Development of a Robust and Combined Controller for On-Orbit Servicing Missions

Jean-Daniel Crépy-Marglais

Technische Universiteit Delft

Development of a Robust and Combined Controller for On-Orbit Servicing Missions

by

Jean-Daniel Crépy-Marglais

in partial fulfillment of the requirements for the degree of

Master of Science
in Aerospace Engineering

at the Delft University of Technology,
to be defended publicly on Thursday December 13, 2018 at 9:00 AM.

Student number:	4600045	
Project duration:	April 17, 2018 - December 13, 2018	
Supervisors:	Dr. ir. R. Fónod,	TU Delft
	Ir. R. Krenn	DLR, Wessling, Germany
Thesis committee:	Dr. ir. A. Cervone,	TU Delft
	Dr. ir. A. Schiele	TU Delft and ESA
	Dr. ir. R. Fónod,	TU Delft
	Ir. R. Krenn	DLR, Wessling, Germany

An electronic version of this thesis is available at `http://repository.tudelft.nl/`.
Cover image credits: D. Sheppard and
`https://www.nasa.gov/content/view-of-moon-from-space-station`

Abstract

Nowadays, a satellite is considered as out of range forever once it is launched. This may soon no longer hold thanks to on-orbit servicing. However, the complexity of that topic makes it highly challenging on many aspects. On-orbit servicing using a robotic arm is nonetheless the most investigated technology in order to achieve the berthing of two satellites in orbit. This research project aims at developing a robust controller for the deployment of a robotic arm, which is mounted on a servicer satellite, and for the reaching of a defined handle on the target satellite under mission constraints. While operating the space manipulator, the combined control of both the manipulator and the spacecraft base has a great potential in terms of achievable performance and may bring robotic on-orbit servicing one step closer to reality. In order to reveal the potential of this innovative control strategy, a model of the system dynamics was developed, before being used to design a robust controller. The latter was then successfully implemented in different simulators to test its performance and robustness to a variety of constraints including hardware, environment and mission constraints. The simulations proved that combined control can be used for the entire maneuver. Finally, this controller was successfully implemented in the on-board computer of the on-orbit servicing simulator at the German Aerospace Center. From now on, the maneuver can be completed from the ground control room. The implementation of the robust and combined controller has increased the technology readiness level of the reach phase during an on-orbit servicing mission.

Preface

This thesis was completed during my Master of Science in aerospace engineering at the Delft University of Technology, more precisely within the Space Engineering department.
I have always been fascinated by space flight and during my studies, I developed a real interest for attitude and orbit dynamics and control systems. With the growing interest that the space industry is generating, the number of satellites on orbit around the Earth keeps increasing. With it, the number of space debris orbiting our planet is soaring. As an aerospace engineering student, I want to make my knowledge and my efforts available for a meaningful purpose. On-orbit servicing appeared naturally as a perfect match for my interests.
As written in my application to Rainer Krenn, who later became one of my MSc thesis supervisors, I was looking for a challenging and innovative project on on-orbit servicing. A few months later, I started working at the DLR Institute of System Dynamics and Control, Department of Space System Dynamics, as part of an on-orbit servicing technology development project called RICADOS. Joining this team was a great opportunity. I was able to study theoretical approaches I have never investigated before, to apply these theories to a real system and finally to witness and assess their effects. Having the chance to have such an overview of the whole research process was really interesting, especially because scientific research was new to me.
There is no doubt: I have learned a lot during my stay in Oberpfaffenhofen (good luck with that one), Germany. This would have not been possible for sure without the help of my supervisors on both sides of the Rhine. Thus, I would like to deeply thank Rainer Krenn and Róbert Fónod for supervising my work, for supporting me and for their enthusiasm no matter the challenges being faced along the way. I wish to also thank everyone at DLR, in particular the team of the RICADOS project for their willingness to answer my questions as well as all my fellow students I had the pleasure to share an office with. Finally, I would like to thank my family for their unconditional support during this thesis but also at every step taken to get there.

Jean-Daniel Crépy-Marglais
Munich, December 2018

Contents

List of Figures

List of Tables

List of Acronyms

AOCS	Attitude and Orbit Control System, 4, 6–8, 19, 20, 36, 48, 63, 66, 67, 74, 76, 79, 80, 84
BF	Body-Fixed, 14, 16, 17, 24, 38, 48
CCSDS	Consultative Committee for Space Data Systems, 76, 79
CoM	Center of Mass, 11, 14, 16, 17, 24, 25, 33, 38, 50, 69, 88, 90
DEOS	Deutsche Orbitale Servicing, 3, 4, 11
DH	Denavit-Hartenberg, 15, 24
DLR	Deutsches Zentrum für Luft- und Raumfahrt e.V., 1–5, 20, 21, 29, 38, 43, 73, 75, 76, 79, 84, 87, 90
DoF	Degree(s) of Freedom, 4, 25, 34, 35, 40, 69, 91
DYSI	Dynamic Simulator, 4, 76, 84
E2E	End-to-End, 4, 5, 20, 73, 76, 80, 82–84, 87
ECI	Earth-Centered Inertial, ix, 14
ECSS	European Cooperation for Space Standardization, 79
EE	End-Effector, 11, 23–25, 29, 38, 48, 68, 69, 74, 76, 79, 83
EPOS	European Proximity Operations Simulator, 4, 5, 20, 76
ESA	European Space Agency, 1, 2, 20
FD	Forward Dynamics, 30
FDIR	Fault Detection, Isolation and Restoration, 5
FK	Forward Kinematics, ix, 24, 29, 76
GNC	Guidance, Navigation and Control, 4
GSOC	German Space Operations Center, 76, 79, 87
HW	Hardware, 76, 90
ID	Identification Name, 79
IK	Inverse Kinematics, ix, 24, 29
I/O	Input/Output, 40, 53, 56
IRF	Inertial Reference Frame, 24, 25
ISS	International Space Station, 3
LEO	Low Earth Orbit, ix, 1, 2
LiDAR	Light Detection And Ranging, 5
LPV	Linear Parameter-Varying, 44
LQ	Linear Quadratic, 44
LQG	Linear Quadratic Gaussian, 44
LTR	Loop Transfer Recovery, 44
LVLH	Local-Vertical Local-Horizontal, ix, 14, 16, 17, 38, 48, 66
LWR	Light-Weight Robot, 4, 11, 18, 71, 88
MIB	Mission Information Base, 79, 80
MIMO	Multiple-Input Multiple-Output, 44
NASA	National Aeronautics and Space Administration, ix, 1, 2
OBC	On-Board Computer, 8, 19, 43, 76, 82, 87
OOS	On-Orbit Servicing, xi, 1–7, 11, 14, 20, 80, 82–84, 87, 88
OOS-SIM	On-Orbit Servicing Simulator, 4, 5, 20, 76, 90
PUS	Packet Utilisation Standard, 79
QFT	Quantitative Feedback Theory, 44
RCT	Robust Control Theory, 44
RF	Reference Frame, 15, 25, 38, 68
RGA	Relative Gain Array, 40, 41, 51
RICADOS	Rendezvous, Inspection, CApturing and Detumbling by Orbital Servicing, 5, 8, 11, 19, 20, 29, 36, 43, 48, 50, 74

SASI	Satellite Simulator, 4, 76
S/C	Spacecraft, 2, 7, 8, 11, 15, 17, 23, 43, 48, 81, 84
SISO	Single-Input Single-Output, 44
SMC	Sliding Mode Control, 44
SW	Software, 76
TC	Telecommand, x, 20, 76, 79–82, 87
TCP	Tool Center Point, 15, 38
TM	Telemetry, 20, 76, 79
TRL	Technology Readiness Level, 4, 5, 7, 20, 87, 88
UDP	User Data Protocol, 76, 79

List of Symbols

$\mathbf{A}_0$	Rotation matrix from the base BF RF to the LVLH	-
$\mathbf{A}$	State matrix	-
$\mathbf{B}$	Input matrix	-
$\mathbf{C}$	Output matrix	-
C	Center of the workspace	-
$\mathbf{Co}$	Controllability matrix	-
$\mathbf{D}$	Feedforward matrix	-
$\boldsymbol{\delta}_0$	Euler angles vector	rad
$\mathbf{e}$	Error vector	-
G	Center of mass of the servicer	-
$\mathbf{G}$	Plant	-
$\mathbf{H}$	Total mass matrix of the system	-
$\mathbf{J}_G^B$	Inertia tensor of the base at ist CoM	$\mathrm{kg} \cdot \mathrm{m}^2$
$\mathbf{K}$	Controller	-
L	Non-relativistic Lagrangian	J
$\mathbf{L}_i$	Distance from the input to the output of link i	m
m	Mass of the servicer base	kg
N_m	Number of links of the arm	-
$\mathbf{O}$	Observability matrix	-
$\boldsymbol{\omega}$	Angular velocity	$\mathrm{rad} \cdot \mathrm{s}^{-1}$
$\mathbf{p}_{EE}$	Position of the end-effector	m
$\mathbf{p}_{Handle}$	Position of the handle	m
P	Point where the arm and the base are linked	-
$\mathbf{P}$	Augmented plant	-
$\mathbf{q}$	Position vector	-
$\mathbf{Q}$	Vector of generalized forces	-
$\mathbf{r}$	Reference vector	-
$\mathbf{R}_{C_0}$	Position of the CoM of the target in the LVLH	m
$\mathbf{RGA}$	Relative gain array matrix	-
$\mathbf{S}$	Sensitivity function	-

$\boldsymbol{\sigma}$	Singular values vector	-
$\mathbf{T}$	Kinetic energy	J
$\boldsymbol{\theta}_j$	Robotic arm joint angles	rad
$\mathbf{u}$	Control vector	-
$\mathbf{v}$	Velocity vector	$\mathrm{m} \cdot \mathrm{s}^{-1}$
$\mathbf{W}_i$	i^{th} weighting function	-
$\mathbf{x}$	State vector	-
$\mathbf{y}$	Output vector	-

1

Introduction

In this chapter, the topic of the thesis is introduced. First of all, a general overview of on-orbit servicing (OOS) is detailed in Section 1.1 followed by a description of this field as well as of space robotics more particularly at DLR in Section 1.2. Then, the thesis problem is stated in Section 1.3 leading to the research objective (Section 1.4). The scientific approach and methodology set up to meet this research objective is described in Section 1.5. Finally, the outline of this report is given in Section 1.6.

1.1. On-Orbit Servicing

1.1.1. Definitions

An OOS mission is a mission during which a target satellite is serviced by a servicer. The servicing mission can be completed by a human in case of a manned OOS mission or by a secondary satellite for an unmanned one. The task to be completed can be of different natures. The servicer may inspect, repair, upgrade, refuel, assemble, relocate or de-orbit the target satellite. The three main categories of OOS missions are observation, motion and manipulation [58]. The first category gathers the observation or remote inspection missions. If after launch, an unexpected phenomenon happens or if the communication between the satellite and the ground station is not possible, a mission to inspect the satellite can be launched to understand the reasons of the failure. The missions in the motion category have in common that the mission of the servicer is to bring the target from its position on its orbit to another position. It may relocate the target satellite, ensure its station-keeping or de-orbit it. Finally, the maintenance, repair, refuel, upgrade and docked inspection of a satellite are all part of the manipulation category. This classification is summed up in Table 1.1.

1.1.2. Motivations for On-Orbit Servicing

There are multiple potential applications for OOS but there are two main motivations for it. First of all, setting up an OOS mission may reduce costs dramatically if the cost of this mission is less than the cost of the mission to be serviced. Otherwise, it is more appealing to launch the same satellite once again

Table 1.1: Classification of OOS missions

Service category	Kind of service
Observation	Remote Inspection
Motion	Station-Keeping
	Relocation
	Disposal and De-Orbiting
Manipulation	Refuelling
	Maintenance
	Repair
	Docked Inspection

and save the difference which is the solution often used nowadays. Then, the second reason why OOS is interesting is to limit space traffic especially in Low Earth Orbit (LEO). Indeed, the number of satellites keeps increasing over the years. In addition to the operational satellites in orbit, many old satellites still orbit the Earth without completing any function. Adding the satellites in orbit and the other detectable objects, NASA came up with the number of more than 11500 space debris in LEO back in 2010 as shown on Figure 1.1. This figure shows clearly that the number of space debris in LEO is skyrocketing. The impact of the destruction of a satellite (such as Fengyun-1C) or the collision of two satellites (such as Iridium 33 and Cosmos 2251, for which the number of resulting debris have not yet been all cataloged) is tremendous and should under any circumstances be avoided in the future. To limit the increase of space debris in LEO, disposal scenarios of satellites must be studied with great attention from now on. OOS appears to have great potential to help with that. Thus international space agencies such as ESA or NASA have ongoing OOS projects as discussed later on in Section 1.1.3.

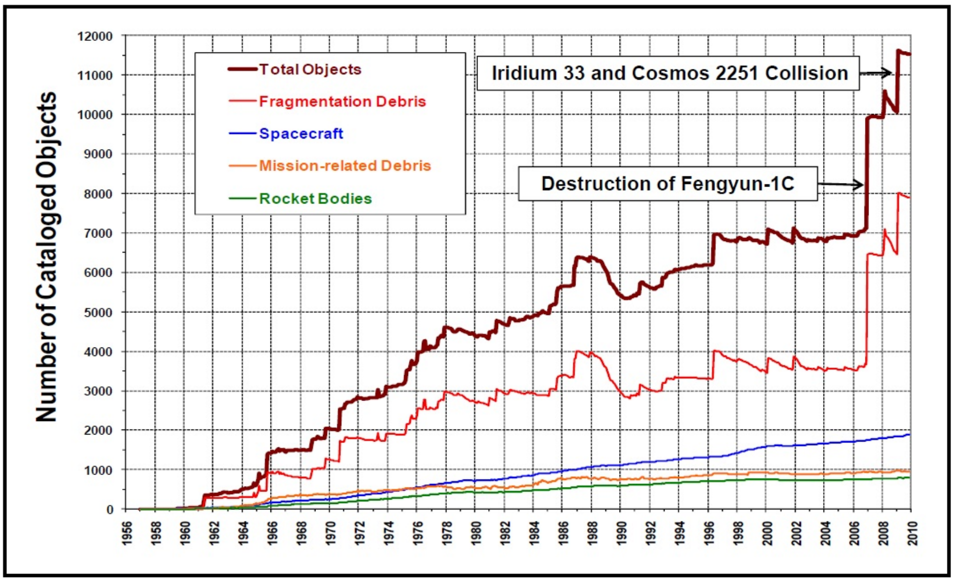


Figure 1.1: Evolution of the number of Cataloged Objects by NASA in LEO from 1956 to 2010 [35]

1.1.3. State of the Art and State of Practice

Because of the increasing level of technology and the previously discussed potential of remote-controlled OOS, there are many ongoing projects to overcome the many remaining challenges. They investigated many concepts which need further development to become reality. The main servicing strategies are the following. Both ESA's Clean Space initiative and NASA's Orbital Debris Program Office are investigating the capture of a satellite with different methods such as using a net or space robotics [63]. These programs are mainly focused on de-orbiting satellites. However, the use of robotics could also be a solution for the other types of OOS missions discussed in Section 1.1.1. Such missions are also being investigated by both space agencies. The development of new technologies through projects such as NASA Satellite Servicing Projects Division's Restore-L [56], Orbital ATK's MEV-1 [49] or DLR's End-to-End technology development projects [9] bring OOS closer to reality. Projects like these require the development of technologies in every domain. For instance, Restore-L has lead to five major technology breakthroughs: the development of an autonomous, real-time navigation system, of servicing avionics, of robotic arms, of advanced tools and of a propellant transfer system to be able to refuel the target satellite. Other capturing techniques are also being developed by several research institutes such as using harpoons or tether grippers as detailed in [54]. Once developed and implemented, these concepts must be tested. Because conditions in space are very different from those on Earth (micro-gravity, vacuum, radiations), state of the art testing facilities for OOS missions are developed to reproduce space conditions as accurately as possible. The most advanced simulators are described in [21] and greatly help the research teams in their works.

The remote-controlled OOS topic has been researched for more than thirty years but only a few technology demonstrators were launched so the state of practice remains pretty low. Among these projects, many failed such as the Experimental Servicing Satellite [29] [58] or the Orbital Life Extension Vehicle [58]. Remote-controlled OOS nonetheless experienced two main successes.

First of all, the Engineering Test Satellite No. 7 (ETS-VII) was launched in 1997 by the National Space Development Agency of Japan (NASDA) [64]. A robotic arm was mounted on the servicer satellite. The main objective of the mission was to demonstrate that it could be used on orbit to complete tasks. It was successfully used for the rendezvous/docking of the servicer with the target and some experiments were done such as teleoperation from ground as well as elementary maintenance tasks and the validation on orbit of a new control strategy of the spacecraft with the robotic arm mounted on it. Two controllers were implemented separately, one was dedicated to the arm and the other to the control of the spacecraft but a link was made between the two subsystems. The two subsystems could exchange their anticipated states and take the other's into consideration as disturbance. This is officially called collaborative control by ESA [1] (but is also called cooperative [47] or coordinated control [64] [48]). It is part of a larger category of OOS control strategies called free-flying configuration. It consists in methods for which the control of the spacecraft (S/C) and of the space manipulator are somehow coupled. The lifetime of the ETS-VII mission was extended and other experiments were suggested by different research teams and completed in orbit. Following this first achievement in robotic OOS, technological progress continued and the next major step in OOS practice happened with the success of the Orbital Express mission of the United States DARPA [22]. Two satellites (the target and the servicer) were launched in 2007 for this mission. The rendezvous and docking activities were done autonomously and maintenance operations on the target satellite such as refueling and battery replacement were completed successfully [22]. Unlike the ETS-VII mission, the control of the robotic arm and the spacecraft were completely separated for this mission. It means that as soon as the robotic arm was operating, the control of the S/C base was turned off and turned on again when the arm was not performing any action anymore. This control method is the easiest to implement but has limited precision. Indeed, when in orbit, the base of the satellite is not fixed and according to Newton's third law of motion, if the robotic arm is moving, an effort in the opposite direction will be transmitted to the base through the interface between the arm and the base. In this configuration - called free-floating configuration - this movement is not compensated and the action performed will thus loose in precision. This technique may be used to complete the mission under tight consumption requirements.

These two major successes are essential milestones of robotic OOS and pushed the limits further by completing multiple tasks that have never been done before, thus paving the way for future ones. Because the robotic OOS state of practice is limited as described previously, only these two control strategies have been studied. However, other techniques are studied and may give promising results for future missions. One of them is called combined control. It is another technique for free-flying S/C and consists in a unique controller implemented to control the system composed of the spacecraft and the arm as one. It is the most complex technique but it is likely to lead to better performances. More details about this technique will be given later, in Section 5.1.

1.2. Space Robotics and OOS at DLR

For many years now, the German Aerospace Center has studied space robotics and OOS and has, project after project, pushed the limits further by developing new technologies.

1.2.1. Previous missions

DLR itself has acquired knowledge and know-how specific to OOS. The space robotics study started in 1993 at DLR with the ROTEX (Robot Technology Experiment) mission which flew with Spacelab-Mission D2. Its objective was to validate the different operational such as off-line programming but also using teleoperation from ground. It was the first remotely controlled robot in space [28]. In 2004, the ROKVISS (RObotic Components Verification on the ISS) experiment was set up on the ISS. Its objectives was to test and validate state of the art space robotics and especially the remote control of a robotic arm from Earth [39]. After the success of this 5-year mission, the next major project related to OOS was DEOS (Deutsche Orbitale Servicing) mission [62]. Its objective was to demonstrate that DLR was able to capture and move an uncooperative client satellite in LEO which was no longer operating [58]. This technology development project could have given a good solution for space debris mitigation but it was aborted. DLR also took part in other OOS missions such as the ETS-VII during which some experiments of DLR were completed on-board of the Japanese satellite (called GETEX - German Technology Experiment) [38] or OLEV [52].

1.2.2. Previous Projects

In addition to the OOS missions completed by DLR, some technology development projects were also achieved which aims were to improve the TRL of the different technologies of interest. One of these projects was based on the DEOS mission and called OOS End-to-End (E2E) Simulation [9]. Its objective was to build up a simulator for testing OOS scenarios. A control center environment was developed, including three different consoles for the control of the simulations (the rendezvous/GNC one, the docking/robotic one and the satellite console) as well as a network for high rate telemetry and telecommand. A satellite simulator was also used in which the data-handling is completed as it is in space (*i.e.* using communication protocols for instance) and a real-time capable orbit dynamics simulator for both the servicer and the target satellites was implemented. The OOS-E2E project also used the Hardware-in-the-Loop simulation facilities at DLR called EPOS (European Proximity Operations Simulator) and OOS-SIM as well as its satellite simulator (SASI). The two simulation facilities are shown on figures 1.2 and 1.3. The EPOS is used to simulate the rendezvous during an OOS mission between the target and the servicer while the OOS-SIM enables the simulation of the berthing and docking of the two satellites. They were both developed in previous projects but during the E2E technology developement project, they were used in the loop and some payload sensors and actuators were integrated such as GNC sensors on EPOS and a robotic arm and camera sensors on OOS-SIM. Finally, some innovative GNC and robotic operation methods were integrated.

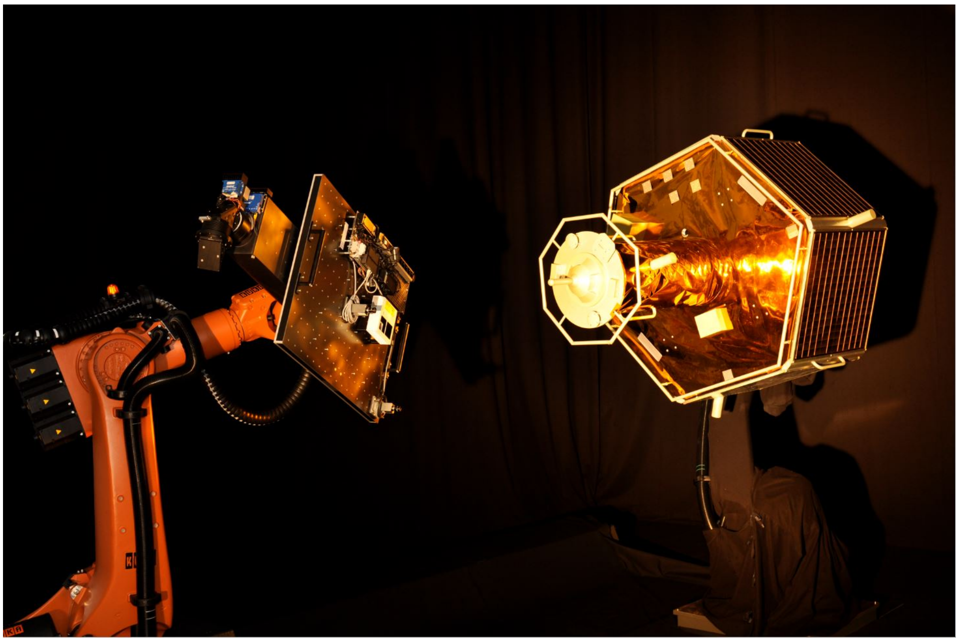


Figure 1.2: EPOS facility at DLR [9]

The aim of the OOS-SIM is to simulate the docking and berthing maneuvers in space. It means that the framework of this simulator is limited to close-range activities between the servicer and the target. As shown on Figure 1.3, it is composed of two satellite mock-ups mounted on robots of type KR120: a passive client satellite (right one on the picture) and an active servicer satellite (on the left). A representation of a space manipulator is mounted on the servicer satellite. This part of the robotic payload is a Kuka Light-Weight Robot (LWR), a seven-axes robotic arm. The end-effector of the LWR will grasp the client satellite on a defined handle mounted on it (defined position on the circular ring, see Figure 1.3) [5]. This hardware-in-the-loop simulation facility has a simulation part - the robots on which the satellites are mounted and the satellite mock-ups - and a mission part composed of the space manipulator and of the control system (e.g. camera and other sensors). Apart from the hardware part of the simulator, the orbital mechanics of both the target and the servicer are simulated using the Dynamics Simulator (DYSI). It gathers for instance multi-body/multi-physics models of the client

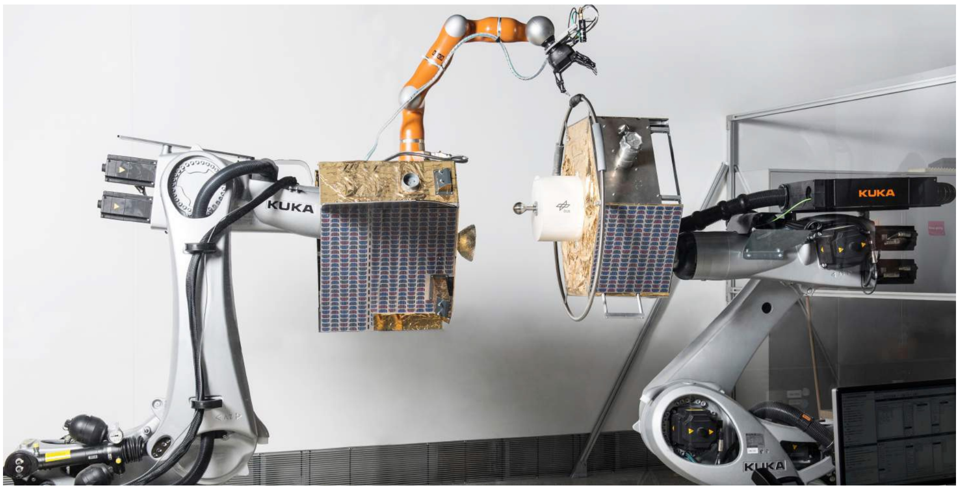


Figure 1.3: OOS-SIM facility at DLR (Credit: DLR)

satellite with a six degrees of freedom (DoF) model, of the servicer satellite with a six DoF model as well as of the seven axes space manipulator. These models, implemented on Modelica/Dymola, are called simulation-specific components. Other models are implemented in C-code such as the servicer Attitude and Orbit Control System (AOCS), several models of power and thermal subsystems, models of communication and control systems [9]. Ground testing of OOS missions is really complex mostly because of micro-gravity but hardware-in-the-loop facilities help to reproduce these conditions more accurately. Using data from the SASI, the two robots can move the satellites and reproduce their floating behaviors.

During the E2E project, the control of the servicer is separated from the control of the robotic arm. The latter has two operational modes: a telepresence mode and a mode based on visual-tracking. In the first mode, the deployment is done using a secondary identical robotic arm as haptic device on ground. The operator moves the arm and telecommands are sent to the arm mounted on the servicer to complete the same actions [57]. One of the key challenges of this mode was to handle the delay between the motion of the two manipulators. The second mode using on visual tracking is based on the processing of images taken by the cameras of the servicer satellite and aligns the end-effector with the grasping frame [9]. The berthing was not fully autonomous, no matter the mode used. Indeed, telecommands were received from ground in case of telepresence as the actions from ground were mirrored by the slave manipulator on the servicer satellite. Moreover, in the vision-based mode, high level commands are sent by the control room to the spacecraft. These commands are decomposed in elementary operations according to the task-directed programming approach developed at DLR [13]. During the whole rendezvous and berthing operation, telemetry are transmitted to ground, hold points are set up regularly and commands from ground are required to move on to the next. This organization allows the supervision and partial control of the operations from ground.

The objectives of DLR's E2E project were met at the end of 2017 after successful simulations and acceptable results [9].

1.2.3. Ongoing Project

The know-how and knowledge acquired by DLR so far have paved the way for the next innovative and challenging project called RICADOS (Rendezvous, Inspection, CApturing and Detumbling by Orbital Servicing). This technology development project aims at developing, deploying and integrating new systems and methods for on-orbit servicing missions. This includes the inspection phase, the combined control of the arm and the base as well as the implementation of an FDIR module. In the previous E2E project, the two robotic hardware-in-the-loop test facilities (the EPOS and the OOS-SIM) have been integrated in a loop to test and validate different on-orbit servicing scenarios and will be used for the recently kicked-off OOS project at DLR. RICADOS has several project goals. First of all, a reference scenario will be defined in an early stage of the project and the realization of this scenario will drive the project. Different OOS phases and techniques will be covered by the scenario to be determined such as the inspection of the target, the rendezvous and the berthing of the servicer with the target. The inspection phase will be based on sensor data using sensors on-board, a geometric model of the target satellite will be generated and examined for damage by comparing it with the expected geometry of the spacecraft. Developing new technologies and improving existing ones is necessary to be able to complete the inspection of the target. On the one hand, some of these innovations are related to OOS applications. It gathers for instance a new LiDAR (Light Detection And Ranging) configuration, a new way of processing LiDAR data and sensor combination of data from cameras and LiDAR instruments. For the rendezvous and berthing phases of the two satellites, a system will be developed to detumble and stabilize the target implying further study of contact dynamics and a high precision of the grasping. The development of a combined control strategy of the servicer and of the space manipulator will help to reach high precision. On the other hand, some innovations are purely simulation related. Indeed, the chosen scenario will also lead to adjustments of the simulation facilities (e.g. the communication system, the protocols used and even the geometry of the target satellite may change). One major development objective is also to increase the degree of autonomy of the mission while maintaining a high reliability. Thus, a Fault Detection, Isolation and Restoration (FDIR) process will be implemented. Improving the simulation facilities is absolutely crucial to increase the chance of success for a real mission. Finally, this three-year long project kicked-off in January 2018 requires many innovations and aims at increasing the TRL of OOS in a specific context. Most of the previous missions and projects completed at DLR in terms of OOS and space robotics discussed in Section 1.2.1 will be crucial.

1.2.4. Limitations and Potential

Despite its huge potential, OOS (and more precisely robotic OOS) is not yet widely accepted by the space community as being a key topic for the future of space. This reluctance has different grounds which make robotic OOS risky or simply not worthwhile [46].

First of all, OOS missions are nowadays very risky because of the criticality of the mission phases such as rendezvous, docking and berthing. If any unplanned contact between the two satellites happens, it could jeopardize the entire mission and release thousands of additional space debris. Moreover, OOS is still a research and development topic. Every time an OOS mission was launched, its success was unsure so such a mission could not at the moment be proposed for commercialization without further development. This lack of experience is mostly due to space robotics which is a relatively new topic. Because of the insufficient maturity of OOS, technical and safety regulations are missing. The lack of regulations makes the insurance and the technology transfer difficult. The current space traffic management is also not adapted for satellites which often change orbits. Some work must be done so that space industry and OOS are compatible. Finally, all the above limitations [27] lead to a cost for an OOS mission usually higher than the one of the initial mission so it is not worth it.

The limitations given above may not be true in the future thanks to research and progress of OOS technologies. Once the risks of an OOS mission and its cost are lowered, it will become valuable for private and governmental companies to extend a satellite's lifetime by refuelling, repairing or upgrading it or to de-orbit or relocate it.

1.3. Problem Statement

As exposed in Section 1.1.3, robotic OOS is a complex topic because it unifies two very different fields: robotics and AOCS. Moreover, space robotics is a very recent scientific field itself. Thus and even though OOS has a great potential for many reasons given previously, its state of practice is hardly existing with only two successful missions.

Furthermore, theoretical models were developed for robotic OOS missions, focusing on both free-floating spacecrafts and free-flying configurations. However in free-flying problems, collaborative control was thoroughly investigated while combined control is barely covered in literature. Combined control may nonetheless have great advantages. It could make the servicing operations quicker and more precise. Both of these pros are of high importance for OOS missions since the communication windows with ground can be very limited and the handle on the target satellite may be difficult to grasp. Another potential advantage of this technique is that the energy consumption of the maneuver may be greatly reduced thanks to an improve in terms of efficiency and of velocity.

Finally, there are many challenges to overcome in this field, making it not worth it yet or too risky. This thesis will tackle some of them by focusing on robust and combined control of the satellite and its robotic arm. The work presented hereafter focuses on the reach phase which is defined as the phase during which the chaser is moved in grip distance [1]. It consists in getting close enough to the target so that the manipulator can be deployed and its end-effector can reach a defined handle on the target.

1.4. Research Objective

The research objective of the thesis project is to make the reach phase under mission constraints autonomous by developing, implementing and testing a robust combined control solution which considers these constraints. The aim is to increase the technology readiness level of this maneuver using combined control. The main research question of this thesis project can be formulated as follows:

How can the TRL of the reach phase of an OOS mission be increased by implementing a robust and combined controller of the arm and of the spacecraft?

To meet the research objective and answer the main research question, many challenges and obstacles have to be overcome. These can be stated as sub-goal answering research sub-questions. The accomplishment of all sub-goals will imply that the main goal is met. They are from various origins. Some of them are related to the general topic of OOS while others are specifically linked to the combined control strategy for instance.

- What model of the system shall be developed and used to design the controller?
  The high complexity of an OOS mission and especially of the chaser spacecraft is difficult to handle. Hence its modelling is a challenging but absolutely crucial task to be able to design a controller. The level of complexity of the model to consider is important too. It shall give a good understanding of the system behavior without being too cumbersome. The first sub-goal is to develop a model of the system.

- What constraints undergone by the system during the reach phase shall be considered during the design of the controller?
  The constraints related to space impose restrictions to methods and solutions which can be used. For example, any object in space must be partly autonomous because the control with ground cannot continuously be achieved and must be able to undergo very harsh conditions due to extreme temperatures or radiations. Finally, due to the limited experience that humans have of space, there are many uncertainties in every space mission. Thus, the challenge of robustness is also crucial in this project. To answer this research sub-question, an overview of the constraints experienced by such a system during its mission will be gained. Not all these constraints can be considered during the design of a controller. The ones to be taken into account during the thesis will be discussed.

- What is the most adapted control technique to this problem?
  Considering combined control of the system is ambitious. This control strategy is very innovative hence explaining the absence of practical experience and the lack of literature. The main challenges of this technology would be once again its complexity because of the complexity of the system itself. Very few control design methods are able to handle such systems. The numerical complexity and computational cost induced are also important. The most suitable control technique for this project will be traded off after studying several possible methods.

- How can the combined controller be designed for this project?
  Neither of the two already developed controllers, for the S/C and for the arm respectively, can be simply re-used to design one combined controller. The latter requires a totally new design and development phase which is made even more complicated because both subsystems are operated at very different frequencies (up to 250Hz for the AOCS compared to 1kHz for the manipulator). The design of the controller is also an important sub-goal of the project.

- How will the designed controller be tested and implemented?
  After its design, the controller will need to be tested to assess its characteristics and the behavior of the controlled loop. To be able to do so, some theoretical tools may be used as well as different simulation environments. Such simulators shall be used to implement the controller. Their characteristics shall be discussed and investigated beforehand.

- What can be concluded from the test results concerning the controlled system behavior?
  Once the controller has been tested and implemented, the analysis of the results and their interpretation is crucial to evaluate its performance, its robustness and the interest of such an innovative control strategy for the reach phase of an OOS mission. The added value of this thesis to the topic has to be assessed in order to answer the main research question.

Finally, the research sub-questions given above will be taken and answered one by one, giving to the thesis its structure. Even if there are many challenges, the potential of the combined control concept is great because all actuators and sensors will be used and coordinated together which may improve the system performance and its stability. Having only one controller makes the architecture simpler and more elegant too.

1.5. Scientific Approach and Methodology

The problem to be studied is complex. Some simplifications can be done without losing in completeness. For instance, only the most important uncertainties and limitations will be implemented in the design phase. Also, the complexity can be reduced by assuming that the number of axes of the arm is lower than seven. The motion of the end effector could even be considered only in two dimensions instead of three.

Because a robotic arm is mounted on the servicer satellite in RICADOS technology development project, both space robotics and AOCS theories will be used together to complete the research objective given in Section 1.4. Moreover, the control solution chosen must be robust. In fact, because of the extreme environment, the constraints of the mission or simply the hardware parts, there are many uncertainties, limitations and disturbances during every space mission. Thus, robust control theory will also be studied and used in this thesis project. Two theoretical approaches will be used in order to reach the subgoals thus the main research objective. On the one hand, a dynamics model of the whole servicer satellite (S/C base and robotic arm combined) will be developed. On the other hand, the model will be used to develop and implement the robust combined controller for the deployment of the space manipulator. In addition to the purely theoretical steps detailed above, other tasks must be completed to answer the research questions hence to meet the objective. Indeed, five main steps (including the theoretical steps) will be undertaken during this project:

- Dynamics modelling of the combined system
- Design of the controller
- Implementation of the controller
- Test of the controller
- Analysis of the results

During the whole thesis, the reader should remember that the controller has to run on an on-board computer (OBC) with limited computational resources in real-time. This is one of the most challenging aspect of it. Moreover, the implementation of the controller is complex and multiple. In fact, the objective is to integrate the controller in the satellite on the one hand so that it can complete the maneuver according to the results of an on-board path-planner (which will also be implemented during this thesis) and at the control center on the other hand such that the combined control parameters can be sent from the console in the control room *via* a telecommand. The testing phase of the controller developed and implemented will be essentially done using numerical simulations.

1.6. Outline of the Thesis

The organization of this thesis follows the steps which are taken to complete the sub-goals hence the main research objective of the thesis. First of all, the system studied is further detailed in Chapter 2. Some notions, conventions and some assumptions on the system will be detailed and explained. In Chapter 3, some constraints concerning the system, the environment and the mission are given before modelling the system in Chapter 4. In the latter, a model of the kinematics and of the dynamics of the system is derived and verified. Based on this model, a controller is designed after choosing the control strategy and its architecture in Chapter 5. Once developed, the controller is tested on different simulators which have an increasing complexity and the results are analyzed. Following this Chapter 6, some conclusions are drawn and recommendations for future works are given in Chapter 7.

2

Conventions and Assumptions

This chapter gives the reader the tools needed to understand this thesis. The system is detailed in Section 4.1. The notations used hereafter are given In the next section (Section 2.2) before describing the conventions used in Section 2.3. Finally, some assumptions are made and explained in the last section of this chapter (Section 2.4).

2.1. Definition of the System

By definition, a robotic OOS mission is composed of two spacecraft, the servicer and the target.
In this thesis project, a KUKA LWR robotic arm is mounted on the servicer. On the one hand, the LWR is a seven axis of rotation robotic arm. All joints have joint angle and joint torque sensors [23]. For its control, desired joint torques are given as inputs. Fig. 2.1 shows the manipulator in its stretched configuration. In a real OOS mission, a tool such as a gripper is mounted at the end-effector of the arm to enable the grasping of the target. It is not shown here for the sake of clarity of the figure. However, in the modelling of the system, the gripper is modelled as being fixed with the last joint. It has the shape of a hand and the distance from the seventh joint to the CoM of the gripper is L_{44} along Z_7 axis. The end-effector (EE) position corresponds to the CoM of the gripper, in its 'mouth'. On the other hand, the S/C base is a cubic structure of two meters and has a mass of 1500 kg. It is controlled using actuators such as thrusters and reaction wheels. The LWR is mounted on the upper side of the base as shown Fig. 2.1 at point P. The first link of the arm is fixed on the base. The distance in the chaser body-fixed reference frame from the center of mass of the base to the position where the arm is attached is:

$$\mathbf{L}_0 = \begin{bmatrix} 0 \\ 1.1 \\ 0.7 \end{bmatrix} \text{ in m} \tag{2.1}$$

The target satellite can be of various shape and size and its design in the RICADOS project is based on the design of the target satellite in the DEOS project described in Section 1.2.1 but some further assumptions about the thesis problem are made and detailed in Section 2.4. A circular ring is mounted on it and part of this ring is defined as the handle which the gripper has to grasp. The target satellite is shown Fig. 2.2.

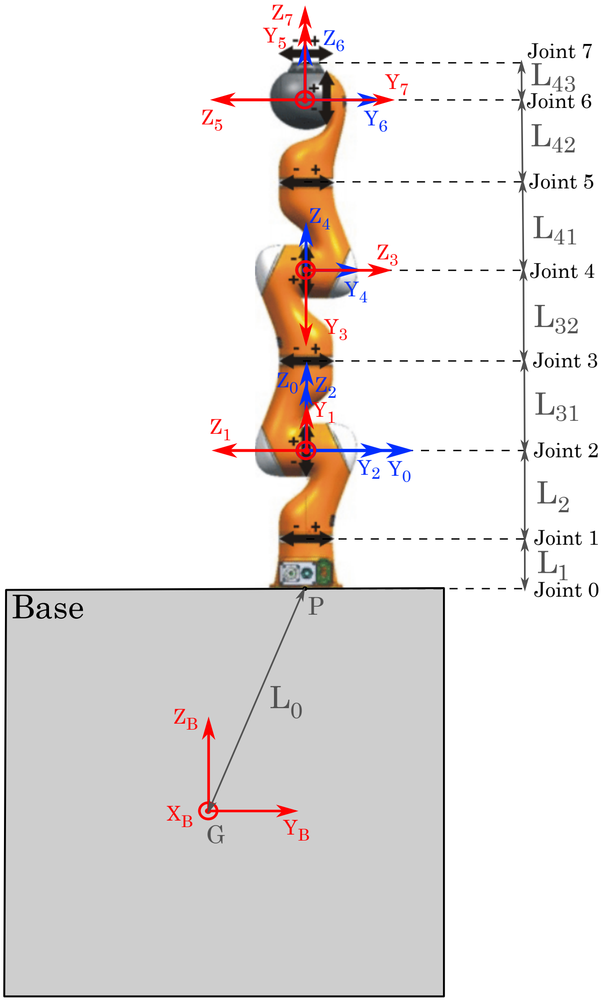


Figure 2.1: Description of the chaser satellite

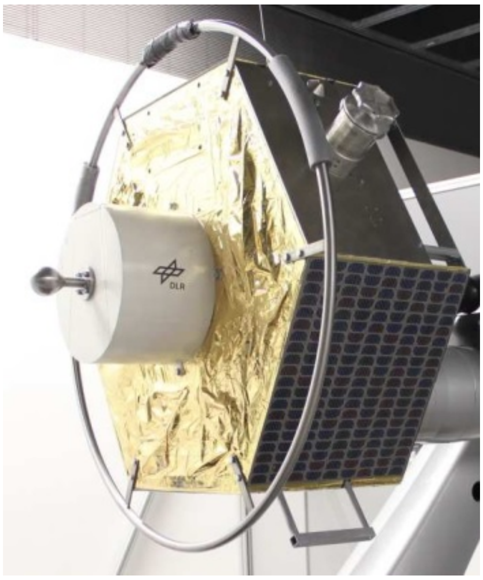


Figure 2.2: Mock-up of the chaser satellite [16]

2.2. Notations

The following notations for vectors, matrices and derivatives will be used throughout this thesis. They are adapted from [61] and [36].

First of all, any arbitrary three-dimensional vector $\mathbf{v}$ can be expressed as a combination of the basis vectors $\hat{\mathbf{e}}_1$, $\hat{\mathbf{e}}_2$, and $\hat{\mathbf{e}}_3$:

$$\mathbf{v} = v_x \cdot \hat{\mathbf{e}}_1 + v_y \cdot \hat{\mathbf{e}}_2 + v_z \cdot \hat{\mathbf{e}}_3 \tag{2.2}$$

where v_x, v_y and v_z are the components of $\mathbf{v}$. The basis vectors are chosen to describe an orthonormal basis. Thus, they can be written as: $\hat{\mathbf{e}}_1 = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix}$, $\hat{\mathbf{e}}_2 = \begin{bmatrix} 0 & 1 & 0 \end{bmatrix}$, and $\hat{\mathbf{e}}_3 = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix}$. Eq. 2.2 can then be rewritten:

$$\mathbf{v} = \begin{bmatrix} v_x & v_y & v_z \end{bmatrix} \cdot \begin{bmatrix} \hat{\mathbf{e}}_1 \\ \hat{\mathbf{e}}_2 \\ \hat{\mathbf{e}}_3 \end{bmatrix} \tag{2.3}$$

From now on $\begin{bmatrix} v_x \\ v_y \\ v_z \end{bmatrix}$ is referred to as a *vector*, and denoted by a lower case, boldface symbol.

A vector $\mathbf{v}$ can be derived in an arbitrary reference frame $\Re_i$. If $\Re_i$ is described using the orthonormal unit vectors $\hat{\mathbf{e}}_{i_1}$, $\hat{\mathbf{e}}_{i_2}$ and $\hat{\mathbf{e}}_{i_3}$, $\mathbf{v}$ can be written as described in Eq. 2.2. The first derivative of $\mathbf{v}$ with respect to time, for instance, in $\Re_i$ is written:

$$\frac{{}^i\partial \mathbf{v}}{\partial t} = \frac{\partial v_x}{\partial t} \cdot \hat{\mathbf{e}}_{i_1} + \frac{\partial v_y}{\partial t} \cdot \hat{\mathbf{e}}_{i_2} + \frac{\partial v_z}{\partial t} \cdot \hat{\mathbf{e}}_{i_3} \tag{2.4}$$

In a more generic way, a vector $\mathbf{v}$ of n scalar variables $q_1, ..., q_n$ is derived with respect to the scalar variable q_k, with $k \in [1; n]$, in the reference frame $\Re_i$ as follows:

$$\frac{{}^i\partial \mathbf{v}}{\partial q_k} = \sum_{j=1}^{3} \frac{\partial v_j}{\partial q_k} \cdot \hat{\mathbf{e}}_{i_j} \tag{2.5}$$

Furthermore, a *matrix* will be denoted using an upper case, boldface symbol. For example, $\mathbf{A}$ is an arbitrary $m \times n$ matrix and a_{ij} is its ijth element:

$$\mathbf{A} = \begin{bmatrix} a_{ij} \end{bmatrix}_{m \times n} = \begin{bmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \vdots & \vdots & & \vdots \\ a_{m1} & a_{m2} & \cdots & a_{mn} \end{bmatrix} \tag{2.6}$$

The *transpose* of the matrix $\mathbf{A} = \begin{bmatrix} a_{ij} \end{bmatrix}$ is $\mathbf{A}^T = \begin{bmatrix} a_{ji} \end{bmatrix}$. If $\mathbf{A}$ is a non singular square matrix, its inverse is denoted $\mathbf{A}^{-1}$.

The notion of *skew-symmetric matrix* of a vector $\mathbf{a}$ will be used in this report. It is denoted as the vector with an × as right superscript $\mathbf{a}^\times$. It is such that:

$$\mathbf{a}^\times \mathbf{b} = \mathbf{a} \times \mathbf{b} \tag{2.7}$$

where × is used as cross product operator and $\mathbf{a}^x$ is defined as

$$\mathbf{a}^\times = \begin{bmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{bmatrix} \tag{2.8}$$

Finally, a system is an operator which maps input signals to output signals. For every input signal $\mathbf{u}$, there is an output $\mathbf{y}$. If the system is represented by its transfer function matrix $\mathbf{P}$, then $\mathbf{y} = \mathbf{Pu}$. The H_∞-norm for transfer function operators is defined as:

$$||\mathbf{P}||_\infty = \sup_\omega \bar{\boldsymbol{\sigma}}(\mathbf{P}(j\omega))$$

where

$$\bar{\boldsymbol{\sigma}}(\mathbf{P}(j\omega)) = \max_i \sigma_i(\mathbf{P}(j\omega))$$

and

$$\sigma_i(\mathbf{P}(j\omega)) = \lambda_i(\mathbf{P}(j\omega)\mathbf{P}^T(j\omega))$$

with

$$\lambda_i(\mathbf{A}), \text{the } i^{th} \text{ eigenvalue of a matrix } \mathbf{A}$$

2.3. Conventions

2.3.1. Reference Frames

Specifying the reference frame in which a vector such as a position, a velocity or an acceleration is expressed is crucial for the correctness and the clarity of the study. It can be represented by three orthogonal axes intersecting at the origin of the reference frame. Every reference frame is taken as a right-handed coordinate system. The following reference frames will be used throughout this report: Earth-Centered Inertial (ECI) reference frame, Local-Vertical Local-Horizontal (LVLH) reference frame, Joint reference frames, Body-Fixed (BF) reference frames, the Links' input and output markers and the relative reference frame. They are described hereafter.

- **Earth-Centered Inertial Reference Frames**

An inertial frame is a coordinate system in which a body with zero net force acting upon it is at rest or is moving at a constant speed in a straight line: it is not accelerating [60]. The origin of ECI coordinate systems is at the CoM of the Earth. One example of Earth-Centered Inertial reference frame which will be used in this thesis is the ECIJ2000. Its axes are defined as follows:

- the x-axis X_I is aligned with the mean equinox at 12:00 Terrestrial Time on 1 January 2000
- the z-axis Z_I is aligned with the Earth's spin axis at that time
- the y-axis Y_I completes the right-handed coordinate system.

In reality, the ECI frames such as the ECIJ2000 are not inertial frames, but pseudo-inertial frames. Indeed, the center of mass of the Earth accelerates due to perturbations from other celestial bodies around the Earth such as the Moon, the Sun and other planets of the solar system. However, the expected effect of these disturbances on the resulting motion of a satellite orbiting the Earth is small enough to be neglected and the ECI frames can be considered as inertial frames.

- **Local-Vertical Local-Horizontal Reference Frame**

The LVLH coordinate system used in this thesis has its origin at the center of mass of the target satellite. In this reference frame, the x-axis X_V is defined along the velocity vector of the vehicle with respect to the inertial frame and the z-axis Z_V along the position vector between the LVLH system's origin and the origin of the inertial frame (i.e. the Earth's CoM). Finally, the y-axis is Y_V completes the right-handed coordinate system: it is perpendicular to the orbital plane.

An LVLH reference frame is used in this thesis to express the attitude and position between the target satellite and the chaser. It has its origin at the (CoM) of the target satellite. It is defined and used for practical reasons. In fact, measurements of position and attitude of the target in the inertial frame may theoretically be used. However, for an OOS mission, the precision required for these measurements are too high to be completed using inertial measurements with the current state of the art sensors. Measuring relative position is more accurate in this case using video-tracking for instance [9]. Thus, the position and attitude of the chaser will be expressed in the LVLH coordinate system of the target. It must be noted that the LVLH is not fixed to the satellite but only to its CoM so that even if the target satellite is tumbling, the axes remain oriented as described above.

Finally, this LVLH reference frame can be considered as inertial since the time between the measurements and the determination of the control inputs is way smaller than the orbital period of the target. Thus, during the deployment of the arm, the target can be assumed to be in a uniform translation.

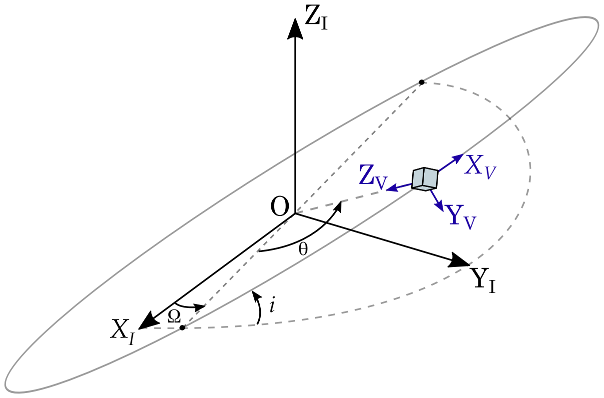


Figure 2.3: ECI and LVLH reference frames

- **Joint Reference Frames**

The Denavit-Hartenberg (DH) convention is often used in robotics to select and describe joints' reference frames. A slightly modified version of it is used in this thesis to set up the set of reference frames for the chaser satellite. They will be used for the kinematics and dynamics modelling of the system.
It consists in defining one reference frame per joint. The attitude of a reference frame $\Re_i$ with respect to the reference frame $\Re_{i-1}$ is expressed using four parameters called the DH parameters. The reference frame attached to the i^{th} joint is defined as follows:

- $\mathbf{z}_{i-1}$ is in the direction of the joint axis between the $(i-1)^{th}$ and i^{th} links.
- $\mathbf{x}_i$ is parallel to the common normal of $\mathbf{z}_{i-1}$ and $\mathbf{z}_i$.
- $\mathbf{y}_i$ completes the right-handed coordinate system with $\mathbf{x}_i$ and $\mathbf{z}_i$.

And the DH parameters from $\Re_{i-1}$ to $\Re_i$ are the following:

- α_i is the angle about common normal, from $\mathbf{z}_{i-1}$ to $\mathbf{z}_i$
- d_i is the offset along $\mathbf{z}_{i-1}$ between $\mathbf{x}_{i-1}$ and $\mathbf{x}_i$
- a_i is the length along $\mathbf{x}_i$ between $\mathbf{z}_{i-1}$ and $\mathbf{z}_i$
- θ_i is the angle about previous $\mathbf{z}_{i-1}$ between $\mathbf{x}_{i-1}$ and $\mathbf{x}_i$. According to the definition of the z-axis of each reference frame, the ith joint angle θ_i is the angle of rotation around $\mathbf{z}_i$ in the reference frame $\Re_i$.

Applying this convention to the system studied in this thesis gives the reference frames as shown Fig. 2.1 and the DH parameters listed in Table 2.1.

Table 2.1: Denavit-Hartenberg parameters for the robotic arm

Joint i	Previous link RF	Next link RF	α_i [rad]	d_i [m]	a_i [m]	θ_i [rad]
1	$\Re_0$	$\Re_1$	$\pi/2$	0	0	θ_1
2	$\Re_1$	$\Re_2$	$-\pi/2$	0	0	θ_2
3	$\Re_2$	$\Re_3$	$-\pi/2$	$L_{31}+L_{32}$	0	θ_3
4	$\Re_3$	$\Re_4$	$\pi/2$	0	0	θ_4
5	$\Re_4$	$\Re_5$	$\pi/2$	$L_{41}+L_{42}+L_{43}$	0	θ_5
6	$\Re_5$	$\Re_6$	$-\pi/2$	0	0	θ_6
7	$\Re_6$	$\Re_7$	0	0	0	θ_7

It is important to note that there is an extra joint 0 when the system $S = \{\text{S/C+Robotic Arm}\}$ is considered. It is the fixed joint between the arm and the S/C. Also, the position of the tool center point (TCP) in the reference frame $\Re_7 = (X_7, Y_7, Z_7)$ is $p_{TCP_7} = \begin{bmatrix}0 & 0 & L_{43}\end{bmatrix}^T$. The gripper is also considered in the

modelling. It is attached to the TCP. Its position in $\mathfrak{R}_7$ is $p_{EE_7} = \begin{bmatrix} 0 & 0 & L_{43} + L_{44} \end{bmatrix}^T$.
The definition of the origins of the joint reference frames as taken here is very convenient since an analogy with a human arm can be made. Indeed, the reference frames $\mathfrak{R}_0$, $\mathfrak{R}_1$ and $\mathfrak{R}_2$ put together act as a spherical joint, exactly as the shoulder does. Similarly, the origins put at Joint 4 act as the elbow and the last one as the wrist. A human arm has however one degree of freedom which is not assigned to the elbow nor to the wrist. It is nonetheless considered here as part of the elbow.

- **Body-Fixed Reference Frames**

A BF reference frame is a reference frame which is fixed to the body of the satellite and its origin is its CoM. A BF reference is defined for both the chaser and the target. They are respectively denoted $\mathfrak{R}_C$ and $\mathfrak{R}_T$. The rotations around the X, Y and Z axes are respectively called roll, pitch and yaw. The rotational equations are often described in this reference frame. When the target's attitude is zero, the X-axis of the target's BF reference frame is in the opposite direction from the x-axis of the LVLH reference frame. Its y-axis points towards the outside of the orbit, in the orbital plane. Finally, the z-axis complete the right-handed coordinate system. This reference frame is the one used in the simulators discussed in Chapter 6. If the attitude of the target does not vary over time, its BF reference frame would be fixed with respect to the LVLH reference frame described above. In this reference frame, if the chaser approaches the target from behind on its orbit, its x position is greater than zero and decreases until the rendezvous phase is completed. Furthermore, concerning the chaser, the orientation of the axes are the same as for the base reference frame described previously in Fig. 2.1 although the origin is not exactly the same since the robotic arm influences the position of the system's CoM. It means that Z_C is along the first joint's axis of rotation and, X_C and Y_C respectively along X_B and Y_B, the axes of the base body-fixed reference frame. Both BF reference frames are shown Fig. 2.4.

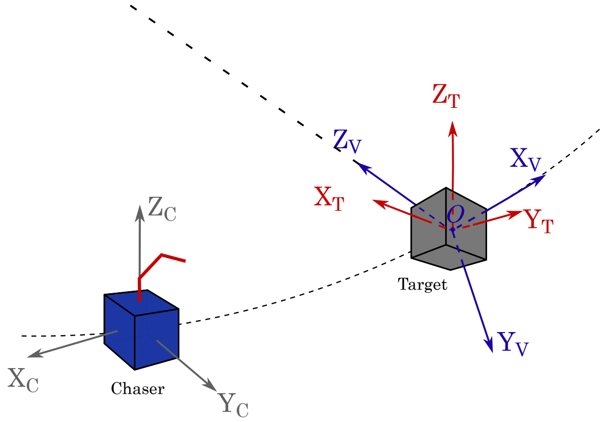


Figure 2.4: Body-fixed and LVLH reference frames

- **Links' Input and Output Markers**

Each link of the robotic arm has an input and an output. The input of the i^{th} link is the section which is in contact with the $(i-1)^{th}$ link while its output is in contact with the $(i+1)^{th}$. The markers at both ends can be considered to facilitate the expression of the kinematics. The input and output markers of the i^{th} link are noted $\mathfrak{R}_{I_i}$ and $\mathfrak{R}_{O_i}$ respectively. In fact, the transformation from the joint reference frame $\mathfrak{R}_i$ to the joint reference frame $\mathfrak{R}_{i+1}$ can then be seen as a succession of transformations: from the joint i (input marker of the link) to the output marker of the link and from the latter to the input marker of the next link. The first transformation is fixed while the second one depends on the joint angle θ_i. The concept of input/output markers in robotics is detailed in [51].

2.3.2. Attitude Parametrisation

The attitude of the spacecraft is described using Euler angles. A set of Euler angles consists of three rotations about the axes of a reference frame. These three rotations are completed following a given sequence. In this thesis, the sequence chosen is XYZ. If the Euler angles vector is noted $\boldsymbol{\delta}_0 = \begin{bmatrix} \phi & \theta & \psi \end{bmatrix}^T$ in the inertial LVLH reference frame, the first rotation is a rotation of ψ about the Z-axis. It gives a new rotated coordinate system (X', Y', Z'). Then a rotation of θ about the Y'-axis, which gives the third coordinate system (X'', Y'', Z'') and finally a rotation of ϕ about X''-axis as shown Fig. 2.5. Any rotation can be described by a set of Euler angles as defined here.

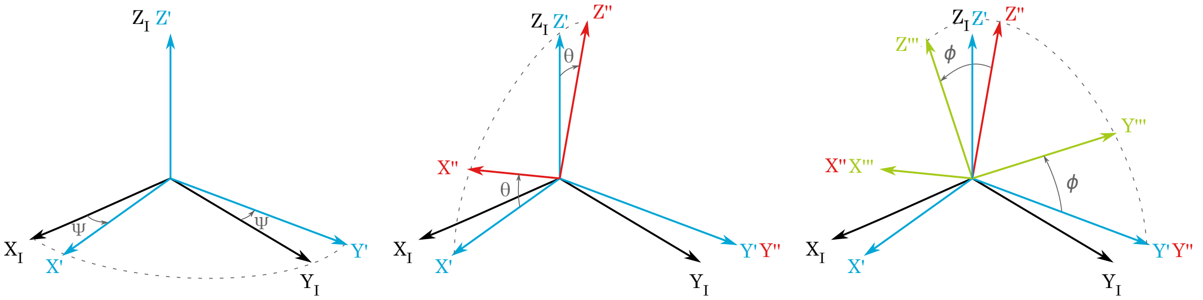


Figure 2.5: Decomposition of a rotation using Euler angles - Sequence ZXZ

2.3.3. Reference Frames Transformations

Transformations from one of the previously presented reference frame to another are crucial to model the kinematics and/or the dynamics of such a complex system. Some of the rotation matrices are given hereafter so that the reader can come back to it later on if needed.

- First of all, the rotation matrix from the BF reference frame to the LVLH inertial reference frame, considering the definition of Euler angles and the convention taken, is:

$$\begin{aligned}\mathbf{A}_0(\boldsymbol{\delta}_0) &= R_X(\phi)R_Y(\theta)R_Z(\psi)\\ &= \begin{bmatrix}1 & 0 & 0\\ 0 & \cos(\phi) & -\sin(\phi)\\ 0 & \sin(\phi) & \cos(\phi)\end{bmatrix}\begin{bmatrix}\cos(\theta) & 0 & \sin(\theta)\\ 0 & 1 & 0\\ -\sin(\theta) & 0 & \cos(\theta)\end{bmatrix}\begin{bmatrix}\cos(\psi) & -\sin(\psi) & 0\\ \sin(\psi) & \cos(\psi) & 0\\ 0 & 0 & 1\end{bmatrix}\end{aligned} \tag{2.9}$$

- Then, the rotation matrix from the input marker of joint i to its output marker is:

$$\begin{aligned}\mathbf{A}_{link_i} &= R_X(\alpha_i)\\ &= \begin{bmatrix}1 & 0 & 0\\ 0 & \cos(\alpha_i) & -\sin(\alpha_i)\\ 0 & \sin(\alpha_i) & \cos(\alpha_i)\end{bmatrix}\end{aligned} \tag{2.10}$$

- Moreover, the rotation matrix from the output marker of joint i to the input marker of joint $i+1$ is:

$$\begin{aligned}\mathbf{A}_{joint_i} &= R_Z(\theta_i)\\ &= \begin{bmatrix}\cos(\theta_i) & -\sin(\theta_i) & 0\\ \sin(\theta_i) & \cos(\theta_i) & 0\\ 0 & 0 & 1\end{bmatrix}\end{aligned} \tag{2.11}$$

- As said in Section 2.3.1, the rotation matrix from the i^{th} joint reference frame to the $(i+1)^{th}$ is:

$$\mathbf{A}_{i,i+1} = \mathbf{A}_{joint_{i+1}}\mathbf{A}_{link_{i+1}}$$

2.3.4. State Variables

The system and the reference frames were defined in Sections 2.1 and 2.3 so that the state variables used in this thesis can be described. All the state variables on which this problem relies are chaser's states some of which are expressed with respect to the the target BF reference frame (position and attitude of the chaser) unless stated otherwise while others are expressed in reference frames of the system (such as joint angles expressed in the joints reference frames). In fact, the states of the target are considered unknown, unless they are measurable from the target, since the target may be a non-operating uncooperative satellite and may not be able to communicate with ground nor with the chaser. The state vector of the chaser is composed of its position, its attitude, the joint angles and their derivatives. However, all the derivatives are assumed not to be measurable.

Spacecraft Position

The position of the CoM of the system consisting in the S/C base and the robotic arm $\mathbf{R}_{C_0}$ is given in the LVLH inertial reference frame. It is decomposed as $\mathbf{R}_{C_0} = \begin{bmatrix} R_{C_{0x}} & R_{C_{0y}} & R_{C_{0z}} \end{bmatrix}^T$.

Spacecraft Attitude

As described in Section 2.3.2, the attitude of the chaser is given by its Euler angle vector $\boldsymbol{\delta}_0$:

$$\boldsymbol{\delta}_0 = \begin{bmatrix} \phi & \theta & \psi \end{bmatrix}^T \tag{2.12}$$

Joint Angles

For each axis of rotation of the manipulator, the joint angle is the rotation angle around the z-axis of the joint reference frame. The joint angle vector can be written:

$$\boldsymbol{\theta}_j = \begin{bmatrix} \theta_1 & \theta_2 & \theta_3 & \theta_4 & \theta_5 & \theta_6 & \theta_7 \end{bmatrix}^T \tag{2.13}$$

Conclusion: State Vector

Finally, the position vector of the chaser is:

$$\mathbf{q} = \begin{bmatrix} \mathbf{R}_{C_0} \\ \boldsymbol{\delta}_0 \\ \boldsymbol{\theta}_j \end{bmatrix} \tag{2.14}$$

and its state vector also considers the first derivatives of $\mathbf{q}$ and is noted $\mathbf{x}$:

$$\mathbf{x} = \begin{bmatrix} \mathbf{q} \\ \dot{\mathbf{q}} \end{bmatrix} \tag{2.15}$$

2.4. Simplifications of the System

Due to the complexity of the system and the challenging thesis objective, some assumptions are taken. They are detailed hereafter and will be considered in this thesis.

First of all, the system is assumed to be rigid. The flexible modes of joints and links are neglected in order to reduce the number of parameters of the problem.

Because the thesis topic does not necessarily address the force and torque allocation issue and to keep the method as generic as possible, the inputs for the control of the spacecraft will be the overall force and torque. The allocation problem of these dynamic actions to the different actuators on board such as reaction wheels and thrusters can be studied separately. Of course, the available overall torque and force depend on the actuators and their location in the spacecraft. Some realistic values will be considered and given later in Section 3.

Moreover, the robotic arm is also simplified. As described in Section 2.1, the LWR is a robotic arm with seven axes of rotation. However, only three axes of rotation will be considered thereafter. Four joints are thus fixed. The thesis focuses on the reach phase. The final position of the end-effector is in grasping distance but the grasping itself is not part of the studied phase. The orientation of the gripper at the end-effector does not influence its position and is thus fixed: $\theta_7 = 0$. The joints 1, 2 and 4 are selected to be the moving joints while the others are fixed with $\theta_i = 0$. The joint angle vector defined in Section 2.3.4 can be rewritten as:

$$\boldsymbol{\theta}_j = \begin{bmatrix} \theta_1 & \theta_2 & 0 & \theta_4 & 0 & 0 & 0 \end{bmatrix}^T \tag{2.16}$$

Other assumptions in terms of path-planning for instance are detailed later in this thesis (Section 5.3).

3

Design Considerations

As explained in Section 1.4, the objective of the thesis is to develop and implement a controller under mission constraints. Any space mission faces many constraints due to the environment, to the hardware and to the specifications of the mission itself. An overview of these constraints, which can be limitations, disturbances and uncertainties, is given in this chapter without being exhaustive. Some of them are taken from [37], [58] and [25]. In these papers, the authors give an overview of the considerations which should be taken during the design of a space mission. Finally, not all these constraints will be considered in this thesis for different reasons. The discussion about what constraints are considered in the following chapters takes place in Section 3.4.

3.1. Hardware Constraints

No piece of hardware is perfectly as designed and expected. There are always uncertainties on the dimensions of a part, on its mass or on its precise composition. Every step of its manufacturing has uncertainties which, once added up, give its final uncertainties. When parts are assembled, these uncertainties are adding up again such that the ideal system considered to develop a model is not exactly the actual system. When modelling a system, one may get the uncertainties the system has such that they can be incorporated in the design. There are uncertainties due to the design *per se*, but other uncertainties are due to the modelling of the parts, of their interactions and of their evolution. Depending on the strength of the modelling assumptions, the model developed in Chapter 4 will be more or less complex. The less complex, the more uncertain. These uncertainties due to the modelling of the system are called unmodelled dynamics uncertainties [55]. They can eventually be considered in the design of the controller.

Hardware parts also have limitations. For example, the actuators cannot produce an infinitely large thrust or torque. These limitations among others have to be considered in order to get results as realistic as possible. Indeed, the maximum torque or force available for the base AOCS and for the robotic arm control will inevitably limit the achievable performance. In addition to this limitation, the actuators are also submitted to their internal dynamic behavior. If a large rotation speed is commanded on a reaction wheel, it will need some time to reach this steady-state value. It does limit once again the performance of the controlled system because the longer it takes to get to the final rotational speed, the less agile the satellite is. Due to the limited energy stored on-board, the required amount of energy to complete a maneuver may not be available for the actuators immediately. In that case, even though the command would not be greater than the actuators' limitations, the power supply and its allocation may constrain the achievable performance. The actuators are not the only hardware parts with limitations which drive the performance of the controller. The computational power of the on-board computer (OBC) is limited and the way it is allocated is essential for the system to work properly. The available computational power on-board is way more limited than on ground (usually less than 30% [43]). If its distribution is not done carefully, some computations may not be doable and may thus jeopardize the whole mission. Also, if the computational power available to compute the control action to be completed is too low, the model used to do it will need to be more simplified thus less accurate than if the computational power was higher. Last but not least, the operating frequency of each part in the control

loop can be different. They highly influence the performance of the maneuver. If the overall operating frequency is too low, the maneuver will take time and the disturbances will have a larger impact since they will be counteracted later than if the frequency was higher. This may lead to important overshoot, large oscillations with low damping and an overall poor performance and robustness of the system during the deployment of the arm.
Finally, some disturbances have to considered in addition to the uncertainties and the limitations. In the RICADOS development project, the relative position and attitude between the target and the chaser is determined using video-tracking [9]. Moreover, the target satellite may not be operational anymore so knowing the chaser's position and attitude would not be enough to capture the target. In any case, noise components are added to the output signal when measured. These cannot be avoided simply because the sensors are not ideal. The measurements of the joint angles will also have noise. It constrains the achievable performance of the control. In fact, the different states are known within a certain range defined based on the measurement noise so this range limits the precision of the control maneuver. For instance, if the noise amplitude would be too large, the achievable precision would not be good enough to reach the handle on the target satellite. Disturbance can also come from the OBC. This piece of hardware is also non-ideal. Its bit error rate for instance is an important parameter. Some techniques exist to detect the errors and to correct them such as forward error correction techniques.

3.2. Environmental Constraints

In addition to the hardware constraints presented in Section 3.1, some environmental constraints apply to an OOS mission and other space missions [58]. These constraints are mostly uncertainties due to the extreme conditions in space such as radiations, vacuum, temperature differences or micro-gravity. However, it is considered that every part of the satellite is space qualified so that these constraints are assumed to be met already. In the simulator used at DLR, micro-gravity effects on the satellites is computed in the dynamics simulator and, as shown Fig. 1.3, two large KUKA robots are used to physically show how the system evolves, based on the simulated relative motion between the servicer and the target. Moreover, space environment is very harsh in terms of temperature variations. Indeed, temperature variations are very wide between the part of the orbit during which solar energy impinges on the satellite and the one when the satellite is in eclipse. The issue of thermal control (active and/or passive) is assumed to have been solved in a previous stage of the project such that the AOCS engineer does not have to take it into account. The constraints undergone by a spacecraft in space are more detailed in [25].

3.3. Mission Constraints

The main objective of the RICADOS project is to get closer to reality by increasing the TRL of the technologies used for OOS. As explained in Section 1.2.2, a simulator was developed during the E2E project. One of its objectives was to develop a complete ground control segment and to obey to constraints which are not directly due to the hardware nor to the environment. For instance, encryption protocols were implemented, a control room was set up from which all activities related to the mission control and supervision are completed. Once in the control room, the difference between controlling the simulator or a real satellite in space is very little. It means that telecommands (TC) are sent and telemetries (TM) are received, the data collected is limited and there is no other way than processing the data on-ground to get a physical overview of the situation in which the simulator is. Some details of the environment can be found in [9]. Implementing a controller in this kind of advanced simulation environment requires that everything about this controller is compatible with the simulator. Some additional uncertainties are also implied such as delays, jitters and a lower reliability of the transmission due to the space protocols used. One objective of the thesis is to be able to comply with all the constraints of the simulator in order to be able to implement it within the simulation environment. The simulation environment will be further detailed in Chapter 6.
Moreover, the procedure followed in RICADOS is as described by ESA in [1]. First of all, the EPOS (Fig. 1.2) is used to simulate the approach phase of the two satellites. Then, the scenario is taken over by the OOS-SIM once the chaser is close enough to the target, it is the park phase. It then enters the *capture phase*. The phase studied in this thesis is a sub-phase of the latter. Indeed, the capture phase can be further decomposed in four main sub-phases. The chaser first synchronizes itself with the target after receiving a TC from ground. This synchronization can be required when the target satellite is tumbling

for instance. In that case, the chaser shall get to the same rotation speed before extending the robotic arm in order to minimize the risk of collision and the effort on the robot when the contact is established. When the two satellites are synchronized, the reach phase starts, it consists in deploying the space manipulator such that the gripper is in grasping distance of the handle. Then comes the form closure, when the gripper is closed such that the target cannot escape and finally a rigid closure sub-phase terminates the capture phase. In this sub-phase, the gripper stiffens itself around the handle. Once the capture phase is completed, the system is stabilized and made rigid: the two satellites move as one. Finally, the chaser satellite can complete its servicing mission. The implementation of the controller developed in this thesis shall be able to fit into this scenario. It means that once the end-conditions of the sync sub-phase are completed, it shall be able to take over, complete its task and meet the end-conditions necessary to move on to the form closure sub-phase.

Finally, another important mission constraint is due to the communication strategy chosen. The ground control team wants as much data as possible to be able to check the correct progress of the mission but also to send commands to the spacecraft if needed. However, the available bandwidth and communication frequency are bounded. The available bandwidth limits the quantity of data that can be sent and received by the spacecraft from ground and *vice versa*. This constraint will force the maneuver not to be extremely quick such that the ground control can still oversee the process and check that every step is done well. Nonetheless, the duration of radio contact is a strong limitation for any maneuver as well. Indeed, in order to be able to keep an eye on the ongoing maneuver, some data packets must be sent to the ground and be studied by the ground control team. The latter must be able to abort the mission at any moment if a problem is detected on ground. Because the spacecraft is not always in contact with the same ground station, the communication with ground is lost if the maneuver is too long. The time of a maneuver must then be lower than the time of radio contact or the maneuver must be split in different tasks completed over different periods of radio contact.

3.4. Discussion

As precised earlier, this section covers only some of the constraints which shall be considered before modelling the system and designing the controller. This overview is not a comprehensive list of all the constraints. Many different constraints have already been described. However, not all of them will be considered during this thesis for different reasons. Some, such as the ones due to the hostile and unknown environment, are not considered because their effects during a short maneuver like the deployment of the arm are negligible. Indeed, every part is assumed to be space qualified already. Some other constraints are neglected because their impact is way smaller than the impact of other constraints. For instance, during the time of the maneuver, the effect of the solar pressure on the variation of the spacecraft's position is negligible compared to the noise of the measurement. In fact, the effect of phenomena such as solar pressure, radiations or possible drag due to a non-vacuum environment for instance have a significant impact on a long period of time compared to the maneuver studied here. Thus it will not be considered hereafter.

Finally, the constraints which will effectively be taken into account are the following. Concerning the actuators, the maximum and minimum torques and forces, the resolution of the measurements and the operating frequency will be important limits and uncertainties. For the robotic arm especially, the workspace definition (taking into consideration the parts around not to collide with) is a crucial limitation. The controller shall meet the requirements given in Section 3.3 in order to be implementable in the simulation environment at DLR. Once implemented in the simulator, the controller will be subject to various other disturbances and its behaviour may be impacted by them. The objective of this thesis is not to consider every single mission constraint but to develop a first version of combined controller and show the potential of this control strategy. Of course, in further studies, wider sets of constraints can be considered.

4

System Modeling

This chapter is used to give the reader some insight in the technical derivations of the model of the system. First, technical aspects of the system are described in Section 4.1. Based on these, a model of the kinematics of the chaser is developed (Section 4.2) before deriving a model of the system dynamics in Section 4.3.

4.1. System Description

As described in Chapter 2, the system considered is composed of a spacecraft base on which a robotic arm is mounted. The space manipulator has three revolute joints, each of which frees one degree of freedom. The base of the spacecraft has six degrees of freedom. Thus, the overall system has nine.
Because the dynamics of the manipulator and of the spacecraft base are coupled, operating the arm implies that the uncontrolled base is moving, according to Newton's third law of motion. However, the objective of the thesis being to control the base while the arm is operating, the S/C will be ideally stationary with respect to the target satellite. In that case, the workspace of the space manipulator, which is the set of positions in space which can be reached by the EE, is found using the same techniques as for a fixed base robotic arm. Thus, the workspace can be described in the chaser body-fixed frame as the spherical shell centered at $\mathbf{C}$ with outer radius R_{out} and inner radius R_{in} where:

$$\mathbf{C} = \begin{bmatrix} 0 & L_{0_y} & L_{0_z} + L_1 + L_2 \end{bmatrix}^T \text{ and } \begin{cases} R_{in} = |L_3 - L_4| \\ R_{out} = L_3 + L_4 \end{cases}$$

and where L_0, L_1, L_2, $L_3 = L_{31} + L_{32}$ and $L_4 = L_{41} + L_{42} + L_{43} + L_{44}$ are as shown Fig. 2.1. L_3 is the length of the third link of the robotic arm once the assumption of a three axes of rotation robotic arm is taken and L_4 is the length of the fourth one with the gripper included.
The workspace is shown Fig. 4.1. On this figure, it is clear that if $L_1 + L_2 < L_3 + L_4$, the workspace of the manipulator is reduced since it shall not collide with the S/C. However, having an end-effector position required to be within the S/C is not realistic since it describes the position of the handle on the target satellite. Thus, it is assumed that the workspace does not have to be restrained but one should keep a critical point of view with respect to that topic when analyzing the results. Moreover, considering that the control of the base is perfect such that the motion of the arm does not cause any motion of the base is an assumption which will be discussed later by analyzing the results.

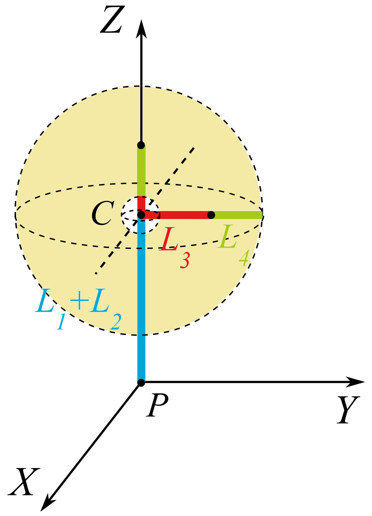


Figure 4.1: Definition of the workspace (in yellow, in between the two spheres)

4.2. Modeling of the Kinematics

The kinematics of a system consists in describing the motion geometrically, without considering any forces or torques applied on it. Studying the kinematics of the system is necessary because the deployment of the robotic arm will be driven by the requested final position of the end-effector which corresponds to the position of the handle mounted on the target satellite. However the manipulator is controlled at joint level, meaning that joint torques are given as inputs and the joint angles are obtained as outputs. To be able to determine the position of the end-effector, the joint angles and the geometry of the arm are used. It is called Forward Kinematics (FK) and it is detailed in Section 4.2.1. Inversely, when an end-effector position is requested, the joint states giving this position are computed using Inverse Kinematics (IK). It is described in Section 4.2.2. The position of the end-effector can be given in the inertial reference frame (IRF) or in the target BF reference frame, as defined in Section 2.3.1.

4.2.1. Forward Kinematics Derivation

The computation of the end-effector position of the arm in the IRF is based on the known position of the spacecraft in the IRF, on its attitude and on the joint angles of the space manipulator.
As explained in Section 2.3.1, the robotic arm is described using DH convention. Based on Table 2.1, the homogeneous transformation matrix from body i to body $(i-1)$, $\mathbf{A}_{i,i-1}$, is derived as:

$$\begin{aligned}\mathbf{A}_{i-1,i}(\theta_i) &= \begin{bmatrix} \cos(\theta_i) & -\cos(\alpha_i)\sin(\theta_i) & \sin(\alpha_i)\sin(\theta_i) & a_i \cdot \cos(\theta_i) \\ \sin(\theta_i) & \cos(\alpha_i)\cos(\theta_i) & -\sin(\alpha_i)\sin(\theta_i) & a_i \cdot \sin(\theta_i) \\ 0 & \sin(\alpha_i) & \cos(\alpha_i) & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \\ &= \begin{bmatrix} & a_i \cdot \cos(\theta_i) \\ \mathbf{A}_{joint_i} \cdot \mathbf{A}_{link_i} & a_i \cdot \sin(\theta_i) \\ & d_i \\ \mathbf{0}_{1\times 3} & 1 \end{bmatrix}\end{aligned} \tag{4.1}$$

The position of the EE in the BF reference frame of body i, $\mathfrak{R}_i$, is noted $\mathbf{p}_{EE_i}$. The corresponding homogeneous coordinates of the EE in $\mathfrak{R}_i$ are $\mathbf{p}_{EE_{h_i}} = \begin{bmatrix}\mathbf{p}_{EE_i} & 1\end{bmatrix}^T$. The latter can be rewritten in the RF $\mathfrak{R}_0$ as:

$$\mathbf{p}_{EE_{h_0}} = \mathbf{A}_{0,1}(\theta_1) \cdot \mathbf{A}_{1,2}(\theta_2) \cdot \mathbf{A}_{2,3}(\theta_3) \cdot \mathbf{A}_{3,4}(\theta_4) \cdot \mathbf{A}_{4,5}(\theta_5) \cdot \mathbf{A}_{5,6}(\theta_6) \cdot \mathbf{A}_{6,7}(\theta_7) \cdot \mathbf{p}_{EE_{h_7}} \tag{4.2}$$

This position corresponds to the distance from the center of the workspace C and the EE. To have it in the base RF, $\mathfrak{R}_B$, the distance from the CoM of the base, G, and C is added:

$$\mathbf{p}_{EE_B} = \mathbf{p}_{EE_0} + \mathbf{GC} \tag{4.3}$$

where $\mathbf{GC}$ is the distance from G to C. In fact, $\mathbf{GC}_B = \mathbf{L_0} + \mathbf{L_1} + \mathbf{L_2}$. The position of the EE can also be expressed in the IRF using the transformation:

$$\mathbf{p}_{EE_I} = \mathbf{T}_{I,B} \cdot \mathbf{p}_{EE_B} + \mathbf{d}_{I,B} \tag{4.4}$$

with $\mathbf{d}_{I,B}$ being the distance between the origin of the IRF and the CoM of the base and $\mathbf{T}_{I,B}$ the rotation matrix from the base RF to the IRF.
Finally, based on the joint angles, the attitude of the spacecraft and its position with respect to the origin of the IRF, the position of the end-effector can be computed in the IRF or in the base RF. It can also be expressed in any other reference frames by replacing $\mathbf{T}_{I,B}$ and $\mathbf{d}_{I,B}$ in Eq. 4.4 by the rotation matrix from the base RF to the reference frame of interest and the distance between the origin of the RF of interest and the base CoM respectively.

4.2.2. Inverse Kinematics Derivation and Choice of Configuration

If a certain position of the EE is required and the position of the base CoM with respect to the origin of the IRF is known, the joint angles can be computed in order to bring the EE to the expected position. With the assumption of a three DoF robotic arm, there are four combinations of joint angles possible for any end-effector position within the workspace of the manipulator. The workspace of the manipulator is defined as the set of EE positions which can be reached for a defined pose of the spacecraft.

- **Derivation**

The desired end-effector position can be given in both the base RF, $\mathbf{p}_{EE_B}$, and the IRF, $\mathbf{p}_{EE_I}$. As described in Section 2.3.3, it can easily be expressed in the base RF. For a robotic arm with three axes of rotation as considered in this thesis, a geometric derivation of the inverse kinematics can be completed. For the sake of clarity, the position of the EE is first expressed in the output RF of base, $\mathfrak{R}_{O_B}$, which has its origin P where the arm is mounted. It is noted $\mathbf{p}_{EE_{O_B}}$. The orientation of these RFs are equal and the distance between their origins, G and P, is equal to $\mathbf{L_0}$. The transformation can be written:

$$\mathbf{p}_{EE_{O_B}} = \mathbf{p}_{EE_B} - \mathbf{L}_0 \tag{4.5}$$

Fig. 4.2 shows the main geometric parameters used in the derivation. Even though the first revolute joint is in reality between the first and second links of the arm, it is modelled as being at the interface between the arm and the base to make the derivation clearer. The latter is not influenced by this change.

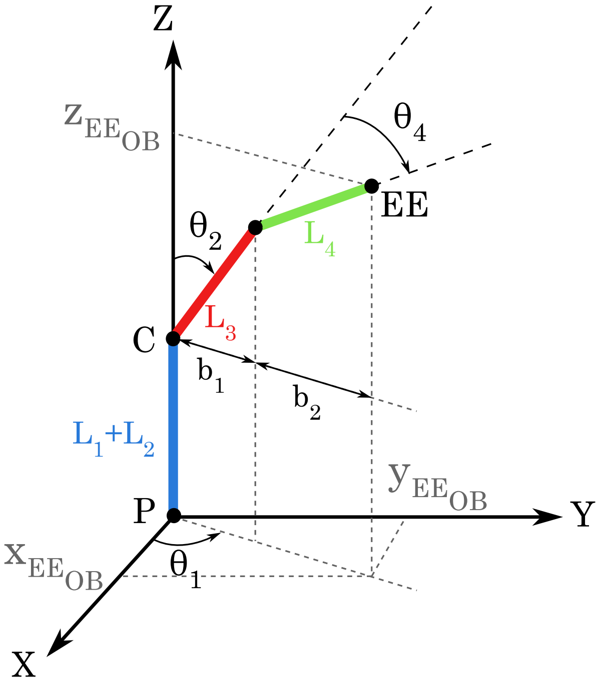


Figure 4.2: Geometric representation of the system

The projection of the end-effector position on all axes of the reference frame gives:

$$\begin{cases} x_{EE_{O_B}} = x_{EE_B} = (b_1 + b_2) \cdot \cos(\theta_1) \\ y_{EE_{O_B}} = y_{EE_B} - L_{0_y} = (b_1 + b_2) \cdot \sin(\theta_1) \\ z_{EE_{O_B}} = z_{EE_B} - L_{0_z} = L_1 + L_2 + L_3 \cdot \cos(\theta_2) + L_4 \cdot \cos(\theta_2 + \theta_4) \end{cases} \tag{4.6}$$

and

$$\begin{cases} b_1 = L_3 \cdot \sin(\theta_2) \\ b_2 = L_4 \cdot \sin(\theta_2 + \theta_4) \end{cases} \tag{4.7}$$

so that it can be written:

$$\begin{cases} x_{EE_{O_B}} = (L_3 \cdot \sin(\theta_2) + L_4 \cdot \sin(\theta_2 + \theta_4)) \cdot \cos(\theta_1) \\ y_{EE_{O_B}} = (L_3 \cdot \sin(\theta_2) + L_4 \cdot \sin(\theta_2 + \theta_4)) \cdot \sin(\theta_1) \\ z_{EE_{O_B}} = L_1 + L_2 + L_3 \cdot \cos(\theta_2) + L_4 \cdot \cos(\theta_2 + \theta_4) \end{cases} \tag{4.8}$$

The combination $((1)^2 + (2)^2 + ((3) - (L_1 + L_2))^2)$ of the equations in Eq. 4.8 gives:

$$\begin{aligned} x^2_{EE_{O_B}} + y^2_{EE_{O_B}} + (z_{EE_{O_B}} - (L_1 + L_2))^2 &= (L_3 \cdot \sin(\theta_2) + L_4 \cdot \sin(\theta_2 + \theta_4))^2 \\ &+ (L_3 \cdot \cos(\theta_2) + L_4 \cdot \cos(\theta_2 + \theta_4))^2 \\ &= L_3^2 + L_4^2 \\ &+ 2 \cdot L_3 \cdot L_4 \cdot (\sin(\theta_2) \cdot \sin(\theta_2 + \theta_4) + \cos(\theta_2) \cdot \cos(\theta_2 + \theta_4)) \\ &= L_3^2 + L_4^2 + 2 \cdot L_3 \cdot L_4 \cdot \cos(\theta_4) \end{aligned} \tag{4.9}$$

Thus

$$\cos(\theta_4) = \frac{x^2_{EE_{O_B}} + y^2_{EE_{O_B}} + (z_{EE_{O_B}} - (L_1 + L_2))^2 - L_3^2 - L_4^2}{2L_3L_4} \tag{4.10}$$

and it is known that:

$$\sin(\theta_4) = \pm\sqrt{1 - \cos^2(\theta_4)}$$

The latter gives two possibilities:

$$\begin{cases} \theta_4^+ = \operatorname{atan2}(\sin(\theta_4), \cos(\theta_4)) \\ \theta_4^- = \operatorname{atan2}(-\sin(\theta_4), \cos(\theta_4)) \end{cases} \tag{4.11}$$

The first one, when $\sin(\theta_4) > 0$, gives a configuration in which the elbow is up ('u') whereas the second one describes a solution with the elbow down ('d') as shown Fig. 4.3. This figure shows 'u' configurations in the left column and 'd' configurations in the right one. Moreover

$$b_1 + b_2 = x^2_{EE_{O_B}} + y^2_{EE_{O_B}} = (L_3 \cos(\theta_2) + L_4 \cos(\theta_2 + \theta_4))^2 \tag{4.12}$$

Now, following the definitions of sine and cosine:

$$\begin{cases} \cos(\theta_1) = \dfrac{x_{EE_{O_B}}}{\pm\sqrt{x^2_{EE_{O_B}} + y^2_{EE_{O_B}}}} \\ \sin(\theta_1) = \dfrac{y_{EE_{O_B}}}{\pm\sqrt{x^2_{EE_{O_B}} + y^2_{EE_{O_B}}}} \end{cases} \tag{4.13}$$

so

$$\begin{cases} \theta_1^+ = \operatorname{atan2}(y_{EE_{O_B}}, x_{EE_{O_B}}) \\ \theta_1^- = \operatorname{atan2}(-y_{EE_{O_B}}, -x_{EE_{O_B}}) \end{cases} \tag{4.14}$$

These two configurations correspond to different rotations around Z-axis. θ_1^+ corresponds to the configuration in which, after the rotation, the rotation of the second joint is positive (in the sense of the conventions taken and presented in Fig. 4.2) to reach the required end-effector position. This configuration is called 'front' ('f' for short). Similarly, θ_1^- corresponds to the configuration in which the rotation of the second joint is negative, it is described as the configuration 'back' ('b'). These names are physically related to Fig. 4.3 in which the first and second rows respectively depict the 'f' and 'b' configurations. Also, based on Eq. 4.6:

$$\begin{aligned} \cos(\theta_1)\cdot x_{EE_{O_B}} + \sin(\theta_1)\cdot y_{EE_{O_B}} &= (\cos^2(\theta_1)+\sin^2(\theta_1))\cdot(b_1+b_2) \\ &= (L_3\cdot\sin(\theta_2) + L_4\cdot\sin(\theta_2+\theta_4)) \end{aligned} \tag{4.15}$$

or the following trigonometric identity is known:

$$\sin(\theta_2+\theta_4) = \sin(\theta_2)\cdot\cos(\theta_4) + \cos(\theta_2)\cdot\sin(\theta_4)$$

so, if the latter is replaced in Eq. 4.15, one gets:

$$\begin{cases} \cos(\theta_1)\cdot x_{EE_{O_B}} + \sin(\theta_1)\cdot y_{EE_{O_B}} & = L_3\sin(\theta_2) + L_4\sin(\theta_2)\cos(\theta_4) + L_4\cos(\theta_2)\sin(\theta_4) \\ z_{EE_{O_B}} - (L_1+L_2) & = L_3\cos(\theta_2) + L_4(\cos(\theta_2)\cos(\theta_4) - \sin(\theta_2)\sin(\theta_4)) \end{cases} \tag{4.16}$$

thus, rewriting the second equation of this system gives:

$$z_{EE_{O_B}} - (L_1+L_2) = \cos(\theta_2)\cdot(L_3 + L_4\cos(\theta_4)) - L_4\sin(\theta_4)\sin(\theta_2) \tag{4.17}$$

The previous equations imply the following equation:

$$\begin{bmatrix} L_4\sin(\theta_4^{(+,-)}) & L_3 + L_4\cos(\theta_4) \\ L_3 + L_4\cos(\theta_4) & -L_4\sin(\theta_4^{(+,-)}) \end{bmatrix} \cdot \begin{bmatrix} \cos(\theta_2) \\ \sin(\theta_2) \end{bmatrix} = \begin{bmatrix} \cos(\theta_1^{(+,-)})\cdot x_{EE_{O_B}} + \sin(\theta_1^{(+,-)})\cdot y_{EE_{O_B}} \\ z_{EE_{O_B}} - (L_1+L_2) \end{bmatrix} \tag{4.18}$$

which is equivalent to a linear system such as $\mathbf{A}\cdot\mathbf{x} = \mathbf{b}$. It can then be solved ($\mathbf{A}$ assumed to be invertible) and:

$$\mathbf{x} = \mathbf{A}^{-1}\cdot\mathbf{b} \tag{4.19}$$

Because there are two solutions for θ_1, two solutions for θ_4 and because θ_2 is a combination of the two, there are four different solutions for θ_2. Fig. 4.3 describes these four solutions for a given end-effector position.

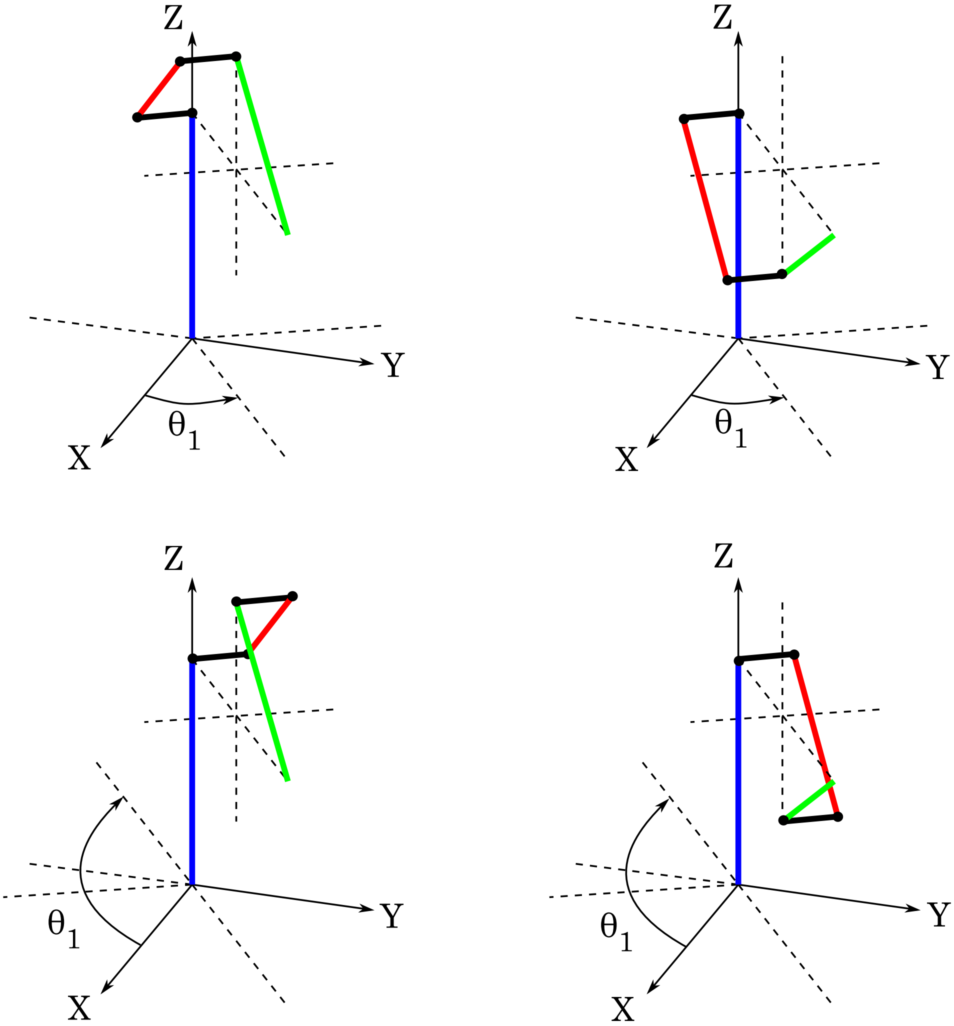


Figure 4.3: Four solutions of joint angles triplets to achieve a given end-effector position

The four solutions which can be called front-up ('fu'), back-up ('bu'), front-down ('fd') and back-down ('bd'). The last joint angle θ_2 can be generally described as:

$$\theta_2^{(b,f),(d,u))} = \mathrm{atan2}(\sin(\theta_2^{(b,f),(d,u)}), \cos(\theta_2^{(b,f),(d,u)})) \tag{4.20}$$

meaning that there are four solutions, combinations of (r,l) and (d,u).

- **Choice of configuration**

To achieve the end-effector position required, one of the four configurations detailed previously must be chosen. The strategy to select the configuration is free to choose. The optimal choice of configuration has not yet been implemented. There are several methods possible to choose the optimal set of joint angles computed using the inverse kinematics. One of which is detailed hereafter.

First of all, the current joint angles are given as inputs $\boldsymbol{\theta}_0$ to the algorithm as well as the expected final end-effector position $\mathbf{p}_{EE_{fin}}$. The four configurations - $\boldsymbol{\theta}_{c1}$, $\boldsymbol{\theta}_{c2}$, $\boldsymbol{\theta}_{c3}$ and $\boldsymbol{\theta}_{c4}$ - are derived and the sets of angles are compared to the input set. Instead of choosing the set which is the closest to the current configuration of the arm in terms of angles, an energy-based optimization can be conducted. In fact, if the energy is not considered, the resulting motion to reach the required end-effector position may not be optimal. Indeed, there may be a case in which one configuration consists in completing a very small rotation for the first joint and larger ones for the second and third joints and another configuration may dictate a large rotation of the first joint and small ones for the two other joints, with an overall rotation smaller than for the previous one. However, if the first joint requires a way larger amount of energy to rotate than the two other joints, the first configuration may lead to a lower amount of energy required to achieve the target position. Each joint i consumes a defined amount of energy per degree E_i. This value, for each joint, can be used as a weighting factor. The weighted angle motion for all three joints are then added and their sum can be taken as a rating criterion C_E: the lower the criterion, the more efficient the motion to reach the expected end-effector position.

$$\begin{aligned} C_{E_k} &= \sum_i \Delta\theta_{ck_i} \cdot E_i \\ &= \Delta\theta_{ck_1} \cdot E_1 + \Delta\theta_{ck_2} \cdot E_2 + \Delta\theta_{ck_4} \cdot E_4 \end{aligned} \tag{4.21}$$

where $k = \{1, 2, 3, 4\}$ is the configuration considered. Finally, the set $\boldsymbol{\theta}_{cj}$ chosen is such that:

$$C_{E_j} = \min_{k \in \{1, 2, 3, 4\}} C_{E_k} \tag{4.22}$$

In a later stage of the RICADOS technology development project at DLR, a study for obstacle avoidance will be conducted and it will also influence the choice of configuration because the most efficient sequence of rotations to reach the handle (final EE position) may require to go through obstacles. To avoid collisions, the different configurations will be studied with respect to a map referencing the obstacles in the vicinity of the chaser (e.g. an appendage on the target satellite) and the ones going through obstacles will be dismissed.

In the RICADOS simulator, the choice of configuration during the inverse kinematics is based on the amplitude of the change required to reach each of the four configurations. The lower the sum of the change on each joint angle, the better the set of joint angles. This algorithm will then be used when the controller will be implemented in this simulation environment (Chapter 6).

4.2.3. Verification of the Kinematics Model

The objective of this section is to verify that the results of the forward and inverse kinematics described in the previous sections give correct results.

On the one hand, an end-effector position could be given as input, used to compute the inverse kinematics to obtain the joint angles and to compute the forward kinematics using this set of joint angles. If the resulting end-effector position of the forward kinematics is the same as the initial position given as input, both the FK and the IK are considered correct. Of course, this is possible only because the methods implemented for the FK and for the IK are different. If one would have been to take the inverse of the other, this verification method may have been sound but not valid for sure. However, the aim of the validation is to test this loop for many tests. Since the possible end-effector position is limited by the workspace of the robotic arm, random test values of positions is not optimal.

On the other hand, the joint angles vector can be given as input. The corresponding end-effector position is then derived using FK. And the loop is closed by running the IK using this position. If the final set of joint angles given as output of the IK is equal to the one given as input initially, then the results are correct. However, this is doable only if the right configuration of the joints is taken as described in Section 4.2.2. Since the choice of configuration has not been implemented at this stage, an additional step is used in the verification procedure: the set of joint angles is used to run the FK once again and if the final end-effector position $\mathbf{p}_1$ is equal to the one obtained previously $\mathbf{p}_0$, then the FK and the IK are verified. Indeed, two totally different methods were used so coherent results must induce that both methods are coherent. This procedure is detailed using a block diagram (Fig. 4.4).

This verification can be considered valid only if a large number of tests (N large) are proven to be successful. N was taken equal to ten thousand during the verification phase of the forward and inverse kinematics developed in previous sections. The differences along all three components of the initial and final end-effector positions, respectively $\mathbf{p}_{EE_{1_i}}$ and $\mathbf{p}_{EE_{2_i}}$, were computed resulting in a vector $\Delta\mathbf{p}_{EE_i}$. The norm of the latter was taken and its maximum over the N tests corresponds to the worst case scenario. The vectors norm can be compared, as shown Fig. 4.5. The criterion for each test i, C_{p_i}, used and plotted in this figure can be described as:

$$C_{p_i} = \left\|\Delta\mathbf{p}_{EE_i}\right\| = \left\|\mathbf{p}_{EE_{2_i}} - \mathbf{p}_{EE_{1_i}}\right\| \tag{4.23}$$

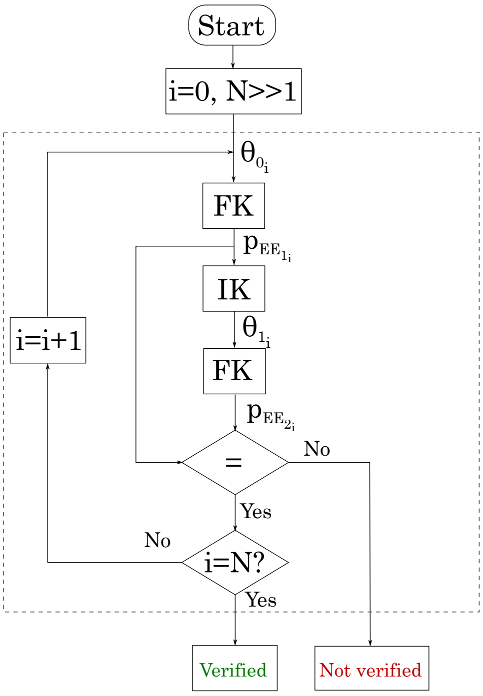


Figure 4.4: Block diagram for the verification of the FK and IK

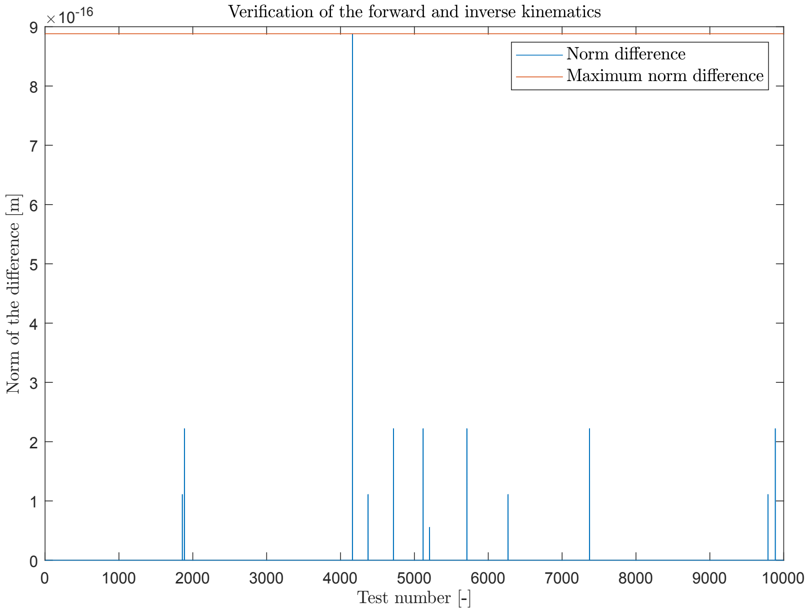


Figure 4.5: Norm of C_{p_i} for $i \in [1 : N]$ and $N = 10000$

Fig. 4.5 shows that the largest difference obtained over the N tests is $8.88 \cdot 10^{-16} m$. This value is small and close to the floating-point relative accuracy (which is $eps = 2.22 \cdot 10^{-16}$). It can be considered as a numerical error. Thus, according to the verification procedure described earlier in this section, the algorithms used to derive the forward and inverse kinematics are verified.

4.3. Modeling of the Dynamics

4.3.1. Forward Dynamics Derivation

Developing a model for the forward dynamics (FD) of the system is necessary in order to establish a relationship between the efforts applied to the system (i.e. the force on the spacecraft base, the torque at the center of mass of the base and the joint torques applied at each torque) and the motion of the system resulting from these forces using the accelerations which are equivalent to the second derivatives of the state vector **q** given previously 2.3.4. For the derivation of the model, some parameters are used. They are described Fig. 4.6.

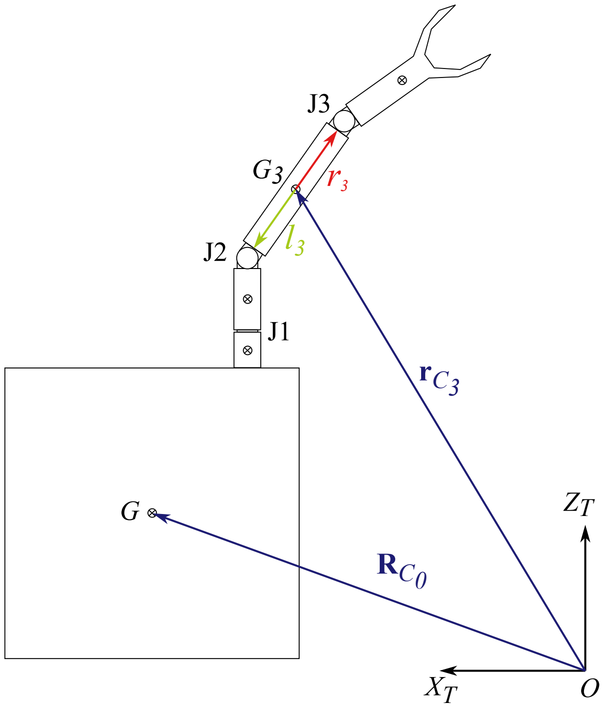


Figure 4.6: Representation of the system and definition of important parameters

Because of the complexity of the system, the dynamics derivation is conducted based on the Lagrangian mechanics. If the kinetic energy of the system is noted T and the potential energy of the system V, then the non-relativistic Lagrangian L is expressed as $L = T - V$. Using the notation **Q** as the vector of generalized forces as detailed above, the Lagrange's equations of the first kind can be written:

$$\frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial L}{\partial \dot{q}_i}\right) - \frac{\partial L}{\partial q_i} = Q_i, \quad i = 1, \dots, N \tag{4.24}$$

where N is the system degrees of freedom i.e the number of components of the state vector and q_i, $\dot{q}_i$ and Q_i are the i^{th} components of **q** (Eq. 2.15), $\dot{\mathbf{q}}$ and **Q** respectively.

Eq. 4.24 can be simplified in the case studied. Indeed, a typical manoeuvre of an on-orbit servicing satellite is of relatively short length and duration so the effects of microgravity and orbital mechanics are negligible compared to control forces **Q**. The motion of the satellite can then be considered with respect to an in-orbit inertial reference frame and the system potential energy is constant. Moreover, the zero point of the potential energy is arbitrary. Finally, the constant is taken equal to zero in the actual configuration and the Eq. 4.24 becomes:

$$\frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial T}{\partial \dot{q}_i}\right) - \frac{\partial T}{\partial q_i} = Q_i, \quad i = 1, \dots, N \tag{4.25}$$

Based on Moosavian and Papadopoulos' derivations [45] , the equations of motion are obtained:

$$\mathbf{H}(\boldsymbol{\delta}_0, \boldsymbol{\theta})\ddot{\mathbf{q}} + \mathbf{C}(\boldsymbol{\delta}_0, \dot{\boldsymbol{\delta}}_0, \boldsymbol{\theta}, \dot{\boldsymbol{\theta}}) = \mathbf{Q}(\boldsymbol{\delta}_0, \boldsymbol{\theta}) \tag{4.26}$$

$\mathbf{Q}$ are the generalized forces ($N \times 1$ vector) and can be written as:

$$\mathbf{Q} = \mathbf{J}_Q \begin{bmatrix} {}^0\mathbf{f}_s \\ {}^0\mathbf{n}_s \\ \boldsymbol{\tau}_{K\times 1} \end{bmatrix} = \mathbf{J}_Q \mathbf{u} \tag{4.27}$$

where:

- ${}^0\mathbf{f}_s$ the net force applied on the spacecraft
- ${}^0\mathbf{n}_s$ the net torque applied on the spacecraft
- $\boldsymbol{\tau}_{K\times 1}$ the vector of joint torques (K being the number of joints)
- $\mathbf{J}_Q$ is an $N \times N$ Jacobian matrix (detailed later on)

The matrix $\mathbf{H}$ is the total mass matrix of the system ($N \times N$) while $\mathbf{C}$ is an $N \times 1$ vector containing all the non-linear terms. Their expressions are derived in [45] and give the following results:

$$H_{ij} = M\frac{\partial \mathbf{R}_{C_0}}{\partial q_i} \cdot \frac{\partial \mathbf{R}_{C_0}}{\partial q_j} + \frac{{}^0\partial \boldsymbol{\omega}_0}{\partial \dot{q}_i} \cdot \mathbf{I}_0 \cdot \frac{{}^0\partial \boldsymbol{\omega}_0}{\partial \dot{q}_j} + \sum_{k=1}^{N}\left(m_k \frac{\partial \mathbf{r}_{C_k}}{\partial q_i} \cdot \frac{\partial \mathbf{r}_{C_k}}{\partial q_j} + \frac{{}^k\partial \boldsymbol{\omega}_k}{\partial \dot{q}_i} \cdot \mathbf{I}_k \cdot \frac{{}^k\partial \boldsymbol{\omega}_k}{\partial \dot{q}_j}\right) + \left(\sum_{k=1}^{N} m_k \frac{\partial \mathbf{r}_{C_k}}{\partial q_i}\right) \cdot \frac{\partial \mathbf{R}_{C_0}}{\partial q_j} + \left(\sum_{k=1}^{N} m_k \frac{\partial \mathbf{r}_{C_k}}{\partial q_j}\right) \cdot \frac{\partial \mathbf{R}_{C_0}}{\partial q_i} \tag{4.28}$$

and $\mathbf{C}$ can be written as:

$$\mathbf{C}(\boldsymbol{\delta}_0, \dot{\boldsymbol{\delta}}_0, \boldsymbol{\theta}, \dot{\boldsymbol{\theta}}) = \mathbf{C_1}(\boldsymbol{\delta}_0, \dot{\boldsymbol{\delta}}_0, \boldsymbol{\theta}, \dot{\boldsymbol{\theta}})\dot{\mathbf{q}} + \mathbf{C_2}(\boldsymbol{\delta}_0, \dot{\boldsymbol{\delta}}_0, \boldsymbol{\theta}, \dot{\boldsymbol{\theta}}) \tag{4.29}$$

where

$$C_{1_{ij}} = M\frac{\partial \mathbf{R}_{C_0}}{\partial q_i} \cdot \left(\sum_{s=1}^{N} \frac{\partial^2 \mathbf{R}_{C_0}}{\partial q_s \partial q_j}\dot{q}_s\right) + \frac{{}^0\partial \boldsymbol{\omega}_0}{\partial \dot{q}_i} \cdot \mathbf{I}_0 \cdot \frac{{}^0\partial \boldsymbol{\omega}_0}{\partial q_j} + \boldsymbol{\omega}_0 \cdot \mathbf{I}_0 \cdot \frac{{}^0\partial^2 \boldsymbol{\omega}_0}{\partial \dot{q}_i \partial q_j} + \frac{\partial \mathbf{R}_{C_0}}{\partial q_i} \cdot \sum_{k=1}^{N}\left(m_k \sum_{s=1}^{N} \frac{\partial^2 \mathbf{r}_{C_k}}{\partial q_s \partial q_j}\dot{q}_s\right) + \left(\sum_{s=1}^{N} \frac{\partial^2 \mathbf{R}_{C_0}}{\partial q_s \partial q_i}\dot{q}_s\right) \cdot \sum_{k=1}^{N}\left(m_k \frac{\partial \mathbf{r}_{C_k}}{\partial q_j}\right) + \sum_{k=1}^{N}\left(m_k \frac{\partial \mathbf{r}_{C_k}}{\partial q_i} \cdot \left(\sum_{s=1}^{N} \frac{\partial^2 \mathbf{r}_{C_k}}{\partial q_s \partial q_j}\dot{q}_s\right) + \frac{{}^k\partial \boldsymbol{\omega}_k}{\partial \dot{q}_i} \cdot \mathbf{I}_k \cdot \frac{{}^k\partial \boldsymbol{\omega}_k}{\partial q_j} + \boldsymbol{\omega}_k \cdot \mathbf{I}_k \cdot \frac{{}^k\partial^2 \boldsymbol{\omega}_k}{\partial \dot{q}_i \partial q_j}\right) \tag{4.30}$$

and

$$C_{2_i} = -\left(\boldsymbol{\omega}_0 \cdot \mathbf{I}_0 \frac{{}^0\partial \boldsymbol{\omega}_0}{\partial q_i} + \sum_{k=1}^{N} \boldsymbol{\omega}_k \cdot \mathbf{I}_k \cdot \frac{{}^k\partial \boldsymbol{\omega}_k}{\partial q_i}\right) \tag{4.31}$$

In the equations of $\mathbf{H}$, $\mathbf{C}_1$ and $\mathbf{C}_2$, some parameters appear and need to be defined. First of all, the angular velocities of the spacecraft and of the k^{th} link are $\boldsymbol{\omega}_0$ and $\boldsymbol{\omega}_k$ respectively. The angular velocity of any link k of the system can be derived as:

$$\boldsymbol{\omega}_k = \boldsymbol{\omega}_0 + \sum_{i=1}^{k} \dot{\boldsymbol{\theta}}_i \mathbf{z}_i, k = 1, ..., N_m \tag{4.32}$$

with N_m being the number of links of the arm considered ($N_m = 3$ here), $\dot{\boldsymbol{\theta}}_i$ the joint angle rate around $\mathbf{z}_i$, the unit vector along the axis of rotation of the i^{th} joint.
Moreover, the derivatives of the angular velocities of the different parts of the system are taken with respect to the generalized coordinates and their derivatives in these equations. More precisely, these

derivations are taken in different frames specified using a superscript. When there is none, the frame in which the derivation is done is the inertial frame. For example, ${}^{k}\partial\boldsymbol{\omega}_k/\partial q_i$ is the derivative of $\boldsymbol{\omega}_k$ with respect to q_i taken in the k reference frame. To do so, $\boldsymbol{\omega}_k$ has to be expressed in that reference frame:

$$ {}^{k}\boldsymbol{\omega}_k = \mathbf{A}^T_{k-1,k}\mathbf{A}^T_{k-2,k-1}...\mathbf{A}^T_{0,1}\mathbf{S}_0\boldsymbol{\delta}_0 + \sum_{s=1}^{k-1}(\mathbf{A}^T_{k-1,k}\mathbf{A}^T_{k-2,k-1}...\mathbf{A}^T_{s,s+1}\dot{\boldsymbol{\theta}}_s\mathbf{z}_s) + \dot{\boldsymbol{\theta}}_k\mathbf{z}_k \tag{4.33} $$

where $\mathbf{A}_{i-1,i}$ is the rotation matrix between the i^{th} the $(i-1)^{th}$ body-fixed reference frames and the vectors $\mathbf{z}_i$ are unit vectors along the axis of rotation of the i^{th} expressed in the i^{th} body-fixed reference frame. The different derivatives of the angular velocities - such as ${}^{k}\partial\boldsymbol{\omega}_k/\partial q_i$ - can be derived symbolically from the expression given in Eq. 4.33.
Also, $\mathbf{r}_{C_k}$ is defined as the distance between the spacecraft CoM and the CoM of link k. Based on Fig. 4.6, the expression of this distance is:

$$ \mathbf{r}_{C_k} = \mathbf{r}_0 + \sum_{i=1}^{k-1}(\mathbf{r}_k - \mathbf{l}_k) - \mathbf{l}_i, i = 1, ..., N_m \tag{4.34} $$

In the latter, $\mathbf{l}_k$ and $\mathbf{r}_k$ describe the position of joints k (the previous one) and $k+1$ (the next one) with respect to the center of mass of the link k, C_k. Similarly, $\mathbf{r}_0$ describes the position of the first joint with respect to the spacecraft CoM.

Using Eq. 4.43, the angular velocity can be expressed as a function of the Euler rates. Thus Eq. 4.31 can be rewritten as:

$$ \mathbf{C}_2(\boldsymbol{\delta}_0, \dot{\boldsymbol{\delta}}_0, \boldsymbol{\theta}, \dot{\boldsymbol{\theta}}) = \tilde{\mathbf{C}}_2(\boldsymbol{\delta}_0, \dot{\boldsymbol{\delta}}_0, \boldsymbol{\theta}, \dot{\boldsymbol{\theta}})\dot{\mathbf{q}} \tag{4.35} $$

and Eq. 4.29 as follows:

$$ \mathbf{C}(\boldsymbol{\delta}_0, \dot{\boldsymbol{\delta}}_0, \boldsymbol{\theta}, \dot{\boldsymbol{\theta}}) = \mathbf{C}_3(\boldsymbol{\delta}_0, \dot{\boldsymbol{\delta}}_0, \boldsymbol{\theta}, \dot{\boldsymbol{\theta}})\dot{\mathbf{q}} \tag{4.36} $$

where

$$ \mathbf{C}_3(\boldsymbol{\delta}_0, \dot{\boldsymbol{\delta}}_0, \boldsymbol{\theta}, \dot{\boldsymbol{\theta}}) = \mathbf{C}_1(\boldsymbol{\delta}_0, \dot{\boldsymbol{\delta}}_0, \boldsymbol{\theta}, \dot{\boldsymbol{\theta}}) + \tilde{\mathbf{C}}_2(\boldsymbol{\delta}_0, \dot{\boldsymbol{\delta}}_0, \boldsymbol{\theta}, \dot{\boldsymbol{\theta}}) \tag{4.37} $$

It is important to note that the previous transformation of $\mathbf{C}_2$, hence the expression of $\mathbf{C}_3$, are not unique. To illustrate this statement, the example of a spacecraft without any manipulator can be taken (thus a system with six degrees of freedom). In that case and with the assumption that the inertia tensor of the base is a diagonal matrix with the same diagonal terms J, then, the overall $\mathbf{C}$ matrix can be written:

$$ \mathbf{C} = \begin{bmatrix} 0 \\ 0 \\ 0 \\ J\dot{\psi}\dot{\theta}\cos(\theta) \\ -J\dot{\phi}\dot{\psi}\cos(\theta) \\ J\dot{\phi}\dot{\theta}\cos(\theta) \end{bmatrix} \tag{4.38} $$

The focus can be put on the three last components, which are the rotational components, for the sake of clarity. For instance, it is clear that:

$$ \begin{aligned} \mathbf{C}_{rot} &= \begin{bmatrix} J\dot{\psi}\dot{\theta}\cos(\theta) \\ -J\dot{\phi}\dot{\psi}\cos(\theta) \\ J\dot{\phi}\dot{\theta}\cos(\theta) \end{bmatrix} \\ &= \begin{bmatrix} 0 & J\dot{\psi}\cos(\theta) & 0 \\ 0 & 0 & -J\dot{\phi}\cos(\theta) \\ J\dot{\theta}\cos(\theta) & 0 & 0 \end{bmatrix} \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} \\ &= \begin{bmatrix} 0 & 0 & J\dot{\theta}\cos(\theta) \\ -J\dot{\psi}\cos(\theta) & 0 & 0 \\ 0 & J\dot{\phi}\cos(\theta) & 0 \end{bmatrix} \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} \end{aligned} \tag{4.39} $$

There are in that case eight different possibilities to factorize the equation. Eq. 4.26 is then equivalent to:

$$\mathbf{H}(\boldsymbol{\delta}_0,\boldsymbol{\theta})\ddot{\mathbf{q}} + \mathbf{C}_3(\boldsymbol{\delta}_0,\dot{\boldsymbol{\delta}}_0,\boldsymbol{\theta},\dot{\boldsymbol{\theta}})\dot{\mathbf{q}} = \mathbf{Q}(\boldsymbol{\delta}_0,\boldsymbol{\theta}) \tag{4.40}$$

in which, based on Eq. 4.27, $\mathbf{Q}$ can be written using the Jacobian matrix $\mathbf{J}_Q$.

$$\mathbf{J}_Q = \begin{bmatrix} \mathbf{T}_0 & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times K} \\ \mathbf{0}_{3\times3} & \mathbf{S}_0^T & \mathbf{0}_{3\times K} \\ \mathbf{0}_{K\times3} & \mathbf{0}_{3\times3} & \mathbf{1}_{K\times K} \end{bmatrix} \tag{4.41}$$

where

- $\mathbf{T}_0 = \mathbf{I}_3$
- $\mathbf{S}_0$ describes the relationship between the angular rates into Euler rates [17] such that:

  $$^0\boldsymbol{\omega}_0 = \mathbf{S}_0\left(\boldsymbol{\delta}_0\right)\dot{\boldsymbol{\delta}}_0 \tag{4.42}$$

  where

  $$\mathbf{S}_0 = \begin{bmatrix} \cos(\theta)\cdot\cos(\psi) & \sin(\psi) & 0 \\ -\cos(\theta)\cdot\sin(\psi) & \cos(\psi) & 0 \\ \sin(\theta) & 0 & 1 \end{bmatrix} \tag{4.43}$$

At the end, the system of equations of motion 4.40 is obtained. It is a general non-linear model of the system dynamics. The design of the controller is based on the matrices $\mathbf{H}$, $\mathbf{C}_3$ and $\mathbf{J}_Q$. However, these matrices have to be constant because only one controller will be developed and it is assumed not to be a time-varying one. Thus, the matrices, dependant on $\boldsymbol{\delta}_0$, $\dot{\boldsymbol{\delta}}_0$, $\boldsymbol{\theta}$ and $\dot{\boldsymbol{\theta}}$, have to be considered in a certain configuration, as detailed in Section 4.3.3. Because it is not unique, the factorization of $\mathbf{C}_2$ by $\dot{\boldsymbol{\delta}}_0$ will influence the results and the behaviour of the linearized system.

4.3.2. Verification of the Dynamics Model

The method developed here based on [45] is very powerful when studying such a complex system. Having a symbolic model of the system is very handy since it can be saved and variations in terms of system specifications can be easily implemented. The dynamics is also developed in such a way that it is straightforward to develop the state-space model of the system as will be shown in Section 4.3.3. However, the results obtained shall first be verified. To do so, a second method based on the Lagrangian mechanics is first developed and then a simplified system with six DoF is modeled using Newton's mechanics and it is compared with a model based on the method described in 4.3.1.

- **Modeling based on the Direct Derivation of the Lagrangian**

The second method, called the *direct* method, used to derive a model of the dynamics of the system is also based on the Lagrangian mechanics *i.e.* it starts with the computation of the non-relativistic Lagrangian L. However, this second modeling method directly replaces L (which is equal to the kinetic energy T in this case) in Eq. 4.25 and computes the different terms of the equation. Once derived, the terms are ordered such that the results are comparable: the ones dependent on the second derivative of $\mathbf{q}$ are taken apart such that the matrices $\mathbf{H}$ and $\mathbf{C}$ from Eq. 4.25 can be written. Because the derivation in this second method of the equations of motion is very direct, the equations are larger than in the first method, in which they were reduced in size thanks to the optimization of the equations detailed by Moosavian and Papadopoulos. Thus, comparing the symbolic expressions is not conveniently doable. Numerical comparisons are run instead. For each comparison, a set of parameters on which the matrices depend is taken and substituted in the symbolic matrices obtained. Several tests are done in which the constant parameters of the system (masses, inertia tensors of the links or dimensions) are set and a different set of variables is taken at every iteration. The $\mathbf{H}$ and $\mathbf{C}$ matrices for both methods are compared by subtracting one to the other and taken the norm of the difference matrix. The smaller the norm, the closer the two matrices to each other. It can be clarified with the following equations:

$$\begin{cases} \Delta\mathbf{H} = \mathbf{H}^\circ - \mathbf{H}^\bullet \\ \Delta\mathbf{C} = \mathbf{C}^\circ - \mathbf{C}^\bullet \end{cases} \tag{4.44}$$

In the latter, $\mathbf{H}^\circ$ and $\mathbf{C}^\circ$ are the matrices obtained using Moosavian's and Papadopoulos' paper while the matrices $\mathbf{H}^\bullet$ and $\mathbf{C}^\bullet$ are computed using the direct method. The results are shown Fig. 4.7. It shows that the norm of the difference between the two methods is very small so the results obtained using the method detailed in [45] are coherent with respect to the results derived with the second, more direct, method.

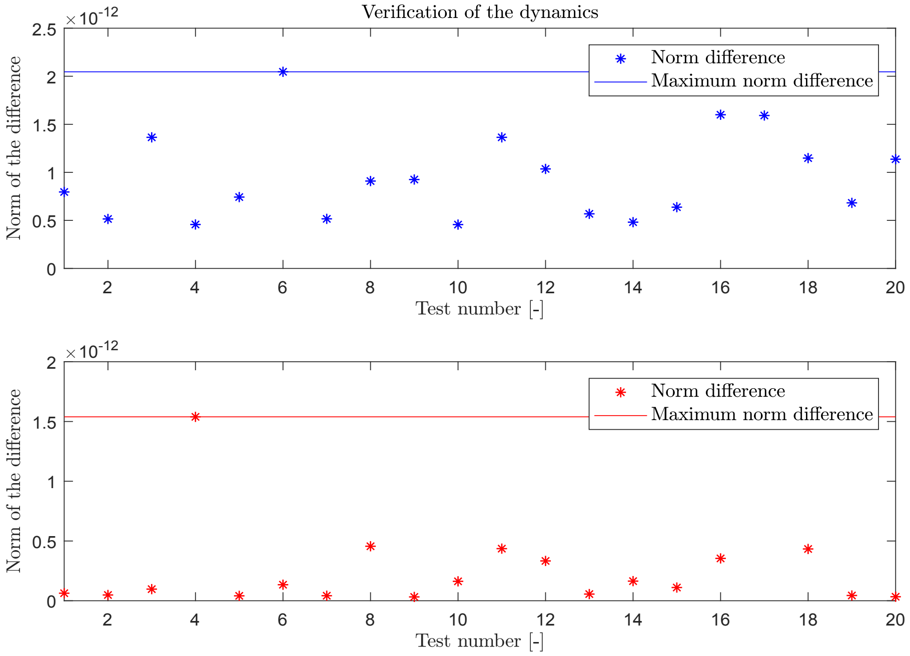


Figure 4.7: Comparison of results from both Lagrangian methods (up: $\|\Delta\mathbf{H}\|$, down: $\|\Delta\mathbf{C}\|$)

• Modeling of a Six DoF Model based on Newton's Mechanics

However, the two methods are based on the method discovered by Lagrange which means that they are both derived from the expression of the kinetic energy. The definition of the latter should be verified too. To do so, a dynamics model of the system without any manipulator is developed using Newton's and Euler's equations. The objective is to show that both methods (using Lagrangian and Newtonian mechanics) are equivalent for a six degree of freedom system consisting in the spacecraft only. According to Newton's second law and Euler's laws of motion, the following system of equations can be written:

$$\begin{cases} \mathbf{F} = m \cdot \mathbf{a}_G \\ \mathbf{T}_G = \mathbf{J}_G^B \cdot \dot{\boldsymbol{\omega}} + \boldsymbol{\omega} \times \mathbf{J}_G^B \boldsymbol{\omega} \end{cases} \tag{4.45}$$

In Eq. 4.45, $\mathbf{F}$ and $\mathbf{T}_G$ are respectively the force vector and the torque vector acting on and around the center of mass of the base. m is the mass of the base, $\mathbf{J}_G^B$ is the inertia tensor of the base at its center of mass G expressed in the base body-fixed reference frame B and $\boldsymbol{\omega}$ is the angular velocity of the base in the same reference frame. Eq. 4.45 can be written using matrices:

$$\begin{bmatrix} \mathbf{F} \\ \mathbf{T} \end{bmatrix} = \begin{bmatrix} m\mathbf{I}_3 & \mathbf{0} \\ \mathbf{0} & \mathbf{J}_G^B \end{bmatrix} \begin{bmatrix} \dot{\mathbf{v}} \\ \dot{\boldsymbol{\omega}} \end{bmatrix} + \begin{bmatrix} \mathbf{0} \\ \boldsymbol{\omega} \times \mathbf{J}_G^B \boldsymbol{\omega} \end{bmatrix} \tag{4.46}$$

Both $\mathbf{v}$ and $\boldsymbol{\omega}$ can be written with respect to the generalized states contained in $\dot{\mathbf{q}}$. In fact, the following relationships are known:

$$\begin{cases} \mathbf{v} = \begin{bmatrix} \mathbf{I}_3 & \mathbf{0} \end{bmatrix} \dot{\mathbf{q}} = \mathbf{J}_v \dot{\mathbf{q}} \\ \boldsymbol{\omega} = \begin{bmatrix} \mathbf{0} & \mathbf{S}_0 \end{bmatrix} \dot{\mathbf{q}} = \mathbf{J}_\omega \dot{\mathbf{q}} \end{cases} \tag{4.47}$$

where $\mathbf{S}_0$ is as defined previously, in Eq. 4.43. If $\mathbf{J}$ is defined as $\mathbf{J} = \begin{bmatrix}\mathbf{J}_v & \mathbf{J}_\omega\end{bmatrix}^T$ Eq. 4.46 can then be rewritten [42]:

$$\begin{bmatrix}\mathbf{F}\\ \mathbf{T}\end{bmatrix} = \begin{bmatrix} m\mathbf{I}_3 & \mathbf{0}\\ \mathbf{0} & \mathbf{J}_G^B\end{bmatrix}(\dot{\mathbf{J}}\dot{\mathbf{q}} + \mathbf{J}\ddot{\mathbf{q}}) + \begin{bmatrix}\mathbf{0}\\ \mathbf{J}_\omega\dot{\mathbf{q}} \times \mathbf{J}_\omega\mathbf{J}_G^B\dot{\mathbf{q}}\end{bmatrix} \tag{4.48}$$

By pre-multiplying the latter by the transpose of the Jacobian matrix $\mathbf{J}$, one obtains:

$$(\mathbf{J}^T\mathbf{M}_c\mathbf{J})\ddot{\mathbf{q}} + (\mathbf{J}^T\mathbf{M}_c\dot{\mathbf{J}} + \mathbf{J}^T\tilde{\boldsymbol{\omega}}\mathbf{M}_c\mathbf{J})\dot{\mathbf{q}} = \mathbf{J}_v^T\mathbf{F} + \mathbf{J}_\omega^T\mathbf{T} \tag{4.49}$$

in which:

- $\mathbf{M}_c = \begin{bmatrix} m\mathbf{I}_3 & \mathbf{0}\\ \mathbf{0} & \mathbf{J}_G^B\end{bmatrix}$
- $\tilde{\boldsymbol{\omega}} = \begin{bmatrix}\mathbf{0} & \mathbf{0}\\ \mathbf{0} & (\mathbf{J}_\omega\dot{\mathbf{q}})^x\end{bmatrix}$

By analogy with Eq. 4.25, the following can be written:

$$\begin{cases} \mathbf{H} = \mathbf{J}^T\mathbf{M}_c\mathbf{J}\\ \mathbf{C} = (\mathbf{J}^T\mathbf{M}_c\dot{\mathbf{J}} + \mathbf{J}^T\tilde{\boldsymbol{\omega}}\mathbf{M}_c\mathbf{J})\dot{\mathbf{q}}\\ \mathbf{Q} = \mathbf{J}_v^T\mathbf{F} + \mathbf{J}_\omega^T\mathbf{T} = \begin{bmatrix}\mathbf{J}_v^T & \mathbf{J}_\omega^T\end{bmatrix}\begin{bmatrix}\mathbf{F}\\ \mathbf{T}\end{bmatrix}\end{cases} \tag{4.50}$$

Comparing the last equation of Eq. 4.50 with Eq. 4.27 gives: $\mathbf{J}_Q = \mathbf{J}^T$. The inertia tensor of the base $\mathbf{J}_G^B$ is hereafter, as in RICADOS project, taken as a diagonal matrix. For the sake of clarity, the diagonal terms are assumed to be equal to J:

$$\mathbf{J}_G^B = J\mathbf{I}_3 \tag{4.51}$$

Finally, these matrices can be compared to the ones obtained using the Lagrangian method [45]. Because the system is less complex in that case, symbolic expressions can directly be compared and are found to be the same:

$$\mathbf{H} = \begin{bmatrix} m & 0 & 0 & 0 & 0 & 0\\ 0 & m & 0 & 0 & 0 & 0\\ 0 & 0 & m & 0 & 0 & 0\\ 0 & 0 & 0 & J & 0 & J\sin(\theta)\\ 0 & 0 & 0 & 0 & J & 0\\ 0 & 0 & 0 & J\sin(\theta) & 0 & J\end{bmatrix} \tag{4.52}$$

$$\mathbf{C} = \begin{bmatrix} 0\\ 0\\ 0\\ J\dot{\psi}\dot{\theta}\cos(\theta)\\ -J\dot{\phi}\dot{\psi}\cos(\theta)\\ J\dot{\phi}\dot{\theta}\cos(\theta)\end{bmatrix} \tag{4.53}$$

and

$$\mathbf{J} = \begin{bmatrix}\mathbf{I}_3 & \mathbf{0}\\ \mathbf{0} & \mathbf{S}_0\end{bmatrix} \tag{4.54}$$

It was shown that both the Lagrangian and the Newtonian methods give the same results for a system with six degrees of freedom. It means that the definition of the kinetic energy is correct for such a system. Adapting the expression of the kinetic energy to a larger system is straightforward so that the two methods are considered equivalent.

4.3.3. State-Space Representation

In this section, the state-space model of the system is first derived based on the dynamics modeling of the system described in the previous sections. It is then verified using a numerical model on Simscape and finally, the linear plant obtained is studied before being used for the design of the controller in the next chapter.

- **Derivation of the State-Space Model**

The dynamics model of the system was developed and verified in the previous sections 4.3.1 and 4.3.2. This model is complex and highly non-linear. It has nine inputs (AOCS forces and torques as well as joint torques) and nine outputs contained in the state vector $\mathbf{q}$ described in Section 2.3.4. The equations of motion described in Eq. 4.40 show the non-linearity of the system. Indeed $\mathbf{H}$, $\mathbf{C}$ and $\mathbf{Q}$ matrices vary with $\boldsymbol{\delta}_0$, $\dot{\boldsymbol{\delta}}_0$, $\boldsymbol{\theta}$ and $\dot{\boldsymbol{\theta}}$. The two first matrices are respectively multiplied by $\ddot{\mathbf{q}}$ and $\dot{\mathbf{q}}$ which involves non-linearities. Furthermore, there are also non-linearities within the matrices themselves as shown in the example in Eq. 4.38.

The simplest way to design the controller is to use only one controller designed for the whole procedure (*i.e.* no gain-scheduling). Thus the matrices of the equations of motion have to be constant. They have to be computed for a chosen configuration. The latter is the final configuration, that means the configuration expected when the end-effector reaches the handle on the target satellite. The point of choosing this configuration instead of any other one is that the maximum precision should be achieved when the end-effector is close to the required position (i.e. the handle on the target satellite) in order to avoid any collision between the two satellites.

A tool has been developed and is used as a pre-study of the scenario (Fig. 4.8). The objective is to determine if the handle position (given as input) is within the workspace of the robotic arm and can be reached while keeping the relative distance and attitude between the chaser and the target constant. In its stretched configuration, the end-effector is on the upper-edge of the workspace $\mathbf{p}_{EE_0}$. The position of the handle defined is shown as being within the workspace. It is reachable for the end-effector without moving the base. The center of the workspace corresponds to the upper-end of the second link, as shown in Fig. 4.2.

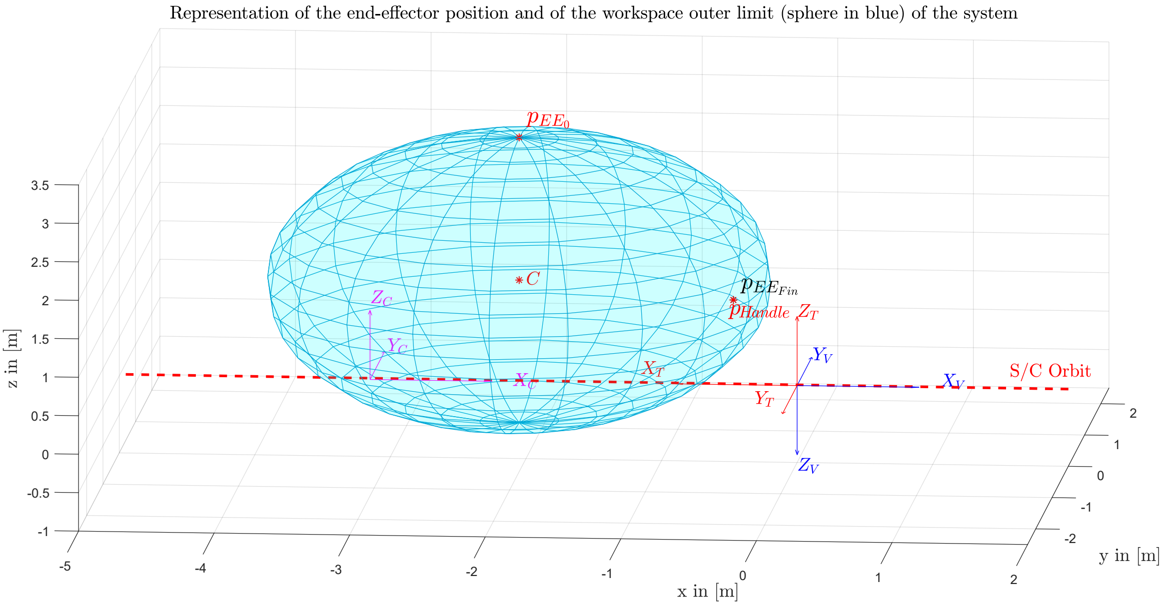


Figure 4.8: Tool used to determine the linearization configuration

Even if, as shown previously, the matrices do not depend on $\mathbf{R}_{C_0}$ nor its derivative $\dot{\mathbf{R}}_{C_0}$, the preliminary study requires $\mathbf{R}_{C_0}$ and the position of the handle in order to compute the linearization configuration. These parameters are taken as follows:

$$\begin{cases} R_{C_{0_x}} = 3.15 \\ R_{C_{0_y}} = 0 \\ R_{C_{0_z}} = 0 \end{cases} \quad \text{in m} \tag{4.55}$$

and

$$\mathbf{p}_{Handle} = \begin{bmatrix} 0.47 \\ 0 \\ 1.10 \end{bmatrix} \text{ in m} \tag{4.56}$$

$\mathbf{R}_{C_0}$ and $\mathbf{p}_{Handle}$ are expressed here in the target BF reference frame. Fig. 4.8 shows that this configuration puts the handle within the workspace of the arm without any motion of the base. The axes of this figure are expressed in the LVLH reference frame. Based on the assumption that the base is controlled to be fixed in the inertial reference frame, the joint angles of the arm allowing the end-effector to reach the handle position are computed. Thus, the attitude angles $\boldsymbol{\delta}_0$ and the joint angles $\boldsymbol{\theta}$ corresponding to the moment when the end-effector reaches the handle are known:

$$\begin{cases} \phi_f = 0 \\ \theta_f = 0 \\ \psi_f = \pi \\ \theta_{1_f} = 0 \\ \theta_{2_f} = -1.0839 \\ \theta_{3_f} = 1.0839 \end{cases} \text{ in rad}$$

So the state vector corresponding to the linearization configuration is:

$$\mathbf{q}_f = \begin{bmatrix} 3.15 & 0 & 0 & 0 & 0 & \pi & 0 & -1.0839 & 1.0839 \end{bmatrix} \tag{4.57}$$

The six first states are expressed in the target BF reference frame while the three joint angles are expressed in their respective joint's RF. The TCP is where the gripper is fixed on the robotic arm while the EE is at the CoM of the gripper. Thus, the TCP has to be a bit further away from the handle such that there is some room in between the TCP and the handle for the gripper. When the CoM of the gripper is at the handle, the TCP is as shown Fig. 4.9.It corresponds to the final configuration of the system chosen. The geometry of the servicer is simplified in the model compared to the one depicted on this figure. Also, the gripper is not represented on this visual for the sake of clarity. Indeed, the Z-axis of the last joint (in green) is used to check that the handle is attainable. If it goes through the circular ring, it means that when the gripper is added, the handle can be grasped. Finally, the way the gripper goes *around* the handle was not studied nor its orientation. However, a model of the gripper is used in the dynamics modelling. The last link considered in the model corresponds to the *actual* last link plus the gripper which is fixed to it.

Because the configuration considered is the final configuration, the angular rates of the joint angles shall be equal to zero such that the position of the end-effector does not vary anymore and the grasping of the handle could be completed by the team in charge at DLR. Concerning the attitude angular velocities, the assumption taken in Section 2.4 stating that the tumbling of the target is not considered yet implies that the step of synchronization of the two spacecrafts consists in keeping the attitude of the chaser constant. Thus, the first derivatives of the chaser's Euler angles and of the joint angles in the linearization configuration should be:

$$\begin{cases} \dot{\varphi}_f = 0 \\ \dot{\theta}_f = 0 \\ \dot{\psi}_f = 0 \\ \dot{\theta}_{1_f} = 0 \\ \dot{\theta}_{2_f} = 0 \\ \dot{\theta}_{3_f} = 0 \end{cases} \text{ in rad/s}$$

The matrices evaluated with the configuration given above are noted $\mathbf{H}_0$, $\mathbf{C}_{3_0}$ and $\mathbf{J}_{Q_0}$. The linearized version of Eq. 4.40 is as follows:

$$\mathbf{H}_0 \cdot \ddot{\mathbf{q}} + \mathbf{C}_{3_0} \cdot \dot{\mathbf{q}} = \mathbf{J}_{Q_0} \cdot \mathbf{u} \tag{4.58}$$

Linearizing the plant represents a loss of information and of precision. Indeed, the non-linear phenomenons present in the non-linear model are now reduced to linear representations.

The mass matrix $\mathbf{H}$ is symmetric and so is $\mathbf{H}_0$. Moreover, it is positive definite and hence invertible. The previous equation is equivalent to:

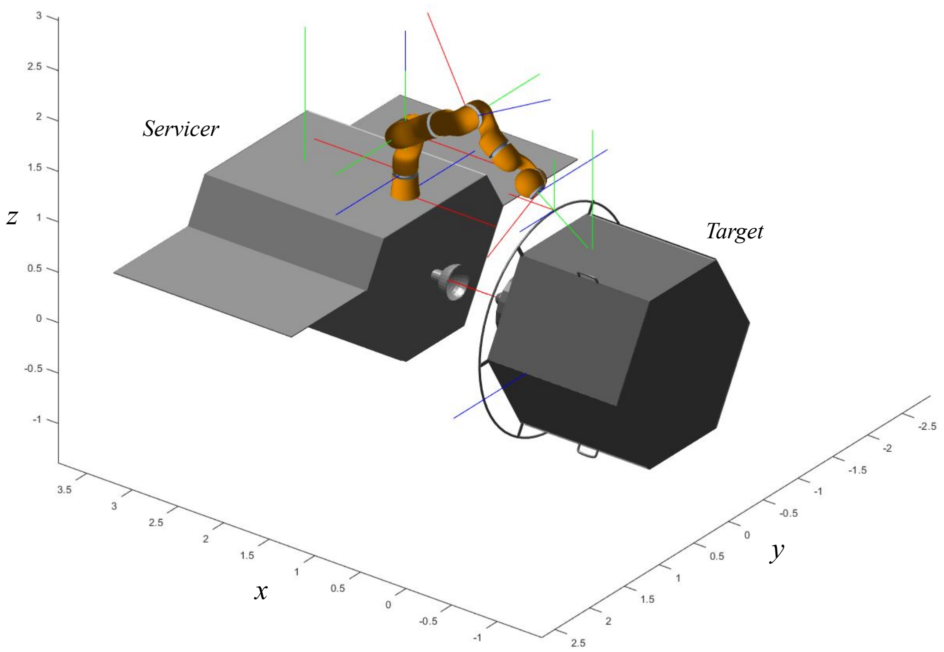


Figure 4.9: Final configuration of the system

$$\ddot{\mathbf{q}} = \mathbf{H}_0^{-1} \cdot (\mathbf{J}_{Q_0} \cdot \mathbf{u} - \mathbf{C}_{3_0} \cdot \dot{\mathbf{q}}) \tag{4.59}$$

Due to the complexity of the symbolic matrix $\mathbf{H}$, it is linearized before taking the inverse. This is appropriate since the differences between the matrix for which the linearization is completed before taking the inverse and the matrix obtained with linearization after taking the inverse are of the order of magnitude of the floating-point relative accuracy. Indeed, evaluating a matrix $\mathbf{A}$ for $\mathbf{x} = \mathbf{x}_0$ before taking its inverse or after is equivalent (with the condition that the matrix is invertible for the evaluation point):

$$\left(\mathbf{A}(\mathbf{x})\big|_{\mathbf{x}=\mathbf{x}_0}\right)^{-1} = \mathbf{A}(\mathbf{x})^{-1}\big|_{\mathbf{x}=\mathbf{x}_0} \tag{4.60}$$

The state vector $\mathbf{x}$ and the control vector $\mathbf{u}$ can be used to rewrite the previous system of equations, it is the state equation of the state-space representation.

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u} \tag{4.61}$$

where

$$\begin{cases} \mathbf{A} = \begin{bmatrix} \mathbf{0}_{9\times9} & \mathbf{I}_{9\times9} \\ \mathbf{0}_{9\times9} & -\mathbf{H}_0^{-1}\mathbf{C}_{3_0} \end{bmatrix} \\ \mathbf{B} = \begin{bmatrix} \mathbf{0}_{9\times9} \\ \mathbf{H}_0^{-1}\mathbf{J}_{Q_0} \end{bmatrix} \end{cases}$$

As a first model, the measured outputs are assumed to be ideal. It can then be written as:

$$\mathbf{y} = \mathbf{C}\mathbf{x} + \mathbf{D}\mathbf{u} \tag{4.62}$$

where

$$\begin{cases} \mathbf{C} = \begin{bmatrix} \mathbf{I}_{9\times9} & \mathbf{0}_{9\times9} \end{bmatrix} \\ \mathbf{D} = \mathbf{0}_{9\times9} \end{cases}$$

It expresses the measured output vector of the plant $\mathbf{y}_m$ to the state vector. The whole state vector cannot be measured and there is no direct influence of the control inputs on the measured outputs.

Indeed, $\mathbf{y}_m$ is as follows:

$$\mathbf{y}_m = \begin{bmatrix} \mathbf{R}_{C_0} \\ \boldsymbol{\delta}_0 \\ \boldsymbol{\theta} \end{bmatrix} \tag{4.63}$$

Finally, the state-space plant $\mathbf{G}$ consists of the matrices $\mathbf{A}$, $\mathbf{B}$, $\mathbf{C}$ and $\mathbf{D}$ derived in this section.

- ### Verification of the State-Space Model

In addition to the verification of the dynamics using the Newton mechanics for the six DoF system, a numerical modelization is made using Simulink's multi-body mechanics tool, Simscape. Using blocks to define the different mechanical parts, interactions, sensors and actuators of the system, the system is modelled. The system $\{S/C + Arm\}$, as modelled on Simscape, is shown Fig. A.1 in Appendix A.
The system is nonlinear and can be linearized using *linmod* function in order to be comparable with the symbolic model described above. The result is a state-space representation of the nine DoF system. The $\mathbf{A}$ matrix is a 18×18 matrix:

$$\mathbf{A} = \begin{bmatrix} \mathbf{0}_{9\times9} & \mathbf{I}_{9\times9} \\ \mathbf{0}_{9\times9} & \mathbf{0}_{9\times9} \end{bmatrix} \tag{4.64}$$

It is coherent with the symbolic modelization since the $\mathbf{C}_{3_0}$ matrix appears in the bottom-right block of the matrix. As said before, every term in $\mathbf{C}_3$ consists in a multiplication with one of the joint angles derivative. These are taken equal to zero in the linearization configuration so $\mathbf{C}_{3_0} = \mathbf{0}_{9\times9}$. The $\mathbf{B}$ matrix found using Simscape is given below:

$$\mathbf{B} = \begin{bmatrix} \mathbf{0}_{9\times9} \\ \mathbf{B}_2 \end{bmatrix} \tag{4.65}$$

where $\mathbf{B}_2$ is a 9×9 matrix too. It can be compared with the bottom part of the $\mathbf{B}$ matrix found using the method developed in [45]. The results found are very close. The differences for the large components of the matrices are within two percent. Some components are a bit further from each other but these are very small values (of the order of 10^{-11}). The unit of a component of the $\mathbf{B}$ matrix is given by its row and its column. The matrices and the components' units are given in Appendix A.
Finally, $\mathbf{C}$ and $\mathbf{D}$ matrices are also equal for both modelizations. The model developed for the nine DoF system, based on [45] and linearized around the configuration chosen is used in the next chapter to design a combined and robust controller. Before designing the controller, the model of the plant shall be studied in order to make sure that the results are coherent and that the design of the controller is possible.

- ### Study of the Linearized Plant

The study of the dynamic behaviour of the plant is done. The poles of the state-space representation of the system are the eigenvalues of the $\mathbf{A}$ matrix described before. All the poles are at the origin. They correspond to pure integration terms in the transfer function. The system is then not stable. It is clearer when looking at the step response of the system Fig. A.2 in Appendix B. First of all, once receiving a step, the excited outputs diverge. Furthermore, it is clear that the base actuators mostly influence the dynamics of the base itself but not of the arm. More precisely, any force or torque excites the corresponding state only (e.g. a force along F_x only results in a velocity along the x-axis without any velocities on the other axes nor any significant rotational motion). Finally, concerning the arm actuators, when a torque is applied to a joint, the effect is seen on various states, with different amplitudes. The system is not fully decoupled. For instance, when a torque is applied on the third joint, the position of the spacecraft varies along the x- and z-axes, its attitude is influenced too with the second Euler angle θ_o varying and the second joint angle θ_{2_o} is also influenced. In addition to that, the last joint angle is undergoing a large amplitude variation.
The outputs have different units and different meanings. Their amplitude cannot be directly compared. However, one need to know what output is the most affected one when a certain input is applied to the system. Indeed, this is necessary to understand how to control the system. To do so, the study of the

Relative Gain Array (RGA) is conducted. This tool gives very useful information on the control properties of a plant [55]. More precisely, it helps measuring the interactions between the input/output (I/O) couplings. It is defined as:

$$\mathbf{RGA}(\mathbf{G}) = \mathbf{G} \times (\mathbf{G}^{-1})^T \tag{4.66}$$

where × denotes element-by-element multiplication and $\mathbf{G}$ is the matrix of transfer functions between each input and each output, obtained from the plant state-space representation. Its rows and columns sum to one. The evolution of the RGA can be studied but its most important value is for crossover frequencies. Here, the RGA is constant with respect to frequencies. It is equal to the following:

$$\begin{aligned}
&\mathbf{RGA}(\mathbf{G})\\
&= \begin{bmatrix}
1.0057 & 0 & 0 & 0 & -0.0004 & 0 & 0 & -0.0005 & -0.0048\\
0 & 1.0095 & 0 & -0.0004 & 0 & -0.0004 & -0.0087 & 0 & 0\\
0 & 0 & 1.0093 & 0 & -0.0003 & 0 & 0 & -0.0114 & 0.0024\\
0 & -0.0004 & 0 & 1.022 & 0 & -0.0011 & -0.0205 & 0 & 0\\
-0.0004 & 0 & -0.0003 & 0 & 1.116 & 0 & 0 & -0.154 & 0.0386\\
0 & -0.0004 & 0 & -0.0011 & 0 & 1.0887 & -0.0872 & 0 & 0\\
0 & -0.0087 & 0 & -0.0205 & 0 & -0.0872 & 1.1164 & 0 & 0\\
-0.0005 & 0 & -0.0114 & 0 & -0.1540 & 0 & 0 & 3.0831 & -1.9173\\
-0.0048 & 0 & 0.0024 & 0 & 0.0386 & 0 & 0 & -1.9173 & 2.8810
\end{bmatrix}
\end{aligned}$$

For each row and each column of the RGA matrix, the maximum magnitude is on the diagonal. It means that each actuation action is capable of controlling one of the outputs. Of course, these are coupled so far but centralized control is possible.

Moreover, some of the conditions on the plant necessary to ensure that an H_∞ controller which stabilizes the plant are given hereafter. The plant shall be stabilizable and detectable according to [6].

First of all the controllability of the states is studied. If the plant is controllable, it is stabilizable too since the notion of stabilizability is weaker than the controllability. The controllability matrix is computed:

$$\mathbf{Co} = \begin{bmatrix}\mathbf{B} & \mathbf{AB} & \cdots & \mathbf{A}^{n-1}\mathbf{B}\end{bmatrix} \tag{4.67}$$

where $\mathbf{A}$ is an $n \times n$. Here, $n = 18$. To be controllable, the controllability matrix $\mathbf{Co}$ of the plant shall have full rank, $rank(\mathbf{Co}) = 18$. The rank of $\begin{bmatrix}\mathbf{B} & \mathbf{AB}\end{bmatrix}$ is equal to 18 so the controllability has full rank. Sometimes, even though $\mathbf{Co}$ has full rank, the system is close to be uncontrollable hence making it difficult to design a controller. To check how controllable the system is, a further study of the $\mathbf{Co}$ matrix can be done and a singular value decomposition can be conducted on the reduced controllability matrix $\begin{bmatrix}\mathbf{B} & \mathbf{AB}\end{bmatrix}$ which is square. Since $\mathbf{A}$ is an empty matrix except for its upper-right block which is an identity matrix, the product of $\mathbf{A}$ and $\mathbf{B}$ is an 18×9 matrix with an empty bottom part and an upper part equal to the bottom section of the $\mathbf{B}$ matrix as put into Eq. 4.68.

$$\begin{aligned}
\mathbf{Co}_{reduced} &= \begin{bmatrix}\mathbf{B} & \mathbf{AB}\end{bmatrix}\\
&= \begin{bmatrix}\mathbf{0}_{9\times9} & \mathbf{B}_2\\ \mathbf{B}_2 & \mathbf{0}_{9\times9}\end{bmatrix}
\end{aligned} \tag{4.68}$$

As shown in Appendix A, every row of $\mathbf{B}_2$ has a relatively large component which means that each component of $\dot{\mathbf{x}}$ is influenced by at least one component of the control input vector $\mathbf{u}$. When the singular value decomposition of the $\mathbf{Co}_{reduced}$ matrix is taken, the singular values are in between $6.5 * 10^{-4}$ and 0.32. As shown in [33], the rank of $\mathbf{Co}_{reduced}$ equals the number of nonzero singular values. Moreover, the lower the singular values, the closer to uncontrollability. In this case, the singular values are small but far from being of the order of the floating-point relative accuracy of MATLAB. It means that the plant is in fact controllable thus stabilizable.

The observability of the states is then studied. Indeed, if all the states are observable, the plant is observable hence detectable since the latter is a weaker condition than the observability. The observability matrix $\mathbf{O}$ is computed:

$$\mathbf{O} = \begin{bmatrix} \mathbf{C} \\ \mathbf{CA} \\ \vdots \\ \mathbf{CA}^{n-1} \end{bmatrix} \tag{4.69}$$

The linearized plant is observable if the $\mathbf{O}$ matrix has full rank:

$$rank(\mathbf{O}) = 18 \tag{4.70}$$

When the matrix is computed, one can see that the reduced observability matrix $\mathbf{O}_{reduced}$ has full rank. The complete observability matrix also has full rank and the plant is observable hence detectable.

$$\mathbf{O}_{reduced} = \begin{bmatrix} \mathbf{C} \\ \mathbf{CA} \end{bmatrix} = \mathbf{I}_{18} \tag{4.71}$$

Other conditions necessary for the design of the controller, especially considering the loop-shaping procedure, will be detailed in the next chapter.

5

Design of the Controller

Since ETS-VII, launched in 1997 [48], important technological as well as theoretical progress were made. Even if the computational power of space OBC is still limited and lower than the computers used on Earth, it is now able to handle more complex models and computations. Thus, one objective of the ongoing project at DLR, RICADOS, is to implement a robust and combined controller of the S/C and the manipulator mounted on it. The model of the complete satellite as one was developed and presented in Section 4. Its control will be discussed in the following sections. First of all, the control strategy and architecture are discussed in Section 5.1. Based on its results and on a literature study, the control technique to be implemented is chosen in Section 5.2. Finally, the design of the controller is conducted in Section 5.4.

5.1. Control Strategy and Architecture

DLR's RICADOS technology development project has many challenging objectives. One of them is to implement a combined controller of the base and the robotic arm. This strategy has not yet been thoroughly investigated. In fact, it is not the easiest control strategy to implement compared to the collaborative strategy and the free-floating one as discussed in 1.1.3. The combined system is a complex system with a large number of inputs and outputs. Thus, the computational power needed to design and run the controller is larger than when a strategy with two separated controllers is used. Moreover, not many control design methods are able to handle the complexity of such a system. On the one hand, in a mission where collaborative control or a free-floating configuration is used, the two controllers implemented can be derived from existing ones. Designing a robust controller for the robotic arm only is a task widely described in literature as well as the robust control of a spacecraft base in orbit. Thus, two such controllers can be used in a mission where one of these two strategies has been chosen. On the other hand, for the design of a combined controller, these controllers cannot be adapted and used, a totally new controller shall be designed which implies some development and testing efforts. Even though combined control represents a real challenge, the advantages it would bring are important. In addition to making the architecture simpler and more elegant, using a combined controller would unite and coordinate the actuators of both subsystems which represents a higher potential in terms of performance. The stability of the system would also be increased since it will directly be integrated in the design of the controller [50].

The long-term objective is to use such a controller in space. As described in Chapter 2, there are multiple constraints and uncertainties for any mission to space. The controller shall still have good performance under these mission constraints. In order to be able to operate at a sufficient level of performance, the controller shall be robust. It will be able to overcome sensor inaccuracies, parameter uncertainties and unmodeled dynamics of the system as well as rejecting possible disturbances.

5.2. Control Technique

The control strategy was discussed in the previous section. The controller shall be robust. To do so, there are multiple robust control theories more or less widely used in literature but they all have the same main objective which is to design controllers to be used in problems where there are uncertainties, limitations and/or disturbances. Each method has advantages and disadvantages. They will be studied and detailed hereafter and the most adapted one for this thesis will be further described.

5.2.1. Trade-Off among Possible Robust Control Techniques

The objective of this section is to get an overview of the techniques so that the most adapted method to the problem is chosen. The methods investigated below are Quantitative Feedback Theory (QFT), Linear Quadratic Gaussian with Loop Transfer Recovery (LQG/LTR), Sliding Mode Control (SMC), Linear Parameter-Varying (LPV) control and H_∞ using loop-shaping or μ-synthesis.

The choice of the most adapted technique for the thesis topic is made based on four main criteria and their relative importance: the ability to design non-linear controllers (low priority), the multiple-input and multiple-output (MIMO) friendliness (top priority), the controller complexity (high priority) and the implementation complexity (top priority). Of course, the level of importance given to each criterion is based on the thesis characteristics and on the system studied. Thus, it can differ from the *optimal* case. For instance, the implementation complexity is of top importance because the time of the thesis is limited. In the optimal case without a time limit, the implementation complexity would be of low importance. Also, the controller shall be a MIMO controller since the system itself has multiple inputs and multiple outputs. Finally, the controller complexity shall be limited since it has to run on-board. Of course, some steps of the controller design may be completed on ground but some others will still have to be completed on-board, with reduced computational power. These main trade-off criteria are listed in Table 5.1.

Table 5.1: Degree of Importance of the Criteria for the Choice of Control Technique

Criterion	**Optimal Case**	**Thesis Project**
Non-Linearity	High	Low
MIMO-Friendly	Top	Top
Controller Complexity	High	High
Implementation Complexity	Low	Top

First of all, QFT, developed by Horowitz in 1982 [30], is a frequency-domain design technique based on the use of plots of loop-gains on Nichols charts. It relies on graphical representation of the loop gain on these charts to obtain a desired robust design for specified plant parameters uncertainties [31]. This technique is mostly used during the design of single-input single-output (SISO) controllers. The interpretation of the Nyquist plots for MIMO systems is challenging [20].

The LQG/LTR robust control technique is the result of successive improvements made on the initial method called Linear Quadratic (LQ) theory [12]. The latter was then completed to be more realistic in the Linear Quadratic Gaussian (LQG) theory before being improved using Loop Transfer Recovery (LTR). The working principle of this technique was first to determine a state feedback law minimizing the so-called quadratic performance index. This optimization problem is called the Linear Quadratic Regulator problem [44]. First based on the assumption that the feedback was directly accessible, this method was improved by considering an output feedback law instead of an output feedback one. This improved technique is called the LQG theory. The output signal is observed using a Kalman filter. Even if the LQ controller and the Kalman filter have good robustness properties, the combination of both is not guaranteed to be robust. To make sure that it is, LTR techniques are used [18]. Although it is pretty convenient for MIMO-systems, designing and implementing a controller using LQG/LTR is a complicated task [12]. Indeed, it involves the resolution of a Riccati equation, the design of a Kalman filter and an iterative process for the LTR.

Moreover, SMC would allow to design a non-linear controller [15]. Indeed, this technique is used with variable structure systems. The main difference between this technique and the others is the fact that the control can change its architecture. As many control structures as needed can be defined as long as the overall organization is continuous. The control design problem is then to define the structures as well as the switching logic between these. Moreover, the sliding surface is the region linking two structures. Once the sliding surface is reached, the control logic is ideally well chosen so that the signal keep track of the sliding surface and stays on it. However, a phenomenon of chattering appears for physical systems because of the finite speed of switching between one structure and another one. A new kind of sliding mode controller has been developed to cope with that problem and is called second order sliding mode controller [8]. Even though this is a non-linear control method, it has a major drawback. In order to obtain a satisfying controller for a complex system, several structures and sliding functions have to be chosen. This process is intuitive and may be complex [59]. Moreover, a phenomenon of chattering appears for physical systems because of the finite speed of switching between one structure and another one. This may have a detrimental effect on the system since the motion of the arm would in that case not be smooth. However, a new kind of sliding mode controller has been developed to cope with that problem and is called second order sliding mode controller [8]. Its implementation is also complex.

LPV control gathers control techniques developed especially to control a specific class of systems called linear parameter-varying systems. These systems are SISO or MIMO non-linear systems which can be linearized and represented with state-space descriptions based on functions of time-varying parameters [53]. Some sensors are used to measure the outputs of the system as well as exogenous variables. The evolution of the exogenous variables is not known but can be measured in real-time and used in the control scheme. This method is a type of gain scheduling method. As any other of these methods, the gain scheduling description and strategy is tedious and time consuming [7].

Last but not least, another robust control method consists in using H_∞ techniques. They are optimization problems according to the H_∞ norm [12]. Two different methods of controller design based on this optimization were studied: loop-shaping [26] and μ-synthesis [19]. Once a controller guaranteeing closed-loop stability and a level of robust stability for every frequency as required, adding one of these two methods enables the control of the performance of the controller [65]. In terms of implementation, these techniques have the advantage over the ones previously discussed that functions to design the controllers were developed in MATLAB for instance where they are gathered in the Robust Control Toolbox (RCT). Both loop-shaping and μ-synthesis have various applications in real-world problems [41] [32] [3] [14] [10]. These techniques also offer a formal stability proof which make them potential candidates to be used in a real space mission. Moreover, they are sometimes combined [40] in order to achieve better robustness. One of the main advantages of μ-synthesis over loop-shaping is that the structure of the uncertainty can be considered in the design of the controller [24]. However, based on the description of constraints and uncertainties considered in this problem and described in Chapter 3, this main advantages will not have consequences on the overall design.

Finally, based on the characteristics of each robust control technique with respect to the trade-off criteria given in Table 5.1, the H_∞ technique with loop-shaping was chosen. Two different loop-shaping techniques are implemented in the RCT. Mixed-sensitivity loop-shaping was considered in this thesis and will thus be detailed hereafter.

5.2.2. Description of the H_∞ Technique using Mixed-Sensitivity Loop-Shaping

Generally speaking, the standard H_∞ problem, described in [55], considers the model of the system between the control signal **u** and the observed output **v**, some disturbances, noises and other exogenous inputs, gathered in the input **w** and some exogenous outputs **z** which represents the control error. It is usually represented as shown Figure 5.1, with **K** the controller and **P** the generalized plant. The latter includes the plant model **G** derived in Section 4.3.3, the interconnection structure between the plant and the controller and weighting functions. The way it is computed depends on the control technique chosen and is further detailed hereafter (Fig. 5.3).

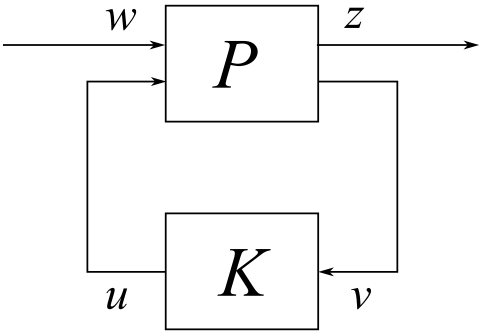


Figure 5.1: Standard H_∞ problem

The mathematical interpretation of Figure 5.1 is as follows:

$$\begin{cases} \begin{bmatrix} \mathbf{z} \\ \mathbf{y} \end{bmatrix} = \mathbf{P} \begin{bmatrix} \mathbf{w} \\ \mathbf{u} \end{bmatrix} = \begin{bmatrix} \mathbf{P}_{11} & \mathbf{P}_{12} \\ \mathbf{P}_{21} & \mathbf{P}_{22} \end{bmatrix} \begin{bmatrix} \mathbf{w} \\ \mathbf{u} \end{bmatrix} \\ \mathbf{u} = \mathbf{K}\mathbf{y} \end{cases} \tag{5.1}$$

Thus, the following equation is obtained using the lower loop linear fractional transformation $\mathbf{F}_l$:

$$\mathbf{z} = [\mathbf{P}_{11} + \mathbf{P}_{12}\mathbf{K}(\mathbf{I} - \mathbf{P}_{22}\mathbf{K})^{-1}\mathbf{P}_{21}]\mathbf{w} = \mathbf{F}_l(\mathbf{P}(s), \mathbf{K}(s))\mathbf{w} \tag{5.2}$$

The objective of the H_∞ problem is to find a stabilizing controller K which minimizes $\left\|\mathbf{F}_l(P(s), K(s))\right\|_\infty$. To solve this minimization problem using Riccati equations, some assumptions are taken concerning the system. The latter has the following state-space representation:

$$\begin{cases} \dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}_1\mathbf{w} + \mathbf{B}_2\mathbf{u} \\ \mathbf{z} = \mathbf{C}_1\mathbf{x} + \mathbf{D}_{11}\mathbf{w} + \mathbf{D}_{12}\mathbf{u} \\ \mathbf{v} = \mathbf{C}_2\mathbf{x} + \mathbf{D}_{21}\mathbf{w} + \mathbf{D}_{22}\mathbf{u} \end{cases} \tag{5.3}$$

Also, it is assumed that:

- $(\mathbf{A}, \mathbf{B}_2)$ is controllable and $(\mathbf{C}_2, \mathbf{A})$ is observable
- $\begin{bmatrix} \mathbf{A} - j\omega\mathbf{I} & \mathbf{B}_2 \\ \mathbf{C}_1 & \mathbf{D}_{12} \end{bmatrix}$ has full column rank for all $\omega \in \mathbb{R}$
- $\begin{bmatrix} \mathbf{A} - j\omega\mathbf{I} & \mathbf{B}_1 \\ \mathbf{C}_2 & \mathbf{D}_{21} \end{bmatrix}$ has full row rank for all $\omega \in \mathbb{R}$
- $\mathbf{D}_{12}$ and $\mathbf{D}_{21}$ have full rank

Then there exists a stabilizing controller for which $\left\|\mathbf{F}_l(\mathbf{P}(s), \mathbf{K}(s))\right\|_\infty \leq \gamma$ if and only if the following conditions hold:

1. $\mathbf{A}\mathbf{Q} + \mathbf{Q}\mathbf{A}^T + \mathbf{Q}(\frac{1}{\gamma^2}\mathbf{C}_1^T\mathbf{C}_1 - \mathbf{C}_2^T\mathbf{C}_2)\mathbf{Q} + \mathbf{B}_1\mathbf{B}_1^T = 0$ has a stabilizing solution $\mathbf{Q} \geq 0$
2. $\mathbf{P}\mathbf{A} + \mathbf{A}^T\mathbf{P} + \mathbf{P}(\frac{1}{\gamma^2}\mathbf{B}_1\mathbf{B}_1^T - \mathbf{B}_2\mathbf{B}_2^T)\mathbf{P} + \mathbf{C}_1^T\mathbf{C}_1 = 0$ has a stabilizing solution $\mathbf{P} \geq 0$
3. All eigenvalues of $\mathbf{Q}\mathbf{P}$ have magnitude less than γ^2

In that case, one solution of the H_∞ problem is the controller $\mathbf{K}$ such that $\mathbf{u} = \mathbf{K}\mathbf{y}$ and $||\mathbf{P}||_\infty < \gamma$ is

$$\begin{cases} \dot{\hat{\mathbf{x}}} = (\mathbf{A} + [\frac{1}{\gamma^2}\mathbf{B}_1\mathbf{B}_1^T - \mathbf{B}_2\mathbf{B}_2^T]\mathbf{P})\hat{\mathbf{x}} + (\mathbf{I} - \frac{1}{\gamma^2}\mathbf{P}\mathbf{Q})^{-1}\mathbf{Q}\mathbf{C}_2^T(\mathbf{y} - \mathbf{C}_2\hat{\mathbf{x}}) \\ \mathbf{u} = -\mathbf{B}_2^T\mathbf{P}\hat{x} \end{cases} \tag{5.4}$$

This solution is called the central controller. It is a sub-optimal solution of the problem. Finding the optimal controller is a complex task consisting in searching for the optimal γ as detailed in [12]. Because there are no specifications of performance requirements for the standard H_∞ problem, the following H_∞-based optimization method called loop-shaping is described.

More precisely, the H_∞ control theory guarantees closed-loop stability and a level of robust stability for every frequency but adding loop-shaping in the design of the controller enables the control of the performance of the controller [65]. The objective of this technique is to design a controller satisfying performance and robustness objectives determined by the control engineer using weighting functions. To do so, the open-loop gain is shaped using weighting functions so that the closed-loop system meets the objectives. To better understand the following, a general control system is shown Fig. 5.2.

Based on Fig. 5.2, the following equation is derived:

$$\begin{aligned} \mathbf{y}(s) &= \left(\mathbf{I} + \mathbf{G}(s)\mathbf{K}(s)\right)^{-1}\left(\mathbf{G}(s)\mathbf{K}(s)\mathbf{r}(s) + \mathbf{d}(s) - \mathbf{G}(s)\mathbf{K}(s)\mathbf{n}(s)\right) \\ &= \left(\mathbf{I} + \mathbf{L}(s)\right)^{-1}\mathbf{L}(s)\mathbf{r}(s) + \left(\mathbf{I} + \mathbf{L}(s)\right)^{-1}\mathbf{d}(s) - \left(\mathbf{I} + \mathbf{L}(s)\right)^{-1}\mathbf{L}(s)\mathbf{n}(s) \end{aligned} \tag{5.5}$$

where $\mathbf{L}(s) = \mathbf{G}(s)\mathbf{K}(s)$ is the open-loop transfer matrix. By defining

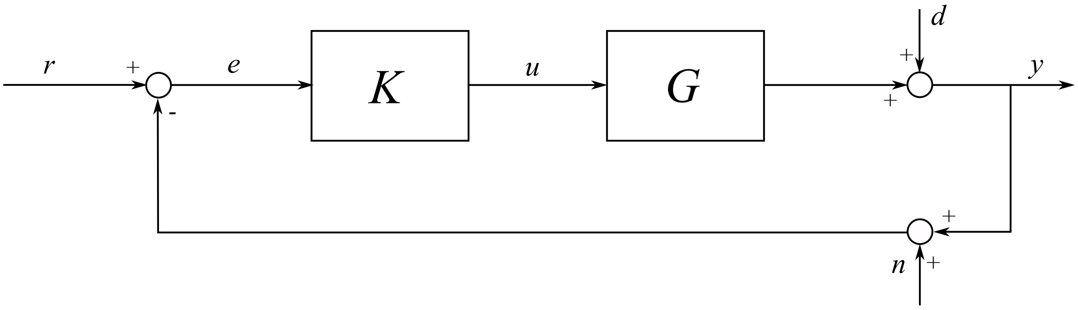


Figure 5.2: General control system

- the output sensitivity function matrix $\mathbf{S}$ as $\mathbf{S}(s) = \left(\mathbf{I} + \mathbf{L}(s)\right)^{-1}$
- the output complementary sensitivity function matrix, or closed-loop transfer function matrix, $\mathbf{T}$ as $\mathbf{T}(s) = \left(\mathbf{I} + \mathbf{L}(s)\right)^{-1}\mathbf{L}(s)$

It finally gives:

$$\mathbf{y}(s) = \mathbf{T}(s)\mathbf{r}(s) + \mathbf{S}(s)\mathbf{d}(s) - \mathbf{T}(s)\mathbf{n}(s) \tag{5.6}$$

It is also worth noting that, by definition:

$$\mathbf{T}(s) + \mathbf{S}(s) = \mathbf{I} \tag{5.7}$$

thus the name of $\mathbf{T}$ as complementary sensitivity.
Based on Eq. 5.6, some properties on the open-loop transfer function matrix $\mathbf{L}$ can be deduced. Firstly, achieving good command following requires the singular values of $\mathbf{T}$, $\boldsymbol{\sigma}(\mathbf{T})$, are close to one which means that $\boldsymbol{\sigma}(\mathbf{L})$ shall be large: $\boldsymbol{\sigma}(\mathbf{GK}) \gg 1$. Secondly, in order to have good disturbance rejection, $\boldsymbol{\sigma}(\mathbf{S}) \approx \mathbf{0}$ which implies that $\boldsymbol{\sigma}(\mathbf{GK}) \gg 1$. Finally, the ability of the controller to attenuate sensor noise depends on magnitude of $\boldsymbol{\sigma}(\mathbf{T})$. To obtain good noise attenuation, it shall be small, meaning that $\boldsymbol{\sigma}(\mathbf{GK}) \ll 1$. One realizes easily that the two first conditions and the last one are conflicting. However, because noise is a high frequency phenomenon and the frequency of the command and the possible disturbances is small, a trade-off can be found between sensitivity and complementary sensitivity for different frequency ranges. In fact, the previous properties on $\boldsymbol{\sigma}(\mathbf{S})$ and $\boldsymbol{\sigma}(\mathbf{T})$ can all be achieved if compromises are made: if $\boldsymbol{\sigma}(\mathbf{S})(s)$ is small at low frequencies (*i.e.* $\boldsymbol{\sigma}(\mathbf{S})(s)$ is large) and $\boldsymbol{\sigma}(\mathbf{T})(s)$ is low at high frequencies (*i.e.* $\boldsymbol{\sigma}(\mathbf{T})(s)$ is large), all conditions can be met. The key element is the choice of crossover frequencies which is proper to each system. To shape the loop according to the objectives in terms of singular values of the sensitivity, weighting filters are used. When using the mixed-sensitivity technique, up to three filters can be specified. They are shown on Fig. 5.3. The plant is said to be augmented.

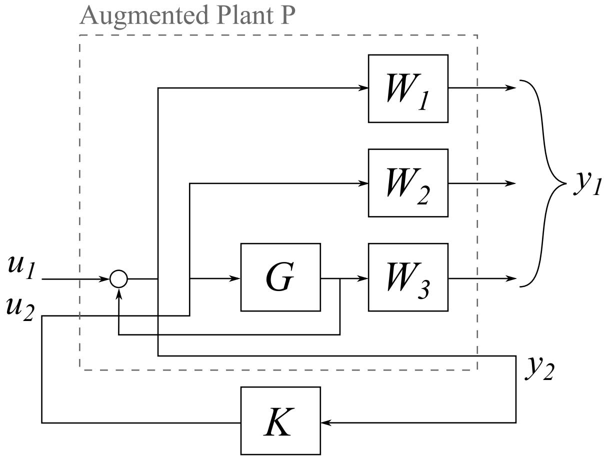


Figure 5.3: Definition of the weighting filters and the augmented plant

Once the plant is augmented, the H_∞ norm optimization can be completed. On MATLAB, the function *hinfsyn* can be used to solve the optimization problem. The returned controller $\mathbf{K}$ is such that, for all frequency ω:

$$\begin{cases} \boldsymbol{\sigma}(\mathbf{S}(j\omega)) & \leq & \gamma\boldsymbol{\sigma}(\mathbf{W}_1^{-1}(j\omega)) \\ \boldsymbol{\sigma}(\mathbf{KS}(j\omega)) & \leq & \gamma\boldsymbol{\sigma}(\mathbf{W}_2^{-1}(j\omega)) \\ \boldsymbol{\sigma}(\mathbf{T}(j\omega)) & \leq & \gamma\boldsymbol{\sigma}(\mathbf{W}_3^{-1}(j\omega)) \end{cases} \tag{5.8}$$

If γ is lower than one, then Eq. 5.8 implies that:

$$\begin{cases} \boldsymbol{\sigma}(\mathbf{S}(j\omega)) & \leq & \boldsymbol{\sigma}(\mathbf{W}_1^{-1}(j\omega)) \\ \boldsymbol{\sigma}(\mathbf{KS}(j\omega)) & \leq & \boldsymbol{\sigma}(\mathbf{W}_2^{-1}(j\omega)) \\ \boldsymbol{\sigma}(\mathbf{T}(j\omega)) & \leq & \boldsymbol{\sigma}(\mathbf{W}_3^{-1}(j\omega)) \end{cases} \tag{5.9}$$

These conditions, if met, justify that the requirements imposed on the $\mathbf{S}$, $\mathbf{KS}$ and $\mathbf{T}$ functions are respected by choosing weighting filters accordingly. Once a controller is found, one should check that these requirements are met. This can be conducted based on Eq. 5.9 by comparing the singular values of $\mathbf{S}$ with the ones from $1/\mathbf{W}_1$ as well as $\mathbf{T}$ with $1/\mathbf{W}_3$ and $\mathbf{KS}$ with $1/\mathbf{W}_2$. If for each of these graphs, the $\mathbf{S}$, $\mathbf{T}$ and $\mathbf{KS}$ curves are above the singular values curve of the inverse of the weighting filters, then the performance and robustness requirements are met.

5.3. Simplifications of the Mission Path-Planning

The path or motion planning of the end-effector is necessary since there are an infinite number of possible paths to take from any initial set of states of the system to any final set. Since this topic is not central in the thesis, in the sense that the main objective is not to find the fastest trajectory to reach the final configuration, the path-planning task will be kept to a minimum. For example, the initial configuration of the chaser with respect to the target is such that the handle (*i.e.* the target position of the EE) is within the workspace of the robotic arm which is shown Fig. 4.1.

There is no criterion chosen for the optimization of the change of states of the system to reach the final position. Thus, the easiest case to consider is that the final position and attitude of the spacecraft in the target BF reference frame are kept stationary (or bridge small differences) between the initial conditions, when the combined control mode is activated, and the desired final configuration, at the end of the deployment process. The objective is then for the chaser to remain (almost) immobile during the deployment of the arm. The base actuators will then be used to compensate the reaction on the base from the moving robotic arm. Even though such a compensation was also used during ETS-VII mission [48], there is a fundamental difference between the two control strategies. Indeed, ETS-VII used two different controllers for the arm and the base while they are combined in one controller in this thesis. There is no exchange of states between two separated subsystems anymore, which may make the counteraction of the reaction on the base faster and more efficient. The better the compensation of reaction efforts on the base by its actuators, the closer the operation of the space manipulator to the operation of a robotic arm on ground. If the compensation would be perfect, the operation of the arm on the satellite would be similar to an arm fixed to an inertial space being operated on Earth.

Moreover, in a further stage of the project, robotic singularities and obstacle avoidance will be investigated. Indeed, any collision has to be avoided during this critical phase. This topic is challenging too. As it is out of the scope of the current project, it will not be studied here. It is thus assumed that there is no obstacle whatsoever. The robotic arm is free to move, without restrictions within its workspace.

Finally, a real satellite may be tumbling around one or more axes. However, it is assumed in this thesis that the target's attitude does not change with respect to the inertial LVLH centered at its center of mass. If the tumbling would be considered, the chaser would need to first synchronize its motion with respect to the target before deploying its manipulator such that the efforts at the end-effector when grasping the handle are not too large. Synchronizing with the target also allows to reduce the number of obstacles to avoid. Indeed, if it is not done, the possible appendages of the target would represent an important threat as the arm is being deployed.

Table 5.2: List of requirements

Requirement	Notation	Value	Unit
Steady-state tracking error	amp	1	dB
Maximum gain	M	20	dB
Good command following until	ω_B	10	Hz
Noise amplification			
↳ *at low frequency*	glf	0.1	dB
↳ *at high frequency*	ghf	-20	dB
Noise frequency from	ω_C	1	kHz
Energy consumption gain	a_u	-10	dB

5.4. Design of the Controller

5.4.1. Requirements and Constraints

The task of the control engineer is to be able to formulate requirements which are not given initially, in terms of robustness or performance. As explained in Chapter 3, due to the duration of this thesis, only some mission constraints were modelled and the objective of the controller is to be able to overcome these constraints. Moreover, it shall achieve good performance in the sense that it shall be precise and fast enough such that the assumptions made previously are not made invalid because of a too long deployment phase. To make the control of any state easier, the controlled system shall be decoupled such that when a force or a torque is applied to the system, only the corresponding state is impacted. Also, there shall be no significant overshoot of the states with respect to the command because any overshoot may lead to the collision of the gripper with the target. The numerical values of the various requirements which were considered in this thesis are given in Tab. 5.2.

Furthermore, another constraint of the system is that the arm and the base are usually operated at very different frequencies. The order of magnitude of the operating frequency of the space manipulator is about 1kHz while being only a few Hz for the S/C AOCS. Even though the operating frequency of the servicer can go be up to 250Hz, this AOCS was chosen since it is of the order of magnitude of what is usually considered nowadays. To control simultaneously subsystems with very different operating frequencies, the control can be done at the lowest frequency. Although other techniques may be available, this strategy was implemented because the system is not made even more complex that way. In a further stage of the RICADOS project, another technique may be considered. The discretization of the controller and its effect will be discussed later on in Section 6.1.2.

Finally, the actuators are limited in terms of available force and torque produced. Of course, the controller shall respect these limitations and the control inputs to the plant shall be within the available range of forces and torques.

5.4.2. Scaling and Ordering of the Inputs and Outputs

The linear model of the system was derived in Chapter 4. Using the plant model $\mathbf{G}$ found in the latter and the block diagram shown on Fig. 5.2, the following is written:

$$\mathbf{y}_u = \mathbf{G}_u \mathbf{u}_u + \mathbf{d}_u \tag{5.10}$$

where

- $\mathbf{G}_u$ is the plant model
- $\mathbf{u}_u$ is the control inputs vector
- $\mathbf{y}_u$ is the measured outputs vector
- $\mathbf{d}_u$ is the disturbance vector

and

$$\mathbf{e}_u = \mathbf{y}_u - \mathbf{r}_u \tag{5.11}$$

where

- $\mathbf{r}_u$ is the reference vector given as input
- $\mathbf{e}_u$ is the control error

In the two previous equations, the subscript 'u' is used to expressed the unscaled variables. Indeed, scaling is an important task to be completed before the design of the controller. Its aim is to obtain magnitude of cross coupling terms of the same order. It improves the conditioning of the problem and makes the optimization more efficient. The method to do so is exposed hereafter and highly depends on the judgment of the control engineer. Indeed, choices are first made on the expected magnitudes of the different inputs and outputs. Then the unscaled variables are scaled based on the decisions made in the previous step.

On the theoretical part, the unscaled maximum magnitudes expected for the disturbance, the input change and the control error are respectively noted $\mathbf{d}_{u_{max}}$, $\mathbf{u}_{u_{max}}$ and $\mathbf{e}_{u_{max}}$. Of course the size of these vectors is equal to the size of the equivalent unscaled variables. Based on these, scaled variables are derived:

$$\begin{cases} \mathbf{d}_s &= \mathbf{d}_u / \mathbf{d}_{u_{max}} \\ \mathbf{e}_s &= \mathbf{e}_u / \mathbf{e}_{u_{max}} \\ \mathbf{r}_s &= \mathbf{r}_u / \mathbf{e}_{u_{max}} \\ \mathbf{u}_s &= \mathbf{u}_u / \mathbf{u}_{u_{max}} \\ \mathbf{y}_s &= \mathbf{y}_u / \mathbf{e}_{u_{max}} \end{cases} \tag{5.12}$$

$\mathbf{e}_s$, $\mathbf{u}_s$ and $\mathbf{y}_s$ are scaled with respect to the same scaling vector $\mathbf{e}_{u_{max}}$ because they are expressed in the same units. The vectors of maximum magnitudes of the variables can be expressed as diagonal matrices D_d, D_e and D_u without any diagonal term equal to zero. Using these matrices, the scaled variables are obtained from the unscaled ones as:

$$\begin{cases} \mathbf{d}_s &= \mathbf{D}_d^{-1} \mathbf{d}_u \\ \mathbf{e}_s &= \mathbf{D}_e^{-1} \mathbf{e}_u \\ \mathbf{r}_s &= \mathbf{D}_e^{-1} \mathbf{r}_u \\ \mathbf{u}_s &= \mathbf{D}_u^{-1} \mathbf{u}_u \\ \mathbf{y}_s &= \mathbf{D}_e^{-1} \mathbf{y}_u \end{cases} \tag{5.13}$$

and Eq. 5.10 and Eq. 5.11 becomes:

$$\begin{cases} \mathbf{D}_e \mathbf{y}_s &= \mathbf{G}_u \mathbf{D}_u \mathbf{u}_s + \mathbf{D}_d \mathbf{d}_s \\ \mathbf{D}_d \mathbf{e}_s &= \mathbf{D}_e \mathbf{y}_s - \mathbf{D}_e \mathbf{r}_s \end{cases} \tag{5.14}$$

The expression of the scaled plant is deduced from Eq. 5.14:

$$\mathbf{G}_s = \mathbf{D}_e^{-1} \mathbf{G}_u \mathbf{D}_u \tag{5.15}$$

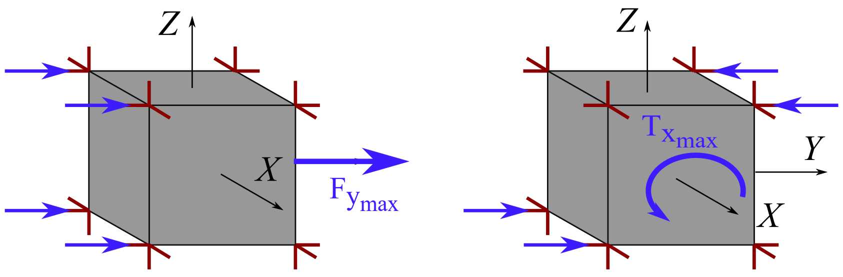
And finally, Eq. 5.10 can be adapted for scaled variables:

$$\mathbf{y}_s = \mathbf{G}_s \mathbf{u}_s + \mathbf{d}_s \tag{5.16}$$

Based on the constraints specific to RICADOS project and the understanding of the system so far, the scaling matrices can be derived based on the maximum disturbance expected, on the maximum control inputs available and on the maximum control error, as detailed below.

- First of all, the precision of the measurements for the position and attitude of the base as well as for joint angles of the arm are respectively of 4mm, 1° and 0.1°. The precision along all three axes for the position and attitude are the same as well as for the three joint angles. These numerical values are used to determine the components of the noise vector. It is also considered that the amplitude of the disturbance vector's components are equivalent to the numerical values taken for the noise vector.

- The determination of the diagonal terms in $\mathbf{D}_u$ is based on the assessment of the maximum control inputs (*i.e.* maximum forces and torques available). In the system considered, there are three thrusters positioned at each corner of the cubic base. There are in total 24 thrusters. The allocation topic is not within the scope of this thesis so only the resulting force and torque on and around the CoM of the base are considered. However, these values can be approximated. For each direction of thrust, there are four thrusters providing 10N each. The maximum thrust along an axis is then close to 40N. Concerning the maximum torque around the base CoM, if it is assumed that the distance from the CoM to the thruster is of one meter, each thruster can produce a torque of 10N.m around one axis. Four thrusters can be fired at the same time to achieve maximum torque. The maximum torque is about 40N.m around each axis. The configurations giving the highest force and torque are shown on Fig. 5.4. These computations are very approximate but what is important during the scaling is not the exact values but the order of magnitude of the maximum amplitudes. Also, it is clear that since the maximum force and the maximum torque need different thrusters firing to be achieved, both cannot be achieved simultaneously. However, this characteristics is not considered here in order to keep the problem as simple as possible. The joints of the arm are equipped with different motors. The values of the maximum torques are given in [34]. For the joints considered here, the maximum torques are 200N.m for the first and second joints and 100N.m for the third one.
- Finally, based on the assumption given earlier that the deployment is controlled using a smooth trajectory, the maximum magnitude for the control error is dependent on the behavior of the controller itself but, once again, a first guess can be given based on the expected characteristics of the controlled closed-loop. At each step, the actual states will be delayed with respect to the command with Δt. The control error e is then the difference in amplitude, at a given time, between the expected states and the current ones. The reasoning is shown Fig. 5.5. The numerical values for the maximum control error are thus based on the judgment of the control engineer since the controller has not yet been developed so the comparison detailed previously is not achievable in reality. However, it is useful to select realistic values of the correct order of magnitude. The maximum control error for the translational states of the base, the rotational ones and the joint angles of the arm are respectively taken equal to 1cm, 0.02° and 0.02°.



(a) (b)

Figure 5.4: Configurations giving maximum force (a) and maximum torque (b) along and around one axis

Ordering the inputs and the outputs to make the plant as diagonal as possible is highly recommended to ease the design of the loop-shaping [55]. However, the RGA derived in Section 4.3.3 showed that the maximum influence of each input on all the outputs are already on the diagonal. The ordering of the inputs and outputs is not necessary in this case.

5.4.3. Loop-Shaping

Using the requirements chosen previously and shown in Table 5.2, the loop-shaping procedure described in Section 5.2 can be conducted. Based on the guidelines given in [11], the three filters (as shown Fig. 5.3 can be determined. Of course it is an iterative process in order to find a controller that meets the requirements. On the one hand, one can say that the first weighting filter $\mathbf{W}_1$ should be large inside the bandwidth in which the control should be good such as it achieves good performance. On the other hand, the third weighting filter $\mathbf{W}_3$ should be large outside the control bandwidth so that the noise is well attenuated ensuring a good robustness. As explained before, these two characteristics

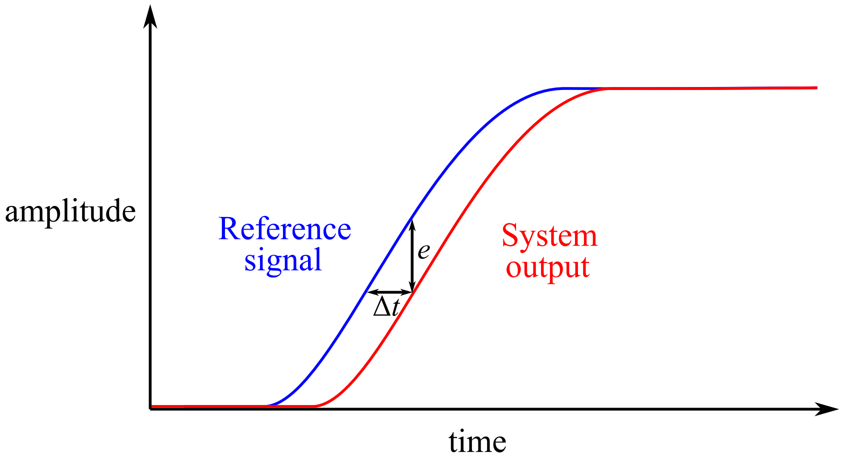


Figure 5.5: Determination of the maximum control error vector

have to be traded-off based on Eq. 5.7. In this case, $\mathbf{W}_1$ is taken as a low-pass weight with a bandwidth of about 10Hz and $\mathbf{W}_3$ is a high-pass one, with a cut-off frequency of about 1kHz which is the lowest frequency at which noise appears as considered in Table 5.2. In this table, other gains are stated and help determine the full expression of these two filters given in Eq. 5.17. All the gains present in the expression of the filters below are first converted from dB to magnitude.

$$W_{1_i} = \frac{\frac{s}{M}+\omega_B}{s+\omega_B*amp} \quad \text{and} \quad W_{3_i} = \frac{\frac{s}{ghf}+\omega_C}{s+\omega_C*glf} \tag{5.17}$$

Based on the requirements, these two filters will be the same for all three types of dynamics behaviors (translation of the base, rotation of the base and rotation of the arm joints). However, during the optimization process, the filters will change and the final ones are bot constrained to be the same for these three types. Moreover, the overall weighting is composed of three types of dynamics, each of which is composed itself of three different filters (identical within one kind of dynamics). At the end, the blocks $\mathbf{W}_1$ and $\mathbf{W}_3$ on Fig. 5.3 are diagonal matrices with the filters as diagonal terms:

$$\mathbf{W}_1 = \begin{bmatrix} \mathbf{W}_{1_t} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{W}_{1_r} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{W}_{1_a} \end{bmatrix} \tag{5.18}$$

where

- $\mathbf{W}_{1_t}$ is a 3×3 matrix with three identical filters on the diagonal to be applied on the translational dynamics of the base
- $\mathbf{W}_{1_r}$ is a 3×3 matrix with three identical filters on the diagonal to be applied on the rotational dynamics of the base
- $\mathbf{W}_{1_a}$ is a matrix with three identical filters on the diagonal to be applied on the dynamics of the joints of the arm. The size of this matrix depends on the number of joints. Since three joints are considered here, it is a 3×3 matrix too but if seven axes of rotation happened to be considered, it would be a 7×7 matrix.

The exact same configuration is present in $\mathbf{W}_3$. Concerning the last weighting function to be initialized, $\mathbf{W}_2$, it is taken as a static gain for the three different kinds of dynamic behavior. The objective of $\mathbf{W}_2$ is to limit the energy necessary to achieve the command. After a few tries, it seems that taking the requirement a_u (after being converted from dB to magnitude) as the static gain gives, after computation of the controller, a maximum control input on each of the scaled inputs of less than one which means that the actuators are not saturated during the motion of the arm.

Finally, the numerical expressions of the initial guess of the filters are given below, as well as their Bode diagrams 5.6. It shows that indeed, the first weight is a low-pass filter as expected and that the third one is a high-pass filter.

$$W_{1_i} = \frac{0.1s+62.83}{s+70.5} \quad \text{and} \quad W_{2_i} = 0.3162 \quad \text{and} \quad W_{3_i} = \frac{10s+6283}{s+6356} \tag{5.19}$$

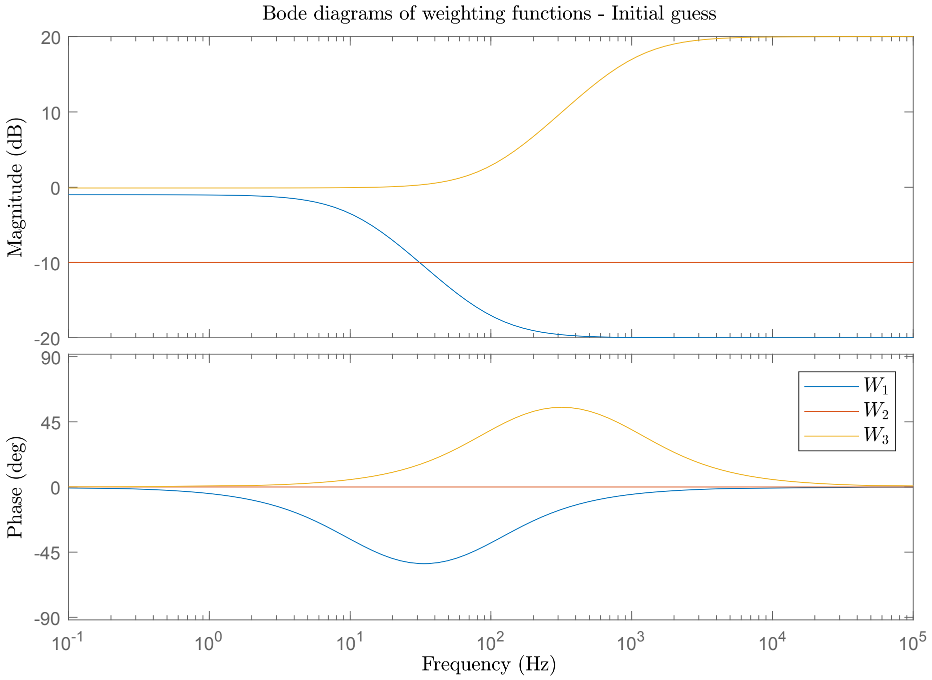


Figure 5.6: Bode diagrams of the initial guess for all three weighting functions

5.4.4. H_∞ Control Design

Once the plant model is developed, then linearized and the initial weighting functions are chosen, the plant is being augmented as required in the mixed-sensitivity H_∞ loop-shaping design. The function *augw* is used to complete this task on MATLAB. The next step is the H_∞ optimization problem. To be able to find a controller, the conditions given in Section 5.2 on the matrices of the system of equations 5.3 have to be met. However, the model of the system is not compatible with completing the third condition which is:

$$\begin{bmatrix} \mathbf{A} - j\omega\mathbf{I} & \mathbf{B}_1 \\ \mathbf{C}_2 & \mathbf{D}_{21} \end{bmatrix} \text{has full row rank for all } \omega \in \mathbb{R}$$

Investigation has shown that this condition cannot be completed using the plant **G** as it is, no matter the filters chosen to augment the plant. This analysis is detailed and summed up in Section B.1 of Appendix B. According to MATLAB documentation [6], if this condition is not met for a certain frequency, the controller might have very lightly damped poles near that frequency. To bypass that condition and to obtain a controller, an identity matrix multiplied by a very low value (e.g. 10^{-9}) is added in the bottom-left block of the **A** matrix of the plant **G**. One should avoid to add an identity matrix with too low diagonal terms. Otherwise, MATLAB will fail to run the *hinfsyn* algorithm.

The initial plant and the one with the identity matrix added can be compared both in frequency and time domains. To have a better understanding, a complex configuration is considered. The velocities and the positions of the linearization configuration are taken different from zero to explore the general behavior of the two plants. The more complex it is, the higher the probability to point out some differences by exciting all modes. The comparison in frequency domain is shown Fig. 5.7. The other Bode diagrams are shown in Section B.2 of Appendix B. The two plants are very similar for most of the frequency domain. The difference appears clearly for low frequencies. However, these are so low that nothing in the plant is really influenced by the behaviour at such frequencies. The modification of the plant is then acceptable in that case and will from now on be used for the design of the controller.

An optimization on the filters is also completed. This step is important because the initial filters given may not directly give good results. Indeed, one chooses the specifications of the control loop which shall be achieved but it happens that for very small variations on the filters, the performance of the closed loop is highly improved. Thus, running an optimization on the filters may give slightly different filters but a controller with very high performances compared to the initial guess. Of course, the co-

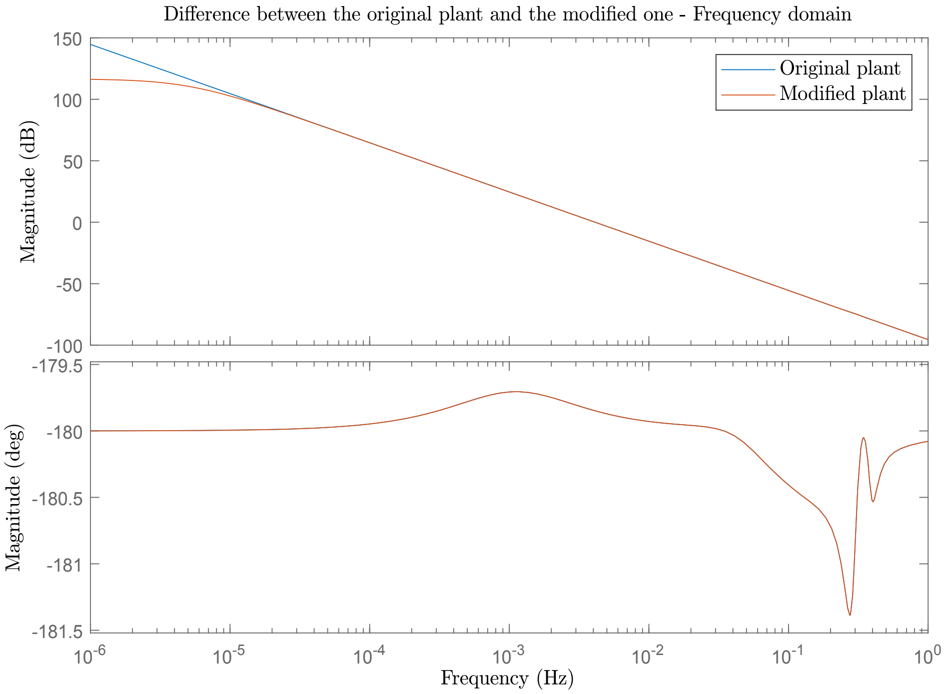


Figure 5.7: Difference in the frequency-domain between the initial plant and the modified one

Table 5.3: List of the final weighting functions

Requirement	W_1	W_2	W_3
Translation	$\frac{0.44s+62.61}{s+70.53}$	0.2818	$\frac{9.94s+6283}{s+6356}$
Rotation	$\frac{-0.43s+62.79}{s+70.5}$	0.3278	$\frac{10.03s+6283}{s+6356}$
Arm	$\frac{-0.78s+62.88}{s+70.49}$	0.2818	$\frac{11s+6283}{s+6356}$

herence of these filters with the requirements given initially should be checked in order to achieve the expected performance and robustness. For each of these optimization steps, *hinfsyn* is used to compute a controller which stabilizes and is a solution to the $\mathbf{H}_\infty$ problem. For each of these controllers, the step response of the closed loop composed of the controller followed by the plant with a unit feedback (as shown Fig. 5.2) is studied. The objective is to minimize the maximum settling time of the 81 I/O couplings. Moreover, a condition on γ is taken: it shall be less than one. In that case, the requirements given as inputs are met.

The controller found using this optimization procedure gives acceptable performance and robustness to the closed-loop. Moreover, the filters used to augment the plant are slighlty different from the ones given as inputs. They are also a bit different depending on the category of dynamics of interest. In fact, as discussed before, the optimal weighting function for the sensitivity, W_1, may not be the same for the arm and for the translational dynamics of the base. The final weighting functions are given in Table 5.3. Most of the coefficients of these weights are very close to those given as initial guess. However, the coefficient, called $1/M$ in Eq. 5.17, multiplied by s in the numerator of each W_1 function is not so close to the initial one of 0.1. The Bode diagram in magnitude of all three W_1 weights can be compared to the initial one. The superimposition of all four diagrams is shown Fig. 5.8. The magnitude for high frequency is lower for the initial guess than for the final filters. The consequences of this difference will be investigated later on in this section.

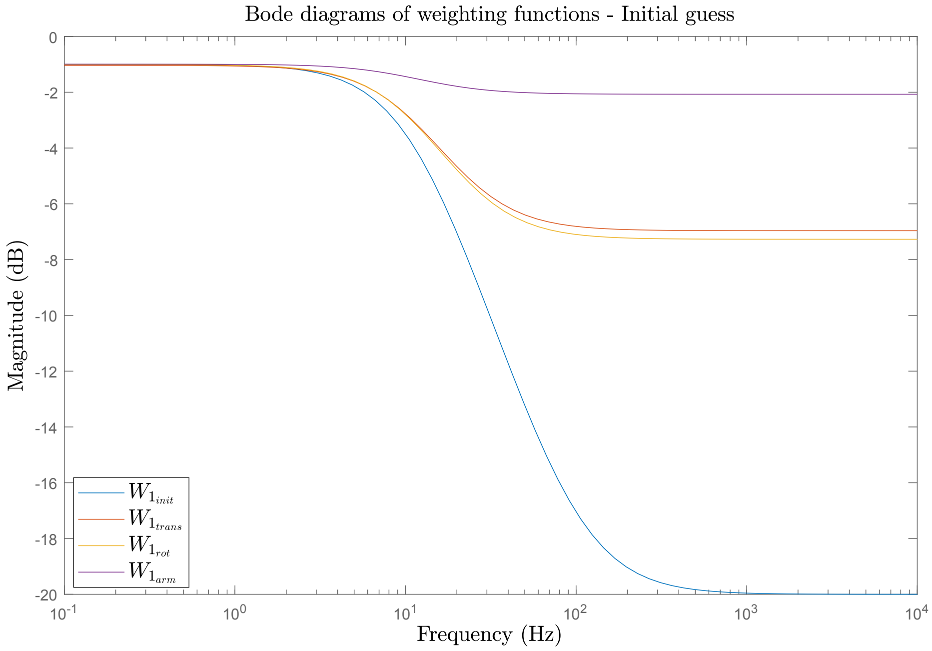


Figure 5.8: Comparison of the final weighting functions W_1 compared to the initial guess

The controller found at the end of the procedure is detailed hereafter. First of all, it stabilizes the plant by design according to the description of the *hinfsyn* function. This first statement is verified by looking at the zeros and poles of the closed-loop as described Fig. 5.2. The pole-zero map is shown Fig. 5.9. All the zeros and poles are in the left half-plane: the closed-loop is stable.

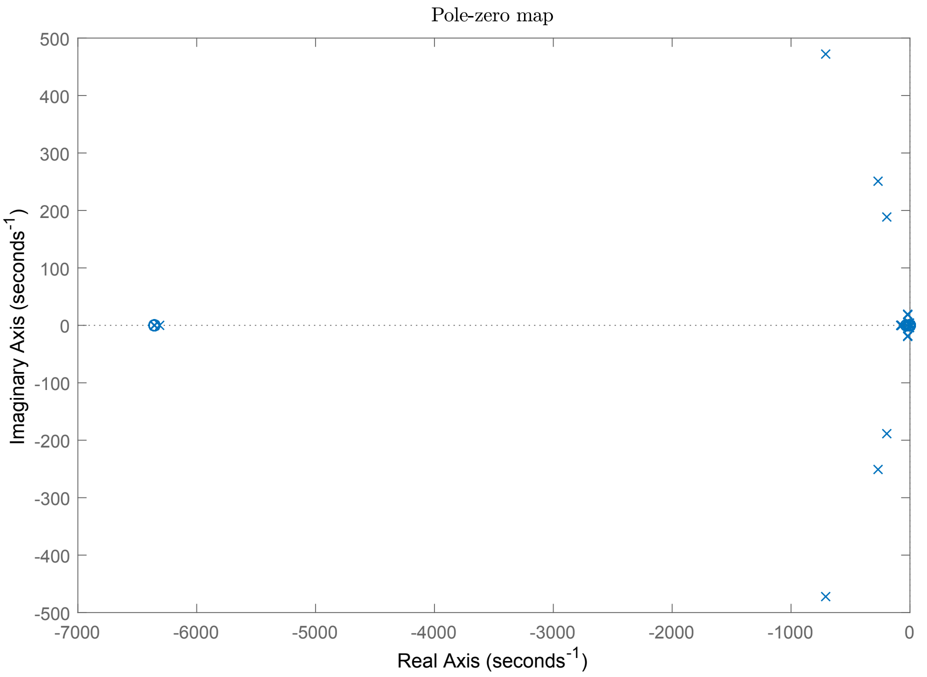


Figure 5.9: Pole-zero map for the closed-loop

Furthermore, it gives a γ value equals to 0.988 which is lower than one. Thus, the requirements imposed using the weighting filters are respected. The objective of the controller is to control the scaled plant so it is implemented in a control loop such as the one shown Fig. 5.2. The reference vector was scaled as explained in Section 5.4.2. So when the controlled loop is tested, the reference vector components cannot exceed one. The step response of the control system is tested with the maximum reference vector possible: $\mathbf{u}_s = \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{bmatrix}^T$. The results are shown Fig. 5.10. With a maximum settling time of about 20 seconds ($\max_{i,j}(\tau_{90\%}) = 18.05$s where i and j are the indices of the inputs and outputs) and most of the settling times lower than ten seconds, the control is realtively fast. It is also precise since the steady-state error is below 1dB as required (Table 5.2).

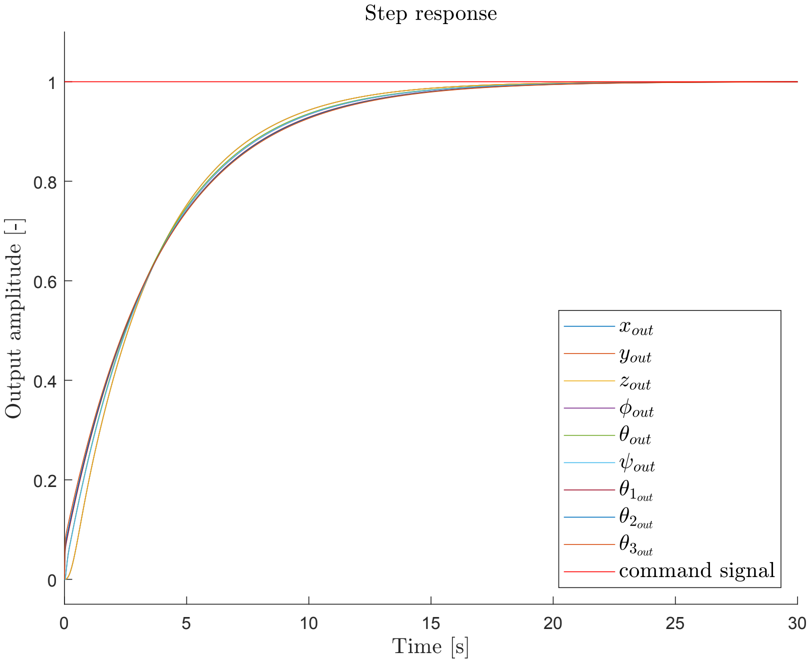


Figure 5.10: Step response of the controlled linear system

The behavior of the extra-diagonal terms are also important since they represent the way a variable is impacted by the motion of an actuator of which it is not the primary effect. For instance, the objective is to keep the base stable and fixed during the manipulation of the arm. The effects of each actuator should have an impact on one output only: the states should be decoupled. All 81 I/O couplings responses to a step input are presented in Appendix C for the sake of clarity. It shows that most of the extra-diagonal time responses are not significantly influenced and that the motion is almost perfectly decoupled. The maximum coupling between an input and an output other than the input state itself is shown Fig. 5.11. When an input on the second joint is commanded, the base rotates a bit as θ_0 is slightly impacted but these unwanted movements are quickly counteracted. All the steady state values are zero or one, as expected. The behavior of the controlled system is satisfying in terms of step response.

The noise rejection capabilities of the controller can also be investigated. The noise chosen here is just a first test, the real noise vector is applied on the unscaled loop. The noise attenuation will be measured using a white noise source with zero mean value. A large noise vector is chosen. A standard deviation is taken on each output equal to 0.1. The results are shown Fig. 5.12. The noise is well attenuated. The requirement defined earlier, in Table 5.2, states that the amplitude of the noise shall be attenuated by at least 20dB. If a closer look is given to the first graph of Fig. 5.12, the maximum amplitude of the signal with noise x_{WN} is 0.888. There is noise remaining in the output signal. Its maximum amplitude was measured as 0.0068. The ratio of these two values is about 130. Converted in dB, it gives an attenuation of 40dB which is greater than the required 20dB.

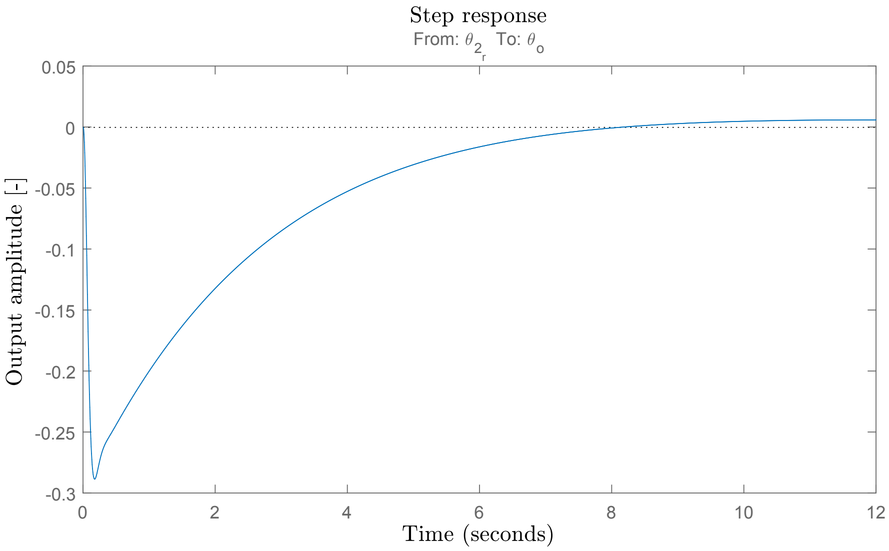


Figure 5.11: Influence on θ_{2_0} of a step command on θ

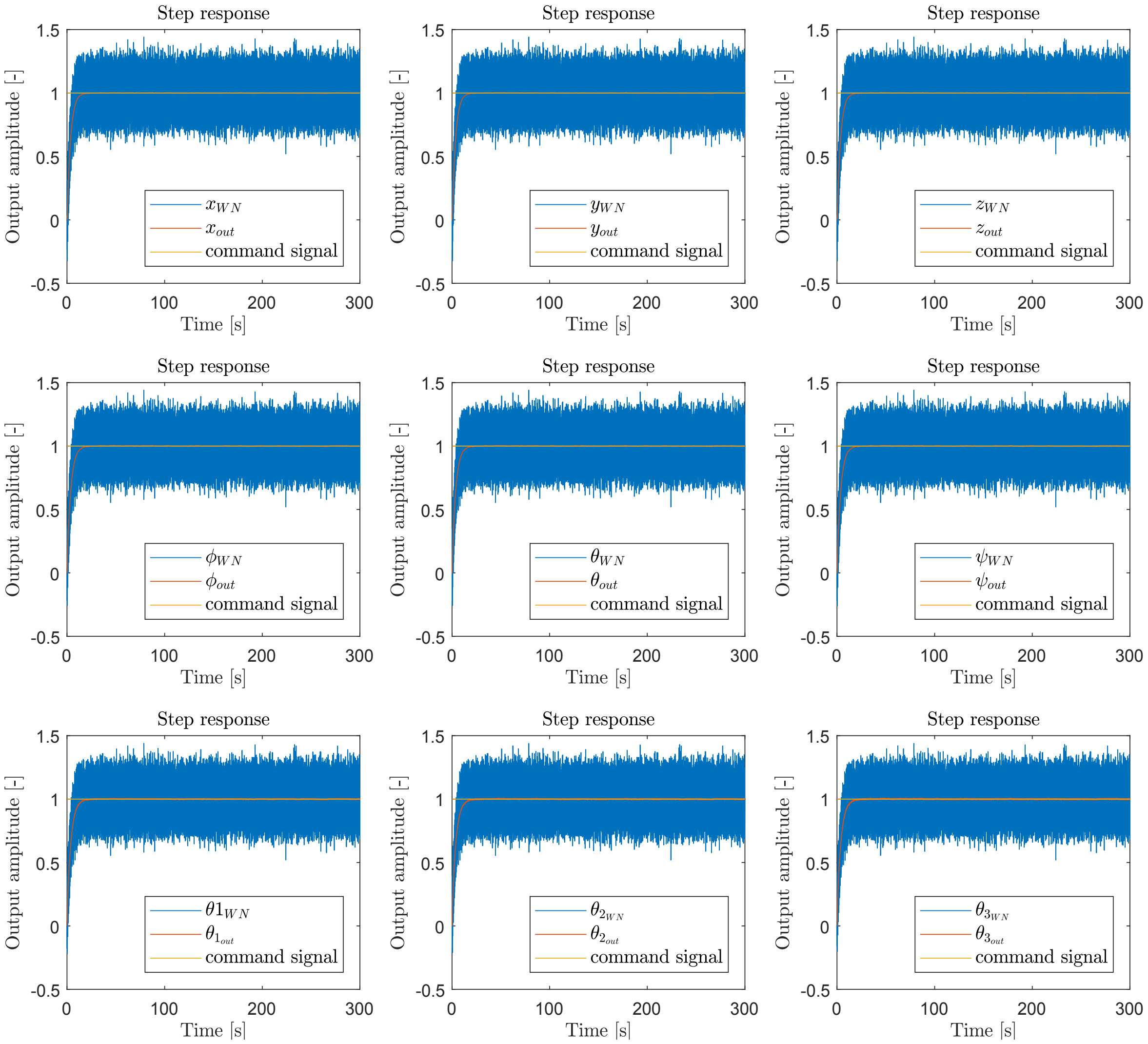


Figure 5.12: Noise attenuation for each output signal

As described in Section 5.4.4, studying the sensitivity and complementary sensitivity functions is important and shows whether or not the controller enables the controlled system to meet the requirements. Based on the three conditions given in Eq. 5.9, the graphs of the singular values of the different functions can be studied and compared with the inverse of the different weighting functions accordingly.
First of all, one should remember that the first weight $\mathbf{W}_1$ is used to shape the sensitivity function $\mathbf{S}$ such that:

$$\|\mathbf{W}_1\mathbf{S}\|_\infty < 1$$

thus, the sensitivity is compared with the inverse of the first weighting function. The evolution of their singular values is shown Fig. 5.13. On this figure, the final weighting functions are also compared to the initial one. As explained previously, the same weighting filter is given at the beginning to the three types of dynamics involved in this problem. At the end, these filters evolved in different manners. This explains the three different red and dashed curves on the figure.

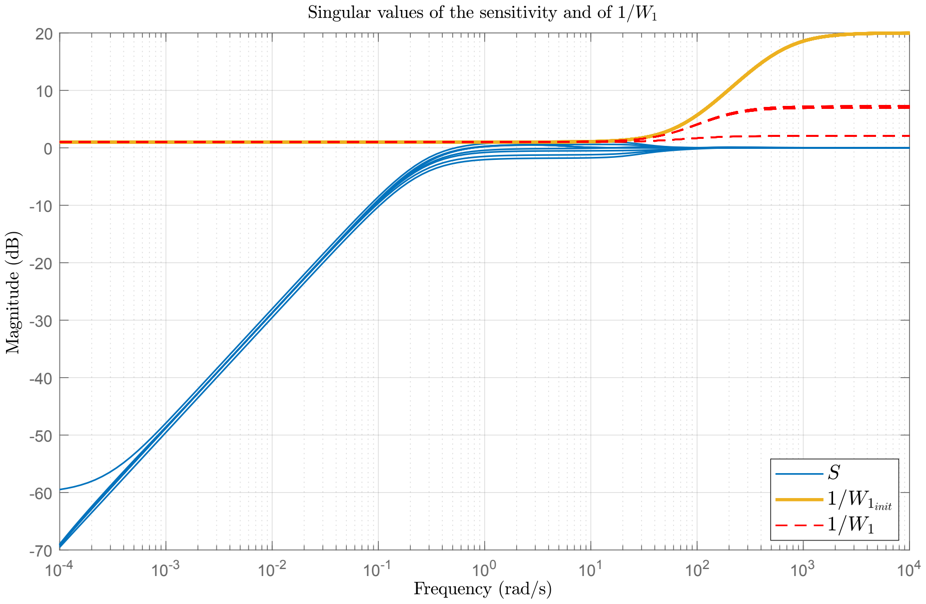


Figure 5.13: Comparison of the singular values evolution of the sensitivity and of the inverse of the initial and final weighting functions $\mathbf{W}_1$

In the frequency range greater than ω_B, the sensitivity function will be constrained to stay lower than initially. It means that the complementary sensitivity will not be able to be as low as allowed initially. However, this value may not even have been reachable anyway since the noise attenuation is highly influenced by the third weighting function. To conclude, having the high frequency gain of $\mathbf{W}_1$ different than the initial one without having a very different third weighting function means that the requirements on $\mathbf{W}_1$ were not necessary for high frequencies because they were less strong than the requirements on $\mathbf{W}_3$. This behaviour is shown Fig. 5.13. One can see that from the initial guess to the final filters, the constraint on the sensitivity function has been made stronger in that range of frequencies. Concerning low frequencies, the initial guess is very similar to the final filters. However, if a very close look is given to the range of frequencies close to ω_B, one can see Fig. 5.14 that one sensitivity curves goes above the initial guess even though it stays below the singular value curve of the inverse of the final weight. It means that a requirement initially given in Table 5.2 is not fulfilled. The γ value obtained and given before describes how easily the requirements fixed by the *final* weighting functions are met but they have changed slightly compared to the initial ones. In this thesis, there were no precise requirements to fulfill. They were taken based on realistic approximated values. Indeed, one can get a general idea of how the system should behave: it should have a small steady-state error, follow the command until at least the frequency of the actuators and noise amplification should be as low as possible for high frequencies. The sensitivity function singular values curve goes over the initial weighting filter by 0.03dB maximum. This difference is very little and since the requirements are not fully rigid, the new weighting function is considered as acceptable even though the performances of the controller may be slightly diminished.

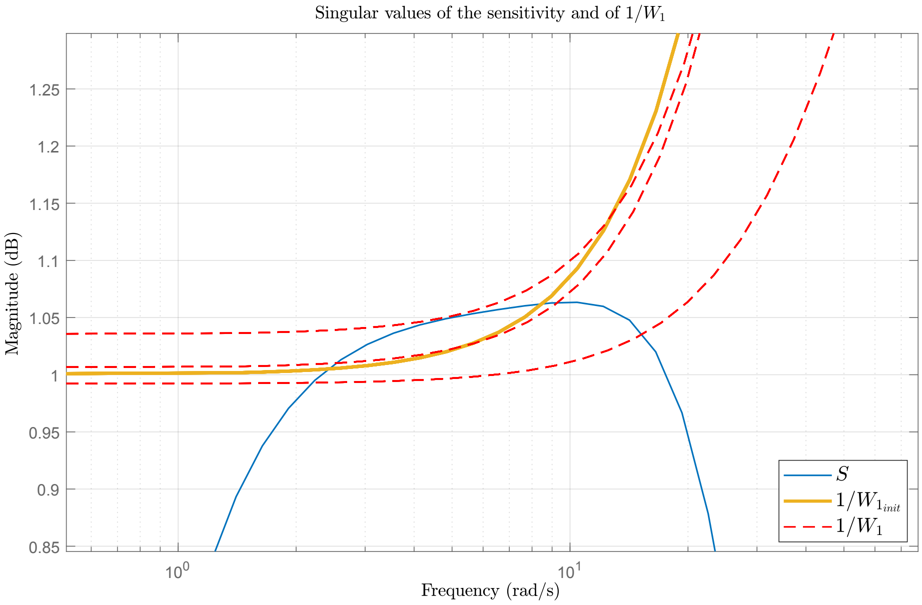


Figure 5.14: Comparison of the singular values evolution of the sensitivity and of the inverse of the initial and final weighting functions

Then, the respect of the constraints imposed by the second weighting function $\mathbf{W}_2$ on the control effort **KS** is checked Fig. 5.15. For every frequency, the singular values of **KS** are lower than the singular values of the second weight: the energy needed to control the satellite is lower than the maximum energy available on-board using the thrusters.

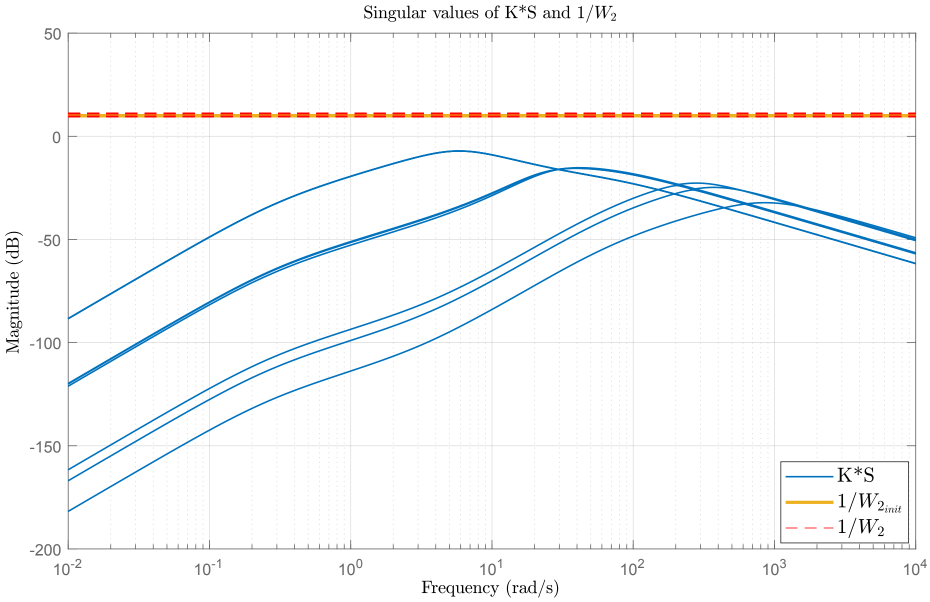


Figure 5.15: Comparison of the singular values evolution of **KS** and of the inverse of the initial and final weighting functions $\mathbf{W}_2$

More precisely, when the step response of the system is simulated with maximum amplitude of the inputs (*i.e.* the command signal is a 9 × 1 vector of ones, as shown Fig. 5.10), the command signal computed by the controller and given to the plant as inputs is depicted Fig. 5.16. It shows that the second weighting filter sets up a boundary on the maximum control energy which can be used for a maneuver. Using the $\mathbf{W}_2$ weights as given in Table 5.3 gives a controller for which the actuators are never used at a hundred percent of their capacities. Some forces and torques computed are about 30% of their maximums while some others are way lower. On the one hand, it is good not to have maximum power required on any control inputs of the plant. Indeed, the computation of the maximum torques and forces available completed previously in Section 5.4.2 is approximated and highly idealized. On the other hand, some of the values remain very small (about a few percents only). It means that further optimization is still possible on the second weighting function. Taking relatively different initial values for this weight may be necessary to get better results since the maximum forces and torques of the base are more than three times lower than the available torques at the arm joints.

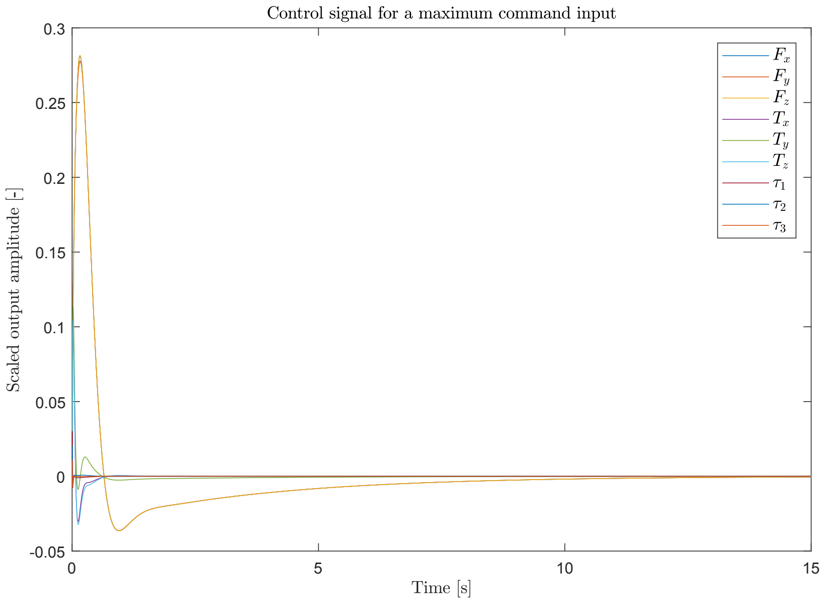


Figure 5.16: Control signal commanded by the controller for a maximum command input vector

Finally, the third weight, imposing constraints on the complementary sensitivity $\mathbf{T}$, is used to determine the noise attenuation of the system. In Eq. 5.6, the complementary sensitivity is also multiplied by the reference vector $\mathbf{r}$. The objective is to have $\mathbf{T}$ as close to one as possible in magnitude for the frequency range in which good command following is expected, meaning that $\mathbf{S}$ is close to zero in this range. Command following was thus already characterized by the first weight factor. It is nonetheless interesting to check the coherence of the resulting behavior. Fig. 5.13 shows that the gain for $\mathbf{S}$ in dB for low frequencies (below ω_B) is very low and it is very close to 0dB (1 in magnitude) for the $\mathbf{T}$ function. The results are consistent and the requirements specified earlier are satisfied.

5.4.5. Unscaling of the Controller

The controller found in the previous section 5.4.4 was designed based on the scaled plant as detailed in Section 5.4.2. However, the objective is to implement the controller in a loop with the unscaled non-linear plant. The controller has to be unscaled such that the whole loop is not influenced by any scaling constraints. Using a block diagram such as Fig. 5.18 shows the desired final configuration.

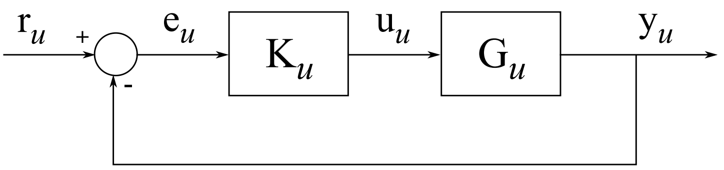


Figure 5.18: Block diagram of the unscaled control loop

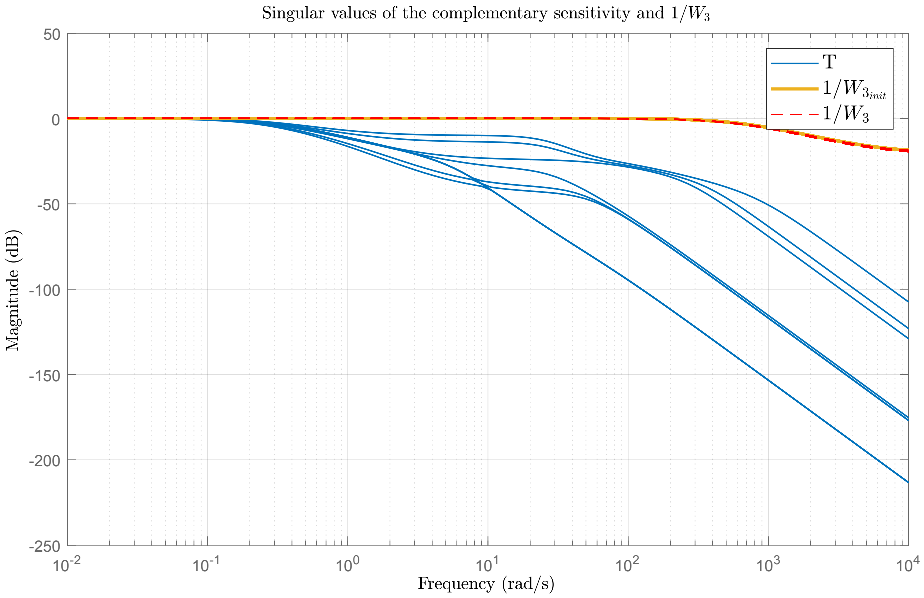


Figure 5.17: Singular values evolution of the complementary sensitivity and of $1/\mathbf{W}_3$

As in Section 5.4.2, a subscript 'u' designates an unscaled object while a subscript 's' is used for a scaled one. So far, the controller takes a scaled error vector as input and gives a scaled control input vector as output. Using the system of equations 5.13, the unscaled controller is obtained from the scaled one by scaling the input vector and unscaling its output vector. Put into an equation, it gives:

$$\mathbf{K}_u = \mathbf{D}_u \mathbf{K}_s \mathbf{D}_e^{-1} \tag{5.20}$$

Fig. 5.18 is given in more details in Fig. 5.19.

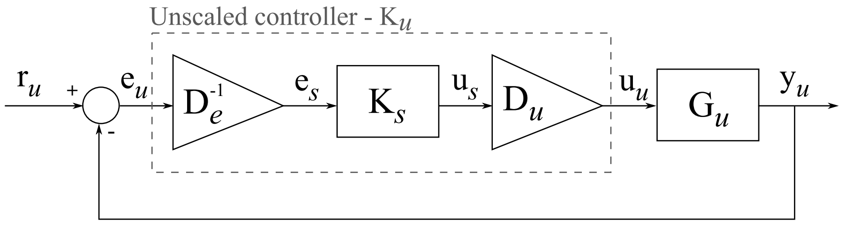


Figure 5.19: Unscaled controller detailed

Finally, even though the performances will be slightly different than initially required (Section 5.4.4), the performance and robustness characteristics of the continuously controlled loop are acceptable. In this chapter, a robust and combined controller was developed based on the linear model of the plant. This controller will be implemented and tested in the next chapter to assess its behaviour in different simulators imposing mission constraints on the control loop.

6

Implementation of the Controller and Results

This chapter focuses on the implementation of the controller designed in Chapter 5 based on the model of the system developed in Chapter 4. First of all, the controller is tested on a linear simulator in Section 6.1 before being tested on a non-linear one (Section 6.2). The latter is not restricted to the mission constraints given in Section 3.3. However, the final implementation in the simulator developed at DLR takes these constraints into account and the implementation of the controller in this environment is described in Section 6.3.

6.1. Linear Simulator

6.1.1. Implementation in Continuous Time

Based on Section 5.4.5, a simulation environment is set up based on Fig. 5.18. The step response of the linear simulation in continuous time is presented Fig. 6.1. The command vector was taken such that the maximum controller scaled input vector $\mathbf{e}_s$ is equal to a 9×1 vector of ones. It means that the unscaled step vector is equal to the maximum error vector expected. The results are coherent since the three translational outputs reach the same final value within the same time as when the scaled loop is studied (Fig. 5.10). The same conclusions and comparisons can be done for the two other types of dynamics investigated. The noise amplitude corresponds to the real measurement noise of the sensors. It is large compared to the command signal but the noise in the output signal is well attenuated.

However, this case is not realistic since the controller is in reality limited by the operating frequency of the actuators as well as of the sensors. Thus, the discretization of the controller will be done and its effects investigated and discussed in the next section.

6.1.2. Implementation of the Discretized Controller

The current AOCS can be operated at various frequencies, up to 250Hz. The objective of the project is to get as close to reality as possible with the hardware available and space certified nowadays. Current AOCS operating frequencies are of the order of magnitude of 10Hz. The continuous controller found in the previous section is then discretized using Tustin's approximation. Among the different algorithms available (zero-order hold, first-order hold, etc.), Tustin's approximation has the advantage that the left half-plane in s-domain transforms in the unit disc in the z-domain which makes the stability analysis more straightforward than other techniques. Also, because the continuous closed-loop was shown to be stable by design of the controller but also by mapping the poles and zeros of the controlled system (Fig. 5.9), the controlled loop with the discrete controller is also stable.

Once the controller is discretized, a simulation of the linear discrete controlled loop can be conducted. In this simulation setup, the plant is kept continuous as it is in reality.

As explained in Section 5.3, the command will be smooth in order to limit the structural efforts on the structure and to reduce the risk of collision. In this simulation step, such a command input is approximated based on the filtering of second order of a step function. The Simulink block diagram is

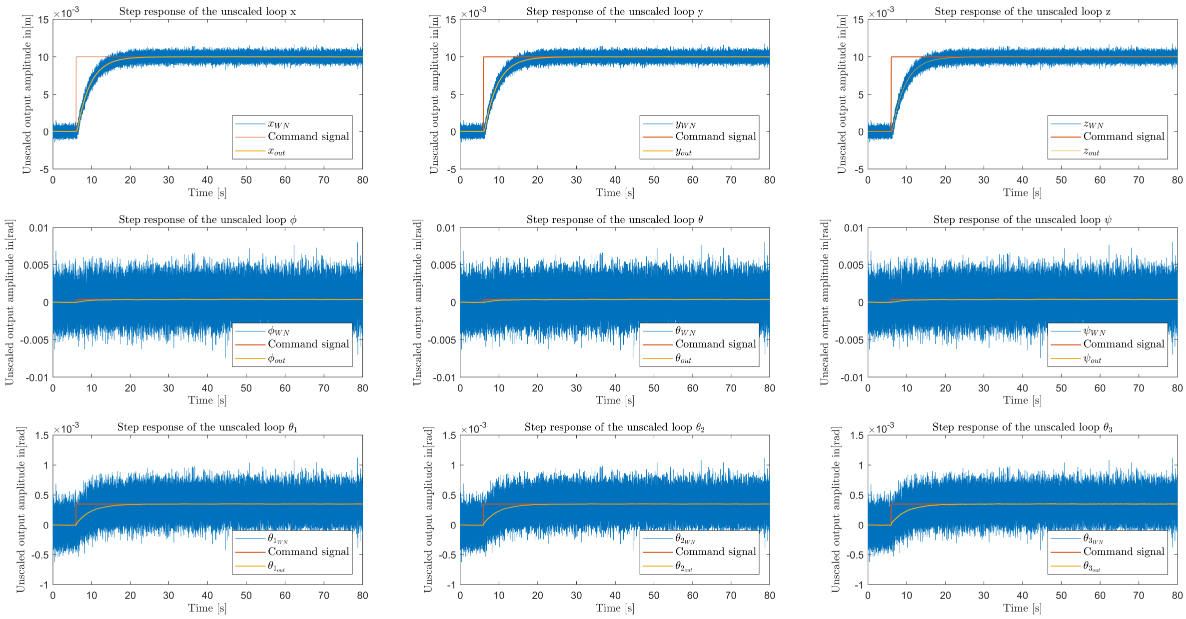


Figure 6.1: Unscaled step response with noise

shown Fig. 6.2 and one component of the output signal is shown Fig. 6.3. The time it takes to reach the steady-state value can be adjusted using the pulsation ω of the second order filter. For the simulation, the final command value is equal to the configuration in which the model was linearized. The initial conditions vector is free to choose. They were chosen here realistically: the base shall stay at the same position and attitude while the second and third joints of the arm rotate slightly.

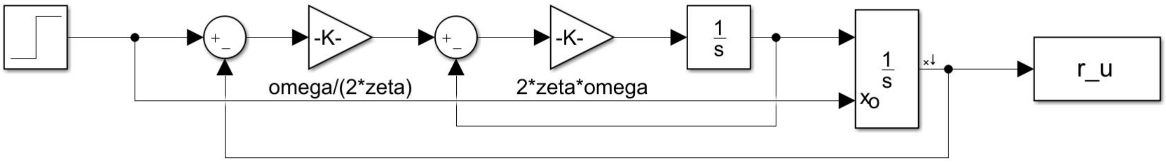


Figure 6.2: Block diagram used to design the command input

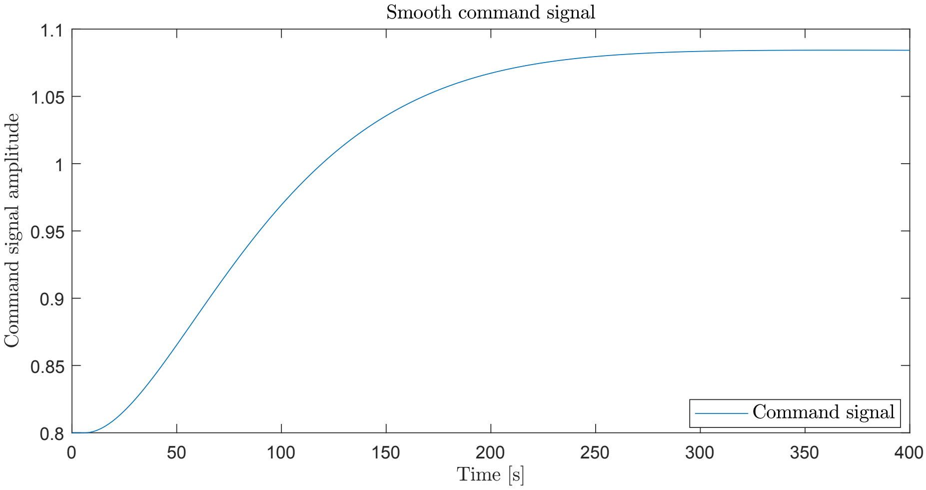


Figure 6.3: Smooth command signal

As discussed in Section 5.3, the six first states ($\mathbf{R}_{C_0}, \boldsymbol{\delta}_0$) shall be the same before and after the deployment maneuver. Only the three joint angles vary between these two times. The state vector chosen for the initial conditions is:

$$\mathbf{q}_0 = \begin{bmatrix} 3.15 & 0 & 0 & 0 & 0 & \pi & 0.3 & -0.9 & 0.8 \end{bmatrix} \tag{6.1}$$

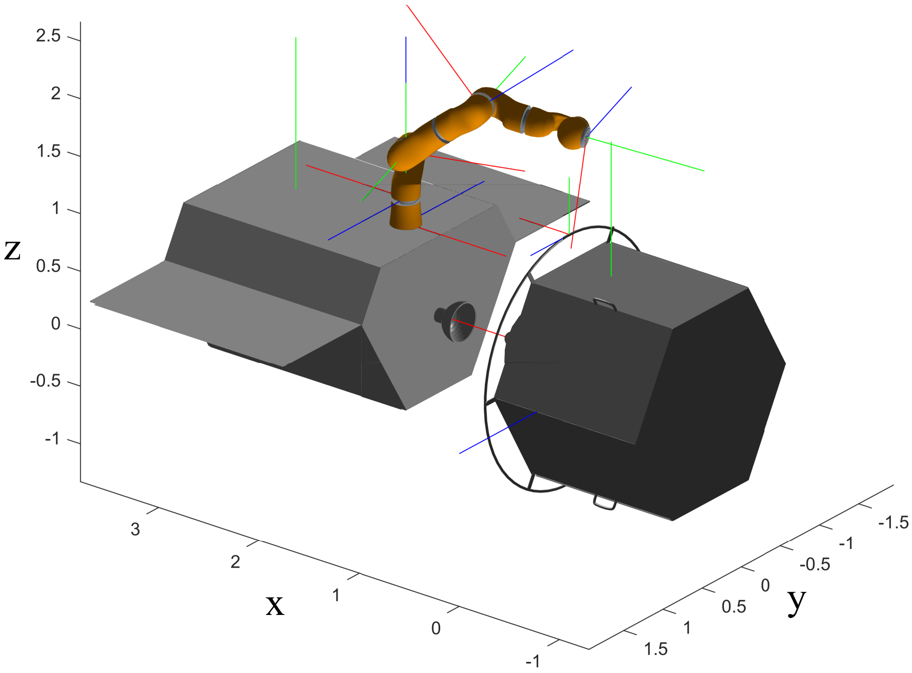


Figure 6.4: Initial configuration of the system

The configuration corresponding to this state vector is shown Fig.6.4. The state vector corresponding to the final configuration is the same as the one used for the plant linearization:

$$\mathbf{q}_f = \begin{bmatrix} 3.15 & 0 & 0 & 0 & 0 & \pi & 0 & -1.0839 & 1.0839 \end{bmatrix} \tag{6.2}$$

The step response to such a command signal is shown Fig. 6.5 in the case without noise and Fig. 6.6 for the case with noise. In the first case, the time needed to reach the final value is about 250 seconds, about ten times greater than the time needed to reach the steady-state value with a continuous controller. Concerning the second case, with noise, the time to reach the steady-state configuration is even four times bigger. Moreover, an important overshoot can be seen. It reaches about 50% of the configuration variation (command step of 0.18 and an overshoot of 0.07). Even though the values are within 10% of the final value after about 400 seconds, it keeps oscillating and if the command step would be greater, the overshoot would be greater too and even the second oscillation may be greater than 10% of the final value.

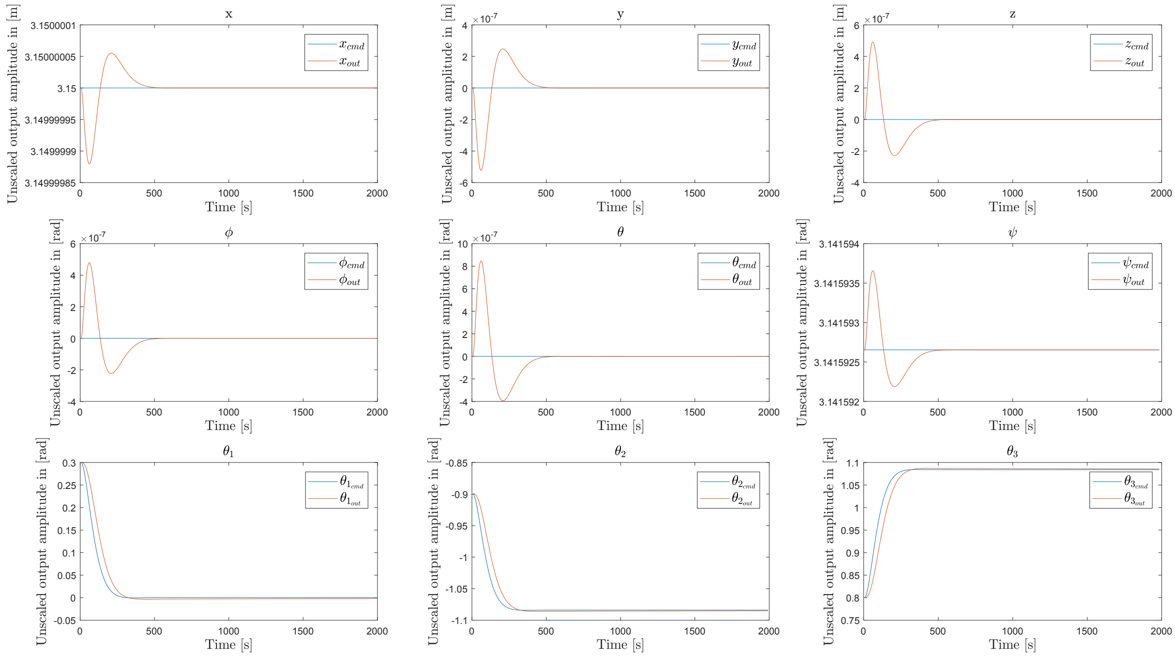


Figure 6.5: Step response with the unscaled discrete ($f_s = 10Hz$) controller implemented - without noise

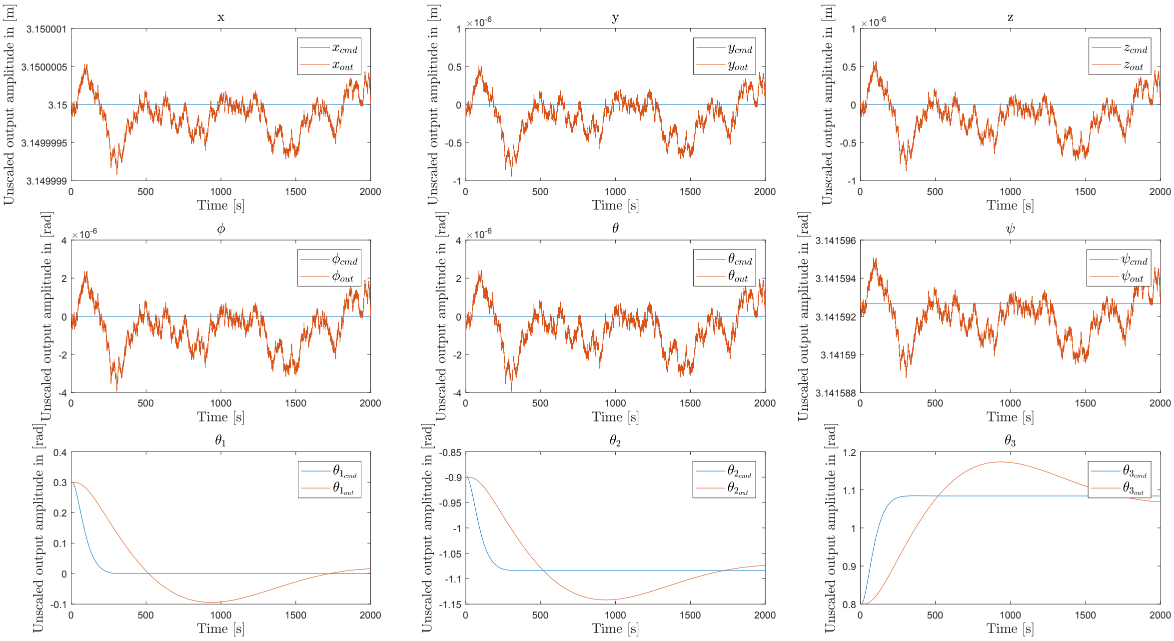


Figure 6.6: Step response with the discrete ($f_s = 10$Hz) controller implemented - with noise

Although the continuous controller found previously was shown to bring good performance and robustness to the controlled loop, its performances, once discretized at $10Hz$, are not satisfying. Indeed, taking more than thirty minutes to complete the deployment of the arm is not acceptable since the assumption made previously about the LVLH reference frame of the target being an inertial frame is not valid for a long period of time. Moreover, having a large overshoot is dangerous in terms of collisions with the target satellite. There are two solutions to this issues. The first one would be to include the discretization in the optimization process in order to obtain the best discrete controller instead of getting the best continuous one. The difference between these two may not be tremendous and the issue may remain. The second solution would be to increase the operating frequency of the AOCS. With a higher frequency, the control efficiency may be better thus reducing the time to reach the final configuration and limiting the overshoot more efficiently. A new AOCS operating frequency of $f_s = 100$Hz is selected. The step response with noise is shown Fig. 6.7.

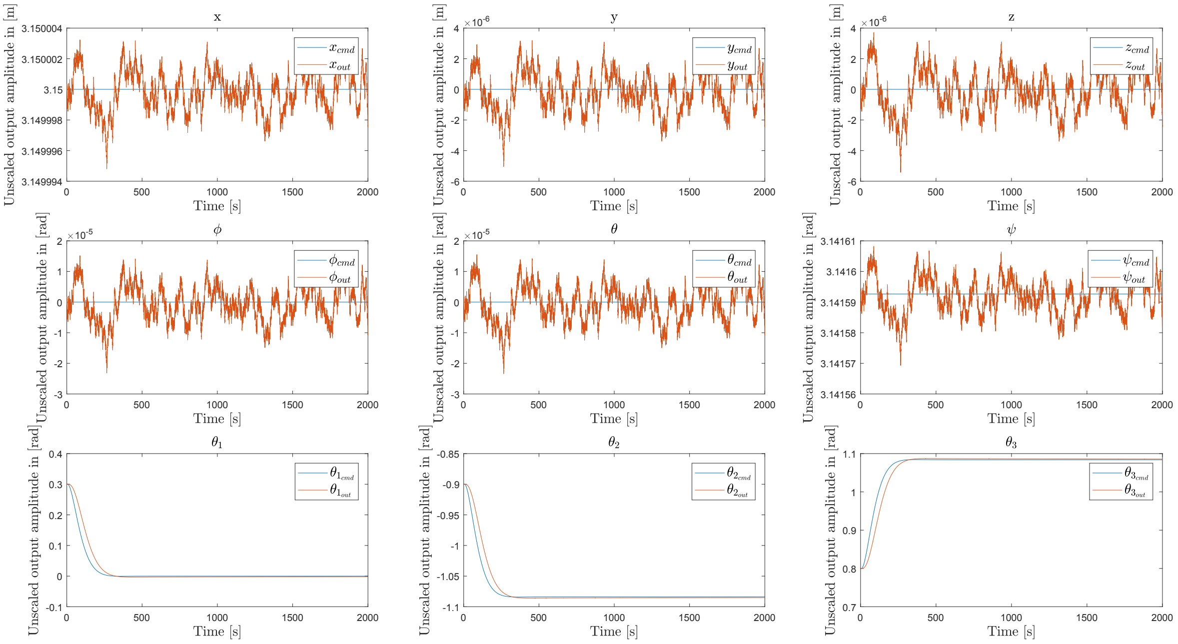


Figure 6.7: Step response with the unscaled discrete ($f_s = 100$Hz) controller implemented - with noise

As shown Fig. 6.7, taking a higher sampling frequency greatly reduces the settling time and the overshoot (very close to zero here). Going from $f_s = 10$Hz to $f_s = 100$Hz is an important step which may not be possible with the technologies nowadays. However, it is clear that increasing the sampling frequency gives better performance and robustness characteristics to the control-loop so even an AOCS operating frequency of 20Hz will improve the response signal for example.
From now on, the robust and combined controller, sampled at $f_s = 100$Hz is considered and will be implemented in different simulation environments with an increasing degree of complexity.

6.2. Non-linear Simulator

A non-linear simulator was developed in order to get one step closer to reality. The principle of this simulator follows the control system shown Fig. 5.2. The controller used is the discretized robust and combined controller designed in Chapter 5. During the design, a linearized model of the plant was used. This time, a non-linear model of the plant is used. The command and the discrete controller form a subsystem triggered by a signal at the operating frequency while the rest of the system is kept continuous. The noise frequency is 1kHz and its amplitude corresponds to the measurement precision. The whole control loop is shown Fig. 6.8. The objective of this first implementation is to evaluate the importance of the non-linearities in the system and to test how it reacts to other uncertainties it was not designed for.

6.2.1. Nominal Tests and Discussion

- **Nominal Tests**

The first test consists in testing the implementation of the controller in the non-linear simulator without taking the noise into account. The effect of the command signal on the states is shown Fig. 6.9. First of all, the states of the base (the six first ones) are slightly impacted by the motion of the arm at the beginning but the controller efficiently corrects this motion in order to bring the spacecraft back to its initial pose. The maximum steady state error is about 0.02mm for the position of the spacecraft and about $7 * 10^{-5}$rad for its attitude. These values are lower than the expected noise on the measurements.

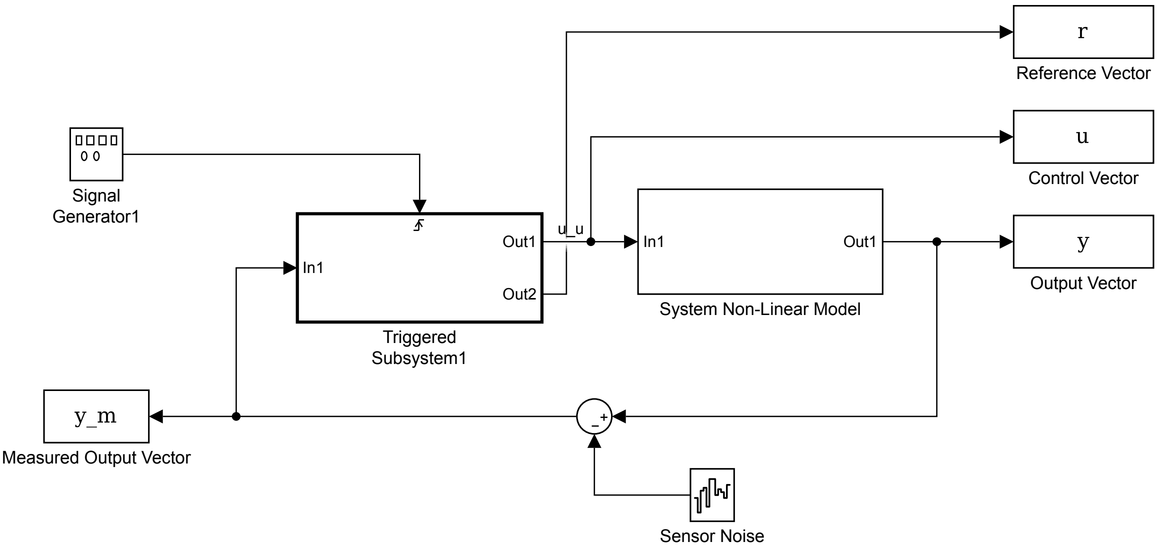


Figure 6.8: Non-linear simulator

Finally, the three last states follow accurately the command. The final steady-state error is lower than $6*10^{-3}$rad which is about two percent of the change in angle between the initial and the final positions. It is important to note that for the second joint angle, the output goes beyond the command, there is an overshoot of about 2%. The performance of the control in this case is good. Neither the non-linearities of the system nor the discretization of the controller have a large impact on it.

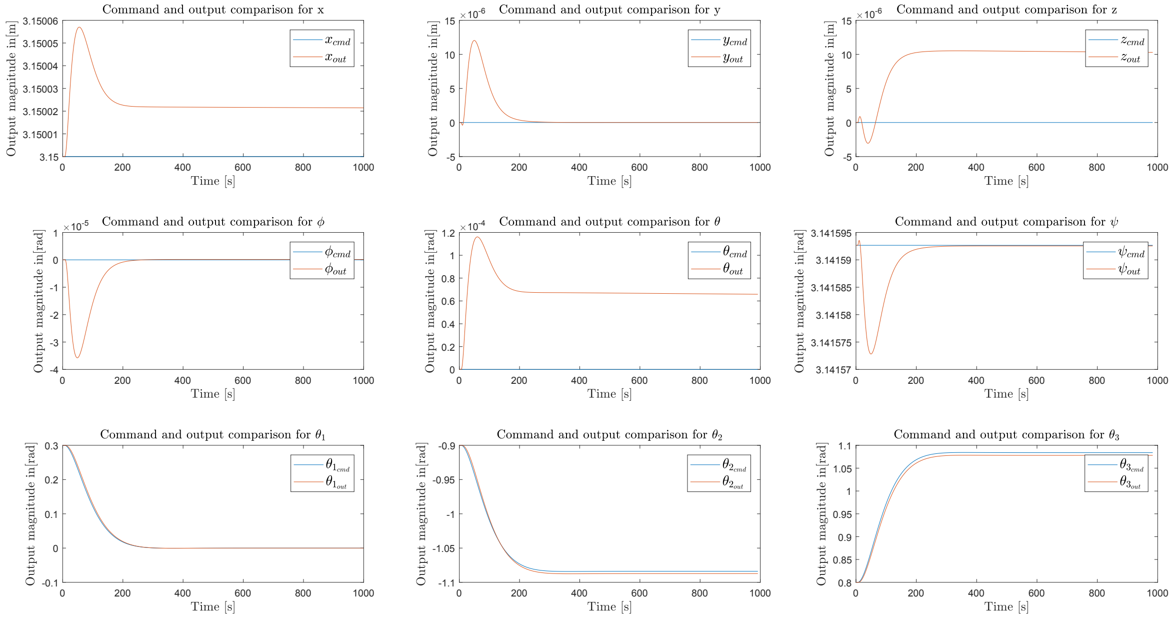


Figure 6.9: Simulation with the discrete controller implemented - without noise

Adding the noise to the system would enable the consideration of the measurement imprecision. It was proven in Section 6.1.2 that the noise does not affect the final state value nor the settling time of the system when a discrete controller is used. Simulating both the noise on the output and the non-linearities of the plant at the same time is closer to the real system. The comparison between the output and the command signals is presented in Fig. 6.10. The first six states are hardly affected by the noise. The shape of the output is the same as for the simulation without noise. However, the impact of the noise is clearly seen on the last three states. Even though the steady-state value for the first joint angle θ_1 is very close to the command, the time it takes to reach it is about one-third longer that without noise. The same trend was observed Fig. 6.7 when the noise was added in the linear simulator. Nonetheless, not all three last states behave in the non-linear simulator as they do in the linear one. In fact, the influence of the noise in the non-linear simulator is important compared to its influence when the system non-linearities are not considered (Fig. 6.7). In the latter, the final values for the last two joint angles were very close to the command values. In this case, they are not. More precisely, the values after one thousand seconds for the second and third joints are respectively $\theta_{2_f} = -1.108\text{rad}$ and $\theta_{3_f} = 1.046\text{rad}$. The differences between the actual value and the expected one are $\Delta_2 = -0.0241\text{rad}$ and $\Delta_3 = 0.0379\text{rad}$. The magnitude changes of the command are -0.1839rad and 0.2839rad. Finally, the steady state error is in both cases about 13% of the expected state variation. In the relative RF, the final EE position is:

$$\mathbf{p}_{EE_f} = \begin{bmatrix} 0.45 \\ 0 \\ 1.09 \end{bmatrix} \text{ in m} \tag{6.3}$$

The difference between the desired and the real final EE positions is then:

$$\Delta\mathbf{p}_{EE} = \mathbf{p}_{Handle} - \mathbf{p}_{EE_f} = \begin{bmatrix} 20 \\ 0 \\ 10 \end{bmatrix} \text{ in mm} \tag{6.4}$$

Even though, the difference in terms of joint angles is important, the difference in terms of position of the EE, $\Delta\mathbf{p}_{EE}$ is small. The mouth of the gripper is way greater than this value (about 10 times greater) so this error of its CoM position is not significant in that case. Also, the diameter of the handle is about 2cm which is small compared to the size of the gripper.

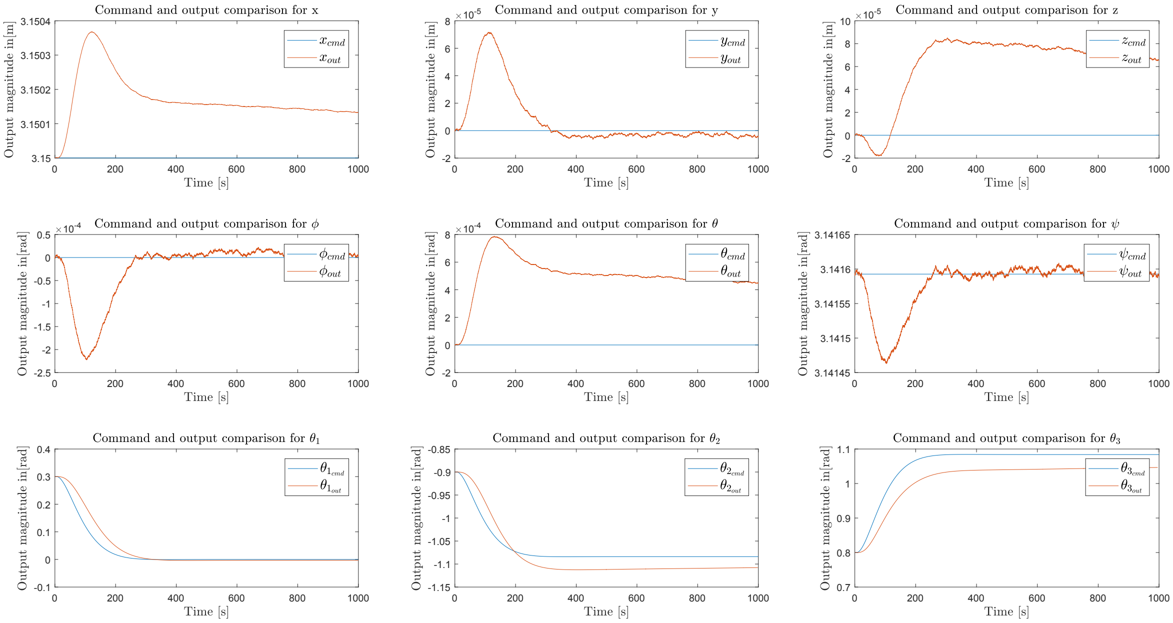


Figure 6.10: Simulation with the discrete controller implemented - with noise

This difference is quite important compared to the amplitude of the change of configuration. As shown previously, no such difference is seen when the discrete controller is implemented in the loop with the linear plant and noise (Fig. 6.7) nor when the discrete controller is implemented in a loop with the non-linear plant and noise (Fig. 6.9). Another simulation is completed: the continuous controller is implemented in the loop with the non-linear plant and noise. The six DoF components of the spacecraft do not vary significantly in any of the tests showed before. From now on, only the last three states (*i.e.* the joint angles) will be displayed and studied. The results of this test are shown Fig. 6.11. The command is well followed with a small steady-state error on the first, second and third joint angles of about $6 * 10^{-5}$ rad, 0.003 rad and 0.006 rad respectively. The impact of the non-linearities on the results for the continuous control is limited. These control errors are smaller than the precision of the sensors. Thus, the errors pointed out in the interpretation of the test in the non-linear simulator with the discrete controller implemented and the noise input are not only due to the addition of the non-linearities of the system.

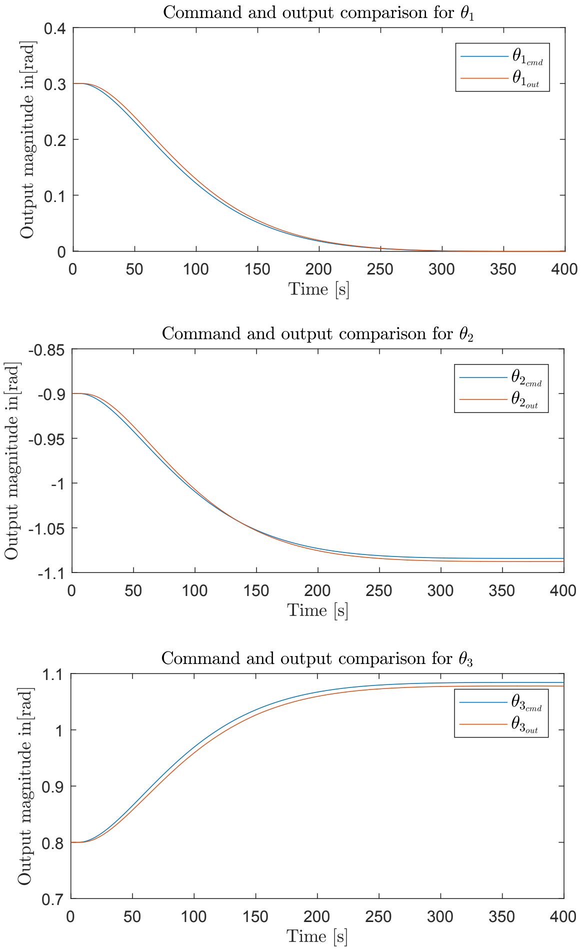


Figure 6.11: Simulation with the continuous controller implemented - with noise

The results are gathered in Table 6.1. The cases discussed in the table which were not presented before are given in Appendix D.

Table 6.1: Results of the simulations

Configuration		Linear simulator	Non-linear simulator
Continuous	w/o noise	Fig. D.1	Fig. D.3
	w/ noise	Fig. 6.1	Fig. 6.11
Discrete - $f = 10Hz$	w/o noise	Fig. 6.5	Fig. D.4
	w/ noise	Fig. 6.6	Fig. D.5
Discrete - $f = 100Hz$	w/o noise	Fig. D.2	Fig. 6.9
	w/ noise	Fig. 6.7	Fig. 6.10

Table 6.1 clearly shows that, based on the simulations completed, a discretization of the controller at $10Hz$ is in fact not satisfying in most of the cases. It also demonstrates that the error found for the test of the discrete controller at $100Hz$ with noise in the non-linear simulator is neither a consequence

of the non-linearities only nor of the noise only. Indeed, when these two cases are separated, the results obtained are satisfying. Only when these two constraints are applied, the steady-state error gets relatively large. During the design of the controller, a requirement on the steady-state error was formulated for the continuous controller. It is shown in the table that this requirement is met. However, when the controller is discretized and implemented with the non-linear plant and measurement noise, the requirement does not hold anymore. Because the discretized case was not considered during the design of the controller, one cannot be sure that the steady-state error after discretization will be below the required value for a continuous control loop.

- **Discussion**

Additional tests were completed to clarify the reason of this steady-state error on the joint angles and to study the effects of the non-linearities of the model. The results of these tests are not plotted here, but they can be found in Appendix D.
First of all, the decoupling of the joints is investigated. It was noticed earlier that the states are decoupled in the continuous and linear case (Section 5.4.4). The effects of the discretization and of the non-linearities are studied. On the one hand, when the first joint angle is the only state with a different final value than its initial value, the final value for θ_1 is very close to the command. However, a non-negligible variation of the two last joint angles is observed (Fig. D.6). On the other hand, when the two last joint angles are commanded, the final value of the first joint remains almost unchanged but the final angular values of the second and third joints do not reach the final value within 1000s (Fig. D.7). It means that the two last joints are affected by the motion of the first joint but the latter is not influenced by the motion of the two last joints. When commanding the second joint only, the third joint is also slightly influenced (about 1% of its amplitude and about 5% of the difference between the initial and the final states of the second joint, Fig. D.8). The states are not completely decoupled anymore.
However, the amplitude of the error vector between the commanded and the real steady-state values depends on the amplitude of the change of configuration as well as how far from the linearization configuration the simulation takes place. For instance, based on the same final configuration $\mathbf{q}_f$, different simulations were conducted. The first one consisted in testing a change of configuration during which the commanded change of each joint angle is less than the half of what was considered during the nominal tests. The second one tested a change of configuration which was larger than twice the change required nominally. The exact values are given in Table 6.2. Of course, the error vector itself between these two tests and the nominal one (Fig. 6.10) cannot be compared directly. However, the percentage of the error compared to the amplitude of the change of configuration could be compared. For the configuration tested previously, the difference vector between the end configuration and the initial one was:

$$\Delta\mathbf{C}_1 = \begin{bmatrix} 0 \\ -1.0839 \\ 1.0839 \end{bmatrix} - \begin{bmatrix} 0.3 \\ -0.9 \\ 0.8 \end{bmatrix} = \begin{bmatrix} -0.3 \\ -0.1839 \\ 0.2839 \end{bmatrix} \text{ in [rad]} \qquad (6.5)$$

The results showed that the component with the steady-state error for the last two joints compared to the amplitude of the change in configuration was about 13%. If the non-linearities play a large role, the error for a small change of configuration close to the linearization point would be smaller with respect to the amplitude of the change. Inversely, for an important change of configuration, it would be larger than 13%. The configurations chosen and the results obtained are shown in Table 6.2.
It demonstrates that the precision of the deployment depends on how large the change of configuration is. Indeed, from a small variation to a big one, the percentage of error with respect to the amplitude of the change goes from 6% to about 20%. This increase of the percentage difference between the expected and the real steady values can have two explanations. On the one hand, it may be the effect of non-linearities: the further from the linearization configuration, the larger the error. On the other hand, it can be related to the large amplitude of the maneuver, the errors build up faster than linearly. To determine the impact of the non-linearities, another test is carried out. It consists in considering changes of configuration with the exact same amplitude on each joint axis but more or less far from the linearization configuration. Table D.1 shows the different maneuvers tested. These final configurations correspond to very different sets of joint angles. The results, given in Appendix D, show that the amplitude of the difference between the expected steady-state and the real one is not smaller close to the linearization configuration than far from it. It means that for a same amplitude of configuration

Table 6.2: Impact of the amplitude of the configuration change on the steady-state error

Parameter	Expression	Unit	Smaller Amplitude	Nominal of Configuration	Larger Amplitude
Initial configuration	$\mathbf{C}_i$	rad	$\begin{bmatrix}0.1\\-1\\1\end{bmatrix}$	$\begin{bmatrix}0.3\\-0.9\\0.8\end{bmatrix}$	$\begin{bmatrix}0.8\\1\\-1\end{bmatrix}$
Final *commanded* configuration	$\mathbf{C}_{fc}$	rad	$\begin{bmatrix}0\\-1.0839\\1.0839\end{bmatrix}$	$\begin{bmatrix}0\\-1.0839\\1.0839\end{bmatrix}$	$\begin{bmatrix}0\\-1.0839\\1.0839\end{bmatrix}$
Final *real* configuration	$\mathbf{C}_{fr}$	rad	$\begin{bmatrix}0\\-1.088\\1.079\end{bmatrix}$	$\begin{bmatrix}0\\-1.108\\1.046\end{bmatrix}$	$\begin{bmatrix}0\\-1.268\\0.67\end{bmatrix}$
Steady-state error wrt the magnitude of the change	$\boldsymbol{\Delta} = \left\lvert\frac{\mathbf{C}_{fr}-\mathbf{C}_{fc}}{\mathbf{C}_{fc}-\mathbf{C}_i}\right\rvert$	%	$\begin{bmatrix}0\\4.8\\6.0\end{bmatrix}$	$\begin{bmatrix}0\\13.1\\13.3\end{bmatrix}$	$\begin{bmatrix}0\\9\\19.9\end{bmatrix}$

change, the steady-state error is about the same. It demonstrates that the non-linearities of the plant developed using Simscape do not have a large impact on the results.

In the ideal case, the servicer would approach the target with its robotic arm stowed. Because of the its architecture, the stowed configuration of the KUKA LWR is limited to the following set of joint angles:

$$\begin{bmatrix}\theta_1\\\theta_2\\\theta_4\end{bmatrix} = \begin{bmatrix}-0.1077\\1.0790\\2.0897\end{bmatrix} \text{ in rad} \tag{6.6}$$

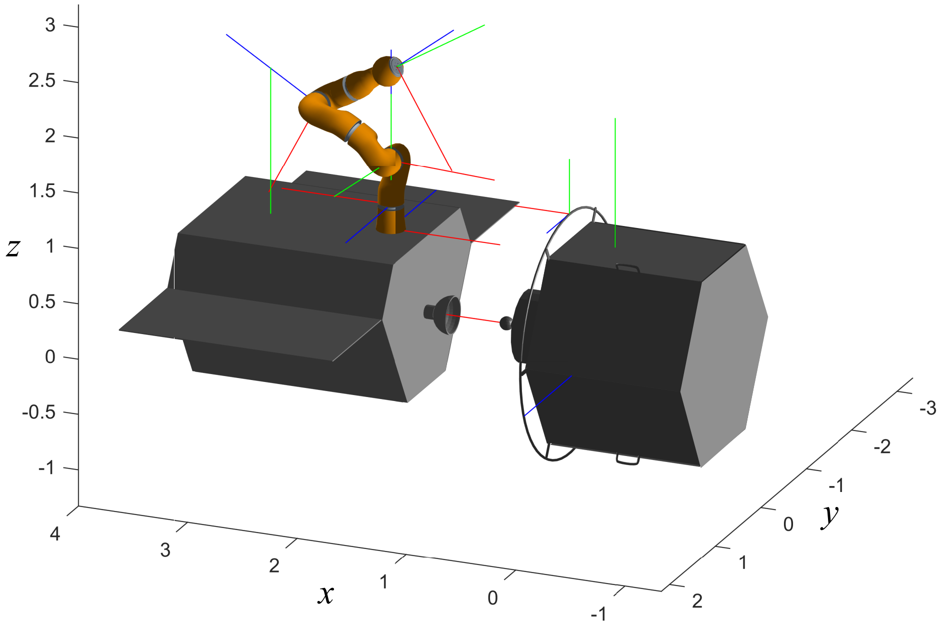


Figure 6.12: Stowed configuration of the satellite

This configuration is shown Fig. 6.12. The whole maneuver, from the stowed to the final configurations is simulated. The results are presented Fig. 6.13. The final end-effector position is:

$$\mathbf{p}_{EE_f} = \begin{bmatrix} 0.39 \\ 0 \\ 1.13 \end{bmatrix} \text{ in m} \tag{6.7}$$

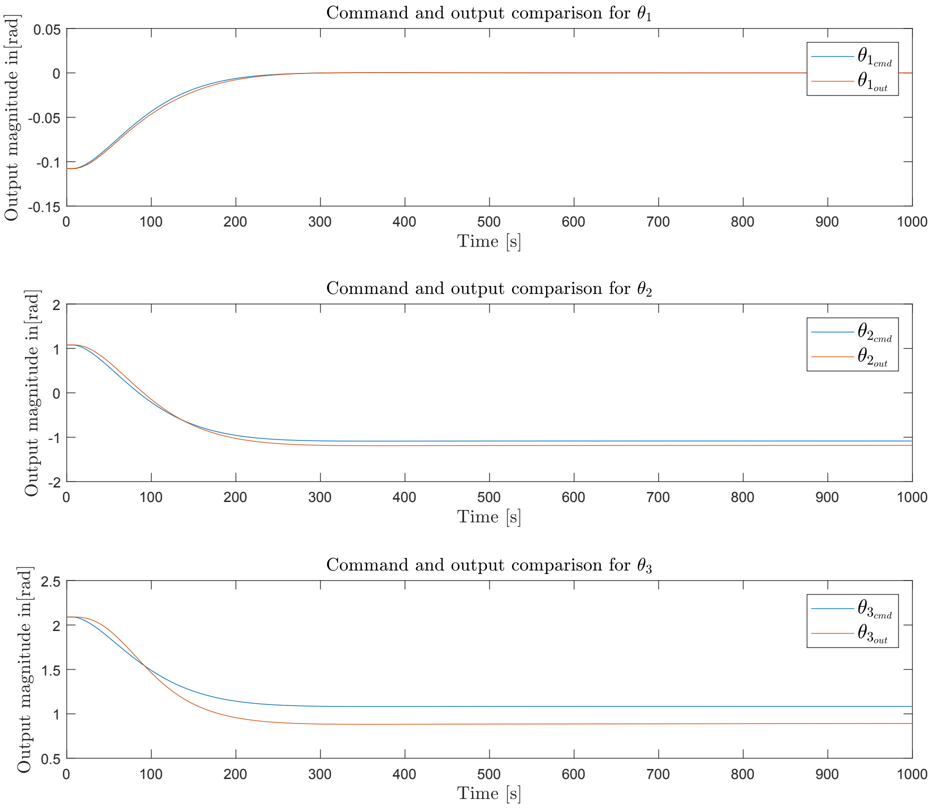


Figure 6.13: Evolution of the joint angles during the simulation from the stowed configuration to the handle in the non-linear simulator, with noise and the controller discretized at 100Hz

It means that the error along the x-axis is about 8cm which is large compared to the size of the gripper. Completing the whole maneuver with this unique command and this controller would not be safe. As explained earlier, the optimization of the controller done in this thesis is very elementary since having the optimal controller for the maneuver is not the objective of the thesis. A better optimization may bring satisfying results. Another method would be to increase the operating frequency. The E2E simulator used in the next section is a real-time simulation environment. Thus, increasing the frequency may not be possible. Indeed, the computations may be too slow to be completed in one time-step. This aspect will be further detailed in Section 6.3. Some tests with the operating frequency pushed to its maximum (*i.e.* 250Hz) are completed. The controller, discretized at 250Hz is then implemented in the non-linear simulator with noise. Fig. 6.14 shows the evolution of the joint angles compared to the command signals. The steady-state errors on each joint angle is smaller than the one observed for the simulation with the discretized controller at 100Hz. The higher the frequency, the higher the performance of the control. This conclusion has already been drawn in Section 6.1. This controller will be the one implemented in the simulator at DLR if it is proven that the simulation can run at this frequency in real-time. If so, the results obtained will be compared to the ones obtained on the non-linear simulator (Fig. 6.14).

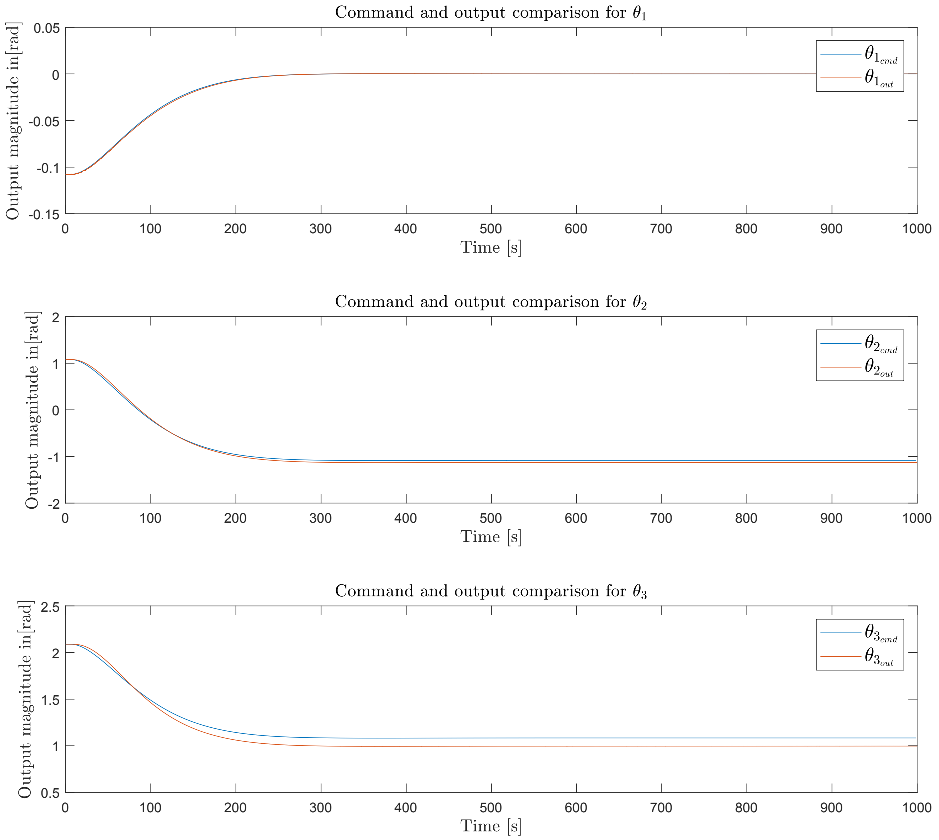


Figure 6.14: Simulation using a controller discretized at 250Hz in the non-linear simulator, with noise

Similarly, for the whole maneuver, from the stowed configuration to the required EE position, the steady-state error is greatly reduced by increasing the operating frequency to 250Hz. The error vector would in this case be:

$$\Delta \mathbf{p}_{EE} = \mathbf{p}_{Handle} - \mathbf{p}_{EE_f} = \begin{bmatrix} 40 \\ 0 \\ 7 \end{bmatrix} \text{ in mm} \tag{6.8}$$

These results are very satisfying with respect to the precision required to be sure that the handle is within the mouth of the gripper at the end of the maneuver.

In this section, the controller was tested in a loop with the non-linear model of the plant. First of all, it showed that the controller, discretized at 100Hz, handles well the non-linearities when there is no measurement noise. It also showed that the continuous controller handles the non-linearities well even with noise. However, when the measurement noise is added to the non-linearities of the plant and the system is controlled with a discrete controller, the performance decreases. By considering all these uncertainties and constraints for which the controller was not formerly designed (except the measurement noise), the controller was pushed out of the zone in which it works properly. However, its behavior remains acceptable even in this complex case. When it is discretized at a higher frequency, like 250Hz, the performance of the loop is even more acceptable. However, in a real mission, many more constraints are applied to the spacecraft and most of them were not considered during the design of the controller. In the next section, the controller will be submitted to harder conditions and its behavior will be analyzed.

6.2.2. Robustness Complementary Study

In the previous section, the controller was implemented in the non-linear simulator with noise, after being discretized. Even though the requirements considered during the design of the controller in Chapter 5 were imposed to the continuous case, the discrete one has given decent results. Now, the reach phase will be simulated under some constraints which were *not* considered during the design. The change in behavior of the loop will show its limitations.

First of all, some uncertainties in terms of masses and geometry are tested. If the masses of each link of the arm (including the gripper) is increased by 10%, the difference in terms of steady-state values with respect to the *nominal* case (shown Fig. 6.14) is very small, with a maximum difference along the z-axis of about 1cm. If the masses are increased by 30%, the results are also about 1cm away from the final EE position of the nominal case. The length of link is increased by 10%. The difference of the results cannot be compared in terms of EE position anymore but comparing the joint angles is still valid. In this case, the difference with the nominal case is about 0.01rad. For an increase in length of 30% of all the links, the difference reaches 0.06rad. These values are relatively small. The magnitude of the increases investigated here are greater than the ones usually encountered. Indeed, the mass of an object is usually known with a better precision than 10%. The same goes for the lengths. The controller designed previously was not designed to handle such uncertainties but it does because it was designed very conservatively.

The effect of some other uncertainties are interested to study. For instance, the measurement of the relative position and attitude of the servicer with respect to the target is done using cameras. A visual tracking algorithm computes the relative pose of the servicer. The processing of the image may take time which implies that delays may appear on the feedback loop. The effect of such a delay is investigated hereafter. The effect of such a delay appears to be small. Indeed, even with a delay of one second, the difference with the nominal case is less than 1mm. However, if this delay is too large, the system gets unstable (for a delay of two seconds with the controller designed and discretized at 250Hz for instance). One second to process the data collected by the camera is also large so the instability in that case will be avoided. Another source of delay may be the during the transfer of the control vector, from the controller output to the system. Simulations show that for a delay of four time steps (one time step being 0.004s for an operating frequency of 250Hz), the results are still acceptable with a difference of less than one centimeter. However, the results diverge quickly with a growing delay. For a ten-step delay, the difference is about 5cm and it goes to 15cm when the delay is of 30 steps. The magnitude of this delay in the OOS-E2E simulator will be discussed in the next section.

The dynamics of the actuators was taken into account neither during the design of the plant nor of the controller. However, the response of the actuators to a step is smoothed due to their limited time constants. Simulations were conducted using first order filters between the controller and the plant, such that when the controller requires an action from the plant, it takes some time to reach the expected value. Thrusters usually quickly reach their final thrust values. The time constant of the first order filter was taken equal to 0.1 and 0.3 seconds and the final EE positions obtained are similar to the nominal test.

Another very strong constraint in reality is the difference of operating frequencies between the arm (about 1kHz) and the base (a few Hz only). Until now, everything has been simulated at the same frequency so the arm and the base are both controller at the frequency chosen. The last test on this simulator consists in discretizing the controller at 1kHz and changing the frequency of the triggered system to 1kHz too. The inputs of the plant would then become signals at the new operating frequency. However, the AOCS is assumed to be operated at 10Hz so the first six signals coming out of the controller are sampled using a zero-order hold with a sample time of 0.1s. The final EE position is very close to the one found in the nominal case. The error is about 2mm only along both the x- and z-axes. It means that when the controller was first discretized at 10Hz, it was too slow for the arm but not for the base. This is certainly specific to the characteristics of the RICADOS project. Indeed, the mass of the arm is very small compared to the mass of the base. It was shown in Chapter 4 that the plant was unstable but the effect of an action of a realistic amplitude on the base is very small thus easier to control than on the arm. If the base was only a few times heavier than the arm, the operating frequency of the AOCS may need to be higher to counteract efficiently the reaction on the base of the arm's motion.

Finally, all the results are gathered in Table D.2 in Appendix D. Moreover, several sources of uncertainties were examined. Some of them such as the noise simulated in all the simulations as well as in the tests completed in Section 6.2.1 were considered during the design of the controller. Some others such as the ones discussed above were not. However, thanks to the way the controller was designed, it is enough conservative to handle these uncertainties to large levels. The robustness characteristics of the simulated controlled loop is good even though the precision of the nominal case is limited.

6.3. OOS-E2E Project Simulator

The controller was discretized and tested in several simulators already in the previous sections. In this section, its implementation in the simulation environment developed at DLR is described and the results are analyzed. An overview of this simulator was given in Section 1.2.2. Some technical specifications will however be added here in order to have the keys necessary to successfully implement the controller.

6.3.1. Description of the OOS-E2E Simulator

First of all, the simulation environment developed during the E2E project at DLR is a hybrid simulator. Both hardware and software parts are integrated in the loop. During the reach phase, which is the one studied in this thesis, the EPOS section of the facility will not be used. However, a representation of the complete architecture using a block representation can be found Fig. 6.15. Because the duration of the thesis is limited and because the required performance to meet in order to be allowed to use the hardware-in-the-loop simulator is high, the robotic arm in the OOS-SIM is also replaced by a numerical model. One part of the simulator is a SASI. This software is capable of computing the dynamics of both satellites based on the equations of motion. Some additional models are also implemented in the SASI such as the actuators models (as discussed in Section 3.1), the models of power and thermal subsystems as well as the whole on-board data-handling system [9]. The latter is the subsystem enabling the reception of TCs and the emission of TMs on-board. These TMs are transmitted from the SASI to the ground segment which was developed, while the TCs are sent from the ground segment to the SASI. The ground segment is designed exactly as for a real mission. The three consoles described in Section 1.2.2 are a part of it. The operators can use them to communicate with the satellite. The rendezvous console can be used to change guidance modes and to receive camera images for instance while the robotic one is used to require a change in joint angles and allows the operator to receive data from the arm.

From the consoles to the space segment, any data goes through the network infrastructure of GSOC which handles the data transmission on ground. On both sides of the border between space and ground segments, the data has to be organized and encrypted according to standardized network protocols such as UDP (User Data Protocol) as well as spaceflight protocols like CCSDS (Consultative Committee for Space Data Systems). Once received on the other side, it has to be decoded, reorganized and interpreted in order to be ready to use by the consoles or the SASI. During this thesis, the data-handling procedures are not studied but they have an impact on the control of the reach phase. In fact, every transformation through which data goes adds delays and because these transformations are not ideal, pieces of data can be lost or end up giving wrong information to the receiver. In order to overcome this issue, ring buffers are used. When a piece of information is lost or erroneous, it is replaced by another value, stored in the buffer and interpolated based on the previous data transmitted. Even though this value used to replace the missing or wrong one is not exactly the expected one, it is used in order to keep the real-time system going. When a value is replaced by its interpolated corresponding one, the system will keep working. However, the output of the system will be a bit different than what it should have been if the real data point would have been received instead of the replacement value. It may bring imprecision in the control loop. Especially if the error rate of these protocols is high.

As mentioned in the previous paragraph, the simulator works in real time. This is a major constraint for each subsystem. Indeed, nothing can be computed in advance and every piece of information shall be processed quickly in order to limit delays. For instance, during the reach phase, the controller shall be able to process the inputs and to determine the control inputs quickly enough in order to achieve good performance. Implementing a controller as a state-space system is a good solution. In fact, the computational power needed to find the control input from the error vector is low because it is only a matter of matrices multiplications.

The robotic arm is replaced by its model in the simulator used to test the controller designed in Chapter 5. These changes are not of major importance since the architecture of the simulator is only slightly modified. They are nonetheless detailed in Fig. 6.16. The dashed lines represent the organization of the simulator when the HW-in-the-loop parts are considered. In the case studied here, they are not and they are replaced by the blocks linked with continuous lines on the figure. The HW in the loop is replaced by a SW in the loop block. In the OBC, different subsystems are implemented such as the thermal or power ones. The combined controller designed earlier is also implemented in the OBC. It

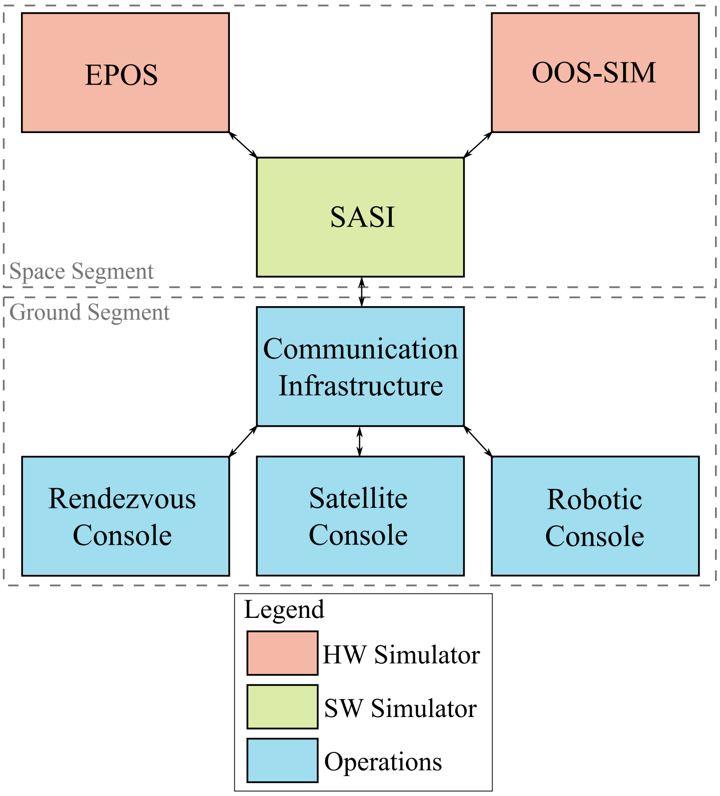


Figure 6.15: Overview on the sub-system level of the simulation environment - adapted from [9]

exchanges with both the model of the simulator and the DYSI. The space manipulator control task, previously operated from ground, is now replaced by the autonomous combined controller.

The model of the system is split. On the one hand, the spacecraft dynamics is handled by the DYSI in which the orbital dynamics is computed. Compared to the non-linear simulator developed in the previous section, the DYSI is more advanced. The influence of celestial bodies (such as the Earth) is considered as well as constraints due to the environment such as solar pressure for instance. On the other hand, the model of the robotic arm is a model of a three-axis of rotation robotic arm developed on Simscape as the one mounted on the base in the Simscape modelization described in Fig. A.1. So the model of the dynamics of the system is not implemented as one in the simulator. However, they are physically linked: if the pose of the base changes, the arm will be influenced and *vice versa*. Thus, a system of master/slave models is implemented. The DYSI computes the states of the base. These states are sent to a kinematically driven model of it, its slave. The master numerical model of the arm is mounted on it and affected by this change of pose. At the same time, the arm is moving during the reach phase. Its new configuration is transmitted back to the DYSI where a slave model of the arm is implemented. The effects of the arm motion on the base are thus effectively taken into account in the dynamics of the base. This way, the effects of the base motion are taken into account in the motion of the arm and the latter influences the motion of the base, as it does in reality.

For the reach phase to start, the operators in the control room send TCs. After being encoded and arranged, they are transmitted to the OBC. The latter receives information from the DYSI and the arm model such as measured data. It is used by the combined controller to determine the error between the command received and the current state of the system. The controller then computes the forces and torques needed to reach the command. They are communicated to the numerical model of the space manipulator as well as to the dynamics simulator. The effects of the control input determined by the controller is thus simulated by these models of the system. The new state of the system is obtained and goes through the controller again. This loop is completed at every time step, in real-time, until the maneuver is over.

To complete the implementation and be able to use the combined controller implemented on-board from ground, a new AOCS mode had to be created within the on-board software. This mode is activated and deactivated using a TC from ground via the satellite console. A TC is also dedicated to the setting of combined control parameters. The AOCS subsystem with combined control gets values from sensors as inputs. When the HW-in-the-loop parts are used, the joint angles of the manipulator are obtained from sensors in the joints and the pose of the base is computed. Indeed, a camera is placed at the EE. Using visual tracking of the target, the relative pose of the camera is determined. The pose of the servicer is then derived using FK (from the EE to the base this time), based on the camera pose and on the measured joint angles. When numerical models are used instead of HW parts, the position vector is obtained differently. On the one hand, joint angles are obtained from the model. On the other hand, the pose of the camera cannot be computed with the visual tracking method anymore. The pose of the base is instead obtained directly by considering the states of the model. These measurements are compared to the command signal. The latter is not a step response, as explained in Section 6.2. Indeed, to reduce the structural efforts on the system and to increase the precision of the maneuver, an smooth command signal is designed to go from the initial state vector to the final one. The interpolated signal is computed based on the initial and final configurations as well as on the duration of the maneuver required. The longer the maneuver, the smoother the command signal. At each step, the error between the commanded states and the measured ones is computed. Based on this vector, the controller computes the actions to be completed and forward them to the thrusters and to the actuators at the joints.

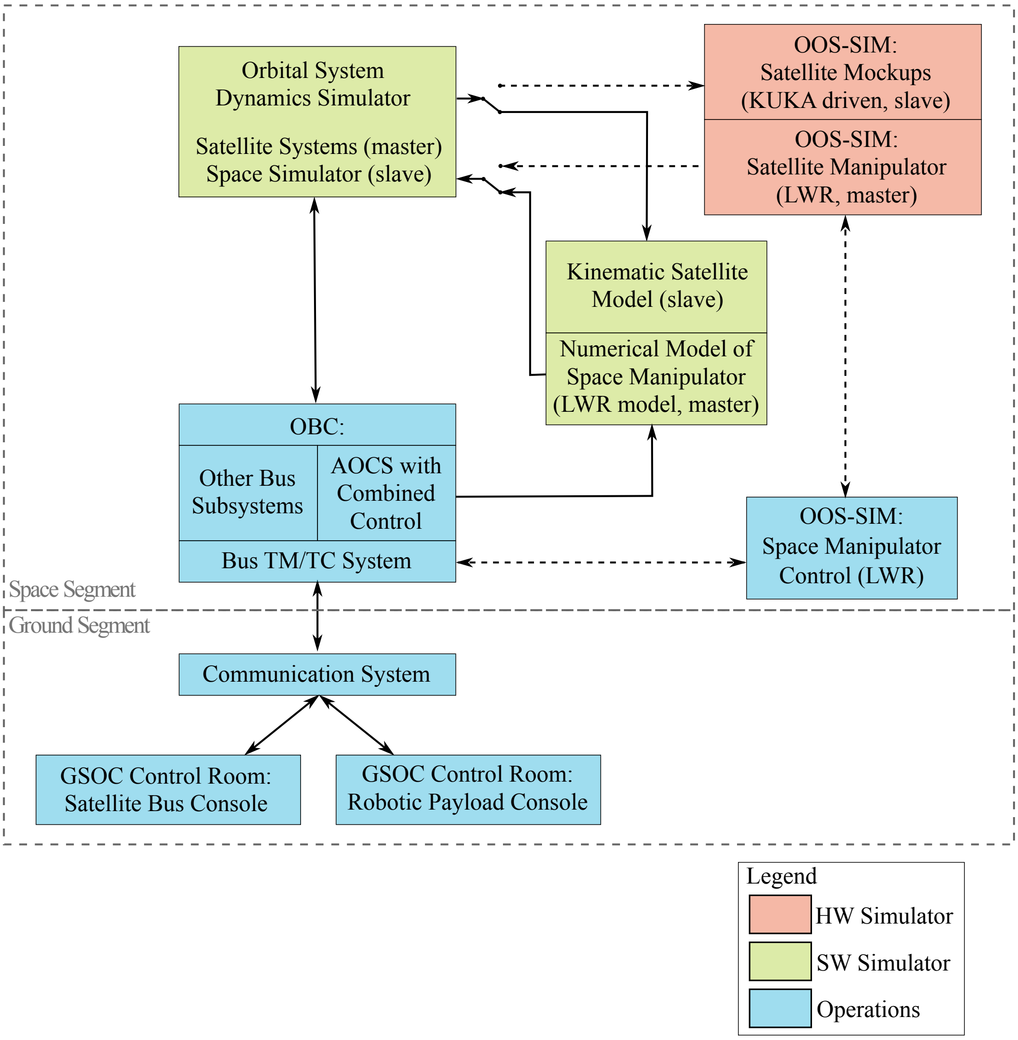


Figure 6.16: Simulator used for the implementation of the controller detailed

A detailed explanation of the data circulation and of the interfaces within the simulator would help to understand the way the simulator works and how the controller shall be implemented. The uplink is graphically described in Fig. 6.17. In order to operate the new controller from the control room at the German Space Operations Center (GSOC), some additional TCs are created. On ground, they are stored in the Mission Information Base (MIB). They shall be correctly interpreted on-board. Thus, they are added to the database in the on-board software. These databases are represented on the left side of Fig. 6.17. The implementation of new TCs is done offline beforehand. Based on these, an operator in the control room can send a TC selected from the telecommand pool to the satellite via the satellite console. Every TC has a corresponding identification name (ID) and a set of parameters. Before being transmitted to the satellite, it is encoded in space packets, following international standards protocols. The transmission from ground station to spacecraft is done in space packets according to the CCSDS protocol. All subsystems on-board provide an ECSS (European Cooperation for Space Standardization) PUS (Packet Utilisation Standard) compatible interface such that an operator can have access to all subsystems by predefined TCs and TMs [9]. Once uplinked, the function management service use a pointer to find the function in the on-board database, based on the ID of the TC received. The latter is composed of a packet primary header, a packet secondary header, the application data and the error control. For the combined controller, the primary and secondary headers are respectively the appropriate address *i.e.* AOCS and the PUS 8.1 (Service Type 8, Subtype 1). The choice of secondary header is arbitrary and depends on the architecture of the system. The application data of the TC consists in the desired satellite pose and the camera pose, the trajectory duration and two control flags. The first one is used to choose if the base pose in the TC shall be considered or if it shall remain still. The latter may be useful because the position of the base is not continuously communicated to ground such that it is not always possible to copy the current position and send it in the TC. The second flag is used to define the way the approach is completed. It can be either by a point-to-point or a Cartesian transition. The latter means that the EE trajectory is a straight line while the point-to-point approach is done by controlling the motion of the joints instead of the EE. Once the function is found and called, the AOCS mode is set to combined control mode for instance and the parameters are set. Based on these parameters, the combined control algorithm (or another one in case another mode is activated) is executed. At every step, the algorithm checks that the parameters have not changed. If they have, they are updated.

In addition to the format in which data is sent and received, the way it is transported is also important. Indeed, it is a major difference between a simulation on Earth and a real satellite in space. During a space mission, a satellite is launched and it communicates with ground using radio-frequency signals sent and received via antennas and other electronic devices. However, using this kind of setup on Earth is not realistic and complex to realize. In order to get results close to the real space mission, the signal would for instance need to travel a long way in order to get the right delay. At DLR, another communication system was designed using UDP for the terrestrial data transfer [9]. The similar path as for a real mission is realized on ground. It allows very strict timing settings: delays of less than 500ms with jitter (deviation from true periodicity) lower than 1ms and a 2.5ms period for single data packet.

Concerning the TMs, the standard housekeeping TM of the servicer includes the relative motion of the servicer with respect to the target at an update rate of 10Hz. These details are displayed on the servicer console such that the maneuver can be overseen from ground. Moreover, on the robotic console, the entire scenario, including the robotic arm is displayed.

Finally, the controller needs to be compatible with the C environment of the simulation environment. The sate-space matrices are then exported from Matlab to a text file which can be implemented in the simulator. The sampling frequency of the controller is taken as high as the real-time machine can handle. Indeed, there is a frequency above which the computation time of any of the parts of the simulator cannot keep up with the real-time simulation. A frequency of 250Hz is below this critical frequency and corresponds to the maximum achievable frequency for the AOCS implemented. This frequency is then considered for the operating frequency of the real-time simulator.

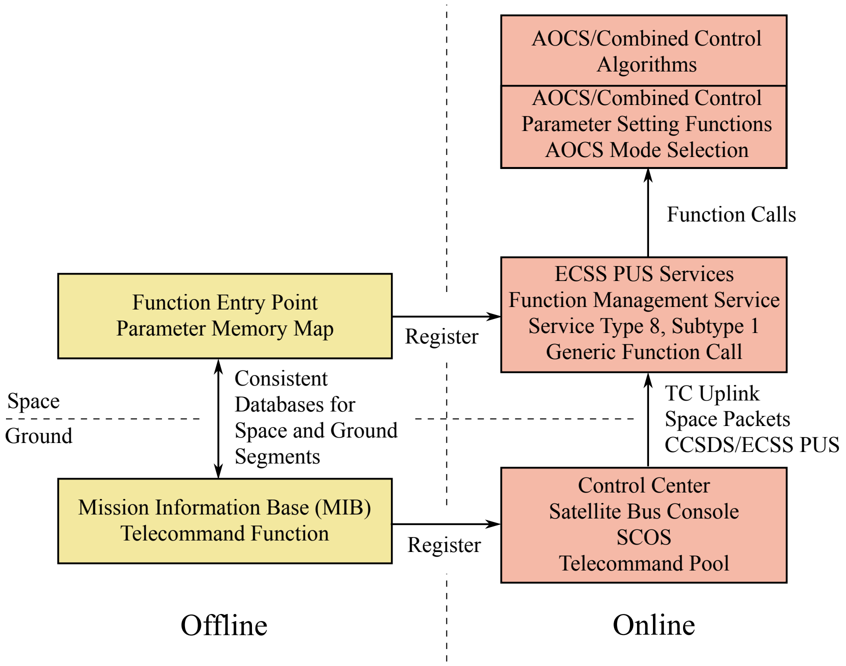


Figure 6.17: Details of the uplink communication

The way the data is handled on-board has been barely discussed so far. To work properly, the combined controller has to receive measurements from the arm and the base. These signals go through the communication front-end before being received by the controller. The communication front-end completes the task of a relay station between the simulator and the real-time machine. It sorts the packets of data received and checks that everything is on time. If not, values interpolated from the previous steps and stored in a ring buffer are used. Once the controller has received the states of the system, the torques and forces to be applied are computed depending on the command vector. The joint torques are directly sent to the robotic arm. The AOCS efforts however go through the communication front-end again, before being transmitted to the satellite motion simulator. All these interfaces are operated at 250Hz. Every time transmission of data between two subsystems, it gets delayed of one time step. If the analogy is made with the non-linear simulator shown Fig. 6.8, it means that between the controller and the plant, there is one step delay for the joint torque and two steps for the AOCS forces and torques. Moreover, there is some delay on the feedback loop too. Indeed, between the time at which the measurement is taken until the time when it is considered by the controller, there are two time steps for both the measurements coming from the arm and from the base. Finally, the robotic arm and the base master models send their measurements to their 'slaves'. For the joint angles, it goes only through the communication front-end. For the base pose measurement, it goes through it as well but also through the combined controller for convenience reasons. The circulation of these signals are shown Fig. 6.18.

6.3.2. Results of the Implementation

The description of the E2E-OOS simulator in the previous section (Section 6.3.1) shows the way the controller shall be implemented to work properly. First of all, some tests were conducted to check if the system could handle 250Hz in real-time and showed that the simulator is keeping up with the high operating frequency. Thus, the controller discretized at 250Hz can be used and the whole simulation environment can run in real-time at this frequency. Then, a new TC was developed to be able to control the maneuver from ground. It is used to set up the parameters of the reach phase. It consists of the initial position vector of the system as well as the final one. Moreover, the transition time is also to specify. The control flag shall be sent too as explained in the previous section, in order to clarify the way the system is controlled. Another TC was modified in the MIB. It is the one used to switch the active control mode to the combined control one such that the combined controller is made active and the maneuver is completed. The combined control mode had to be referenced in this TC. Of course, the parameters of the maneuver have to be set before activating the control mode. These TCs can be selected from the control room and be sent to the satellite. A specific interface is used on ground. It

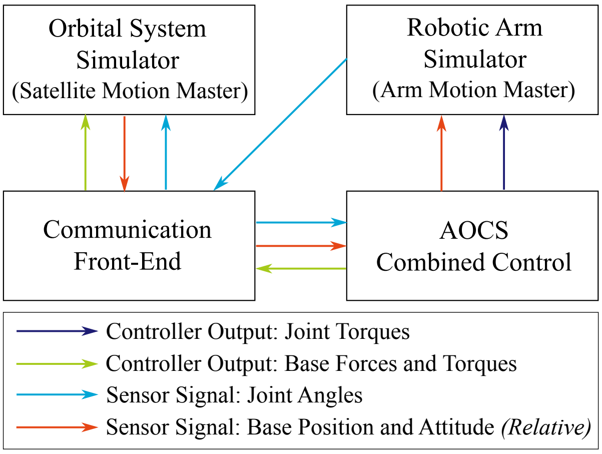
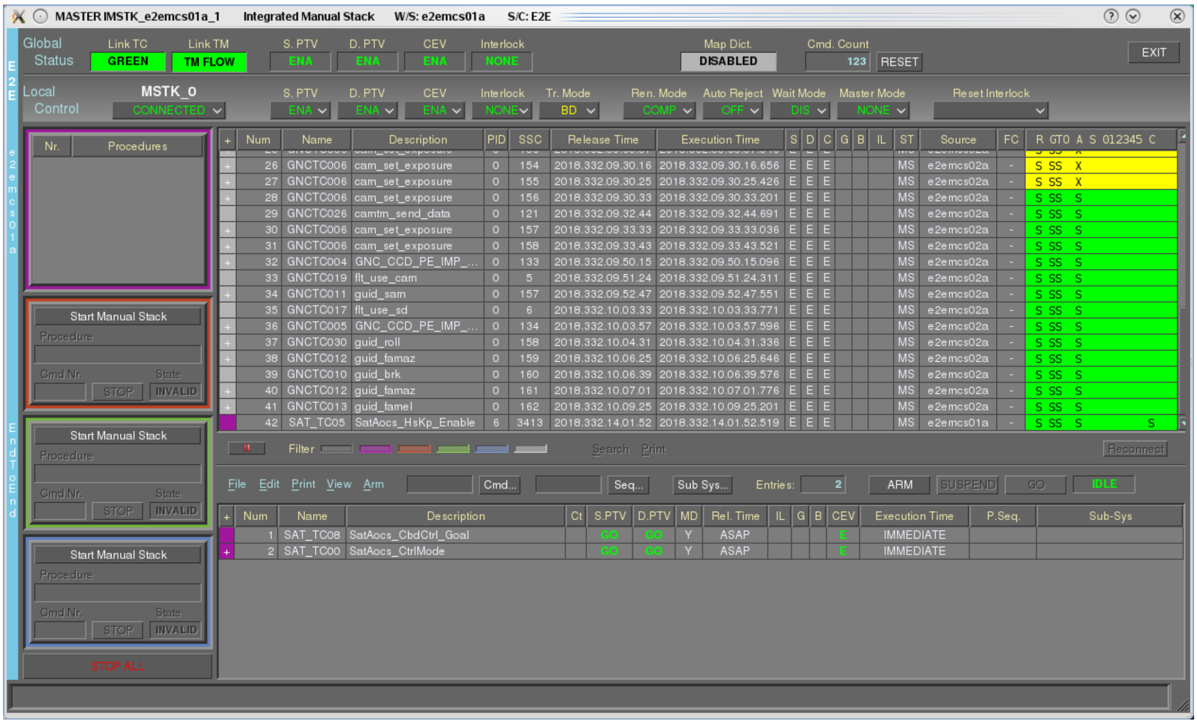


Figure 6.18: Description of the system inputs and outputs signals circulation

is presented in Fig. 6.19. On this figure, the TCs to be sent are stacked in the queue. Once sent, they appear in the upper list with a description of its state in the last column. When the cell color is green, the commands have successfully been sent and acknowledged, as shown Fig. 6.20.



MASTER IMSTK_e2emcs01a_1 Integrated Manual Stack W/S: e2emcs01a S/C: E2E

Global Status: Link TC GREEN; Link TM TM FLOW; S. PTV ENA; D. PTV ENA; CEV ENA; Interlock NONE; Map Dict. DISABLED; Cmd. Count 123

Local Control: MSTK_0 CONNECTED; S. PTV ENA; D. PTV ENA; CEV ENA; Interlock NONE; Tr. Mode BD; Ren. Mode COMP; Auto Reject OFF; Wait Mode DIS; Master Mode NONE; Reset Interlock

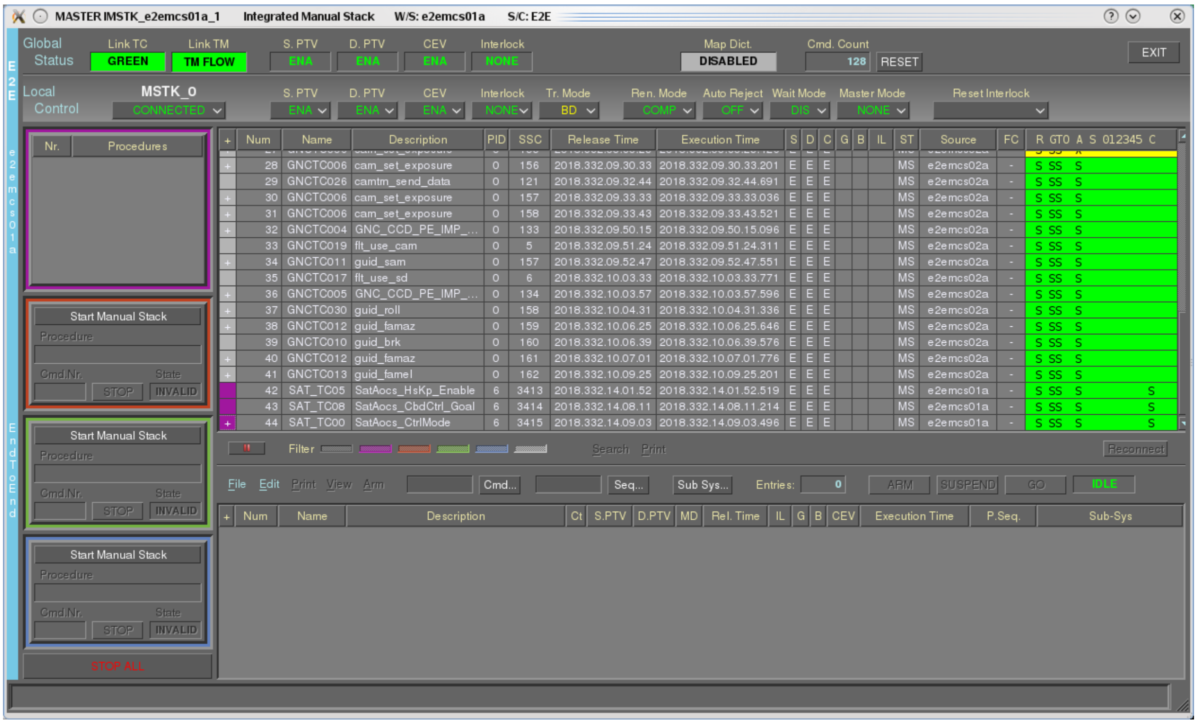
+	Num	Name	Description	PID	SSC	Release Time	Execution Time	S	D	C	G	B	IL	ST	Source	FC	R GT0 A S 012345 C
+	26	GNCTC006	cam_set_exposure	0	154	2018.332.09.30.16	2018.332.09.30.16.656	E	E	E				MS	e2emcs02a	-	S SS X
+	27	GNCTC006	cam_set_exposure	0	155	2018.332.09.30.25	2018.332.09.30.25.426	E	E	E				MS	e2emcs02a	-	S SS X
+	28	GNCTC006	cam_set_exposure	0	156	2018.332.09.30.33	2018.332.09.30.33.201	E	E	E				MS	e2emcs02a	-	S SS S
	29	GNCTC026	camtm_send_data	0	121	2018.332.09.32.44	2018.332.09.32.44.691	E	E	E				MS	e2emcs02a	-	S SS S
+	30	GNCTC006	cam_set_exposure	0	157	2018.332.09.33.33	2018.332.09.33.33.036	E	E	E				MS	e2emcs02a	-	S SS S
+	31	GNCTC006	cam_set_exposure	0	158	2018.332.09.33.43	2018.332.09.33.43.521	E	E	E				MS	e2emcs02a	-	S SS S
+	32	GNCTC004	GNC_CCD_PE_IMP_...	0	133	2018.332.09.50.15	2018.332.09.50.15.096	E	E	E				MS	e2emcs02a	-	S SS S
	33	GNCTC019	flt_use_cam	0	5	2018.332.09.51.24	2018.332.09.51.24.311	E	E	E				MS	e2emcs02a	-	S SS S
+	34	GNCTC011	guid_sam	0	157	2018.332.09.52.47	2018.332.09.52.47.551	E	E	E				MS	e2emcs02a	-	S SS S
	35	GNCTC017	flt_use_sd	0	6	2018.332.10.03.33	2018.332.10.03.33.771	E	E	E				MS	e2emcs02a	-	S SS S
+	36	GNCTC005	GNC_CCD_PE_IMP_...	0	134	2018.332.10.03.57	2018.332.10.03.57.596	E	E	E				MS	e2emcs02a	-	S SS S
+	37	GNCTC030	guid_roll	0	158	2018.332.10.04.31	2018.332.10.04.31.336	E	E	E				MS	e2emcs02a	-	S SS S
+	38	GNCTC012	guid_famaz	0	159	2018.332.10.06.25	2018.332.10.06.25.646	E	E	E				MS	e2emcs02a	-	S SS S
	39	GNCTC010	guid_brk	0	160	2018.332.10.06.39	2018.332.10.06.39.576	E	E	E				MS	e2emcs02a	-	S SS S
+	40	GNCTC012	guid_famaz	0	161	2018.332.10.07.01	2018.332.10.07.01.776	E	E	E				MS	e2emcs02a	-	S SS S
+	41	GNCTC013	guid_famel	0	162	2018.332.10.09.25	2018.332.10.09.25.201	E	E	E				MS	e2emcs02a	-	S SS S
🟪	42	SAT_TC05	SatAocs_HsKp_Enable	6	3413	2018.332.14.01.52	2018.332.14.01.52.519	E	E	E				MS	e2emcs01a	-	S SS S S

Entries: 2 IDLE

+	Num	Name	Description	Ct	S.PTV	D.PTV	MD	Rel. Time	IL	G	B	CEV	Execution Time	P.Seq.	Sub-Sys
🟪	1	SAT_TC08	SatAocs_CbdCtrl_Goal		GO	GO	Y	ASAP				E	IMMEDIATE		
+	2	SAT_TC00	SatAocs_CtrlMode		GO	GO	Y	ASAP				E	IMMEDIATE		

Figure 6.19: Interface used in the control room to send TCs - TCs in the queue

Once everything is implemented as detailed earlier, a maneuver can be simulated. The stowed configuration is considered for the initial one. The final configuration remains the same, equal to $\mathbf{q}_f$. The transition time is taken equal to 300 seconds and the approach is completed using the point-to-point approach. Finally, the initial position of the S/C is set as defined in Eq. 4.55. The TC with these parameters is sent and acknowledged by the system. The second TC which enables the combined control mode is sent and acknowledged. The maneuver is ongoing.



Global Status: Link TC GREEN; Link TM TM FLOW; S. PTV ENA; D. PTV ENA; CEV ENA; Interlock NONE; Map Dict. DISABLED; Cmd. Count 128

Local Control: MSTK_0 CONNECTED; S. PTV ENA; D. PTV ENA; CEV ENA; Interlock NONE; Tr. Mode BD; Ren. Mode COMP; Auto Reject OFF; Wait Mode DIS; Master Mode NONE; Reset Interlock

+	Num	Name	Description	PID	SSC	Release Time	Execution Time	S	D	C	G	B	IL	ST	Source	FC	R GT0 A S 012345 C
+	28	GNCTC006	cam_set_exposure	0	156	2018.332.09.30.33	2018.332.09.30.33.201	E	E	E				MS	e2emcs02a	-	S SS S
	29	GNCTC026	camtm_send_data	0	121	2018.332.09.32.44	2018.332.09.32.44.691	E	E	E				MS	e2emcs02a	-	S SS S
+	30	GNCTC006	cam_set_exposure	0	157	2018.332.09.33.33	2018.332.09.33.33.036	E	E	E				MS	e2emcs02a	-	S SS S
+	31	GNCTC006	cam_set_exposure	0	158	2018.332.09.33.43	2018.332.09.33.43.521	E	E	E				MS	e2emcs02a	-	S SS S
+	32	GNCTC004	GNC_CCD_PE_IMP_...	0	133	2018.332.09.50.15	2018.332.09.50.15.096	E	E	E				MS	e2emcs02a	-	S SS S
	33	GNCTC019	flt_use_cam	0	5	2018.332.09.51.24	2018.332.09.51.24.311	E	E	E				MS	e2emcs02a	-	S SS S
+	34	GNCTC011	guid_sam	0	157	2018.332.09.52.47	2018.332.09.52.47.551	E	E	E				MS	e2emcs02a	-	S SS S
	35	GNCTC017	flt_use_sd	0	6	2018.332.10.03.33	2018.332.10.03.33.771	E	E	E				MS	e2emcs02a	-	S SS S
+	36	GNCTC005	GNC_CCD_PE_IMP_...	0	134	2018.332.10.03.57	2018.332.10.03.57.596	E	E	E				MS	e2emcs02a	-	S SS S
+	37	GNCTC030	guid_roll	0	158	2018.332.10.04.31	2018.332.10.04.31.336	E	E	E				MS	e2emcs02a	-	S SS S
+	38	GNCTC012	guid_famaz	0	159	2018.332.10.06.25	2018.332.10.06.25.646	E	E	E				MS	e2emcs02a	-	S SS S
	39	GNCTC010	guid_brk	0	160	2018.332.10.06.39	2018.332.10.06.39.576	E	E	E				MS	e2emcs02a	-	S SS S
+	40	GNCTC012	guid_famaz	0	161	2018.332.10.07.01	2018.332.10.07.01.776	E	E	E				MS	e2emcs02a	-	S SS S
+	41	GNCTC013	guid_famel	0	162	2018.332.10.09.25	2018.332.10.09.25.201	E	E	E				MS	e2emcs02a	-	S SS S
	42	SAT_TC05	SatAocs_HsKp_Enable	6	3413	2018.332.14.01.52	2018.332.14.01.52.519	E	E	E				MS	e2emcs01a	-	S SS S S
	43	SAT_TC08	SatAocs_CbdCtrl_Goal	6	3414	2018.332.14.08.11	2018.332.14.08.11.214	E	E	E				MS	e2emcs01a	-	S SS S S
+	44	SAT_TC00	SatAocs_CtrlMode	6	3415	2018.332.14.09.03	2018.332.14.09.03.496	E	E	E				MS	e2emcs01a	-	S SS S S

Entries: 0 IDLE

+	Num	Name	Description	Ct	S.PTV	D.PTV	MD	Rel. Time	IL	G	B	CEV	Execution Time	P.Seq.	Sub-Sys

Figure 6.20: Interface used in the control room to send TCs - TCs acknowledged

Once the maneuver has been completed, the data stored can be analyzed. First of all, for this test, the final end-effector position is:

$$\mathbf{p}_{EE_f} = \begin{bmatrix} 0.476 \\ 0.002 \\ 1.136 \end{bmatrix} \text{ in m} \tag{6.9}$$

The error in terms of position with respect to the command $\mathbf{p}_{EE_c}$ is:

$$\Delta\mathbf{p}_{EE} = \mathbf{p}_{EE_c} - \mathbf{p}_{EE_f} = \begin{bmatrix} -6 \\ -2 \\ -36 \end{bmatrix} \text{ in mm} \tag{6.10}$$

This steady-state error vector is very acceptable even though in the non-linear simulator studied previously, the steady-state error without noise was smaller than the one witnessed on the E2E-OOS simulator (Fig 6.9). For the latter, it is important to precise that the maneuver had a much smaller amplitude. Moreover, the evolution of the states of the system can be compared to the command signal. The results are shown Fig. 6.21. The command is well followed and the final configuration is close to the required one. The implementation of the controller in the simulator was done correctly. The TCs are effective and the OBC interpret them correctly. The few steps of delay described earlier seem to have a low impact on the results.

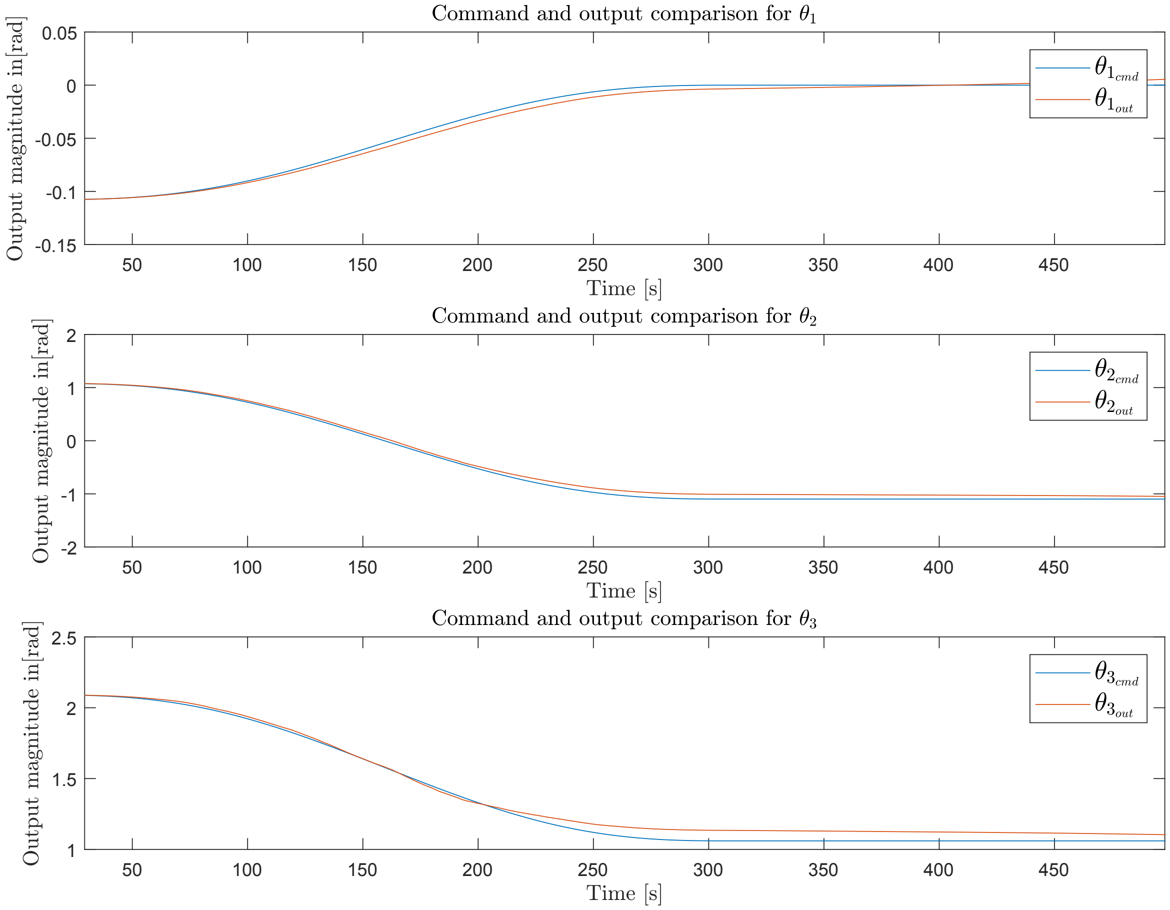


Figure 6.21: Comparison of the command and the real signals on the E2E-OOS simulator without noise

However, this simulation is far from reality. Indeed, the mission constraints were considered during this simulation but other uncertainties such as the measurement noise were left out. The objective of the next test is to study the behavior of the system when this noise is taken into account. It was shown in Section 6.2 that the noise may imply larger steady-state errors. Nonetheless, with an operating frequency of 250Hz, the results obtained on the non-linear simulator were satisfying (Fig. 6.14). The same simulation is completed on the E2E-OOS simulator. The effect of the noise on the states is important and the noise is not well attenuated compared to how it is on the non-linear simulator (Section 6.2). The error with respect to the command on the final EE position is much larger than on the non-linear simulator and not fully stable as shown Fig. 6.22.
The final joint angles are approximately:

$$\boldsymbol{\theta}_f = \begin{bmatrix} -0.003 \\ -0.923 \\ 1.191 \end{bmatrix} \text{ in rad} \tag{6.11}$$

which means that the EE position is:

$$\mathbf{p}_{EE_f} = \begin{bmatrix} 0.509 \\ -0.008 \\ 1.257 \end{bmatrix} \text{ in m} \tag{6.12}$$

The difference with the commanded EE position is quite small on both the x- and y- axes but the error is about 15cm along the z-axis. This error is important compared to the size of the handle and of the gripper. Nonetheless, the changes in terms of position and attitude of the base remain low (about 1cm for the highest change in position and a 1° maximum variation of the third Euler angle). It means that

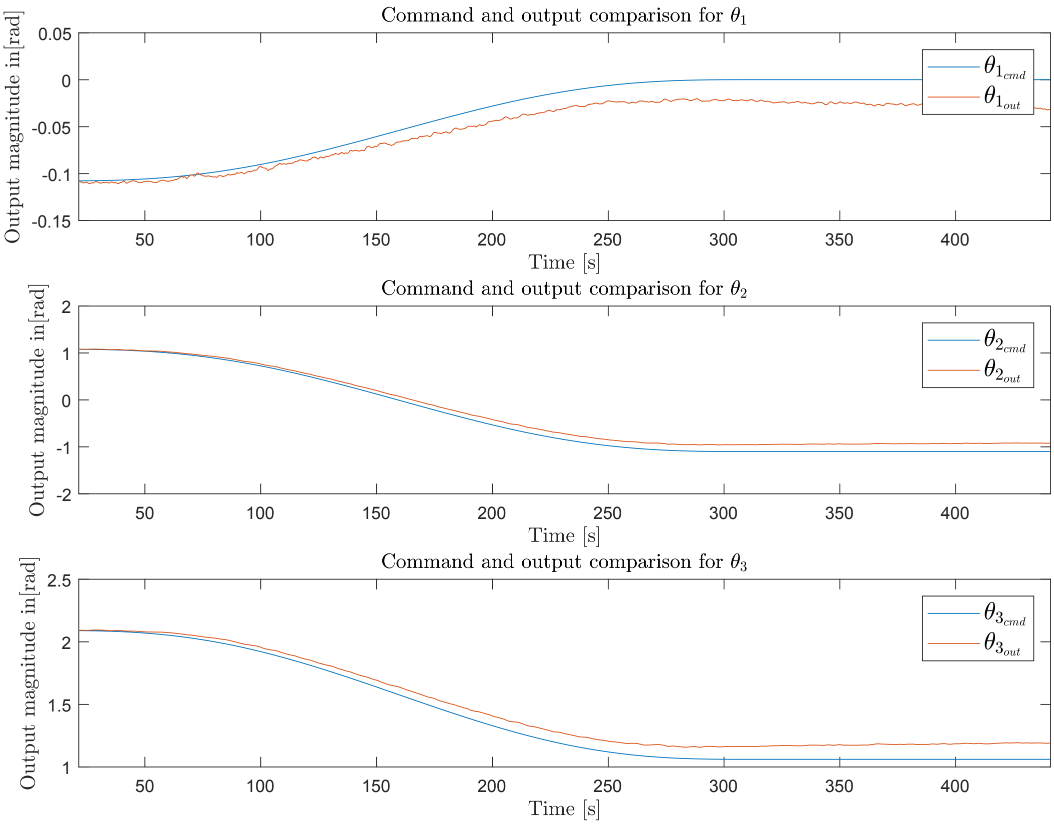


Figure 6.22: Comparison of the command and the real signals on the E2E-OOS simulator with noise

the states of the system are still decoupled. Only the steady-state error vector is largely influenced by the measurement noise. The reason of this discrepancy is not fully understood but several possible reasons are discussed hereafter.

6.4. Interpretation of the Results

First of all, simulating various maneuvers in different conditions in the first section of this chapter helped understanding how well the controller was designed when submitted to the constraints it was designed for. The continuous control loop with noise (Fig. 6.1) corresponds to the case used for the design of the controller. The states are decoupled, the noise is well attenuated and the final configuration is reached with a satisfying precision. The behavior required during the design of the controller is followed. During the simulations with the linear plant, the controller was discretized at different frequencies and tested. The discretization which was not taken into account during the design appeared to have a non-negligible influence. Indeed, with the original sampling frequency of 10Hz, the results are not acceptable (Fig. 6.6): oscillations appear, overshoots are important and the settling time is greatly increased compared to the continuous case. However, when the controller is discretized at a higher frequency (e.g. 100Hz), the results are acceptable.

Then, a new and more complex simulation environment was developed. The controller and the non-linear plant were implemented in a control loop as described Fig. 5.2. To start with, the chosen maneuver was small in order to minimize the influence of non-linearities. The states were shown to be still well decoupled and the final configuration of the arm was close to the commanded one in all cases tested excepted the most complex one, when some noise is added on the measurements and the controller is discretized. Only the combination of these two characteristics implies an increase of the steady-state error vector. To minimize it, the frequency at which the controller is discretized was increased to the maximum frequency that the AOCS of the simulator at DLR can handle. The error on the final configuration is finally acceptable. Then, a study of the influence of non-linearities was completed. It showed that their effects was very small. The complete maneuver (from the stowed configuration to the handle) could then be simulated at once even though the error vector gets a bit larger. Furthermore, an additional robustness study was conducted in order to assess the performance of the controller to various sources of uncertainties discussed in Chapter 3 but which were not taken into account in the controller design. The main conclusion of this investigation is that the controller was designed very conservatively. Indeed, it can handle large imprecision in terms of mass and length of the links of the robotic arm as well as important delays and the dynamics of the actuators. But most importantly, a test was run in which the base was controller at 10Hz and the arm at 1kHz. These values are of the orders of magnitude often used for robotics and for AOCS. The results were equivalent to the ones of the nominal case.

Finally, the mission constraints were taken into consideration when the controller was tested in the E2E-OOS simulator developed at DLR. The constraints to adapt to are detailed in Section 6.3.1. Once correctly implemented, a maneuver was simulated without measurement noise, only with the intrinsic constraints of the simulation environment. The results were acceptable meaning that the implementation was proven to be correct. The final configuration is also close to the commanded one. When the noise is added to the plant outputs and fed back to the controller, the results obtained are very different from the ones described before on the other simulators. Indeed, the noise has a large impact on the states of the system. It comes from one of the differences between the non-linear simulator and the E2E-OOS simulator or from a combination of these differences. Four main dissimilarities can be pointed out. First of all, the dynamics models used in both simulators are different and most importantly, there are two dynamics model in the E2E-OOS simulator against one combined model in the non-linear simulator. This difference in the architecture implies delays. Indeed, the time in between the effort on the base due to the arm's motion and its consideration in the DYSI is not zero in this case but it is in the non-linear simulation environment. Furthermore, the fact that there are two dynamics simulators for the simulator developed by DLR brings another difference. In fact, because a master/slave scheme is set up as explained before, the data is sent from the master to its slave in order to be taken into account by the master of the other subsystem. For instance, the joint angles of the arm are sent to its slave such that the effect of the motion on the base is computed. However, when the DYSI receives the joint angles, they are first filtered to avoid dynamics instabilities before being interpreted. It means that the motion of the arm is interpreted as smoother in the DYSI than what it is in the arm master. Thus, the motion of the base due to the arm would be smoother as well. The effect of this difference is certainly small in the case studied here but would be important if the transition between the initial and final configurations was very short and if the mass of the arm would be comparable to the mass of the base. Moreover, the model of the S/C dynamics in the E2E-OOS simulator is quite different from the second simulator in the sense that it takes into account orbital dynamics as well for instance. This additional simulated phenomena may bring some numerical differences in the simulations. However, these are likely to be quite small and only significant over a long period of time. Finally, one main difference of course comes from the way data is handled. The way it is done at DLR was shown in the previous section. It brings delays both in the feedback loop and in between the controller and the plant. It also brings uncertainties due to the possible errors in transmission. All these differences are certainly at least part of the explanation of the differences between the results of the simulations completed on these two simulators. Narrowing the source of the differences is not straightforward due to the complexity of the E2E-OOS simulation environment. The results of the last test clearly show that an important noise component remains. The controller was designed such that, in continuous time, the noise attenuation is greater than 20dB. In the linear and non-linear simulators, even though the noise may have brought some error on the steady-state configuration, the noise itself was not clearly visible on the output signals. Because it is noticeable on Fig. 6.22, it might come from a secondary noise source in the E2E-OOS simulator which has not yet been identified.

7

Conclusions and Recommendations

Based on the previous chapters and on the results obtained from the simulations, conclusions are drawn in Section 7.1. Then, recommendations are given for future works in the last section (Section 7.2).

7.1. Conclusions

The objective of this thesis was to develop, implement and test a robust and combined controller for the reach phase of a robotic arm mounted on a servicer satellite. In this section, the achievements of this thesis will be exposed and discussed. As explained in Section 1.4, different sub-goals had to be met in order to consider the main research objective reached. These sub-goals were reached one by one during this thesis.

First of all, a model of both the base of the spacecraft and the space manipulator at once was developed. The derivation of such a model is more complex than developing two simpler and distinct models, one for the arm and one for the base. However, developing one unified model was important for the design of the controller and would make the overall model-based design more convenient. Furthermore, the models were developed symbolically. They are thus very easily adaptable to any other spacecraft with the same architecture. Such a tool may be very valuable for the design of a future OOS satellite. Also, the numerical inaccuracies and approximations of a numerical model are avoided.

Then, an overview of the constraints was obtained based on literature and on common knowledge. There are plenty of limitations, uncertainties and disturbances due to the hardware, to the environment and to the definition of the mission. Not all of them can be considered during the design of the controller. After being discussed, a set of constraints was considered for the design of the controller. However, some other constraints could be tested later on, after the implementation.

After studying the constraints that the system would undergo during the reach phase, different control techniques were investigated. H_∞ robust control technique with loop-shaping has been chosen as the most appropriate technique for this thesis project for various reasons which were described in Section 5.2.

Based on this choice, requirements on the performance and robustness of the controller were selected and conditioned the next steps of the controller design. Indeed, based on these requirements, a first guess for the loop-shaping weighting functions was taken and then optimized. Decisions were also made during the scaling of the plant. Even though the scaling factors do not always have a great influence on the results [55], considering the right order of magnitude for these is recommended in order to fully benefit from the potential of this control technique. The controller obtained was studied. It showed that most of the requirements specified were reached and it was ready to be implemented. It proved that, even though designing a combined controller is more complex than designing two different ones for the arm and the base, it can bring good performance and robustness to the controlled system.

Once designed, the robust controller was implemented in several different simulators. First of all, it was set up in a control loop with the linear model of the plant. This scenario was exactly what it was designed for and the results proved that it was correctly done. The second simulator used was close to the first one but instead of the linear plant, the non-linear one was used in the loop. Most of the cases presented great results. The effect of the noise was barely visible and the steady-state configuration was very close to the commanded one. Even after being discretized, the results were acceptable even though the steady-state error was a bit larger in the case where the discretized controller is in the loop with noise and the non-linear plant. However, this case is very extreme compared to the case imagined during the design of the controller. Moreover, the controller demonstrated good robustness with respect to other uncertainties which were not taken into account during the design such as variations in masses and lengths of the arm links, additional delays as well as different operating frequencies of the arm and the base. The outcome of the simulations on the non-linear simulator was beyond expectations. The robustness characteristics of the controller have been investigated and proven satisfying. Finally, the implementation of the controller in the third and last simulator, the E2E-OOS simulator, developed during previous and ongoing projects at DLR was more complex. In fact, a set of hardware constraints were applied to the controller in the previous simulators but mission and environment constraints were barely considered (except for the limitation of computational power on-board). To be able to implement the controller in this complex simulation environment, some interfaces had to be modified and some procedures had to be clarified in order to be able to control the maneuver from the control room at GSOC. After developing a new control mode and adding new TCs in the different pieces of software (on ground as well as on-board), the controller could be implemented. A nominal test was run without the measurement noise and the results were coherent with what has been obtained earlier with the non-linear simulator. However, once the noise source is included, unexpected results are achieved. The reason of this discrepancy is yet to be investigated.

The previous paragraphs demonstrate that the different sub-goals exposed in Section 1.4 and give their respective outcome. Thus, the main research objective is reached and the main research question can be answered. Every step of this thesis increased the TRL of the reach phase of an OOS mission one way or another. First of all, the tool developed to model the dynamics and the kinematics of the arm and the base as one is valuable for the modeling of future servicers. Furthermore, the design of a combined control for the reach phase of an OOS mission is a real innovation. This thesis formulates some guidelines for its design. The lessons learned during this project are given and the final controller proved to have good performance and robustness by meeting the specified requirements. Finally, the implementation of the controller in various simulators also brings the TRL of the reach phase to a higher level. Indeed, the controller was submitted to a wide set of constraints during this thesis and its implementation in the E2E-OOS simulator at DLR is a great accomplishment. The combined control strategy for the deployment of the arm was successfully implemented in the OBC and can be operated from ground. However, to reveal and exploit the great potential of OOS, many more challenges are yet to be taken up. Based on the work completed in this thesis, some recommendations for future works were formulated and they are presented in the next section.

7.2. Recommendations

This thesis project had for main objective to increase the TRL of combined control for the reach phase. The controller developed and implemented in different simulators under some constraints is not fully optimized but it will be used by the DLR as guidelines. Some future works will be based on the work presented in this thesis to increase the TRL again and to get closer to a controller which can meet the requirements of a real space mission in terms of performance, stability and robustness. In this section, some recommendations are given. These are points which would be worth investigating in the future to improve the control of the maneuver. Some of the points stated below are the results of the different assumptions taken at different steps of the project. The TRL would be greatly increased if these assumptions were released in the future.

First of all, the assumptions were more or less strong, but they all made the problem easier to understand, to model and to control. Removing them would make the control of the reach phase more complex, but also closer to reality. It would undoubtedly improve the performance and robustness of the controlled loop. For instance, the flexible modes of the system could be considered. The stiffness of the appendages like the solar panels as well as the robotic arm could be modelled as described in [2] for instance. Sloshing modes as well as other structural flexible modes could also be taken into account. The non-linear model would be different for sure, making the non-linear simulations more realistic. The effect on the linear model would certainly depends on the linearization configuration chosen. Considering a model closer to reality would reduce the unmodelled dynamics uncertainties.

Moreover, uncertainties could be considered as part of the model. Every part could have uncertain characteristics as they have in reality. It would give the model of the system some uncertainties too. They would be considered in the design of the controller such that the robustness of the controller will be able to bring good performance for the total range of the uncertainties. Some other physical aspects of the system such as the model of the behaviour of the actuators could be modelled and influence the behavior of the controller design.

In addition to the modelling of the uncertainties and the improvement of the model, some modifications can be investigated in the design of the controller. Indeed, it was explained in Section 5.4.4 that the same first guesses for the weighting functions for all three kinds of dynamics were equal. Of course, if they were investigated more precisely, better first guesses may be found for the different dynamics. These may be different from one dynamics behavior to the other. This is very likely since the optimization of the set of filters already gives different filters for translation, rotation of the base and for the rotation of the joints, even though they are very close to the first guesses. Furthermore, it may be interesting to run the optimization on the discretized controller instead of the continuous one. The results may be slightly different and another controller might be found at the end.

The definition of the system could also be made closer to reality. For instance, the space manipulator could be taken as it is, with seven axes of rotation instead of the three axes considered in the thesis. It would make the system more complex, especially the derivation of the kinematics and of the equations of motion. The design of the controller would also be more energy- and time-consuming. In fact, the joints may be considered to have the same dynamics behavior at first but they are still very different and taking more than three dynamics behaviors into account may be worth investigating. That would increase the number of variables in the optimization a lot which means that it would take more computational effort to find the optimal controller. Moreover, the forces and torques on the base have been considered as applied to the CoM of the system so far. However, solving the allocation of the efforts to the real actuators (*i.e.* to the thrusters and to the reaction wheels) would be more realistic. The study of this allocation problem was almost not discussed at all in the thesis, except in Section 5.4.2. In the latter, the maximum forces and torques on the base were roughly estimated in order to have a coherent scaling of the plant but a more precise study would be interesting to investigate and the results to compare with those found in this thesis.

Even the whole control strategy would be worth investigating more. This thesis demonstrated that combined control for an OOS mission could be a good control architecture. However, the H_∞ controller designed in this thesis project requires the linearization of the model. A loss in terms of performance will be witnessed when the non-linearities become important (*e.g.* when the relative angular velocities between the two satellites is not exactly equal to zero or when the joints are in motion, with angular velocities different than zero). To keep a good control efficiency during the whole maneuver, a gain-scheduling strategy could be set up [4]. It consists in linearizing the model at different system configurations and designing a controller for each of these linearized models. Then, the space of possible configurations to reach the final state from the initial one is mapped in several regions. Every time the state of the controller goes from one region to another one, the controller used changes for one which has been designed using a linearization point within this region. A look-up table could be developed corresponding to the different regions. In terms of computational power required on-board, the difference is not significantly larger. However, the space to store the controllers' structure would be larger but still low since it remains a few state-space matrices only.

One of the main topics which could be further researched in parallel with the design of the controller is the path-planning during the reach phase. The path-planning assumptions taken in this thesis (Section 5.3) were very strong and resulted in a very straightforward path-planning. These assumptions were made since developing a path-planner was not directly required by the topic of the thesis. However, a minimum of path-planning instructions were needed in order to make the maneuver possible. In a later stage of the research, studying path-planning of the reach phase may be important and add constraints during the design of the controller. For instance, the real workspace of the robotic arm could be considered. The robotic singularities as well as the impossibility for the KUKA LWR to be fully stowed because of its architecture would then be considered. Even though it is computed as being within the workspace as describe in Fig. 4.1, it may not be within the real workspace of the arm. For the cases and with the system simplifications studied in this thesis, this simplified workspace had a small influence only but in a real mission, it may be different and jeopardize the mission. Moreover, obstacle avoidance is part of the path-planning too. Without it being investigated and implemented, the combined control strategy will never make it to a real space mission. Thus, some work will be needed in the future to make it happen. According to [1], the reach phase is followed by the form closure phase during which the gripper closes on the grasping area such that the target cannot escape. For the reach phase to be complete, the start conditions of the form closure have to be met. It means that the handle shall be within the mouth of the gripper at the end of the reach phase. Since the joint commanding the orientation of the gripper was not considered in this thesis, this condition could not be checked. Indeed, the gripper was included in the model but its orientation was not studied since it was assumed as fixed with the last link of the arm. Also the way the handle gets into the mouth of the gripper was not studied but it is absolutely crucial. Indeed, if the gripper is like a hand as implemented now on the HW-in-the-loop simulator (Fig. 1.3), the handle should be gripped correctly. Otherwise, the fingers will collide with the handle (left side of Fig. 7.1). One way to do it would be to constrain the way the gripper gets to the handle. From a certain distance (greater than the length of the fingers) until the end of the reach phase, the gripper shall approach the handle in a straight line with the z-axis of the last joint going through the handle. This strategy is shown on the right side of Fig. 7.1.

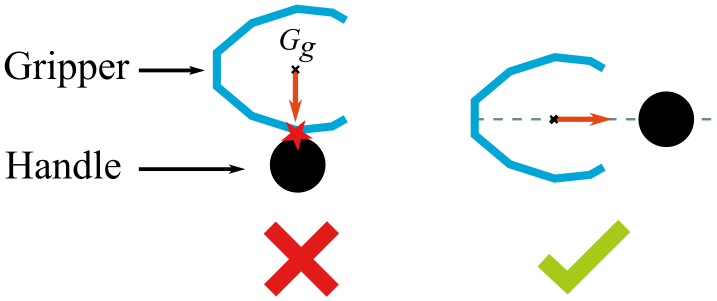


Figure 7.1: Path-planning at the end of the reach phase

Another possible improvement for path-planning is to consider a target satellite which is tumbling. In that case, the chaser will have to synchronize with the target's motion. Even though the synchronization phase is completed before the reach phase, the fact that the two spacecraft would be tumbling would be interesting to study. Indeed, the derivation of the model was completed in the LVLH centered at the target CoM so the new model may be different if the chaser is tumbling at the same rate as the target satellite. One way to handle it, as considered in the simulator at DLR, would be to express everything in the relative reference frame. In this reference frame, as explained in Section 2.3.1, only the difference between the states of the chaser with respect to the target are measured and controlled. No exterior reference frame is needed. In this reference frame, the chaser's base, once synchronized, will be fixed. However, this reference frame is rotating with respect to the LVLH pseudo-inertial reference frame. It means that it is not an inertial reference frame. Thus, the derivation of the model may be more complex. But taking that step is important since it is not realistic to consider a target satellite exactly still at a constant attitude in the LVLH reference frame. The scenario considered in this thesis is a simplistic one which shall be made more complex in order to get closer to reality.

Finally, more specifically in terms of implementation in the simulator at DLR, two main improvements can be done. The first one would be to implement the downlink as TMs of data instead of using the logs once the maneuver is completed. The second amelioration would be to test the controller with the HW-in-the-loop facility developed at DLR, called the OOS-SIM. Further optimization of the controller and some assumptions shall be released in order to meet with the requirements of the testing facility.

A

Comparison between the Symbolic and the Numerical Linearized Model

As explained in Section 4.3, a Simscape modelization was used to verify that the derivation of the system described in [45] was correct. In this Appendix, the comparison between the two models will be further discussed and detailed. The matrices found using the symbolic derivation are noted with a circle (e.g. $\mathbf{B}^{\circ}$) as a superscript while a triangle (e.g. $\mathbf{B}^{\triangle}$) is used for the Simscape derivation.

A.1. Description of the Model on Simscape

The final model of the system on Simscape is presented in Fig. A.1. This figure presents all parts considered in the modelling of the plant. The satellite body is modelled in space thanks to a six DoF joint. The first link of the arm is rigidly fixed on it. Then, a chain of three revolute joints and three links is implemented. Finally, the gripper is fixed on the last link ('Link 4') in order to have the right mechanical structure in the model, even though the gripper will not be used. Moreover, it is fixed on the last link since it is assumed that the last link is fixed with an angle equal to zero. In the viewer however, the gripper is not shown in order to have a clear visibility on the alignment between the end-effector and the handle. It means that the z-axis of the last joint (in green) goes through the circular ring attached to the target on which the handle is. The sensors and actuators are also shown, as well as the input u_u and the output y_u. In the initial conditions blocks (IC), the initial position and velocity of each DoF are given. The 'Interpreted MATLAB Fcn' is the block used to view the configuration of the two satellites as shown Fig. 4.9.
In Section A.2, a simplified version of this model, where the only body considered is the satellite is made while the complete one is used for the comparison of the nine DoF system models.

A.2. Comparison for a Simplified System, with Six Degrees of Dreedom

In this case, the robotic arm is not considered. The system consists in the spacecraft base only, with its six DoF. The state vector $\mathbf{x} = \begin{bmatrix}\mathbf{q} & \dot{\mathbf{q}}\end{bmatrix}^T$, which is in this case a 12×1 vector, is chosen as:

$$\mathbf{x} = \begin{bmatrix}\mathbf{0}_{1\times3} & 0 & 0 & \pi & \mathbf{0}_{1\times3} & \mathbf{1}_{1\times3}\end{bmatrix} \tag{A.1}$$

In this case, the $\mathbf{B}$ matrices of both the symbolic and the numerical models are equal:

$$\mathbf{B}^{\circ} = \mathbf{B}^{\triangle} = 10^{-3}\begin{bmatrix}\mathbf{0}_{3\times3} & \mathbf{0}_{3\times3}\\ 0.6667\mathbf{I}_3 & \mathbf{0}_{1\times3}\\ \mathbf{0}_{3\times3} & \mathbf{I}_3\end{bmatrix} \tag{A.2}$$

If a closer look is given to the $\mathbf{A}$ matrix, the bottom-left block for the two models are:

$$\mathbf{A}_{22}^{\circ} = \begin{bmatrix}\mathbf{0}_{3\times3} & \mathbf{0}_{3\times3}\\ \mathbf{0}_{3\times3} & \mathbf{a}^{\circ}\end{bmatrix} \text{ and } \mathbf{A}_{22}^{\triangle} = \begin{bmatrix}\mathbf{0}_{3\times3} & \mathbf{0}_{3\times3}\\ \mathbf{0}_{3\times3} & \mathbf{a}^{\triangle}\end{bmatrix} \tag{A.3}$$

where

$$\mathbf{a}^{\circ} = \begin{bmatrix} 0 & -0.3 & 0 \\ 0 & 0 & 0.1 \\ 0.2 & -0.2 & 0 \end{bmatrix} \text{ and } \mathbf{a}^{\triangle} = \begin{bmatrix} 0 & -0.3 & -0.2 \\ 0.3 & 0 & 0.1 \\ -0.2 & -0.1 & 0 \end{bmatrix} \tag{A.4}$$

Based on the argument developed in 4.3.1 according to which there are many ways of obtaining the matrix $\mathbf{C}_3$ in the equations of motion Eq. 4.40. Looking at the expression of the $\mathbf{C}_{rot}$ matrix given in Eq. 4.39 and at the two matrices in the previous equation. One realizes that in the symbolic model, the choice was to keep the coefficient on one component only while it is split on two different ones in the numerical modelling process. In terms of equation, the result for the symbolic model is then translated as:

$$\begin{aligned} \mathbf{C}^{\circ}_{rot} &= \begin{bmatrix} J\dot{\psi}\dot{\theta}\cos(\theta) \\ -J\dot{\phi}\dot{\psi}\cos(\theta) \\ J\dot{\phi}\dot{\theta}\cos(\theta) \end{bmatrix} \\ &= \begin{bmatrix} 0 & J\dot{\psi}\cos(\theta) & 0 \\ 0 & 0 & -J\dot{\phi}\cos(\theta) \\ -J\dot{\theta}\cos(\theta) & 2J\dot{\phi}\cos(\theta) & 0 \end{bmatrix} \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} \end{aligned} \tag{A.5}$$

$$\tag{A.6}$$

For the numerical model, it is known that the mass matrix is simply:

$$\mathbf{H} = 1000\mathbf{I}_3 \tag{A.7}$$

From there and based on the expression of the $\mathbf{A}$ matrix given in Eq. 4.61, the expression of $\mathbf{C}_{3_0}$ can be deduced:

$$\begin{aligned} \mathbf{C}_{3_0} &= -\begin{bmatrix} 1000 & 0 & 0 \\ 0 & 1000 & 0 \\ 0 & 0 & 1000 \end{bmatrix} \begin{bmatrix} 0 & -0.3 & -0.2 \\ 0.3 & 0 & 0.1 \\ -0.2 & -0.1 & 0 \end{bmatrix} \\ &= \begin{bmatrix} 0 & 300 & 200 \\ -300 & 0 & -100 \\ 200 & 100 & 0 \end{bmatrix} \end{aligned} \tag{A.8}$$

The following is then derived from the previous equation:

$$\mathbf{C}^{\triangle}_{rot} = \begin{bmatrix} 0 & J\dot{\psi}\cos(\theta) & J\dot{\theta}\cos(\theta) \\ -J\dot{\psi}\cos(\theta) & 0 & -J\dot{\phi}\cos(\theta) \\ J\dot{\theta}\cos(\theta) & J\dot{\phi}\cos(\theta) & 0 \end{bmatrix} \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} \tag{A.9}$$

It is clear that $\mathbf{C}^{\triangle}_{rot} = 2\mathbf{C}^{\circ}_{rot}$. The symbolic has been verified using Newton's equations as well as a second method based on Lagrangian mechanics. Thus, the Simscape is the only method implemented during this thesis which gives different results for the simplified case of a 6 DoF system. Because the equations of motion or the way the state-space matrices are computed is not fully given to the operator on Simscape, it is difficult to understand these differences.

A.3. Comparison for the Complete System, with Nine Degrees of Freedom

As given in Section 4.3.2, the $\mathbf{A}$, $\mathbf{C}$ and $\mathbf{D}$ matrix are exactly the same when the models are linearized around the state vector $\mathbf{q}_0$. Now, the $\mathbf{B}$ matrices found symbolically and using Simscape are very close to each other. The bottom part of both are given below.

$$\mathbf{B}_2^{\circ} = \begin{bmatrix} 0.0007 & 0 & 0 & 0 & 0 & 0 & 0 & -0.0004 & -0.0008 \\ 0 & 0.0007 & 0 & 0 & 0 & 0 & 0.0005 & 0 & -0 \\ 0 & 0 & 0.0007 & 0 & 0 & 0 & 0 & -0.0006 & -0.0004 \\ 0 & 0 & 0 & 0.0010 & 0 & 0 & -0.0009 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.0010 & 0 & 0 & -0.0027 & -0.0021 \\ 0 & 0 & 0 & 0 & 0 & 0.0010 & -0.0018 & 0 & 0 \\ 0 & 0.0005 & 0 & -0.0009 & 0 & -0.0018 & 0.0601 & -0.0003 & -0.0008 \\ -0.0004 & 0 & -0.0006 & 0 & -0.0027 & 0 & -0.0003 & 0.1206 & 0.1493 \\ -0.0008 & 0 & -0.0004 & 0 & -0.0021 & 0 & -0.0008 & 0.1493 & 0.3360 \end{bmatrix} \tag{A.10}$$

$$\mathbf{B}_2^{\triangle} = \begin{bmatrix} 0.0007 & 0 & 0 & 0 & 0 & 0 & 0 & -0.0004 & -0.0008 \\ 0 & 0.0007 & 0 & 0 & 0 & 0 & 0.0005 & 0 & 0 \\ 0 & 0 & 0.0007 & 0 & 0 & 0 & 0 & -0.0006 & -0.0004 \\ 0 & 0 & 0 & 0.0010 & 0 & 0 & -0.0009 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.0010 & 0 & 0 & -0.0027 & -0.0021 \\ 0 & 0 & 0 & 0 & 0 & 0.0010 & -0.0018 & 0 & 0 \\ 0 & 0.0005 & 0 & -0.0009 & 0 & -0.0018 & 0.0615 & -0.0003 & -0.0009 \\ -0.0004 & 0 & -0.0006 & 0 & -0.0027 & 0 & -0.0003 & 0.1231 & 0.1522 \\ -0.0008 & 0 & -0.0004 & 0 & -0.0021 & 0 & -0.0009 & 0.1522 & 0.3422 \end{bmatrix} \tag{A.11}$$

The minor differences which can be seen when comparing the two previous previous matrices have a very slight influence on the step response of both linear models. The figure below (Fig. A.2) shows the differences between the two step responses. Thus, for the linearization configuration considered, the two models are very close. Because the derivation of the matrices is not detailed nor accessible in Simscape, the origin of the slight differences is difficult to determine.

That being said, it is interesting to look at different configurations for example when the linearization velocities are not equal to zero. In that case, there are important differences between the two models. Indeed, as said in Section A.2, the way the **C** matrix is determined in the numerical model is different than in the symbolic one. If the velocities are taken as:

$$\begin{bmatrix} \dot{\phi} & \dot{\theta} & \dot{\psi} & \dot{\theta}_1 & \dot{\theta}_2 & \dot{\theta}_3 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 0 \end{bmatrix} \tag{A.12}$$

The bottom right part of the **A** matrices are very different while the **B** matrices are the same for both the numerical and the symbolic models. However, these two models should be equal at the linearization configuration. To check this point, one can test if:

$$\begin{bmatrix} \mathbf{A}_{21}^{\circ} & \mathbf{A}_{22}^{\circ} \end{bmatrix} \mathbf{x} = \begin{bmatrix} \mathbf{A}_{21}^{\triangle} & \mathbf{A}_{22}^{\triangle} \end{bmatrix} \mathbf{x} \tag{A.13}$$

where

$$\mathbf{A}^{\circ} = \begin{bmatrix} \mathbf{A}_{11}^{\circ} & \mathbf{A}_{12}^{\circ} \\ \mathbf{A}_{21}^{\circ} & \mathbf{A}_{22}^{\circ} \end{bmatrix} \text{ and } \mathbf{A}^{\triangle} = \begin{bmatrix} \mathbf{A}_{11}^{\triangle} & \mathbf{A}_{12}^{\triangle} \\ \mathbf{A}_{21}^{\triangle} & \mathbf{A}_{22}^{\triangle} \end{bmatrix} \tag{A.14}$$

It was shown in Section 4.3.3 that for the symbolic model, $\mathbf{A}_{21}^{\circ}$ is empty. However, in the numerical model given by Simscape, the bottom-left block in $\mathbf{A}^{\triangle}$ is not empty. This difference could not be understood during this thesis. Moreover, because the models are equal for the linearization configuration chosen, this difference was not thoroughly investigated since it was found that the way the state-space matrices are derived with Simscape is not straightforward.

Finally, if the bottom-right part of the matrix only is considered $\mathbf{A}_{22}^{\triangle}$ and multiplied by $\dot{\mathbf{q}}$, the result from the numerical model is about twice as big as the result using the symbolic method. The same conclusion was drawn from the results found in Section A.2.

To conclude, the results found in the specific case used in this thesis and derived symbolically are coherent with the results from Simscape. However, in some cases which may be interesting in future works, the dynamics model could not be fully validated with the numerical results. However, the symbolic model was verified based on two other methods, the derivation of the Lagrangian of the system and the model derived from Newton's equations in the case of a system with six DoF only. In the Lagrangian method, which considers the full system with nine DoF, the results obtained were similar to the ones found based on the paper written by Papadopoulos and Moosavian [45]. Furthermore, for the simplified system with six DoF only, the $\mathbf{C}_{3_0}$ and $\mathbf{H}_0$ matrices in the equations of motion (Eq. 4.58) were shown to be symbolically equal, without any error of factor two. These issues should be further investigated to understand where the modelling difference appear.

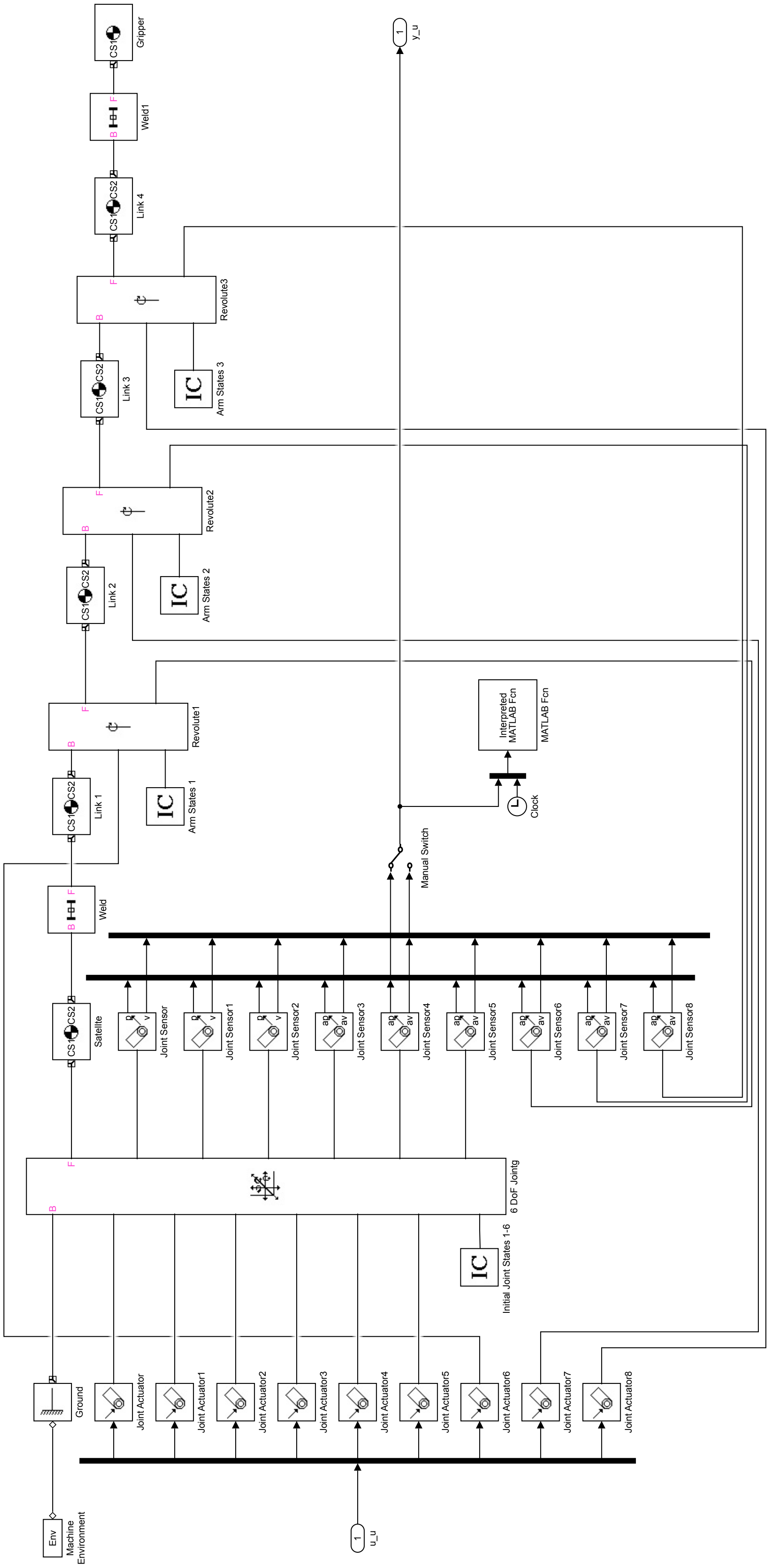


Figure A.1: Model of the plant in Simscape

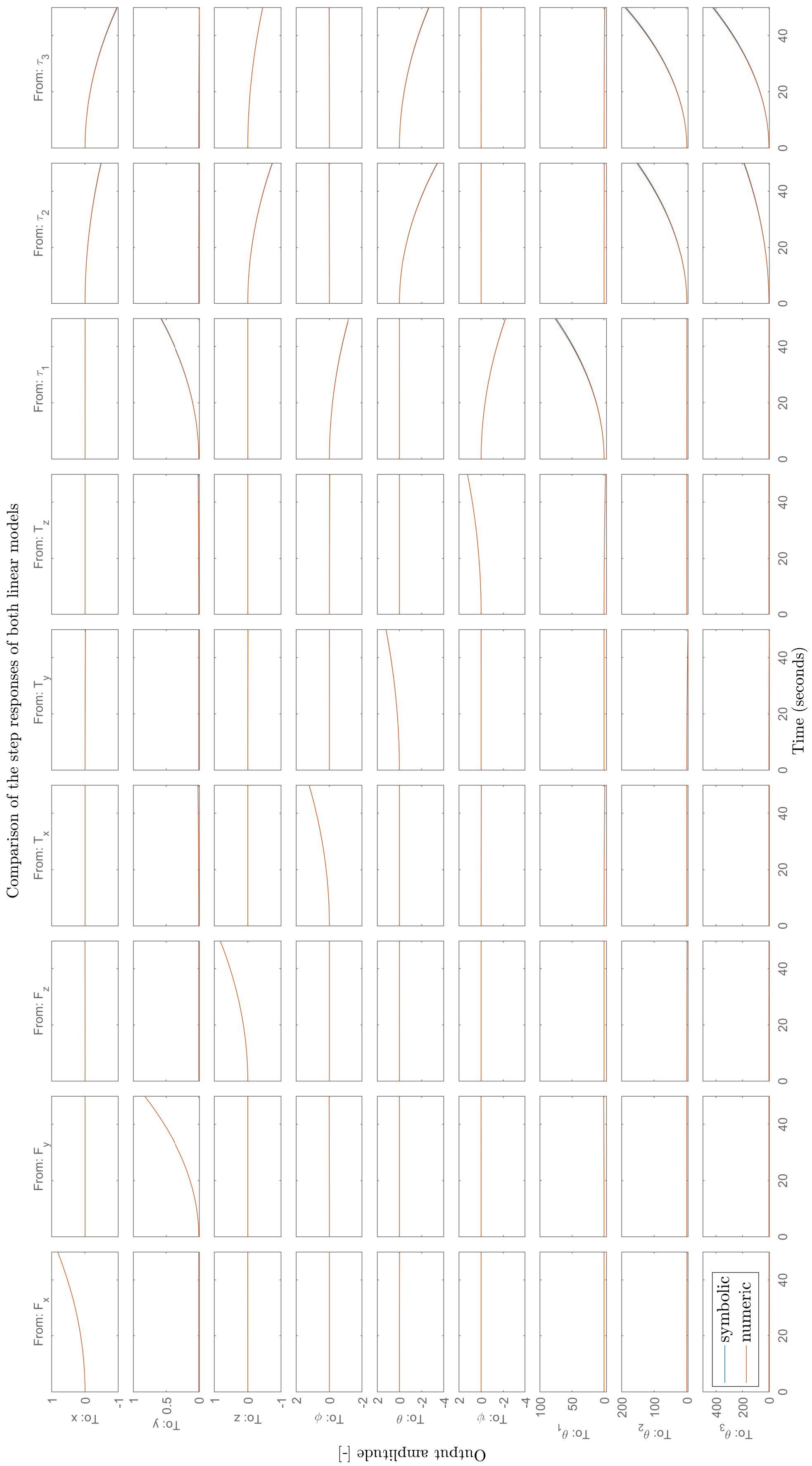


Figure A.2: Comparison of the step response using the symbolic and the numerical models

B

hinfsyn Matlab Function - Investigation

As discussed in Section 5.4.4, the model of the system cannot be used as it is to design an H_∞ controller because no matter what the filters used in the loop-shaping procedure, the following condition cannot be completed:

$$\begin{bmatrix} \mathbf{A} - j\omega\mathbf{I} & \mathbf{B}_1 \\ \mathbf{C}_2 & \mathbf{D}_{21} \end{bmatrix} \text{has full row rank for all } \omega \in \mathbb{R} \tag{B.1}$$

where $\mathbf{A}$, $\mathbf{B}_1$, $\mathbf{C}_2$ and $\mathbf{D}_{22}$ are block matrices of the augmented plant.

B.1. Investigation and Workaround

The expression of these matrices can be derived but first, the different blocks of the augmented plant are defined:

- $\mathbf{G} = \begin{bmatrix} \mathbf{A}_G & \mathbf{B}_G \\ \mathbf{C}_G & \mathbf{D}_G \end{bmatrix}$
- $\mathbf{W_1} = \begin{bmatrix} \mathbf{A}_{W_1} & \mathbf{B}_{W_1} \\ \mathbf{C}_{W_1} & \mathbf{D}_{W_1} \end{bmatrix}$
- $\mathbf{W_2} = \begin{bmatrix} \mathbf{A}_{W_2} & \mathbf{B}_{W_2} \\ \mathbf{C}_{W_2} & \mathbf{D}_{W_2} \end{bmatrix}$
- $\mathbf{W_3} = \begin{bmatrix} \mathbf{A}_{W_3} & \mathbf{B}_{W_3} \\ \mathbf{C}_{W_3} & \mathbf{D}_{W_3} \end{bmatrix}$

When the plant is augmented, one obtains the *new* plant $\mathbf{P}$, called the augmented plant. The state-space matrices of this plant can be determined based on its expression:

$$\mathbf{P} = \begin{bmatrix} \mathbf{A}_P & \mathbf{B}_P \\ \mathbf{C}_P & \mathbf{D}_P \end{bmatrix} \tag{B.2}$$

where

$$\mathbf{A}_P = \begin{bmatrix} \mathbf{A}_{W_1} & \mathbf{0} & \mathbf{0} & -\mathbf{B}_{W_1}\mathbf{C}_G \\ \mathbf{0} & \mathbf{A}_{W_2} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{A}_{W_3} & \mathbf{B}_{W_3}\mathbf{C}_G \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{A}_G \end{bmatrix} \quad \mathbf{B}_P = \begin{bmatrix} \mathbf{B}_{W_1} & \mathbf{0} \\ \mathbf{0} & \mathbf{B}_{W_2} \\ \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{B}_G \end{bmatrix}$$

$$\mathbf{C}_P = \begin{bmatrix} \mathbf{C}_{W_1} & \mathbf{0} & \mathbf{0} & -\mathbf{D}_{W_1}\mathbf{C}_G \\ \mathbf{0} & \mathbf{C}_{W_2} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{C}_{W_3} & \mathbf{D}_{W_3}\mathbf{C}_G \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & -\mathbf{C}_G \end{bmatrix} \quad \mathbf{D}_P = \begin{bmatrix} \mathbf{D}_{W_1} & \mathbf{0} \\ \mathbf{0} & \mathbf{D}_{W_1} \\ \mathbf{0} & \mathbf{0} \\ \mathbf{I}_9 & \mathbf{0} \end{bmatrix}$$

These matrices are derived from the results of *augw(*$\mathbf{G}$,$\mathbf{W}_1$,$\mathbf{W}_2$,$\mathbf{W}_3$*)*. Of course, if the plant is different (*e.g.* if its $\mathbf{D}$ matrix is not empty), the matrices of $\mathbf{P}$ may be more general. For instance, if $\mathbf{D}_G$ is the identity matrix, the blocks in the right column of matrix $\mathbf{D}_P$ are not empty. Also, the $\mathbf{B}_P$ matrix gets additional terms. The matrix which shall have full row rank can be rewritten using the block matrices of the state-space matrices of $\mathbf{P}$. Indeed, for $s = 0$, Eq. B.1 becomes:

$$\begin{bmatrix} \mathbf{A} & \mathbf{B}_1 \\ \mathbf{C}_2 & \mathbf{D}_{21} \end{bmatrix} \text{where} \begin{cases} \mathbf{A} = \mathbf{A}_P \text{ is a } 45 \times 45 \text{ matrix} \\ \mathbf{B}_1 = \begin{bmatrix} \mathbf{B}_{W_1} & \mathbf{0}_{36\times9} \end{bmatrix}^T \text{ is a } 45 \times 9 \text{ matrix} \\ \mathbf{C}_2 = \begin{bmatrix} \mathbf{0}_{9\times36} & -\mathbf{C}_G \end{bmatrix} \text{ is a } 9 \times 45 \text{ matrix} \\ \mathbf{D}_{21} = \mathbf{I}_9 \text{ is a } 9 \times 9 \text{ matrix} \end{cases}$$

This matrix can be symbolically written in order to get a visual interpretation of it. It is given below in Eq. B.3. Every $\mathbf{0}$ is a 9×9 empty block. Every block with a diagonal line is a 9×9 with all the extra-diagonal terms equal to zero.

$$\begin{bmatrix} \mathbf{A} & \mathbf{B}_1 \\ \mathbf{C}_2 & \mathbf{D}_{21} \end{bmatrix} = \begin{bmatrix} \backslash & \mathbf{0} & \mathbf{0} & \backslash & \mathbf{0} & \backslash \\ \mathbf{0} & \backslash & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \backslash & \backslash & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \backslash & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \backslash & \mathbf{0} & \backslash \end{bmatrix} \tag{B.3}$$

Because the last nine rows of the $\mathbf{A}_G$ matrix are empty, the rank of this matrix cannot be full, no matter the choice of the weighting function $\mathbf{W}_i$ with $i \in \{1, 2, 3\}$. In order to be able to compute an H_∞ controller of the plant, an identity matrix with very low multiplicative coefficient is added in the bottom-left of it. the difference between the modified plant and the initial one is investigated in Section 5.4.4 and hereafter.

B.2. Comparison of the Modified Plant with the Initial Plant

As discussed in Section 5.4.4, in order to be able to find an H_∞ controller for the linear model of the system, an identity matrix, multiplied by a very small coefficient (10^{-9}) is added to the lower-left block matrix of the A matrix of the state-space representation of the system. The influence of this change was shown Fig. 5.7 for a more complex and general model of the plant than the one used in this thesis. Indeed, taking the linearization configuration with all velocities equal to zero reduces the complexity of the plant. In this Appendix, the impact of the addition of the identity matrix to the model used in the thesis is shown Fig. B.1. For the sake of clarity, only the magnitude are shown. The difference on the phase is not visible. However, on the magnitude plots, it is clear that for very small frequencies (lower than 10^{-4}Hz), the difference between the two plants is not negligible. Such low frequencies are not present in the problem investigated during this thesis. Thus, it is considered that the additional matrix has a negligible effect on the dynamics behavior of the system.

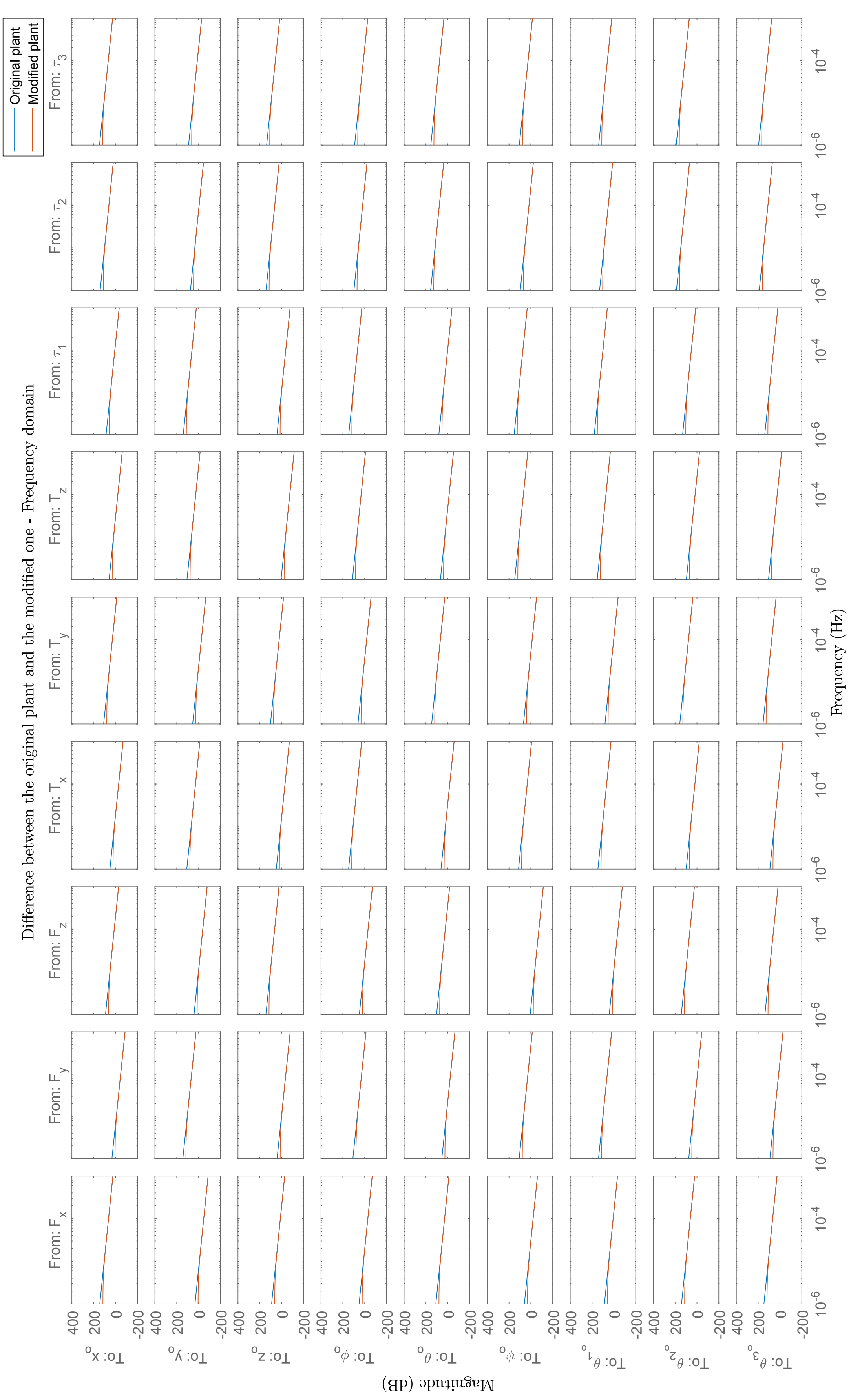


Figure B.1: Bode magnitude diagrams of all I/O couplings - Difference between the initial plant and the modified one

C

Step Response of all 81 Couplings for the Continuous Linear Controlled Loop

As explained in Section 5.4.4, the objective of the controller is to decouple all the states such that when a state has to be controlled, it can be without having too much influence on the other states. This way, the base could remain fixed in the target reference frame. It would increase the precision of the maneuver and may also keep the energy needed to bring the end-effector close to the handle low. Fig. C.1 shows that using the controller developed in Chapter 5, the step responses have an influence only on the corresponding states. In fact, only the diagonal terms reach the steady-state values of one while every extra-diagonal response converges to zero. The states are almost fully decoupled.

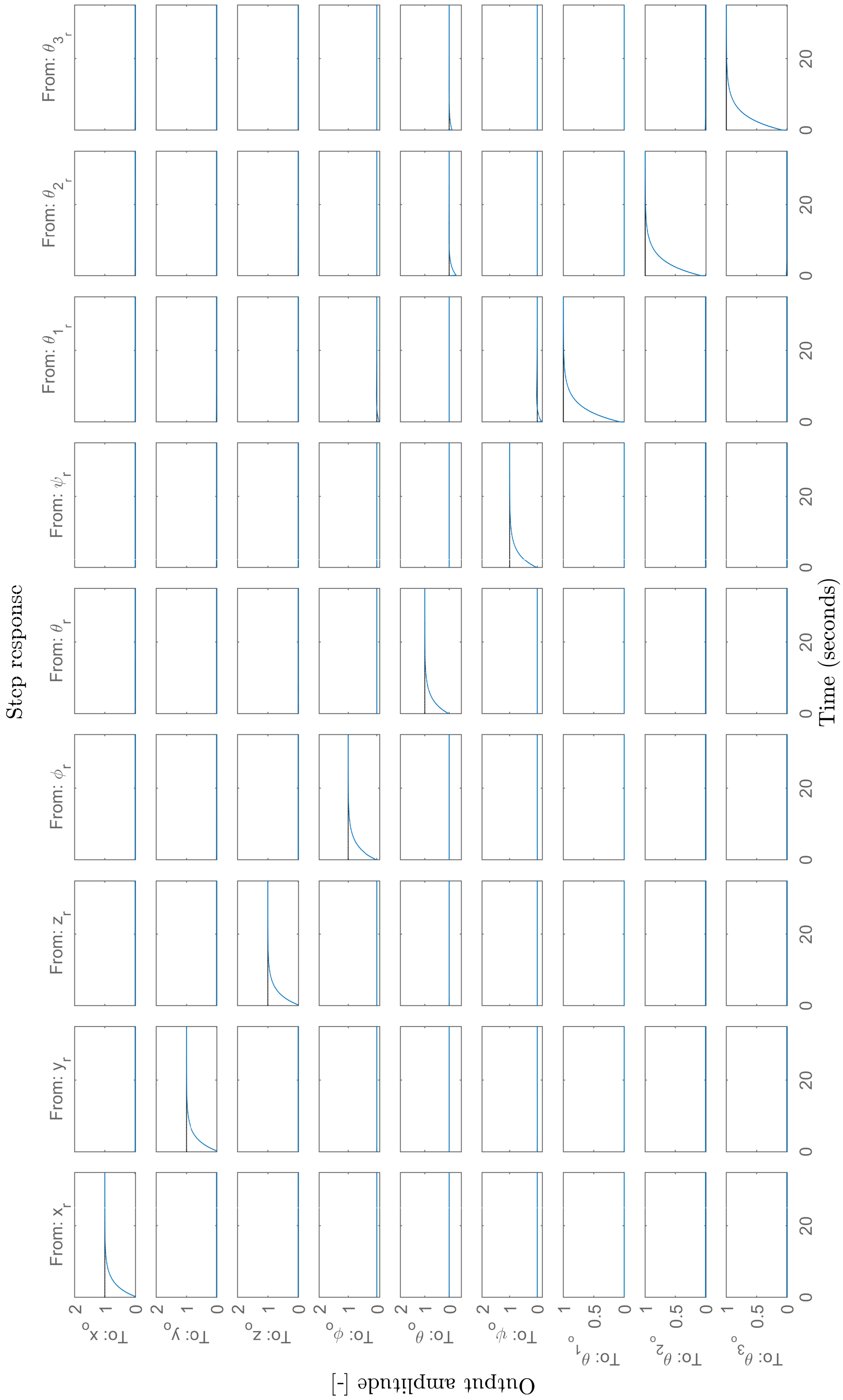


Figure C.1: Step response of the 81 input/output couplings - Continuous linear closed-loop

D

Additional Simulation Results

As mentioned in Section 6.2, some simulation results were not shown in the report but reported in Table 6.1. These additional results are given hereafter.

Additional tests with the linear plant

First of all, two extra simulations are completed on the linear simulator. The first one (Fig. D.1) shows that the continuous controller without noise gives good results, as expected since it also gives good results with noise (Fig. 6.1). The second one is a test without noise of the discretized controller at 100Hz. The results are shown Fig. D.2). Once again, since the simulation gave good results with noise Fig. 6.7, it was expected that the results with noise would be satisfying.

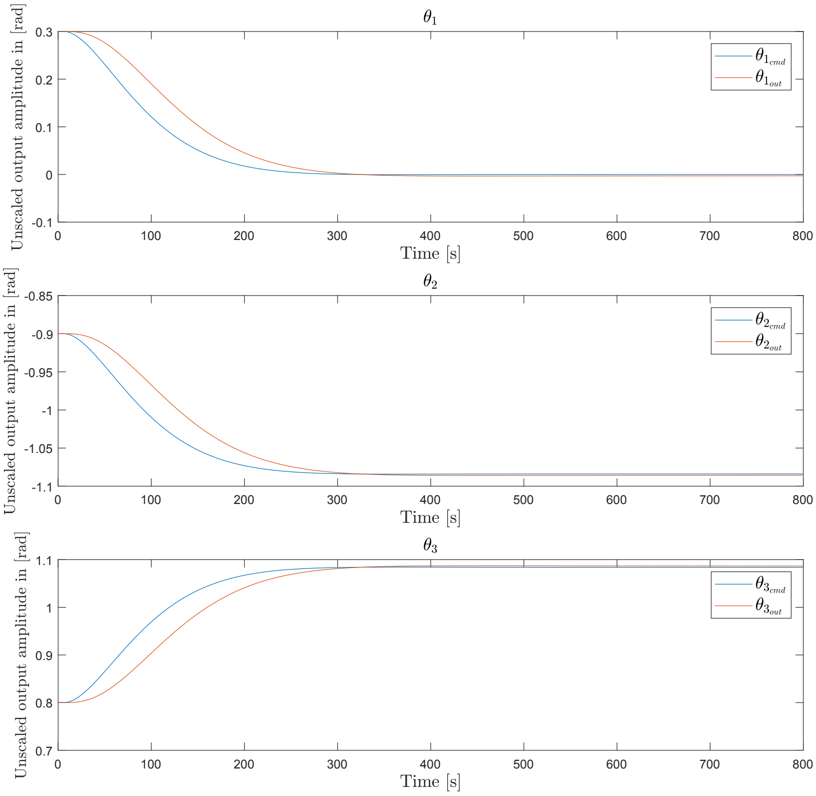


Figure D.1: Simulation with the continuous controller implemented in the linear simulator - without noise

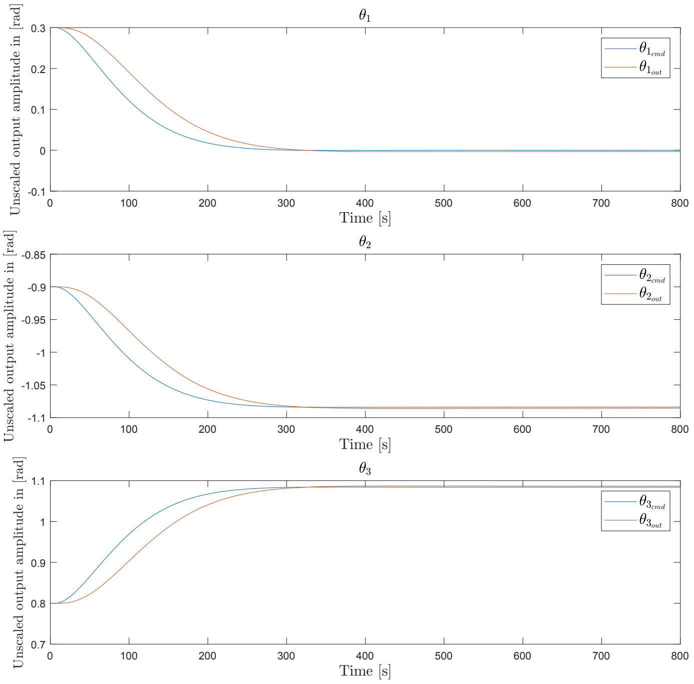


Figure D.2: Simulation with the discretized controller at $100Hz$ implemented in the linear simulator - without noise

Testing of the continuous controller in the non-linear simulator

Second of all, the continuous controller is implemented in the loop with the non-linear plant without noise. Fig. D.3 shows that the states reach the command values relatively quickly and the steady-state error vector is very small. The control of the outputs to reach the command is well and efficiently done.

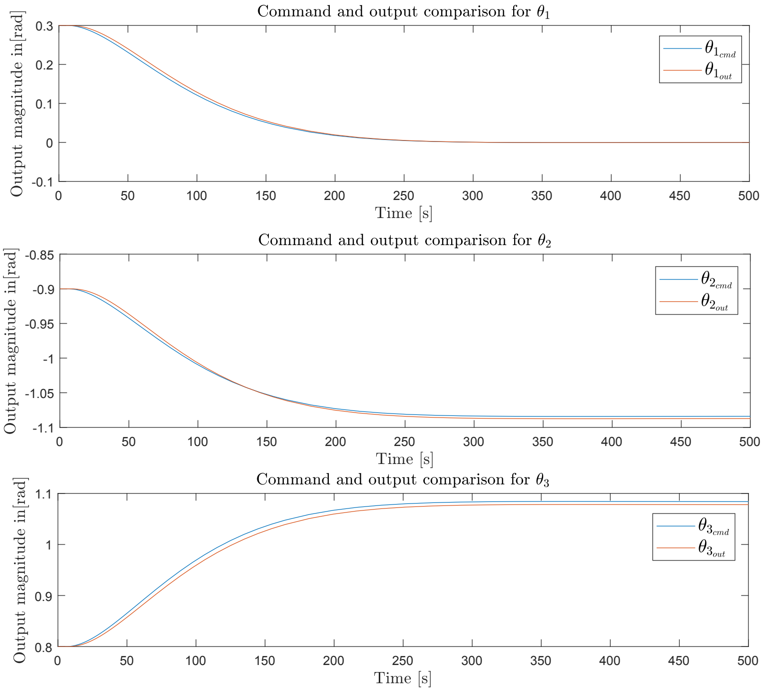
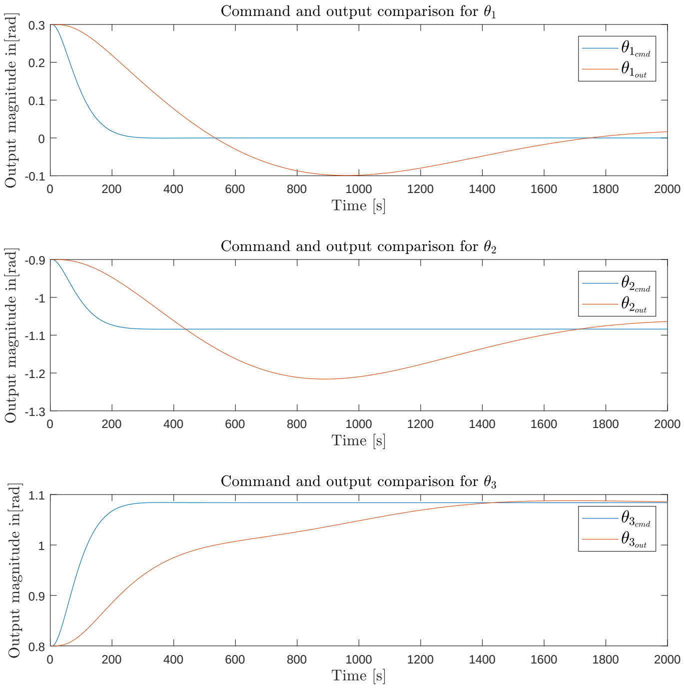


Figure D.3: Simulation with the continuous controller implemented in the non-linear simulator - without noise

Tests with the controller discretized at 10Hz in the non-linear simulator

Moreover, the controller, discretized at 1Hz is implemented in the non-linear simulator without noise at first (Fig. D.4) and then with noise (Fig. D.5). As explained in the thesis, a sampling frequency of $10Hz$ was proven to be too low to obtain satisfying results. Indeed, the states oscillate with a low frequency beyond $2000s$. Moreover, the overshoot on the two first joint angles is important which is not acceptable during a mission requiring high precision and where a large overshoot is likely to imply a collision between the gripper and the target satellite. It is worth noticing that the influence of the noise in the non-linear case is very low compared to the case where the discrete controller was implemented in the linear system (Fig. 6.5 and Fig. 6.6).


Command and output comparison for θ_1

Command and output comparison for θ_2

Command and output comparison for θ_3

Figure D.4: Simulation with the discretized controller at $10Hz$ implemented in the non-linear simulator - without noise

In addition to the tests completed to cover the different cases given in Table 6.1. Some additional tests were conducted to better understand the performance, the robustness and the limitations of the controller designed in Chapter 5 in the non-linear simulator.

Investigation of the decoupling of the joints

The decoupling of the joints' states is also investigated. Fig. D.6 shows the influence of a motion of the first joint angle on the two others. Fig. D.7 demonstrates that the motion of the two last joints have a negligible influence on the first one. Finally, Fig. D.8 presents the effect of a motion of the second link on the two others.

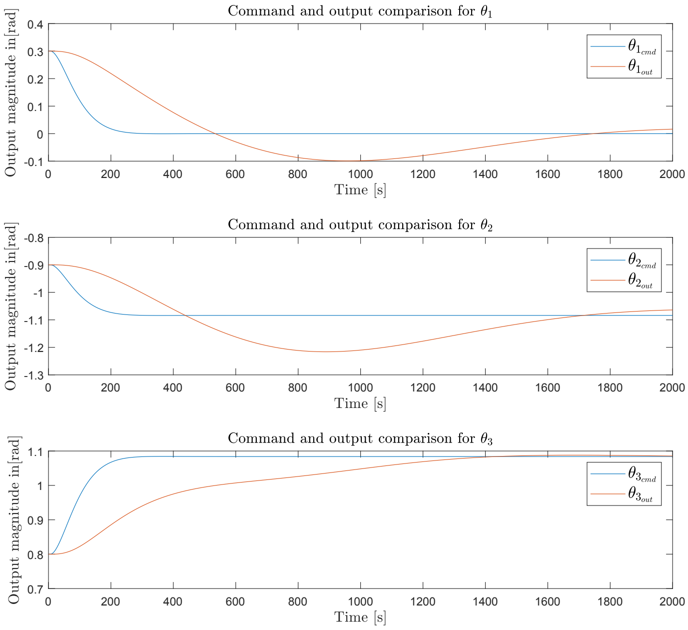


Figure D.5: Simulation with the discretized controller at $10Hz$ implemented in the non-linear simulator - with noise

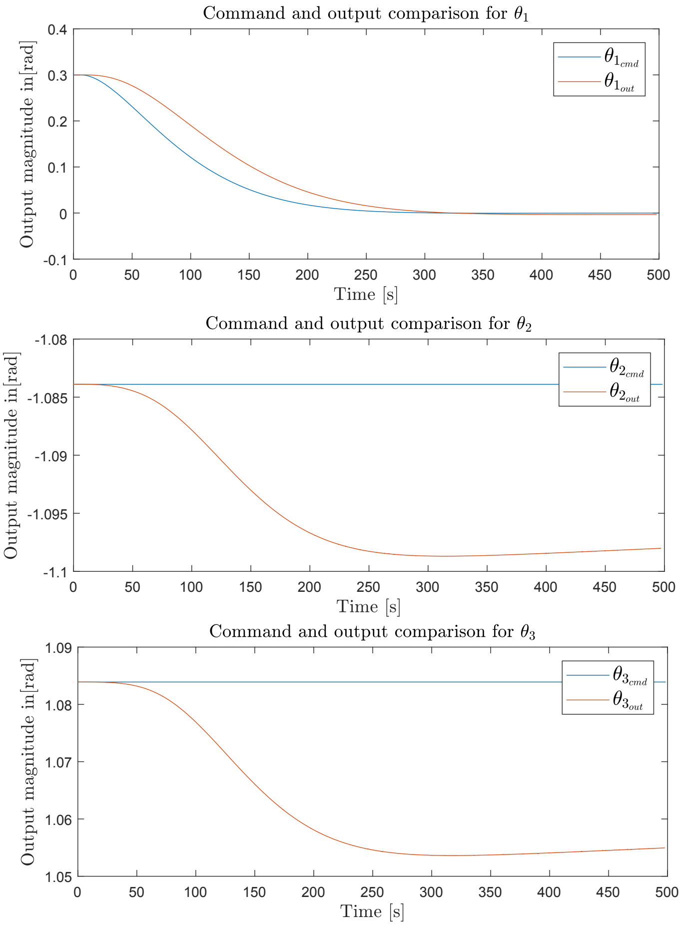
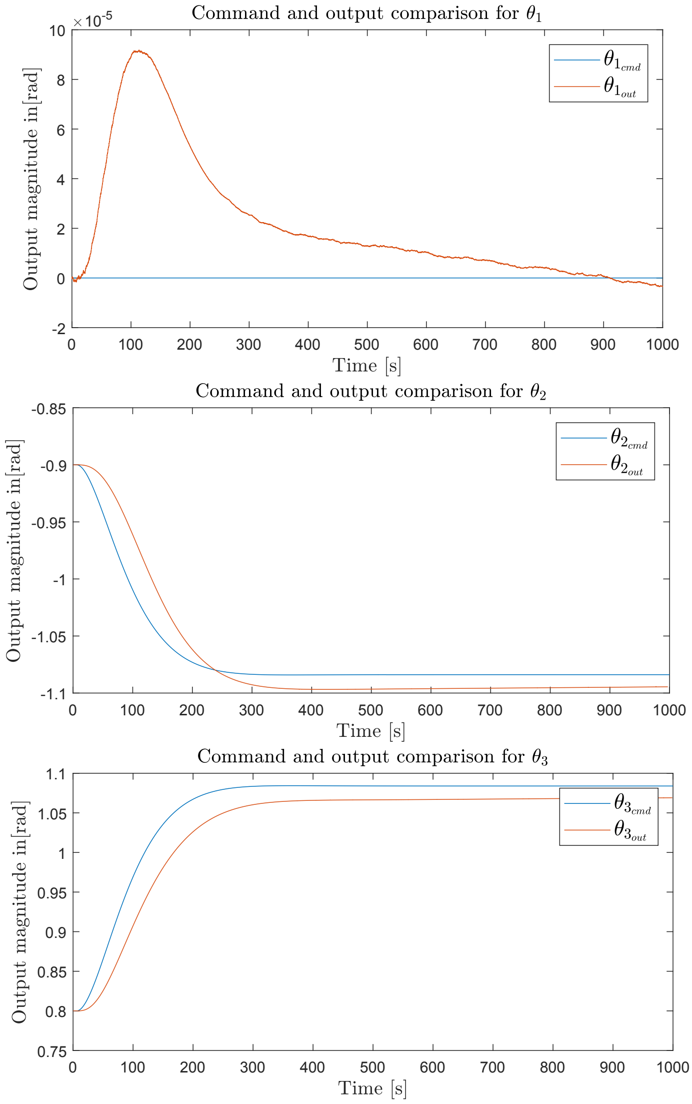


Figure D.6: Discretized controller at $100Hz$ in the loop with noise and the non-linear plant - First joint commanded only



Command and output comparison for θ_1

Command and output comparison for θ_2

Command and output comparison for θ_3

Figure D.7: Discretized controller at $100Hz$ in the loop with noise and the non-linear plant - Second and third joints commanded only

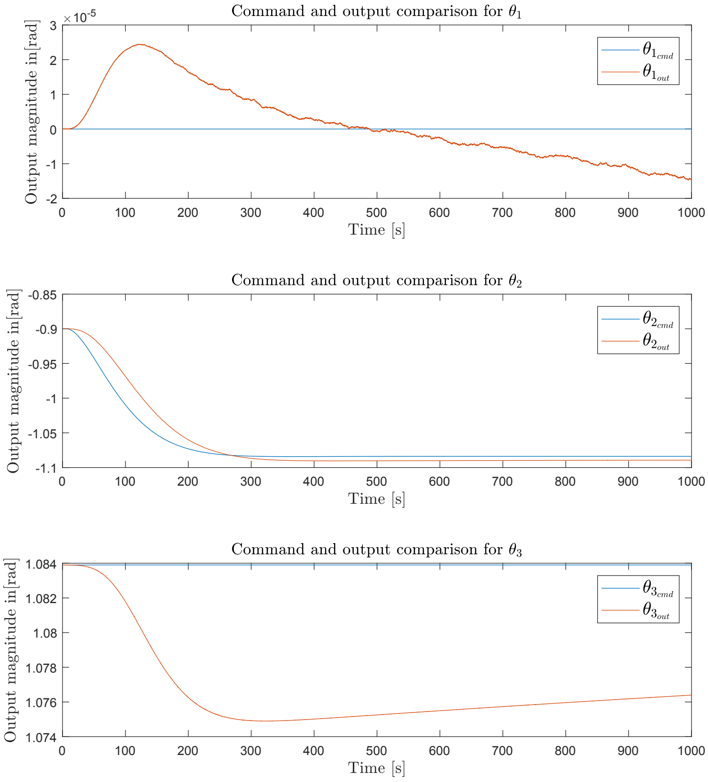


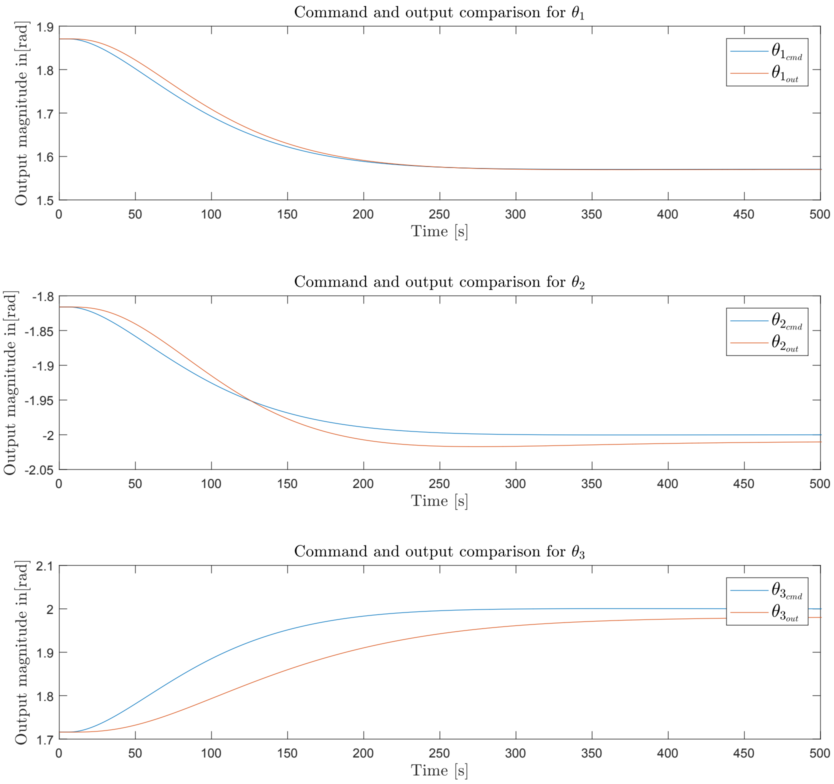
Figure D.8: Discretized controller at $100Hz$ in the loop with noise and the non-linear plant - Second joint commanded only

Investigation of the effects of non-linearities of the plant

As explained in Section6.2, different changes in configuration were tested with the same change in terms of amplitude on each joint to evaluate the impact of the non-linearities depending on how far from the linearization point the maneuver is completed. The central configuration is the one used previously, with $\mathbf{q}_0$ and $\mathbf{q}_f$ respectively used as initial and final configurations. The two other configurations are detailed in the Table D.1.

Table D.1: Effects of the non-linearities of the plant on the performance of the control

Test number and others	Initial configuration	Final *commanded* configuration	Final *real* configuration	Percentage of the error compared to the amplitude of the maneuver
Expression	$\mathbf{C}_i$	$\mathbf{C}_{f_c}$	$\mathbf{C}_{f_r}$	$\mathbf{\Delta} = \left\lvert \frac{\mathbf{C}_{f_r} - \mathbf{C}_{f_c}}{\mathbf{C}_{f_c} - \mathbf{C}_i} \right\rvert$
Unit	rad	rad	rad	%
1	$\begin{bmatrix} 1.8708 \\ -1.8161 \\ 1.7161 \end{bmatrix}$	$\begin{bmatrix} \pi/2 \\ -2 \\ 2 \end{bmatrix}$	$\begin{bmatrix} 1.5699 \\ -2.0104 \\ 1.98 \end{bmatrix}$	$\begin{bmatrix} 0.3 \\ 5.7 \\ 7 \end{bmatrix}$
2	$\begin{bmatrix} 0.3 \\ -0.9 \\ 0.8 \end{bmatrix}$	$\begin{bmatrix} 0 \\ -1.0839 \\ 1.0839 \end{bmatrix}$	$\begin{bmatrix} -0.003 \\ -1.108 \\ 1.046 \end{bmatrix}$	$\begin{bmatrix} 1 \\ 13.1 \\ 13.3 \end{bmatrix}$
3	$\begin{bmatrix} 1.3 \\ 1.6839 \\ 0.7161 \end{bmatrix}$	$\begin{bmatrix} 1 \\ 1.5 \\ 1 \end{bmatrix}$	$\begin{bmatrix} 0.996 \\ 1.484 \\ 0.966 \end{bmatrix}$	$\begin{bmatrix} 1.3 \\ 8.7 \\ 12 \end{bmatrix}$

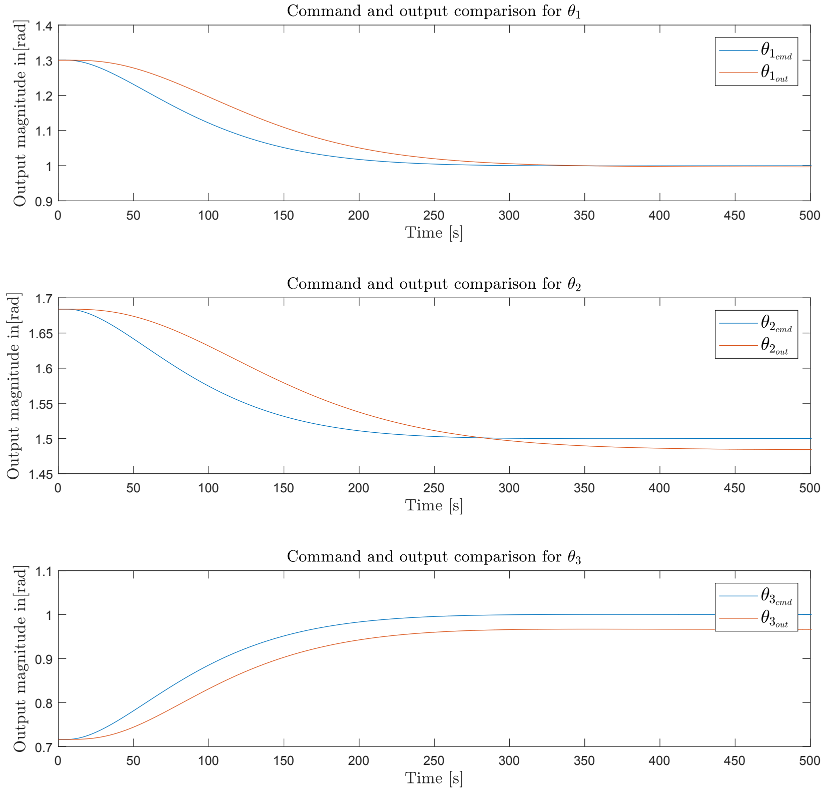


Command and output comparison for θ_1

Command and output comparison for θ_2

Command and output comparison for θ_3

Figure D.9: Results of test number 1



Command and output comparison for θ_1

Command and output comparison for θ_2

Command and output comparison for θ_3

Figure D.10: Results of test number 3

Robustness Investigation

For the sake of clarity and because all first joint angles' final value is within 0.01° from the nominal final value (which is zero), only the second and third joint angles are displayed in the Table D.2.

Table D.2: Results of the robustness further study

Case	Final $[\theta_2 \quad \theta_4]^T$ in [rad]	Final EE position $\mathbf{p}_{EE_f}$ in [m]	Difference on the EE position $\Delta\mathbf{p}_{EE} = \mathbf{p}_{EE_{Nominal}} - \mathbf{p}_{EE_f}$ in [m]
Nominal case	$\begin{bmatrix} -1.128 \\ 0.998 \end{bmatrix}$	$\begin{bmatrix} 0.431 \\ 0 \\ 1.107 \end{bmatrix}$	$\begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$
$1.1m_i$	$\begin{bmatrix} -1.132 \\ 0.989 \end{bmatrix}$	$\begin{bmatrix} 0.426 \\ 0 \\ 1.108 \end{bmatrix}$	$\begin{bmatrix} 0.005 \\ 0 \\ -0.001 \end{bmatrix}$
$1.3m_i$	$\begin{bmatrix} -1.141 \\ 0.973 \end{bmatrix}$	$\begin{bmatrix} 0.419 \\ 0 \\ 1.108 \end{bmatrix}$	$\begin{bmatrix} 0.012 \\ 0 \\ -0.001 \end{bmatrix}$
$1.1l_i$	$\begin{bmatrix} -1.137 \\ 0.98 \end{bmatrix}$	$\begin{bmatrix} 0.423 \\ 0 \\ 1.108 \end{bmatrix}$	$\begin{bmatrix} 0.008 \\ 0 \\ -0.001 \end{bmatrix}$
$1.3l_i$	$\begin{bmatrix} -1.156 \\ 0.94 \end{bmatrix}$	$\begin{bmatrix} 0.405 \\ 0 \\ 1.113 \end{bmatrix}$	$\begin{bmatrix} 0.0260 \\ 0 \\ -0.0060 \end{bmatrix}$
$1.1m_i$ and $1.1l_i$	$\begin{bmatrix} -1.142 \\ 0.971 \end{bmatrix}$	$\begin{bmatrix} 0.419 \\ 0 \\ 1.108 \end{bmatrix}$	$\begin{bmatrix} 0.012 \\ 0 \\ -0.001 \end{bmatrix}$
Feedback delay - one second	$\begin{bmatrix} -1.128 \\ 0.997 \end{bmatrix}$	$\begin{bmatrix} 0.43 \\ 0 \\ 1.108 \end{bmatrix}$	$\begin{bmatrix} 0.001 \\ 0 \\ -0.001 \end{bmatrix}$
Delay $\mathbf{K}/\mathbf{G}$ - 1 step	$\begin{bmatrix} -1.112 \\ 1.011 \end{bmatrix}$	$\begin{bmatrix} 0.434 \\ 0 \\ 1.121 \end{bmatrix}$	$\begin{bmatrix} -0.003 \\ 0 \\ -0.014 \end{bmatrix}$
Delay $\mathbf{K}/\mathbf{G}$ - 10 steps	$\begin{bmatrix} -1.1058 \\ 1.113 \end{bmatrix}$	$\begin{bmatrix} 0.48 \\ 0 \\ 1.116 \end{bmatrix}$	$\begin{bmatrix} -0.049 \\ 0 \\ -0.009 \end{bmatrix}$
Delay $\mathbf{K}/\mathbf{G}$ - 30 steps	$\begin{bmatrix} -0.964 \\ 1.285 \end{bmatrix}$	$\begin{bmatrix} 0.57 \\ 0 \\ 1.114 \end{bmatrix}$	$\begin{bmatrix} -0.139 \\ 0 \\ -0.007 \end{bmatrix}$
Actuators dynamics model 0.1 second	$\begin{bmatrix} -1.128 \\ 0.998 \end{bmatrix}$	$\begin{bmatrix} 0.431 \\ 0 \\ 1.107 \end{bmatrix}$	$\begin{bmatrix} 0 \\ 0 \\ 0 \end{bmatrix}$

Bibliography

[1] Statement of Work for the Control and Management of Robotics Active Debris removal (COMRADE). Technical report, 2015. URL `www.esa.int`.

[2] D. Alazard, C. Cumer, and K. Tantawi. Linear dynamic modeling of spacecraft with various flexible appendages. In *IFAC Proceedings Volumes (IFAC-PapersOnline)*, 2008.

[3] F. Amato and R. Iervolino. μ-synthesis for a small commercial aircraft: design and simulator validation. *Journal of guidance, control, and dynamics*, 27(3):479–490, 2004.

[4] P. Apkarian and R.J. Adams. *Advanced gain-scheduling techniques for uncertain systems*. SIAM, 2000.

[5] J. Artigas, M. De Stefano, W. Rackl, R. Lampariello, B. Brunner, W. Bertleff, R. Burger, O. Porges, A. Giordano, C. Borst, and A. Albu-Schaeffer. The OOS-SIM: An on-ground simulation facility for on-orbit servicing robotic operations. In *Proceedings - IEEE International Conference on Robotics and Automation*, 2015.

[6] G. Balas, R. Chiang, A. Packard, and M. Safonov. Robust control toolbox: Reference. *Mathworks, Natick, MA*, 2018.

[7] G.J. Balas. Linear, parameter-varying control and its application to a turbofan engine. *International Journal of Robust and Nonlinear Control*, 12(9):763–796, 2002.

[8] G. Bartolini, A. Ferrara, and E. Usai. Applications of a sub-optimal discontinuous control algorithm for uncertain second order systems. *International Journal of Robust and Nonlinear Control*, 7(4):299–319, 1997.

[9] H. Benninghoff, F. Rems, M. Gnat, R. Faller, R. Krenn, M. Stelzer, B. Brunner, and G. Panin. End-to-end simulation and verification of rendezvous and docking/berthing systems using robotics. In *5th International Workshop on Verification and Testing of Space Systems*, 2016.

[10] J. Bibel and H. Stalford. μ-synthesis autopilot design for a flexible missile. In *29th Aerospace Sciences Meeting*, page 586, 1991.

[11] J.E. Bibel and D.S. Malyevac. Guidelines for the selection of weighting functions for H_∞ control. *Naval Surface Warfare Center Dahlgren Div VA*, 1992.

[12] O.H. Bosgra and G. Kwakernaak, H.and Meinsma. Design methods for control systems. *Notes for a Course of the Dutch Institute of Systems and Control, Winter term*, 2002, 2001.

[13] B. Brunner, K. Zettel, B. Steinmetz, and G. Hirzinger. Tele-sensor programming - a task-directed programming approach for sensor-based space robots. 1995.

[14] R.Y. Chiang, M.G. Safonov, and J.A. Tekawy. H_∞ flight control design with large parametric robustness. In *American Control Conference, 1990*, pages 2496–2501. IEEE, 1990.

[15] K. David Young and V.I. Utkin. A Control Engineer's Guide to Sliding Mode Control. Technical Report 3, 1999.

[16] M. De Stefano, R. Balachandran, J. Artigas, and C. Secchi. Reproducing physical dynamics with hardware-in-the-loop simulators: A passive and explicit discrete integrator. In *Proceedings - IEEE International Conference on Robotics and Automation*, pages 5899–5906, 2017.

[17] J. Diebel. Representing Attitude: Euler Angles, Unit Quaternions, and Rotation Vectors. Technical report, 2006.

[18] J. C. Doyle and G. Stein. Robustness with observers. *IEEE Transactions on Automatic Control*, 7(4):472–489, 1979.

[19] J.C. Doyle. Structured uncertainty in control system design. In *24th Conference on Decision and Control*. IEEE, 1985.

[20] J.C. Doyle. Quantitative feedback theory (QFT) and robust control. In *American Control Conference*, pages 1691–1698. IEEE, 1986.

[21] A. Flores-Abad, O. Ma, K. Pham, and S. Ulrich. A review of space robotics technologies for on-orbit servicing, 2014.

[22] R.B. Friend. Orbital Express program summary and mission overview. In *Proceedings of SPIE*, volume 24, 2008.

[23] C. Gaz, F. Flacco, and A. De Luca. Identifying the dynamic model used by the KUKA LWR: A reverse engineering approach. In *Proceedings - IEEE International Conference on Robotics and Automation*, pages 1386–1392, 2014.

[24] E.J.M. Geddes and I. Postlewaite. An H_∞-based loop shaping method and μ-synthesis. In *Decision and Control, 1991., Proceedings of the 30th IEEE Conference on*, pages 533–538. IEEE, 1991.

[25] D. Glover. Design Considerations for space flight hardware. Technical report, NASA, 1990.

[26] K. Glover. A Loop Shaping Design Procedure Using H_∞ Synthesis. *IEEE Transactions on Automatic Control*, 37(6):759–769, 1992.

[27] D. Grover, S. Jacobs, V. Abbasi, D. Cree, M. Daae, J. Hay, W. He, X. Huang, Z. Jun, S. Kearney, T. Kuwahara, F. Lenzi, H. Mirahmetoglu, S. Morley, M. Otani, M. Pastena, M. Pinni, J. Schwartz, K. Shala, and C. Weeden. Development of on-orbit servicing concepts, technology option and roadmap (part i) - commercial aspects. *Journal of the British Interplanetary Society*, 61:203–212, 06 2008.

[28] G. Hirzinger, B. Brunner, J. Dietrich, and J. Heindl. Rotex-the first remotely controlled robot in space. In *Robotics and Automation, 1994. Proceedings., 1994 IEEE International Conference on*, pages 2604–2611. IEEE, 1994.

[29] G. Hirzinger, K. Landzettel, B. Brunner, M. Fischer, C. Preusche, D. Reintsema, A. Albu-Schäffer, G. Schreiber, and B.M. Steinmetz. Dlr's robotics technologies for on-orbit servicing. *Advanced Robotics*, 18(2):139–174, 2004.

[30] I. Horowitz. Quantitative feedback theory. In *IEE Proceedings D (Control Theory and Applications)*, volume 129, pages 215–226. IET, 1982.

[31] I. Horowitz. Quantitative feedback theory (QFT). In *American Control Conference*, pages 2032–2037. IEEE, 1988.

[32] R.A. Hyde and K. Glover. The application of scheduled H_∞ controllers to a VSTOL aircraft. *IEEE Transactions on Automatic Control*, 38(7):1021–1039, 1993.

[33] E.J. Ientilucci. Using the singular value decomposition. *Center for Imaging Science, Rochester Institute of Technology*, 2003.

[34] T.P. Jitendra, A.A. Selvakumar, and J.A. Corrales Ramon. Manipulation and Path Planning for KUKA (LWR/LBR 4+) Robot in a Simulated and Real Environment. *Journal of Automation, Mobile Robotics & Intelligent Systems*, 11(3):15–21, 2017.

[35] N.L. Johnson. Orbital debris: the growing threat to space operations. 2010.

[36] T.R. Kane and D.A. Levinson. Dynamics Theory and Applications. Technical report, 1985.

[37] R.E. Korf. *Space Robotics*. Number August. 1982.

[38] K. Landzettel, B. Brunner, G. Hirzinger, R. Lampariello, G. Schreiber, and B.M. Steinmetz. A unified ground control and programming methodology for space robotics applications-demonstrations on ETS-VII. In *International Symposium on Robotics*, volume 31, pages 422–427. German Aerospace Center, 2000.

[39] K. Landzettel, C. Preusche, A. Albu-Schaffer, D. Reintsema, B. Rebele, and G. Hirzinger. Robotic on-orbit servicing-dlr's experience and perspective. In *Intelligent Robots and Systems, 2006 IEEE/RSJ International Conference on*, pages 4587–4594. IEEE, 2006.

[40] A. Lanzon and P. Tsiotras. A Combined Application of H∞Loop Shaping and μ-Synthesis to Control High-Speed Flywheels. *IEEE Transactions on Control Systems Technology*, 13(5):766–777, 2005.

[41] S. Le Ballois and G. Duc. H_∞ control of a satellite axis: Loop-shaping, controller reduction and μ-analysis. *Control Engineering Practice*, 4(7):1001–1007, 1996.

[42] X.F. Liu, H.Q. Li, Y.J. Chen, G.P. Cai, and X. Wang. Dynamics and control of capture of a floating rigid body by a spacecraft robotic arm. *Multibody System Dynamics*, 33(3):315–332, 2014.

[43] T.M. Lovelly, K. Cheng, W. Garcia, and A.D. George. Comparative analysis of present and future space processors with device metrics. In *Proc. of Military and Aerospace Programmable-Logic Devices Conference (MAPLD)*, pages 19–22, 2014.

[44] U. Mackenroth. *Robust Control Systems: theory and case studies.* 2013.

[45] S. Ali A. Moosavian and Evangelos Papadopoulos. Explicit dynamics of space free-flyers with multiple manipulators via SPACEMAPLE. *Advanced Robotics*, 18(2):223–244, 2004.

[46] National Aeronautics and Space Administration. On-Orbit Satellite Servicing Study. Technical Report October, 2010.

[47] M. Oda. On the dynamics and control of ETS-7 satellite and its robot arm. In *IEEE International Conference on Intelligent Robots and Systems (IROS)*, volume 3, pages 1586 – 1593, 1994.

[48] M. Oda and Y. Ohkami. Coordinated control of spacecraft attitude and space manipulators. *Control Engineering Practice*, 5(1):11–21, 1997.

[49] Inc. Orbital ATK. Orbital ATK on track to launch industry's first commercial in-space satellite servicing system in 2018. `https://www.orbitalatk.com/news-room/feature-stories/MEV/default.aspx?prid=92`. [Online; accessed 05-February-2018].

[50] M. Reiner, J.G. Fernandez, and G. Ortega. Combined control for active debris removal using a satellite equipped with a robot arm. In *10th International ESA Conference on Guidance, Navigation & Control*, 2017. URL `https://elib.dlr.de/115866/`.

[51] B. Schäfer, R. Krenn, and B. Rebele. On inverse kinematics and kinetics of redundant space manipulator simulation. *Journal of Computational and Applied Mechanics*, 4(1):53–70, 2003.

[52] F. Sellmaier, J. Spurmann, and T. Boge. On-orbit servicing missions at dlr/gsoc. In *61st International Astronautical Congress*, pages 8376–8382, 2010.

[53] O. Sename. Robust and LPV control of MIMO systems Part 1 : Tools for analysis and control of dynamical systems. (July), 2016.

[54] M. Shan, J. Guo, and E. Gill. Review and comparison of active space debris capturing and removal methods, 2016.

[55] S. Skogestad and I. Postlethwaite. *Multivariable feedback control: analysis and design.* Wiley New York, 2007.

[56] NASA SSPD. Restore-l: Robotic servicing mission. `https://sspd.gsfc.nasa.gov/restore-l.html`, January 2018. [Online; accessed 05-February-2018].

[57] E. Stoll, J. Letschnik, U. Walter, J. Artigas, C. Kremer, P.and Preusche, and G. Hirzinger. On-orbit servicing. *IEEE Robotics and Automation Magazine*, 16(4):29–33, 2009.

[58] T. Uhlig, F. Sellmaier, and M. Schmidhuber. *Spacecraft operations*. 2015.

[59] V. Utkin. Sliding mode control. *Control Systems, Robotics and Automation - Volume XIII: Nonlinear, Distributed, and Time Delay Systems-II*, page 130, 2009.

[60] K.F. Wakker. Fundamentals of astrodynamics. 2015.

[61] B. Wie and J.A. Schetz. *Space vehicle dynamics and control second edition*. 2008.

[62] T. Wolf. Deutsche orbitale servicing mission. *Astra*, 2011.

[63] K. Wormnes, R. Le Letty, L. Summerer, H. Krag, R. Schonenborg, O. Dubois-Matra, E. Luraschi, J. Delaval, and A. Cropp. ESA technologies for space debris remediation. In *6th European Conference on Space Debris*, pages 1–2, 2013.

[64] K. Yoshida. Engineering test satellite VII flight experiments for space robot dynamics and control: theories on laboratory test beds ten years ago, now in orbit. Technical report, 2003.

[65] K. Zhou and J.C. Doyle. *Essentials of robust control*, volume 104. Prentice hall Upper Saddle River, NJ, 1998.