DIGITAL ISSN 2953-4739
RSO - Safety Journal
(may-jul, 2023): [97 - 112]
https://id.caicyt.gov.ar/ark:/s29534739/s7vkglxst

RESEARCH ARTICLE

Three Degree of Freedom Trajectory Simulator for Axisymmetric Spacecraft

Aníbal Vettorel
Engineer and Professor (UTN).
Lead Project Engineer in
Opener-Renewable Energies.

avettorel@frh.utn.edu.ar

Carlos Mario Carbonel
Engineer.

ccarbonel@frh.utn.edu.ar

Agustín Tristant
Engineering Student.

Aerospace Technology
Group (GTA), UTN Aerospace
Technology Research Group.
National Technological
University (UTN), Haedo
Regional School, Buenos
Aires, Argentina.

Keywords: *Transportation - Safety – Aerospace - Launch Vehicles - 3dof - Flight Simulators.*

Received: 04/06/23
Approved: 25/06/23

Abstract

This work presents a flight simulator with 3 degrees of freedom as the first stage of development of a future simulator with 6 degrees of freedom, to be used within the framework project of the Aerospace Technology Group (GTA, Spanish acronym): "Design and optimization of a University Orbital Launcher for Cubesat satellites". The language chosen for its implementation is Matlab® and applies to vehicles with time-varying mass and propulsive properties, moving in atmospheric flight as well as in orbital trajectories. The elliptical Earth model based on the WGS 84 reference ellipsoid is used, taking as inertial reference frame a geocentric system aligned with the ecliptic plane, with the x-axis in the direction of the vernal point (J2000 system).

Paper presented at the XII Argentine Congress of Space Technology in April 2023.

ACKNOWLEDGEMENTS: The authors of the paper in this publication wish to express their gratitude to all members of the Aerospace Technology Group (GTA) for their constant and unwavering support in favor of science and research, by providing facilities and equipment for manufacturing, integration, and testing activities. They also extend their special thanks to Alexis Maximiliano Caratozzolo and Tomás Martín Ziroldo, members of the GTA, for their contributions in implementing a model rocket database, and to Pablo M. González, a GTA research professor, for his support.

1. INTRODUCTION

The ascent of a rocket is fundamentally different from the accelerated ascent of an airplane, primarily due to the faster decrease in the former's mass and the nearly constant thrust with altitude. For these reasons, a rocket can deliver a payload to an orbital trajectory in a fraction of the time it takes an airplane to reach its cruising altitude. The simplest ascent is the vertical trajectory of a sounding rocket, commonly used in meteorological observations of the upper atmospheric layers, for which a closed-form solution can easily be obtained. However, the curved ascent of a spacecraft launch vehicle, from φ = 90º to φ ≈ 0, is much more interesting from a modeling perspective. It is important to note that even a slight build-up of the normal load factor could cause the thin, shell-like structure of the launch vehicle (often almost completely filled with liquid propellants) to collapse. Therefore, since maneuvering is completely ruled out during launch, it is natural to ask what causes the launch vehicle to change its flight trajectory from vertical to nearly horizontal. The answer lies in the combined effects of gravity and planetary rotation. This natural tendency of the trajectory to curve from the planet's surface to orbit is called a gravity turn.

An attitude control system is an indispensable feature of the launch vehicle, which always maintains a zero angle of attack despite small atmospheric disturbances. Rocket attitude control is carried out through thrust vectoring, using a feedback mechanism based on gyroscopes and a control law that keeps the vehicle in its inherently unstable open-loop equilibrium. In addition to attitude control, a guidance system is required to maintain a nearly vertical flight during the first few seconds after launch, during which the launch tower is cleared, and the trajectory is unstable due to near-zero flight velocity. The gravity turn is initiated by disabling the automatic guidance system as soon as the vehicle has accelerated enough for | φ | to become small. However, attitude control is always performing its task of maintaining zero angles of attack and lateral slip throughout the entire launch.

2. METHODOLOGY

For the simulation of atmospheric flights or orbital trajectories with rapid mass variations, such as those of launch vehicles and satellites, it is necessary to consider a series of special assumptions regarding the choice of the reference frame used and the modeling of gravitational forces.

During an orbital flight, Earth's geometry cannot be assumed to be spherical, as the flattening effect at the poles introduces components of gravitational forces in the transverse direction, which primarily affect orbital trajectories. Additionally, Earth's rotation does not allow a reference frame fixed to the Earth to be assumed as inertial, and reference systems aligned with other astronomical references must be used. For this reason, the chosen equations of motion refer to an inertial system aligned with Earth's ecliptic (J2000 frame), and for the calculation of the gravitational field, an elliptical Earth geometry (WGS 84 ellipsoid) is considered.

Regarding the vehicle modeling, it should be noted that, for preliminary analysis stages, a model considering only translational degrees of freedom may be sufficient. The next stage of this development will allow for a more accurate assessment of the vehicle's performance during flight by including rotational movements in the equations of motion, thus obtaining a more complex six-degree-of-freedom model.

The integration of the equations of motion is carried out using a fourth-order Runge-Kutta scheme, utilizing latitude, longitude, azimuth, flight angle, altitude, and initial velocity as state variables (Tewari, 2016 [1]) for the determination of the transformation matrix from the body-fixed system to the inertial system. The code validation is performed by verifying known orbital trajectories, and simulations of the selected launch vehicle are also compared as a reference.

2. DEVELOPMENT

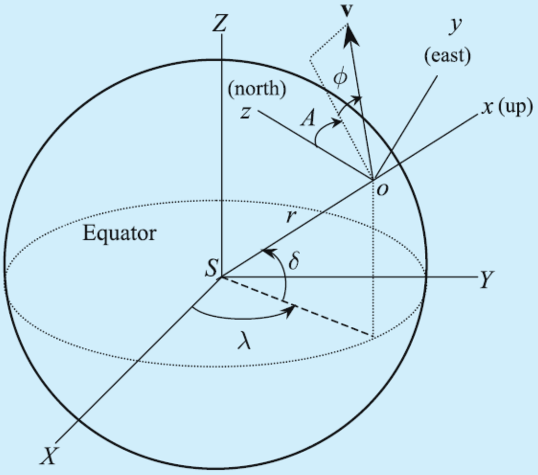
3.1. Equations of Motion

Atmospheric flight is influenced by the forces created by the vehicle's movement relative to the atmosphere and, therefore, requires a reference system fixed to it. Since a planet's atmosphere rotates with it, we use a reference frame fixed to the planet to express the atmospheric flight equations. This is the rotating planet-centered frame (SXYZ) with axes represented by the unit vectors I, J, K, respectively.

To determine the complete set of equations governing translation within the atmosphere, a set of 3 kinematic equations and 3 dynamic equations is necessary. Once the relative velocity vector (v, φ, A) is determined from the solution of the dynamic equations, the position vector (r, δ, λ) is calculated from the kinematic equations, thus completing the trajectory solution.

The kinematic equations are derived from the expression of velocity and a coordinate transformation between a planet-centered reference system (SXYZ) and the local horizon (oxyz), which has axes ox (i), oy (j), and oz (k), aligned with the local vertical (upward), local East, and local North, respectively, as shown in Figure 1.

Figure 1. Reference frames: planet-fixed system and local horizon [1]



Therefore, both reference systems are related by:

$$\begin{Bmatrix} \mathbf{i} \\ \mathbf{j} \\ \mathbf{k} \end{Bmatrix} = \begin{pmatrix} \cos\delta\ \cos\lambda & \cos\delta\ \sin\lambda & \sin\lambda \\ -\sin\lambda & \cos\lambda & 0 \\ -\sin\delta\ \cos\lambda & -\sin\delta\sin\lambda & \cos\delta \end{pmatrix} \begin{Bmatrix} \mathbf{I} \\ \mathbf{J} \\ \mathbf{K} \end{Bmatrix} \qquad (1)$$

The relative velocity can be expressed in the local horizon frame as follows:

$$\mathbf{v} = v\,(\sin\phi\ \mathbf{i} + \cos\phi\ \sin A\ \mathbf{j} + \cos\phi\ \cos A\,\mathbf{k}) \qquad (2)$$

The relative velocity is:

$$\mathbf{v} = \dot{r}\mathbf{i} + \boldsymbol{\Omega} \times (r\mathbf{i}) \qquad (3)$$

Where

$$\boldsymbol{\Omega} = \Omega_x\mathbf{i} + \Omega_y\mathbf{j} + \Omega_z\mathbf{k} \qquad (4)$$

Is the velocity of the local horizon reference frame (oxyz) relative to the planet-centered frame (SXYZ). Substituting equation 4 into equation 3:

$$\mathbf{v} = \dot{r}\mathbf{i} + r\Omega_z\mathbf{j} + r\Omega_y\mathbf{k} \qquad (5)$$

A comparison of equations 5 and 2 results in the following kinematic relationships:

$$\dot{r} = v \sin\phi \tag{6}$$

$$\Omega_y = -\frac{v}{r}\cos\phi \ \cos A \tag{7}$$

$$\Omega_z = \frac{v}{r}\cos\phi \ \sin A \tag{8}$$

From the coordinate transformation between the planet-centered and local horizon reference systems:

$$\mathbf{\Omega} = \dot{\lambda}\mathbf{K} - \dot{\delta}\mathbf{j} = \dot{\lambda}\sin\delta\,\mathbf{i} - \dot{\delta}\mathbf{j} + \dot{\lambda}\cos\delta\,\mathbf{k} \tag{9}$$

which, when compared with Eqs. (7) and (8), yields:

$$\dot{\delta} = \frac{v}{r}\cos\phi \ \cos A \tag{10}$$

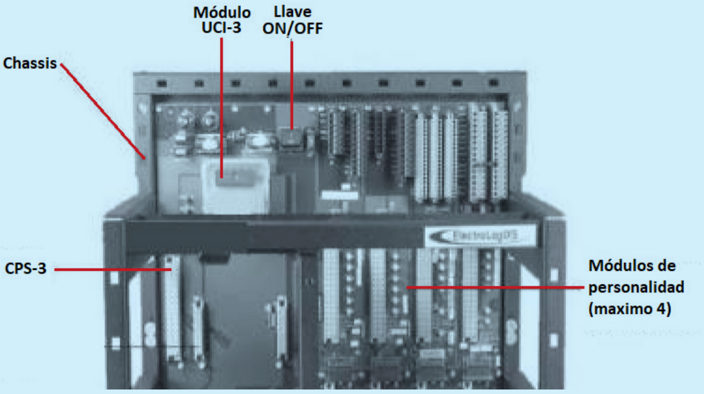
$$\dot{\lambda} = \frac{v}{r}\frac{\cos\phi \ \sin A}{\cos\delta} \tag{11}$$

The kinematic equations of motion relative to a rotating planet, Eqs. (6), (10), and (11), are the same as those used in orbital mechanics.

The dynamic equations of motion are derived by expressing the inertial acceleration and force vectors in a wind-axis frame (Sxv yv zv), with axes in directions opposite and normal to the relative wind velocity.

The orientation of the wind axes (Sxv yv zv), relative to the local horizon frame (oxyz), is shown in Fig. 2.

Figure 2. Orientation of the wind axes relative to the local horizon [1]



The two frames are therefore related by

$$\begin{Bmatrix}\mathbf{i_v}\\ \mathbf{j_v}\\ \mathbf{k_v}\end{Bmatrix} = \begin{pmatrix}\sin\phi & \cos\phi\sin A & \cos\phi\cos A\\ 0 & \cos A & -\sin A\\ -\cos\phi & \sin\phi\sin A & \sin\phi\cos A\end{pmatrix}\begin{Bmatrix}\mathbf{i}\\ \mathbf{j}\\ \mathbf{k}\end{Bmatrix} \tag{12}$$

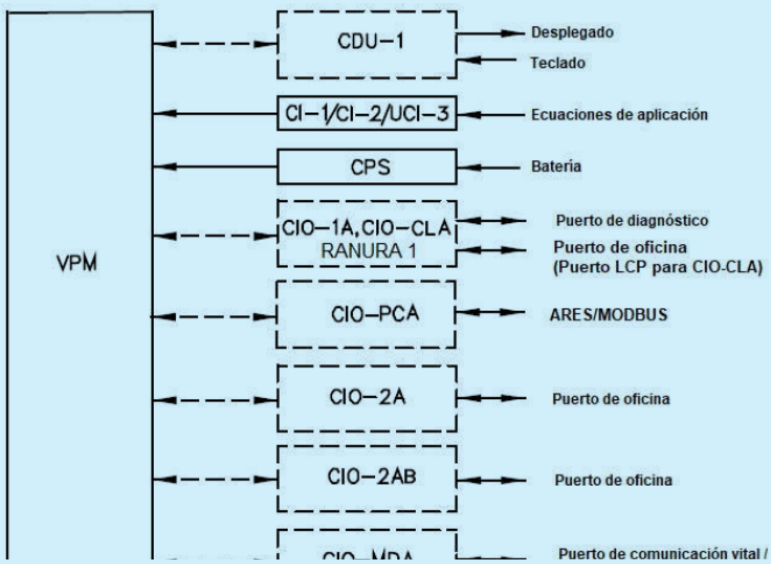
The dynamic equations of translational motion in vector form are derived from Newton's second law, expressed as:

$$\mathbf{f} = m\mathbf{a_I} = m\frac{d\mathbf{v_I}}{dt} \qquad (13)$$

where f denotes the net external force and m represents the total mass of the vehicle. Since f consists of the aerodynamic force vector (-Div + fY jv - Lkv), thrust fT, and gravitational force (-mgci + mgδk), resolved in the wind axes (as shown in Fig. 3), we have:

$$\begin{aligned}&\mathbf{f} = (f_T\cos\epsilon\cos\mu - D - mg_c\sin\phi + \\ &mg_\delta\cos\phi\cos A)\mathbf{i_v} + (f_T\sin\mu + f_Y - \\ &mg_\delta\sin A)\mathbf{j_v} + (-f_T\sin\epsilon\cos\mu - L + \\ &mg_c\cos\phi + mg_\delta\sin\phi\cos A)\mathbf{k_v}\end{aligned} \qquad (14)$$

Figure 3. External forces in the wind axes [1]



Now, since the force vector is conveniently resolved along the wind axes, we must transform the inertial acceleration of the vehicle's center of mass, expressed in the local horizon (oxyz), into the wind axis frame (Sxvyvzv) in order to write the dynamic equations of motion [1]:

$$\begin{aligned}&m\,\dot{v} = f_T\cos\epsilon\cos\mu - D - m\,g_c\sin\phi + \\ &m\,g_\delta\cos\phi\cos A\end{aligned} \qquad (15)$$

$$\begin{aligned}&m\,v\cos\phi\;\dot{A} = m\frac{v^2}{r}\cos^2\phi\sin A\tan\delta + \\ &f_T\sin\mu + f_Y - m\,g_\delta\sin A + \\ &m\,\omega^2\,r\sin A\sin\delta\cos\delta - \\ &2m\,\omega\,v(\sin\phi\;\cos A\cos\delta - \cos\phi\sin\delta)\end{aligned} \qquad (16)$$

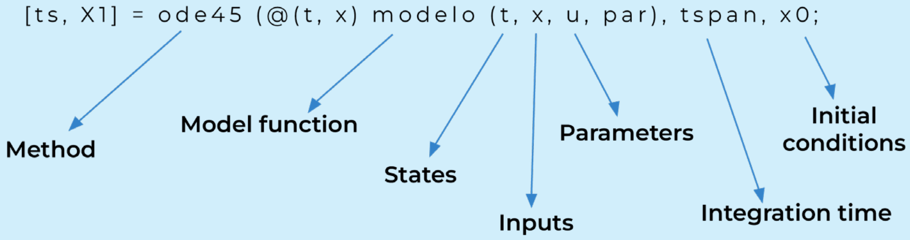
$$\begin{aligned}&m\,v\,\dot{\phi} \\ &= m\frac{v^2}{r}\cos^2\phi + f_T\sin\epsilon\cos\mu + L \\ &- mg_c\cos\phi - mg_\delta\sin\phi\cos A \\ &+ m\,\omega^2\,r\cos\delta\,(\sin\phi\;\cos A\sin\delta \\ &+ \cos\phi\cos\delta) + 2m\,\omega\,v\;\sin A\cos\delta\end{aligned} \qquad (17)$$

Here, the left side of the first equation denotes the acceleration along the instantaneous flight path, while the second and third represent the centripetal acceleration caused by the curvature of the flight path in the local horizontal and local vertical directions, respectively. The centripetal and Coriolis acceleration terms, due to planetary rotation, appear on the right side of these equations of motion, along with the respective components of the external force.

Thus, all atmospheric trajectories must satisfy the kinematic equations (6), (10), and (11), as well as the dynamic equations (15), (16), and (17). Their solution vector, r(t), λ(t), δ(t), v(t), φ(t), A(t), yields the position and velocity based on time. However, atmospheric, gravity, aerodynamic, and propulsion models are also required, which are nonlinear in nature, necessitating an iterative numerical solution procedure such as Runge-Kutta methods.

While it is common to use constant gravitational acceleration (flat planet approximation) or, at best, Newton's law of gravitation (spherical planet model), for atmospheric flight, we will take into account radial and latitudinal gravity variations (non-spherical planet model). Our atmospheric model will also be as general as possible, using a 21-layer model. The aerodynamic model we use considers the variation of aerodynamic force with position (altitude) and velocity based on dimensionless coefficients. Finally, the propulsion model must account for the variation of thrust with altitude and velocity, as well as fuel consumption, which in turn determines the vehicle's instantaneous mass. The additional set of nonlinear ordinary differential equations resulting from these models is combined with the previously derived kinematic and dynamic equations. These are solved using Matlab®'s internal solver ode45, which is based on an explicit Runge-Kutta (4,5) formula, the Dormand-Prince pair [7], and has the following syntax:

Figure 4. Ode45 syntax [8]



```
[ts, X1] = ode45 (@(t, x) modelo (t, x, u, par), tspan, x0;
```

All the necessary modules for the program will be explained in the following sections.

3.2. Modeling of the Earth

The Earth is not perfectly spherical; it more closely resembles an ellipsoid of revolution, with an equatorial radius larger than its polar radius. Considering the planet as having a non-spherical radius is crucial for improving the accuracy of a launch vehicle's trajectory calculation program, as it enhances the prediction of movement.

For trajectory calculation, having a reference is essential, and since the Earth's radius is not constant everywhere, a reference sea level is defined by calculating a surface radius R based on the local geocentric latitude δ. This is useful for accounting for atmospheric variations with respect to altitude.

We will use Britting's development (1971 [2]), which derived the following expansion to calculate the Earth's radius based on latitude:

$$R = R_e\left(1-\frac{\epsilon}{2}(1-\cos 2\delta)+\frac{5\,\epsilon^2}{16}(1-\cos 4\delta)\ \ldots\right) \qquad (18)$$

Where

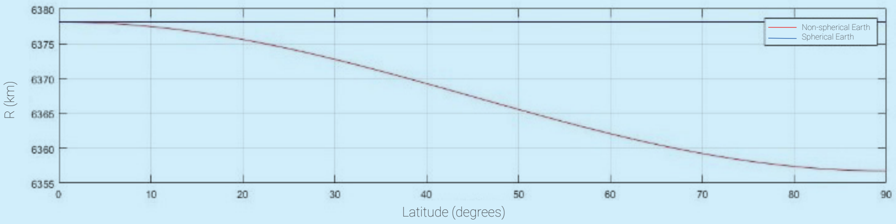
$$\epsilon = 1-\frac{R_p}{R_e} \qquad (19)$$

Is the ellipticity of the planet, a ratio between the polar radius (Rp) and the equatorial radius (Re). Generally, it is a small number, with =0 representing a perfectly spherical planet. For Earth, =1/298.257.

The interesting aspect of this expansion is that in most cases, it is sufficiently accurate to retain only the first two terms of the series, leading to the following approximation:

$$R \approx R_e(1 - \epsilon \sin^2 \delta) \qquad (20)$$

Figure 5. Earth radius for spherical and non-spherical models



3.3. Gravity Modeling

As mentioned earlier, the Earth, like other celestial bodies, deviates from the perfect symmetry of a spherical shape. This is due to its rotation around an axis, which produces a centrifugal mass displacement, resulting in an axisymmetric body that bulges at the equator and flattens at the poles. Additionally, the axisymmetric shape of a planet can deviate from a sphere in many other ways. For calculations, to account for the deviation caused by all these effects, spherical harmonics are used.

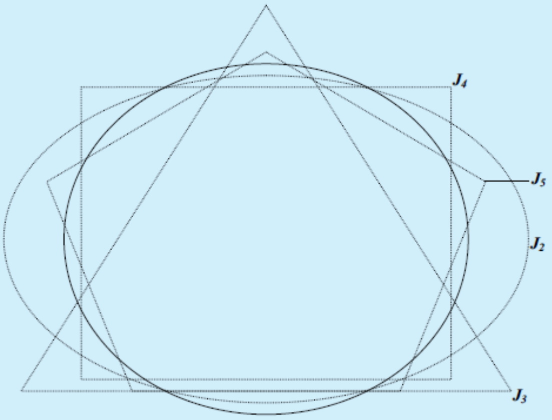
To model spherical harmonics, Jeffery's constants are used—a set of parameters that describe the shape and mass of a celestial body in gravity modeling and allow for representing its variations. The main values are known as J2, J3 and J4.

The J2 constant describes the body's shape as an ellipsoid rather than a perfect sphere and is related to the body's flattening..

The J3 constant is used to describe the body's asymmetry relative to its equatorial plane.

The J4 constant is used to describe the shape of the body in terms of its curvature.

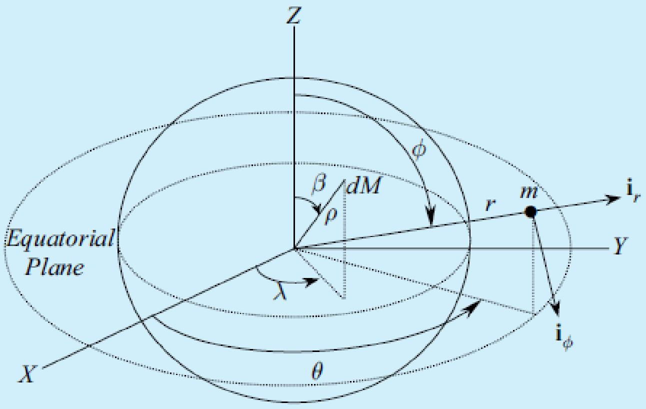
Figure 5. Earth radius for spherical and non-spherical models



Newton's law of gravity is not valid for a test mass near a non-spherical planet. Therefore, we adopt the energy approach to derive the non-spherical gravity model. In this way, gravitational acceleration is obtained from the gradient of the gravitational potential with respect to the position vector:

$$\mathbf{r} = r\boldsymbol{i}_r + r\phi\boldsymbol{i}_{\boldsymbol{\phi}} \qquad (21)$$

Figure 7. Position vector of a test mass [1]



The final result of the gravity calculation based on the aforementioned hypotheses is:

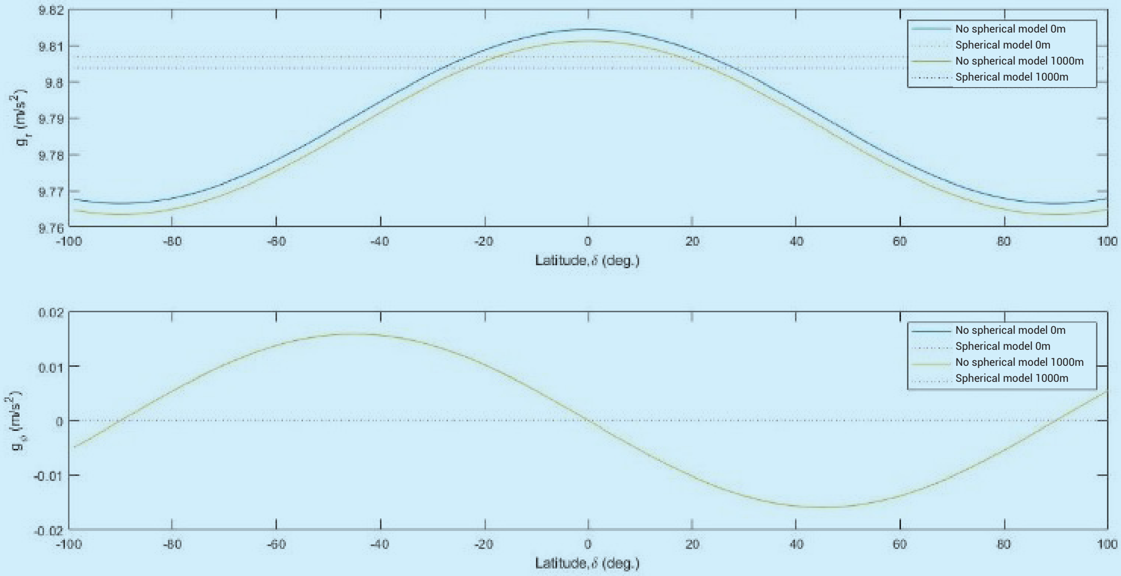
$$\begin{aligned}\mathbf{g} = -\left(\frac{\partial\Phi}{\partial r}\right)^T &= -\frac{\partial\Phi}{\partial r}\boldsymbol{i}_r - \frac{\partial\Phi}{r\,\partial\phi}\boldsymbol{i}_{\boldsymbol{\phi}} \\ &= g_r\boldsymbol{i}_r + g_\phi\boldsymbol{i}_{\boldsymbol{\phi}}\end{aligned} \qquad (22)$$

$$\begin{aligned}g_r &= \\ &-\frac{GM}{r^2}\left(1 - 3J_2\left(\frac{R_e}{r}\right)^2 P_2(\cos\phi)\right. \\ &\qquad - 4J_3\left(\frac{R_e}{r}\right)^3 P_3(\cos\phi) \\ &\qquad \left. - 5J_4\left(\frac{R_e}{r}\right)^4 P_4(\cos\phi)\right)\end{aligned} \qquad (23)$$

$$\begin{aligned}g_\phi &= \\ &-\frac{3GM}{r^2}\left(\frac{R_e}{r}\right)^2 \sin\phi\cos\phi\left(J_2\right. \\ &+\frac{1}{2}J_3\left(\frac{R_e}{r}\right)\sec\phi\,(5\cos^2\phi - 1) \\ &\left.+\frac{5}{6}J_4\left(\frac{R_e}{r}\right)^2(7\cos^2\phi - 1)\right)\end{aligned} \qquad (24)$$

Where two components of gravity can be considered: one in the radial direction gr (from the center of the Earth to the point where gravity is measured), and another in the tangential direction gr, which acts perpendicular to the first. Due to a non-zero transverse gravity component, g , the direction of g differs from the radial direction, while its radial component, gr, is of lesser magnitude compared to that predicted by a spherical gravity model, as observed in Fig. 8.

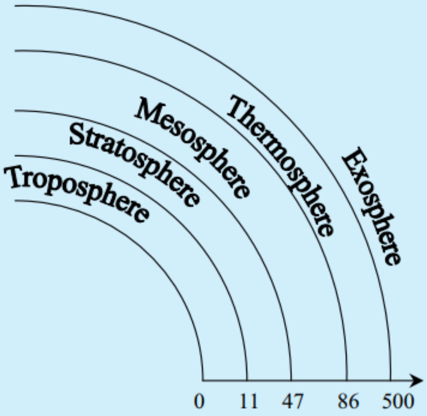
Figure 8. Comparison of the radial and tangential components of gravity for a spherical and a non-spherical planet at 0 m and 1000 m altitude



3.4. Atmospheric Modeling

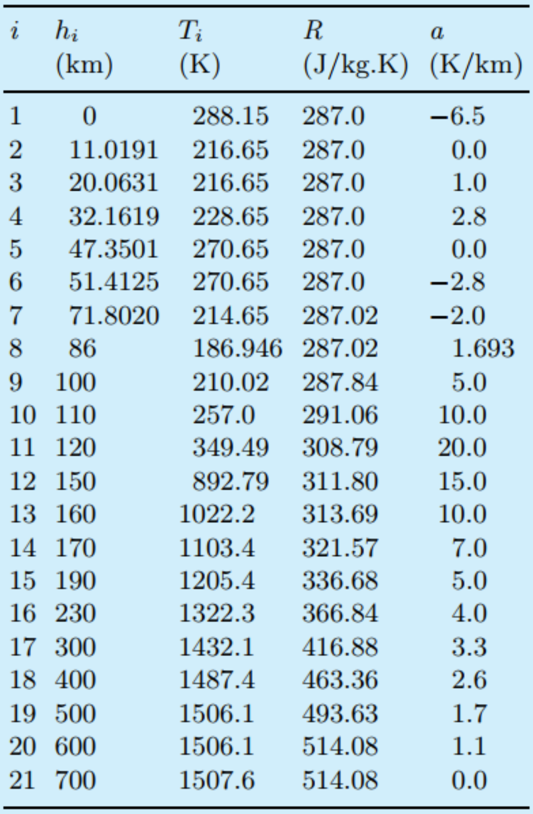
The aerodynamic loads on an atmospheric flight vehicle depend on the thermodynamic properties of the atmospheric gases, making a carefully developed model crucial for the analysis and design of aerospace vehicles. Standard atmospheric models are based on hydrostatic and thermal equilibrium and consist of multiple layers, each with different temperature variations with altitude. The troposphere, extending from standard sea level to h = 11 km (tropopause), experiences a temperature that decreases linearly with altitude. The next layer above 11 <h≤ 47 km, known as the stratosphere, consists of three layers with constant (isothermal) temperatures and linearly increasing temperatures at different rates, respectively. The operation of most aircraft is limited to the troposphere and stratosphere. Immediately above the stratosphere is the mesosphere, which extends up to h = 86 km and features one isothermal layer along with two consecutive layers with linearly decreasing temperatures. A linear variation of temperature with geopotential altitude is a standard approximation used in most atmospheric models. However, the upper atmospheric layers (thermosphere and exosphere) do not achieve thermal and chemical equilibrium, and therefore exhibit an inherently nonlinear temperature variation with altitude. An appropriate exponential model can be used to fit the variation of density with altitude over a specific (and limited) range of altitudes.

Figure 9. Atmospheric layers [1]



We will follow the convention of the 1976 U.S. Standard Atmosphere [3] in the range 0 ≤ h ≤ 86 km and above 86 km in altitude, we will use the 1962 U.S. Standard Atmosphere [4], which conveniently models all layers with linearly varying temperatures up to h = 2000 km. The numerical values of the hybrid standard atmosphere consisting of 21 layers are tabulated in Table 1.

Chart 1. Standard Atmosphere derived from the 1962 and 1976 U.S. Standard Atmosphere [1]



i	h_i	T_i	R	a
	(km)	(K)	(J/kg.K)	(K/km)
1	0	288.15	287.0	−6.5
2	11.0191	216.65	287.0	0.0
3	20.0631	216.65	287.0	1.0
4	32.1619	228.65	287.0	2.8
5	47.3501	270.65	287.0	0.0
6	51.4125	270.65	287.0	−2.8
7	71.8020	214.65	287.02	−2.0
8	86	186.946	287.02	1.693
9	100	210.02	287.84	5.0
10	110	257.0	291.06	10.0
11	120	349.49	308.79	20.0
12	150	892.79	311.80	15.0
13	160	1022.2	313.69	10.0
14	170	1103.4	321.57	7.0
15	190	1205.4	336.68	5.0
16	230	1322.3	366.84	4.0
17	300	1432.1	416.88	3.3
18	400	1487.4	463.36	2.6
19	500	1506.1	493.63	1.7
20	600	1506.1	514.08	1.1
21	700	1507.6	514.08	0.0

3.5. Computational Code

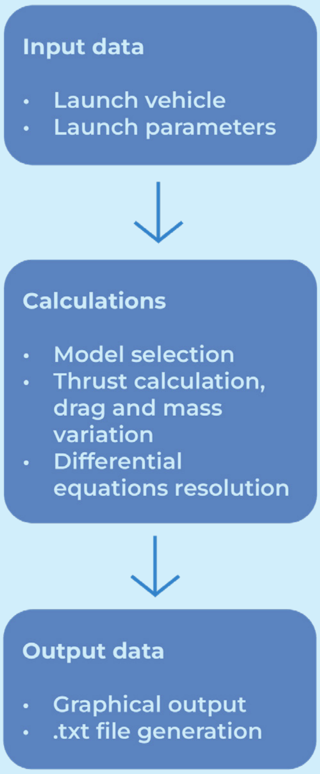
The trajectory simulation program was developed in Matlab using a modular programming approach. This type of programming allows for breaking down a complex program into smaller, manageable programs, making it easier to write, debug, and correct. The main advantages of this method lie in the simplicity and organization of the main code, the ability to choose which modules to use, the facilitation of collaborative development, and the ease of enhancing the program by adding new modules.

The program begins with a data input module that collects the basic information necessary for the launch vehicle to be analyzed, which includes mission details, geometric properties of the launcher, fuel characteristics, and the launch site.

Next, we have the fundamental modules for accurate calculations, which model the various physical phenomena described initially and are based on the established hypotheses. These modules include:

- Earth modeling
- Gravity modeling
- Atmosphere modeling
- Calculation module (solving equations of motion)
- Results data collection
- Graphical output

Figure 10. Flowchart

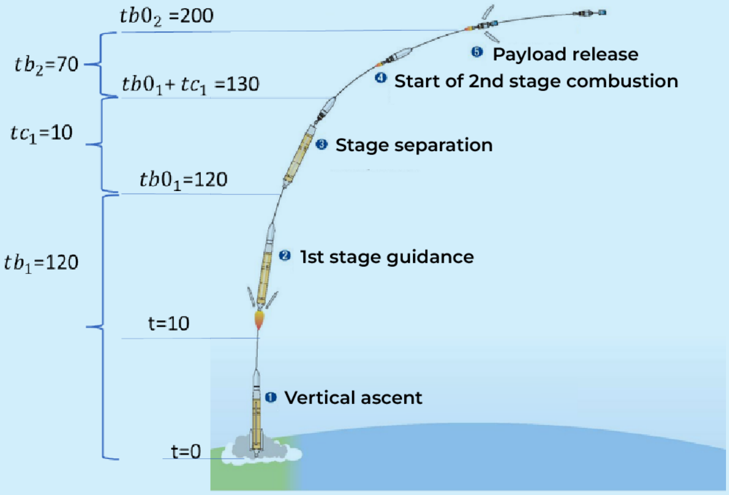


The program divides the different phases of flight into events. Each event represents a specific moment during the launch vehicle's trajectory, which is important for the control and monitoring of the launch.

They include:

- Launch
- *Start of gravity turn*
- Start of stage combustion
- End of stage combustion
- Stage separation
- Maximum altitude achieved

Figure 11. Event diagram of a generic 2-stage launch vehicle based on burn time tb and stage separation time tc



After executing the Calculation Module and solving the equations of motion, the variation of the variables is plotted throughout the entire trajectory.

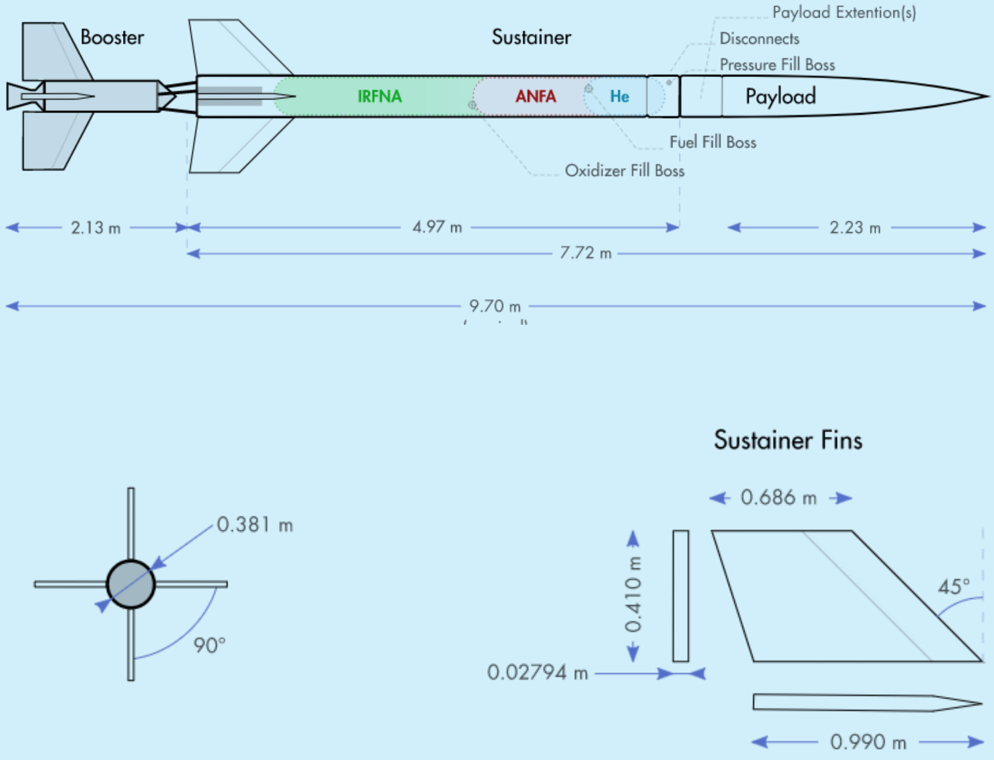
The graphical outputs are grouped into three sets:

- Characteristic Curves: The variables of altitude, velocity, acceleration, angle of attack, and azimuth angle are plotted based on time. Additionally, the path of latitude vs. longitude is graphed along the trajectory.
- Atmosphere: pressure, temperature, density, and speed of sound are visualized based on altitude in the modeled atmosphere.
- Launcher: thrust, mass, mach number, and drag are observed based on time.

3.5. Results Obtained

The validation of the code is carried out by verifying known trajectories. The chosen launch vehicle for this validation is the Aerobee 150A (Russ and Randall, 1961 [5]).

Figure 12. Diagram of the Aerobee 150A launch vehicle [9]



The Aerobee 150A is a suborbital rocket with four fins, approximately 10 meters tall and 0.38 meters in diameter. It is a tower-launched vehicle stabilized by free-flight fins, using a solid propellant booster for stage 1 and a liquid propellant sustainer for stage 2.

The payload is mounted on the nose structure. The mission sequence begins with the ignition of the stage 1 engine at t=0s. After t=0,3s, the stage 2 engine ignites, and both stages burn together for 2.2 s (it is important to note that the stage 1 engine has a deflector nose to divert the exhaust from stage 2). At t = 2,5 s, the stage 1 engine completes its burn and falls away, while the stage 2 engine continues to burn fuel until t=51,8s.

The input data in the program is summarized in Charts 2 and 3:

Chart 2. Input data: launch vehicle

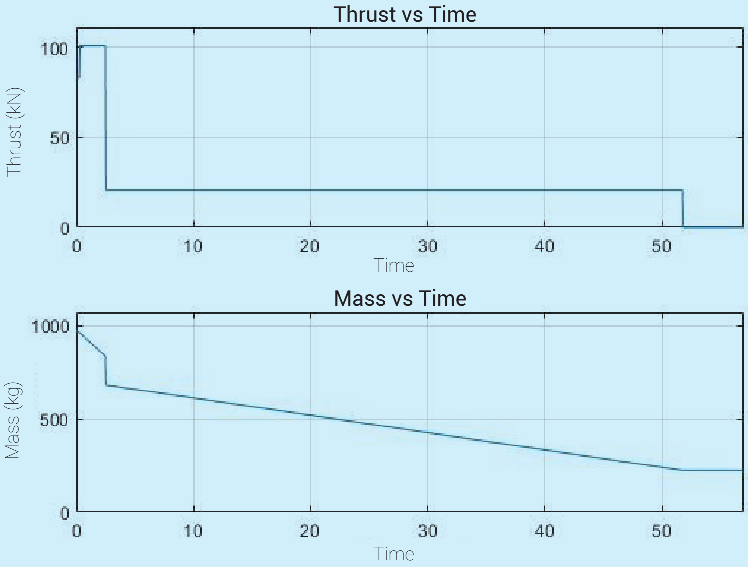
Launch Vehicle Data			
Sustainer	Structural mass	mS	132.9 kg
	Fuel mass	mp	478.3 kg
	Payload mass	mL	90.7 kg
	Specific impulse	Isp	198 s
	Start of combustion time	ti	0.3 s
	Burn time	tb	51.5 s
	Separation time	tc	0 s
	Reference area	areaS	0.1141 m^2
Booster	Structural mass	mS	153.3 kg
	Fuel mass	mp	118.8 kg
	Specific impulse	Isp	178 s
	Start of combustion time	ti	0 s
	Burn time	tb	2.5 s
	Separation time	tc	0 s
	Reference area	areaB	0.1141 m^2

Chart 3. Input data: launch conditions

Launch Conditions		
Altitude	alt	5 m
Length	lon	-75.439339 º
Latitude	lat	37.883255 º
Initial velocity	vi	0 m/s
Launch angle	fpa	88 º
Azimut	Azi	180 º

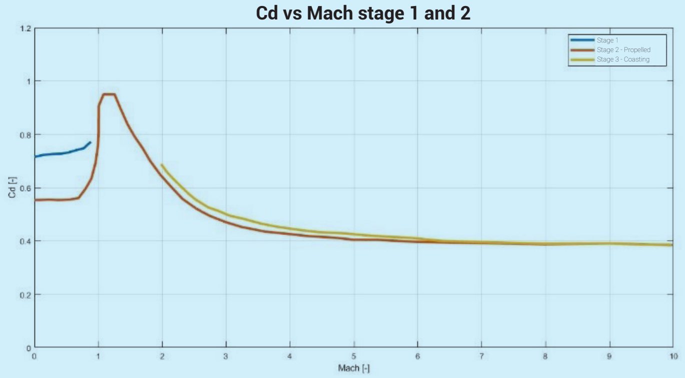
In this way, the simulator calculates thrust and current mass based on time:

Figure 13. Thrust and mass based on time



Regarding the drag coefficient, it is entered as one of the input data in this first version of the program (later, there are plans to add a module for calculating drag):

Figure 14. Drag coefficient based on mach number



Additionally, the following considerations will be taken for this first stage of calculation:

- The aerodynamic force vector is composed solely of drag forces, meaning there are no lateral loads (fY jv=0) or lift forces (Lkv=0).
- The thrust vector aligns with the direction of the vehicle's longitudinal axis.

Thus, the solution vector r(t), λ(t), δ(t), v(t), φ(t), A(t) can be represented in Figures 15, 16, and 17, with the caveat that the Earth's radius r(t) is substracted from the position vector R(δ) base don latitude, in order to express the results in terms of altitude. It was also decided to graph the acceleration (Eq. 15).

Figure 15. Latitude and length

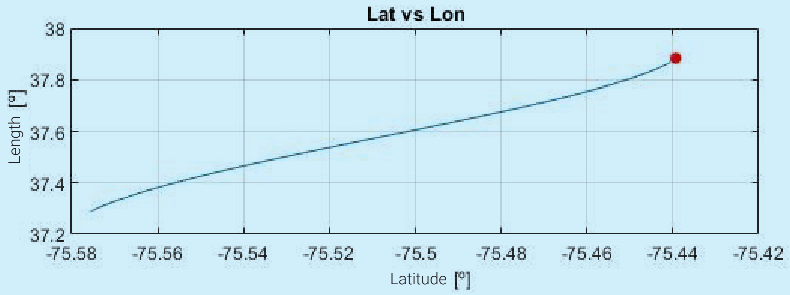
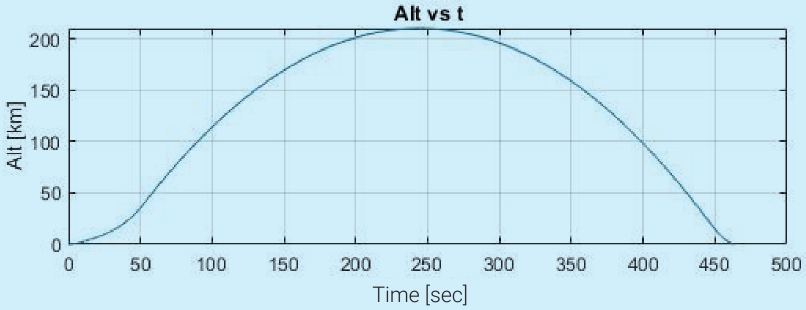


Figure 16. Altitude, velocity and acceleration based on time



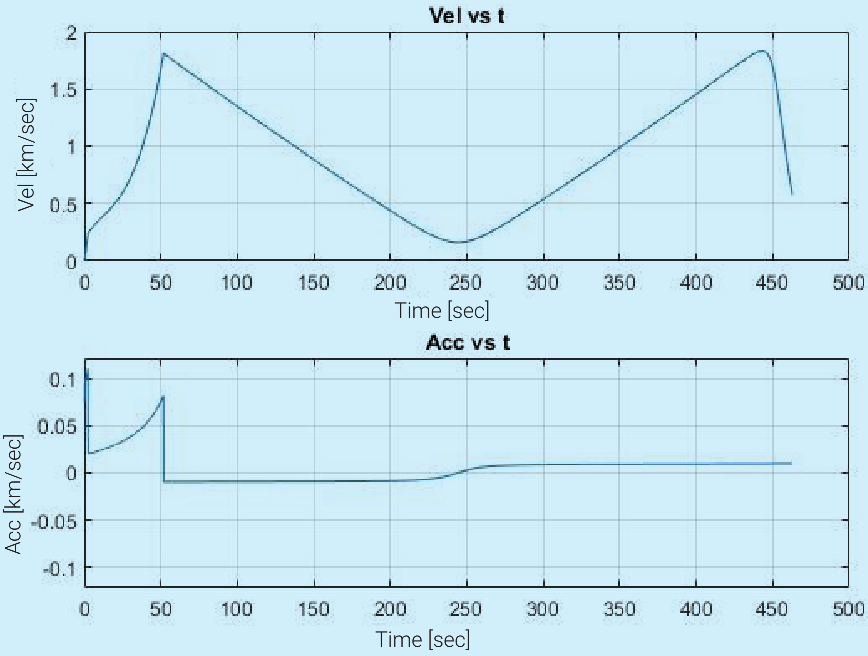
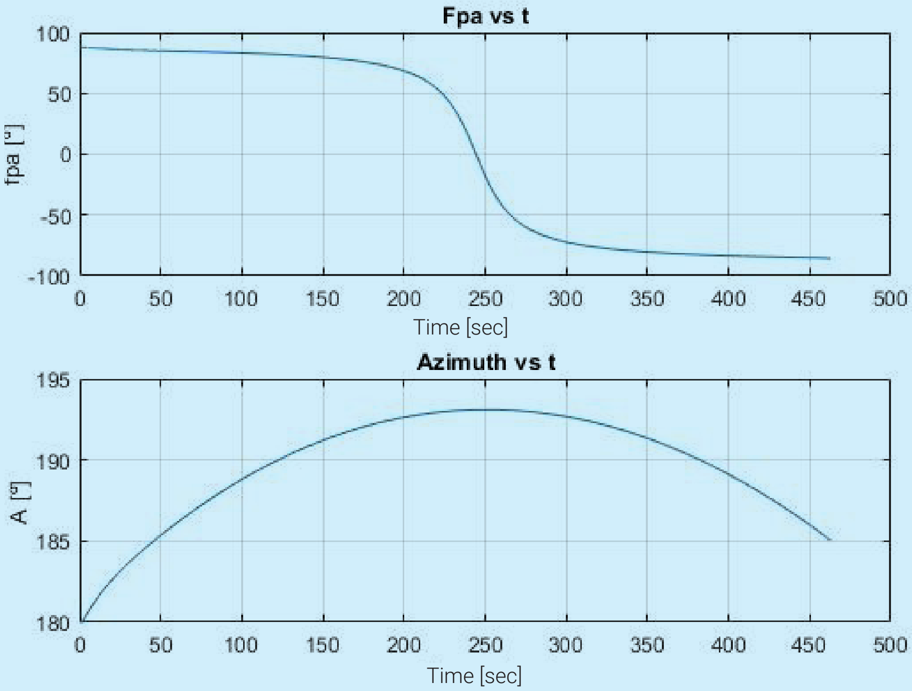
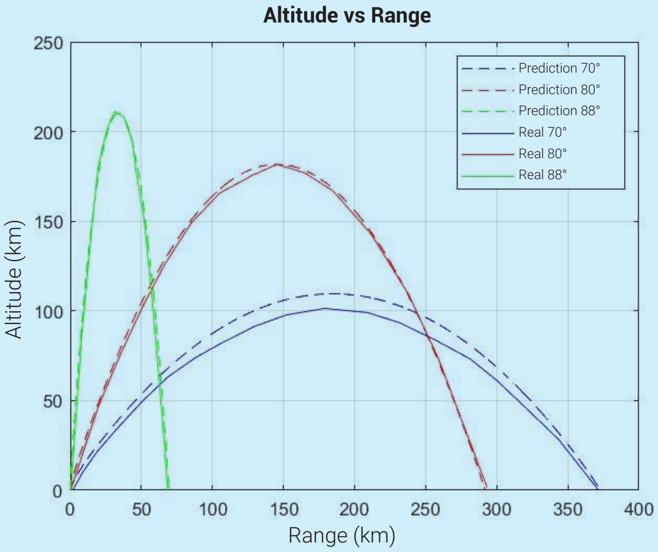


Figure 17. Flight path angle and azimuth based on time



For comparative purposes, the actual altitude-range curves for three different launch configurations are plotted alongside those obtained by the program (Fig. 18). It can be observed that the greater the launch angle (more vertical), the smaller the error between the actual and simulated curves.

Figure 18. Altitude-range for launch angles of 88º, 80º and 70º



CONCLUSIONS

The results obtained allow us to conclude that as the launch angle increases, the error between the simulation and the actual values decreases. For this reason, the three degrees of freedom program is considered satisfactory and lays the groundwork for the development of a future six degrees of freedom simulator, which will be used within the framework project of the Aerospace Technology Group (GTA), "Design and Optimization of a University Orbital Launcher for CubeSat-type satellites." A modified sounding rocket will be used as input data for the injection of microsatellites into low Earth orbit (LEO). This is a preliminary modification of the service module of a two-stage sounding rocket with solid propellant, similar to those widely used in Argentina during the years of high-altitude exploration and microgravity experiments. This modification transforms it into a third powered stage that enables the launch and subsequent insertion of CubeSat-type satellites into low Earth orbit (LEO) as a low-cost operational alternative to contribute to the national launcher program. It is designed to fly in an unguided mode, stabilized by rotation. The main characteristics of the third stage result from a simplified review of existing commercial launchers, focusing on data from the SS-520 modified by ISAS/JAXA, the Japan Aerospace Exploration Agency, which successfully launched the SS-520-5 and placed the 3U CubeSat TRICOM-1R into orbit in February 2018.

This achievement makes the SS-520-5 the first rocket to attain Guinness World Record certification as the "Smallest Orbital Rocket," and it serves as the best starting point for analyzing improvements in sounding rockets to transform them into small satellite launchers.

REFERENCES

[1] A. Tewari (2006), Atmospheric and space flight dynamics: Modeling and simulation with MATLAB and Simulink. Estados Unidos: Birkhauser Boston.
[2] Britting, K.R. (1971), Inertial Navigation Systems Analysis. Wiley Interscience: Somerset, NJ.
[3] Committee on Extension to the Standard Atmosphere (COESA) (1976), U.S. Standard Atmosphere 1976. Washington, DC: Government Printing Office.
[4]COESA (1962), U.S. Standard Atmosphere 1962. Washington, DC: Government Printing Office.
[5] K. M. Russ y F. W. Randall (1961), Performance summary for the Aerobee 150A Sound Rocket Vehicle, report no. AST/EIR-13319, Vought Astronautics.
[6] Mircea-Vlad Radulescu (2016), Three Degree of Freedom Sounding Rocket Model with Pitch Control Study. https://www.academia.edu/26204338/Three_Degree_of_Freedom_Sounding_Rocket_Model_with_Pitch_Control_Study
[7] The MathWorks Inc, Mathworks.com. https://la.mathworks.com/
[8] S.A. Castaño Giraldo, Ecuaciones Diferenciales EDO, Control Automático Educación. https://controlautomaticoeducacion.com/matlab/ecuaciones-diferenciales-edo/
[9] Nathan Bergey, Aerobee 150 A, Open Aeroespace. https://open-aerospace.github.io/Aerobee-150/