24th COBEM - 2017

24th ABCM International Congress of Mechanical Engineering
December 3-8, 2017, Curitiba, PR, Brazil

COBEM-2017-0915
MODELING, SIMULATION AND CONTROL OF A LAUCH VEHICLE

Rodrigo Zanatta
Universidade Federal de Itajubá - UNIFEI
rodrigozanatta@live.com

Marcelo Santiago de Sousa
Universidade Federal de Itajubá - UNIFEI
marcelosousa.unifei@gmail.com

Sebastião Simões da Cunha Júnior
Universidade Federal de Itajubá - UNIFEI
sebas@unifei.edu.br

Abstract. *A 6DoF model of a thrust vectoring rocket launch dynamics was developed. The primary objective was to access the performance of different control laws in stabilizing and controlling the rocket. The model was simplified in many aspects, especially in the use of a flat earth reference frame. PID and Universal Integral Regulator (RIU) control laws were implemented, and manoeuvres consistent with a gravitational turn were simulated. The results do not allow to draw a conclusion on the superiority of any control law. It is expected that further developments may contribute to this study.*

Keywords: *rocket modelling, rocket dynamics, rocket control*

1. INTRODUCTION

This is a partial report on the development of a rocket launch model and it's implementation, which is expected to serve varied purposes from rocket sizing, trajectory studies, and assessment of control performance. The main objective in this first phase is to implement the basic model and access the performance of PID and UIR (Universal Integral Regulator, (Seshagiri and Khalil, 2005)) controllers.

The workflow consists in the following: 1) the development of the model and its implementation in a computer environment, 2) planning of the manoeuvres, 3) implementation and tuning of control laws, 4) Execution of simulations and evaluation of the results.

Although many good models have been developed and are sometimes readily available from the literature (from classical textbooks such as Zipfel (2007) and Tewari (2007) to specialized cases such as Baldesi *et al.* (2006) and Betts *et al.* (2007)), the one presented here was developed from scratch, as much as possible based on fundamental physical principles. The model consists essentially in a set of second order non-linear differential equations representing some of the most important aspects of the system. It is a six degree of freedom model where some simplifications and limits are as follows: flat earth reference system, no sloshing, structural deformation of thermal effects; the simulation is limited to the first stage of a thrust vectoring control axiallly simmetric rocket, where aerodynamic fluctuations, nozzle vibrations and thrust instabilities are represented. The model has been implemented in Simulink®.

Control laws were added and controlled and uncontrolled simulations were realized. Uncontrolled simulations are performed to acess rocket basic and natural behavior, such as variations of mass and propper relation between dynamic and kinematic states. A maneuvre consistent with gravitational turn trajectory was developed and simulated, and the performance of PID and UIR laws were compared. No substantial differences were found, since both control laws were able to perform in a very similiar and satisfatory way.

Although unconclusive results, the developed plataform was satisfatory to the objectives and promising enough to justify ongoing work in the model development and addition of other control laws and features in future works.

2. MATHEMATICAL MODEL

2.1 Reference Frames, Angles, Forces and Moments

Figure 1 shows the relationship between the two main adopted reference frames: frame $OXYZ$ is the earth reference frame, fixed to the launch point, and $oxyz$ is the body reference frame, fixed to the rocket's body, with origin at its the center of gravity (CG). Between these, one can see a rotation defined by the 3 angles Ψ, Θ and Φ. In the figure, earth frame has been translated to the rocket's CG for clarity.

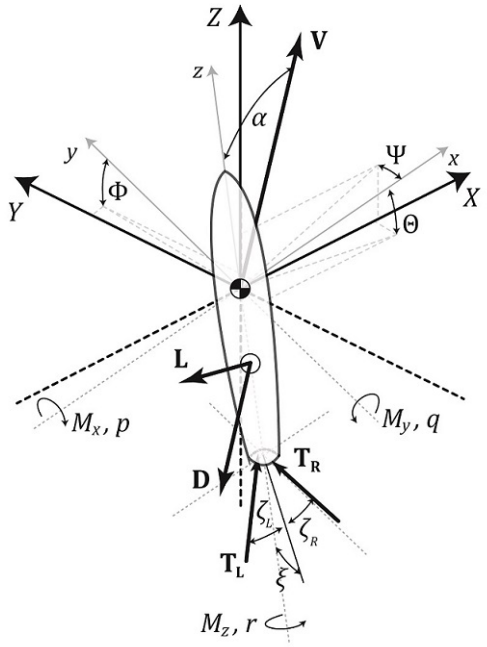

Figure 1. Free body diagram of the generic rocket

Except for Φ, angles are represented in their negative direction. The figure also shows the velocity vector ($\mathbf{V}$) in earth frame, applied to the origin of the body frame, forming with axis oz angle α. Aerodynamic forces lift ($\mathbf{L}$) and drag ($\mathbf{D}$) are also represented, both in a plane defined by vectors $\mathbf{V}$ and the unity "body vector" $\mathbf{v}_c^{'} = \begin{bmatrix}0 & 0 & 1\end{bmatrix}^T$, in body frame, and applied at the rocket's center of pressure (CoP), distant d_{CoP} from the CG. Two thrust vectors, $\mathbf{T}_L$ and $\mathbf{T}_R$, representing two nozzles are shown, distant $d_T/2$ from the axis defined by the intersection of plane xz and the rocket's base plane.

These form with the body plane yz angles ζ_L and ζ_R, and with plane xz the single angle ξ. These angles represent thrust vectoring, which accounts for the rocket's maneuver control. The two thrust vectors are applied at the rocket's base, distant z_{CG} from the CG. These forces produce moments M_x, M_y and M_z about the CG, and around the body axis, resulting in angular accelerations and velocities, the later represented by p, q and r. In the figure, except for M_y and q, moments and angular velocities are represented in their positive direction.

Once it is paramount to constantly move from one reference frame to the other, it is convenient to define a coordinate transformation function and it's inverse. If $\mathbf{v}$ is a tridimensional vector in earth frame and $\mathbf{v}^{'}$ is the same vector in body frame, then

$$\varphi(\Phi,\Theta,\Psi) = \mathbf{v}^{'} = \mathbf{R}\mathbf{v} \tag{1a}$$

$$\varphi^{-1}(\Phi,\Theta,\Psi) = \mathbf{v} = \mathbf{R}^T\mathbf{v}^{'} \tag{1b}$$

where

$$\mathbf{R} = \begin{bmatrix} \cos\Psi\cos\Theta & \cos\Theta\sin\Psi & -\sin\Theta \\ \cos\Psi\sin\Phi\sin\Theta & \cos\Phi\cos\Psi + \sin\Phi\sin\Psi\sin\Theta & \cos\Theta\sin\Phi \\ \sin\Phi\sin\Psi + \cos\Phi\cos\Psi\sin\Theta & \cos\Phi\sin\Psi\sin\Theta - \cos\Psi\sin\Phi & \cos\Phi\cos\Theta \end{bmatrix} \tag{2}$$

is the coordinate transfomation matrix, as formalized by Diebel (2006).

2.2 Accelerations

From basic dynamics, the rocket's linear acceleration in earth coordinate frame, can be written as:

$$\mathbf{a} = \frac{d\mathbf{V}}{dt} = (m\mathbf{I})^{-1}\,\mathbf{F} = \frac{1}{m}\begin{bmatrix} T_X + D_X + L_X \\ T_Y + D_Y + L_Y \\ T_Z + D_Z + L_Z + mg \end{bmatrix} \tag{3}$$

In Eq. 3, m is the instantaneous mass, $\mathbf{I}$ is the identity matrix, and $\mathbf{F} = \mathbf{T} + \mathbf{D} + \mathbf{L} + m\mathbf{g}$ is the sum of forces on the rocket, with components in the OX, OY and OZ directions.

Since the rocket's axial symmetry allows one to consider its longitudinal axis as a principal axis, and since by the same symmetry $J_s = J_x x = J_y y$ is the moment of inertia around body's ox and oy axis, then the inertia tensor is given by:

$$\mathbf{J} = \begin{bmatrix} J_s & 0 & 0 \\ 0 & J_s & 0 \\ 0 & 0 & J_{zz} \end{bmatrix} \tag{4}$$

The total moment vector, which is the sum of aerodynamic and thrust moments, is given by:

$$\mathbf{M} = \mathbf{J}\frac{d\omega}{dt} + \omega \times (\mathbf{J}\omega) \tag{5}$$

Where ω is the angular velocity, then angular accelerations, after some rearrangement are:

$$\dot{\omega} = \frac{d\omega}{dt} = \begin{bmatrix} \frac{1}{J_s} & 0 & 0 \\ 0 & \frac{1}{J_s} & 0 \\ 0 & 0 & \frac{1}{J_z} \end{bmatrix} \begin{bmatrix} M_x - \omega_y\omega_z\,(J_{zz} - J_s) \\ M_y - \omega_x\omega_z\,(J_s - J_{zz}) \\ M_z \end{bmatrix} \tag{6}$$

The task now is to find expressions for the terms in Eqs. 3 and 6.

2.3 Aerodynamic Forces and Moments

One can proceed by first determining aerodynamic forces directions, then their magnitudes, and then the position of the CoP, where they are applied, from which the aerodynamic moment ($\mathbf{M}_{aero}$) can be computed.

By definition, drag is parallel and opposite to velocity, and so, if D is drag's magnitude, then

$$\mathbf{D} = -D\frac{\mathbf{V}}{\|\mathbf{V}\|} \tag{7}$$

Lift, by definition, is perpendicular to velocity. By inspection of Fig. 1, one can see that lift is in the same plane defined as the plane that contains $\mathbf{V}$ and $\mathbf{v}_c = \mathbf{R}^T\mathbf{v}_c^{'}$. From the right hand rule for the cross product of two vectors, $\mathbf{V} \times \mathbf{v}_c$ gives a vector perpendicular to both, that points in the direction of one's thumb when her finger's close from $\mathbf{V}$ to $\mathbf{v}_c$. If now one makes the cross product of this resulting vector again with $\mathbf{V}$, then the direction of lift in earth frame is found. So, if L is the magnitude of lift, then

$$\mathbf{L} = L\frac{(\mathbf{V} \times \mathbf{v}_c) \times \mathbf{V}}{\|(\mathbf{V} \times \mathbf{v}_c) \times \mathbf{V}\|} \tag{8}$$

And so, being d_{CoP} the distance from the CG to the CoP, the aerodynamic moment is

$$\mathbf{M}_{aero} = d_{_{CoP}} \begin{bmatrix} L_{y}^{'} + D_{y}^{'} \\ -L_{x}^{'} - D_{x}^{'} \\ 0 \end{bmatrix} \tag{9}$$

The primes in the components of the force vectors in the Eq. 9 represent the fact that they have been converted to body coordinates. For the aerodynamic forces magnitudes, one can start with the well-known equations (Anderson, 2007):

$$L = \frac{1}{2}\rho V^2 S_{ref} C_L \tag{10a}$$

$$D = \frac{1}{2}\rho V^2 S_{ref} C_D \tag{10b}$$

In these, S_{ref} and ρ are, respectively, a reference area and density. Now the aerodynamic coefficients must be defined, and there is no simple way for doing it. Most of the times, as in the works mentioned before, aerodynamic data is input into the model from data tables obtained experimentally. In the present modeling, it was decided to use interpolating functions over the range of a limited set of data, since this kind of experimental data was not available. The limited set of data used was obtained through CFD, and were fed into functions of the form:

$$C_L^{'} = C_{L_\alpha}(Ma)\,\alpha \tag{11a}$$

$$C_D^{'} = C_{D_{\alpha=0}}(Ma)\,\alpha + k_1\alpha^{k_2} \tag{11b}$$

$$d_{CoP}^{'} = a(Ma)\,\alpha^{b(Ma)} \tag{11c}$$

In Eqs. 11, $C_{L_\alpha}(Ma)$ is an interpolating function of a set of data that gives the slope of the C_L curve as a function of Mach, $C_{D_{\alpha=0}}(Ma)$ is an interpolation function over a set of data for the drag coefficient at zero angle of attack α, and $a(Ma)$ and $b(Ma)$ are interpolation functions over sets of data for the CoP position modeled as a power function. Coefficients k_1 and k_2 should be adjusted to fit drag coefficient as α varies.

The primes in this case indicate that they are nominal values. As already mentioned, it is intended to introduce fluctuations in the aerodynamic coefficients, representing all sorts of buffeting and other interferences, especially those related to shock waves. So these fluctuations must be proportional to the coefficients themselves, but also maximum when $Ma \longrightarrow 1$.

For this, a function that returns a random number with Gaussian distribution around 0 and maximum standard deviation when $Ma = 1$ was defined. The returned value of the function is then multiplied by a fraction of the coefficient, defining the maximum fluctuation value in general.

As for the standard deviation value, a Gaussian function of the form

$$\sigma_a = e^{-\left(\frac{Ma-1}{\delta}\right)^2} \tag{12}$$

will make it maximum ($\sigma_a = 1$) when $Ma = 1$. The term δ controls the spread of the bell shaped curve. A larger value implies a fatter curve. The value of this standard deviation shall be then fed into a random number generator function with Gaussian distribution around 0. Obviously, there is no such a thing as a random number function in mathematics, but computer environments such as Matlab/Simulink® can provide such functions.

Now, an implicit function that takes as argument the standard deviation and returns a random number as discussed above is defined. Let C_{fr} be a generic coefficient ratio and $f_g(\sigma_a)$ be the function. Then the fluctuations are simply $C_{fr}f_g(\sigma_a)$. Care was taken to make that 95% of the random numbers generated will be the in the $\pm 2\sigma$ range around nominal values of the coeffcients, and the deviation (σ itself) will be the most when $Ma = 1$. So, in this case:

$$C_L = C_L^{'} + C_{L_{fr}}f_g(\sigma_a) \tag{13a}$$

$$C_D = C_D^{'} + C_{L_{fr}}f_g(\sigma_a) \tag{13b}$$

$$d_{CoP} = d_{CoP}^{'} + C_{CoP_{fr}}f_g(\sigma_a) \tag{13c}$$

Equations 13 give the actual values of the aerodynamic coefficients. In a computer implementation, it may be useful to conditionally use these fluctuations, so that they may be shut off and on, depending on the needs. The last term needed is α, which is not exactly the angle of attack, since there is no need to define the angle of side-slip, given the simmetry of the rocket:

$$\alpha = \arccos\frac{\mathbf{V}\cdot\mathbf{v_c}}{\|\mathbf{V}\|\,\|\mathbf{v_c}\|} \tag{14}$$

2.4 Thrust Forces

From inspection of Fig. 1, thrust vector in body frame is given by:

$$\mathbf{T}^{'} = \begin{bmatrix} T_L\sin\zeta_L + T_R\sin\zeta_R \\ (T_L + T_R)\sin\xi \\ (T_L\cos\zeta_L + T_R\cos\zeta_R)\cos\xi \end{bmatrix} \tag{15}$$

To have moments about the oz axis, a differential thrust must also be defined as:

$$T_{diff} = T_R\sin\zeta_R - T_L\sin\zeta_L \tag{16}$$

And so, the moment vector of the thrust forces, in body frame, is:

$$\mathbf{M}_T = \begin{bmatrix} z_{CG} T_y^{'} \\ -z_{CG} T_x^{'} \\ \frac{d_T}{2} T_{diff} \end{bmatrix} \tag{17}$$

Usually, the thrust of rocket engines is given as values for sea level (SL) and vacuum (vac). If so, a simple linear function for thrust magnitude can be written as:

$$T = T_{vac} + \frac{T_{SL} - T_{vac}}{\rho_{SL}} \rho \tag{18}$$

As with aerodynamics, fluctuations in thrust magnitude can be introduced, but in this case, they may be simpler, once they have no dependency on Mach number. Defining nominal values for each nozzle's thrust as:

$$T_L^{'} = T_R^{'} = \frac{1}{2} T \tag{19}$$

Now, being σ_T the standard deviation for thrust, and defining a fraction as T_{fr}, then the actual values are:

$$T_L = T_L^{'} + T_{fr} f_g\left(\sigma_T\right) \tag{20a}$$

$$T_R = T_T^{'} + T_{fr} f_g\left(\sigma_T\right) \tag{20b}$$

2.5 Mass and Inertia

Considering a liquid fuel rocket with fuel and oxidizer tanks, and considering as structural mass (m_s) all mass that is not propellant, then if $m_p = m_f + m_o$ is the mass of propellant (fuel plus oxidizer), then the instantaneous mass of the rocket is:

$$m = \begin{cases} m_s + m_{p_{ini}} - \dot{m}_p t & \text{if } m_p > 0 \\ m_s & \text{if } m_p \leq 0 \end{cases} \tag{21}$$

Subscripts ini and fin account for initial and final. If α_f is oxidizer-fuel ratio, then:

$$m_p = m_f + m_o = \alpha_f m_f + m_f = m_f\left(1 + \alpha_f\right) \tag{22}$$

From where the mass of fuel and oxidizer can be computed. Considering now the mass of fuel and oxidizer in the tanks, indicating their initial and final CG's positions as $z_{CG_{f,ini}}$, $z_{CG_{f,fin}}$, $z_{CG_{o,ini}}$ and $z_{CG_{o,fin}}$, the instantaneous CG position of the mass of fuel and oxidizer are given by:

$$z_{CG_f} = \left(1 - \frac{m_f}{m_{f_{ini}}}\right) z_{CG_{f,fin}} + \frac{m_f}{m_{f_{ini}}} z_{CG_{f,ini}} \tag{23a}$$

$$z_{CG_o} = \left(1 - \frac{m_o}{m_{o_{ini}}}\right) z_{CG_{o,fin}} + \frac{m_o}{m_{o_{ini}}} z_{CG_{o,ini}} \tag{23b}$$

And the overall instantaneous CG position is given by:

$$z_{CG} = \frac{z_{CG_s} m_s + z_{CG_f} m_f + z_{CG_o} m_o}{m_s + m_f + m_o} \tag{24}$$

Where z_{CG_s} is the CG's position of the unvarying mass. For the moments of inertia, it is acceptable to consider the rocket and the fuel and oxidizer masses as solid cylinders with the same diameter D_J. Being so, then the moment of inertia about the oz axis is given by:

$$J_{zz} = \frac{1}{8}\left(m_s + m_f + m_o\right) {D_{ref}}^2 \tag{25}$$

For the moment of inertia J_s about ox and oy axis, the lengths of the masses and the distances from their CG's and the overall CG must me computed. The length of the rocket mass is the length of the rocket, a constant geometric data. The lengths of the fuel and oxidizer masses depend on how much propellant remains in the tanks, so:

$$L_f = 2\left(z_{CG_f} - z_{CG_{f,fin}}\right) \tag{26a}$$

$$L_o = 2\left(z_{CG_o} - z_{CG_{o,fin}}\right) \tag{26b}$$

The distances are:

$$d_s = z_{CG} - \frac{L_s}{2} \tag{27a}$$

$$d_f = z_{CG} - z_{CG_f} \tag{27b}$$

$$d_o = z_{CG} - z_{CG_o} \tag{27c}$$

And so:

$$J_s = \frac{1}{12}\left(m_s {L_{ref}}^2 + m_f {L_f}^2 + m_o {L_o}^2\right) + m_s {d_{ref}}^2 + m_f {d_f}^2 + m_o {d_o}^2 \tag{28}$$

2.6 Environment

Environmental models are well known and one is free to use the one that suits better. Nonetheless, care must be taken that the model used gives good values for density ρ, gravitational acceleration ($\mathbf{g}$), and speed of sound (V_s) up to the altitude where the rocket is supposed to go.

In the present case, with respect to gravitational acceleration, a good approximation is given by:

$$g = g_{SL}\left(\frac{R_e}{R_e + h}\right)^2 \tag{29}$$

where R_e and h are the mean equatorial radius of earth and altitude, respectively.

With respect to the other parameters, the COESA 1976 atmospheric model was used (NOAA *et al.*, 1976). Once the computations are to be implemented in a computer environment, the pre-defined Matlab/Simulink® function may suffice. Note should be taken, though, that this model is good enough only up to about 85 km, over which density was set to zero and speed of sound was set to constant and equal to its value at 85 km.

2.7 Thrust Vectoring

From Fig. 1, rocket vectoring is defined by angles ζ_L, ζ_R and ξ. A very simple actuation model was implemented, where a desired (subscripted as dsr) angular position of the nozzles results from the controller's commands (subscripted as cmd). As for angular position ξ, $\xi_{dsr} = \xi_{cmd}$, but this is not the case for ζ angles, where a differential (subscripted as $diff$) command is also present. In this case, one has:

$$\zeta_{L_{dsr}} = \zeta_{cmd} + \frac{1}{2}\zeta_{diff} \tag{30a}$$

$$\zeta_{R_{dsr}} = \zeta_{cmd} - \frac{1}{2}\zeta_{diff} \tag{30b}$$

In each case, the position is the result of the angular movement of the nozzle from its actual (subscripted as act) angular position to the desired one, at a constant angular speed λ. Once there will be an error e_n from the actual to the desired position, a very simple controller should move the nozzle in accordance with:

$$\xi^{'} = \xi_{act} + \int \lambda \text{sign}\left(e_n\right) dt \tag{31}$$

Analogous equations are immediately written for ζ_L and ζ_R. Care must be taken to not allow movement beyond the possible range of the nozzles, using some sort of function saturation.

Also, to represent nozzle vibrations, a nominal angular position is defined as in Eq. 31, and "real" positions are determined after some Gaussian random perturbation of the nominal position. Using the same notation as before:

$$\xi = \xi^{'} + \xi_{fr} f_g\left(\sigma_n\right) \tag{32a}$$
$$\zeta_L = \zeta_L^{'} + \zeta_{fr} f_g\left(\sigma_n\right) \tag{32b}$$
$$\zeta_L = \zeta_R^{'} + \zeta_{fr} f_g\left(\sigma_n\right) \tag{32c}$$
$$\tag{32d}$$

2.8 Attitude

Attitude is that which is to be controlled, after all. For representing it, the most common and logical choice is the use of quaternions. In the adopted convention, if a quaternion is a four-dimensional vector defined as $\mathbf{Q} = \begin{bmatrix} q_0 & q_1 & q_2 & q_3 \end{bmatrix}^T$, where q_0 is the scalar part, then it may represent Euler angles as:

$$\mathbf{Q} = \begin{bmatrix} \cos\frac{\Phi}{2}\cos\frac{\Theta}{2}\cos\frac{\Psi}{2} + \sin\frac{\Phi}{2}\sin\frac{\Theta}{2}\sin\frac{\Psi}{2} \\ -\cos\frac{\Phi}{2}\sin\frac{\Theta}{2}\sin\frac{\Psi}{2} + \cos\frac{\Theta}{2}\cos\frac{\Psi}{2}\sin\frac{\Psi}{2} \\ \cos\frac{\Phi}{2}\cos\frac{\Psi}{2}\sin\frac{\Theta}{2} + \sin\frac{\Phi}{2}\cos\frac{\Theta}{2}\sin\frac{\Psi}{2} \\ \cos\frac{\Phi}{2}\cos\frac{\Theta}{2}\sin\frac{\Psi}{2} - \sin\frac{\Phi}{2}\cos\frac{\Psi}{2}\sin\frac{\Theta}{2} \end{bmatrix} \tag{33}$$

Inversely, one can extract Euler angles from a quaternion through:

$$\begin{bmatrix} \Phi \\ \Theta \\ \Psi \end{bmatrix} = \begin{bmatrix} \arctan\frac{2q_2q_1+2q_0q_1}{{q_3}^2-{q_2}^2-{q_1}^2+{q_0}^2} \\ -\arcsin\left(2q_1q_3 - 2q_0q_2\right) \\ \arctan\frac{2q_1q_2+2q_0q_3}{{q_1}^2+{q_0}^2-{q_3}^2-{q_2}^2} \end{bmatrix} \tag{34}$$

One should remember to normalize the quaternion before applying Eq. 34. Defining vector $\omega^* = \begin{bmatrix} 0 & \omega \end{bmatrix}^T$, then the derivative of quaternion $\mathbf{Q}$ containing the representation of attitude is:

$$\dot{\mathbf{Q}} = \frac{1}{2}\mathbf{Q} \otimes \omega^* \tag{35}$$

Being $\otimes$ the quaternion product. Integrated in time, the "next" attitude is found from Eq. 34.

2.9 State Equations

The rocket dynamics has been modeled. One can see that it is a highly non-linear second order system. The choice of state variables should reflect the objectives one must accomplish. In the case of control analysis, angular and linear positions and velocities, plus angular nozzle positions are a good choice. It's possible then to define state variables as vectors:

$$\mathbf{X} = \begin{bmatrix} X & Y & Z \end{bmatrix}^T \tag{36a}$$
$$\mathbf{V} = \begin{bmatrix} V_X & V_Y & V_Z \end{bmatrix}^T \tag{36b}$$
$$\mathbf{\Theta} = \begin{bmatrix} \Psi & \Theta & \Phi \end{bmatrix}^T \tag{36c}$$
$$\omega = \begin{bmatrix} \omega_x & \omega_y & \omega_z \end{bmatrix}^T \tag{36d}$$
$$\mathbf{\Xi} = \begin{bmatrix} \xi & \zeta_L & \zeta_R \end{bmatrix}^T \tag{36e}$$

These can be put together in a state vector $\mathbf{S}$ as:

$$\mathbf{S} = \begin{bmatrix} \mathbf{X}^T & \mathbf{V}^T & \mathbf{Q}^T & \omega^T & \mathbf{\Xi}^T \end{bmatrix}^T \tag{37}$$

Where $\mathbf{Q} = \mathbf{f}(\mathbf{\Theta})$. Deriving Eq. 37, one obtains:

$$\dot{\mathbf{S}} = \begin{bmatrix} \dot{\mathbf{X}}^T & \dot{\mathbf{V}}^T & \dot{\mathbf{Q}}^T & \dot{\omega}^T & \dot{\mathbf{\Xi}}^T \end{bmatrix}^T = \begin{bmatrix} \mathbf{V}^T & \mathbf{a}^T & \dot{\mathbf{Q}}^T & \dot{\omega}^T & \dot{\mathbf{\Xi}}^T \end{bmatrix}^T \tag{38}$$

The input vector is $\mathbf{u} = \begin{bmatrix} \xi_{cmd} & \zeta_{cmd} & \zeta_{diff} \end{bmatrix}^T$, and considering previous developments, Eq. 38 is equivalent to:

$$\dot{\mathbf{S}} = \begin{bmatrix} \mathbf{V} \\ \mathbf{a} \\ \dot{\mathbf{Q}} \\ \dot{\omega} \\ \dot{\boldsymbol{\Xi}} \end{bmatrix} = \begin{bmatrix} \mathbf{f}_1(\mathbf{a}) \\ \mathbf{f}_2(\mathbf{T}, \mathbf{F}_{\mathbf{aero}}, \mathbf{g}, m) \\ \mathbf{f}_3(\mathbf{Q}, \omega) \\ \mathbf{f}_4(\mathbf{M}, \omega, \mathbf{J}) \\ \mathbf{f}_5(\mathbf{u}, t) \end{bmatrix} \tag{39}$$

After examining the equations written so far, Eq. 39 can be reduced to:

$$\dot{\mathbf{S}} = \begin{bmatrix} \mathbf{V} \\ \mathbf{a} \\ \dot{\mathbf{Q}} \\ \dot{\omega} \\ \dot{\boldsymbol{\Xi}} \end{bmatrix} = \begin{bmatrix} \mathbf{f}_1(\mathbf{a}) \\ \mathbf{f}_2(\mathbf{X}, \mathbf{V}, \mathbf{Q}, \boldsymbol{\Xi}, t) \\ \mathbf{f}_3(\mathbf{Q}, \omega) \\ \mathbf{f}_4(\mathbf{X}, \mathbf{V}, \boldsymbol{\Xi}, t) \\ \mathbf{f}_5(\mathbf{u}, t) \end{bmatrix} \tag{40}$$

Explicit forms for functions f_1,..., f_5 are too complex to be written here, although they are what has been developed so far. This system has one vector (three scalars) inputs and outputs, so that the output function g_{out} is defined as:

$$\boldsymbol{\Theta} = \mathbf{g}_{out}(\mathbf{X}, \mathbf{V}, \boldsymbol{\Xi}, t) \tag{41}$$

3. Simulation Analysis and Discussion

The model has been implemented in Simulink® and fed with data based on the Russian rocket Angara 1.2, after some minor adaptations. Tab. 1 summarizes these data.

Aerodynamic data was produced with a simplified CFD analysis and corrected by general literature data (Anderson (2007), Hinson *et al.* (1969) and Hill and Peterson (1965)). The graphs in Fig. 2 show the resulting data for the coefficients in Eqs. 11. Note that here $k_1 = 1.4$ and $k_2 = 2$ were chosen, and interpolation was made using Matlab®'s spline function

3.1 Uncontrolled Simulations

A first set of simulations were made without any control to analyse model behavior and verify if it responds as expected. In this set, the first simulation started the rocket in a perfect vertical attitude at launch, and no maneuver was programmed.

In this scenario, an undisturbed ascent was supposed to happen. Fig. 3 shows that an altitude of about 800 km was reached, which is 150 km more than estimated with Tsiokolvsky's equation (using $V_e = 3757$ m/s). The other graphs in the figure show some other aspects of the rocket's behavior. The drag rise around $Ma = 1$ can be noticed, and the engine shutdown at $t = 240$ s, when the propellant is over. At the same time, the increasing speed starts to diminish, as expected.

Table 1. Data used in simulations

m_s	39812 kg
$m_{o_{ini}}$	92738 kg
$m_{f_{ini}}$	35262 kg
$\dot{m}_p$	533.33 kg/s
z_{CG_s}	24.73 m
$z_{CG_{f,ini}}$	7.45 m
$z_{CG_{f,fin}}$	5.1 m
$z_{CG_{o,ini}}$	18.5 m
$z_{CG_{o,fin}}$	10.0 m
α_f	2.63 kg
T_{SL}	1922 kN
T_{vac}	2085 kN
λ	0.56 rad/s
$(\xi, \zeta_L, \zeta_R)_max$	$\pm$0.14 rad
S_{ref}	10.75 m^2
L_{ref}	41.0 m
D_{ref}	3.0 m
d_T	2.0 m

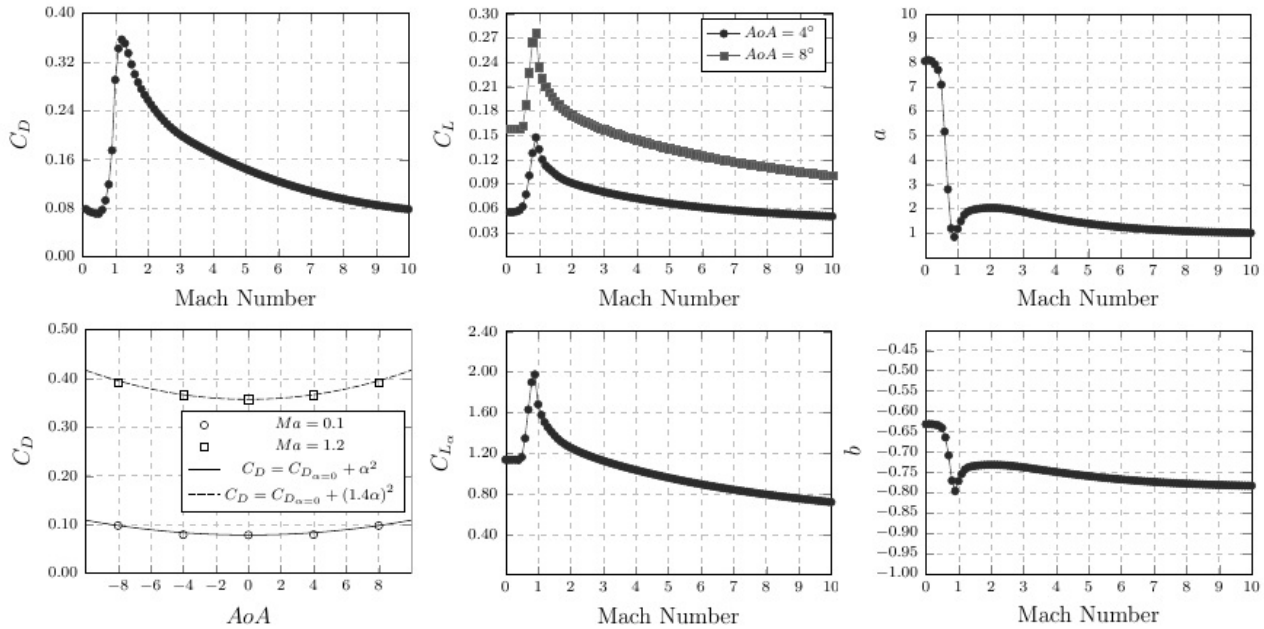

Figure 2. Free body diagram of the generic rocket

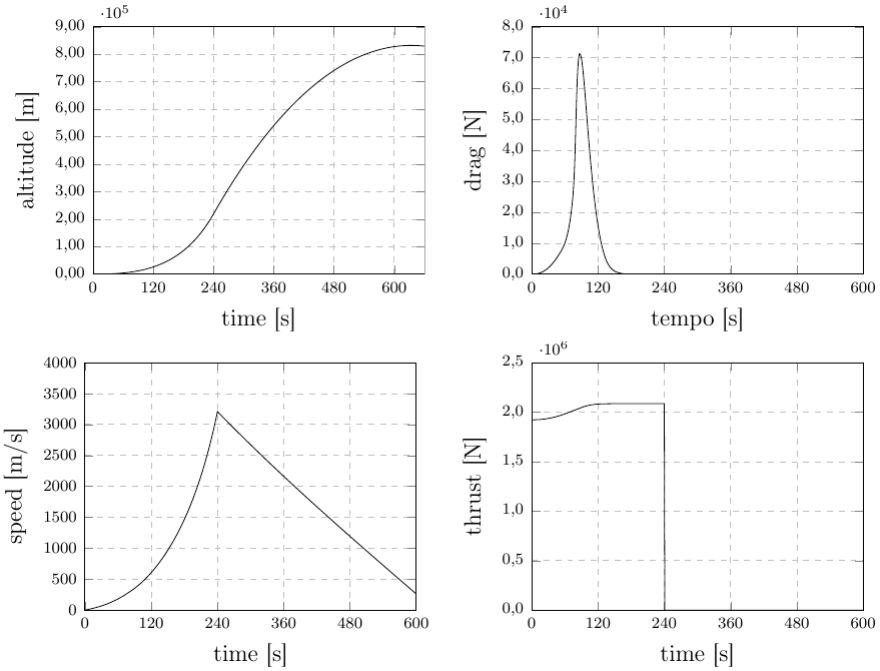

Figure 3. Simulation 1: rocket's apogee, drag, speed and thrust. Disturbances are off, and there is no control input.

As for mass and inertia parameters, the graphs in Fig. 4 show their behavior. Instantaneous mass and moment of inertia about oz axis diminish linearly to a minimum, when there is no more mass variation, as expected from Eqs. 21 to 28. Moment of inertia about ox and oy axes cannot be linear, since they depend on the distances of varying masses to a varying CG position. The same is obviously true for the instantaneous CG position.

One should notice that the parameters shown in the graphs of Fig. 4 depend only on time, and will vary identically in any case where the same set of configuration data is used. These graphics show that the rocket's parameters are behaving as expected, thus validating the model so far.

The second simulation in the set was made with initial attitude $\Phi = 1$ degree. In this case, since CoP position is ahead of CG position, a divergent behavior is expected. In fact, any initial attitude different than the perfect vertical alignment leads to instability of the rocket.

A plot (Fig. 5) was made relating the rocket's trajectory with velocity, lift and body vectors, so that the correct direction of lift force could be verified.

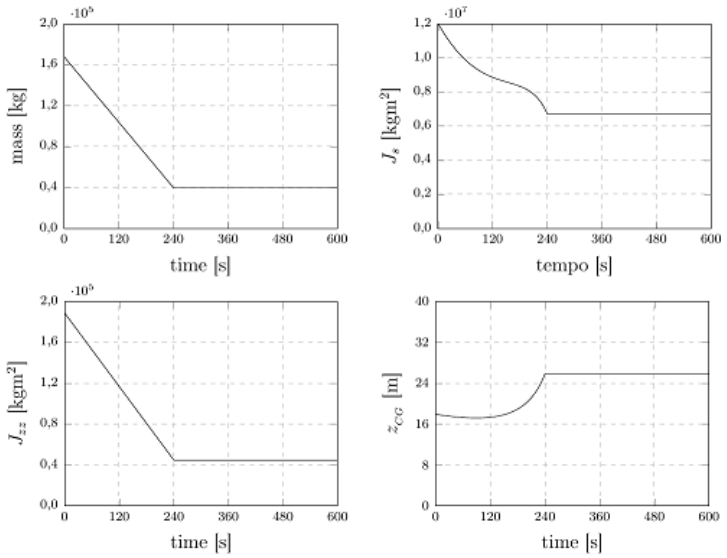

Figure 4. Simulation 1: mass, inertia properties and CG position.

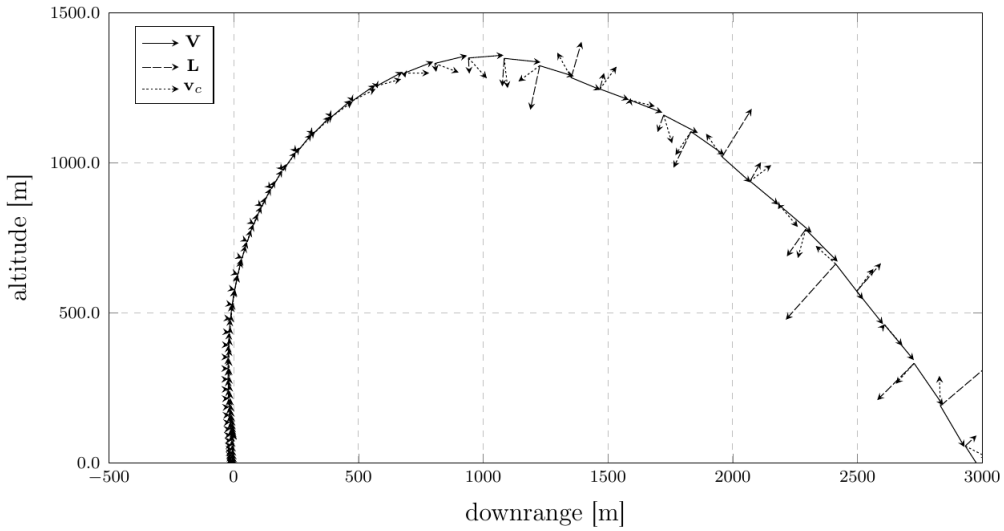

Figure 5. Simulation 2: rockets's trajectory from non vertical initial attitude. Velocity, lift and body vectors relations.

It can be noticed that after about 500 m downrange, the body vector starts to spin around, while lift force, always perpendicular to $\mathbf{V}$ changes direction as the body vector rotates, accordingly to Eq. 8. There are no forces or movement in the OX direction.

Attitude variation and trajectory are given for other 2 different initial Φ values in Fig. 6. The similitudes between the graphs in Fig. 5 and Fig. 6 (right) with the phase portraits should be noticed (Slotine and Li, 1991), further reinforcing the fact that the rocket is unstable for all but the perfect vertical initial condition.

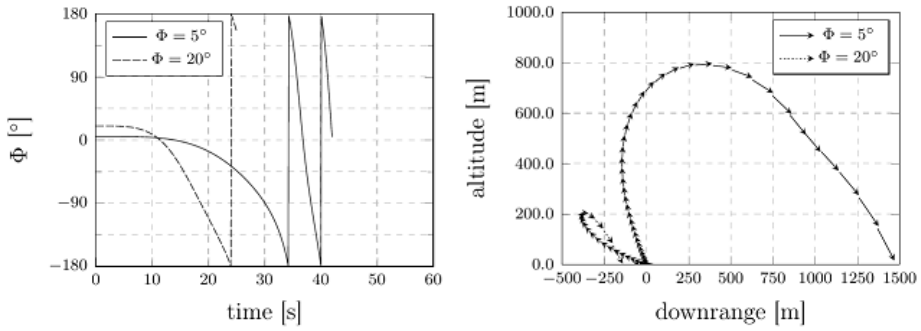

Figure 6. Free body diagram of the generic rocket

Now turning on all disturbances while keeping the controllers off, the perturbations introduced dramatically change the rocket's behavior, as can be seen from the graphs in Fig. 7.

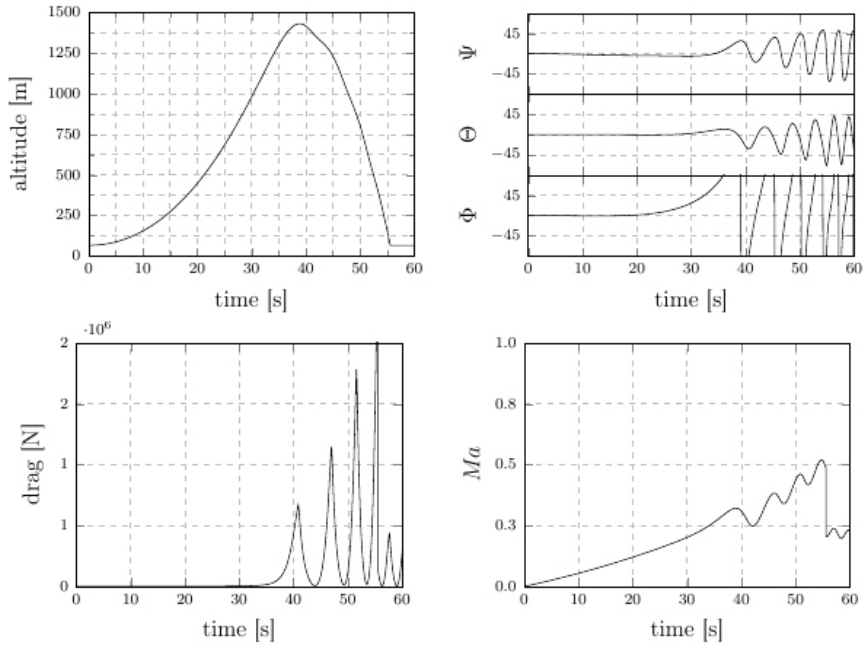

Figure 7. Simulation 3: apogee, drag and mach number. Disturbances are on, and controls are off.

The initial attitude is the same as in the first example (C.f. Fig. 3), i.e. perfect vertical. It can be seen that the perturbations introduced break out the stability of the rocket even in this case. The effects of these instabilities in nozzle position and thrust magnitude can be seen in the graphs of Fig. 8.

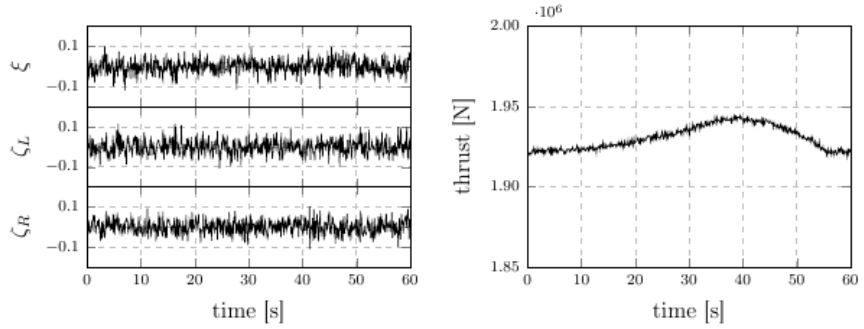

Figure 8. Simulation 3: nozzle positins and thrust magnitude with instabilities on.

The trajectory, together with velocity, body and lift vectors can be seen in Fig. 9. Inspection of this figure, though not trivial, should convince the reader of the appropriate vector's directions.

3.2 Controlled Simulations

For the controlled simulations, a gravity turn trajectory was designed in the form of a sequence of attitudes as functions of time. The sequence of attitudes is such that the maneuver to be executed consists of a rotation around axis oz, together with a banking due east, that translates itself in a variation of Θ and Φ so that the rocket's body axis remains in a plane while maneuvering. The first stage part of this trajectory constitutes the reference values of the Euler angles that the control system is supposed to seek.

Three independent control circuits were implemented, each having an angle error as input, and a vectoring angle as output. The error on Θ is the input to the circuit where desired ζ is the output; the error on Φ is the input where desired ξ is the output, and the error on Ψ is the input where differential ζ is the output.

PID and UIR control laws were set for the control circuits in the model. PID law is well known, while UIR is a development and simplification of sliding mode control, which was developet by Seshagiri and Khalil (2005) and applied in a similiar context by de Sousa and Paglione (2012).

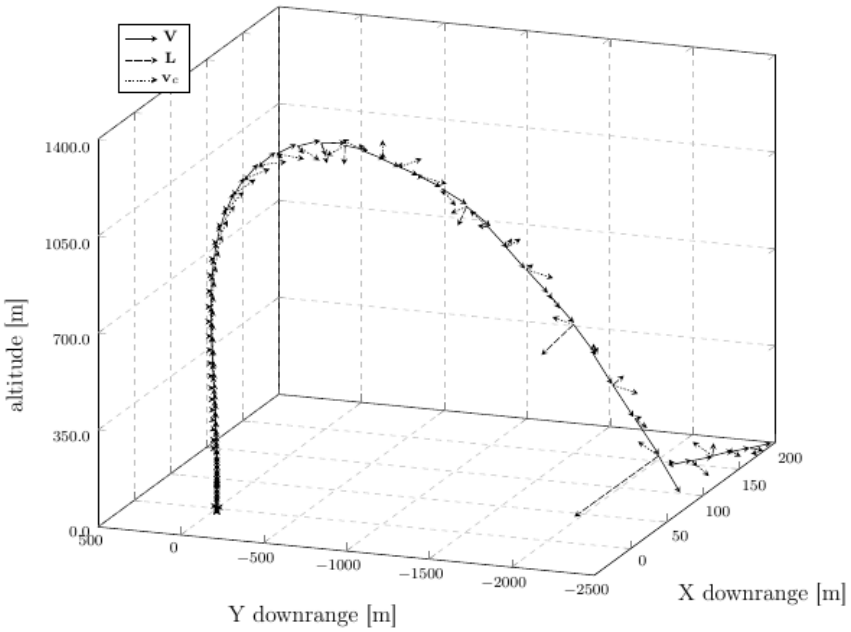

Figure 9. Simulation 3: trajectory and velocity, body and lift vectors.

After some optimization of the controller's gains, the simulation results show that both control laws were able to control the maneuver under the configured conditions.

As can be seen from Fig. 10, there is no substantial difference from the results of the two controllers. In both cases the reference values of the attitude angles were perfectly tracked, being impossible to distinguish from the instantaneous and the desired values of the angles in the graph (left). It can also be seen that the deflection of the nozzles remains inside a small amplitude (right), with a little tilting where transonic speed is achieved.

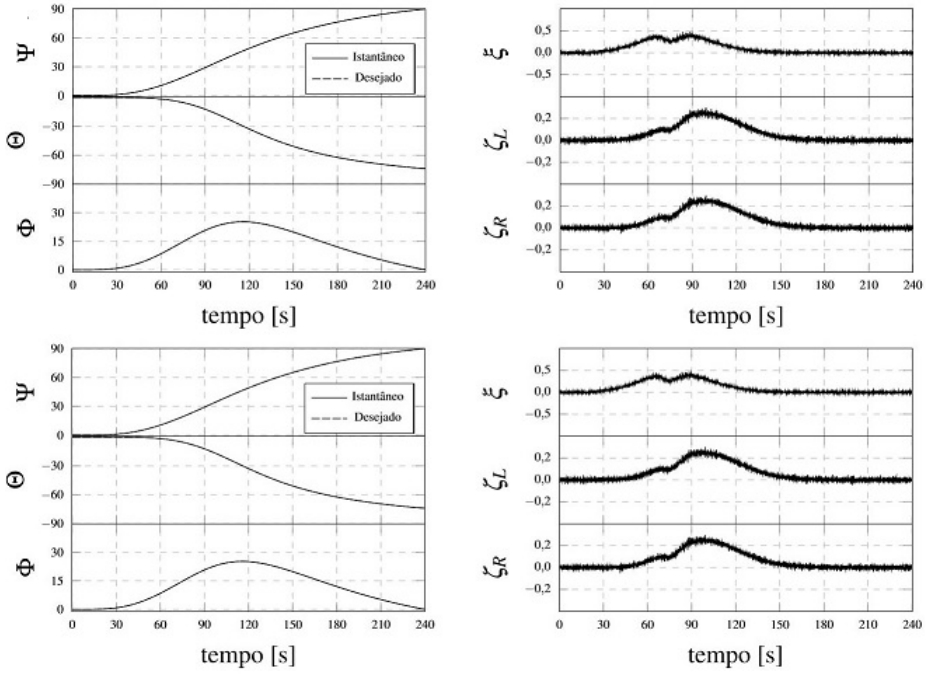

Figure 10. Attitude and thrust vectoring in controlled simulation. PID results above, and UIR results below.

Translation results can be seen in Fig. 11. Again, there is no substantial difference between the PID and the UIR results. It may take some magnifying and close attention to notice that under the UIR control, deviation from the plane – from 0 in the OX axis – was a very little bit greater. Nonetheless, in both cases this deviation remained under 30 cm. Altitude (OZ) and downrange (OY) results are essentially the same, so are the velocity components (right).

One can see that the results just presented are so alike, that no meaningful information can be extracted from them

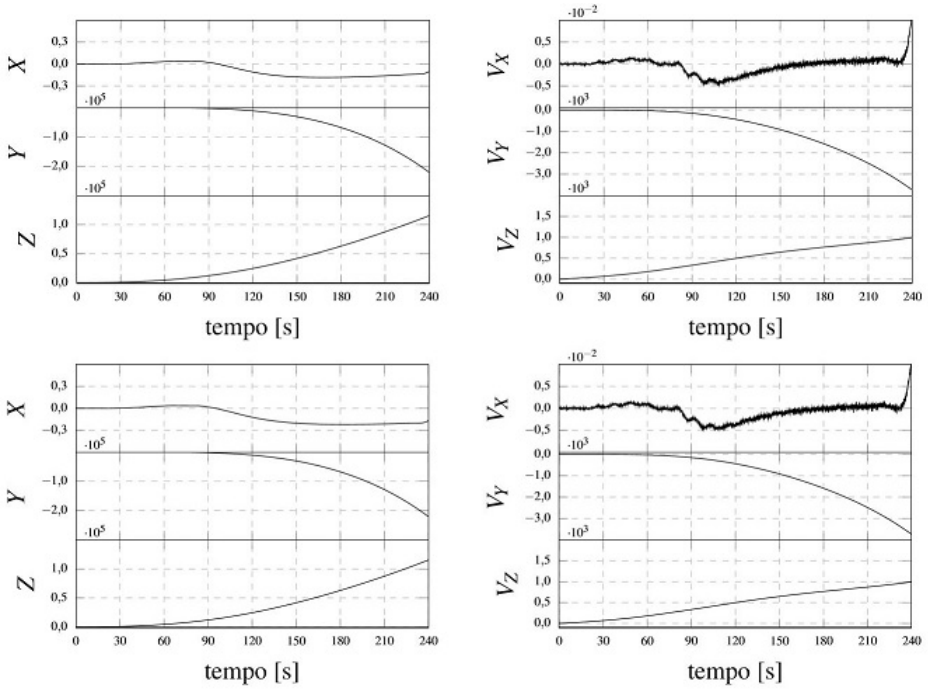

Figure 11. Position and velocity in controlled simulation. PID results above, and UIR results below.

about differences between the two controllers, in terms of efficiency or robustness. For this reason, an error index was defined as the integral of the square root of the sum of the errors in each angle. Fig. 12 (right) allows us to see the evolution in time of the error index in both cases. It can be seen that the PID controller performed a little better, although the difference may be attributed to other factors, like gain tuning, instead of some intrinsic quality of the controller.

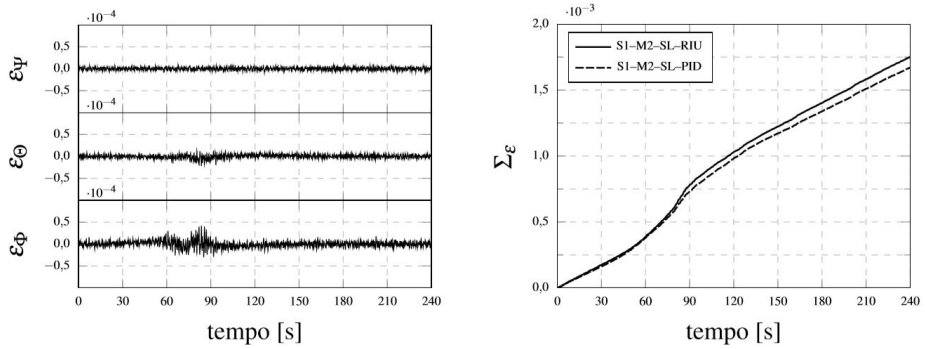

Figure 12. Left: error in each angle of the UIR controlled simulation. Right: evolution of error index in both cases.

It is interesting to observe that a small jump in the error index is present just before $t = 90$ s, which is associated with an increased amplitude in the angle errors (specially in Φ). As already noticed, this region coincides with the transonic speed, where the model imposes greater uncertainties related to aerodynamic phenomena. This results in a more robust requirement on the controllers.

The gravitational turn trajectory and the maneuver associated imposed a precise requirement about position and the state of the velocity vector at the engine shut-off point ($t = 240$s). One may verify how close this requirement could be attained with the proposed maneuver analyzing Fig. 13.

Again, there is no noticeable difference between the results from the two controllers, but the attained results fall short from the requirements. Altitude is about 25 km lower than required, and horizontal velocity is about 250 m/s slower.

The aerodynamic forces acting on the rocket can be seen in the graphs of Fig. 14. Drag rise around transonic speed is noticeable, as much as the increase in lift. The increase in the uncertainty of these forces around $Ma = 1$, just as desired in the modeling, can also be noticed.

One of the general requirements in rocket launching is that the transverse forces (like lift) remain as low as possible. This is so to minimize stresses in the rocket structure. Usually, modern rockets have some kind of thrust control which allows them to keep dynamic pressure (velocity) under certain limits. In the present case, no thrust variation was simulated, and through the whole flight power was at maximum.

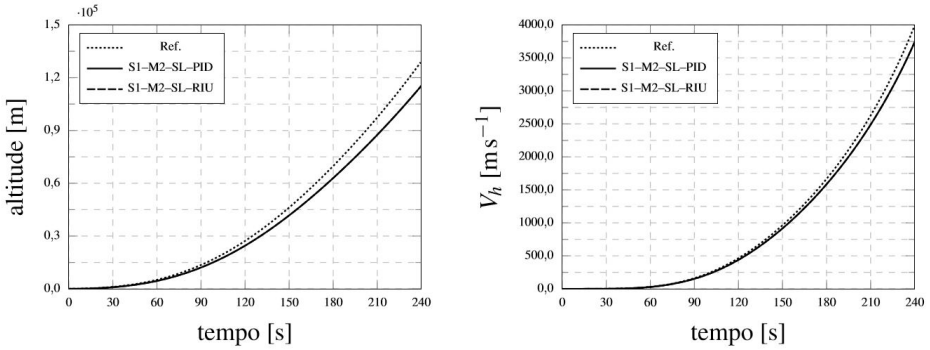

Figure 13. Altitude and horizontal velocity in controlled simulation.

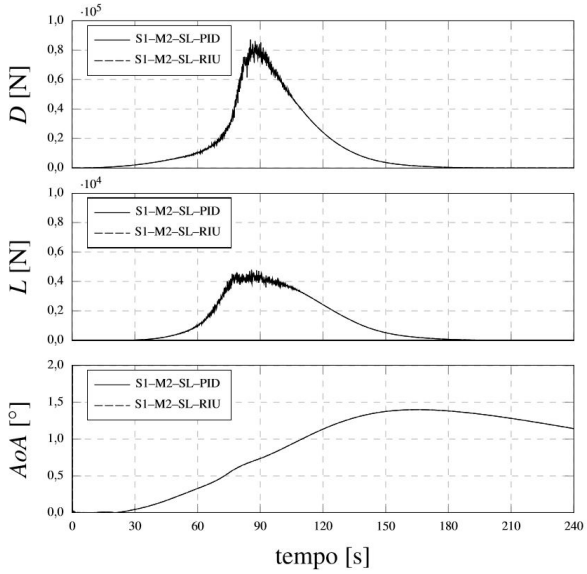

Figure 14. Drag, lift and angle of attack in the controlled simulation.

Nevertheless, as can be seen in the lift evolution curve, lift force was maintained under 5000 N, which is as small as the lift force needed to sustain the flight of an ultralight aircraft. Angle of attack reaches its maximum ($\alpha \approx 1.5$ degrees) when altitude is such that no aerodynamic forces are significant. These results can be considered good enough, although only an structural analysis could give the final word.

4. Conclusions

From the data and results presented so far, it seems reasonable to conclude that the model here proposed is good enough in the stablished limits. Nonetheless, many improvements can yet be made. To cite a few: A better and complete reference frame system to include planetary form, giving the model uses other than only control analysis. This is particularly true when considering the results shown in Fig. 13, suggesting a potential for maneuver and trajectory analysis.

Also, Eqs. 11 are inaccurate when α is large. In general, this is not a problem since one does not expect a rocket to operate in this condition. But it would give a better understanding of its behavior when uncontrolled, as in the case of its free fall, for instance.

Other sources of perturbations should be introduced. Atmospheric winds, gusts and turbulence would be a good addition, allowing to understand and develop better control laws. The same is true for attitude sensor noise. The ability of a control law as simple and popular as the PID to keep such good track of the reference values was surprising. Maybe it was due to the fact that the kind of maneuver in question is of very slow execution, and no atmospheric disturbances were present.

The same may be true for the almost identical results obtained from the two controllers. Nevertheless, only with the maturation of the model – which is a work in progress – a better understanding of the role of each phenomenon can be understood, and a better appreciation of the effectiveness and robustness of different control laws can be realized.

5. ACKNOWLEDGEMENTS

The authors thank FAPEMIG for the support.

6. REFERENCES

The list of references must be introduced as a new section, located at the end of the paper. The first line of each reference must be aligned at left. All the other lines must be indented by 0.5 cm from the left margin. All references included in the reference list must have been mentioned in the text.

References must be listed in alphabetical order, according to the last name of the first author. See the following examples:

Anderson, J.D., 2007. *Fundamentals of Aerodynamics*. McGraw-Hill Series in Aeronautical and Aerospace Engineering. McGraw Hill, New York, NY (USA), 4th edition. ISBN 978-0-07-295046-5.

Baldesi, G., Sciacovelli, D. and Thirkettle, A., 2006. "Simulation tool for generic launcher flight dynamic-control interaction analysis". In *Proceedings of 6th International Symposium on Launcher Technologies: Flight Environment Control for Future and Operational Launchers*. Munich, Germany.

Betts, K.M., Rutherford, R.C., McDuffie, J., Johnson, M.D., Jackson, M. and Hall, C., 2007. "Time domain simulation of the nasa crew launch vehicle". In *Proceedings of the AIAA Modeling and Simulation Technologies Conference and Exibit*. Hilton Head, USA.

de Sousa, M.S. and Paglione, P., 2012. "Proposição de valores para os graus relativos de parâmetros da dinâmica de voo de aeronaves". In *VII Congresso Nacional de Engenharia Mecânica*. Associação Brasileira de Ciências Mecânicas - ABCM, São Luis, MA (Brasil), p. 10.

Diebel, J., 2006. "Representing attitude: Euler angles, unit quaternions, and rotation vectors". URL `https://pdfs.semanticscholar.org/5c0e/dc899359a69c3769da238491f93e7a2f6d6d.pdf`.

Hill, P. and Peterson, C., 1965. *Mechanics and Thermodynamics of Propulsion*. Addison-Wesley Series in Aerospace Science. Addison-Wesley Publishing Company, Inc., Reading, MA (USA), 2nd edition.

Hinson, W.F., Langhans, R.A. and Fournier, R.H., 1969. "Aerodynamic characteristics in pitch of a 1/7-scale model of a two- and three-stage rocket configuration at mach numbers of 0.4 to 4.63". NASA TN D-5378 NASA TN D-5378, NASA, Langley Research Center, Langley, VA.

NOAA, NASA and USAF, 1976. "United states standard atmosphere 1976". NOAA S/T 76-1562 NOAA S/T 76-1562, NOAA, NASA, USAF, Washington, DC (USA).

Seshagiri, S. and Khalil, H.K., 2005. "Robust output feedback regulation of minimum-phase nonlinear systems using conditional integrators". *Automatica*, Vol. 41, No. 41.

Slotine, J.J.E. and Li, W., 1991. *Applied Nonlinear Control*. Prentice Hall, Englewood Cliffs, NJ (USA). ISBN 0-13-040890-5.

Tewari, A., 2007. *Atmospheric and Space Flight Dynamics*. Birkhäuser, Boston, MA (USA), 1st edition.

Zipfel, P.H., 2007. *Modeling and Simulation of Aerospace Vehicle Dynamics*. American Institute of Aeronautics and Astronautics (AIAA), Reston, VA (USA), 2nd edition.

7. RESPONSIBILITY NOTICE

The author(s) is (are) the only responsible for the printed material included in this paper.