Guidance, Navigation, and Control enabling Retrograde Landing of a First Stage Rocket

Christian Canham Meaghan Podlaski Luigi Vanfretti

Department of Electrical, Computer, and Systems Engineering
Rensselaer Polytechnic Institute
Troy, NY, United States
{canhac, podlam, vanfrl}@rpi.edu

Abstract

A Modelica model of a of a rocket's first stage is developed, designed to be representative of the launch vehicles in use in the United States in the late 2010s. The model uses initial conditions similar to those observed immediately after a second stage separation at 166 km altitude. A control system is developed enabling the rocket first stage to land back on Earth's surface at a predetermined landing pad in a controlled manner. The control system is evaluated based on its ability to compensate for altered initial conditions, as well as its ability to minimize acceleration forces and fuel consumption. The flight path of the simulated first stage rocket is compared to real-life telemetry data from a first stage rocket landing showing a similar trajectory.

Keywords: Rocket, Flight controller, GN&C, Retrograde Landing, Reaction Control Systems

1 Introduction

List of Acronyms and Definitions

Acronyms

GN&C · Guidance, Navigation, and Control
HIL · Hardware-in-the-Loop
IMU · Inertial Measurement Unit
LEO · Low Earth Orbit
NASA · National Aeronautics and Space Administration
PD · Proportional and Derivative Controller
RCS · Reaction Control System
STS · Space Transport System

Motivation

Rapid reusable space launch vehicles have long been a pursuit in the United States since it would dramatically increase accessibility to space. This was partially achieved with NASA's Space Transportation System (STS) Space Shuttle but failed to offer a fully reusable or low cost method. Rocket re-usability made significant strides when private space launch companies, including SpaceX and Blue Origin, demonstrated the ability to recover the first stage of the rocket by vertically landing it back on Earth's surface. This is achieved by relighting the rocket's engines in retrograde long enough to remove its horizontal and vertical velocities. This paired with gimbaled engines and control surfaces, such as grid fins, allow the first stage to be maneuvered back to a predetermined landing pad. The flight controller is responsible for making these engine and control surface control adjustments using input data from an inertial measurement unit (IMU) and GPS positional data.

Modeling and simulation of the launch vehicle is critical in the development in the GN&C control system. Hardware-in-the-loop (HIL) test beds are often times created to offer a low cost and rapid platform for the design of the control system and the calibration of their parameters before moving onto developmental prototypes.

The Modelica first stage rocket model and subsequent control system in this work represents early phase developmental activities that might occur when studying the feasibility of certain flight maneuvers. In this case, landing a rocket back on Earth after launching a payload into orbit. Many simplifications and assumptions are made in the first stage rocket model including the simplification of the rocket solely operating in the X-Z plane. Nevertheless, the control system core principles are fundamental and could be expanded to address these assumptions as the model grows in complexity. The key principle is to develop this control system despite these simplifications made in the model and show the ability to add features over time.

While Modelica models for a variety of aerospace applications have been successfully demonstrated (Wei et al. 2015; Re 2011; Briese, Klöckner, and Reiner 2017; Milz et al. 2019; Batteh et al. 2018; Posielek 2018; Hellerer, Bellmann, and Schlegel 2014), to the knowledge of the authors, there are no publicly available models similar to the one proposed in this work. The authors' believe that the growing interest and on-going advancements on rocket re-usability can benefit from the availability of an open source model that allows interested parties to exploit the advantages offered by object-oriented modeling provided by Modelica tools.

Contribution

This work is relevant to a user of the Modelica language looking for ways to rapidly develop a control system. In this case the control system is developed for a first stage rocket falling back to Earth but similar methods could be applied to other challenging control problems. The work shows how a simplified rocket model is nevertheless, an effective test-bed for the development of a control system that guides the rocket on a trajectory similar to that used by actual rockets, such as SpaceX's Falcon 9. The main contributions of the paper are the following:

- Implementation of the control system needed to recover a first stage rocket by vertically landing at a predetermined landing pad.
- Demonstration of Modelica's flexibility in creating models with increasing complexity leading to meaningful simulations.
- Documenting an open-source Modelica-based implementation of the aforementioned models available online at: `https://github.com/ALSETLab/RocketLanding`

Paper Organization

The paper is broken down in the following sections: Section 2 describes the first stage rocket model and the surrounding environment it operates in. Section 3 describes the control system designed to recover the first stage rocket by vertically landing at a predetermined landing pad. Section 4 compares the simulated Modelica rocket landing trajectory to real telemetry data from a SpaceX Falcon 9 rocket landing. Finally, Section 5 concludes the work.

2 Modelica First Stage Rocket Model

The model developed represents the first stage of a SpaceX Falcon 9 rocket immediately after second stage separation. It uses similar mass properties and initial conditions to those observed from publicly available SpaceX telemetry data (Pelham 2020). The model also includes the rocket engines and grid fins which are necessary to maneuver the rocket to the landing pad. Lastly the operating environment is taken into account by modeling drag on the vehicle as it falls through the Earth's atmosphere. The purpose was to design this model to be as representative of the actual rocket while making key simplifications that allow for the implementation of the control system presented in Section 3. The complete rocket model is shown in Figure 1.

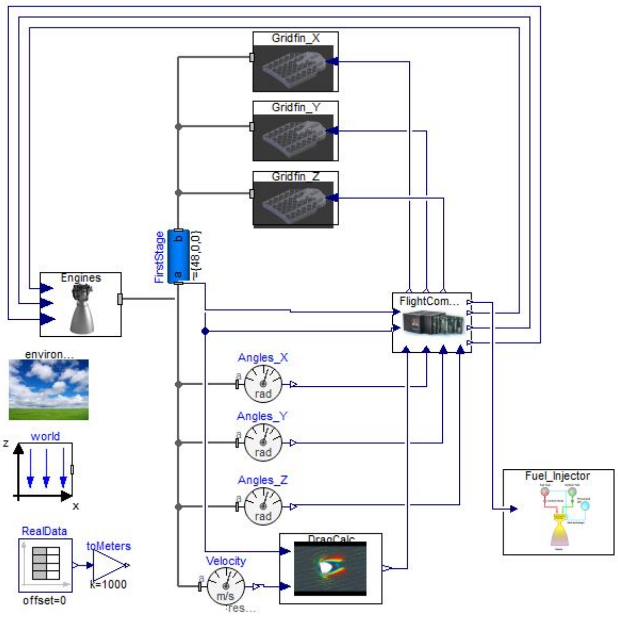

Figure 1. First stage rocket model

2.1 Mass Properties and Initial Conditions

The first stage rocket is modeled with a cylinder with a diameter of 3.7m and a length of 48 m. A density of 0.3g/cm^3 was selected based on the Falcon 9 rocket being composed of principally aluminum but with a mostly hollow interior.

In this model the frame of reference is the surface of the Earth which is simplified with a flat plane. The rocket is also assumed to be bounded to the X-Z plane. These two simplifications allow for easier control system development in Section 3 and more easily defined initial conditions. The initial conditions for the rocket are determined from live telemetry data from SpaceX's own webcasts (Pelham 2020). The simulation starts immediately after the second stage separation, where the first stage is assumed to be in orbit, and therefore, the vertical velocity in a flat Earth frame of reference is set to zero. The rocket fuselage also starts parallel to the Earth's surface. The initial velocity is purely measured by the horizontal component of 263 m/s in the X-axis. The altitude is set at 166km which is representative of the Falcon 9 stage separation for payload deployment in low earth orbit (LEO).

2.2 Engines and Grid Fins

The model includes a single rocket engine with a maximum thrust of 914 kN. This engine is simplified as a force acting in line with the fuselage of the rocket. The Falcon 9 rocket includes nine engines each with a maximum thrust of 914 kN but only one engine is typically used for landing.

The fuel consumption of the rocket engine is modeled in Figure 2 using fuel tanks and release valves that control the injection of liquid kerosene and oxygen into the engine at a proportion of 2:1 respectively. The same flight controller output that controls the simulated thrust from the engine is also used to control the fuel injector valves that

DOI
10.3384/ecp21181171

release the fuel into the engine simulated with two tanks at ambient pressure. The flow volume through each valve is recorded and can be observed during the rocket simulation.

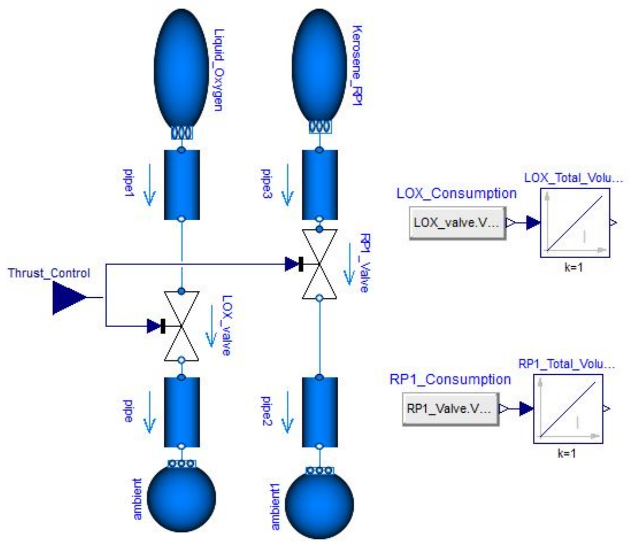

Figure 2. Fuel injector model

The model includes grid fins which impart a torque on the rocket depending on the grid fin's rotation angle relative to the fuselage. This allows the rocket to "steer" itself to a desired landing zone. On the Falcon 9 rocket these grid fins act as a control surface that redirect airflow therefore they are only effective at lower altitudes where the air is more dense. In the upper atmosphere Reaction Control Systems (RCS) are used to redirect the space vehicle using jets of compressed nitrogen. Both these actuators have the same intended effect so for simplification this model only uses the grid fins which are assumed to be equally effective at all altitude.

The torque on the rocket is a result of the rotating grid fins depending on the angle of the grid fin relative to the first stage fuselage. Maximum torque is imparted when the grid fin is at $\pm 45°$ which decreases closer to zero degrees modeled as a sine function. At zero degrees the grid fin is perpendicular to the fuselage and no torque is applied.

The model includes three grid fins mounted 120° apart on the circumference of the rocket. Since the rocket is assumed to operate only in the X-Z plane only one grid fin is used to control the rockets rotation. The other two are simply used to keep the rocket from drifting out of the X-Z plane due to numerical artifacts in the simulation.

2.3 Reentry Drag

A space vehicle reentering the Earth's atmosphere experiences a considerable amount of drag and successive heating as a result of the high speeds acquired during orbital insertion - this velocity term is squared in the drag equation. If the reentry trajectory is optimized this inherit drag will help slow down the the vehicle reducing the need to use as much fuel with the rocket engines. For these reasons it is critical to model the atmospheric drag with accuracy.

The drag of the first stage rocket is modeled with a force vector normal to the direction of the rocket's motion. This drag force is calculated using equation 1 where the area A is calculated based on the rockets angle relative to the direction of motion. Velocity V accounts for all vertical and horizontal velocity components. The coefficient of drag was estimated based on the shape of the cylindrical aluminum fuselage. Selecting this coefficient of drag to be constant is a critical simplification. In reality this coefficient of drag would change based on the wake created behind the falling first stage. The air density is calculated using equation 2 where h is the altitude above sea level. Other variables in equation 2 are held constant and are based on the properties of air at sea level.

$$F_{Drag} = \frac{1}{2} C_D V^2 A \rho \tag{1}$$

$$\rho = \rho_0 \left(1 - \frac{Lh}{T_0}\right)^{\frac{gM}{RL}} \tag{2}$$

3 Modelica Flight Controller

After developing the rocket model, the next step was to create the GN&C flight controller that guides the first stage down to the landing pad. A diagram of the full flight controller is pictured in Figure 3.

The flight controller model is further broken down into four different controllers. The first is the grid fin controller in Section 3.1 which is directly responsible for controlling the angle of each of the three grid fins. The other three controller are related to the engine thrust. Each of these three controllers take control of the vehicle at different phases of flight but their combined outputs are summed so if need be they could be working together simultaneously. These three controller include the boost-back controller in Section 3.2, the re-entry controller in Section 3.3, and the landing controller in Section 3.4.

The flight controller takes as input the altitude and horizontal X-axis displacement relative to the landing pad. On an actual rocket these parameters would be obtained though pressure or GPS signals.The flight control also has inputs for the angle from all three axes. Normally these would be obtained through an internal IMU internal to the flight controller. The flight controller also has an internal clock for landing sequences that require a set duration.

As outputs the flight controller has angles for each of the three grid fins. The flight controller also has three outputs for each of the three axes components in the engine thrust. The engine model then uses these thrust components to determine the total thrust force vector.

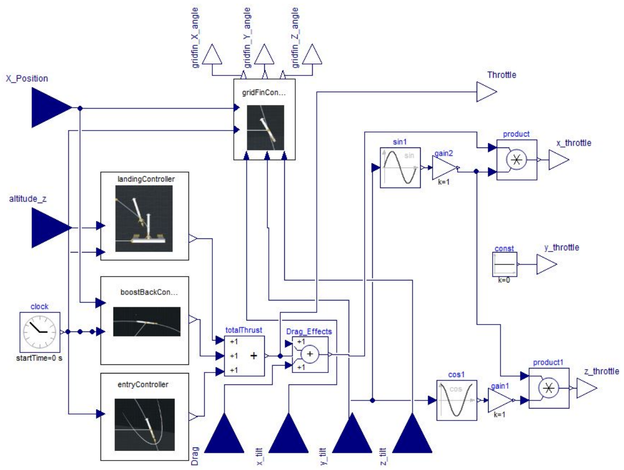

Figure 3. Flight controller model

3.1 Grid Fin Controller

The grid fin controller in Figure 4 is responsible for determining the angle of all three grid fins which impart a torque on the rocket allowing it to maneuver to the landing pad. As mentioned in Section 2.2 the rocket is assumed to operate only in the X-Z plane. Therefore, only the Y-axis grid fin is responsible for guiding the rocket to the landing pad. The other two grid fins are simply present to keep the first stage rocket from drifting out the X-Z plane. Numerical artifacts from the simulation solver have shown to result in slight drifting which then compounds until the rocket is out of control and tumbling through the atmosphere.

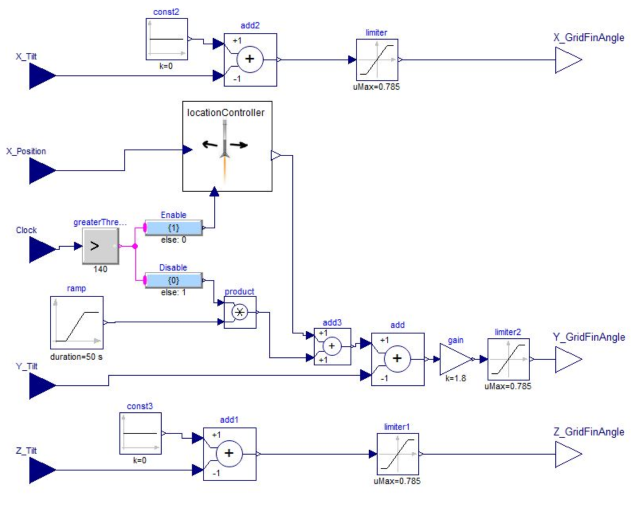

Figure 4. Grid fin controller model

The grid fin model discussed in Section 2.2 applies a maximum torque when rotated to 45 degrees relative to the first stage fuselage length. For this reason the grid fin controller applies a hard limit at $\pm\pi/4$ radians for each of the grid fin input angles. Angles in excess of 45° decrease the applied torque. As such there is no reason to operate in those regions.

The Y-axis grid fin is critical in accomplishing two tasks. The first is maneuvering the first stage rocket to an upright vertical position relative to the Earth's surface. As discussed in Section 2.1 the first stage initially only has a horizontal velocity. Once the rocket engines nearly remove this horizontal velocity the rocket needs to be rotated to a vertical position so the engines can begin slowing down the rocket in the Z-axis. This rotation is accomplished by comparing the rocket's rotational angle with that of a gradual ramp function which initiates at a predetermined altitude. A simple proportional controller relative to the changing ramp function is used to bring the rocket to the upright position. Even after the ramp function has finished bringing the rocket to the vertical position it still plays a critical roll in keeping the rocket vertical as it comes down for a landing.

The second task of the Y-axis grid fin is to guide the rocket first stage to the landing pad. This is handled by the location controller which operates within the grid fin controller. This location controller is only enabled after the rocket has transitioned to the upright position. The location controller uses proportional and derivative gains using the horizontal displacement from the landing pad as input. The gains within this PD controller were later tweaked upon evaluated the flight trajectory of the rocket.

3.2 Boost-Back Controller

The initial 263 m/s horizontal X-axis velocity gained during orbital insertion needs to be removed in order to land the first stage back at the landing pad. This is achieved by firing the rocket engines in retrograde where the thrust of the engine is normal to the direction of travel. The rocket maintains its horizontal angle during this flight phase with the help of the grid fin controller discussed in Section 3.1.

As input, the boost-back controller takes the horizontal displacement relative to the landing pad as well and an internal clock. The controller directs the engine to fire at time zero and continue until the horizontal velocity falls below a certain thresh-hold value of 50 m/s. Some horizontal velocity is desirable in order to minimize the flight path distance taken by the first stage rocket.

3.3 Reentry Controller

Space vehicles experience considerable heating during reentry into the Earth's atmosphere. The space shuttle and smaller bluff body space capsules use heat shielded tiles to protect the spacecraft and the payload inside. A first stage rocket reentering the Earth's atmosphere doesn't have this luxury of a heat shielded exterior since critical external components like the engines will be exposed regardless. To mitigate this excessive heating a rocket can instead use its own engines to slow down its velocity.

The purpose of the reentry controller is to conduct a 30 second engine burn during the critical phase of flight where aerodynamic forces are the greatest. This reentry burn decreases the rocket velocity considerably minimizing excessive heating. During this entry burn the grid fin controller discussed in Section 3.1 keeps the rocket pointed normal to the direction of motion thereby maxi-

DOI
10.3384/ecp21181171

mizing the effectiveness of the entry burn in slowing down the rocket.

3.4 Landing Controller

The final controller is responsible for landing the rocket first stage in a controlled manner that minimizes excessive accelerations. This landing controller takes the altitude and an internal clock signal as input.

Once the rocket's altitude falls below the threshold altitude of 5 km, a PD controller within the landing controller is enabled which takes over in guiding the rocket safely down to the landing pad. The gain values in this PD controller were determined through simulation. Ideally the vertical velocity component of the rocket should be zero at the moment of touch down. Simultaneously, the rocket should not undershoot the ground else it would end up hovering thereby wasting fuel. A hovering rocket is also no longer easily maneuverable since the grid fins are only effective when a velocity component is present.

4 Model Validation

The completed model and early iteration flight controller were simulated with performance evaluated based on the rocket's ability to land in a 50 m diameter landing pad. These early simulations were critical in the identification of PD controller gain constants used during different phases of flight. Landing controller PD constants were selected such that the touch down velocity was less than 2 m/s. Meanwhile, grid fin controller PD gains were selected to guide the rocket to the landing pad while minimizing overshoot and horizontal oscillations.

With all flight controllers tuned the rocket first stage was successfully able to land at the predetermined landing pad. Figure 5 shows three different flight paths of the rocket first stage with altitude on the y-axis and horizontal displacement plotted on the x-axis. Each of the four curves shown use different initial conditions for the horizontal velocity. The curves in green show successful landings where the first stage was able to touch down within the 50 m diameter predetermined landing pad. The initial condition originally selected for this model was was 293 m/s. The success of these other two trajectories show the flight controller is robust against initial conditions of at least ±10 m/s. The curve in red shows an unsuccessful landing where the rocket undershoot the pad. In this case an initial condition of 240 m/s was not enough horizontal velocity to carry the rocket close to the landing pad before entering the Earth's atmosphere where its horizontal velocity is nearly removed.

This first stage rocket model was designed to be representative of the SpaceX Falcon 9 in terms of mass properties, initial conditions, and actuator capabilities. A critical next step was to validate this simulated Falcon 9 rocket first stage landing against actual SpaceX telemetry data (Pelham 2020). Figure 6 shows the simulation trajectory in blue and the SpaceX Falcon 9 telemetry data in red. Altitude is plotted on the Y-axis with time one the x-axis.

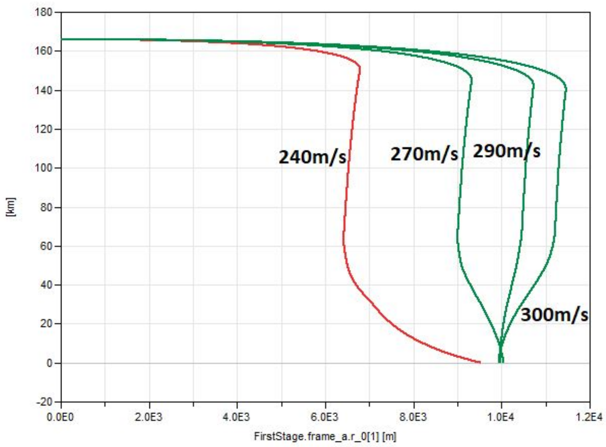

Figure 5. Flight trajectories

Comparison of the two trajectories show near identical overlap. Overlaid onto the plot are four boxes showing different phases of flight where key maneuvers occur. Each of these maneuvers is controlled by one of the controllers discussed in Section 3. One subtle difference between the simulation and SpaceX trajectories can be observed in the the landing zone box from 180 - 250 sec. The simulation data shows the first stage rocket slowing down earlier resulting in the rocket maintaining a higher altitude. The SpaceX Falcon 9 meanwhile appears to wait for a lower altitude before relighting its engines resulting in higher accelerations. This lower engine burn is likely attributed to the desire to minimize fuel on landings in order to maximize payload carrying capabilities during launch.

To better visualize the trajectory of the first stage rocket, a virtual simulation environment was created using the DLR Visualization library (Hellerer, Bellmann, and Schlegel 2014) shown in Figure 7. Using this virtual environment, a user can track the rocket as it maneuvers through the different phases of flight including visualization of grid fin rotation, engine burns, and landing at the pad at Cape Canaveral, FL.

5 Conclusions

The rocket model developed in this work and subsequent flight controller demonstrates the basic control fundamentals in recovering a rocket by landing vertically back on Earth. In this case, the rocket model was based on the SpaceX Falcon 9 first stage which has proven itself in its ability to land in a controlled manner allowing for multiple reuses. Comparing the simulated flight trajectory to data from a Falcon 9 landing shows nearly identical flight characteristics. While normally these types of simulations would occur in the early development iterations of a control system rather than recreating one *a posteriori*, there is value in having a open source Modelica model of a rocket landing. Even with the primary goal of recovery demonstrated in simulation and if real-life, there is likely opportunity for optimization of these flight paths with regards to minimizing fuel consumption and accelerations.

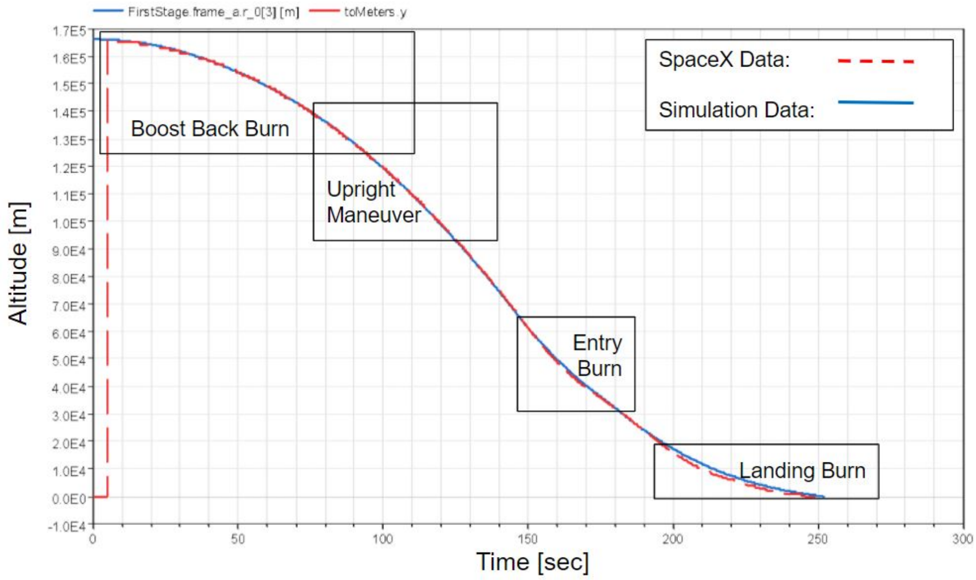

Figure 6. Comparison to Falcon 9 data

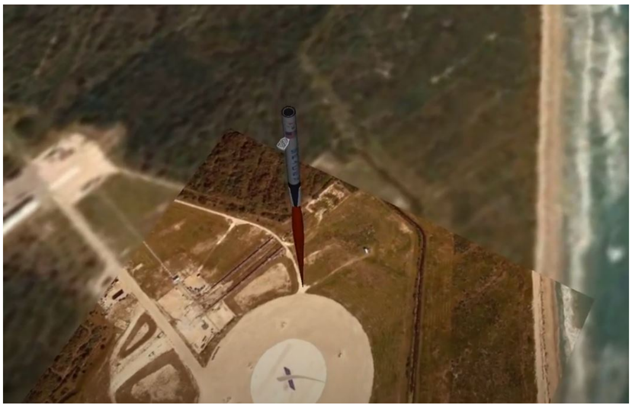

Figure 7. Visualization of rocket landing

Acknowledgements

This work was supported in part by the National Aeronautics and Space Administration through the University Leadership Initiative Award Number 80NSSC19M0125 for the Center for High-Efficiency Electrical Technologies for Aircraft (CHEETA). The second author is supported through the National Science Foundation Graduate Research Fellowship Program under Grant No. DGE 1744655 and the Chateaubriand Fellowship of the Office for Science & Technology of the Embassy of France in the United States.

References

Batteh, John et al. (2018-11). "Development and Implementation of a Flexible Model Architecture for Hybrid-Electric Aircraft". In: *Proceedings of the 1st American Modelica Conference, Cambridge, MA, USA*, pp. 37–45. DOI: 10.3384/ecp1815437. URL: https://ep.liu.se/konferensartikel.aspx?series=ecp&issue=154&Article_No=4.

Briese, Lâle Evrim, Andreas Klöckner, and Matthias Reiner (2017-05). "The DLR Environment Library for Multi-Disciplinary Aerospace Applications". In: *Proceedings of the 12th International Modelica Conference, Prague, Czech Republic*, pp. 929–938. DOI: 10.3384/ecp17132929. URL: https://ep.liu.se/ecp/article.asp?issue=132&article=102&volume=0#.

Hellerer, Matthias, Tobias Bellmann, and Florian Schlegel (2014-03). "The DLR Visualization Library — Recent Developments and Applications". In: *Proceedings of the 10th International Modelica Conference, Lund, Sweden*, pp. 899–911. DOI: 10.3384/ECP14096899. URL: https://ep.liu.se/en/conference-article.aspx?series=ecp&issue=96&Article_No=94.

Milz, Daniel et al. (2019-03). "Advances in Flight Dynamics Modeling and Flight Control Design by Using the DLR Flight Visualization and Flight Instruments Libraries". In: *Proceedings of the 13th International Modelica Conference, Regensburg, Germany*, pp. 481–488. DOI: 10.3384/ecp19157481. URL: https://ep.liu.se/ecp/article.asp?issue=157&article=049&volume=0.

Pelham, Jonathan (2020-05). "SpaceXtract". In: *Extraction and analysis of telemetry from rocket launch webcasts*. URL: https://github.com/shahar603/SpaceXtract.

Posielek, Tobias (2018-11). "A Modelica Library for Spacecraft Thermal Analysis". In: *Proceedings of the 1st American Modelica Conference, Cambridge, MA, USA*, pp. 46–55. DOI: 10.3384/ecp1815446. URL: https://ep.liu.se/konferensartikel.aspx?series=ecp&issue=154&Article_No=5.

Re, Fabrizio (2011). "Modelica Landing Gear Modelling and On-Ground Trajectory Tracking with Sliding Mode Control". In: *Advances in Aerospace Guidance, Navigation and Control*. Ed. by Florian Holzapfel and Stephan Theil. Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 103–115. ISBN: 978-3-642-19817-5.

Wei, Liu et al. (2015-09). "Modeling and Simulation of Liquid Propellant Rocket Engine Transient Performance Using Modelica ". In: *Proceedings of the 11th International Modelica Conference, Versailles, France*, pp. 485–490. DOI: 10.3384/ecp15118485. URL: https://ep.liu.se/en/conference-article.aspx?series=118&issue=118&Article_No=52.