Politecnico di Torino

Master's degree in mechanical engineering

Master Thesis

Development of a simulation environment for the analysis of an OOS manoeuver

Student:

Christian Buongiorno

Supervisors:

Prof. Stefano Mauro

Dott. Pierpaolo Palmieri

Dott. Matteo Gaidano

Academic year 2022-2023

A Maria e alla mia famiglia.

Per aspera ad astra

TABLE OF CONTENENTS

Abstract

Nowadays the growing interest in space explorations, has brought the humankind to develop technologies and to expand the research for space application. As statistical data reveal, on average, 100 satellites have been launched every year in the past decade and few of them went through anomalies and failures. The growing number of space debris threatens access to space, therefore, human intervention is required to avoid ''Kessler syndrome''. The Kessler syndrome proposed by Donal J. Kessler is a scenario in which the density of objects in low earth orbit (LEO) due to space pollution is high enough to impede space exploration. Space robot systems have been identified as a key technology to perform active debris removal or on-orbit servicing and further studies are needed to make these operations fully autonomous.

The aim of the present thesis is to develop a Simulink environment for the analysis and simulation of the final approach phase of a rendezvous manoeuvre for an on-orbit servicing applications which consist in a target capturing operation.

In chapter 1, an overview upon the main topics covered by the present thesis is reported. Firstly, a general discussion on the use of space robots for on-orbit servicing is reported followed by a brief description of the main capturing methods and rendezvous strategies, then a description on the GNC function is reported. Lastly, a description on the main methods for dynamic modelling of the systems is introduced and the POPUP robotic arm first prototype is reported.

In chapter 2, the active stabilization system is designed. In this chapter the thruster allocation function is developed using the pseudo-inverse method and the pulse-width pulse-frequency modulator (PWPFM) is used to convert the continuous signal from the controller into an ON/OFF signal.

Chapter 3 focuses on the design of the controller. The position and attitude control are performed using the optimal controller called linear quadratic regulator (LQR) and the artificial potential field (APF) function for obstacle avoidance.

Chapter 4 focuses on the development of the differential kinematics algorithms to control the robotic arm in the different motion controls scenarios. Furthermore, a detailed description of the algorithm is reported describing the theoretical approach used.

Lastly, in chapter 5 the chosen parameters are presented, and different scenarios are analysed. The model is first simulated in the free-flying mode then in translation-flying mode, lastly in free-floating and rotation-floating modes. Results underline the differences in terms of the robotic arm parameters and the position and attitude errors of the base of the spacecraft. The results are obtained by varying the mass of the chaser spacecraft and the end-effector velocity.

CHAPTER 1

1 Introduction

1.1 On-orbit servicing.

When a spacecraft is set up into its operative orbit it becomes almost inaccessible and, if a failure occurs or the satellite runs out of propellant, it is likely to become a space debris. In order to remove this debris, the interest in the development of On-Orbit Servicing (OOS) missions has increased in the past decades.
Therefore, OOS missions are defined as orbital activities conducted by a space vehicle that performs up-close inspection of, or beneficial changes to, another resident space object. These activities include non-contact support, orbit maintenance or modification, refuelling and resources replenishment, upgrade, repair, assembly, and debris removal. These activities conducted in space have a lot of beneficial aspects, for example servicing a satellite in orbit offers the possibility to extend its operational life [24].
In addition, the technologies developed in the framework of OOS can be employed to address the space debris problem providing the chance to restore a failed spacecraft preventing it to become a space debris.
Space Debris is a growing problem, e.g., as shown in [70], 5-10 debris need to be removed every year to avoid Klesser syndrome.
To execute debris removal operations, manipulator systems mounted on a spacecraft (SMSs) can be employed [1]. A space manipulator system (SMSs), consist of a spacecraft equipped with one or more robotic manipulators with grappling devices (end effector) on them to which is demanded the grappling phase, moreover it is usually driven by a vision system.
Therefore, in the last decades is growing the interest in developing advanced technologies for autonomously capture a non-cooperative target, since they can be employed to address both the OOS and Active Debris Removal (ADR) missions.
In the literature is possible to find a large number of papers for space manipulator activity [6, 8]. In [2] a control law is obtained to capture a tumbling object in space, relatively to the tumbling target in the paper [5] the minimization of the base attitude deviation before and after the contact with the target is discussed from the viewpoint of the angular momentum distribution and a possible control sequence for the operation is proposed , while in [7] is introduced an optimal capture strategy with minimal impact on the robot's base spacecraft where an optimal future time is calculated.
In [3] is studied the motion control of a multi-arm free-flying space robot chasing a passive object in close proximity.
Deep research on dynamic singularities and on the dynamics coupling between manipulator and spacecraft can be found in [4].
Furthermore, in [9] is presented a combined prediction and motion planning scheme for robot capturing of a drifting and tumbling object with unknown dynamic using

visual feedback, while in [10], a computationally efficient noise adaptive Kalman filter for the motion estimation and prediction of a target is proposed.
In the framework of ADR missions, many solutions have been proposed to autonomously capture non-cooperative objects such as space debris.
However, an accurate and robust performance is required to the system to autonomously capture large, non-cooperative objects with a robotic arm. The actuation of the robotic arm to grasp the target generates reaction torques at the base of the manipulator acting as spacecraft attitude and position disturbances. A possible solution to minimize the attitude error generated by the robotic arm movement consists in minimizing the base reaction torques generated during the manipulator manoeuvre, otherwise it could be possible to build an efficient controller to ensure good performance for the chaser control [7,23].
The target can be cooperative, or non-cooperative and collaborative or non-collaborative as will be specified in section 1.4.1, generally the worst case possible is when the target has an uncontrolled motion, and no information are available.
Based on these considerations space manipulator system can be designed in different ways to execute the mission.
The future for this technology is identified in the autonomy of space manipulators in executing an entire capture operation [11].
The automatic onboard system must fulfil the following tasks: Guidance, Navigation and control (GNC) which allows the calculation and execution of manoeuvres and continuous control actions of trajectory and attitude, MVM (mission vehicle management) to manage the GNC modes or manoeuvres and sequence of phases, FDIR (failure detection, isolation and recovery) and the data exchange concerning the rendezvous process and the onboard control system [14].

1.2 Capturing strategies.

In general, an ADR mission consists of different phases [71]: Launch and Early Orbit Phase, far-range rendezvous, close-range rendezvous, capturing phase and removal phase. The main methods to execute an ADR mission are divided in two main categories: contact and contactless capturing methods. The second category has been though for asteroid deflection, hence it will not be treated here. A comparison between all the available technologies can be found in table 2 of [71]. As highlighted in the paper, the drawback of using a single arm strategy is: Higher probability of collision, grappling point required, and rendezvous and docking are needed.
In this thesis a contact capturing method is investigated, using a stiff connection performed by a single robotic arm. The main steps involved in the capturing phase are shown in figure 1.1 [71].

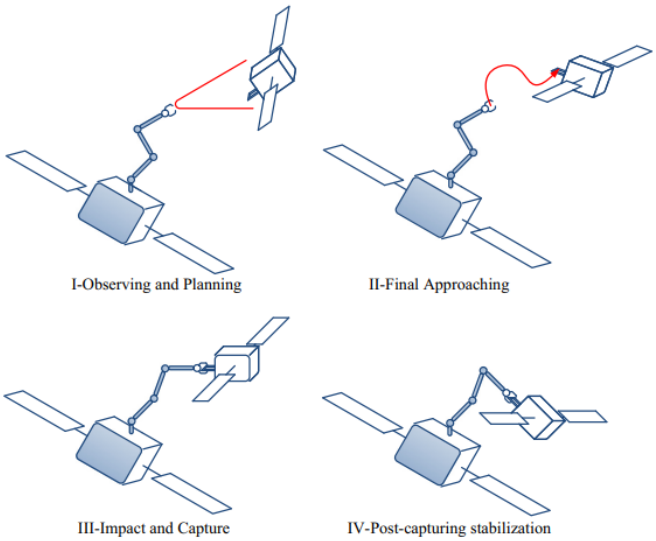

Figure 1.1: Capturing phase.

1.2.1 Single arm capturing.

Different operations have been already implemented such as ETS-7 of JAXA [72], Canadarm [73], orbital express of DARPA [74]. Since the non-cooperative target are usually tumbling, it is very difficult to perform a grasping operation using a single robotic arm for an ADR mission, indeed it is easier to perform an OOS operation from this point of view. The research landscape nowadays is focused in facing the following problem: minimizing the impact influence, de-tumbling, and attitude synchronization.

In the first case is possible to minimize the contact influence by controlling the direction of relative velocity between chaser and target which should pass through the centre of mass of the servicing system, otherwise would be possible to perform the minimization through approaching phase, impact phase and post-impact from the angular momentum point of view [75]. Another strategy uses visual servoing or, the concept of percussion points to minimize, as in [76].

According to [71], usually failed satellite are tumbling in space and do not have functioning attitude control, therefore based on the rotational rate the capturing scenario can be defined, e.g., if the target is orbiting at 3°/s the capture is easy, while, above 30°/s will not be regarded as a target, while in this range, since fly-around or grappling an object by a robotic arm will be difficult, is necessary to reduce the target's rotation to a rate at which capture can be performed using visual feedback control. A prototype of brush contactor can be used to accomplish this de-tumbling operation as reported in [99].

Lastly, the attitude synchronization allows to ensure the capturing point to be always directed towards the services satellite. To ensure the stability of the capturing motion is a common choice to keep a constant relative distance and attitude.

1.2.2 Rendevouzs strategies.

As already said, an ADR operation is executed initially as a typical rendezvous mission, in the following a brief summary of the common rendezvous and docking/berthing strategies is provided.
In general, as shown in figure 1.2, rendezvous and docking can be divided in four main manoeuvres: phasing, far range rendezvous, close range rendezvous and final approach [12, 13, 14].

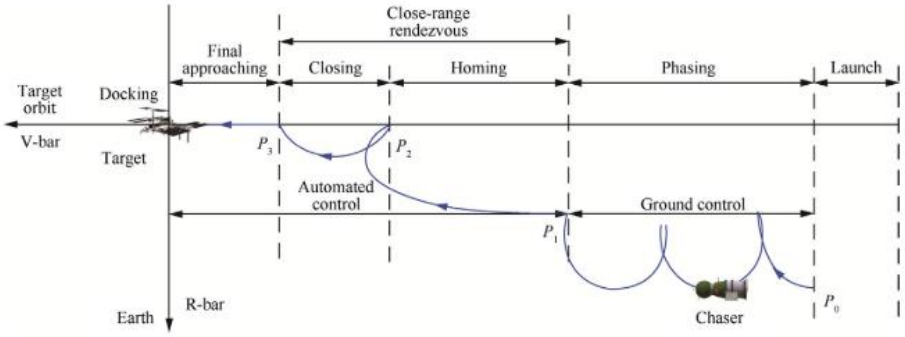

Figure 1.2: Rendezvous phase.

1.2.3 Phasing

The aim of this first orbital phase of a rendezvous mission is to reduce the phase angle between the chaser and target spacecraft, by making use of the fact that a lower orbit has a shorter orbital period. During this phase, launch injection errors for inclination and RAAN will successively be corrected. As a rule, all phasing manoeuvres are usually controlled from ground. Phasing ends with the acquisition of either an 'initial aim point', or with the achievement of a set of margins for position and velocity values at a certain range, called the 'trajectory gate' or 'entry gate'.
Usually, this first step consists in a series of Hohmann transfers and drift times. This approach offers several advantages over a direct injection into the target's orbit. First, the passive collision avoidance safety is always guaranteed given that, even in the case of chaser's complete control inability, the spacecraft will only drift below the target indefinitely. Second, the Hohmann transfers are generally the most fuel-efficient orbital transfers in LEO, which makes this approach very fuel efficient. Third, the timing of Hohmann manoeuvres and the duration of drift times can be appropriately tuned to meet specific mission requirements. I figure 1.3 is shown the orbital plane with a phase angle representative of the initial condition of phasing [12].

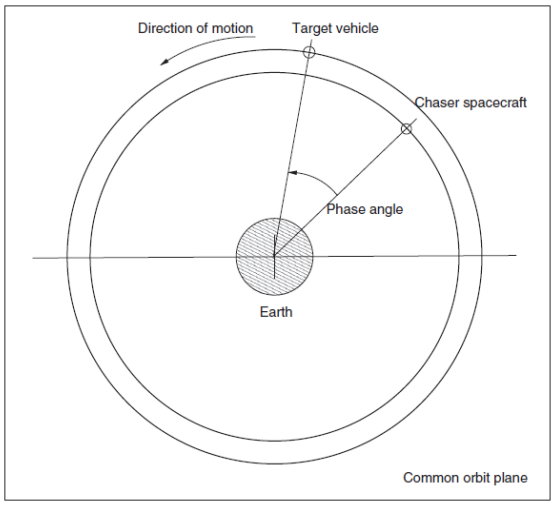

Figure 1.3: Orbit plane.

1.2.4 Far range rendezvous

This phase is also called ''homing'' phase, aims to bring the chaser in the vicinity of the target and create the conditions for close range rendezvous or final approach.
This phase consists respectively of a homing and closing rendezvous manoeuvre.
The objective to be accomplished in this phase is the reduction of the trajectory dispersions, achievement of position, velocity, and angular rate conditions necessary to start the next phase.
This phase can be as started when the relative navigation between chaser and the target is available, it is usually navigation performed using the on-board sensors such as an optical or IR camera and/or a LIDAR, the required measurements accuracy of the relative navigation sensor at the beginning of the far range rendezvous is in the order of 100 m, while the accuracy in the last part of this phase must be commensurate with the requirements for the close range operation.

1.2.5 Close range rendezvous.

The close-range phase is divided in two different subphases: a preparatory phase leading to the final approach corridor and a final approach leading to mating conditions.
The approach trajectory will vary according to the closing method chosen and the requirements of the robotic capture mechanisms. However, in any case it shall guarantee passive safety and to some extent fuel efficiency. The capture axis will generally be the main axis of rotation of the target body.
Once the capture axis has been reached, the manoeuvre will consist either in: (a) a straight-line trajectory, consisting of a series of hold points and constant rate motion within a predefined corridor or (b) in an optimized trajectory that limits as much as possible the active safety requirement and is fuel-efficient. The final selection of one of the two depends greatly upon the requirements of the robotic capture mechanism that will be defined in future studies. Nevertheless, in both cases the capture approach lasts until the berthing box is reached or the conditions for the capture are met.

1.2.6 Final approach.

This final manoeuvre is completed by executing the Docking or the Berthing with the chaser spacecraft [34].
During **Docking** operation, the chaser spacecraft get closer to the docking port of the target during the final approach manoeuvre until the contact between the two spacecraft is achieved. The docking is articulated in two main phases, the sof docking and hard docking. The first phase is achieved when the first contact between the two spacecraft is computed resulting in a mechanical connection, however, in this first step the link still allows a limited relative motion between them.
Once the soft mechanical link is activated, a second mechanism moves closer the two spacecraft and it performs a connection between them: at this step, the two spacecraft are strongly connected and they can be considered as a single vehicle (by the point of view of the dynamical behaviour), and then the hard docking is completed.
The **Berthing** manoeuvre consists in reducing the distance between the chaser and the target. The pre-grasping operation is stopped at few meters from the target spacecraft to perform the grasping manoeuvre with the robotic arm located on the chaser spacecraft. The robotic arm will operate until the chaser is fully connected to the target. Therefore, the chaser will hold the imposed relative position with the control action or operate in a free-floating mode. In this case, a "passive" grasping mechanism, or grasping spot, compatible with the "active" grasping mechanism of the robotic arm shall be installed on the chaser. The berthing strategy can be considered safer than the docking one, since the connection by the two spacecraft is executed operating a robotic arm instead of being driven by the on-board computer of the chaser and actuated by the thruster system, and a smaller number of failures may occur, in addition to the lower mechanical energy involved.
The present thesis has the focus of study the final approach with a berthing operation.

1.3 Capture of the target.

Typically, the capture operation of a tumbling target with a robotic arm is accomplished in the following steps: a pre-grasping phase, a post grasping phase and a grasping phase. The pre-grasping phase can be further divided in two steps: an approach manoeuvre to a predefined mating point and a manoeuvre of the robotic arm to move the end effector onto a capture point, done in this case by the deployment of the inflatable robotic arm through the pneumatic line [1,63]. In general, the capture phase can be extended to a phase which usually takes place before the approach manoeuvre, that is the observing and planning phase for acquiring motion and physical properties information about the target, to plan the successive manoeuvres [24].

1.3.1 Approach manoeuvre.

As already outlined before, the approach manoeuvre is demanded to the AOCS system in order to reach a predefined mating point.

Its execution depends on the target state for which a specification can be found in [1,71], so, with a locally stationary target is possible to achieve the mating point through a point-to-point planning using simple on-off thruster control and the attitude fine pointing with reaction wheels.
If the target is non-cooperative the approach phase become very difficult since is required the matching of the velocities between SMS end-effector and capture point, the phase can be conducted without the need of the synchronization of the chaser and the target. Further research can be found in [1].

1.3.2 Pre-grasping phase.

In deploying a SMS for target capture, a manipulator trajectory needs to be executed to achieve the goal and the capture point.
During this operation, other optimization goals are, singularities avoidance, fuel consumption and base disturbances minimization.
The manipulator moves from its collected position to a grapple point with zero relative velocity. In [78], the manipulator manoeuvres are optimized minimizing a cost function which includes the time of travel and the weighted norms of the end effector velocity and acceleration.
Moreover, the pre-grasping trajectory manoeuvre and grasping of tumbling target using actual vision feedback were successfully demonstrated in [80] where a combined prediction and motion planning scheme or robot capturing of a drifting and tumbling target was used. In the system is implemented a Kalman filter which allow to estimate the states and a set of dynamic parameters of the target needed for a long-term prediction of the motion from noisy measurements of visual servoing. Here, the optimal trajectory to catch the target minimizes a cost function which is a weighted linear sum of ravel time, distance and a line-of-sight angle, and a penalty function acting as a constrain on acceleration magnitude. In this thesis is assumed that the deployment of the robotic arm has already been computed.
Therefore, as reported in [63], these initial phases are shown in the simulation environment in the figure 1.4, where the approach, deployment and capture phases are exposed.

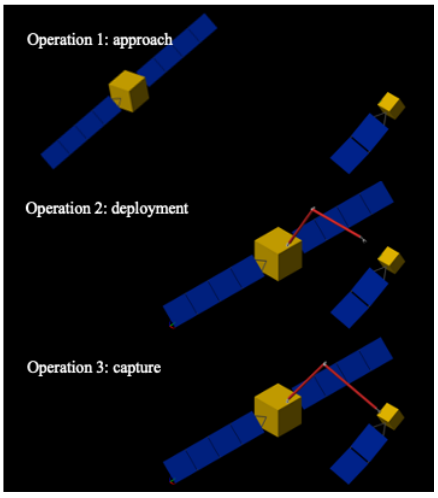

Figure 1.4:Operation sequence.

1.3.3 Post-grasping phase.

After that the grasping operation is concluded, the SMS should start to apply force and torques through the control system to stabilize the overall system and transferring the angular momentum to the SMS. A path planning method must consider the permissible bounds on the interaction moments between the SMS and its target during detumbling, otherwise, excessive moments and forces could lead to mechanical damage or actuation saturation. In [78], the manipulator dumps the initial velocity of the tumbling target in minimum time, subject to the constrain that the magnitude of the torque applied remain below a safe value.
Other research [79], realized a stabilization of tumbling combinations after target capture by combination of a tethered space manipulator and thrusters.

1.4 Operational Orbit

A low Earth orbit (LEO) is, as the name suggests, an orbit that is relatively close to Earth's surface. It is normally at an altitude of less than 1000 km and greater than 300 km above Earth.
Unlike satellites in geosyncronous equatorial orbit (GEO) that must always orbit along Earth's equator, LEO satellites do not always have to follow a particular path around Earth in the same way, their plane can be tilted. This means there are more available routes for satellites in LEO, which is one of the reasons why LEO is a very commonly used orbit. LEO's close proximity to Earth makes it useful for several reasons. It is the orbit most commonly used for satellite imaging, as being near the surface allows it to take images of higher resolution [100]. It is also the orbit used for the International Space Station (ISS), as it is easier for astronauts to travel to and from it at a shorter distance. Satellites in this orbit travel at a speed of around 7.8 km per second; at this speed, a satellite takes approximately 90 minutes to circle Earth, meaning the ISS travels around Earth about 16 times a day [100]. However, individual LEO satellites are less useful for tasks such as telecommunication, because they move fast across the sky, requiring a lot of effort to track from ground stations. Instead, communications satellites in LEO often work as part of a large combination or constellation, of multiple satellites to give constant coverage. In this thesis the operative orbit is assumed to be LEO orbit since it is the place where we can observe the Klesser Syndrome [15], one of its worst implications could be to inhibit the future space exploration.
Therefore, the LEO represents the environment of a typical ADR mission.
One of the main possible solutions to stabilize the growth of debris and to avoid the Kessler syndrome is to capture and de-orbit non-functioning satellites and debris (Figure 1.5). As known from literatures, deorbiting five massive debris a year could be enough [16] to prevent this phenomenon from growing. Several possible solutions have been proposed to perform an *active debris removal (ADR)* mission [17], as reported in [18] a robotic arm could be used to capture the debris [18]. On top of removing debris, servicing satellites is key to expand the lifespan of spacecrafts and ensuring the Earth orbit is clean. In figure 1.5, the space debris evolution method is reported [100].

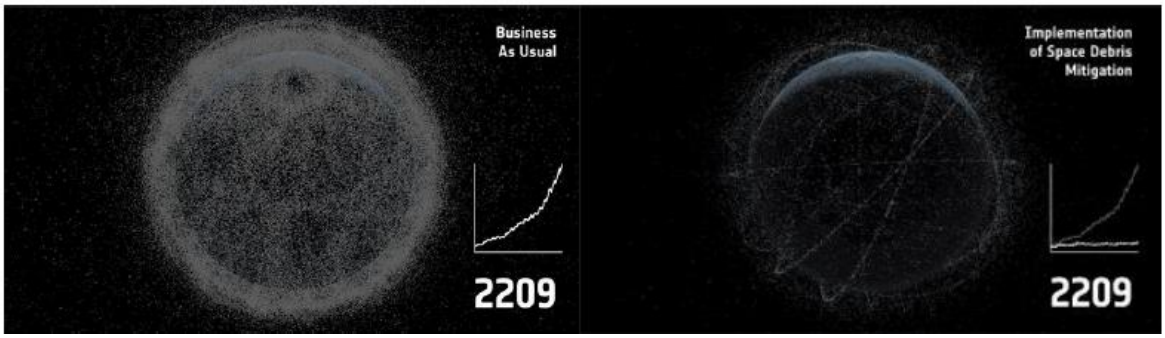

Figure 1.5: Space debris evolution with and without mitigation methods.

1.4.1 Target identification

According to [19], the design of the navigation and control subsystems is strongly affected by the degree of cooperation and collaboration of the target. The definitions of the cooperativeness and collaboration of the target are reported in Table 1.1 [19].

Target type	Description
Cooperative	The target can provide direct information about its relative states in real-time on-board to the servicer to aid the relative navigation task.
Semi-cooperative	The target can provide indirect information about its relative states to the servicer through exploitation of active/passive markers.
Non-cooperative	The target does not offer any support for the relative navigation
Collaborative	The target can actively and accurately maintain an attitude profile that can aid the approach and docking/capture process.
Semi-collaborative	The target can actively keep an attitude profile to aid the approach but not accurately enough to aid the docking/capture process, i.e., only coarse attitude control is operative
Non-Collaborative	The target attitude is uncontrolled, and it cannot aid the capture operation in any way.

Table 1-1: Target description.

1.5 Guidance, navigation, and control subsystem

The control loops for attitude and trajectory control include the sensors for position and attitude measurement, the navigation, guidance and control functions, which are implemented in software in the onboard computer, and the thrusters and other actuators for attitude and position control. A block diagram of a typical control loop for one of the six degrees of freedom (DOF) is shown in figure 1.4 [14].

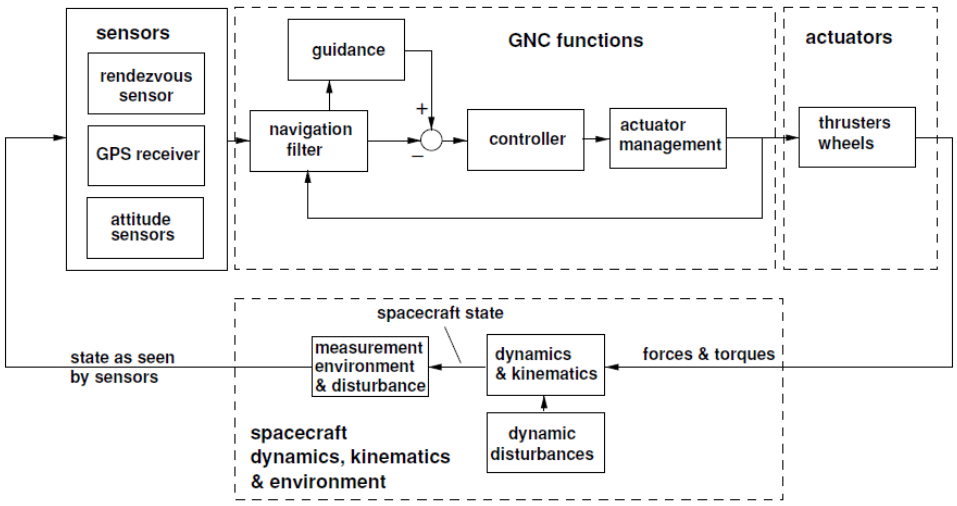

Figure 1.6:GNC architecture.

During the approach, depending on the distance between the chaser and the target spacecraft and based on the objective, different translational and rotational manoeuvres need to be computed, and various types of trajectories must be controlled, for which different sensor types must be used and different guidance, navigation and control mode have to be chosen. This requires a reconfiguration of the control loops each time, in which algorithms and parameters of the navigation, guidance and control functions may have to be changed [14].
As in the research of [20], each software module is a set of algorithms capable of executing a task. Those modules and their principal tasks are:

1) **Navigation module:** performing the pose estimation of the target.
2) **Guidance module**: performing trajectory planning towards the capture axis and ultimately towards the target with safety and fuel-efficiency.
3) **Control module:** performing the execution of manoeuvres according to the guidance function and the navigation and suppression of external disturbances.

1.5.1 Guidance

The Guidance function shall, in general, drive the spacecraft to follow a defined path or trajectory, in order to reach a different orbit or to keep a specific slot in the actual orbit of the spacecraft, counteracting external disturbances. Therefore, an efficient guidance for a rendezvous and docking mission shall compute the optimal trajectory to reach the target spacecraft and shall generate the force or velocity change (ΔV) command which allows the chaser spacecraft to follow the computed path.

Assuming a closed loop guidance for an autonomous GNC system, the guidance command is computed considering the actual state (position and velocity) of the spacecraft fed back by sensors measurements with respect to the desired state, there are two possible strategies in designing the guidance algorithm.
In general, the guidance function mode is managed by a flight manager which can select the mode based on the target distance as reported earlier.
Depending by the thrust to mass ratio of the spacecraft, can be chosen a two impulse or a continuous thrust manoeuvre. The two-impulse manoeuvre consists in commanding a first acceleration to reach the commanded ΔV and a second impulse at the end of the scheduled manoeuvre. Usually, the second impulse is commanded immediately after half an orbital period is completed, since it is the cruise time typical for an impulsive manoeuvre [34].
There are different kind of algorithm which can be used to cover this function, here there are some: LQR for guidance, ZEM/ZEV, MPC (model predictive control), Lambert guidance, SMC-sliding mode control [34].
The main limit in such use is the on-board implementations, since this thesis has the aim to model in Simulink environment a SMS close to realty, an easy guidance algorithm is used to face the final approach.
Another guidance function in the model must be absolved by the trajectory tracking for the end effector of the robotic arm, which will have to follow a certain path to grasp the target.
In this thesis the guidance function must provide trajectories and attitude profile for the final approach in order to reach a berthing point and start the capture phase.
In a real system, usually, errors and uncertainties make compulsory the choice of a Feedback control system to improve the performance of the system.

1.5.2 Navigation

The task of the navigation function is to provide the controller and the guidance function with the necessary information on the present state of the vehicle. As a rule, this function is implemented as a digital filter which processes the various information inputs related to the vehicle state obtained from different sensors, from the actuators or via communication links from external sources. The purpose of such a filter is to obtain out of several inputs related to the vehicle state an estimation of the state vector with reduced noise errors. A filter, which propagates the state, will also be helpful in cases where the sensor information is only intermittently available. If there were a single sensor continuously providing all necessary information on the state vector with sufficiently low noise, the navigation function could be reduced to converting the sensor information to formats as required by the guidance and control functions [14, 34].

1.5.3 Control function

Analysis of control system will focus on actuation systems used for position and attitude control and related control techniques, while investigation of control algorithms will be left to the following sections. Generally, with the term control it is intended both attitude and position control [14, 34].

Attitude Control

Controlling the attitude of the spacecraft is compulsory in space application since there is a high number of factors which need to be considered in the definition of the attitude pointing requirements. Indeed, controlling the attitude of the spacecraft is fundamental to control the temperature, to maintain solar arrays pointing the Sun, to maintain or reorient the payload along with a desired orientation, to align the docking mechanism with the docking port, to align the thrust vector of the apogee engine through a specific direction to execute the desired orbital manoeuvre, to reduce the disturbance of the robotic arm operation and many other purposes.

To fulfil such functions, the spacecraft is equipped with an actuation system which can modify and control the attitude of the spacecraft, the problem is addressed in the next chapter.

Actuators for attitude control and active stabilization are usually selected among reaction control thrusters, reaction wheels, gyroscopes, magneto torquers. Attitude control, or stabilization, can be achieved also with passive techniques, such as gravity gradient stabilization and spin stabilization, the choice of one of this method is based on the aim of the space mission and on the pointing accuracy requirements.

The main disadvantage of performing the attitude control using reaction control thrusters is a very poor control accuracy, since reaction control thrusters used for attitude control are usually providing low thrust, compared to thruster used for orbital manoeuvres, and they have an ON/OFF working principals, the thruster can be activated or not generating the maximum thrust, which is designed for, without the possibility to modulate the magnitude of thrust.

The use of modulation techniques, such as Pulse-With Modulation (PWM) or PWPFM, can partially overcome this problem, but the presence of thresholds and dead-bands due to the minimum impulse bit that can be provided by thrusters introduce an intrinsic limitation. For this reason, attitude control using thrusters is executed mainly during the de-saturation of the reaction wheels or other momentum exchange devices, during contingency manoeuvres or when the required control torque is too high and it cannot be provided by other actuators, concurrently with them. To generate control torque, reaction control thrusters are used coupling activation of two thrusters firing in opposite side and direction in the spacecraft configuration. However, in this work is implemented a flight mode which use just thruster for the stabilization.

Fine attitude control can be achieved using momentum exchange devices, such as reaction wheels. Such devices are composed by a flywheel (rigid disk) which is rotating and controlled in velocity. Since the disk of a reaction wheel has its own inertia, depending by the mass of the disk, it is required a torque to accelerate it to a different speed. The torque is provided by the controlled electric motor and, during the disk acceleration and by the third Newton's law, the reaction torque, with the same magnitude but opposite sign, it is transmitted to the mechanical support of the motor, and eventually to the spacecraft structure, since the motor is mechanically linked with it, changing the attitude of the spacecraft. Such systems are subjected to two main limitations of maximum torque provided by the motor and the maximum angular speed of the wheel. To overcome this problem a specific design must be considered.

The first limit, related to the maximum torque, is due to the maximum torque that the electric motor can provide, and hence this limit is due to the maximum current which can flow inside the coils of the motor. Increasing the maximum torque will increase the size of the motor, and so the mass of the system, and this issue it should be considered in the design of the attitude control system.
Another limit is related to the maximum speed, it is instead due to the maximum velocity that can be achieved by bearings which support the disk and the engine. Exceeding the maximum velocity can damage the device, which can result in a mission failure if there are no redundancy due to the impossibility to control the attitude along the axis related to the damaged wheel. When a disk reaches the maximum velocity, the system is not able to provide torque, since it cannot accelerate the disk anymore. In such condition, to make the wheel usable again, it is required to slow down the wheel. Slowing down the wheel means applying a torque in the opposite direction, and hence the attitude of the spacecraft will change if no other techniques are used to counteract the decelerating wheel. Commonly, reaction control thrusters and magneto torquers are used to generate a torque which counteract the deceleration of the wheel, maintaining the attitude of the spacecraft within a limited drift (desaturation).
A further effect which affects reaction wheels is the friction, in particular the static friction since the rolling friction is much lower in magnitude. Static friction is mainly responsible of delay in the response of the wheel system since it is required a greater torque to start moving an object with respect to change the momentum of a moving one. Static friction can become an important problem if the design of the reaction wheel is not done properly, or the thermal control of spacecraft fails. Indeed, mechanical deformation due to temperature can be high enough to cause choking of the bearings, blocking the rotation of the disk. An additional effect of friction is due to the drag acting on the disk rotating at high speed, but in non-airtight system this effect is not present. A common way to minimize the effect of (static) friction is to set a minimum rotation speed as "rest" condition: when the wheel is de-saturated is slowing down to this velocity instead to zero velocity, making the system more reactive and freer from related oscillation caused by static starting of the wheel.
As mentioned at the beginning of this section, attitude control can be achieved also with passive techniques. Passive stabilization techniques are based on the dynamic response of a rigid body, in some case rotating, in other case spinning about an axis.
Passive stabilization can be realized by the gravity gradient stabilization technique. This technique uses inertia properties of the spacecraft in the gravitational field of the Earth. Due to the gravitational law, in an ideal system consisting of a mass-less bar with two equal masses fixed at the extremities of the bar, the mass located closer to the Earth is subjected to an attractive force greater than the attractive force acting on the mass farther from the Earth. This differential force acting on the system generates a torque which tends to align the system along the local vertical. Extending this principle to objects designed with more complex inertia properties, this results that the object subjected to the gravitational field of Earth will tend to align its axis with lower moment of inertia to the local vertical axis. Some precautions can be adopted to minimize oscillation of spacecraft using this stabilization technique, such as by equipping the vehicle with dampers. This technique is used when less restrictive

requirements are adopted, since no control or stabilization can be realized about the local vertical axis using gravity gradient stabilization.
Spin stabilization consists in making the spacecraft spinning about an axis and, by conservation of angular momentum, the attitude is stabilized to the initial orientation by the gyroscopic effect. Spin stabilization requires to be started while in orbit, for example using thrusters to starts the spacecraft spinning. Due to imperfection in inertia estimation and thruster errors, the resulting angular momentum is not perfectly related to a pure spin, so there can be present residual nutation movement that has to be damped. In addition, spin stability has to be satisfied: to reach spin stability it is used to spin the spacecraft about the axis with the lower inertia [34].

Position control

Position control is usually required for station keeping or rendezvous and docking purposes, as well as deep space exploration. The station keeping maneuverer consists in correcting orbital parameters of the spacecraft, restoring them to the designed values. Main cause of orbital drift in LEO orbits is the effect due to the residual atmospheric drag, while in higher orbits the effects due to the geopotential field of the Earth (the so called J2 effect) is the leading one. Other effects, such as third body perturbation and solar radiation are present, but a non-negligible value can be found only in higher orbits. Depending by several design parameters, such as station keeping accuracy, operational life expectation of the spacecraft, spacecraft mass, and more, actuators used to ensure the station keeping affect the design of the propulsion system, since they are de facto part of this subsystem.
Rendezvous and docking purposes require execution of many orbital maneuverer in a limited period.
The common practice is the use of the reaction control thrusters to execute all the orbital maneuverer. The reaction control system can also be used for attitude control purposes, as discussed in the previous section. The basic principle of a mono-propellant cold gas thruster is very simple: the gas is stored in a pressurize tank and it is connected to the nozzle using a pressure regulation valve and a flow control valve; the pressure regulation valve stabilize the output pressure of the gas to a specific value, and the flow control valve can be opened or closed allowing the gas flowing; the gas flows in the throat of the nozzle, expanding and accelerating, generating thrust. More complex and more effective thrusters include a combustion chamber before the throat of the nozzle, in which two propellants are ignited (fuel and oxidizer) increasing the total pressure inside the combustion chamber and generating higher thrust, increasing the specific impulse as well.

1.6 Dynamics of serial multibody systems.

As can be found in [23], there are two common methods for modelling the dynamics and deriving the equations of motion of multibody systems: the recursive Newton-Euler method and the Lagrangian method.
A general description of the Newton-Euler method, given in [21,22,24], provides a detailed presentation of the use of Newton-Euler dynamics to develop the equations of motion for a multibody spacecraft.

In the Newton-Euler method, the equations of motion of the multibody system are computed from the equilibria of forces and torques acting on each link of the system. From this, a recursive algorithm can be developed. In the forward recursion through the structure of the multibody system, the linear and angular link accelerations and velocities are computed. The forces and moments acting on the links are then computed in the backward recursion. To develop the equations of motion of a system of flexible links, the Direct Path Method was developed. In the Direct Path Method, the point of reference of the equations of motion is moved from the system centre of- mass to a fixed point in one of the bodies, which is typically selected to be the centre-of-mass of the base spacecraft. The structure of the system is then described following the most direct path through the links. The torques acting on the links are taken about the joints instead of the link centres-of-mass, thus eliminating constraint forces and torques between the links.

The Lagrangian method develops the equations of motion of a multibody system from its kinetic and potential energies, using a set of generalized coordinates describing the positions of the link. Following [21], the Lagrangian method is advantageous in it being systematic and easily comprehensible and in providing the equations of motion in a compact analytical form facilitating control systems design. The fundamental advantage of the Newton-Euler approach is its computational efficiency as a recursive algorithm.

Classification of Spacecraft-Manipulator System Manoeuvring, and thus not fixed in space, any motion of the manipulator will cause a rotation and translation of the base spacecraft. A comprehensive overview of methods to account for the dynamic coupling in controlling the position and orientation of both the end-effector of the manipulator and the base-spacecraft is provided in [24].

Three of these methods shall be highlighted: The Virtual Manipulator (VM) approach, the Dynamically Equivalent Manipulator (DEM) approach, and the Generalized Jacobian Matrix (GJM) approach.

Another method for modelling the dynamics of a multibody system is the usage of Simulink, in this thesis work the dynamic of the system is obtained using Simscape multibody which allow us to implement the mechanical system in our simulation environment.

Moreover, the dynamic description of the system gets more complicated if the non-linearities, such as flexible links, flexible joints to friction are considered.

1.6.1 Classification of spacecraft-manipulator system manoeuvring

The spacecraft-manipulator system can manoeuvre in different modes, typically designated by the terms free-floating and free-flying. The following classification has been taken from [23].

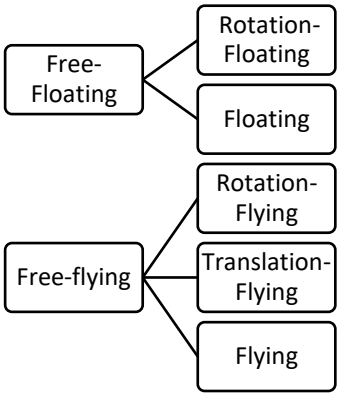

Figure 1.7: Classification.

Free-Floating
A spacecraft-manipulator system is defined to be floating when the degree of freedom of the spacecraft, 3 DOF for the translation and 3 DOF for the orientation, are not actively controlled, while the joints are actively controlled by the manipulator's joint motor. Therefore, the system moves only under the effect of the internal reaction due to the actuation of the motors. As robotic manipulators mounted on spacecraft typically only use revolute joints, these internal reactions are typically torques.

Rotation-Floating
A spacecraft-manipulator system is here defined to be rotation floating when only the 3 DOF of orientation are controlled, this is done sing momentum exchange devices such as the reaction wheels, which are considered to be internal torques. While the 3 DOF of translation are not actively controlled, therefore the COM of the SMS will move under the effect of the robotic arm joint torques. Indeed, as in the previous case the joint DOF are actively controlled by manipulator's joint motors.
The rotation-floating manoeuvre case thus differs from floating in that the attitude of the base spacecraft is actively controlled by momentum-exchange devices. The three DOF of translation of the system's centre-of mass are not actively controlled.

Rotation-Flying
A spacecraft-manipulator system is here defined to be rotation flying when the DOF at the manipulator joints are actively controlled by joint motor torques and the three DOF of orientation of the base-spacecraft are actively controlled by external torques only. This is typically done by using reaction-jet thrusters firing in couples, thus generating a pure torque with total null force. The three DOF of translation of the system centre-of-mass are not actively controlled. Therefore, the system's total linear momentum is in this case constant while the angular momentum is time-varying.

Translation-Flying
A spacecraft-manipulator system is here defined to be translation-flying when the 3 DOF of translation of the base and the joint's DOF are actively controlled by external forces and torques, while the attitude is controlled using internal torques by reaction momentum exchange devices. Usually, the base translation is controlled using reaction-jet thrusters.
Therefore, the system's total angular momentum is in this case constant while the linear momentum is time-varying.

Flying
A spacecraft-manipulator system is defined to be flying when the 3 DOF of orientation and the 3 DOF of translation are actively controlled using external forces usually provided by thrusters. Moreover, as in the other case the manipulator joints are actively controlled by joint motor torques. To perform an attitude control and a position control a minimum number of 12 thrusters are needed. In this case, both the system's total angular momentum and the linear momentum are time varying.

The classification above is rigorously valid only for an isolated spacecraft-manipulator system. A spacecraft manipulator system is never isolated but orbiting an extended body (e.g., the Earth) under its gravitational attraction. However, the classification above can still be used in an approximate sense, due to the weightless (e.g., free-falling) condition of the system centre-of-mass (due to the balancing of gravitational attraction and centrifugal force) and neglecting the effect of environmental torques (typically dominated by gravity-gradient torque, atmospheric torque, and solar radiation-pressure torque) and non-gravitational environmental forces (typically dominated by atmospheric drag and solar radiation-pressure). The analysis and simulation of floating, rotation-floating, and rotation-flying manoeuvring modes can be typically conducted with good accuracy as if the system was isolated. In those three cases, a coordinate system centred at the centre-of-mass of the orbiting spacecraft-manipulator system and having axes oriented in a fixed way with respect to an inertial frame (i.e., having zero absolute angular velocity) can be considered as equivalent to an inertial coordinate system for the description of the spacecraft manipulator system motion.
In this thesis work all the flight mode control are studied and analysed for the spacecraft POPUP manipulator system.
For the Free-Flying mode, the base-spacecraft is assumed to be always stabilized, therefore, the system can be treated as a terrestrial manipulator with a fixed base in an inertial frame, as will be further treated in chapter 4. However, in some cases, the AOCS might not be able to achieve good performance due to the perturbation environment or parametric uncertainties. Thus, the assumption of fixed base-manipulator is not always valid; this introduces additional complexities. Moreover, for the Free-Floating case, the COM of the overall space-robot is assumed to be fixed in space, thus, the total linear and angular momentum of the space robot is conserved, and this COM point can be used as the inertial reference frame of the system. However, under an externally perturbed environment or whenever the on-board thrusters are used, this assumption is not valid.

In conclusion, the best strategy to follow would be to employ all the previous mentioned motion control technique organizing them in different phases of operation make us able to optimize the fuel consumption. Of course, the AOCS system should be accurate enough to cancel all residual angular and linear momentum before starting the free-floating phase.
The trajectory for the base-spacecraft can be even designed in such a way that minimizes fuel consumption. Additionally, this approach uses the COM of the target spacecraft as the reference frame of the system, thus, avoiding the problems faced by the Free-Floating approach. This mode of operation has the benefits of both Free-Flying and Free-Floating systems, i.e., it minimizes fuel consumption and does not encounter any dynamic singularities in the workspace.
Moreover, this mode of operation results in a highly redundant system and offers unlimited workspace, which is not possible with the traditional approaches. Additionally, the Controlled-Floating Space Robot (CFSR) is even more efficient when the grasping point on the target spacecraft is out of reach of the robotic arm. This is because the joints of the robotic arm and the pose of the base-spacecraft can be controlled simultaneously to access the desired grasping point. The following Table gives a breakdown of the Pros and Cons of the main approaches [35].

	Pros	**Cons**
Free-Floating	No Fuel consumption due to absence of active control of base-spacecraft	Base is allowed to move and rotate freely. Kinematics affect the Dynamic properties. Dynamic singularities occur in workspace. Un-defined workspace Non-holonomic redundancy
Free-Fliyng	Stabilized and Controlled base No dynamic singularities Like Earth-based manipulators Only Kinematic singularities	Excessive fuel consumption Restricted workspace Actuator saturation
Controlled-Floating	Optimum Performance Infinite Workspace Matching Linear and Angular rates No Dynamic singularities	High Complexity Fuel consumption due to active control of base-spacecraft

Table 1-2: Flight mode control.

.

1.6.2 Engineering simulator.

The simulation support for the testing and validation of the Guidance and Navigation technologies is mainly developed in the MATLAB/Simulink environment producing as an output a highly modular and flexible tool namely the Functional Engineering Simulator (FES).
In the Simulink model, the dynamics of the chaser, target and robotic arm are represented using the Simscape Multibody package, which is also coupled with a useful mechanical visualizer. This allows the developer to focus on the modelling of the physical phenomena, without having to worry about the correct formulation of the complex differential equations which govern the kinematics and the dynamics of

the single bodies. Moreover, using Simscape Multibody many rigid bodies can be modularly defined and connected using a variety of different joints.
Simscape Multibody was also found to be suitable to simulate the orbital dynamics of the spacecraft. By means of a fictitious 6 DOF joint, the two spacecraft can be separately connected to the ECI frame (centered in the center of the Earth). By applying the correct gravitational attraction, the two bodies gravitate according to the real orbital parameters and the relative motions between the two satellites is automatically reproduced with an acceptable degree of error without the need of using the CW equations of motion.
This thesis has the aim to develop a complete Simulink model to simulate the dynamics of the spacecraft on which is mounted the **POPUP robot,** a prototype built and projected by the mechanical and aerospace department of Politecnico di Torino. Some examples of the use of Simulink and Simscape multibody can be found in literature [65, 66].

1.7 Lightweight robotic arms.

In the actual landscape of research in space manipulator system, the design of lightweight and long reach robotic arm represents a rising trend in recent years, due to the possibility to increase the reachable workspace and reduce the launch mass. One of the most common problems in this kind of robotic arm is the structural flexibility of the links that could cause structural vibrations, which are profound when manipulating heavy payloads. As outlined in [60, 61], space manipulators are also subject to joint flexibility due to motor torque ripples, joint transmission elements and actuator shafts.
Moreover, the number of degrees of freedom of the system is twice as the number of control inputs. The flexibility of the structure can be neglected if the manipulator is moving a very low speed, or the structure is stiff enough. However, oscillations can result when very large payloads are handled.
In general, these flexibilities may cause vibrations in the manipulator and in the spacecraft especially when physical contact occurs. The assumption that allows to consider the flexibilities lumped in the joints is reasonable only for manipulator with short links. Flexible links can be modelled as Euler-Bernoulli beams and a finite-dimensional model of the link flexibility can be obtained [69].
Moreover, to study the flexibilities, is possible to refer to different paper [61], in this case flexibilities are addressed to joint flexibilities, aiming to study their effects in the design of the control system.
As it is well known, the manipulator natural frequencies change with the manipulator configuration and payload [67]. Another important result is obtained in [68], where the interactions between attitude control of the spacecraft and flexible modes are investigated.
As described in [69], to supress the vibration is necessary to design a proper trajectory of point-point motion. The trajectory planning strategies can be classified in two main categories: The direct method, in which trajectory planning problem can be transferred to parameter optimal problems after defining trajectories as specific function. The second one is the indirect method, where the trajectory planning problem can be transferred to optimal control problems.

Another aspect to consider to supress the vibrations, is to devise an effective controller which can eliminate the intense vibration of flexible mechanism.

1.7.1 POPUP robot.

Soft robotics allows the design and development of systems able to adapt unstructured and hostile environment, they are divided in two main categories, articulated and continuum. Specifically, the inflatable structure applied in the field of soft robotics allows the design of lightweight robotic arm and physical compliance increasing payload-to-weight ratio. The main disadvantages are the low payload capacity, complexity in kinematic and dynamic description, due to the non-linearities induced by large deformations in the material and physical coupling [62].
POPUP robot concept is shown in figure 1.8 [62], consisting of two inflatable links, three electric motors and rigid joints made by additive manufacturing.

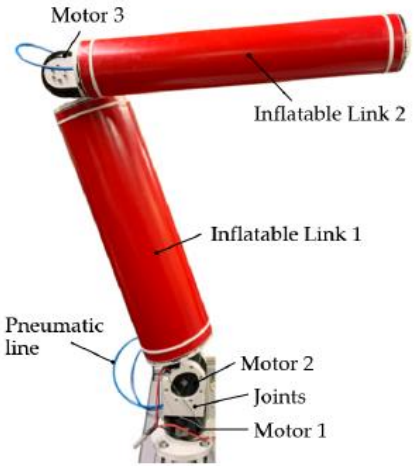

Figure 1.8: PUPUP prototype.

The system includes a pneumatic line for the inflation of the links an electric circuit and built in sensors.
The robot links have cylindrical shape, the surface is made by wounding a sheet of polymeric material. On the extremities two rigid links are glued. As shown in figure 1.10 [62].


Figure 1.9: Link structure.

The pneumatic plant described in [62] is responsible for the deployment phase of the robotic arm setting the inner pressure. Once inflated it can be modelled as a traditional space manipulator with flexible links [63].

In the above description has been introduced the robotic POPUP concept, the simscape modelling will be further detailed in next chapters. Furthermore, another prototype is under development by the Polytechnic of Turin for simulations under microgravity conditions.

1.8 Problem formulation

To treat rigorously the problem, in this thesis some assumptions are made to simplify the problem and to neglect some parameter. This work focuses on the last operation of capture, starting from an initial configuration. The distance between the chaser and the target is imposed. The following assumptions are made:

- Both the chaser and the target spacecraft are composed of rigid bodies in three-dimensional space, the manipulator has rigid bodies except for the two main links which are flexible.
- Environmental forces (gravity gradient, solar radiation pressure) as well as relative orbital dynamics are neglected, this is justified by the fact that the manoeuvre and proximity have short duration (around 200 to 800 seconds).
- The state and inertia properties of the chaser and the target are known.
- The target has a specified grasping point (or grappling fixture).
- The manipulator's initial configuration as the base spacecraft initial state is known and the close proximity manoeuvre is already executed.

CHAPTER 2

2 Actuation system design.

In this chapter the approach to design the GNC architecture is introduced, and a preliminary design of the propulsion system is treated. In the figure 2.1 [14] is depicted a block diagram representative of the GNC structure, defined by an online logic, as shown the guidance function allow to generate the reference signal which is fed to the control function together with the feedback signal coming from the field obtained using an estimation algorithm from the sensors. Furthermore, the reference and the feedback allow to compute the error and through the designed controller the control law is obtained. The control law is then sent to the thruster allocation function allowing the computation of the ON/OFF modulated signal to open the control valve and to exert the control action using an external force. Lastly, a block of the mechanical model is reported, and it's entirely modelled in Simscape multibody.

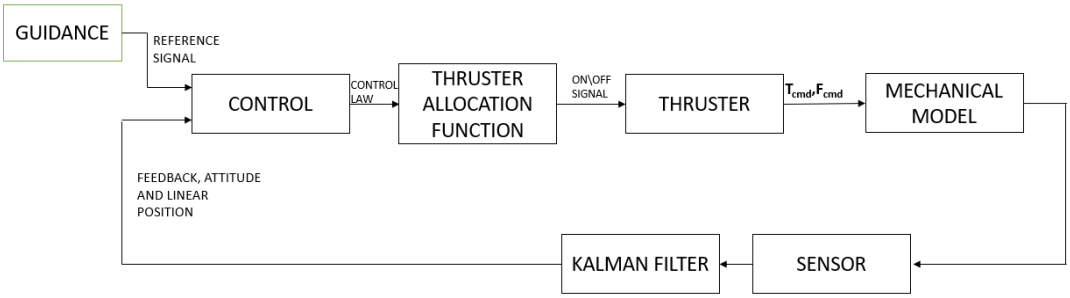

Figure 2.1: GNC design approach.

The GNC structure will be deeply discussed in chapter 3. In this section is going to be deeply discussed the design of the thruster subsystem, therefore the approach used need to be introduced. In the figure 2.2 is reported a scheme where is shown the control logic for the actuation system, the LQR generates the control signal which is sent to the thruster mapping function where the selection of the path is carried out addressing the control signal to the correct thruster. PWPFM will generate the ON/OFF signal for the control valve of the thruster.

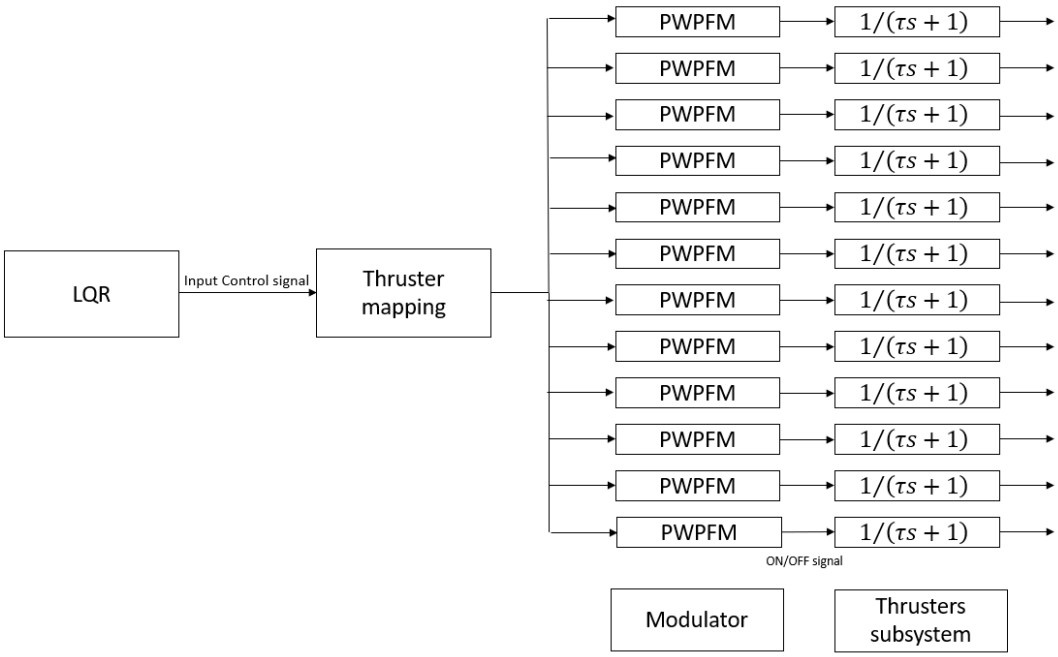

Figure 2.2: Thruster's scheme.

The first step in this thesis was to design the actuation system for the attitude and position stabilization of the spacecraft. There are different kind of actuation systems, they can be divided in two main categories:

- Reaction-type actuators.
- Momentum exchange actuators.

The first class generate torques and forces that can be considered to be external to the spacecraft, for example thrusters and magnetic torquers. The second class generate torques that can be considered internal to the spacecraft, such as Reaction wheels and control gyros.

Thrusters generate both forces and torques, so they can be used for both position and attitude control.

In this thesis work two types of actuation systems are designed to provide the spacecraft stabilization during the robotic arm operation. One of those is formed by a propulsion system consisting of 12 thrusters appropriately arranged and selected, in this case, the attitude and position control are coupled, therefore, a certain thruster combination need to be chosen to perform a manoeuvre or to keep an equilibrium state of the system.

The other stabilization system uses 3 reaction wheels for the attitude control and thrusters for the position control.

2.1 Propulsion system design

The task of a thruster propulsion system is to provide forces and torques acting on the body of the spacecraft, thus enabling changes in its translatory and angular velocities or momentum. Spacecraft thruster propulsion system can be divided into three categories, cold gas, chemical (solid and liquid) and electrical. A thruster develops its thrust by expelling propellant (such as gas molecules or ions) at a high

exhaust velocity relative to the satellite body. When thrusters are used as actuators, it is important to consider the thrust level, the required numbers and their arrangement or configurations in the body frame of the spacecraft. Generally, six or more thrusters are used to complete a reaction control system taking into consideration the failure and fault tolerant aspects like redundancy. The level of torque that a reaction thruster can apply about a satellite axis depends not only on the thrust level but also on the torque-arm length about the axis. It means that correct thruster use depends primarily on its location on the satellite and its inclination to the satellite body axis. As different torque levels are needed about three principal body axes, so the locations of the thruster are carefully studied before a final physical setup is adopted. The location and direction of the thruster is also influenced by the location of the optical sensors and solar panels to avoid damage by thruster plume. Once the location and the canting angle of each thruster is decided along with the thrust level, the torque components applied by a thruster about each body axis, which is a function of the thruster's location and direction denoted by the elevation and azimuth angle are computed.

To model the thruster, the total thruster force F_{thr_i} of the i-th thruster is described as [101]:

$$F_{thr_i} = \beta_i T_{mag_i} d_{thr_i} \quad i = 1, \dots, N, \tag{2.1}$$

Defining N_{thr} as the total number of thrusters, the vector $\beta_i \in \mathbb{R}^{N_{thr}}$ is a Boolean vector related to the thruster switching on/off dimension equal to the number of thrusters. The magnitude $T_{mag_i} \in \mathbb{R}$ of the force applied by the i-th thruster is modelled considering the maximum thrust available. The vector $d_{thr_i} \in \mathbb{R}^3$ is the vector representing the shoot direction of the i-th thruster.

When more than one thruster is fired, the torques in each axis due to the individual contribution from each thruster is added to obtain the total torque along each axis of the spacecraft.

The resulting total force applied by thrusters is given by:

$$F_{thr} = \sum_{i=1}^{N_{thr}} F_{thr_i} \tag{2.2}$$

The total moment due to the thrusters is evaluated as

$$M_{thr} = \sum_{i=1}^{N_{thr}} F_{thr_i} \wedge li \tag{2.3}$$

Two thrusters are needed in order to be able to produce a negative or positive torque about a single axis, so that we can eliminate the translational effect.

Since thrusters have an ON\OFF behaviour, we need to convert the control laws which are feed from the controllers, into a modulated signal to generate a square wave signal, the two major approaches are bang-bang, and pulse modulation. Bang-bang control is simple in formulation but results in excessive thruster actions causing more consumption of valuable fuel. Also, its discontinuous control actions often

interact with the flexible mode of the spacecraft. Therefore, it is not commonly used and hence its modelling is not discussed here.
On the other hand, pulse modulators are commonly employed due to their advantages of reduced propellant consumption and near linear duty cycle. In general, pulse modulators produce a pulse command sequence to the thruster valves by adjusting pulse width and/or pulse frequency. Pulse modulators such as pseudo rate modulator, integral-pulse frequency modulator, and pulse-width pulse frequency modulator (PWPFM) or PWM are also found in literature. Among these, the PWPFM holds several superior advantages such as close to linear operation, high accuracy, etc. and has been used in many satellites like INTELSAT, INSAT and ARABSAT [27].
The design of the thruster system has been made using the fallowing approach simplified in a block diagram:

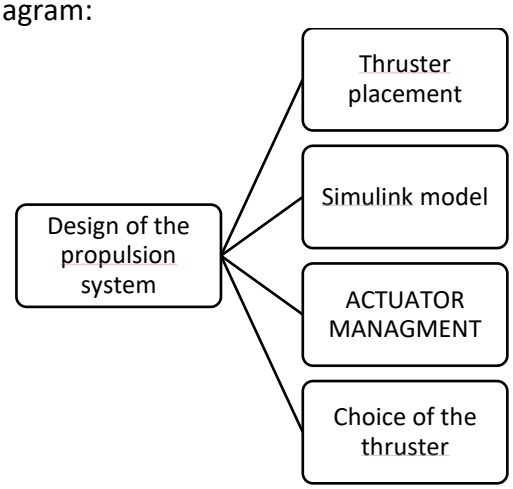

Figure 2.3 : Design approach for actuation system.

2.1.1 Thruster choice

In this thesis the mechanical design of the thruster is neglected, and a black box approach has been adopted, thus, neglecting the full dynamics of the thruster.
To allow the modelling of the thruster is necessary to specify the main parameters of the thruster.
The choice of the thrusters is usually based on the space-mission plan and must be taken in account the fallowing characteristics: maximum mass of the spacecraft, volume, required power, performance, and costs.
As reported in [26] many satellites have some kind of propulsion, which is used in the Attitude Control System (ACS). A cold gas propulsion system has the potential to fulfil this purpose. Its simple design and re-liability make it a cost-effective solution. It is fuelled by gaseous propellant delivered directly into the nozzle, where it expands to generate thrust. The ACS system can perform various rotational manoeuvres, enabling the satellite to be oriented in a specific di-rection or rotated on command into a precise angular position. For instance, these manoeuvres are required to position the satellite's antenna for communication with Earth or to set optical apparatus to follow an object in space or on Earth.
Considering all type of propulsion system, electric and chemical, the cold gas thruster is the most reliable thruster for the spacecraft and can be miniaturized to meet small satellites' design requirements.

The cold gas propulsion system has many advantages. It is small and lightweight, which are very desirable features in the space industry.
On the negative side, cold gas propulsion generates low thrust and low specific impulse, defined as the ratio of thrust (in Newtons) to mass flow rate (in kilograms per seconds). The thrust achievable is only up to 10 N [28] the propellant available is limited by the volume of the tank and the propellant's physical state. The cold propulsion system usually consists of control valves, sensors, several nozzles, and pressurized gaseous propellant stored in a tank. In the fallowing figure the interaction between the different element is shown:

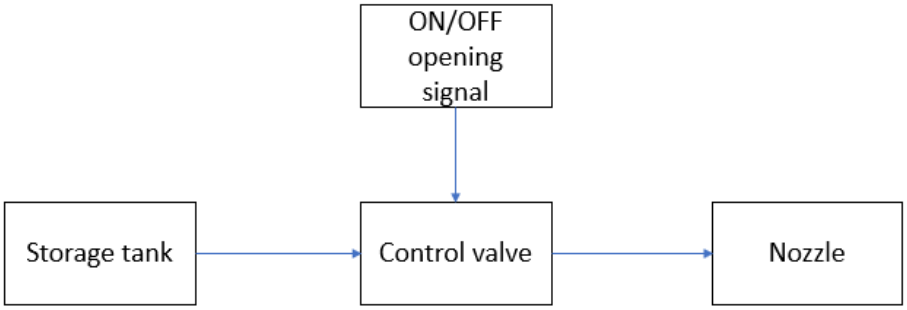

Figure 2.4: Thruster black box representation

Various gases can be considered as candidates for the propellant, the most popular being nitrogen, oxygen, hydrogen, and helium. The major advantage of helium and hydrogen is their much higher specific impulse relative to the others. Following the definition of specific impulse, the same thrust is achievable using a lower amount of propellant. However, helium and hydrogen have small molecules, which increase the chances of leakage, and very low density. Moreover, a larger, heavier storage tank is needed to store the same propellant mass. Hydrazine was selected as the propellant gas in this project, as it is easy to handle and has a good combination of specific impulse and density. Table 2.1 [28] shows a comparison of characteristics of various gases.

Working medium	**Molecular weight**	**Density [kg/m^3] at 100kPa, 288.15K**	**Specific impulse [m\s]**
H_2	2.106	0.084	2786
He_2	4.003	0.167	1756
N_2	28.0134	1.167	746
Air	28.96	1.213	726

Table 2-1:Propellent type.

In order to design a cold gas propulsion system, it is important to find out the ΔV requirements for the maneuvers. The ΔV required can be calculated through Homann transfer governing equation [36].
Tsiolkowski equation and its corollaries are used to convert these velocity change requirements into propellant requirements:

$$\Delta V = g_c I_{sp} \ln\left(\frac{W_i}{W_f}\right) \tag{2.4}$$

$$W_f = W_i \left[1 - \exp\left(-\frac{\Delta V}{g_c I_{sp}}\right)\right] \tag{2.5}$$

Where I_{sp} is the specific impulse, W_i and W_f are the initial and final mass and g_c is the gravitational effects.
The maximum thrust that the thruster can achieve can be selected referring to what is written in the literature [28, 29].
According to this information the parameters selected for the thruster are reported in the fallowing table 2.2.

Propellant	Hydrazine
Maximum thrust level	1 N
Reference thruster	CGT20 sterer

Table 2-2: Main parameter

The process of designing a thruster is not the ai of this thesis, therefore the assumption has been made to analyse the ADR from a dynamic point of view, therefore we are interested in a preliminary choice of the thruster to set he appropriate thrust level. For the choice of the thruster can be even taken into consideration the paper [77].

2.1.2 Thruster placement

As said earlier the Reaction Control Thrusters System is composed by 12 cold gas thrusters arranged in the configuration depicted in figure 2.5 [101], which is an anti-symmetric thruster. The adopted configuration allows the use of the thruster's system to generate both pure force and pure torque along a desired axis. To generate pure force, it is required to activate the couple of thrusters firing in the same direction to remove the torque effect; instead, to generate pure torque, it is required to activate the couple of thrusters firing in opposite direction. For example, to generate pure force along the +XBODY axis it is necessary to activate thrusters 1x and 3x, while to generate pure torque along +YBODY it is necessary to activate thrusters 1y and 3y. Therefore, is possible to obtain the tables 2-3 and 2-4 where with 1 is indicated if a certain thruster gives a contribution in a specific direction, with zero if this contribution is null.

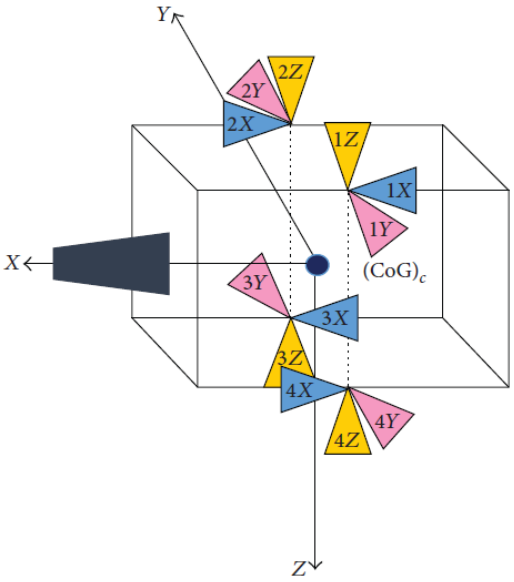

Figure 2.5: Thruster configuration.

In Table 2.2 and 2.3 are summarized the nominal firing direction:

Torque	1x	2x	3x	4x	1y	2y	3y	4y	1z	2z	3z	4z
Tx+(Txp)	0	0	0	0	0	0	0	0	0	1	0	1
Tx-(Txn)	0	0	0	0	0	0	0	0	1	0	1	0
Ty+(Typ)	0	1	1	0	0	0	0	0	0	0	0	0
Ty-(Tyn)	1	0	0	1	0	0	0	0	0	0	0	0
Tz+(Tzp)	1	1	0	0	0	0	0	0	0	0	0	0
Tz-(Tzn)	0	0	1	1	0	0	0	0	0	0	0	0

Table 2-3: Torque selection.

Force	1x	2x	3x	4x	1y	2y	3y	4y	1z	2z	3z	4z
Fx+(Fxp)	1	0	1	0	0	0	0	0	0	0	0	0
Fx-(Fxn)	0	1	0	1	0	0	0	0	0	0	0	0
Fy+(Fyp)	0	0	0	0	0	1	1	0	0	0	0	0
Fy-(Fyn)	0	0	0	0	1	0	0	1	0	0	0	0
Fz+(Fzp)	0	0	0	0	0	0	0	0	1	1	0	0
Fz-(Fzn)	0	0	0	0	0	0	0	0	0	0	1	1

Table 2-4: Force selection.

Given the thruster configuration, is possible to observe that they are expressed in the geometric reference frame, but to use them to manage the roto-translational dynamics of the spacecraft, they must be projection in the body frame, thus we need to express their position in the body frame. This will be carried out in the management section.

2.1.3 Actuator management function

The task of the thruster management function is to translate the force and torque commands generated by the control function into 'on/off' commands for the individual thrusters according to their direction and to their location w.r.t. the momentary centre of mass of the vehicle. Depending upon the control error, a request for a force of varying level to be applied along the individual axes is made. The position controller generates force commands, and the attitude controller generates torque commands, of varying amplitude along and around the body axes. So, this function is computed through two aspects: The thruster selection logic and the PWPFM.

2.1.3.1 Thruster allocation function

The main task of thruster control allocation is to select specific thrusters and calculate their firing durations to realize force and torque commands derived from the control system of a spacecraft. It has a direct effect on the control accuracy and fuel consumption of the complicated space missions. So far, there are three main approaches for the thruster control allocation problem: the decoupling method, the linear programming (LP) method and the optimal catalogue method. The decoupling

method divides thrusters into combinations according to predefined manoeuvres and control modes of a spacecraft. Control systems of many spacecrafts are based on this method for its simplicity in onboard computations. However, this method is only applicable for regular thruster configurations which results in a limitation on the thruster configuration design. Besides, it cannot minimize propellant consumption, which is important for extending the mission-life time. Thereupon, the LP method was proposed based on the simplex method [2],[5]-[7]. Although it can find the optimum solution to the thruster control allocation problem, the processing power needed by this algorithm was too high for the on-board processing budget and is thus not often considered for real time application in space.

A. General statement of the problem.

The rotational and translational dynamics are decoupled if the controls $M_C^S$and F_C^S are independent, where S is the spacecraft reference frame. However, in the thrusters-only case, the attitude control and the translational control are coupled and cannot be designed separately.

The required directional force F_C^S and torque M_C^Smust be produced by the combined firing of the thrusters, thus these control variables are not independent.

Let $u = u_a[u_1\ u_2 \dots\ u_n]^T = u_a U$ be the vector of thrusts of the thrusters, where:

$$u_i = \begin{cases} = 1\ if\ ith\ thruster\ on \\ = 0\ if\ ith\ thruster\ off \end{cases}$$

$$i = 1,2, \dots n.$$

Generally, the thruster control allocation problem is formulated as follows:

$$F_{LQR} = \begin{bmatrix} F_C^S \\ M_C^S \end{bmatrix} = \begin{bmatrix} H_F \\ H_M \end{bmatrix} \underline{u} = H\underline{u} = Hu_a U \tag{2.6}$$

Where H is the $6 \times n$ thrust distribution matrix or structure matrix related to the geometrical structure of the thruster's placement on the spacecraft, where n is the number of thrusters.

B. Thrust distribution matrix.

The *H* matrix must allow for forces and torques generation capability around the three axes of the body reference frame, a full rank requirement, otherwise a thruster configuration $\underline{u}_0 \neq 0$ exists so that $H\underline{u}_0 = 0$. Moreover, the *H* matrix satisfies the following properties [64]:

1) Guarantees existence of a configuration $\underline{u}_F$, such that $H_F\underline{u}_F \neq 0$ and $H_M\underline{u}_F = 0$.
2) Guarantees existence of a configuration $\underline{u}_M$, such that $H_M\underline{u}_M = 0$.and $H_F\underline{u}_M \neq 0$.

The above-mentioned properties guarantee the capability to generate pure forces without torques and vice versa. These are intrinsic properties of the geometrical placement of the thrusters.

Note that this is true only from a theoretical point of view, since systematic errors in the thruster's placement and orientation, and slight differences of the nominal thrust among the different thrusters introduce undesired disturbance forces and torques for a given manoeuvre.
To obtain the H matrix we need to consider the tables reported in section 2.1.2, from which we can extract the fallowing matrices:

$$H_F = \begin{bmatrix} 1 & 0 & 0 & -1 & 0 & 0 & 1 & 0 & 0 & -1 & 0 & 0 \\ 0 & 1 & 0 & 0 & -1 & 0 & 0 & -1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & -1 & 0 & 0 & -1 \end{bmatrix}$$

$$H_M = \begin{bmatrix} 0 & 0 & -l & 0 & 0 & l & 0 & 0 & -l & 0 & 0 & l \\ 0 & 1 & 0 & 0 & -l & 0 & 0 & -l & 0 & 0 & 1 & 0 \\ 0 & 0 & l & 0 & 0 & 1 & 0 & 0 & -l & 0 & 0 & -l \end{bmatrix}$$

Where l is the distance of the thruster from the centre of mass.
These are expressed in the geometric frame, so they need to be reported in the body frame, this can be done by multiplying those for the augmented rotation matrix that express the orientation of the geometric frame with respect to the body frame.

$$\tilde{R}_G^B = \begin{pmatrix} R_G^B & 0_{3\times3} \\ 0_{3\times3} & R_G^B \end{pmatrix}$$

$$with\ R_G^B = \begin{bmatrix} 1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & -1 \end{bmatrix}$$

Where G stands for geometric frame and B for body reference frame.

$$H^B = \tilde{R}_G^B H^G \tag{2.7}$$

Since the orientation of the spacecraft changes during manoeuvring operation of the spacecraft, the distribution matrix needs to update with the actual attitude information, this can be obtained through the multiplication with the direct cosine matrix expressing the orientation of the body frame in the inertial frame:

$$H = \begin{bmatrix} DCM_B^I(X_B)H_F \\ H_M \end{bmatrix} \tag{2.8}$$

Where I is the inertial frame and X_B is the actual state variable of the chaser in terms of rotation.

C. Control design approach

The classical control design for spacecraft with only thrusters is typically conducted by solving the control problem for the roto-translational dynamic, considering the space vehicle actuated by ON/OFF actuators. This approach leads to finding the continuous control law M_C^B and F_C^B subsequently, the suitable configuration of thrusters is found such that is met with a minimum number of thrusters.
This problem is usually solved by applying simplex-based algorithms, as shown in [32, 36], we can solve it through the pseudo-inverse of the configuration matrix, this result

can be obtained by solving and optimization problem using the Lagrange multipliers method, in the end the following result is obtained:

$$U = \frac{pinv(H)F_{LQR}}{T_{thrust}} \tag{2.9}$$

Obtaining the 12 normalized continuous input which need to be sent to the PWPFM where a square signal is produce.

D. Simulink application

Considering the previous points, since the thruster system can only manage positive values, because the thrust in mono-directional, the code in the Simulink model is implemented with the augmented matrices, using the following logic:

$$F_{LQR} = \begin{bmatrix} F_C^S \\ M_C^S \end{bmatrix} = \begin{bmatrix} H_F \\ H_M \end{bmatrix} \underline{u} \tag{2.10}$$

We obtain from the previous relationship:

$$F_{LQR} = \begin{bmatrix} F_x \\ F_y \\ F_z \\ M_x \\ M_y \\ M_z \end{bmatrix}$$

These are the control laws generated from the LQR control block, those are sent to the thruster allocation system where the augmented matrix is obtained:

$$\tilde{F}_{LQR} = \begin{bmatrix} F_{x+} \\ F_{x-} \\ F_{y+} \\ F_{y-} \\ F_{z+} \\ F_{z-} \\ M_{x+} \\ M_{x-} \\ M_{y+} \\ M_{y-} \\ M_{z+} \\ M_{z-} \end{bmatrix} = \tilde{H}\tilde{u} \tag{2.11}$$

Here all the terms are positive value, notably $\tilde{F}_{LQR} \in \mathbb{R}^{12\times1}$ is a 12x1 matrix, exactly the number of thrusters mounted on the spacecraft, therefore, the $\widetilde{H} \in \mathbb{R}^{12\times12}$ and $u \in \mathbb{R}^{12\times1}$, obtaining so 12 control laws for the thrusters. Moreover, taking into account H, we observe the following matrix:

$$H = \begin{bmatrix} 1 & 0 & 0 & -1 & 0 & 0 & 1 & 0 & 0 & -1 & 0 & 0 \\ 0 & -1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & -1 & 0 \\ 0 & 0 & -1 & 0 & 0 & -1 & 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & -1 & 0 & 0 & 1 & 0 & 0 & -1 & 0 & 0 & 1 \\ 1 & 0 & 0 & -1 & 0 & 0 & 0 & -1 & 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & -1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \end{bmatrix}$$

Where $l = 1$, computing the pseudo-inverse and computing the augmented matrix we obtain a matrix where the ith raw represents for a specific request of linear force or torque the thrusters that needs to be turn on, and the ith column represents the contribution of a certain thruster for a request of force and torque.
While in the pseudo-inverse, every raw represents a certain thruster, while each columns represent a degree of freedom. Does important to consider that in the following expression:

$$\tilde{u} = pinv(H)F_{LQR} \tag{2.12}$$

The vector $\tilde{u}$ is defined as follows:

$$\tilde{u} = \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \\ u_6 \\ u_7 \\ u_8 \\ u_9 \\ u_{10} \\ u_{11} \\ u_{12} \end{bmatrix}$$

Where the assignment is the following one:

$$1x: u_1 = T_{thrust} * \hat{\imath}$$
$$1y: u_2 = T_{thrust} * \hat{\jmath}$$
$$1z: u_3 = T_{thrust} * \hat{k}$$
$$2x: u_4 = T_{thrust} * \hat{\imath}$$
$$2y: u_5 = T_{thrust} * \hat{\jmath}$$
$$2z: u_6 = T_{thrust} * \hat{k}$$
$$3x: u_7 = T_{thrust} * \hat{\imath}$$
$$3y: u_8 = T_{thrust} * \hat{\jmath}$$
$$3z: u_9 = T_{thrust} * \hat{k}$$
$$4x: u_{10} = T_{thrust} * \hat{\imath}$$
$$4y: u_{11} = T_{thrust} * \hat{\jmath}$$
$$4z: u_{12} = T_{thrust} * \hat{k}$$

Where $\hat{\imath}, \hat{\jmath}, \hat{k}$ are verses of the body reference frame, and the right sign is computed in H matrix.
So, after these considerations, can be explained that the signal contained in the $\tilde{F}_{LQR}$ feed the allocation subsystem, where the pseudoinverse is solved and the request of forces and torques are allocated to the right thrusters, so that if a thrust along the positive x is required F_{x+}, the thruster 1 and 7 (1x and 3x) are turned on, the same logic can be applied to the system. The same result can be obtained observing the following relationships:

$$\begin{bmatrix} F_x \\ F_y \\ F_z \\ M_x \\ M_y \\ M_z \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & -1 & 0 & 0 & 1 & 0 & 0 & -1 & 0 & 0 \\ 0 & -1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & -1 & 0 \\ 0 & 0 & -1 & 0 & 0 & -1 & 0 & 0 & 1 & 0 & 0 & 1 \\ 0 & 0 & -1 & 0 & 0 & 1 & 0 & 0 & -1 & 0 & 0 & 1 \\ 1 & 0 & 0 & -1 & 0 & 0 & 0 & -1 & 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & -1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \\ u_5 \\ u_6 \\ u_7 \\ u_8 \\ u_9 \\ u_{10} \\ u_{11} \\ u_{12} \end{bmatrix} \quad (2.13)$$

We obtain then a matrix, in this case the body frame coincides with the inertial frame:

$$\tilde{H} = \begin{bmatrix} 0.25 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.25 & 0 & 0 & 0.25 \\ 0 & 0 & 0 & 0.25 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.25 & 0 & 0.25 & 0 & 0 & 0 & 0 \\ 0 & 0.25 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.25 & 0 & 0.25 \\ 0 & 0 & 0.25 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.25 & 0.25 & 0 & 0 & 0 & 0 & 0 \\ 0.25 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0.25 & 0.25 & 0 \\ 0 & 0 & 025 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.25 & 0 & 0 & 0.25 & 0 & 0 & 0 & 0 \\ 0 & 0.25 & 0 & 0 & 0 & 0 & 0 & 0 & 0.25 & 0 & 0.25 & 0 \\ 0 & 0 & 0 & 0.25 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.25 & 0 & 0.25 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$\tilde{H}\tilde{F}_{LQR} = \tilde{u} \quad (2.14)$$

As can be seen, if F_{x+} is required along the positive axes of x, the first and the 7th thrusters, which corresponds to 1x and 3x, are fed, so in this case the raw correspond to a specific thruster and the column to a specific DOF.

2.1.4 Thruster Simulink model

The twelve thrusters are ideally fixed in their position and perfectly distributed with respect to the centre of mass, so there is no residual torque error due to misalignment. The ON/OFF thrusters switching time is modelled applying a transfer function of the first order to the PWPFM output signal of each thruster [81]:

$$F(s) = \frac{T_{thrust}}{\tau s + 1} \quad (2.15)$$

Since the ON communication time is equal to 8 ms, the coefficient τ indicates that the 98% of the thrust is reached after 8 ms, an example of he thrust characteristics is shown in figure 2.6 [81].
Then the thrust is transferred to the multibody system using an external force block which acts over the spacecraft through reference frame appropriately defined.

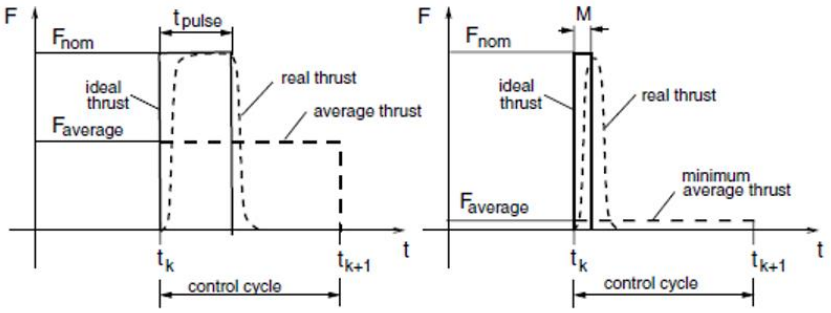

Figure 2.6: Thruster pulse characteristics.

2.2 Reaction wheels

The Reaction Wheels System is the actuation system used to control the attitude of the chaser spacecraft. The system is composed by a number of N reaction wheels arranged with different types of configurations, depending by the specific application, control strategy, redundancy, and more. It is compound by a massive disk connected to an electric motor controlled in both torque and velocity: to accelerate the wheel, the motor apply a torque to the disk and, since the wheel assembly is mechanically connected to the entire spacecraft structure, the reaction torque of the accelerating disk (equal in magnitude but with opposite direction) is transmitted to the motor and hence it is transmitted to the spacecraft, which reacts modifying its attitude. The reaction wheel is subjected to two main saturation conditions: torque saturation and velocity saturation. The torque saturation is the maximum torque the wheel can provide it is a physical limitation of the maximum current that flows into the electric motor before damaging it. Usually, the torque saturation is implemented in the control law, in such a way that the torque required by the controller should never exceed the maximum torque provided by the wheel, preventing damaging of the electric motor. The velocity saturation is the maximum angular velocity that can be sustained by bearings of the electric motor. If the velocity is greater than the maximum allowable, the bearings may be damaged and makes the wheel unusable. Since the wheel has reached the maximum velocity it cannot accelerate anymore, then the torque provided in this condition is null. In order to make the wheel usable again, it is necessary to slow down the disk, reducing the velocity. Decelerating the disk means applying a torque opposite to the rotation, but this will cause a modification of the attitude of the spacecraft due to the reaction torque, hence it is necessary to apply an external torque to counteract the torque required to decelerate the wheel. This external torque is provided by the thruster's system for almost all the typologies of spacecraft, except for the recent CubeSat typology which use magneto torquers to de-saturate reaction wheels.

2.2.1 Reaction wheel distribution matrix

The distribution matrix of reaction wheels, which is uniquely decided by reaction wheel configuration, consists of n columns corresponding to n reaction wheels. Each column vector represents the distribution of the reaction wheel torques on to the axes of rotation of the satellite. The reaction wheels configuration choose is the standard configuration.

- **Standard configuration.**

If T_1, T_2, T_3 represent the torques produced by each reaction wheel, then the moments acted on the satellite can be defined as:

$$\begin{bmatrix} T_x \\ T_y \\ T_z \end{bmatrix} = L_{3\times3} \begin{bmatrix} T_1 \\ T_2 \\ T_3 \end{bmatrix} \tag{2.16}$$

The distribution matrix of standard orthogonal 3-wheel configuration can be defined as:

$$L_{3\times3} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

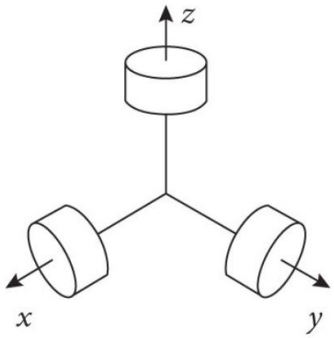

Figure 2.7: Reatcion wheel configuation.

2.2.2 Reaction wheel Simulink model.

As reported above, the reaction wheels consist of an electric motor and a wheel with a specific inertia, so for the representation in a Simulink model let us write down the equation to exploit the dynamic of the system.
For the electric motor we have the followings formulation:

$$V_a - K_e\left(\dot{\theta}_{RW} - \dot{\theta}_{sc}\right) = L_a \frac{di}{dt} + R_a i_a \tag{2.17}$$

$$T = K_t i_a \tag{2.18}$$

The workload of the DC-motor consists of a reaction wheel mounted on the rotor shaft, resulting in the following workload torque:

$$T_{RW} = I_{RW}\ddot{\theta}_{RW} \tag{2.19}$$

Where:

- V_a : Armature Voltage (V);
- K_e : Voltage constant of DC motor;
- R_a : Armature resistance (Ω);
- i_a : Current armature (A) ;
- L_a: Inductance (H);
- $\dot{\theta}_m$: angular acceleration of the motor (rad/s^2);
- $\ddot{\theta}_{RW}$: angular acceleration of the wheel (rad/s^2);
- $\dot{\theta}_{sc}$: Spacecraft angular velocity (rad/s);
- T, T_{RW}: Torque respectively, in output from the motor and in input to the RW (Nm);
- K_t: Torque constant (Nm);
- I_{RW}: RW inertia (Kgm^2);

The rotor shaft of the motor, as a matter of course, has the same angular velocity and acceleration with the reaction wheel which is fixed to the rotor shaft. Therefore $\ddot{\vartheta} = \ddot{\theta}_{RW}$ and $\dot{\theta}_m = \dot{\theta}_{RW}$. The moment balance of the rotor shaft is given as

$$T - T_{RW} = I_m\ddot{\theta}_m \tag{2.20}$$

Where I_m and $\ddot{\theta}_m$ are the inertia angular acceleration of the motor. Substituting the equation of T_{RW} into the previous one, results in:

$$T = (I_m + I_{RW})\ddot{\theta}_m \tag{2.21}$$

Since $I_{RW} \gg I_m$, can be obtained the following equation for the motor Torque:

$$T = I_{RW}\ddot{\theta}_{RW} \tag{2.22}$$

From these equations we can build the following block scheme representative of the system dynamics.

From figure 2.8 [65] can be seen that there is an internal current control loop, a summing node between the current command and the actual motor current produces a specific error signal for the proportional integrative controller PI. This is an integral controller with gain K*, which is fundamental to express a relation between the torque command and the actual motor torque such that the motor dynamics can be neglected. The controller output is the voltage that must be applied

to the motor. However, it is evident the presence of a voltage limitation since there is a maximum value of voltage that can be applied to the motor.

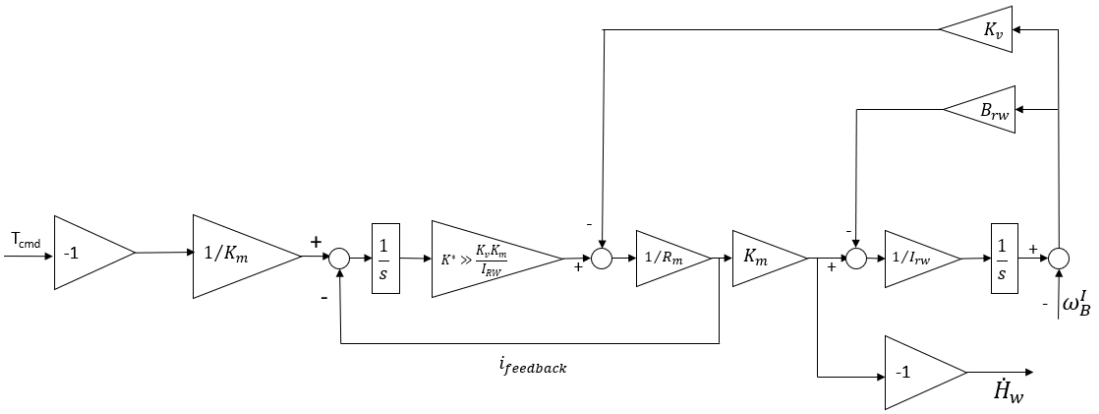

Figure 2.8: Reaction wheel model.

Now, it is important to show the transfer function that relates the torque command and actual motor torque. It is possible to write the following relation:

$$\frac{T(s)}{T_{\text{set}}(s)} = \frac{\frac{K^*}{R_M}}{s + \frac{K^*}{R_M}} = \frac{1}{1 + s\frac{R_M}{K^*}} \tag{2.23}$$

This transfer function is obtained considering $\frac{1}{I_{sc}} \ll \frac{1}{I_{RW}}$. Then, if the control gain K^* is chosen such that $K^* \gg \frac{K_v K_m}{I_{RW}}$, the following relation holds $T \approx T_{set}$ and so all motor dynamics can be neglected. The toque which is transferred to the flywheel is the one exploited from the motor.

For this thesis the RW's disturbances are neglected for simplicity, indeed for a proper design and simulation accuracy we should consider the static unbalance and the dynamic unbalance. The first source is due to the asymmetric distribution of the flywheel mass in the radial direction, which leads to an eccentricity between the centre of mass and the axis of rotation of the flywheel. The second source of disturbances depends by the inclination of the principal axes of inertia with respect to the rotation axis, caused by an asymmetric distribution of mass along the axial direction [40].

2.3 PWPFM

The pulse-width-pulse-frequency modulator is a continuous time technique, which consist of a Schmitt trigger, a first order filter and a feedback loop. It has the following advantages [39]:

- Close to linear operation [37, 38].
- A wide range of parameters to tune give the designer freedom with respect to a system-specific consideration.
- Tunability dives flexibility during the operation, the parameters can be tuned to meet different requirements through different phases of operation.
- Low fuel consumption and good pointing accuracy especially when vibrations are present.

In the following figure 2.9 [39] is reported a block diagram of the modulator:

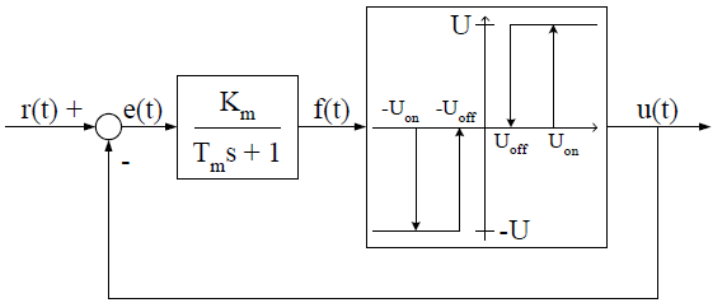

Figure 2.9: PWPFM.

Where r(t) is the reference signal and e(t) the error. Furthermore, are shown different parameters with the following meaning:

- K_m: Filter gain.
- T_m: Filter time constant.
- U_{ON}: Schmitt trigger on-value.
- U_{OFF}: Schmitt trigger off-value.
- $h = U_{ON} - U_{OFF}$

In order to design the control block, some basic properties need to be taken into account [77]:

- U_{OFF} can be seen as a dead zone, used to limit the activation time and cope with noises and spurious torque commands.
- U is the thrust, or a unit factor which is then used sent to the first order thruster.
- U_{ON}: is related to the sensitivity in thruster activation, small value mean fast commutations;
- T_m: is related with the order of magnitude of on and off times.

The presence of the ideal hysteresis introduces a higher non -linearity into the system. To tune appropriately the parameters, we can refer to the analysis in [39], where is given the fallowing table [39]:

Parameters	Analysis type			Raccomandations
	Static	Dynamic	System	
K_m	$2.5 \rightarrow 7.5$			$2.5 \rightarrow 7.5$
T_m	$0.1 \rightarrow 1.0$	$0.1 \rightarrow 1.0$	$0.1 \rightarrow 0.5$	$0.1 \rightarrow 1.0$
U_{on}	$0.1 \rightarrow 1.0$		$0.1 \rightarrow 1.0$	$0.1 \rightarrow 1.0$
h	$0.2 \rightarrow 2U_{on}$		$0.2 \rightarrow 2U_{on}$	$0.2 \rightarrow 2U_{on}$

Table 2-5: Optimal PWPFM value.

In the figure 2.10 [39] a general behaviour is reported.

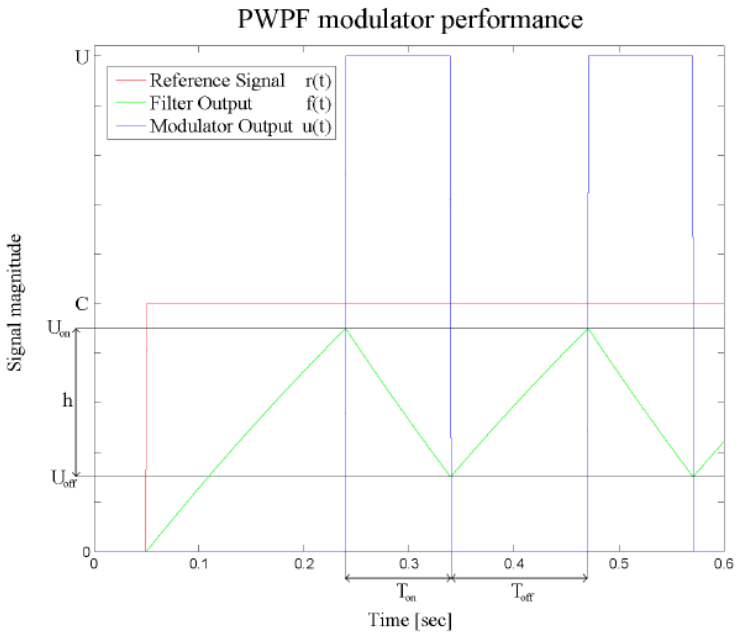

Figure 2.10: PWPFM.

Despite optimized value are reported in literature, a preliminary tuning analysing on the behaviour of the system with a parametric analysis has been conducted referring to a simplified model in simscape multibody environment. Thus, the analysis is conducted changing the parameter and observing the time response of the system, furthermore for completeness are reported the figures where the system response is obtained in terms of number of firings of the thrusters and fuel consumption.

2.3.1 Time response with respect to U_{ON} and h.

The figures reported in this section are obtained in the multibody Simulink model feeding the system with a reference roll signal equal to 0.3 radiant, the controller will try to stabilize the spacecraft around the reference signal, coinciding with the equilibrium point, through the 12 thrusters available, therefore after an initial transient, which bring the actual state to the reference signal in a specific settled time, the system will oscillate trying to minimize the static error.

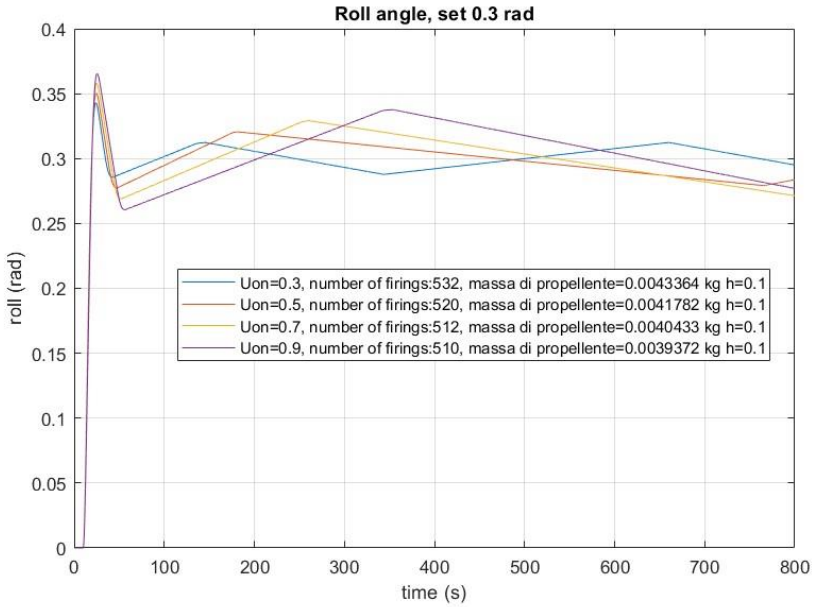

Figure 2.11.

As can be seen in the figure 2.11 increasing U_{ON} with h constant the signal increases the steady error and the overshoot as well, but the number of oscillations reduce. The same result can be observed in the figures 2.12, 2.13, 2.14.

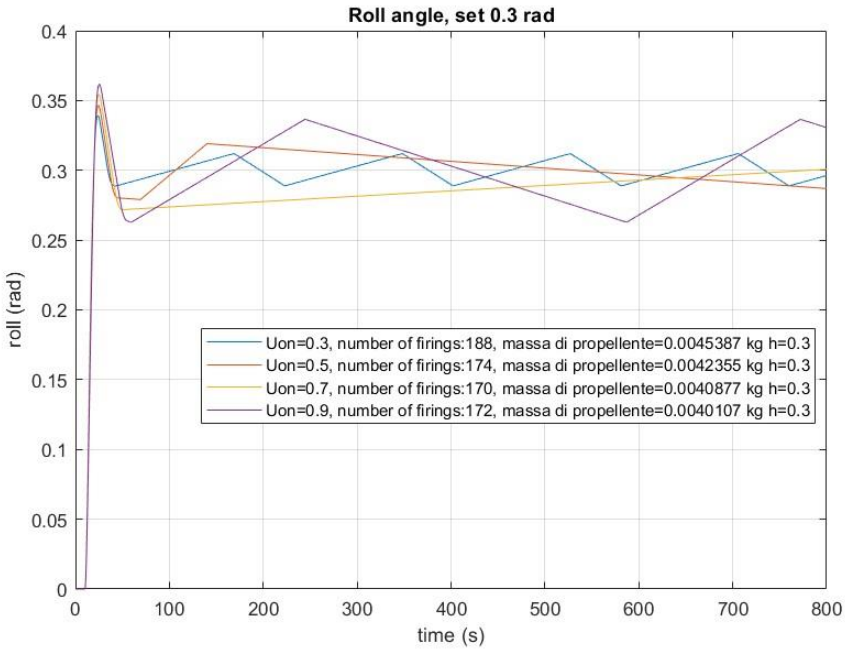

Figure 2.12

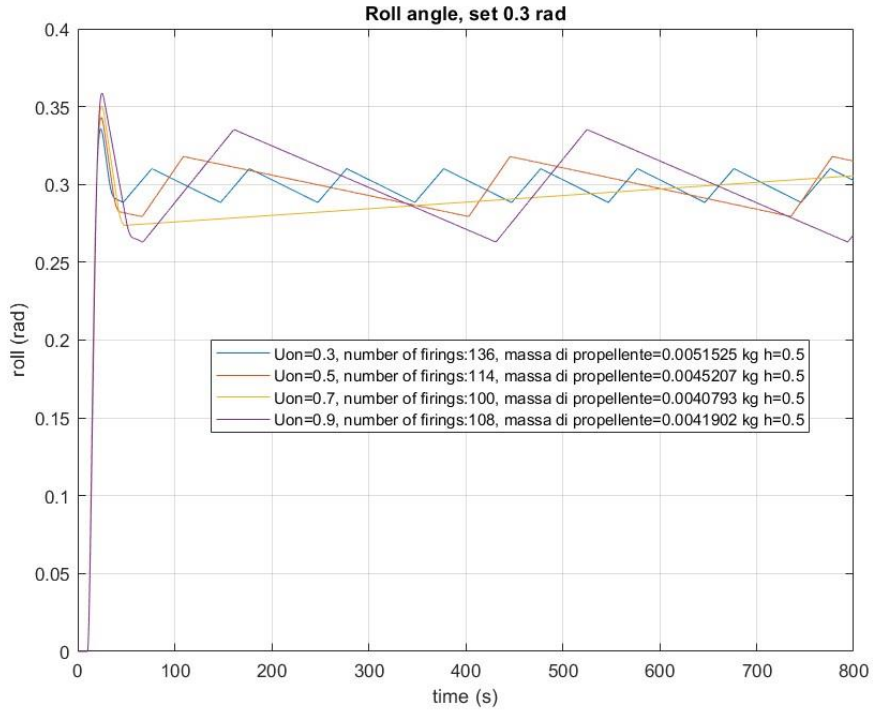

Figure 2.13

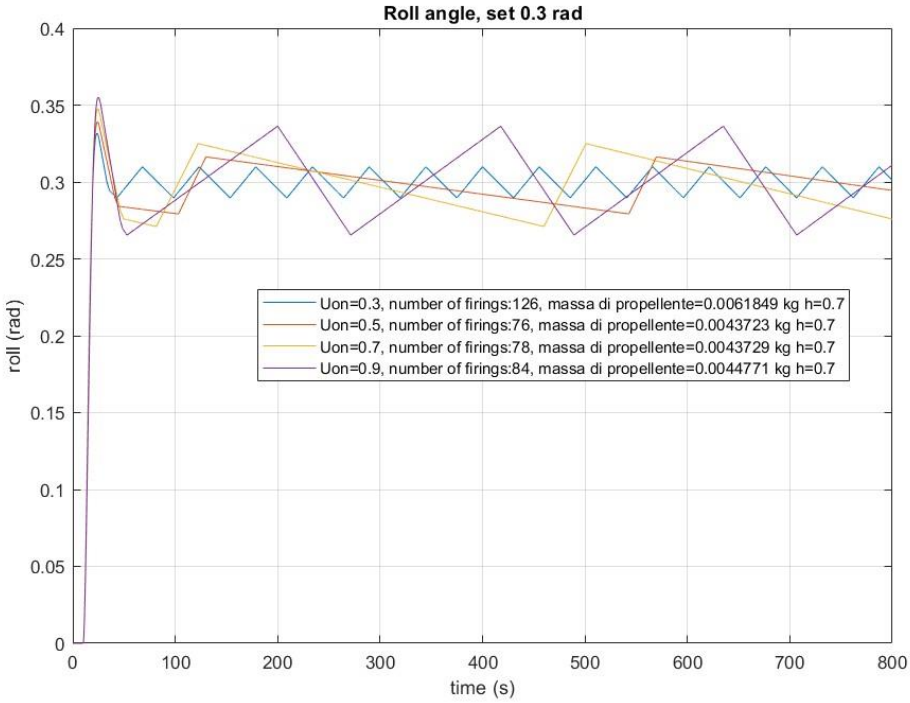

Figure 2.14

Following the same patterns and increasing h for every diagram can be observed that the that increase leads to a major number of oscillations around the reference signal, on the other side it allows the decrease of the steady state error and the number of thruster firings.

2.3.2 Time response with respect to T_f and K_f.

As in the previous section, the time response is reported here, analysing the result with respect to the variation of T_f and K_f characterizing the PWPFM behaviour.
In general, the increase of the filter gain, maintaining a constant value for T_f, leads to a major number of thruster firings and to a higher fuel consumption, however, as can be noted in the figures reported below, it allows to reduce the static error and the signal overshoot improving the stability performance of the system.
While, increasing the constant of time, there is a higher hysteresis in the filer signal PWPFM, therefore, the number of thruster firings decrease while the fuel consumption increase, since the time on of the thruster activity signal is higher than the others cases. Furthermore, is possible to observe a higher number of oscillations around the reference value.

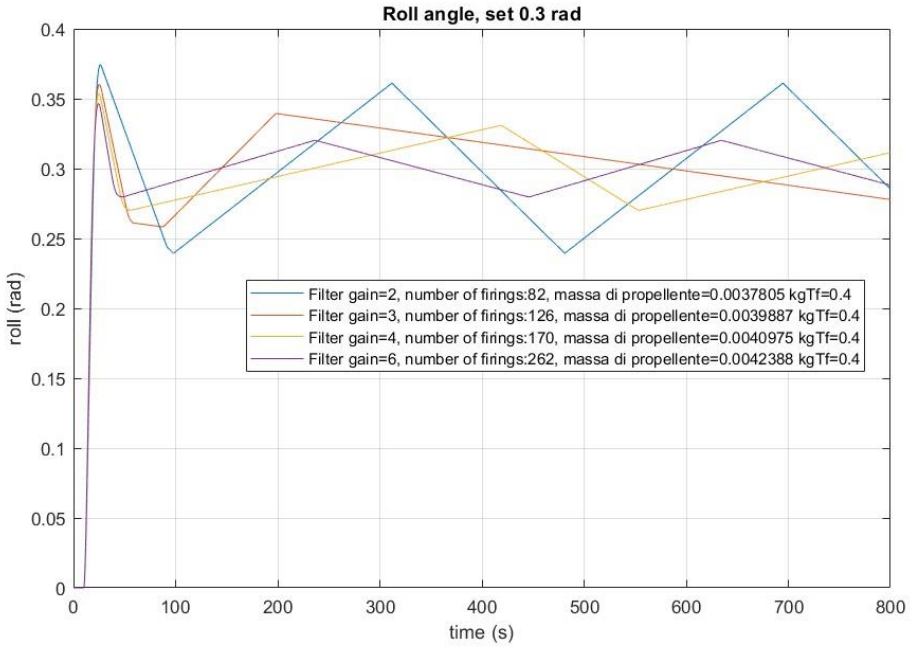

Figure 2.15

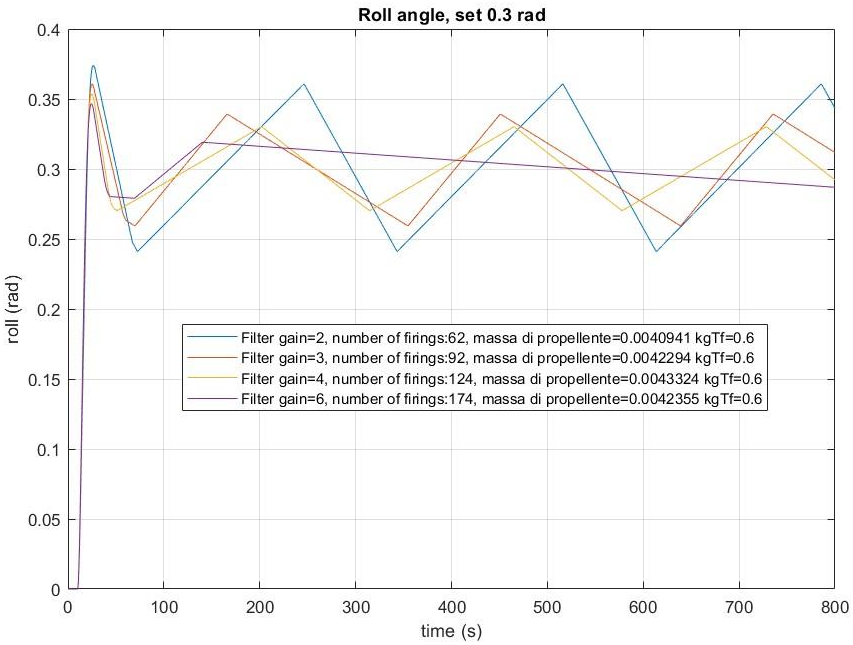

Figure 2.16

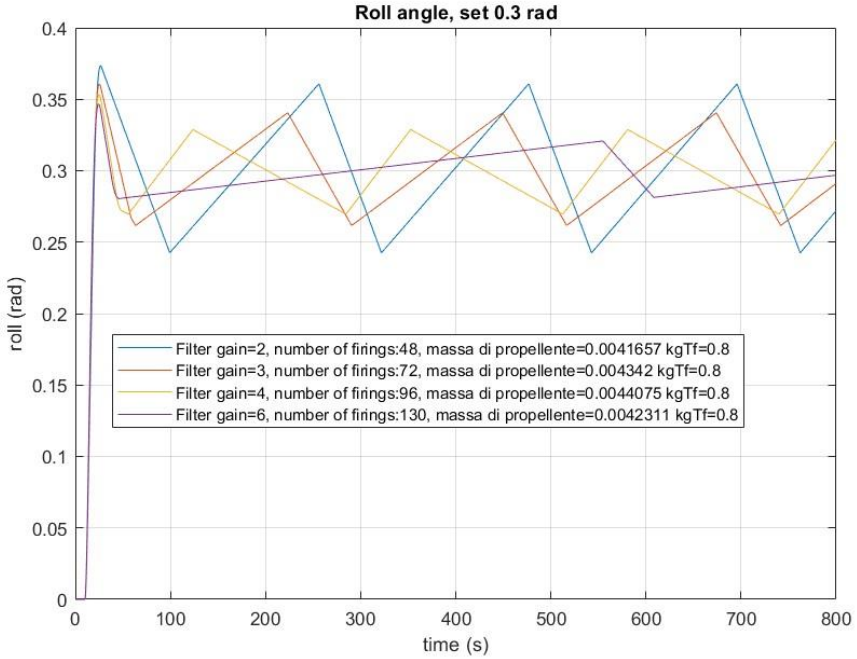

Figure 2.17

2.3.3 Optimizing with respect to thruster activity

Thruster valves are subject to the wear when they are opened and closed. To minimise the thruster valve wear, it is desirable to minimise the number of commanded thruster firings from the PWPFM modulator. First the parameter filter is tested against each other.

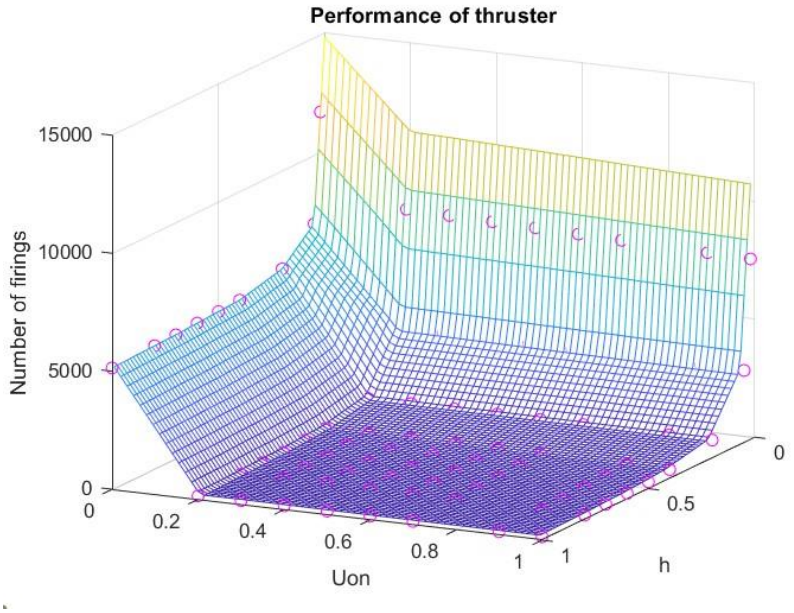

Figure 2.18: Number of firings with respect to U_{on} and h.

Higher T_f and K_f give thruster activity. For every low filter time constant the thruster activity increases dramatically.
Finally, since the dumping effect that normally give disappears almost entirely. Furthermore, we can notice that the region with high thruster activity becomes bigger as filter gain increase.
Then the number of thruster firings was examinate with respect to U and h. In the following figure can be seen that low hysteresis shell be avoided, which means h>0.2.

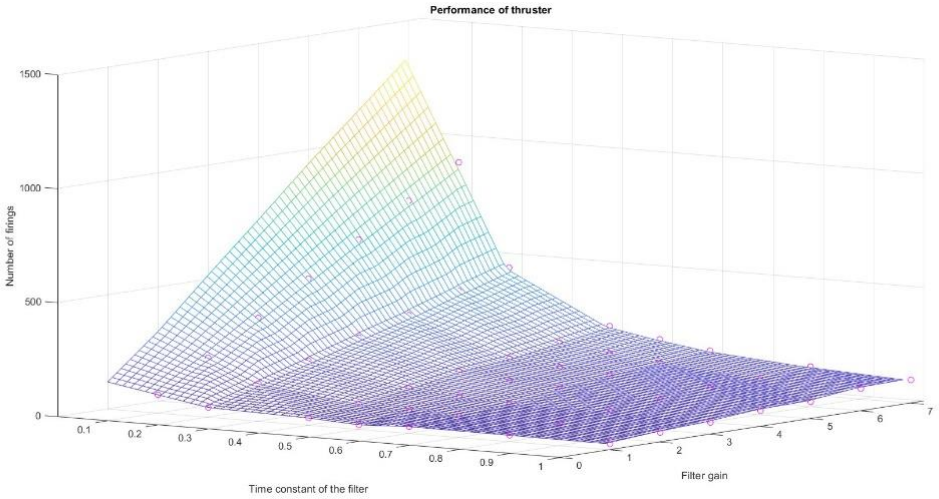

Figure 2.19: Number of firings with respect to K_f and T_f.

Can be concluded that for thruster activity the tome constant should be as large as possible.

2.3.4 Optimizing with respect to fuel consumption.

It is obvious that thruster activity and fuel consumption is closely related. But two parameter configurations that give the same fuel consumption, does not necessarily turn the thruster on and off the same amount of time. In the previous section the parameter ranges that give high thruster activity were identified, in this section, the focus is on fuel consumption.

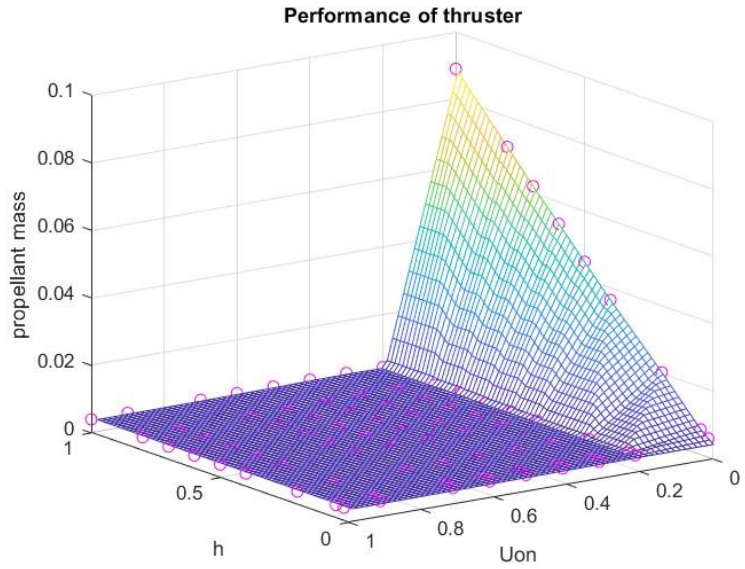

Figure 2.20: Propellant mass with respect to U_{on} and h.

First the U and h are evaluated. In the figure 2.20 is showed that shall be avoided small value of U_{ON}, since these can increase the fuel consumption.

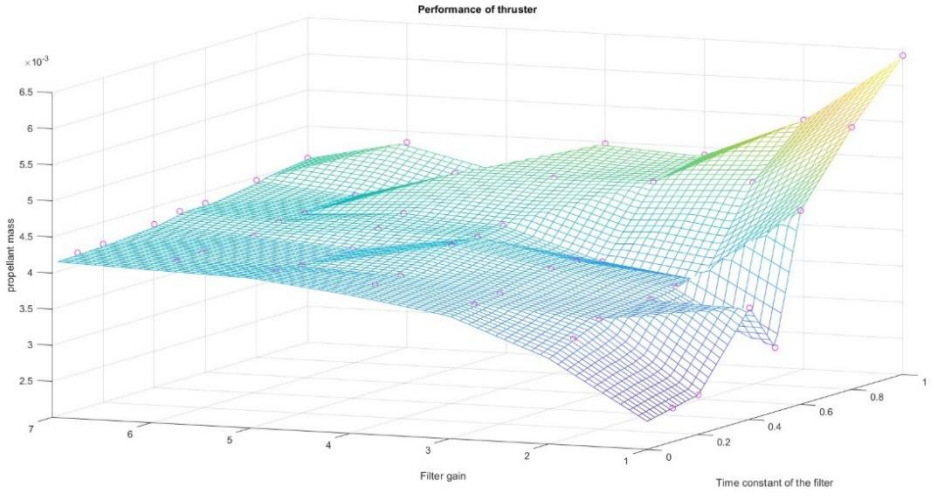

Figure 2.21: Propellant mass with respect to K_f and T_f.

In this case is possible to conclude that high value of time constant shall be avoided. In the end the parameters selected are reported in the following table:

Parameter	Value
k_f	6
τ_f	0.7
U_{on}	0.7
U_{off}	0.2

Table 2-6: Selected parameter.

CHAPTER 3

3 Guidance, navigation and control

3.1 Control function

The Control function computes the output to be sent to the actuators. In other word, the control function is in charge to drive the spacecraft trough a specific trajectory according to the guidance commands and to stabilize the attitude of the spacecraft with respect to a defined reference frame. The control function developed in the present dissertation has been implemented in order to support the design of the guidance function and to test the implemented Simulink model. In addition, the control function is continuously monitoring the reaction wheels in order to execute wheels de-saturation commands and restore the availability of the wheels. To control thrust and torque provided by the reaction control system, the control function implements a thrusters modulation function, based on the Pulse-Width Pulse-Frequency Modulation (PWPFM) technique.
The general approach used for the control of the spacecraft manipulator systemin this thesis is shown in figure 3.1:

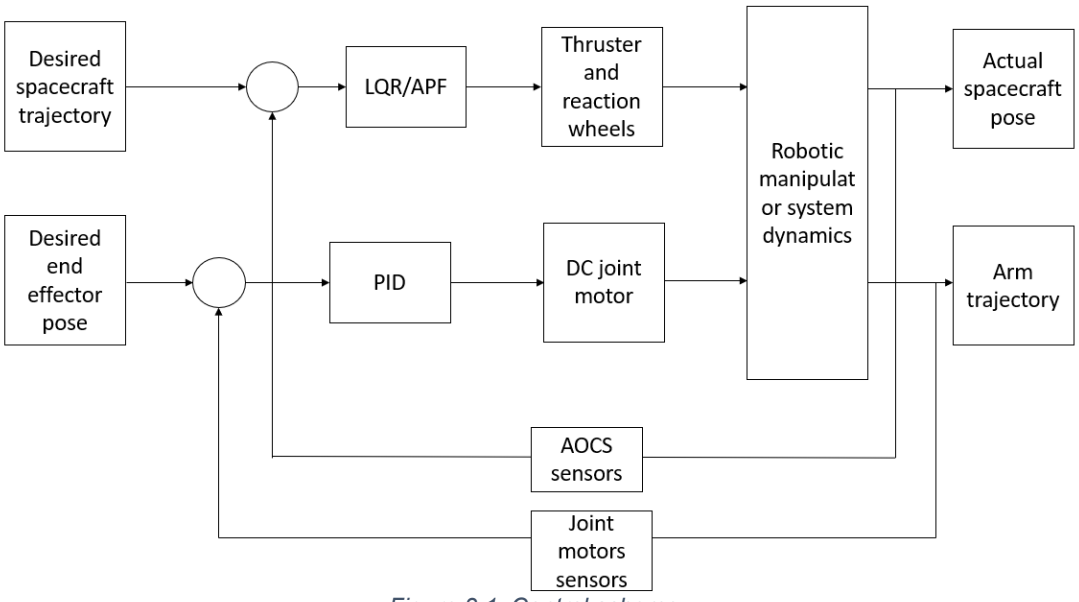

Figure 3.1: Control scheme.

Notably, the control of the spacecraft is executed using an LQR controller while a proportional-integrative-derivative controller is used for the control of the DC motor. The robot dynamics is implemented in Simcape multibody. During the robotic arm manoeuvring the desired trajectory and attitude of the chaser is settled to be the equilibrium position and orientation which maintain the spacecraft towards the target debris making easier the grasping operation.
The control scheme designed in this thesis work for the spacecraft control are the following one, represented in the figures as follow:

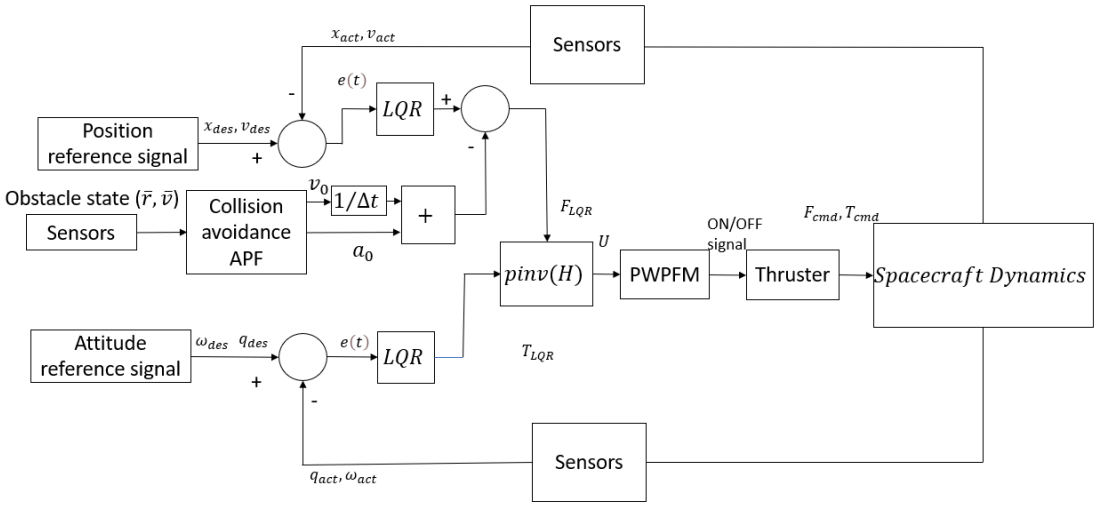

Figure 3.2: Designed control system of the spacecraft with thrusters.

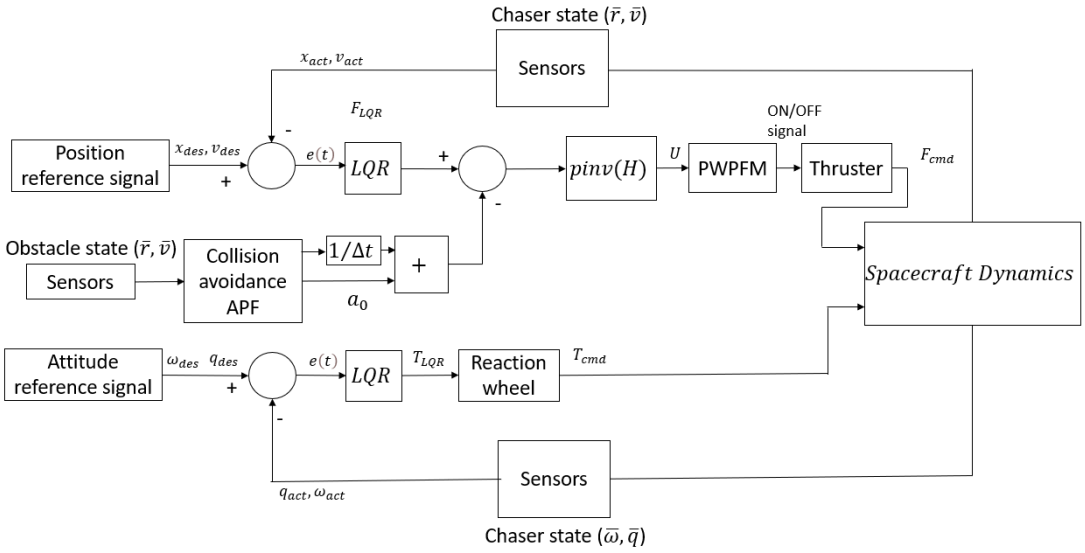

Figure 3.3: Designed control system of the spacecraft with thrusters and reaction wheels.

3.1.1 Basics of attitude control

As extensively mentioned in the previous section, attitude control is a fundamental function to be implemented in the GNC software for almost all space applications. Passive stabilization techniques will not be further discussed, since the present dissertation is based on active control systems. In the landscape of research two of the main control modes are PID and LQR [34].

- **Proportional integrative derivative controller**

The Proportional Integral Derivative (PID) Controller is commonly used in space application due to its simple design and its inherently robustness, especially if it is designed with proper techniques. In general, PID controller, typically use control loop feedback with the following general law:

$$u(t) = K_p e(t) + K_i \int e(t) + K_d \frac{de(t)}{dt} \quad (3.1)$$

The controller first computes the value of the error $e(t)$ as the difference between the reference signal and the feedback and then tries to minimize the error to match the reference with the actual state. Furthermore, we have K_p is the proportional gain, K_i is the integral gain and K_d is the derivative gain. In order to ensure stability and robustness, such gains have to be tuned properly. The proportional action, $K_p e(t)$, is the basic control action: the control is commanded proportionally to the error. The integral action, $K_i \int e(t)$ integrates the error and compute the control action in the long term. Applying only the proportional action, it can be present a residual static error which cannot be reduced only with the proportional control, and the integral action tends to reduce such error and allows to reach exactly the desired reference. Eventually, the derivative action, $K_d \frac{de(t)}{dt}$, acts on the error derivative of $e(t)$, hence it is used to increase dynamic performance of the controller and reducing residual oscillations derived by the proportional action alone. In many applications, including space applications, not all the control action are included in the design of the PID controller, which may result in a simpler PD or PI controller, depending by control requirements specified for each application. The error fed to the controller could be a position error or an orientation error, the last one can be expressed using the quaternion notation:

$$e(t) = q_{err} = q_{BI}^{-1} * q_{des} \quad (3.2)$$

where q_{err} is the quaternion error, q_{BI} is the Body quaternion with respect to the ECI frame, q_{des} is the desired reference quaternion and the symbol $*$ express the quaternion product.

- **Linear quadratic regulator LQR**

LQR is an optimal controller which consist of a full state feedback gain able to offer high efficiency controls that ensure both positioning accuracy and fuel optimization, it is the solution to a least-squares optimization problem. Moreover, they are widely used to control space systems [34]. Additionally, LQR method calculates the feedback gain, which stabilizes the motion and overcome uncertainties due to disturbance.
The LQR controller is based on state feedback and its gains are determined based on the trade-off between transient performance and control effort. Thus, the optimal control approach to this trade-off is to define and minimize a cost function.
The general formulation of a LQR controller starts from the state-space system representation:

$$\dot{x} = Ax + Bu \quad (3.3)$$

$$y = Cx + Du \quad (3.4)$$

Where y is the output of the system which is considered to be coincident with the state variables assuming the complete state observability, moreover D is chosen in such a way that $y = Cx$. Therefore, the chosen matrices are the following one:

$$C = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

$$D = 0_{6\times 3}$$

where x and u are respectively the state variables and the control actions, A is the state matrix and B is the control matrix. The goal is to find a control law in the form:

$$u = -Kx \tag{3.5}$$

which minimize the quadratic cost function J defined by:

$$J = \int_0^{\infty} (x^T Q x + u^T R u)\, dt \tag{3.6}$$

where K is the LQR gain, and Q and R are weighting matrices related respectively to the state and the control action. The selection of the gains of the LQR control law and the weighting matrices of the LQR is a highly laborious process. Thus, a reasonable choice of Q and R makes the closed-loop system acquire stable performance, while limiting the control action and minimizing fuel expenditure. However, in simulations, the choices of the elements of the Q and R matrices were made based on the diagonal weighting method. It verifies which values best meet the performance criteria such as the overshoot, control effort and settling time; thus, it results in better performance of the system. The optimal solution for K is in the form:

$$K = R^{-1} B^T P \tag{3.7}$$

where P is the solution of the Algebraic Riccati Equation (ARE):

$$A^T P + PA - PBR^{-1}B^T P + Q = 0 \tag{3.8}$$

Usually, the computation of the matrix P is not trivial, and it is computationally expensive. For this reason, the solution of the Riccati equation is computed off-line, as in the simple implementation of LQR.
Application of LQR to control the attitude dynamics of a spacecraft requires a linearization of the system dynamics, since LQR can be applied only to linear systems. In the classical formulation of a LQR controller, the control law drives the $\hat{x}$ state to zero. According to this definition, defining a control law related to the local vertical frame or inertial frame means to regulate the attitude of the spacecraft aligned with that frame.

In this thesis, two LQR are implemented, one for position and the other for attitude control, using MATLAB function block in Simulink.

3.1.2 Attitude control

As known from literature the attitude of a spacecraft can be represented using the rigid body dynamics, thus can be found that the time evolution of the spacecraft attitude dynamics is exploit by the Euler equation of motion reported here:

$$\boldsymbol{T} = \boldsymbol{J}\dot{\vec{\omega}} + \vec{\omega} \times \boldsymbol{J}\vec{\omega} \tag{3.9}$$

Or, equivalently in matrix form:

$$J\dot{\omega} = -\omega^{\times}(J\omega) + T \tag{3.10}$$

With $\omega \times$ or $\omega^{\times}$ that denote a matrix operator that maps the cross product $\omega \times x$ to a matrix product called skew-symmetric matrix, defined as follows:

$$\omega \times = \begin{bmatrix} 0 & -\omega_3 & \omega_2 \\ \omega_3 & 0 & -\omega_1 \\ -\omega_2 & \omega_1 & 0 \end{bmatrix}$$

where T is the vector of the torques acting on the spacecraft, J is the spacecraft inertia matrix and $\vec{\omega}$ is the spacecraft angular velocity. It is assumed that the constant inertia matrix J is a diagonal matrix because this is approximately correct in most real spacecraft designs.
If a spacecraft equipped with a robotic arm is considered, the base reaction torques generated by the manipulator on the spacecraft acts as an attitude disturbance. So, the equation can be rewritten in the following way [65]:

$$\boldsymbol{T}_{ACS} + \boldsymbol{T}_d = J\dot{\vec{\omega}} + \vec{\omega} \times J\vec{\omega} \tag{3.11}$$

where the controlled torques of the ACS (T_{ACS}) separated from the disturbance torques generated by the manipulator (T_d). According to the robotics inverse dynamics of manipulators, T_d can be expressed as:

$$\boldsymbol{T}_d = \boldsymbol{M}\ddot{\vec{q}_J} + \vec{n} \tag{3.12}$$

where $\boldsymbol{M}$ is the mass matrix, $\vec{n}$ is the centrifugal and Coriolis term and q_J is the vector of the manipulator joint variables (in this case j stands for joints). Moreover, in the $\boldsymbol{T}_d$ term, could be considered the gravity gradient torque and the atmospheric drag which are the predominant environmental disturbance torques in a LEO mission. From [54,83], can be found that the gravity gradient torque could be expressed, for small Euler angles by:

$$T_{gg} = \begin{bmatrix} 6\omega_0^2(J_{33} - J_{22})q_1 \\ 6\omega_0^2(J_{33} - J_{11})q_2 \\ 0 \end{bmatrix} \tag{3.13}$$

The atmospheric effect is difficult to find because of the dependence on solar activity, geomagnetic index, spacecraft geometry, spacecraft attitude, altitude, and others factor.

Is known from literature [91] that the most popular spacecraft models for attitude determination and control design methods are Euler angles and quaternions. The Euler angles are very efficient because the linearized Euler angles models are controllable, and all standard linear control system are directly applicable. However, the drawbacks are that the designs based on these models may not globally stabilize the nonlinear system, and the controller might have bad performance away from the equilibrium point where the linearization is performed, the models depend on rotational sequences and a singular point exist for any rotational sequence. While the quaternion models do not have singular points and can globally stabilize the nonlinear system. The drawback is that the linearized quaternion model is non-controllable, in [54, 55] the reduced quaternion model is introduced and is demonstrated to be fully controllable. Therefore, all standard linear system theory can be directly applied to the analysis and design of the spacecraft control systems and globally stabilize the nonlinear system.

Considering what has been stated, the control design method here is based on reduced quaternion spacecraft model. To construct a feedback control system, we need to introduce the definition of quaternion and then a linearized state space model based on quaternion is required to design the LQR.

Let the quaternion q represent the orientation of the rigid body with respect to the reference system at time *t,* defined as follows:

$$q_0 = \cos\left(\frac{\alpha}{2}\right)$$

$$q = [q_1 \quad q_2 \quad q_3]^T = \hat{e}^T \sin\left(\frac{\alpha}{2}\right)$$

Where q_0 is the scalar part and q the vectorial part, so the quaternion is defined as follows:

$$\overline{q} = [q_0 \quad q^T]^T = \left[\cos\left(\frac{\alpha}{2}\right) \quad \hat{e}^T \sin\left(\frac{\alpha}{2}\right)\right]^T$$

This one represents the rotation of the body frame relative to the inertial frame, where $\hat{e}^T$ is the unit rotational axis and α is the rotational angle about the rotational axis. The nonlinear spacecraft kinematics equations of motion can be represented by the quaternion, first let us consider the derivative of the quaternion:

$$\dot{q}(t) = \lim_{\Delta t \to 0} \frac{q(t + \Delta t) - q(t)}{\Delta t} \tag{3.14}$$

Now is possible to write the rotation from the quaternion at t to $t + \Delta t$ as the exponential of a rotation vector:

$$q(t+\Delta t) = \exp[(\Delta\vartheta/2)\otimes]\, q(t) \approx q(t) + [(\Delta\vartheta/2)\otimes] q(t) \tag{3.15}$$

Inserting this into the previous equation and taking the limit as Δt goes to zero gives:

$$\dot{q}(t) = \frac{1}{2}[\omega(t)\otimes] q(t) = \frac{1}{2}\omega(t)\otimes q(t) = \frac{1}{2}\Omega\big(\omega(t)\big) q(t) \tag{3.16}$$

Where:

$$\omega(t)\otimes = \Omega\big(\omega(t)\big) = \begin{bmatrix} -[\omega^{\times}] & \omega \\ -\omega^{T} & 0 \end{bmatrix} \tag{3.17}$$

It is often more convenient to use:

$$q\otimes\bar{q} = [q\otimes]\bar{q} = \bar{q}\odot q \tag{3.18}$$

$$[\bar{q}\odot] = \begin{bmatrix} q_4 I_3 + [q_{1:3}\times] & q_{1:3} \\ -{q_{1:3}}^{T} & q_4 \end{bmatrix} = [\Xi(q) q] \tag{3.19}$$

$$\Xi(q) = \begin{bmatrix} q_4 & -q_3 & q_2 \\ q_3 & q_4 & q_1 \\ -q_2 & q_1 & q_4 \\ -q_1 & -q_2 & -q_3 \end{bmatrix}$$

to write the kinematic equation for the quaternion in the form:

$$\dot{q} = \frac{1}{2} q \odot \omega = \frac{1}{2}\Xi(q)\omega \tag{3.20}$$

So, the nonlinear spacecraft kinematic equations are:

$$\begin{cases} \dot{q} = -\dfrac{1}{2}\omega \times q + \dfrac{1}{2} q_0 \omega \\ \dot{q_0} = -\dfrac{1}{2}\omega^{T} q \end{cases} \tag{3.21}$$

Yang, [54,55] showed that the nonlinear spacecraft kinematic equations of motion can be replaced by a set of independent nonlinear spacecraft kinematics equations of motion that leads to a controllable linearized quaternion model.

[54, 55] For any given quaternion q, if $\alpha \neq \pi$, there exists a one-to-one mapping between ω and $\dot{q}$. Moreover, let:

$$f(q) \coloneqq q_0 = \sqrt{1 - q_1^2 - q_2^2 - q_3^2} \tag{3.22}$$

And

$$\Phi = \begin{bmatrix} f(q) & -q_3 & q_2 \\ q_3 & f(q) & -q_1 \\ -q_2 & q_1 & f(q) \end{bmatrix} \tag{3.23}$$

Then, the one-to-one mapping is given by:

$$\omega = 2\Phi^{-1}\dot{q} \tag{3.24}$$

So, the non-linear system can be replaced by the following:

$$\begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \dot{q}_3 \end{bmatrix} = \frac{1}{2}\begin{bmatrix} f(q) & -q_3 & q_2 \\ q_3 & f(q) & -q_1 \\ -q_2 & q_1 & f(q) \end{bmatrix}\begin{bmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \end{bmatrix} = \frac{1}{2}\Phi\omega = g(q_1, q_2, q_3, \omega) \tag{3.25}$$

The linearized spacecraft system can be derived from previous equations by using the first order Taylor expansion around the stationary point as follows:

$$\dot{\omega} = J^{-1}u \tag{3.26}$$

$$\left.\frac{\partial g}{\partial \omega}\right|_{\substack{\omega=0 \\ q_1=q_2=q_3=0}} = \frac{1}{2}I_3 \tag{3.27}$$

$$\left.\frac{\partial g}{\partial q}\right|_{\substack{\omega=0 \\ q_1=q_2=q_3=0}} = 0_3 \tag{3.28}$$

Therefore, the state space form of the system is the following one, obtaining the linearized inertial pointing spacecraft model:

$$\begin{bmatrix} \dot{\omega} \\ \dot{q} \end{bmatrix} = \begin{bmatrix} 0_{3\times3} & 0_{3\times3} \\ \frac{1}{2}I_3 & 0_{3\times3} \end{bmatrix}\begin{bmatrix} \omega \\ q \end{bmatrix} + \begin{bmatrix} J^{-1} \\ 0_{3\times3} \end{bmatrix}u = Ax + Bu \tag{3.29}$$

Where:

$$A = \begin{bmatrix} 0_{3\times3} & 0_{3\times3} \\ \frac{1}{2}I_3 & 0_{3\times3} \end{bmatrix}, x = \begin{bmatrix} \omega \\ q \end{bmatrix}, B = \begin{bmatrix} J^{-1} \\ 0_{3\times3} \end{bmatrix} \tag{3.30}$$

And u is the torque control law. It is easy to verify that this linearized spacecraft system equation is controllable. From this we can obtain the gain for the full state feedback by the LQR design.

- **LQR design**

Since the linearized spacecraft system is fully controllable, LQR design can be directly obtained. The advantage of the LQR design is obvious because we can find optimal K to achieve other design goals, such as minimizing the control energy (required because of the restrictions on control authority and saturation) and optimizing the response performance. Instead of solving nonlinear Lyapunov matrix equation to get Q and R matrices, analytical feedback formulas can be found because of the simple and special structure of the linearized reduced quaternion spacecraft model [54, 55]. Moreover, the closed loop poles are implicitly designed with LQR as reported in [54, 55]. In addition, the LQR has a diagonal structure in the feedback gain matrix, this structure is simply a robust pole assignment design.
The procedure to design the LQR are two, is possible to do an on-line or off-line design, in the first case the Q and R matrices and the state space form are obtained during the spacecraft operations and constantly updated, the second case is characterized by the following scheme [55]:

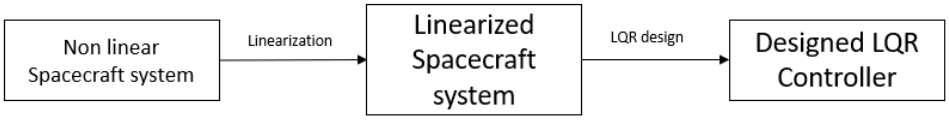

Figure 3.4: Design of the LQR.

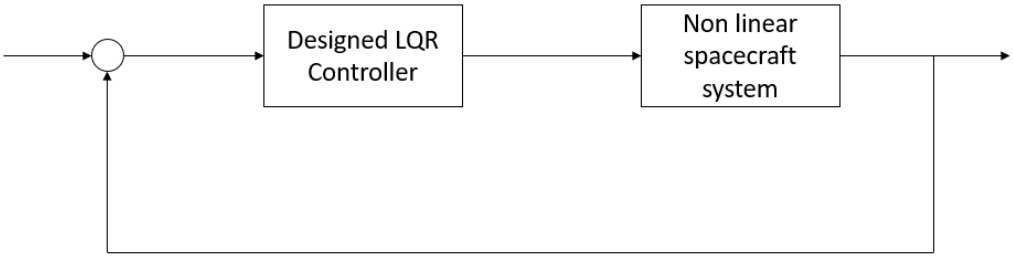

Figure 3.5: Application of LQR to nonlinear system.

Thre are two main effects of the designed LQR on the nonlinear system, first the controller globally stabilizes the nonlinear system as demonstrated in [53, 54,91], indeed for any initial value of attitude rate, the controller will bring them to $(\omega, q) = 0$. The second one is that when the nonlinear spacecraft is close to the origin, the linearized model is a valid approximation, therefore LQR can be defined as a suboptimal control for the original nonlinear system.
In the Simulink model the K matrix is multiplied for the state error expressed as:

$$\delta\vec{x}(t) = [\delta\vec{\omega}(t), \delta\vec{q}_{1:3}(t)]^T \tag{3.31}$$

where $\delta\vec{\omega}(t)$ is the angular velocity of the body frame relative to the inertial frame expressed in inertial system and $\delta\vec{q}_{1:3}(t)$ are the last three components of the quaternion that describes the orientation error. The orientation error can be described through the following quaternion product [34]:

$$q_{err} = q_{BI}^{-1} * q_{des} \tag{3.32}$$

Where q_{BI} express the rotation from the body frame into the inertial frame. As can be observed the information reported is just the vectorial part of the quaternion.
The next step will be to multiply the state error for the matrix K, which is merely a gain block:

$$\vec{u}(t) = -K\delta\vec{x}(t) \tag{3.33}$$

In the multibody system this signal feed the thruster allocation function or the reaction wheels in order to apply a torque to the spacecraft and to reduce the $\delta\vec{x}(t)$ to zero, allowing the chaser spacecraft to always point to the target in the final approach.
The tuning of the Q and R values are carried out with different simulations; indeed, they are usually selected based on the experience of the controller, in such a way that performance of the system in terms of fuel consumption, settling time, overshoot, static error and response time are met.

3.1.3 Simulation result.

The implementation of the designed LQR controller for the attitude control of the spacecraft can be easily implemented in Simulink through a MATLAB function where the lqr MATLAB function is recall given in input the state space form of the system.
Initially and for the results reported in chapter 5 the inertia matrix selected was the one directly computed form the Brick solid block of Simscape mechanical supposing a homogeneous distribution and the products of inertia to be zero, obtaining so a diagonal matrix. Chosen a chaser mass of 500 kg, the quaternion response is obtained in the following figure in two simulations with different initial conditions in case of attitude control performed by thrusters:

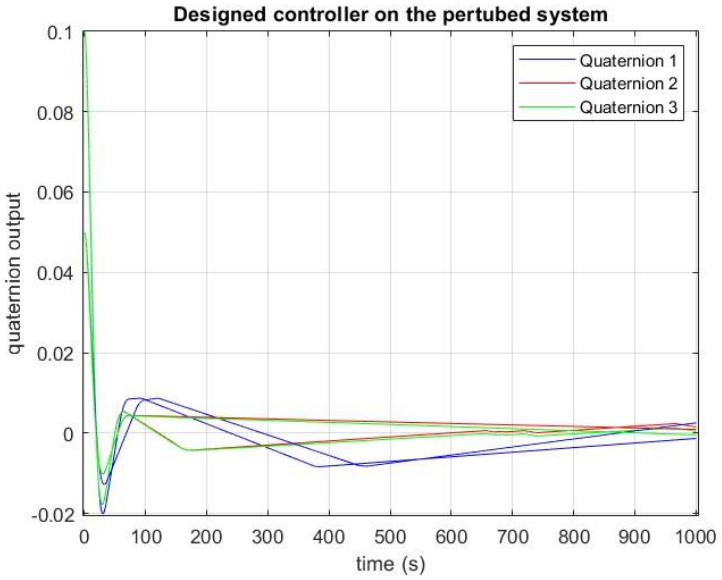

Figure 3.6: Quaternions performance in linear system.

As is possible to observe, because of the symmetric distribution of mass, the quaternions have initially the same trend.
While, as an additional step to study the influence of a different mass distribution, has been inserted in the model an inertia matrix taken from [82]:

$$J = \begin{bmatrix} 1200 & 100 & -200 \\ 100 & 2200 & 300 \\ -200 & 300 & 3100 \end{bmatrix} \tag{3.34}$$

In the same conditions of the two previous simulations, except for the inertia matrix, the following results was obtained with chaser mass of 500 kg and initial attitude condition different from zero:

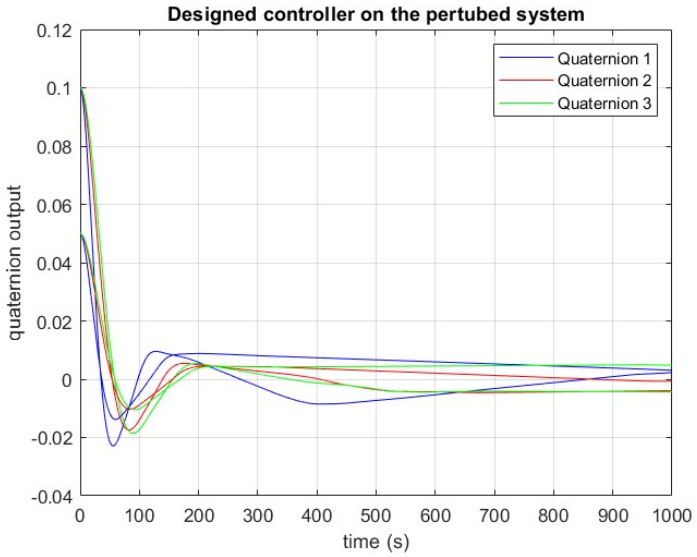

Figure 3.7: Quaternions for a different inertia matrix.

Notably, having imposed as equilibrium condition the unit quaternion $[1\ 0\ 0\ 0]$, which corresponds to coincidence in terms of orientation between inertial frame and body frame, the quaternions go to zero as expected similarly to the previous case. As can be noted, the last case differs from the previous one since the inertia matrix has an influence on the attitude, indeed the trend is completely different, and a small error can be observed in the final time step of the simulation. From graphic results can be ascertained that the simulation runs are asymptotically stable. This property persists even if a disturbance gravity gradient torque is introduced in the model as shown in the following result:

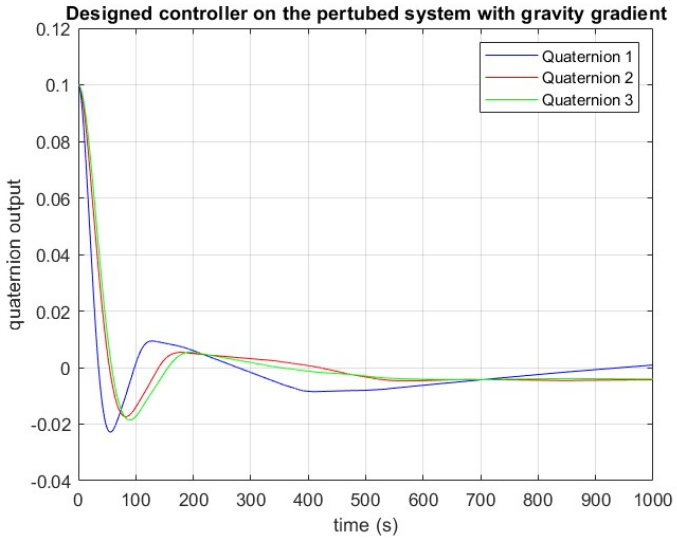

Figure 3.8: Quaternion with gravity gradient torque.

The order of magnitude of the disturbance torques caused by the gravitational field is evaluated to be around $10^{-6}\ Nm$ [65].
In this work thesis the inertia matrix of the spacecraft can be assumed to be diagonal, that is reasonable because in practical spacecraft design it is always designed to be close to a diagonal matrix, therefore in the result reported in chapter 5 only the mass is setted up while the inertia matrix is automatically computed from Simscape Multibody.

3.2 Position control

Before to design the LQR for the translational control, the equation that describes the translational dynamics of the chaser spacecraft need to be considered.
As is possible to read in the literature [30], the relative dynamics of the chaser with respect to the LVLH frame of the target can be described with different form. In [30] is available the following classification based on the eccentricity of the orbit and the J2 effect, which take into consideration the shape of the earth:

Theory	Eccentricity	J2 effect	Nonlinearity
Clohessy & Wiltshere	$e = 0$	No	No
Lawden	$0 \leq e < 1$	No	No
Gim-Alfired	$0 \leq e < 1$	Yes	No
Small Eccentricity	$e \ll 1$	Yes	No
Yan-Alfried	$0 \leq e < 1$	Yes	Yes

Table 3-1:Orbital equations.

For simplicity, the first model has been considered in this thesis; The Clohessy and Wiltshire theory express the following Hill's equations:

$$\ddot{x} = \frac{F_x}{m_c} + 2\omega_0 \dot{z} \tag{3.35}$$

$$\ddot{y} = \frac{F_y}{m_c} - {\omega_0}^2 y \tag{3.36}$$

$$\ddot{z} = \frac{F_z}{m_c} - 2\omega_0 \dot{x} + 3{\omega_0}^2 z \tag{3.37}$$

Where $r = [x, y, z]^T \in \mathbb{R}^3$is the position vector, m_c is the spacecraft mass, $\omega_0 = \sqrt{\mu/(r_{HF})^3}$ is the angular velocity of the LVLH frame at a distance r_{HF} from the centre of the Earth, μ is the gravitational parameter of the Earth and $F = \left[F_x, F_y, F_z\right]^T$ is the vector force which is usually defined as the sum of the forces due to the thrusters and the forces due to the action of the external environment disturbances affecting the chaser.

In this thesis, when formulating the control and guidance problem some assumption has been taken. For this purpose, the environment as well as the relative orbital dynamics are neglected as a consequence of the assumptions in chapter 1. This implies that an orbiting reference frame can act as an inertial frame.

Therefore, the LVLH of the target frame coincide with the inertial frame for a non-tumbling or non-drifting object, so the angular rate ω_0 can be neglected obtaining the following linear differential equation of motion:

$$\ddot{x} = \frac{F_{thrustx}}{m_c} \tag{3.38}$$

$$\ddot{y} = \frac{F_{thrusty}}{m_c} \tag{3.39}$$

$$\ddot{z} = \frac{F_{thrustz}}{m_c} \tag{3.40}$$

The forces in this case are given by the only contributes of the 12 thrusters, assumed to be expressed in the coordinate fixed to the spacecraft requiring the rotation matrix to obtain them in the inertial frame.

Remembering the general state space form:

$$\dot{x} = Ax + Bu \tag{3.41}$$

Where x is the state vector given by:

$$x = [x_{chaser}, y_{chaser}, z_{chaser}, \dot{x}_{chaser}, \dot{y}_{chaser}, \dot{z}_{chaser}]$$

$$A = \begin{bmatrix} 0_{3\times3} & I_{3\times3} \\ 0_{3\times3} & 0_{3\times3} \end{bmatrix}$$

$$B = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1/m_c & 0 & 0 \\ 0 & 1/m_c & 0 \\ 0 & 0 & 1/m_c \end{bmatrix} = \begin{bmatrix} 0_{3\times3} \\ \frac{1}{m_c} I_{3\times3} \end{bmatrix}$$

$$u = \left[F_{thrustx}, F_{thrusty}, F_{thrustz}\right]^T$$

- **LQR design**

As done in the previous section, the LQR design can be performed on-line or off-line, in the first case the value of the matrices Q and R are constantly updated.
As showed in [31, 32], the LQR can be designed following two different ideas. The first one is to define two optimal values that are updated based on the distance of the chaser from the target, or the second one to define a gain matrix which is computed with time-varying weighting matrices, these are updated at every loop based on the current distance, on the maximum velocity and on the maximum distance which is needed for safety reasons.
Before to introduce the control strategy adopted can be shown that in this work, we do not define a path or a trajectory, which are usually given from the guidance block, but the LQR control drives the spacecraft toward the target exerting an attractive effect on the chaser as a sort of APF, so that the tracking error is always the difference between the current state vector and the capture point where the approach manoeuvre end.
For this reason, the guidance function is accomplished by the control module for the translational control.

3.3 Guidance and motion planning framework.

To absolve this function as seen earlier in literature can be found different algorithms, in the past decades several solutions to autonomous RPO trajectory planning problem have been proposed. Among them we can find nonlinear optimization, model predictive control (MPC), mixed integer linear programming, feedback control methods (such as LQR) and convex programming. One benefit of nonlinear control methods is the implementation of nonlinear constraints, unfortunately those methods are computationally intensive and may not be a good candidate for autonomous guidance function. In the research the most advanced method is MPC or NMPC which is an optimal control that can be used for linear and nonlinear system. Every detail is reported in the following figure:

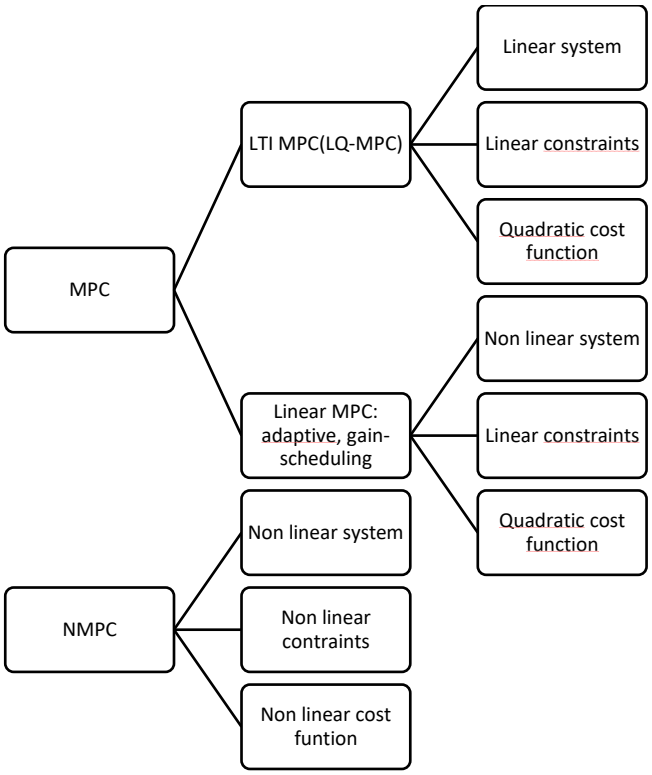

Figure 3.9: MPC and NMPC specification.

The first MPC leads to convex optimization problem, instead the NMPC brings to non-convex optimization problem [41, 42, 43, 44, 45]. Otherwise, the solution to the guidance problem can be found by solving a collection of convex programming problems, indeed it offers deterministic convergence properties, which makes it suitable for on-board implementations. Moreover, the guidance function can be solved using other types of algorithms which allow the system to compute a trajectory planning process, this trajectory generated is then fed to the system and executed by the spacecraft using an LQR which drive the spacecraft along the trajectory reducing the error between the actual state and the desired one. In general, to compute the trajectory of two main methods are available:

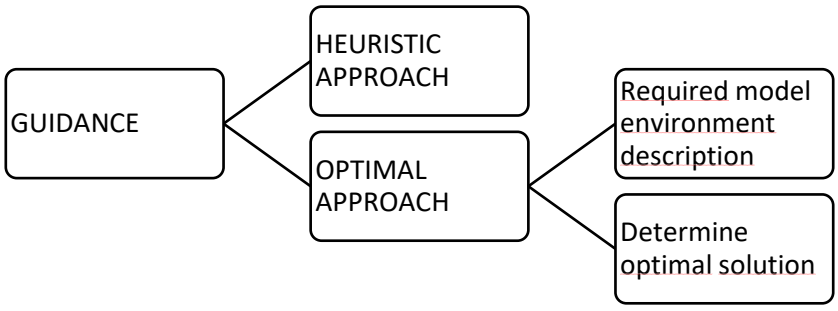

Figure 3.10 : Guidance approach.

So, the trajectory can be generated using an RPM probabilistic road map or a RRT and RRT* algorithm, rapidly exploring Random Tree algorithm.

3.3.1 LQR component for guidance.

In this case the LQR is used to drive the spacecraft along a trajectory, defining successive waypoints, or directly to the desired final position where the deployment and capture operations get started. The resulting LQR guidance allow the system to reduce the error, given by the difference between the actual and the desired state, to zero, both for the velocities and the position, therefore the LQR exert an attractive effect toward the goal on the chaser.

Assuming the state space formulation as given by:

$$\delta\dot{x} = A\delta x + Bu \tag{3.42}$$

Where:

$$\delta x = \begin{bmatrix} x_{des} \\ y_{des} \\ z_{des} \\ \dot{x}_{des} \\ \dot{y}_{des} \\ \dot{z}_{des} \end{bmatrix} - \begin{bmatrix} x_{act} \\ y_{act} \\ z_{act} \\ \dot{x}_{act} \\ \dot{y}_{act} \\ \dot{z}_{act} \end{bmatrix} \tag{3.43}$$

Where des means desired and act actual. The other component can be expressed as reported in the previous section. For further information see [31, 32, 33, 34].
The design of the LQR, as said earlier, in this case follow the same path as done before for the attitude control, indeed the goal is to find a control law in the form:

$$u = -K\delta x \tag{3.44}$$

which minimize the cost function J defined by:

$$J = \int_0^\infty (x^T Q x + u^T R u)\, dt \tag{3.45}$$

In this case to compute the Q and R matrices two different approaches are taken into consideration, and are respectively reported in the following matrices [31, 32]:

$$Q = K_{gain} * \begin{bmatrix} \alpha_Q/x_{max}^2 & 0 & 0 & 0 & 0 & 0 \\ 0 & \alpha_Q/y_{max}^2 & 0 & 0 & 0 & 0 \\ 0 & 0 & \alpha_Q/z_{max}^2 & 0 & 0 & 0 \\ 0 & 0 & 0 & \alpha_Q/\dot{x}_{max}^2 & 0 & 0 \\ 0 & 0 & 0 & 0 & \alpha_Q/\dot{y}_{max}^2 & 0 \\ 0 & 0 & 0 & 0 & 0 & \alpha_Q/\dot{z}_{max}^2 \end{bmatrix} \tag{3.46}$$

$$R = \begin{bmatrix} \beta_R/u_{xmax}^2 & 0 & 0 \\ 0 & \beta_R/u_{ymax}^2 & 0 \\ 0 & 0 & \beta_R/u_{zmax}^2 \end{bmatrix} \tag{3.47}$$

The relative position along each axis is equally weighted in the denominators by square of the distance of the spacecraft's centre of mass from the goal position:

$$x_{max} = y_{max} = z_{max} = r_g \tag{3.48}$$

The relative velocity error along each axis is weighted as follows:

$$\dot{x}_{max} = \dot{y}_{max} = \dot{z}_{max} = \frac{r_{init}}{r_m} v_m \tag{3.49}$$

The numerator terms are:

$$\alpha_Q = r_g = \beta_R \tag{3.50}$$

$$u_{x_{max}} = u_{y_{max}} = u_{z_{max}} = a_{z_{max}} = u_m \tag{3.51}$$

$$u_m = \frac{F_{thr}}{m_c} \tag{3.52}$$

Where:

- r_g: chaser's current distance from the COM of the target.
- $r_m = [p_{xmax} \;\; p_{ymax} \;\; p_{zmax}]$: maximum allowed distance between chaser and target COM.
- r_{init}:Initial distance between chaser and target COM.
- v_m: Maximum allowed relative velocity between spacecraft.
- u_m: Maximum acceleration along axes.

Where the K_{gain} is a value opportunely chosen that can take the following values:

$$K_{gain} = \begin{cases} K_{gain1} & if \;\; r_{init} - 1 < r_g < \; r_{init} + 1 \\ K_{gain2} & if \; the \; condition \; is \; not \; satisfied \end{cases} \tag{3.53}$$

Its tuning is based on the required performance of the system, the initial distance from the goal and from the fuel efficiency required. This method of choosing the Q and R matrices is very efficient for the final approach manoeuvre, where the mating point must be reached. After this phase through a flight manager the control mode can be changed on a different strategy of choosing Q and R.
An easier approach is to take only two values opportunely tuned trough simulations.

3.3.2 APF component.

The control algorithm previously showed, combine the LQR with a robust collision avoidance capability of artificial potential field (APF) control [31]. This portion of the

position controller uses position-based potential function to modify the chaser spacecraft's velocity as it moves close to the obstacle. The obstacle can be either moving or stationary, furthermore, it irradiates a repulsive potential field which allow to avoid the contact. The total effort of the combination between LQR and APF is the following one [31]:

$$\bar{a} = \bar{a}_{LQR} - \left(\left(k_v \frac{v_0}{\Delta t} \right) + k_s k_a a_0 \right) \tag{3.54}$$

Where, the velocity shaping parameter is defined as:

$$k_v = \left(e^{-(r_0^2/2\sigma^2)} - e^{-(D_0^2/2\sigma^2)} \right) \left(e^{-(L_0^2/2\sigma^2)} - e^{-(D_0^2/2\sigma^2)} \right)^{-1} \tag{3.55}$$

Based on the obstacle distance relative to the chaser, r_0. This factor is multiplied with the elative velocity ensuring the chaser to slow to zero at the boundary of the obstacle region defined as: $r_0 = L_0$, that is the minimum approach distance between the chaser COM and the obstacle. The region of influence is obtained as:

$$D_0 = d_0 \left(L_0 + D_{stop} \right) = (d_0 L_0) + \frac{d_0 |\bar{v}|^2}{4 u_m} \tag{3.56}$$

Where D_{stop} is the minimum distance in which the chaser spacecraft could stop when applying the maximum thrust. Therefore, the region of influence change with the relative velocity between chaser and obstacle.
The other terms are:

- σ: Standard deviation term obtained as $\sigma = (D_0 + L_0)/3$;
- k_a:Acceleratiom shaping parameter obtained as $k_a = e^{-d_a(r_0 - L_0)}$;
- k_s:The safety function $k_s = 1 - e^{-(2 d_a r_g)}$;
- d_a: Is a positive constant that determines the parameter's rate of decay.

The parameter k_a and $k_s$are multiplied for the following component of the LQR commanded acceleration directed towards the obstacle:

$$a_0 = \frac{\bar{r}_0 \cdot \bar{a}_{LQR}}{r_0} \frac{\bar{r}_0}{r_0} \tag{3.57}$$

To slow or stop the motion in that direction. Furthermore, as described in [31], with the aim of avoiding undesired condition where the attractive force exerted by the LQR and the repulsive cancel each other obtaining an undesired minimum. Therefore, using a technique similar to wall falling, an in-plane control component is introduced. It is orthogonal and proportional to the APF command and is added to the overall commanded force. The component is only generated in the xy plane of the inertial frame since, it is sufficient to avoid local minima and usually during the final approach the spacecrafts are in the same orbital plane. Therefore, there are two possibilities [31]:

$$\bar{a}_{\perp} = c \begin{bmatrix} -a_{APF_y} \\ a_{APF_x} \\ 0 \end{bmatrix} \tag{3.58}$$

$$\bar{a}_{\perp} = c \begin{bmatrix} a_{APF_y} \\ -a_{APF_x} \\ 0 \end{bmatrix} \tag{3.59}$$

The scalar c is a positive proportional value chosen by the control designer. The choice between the two components is made based on the shortest way in the case of docking manoeuvre [31]:

$$acos\left(\bar{a}_{\perp} \frac{\bar{r}_{dock}}{a_{\perp} r_{dock}}\right) \tag{3.60}$$

The chosen parameter ae reported in the following table:

Parameter	Value
$\boldsymbol{L_{max}}$	2.82 m
$\boldsymbol{L_0}$	$3 * L_{max}$
$\boldsymbol{d_a}$	15
$\boldsymbol{d_0}$	3
$\boldsymbol{u_m}$	$0.002\ m/s^2$
c	0.1
$\boldsymbol{\Delta t}$	5s
$\boldsymbol{v_m}$	$0.02\ m/s$

Table 3-2: Chosen parameter.

3.3.3 Approach maneuver simulation: preliminary analysis.

After having defined the control logic and the guidance algorithm to perform the approach maneuver with obstacle avoidance, some simulations are carried out to observe the spacecraft trajectory in the inertial space, stetting the following parameters:

Parameter	Value
Initial position [m]	[-40,0,0]
Obstacle position [m]	[-20,0,0]
Target position [m]	[0,0,0]

Table 3-3: Simulation environment coordinates.

Through the simulation is possible to obtain the result reported in figure 3.11, where in red is highlighted the region of the minimum distance from the obstacle and in blue the trajectory of the chaser spacecraft. As can be noted in the first part of the approach the chaser follows a straight line until it starts to slow down in the region of influence of the obstacle. As the chaser reaches the region in red, it starts to deviate from the linear path, the trajectory is the result of 3 acceleration component given respectively by the attractive contribute exerted by the LQR, the repulsive effect of the APF and the acceleration perpendicular to the two components.
After that the obstacle has been avoided, the chaser reaches the goal position where the deployment of the robotic arm starts.

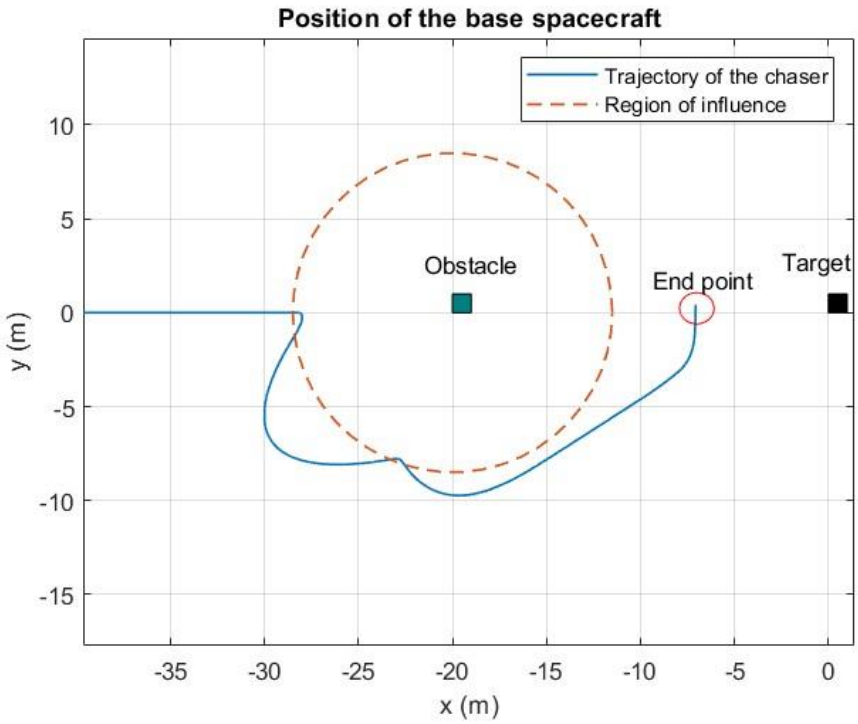

Figure 3.11: Chaser trajectory in inertial space $K_{gain} = 1$.

Others obtained results are reported in the following table as mass consumption, thruster firings and the required time to complete the manoeuvre. Moreover, $K_{gain} = 1$ is the setted value.

Parameter	**Value**
Mass consumption [Kg]	0.1466
Number of thruster firings	7486
Required time [s]	1250

Table 3-4: Simulation results.

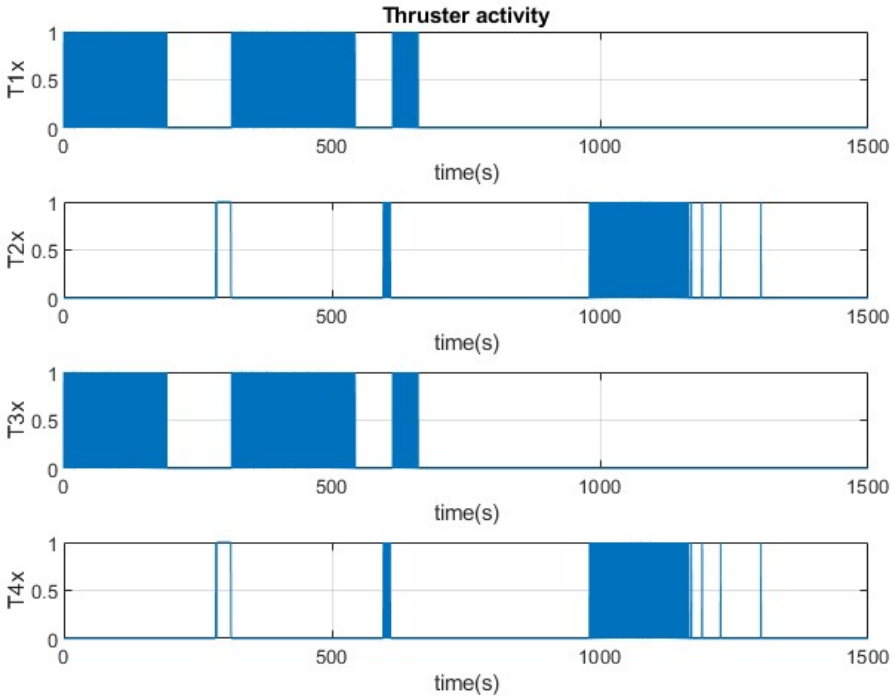

Figure 3.12: Group x thrusters.

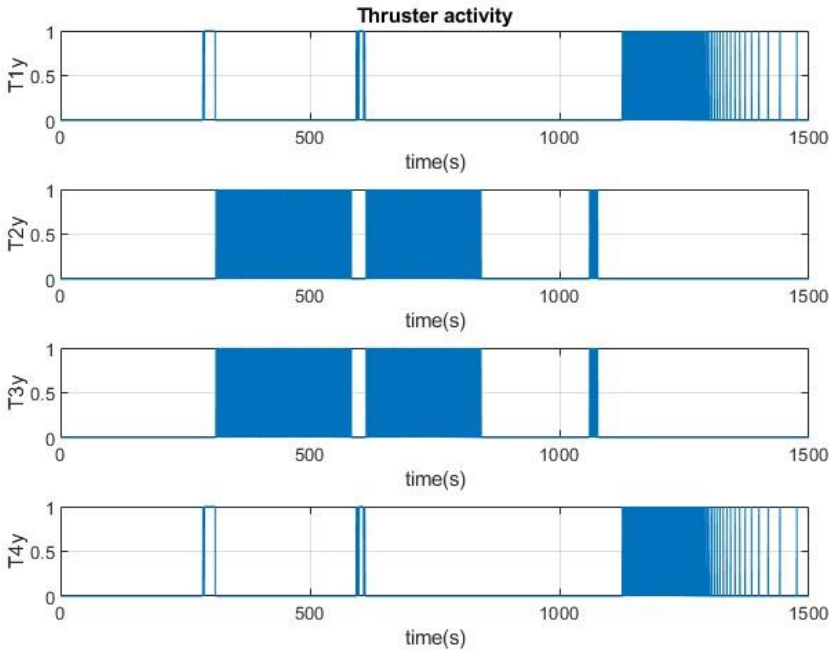

Figure 3.13: Group y thrusters.

Increasing the value $K_{gain} = 10$ and therefore the component along the diagonal of the weighting matrix Q, we can observe the following results.

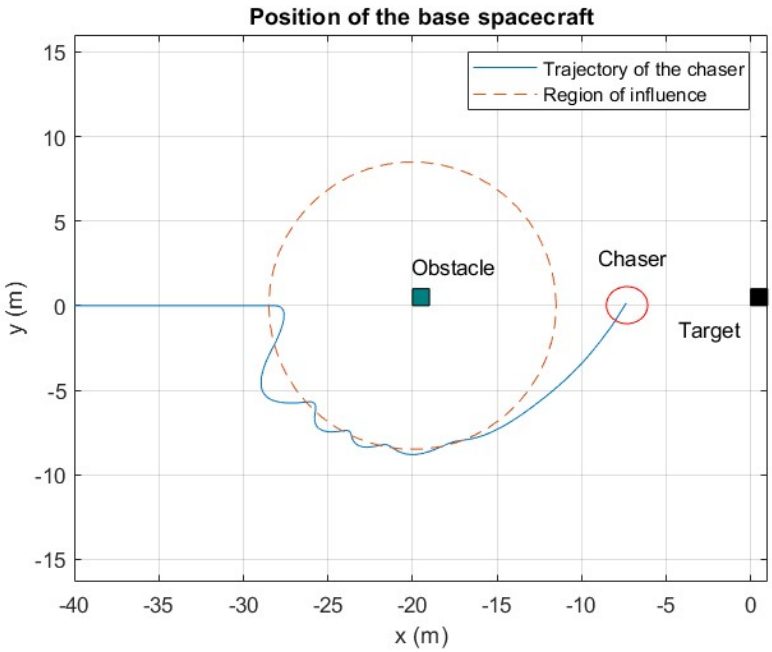

Figure 3.14: Chaser trajectory in inertial space.

Parameter	**Value**
Mass consumption [Kg]	0.3097
Number of thruster firings	10710
Required time [s]	930

Table 3-5: Parameter results.

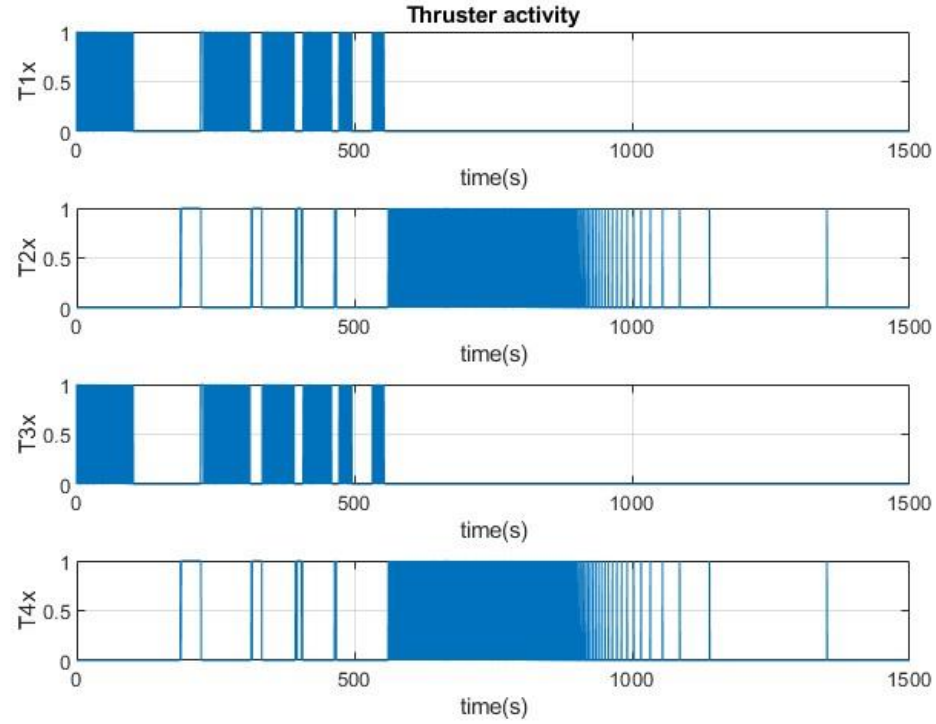

Figure 3.15: Group x thruster.

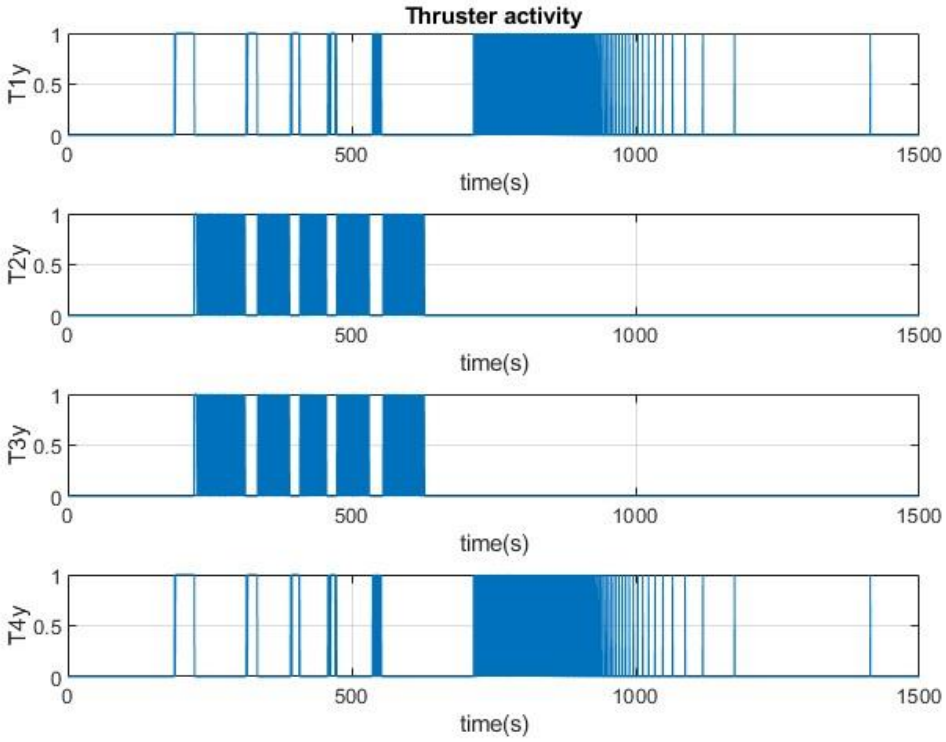

Figure 3.16: Group y thruster.

Notably, the approach maneuver is executed in a shorter time but with a higher mass consumption and thruster firings. Furthermore, the chaser trajectory passes through

the region boundary a higher number of times because of the increased effect of attractive portion.
This last section was dedicated to the LQR/APF guidance function to execute the approach maneuver, however, the simulation results reported in chapter 5 assumes that the above-mentioned maneuver has already been performed, as we focus on the study of the dynamic behaviour of the spacecraft in response to the robotic arm movement in different flight mode control during the capturing phase of the target.

CHAPTER 4

4 Spacecraft manipulator system (SMS).

In this chapter a detailed description of the spacecraft manipulator system is provided with an explanation of the approach followed in each case of study and of control algorithm.

4.1 Control algorithms for space manipulators.

The planning and control of the robotic manipulator is deeply discussed in literature [1,23,46,47,59,85,86,87,88,89]. Generally we can find three main types of control schemes that can be used. The first one consist of using, as shown in previous section, reaction jets to control spacecraft for attitude and position, in this case control algorithms used for fixed based manipulators can be adopted since the stabilization system compensate any manipulator dynamic forces exerted on the spacecraft, specifically in this case the manipulator can be considered as a disturbance factor for the base spacecraft, however the use of this kind of control algorithms may be limited because manipulator motion can saturate the reaction system. Moreover, the use of thrusters to stabilize the spacecraft leads to fuel efficiency optimization problem to extend the operative life of the spacecraft and to less accuracy in the attitude control, however momentum exchange devices can be used for fine pointing attitude. In the second category is possible to control only the attitude and not the translation, this kind of problem can be simplified using the virtual manipulator method [90]. Lastly, the motion control can be absolved using the free-floating mode, in this case the GJM is used to control the manipulator, as explained in chapter 1 this flight mode is characterized by absence of active stabilization method, which are turn-off during the motion, therefore, dynamic singularities can occur during the grasping operation.

4.1.1 Differences between free-floating and fixed based

Here we are going to describe the main differences between the terrestrial fixed based manipulator and the free floating:

- Terrestrial fixed based manipulator Jacobian depend on the joint angles only. In the free-floating case the Jacobian depends also on spacecraft orientation, which could be calculated or measured on-line by sensors.
- In general, the kinematic parameters are enough for fixed base manipulator control purposes. The Jacobian of the free-floating case also depends on dynamic properties of the system. External sensing and on or off-line parameter identification, can be very important for these systems.
- In case of fixed based system, we talk about kinematics singularities, in space, dynamic singularities exist, and they depend on inertia and mass properties.
- Is not possible to map desired Cartesian workspace points to a unique set of desired joint angles q for free-floating systems, because infinite sets of joint angles correspond, in general, to any workspace point.

4.2 Spacecraft manipulator system.

In this thesis the robotic arm POPUP mounted on the spacecraft is built as a 7-DOF anthropomorphic manipulator powered by electric motors, that consist of 2 inflatable flexible links, a spherical wrist, and a gripper, so in total the space manipulator is composed by $n + 7$ DOF where n is 6 and represent the degree of freedom of the base rigid body in the space. A dynamic physical model has been developed in Simscape Multibody environment where have been considered the dynamics of the motor, the attitude, and the position control of the spacecraft, the robotic arm controller, and zero-gravity environment has been set up.
The target has been then modelled as a cube to preserve position and orientation information, moreover it is equipped with a handle point to allow the grasping operation. The robotic arm has been implemented following [63], where is addressed the comparative study between the rigid body model RBM and the flexible body model FBM. In this thesis both the RBM and FBM will be analysed and compare observing the simulation results for the attitude and position error and for the end effector trajectory. In the below figures 4.1 is reported a graphic description of the system [102].

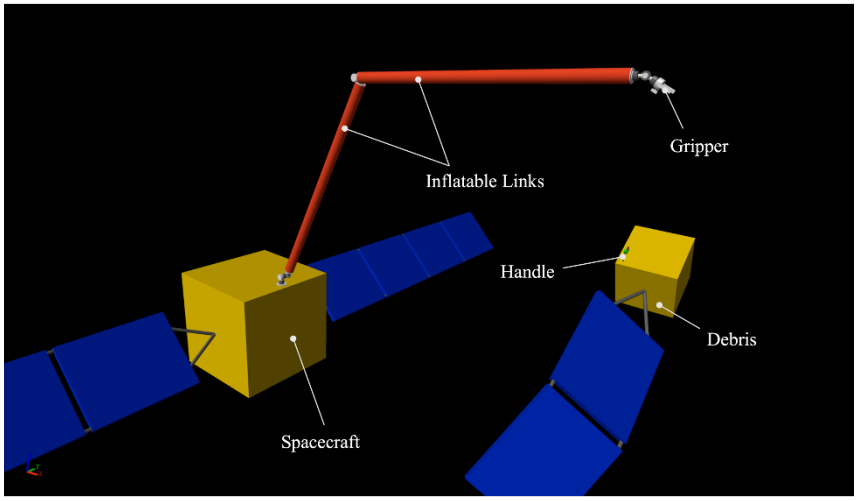

Figure 4.1: SMSs description.

Furthermore, the chaser and target spacecrafts are left free to move in the inertial space and this condition is obtained in the Simulink model thanks to the presence of a 6 DOF joint in the multibody library.
In the free-floating approach the manipulator will be considered as if it was composed by 7 links, considering the motor as a link, two of them are flexible and are longer than the other in the model, the links are numbered going from the base to the end effector. The base spacecraft is defined with link 0, with joint 0 is identified in the base-spacecraft centre of mass, instead, the end effector is considered as part of the link N ad its location can be treated as a joint N+1. Furthermore, the spacecraft is equipped with two cameras, one positioned on the end effector and the other one on the base of the spacecraft, they allow to detect the target position and orientation, the first is expressed in terms of cartesian coordinates and the second in terms of quaternion. Specifically, in the Simulink model the reference frame detected is positioned on the grasping point that consist of a handle on the target spacecraft. So,

in this case for the control of the robotic arm and inertially referenced end-point motion control will be used, where the manipulator end-effector is commanded to move with respect to an inertial space.
The first step to implements the control algorithms for the SMS is to obtain the kinematic equations which allow us, together with the dynamic equations, to write algorithms for the motion control of the robotic arm.

4.2.1 Kinematics of the SMS for free-floating algorithm.

For the kinematic study of the robotic manipulator the homogeneous transformation matrix (HTM) is introduced here. It is defined as a 4x4 matrix that belongs to the so called Special Euclidean group:

$$^{A}T_{B} \in SE(3)$$

Where:

$$SE(3) = R^3 \times SO(3)$$

These can be used to express the pose of a rigid body in the inertial space, let us consider the pose of a CCS (cartesian coordinate system) B with respect to another pose express by the CCS A, it is univocally defined by the 4x4 HTM from B to A and it is indicated by $^{A}T_{B}$ defined as:

$$^{A}T_{B} = \begin{bmatrix} ^{A}R_{B} & ^{A}r_{B} \\ 0_{1\times3} & 1 \end{bmatrix} \quad (4.1)$$

Where $^{A}R_{B}$ denote the rotation matrix $^{A}R_{B} \in SO(3)$ from CCS A to CCS B, which are direct cosine matrix., while $^{A}r_{B}$ express the position of the origin of the CCS B with respect to CCS A.

Notably, the homogeneous transformation matrix is:

- Not orthonormal.
- Has determinant always equal to 1.

Generally, HTM can be defined between the elements of any pair of CCSs. Therefore, the geometry of a base-manipulator system with a single N-link manipulator is illustrated in the figure 4.1 [23].

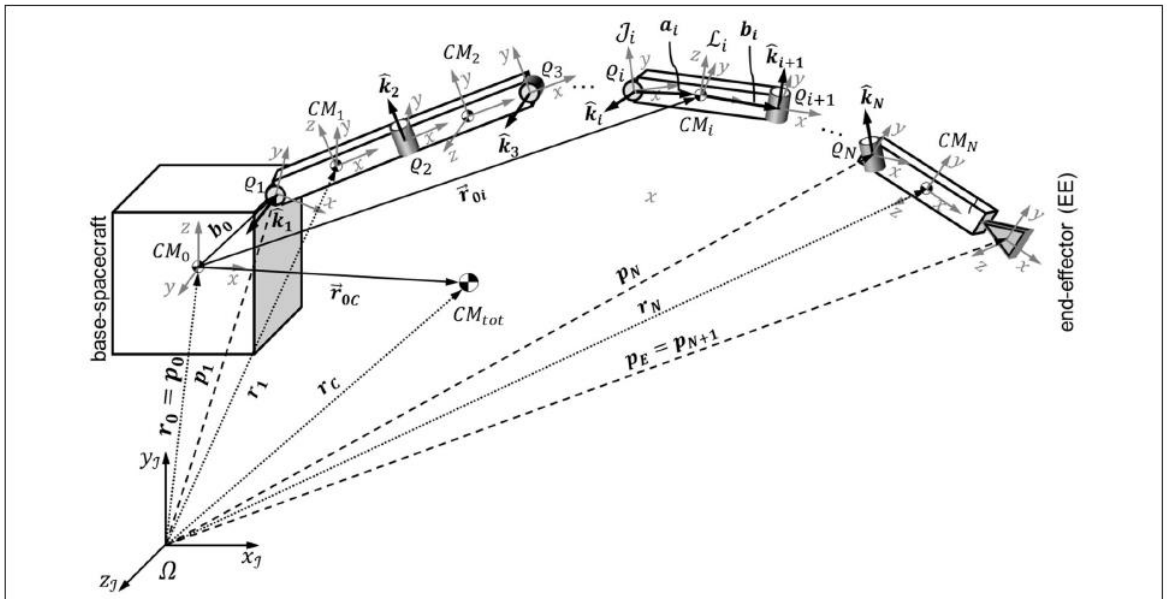

Figure 4.2: Geometry of a spacecraft-manipulator system.

The configuration of the space manipulator can be described unequivocally through the vector position of the joints reported in the above figure p_i $(0 \leq i \geq N+1)$, and by the position vector of the centre of mass r_j $(0 \leq j \geq N)$. These two vectors are expressed in the inertial coordinate system J.

As done in [23], three cartesian coordinate system are being used here: an inertia coordinate system J, a set of $N+2$ joint coordinate system J_i $(0 \leq i \geq N+1)$, and a set of $N+1$ ink-fixed coordinate system L_i $(0 \leq i \geq N)$, with $J_0 \equiv L_0$ since the base has no joints and $J_{N+1} \equiv J_{EE}$. The spacecraft coordinate system has its origin in the centre of mass of the base and has an arbitrary orientation which can choose has the principal directions of inertia. Instead, each joint reference frame is chosen through the of Denavit-Hartenberg convention as specified in [23, 21].

Based on this convention and referring to the figure 4.2 [23], we can choose the reference frame with the following procedure:

- Choose z_{J_i} of the joint reference frame set J_i parallel to the joint rotation axis $\hat{k}_i$;
- Locate the origin O_i at the intersection of the axis z_{J_i} with the common normal to axes z_{J_i} and $z_{J_{i-1}}$.
- Choose axis x_{J_i}along the common normal to axes z_{J_i} and $z_{J_{i-1}}$ with direction from joint J_{i-1} to J_i, if the axes are orthogonal to each other the x_{J_i} direction is arbitrary.
- In the end we can choose y_{J_i} to complete a right-handed frame.

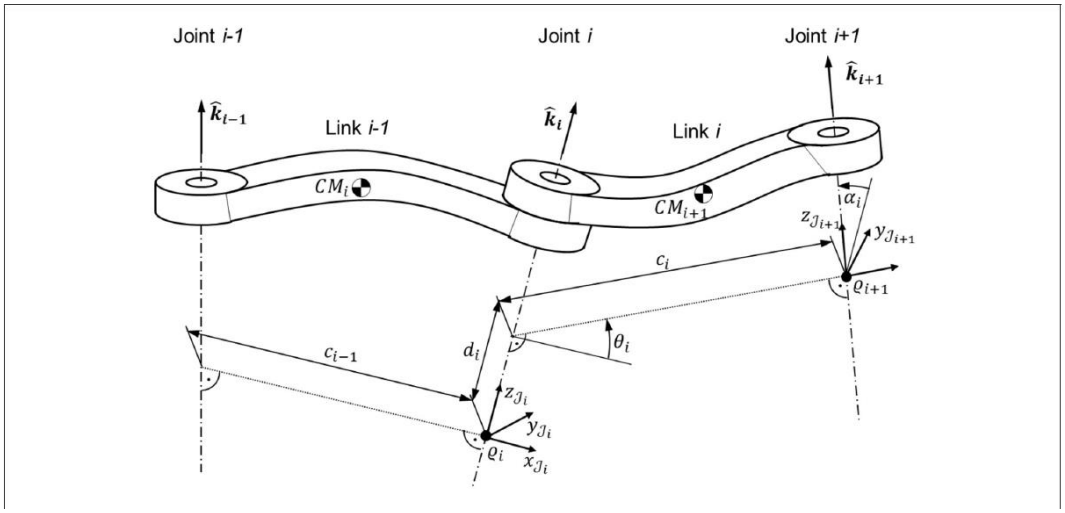

Figure 4.3: Denavit-Harteberg convention.

After having choose the reference frame for the joints, we can choose the links reference frames L_i, in this case we can follow the convention as done for the joints, the origin of the links reference frame are selected to be in the links COM. Notably, there is a difference between the procedure exposed with the one explained in [21], indeed, Siciliano's link I coordinate system is identical to our joint i+1 coordinate system J_{i+1}. The origins of the link coordinate systems L_i are placed in order to comply with [46]. After having defined the reference frames, the relationship between them can be found using the homogeneous transformation matrix expressed as function of DH parameters which are exposed in the figure 4.2. As already said, the joints in the space manipulator applications are usually revolute joints, so the only variable parameter is θ_i which corresponds to the variable joint q_i, all the other parameters are usually constant since they depend on geometrical properties of the serial chain. So, the HTM that express the relationships between J_i and J_{i+1} is given by:

$$^{Ji}T_{J_{i+1}} = A(q_i, d_i, \alpha_i, c_i) \tag{4.2}$$

Where the D transformation matrix function is defined as:

$$A(q_i, d_i, \alpha_i, c_i) = \begin{bmatrix} \cos q_i & -\sin q_i \cos \alpha_i & \sin q_i \sin \alpha_i & c_i \cos q_i \\ \sin q_i & \cos q_i \cos \alpha_i & -\cos q_i \sin \alpha_i & c_i \sin q_i \\ 0 & \sin \alpha_i & \cos \alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{4.3}$$

Therefore, the HTM so obtained can be expressed in the inertial frame and for each joint coordinate system can be expressed recursively with a product of HTM. So, we can obtain the following product:

$$^{J}T_{J_{i+1}} = {}^{J}T_{J_i}\,{}^{J_i}T_{J_{i+1}} \;\; \forall\, 2 \leq i \leq N \tag{4.4}$$

Where:

$$ {}^{J}T_{J_i} = \begin{bmatrix} {}^{J}R_{J_i} & {}^{J}p_i \\ 0_{1\times3} & 1 \end{bmatrix} \tag{4.5} $$

Analogously, the HTM of the links can be expressed as reported the with DH parameters:

$$ {}^{J_i}T_{L_i} = A(q_i, d_i, \alpha_i, a_i) \tag{4.6} $$

Where a_i is a position vector that connects the origin ϱ_i of the joint i to the centre of mass of link i CM_i, moreover, if CM_i lies along a straight line connecting ϱ_i to ϱ_{i+1}, a_i and b_i can be substituted by their scalar components along x direction, in this thesis we are going to use f_i instead of a_i and c_i instead of d_i.

As done earlier or the joints, we can obtain the following relationship:

$$ {}^{J}T_{L_i} = {}^{J}T_{J_i}\,{}^{J_i}T_{L_i} \;\; \forall\, 1 \leq i \leq N \tag{4.7} $$

Since the spacecraft is designated a link 0 and the joint 0 is in its centre of mass, we can obtain ${}^{J}T_{J_0} = {}^{J}T_{L_o}$.
The position of the system centre of mass with respect to the base-spacecraft is given by:

$$ {}^{\mathcal{J}}r_{0C} = {}^{\mathcal{J}}r_C - {}^{\mathcal{J}}r_0 \tag{4.8} $$

Analogously, the position of the centre of mass of every link with respect to the inertial frame is given by:

$$ {}^{\mathcal{J}}p_i = {}^{\mathcal{J}}r_C + {}^{\mathcal{J}}r_{Ci} \tag{4.9} $$

Where the position of the centre of mass of the link I with respect to the centre of the system is:

$$ {}^{\mathcal{J}}r_{Ci} = {}^{\mathcal{J}}r_{0i} - {}^{\mathcal{J}}r_{0C} \tag{4.10} $$

From the figure 4.. we can see that the position of a specific point X_i of a rigid body in the serial chain can be expressed in different ways, in the inertial frame we have:

$$ r_{X_i} = {}^{\mathcal{J}}r_C + {}^{\mathcal{J}}r_{Ci} + {}^{\mathcal{J}}r_{i,X_i} \tag{4.11} $$

Where ${}^{\mathcal{J}}r_{i,X_i}$ is the position of a certain point in body I with respect to the centre of mass of the same link. Therefore, the end effector position in the inertial frame is given by:

$$ r_{EE} = {}^{\mathcal{J}}r_C + {}^{\mathcal{J}}r_{CN} + {}^{\mathcal{J}}r_{N,EE} \tag{4.12} $$

Taking the derivative of the position, we can thus obtain in the general case:

$$ {}^{J}v_{X_i} = {}^{J}v_0 + {}^{J}\omega_0^x\left({}^{J}x_i - {}^{J}r_0\right) + \sum_{k=1}^{i}\left\{\left[{}^{J}\hat{k}_k^x\left({}^{J}x_i - {}^{J}p_k\right)\right]\dot{q}_k\right\} \tag{4.13} $$

$$ {}^{J}\omega_{L_i} = {}^{J}\omega_0 + \sum_{k=1}^{i}\left({}^{J}\hat{k}_k\dot{q}_k\right) \tag{4.14} $$

4.2.2 Kinematics of the SMS for free-flying case.

To describe the kinematic of the free-flying and translation flying case, as said before, is going to be used the fixed base algorithms. As done in the previous section the DH convention allow us to determine the joints reference frames of the robotic manipulator, however, in this case the kinematic is described with respect to the base frame at which the manipulator is mounted on the spacecraft. Following the DH convention as reported in [23], the homogeneous transformation matrix can be expressed in the following way:

$$ {}^{J_i}T_{J_{i+1}} = A(q_i, d_i, \alpha_i, a_i) \tag{4.15} $$

$$ {}^{0}T_{J_{i+1}} = {}^{0}T_{J_i}\,{}^{J_i}T_{J_{i+1}} \;\; \forall\, 2 \leq i \leq N \tag{4.16} $$

Where:

$$ {}^{0}T_{J_i} = \begin{bmatrix} {}^{0}R_{J_i} & {}^{0}p_i \\ 0_{1\times 3} & 1 \end{bmatrix} \tag{4.17} $$

With zero is indicated the base frame and J_i the joint frame chosen with DH convention. The following image express the DH convention [21]:

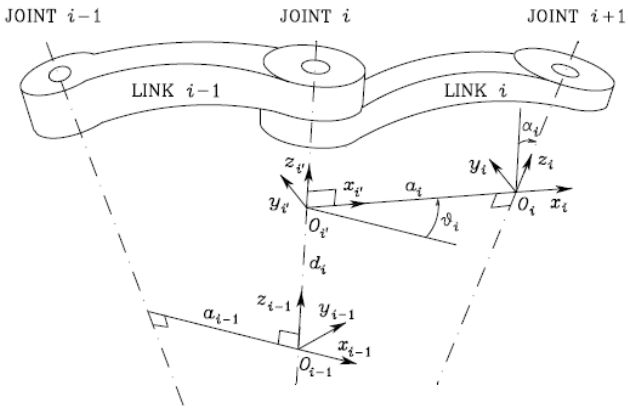

Figure 4.4 : Denavit-harteberg convention.

4.2.3 POPUP DH tables.

As shown in the previous section, following the DH convention, the HTM for the robotic arm can be obtained, in this section we are going to define two different choices of reference frames which are going to be used respectively for free flying/translation flying modes and for free floating/rotation floating modes. Therefore, first an analyse of robotic structure of the 7 DOF POPUP is needed, then is necessary to build a stick robot configuration to choose the reference frames. So, in the following figures are showed the reference frame choice and e robotic arm structure:

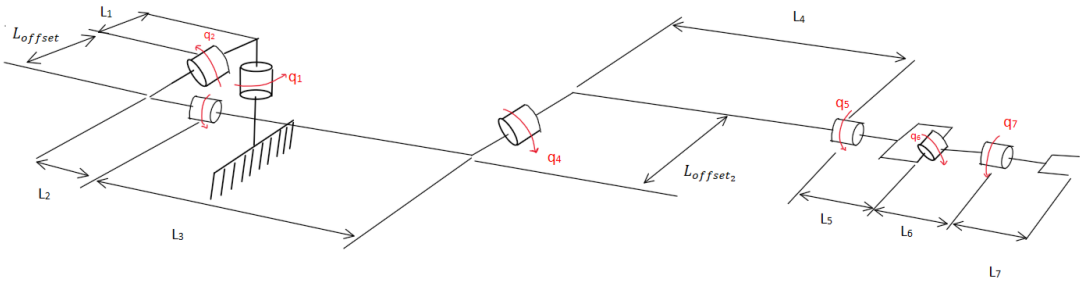

Figure 4.5: Robotic arm.

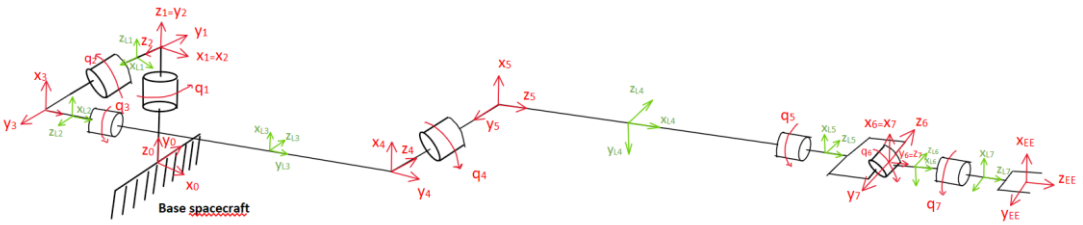

Figure 4.6 : Denavit-Hartenberg convention for free-floating.

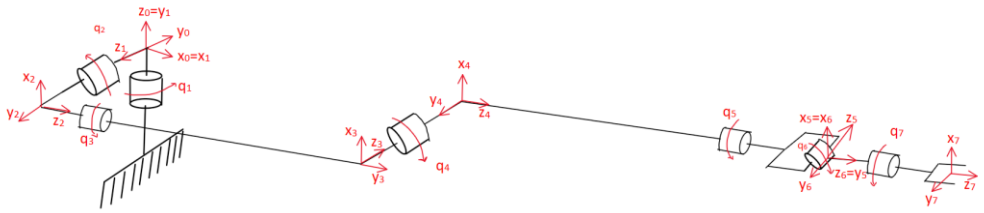

Figure 4.7: Denavit-Hartenberg convention for fixed base.

Considering the above figures, we can find the Denavit-Harteberg parameters referred to the reference frame choice, referring to the figure for the free-floating case one can get:

Joint Reference frame	a	d	α	θ_i
2	0	0	$\pi/2$	q_1
3	0	$L_1 + L_{offset}$	$\pi/2$	$q_2 + 90$
4	0	L_3	$\pi/2$	q_3
5	0	L_4	$-\pi/2$	q_4
6	0	L_5	$\pi/2$	q_5
7	0	0	$-\pi/2$	q_6
EE	0	L_6	0	q_7

Table 4-1: DH parameters for joint reference frame.

Links Reference frame	f	c	α	θ_i
$\mathcal{L}_1$	$L_1/2$	0	0	$q_1 - 90$
$\mathcal{L}_2$	$L_2/2$	$L_1 + L_{offset}$	0	q_2
$\mathcal{L}_3$	0	$L_3/2$	$\pi/2$	q_3
$\mathcal{L}_4$	$L_5/2$	L_4	0	q_4+90
$\mathcal{L}_5$	0	$L_5 + L_{6/2}$	$\pi/2$	q_5
$\mathcal{L}_6$	$L_6/2$	0	0	$q_6 + 90$
$\mathcal{L}_7$	0	$L_6 + L_7/2$	0	q_7

Table 4-2: DH parameters for links reference frame.

For the ‘‘fixed based’’ manipulator we are going to use the following parameters:

Joint Reference frame	a	d	α	θ_i
1	0	0	$\pi/2$	q_1
2	0	$L_1 + L_{offset}$	$\pi/2$	$q_2 + 90$
3	0	$L_3 + L_2$	$\pi/2$	q_3
4	0	L_{offset_2}	$-\pi/2$	q_4
5	0	$L_4 + L_5$	$\pi/2$	q_5
6	0	0	$-\pi/2$	q_6
7	0	$L_6 + L_7$	0	q_7

Table 4-3: DH parameters for joints reference frame.

The tables have been obtained using the Siciliano convention as it is reported in [21].

4.3 Control strategy of the manipulator system.

The debris capture procedure start with the approach operation, where the chaser spacecraft performs a manoeuvre to ensure that the target is in the working operational space of manipulator. Then the deployment of the robotic arm is executed using a joint position control, setting the starting q_0 confiuration. Then the capture of the target is performed through the following algorithm starting from the initial configuration.

The trajectory planning of the end effector is executed with the following strategy:

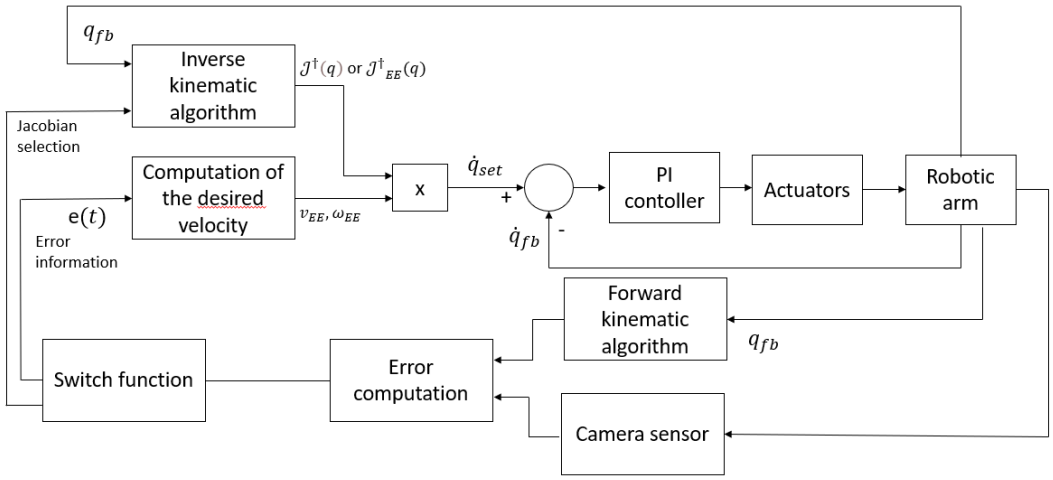

Figure 4.8: control scheme for fixed base.

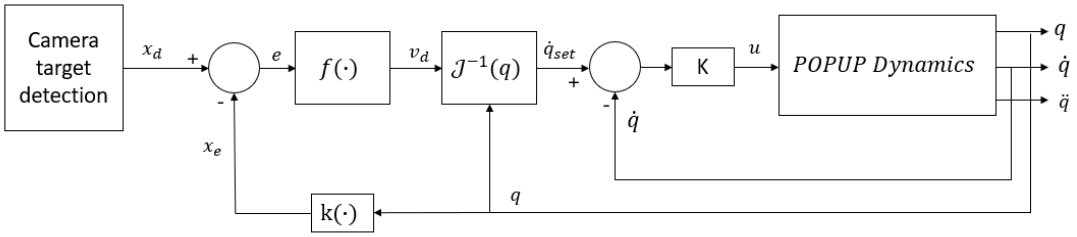

Figure 4.9: Control strategy with differential kinematics.

As can be seen the trajectory planning is executed in the operational space defining a velocity reference signal generated by the computation of the desired velocity and it is multiplied by the pseudo-inverse as considered in inverse kinematic algorithm section 4.3.1, then the velocity control loop is obtained through the feedback of the joint velocities from mechanical model of the robot, the error so generated is fed to the PI controller that generates the current reference signal for the actuators.
The POPUP robotic arm is equipped with two cameras, one is positioned on the end-effector, and it allows the system to detect the grasping point position and to execute the grasping during the final part of the manoeuvre, the other one is positioned on the base of the spacecraft and allow to drive the end effector in the first part of the grasping manoeuvre detecting the grasping point with respect to the base frame. The choice allows to successfully catch the target avoiding the loss of the mapping between the base and the end effector due to the links deformations.
Therefore, the manoeuvre can be divided in two parts, the first one is accomplish by the base camera which drive the end-effector towards the grasping point until the end effector camera do not detect the position of the grasping point, as the grasping point enter in the range of the camera then the switch is executed and the Jacobian change from the one referred to the base frame to the end effector frame (section 4.3.1).
In the figure 4.8 is shown that the error can be obtained from the difference between the desired value and the information evaluated from direct kinematic algorithm, after that the desired velocity is computed according to a certain function f, lastly the velocity joint reference is obtained by inverse kinematics algorithm.

Where $x_d = [p_d, \varphi_d]^T$ is the target pose where p_d is the position and φ_d is the orientation, they are assumed to be already estimated during the rendezvous operation.

4.3.1 Inverse kinematic algorithm for fixed-base SMS.

For the case where the attitude and the position of the spacecraft are controlled, fixed based algorithms can be used, in this case suppose that a motion trajectory is assigned to the ed effector in terms of velocity v_e and ω_e, the aim of such algorithm is to determine a feasible joint trajectory ($q(t)$, $\dot{q}(t)$), that reproduces the given trajectory.
The joint velocities can be obtained by simply inverting the Jacobian matrix that in this case is not square since the robotic arm is redundant, when the manipulator is redundant a solution method is to formulate the problem as a constrained linear optimization problem and solve it with Lagrange multipliers method [21].
So, we need to compute the pseudoinverse:

$$\dot{q} = \mathcal{J}^{\dagger}(q) v_e \tag{4.18}$$

Thus, the Jacobian allows us to express a linear mapping between the joint space and the operational space. Where the pseudoinverse is calculated as:

$$\mathcal{J}^{\dagger} = \mathcal{J}^T (\mathcal{J}\mathcal{J}^T)^{-1} \tag{4.19}$$

The joint solution locally minimizes the norm of joint velocities, that is the benefit of using a redundant robotic arm, indeed the pseudo inverse uses the redundant degree of freedom to minimize the velocity, other kind of choices could be the use of the damped pseudo-inverse which allow to avoid the kinematics singularities. The Jacobians utilized in the Simulink model are two, one is referred to the base frame and the other one referred to the EE frame.
The Jacobian referred to the base frame 0 is obtained as follows:

$$\mathcal{J} = \begin{bmatrix} \hat{k}_0 \times p_{EE} & \hat{k}_1 \times (p_{EE} - p_1) & \dots & \hat{k}_6 \times (p_{EE} - p_6) \\ \hat{k}_0 & \hat{k}_1 & \dots & \hat{k}_6 \end{bmatrix} \tag{4.20}$$

Where we have:

$$\hat{k}_0 = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}, and\ \hat{k}_i = {}^0R_{J_i} \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \ \forall\ 1 \leq i \leq 6$$

$$p_i = {}^0A_i \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} \ \forall\ 1 \leq i \leq 6$$

$$p_{EE} = {}^{0}A_{EE}\begin{bmatrix}0\\0\\0\\1\end{bmatrix}$$

This allows us to extract the third column from the rotation matrix and the fourth column from the HTM. The Jacobian is going to be a 6×7 matrix, where 6 is the number of controllable DOF in the operational space and 7 the number of DOF in joint space.
To determine the Jacobian matrix referred to the End effector frame we can proceed in this way, given:

$${}^{0}\tilde{A}_{EE} = {}^{0}\tilde{A}_{7} = \begin{bmatrix} {}^{0}A_{7}{}^{T} & 0_{3\times3} \\ 0_{3\times3} & {}^{0}A_{7}{}^{T} \end{bmatrix} \tag{4.21}$$

Thus, we obtain:

$$\mathcal{J}_{EE} = {}^{EE}\tilde{A}_{0} * \mathcal{J} \tag{4.22}$$

The information of position and orientation of every joint is obtained with a recursive approach:

$$\begin{gathered}
{}^{0}A_{2} = {}^{0}A_{1}\,{}^{1}A_{2} \\
{}^{0}A_{3} = {}^{0}A_{1}\,{}^{1}A_{2}\,{}^{2}A_{3} \\
{}^{0}A_{4} = {}^{0}A_{1}\,{}^{1}A_{2}\,{}^{2}A_{3}\,{}^{3}A_{4} \\
{}^{0}A_{5} = {}^{0}A_{1}\,{}^{1}A_{2}\,{}^{2}A_{3}\,{}^{3}A_{4}\,{}^{4}A_{5} \\
{}^{0}A_{6} = {}^{0}A_{1}\,{}^{1}A_{2}\,{}^{2}A_{3}\,{}^{3}A_{4}\,{}^{4}A_{5}\,{}^{5}A_{6} \\
{}^{0}A_{7} = {}^{0}A_{1}\,{}^{1}A_{2}\,{}^{2}A_{3}\,{}^{3}A_{4}\,{}^{4}A_{5}\,{}^{5}A_{6}\,{}^{6}A_{7}
\end{gathered}$$

4.3.2 Visual servoing.

Vision plays a key role in a robotic system, as it can be used to obtain geometrical and qualitative information on the environment where the robot operates, without physical interaction. Such information may be employed by the control system at different levels, for the sole task planning and for feedback control.
As well known, due to the refraction phenomenon, the point in the camera frame is transformed into a point in the image plane via perspective transformation:

$$X_f = -\frac{f p_x^c}{p_z^c} \tag{4.23}$$

$$Y_f = -\frac{f p_y^c}{p_z^c} \tag{4.24}$$

Where (X_f, Y_f), are the new coordinates in the frame defined on the image plane, and f is the focal length of the lens. While p_x^c, p_y^c, p_z^c, are the coordinates of a point of an object in a frame attached to the camera.

In this thesis the image processing is neglected and is assumed that the camera feed directly the needed information.
Using the transformation sensor available in the multibody library, the information transmitted by the camera block on the end-effector are the distance between the end effector frame and the target frame and the orientation, expressed in quaternion, of the grasping point reference frame with respect to the end effector frame, they can be defined as ${}^{EE}p_T$ and ${}^{EE}Q_T$. While the camera on the base spacecraft feed the information of position of the target frame with respect to the base frame and its orientation defined as ${}^{B}Q_T$ and ${}^{B}p_T$.
To feed the ''computation of the desired velocity'' block with the orientation and position error the camera block implements the following function. Let us define first the quaternion representing the difference between the TCP frame and camera frame $Q_d = [1\ 0\ 0\ 0] = [\eta_1, \epsilon_1]$, and the two quaternions from the plant ${}^{EE}Q_T$ and ${}^{EE}Q_T^*$ defined as:

$$ {}^{EE}Q_T = [\eta_2\ , \boldsymbol{\epsilon_2}] \tag{4.25} $$

$$ {}^{EE}Q_T^* = {}^{EE}Q_T * Q_\psi = [\eta_3\ , \boldsymbol{\epsilon_3}] \tag{4.26} $$

Where:

$$ \eta = \cos\left(\frac{\vartheta}{2}\right) $$

$$ \boldsymbol{\epsilon} = \sin\left(\frac{\vartheta}{2}\right)\boldsymbol{r} $$

$$ Q_\psi = [0\ 0\ 0\ 1] $$

Where Q_ψ corresponds to a -180º rotation around z direction of the target fame. The quaternion error can be obtained from the quaternion product:

$$ Q_{err} = Q_d * {}^{EE}Q_T{}^{-1} = \{\eta_1\epsilon_2 - \epsilon_1\eta_2 - \epsilon_1 \times \epsilon_2\} = \epsilon_{err} \tag{4.27} $$

$$ Q_{err}^* = Q_d * {}^{EE}Q_T^{*\,-1} = \{\eta_1\epsilon_3 - \epsilon_1\eta_3 - \epsilon_1 \times \epsilon_3\} = \epsilon_{err}^* \tag{4.28} $$

Where the cross product can be represented by the skew symmetric matrix:

$$ S(\epsilon_1 = \epsilon_d) = \begin{bmatrix} 0 & -\epsilon_1(3) & \epsilon_1(2) \\ \epsilon_1(3) & 0 & -\epsilon_1(1) \\ -\epsilon_1(2) & \epsilon_1(1) & 0 \end{bmatrix} \tag{4.29} $$

$$ \epsilon_1 \times \epsilon_2 = S(\epsilon_1 = \epsilon_d)\epsilon_2 \tag{4.30} $$

After having define the quaternion errors, the information transmitted is the vectorial part, specifically the block allows to select the shortest path between the two quaternions, in a way that the end effector would reach the target reference orientation in less time. Therefore, the error transmitted is:

$$\epsilon_{delta} = \begin{cases} \epsilon_{err}\ if\ |\epsilon_{err}| < |\epsilon^*_{err}\ | \\ \epsilon^*_{err}\ if\ |\epsilon_{err}| \geq |\epsilon^*_{err}\ | \end{cases} \quad (4.31)$$

The second camera, which accomplish the first part of the maneuver, provides the position of the target reference frame with respect to the base frame and the orientation, in this case the camera block process the information and derives the errors to be sent to auto-control block where the detection function is computed. In terms of position, we obtain ${}^{B}p_{T}$, therefore:

$${}^{B}p_{T}^{EE} = {}^{B}p_{T} - {}^{B}p_{EE} \quad (4.32)$$

$${}^{B}p_{EE} = {}^{B}A_{0}\ {}^{0}A_{7} \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}$$

$${}^{B}A_{0} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & offset \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Where ${}^{B}p_{T}^{EE}$ is the relative position between the target and the end effector frame expressed in base frame components.
Moreover, the camera provides the quaternion that describes the orientation of the grasping point frame in the base frame, the following error can be obtained with the quaternion product (d: desired quaternion, T: target):

$$Q_{d} = {}^{B}Q_{T}$$

Where ${}^{B}Q_{T}$ is the information from the camera,

$$Q_{err} = Q_{d} * {}^{B}Q_{EE}{}^{-1} \quad (4.33)$$

With ${}^{B}Q_{EE}$ that express the orientation of the end effector in the base frame and is obtained using the forward kinematics algorithm:

$${}^{B}R_{7} = {}^{B}A_{7}(1{:}3,1{:}3) = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} \quad (4.34)$$

Where was used the MATLAB notation to extract the Rotation matrix, after this one is obtained the quaternion feedback is calculated from the R matrix:

$$Q_{fb} = {}^{B}Q_{EE} = [\eta_{fb}, \epsilon_{fb}] \quad (4.35)$$

$$\eta_{fb} = \frac{1}{2}\sqrt{r_{11} + r_{22} + r_{33} + 1} \quad (4.36)$$

$$\epsilon_{fb} = \frac{1}{2}\begin{bmatrix} sgn(r_{32} - r_{23})\sqrt{r_{11} - r_{22} - r_{33} + 1} \\ sgn(r_{13} - r_{31})\sqrt{r_{22} - r_{33} - r_{11} + 1} \\ sgn(r_{21} - r_{12})\sqrt{r_{33} - r_{11} - r_{22} + 1} \end{bmatrix} \quad (4.37)$$

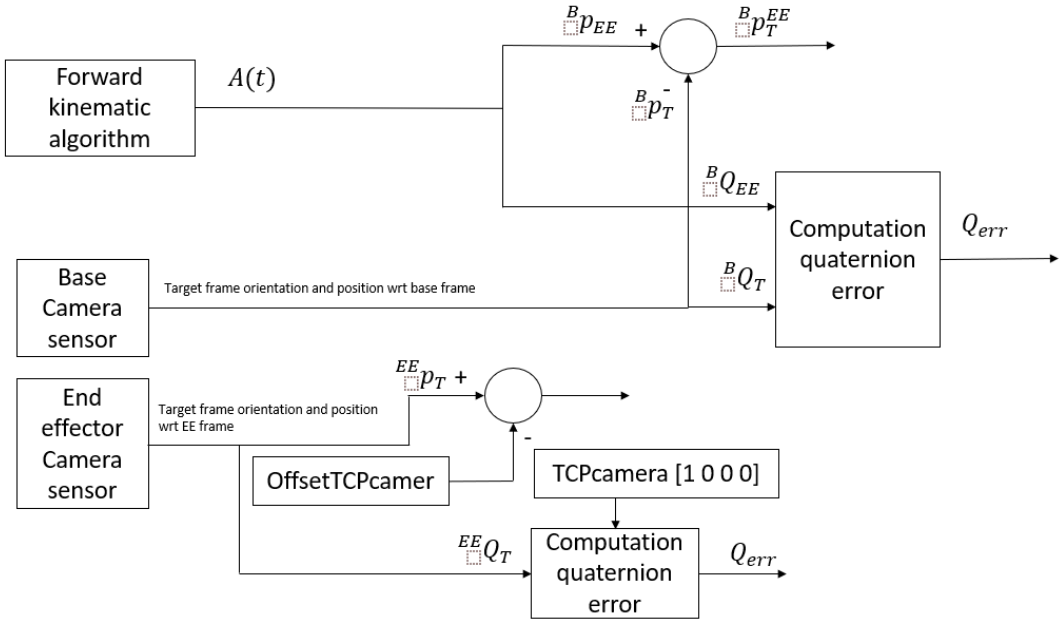

Figure 4.10: Visual servoing function.

4.3.3 Computation of desired velocity.

The desired linear velocity is computed in order to have a trapezoidal profile [63]:

$$\dot{p}_d = \min\left(a_0 t, v_0, \sqrt{2a_0\|\Delta p\|}\right)\frac{\Delta p}{\|\Delta p\|} \tag{4.38}$$

$$\Delta p = p_d - p_e$$

Where a_0 is the maximum linear acceleration, v_0 is the maximum linear velocity and Δp is the error position between the desired position and the end effector position. While the orientation is computed from the quaternion error already explained in the previous section in the following way [63]:

$$\dot{\varphi}_d = \min\left(\alpha_0 t, \omega_0, \sqrt{2\alpha_0\|\Delta\epsilon\|}\right)\frac{\Delta\epsilon}{\|\Delta\epsilon\|} \tag{4.39}$$

4.3.4 Control loop for motor actuation joints.

The control strategy used to for the robot joint is the decentralized control which consider the manipulator as formed of n independent systems (n joint) and control each joint as a single input/output system, furthermore an operational space control scheme is adopted, as reported in figure 4.10 [21], where measurement of operational space variables is often performed not directly.

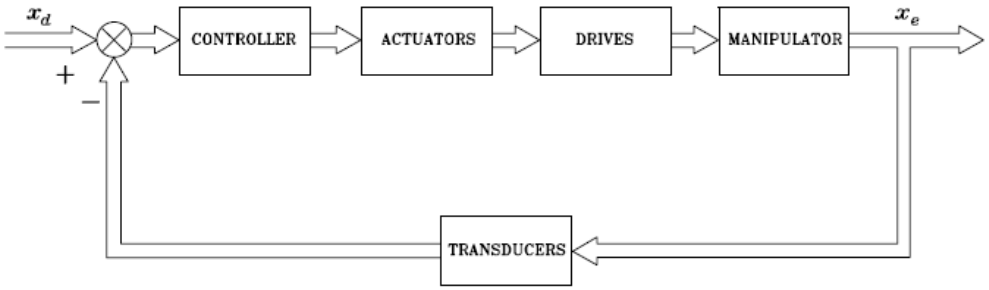

Figure 4.11:Control loop in operational space.

The robotic arm is controlled using the control logic scheme with PI controller:

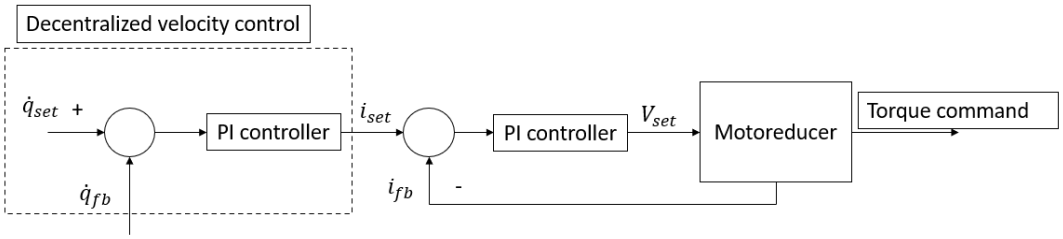

Figure 4.12: Control logic for joints.

As can be seen, during the capturing phase, the joint degree of freedom is controlled in velocity using the feedback of the joint rate from the POPUP Dynamics block, therefore the control is composed by a velocity external loop and an inner current

loop, the error generated from both are used in the first case to generate a reference signal for the current and in the second case a voltage reference signal that is send to the moto reducer.
The trajectory planning of the robotic arm is executed defining the velocity profile as expressed in the previous sections, they are converted in joint rate from the invers kinematic algorithm.
The motor is implemented in Simulink as a mechatronic system with the transfer functions, well known from literature.
The torque obtained is fed to the revolute joint in the multibody system, which is implemented between two links using the rigid transform block, the used approach is the one showed in the following figure:

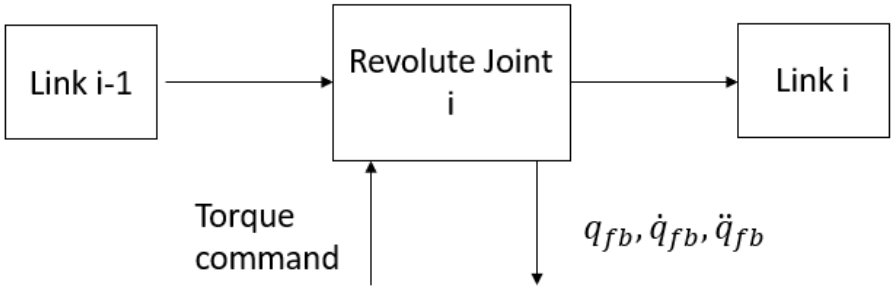

Figure 4.13: Multibody implementation.

4.4 Control strategies for free-floating.

In the free-floating case the transformation between joint-space and task-space is executed using the Generalized Jacobian Matrix that allows to map generalized joint velocities to a spatial velocity of the end effector.
To find the expression of this Jacobian matrix we can proceed following two different procedures as explained in [23,46], therefore we need to derive the Lagrange equation of motion of the free-floating system, so we can write that the Lagrange equation is equal the kinetic energy of the system:

$$L = T = \frac{1}{2}\sum_{i=0}^{N}\left({}^{\mathcal{J}}\omega_i^T\ {}^{\mathcal{J}}I_i\ {}^{\mathcal{J}}\omega_i + m_i\ {}^{\mathcal{J}}\dot{r}_i\ {}^{\mathcal{J}}\dot{r}_i\right) \tag{4.40}$$

Collecting the terms and following [23], we obtain:

$$T = \frac{1}{2}\begin{bmatrix} {}^{\mathcal{J}}\dot{x}_0^T & \dot{q}^T \end{bmatrix}\begin{bmatrix} H_0 & H_{0m} \\ H_{0m}^T & H_m \end{bmatrix}\begin{bmatrix} {}^{\mathcal{J}}\dot{x}_0 \\ \dot{q} \end{bmatrix} \tag{4.41}$$

Where:

- H_0: [6x6] Base-spacecraft inertia matrix;
- H_m: [NxN] manipulator inertia matrix.
- H_{0m}[6xN]: Dynamic-coupling inertia matrix.
- ${}^{\mathcal{J}}\dot{x}_0 = \begin{bmatrix} {}^{\mathcal{J}}\dot{x}_0^T & \dot{q}^T \end{bmatrix}$: is the combined and angular velocity matrix of the base-spacecraft.

The algebraic steps are reported in [46].

4.4.1 Base-spacecraft inertia matrix.

The [6x6] matrix result to be expressed in the following way:

$$H_0 = \begin{bmatrix} m_{tot}\mathbb{I}_{3\times3} & -m_{tot}\,{}^{\mathcal{J}}r_{0C}^{\times} \\ m_{tot}\,{}^{\mathcal{J}}r_{0C}^{\times} & H_S \end{bmatrix} \tag{4.42}$$

Where $\mathbb{I}_{3\times3}$ is the identity matrix and the H_S collects the moments-of-inertia of the spacecraft-manipulator system about the base-spacecraft centre of mass, expressed in the inertial frame as follows:

$$H_S = \sum_{i=1}^{N} \left({}^{\mathcal{J}}I_i - m_i\,{}^{\mathcal{J}}r_{0i}^{\times}\,{}^{\mathcal{J}}r_{0i}^{\times} \right) + {}^{\mathcal{J}}I_0 \tag{4.43}$$

$${}^{\mathcal{J}}I_i = {}^{\mathcal{J}}R_{Li}\,{}^{L_i}I_i\,{}^{\mathcal{J}}R_{L_i}^{T} \tag{4.44}$$

Where the $i-th$ link moments of inertia matrix in the inertial coordinate system is derived from the moments-of-inertia in the $i-th$ link coordinate system.

4.4.2 Dynamic -Coupling Inertia matrix.

The matrix H_{0m} express the contribution of the dynamic coupling between the manipulator and the base-spacecraft to the kinetic energy of the system:

$$H_{0m} = \begin{bmatrix} J_{TS} \\ H_{Sq} \end{bmatrix} \tag{4.45}$$

Where J_{TS} is a [3xN] matrix which collects the contribution to the system kinetic energy of the combination of the effect of the manipulator joint velocity $\dot{q}$ and he base-spacecraft linear velocity ${}^{\mathcal{J}}v_0$.
We have:

$$J_{TS} = \sum_{i=1}^{N} (m_i J_{Ti}) \tag{4.46}$$

Where the [3xN] matrix J_{Ti} represents the linear velocity Jacobian for the centre -of-mass of the link, and is given by:

$$J_{Ti} = \begin{bmatrix} {}^{I}k_1^{\times}\left({}^{\mathcal{J}}r_i - {}^{\mathcal{J}}p_1 \right) & \dots & {}^{I}k_i^{\times}\left({}^{\mathcal{J}}r_i - {}^{\mathcal{J}}p_i \right) & 0_{3,N-i} \end{bmatrix} \tag{4.47}$$

$$\forall\ (1 \le i \le N)$$

The [3xN] matrix H_{Sq} contains the contribution to the system kinetic energy of the combination of the effect of the manipulator joint velocity $\dot{q}$ and he base-spacecraft angular velocity ${}^{\mathcal{J}}\omega_0$:

$$H_{Sq} = \sum_{i=1}^{N} \left({}^{\mathcal{J}}I_i J_{Ri} + m_i \, {}^{\mathcal{J}}r_{0i}^{\times} J_{Ti} \right) \tag{4.48}$$

Where:

$$J_{Ri} = \left[{}^{\mathcal{J}}k_1 \quad \dots \quad {}^{\mathcal{J}}k_i \quad 0_{3,N-i} \right], \forall \, (1 \leq i \leq N) \tag{4.49}$$

4.4.3 Manipulator Inertia Matrix.

The [NxN] inertia matrix H_m, is like the one used for the fixed base manipulator:

$$H_{0m} = \sum_{i=1}^{N} \left(J_{Ri}^{T} \, {}^{\mathcal{J}}I_i J_{Ri} + m_i J_{Ti}^{T} J_{Ti} \right) \tag{4.50}$$

4.4.4 GJM.

Can be shown that the velocity of a certain point X_i can be expressed by the following relationship:

$$\begin{bmatrix} v_{X_i} \\ \omega_i \end{bmatrix} = J_{0Xi} \dot{x}_0 + J_{mXi} \dot{q} \tag{4.51}$$

Where the contribution of the joint rates to generate a velocity of X_i is expressed by the [6xN] manipulator Jacobian matrix:

$$J_{mXi} = \begin{bmatrix} \hat{k}_1^{\times}(x_i - p_1) & \dots & \hat{k}_i^{\times}(x_i - p_i) & 0_{3,N-i} \\ \hat{k}_1 & \dots & \hat{k}_i & 0_{3,N-i} \end{bmatrix} \tag{4.52}$$

Where x_i is the position of the point X_i. The contribution of the base-spacecraft combined velocity is expressed by the [6x6] base-spacecraft Jacobian:

$$J_{0Xi} = \begin{bmatrix} E & -x_{0i}^{\times} \\ 0_{3,3} & E \end{bmatrix} \tag{4.53}$$

Where:

$$x_{0i} = x_i - p_0 \tag{4.54}$$

Using the generalized form, we can find that:

$$\begin{bmatrix} v_{X_i} \\ \omega_i \end{bmatrix} = J_{X_i}^{*} \dot{q} \tag{4.55}$$

It follows that:

$$J_{X_i}^{*} \dot{q} = J_{0Xi} \dot{x}_0 + J_{mXi} \dot{q} \tag{4.56}$$

Using the equation of motion, in the form [23,47], one can write the following momentum equation:

$$\begin{bmatrix} P \\ L \end{bmatrix} = H_0 \dot{x}_0 + H_{0m} \dot{q} = M_0 \tag{4.57}$$

Where P and L are respectively the linear and angular momentum and the $M_0 = 0_{6,1}$ is the combined matrix, because of the hypothesis for the free-floating case this one be set to zero. So, we can obtain:

$$\dot{x}_0 = -H_0^{-1} H_{0m} \dot{q} \tag{4.58}$$

So, the generalized Jacobian can be defined as follow:

$$J_{X_i}^* = J_{mXi} - J_{0Xi} H_0^{-1} H_{0m} \tag{4.59}$$

Furthermore, if X_i coincide with the end effector the resulting Jacobian matrix becomes the generalized that can be used in an inverse kinematic algorithm to determine the joint rate:

$$\dot{q} = J^{*-1} \begin{bmatrix} v_{EE} \\ \omega_{EE} \end{bmatrix} \tag{4.60}$$

The inverse of the H_0 matrix is computed using the Banachiewicz inversion formula [23,48], and the Shur complement S_U of the non-singular submatrix U. The matrix is partitioned in the form:

$$H_0 = \begin{bmatrix} m_{tot} \mathbb{I}_{3\times3} & -m_{tot}\, {}^{\mathcal{J}} r_{0C}^{\times} \\ m_{tot}\, {}^{\mathcal{J}} r_{0C}^{\times} & H_S \end{bmatrix} = \begin{bmatrix} U & V \\ W & X \end{bmatrix} \tag{4.61}$$

$$H_0^{-1} = \begin{bmatrix} \frac{1}{m_{tot}} \mathbb{I}_{3\times3} - r_{0C}^{\times} S_U^{-1} r_{0C}^{\times} & r_{0C}^{\times} S_U^{-1} \\ S_U^{-1} r_{0C}^{\times} & S_U^{-1} \end{bmatrix} \tag{4.62}$$

With:

$$S_U = X - W U^{-1} V = H_S + m_{tot} r_{0C}^{\times} r_{0C}^{\times} \tag{4.63}$$

4.4.5 Control scheme for free-floating case.

As done for the controlled base motion, a similar approach has been adopted here to build the control scheme for the free-floating case. As can be seen in the following figure, the desired end effector poses and the feedback from the camera sensor are used to obtain an error in terms of position and orientation, this information is then used to compute the desired linear and angular velocity which are then multiply with the GJM to obtain the velocity reference signal for the joint motors. The control scheme of the motor has not been changed. The output of the control system is the torque command for the joints which allows the manipulator movement, therefore the base of the spacecraft will freely move accordingly to the dynamic coupling between the manipulator and the base, the attitude and position are not controlled. The control strategy is the same as already explained in the control strategy section.

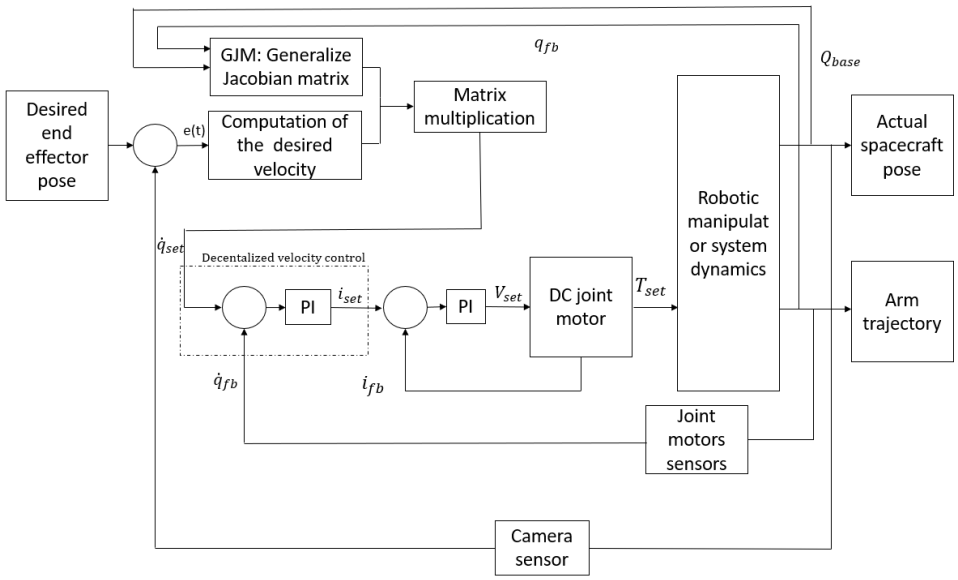

Figure 4.14: Control scheme for free-floating.

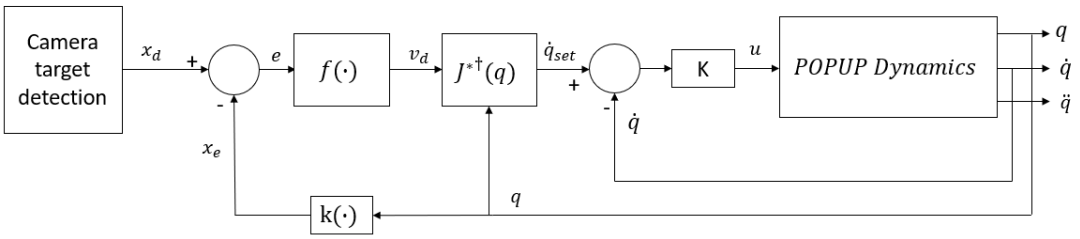

Figure 4.15: Control strategy with differential kinematics for free-floating.

As we focused on, in the previous sections are reported the relationships utilized to write the kinematic algorithm to obtain the GJM to perform the grappling phase, therefore the code needs some inputs to accomplish the operations. These can be found in the mathematical formulations of the previous sections.
First, we need the principal moment of inertia matrices expressed in the $\mathcal{L}_i$ frames, which are then transformed in the inertial frame, given in $kg * m^2$:

$$^{\mathcal{L}_1}I_1 = \begin{bmatrix} 0.9875 & 0 & 0 \\ 0 & 0.8641 & 0 \\ 0 & 0 & 0.8641 \end{bmatrix} e-03$$

$$^{\mathcal{L}_2}I_2 = \begin{bmatrix} 0.9875 & 0 & 0 \\ 0 & 0.8641 & 0 \\ 0 & 0 & 0.8641 \end{bmatrix} e-03$$

$$^{\mathcal{L}_3}I_3 = \begin{bmatrix} 41.0432 & 0 & 0 \\ 0 & 0.1801 & 0 \\ 0 & 0 & 41.0432 \end{bmatrix}$$

$$^{\mathcal{L}_4}I_4 = \begin{bmatrix} 0.1801 & 0 & 0 \\ 0 & 41.0432 & 0 \\ 0 & 0 & 41.0432 \end{bmatrix}$$

$$^{\mathcal{L}_5}I_5 = \begin{bmatrix} 0.0009 & 0 & 0 \\ 0 & 0.0007 & 0 \\ 0 & 0 & 0.0012 \end{bmatrix}$$

$$^{\mathcal{L}_6}I_6 = \begin{bmatrix} 0.001 & 0 & 0 \\ 0 & 0.0009 & 0 \\ 0 & 0 & 0.0009 \end{bmatrix}$$

$$^{\mathcal{L}_7}I_7 = \begin{bmatrix} 0.004 & 0 & 0 \\ 0 & 0.0026 & 0 \\ 0 & 0 & 0.0033 \end{bmatrix}$$

$$^{0}I_0 = \begin{bmatrix} I_{xx} & 0 & 0 \\ 0 & I_{yy} & 0 \\ 0 & 0 & I_{zz} \end{bmatrix}$$

Where the last matrix is the principal inertia matrix of the base spacecraft with respect to the base frame aligned along the principal axes that depends on the mass property and that is automatically computed.
Other information needed are the mass properties and the length of every link:

Links	Mass [kg]	Length [m]
Base-Spacecraft	Variable (500,80,1000)	2 (in x y z directions)
$\mathcal{L}_1$	790e-3	0.075
$\mathcal{L}_2$	790e-3	0.075
$\mathcal{L}_3$	28.7022	4
$\mathcal{L}_4$	29.1595	4
$\mathcal{L}_5$	0.9314	0.05
$\mathcal{L}_6$	0.8673	0.05
$\mathcal{L}_7$	1(payload)	0.05

Table 4-4: Physical characteristics.

After having defined the manipulator characteristics, we need to find the inertial frame, in this case its origin has been considered in the COM of the target spacecraft or debris, and the x, y and z directions are selected to coincide with the predefined world frame in Simscape environment.
In the free-floating case, the needed information from the camera sensor device (visual servoing function) are the end effector position and orientation in the inertial or world frame, and the position and orientation of grasping point with respect to the world frame.

4.5 Control strategies for Rotation-floating.

Rotation floating is a flight mode where the control action is applied to the manipulator motion and the attitude of the spacecraft base. The base is left free to translate in the inertial space in consequence of the manipulator movement. As already said in the previous sections, in this case internal torques are applied using reaction wheels. As can be found in [59], the system dynamics of the combined system can be found using Euler-Lagrange formulation, similarly at what is doe in the previous section, the Lagrange equation can be written as:

$$L = T = \frac{1}{2}\sum_{i=0}^{N}\left({}^{\mathcal{J}}\omega_i^T \, {}^{\mathcal{J}}I_i \, {}^{\mathcal{J}}\omega_i + m_i \, {}^{\mathcal{J}}\dot{r}_i \, {}^{\mathcal{J}}\dot{r}_i\right) + \frac{1}{2}\left[\omega_{r\omega}^T I_{r\omega}\omega_{r\omega} + m_{r\omega}\dot{r}_{r\omega}^T\dot{r}_{r\omega}\right] \tag{4.64}$$

Where $\omega_{r\omega}$ is the angular velocity of the reaction wheel, $I_{r\omega}$ is the inertia of the reaction wheel, $m_{r\omega}$ is the reaction wheel mass and $\dot{r}_{r\omega}$ is the translational velocity of the reaction wheel. The second term of the equation represents the kinetic energy of the reaction wheel. As done in the previous section, the dynamics equation can be derived using the differential equations [59]:

$$\begin{gathered}\frac{d}{dt}\left(\frac{\delta L}{\delta \dot{x}_0}\right) - \frac{\delta L}{\delta x_0} = 0_{6\times 1} \\ \frac{d}{dt}\left(\frac{\delta L}{\delta \dot{q}}\right) - \frac{\delta L}{\delta q} = \tau \\ \frac{d}{dt}\left(\frac{\delta L}{\delta \omega_{r\omega}}\right) - \frac{\delta L}{\delta \theta_{r\omega}} = \tau_{r\omega}\end{gathered} \tag{4.65}$$

Where the first equation is obtained with respect to the $x_0 = [r_0, \theta_0]$, the degree of freedom of the base of the spacecraft, on which no external forces are applied. Then we have the joint degree of freedom and the torque motor, q and τ. The last relationship is relative to the reactions wheel where $\tau_{r\omega}$ is the internal toque.
Solving the above differential equation equations, the resulting dynamics can be written in the following second order simplified form:

$$M(q_p)\ddot{q}_p + C(q_p, \dot{q}_p) = u \tag{4.66}$$

$$q_p = [x_0, q, \theta_{r\omega}]^T$$

$$u = [0, \tau, \tau_{r\omega}]^T$$

Rearranging the Lagrange equation, we obtain [59]:

$$T = \frac{1}{2}\begin{bmatrix} {}^{J}\dot{x}_0^T & \dot{q}^T & {\theta_{r\omega}}^T \end{bmatrix}\begin{bmatrix} H_0 & H_{0m} & H_{0r} \\ H_{0m}^T & H_m & 0 \\ {H_{0r}}^T & 0 & H_r \end{bmatrix}\begin{bmatrix} {}^{J}\dot{x}_0 \\ \dot{q} \\ \theta_{r\omega} \end{bmatrix} \quad (4.67)$$

Where the matrix is the same expressed in the previous section except for H_{0r} that express the dynamic coupling between the base and the reaction wheel, and H_0 that is the inertia of the reaction wheel.

$$H_{0r} = \left[0_{3\times3}; I_{rwx}, 0,0; 0, I_{rwy}, 0; 0,0, I_{rwz}\right]$$

Where $I_{rwx}, I_{rwy}, I_{rwz}$ are the inertia value of the reaction wheel along the rotation axes in the specific designed configuration.
To express the generalized Jacobian, we can write the momentum conservation relation:

$$\begin{bmatrix} P \\ L \end{bmatrix} = H_0\dot{x}_0 + H_{0m}\dot{q} + H_{0r}\omega_{r\omega} \ = M_0 = 0_{6\times1} \quad (4.68)$$

Therefore, we have:

$$\dot{x}_0 = -H_0^{-1}H_{0m}\dot{q} - H_0^{-1}H_{0r}\omega_{r\omega} \quad (4.69)$$

As already outlined the relationship between the velocity of the end effector and the velocity in the joint space is:

$$\begin{bmatrix} v_E \\ \omega_E \end{bmatrix} = J_0\dot{x}_0 + J_m\dot{q} \quad (4.70)$$

Substituting we get:

$$\begin{bmatrix} v_E \\ \omega_E \end{bmatrix} = J_{0EE}(-H_0^{-1}H_{0m}\dot{q} - H_0^{-1}H_{0r}\omega_{r\omega}) + J_{mEE}\dot{q} \quad (4.71)$$

$$\begin{bmatrix} v_E \\ \omega_E \end{bmatrix} = (J_{mEE} - J_{0EE}H_0^{-1}H_{0m})\dot{q} - J_{0EE}H_0^{-1}H_{0r}\omega_{r\omega} \quad (4.72)$$

In the end we obtain:

$$J_{EE}^{*} = J_{mEE} - J_{0EE}H_0^{-1}H_{0m} \quad (4.73)$$

$$J_{EE}^{\dagger}\left(\begin{bmatrix} v_E \\ \omega_E \end{bmatrix} + J_{0EE}H_0^{-1}H_{0r}\omega_{r\omega}\right) = \dot{q} \quad (4.74)$$

Those equations are then inserted in the algorithm GJM of the Simulation environment obtaining a control scheme equal to the previous one, except for the variation related to the GJM:

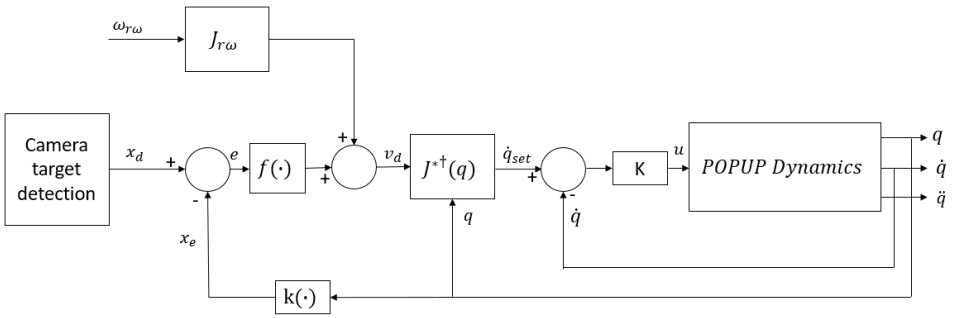

Figure 4.16: Control strategy with differential kinematics for Rotation-Floating.

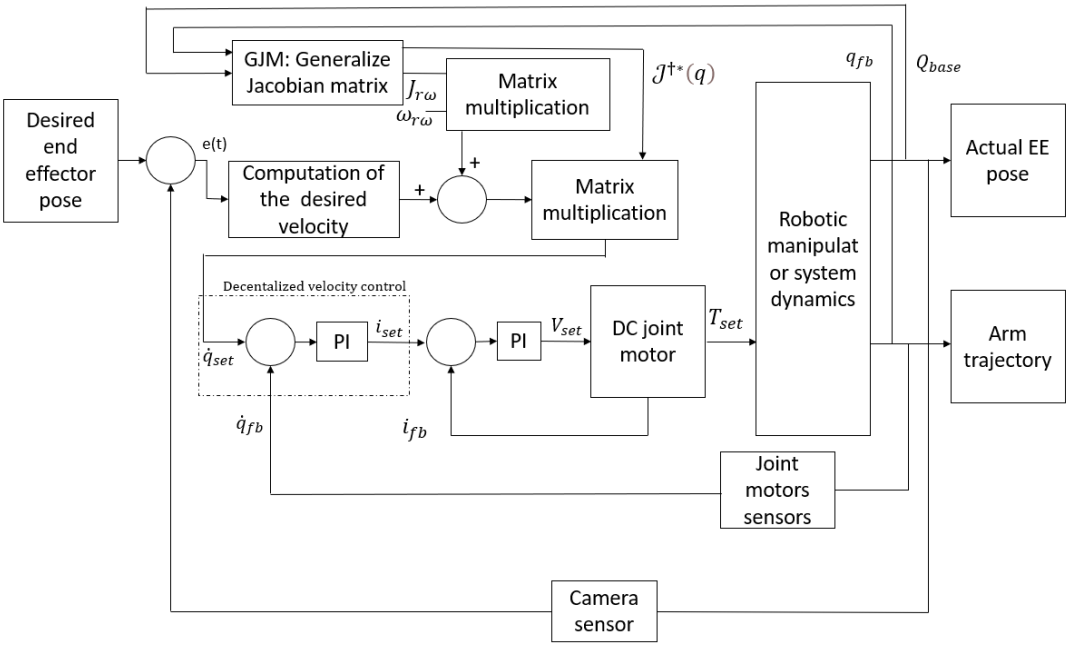

Figure 4.17: Control scheme for Rotation-floating.

CHAPTER 5

5 Simulations and results.

In the following chapter the results are reported from the Simulink simulations and are discussed in order to exploit if the actuation system has good performance and if the base of the spacecraft is stable enough to allow the use of the inverse kinematic algorithm, to obtain the joint variable in function of the desired end effector velocity, for the fixed base manipulators. For the purpose of studying this behaviour has been decided to initially observe the grasping operation while the spacecraft is arrived at the final approach point and so no translation has to performed to reach the target spacecraft, just the active position control will be used to stabilize the base during the robotic arm manoeuvring. Therefore, the attitude control is performed using thrusters or reaction wheels as expose in the previous chapters, while the position control is accomplished with the thrusters. The target is considered as non-cooperative and non-collaborative even tough is not moving in the inertial frame, this assumption is valid as long as the target is moving slowly in the inertial space. Furthermore, the simulations are computed for the different flight motion control strategies exposed in chapter 1, and graphical results are showed in the following chapter. Every simulation is obtained varying different parameter as **mass of the base of the spacecraft**, and **velocity/acceleration** of the robotic arm. To start the simulation some parameters, must be set, like the initial configurations (joint variable initial value, equilibrium position and attitude of the spacecraft):

SIMULATION DATA	**Value**
Gravity	0 (m/s^2)
Max EE velocity	$0.01\ m/s$
Max EE acceleration	$0.001\ m/s^2$
Initial joint DOF q	$[15, 55, 0, 70, 0, 45, 0] * pi/180$
Equilibrium position of the spacecraft with respect to the inertial frame	$[-7.5,0,0](m)$
Equilibrium attitude of the spacecraft with respect to the inertial frame	$q = [1\ 0\ 0\ 0]$, *pointing to the target.*

Table 5-1: Simulation Parameter.

5.1 Free-Flying case

As Highlighted in the previous chapters, in this flight mode the position and attitude control are performed using only thrusters, with respect to the translation-flying case this leads to a less accuracy in the control and a greater disturbance magnitude of the spacecraft trajectory during the robotic arm operation. In the following section different case are analysed varying the mass of the chaser spacecraft.

1) Chaser mass 500 Kg

As we can see from figure 5.1, the end effector trajectory in the inertial space should follow a straight line, but since the base is moving in response to the control action, exerted by the thrusters, and to the robotic arm disturb, the path followed is characterized by a deviation from the theoretical one. As explained in the previous chapter the first phase of the movement is achieved by the Jacobian expressed in the base frame until the switching point is reached. Notably, since the Jacobian is defined to be smooth, the switching point does not produce any important deviation on the end effector trajectory as an be noted from figure 5.1.

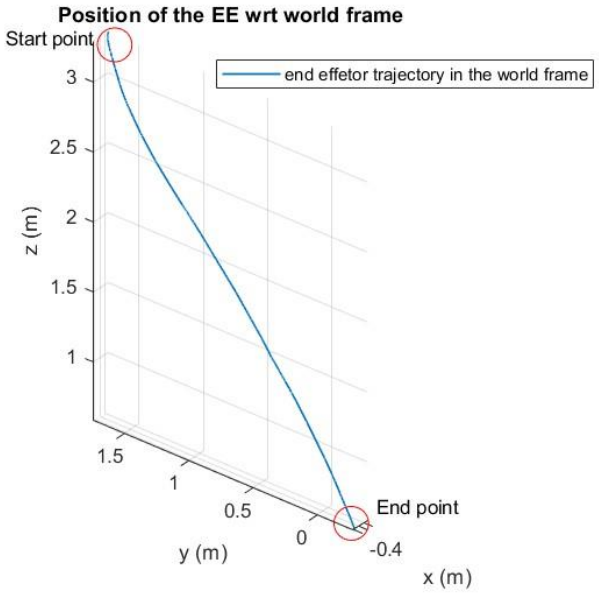

Figure 5.1: End effector trajectory.

Making a zoom we can have a look of the entire path followed by the end effector, as showed from the figure 5.2, along the trajectory there are same small deviations from the ideal linear behaviour due to the control action and the motion of the robotic arm.

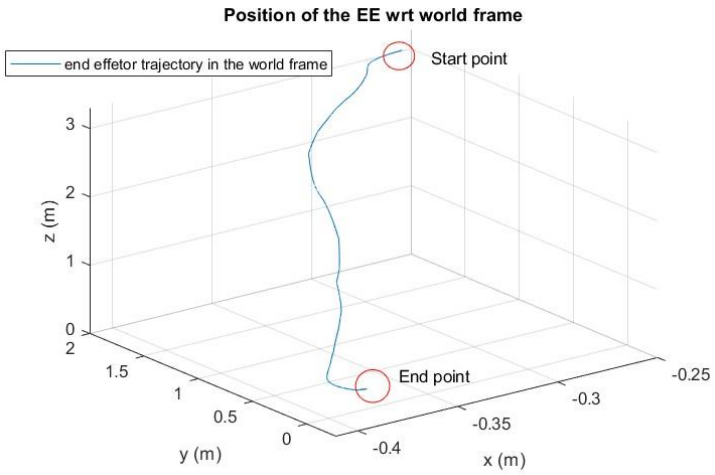

Figure 5.2: Zoom of the end effector trajectory.

Since as said, the robotic arm movement represent an external disturb for the spacecraft, on which the control action exert effort to stabilize the base around an equilibrium state variable (position and attitude), observing the position of COM in the inertial space is possible to estimate the disturb of the robotic arm movement on the spacecraft.

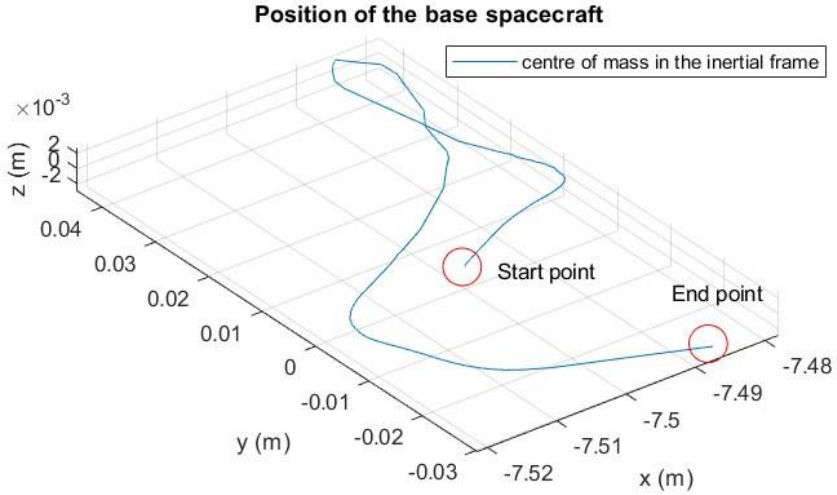

Figure 5.3: COM spacecraft in world frame.

In figure 5.4 is plotted a representation of the spacecraft's attitude using the angles roll, pitch, and yaw in time. As can be noted, the maximum peak reached is below 0.03 rad. Furthermore, at 150 s, there are not variation in the attitude due to the Jacobian switch.

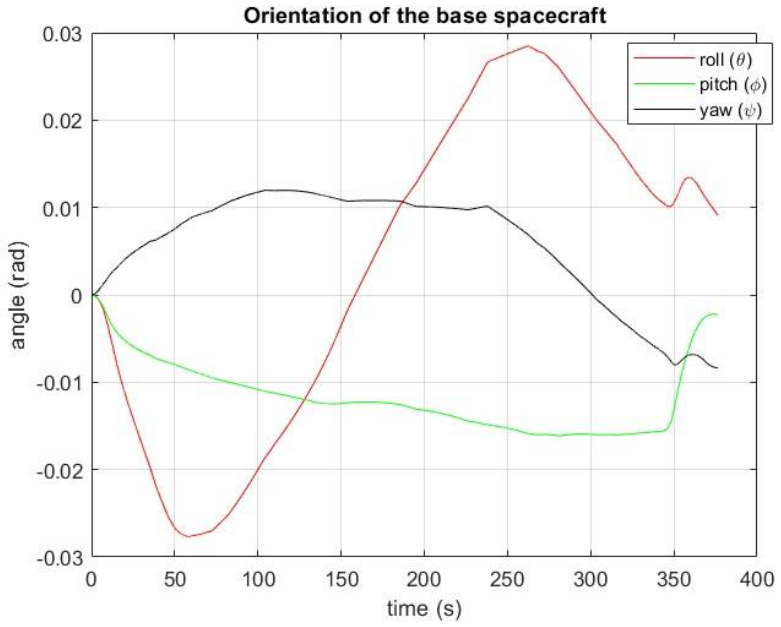

Figure 5.4: Orientation of the end effector.

Another interesting result is the one showed in figure 5.5, where the joint DOF are plotted, as can be seen around the second 150s, where the switch between the two Jacobian types is executed, a variation can be observed, however the continuity is guaranteed. Indeed, there are not instantaneous variation or discontinuity which would cause vibration and high requested torques.

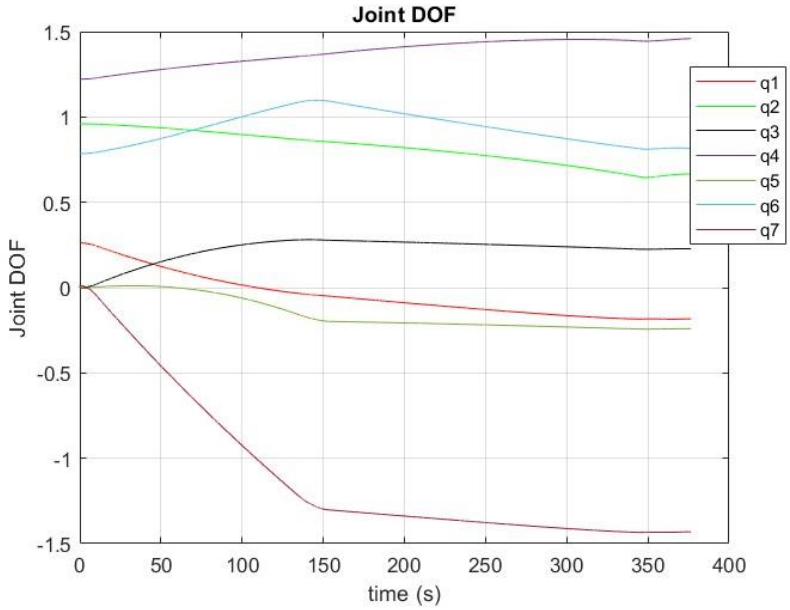

Figure 5.5: Joint DOF.

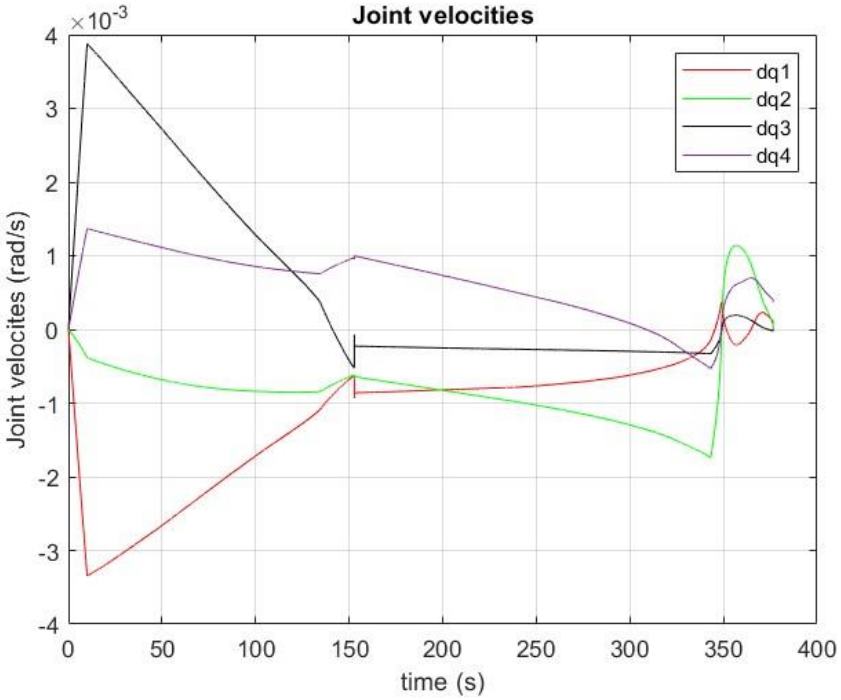

Figure 5.6: Joint velocities feedback of the first 4 joints.

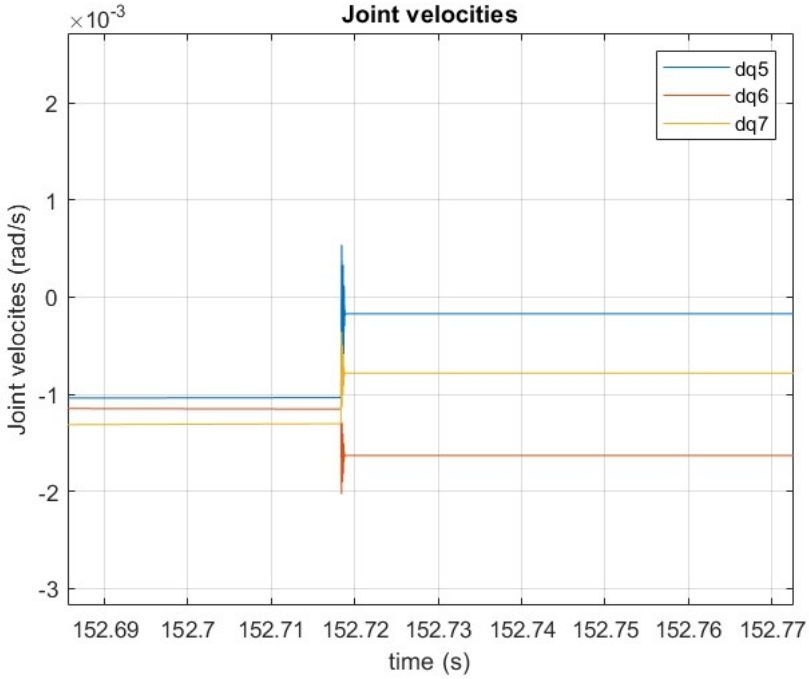

Figure 5.7: Joint velocities feedback of the last 3 joints.

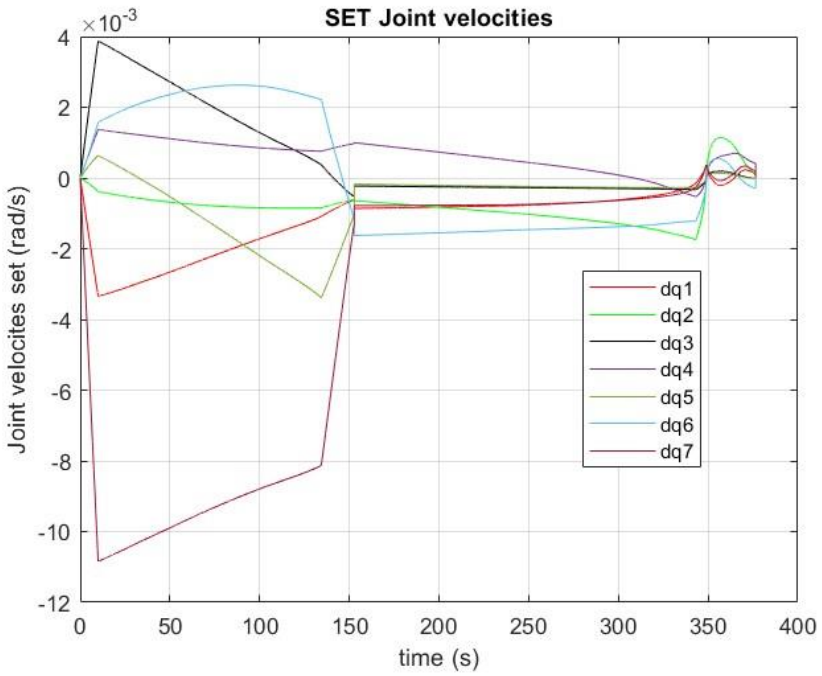

Figure 5.8: Set joint velocities.

Lastly in the above figures are plotted the joint velocities set and feedback. From the figure 5.8 can be seen that the Jacobian switch is executed at the second 150. As can be noted there is an initial phase where the joint as a high acceleration where the velocities increase rapidly, after that can be observed a decrease, due to the constant profile of the require velocity in operational space. As the switching point is reached, the velocity decreases rapidly.

2) Chaser mass 250 Kg

Decreasing the mass of the spacecraft's base, we can observe in figure 5.9 a higher deviation of the end effector trajectory from the linear path due to the lower spacecraft inertia, consequently the effort required for the spacecraft stabilization will increase.

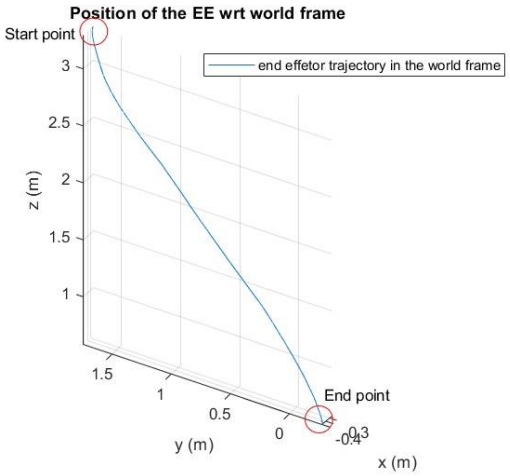

Figure 5.9: End effector trajectory.

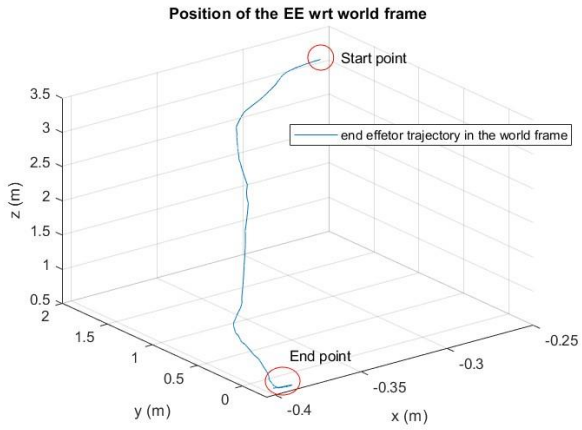

Figure 5.10: Zoom of the end effector trajectory.

Similarly to the end effector trajectory, even the COM position of the base, as can be observed in figure 5.11, is affected by a higher disturb in the inertial frame.

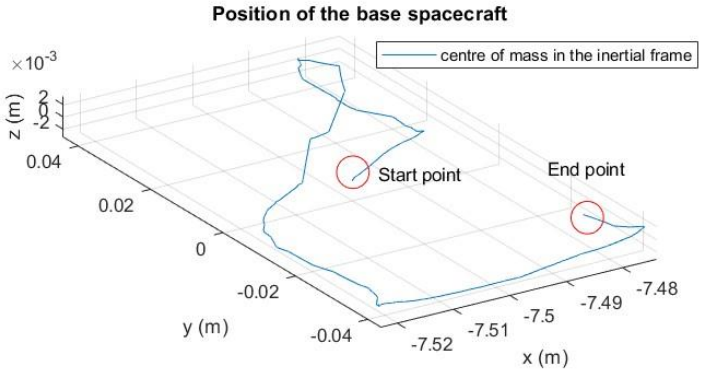

Figure 5.11: COM spacecraft in world frame.

In figure 5.12, the joint DOF are reported with respect to time.

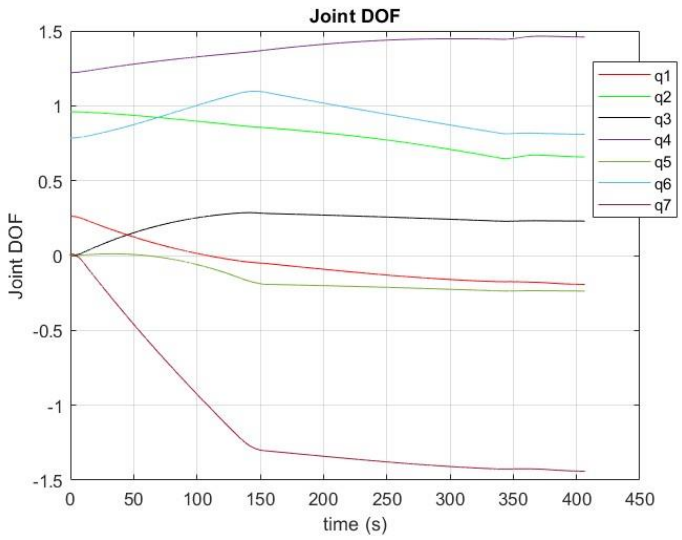

Figure 5.12: Joint DOF.

Furthermore, there are small differences between the diagram representing the joint DOF and velocities with respect to the previous case.

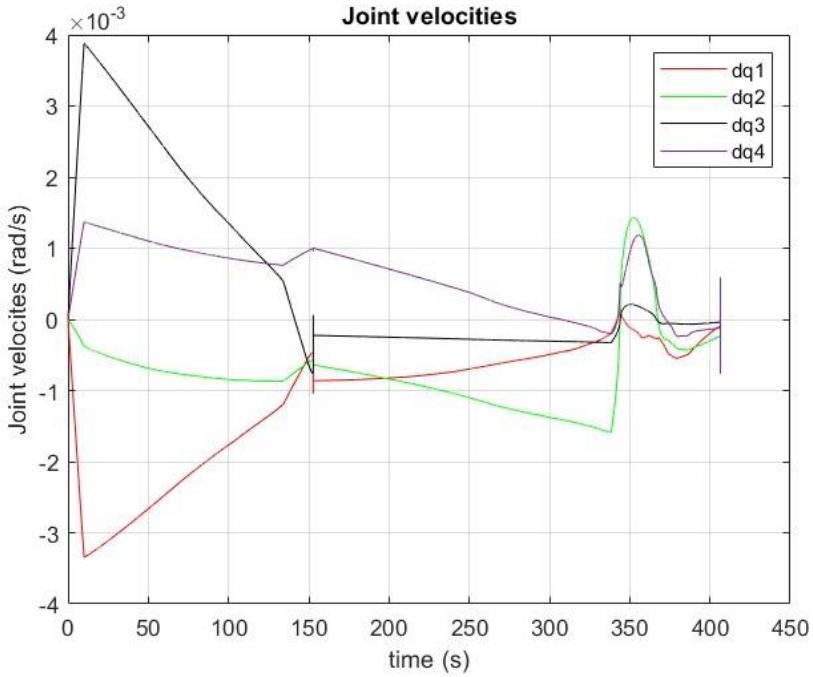

Figure 5.13: Joint velocities feedback of the first 4 joints.

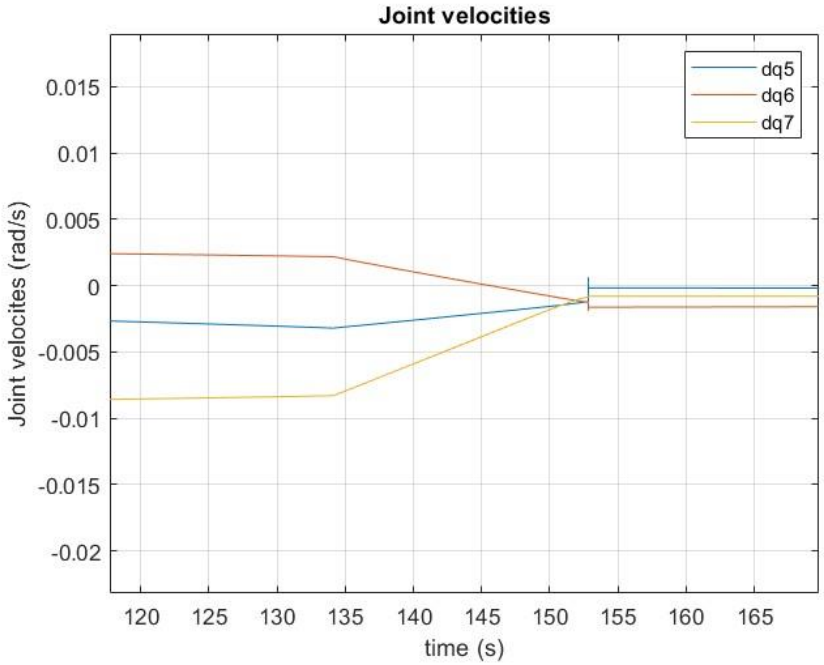

Figure 5.14: Joint velocities feedback of the last 3 joints.

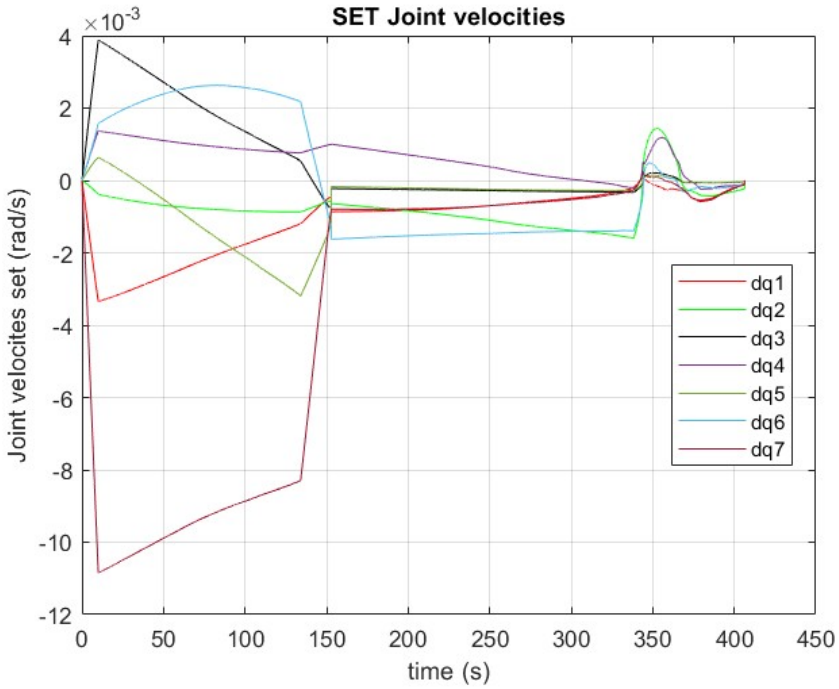

Figure 5.15: Set joint velocities.

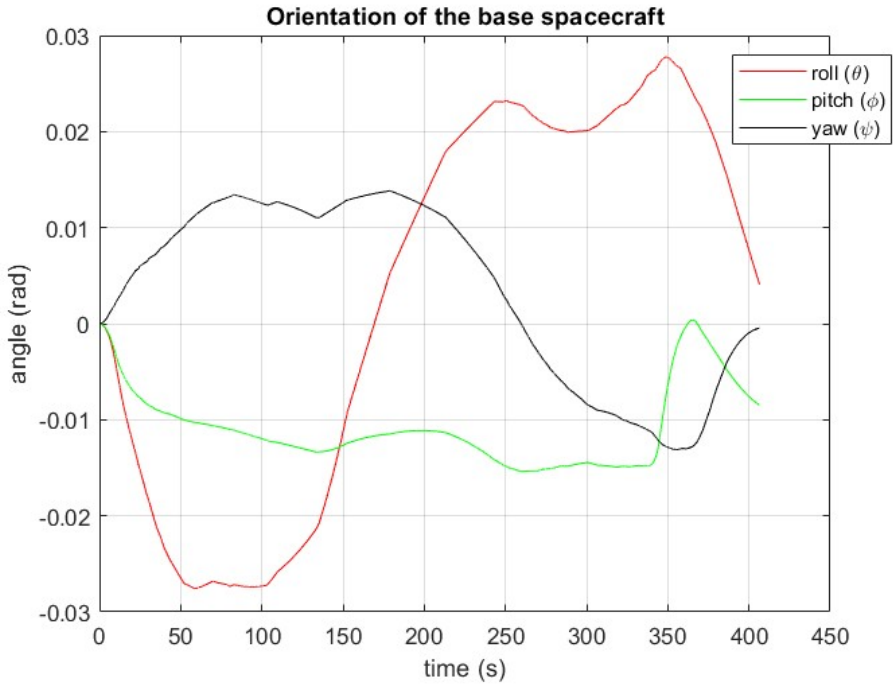

Figure 5.16: Orientation of the base spacecraft.

3) Chaser mass 1000 Kg.

For completeness, this is reported as well. Increasing the chaser mass there is a smaller deviation of the end effector trajectory from the linear path.

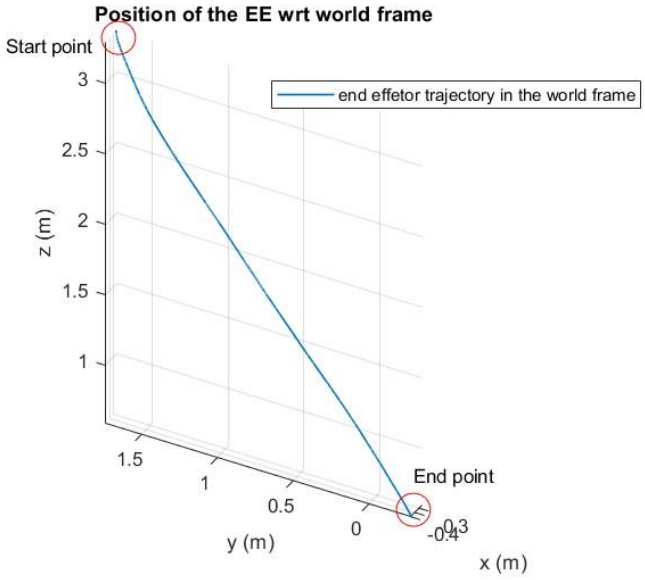

Figure 5.17: End effector trajectory.

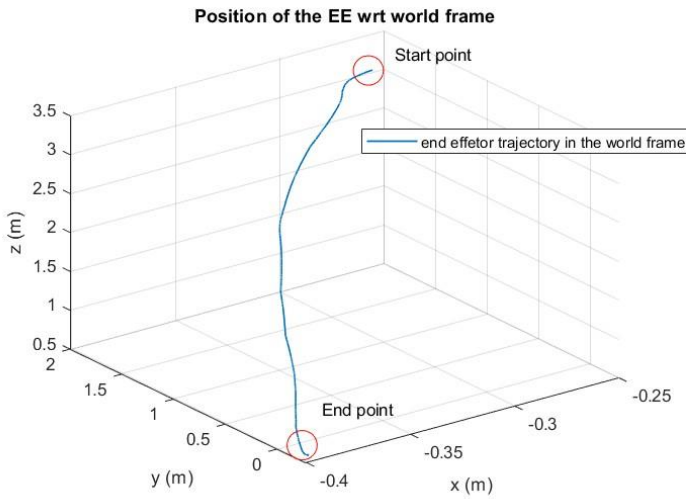

Figure 5.18: Zoom end effetor trajectory.

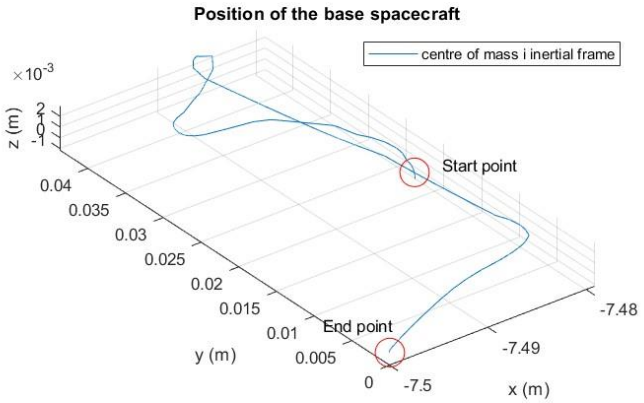

Figure 5.19: COM position.

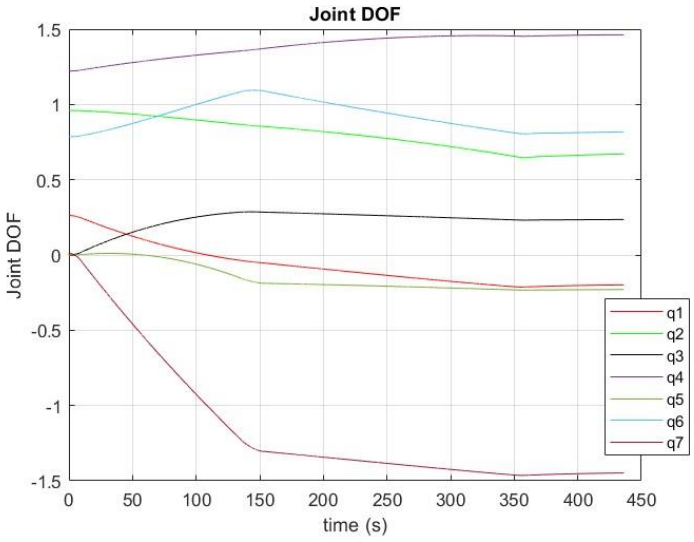

Figure 5.20: Joint DOF.

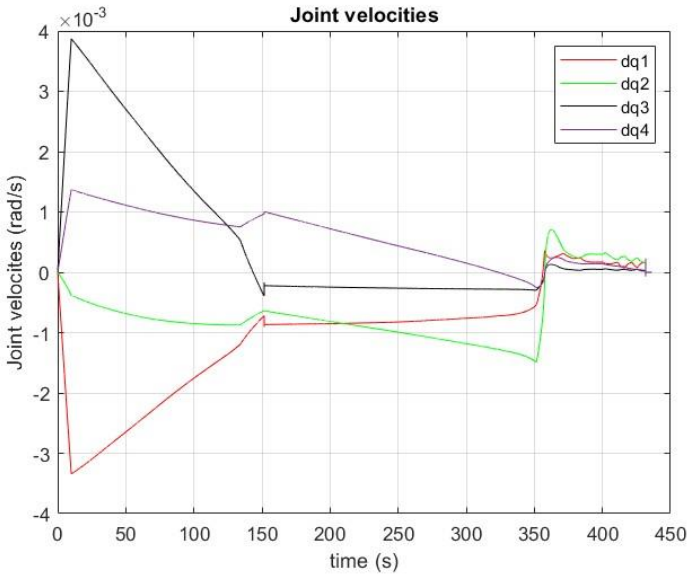

Figure 5.21: Joint velocities of the first 4 joints.

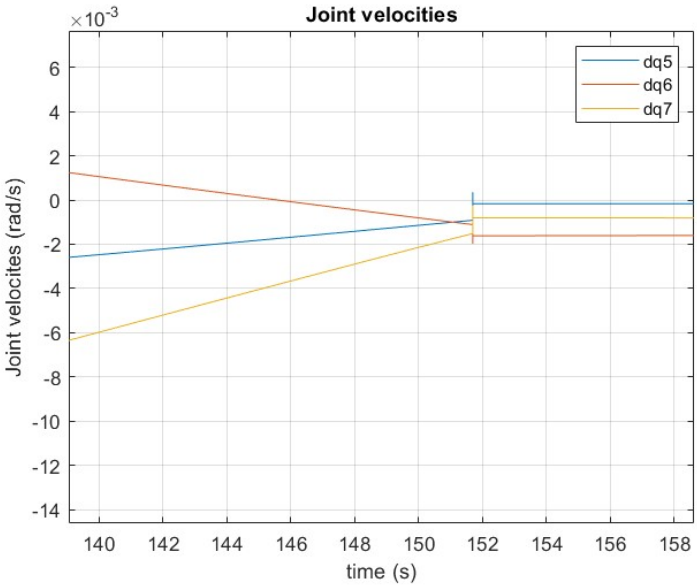

Figure 5.22: Joint velocities of last 3 joints.

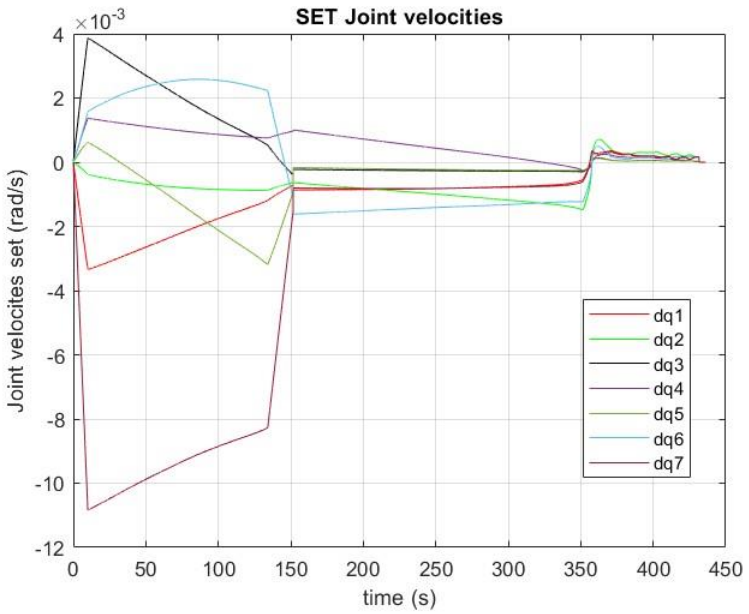

Figure 5.23: Joint velocities set.

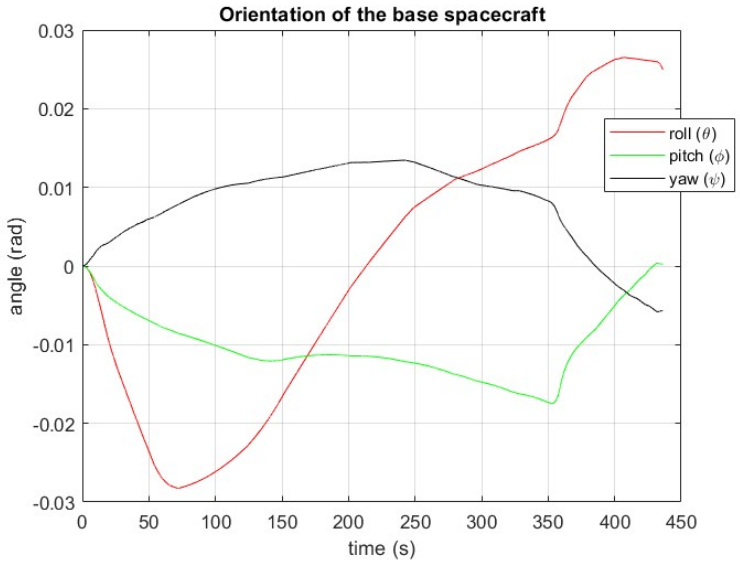

Figure 5.24: Orientation of base spacecraft.

4) Comparison of different mass.

As can be noted from the comparison reported in the following figures, increasing the spacecraft mass allow us to obtain a straight trajectory with a smaller deviation from the direct path and therefore a lower error in computing the trajectory.

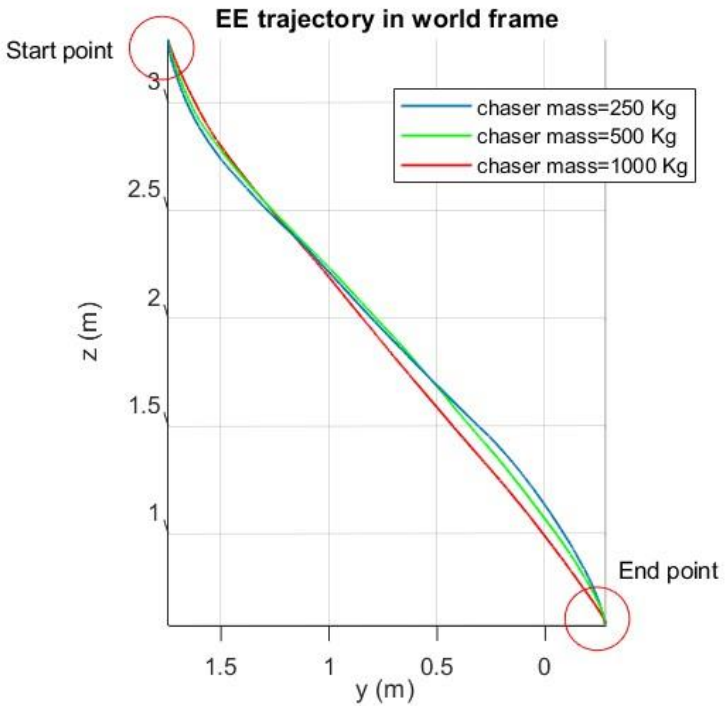

Figure 5.25: End effector trajectory with different mass.

Another interesting result can be obtained comparing the position of chaser COM in the inertial space, from the figure 5.26, can be seen that the 250 Kg case is highly disturbed with respect to the others, therefore increasing the mass the COM position is less disturbed and the spacecraft is stabilized around a certain equilibrium point with a higher accuracy. A better description of the difference can be obtained observing the result reported in figure 5.27 where the normalized position of the spacecraft in the inertial space is represented, as can be seen increasing the mass the error decrease.

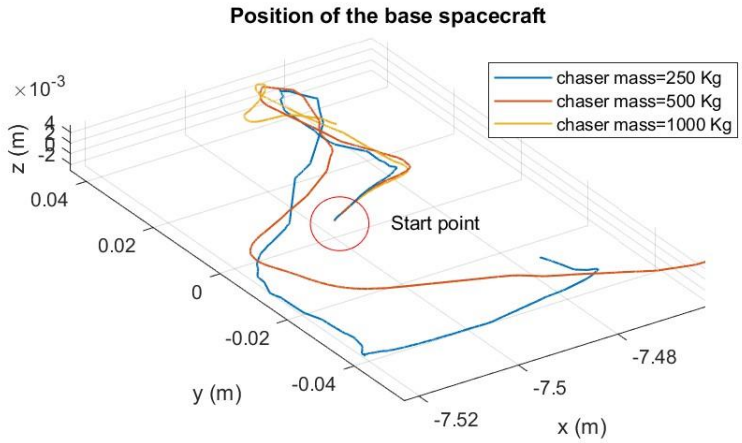

Figure 5.26: COM position.

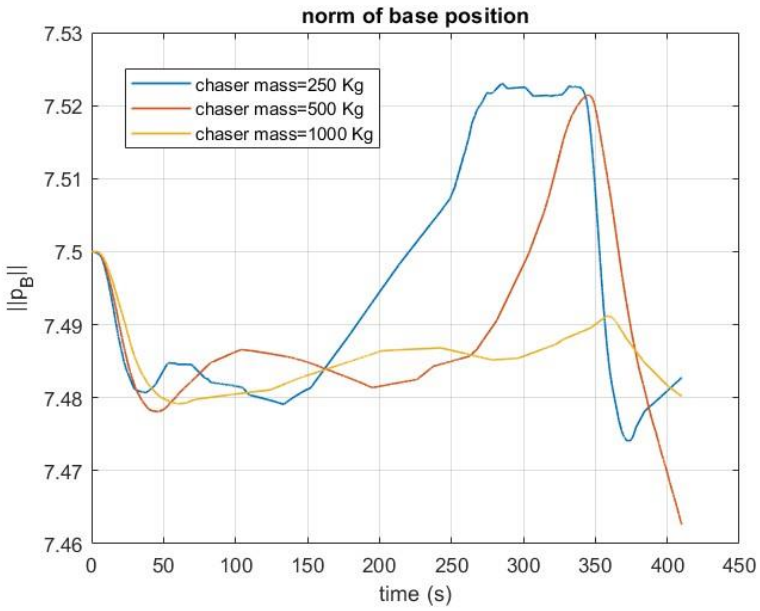

Figure 5.27: Norm of the spacecraft position.

5.2 Translation-Flying case.

In this flight mode the control action is performed on the spacecraft using reaction wheels for attitude and on-off thrusters for position control. The usage of momentum exchange devices, is a choice suitable for fine pointing operations, allowing a decrease in the attitude error, so it presents good performances for capturing a target with fixed base algorithms. In the following sections are reported the different case studied varying the mass.

1) Chaser mass 500 kg.

Here we can observe that the end effector trajectory in the inertial frame is a path described by a certain curvature, which with respect to the free-flying case is much closer to the theoretical straight line. This result can prove that the attitude control executed with reaction wheels is more accurate than the one executed with thrusters.

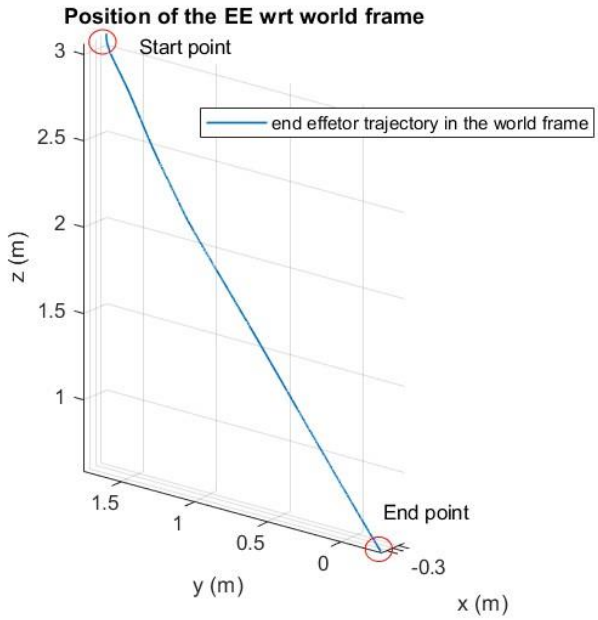

Figure 5.28: End effector trajectory.

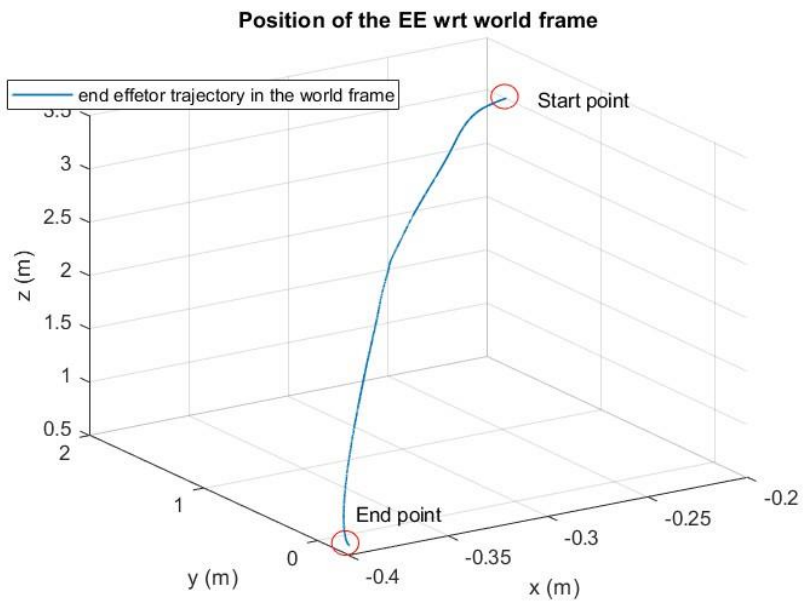

Figure 5.29: Zoom of the end effector trajectory.

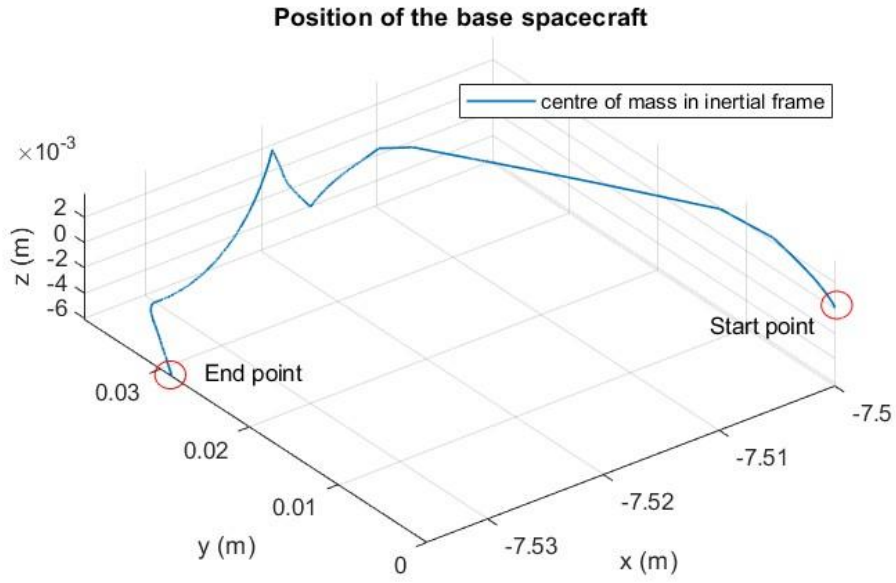

Figure 5.30: COM position.

The above observation can be further validated through the COM position with respect to the inertial frame, as can be seen the chaser spacecraft during the motion is highly stable.

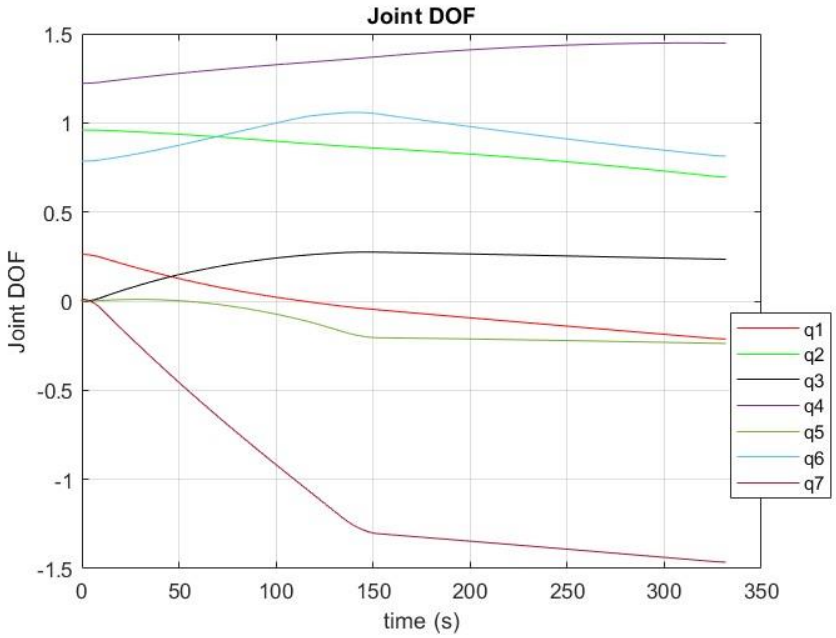

Figure 5.31: Joint DOF.

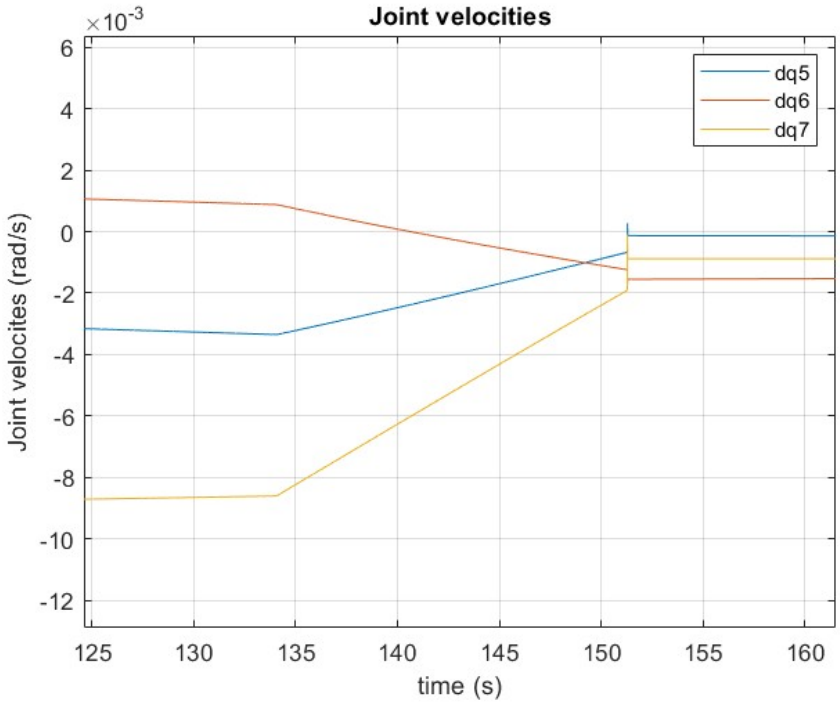

Figure 5.32: Joint velocities of the last 3 joints.

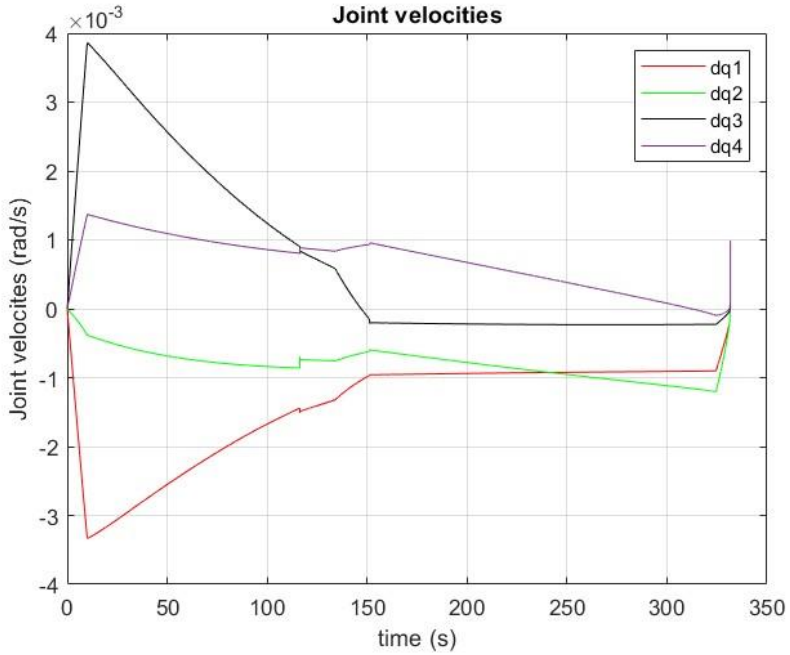

Figure 5.33: Joint velocities of the first 4 joints.

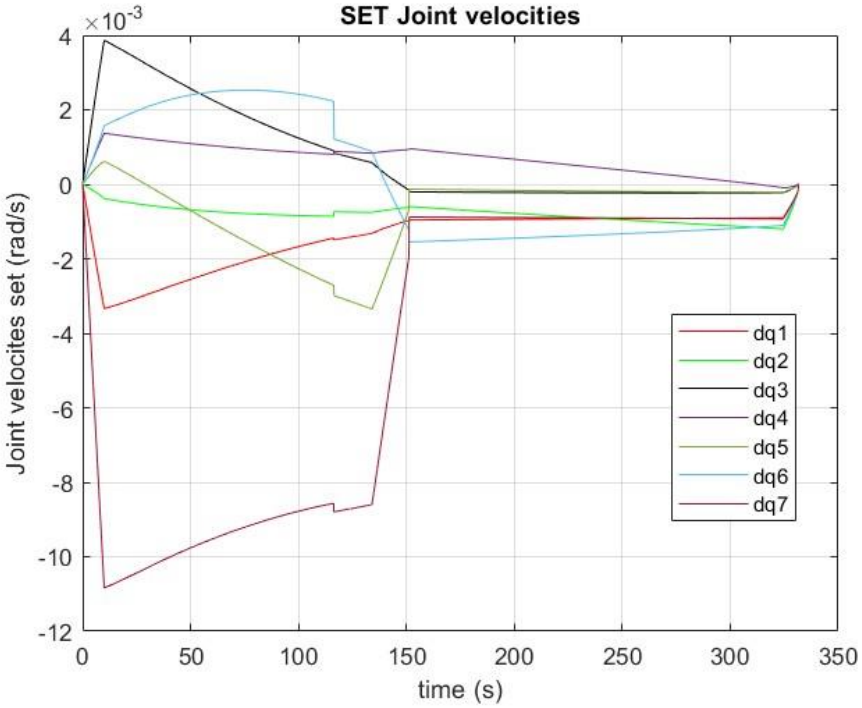

Figure 5.34: Joint velocities set.

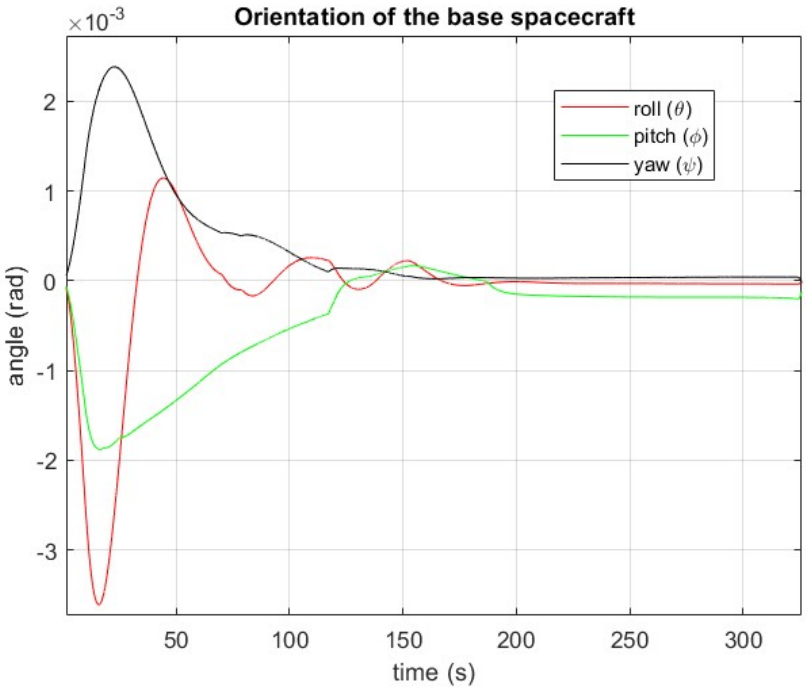

Figure 5.35: Orientation the spacecraft.

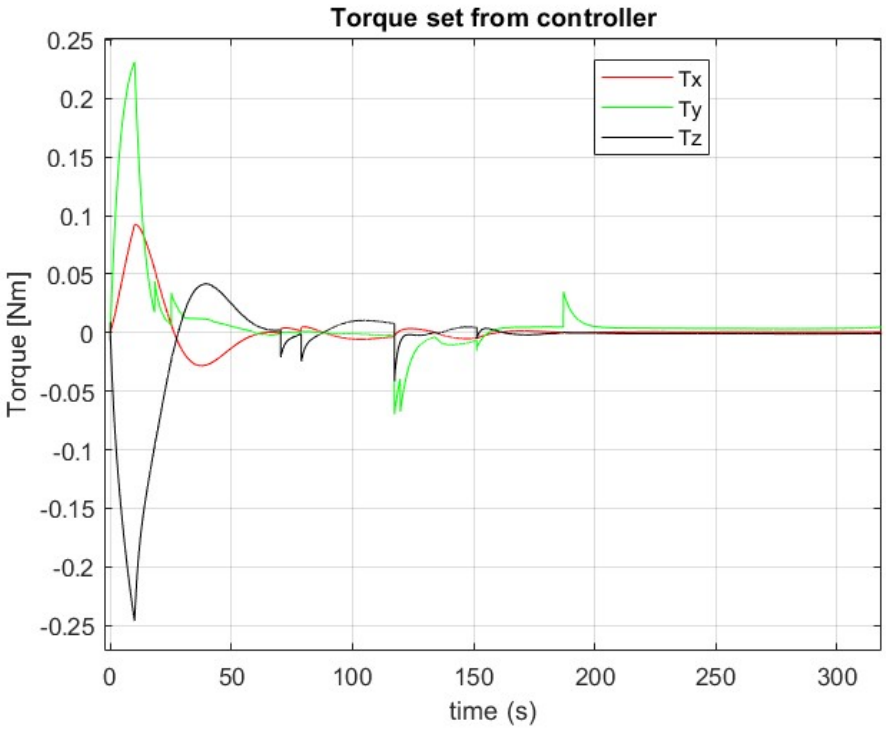

Figure 5.36: Torque set from controller.

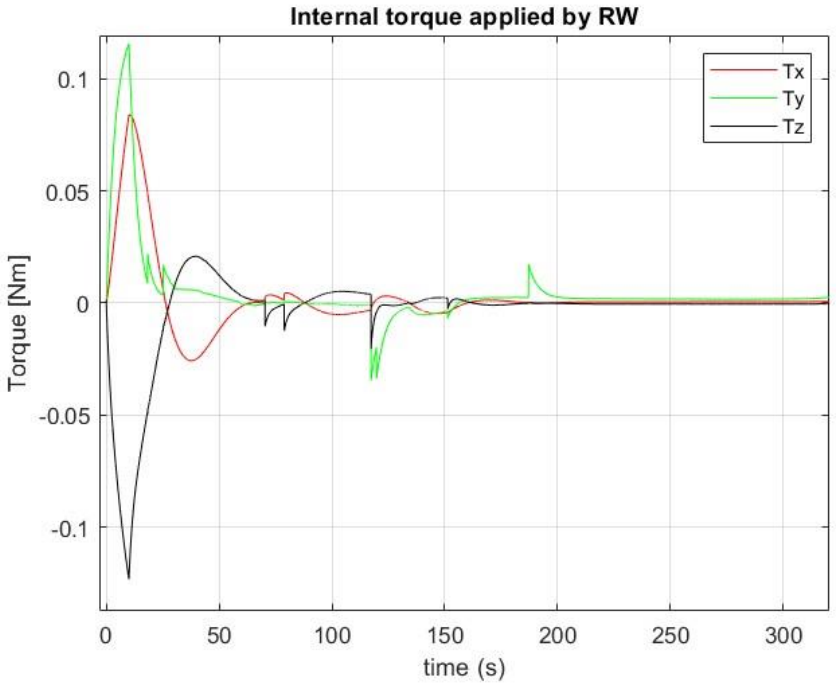

Figure 5.37: Torque applied from the reaction wheel.

2) Chaser mass of 250 Kg

Decreasing the mass of the spacecraft we can observe a reduction in term of accuracy of the end effector along the trajectory, though it still demonstrates a good performance.

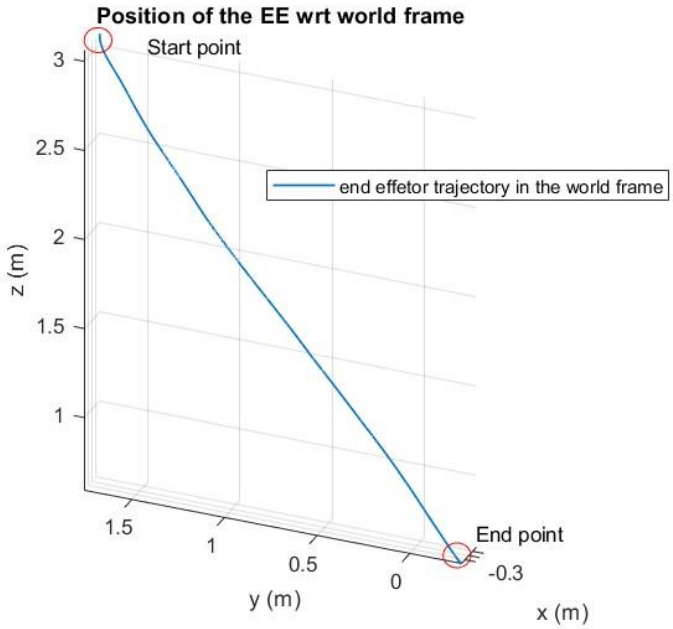

Figure 5.38: End effetor position.

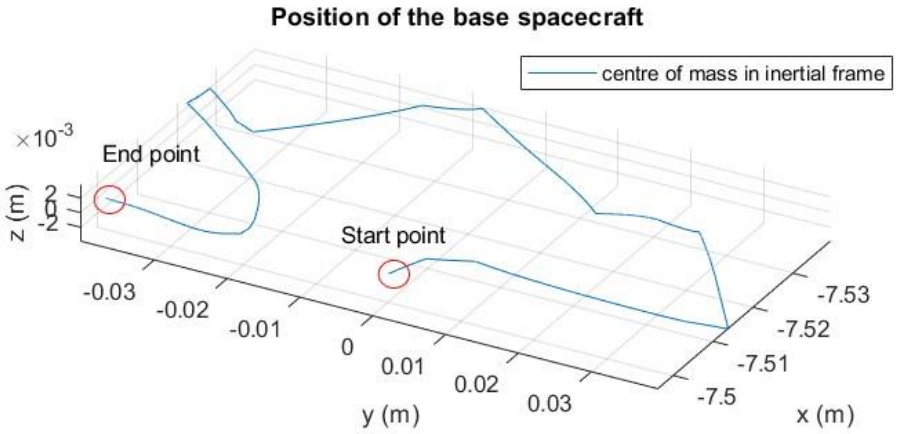

Figure 5.39: COM position.

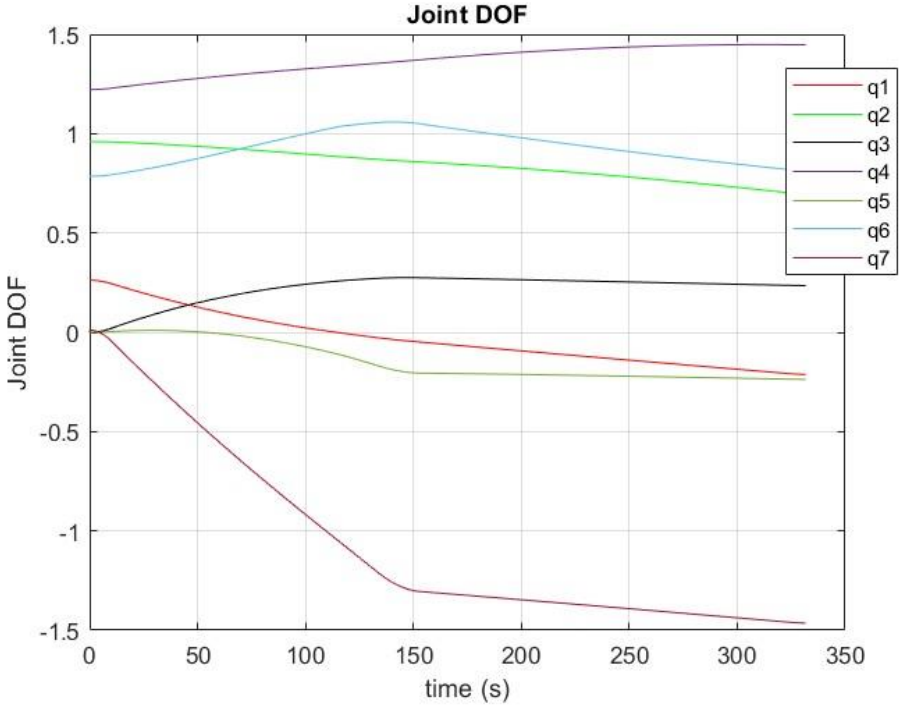

Figure 5.40: Joint DOF.

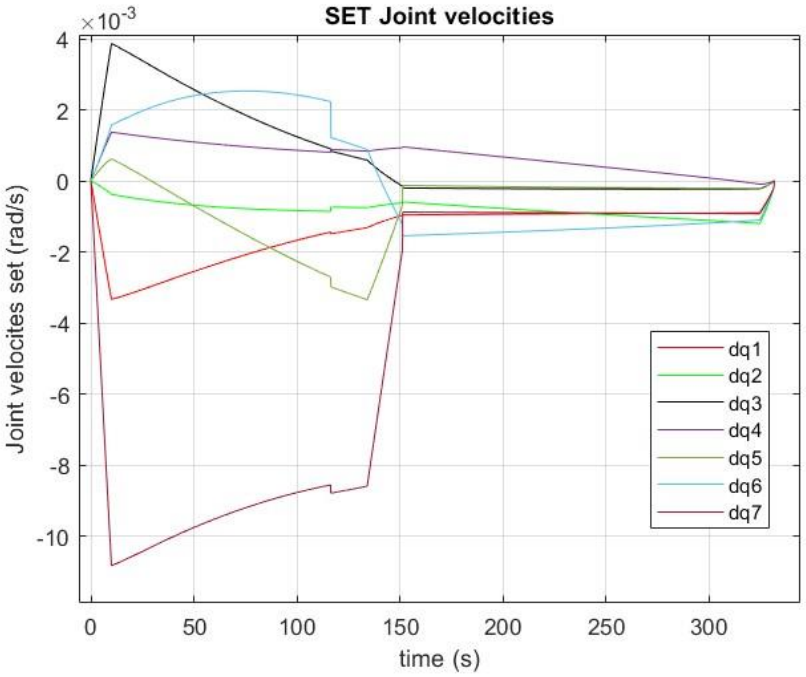

Figure 5.41: Joint velcities set.

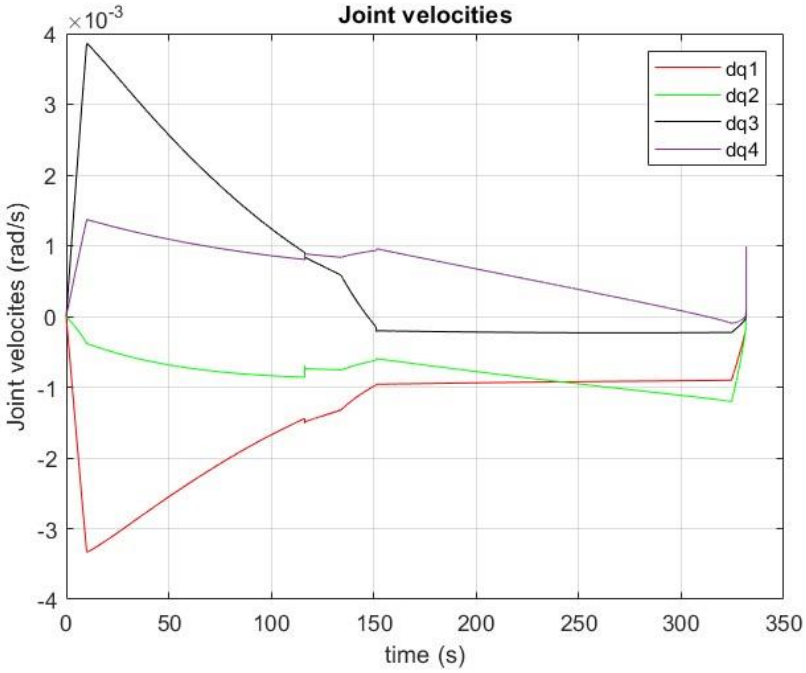

Figure 5.42: joint velocities feedback.

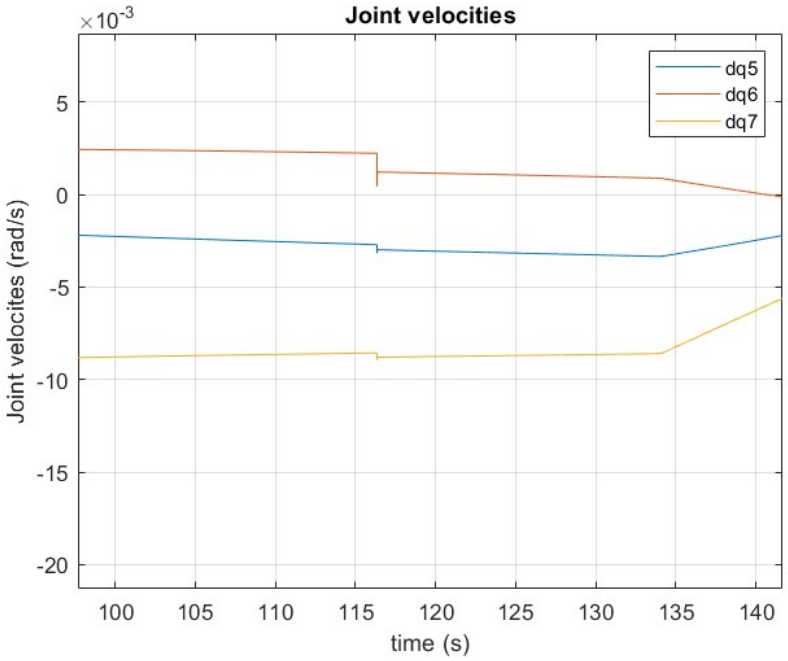

Figure 5.43: joint velocities feedback.

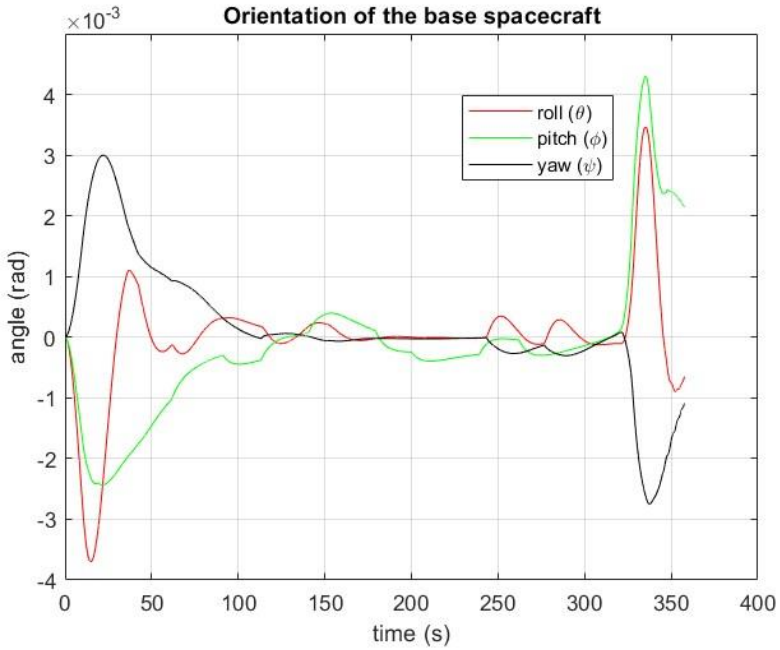

Figure 5.44: Orientation of the spacecraft.

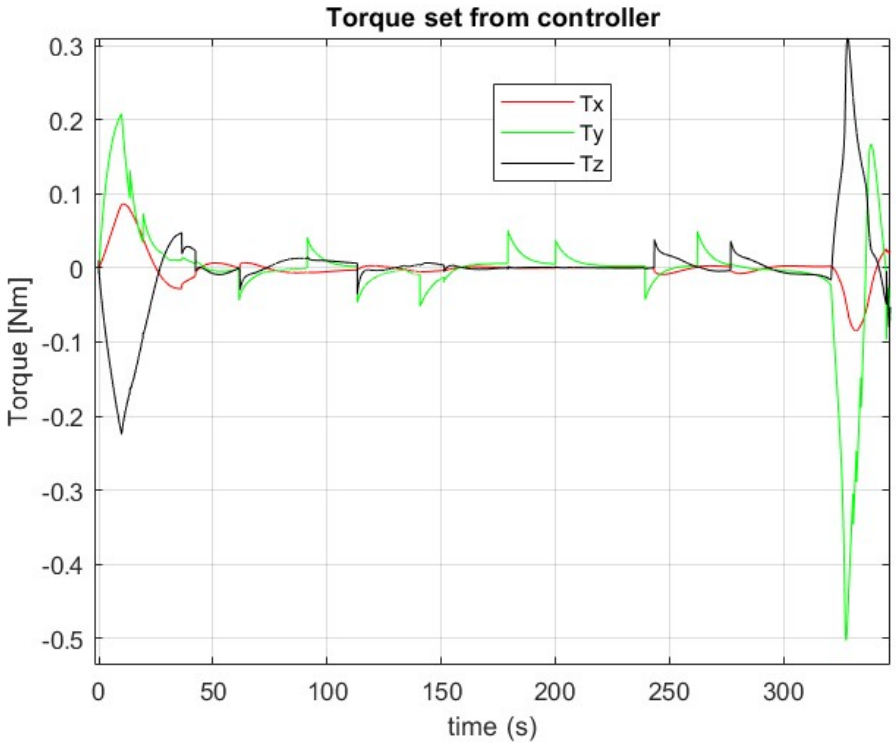

Figure 5.45: Torque set from controller.

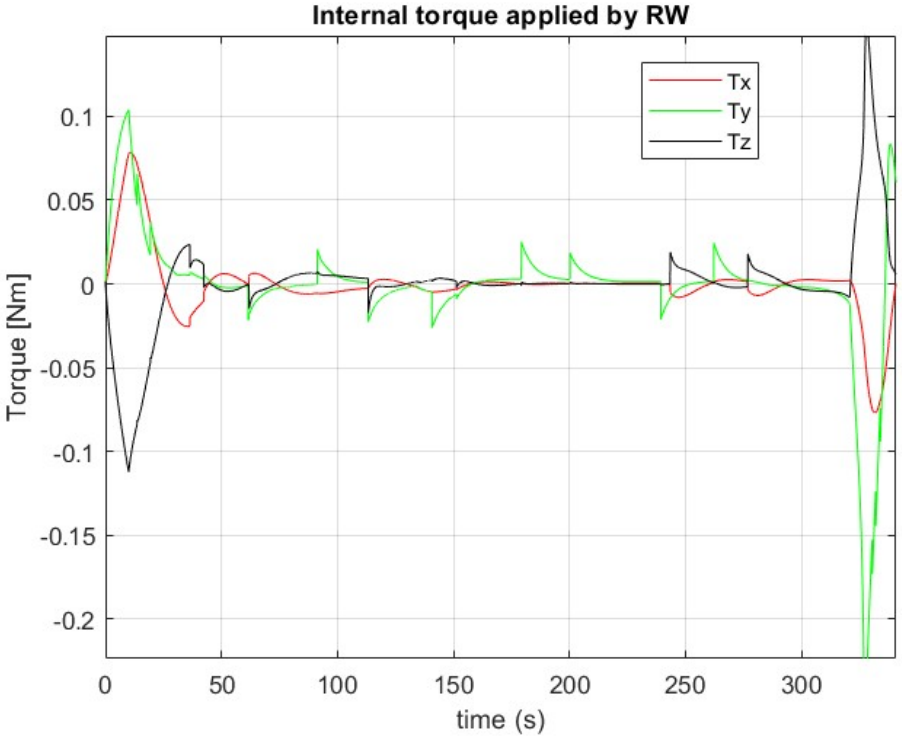

Figure 5.46: Torque applied by the reaction wheels.

3) Chaser mass 1000 Kg.

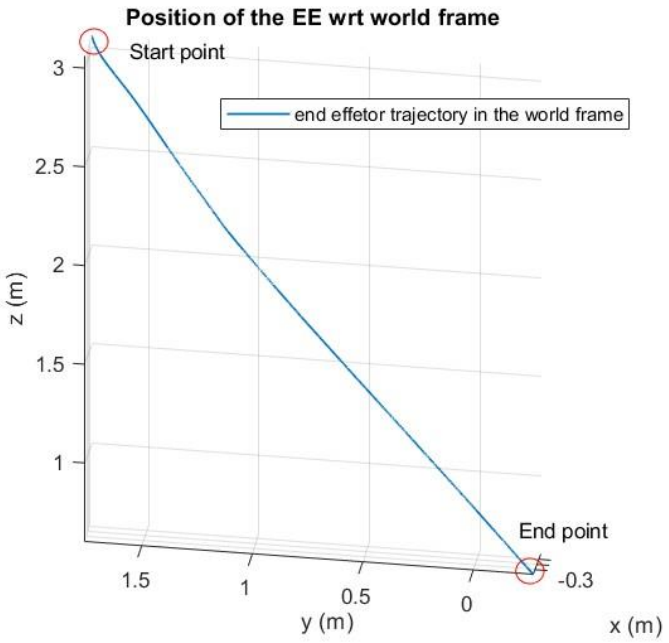

Figure 5.47: End effector trajectory

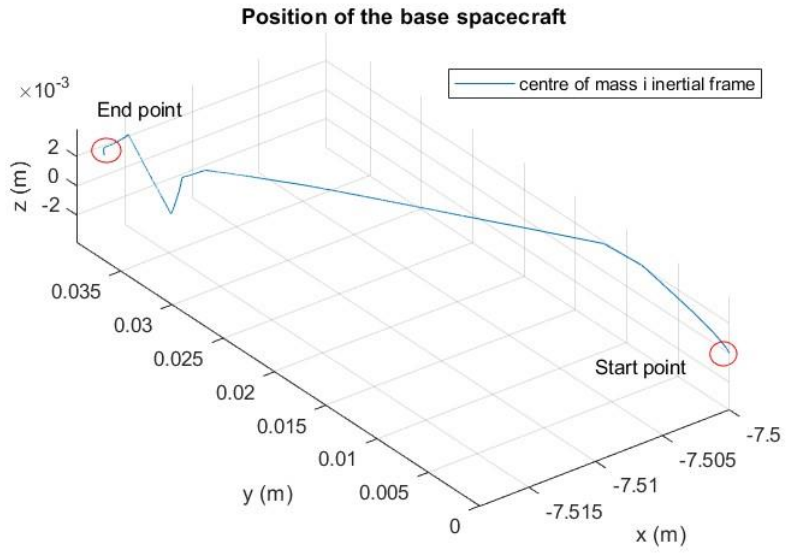

Figure 5.48: COM position.

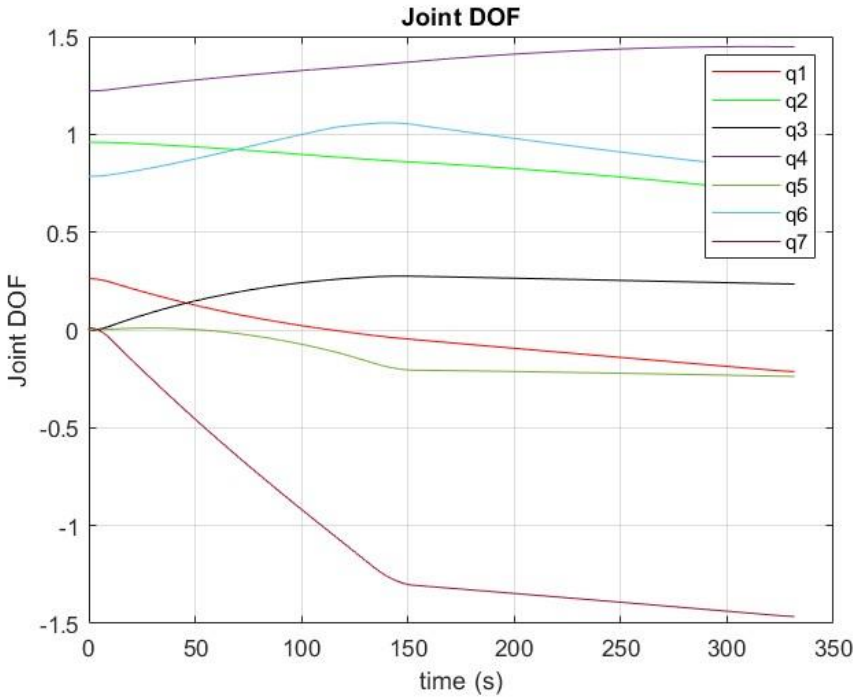

Figure 5.49: Joint DOF.

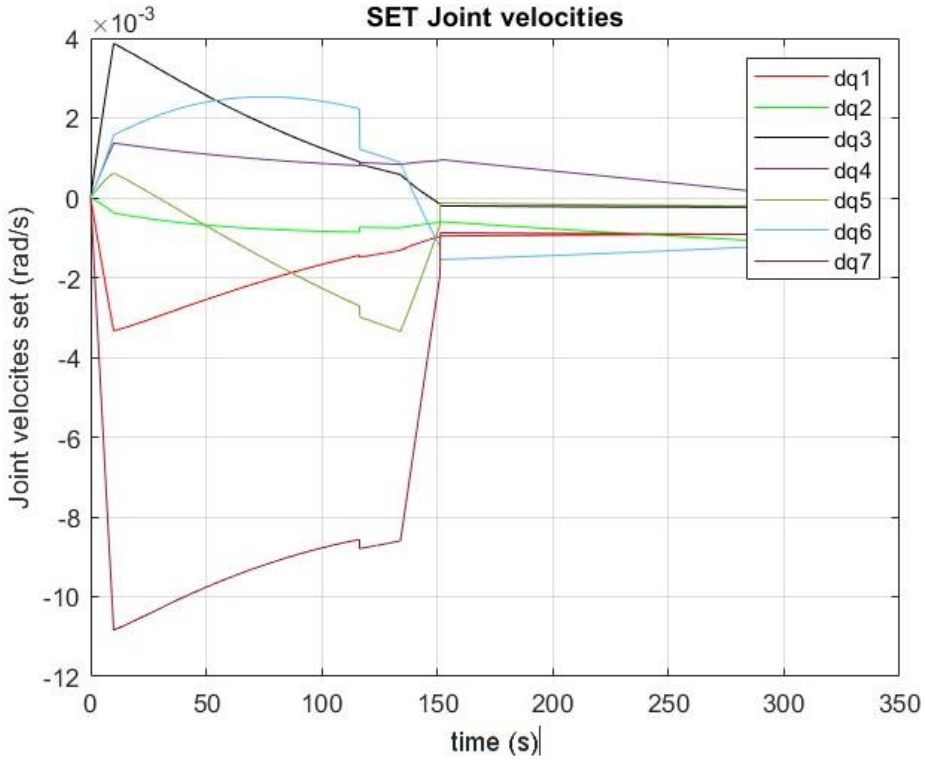

Figure 5.50: joint velocities set.

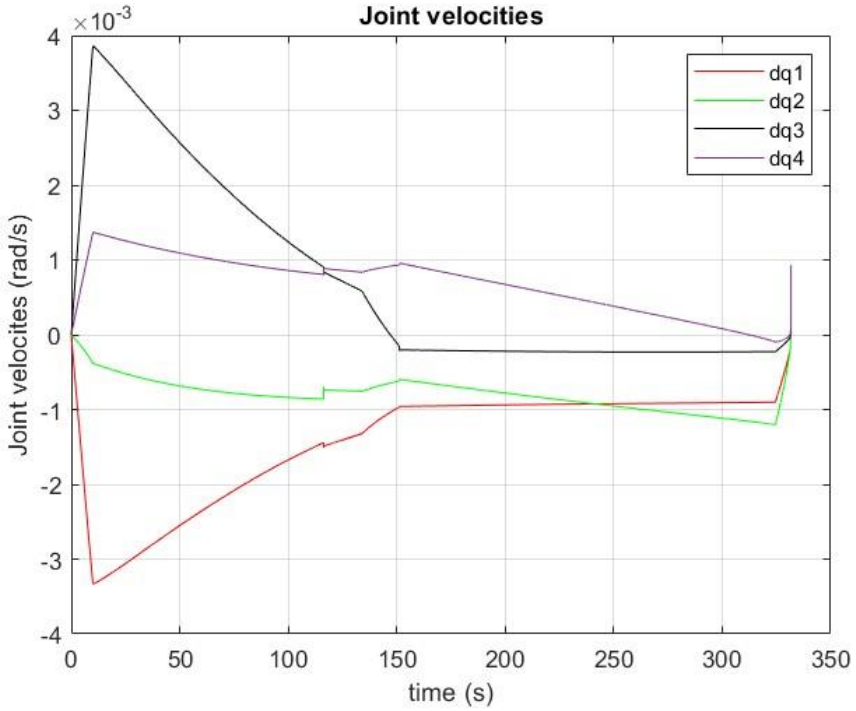

Figure 5.51: Joint velocities feedbeck.

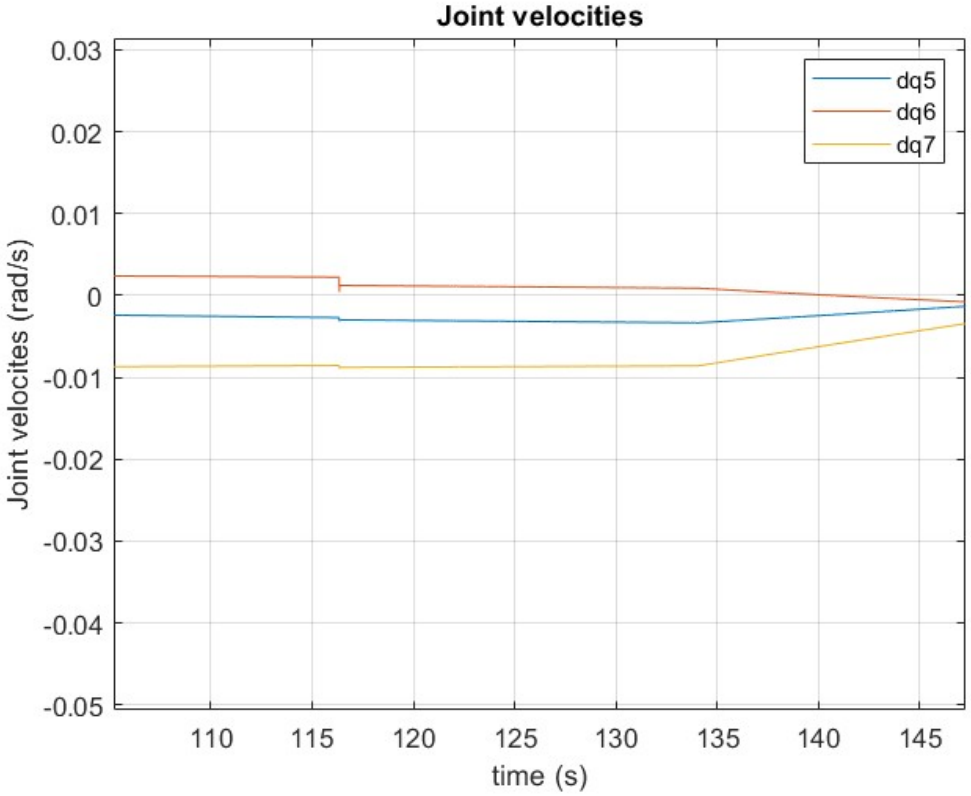

Figure 5.52: Joint velocities feedbeck.

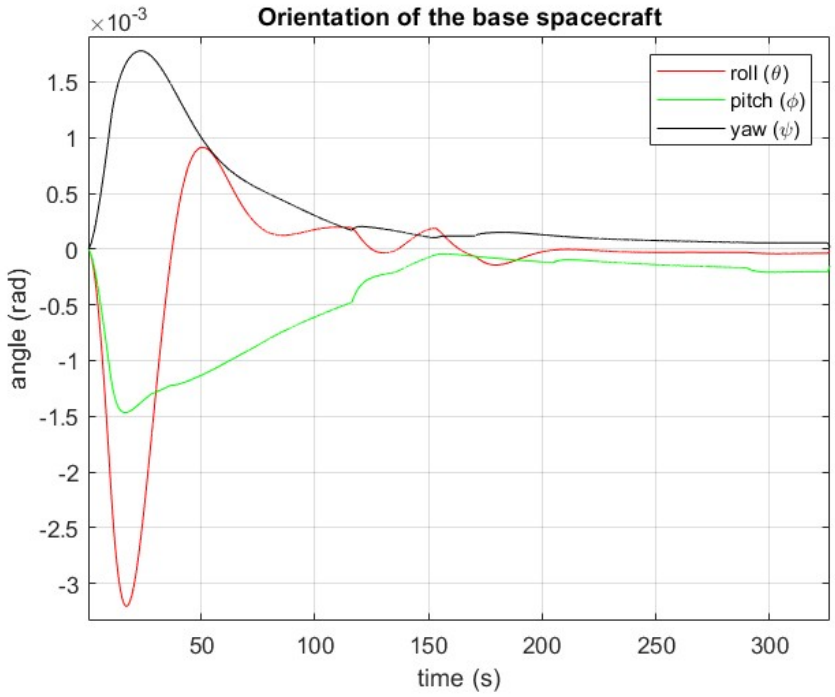

Figure 5.53: Orientation of the spacecraft.

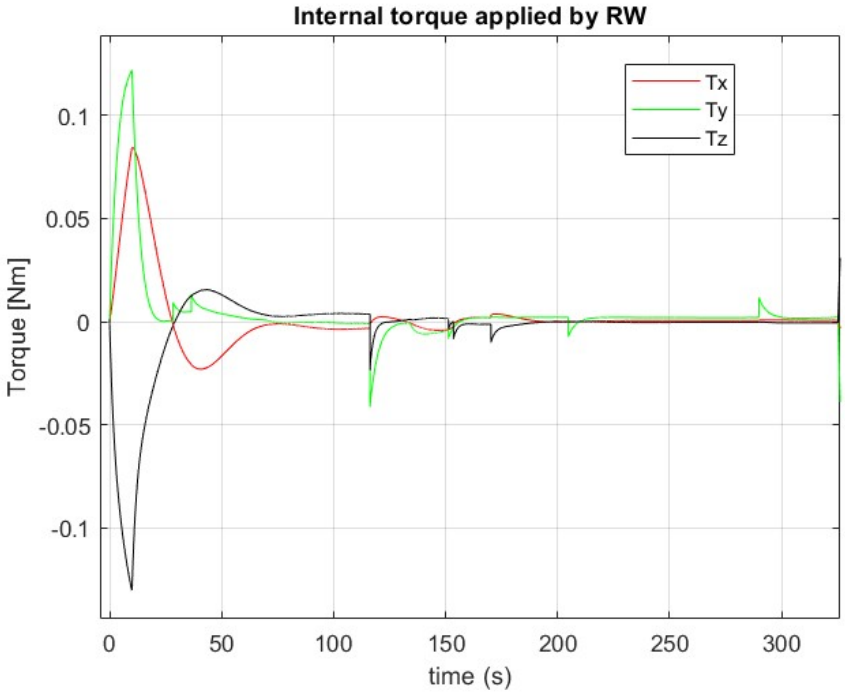

Figure 5.54: Internal torque applied by reacion wheels.

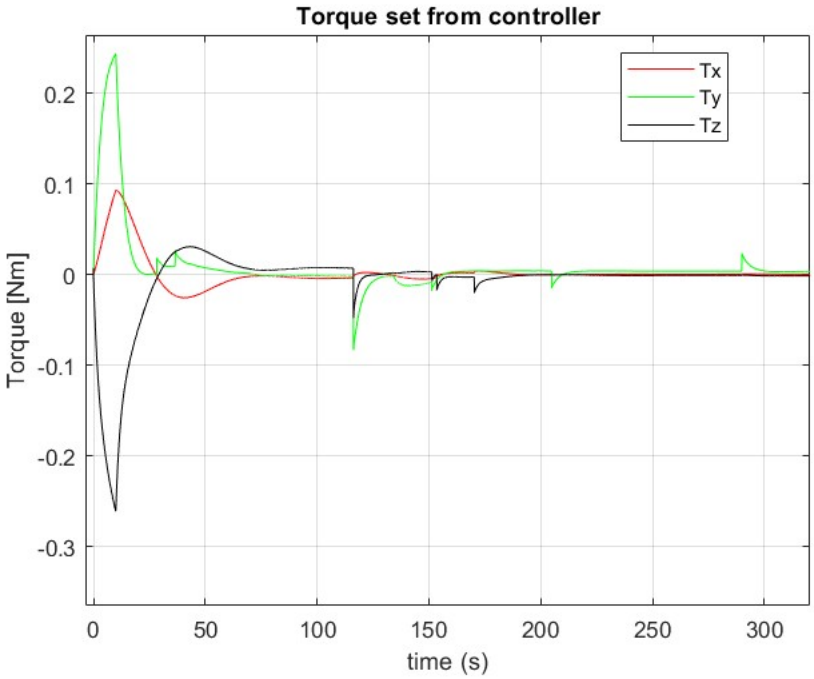

Figure 5.55: Torque set from controller.

5.3 Free-Floating case

As outlined before, in this case no control actions are performed on the spacecraft, therefore it is free to move in the inertial space as a consequence of the robotic arm operation.

1) Chaser mass of 500 Kg.

Choosing a mass equal to 500 Kg we can observe the main results obtained in the following figures. The end effector trajectory in the inertial space, instead of been straight, presents a curvilinear path as a consequence of the compensative effect of the kinematic algorithm, indeed, while the joint motors exert a torque to move the end-effector over the serial links chain, a rection force is applied from the robotic arm to the base of the spacecraft causing its variation in terms of position and attitude. The main results are plotted in the following figures.

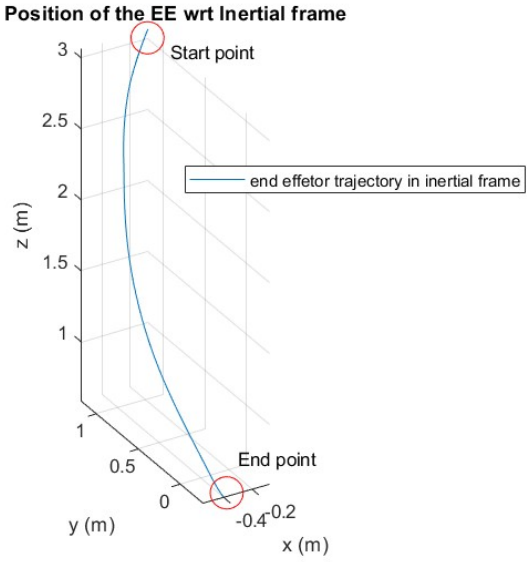

Figure 5.56: End effector trajectory.

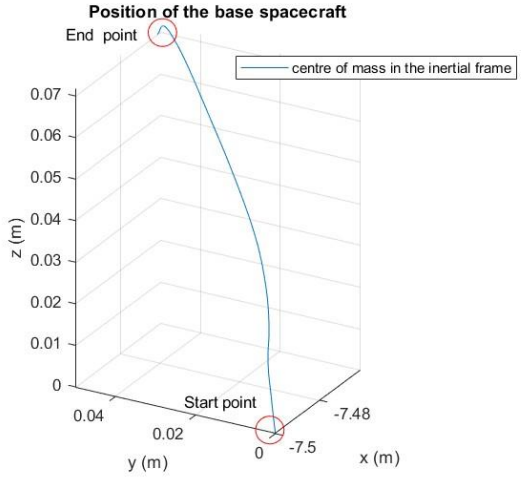

Figure 5.57: Zoom of end effector trajectory.

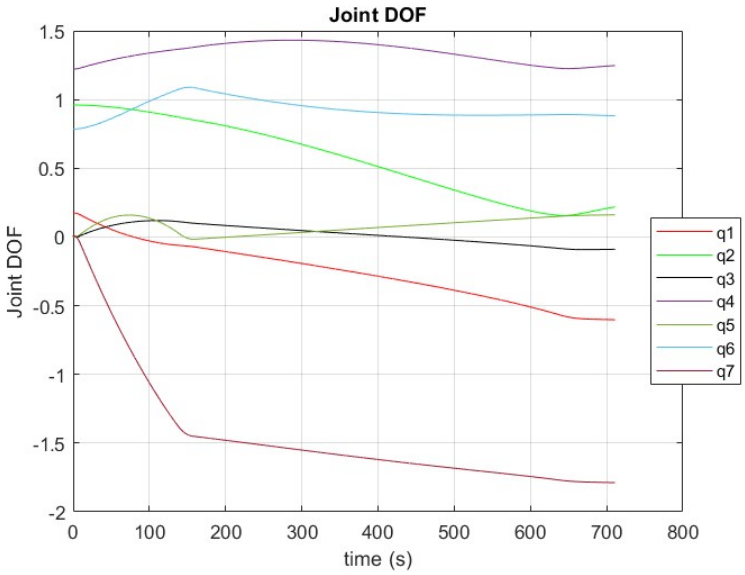

Figure 5.58: Joint DOF.

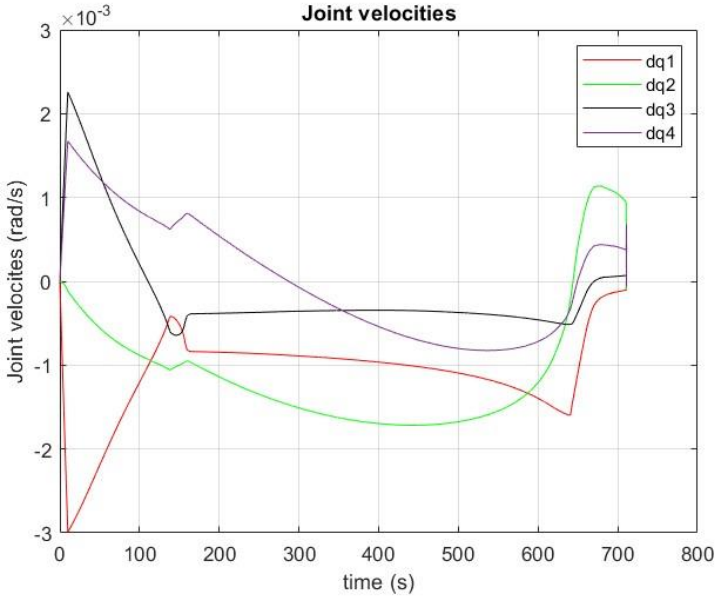

Figure 5.59: Joint velocities feedback.

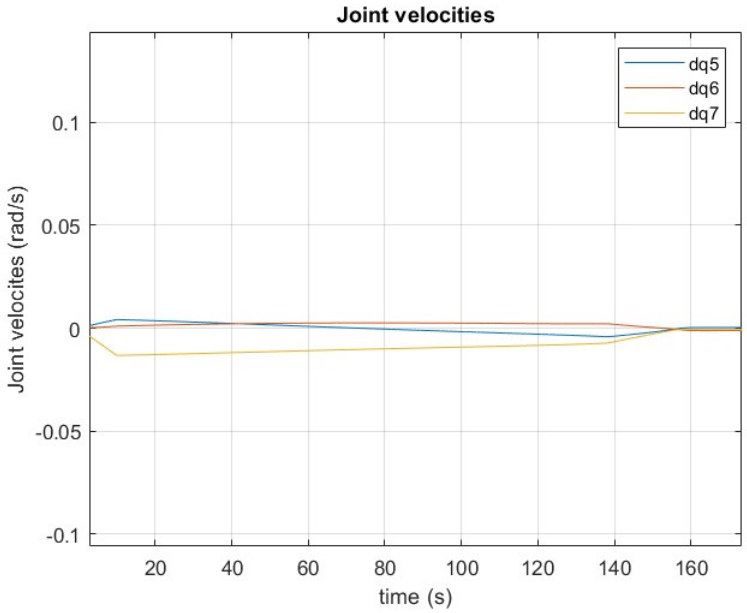

Figure 5.60: Joint velocities feedback.

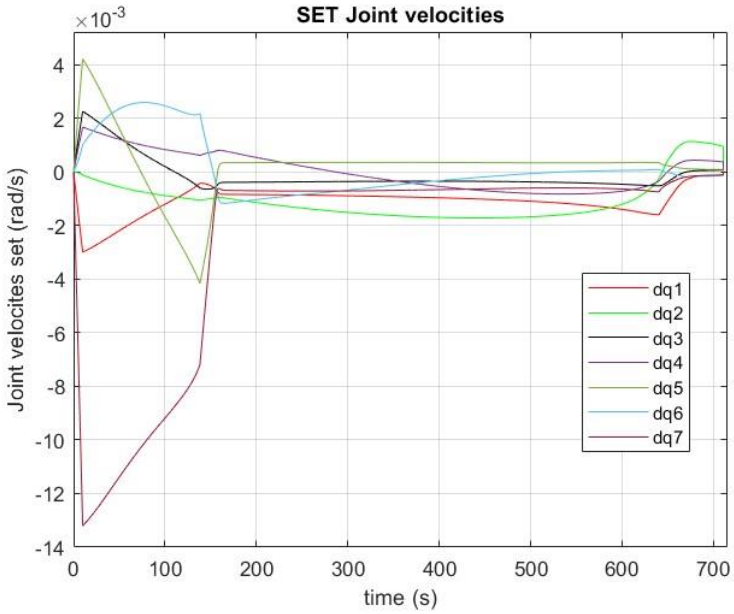

Figure 5.61: Joint velocities set.

As already said, in this flight mode there isn't the switch of Jacobians and therefore the generalized Jacobian compute all the path for the end effector.

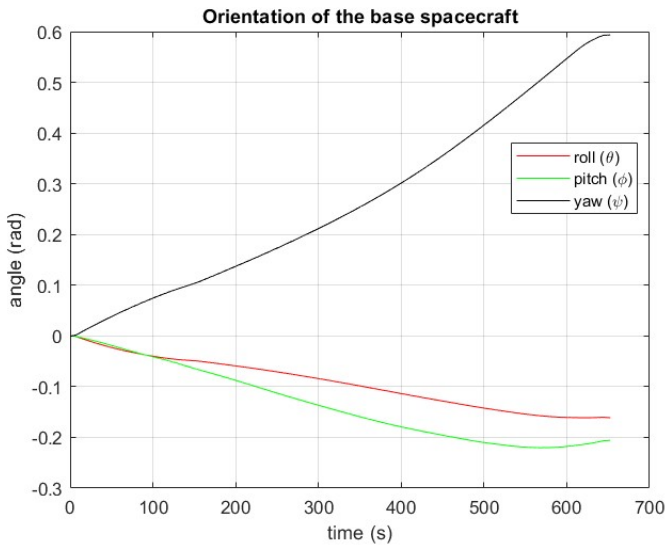

Figure 5.62: Orientation of the spacecraft.

2) Chaser mass 800 Kg.

As can be seen in the following figures, the end effector trajectory has a less curvature and he COM of the base is less perturbed with respect to the previous case.

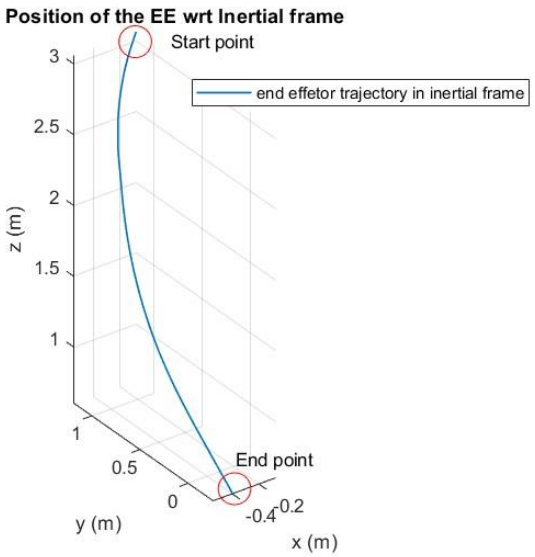

Figure 5.63: End effector trajectory.

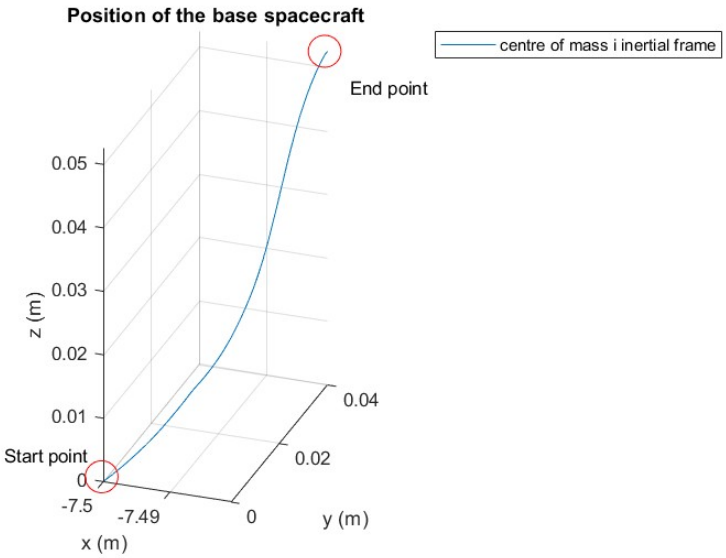

Figure 5.64: COM position.

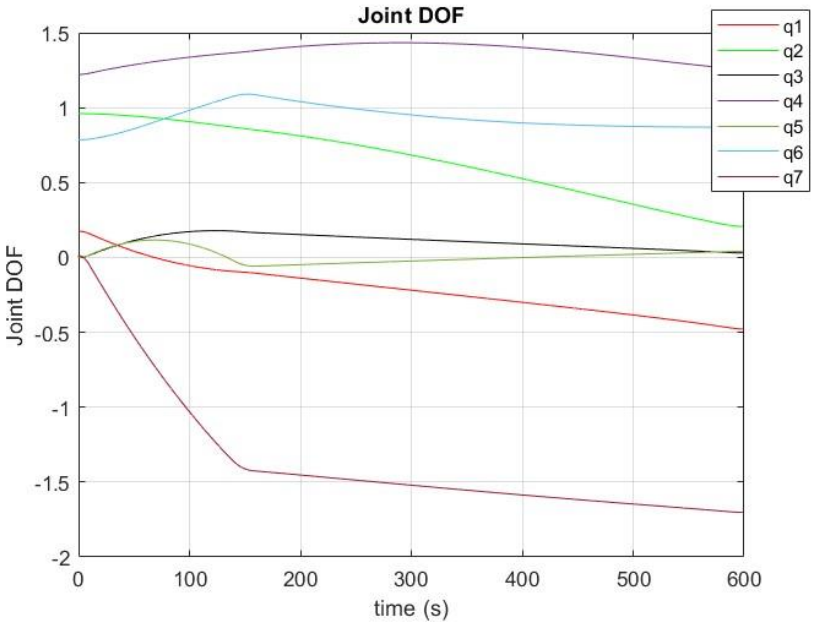

Figure 5.65: Joint DOF.

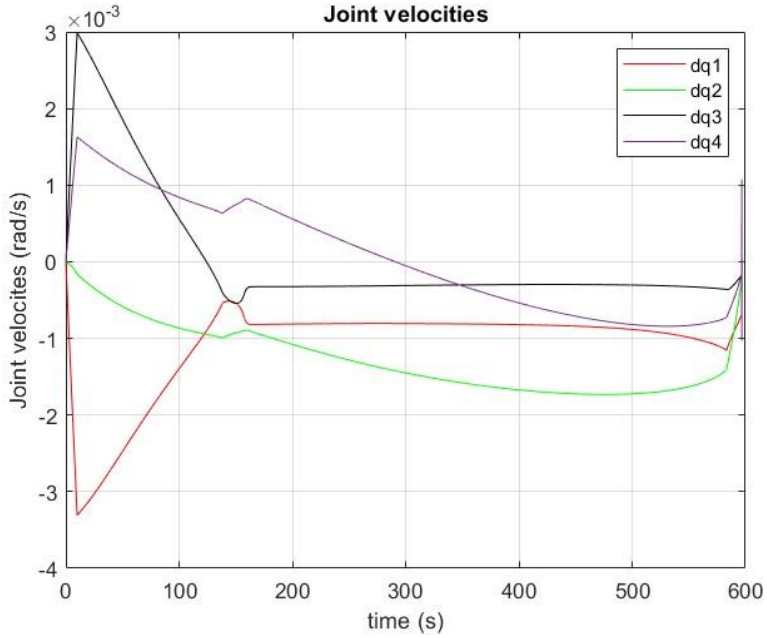

Figure 5.66: Joint velocities feedback.

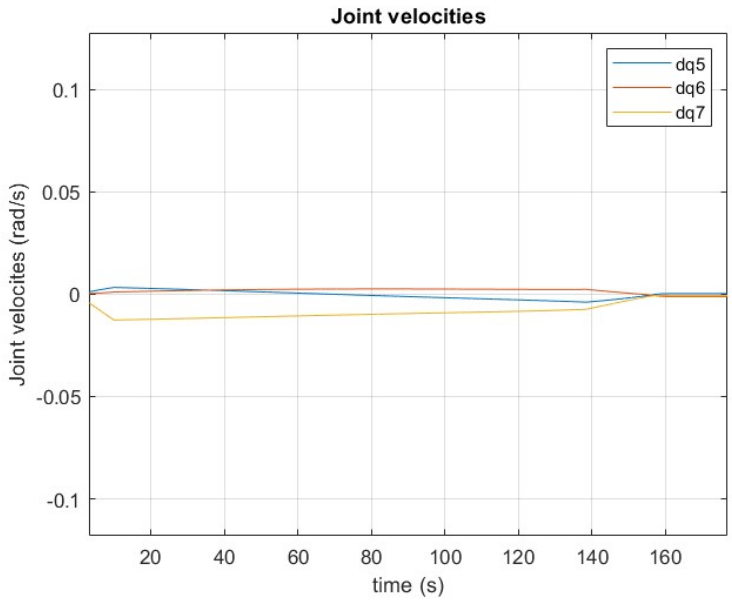

Figure 5.67: Joint velocities feedback.

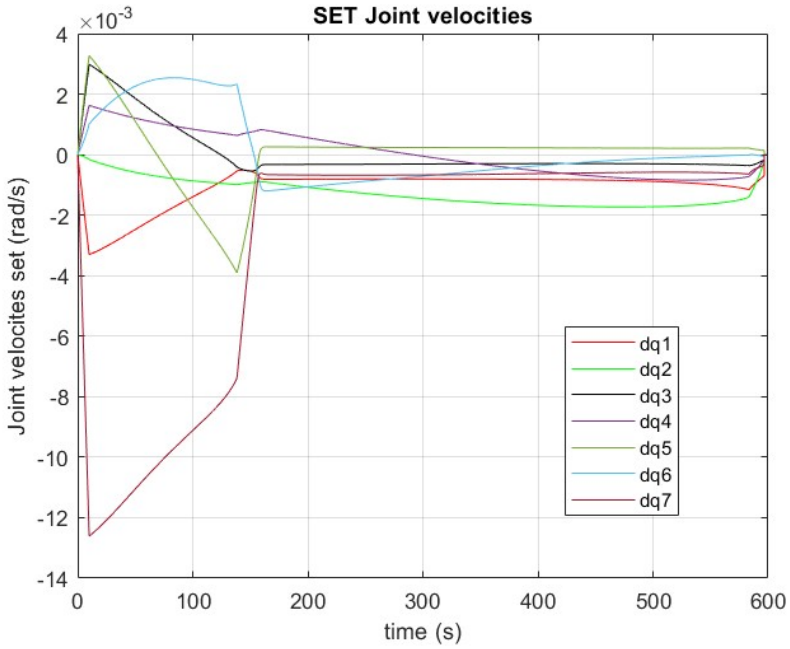

Figure 5.68: Joint velocities set.

3) Variation of end effector velocities and acceleration.

Another interesting result can be obtained in the free-floating case increasing the maximum velocity and the maximum acceleration that can be imposed to the end effector staring from the computation of the velocity profile.

Max EE velocity	$\mathbf{0.05}\ \boldsymbol{m/s}$
Max EE acceleration	$0.01\ m/s^2$
Chaser mass	*1000 Kg*

Table 5-2: Parameter Choice.

The results are then reported in the following figures. As can be seen, the grasping operation time is reduced around 140 seconds as an obvious consequence of the major velocity and acceleration. Moreover, can be observed that the minimum chaser mass needed to execute the operation without any compliance of interfering between base and manipulator increased with respect to the low-speed model.
So, the end effector deviation from the linear path increases with respect to the previous case considering the same mass for the chaser.

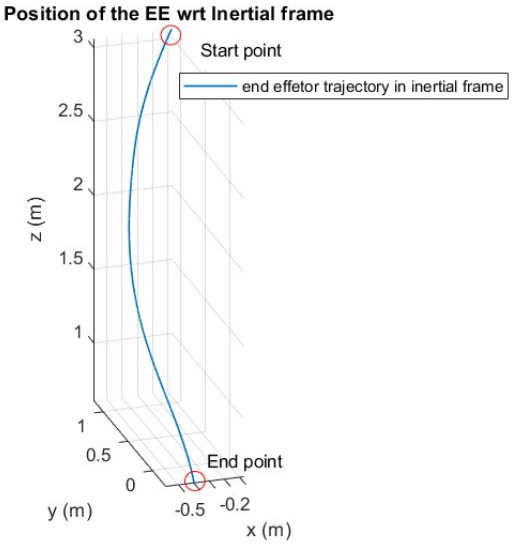

Figure 5.69: End effector trajectory.

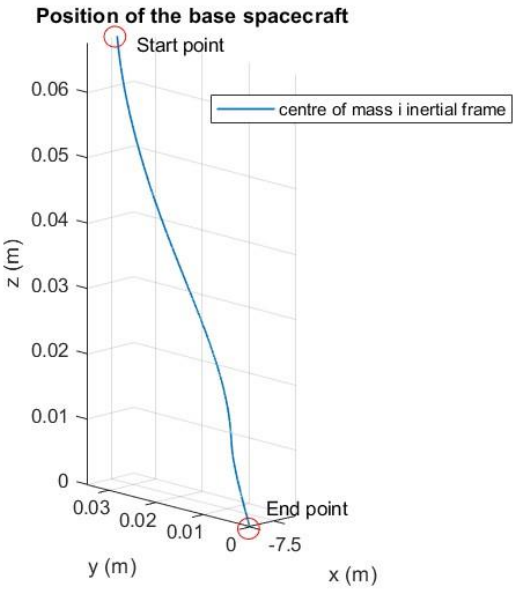

Figure 5.70: COM position.

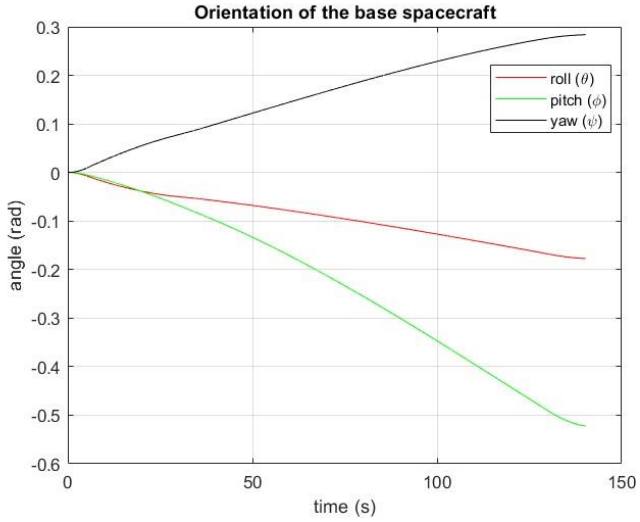

Figure 5.71: Orientation of the spacecraft.

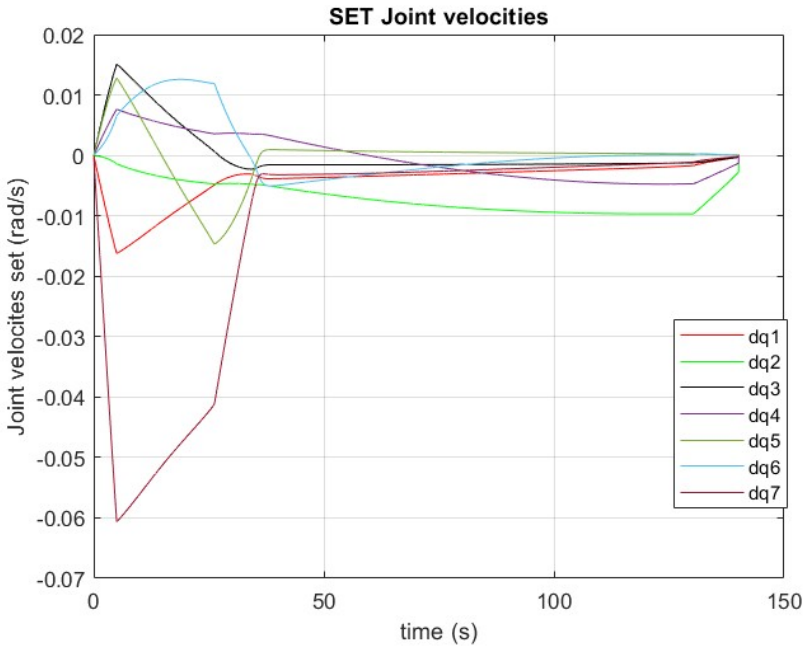

Figure 5.72: Joint velocities set.

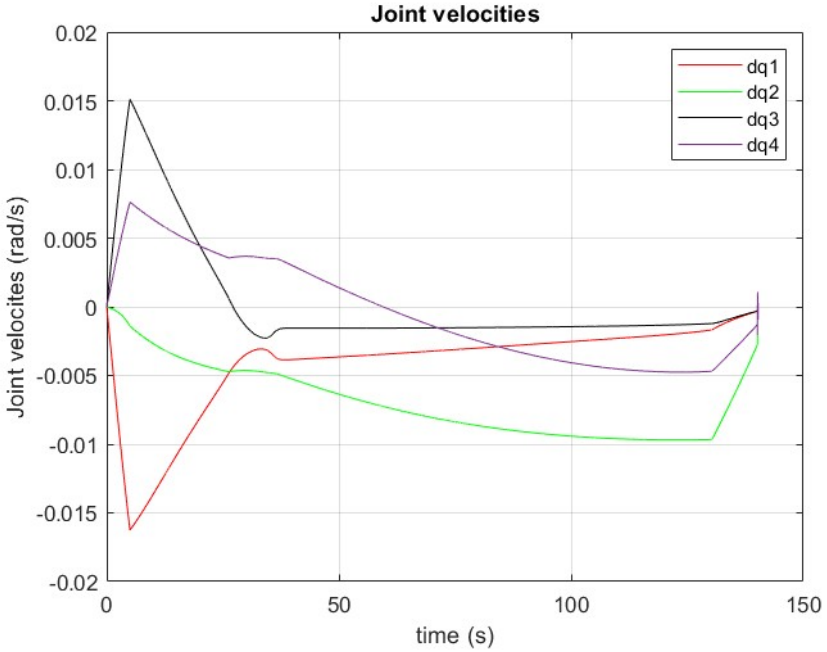

Figure 5.73: Joint velocities feedback.

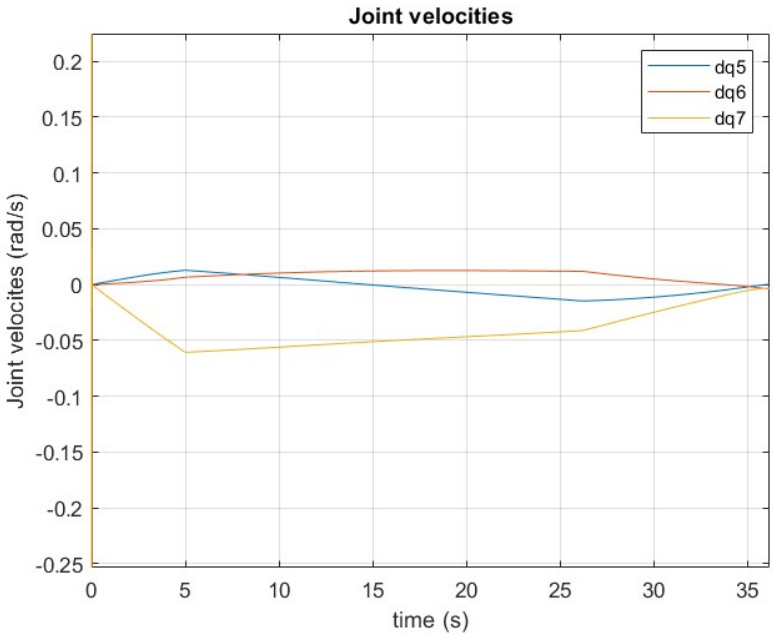

Figure 5.74: Joint velocities feedback.

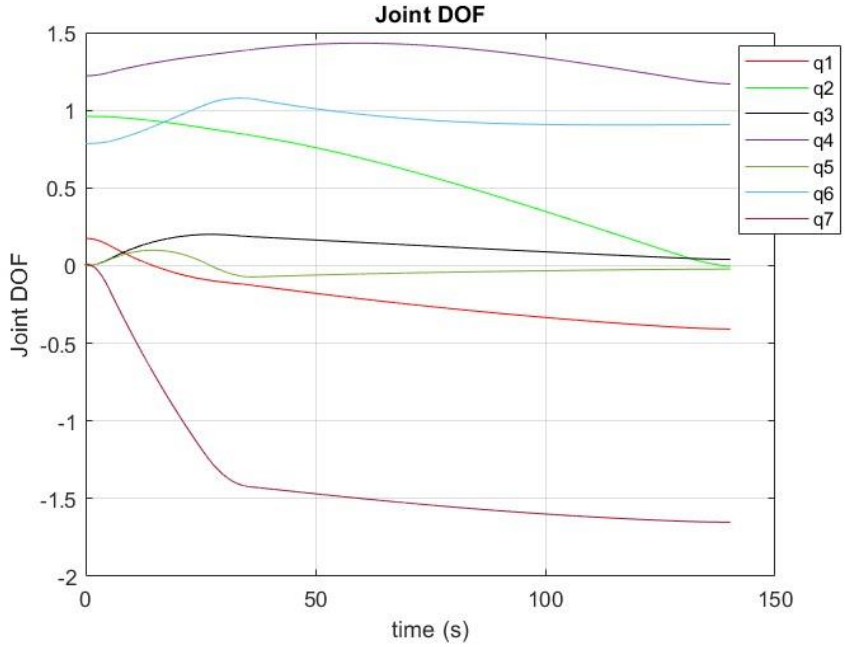

Figure 5.75: Joint DOF.

Comparing the case of free-floating with different end-effector velocity and a mass of 800 Kg of the chaser spacecraft we obtain the following results, where case 1 indicates the modified velocity in the free-floating case, and case 2 stands for the case reported in figure 5.69, as can be observed a higher velocity brings to a higher deviation and:

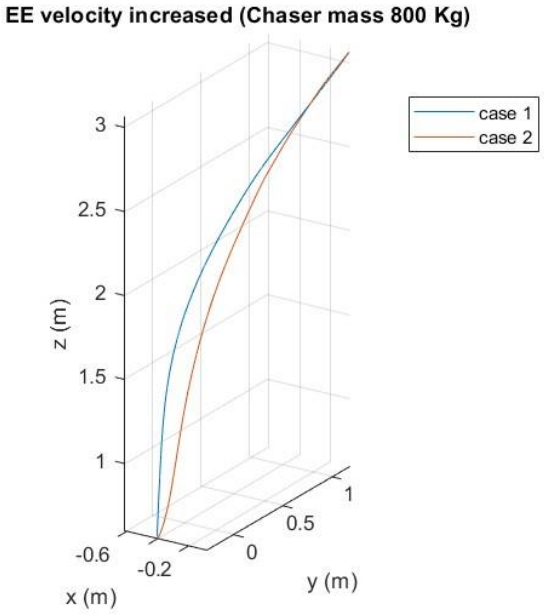

Figure 5.76: Compaing end-effector trajectory.

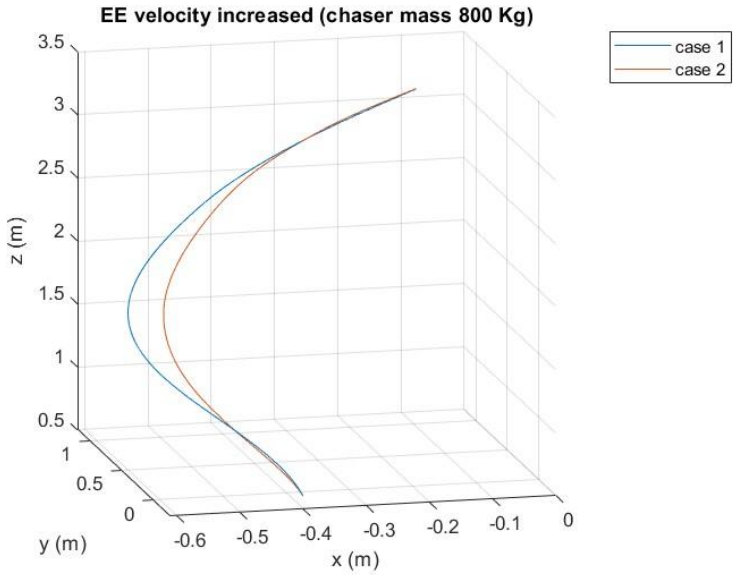

Figure 5.77: Zoom of figure 5.76.

5.4 Rotation floating case.

This flight mode is characterized by the use of reaction wheels for the attitude control while the position is not controlled, therefore, the base is left free to translate in the inertial space.

1) Chaser mass 1000 Kg.

Is possible to see that the end effector trajectory is close to a straight line and the error in this case is lower with respect to the free-floating mode.

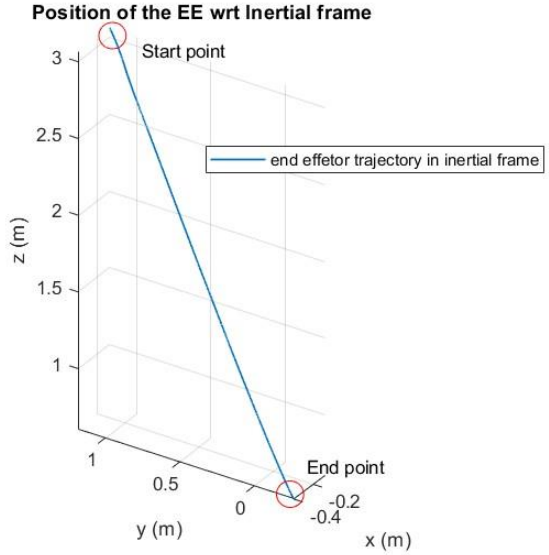

Figure 5.78: End effector trajectory.

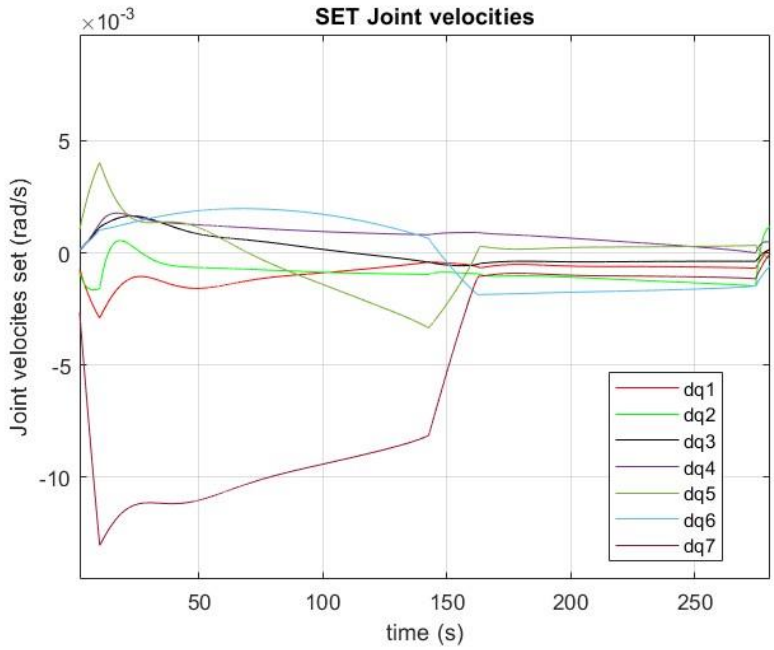

Figure 5.79: Joint velocities set.

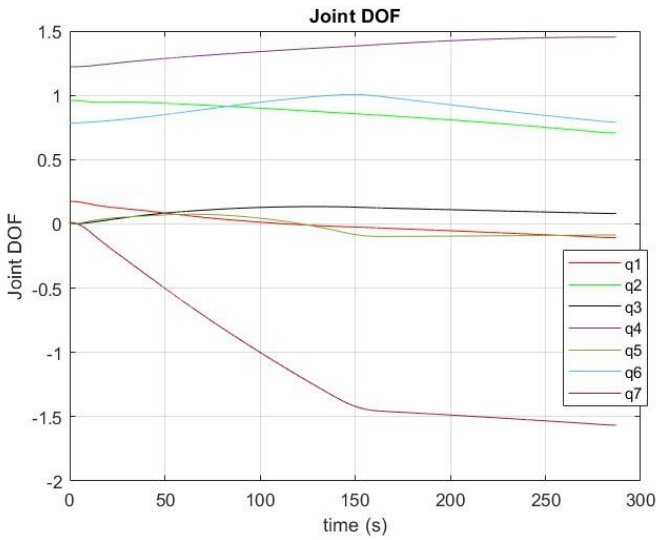

Figure 5.80: Joint DOF.

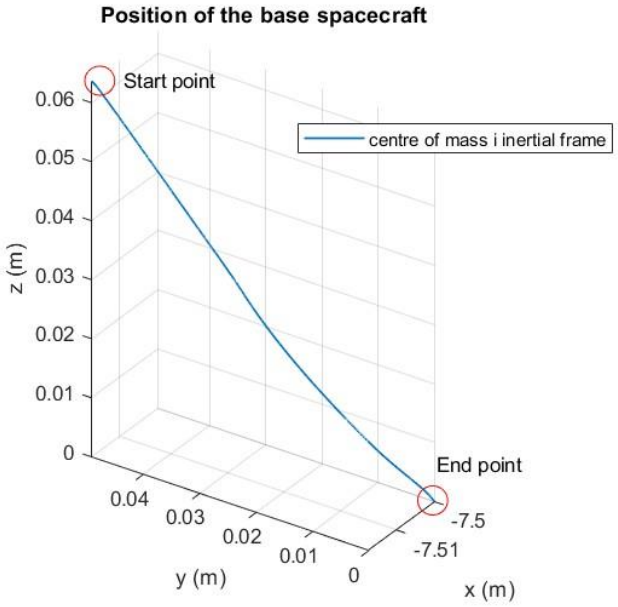

Figure 5.81: COM position.

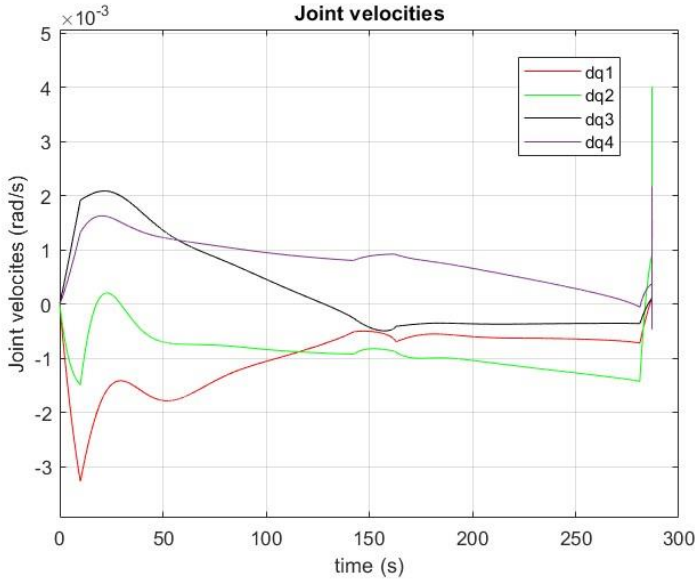

Figure 5.82: Joint velocities feedback.

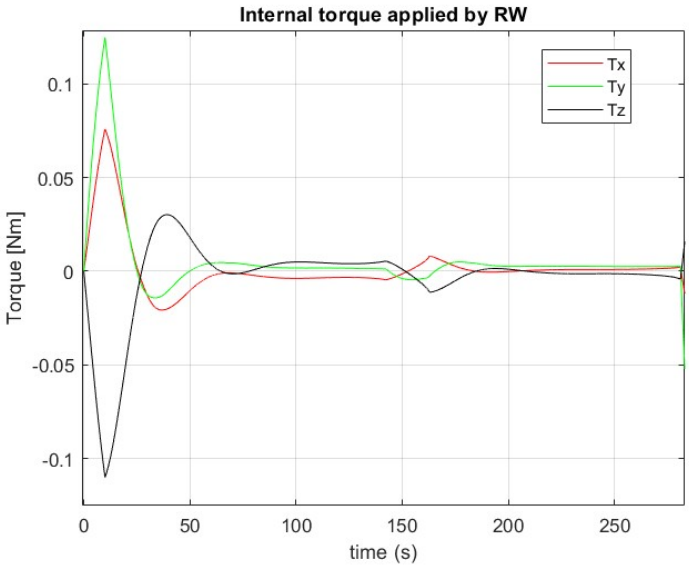

Figure 5.83: Internal torque of the reaction wheel.

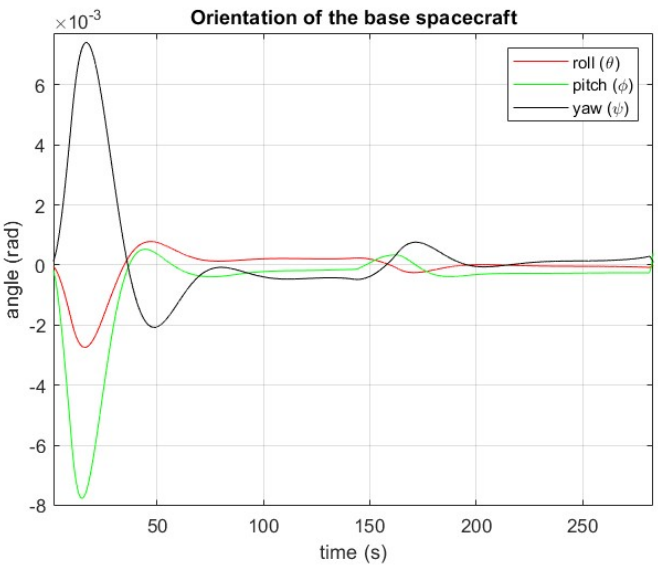

Figure 5.84: Orientation of the base spacecraft.

Notably, the attitude of the spacecraft is below 0.008 rad, which is an acceptable error.

2) Chaser mass 800 Kg.

Decreasing the mass of the spacecraft, the error in the end effector trajectory increase.

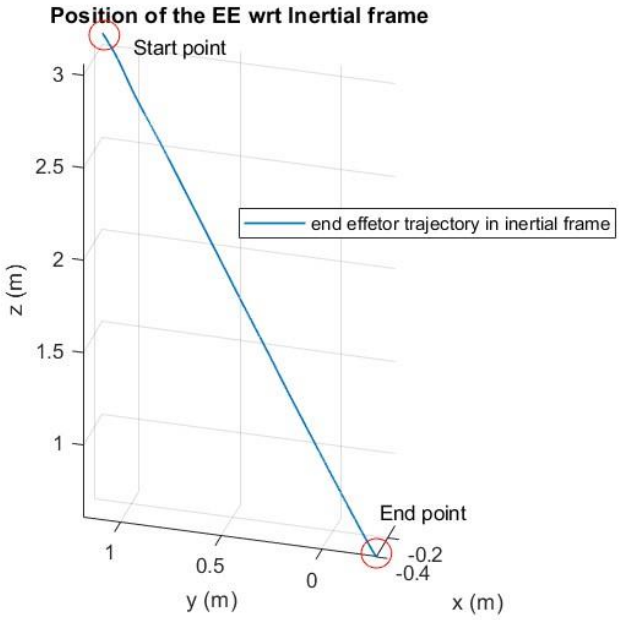

Figure 5.85: End effector trajectory.

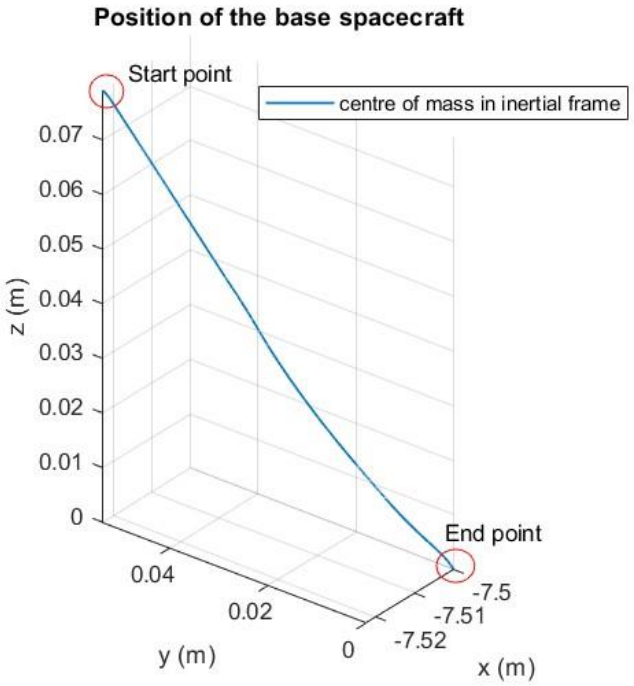

Figure 5.86: COM position.

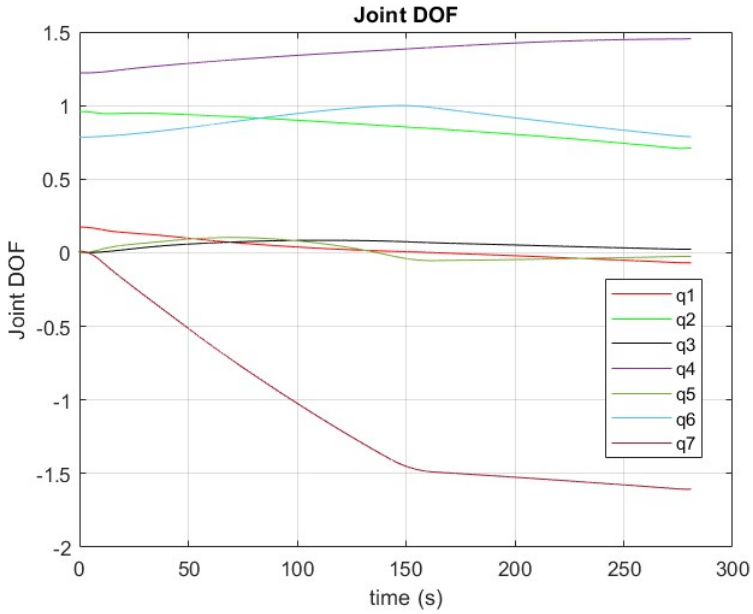

Figure 5.87: Joint DOF.

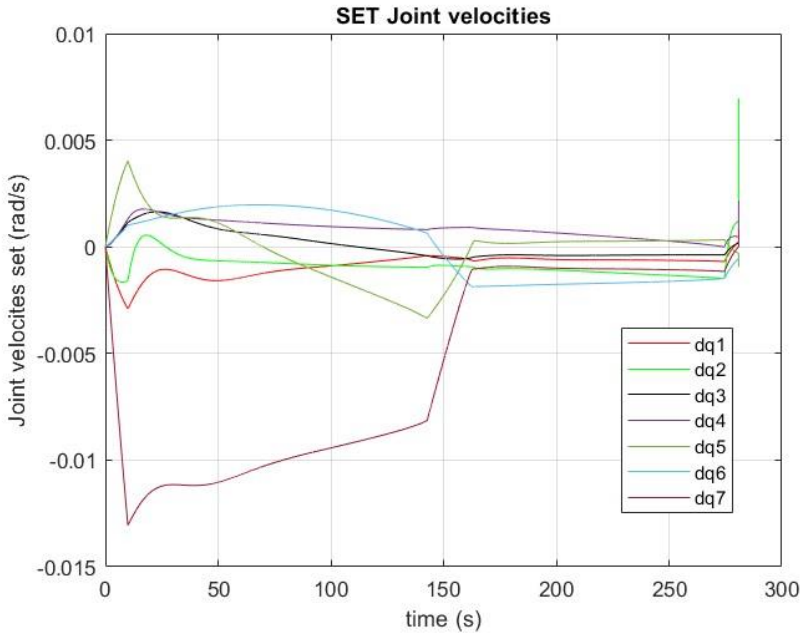

Figure 5.88: SET of joint velocities.

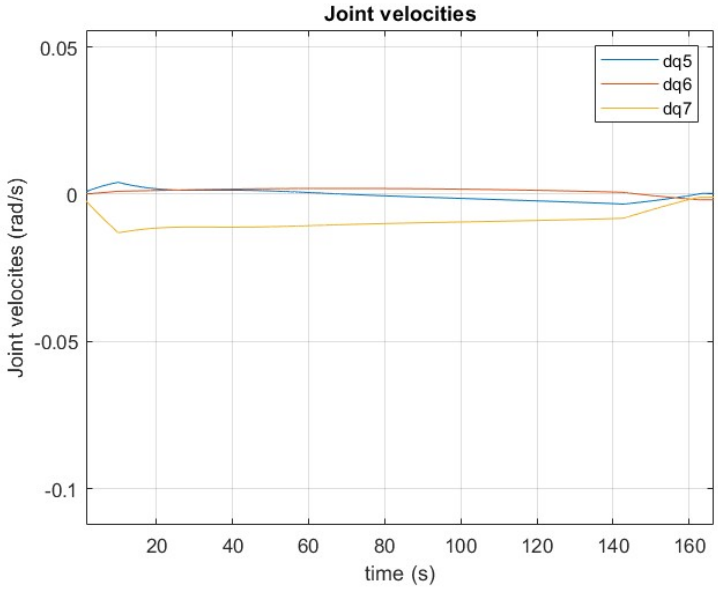

Figure 5.89: Joint velocities feedback.

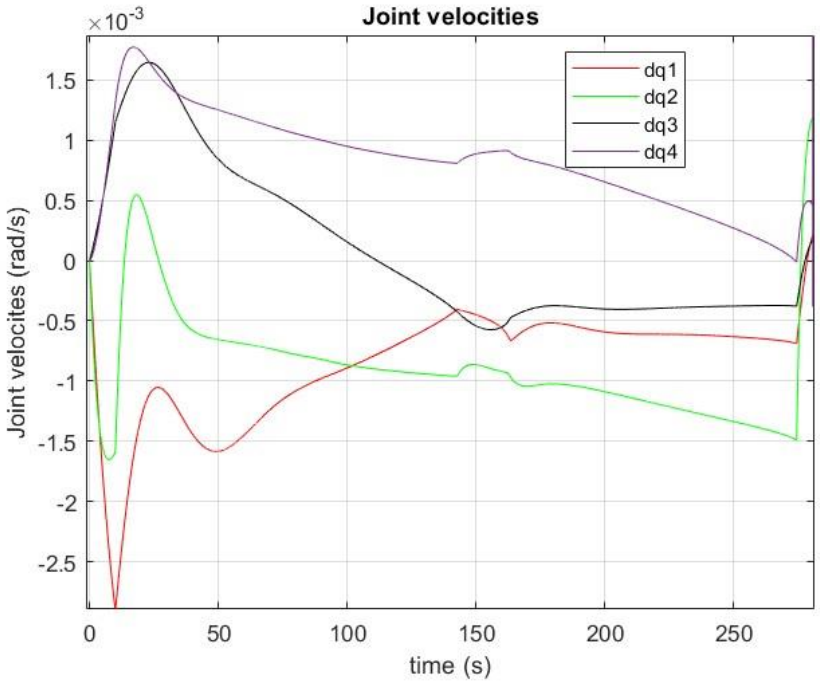

Figure 5.90: Joint velocities feedback.

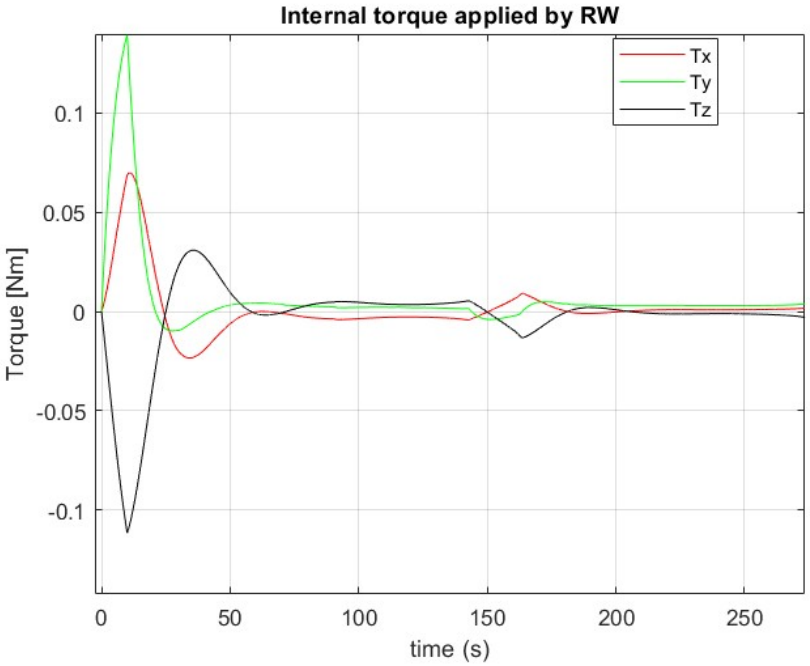

Figure 5.91: Torque applied by Reaction wheel.

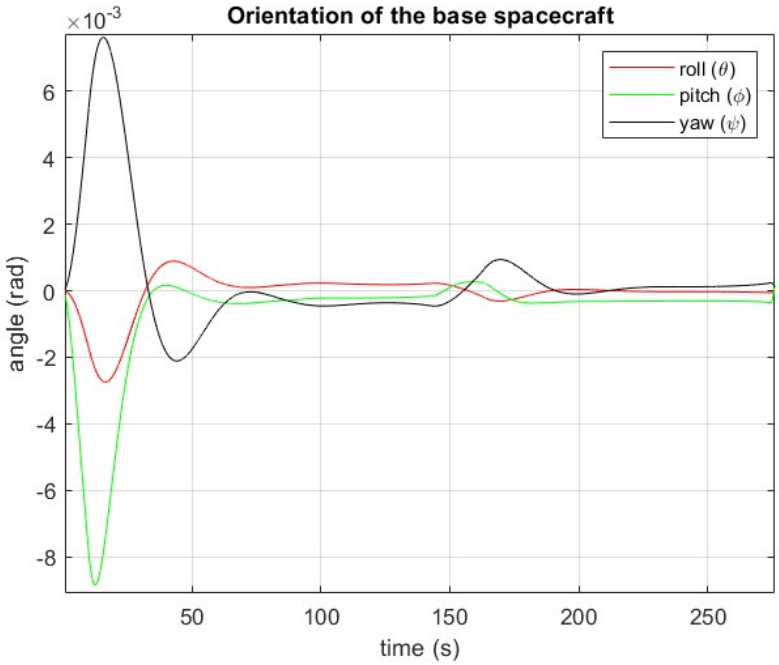

Figure 5.92: Orientation of the spacecraft.

The orientation error reach peaks higher than the previous case, the torque applied by the reaction wheels on the spacecraft is higher as shown in 5.88.

5.5 RBM vs FBM.

Using the free-flying mode, another interesting result can be obtained replacing the flexible links with rigid links. Choosing a mass of 500 Kg, and setting the same parameter used in the section 5.1, the difference in terms of end effector trajectory and disturbance on the spacecraft are reported.

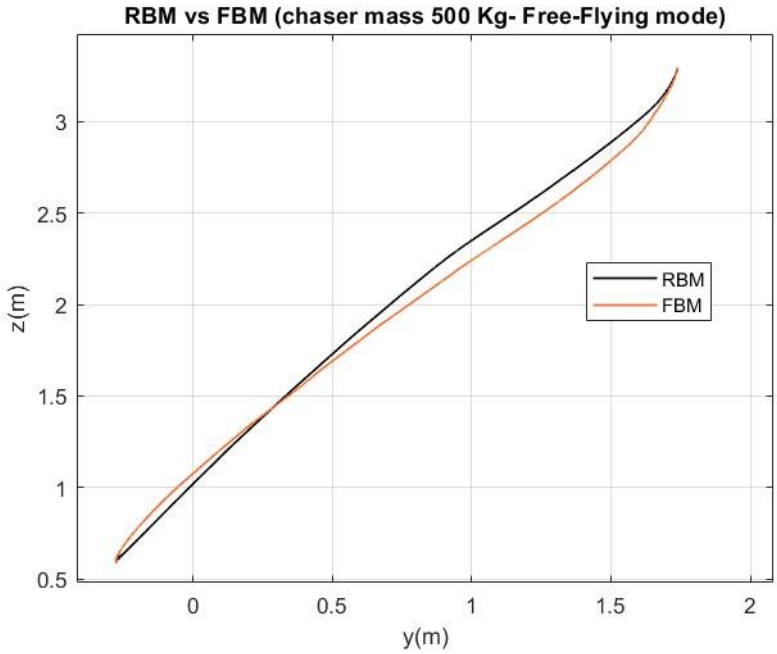

Figure 5.93: End effector trajectory.

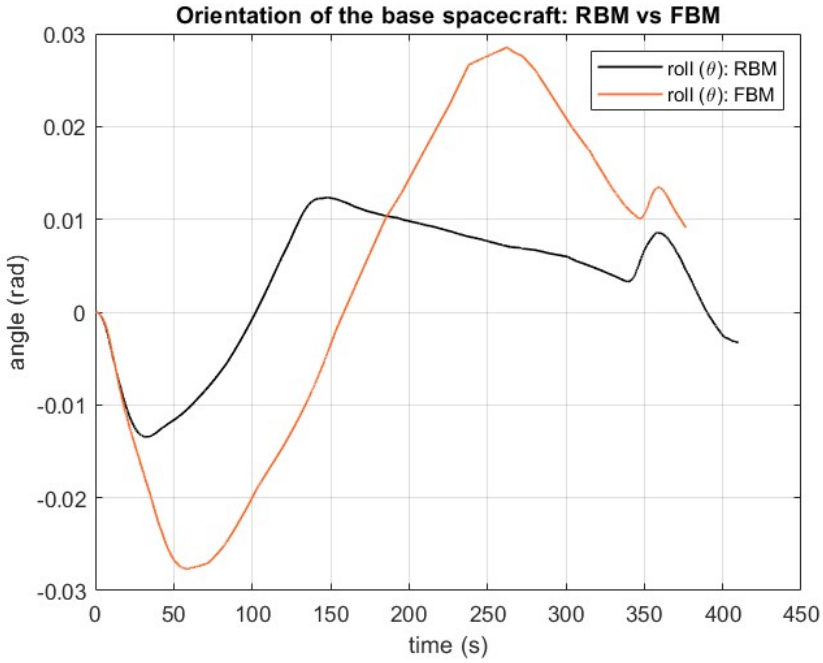

Figure 5.94: Roll angle.

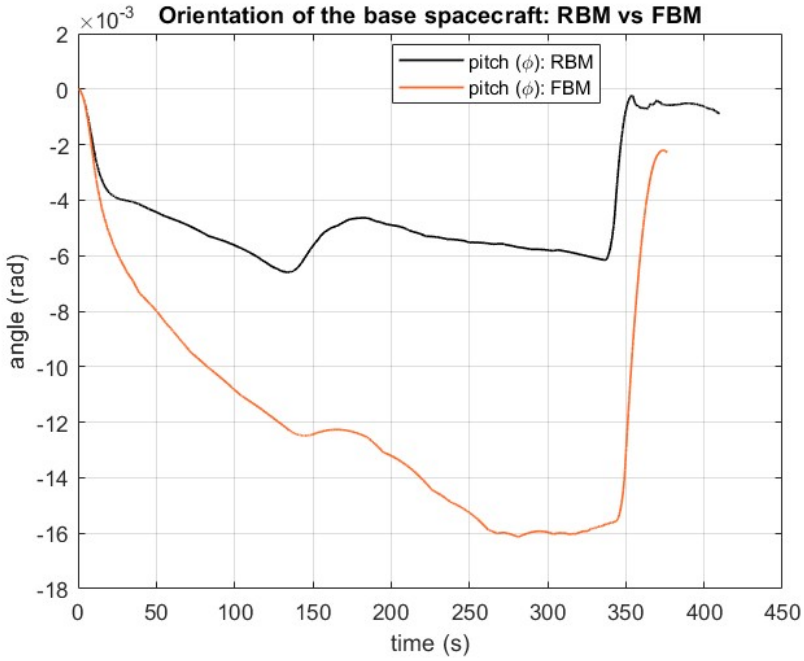

Figure 5.95: Pitch angle.

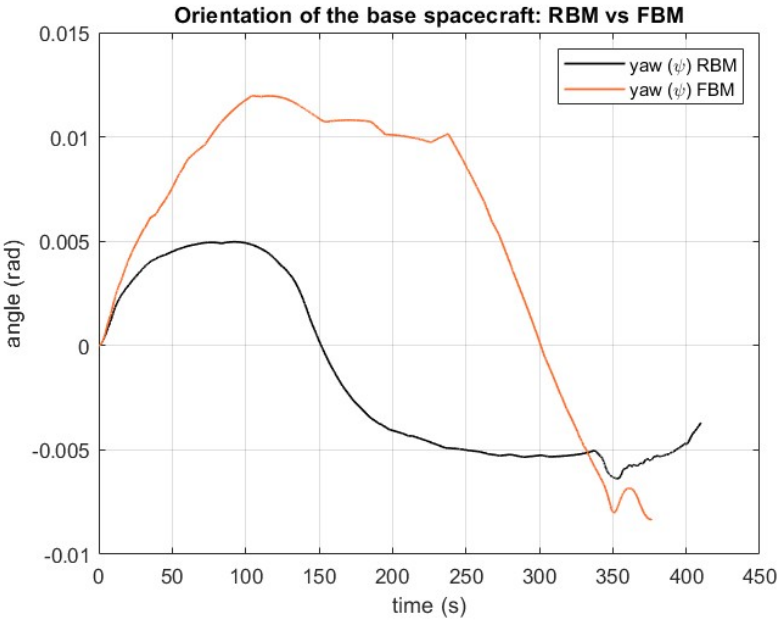

Figure 5.96: Yaw angle.

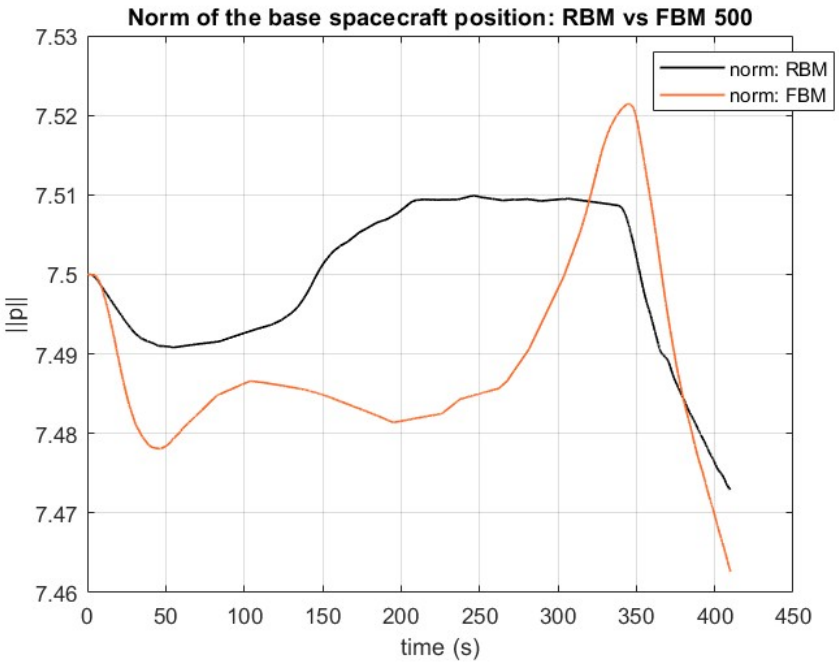

Figure 5.97: Norm of spacecraft position.

As can be noted, the presence of flexible links leads to higher disturbance in attitude and position of the spacecraft.

5.6 Non-diagonal inertia matrix

Setting the free-flying mode, with the same parameters as in the case with a chaser mass equal to 500 Kg. Anther result can be obtained imposing in the multibody property of the brick solid the non-diagonal matrix introduced in section 3.1.3, the results obtained shows some differences from the inertia matrix automatically generated from spacecraft mass. In figure 5.95, the end effector shows a similar behaviour.

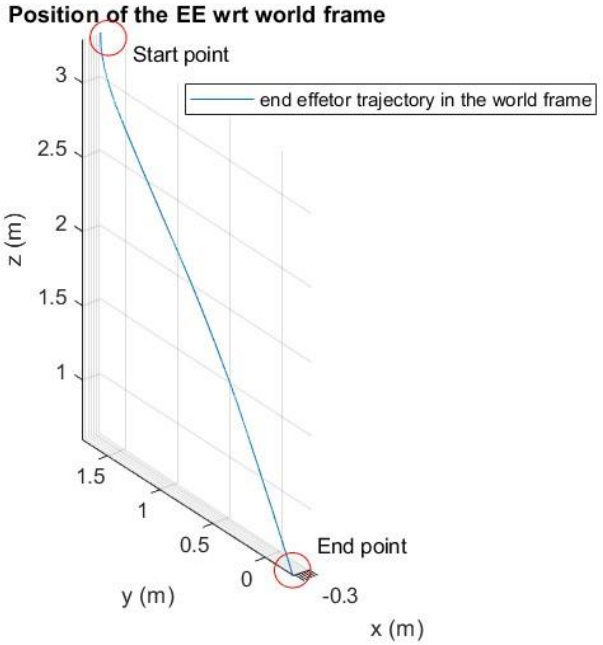

Figure 5.98: End effector trajectory.

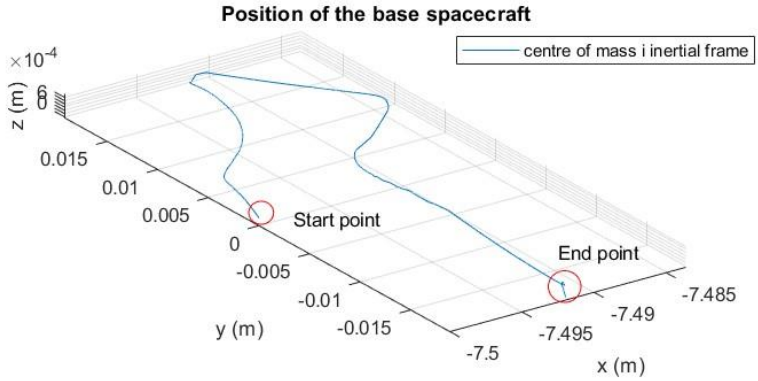

Figure 5.99: Chaser trajectory in the inertial space.

In figure 5.97 the attitude is reported showing a smoother progression in time with lower peaks reached by the attitude error.

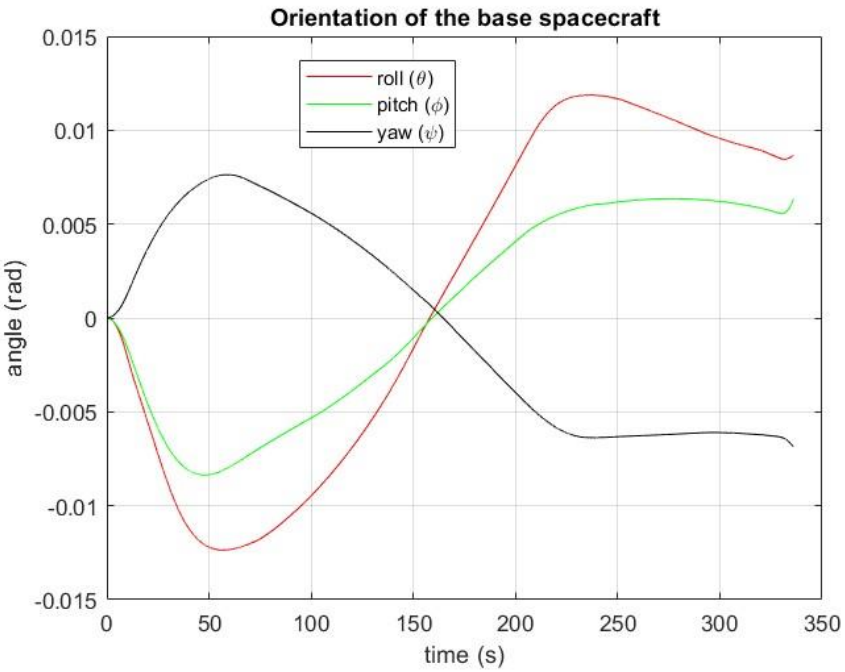

Figure 5.100: Orientation.

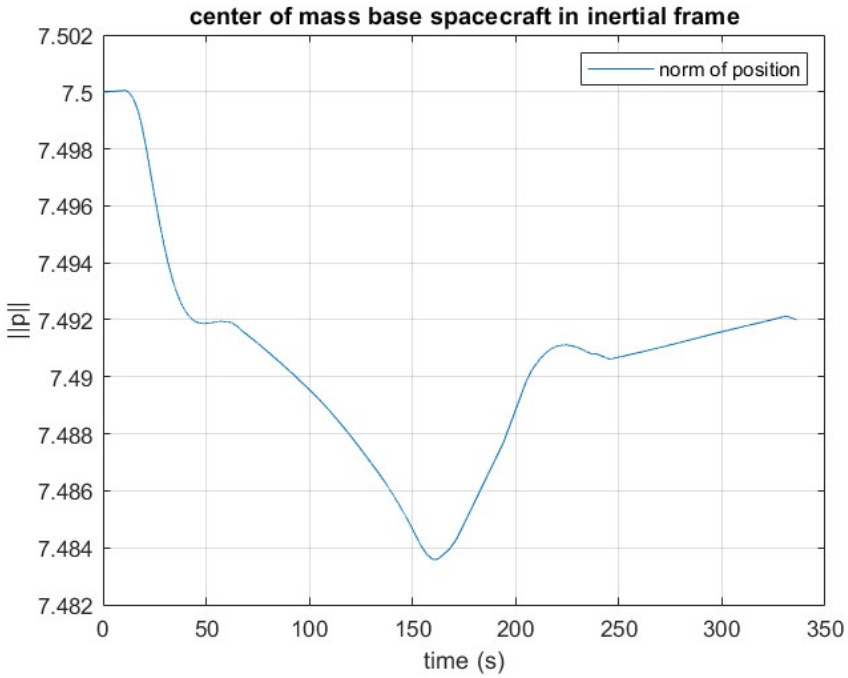

Figure 5.101: Norm of the position.

CHAPTER 6

6 Conclusions and future research

6.1 Dynamic singularities

As argued in literature [49,50,51,52,53], when the system is operating in free-floating, there are no external forces on it since the actuation system is turned off, therefore the system will translate and rotate in response to the manipulator motion. Considering a system whose manipulator has just revolute joints, the robotic arm has generally N DOF controlled, while the spacecraft is described in terms of position and orientation by 6 variables, resulting in a N+6 DOF system. The assumption of a zero angular momentum and an initial zero linear momentum result in a fixed system centre of mass. So, in absence of external forces and torques, the system angular momentum is conserved, and we can write the following conservation equation:

$$ {}^{0}D(q)\ {}^{0}\omega_0 + {}^{0}D_q(q)\dot{q} = 0 \tag{6.1}$$

While, if the initial angular momentum is not zero, we have:

$$ {}^{0}D(q)\ {}^{0}\omega_0 + {}^{0}D_q(q)\dot{q} = R_0^T(\varepsilon, n)h_{CM} \tag{6.2}$$

Where ${}^{0}\omega_0$ is the spacecraft angular velocities expressed in the spacecraft frame, the Nx1 columns vectors $\dot{q}$ and q represents the joint rates and angles, and ${}^{0}D_q$ and ${}^{0}D$ are respectively inertia matrices of appropriate dimension. Although zero initial momentum is desired, small amount of angular momentum can be present, this may cause the system to drift away failing the grasping operations, this is due to some loss in accuracy of the control implemented.
To arrive obtain these equations, referring to [53] we can see that, taking into account the following spacecraft manipulator system:

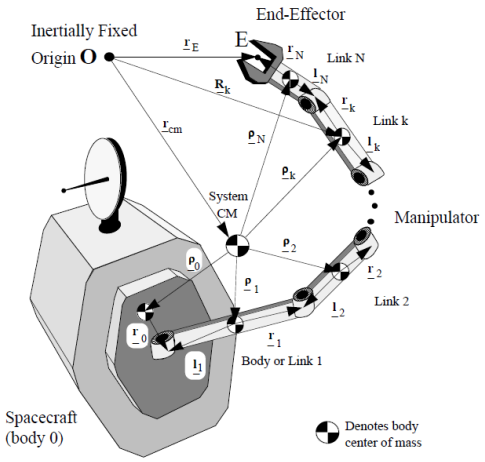

Figure 6.1: Spacecraft manipulator system.

We can express the position of an arbitrary point m $\bar{R}_{k,m}$ on body k:

$$\bar{R}_{k,m} = \bar{R}_k + r_{k,m} \tag{6.3}$$

$$\bar{R}_k = \bar{r}_{cm} + \bar{\rho}_k \tag{6.4}$$

So, the position of the end effector is:

$$\bar{r}_{EE} = \bar{r}_{cm} + \bar{\rho}_N + \bar{r}_N \tag{6.5}$$

From figure 1 we have the following N equations in N+1 unknows:

$$\bar{\rho}_k - \bar{\rho}_{k-1} = \bar{r}_{k-1} - \bar{l}_k \;\; k = 0, \dots, N \tag{6.6}$$

Observing that $\bar{\rho}_k$ vectors are defined in COM we can write:

$$\sum_{k=0}^{N} m_k \bar{\rho}_k = 0 \tag{6.7}$$

Where m_k is he mass of body k. the two previous equations can be solved for $\bar{\rho}_k$as a function of $\bar{r}_k$ and $\bar{l}_k$:

$$\bar{\rho}_k = \sum (\bar{r}_{i-1} - \bar{l}_i)\mu_i - \sum \left(\bar{r}_{i-1} - \bar{l}_i\right)(1 - \mu_i) \;\; k = 0, \dots N \tag{6.8}$$

$$\mu_i = \begin{cases} 0 \; if \; i = 0 \\ \sum_{j=0}^{i-1} \frac{m_j}{M} \;\; i = 1, \dots, N \\ 1 \; id \; i = N + 1 \end{cases} \tag{6.9}$$

With M that is the total mass. Through some steps reported in [53] we obtain:

$$\bar{\rho}_k = \sum_{i=0}^{N} \bar{v}_{ik} \quad k = 0, \dots, N \tag{6.10}$$

$$\bar{v}_{ik} = \begin{cases} \bar{r}_i^* & i < k \\ \bar{c}_i^* & i = k \\ \bar{l}_i^* & i > k \end{cases} \tag{6.11}$$

Since each $\bar{v}_{ik}$ is defined in body i which rotates with angular velocity $\bar{\omega}_i$, and because we assume that the manipulator has only rotative joints, the time derivative would be:

$$\dot{\boldsymbol{\rho}}_k = \sum_{i=0}^{N} \bar{\omega}_i \times \bar{v}_{ik} \quad k = 0, \dots, N \tag{6.12}$$

Therefore, differentiating the end effector position, we can obtain:

$$\dot{\bar{r}_{EE}} = \dot{\bar{r}_{cm}} + \sum_{i=0}^{N} \bar{\omega}_i \times \bar{v}_{iN} + \bar{\omega}_N \times \bar{r}_N \tag{6.13}$$

For this system the linear momentum can be expressed in the following way:

$$\bar{\boldsymbol{p}} = M\dot{\bar{r}_{cm}} = \sum_{k=0}^{N} m_k \dot{\bar{R}}_k \tag{6.14}$$

Notably, since in the free-floating control mode is assumed zero initial linear momentum, therefore, in absence of external force acting on the spacecraft, $\bar{\boldsymbol{p}} = \mathbf{0}$ and since $M \neq 0$, $\dot{\bar{r}_{cm}} = 0$, therefore the COM position in the inertial frame is fixed, so can be chosen the origin of the inertial frame in the system centre of mass. Consequently:

$$\dot{\bar{r}_{EE}} = \sum_{i=0}^{N} \bar{\omega}_i \times \bar{v}_{iN} + \bar{\omega}_N \times \bar{r}_N = \sum_{i=0}^{N} \bar{\omega}_i \times \bar{v}_{iN}{}' \tag{6.15}$$

Where $\bar{v}_{iN}{}'$ is equal to $\bar{v}_{iN}$ for all i,k except for $\bar{v}_{NN}{}'$, $\bar{v}'_{NN} = \bar{v}_{NN} + \bar{r}_N$.
The end effector inertial velocity required to find the Jacobian matrix J*, and is he angular velocity of the last body in the chain:

$$\bar{\omega}_E = \bar{\omega}_N$$

And with a recursive approach we can get:

$$\bar{\omega}_j = \bar{\omega}_0 + \sum_{i=1}^{j} \omega_i^{i-1} \quad j = 1, .., N \tag{6.16}$$

The system angular momentum is:

$$\bar{h} = \bar{r}_{cm} \times \bar{p} + \sum_{k=0}^{N} \{\bar{I}_k \cdot \bar{\omega}_k + m_k \bar{p}_k \times \dot{\bar{p}}_k\} \tag{6.17}$$

Where $\bar{I}_k$ is the inertia dyadic of body k with respect to COM. Assuming initial zero linear momentum the first term is zero, therefore the angular momentum with respect to the in COM is:

$$\bar{h}_{cm} = \sum_{j=0}^{N} \sum_{i=0}^{N} \sum_{k=0}^{N} \bar{D}_{ijk} \cdot \bar{\omega}_j \tag{6.18}$$

Where:

$$\bar{D}_{ijk} = \bar{I}_i \delta_{ij} \delta_{jk} + m_k \{(\bar{v}_{ik} \cdot \bar{v}_{jk})\mathbf{1} - \bar{v}_{ik}\bar{v}_{jk}\} \; for \; i,j,k = 0,..,N \tag{6.19}$$

The dyadic $\bar{D}_{ijk}$ are function of the distribution of inertia through the system and are formed from the barycentric $\bar{v}_{ik}$. The terms $\delta_{ij}\delta_{jk}$ are delta Kronecker.
It can be shown that the angular momentum can be written as:

$$\bar{h}_{cm} = \sum_{j=0}^{N} \sum_{i=0}^{N} \bar{D}_{ij} \cdot \bar{\omega}_j \tag{6.20}$$

Where:

$$\bar{D}_{ij} = \begin{cases} -M\{(\bar{l}_j^* \cdot \bar{r}_i^*)\mathbf{1} - \bar{l}_j^* \cdot \bar{r}_i^*\} & i < j \\ \bar{I}_i + \sum_{k=0}^{N} m_k \{(\bar{v}_{ik} \cdot \bar{v}_{jk})\mathbf{1} - \bar{v}_{ik}\bar{v}_{jk}\} & i = j \\ -M\{(\bar{r}_j^* \cdot l_i^*)\mathbf{1} - \bar{r}_j^* \cdot \bar{l}_i^*\} & i > j \end{cases} \tag{6.21}$$

Where $\mathbf{1}$ is the unit dyadic. In absence of external torques the angular momentum is constant. If there is a certain amount of residual momentum, the system may be able to continue operating for a limited period. In practice to set the momentum to zero, to start the free-floating mode, the attitude control system would be turned on to perform a momentum dump maneuver to eliminate any accumulated momentum.
As we can see, the angular momentum $\bar{h}_{cm}$ is a conserved quantity, or an integral of motion. Therefore, solving the equation for ${}^0\omega_0$, then it provides the angular velocity as a function of the joint rates.

$${}^0D(q)\ {}^0\omega_0 + {}^0D_q(q)\dot{q} = 0 \tag{6.22}$$

Integrating the equation, a second time, we would obtain the spacecraft attitude as a function of joint rates, in this case we could eliminate the attitude from the kinematics and dynamics equation. However, as known the spacecraft attitude is described by the matrix $R_0\ (\varepsilon, n)$, and it is known that:

$$\dot{R_0}\ (\varepsilon, n), = R_0\ (\varepsilon, n)\ {}^0\omega_o^x \tag{6.23}$$

Combining it with the previous equation, in case of non-zero initial angular momentum, we obtain:

$$\dot{R_0}\ (\varepsilon, n) = R_0\ (\varepsilon, n)\{\ {}^0D^{-1}(T_0^T h_{cm,0} - {}^0D_q\dot{q})\}^{\times} \tag{6.24}$$

${}^0D^{-1}$ is aways non-singular since it represents the inertia of the whole system. Trough Frobenius condition can be demonstrated that the obtained equation is not integrable, another way to show his result is from a physical point of view as reported in [53]. So, the attitude of the spacecraft is not only a function of joint angles q, but also of the path taken in joint space; different path in the joint space will result in a different attitude of the spacecraft.

To construct in system Jacobian J^* the kinematics expression needs to be expressed in matrix form. Since in the SMS system all joints are revolute, a reference frame with axes parallel to each body's axes is attached to each COM. The column vector ${}^{i}\bar{v}_{ik}$ can be expressed in ram I and transformed in the inertial frame in the following way:

$$v_{ik} = T_i \ {}^{i}v_{ik} = T_0 \ {}^{0}v_{ik} \tag{6.25}$$

$${}^{0}v_{ik} = \ {}^{0}T_i \ {}^{i}v_{ik} \tag{6.26}$$

Where T_i is a transformation matrix given by:

$$T_i(\boldsymbol{e}, n, q_1, \dots, q_i) = T_0(\boldsymbol{e}, n) \ {}^{0}T_i(q_1, \dots, q_i) \tag{6.27}$$

$${}^{0}T_i(q_1, \dots, q_i) = \ {}^{0}A_1(q_1) \dots {}^{i-1}A_i(q_i) \tag{6.28}$$

Note that ${}^{i-1}A_i(q_i)$ transforms a column vector in frame I to a column vector expressed in frame i-1 and is a function of the relative joint angle of the two frames q_i. The 3x3 inertia matrix $T_0(\boldsymbol{e}, n)$ can be committed using quaternions or Euler parameters:

$$T_0(\boldsymbol{e}, n) = (n^2 - e^T e)1 + 2ee^T + 2ne^{\times} \tag{6.29}$$

$$T_0(\boldsymbol{e}, n) = (n^2 - e^T e)1 + 2ee^T + 2ne^{\times} \tag{6.30}$$

$$\boldsymbol{e}(\boldsymbol{a}, \theta) = \boldsymbol{a}\sin(\theta/2)$$

$$\boldsymbol{n}(\boldsymbol{a}, \theta) = \cos(\theta/2)$$

Where $\boldsymbol{a}$ is a unit vector of an axis around which the spacecraft is rotated of an angle θ. Therefore, the scalar form of the angular velocity is:

$$\boldsymbol{\omega}_j = \boldsymbol{\omega}_0 + \boldsymbol{\omega}_j^0 = \boldsymbol{\omega}_0 + T_0 \sum_{i=1}^{j} \ {}^{0}T_i \ {}^{i}u_i \, \dot{q}_i \tag{6.31}$$

$$\boldsymbol{\omega}_j = \boldsymbol{\omega}_0 + T_0 \ {}^{0}F_j \dot{q} \tag{6.32}$$

Where ${}^{i}u_i$ is the unit column vector in frame i parallel to revolute axis through joint i, and ${}^{0}F_j$ is a 3xN matrix given by:

$${}^{0}F_j = \left[\ {}^{0}T_1 \ {}^{1}u_1, \ {}^{0}T_2 \ {}^{2}u_2, \dots, \ {}^{0}T_j \ {}^{j}u_j, 0 \right] \ \ j = 1, \dots, N. \tag{6.33}$$

Where 0 is a 3x(N-j) zero element matrix, and:

$$q = [q_1, \dots, q_N]^T$$

In the end we obtain the following equations:

$$\dot{r}_E = T_0\{\ {}^0J_{11}\ {}^0\omega_0 + \ {}^0J_{12}\dot{q}\} \tag{6.34}$$

$$\omega_E = T_0\{\ {}^0\omega_0 + \ {}^0J_{22}\dot{q}\} \tag{6.35}$$

$${}^0D(q)\ {}^0\omega_0 + \ {}^0D_q(q)\dot{q} = 0 \tag{6.36}$$

Where:

$${}^0J_{11} = -\sum_{i=0}^{N}\left[\ {}^0T_i\ \ {}^iv_{iN}{}'\right]^{\times} \tag{6.37}$$

$${}^0J_{12} = -\sum_{i=1}^{N}\left[\ {}^0T_i\ \ {}^iv_{iN}{}'\right]^{\times}\ {}^0F_i \tag{6.38}$$

$${}^0D_j = \sum_{i=0}^{N}\ {}^0D_{ij}\quad (j = 0, \dots, N) \tag{6.39}$$

$${}^0D = \sum_{j=0}^{N}\ {}^0D_j \tag{6.40}$$

$${}^0D_q = \sum_{j=1}^{N}\ {}^0D_j\ \ {}^0F_j \tag{6.41}$$

Therefore, the free-floating system Jacobian is J*, we can write:

$${}^0\omega_0 = \ {}^oD(q)^{-1}(-\ {}^0D_q(q)\ \dot{q}) \tag{6.42}$$

Substituting into the previous equations we obtain:

$$J^*(e,n,q) = diag(T_0, T_0)\begin{bmatrix} -\ {}^0J_{11}\ {}^oD^{-1}\ {}^0D_q(q)\ +\ {}^0J_{12} \\ -\ {}^0D^{-1}\ {}^0D_q(q) + \ {}^0J_{22} \end{bmatrix} = diag(T_0, T_0)\ {}^0J(q)^* \tag{6.43}$$

6.1.1 Dynamic singularities

After having found a way to construct the generalized Jacobian that relates the motion of the end effector to the joint rates, can be observed that, as in the fixed based robotic manipulator there are kinematic singularities where the determinant

of the Jacobian become zero, so it is in the spacecraft manipulator case, where the singularities are called dynamic singularities. In the fixed based case singularities occurs at the workspace boundaries and when end effector velocity due to the motion of one joint is parallel to the velocity due to the motion of some other joint.
In the singular points the at least one DOF is lost, and the Jacobian matrix is not of full rank. At this singularity points or in its neighbour, the velocities required to the motor are high and cannot be performed.
Lastly, the singularities of the fixed-base robotic arm depend only on the kinematic structure of the manipulator, and they usually correspond to a specific point in the workspace.
In the free-floating case the singularities are obtained analysing the following result:

$$\det\left[\ {}^{0}J^{*}(q)\right] = 0 \tag{6.44}$$

Where the term $diag(T_0, T_0)$, translating the information of attitude of the spacecraft in the inertial frame, has not been considered since it is always square and invertible. So, all singularities are function of the manipulator configuration with respect to its spacecraft.
Therefore, the dynamic singularities correspond to specific points in the joint space, so they cannot be mapped into points in the workspace, indeed we can observe this as a consequence of the fact that the position of the EE and its orientation are function of joint variable and the spacecraft attitude as already sowed earlier:

$$\bar{r}_{EE} = \bar{\rho}_N + \bar{r}_N = T_0(e,n) \sum_{i=0}^{N} {}^{0}T_i(q_1,..,q_i)\ {}^{i}v'_{iN} \tag{6.45}$$

$$T_{EE}(e,n,q_1,..,q_N) = T_0(e,n)\ {}^{0}T_N(q_1,..,q_N) \tag{6.46}$$

So, for a given position and orientation of the EE the last relationship cannot be inverted to obtain the manipulator's angles, because the spacecraft attitude is not a function of the joints angles but of the path followed by the EE in the inertial space, that's because of the non-holonomic nature of the angular momentum.
So, specific point in the workspace can be reached with infinite system configurations q and spacecraft attitude (e, n), therefore the dynamic singularities in the workspace are path dependent.
Notably, all resolved rate or resolved acceleration control scheme will fail because at these points the GJM has no inverse.

6.1.2 Space manipulator workspace

As seen in the previous section, dynamic singularities are a function of the configuration and on path taken by the end effector. The solutions of the determinant of the Jacobian are a collection of hypersurfaces in the manipulator joint space $Q_{s,i}$(i=1, 2,..). These hypersurfaces are composed of points q that result in dynamically singular configurations. These hypersurfaces become surface in 3D joint space for spatial system and curves in the planar case. The workspace can be obtained as follow:

$$R = R(q) = \left\| \sum_{i=0}^{N} {}^{0}T_i(q_1, \dots, q_i)\ {}^{i}v'_{iN} \right\| \tag{6.47}$$

Where $\|\cdot\|$ denotes the vector length. As can be seen in [52, 53], the workspace is defined around the system COM, and consist in a spherical shell for the spatial system case in the inertial space. Furthermore, can be noted that the attitude of the spacecraft does not change the length of the vector.

Based on the definition of the dynamic singularities we can divide the workspace in two regions, the so called PDW (path dependent workspace) and the PIW (path independent workspace). In the PIW dynamic singularities does not occur and every desired trajectory of the end effector can be performed, instead in the PDW dynamic singularities can occur depending on the path taken by the end effector. The union of these two areas is called the reachable workspace. Clearly, the aim of the designer is to reduce the PDW and increase as much as possible the PIW, to find the portion of workspace in the PSW area we can write:

$$R_{min,i} = \min_{q\_Q_{s,i}} R(q) \quad i = 1,2, \dots \tag{6.48}$$

$$R_{max,i} = \max_{q\_Q_{s,i}} R(q) \quad i = 1,2, \dots \tag{6.49}$$

6.1.3 Avoidance of dynamic singularities.

In general, to reduce the effect of DSs we can maximize the PIW using manipulator redundancy, or this can be achieved by choosing a high mass and inertia of the base spacecraft in such a way that the manipulator activity becomes a negligible disturbance for the spacecraft.

To avoid the dynamic singularities problem, there are different research that works with manipulator redundancy or point-to-point path planning technique [56, 57]. However as mentioned in [52,53], the redundancy complicates the system design. Therefore, as remarked in [52,58], has been introduced a technique which permits to perform straight-line planning of a non-redundant free-floating system, and later it has been developed into a method allowing motion along any end-effector path that may lay in the entire workspace.

The basic idea of this method is to avoid dynamic singularities for a given predefined path, defined as $v_E(t)$ in the workspace, by choosing the initial attitude of the spacecraft system configuration that avoid dynamic singularities during the end-effector motion. To start with it, we should recall a description of the system.

The required minimum number of manipulator joints to perform an end effector trajectory $v_E(t)$ in e spatial space is six N=6, so the dynamic singularities do not occur when, as seen earlier, the generalized Jacobian matrix has full rank:

$$\det\left({}^{0}J^{*}(q) \right) = S(q) \neq 0 \tag{6.50}$$

$${}^{0}J^{*}(q) = -\ {}^{0}J^{*}_{11}\ {}^{0}D^{-1}\ {}^{0}D_q + \ {}^{0}J^{*}_{12} \tag{6.51}$$

In joint space, $\det\left({}^{0}J^{*}(q)\right) = S_0 = const$, defines surface in the spatial systems or curves in planar systems. If $S_0 = 0$ they will be called singularity surfaces or singularity curves.
Therefore, the DS avoidance is achieved if during the motion the previous relationships is satisfied, this is possible if during the motion $S(q)$ does not become zero and so, since $S(q)$ can take positive and negative values, its minimum value is $S_{min} > 0$ is $S > 0$ or its maximum value if $S_{max} < 0$ is $S < 0$. As for the fixed base method even values of S close to zero are undesired since they may result in large values in accelerations and torques. As showed in [58], S is a function of the configuration q, since q is a function of the time and the initial condition $q = f(t; \epsilon_{in}, n_{in}, q_{in})$. Moreover, it can be showed that S_{min} is a continuous function of the initial spacecraft attitude $S_{min} = f(t; \epsilon_{in}, n_{in}, q_{in})$ where for a given end effector initial position, q_{in} depends only on the spacecraft initial attitude, so we can write:

$$S_{min} = f(t; \epsilon_{in}, n_{in}) \tag{6.52}$$

In the paper two cases are distinguish, depending on when the $\dot{S} = 0$. The function can have a minimum either in an intermediate point along the end effector trajectory or in at a path extreme point. In the first case the manipulator configuration q which corresponds to minimum value of $S(q)$ is computed setting the time derivative of $S(q)$ equal to zero:

$$\dot{S} = \nabla S^T \dot{\boldsymbol{q}} = 0 \tag{6.53}$$

Where ∇ is the Nabla operator used to obtain the gradient of the function. If the desired minimum value of S during the motion is S_0, then we get:

$$S(q) = S_0 \tag{6.54}$$

The above equations represent the necessary condition for the computation of the manipulator configuration.
In the second case, the minimum value of S corresponds to the initial or final value of S, or equivalently of the initial and final position of the end effector. Since the end effector distance from the origin is a function of the manipulator configuration only, the equation is as follows:

$$R(q) = R_i \tag{6.55}$$

Where R_i represents the radius of the initial and final end-effector position.
In [58] the method is first applied to the planar system and then expanded to the spatial system, for the free-floating case manipulator. The end effector position and orientation are described as follow:

$$\begin{aligned} r_{EE} &= [x_{EE}(\varepsilon, n, q) \quad y_{EE}(\varepsilon, n, q) \quad z_{EE}(\varepsilon, n, q)]^T \\ &= [x_{EE}(s) \quad y_{EE}(s) \quad z(s)] \end{aligned} \tag{6.56}$$

$$R_{EE}(\varepsilon, n, q) = R_{EE}(s) \tag{6.57}$$

Where R_{EE} is the rotation matrix which describe the orientation of the end-effector with respect to e inertial frame, s is a parametrization given in [58].
The determinant of the generalized Jacobian is given by:

$$S(q) = \det(S^*) = S(q_1, q_2, q_3, q_4, q_5, q_6) \tag{6.58}$$

Since the representation of singularities in a 6-dimensional joint space is complex, the analysis can be simplified using a spherical wrist which allow the analysis to be reduced to the study of a 3-dof manipulator. So, the end-effector position will be described by:

$$r_E = r_3 + R_0 \, {}^0R_6 \, {}^6u_{6,E} \tag{6.59}$$

Where r_3 is the third link barycentre reported in the figure 6.2 [58], ${}^0R_6$ is the rotation matrix between the last-link frame and the spacecraft frame, and ${}^6u_{6,E}$ is the fixed barycentric vector. Therefore, as outlined in [58], the motion of the end effector can be studied through the motion of point 3 in figure …. So the generalized Jacobian is given by:

$$S = \det(S^*) = S(q_1, q_2, q_3) \tag{6.60}$$

So, the relationship with the gradient can be simplified to:

$$\dot{q}_1 \frac{\partial S}{\partial q_1} + \dot{q}_2 \frac{\partial S}{\partial q_2} + \dot{q}_3 \frac{\partial S}{\partial q_3} = 0 \tag{6.61}$$

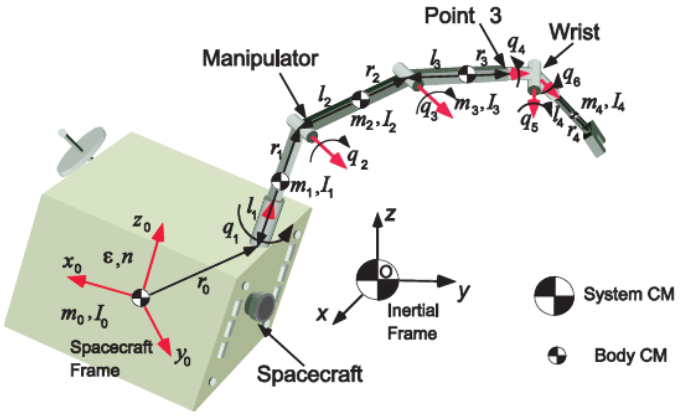

Figure 6.2: Spatial free-floating.

The problem can be further simplified if the manipulator is mounted along a principal axis of inertia, indeed in this case we can observe the following result:

$$S = \det(S^*) = S(q_2, q_3) \tag{6.61}$$

So, the dynamic singularities do not depend on the first degree of freedom and can be shown that the equation has analytical solutions similar to the one obtained for the planar system [58].

6.2 Flexible joints.

As already stated in space application to reduce the launch mass and increase the workspace the design of lightweight and long reach manipulators is strongly prefeed. The drawback is the increasing structural flexibility of the links causing vibrations.
Is possible to study he system performance in different ways. As reported in [61], if the system has short links all system flexibilities can be lumped into joint flexibilities aiming in studying their effects in the design of control system and in the endpoint positioning. If this is not a valid assumption then, the coupling of the link flexibility and joint flexibility has to be considered, since it may lead to elastic deformation and vibration of the system, which could also accelerate the fatigue of the mechanical component. Therefore, an investigation on dynamic modelling can be found in [69].
In [92], the model is obtained assuming the joint to be a linear elastic spring, while in [93] is developed a flexible joint with non-linear elasticity and friction.
The manipulator natural frequencies continuously change with the manipulator configuration and payload, moreover, when handling large payloads, manipulator joint or structural flexibility becomes important and can presents payload attitude controller fuel-replenishing dynamic interaction.
In [68], a study the possible dynamic interactions between the attitude controller of a spacecraft and the flexible modes of a space manipulator is reported.
Future research in this field could be the implementation of the flexibility of the joints into the spacecraft model and analyse through simulations its impact on the robotic arm trajectory and on the attitude and position of the spacecraft.

6.3 Tumbling target.

Capturing a non-cooperative target, such as an out-of-control satellite or a tumbling space debris, is still a very challenging and risky operation. One of the major problems is the attitude disturbances caused by the dynamic interaction between the target and the chaser, which may cause spacecraft destabilization or damage to the hardware. Several approaches are proposed in literature to deal with this problem.
In [93], the contact between free-flying space robots, and the minimization of the impulse contact are studied. While in [94], a discussion on impact dynamics analysis when a free-floating space robot has impact with infinite small time period by the contact and capturing of a target, moreover, the concept of the reaction null space is applied to estimate and minimize the impulsive reaction force or attitude disturbance at the base of a space robot.
In [95], a methodology that uses the property of the percussion point of bodies that can rotate about a fixed axes is presented to minimize the forces transmitted to the base of the manipulator when grasping an object.
In reference [96], propose an approach to search the optimal approach trajectory of space robot in order to minimize or avoid the impact, using a genetic algorithm.
While in [7], is proposed an approach to nullify the attitude disturbance at the moment of capture. This is done by predicting the best capturing time and configuration such that the contact force resulting from the physical contact passes through the centre of mass of whole servicing system. In this way the attitude disturbance caused by the contact is almost zero or minimal.
The problem of the optimal trajectory planning for a space manipulator was addressed in [97], where a motion planning for a zero-reaction manipulator was developed. In [98] is proposed an optimal control method of a free-floating space robot to capture a tumbling target.
A future progress in this area could be to use the multibody environment to simulate a capturing scenario using one of the concepts introduced in the above-mentioned papers. Other possibility would be to create a proper simulation environment with a flight manager function which can manage the guidance and control modes based on the phase of the space mission.

6.4 Conclusions

The main goals in this research were to develop a complete simulation environment in MATLAB/Simulink for the analysis of an OOS manoeuvre using a flexible link robotic arm designed by the Mechanical department of Polytechnic of Turin, to address the issue of the dynamic coupling between the base of chaser spacecraft and the robotic arm movement in a multibody environment, to validate the use of fixed based algorithm in space applications and provide a multibody application of the Generalized Jacobian matrix in the free-floating and rotation floating mode.

In chapter 2, the active stabilization system is designed, with a block diagram representation of the thruster and the reaction wheel. In this section the thruster allocation function is developed using the pseudo-inverse of the configuration matrix and the PWPFM to convert the continuous signal from the controller into an ON/OFF signal.

Chapter 3 focuses on the design of the controller. The position and attitude control is performed using the optimal controller LQR and the APF function for obstacle avoidance.

Chapter 4 focuses on the development of the differential kinematics algorithms to control the robotic arm in the different motion controls.

Lastly, in chapter 5 the chosen parameters are presented, and different scenarios are analysed. The model is first simulated in the free-flying mode then in translation-flying mode, lastly in free-floating and rotation-floating modes. Then the differences are observed in terms of the robotic arm parameters and position and attitude errors of the base of the spacecraft. Furthermore, a comparative study between flexible body and rigid body model is reported.

To conclude, the research objective has been achieved. A simulation environment with different scenarios have been developed and validated, a control law for the capture of a non-cooperative space debris has been designed and the use of fixed based algorithm and GJM for robotic arm manoeuvring are accomplish. Furthermore, a demonstration of the robustness of LQR controller designed using the quaternion reduced model, and its asymptotically stability, is shown.

6.5 Future developments.

Many different improvements could be made to the present thesis to enhance the modelling and control of space robots.

First, a modelling of an orbiting system can be carried out to include the orbital dynamics in the simulation environment, together with it a GNC design can be implemented with a flight manager which switch the modes of the GNC function based on the phase of the rendezvous. Furthermore, for a more realistic simulation environment, state observers and estimation filters, like Kalman filter, could be added to absolve the navigation function.

The model could be extended to include multiple robotic arms, where one of them perform the grappling phase and the other maintenance activities, such as fuel refilling.

Further studies could include fuel sloshing, an accurate system for obstacle avoidance, fuel impingement and variable mass due to the fuel consumption and the change of the spacecraft centre of mass in terms of position.

Moreover, flexible appendices can be added to the model, like flexible solar panels, or flexible joints.

Other important topic could be the implementation of the deployment phase of the POPUP robotic arm, and the study of the dynamic singularities in the 3 DOF case together with the trajectory planning to avoid them.

Lastly, the case with a tumbling target could be implemented with a de-tumbling phase and the execution of successive grasping operation.

Bibliography

[1] Evangelos Papadopoulos, Farhad Aghili, Ou Ma and Roberto Lampariello, Robotic Manipulation and Capture in Space: A Survey, 19 July 2021, frontiers in robotics and AI.

[2] Luo, Z., and Sakawa, Y. (1990). A Control of a Space Manipulator for Capturing a Tumbling Object. Proc. 29th IEEE Conf. Decis. Control., 103–108

[3] Papadopoulos, E., and Moosavian, S. (1994). Dynamics and Control of Multi-Arm Space Robots during chase and Capture Operations. IEEE/RSJ Int. Conf. Intell. Robots Syst., 1554–1561.

[4] Papadopoulos, E. (1991). On the Dynamics and Control of Space Manipulators. Cambridge, MA, USA: Dept. of Mechanical Engineering, Massachusetts Institute of Technology. doi:10.23919/acc.1991.4791629

[5] Yoshida, K., Dimitrov, D., and Nakanishi, H. (2006). On the Capture of Tumbling Satellite by a Space Robot. Beijing, China: IEE/RSJ Int. Conference on Intelligent Robots and Systems. doi:10.1109/ iros.2006.281900

[6] Yoshida, K. (2003). Engineering Test Satellite VII Flight Experiments for Space Robot Dynamics and Control: Theories on Laboratory Test Beds Ten Years Ago, Now in Orbit. Int. J. Robotics Res. 22 (5), 321–335. doi:10.1177/0278364903022005003

[7] Flores-Abad, A., Zhang, L., Wei, Z., and Ma, O. (2017). Optimal Capture of a Tumbling Object in Orbit Using a Space Manipulator. J. Intell. Robot Syst. 86 (2), 199–211. doi:10.1007/s10846-016-0417-1

[8] Aghili, F. (2020). Optimal Trajectories and Robot Control for Detumbling a Noncooperative Satellite. AIAA J. Guidance, Control Dyn. 43 (2), 1952–1959. doi:10.2514/1.g005037

[9] Aghili, F. (2012). A Prediction and Motion-Planning Scheme for Visually Guided Robotic Capturing of Free-Floating Tumbling Objects with Uncertain Dynamics. IEEE Trans. Robotics 28 (3), 634–649. doi:10.1109/tro.2011.2179581

[10] Aghili, F., and Parsa, K. (2009). Motion and Parameter Estimation of Space Objects Using Laser-Vision Data. J. Guidance, Control Dyn. 32 (2), 538–550.doi:10.2514/1.37129

[11] Center, G. S. (2010). On-Orbit Satellite Servicing Study. National Aeronautics and Space Administration (NASA).

[12] Marko Jankovic, Jan Paul, Frank Kirchner, GNC architecture for autonomous robotic capture of a non-cooperative target, Peliminary design.

[13] Lau Yazhoung, Zhang Jin, Tang Guojin, Survey of orbital dynamics and control of space rendezvous ,2013

[14] Fehse, Automated Rendevouzs and docking of Spacecraft.

[15] Donald J. Kessler, Nicholas L. Johnson , J.-C. Liou , and Mark Matney, The Kessler Syndrome: Implications to Future Space operations, 2010

[16] J.-C. Liou, N. L. Johnson and N. M. Hill, "Controlling the growth of future LEO debris populations with active debris removal," *Acta Astronautica,* vol. 66, pp. 648-653, 2010.

[17] C. P. Mark and S. Kamath, "Review of Active Space Debris Removal Methods," vol. 47, pp. 194-206, 2019.

[18] R. Biesbroek, L. Innocenti, A. Wolahan and S. M. Serrano, "e. Deorbit-ESA's active debris removal mission," in *Proceedings of the 7th European Conference on Space Debris*, 2017.

[19] Zeno Pavanello, Francesco Branza, combined control and navigation approach to the robotic capture of space vehicles, 72nd International Astronautical Congress (IAC), Dubai, United Arab Emirates, 25-29 October 2021.

[20] Nolet, S., Miller, D.W., June 2007. Development of a guidance, navigation and control architecture and validation process enabling autonomous docking to a tumbling satellite (Ph.D. thesis). Massachusetts Institute of Technology, Cambridge, MA, USA. <http://goo.gl/kRwGlk>.

[21] Siciliano, B., Sciavicco, L., Villani, L., and Oriolo, G. (2010). Robotics. London:Springer-Verlag.

[22] Stoneking, E. (2007). "Newton-euler dynamic equations of motion for a multibody spacecraft," in AIAA Guidance, Navigation and Control Conference and Exhibit, Guidance, Navigation, and Control and Co-located Conferences (Hilton Head, SC: AIAA), 1–13

[23] Wilde M, Kwok CS, Grompone A, e al. Equations of Motion of Free- Floating Spacecraft-Manipulator Systems: An Engineer's Tutorial. *Frontiers in Robotics and AI*. 2018.

[24] Flores-Abad, A., Ma, O, Pham, K., and Ulrich, S. (2014). A review of space robotics technologies for on-orbit servicing. Progr. Aerospace Sci. 68, 1–26. doi: 10.1016/j.paerosci.2014.03.002

[25] 72nd International Astronautical Congress (IAC), Dubai, United Arab Emirates, 25-29 October 2021

[26] JanKindracki, KarolinaTur, PrzemysławPaszkiewicz, ŁukaszM˛e˙zyk, ŁukaszBoruc, PiotrWola´nski, Experimental research on low-cost cold gas propulsion for a space robot platform, 2016

[27] Santanu Sarma*, A. K. Kulkarni, A. Venkateswaralu, P. Natarajan, and N.K. Malik,Spacecraft Dynamics Modeling and Simulation Using Matlab-Simulink, ISRO SATELLITE CENTRE, Vimanapura PO, Bangalore-17, India.

[28] B.T.C. Zandbergen, Modern Liquid Propellant Rocket Engines, 2000 Outlook, Internal Publication, Delft University of Technology, Faculty of Aerospace Engineering, Delft, The Netherlands, March 2000.

[29] Gilberto Arantes Jr., Luiz S. Martins-Filho, and Adrielle C. Santana, Optimal On - Off Attitude Control for the Brazilian Multi-mission Platform Satellite, 2009

[30] Gilberto Arantes Jr. and Luiz S. Martins-Filho, Guidance and control of position and attitude for Rendevouzs and dock/berthing with a Noncooperative/target spacecraft

[31] R. Bevilacqua, T.Lehmann, M.Romano Development and experimentation of LQR/APF guidance and control for autonomous proximity maneuvers of multiple spacecraft

[32] Leone Guarnaccia, Riccardo Bevilacqua, Stefano P. Pastorelli Suboptimal LQR-based spacecraft full motion control: Theory and experimentation

[33] Riccardo Bevilacqua, Marcello Romano, Fabio Curti, Andrew P Caprari, and Veronica Pellegrini. Guidance navigation and control for autonomous multiple spacecraft assembly: analysis and experimentation. International Journal of Aerospace Engineering, 2011, 2011.

[34] Matteo Dentis, Development of Innovative GNC Algorithms for Aerospace Applications

[35] Seddaou A, Saaj C. *Hα Controller for a Free-Flying Robotic Spacecraft*. Proc 14th, International Symposium on Artificial Intelligence, Robotics and Automation in Space (i-SAIRAS 2018): Madrid, Spain; 2018.

[36] Design and implementation of pseudo-inverse thrust allocation algorithm for ship dynamic positioning Baoyu Ye, Jianbin Xiong, 2019.

[37] Wie, B. (1998) space vehicle dynamics and control, AIAA Education series, AIAA.

[38] Sidi, M. J. (1997) spacecraft dynamics and control- A practical engineering approach Cambridge university Press.

[39] Krovel, Optimal tuning of PWPFM for attitude control.

[40] Stefano Oliva, Initial analysis of the reaction wheels disturbances impact on the attitude control system of a small satellite, Master's thesis

[41] Josep Virgili-Llop, C. Zagaris, R. Zappullam, A. Bradstree and Marcello Romano A convex-programming-based guidance algorithm to capture a tumbling object on orbit using a spacecraft equipped with a robotic manipulator 2018.

[42] Nicholas Ortolano, David Geller, Aaron Avery, Autonomous optimal trajectory planning for orbital rendezvous, satellite inspection, and final approach based on convex optimization., 2021.

[43] A tutorial on model predictive control for spacecraft rendezvous, Edward Hartley.

[44] Christpher Jewison, R. S. Erwin, Alvar saenz-Otero, Model predictive control with ellipsoid obstacle constraints for spacecraft rendezvous , 2015.

[45] Hyeongjun Park, Stefano Di Caiarano and Ilya kolmanovsky, Model predictive control for spacecraft Rendezvous ad docking with a rotating/Tumbling platform and for debris avoidance.

[46] Yoshida, K., and Umetani, Y. (1993). "Control of space manipulators with generalized Jacobian," in Space Robotics: Dynamics and Control, eds Y. Xu and T. Kanade (Boston, MA: Springer), 165–204.

[47] Umetani, Y., and Yoshida, K. (1989). Resolved motion rate control of space manipulators with generalized Jacobian Matrix. IEEE Trans. Robotics Autom. 5, 303–314.

[48] Baksalary, J. K., and Styan, G. P. H. (2002). Generalized inverses of partitioned matrices in Banachiewicz-Schur form. Linear Algeb. Appl. 354, 41–47. doi: 10.1016/S0024-3795(02)00334-8

[49] Papadopoulos, E., and Dubowsky, S. Dynamic singularities in free-floating space manipulators. *Journal of Dynamic Systems, Measurement, and Control*

[50] Papadopoulos, E. G. Path planning for space manipulators exhibiting nonholonomic behavior. In *Proceedings of the 1992 IEEE/RSJ International Conference on Intelligent Robots and Systems*.

[51] Papadopoulos, E., and Dubowsky, S. On the nature of control algorithms for free-floating space manipulators. *IEEE Transactions on Robotics and Automation*

[52] Nanos, K., and Papadopoulos, E. On Cartesian motions with singularities avoidance for free-floating space robots. In *2012 IEEE International Conference on Robotics and Automation*, St. Paul, MN, May 2012, 5398–5403.

[53] Papadopoulos On dynamics and control of space manipulators, Doctor of philosophy, 1990

[54] Yang, Y. (2010). "Quaternion based model for momentum biased nadir pointing spacecraft." Aerosp. Sci. Technol., 14(3), 199–202.

[55] Yang 2012, Analytic LQR Design for Spacecraft Control System Based on Quaternion Model

[56] Yamada, K., Yoshikawa, S., and Fujita, Y. Arm path planning of a space robot with angular momentum. *Advanced Robotics*, **9**, 6 (1994), 693–709.

[57] Nenchev, D., Umetani, Y., and Yoshida, K. Analysis of a redundant free-flying spacecraft/manipulator system. *IEEE Transactions on Robotics and Automation*, **8**, 1 (Feb. 1992), 1–6.

[58] Kostas Nanos, E. Papadopoulos, Avoiding dynamic singularities in cartesian motions of free-floating manipulators, Athens, Greece, 2015

[59] Srivastava Raunak, Roshan Sah and Das K Free Floating and rotation floating approaches for control of space robots: A comparative study, 2022.

[60] Nanos, K., and Papadopoulos, E. (2019). On Parameter Estimation of Space Manipulator Systems with Flexible Joints Using the Energy Balance. In IEEE International Conference on Robotics and Automation. Montreal, Canada. doi:10.1109/icra.2019.8793960

[61] Nanos, K., and Papadopoulos, E. G. (2015b). On the Dynamics and Control of Flexible Joint Space Manipulators. Control. Eng. Pract. 45, 230–243. doi:10.1016/j.conengprac.2015.06.009

[62] Pierpaolo Palmieri, Matteo Melchiorre and Stefano Mauro Design of a lightweight and deployable soft robotic arm, 2022.

[63] Pierpaolo Palmieri, Matteo Melchiorre, Stefano Mauro, Mario Troise and Matteo Gaidano, inflatable robotic manipulator for space debris mitigation by visual servoing, 2023.

[64] Fabio Curti, Marcello Romano and Riccardo Bevilacqua 2010, Lyapunov-based thruster selection for spacecraft control: Analysis and experimentation.

[65] Federico Basana, Francesco Branz, 2022 Simulation of robotic space operations with minimum base reaction manipulator.

[66] Lingling Shi, E Yao, Minghie Shan, Qinbin Gao and Xin Jin, 2022 Robust control of a space robot based on an optimized adaptive variable structure control method.

[67] Book, J.W.(1993).Structural flexibility of motion systems in the space environment. IEEE Transactionson Robotics and Automation.

[68] Martin, E.,Papadopoulos,E.,&Angeles,J.,(1999).A novel approach to reduce dynamic interactions in space robots. I1999 IASTED Interna.

[69] Zhang, Q., Liu, X., and Cai, G. (2020). Dynamics and Control of a Flexible-Link Flexible-Joint Space Robot with Joint Friction. Int. J. Aeronaut. Space Sci.

[70] K. Wormnes, R. Le Letty, L. Summerer, R. Schonenborg, O. Dubois-Matra, E. Luraschi, A Copp, H. Krag and J. Delaval, ESA TECHNOLOGIES FOR SPACE DEBRIS REMEDITION.

[71] Minghe Shan, Jian Guo, Eberhard Gill Review and comparison of active space debris capturing and removal methods, 2016.

[72] T. Kasai, M. Oda, T. Suzuki, Results of the ets-7 mission-rendezvous docking and space robotics experiments, in: Artificial Intelligence, Robotics and Automation in Space, vol. 440, 1999, p. 299.

[73] A. Kauderer, Nasa—canadarm2 and The Mobile Servicing System, Internet: http://www.nasa.gov/missionpages/station/structure/elements/mss.html⟩.

[74]] D.A. Whelan, E.A. Adler, S.B. Wilson III, G.M. Roesler Jr., Darpa orbital express program: effecting a revolution in space-based systems, in: International Symposium on Optical Science and Technology, International Society for Optics and Photonics, San Diego, CA, USA, 2000, pp. 48–56.

[75] K. Yoshida, D. Dimitrov, H. Nakanishi, On the capture of tumbling satellite by a space robot, in: 2006 IEEE/RSJ International Conference on Intelligent Robots and Systems, IEEE, Beijing, China, 2006, pp. 4127–4132.

[76] E. Papadopoulos, I. Paraskevas, Design and configuration control of space robots undergoing impact, in: 6th International ESA Conference on Guidance, Navigation and Control Systems, Loutraki, Greece, 2005.

[77] Bruno Victorino Sarli · André Luís da Silva · Pedro Paglione, sliding mode attitude control using thrusters and pulse modulation for the ASTER mission, 2013.

[78] Aghili, F. (2008). Optimal Control for Robotic Capturing and Passivation of a Tumbling Satellite with Unknown Dynamics AIAA Guidance, Navigation and Control Conference. Honolulu, Hawaii. doi:10.2514/6.2008-7274

[79] Wang, D., Huang, P., and Meng, Z. (2015). Coordinated Stabilization of Tumbling Targets Using Tethered Space Manipulators. IEEE Trans. Aerosp. Electron. Syst. 51 (3), 2420–2432. doi:10.1109/taes.2015.140530

[80] Aghili, F. (2012). A Prediction and Motion-Planning Scheme for Visually Guided Robotic Capturing of Free-Floating Tumbling Objects with Uncertain Dynamics. IEEE Trans. Robotics 28 (3), 634–649. doi:10.1109/tro.2011.2179581.

[81] Design and development of guidance navigation and control algorithms for spacecraft rendezvous and docking experimentation Giorgio Guglieri, Franco Maroglio , Pasquale Pellegrino, Liliana Torre

[82] Zhou, Z., and Colgren, R. (2005). "A non-linear spacecraft attitude tracking controller for large non-constant rate commands." Int. J. Control, 78(5), 311–325

[84] M.J. Sidi, Spacecraft Dynamics and Control: A Practical Engineering Approach, Cambridge University Press, 1997

[85] E. Papadopoulos On the nature of control algorithms for Free-Floating space manipulators, 1991, IEE transaction on robotics and automation, VOL.7, NO. 6.

[86] Kazuya Yoshida and Yuji Umetani Control of space free-flying robot, 1990.

[87] Yoshihiko Nakamura and Ranjan Mukherjee Nonholonomic path planning of space robots, centre for robotics system in Microelectronics, University of California, 1989 IEEE.

[88] Sofiane Kraiem, Mathieu Rognant, Jean-Marc Biannic, Yves Briere Robust control of rotation floating space robots with flexible apendges for on-orbit servicing, IFAC Papers Online 54-20 (2021) 134–140.

[89] A. Ellery, An engineering approach to the dynamic control of space robotic on-orbit servicers, Surrey Space Centre, University of Surrey, Guildford, Surrey, UK, 2004.

[90] Z. Vafa and S.Dubowsky On the Dynamics of Space Manipulators Using the Virtual Manipulator, with Applications to Path Planning, space robotics : dynamic and control.

[91] Y. Yang Spacecraft attitude determination and control: Quaternion based method, 2012, Annual Reviews in Control 36 (2012) 198–219.

[92] Spong MW (1987) Modelling and control of elastic joint robots. Dyn Syst Meas Control Trans ASME 109(4):310–319. https :// doi.org/10.1115/1.31438 60

[93] Wee, L., Walker, M.W.: On the dynamics of contact between space robots and configuration control for impact minimization. IEEE Trans. Robot. Autom. **9**(5), 581–591 (1993).

[94] Yoshida, K., Nenchev, D.N.: Space robot impact analysis and satellite-base impulse minimization using reaction nullspace. In: IEEE International Conference on Robotics and Automation, pp. 1271–1277. Nagoya (1995).

[95] Papadopoulos, E., Paraskevas, I.: Design and configuration control of space robots undergoing impacts. In: 6th International ESA Conference on Guidance, Navigation and Control Systems, pp. 17–20. Loutraki (2005).

[96] Huang, P., Yuan, J., Xu, Y., Liu, R.: Approach trajectory planning of space robot for impact minimization. In: IEEE International Conference on Information Acquisition, pp. 382–387. Weihai (2006).

[97] Papadopoulos, E., Abu-Abed, A.: Design and motion planning for a zero-reaction manipulator. In: IEEE International Conference on Robotics and Automation, pp. 1554–1559. San Diego (1994).

[98] Oki, T., Nakanishi, H., Yoshida, K.: Time-optimal manipulator control of a free-floating space robot with constraint on reaction torque. In: IEEE International Conference on Intelligent Robots and Systems, pp. 2828–2833. Nice (2008).

[99] S.-I. Nishida, S.Kawamoto,Strategy for capturing of a tumbling space debris, Acta Astronaut. 68(1) (2011)113–120.

[100] ESA https://www.esa.int/Space_Safety/Space_Debris/Active_debris_removol.

[101] Matteo Dentis, Elisa Capello and Giorgio Guglieri, A Novel Concept for guidance and control of spacecraft orbital maneuvers, 2016.

[102] Pierpaolo Palmieri, Mario Troise, Laura Salamina, Matteo Gaiadano, Matteo Melchiorre and Stefano Mauro, An inflatable 7-DOF space robotic arm for active debris removal, 2023, Politecnico di Torino.

List of figures

List of tables