POLITECNICO
MILANO 1863

SCUOLA DI INGEGNERIA INDUSTRIALE
E DELL'INFORMAZIONE

Executive Summary of the Thesis

Near Optimal Incremental Sampling-Based Path Planning for Free-Flying Space Manipulator

Laurea Magistrale in Space Engineering - Ingegneria Spaziale

Author: Andrea Allevi

Advisor: Prof. Mauro Massari

Academic year: 2021-2022

1. Introduction

The growing demand of satellites and space structures for terrestrial applications and both commercial and scientific purposes is prompting the way towards mission for on-orbit servicing, on-orbit assembly and active debris removal to maintain a cost-effective and sustainable space exploitation. Space robots are considered one of the most promising technologies to attend these tasks. A *space robot* or a *space manipulator system* consists of a six degrees of freedom (DoF) spacecraft base equipped with a N degrees of freedom (DoF) robotic manipulator with a grappling device, called *end-effector*, which allow it to capture a *target* orbiting object [5]. Despite the potential of space robots, their use is still very limited due to the complexity involved in this kind of missions, which require advanced algorithms to go through very different and demanding phases [2]. One of these phases consists in defining low-level instructions of how to move the system and successfully capture the target, avoiding obstacles and satisfying mission constraints: this problem is known as *path planning*. The aim of this work is proposing a flexible strategy and a new algorithm based on incremental exploration via sampling of the configuration space to plan the trajectory of the system for capturing a tumbling, uncooperative orbiting target.

2. Problem Formulation

The goal of the path planning problem proposed in this work is to find a *feasible* trajectory for a free-flying space manipulator system system, fully described by the chaser base homogeneous transform matrix $\boldsymbol{T}_{\mathrm{Cb}}^{\mathrm{LVLH}}(t)$ and the robotic arm joint variables $\boldsymbol{\theta}(t)$, driving the system from its initial conditions $\boldsymbol{T}_{\mathrm{Cb}}^{\mathrm{LVLH}}(t_0)$, $\boldsymbol{\theta}(t_0)$ to a *successful* capture of a tumbling target body. For a *successful* capture in a specific time instant t_f, the end-effector of the robotic arm must match the position and the orientation of the grasping point reference frame on the target with zero relative linear and angular velocities. In particular following conditions have to be satisfied:

$$\boldsymbol{r}_{\mathrm{EE}}(t_f) = \boldsymbol{r}_{gr}(t_f) \tag{1}$$
$$\boldsymbol{q}_{\mathrm{EE}}(t_f) = \boldsymbol{q}_{gr}(t_f) \tag{2}$$
$$\dot{\boldsymbol{r}}_{\mathrm{EE}}(t_f) = \dot{\boldsymbol{r}}_{gr}(t_f) \tag{3}$$
$$\boldsymbol{\omega}_{\mathrm{EE}}(t_f) = \boldsymbol{\omega}_{Tb}(t_f) \tag{4}$$

where $\boldsymbol{r}$ represents position vector, $\boldsymbol{q}$ denotes the unit quaternion vector to represent reference frame orientation and $\dot{\boldsymbol{r}}$, $\dot{\boldsymbol{\omega}}$ are respectively the linear and angular velocities. The subscript

"EE", "gr", "Cb" and "Tb" refer respectively to the reference frames of the end-effector, the grasping point, the chaser base body and the target body
To be *feasible* the trajectory have to satisfy several *hard constraints*: the trajectory must be such that the chaser approaches the target spacecraft in a safe way, avoiding any collision with the target or itself and it must ensure that the target body is kept within the Field of View (FoV) of sensors mounted on the chaser spacecraft at any time instant for navigation purposes. The planning of the trajectory must also take into account several operational constraints due to mechanisms or actuators limits. In particular, the constraint imposed by thrusters limits is:

$$|u_{x,y,z}| \leq u_{\text{MAX}} \quad \text{with} \quad u_{\text{MAX}} = \frac{F_{\text{MAX}}}{m_c} \tag{5}$$

where $u_{x,y,z}$ represents the generic component of the control acceleration to maneuver the base of the space manipulator sytem,m_c represents the mass of the chaser spacecraft and u_{MAX} is the maximum available control acceleration, limited by the maximum thrust F_{MAX} available. The joints of the robotic manipulator have bounded displacements as well as bounded velocities and accelerations:

$$\theta_{i\,\min} \leq \theta_i(t) \leq \theta_{i\,\text{MAX}} \quad \text{for} \quad i = 1, ..., N \tag{6}$$

$$\|\dot{\theta}_i(t)\| \leq \dot{\theta}_{i\,\text{MAX}} \quad \text{for} \quad i = 1, ..., N \tag{7}$$

$$\|\ddot{\theta}_i(t)\| \leq \ddot{\theta}_{i\,\text{MAX}} \quad \text{for} \quad i = 1, ..., N \tag{8}$$

where $\theta_{i\,\min}$ and $\theta_{i\,\text{MAX}}$ denote respectively the minimum and maximum angular displacement of i^{th} joint, $\dot{\theta}_{i\,\text{MAX}}$ represents the maximum absolute value of i^{th} joint speed and $\ddot{\theta}_{i\,\text{MAX}}$ represents the maximum absolute value of i^{th} joint acceleration.
Additional aspects have been taken into account to improve the quality of the trajectory, in particular:

- minimize *fuel consumption*;
- reduce the *duration* of the maneuver;
- avoid *kinematic singularities* that could affect in negative way the motion of the robotic manipulator [8].

The mission framework and the proposed solution to the problem is based on the following assumptions:

- the orbit of the target body is circular or near circular;
- the chaser is initially in proximity of the target body, at a relative distance d $\leq 100m$;
- both the chaser and target object are composed of rigid bodies;
- disturbances such as gravity gradient, solar radiation pressure and atmospheric drag are neglected;
- the state, the geometry and the inertia properties of the chaser and of the tumbling target are known;
- the mass of the chaser remains constant since the amount of propellant used is small when compared with the chaser mass,
- the chaser operates in *free-flying* mode, which means that the position and the orientation of the chaser base are actively controlled.

2.1. System Modeling

This section provides a brief overview of the mathematical models used in the current work. In particular are here reported the linear model to describe the relative translational motion in orbit, the model to describe the kinematics and the dynamics of an uncontrolled tumbling satellite and finally the kinematics model of the free-flying space manipulator system.

Translational Relative Motion Model

The relative motion in Local Vertical Local Horizon (LVLH) frame of a chaser spacecraft with respect to the Center of Mass (CoM) of a target body that is in a circular orbit around a central body can be described by Clohessy-Wiltshire (CW) equations (Eq. (9)):

$$\begin{cases} \ddot{x} - 3n_T^2 x - 2n_T \dot{y} = u_x \\ \ddot{y} + 2n_T \dot{x} = u_y \\ \ddot{z} + n_T^2 z = u_z \end{cases} \tag{9}$$

where x, y and z are the coordinates of the CoM of the chaser base in LVLH frame, n_T is the mean orbital motion of the target body and $\mathbf{u} = [u_x, u_y, u_z]^T$ denotes control accelerations. The set (9) of second-order differential equations can be converted in *state space form* as follow:

$$\dot{\mathbf{x}}_{\text{cw}} = f(\mathbf{x}_{\text{cw}}, \mathbf{u}, t) = \mathbf{A}_{\text{cw}}\mathbf{x}_{\text{cw}} + \mathbf{B}_{\text{cw}}\mathbf{u} \tag{10}$$

where $\mathbf{x}_{\text{cw}}(t) = [x, y, z, \dot{x}, \dot{y}, \dot{z}]^T$ represents the state vector and state matrix $\mathbf{A}_{\text{cw}}$ and state input matrix $\mathbf{B}_{\text{cw}}$ are given by Eqs. (11) and (12)

$$\mathbf{A}_{\text{cw}} = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 3n_T^2 & 0 & 0 & 0 & 2n_T & 0 \\ 0 & 0 & 0 & -2n_T & 0 & 0 \\ 0 & 0 & -n_T^2 & 0 & 0 & 0 \end{bmatrix} \tag{11}$$

$$\mathbf{B}_{\text{cw}} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{12}$$

Target Rotational Motion Model

The attitude of the target's body frame $\mathcal{B}_{Tb}$ with respect to LVLH frame is represented by using *unit quaternions* and the *attitude kinematics* equation is given by [4]:

$$\dot{\boldsymbol{q}}_{\text{Tb}} = \frac{1}{2}\,\boldsymbol{\Xi}(\boldsymbol{q}_{\text{Tb}})\,\boldsymbol{\omega}_{\text{Tb}}^{\text{Tb}} \tag{13}$$

where $\boldsymbol{q}_{\text{Tb}}$ represents the attitude of $\mathcal{B}_{Tb}$ with respect to LVLH frame, $\boldsymbol{\omega}_{\text{Tb}}^{\text{Tb}}$ represents the angular speed of the target's body frame with respect LVLH frame, expressed in $\mathcal{B}_{Tb}$ and matrix $\boldsymbol{\Xi}(\boldsymbol{q})$ is defined as follow [4]:

$$\boldsymbol{\Xi}(\boldsymbol{q}) = \begin{bmatrix} q_4 & -q_3 & q_2 \\ q_3 & q_4 & -q_1 \\ -q_2 & q_1 & q_4 \\ -q_1 & -q_2 & -q_3 \end{bmatrix} \tag{14}$$

The rotational dynamics of the uncontrolled target spacecraft is governed by Euler's equation that can be represented as [4]:

$$\mathbb{J}\dot{\boldsymbol{\omega}}_{\text{Tb}}^{\text{Tb}} = \mathbb{J}\boldsymbol{\omega}_{\text{Tb}}^{\text{Tb}} \times \boldsymbol{\omega}_{\text{Tb}}^{\text{Tb}} \tag{15}$$

where $\mathbb{J}$ denotes the inertia matrix of the target spacecraft expressed in $\mathcal{B}_{Tb}$. The system of equations composed by Euler's equation (15) and attitude kinematics equation (13) provide a complete description of the rotational motion of the target body.

Space Manipulator System Kinematics

A kinematics model of a free-flying space manipulator is needed to describe the motion of the system, in particular of the *end-effector*, in LVLH frame. The position and the orientation of the end-effector reference frame can be represented by the homogeneous transform matrix $\boldsymbol{T}_{\text{EE}}^{\text{LVLH}}$. It can be computed as [6, 7]:

$$\boldsymbol{T}_{\text{EE}}^{\text{LVLH}} = \boldsymbol{T}_{\text{Cb}}^{\text{LVLH}}\,\boldsymbol{T}_{\mathcal{J}_1}^{\text{Cb}}\left(\prod_{i=2}^{N+1}\boldsymbol{T}_{\mathcal{J}_i}^{\mathcal{J}_{i-1}}\right)\boldsymbol{T}_{\text{EE}}^{\mathcal{J}_{N+1}}$$

where $\boldsymbol{T}_{\text{Cb}}^{\text{LVLH}}(t)$ is the chaser base homogeneous transform matrix, $\boldsymbol{T}_{\text{Cb}}^{\text{LVLH}}$ and $\boldsymbol{T}_{\text{EE}}^{\mathcal{J}_{N+1}}$ are homogeneous transform matrices related to the geometry of ends of the arm and $\boldsymbol{T}_{\mathcal{J}_i}^{\mathcal{J}_{i-1}}$ is the homogeneous transformation matrix from joint frame $\mathcal{J}_{i+1}$ to joint frame $\mathcal{J}_i$ can be expressed as a function of Denavit-Hartenberg (DH) parameters as follow:

$$\boldsymbol{T}_{\mathcal{J}_{i+1}}^{\mathcal{J}_i} = \boldsymbol{A}(\theta_i, d_i, \alpha_i, c_i) \tag{16}$$

where the DH transformation matrix function $\boldsymbol{A}(\theta_i, d_i, \alpha_i, c_i)$ and customized DH parameters are defined in [7]. The relationship between the velocities of the end-effector and the velocities of the joint and of the chaser base is linear and it can be expressed as [7]:

$$\begin{bmatrix} \dot{\boldsymbol{r}}_{\text{EE}} \\ \boldsymbol{\omega}_{\text{EE}} \end{bmatrix} = \boldsymbol{J}\begin{bmatrix} \dot{\boldsymbol{X}}_{\text{Cb}} \\ \dot{\boldsymbol{\theta}} \end{bmatrix} \qquad \text{with} \quad \boldsymbol{J} = \begin{bmatrix} \boldsymbol{J}_{\text{Cb}} & \boldsymbol{J}_{\text{m}} \end{bmatrix}$$

where $\dot{\boldsymbol{r}}_{\text{EE}}$ $\boldsymbol{\omega}_{\text{EE}}$ are respectively the linear and the angular velocities expressed in LVLH frame, $\boldsymbol{J} \in \mathbb{R}^{6\times(6+N)}$ represents the Jacobian matrix of the overall free-flying space manipulator system, $\boldsymbol{J}_{\text{Cb}} \in \mathbb{R}^{6\times 6}$ Jacobian matrix of the chaser spacecraft base, $\boldsymbol{J}_{\text{m}} \in \mathbb{R}^{6\times N}$ is the robotic manipulator Jacobian, $\dot{\boldsymbol{X}}_{\text{Cb}} = [\dot{\boldsymbol{r}}_{\text{Cb}}^{\text{LVLH}}; \boldsymbol{\omega}_{\text{Cb}}^{\text{LVLH}}]$ and $\dot{\boldsymbol{\theta}}$ represent vectors containing the velocities of the chaser spacecraft base and the velocities of the joints respectively.

3. Path Planning Approach

To make the path planning problem described in Sec. 2 more tractable, the design of the trajectory has been divided in three parts, each one addressed to deal with different tasks and constraints:

- **Chaser Base CoM Path $\boldsymbol{r}_{\text{Cb}}(t)$**
  The chaser base CoM trajectory is the first to be designed. Its aim is to drive the space manipulator system from an initial position $\boldsymbol{r}_{\text{Cb}}(t_0)$ away from the target body in which the grasping point is outside the workspace

of the robotic manipulator, to a final desired safe position ${}^{d}\boldsymbol{r}$ near the target in which the grasping point is inside the workspace of the robotic arm and the capture can be performed. The trajectory must be such as to avoid collisions with the appendages of the target, minimize fuel consumption and reduce duration of the maneuver. For these reasons a modified version of the adaptive LQR/APF proposed in [1] has been used for the design of the chaser base CoM trajectory.
  - *adaptive Linear Quadratic Regulator*: it derives from optimal control theory of linear system, it design a trajectory aware of fuel consumption, adaptive control gain matrix allow to reduce the duration of the maneuver.
  - *Artificial Potential Field*: the space manipulator system moves under the action of a fictitious force that has opposite direction of the gradient of a potential function. If the potential function has a maximum value in correspondence of obstacles regions, the fictitious force repels the space robot from those regions.
- **Chaser Base Attitude Path $\boldsymbol{R}_{\mathrm{Cb}}(t)$**
  The attitude of the chaser base is designed from the knowledge of the trajectory of the chaser base CoM position $\boldsymbol{r}_{\mathrm{Cb}}$ imposing the constraint that navigation sensors have to point always toward the target. It is assumed that chaser sensors boresight axis coincides with x-axis of chaser's base body frame $\hat{\boldsymbol{x}}_{Cb}$ and *Line-Of-Sight* frame proposed in [3] has been implemented.
- **Joint Variables Path $\boldsymbol{\theta}(t)$**
  The trajectory of joint variables is obtained through an original path planning algorithm based on an optimal-guided, incremental exploration of configuration space through sampling. The strength of the algorithm is that it is able to compute the trajectory of the end-effector passing from joint angles regardless of how the trajectory of the base is obtained. The issues on the other hand are related to the need of an inverse kinematics algorithm to fully defined the joint trajectory and attain a successful capture.

3.1. Adaptive LQR/APF

The adaptive LQR/APF algorithm combines the the optimal control for linear system of the Linear Quadratic Regulator and the collision avoidance capabilities of the Artificial Potential Function algorithm [1]. The space manipulator system moves under the combined action of two control inputs:

$$\mathbf{u} = \mathbf{u}_{\mathrm{LQR}} + \mathbf{u}_{\mathrm{APF}} \tag{17}$$

where $\mathbf{u}$ is the total control acceleration, $\mathbf{u}_{\mathrm{LQR}}$ is the LQR component of control acceleration that drives the chaser towards the desired final state $\mathbf{x}_d$ and $\mathbf{u}_{\mathrm{APF}}$ is the APF component of the control acceleration that avoids that the chaser collides with the target. The two components are described in detail in the following sections. The LQR controllers are based on finding the gain matrix $\mathbf{K}_{\mathrm{LQR}}$, generating the optimal control acceleration $\mathbf{u}_{\mathrm{LQR}} = -\mathbf{K}_{\mathrm{LQR}}\mathbf{x}_{\mathrm{e}}$ that minimizes the following quadratic cost function $J_{\mathrm{LQR}}(\mathbf{Q}, \mathbf{R}, \mathbf{x}_e, \mathbf{u})$ according to the choice of weighting matrices $\mathbf{Q}$ and $\mathbf{R}$ [1]. The *state weighting matrix* $\mathbf{Q} \in \mathbb{R}^{6\times 6}$ and the *control weighting matrix* $\mathbf{R} \in \mathbb{R}^{3\times 3}$ are updated at constant time intervals Δt_{LQR} along the trajectory as a function of the distance $\mathrm{d}(t)$ between the actual position of the chaser base CoM $\boldsymbol{r}_{\mathrm{Cb}}(t)$ and its desired final position ${}^{d}\boldsymbol{r}_{\mathrm{Cb}}$; the gain matrix $\mathbf{K}_{\mathrm{LQR}}$ is updated accordingly. In this work, the weighting matrices $\mathbf{Q}$ and $\mathbf{R}$ are assumed diagonal matrices and their terms Q_{jj} and R_{jj} are respectively defined as:

$$\mathrm{Q}_{jj} = \alpha \qquad j = 1, ..., 6 \tag{18}$$

$$\mathrm{R}_{jj} = \frac{\beta}{u_{\mathrm{MAX}}^2} \qquad j = 1, ..., 3 \tag{19}$$

where the coefficients α and β are:

$$\alpha(\mathrm{d}) = \frac{1 + \ln\left(\frac{\mathrm{d}_0}{\mathrm{d}}\right)}{\mathrm{d}_0} \tag{20}$$

$$\beta(\mathrm{d}) = \left[1 + \exp\left(-\frac{\mathrm{d}}{\mathrm{d}_0}\right)\right] \mathrm{d} \tag{21}$$

The weights Q_{jj} increase as the chaser approaches the desired final position while the control weights R_{jj} have opposite behaviour since they decrease as the chaser approaches final desired position.

The APF component of the LQR/APF control is used to provide collision avoidance capabilities to the chaser when approaching the target. The control action $\mathbf{u}_{\text{APF}}$ has opposite direction of the gradient of the potential function $U(\boldsymbol{r})$, where $\boldsymbol{r} = [x, y, z]^T$ denotes a generic position of the chaser CoM in LVLH frame. The function is defined as:

$$U(\boldsymbol{r}) = k_p \left(\|\boldsymbol{r}\| - \boldsymbol{r} \cdot {}^d\hat{\boldsymbol{r}} \right) \exp\left(-\frac{1}{2\,k_h} |h(\boldsymbol{r})| \right)$$

where $\boldsymbol{r} = [x, y, z]^T$ denotes the generic position of the chaser base CoM in LVLH frame, ${}^d\hat{\boldsymbol{r}}$ is the unit length vector pointing from target CoM to the final desired position of the chaser base, identifying the direction of the final approaching corridor, $h(\boldsymbol{r})$ is a scalar function such as $h(\boldsymbol{r}) = 0$ denotes the equation of the surface of the keep-out zone around the target where the exponential part is a Gaussian function that creates a maximum value region in correspondence of the surface of the keep-out zone around the target defined by equation $h(\boldsymbol{r}) = 0$, and the first part minimizes the potential function U along the desired approaching direction ${}^d\hat{\boldsymbol{r}}$ making it equal to zero, creating a minimum value potential region to allow the chaser to get close to the target.

3.2. NOISBPP Algorithm

The proposed algorithm divides the time interval $[t_{start} : t_f]$ into a number of sub-intervals N_{int}, each one of duration Δt_{int}, such that the vector $\boldsymbol{t} \in \mathbb{R}^{N_{int}+1}$ results:

$$\boldsymbol{t} = [\, t_1 \,, ..., t_{k-1} \,, t_k \,, ..., t_{N_{int}+1} \,]^T \tag{22}$$

Then, at each time instant t_k from t_2 to $t_{N_{int}}$, the algorithm computes the trajectories of the joints by incrementally selecting joint angles waypoints $\boldsymbol{\theta}^{(k)}$ and joint velocities waypoints $\dot{\boldsymbol{\theta}}^{(k)}$. The selected waypoints $\boldsymbol{\theta}^{(k)}$ and $\dot{\boldsymbol{\theta}}^{(k)}$ at instant t_k are those which, among all the possible solutions that are sampled, minimize a cost functional that takes into account multiple objectives. Each waypoint is then connected with the next one by a polynomial trajectory.
The proposed algorithm is based on the repetition of four main steps at each time instant t_k, with k that goes from 2 to N_{int}:

1. *define sampling limits*: given the current waypoint defined by $\boldsymbol{\theta}_{\text{current}}^{(k-1)}$ and $\dot{\boldsymbol{\theta}}_{\text{current}}^{(k-1)}$, define the portions of joint configuration and joint velocity spaces where searching for a solution $\boldsymbol{\theta}^{(k)}$ and $\dot{\boldsymbol{\theta}}^{(k)}$ at time t_k. To find sampling limits, an accelerated and decelerated motion of the joints is assumed starting from *current* conditions throughout the interval Δt_{int}. The *current* condition denotes the last waypoint before the algorithm selects the new one;
2. *sample*: the defined portions of joint configuration and joint velocities space are explored through random sampling, generating N_S multiple possible solutions;
3. *evaluate the cost of each sample and select the cost minimizing one*: each sample is evaluated by multiple cost functions and a total cost is then assigned to it. The sample with the lowest total cost is selected as a waypoint at instant t_k. The minimization of the cost functions guides the end-effector to the grasping point in LVLH frame, imposes constraints on joint angles, velocities and accelerations and avoid that kinematic singularities occur during motion.

Algorithm 1 NOISBPP algorithm

INPUT: t_{start}, t_f, Δt_{int}, N_S, $\boldsymbol{T}_{\text{Cb}}(t)$, $\dot{\boldsymbol{X}}_{\text{Cb}}(t)$, $\boldsymbol{T}_{gr}(t)$, $\dot{\boldsymbol{X}}_{gr}(t)$, $\boldsymbol{\theta}(t_0)$, $\dot{\boldsymbol{\theta}}(t_0)$
define $\boldsymbol{t} = [t_1, ..., t_{k-1}, t_k, t_{k+1}, ..., t_{N_{int}+1}]$
initialize $\boldsymbol{\theta}_{\text{current}} = \boldsymbol{\theta}(t_0)$; $\dot{\boldsymbol{\theta}}_{\text{current}} = \dot{\boldsymbol{\theta}}(t_0)$
for each instant t_k, with $k = 2, ..., N_{int}$ **do**
  define sample limits $\leftarrow \{\boldsymbol{\theta}_{\text{current}}, \dot{\boldsymbol{\theta}}_{\text{current}}\}$
  sample: $\boldsymbol{\theta}_s^{(k)}$; $\dot{\boldsymbol{\theta}}_s^{(k)}$, with $s = 1, ..., N_S$
  for each sample $\boldsymbol{\theta}_s^{(k)}$, $s = 1, ..., N_S$ **do**
    for each cost function $f_{j\theta}$ **do**
      evaluate cost of $\boldsymbol{\theta}_s^{(k)}$
    end for
    compute total cost c_s^{tot} of sample $\boldsymbol{\theta}_s^{(k)}$
  end for
  select min. total cost configuration: $\boldsymbol{\theta}^{(k)}$
  for each sample $\dot{\boldsymbol{\theta}}_s^{(k)}$, $s = 1, ..., N_S$ **do**
    for each cost function $f_{j\dot{\theta}}$ **do**
      evaluate cost of $\dot{\boldsymbol{\theta}}_s^{(k)}$
    end for
    evaluate cost of the trajectory between $[\boldsymbol{\theta}_{\text{current}}, \dot{\boldsymbol{\theta}}_{\text{current}}]$ and $[\boldsymbol{\theta}^{(k)}, \dot{\boldsymbol{\theta}}_s^{(k)}]$
    compute total cost c_s^{tot} of sample $\dot{\boldsymbol{\theta}}_s^{(k)}$
  end for
  take min.total cost velocities vector $\dot{\boldsymbol{\theta}}^{(k)}$
  update $\boldsymbol{\theta}_{\text{current}} = \boldsymbol{\theta}^{(k)}$; $\dot{\boldsymbol{\theta}}_{\text{current}} = \dot{\boldsymbol{\theta}}^{(k)}$
end for

4. *update*: the selected waypoint becomes the new current waypoint and steps from 1 to 4 are repeated moving forward in time.

The last part of the trajectory, between instant $t_{N_{int}}$ and capture instant $t_{N_{int}+1}$ is computed by solving the inverse kinematics problem.

4. Simulation Results

In figure 1 is shown the trajectory of the end-effector planned by the proposed algorithm.

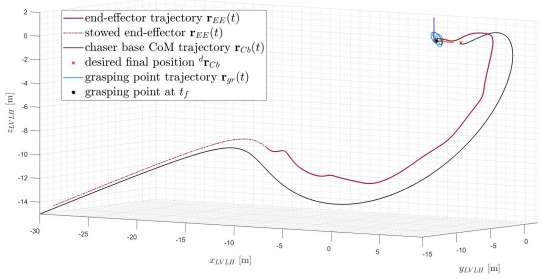

Figure 1: End-effector trajectory in LVLH frame

The proposed NOISBPP algorithm does not produce always a feasible solutions. Two kinds of errors are encountered: one classified as *serious* errors the other classified as *minor* errors. The algorithm has been simulated with different parameters in terms of interval between two sampling stations Δt_{int} and in terms of number of samples N_S. For a low number of samples N_S, an intermediate interval returns feasible results with higher probability: by its derivation, the sampling interval can be considered proportional to the time interval Δt_{int}. Then, a low number of samples can be enough to efficiently explore a not too wide sampling interval. However, there seem to be no advantages in a smaller sampling interval (correspondent to the shorter Δt_{int} considered) that can be better explored with the same number of samples. A smaller sampling interval does not allow to explore the entire configuration space and the arm remains stuck in a non-optimal configuration to perform the capture. This behaviour is in contrast with a solution obtained with a longer time interval Δt_{int} that allows a better exploration of the whole configuration space and wider movements to reach an optimal configuration for the capture.

5. Conclusions

This thesis work proposes a flexible strategy to plan the motion of a free-flying space manipulator system and capture a target body.The proposed strategy consists in a reworking of methods already proposed in literature such as the adaptive LQR/APF algorithm and the Line-Of-Sight reference frames and in a new incremental sampling-based algorithm for planning the trajectory of a robotic manipulator on a free-flying base in the configuration space. This work has shown, through numerical simulations, the validity of the strategy and of the new algorithm, that, under some hypothesis on the motion of the target body and on the operative modes of the space robot, are able to deal with common constraints and aspects of an on-orbit servicing mission with a robotic manipulator, such as collision avoidance, limited thrust, bounded joint displacements and kinematic-singularities avoidance.The performances of the algorithm has been discussed as well as its failures. Future developments would aim to solve failures and reduce the number of hypothesis on which the work is based.

References

[1] R Bevilacqua, T Lehmann, and Marcello Romano. Development and experimentation of lqr/apf guidance and control for autonomous proximity maneuvers of multiple spacecraft. *Acta Astronautica*, 68(7-8):1260–1275, 2011.

[2] Angel Flores-Abad, Ou Ma, Khanh Pham, and Steve Ulrich. A review of space robotics technologies for on-orbit servicing. *Progress in aerospace sciences*, 68:1–26, 2014.

[3] Qinglei Hu, Yueyang Liu, and Youmin Zhang. Control of non-cooperative spacecraft in final phase proximity operations under input constraints. *Control Engineering Practice*, 87:83–96, 2019.

[4] F Landis Markley and John L Crassidis. *Fundamentals of spacecraft attitude determination and control*. Springer, 2014.

[5] S Ali A Moosavian and Evangelos Papadopoulos. Free-flying robots in space: an overview of dynamics modeling, planning and control. *Robotica*, 25(5):537–547, 2007.

[6] B. Siciliano, L. Sciavicco, L. Villani, and G. Oriolo. *Robotics: Modelling, Planning and Control*. Advanced Textbooks in Control and Signal Processing. Springer London, 2009.

[7] Markus Wilde, Stephen Kwok Choon, Alessio Grompone, and Marcello Romano. Equations of motion of free-floating spacecraft-manipulator systems: an engineer's tutorial. *Frontiers in Robotics and AI*, 5:41, 2018.

[8] Tsuneo Yoshikawa. Manipulability and redundancy control of robotic mechanisms. *Proceedings. 1985 IEEE International Conference on Robotics and Automation*, 2:1004–1009, 1985.