Adaptive Model Predictive Control with online parameter learning during spacecraft proximity operations

Available online at www.sciencedirect.com

ScienceDirect

IFAC PapersOnline CONFERENCE PAPER ARCHIVE

IFAC PapersOnLine 58-16 (2024) 235–240

Adaptive Model Predictive Control with Online Parameter Learning during Spacecraft Proximity Operations

D'ORTONA Antonio, LOVAGLIO Lucrezia, STESINA Fabrizio

Politecnico di Torino, Turin, Corso Duca Degli Abruzzi 24, 10129 ITA (e-mail: antonio.dortona@polito.it).
Politecnico di Torino, Turin, Corso Duca Degli Abruzzi 24, 10129 ITA (e-mail: lucrezia.lovaglio@studenti.polito.it).
Politecnico di Torino, Turin, Corso Duca Degli Abruzzi 24, 10129 ITA (e-mail: fabrizio.stesina@polito.it).

Abstract: This paper presents a novel Model Predictive Control (MPC), developed by PhD students, that integrates online parameter learning to manage dynamic systems. In this adaptive MPC, system parameters are continuously updated using a custom gradient-based function. This function makes real-time adjustments to critical parameters based on current state errors and system feedback, enabling a dynamic adaptation to changes. Simulation are conducted on a simulator developed over the years by students of different grade Results show the effectiveness of this approach, in terms of control accuracy, robustness, and adaptability in evolving conditions, making it a promising solution for real-time control applications in complex and uncertain environments.

Keywords: model predictive control, proximity operations, dynamic systems, on-orbit servicing, rendezvous and docking, parameter learning, gradient descent, CubeSat

1. INTRODUCTION

Spacecraft rendezvous and docking (RVD) maneuvers are critical tasks for on-orbit servicing, assembly, and exploration missions. In recent years, small satellites have also been increasingly involved in such missions,(Corpino Corpino et al. (2022) (2022), (Botta Votta et al. (2022),Roscoe et al. (2018)). In these scenarios, the final approach before docking is notably challenging not only for strict safety constraints but also for the smaller size of small satellites and their docking mechanisms. Additionally, the used technology is generally less advanced than the one used for larger satellites, necessitating higher accuracy with lower-performance equipment. To improve the success of such maneuvers, several strategies have been proposed to control the relative distance between the target and the chaser. Traditionally, these approaches rely on pre-planned control strategies based on high-fidelity models. However, uncertainties in the spacecraft dynamics and the space environment can lead to performance degradation or even mission failure. To address these challenges, various algorithms have been proposed. Adaptive control laws for spacecraft rendezvous and docking are presented in Pirat et al. (2021), considering measurement uncertainty, such as aggregation of sensor calibration parameters, systematic bias, or some stochastic disturbances.Ventura et al. (2017) propose a guidance scheme for autonomous docking where the trajectory components of the controlled spacecraft are imposed by using polynomial functions determined through optimization processes. In Botta Votta et al. (2022), the authors show an optimized state-dependent MPC that integrates a pulse width pulse frequency modulation model: the results highlight a good accuracy at the final state minimizing the control efforts and approaching time. A model predictive solution is also implemented by Stesina et al. (2022) with details on the applied constraints. Pirat et al. (2019) present an H-infinity controller taking care of the robust stability and performance through the mu-synthesis. Mammarella et al. (2018) propose and validate on a test-bench a sampling-based stochastic model predictive control (SMPC) algorithm with off-line determination of the controller weights for discrete-time linear systems subject to both parametric uncertainties and additive disturbances. Stesina (2021) proposes a Tracking Model Predictive Control stressing the importance of the achievement of terminal point conditions and trajectory holding. In recent times, online learning has been used to enhance the performance of Model Predictive Control (MPC) by continuously updating model parameters and control laws in real-time. This allows MPC to effectively handle uncertainties and changing conditions. Kostadinov and Scaramuzza (2020) have proposed an online weight-adaptive approach that dynamically updates state and control weights, enabling MPC to dynamically respond to new data.

This paper presents an innovative approach that integrates Model Predictive Control (MPC) with online learning to enhance the robustness of Rendezvous and Docking (RVD) maneuvers, particularly for small satellite missions. In the context of RVD for small satellites, MPC can be employed to guide a chaser spacecraft (e.g. a 12U CubeSat) towards

10.1016/j.ifacol.2024.08.492

a target spacecraft while ensuring safety and fuel efficiency. The introduction of online learning enables the MPC controller to dynamically adapt its model of the CubeSat's dynamics in real-time, based on sensor data collected during the RVD maneuver. This continuous learning process allows the controller to compensate for uncertainties and improve performance over time, highlighting the advanced AI aspects of the strategy.

The structure of this paper is organized as follows. Section 2 details the orbital model, control strategy, and online learning adaptation. Section 3 presents numerical evaluation of the proposed control approach. It presents simulation results and discusses performance metrics to assess the efficacy of the adaptive MPC strategy. Lastly, Section 4, concludes the paper by summarizing the key findings and contributions of the presented study.

2. METHODS

The relative motion between two spacecraft is described by a system of nonlinear differential equations. Assuming that the orbit of the chaser is circular and the orbit of the target is just slightly elliptic or inclined with respect to it, the motion of the two spacecraft looks very similar and the system of equations can be simplified, obtaining the linear Hill-Clohessy-Wiltshire equations (1960) described in 1.

$$\begin{aligned} \ddot{x} - 3\omega^2 x - 2\omega\dot{y} &= \frac{F_x}{m_c} \\ \ddot{y} + 2\omega\dot{x} &= \frac{F_y}{m_c} \\ \ddot{z} + \omega^2 z &= \frac{F_z}{m_c} \end{aligned} \tag{1}$$

$\omega = \sqrt{\mu/R^3}$ represents the mean angular motion of the target spacecraft and F_x, F_y, F_z are the components of the external force vector (excluding gravity) acting on the spacecraft, composed by the thrust generated by the propulsion system and the forces due to the interaction of the external environment. The equations are expressed with respect to the local orbiting reference frame RIC (Radial InTrack CrossTrack) centered on the target CoM (Center of Mass). The RIC reference system consists of a local tangent plane coordinate system that is defined with respect to the target spacecraft. The origin coincides with the target center of mass. The radial axis X_{RIC} is aligned with the radial direction (positive outwards). The InTrack Y_{RIC} axis coincides with the direction of the target's motion. The CrossTrack Z_{RIC} completes the right-hand coordinates system and it is parallel to the orbit angular momentum vector (positive in the orbit normal direction). The frame is shown in Fig. 1.
The relative dynamics described in 1 can be rewritten as the following state space model:

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u} + \boldsymbol{\delta} \tag{2}$$

where $\mathbf{x} = [x, y, z, \dot{x}, \dot{y}, \dot{z}]$ is the state vector, $\mathbf{A} \in \mathbb{R}^{6\times6}$ is the state (or system) matrix, $\mathbf{B} \in \mathbb{R}^{6\times3}$ is the input matrix and can be written as follows:

$$\mathbf{A} = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 3\omega^2 & 0 & 0 & 0 & 2\omega & 0 \\ 0 & 0 & 0 & -2\omega & 0 & 0 \\ 0 & 0 & -\omega^2 & 0 & 0 & 0 \end{bmatrix} \tag{3}$$

$$\mathbf{B} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 1/m_c & 0 & 0 \\ 0 & 1/m_c & 0 \\ 0 & 0 & 1/m_c \end{bmatrix} \tag{4}$$

$\mathbf{u} \in \mathbb{R}^3$ is the control input associated to the activation of the thrusters calculated with a Model Predictive Control (MPC) discussed in 2.2, and $\boldsymbol{\delta} \in \mathbb{R}^6$ represent the disturbances entering the system that consider environmental effects and model uncertainties. Particularly, the considered ones can be broadly categorized into disturbance forces and disturbance torques. The disturbance forces include aerodynamic drag due to the residual atmosphere, geopotential anomalies (specifically J2 perturbations), and differential drag. The disturbance torques encompass aerodynamic drag-induced torques, gravitational gradient torques resulting from the uneven gravitational pull on different parts of the spacecraft and residual magnetic dipole torques caused by interactions between the spacecraft's magnetic properties and the Earth's magnetic field.

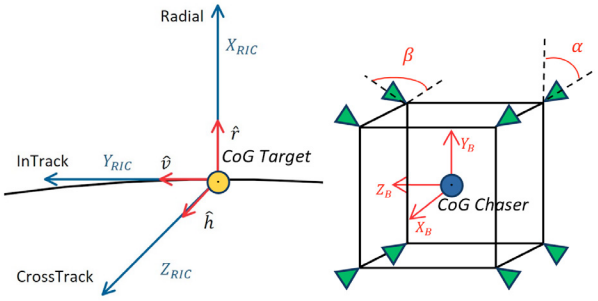

Fig. 1. RIC reference system and chaser configuration

2.1 Actuator Models

The propulsion module is based on 8 cold gas thrusters using ON-OFF-control as shown in Fig. 1. The configuration provides a 6DoF (Degree of Freedom) maneuvrability by means indipendently controlled nozzles. As depicted in Fig. 1 each nozzle axis is inclined with respect to the body reference frame of the angle α with the X_BY_B plane of the body reference frame, β represents the angle between the Z_B axis and the direction of the angle α. The thrust generated by the single nozzle can be calculated in the body reference system as:

$$\begin{aligned} t_x &= t\cos\alpha\sin\beta \\ t_y &= t\sin\alpha\sin\beta \\ t_z &= t\sin\beta \end{aligned} \tag{5}$$

in which t is the thrust generated by each nozzle. In consideration of the spatial arrangement within the propulsion

module, the control force components among the eight thrusters is delineated as follows:

Table 1. Thrust components in body axis

Thruster n.	Thrust components		
Thruster 1	t_x	t_y	t_z
Thruster 2	$-t_x$	t_y	t_z
Thruster 3	t_x	$-t_y$	t_z
Thruster 4	$-t_x$	$-t_y$	t_z
Thruster 5	t_x	t_y	$-t_z$
Thruster 6	$-t_x$	t_y	$-t_z$
Thruster 7	t_x	$-t_y$	$-t_z$
Thruster 8	$-t_x$	$-t_y$	$-t_z$

The Equation 4 can be rewritten considering the thruster configuration:

$$\dot{x} = Ax + BL_{RIC\_B}\, T^T u \tag{6}$$

where $L_{RIC\_B} \in \mathbb{R}^{3\times3}$ is the rotation matrix from body to RIC reference frame, T is:

$$\mathbf{T} = \begin{bmatrix} t_x & t_y & t_z \\ -t_x & t_y & t_z \\ t_x & -t_y & t_z \\ -t_x & -t_y & t_z \\ t_x & t_y & -t_z \\ -t_x & t_y & -t_z \\ t_x & -t_y & -t_z \\ -t_x & -t_y & -t_z \end{bmatrix} \tag{7}$$

The components of $u \in \mathbb{R}^{8\times1}$ can be either 0 or 1 based on the thruster activation.

2.2 Model Predictive Control formulation

The considered Model Predictive Control solves a finite-horizon optimization problem. At a time instant t the following optimization is solved:

$$u = \min_{u:[0,N]\to\mathbb{R}^m} J(x,u) \tag{8}$$

The objective is to minimize the cost function J with respect to the control input u over a prediction horizon N, subject to state-input constraints and state only constraints. The quadratic cost function J is:

$$J(x,u) = \Delta x_N^T P \Delta x_N + \sum_{k=1}^{N-1} \Delta x_k^T Q \Delta x_k + u_k^T R u_k \tag{9}$$

where $Q \in \mathbb{R}^{n\times n}$ and $R \in \mathbb{R}^{m\times m}$ are the state and control weight matrices, P is the state weight matrix for the $N-th$ step calculated as the solution of the discrete algebraic Riccati equation since it guarantees closed-loop stability. $\Delta x = x_{r,k} - x_k$ is the difference between the reference state and the spacecraft state at the $k-th$ step. Considering $x_k = [p_{R,k}\, p_{I,k}\, p_{C,k}\, v_{R,k}\, v_{I,k}\, v_{C,k}]^T$ with $p_{i,k}$ the relative position and $v_{i,k}$ the relative velocity in RIC reference frame (the subscript i stands for the corresponding Radial, Intrack or CrossTrack component), the entire problem can be rewritten as:

$$\begin{aligned} u = \min_u \Bigg(& \Delta x_N^T P \Delta x_N + \sum_{k=1}^{N-1} \Delta x_k^T Q \Delta x_k + u_k^T R u_k \Bigg) \\ \text{subject to} \quad & 1.\, x_{k,k=1} = \tilde{x}_{k,k=1} \\ & 2.\, x_{k+1} = x_k \tilde{A} + \tilde{B} u_k \\ & 3.\, \sqrt{p_{R,k}^2 + p_{C,k}^2} < p_{I,k} \tan(\theta/2) \\ & 4.\, v_{I,k} < 0.1 \\ & 5.\, 0 < u_k < 1 \end{aligned} \tag{10}$$

with τ representing the MPC sampling time. The constraint 10.1 sets the initial condition for the prediction by setting $x_{k,k=1}$ to match the latest estimate of the spacecraft state $\tilde{x}_{k,k=1}$. The state and control predictions are subject to the system model in constraint 10.2 through the forward Euler method. Considering

$$\tilde{A} = I + \tau A \tag{11}$$

with $I \in \mathbb{R}^6 \times 6$ the identity matrix, and

$$\tilde{B} = \tau B L_{RIC\_B} T^T \tag{12}$$

and

$$F = \begin{bmatrix} \tilde{A} \\ \tilde{A}^2 \\ \tilde{A}^3 \\ \vdots \\ \tilde{A}^N \end{bmatrix} \quad G = \begin{bmatrix} \tilde{B} & 0 & 0 & \dots & 0 \\ \tilde{A}\tilde{B} & B & 0 & \dots & 0 \\ \tilde{A}^2\tilde{B} & \tilde{A}\tilde{B} & \tilde{B} & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ \tilde{A}^{N-1}\tilde{B} & \tilde{A}^{N-2}\tilde{B} & \dots & \tilde{A}\tilde{B} & \tilde{B} \end{bmatrix} \quad u = \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ \vdots \\ u_N \end{bmatrix}$$

$$r = \begin{bmatrix} r_1 \\ r_2 \\ r_3 \\ \vdots \\ r_N \end{bmatrix} \quad \tilde{Q} = \begin{bmatrix} Q & 0 & \dots & 0 \\ 0 & Q & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \dots & P \end{bmatrix} \quad \tilde{R} = \begin{bmatrix} R & 0 & \dots & 0 \\ 0 & R & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \dots & R \end{bmatrix}$$

Equation 9 can be rewritten in matrix form:

$$J = (Fx_{k,k=1} + Gu - r)^T \tilde{Q} (Fx_{k,k=1} + Gu - r) + u^T R u \tag{13}$$

where r represents the reference trajectory to be followed during the Rendezvous approach. The constraint 10.3 defines a cone-shaped region that the chaser must stay within for safety reasons, ensuring it does not cross the designated boundaries as shown in Fig. 2, where the chaser is represented as a green square and the target as a black dot.

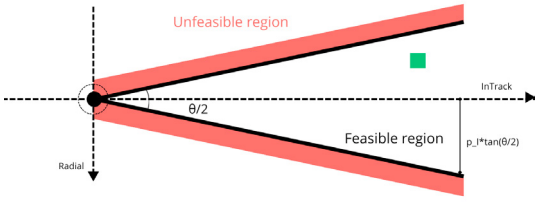

Fig. 2. Control loop architecture

The constraint 10.4 limits the maximum velocity to meet mission requirements in this phase of the final approach. Constraint 10.5 ensures the control input stays within the range that spans from 0 to 1 to comply with the thruster's ON-OFF functionality.

2.3 Online Parameter Update

The study introduces an adaptive control algorithm designed to dynamically update key parameters based on current state errors and system inputs. In particular, the focus is on the minimization of a loss function defined as

$$\mathcal{L} = \frac{1}{2}\, x_{\text{err}}^T\, x_{\text{err}} \tag{14}$$

with

$$x_{\text{err}} = x_{\text{est}} - x_{\text{pr}} \tag{15}$$

representing the squared error between the estimated state x_{est}, provided by the Extended Kalman Filter (EKF), and the predicted one, x_{pr} computed by the Model Predictive Control (MPC) and defined as

$$x_{\text{pr}} = \tilde{A} x_{\text{k-1}} + \tilde{B} u_{\text{k-1}} \tag{16}$$

To minimize this function, two main varying parameters are taken into account: the spacecraft's mass, here defined as m_c, and the thrust, here defined as t, generated by the thrusters. During the mission, the spacecraft's mass decreases due to fuel consumption, while the thrust varies with the changing pressure in the fuel tank, affecting nozzle thrust. Considering Equation 16, a modification to the chaser's mass induces an equivalent effect on the system dynamics as an opposite modification to the maximum thrust, and vice versa. To efficiently incorporate the influence of both force and mass, the update strategy utilizes the derivative of the loss function with respect to w, where w represents the thrust-to-mass ratio t/m_c.

To minimize this function, *gradient descent* approach is used:

- **Initialization:** both thrust t and mass m_c values are set up from previous iterations. If first iteration is considered, the initial values are taken into account;
- **Gradient calculation:** The gradient of the loss function is computed with respect to the change of w;
- **Parameter update:** the direction along which the loss function decreases the most is the negative gradient. In order to minimize it, the value of w is updated by taking a small step in the negative gradient direction, setting them a little close to the minimum.

$$w_{\text{k+1}} = w_{\text{k}} - \eta\alpha \left(\frac{\partial \mathcal{L}}{\partial w}\right)_{\text{k}} \tag{17}$$

where η can be regulated depending on how small the step is decided to be taken. In the present approach, η is equal to 0.1.

The chosen loss function is quadratic, and therefore convex. This implies that the gradient changes linearly from one point to another, providing a smooth and predictable gradient descent path, and that the local minimum is also the global minimum. For each iteration, the gradient of $\mathcal{L}$ with respect to w is calculated as:

$$\begin{aligned} x_{err} &= x_{est} - \left(\tilde{A} x_{k-1} + \frac{t}{m_c}\begin{bmatrix}0^{3\times3}\\ I^{3\times3}\end{bmatrix} L_{RIC\_B} T_{norm}^T u_{k-1}\right) \\ \frac{\partial \mathcal{L}}{\partial w} &= x_{err}^T \left(-\tau \begin{bmatrix}0^{3\times3}\\ I^{3\times3}\end{bmatrix} L_{RIC\_B} T_{norm}^T u_{k-1}\right) \end{aligned} \tag{18}$$

where T_{norm} is $\frac{1}{t}T$. After the calculation of the gradient the system and computing Equation 17 the matrix $\tilde{B}$ can be updated as:

$$\tilde{B} = \tau w_{k+1} \begin{bmatrix}0^{3\times3}\\ I^{3\times3}\end{bmatrix} L_{RIC\_B} T_{norm}^T u_{k-1} \tag{19}$$

$\tilde{B}$ can be updated inside the MPC formulation of Equation 10 to adjust its prediction.

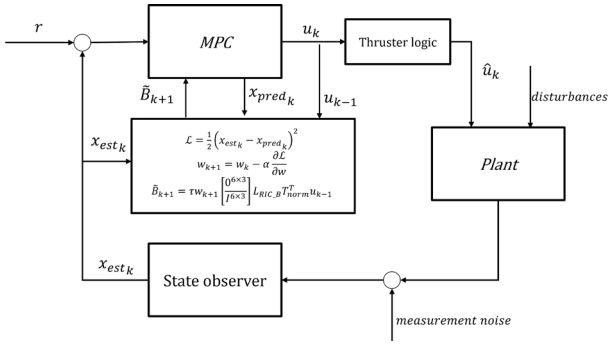

Fig. 3. Control loop architecture

3. SIMULATOR

The controller's effectiveness is evaluated using a Guidance, Navigation, and Control (GNC) simulator, which is entirely developed within an academic context by students. The simulator architecture consists of main environment were the user can setup the simulation in terms of models and simulation parameters. A graphical user interface facilitates to surface on the simulator and user can 1) setup the simulation flow and generate the code for the simulation, 2) select the models for the representation of the dynamics and kinematics, the system models (i.e. sensors and actuators, filters and control logic), the disturbance and the uncertainties due to the space environment 2) setup the outputs (in terms of number and type of graphs and table), 3) execute the simulation and follow the trends in real time, 4) manage the results and prepare the final documentation. The contribution of the students is vital and they are organised in this way:

- PhD students bear the responsibility for maintaining the entire Matlab/Simulink architecture and managing its use for didactic purposes or thesis projects, providing support to master's degree and bachelor's Degree students in the activities that involve the simulator. Their role is to ensure the robustness and accuracy of the simulation environment, thereby facilitating the integration or validation of new components into a satellite system. This encompasses hardware-level aspects, such as creating a simulation model of a real sensor, and software-level aspects, such as developing new control algorithms and filters.

- MSc students, primarily engaged in their thesis work or participating in student team projects, can create new blocks within the simulator's three macro areas. For instance, they might focus on implementing Navigation algorithms for pose estimation or design Control algorithms for different mission phases. They utilize the simulator to perform dedicated software-in-the-loop simulations, which are critical for validating their designs and ensuring they meet the required performance criteria. On the other hand.
- BSc students may utilize the developed simulator to implement new features for already existing algorithms and validate their models through simulation. For example, they might work on sensor fusion algorithms or implement a first and ideal version of an onboard sensor. Through the usage and development of the GNC simulator, students gain practical, hands-on experience in systems engineering, enhancing their proficiency in simulation modeling, control systems, and algorithm development. This project deepens students' understanding of the intricate relationship between hardware and software within satellite systems, thereby preparing them to address the complexities inherent in real-world aerospace engineering challenges.

4. RESULTS

The following details outline the initial conditions and parameter variations considered in the simulations:

- Initial Relative Position: The simulations focus on the final approach phase of the rendezvous maneuver, assuming the chaser spacecraft starts at a relative position of $[-1.46, 58.88, 1.032]_{RIC}\, m$ in the reference frame (RIC) with a relative velocity of $[0.0137, -0.004, -0.0197]_{RIC}\, m/s$.
- Randomized Mass and Thrust: To analyze the effectiveness of the proposed adaptation logic of the control system, both the chaser's mass (m_c) and its maximum thrust capability (t) are subjected to random variations at the beginning of the maneuver. These variations occur within a range of $\pm 20\%$ of their nominal values.

A total of 6 iterations were performed with different values of the thrust-to-mass ratio t/m_c as reported in the following table.

Table 2. Thrust-to-mass ratio at the beginning of simulation

t/m_c
3.2905e-04
3.3778e-04
2.9529e-04
3.4249e-04
2.6363e-04
2.6560e-04

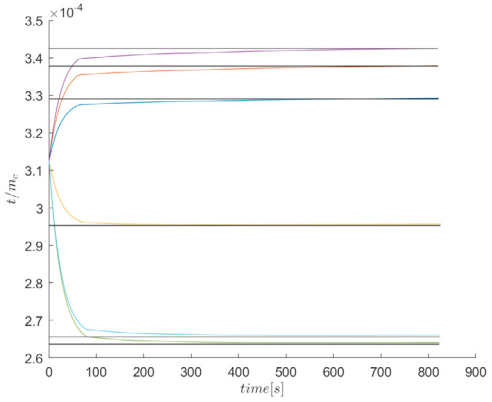

Fig. 4. Thrust-to-mass ratio over time

Fig. 3 shows that in all the cases analyzed the t/m_c converges towards the real value represented by the black horizontal line. The convergence time to achieve a 0.3% error in the force-to-mass ratio (f/m) varied between 400 and 500 seconds. Notably, a coarse 1% error level was reached much faster, after 70 seconds. The entire approach maneuver are reported in the Fig. 4.

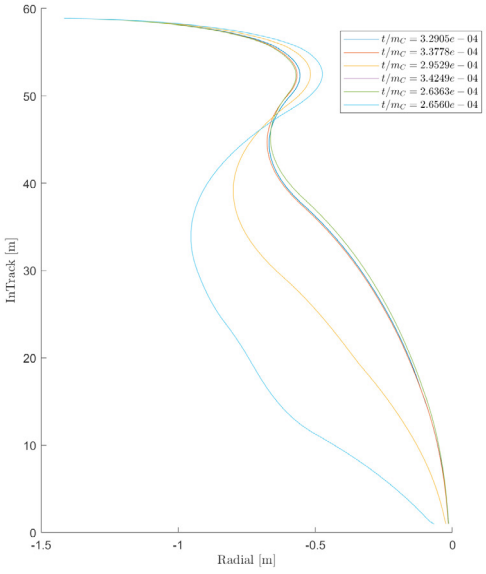

Fig. 5. Trajectory of the final approach maneuver in all cases studied

5. CONCLUSION

The proposed online learning MPC strategy has demonstrated its effectiveness in achieving the initial objective of a successful spacecraft rendezvous maneuver. This success is attributed to the controller's ability to adapt its model of the spacecraft dynamics in real-time based on sensor measurements. This adaptation capability allows the MPC to compensate for uncertainties and disturbances that are present in the actual spacecraft system, which may not be perfectly captured by the initial system model.

The online learning aspect of the MPC continuously refines the model parameters based on the real-time measured data from the plant provided by the state observer. This ongoing refinement enables the controller to maintain accurate predictions despite the presence of uncertainties of the control model. As a result, the control strategy can adjust the thrust application to ensure the spacecraft follows the desired trajectory and achieves a successful rendezvous with the target. While the current online learning approach has proven successful, One potential area for further development is the introduction of a variable learning rate. A fixed learning rate determines the step size by which the model parameters are updated based on the learning algorithm. However, a variable learning rate can potentially improve the convergence speed of the learning process. During the initial stages of the maneuver, when the model has significant deviations from the actual system, a high learning rate can be beneficial. This allows the controller to rapidly adjust its model parameters and improve its prediction accuracy. As the learning process progresses and the model becomes more accurate, a reduced learning rate can be implemented. By incorporating a variable learning rate strategy, the convergence time of the learning process could be reduced avoiding not reaching the global minimum.

From the educational point of view this work again represents a valid test bench for GNC simulator developed by the students beyond the importance of the technical results. The novel adaptive MPC is completely studied and designed by a PhD student that focuses his thesis on the GNC for RVD of smallsats. In this context, he could count on the support of MSc that develop parts of the simulator in terms of subsystems and equipment models, such as the propulsion system, the navigation and the filtering with all the uncertainties, the disturbance introduced by the space environment. Moreover, two bachelors students provide new versions of the models for sensors. All this contribution enlarges the capability of the simulator that now is enriched of new model and elements that give simulation activities easier and, even, allow to improve the confidence in the obtained results. Other important aspect is that all the new elements will result useful for the teaching activities on the courses because through the simulator was possible the validation on models, only theoretically described in the course, at the moment. The actual availability of the models simplifies and/or improves the practice exercise in course of avionics system design and space mission and system design where a reduced version of the simulator is adopted.

REFERENCES

Corpino, S., Ammirante, G., Daddi, G., Stesina, F., Corradino, F., and Basler, A. (2022). Space rider observer cube - sroc: a cubesat mission for proximity operations demonstration. *Proceedings of the International Astronautical Congress, IAC*, 2022.

Kostadinov, D. and Scaramuzza, D. (2020). Online weight-adaptive nonlinear model predictive control.

Mammarella, M., Lorenzen, M., Capello, E., Park, H., Dabbene, F., Allgower, F., Guglieri, G., and Romano, M. (2018). An offline-sampling smpc framework with application to automated space maneuvers.

Pirat, C., Ankersen, F., Walker, R., and Gass, V. (2019). H-∞ and *IEEE Transactions on Control Systems Technology*, PP, 1–8. doi:10.1109/TCST.2019.2892923.

Pirat, C., Ribes Pleguezuelo, P., Keller, F., Zuccaro Marchi, A., and Walker, R. (2021). Toward the autonomous assembly of large telescopes using cubesat rendezvous and docking. *Journal of Spacecraft and Rockets*, 59. doi:10.2514/1.A34945.

Roscoe, C.W., Westphal, J.J., and Mosleh, E. (2018). Overview and gnc design of the cubesat proximity operations demonstration (cpod) mission. *Acta Astronautica*, 153, 410–421. doi: https://doi.org/10.1016/j.actaastro.2018.03.033.

Stesina, F., Corpino, S., Novara, C., and Russo, S. (2022). Docking manoeuvre control for cubesats. *The Journal of the Astronautical Sciences*, 69. doi:10.1007/s40295-022-00307-1.

Stesina, F. (2021). Tracking model predictive control for docking maneuvers of a cubesat with a big spacecraft. *Aerospace*, 8(8). URL `https://www.mdpi.com/2226-4310/8/8/197`.

Ventura, J., Ciarcià, M., Romano, M., and Walter, U. (2017). Fast and near-optimal guidance for docking to uncontrolled spacecraft. *Journal of Guidance, Control, and Dynamics*, 40(12), 3138–3154. doi:10.2514/1.G001843. URL `https://doi.org/10.2514/1.G001843`.

Votta, R., Gardi, R., Zaccagnino, E., Cardi, M., Basler, A., Corradino, F., Taiano, G., Carrai, F., Carubia, F., Di Clemente, M., Albano, M., and Fedele, A. (2022). David the first 6u cubesat mission of the italian space agency programme iperdrone as demonstration of new on orbit services performed by space drones. *36th Annual Small Satellite Conference*.