

ESCOLA DE ENGENHARIA DE SÃO CARLOS

João Pedro Sansão Buzzatto

**Manipulação Aérea Baseada em Feedback de Força e
Torque**



São Carlos

2019



João Pedro Sansão Buzzatto

Manipulação Aérea Baseada em Feedback de Força e Torque



Dissertação apresentada à Escola de Engenharia de São Carlos da Universidade de São Paulo, para obtenção do título de Mestre em Ciências - Programa de Pós-Graduação em Engenharia Mecânica.

Área de concentração: Dinâmica e Mecatrônica

Supervisor: Prof. Dr. Glauco Augusto de Paula Caurin

ESTE EXEMPLAR TRATA-SE DA VERSÃO CORRIGIDA. A VERSÃO ORIGINAL ENCONTRA-SE DISPONÍVEL JUNTO AO DEPARTAMENTO DE ENGENHARIA MECANICA DA EESC-USP.

São Carlos

2019

Class.	TESE
Cott.	10.416
Tombo	T240/19
Sysno	2961962

3 L 1002132 01

10 09 19

AUTORIZO A REPRODUÇÃO TOTAL OU PARCIAL DESTA TRABALHO,
POR QUALQUER MEIO CONVENCIONAL OU ELETRÔNICO, PARA FINS
DE ESTUDO E PESQUISA, DESDE QUE CITADA A FONTE.

Ficha catalográfica elaborada pela Biblioteca Prof. Dr. Sérgio Rodrigues Fontes da
EESC/USP com os dados inseridos pelo(a) autor(a).

Sansão Buzzatto, João Pedro
Manipulação Aérea Baseada em Feedback de Força e
Torque / João Pedro Sansão Buzzatto; orientador Glauco
Augusto de Paula Caurin. São Carlos, 2019.

Dissertação (Mestrado) - Programa de Pós-Graduação
em Engenharia Mecânica e Área de Concentração em
Dinâmica e Mecatrônica -- Escola de Engenharia de São
Carlos da Universidade de São Paulo, 2019.

1. Manipulação Aérea. 2. UAV. 3. Controle. 4.
Sensor de Força. 5. Sensor de Torque. I. Título.

FOLHA DE JULGAMENTO

Candidato: Engenheiro **JOÃO PEDRO SANSÃO BUZZATTO**.

Título da dissertação: "Manipulação aérea baseada em feedback de força e torque".

Data da defesa: 15/04/2019

Comissão Julgadora:

Resultado:

Prof. Titular **Glauco Augusto de Paula Caurin**
(Orientador)
(Escola de Engenharia de São Carlos/EESC)

APROVADO

Prof. Dr. **André Carmona Hernandes**
(Universidade Federal de São Carlos/UFSCar)

APROVADO


Profa. Dra. **Tatiana de Figueiredo Pereira Alves Taveira Pazelli**
(Universidade Federal de São Carlos /UFSCar)

APROVADO

Coordenador do Programa de Pós-Graduação em Engenharia
Mecânica:

Prof. Titular **Carlos De Marqui Junior**

Presidente da Comissão de Pós-Graduação:
Prof. Titular **Murilo Araujo Romero**

ENSCUEP
Secretaria de Pós-Graduação
Protocolo nº 05.09.2019


Aos meus pais, pelo amor, confiança e apoio incondicional. Ao meu irmão, pelo exemplo e sabedoria compartilhada, e a Camila, pelo amor, suporte e compreensão.

ACKNOWLEDGEMENTS

Agradeço ao orientador pela confiança e apoio, ao Prof. André Carmona, e aos amigos de laboratório pela ajuda e troca de experiências.

*“O estudo, a busca da verdade e da beleza são domínios
em que nos é consentido sermos crianças por toda a vida.”*

Albert Einstein

ABSTRACT

BUZZATTO, J. P. S. **Aerial Manipulation based on force and torque sensory feedback**. 2019. 85p. Dissertação (Mestrado) - Escola de Engenharia de São Carlos, Universidade de São Paulo, São Carlos, 2019.

The interest of the research community in studying Unmanned Aerial Vehicles (UAV) have increased greatly in the last decade. Due to advancements on sensors and batteries, those aerial robots became more accessible by the day. However, they are mostly destined to passive applications, such as surveillance and inspection, and to never physically interact with the environment. Aerial manipulation is a relatively new area that exploits the idea of using aerial robots for tasks that demands physical interaction. This work presents three scenarios in aerial manipulation from the control perspective, and two of them are studied in depth. For the first scenario, it is investigated the possibilities and challenges of aerial manipulation pick-and-placing tasks based on Force and Torque (F/T) sensory feedback. A framework is proposed based on the feedback compensation of internal and external efforts measured by a six-axis F/T sensor, placed between the UAV and a robotic arm attached below it. An attitude controller that compensates for the sensed torques and a novel position controller are presented. For the second scenario, a control framework is proposed in order for the aerial manipulator to compensate generalized 3D forces applied at its end-effector. In this case, the system considered is composed of a UAV and one Degree of Freedom actuated rod. The frameworks capabilities are evaluated on simulations done with the MuJoCo physics engine. The proposed system could be useful on situations and tasks that are too risky for humans, such as working in high altitudes, or in hazardous ambients, as in nuclear power plants.

Keywords: Aerial Manipulation. Quadrotors. UAVs. Force Sensor. Torque Sensor.

RESUMO

BUZZATTO, J. P. S. **Manipulação Aérea Baseada em Feedback de Força e Torque**. 2019. 85p. Dissertação (Mestrado) - Escola de Engenharia de São Carlos, Universidade de São Paulo, São Carlos, 2019.

O interesse da comunidade científica em estudar Veículos Aéreos não Tripulados ou *Unmanned Aerial Vehicles* (UAV) aumentou muito na última década. Como consequência dos avanços em sensores e baterias, esses robôs aéreos ficam cada vez mais acessíveis. Entretanto, eles são, em sua maioria, destinados a aplicações passivas, como vigilância e inspeções, e nunca interagindo fisicamente com o ambiente. Manipulação Aérea é uma área nova que explora a ideia de usar robôs aéreos para tarefas que demandam interação física.

Neste trabalho são apresentados três cenários em Manipulação Aérea dados pela perspectiva de controle, dos quais os dois primeiros são estudados aqui. Para o primeiro cenário são investigadas e estudadas as possibilidades e desafios de tarefas de *pick-and-placing* em Manipulação Aérea baseada no *feedback* de sensores de força e torque. É proposto um *framework* baseado na compensação de esforços internos e externos medidos por um sensor de força de seis eixos, posicionado entre o UAV e um braço robótico fixado embaixo deste.

São apresentados um controle de atitude que compensa os torques medidos e um novo controle de posição. Para o segundo cenário um *framework* de controle é proposto para que um manipulador aéreo compense forças 3D generalizadas aplicadas no sua ponta de ferramenta. Neste caso, o sistema considerado é composto por um UAV e um braço robótico de um grau de liberdade. As capacidades dos *frameworks* propostos são avaliadas através de simulações feitas com o motor de física MuJoCo. O sistema proposto pode ser útil em situações e tarefas que são muito arriscadas para humanos, como trabalhos em altas altitudes ou em ambientes perigosos, como plantas de usinas nucleares.

Palavras-chave: Manipulação Aérea. Quadrorotores. UAVs. Sensores de Força. Sensores de Torque.

LIST OF FIGURES

Figure 1 – Aerial manipulation application examples.	25
Figure 2 – Number of publications per year on aerial manipulation, for the last nine years. Analysis from the Web of Science website for when searching with the words "Aerial Manipulation", for the Robotics, Engineering Electrical Electronic, Computer Science Artificial Intelligence and Automation Control Systems categories.	26
Figure 3 – Images form seminal works on aerial manipulation.	28
Figure 4 – Illustration of the CG's position of a quadrotor with a fixed rigid tool performing a screw-drive operation.	29
Figure 5 – Aerial manipulators with simple tools placed above their CG.	30
Figure 6 – Aerial manipulation approach with multi-tiltrotors UAV	30
Figure 7 – Aerial manipulation examples with cable suspension.	31
Figure 8 – Aerial manipulation with robotic manipulators.	33
Figure 9 – Decouple position control block diagram for an UAM.	34
Figure 10 – Couple position control block diagram for an UAM.	34
Figure 11 – Quadrotor modeling reference frame and forces.	35
Figure 12 – Reference frames used for modeling and control.	41
Figure 13 – Control block diagram for the quadrotor position controller.	43
Figure 14 – Graph of the transformation function for the angles of the position controller.	44
Figure 15 – Quadrotor model flying on the MuJoCo Physics Engine.	45
Figure 16 – Position controller response to a finite ramp input.	45
Figure 17 – Position controller response to a finite ramp input under the effect of Blade Flapping.	46
Figure 18 – Displaced mass on the quadrotor.	47
Figure 19 – Position controller error responses to a finite ramp input of high inclination with a 50 grams mass displaced 0.1, 0.2, and 0.3 m on the positive x axis, on the x-y plane of the quadrotor model.	47
Figure 20 – Position controller response to a finite ramp input of high inclination with a 50 grams mass displaced by 0.1, 0.2 and 0.3 m, from top to bottom, on the positive x axis, on the x-y plane of the quadrotor model.	48
Figure 21 – Illustration of quadrotor with a rod moving over time.	49
Figure 22 – Position controller response over time under several torque applications.	50
Figure 23 – Position controller response under several torque applications. Relation between applied torque and error in position.	50
Figure 24 – Asctec Pelican CAD model with approximated dimensions.	52

Figure 25 – Robotic arm model with dimensions	53
Figure 26 – Whole model on MuJoCo native visualizer. On the right image the model is set to be translucent, and there are indications of joints and inertia boxes.	54
Figure 27 – System of reference frames and indication of the robotic arms degrees of freedom (blue arrows).	55
Figure 28 – Control block diagram of the aerial manipulator system.	56
Figure 29 – Trajectory with trapezoidal velocity profile.	58
Figure 30 – Time lapse of the simulated task.	59
Figure 31 – Attitude error for the attitude controller as defined by (4.2).	60
Figure 32 – Position of the quadrotor. Dashed black line is the desired position and solid red line is the actual position.	61
Figure 33 – Position of the systems End Effector. Dashed black line is the desired position and solid red line is the actual position.	61
Figure 34 – Torques measured by the simulated six axis F/T sensor.	62
Figure 35 – Forces measured by the simulated six axis F/T sensor.	62
Figure 36 – Cups and it's respective weights in grams.	63
Figure 37 – Time lapse of the cups piling task.	64
Figure 38 – Aerial Manipulator model and frames of reference.	66
Figure 39 – Illustration of the Ω plane, formed by $\hat{b}_x - \hat{b}_z$, to which the articulated arm is confined.	70
Figure 40 – Geometrical relationship between α , the tilting angle of ${}_B\vec{F}_{e\parallel\Omega}$ with respect to \hat{b}_x and γ , the angle the rod makes with respect to its resting position, i. e., parallel to \hat{b}_x	71
Figure 41 – Illustration of \hat{b}_{xd} and \hat{b}_{xdd}	72
Figure 42 – Combination of solutions for static sum of forces. a) Stable configuration for <i>pushing</i> b) Unstable configuration for <i>pulling</i> c) Unstable configuration for <i>pushing</i> d) Stable configuration for <i>pulling</i>	74
Figure 43 – Set of forces on the plane Γ	75
Figure 44 – Simulations showing <i>pushing</i> on the left and <i>pulling</i> on the right.	75
Figure 45 – Attitude error for direction y across time, for several applied forces on direction x	76
Figure 46 – Simulation of a <i>pushing</i> case that goes unstable.	77
Figure 47 – Simulations of a extreme <i>pushing</i> case with integral term on attitude controller, where the direction of the applied force is rotated 360 degrees in a 3 second time interval.	78

LIST OF TABLES

Table 1 – Inertial parameters for the Pelican quadrotor model	52
---	----

LIST OF ABBREVIATIONS AND ACRONYMS

3D	Three Dimensions
API	Application Programming Interface
CAD	Computer Aided Design
CG	Center of Gravity
DLL	Dynamic-Link Library
DoF	Degrees of Freedom
EEF	End-Effector
F/T	Force and Torque
IDE	Integrated Development Environment
LCP	Linear Complementarity Problems
ODE	Open Dynamics Engine
PID	Proportional Integral Derivative
UAV	Unmanned Aerial Vehicle
UAM	Unmanned Aerial Manipulator
XML	Extensible Markup Language

LIST OF SYMBOLS

A	Rotor disk area
C_Q	Non-dimensionalised drag coefficient
C_T	Non-dimensionalised thrust coefficient
M	Moment or Torque
\mathbf{R}	Rotation matrix of a body
\mathbf{R}_{ZYX}	ZYX Euler angles rotation matrix
$SO(3)$	Special orthogonal group
a_{1_s}	Body fixed blade flapping longitudinal tilt angle
b_{1_s}	Body fixed blade flapping lateral tilt angle
c	Cosine
c_q	Rotor reaction torque coefficient
c_t	Rotor thrust coefficient
d_D	Rotor distance from geometrical center
e	Error
f	Rotor force
r	Rotor radius
s	Sine
t	time
u_{1_s}	Body fixed blade flapping longitudinal tilt angle
v	Velocity of a body
x	Position
\times	The cross product
\vee	The <i>vee map</i> operator

Greek symbols

Θ	Angular position of a joint
Θ_t	Blade collective pitch angle
ω	Angular velocity of a joint or body
ρ	Air density
ν_{1s}	Body fixed blade flapping lateral tilt angle

Subscripts

s	Indication of sensed variable
d	Indication of desired
i	Indication of integral component of error
j	Variable index
p	Indication of proportional component of error
v	Indication of derivative component of error
$*_x *_y *_z$	Indication of axis of a reference frame

CONTENTS

1	INTRODUCTION	25
2	REVIEW OF LITERATURE	27
2.1	Aerial Manipulation	27
2.1.1	Mechanical Construction of UAMs	27
2.1.1.1	First approach: Single DoF gripper or tool	28
2.1.1.2	Second approach: Cable suspended	31
2.1.1.3	The third approach: Robotic manipulator	31
2.1.2	Control of UAMs	33
2.1.2.1	Position Control	33
2.2	Modeling of Quadrotor Aerodynamic Forces	35
2.2.1	Blade flapping	35
2.2.2	Rotor damping	36
3	METHODOLOGY	37
3.1	Simulation Environment: MuJoCo Physics Engine	38
3.1.1	Theoretical aspects of MuJoCo	38
3.1.2	Technical aspects of MuJoCo	39
4	SCENARIO ZERO	41
4.1	Attitude Controller	41
4.2	Position Controller	43
4.3	Pure Position Controller Responses	44
4.3.1	Pure Position Control	44
4.3.2	Position Control with Blade Flapping	45
4.4	Position Control with Static CG displacement	46
4.4.1	Discrete CG displacement	46
4.4.2	Position Control and External Applied Torque	48
5	SCENARIO ONE: PICK-AND-PLACING WITH F/T SENSING	51
5.1	Our Aerial Manipulator Simulation Model	52
5.1.1	The Quadrotor	52
5.1.2	The Robotic Arm	53
5.2	Control	55
5.2.1	Attitude Controller	56
5.2.2	Arm Controller	57
5.2.3	Arm Actuators	57



5.2.4	Trajectories	57
5.3	Simulation Results and Discussions	58
5.3.1	Pouring Task	58
5.3.1.1	Results and Discussion	59
5.3.2	Cups Piling Task	63
6	SCENARIO TWO (PARTIAL): COMPENSATION OF 3D FORCES	
	WHILE IN HOVER	65
6.1	Modeling	67
6.1.1	Representation and Main Assumptions	67
6.1.2	Equations of Motion	67
6.2	Control	68
6.2.1	Controlling the Arm	69
6.2.2	Defining the Desired Attitude	71
6.3	Geometrical Stability Conditions	72
6.4	Simulation Results and Discussions	73
7	CONCLUSION	79
7.1	Future Works	80
	BIBLIOGRAPHY	81



1 INTRODUCTION

In recent years became noticeable that research activity on Unmanned Aerial Vehicles (UAVs) increased considerably (Ruggiero; Lippiello; Ollero, 2018). Becoming more accessible by the day, mostly due to advancements on sensors and batteries, those aerial robots can be found in a great range of sizes and shapes, with quadrotors being amongst the most popular ones. However, most of the applications to which they are designated to being passive, such as surveillance and inspection, never physically interacting with the environment. Breaching through this limitation, Aerial Manipulation exploits the idea of using aerial robots for contact demanding tasks.

Figure 1: Aerial manipulation application examples.

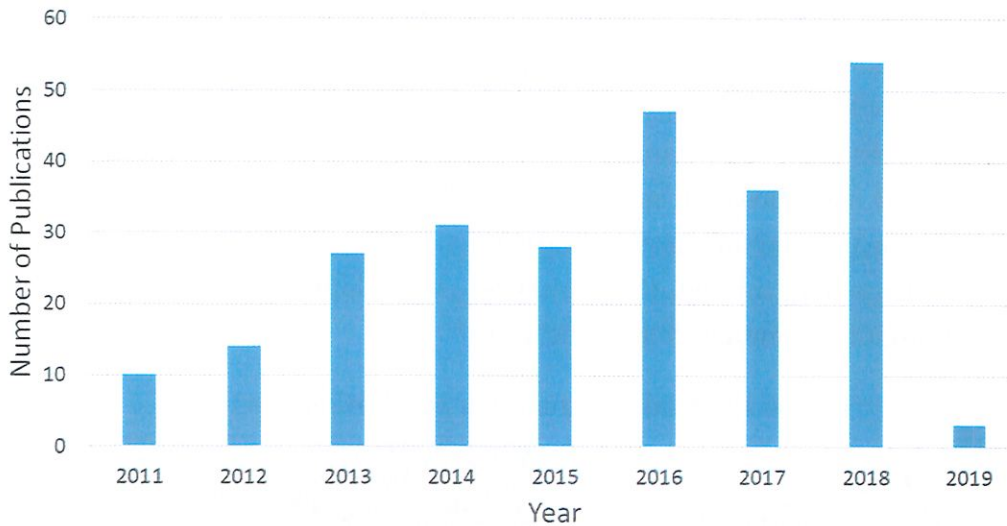


Source: a) Bridge maintenance (From: <http://www.bridgeriggers.com/>):
 b) Power line cleaning (From: <https://www.dailymercury.com.au/news/low-flying-helicopters-will-clean-high-voltage-pow/2378901/>):

From the extension of capabilities inside the same traditional surveillance and inspections applications to new ones such as maintenance and repair of structures that are hard to reach, the applications of aerial manipulations are very diverse. Imagine, for example, a large bridge located high above ground or water, that needs constant inspections (Figure 1 a)). Or the maintenance of power lines that works at high voltages (Figure 1 b)). Those kinds of tasks pose many risks to human workers, requiring highly trained and expensive professionals. In fact, aerial manipulators have already proven to be capable of working on such dangerous places (RUGGIERO; LIPPIELLO; OLLERO, 2018).

This potential has motivated much research since the beginning of this decade, as

Figure 2: Number of publications per year on aerial manipulation, for the last nine years. Analysis from the Web of Science website for when searching with the words "Aerial Manipulation", for the Robotics, Engineering Electrical Electronic, Computer Science Artificial Intelligence and Automation Control Systems categories.



(WOS, 2019).

shown in Figure 2. Several of those have been financed by big projects, such as ARCAS (ARCAS, 2018), AIRobots (AIROBOTS, 2018) and AEROWORKS (AEROWORKS, 2018). This demonstrates great interest and effort of the academic community in developing this area. As a matter of fact, the European Robotics Strategic Research Agenda (eSRA) have manifested that space and aerial robots technology shall evolve to a point where it is employed as robotic workers and co-workers, logistic robots, and robots for exploration and inspection (Ruggiero; Lippiello; Ollero, 2018; EUROPE, 2013).

So far, developments on the area have been diverse. There are researches focusing on vision, autonomous navigations for aerial manipulation, control, and many developing new robots designs. This work focuses more on the control aspect, addressing two critical problems in aerial manipulation, which are pick-and-placing and compensation of generalized 3D forces while hovering. Each of those scenarios will be explained better in the following chapters.

2 REVIEW OF LITERATURE

2.1 Aerial Manipulation

The first recognized papers in Aerial Manipulation were published around the year of 2011 (MELLINGER et al., 2011; POUNDS; BERSAK; DOLLAR, 2011; MICHAEL; FINK; KUMAR, 2011; FINK et al., 2011; LINDSEY; MELLINGER; KUMAR, 2011). These seminal works used traditional UAVs, attaching single DoF gripper below it or were based on tethers attached from the payload to one or multiple UAVs (KHAMSEH; JANABI-SHARIFI; ABDESSAMEUD, 2018). Even though those constructional additions were simple, they extended the capabilities of UAVs, allowing them to do pick-and-place and transportation tasks. More than that, it showed that aerial vehicles could be used for more than passive tasks, laying the ground and inspiring many future works.

Khamseh, Janabi-Sharifi and Abdessameud (2018) defined an Unmanned Aerial Manipulator (UAM) system as characterized by two main subsystems, namely the UAV, which is the flying platform containing the propulsion system, and the manipulation or interaction mechanism. The later ranges from sophisticated multi-DoF robotic arms to simple rigid tools. Plenty of research addressed the above two subsystems separately (GOERZEN; KONG; METTLER, 2009; Hua et al., 2013; Bicchi; Kumar, 2000), however, it is but very recently that the collective work on Aerial Manipulation yielded enough material for surveys on the area to take place (KHAMSEH; JANABI-SHARIFI; ABDESSAMEUD, 2018; DING et al., 2019; Ruggiero; Lippiello; Ollero, 2018).

Literature analysis indicates that quadrotors are by far the UAV platforms most used on research developments in aerial manipulation (KHAMSEH; JANABI-SHARIFI; ABDESSAMEUD, 2018). This is mostly due to quadrotors' simple mechanical structure, low-cost, hovering and maneuverability capacities, and its widespread popularity on other research areas and the industry (Bouabdallah; Becker; Siegwart, 2007). Given such information and the fact that UAVs platforms are a research field of its own, with plenty of published material, this work adopted quadrotors as its UAV platform and focuses mainly on the literature concerned with the manipulation/interaction aspect of UAMs.

2.1.1 Mechanical Construction of UAMs

There are a couple of suggested classifications for UAMs with regard to its mechanical construction (KHAMSEH; JANABI-SHARIFI; ABDESSAMEUD, 2018; DING et al., 2019; FANNI; KHALIFA, 2017). Even though each category has its own limitations and advantages, one problem is true for all of UAMs, which is payload capacity. Most manipulation mechanisms added to a UAV consumes a good part of the payload capability

originally designed for it, therefore restraining the range of objects a UAM can manipulate. That is a key limitation in aerial manipulation and it is argued that it will be a technological issue continuously addressed by the research community in the near future (KHAMSEH; JANABI-SHARIFI; ABDESSAMEUD, 2018). Given this omnipresent limitation, on the following subsections, we are going to explore the characteristics of UAMs' mechanical construction individually, according to the division proposed by Fanni and Khalifa (2017), which suggests three main approaches.

2.1.1.1 First approach: Single DoF gripper or tool

In the first approach, a simple gripper or a tool is installed at the bottom or on top of a UAV to transport a payload or interact with ambient structures. It can be used to pick and place objects, for construction, delivery, inspection or to exert forces on surfaces (FANNI; KHALIFA, 2017).

Mellinger et al. (2011) considered the payload as a big challenge for aerial manipulation, in the sense that it imposes many limitations with regard to the objects it can pick. In their approach, they were the first to consider explicitly on their control the center of mass of the system (UAV plus object) out of the UAV geometric center. They used a linear least-squares method for estimations of unknown parameters, such as the position of the center of mass and mass of the grasped object, allowing the payload to be unknown. This adaptation made the PID position and attitude controller much better at trajectory tracking.

Another pioneering work used a small helicopter (Figure 3) equipped with a single DoF gripper (POUNDS; BERSAK; DOLLAR, 2011). They reported that the main problems arise in the transition to free flight when the aircraft grab the object and is lifting it from the ground. Also, approach and alignment with the object was a challenge. Some of the problems mentioned were solved by using a highly adaptive compliant material for the gripper.

Figure 3: Images form seminal works on aerial manipulation.

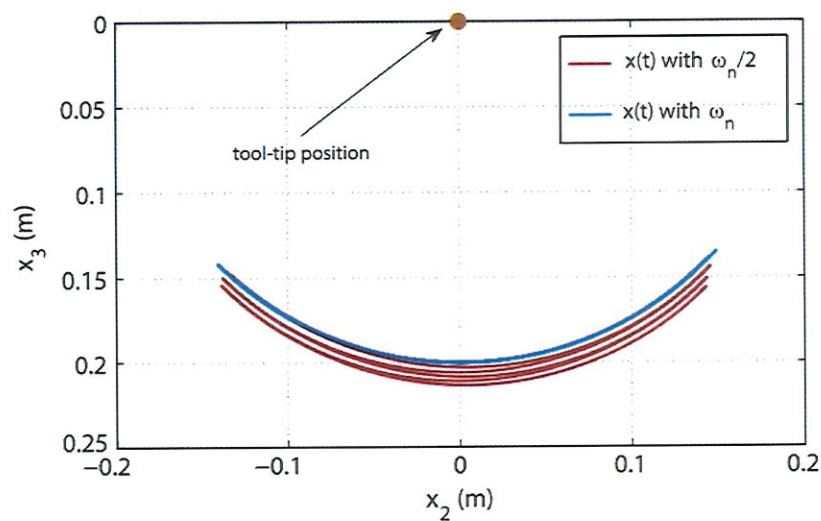


Left: (MELLINGER et al., 2011) Right: (POUNDS; BERSAK; DOLLAR, 2011)

Nguyen, Ha and Lee (2015) developed a framework that explores quadrotors as

tool operation devices, that can act as the actuator of a screwdriver, for example (Figure 4). With a very theoretical approach, they showed that any Cartesian control can be generated at the tool-tip, if and only if, the tool-tip is strictly located above or below the quadrotor's center of mass. Then they fully characterize the internal dynamics of the proposed system, which arises due to a quadrotor being an underactuated robot. Next, they showed that a necessary condition for internal stability is that the tool-tip must be located above the quadrotor's CG. Their proposed framework is evaluated on simulations.

Figure 4: Illustration of the CG's position of a quadrotor with a fixed rigid tool performing a screw-drive operation.

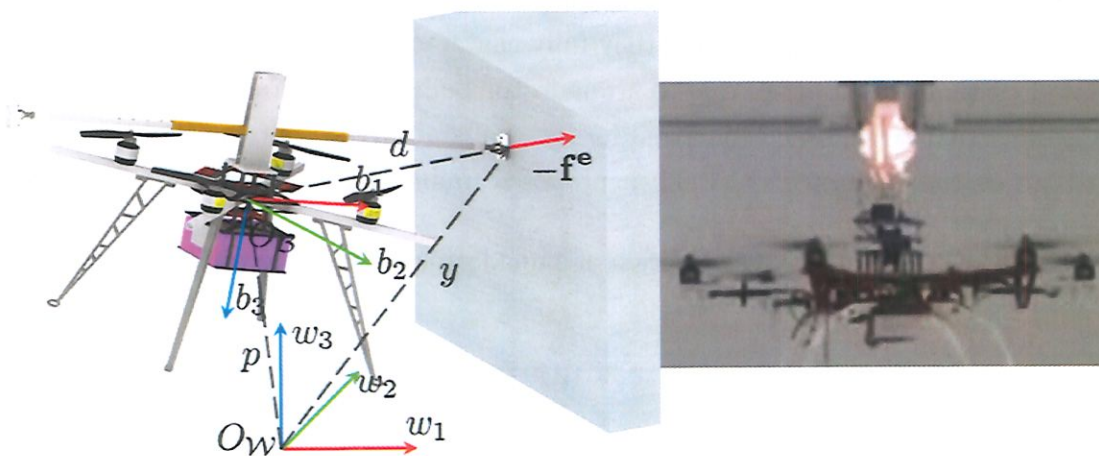


Source: (NGUYEN; HA; LEE, 2015)

In agreement with what is presented on the work mentioned above, some other researches on the literature used simple tools or grippers placed above the CG of the quadrotor. Gioioso et al. (2014) developed a theoretical design which allows a quadrotor equipped with a rigid tool to exert arbitrary 3D forces (Figure 5). This is a similar proposed goal to that of Nguyen, Ha and Lee (2015), but a different approach, both developed roughly at the same time. Shimahara et al. (2015) and Shimahara et al. (2016) explored a design where a UAV was equipped with a gripper attached atop it. Through visual control, the system was used to grab and released a bar and to successfully unscrew a light bulb (Figure 5).

Another strategy exploit for aerial manipulation is to create new UAV models that are not underactuated, and therefore are capable of directly controlling its position and exerting forces on all three directions. Papachristos, Alexis and Tzes (2014) developed a *tiltrotor* UAV and a methodology to achieve controlled forward thrust force and rotating torque exertion while maintaining safe operation for near hovering attitude pose, that

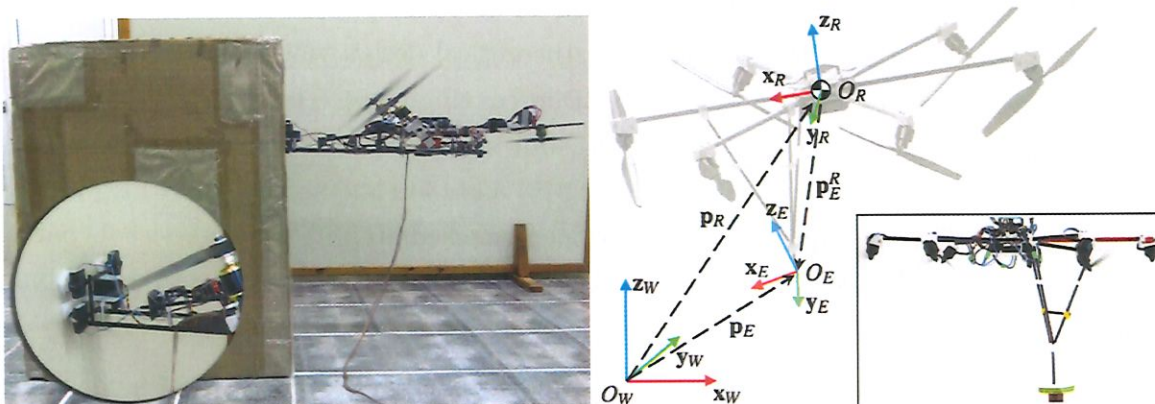
Figure 5: Aerial manipulators with simple tools placed above their CG.



Left: (GIOIOSO et al., 2014) Right: (SHIMAHARA et al., 2016)

was verified experimentally. Ryll et al. (2017) designed a new tilted-propeller hexarotor, aimed for physical interaction tasks. They verified with experiments that their design was able to fully control its pose (orientation and position independently) with geometric control. Additionally, they reported that the robot could exert full-wrench (torque and force independently) with a rigid EEF and using admittance control. Both designs are illustrated on Figure 6

Figure 6: Aerial manipulation approach with multi-tiltrotors UAV



Left: (PAPACHRISTOS; ALEXIS; TZES, 2014) Right: (RYLL et al., 2017)

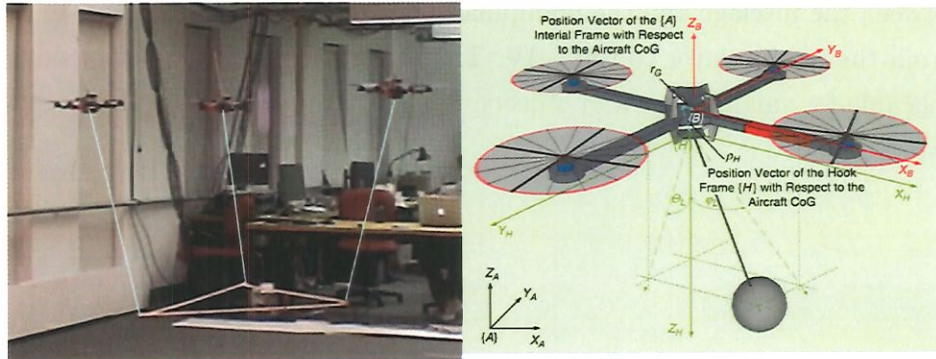


A downside of the first approach is that the attitude of the payload or tool is constrained to the attitude of the UAV, and the accessibility of the end-effector is confined, fixed with respect to the UAV. This configuration delivers a system with four DOF, three translational and one rotational (Yaw), meaning that the end-effector cannot pitch or roll without moving the whole system horizontally (FANNI; KHALIFA, 2017).

2.1.1.2 Second approach: Cable suspended

In the second approach, the payload is suspended with cables (FANNI; KHALIFA, 2017). Much research effort on this approach focuses on aerial manipulation with multiple aerial robots. On this version of the approach, all UAVs are connected by cable to the payload, and the control is developed so to pose it as desired (JIANG; KUMAR, 2013; FINK et al., 2011; GOODARZI; LEE, 2015). The other strategy found is transportation with only one UAV, where the controllers are mostly designed so to avoid swinging of the payload (BISGAARD; COUR-HARBO; BENDTSEN, 2010; PALUNKO; CRUZ; FIERRO, 2012). An illustration of this approach is shown in Figure 7. The drawback of the second approach is that it is not always possible to control payload motion as wanted, limiting its uses (FANNI; KHALIFA, 2017).

Figure 7: Aerial manipulation examples with cable suspension.



Left: (JIANG; KUMAR, 2013) Right: (PALUNKO; CRUZ; FIERRO, 2012)

2.1.1.3 The third approach: Robotic manipulator

The third approach consists of equipping a UAV with a robotic manipulator that can actively interact with the environment. It was developed to overcome the limitations of the first two approaches. By combining the mobility and versatility of a UAV with the capacities of a robotic manipulator, the utility of the UAM is greatly increased. When a robotic manipulator is employed in a UAV for aerial manipulation, the dynamics of it became highly coupled with that of the UAV. Also, reaction forces from the interactions with objects or external environment may play an important role in system stability and

performance. These effects should be carefully considered in the design of the controller. Very few reports in the literature investigate the combination of UAV with robotic manipulator (FANNI; KHALIFA, 2017). Even so, such an approach is recently being considered as the most relevant one, and chronological analysis of the literature suggest that it is becoming more common (KHAMSEH; JANABI-SHARIFI; ABDESSAMEUD, 2018).

In (KIM; CHOI; KIM, 2013), a quadrotor with a two DoF robot arm is controlled by an adaptive sliding mode controller, and successfully picked up a wood block and placed it inside a shelf, exploring the geometry of the manipulator. In (LIPPIELLO; RUGGIERO, 2012), the authors developed Cartesian-based impedance control, testing in simulation for a manipulator with two links attached to a quadrotor. The authors in (JIMENEZ-CANO et al., 2013) compared a Variable Parameter Integral Backstepping controller with traditional PID controllers, using a large quadrotor with a 3-link manipulator. In (KIM; SEO; KIM, 2015), the authors developed a framework capable of exerting forces to open and close an unknown drawer. In (FANNI; KHALIFA, 2017), a system composed by a quadrotor and a 2-DoF manipulator with unique topology was presented. The authors demonstrated that the proposed manipulator has the minimum number of DoF needed to perform arbitrary 6-DoF trajectories. Kondak et al. (2014) studied a system composed of a helicopter endowed with a 7 DoF industrial manipulator, where a 6 axis F/T sensor was placed between the fuselage and the manipulator to compensate for transmitted forces and torques from the manipulator to the UAV. That is a design fit for outdoor environments, but inapplicable for small rooms and objects. Some of the presented designs are illustrated in Figure 8.

Figure 8: Aerial manipulation with robotic manipulators.



Top left: (HEREDIA et al., 2014) Top right: (KHALIFA; FANNI, 2017) Bottom left: (KONDAK et al., 2014) Bottom right: (KIM; CHOI; KIM, 2013)

2.1.2 Control of UAMs

In this subsection, we briefly discuss the categorization of control approaches for UAMs as presented by Khamseh, Janabi-Sharifi and Abdessameud (2018).

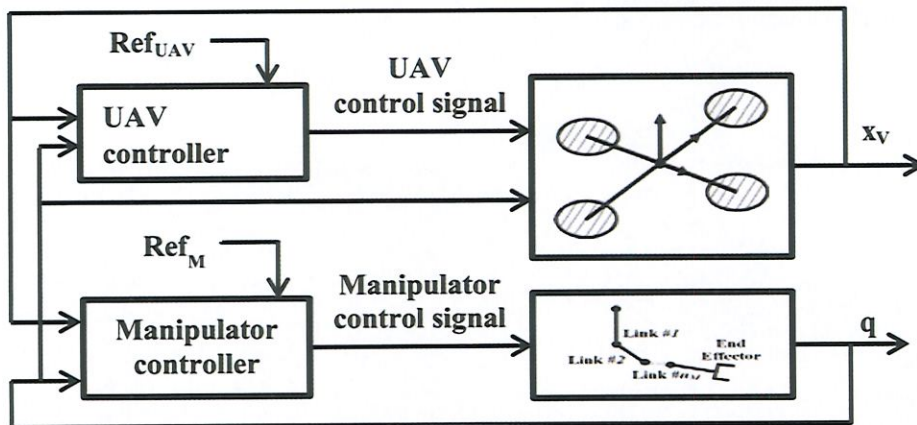
2.1.2.1 Position Control

The position control for UAMs can be divided into two categories which are *couple* and *decouple* and it is said that they are dependent on the modeling strategy used. For instance, there have been observed two modeling approaches for UAMs, which are the Recursive Newton-Euler method and the Euler-Lagrange method.

The *decouple* control approach is the simplest one, and usually is based on the Newton-Euler method. In such a control strategy, the UAV and the manipulator are treated as two decouple systems, each one with its own controller. Generally speaking, in *decouple* position control the UAV controller has the basic role of positioning the whole system as desired and it is designed in a way so to compensate for the movements of the robotic manipulator attached to it. This approach, therefore, assumes that the robotic manipulator is not concerned with the UAV's dynamics or the dynamics of the whole

system. Each system receives its own desired reference states, as shown in Figure 9. This control strategy is very simple, but usually, it leads to low performance for more complex scenarios when there are large changes in inertial parameters of the system and its CG location (KHAMSEH; JANABI-SHARIFI; ABDESSAMEUD, 2018).

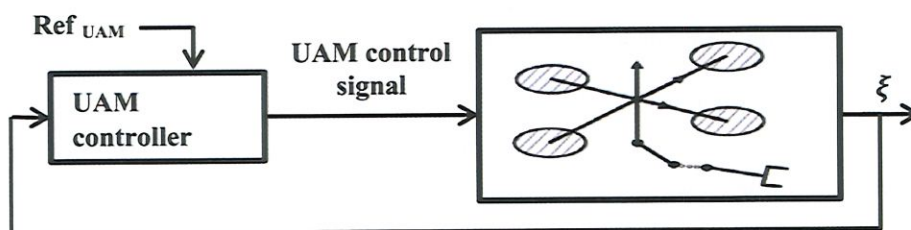
Figure 9: Decouple position control block diagram for an UAM.



(KHAMSEH; JANABI-SHARIFI; ABDESSAMEUD, 2018)

For the *couple* position control strategy, the model base is usually obtained by the Euler-Lagrange formalism. In this approach, the UAV and robotic manipulator are treated as one system, receiving just one desired reference state, as indicated in Figure 10. The dynamic model of the combined UAM system usually takes a nonlinear and usually more complex form when compared to the decouple dynamic model of each subsystem. Therefore, a model based unified control design in the *couple* approach tends to be theoretically difficult. Nevertheless, it is argued that the couple control yield better performance with more accurate position control. For those reasons, it is expected that in the near future the couple approach will receive greater attention from the research community (KHAMSEH; JANABI-SHARIFI; ABDESSAMEUD, 2018).

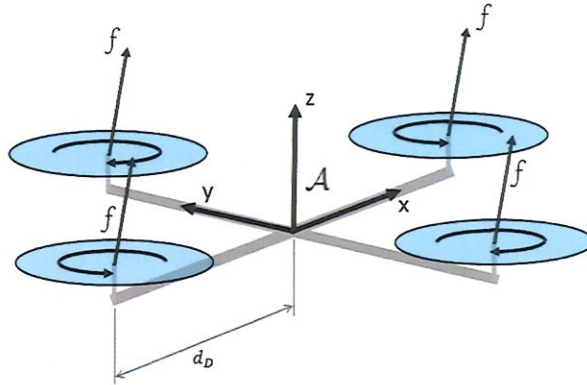
Figure 10: Couple position control block diagram for an UAM.



(KHAMSEH; JANABI-SHARIFI; ABDESSAMEUD, 2018)

2.2 Modeling of Quadrotor Aerodynamic Forces

Figure 11: Quadrotor modeling reference frame and forces.



Source: Author

In this work, the modeling of aerodynamic forces was mainly based on [Pounds, Mahony and Corke \(2010\)](#). Thrust and torque generated by each individual rotor of common multi-rotor UAVs can be expressed as in (2.1) and (2.2), respectively. These forces act along the \mathcal{A}_z direction, positioned at the rotor spinning axis.

$$f_j = C_T \rho A r^2 \omega_j^2 \quad (2.1)$$

$$q_j = C_Q \rho A r^3 \omega_j |\omega_j| \quad (2.2)$$

The mapping from the rotor forces to the net forces acting on the quadrotor is represented by equation (2.3). The 4x4 matrix in (2.3) correlates the four thrusting forces generated by each rotor to the total thrusting force exerted on the quadrotor and the vector of torques acting on all axis, x, y, and z, of the \mathcal{A} frame.

$$\begin{bmatrix} f \\ M_x \\ M_y \\ M_z \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 0 & d_D & 0 & -d_D \\ -d_D & 0 & d_D & 0 \\ -c_q/c_t & c_q/c_t & -c_q/c_t & c_q/c_t \end{bmatrix} \begin{bmatrix} f_1 \\ f_2 \\ f_3 \\ f_4 \end{bmatrix} \quad (2.3)$$

2.2.1 Blade flapping

Blade flapping appears when rotors translate, rotate or is faced with the horizontally coming wind. On those conditions, a variation in blade lift appears amidst advancing

and retreating blades. This causes the rotor plane to tilt, which generate forces on the other two directions perpendicular to the rotating axis of the rotor. This is important considering that such tilting of blades can inject notable stability effects on the UAV (POUNDS; MAHONY; CORKE, 2006).

Unfortunately, most of the parameters used to calculate blade flapping are dependent on blade geometry and can only be estimated experimentally. To obtain these data is out of the scope of this work, and also, they are not freely available. For those reasons, here the values presented on Pounds, Mahony and Corke (2010) was used with the only purpose of approximately evaluate the effects of such phenomenon on our system. When blade flapping is considered, the thrust model transforms into (2.4).

$$f_j = C_T \rho A r^2 \omega_j^2 \begin{bmatrix} \sin(a_{1sj}) \\ -\cos(a_{1sj}) \sin(b_{1sj}) \\ \cos(b_{1sj}) \sin(a_{1sj}) \end{bmatrix} \quad (2.4)$$

2.2.2 Rotor damping

When quadrotors pitches and rolls, the rotors undergo vertical velocity, changing the flow through the propellers and causing a variation in the inflow angle (POUNDS; MAHONY; CORKE, 2010). This impacts the thrust coefficient C_T , which can be associated with the vertical velocity, V_c , as in (2.5).

$$\frac{C_T}{\sigma} = \frac{a(\alpha)}{4} \left(\theta_{tip} - \frac{v_i + V_c}{\omega r} \right) \quad (2.5)$$

with θ_{tip} being the blade angle at the rotor's tip, a is the polar lift inclination of the airfoil, v_i the velocity induced over the rotor, and σ is the disc's solidity, which is the surface area of the blades over the rotor disc area. However, because the polar lift slope is a function of the blade angle of attack, α , which is nonlinear for some models of propellers. The effect of rotor damping can be simplified if expressed as

$$C_{Tj} = C_{T0} + \Delta C_{Tj} \quad (2.6)$$

where C_{T0} is the thrust coefficient when stably hovering and ΔC_{Tj} is the change due to inflow changes and can be calculated as in (2.7).

$$\Delta C_{Tj} = -\frac{a_0}{4} \frac{\sigma}{\omega_j r} \left(\vec{v} + \vec{\omega} \times \vec{d}_j \right) \quad (2.7)$$

where a_0 is the lift slope for stable hovering.

3 METHODOLOGY

Here, we propose that the third approach of Aerial Manipulation have a total of three scenarios it can face, from the control perspective:

1. Scenario One: Pick-and-placing
2. Scenario Two: Compensation of 6D efforts while in hover
3. Scenario Three: Force and torque exertion with its movement constrained.

It is also probably true that all of the three scenarios can be viewed as of particular cases of a more generic situation, where the UAV is modeled with soft constraints of variable impedance. However, for the sake of control development, here we adopt the above division and address the first and partially the second scenarios of Aerial Manipulation using a quadrotor as fling platform. In order to do that, first, we studied a more basic scenario that we are going to call Scenario Zero, which is simply a hovering UAV with a displaced center of gravity.

Consider then an Aerial Manipulator system of the third approach, composed of two main subsystems: The UAV platform and a robotic arm. As mentioned in Chapter 2, this approach is considered by some authors the most promising in the area (FANNI; KHALIFA, 2017; KHAMSEH; JANABI-SHARIFI; ABDESSAMEUD, 2018). On this approach, the simple movement of the robotic arm, regardless of where it is attached to the UAV, results in a Scenario Zero. Therefore, the first part of this work does a brief study on how changes of the CG's position affects a hovering quadrotor UAV subjected to a regular PD position controller.

With the insights provided in the first part, an approach to address the problem is proposed and implemented in the second part of this work, which deals with the pick-and-placing scenario. In such activities, the aerial manipulator tends to have short contact time with the environment plus the object picked up. Once the object is lifted off of its resting place, the aerial manipulator system plus the object can be viewed again as a hovering UAV platform with the CG displaced. Therefore, from that perspective, compensation of displacement of the CG is the main issue for pick-and-place tasks. In that sense, we also consider that a UAV with a CG displaced is equivalent to a UAV with forces and torques applied at its original symmetrically place CG. The approach we propose uses a six-axis Force and Torque (F/T) sensor to measure the applied net of efforts on the UAV's frame. Given such data, then it is easy to compensate for the displacements of the system's CG. With simulations, we found out that such an approach works very well for pick-and-placing tasks.

In the third part, we address Scenario Two partially. There, we consider a hovering UAV equipped with a single DoF robotic arm that is subjected to generalized 3D forces applied at its end-effector. Here it is relevant to say that all of the three scenarios have already been addressed previously in the literature. However, to the best of authors' knowledge, all of the presented solutions were limited in performance in terms of the Payload/System mass ratio and the magnitude of forces applied (for Scenario Three) or compensated (for Scenario Two) relative also to the system's weight, with very few exceptions ([WOPEREIS et al., 2017](#)). Both solutions presented on this work were evaluated with numerical simulations and proved to be efficient for relatively large efforts.

3.1 Simulation Environment: MuJoCo Physics Engine

Simulations are always a great tool for evaluating preliminary ideas. They are low cost, fast, easy to implement any changes and can be reliable. There are several options of physics engines that simulate rigid body dynamics. Some of the most popular ones are the Open Dynamics Engine (ODE) ([SMITH, 2009](#)) and Bullet ([COUMANS, 2015](#)), which are open source. Other famous proprietary ones are Nvidia PhysXTM ([PHYSX, 2018](#)) and Havok ([HAVOK, 2018](#)). In this work, the MuJoCo physics engine is used.

3.1.1 Theoretical aspects of MuJoCo

Many of the physics engines available are usually directed to Real-Time simulation in games, such as PhysX, ODE, and Bullet. They are not built aiming for accuracy, but instead, the goal is to achieve a physically convincing and stable simulation. Also, they commonly do the bodies state representation in overcomplete Cartesian coordinates and enforce joint constraints numerically. This is good for simulations with a large number of mostly disconnected bodies with few joints. However, this method of representation is inaccurate and inefficient for simulations with complex multi-joint models. Furthermore, game-oriented physics engines models contacts dynamics as spring-damper or Linear Complementarity Problems (LCPs), which are an improvement when compared to the first, but still demands manual tuning and small time steps ([TODOROV; EREZ; TASSA, 2012](#)).

MuJoCo stands for **M**ulti-**J**oint dynamics with **C**ontact. It is specially designed for simulating model-based control of robotic systems, representing states in joint coordinates and simulating contacts in a similar manner to LCP, but better. When compared to other physics engines, namely Bullet, Havok, ODE, and PhysX, MuJoCo proved to perform better on robotics-related tests, to which it was made for. Likewise, the other engines performed better on gaming-related tests, without a clear winner among them ([EREZ; TASSA; TODOROV, 2015](#)).

3.1.2 Technical aspects of MuJoCo

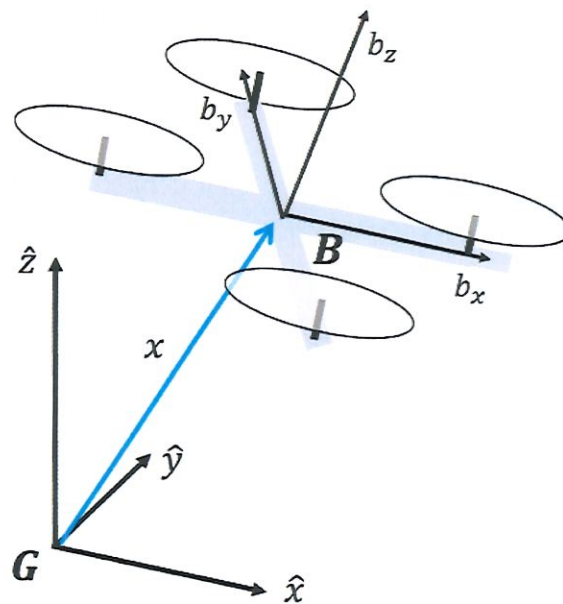
Currently, MuJoCo has a couple of flavors: MuJoCo Pro, HAPTIX, Unity Plugin and VR. The most complete and the only one under commercial license is MuJoCo Pro, which is available for several platforms, including Windows, Linux, and OSX (MUJOCO.ORG, 2018), and it is the flavor used in this work. On Windows, it is basically a Dynamic-Link Library (DLL), with a C/C++ API. Its simulation pipeline is made of functions that the user can call individually, facilitating the implementation of non-standard computations. Models are created in XML format files or using the C++ API, which is then compiled into low-level data structures that are optimized for runtime computation ([TODOROV; EREZ; TASSA, 2012](#)). In this work the programming was done using the Visual Studio Community IDE ([MICROSOFT, 2018](#)), running on Windows 10.

4 SCENARIO ZERO

To gain better insight into the behavior of a quadrotor aerial manipulator of the third approach, when such a system is under a PD position controller, we first consider the scenario zero studied here. As defined previously, the scenario zero consists of a UAV quadrotor subjected to a PD position controller with some unbalanced mass or with torques applied on its structural frame. Most position controllers for quadrotors are based on the more basic attitude controller. This is due to the underactuated nature of a quadrotor, which has six DoF but only four rotors to control its states. On the following sections, the attitude and position controller employed in this chapter are presented. After that, two sections follow analyzing the system in different situations. First, we present the response of the quadrotor alone with a position controller for a finite ramp input. Then, for the sake of completeness of the modeling, we analyze the effects of blade flapping on the position control. After that, the other section follows considering respectively the quadrotor with increasing unbalance by adding a mass at its CG horizontal plane and the quadrotor with a one DoF arm attached at its bottom.

4.1 Attitude Controller

Figure 12: Reference frames used for modeling and control.



Source: Author



The attitude controller presented herein is based the one in (LEE; LEOKY; MC-CLAMROCH, 2010). This is a geometric controller with nonlinear terms, that explicitly compensate for position and velocity, similarly to a PD controller. The system is fully represented with the inertial reference frame $\mathbf{G} = \{\hat{x}, \hat{y}, \hat{z}\}$ and reference frame $\mathbf{B} = \{\hat{b}_x, \hat{b}_y, \hat{b}_z\}$, attached to the quadrotors CoG. The ZYX Euler angles representation was chosen to generate rotation matrices, as in (4.1), to represent the frames orientation (Figure 12).

$$R_{ZYX} = \begin{bmatrix} c_y c_z & c_z s_x s_y - c_x s_z & c_x c_z s_y + s_x s_z \\ c_y s_z & c_x c_z + s_x s_y s_z & -c_z s_x + c_x s_y s_z \\ -s_y & c_y s_x & c_x c_y \end{bmatrix} \quad (4.1)$$

where c_j and s_j represent the cosine and sine of the rotation angle of the specified axis. Given a desired pose for the quadrotor, $\mathbf{R}_d(t) \in SO(3)$, the attitude error is defined as:

$$\vec{e}_R = \frac{1}{2}(\mathbf{R}_d^T \mathbf{R} - \mathbf{R}^T \mathbf{R}_d)^\vee \quad (4.2)$$

where \mathbf{R}_d and \mathbf{R} are expressed with respect to the inertial frame \mathbf{G} . The terms on the right-hand side of (4.2) wield a skew-symmetric matrix, which its non-zeros terms can be expressed as a vector $\vec{e}_R \in \mathcal{R}^3$ through the *vee map* $^\vee : SO(3) \rightarrow \mathcal{R}^3$ operator, as shown by (4.3).

$$\vec{e}_R = \begin{bmatrix} e_{Rx} \\ e_{Ry} \\ e_{Rz} \end{bmatrix} = \begin{bmatrix} 0 & -e_{Rz} & e_{Ry} \\ e_{Rz} & 0 & -e_{Rx} \\ -e_{Ry} & e_{Rx} & 0 \end{bmatrix}^\vee \quad (4.3)$$

The error for the angular velocities are then defined based on the other controller input, the desired angular velocity, $\vec{\omega}_d \in \mathcal{R}^3$:

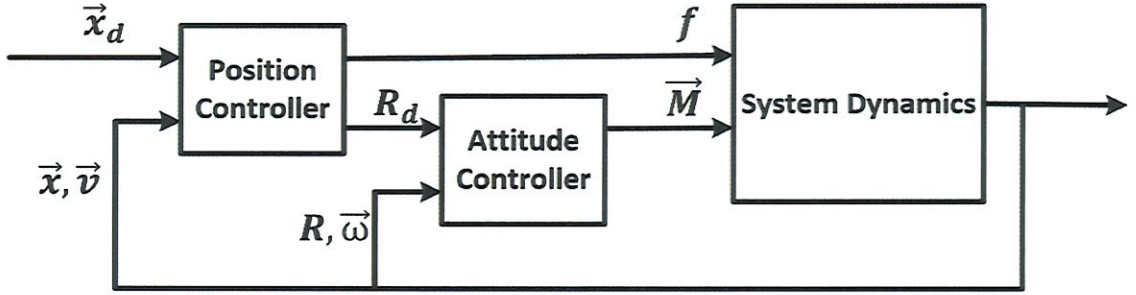
$$\vec{e}_\omega = \vec{\omega} - \mathbf{R}^T \mathbf{R}_d \vec{\omega}_d \quad (4.4)$$

with $\vec{\omega}$ and $\vec{\omega}_d$ being expressed with respect to the reference frame of the quadrotor, \mathbf{B} . Then, the control law (4.5) correlates the above defined errors, and other terms, to the torques M_x , M_y and M_z , also expressed in the \mathbf{B} base.

$$\vec{M} = -\mathbf{J}(\mathbf{K}_R \vec{e}_R + \mathbf{K}_\omega \vec{e}_\omega) + \vec{\omega} \times \mathbf{J} \vec{\omega} \quad (4.5)$$

where \mathbf{K}_R and \mathbf{K}_ω are gain matrices.

Figure 13: Control block diagram for the quadrotor position controller.



Source: Author

4.2 Position Controller

As mentioned, the position controller makes use of the attitude controller (see Figure 13) to control positions on \hat{x} and \hat{y} directions. However, controlling the position along the \hat{z} axis is quite straight forward, for all the rotors of the quadrotor are always acting on this direction when stably hovering. A simple PID was used to control position along the \hat{z} axis.

$$f = M_d g_z + \mathbf{K}_{pz} e_{pz} + \mathbf{K}_{vz} e_{vz} \quad (4.6)$$

where K_{pz} and K_{vz} are gain matrices and the errors \vec{e}_p and \vec{e}_v are defined as:

$$\vec{e}_p = \vec{x}_d - \vec{x} \quad (4.7)$$

$$\vec{e}_v = \vec{v}_d - \vec{v} \quad (4.8)$$

For the remaining directions, the controller simply calculates rotation angles around \hat{b}_y and \hat{b}_x axis, construct a rotation matrix as in (4.1) and send it as the desired attitude to the attitude controller. These angles are the output of the control law for directions \hat{y} and \hat{x} , which is a PD.

$$\beta_d = k_p \theta_{px} + k_v \theta_{vx} \quad (4.9)$$

$$\gamma_d = -k_p \theta_{py} - k_v \theta_{vy} \quad (4.10)$$

with k_p and k_v being the PD gains. β_d and γ_d are the desired attitude angle for the \hat{b}_y and \hat{b}_x axis, respectively. The variables θ_p and θ_v are a kind of transformation of the position errors to desired angles that are treated as the errors in a common PD controller. This transformation is done by using a hyperbolic tangent function, with gains to manipulate its shape. The transformations are defined in (4.11) and (4.12).

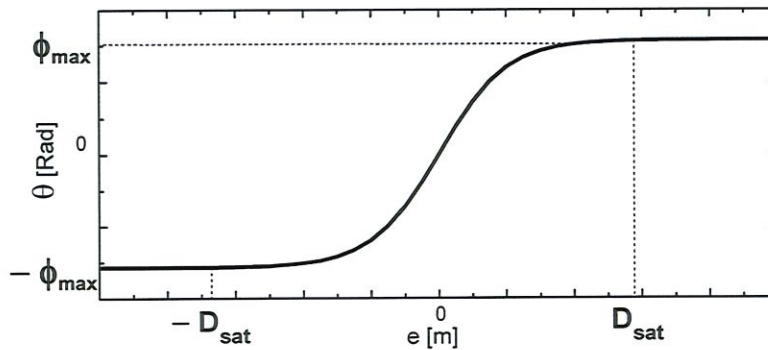
$$\theta_{pj} = \phi_{pmax} \tanh(k_{ph}e_{pj}) \quad (4.11)$$

$$\theta_{vj} = \phi_{vmax} \tanh(k_{vh}e_{vj}) \quad (4.12)$$

The first gain, ϕ_{max} , in each equation determines the maximum and minimum output angle, since the hyperbolic tangent only output values between +1 and -1. The second gain indicates the saturation distance, D_{sat} , calculated as in (4.13), and as illustrated in Figure 14. It is nice to notice that the position controller does not act on the rotation about the \hat{b}_z axis, making possible for the vehicle to move in any direction on the horizontal plane regardless of its yaw orientation.

$$k_h = \frac{2}{D_{sat}} \quad (4.13)$$

Figure 14: Graph of the transformation function for the angles of the position controller.



Source: Author

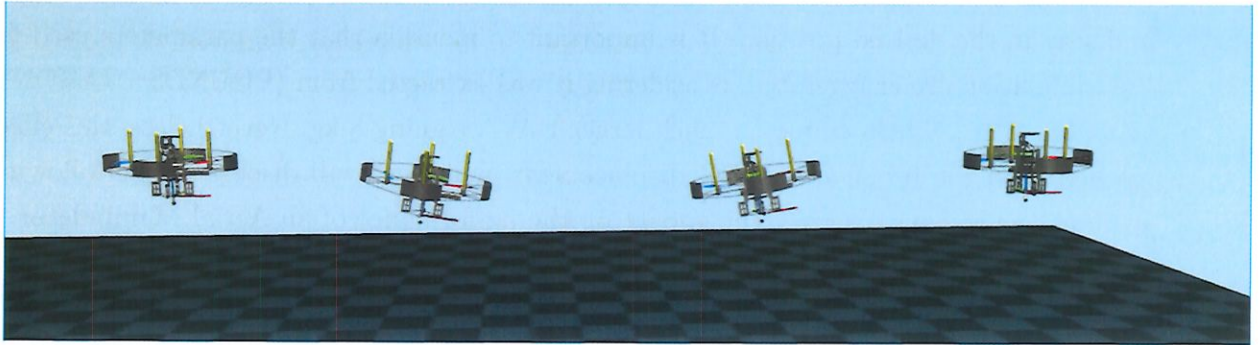
4.3 Pure Position Controller Responses

4.3.1 Pure Position Control

Figure 16 presents the numerical results of the following situation: At first, the quadrotor is stably hovering at the origin of the coordinate system. At time equal to one second the commanded desired position moves with a constant velocity of 1 m/s along the positive \hat{x} axis, for three seconds. In the response we observe that the quadrotor is

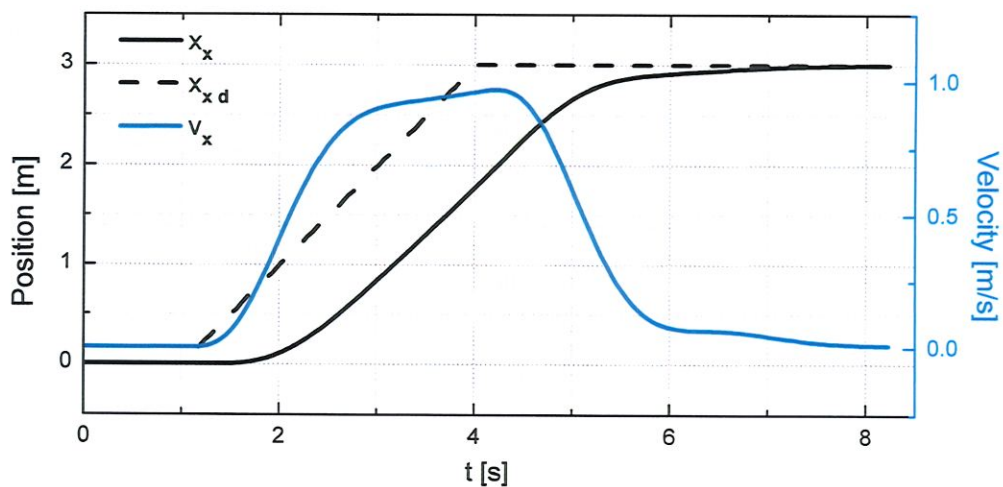
considerable slow to start changing its position, staying more or less one second behind the input signal. Nevertheless, one important characteristic of the response is that it has no static or regime error even under a PD control. Figure 15 shows screenshots of the simulated flying drone.

Figure 15: Quadrotor model flying on the MuJoCo Physics Engine.



Source: Author

Figure 16: Position controller response to a finite ramp input.



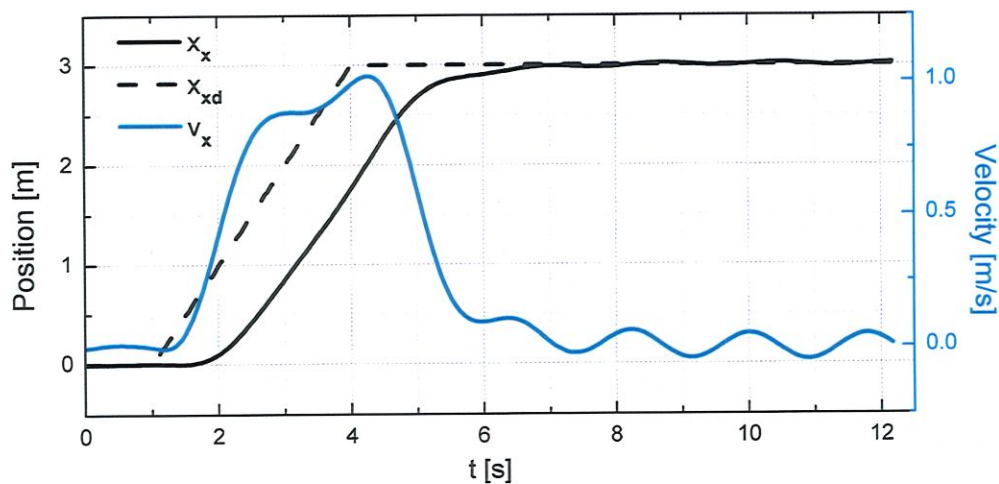
Source: Author

4.3.2 Position Control with Blade Flapping

The modeling that leads to the results presented above is used widely in the literature. However, some interesting effects such as blade flapping are not considered there. To investigate how such commonly unaccounted phenomena could influence aerial manipulation tasks, we simulated a situation considering blade flapping in our model, as presented in Subsection 2.2.1. On the simulation, presented in Figure 17, the situation

was the same as the one considered in the previous subsection but accounting for Blade Flapping. Corroborating with its model, Blade Flapping induces an oscillation in the horizontal plane, as we can see mainly in the velocity curve for when the system has already reached its desired position. Even though the image does not show very well the impact of this oscillation in position, such effect could, for example, affect an Aerial Manipulator in a pick-and-place task by making it more difficult for the EEF to reach and stay in the desired position. It is important to mention that the parameters used for this simulation are exaggerated, considering it was extracted from (POUNDS; MAHONY; CORKE, 2010), which employ a much larger UAV weighing 8 kg. Nevertheless, this effect aligned with the influences of CG displacement, which we will discuss in the following sections, could have a significant impact on the performance of an Aerial Manipulator.

Figure 17: Position controller response to a finite ramp input under the effect of Blade Flapping.



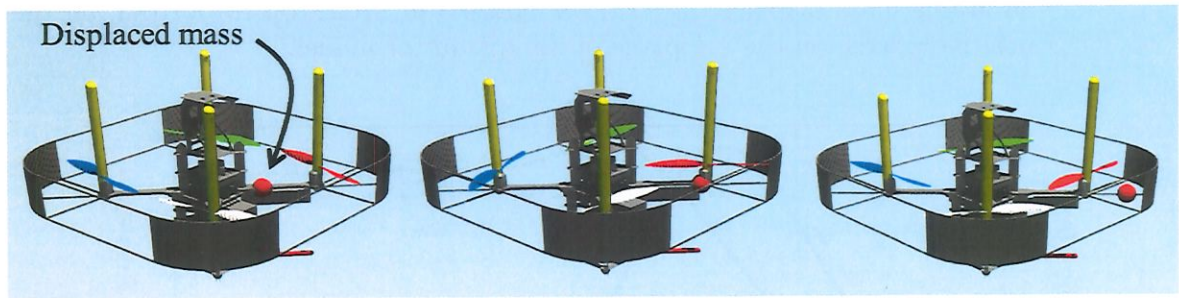
Source: Author

4.4 Position Control with Static CG displacement

4.4.1 Discrete CG displacement

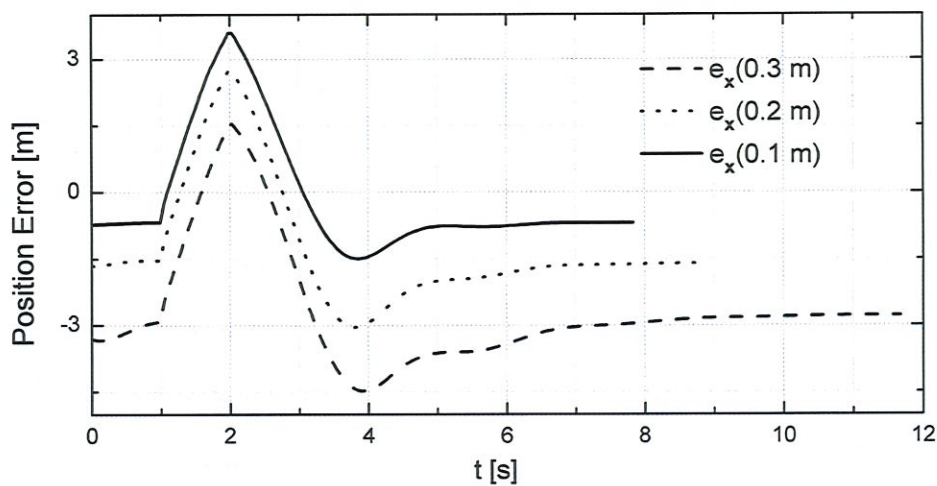
After having established a position controller and seen how it behaves normally, now we consider the effects of CG displacement on such position controller. In order to do that, we first consider the quadrotor model with an additional mass placed on its frame along the \hat{b}_x axis, as illustrated in Figure 18. When allocated away from the geometric center of the quadrotor, this mass causes an unbalance which shifts the CG in the same direction. The effects of such displacement are shown in Figure 20, where a mass of 50 g is placed at 0.1, 0.2 and 0.3 m from the UAV's geometric center.

Figure 18: Displaced mass on the quadrotor.



Source: Author

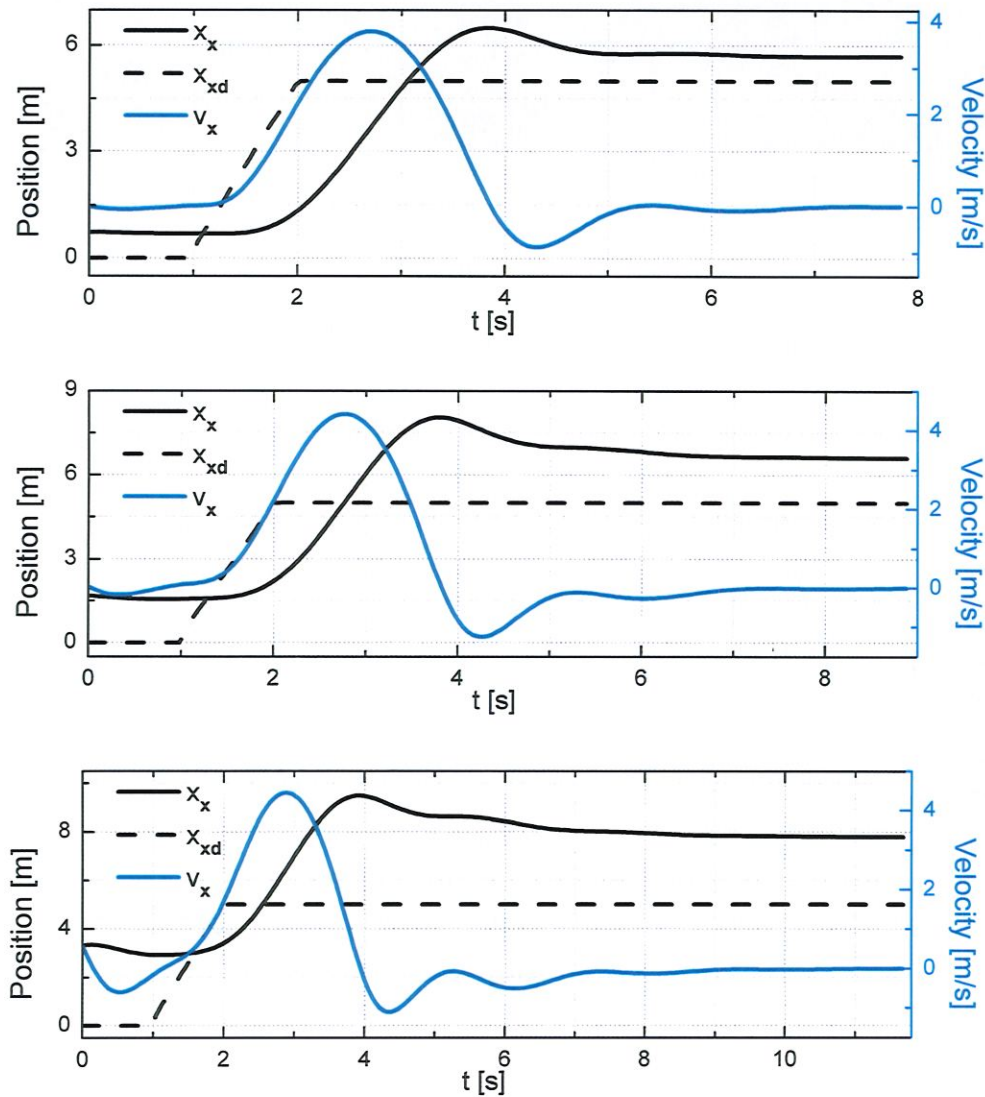
Figure 19: Position controller error responses to a finite ramp input of high inclination with a 50 grams mass displaced 0.1, 0.2, and 0.3 m on the positive x axis, on the x-y plane of the quadrotor model.



Source: Author

The evolution of the system over time when subjected to this discrete increment of unbalance reveals that the position controller now presents a regime error and that it increases with the unbalance. Nevertheless, it is relevant to point out that such changes do not seem to affect the dynamics of the system significantly. To further enforce that, Figure 19 shows the position error on the \hat{x} direction across time for all three systems. From it can be observed that the curves have basically the same shape, but presenting a vertical displacement and a longer time to achieve a steady state.

Figure 20: Position controller response to a finite ramp input of high inclination with a 50 grams mass displaced by 0.1, 0.2 and 0.3 m, from top to bottom, on the positive x axis, on the x-y plane of the quadrotor model.



Source: Author

4.4.2 Position Control and External Applied Torque

Even though the observation made above are interesting, they are not very useful. If we consider, however, that in stable hover, with all rotors effectively pointing upwards, a displaced mass can be replaced by the equivalent resulting torque its weight generates with respect to the geometric center of the quadrotor, and by its weight applied at this same point, we can relate CG displacement with its equivalent torque. Such relation is much more useful since it would offer a way to compensate for the regime error other than adding an integral term to the PD controller. Because of the nonlinear and underactuated

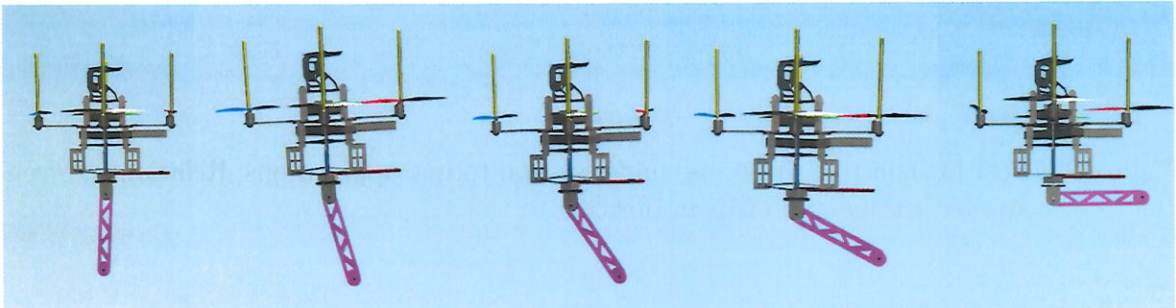
nature of the system, it is hard to obtain such relation analytically. Then, a way around this is to obtain the relation through experiments.

Consider then our quadrotor model with a one DoF articulated rod attached at its bottom. Between the rod and the UAV's frame, we place a six-axis force and torque sensor. This sensor will measure, therefore, all possible efforts that the actuated rod could transfer to the quadrotor. With such information, we can compute the net torque applied on the UAV by the rod as

$$\vec{T}_{net} = \vec{T}_S + \vec{F}_S \times \vec{d}_S \quad (4.14)$$

where \vec{T}_S is the sensed torque, \vec{F}_S and \vec{d}_S is the vector that points from the quadrotor's geometric center to where the sensor is attached. The experiment we proposed to obtain the relation between externally applied torque and the position error uses the setup just described with the rod moving from the vertical to horizontal orientation in a quasi-static manner, as illustrated by Figure 21. The results of the simulated experiment are presented in Figures 22 and 23.

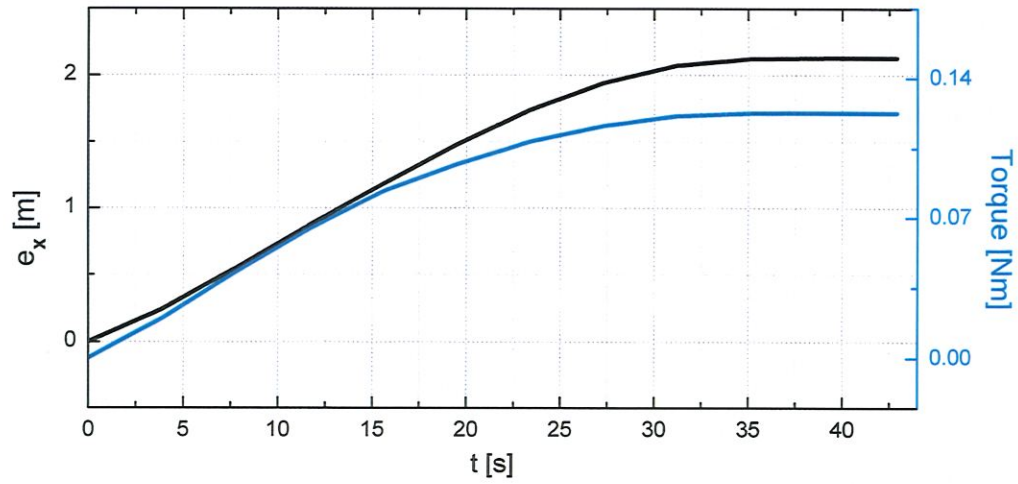
Figure 21: Illustration of quadrotor with a rod moving over time.



Source: Author

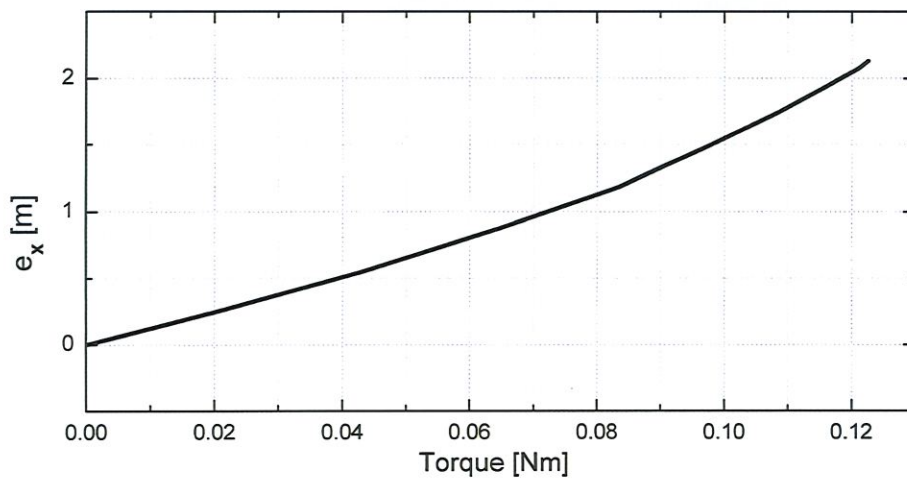
Figure 22 shows the curves of position error along \hat{x} and resulting torque generated by the rod for the whole experiment over time. We can observe that the curves have similar shapes, indicating a strong and direct coupling between the two. From Figure 23, which relates position error to applied torque only, excluding the time component, we notice that the relation between both quantities is almost linear, for the interval the experiments covered. This is a very convenient observation, in the sense that it gives a much more predictable character to the system and makes it easier to address the problem with simple solutions. In the next chapter, a linear compensation based on six-axis F/T sensory feedback is presented as a possible approach to the issues mentioned here.

Figure 22: Position controller response over time under several torque applications.



Source: Author

Figure 23: Position controller response under several torque applications. Relation between applied torque and error in position.



Source: Author

5 SCENARIO ONE: PICK-AND-PLACING WITH F/T SENSING

In this chapter, we address the control problem of Scenario One of aerial manipulation when using force and torque sensors to estimate and compensate for efforts external to the UAV. The majority of the contents presented here are taken directly from [Buzzatto et al. \(2018\)](#), which is an international conference paper published by the author of this work in collaboration with others.

Considering that aerial manipulation has the goal to reach a level where UAVs are capable of dexterously interact with movable objects and fixed features of the environment, such as pipes, structures, and walls, it is reasonable to think that such robots will have the need for Force and Torque (F/T) sensory feedback in their controllers. However, as shown in the literature review, there are very few reports of works using F/T sensory feedback in the control of an aerial manipulator.

[Lippiello and Ruggiero \(2012\)](#) mentioned that classical impedance controllers, as done by [Siciliano and Villani \(1999\)](#), require measurements of external forces, typically obtained at the manipulator end-effector. These authors also argue that this approach is unfeasible for aerial robotics due to disturbances and unmodeled aerodynamic effects that can arise when performing tasks. However, no references or results were presented to support this argument. In another work, [Kim, Seo and Kim \(2015\)](#) defended that, although various researches on ground mobile manipulators used force and torque sensor data on their control ([KARAYIANNIDIS et al., 2012](#); [KARAYIANNIDIS et al., 2013](#); [JAIN; KEMP, 2010](#)), the same measurements may not give beneficial information for aerial manipulators, with the argument that aerial manipulators comply to external forces. Again, no references are cited to support their position.

Going in the opposite direction, this chapter focuses on exploring the possibilities that aerial manipulation based on F/T sensory feedback might offer, applying it mainly for pick-and-placing tasks. The design we propose uses of a 6-axis F/T sensor on an aerial manipulator composed by a quadrotor and a 4-DoF robotic arm. Much like in the previous chapter's last situation simulated, the 6-axis F/T sensor is placed on the interface of the UAV with the robotic manipulator. This setup would allow the attitude and position controllers to compensate for forces and torques exerted on the UAV's structure, that passes through the sensor.

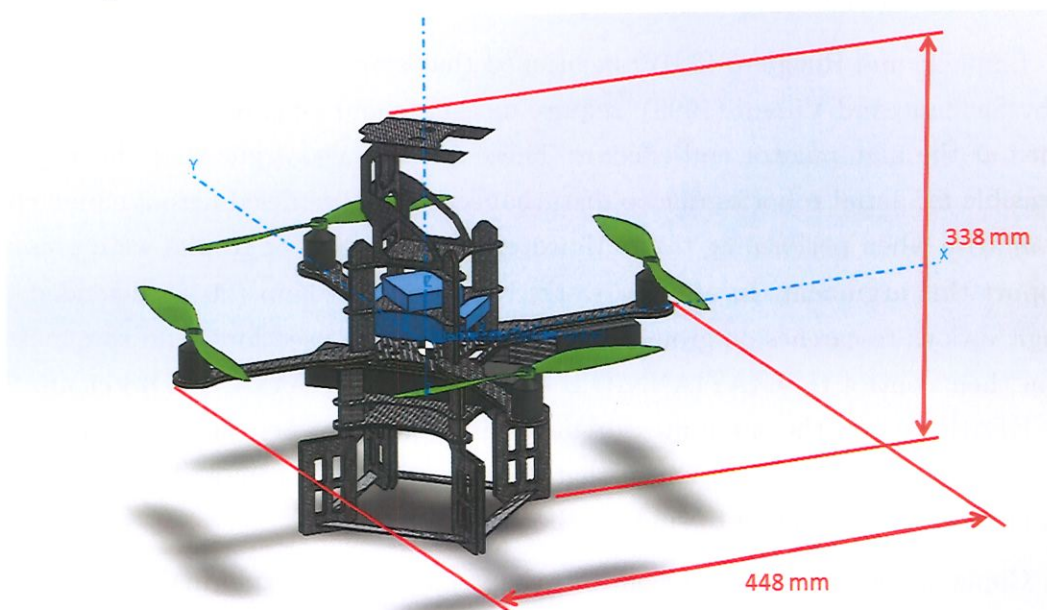


5.1 Our Aerial Manipulator Simulation Model

5.1.1 The Quadrotor

The UAV simulation model is an Asctec Pelican quadrotor (PELICAN, 2018). Since MuJoCo allows the usage of meshes on its models, we imported the freely available CAD files of the UAV found on Asctec web site, which gave a visual appearance to the vehicle that is very similar to the original Pelican. With regard to physical properties, the same CAD model was accordingly setup, in terms of materials and components, to reflect the Pelican exemplar that is available on the dependencies of the Mobile Robotics Laboratory (LabRom) at EESC. From this model, the weight and inertia matrix of the structure and rotors were estimated by the CAD software SolidWorks (SOLIDWORKS, 2018). The CAD model with its approximate dimensions can be seen in Figure 24. The quadrotor inertial parameters used in the model are presented in Table 1.

Figure 24: Asctec Pelican CAD model with approximated dimensions.



Source: Author

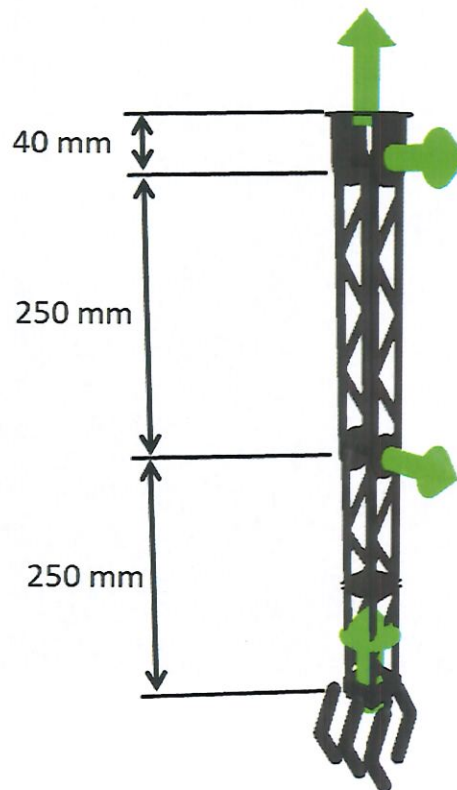
Table 1: Inertial parameters for the Pelican quadrotor model

Mass [kg]	Inertial matrix [kg m ²]		
	0.0031887	-0.0000038	-0.0000881
1.4699013	-0.0000038	0.0032245	0.0000846
	-0.0000881	0.0000846	0.0013857

5.1.2 The Robotic Arm

An arm with four DoF and 0.5 m of total length was our choice to explore pick-and-placing tasks. The whole arm weights 0.34 kg, and it is attached below the UAV. The chosen configuration of the four DoF (see Figure 25) gives the arms control over its position on all three axes, and enable rotation of its gripper. Controlling the position of the EEF in 3D space independently from UAV's position provides the capability of compensating for any position errors of the UAV. This is an advantage since the dynamics of the UAV's position controller is much slower than that of the robotic arm. Also, the possibility of controlling the gripper's orientation allows grasping and manipulation of objects with different shapes, adding more dexterity to the system. Differentiating from what was done with the quadrotor, the inertia matrix for each body of the arm model was left to be calculated by MuJoCo. When a model is loaded on MuJoCo, geometries, and meshes that have no explicitly defined inertia matrix receive one that is calculated by MuJoCo based on its geometry and given mass or density.

Figure 25: Robotic arm model with dimensions

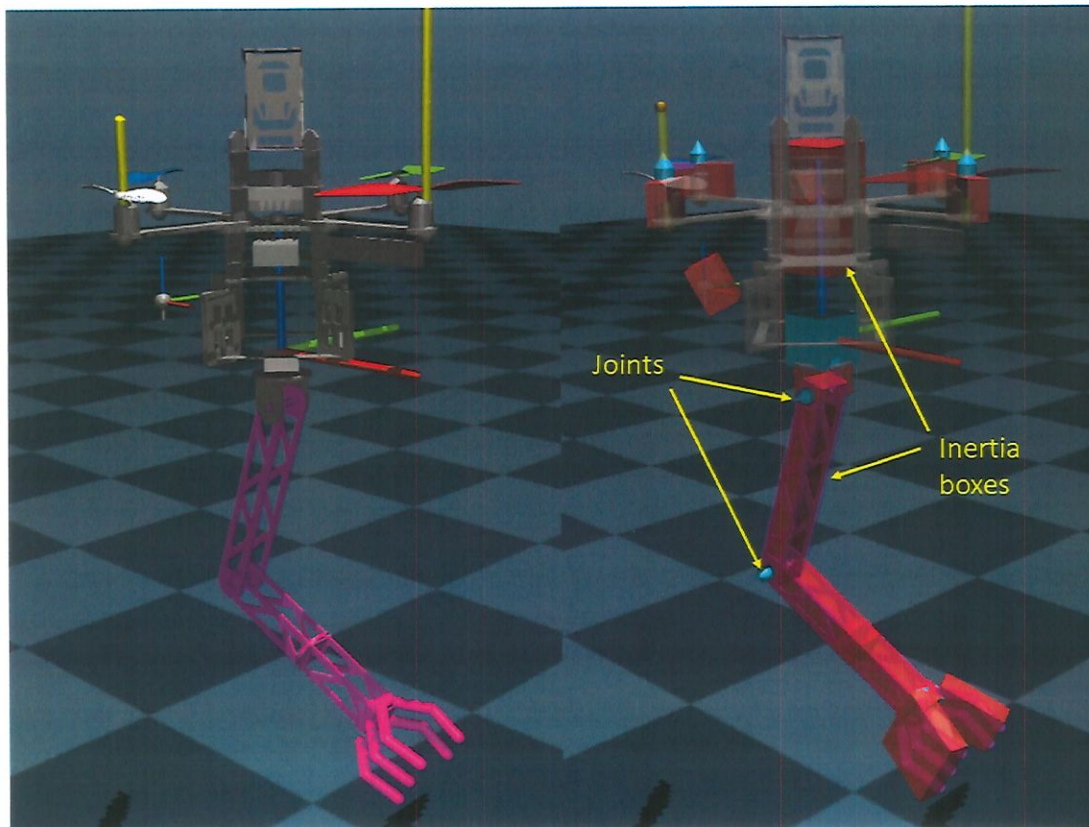


Source: Author

The other essential elements that must be defined on MuJoCo to simulate a system are joints. There are some fundamental joint types already implemented on the engine

that allows the user to represent almost any system. Those are *free*, *ball*, *slide* and *hinge* joints. On our model, only free and hinge joints were used. The first makes the associated body a floating body, allowing motion on all six DoF. This joint is used on the quadrotor body. All other joints on the model are hinge joints, which allow motion on only one rotational DoF. Those are represented by the green arrows on Figure 25. Also, every joint on the model has an associated damping value. All it does is to apply a force that is linear in velocity (MUJOCO.ORG, 2018). It is an important attribute since zero damping can make simulations unstable, especially regarding hinge joints. On Figure 26 is shown the whole model as it appears on the native MuJoCo visualizer, with indications of inertia parameters and joints. On the visualizer, the inertia matrix associated with each body is represented as a box, called *Inertia Box*.

Figure 26: Whole model on MuJoCo native visualizer. On the right image the model is set to be translucent, and there are indications of joints and inertia boxes.



Source: Author

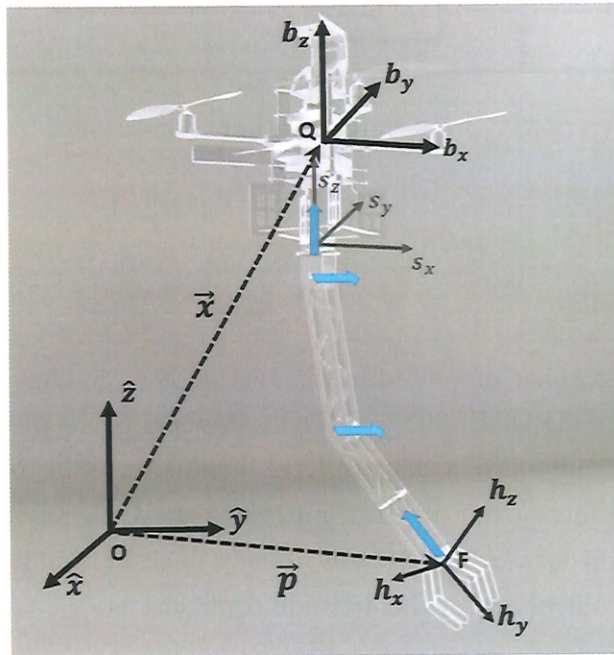
Another important but not necessary element for simulations is actuators. Actuators are associated with joints, and again there are some types already implemented on MuJoCo. The only one used on the model is of the type *velocity*. This type of actuator creates a velocity servo with an associated gain k_v , and it was used to model the actuators of the rotors. The other actuators of the model (defined on the next section) were created for

this work, given the need of implementing a PID controller directly on the actuator. This is because this kind of approach is not possible when using only the already implemented actuators on the engine. Finally, the model contains a body representing the 6 axis F/T sensor, placed between the quadrotor and the arm. On MuJoCo, this kind of sensor is a composition of a 3 axis force sensor and a 3 axis torque sensor, defined on the same body.

5.2 Control

Consider the Aerial Manipulator represented on Figure 27. The reference frames $\mathbf{B} = \{\hat{b}_x, \hat{b}_y, \hat{b}_z\}$ and $\mathbf{H} = \{\hat{h}_x, \hat{h}_y, \hat{h}_z\}$ are attached to the quadrotor body and the gripper center point, respectively, and move together with it. Both frames position and orientation are represented with respect to the inertial frame $\mathbf{G} = \{\hat{x}, \hat{y}, \hat{z}\}$, as it is the global frame of reference.

Figure 27: System of reference frames and indication of the robotic arms degrees of freedom (blue arrows).



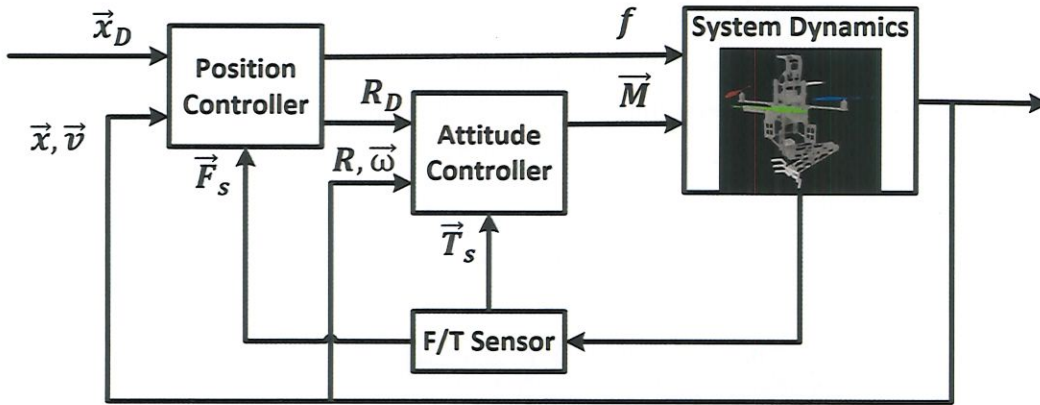
Source: Author

On the control point of view, the aerial manipulator is two separated systems, the quadrotor, and the robotic arm. Therefore, two controllers were used to control the whole system, the quadrotor's controller, responsible for controlling attitude and position of the UAV, and the arm controller, which does the inverse kinematics of the manipulator and controls the position of its EEF, taking into account the position of the quadrotor.

The quadrotor controller is a composition of two controllers, one for attitude and one for position control. Due to the underactuated nature of a quadrotor, it is only possible

to directly control four states of the UAV, these being the position on \hat{z} direction, and the rotations about \hat{b}_x , \hat{b}_y and \hat{b}_z direction. To control the position on \hat{x} and \hat{y} directions, the position controller makes use of the attitude controller, moving the system in these directions by changing the attitude of the UAV in a simple coordinated way. The control block diagram for the system is illustrated in Figure 28.

Figure 28: Control block diagram of the aerial manipulator system.



Source: Author

5.2.1 Attitude Controller

The attitude controller presented in this section is a modified version of the one in (LEE; LEOKY; MCCLAMROCH, 2010) and it is very similar to the one presented in the previous chapter. One difference is that we have included a term to linearly compensate for sensed torques and forces that are transmitted to the UAV's body. The derivation of this term is very straight forward, and it was already done by Kondak et al. (2014) and it based on (4.14). The control law of the attitude controller is

$$\vec{M} = -\mathbf{J}(\mathbf{K}_R \vec{e}_R + \mathbf{K}_\omega \vec{e}_\omega) + \vec{\omega} \times \mathbf{J}\vec{\omega} + \mathbf{K}_T \vec{T} \quad (5.1)$$

where \mathbf{K}_R , \mathbf{K}_ω and \mathbf{K}_T are gain matrices. The $\mathbf{K}_T \vec{T}$ term compensate for sensed torques and forces that generate torques on the UAV body. Given that the relation between position error and applied torque is practically linear, the constant gain matrix \mathbf{K}_T is enough to compensate for those external efforts, as it will be shown later on. The position controller is then based on this new attitude controller, much like the one presented in the last chapter. The only difference is that now we also add an integral term to correct some fine errors and make the system more precise.

5.2.2 Arm Controller

To control the robotic arm, a simple PID was used in every joint. The input of the controller is the global desired position, expressed in \mathbf{G} . The controller then transforms this position to be expressed in the frame \mathbf{B} , and passes it down to the inverse kinematics, represented on (5.2) through (5.4) (SICILIANO et al., 2008). The inverse kinematics calculates the desired angle positions and passes it down to the PID of the joints.

$$\Theta_1 = \arctan\left(\frac{x_{dy}}{x_{dx}}\right) \quad (5.2)$$

$$\Theta_2 = \arctan\left(\frac{\sqrt{x_{dx}^2 + x_{dy}^2}}{-x_{dz} - l_1}\right) - \arccos\left(\frac{l_2^2 + (-x_{dz} - l_1)^2 + x_{dx}^2 + x_{dy}^2 - l_3^2}{2l_2\sqrt{(-x_{dz} - l_1)^2 + x_{dx}^2 + x_{dy}^2}}\right) \quad (5.3)$$

$$\Theta_3 = \arccos\left(\frac{l_2^2 + (-x_{dz} - l_1)^2 + x_{dx}^2 + x_{dy}^2 - l_3^2 - 2l_2^2}{2l_2l_3}\right) \quad (5.4)$$

where $l_1 = 40$ mm and $l_2 = l_3 = 250$ mm are the lengths of the arm's links, and Θ_j is the calculated angle of the respective DoF.

5.2.3 Arm Actuators

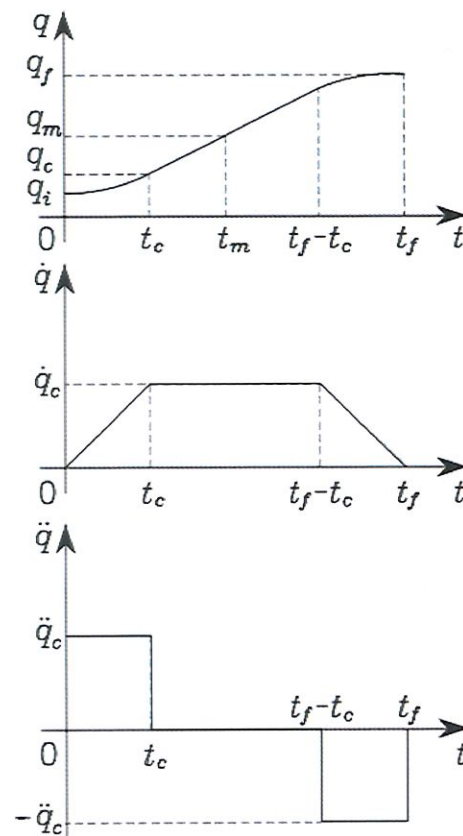
The other actuators used on this aerial manipulator model were defined outside of the already existing actuators types of MuJoCo. They were implemented via a MuJoCo's function designed to apply forces on the given DoF of the model. The output force, or as in the case, torque, have the dynamics of a common PID controller, as expressed by (5.5). This approach makes it easier to control the applied torque on the joints, on the coding point of view.

$$\tau_j = k_{pj}(\Theta_{dj} - \Theta_j) + k_{vj}(\omega_{dj} - \omega_j) + k_{ij} \int (\Theta_{dj} - \Theta_j) dt \quad (5.5)$$

5.2.4 Trajectories

The trajectories generated and passed to all controllers were based on Point-to-Point motion. The framework for it was based on functions that received as inputs the final position, maximum velocity, starting time and duration time. Given this data, the trajectory generating function will get the actual position of the regarded element, being it a joint actuator or a desired position vector, and create a trajectory with a typical trapezoidal profile for the velocity (Figure 29).

Figure 29: Trajectory with trapezoidal velocity profile.



Source: (SICILIANO et al., 2008)

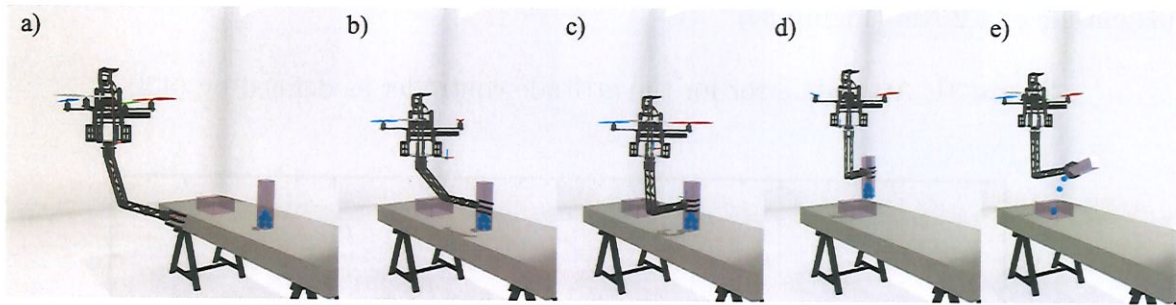
5.3 Simulation Results and Discussions

5.3.1 Pouring Task

One task we proposed to evaluate the system capabilities and the control is a pouring task. At the starting point, the quadrotor is stably hovering 1 m far, on the \hat{x} direction, from a long jar it needs to grasp (with dimensions 0.07 x 0.07 x 0.3 m and weighing 45 g). This jar is filled with six balls (with 0.04 m of diameter and weighing 25 g each). Then, the whole system approaches the jar, maintaining the gripper 0.1 m away from it. Next, only the arm moves and grasp the object. After the jar is grasped, the whole system moves 0.2 m in the \hat{z} . Again, the whole system now moves 0.3 m in the \hat{y} direction. Finally, the gripper rotates 90 degrees to empty the long jar on to a short and wide jar (with dimensions 0.2 x 0.2 x 0.1 m) that is laying on the table. The long jar plus the six balls weight a total of 195 g.

The described task is very demanding for an aerial manipulator for some reasons. One is that the total payload represents approximately 10% of the total weight of the

Figure 30: Time lapse of the simulated task.



a) $t = 0$ s, the aerial manipulator starts to move in the direction of the jar. b) $t = 4$ s, approaching to grasp. c) $t = 9$ s, object is grasped. d) $t = 15$ s, the system lifts and transports the payload. e) $t = 19$ s, the contents of the long jar are poured on to another recipient.

Source: Author

whole system (UAV + arm), offering the potential to cause considerable disturbances. Another reason is the geometry of the grasped object and the contents of it. The long length and small base results in a great unbalance during manipulation, demanding more torque of the gripper for turning the object, which is transferred to the frame of the quadrotor. Moreover, the pouring causes changes to the object's mass and moments of inertia.

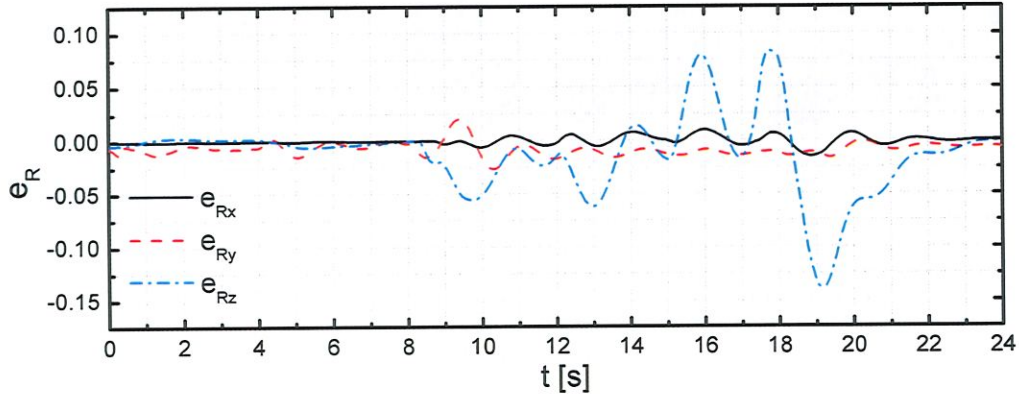
5.3.1.1 Results and Discussion

The control of the proposed aerial manipulation system can be broken down to three main controllers: Attitude controller of the quadrotor, position controller of the quadrotor and position controller of the robotic arm. Figures 31 through 33 shows the desired and actual attitude or position across the time of the three controllers during the task. To give a better understanding of their behaviors and to see each step of the task clearly, Figures 34 and 35 shows the measurements of the F/T sensor across time. Also, screenshots of the time lapse of the task are illustrated in Figure 30.

One problem is the movement of the robotic arm in the \hat{y} direction when the EEF is distant from the CG of the quadrotor. The EEF is always moving faster than the quadrotor, and this difference causes small torques in the \hat{b}_z direction, and forces on \hat{b}_y direction, resulting in small oscillating deviations for attitude in \hat{b}_z and position in \hat{b} . However, concerning the attitude controller, rotations on the \hat{b}_z directions are always more problematic due to the net torque of actuation in this direction being composed only of the torques generated by drag on the rotors. These torques are much smaller when compared with the torque generated by the thrust of the rotors on the other directions. For that reason, controlling attitude on \hat{b}_x and \hat{b}_y are easier, resulting in small attitude

errors for these directions (Figure 31), even though the torques measured there reached a magnitude of 1.2 Nm (Figure 34).

Figure 31: Attitude error for the attitude controller as defined by (4.2).

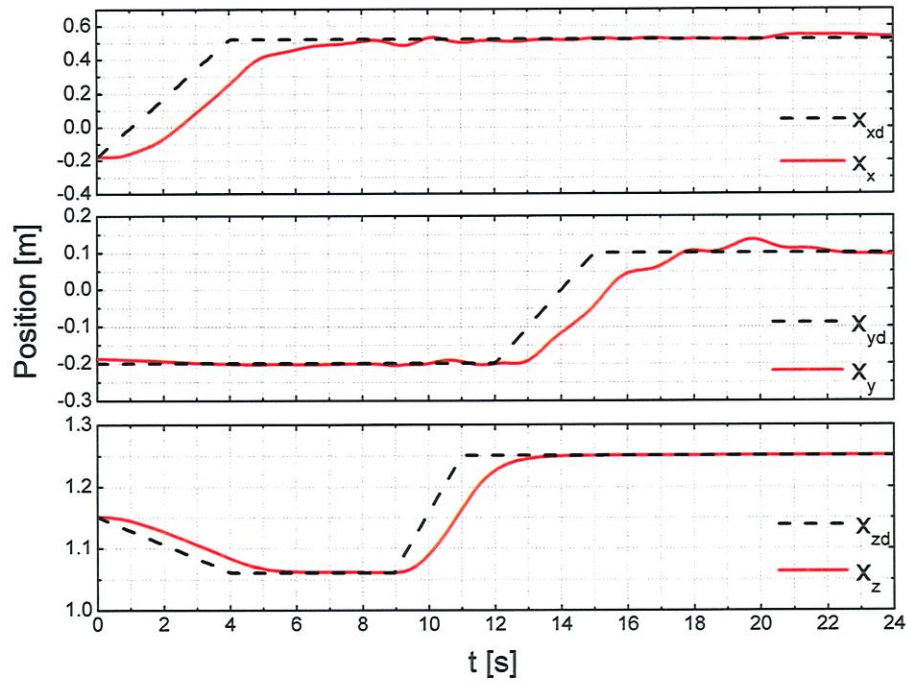


Source: Author

The good performance of the attitude controller directly impacts on the good performance of the position controller. This is because the position controller for the quadrotor uses the attitude controller, as mentioned before. Figure 32 shows that the controller responds rapidly to changes on the forces exerted on the quadrotor through the robotic arm. A good example is the task's step where the jar is being emptied. During this step, the quadrotor is hovering in a fixed position, and only the gripper must rotate. In $t = 19$ s on Figure 35, the change in the force for the \hat{b}_z direction can be seen, representing the jar being emptied, and this change does not alter the position of the quadrotor, in Figure 32.

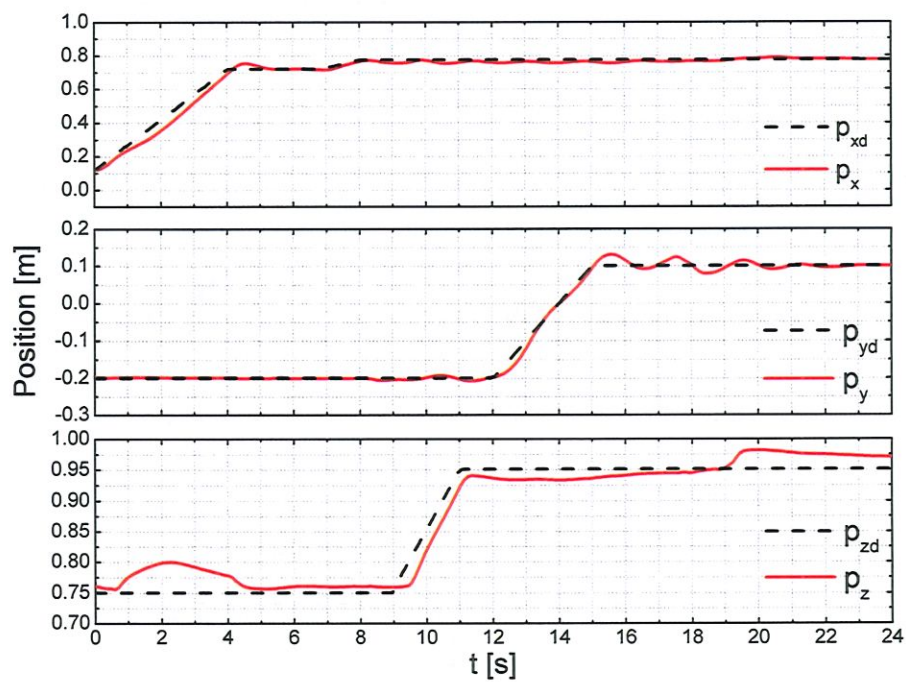
However, it is observed that when the quadrotor should move, it always does it with some latency. Compensating for that is the position control of the arm. Figure 33 shows that most of the time the EEF tracks its desired position well enough, save for some points, as in the beginning of the task, where due to the latency of the quadrotor, the EEF reaches its maximum range and still do not catch up with its target for some time ($t = 1$ through $t = 5$ s). Also, by the end of the task, after the pouring step, the integral term of the arm controller takes some time to decrease its error, resulting in the slow decreasing shift in the \hat{z} position of the EEF.

Figure 32: Position of the quadrotor. Dashed black line is the desired position and solid red line is the actual position.



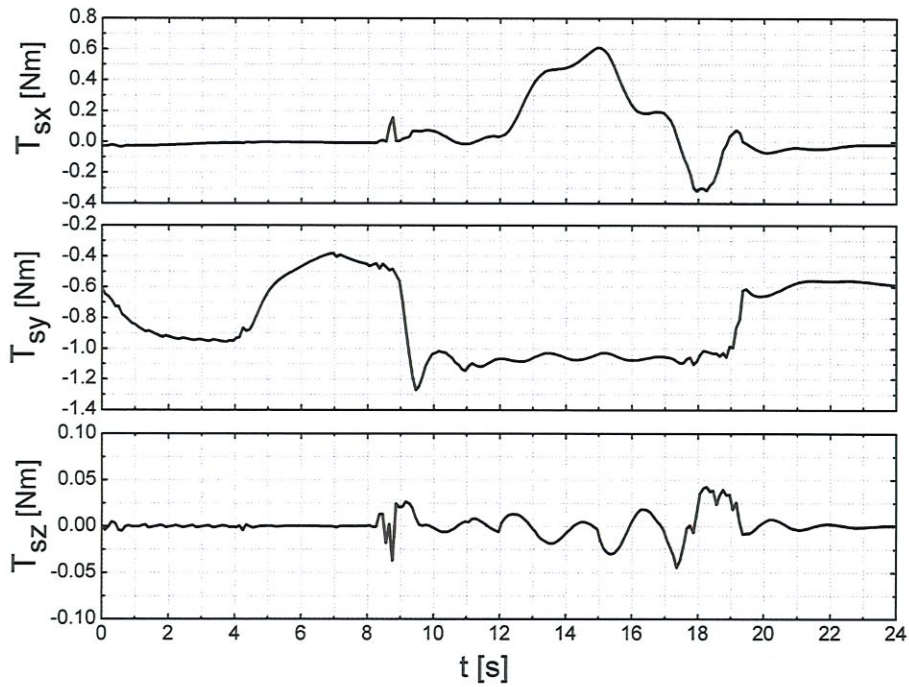
Source: Author

Figure 33: Position of the systems End Effector. Dashed black line is the desired position and solid red line is the actual position.



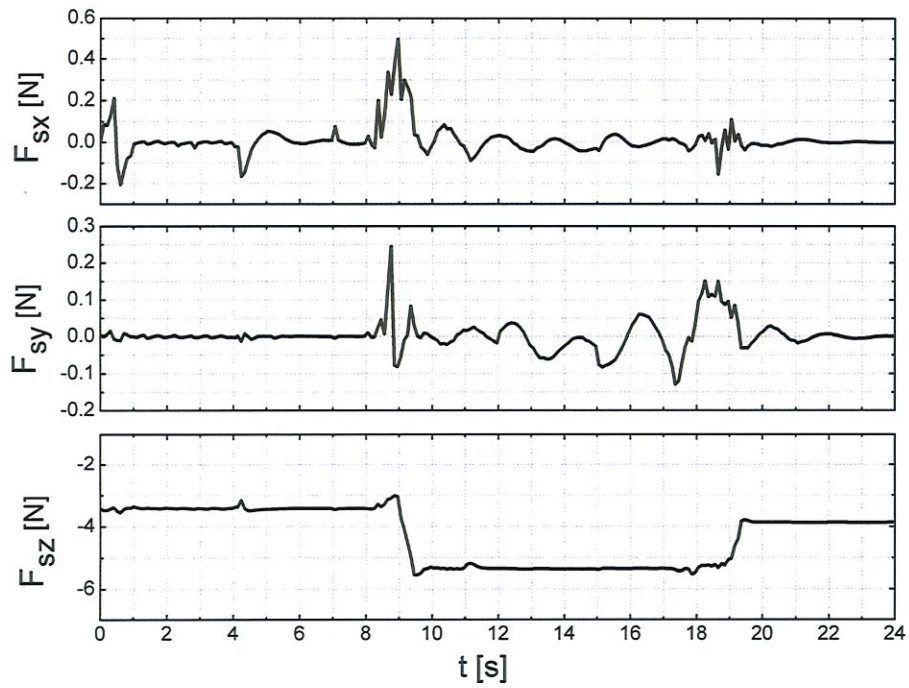
Source: Author

Figure 34: Torques measured by the simulated six axis F/T sensor.



Source: Author

Figure 35: Forces measured by the simulated six axis F/T sensor.

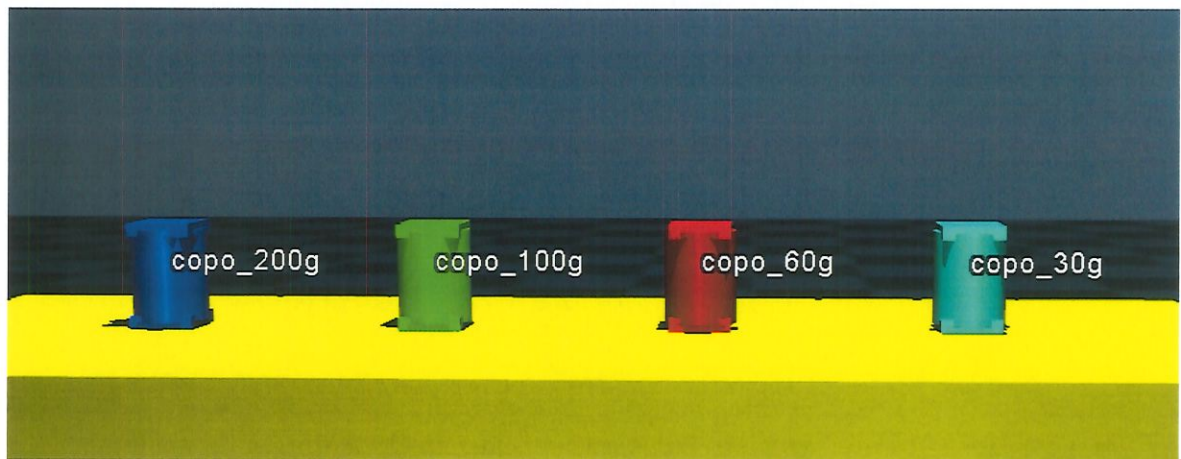


Source: Author

5.3.2 Cups Piling Task

Here we consider a more precise task, where four cups with 54 mm of diameter and 70 mm of height are piled by the aerial manipulator. Each cup has a different weight. The weights are 30 g, 60 g, 100 g and 200 g which are associated with the color light blue, red, green and blue, respectively (Figure 36). In this task, the aerial manipulator starts hovering in a position near to the table where the cups are resting. Then, the system piles the cups in a sequence. First, the 30 g cup is piled over the 60 g one. Next, the 200 g cup is piled over the 100 g cup. Finally, the pile with the 30 g and 60 g cups are picked up and placed over the other pile with the cups of 100 g and 200 g. Each step in the simulation is programmed by hand. Figure 37 shows a time lapse of this task simulated on the MuJoCo environment. As one can see, our aerial manipulation system manages to do this precise and demanding task satisfactorily.

Figure 36: Cups and it's respective weights in grams.

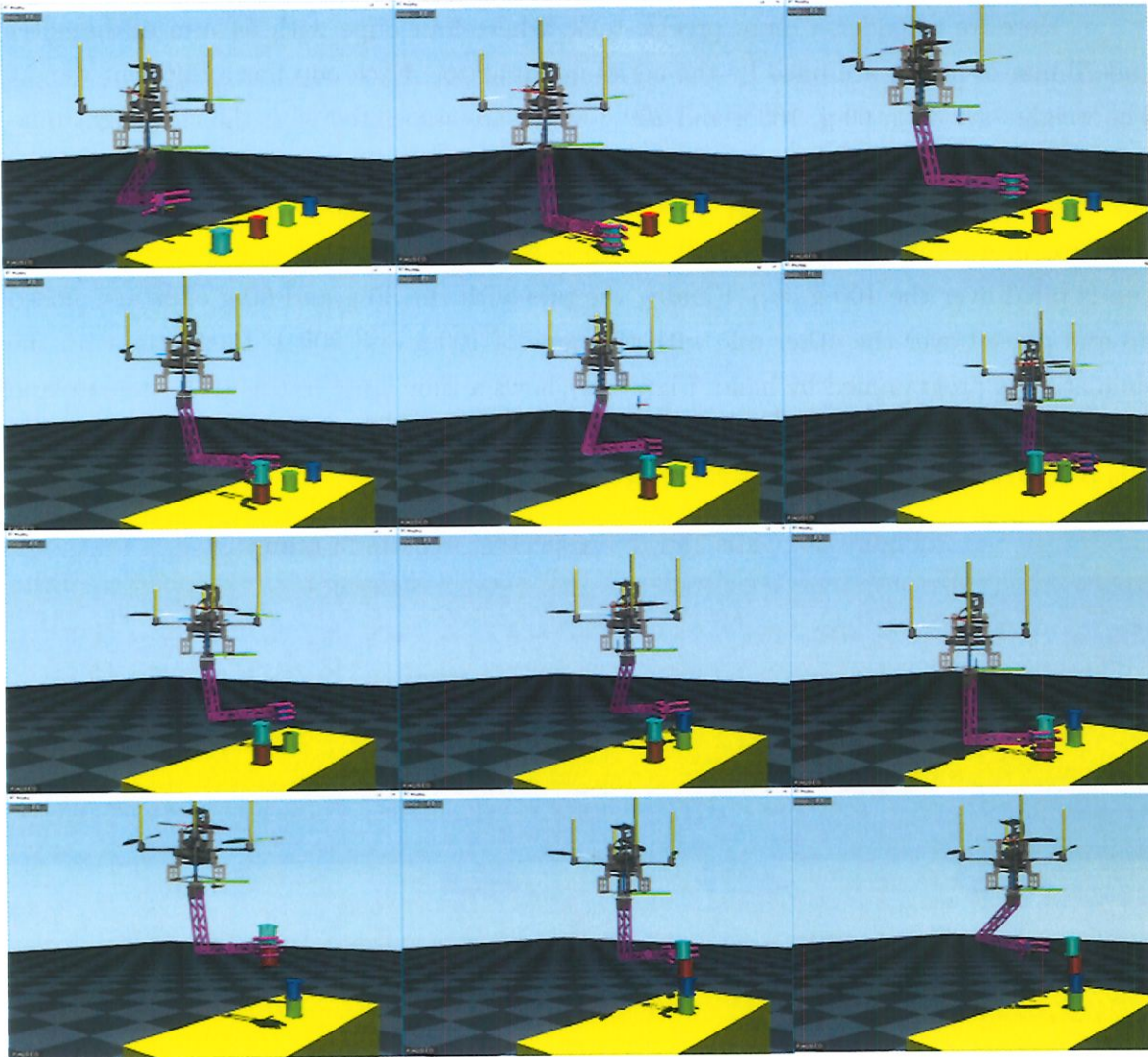


Source: Author

However, when simulating this same task with the effects of blade flapping, the aerial manipulator no longer managed to pile all the cups. Because of the relative small diameters of the cups, and also the exaggerated parameters we used to simulate the blade flapping effect, the oscillation in position and velocity it causes made the precise positioning of the cups hard. The swinging of the quadrotor was too fast for the robotic arm movement to compensate for the errors. This was not a problem for the picking-up parts of the task. However, the system was not able to place the cups precisely enough, where in many cases the cups fell after being released.

The tasks presented in this chapter illustrate, at least in simulations, the capacities of the proposed framework for pick-and-placing of objects that are relatively heavy with respect to the aerial manipulator's total weight. The approach is promising and seems to be a good and simple solution for a complex problem in aerial manipulation. It takes

Figure 37: Time lapse of the cups piling task.



Source: Author

advantage of the usage of sensors instead of complex estimation algorithms employed on some works in the literature (KIM; CHOI; KIM, 2013; MELLINGER et al., 2011). However, in order to prove the concept, experiments must be made with a real robot. This would further clarify important issues such as noise on the F/T sensor and how the system would work with it (this was not taken into account on simulations).

6 SCENARIO TWO (PARTIAL): COMPENSATION OF 3D FORCES WHILE IN HOVER

In this chapter, we address a completely different problem from the one presented in the previous chapter. Here, we consider the control problem of a hovering quadrotor aerial manipulator equipped with a one DoF rod that is subject to generalized 3D forces applied at its end-effector. This partially covers Scenario Two since we do not address the problem of a hovering aerial manipulator subjected to generalized 3D torques. Nevertheless, the situation we consider have, intuitively, more potential applications than its complementary part. The majority of the contents herein are taken directly from [Buzzatto et al. \(2019\)](#), a paper that was submitted for publishing at an international conference by the author of this work in collaboration with others.

Much has been done on task orientated research, such as assembly of structures with quadrotors teams ([LINDSEY; MELLINGER; KUMAR, 2012](#)), cooperative transportation with cables ([JIANG; KUMAR, 2013](#)), and autonomous manipulation on complex environments ([BAIZID et al., 2017](#)). Similarly, some works explored aerial manipulation with Image-Based Visual Servoing ([MEBARKI; LIPPIELLO; SICILIANO, 2015](#)), ([KIM et al., 2016](#)). Many successful control strategies for motion control of an aerial manipulator and compensation for changes on the Center of Gravity (CG) position are already available ([JIMENEZ-CANO et al., 2013](#); [KIM; CHOI; KIM, 2013](#); [MELLINGER et al., 2011](#)), as well as several mechatronic design proposals ([SUAREZ; HEREDIA; OLLERO, 2015](#); [NIKOU; GAVRIDIS; KYRIAKOPOULOS, 2015](#); [KAMEL; COMARI; SIEGWART, 2016](#); [FUMAGALLI; STRAMIGIOLI; CARLONI, 2016](#); [DANKO; OH, 2014](#)). However, little is known concerning the behavior of aerial manipulator's dynamics when under substantially demanding tasks, whether it is a pick-and-place task or a force exertion situation. This is especially true for systems of the third approach, leading to the poor performance of controllers, and limiting their application to lightweight payloads or small interaction forces. To best of our knowledge, few works have contributed to filling in this gap in the literature.

On ([GIOIOSO et al., 2014](#)), an indirect force controller for a near-hovering quadrotor equipped with a rigid tool was developed. The same controller was used for free-flight and contact phase. The system performance, however, was limited, managing to apply only small magnitude pushing forces. In ([KIM; SEO; KIM, 2015](#)), a framework for operating an unknown drawer with a quadrotor and a robotic arm was presented. Postures for pushing and pulling the drawer were determined based on modeling, where efforts were also concentrated on estimating the direction of the drawer's motion. The use of a quadrotor as a tool operator is explored in ([NGUYEN; HA; LEE, 2015](#)), where a rigid tool is used. Some interesting observations were carried out with regard to the system dynamics and stability

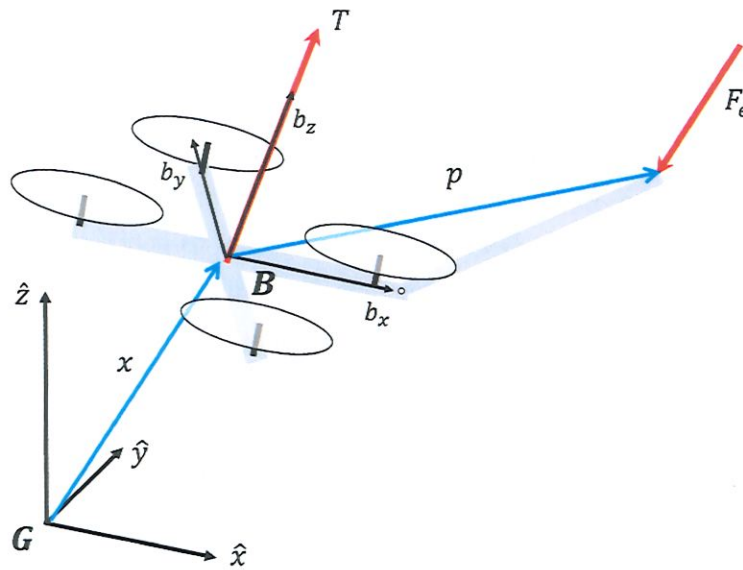


Figure 38: Aerial Manipulator model and frames of reference.

Source: Author

when in contact. However, their conclusions are limited to its proposed application. In (WOPEREIS et al., 2017), a very efficient approach to exert pushing forces on a vertical surface was described. Basically, a quadrotor with a one DoF actuated arm was used and forces equivalent to the quadrotor's mass were sustained for large time intervals. However, the proposed controller used Linear-Quadratic Regulator (LQR) optimization to find an ideal set of gain matrices for each pitch angle performed, complicating the generalization of the method.

Most of the works that address force exertion are limited to low force magnitudes. In addition to this, research tends to be limited to either pushing or pulling strategies, not covering the full spectrum of applications. This chapter was motivated by this lack of solutions and understanding of aerial manipulators dynamics. Here, we pushed forward the fringe of our knowledge on aerial manipulation dynamics by considering a simplified aerial manipulator system. This system is composed of a quadrotor and a one DoF actuated rod subjected to generalized 3D forces applied on its End-Effector (EEF). A control framework for compensating largely sustained pushing and pulling forces is proposed. Pushing and pulling are formally defined, posture conditions for stability are proposed, and simulations are presented to validate both control and propositions. As an example, the findings of this particular part of the work could be directly applied to a liquid spraying task, with the purpose of cleaning or painting surfaces.

6.1 Modeling

Since we consider a different robot model subjected to also a different set of conditions, modeling and representation are presented once again, as this new problem requires.

6.1.1 Representation and Main Assumptions

Consider the schematic representation of an aerial manipulator illustrated in Figure 38. The system is composed of a quadrotor with an embedded one DoF robotic arm of negligible mass. It is important to highlight that the negligible mass assumption was previously addressed in the literature by (NGUYEN; HA; LEE, 2015) and tested experimentally on (WOPEREIS et al., 2017) for systems where the arm is much lighter than the UAV. Reference frame $\mathbf{B} = \{\hat{b}_x, \hat{b}_y, \hat{b}_z\}$ is fixed on the quadrotor's body, with origin coincident with the vehicle CoG, and it moves together with the CoG. The inertial frame of reference is $\mathbf{G} = \{\hat{x}, \hat{y}, \hat{z}\}$. The rod is attached to the UAV's body on the level of its CoG, i. e., confined on the plane formed by \hat{b}_x and \hat{b}_y , and placed along \hat{b}_x to make further analysis simpler. The only actuator of the arm rotates about the \hat{b}_y axis, constraining the manipulator's EEF to the $\hat{b}_x - \hat{b}_z$ plane. Also, it is assumed that external forces are only applied to the EEF.

Orientation is represented by body fixed ZYX Euler angles combination. A preceding subscript indicates the reference frame on which the variable is expressed. This subscript is omitted only when the variable is expressed with respect to the inertial reference frame, \mathbf{G} .

6.1.2 Equations of Motion

Given the aforementioned assumptions, the equations of motion for the system can be derived using the Newton-Euler formalism.

$$m \ddot{\vec{x}} = \vec{T} + \vec{F}_e + \vec{W} \quad (6.1)$$

$${}_B \mathbf{J}_M {}_B \ddot{\vec{\theta}} + {}_B \dot{\vec{\theta}} \times {}_B \mathbf{J}_M {}_B \dot{\vec{\theta}} = {}_B \vec{M} + {}_B \vec{p} \times {}_B \vec{F}_e \quad (6.2)$$

were m is the manipulator mass, \vec{x} is the position of the aerial manipulator's CoG, \vec{T} is the total thrust vector generated by the quadrotor, \vec{F}_e is the perturbation force on the EEF, \vec{W} is the weight vector, ${}_B \mathbf{J}_M$ is the inertia matrix, $\vec{\theta} = (\phi, \theta, \psi)$ is the vector of Euler angles, ${}_B \vec{M}$ is the torque vector generated by the quadrotor, and ${}_B \vec{p}$ is the vector pointing from the CoG to the EEF.

6.2 Control

The proposed attitude and position controllers are again a modified version of those presented in (LEE et al., 2010). The only difference is that it takes into account the external force applied at the aerial manipulator's EEF on the calculation of total thrust and the yaw component of attitude. Considering \vec{F}_e as known and given a desired position \vec{x}_d , \vec{T} and ${}_B\vec{M}$ are calculated as follows:

$$\vec{T}_d = -\vec{W} - \vec{F}_e + \mathbf{K}_p \vec{e}_x + \mathbf{K}_v \vec{e}_v + \mathbf{K}_i \vec{e}_i \quad (6.3)$$

$$\vec{T} = \|\vec{T}_d\| \hat{b}_z \quad (6.4)$$

$${}_B\vec{M} = -\mathbf{K}_R \vec{e}_R - \mathbf{K}_{\omega B} \vec{e}_\omega + {}_B\dot{\vec{\theta}} \times {}_B J_{MB} \dot{\vec{\theta}} \quad (6.5)$$

where \vec{T}_d can be interpreted as the desired total thrust vector. The elements \mathbf{K} are gain matrices. For the position controller, \vec{e}_x , \vec{e}_v and \vec{e}_i are the position error, velocity error and integral error, respectively. They are defined as:

$$\vec{e}_x = \vec{x}_d - \vec{x} \quad (6.6)$$

$$\vec{e}_v = \dot{\vec{x}}_d - \dot{\vec{x}} \quad (6.7)$$

$$\vec{e}_i = \int \vec{e}_x(t) dt \quad (6.8)$$

Meanwhile, for the attitude controller, the errors are defined as:

$$\vec{e}_R = \frac{1}{2} (\mathbf{R}_d^T \mathbf{R} - \mathbf{R}^T \mathbf{R}_d)^\vee \quad (6.9)$$

where \mathbf{R} and \mathbf{R}_d are the rotation matrices representing the actual and desired orientation for the quadrotor's frame of reference, respectively. Both of them are expressed with respect to the inertial frame \mathbf{G} . The terms on the right-hand side of (4.2) yield a skew-symmetric matrix, which can be represented as a vector $\vec{e}_R \in \mathcal{R}^3$ through the *vec* map^v : $SO(3) \rightarrow \mathcal{R}^3$ operator, again as in (4.3).

The error for the angular velocities can be similarly defined, given the desired angular velocity, $\vec{\omega}_d \in \mathcal{R}^3$:

$${}_B\vec{e}_\omega = {}_B\vec{\omega} - \mathbf{R}^T \mathbf{R}_d \vec{\omega}_d \quad (6.10)$$

with ${}_{B}\vec{\omega}$ and ${}_{B}\vec{\omega}_d$ being expressed with respect to the reference frame of the quadrotor, \mathbf{B} .

Considering the underactuated construction of a quadrotor, the position controller relies on the attitude controller by changing the vehicle's orientation to achieve a change in position. This is carried out by defining the desired thrust so it is dependent on the position errors, and then passing the direction of this desired thrust as the z component of the desired attitude rotation matrix, as implied by (6.3) and (6.4).

Given the definition of \vec{T}_d , the relation $\hat{b}_z = \mathbf{R}\hat{z}$, and the property of rotation matrices $\mathbf{R}^{-1} = \mathbf{R}^T$ we can obtain the following relation:

$$\frac{\vec{T}_d}{\|\vec{T}_d\|} = \mathbf{R}_d \hat{z} \quad (6.11)$$

Replacing (6.11) into (6.4) we get:

$$\vec{T} = \mathbf{R}\mathbf{R}_d^T(-\vec{W} - \vec{F}_e + \mathbf{K}_p\vec{e}_x + \mathbf{K}_v\vec{e}_v + \mathbf{K}_i\vec{e}_i) \quad (6.12)$$

When applying the control equations (6.12) and (6.5) to the equations of motion (6.1) and (6.2), we find:

$$m\ddot{\vec{x}} = (I - \mathbf{R}\mathbf{R}_d)(\vec{F}_e + \vec{W}) + \mathbf{R}\mathbf{R}_d(\mathbf{K}_p\vec{e}_x + \mathbf{K}_v\vec{e}_v + \mathbf{K}_i\vec{e}_i) \quad (6.13)$$

$${}_{B}J_M {}_{B}\ddot{\theta} = -\mathbf{K}_R\vec{e}_R - \mathbf{K}_\omega\vec{e}_\omega + {}_{B}\vec{p} \times {}_{B}\vec{F}_e \quad (6.14)$$

where I is the 3×3 identity matrix. The equations above represent the whole dynamic system. Assuming that as $t \rightarrow \infty$, $\vec{e}_x, \vec{e}_v, \vec{e}_i \rightarrow 0$, and $\vec{e}_R, \vec{e}_\omega \rightarrow 0$, then $\mathbf{R} \rightarrow \mathbf{R}_d$, and all the right-hand side terms of both equations tends to zero, except for the torque caused by the external force, represented by the term ${}_{B}\vec{p} \times {}_{B}\vec{F}_e$. Here is where we can use the definition of \mathbf{R}_d and the control of the EEF position in our favor, to make ${}_{B}\vec{p} \times {}_{B}\vec{F}_e \rightarrow 0$.

6.2.1 Controlling the Arm

First, it is necessary to address the position of the EEF with respect to the quadrotor's CoG. Considering Figure 39 and projecting the external force onto the Ω plane formed by $\hat{b}_x - \hat{b}_z$, that here is denoted by ${}_{B}\vec{F}_{e\|\Omega}$, we can eliminate the torque that this force generates on the \hat{b}_y direction by aligning it with vector \vec{p} .

Now, considering Figure 40, the size of the rod from its articulation to the EEF is denoted by l_r and α is the angle formed by ${}_{B}\vec{F}_{e\|\Omega}$ and the axis \hat{b}_x . In order to satisfy

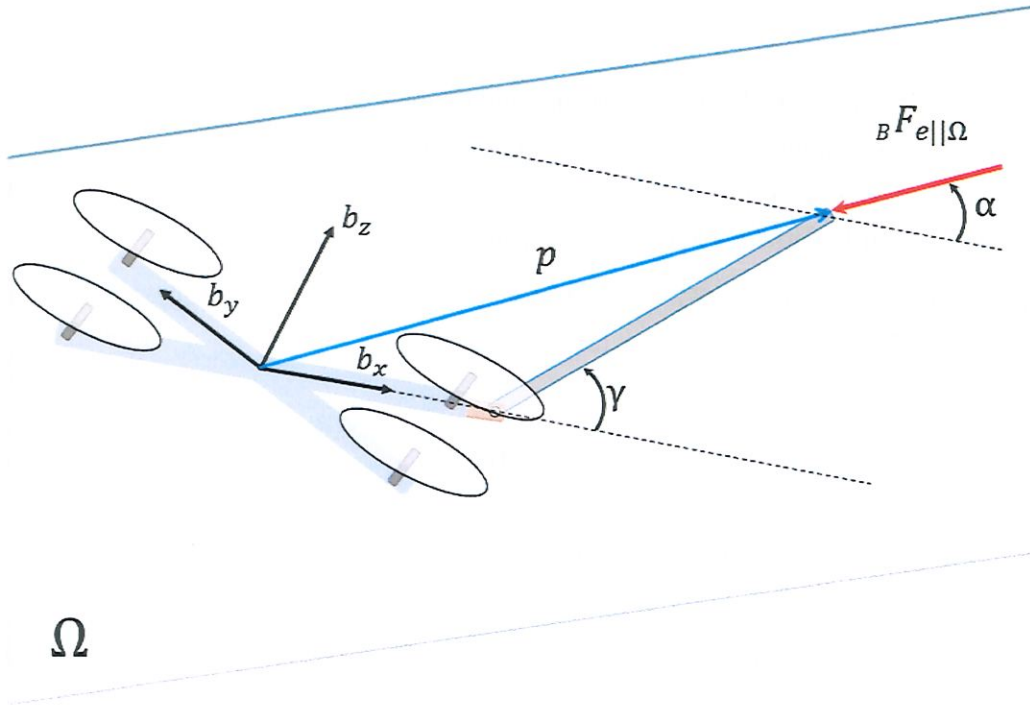


Figure 39: Illustration of the Ω plane, formed by $\hat{b}_x - \hat{b}_z$, to which the articulated arm is confined.

Source: Author

our condition of zero torque on the \hat{b}_y axis, formed by ${}_B \vec{F}_{e||\Omega}$ with respect to CoG of the quadrotor, the angle between \vec{p} and \hat{b}_x must also be equal to α . Under this condition, the following relationship between γ and α can be found:

$$\tan(\alpha) = \frac{{}_B F_{e||\Omega z}}{{}_B F_{e||\Omega x}} = \frac{l_r \sin(\gamma)}{d_m + l_r \cos(\gamma)} \quad (6.15)$$

$$\cos(\gamma) = \frac{-b + \sqrt{b^2 - 4ac}}{2a} \quad (6.16)$$

with

$$c = \tan^2(\alpha) d_m^2 - l_r^2 \quad (6.17)$$

$$b = 2 \tan^2(\alpha) d_m l_r \quad (6.18)$$

$$a = l_r^2 (1 + \tan^2(\alpha)) \quad (6.19)$$

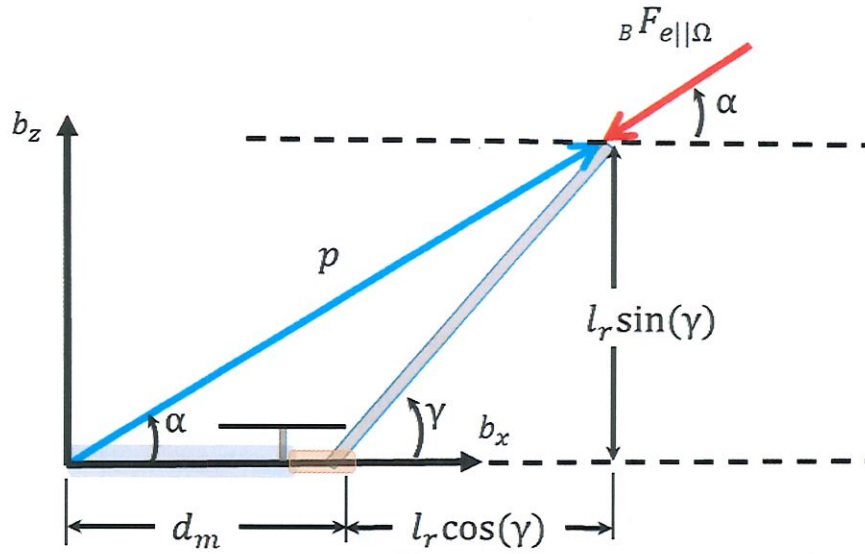


Figure 40: Geometrical relationship between α , the tilting angle of ${}_B \vec{F}_{e||\Omega}$ with respect to \hat{b}_x and γ , the angle the rod makes with respect to its resting position, i. e., parallel to \hat{b}_x .

Source: Author

To guarantee ${}_B \vec{p} \times {}_B \vec{F}_e = 0$ in all directions, ${}_B \vec{F}_e$ must be fully contained within Ω . In order to achieve that, \mathbf{R}_d can be chosen as explained in the following section.

6.2.2 Defining the Desired Attitude

The construction of the desired attitude, \mathbf{R}_d , is carried out as previously presented in (LEE et al., 2010), as (6.20).

$$\mathbf{R}_d = [\hat{b}_{xd}, \hat{b}_{zd} \times \hat{b}_{xd}, \hat{b}_{zd}] \quad (6.20)$$

As mentioned before and formalized in (6.21), \hat{b}_{zd} is chosen so to be parallel to \vec{F}_d . The equivalent x component of \mathbf{R}_d matrix, \hat{b}_{xd} , however, is harder to visualize. It is easier to define a vector contained on the $\hat{x} - \hat{y}$ first, to express a general desired direction. Then, this vector, \hat{b}_{xdd} , is projected onto the plane defined by \hat{b}_{zd} . For our goal of making ${}_B \vec{F}_e$ fully contained within the Ω plane, we define \hat{b}_{xdd} so that it is equal to \vec{F}_e . Figure 41 illustrate the transformation of \hat{b}_{xdd} in to \hat{b}_{xd} and (6.22) and (6.23) defines it mathematically.

$$\hat{b}_{zd} = \frac{\vec{T}_d}{\|\vec{T}_d\|} \quad (6.21)$$

$$\hat{b}_{xd} = \frac{1}{\|\hat{b}_{zd} \times \hat{b}_{xdd}\|} (\hat{b}_{zd} \times (\hat{b}_{zd} \times \hat{b}_{xdd})) \quad (6.22)$$

$$\hat{b}_{xdd} = \frac{\vec{F}_e}{\|\vec{F}_e\|} \quad (6.23)$$

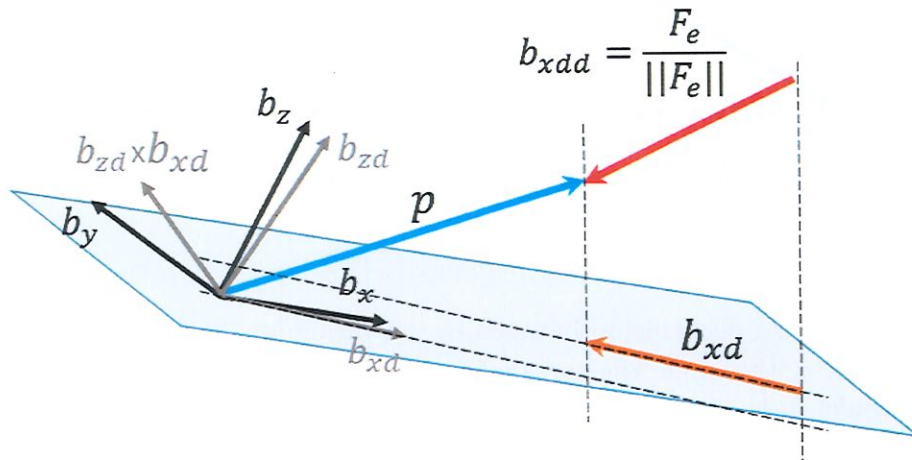


Figure 41: Illustration of \hat{b}_{xd} and \hat{b}_{xdd} .

Source: Author

6.3 Geometrical Stability Conditions

Given the underactuated construction of a quadrotor and the fact that it can only apply forces directly in one direction, \hat{b}_z on our case, position control is built on top of the attitude controller, as mentioned before. If, for instance, we assume that the system is already in equilibrium at its desired Cartesian position, i. e., $\vec{e}_x = \vec{e}_v = 0$, the direction of the desired applied force, or \hat{b}_{zd} , will only be governed by \vec{F}_e , since \vec{W} does not change. Then, to remain in equilibrium, the system must keep the sum of forces and torques equal to zero. Considering this, there are four combination of postures that satisfy this condition, as illustrated on Figure 42. The same direction of \vec{T} is kept for two different quadrotor's attitude, when vector \vec{p} points on the opposite direction of \vec{F}_e , Figure 42 a) and c), resulting in $\frac{\vec{p} \cdot \vec{F}_e}{\|\vec{p}\| \|\vec{F}_e\|} = -1$ and when vector \vec{p} points on the same direction of \vec{F}_e , Figure 42 b) and d), resulting in $\frac{\vec{p} \cdot \vec{F}_e}{\|\vec{p}\| \|\vec{F}_e\|} = 1$. Here we name those two solutions *Pushing*, for $\frac{\vec{p} \cdot \vec{F}_e}{\|\vec{p}\| \|\vec{F}_e\|} = -1$ and *Pulling*, for $\frac{\vec{p} \cdot \vec{F}_e}{\|\vec{p}\| \|\vec{F}_e\|} = 1$.

Given the above considerations, (6.23) alone does not allow *Pushing* postures. To remedy that, the following correction is needed:

$$\hat{b}_{xdd} = -\frac{\vec{F}_e}{\|\vec{F}_e\|} \frac{\vec{p} \cdot \vec{F}_e}{\|\vec{p} \cdot \vec{F}_e\|} \quad (6.24)$$

This shall prevent the rotation of π radians on ψ for the desired attitude on the case of a sudden change in direction of the force to be compensated, which would probably result in a loss of stability for the system.

However, permuting those two solutions with the other two obtained from the solution of (6.16), it is noticeable that postures b) and c) does not satisfy the condition $\vec{p} \times \vec{F}_e = 0$. Therefore not achieving equilibrium for the sum of torques in (6.14). This result requires an additional correction for (6.16):

$$\gamma = \arccos\left(\frac{-b + \sqrt{b^2 - 4ac}}{2a}\right) \frac{\vec{p} \cdot \vec{F}_e}{\|\vec{p} \cdot \vec{F}_e\|} \quad (6.25)$$

Such correction automatically adjusts the aerial manipulator posture so to minimize torque around CM. The above geometric analysis formalize the intuitive notion that, under the considered conditions, to exert *pulling* forces the EEF of the manipulator must be positioned below the quadrotor's CM, when expressed with respect to frame **B**, and to exert *pushing* forces the EEF must be positioned above the quadrotor's CM, when expressed with respect to frame **B**. This concept have already been exploited on some works when trying to exert substantial forces, as in (WOPEREIS et al., 2017; GIOIOSO et al., 2014) for *pushing* and in (WUTHIER et al., 2017) for *pulling*. However, the concept has not yet been formalized. Considering just the stable postures, there is another observation that can be made. Consider the scheme on Figure 43. Both situations of *pushing* and *pulling* are depicted when the system is projected on the Γ plane defined by vector \vec{p} and \hat{b}_y . It is unavoidable not to make a comparison of the system seen from Γ plane with the vertical pendulum problem. Right away we see that the *pushing* posture is similar to the upward equilibrium point for the vertical pendulum, which is naturally unstable, while the *pulling* posture is similar to the downward equilibrium point, which is naturally stable. Such observation gives an interesting insight concerning the dynamic behavior of the aerial manipulation problem considered here, that also has not yet been formally pointed out. On the following section, simulation results are presented, where the concepts highlighted above are illustrated and the control strategy is validated.

6.4 Simulation Results and Discussions

Figure 44 shows the simulator view for the aerial manipulator model. The quadrotor used is again based on the Asctec Pelican (PELICAN, 2018). The pink cylinder is the

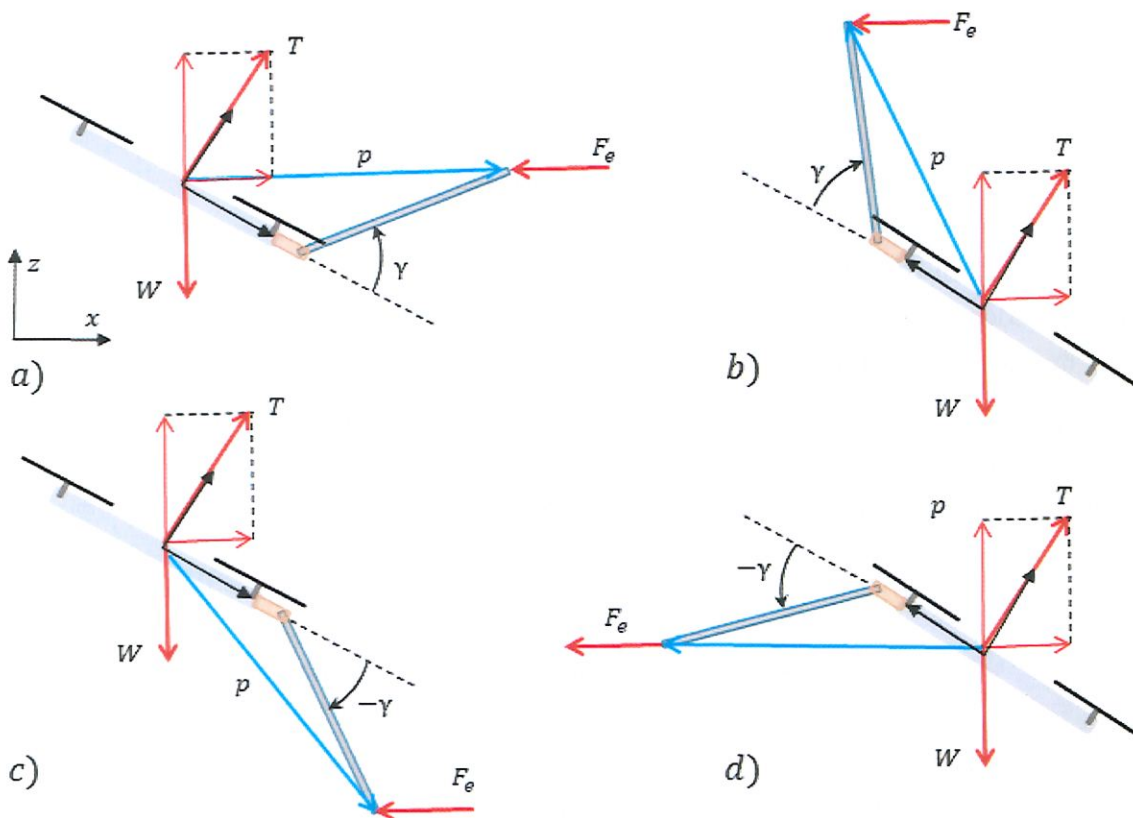


Figure 42: Combination of solutions for static sum of forces. a) Stable configuration for *pushing* b) Unstable configuration for *pulling* c) Unstable configuration for *pushing* d) Stable configuration for *pulling*

Source: Author

actuated rod. On green and yellow are representations of the applied force \vec{F}_e and the thrust of each rotor, respectively. The whole model weights approximately 1.46 kg.

To demonstrate first the stability of the two equilibrium points for the aerial manipulation concept herein described, i. e., *pushing* and *pulling*, consider Figure 45. It presents the simulation results for several applied forces. All forces were applied with a time-varying hyperbolic tangent function, so to amortize the imposed effort, and with \hat{z} component equal to -3 . For this set of tests, the signal of the applied force shown in the legend indicates *pushing* for negative, and *pulling* for positive, due to the initial attitude of the quadrotor, with \hat{b}_x being parallel to \hat{x} . We can see that for *pushing*, magnitudes equal to 8 N or larger turns the system unstable over time. Meanwhile, for *pulling*, the system proved to be stable for magnitudes up to 20 N and larger.

For a better understanding of the *pushing* instability, consider Figure 46. It shows the attitude error for the y axis, e_{RY} , on the left and the $\vec{p} \times \vec{F}_e$ on the right, across

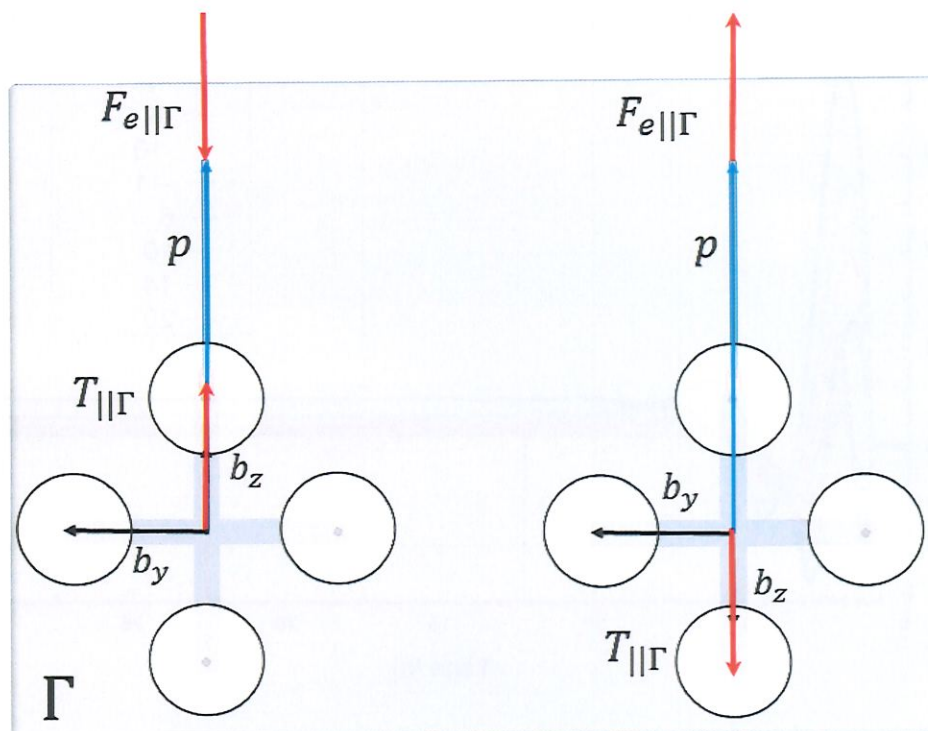


Figure 43: Set of forces on the plane Γ .

Source: Author

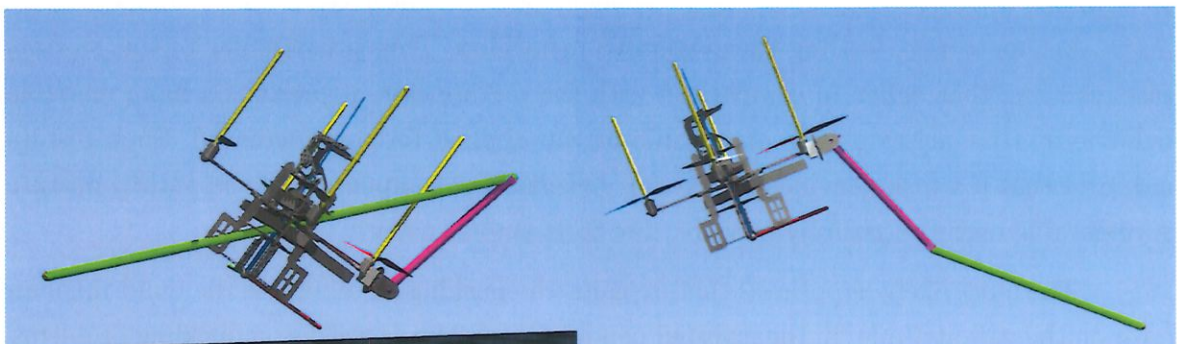


Figure 44: Simulations showing *pushing* on the left and *pulling* on the right.

Source: Author



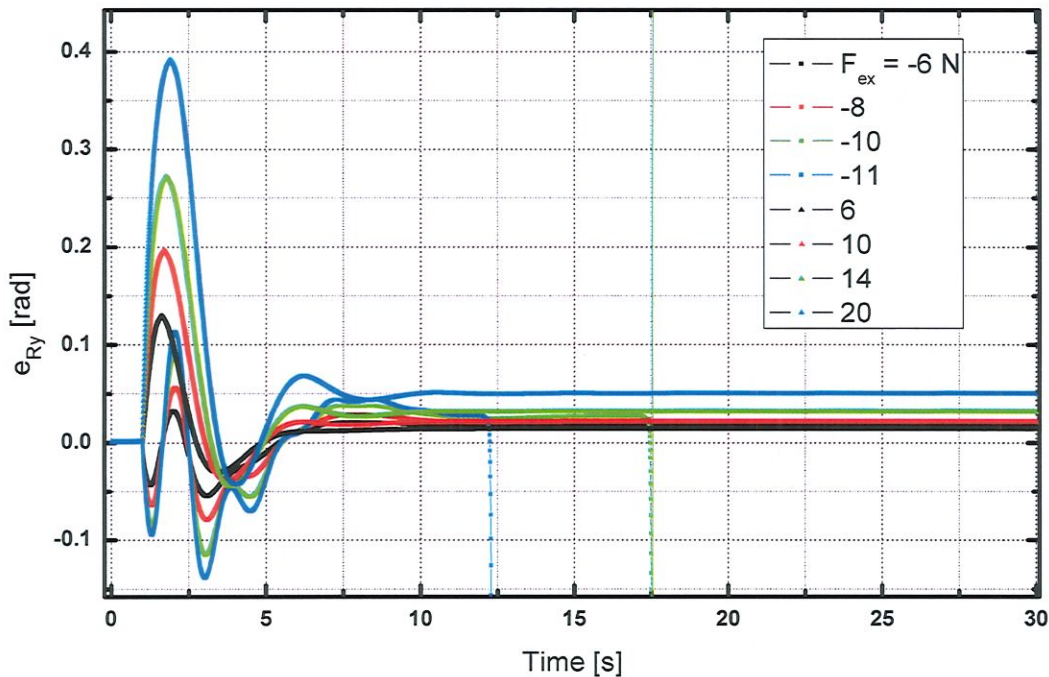


Figure 45: Attitude error for direction y across time, for several applied forces on direction x .

Source: Author

simulation time. The applied force is $\vec{F}_e = \{-8, 0, 0\}$. Notice that the system seems to stabilize after the step entry \vec{F}_e is applied, but as time passes, the \hat{z} component of $\vec{p} \times \vec{F}_e$ begins to rise, leading the system to instability. This is particular of the *pushing* case, showing that, when in comparison with the *pulling* case, it presents a more unstable behaviour, that arises as the magnitude of the applied force is increased. Such results indicates that if a task demands a large force exertion or compensation, the *pulling* posture is preferable over the *pushing* posture, due to its stabler nature.

The most likely hypothesis that explains the pushing instability is the build-up of an error on the yaw attitude. In the inverted pendulum analogy as well as on buckling problems, the miss-alignment of direction between the reaction force and the force being exerted on the other end of the body will cause a reaction torque that if not compensated, leads the system to instability. Besides this notion being intuitive, the simulations corroborate with the hypothesis. Nevertheless, this instability can be managed by means of a properly designed controller. Since the instability arises from a miss-alignment build-up, which can also be seen as a kind of a steady-state error, here, this problem was solved by simply

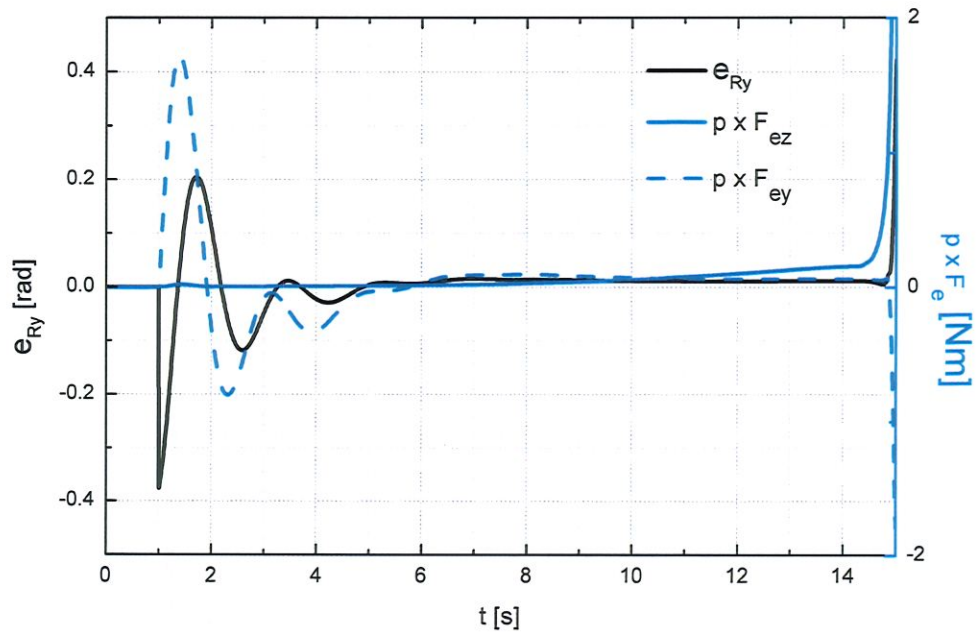


Figure 46: Simulation of a *pushing* case that goes unstable.

Source: Author

adding an integral term for the z direction of the attitude controller. Figure 47 shows an extreme case where the applied force is $\vec{F}_e = \{-14, 0, -3\}$ and the system had no difficulties on compensating it. Also, to illustrate the robustness of the controller and its ability to maintain \vec{F}_e on plane Ω , the applied force was rotated 360 degrees in 3 seconds, and yet the system remained stable.

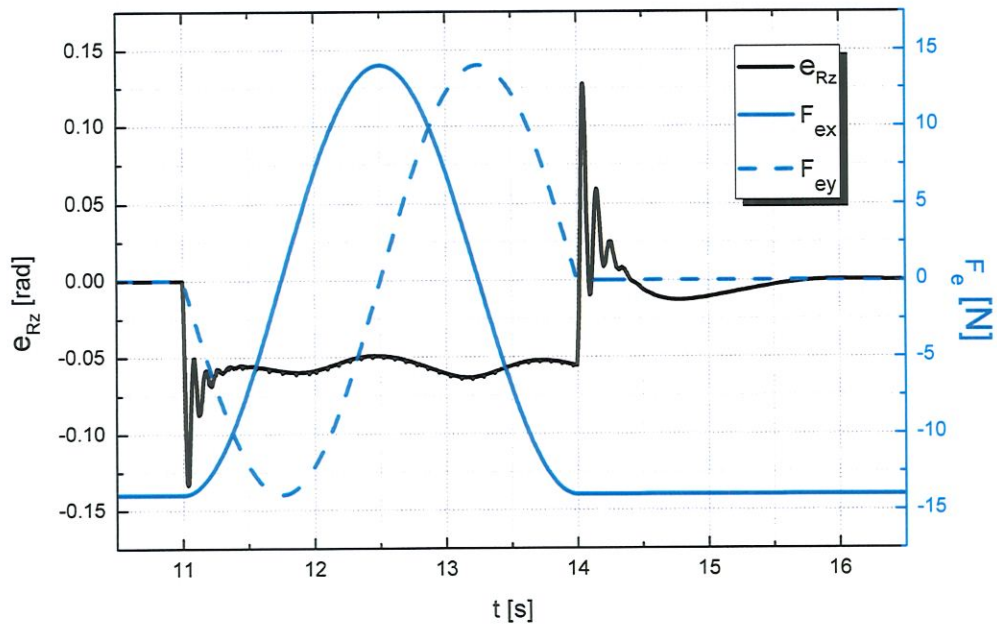


Figure 47: Simulations of a extreme *pushing* case with integral term on attitude controller, where the direction of the applied force is rotated 360 degrees in a 3 second time interval.

Source: Author

7 CONCLUSION

In this work, a new view of the control problems found in aerial manipulation was presented. The author identifies three main scenarios which require different approaches from the control perspective. Chapter 4 address a preliminary situation called here Scenario Zero. Such a scenario regards the response of a UAV quadrotor in hovering state under a position controller and subjected to torques applied on the axis of its horizontal plane. It is shown that a displaced CG has this same effect on an aerial manipulator and that the system responds with a stationary error in the position that can be satisfactorily treated as linearly proportional to the applied net torque.

In Chapter 5 we present a control strategy for Scenario One, Pick-and-placing, based on the insights provided by Chapter 4 and its study of Scenario Zero. The proposed framework for aerial manipulation is based on linear compensations of torques and forces through the use of six-axis F/T force sensor between the UAV and its robotic arm. Simulation results showed a system that is effective for pick-and-place tasks with a payload representing approximately 10% of the whole system weight. The control framework proved to be simple but powerful, handling unknown payloads without great difficulties.

Chapter 6 presented a theoretical approach to achieve stability of a quadrotor aerial manipulator subjected to generalized forces acting on its EEF. This situation represents Scenario Two partially, but nevertheless, it addresses its most relevant part. A geometrical analysis was used to define the two existent equilibrium points of the presented model, naming then *pushing* and *pulling*. Furthermore, the same analysis allowed to formalize the intuitive notion that to achieve stability, *pushing* efforts demands that the EEF must be positioned above the quadrotor's CM, and that for *pulling* efforts, it must be positioned below the CM. An important observation was made concerning the stability of the two equilibrium points, revealing that the *pushing* case is more unstable than the *pulling* case, in the same manner, that the upward position of a vertical pendulum is unstable, while the downward position is stable. A controller was proposed and evaluated under simulations, proving to be stable for both of the equilibrium points under the substantial magnitude of applied forces. Simulations also confirmed the statements concerning the stability of the two equilibrium points. The findings discussed here can be easily generalized to other aerial manipulators designs, providing better capabilities to such systems. Possible direct applications for future works are liquid spraying tasks, such as surface cleaning and painting.

7.1 Future Works

The next logical step to continue developing the works presented here is to prove it's concepts with real-world experiments. In order to do that, a reasonable approach is for one to follow the steps in the order considered here. Starting by addressing Scenario Zero and obtaining the curve of position error versus externally applied torques to later apply what is learned in Scenario One. Maybe not so straight forward is to implement Scenario Two as presented in Chapter 6. The author suggests at least two ways it could be done. One is by employing a liquid spray device at the end-effector of an aerial manipulator, as implied at the end of the above paragraph. A second and maybe cleaner way to test it in laboratory conditions is to use an additional rotor also at the end-effector of the aerial manipulator. The modeling of rotors are very well established and therefore it would be a more straight forward option.

An additional and more challenging option is to address Scenario Three, which is not covered in this work. This is a complex situation from the control point of view since when the aerial manipulator became constrained to a point in space, it's dynamical model changes and it no longer rotates around its center of mass. As mentioned in the literature review, researches have addressed this is at least two ways. One by switching controllers after contact ([WOPEREIS et al., 2017](#)) and another by employing indirect force control, such as impedance ([GIOIOSO et al., 2014](#)). Both approaches have their respective advantages and disadvantages. However, to the author's best knowledge, no published work in aerial manipulation has yet managed to fully characterize the dynamics of such systems and proposed a generic and comprehensive solution for it. There is still much to be developed in the field.

BIBLIOGRAPHY

- AEROWORKS. 2018. Available from the Internet: <<http://www.aeroworks2020.eu/>>.
- AIROBOTS. 2018. Available from the Internet: <<http://airobots.dei.unibo.it/>>.
- ARCAS. 2018. Available from the Internet: <<http://www.arcas-project.eu/>>.
- BAIZID, K. et al. Behavioral control of unmanned aerial vehicle manipulator systems. **Autonomous Robots**, Springer US, v. 41, n. 5, p. 1203–1220, 2017. ISSN 15737527.
- Bicchi, A.; Kumar, V. Robotic grasping and contact: a review. In: **Proceedings 2000 ICRA. Millennium Conference. IEEE International Conference on Robotics and Automation. Symposia Proceedings (Cat. No.00CH37065)**. [S.l.: s.n.], 2000. v. 1, p. 348–353 vol.1. ISSN 1050-4729.
- BISGAARD, M.; COUR-HARBO, A. la; BENDTSEN, J. D. Adaptive control system for autonomous helicopter slung load operations. **Control Engineering Practice**, v. 18, n. 7, p. 800–811, 2010.
- Bouabdallah, S.; Becker, M.; Siegwart, R. Autonomous miniature flying robots: coming soon! - research, development, and results. **IEEE Robotics Automation Magazine**, v. 14, n. 3, p. 88–98, Sep. 2007. ISSN 1070-9932.
- Buzzatto, J. P. S. et al. Aerial manipulation with six-axis force and torque sensor feedback compensation. In: **2018 Latin American Robotic Symposium, 2018 Brazilian Symposium on Robotics (SBR) and 2018 Workshop on Robotics in Education (WRE)**. [S.l.: s.n.], 2018. p. 158–163.
- _____. Quadrotor aerial manipulator: Modeling and control under substantial generalized 3d forces. In: **Manuscript submitted to 2019 Robotics: Science and Systems**. [S.l.: s.n.], 2019.
- COUMANS, E. Bullet physics simulation. In: **ACM SIGGRAPH 2015 Courses**. New York, NY, USA: ACM, 2015. (SIGGRAPH '15). ISBN 978-1-4503-3634-5. Available from the Internet: <<http://doi.acm.org/10.1145/2776880.2792704>>.
- DANKO, T. W.; OH, P. Y. Design and control of a hyper-redundant manipulator for mobile manipulating unmanned aerial vehicles. **Journal of Intelligent and Robotic Systems: Theory and Applications**, v. 73, n. 1-4, p. 709–723, 2014. ISSN 15730409.
- DING, X. et al. A review of aerial manipulation of small-scale rotorcraft unmanned robotic systems. **Chinese Journal of Aeronautics**, v. 32, n. 1, p. 200 – 214, 2019. ISSN 1000-9361. Available from the Internet: <<http://www.sciencedirect.com/science/article/pii/S1000936118301894>>.
- EREZ, T.; TASSA, Y.; TODOROV, E. Simulation tools for model-based robotics: Comparison of Bullet, Havok, MuJoCo, ODE and PhysX. **Proceedings - IEEE International Conference on Robotics and Automation**, v. 2015-June, n. June, p. 4397–4404, 2015. ISSN 10504729.

EUROPE, T. P. for Robotics in. **The Strategic Research Agenda**. 2013. Available from the Internet: <http://roboproject.h2214467.stratoserver.net/cms/upload/PPP/SRA2020_SPARC.pdf>.

FANNI, M.; KHALIFA, A. A new 6-DOF quadrotor manipulation system: Design, kinematics, dynamics, and control. **IEEE/ASME Transactions on Mechatronics**, v. 22, n. 3, p. 1315–1326, 2017. ISSN 10834435.

FINK, J. et al. Planning and control for cooperative manipulation and transportation with aerial robots. **The International Journal of Robotics Research**, v. 30, n. 3, p. 324–334, 2011. ISSN 16107438.

FUMAGALLI, M.; STRAMIGIOLI, S.; CARLONI, R. Mechatronic design of a robotic manipulator for Unmanned Aerial Vehicles. **2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)**, p. 4843–4848, 2016.

GIOIOSO, G. et al. Turning a near-hovering controlled quadrotor into a 3D force effector. **Proceedings - IEEE International Conference on Robotics and Automation**, p. 6278–6284, 2014. ISSN 10504729.

GOERZEN, C.; KONG, Z.; METTLER, B. A survey of motion planning algorithms from the perspective of autonomous uav guidance. **Journal of Intelligent and Robotic Systems**, v. 57, n. 1, p. 65, Nov 2009. ISSN 1573-0409. Available from the Internet: <<https://doi.org/10.1007/s10846-009-9383-1>>.

GOODARZI, F. A.; LEE, T. Dynamics and control of quadrotor uavs transporting a rigid body connected via flexible cables. In: **Amecan Control Conference**. [S.l.: s.n.], 2015. p. 4677–4682.

HAVOK, P. E. 2018. Available from the Internet: <<https://www.havok.com>>.

HEREDIA, G. et al. Control of a multirotor outdoor aerial manipulator. **IEEE International Conference on Intelligent Robots and Systems**, n. Iros, p. 3417–3422, 2014. ISSN 21530866.

Hua, M. et al. Introduction to feedback control of underactuated vtolvehicles: A review of basic control design ideas and principles. **IEEE Control Systems Magazine**, v. 33, n. 1, p. 61–75, Feb 2013. ISSN 1066-033X.

JAIN, A.; KEMP, C. C. Pulling open doors and drawers: Coordinating an omni-directional base and a compliant arm with equilibrium point control. In: **Proceedings - IEEE International Conference on Robotics and Automation**. [S.l.: s.n.], 2010. p. 1807–1814. ISBN 9781424450381. ISSN 10504729.

JIANG, Q.; KUMAR, V. The inverse kinematics of cooperative transport with multiple aerial robots. **IEEE Transactions on Robotics**, v. 29, n. 1, p. 136–145, 2013. ISSN 15523098.

JIMENEZ-CANO, A. E. et al. Control of an aerial robot with multi-link arm for assembly tasks. **Proceedings - IEEE International Conference on Robotics and Automation**, p. 4916–4921, 2013. ISSN 10504729.

KAMEL, M.; COMARI, S.; SIEGWART, R. Full-body multi-objective controller for aerial manipulation. **24th Mediterranean Conference on Control and Automation, MED 2016**, p. 659–664, 2016.

KARAYIANNIDIS, Y. et al. Adaptive force/velocity control for opening unknown doors. In: **IEEE/RSJ International Conference on Intelligent Robots and Systems**. [S.l.: s.n.], 2012. p. 4040–4047. ISBN 9783902823113. ISSN 14746670.

_____. Model-free robot manipulation of doors and drawers by means of fixed-grasps. In: **Proceedings - IEEE International Conference on Robotics and Automation**. [S.l.: s.n.], 2013. p. 4485–4492. ISBN 9781467356411. ISSN 10504729.

KHALIFA, A.; FANNI, M. A new quadrotor manipulation system: Modeling and point-to-point task space control. **International Journal of Control, Automation and Systems**, v. 15, n. 3, p. 1434–1446, 2017. ISSN 20054092.

KHAMSEH, H. B.; JANABI-SHARIFI, F.; ABDESSAMEUD, A. Aerial manipulation—a literature survey. **Robotics and Autonomous Systems**, v. 107, p. 221 – 235, 2018. ISSN 0921-8890. Available from the Internet: <http://www.sciencedirect.com/science/article/pii/S0921889017305535>.

KIM, S.; CHOI, S.; KIM, H. J. Aerial manipulation using a quadrotor with a two DOF robotic arm. **IEEE International Conference on Intelligent Robots and Systems**, p. 4990–4995, 2013. ISSN 21530858.

KIM, S. et al. Vision-guided aerial manipulation using a multirotor with a robotic arm. **IEEE/ASME Transactions on Mechatronics**, v. 21, n. 4, p. 1912–1923, 2016. ISSN 10834435.

KIM, S.; SEO, H.; KIM, H. J. Operating an unknown drawer using an aerial manipulator. **Robotics and Automation (ICRA), 2015 IEEE International Conference on**, p. 5503–5508, 2015. ISSN 1050-4729.

KONDAK, K. et al. Aerial manipulation robot composed of an autonomous helicopter and a 7 degrees of freedom industrial manipulator. **Proceedings - IEEE International Conference on Robotics and Automation**, p. 2107–2112, 2014. ISSN 10504729.

LEE, T. et al. Geometric Tracking Control of a Quadrotor UAV on $SE(3)$. n. 3, p. 5420–5425, 2010.

LEE, T.; LEOKY, M.; MCCLAMROCH, N. Geometric tracking control of a quadrotor UAV on $SE(3)$. **49th IEEE Conference on Decision and Control (CDC), 2010.**, p. 5420–5425, 2010. ISSN 0743-1546. Available from the Internet: <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=5717652>.

LEISHMAN, J. G. **Principles of helicopter aerodynamics**. 2. ed. [S.l.]: Cambridge, UK: Cambridge University Press, 2006.

LINDSEY, Q.; MELLINGER, D.; KUMAR, V. Construction of cubic structures with quadrotor teams. In: **Robotics: Science and Systems**. [S.l.: s.n.], 2011.

_____. Construction with quadrotor teams. **Autonomous Robots**, v. 33, n. 3, p. 323–336, 2012. ISSN 09295593.

LIPPIELLO, V.; RUGGIERO, F. Cartesian impedance control of a UAV with a robotic Arm. **IFAC Proceedings Volumes (IFAC-PapersOnline)**, v. 10, n. PART 1, p. 704–709, 2012. ISSN 14746670.

MEBARKI, R.; LIPPIELLO, V.; SICILIANO, B. Toward image-based visual servoing for cooperative aerial manipulation. **Proceedings - IEEE International Conference on Robotics and Automation**, v. 2015-June, n. June, p. 6074–6080, 2015. ISSN 10504729.

MELLINGER, D. et al. Design, modeling, estimation and control for aerial grasping and manipulation. **IEEE International Conference on Intelligent Robots and Systems**, p. 2668–2673, 2011. ISSN 2153-0858.

MICHAEL, N.; FINK, J.; KUMAR, V. Cooperative manipulation and transportation with aerial robots. **Auton Robot**, v. 30, p. 73–86, 2011. ISSN 16107438.

MICROSOFT. 2018. Available from the Internet: <<https://www.visualstudio.com>>.

MUJOCO.ORG, H. 2018. Available from the Internet: <<http://www.mujoco.org>>.

NGUYEN, H. N.; HA, C.; LEE, D. Mechanics, control and internal dynamics of quadrotor tool operation. **Automatica**, Elsevier Ltd, v. 61, p. 289–301, 2015. ISSN 00051098. Available from the Internet: <<http://dx.doi.org/10.1016/j.automatica.2015.08.015>>.

NIKOU, A.; GAVRIDIS, G. C.; KYRIAKOPOULOS, K. J. Mechanical Design, Modelling and Control of a Novel Aerial Manipulator. **IEEE International Conference on Robotics and Automation**, p. 4698–4703, 2015. ISSN 1050-4729.

PALUNKO, I.; CRUZ, P.; FIERRO, R. Agile load transportation : Safe and efficient load manipulation with aerial robots. **IEEE Robotics and Automation Magazine**, v. 19, n. 3, p. 69–79, 2012. ISSN 10709932.

PAPACHRISTOS, C.; ALEXIS, K.; TZES, A. Efficient Force Exertion for Aerial Robotic Manipulation : Exploiting the Thrust – Vectoring Authority of a Tri – TiltRotor UAV. In: **2014 IEEE International Conference on Robotics & Automation (ICRA)**. [S.l.: s.n.], 2014. p. 4500–4505. ISBN 9781479936854.

PELICAN, A. 2018. Available from the Internet: <<http://www.asctec.de/>>.

PHYSX, P. E. 2018. Available from the Internet: <<http://www.nvidia.com/object/physx-9.17.0524-driver.html>>.

POUNDS, P.; MAHONY, R.; CORKE, P. . Modelling and control of a quadrotor robot. In: **Proceedings of the Australasian conference on robotics and automation**. [S.l.: s.n.], 2006.

POUNDS, P.; MAHONY, R.; CORKE, P. Modelling and control of a large quadrotor robot. **Control Engineering Practice**, Elsevier, v. 18, n. 7, p. 691–699, 2010. ISSN 09670661. Available from the Internet: <<http://dx.doi.org/10.1016/j.conengprac.2010.02.008>>.

POUNDS, P. E.; BERSAK, D. R.; DOLLAR, A. M. Grasping from the air: Hovering capture and load stability. **Proceedings - IEEE International Conference on Robotics and Automation**, p. 2491–2498, 2011. ISSN 10504729.

- PROUTY, R. W. **Helicopter Performance, Stability, and Control**. 1. ed. [S.l.]: Krieger Publishing Company, 2002. 469 p. ISBN 1575242095.
- Ruggiero, F.; Lippiello, V.; Ollero, A. Aerial manipulation: A literature review. **IEEE Robotics and Automation Letters**, v. 3, n. 3, p. 1957–1964, July 2018. ISSN 2377-3766.
- RUGGIERO, F.; LIPPIELLO, V.; OLLERO, A. Introduction to the special issue on aerial manipulation. **IEEE ROBOTICS AND AUTOMATION LETTERS**, v. 3, 2018.
- RYLL, M. et al. 6D physical interaction with a fully actuated aerial robot. **Proceedings - IEEE International Conference on Robotics and Automation**, p. 5190–5195, 2017. ISSN 10504729.
- SHIMAHARA, S. et al. Aerial manipulation for the workspace above the airframe. **IEEE International Conference on Intelligent Robots and Systems**, v. 2015-Decem, p. 1453–1458, 2015. ISSN 21530866.
- _____. Aerial torsional manipulation employing multi-rotor flying robot. **IEEE International Conference on Intelligent Robots and Systems**, v. 2016-Novem, p. 1595–1600, 2016. ISSN 21530866.
- SICILIANO, B. et al. **Robotics: Modeling, Planning and Control**. [S.l.]: Springer, 2008. 96 p. ISSN 0009-4978. ISBN 9781852339944.
- SICILIANO, B.; VILLANI, L. **ROBOT FORCE CONTROL**. [S.l.: s.n.], 1999. v. 540. 146 p. ISBN 978-0-7923-7733-7.
- SMITH, R. Ode: Open dynamics engine. 01 2009.
- SOLIDWORKS. 2018. Available from the Internet: <<https://www.solidworks.com/>>.
- SUAREZ, A.; HEREDIA, G.; OLLERO, A. Lightweight Compliant Arm for Aerial Manipulation. In: **2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)**. [S.l.: s.n.], 2015. p. 1627–1632. ISBN 9781479999941.
- TODOROV, E.; EREZ, T.; TASSA, Y. MuJoCo: A physics engine for model-based control. **IEEE International Conference on Intelligent Robots and Systems**, p. 5026–5033, 2012. ISSN 21530858.
- WOPEREIS, H. W. et al. Application of substantial and sustained force to vertical surfaces using a quadrotor. **Proceedings - IEEE International Conference on Robotics and Automation**, p. 2704–2709, 2017. ISSN 10504729.
- WOS. **Web of Science**. 2019. Available from the Internet: <<https://webofknowledge.com/>>.
- WUTHIER, D. et al. A Geometric Pulling Force Controller for Aerial Robotic Workers. **IFAC-PapersOnLine**, Elsevier B.V., v. 50, n. 1, p. 10287–10292, 2017. ISSN 24058963. Available from the Internet: <<https://doi.org/10.1016/j.ifacol.2017.08.1487>>.