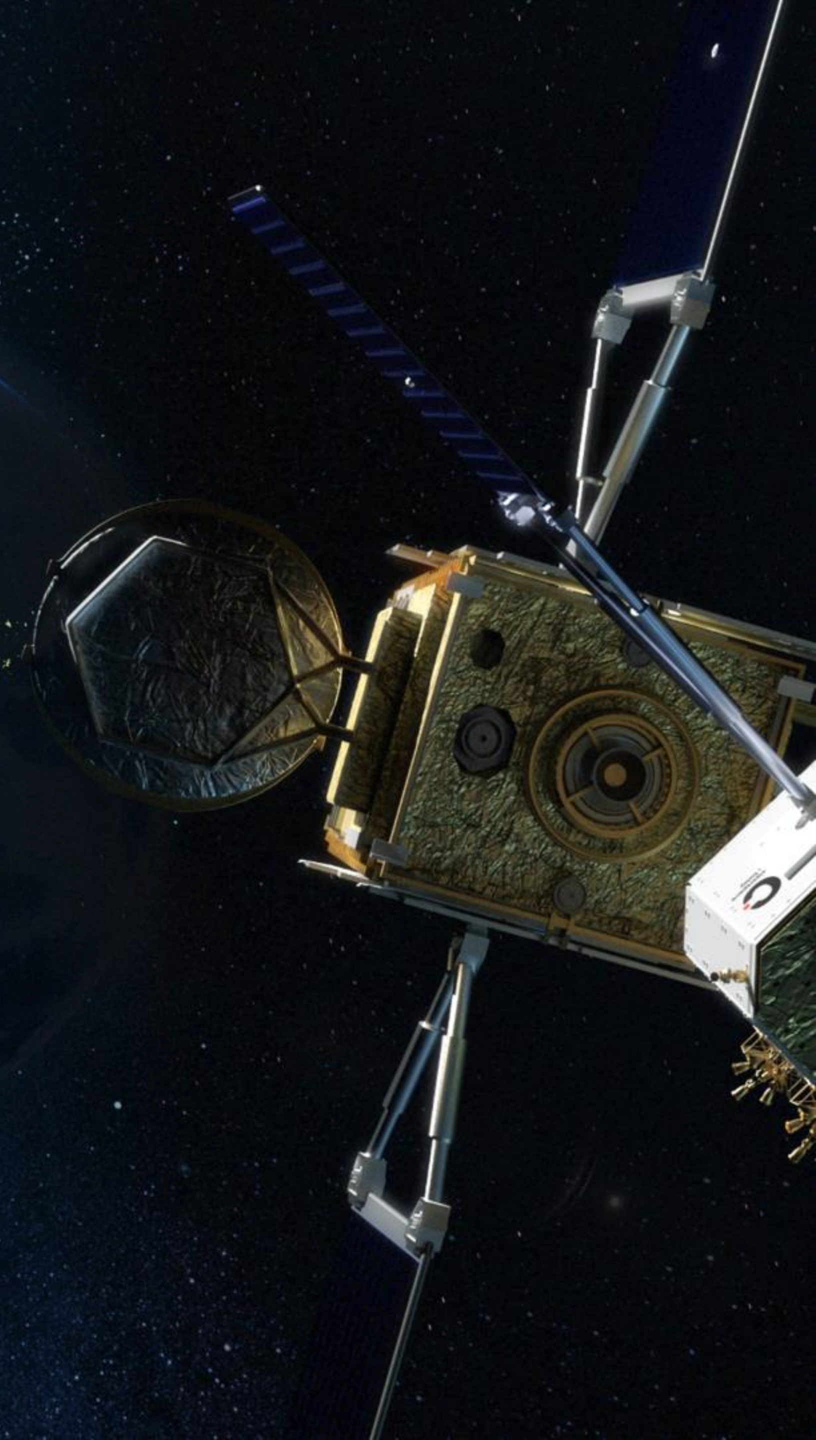
MSc in Aerospace Engineering

Design of an optimization algorithm for in-orbit inspection relative trajectories

Advisor: Prof. Elisa CAPELLO

TAS-I Tutor: Ing. Davide FABRO

Candidate: Luca LAGUARDIA



Thesis Discussion – 25th July 2024

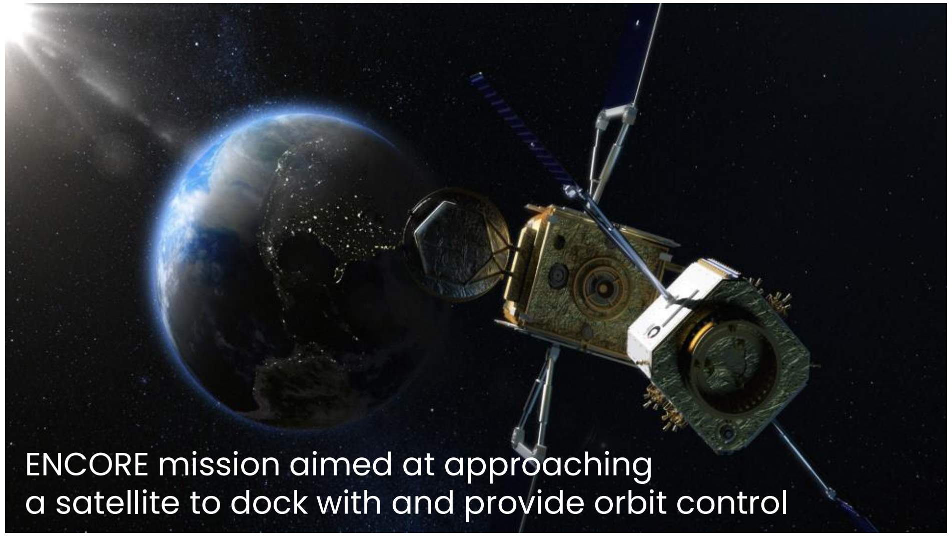
TABLE OF CONTENTS

1. INTRODUCTION – IN-ORBIT SERVICING

The term **In-Orbit Servicing (IOS)** refers to the activities aimed at extending the lifespan or enhancing the functionality of spacecraft already in orbit

It provides opportunities for:
- Refueling
- Inspection
- Adjusting orbital paths and reorienting the satellite
- Repairing and upgrading onboard instruments


ClearSpace-1 capturing the VEGA upper section for a controlled atmospheric reentry


ENCORE mission aimed at approaching a satellite to dock with and provide orbit control

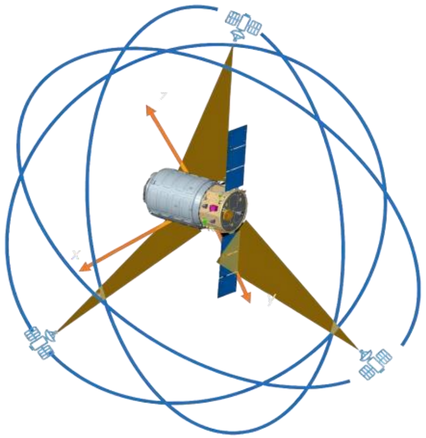
Rather than de-orbiting or replacing the spacecraft, conducting these tasks in space provides substantial economic and logistical advantages, paving the way for a new era of space utilization

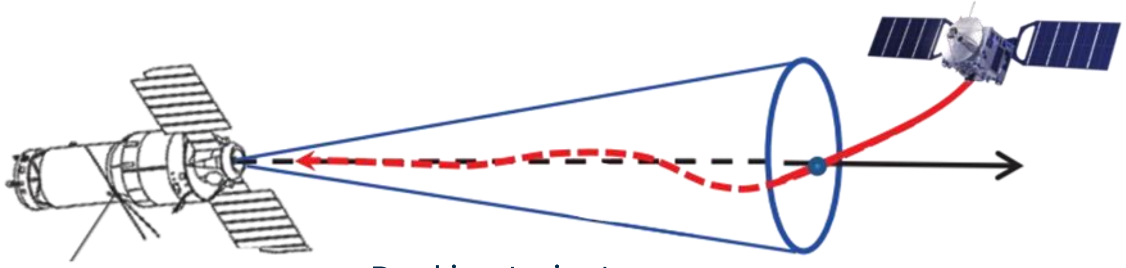
Given the complexity of IOS operations, minimizing propellant consumption plays a critical role

1. INTRODUCTION – THESIS OBJECTIVE

- Optimize relative trajectories for IOS missions, focusing on **inspection** and **docking** operations between Servicer or Chaser and Target spacecraft

Inspection trajectory





Docking trajectory

- Develope effective strategies to ensure favorable conditions for visual inspection while minimizing collision risks

- Seek potential extension of the work to the inspection of celestial bodies such as asteroids

1. INTRODUCTION – THESIS OBJECTIVE

Objective: Given a maximum time-of-fligth $\boldsymbol{t_f}$ and the number of impulses $\boldsymbol{N}$ to be applied, find the trajectory that minimizes the overall ΔV in terms of:

- $\Delta \boldsymbol{V}$ associated with each boost
- $\Delta \boldsymbol{t}$ between each boost
- drag area $\boldsymbol{A_c}$ of the Servicer

Control variables

while complying with the mission constraints

Cost function: Total mission ΔV → $J = \Delta V_{TOT} = \sum_{i=1}^{N} |\Delta V_i|$

Constraints:

- Max ΔV of each boost
- Mission duration
- Effective inspection of the Target
- Safety of the Servicer's trajectories

Constraints on the position and velocity of the Servicer

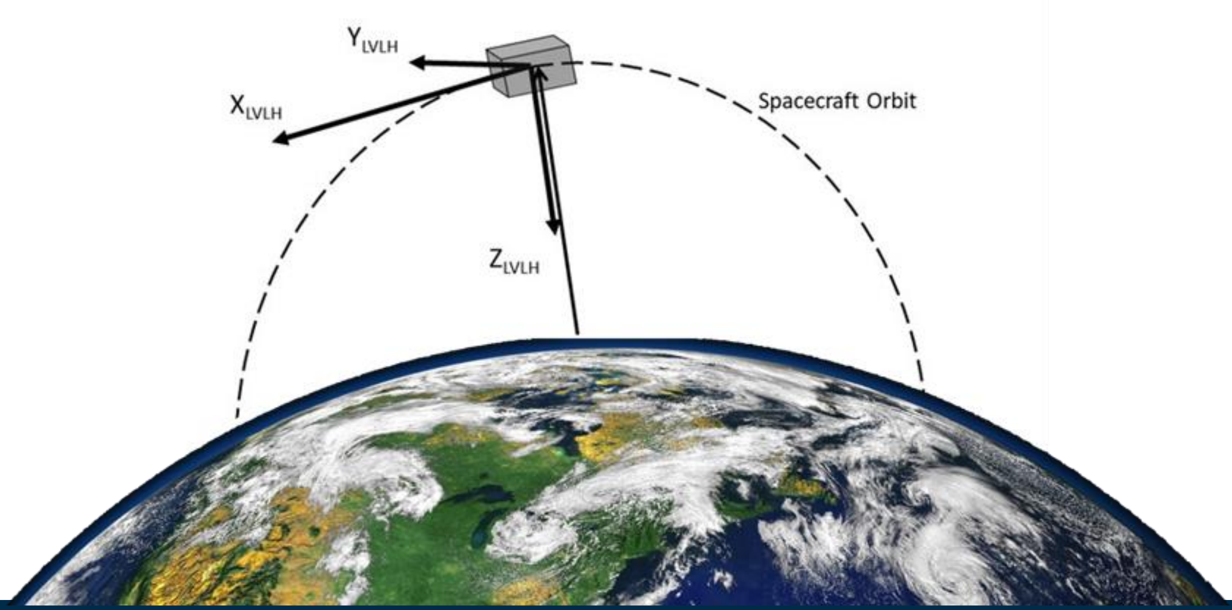
2. PHYSICAL MODEL

Impulsive model: Sudden increase in velocity ($\|\Delta V\|$>0) with zero thrust time (Δt_T = 0)

Orbit dynamics:

> Target Local Orbital Frame F_{lo} or Local-Vertical/Local-Horizontal (**LVLH**) **reference frame**

> **Hill's equations** of relative motion of two nearby orbiting objects in a circular orbit around a central body in the LVLH frame



$$\ddot{x} - 2\omega\dot{z} = \frac{1}{m_c}F_x$$

$$\ddot{y} + \omega^2 y = \frac{1}{m_c}F_y$$

$$\ddot{z} + 2\omega\dot{x} - 3\omega^2 z = \frac{1}{m_c}F_z$$

2. PHYSICAL MODEL

$$\ddot{x} - 2\omega\dot{z} = \Delta\gamma_D$$

$$\ddot{y} + \omega^2 y = 0$$

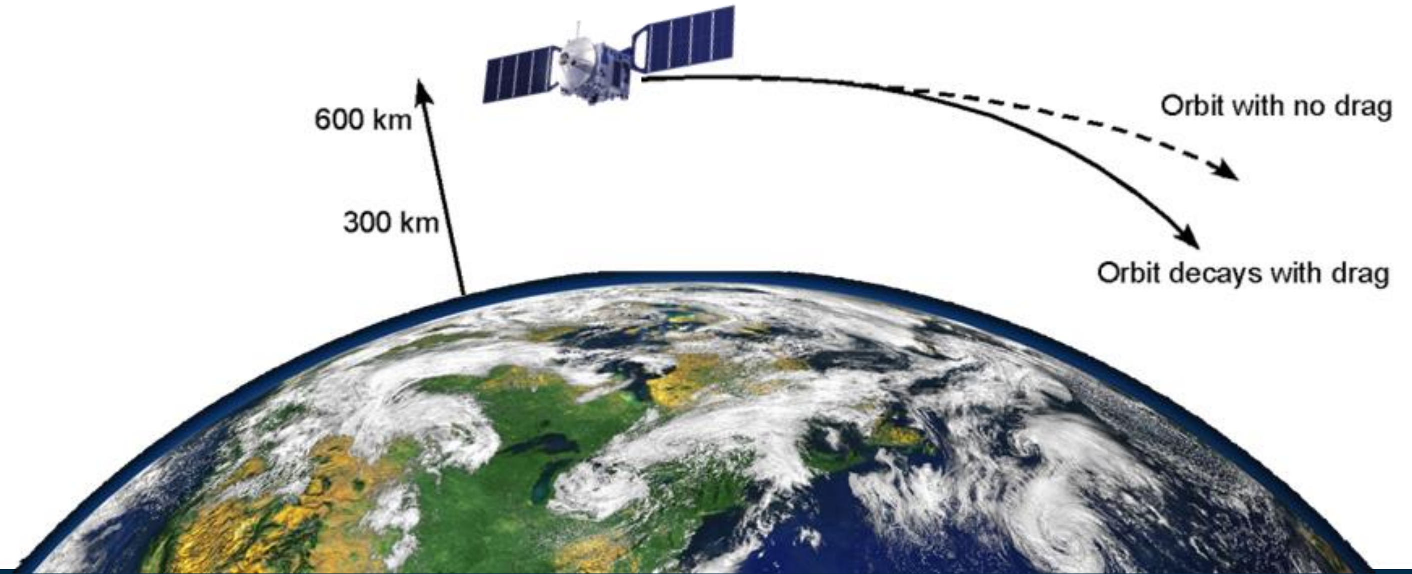
$$\ddot{z} + 2\omega\dot{x} - 3\omega^2 z = 0$$

Drag force: $F_D = -\frac{\rho}{2} V_x^2 C_D A \ [N] \rightarrow \gamma_D = \frac{1}{m} F_D \ [m/s^2]$

Differential drag acceleration:

$$\Delta\gamma_D = \gamma_{D\,c} - \gamma_{D\,t} = -\frac{\rho}{2}\omega^2 r^2 \frac{1}{C_{B_c}}\left(1 - \frac{C_{B_c}}{C_{B_t}}\right) \ [m/s^2]$$

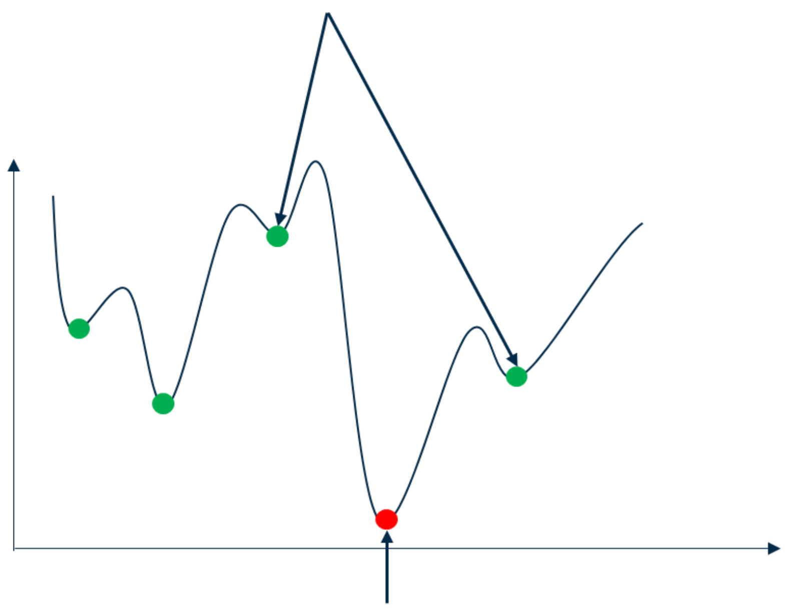
Ballistic coefficient: $C_{B_c} = \frac{m_c}{C_{D_c} A_c} \ [kg/m^2]$



3. OPTIMIZATION STRATEGY

Planning efficient and safe trajectories for IOS missions poses a complex challenge due to:

- **Non-convexity** of the problem
- Need to manage **multiple constraints**
- No guess solutions available



Adopted optimization strategies:

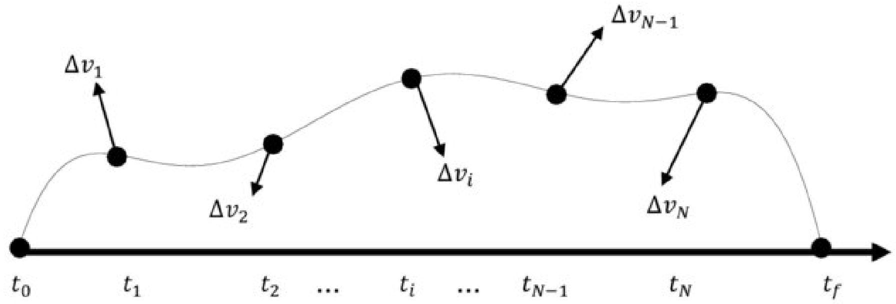
- Multiple-Shooting method
- Sequential Quadratic Programming (SQP) + Multistart algorithms

3. OPTIMIZATION STRATEGY

Multiple-Shooting Method

- Overall time horizon $[t_0, t_f]$ discretized into $\boldsymbol{N-1}$ **smaller subintervals**:

$$\boldsymbol{t_0 < t_1 < \cdots < t_i < \cdots < t_N = t_f}$$



- State values at the beginning of the subinterval and control variables unknowns to be determined in the optimization

- Dynamics satisfied by integrating the differential equations of motion with a time-marching algorithm, propagating the solution from one time instant to the next

3. OPTIMIZATION STRATEGY

Multiple-Shooting Method

SQP Algorithm (local optimizer)

- **Local optimization method** designed for solving optimization problems with non-linear objective function and constraints
- Local quadratic approximation of both the objective function and constraints, providing an effective balance between accuracy and computational efficiency
- **Solution close to an initial guess** → Global optimum not necessarily found , especially in the presence of multiple local minima

3. OPTIMIZATION STRATEGY

Multiple-Shooting Method

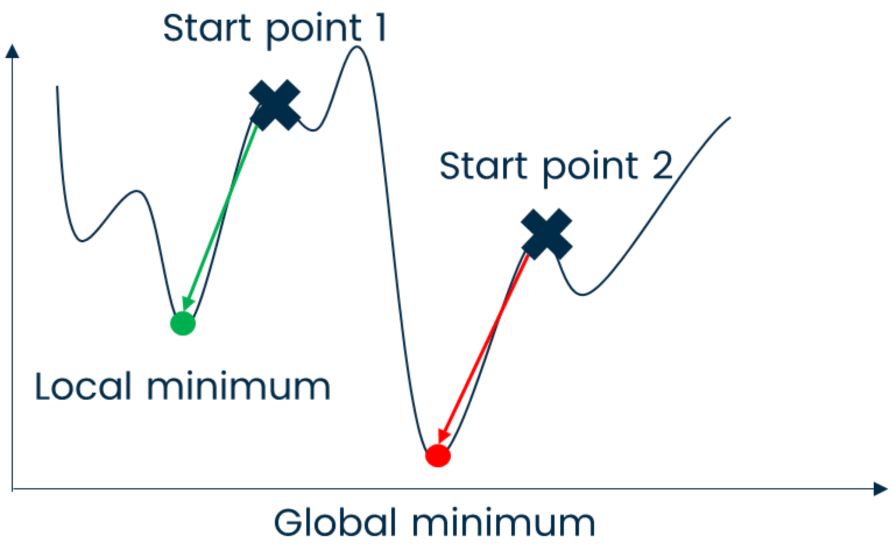
SQP Algorithm (local optimizer)

Multistart Algorithm (global optimizer)

- Local optimization algorithm run multiple times from **various initial points** in the search space

- Broader region of the solution space explored

Increased likelihood of finding a global minimum



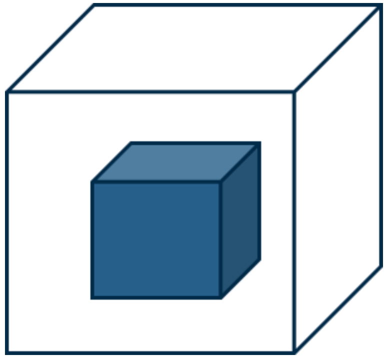
4. SIMULATION RESULTS

Optimization tool developed following a gradual approach:

- Constraints incrementally introduced to increase the complexity of the optimization problem
- Testing and validation of the code at each stage

Mission constraints

1. Outer and Inner control boxes



$\|r\| \leq$ size of the outer control box

$\|r\| \geq$ size of the inner control box

4. SIMULATION RESULTS

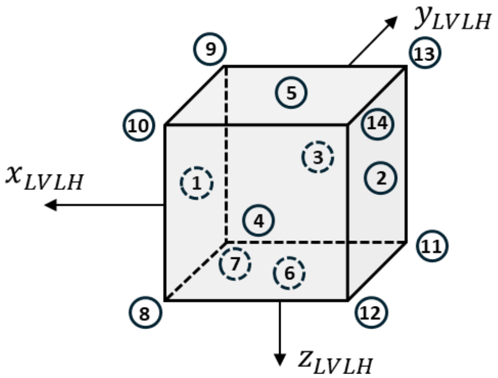
Optimization tool developed following a gradual approach:

- Constraints incrementally introduced to increase the complexity of the optimization problem
- Testing and validation of the code at each stage

Mission constraints

1. Outer and Inner control boxes

2. Inspection points



$$d_r \leq d_{observe}$$

$$\frac{(S_f - S_0) \cdot S_f}{d_r \cdot d_f} \geq cos\left(\frac{\theta_{max}}{2}\right)$$

$$\frac{(S_f - S_0) \cdot n}{d_r \cdot d_n} \geq cos(\vartheta_{occlusion})$$

4. SIMULATION RESULTS

Optimization tool developed following a gradual approach:

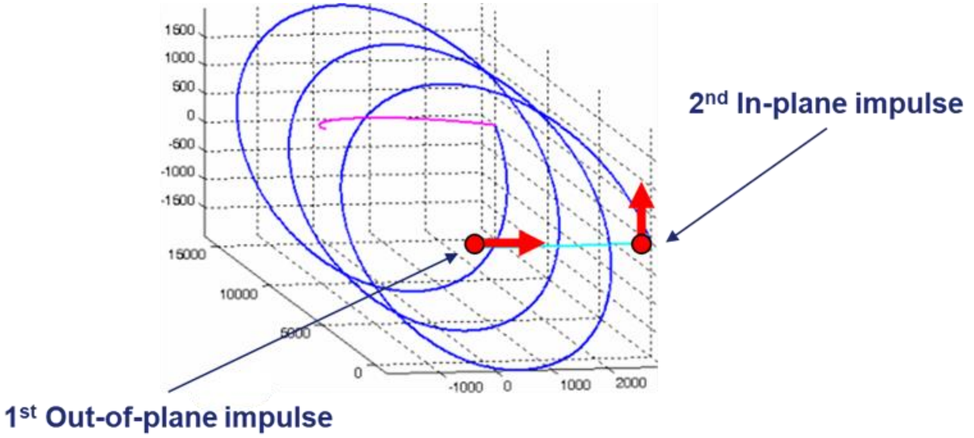
- Constraints incrementally introduced to increase the complexity of the optimization problem
- Testing and validation of the code at each stage

Mission constraints

1. Outer and Inner control boxes

2. Inspection points

3. Safety Ellipse



4. SIMULATION RESULTS

Optimization tool developed following a gradual approach:

- Constraints incrementally introduced to increase the complexity of the optimization problem
- Testing and validation of the code at each stage

Mission constraints

1. **Outer and Inner control boxes**
2. **Inspection points**
3. **Safety Ellipse**
4. **Return to starting position**

$$(X, Y, Z)_{t_f} = (X, Y, Z)_{t_0} \ [m]$$

$$\left(V_x, V_y, V_z\right)_{t_f} = \left(V_x, V_y, V_z\right)_{t_0} \ [m/s]$$

4. SIMULATION RESULTS

Optimization tool developed following a gradual approach:

- Constraints incrementally introduced to increase the complexity of the optimization problem
- Testing and validation of the code at each stage

Mission constraints

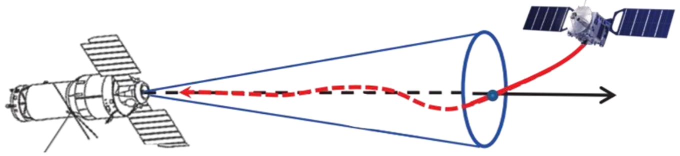
1. Outer and Inner control boxes

2. Inspection points

3. Safety Ellipse

4. Return to starting position

5. Docking

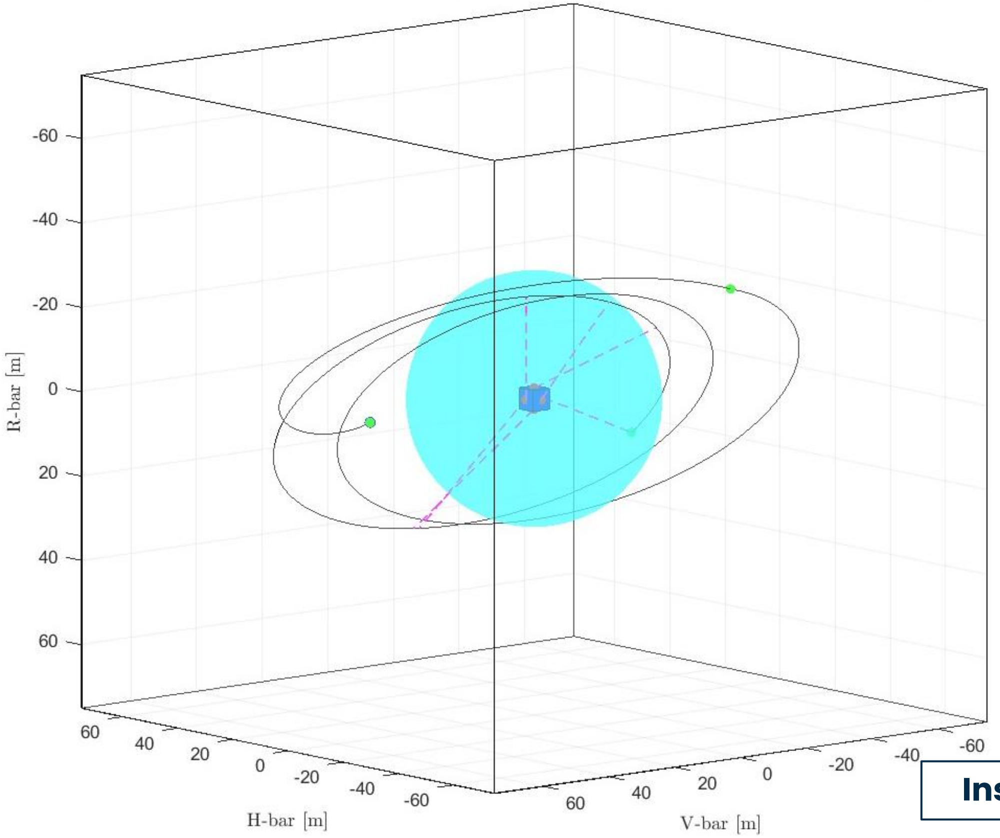


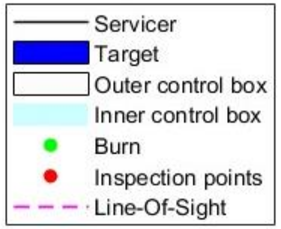
$$\frac{\left(S_f - D_0\right) \cdot c}{d_{r,app} \cdot d_c} \geq cos(\alpha)$$

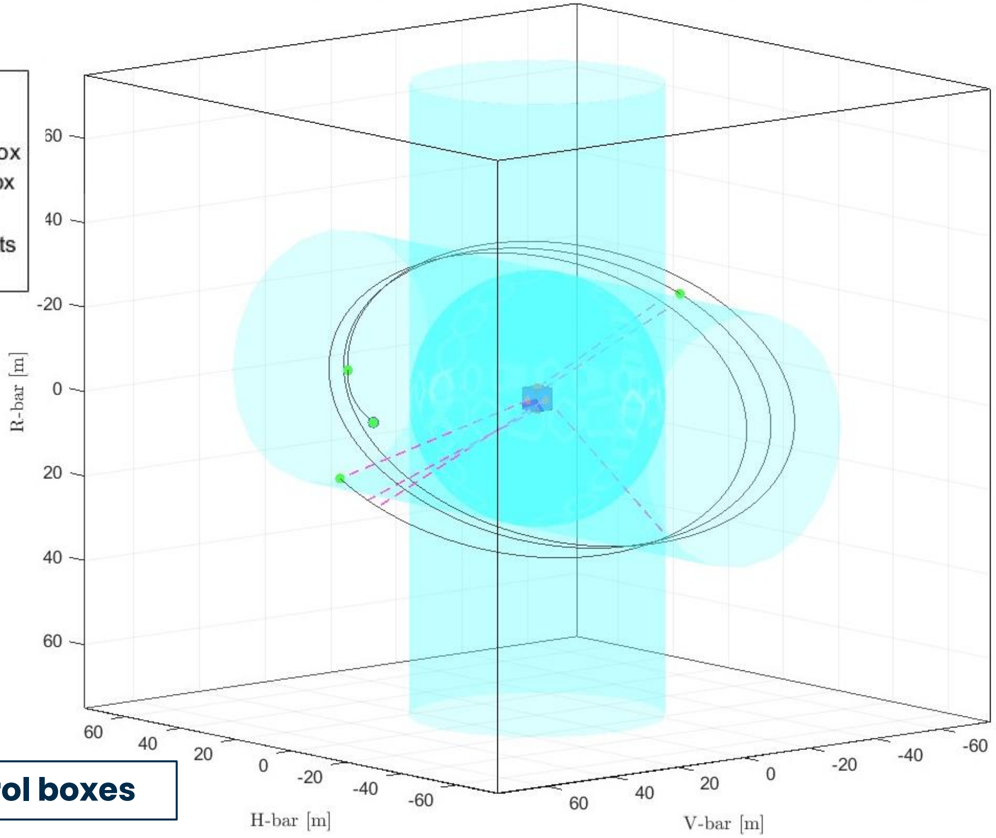
4. SIMULATION RESULTS

	Simulation values
N° of impulses	3
Drag area $[\mathrm{m}^2]$	9.581808
Time of flight $[\mathrm{s}]$	17216.989628
Total ΔV $[\mathrm{m/s}]$	**0.031272**

	Simulation values
N° of impulses	4
Drag area $[\mathrm{m}^2]$	10.356388
Time of flight $[\mathrm{s}]$	17216.989628
Total ΔV $[\mathrm{m/s}]$	**0.046871**





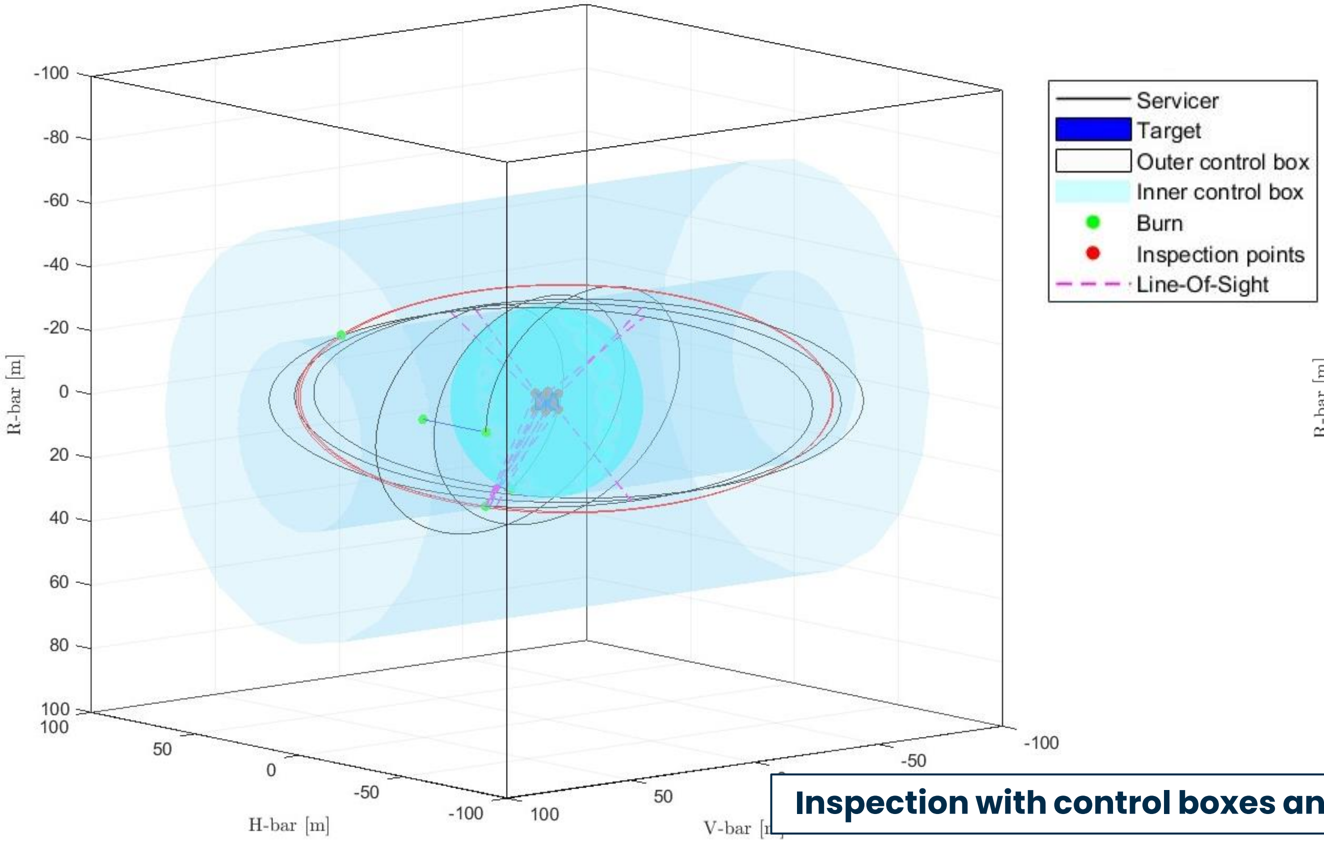


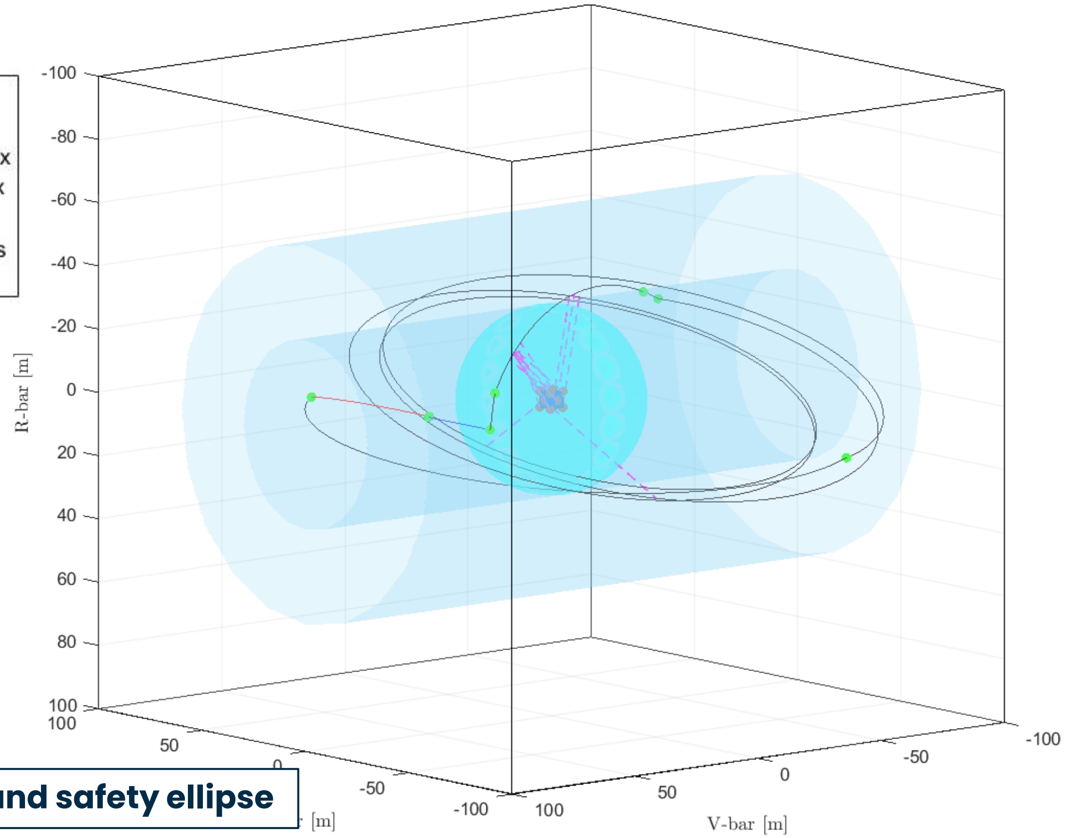
Inspection with control boxes

4. SIMULATION RESULTS

	Simulation values
Nº of impulses	5
Drag area $[\mathrm{m}^2]$	10.079848
Time of flight $[\mathrm{s}]$	46893.868751
Total ΔV $[\mathrm{m/s}]$	**0.253824**

	Simulation values
Nº of impulses	8
Drag area $[\mathrm{m}^2]$	9.157002
Time of flight $[\mathrm{s}]$	27562.069162
Total ΔV $[\mathrm{m/s}]$	**0.162885**



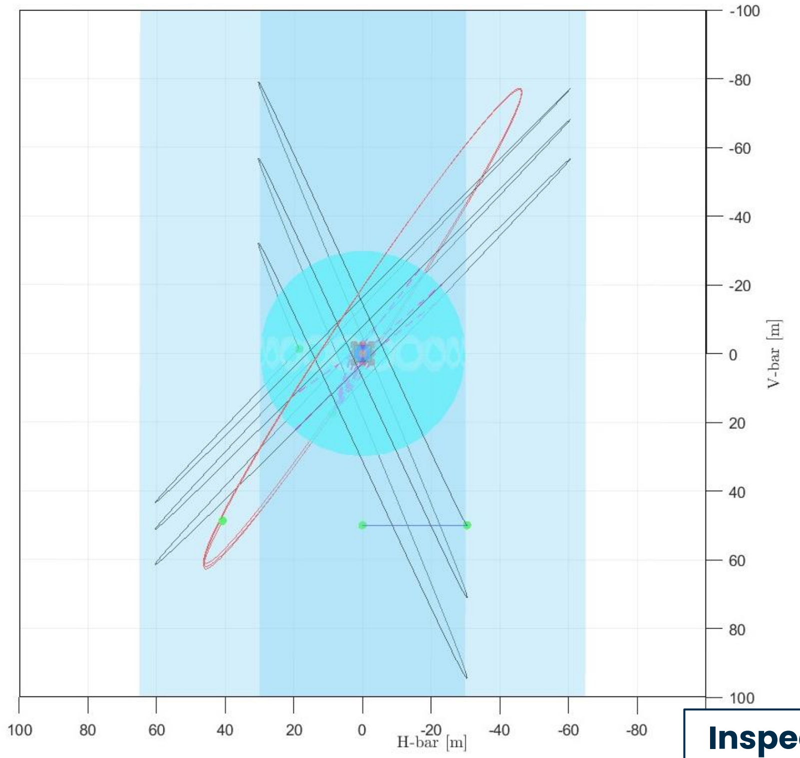


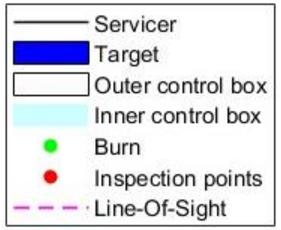
Inspection with control boxes and safety ellipse

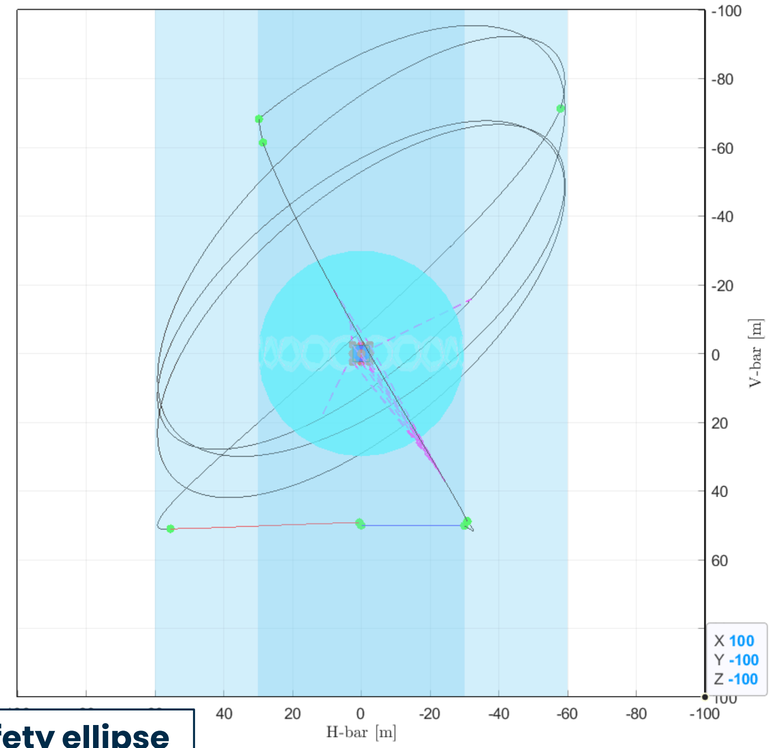
4. SIMULATION RESULTS

	Simulation values
N° of impulses	5
Drag area $[\mathrm{m}^2]$	10.079848
Time of flight $[\mathrm{s}]$	46893.868751
Total ΔV $[\mathrm{m/s}]$	**0.253824**

	Simulation values
N° of impulses	8
Drag area $[\mathrm{m}^2]$	9.157002
Time of flight $[\mathrm{s}]$	27562.069162
Total ΔV $[\mathrm{m/s}]$	**0.162885**







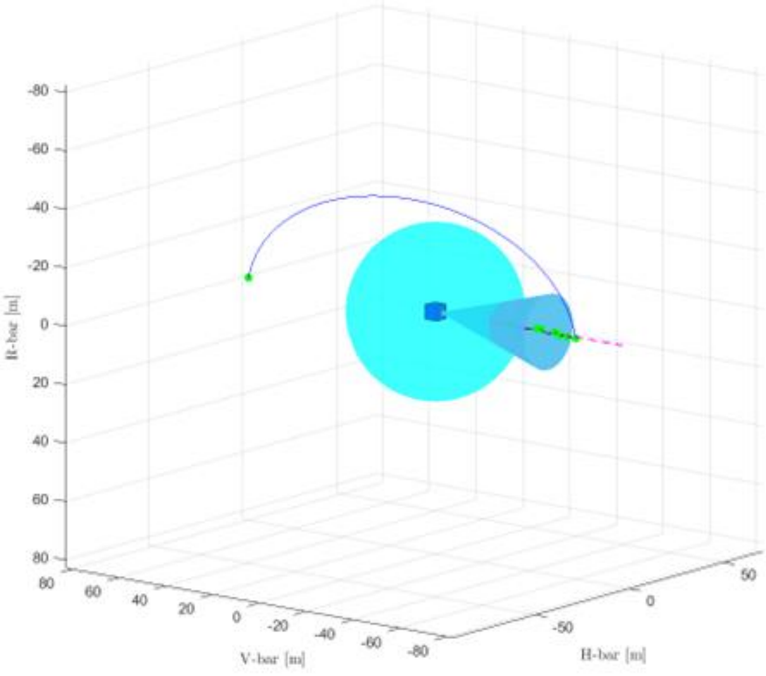
Inspection with control boxes and safety ellipse

4. SIMULATION RESULTS

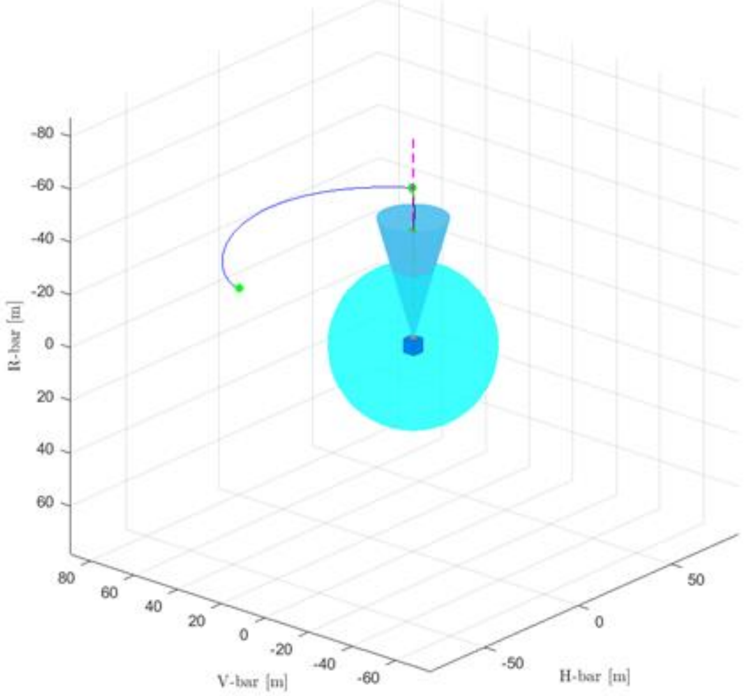
	Simulation values
N° of impulses	10
Drag area $[\mathrm{m}^2]$	9.999990
Time of flight $[\mathrm{s}]$	17216.989628
Total ΔV $[\mathrm{m/s}]$	**0.254273**

	Simulation values
N° of impulses	10
Drag area $[\mathrm{m}^2]$	4.000000
Time of flight $[\mathrm{s}]$	3096.243067
Total ΔV $[\mathrm{m/s}]$	**0.249194**

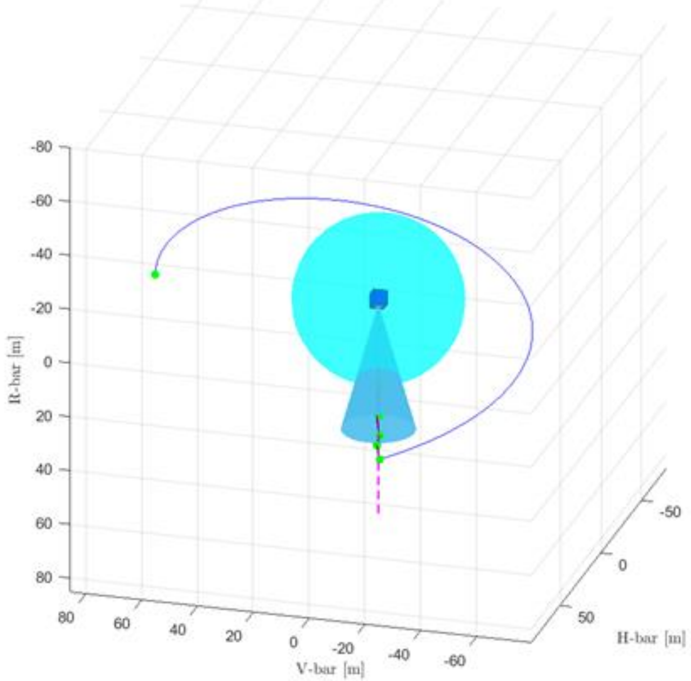
	Simulation values
N° of impulses	10
Drag area $[\mathrm{m}^2]$	9.999999
Time of flight $[\mathrm{s}]$	17216.989628
Total ΔV $[\mathrm{m/s}]$	**0.251019**



Docking along –V-bar
Docking port = [-2.5, 0, 0] m



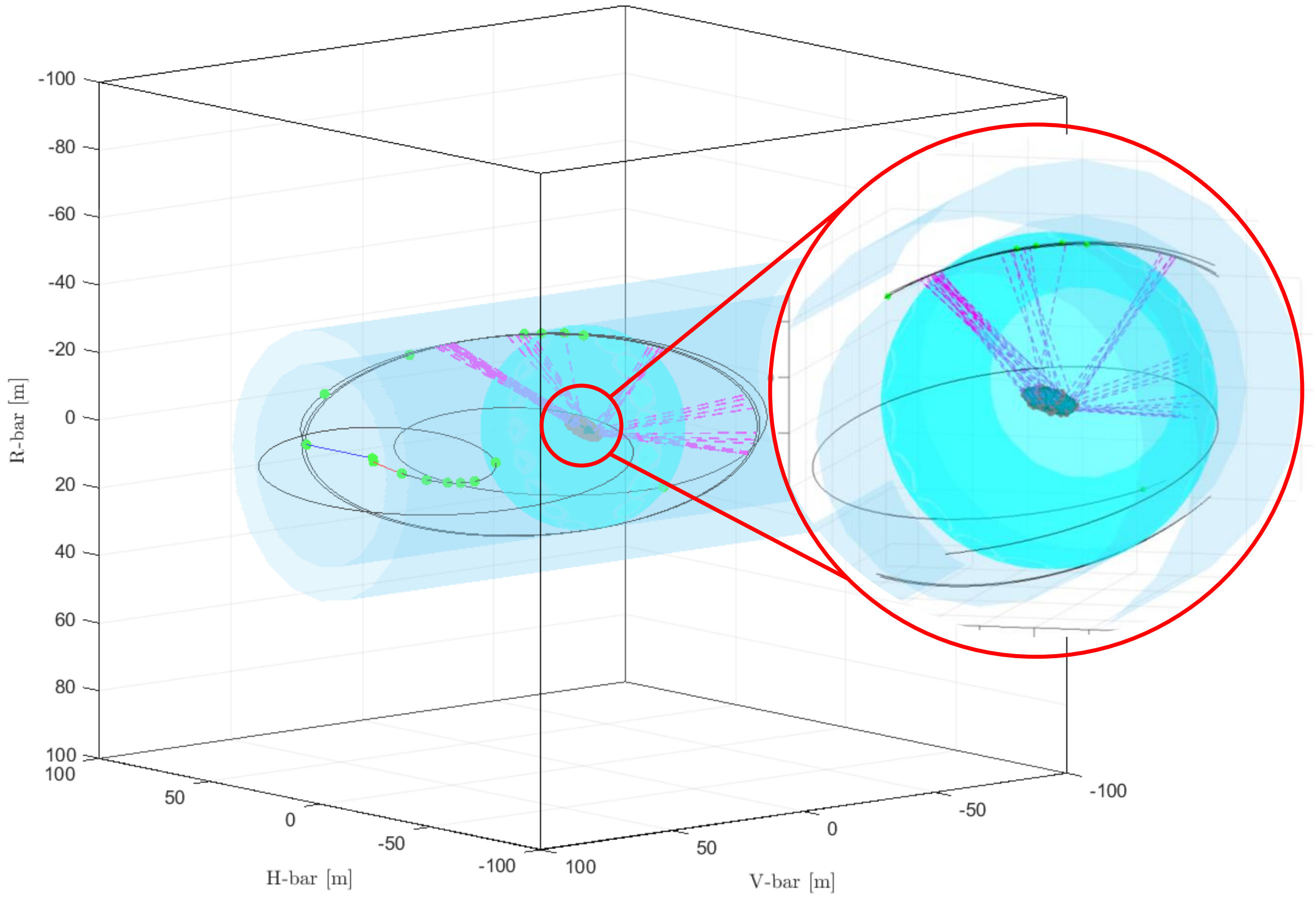
Docking along –R-bar
Docking port = [0, 0, -2.5] m



Docking along +R-bar
Docking port = [0, 0, 2.5] m

4. SIMULATION RESULTS

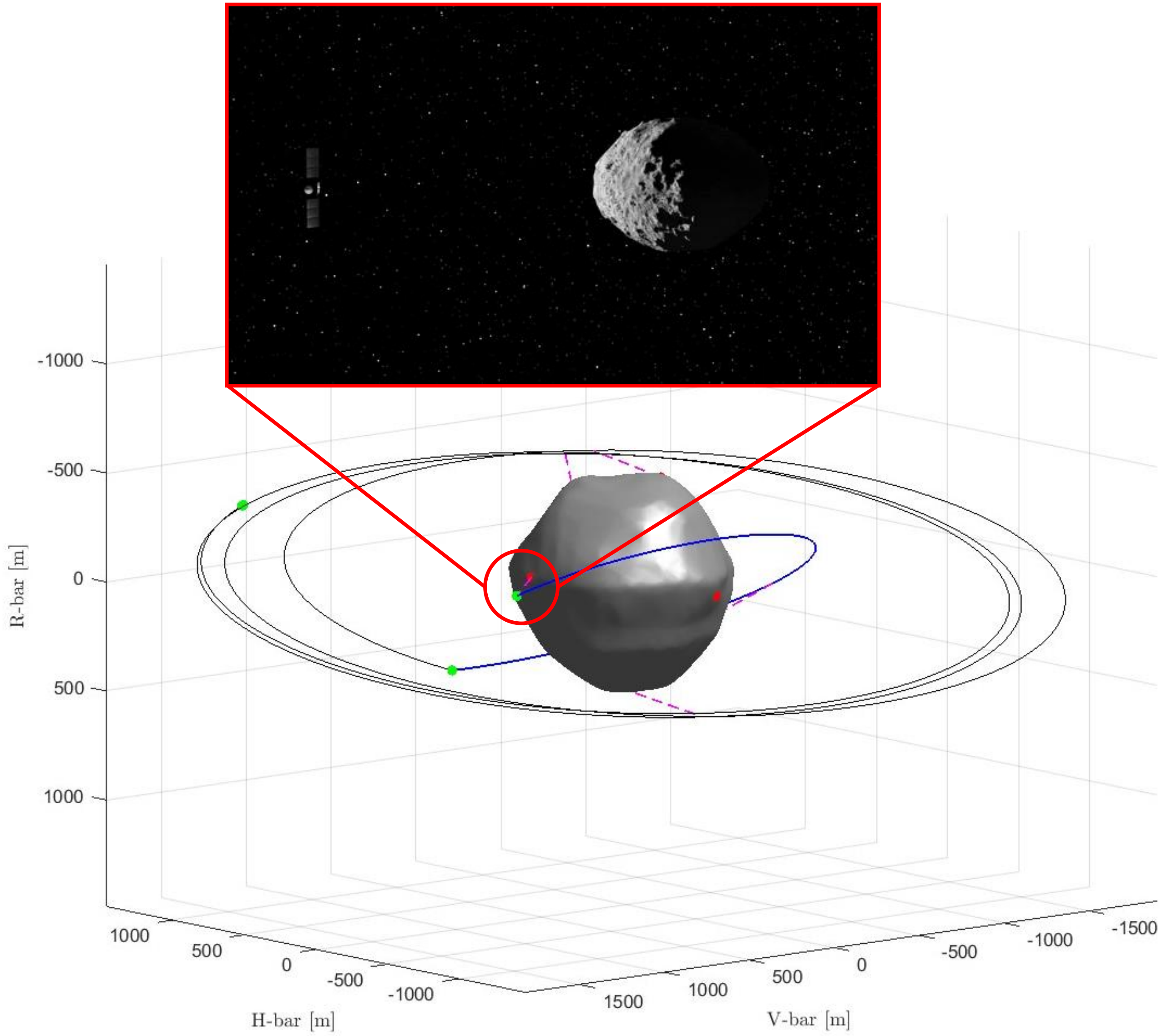
		Simulation values
Nº of impulses		16
1^{st} boost	Burn time [s]	0.000000
	ΔV [m/s]	[0.000000, 0.032420, 0.000000]
2^{nd} boost	Burn time [s]	1173.081731
	ΔV [m/s]	[0.007821, 0.001709, -0.015599]
3^{rd} boost	Burn time [s]	1908.819190
	ΔV [m/s]	[-0.006354, -0.000324, -0.004771]
4^{th} boost	Burn time [s]	3150.203717
	ΔV [m/s]	[-0.001442, -0.000267, 0.000405]
...		
16^{th} boost	Burn time [s]	28688.203720
	ΔV [m/s]	[0.000000, 0.000000, 0.000000]
Drag area $[\mathrm{m}^2]$		9.983421
Mission	Time of flight [s]	28688.203720
	Total ΔV [m/s]	**0.078210**



Inspection with control boxes and safety ellipse of an ellipsoidal Target

4. SIMULATION RESULTS

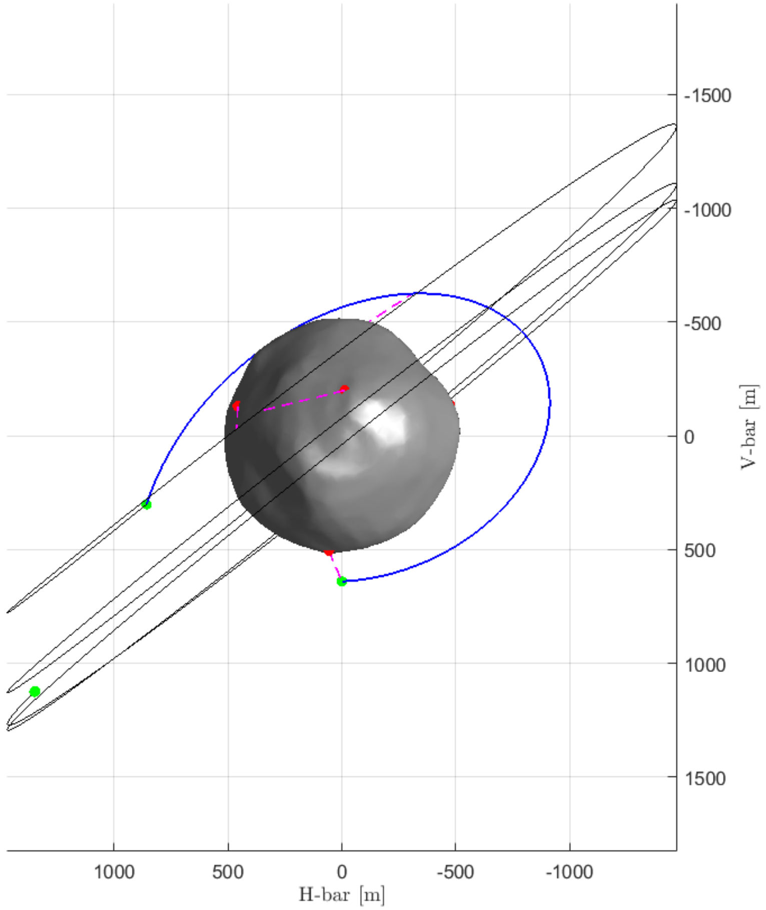
		Simulation values
N° of impulses		5
1^{st} boost	Burn time [s]	0.000000
	ΔV [m/s]	[0.641365, -0.602313, 0.585671]
2^{nd} boost	Burn time [s]	165.934536
	ΔV [m/s]	[-0.662803, 0.176192, -0.300884]
3^{rd} boost	Burn time [s]	6850.928934
	ΔV [m/s]	[-0.392839, -0.560084, -0.582511]
4^{th} boost	Burn time [s]	7372.481582
	ΔV [m/s]	[0.414169, 0.620834, -0.551894]
16^{th} boost	Burn time [s]	21458.882603
	ΔV [m/s]	[0.605800, -0.605800, -0.605800]
Drag area $[\mathrm{m}^2]$		10.820987
Mission	Time of flight [s]	21458.882603
	Total ΔV [m/s]	**4.681865**



Inspection with control boxes of an asteroid

4. SIMULATION RESULTS

		Simulation values
N° of impulses		5
1st boost	Burn time [s]	0.000000
	ΔV [m/s]	[0.641365, -0.602313, 0.585671]
2nd boost	Burn time [s]	165.934536
	ΔV [m/s]	[-0.662803, 0.176192, -0.300884]
3rd boost	Burn time [s]	6850.928934
	ΔV [m/s]	[-0.392839, -0.560084, -0.582511]
4th boost	Burn time [s]	7372.481582
	ΔV [m/s]	[0.414169, 0.620834, -0.551894]
16th boost	Burn time [s]	21458.882603
	ΔV [m/s]	[0.605800, -0.605800, -0.605800]
Drag area [m^2]		10.820987
Mission	Time of flight [s]	21458.882603
	Total ΔV [m/s]	**4.681865**



Inspection with control boxes of an asteroid

5. CONCLUSIONS AND FUTURE DEVELOPMENTS

Conclusions:

- Ability to effectively employ **direct methods** for **optimizing in-orbit inspection trajectories** in LEO
- Solid robust framework adaptable to **various operational scenarios**
- Several approaches introduced that lay the foundation for future, more in-depth studies on IOS missions

Future developments:

- **Higher-fidelity dynamics** to account for perturbations such as aerodynamic drag using an atmospheric database, Earth's magnetic field (J2 effect), and solar radiation pressure
- **Continuous-thrust model** to improve the accuracy of the Servicer's maneuvers and consequently produce more realistic optimized trajectories

THANKS FOR YOUR ATTENTION

Politecnico di Torino
1859

ThalesAlenia Space
a Thales / Leonardo company