73rd International Astronautical Congress (IAC), Paris, France, 18-22 September 2022.

IAC-22,D2,IPB,1,x69634

A Shrinking Horizon Model Predictive Control for Landing of Reusable Launch Vehicles

Guillermo Zaragoza Prous[a]*, Leonard Felicetti[b]

[a] Cranfield University, United Kingdom, guillermozaragozaprous@gmail.com
[b] Cranfield University, United Kingdom, leonard.felicetti@cranfield.ac.uk
* Corresponding author

Abstract

The recent advancements of the onboard computational capabilities enable the deployment of GNC algorithms to perform autonomous decisions and complex operations in the final stages of a landing manoeuvre of Reusable Launch Vehicles (RLV). In most cases, such algorithms embed model predictive schemes to optimize during flight by offering a wider versatility compared to classical schemes and the capabilities of identifying potential future behaviours and risks for the mission. Applications such as the vertical landing of reusable launchers (i.e. SpaceX Falcon 9, and Blue Origin New Shepard) and planetary landers (NASA Perseverance) are currently operating or under development by considering the potential benefits of such a technology. A shrinking horizon Model Predictive Control (MPC) is proposed for the guidance and control of RLV during powered descent phases. Standard MPC schemes use receding horizons where the optimal controls are calculated during constant time-length intervals. The direct implementation of such schemes leads to a hovering-like behaviour of the vehicle, which will never reach the landing platform as the final time of landing is always postponed at each iteration step. The solution proposed in this paper consists on implementing an MPC algorithm that calculates and updates the optimal thrust profile along time-dependent decreasing horizons. The algorithm updates and adapts the time-length of the receding horizon as a function of a time-scaling factor and the time that has passed since the last MPC iteration. It introduces a new concept called terminal horizon, which determines the maximum time in which the RLV must be landed. The optimal solutions are found through convex optimisation algorithms. Numerical simulations and results show an enhanced performance of the guidance scheme and validate the idea that a decreasing horizon is more suitable than a receding one in a powered descent scenario. Monte Carlo and Parametric analyses are performed to assess the performance of the proposed algorithm in a landing test case scenario. This simulated case considers the disturbances caused by Earth's atmosphere drag force in interaction with a descending first stage of SpaceX Falcon 9 rocket. This study demonstrates the applicability of the proposed MPC technique identifying feasibility boundaries for tuning the MPC parameters and determining the range of initial conditions that allow for a successful landing.

Keywords: Model Predictive Control, Powered Descent Landing, Convex Optimization

1. Introduction

Vertical descent and landing of rockets after launch has been proven to have a significant impact on the overall mission cost and has enabled several companies, e.g. SpaceX, to provide favourable launch-services to new-space actors and enterprises [1].

Controlling in an accurate and robust way the powered descent of such systems is one of the critical tasks from a technical and theoretical point of view. The unstable dynamics of the fast free-falling launcher shall be controlled by modulating and re-orienting the thrust in such way that a soft-landing manoeuvre can be accomplished [2].

The study of powered descent guidance initiated with the Apollo program in the early 1960s using optimal control theory [3–5]. The interest of exploration missions to Mars in the late 1990s and 2000s lead to the utilization of constrained optimisation methods to solve the powered descent guidance problem and the pinpoint landing of the different missions. During the late 2000s, a series of papers were published by Açikmese where the focus was given to the application of convex programming for solving in a lightweight way the planetary descent problem [6] [7]. Such papers demonstrated that a second-order cone programming (SOCP) set of equations can be used to solve the powered descent problem, making use of state of the art convex optimisation solvers, such as interior point methods.

In the last decade, the interest for the guidance and control of launchers using this methodology has raised exponentially. Convex models have been developed not only for the translational dynamics, but also including the rotational dynamics for 6 degrees-of-freedom models. In [8], for example, the powered descent guidance problem is tackled with the aim of successive convexification, which is a sequential convex programming (SCP) method that uses virtual control and trust region modifications to aid the convergence of the problem. In [9], a similar approach is followed adding aerodynamic controls in the system for

a 2-dimensional problem over a flat Earth.

However, for the Earth descent and landing case, perturbing actions, such as the atmospheric drag, winds and flexibility effects along the body of the launcher might strongly affect the performance of the manoeuvres. It appears then evident that open-loop schemes, based only on the optimisation of trajectories, might not be able to overcome these disturbances and closed-loop algorithms need to be developed. Model Predictive Control (MPC) has been studied and proposed as a form to close the guidance and control loop for the powered descent problem of reusable launch vehicles (RLV) [10]. This methodology had being widely used in a variety of industrial applications characterised by extremely low dynamics, such as for chemical and oil processing [11] [12], but the application to the fast dynamics of RLV powered descent still presents challenges. Typically, MPC uses a receding horizon strategy where the optimal control is always calculated for a fixed period of time shifting at each iteration step [11]. Nonetheless, the direct implementation of such schemes in a powered descent scenario is not trivial and it can lead to unexpected behaviours such as hovering-like trajectories with the vehicle never reaching the landing target. In [13] for example, a MPC scheme is presented by finding an optimal time of the horizon at each iteration, because a fixed receding horizon is foreseen as not suitable for the guidance and control of a RLV.

This paper proposes a novel algorithm which aims for a simplification of the time-length update phase of the horizon within a Model Predictive Control algorithm. Specifically, an iterative method which modifies the length of the horizon according to a predefined update factor is presented. The so obtained solution is sub-optimal, but less demanding in terms of computational effort compared to previous optimised algorithms, such as in [13]. These optimised schemes tried to obtain an optimum horizon at the initial step of each iteration of the MPC controller. Nonetheless, this calculation implied that the problem must be solved several times per iteration, which was computationally inefficient. The approach proposed in this paper sensibly reduces the number of iterations required to obtain a feasible time horizon with the consequent decrease of computational time. Such a characteristic might have potential benefits when implemented in real-time embedded systems.

The outline of this paper is as follows: Section 2 presents a simplified set of equations and constraints that defines the guidance problem, a convexification of such system of equations is proposed in Section 3 so that and MPC algorithm can be developed. The latter is presented in Section 3.2 along with the proof of why the fixed-time receding horizon strategy is not suitable for the powered descent guidance problem. Lately, the simulation results and a Monte Carlo analysis, in Section 4, prove the robustness of the algorithm against different initial and environmental conditions. Finally, Section 5 provides the concluding remarks and future works.

2. Problem Statement

In this section, the rocket plant used for the simulation is presented, along with the mathematical definition of the landing problem.

2.1. Model

A point mass 2-dimensional model is used for the simulations, where the thrust T is decomposed in components T_x and T_z, as in [6] and the RLV is subjected to atmospheric drag D:

$$\begin{cases} \dot{x} = V_x \\ \dot{z} = V_z \\ \dot{V}_x = \frac{T_x - D\cos(\gamma)}{m} \\ \dot{V}_z = \frac{T_z - D\sin(\gamma)}{m} - g \\ \dot{m} = -\frac{|T|}{I_{sp} g_0} \end{cases} \tag{1}$$

where $\gamma = tan(\frac{V_z}{V_x})$ is the ascent/descend angle of the RLV. The orientation of the RLV is not explicitly represented in the set of Equations 1, although one could consider the tangent of the thrust vector as that orientation of the RLV, $\theta = tan(\frac{T_z}{T_x})$, with the thrusters fixed to the body. The atmospheric drag D is defined as follows:

$$D = \frac{1}{2}\rho(z) \cdot V^2 \cdot S(\alpha) \cdot C_D(\alpha) \tag{2}$$

where $\rho(z)$ is the atmospheric density encountered by the RLV, V is the magnitude of the velocity, $S(\alpha)$ is the exposed area of the launcher to the atmospheric flow and $C_D(\alpha)$ is the drag coefficient of the RLV. The angle of attack α is defined as:

$$\alpha = |\theta - \gamma| \tag{3}$$

Under the assumption of isothermal atmosphere, an exponential model is considered [14]:

$$\rho(z) = \rho_0 \cdot e^{-z/H} \tag{4}$$

For the atmospheric drag computation it is assumed that the RLV has a cylindrical shape, thus the exposed area reads as:

$$S(\alpha) = \pi {R_L}^2 \cdot cos(\alpha) + 2R_L L_L \cdot sin(\alpha) \tag{5}$$

where R_L and L_L are the radius and the length of the cylinder, respectively. The dependency of the $C_D(\alpha)$ with the angle of attack (α) is given by [15].

2.2. Definition of Landing Problem

The Landing Problem can be defined as a Two Boundary Value Problem (TBVP) with constraints on the initial and final values of the state vector as follow:

$$\begin{cases} x(t_0) = x_0 \\ z(t_0) = z_0 \end{cases} \quad \begin{cases} V_x(t_0) = V_{x_0} \\ V_z(t_0) = V_{z_0} \end{cases} \tag{6}$$

$$\begin{cases} x(t_f) = 0 \\ z(t_f) = 0 \end{cases} \quad \begin{cases} V_x(t_f) = 0 \\ V_z(t_f) = 0 \end{cases} \tag{7}$$

where t_0 and t_f are the initial and final time. The landing pad is assumed to be fixed in the origin of the reference frame (Constraint 7) and needs to be reached by the RLV with zero velocity, in order to have a soft landing.

The TBVP assumes a cost function that minimises the required thrust:

$$J = \int_0^{t_f} |T| \, dt \tag{8}$$

The thrust magnitude must be maintained within the limits of operation of the rocket engine [6]. Thus, the following range of thrust values is assumed as applicable to the model:

$$0 < T_{min} \leq T(t) \leq T_{max} \tag{9}$$

The position of the RLV necessarily cannot be below ground surface: this is taken into account by imposing the Constraint 10. Additionally, the main engines should point always downwards (i.e. $0 \leq \theta \leq \pi$), that leads to set the Constraint 11, as the thrust should push anyway upwards the RLV in any time of the manoeuvre.

$$0 \leq z(t) \tag{10}$$

$$0 \leq T_z(t) \tag{11}$$

Constraint 12 is defined on the final mass of the RLV, as it has to be larger than the dry mass of the RLV to be considered a feasible solution.

$$m_{dry} \leq m(t_f) \tag{12}$$

The orientation θ of the RLV is constrained by 13, where θ_B is a predefined boundary parameter that can change over time. θ_B restricts the range of the possible orientations of the RLV at that specific time.

$$|\frac{\pi}{2} - \theta_B(t)| \geq |\frac{\pi}{2} - \theta(t)| \tag{13}$$

3. Model Predictive Control

The adopted Model Predictive Control strategy is shown in Figure 1, where the physical model of launcher (Equation 1) is controlled by an optimal thrust profile obtained by a convex optimisation solver (SeDuMi [16]). The solver utilises the convexified set of constraints and equations of motion presented in Section 3.1. The implementation of the shrinking, also called decreasing, horizon model predictive algorithm is then described in Section 3.2.

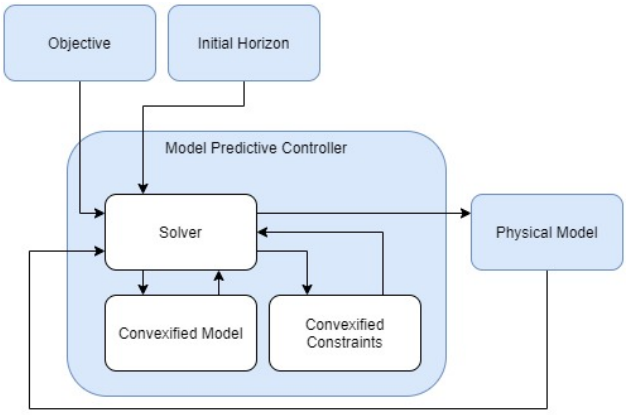


Figure 1: Flowchart of MPC Strategy

3.1. Convexification of the Problem

A convexification strategy is needed to turn the landing problem into a Semidefinite Programming (SDP) problem [6]. The sources of non-convexity of the model are: (1) the change of mass in the last of Equation 1 and (2) the thrust limit in Constraint 9.

In order to revert the non-convexity of those equations, the procedure described in [6] is followed. For solving the non-convexity (1), a change of variable, as in Equation 14, is introduced to take into account the non-linear decrease of the RLV mass.

$$\eta = ln(m) \tag{14}$$

By defining the fuel consumption rate as in Equation 15, the last of Equation 1 can be rewritten as in Equation 16. σ is defined later on in Equation 18.

$$\beta = -\frac{1}{I_{sp}\, g_0} \tag{15}$$

$$\dot{\eta} = \frac{\dot{m}}{m} = -\beta \cdot \sigma \tag{16}$$

The source of non-convexity (2) is tackled by introducing a slack variable Γ as in Equation 17. According to Lemma 1 in [6], this equation converts to $\|\vec{T}\| = \Gamma$ for the optimal solution.

$$\|\vec{T}\| \leq \Gamma \tag{17}$$

Two new variables (σ and $\vec{u}$) can be defined, in Equations 18 and 19, as the magnitude and the vector of the specific thrust, respectively. These are used to redefine Equation 17 as in Equation 20.

$$\sigma \triangleq \frac{\Gamma}{m} \tag{18}$$

$$\vec{u} \triangleq \frac{\vec{T}}{m} \tag{19}$$

$$\|\vec{u}\| \leq \sigma \tag{20}$$

The objective index defined in Equation 8 can now be reformulated with the new σ variable as Equation 21.

$$J = \int_0^{t_f} \sigma \, dt \tag{21}$$

Thus, the state variables can be updated at each iteration step of the prediction phase in the MPC by using the linearised set of equations of motion in Equation 22, where the atmospheric drag is considered as a external disturbing factor for the controller.

$$\begin{cases} x(t+\Delta t) = \frac{1}{2}u_x(t)\,\Delta t^2 + V_x(t)\,\Delta t + x(t) \\ z(t+\Delta t) = \frac{1}{2}(u_z(t) - g)\,\Delta t^2 + V_z(t)\,\Delta t + z(t) \\ V_x(t+\Delta t) = u_x(t)\,\Delta t + V_x(t) \\ V_z(t+\Delta t) = (u_z(t) - g)\,\Delta t + V_z(t) \\ \eta(t+\Delta t) = -\beta\sigma(t)\,\Delta t + \eta(t) \end{cases} \tag{22}$$

Such a system can be rewritten as in Equation 23, where $X = [x, z, V_x, V_z, \eta]^T$ is the state vector, $U = [u_x, u_z, \sigma]^T$ is the control action vector, and Φ, Ξ and G are the transition, control distribution and gravity matrices defined in Equations 24, 25 and 26, respectively.

$$X(t+\Delta t) = \Phi(\Delta t)X(t) + \Xi(\Delta t)U(t) + G(\Delta t) \tag{23}$$

$$\Phi(\Delta t) = \begin{bmatrix} 1 & 0 & \Delta t & 0 & 0 \\ 0 & 1 & 0 & \Delta t & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{24}$$

$$\Xi(\Delta t) = \begin{bmatrix} \frac{1}{2}\Delta t^2 & 0 & 0 \\ 0 & \frac{1}{2}\Delta t^2 & 0 \\ \Delta t & 0 & 0 \\ 0 & \Delta t & 0 \\ 0 & 0 & -\beta\Delta t \end{bmatrix} \tag{25}$$

$$G(\Delta t) = \begin{bmatrix} 0 & -\frac{1}{2}g\Delta t^2 & 0 & -g\Delta t & 0 \end{bmatrix} \tag{26}$$

3.2. MPC Algorithm

Two different horizon management strategies are analysed in the following subsections. First, an example of a simpler fixed length *Receding Horizon* strategy is described in Section 3.2.1 showing, with a dedicated simulation of a test case, the intrinsic limitations when applied to a landing scenario. Then, the new *Shrinking Horizon* strategy is proposed and described in Section 3.2.2.

3.2.1. Receding Horizon MPC

A fixed prediction horizon time t_h can be used to predict the behaviour of the RLV ahead on time at each iteration step t_I. In Figure 2, a test case simulation of the landing of a Falcon 9 rocket is presented. The initial conditions and mass of the RLV are listed in Table 1 (for the physical properties of this rocket see Table 2 later in the paper). Please note that in Table 1 the mass of the first stage of a Falcon 9 rocket is assumed still having 2% of propellant left.

Variable	Initial Value
Position $X[m]$	2800
Position $Z[m]$	5000
Velocity $V_x[m/s]$	-150
Velocity $V_z[m/s]$	-280
Fuel percentage [%]	2
Mass m $[t]$	33.5
Orientation θ_B $[deg]$	45

Table 1: Initial conditions for the Nominal Scenario.

The behaviour of the components of the state vector over the time are represented in Figure 2. The prediction horizon time t_h is set to 35 s, the maximum simulation time is 200 s and the time between each iteration of the MPC is $\Delta t_I = 1$ s.

A hovering behaviour can be clearly seen in Subfigure 2a): after an initial transient, the RLV maintains a stationary altitude of about 75 m, even though it reaches the desired final position in the x-axis, as shown in Subfigure 2c). This can be corroborated in the two velocity subplots, Subfigures 2b) and 2d), showing both V_x and $V_z \approx 0$ m/s during the hovering period. To highlight the hovering phase of the RLV, the consumption of fuel defined by the last equation in Equation 1 has been neglected, in order to prevent the RLV to fall due to the lack of fuel.

In Figure 2e) the engine is at maximum thrust during the initial and middle stage of the simulation, to compensate the fast initial free falling descent but, once the vertical velocity is cancelled out, it smoothly decreases to maintain the altitude. The orientation of the thrust, also considered the orientation of the RLV, is nearly 90 degrees during the hovering phase, as see in 2f). It is worth to note that the final landing condition is reached at the end of the

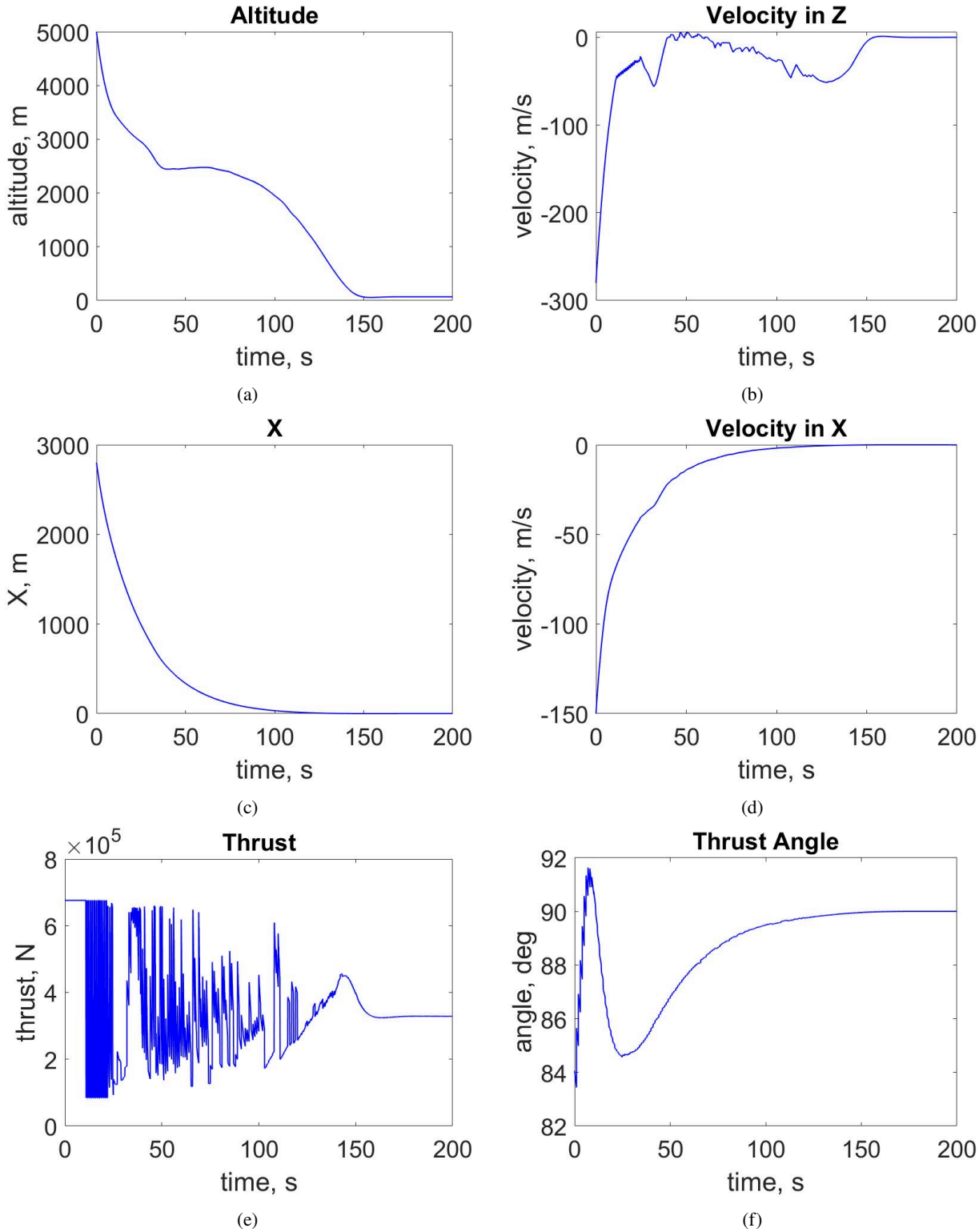
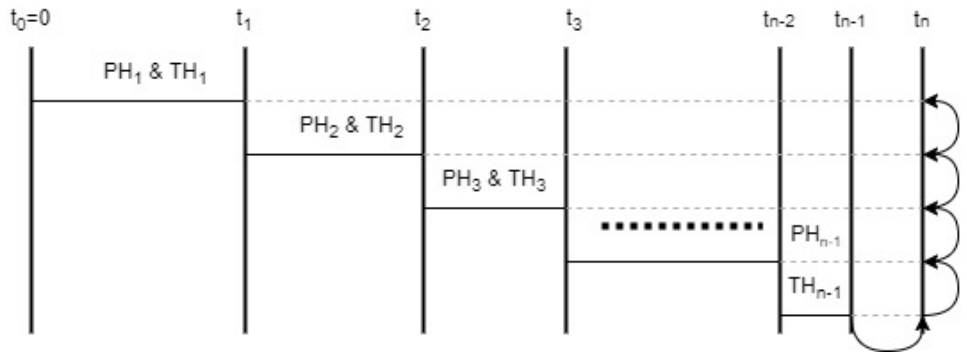


Figure 2: Receding horizon MPC simulation with hovering behaviour.



(a)

Figure 3: Generalisation of the shrinking MPC algorithm

prediction horizon but this is continuously postponed at each iteration step, leading to the hovering behaviour of the RLV, as also noted by [13]. This justifies the effort in this paper to find an alternative MPC implementation that avoids the hovering condition, that has led us to formulate the decreasing horizon MPC.

3.2.2. Shrinking Horizon MPC

The proposed modification to the MPC method is based on the adjustment of the horizon length to solve the aforementioned issues and to allow the RLV to land. For this purpose, it is necessary to define several time-related parameters:

a) *Terminal Horizon* time (TH): which serves for setting the global desired time to be landed. This represent the upper-bound limit for the touch down, that in any case can happen anytime before.

b) *Prediction Horizon* time (PH): time used in the prediction step by the optimisation solver.

c) Δt_I: time between each MPC iteration, when the optimiser recalculates the optimal solution by getting the updated state of the RLV.

d) Δt_P: time between each of the prediction states computed within the same prediction horizon.

e) *Update Factor* (UF): determines when and by which rate TH and PH are decreased. Its value must be set between 0 and 1.

Algorithm 1 and Figure 3 describe how this new strategy works. In the first step of Algorithm 1, the initial values of the new parameters are defined. Two nested loops are then executed. The outer one updates TH and PH in each iteration of this method and it runs until the prediction time PH is smaller than Δt_I. The inner loop executes the prediction and calculates the optimised control for the RLV. The prediction step is performed by the function *convexOptimiser(X, PH,* Δt_P*)* with the given PH and Δt_P and it solves the problem in Equation 21, implementing the equations and constraints described in Section 3.1. When the outer loop is finished, the algorithm still applies the last part of the generated control signal U to the RLV to finish the landing manoeuvre.

Algorithm 1 Shrinking Horizon MPC Algorithm

1: Define TH, PH, Δt_I, Δt_P, UF and X_0
2: **while** $PH > \Delta t_I$ **do**
3: **for** $iteration = 0, 1, 2, \ldots, TH \cdot \text{UF}$ **do**
4: U = convexOptimiser(X, PH, Δt_P)
5: Apply U for Δt_I
6: **end for**
7: Update $TH = TH \cdot (1 - \text{UF})$
8: Update $PH = PH \cdot (1 - \text{UF})$
9: **end while**
10: Apply the rest of U

Figure 3 shows how the update of TH and PH works within the algorithm. Assuming initial values TH_1, PH_1, Δt_I, Δt_P and UF, the horizons TH_1 and PH_1 are applied during a time t_1 equal to $TH \cdot \text{UF}$. At that moment, TH_2 and PH_2 are obtained multiplying TH_1 and PH_1 by $(1-UF)$, which corresponds to lines 7 and 8 in Algorithm 1. Then, the algorithm uses these new horizons into the new iteration. It is important to remark that only the terminal and prediction horizon values are modified during the execution of the MPC algorithm. The algorithm runs until when the prediction horizon PH is shorter in time than the Δt_I, the PH_{n-1} in Figure 3. In this case, the RLV uses the remaining part of the predicted control generated by the MPC. As it is stated in Section 4, the selection of the horizon times and initial parameters is key to the performance and success of the algorithm, and their values change depending on the initial values of the problem.

4. Numerical Results

The applicability of the proposed MPC algorithm to RLV scenarios is assessed via numerical simulations. Multiple tests are shown in this section, simulating landings on Earth with the first stage of SpaceX's Falcon 9 rocket [17]. The parameters of the launcher and the planet are reported in Tables 2 and 3, respectively. For the numerical simulations only 1 engine of the rocket is used. This is compliant with what can be seen in the landings of SpaceX, as in Figure 4. The initial mass of the RLV in each of the several cases covered in this section will be determined by the percentage of fuel mass left at the beginning of the landing scenario. Likewise, the environment used for the simulations is the lower section of the atmosphere of the Earth.

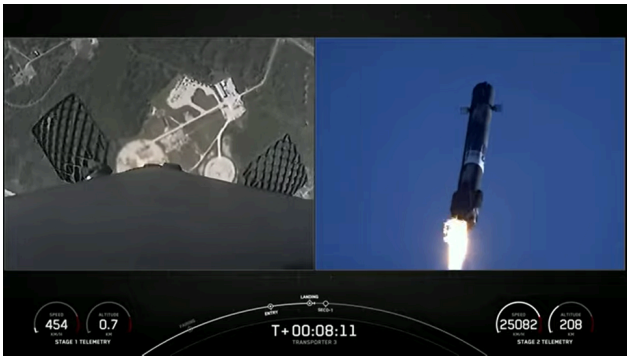


Figure 4: Capture of the Landing of the First Stage of a Falcon 9. Transporter-3 mission.[18]

Rocket	Falcon 9 1st Stage
N. Engines	9
Engine Thrust $[kN]$	845
Specific Impulse $[s]$	311
Propellant Mass[1] $[t]$	395.7
Dry Mass $[t]$	25.6
Length $[m]$	41.2
Diameter $[m]$	3.7
Maximum Thrust [%]	0.8
Minimum Thrust [%]	0.1

Table 2: Falcon 9 first stage parameters used in the simulations.

The motivation of this study is to demonstrate that the presented algorithm is capable of landing a RLV in a real scenario. The selected scenario follows the particular case of the landing of the 1st stage of the Falcon 9 that launched the Iridium NEXT 3 satellite the 9th of October 2017. In the telemetry data collected in [19], it can be seen that when the Landing Burn Ignition starts, the RLV is in a position around $x_0 = 2800$ m, $z_0 = 5000$ m and velocity of $V_{x_0} = -150$ m/s, $V_{z_0} = -280$ m/s. This data will be considered for the initial state vector of the problem to solve.

Environment	Earth
Air density at sea level $[kg/m^3]$	1.217
Scale Height $[m]$	85000
Gravity acceleration at sea level $[m/s^2]$	9.81
Speed of Sound $[m/s]$	340

Table 3: Earth parameters used in the simulations. Data from [20].

4.1. Understanding the feasibility region of the optimisation problem

4.1.1. Sensitivity analysis with respect to initial conditions and prediction horizon duration

The first step into understanding where is the feasibility region of the problem, is to perform an analysis of the initial conditions for a single run of the convexified problem without running the MPC algorithm yet. Table 4 shows the boundaries set for a Monte Carlo analysis in which the test iterates between 20 and 50 s of prediction horizon, with steps of 2 s, and saving the most suitable of them. The most suitable PH is understood as the one that uses the less quantity of fuel. The value of θ_B starts at 45 degrees and increases linearly until 85 degrees at the end of the TH for each simulation.

Variable	Min. Value	Max. Value
Position $X[m]$	2000	3500
Position $Z[m]$	4000	6000
Velocity $V_x[m/s]$	-250	-50
Velocity $V_z[m/s]$	-350	-200
Fuel percentage [%]	1.5	3.5
Mass m $[t]$	31.5	39.5
Orientation θ_B $[deg]$	45	45

Table 4: Monte Carlo minimum and maximum boundaries for the initial conditions of the problem.

The Monte Carlo analysis ran 1000 simulations with different initial conditions: 680 of them allowed for a successful landing. Having a look to Subfigure 7a), the variability of the initial position does not directly affect the solution of the problem. The feasible and unfeasible cases are spread all over the figure. On the other hand, Subfigure 7b) clearly demonstrates a relation between the successful landing and the components of the velocity. Nevertheless, the impact of each component of V is not the same. While a $|V_z| > 300$ m/s makes the problem practically unfeasible, the $|V_x|$ still allows for a larger margin in its value, with a slightly greater number of unfeasible problems when it

[1] Initial propellant mass at launch.

reaches 250 m/s in the negative direction. Considering that the X position of the RLV is positive and the V_x is negative, a large magnitude in the latter can cause an overshoot of the RLV that then the optimiser has to overcome with the help of the X component of the thrust T_x. Furthermore, this behaviour also applies to the relation between Z and V_z where, as imposed by the Constraint 10, the system cannot overshoot as it would mean crashing to the ground. This behaviour is represented in Figure 5, with the first case in green and the second in red.

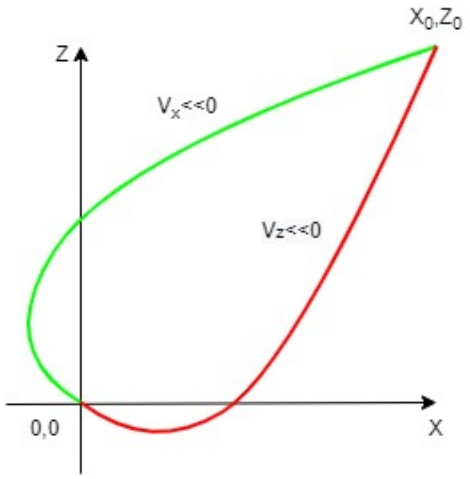


Figure 5: Overshoot of the position of the RLV when $V_x << 0$ in green and when $V_z << 0$ in red.

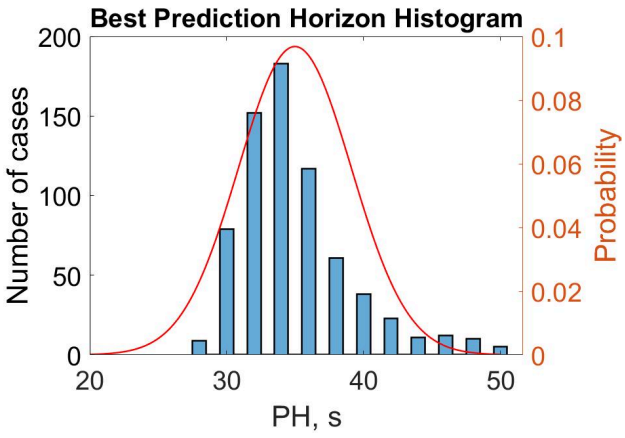


Figure 6: Histogram of the best PH for the different scenarios in the Monte Carlo simulation compared to the Gaussian distribution with the mean and variance of the data set

The sensitivity of the problem is not only affected by the initial dynamic conditions, but also by the initial mass of the RLV. Subfigure 7c) demonstrates that at high velocities, $|V| > 350$ m/s, there is a higher chance of success if the fuel mass left in the RLV is below 0.025%. This can be explained by the fact that with a higher velocity, the decrease of mass in the RLV can compensate the thrust needed to overcome that velocity. Nevertheless, the amount of fuel mass must always be sufficient to land. Likewise, it is appealing to look at the best prediction horizon, i.e. the PH with the less amount of fuel consumed, for each case in the Monte Carlo study. In Subfigure 7d), the data shows that for this specific case, a good PH is between 32 and 36 s, as there are the highest number of feasible results in this range of values of the prediction horizon. The red dots in $PH = 20$ s represent the cases in which the problem was unfeasible and do not mean that 20 s is a suitable value for those cases. In accordance to Subfigures 7b) and 7c), it can be seen how mostly all the unfeasible cases are placed when $|V| > 350$ m/s. In a further exercise to understand the feasibility region of this problem, the same results have been plotted as the histogram shown in Figure 6. This graph demonstrates that the distribution of the PH for the most optimised solution follows a Gaussian distribution with a mean value of 34.98 s and a variance of 4.12. This data will help to select the proper PH for Section 4.2, as it must be close to the mean value to ensure the best possible performance.

4.1.2. Finding the applicable ranges of the prediction horizon for different initial velocities

The magnitude of the initial velocity V and its components V_x and V_z are the main factors that affect the feasibility of the problem as seen in section 4.1.1. To understand better the effect of these variables, a parametric analysis has been performed for the initial conditions of the nominal scenario presented in section 4.2. It allows to understand the range of feasible PH values for each velocity V considered. Table 5 describes the initial conditions of this scenario, while the range of velocities is specified in Table 4. The parametric analysis' results can be seen in Figure 8. In Subfigure 8a) the values for the problem show a global minimum of 30 s around $V = (-180, -280)$ m/s. If these values are propagated during time from the initial position $(2800, 5000)$ m, it can be understood why it happens. These velocity values bring the RLV close to the landing site $(0, 0)$ m without the need of thrust, and therefore minimising the error correction to overcome in the optimised trajectory.

Variable	**Initial Value**
Position $X[m]$	2800
Position $Z[m]$	5000
Fuel percentage [%]	2
Mass m $[t]$	33.5
Orientation θ_B $[deg]$	45

Table 5: Initial conditions for the Parametric Analysis of the Prediction Horizon of the problem.

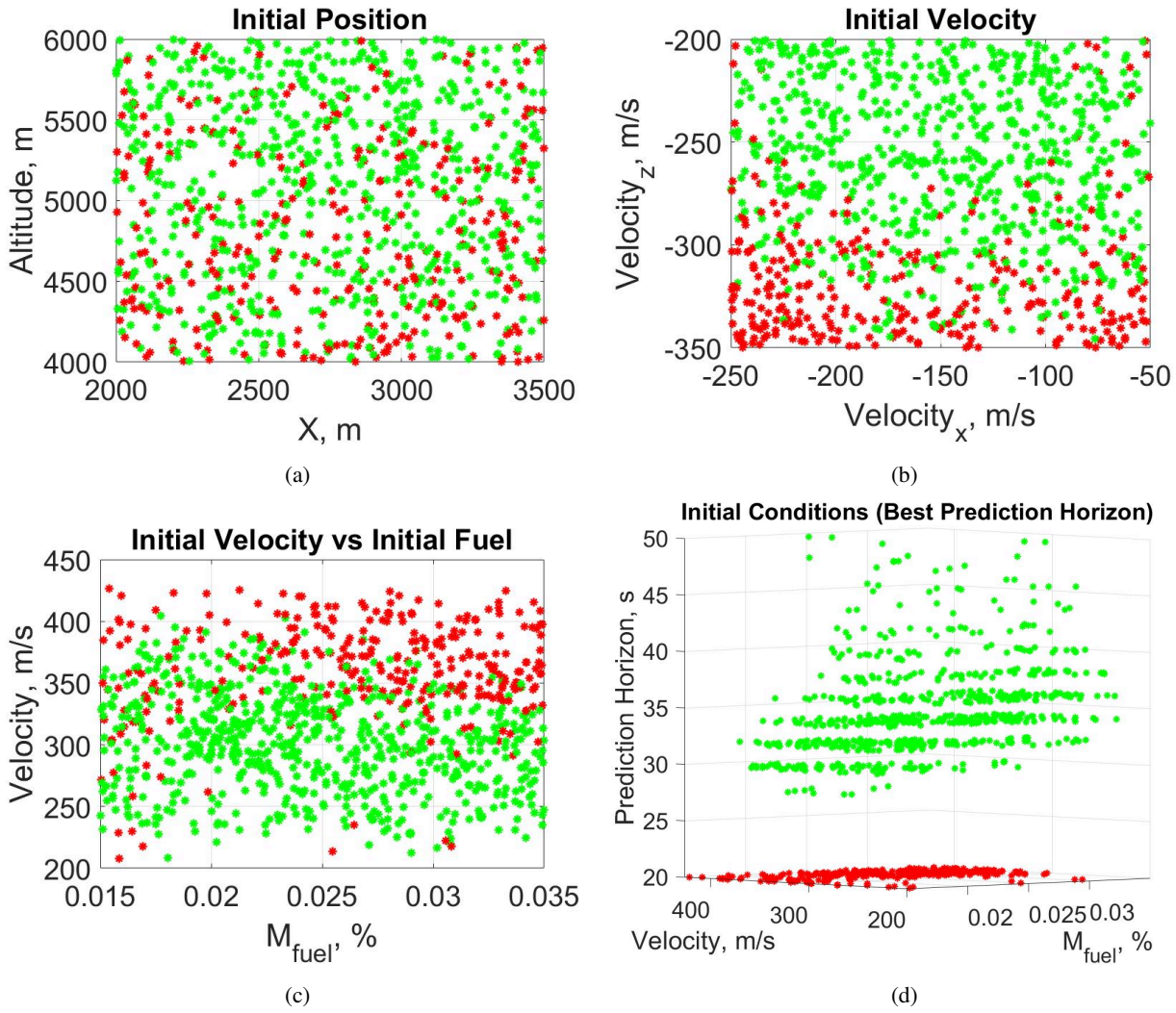
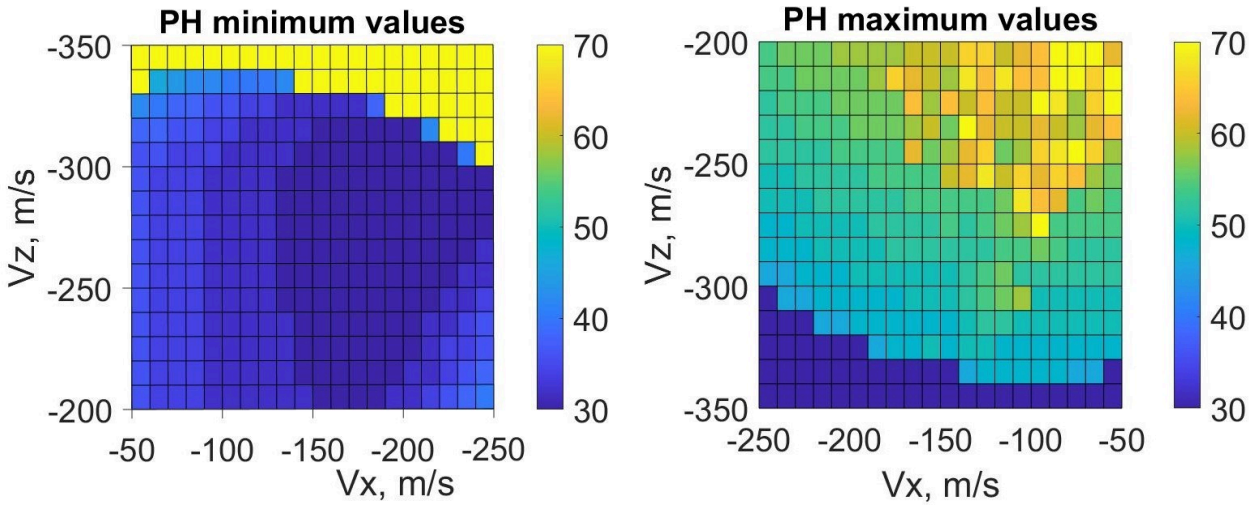


Figure 7: Sensitivity Analysis: Feasibility of the problem regarding the initial position, velocity, fuel mass and prediction horizon. Green means feasible while red means unfeasible.



(a) (b)

Figure 8: Parametric Analysis of the initial velocity for the optimisation problem: a) Minimum PH and b) Maximum PH in seconds. The values in yellow in a) and purple in b) define the infeasible region of the problem.

Depending on the combination of values of V_x and V_z, the minimum feasible value of PH increases in a different manner. For example, combining low values of V_x with high values of V_z increases rapidly the minimum feasible value of PH as the RLV has to push itself towards the landing pad ($\sim$ 44 s), upper-left corner of Subfigure 8a). In the same way, high values of V_x and low values of V_z increase the minimum feasible PH, although slightly less ($\sim$ 40 s). In this scenario the acceleration due to the force of gravity is not adding the component with a higher magnitude and therefore the overshoot in that direction is smaller. Likewise, for a $V_z > -300$ m/s, the minimum PH is indirectly proportional to the absolute magnitude of V_x, reaching a top value of 36 s when $V_x = -50$ m/s. Moreover, the values in yellow in Subfigure 8a) and in purple in Subfigure 8b) when V_z is close to -350 m/s represent the cases in which the problem is unfeasible. In Subfigure 8b) the region with the maximum feasible PH values is the upper-right corner as the values of V_x and V_z are smaller. It is due to a lower thrust needed and therefore a higher quantity of fuel available to descend slowly or even hover before touching ground.

4.2. Nominal landing scenario

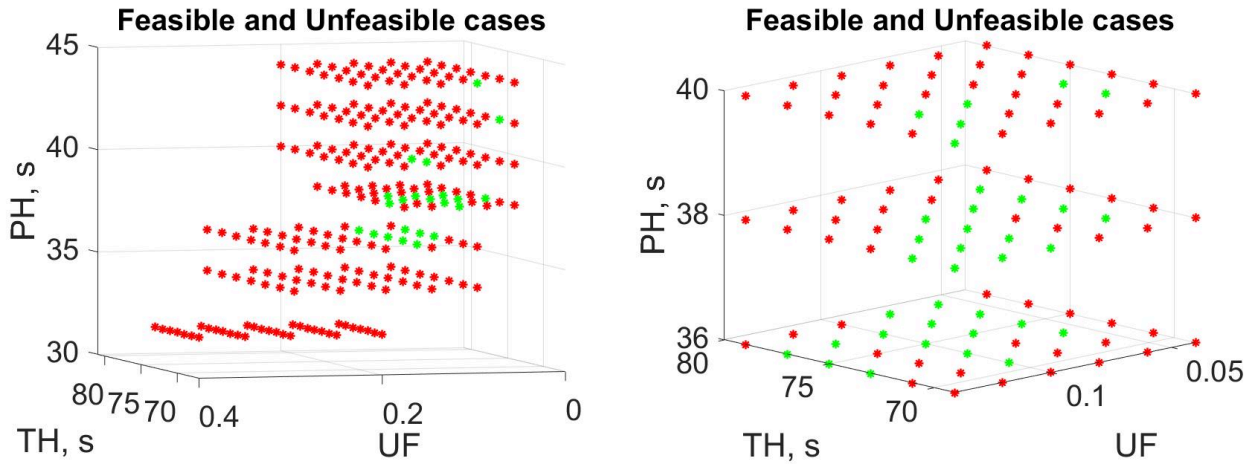
4.2.1. Finding the applicable ranges of terminal horizon and update factor

After understanding the applicability of the presented optimiser for the landing scenario, the performance of the proposed MPC algorithm is analysed. The first step is to understand the ranges in which the problem is feasible for the TH and UF, based on the results of the Subsection 4.1.2. In this aspect, a set of parametric tests has been performed. Multiple cases of the proposed MPC have been run. First through a wider scan to obtain a larger picture of where the feasible zone is, using the values reflected in Table 6, and then through a more focused scan, with values reflected in Table 7. A $TH \approx 2 \cdot PH$ and a UF contained in $[0.01, 0.16]$ have been found to be suitable for the solution of the problem. This logic has been followed to select the values of Table 6.

TH			
PH [s]	**Min. Value [s]**	**Max. Value [s]**	**Step Size [s]**
32	67	73	1
[34, 44]	69	81	2
UF			
PH [s]	**Min. Value [-]**	**Max. Value [-]**	**Step Size [-]**
32	0.20	0.40	0.05
[34, 36]	0.08	0.28	0.05
38	0.04	0.16	0.03
[40,44]	0.04	0.20	0.04

Table 6: Range of values of the TH, the UF and the PH for the Parametric Analysis of the TH and UF based on the result obtained in subsection 4.1.2.

The feasibility of the problem is determined by ensuring that the constraints in Section 2.2 are fulfilled and by comparing the state of the RLV at the end of an MPC iteration with the final objective, which is [0 m, 0 m, 0 m/s, 0 m/s]. If the position and velocity er-



(a) (b)

Figure 9: Feasible and unfeasible cases of the Parametric Analysis of PH, TH and UF.

rors are less than a certain margin, the problem is considered solved. In the numerical simulations presented in this paper, and in this section in particular, these margins are 2 m and 2 m/s.

Variable	Min. Value	Max. Value	Step Size
PH[s]	36	40	2
TH[s]	69	79	2
UF[-]	0.04	0.14	0.02

Table 7: Range of values of the TH, the UF and the PH for the Parametric Analysis of the TH and UF based on the result obtained using the values in Table 6.

The execution of the first set of scenarios revealed a zone in which the problem is feasible, as seen in Subfigure 9a). This set contained 294 cases of which 22 were feasible. The majority of them were within a $PH = [36, 40]$ s, a $TH = [69, 79]$ s and a $UF = [0.04, 0.14]$. There were other feasible cases seen in Subfigure 9a), but as they were isolated, they were not considered for the second set of the parametric analysis in order to focus on the larger feasibility region.

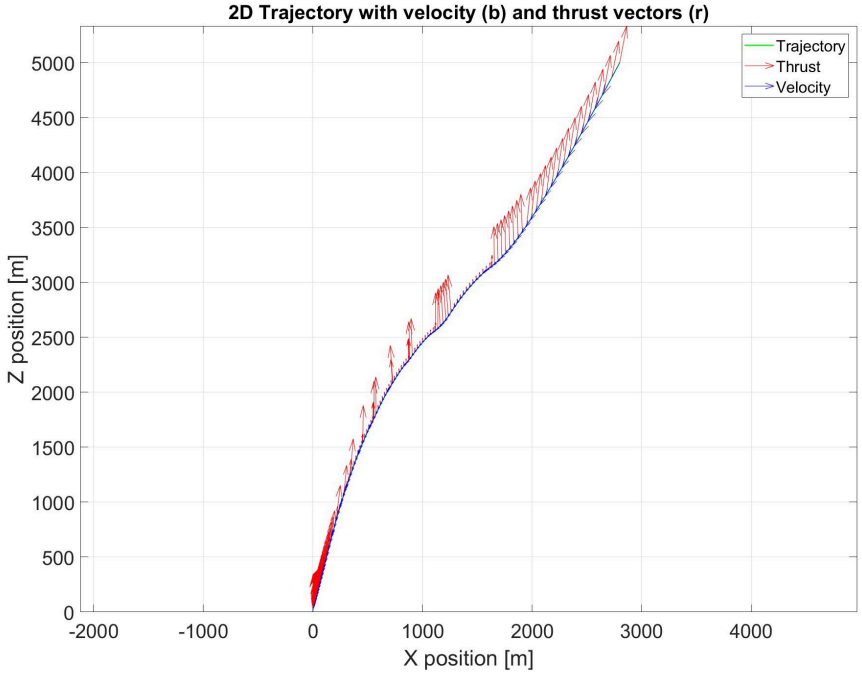
The second set of scenarios showed a more detailed shape of the feasibility region. This set contained 108 cases of which 35 were feasible. In Subfigure 9b), it is seen that for a lower PH, the feasibility conditions are met with a higher TH, where as if the PH increases, the TH needs to decrease. This is understood because the amount of fuel limits the feasibility region, causing problems with the TH. Likewise, the combination of UF and PH is considered important to maintain the feasibility of the problem, meeting the constraints of the problem. Once the initial PH reaches 40 s, the feasibility region splits in 2, decreasing in number the feasible cases and disappearing if going further, as seen in Subfigure 9a). The irregular shape of the feasibility region is caused by the absence of a constraint restraining the angle α in the optimiser. With a smaller α the aerodynamic forces caused by the atmosphere would have less impact on the performance of the convex optimiser as the differences in the state vector between the simplified dynamics of the optimiser and the simulator would be decreased. This is considered as part of the future steps to be performed.

4.2.2. Optimal trajectory

The last step left to discuss is the performance of the MPC algorithm during a feasible scenario. The selected case considers a $PH = 38$ s, $TH = 69$ s and $UF = 0.12$ and is part of the second set analysed in subsection 4.2.1.

Figure 10 shows the path followed by the RLV during the powered descent. The red vectors represent the thrust applied at each moment in time, with the most significant amount of thrust applied at the beginning and at the end of the landing. This is corroborated in Figure 11, which shows the state and control vectors of the landing case. In the Subfigure 11e), the thrust follows a bang-bang profile, with the thrust being at maximum during more time at the beginning and at the end of the simulation.

In Figure 10 and Subfigure 11a), the Altitude decreases rapidly during the first 2000 m, at the same time as the proposed MPC algorithm decreases the V_z, in Subfigure 11b), counteracting the initial V_z. It is worth noting that, between $t \approx 15$ s and $t \approx 42$ s, the V_z is maintained almost constant to allow the RLV to reach the landing target in the X-axis, see Subfigures 11a) to 11d), without loosing much Altitude. In the final part of the flight, the MPC reduces the magnitude of both V_x and V_z components and achieves a safe landing at the target position. The final state vector of



(a)

Figure 10: MPC: Landing trajectory with thrust vector.

the RLV is $[0.001\text{ m}, 0.243\text{ m}, 0.001\text{ m/s}, -1.152\text{ m/s}]$, meeting the margins mentioned, and the final mass is 26170 kg, which is greater than the dry mass of the first stage of the Falcon 9 rocket.

The instantaneous changes of the Thrust Angle θ seen in Subfigure 11f) are caused by the absence of an constraint on the initial control actions, and therefore at every step of the MPC algorithm the initial θ might vary from the previous orientation. It is part of future research to investigate how to include this constraint into the algorithm to ensure that the optimiser considers the current orientation of the RLV. Moreover the model adopted in Equation 1 does not consider the attitude dynamics of the RLV, meaning that the obtained θ is assumed to be instantaneously reached. In any case, the instantaneous jumps on θ are limited to ± 10 deg. It is reasonable to interpret that with the inclusion of the attitude dynamics into the simulation model, this angle would be reached within in a shorter period compared to the characteristic time of the simulation.

5. Conclusion

The present paper intended to introduce a new MPC strategy for the Powered Descend of a RLV problem. This strategy updates the prediction horizon with an update factor in order to land the RLV within a terminal horizon. The paper shows the benefits of this new strategy with respect to the classical MPC approach: no hovering overtime until the RLV runs out of fuel and better tuning of the MPC strategy by setting the initial PH, TH and UF values.

The numerical simulations and the tuning carried on within this paper shows that this approach is viable and promising. In the near future, the intention of the researchers is to update the simulation model into a 3-dimensional rigid-body model including the attitude dynamics and to include the initial value of the control law to apply at the prediction step, to avoid instantaneous jumps in the thrust direction and analyse how these changes affect the feasibility region in a more realistic case.

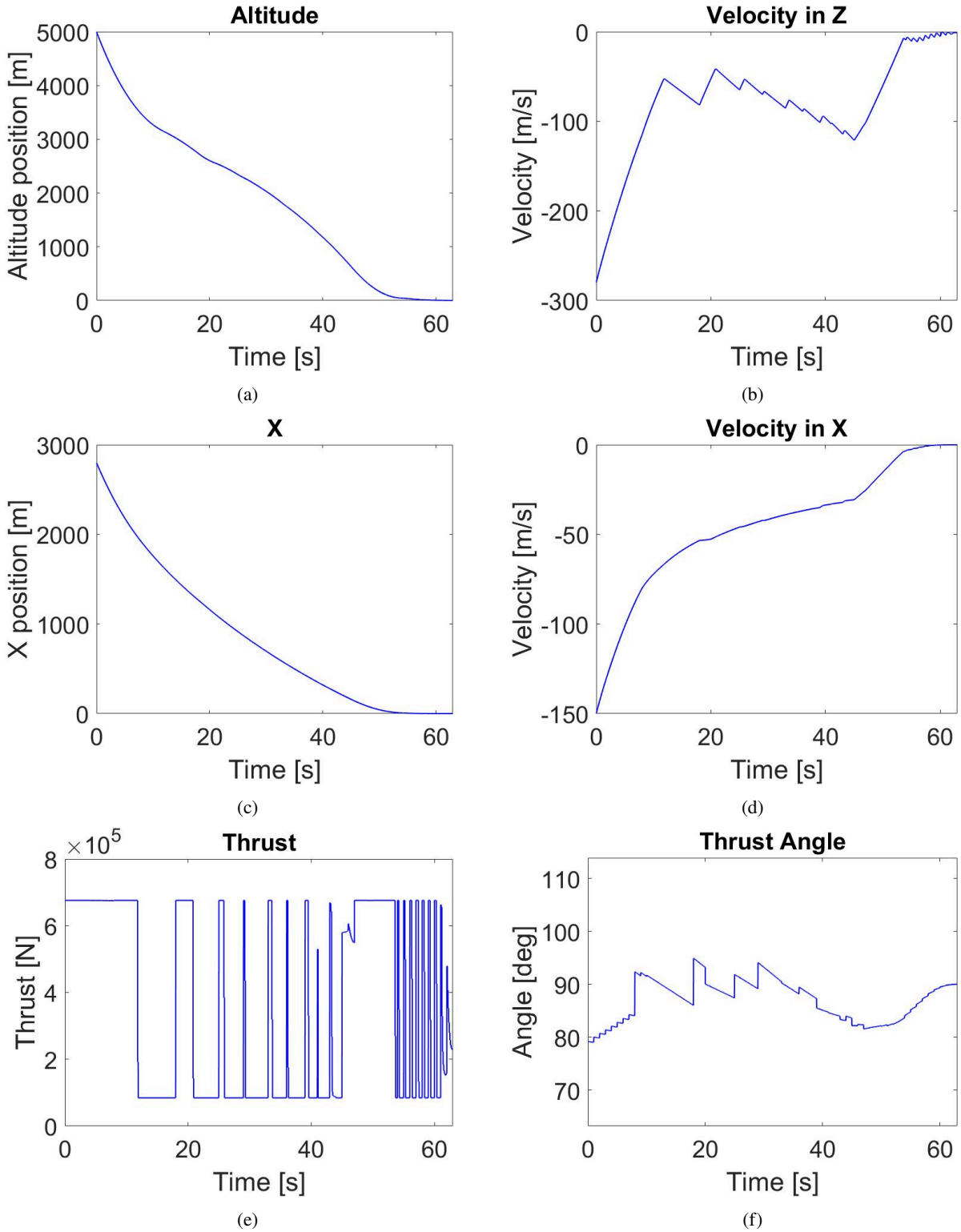


Figure 11: MPC: Landing variables.

References

[1] L. Blackmore. Autonomous precision landing of space rockets. 46:15–20, Jan. 2016.

[2] M. Gallaher, D. Coughlin, and D. Krupp. A guidance and control assessment of three vertical landing options for rlv. In *Guidance, Navigation, and Control Conference*, 1996. doi: 10.2514/6.1996-3702. URL https://arc.aiaa.org/doi/abs/10.2514/6.1996-3702.

[3] D. F. Lawden. *Optimal Trajectories for Space Navigation*. Butterworths, London, United Kingdom, 1963.

[4] J. S. Meditch. On the problem of optimal thrust programming for a lunar soft landing. *IEEE Transactions on Automatic Control*, 9(4):477–484, 1964.

[5] D. Eyles. *Sunburst and Luminary: An Apollo Memoir*. Fort Point Press, 2018.

[6] B. Açıkmeşe and S. Ploen. A powered descent guidance algorithm for mars pinpoint landing. In *Proc. AIAA Guidance, Navigation, and Control Conference and Exhibit*, San Francisco, United States of America, Aug. 2005. ISBN 978-1-62410-056-7. doi: 10.2514/6.2005-6288.

[7] S. Ploen, B. Acikmese, and A. Wolf. *A Comparison of Powered Descent Guidance Laws for Mars Pinpoint Landing*. doi: 10.2514/6.2006-6676. URL https://arc.aiaa.org/doi/abs/10.2514/6.2006-6676.

[8] M. Szmuk, T. P. Reynolds, and B. Açıkmeşe. Successive convexification for real-time six-degree-of-freedom powered descent guidance with state-triggered constraints. *Journal of Guidance, Control, and Dynamics*, 43(8):1399–1413, 2020. doi: 10.2514/1.G004549. URL https://doi.org/10.2514/1.G004549.

[9] X. Liu. Fuel-optimal rocket landing with aerodynamic controls. *Journal of Guidance, Control, and Dynamics*, 42(1):65–77, 2019.

[10] C. A. Pascucci, S. Bennani, and A. Bemporad. Model predictive control for powered descent guidance and control. In *2015 European Control Conference (ECC)*, pages 1388–1393, 2015. doi: 10.1109/ECC.2015.7330732.

[11] J. M. Maciejowski. *Predictive Control with Constraints*. Prentice Hall, 2000.

[12] D. P. Miotto and R. C. LePome. Design of a model predictive control flight control system for a reusable launch vehicle. *AIAA Guidance, Navigation, and Control Conference and Exhibit*, 2003.

[13] M. L. Jacopo Guadagnini and P. Rosa. Model predictive control for reusable space launcher guidance improvement. *71 International Astronautical Congress (IAC)*, 2020.

[14] U. of Nebraska-Lincoln. Definition of scale height. URL https://astro.unl.edu/naap/scaleheight/sh_bg1.html.

[15] P. Simplício, A. Marcos, and S. Bennani. Reusable launchers: Development of a coupled flight mechanics, guidance, and control benchmark. *Journal of Spacecraft and Rockets*, 57(1):74–89, 2020. doi: 10.2514/1.A34429. URL https://doi.org/10.2514/1.A34429.

[16] J. F. Sturm. Using sedumi 1.02, a matlab toolbox for optimization over symmetric cones. *Optimization Methods and Software*, 11(1-4):625–653, 1999. doi: 10.1080/10556789908805766. URL https://doi.org/10.1080/10556789908805766.

[17] Anonymous. *Falcon User's Guide*. SpaceX, 2021. URL https://www.spacex.com/media/falcon-users-guide-2021-09.pdf.

[18] Anonymous. Transporter-3 mission, . URL https://youtu.be/mFBeuSAvhUQ.

[19] Anonymous. Iridium next 3 data, . URL https://github.com/shahar603/Telemetry-Data/tree/master/Iridium%20NEXT%203.

[20] D. R. Williams. Earth fact sheet. URL https://nssdc.gsfc.nasa.gov/planetary/factsheet/earthfact.html.

A shrinking horizon model predictive control for landing of reusable launch vehicles

Zaragoza Prous, Guillermo

2022-09-22

Zaragoza Prous G, Felicetti L. (2022) A shrinking horizon model predictive control for landing of reusable launch vehicles. In: 73rd International Astronautical Congress (IAC-22), 18-22 September 2022, Paris, France

https://iac2022.org/