

Convex Trajectory Optimization and Model Predictive Control for Fuel-Optimal Rocket Landing

Elio Saade

School of Electrical, Computer and Energy Engineering

Arizona State University

Tempe, USA

esaade@asu.edu

Abstract—This paper presents a hierarchical guidance framework for the fuel-optimal landing of a rocket in a longitudinal plane. The landing problem is formulated as a free-final-time nonlinear optimal control problem with the objective of maximizing residual propellant mass at touchdown. A series of convexification techniques are applied to convert the nonlinear dynamics into a sequence of convex subproblems solvable via second-order cone programming. The resulting fuel-optimal trajectory serves as a reference for a Model Predictive Control (MPC) scheme, which tracks the reference at a finer guidance frequency by solving a constrained quadratic program over a receding horizon at each timestep. Both the trajectory optimization and MPC problems are implemented in Python using the CVXPY library and validated through numerical simulation. Results demonstrate that the trajectory optimizer converges in nine iterations to a fuel-optimal solution, and the MPC controller successfully tracks the reference trajectory.

Index Terms—Rocket Landing, Fuel-Optimal, Trajectory Optimization, MPC, Convexification

I. INTRODUCTION

The recovery and reuse of orbital-class rocket boosters has emerged as a defining challenge in modern aerospace engineering. Pioneered by SpaceX, propulsive vertical landing has demonstrated that recovering first-stage boosters is not only technically feasible but economically transformative, drastically reducing the cost of access to space. At the heart of this capability lies a sophisticated guidance system that must compute and execute a safe, fuel-efficient landing trajectory in real time, subject to strict physical and operational constraints.

The rocket landing problem is fundamentally an optimal control problem: given a set of initial conditions — altitude, velocity, flight-path angle, and propellant mass — the system must determine a sequence of thrust commands that bring the vehicle to a controlled, near-zero-velocity touchdown at a designated landing site while consuming as little fuel as possible. This problem is inherently challenging due to the nonlinear dynamics governing atmospheric flight, the coupling between trajectory shape and propellant consumption, and the need to satisfy hard constraints on thrust magnitude, attitude, and final state.

Two complementary approaches are central to addressing this challenge. First, trajectory optimization techniques are used offline (or at low frequency) to generate a fuel-optimal reference trajectory from the initial conditions to the landing

site. Second, Model Predictive Control (MPC) is employed online to track this reference trajectory in real time, correcting for deviations while respecting system constraints. Together, these methods form a hierarchical guidance architecture.

In this work, the fuel-optimal vertical landing of a rocket in a longitudinal plane is studied. The rocket dynamics are modeled as a nonlinear system. The trajectory optimization problem is formulated as a free-final-time optimal control problem with the objective of maximizing the remaining propellant at touchdown. Because this problem is non-convex, a series of convexification techniques are applied. The resulting convex problem, solvable via second-order cone programming, is solved iteratively until convergence. The optimal trajectory is then used as a reference for a linear MPC controller, which re-optimizes control inputs at each guidance step over a receding horizon. The MPC objective penalizes both state and control deviations from the reference, with an additional terminal cost term to enforce accurate arrival at the landing site.

The rest of the document is organized as follows: section II introduces the rocket dynamic model, section III derives the fuel-optimal trajectory optimization problem, section IV discusses the Model Predictive Control reference tracking problem, section V presents the simulation results, and VI concludes the work and shares future research directions.

II. SYSTEM DYNAMICS

In this work, a rocket's two-dimensional flight over a flat Earth is considered. Most of the problem formulations are based on [1] and [2]. It is assumed that the thrust direction of the engine always coincides with the rocket's axial direction, and the lift force is neglected. Also, the gravitational acceleration is taken to be constant given the altitudes studied, i.e., $g \approx g_0$. As such, the dynamic equations of rocket landing are presented in Eq. (1). The model is nonlinear of the form

$$\frac{dx}{dt} = f(x, u).$$

$$\begin{aligned}\dot{r} &= V \sin \gamma \\ \dot{s} &= V \cos \gamma \\ \dot{V} &= \frac{-T \cos \alpha - D}{m} - g_0 \sin \gamma \\ \dot{\gamma} &= \frac{-T \sin \alpha}{mV} - \frac{g_0 \cos \gamma}{V} \\ \dot{m} &= -\frac{T}{g_0 I_{sp}}\end{aligned}\quad (1)$$

Where r is the radial distance from the Earth's center to the rocket, s is the downrange, V is the velocity, γ is the flight-path angle measured from the horizon to the velocity vector, m is the rocket mass, T is the thrust magnitude, α is the angle of attack, D is the atmospheric drag force, g_0 is the standard acceleration due to Earth's gravity, and I_{sp} is the specific impulse of the rocket engines. In the model, the control variables are T and α .

The atmospheric drag is a function of V and is expressed in Eq. (2).

$$\begin{aligned}D &= 0.5 \rho V^2 S_{ref} C_D \\ \rho &= \rho_0 \exp(-\beta h) \\ h &= r - R_0\end{aligned}\quad (2)$$

Where ρ is the air density, S_{ref} is the reference area, C_D is the drag coefficient, ρ_0 is the air density at sea level, β is the density scale, h is the flight altitude, and R_0 is the Earth radius.

III. TRAJECTORY OPTIMIZATION

A. Optimal Control Problem

The first optimization problem deals with generating the optimal reference trajectory for the rocket. The optimality criterion is selected as minimizing the fuel consumption during the landing process. In relation to Eq. (1), this is equivalent to maximizing the final mass of the rocket, or minimizing the objective function in Eq. (3).

$$J = -m_f \quad (3)$$

Thus, the trajectory optimization problem is described below and is denoted as *Problem 0*.

Problem 0 :

$$\min -m_f$$

subject to

Eq.(1)

$$x_0 = [r_0, s_0, V_0, \gamma_0, m_0] \quad (4)$$

$$x_f = [r_f, s_f, V_f, \gamma_f, m_f] \quad (5)$$

$$\gamma_f = -90^\circ \quad (6)$$

$$V_f \leq V_{safe} \quad (7)$$

$$m_f \geq m_{dry} \quad (8)$$

$$\alpha_{f,min} \leq \alpha_f \leq \alpha_{f,max} \quad (9)$$

$$T_{min} \leq T \leq T_{max} \quad (10)$$

$$-\alpha_{safe} \leq \alpha \leq \alpha_{safe} \quad (11)$$

The safe landing process is subject to the rocket dynamics (Eq. (1)), initial conditions (Eq. (4)), terminal constraints (Eq. (5), (6), (7), (8), (9)), as well as control limits (Eq. (10), (11)).

B. Convexification

1) *Change of Variables:* As a first step in the convexification of the problem, a change of variables is introduced with the goal of making the dynamics affine in the controls.

$$\begin{aligned}z &= \ln(m) \\ u_1 &= \frac{T \cos \alpha}{m} \\ u_2 &= \frac{T \sin \alpha}{m} \\ u_3 &= \frac{T}{m}\end{aligned}\quad (12)$$

With the additional constraint that $u_1^2 + u_2^2 = u_3^2$. [2] shows that the optimal solution is always active on the boundary of the second order cone of this constraint; so, it can be relaxed as $u_1^2 + u_2^2 \leq u_3^2$. The dynamics model is updated in Eq. (13).

$$\begin{aligned}\dot{r} &= V \sin \gamma \\ \dot{s} &= V \cos \gamma \\ \dot{V} &= -u_1 - De^{-z} - g_0 \sin \gamma \\ \dot{\gamma} &= -\frac{u_2}{V} - \frac{g_0 \cos \gamma}{V} \\ \dot{z} &= -\frac{u_3}{g_0 I_{sp}}\end{aligned}\quad (13)$$

2) *Terminal Time:* The fuel consumption is related to the flight time. In *Problem0*, flight time is not optimized; hence, it is a terminal time free optimization problem, which is non-convex. To address this, the time $[t_0, t_f]$ is mapped to a new domain $\tau \in [0, 1]$, and t_f is included as an optimization variable.

$$\tau = \frac{t - t_0}{t_f - t_0} \quad (14)$$

The modified dynamics model gets differentiated by τ , $\frac{dx}{d\tau} = g(x, u)$, and is present in Eq. (15).

$$\begin{aligned}\dot{r} &= t_f V \sin \gamma \\ \dot{s} &= t_f V \cos \gamma \\ \dot{V} &= t_f \left[-u_1 - De^{-z} - g_0 \sin \gamma \right] \\ \dot{\gamma} &= t_f \left[-\frac{u_2}{V} - \frac{g_0 \cos \gamma}{V} \right] \\ \dot{z} &= -t_f \frac{u_3}{g_0 I_{sp}} \\ \dot{t} &= t_f\end{aligned}\quad (15)$$

Where the vector of states becomes $x = [r, s, V, \gamma, z, t]^T$, and the vector of controls becomes $u = [u_1, u_2, u_3, t_f]^T$.

3) *Linearization*: The nonlinearity of the dynamics equations in the states is another source of non-convexity. Successive linearization is performed along the trajectory, that is:

$$\dot{x} = A(x^k, u^k)x + B(x^k, u^k)u + C(x^k, u^k) \quad (16)$$

Where k indicates the optimization iteration, not the discrete time step.

$$A(x^k, u^k) = \left. \frac{\partial g(x, u)}{\partial x} \right|_{x^k, u^k} \quad (17)$$

$$A = \begin{bmatrix} 0 & 0 & t_f \sin \gamma & t_f V \cos \gamma & 0 & 0 \\ 0 & 0 & t_f \cos \gamma & -t_f V \sin \gamma & 0 & 0 \\ 0 & 0 & -2t_f \frac{D}{V} e^{-z} & -t_f g_0 \cos \gamma & t_f D e^{-z} & 0 \\ 0 & 0 & \frac{t_f (u_2 + g_0 \cos \gamma)}{V^2} & \frac{t_f g_0 \sin \gamma}{V} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$B(x^k, u^k) = \left. \frac{\partial g(x, u)}{\partial u} \right|_{x^k, u^k} \quad (18)$$

$$B = \begin{bmatrix} 0 & 0 & 0 & V \sin \gamma \\ 0 & 0 & 0 & V \cos \gamma \\ -t_f & 0 & 0 & -u_1 - D e^{-z} - g_0 \sin \gamma \\ 0 & -\frac{t_f}{V} & 0 & -\frac{u_2}{V} - g_0 \cos \gamma \\ 0 & 0 & -\frac{t_f}{g_0 I_{sp}} & -\frac{u_3}{g_0 I_{sp}} \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$C(x^k, u^k) = g(x^k, u^k) - A(x^k, u^k)x^k - B(x^k, u^k)u^k \quad (19)$$

Moreover, Eq. (20) presents the linearization of the term e^{-z} .

$$e^{-z} = e^{-z^k} - e^{-z^k}(z - z^k) \quad (20)$$

C. Convex Optimization Problem

Following the convexification steps, the convex trajectory optimization problem, denoted as *Problem 1* is presented:

Problem 1 :

min $-z_f$

subject to Eq.(16)

$$x_0 = [r_0, s_0, V_0, \gamma_0, z_0, t_0]$$

$$x_f = [r_f, s_f, V_f, \gamma_f, z_f, t_f]$$

$$\gamma_f = -90^\circ$$

$$V_f \leq V_{safe}$$

$$z_f \geq z_{dry}$$

$$\tan(\alpha_{f,min}) \leq \frac{u_2}{u_1} \Big|_{t_f} \leq \tan(\alpha_{f,max})$$

$$T_{min} e^{-z} \leq u_3 \leq T_{max} e^{-z}$$

$$\tan(\alpha_{min}) \leq \frac{u_2}{u_1} \leq \tan(\alpha_{max})$$

$$u_1^2 + u_2^2 \leq u_3^2$$

$$t_f \geq 5$$

The solution procedure of the optimization problem requires discrete time steps for the continuous-time dynamics of Eq. (16). Therefore, the dynamics are discretized via the trapezoidal rule. If we consider N discrete intervals defined by $N+1$ points, and denote the step size of each interval as Δt_i , then, following the trapezoidal rule, the discrete-time linear system can be written as:

$$x_i = x_{i-1} + \frac{\Delta t_i}{2}(\dot{x}_{i-1} + \dot{x}_i), \quad i = 1, \dots, N \quad (21)$$

Where i denotes the discrete time step index.

The solution of the optimization problem consists of the states and controls at each discrete point; that is, $6N$ state variables corresponding to N state vectors $[r_i, s_i, V_i, \gamma_i, z_i, t_i]$, and $4N$ control variables corresponding to N control vectors $[u_{1,i}, u_{2,i}, u_{3,i}, t_{f,i}]$. It is important to note that the last element of all the control vectors must be the same since it corresponds to t_f . This is added as a constraint in the implementation.

IV. MPC REFERENCE TRACKING

A. Model Predictive Control

Model Predictive Control (MPC) is a control strategy that leverages an explicit model of the system to optimize the control inputs over a finite horizon of length M [3]. At each sampling instant, MPC solves a constrained optimization problem that satisfies the system dynamics and minimizes a predefined objective function. It uses the current state of the plant as the initial state; the optimization yields an optimal control sequence and only the first control of the solution is applied. Then, the prediction horizon of the MPC is moved one step forward and the optimization is repeated. One major advantage of MPC is that it can naturally incorporate constraints on states and controls, and implement different objective functions to target a variety of controller behaviors.

A typical objective is to minimize the tracking error to a reference trajectory of states and controls:

$$J = \sum_{j=1}^M \Delta x_j^T Q \Delta x_j + \Delta u_j^T R \Delta u_j \quad (22)$$

$$\Delta x_j = x_j - x_j^{ref}$$

$$\Delta u_j = u_j - u_j^{ref}$$

Where $Q \succeq 0$ is the state weighting matrix (or state penalty matrix) and $R \succeq 0$ is the control weighting matrix.

B. Reference Tracking

The solution of *Problem 1* is used as a reference trajectory for MPC. The MPC follows the dynamics model in Eq. (13). A modification is made to the general tracking objective function by separating the cost of the last state of the horizon from the rest of the states; this is known as the "terminal cost". The objective function of MPC with terminal cost is presented in Eq. (23).

$$J = \sum_{j=1}^{M-1} \left[\Delta x_j^T Q \Delta x_j + \Delta u_j^T R \Delta u_j \right] + \Delta x_M^T Q_f \Delta x_M \quad (23)$$

$$\Delta x_j = x_j - x_j^{ref}$$

$$\Delta u_j = u_j - u_j^{ref}$$

Where $Q_f \succeq 0$ is the terminal cost matrix. Adding a terminal cost allows to penalize the deviation from the last state of the horizon further than the other states, incentivizing the controller to track the last state more accurately. This is necessary in the rocket landing problem in order to land precisely at the target landing site.

The MPC optimization problem is presented below and labeled *Problem 2*.

Problem 2 :

min Eq.(23)

subject to Eq.(13)

$$x_0 = [r_0, s_0, V_0, \gamma_0, z_0]$$

$$T_{min} e^{-z} \leq u_3 \leq T_{max} e^{-z}$$

$$\tan(\alpha_{min}) \leq \frac{u_2}{u_1} \leq \tan(\alpha_{max})$$

$$u_1^2 + u_2^2 \leq u_3^2$$

Problem 2 is solved at every rocket guidance interval and the first step of the control solution is applied.

V. SIMULATION RESULTS

This section presents the numerical simulations of the fuel optimal trajectory optimization (*Problem 1*), as well as the performance of the MPC in tracking the optimal reference trajectory (*Problem 2*). The rocket parameters used are presented in Table I.

TABLE I
ROCKET PARAMETERS

Parameter	Value
m_{dry}	49000kg
S_{ref}	8m ²
C_D	0.25
I_{sp}	443s
T_{min}	412.7kN
T_{max}	1375.6kN
α_{min}	-10°
α_{max}	10°

All simulations are coded in Python, and the CVXPY [4] library is used to implement the optimization problems. The Splitting Conic Solver (SCS) of CVXPY is used to solve all the optimizations. The full implementation is found in the following GitHub repository .

A. Fuel Optimal Trajectory

The fuel optimal trajectory optimization of *Problem 1* is run with $N = 31$ discrete points, corresponding to 30 discrete intervals, starting from the initial conditions in Table II. The final radial distance and downrange are $r_f = R_0$ and $s_f = 1000m$. The safe limits for the final velocity and angle of attack are $V_{safe} = 1m/s$ and $\alpha_{safe} = 2^\circ$.

TABLE II
INITIAL CONDITIONS

r_0	s_0	V_0	γ_0	m_0	t_0
3000 + R_0 m	0m	280m/s	-65°	55000kg	0s

The optimization is terminated when the maximum absolute change in states and controls is smaller than a tolerance $\epsilon = 10^{-2}$:

$$\max_i |x_i^{k+1} - x_i^k| \leq \epsilon, \quad \max_i |u_i^{k+1} - u_i^k| \leq \epsilon \quad (24)$$

The optimization converges after 9 iterations. The optimal flight time is $t_f = 19.27s$, which corresponds to $dt = 0.642s$ for 31 intervals. The final mass of the rocket is $m_f = 49614.4kg$, meaning that the rocket requires 5385.6kg of fuel to perform the landing. Figure 1 shows the states of the optimal trajectory, while Figure 2 shows the actions of the optimal trajectory. The circular markers represent the 31 discrete points.

The optimal thrust profile is particularly interesting. It exhibits a switching behavior, beginning near the minimum thrust level T_{min} and transitioning to near-maximum thrust T_{max} as the rocket approaches touchdown. This structure is consistent with the bang-bang nature of fuel-optimal control solutions, where the control input tends to operate at its bounds due to the linear dependence of the cost on the input. In this case, the initial low-thrust phase allows the vehicle to descend efficiently, while the final high-thrust phase provides the necessary deceleration to achieve a soft landing.

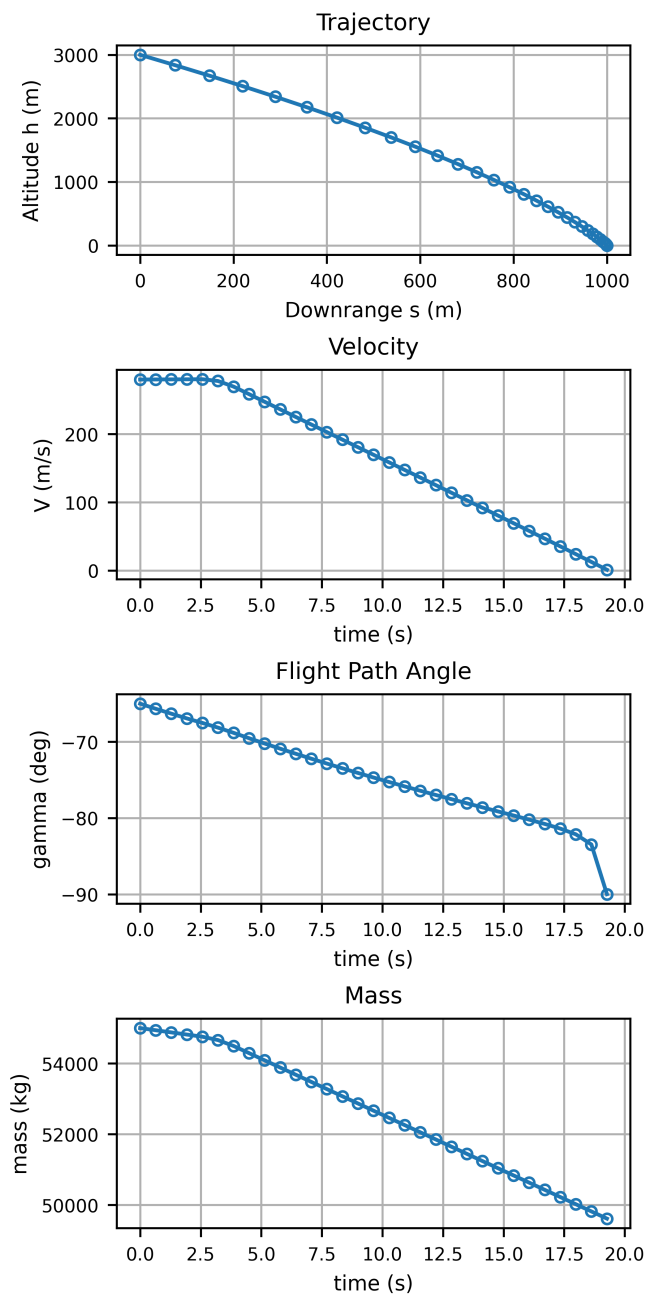


Fig. 1. States of the fuel-optimal reference trajectory. The circular markers represent the discrete time points of the optimization problem, i.e., the solution points.

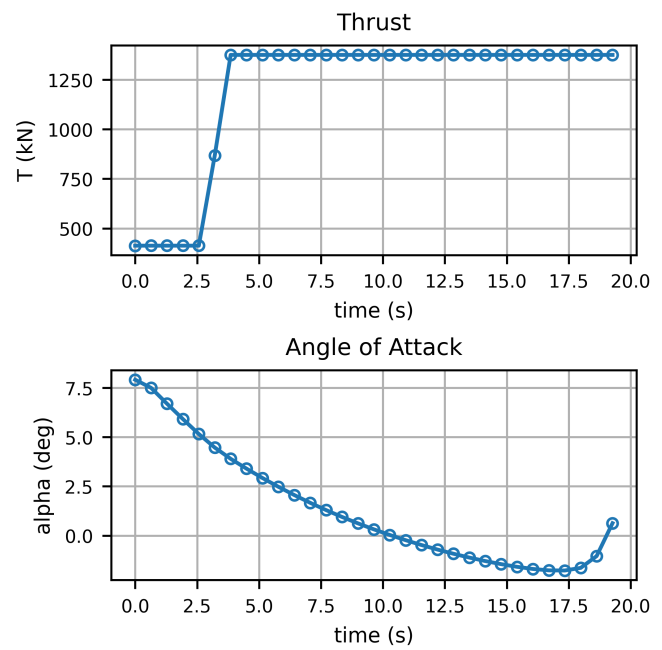


Fig. 2. Controls of the fuel-optimal reference trajectory. The circular markers represent the discrete time points of the optimization problem. The optimal thrust profile follows a bang-bang pattern, where the initial minimum-thrust phase provides an efficient decent, while the maximum-thrust phase decelerates the rocket to a soft touchdown.

B. Reference Tracking

The trajectory obtained in the previous section is used as a reference for the MPC optimization of *Problem 2*. The guidance interval for MPC is selected to be $dt_{MPC} = 0.1s$. Because of the discrepancy in timestep size between the reference trajectory (0.642s) and the MPC (0.1s), the reference trajectory is interpolated into a time interval $[0, 19.3s]$ with steps of 0.1s, which generates a reference trajectory of 194 points.

The MPC horizon is chosen as $M = 20$, which covers 2s of future trajectories at every iteration. Since the reference trajectory has a finite number of points, the size of the horizon is reduced to match the remaining points at the end of the trajectory. In other words, when the last 19 points of the reference trajectory are reached, the MPC horizon becomes $M = 19, 18, 17 \dots 2$. This results in solving the MPC optimization of *Problem 2* for 193 times.

The state and control weighting matrices, Q and R are initialized following Bryson's rule, that is, each diagonal entry of the matrices corresponds to the inverse of the square of the maximum acceptable tracking error for that variable.

$$Q_{ii} = \frac{1}{(\Delta x_i^{max})^2}, \quad R_{jj} = \frac{1}{(\Delta u_j^{max})^2} \quad (25)$$

Manual fine-tuning is then performed to obtain the desirable MPC performance. The terminal cost matrix Q_f is set by scaling the matrix Q by a factor of 100, with the goal of enforcing landing accuracy. The tuned weighting matrices are presented in Eq. (26).

$$Q = \begin{bmatrix} \frac{1}{1^2} & 0 & 0 & 0 & 0 \\ 0 & \frac{1}{10^2} & 0 & 0 & 0 \\ 0 & 0 & \frac{1}{20^2} & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{0.1^2} & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{0.05^2} \end{bmatrix} \quad (26)$$

$$R = \begin{bmatrix} \frac{1}{10^2} & 0 & 0 \\ 0 & \frac{1}{5^2} & 0 \\ 0 & 0 & \frac{1}{10^2} \end{bmatrix}$$

$$Q_f = 100Q$$

Figure 3 shows the states tracking performance of the MPC, while Figure 4 shows the controls used. The MPC follows the reference trajectory well, while keeping the controls stable and close to the reference. The final landing states using the MPC controller are presented in Table III. The results of Figure 3, Figure 4, and Table III show that the MPC controller successfully tracks the reference trajectory, achieving touchdown within centimeters of the target site with a velocity of 0.4 m/s and a flight path angle of -90.8° .

TABLE III
MPC FINAL VALUES

r_f	s_0	V_0	γ_0	m_0
$(-0.2 + R_0) m$	1000.3m	$-0.4m/s$	-90.8°	49581.1kg

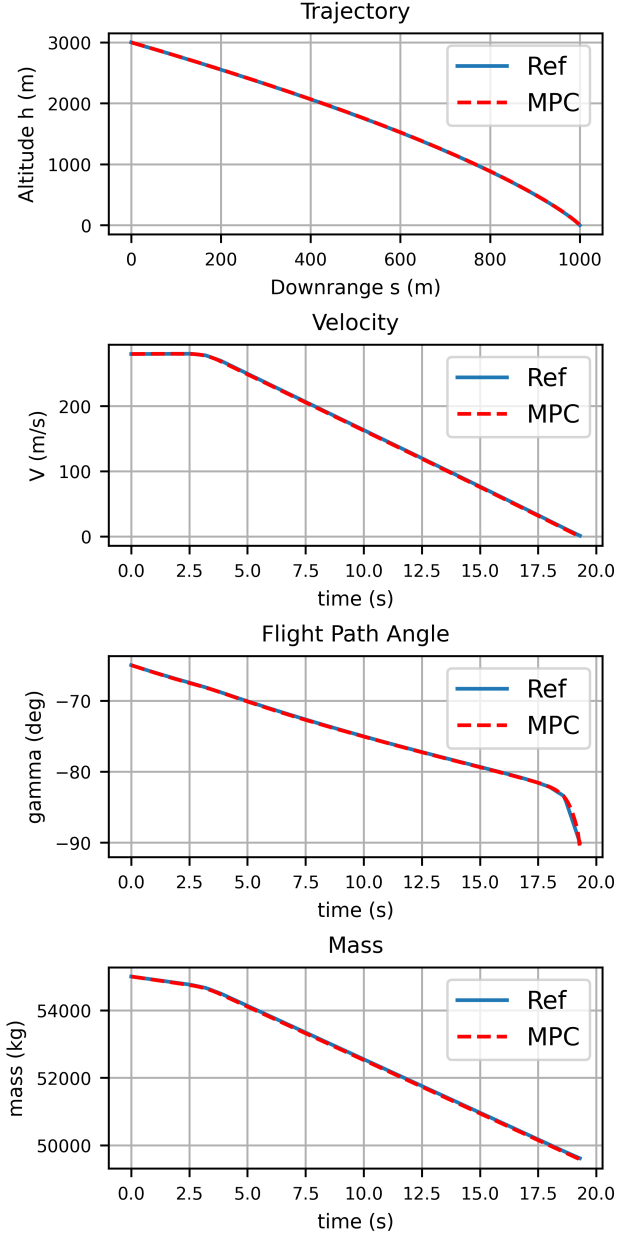


Fig. 3. Performance of the MPC reference tracking algorithm. The MPC follows the reference states accurately and performs a soft landing within centimeters.

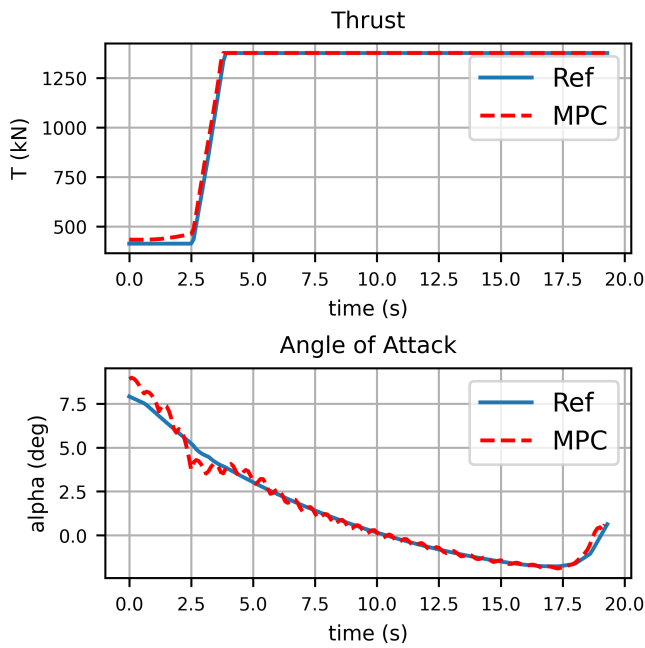


Fig. 4. Controls of MPC reference tracking algorithm. The controls remain stable and particularly close to the reference.

It should be noted that the simulations presented assume a perfect model with no external disturbances, sensor noise, or model uncertainty. In practice, a rocket during landing is subject to wind gusts, atmospheric density variations, and actuator imperfections, all of which would degrade tracking performance. The results presented here therefore represent an upper bound on achievable performance.

VI. CONCLUSION

In this work, the fuel-optimal landing of a rocket in a longitudinal plane is studied. The reference trajectory optimization problem is presented, along with nonlinear dynamics and constraints. Then, several steps are taken to convexify the problem. Moreover, a model predictive controller optimization problem is formulated to track the reference trajectory generated. Both problems are solved in simulation and the results are presented. The studied optimization problems are crucial to the guidance systems of reusable rockets. This work presents a stepping stone, but has limitations. Future work can study the effect of disturbances on the performance of the MPC. Moreover, other MPC formulations can be tested; for example, one that has the objective of minimizing fuel for the remaining trajectory instead of tracking a reference. Finally, the execution speed of the algorithms can be tested on embedded hardware with the goal of improving the execution speed for real-time implementation.

REFERENCES

- [1] C. Wang and Z. Song, "Convex Model Predictive Control for Rocket Vertical Landing," in 2018 37th Chinese Control Conference (CCC), Jul. 2018, pp. 9837–9842. doi: 10.23919/ChiCC.2018.8483147.
- [2] X. Liu, "Fuel-Optimal Rocket Landing with Aerodynamic Controls," *Journal of Guidance, Control, and Dynamics*, vol. 42, no. 1, pp. 65–77, Jan. 2019, doi: 10.2514/1.G003537.
- [3] D. Q. Mayne, J. B. Rawlings, C. V. Rao, and P. O. M. Scokaert, "Constrained model predictive control: Stability and optimality," *Automatica*, vol. 36, no. 6, pp. 789–814, Jun. 2000.
- [4] S. Diamond and S. Boyd, "CVXPY: A Python-embedded modeling language for convex optimization," *Journal of Machine Learning Research*, vol. 17, no. 83, pp. 1–5, 2016.