2024 IEEE 63rd Conference on Decision and Control (CDC)
December 16-19, 2024. MiCo, Milan, Italy

MPC for Tracking applied to rendezvous with non-cooperative tumbling targets ensuring stability and feasibility

Jose Antonio Rebollo, Rafael Vazquez, Ignacio Alvarado and Daniel Limon

Abstract— **A Model Predictive Controller for Tracking is introduced for rendezvous with non-cooperative tumbling targets in active debris removal applications. The target's three-dimensional non-periodic rotational dynamics as well as other state and control constraints are considered. The approach is based on applying an intermediate coordinate transformation that eliminates the time-dependency due to rotations in the constraints. The control law is then found as the solution to a Quadratic Programming problem with linear constraints and dynamics, as derived from the Hill-Clohesy-Wiltshire equations, that provides feasibility and stability guarantees by means of a terminal Linear Quadratic Regulator and dead-beat region. The proposed control algorithm performs well in a realistic simulation scenario, namely a near rendezvous with the Envisat spacecraft.**

I. Introduction

Space debris presents a growing threat to operational spacecraft, with increased satellite deployments raising the risk of in-orbit collisions, underscoring the need for Active Debris Removal (ADR) strategies. ADR aims to remove defunct objects, reducing collision risks and debris population [1].

Various rendezvous-based ADR technologies have been proposed [2], involving mechanical attachment to deploy deorbiting devices like Electrodynamic Tethers (ET) [3], [4] or solid propellant kits [5]. A key challenge is safely and precisely rendezvousing with non-cooperative, often tumbling targets [6].

Model Predictive Control (MPC) is well-suited for complex, safety-critical missions like ADR due to its ability to handle constraints and optimize control actions in real-time. While previous research focuses on 3-DOF rendezvous with tumbling targets, time-varying constraints complicate finding and maintaining feasible solutions, often requiring longer prediction horizons or optimal trajectories, which increases computational complexity. This "feasibility problem" affects the current scenario. For instance, [7] solved time-constrained rendezvous but didn't explicitly address collision avoidance, while [8] assumed attitude synchronization and used quadratic programs but limited collision avoidance to keeping the chaser outside a sphere around the target. More sophisticated strategies like [9] divide docking into phases for improved safety.

MPC has shown robustness under uncertainties, especially using tube-based approaches [10]. For instance, [11] applied Tube-Based MPC for rendezvous without considering tumbling targets, while [12], [13] used Tube-Based MPC with robust control for tumbling targets, relying on motion planners for reference trajectories. Traditional controllers focus on fixed equilibria [14], but dynamic targets complicate control design. Recent controllers ensure stability for moving targets [15]. In [16], a multistage MPC for Tracking (MPCT) ensured safe, stable 3-DOF rendezvous with tumbling targets.

This paper is organized as follows: Section II outlines the problem and constraints, Section III introduces the control strategy, applied in Section IV. Section V evaluates the controller's implementation, and Section VI presents simulation results for the Envisat scenario. Finally, concluding remarks are in Section VII. For further details, see the extended version of this paper [17].

II. Problem formulation

Let D be the target in the rendezvous operation, assumed to follow a circular geocentric orbit of radius R with an angular rate $n = \sqrt{\frac{\mu}{R^3}}$, where $\mu = 398600.4\,\mathrm{km}^3\mathrm{s}^{-2}$. The corresponding orbital period is $T_D = 2\pi/n$. The chaser spacecraft, C, must rendezvous with D, respecting the engineering constraints defined later.

The chaser's relative position and velocity to the target are described by vectors $\boldsymbol{r}$ and $\boldsymbol{v}$, expressed in the Local Vertical Local Horizontal (LVLH) frame, L. The target's position and velocity are given by $\boldsymbol{r}_D$ and $\boldsymbol{v}_D$. The relative motion of C and D is governed by the Hill-Clohessy-Wiltshire (HCW) equations [18], [19], where for an impulsive control $\boldsymbol{u}$ and propagation time T, the linearized evolution from $t_1 = kT$ to $t_2 = (k+1)T$ is

$$\begin{pmatrix} \boldsymbol{r}^L(k+1) \\ \boldsymbol{v}^L(k+1) \end{pmatrix} = A_L \begin{pmatrix} \boldsymbol{r}^L(k) \\ \boldsymbol{v}^L(k) \end{pmatrix} + B_L \boldsymbol{u}^L(k), \qquad (1)$$

where the components of A_L are affine functions of $s = \sin nT$, $c = \cos nT$ (see [17]), and for $B_L = \begin{bmatrix} 0_{3\times 3} & I_3 \end{bmatrix}$.

The target is modeled as a rigid body that rotates freely around its center of mass. In particular, let ${}^B_L C(k)$ be the rotation matrix from L coordinates to the target's body axes B at time kT. Given two vectors $\boldsymbol{a}, \boldsymbol{b} \in \mathbb{R}^3$, the cross product matrix operator is defined such as $\boldsymbol{a}^\times \boldsymbol{b} = \boldsymbol{a} \times \boldsymbol{b}$.

For a given initial attitude and angular velocity of the target ${}^B_L\boldsymbol{\omega}^B$, the evolution of this matrix is written as the solution of the Euler-Poinsot differential equations, depending on the target's inertia tensor I_D^B in body axes. In particular, if $n \ll \left\| {}^B_L\boldsymbol{\omega}^B \right\|$, the rotational kinematics can be simplified to

$${}^B_L\dot{\boldsymbol{\omega}}^B = -\left(I_D^B\right)^{-1} {}^B_L\boldsymbol{\omega}^B \times I_D^B \, {}^B_L\boldsymbol{\omega}^B \qquad (2)$$

$${}^B_L\dot{C} = -\left({}^B_L\boldsymbol{\omega}^B\right)^\times {}^B_L C. \qquad (3)$$

The attitude of the chaser, in the other hand, is assumed to match the one required to perform all maneuvers.

J.A. Rebollo and R. Vazquez are with the Department of Aerospace Engineering, Universidad de Sevilla, Camino de los Descubrimiento s.n., 41092 Sevilla, Spain.

I. Alvarado and D. Limon are with the Department of Systems Engineering and Automation, Universidad de Sevilla, Camino de los Descubrimiento s.n., 41092 Sevilla, Spain.

979-8-3503-1632-2/24/$31.00 ©2024 IEEE

The chaser's attitude is assumed to be aligned with the required orientation for docking, so its angular dynamics are not considered. To ensure collision avoidance, the chaser must stay within a safety region defined by the target's geometry. The Line of Sight (LOS) prism, attached to the target's rendezvous point, defines a convex and linear constraint in the body frame B,

$$\begin{bmatrix} 0 & -1 & 0 \\ c_x & -1 & 0 \\ -c_x & -1 & 0 \\ 0 & -1 & c_z \\ 0 & -1 & -c_z \end{bmatrix} \boldsymbol{r}^B \leq \begin{bmatrix} 0 \\ c_x x_0 \\ c_x x_0 \\ c_z z_0 \\ c_z z_0 \end{bmatrix}. \tag{4}$$

A diagram of the LVLH and body frames, along with the LOS prism, is shown in Fig. 1.

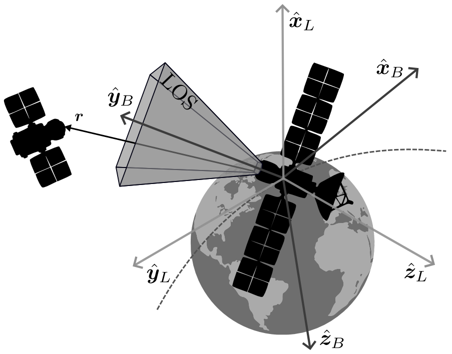

Fig. 1: Rendezvous operation considering debris body (B) and LVLH (L) reference frames.

Control inputs in the chaser's body frame are bounded as $u_{min} \leq u_i^B \leq u_{max}, i = 1, 2, 3$. Additionally, constraints on state and control inputs are compactly written as

$$A_x \left(\left(\boldsymbol{r}^B\right)^T \quad \left(\boldsymbol{v}^L\right)^T \right)^T \leq b_x, \quad A_u \boldsymbol{u}^B \leq b_u, \tag{5}$$

and fuel efficiency is modeled by minimizing

$$\sum_{k=1}^{N} \Delta v(k)^2 = \sum_{k=1}^{N} \|\boldsymbol{u}(k)\|^2. \tag{6}$$

III. MPCT-BASED RENDEZVOUS

In general, the safe rendezvous operation can be written as a constrained optimization problem, the structure of which is given by the considered control framework, and has an effect on the properties and guarantees of the controller. As discussed in Section V, the approach considered herein, which based on a novel extension of MPCT [16] for rotation-based LTV systems, leads to a promising results in terms of computational cost, optimality and feasibility. In particular, this strategy leads to the QP problem in (8)–(17), which is solved for each iteration k,

$$\begin{aligned} V(k, x_k, U(k), \bar{\theta}(k)) = & \\ & \sum_{i=0}^{N_p-1} \|x(k+i|k) - x_s(k+i|k)\|_Q^2 \\ & + \sum_{i=0}^{N_p} \|u(k+i|k) - u_s(k+i|k)\|_R^2 \\ & + \left\|\bar{\theta}(k)\right\|_T^2 \end{aligned} \tag{7}$$

$$V^*(k, x_k) = \min_{U(k), \bar{\theta}(k)} V\left(k, x_k, U(k), \bar{\theta}(k)\right) \tag{8}$$

s.t.

$$U(k) = \begin{pmatrix} u(k|k) & \dots & u(k+N_c-1|k) \end{pmatrix} \tag{9}$$

$$x(k|k) = x_k \tag{10}$$

$$\begin{aligned} x(k+i+1|k) &= A(k+i)x(k+i|k) + B(k+i)u(k+i|k), \\ & \quad i = 1, 2, ..., N_p - 1 \end{aligned} \tag{11}$$

$$\begin{pmatrix} x_s(k+i|k) \\ u_s(k+i|k) \end{pmatrix} = M(k+i)\bar{\theta}(k), \quad i = 1, 2, ..., N_p \tag{12}$$

$$\begin{aligned} u(k+i|k) &= K_{LQR}(k+i)\left(x(k+i|k) - x_s(k+i|k)\right) \\ & \quad + u_s(k+i|k), \quad i = N_c, ..., N_p - 2 \end{aligned} \tag{13}$$

$$\begin{aligned} u(k+i|k) &= K_{DB,x}(k+i)x(k+i|k) \\ & \quad + K_{DB,\theta}(k+i)\bar{\theta}(k), \quad i = N_p - 1, N_p \end{aligned} \tag{14}$$

$$A_x x(k+i|k) \leq b_x, A_u u(k+i|k) \leq b_u, \quad i = 1, 2, ..., N_p \tag{15}$$

$$A_x x_s(k+i|k) \leq b_x, A_u u_s(k+i|k) \leq b_u, \quad i = 1, 2, ..., N_p \tag{16}$$

$$A_{DB}\bar{\theta}(k) \leq b_{DB}. \tag{17}$$

Note that the MPC controller has one control horizon N_c and two prediction horizons $N_p - 2$ and N_p for the terminal controllers. Firstly, the reader is presented with a concise overview of the problem in (8)–(17). Subsequently, a detailed analysis of the definition of variables, linear transformations and constraints is provided.

A. Brief review of the optimization problem

The optimization problem has two variables. As in (9), $U(k)$ represents the predicted control inputs from k to the control horizon N_c, with only $u(k|k)$ applied in a receding horizon. In MPCT, $\bar{\theta}(k) \in \mathbb{R}^3$, the artificial reference, defines an equilibrium trajectory. The quadratic cost function in (8) includes a term for $\bar{\theta}(k)$, steering the system to an equilibrium while improving feasibility and region of attraction with minimal computational effort.

After N_c, explicit control laws from Section IV (see (13), (14)) are applied, reducing variables and enhancing performance. The predicted state, initialized at k, is propagated using the time-dependent law in (11).

The equilibrium trajectory is a linear transformation of $\bar{\theta}(k)$, as per (12). While $\bar{\theta}(k)$ is constant, the map $M(k+i)$ and the trajectory are time-dependent. Time-independent constraints are applied to the equilibrium trajectory in (16) and to the artificial reference in (17).

B. Definition of state and control variables

The structure of the optimization problem in (8)–(17) is such that all constraints are time-invariant. This is particularly convenient, as it allows to mathematically prove the recursive feasibility of the controller, as detailed later on. Note, however, that the state and control definitions used for propagation in (1), this is, measured in the LVLH reference frame, do not coincide with the ones for which constraints are constant inequalities, included in (5). For this reason, a partial change of reference frame is introduced.

If the initial attitude and angular velocity of the target are known, the corresponding rotation matrix evolution ${}^B_L C(k)$ can be computed up to an arbitrary horizon from (2) and (3). For the described attitude matrix, the following linear transformations hold for any given vector $\boldsymbol{a}$

$$\boldsymbol{a}^B(k) = {}^B_L C(k)\boldsymbol{a}^L(k),\ \boldsymbol{a}^L(k) = \left({}^B_L C(k)\right)^T \boldsymbol{a}^B(k). \quad (18)$$

Substituting in (1),

$$\begin{pmatrix} \left({}^B_L C(k+1)\right)^T \boldsymbol{r}^B(k+1) \\ \boldsymbol{v}^L(k+1) \end{pmatrix} = A_L \begin{pmatrix} \left({}^B_L C(k)\right)^T \boldsymbol{r}^B(k) \\ \boldsymbol{v}^L(k) \end{pmatrix} + B_L \left({}^B_L C(k)\right)^T \boldsymbol{u}^B(k). \quad (19)$$

It is useful to decompose A_L and B_L into the submatrices connecting position, velocity and control, this is,

$$A_L = \begin{bmatrix} A_{L,rr} & A_{L,rv} \\ A_{L,vr} & A_{L,vv} \end{bmatrix}, \quad B_L = \begin{bmatrix} B_{L,ru} \\ B_{L,vu} \end{bmatrix}. \quad (20)$$

Therefore, and given that ${}^B_L C(k)$ is known beforehand, (19) can be written as

$$\begin{pmatrix} \boldsymbol{r}^B(k+1) \\ \boldsymbol{v}^L(k+1) \end{pmatrix} = A_B(k) \begin{pmatrix} \boldsymbol{r}^B(k) \\ \boldsymbol{v}^L(k) \end{pmatrix} + B_B(k)\boldsymbol{u}^B(k), \quad (21)$$

where

$$A_B(k) = \begin{bmatrix} {}^B_L C(k+1) A_{L,rr} \left({}^B_L C(k)\right)^T & {}^B_L C(k+1) A_{L,rv} \\ A_{L,vr} \left({}^B_L C(k)\right)^T & A_{L,vv} \end{bmatrix} \quad (22)$$

$$B_B(k) = \begin{bmatrix} {}^B_L C(k+1) B_{L,ru} \left({}^B_L C(k)\right)^T \\ B_{L,vu} \left({}^B_L C(k)\right)^T \end{bmatrix}. \quad (23)$$

For the modified state $x(k) = \begin{pmatrix} \boldsymbol{r}^B(k) \\ \boldsymbol{v}^L(k) \end{pmatrix}$ and control input $u(k) = \boldsymbol{u}^B(k)$, all constraints are time-invariant, directly as written in (5). Therefore, if a position given by the first three components of $x(k)$ is chosen, and provided that it satisfies the LOS constraints at k, it is guaranteed to verify them at all future time steps. The cost incurred of introducing this new definition is that the propagation equations are no longer time-invariant. Indeed, the $A(k)$ and $B(k)$ matrices in (11) are given by $A_B(k)$ and $B_B(k)$, as written in (22) and (23).

C. Artificial reference and equilibrium trajectory

MPCT makes use of equilibrium points of the linear system as decision variables to increase the region of the attraction of the controller and enhance its stability and feasibility guarantees [20]. For an LTI linear system such as $x(k+1) = Ax(k) + Bu(k)$ an *equilibrium point* is given by any pair $\{x_e, u_e\}$ verifying the invariant equation $x_e = Ax_e + Bu_e$. In this work, this definition is successfully extended to LTV systems by means of *equilibrium trajectories*. For a time instant k, an equilibrium trajectory is given by the pair $\{x_e, u_e\}$ satisfying $x_e = A(k)x_e + B(k)u_e$. Consequently, let the set of all equilibrium trajectories at time k, $\mathcal{Z}(k)$ such that

$$\mathcal{Z}(k) = \left\{ z = \begin{pmatrix} x_e \\ u_e \end{pmatrix} \in \mathbb{R}^{n_x+n_u} : x_e = A(k)x_e + B(k)u_e \right\}. \quad (24)$$

The equilibrium trajectories are mapped by means of the artificial reference $\theta \in \mathbb{R}^{n_\theta(k)}$, for $n_\theta(k)$ the dimension of the subspace of equilibrium points at time k,

$$z = M(k)\theta = \begin{bmatrix} M_x(k)^T & M_u(k)^T \end{bmatrix}^T \theta, \forall z \in \mathcal{Z}(k). \quad (25)$$

The dimension $n_\theta(k)$ is in principle time-varying, as are matrices $A(k)$ and $B(k)$. Introducing (25) in the invariant equation, $M(k)$ can be derived as $\begin{bmatrix} A(k) - I_{n_x} & B(k) \end{bmatrix} M(k) = 0$. This is equivalent to $M(k) = \ker \begin{bmatrix} A(k) - I_{n_x} & B(k) \end{bmatrix}$. The following result, proven in [17], allows to define a consistent framework of equilibrium trajectories.

Lemma 1: The dimension of the space of equilibrium points for the LTV system given in (21) is constant and equal to $n_\theta(k) = 3$. Furthermore, it is possible to define $M(k)$ such that θ is equal to the position of the considered equilibrium point in B, $\theta = \boldsymbol{r}_e^B$, which is constant by definition.

As a consequence $M(k)$ can be written as

$$M(k) = \begin{bmatrix} I_3 & \cdots \end{bmatrix}^T. \quad (26)$$

In this formulation, the bijection $\theta \leftrightarrow \boldsymbol{r}_e^B(k)$ is the identity, thus equilibrium trajectories are uniquely associated to their constant position. Hence, if an equilibrium trajectory associated to θ initially verifies the LOS constraints, it does verify them for all times. The computation of (26) is straightforward and computationally efficient; any basis of the kernel defining $M(k)$ can be obtained and written in column echelon form to derive $M(k)$ as defined in this work.

IV. Design of the controller

With the state, control, and equilibrium trajectories defined, the next step is to characterize the virtual controllers in (13) and (14) and establish the additional artificial reference constraint in (17). These elements are closely linked to controller properties. For example, (13) enhances optimality and stability, while both (14) and (17) ensure recursive feasibility. Additionally, the controller requires defining weighting matrices for both the cost function (8) and the terminal LQR (13). This section covers these topics in detail, leading to a full characterization of the controller.

A. Terminal controllers design

After the control horizon, the optimizer explicitly computes a control input given by a two-phases policy. This leads to a terminal *virtual controller* applied up to the prediction horizon, as displayed schematically in Fig. 2. Between $k + N_c$ and $k + N_p - 2 = k + N_c + N$, with horizon $N = N_p - 2 - N_c$, an infinite-time time-varying LQR, derived from a rotation of the LVLH frame, steers the

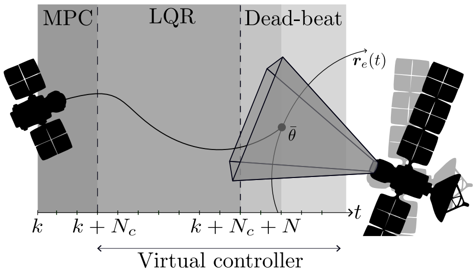

Fig. 2: Real and virtual controller layout.

state toward the target equilibrium trajectory. At $k + N_p - 1$ and $k + N_p$, an explicit dead-beat controller is applied to reach the target equilibrium. From $k + N_p + 1$ onward, a dead-beat controller tracks the target trajectory.

An infinite-time time-varying LQR for a moving target can be derived by constructing the controller in the LVLH reference frame, in which dynamics are LTI. Indeed, let K_F and P_{LQR} be the gain and Lyapunov matrix for state and control weighting matrices Q_{LQR} and R_{LQR}, respectively. For system $\{A_L, B_L\}$, an infinite horizon LQR is described by

$$\begin{aligned}\boldsymbol{u}^L(k+i|k) = K_F &\begin{pmatrix}\boldsymbol{r}^L(k+i|k) - \boldsymbol{r}_s^L(k+i|k) \\ \boldsymbol{v}^L(k+i|k) - \boldsymbol{v}_s^L(k+i|k)\end{pmatrix} \\ &+ \boldsymbol{u}_s^L(k+i|k).\end{aligned} \tag{27}$$

Thus, expressing the control law in the modified reference frame, a control law as given in (13) is obtained, where

$$K_{LQR}(k+i) = {}^B_L C(k+i) K_F \begin{bmatrix}{}^B_L C(k+i)^T & 0_3 \\ 0_3 & I_3\end{bmatrix}. \tag{28}$$

The weighting matrices Q_{LQR} and R_{LQR} can differ from those in the MPC cost function. Notably, $K_{LQR}(k+i)$ is state-independent and can be precomputed, enabling long horizons without impacting computational performance.

After the LQR, a two-step dead-beat controller is used to reach the target. This controller is first computed in the LVLH frame, then aligned with (14). Without constraints, it steers any state to the target as $\begin{pmatrix}B_L & A_L B_L\end{pmatrix}$ is full rank. Given B_L's structure, $\boldsymbol{u}^L(k)$ only impacts the state at $k+2$. By applying $\boldsymbol{u}^L(k+N_p-1|k)$ to guide $\boldsymbol{r}^L(k+N_p+1|k)$ to $\boldsymbol{r}_e^L(k+N_p+1|k)$, and $\boldsymbol{u}^L(k+N_p|k)$ to ensure $\boldsymbol{r}^L(k+N_p+2|k) = \boldsymbol{r}_e^L(k+N_p+2|k)$, the target is reached and tracked precisely from $k+N_p+1$ onward.

Following this strategy, the control actions at $k+i$, with $i = N_p - 1$ and $i = N_p$, are proved in [17] to be linear functions of $x(k+i|k)$ and $\bar{\theta}(k)$ only, such as

$$u(k+i|k) = K_{DB,x}(k+i)x(k+i|k) + K_{DB,\theta}(k+i)\bar{\theta}(k), \tag{29}$$

where $\hat{K}_{DB} = \begin{pmatrix}\hat{K}_{DB,r} & \hat{K}_{DB,v}\end{pmatrix}$, and for

$$\hat{K}_{DB,r} = -A_{L,rv}^{-1}\left(A_{L,rr}^2 + A_{L,rv}A_{L,vr}\right) \tag{30}$$

$$\hat{K}_{DB,v} = -\left(A_{L,rr}A_{L,rv} + A_{L,rv}A_{L,vv}\right) \tag{31}$$

$$K_{DB,x}(k+i) = {}^B_L C(k+i)\hat{K}_{DB}\begin{bmatrix}{}^B_L C(k+i)^T & 0_3 \\ 0_3 & I_3\end{bmatrix} \tag{32}$$

$$K_{DB,\theta}(k+i) = {}^B_L C(k+i) A_{L,rv}^{-1}\, {}^B_L C(k+i+2)^T. \tag{33}$$

B. Feasibility constraint

Once the initial dead-beat controller guides the system to the target equilibrium trajectory, a secondary controller is employed for trajectory tracking. Rather than imposing constraints on each input, the maximum control authority required for indefinite tracking is calculated and compared to available limits for feasibility. Additionally, a linear constraint on the artificial reference θ is introduced to ensure recursive feasibility, evaluated using a dead-beat controller.

For analysis, consider the following results: The position of a given equilibrium point remains at a constant distance from the target, i.e., $\boldsymbol{r}_e^L(k) \in \partial B(|\theta|)$, where $B(|\theta|)$ is a 3-ball of radius θ and $\partial B(|\theta|)$ its contour. The norm of the velocity vector is bounded by angular momentum conservation. For inertia components $I_3 \geq I_2 \geq I_1$, $|\boldsymbol{v}_e^L(k)| \in [0, |\boldsymbol{h}|/I_3]$, with $\boldsymbol{h}$ as the target's angular momentum. The equilibrium point velocity in LVLH axes is given by $\boldsymbol{v}_e^L(k) = {}^B_L\boldsymbol{\omega}(k) \times \boldsymbol{r}_e^L(k)$, and the evolution of ${}^B_L\boldsymbol{\omega}(k)$ is derived from Euler-Poinsot equations in (2), ensuring that both momentum $\boldsymbol{h}$ and rotational energy E remain invariant.

For small angular velocities with respect to the sampling time T, the motion of a given equilibrium point can be written as $\boldsymbol{r}_e^L(k+1) = \boldsymbol{r}_e^L(k) + \boldsymbol{v}_e^L(k)T$. The position of the chaser is propagated as $\boldsymbol{r}^L(k+1) = A_{L,rr}\boldsymbol{r}^L(k) + A_{L,rv}\boldsymbol{v}^L(k)$. Therefore, the chaser can follow an equilibrium point position only if

$$\boldsymbol{v}^L(k) = A_{L,rv}^{-1}\left[\left(I_3 - A_{L,rr}\right)\boldsymbol{r}_e^L(k) + \boldsymbol{v}_e^L(k)T\right]. \tag{34}$$

The control itself is involved in the propagation of the chaser's velocity, so that

$$\begin{aligned}\boldsymbol{u}^L(k) &= \boldsymbol{v}^L(k+1) - A_{L,vv}\boldsymbol{v}^L(k) - A_{L,vr}\boldsymbol{r}_e^L(k) \\ &= A_{L,rv}^{-1}\left[\left(I_3 - A_{L,rr}\right)\boldsymbol{r}_e^L(k+1) + \boldsymbol{v}_e^L(k+1)T\right] \\ &\quad - A_{L,vv}A_{L,rv}^{-1}\left[\left(I_3 - A_{L,rr}\right)\boldsymbol{r}_e^L(k) + \boldsymbol{v}_e^L(k)T\right] \\ &\quad - A_{L,vr}\boldsymbol{r}_e^L(k).\end{aligned}$$

The kinematics of the equilibrium points relate position in two consecutive iterations. As for velocity, the following can be written for a small sampling time,

$$\begin{aligned}\boldsymbol{v}_e^L(k+1) &= {}^B_L\boldsymbol{\omega}^L(k+1) \times \boldsymbol{r}_e^L(k+1) \\ &= \left({}^B_L\boldsymbol{\omega}^L(k) + T\,{}^B_L\dot{\boldsymbol{\omega}}^L(k)\right) \times \left(\boldsymbol{r}_e^L(k) + \boldsymbol{v}_e^L(k)T\right),\end{aligned}$$

where $\boldsymbol{v}_e^L(k) = {}^B_L\boldsymbol{\omega}^L(k) \times \boldsymbol{r}_e^L(k)$. Simplifying from orthogonality, retaining terms up to the first order and introducing the first order kinematics in the control law, where $A'_{L,rr} = (I_3 - A_{L,rr})$ and $\Delta\boldsymbol{v}_e^L(k) = T\,{}^B_L\dot{\boldsymbol{\omega}}^L(k) \times \boldsymbol{r}_e^L(k)$.

$$\begin{aligned}\boldsymbol{u}^L(k) = &\left[A_{L,rv}^{-1}A'_{L,rr} - A_{L,vv}A_{L,rv}^{-1}A'_{L,rr} - A_{L,vr}\right]\boldsymbol{r}_e^L(k) \\ &+ \left[TA_{L,rv}^{-1}A'_{L,rr} - TA_{L,vv}A_{L,rv}^{-1} + TA_{L,rv}^{-1}\right]\boldsymbol{v}_e^L(k) \\ &+ \left[TA_{L,rv}^{-1}\right]\Delta\boldsymbol{v}_e^L(k).\end{aligned}$$

Condensing notation,

$$\boldsymbol{u}^L(k) = \mathcal{U}^L\begin{pmatrix}\boldsymbol{r}_e^L(k)^T & \boldsymbol{v}_e^L(k)^T & \Delta\boldsymbol{v}_e^L(k)^T\end{pmatrix}^T, \tag{35}$$

where $\mathcal{U}^L = \begin{bmatrix}\mathcal{U}_r^L & \mathcal{U}_v^L & \mathcal{U}_{\Delta v}^L\end{bmatrix}$ is a constant matrix, given a sampling time, for varying rotational states of the target. This control policy can be rotated to body axes B, leading to

$$\boldsymbol{u}^B(k) = {}^B_L C(k)\mathcal{U}^L\left({}^B_L\bar{C}(k)\right)^T\begin{pmatrix}\boldsymbol{r}_e^B(k) \\ \boldsymbol{v}_e^B(k) \\ \Delta\boldsymbol{v}_e^B(k)\end{pmatrix}, \tag{36}$$

where ${}^B_L\bar{C}(k) = I_3 \otimes {}^B_L C(k)$, for $\otimes$ the Kronecker product. The proposed control action must satisfy the corresponding constraints, this is,

$$A_u\,{}^B_L C(k)\mathcal{U}^L\left({}^B_L\bar{C}(k)\right)^T\begin{pmatrix}\boldsymbol{r}_e^B(k) \\ \boldsymbol{v}_e^B(k) \\ \Delta\boldsymbol{v}_e^B(k)\end{pmatrix} \leq b_u, \forall k. \tag{37}$$

Interestingly, as proved in [17], this constraint is satisfied for an equilibrium trajectory, provided that

$$A_u \left\|\mathcal{U}^L\right\| \begin{pmatrix} x_1 & x_2 & x_3 \end{pmatrix}^T \le b_u,$$
$$\forall\ x_1 \in B(\|\theta\|), x_2 \in B\left(\|\theta\|\frac{\|\boldsymbol{h}\|}{I_3}\right), x_3 \in B\left(\sqrt{\varpi^*}T\|\theta\|\right), \tag{38}$$

for the *matrix 2-norm* operator $\left\|\mathcal{U}^L\right\|$ defined as usual, and where,

$$\varpi^* = \max_{i \neq j \in \{1,2,3\}} \left\{ -\frac{\left(I_i + I_j\right)^2}{\left(I_i I_j\right)^3} \left(h^2 - 2I_i E\right)\left(h^2 - 2I_j E\right) \right\}. \tag{39}$$

Inequality (38), dependent only on the norm of θ, remains constant over time. Therefore, the set of feasible trajectories, defined by θ, can be precomputed offline for all controllers, forming a sphere. This constraint determines the maximum distance for equilibrium points based on $|\boldsymbol{h}|/I_3$ and $\sqrt{\varpi^*}T$. For a given chaser sampling time, the maximum allowable θ can be computed from the target's rotational state, defining key admissible rendezvous missions and aiding chaser design. Explicit computation of θ minimizes implementation costs. Details are provided in [17].

Lemma 2: Let the nonzero matrices $A^1, ..., A^l \in \mathbb{R}^{m \times n}$. Let $b \in \mathbb{R}^m$, $r \in \mathbb{R}^l$ such that $b \geq 0, r > 0$. For the scalar $\theta \in \mathbb{R}^+$, let the sets given by n-balls of radius $r_j\theta$, $\Omega_j = B^n(r_j\theta)$, $j = 1, ..., l$. Consider the following inequality,

$$\sum_{j=1}^{l} A^j x_j \le b, \forall x_j \in \Omega_j. \tag{40}$$

The maximum value θ for which (40) holds is given by

$$\theta_{max} = \min_{i \in \mathbb{N} \cap [1,m]} \left\{ \frac{b_i}{\sum_{j=1}^{l} r_j \sqrt{\sum_{k=1}^{n} \left(A_{ik}^j\right)^2}} \right\}. \tag{41}$$

This allows solving (38) explicitly with minimal extra cost. After computing θ_{max} offline, the optimization adds the constraint $\theta^T\theta \le \theta_{max}^2$ to ensure stability and feasibility, turning the problem into a convex QCQP. Approximating the constraint on $|\theta|$ with a polytope simplifies the MPC to a QP with linear constraints, making it more practical. A single polytope can include both the LOS constraint and the maximum norm for θ, approximating with a few extra linear inequalities. Specifically, the limit sphere for θ is approximated as the XZ plane in body axes, with y^B bounded by

$$y_{max}^B = c_x c_z \frac{\left(\sqrt{\xi_3 \theta_{max}^2 - \xi_1} - \xi_2\right)}{\xi_3} \tag{42}$$

where

$$\xi_1 = c_x^2 c_z^2 (x_0^2 + z_0^2) + (c_z z_0 + c_x x_0)^2 \tag{43}$$
$$\xi_2 = c_x z_0 + c_z x_0 \tag{44}$$
$$\xi_3 = c_x^2 c_z^2 + c_x^2 + c_z^2. \tag{45}$$

The constraint (17) for the artificial reference is then $\begin{bmatrix} 0 & 1 & 0 \end{bmatrix} \bar{\theta} \le y_{max}^B$.

For the rendezvous conditions in [13], $\theta_{max} = 3.597$ m ensures control feasibility and stability, assuming initial conditions within this region Importantly, θ_{max} represents the maximum distance the algorithm can adjust θ in a given iteration, not the maximum initial distance. This suggests a potentially larger region of attraction, depending on the control horizon.

V. Implementation and properties of the controller

This section shows that the controller designed in Section IV is stable, recursively feasible to infinite horizon and computationally efficient. Let U_k be a feasible solution of (8)–(17) at some time k. Due to the structure of the virtual controller, U_k predicts an infinite sequence of control inputs such as

$$U_k = \begin{pmatrix} U_{MPC}(k|k+N_c-1) \\ U_{LQR}(k+N_c|k+N_p-2) \\ U_{DB}(k+N_p-1|\infty) \end{pmatrix}^T. \tag{46}$$

Assume that at least one feasible solution such as U_k exists, this is, the optimization problem is feasible at k. Furthermore, let the control input at k be given by $u(k|k)$, this is, the solution proposed by the MPC policy. Let the candidate control input at $k+1$ given by

$$U_{k+1} = \begin{pmatrix} U_{MPC}(k+1|k+N_c-1) \\ U_{LQR}(k+N_c|k+N_p-1) \\ U_{DB}(k+N_p|\infty) \end{pmatrix}^T. \tag{47}$$

The construction of this candidate feasible solution is shown in Fig. 3. The following result holds (see [17]).

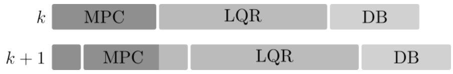

Fig. 3: Finding a feasible solution at $k+1$ from k.

Lemma 3: Let a solution U_k of the QP problem in (8)–(17) that is feasible, for an initial state $x(k|k)$ and a number of iterations N within the LQR that satisfy $\|x(k+N_c+N|k) - x_e(k+N_c+N|k)\|_{Q_{LQR}+K_F^T R_{LQR} K_F} \ll 1$. In the next iteration, there is at least one solution such as U_{k+1} that is feasible, hence leading to recursive feasibility.

A Lyapunov stability analysis of (8) (see [21]) confirms that the controller is asymptotically stable if feasible. The optimization problem (8)–(17) reduces to a QP problem, minimizing a quadratic cost with linear constraints. Using the linear propagation equation (11) and known $A(k)$ and $B(k)$, state propagation over the horizon is efficiently handled with a single linear transformation. Both LQR and dead-beat controllers are explicit linear functions of the state, allowing for a high LQR horizon N with minimal computational cost, while maintaining a low N_c, as done here, without compromising performance.

VI. Case study and simulation results

We follow [13] and consider for simulation a near rendezvous with the Envisat spacecraft, a relevant scenario typical in ADR. The simulation initial conditions are included in Table I. As for the state and control constraints, the corresponding parameters are $c_x = c_z = 1$, $x_{min} = z_{min} = 0.1$ m, $u_{max} = u_{min} = 0.075$ m/s.

$\boldsymbol{r}^B$	$(1.5, 2.5, 1.5)^T$ m
$\boldsymbol{v}^L$	$(0, 0, 0)^T$ m/s
${}^B_L q$	$(1, 0, 0, 0)^T$
${}^B_L \boldsymbol{\omega}^B$	$(0, 3.53, 3.53)^T$ deg/s

TABLE I: Simulation initial conditions.

For the controller design, a sampling period of $T = 1$ s is selected, a standard for final rendezvous phases. The MPC phase uses $N_c = 3$ control inputs, while the final LQR has a larger horizon of $N = 30$. Both are heuristically tuned, with a total of 35 iterations. Feasibility conditions are imposed per Section VI, using a simplified dead-beat constraint, resulting in $y_{max} = 2.008$ m. Simulation results are shown in Figs. 4–5. Since C's initial state lies within the controller's feasible polytope, successful rendezvous is guaranteed. The proposed controller expands the domain of attraction, especially with MPCT. To ensure a fair comparison, the structure of (8)–(17) is retained, with $\bar{\theta} = 0$ to simplify optimization. The problem becomes infeasible until $N_c = 35$, whereas the proposed MPCT strategy uses an equivalent horizon of $N_c + 1 = 4$. This approach reduces decision variables while improving feasibility.

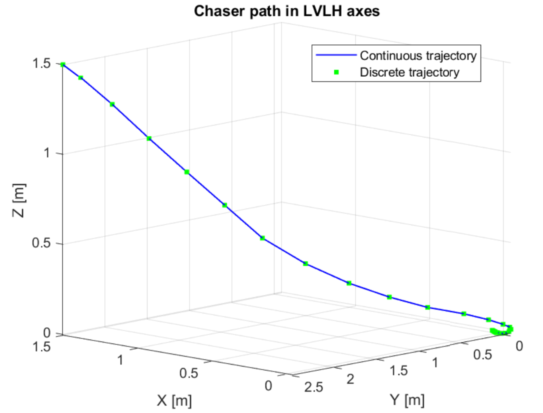

Fig. 4: Chaser controlled rendezvous path in LVLH axes.

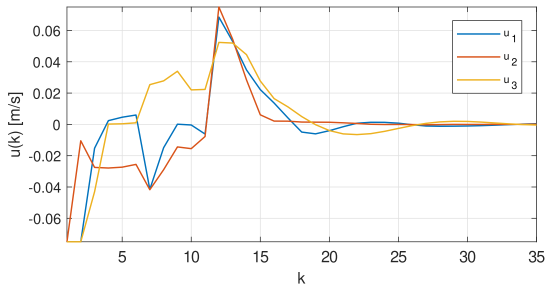

Fig. 5: Control signal evolution for the controlled approach.

VII. Conclusions and Future Lines of Work

As shown in simulation, the controller steers the chaser to the rotating target respecting highly restrictive actuator constraints by making use of the orbit and target's rotation dynamics. Moreover, the controller is asymptotically stable, locally optimal, provides recursive feasibility by design, and is computationally efficient. Future work includes robustifying the controller by using e.g. tube-based MPC, including uncertainties in the estimation of the target's rotational state and inertia tensor, extending the result to elliptic orbits or including 6DOF chaser's dynamics.

Acknowledgments

This work was supported by grant TED2021-132099B-C33 funded by MICIU/AEI/10.13039/501100011033 and by "European Union NextGenerationEU/PRTR."

References

[1] M. Shan, J. Guo, and E. Gill, "Review and comparison of active space debris capturing and removal methods," *Prog. Aerosp. Sci.*, vol. 80, pp. 18–32, 2016.

[2] C. P. Mark and S. Kamath, "Review of active space debris removal methods," *Space Policy*, vol. 47, pp. 194–206, 2019.

[3] G. Sánchez-Arriaga *et al.*, "The e.t.pack project: Towards a fully passive and consumable-less deorbit kit based on low-work-function tether technology," *Acta Astronaut.*, vol. 177, p. 821–827, 2020.

[4] G. Sarego *et al.*, "Deployment requirements for deorbiting electrodynamic tether technology," *CEAS Space J.*, vol. 13, p. 567–581, 2021.

[5] M. Castronuovo, "Active space debris removal—a preliminary mission analysis and design," *Acta Astronaut.*, vol. 69, pp. 848–859, 2011.

[6] L. Zhang, S. Zhang, H. Yang, H. Cai, and S. Qian, "Relative attitude and position estimation for a tumbling spacecraft," *Aerosp. Sci. Technol.*, vol. 42, p. 97–105, 2015.

[7] G. Behrendt, A. Soderlund, M. Hale, and S. Phillips, "Autonomous satellite rendezvous and proximity operations with time-constrained sub-optimal model predictive control," *IFAC-PapersOnLine*, vol. 56, no. 2, pp. 9380–9385, 2023.

[8] Q. Li, J. Yuan, B. Zhang, and C. Gao, "Model predictive control for autonomous rendezvous and docking with a tumbling target," *Aerosp. Sci. Technol.*, vol. 69, pp. 700–711, 2017.

[9] A. Caubet and J. D. Biggs, "An inverse dynamics approach to the guidance of spacecraft in close proximity of tumbling debris," *2015 IAC*, vol. 7, p. 9, 2015.

[10] D. Limon, T. Alamo, D. M. Raimondo, D. M. De La Peña, J. M. Bravo, A. Ferramosca, and E. F. Camacho, "Input-to-state stability: a unifying framework for robust model predictive control," *NMPC: Towards New Challenging Applications*, pp. 1–26, 2009.

[11] M. Mammarella, E. Capello, H. Park, G. Guglieri, and M. Romano, "Tube-based robust model predictive control for spacecraft proximity operations in the presence of persistent disturbance," *Aerosp. Sci. Technol.*, vol. 77, pp. 585–594, 2018.

[12] C. Buckner and R. Lampariello, "Tube-based model predictive control for the approach maneuver of a spacecraft to a free-tumbling target satellite," in *2018 ACC*. IEEE, 2018, pp. 5690–5697.

[13] C. E. Oestreich, R. Linares, and R. Gondhalekar, "Tube-based model predictive control with uncertainty identification for autonomous spacecraft maneuvers," *J. Guid. Control Dyn.*, vol. 46, pp. 6–20, 2023.

[14] J. B. Rawlings, D. Q. Mayne, and M. Diehl, *Model predictive control: Theory, computation, and design*. Nob Hill Publishing, 2017.

[15] D. Limon, A. Ferramosca, I. Alvarado, and T. Alamo, "Nonlinear MPC for tracking piece-wise constant reference signals," *IEEE Trans. Aut. Control*, vol. 63, no. 11, pp. 3735–3750, 2018.

[16] K. Dong, J. Luo, and D. Limon, "A novel stable and safe model predictive control framework for autonomous rendezvous and docking with a tumbling target," *Acta Astronaut.*, vol. 200, pp. 176–187, 2022.

[17] J. A. Rebollo, R. Vazquez, I. Alvarado, and D. Limon, "MPC for Tracking applied to rendezvous with non-cooperative tumbling targets ensuring stability and feasibility," 2024, available at: http://arxiv.org/abs/2403.10986.

[18] G. W. Hill, "Researches in the lunar theory," *American journal of Mathematics*, vol. 1, pp. 5–26, 1878.

[19] W. Clohessy and R. Wiltshire, "Terminal guidance system for satellite rendezvous," *J. Aerosp. Sci.*, vol. 27, pp. 653–658, 1960.

[20] A. Ferramosca, D. Limon, I. Alvarado, T. Alamo, and E. F. Camacho, "MPC for tracking with optimal closed-loop performance," *Automatica*, vol. 45, pp. 1975–1978, 2009.

[21] D. Limon, I. Alvarado, T. Alamo, and E. Camacho, "MPC for tracking piecewise constant references for constrained linear systems," *Automatica*, vol. 44, no. 9, pp. 2382–2387, 2008.