POLITECNICO
MILANO 1863

SCUOLA DI INGEGNERIA INDUSTRIALE
E DELL'INFORMAZIONE

Executive Summary of the Thesis

An object-oriented approach to the modeling of a student-developed sounding rocket and its control systems

Laurea Magistrale in Automation and Control Engineering - Ingegneria dell'automazione

Author: Luca Erbetta

Advisor: Prof. Francesco Casella

Academic year: 2020-2021

1. Introduction

In recent years, sounding rockets developed by university student teams are becoming more complex. Despite this, the development lifecycle of these rockets is much shorter than in the industry, taking one to two years from conception to launch, challenging students to perform rapid iterations in the design. Tools based on causal modeling approaches are frequently employed to evaluate the performance of the various subsystems while keeping costs down, but these models are usually difficult to implement, extend and reuse. For this reason, an approach based on acausal and object-oriented modeling using the Modelica language is evaluated in this thesis, with the objective of simplifying the model design process and increasing its reusability and extendibility [1]. This model is then employed in the development of guidance, navigation and control algorithms with the objective of making the trajectory of the rocket less susceptible to wind in the first instants of the flight.

2. Object oriented approach

The development of a complete model to simulate the flight of a rocket is a multi-disciplinary task, which includes systems spanning multiple physical domains. The object-oriented modeling approach is well suited for this task: the topology of the model is close to the one of the real system, where different subsystems are connected together using *connectors* representing their physical interaction, while hiding the implementation details through the concept of *encapsulation*. The systems are described by *equations* and not *assignments*, freeing the user from the burden of performing manual symbolic manipulations to obtain an Ordinary Differential Equations (ODE) system. Finally, the model has a modular and hierarchical structure, taking advantage of object-oriented features such as *abstraction* and *inheritance*. All these features result in a model that is simpler to develop, maintain and extend in the future with respect to traditional causal approaches.

3. Model structure

The rocket model developed in this thesis considers a modification of the "Lynx" rocket developed by Skyward Experimental Rocketry: a solid-propellant rocket capable of reaching up to 3000 m of apogee to which 4 canards are added to control its attitude and trajectory (Fig. 1).

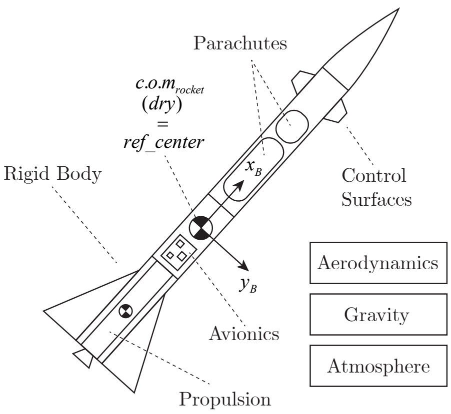

Figure 1: Subsystems and domains of a sounding rocket

Fuselage The fuselage is modeled as a rigid body using models provided by the Modelica Multibody Library.

Propulsion An abstract model for a generic propulsion system is extended to implement a solid propellant motor based on commercially available ones, where the mass flow rate is calculated from the thrust curve in order to describe the evolution of the mass and inertia of the propellant during the burn.

Aerodynamics Two models for the aerodynamics of the rocket during the ballistic phase of the flight are provided.
A more accurate model interpolates multi-dimensional aerodynamic coefficient tables obtained from external software and CFD simulations, indexing the tables based on the current aerodynamic state of the rocket (angle of attack and sideslip, Mach number, altitude and canard deflection angles). Due to the high number of dimensions of the tables, this method is however computationally expensive.
The second method is faster but less accurate, linearizing the aerodynamic coefficients around the equilibrium $\alpha = 0, \beta = 0, \boldsymbol{\delta} = \mathbf{0}$ while neglecting the dependence on the altitude and Mach number of the coefficients (with the exception of the axial force coefficient, which is still interpolated from a vector based on the Mach number).

Parachutes The parachutes are approximated by applying the drag force directly on their attachment point on the fuselage, and the parachute surface during the opening transient is assumed to increase linearly within a preset time interval. The elasticity of the lines and mass properties of the parachutes are neglected, mainly to improve the performance of the simulation.

Actuators The canard actuators are modeled as a first order system.

Environment Due to the short duration of the flight of the rocket and relatively small altitudes reached, the roundness and rotation of the Earth are neglected, and the world is assumed flat.
The atmosphere is modeled up to an altitude of 11 km using the International Standard Atmosphere (ISA) model, and the wind force is composed of two contributions: a static wind speed that can change in magnitude and direction between a number of discrete layers, and a colored noise representing the wind turbulence, shaped through appropriate filters described by the Dryden Wind Turbulence Model.
Finally, the Earth's geomagnetic field is modeled using the dipole approximation of the International Geomagnetic Reference Field (IGRF) model.

Sensors Models for various sensors are implemented: accelerometers, gyroscopes, magnetometers, barometers and Global Navigation Satellite System (GNSS) modules. At first, "true" sensors without errors are modeled, finally extending them adding errors such as biases, discretization, quantization and noises. The measurement model for the gyroscopes and accelerometers takes into account the presence of both constant bias and bias instability (modeled as a first order random walk) as well as thermo-mechanical noise (Eq. 1):

$$\begin{aligned} \boldsymbol{f}(t) =& \boldsymbol{f}^{true}(t) + \boldsymbol{\beta}_f \\ &+ \boldsymbol{\eta}_{\boldsymbol{f}_{\boldsymbol{VRW}}}(t) + \int_0^t \boldsymbol{\eta}_{\boldsymbol{f}_{\boldsymbol{BI}}}(t) dt \end{aligned} \tag{1}$$

The magnetometer model accounts for errors in the calibration of the Hard Iron and Soft Iron effects introducing a misalignment term, and the GNSS modules present a sinusoidal bias in both the position and velocity measurements.

Blocks for low cost sensors commonly used in student-developed sounding rockets are finally provided, ready to use in the model.

Avionics Bus The outputs of the sensors are made available to the rest of the model (and in particular the Guidance, Navigation and Control systems) through an expandable connector, where signals can be added on the go without needing to declare them beforehand.

4. Navigation

The navigation system uses the output of the sensors corrupted with errors to estimate the state of the rocket: the position, velocity and attitude, as well as the biases of the gyroscope. An integrated inertial navigation approach is implemented: measurements from gyroscopes are integrated to obtain the attitude of the rocket, while accelerometers are used to obtain the inertial velocity and position [2]. Due to the use of low cost MEMS sensors this approach is susceptible to drift already after a few seconds of operation, that can however be corrected using measurements from the magnetometer and GNSS module. The navigation solution is produced using a Multiplicative Extended Kalman Filter (MEKF) for the estimation of the attitude, that is then fed into a linear Kalman Filter for the estimation of the velocity and position. The MEKF was chosen over the Extended Kalman Filter in order to maintain the unitary norm of the attitude quaternion estimate [3]: the MEKF estimates in fact the attitude error, which is then propagated into the global attitude quaternion through a *reset* operation that maintains its unitary norm. The filter is updated using a single vector measurement from the magnetometer, leaving the drift uncorrected around its direction: this was shown to be acceptable for the short flights considered, as the direction of this vector changes with respect to the body frame due to the rotation of the rocket (Fig. 3). The navigation algorithm was also tested using flight data from the first flight of Lynx as an input, and its output was compared, through a 3D animation based on the estimates of the filter, with the recording from the on-board camera (Fig. 2). The full comparison video can be found at `https://youtu.be/15U2QjJIRWA`.

Figure 2: Still from virtual flight recreation video

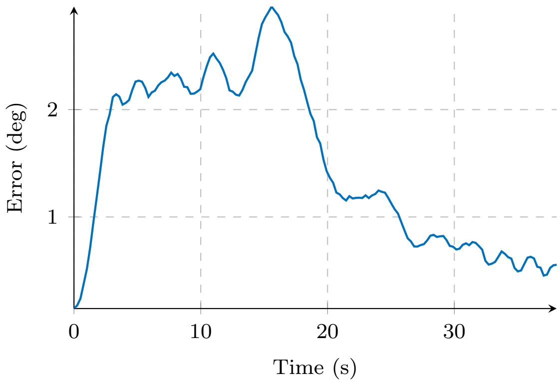

Figure 3: Attitude estimation error during a flight

5. Guidance

The guidance algorithm outputs body angular rates commands to be used by the control system with the objective of tracking a predefined analytical parabolic trajectory, which approximates the trajectory that the rocket would take in the absence of wind. At each instant, the desired direction of the inertial velocity vector of the rocket ($\boldsymbol{d_{ref}}$) is obtained from the analytical reference trajectory, and the angle error between the current and desired velocity direction is computed (Eq. 2). The angular velocity command is then computed to be proportional to this error, perpendicular to the two vectors in order to rotate the current velocity vector into the target direction (Eq. 3).

$$\chi = \text{acos}\left(\frac{\boldsymbol{V} \cdot \boldsymbol{d}_{ref}}{\|\boldsymbol{V}\|}\right) \tag{2}$$

$$\bar{\boldsymbol{\omega}}_{ref} = \frac{\boldsymbol{V} \times \boldsymbol{d}_{ref}}{\|\boldsymbol{V} \times \boldsymbol{d}_{ref}\|} k\chi \tag{3}$$

The roll angle of the rocket is then commanded in a similar fashion with the objective of making the rocket point into a specific direction.

6. Control

The angular rate commands from the guidance algorithms are finally tracked by the control algorithm using a Linear Quadratic Integral (LQI) controller. The nonlinear equations of motion of the rocket, expressed in the body frame (Eq. 4), are linearized in the equilibrium condition with zero angle of attack and sideslip and null angular rates [4].

$$\begin{aligned} \dot{\boldsymbol{V}} &= -\boldsymbol{\omega} \times \boldsymbol{V} + \frac{\boldsymbol{F}}{m} \\ J\dot{\boldsymbol{\omega}} &= (J\boldsymbol{\omega}) \times \boldsymbol{\omega} + \boldsymbol{M} \end{aligned} \quad (4)$$

The aerodynamic forces are modeled as in the simplified model described in Sec. 3, with the additional simplification that the axial force coefficient doesn't depend on the Mach number. The dynamics of the canard actuators are also included in the system, as well as the integral of the errors of the angular rates with respect to the guidance output. Due to the great range of velocities that the rocket reaches during the flight, the equations of motion cannot be linearized around a single condition, and the resulting linear system depends on the velocity of the rocket, making it time varying: a Discrete-time Algebraic Riccati (DARE) equation must thus be solved at each step in order to synthesize the gain-scheduling controller. To improve performance, this operation could be performed offline over a range of velocities, storing the controller gains into a table that can be then interpolated online.

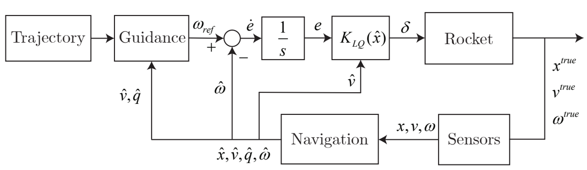

Figure 4: Control loop diagram

7. Results

The model has been validated comparing it with flight data obtained from two launches of the Lynx rocket in 2021, providing results that closely match reality (Fig. 5).

The two aerodynamic models perform similarly on the longitudinal dynamics of the rocket, but present some differences in the canard actuator

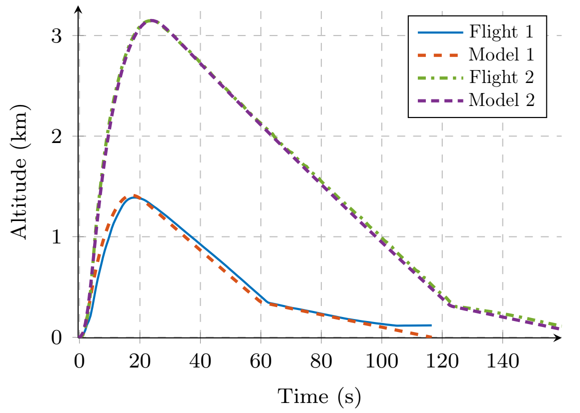

Figure 5: Flight data vs simulation altitude plot

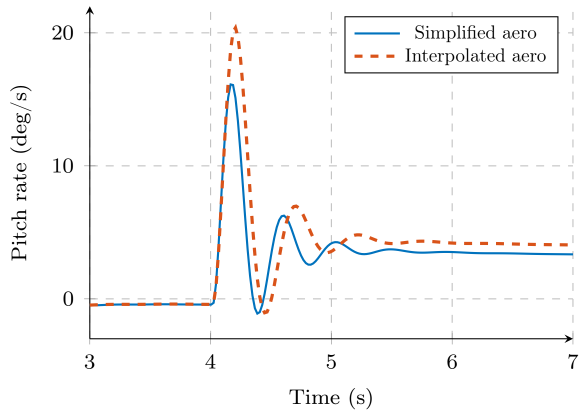

Figure 6: Canard actuation step response ($t = 4\ s$, $\delta_p = 3\ deg$)

response (Fig. 6). The simplified aerodynamic model is however 30 times faster than the interpolated model and it is suitable when fast iterations are needed or when multiple simulations need to be performed. Finally, the GNC system is proven effective in maintaining the intended trajectory in presence of wind: Fig. 7 shows the flight path angle of the rocket subject to a 10 m/s wind, in the cases where the controller is activated 1 or 3.5 seconds after lift-off.

7.1. Monte Carlo Analysis

A Monte Carlo analysis has been performed, simulating the model with a large number of wind conditions in the cases of a non-nominal ballistic flight and with a nominal descent under parachutes. In the case of a ballistic flight, the impact position ellipse area is reduced by about 99% when the controller is activated both 1 and 3.5 s into the flight, with the ellipse being slightly larger in the latter case due to more aggressive maneuvers needed to steer the rocket

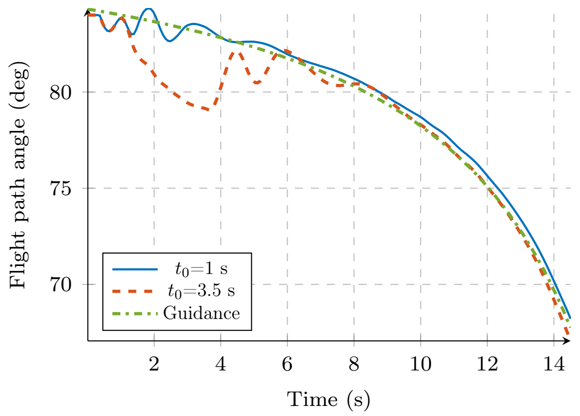

Figure 7: Closed loop flight path angle

into the desired trajectory after it was already affected by wind. In the case of a nominal flight, however, the landing point dispersion is larger when the controller is active: the reason being that in an uncontrolled flight the rocket flies *into* the wind when ascending due to it being aerodynamically stable, but travels in the direction of the wind while descending under parachutes, partially negating the two effects and making the landing location uncertainty smaller.

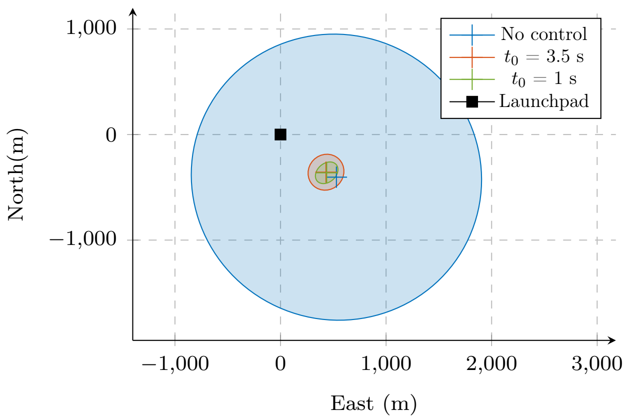

Figure 8: Ballistic flight impact location, 95% confidence ellipses (N = 360)

8. Conclusions

The use of an object-oriented approach to the simulation of the flight of a sounding rocket has proven to have clear advantages in terms of modularity, as different models can be implemented and modified with ease, and in the integration of different domains in the same model, allowing a complete description of the dynamics of the rocket and its subsystems. Another advantage, that however can also be a drawback, is

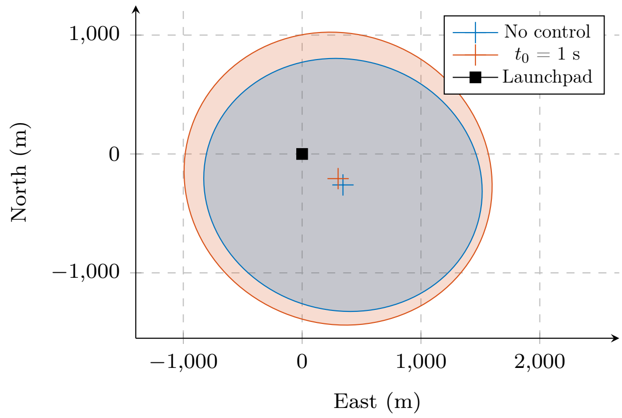

Figure 9: Nominal parachute descent landing location, 95% confidence ellipses (N = 360)

that the symbolic manipulation of the system of equations is performed by the compiler, reducing the workload on the user but also making it difficult to debug problems that may arise in the compilation process.
The implemented model has been shown to be able to represent reality by comparing the simulation with flight data. The implemented GNC algorithm is capable of making the trajectory less susceptible to wind disturbances and to reduce the uncertainty in the impact location in case of a ballistic flight, but does not achieve the same results in the case of a nominal descent under the parachutes, where additional control systems like guided parachutes may be needed to control the rocket also in the descent phase.

References

[1] Gertjan Looye. *An Integrated Approach to Aircraft Modelling and Flight Control Law Design.* PhD thesis, TU Delft, 2008.

[2] Oliver J. Woodman. An Introduction to Inertial Navigation. Technical report, University of Cambridge, 2007.

[3] John L. Crassidis, F. Landis Markley, and Yang Cheng. Survey of nonlinear attitude estimation methods. *Journal of Guidance, Control, and Dynamics*, 2007.

[4] Rafael Yanushevsky. *Modern Missile Guidance.* CRC Press, 2008.