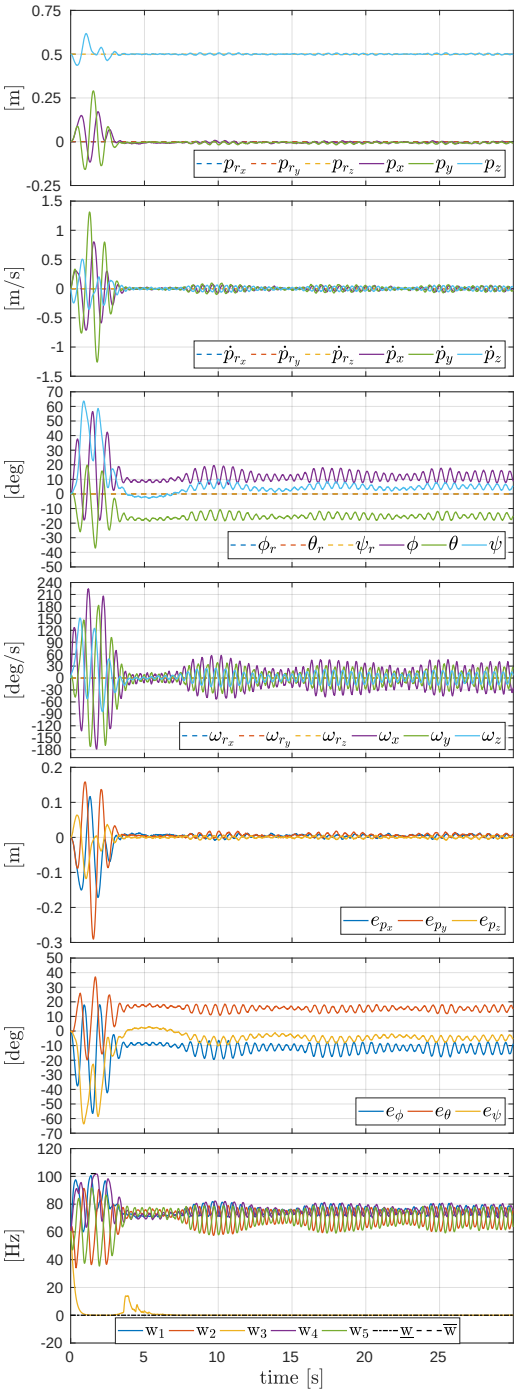


Figure 7.35 – Plots of the Tilt-Hex with rotor failure and $\beta = 0\,\text{deg}$ while hovering. The robot is disturbed with a constant external torque. *From top to bottom:* the position, linear velocity, orientation and angular velocity tracking, the position and orientation errors, and the actuator spinning velocities.

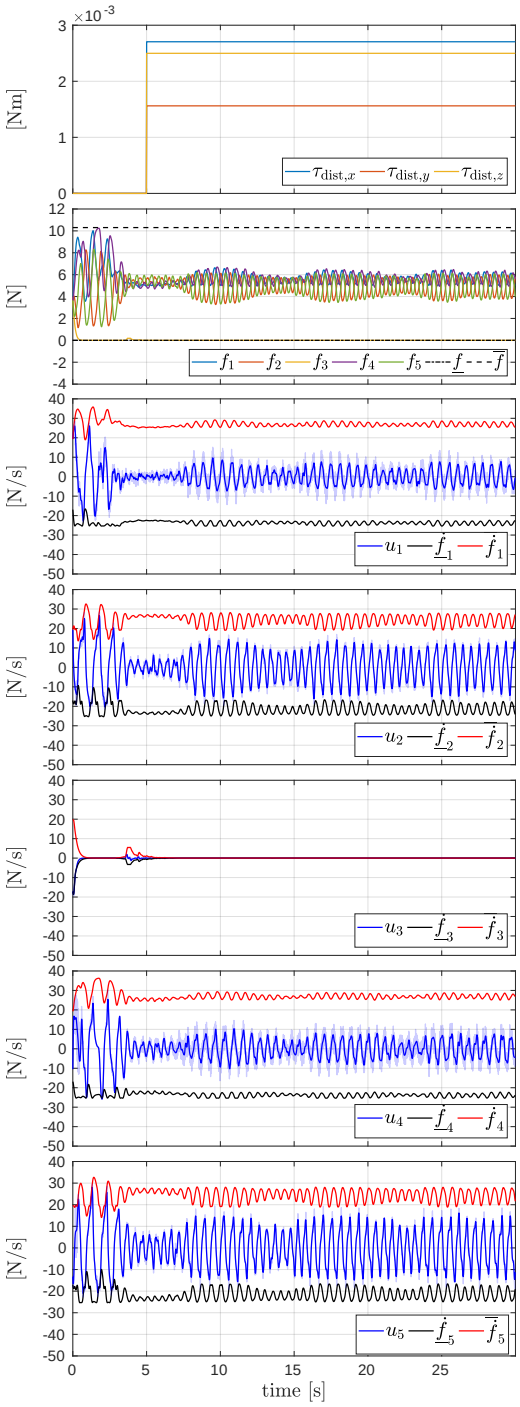


Figure 7.36 – Plots of the Tilt-Hex with rotor failure and $\beta = 0\,\text{deg}$ while hovering. The robot is disturbed with a constant external torque $\boldsymbol{\tau}_{\text{dist}}$, activated at $t = t_1$. *From top to bottom:* the disturbance torque, the actuator forces and their derivatives. In particular, all the signals remain inside the feasible region delimited by the constraints.

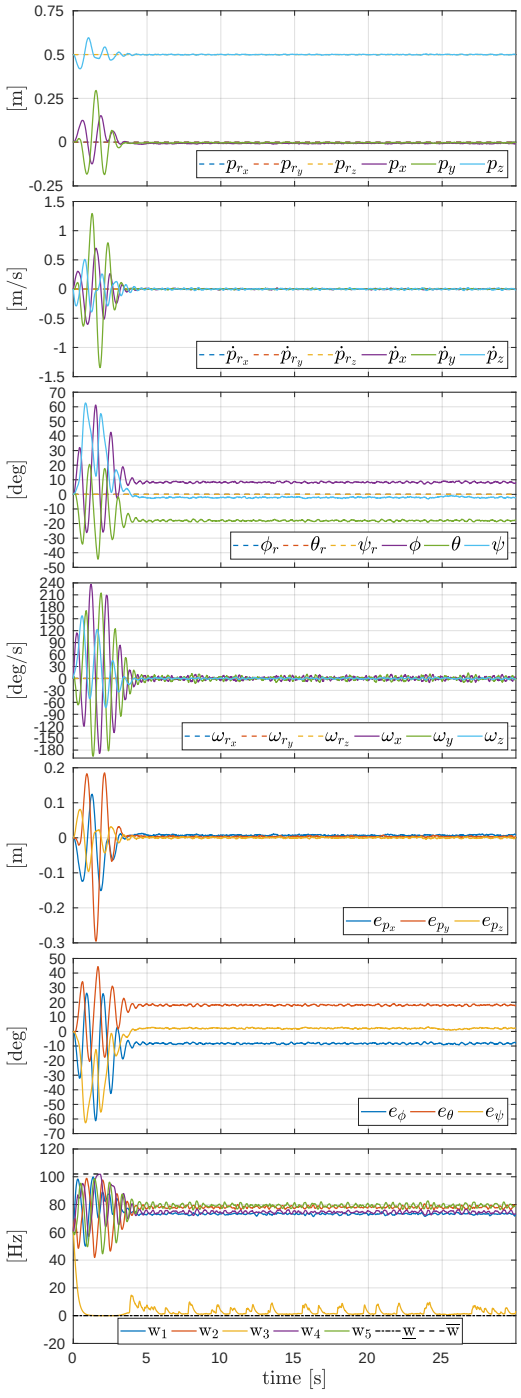


Figure 7.37 – Plots of the Tilt-Hex with rotor failure and $\beta = -25$ deg while hovering. The robot is disturbed with a constant external torque. *From top to bottom:* the position, linear velocity, orientation and angular velocity tracking, the position and orientation errors, and the actuator spinning velocities.

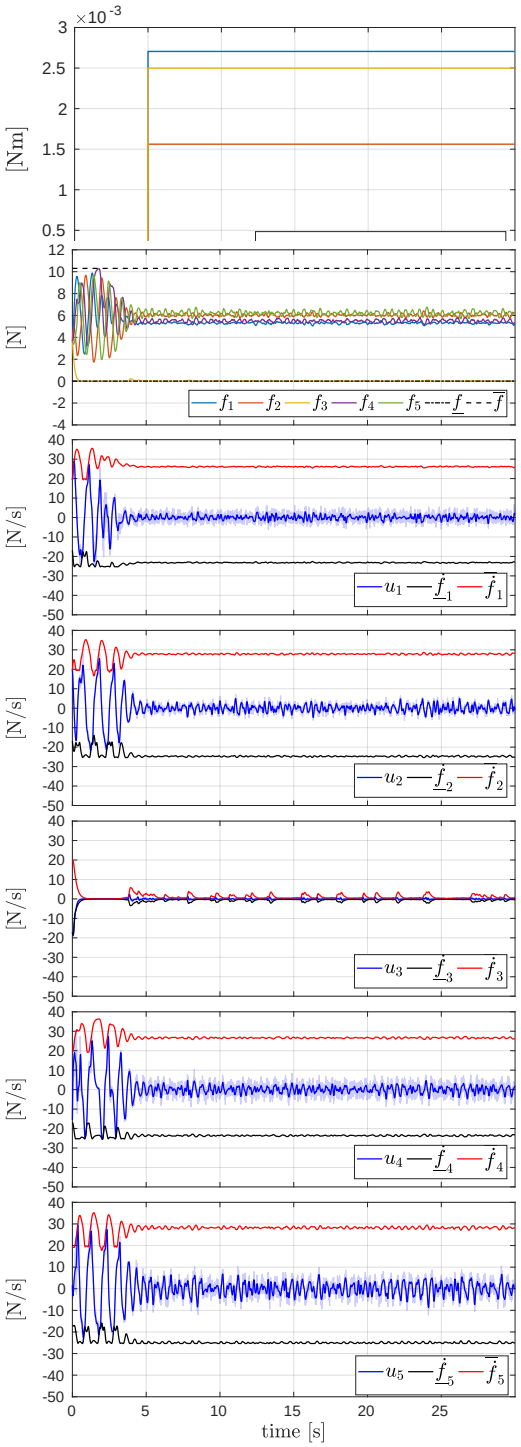
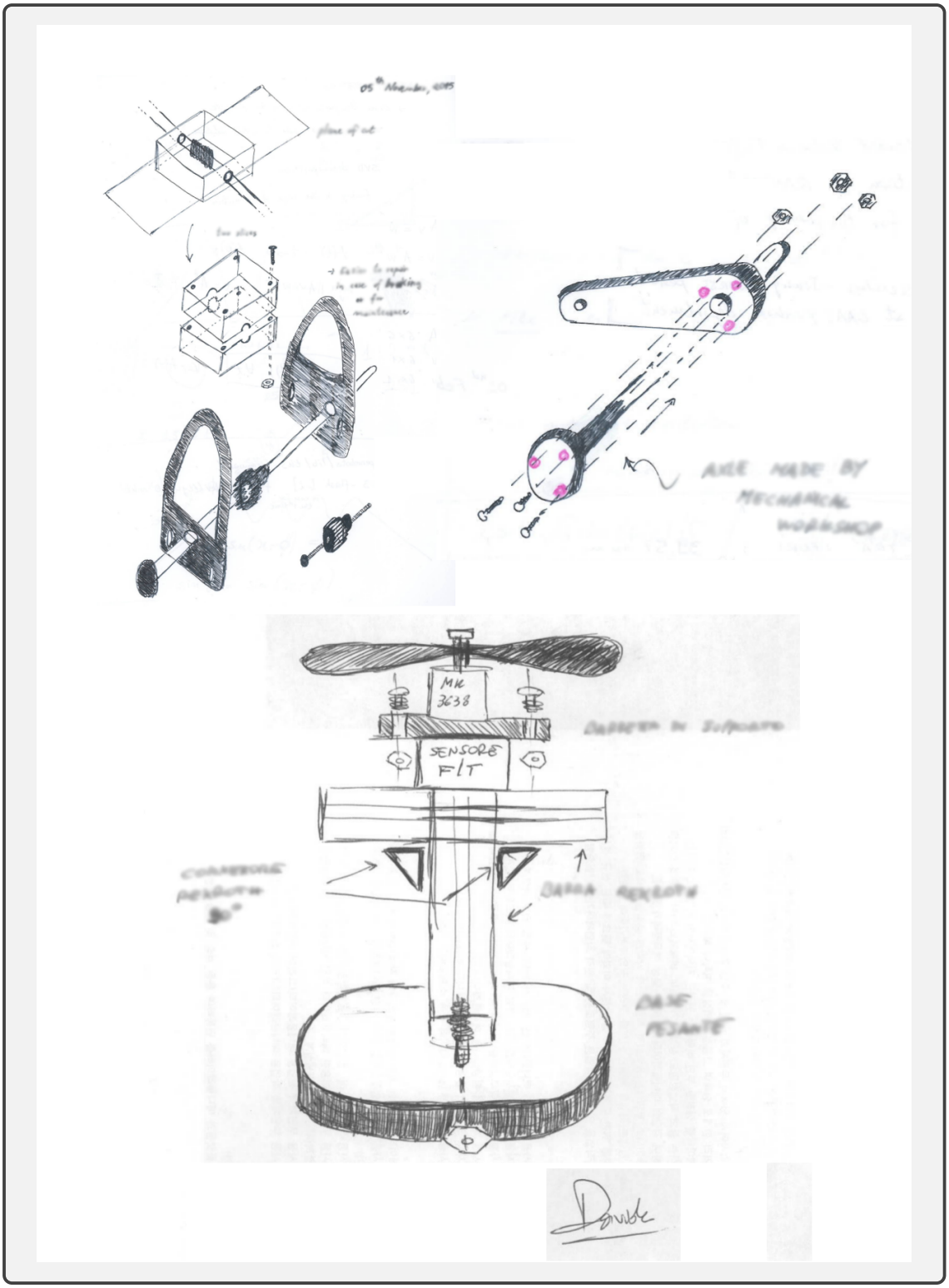


Figure 7.38 – Plots of the Tilt-Hex with rotor failure and $\beta = -25$ deg while hovering. The robot is disturbed with a constant external torque $\boldsymbol{\tau}_{\text{dist}}$, activated at $t = t_1$. *From top to bottom:* the disturbance torque, the actuator forces and their derivatives. In particular, all the signals remain inside the feasible region delimited by the constraints.

torque disturbance for the platform when $\beta = 0$ and only actuators $\{1, 2, 4, 5\}$ are effectively working. In order ensure that such perturbation cannot be compensated by a MRAV with this tilting configuration, even if the $3-rd$ actuator is actively used, we verify that $\mathbf{v}_3$ has a positive projection along the direction of the total torque $\boldsymbol{\tau}_3^B$ that can be generated by such actuator. In mathematical terms, we select $\mathbf{v}_3' = \text{sgn}(\mathbf{v}_3^\top \boldsymbol{\tau}_3^B)\mathbf{v}_3$. Finally, we scale down the vector norm in order to obtain a meaningful order of magnitude for the disturbance, i.e., $\boldsymbol{\tau}_{\text{dist}} = \frac{1}{250}\mathbf{v}_3'$. In the presented simulations, it is activated at $t = t_1$. The evolution of such perturbation, constant in body frame, is depicted in the first plot of Fig. 7.36.

The plots of this simulation related to the case $\beta = 0\,\text{deg}$ are depicted in Fig. 7.35 and in Fig. 7.36, while the ones obtained with $\beta = -25\,\text{deg}$ are portrayed in Fig. 7.37 and in Fig. 7.38. Comparing both the position and the orientation errors in the two cases, for $\beta = 0\,\text{deg}$ the platform cannot hover statically, since it periodically oscillates, with peaks of almost $\pm 2\text{cm}$ and $\pm 7.5\,\text{deg}$, around the steady-state configurations. On the other hand, for $\beta = -25\,\text{deg}$ the MRAV can fulfill the challenging goal of remaining still. This is a consequence of the fact that, for $\beta \neq 0$ the span of ${}^3\mathbf{G}_2^6$ is already 3-dimensional and so the perturbation can be annihilated while being in static hovering. In both cases, the first part of the simulation is characterized by consistent oscillations of the state components, as it is clear from the plots 1-4 of the two figures. In particular, these transients are caused by the fact that the initial robot orientation is $\boldsymbol{\eta}_0 = [0\,0\,0]^\top\,\text{deg}$, which is not attainable in steady-state for the MRAV in both configurations.

Some final remarks are in order. First of all, the aforementioned claim that, in this case, the $3 - rd$ actuator is almost never used is confirmed by the last plots of Fig. 7.35 and Fig. 7.37. Indeed, the control algorithm regulates to zero the related force component almost everywhere. In particular, during the initial transient phase, we see how the rotor velocities (and so the generated thrust forces) approach their upper bounds. Regulating the spinning rate of the $3 - rd$ rotor to a value grater than zero, would cause the other components to saturate, with large chances to destabilize the platform. Secondly, the platform orientation converges (for $\beta = -25\,\text{deg}$) to a certain value, as depicted in the third and in the sixth plots of Fig. 7.37. Note that such steady-state orientation value, which depends on α, β and on $\boldsymbol{\tau}_{\text{dist}}$, is automatically computed by the NMPC algorithm, in relation to the state and input limitations, and it is not a-priori given. This feature guarantees the optimality of the trajectory with respect to the robot dynamic capabilities and relieves the user from performing any explicit computations. Finally, remark that the proposed controller can achieve better results with respect to the one designed in [Michieletto–2017; Michieletto–2018], since the errors on the state keep bounded without diverging also in the case $\beta = 0$, despite the addition of a constant challenging disturbance which remain unknown to the NMPC algorithm. This fact highlights the potentiality of predictive controllers with respect to reactive ones.



Drawing II

Part III

Aerial Physical Interaction applications

Chapter 8

From motion control to interaction control

"Thoroughly conscious ignorance is the prelude to every real advance in science."
James Clerk Maxwell

Throughout the chapters of Part II, we have presented and validated control strategies to deal with the stabilization and the trajectory tracking of Multi-Directional Thrust (MDT) Multi-Rotor Aerial Vehicles (MRAVs), in particular with actuation constraints. All the techniques explained therein suppose that the aerial vehicle under examination is in a *contact-free* condition. Despite being fundamental to target a big set of useful real applications, as mentioned in Chapt. 1, these methodologies alone are not suitable to analyze and regulate the behavior of the aerial platform during operations *in contact* with the external environment, i.e., when the MRAV nominal dynamics is influenced by an external agent like, e.g., another robot, a human operator, or, more in general, another physical body. Pure motion control turns out to be inadequate because the unavoidable modeling uncertainties and errors may cause a rise of the contact forces (or moments), ultimately leading to an unstable behavior of platform during the interaction, especially in the presence of rigid, non-deformable environments.

In this part of the thesis, we will address more closely the topic of aerial physical interaction. More in detail, in this chapter we will present the theoretical control framework that we exploit to shape the behavior of the aerial robot[1] during the interaction. Then, in the two following chapters, we will validate such methods by outlining two interesting use-case experimental applications.

Before shaping the behavior of the controller, it is important to make sure to have a proper model of the MRAV, capable to properly describe the system also during the interaction. To this purpose, we recall that in Sec. 3.2 we mentioned that, during contact-less operations, the contribution given by the external wrench on the MRAV Center of Mass (CoM) can be neglected. On the other hand, during the interaction phase it is of primary importance to explicitly take into account the forces and torques that the MRAV actively exchanges with the surrounding environment. Indeed, neglecting these contributions in this case could easily lead

[1] The term comes from the Czech word *robota* meaning "forced labor". The use of this word in this context is motivated by the intention to fulfill more "physical" tasks with the aerial vehicle.

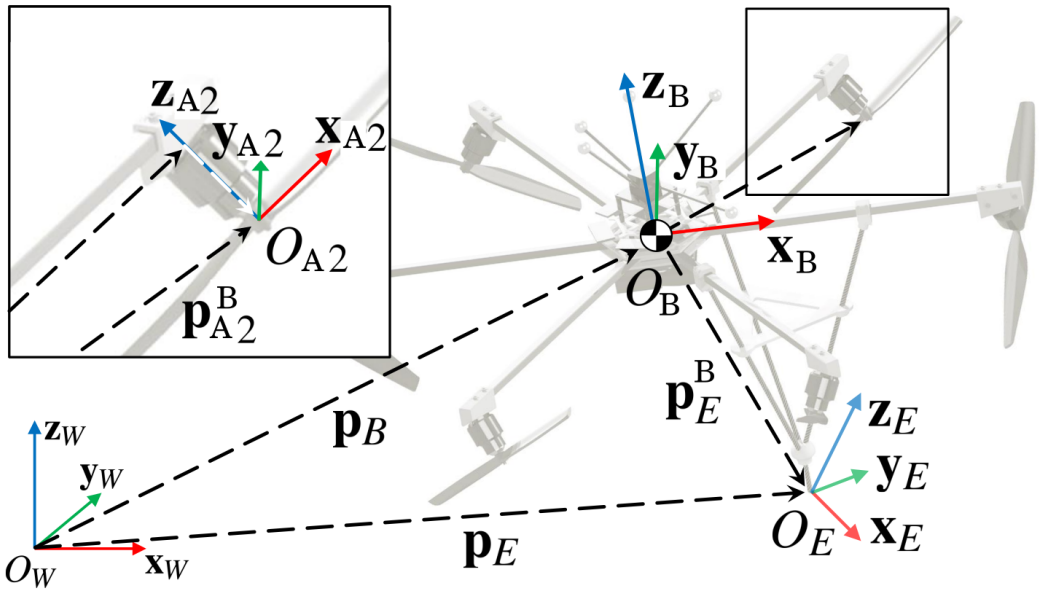


Figure 8.1 – Schematic view of important frames defined for the modeling of the interaction for the Tilt-Hex robot, endowed with a rigid EE tool.

to instability. In order to do this, we re-write (3.7) as follows:

$$\begin{bmatrix} m\mathbf{I}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{J} \end{bmatrix} \begin{bmatrix} \ddot{\mathbf{p}} \\ \dot{\boldsymbol{\omega}} \end{bmatrix} = \begin{bmatrix} -mg\mathbf{e}_3 \\ -\boldsymbol{\omega} \times \mathbf{J}\boldsymbol{\omega} \end{bmatrix} + \begin{bmatrix} \mathbf{R} & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{I}_3 \end{bmatrix} \begin{bmatrix} \mathbf{G}_1 \\ \mathbf{G}_2 \end{bmatrix} \boldsymbol{\gamma} + \begin{bmatrix} \mathbf{f}_{B\text{-ext}} \\ \boldsymbol{\tau}^{B}_{B\text{-ext}} \end{bmatrix} \tag{8.1}$$

where $\mathbf{f}_{B\text{-ext}}$ and $\boldsymbol{\tau}^{B}_{B\text{-ext}}$ represent the overall external force and torque applied at the MRAV CoM from the environment, expressed in the inertial world frame and in the body frame, respectively. It is worth to underline that, in most cases, the physical interaction between the aerial robot and the external agent takes place in a point which might be located far away from the platform CoM. Without loss of generality, we can assume that this point is located at the end of a physical interaction tool, explicitly design for the fulfillment of some tasks. In particular, we will assume that the interaction always takes place in this particular point. This motivates the introduction of a new reference frame, i.e., the End-Effector (EE) frame, which is denoted with $\mathcal{F}_E = O_E - \{\mathbf{x}_E, \mathbf{y}_E, \mathbf{z}_E\}$. The position of O_E w.r.t. O_B, expressed in $\mathcal{F}_B$, is indicated with $\mathbf{p}^{B}_{E}$, while the rotation representing $\mathcal{F}_E$ w.r.t. $\mathcal{F}_B$ is denoted with $\mathbf{R}^{B}_{E}$. The reference frames of interest, in the case of the Tilt-Hex robot endowed with a rigid tool, are represented in Fig. 8.1, while the main symbols used for the modeling are condensed in Tab. 8.1. In particular, the force and the torque acting on the EE are indicated with $\mathbf{f}_{E\text{-ext}}$ and with $\boldsymbol{\tau}_{E\text{-ext}}$, respectively. It is worthwhile to remark that, due to the offset $\mathbf{p}^{B}_{E}$, the effect of a pure force in the EE point could induce, in general, also a moment in the body frame.

Once the external interaction wrench is properly modeled, two more additional steps are needed. First of all, it is mandatory to retrieve a good estimation of such

Table 8.1 – Overview of the main symbols used for the interaction modeling.

Definition	Symbol
World Inertial Frame	$\mathcal{F}_W$
Multi-rotor Body Frame	$\mathcal{F}_B$
End Effector Frame	$\mathcal{F}_E$
Position, velocity, acceleration of O_B w.r.t. O_W, in $\mathcal{F}_W$	$\mathbf{p}$, $\dot{\mathbf{p}}$, $\ddot{\mathbf{p}}$
Rotation matrix representing $\mathcal{F}_B$ w.r.t. $\mathcal{F}_W$	$\mathbf{R}$
Angular velocity of $\mathcal{F}_B$ w.r.t. $\mathcal{F}_W$, expressed in $\mathcal{F}_B$	$\boldsymbol{\omega}$
Angular acceleration of $\mathcal{F}_B$ w.r.t. $\mathcal{F}_W$, expressed in $\mathcal{F}_B$	$\dot{\boldsymbol{\omega}}$
Position of O_E w.r.t. O_B, expressed in $\mathcal{F}_B$	$\mathbf{p}_E^B$
Rotation matrix representing $\mathcal{F}_E$ w.r.t. $\mathcal{F}_B$	$\mathbf{R}_E^B$
Mass of the vehicle	m
Vehicle's inertia matrix w.r.t. to O_B, expressed in $\mathcal{F}_B$	$\mathbf{J}$
Gravity acceleration	g
Total control force acting on the CoM	$\mathbf{f}_B$
Total control moment acting on the CoM	$\boldsymbol{\tau}_B$
External force acting on the CoM	$\mathbf{f}_{B\text{-ext}}$
External moment acting on the CoM	$\boldsymbol{\tau}_{B\text{-ext}}$
External force acting on the EE	$\mathbf{f}_{E\text{-ext}}$
External moment acting on the EE	$\boldsymbol{\tau}_{E\text{-ext}}$

wrench. Secondly, the MRAV motion controller needs be modified and tuned to guarantee a compliant behavior of the system. In this context, this will be ensured thanks to the use of the *admittance control* paradigm. A manipulator under admittance control is described by a mass-spring-damper system with adjustable parameters. In particular, this relation is an admittance if the robot control reacts to interaction forces (and moments) by imposing a deviation from the desired motion[2] [Villani–2008]. Therefore, instead of feeding the motion controller directly with the desired trajectory coming from the planner, we use as reference another trajectory, which is re-generated by taking into account the external action on the MRAV, as we will see in the following. The block diagram of the proposed interaction control architecture is depicted in Fig. 8.2.

Interaction wrench estimation

In order to properly handle physical interaction of the aerial robot with the external environment, the knowledge of the contact interaction wrench between the tool tip and the environment, $\boldsymbol{w}_{E\text{-ext}} = [\mathbf{f}_{E\text{-ext}}^\top \ \boldsymbol{\tau}_{E\text{-ext}}^\top]^\top \in \mathbb{R}^6$ is essential. To this aim, a force/torque sensor could be mounted on the robot's tool-tip, which is usually

[2]In a complementary way, the relation is an *impedance* if the robot control reacts to motion deviation by generating forces.

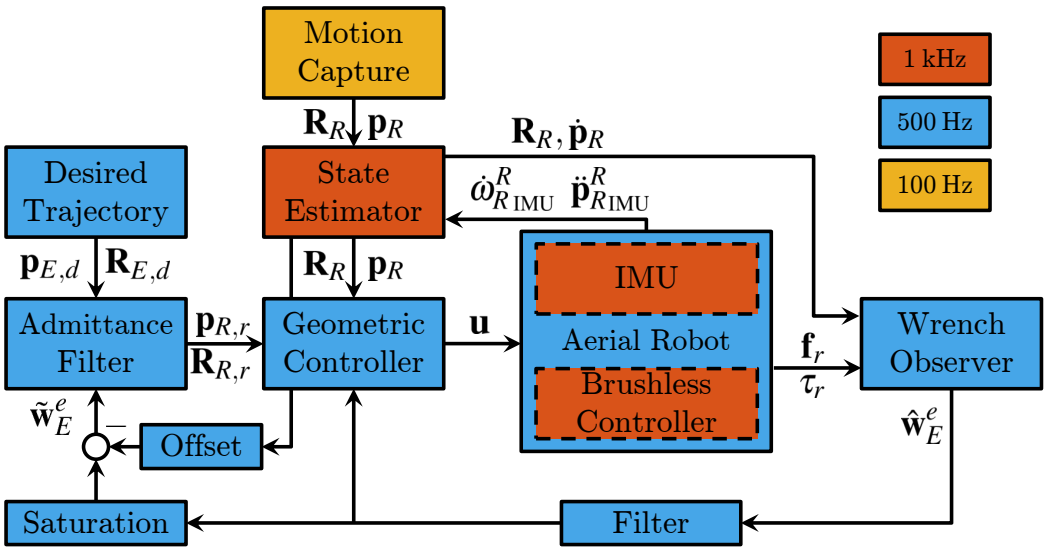


Figure 8.2 – Signal block diagram of the overall control framework. The typical runtime is highlighted. Higher derivatives of the signals have been omitted.

capable to provide a reliable measure, but this solution increases both the cost and the weight of the robot. In the aerial robotics field, a more viable solution is the adoption of a wrench estimator, that can provide a sufficiently accurate estimation, denoted as $\hat{\boldsymbol{w}}_{E\text{-ext}} = [\hat{\mathbf{f}}^\top_{E\text{-ext}} \;\; \hat{\boldsymbol{\tau}}^\top_{E\text{-ext}}]^\top \in \mathbb{R}^6$, in the presence of accurate measurements of position, velocities and, if available, accelerations.

The external wrench on the robot, $\boldsymbol{w}_{B\text{-ext}} = [\mathbf{f}^\top_{B\text{-ext}} \;\; (\mathbf{R}\boldsymbol{\tau}^B_{B\text{-ext}})^\top]^\top$, can be viewed as the effect on the robot CoM of the wrench $\boldsymbol{w}_E$ exerted by the environment on the tool tip, namely

$$\boldsymbol{w}_{B\text{-ext}} = \mathbf{H}^\top_E(\mathbf{R}_R)\boldsymbol{w}_{E\text{-ext}}, \quad \mathbf{H}_E(\mathbf{R}_R) = \begin{bmatrix} \mathbf{I}_3 & -\left[\mathbf{R}\mathbf{p}^B_E\right]_\times \\ \mathbf{O}_3 & \mathbf{I}_3 \end{bmatrix}, \tag{8.2}$$

The sensor equipment of the MDT MRAVs used in the experiments that will be presented provides accurate enough measurements of the platform position and velocities, both angular and linear, while only the linear acceleration, provided by the Inertial Measurement Unit (IMU), can be reasonably used in a wrench observer. Thus, in our work, the hybrid approach already proposed in [Tomic–2017], has been followed. More in detail, the acceleration based observer proposed by [Yüksel–2014a] is adopted in order to estimate the external interaction forces on the robot CoM, $\mathbf{f}_R$, while the external torques, $\boldsymbol{\tau}^R_R$ are obtained exploiting a momentum-based observer [De Luca–2005].

Estimation of contact forces

The following disturbance observer requiring the vehicle acceleration measure, firstly proposed for aerial robots in [Yüksel–2014a], is adopted for estimating the contact forces:

$$\begin{aligned}\dot{\hat{\mathbf{f}}}_{B\text{-ext}} &= \mathbf{L}(\mathbf{f}_{B\text{-ext}} - \hat{\mathbf{f}}_{B\text{-ext}}) \\ &= -\mathbf{L}\hat{\mathbf{f}}_{B\text{-ext}} + \mathbf{L}(m\ddot{\mathbf{p}}_R + mg\mathbf{e}_3 - \mathbf{R}_R\mathbf{G}_1\boldsymbol{\gamma}),\end{aligned} \tag{8.3}$$

where $\mathbf{L} \in \mathbb{R}^{3\times3}$ is a gain matrix to be designed and $\hat{\mathbf{f}}_{B\text{-ext}}$ is an estimate of $\mathbf{f}_{B\text{-ext}}$. By defining the observer error as $\mathbf{e}_f = \mathbf{f}_{B\text{-ext}} - \hat{\mathbf{f}}_{B\text{-ext}}$, the error dynamics, in the presence of a constant or slowly varying external force is given by [Yüksel–2014a],

$$\dot{\mathbf{e}}_f + \mathbf{L}\mathbf{e}_f = \mathbf{0}_3. \tag{8.4}$$

Thus, the error dynamics is exponentially convergent to the origin for any positive definite matrix $\mathbf{L}$.

Estimation of contact torques

In order to estimate the interaction torques, exerted by the external environment on the tool-tip, a momentum-based observer [De Luca–2005] has been designed. With reference to the rotational part of system dynamics (8.1), the angular momentum $\mathbf{q}^B \in \mathbb{R}^3$ in frame $\mathcal{F}_B$ can be computed as:

$$\mathbf{q}^B = \mathbf{J}\boldsymbol{\omega}. \tag{8.5}$$

From (8.1), the time-derivative of (8.5) can be expressed as

$$\dot{\mathbf{q}}^B = \mathbf{J}\dot{\boldsymbol{\omega}} = -\boldsymbol{\omega} \times \mathbf{J}\boldsymbol{\omega} + \mathbf{G}_2\boldsymbol{\gamma} + \boldsymbol{\tau}^B_{B\text{-ext}}. \tag{8.6}$$

By exploiting (8.6), the estimate $\hat{\boldsymbol{\tau}}^B_{B\text{-ext}}$ can be seen as the residual vector

$$\hat{\boldsymbol{\tau}}^B_{B\text{-ext}} = \mathbf{K}_I \left[(\mathbf{q}^B(t) - \mathbf{q}^B(t_0)) + \int_{t_0}^{t} (\boldsymbol{\omega} \times \mathbf{J}\boldsymbol{\omega} - \mathbf{G}_2\boldsymbol{\gamma} - \hat{\boldsymbol{\tau}}^B_{B\text{-ext}})\, d\tau \right], \tag{8.7}$$

where t and t_0 are the current and initial time instant respectively, $\mathbf{K}_I$ is a positive definite gain matrix. By reasonably assuming that $\boldsymbol{\omega}(t_0) = \mathbf{0}_3$, it implies that $\mathbf{q}^B(t_0)$ is null as well. By taking the time derivative of (8.7), through (8.6), the following dynamics for the residual vector is obtained:

$$\dot{\hat{\boldsymbol{\tau}}}^B_{B\text{-ext}} + \mathbf{K}_I\hat{\boldsymbol{\tau}}^B_{B\text{-ext}} = \mathbf{K}_I\boldsymbol{\tau}^B_{B\text{-ext}}. \tag{8.8}$$

Equation (8.8) is a first order low-pass dynamic system: it can be easily recognized that $\hat{\boldsymbol{\tau}}^B_{B\text{-ext}} \to \boldsymbol{\tau}^B_{B\text{-ext}}$ when $t \to \infty$ for any positive definite gain matrix $\mathbf{K}_I$. The choice of the matrix $\mathbf{K}_I$ is a trade-off between the convergence rate and the filtering properties of the observer: greater values of the gains allow faster convergence while

smaller values allow to filter the high-frequency noise.

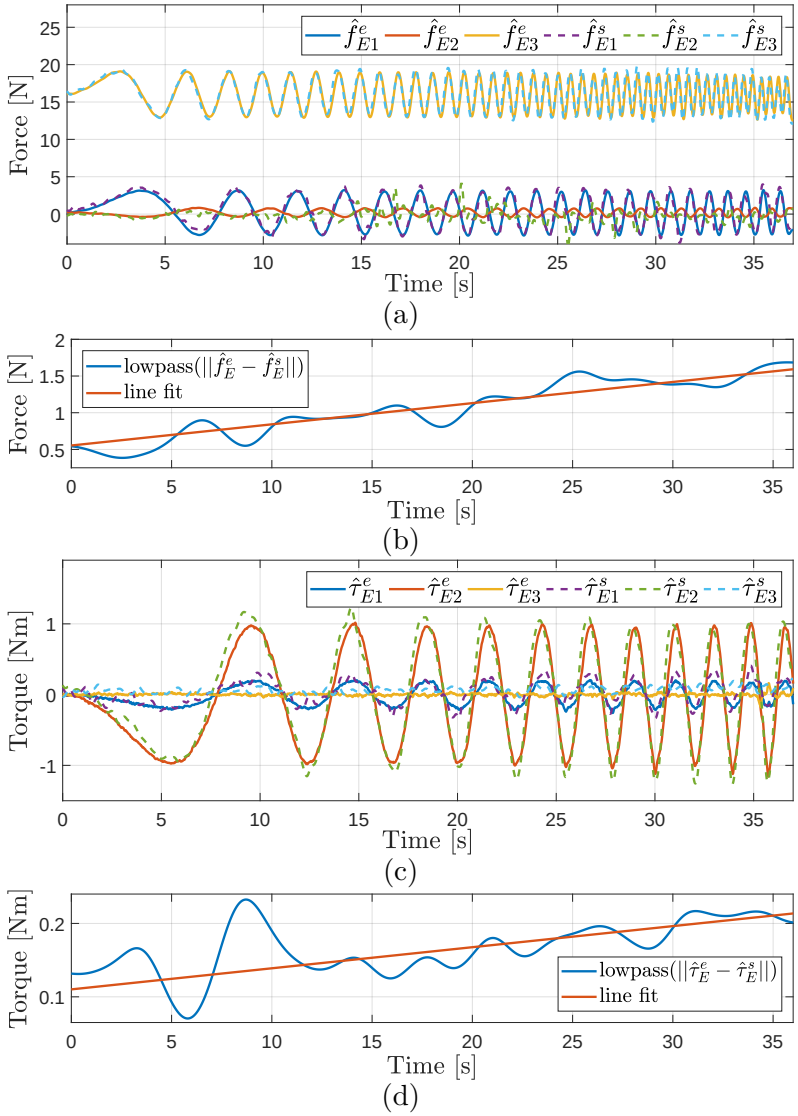


Figure 8.3 – Estimated versus measured interaction wrench. The ground truth (dashed) is measured with an ATI45 force torque transducer. The aerial robot is rigidly connected with the sensor and simultaneously performs a translational chirp signal along $\mathbf{x}_W$ and $\mathbf{z}_W$ and a rotational chirp about the $\mathbf{y}_W$ with a peak frequency of 2.5 Hz. Both, the estimated and the ground truth signals have been filtered with a non-causal low-pass filter with a 6 Hz cut-off frequency. (a) Estimated (solid) against sensor-measured (dashed) forces. (b) Low pass filtered norm of difference of the estimated and the measured forces (blue). The difference increases monotonically with an increasing oscillation frequency indicated by the line fit (red). (c) Estimated (solid) against sensor-measured (dashed) torques. (d) Low pass filtered norm of difference of the estimated and the measured torques (blue). Again, the difference increases monotonically with an increasing oscillation frequency indicated by the line fit (red).

Wrench acting on the tool tip

Once both $\hat{\mathbf{f}}_{B\text{-ext}}$ and $\hat{\boldsymbol{\tau}}^{B}_{B\text{-ext}}$ are known, the estimated wrench acting on the tool tip, $\hat{\boldsymbol{w}}_{E\text{-ext}}$ is computed as follows:

$$\hat{\boldsymbol{w}}_{E\text{-ext}} = \mathbf{H}_{E}^{-\top} \begin{bmatrix} \hat{\mathbf{f}}_{B\text{-ext}} \\ \mathbf{R}\hat{\boldsymbol{\tau}}^{B}_{B\text{-ext}} \end{bmatrix}. \tag{8.9}$$

An illustrative example of the wrench observer's precision is presented in Fig. 8.3. For this test, the Tilt-Hex the aerial robot described in Sec. 6.1 has been rigidly connected to an ATI45 force-torque sensor, which itself has been mounted on a test-stand. By letting the aerial robot tracking a trajectory and only utilizing the inner loop pose controller, the aerial robot applies a force-torque profile defined by the trajectory on the force-torque sensor. To properly test the limits of the wrench observer a chirp signal (sine with increasing frequency), simultaneously about multiple axes, has been used as trajectory. The resulting force and torque profiles are presented in Fig. 8.3. For the sake of clarity, the Fig. 8.3-(a) reports the components of $\hat{\mathbf{f}}_{E}$ and the Fig. 8.3-(c) presents the components of $\hat{\boldsymbol{\tau}}_{E}$ in (8.9) (continuous lines) against data of an ATI45 force-torque sensor (dashed lines). It is obvious that the observer can track well multiple signals, while the tracking quality slowly decreases for increasing frequencies (see Fig. 8.3-(b) and -(d)).

Interaction wrench compensation

To achieve optimal results of the admittance filter a highly stiff low-level tracking is desired. This could be achieved by increasing the gains in the controller, i.e., in (5.8) and in (5.7). A drawback of this solution would be that noise would as well be amplified and this could drive the low-level system closer to instability. Furthermore, a real zero tracking error would still not be achieved. Instead, in order to improve the convergence, the estimated wrench $\hat{\boldsymbol{w}}_{B\text{-ext}}$ is fed back to the low-level controller as an additional term in (5.7) and in (5.8), as shown in the overall block diagram of Fig. 8.2. Hereby, even the contact free flight tracking is improved as any steady state error is driven to zero. More in detail, the final control input is given by

$$\tilde{\mathbf{u}} = \begin{bmatrix} \mathbf{u}_1 \\ \mathbf{u}_2 \end{bmatrix} - \mathbf{K}_{\hat{\boldsymbol{w}}} \begin{bmatrix} \hat{\mathbf{f}}_{B\text{-ext}} \\ \hat{\boldsymbol{\tau}}^{B}_{B\text{-ext}} \end{bmatrix}, \tag{8.10}$$

where $\mathbf{K}_{\hat{\boldsymbol{w}}} \in \mathbb{R}^{6\times 6}_{>0}$ is used to properly scale the control input.

In case of perfect compensation of both the interaction forces and torques, the error dynamics is the same of (5.18) and, thus, the same stability properties hold. Otherwise, under the trivial assumption that the interaction wrench is bounded, the wrench estimation error can be viewed as a bounded term as well. In Sec. 5.1 the stability properties of the proposed controller, in absence of interaction, have been studied. In detail, it has been proven that, provided that the reference orientation

is feasible for the MRAV, the tracking errors exponentially converge to zero under mild conditions on the initial angular velocity error $\mathbf{e}_\omega$. Thus, by recurring to the theory of stability of perturbed systems [Khalil–2001] in the presence of non-vanishing perturbations, it is possible to state that, in the presence of bounded wrench estimation errors, the tracking error is ultimately bounded. Moreover, if the interaction wrench is constant, the wrench estimation error is convergent to zero, and, thus, after it vanishes also the tracking error will converge to zero as well.

Admittance filter

In order to achieve bounded forces exchanged with the environment, a compliant behavior could be enforced between the position and orientation of the end-effector and the interaction generalized forces. Assigned a planned desired trajectory for the EE in terms of position $\mathbf{p}_{E,d}$, orientation $\mathbf{R}_{E,d}$, velocities $\boldsymbol{\nu}_{E,d} = [\dot{\mathbf{p}}_{E,d}^\top \ (\mathbf{R}_{E,d}\boldsymbol{\omega}_{E,d})^\top]^\top$, and accelerations $\dot{\boldsymbol{\nu}}_{E,d}$, the corresponding set of reference motion variables to be fed to the motion controller (in particular the one presented in Sec. 5.1) $(\mathbf{p}_{E,r}, \mathbf{R}_{E,r}, \boldsymbol{\nu}_{E,r}, \dot{\boldsymbol{\nu}}_{E,r})$, can be generated via an admittance filter, characterized by the following dynamics

$$\mathbf{M}_E \Delta\dot{\boldsymbol{\nu}}_E + \mathbf{D}_E \Delta\boldsymbol{\nu}_E + \mathbf{K}_E \mathbf{e}_E = \hat{\boldsymbol{w}}_{E\text{-ext}}, \tag{8.11}$$

where $\Delta\boldsymbol{\nu}_E = \boldsymbol{\nu}_{E,d} - \boldsymbol{\nu}_{E,r}$ is the velocity error, while $\mathbf{e}_E$ is the pose error given by

$$\mathbf{e}_E = \begin{bmatrix} \mathbf{p}_{E,d} - \mathbf{p}_{E,r} \\ \frac{1}{2}(\mathbf{R}_{E,d}\mathbf{R}_{E,r} - \mathbf{R}_{E,r}\mathbf{R}_{E,d})^\vee \end{bmatrix}. \tag{8.12}$$

Equation (8.11) represents the dynamics of a 6-Degree of Freedom (DoF) mechanical impedance [Siciliano–2009] of inertia $\mathbf{M}_E$, damping $\mathbf{D}_E$ and stiffness $\mathbf{K}_E$: those matrices are all positive-definite and suitably chosen in a way to impose an over-damped behavior to the system. Moreover, in order to guarantee the stability of the overall system, the gain matrices must ensure that the motion controller (inner loop) is characterized by a faster dynamics with respect to the admittance filter.

Once the reference trajectory of the end-effector has been computed it should be expressed in terms of CoM reference trajectory in order to be tracked by the inner loop pose controller. The reference position and orientation of the robot are then computed (see Fig. 8.1) as

$$\begin{cases} \mathbf{p}_r = \mathbf{p}_{E,r} - \mathbf{R}_r \mathbf{p}_E^B, \\ \mathbf{R}_r = \mathbf{R}_{E,r} (\mathbf{R}_E^B)^\top, \end{cases} \tag{8.13}$$

while the CoM reference velocities and accelerations are obtained taking the time

derivatives of (8.13). In detail, the reference velocities are given by

$$\begin{bmatrix} \dot{\mathbf{p}}_{,r} = \dot{\mathbf{p}}_{E,r} - \mathbf{R}\,[\boldsymbol{\omega}_r]_\times\,\mathbf{p}_E^B, \\ \boldsymbol{\omega}_r = \boldsymbol{\omega}_{E,r}, \end{bmatrix}, \tag{8.14}$$

while the reference accelerations are

$$\begin{bmatrix} \ddot{\mathbf{p}}_r = \ddot{\mathbf{p}}_{E,r} - \mathbf{R}\,[\dot{\boldsymbol{\omega}}_r]_\times\,\mathbf{p}_E^B - \mathbf{R}\,[\boldsymbol{\omega}_r]_\times^2\,\mathbf{p}_E^B, \\ \dot{\boldsymbol{\omega}}_r = \dot{\boldsymbol{\omega}}_{E,r} \end{bmatrix}. \tag{8.15}$$

It is worthwhile to underline that the parameter matrices $\mathbf{M}_E$, $\mathbf{D}_E$, $\mathbf{K}_E$ in (8.11) can be adjusted in order to shape the admittance filter physical properties, i.e., to tune the behavior of the mass-spring-damper system at will. In particular, the desired behavior shall depend on the particular task to be fulfilled. Moreover, the compliance of the translational dynamics can be regulated independently from the rotational one. An example of the different responses that can be achieved is presented in App. A, while these and other additional results can be appreciated in [video05–2019].

Remark (admittance vs. impedance paradigm). The admittance approach has been preferred to an impedance one since it allows to better counteract the model uncertainties and to separate the impedance control action from the motion control action, which can be made purposefully stiff so has to enhance disturbance rejection and ensuring good tracking performance in free space [Villani–2008]. The motion controller described above is characterized by a bandwidth wide enough to guarantee the stability of the inner/outer loop and at same time guarantee high performance in the free space motion. Moreover, even if the environment is rigid, the flying platform and the considered tool ensure a certain level of passive compliance which confers robustness to the scheme.

Practical implementation

To further improve the control scheme several adjustments have been made to enable a better performance of the system in flight and during contact.

- Modeling errors (or, more precisely, errors in the manufacturing of the aerial robot with respect to the desired model) do cause a constant error in $\hat{\boldsymbol{w}}_{B\text{-ext}}$. To eliminate this error, the steady state error has been estimated during a contact-free hovering flight. Such offset has been then taken into account both in the wrench observer and in the controller.
- To suppress non-existent small force and torque estimations due to sensor noise, we implemented a *dead zone* on the admittance filter input. Any norm value of force component below 0.2 N and any norm value torque component below 0.2 N m will be neglected. To achieve a continuous wrench signal, the same thresholds are subtracted from higher estimations. This implies that

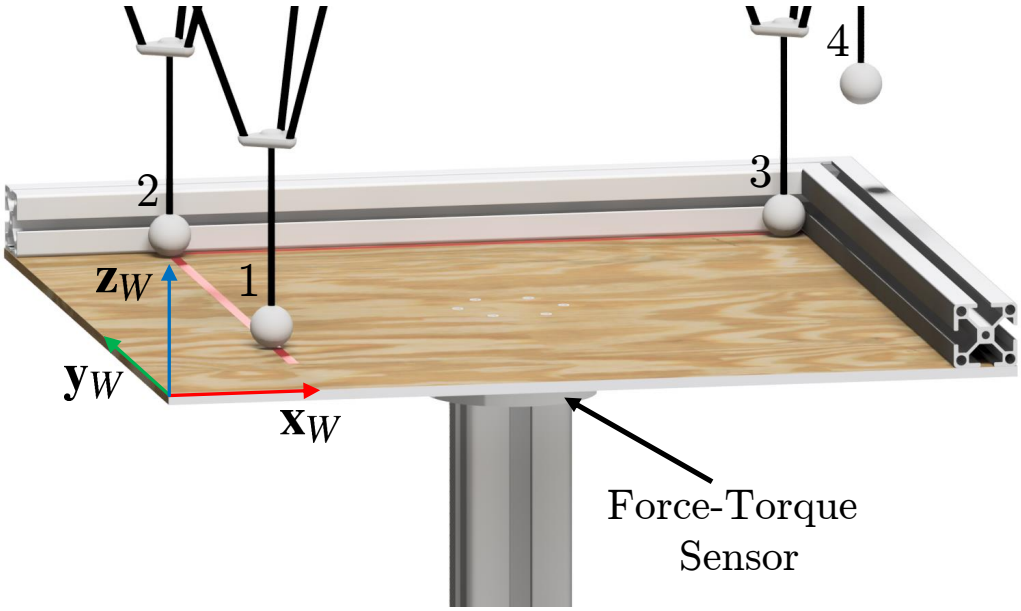


Figure 8.4 – Sliding with multiple contacts: The end-effector slides along the surface from the initial contact point (numbered with 1) to the final point (indicated with 3) along the red line. The numbers correspond to the numbers in Fig. 8.5 and indicate where the phases change. 1 indicates a single contact, 2 indicates two contacts, 3 indicates three contacts, with no lateral motion possible and 4 indicates free-flight.

the admittance filter receives as input a wrench slightly lower than the real interaction wrench.

- The estimated contact force $\hat{\boldsymbol{w}}_{E\text{-ext}}$ is filtered with a digital lowpass-filter before it is further used.

Validation of the interaction framework

In order to experimentally validate the aforementioned framework designed for aerial physical interaction, we present here the results of an experiment where the Tilt-Hex slides on a surface mounted on a force/torque sensor (ATI45). This will assess the effectiveness of the wrench estimate also in the presence of time-varying forces, and the capability of the admittance paradigm to preserve the system stability during the contact phase. Where needed, we will indicate the components of a generic vector $[\bullet] \in \mathbb{R}^3$ by the letters x, y and z, i.e., $[\bullet] = \left[[\cdot]_x [\cdot]_y [\cdot]_z\right]^\top$.

The presented experiment is inspired by common industrial tasks like inspection with contact, surface polishing or welding where, while translating, a particular force application on a surface is needed. In the experiment, the end-effector is commanded to slide along a horizontal plane and applies a dynamic force profile. At the same time, the number of translational constraints changes from one, to two, to finally three—allowing for no lateral movement. For this experiment a horizontal plate has been mounted on top of a force-torque sensor (which serves as

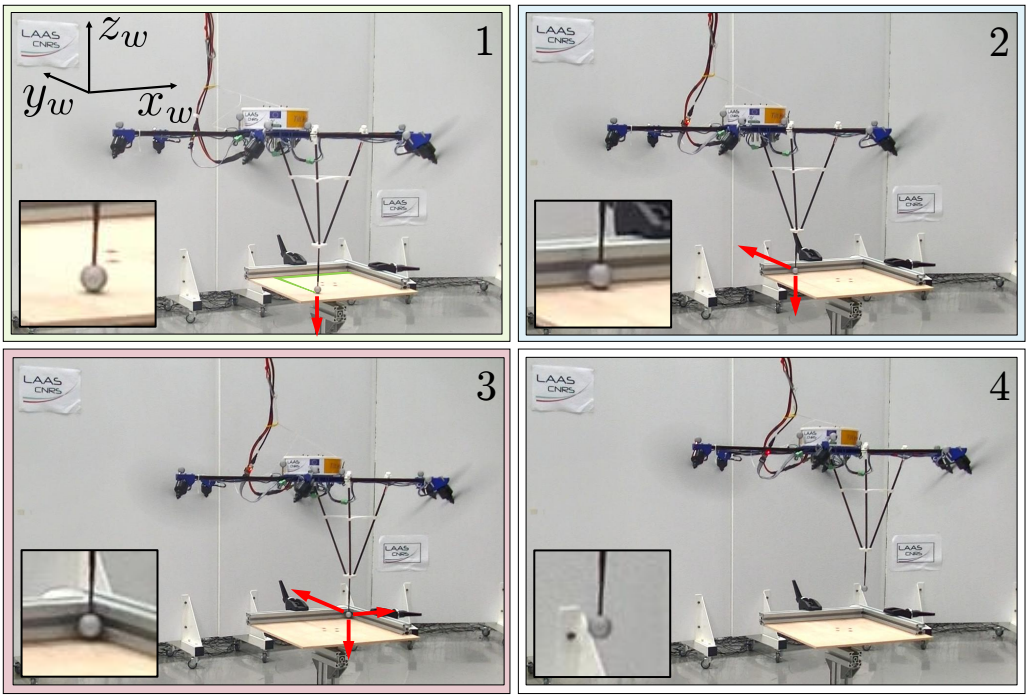


Figure 8.5 – Snapshots of the multiple contacts experiment, contact forces are highlighted with red arrows. The background colors match the colors of the contact phases in Fig. 8.6. 1) Single contact with surface. The Sliding trajectory is marked in green. 2.) Double Contact. 3.) Triple contact. 4.) Free flight phase.

ground truth). Additionally, two ledges aligned with $\mathbf{x}_W$ and $\mathbf{y}_W$ limit the plate (see Fig. 8.4 and Fig. 8.5). After establishing contact between the end-effector and the horizontal plate, the experiment consists of three phases. In the first phase, highlighted with bright green in Fig. 8.5 (see 1) and in Fig. 8.6, the end-effector applies a vertical force along $\mathbf{z}_W$ while sliding in direction $\mathbf{y}_W$, until the first ledge is touched. Then, the second phase starts with applying a constant force against the ledge, in direction $\mathbf{y}_W$ (highlighted with bright blue in Fig. 8.5 (see 2) and in Fig. 8.6. Next, the EE slides along the ledge in direction $\mathbf{x}_W$ and additionally a saw-tooth force profile is applied along $\mathbf{z}_W$, marked with a circled 1 in Fig. 8.6-(a). Once the second ledge is reached, the third phase starts (highlighted with bright red in Fig. 8.6. The end-effector is now in contact with three sides and cannot translate anymore. Saw-tooth force profiles are now consecutively applied along $\mathbf{x}_W$ and $\mathbf{y}_W$, marked with a circled 2 and 3 in Fig. 8.6-(a). Finally, the Tilt-Hex takes off and detaches the contact between the surface and the end-effector.

Fig. 8.6 presents the experimental results. The reference $\mathbf{p}_r$, desired $\mathbf{p}_d$ and actual positions $\mathbf{p}$ are depicted in Fig. 8.6-(a), showing first, an error free matching of $\mathbf{p}, \mathbf{p}_r$ and $\mathbf{p}_d$ in the absence of a contact force and second, a divergence between $\mathbf{p}_d, \mathbf{p}_r$ during contact, while the end-effector $\mathbf{p}$ still perfectly tracks $\mathbf{p}_r$. The same holds for the oreintation profiles, depicted in Fig. 8.6-(b). The admittance filter effect, i.e., the difference between $\mathbf{p}_d$ and $\mathbf{p}_r$ is presented in Fig. 8.6-(c). It is nice

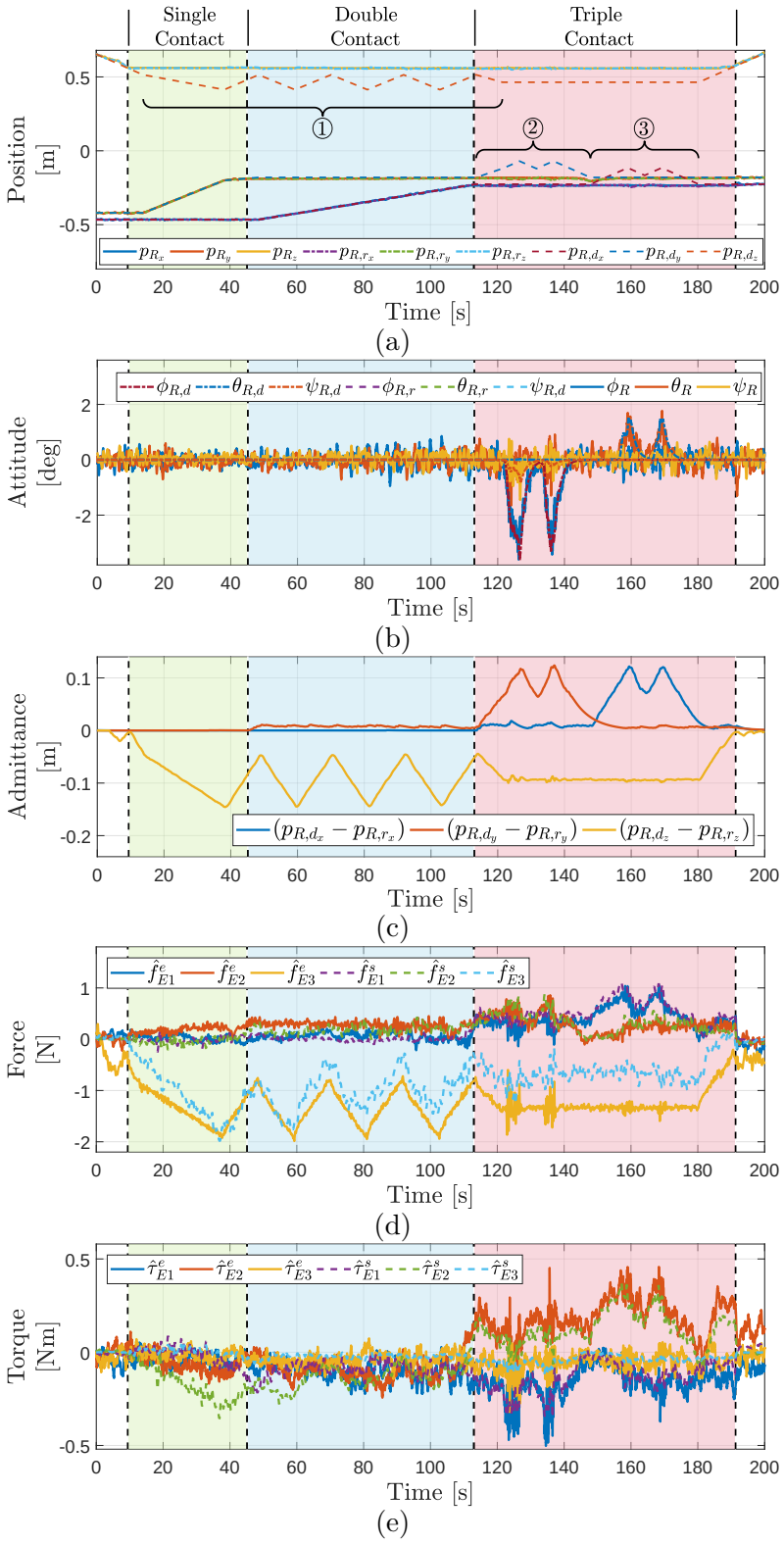


Figure 8.6 – Multiple Contact Point task: (a) Reference (dashed), desired (dot-dashed) and actual position (solid). (b) Reference (dashed), desired (dot-dashed) and actual orientation (solid). (c) Difference between desired trajectory $\mathbf{p}_{R,d}$ and the admittance filter output $\mathbf{p}_{R,r}$ due to the contact force. (d) Estimated (solid) and measured (dashed) tool-tip contact forces - both low pass filtered. (e) Estimated (solid) and measured (dashed) tool-tip contact torque - both low pass filtered.

to see how the three contact phases (single, double and triple) match well with the admittance difference. In contrast to the first three experiments, aerodynamic effects between the aerial robot and the solid surface are observed, resulting in a mismatches between the estimated and measured forces and, see Fig. 8.6-(d). While particularly for the horizontal force, the saw-tooth profile is clearly visible, the estimated force $\hat{f}^e_{E3}$ and measured force $\hat{f}^s_{E3}$ data drift apart, as the hexarotor flies over the platform. Interestingly, the estimated force is larger than the measured value. We assume that the down-wash of the propeller is reflected on the ground and lifts the surface, resulting in a wrong measured force $\hat{f}^s_{E3}$. In order to complete the data presentation, the estimated and measured torques are depicted in Fig. 8.6-(e). While the trend of the estimated torques matches very well, we again see mismatches during phases where the propeller down-wash particularly hits the plane far away from the center, where the force torque sensor is mounted (e.g., , at $t = 37\,\mathrm{s}$).

With the previous experiments, we assessed the good estimation capability of the external wrench observer. Moreover, we showed that the admittance filter is able to preserve the stability of the aerial robot when during the interaction phase with the environment. In the next two chapters, we will present two interesting applications which are motivated by real use-cases. The first one is the installation/decommissioning of long objects in an industrial scenario. The aerial robot task is to lift a long load, like a metallic beam, grasping it from one side. Such operation will be performed by the robot alone and also in cooperation with another agent, i.e., a ground robot, in order to reduce the torque needed at the agent side. In the second operational mode, the aerial robot acts as *flying assistant*. As far as the second application is concerned, we target the inspection with contact of pipelines. Despite being contextualized in in an indoor laboratory environment, the experiments that will be presented epitomize the essential starting point towards the final outdoor end-use, which is pursued by European Projects like [AeRoArms–]. Examples of the impact of the topics presented in this thesis are given in Chapt. 11.

Chapter 9

Bar lifting task

"Either write something worth reading or do something worth writing."
Benjamin Franklin

In the scope of aerial physical interaction, there are many real cases in which only a part of the whole load has to be substantially lifted from the ground, while another part can (or has to) remain close to it. Tasks like assembly, maintenance or decommissioning involving the manipulation of long bars are concrete examples of such cases. As an illustration of such tasks, one can think about gutters and pipes installation, cropping fruits or trimming trees using a pole saw. Therefore, it is worth to focus the efforts on an approach where the load is mainly carried by a ground robot or a human operator helped by one or more aerial robots [Staub–2017] used as *Flying Assistants*. A ground robot or a human operator handling a long bar by one of its ends have to face the following challenges: *i)* high torque at the grasping point; *ii)* reduced dexterity in the manipulation; and *iii)* vibrations of the bar and its other end. All three can be mitigated by using Flying Assistants.

In this chapter, we present the experimental results of the bar lifting task obtained with the OT-Hex, the third in-house developed hexarotor presented in Sec. 6.1. Differently from other solutions [Staub–2017], we propose to address this task with a Multi-Directional Thrust (MDT) flying manipulator, in order to better resist lateral perturbations and simplify the decoupling mechanism for the load-induced torque. The control framework is composed of a pose controller wrapped by an admittance filter, which is fed by a wrench observer, as outlined in Chapt. 8. In particular, we employed the motion controller presented in Sec. 5.1, mainly for a matter of simplicity of integration.

In Sec. 9.1, we demonstrate the capabilities of the OT-Hex for autonomously lifting bars, thanks to its tailored design, as well as the performance of the interaction control outlined in Chapt. 8. Moreover, the robustness of the controller to external perturbations and parameter uncertainties is validated. Finally, in Sec. 9.2, we present an evolution of the manipulation task, where the aerial robot is asked to lift and cooperatively manipulate the load together with a ground robot. Thanks to these additional results, we also demonstrate the OT-Hex promising capabilities as a Flying Assistant for cooperative manipulation with another heterogeneous agent.

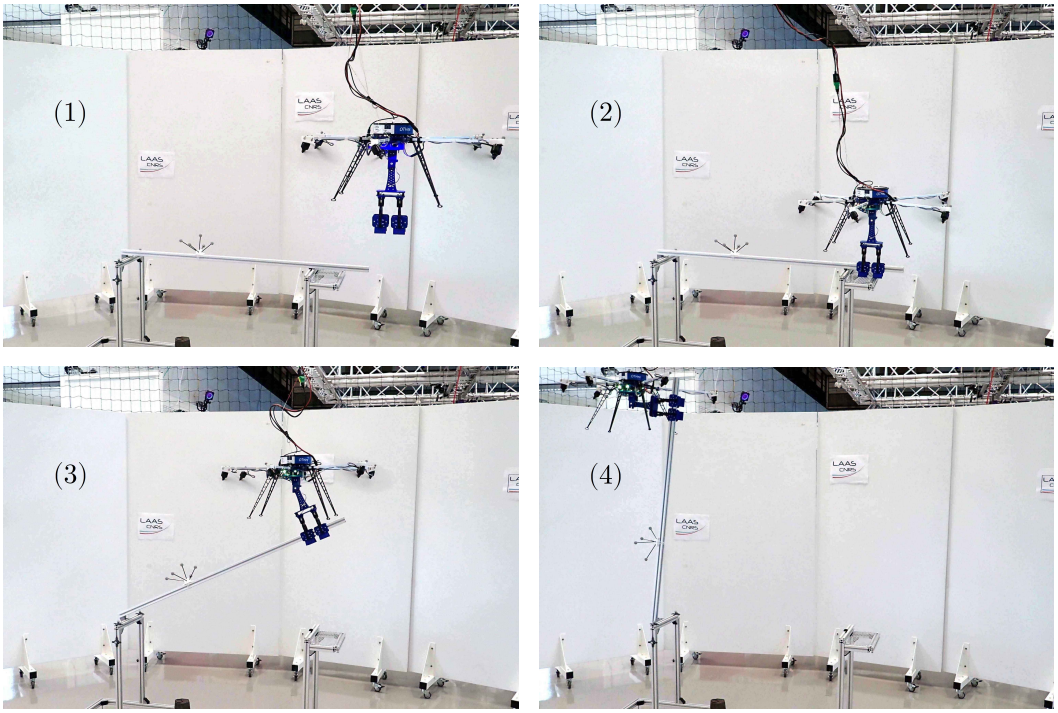


Figure 9.1 – Time-lapse of the OT-Hex in the bar lifting experiment; (1) contact free-flight, (2) grasping the horizontal bar, (3) lifting the bar, (4) bar just lifted, inverting the motion.

Single robot approach

In this section, the considered task for the OT-Hex is to lift a bar, which is fixed to the ground by means of a revolute joint one one side, from an initial horizontal configuration. For our experimental purpose a planner was devised, composed of: *i)* a task planner; *ii)* a Finite-State Machine (FSM); and *iii)* a trajectory generator with a different policy for each state of the FSM. It generates the reference state and nominal force trajectory to be followed by the OT-Hex and also triggers the grippers and control/warning lights of the robot, easing system state monitoring for the operator. The FSM allows to switch trajectory generation policies for contact and contact-less operations.

Experiment in nominal conditions

In a first experiment, we perform a nominal bar lift, from a rest angle of $\theta_0 = 0$ to a final angle of $\theta_f = \pi/2$, as depicted in Fig. 9.1, which highlights the important stages in the motion. In this experiment the contact-less flight is commanded manually by a human operator, to reach the desired grasping point. At the beginning of the experiment, the OT-Hex takes-off from the landing structure and can be driven in free-flight either manually by the human operator (with a joystick or an haptic interface), or autonomously with a trajectory given by our planner. In the free-flight phase, the system is tasked to reach the desired grasping point on the bar. Since

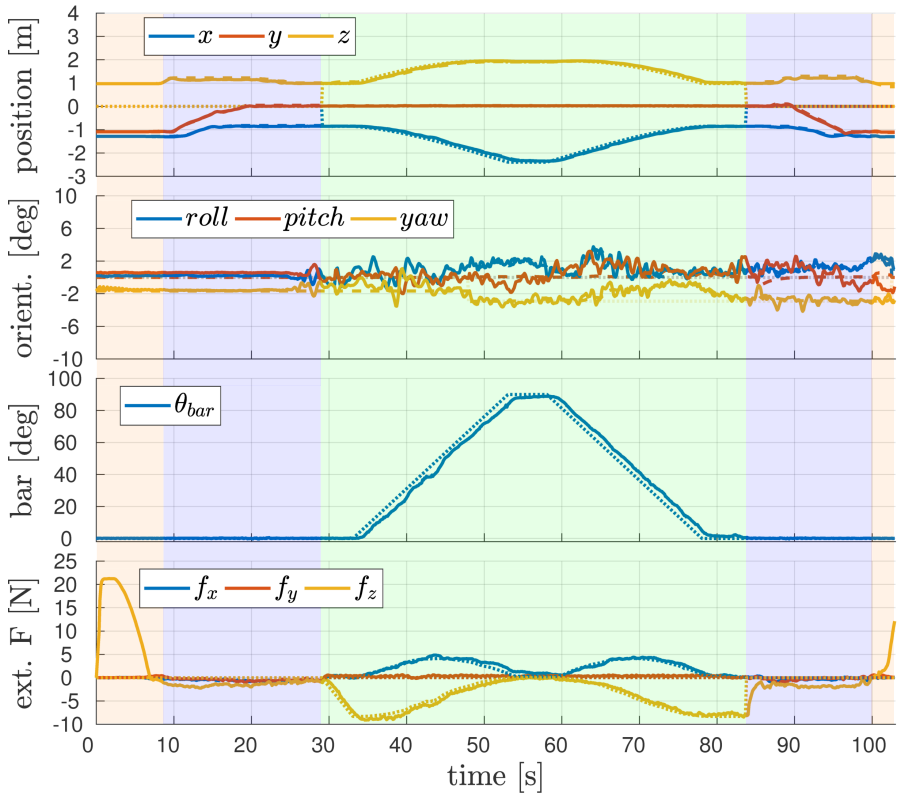


Figure 9.2 – Experimental data of a nominal bar lift. *From top to bottom:* the OT-Hex position and orientation, the bar tilting angle and the wrench observer forces. Quantities are denoted as follow; measured (solid), from the planner (dotted), from the admittance filter (dashed). The admittance filter is activated only during the lifting phase (green). In the contact-free flight phase (blue) the OT-Hex tracks a trajectory given by the planner. The moment the OT-Hex rests on the landing platform is colored in orange.

there should be no physical interaction during this phase, the admittance filter is turned off until the lifting phase starts. Once the contact has been established, a change of the state in the FSM is triggered to toggle the autonomous bar lifting, in which the planner provides a desired trajectory and nominal force to lift the bar, computed based on the bar kinematics. The admittance filter becomes active.

Associated results for the nominal case are presented in Fig. 9.2. In particular, measured quantities are in solid lines, while the desired trajectory from the planner is dotted and the compliant reference trajectory from the admittance filter is dashed. The low-level controller is fed by the dotted trajectory in free-flight and by the dashed trajectory when the admittance filter is running in contact phase. The attitude tracking is detailed in the second plot from the top, with the same convention for the lines. The third plot from top displays the desired and the measured altitude angle of the bar. It can be seen that the actual angle is behind the desired when the OT-Hex lifts the bar, and in advance when the bar is descending. This can be explained by the fact that the bar position (i.e., the planar trajectory) is not the regulated quantity, since the admittance filter is generating a compliant

reference trajectory based on the external wrench. Therefore, the OT-Hex has a less aggressive behavior lifting the bar, while in the second phase it is pulled down by the descending bar. Finally, the bottom plot shows the external wrench estimated against the nominal lift force computed by the planner. Note at the beginning of the experiment the OT-Hex rest on the landing platform, hence the wrench estimator senses a force along $\mathbf{z}_B$ of about 20 N (due to its weight), which vanishes as the aerial robot takes off.

Experiment in non-nominal conditions

Two additional experiments intend to show the system robustness in case of : *i)* parameter uncertainties about the bar physics in the planner; and *ii)* blocked bar, to highlight the compliant behavior induced by the admittance filter. We first induce parameter uncertainty by introducing a 20 cm bias in the grasping location of the bar in the planner. In this way the planner computes a reference trajectory for the OT-Hex which is unfeasible, as it is rigidly attached to the bar 20 cm away from the point considered by the planner. If this trajectory was directly sent to the pose controller, it would result in unstable behavior of the system, most likely leading to destruction. The presence of the admittance filter is then extremely important, since it modifies the nominal trajectory given by the planner, using the information of the estimated external wrench, to produce a feasible trajectory. The results of this test are gathered in Fig. 9.3, where we focus our interest on the autonomous lifting part, which highlights the compliant behavior. The reference trajectory from the planner (dotted) is altered by the admittance (dashed) to accommodate the physical constraints of the system (via the estimated wrench). One can remark the good tracking performances of the low-level controller, as it follows effectively the trajectory provided by the admittance filter, with absolute error in position below 3 cm in $\mathbf{x}_W$ and $\mathbf{y}_W$, and below 14 cm for $\mathbf{z}_W$ (which can be explained by the presence of the safety cable) and absolute error in orientation below 2.1° along all axis, along all the trajectory. The maximal bar tilting angle θ is also reached, indeed for the vertical configuration of the bar the position of the OT-Hex is the same along $\mathbf{x}_W$ independently from the grasping point, as visible in Fig. 9.3.

The second additional experiment is devised to highlight the system robustness consists in blocking the bar in its ascent. To do so, a rope is attached to the ground preventing the bar tilting angle to reach more than 18°. The wrench estimator is sensing an additional external force as soon as the rope is taut, see at the instant around 9 s in Fig. 9.4, with the same line convention as before. Once the ascent is blocked, a second compliance mechanism is triggered, the desired trajectory is "waiting" for the measured trajectory to be in its vicinity. This mechanism consists in thresholding the error in the admittance filter to prevent unstable behavior due to excessive control action, which could be overcome either by adapting the admittance filter gain once the bar is blocked or by making the planner aware of tracking error as we did, thus avoiding tedious tuning of the admittance behavior. As it can be seen from the plot, the reaction to the blockage is smooth, and once the position error

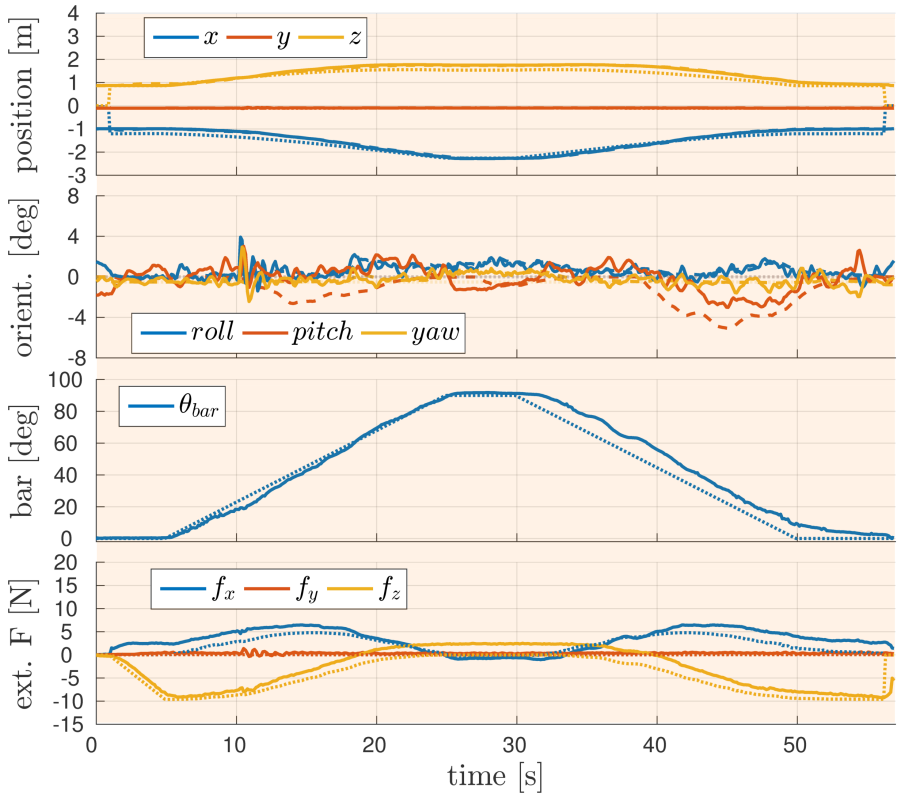


Figure 9.3 – Experimental data of a bar lift, with parametric uncertainty in the grasping point of 20 cm. Same signals and conventions as in Fig. 9.2. Note the difference between the desired position and the reference one due to the parameter uncertainty. A stable behavior is achieved thanks to the compliance enabled by the control scheme.

between the planned trajectory and the actual measurements is below a threshold, the behavior described in the nominal case is prevailing. This shows the robustness of our proposed solution to force disturbances. To better appreciate the presented experimental results, the reader is referred to the multimedia attachment [video08–2018] and experimental data (with suitable scripts to plot them), provided for download at the following link `https://zenodo.org/record/2640502#.XfDgaNF7nkw`.

Multi robot approach

In this section, we present a novel class of heterogeneous systems which tackles the problem of manipulating long objects that cannot be grasped close to their Center of Mass (CoM). Such systems go beyond the limitations of the previous approaches, which were using either only ground manipulators [Knepper–2013; Machado–2016], either only aerial robots [Lindsey–2012; Sreenath–2013; Augugliaro–2014]. This is achieved by leveraging the advantages of both *aerial* and *ground* robots together. The small payload of the aerial robots is compensated by the strength of the ground manipulators, while the limited workspace and poor Cartesian torque

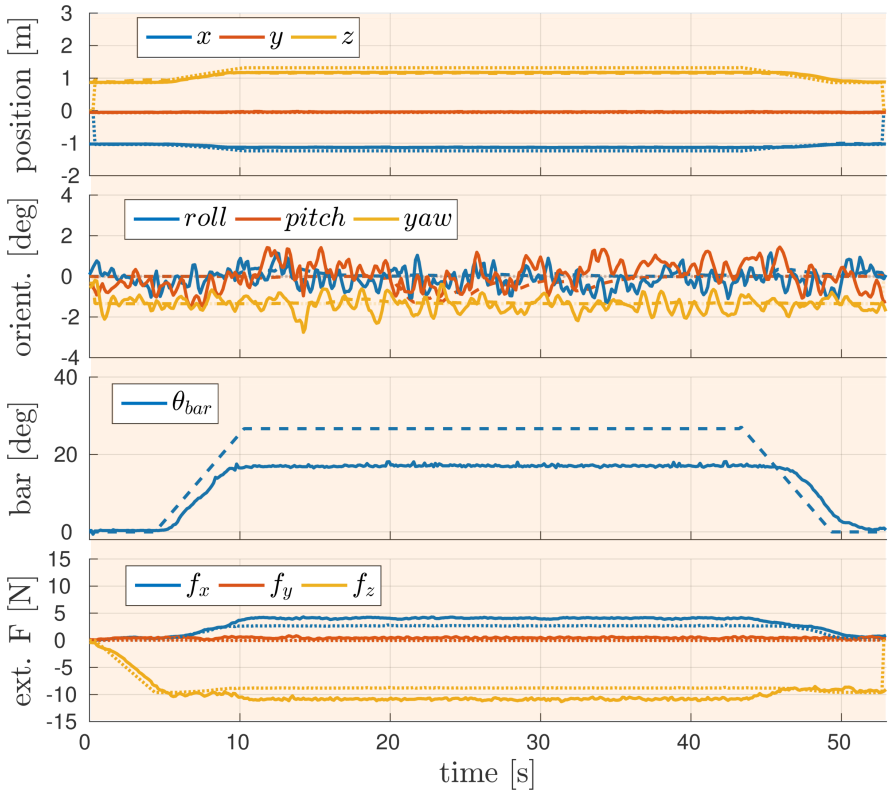


Figure 9.4 – Experimental data of a bar lift, blocked at 18° to emphasize the control framework compliance. Same signals and conventions as in Fig. 9.2. Note the difference between the desired position and the reference one coming from the admittance filter when the bar is blocked.

at End-Effector (EE) of the ground side is balanced by the virtually unlimited workspace and the favorable lever provided by aerial robots. In this way, Multi-Rotor Aerial Vehicles (MRAVs) act as *Flying Assistants*, as they assist ground robots from the air. The proposed class is called Multiple Aerial-Ground Manipulator System (MAGMaS), c.f. Fig. 9.5 and Fig. 9.6.

For this validation, we conducted a set of experiments with a successful co-manipulation of a 2.5 m long bar. The goal is to validate cooperative manipulation with a MAGMaS for both horizontal displacements of the object and lifting. These basic motions are considered to be representative of the possible construction/decommissioning scenario. The experiment sequence is depicted in Fig. 9.7 and consists in the following: at first the OT-Hex is manually flown to grasp the bar from one of its ends, while the ground robot autonomously grasps the other end. Once both manipulators are attached to the bar the co-manipulation is fully autonomous: they lift the bar from its supports, move it twice along a line in the horizontal plane (blue part) and then synchronously lift the bar up to 30° (green part). Then they bring the bar back to its starting position. This experiment highlights both the vibration stabilization induced by the OTHex and the feasibility of MAGMaS. This is also illustrated in the video related to this exper-

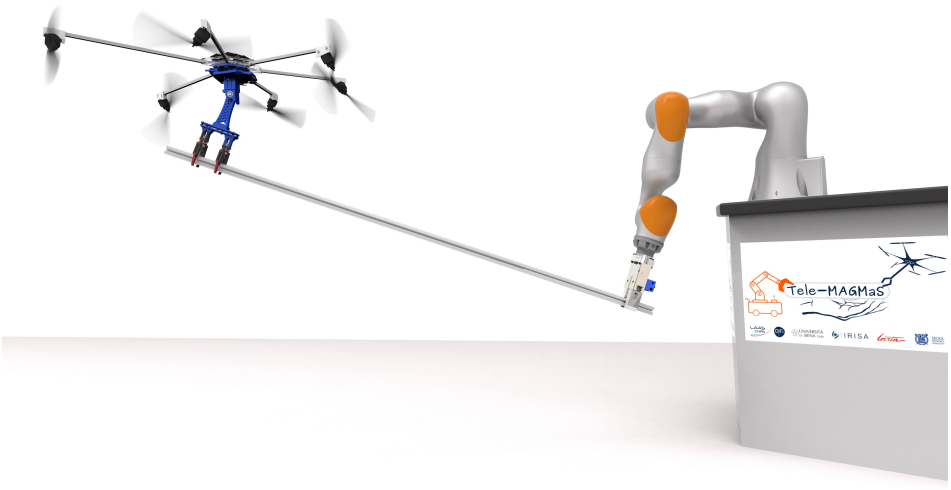


Figure 9.5 – CAD model of the MAGMaS system used for the experimental validation. The ground robot is a KUKA LBR iiwa industrial manipulator, while the aerial robot is the OT-Hex MRAV presented in Sec. 6.1.

iment [video09–2018]. Furthermore, the experimental data related to this experiment (with suitable scripts to plot them) are provided for download at the following link `https://zenodo.org/record/2640461#.XfDhFdF7nkw`. Key quantities of the system are displayed in Fig. 9.8 and in Fig. 9.9. In particular, Fig. 9.9 illustrates that a passive joint is sufficient to complete the task, as the bar motion is guided by the ground manipulator. Note that the small oscillation would have required some active suppression with an actuated joint, leading to more external wrench on the aerial robot side. This implies that the passive joint has a stabilizing effect on the system. In Fig. 9.8 the Cartesian wrench at the ground manipulator end-effector is depicted (top), which transcribes the wrench exchanged between the manipulated object and the ground robot. Note that during the initial lifting of the bar from the stand there is a transient with oscillations which disappears during motion, this arises most likely because the model does not take into account the discontinuity of the object breaking contact with the stand, nevertheless, the system responds in a satisfactory manner. Also, note that the torque generated by the object's weight (τ_y) diminishes during the lifting of the object, as it is proportional to the cosine of the object inclination. The bottom part of Fig. 9.8 depicts the joint torque of the ground manipulator. It is clear that joint torques start to be noisy when the breaks are disengaged ($\approx$30 s). Given the particular configuration of the ground robot, most efforts are furnished by the joint $A2$ (shoulder) and the three joints not aligned with the motion are not solicited ($A1$, $A3$, $A5$). All joint torques are well within their limits.The aerial robot desired trajectory from the admittance filter is presented in Fig. 9.9. The good tracking performances validate our approach. Indeed, thanks to the MDT capability, the error in orientation does not impact the position during the cooperative manipulation.

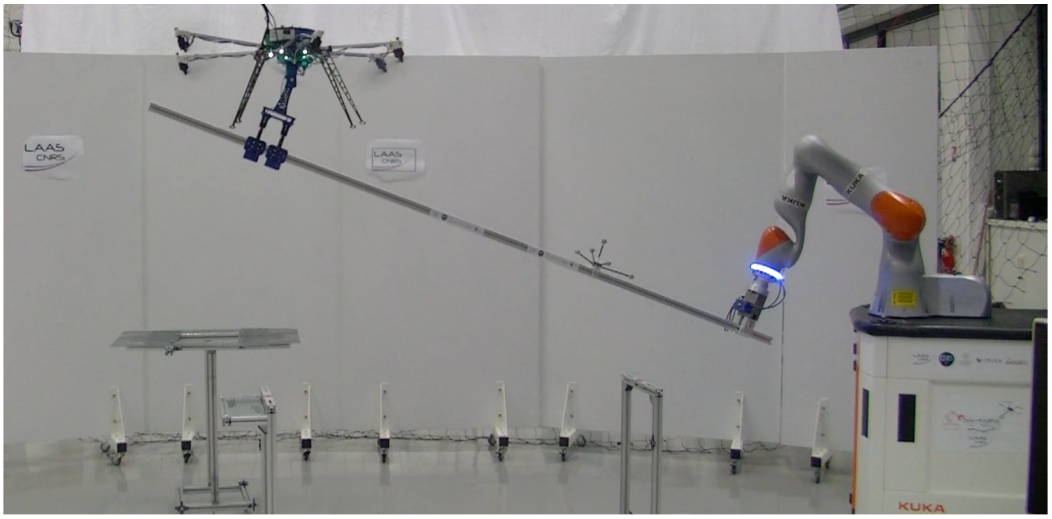


Figure 9.6 – A photo of the MAGMaS during the co-manipulation of a beam.

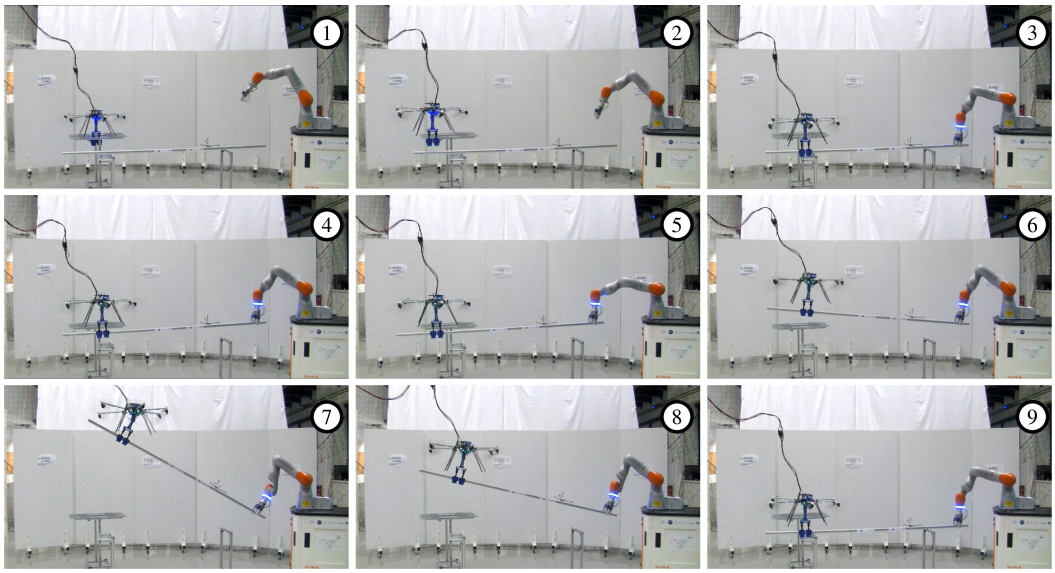


Figure 9.7 – Time-lapse of a MAGMaS cooperative manipulation task. Both robots are at their initial position (1), approach to the bar (2), grasping the bar (3), cooperative lifting (4), cooperative lateral motion (5), cooperative lifting up to 30° (6-7-8) and release of the bar (9).

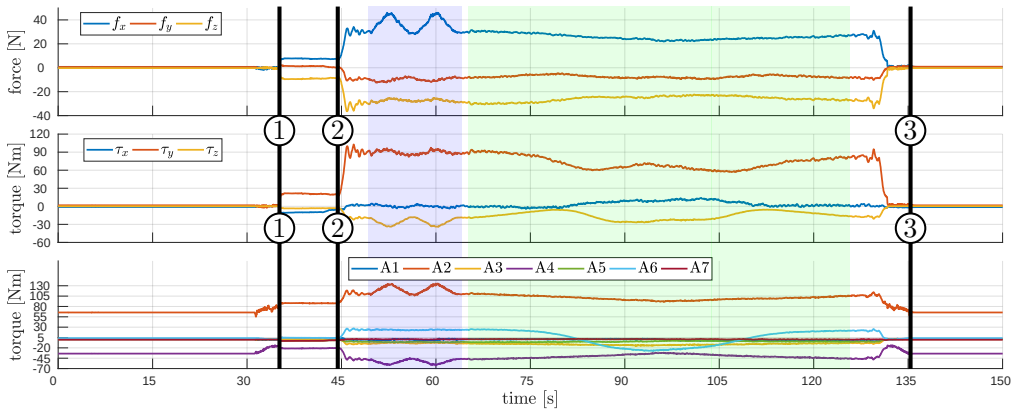


Figure 9.8 – On *top* and *middle*, external wrench as sensed from the joint sensors and projected on Cartesian space. *On the bottom*, joint torques for each articulation of the LBR-iiwa. The three instants highlighted are LBR-iiwa grasping (1), cooperative lifting (2) and LBR-iiwa un-grasping (3). The blue part highlights the horizontal motion and the green part the bar tilting.

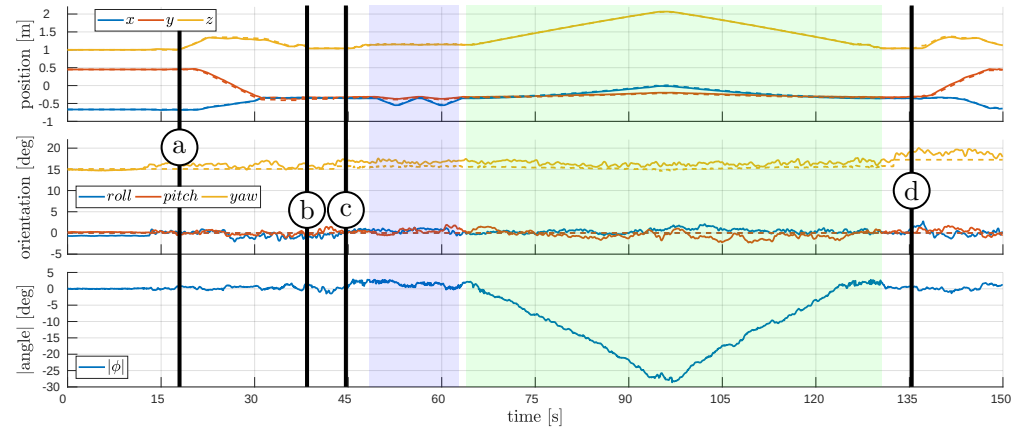


Figure 9.9 – On *top* and *middle*, position and orientation of the OT-Hex aerial manipulator. *On the bottom*, evolution of the OT-Hex passive joint angle, during a typical task with free-flight, horizontal motion and object tilting.The four instants highlighted are take-off (a), OT-Hex grasping (b), cooperative lifting (c) and un-grasping (d). The blue part highlights the horizontal motion and the green part the bar tilting.

Chapter 10

Inspection with contact task

"See now the power of truth; the same experiment which at first glance seemed to show one thing, when more carefully examined, assures us of the contrary."
Galileo Galilei

In industrial facilities, the assessment of the structural integrity is a mandatory process to be performed regularly, especially in sectors like oil&gas and water industries, where the integrity inspection of low carbon steel welds over pipes is very frequent, since cracks or defects can possibly occur. A very important role is played by Non-Destructive Testing (NDT), since it allows assessing the status of an industrial plant without damaging or altering its parts. Among the available NDT techniques for weld inspection, Eddy Current (EC) [GarcíaMartín–2011] is particularly advantageous because it does not require the preparation of the inspecting surface. Furthermore, it is also used for other applications as, e.g., wall-thinning.

Nowadays, the inspection task is typically conducted by human operators that often have to access dangerous areas (e.g., elevated points) with the use of hazardous equipment like climbing ropes or temporary scaffolds. This aspect led to a growing interest in the development and deployment of structural health monitoring solutions [Gasparin–2018] or remotely operated robots. The use of robots also allows generating very useful inspections maps [Miro–2017]. In fact, the precise location of a weld on a pipe is not always known a priori and is neither easily retrievable with visual sensors, especially when pipes are painted or covered by insulating materials.

Inspection tasks with robots are challenging because they require an accurate physical interaction in which the probe has to be kept in contact with, and perpendicular to, a curved surface. This contact-based inspection is only one example of the many other applications requiring robots to slide an end-effector on a curved surface while pushing it against the surface, ensuring the contact. Both the pose of the End-Effector (EE) and the interaction force have to be accurately controlled.

EC inspections of industrial plants with aerial robots is gaining interest in the last years. However, aerial vehicles are still practically employed only for contact-free tasks. This is because of the extremely challenging nature of aerial physical interaction problem, currently under investigation by several research labs and projects like *Aeroarms* [Ollero–2018; AeRoArms–]. The aerial systems proposed to face the physical interaction problem range from Uni-Directional Thrust (UDT) aerial vehicles endowed with rigid or articulated arms [Ruggiero–2015; Kamel–2016], to more recent Multi-Directional Thrust (MDT) [Ryll–2017], as already deeply discussed. The control methods range from decentralized methods to admittance based

methods [Ryll–2017], passing through flatness-based [Tognon–2017] and dynamic-inversion-based methods [Ryll–2018]. The majority of these works present methods to enhance aerial interaction capabilities of aerial vehicles, but only a few addressed real physical interaction tasks although for simple vertical flat surfaces, like [Fumagalli–2014; Alexis–2016a].

In this chapter, we address the aerial physical interaction task of contact inspection, presenting the work that we developed exploiting the MDT Multi-Rotor Aerial Vehicles (MRAVs) that we introduced in Sec. 6.1. In the first section, we present the results of some experiments in which the Tilt-Hex robot is endowed with a rigid tool (see Fig. 8.1 and Fig. 8.5) and is tasked to perform the sliding on a tilted surface (see Fig. 10.1). This experiment epitomizes an additional validation of the force control framework particularized in Chapt. 8 and serves as a entry point for the more complete experiment reported in the second section. In this second experiment, we integrated a 2 Degrees of Freedom (DoFs) manipulator arm on the OT-Hex and perform a contact inspection with a real portion of pipe in order to scan it and detect the soldering profile. In this case, the control of the robot is designed in a different way in order to cope with the flexibility of the arm.

Sliding on surface

In this experiment, we conducted a hard contact between the tool-tip of the rigid end-effector and a tilted wooden surface (see Fig. 10.1). The surface is tilted about $\mathbf{x}_W$ by 10° and about $\mathbf{y}_W$ by -10°. For a better understanding, the tilting of the surface is visualized by orange and green colored bricks in Fig. 10.1. The pre-planned translational trajectory consists out of five phases. First, the desired trajectory approaches without contact from an initial position to a position 0.14 m above the surface (*approaching phase*). Then, it descends from a height of 0.6 m to 0.35 m while piercing through the surface (initial contact point with the surface at 0.46 m) with $0.05\,\mathrm{m\,s^{-1}}$ (*contacting phase*). The desired trajectory then translates along the $\mathbf{x}_W$-axis for 0.4 m with a peak velocity of $0.12\,\mathrm{m\,s^{-1}}$ while the other two axes remain constant (*sliding phase*). After the lateral motion stops the reference trajectory lifts off to its initial height, resulting in a release of the contact (*release phase*). Finally the desired trajectory achieves a stopping position without any further physical contact (*departing phase*). To reduce stick slip effects between the tool-tip and the surface during the sliding-phase the desired orientation of the tool-tip is tilted forward by ($\theta_{R,d} = 7.5^\circ$). To demonstrate the influence of the spring gain $\mathbf{K}_E = \mathrm{diag}[10\ K_{E_y}\ 5]$ and the usefulness of its tuning with respect to the task at hand, the experiment has been conducted twice – first with $K_{E_y} = 2.5\,\mathrm{kg\,s^{-2}}$ and then with $K_{E_y} = 10\,\mathrm{kg\,s^{-2}}$. An illustrative snapshot series of the experiment is depicted in Fig. 10.2.

The results of the first experiment ($K_{E_y} = 2.5\,\mathrm{kg\,s^{-2}}$) are visualized in Fig. 10.3 and in Fig. 10.4. To simplify the understanding of the plots, the *contacting phase* and *sliding phase* have been highlighted with a green background, bordered by

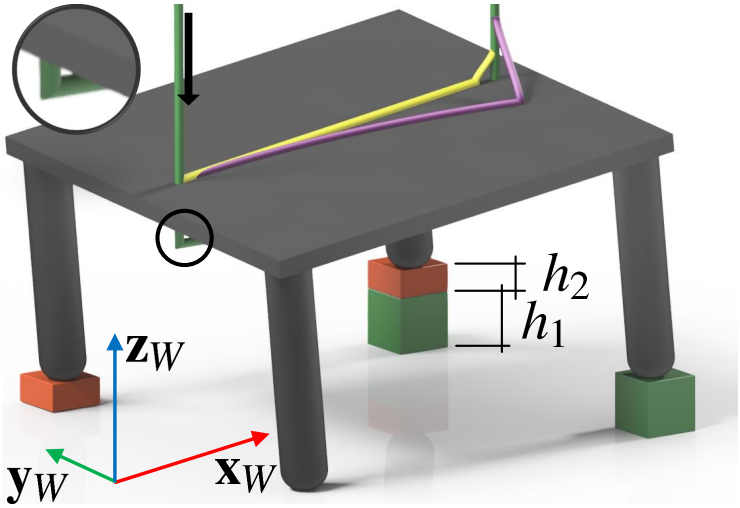


Figure 10.1 – Sliding with constant force application on a rigid, tilted surface. The black arrow indicates the direction of the trajectory. The green line represents the desired trajectory $\mathbf{p}_{E,d}$, which pierces the tilted surface (magnification shows the reference trajectory below the table). The yellow and violet lines represent the two reference trajectories, due to the contact forces, generated by two different values of the admittance parameter $\mathbf{K}_E$, which is related to the virtual spring stiffness.

dashed black lines in all plots. The gray dashed line indicates the moment when the *sliding phase* starts. Let us first discuss the position results. Fig. 10.3-(a) presents the three components of the robot's desired position $\mathbf{p}_{R,d}$ (dashed lines) versus the output of the admittance filter, namely the three components of the reference position $\mathbf{p}_{R,r}$ (doted dashed lines) and the components of actual robot position $\mathbf{p}_R$ (solid lines). During the *approaching phase* the single components of the desired, reference and actual trajectory overlap perfectly. Starting from the *contacting phase* the z-component of the reference trajectory (doted dashed blue line) starts to diverge from the desired trajectory (red dashed line). The desired trajectory pierces the surface, while the reference position remains on the surface of the contact due to the sensed force (see Fig. 10.3-(c)). Fig. 10.3-(b) illustrates the aerial robot's position error ($\mathbf{e}_p = \mathbf{p}_r - \mathbf{p}$). The norm of the position error $\|\mathbf{e}_p\|$ remains marginal, below 2 cm, during all phases of the experiment. Thus the tool-tip is perfectly tracking the reference trajectory. Fig. 10.3-(c) presents a low-pass filtered output of the first three components of the wrench observer as in (8.9), namely the estimated forces $\hat{f}_{E_1}$, $\hat{f}_{E_2}$ and $\hat{f}_{E_3}$. In order to better appreciate the results of this experiment, the reader is referred to the correspondent video [video10–2018].

The moment of contact establishment is easily recognizable with a peak force in $\hat{f}_{E_3}$ (yellow line). Starting from the sliding phase (vertical dashed gray line), smaller forces $\hat{f}_{E_1}$ (blue line) and $\hat{f}_{E_2}$ (red line) are estimated as well. $\hat{f}_{E_1}$ is the force opposing the sliding direction resulting from the increased slope and friction effects between the tool-tip and the surface. The force $\hat{f}_{E_2}$ is a result of the tilted surface

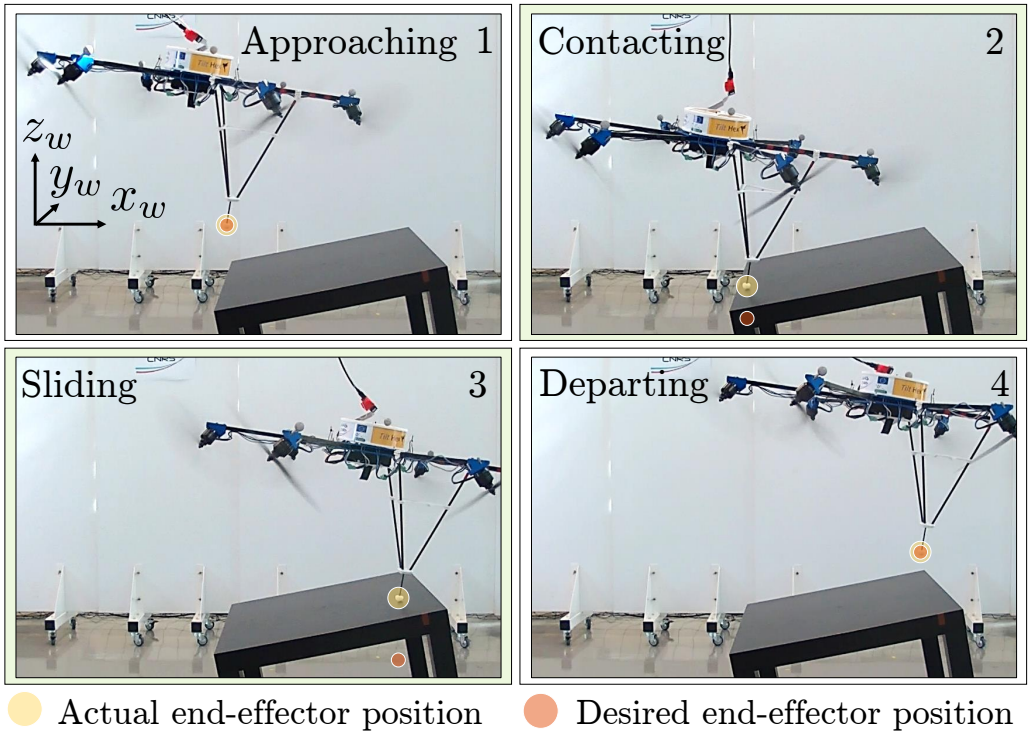


Figure 10.2 – Snapshots of the sliding task: the desired end-effector position (brown circle) and the actual position (yellow circle) are highlighted. The background colors match the colors of the contact phases in Fig. 10.3 and in Fig. 10.4. 1) Approaching the surface. 2.) Establishing contact with surface. The desired and actual end-effector position separate, resulting in the contact force 3.) Sliding along the tilted surface with an increasing distance between desired and actual end-effector position. 4.) Departing the surface. Desired and actual position unite again.

and the contact force $\hat{f}_{E_3}$. Without any position controller the tool-tip would slide down the tilted surface. Fig. 10.3-(d) presents the difference between the components of the pre-planned desired position trajectory and the admittance filter output $(\mathbf{p}_d - \mathbf{p}_r)$. It shows the deflection of the reference trajectory due to the sensed forces $\hat{f}_E$. Fig. 10.4 presents the orientation results of the same experiment. During the *approaching phase* the aerial robot's desired pitch orientation changes from $\theta = 0°$ to $\theta = 7.5°$ and remains constant hereafter. The pitching angle is chosen to achieve a better sliding. During the contact free phase, the reference orientation follows the desired orientation perfectly as only negligible torques are sensed (see last plot Fig. 10.4-(c)). As well, the tracking error between the reference orientation and the actual orientation (see $\mathbf{e}_R$ in plot Fig. 10.4-(b)) remains very small. During the sliding phase, a non-negligible torque $\hat{\tau}_{E_2}$ is estimated, resulting in an additional pitching of more than 2.5° and an total pitch angle θ_R of 10° of the aerial robot. Furthermore, the tilting of the surface about $\mathbf{x}_W$ causes an additional small adaptation of the reference roll trajectory (see plot Fig. 10.4-(d)).

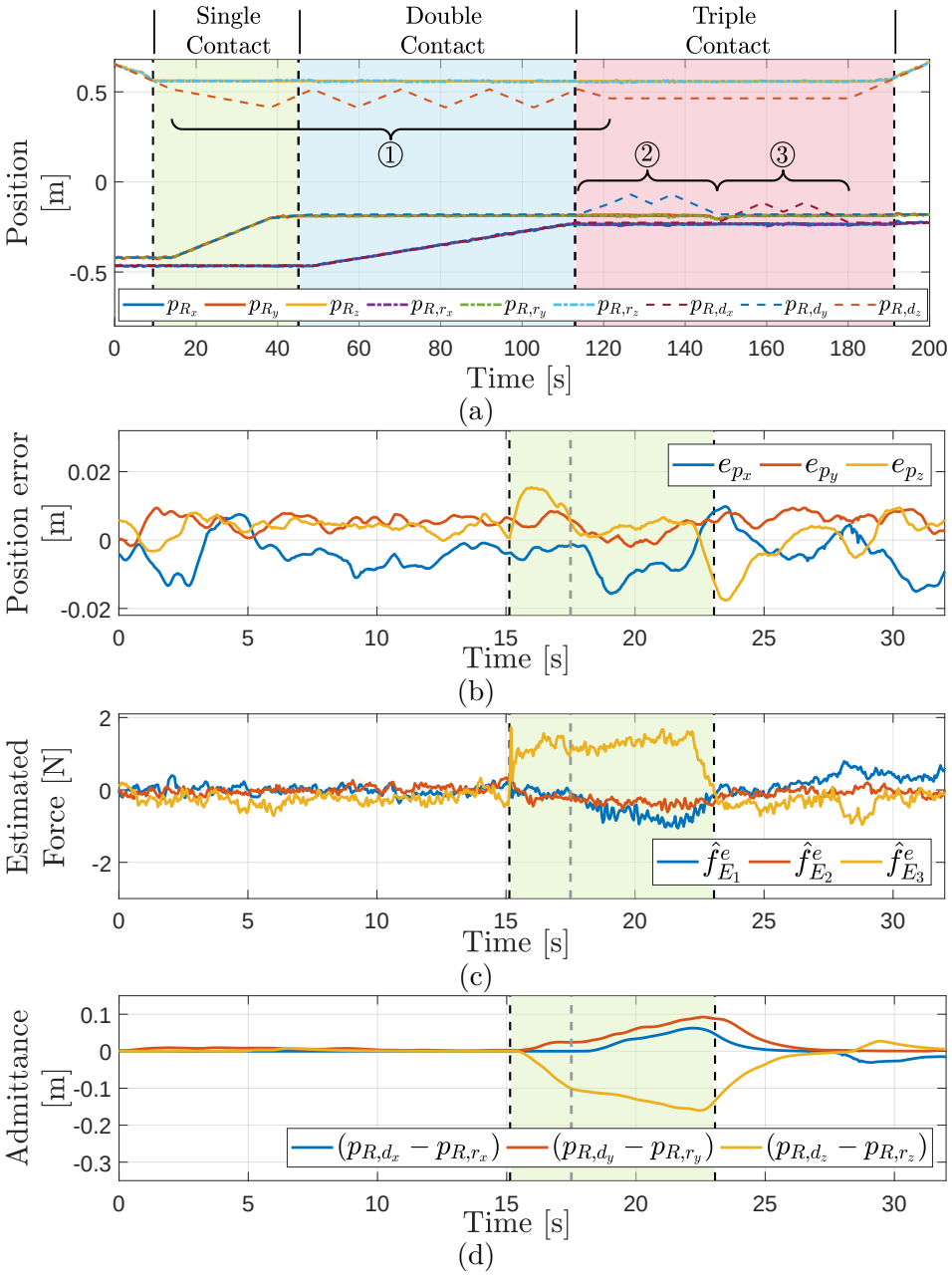


Figure 10.3 – Sliding with the tool-tip on a tilted surface (see Fig. 10.1). The contact phase is highlighted in green, while the actual sliding starts at the gray dashed line: a) Desired, reference and actual position of the tool-tip. b) Actual position error between tool-tip and reference trajectory as $\mathbf{e}_p$ in (5.1). c) Estimated tool-tip contact forces - low pass filtered (8.9). d) Difference between desired trajectory $\mathbf{p}_d$ and the admittance filter output $\mathbf{p}_r$ due to the contact force.

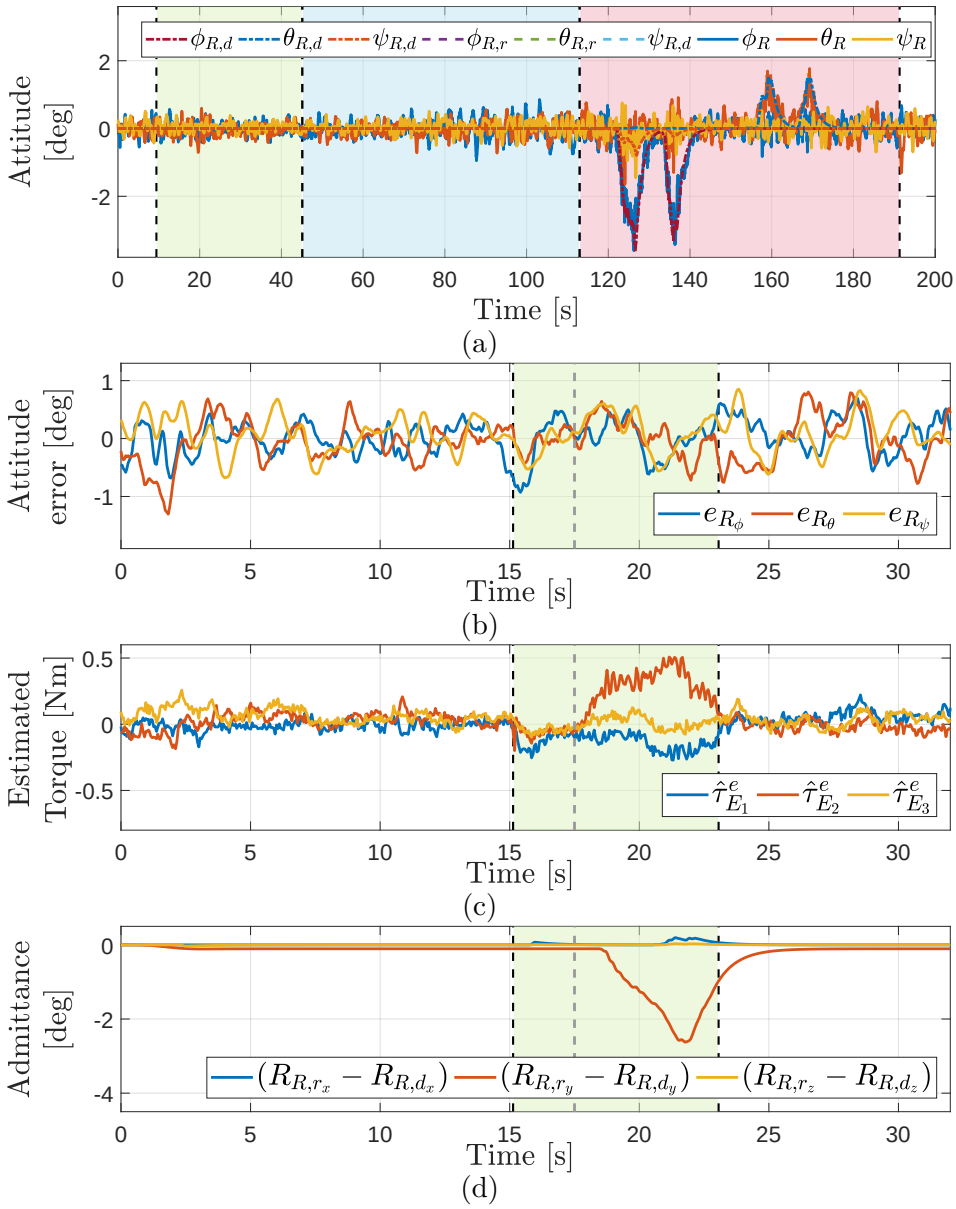


Figure 10.4 – Sliding with the tool-tip on a tilted surface (see Fig. 10.1. The contact phase is highlighted in green, while the actual sliding starts at the gray dashed line: a) Desired, reference and actual tool-tip orientation. b) Actual orientation error between tool-tip and reference trajectory as in (5.5). c) Estimated tool-tip contact torques - low pass filtered output of (8.9). d) Difference between desired trajectory $\mathbf{R}_d$ and the admittance filter output $\mathbf{R}_r$ for convenience expressed in Euler angles.

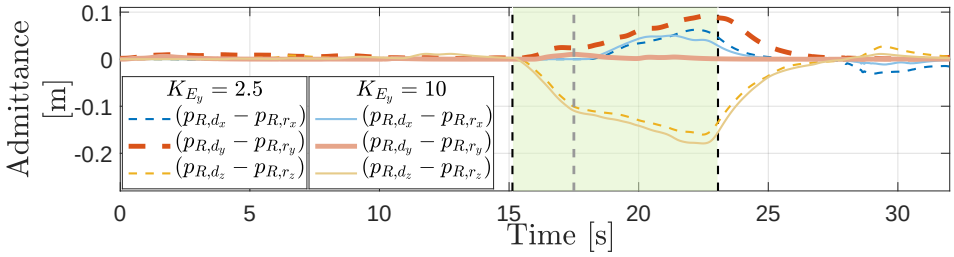


Figure 10.5 – Comparison of the sliding experiment with a low spring gain $k_{E_y} = 2.5$ (dashed) and a high spring gain $k_{E_y} = 10$ (solid line). Only the spring gain along y-component has been altered while the gains along the other two axis remains the same. In the first case, the aerial robot is compliant along y_B (red dashed line) and slides down the tilted surface by almost 0.1 m. In the second case, the tool-tip precisely follows the reference trajectory (red solid line) with negligible deviation. The other two axes (x_B and z_B) behave almost identical in the two cases.

Now we want to make use of the property shaping capabilities of the admittance filter, i.e., the possibility of regulating the behavior of the filter by tuning the parameters $\mathbf{M}_E$, $\mathbf{D}_E$, $\mathbf{K}_E$ defined in (8.11).. In a fictitious task it is desirable that the aerial robot is compliant along the $\mathbf{z}_B$ axis but stiff along $\mathbf{x}_B$ and $\mathbf{y}_B$ and should therefore not glide down the slope of the tilted surface towards $-\mathbf{x}_W$. In the first experiment, the tool-tip diverged from p_{d_y} by ≈ 0.1 m. To achieve a more desirable behavior we increase the gain K_{E_y} in (8.11) from 2.5 to 10. The result and the original values are plotted in Fig. 10.5. It becomes directly clear, that the aerial robot follows a very similar trajectory as before in the $\mathbf{x}$ and $\mathbf{z}$ directions. In the $\mathbf{y}$ direction the aerial robot slid down the slope before. In the second experiment the tool-tip is much stiffer in y direction and does almost not diverge. The difference between the desired and the reference trajectory $(p_{d_y} - p_{r_y})$ is almost zero at all times. Now, the tool-tip of the EE follows a trajectory close to the yellow example trajectory in Fig. 10.1, while before the trajectory was similar to the purple trajectory in the same figure. The comparison between the different behaviors of the robot in the two cases can be appreciated also in the related video [video06–2019].

Finally, it should be remarked that the MDT capability allows the MRAV to perform different aerial physical interaction tasks even with a simple rigid tool, as highlighted in Fig. 10.6, which captures a snapshot of a public demonstration at LAAS–CNRS where the Tilt-Hex was tasked to touch the surface of a mock-up pipe, disposed horizontally on the ground, with the tool-tip. However, the actuation limitations of the platform could restrict the set of the EE feasible poses. With reference to the example of Fig. 10.6, it might be possible to inspect only the upper arch portion of the pipe. This limitation motivates the use of a manipulator arm with more DoFs, that is addressed in the next section.

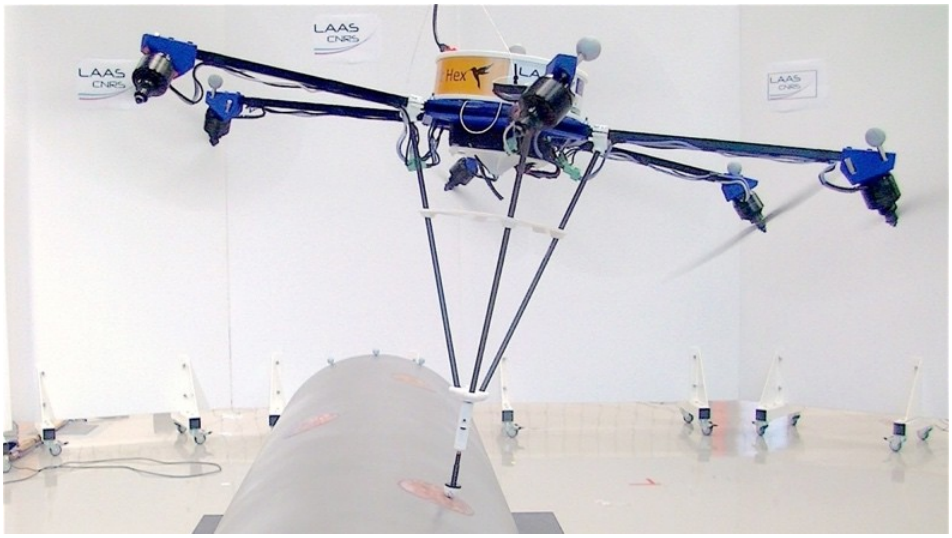


Figure 10.6 – Photo of the Tilt-Hex sliding on the surface of a mock-up pipe by means of a rigid tool. In particular, it is interesting to remark that the aerial robot can hover with a non-flat orientation, thanks to the MDT capability.

Inspection with contact of a pipe

The first contribution of the work presented in this section is to present one of the first complete aerial robotic solutions with sufficient physical interaction capabilities for generic push-and-slide tasks on curved surfaces. Such achievement goes substantially beyond simpler tasks such as pick&place and pull/push objects. The second main contribution is to demonstrate that such a system, endowed with an EC sensor, can successfully inspect a metallic pipe and localize a weld on it. The experiments show that the proposed aerial manipulator is able to autonomously scan the pipe surface sliding the sensor while ensuring the contact and its perpendicularity w.r.t. the surface. From the EC sensor data, the weld is detected and exactly located.

The success of such automatic inspection experiment comes from a wise conception of the aerial manipulator and from the design and integration of state-of-the-art-methods for the EC sensing, motion planning, and control of the robot. Our aerial manipulator is composed of the OT-Hex vehicle endowed with a lightweight arm ending with the EC sensor. The system has been designed in order to have a certain redundancy w.r.t. the task, which grants it a greater dexterity compared to the rigid tool case. The focus of this work is on the integration of design, control, motion planning and sensing system for pipe inspection. For the proof of concept, the experimental validation was conducted indoor using a Motion Capture (MoCap) system. Nevertheless, we degraded the MoCap measurements to emulate the effect of a less accurate localization system.

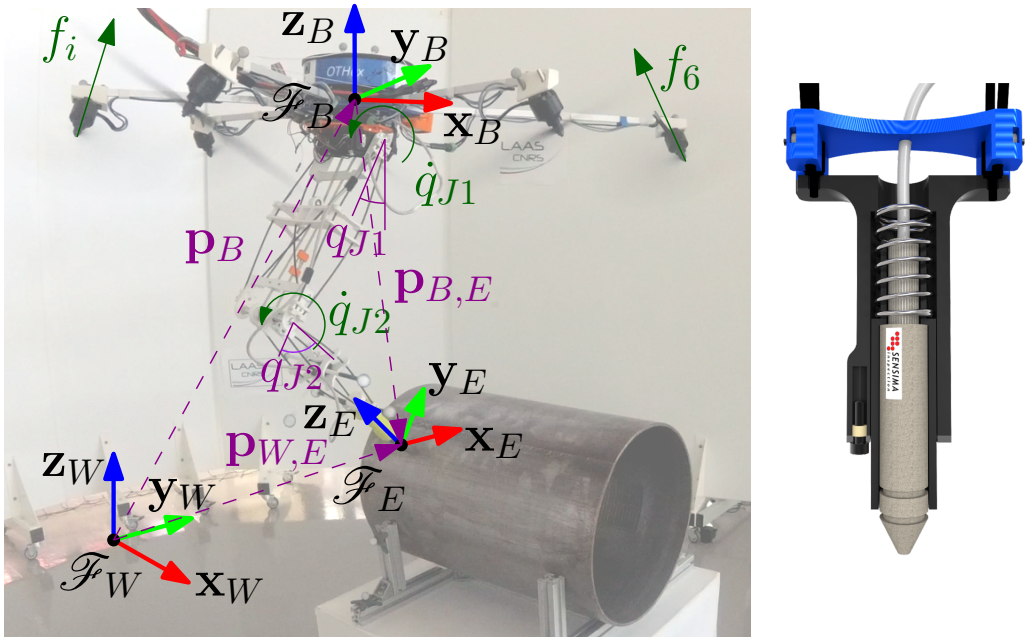


Figure 10.7 – Aerial manipulator with main variables (*left*). Sliced visualization of the compliant EC sensor holder (*right*).

Pipe inspection tools

The instrument used for EC inspection is a *Sensima UPec* kit, visible in Fig. 10.8. This consists of a miniaturized eddy current controller, with wireless connection, and a pen-sized probe to be held by the manipulator. All the data are acquired and processed with the manufacturer application *UPecView* through a PC provided of Bluetooth connectivity. The output of the sensor, $w(t) \in \mathbb{C}$ (the field of complex numbers), is related to the properties of the material in contact with the sensor. The trajectory of $w(t)$ allows recognizing features like a crack, variations of the metal composition, etc. For a more detailed and exhaustive explanation, we refer the interested reader to [GarcíaMartín–2011]. An example of signal is shown in Fig. 10.11, where the signal path relative to contact-free flight and weld detection are highlighted.

In particular, from w, it is possible to retrieve the magnitude of the signal directly linked to the *lift-off* and to the *weld* real signals, denoted by w_l and w_w, respectively. The first is related to the distance from the surface, while the second to the presence or not of a weld. Methods like Principal Component Analysis (PCA) and basis transformation [Jolliffe–2011] are used to extract those two values. The probe is considered

- in contact with the inspected surface if $w_l < \bar{w}_l$;
- on the weld if both $w_l < \bar{w}_l$ and $w_w > \bar{w}_w$.

A preliminary calibration is conducted to define the acceptance thresholds $\bar{w}_l$ and $\bar{w}_w$ for w_l and w_w, respectively. The pipe inspection is conducted performing a *raster*

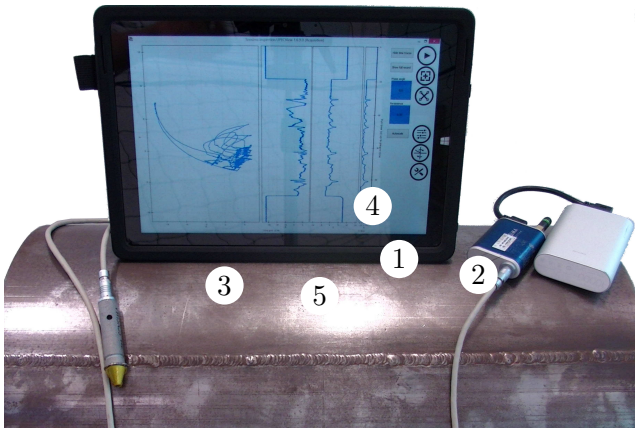


Figure 10.8 – Measurement setup: The eddy current controller *Sensima UPec* (1), its battery (2), and the pen-probe (3) are installed on the robot. The measurements are sent through Bluetooth to a PC (4) and processed by the software *UPecView*. We removed the usual coating/paint that insulates the pipe to make the weld (5) visible. In a real scenario the weld would be detectable but not visible.

scan (or any other types of scans) under some important manipulation constraints to ensure the quality of the measurements. According to the probe model used in this activity, the sensor should be maintained *i)* perpendicular to the surface, with a maximum deviation of around 10 deg, and *ii)* with the sensing tip as close as possible to contact, with an air gap always less than 1 mm. These two important constraints have to be considered at the motion planning and control levels.

Aerial manipulator

The inspection task requires the robot to control both the position and orientation of the EE. Therefore, the robot must have at least 6 DoFs. This directly excludes the use of UDT vehicles equipped with a rigid tool, for which only the EE position and yaw can be independently controlled [Nguyen–2015]. One could endow such an aerial vehicle with an articulated arm with at least 2 DoFs. However, existing control techniques for this type of systems lack of accuracy and robustness w.r.t. disturbances and unknown model parameters.

A recent successful solution for aerial physical interaction, as already discussed, is the use of MDT platforms, like the one proposed in [Park–2016]. Such a capability allows the aerial platform exerting, in a certain extent, a six-dimensional wrench in a decoupled fashion [Ryll–2016], permitting to independently control the position and orientation of the robot, and to balance external disturbances almost instantaneously and without the need of reorienting the body, as instead needed by UDT vehicles. In our previous experiment in Sec. 10.1, we showed that a MDT vehicle can successfully interact with the environment by means of a simple rigid tool. However, the input limits restrict the admissible orientation of the platform, and in turn the one of the EE. Considering the previously described task, this limits

the area that can be inspected with such robots. To enlarge the feasible scanning area to the full external surface of a pipe we added a planar 2-DoFs lightweight arm ending with a compliant support for the sensor. An image of the robotic system is shown in Fig. 10.7. For more details regarding the lightweight arm, we refer the interested reader to [Tognon–2019]. The overall system is characterized by 8 DoFs, which makes it redundant w.r.t. the inspection task. This redundancy can be also exploited during motion planning to choose the best configuration in terms of input feasibility, manipulability, and energy efficiency, and to avoid surrounding obstacles (e.g., see [Tognon–2018a]).

In order to model the entire aerial system, we denote by $\mathbf{q} = [\mathbf{p}_R^\top \ \boldsymbol{\eta}_R^\top \ \mathbf{q}_A^\top]^\top$ and $\boldsymbol{v} = [\dot{\mathbf{p}}_R^\top \ \boldsymbol{\omega}_R^\top \ \dot{\mathbf{q}}_A^\top]^\top$ the configuration and the corresponding velocity vector of the aerial manipulator. Its dynamics can be formulated using the Lagrangian methodology in the following form:

$$\mathbf{M}(\mathbf{q})\dot{\boldsymbol{v}} = \mathbf{c}(\mathbf{q}, \boldsymbol{v}) + \mathbf{g}(\mathbf{q}) + \mathbf{G}_m(\mathbf{q})\mathbf{u}_m, \tag{10.1}$$

where $\mathbf{M}(\mathbf{q}) \in \mathbb{R}^{(6+2)\times(6+2)}$ is the positive-definite inertia matrix, $\mathbf{c}(\mathbf{q}, \boldsymbol{v}) \in \mathbb{R}^{(6+2)}$ is the vector collecting the centrifugal and Coriolis forces, $\mathbf{g}(\mathbf{q}) \in \mathbb{R}^{(6+2)}$ represents the gravitational term, $\mathbf{G}_m \in \mathbb{R}^{(6+2)\times(6+2)}$ is the input allocation matrix, and $\mathbf{u}_m = [\mathbf{f}_R^\top \ \boldsymbol{\tau}_R^\top \ \tau_{A_1} \ \tau_{A_2}]^\top \in \mathbb{R}^{(6+2)}$ is the vector containing all the inputs of the aerial manipulator. A more detailed derivation can be found in [Ryll–2018].

Motion planning and control

To properly identify and map a weld, one solution is to perform a *raster scan* over the surface of the pipe. A task-constrained motion planner is required to let the end effector obey the aforementioned specifications and also take into account the dynamics of the aerial manipulator, as well as its kinematics and input limits, considered to be the actuator forces, in this case. For these reasons, we use the *Control-Aware Motion Planner* presented in [Tognon–2018a]. It is a kinodynamic task-constrained motion planner based on a tight combination of motion planning algorithms and control methods, that was customized for MDT aerial manipulators. For the details, we refer the interested reader to [Tognon–2018a]. The planner provides the nominal trajectory of each DoF of the aerial manipulator, $(\mathbf{p}_R(t), \boldsymbol{\Theta}_R(t), \mathbf{q}_A(t))$, and, as a consequence, the EE desired trajectory, $(\mathbf{p}_E^d(t), \mathbf{R}_E^d(t))$. Such trajectories respect the task-constraints, the dynamic constraints, and the actuator force limits of the system.

As far as the controller of the whole aerial manipulator is concerned, the direct use of the techniques presented in Chapt. 8 reveals not possible in this case. Indeed, the low stiffness of the lightweight arm implies deformations that cannot be easily identified and considered in the model. In cases like this one, where a precise force feedback is not available, the common approach is to design a compliant behavior of the EE and to plan its trajectory slightly "inside" the surface of interest, as for impedance/compliance control [Wit–1996]. Thanks to the compliance, the position

error will make the system slightly push on the surface preserving the contact. The presented aerial manipulator has three sources of compliance: *i)* the PD-based pose control of the aerial vehicle (explained in the following), *ii)* the natural flexibility of the arm, and *iii)* the spring in the sensor support. The last was added to absorb most of the interaction force, avoiding to stress the arm too much, thus reducing the risk to break it. Given the redundancy of the system, the pose control of the EE could be done with a standard *Inverse Kinematics Control* [Ruggiero–2015]. However, we experimentally noticed that, due to the flexibility of the arm, a full feedback from the EE makes the system unstable. In view of this, we naturally took inspiration from the state-of-the-art on control of manipulators with elastic joints [De Luca–2008]. Comparing our situation with a manipulator with elastic joints, we can consider the OT-Hex as a 6-dimensional "motor" and the arm plus the compliant sensor holder as a single link connected to the "motor" through 6D passive elastic joint. Inspired by this analogy, we decided to apply a "dislocated" PD control law, i.e., a PD with a mixed feedback strategy. In fact, it has been proved that for a 1-DoF manipulator with an elastic joint in the one-dimensional case, a feedback action entirely based on the EE position and velocity leads to instability, no matter the gain values. On the other hand, a mixed feedback from the link position and motor velocity ensures stability if the value of the proportional gain is lower than the stiffness of the mechanical elasticity. The interested reader in the details of the controller is referred to [Tognon–2019].

Experiment

The following experiment – reached after a series of simulations and other experimental tests – is aimed to demonstrate the effectiveness of the proposed aerial manipulator system to perform inspection tasks requiring physical contact. A detailed description of the software framework and some snapshots of the experiment are available in the technical report added to the multimedia materials.

The goal of the experiment is to identify and localize a weld on a portion of metallic pipe (see Fig. 10.7 and Fig. 10.8) of length 0.5 m and diameter 0.4 m, using an EC probe. The inspection is performed on a section of the pipe by means of a raster scan path passing over the weld 4 times. Giving the set of way-points as input to the proposed motion planner, we obtain the desired EE trajectory and the nominal trajectories of the robot DoFs shown in Fig. 10.9 and Fig. 10.10, respectively. The contact-inspection trajectory has a duration of 110 s (only the contact phase) and is planned in around 9 s on a standard laptop.

The overall experiment is composed of three parts: *i)* approach to the pipe (arm folded and sensor not in contact); *ii)* contact-based inspection; and *iii)* departure from the pipe. Here, we focus our analysis on the most interesting part *ii)*). For this part, Fig. 10.9, Fig. 10.10 and Fig. 10.11 show the behavior of the robot and the sensor data, from right before contact (time 25 s) to right after contact (once the inspection is over, time 144 s).

From Fig. 10.9 and Fig. 10.10, one can appreciate the accuracy of the robot to

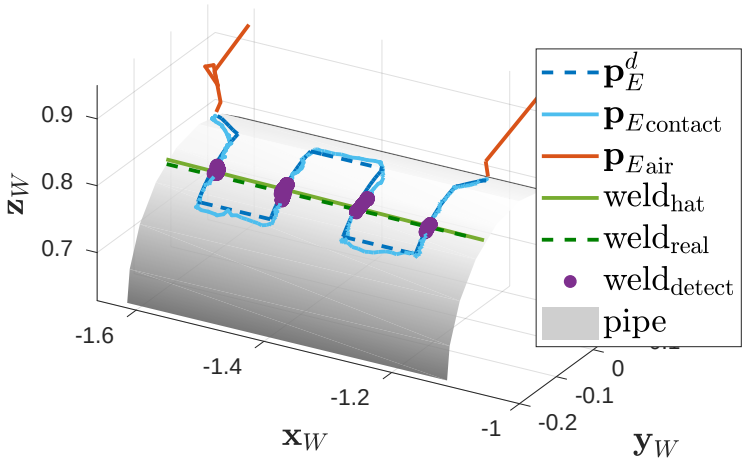


Figure 10.9 – Representation of the desired and actual trajectories of the EE over the pipe. The purple dots highlight the points in which the weld is detected.

track the desired EE trajectory keeping always the contact (confirmed by the value of w_l). This is done despite the presence of many uncertainties, elasticity, frictions, and no force feedback. The presence of errors in the kinematic model is confirmed by the non-zero error between the nominal and actual pose of the aerial vehicle.

Notice that once the robot gets in contact with the surface, the (unknown) interaction force exerts an extra torque on the aerial vehicle, which implies an initial orientation error. This error goes to zero in few seconds as a result of the decoupled control of the orientation and the position. During the transient, this inaccuracy induces a small error in the EE orientation too. The latter is defined by the vector $\mathbf{e}_{E\eta} = [e_{E\phi} \;\; e_{E\theta} \;\; e_{E\psi}]^\top$ where $e_{E\phi} = \phi_E^d - \phi_E$. Analogously for $e_{E\theta}$ and $e_{E\psi}$. For display purposes we show $(\psi_E, \theta_E, \phi_E)$ as the Euler-angles describing the orientation of $\mathcal{F}_E$, following the convention Z-Y-X. The EE pose error always respects the task requirements. $\mathbf{e}_{E\eta}$ shows the deviation of the EE orientation from being perpendicular to the surface. The perpendicularity of EE is further shown in the related video [video10–2018].

Fig. 10.11 shows the raw signal $w(t)$ coming from the EC sensor. It is interesting to notice its evolution when the probe passes from air to contact and over the weld. From $w(t)$, applying statistical methods like the mentioned PCA, we can retrieve the more informative signals w_l and w_w (shown in Fig. 10.10). The contact phase can be identified by looking at when $w_l < \bar{w}_l = 15$. On the other hand, looking at w_w for $w_w > \bar{w}_w = 1.4$ and $w_l < \bar{w}_l$, we can identify when the probe is in contact with the weld. The thresholds $\bar{w}_l$ and $\bar{w}_w$ were identified in a preliminary calibration phase. In all the plots, we highlight the no-contact and contact-with-weld phases with red and purple colors, respectively.

Combining w_w with the measured position of the probe, we retrieved an estimation of the weld position all along the surface, using simple linear regression. The estimated and real weld positions, defined by $\text{weld}_{\text{real}}$ and weld_{hat}, are shown in Fig. 10.9 by green solid and dashed lines, respectively.

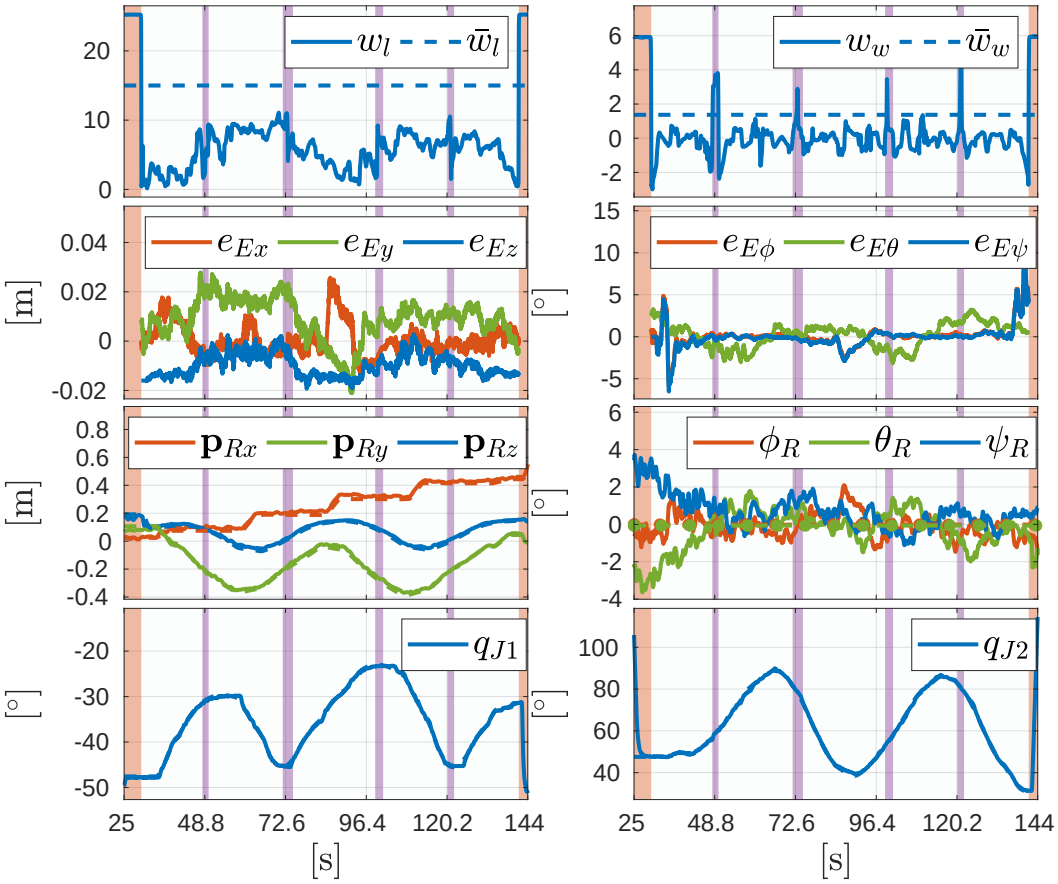


Figure 10.10 – Evolution of the main variables. The dashed lines in the four bottom plots represent the nominal trajectories given by the motion planner. For convenience, the aerial vehicle orientation is displayed by the Euler-angles $(\psi_R, \theta_R, \phi_R)$ following the convention Z-Y-X.

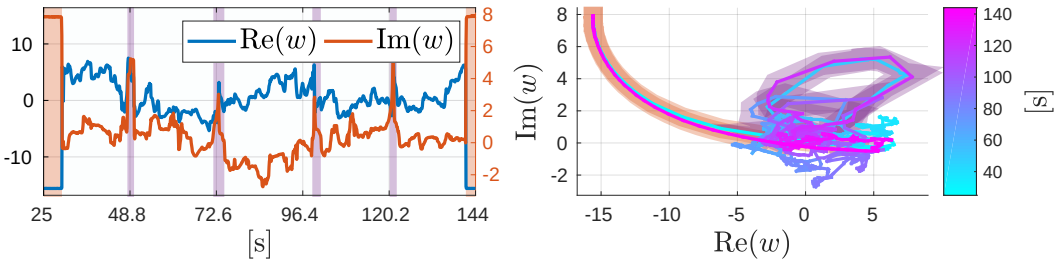
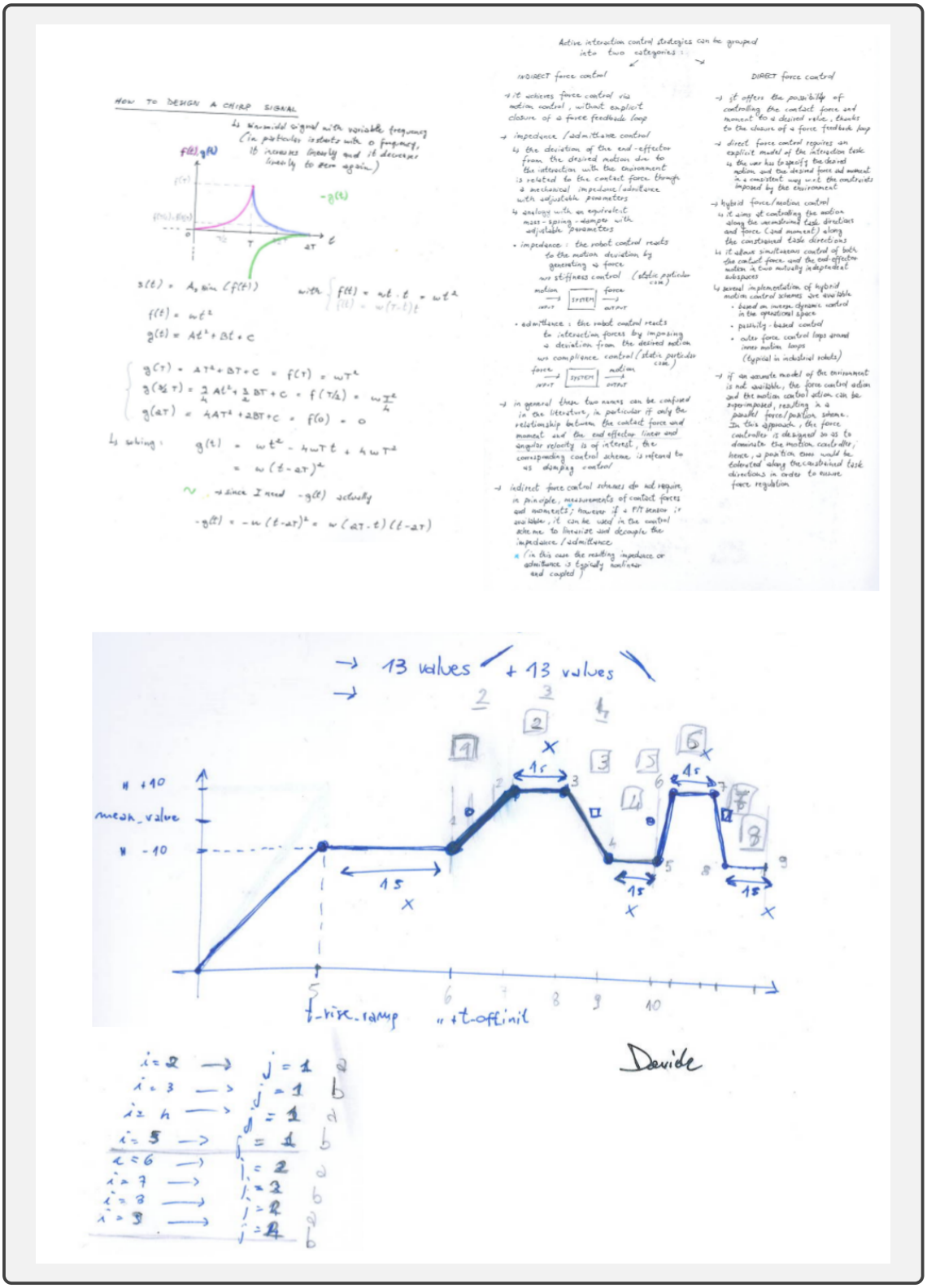


Figure 10.11 – On the *left*, acquired raw data w, showing its real and imaginary parts and its evolution in $\mathbb{C}$ (complex numbers). On the *right*, the color represents the time.

In view of a future integration with a Global Positioning System (GPS) and a vision-based system for outdoor experimentation, we tested the proposed robotic system with degraded Motion Capture System (MoCap) measurements. We reduced the state-update frequency down to 30 Hz and added a white noise signal of standard deviation equal to 1 cm and 2 deg to the position and orientation components of the MoCap measurement (comparable to an outdoor localization system based on differential GPS and cameras). Under this condition, the tracking errors obviously increase, but remain sufficiently small to perform the desired task, preserving the contact between the sensor and the pipe surface. We tested the system also in the presence of wind, produced by a fan blowing air on the robot at a speed of around 6 m/s. This disturbance has almost no effect on the quality of the results. To appreciate these results, the interested reader is addressed to [Tognon–2019] and the related technical report.


HOW TO DESIGN A CHIRP SIGNAL
Active interaction control strategies can be grouped into two categories:
INDIRECT force control
DIRECT force control
13 values + 13 values
t_risk_ramp
t_off_init
Davide

Drawing III

Part IV

Technology transfer and Conclusion

Chapter 11

Industrial technology transfer

"Science knows no country, because knowledge belongs to humanity, and is the torch which illuminates the world."
Louis Pasteur

As already mentioned, the work presented in this thesis has been developed within the scope of [AeRoArms–]. This project brings together the knowledge in the fields of mechatronic design, control, planning, perception and navigation of different European partners, forming a heterogeneous consortium composed of universities, scientific laboratories, and also companies. The integration between academic and industrial institutions mirrors the combination of research and technology advancements pursued by the project. Thus, the goal of this chapter is to present the technological applications and products that arose from a wise integration of the theoretical concepts presented so far in this manuscript, with the collateral intent of underlining the impact of this work also in the industrial field. In particular, it is worthwhile to remark that the expertise developed at LAAS–CNRS regarding the design and the control of Multi-Directional Thrust (MDT) Multi-Rotor Aerial Vehicles (MRAVs) has been exported in the consortium and has influenced, directly or indirectly, the work of some other partners. This led to development of other two aerial robot prototypes, this time with a much higher level of Technology Readiness Level (TRL) compared to the hexarotor platforms presented in this thesis. Before presenting these industrial robots, we will briefly outline an interesting application, also resulting from international collaborations with other partners, of an enhanced version of the MAGMaS system introduced in Sec. 9.2. Ultimately, the results presented in this chapter should possibly underline the importance and the fruitfulness of scientific collaborations for the achievement of relevant advancements in the research and the technology fields.

Tele-MAGMaS at the 2017 KUKA Innovation Award

The manipulation of large size objects by robotic systems is a challenge for applications in the field of construction industry, industrial decommissioning, and urban search and rescue. These are associated to dangerous environments and thus motivate the need of devising robotic solutions for replacing human presence. Furthermore, they often require manipulation of long objects, such as pipes, bars, beams and metal frameworks, with limited access to their Center of Mass (CoM).

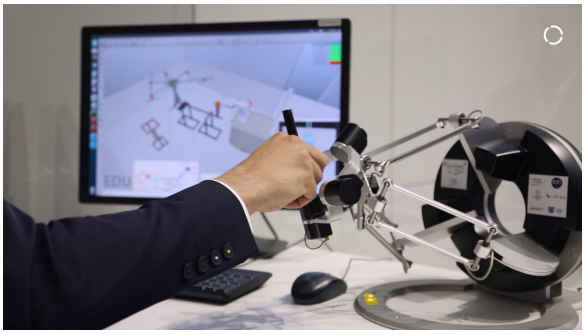


Figure 11.1 – Photo of a human operator controlling the state of the simulated Tele-MAGMaS system with an Omega.6 haptic device.

In this section, we briefly present the work we have developed in the scope of an international competition of robotics, i.e., the 2017 KUKA[1] Innovation Award[2]. This competition is a very competitive worldwide award sponsored by KUKA to stimulate high quality and innovative research in the domain of human robot co-operation and manipulation. Our team, collecting researchers from LAAS–CNRS (Toulouse, France), IRISA-Inria (Rennes, France), University of Siena (Siena, Italy) and Seoul National University (Seoul, South Korea), has been selected among the five finalists of this competition, within almost forty teams from all over the world. The final phase of the award consisted in live demonstrations at the Hanover Fair 2017. The team name has been chosen to be *Tele*-MAGMaS, where MAGMaS stands for 'Multi Aerial Ground Manipulator System' and designates the kind of system selected for the competition, while *Tele-* reflects the tele-operation capabilities with which the system has been enhanced. Indeed, this new system also allows for remote human operation at different autonomy levels. For a detailed descriptions of this point, we refer the interested work to [Staub–2018b].

The presence of a human operator, who owns superior intelligence and cognitive capabilities, is necessary in most of the realistic applications to cope with unknown/-partially known environments with possibly unpredicted events. In the proposed Tele-MAGMaS scheme, the human operator is also provided with haptic feedback to enrich her/his tele-presence and improve her/his performance. A haptic device (or haptic interface) is a robot that works in the master side of a tele-operation system. The human operator can move the master robot by applying force to its handle and reversely the haptic interface can apply forces to the human operator, called "haptic cues" or "haptic feedback." Haptic feedback is used to provide the human operator with information about the state of the remote system. In our testbed, we used an Omega.6 haptic device, manufactured by force dimension[3], as shown in Fig. 11.1, which has six Degrees of Freedom (DoFs).

The Tele-MAGMaS demonstration represents an evolution of the task presented

[1] `https://www.kuka.com/en-de`

[2] `https://www.kuka.com/fr-ch/technologies/research-and-development/kuka-innovation-award/kuka-innovation-award-2017`

[3] `http://www.forcedimension.com/`

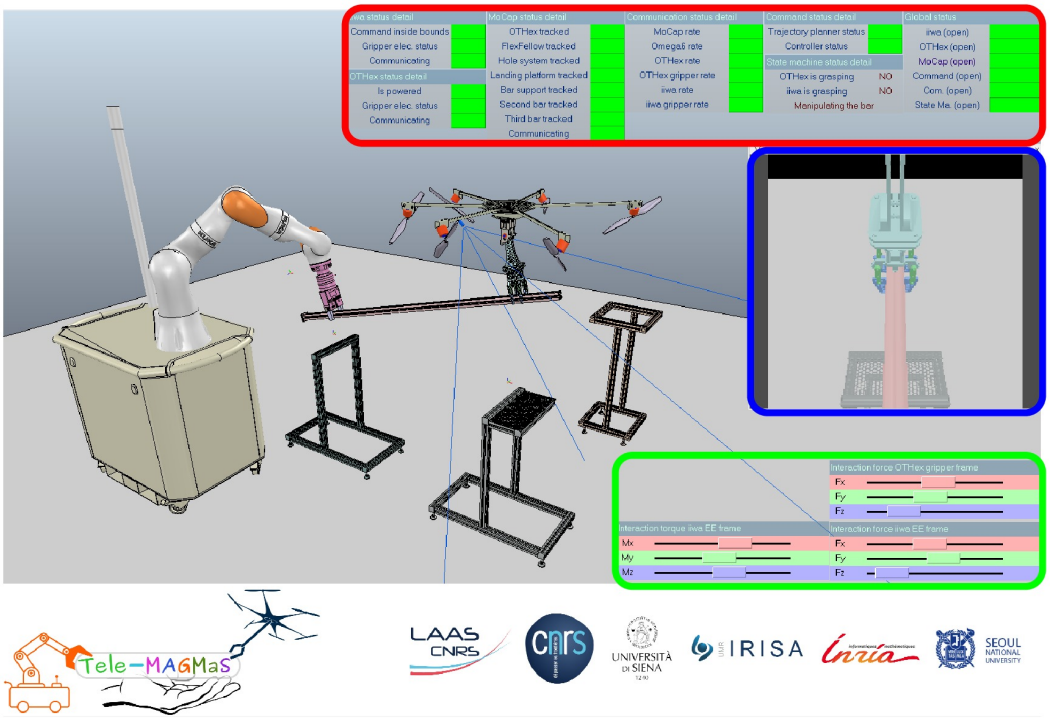


Figure 11.2 – Display of the 3D scene and graphical user interface within V-REP. In red the sub-system status indicators , in blue the emulated FVP camera on the OT-Hex, in green the visualization of the interaction forces and torques.

in Sec. 9.2, i.e., the cooperative manipulation of a long metallic beam by means of a ground manipulator, a KUKA iiwa in the specific case, and an aerial robot, the OT-Hex (see Sec. 6.1). To the best of our knowledge, this is the first time a flying assistant has been implemented and robustly demonstrated in live experiments. Apart from the addition of the tele-manipulation functionalities associated with the haptic interface, we also improved the previous demonstration by implementing a simulation environment to test the behavior of the system, and also a graphical user interface to extend the human situation awareness. Simulation has to trade-off accuracy of the simulated behavior and computational load, weighted by the real-time constraint imposed by the human-in-the-loop control scheme. The chosen compromise is a multi rigid-body dynamics integration with collisions processing and re-configurable joints. The physics simulation is handled by the software V-REP[4], a robot simulator providing a design environment and incorporating the simulation library Bullet[5]. As the overall controller of the system relies on MATLAB-Simulink®, a bridge enabling bilateral communication between V-REP and the controller has been implemented, exploiting the Remote API functions, which are integrated within S-functions in Simulink, thus providing *data transceiver blocks*. Additionally, a graphical user interface was implemented for providing some teleme-

[4] `http://www.coppeliarobotics.com/`
[5] `http://bulletphysics.org`

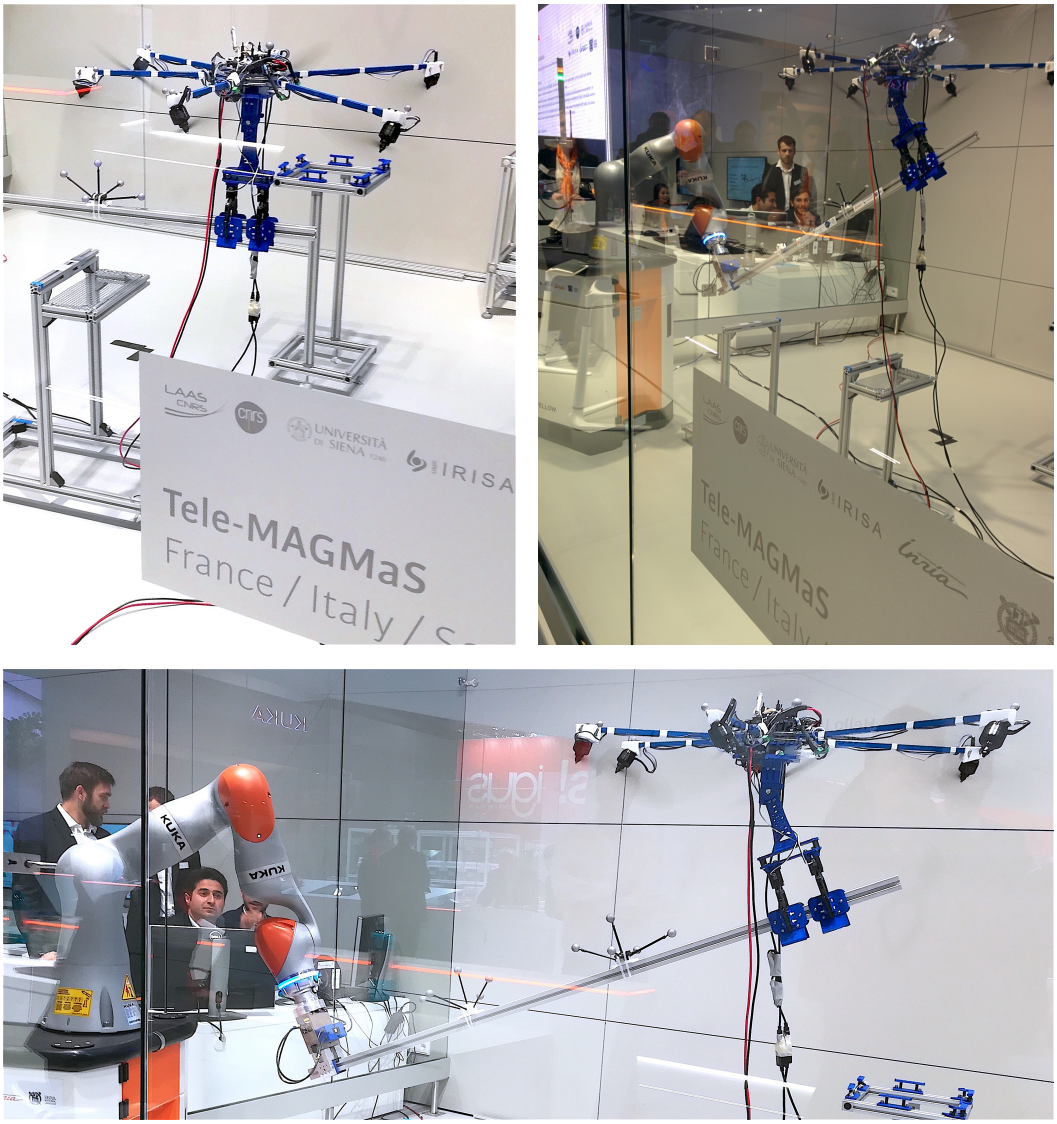


Figure 11.3 – Snapshots of the cooperative manipulation state of Tele-MAGMaS performed during the 2017 KUKA Innovation Award at the Hanover Fair.

try to the operator, both in simulations and during real experiments, mainly for monitoring and debugging purposes. The displayed information is a visual feedback, via emulated camera mounted on the OT-Hex, the interaction wrenches and general meta-data about the sub-systems state. Each of these items appear in a tailor-made OpenGL sub-window layered on V-REP's main screen. Considering the large amount of possible data to be shown, we have chosen to set up a hierarchical user interface, which allows the user to choose the level of detail. The operator is informed about the status of the system and sub-systems by means of color-changing indicators. Each detailed sub-window can be folded to reduce the visual load on the screen, see Fig. 11.2.

The Tele-MAGMaS first experiments were showcased at the Hanover Fair 2017, as part of the finals of the KUKA Innovation Award, see Fig. 11.3. The demonstration presented the main features of the system and a proof-of-concept application. During the fair week, the demonstration was running every hour (or more), thus demonstrating the high reliability of the proposed system and control architecture. Videos highlighting the key features of the demonstration, i.e., cooperative aerial-ground manipulation with human tele-presence, can be found online [video11–2017; video13–2017; video12–2017].

DLR Suspended Aerial Manipulator

The first industrial platform discussed in this chapter is the SAM, whose design has already been partially introduced in Sec. 6.2. This aerial robotic system is the result of a joint work between DLR[6] and LAAS–CNRS regarding the design of the actuator arrangement and some preliminary tests, while the conception of other technology features and the complete platform construction and assembly has been handled by DLR and the Robo-Technology[7] company. The complete platform design has been patented by DLR. In this context, we briefly discuss the integration of some other features that have been included in the final robot implementation.

Winches

The SAM actuation system has been enhanced with three cables, attached to the platform frame as shown in Fig. 11.4. Each cable is actuated by a winch, i.e., is a mechanical device that can control the length of the wire rope. While winches are used to compensate for slowly-varying displacements of the manipulator CoM, due to the manipulation task, the propulsion units can be used to reduce dynamic deviations. Such a strategy allows to reduce the amount of energy consumption since no continuous torque is required from the propulsion units. Each winch consists of

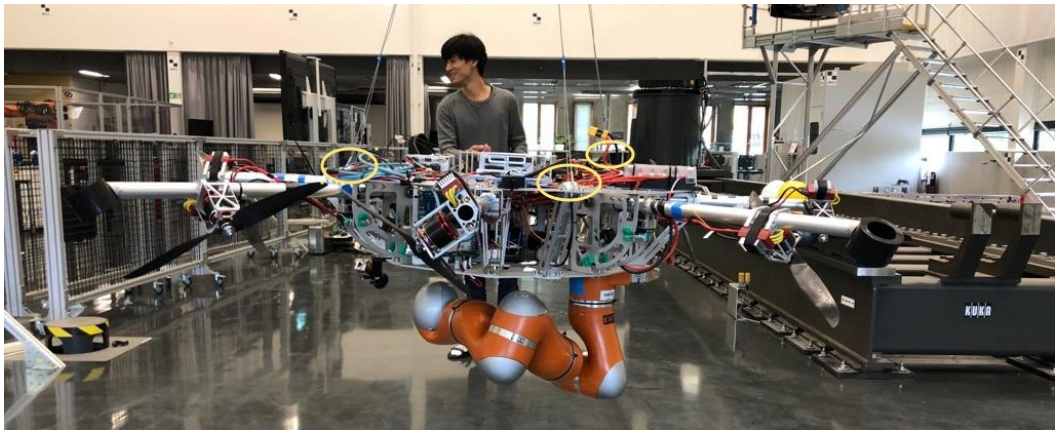


Figure 11.4 – Photo of the SAM with detail on the winches attachment points.

[6]German Aerospace Center, `https://www.dlr.de/dlr/en/`

[7]Robo-Technology GmbH, `https://www.robo-technology.de/robo/en/`

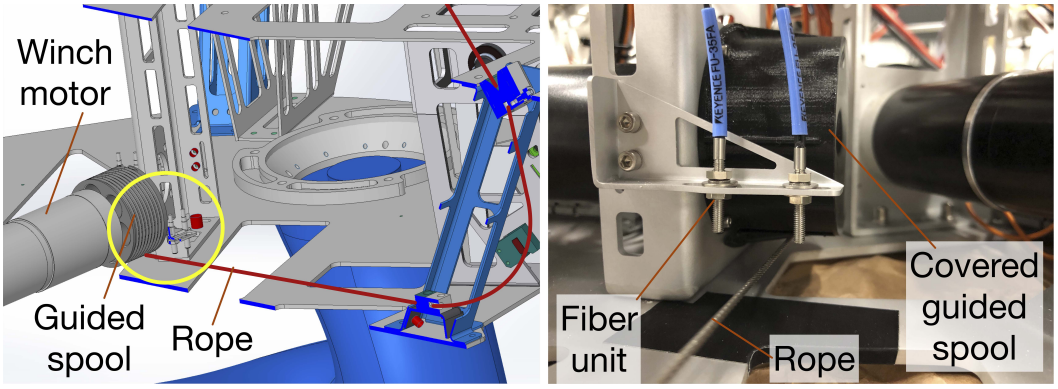


Figure 11.5 – Winch structure: the whole operation scheme (*left*), and detection system (*right*), located inside the yellow circle in left subfigure.

a DC-motor with installed guided spool and cover (see Fig. 11.5). The wire rope is wound on the spool. Due to the presence of the brakes, no energy is required to keep the rope winding angle during operation work. The SAM contains three Maxon[8] motor-based winches which help to maintain the CoM below the hanging point via controlling three DoFs, i.e., the roll, pitch, and height dynamics of the platform. Control of the height using winches allows us to completely exclude aerial carrier from the aerial manipulation. Optical fiber sensing system from Keyence[9] is used in order to remain within the operational point of the guided spool and to calibrate winches. By sensing light beam interruption and reflection, this system allows to detect an approaching rope to a reflection fiber unit (light source), see Fig. 11.5 (right). In the calibrated configuration, wire rope is located in the middle between two light sources. The total length of the winch wire rope in the workspace of the spool is about 1.5 meter. Finally, it is very interesting to remark that the high control authority around the yaw axis, obtained as a result of the optimization of the propeller arrangement in Sec. 6.1 (in particular, see Fig. 6.21), is very useful since this DoF cannot be exploited from the winches control.

Electronics, sensors, and communication

The architecture of the SAM is presented in Fig. 11.6. As it can be seen, three different computers are installed onboard: robot control computer, machine vision computer, and flight control computer. The flight control computer is responsible for the control of the winches and propulsion units. Also, it is directly connected to the set of sensors for estimation of platform state during the operation: Inertial Measurement Unit (IMU), Global Positioning System (GPS) with real-time kinematics, commonly known as Real-Time Tinematic (RTK), and two cameras for the machine vision system. It is worth noting that cameras have a dual-use. Their second purpose is to provide visual feedback to the human operator during tele-operation tasks. The connection between computers is established via a switch,

[8] `https://www.maxonmotor.com`
[9] `https://www.keyence.com/`

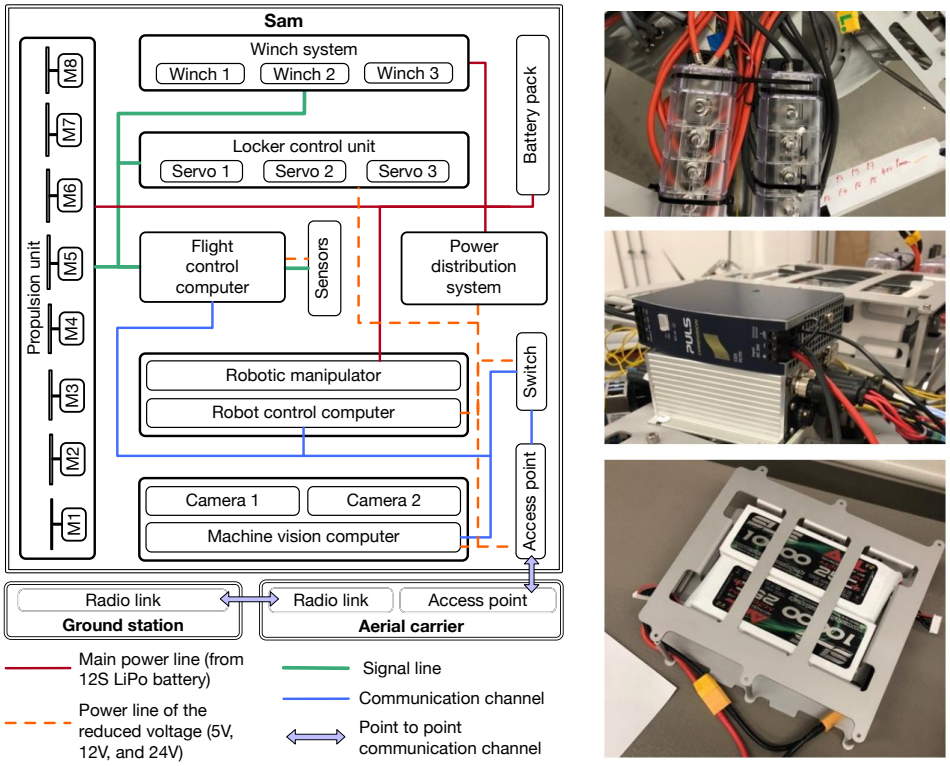


Figure 11.6 – Diagram of the architecture of the SAM (*left*) and photos of the power distribution system (*right)*, composed of two power bus-bars (*top*), a power distribution unit (*middle*), and a battery pack (*bottom*).

which is connected to the access point. Through the access point, the point-to-point communication channel between the SAM and the main aerial carrier (i.e., the helicopter) is established. Additionally, there is a communication channel between the aerial carrier and a ground station via a radio link.

Power distribution system

To power the whole system, 48 V with 100 A input is required for the worst case in terms of power consumption. As a battery pack, SLS[10] 12S 21000 mAh is used. Thus, this battery pack can maintain the SAM for time of approximately

$$\frac{V \cdot Capacity}{P_{consumption}} = \frac{12 \cdot 3.7 \cdot 21 \cdot 60}{48 \cdot 100} = 11.65\,[\text{min}]. \tag{11.1}$$

In the view of increasing the operational time, additional power could be provided by the main aerial carrier.

[10] `https://www.stefansliposhop.de/`

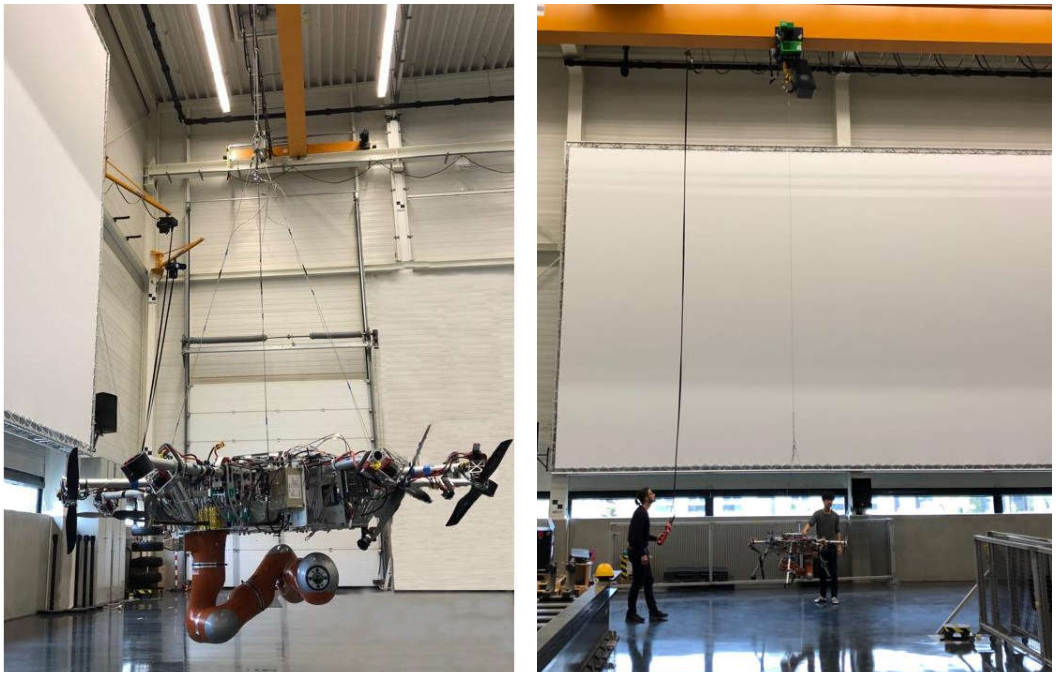


Figure 11.7 – The SAM prototype hanging on a suspended rail at DLR.

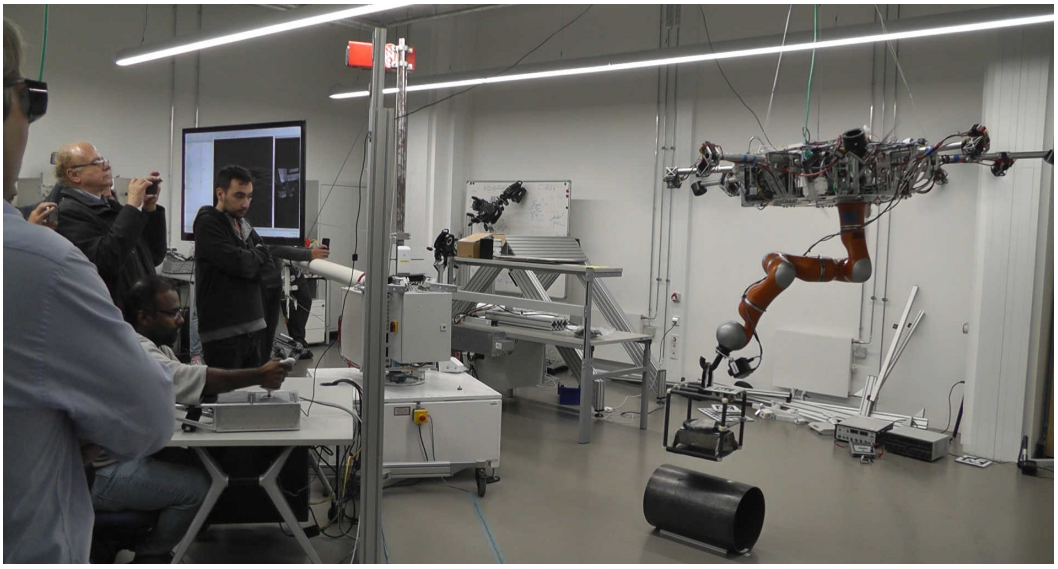


Figure 11.8 – The SAM prototype performing a tele-manipulated pick and place operation in the DLR facility.

The SAM represents one of the high-TRL aerial manipulator platforms delivered by the Aeroarms project. Actually still under development and integration, see Fig. 11.7 and Fig. 11.9, it could reach the market in the next years for the fulfillment of many manipulation task, as shown if Fig. 11.8, which depicts a preliminary tele-manipulation test for inspection and maintenance with contact of pipelines.

Figure 11.9 – Exclusive photos of the SAM prototype undergoing outdoor tests.

CATEC AeroX

The second industrial platform, called AeroX, has been developed by the project partners in CATEC[11], thanks to a tight collaboration with the University of Seville. The development of this aerial manipulator tries to reply to the strong demand in the oil and gas industry to realize alternatives to manual inspection. Indeed, such procedures involve extensive inspections, many of which have to to be performed at high altitude and using sensors that require to be in contact with the surface to be inspected [Trujillo–2019]. Contact inspection is traditionally performed by human operators accessing the specific point of interest using man-lifts, cranes or scaffolds, or rope-access techniques. Apart from involving very high costs, such operations can be substantially dangerous, considering that the pipes or other surrounding objects can have high-temperature surfaces (up to 400 °C) or contain potentially unhealthy material or fluids, thus increasing the risk of falls or other incidents. Alternative solutions, as the deployment of sensors at selected locations in the plant or the employment of crawler robots[12] with magnetic wheels to move on the outer or inner surfaces of pipes, have been already explored. Nevertheless, the first technique results suitable only for intensive monitoring of few or small critical components but is not ideal for covering a full plant, while the second one is valid only for ferromagnetic surfaces and often have low accessibility for locations at high. To overcome these issues, the AeroArms project has proposed solutions involving the use of aerial robotic manipulators.

The AeroX represents probably the highest-TRL-prototype intended for industrial exploitation that has been developed within the AeroArms project. This novel aerial robotic manipulator is composed by three main parts, which makes his design unique:

i) a MDT *floating base* exploiting eight tilted actuators. This design configurations, influenced by LAAS–CNRS, and in particular from some results shown in this thesis, allows movements in any directions of the 3D space, high maneuverability and agility, and robustness towards wind perturbations and also to rotor failure;

ii) a six DoFs *robotic arm*, which transmits the surface contact forces and perturbations to the robot CoM thus minimizing the torque disturbance on the latter. Furthermore, a particular arrangement of the actuators allows the arm to completely rotate around the platform frame without colliding with the propellers, in a similar way to what is partially implemented in the OTHex design of the frontal aperture (cf. Sec. 6.1); and

iii) a robotic *end-effector* endowed with wheels for moving on the surface under inspection. It integrates the sensors to be used for inspection and also additional ones to facilitate the operations like, e.g., a camera to help the inspector identify the surface defeats.

[11] Center for Advanced Aerospace Technologies, `http://www.catec.aero/en`

[12] `https://inspection-robotics.com/`

Figure 11.10 – The AeroX platform performing ultrasonic thickness measurements in a refinery located in Germany.

The overall design of this aerial manipulator has been patented by CATEC[13] and has recently been awarded by the EU Innovation Radar Price 2017[14]. The aerial robot underwent extensive validation experiments, with more than 200 flights, both in an outdoor testing scenario and also in a real refinery. Photos of the AeroX performing ultrasonic thickness measurements in a refinery located in Germany are depicted in Fig. 11.10, while a video is available online[15].

[13] Aeronave con Dispositivo de Contacto. Spain Patent ES 2 614 994 B1, 2 March 2018.
[14] https://www.innoradar.eu/innoradarprize
[15] `https://youtu.be/Hy45WQ3GLcI`

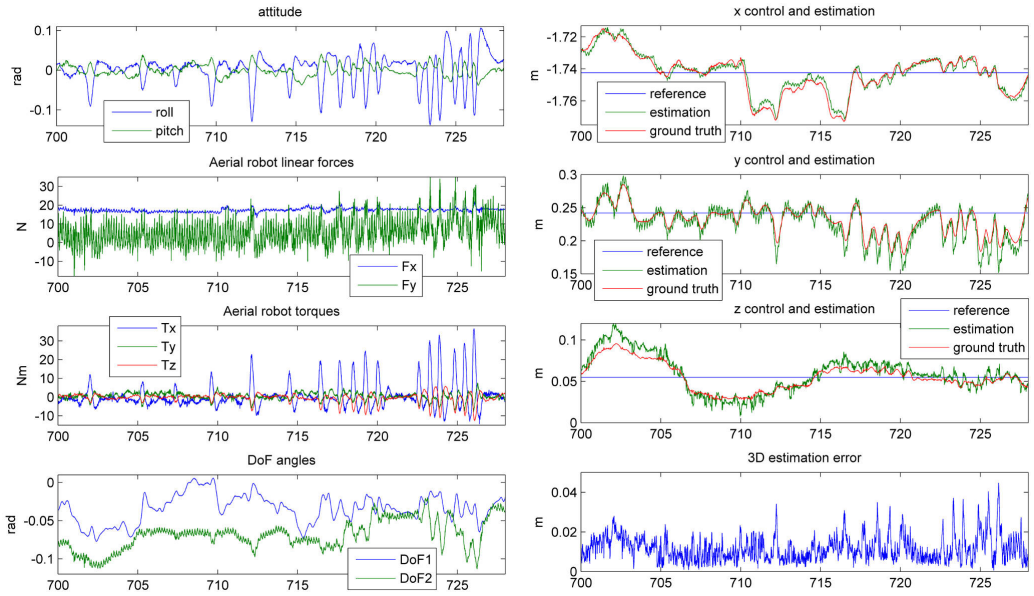


Figure 11.11 – Plots of the AeroX performing a fully-autonomous flight during contact during indoor robustness analysis experiments. The platform is perturbed with external forces of intensity up to 100 N. The injected perturbations generates the response of the controller, which can be noticed in the attitude and in the torque plots (*left column*). The robot relative position w.r.t. the surface contact point (*right column*) is shown together with the ground truth and the reference value for each axis. The position error w.r.t. the reference value is always smaller than 6 cm.

Before performing any outdoor experiment, it has been necessary to properly assess the aerial manipulator resilience to the typical disturbance effects that could arise during the contact and the manipulation phase. In order to do this, preliminary indoor tests have been performed at the CATEC facility. During these evaluations, the robot has been tasked to approach and contact a vertical surface with its end-effector, localizing itself w.r.t. the point of interest without relying on any Motion Capture System (MoCap) or GPS measurements. At the same time, the platforms has been perturbed with lateral pulling forces with an intensity up to 200 N. The telemetry of AeroX during one of these experiments is shown in Fig. 11.11. The perturbations induced sudden changes in the attitude of AeroX in the **x** axis, as can be clearly noticed top-left plot. To compensate for such disturbances, the controller generated responses with high torques in the **x** direction and lateral forces in in the **y** one. On the other hand, the plots on the left column of Fig. 11.11 show the relative position components of the robot w.r.t. the surface contact point computed by the developed method and also the ones obtained by the ground truth distance. As can be appreciated, the robot estimation error in the position has been smaller than 1 cm in all axes, while the 3D position estimation errors were lower than 4 cm. A video of this experiment is available online[16].

[16] `https://youtu.be/mp4UAuhNHWc`

Figure 11.12 – The AeroX platform performing the contact inspection of the underside of a bridge.

A more detailed description of this and other experiments, as well as a thorough description of the Aerox design, control, localization and navigation techniques can be found in [Trujillo–2019]

To conclude, it is important to point out that this and the other aerial manipulators delivered to the market by the AeroArms project can be employed in many other applications like, e.g., inspection and maintenance of general infrastructures, as shown in Fig. 11.12, which illustrate a bridge-inspection task.

Chapter 12

Conclusion

"In theory there is no difference between theory and practice, while in practice there is."
Benjamin Brewster
(even if later attributed to Richard Feynman, Albert Einstein and Yogi Berra)

This final chapter serves as conclusion to the work presented in this thesis. In particular, it pursues a triple goal. First of all, it aims to offer a concise summary of the content and the contributions delivered by this manuscript. Secondly, it proposes an analysis of the lessons learned from the challenges that we faced and tried to solve. Last but not least, in the view that a conclusion of one research work is always the beginning of another one, it outlines hints for intended and possible future works.

Summary

The overall goal of this thesis has been to advance in the design and control for aerial robots with actuation constraints that should interact with the environment. As a result, a relevant step forward has been achieved for the theoretical study and the practical implementation of Multi-Directional Thrust (MDT) Multi-Rotor Aerial Vehicles (MRAVs). For these general systems, which extend the dynamic capabilities and the application fields of standard Uni-Directional Thrust (UDT) vehicles, we proposed two alternative modeling descriptions with different levels of abstraction for the actuation, and two related and mutually complementary control strategies. Furthermore, building on top the theoretical results available from the state of the art and on the additional ones achieved in this work, we designed and built three in-house real MDT hexarotor prototypes and contributed to the development of an Omni-Directional (OD) octorotor platform, which has been finalized at DLR. This finally led to the fulfillment of aerial physical interaction tasks, which are motivated by truly industrial use-cases.

As far as the modeling is concerned, we extended the Newton-Euler representation of the dynamics of a rigid body developed in the literature to take into account for actuation constraints with two levels of detail. The first model, presented in Sec. 4.1, is suitable to describe MDT aerial vehicles that can exert a limited amount of body frame lateral forces, thus allowing to partially decouple, at the control level, the translational dynamics from the rotational one. To characterize this kind of platforms, we defined the broad class of Laterally-Bounded input

Force (LBF) aerial vehicles. Considering the body wrench as control input, this model results very general and convenient to take into account the system actuation limits in a simple but still effective way. Nevertheless, it should be kept in mind that such model epitomizes a high-level abstraction of the real system. In particular, it considers an unbounded set of forces along the body frame vertical direction, it supposes that the platform can exert a given lateral force disregarding the torque simultaneously generated and, finally, it assumes that the MRAV actuators can produce an instantaneous change in the thrust forces. Despite being more than acceptable to explain the essential rigid body dynamics, the aforementioned assumptions included in the LBF model might result too simplistic in the case a precise description of the real system and its actuator constraints are pursued. For this reason, believing that a proper characterization of the system dynamic capabilities could be achieved only through a rigorous analysis of the robot actuation means, we formally investigated a way to enhance the previous model. This was achieved by considering as control input the force derivatives produced by each actuator. This choice embodied a trade-off between model fidelity and effectiveness/feasibility of the controller implementation. Contrarily from what has been normally done in the literature, we considered the force derivative constraints as state dependent, rather than assuming them merely constant. Furthermore, we proposed a simple procedure to experimentally identify such limits. It should be appreciated that this second model, introduced in Sec. 4.2, has a much lower level of abstraction w.r.t. the first one. On the other hand, it requires a deeper analysis of the system, it contains much more parameters, and it is more robot-specific. In both cases, we could reasonably neglect the aerodynamic effects arising at very dynamic maneuvers of the vehicle and in conditions of intense air disturbances. This conscious choice was motivated by the fact that, during aerial physical interaction tasks, the manipulation of the load and the exchange of forces typically occurs at limited speed in order to prevent the rise of disturbance effects that could lead to system instability, and also by the fact that most of the experimental validations presented in this thesis have been performed in an indoor environments, where the presence of wind can be safely ignored. Nevertheless, we are completely aware of the importance of such effects in outdoor conditions. In the view of operating aerial robots in such conditions, modest additional effort can be made to take into account for such effects, already known in the state of the art.

Aiming to control the motion of MDT MRAVs, at first during contact-less operations, we developed two control algorithms which exploit the physical descriptions previously defined. The first one, outlined in Sec. 5.1, takes advantage from the first model to apply a static feedback control action to stably steer the aerial vehicle towards a 6D reference trajectory. By simplifying the ensemble of body frame forces with a conservative set, the algorithm performs a prioritized control of the translational dynamics over the rotational ones. If the reference motion profile is feasible w.r.t. the chosen abstraction of the actuation capabilities, the full-pose trajectory tracking is guaranteed. On the other hand, only the position tracking and the chase of the closest feasible orientation, chosen as the solution of an on-

line optimization problem, is ensured. The good points of this controller are its theoretically proven stability and its simple form. Conversely, its main drawback is the fact of being conservative w.r.t. the real dynamic capability of the MRAV. In order to fill this gap, motivated by the theoretical and practical relevance of the problem of obtaining improved tracking while complying with the actuator limitations, we implemented another control algorithm to capitalize on the more detailed physical description given by the second model. This gave birth to the nonlinear model predictive controller presented in Sec. 5.2. In this case, the command for the actuator force derivatives (equivalently, the rotor accelerations) is found as the solution of an optimization problem along a future time windows, along which the system evolution is predicted using the model. This implies a faster response to tracking errors and, above all, the compliance with the low-level constraints of the system previously identified. As a consequence, an improved an less conservative trajectory tracking is produced. Nevertheless, a formal study of system stability in the case the reference trajectory is unfeasible is very difficult to be carried out. This is why the two controllers are mutually complementary, in a sense.

Starting from the formal study of the allocation matrix in a particular case, we designed three real MDT hexarotor prototypes, each one aimed to fill a well-defined gap in the state of the art. Hints on the key points have been pointed out in Sec. 6.1. The first platform, called Tilt-Hex, aims to achieve the partial decoupling of the translational dynamics from the rotational one without the addition of any active mechanism and, furthermore, the resilience towards the loss of one propeller. In order to do this, the actuators have been arranged in a peculiar way. Despite not being entirely new (cf. [Voyles–2012; Rajappa–2015]), this design has allowed the development of more mature control strategies for the full-pose trajectory tracking and the handling of a rotor failure. Furthermore, the rigorous analysis of the rank deficiencies in the allocation matrix granted the identification of undesirable non-trivial configurations. Finally, the Tilt-Hex design eased the creation of more manipulation-oriented platforms. On the other hand, the design of the second prototype, called FAST-Hex, was completely new w.r.t. the existing literature, at the best of our knowledge. Thanks to only one additional actuator, this aerial robot can change its configuration from the efficient but under-actuated UDT to the more dynamically effective but less efficient MDT one. This additional Degree of Freedom (DoF) can be tuned w.r.t. the task to be fulfilled. Finally, driven by the high demand of industrial sectors for truly aerial manipulators capable to perform both construction/decommissioning tasks and contact-based operations, we conceived the OT-Hex. The aperture within its frame eases interaction tasks involving frontal manipulation. This unique aerial robot allowed the accomplishment of ambitious and challenging operations. Despite the quite low Technology Readiness Level (TRL) of all the three aerial robots, which is however expected from laboratory products, their concepts had a meaningful impact on the two MDT octorotor manipulator platforms developed within the AeroArms project, as outlined in Sec. 11.2 and in Sec. 11.3. While the CATEC AeroX has a an actuator arrangement which almost mirrors the ones of the Tilt-Hex and of the OT-Hex, the

one of the DLR SAM is the result of an optimization method, obtained with small modifications w.r.t. the one described in [Tognon–2018b], that we implemented at LAAS–CNRS to obtain the OD capability with a minimum number on unidirectional actuators. The initial and exploratory steps achieved in this direction suggested that considerable work could be built on top of those findings.

In order to verify the motion control strategies proposed in Chapt. 5, we exploited the in-house developed MDT MRAVs to conduct an intense experimental validation campaign, encompassing challenging 6D pose trajectories and also discontinuous motions. Furthermore, an additional and realistic numerical validation was carried out to assess the capability of the predictive controller to deal with a convertible MRAV and also with the Tilt-Hex subject to the failure of one actuator. The results of the conducted experimental tests and of the additional numerical simulations, presented in Chapt. 7, proved the validity and the effectiveness of both control methods.

With the goal of addressing not only the fulfillment of 6D motion control with MDT MRAVs, but also the accomplishment of interaction-controlled operations, in Chapt. 8 we laid the basis for aerial physical interaction control exploiting already existing techniques mainly developed in the literature of ground manipulators. Thanks to a model-based wrench observer, we were able to estimate the external forces and torques acting on the platform. Furthermore, thanks to an admittance control paradigm, built on top of the motion control, we could model the complete scheme as a mass-spring-damper system, whose stiffness, damping and inertia coefficients can be tuned at will in order to confer the proper compliance on the aerial robot, depending on the task to be carried out. From the theoretical study of the interaction control, we passed to more applicative and practical problems, motivated by truly industrial use-cases. This led to the accomplishment of the challenging task of lifting long loads like, e.g., beams with the OT-Hex, outlined in Chapt. 9. For the first time, at the best of our knowledge, a MDT MRAV has been designed and concretely exploited to act as a flying assistant that can aid other heterogeneous robot and, possibly, human operators, to move long bars for assembly and maintenance tasks. Thus, we considered also a multi-robot extension of the previous demonstration. In that scope, we fulfilled a cooperative manipulation between an aerial and a ground robot. This gave birth to the MAGMaS system and also to its evolution with human-in-the-loop tele-presence. Furthermore, driven by the aim of the AeroArms project to develop aerial manipulator capable of performing contact inspection of industrial sites, we endowed the OT-Hex with a rigid tool, first, and with a two DoFs lightweight arm. In the first case, thanks to the MDT capability of the platform, we could easily achieve the sliding on sloped surfaces and on a mock-up pipe. In the second instance, we could obtain a truly redundant aerial manipulator able to carry out a complete inspection of a metallic pipe by means of an Eddy Current (EC) sensor mounted of the arm End-Effector (EE). The results of this meaningful experiments are demonstrated in Chapt. 10. To the best of our knowledge, this has been the first time such a task has been autonomously accomplished with a redundant MDT MRAV.

Ultimately, the significant impact of the thesis on the industrial sector has been shown in Chapt. 11, where the features of the DLR SAM and the CATEC AeroX have been briefly presented, together with their application fields. In the next years, we might see more and more aerial robots of this kind intervene in real industrial operations involving physical interaction.

Lesson learned

Throughout all the thesis, and for each problem we focused on, we tried with strong effort to first conduct a fundamental theoretical study and then, once reached an adequate level of maturity, to exploit the obtained results in order to target a meaningful practical application. We[1] are convinced that a formal study is the fundamental foundation of every scientific work. When performed in a wise way, it allows to deeply understand the underneath properties of the system under analysis and, possibly, to predict and/or control its evolution. On the other hand, we also believe that such theoretical formalization has to be, at least partially, influenced by real experience. Furthermore, it should be aimed at the achievement of a useful (w.r.t. some criteria) results, in order to avoid merely becoming a end itself. The mutual relationship between theory and practice might be possibly explained by saying that theory is abstract practice, and practice is applied theory.

With this is mind, we conceived the design of the MDT hexarotors presented in this thesis after the formal analysis of the robot allocation matrix in a well-representative case. In particular, this highlighted the presence of non-trivial configurations causing under-actuation due to allocation singularities that were not easily predictable. This theoretical result turned out to be particularly useful in the conception of the convertible FAST-Hex prototype, given the fact that the synchronized re-orientation of its actuators allow the crossing of such singularities. On the other hand, practical considerations regarding the accomplishment of aerial physical interaction tasks led to strategic modifications of the initial Tilt-Hex design. In particular, to conceive the OT-Hex we created an aperture within the platform chassis in order to guarantee the lift of long loads of a given size while safely avoiding the collision with the propellers. Furthermore, this allowed to endow the platform with a lightweight arm in order to effectively perform frontal manipulation tasks. Regarding the design of the actuator arrangement for the DLR SAM, we took advantage of the preliminary theoretical results achieved by the state of the art to obtain an OD design, combining the theoretical analysis with the specific needs imposed by the practical application.

In a very similar way, the interplay between theory and practice can be appreciated in the design of the presented motion control strategies. The static feedback control law originated from a theoretical abstraction of the MDT MRAV concept,

[1]The use of first plural person throughout this thesis is motivated by the fact that most of the presented results have been achieved together with my colleagues. In this context, the proposed considerations are probably more personal.

epitomized by the LBF class, and was the result of a formal analysis aimed at guaranteeing the system stability. Nevertheless, the empirical experience achieved thanks to an extensive validation campaign taught us that improved tracking performance and the compliance with particular dynamic constraints could be achieved provided that a more thorough description of the actuators was accomplished. This motivated the development of the predictive numerical controller, which exploits a much more precise model of the system and, in particular, of its actuators. In this case, the features of the model have been enhanced with a methodical procedure for the identification of the maximum force derivatives that can be generated. It should be remarked that this operation still represents an abstraction of the "true" control inputs, which could be reasonably assumed to be the voltage applied to the motors. This strategy was motivated by the practical need of implementing a controller for this model and was fulfilled by modeling the input constraints as state-dependent, thus indirectly taking into account for the low-level physical effects in a simple but effective way.

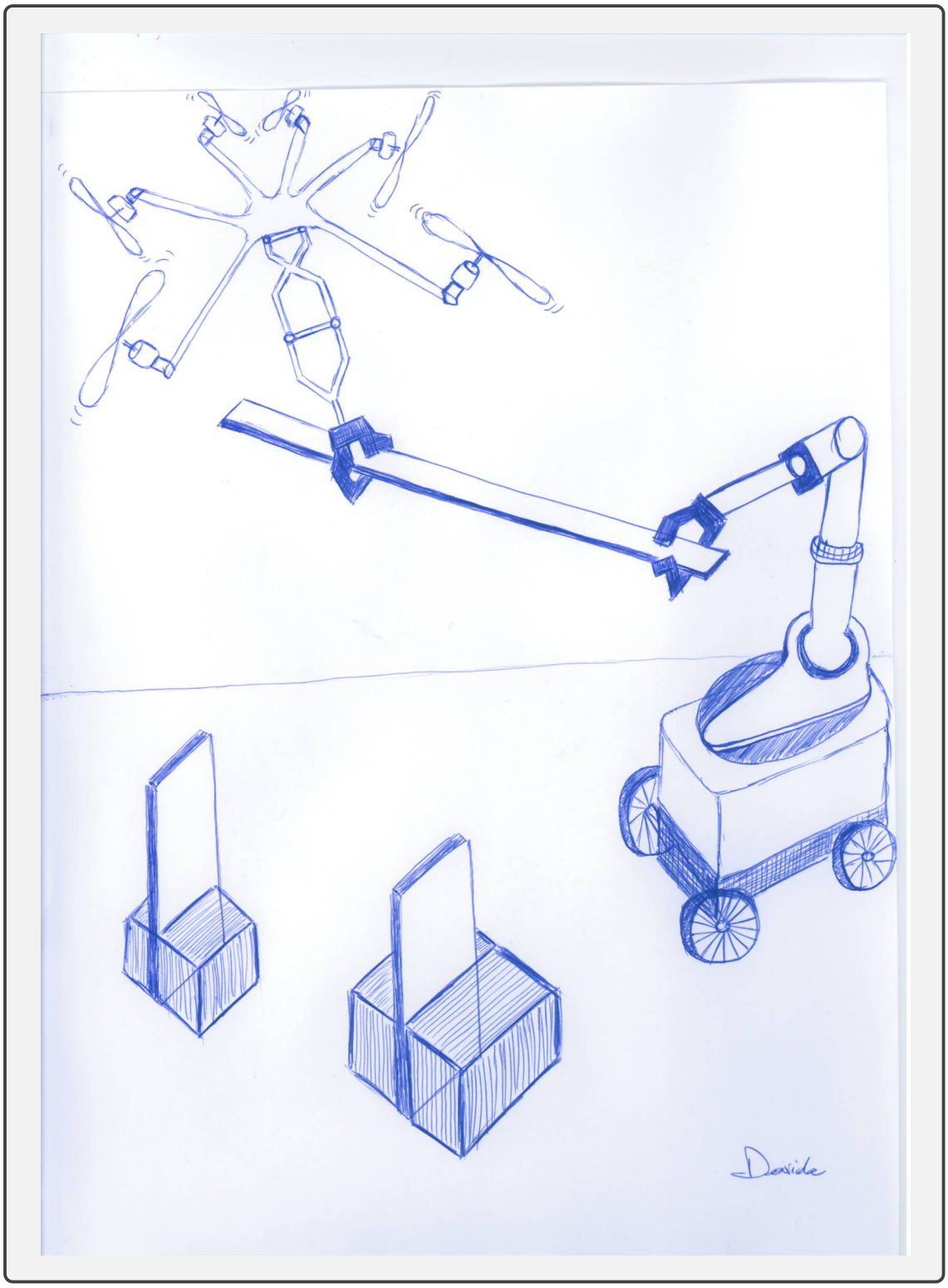
Ultimately, the numerous experimental tests described alongside the thesis did teach us that to further improve the theoretical control schemes it is sometimes necessary to implement different practical adjustments. Examples of such technical modifications operated in the illustrated works have been: *i)* the calibration of the wrench estimator in order to avoid taking into account for manufacturing errors in the estimation of the interaction forces and torques; *ii)* the implementation of a dead zone on the admittance filter input to suppress non-existent small force and torque estimations due to sensor noise; *iii)* the low-pass filtering of the contact wrench to reduce high-bandwidth noise from the sensor measurements; and *iv)* the implementation of a dislocated PD controller with mixed-feedback strategy to deal with the flexibility introduced by the 2-DoF lightweight arm.

Future works

As far as Part II is concerned, we presented a complete study on MDT MRAVs with actuation constraints, ranging from the design to the control of such robots and tackling the problem from both the theoretical and the practical perspective. Nevertheless, some additional work can be conceived. First of all, in the view of validating the presented techniques in an outdoor scenario, that was not addressed in this thesis, the inclusion of aerodynamic effects such as the rotor drag force, the blade flapping and possibly the interaction effects arising in the proximity of large surfaces, should be considered both at the modeling and at the control level. Indeed, they could have a much more significant impact on the vehicle dynamics. In our view, this could be achieved with moderate additional effort. Secondly, we believe that interesting works could be done in relation to the introduced Nonlinear Model Predictive Control (NMPC) strategy. In particular, the framework could be enhanced in order to include perception objectives inside the cost function, as recently done by some works in the literature. This could somehow unify control

and planning w.r.t. action and perception objectives. Another attractive extension could be done towards the accomplishment of physical interaction tasks. The framework could be upgraded with a wrench estimator and an admittance filter outer loop, similarly to what we have done with the other motion control approach. Furthermore, a fruitful enhancement in the scheme functionalities could be carried out in order to automatically tune the gains inside the cost function, since an extensive and heuristic manual procedure is traditionally required. Moreover, few additional work could be developed to obtain optimal MDT MRAV designs to accomplish other different tasks. In particular, not only the orientation but also the position of the actuators could be considered, in the view of minimizing the mutual aerodynamic disturbance between the propellers. Additionally, a proper exploitation of the redundancy of the SAM manipulator is advisable. Finally, the aerial vehicles could be equipped with an onboard vision system for the localization and the navigation through unknown environments.

Regarding the work detailed in Part III, we believe that there are many possible extensions and open questions in the field of aerial physical interaction, with significant margin to improve the manipulation capabilities of aerial robots. As far as the work done with the OT-Hex is concerned, a fascinating research line could be followed towards the collaboration of the aerial robot with human operators, in particular for the fulfillment of cooperative assembly and decommissioning tasks. In the scope of (tele-) MAGMaS systems, extensions of the presented work could be achieved by exploiting a moving platform for the ground robot. Ultimately, in relation to the detection and mapping of a weld on a metallic pipe, performed by the OT-Hex with a two-DoF arm, we think that profitable contributions to the state of the art could be provided by the definition of controllers to suitably deal with the challenges introduced by the arm flexibility. Regarding the application, the next step should be the detection of cracks along the weld. This could require a combination of force interaction and vision techniques, which we believe to be a very interesting and motivating research direction to explore. The natural upgrade for the system, also in this case, should be the integration of an onboard localization system to perform the task outdoor in a real industrial plant, as realized with the AeroX manipulator.



Drawing IV

Appendix

Appendix A

Identification of the MRAV allocation matrix

Apart from the identification of the model parameters, to conduct the experiment presented in Chapt. 7, we also experimentally identify the entries of the allocation matrix $\mathbf{G}$ defined in (3.13). In order to control the aerial robot, a possible legitimate choice could also be to use a nominal matrix, calculated geometrically w.r.t. the model, as done in (3.14), by plugging in it the numerical values of the parameters that can be calculated by some Computer Aided Design (CAD) programs (in case the drawing of the platform is available). Indeed, we have been exploiting such solution for the experimental validation of other works in the past. However, such method disregards the real physical parameters of the robot, which could be – and actually are – quite different from the ideal ones. This happens mainly due to mechanical inaccuracies unavoidably associated with the manufacturing and the assembly of the robot parts. The effects of these un-modeled effects are particularly important in the case a predictive controller is used to command the Multi-Rotor Aerial Vehicle (MRAV), given the fact that it would exploit a wrong model to simulate the future evolution of the system dynamics, thus jeopardizing the tracking performance. Therefore, in order to achieve an improved trajectory tracking in the real experiments, we decided to identify a more realistic and representative allocation matrix. In the following, we only briefly outline the *least-squares* approach we used, since it has been extensively used in the literature and it is very well-known from the community. For a review of other more complex identification methods, outside the scope of this thesis, the interested reader is addressed to [Ljung–1987].

First of all, by using the nominal allocation matrix in the controller model, we flew the platform by generating particular 6D trajectories that were *exciting enough* for the parameter identification. To this purpose, we generate chirp trajectories for all the components of the desired position $\mathbf{p}_d$ and of a minimum representation for the orientation $\boldsymbol{\eta}_d$. This allowed us to excite the robot with sinusoidal trajectories with increasing frequencies. While doing this, we collected the measured data $\mathbf{p}$ and $\mathbf{R}$ thanks to the Motion Capture System (MoCap) system, used as ground truth. A posteriori, thanks to a properly tuned post-processing of the data which mainly consisted in a constant frame-rate signal re-sampling, an anti-causal low-pass filtering and the computation of numerical derivatives, we were able to retrieve a precise-enough estimation of $\ddot{\mathbf{p}}, \boldsymbol{\omega}, \dot{\boldsymbol{\omega}}$ defined in (3.7). On the other hand, $\boldsymbol{\gamma}$ was reconstructed by collecting the measured spinning rates of the motors w_i and using 3.17. Finally, m was directly measured and $\mathbf{J}$ estimated by a precise CAD

model of the robot. At this point, we re-wrote 3.7 in the following way:

$$\underbrace{\begin{bmatrix} m\mathbf{R}^\top(\ddot{\mathbf{p}} + g\mathbf{e}_3) \\ \mathbf{J}\dot{\boldsymbol{\omega}} + \boldsymbol{\omega} \times \mathbf{J}\boldsymbol{\omega} \end{bmatrix}}_{:=\mathbf{y}} = \underbrace{\begin{bmatrix} f_1\mathbf{I}_3 & \dots & f_n\mathbf{I}_3 & \mathbf{0}_{3\times 3n} \\ \mathbf{0}_{3\times 3n} & & & f_1\mathbf{I}_3 & \dots & f_n\mathbf{I}_3 \end{bmatrix}}_{:=\mathbf{A}} \boldsymbol{\beta}, \tag{A.1}$$

with $\mathbf{A} \in \mathbb{R}^{6\times 6n}$. In such form, the equation allows to express the vector of measurable quantities $\mathbf{y} \in \mathbb{R}^{6\times 1}$ as a linear function of a vector of parameters $\boldsymbol{\beta} \in \mathbb{R}^{6n\times 1}$, obtained re-arranging the entries of $\mathbf{G}$

$$\boldsymbol{\beta} := \left[\mathbf{G}_1(:,1)^\top \dots \mathbf{G}_1(:,n)^\top \mathbf{G}_2(:,1)^\top \dots \mathbf{G}_2(:,n)^\top\right]^\top \tag{A.2}$$

Collecting a good number of measurements $p >> 6n$ and stacking them in vectorial form, we obtained

$$(\boldsymbol{\xi} = \boldsymbol{\Lambda}\boldsymbol{\beta}) := \left(\begin{bmatrix} \mathbf{y}_1 \\ \vdots \\ \mathbf{y}_p \end{bmatrix} = \begin{bmatrix} A_1 \\ \vdots \\ A_p \end{bmatrix} \boldsymbol{\beta}\right) \tag{A.3}$$

At this point, applying the standard linear least-squares identification method, we know that the vector which minimizes the 2-norm of the error $||\boldsymbol{\Lambda}\boldsymbol{\beta} - \boldsymbol{\xi}||^2$ is

$$\hat{\boldsymbol{\beta}} = \boldsymbol{\Lambda}^\dagger \boldsymbol{\xi} \tag{A.4}$$

Finally, re-arranging the elements of the vector $\hat{\boldsymbol{\beta}}$ using the convention of (A.2) we obtained the entries of the identified allocation matrix $\hat{\mathbf{G}}$ that we used in the experimental validation presented in Chapt. 7.

The Tilt-Hex case

By comparing the entries of the nominal and the identified allocation matrices in the Tilt-Hex case, it is interesting to notice that the difference between some elements is pretty consistent. This confirms that the physical parameters of the real robot can be very dissimilar from the nominal ones.

$$e_{G,\%} = 100 \left[\frac{g_{i,j} - \hat{g}_{i,j}}{g_{i,j}}\right] = \begin{bmatrix} 42 & -12 & -18 & 41 & 9 & 16 \\ 4 & 21 & 25 & 4 & 80 & 104 \\ 4 & 6 & 11 & 10 & 5 & 2 \\ 72 & 31 & 30 & 58 & 25 & 28 \\ 26 & 24 & 31 & 27 & 29 & 28 \\ 9 & 15 & 16 & 12 & 14 & 13 \end{bmatrix} \tag{A.5}$$

To conclude, we would like to point out that using the identified matrix in the Nonlinear Model Predictive Control (NMPC) controller (see Sec. 5.2 and Sec. 7.2) instead of the nominal one allowed to consistently reduce both the position and the orientation errors in all the experiments that we performed. This happens already in hovering condition, as it is shown in the plot of Fig. A.1.

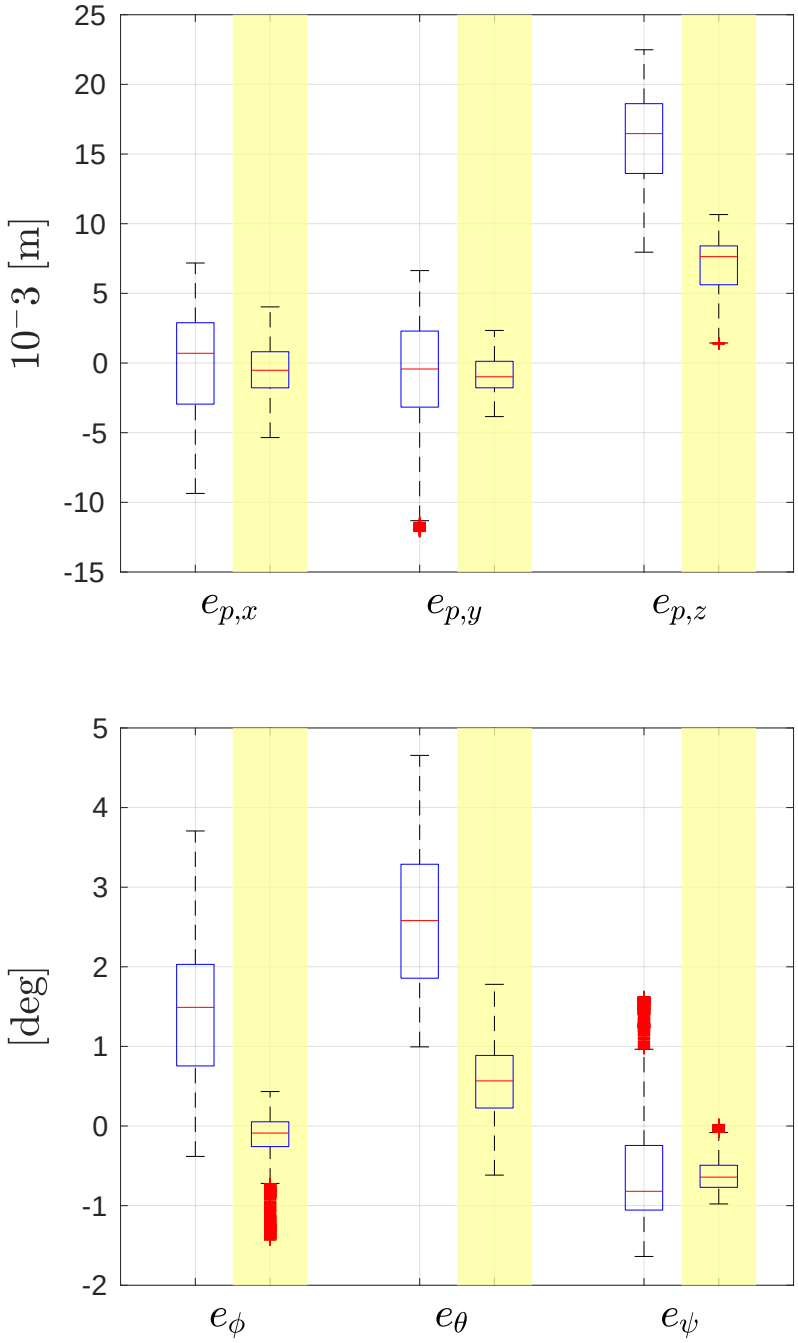


Figure A.1 – Box-plots for the position error (*top*) and the orientation error (*bottom*) of the Tilt-Hex when hovering using the nominal and the identified allocation matrices. The results for the latter case have been highlighted with yellow bands.

Appendix B

Shaping of the admittance filter physical properties

In this context, we experimentally test and demonstrate the physical property shaping capabilities of the outer loop admittance filter (see Chapt. 8) with respect to the end-effector tool-tip. By exerting a step-like force profile on the aerial robot we show that we can achieve a large variety of desired mass-spring-damper behaviors.

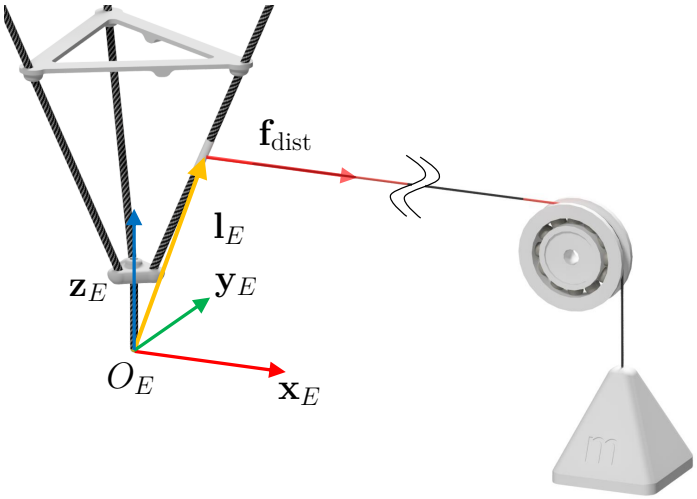


Figure B.1 – CAD representation of the property shaping experiment performed with the Tilt-Hex. Left side shows the end-effector of the aerial robot. A mass pulls on a rope fixated on a point close to the end-effector, resulting in a force $\mathbf{f}_{\text{dist}}$ in the direction of $\mathbf{x}_W = \mathbf{x}_E$. The force is applied with an offset $\mathbf{l}_E$ with respect to the tool-tip, thus resulting also in a moment w.r.t. O_E.

During all these experiments, the desired end-effector position, $\mathbf{p}_{E,d} = [0\ 0\ 1]^\top$ m, and orientation, $\mathbf{R}_{E,d} = \mathbf{I}_3$, in (8.11)-(8.12) are kept constant over time. During steady state hovering of the aerial robot, a step disturbance force is applied. The disturbance force is unaligned with the tool-tip by the vector $\mathbf{l}_E$ (see Fig. B.1), resulting not only in a force but as well in a torque with respect to the tool-tip. The disturbance step is realized by a released mass ($m = 0.14\,\text{kg}$) pulling on a taut cable along $\mathbf{x}_E$, fixated on a point close to the end-effector of the aerial robot, thus resulting in a pulling disturbance force of about $\mathbf{f}_{\text{dist}} = [1.4\ \ 0\ \ 0]^\top$ N. Once the aerial robot reaches a new steady state hovering, the force is removed by lifting the weight and keeping the cable slack. To demonstrate the full capabilities of the

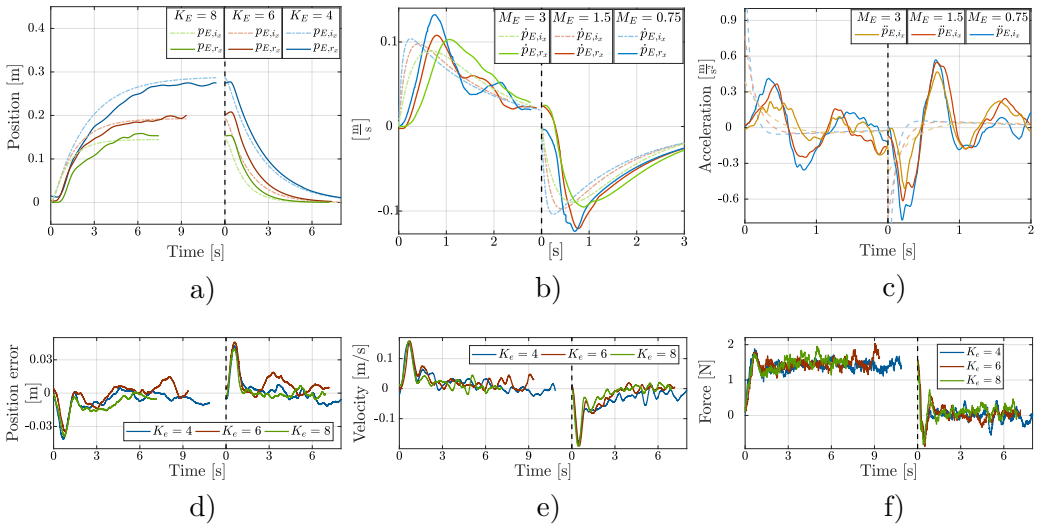


Figure B.2 – Physical property shaping: During this experiment, the aerial robot is charged with a constant force disturbance, at the dashed line the force is removed. For more clarity the time is restarted in the moment of force removal in all plots. The force is applied along $\mathbf{x}_E$, therefore only the first component of the position or velocity vector is presented in all plots (solid lines). The dashed lines show an ideal mass spring damper system. a) The spring constant $\mathbf{K}_E$ in (8.11) is varied, resulting in different steady state positions. b) The damping constant $\mathbf{D}_E$ is varied. The steady state position is constant in all three cases but velocity differs significantly. c) The mass constant $\mathbf{M}_E$ is varied, resulting in different rising times. For better visualization the x-axis scaling is different than before. Last three plots: Further data of for this experiment. d) Low level position error along p_x. e) Velocity component along $\dot{p}_x$. f) Force estimate $\hat{f}_{E_1-\text{ext}}$, being the first component of $\hat{\mathbf{w}}_E$ in (8.9).

admittance filter, the three parameters $\mathbf{M}_E$, $\mathbf{D}_E$ and $\mathbf{K}_E$ in (8.11) have been individually modified. As reference values we chose $\mathbf{M}_E = 1.5\mathbf{I}_3\,\text{kg}$, $\mathbf{D}_E = 10\mathbf{I}_3\,\text{kg}\,\text{s}^{-1}$ and $\mathbf{K}_E = 6\mathbf{I}_3\,\text{kg}\,\text{s}^{-2}$. In three experimental batches we increased and decreased each of the three parameters while keeping the other two unchanged.

First of all, we changed the spring constant to $\mathbf{K}_E = 4\mathbf{I}_3\,\text{kg}\,\text{s}^{-2}$ and $\mathbf{K}_E = 8\mathbf{I}_3\,\text{kg}\,\text{s}^{-2}$, see Fig. B.2-a). To allow an evaluation of the results we simulated additionally an ideal mass-spring-damper system (msd-system) under the same force disturbance and compared the position output p_{E,i_x} of the ideal msd-system with the reference position output $\mathbf{p}_E$ in (8.12) of a real experiment. The results show firstly that an ideal msd-system is emulated very well by our admittance scheme as the ideal trajectory is tracked precisely. The maximum position difference between the reference position and the msd-system (p_{E,i_x}-p_{E_x}) is below 2 cm for $\mathbf{K}_E = 6\mathbf{I}_3\,\text{kg}\,\text{s}^{-2}$. Secondly, as desired, the modification of the spring constant results in three different behaviors and different final steady state positions (14 cm, 19 cm, 29 cm). These results match very well the expected steady state positions (15 cm, 20 cm, 30 cm) considering the utilized dead-zone (see Sec. 8.4). Every single trial is tracked well. Thirdly, it is worth to note the differences in charging and discharging of the spring. The reproduction of an ideal msd-system is much better mirrored during discharg-

ing - as well the ideal trajectory is better tracked. We assume this is due to small but immanent friction and stick-slip effects that are in-existent during discharging. To give more insides of the msd-system, we plotted the first component of the low level position error $\mathbf{e}_p = [e_{p_x} \;\; e_{p_y} \;\; e_{p_z}]^\top$ (see (5.1)) in Fig. B.2-d) for the three different $\mathbf{K}_E$ values. The largest position error occurs shortly ($t < 1\,\mathrm{s}$) after the force application. This is explained by the nature of the wrench observer, which suppresses high frequency signals. Fig. B.2-e) presents the velocity profile along the first component of the velocity vector $\dot{\mathbf{p}} = [\dot{p}_x \; \dot{p}_y \; \dot{p}_z]^\top$ which has the same direction of the applied force. It is nicely visible that changing $\mathbf{K}_E$ has only minimal effect on the velocity. Finally, Fig. B.2-f) shows the first component of the wrench estimator acting on the tool-tip as in (8.9). It is clear that the estimated forces overlap very well in all cases, which is a further demonstration of the good performance of the estimator.

In the second experimental batch, the damping constant was changed to $\mathbf{D}_E = 5\mathbf{I}_3\,\mathrm{kg\,s^{-1}}$ and $\mathbf{D}_E = 20\mathbf{I}_3\,\mathrm{kg\,s^{-1}}$, while $\mathbf{K}_E$ was set back to its initial value. In contrast to the first experimental batch, the steady state value is now identical in all three cases since it depends only on $\mathbf{K}_E$. To get a better insight of the effect of a changed damping parameter, we compared the ideal msd-system velocity $\dot{\mathbf{p}}_{E,i}$ with the real reference velocity output $\dot{\mathbf{p}}_E$ in Fig. B.2-b). The actual velocity tracks the reference velocity of the msd-model although a time delay is recognizable. The delay is an expected result of the wrench estimator. Again the charging process differs from the discharging process, which appears to be much smoother.

Finally, in the last experimental batch we compared the influence of a changed mass property $\mathbf{M}_E$ comparing the nominal mass $\mathbf{M}_E = 1.5\,\mathrm{kg}$ with a reduced mass of $\mathbf{M}_E = 0.75\,\mathrm{kg}$ and an increased mass of $\mathbf{M}_E = 3\,\mathrm{kg}$. The results are not as obvious as in the previous experiments (see Fig. B.2-c)). The ideal trajectory in this case has an instantaneous acceleration response to the applied force. Comparing the results in Fig. B.2-c) it turns out that the actual acceleration response is slower and lower than expected from the ideal msd-system. This behavior is explained by the first order low-pass dynamics of the wrench observer. Nevertheless, the highest simulated mass $\mathbf{M}_E$ results, as expected, in the smallest acceleration and vice-versa.

These and other additional experimental results can be appreciated in the related multimedia content [video05–2019]. Moreover, the tuning of the msd-system during the sliding experiment presented in Sec. 10.1 can be visually enjoyed in [video06–2019]. Finally, the video related to an additional peg-in-hole task, which has not been presented in this thesis, is available in [video07–2019].

Bibliography

[–] *OpenBLAS*. `https://github.com/xianyi/OpenBLAS` (cited on page 131).

[Achtelik–2012] M. Achtelik, K.-M. Doth, D. Gurdan, and J. Stumpf. "Design of a multi rotor MAV with regard to efficiency, dynamics and redundancy". In: *AIAA Guidance, Navigation, and Control Conference*. 2012, p. 4779 (cited on pages 52 and 147).

[AeRoArms–] AeRoArms. *EU Coll. Proj. ICT-644271*. `www.aeroarms-project.eu` (cited on pages 97, 175, 187 and 207).

[Alamir–1995] M. Alamir and G. Bornard. "Stability of a truncated infinite constrained receding horizon scheme: the general discrete nonlinear case". In: *Automatica* 31.9 (1995), pp. 1353–1356 (cited on page 81).

[Alessandretti–2013] A. Alessandretti, A. P. Aguiar, and C. N. Jones. "Trajectory-tracking and path-following controllers for constrained underactuated vehicles using Model Predictive Control". In: *2013 European Control Conference (ECC)*. 2013, pp. 1371–1376 (cited on page 81).

[Alessio–2009] A. Alessio and A. Bemporad. "A survey on explicit model predictive control". In: *Nonlinear model predictive control*. Springer, 2009, pp. 345–369 (cited on page 32).

[Alexis–2014] K. Alexis, C. Papachristos, R. Siegwart, and A. Tzes. "Robust explicit model predictive flight control of unmanned rotorcrafts: Design and experimental evaluation". In: *Control Conference (ECC), 2014 European*. IEEE. 2014, pp. 498–503 (cited on pages 27, 32, 34 and 36).

[Alexis–2016a] K. Alexis, G. Darivianakis, M. Burri, and R. Siegwart. "Aerial robotic contact-based inspection: planning and control". In: *Autonomous Robots* 40.4 (2016), pp. 631–655 (cited on page 188).

[Alexis–2016b] K. Alexis, C. Papachristos, R. Siegwart, and A. Tzes. "Robust model predictive flight control of unmanned rotorcrafts". In: *Journal of Intelligent & Robotic Systems* 81.3-4 (2016), pp. 443–469 (cited on pages 27, 32, 34 and 36).

[Anderson Jr–2010] J. D. Anderson Jr. *Fundamentals of aerodynamics*. Tata McGraw-Hill Education, 2010 (cited on page 6).

[Andersson–2013] J. Andersson. "A general-purpose software framework for dynamic optimization". PhD thesis. PhD thesis, Arenberg Doctoral School, KU Leuven, Department of Electrical Engineering (ESAT/SCD) and Optimization in Engineering Center, Kasteelpark Arenberg 10, 3001-Heverlee, Belgium, 2013 (cited on page 131).

[Anglade–2019] A. Anglade, J.-M. Kai, T. Hamel, and C. Samson. "Automatic control of convertible fixed-wing drones with vectorized thrust". PhD thesis. INRIA Sophia Antipolis-I3S, 2019 (cited on page 6).

[Antonelli–2016] G. Antonelli, E. Cataldi, G. Muscio, M. Trujillo, Y. Rodriguez, F. Pierri, F. Caccavale, A. Viguria, S. Chiaverini, and A. Ollero. "Impedance Control of an aerial-manipulator: Preliminary results". In: *2016 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*. Daejeon, South Korea, 2016, pp. 3848–3853 (cited on page 9).

[Augugliaro–2013] F. Augugliaro and R. D'Andrea. "Admittance Control for Physical Human-Quadrocopter Interaction". In: *12th European Control Conference*. Zurich, Switzerland, July 2013, pp. 1805–1810 (cited on page 10).

[Augugliaro–2014] F. Augugliaro, S. Lupashin, M. Hamer, C. Male, M. Hehn, M. W. Mueller, J. S. Willmann, F. Gramazio, M. Kohler, and R. D'Andrea. "The Flight Assembled Architecture installation: Cooperative construction with flying machines". In: *IEEE Control Systems Magazine* 34.4 (2014), pp. 46–64 (cited on pages 10 and 181).

[Baca–2016] T. Baca, G. Loianno, and M. Saska. "Embedded model predictive control of unmanned micro aerial vehicles". In: *Methods and Models in Automation and Robotics (MMAR), 2016 21st International Conference on*. IEEE. 2016, pp. 992–997 (cited on pages 27, 29, 33 and 36).

[Baca–2017] T. Baca, P. Stepan, and M. Saska. "Autonomous landing on a moving car with unmanned aerial vehicle". In: *Mobile Robots (ECMR), 2017 European Conference on*. IEEE. 2017, pp. 1–6 (cited on pages 26 and 27).

[Baizid–2016] K. Baizid, G. Giglio, F. Pierri, M. Trujillo, G. Antonelli, F. Caccavale, A. Viguria, S. Chiaverini, and A. Ollero. "Behavioral control of unmanned aerial vehicle manipulator systems". In: *Autonomous Robots* 35.8 (2016), pp. 1–18 (cited on page 10).

[Bangura–2014a] M. Bangura and R. Mahony. "Real-time Model Predictive Control for Quadrotors". In: *IFAC Proceedings Volumes* 47.3 (2014). 19th IFAC World Congress, pp. 11773–11780. ISSN: 1474-6670 (cited on pages 27, 29, 33 and 36).

[Bangura–2014b] M. Bangura and R. Mahony. "Real-time Model Predictive Control for Quadrotors". In: *IFAC Proceedings Volumes* 47.3 (2014). 19th IFAC World Congress, pp. 11773–11780. ISSN: 1474-6670 (cited on pages 50 and 51).

[Bangura–2017] M. Bangura and R. Mahony. "Thrust control for multirotor aerial vehicles". In: *IEEE Transactions on Robotics* 33.2 (2017), pp. 390–405 (cited on page 51).

[Bemporad–2002] A. Bemporad, M. Morari, V. Dua, and E. N. Pistikopoulos. "The explicit linear quadratic regulator for constrained systems". In: *Automatica* 38.1 (2002), pp. 3–20 (cited on page 32).

[Bemporad–2009] A. Bemporad, C. A. Pascucci, and C. Rocchi. "Hierarchical and hybrid model predictive control of quadcopter air vehicles". In: *IFAC Proceedings Volumes* 42.17 (2009), pp. 14–19 (cited on pages 27, 28 and 64).

[Berkenkamp–2015] F. Berkenkamp and A. P. Schoellig. "Safe and robust learning control with Gaussian processes". In: *Control Conference (ECC), 2015 European*. IEEE. 2015, pp. 2496–2501 (cited on page 35).

[Bicego–2019] D. Bicego, J. Mazzetto, R. Carli, M. Farina, and A. Franchi. "Nonlinear Model Predictive Control with actuator constraints for Multi-Rotor Aerial Vehicles". In: *Submitted to Journal of Intelligent & Robotic Systems* (2019) (cited on page 16).

[Bock–1984] H. G. Bock and K.-J. Plitt. "A multiple shooting algorithm for direct solution of optimal control problems". In: *Proceedings of the IFAC World Congress*. 1984 (cited on page 131).

[Bouabdallah–2005] S. Bouabdallah and R. Siegwart. "Backstepping and sliding-mode techniques applied to an indoor micro quadrotor". In: *2005 IEEE Int. Conf. on Robotics and Automation*. May 2005, pp. 2247–2252 (cited on pages 23 and 109).

[BouAmmar–2010] H. Bou-Ammar, H. Voos, and W. Ertel. "Controller design for quadrotor uavs using reinforcement learning". In: *2010 IEEE International Conference on Control Applications*. IEEE. 2010, pp. 2130–2135 (cited on page 23).

[Bouffard–2012] P. Bouffard, A. Aswani, and C. Tomlin. "Learning-based model predictive control on a quadrotor: Onboard implementation and experimental results". In: *Robotics and Automation (ICRA), 2012 IEEE International Conference on*. IEEE. 2012, pp. 279–284 (cited on pages 27, 35 and 36).

[Brescianini–2016] D. Brescianini and R. D'Andrea. "Design, Modeling and Control of an Omni-Directional Aerial Vehicle". In: *2016 IEEE Int. Conf. on Robotics and Automation*. Stockholm, Sweden, May 2016, pp. 3261–3266 (cited on pages 13, 50 and 53).

[Bullo–2005] F. Bullo and A. D. Lewis. *Geometric control of mechanical systems: modeling, analysis, and design for simple mechanical control systems*. Springer New York, 2005 (cited on page 23).

[Cai–2014] G. Cai, J. Dias, and L. Seneviratne. "A survey of small-scale unmanned aerial vehicles: Recent advances and future development trends". In: *Unmanned Systems* 2.02 (2014), pp. 175–199 (cited on page 7).

[Campion–1993] G. Campion, G. Bastin, and B. D'Andrea-Novel. "Structural properties and classification of kinematic and dynamic models of wheeled mobile robots". In: *[1993] Proceedings IEEE International Conference on Robotics and Automation*. IEEE. 1993, pp. 462–469 (cited on page 3).

[Chen–2017] Y. Chen, D. Cuccato, M. Bruschetta, and A. Beghi. "A fast Nonlinear Model Predictive Control strategy for real-time motion control of mechanical systems". In: *Advanced Intelligent Mechatronics (AIM), 2017 IEEE International Conference on*. IEEE. 2017, pp. 1780–1785 (cited on pages 32 and 131).

[Chen–2018a] Y. Chen, M. Bruschetta, D. Cuccato, and A. Beghi. "An Adaptive Partial Sensitivity Updating Scheme for Fast Nonlinear Model Predictive Control". In: *IEEE Transactions on Automatic Control* (2018) (cited on page 131).

[Chen–2018b] Y. Chen, M. Bruschetta, E. Picotti, and A. Beghi. "MATMPC - A MATLAB Based Toolbox for Real-time Nonlinear Model Predictive Control". In: *CoRR* abs/1811.08761 (2018). arXiv: `1811.08761`. URL: `http://arxiv.org/abs/1811.08761` (cited on page 131).

[Convens–2017] B. Convens, K. Merckaert, M. M. Nicotra, R. Naldi, and E. Garone. "Control of Fully Actuated Unmanned Aerial Vehicles with Actuator Saturation". In: *20th IFAC World Congress*. Toulouse, France, July 2017, pp. 12715–12720 (cited on pages 23 and 55).

[Corke–2017] P. Corke. *Robotics, vision and control: fundamental algorithms In MATLAB® second, completely revised*. Vol. 118. Springer, 2017 (cited on page 43).

[Crowther–2011] B. Crowther, A. Lanzon, M. Maya-Gonzalez, and D. Langkamp. "Kinematic analysis and control design for a nonplanar multirotor vehicle". In: *AIAA Journal of Guidance, Control, and Dynamics* 34.4 (2011), pp. 1157–1171 (cited on pages 12 and 13).

[Dai–2014] S. Dai, T. Lee, and D. S. Bernstein. "Adaptive control of a quadrotor UAV transporting a cable-suspended load with unknown mass". In: *Decision and Control (CDC), 2014 IEEE 53rd Annual Conference on*. IEEE. 2014, pp. 6149–6154 (cited on page 23).

[Darivianakis–2014] G. Darivianakis, K. Alexis, M. Burri, and R. Siegwart. "Hybrid predictive control for aerial robotic physical interaction towards inspection operations". In: *Robotics and Automation (ICRA), 2014 IEEE International Conference on*. IEEE. 2014, pp. 53–58 (cited on pages 27, 28, 29 and 36).

[De Croon–2009] G. De Croon, K. De Clercq, R. Ruijsink, B. Remes, and C. De Wagter. "Design, aerodynamics, and vision-based control of the DelFly". In: *International Journal of Micro Air Vehicles* 1.2 (2009), pp. 71–97 (cited on page 6).

[De Luca–1998] A. De Luca. "Decoupling and feedback linearization of robots with mixed rigid/elastic joints". In: *International Journal of Robust and Nonlinear Control: IFAC-Affiliated Journal* 8.11 (1998), pp. 965–977 (cited on page 53).

[De Luca–2005] A. De Luca and R. Mattone. "Sensorless robot collision detection and hybrid force/motion control". In: *2005 IEEE Int. Conf. on Robotics and Automation*. Barcelona, Spain, Apr. 2005, pp. 999–1004 (cited on pages 9, 166 and 167).

[De Luca–2008] A. De Luca and W. Book. "Robots with Flexible Elements". In: *Springer Handbook of Robotics, B. Siciliano, O. Khatib*. Springer, 2008. Chap. 13, pp. 287–317 (cited on page 198).

[Diehl–2002] M. Diehl, H. G. Bock, J. P. Schlöder, R. Findeisen, Z. Nagy, and F. Allgöwer. "Real-time optimization and nonlinear model predictive control of processes governed by differential-algebraic equations". In: *Journal of Process Control* 12.4 (2002), pp. 577–585 (cited on page 131).

[Diehl–2006] M. Diehl, H. G. Bock, H. Diedam, and P.-B. Wieber. "Fast direct multiple shooting algorithms for optimal robot control". In: *Fast motions in biomechanics and robotics*. Springer, 2006, pp. 65–93 (cited on page 33).

[Du–2015] G. X. Du, Q. Quan, B. Yang, and K. Cai. "Controllability analysis for multirotor helicopter rotor degradation and failure". In: *AIAA Journal of Guidance, Control, and Dynamics* 38.5 (2015), pp. 978–985 (cited on page 52).

[Ducard–2011] G. Ducard and M.-D. Hua. "Discussion and Practical Aspects on Control Allocation for a Multi-rotor Helicopter". In: *1st Int. Conf. on UAVs in Geomatics*. Zurich, Switzerland, Sept. 2011, pp. 95–100 (cited on page 55).

[Euler–1736] L. Euler. *Mechanica: sive Motvs scientia analytice exposita*. v. 1. ex typographia Academiae scientiarvm, 1736. URL: `https://books.google.fr/books?id=NvhYAAAAYAAJ` (cited on page 38).

[Faessler–2018] M. Faessler, A. Franchi, and D. Scaramuzza. "Differential Flatness of Quadrotor Dynamics Subject to Rotor Drag for Accurate Tracking of High-Speed Trajectories". In: *IEEE Robotics and Automation Letters* 3.2 (2018), pp. 620–626 (cited on pages 50, 63 and 75).

[Falanga–2018] D. Falanga, P. Foehn, P. Lu, and D. Scaramuzza. "PAMPC: Perception-Aware Model Predictive Control for Quadrotors". In: *Intelligent Robots and Systems (IROS), 2018 IEEE/RSJ International Conference on*. IEEE/RSJ. 2018 (cited on pages 26, 27 and 132).

[Fantoni–2001] I. Fantoni and R. Lozano. *Non-linear control for underactuated mechanical systems*. Springer Science & Business Media, 2001 (cited on page 12).

[Fantoni–2002] I. Fantoni, A. Zavala, and R. Lozano. "Global stabilization of a PVTOL aircraft with bounded thrust". In: *Proceedings of the 41st IEEE Conference on Decision and Control, 2002.* Vol. 4. IEEE. 2002, pp. 4462–4467 (cited on page 23).

[Ferreau–2014] H. Ferreau, C. Kirches, A. Potschka, H. Bock, and M. Diehl. "qpOASES: A parametric active-set algorithm for quadratic programming". In: *Mathematical Programming Computation* 6.4 (2014), pp. 327–363 (cited on page 131).

[Filippone–2006] A. Filippone. *Flight performance of fixed and rotary wing aircraft*. Elsevier, 2006 (cited on page 7).

[Foehn–2018] P. Foehn and D. Scaramuzza. "Onboard State Dependent LQR for Agile Quadrotors". In: *Robotics and Automation (ICRA), 2018 IEEE International Conference on*. IEEE. 2018 (cited on pages 27, 30 and 36).

[Franchi–2017] A. Franchi and A. Mallet. "Adaptive Closed-loop Speed Control of BLDC Motors with Applications to Multi-rotor Aerial Vehicles". In: *2017 IEEE Int. Conf. on Robotics and Automation*. Singapore, May 2017, pp. 5203–5208 (cited on pages 9, 51, 65, 89, 113 and 128).

[Franchi–2018] A. Franchi, R. Carli, D. Bicego, and M. Ryll. "Full-Pose Tracking Control for Aerial Robotic Systems with Laterally-Bounded Input Force". In: *IEEE Trans. on Robotics* 34.2 (2018), pp. 534–541. DOI: 10.1109/TRO.2017.2786734 (cited on pages 16, 89 and 148).

[Fumagalli–2012] M. Fumagalli, R. Naldi, A. Macchelli, R. Carloni, S. Stramigioli, and L. Marconi. "Modeling and control of a flying robot for contact inspection". In: *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE. 2012, pp. 3532–3537 (cited on page 10).

[Fumagalli–2014] M. Fumagalli, R. Naldi, A. Macchelli, F. Forte, A. Q. Keemink, S. Stramigioli, R. Carloni, and L. Marconi. "Physical Interaction with the Environment". In: *IEEE Robotics & Automation Magazine* 21.3 (2014) (cited on page 188).

[Furci–2018] M. Furci, D. Bicego, and A. Franchi. "Design and Input Allocation for Robots with Saturated Inputs via Genetic Algorithms". In: *IFAC Symp. on Robot Control*. Budapest, Hungary, Aug. 2018 (cited on pages 22, 49 and 92).

[GarcíaMartín–2011] J. García-Martín, J. Gómez-Gil, and E. Vázquez-Sánchez. "Non-Destructive Techniques Based on Eddy Current Testing". In: *Sensors* 11.3 (2011), pp. 2525–2565 (cited on pages 187 and 195).

[Gasparin–2018] E. Gasparin, G. Santi, and A. Nussbaumer. "Eddy Current Crack Monitoring System for Structural Health Monitoring (SHM) Applications". In: *68th Int. Institute for Welding (IIW) Annual Assembly and Int. Conf.* Helsinki, Finland, July 2018, pp. 1084–1090 (cited on page 187).

[Geisert–2016] M. Geisert and N. Mansard. "Trajectory generation for quadrotor based systems using numerical optimal control". In: *Robotics and Automation (ICRA), 2016 IEEE International Conference on*. IEEE. 2016, pp. 2958–2964 (cited on pages 27, 36 and 65).

[Gioioso–2014a] G. Gioioso, A. Franchi, G. Salvietti, S. Scheggi, and D. Prattichizzo. "The Flying Hand: a Formation of UAVs for Cooperative Aerial Tele-Manipulation". In: *2014 IEEE Int. Conf. on Robotics and Automation*. Hong Kong, China, May 2014, pp. 4335–4341 (cited on page 10).

[Gioioso–2014b] G. Gioioso, M. Ryll, D. Prattichizzo, H. H. Bülthoff, and A. Franchi. "Turning a Near-hovering Controlled Quadrotor into a 3D Force Effector". In: *2014 IEEE Int. Conf. on Robotics and Automation*. Hong Kong, China, May 2014, pp. 6278–6284 (cited on pages 9 and 10).

[Giribet–2016a] J. I. Giribet, R. S. Sanchez-Pena, and A. S. Ghersin. "Analysis and design of a tilted rotor hexacopter for fault tolerance". In: *IEEE Trans. on Aerospace and Electronic System* 52.4 (2016), pp. 1555–1567 (cited on pages 52, 85 and 90).

[Giribet–2016b] J. I. Giribet, R. S. Sanchez-Pena, and A. S. Ghersin. "Analysis and design of a tilted rotor hexacopter for fault tolerance". In: *IEEE Transactions on Aerospace and Electronic Systems* 52.4 (2016), pp. 1555–1567 (cited on pages 147 and 153).

[Goldstein–2002] H. Goldstein, C. Poole, and J. Safko. *Classical mechanics*. 2002 (cited on pages 38 and 45).

[Goodarzi–2013] F. Goodarzi, D. Lee, and T. Lee. "Geometric nonlinear PID control of a quadrotor UAV on SE (3)". In: *Control Conference (ECC), 2013 European*. IEEE. 2013, pp. 3845–3850 (cited on page 23).

[Gouttefarde–2006] M. Gouttefarde and C. M. Gosselin. "Analysis of the wrench-closure workspace of planar parallel cable-driven mechanisms". In: *IEEE Transactions on Robotics* 22.3 (2006), pp. 434–445 (cited on page 15).

[Gouttefarde–2010] M. Gouttefarde, D. Daney, and J.-P. Merlet. "Interval-analysis-based determination of the wrench-feasible workspace of parallel cable-driven robots". In: *IEEE Transactions on Robotics* 27.1 (2010), pp. 1–13 (cited on page 15).

[Gress–2002] G. Gress. "Using dual propellers as gyroscopes for tilt-prop hover control". In: *2002 Biennial International Powered Lift Conference and Exhibit*. 2002, p. 5968 (cited on page 13).

[Grüne–2010] L. Grüne, J. Pannek, M. Seehafer, and K. Worthmann. "Analysis of Unconstrained Nonlinear MPC Schemes with Time Varying Control Horizon". In: *SIAM Journal on Control and Optimization* 48.8 (2010), pp. 4938–4962 (cited on page 81).

[Hamel–2002] T. Hamel, R. Mahony, R. Lozano, and J. Ostrowski. "Dynamic modelling and configuration stabilization for an X4-flyer." In: *IFAC Proceedings Volumes* 35.1 (2002), pp. 217–222 (cited on pages 7, 23, 46, 50 and 109).

[Hamilton–1834] W. Hamilton. *On a General Method in Dynamics*. Richard Taylor, 1834. URL: `https://books.google.fr/books?id=TAwuvwEACAAJ` (cited on page 38).

[Hofer–2016] M. Hofer, M. Muehlebach, and R. D'Andrea. "Application of an approximate model predictive control scheme on an unmanned aerial vehicle". In: *Robotics and Automation (ICRA), 2016 IEEE International Conference on*. IEEE. 2016, pp. 2952–2957 (cited on pages 6, 27, 28, 32 and 36).

[Hoffmann–2007] G. Hoffmann, H. Huang, S. Waslander, and C. Tomlin. "Quadrotor helicopter flight dynamics and control: Theory and experiment". In: *AIAA guidance, navigation and control conference and exhibit*. 2007, p. 6461 (cited on page 50).

[Hua–2013] M.-D. Hua, T. Hamel, P. Morin, and C. Samson. "Introduction to feedback control of underactuated VTOL vehicles: A review of basic control design ideas and principles". In: *IEEE Control Systems Magazine* 33.1 (2013), pp. 61–75 (cited on pages 23 and 49).

[Hua–2015a] M.-D. Hua, T. Hamel, P. Morin, and C. Samson. "Control of VTOL vehicles with thrust-tilting augmentation". In: *Automatica* 52 (2015), pp. 1–7 (cited on page 75).

[Hua–2015b] M.-D. Hua, T. Hamel, P. Morin, and C. Samson. "Control of VTOL vehicles with thrust-tilting augmentation". In: *Automatica* 52 (2015), pp. 1–7 (cited on pages 23, 50 and 56).

[Invernizzi–2018] D. Invernizzi, M. Lovera, and L. Zaccarian. "Dynamic attitude planning for trajectory tracking in underactuated VTOL UAVs". In: *arXiv preprint arXiv:1810.04494* (2018) (cited on page 75).

[Isidori–2013] A. Isidori. *Nonlinear control systems*. Springer Science & Business Media, 2013 (cited on page 53).

[Jolliffe–2011] I. Jolliffe. "Principal Component Analysis". In: *International Encyclopedia of Statistical Science*. Springer, 2011, pp. 1094–1096 (cited on page 195).

[Kamel–2015] M. Kamel, K. Alexis, M. Achtelik, and R. Siegwart. "Fast nonlinear model predictive control for multicopter attitude tracking on SO (3)". In: *Control Applications (CCA), 2015 IEEE Conference on*. IEEE. 2015, pp. 1160–1166 (cited on pages 27, 30, 36 and 51).

[Kamel–2016] M. Kamel, S. Comari, and R. Siegwart. "Full-body multi-objective controller for aerial manipulation". In: *24th Mediterranean Conf. on Control and Automation*. Athens, Greece, Aug. 2016, pp. 659–664 (cited on page 187).

[Kamel–2017a] M. Kamel, M. Burri, and R. Siegwart. "Linear vs nonlinear MPC for trajectory tracking applied to rotary wing micro aerial vehicles". In: *IFAC-PapersOnLine* 50.1 (2017), pp. 3463–3469 (cited on pages 27, 33, 36 and 132).

[Kamel–2017b] M. Kamel, T. Stastny, K. Alexis, and R. Siegwart. "Model predictive control for trajectory tracking of unmanned aerial vehicles using robot operating system". In: *Robot Operating System (ROS)*. Springer, 2017, pp. 3–39 (cited on page 33).

[Kamel–2018] M. Kamel, S. Verling, O. Elkhatib, C. Sprecher, P. Wulkop, Z. Taylor, R. Siegwart, and I. Gilitschenski. "Voliro: An omnidirectional hexacopter with tiltable rotors". In: *arXiv preprint arXiv:1801.04581* (2018) (cited on page 52).

[Kendoul–2006] F. Kendoul, I. Fantoni, and R. Lozano. "Modeling and control of a small autonomous aircraft having two tilting rotors". In: *IEEE Transactions on Robotics* 22.6 (2006), pp. 1297–1302 (cited on pages 13, 23 and 55).

[Khalil–2001] H. K. Khalil. *Nonlinear Systems*. 3rd. Prentice Hall, 2001. ISBN: 978-0130673893 (cited on pages 77 and 170).

[Khalil–2002] H. K. Khalil and J. W. Grizzle. *Nonlinear systems*. Vol. 3. Prentice hall Upper Saddle River, NJ, 2002 (cited on page 53).

[Kim–2013] 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*. Nov. 2013, pp. 4990–4995. DOI: 10.1109/IROS.2013.6697077 (cited on page 10).

[Kim–2018a] M. J. Kim, K. Kondak, and C. Ott. "A stabilizing controller for regulation of uav with manipulator". In: *IEEE Robotics and Automation Letters* 3.3 (2018), pp. 1719–1726 (cited on page 97).

[Kim–2018b] M. J. Kim, J. Lin, K. Kondak, D. Lee, and C. Ott. "Oscillation Damping Control of Pendulum-like Manipulation Platform using Moving Masses". In: *IFAC Symposium on Robot Control*. 2018 (cited on page 97).

[Knepper–2013] R. A. Knepper, T. Layton, J. Romanishin, and D. Rus. "IkeaBot: An autonomous multi-robot coordinated furniture assembly system". In: *2013 IEEE Int. Conf. on Robotics and Automation*. Karlsruhe, Germany, Oct. 2013, pp. 855–862 (cited on page 181).

[Kocer–2018] B. B. Kocer, T. Tjahjowidodo, and G. G. L. Seet. "Centralized predictive ceiling interaction control of quadrotor VTOL UAV". In: *Aerospace Science and Technology* 76 (2018), pp. 455–465 (cited on pages 27, 28 and 29).

[Kondak–2014] K. Kondak, F. Hubert, 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: *2014 IEEE Int. Conf. on Robotics and Automation*. Hong Kong, China, May 2014, pp. 2108–2112 (cited on pages 10 and 97).

[Lagrange–1811] J. Lagrange. *Mécanique analytique*. Mécanique analytique vol. 1. Courcier, 1811. URL: https://books.google.fr/books?id=I2A%5C_AAAAcAAJ (cited on page 38).

[Langkamp–2011] D. Langkamp, G. Roberts, A. Scillitoe, I. Lunnon, A. Llopis-Pascual, J. Zamecnik, S. Proctor, M. Rodriguez-Frias, M. Turner, A. Lanzon, and W. Crowther. "An engineering development of a novel hexrotor vehicle for 3D applications". In: *Proceedings of the International Micro Air Vehicle conference and competitions 2011* (2011) (cited on page 12).

[Lee–2009] D. Lee, H. J. Kim, and S. Sastry. "Feedback linearization vs. adaptive sliding mode control for a quadrotor helicopter". In: *International Journal of control, Automation and systems* 7.3 (2009), pp. 419–428 (cited on page 23).

[Lee–2010] T. Lee, M. Leoky, and N. H. McClamroch. "Geometric tracking control of a quadrotor UAV on SE(3)". In: *49th IEEE Conf. on Decision and Control*. Atlanta, GA, Dec. 2010, pp. 5420–5425 (cited on pages 23, 75, 76 and 109).

[Liew–2017] C. F. Liew, D. DeLatte, N. Takeishi, and T. Yairi. "Recent developments in aerial robotics: A survey and prototypes overview". In: *arXiv preprint arXiv:1711.10085* (2017) (cited on page 7).

[Ligthart–2017] J. A. Ligthart, P. Poksawat, L. Wang, and H. Nijmeijer. "Experimentally validated model predictive controller for a hexacopter". In: *IFAC-PapersOnLine* 50.1 (2017), pp. 4076–4081 (cited on pages 27, 30 and 36).

[Lin–2016] P. Lin, S. Chen, and C. Liu. "Model predictive control-based trajectory planning for quadrotors with state and input constraints". In: *Control, Automation and Systems (ICCAS), 2016 16th International Conference on*. IEEE. 2016, pp. 1618–1623 (cited on pages 26, 27 and 36).

[Lindsey–2012] Q. Lindsey, D. Mellinger, and V. Kumar. "Construction with quadrotor teams". In: *Autonomous Robots* 33.3 (2012), pp. 323–336 (cited on page 181).

[Lippiello–2012] V. Lippiello and F. Ruggiero. "Exploiting redundancy in Cartesian impedance control of UAVs equipped with a robotic arm". In: *2012 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*. Vilamoura, Portugal, Oct. 2012, pp. 3768–3773 (cited on page 10).

[Liu–2012] C. Liu, W.-H. Chen, and J. Andrews. "Explicit non-linear model predictive control for autonomous helicopters". In: *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering* 226.9 (2012), pp. 1171–1182 (cited on pages 27, 32 and 36).

[Liu–2015] Y. Liu, J. M. Montenbruck, P. Stegagno, F. Allgöwer, and A. Zell. "A robust nonlinear controller for nontrivial quadrotor maneuvers: Approach and verification". In: *Intelligent Robots and Systems (IROS), 2015 IEEE/RSJ International Conference on*. IEEE/RSJ. 2015, pp. 5410–5416 (cited on pages 26, 27 and 36).

[Ljung–1987] L. Ljung. *System identification: theory for the user*. Prentice-hall, 1987 (cited on page 233).

[Long–2013] Y. Long, L. Wang, and D. J. Cappelleri. "Modeling and global trajectory tracking control for an over-actuated MAV". In: *Advanced Robotics* 28.3 (2013), pp. 145–155 (cited on page 13).

[Lynch–2017] K. M. Lynch and F. C. Park. *Modern Robotics: Mechanics, Planning, and Control*. Cambridge University Press, 2017 (cited on page 39).

[Machado–2016] T. Machado, T. Malheiro, S. Monteiro, W. Erlhagen, and E. Bicho. "Multi-constrained joint transportation tasks by teams of autonomous mobile robots using a dynamical systems approach". In: *2016 IEEE Int. Conf. on Robotics and Automation*. Stockholm, Sweden, May 2016, pp. 3111–3117 (cited on page 181).

[Mahony–2012a] R. Mahony, V. Kumar, and P. Corke. "Multirotor Aerial Vehicles: Modeling, Estimation, and Control of Quadrotor". In: *IEEE Robotics & Automation Magazine* 19.3 (2012), pp. 20–32 (cited on page 7).

[Mahony–2012b] R. Mahony, V. Kumar, and P. Corke. "Multirotor aerial vehicles". In: *IEEE Robotics and Automation magazine* 20.32 (2012) (cited on pages 46, 50 and 51).

[Martin–1996] P. Martin, S. Devasia, and B. Paden. "A different look at output tracking: control of a VTOL aircraft". In: *Automatica* 32.1 (1996), pp. 101–107 (cited on page 53).

[Mayne–2000] D. Mayne, J. Rawlings, C. Rao, and P. Scokaert. "Constrained model predictive control: Stability and optimality". In: *Automatica* 36.6 (2000), pp. 789–814 (cited on page 81).

[McKinnon–2016] C. D. McKinnon and A. P. Schoellig. "Unscented external force and torque estimation for quadrotors". In: *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE. 2016, pp. 5651–5657 (cited on page 10).

[Mehrez–2017] M. W. Mehrez, K. Worthmann, G. K. Mann, R. G. Gosine, and T. Faulwasser. "Predictive Path Following of Mobile Robots without Terminal Stabilizing Constraints". In: *IFAC-PapersOnLine* 50.1 (2017), pp. 9852–9857. ISSN: 2405-8963 (cited on page 81).

[Mellinger–2011] D. Mellinger and V. Kumar. "Minimum snap trajectory generation and control for quadrotors". In: *2011 IEEE International Conference on Robotics and Automation*. IEEE. 2011, pp. 2520–2525 (cited on pages 46, 63 and 75).

[Michael–2010] N. Michael, D. Mellinger, Q. Lindsey, and V. Kumar. "The grasp multiple micro-uav testbed". In: *IEEE Robot. Automat. Mag.* 17.3 (2010), pp. 56–65 (cited on page 23).

[Michieletto–2017] G. Michieletto, M. Ryll, and A. Franchi. "Control of Statically Hoverable Multi-Rotor Aerial Vehicles and Application to Rotor-Failure Robustness for Hexarotors". In: *2017 IEEE Int. Conf. on Robotics and Automation*. Singapore, May 2017, pp. 2747–2752 (cited on pages 52, 90, 147, 153, 154 and 157).

[Michieletto–2018] G. Michieletto, M. Ryll, and A. Franchi. "Fundamental Actuation Properties of Multi-rotors: Force-Moment Decoupling and Fail-safe Robustness". In: *IEEE Trans. on Robotics* 34.3 (2018), pp. 702–715. DOI: 10.1109/TRO.2018.2821155 (cited on pages 52, 90, 153, 154 and 157).

[Miro–2017] J. V. Miro, D. Hunt, N. Ulapane, and M. Behrens. "Towards Automatic Robotic NDT Dense Mapping for Pipeline Integrity Inspection". In: *11th Int. Conf. on Field and Service Robotics*. Zurich, Switzerland, Sept. 2017, pp. 319–333 (cited on page 187).

[Mistler–2001] V. Mistler, A. Benallegue, and N. K. M'Sirdi. "Exact linearization and noninteracting control of a 4 rotors helicopter via dynamic feedback". In: *10th IEEE Int. Symp. on Robots and Human Interactive Communications*. Bordeaux, Paris, France, Sept. 2001, pp. 586–593 (cited on pages 63, 75 and 109).

[Mohammadi–2017] M. Mohammadi. "Bilateral Aerial Tele-Manipulation: Single and Multi-Robot Approaches". PhD thesis. University of Siena, 2017 (cited on page 10).

[Morbidi–2018] F. Morbidi, D. Bicego, M. Ryll, and A. Franchi. "Energy-Efficient Trajectory Generation for a Hexarotor with Dual-Tilting Propellers". In: *2018 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*. Madrid, Spain, Oct. 2018 (cited on page 16).

[Morin–2015] P. Morin. "Modeling and control of convertible Micro Air Vehicles". In: *2015 10th International Workshop on Robot Motion and Control (RoMoCo)*. IEEE. 2015, pp. 188–198 (cited on page 6).

[Mueller–2013a] M. W. Mueller and R. D'Andrea. "A model predictive controller for quadrocopter state interception". In: *Control Conference (ECC), 2013 European*. IEEE. 2013, pp. 1383–1389 (cited on pages 26, 27, 31, 36 and 51).

[Mueller–2013b] M. W. Mueller, M. Hehn, and R. D'Andrea. "A computationally efficient algorithm for state-to-state quadrocopter trajectory generation and feasibility verification". In: *Intelligent Robots and Systems (IROS), 2013 IEEE/RSJ International Conference on*. IEEE/RSJ. 2013, pp. 3480–3486 (cited on pages 26 and 27).

[Mueller–2015] M. W. Mueller, M. Hehn, and R. D'Andrea. "A computationally efficient motion primitive for quadrocopter trajectory generation". In: *IEEE Transactions on Robotics* 31.6 (2015), pp. 1294–1310 (cited on pages 26 and 36).

[Murray–2017] R. M. Murray. *A mathematical introduction to robotic manipulation*. CRC press, 2017 (cited on page 3).

[Muscio–2016] G. Muscio, F. Pierri, M. A. Trujillo, E. Cataldi, G. Giglio, G. Antonelli, F. Caccavale, A. Viguria, S. Chiaverini, and A. Ollero. "Experiments on coordinated motion of aerial robotic manipulators". In: *2016 IEEE Int. Conf. on Robotics and Automation*. Stockholm, Sweden, May 2016, pp. 1224–1229 (cited on pages 10 and 11).

[Muscio–2017] G. Muscio, F. Pierri, M. A. Trujillo, E. Cataldi, G. Antonelli, F. Caccavale, A. Viguria, S. Chiaverini, and A. Ollero. "Coordinated Control of Aerial Robotic Manipulators: Theory and Experiments". In: *IEEE Transactions on Control Systems Technology* PP.99 (2017), pp. 1–8. ISSN: 1063-6536. DOI: 10.1109/TCST.2017.2716905 (cited on pages 10 and 11).

[Naldi–2010] R. Naldi, L. Gentili, L. Marconi, and A. Sala. "Design and Experimental Validation of a Nonlinear Control Law for a Ducted-Fan Miniature Aerial Vehicle". In: *Control Engineering Practice* 18.7 (2010), pp. 747–760 (cited on page 6).

[Neunert–2016] M. Neunert, C. De Crousaz, F. Furrer, M. Kamel, F. Farshidian, R. Siegwart, and J. Buchli. "Fast nonlinear model predictive control for unified trajectory optimization and tracking". In: *Robotics and Automation (ICRA), 2016 IEEE International Conference on*. IEEE. 2016, pp. 1398–1404 (cited on pages 27, 28, 30 and 36).

[Newton–1687] I. Newton. *Philosophiae naturalis principia mathematica*. J. Societatis Regiae ac Typis J. Streater, 1687. URL: https://books.google.fr/books?id=-dVKAQAAIAAJ (cited on page 38).

[Nguyen–2013] H. Nguyen and D. Lee. "Hybrid Force/Motion Control and Internal Dynamics of Quadrotors for Tool Operation". In: *2013 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*. Tokyo, Japan, Nov. 2013, pp. 3458–3464 (cited on page 10).

[Nguyen–2015] H. Nguyen, C. Ha, and D. Lee. "Mechanics, control and internal dynamics of quadrotor tool operation". In: *Automatica* 61 (2015), pp. 289–301 (cited on pages 10 and 196).

[Nikou–2015] A. Nikou, G. C. Gavridis, and K. J. Kyriakopoulos. "Mechanical design, modelling and control of a novel aerial manipulator". In: *2015 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE. 2015, pp. 4698–4703 (cited on page 13).

[Ollero–2018] A. Ollero, G. Heredia, A. Franchi, G. Antonelli, K. Kondak, A. Sanfeliu, A. Viguria, J. R. Martinez-de Dios, F. Pierri, J. Cortés, A. Santamaria-Navarro, M. A. Trujillo, R. Balachandran, J. Andrade-Cetto, and A. Rodriguez. "The AEROARMS Project: Aerial Robots with Advanced Manipulation Capabilities for Inspection and Maintenance". In: *IEEE Robotics &*

Automation Magazine, Special Issue on Floating-base (Aerial and Underwater) Manipulation 25.4 (2018), pp. 12–23 (cited on pages 21 and 187).

[Oosedo–2015] A. Oosedo, S. Abiko, S. Narasaki, A. Kuno, A. Konno, and M. Uchiyama. "Flight control systems of a quad tilt rotor unmanned aerial vehicle for a large attitude change". In: *2015 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE. 2015, pp. 2326–2331 (cited on page 14).

[Oriolo–2002] G. Oriolo, A. De Luca, and M. Vendittelli. "WMR control via dynamic feedback linearization: design, implementation, and experimental validation". In: *IEEE Transactions on control systems technology* 10.6 (2002), pp. 835–852 (cited on page 53).

[Pajares–2015] G. Pajares. "Overview and current status of remote sensing applications based on unmanned aerial vehicles (UAVs)". In: *Photogrammetric Engineering & Remote Sensing* 81.4 (2015), pp. 281–330 (cited on page 5).

[Papachristos–2013a] C. Papachristos, K. Alexis, and A. Tzes. "Model predictive hovering-translation control of an unmanned tri-tiltrotor". In: *Robotics and Automation (ICRA), 2013 IEEE International Conference on*. IEEE. May 2013, pp. 5425–5432 (cited on pages 27, 34 and 36).

[Papachristos–2013b] C. Papachristos, K. Alexis, and A. Tzes. "Hybrid model predictive flight mode conversion control of unmanned quad-tiltrotors". In: *Control Conference (ECC), 2013 European*. IEEE. July 2013, pp. 1793–1798 (cited on pages 27 and 34).

[Papachristos–2016] C. Papachristos, K. Alexis, and A. Tzes. "Dual–authority thrust–vectoring of a tri–tiltrotor employing model predictive control". In: *Journal of intelligent & robotic systems* 81.3-4 (2016), pp. 471–504 (cited on pages 34 and 36).

[Park–2016] S. Park, J. J. Her, J. Kim, and D. Lee. "Design, Modeling and Control of Omni-Directional Aerial Robot". In: *2016 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*. Daejeon, South Korea, 2016, pp. 1570–1575 (cited on pages 52, 53 and 196).

[Park–2018] S. Park, J. Lee, J. Ahn, M. Kim, J. Her, G.-H. Yang, and D. Lee. "ODAR: Aerial Manipulation Platform Enabling Omnidirectional Wrench Generation". In: *IEEE/ASME Transactions on Mechatronics* 23.4 (2018), pp. 1907–1918 (cited on pages 13, 50 and 68).

[Paul–1981] R. P. Paul. *Robot manipulators: mathematics, programming, and control: the computer control of robot manipulators*. Richard Paul, 1981 (cited on page 3).

[Pounds–2010a] P. E. Pounds, R. Mahony, and P. Corke. "Modeling and Control of a Large Quadrotor Robot". In: *Control Engineering Practice* 18.7 (2010), pp. 691–699 (cited on page 23).

[Pounds–2010b] P. Pounds, R. Mahony, and P. Corke. "Modelling and control of a large quadrotor robot". In: *Control Engineering Practice* 18.7 (2010). Special Issue on Aerial Robotics, pp. 691–699. ISSN: 0967-0661. DOI: https://doi.org/10.1016/j.conengprac.2010.02.008. URL: http://www.sciencedirect.com/science/article/pii/S0967066110000456 (cited on page 46).

[Raibert–1986] M. H. Raibert. *Legged robots that balance*. MIT press, 1986 (cited on page 3).

[Rajappa–2015] S. Rajappa, M. Ryll, H. H. Bülthoff, and A. Franchi. "Modeling, Control and Design Optimization for a Fully-actuated Hexarotor Aerial Vehicle with Tilted Propellers". In: *2015 IEEE Int. Conf. on Robotics and Automation*. Seattle, WA, May 2015, pp. 4006–4013 (cited on pages 14, 49, 52, 53, 89, 114, 117 and 223).

[Rajappa–2017] S. Rajappa, H. Bülthoff, and P. Stegagno. "Design and implementation of a novel architecture for physical human-UAV interaction". In: *The International Journal of Robotics Research* (2017), p. 0278364917708038 (cited on page 9).

[Ren–2012] B. Ren, S. S. Ge, C. Chen, C.-H. Fua, and T. H. Lee. *Modeling, control and coordination of helicopter systems*. Springer Science & Business Media, 2012 (cited on page 6).

[Romero–2007] H. Romero, S. Salazar, A. Sanchez, and R. Lozano. "A new UAV configuration having eight rotors: dynamical model and real-time control". In: *46th IEEE Conf. on Decision and Control*. New Orleans, LA, Dec. 2007, pp. 6418–6423 (cited on pages 12, 23, 55 and 109).

[Roque–2016] P. Roque and R. Ventura. "Space CoBot: Modular design of an holonomic aerial robot for indoor microgravity environments". In: *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE. 2016, pp. 4383–4390 (cited on page 15).

[Ruggiero–2015] F. Ruggiero, M. A. Trujillo, R. Cano, H. Ascorbe, A. Viguria, C. Peréz, V. Lippiello, A. Ollero, and B. Siciliano. "A multilayer control for multirotor UAVs equipped with a servo robot arm". In: *2015 IEEE Int. Conf. on Robotics and Automation*. Seattle, WA, May 2015, pp. 4014–4020 (cited on pages 187 and 198).

[Ruggiero–2018] F. Ruggiero, V. Lippiello, and A. Ollero. "Aerial Manipulation: A Literature Review". In: *IEEE Robotics and Automation Letters* 3.3 (2018), pp. 1957–1964 (cited on page 10).

[Ryll–2012a] M. Ryll, H. H. Bülthoff, and P. Robuffo Giordano. "Modeling and Control of a Quadrotor UAV with Tilting Propellers". In: *2012 IEEE Int. Conf. on Robotics and Automation*. St. Paul, MN, May 2012, pp. 4606–4613 (cited on page 13).

[Ryll–2012b] M. Ryll, H. H. Bülthoff, and P. R. Giordano. "Modeling and control of a quadrotor UAV with tilting propellers". In: *2012 IEEE International Conference on Robotics and Automation*. IEEE. 2012, pp. 4606–4613 (cited on page 46).

[Ryll–2015] M. Ryll, H. H. Bülthoff, and P. Robuffo Giordano. "A novel overactuated quadrotor unmanned aerial vehicle: modeling, control, and experimental validation". In: *IEEE Trans. on Control Systems Technology* 23.2 (2015), pp. 540–556 (cited on pages 13, 46, 49 and 51).

[Ryll–2016] M. Ryll, D. Bicego, and A. Franchi. "Modeling and Control of FAST-Hex: a Fully-Actuated by Synchronized-Tilting Hexarotor". In: *2016 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*. Daejeon, South Korea, Oct. 2016, pp. 1689–1694 (cited on pages 16, 88, 148 and 196).

[Ryll–2017] M. Ryll, G. Muscio, F. Pierri, E. Cataldi, G. Antonelli, F. Caccavale, and A. Franchi. "6D Physical Interaction with a Fully Actuated Aerial Robot". In: *2017 IEEE Int. Conf. on Robotics and Automation*. Singapore, May 2017, pp. 5190–5195 (cited on pages 148, 187 and 188).

[Ryll–2018] M. Ryll, D. Bicego, and A. Franchi. "A Truly Redundant Aerial Manipulator exploiting a Multi-directional Thrust Base". In: *2018 IFAC Symp. on Robot Control*. Budapest, Hungary, Aug. 2018, pp. 138–143 (cited on pages 10, 22, 188 and 197).

[Ryll–2019] M. Ryll, G. Muscio, F. Pierri, E. Cataldi, G. Antonelli, F. Caccavale, D. Bicego, and A. Franchi. "6D Interaction Control with Aerial Robots: The Flying End-Effector Paradigm". In: *The International Journal of Robotics Research* 38.9 (2019), pp. 1045–1062. DOI: 10.1177/0278364919856694 (cited on page 17).

[Salazar Cruz–2005] S. Salazar Cruz and R. Lozano. "Stabilization and nonlinear control for a novel trirotor mini-aircraft". In: *Proceedings of the 2005 IEEE international conference on robotics and automation*. IEEE. 2005, pp. 2612–2617 (cited on page 13).

[Salazar–2009] S. Salazar, H. Romero, R. Lozano, and P. Castillo. "Modeling and Real-Time Stabilization of an Aircraft Having Eight Rotors". In: *Journal of Intelligent and Robotic Systems* 54.1 (Mar. 2009), pp. 455–470. ISSN: 1573-0409. DOI: 10.1007/s10846-008-9274-x. URL: https://doi.org/10.1007/s10846-008-9274-x (cited on page 12).

[Sanchez Cuevas–2017] P. Sanchez Cuevas, G. Heredia, and A. Ollero. "Characterization of the aerodynamic ground effect and its influence in multirotor control". In: *International Journal of Aerospace Engineering* 2017 (2017) (cited on page 50).

[Sarkisov–2019] Y. Sarkisov, M. J. Kim, D. Bicego, D. Tsetserukou, C. Ott, A. Franchi, and K. Kondak. "Development of SAM: cable-Suspended Aerial Manipulator". In: *2019 IEEE Int. Conf. on Robotics and Automation*. Montreal, Canada, May 2019 (cited on page 16).

[Schoellig–2012] A. P. Schoellig, F. L. Mueller, and R. D'Andrea. "Optimization-based iterative learning for precise quadrocopter trajectory tracking". In: *Autonomous Robots* 33.1 (Aug. 2012), pp. 103–127. ISSN: 1573-7527. DOI: 10.1007/s10514-012-9283-2. URL: https://doi.org/10.1007/s10514-012-9283-2 (cited on page 23).

[Segui Gasco–2014] P. Segui Gasco, Y. Al-Rihani, H.-S. Shin, and A. Savvaris. "A Novel Actuation Concept for a Multi Rotor UAV". In: *Journal of Intelligent & Robotic Systems* 74.1 (Apr. 2014), pp. 173–191. ISSN: 1573-0409. DOI: 10.1007/s10846-013-9987-3. URL: https://doi.org/10.1007/s10846-013-9987-3 (cited on page 14).

[Siciliano–2009] B. Siciliano, L. Sciavicco, L. Villani, and G. Oriolo. *Robotics: Modelling, Planning and Control*. Springer, 2009. ISBN: 978-1-84628-641-4 (cited on page 170).

[Siciliano–2010] B. Siciliano, L. Sciavicco, L. Villani, and G. Oriolo. *Robotics: modelling, planning and control*. Springer Science & Business Media, 2010 (cited on pages 3, 39 and 43).

[Siciliano–2016] B. Siciliano and O. Khatib. *Springer handbook of robotics*. Springer, 2016 (cited on pages 3, 39 and 43).

[Slotine–1991] J.-J. E. Slotine, W. Li, et al. *Applied nonlinear control*. Vol. 199. 1. Prentice hall Englewood Cliffs, NJ, 1991 (cited on page 53).

[Spong–1998] M. W. Spong. "Underactuated mechanical systems". In: *Control problems in robotics and automation*. Springer, 1998, pp. 135–150 (cited on page 12).

[Spong–2006] M. W. Spong, S. Hutchinson, M. Vidyasagar, et al. *Robot modeling and control*. 2006 (cited on pages 39 and 43).

[Sreenath–2013] K. Sreenath and V. Kumar. "Dynamics, Control and Planning for Cooperative Manipulation of Payloads Suspended by Cables from Multiple Quadrotor Robots". In: *Robotics: Science and Systems*. Berlin, Germany, June 2013 (cited on pages 10 and 181).

[Staub–2017] N. Staub, M. Mohammadi, D. Bicego, D. Prattichizzo, and A. Franchi. "Towards Robotic MAGMaS: Multiple Aerial-Ground Manipulator Systems". In: *2017 IEEE Int. Conf. on Robotics and Automation*. Singapore, May 2017 (cited on pages 10, 22 and 177).

[Staub–2018a] N. Staub, D. Bicego, Q. Sablé, V. Arellano-Quintana, S. Mishra, and A. Franchi. "Towards a Flying Assistant Paradigm: the OTHex". In: *2018 IEEE Int. Conf. on Robotics and Automation*. Brisbane, Australia, May 2018, pp. 6997–7002 (cited on pages 17, 21 and 148).

[Staub–2018b] N. Staub, M. Mohammadi, D. Bicego, Q. Delamare, H. Yang, D. Prattichizzo, P. Robuffo Giordano, D. Lee, and A. Franchi. "The Tele-MAGMaS: an Aerial-Ground Co-manipulator System". In: *IEEE Robotics & Automation Magazine* 25.4 (2018), pp. 66–75. DOI: 10.1109/MRA.2018.2871344 (cited on pages 17 and 208).

[Staub–2018c] N. Staub. "Models, Algorithms and Architectures for Cooperative Manipulation with Aerial and Ground Robots". PhD thesis. LAAS-CNRS, Université de Toulouse, 2018 (cited on page 7).

[Suarez–2015] A. Suarez, G. Heredia, and A. Ollero. "Lightweight compliant arm for aerial manipulation". In: *2015 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*. Hamburg, Germany, Sept. 2015, pp. 1627–1632 (cited on page 10).

[Suarez–2018a] A. Suarez, A. Giordano, K. Kondak, G. Heredia, and A. Ollero. "Flexible link long reach manipulator with lightweight dual arm: Soft-collision detection, reaction, and obstacle localization". In: *IEEE International Conference on Soft Robotics (RoboSoft)*. 2018, pp. 406–411 (cited on page 97).

[Suarez–2018b] A. Suarez, P. Sanchez-Cuevas, M. Fernandez, M. Perez, G. Heredia, and A. Ollero. "Lightweight and Compliant Long Reach Aerial Manipulator for Inspection Operations". In: *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. 2018 (cited on page 97).

[Tagliabue–2016] A. Tagliabue, M. Kamel, S. Verling, R. Siegwart, and J. Nieto. "Collaborative transportation using MAVs via passive force control". In: *2017 IEEE Int. Conf. on Robotics and Automation*. Singapore, 2016, pp. 5766–5773 (cited on page 10).

[Tognon–2017] M. Tognon, B. Yüksel, G. Buondonno, and A. Franchi. "Dynamic Decentralized Control for Protocentric Aerial Manipulators". In: *2017 IEEE Int. Conf. on Robotics and Automation*. Singapore, May 2017, pp. 6375–6380 (cited on pages 10 and 188).

[Tognon–2018a] M. Tognon, E. Cataldi, H. Tello Chavez, G. Antonelli, J. Cortés, and A. Franchi. "Control-Aware Motion Planning for Task-Constrained Aerial Manipulation". In: *IEEE Robotics and Automation Letters, Special Issue on Aerial Manipulation* 3.3 (2018), pp. 2478–2484. DOI: 10.1109/LRA.2018.2803206 (cited on page 197).

[Tognon–2018b] M. Tognon and A. Franchi. "Omnidirectional Aerial Vehicles with Unidirectional Thrusters: Theory, Optimal Design, and Control". In: *IEEE Robotics and Automation Letters* 3.3 (2018), pp. 2277–2282. DOI: 10.1109/LRA.2018.2802544 (cited on pages 13, 50, 100, 101 and 224).

[Tognon–2018c] M. Tognon. "Theory and Applications for Control and Motion Planning of Aerial Robots in Physical Interaction with particular focus on Tethered Aerial Vehicles". PhD thesis. LAAS-CNRS, Université de Toulouse, 2018 (cited on page 10).

[Tognon–2019] M. Tognon, H. A. Tello Chávez, E. Gasparin, Q. Sablé, D. Bicego, A. Mallet, M. Lany, G. Santi, B. Revaz, J. Cortés, and A. Franchi. "A Truly Redundant Aerial Manipulator System with Application to Push-and-Slide Inspection in Industrial Plants". In: *IEEE Robotics and Automation Letters* 4.2 (2019), pp. 1846–1851. DOI: 10.1109/LRA.2019.2895880 (cited on pages 17, 197, 198 and 201).

[Tomic–2014] T. Tomic and S. Haddadin. "A unified framework for external wrench estimation, interaction control and collision reflexes for flying robots". In: *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*. 2014, pp. 4197–4204 (cited on page 9).

[Tomic–2017] T. Tomic, C. Ott, and S. Haddadin. "External Wrench Estimation, Collision Detection, and Reflex Reaction for Flying Robots". In: *IEEE Transactions on Robotics* PP.99 (2017), pp. 1–17. ISSN: 1552-3098. DOI: 10.1109/TRO.2017.2750703 (cited on pages 9 and 166).

[Trujillo–2019] M. Á. Trujillo, J. R. Martínez-de Dios, C. Martín, A. Viguria, and A. Ollero. "Novel Aerial Manipulator for Accurate and Robust Industrial NDT Contact Inspection: A New Tool for the Oil and Gas Inspection Industry". In: *Sensors* 19.6 (2019), p. 1305 (cited on pages 216 and 219).

[video01–2017] **video01**. *Full-Pose Tracking Control for Aerial Robotic Systems With Laterally Bounded Input Force*. https://www.youtube.com/watch?v=pCKfqRSSH4g&list=UUFoSjtoDdbt4G4mL7p1jciA&index=10. 2017 (cited on page 115).

[video02–2017] **video02**. *Full-Pose Tracking Control for Aerial Robotic Systems With Laterally Bounded Input Force*. https://www.youtube.com/watch?v=8TQFzNtjuSE&list=UUFoSjtoDdbt4G4mL7p1jciA&index=11. 2017 (cited on page 118).

[video03–2017] **video03**. *Full-Pose Tracking Control for Aerial Robotic Systems With Laterally Bounded Input Force*. https://www.youtube.com/watch?v=irjKUMOyttc&list=UUFoSjtoDdbt4G4mL7p1jciA&index=12. 2017 (cited on page 120).

[video04–2019] **video04**. *Nonlinear Model Predictive Control with Actuator Constraints for Multi-Rotor Aerial Vehicles*. https://www.youtube.com/watch?v=sNOD3_MTHDc. 2019 (cited on pages 136, 142 and 145).

[video05–2019] **video05**. *6D Interaction Control with Aerial Robots: The Flying End-Effector Paradigm*. https://www.youtube.com/watch?v=SDe-2bl_XOk&t=11s. 2019 (cited on pages 171 and 239).

[video06–2019] **video06**. *6D Interaction Control with Aerial Robots: The Flying End-Effector Paradigm*. https://www.youtube.com/watch?v=jQ6qr38Lulg. 2019 (cited on pages 193 and 239).

[video07–2019] **video07**. *6D Interaction Control with Aerial Robots: The Flying End-Effector Paradigm*. https://www.youtube.com/watch?v=5lsje-QtMf0&t=7s. 2019 (cited on page 239).

[video08–2018] **video08**. *Towards a Flying Assistant Paradigm: the OTHex*. https://www.youtube.com/watch?v=AikN3_PgYU4&list=UUFoSjtoDdbt4G4mL7p1jciA&index=15. 2018 (cited on page 181).

[video09–2018] **video09**. *Tele-MAGMaS: an Aerial-Ground Co-Manipulator System*. https://www.youtube.com/watch?v=TrrPEP3CNlY&list=UUFoSjtoDdbt4G4mL7p1jciA&index=8. 2018 (cited on page 183).

[video10–2018] **video10**. *A Truly Redundant Aerial Manipulator System with Application to Push-and-Slide Inspection in Industrial Plants*. https://www.youtube.com/watch?v=_n_f617b-Ho&list=UUFoSjtoDdbt4G4mL7p1jciA&index=7. 2018 (cited on pages 189 and 199).

[video11–2017] **video11**. *Tele-MAGMaS at Hanover Fair 2017: KUKA video*. https://vimeo.com/217252361. 2017 (cited on page 211).

[video12–2017] **video12**. *Tele-MAGMaS at Hanover Fair 2017: interview 1*. https://www.youtube.com/watch?v=o9xmPYNPfRE&list=UUFoSjtoDdbt4G4mL7p1jciA&index=19. 2017 (cited on page 211).

[video13–2017] **video13**. *Tele-MAGMaS at Hanover Fair 2017: interview 2*. https://youtu.be/GRnGSvJGUKk. 2017 (cited on page 211).

[Villani–2008] L. Villani and J. De Schutter. "Springer Handbook of Robotics". In: Heidelberg, D: B. Siciliano, O. Khatib, (Eds.), Springer-Verlag, 2008. Chap. Force Control (cited on pages 165 and 171).

[Voyles–2012] R. Voyles and G. Jiang. "Hexrotor UAV platform enabling dextrous interaction with structures – preliminary work". In: *2012 IEEE Int. Symp. on Safety, Security and Rescue Robotics*. College Station, TX, Nov. 2012, pp. 1–7 (cited on pages 13, 52, 109 and 223).

[Voyles–2014] R. Voyles and G. Jiang. "A nonparallel hexrotor UAV with faster response to disturbances for precision position keeping". In: *2014 IEEE Int. Symp. on Safety, Security and Rescue Robotics*. West Lafayette, IN, Oct. 2014, pp. 1–5 (cited on page 52).

[Wit–1996] C. C. de Wit, B. Siciliano, and G. Bastin. "Motion and force control". In: *Theory of robot control*. Springer, 1996. Chap. 4, pp. 141–175 (cited on page 197).

[Yang–2014] H. Yang and D. Lee. "Dynamics and control of quadrotor with robotic manipulator". In: *2014 IEEE International Conference on Robotics and Automation (ICRA)*. May 2014, pp. 5544–5549. DOI: 10.1109/ICRA.2014.6907674 (cited on page 10).

[Yüksel–2014a] B. Yüksel, C. Secchi, H. H. Bülthoff, and A. Franchi. "A Nonlinear Force Observer for Quadrotors and Application to Physical Interactive Tasks". In: *2014 IEEE/ASME Int. Conf. on Advanced Intelligent Mechatronics*. Besançon, France, July 2014, pp. 433–440 (cited on pages 9, 10, 166 and 167).

[Yüksel–2014b] B. Yüksel, C. Secchi, H. H. Bülthoff, and A. Franchi. "Reshaping the Physical Properties of a Quadrotor through IDA-PBC and its Application to Aerial Physical Interaction". In: *2014 IEEE Int. Conf. on Robotics and Automation*. Hong Kong, China, May 2014, pp. 6258–6265 (cited on pages 10 and 12).

[Yüksel–2017] B. Yüksel. "Design, Modeling and Control of Aerial Robots for Physical Interaction and Manipulation". PhD thesis. Max-Planck-Institute, University of Tübingen, 2017 (cited on page 10).

[Zhao–2014] B. Zhao, B. Xian, Y. Zhang, and X. Zhang. "Nonlinear robust adaptive tracking control of a quadrotor UAV via immersion and invariance methodology". In: *IEEE Transactions on Industrial Electronics* 62.5 (2014), pp. 2891–2902 (cited on page 23).