doi: 10.82124/CEAS-GNC-2026-048 **CEAS-GNC-2026-048**

EuroGNC Madrid, Spain May 5th-7th 2026

Recovering Classical Rendezvous Strategies using MPC for Constrained Closed-Loop Guidance

Diogo Silva PhD Candidate, NOVA School of Science and Technology (NOVA-FCT), 2829-516 Caparica, Portugal. dgn.silva@campus.fct.unl.pt
Center of Technology and Systems and LASI (UNINOVA-CTS).
Doctoral Researcher, Institute for Systems and Robotics (ISR-Lisbon), Instituto Superior Técnico (IST), 1049-001 Lisbon, Portugal.

Pedro Lourenço Section Head Advanced Guidance & Control, GMV Innovating Solutions, Lisbon, Portugal. palourenco@gmv.com

Daniel Silvestre Assistant Professor, NOVA School of Science and Technology (NOVA-FCT), Department of Electrical and Computer Engineering, 2829-516 Caparica, Portugal. dsilvestre@fct.unl.pt
Researcher, UNINOVA – CTS and LASI, Caparica, Portugal
Researcher, Institute for Systems and Robotics Instituto Superior Técnico Universidade de Lisboa, Lisboa, Portugal

José F. Briz GNC Engineer, GMV Innovating Solutions, Madrid, Spain. jbriz@gmv.com

ABSTRACT

Future space missions depend on autonomous rendezvous, ranging from human exploration and large-scale in-orbit assembly to debris removal and satellite servicing. Classical strategies such as V- and R-Bar hopping have been extensively used due to their safety guarantees, holding points, and structured phases, yet their implementation has traditionally relied on pre-generated trajectories that are executed in flight. This approach limits adaptability and does not fully exploit modern control techniques, particularly when mission-specific constraints such as Keep Out Zones or spacecraft restrictions must be considered.

This work addresses these limitations by proposing a model predictive control formulation for the short-range phase of rendezvous that reconstructs classical V-, R-Bar, and generalized glideslope hopping approaches while recovering them directly in the optimization problem. The proposed framework incorporates additional mission constraints, including Keep Out Zones enforced via Control Barrier Functions, alongside the approaches, ensuring they remain valid under different dynamical models. The results show that the controller is able to autonomously recover the expected classical hopping behaviors, maintain passive abort safety, and satisfy constraints in both single-run and Monte Carlo simulations.

Keywords: Model Predictive Control, Autonomous Rendezvous, Guidance Navigation and Control, Constrained Control

Nomenclature

N = Model Predictive Controller (MPC) prediction horizon length
t_{orb} = Orbital period of target
w = Orbital rate
θ = Approach angle
γ = Hop aggressiveness parameter
ρ = Displacement fraction
$h()$ = Control Barrier Function (CBF)

1 Introduction

Guidance Navigation and Control (GNC) systems play a critical role in the success of space missions and continue to evolve as a result of intensive research efforts within both academia and industry [1]. However, the inherently conservative nature of the space sector has led to the persistence of classical mathematical tools that lag behind the state-of-the-art in control and estimation theory, despite significant advancements in computational power [2]. This gap between theoretical progress and practical implementation is further amplified by the emergence of new space applications. The rise of NewSpace and the democratization of access to space have introduced cost-driven technologies such as cubesats, reusable launchers, and space ferrying, while established space agencies (ESA, NASA, etc.) are increasingly pursuing technologically demanding missions [3]. One such challenge is autonomous spacecraft rendezvous, which is becoming ever more relevant with the advent of in-orbit servicing, refueling depots, and human exploration missions. The Artemis program demonstrates the complexity of rendezvous operations, requiring precise control to dock the Orion spacecraft with the Lunar Gateway [4]. More recently, SpaceX's Starship [5] architecture has introduced a new layer of complexity, requiring multiple autonomous refuelings in orbit to enable missions to the Moon and Mars. Beyond human spaceflight, autonomous rendezvous plays a key role in active debris removal, satellite servicing, and large-scale in-orbit assembly [6]. Missions such as Northrop Grumman's MEV [7] and ESA's upcoming ClearSpace-1 [8] demonstrate the importance of precise guidance and control systems for docking with uncooperative targets [9]. Optimization-based methods are a promising answer to those needs, which can take into account uncertainties, dynamic constraints, and mission-specific objectives [10].

In rendezvous missions, classical strategies such as V- and R-Bar hopping have been widely adopted since the Space Shuttle era [11]. These approaches are valued for their safety guarantees, indefinite holding options, predefined checkpoints, and the ability to divide the approach into well-structured phases. They allow precise navigation corrections, ensure safe separation at each stage, and facilitate coordinated operations between ground control, the crew, and the target vehicle [12]. Despite these advantages, the application of such strategies in the modern context often remains limited to reproducing trajectories generated offline, which are then tracked by a guidance algorithm [13–15]. While effective in nominal scenarios, this methodology does not fully exploit the capabilities of modern control. It results in pre-determined trajectories that cannot adapt to contingencies or optimize performance in real time. Even when mission-specific constraints such as Keep Out Zones (KOZ) are considered, they are often incorporated in a restricted manner, without leveraging optimization to address all constraints simultaneously while accounting for the current spacecraft state.

The objective of this work is to develop a Model Predictive Controller (MPC) for the terminal phase of rendezvous, up to 10 km, capable of executing these classical approaches in an optimization-based framework. The proposed formulation recovers these approaches directly from the cost function and constraints, while also allowing the incorporation of additional mission requirements into the same structure. Since the formulation is expressed directly in terms of the spacecraft state, it can be used with different dynamical models by mapping their states into the chosen state representation used in the developed approach.

By recovering classical rendezvous strategies directly within a constrained MPC formulation, this framework also opens the door to extensions that account for realistic actuation effects, such as thruster deadbands and minimum firing times, without sacrificing convexity or real-time implementability, as explored in [16].

1.1 Notation

Vectors are represented in bold non-italic lowercase, $\mathbf{a} \in \mathbb{R}^n$, while matrices are represented in bold non-italic uppercase, $\mathbf{A} \in \mathbb{R}^{n \times m}$. Scalar variables, $a \in \mathbb{R}$, are written in lowercase. The (i, j)-th entry of a matrix $\mathbf{A}$ is denoted by A_{ij}, and the i-th component of a vector $\mathbf{a}$ is denoted by a_i. Sets are written in capital blackboard-bold letters, such as $\mathbb{R}$. The operator $\| \cdot \|_2$ denotes the 2-norm of a vector, $\| \cdot \|_\infty$ represents the ∞-norm, and $\| \cdot \|_1$ denotes the 1-norm, and the notation $\mathrm{diag}(a_1, \ldots, a_N)$ represents a diagonal matrix with diagonal entries $a_1, \ldots, a_N$.

2 Review of Classical V/R-Bar Approaches

Many targets have specific sensors or docking hardware positioned on certain sides, which need to be taken into account in the final phases of approach, constraining angles/directions in which the rendezvous approach can be made. Other constraints, like thruster positioning and minimum firing time, environmental disturbances, lighting conditions necessary for the approach, Field of View requirements, passive abort safety to handle thruster failures, communication with ground stations, and more, also need to be taken into account [17, Pages 109–123]. As a result, while far-range rendezvous can often be planned mainly by trading fuel against time, the final approach phase is driven by these operational constraints, and trajectories that are suboptimal in fuel or time may still be selected because they better satisfy safety and mission requirements.

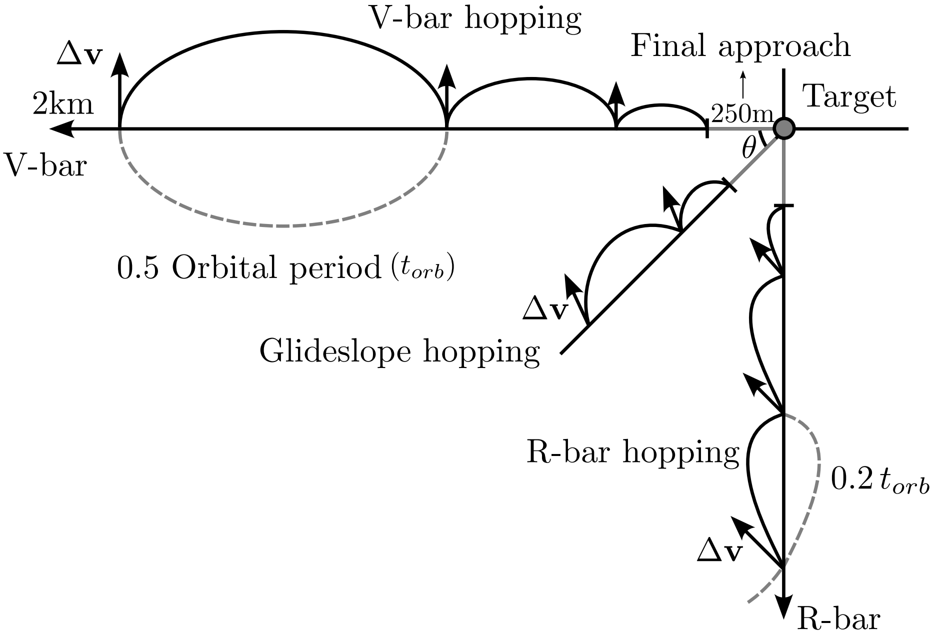


Fig. 1 Final approaches to target, adapted from [18]

The Local Horizontal Local Vertical (LVLH) frame is commonly employed in the final phase of approach, where the x and z axes are referred to as the V-Bar and R-Bar, respectively. This can be observed in Figure 1, which also illustrates typical strategies used in the last stage of the approach. The

LVLH frame represents a relative reference system between the chaser and the target, being centered on the latter. In this work it is defined with the z axis directed towards the Earth's center, the y axis aligned with the negative angular momentum vector of the target orbit, and the x axis completing the orthogonal triad, typically pointing in the direction of the target's motion.

The first approach to be discussed is along the V-Bar direction. After reaching the desired initial distance, typically around a few kilometers, although this depends on the size of the target, an impulsive maneuver is executed in the radial direction, pointing to nadir, to initiate a hop. Alternatively, a tangential transfer maneuver could be performed in the V-Bar direction, aligning with the velocity vector, similar to a small Hohmann transfer [19, Section 14.7]. While this alternative is more fuel-efficient for a two-impulse burn, as previously noted, it has a significant drawback. If the second impulse at the end of the hop cannot be executed, the chaser will not return to its initial position. In contrast, a radial burn, although consuming more fuel by a factor of $\frac{3\pi}{2}$ [20, Page 55], ensures the chaser's return to the starting point in the event of any aborted impulses, leading to the main advantage of this approach, which is the ability to stay indefinitely in a closed relative orbit which passes through the hold point. This behavior is illustrated in Figure 1, where the dotted gray trajectory demonstrates the chaser returning to its starting position after the time it takes to complete a full orbit (t_{orb}).

Another approach involves maneuvering along the R-Bar direction, with the chaser beginning from below the target. Here, impulsive maneuvers after each hop include equal $\Delta\mathbf{v}$ components in both the x and z axes. The primary advantage of this approach is its shorter hop duration of $0.2t_{orb}$, facilitating faster progress both when the operation proceeds as planned and when recovery is required, however, unlike V-Bar hopping, the chaser will not stay indefinitely in a relative closed orbit around the hold point [18]. This approach also comes at the cost of increased fuel consumption due to the need to counteract not only the Coriolis force, which is also present in the V-Bar approach, but also centrifugal forces [14].

To summarize, the V-Bar approach offers advantages in terms of maintaining trajectory stability and fuel efficiency, while the R-Bar approach excels in achieving faster progress and recovery times [12]. Additionally, as seen in Figure 2, the R-Bar approach allows for a larger safety boundary in passive abort scenarios. This is due to the greater centrifugal force acting on the chaser compared to the V-Bar approach.

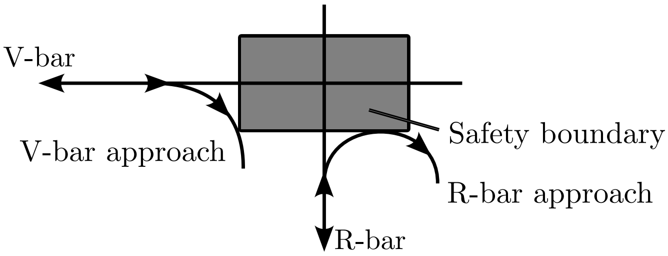


Fig. 2 Safety boundary example, adapted from [12]

By generalizing the V-Bar and R-Bar approaches, it is possible to derive what is known as the Glideslope approach. This method is characterized by parameters such as the number of hops, the duration of each hop, the angle formed with the velocity vector, and the starting and ending distances [21]. This generalization allows the approach to accommodate a wide range of constraints, including those previously discussed, making it a flexible and robust framework for rendezvous operations.

It is worth noting that in these approaches the distance traveled within each hop typically decreases as the chaser approaches the target. When the chaser is far away the target position is more uncertain, which makes smaller steps near the end desirable since they gradually improve the accuracy of the estimate. In addition to this effect, larger hops amplify the errors associated with prediction, which means that a single large hop toward the target would carry a higher margin of error and could even lead to a collision. For these reasons it is preferable to approach the target through successive increments of decreasing length.

As the chaser approaches within approximately 250 m of the target, the final phase of the approach begins. In this phase, a straight-line trajectory is followed, necessitating continuous thrusting. Although this consumes more fuel compared to hopping, the total $\Delta\mathbf{v}$ required is relatively small when compared to the overall mission, and it allows for a more precise, faster, and safer approach.

3 Problem Statement

To define the problem, consider a target spacecraft on an arbitrary orbit around Earth and subject to external disturbances. The evolution of the relative spacecraft dynamics is described in discrete time using the Clohessy-Wiltshire (CW) model [22] in the LVLH frame as:

$$\mathbf{x}(k+1) = \mathbf{A}_{CW}\mathbf{x}(k) + \mathbf{B}_{CW}\mathbf{u}(k) \tag{1}$$

where k denotes the discrete time step, $\mathbf{A}_{CW} \in \mathbb{R}^{6\times 6}$ characterizes the homogeneous dynamics, and $\mathbf{B}_{CW} \in \mathbb{R}^{6\times 3}$ represents the control influence. The state vector is defined as $\mathbf{x} = [p_x, p_y, p_z, v_x, v_y, v_z]^\intercal$, where p and v represent the position (m) and velocity (m/s) in each of the axis respectively, while the control input is $\mathbf{u} = [u_x, u_y, u_z]^\intercal$ (m/s^2). Since the CW model is a Linear Time Invariant (LTI) system, the matrices $\mathbf{A}_{CW}$ and $\mathbf{B}_{CW}$ are constant, which significantly simplifies the formulation.

To design the MPC controller capable of executing the required tasks, the control problem can be formulated as:

$$\begin{aligned}
\min_{\mathbf{x}_i,\mathbf{u}_i} J = J_N(\mathbf{x}_N, k+N) + \sum_{i=0}^{N-1} \ell(\mathbf{x}_i, \mathbf{u}_i, k+i) \\
\text{s.t.} \quad & \mathbf{x}_{i+1} = \mathbf{A}\mathbf{x}(k) + \mathbf{B}\mathbf{u}(k), \\
& \mathbf{x}_0 = \hat{\mathbf{x}}_k, \\
& \mathbf{h}(\mathbf{x}_i, \mathbf{u}_i, k+i) = \mathbf{0}, \\
& \mathbf{g}(\mathbf{x}_i, \mathbf{u}_i, k+i) \leq \mathbf{0}, \quad \forall i \in [0, N-1]
\end{aligned} \tag{2}$$

where i denotes the iterations of the MPC performed at each time step k. The term $\hat{\mathbf{x}}_k \in \mathbb{R}^{n_x}$ represents the state estimate at step k, while $\ell(\mathbf{x}_i, \mathbf{u}_i, k+i) : \mathbb{R}^{n_x} \times \mathbb{R}^{n_u} \mapsto \mathbb{R}$ is the running cost. The terminal cost $J_N(\mathbf{x}_N, k+N) : \mathbb{R}^{n_x} \mapsto \mathbb{R}$ captures the cost associated with the final state at iteration N. $\mathbf{A}(k) \in \mathbb{R}^{n_x \times n_x}$ characterizes the discrete homogeneous dynamics, and $\mathbf{B}(k) \in \mathbb{R}^{n_x \times n_u}$ represents the discrete control influence. The equality constraints are represented as $\mathbf{h}(\mathbf{x}_i, \mathbf{u}_i, k+i) = \mathbf{0} : \mathbb{R}^{n_x} \times \mathbb{R}^{n_u} \mapsto \mathbb{R}^{n_h}$ and inequality constraints as $\mathbf{g}(\mathbf{x}_i, \mathbf{u}_i, k+i) \leq \mathbf{0} : \mathbb{R}^{n_x} \times \mathbb{R}^{n_u} \mapsto \mathbb{R}^{n_g}$. The parameter N represents the horizon length, $\mathbf{x}_i$ denotes the state at iteration i, and $\mathbf{u}_i$ is the corresponding control action.

The range considered in this work usually starts at around 1 to 10 km [23], and the MPC typically employs sampling times ranging from a few minutes to a few hours, with the exact value depending on the type of approach under consideration. Approaches that converge more rapidly towards the target require a finer sampling rate to ensure adequate responsiveness, whereas slower strategies can tolerate coarser discretization, as will be demonstrated later.

The primary challenge lies in formulating MPC constraints that capture the essential characteristics of classical rendezvous approaches (such as passive abort safety, decreasing hop distances, and axis-aligned approaches) while recovering them directly from the optimization problem rather than simply tracking pre-computed trajectories.

4 Design of Proposed Approach

This work proposes a Short Range Framework (SRF), designed to take over the final phase of the rendezvous, typically initiated at distances between 1 and 10 km [13]. The framework aims to execute classical approaches from the literature, such as V-Bar and R-Bar hopping, by incorporating them directly into the MPC. This contrasts with common approaches where such approaches are performed by tracking pre-generated trajectories [13, 14]. Instead, the proposed method enables the MPC to autonomously generate the optimal trajectory by receiving user-defined parameters, including the minimum and maximum hop distances, the rate at which subsequent hops decrease, and other mission constraints.

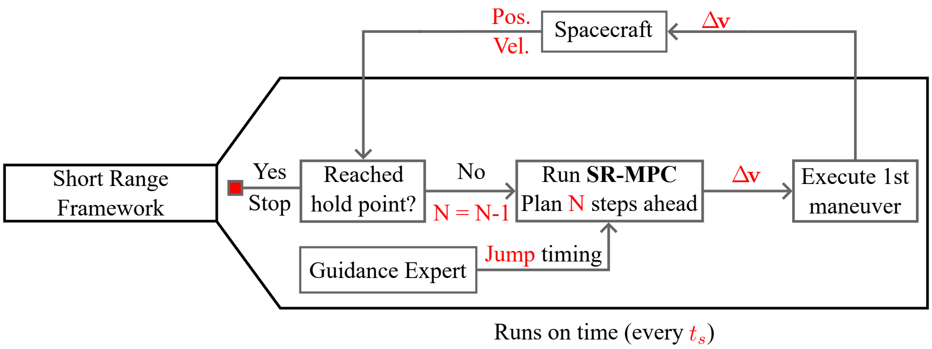


Fig. 3 Overview of the Short Range Framework

An overview of the SRF is shown in Figure 3. At its core lies the guidance expert, which in this work acts as a coordination algorithm, part of which is embedded directly in the MPC. Its role is to provide information such as the timing of hops and the initial hold point, which must be aligned with the chosen approach direction. These parameters are passed to the MPC, which then plans a trajectory over a prediction horizon of N steps, ensuring that the complete evolution from the initial condition to the endpoint is represented. The MPC then computes a sequence of control actions, though only the first input is applied at each step. The spacecraft is simulated using a nonlinear dynamical model subject to noise, J_2 perturbation, and realistic thruster limitations that impose maximum and minimum actuation bounds. An additional requirement of the framework is that the generated hops ensure passive abort safety. This property guarantees that if an approach is interrupted, the uncontrolled relative dynamics cause the chaser to drift away from the target rather than approach it unintentionally. Ideally, after a failed hop the chaser returns to its initial position, allowing a new attempt to be carried out without risk of collision.

The developments that follow build upon the closed-form solution of the CW model, presented in [24, Chapter 7.4] and shown in Equation (3), which forms the foundation for the design of the proposed approaches:

$$\begin{cases} p_{x_t} = p_{x0} + 6\left(wt - \sin(wt)\right)p_{z0} + \left(\dfrac{4\sin(wt) - 3wt}{w}\right)v_{x0} + 2\dfrac{1-\cos(wt)}{w}v_{z0} \\[2ex] p_{z_t} = \left(4 - 3\cos(wt)\right)p_{z0} + 2\dfrac{\cos(wt) - 1}{w}v_{x0} + \dfrac{\sin(wt)}{w}v_{z0} \\[2ex] v_{x_t} = 6w\left(1 - \cos(wt)\right)p_{z0} + \left(4\cos(wt) - 3\right)v_{x0} + 2\sin(wt)v_{z0} \\[2ex] v_{z_t} = 3w\sin(wt)p_{z0} - 2\sin(wt)v_{x0} + \cos(wt)v_{z0} \end{cases} \tag{3}$$

where $w = \sqrt{\frac{\mu}{a^3}}$ represent s the orbital rate, μ is the standard gravitational parameter, and a is the radius of the target body's circular orbit.

In the following discussions t denotes the time at which the chaser crosses the reference axis after completing a hop, and $2t$ denotes the time at which the chaser returns to its original position, if such is the case.

4.1 V-Bar Hopping Approach

In this section, the V-Bar hopping approach shall be studied and developed. The relevant literature on this approach with radial impulses has established that if the time required for a hop to complete, that is, for the chaser to cross the V-Bar axis again, is equal to half of the orbital period, then the trajectory will naturally return it to the same point one full orbit later, as illustrated in Figure 1 [14, 18]. The objective here was to replicate this result by manipulating (3) to verify whether the same property could be obtained analytically, serving as an already tested baseline for the more complex case of the R-Bar and glideslope hopping approaches that follow.

The first step was to impose a set of constraints on the equations. As observed in Figure 1, if the chaser's velocity at t is perpendicular to the V-Bar axis, then its trajectory will return it to the same position at a later time, which is precisely the condition that is sought. This leads to the following constraints:

$$\begin{aligned} p_{x_t} \neq p_{x0} \neq 0, \quad p_{z_t} = p_{z0} = 0 \\ v_{x_t} = v_{x0} = 0, \quad v_{z_t} \neq v_{z0} \neq 0 \end{aligned} \tag{4}$$

These constraints mean that the chaser's x position at time t is different from zero while its z position is zero, which corresponds to crossing the axis, and its velocity at that instant is purely along z, perpendicular to the axis. Under these conditions the goal is to determine the value of t that satisfies the equations. Here it is assumed that this perpendicular velocity at the crossing point is a requirement for the chaser to return to its initial position prior to the hopping maneuver.

Under these conditions the trajectory is described by:

$$\begin{cases} p_{x_t} = p_{x0} + 2\dfrac{1-\cos(wt)}{w} v_{z0} \\ 0 = \dfrac{\sin(wt)}{w} v_{z0} \\ 0 = 2\sin(wt) v_{z0} \\ v_{z_t} = \cos(wt) v_{z0} \end{cases} \tag{5}$$

From the condition $wt = n\pi$ with $n \in \mathbb{N}_0$, it follows that

$$t = \frac{n\pi}{w} \tag{6}$$

Since $w = \frac{2\pi}{t_{orb}}$, where t_{orb} is the orbital period of the target, this becomes

$$t = \frac{n\pi}{\frac{2\pi}{t_{orb}}} = \frac{n}{2} t_{orb} \tag{7}$$

This matches the result in the literature, indicating that setting $t = 0.5 t_{orb}$ yields a perpendicular velocity to the V-Bar at the crossing point, which in turn brings the chaser back to its initial position at $t = t_{orb}$.

Further simplifying, the following is obtained:

$$\begin{cases} \Delta p_x = \dfrac{4}{w} v_{z0} \\ 0 = sin(n\pi) \\ 0 = sin(n\pi) \\ v_{z_t} = -v_{z0} \end{cases} \tag{8}$$

where $\Delta p_x = p_{x_t} - p_{x0}$. This reveals that varying the initial radial thrust v_{z0} only affects the hop length along the V-Bar, while the z-velocity at t remains perpendicular and exactly opposite to its value immediately after the initial impulse. Notably, in the V-Bar case the hop distance Δp_x is independent of the hop duration, provided that $t = 0.5t_{orb}$. The chaser can therefore hop any distance within this time frame and still return to the initial point. As will be shown later, this property is unique to the V-Bar case and does not hold for other approaches.

The analysis confirms that the V-Bar hopping approach inherently returns the chaser to its initial position after one orbital period when the hop duration is set to half the orbital period, with the hop distance determined solely by the initial radial velocity.

4.2 R-Bar Hopping Approach

This section concerns the study of the R-Bar approach, where the objective is to confirm the result in [18] directly from the dynamic equations (3), in the same spirit as the analysis carried out for the V-Bar case in the previous section. If the same behavior can be reproduced analytically, then the extension to a generalized approach angle becomes more reliable. This can be regarded as an intermediate step toward the general formulation.

In [18] the authors discuss the R-Bar case and state that the hop duration must be below $0.21t_{orb}$. Unlike the V-Bar case, the condition is not given as an exact value but instead as an upper bound. This distinction arises because, the V-Bar case is exceptional, where the hop distance does not depend on the hop duration, whereas for other approach directions they are directly coupled. For the R-Bar, $0.21t_{orb}$ defines the maximum hop duration that still guarantees a return trajectory, where this value corresponds to a single hop that spans the entire distance to the desired position. Smaller values correspond to shorter hops. This property has important consequences for the MPC design. In the V-Bar case the MPC could simultaneously act as both guidance expert and controller by directly constraining the hop duration together with the hop conditions. For the R-Bar, since the hop time is linked to the distance, it would be impossible to impose hop constraints on a state whose occurrence time is not known in advance. As a result, the guidance expert must now operate as a pre-processing stage that provides the MPC with admissible hop times. It takes into account an aggressiveness parameter γ that dictates how much each hop reduces the distance relative to the previous one, and the number of hops n_{hops}. Based on these parameters the guidance expert then computes a sequence of hop times, which can then be enforced within the MPC by constraining the chaser to be at the axis crossing with perpendicular velocity at each t^k_{hop}. The generalized procedure is summarized in Algorithm 1.

Here θ is the approach angle (0° for V-Bar and 90° for R-Bar), $\mathbf{p}_0$ is the initial position of the chaser, and ω is the orbital period. The algorithm constructs a sequence of exponentially decreasing distances based on γ and uses them to determine the corresponding hop times. Step 5 is the critical part of the process, where the hop dynamics are solved to obtain the time required to cover each prescribed distance. This step will be described in detail here for the R-Bar case and generalized in the following section.

Algorithm 1 Guidance expert procedure for hop time generation in the general case

```
Inputs: $n_{hops}$, $\mathbf{p}_0$, $\gamma$, $\theta$, $\omega$
Outputs: $\mathbf{t}_{hop} = \{t^1_{hop}, \dots, t^{n_{hops}}_{hop}\}$

Construct exponentially decaying distances $\{d_0, \dots, d_{n_{hops}}\}$ from $\gamma$ and $\mathbf{p}_0$
for $k = 1, \dots, n_{hops}$ do
    Set current hop start $d_k$ and end $d_{k+1}$
    Compute $\rho = \frac{d_{k+1}}{d_k}$
    Solve hop dynamics with $\theta$, $\rho$, and $\omega$, obtain $t^k_{hop}$
end for
return $\mathbf{t}_{hop}$
```

To obtain the analytical expression for the hop time t_{hop}, a sequence of constraints analogous to those used in the V-Bar case is imposed. In this case, the constraints are:

$$\begin{aligned} p_{x_t} &= p_{x_0} = 0, \quad p_{z_t} \neq p_{z_0} \neq 0 \\ v_{x_t} &\neq v_{x_0} \neq 0, \quad v_{z_t} = v_{z_0} = 0 \end{aligned} \tag{9}$$

with the objective of determining t. These conditions correspond to enforcing that the velocity at the axis crossing is perpendicular, so that the trajectory naturally returns to the initial position, as in the V-Bar case.

Applying these constraints to the CW equations (3) yields:

$$\begin{cases} 0 = 6(wt - \sin(wt))p_{z_0} + \left(\dfrac{4\sin(wt)}{w} - 3t\right)v_{x_0} + 2\dfrac{1-\cos(wt)}{w}v_{z_0} \\ p_{z_t} = (4 - 3\cos(wt))p_{z_0} - 2\dfrac{\cos(wt)-1}{w}v_{x_0} + \dfrac{\sin(wt)}{w}v_{z_0} \\ 0 = 3w\sin(wt)p_{z_0} - 2\sin(wt)v_{x_0} + \cos(wt)v_{z_0}. \end{cases} \tag{10}$$

The hop time t is then derived as a function of the position ratio $\rho_z = p_{z_t}/p_{z_0}$. To simplify the equations, dimensionless variables are introduced as $\phi = wt$, $\alpha_z = v_{x_0}/(wp_{z_0})$, and $\beta_z = v_{z_0}/(wp_{z_0})$. Dividing through by p_{z_0} and substituting yields

$$\begin{cases} 0 = 6(\phi - \sin\phi) + (4\sin\phi - 3\phi)\alpha_z + 2(1-\cos\phi)\beta_z \\ \rho_z = 4 - 3\cos\phi - 2(\cos\phi - 1)\alpha_z + \sin\phi\beta_z \\ 0 = 3\sin\phi - 2\sin\phi\alpha_z + \cos\phi\beta_z. \end{cases} \tag{11}$$

From the third equation, β_z is obtained as $\beta_z = (2\alpha_z - 3)\tan\phi$. Substituting this relation into the first equation gives

$$0 = 6(\phi - \sin\phi) + (4\sin\phi - 3\phi)\alpha_z + 2(1-\cos\phi)(2\alpha_z - 3)\tan\phi, \tag{12}$$

which after simplification leads to

$$\alpha_z = \frac{6(\sin\phi - \phi\cos\phi)}{4\sin\phi - 3\phi\cos\phi}. \tag{13}$$

Substituting both α_z and β_z into the second equation yields

$$\rho_z = \frac{3\phi - 4\sin\phi}{3\phi\cos\phi - 4\sin\phi}. \tag{14}$$

This expression matches the one obtained in [18], since their result:

$$\rho_z = \frac{3\phi\sec\phi - 4\tan\phi}{3\phi - 4\tan\phi} \tag{15}$$

is algebraically equivalent after substituting $\sec\phi = 1/\cos\phi$ and $\tan\phi = \sin\phi/\cos\phi$. Finally, the hop time is obtained as $t = \phi/w$. If equation (14) is solved with $\rho_z = 0$, corresponding to the largest admissible hop, a numerical solution yields $\phi \approx 1.27$. This result leads to

$$t = \frac{\phi}{\omega} \implies t = \frac{\phi}{2\pi} t_{orb} \implies t \approx 0.21\, t_{orb}, \tag{16}$$

which confirms the previous finding that the maximum hop corresponds to approximately $0.21\, t_{orb}$, with smaller hops associated with proportionally smaller transfer times.

This concludes the derivation of the analytical expression that correlates the distance covered with the hop duration (14). In the upcoming section, the generalized equation shall be obtained, which allows any hopping angle approach to be executed.

4.3 Generalized Hopping Approach

In this section, the generalized approach for hopping along any approach angle is deduced. It builds on the confirmed deduction for the R-Bar case, which is closer to the generalized formulation since it relates the distance fraction to time. This formulation is conceptually similar to the R-Bar case, where the user selects the aggressiveness parameter and the number of hops, with the additional choice of the approach angle θ, as outlined in Algorithm 1. To extend the derivation to arbitrary approach directions, the original dynamical system is reconsidered, and the approach angle θ is defined with respect to the V-Bar axis. The initial distance to the desired position is denoted by $d_0 = \sqrt{p_{x_0}^2 + p_{z_0}^2}$, and the displacement fraction is defined as $\rho = \|\mathbf{p}_t\|/\|\mathbf{p}_0\|$, which represents the portion of the total displacement in the direction of the target completed in a single step.

The perpendicularity conditions, as in the other approaches, are expressed in this case through the inner product between the final position and velocity vectors:

$$\mathbf{p}_t \cdot \mathbf{v}_t = 0 \iff p_{x_t} v_{x_t} + p_{z_t} v_{z_t} = 0 \tag{17}$$

As in the case of axis-aligned hops, the objective is to describe the trajectory in terms of a fixed fraction ρ of the initial distance and determine the time t at which the constraint is satisfied. The conditions $p_{x_t} = \rho p_{x_0}$ and $p_{z_t} = \rho p_{z_0}$ enforce this requirement.

The dynamics are described using state transition matrices:

$$\begin{aligned} \mathbf{p}_t &= \mathbf{\Phi}_{pp}\mathbf{p}_0 + \mathbf{\Phi}_{pv}\mathbf{v}_0 \\ \mathbf{v}_t &= \mathbf{\Phi}_{vp}\mathbf{p}_0 + \mathbf{\Phi}_{vv}\mathbf{v}_0 \end{aligned} \tag{18}$$

where

$$\mathbf{\Phi}(\phi) = \begin{bmatrix} 1 & 6(\phi - \sin\phi) & \frac{4\sin\phi - 3\phi}{\omega} & \frac{2(1-\cos\phi)}{\omega} \\ 0 & 4 - 3\cos\phi & \frac{2(\cos\phi - 1)}{\omega} & \frac{\sin\phi}{\omega} \\ 0 & 6\omega(1-\cos\phi) & 4\cos\phi - 3 & 2\sin\phi \\ 0 & 3\omega\sin\phi & -2\sin\phi & \cos\phi \end{bmatrix} \tag{19}$$

To simplify the derivation of the analytical expressions, the coordinate frame is rotated such that one axis is aligned with the approach direction, reducing the deductions to a single-axis form. The rotated frame is defined by $e_A = \cos\theta\, e_V + \sin\theta\, e_R$ for the approach direction and $e_{\perp_2} = -\sin\theta\, e_V + \cos\theta\, e_R$ for the orthogonal direction. The corresponding rotation matrix is

$$\mathbf{R} = \begin{bmatrix} \cos\theta & \sin\theta \\ -\sin\theta & \cos\theta \end{bmatrix}, \tag{20}$$

which performs a clockwise rotation that aligns e_A with the horizontal axis. With this definition, 0° corresponds to the V-Bar and 90° to the R-Bar approach.

The dynamics in the rotated frame follow from the transformation of the state transition matrices:

$$\tilde{\mathbf{\Phi}}_{pp} = \mathbf{R}\mathbf{\Phi}_{pp}\mathbf{R}^{-1}, \quad \tilde{\mathbf{\Phi}}_{pv} = \mathbf{R}\mathbf{\Phi}_{pv}\mathbf{R}^{-1}, \quad \tilde{\mathbf{\Phi}}_{vp} = \mathbf{R}\mathbf{\Phi}_{vp}\mathbf{R}^{-1}, \quad \tilde{\mathbf{\Phi}}_{vv} = \mathbf{R}\mathbf{\Phi}_{vv}\mathbf{R}^{-1} \tag{21}$$

leading to

$$\begin{cases} r_{x_t} = \tilde{\phi}_{pp}^{11} r_{x0} + \tilde{\phi}_{pp}^{12} r_{z0} + \tilde{\phi}_{pv}^{11} w_{x0} + \tilde{\phi}_{pv}^{12} w_{z0} \\ r_{z_t} = \tilde{\phi}_{pp}^{21} r_{x0} + \tilde{\phi}_{pp}^{22} r_{z0} + \tilde{\phi}_{pv}^{21} w_{x0} + \tilde{\phi}_{pv}^{22} w_{z0} \\ w_{x_t} = \tilde{\phi}_{vp}^{11} r_{x0} + \tilde{\phi}_{vp}^{12} r_{z0} + \tilde{\phi}_{vv}^{11} w_{x0} + \tilde{\phi}_{vv}^{12} w_{z0} \\ w_{z_t} = \tilde{\phi}_{vp}^{21} r_{x0} + \tilde{\phi}_{vp}^{22} r_{z0} + \tilde{\phi}_{vv}^{21} w_{x0} + \tilde{\phi}_{vv}^{22} w_{z0} \end{cases} \tag{22}$$

Here, $\mathbf{r}$ and $\mathbf{w}$ denote position and velocity in the rotated frame. The initial position is $\mathbf{r}_0 = [d_0, 0]^\top$, while the final position is $\mathbf{r}_t = [\rho d_0, 0]^\top$, ensuring motion along the rotated axis. Passive safety requires the final velocity to be orthogonal to this direction, which is enforced by $w_{x_t} = 0$. The constraints here described applied to Equation (22) yield:

$$\begin{cases} \rho d_0 = \tilde{\phi}_{pp}^{11} d_0 + \tilde{\phi}_{pv}^{11} w_{x0} + \tilde{\phi}_{pv}^{12} w_{z0} \\ 0 = \tilde{\phi}_{pp}^{21} d_0 + \tilde{\phi}_{pv}^{21} w_{x0} + \tilde{\phi}_{pv}^{22} w_{z0} \\ 0 = \tilde{\phi}_{vp}^{11} d_0 + \tilde{\phi}_{vv}^{11} w_{x0} + \tilde{\phi}_{vv}^{12} w_{z0} \end{cases} \tag{23}$$

Solving the linear system defined by the middle and last equations provides the initial velocities in the rotated frame:

$$\begin{bmatrix} w_{x0} \\ w_{z0} \end{bmatrix} = \begin{bmatrix} \tilde{\phi}_{pv}^{21} & \tilde{\phi}_{pv}^{22} \\ \tilde{\phi}_{vv}^{11} & \tilde{\phi}_{vv}^{12} \end{bmatrix}^{-1} \begin{bmatrix} -\tilde{\phi}_{pp}^{21} d_0 \\ -\tilde{\phi}_{vp}^{11} d_0 \end{bmatrix} \tag{24}$$

and substitution into the first equation yields ρ as a function of elements of $\tilde{\mathbf{\Phi}}$, which are dependent on ϕ and θ:

$$\rho = \tilde{\phi}_{pp}^{11} + \tilde{\phi}_{pv}^{11}\left(\frac{w_{x0}}{d_0}\right) + \tilde{\phi}_{pv}^{12}\left(\frac{w_{z0}}{d_0}\right) \tag{25}$$

Beyond this point, finding a closed-form solution becomes complex, and the CasADi library [25] is therefore employed to solve the problem numerically, although other methods could be employed. Note that the 2×2 matrix in (24) is assumed nonsingular for the range of admissible values.

This concludes the derivation of the hop time t, which allows the guidance expert to employ Algorithm 1 for generating the corresponding time constraints. These constraints are then provided to the MPC, enabling it to produce the desired sequence of hops. The next section focuses on how the developments presented here can be implemented so that the MPC is capable of generating such trajectories implicitly.

It is worth noting that, even though the R-Bar and generalized approaches rely on time-based hold points, they remain valuable as they preserve the predictability required for mission coordination while still benefiting from the MPC formulation. The guidance expert defines the strategic timing, but the inner optimization ensures fuel efficiency, disturbance rejection, and constraint satisfaction in real time, which goes beyond simply tracking pre-computed reference trajectories.

4.4 Implementation into the Short Range MPC Controller

In this section, the short range MPC optimization problem is presented. The formulation incorporates a generalized form of the constraints introduced in the previous sections, such that no modification is required when switching between different approaches, other than providing the desired approach angle. The only exception is the V-Bar hopping approach, which does not rely on implicit time constraints to regulate the hop distance but instead requires direct constraints within the MPC. Additionally, a Control Barrier Function (CBF) was incorporated into the MPC control problem as a way of making sure that it never enters a user-defined KOZ.

CBFs [26–28] are functions specifically designed to ensure that the system state remains within a predefined safe set by enforcing forward invariance through appropriate control constraints, while preserving the quadratic structure of the optimization problem. The first step in defining a CBF is to select a barrier function $h(\mathbf{x})$ that is positive within the safe set, zero on its boundary, and negative outside. In this case, the following barrier function was chosen:

$$h(\mathbf{x}) = (p_x - c_x)^2 + (p_y - c_y)^2 + (p_z - c_z)^2 - r^2 \tag{26}$$

This defines a spherical KOZ of radius r centered at $\mathbf{c}$. Unlike a static geometric constraint, a CBF accounts for the system dynamics by incorporating the derivative of h into the constraint formulation. In this case, the derivative is given by:

$$\dot{h}(\mathbf{x}) = 2\big((p_x - c_x)v_x + (p_y - c_y)v_y + (p_z - c_z)v_z\big) \tag{27}$$

The safety condition is then imposed by ensuring that h does not decrease too rapidly, tuned with a constant α, which is formulated as:

$$\dot{h}(\mathbf{x}) + \alpha h(\mathbf{x}) \geq 0 \tag{28}$$

The updated MPC control problem is then expressed as:

$$\min_{\mathbf{x}_i,\mathbf{u}_i^+,\mathbf{u}_i^-} \quad \frac{1}{2}(\mathbf{p}_{t_f}-\mathbf{p}_d)^\intercal\mathbf{Q}(\mathbf{p}_{t_f}-\mathbf{p}_d)+R\sum_{i=0}^{t_f}\mathbf{1}^\intercal(\mathbf{u}_i^+ + \mathbf{u}_i^-) \tag{29a}$$

$$\text{s.t.} \quad \mathbf{x}_0 = \hat{\mathbf{x}}_k, \tag{29b}$$

$$\mathbf{x}_{i+1} = \mathbf{A}_{CW}\mathbf{x}_i + \mathbf{B}_{CW}(\mathbf{u}_i^+ - \mathbf{u}_i^-), \quad \forall i \in [0,...,t_f], \tag{29c}$$

$$\mathbf{x}_{t_f} \in \mathbb{X}_f, \tag{29d}$$

$$\mathbf{0} \le \mathbf{u}_i^+ \le u_{max}, \quad \forall i \in [0,...,t_f], \tag{29e}$$

$$\mathbf{0} \le \mathbf{u}_i^- \le u_{max}, \quad \forall i \in [0,...,t_f], \tag{29f}$$

$$\sum_{i=0}^{t_f} \|\mathbf{u}_i\|_1 \le u_{budget}, \tag{29g}$$

$$\mathbf{x}_{u_1} = \mathbf{x}_1, \tag{29h}$$

$$\mathbf{x}_{u_{i+1}} = \mathbf{A}_{CW}\mathbf{x}_{u_i}, \quad \forall i \in [1,...,t_f], \tag{29i}$$

$$\dot{h}(\mathbf{x}_{u_i}) + \alpha h(\mathbf{x}_{u_i}) \ge 0, \quad \forall i \in [1,...,t_f], \tag{29j}$$

$$\mathbf{r}_j = \mathbf{R}\mathbf{p}_j, \quad \mathbf{w}_j = \mathbf{R}\mathbf{v}_j, \quad \forall j \in [0,...,n_{hops}], \tag{29k}$$

$$-r_{tol} \le r_{z_j} \le r_{tol}, \quad \forall j \in [0,...,n_{hops}], \tag{29l}$$

$$-w_{tol} \le v_{x_j} \le w_{tol}, \quad \forall j \in [0,...,n_{hops}], \tag{29m}$$

$$h_{min} \le r_{x_j} - r_{x_{j+1}} \le h_{max}, \quad \forall j \in [1,...,n_{hops}-1], \tag{29n}$$

$$r_{x_{j+1}} - r_{x_j} \le r_{x_j} - r_{x_{j-1}} + \epsilon, \quad \forall j \in [1,...,n_{hops}-1], \tag{29o}$$

The terminal cost is expressed as the quadratic difference between the terminal position $\mathbf{p}_N \in \mathbb{R}^3$ and the desired position $\mathbf{p}_d \in \mathbb{R}^3$, both defined in the LVLH frame, and weighted by the matrix $\mathbf{Q}$. The running cost consists of the ℓ_1 norm of the control actions $\mathbf{u}_i$, weighted by the scalar R. The system dynamics are represented by the CW model matrices $\mathbf{A}_{CW}$ and $\mathbf{B}_{CW}$. The terminal state is constrained to lie in the set $\mathbb{X}_f$, which defines allowable terminal positions. The control actions are bounded in the infinity norm by u_{max} at each step, ensuring that individual inputs do not exceed specified limits, and the sum of the ℓ_1 norms of the control actions over the horizon is constrained by u_{budget}, which limits the total cumulative actuation. Finally, a shrinking horizon strategy is employed, where the index range $i \in [k,...,N-1]$ and the final time t_f decrease with each iteration. This results in one fewer iteration at each time step and modifies the effective horizon length over the course of the procedure. Note that because the ℓ_1 norm introduces non-differentiability, which is not easily handled by numerical solvers, it is expressed in terms of $\mathbf{u}^+$ and $\mathbf{u}^-$, where $\mathbf{u} = \mathbf{u}^+ - \mathbf{u}^-$ with $\mathbf{u}^+, \mathbf{u}^- \ge 0$.

Constraint (29j) corresponds to the CBF condition in Equation (28). Note that an additional state $\mathbf{x}_u$, referred to as the unforced state, is propagated within the optimization (seen in constraints (29h) and (29i)). This state evolves without control input, except for the influence of the first control action, and represents the system drift in the absence of actuation. The purpose of including $\mathbf{x}_u$ is to guarantee that if the MPC were to lose control authority in the following iteration, the system would still avoid entering the KOZ.

The constraints after constraint (29j) are introduced to recover the hopping behavior. In these constraints j denotes the iterations at which the hops occur. The parameters r_{tol} and w_{tol} enforce proximity to the axis crossing by constraining, the rotated position in the z direction and the rotated velocity perpendicular to the axis, both within tolerances that account for uncertainty. Equation (29k) performs the rotation of position and velocity so that the dynamics are aligned with the chosen approach axis, which makes the formulation applicable to arbitrary approach angles. Constraint (29l) then ensures that the rotated position lies within a tolerance band around the axis crossing at each hop, while (29m) guarantees that the rotated velocity remains nearly perpendicular to the axis. The last two conditions, (29n) and (29o), apply specifically to the V-Bar case. The first constrains the hop distances within the interval $[h_{min}, h_{max}]$, while the second enforces a decrease of hop sizes, tuned by the scalar ϵ. This parameter plays a role equivalent to the aggressiveness factor γ in Algorithm 1, and it generates the decreasing hopping sequence that defines this approach.

This concludes the implementation of the developed approaches within the SRF MPC, and in the following section an analysis of the controller will be made in both single-run and Monte Carlo simulations.

5 Testing and Validation

In this section, a deeper analysis of the performance of the developed SRF MPC is presented, serving as a validation of the results presented. For the Monte Carlo study of 50 runs the simulations include noise, J_2 perturbation, and realistic thrusters.

The initial conditions are varied by 10% in each of the LVLH coordinates of position and velocity. If a nominal value is zero, a minimum offset of 1 m for position and 0.1 m/s for velocity is applied to ensure meaningful perturbations. The orbital parameters adopted for these runs are summarized in Table 2.

Target orbit in OE, $\mathbf{x}_T(t_0)$						Chaser orbit in LVLH, $\delta\mathbf{x}_C(t_0)$					
a_T km	e_T	i_T°	Ω_T°	ω_T°	l_T°	x_C km	y_C km	z_C km	v_{x_C} m/s	v_{y_C} m/s	v_{z_C} m/s
8.1×10^3	0	0	0	0	0	$2\cos(\theta)$	0	$2\sin(\theta)$	0	0	0

Table 2 Hopping analysis initial conditions

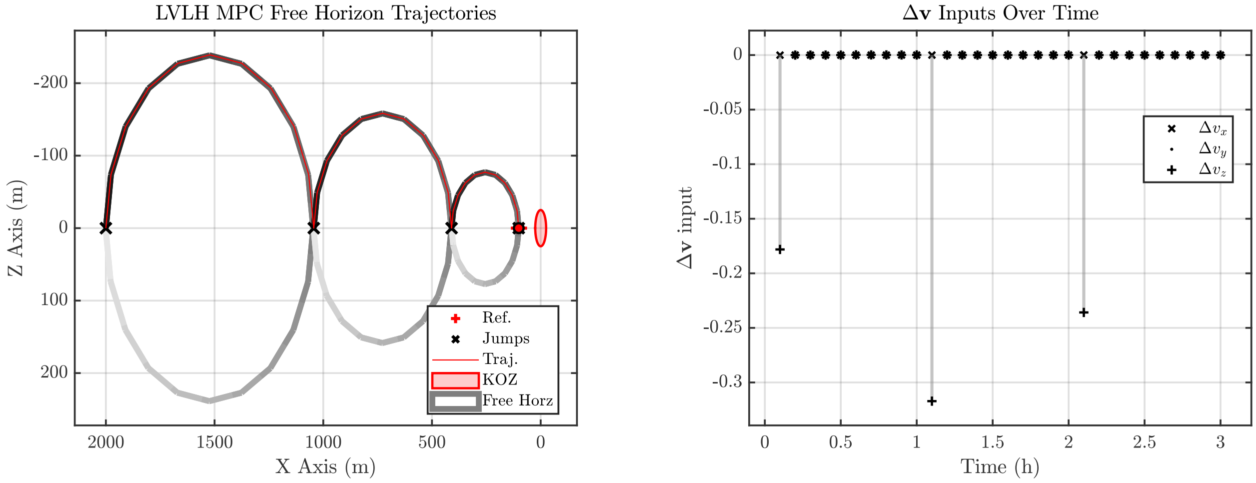
These initial conditions where chosen such that the orbital period is close to 2 h, allowing a more intuitive analysis of the plots. The chaser initial position depends on the approach angle θ so that the starting point is aligned with the approach axis at a distance of 2 km from the target. A reference point is also defined as $\delta x_{ref} = [0.1\cos(\theta), 0, 0.1\sin(\theta), 0, 0, 0]$, which in a real mission would correspond to the location at which a forced motion controller would take over for the terminal phase of the approach.

5.1 V-Bar Approach

In this section, the focus is on the V-Bar approach. In this case the trajectories are generated entirely by the MPC, with no hold points in time or position, simply respecting the constraints imposed, and with the correct sampling time to meet the timing provided by the guidance expert.

In Figure 4a the trajectory of the chaser is shown for a scenario with three hops and a nonzero value of ϵ. The thin red line represents the chaser trajectory and the black crosses mark the moments when the hops occur. The figure also displays the "Free Horz", which is the prediction of the MPC for the variable $\mathbf{x}_u$ defined in the control problem (29), representing the unforced state. This horizon indicates the path the chaser would follow if no further actuation were applied after the impulse, and it is depicted with a gradient that fades from darker to lighter as time progresses. The circles that appear at each hop illustrate that, in the absence of control authority, the chaser would return to its previous position. An interesting property of the V-Bar case is that this circular pattern is cyclic, keeping the chaser on a closed trajectory (as described in Figure 1). The KOZ is also visible in this figure, with the surrounding CBF constraint ensuring the chaser does not approach the target too closely.

Figure 4b shows the corresponding $\Delta\mathbf{v}$ usage, where it can be observed that smaller hops lead to lower fuel consumption (clearer in Figure 5b). Notably, all impulses are applied in the radial direction (z-axis), which matches the expected behavior, although no constraint was explicitly imposed in the MPC.

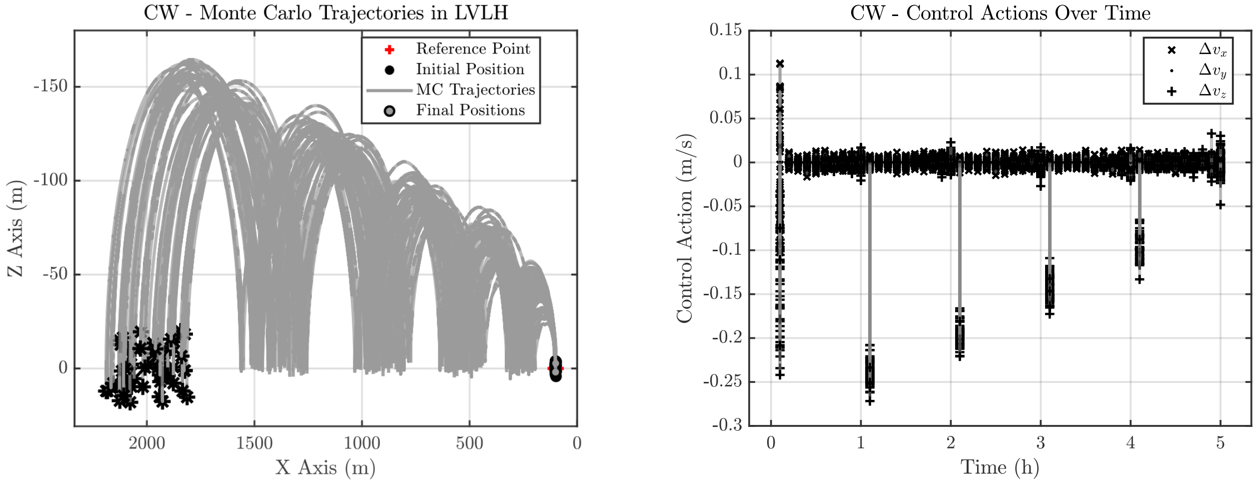


(a) 3 hops trajectory

(b) 3 hops with $\Delta\mathbf{v}$ usage

Fig. 4 V-Bar hopping approach single run with SRF MPC

This confirms that such a trajectory naturally emerges as the optimal solution for hopping safely between two points in the V-Bar. The first impulse in this case is smaller than the second, since at the beginning the relative velocity is zero, while the second hop, although shorter, must counteract the velocity in the opposite direction. Additionally, it is possible to see here the theoretical result that stated that each hop last half of the orbital period (which for these initial conditions is 2 h), as the impulses occur in 1 h intervals of each other.



(a) Trajectory

(b) $\Delta\mathbf{v}$ usage

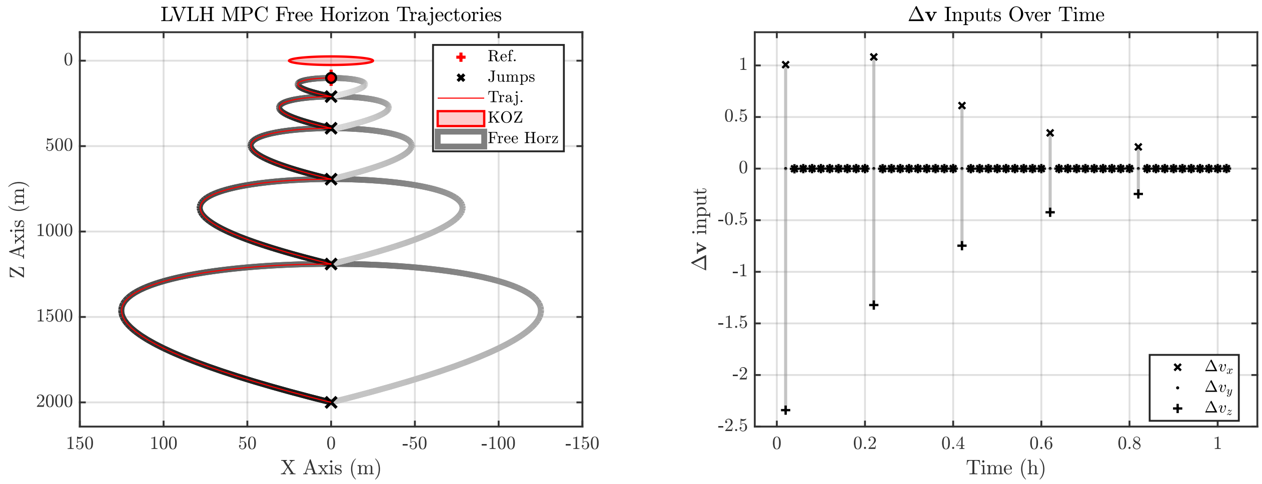
Fig. 5 V-Bar hopping Monte-Carlo with SRF MPC with 5 hops and disturbances

In Figure 5a the trajectories generated with varied initial conditions and disturbances are shown. The SRF MPC consistently produces the imposed five hop behavior, with the hop distances reduced as expected due to the aggressiveness parameter. Figure 5b shows that the $\Delta\mathbf{v}$ profile is similar to the single-run case, although impulses in the opposite direction are introduced in the first hop to compensate for the initial velocity variations of the Monte Carlo initialization.

From these results it can be concluded that the developed V-Bar hopping strategy performs as intended. It is capable of handling variations in the initial conditions as well as disturbances, which confirms its robustness and establishes it as a reliable solution.

5.2 R-Bar Approach

This section now focuses on the R-Bar approach, where a key difference emerges when compared with the V-Bar case. While in the V-Bar strategy the MPC itself determined the hop sizes, in the R-Bar case the guidance expert algorithm generates time-based hold points that dictate when the hops are executed.



(a) 5 hops trajectory **(b) 5 hops Δv usage**

Fig. 6 R-Bar hopping approach single runs with SRF MPC

Figure 6a follows the same legend naming scheme as the V-Bar case (Figure 4a), but now portrays the R-Bar approach. Each hop exhibits a heart-shaped trajectory, consistent with the behavior illustrated in Figure 1, and the hop amplitudes decrease linearly as defined by the aggressiveness parameter γ and Algorithm 1. Figure 6b shows the corresponding $\Delta\mathbf{v}$ usage, where impulses are applied along both the x and z axes, as expected, and the interval between impulses remains below the $0.21t_{orb}$ limit derived theoretically.

Overall these results confirm the observations reported in [18]. R-Bar hopping approaches complete significantly faster than the V-Bar ones (roughly five times faster), but at the cost of much higher fuel consumption (approximately ten times more). Moreover, unlike the V-Bar case, the R-Bar approach does not possess the property of indefinite stability in case of thruster failure, since the chaser does not remain in a passive holding pattern.

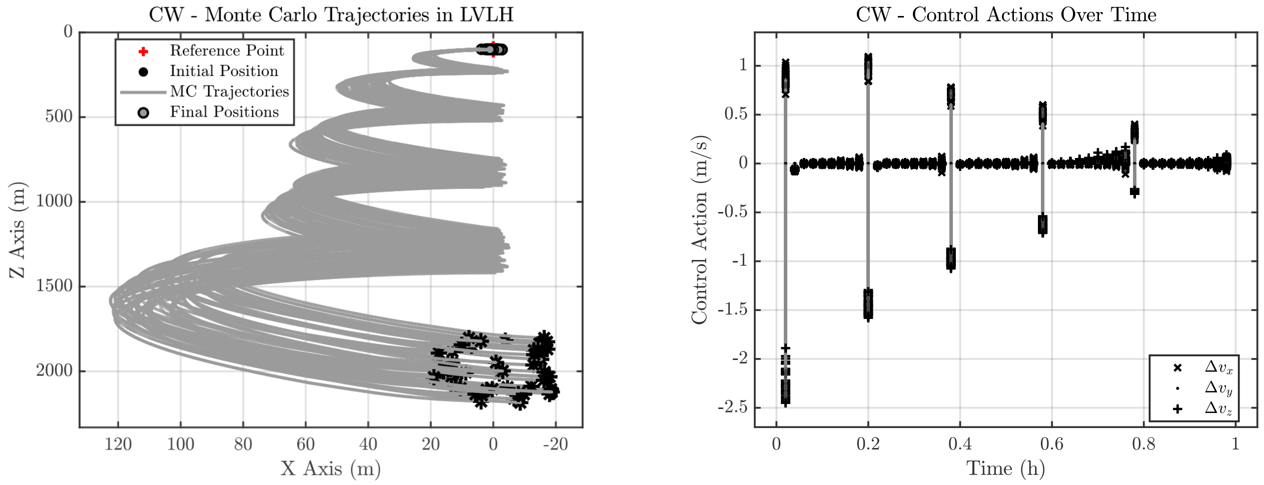
Figure 7a illustrates the performance of the controller under varied initial conditions, where it is possible to see that the MPC is able to consistently drive the chaser to the reference trajectory while executing the planned five-hop approach. Figure 7b shows a behavior analogous to the V-Bar case, where the $\Delta\mathbf{v}$ usage follows the expected profile.

As with the V-Bar analysis, the results confirm that the R-Bar implementation is correctly realized. The controller demonstrates the ability to reliably execute the approach under both nominal and disturbed conditions, with consistent performance across variations in the initial state.

5.3 Generalized Case

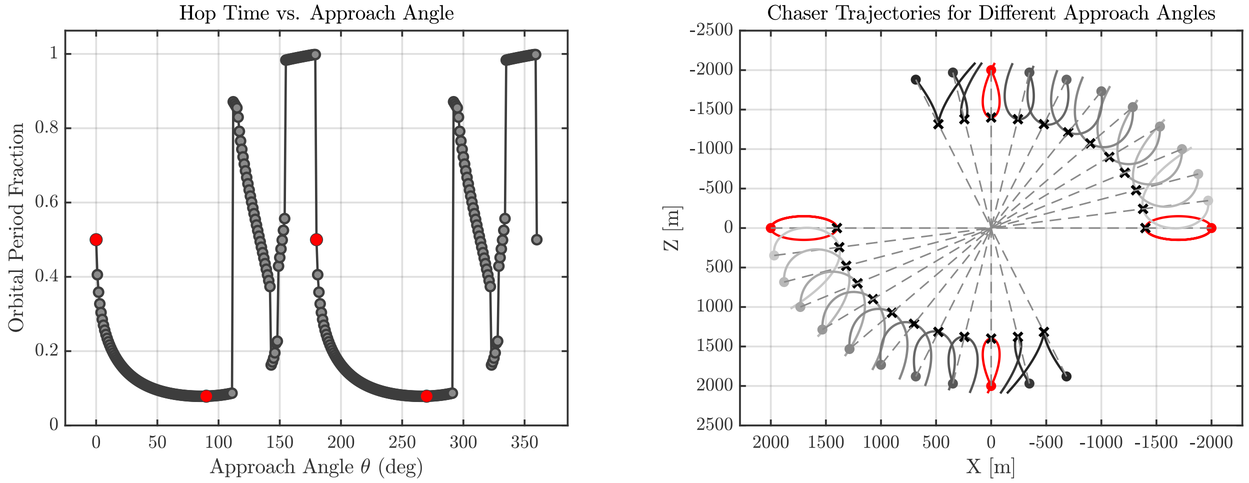
This section analyzes the generalized case and examines its behavior under different approach angles.

Figure 8 presents two different analyses of the solutions of Equation (25), which takes as input the approach angle θ and the desired position ratio ρ, and outputs ϕ, later used to determine the hop duration such that the velocity is perpendicular at the rotated axis crossing. For both analyses a value of $\rho = 0.7$



(a) Trajectory **(b) Δv usage**

Fig. 7 R-Bar hopping Monte-Carlo with SRF MPC with 5 hops and disturbances



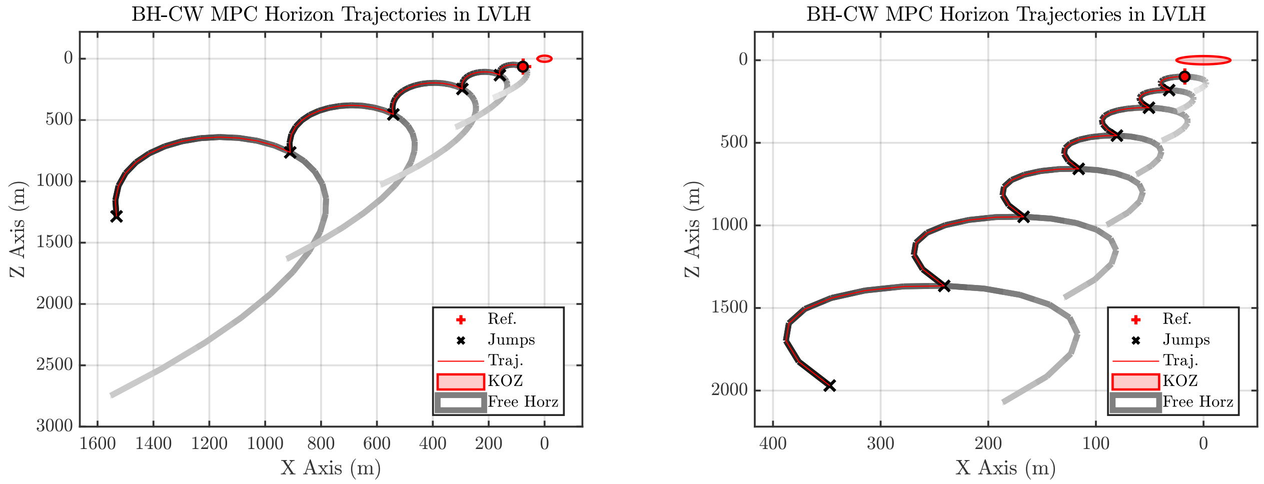
(a) Orbital period fraction sweep **(b) Single hop LVLH trajectory analysis**

Fig. 8 Generalized case hop duration expression analysis for ρ = 0.7 and different approach angles

was selected, corresponding to a hop covering 30% of the distance to the final point, with the R and V-Bar cases highlighted in red.

In Figure 8a the resulting orbital period fractions are shown for this ρ value, as the approach angle varies, where each point represents a sample of Equation (25), numerically solved using CasADi. As expected, the V-Bar case (0°) yields a period fraction of 0.5, which holds for any ρ due to its particularity discussed previously. The R-Bar case (90°) also behaves as expected, with a fraction below 0.21. Between 0 and 90° (first quadrant) the orbital period exhibits a quadratic like behavior, with the hop duration rapidly decreasing as the approach angle grows, reaching its minimum close to the R-Bar, and the starting to increase again. Beyond approximately 110°, however, the function becomes erratic, with no clear solution between 110° and 180° (covering most of the second quadrant). From 180° onwards the behavior repeats symmetrically with a period of π. To better investigate this behavior, Figure 8b shows the chaser trajectories obtained for different approach angles. The velocity values come from Equation (24), without MPC, and the trajectories are propagated using the CW model, with a single hop closing 30% of the of the distance to the final hold point. The red trajectories correspond to the V- and R-Bar cases, while dotted gray lines highlight each approach axis and black crosses mark the respective axis crossings. The known cases behave as expected, with the chaser eventually returning to its initial point after executing the maneuver. For intermediate angles, however, the chaser does not return to its initial position. This discrepancy arises because the assumption in Equation (25) that a perpendicular velocity at the axis crossing ensures precise return, only holds in the V- and R-Bar cases due to their symmetry. For intermediate angles the upper and lower parts of the approach axis experience asymmetric forces. Nevertheless, as shown in Figure 8b, even though the chaser does not return to its initial impulse point, it does return to a point further away, which indicates that when the velocity is perpendicular at the axis crossing in the first and third quadrants, the chaser still achieves passive abort safety. In contrast, after the R-Bar the chaser returns to points progressively closer to the target, breaking this behavior. Beyond 110° the problem becomes unfeasible, as trajectories cannot be generated. At 110° the chaser nearly grazes the crossing point, forming a near straight line between it and the initial point, however passive abort safety is still maintained. To take into account the second and fourth quadrants correctly, a different condition for the velocity at the crossing point would have to be used.

In the following analyses the focus will be on the first and third quadrants, where the problem remains feasible.

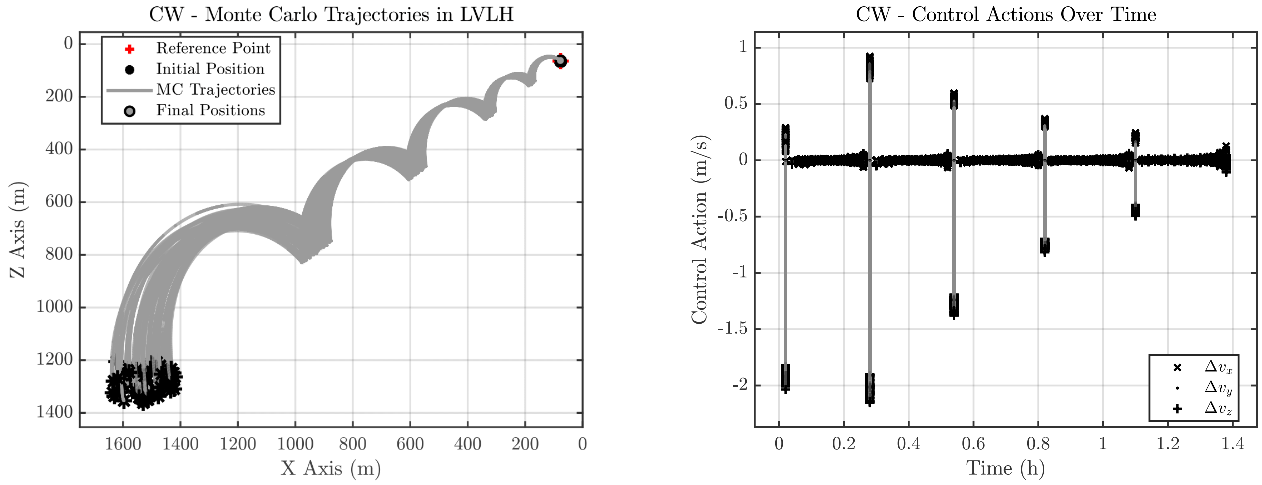


(a) 5 hops at 40° trajectory **(b) 7 hops at 80° trajectory**

Fig. 9 Glideslope hopping approach single runs with SRF MPC

Figure 9 illustrates the generalized hopping approach applied to the SRF for different approach angles and parameters. In Figure 9a the horizon of the unforced state consistently returns to a point behind the initial hopping thrust position, marked by the crosses, which confirms passive abort safety. Figure 9b shows a similar behavior, however the unforced trajectory reaches progressively shorter distances relative to the initial thrust point as the approach angle increases, indicating that the trajectory gradually closes in until the R-Bar case is reached. This observation is further supported by the results shown in Figure 8b.

The subsequent figures extend this analysis to a Monte Carlo study, allowing verification of whether these behaviors remain consistent under initial variations and disturbances.



(a) Trajectory **(b) Δv usage**

Fig. 10 Glideslope hopping at 40° **Monte-Carlo with SRF MPC with 5 hops and disturbances**

Figure 10 presents the results of a Monte Carlo run at an approach angle of 40° with disturbances. As observed in Figure 10a, the trajectories follow the expected paths, remaining within the predicted behavior, and Figure 10b shows the fuel consumption for each run. As expected, the execution time and fuel usage for the 40° approach falls between those of the R-Bar and V-Bar cases.

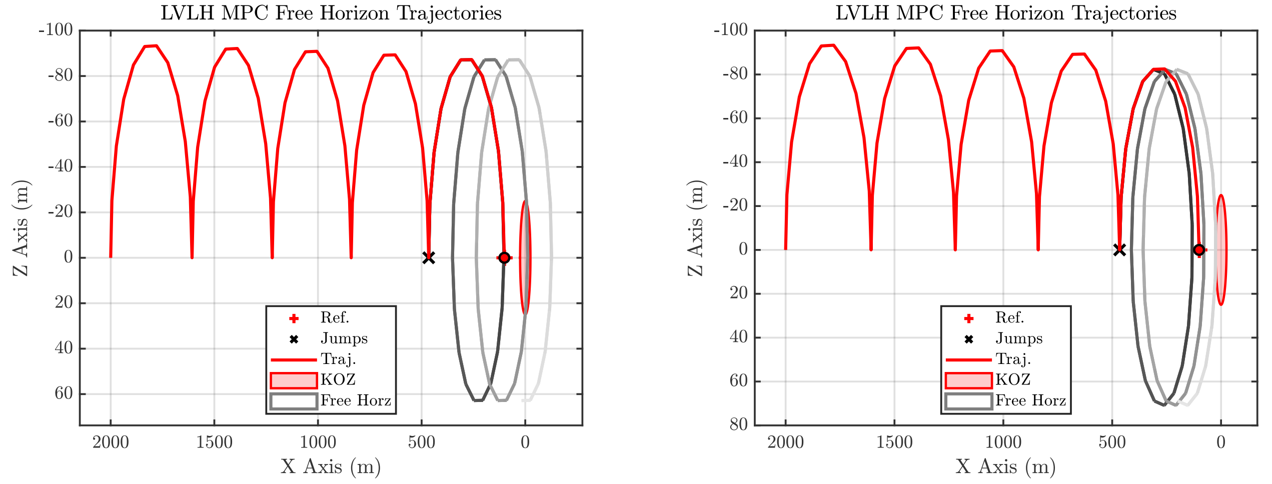
The analyses presented demonstrate that the generalized hopping approach maintains passive abort safety, both in single-run and Monte Carlo scenarios. Although the generalized approach does not return the chaser to its initial condition as initially expected, it still ensures passive abort safety. Achieving a return to the initial point would require modifying the conditions for each approach angle, which would significantly increase the complexity of the analysis. These results confirm the robustness of the approach under varying initial conditions and support its implementation within the SRF MPC.

5.4 CBF Application

In this section, the CBF implemented in the SRF MPC (29) is validated. To construct a scenario in which the KOZ is guaranteed to be violated, the V-Bar MPC conditions were modified such that the hops occurred with a $0.6t_{orb}$ interval instead of the nominal $0.5t_{orb}$ that ensures the trajectory returns to the same location. This modification causes the trajectory to slowly drift forward.

Figure 11 compares two scenarios. In the first, shown in Figure 11a, the CBF constraint in the MPC (29) is disabled, labeled as "No CBF". In this case the horizon of the unforced state $\mathbf{x}_u$ at the hop impulse intersects with the KOZ and eventually drifts into it. In the second case, shown in Figure 11b, the CBF constraint is enabled and the last impulse is adjusted so that the horizon of the unforced state never violates the KOZ.

Although the conditions had to be modified manually in this example, since the V-Bar approach is passively safe by construction, the use of the CBF remains essential. This is particularly true when large disturbances may force the SRF MPC to generate corrective maneuvers that could otherwise result in a violation of the KOZ.



(a) No CBF trajectory **(b) CBF trajectory**

Fig. 11 CBF collision avoidance example

6 Summary

This work presented the development and validation of the SRF, which handles the final phase of a rendezvous mission. The SRF MPC was designed to autonomously generate classical rendezvous approaches, such as V-Bar and R-Bar hopping, by incorporating them directly into the optimization problem through constraints derived from analytical dynamics. This approach bridges the gap between theoretical strategy design and practical implementation, allowing the MPC to produce optimal trajectories that satisfy user-defined parameters like hop aggressiveness and approach angle, while allowing the use of constraints to meet mission requirements.

The V-Bar approach has the chaser returning to its initial position after each hop if no further control is applied, and having it stay in an indefinite hold, making it highly robust to thruster failures. The R-Bar approach, while faster, consumed significantly more fuel and lacked indefinite passive safety. The generalized glideslope approach extended these concepts to arbitrary approach angles, demonstrating consistent performance across the first and third quadrants, though in these cases the chaser did not return to its initial hopping condition.

Monte Carlo simulations confirmed the robustness of the SRF MPC under both ideal and disturbed conditions, including sensor noise and J_2 perturbations. The CBF was successfully implemented to ensure strict adherence to the KOZ, providing an additional layer of safety by preventing unintended approaches to the target.

Overall, the SRF MPC effectively combines the theoretical foundations of classical rendezvous approaches with the flexibility and constraint-handling capabilities of MPC. It enables autonomous, safe, and fuel-efficient execution of final approach trajectories, adapting to a wide range of operational scenarios and requirements. The results validate the framework's performance and reliability, establishing it as a robust solution for the short-range phase of autonomous rendezvous missions.

Declaration of Use of Artificial Intelligence

Artificial intelligence was not used in the work presented.

Acknowledgements

This research was supported in part by CTS and LASI under grant CTS UID/00066/2025, and by project (1801P.01521.1.01) gEICko-101223148- IST-ID at ISR.

References

[1] S. Bennani, J. Dennehy, and J. Sellar. Looking to the future: A call to action for advanced gnc algorithm verification and validation. Technical report, NASA Technical Reports Server, 2024. `https://ntrs.nasa.gov/citations/20240003178`.

[2] D. Silvestre and P. Lourenço. Guidance and control for in-orbit servicing and assembly missions. In *Encyclopedia of Systems and Control Engineering (First Edition)*, pages 404–422. Elsevier, 2026. doi: 10.1016/B978-0-443-14081-5.00027-1.

[3] NASA Small Satellite Institute. State-of-the-art of small spacecraft technology, 2025. `https://ntrs.nasa.gov/citations/20250000142`.

[4] K. Morrison et al. Orion gn&c sequencing for off-nominal rendezvous, proximity operations, and docking. Technical report, NASA, 2021. `https://ntrs.nasa.gov/citations/20210021406`.

[5] Universe Magazine. Spacex prepares for a dangerous maneuver to refuel two starship spacecraft in space, 2024. `https://universemagazine.com/en/spacex-prepares-for-a-dangerous-maneuver-to-refuel-two-starship-spacecraft-in-space/`.

[6] T. Boge, J. D. Kezerle, and C. Dampe. Rendezvous simulation for on-orbit servicing missions. *Acta Astronautica*, 2013. doi: 10.3182/20130902-5-DE-2040.00093.

[7] eoPortal. Mev-1 & 2 (mission extension vehicle-1 and -2), 2025. `https://www.eoportal.org/satellite-missions/mev-1`.

[8] European Space Agency. Clearspace-1, 2021. `https://www.esa.int/Space_Safety/ClearSpace-1`.

[9] NASA. On-orbit satellite servicing study - project report. Technical report, NASA, 2023. `https://www.nasa.gov/wp-content/uploads/2023/10/nasa-satellite-servicing-project-report-0511.pdf`.

[10] S. Patnala and A. Abdin. On-orbit servicing for spacecraft collision avoidance with autonomous decision making. *arXiv preprint*, 2024. doi: 10.48550/arXiv.2409.17125.

[11] J. Goodman. History of space shuttle rendezvous. Technical report, United Space Alliance; Johnson Space Center, NASA, 2011. `https://ntrs.nasa.gov/citations/20110014922`.

[12] K. Yamanaka, K. Yokota, K. Yamada, H. Koyama, K. Tsukahara, S. Yoshikawa, and T. Nakamura. *Guidance and Navigation System Design of R-Bar Approach for Rendezvous and Docking*. doi: 10.2514/6.1998-1299.

[13] S. Di Cairano, H. Park, and I. Kolmanovsky. Model predictive control approach for guidance of spacecraft rendezvous and proximity maneuvering. *International Journal of Robust and Nonlinear Control*, pages 1398–1427, 2012. doi: 10.1002/rnc.2827.

[14] T. Sasaki, Y. Nakajima, and T. Yamamoto. Proximity approaches and design strategies for non-cooperative rendezvous: V-bar hopping vs. spiral approach. *Transactions of the Japan Society for Aeronautical and Space Sciences*, pages 136–146, 2021. doi: 10.2322/tjsass.64.136.

[15] E. N. Hartley. Model predictive control system design and implementation for spacecraft rendezvous. *Control Engineering Practice*, pages 695–713, 2012. doi: 10.1016/j.conengprac.2012.03.009.

[16] Pedro Taborda, Hugo Matias, Daniel Silvestre, and Pedro Lourenço. Convex mpc and thrust allocation with deadband for spacecraft rendezvous. *IEEE Control Systems Letters*, pages 1132–1137, 2024. doi: 10.1109/LCSYS.2024.3407611.

[17] J. Teles and American Astronautical Society. *Orbital Mechanics and Mission Design: Proceedings of an AAS/NASA International Symposium*. 1989.

[18] C. Wen and P. Gurfil. Guidance, navigation and control for autonomous r-bar proximity operations for geostationary satellites. *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, 2016. doi: 10.1177/0954410016638877.

[19] P. Gurfil and P. K. Seidelmann. *Celestial Mechanics and Astrodynamics: Theory and Practice*. 2016. ISBN: 978-3-662-50370-6. doi: 10.1007/978-3-662-50370-6.

[20] W. Fehse. *Automated Rendezvous and Docking of Spacecraft*. 2003. ISBN: 9780511543388. doi: 10.1017/CBO9780511543388.

[21] Y. Ariba, D. Arzelier, S. Urbina, and C. Louembet. V-bar and r-bar glideslope guidance algorithms for fixed-time rendezvous: A linear programming approach. *IFAC-PapersOnLine*, 2016. doi: 10.1016/j.ifacol.2016.09.066.

[22] W. H. Clohessy and R. S. Wiltshire. Terminal guidance system for satellite rendezvous. *Journal of the Aerospace Sciences*, pages 653–658, 1960. doi: 10.2514/8.8704.

[23] E. N. Hartley. A tutorial on model predictive control for spacecraft rendezvous. In *European Control Conference*, pages 1355–1361, 2015. doi: 10.1109/ECC.2015.7330727.

[24] H. D. Curtis. Chapter 12 - introduction to orbital perturbations. In *Orbital Mechanics for Engineering Students*, pages 651–720. 2014. ISBN: 978-0-08-102133-0.

[25] J. Andersson et al. CasADi – A software framework for nonlinear optimization and optimal control. *Mathematical Programming Computation*, pages 1–36, 2019. doi: 10.1007/s12532-018-0139-4.

[26] M. F. Reis, A. P. Aguiar, and P. Tabuada. Control barrier function-based quadratic programs introduce undesirable asymptotically stable equilibria. *IEEE Control Systems Letters*, pages 731–736, 2021. doi: 10.1109/LCSYS.2020.3004797.

[27] X. Xu, P. Tabuada, J. W. Grizzle, and A. D. Ames. Robustness of control barrier functions for safety critical control. *IFAC-PapersOnLine*, pages 54–61, 2015. doi: 10.1016/j.ifacol.2015.11.152.

[28] D. Silva and D. Silvestre. Guaranteed collision avoidance for autonomous surface vehicles equipped with a lidar using a clf-cbf-qp controller. *European Journal of Control*, page 101377, 2025. doi: https://doi.org/10.1016/j.ejcon.2025.101377.