Bachelor's thesis

Degree program in Information Technology

NTIETS13

2017

Gustavo Emanuel Santos Dinis

SUN-SYNCHRONOUS SATELLITE SIMULATOR

– An OpenModelica simulator

TURKU AMK
TURKU UNIVERSITY OF APPLIED SCIENCES

BACHELOR'S THESIS | ABSTRACT

TURKU UNIVERSITY OF APPLIED SCIENCES

NTIETS13

2017 | 58

Gustavo Emanuel Santos Dinis

SUN-SYNCHRONOUS SATELLITE SIMULATOR

- An OpenModelica simulator

Widely used in Aerospace industries, simulators create a virtual environment suitable of verifying and validating mission facts and figures without leaving the Earth's atmosphere. With help from such simulators, it is possible to test physics principles, satellite equipment and software, inject and detect failures on the system, even before flight, saving funds and resources from the mission. The aims of this study were to develop a simulator for a Sun-synchronous satellite where, given a start date and mission parameters, orbits for the Sun, Moon and Earth, in addition to the satellite actuation and attitude data were calculated. Subsequent to this, satellite sensors generate data perceived from the surrounding environment, data that are used to adjust the satellite attitude through means of actuators. The modeling process involved an analysis of laws of planetary motion as well as satellites motion and on-board equipment procedures. As result, an OpenModelica simulation environment capable of generating astronomical and satellite equipment data was created. The results attained from the simulator were then analyzed against data from known sources and tools such as General Mission Analysis Tool, developed by NASA, public and private contributors. Such data reveal that the calculated orbits are accurate to an extent of less than 500 km. Similarly, the satellite attitude and equipment presented reliable data although improvements to actuation and to satellite dynamic and kinematic equations were needed. This work resulted in a dependable simulator capable of solving the inputs specified for the mission into meaningful data ready to be analyzed.

KEYWORDS:

OpenModelica, Satellite, Simulator, Sun-Synchronous, Attitude, Sensors, Spacecraft

ACKNOWLEDGMENTS

Foremost, I would like to thank all TUAS teachers and staff for the warm welcome and amazing experience that they provide me during my stay in Finland. Especially, I would like to express my gratitude to Professor Jari-Pekka Paalassalo, Professor Patric Granholm and Ms. Päivi Oliva for all the help, attention and supervision they gave me. Without this experience in Finland, I would never have become the person that I am and certainly never had found the career path that I ambitioned.

I would like to thank Critical Software S.A. for the opportunity of doing the simulator that generated a topic and investigating for this thesis. Specifically, I would like to thank Mr. João Esteves for accepting me, guide me and teach me about satellite systems. Another special thanks to Mr. Eduardo Pacheco that, with is knowledge, wise commentaries and incredible patience, guide me through the implementation of the simulator.

Obviously, I would like to thank my parents, Paula Duarte and Luís Teixeira, for everything, without them nothing of this would exist. Also a warm thank you for my friends and girlfriend.

Last but not least, my mentor, source of inspiration and friend, Professor Luis Santos from ISEC, that I don’t know how to thank for such generosity and help during all processes that led to the completion of this thesis.

To all of the above and unmentioned persons that helped me, a warm thank you.

CONTENT

FIGURES

EQUATIONS

PICTURES

TABLES

LIST OF ABBREVIATIONS (OR) SYMBOLS

AOCS	Attitude orbit control system
AU	Astronomical Unit
CESS	Coarse Earth Sun Sensors
DCM	Direction Cosine Matrix
ECEF	Earth-centered, Earth-fixed
ECI	Earth-centered Inertial
ESA	European Space Agency
GMAT	General Mission Analysis Tool
IMU	Inertial Measurement Unit
LEO	Low Earth Orbit
LLA	Latitude, Longitude, Altitude coordinates
MAG	Magnetometer
MTQ	Magnetic Torquer
nT	Nanotesla
OBC	On Board Computer
RCS	Reaction Control System
RW	Reaction Wheel
STR	Star Tracker
UT	Universal Time
3D	Three dimensions

1 INTRODUCTION

Simulators help humans formulating a connection between reality and a computer-generated environment. They exist to simulate specific scenarios that are not trivial to produce on Earth even more, their creation is usually associated with massive costs.

The space industry can easily demonstrate hundreds of examples of how they cannot fully test their products without a simulated environment. For example, in order to test the spacecraft and its equipment, on a specific orbit in space, it would implicate a colossal investment and, worst of all, the loss of a complete spacecraft that required years of work. Instead, computer simulations, with high detailed models of the real spacecraft, its equipment and surrounding environment, allow to verify and validate assumptions made on the design process or, even later on the development process, such as to verify and validate the on-board software, the software responsible to command the spacecraft during its lifetime.

The main objective is to model a satellite that, through sensors, gather environment data and is also capable of maintain a Sun-synchronous orbit by operating its actuators. Like in most Earth-observing missions, this type of orbit places the satellite in constant sunlight allowing him to perform imagery or weather analysis in a well-lit Earth whilst producing its electrical power through its solar panels. Among the above objectives, other high level requirements were that an initial date for the simulation could be inserted as a parameter, so that, the simulator could start its orbits calculations from that time forward. The spacecraft and the equipment that it comprises should also allow a custom setup in order to define its specific characteristics.

An open-source modelling software, OpenModelica, was the platform used to model the simulator facilitated by an object-oriented programming approach. This platform was picked due to its plasticity and wide use on several engineering fields.

A previous work has been made on this topic by Emídio Costa (Costa, 2016). One of the aims of this simulator was to continue its work and develop a more dynamic and complete one. His work and ideas were the foundations of an early version of the simulation. Thanks to this baseline, a refined approach was followed so that a complete new simulator could be conceived and produced.

Furthermore, the accuracy and reliability of the simulator as well as the data produced by it, represent a big concern since this is a critical domain and the smallest mistake would have remarkable consequences. To validate the simulator data, other pieces of software, with an assessable extent of accuracy, are used as an evaluation of the simulator performance.

In essence, a satellite simulator, like this one, should provide the means to create unusual scenarios that push the boundaries of technology so that one day, humankind can widen its knowledge of the universe.

2 BACKGROUND

Several fields of science and technology need to be understood in order to implement a satellite simulator. Starting with the environment, planetary orbits can be calculated using Kepler's three laws of planetary motion. To represent the positions of the Sun, Planets, Moon and satellite distinct coordinate systems can be used and the one most pertinent for representing the object in focus must be chosen. In order to create a relation between objects represented in different coordinate systems, coordinate frames transformations are used.

Regarding the satellite attitude, Euler angles and quaternions are used to describe the satellite orientation with respect to a fixed coordinate system.

These concepts are explained in detail on this chapter.

2.1 Previous Work

The proposed simulator had already an initial modelled simulation by Emídio Costa. This early simulator was a perfect starting point. It created an idea of the desired outcome by showing its strong points and its flaws, conveying a concept of what to improve and rebuild.

From this starting point, it was clear that an easy to setup, more complete simulation, with a cleaner interface should be the goal. Although some of this work could be re-used, the way it was designed and implemented didn't offer any benefits to the level of accuracy and completeness ambitioned for the project. Gradually, the initial work was improved and, at one point, it was completely restructured. Further on will be discussed in detail the reasons that lead to this restructuration.

2.2 Sun-synchronous orbit

The sun-synchronous orbit is one of the most used orbits for earth science missions (e.g., ESA's EarthCARE[1] and Sentinel-2[2]). This particular orbit, ranges from 200 to 1680

[1] EarthCARE mission goal is to make global observations of clouds, aerosols and radiation.
[2] Sentinel-2 provides, for example, imagery of vegetation, soil and water cover.

km altitude wise and passes near the poles (Boain, 2004). As the name suggest, this type of orbit is synchronous with the Sun, meaning, the same angle between the orbit plane and the Sun can be maintained without extensive use of propulsion engines in order to make orbit corrections. By combining altitude and inclination, usually higher than 90º degrees, the satellite accomplish to pass over any given point of the Earth's surface at the same local solar time[3].

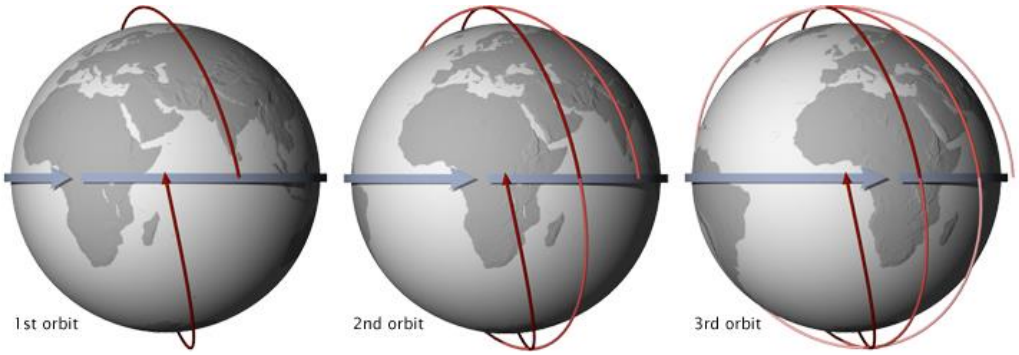


Figure 1. Example of a sun-synchronous orbit crossing the equator at approximately the same local time over three consecutive orbits. (NASA illustration by Robert Simmon).

This fact facilitates the solar panels to receive more solar exposure and grant that the mission specific measuring equipment (the payload) have better conditions to observe Earth's well-lit surface.

2.3 Coordinate Frames

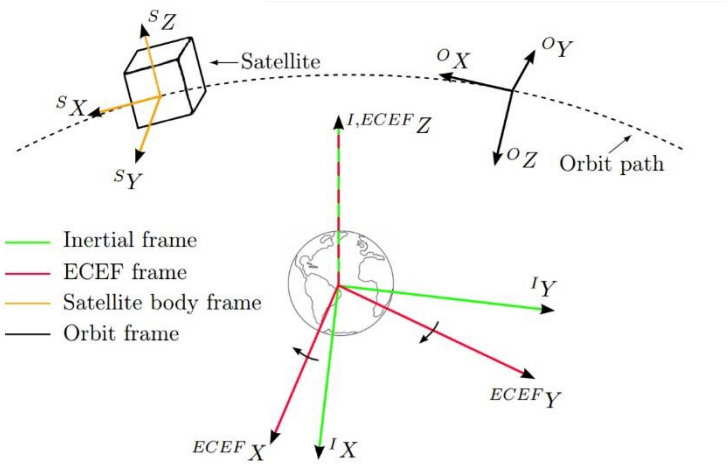
Modelling several objects creates a need of having different coordinate frames. Those are used to represent the object position on the environment. Some are more suited for certain situations and describe more accurately the studied object.

The coordinate frames used to define the position of the modelled objects are the following:

- **Earth-centered Inertial (ECI):** Origin at the center of mass of the Earth, z-axis pointing out of the north-pole, x-y plane coincides with the Earth's equatorial plane. Non accelerated frame.

[3] Solar time is the reckoning of the passage of time based on the Sun's position in the sky.

- **Earth-centered Earth-fixed (ECEF):** Origin at the center of mass of the Earth, z-axis pointing out of the north-pole, x and y rotates at the same rate as Earth ω_{Earth}= 7.2921·10^{-5} rad/s.
- **Orbit frame:** Origin at the Satellite center of mass, z-axis pointing towards the Earth center of mass, perpendicular to the xy plane, x-axis pointing to flight direction and y-axis perpendicular to the x-axis pointing in the direction of the orbit velocity vector.
- **Satellite Body frame:** Origin at the Satellite center of mass, x-axis pointing towards flight direction, z-axis points up through the top of the satellite and y-axis, perpendicular to xz plane, satisfying the right-hand rule.



Picture 1. Illustration of used coordinate frames. Does not show proper proportions (Holst, 2014, p. 6), modified.

Picture 1 illustrates the coordinate frames used to model the problem at hands.

2.4 Coordinate System Transformations

A coordinate system transformation is either used to transform from one Cartesian coordinate system to another or to rotate within one frame. To transform one Cartesian coordinate system into another, three successive rotations are used if their origins are the same or if they both are right-handed or left-handed systems. This gives three rotational matrices:

$$R_x(\phi) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & cos\,\phi & sin\,\phi \\ 0 & -sin\,\phi & cos\,\phi \end{pmatrix},$$

Equation 1. Rotation around the x-axis.

where ϕ is the angle associated to a counterclockwise rotation about the x-axis, also described as a roll in flight dynamics.

$$R_y(\theta) = \begin{pmatrix} \cos\theta & 0 & -\sin\theta \\ 0 & 1 & 0 \\ \sin\theta & 0 & \cos\theta \end{pmatrix},$$

Equation 2. Rotation around the y-axis.

where θ is the angle associated to a counterclockwise rotation about the y-axis, also described as a pitch in flight dynamics.

$$R_z(\psi) = \begin{pmatrix} \cos\psi & \sin\psi & 0 \\ -\sin\psi & \cos\psi & 0 \\ 0 & 0 & 1 \end{pmatrix},$$

Equation 3. Rotation around the z-axis.

where ψ is the angle associated to a counterclockwise rotation about the z-axis, also described as a yaw in flight dynamics.

2.5 Euler angles

Euler angles are frequently used to describe the rotations of a rigid body system. Since the satellite can be approximated to a rigid body, it is possible to describe the attitude and rotation using Euler angles. Additionally, Tait-Bryan angles are a convention used in flight dynamics and aerospace where each of the three Euler angles defines a rotation around one of the three Cartesian axis. By means of one of the six Tait-Bryan's sequences convention, an x-y-z (extrinsic rotations) sequence is used herein forward.

$$\Theta = \begin{pmatrix} \phi \\ \theta \\ \psi \end{pmatrix}$$

Equation 4. Euler angles.

For the satellite rigid body:

- $\boldsymbol{\phi}$ is roll angle, the rotation around x-axis.
- $\boldsymbol{\theta}$ is pitch angle, the rotation around the y-axis.
- $\boldsymbol{\psi}$ is yaw angle, the rotation around the z-axis.

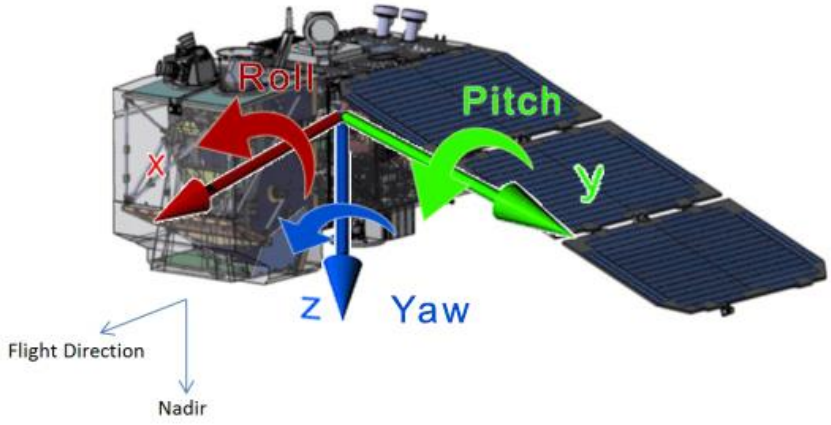


Figure 2. Illustration of the principal axes of a satellite.

Euler angles are an intuitive representation of the satellite attitude in 3D space. On the other hand, using Euler angles to describe the satellite attitude might result in singularities. Those singularities occur when the orientation cannot be uniquely represented by Euler Angles.

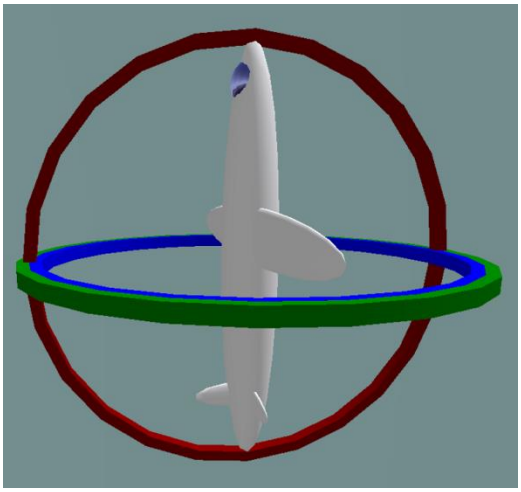


Figure 3. Illustration of Gimbal lock condition.

This event is called Gimbal lock and it occurs when two rotational axis point in the same direction causing the loss of one degree of freedom.

As represented in Figure 3, two out of three gimbals are in the same plane causing the loss of one degree of freedom. Eventually, after several rotations, while using Euler angles to describe the object orientation, this event will occur and so, the usage of Euler angles is avoided.

2.6 Quaternions

To avoid singularities caused by the use of Euler angles, quaternions symbolize a singularity-free representation of the attitude with only four parameters. The quaternion consists of three imaginary parts and a single real, it can be expressed as (Wertz, 1994):

$$\mathbf{q} = \mathbf{i}q_1 + \mathbf{j}q_2 + \mathbf{k}q_3 + q_4$$

Equation 5. Quaternion representation.

or with the imaginary part contracted:

$$\mathbf{q} = \overline{\mathbf{q}} + q4$$

Equation 6. Quaternion with the imaginary part contracted.

Euler angles are only used to setup the starting parameters of the satellite because they are easier to read and help to visualize the satellite attitude. From then on, they are converted to quaternions and conversions between quaternions and Euler angles are only used for user data display purposes. This is an attempt to prevent the above mentioned singularities on the satellite attitude representation.

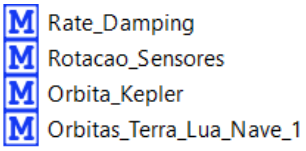
3 MODEL ARCHITECTURE

The performance, quality and scalability of the simulation depends on the architecture implemented, therefore this is a key aspect that should be refined and thought through.

In addition, a good architecture reflects that future work has been considered, for example, if one wishes to add more modules and/or increase the accuracy of the simulation, the architecture should be compliant.

3.1 Previous Architecture

The architecture of the previous simulation/work was quite limited and conglomerated. On the mentioned architecture there are four models (Picture 2): two models contain the logic of the orbits and the satellite and, two others, with the purpose of invoking and define the parameters for the first two. The logical models are: the orbital model, where one object orbit another, and the other include IMU and CESS sensors that coexist on the same model.



Picture 2. Models flat architecture when loaded on the simulation environment.

There is no connection between them, no information flow and they run on different simulation times. This was the first issue that arisen. For instance, from a geocentric perspective, the Sun's position changes over time and if there is no information flow, the CESS could not give a correct and realistic reading. On (Costa, 2016), CESS and IMU are only simulated for 4 seconds and simplifications were made so that the simulation could work, such as the satellite is crossing Earth's ecliptic plane, its orbital coordinate system referential Z-axis points to the Sun and the satellite body had a random initial orientation.

Also, regarding the orbits, it was not possible to set an initial date and time for the simulation to begin with, nor the orbits were very accurate.

Another notorious lack on this architecture was the fact that if one wishes to set different parameters of the simulation models, code interaction was required. This requires knowledge on OpenModelica language and usually requires time and effort to understand the code.

It was clear that, on the long run, those approximations and limitations were not practical for a continuous and easy use of the simulation.

3.2 Architecture

A more suited object-oriented architecture was developed. It was designed to be compliant with a modular design where components like sensors or actuators could be added to the existing system.

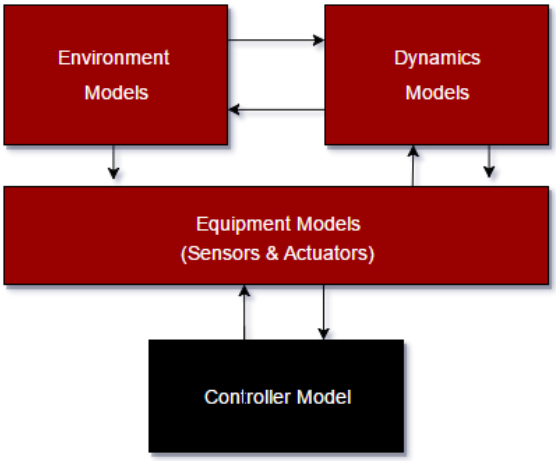


Figure 4. High-level view of the system architecture.

Figure 4 shows a high-level view of the implemented architecture, with exception to the controller model. By modelling the loop of the components, excluding the controller model, one can create a closed-loop system for testing a control system. This type of architecture also consents to change a software module by a real equipment creating an environment suitable for testing real components on simulated situations.

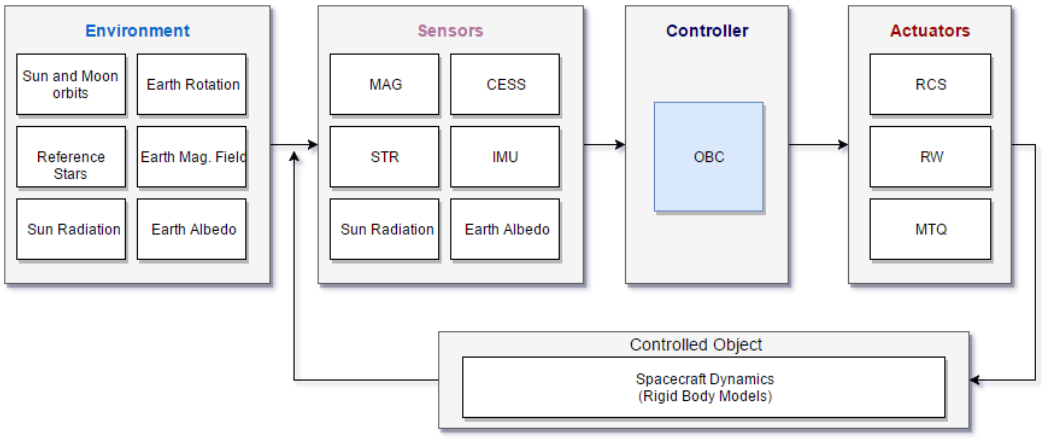
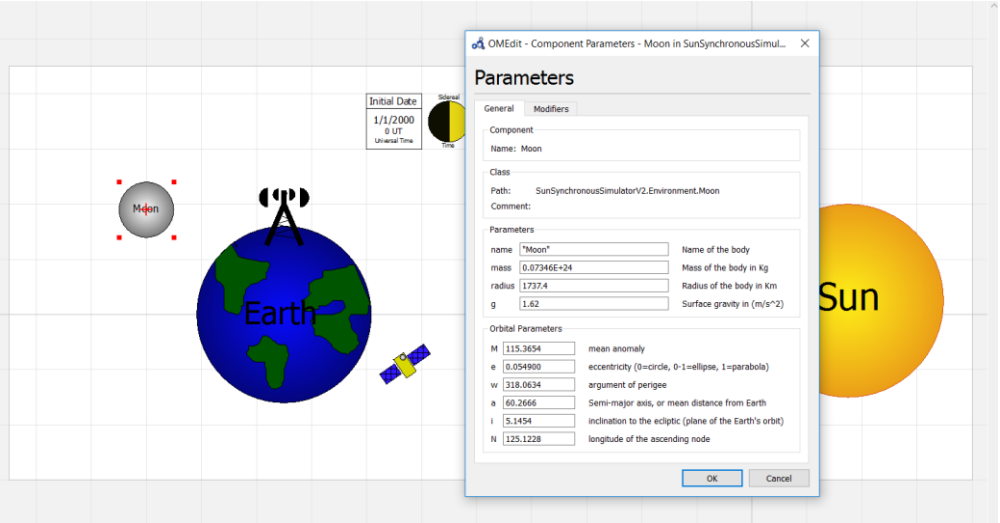


Figure 5. Components from a Sun-synchronous satellite control architecture

Figure 5 shows a more detailed and low-level architecture, close to what has been implement. By listing the larger modules, one has:

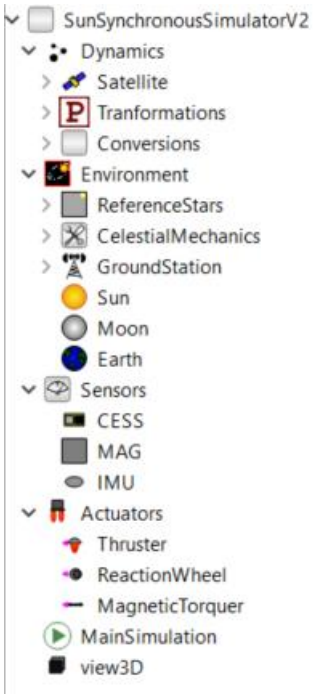
- **Environment** package where non-satellite variables are modelled. Variables like Sun (since the simulation has a geocentric perspective) and Moon positions, reference stars
- **Controller** (not implemented) contains the on board computer (OBC) and it is responsible for control commands, for instance, triggering the actuators or convert analog or digital signals into engineering data ready for use and storing.
- **Controlled Object** refers to the satellite dynamic and kinematic models. Since its attitude is influenced by the environment and actuators, rigid body motion is calculated inside this package and is discussed further on chapter 5. for navigation and Earth rotation, albedo and magnetic field are all calculated inside this package.
- **Sensors** package clusters the satellite sensors. Such sensors will be discussed in detail in section 6.2. Some of these sensors also gather data from the Environment and from the Satellite dynamics as it is visible on Figure 5.
- **Actuators** store models such as reaction control system (i.e., thrusters), reaction wheels and magnetic torquers (also known as torque rods) and are explained in section 6.3. They are commanded by the controller or a dummy controller only to test their actuation.



Picture 3. View of simulation environment with an example object parameterization.

Using OpenModelica capabilities, easy and intuitive parameterization of the models variables has been made possible. In contrast to the previous work, one can now set the parameters without having knowledge or interaction with the source code of the models. This is a cleaner and more natural way of interacting with the simulation.

In Picture 3, is perceptible that objects parameters can be easily set. This feature is present on two main models, the main model, where high-level objects are found, and the satellite model, where sensors, actuators and the dynamic and kinematic are present.



Picture 4. View of the models and its architecture on the simulation environment

The architecture above explained is organized as shown in Picture 4. When compared to the previous work (Picture 2), it's noticeable that this one is more structured and uses the object oriented capabilities of OpenModelica.

In addition to the models developed inside the software tool, external C code was used to improve simulation performance.

4 ORBITAL MODEL

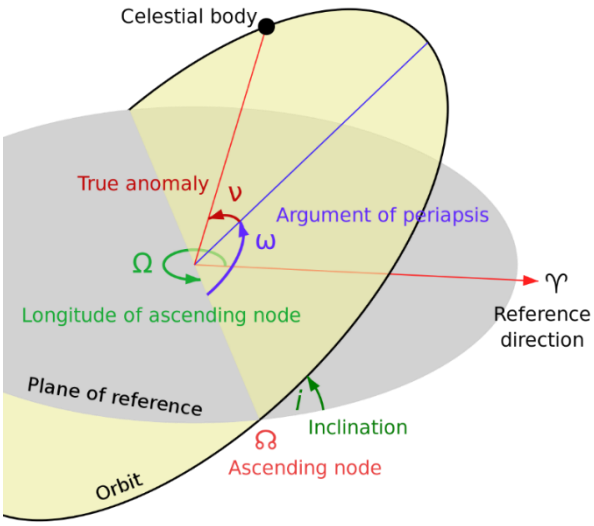
Knowing the position of Earth, Moon and Sun with respect to the spacecraft is essential for this simulator. Spacecraft sensors depend on those positions as inputs so that they can generate data in order to obtain the spacecraft attitude.

Additionally, given a certain UTC Gregorian date it should be possible for the orbital models to calculate the planetary positions from that instance in time forward, in order to allow the user to replicate certain time events.

To compute the Moon and Sun positions, their orbital elements must be known and must change over a defined time scale. As the Earth will be on the center of the "simulated environment", its position is consider to have the following coordinates $Earth_{(x,y,z)} = \{0,0,0\}$. The satellite has its own orbital parameters as well and follows the same computational principle as Moon and Sun.

4.1 Orbital Elements

Orbital elements are used to uniquely describe a specific orbit. The primary elements are the six Keplerian elements, after Johannes Kepler and his three laws of planetary motion, describing the motion of planets around the Sun.



Picture 5. Diagram illustration of the Keplerian elements (Wikipedia).

The shape and size of an orbit can be defined by using two main elements:

- **Eccentricity** (*e*) – a measure of how much the orbit deviates from being circular. With value 0 (zero) shaping a circular orbit, more than 0 (zero) and less than 1 an elliptic orbit, 1 defining a parabolic orbit and lastly, more than 1 describing a hyperbolic orbit.
- **Semi-major axis** (*a*) – considering the ellipse case, its is longest diameter or the sum of periapsis and apoapsis distances divided by two.

Other elements that help defining an orbit are:

- **Inclination** (i) – defines the vertical tilt with respect to the equator of the object being orbited.
- **Longitude of ascending node** (Ω) – is the angle formed from the Vernal Equinox (♈) and the ascending node.
- **Argument of periapsis** (ω) – defines the orientation of the ellipse in the orbital plane, as an angle from the ascending node to the periapsis.
- **True anomaly** (v) – is the angular distance of a point in an orbit past the point of periapsis in degrees.

Other related orbital elements, such as mean and eccentric anomaly, were used but only the most relevant are herein introduced.

4.2 Time scale

In order to compute the planets and satellite positions on a given date, a time scale must be implemented. This time scale is counted in days and it will be included on the formulae of the orbits, since orbital elements change through time.

Day 0.0 occurs at 1 of January of 2000 at 0.0 UT (Universal Time) and the formula for calculating the days since day 0 can be computed as follows:

$$d = 367 * year - 7 * \frac{year + \frac{(month + 9)}{12}}{4} + \frac{275 * month}{9} + date - 730530$$

Equation 7. Date number formula (Schlyter, 2016).

It is to note that every division is an integer division. Adding the UT on the format hours plus decimals is as follows (this one is a floating-point division):

$$d = d + UT/24.0$$

Equation 8. Day number formula with UT (Schlyter, 2016)

This formula is then integrated on the computation of the orbits so that certain orbital parameters can be adjusted to that date.

4.3 Computing the orbits

Orbit computation's method implemented on the simulator was created by Paul Schlyter (Schlyter, 2016) with intent to be a simplified algorithm, and yet, with a fairly good accuracy of about 1-2 arc minutes.

The key aspect that allows the introduction of a date and time on the simulation setup, is to calculate the orbital elements of every planet with respect to the "day number" calculated by Equation 8, letting that a continuous simulation, starting on a pre-defined date and time, can advance throughout the simulation.

The simulation environment frame is considered to be geocentric (Earth centered) instead of heliocentric because is fairly more easy and suited to implement this way, especially considering that the goal is to simulate a satellite orbiting the Earth on a sun-synchronous orbit and most of the relations are expressed between the satellite and the Earth. Therefore, instead of computing the position of Earth orbiting the Sun, it is computed the position of the Sun with respect to Earth.

4.4 Satellite orbital parameters

To set up a Sun-synchronous satellite orbital parameters, an already launched satellite was used as model, making it easier to validate the orbit data. This satellite picked was ESA's Sentinel 2, launched on 23 June 2015.

Table 1. Sentinel 2A orbital data (Sentinel 2A Satellite details).

Semi-major axis	7167 km
Inclination	98.5638 degrees
Right ascension of the ascending node	52.2598 degrees
Eccentricity (decimal point assumed)	0001113
Argument of perigee	78.4565 degrees
Mean anomaly	281.6749 degrees
Period	100.6 minutes
Orbital medium velocity	7,463 km/s

The orbital parameters found on Table 1 were up-to-date on 6 December 2016 and will remain the same herein forward as the reference data used to configure the satellite's orbit.

The correlation between the change of Sentinel 2A orbital elements with the passage of time was not calculated, and revealed to be a laborious and demanding task since the satellite can use its actuators to correct its orbit. Consequently, as this change is not accounted for, the orbit gets out of Sun-synchronous as the simulation time passes.

5 SATELLITE MODEL

The satellite is assumed to be a rigid body with motion. To describe the rigid satellite rotation, dynamic and kinematic equations modelled are herein presented.

When orbiting Earth, disturbances act upon the satellite. Those forces are, but not limited to:

- Gravitational gradient
- Aerodynamic drag
- Environmental radiation torques
- Magnetic torques

These disturbances have not been implemented on the simulator as they account for small torques acting on the satellite.

5.1 Satellite Dynamics and Kinematics

The kinematic equation, below presented, describes the motion of the rigid satellite as a rotation from an orientation at a time instance to another shortly after. This equation in quaternions is given by (Wertz, 1994):

$$\dot{q} = \frac{1}{2}\mathrm{S}(\Omega)\mathrm{q}$$

Equation 9. Kinematic equation (Simplified).

Where $q = (\varepsilon\ \eta)^T$ is the quaternion attitude, ε is the vector part of the quaternion, η is the scalar and T is the transpose matrix. $\mathrm{S}(\Omega)$ is the skew-symmetric representation of the angular velocity defined by:

$$S(\Omega) = \begin{pmatrix} 0 & \omega_z & -\omega_y & \omega_x \\ -\omega_z & 0 & \omega_x & \omega_y \\ \omega_y & -\omega_x & 0 & \omega_z \\ -\omega_x & -\omega_y & -\omega_z & 0 \end{pmatrix}$$

Equation 10. Skew symmetric representation of angular velocity.

Where $\omega = (\omega_1\ \omega_2\ \omega_3)^T$ is the angular velocity measured in the body fixed frame.

6 ATTITUDE MODEL

The satellite attitude represents its orientation in space. It is a crucial assessment, especially, when carrying highly sensitive payloads and precise measurement equipment.

The sensors below described provide, to AOCS, the necessary data for this module to create an estimation of the satellite attitude.

6.1 Control System Overview

Attitude Orbit Control System (AOCS) is the onboard system responsible for controlling the orbit and satellite attitude. Even though this module was not implemented on the simulator, a brief description and explanation on his operating mode is presented.

AOCS is vital when speaking of satellite attitude. It provides the following major functions:

- Sensor data acquisition and pre-processing
- Measurement processing
- Attitude estimation
- Actuator commanding

The AOCS has several function modes[4], such as:

- Standby Mode (SBM)
- Initial Acquisition Mode (IAM)
- Safe Mode (SFM)
- Normal Mode (NOM)
- Orbit Control Mode (OCM)

Each mode serves its function on each step of the mission. For example, Initial Acquisition Mode is selected when the satellite is separated from the launcher. Its main purpose is to stabilize the satellite after being released from the launcher onto an uncontrolled spinning attitude.

[4] AOCS modes may vary depending on the mission and system architecture. The modes presented are the most commonly used.

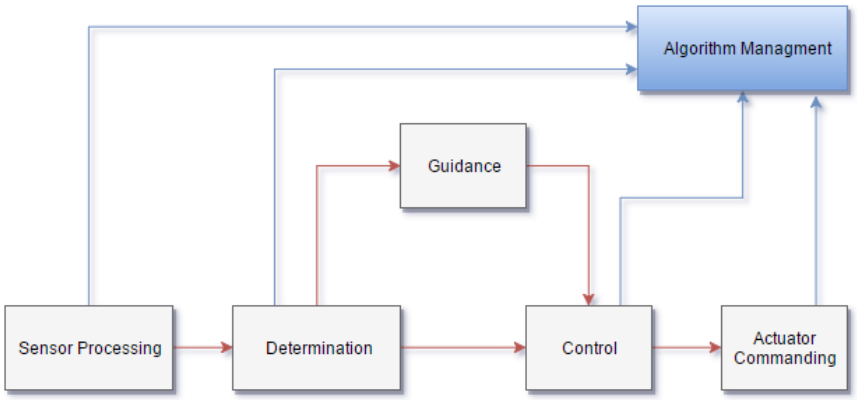


Figure 6. AOCS high level architecture.

After measuring and process the data collection from the sensors, AOCS also commands the actuators to employ torques needed in order to re-orient the satellite to a desired attitude. This is visible on

Figure 6, which illustrates the high level architecture and how the data flows.

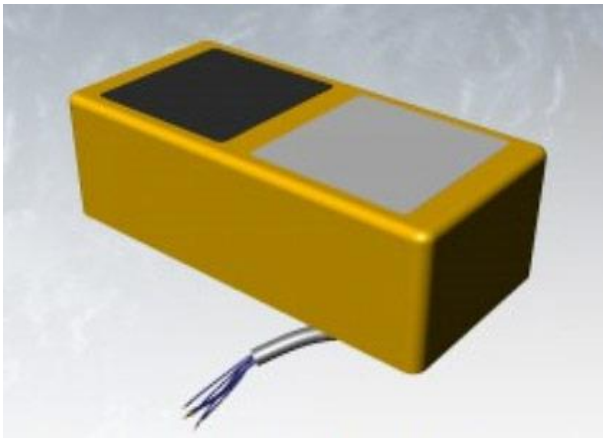
6.2 Sensors

A set of sensors present on the satellite gather information of the surrounding environment and on the satellite itself. They allow the satellite OBC to process and analysis their data to produce the satellite's attitude determination.

6.2.1 Coarse Earth Sun Sensors (CESS)

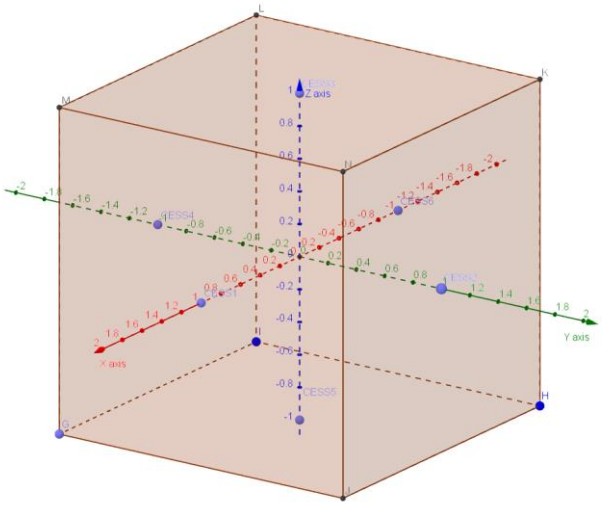
The Coarse Earth Sun Sensor, or simply CESS, is an instrument that provides an estimation of Earth and Sun positions on the satellite reference frame. With a robust and simple design, each CESS head is composed by two active sensor areas that heat differently when exposed to solar and infrared emissions (

Picture 6). Thermistors on the interior give temperature readings on each active area.



Picture 6. Coarse Earth Sun Sensors (SpaceTech GmbH).

Six CESS heads are normally disposed on the satellite positive and negative axes, identical to Picture 7 CESS disposition, in order to compose Earth and Sun direction vectors. The output of each CESS head is given in Ohms (Ω).

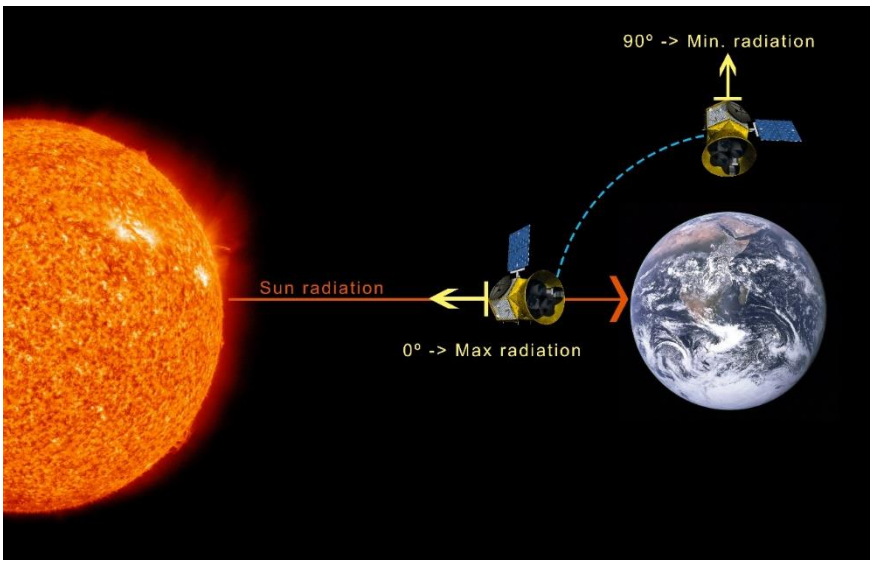


Picture 7. Alignment of the six CESS heads on the satellite body (Light blue dots).

To recreate the interaction between the CESS heads and the incident light, in order to simulate their function, vectors were used to find the relations of $\overrightarrow{Sat\ Earth}$ and $\overrightarrow{Sat\ Sun}$ with respect to each CESS normal vector. From this relation, an analogy was made, equivalent to the operation of electric solar panels, where the angle of incident light is correlated with the electric power output of the panel (Kurjakov, Kurjakov, Mišković, & Carić, 2012), when the angle made by the incident light and the solar panel is closer to 90 degrees, the generated power will be at its maximum. The modification to this

approach is that, in this case, the angle of incident light is compared to the normal vector of the sensor, making a 0 degree angle corresponding to maximum resistance and a 90 degrees angle, corresponding to minimum resistance.

Picture 8 illustrate the working principle and the analogy made in order to model the CESS sensors. In the illustrated example, it is presented the relation between Sun's incident light and the satellite's CESS but, the same principle is applied to model Earth's albedo radiation readings.



Picture 8. CESS operation principle.

Both thermistors have different resistance output maximums, one for the incident sun radiation and another for incident earth albedo. The linear relation between the angle and the output resistance of CESS head is illustrated on

Figure 7, this example represents the result of the sun incident radiation. The same logic but, with different resistance values is used for incident Earth albedo radiation.

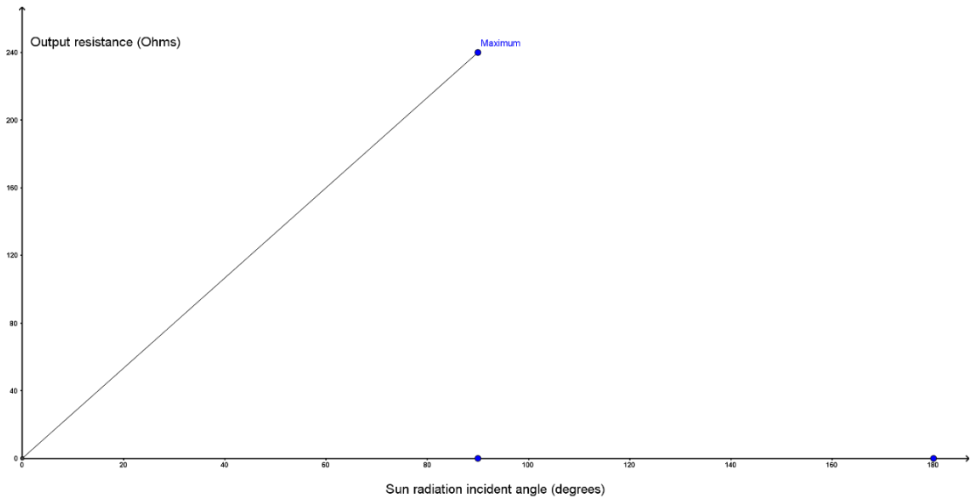


Figure 7. Correlation between incident Sun radiation angle and output resistance.

CESS head positions can be set as well as the maximum resistance values for each thermistor, facilitating a quick displacement of the instruments along the spacecraft and combinations of thermistors configuration.

The reflection coefficient of Earth is not equal throughout the planet, and so, an attempt to replicate this characteristic, although with less precision than reality, was performed.

An approximation was made using an annual mean albedo value that corresponds to a range of latitudes. Accomplishing eighteen different values for Earth's albedo (data extracted from (Jet Propulsion Laboratory, 1995)).

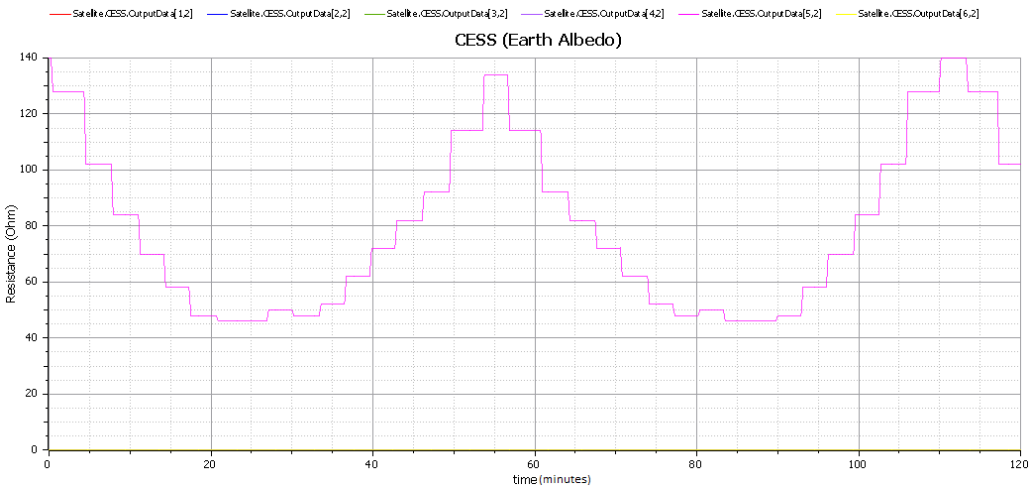


Figure 8. CESS incident albedo plot.

The plot on Figure 8 illustrate how the thermistor on CESS number five, nadir-pointing[5], is reacting to the incident albedo during a period of about one orbit. The output produced draws a stair step graph as a result of the different albedo values at different latitudes. The satellite dynamic was taken into account but the satellite orientation was specified to maintain a nadir-pointing orientation.

6.2.2 Magnetometer (MAG)

A magnetometer is an instrument that measures the flux density of the magnetic field it is placed in. The magnetometer will measure the geomagnetic intensity and direction surrounding the satellite.

According to (SpaceMath@Nasa), the strength of the magnetic field along the axis of a Cartesian coordinate system x, y and z can be calculated as follows:

$$B_x = \frac{3 \cdot x \cdot z \cdot M}{r^5},\ B_y = \frac{3 \cdot y \cdot z \cdot M}{r^5},\ B_z = \frac{(3 \cdot z^2 - r^2)\, M}{r^5}$$

Equation 11. Formulae to compute magnetic strength along x, y and z axis respectively.

Where, *x*, *y* and *z* represent the coordinates of a point in space in multiples of the radius of Earth, where 1.0 Re = 6,378 km, *r* is the distance from (*x*, *y* and *z*) to the center of Earth and *M* is a constant equal to 31,000 nT Re^3. The output unit is nanoTeslas (nT).

6.2.3 Start Tracker (STR)

A star tracker (Picture 9) is a light sensitive instrument, that determines, with high precision (higher than the CESS), the attitude of the satellite by observing remote starts. One can do an analogy with the method used by sailors to navigate the oceans throughout many centuries. There are, depending on the complexity of the mission, at least 58 “selected stars”, found on the *Nautical Almanac*[6], considered on an onboard star

[5] Nadir pointing is the direction pointing directly below a particular location. In this case the satellite’s CESS number five.

[6] Nautical Almanac (Book) contains the positions, brightness and other observable characteristics of celestial bodies. It is computed and updated by the U.S Naval Observatory and Her Majesty’s Almanac Office in annual publications. For example (Nautical Almanac, 2016).

catalog from where the processing unit on the star tracker compares the images taken from its camera.

Picture 9. Hydra star-tracker currently flying on the ESA's Sentinel 3A satellite (ESA).

By assessing the rate of change of the star's positions relative to the satellite, it can determine the attitude, angular and linear velocity.

6.2.4 Inertial Measurement Unit (IMU)

An inertial measurement unit as the name states, is an instrument that measures the components of angular and linear velocities on each axis of the satellite. To do so, it uses gyroscopes and accelerometers.

On the simulator case, the IMU is outputting the differences between the values of one timestamp and the previous one. The values analyzed are: angular and linear velocities and the rate of change of Euler angles.

As the IMU is usually not aligned with the satellite body frame, its data have to be rotated from IMU referential to the satellite body frame.

6.3 Actuators

Actuators are fundamental pieces of equipment that make possible to correct the satellite attitude. Without them, the satellite would just be an expensive and uncontrolled object on space. The most commonly used actuators are described below.

6.3.1 Thrusters (RCS)

The reaction control system (RCS) makes use of a set of thrusters to make the attitude control.

A thruster is a propulsion equipment meant to made adjustments on the satellite attitude, to maneuver the satellite and perform orbit corrections. It uses fuel, like hydrazine, to create a thrust.

Since thrusters can induce the most considerable torque actuation of all the actuators, the amount of fuel remaining on the tanks can be a constraint to the lifespan of the mission. As the fuel runs out, the mass of the satellite decreases increasing the torque forces resulting from a thruster actuation. When the fuel runs completely out, the mission goal can be seriously compromised if yet not achieved, since the satellite is not able to adjust its course no longer.

6.3.2 Reaction Wheels (RW)

A reaction wheel (RW) is composed by an electric motor that spins a freely rotating wheel. As the reaction wheel changes its rate of rotation in one direction it causes the satellite to rotate in the opposite direction.

This event occurs with conformity of Sir Isaac Newton’s Third Law of Motion that states: for every action, there is an equal and opposite re-action.

Usually, a combination of three reaction wheels is mounted on the satellite with different orientations. More reaction wheels can be added taking redundancy into account or, when combined with magnetic torquers fewer reaction wheels can be used.

6.3.3 Magnetic Torquers (MTQ)

Magnetic torquers are electrical devices built from electromagnetic coils. When a current is applied to the magnetic torquer, a magnetic dipole will be created along the main axis of the unit. This artificial magnetic field makes the satellite to line up with the magnetic field vector.

The actual torque produced by magnetic torquers is usually very small and can be determined by:

$$\tau = \mu \times \mathrm{B}$$

Equation 12. Torque provided by a Magnetic torque

Where τ is the torque on the satellite, B is the ambient magnetic field, and μ is the magnetic field of the satellite.

6.4 Visibility to ground stations

From the moment that the satellite is in orbit, it gets almost impracticable to have physical interaction with it and its instruments. So, telemetry commands issued from the control center should reach the satellite. Communication is a key factor in order to download scientific data and send instructions to the satellite, from an attitude adjustment to a hardware malfunction, they all need to be commanded from the control center.

This communication window or visibility between ground control and satellite is tested by measuring the distance between the satellite and a ground station. It's simulated by setting a set of coordinates for a ground station and, by using a formula that calculates the great-circles between two points, that is, the shortest distance over the earth's surface. The formula is called Haversine formula and is given by:

$$a = sin^2\left(\frac{\Delta\varphi}{2}\right) + \cos\varphi_1 \cdot \cos\varphi_2 \cdot sin^2\left(\frac{\Delta\lambda}{2}\right),$$

$$c = 2 \cdot atan^2\left(\sqrt{a}, \sqrt{(1-a)}\right),$$

$$d = R \cdot c$$

Equation 13. Haversine formula.

where, φ is latitude, λ is longitude and R is Earth's radius. [7]

The altitude is neglected and it's assumed that the satellite has a projection of its position on Earth's surface. This is due to the fact that it is this module's purpose to check for visibility windows, so a coarse estimation and assumption that the satellite antenna is nadir-pointing and extrinsic interferences don't apply, such as climate conditions. Its then assumed that, the great-circle distance between them is enough to simulate the ground station visibility of the satellite.

[7] $atan^2$ is the arctangent function with two parameters found in a variety of computer languages.

7 SIMULATION RESULTS AND DISCUSSION

The results acquired from the simulation will be present on this chapter. Discussion on the findings and results will also be included, along the validation of the data attained. For simplicity, the results and discussion are distributed on three major clusters with resemblance to the architecture implemented and discussed on Model architecture.

Most results have been validated using an open-source software, developed by a team of NASA, private industry, public and private contributors, called General Mission Analysis Tool (GMAT). Due to its extent and excellence of features, its high quality user documentation, recognition and accuracy (Hughes, Qureshi, Cooley, Parker, & Grubb, 2014), it is one of the best solutions available to validate orbits and satellite related data.

Other open-source tools, like GeoGebra (GeoGebra, 2016), or online tools, like Wolfram Alpha (Wolfram|Alpha, 2016), allow to verify mathematics and physics concepts implemented on the simulation.

7.1 Environment Results

Most environmental data relates to orbits and the positions of the Sun and Moon with respect to Earth center of mass. Other environment data is used by the sensors on the satellite and become clear when presented and discussed conjointly with those sensors.

7.1.1 Sun position

Starting by the Sun position, the simulator has been configured to simulate one Earth year (365 days) of data. The sun position is vital to gather data that is going to be used by the CESS in order to determine the satellite attitude.

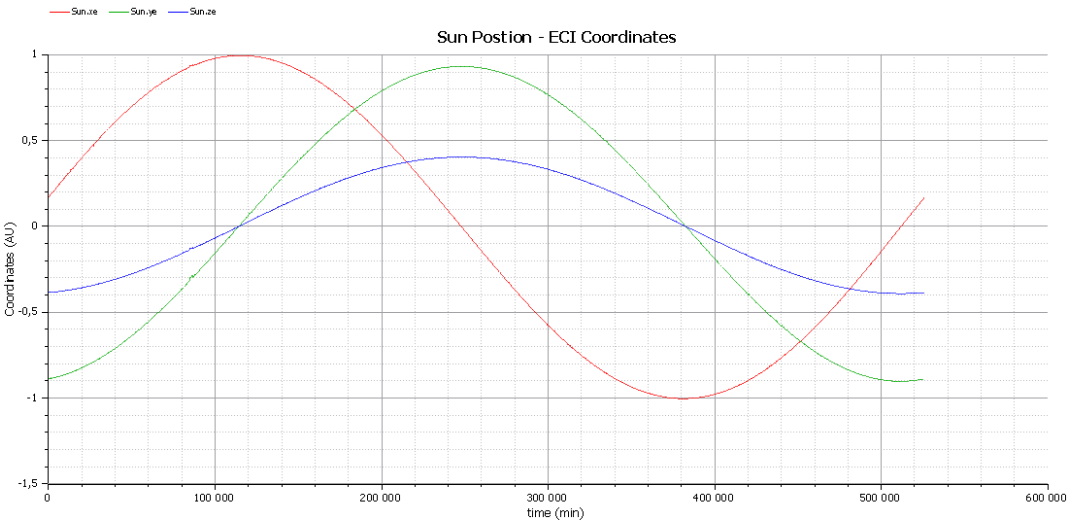


Figure 9. Sun positon in ECI coordinates (AU) over one year of simulation.

Plotting the position of the Sun in Earth-centered inertial coordinates over a course of one Earth year, yields the results on Figure 9. For validation purposes, Figure 10 presents the data obtained on GMAT for the same query. The coordinates unit is in km.

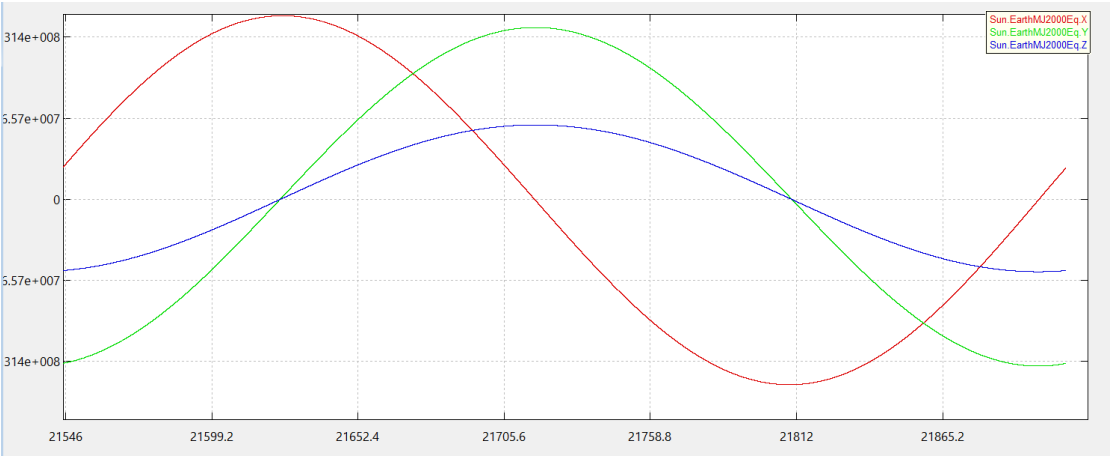


Figure 10. Sun coordinates (km), over one year of simulation, obtained from GMAT.

Over the course of one Earth year, there are two particular timely events where: one, Earth and Sun are at their closest distance and two, where the two bodies are farther from each other. From a heliocentric perspective, these two events are termed perihelion and aphelion, respectively.

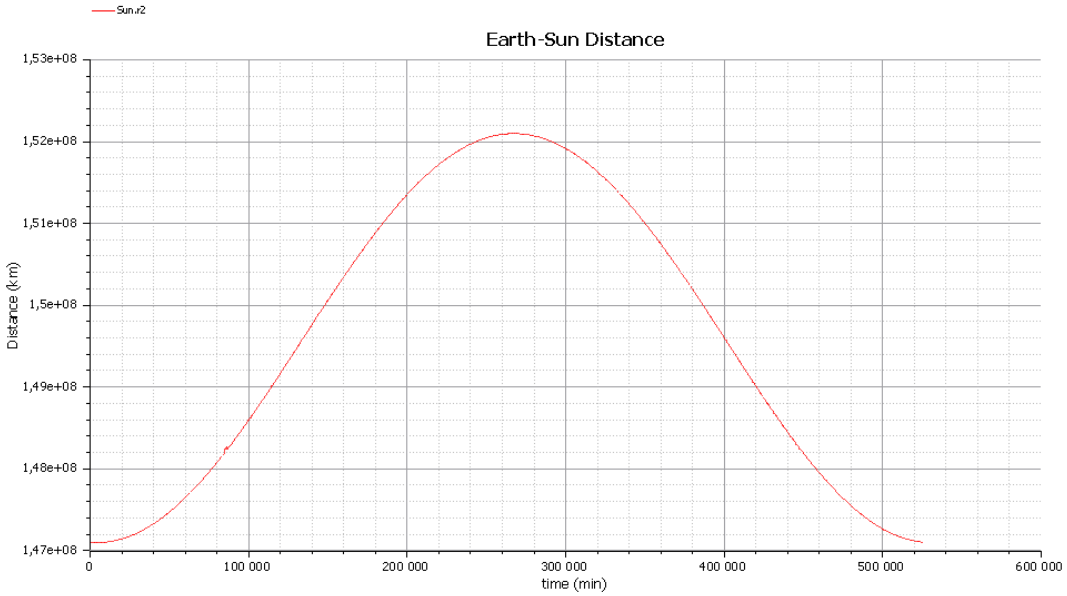


Figure 11. Earth-Sun distance over the course of one year (starting on the 1st of January).

According to (Williams, Earth Fact Sheet, 2016), on perihelion the distance between the two bodies is $147.09\ (10^6\ km)$ and on aphelion is $152.10\ (10^6\ km)$ (these values change slightly over time). When examining the plot on Figure 11, that shows the distance between Earth and the Sun, it yields a minimum value (perihelion) of $147.098\ (10^6\ km)$ and a maximum value of $152.097\ (10^6\ km)$. The data obtained is very accurate and reflects a valid simulation.

7.1.2 Moon position

The same procedure has been applied to the Moon orbiting Earth. The moon has an orbital period of about 27.3217 Earth days (Williams, Moon Fact Sheet, 2016).

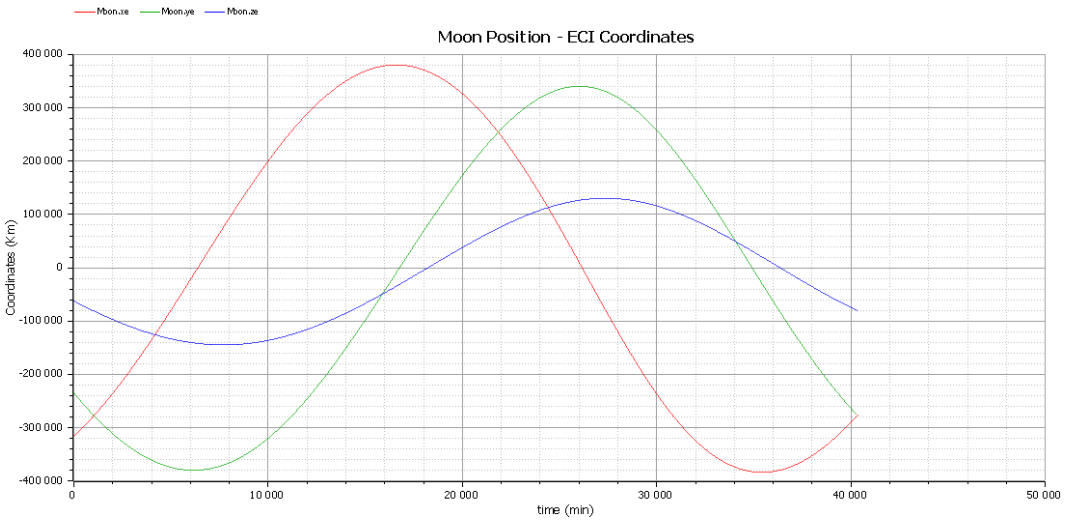


Figure 12. Plot of Moon coordinates over 28 days.

As before, in order to corroborate data, Figure 12 has been compared with Figure 13 (obtained from GMAT) to check its credence.

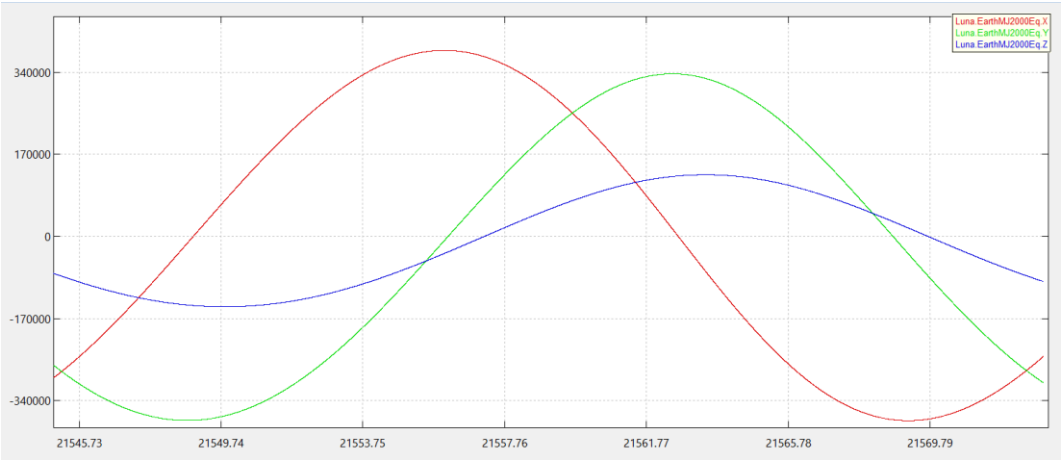


Figure 13. Moon coordinates over a 28 days span, obtained from GMAT.

Moon also have a closer and farther point from Earth over one Earth year span. On this case, they are called perigee and apogee, respectively. According to (Williams, Moon Fact Sheet, 2016), on perigee the distance is 363,300 km and on apogee 405,500 km. The data gathered from the simulation yields that Moon's perigee is 363,278 km and apogee is 405,482 km.

The Moon's orbit is severely perturbed. Although some of these perturbations have been accounted for, to a small extent of exhaustion, there are many other perturbations not

accounted on this work, producing values that may diverge from reality. For the simulation purpose this is not an issue and didn't required a higher level of accuracy.

7.1.3 Visibility to Ground Stations

To verify the ground station visibility windows, a simulated ground station was created in Turku, Finland (60.4518° N, 22.2666° E). The position of the satellite in ECEF coordinates is then converted to LLA and plotted externally on an Earth map. ECEF coordinate system accounts for the rotation angle of the z-axis rate (angular speed of Earth's rotation) of about $\omega_{ie} \approx 72.9211514\,\mu rad/s$, simulation UTC time and Greenwhich mean sidereal time. Using Haversine formula, as previously explained, the distance between the ground station and satellite is then calculated. The final result is presented on Figure 14 below:

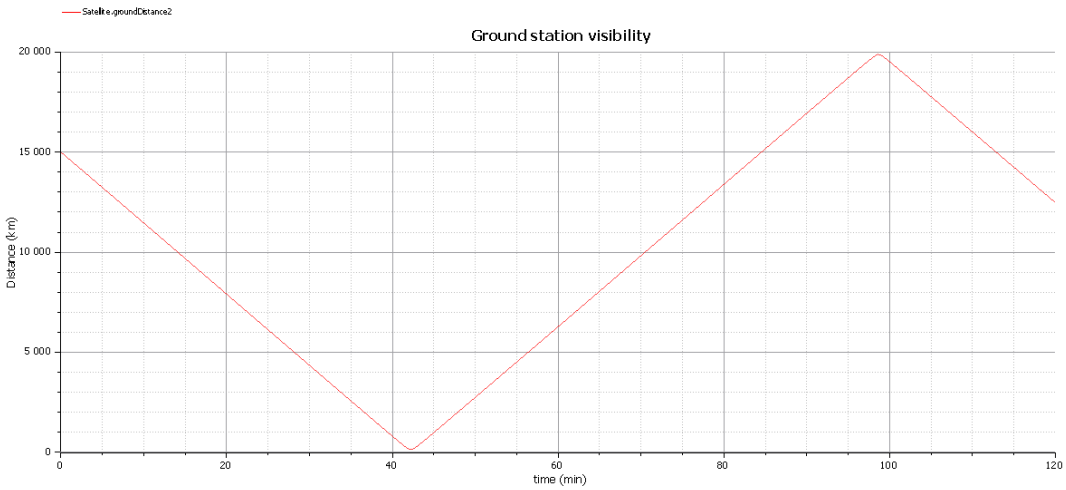
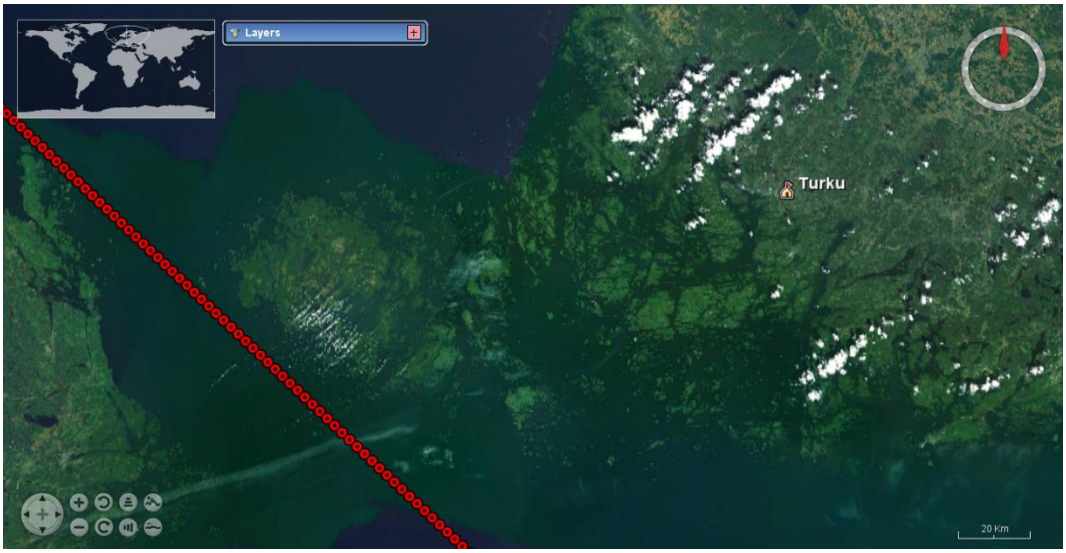


Figure 14. Plot of distance between ground station and satellite projection on Earth surface.

Analyzing the plot, the closest distance, between the simulated ground station and the satellite projection, is about 132 km. When measuring the distance from the map plot of the satellite latitude and longitude over 120 min (Picture 10) and the ground station location, the difference from the data obtained from the simulation and the external tool is hardly noticeable.



Picture 10. Map plot of ground station (Turku, Finland) and satellite projection (red markers).

There is a deviation from the latitude that the satellite is reaching, to the LLA data produced by the simulation. The latitude does not reach the poles as it was supposed to be. This is most probably due to inaccurate calculation of ECI coordinates, as presented on the next section. Also, three or more iterations of Bowring's method could be used when converting between Cartesian and geodetic coordinates in order to increase the data accuracy.

7.2 Satellite Position and Dynamic Results

It is possible to obtain the satellite position, on its orbit, in different coordinated systems, such as: ECI, ECEF and LLA. It is also possible to attain the satellite body frame orientation for dynamic and sensors operation.

7.2.1 Satellite Position

The ECI coordinates are the most used to describe the satellite position on its orbit. Here, the satellite dynamic is discarded. In order to validate the correctness of the position calculated from the orbital elements presented on section 4.4, data was also collected from GMAT.

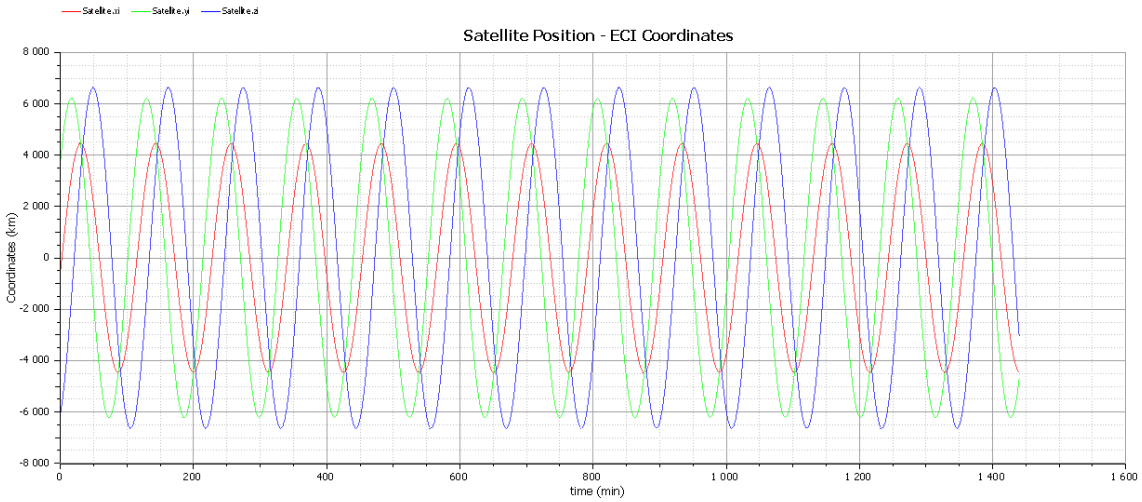


Figure 15. Satellite position in ECI coordinates.

On Figure 15, the simulated position of the satellite over one Earth day is presented. To create a contrast, the GMAT data collected is presented bellow in Figure 16:

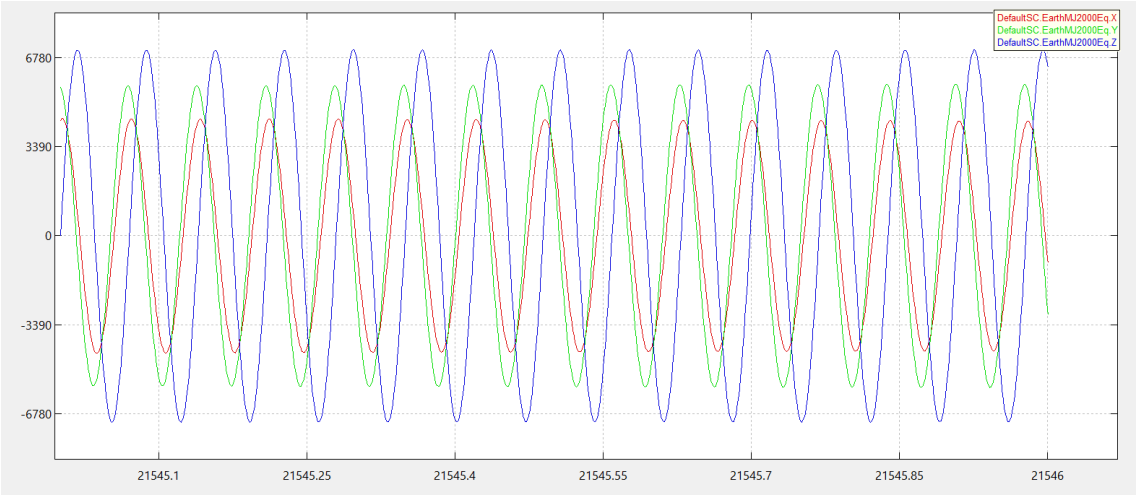


Figure 16. Satellite position in ECI coordinates, extracted from GMAT.

It is noticeable that the *y* axis coordinates simulated deviate about 500 km from the data obtained in GMAT. This is possibly the reason why the conversion to ECEF and then to LLA does not produce an accurate latitude.

This problem is caused to the fact that the simulated orbital elements of the satellite are not being updated accordingly to the “day number” and further calculations on how to correctly change these parameters were necessary.

7.2.2 Satellite Dynamics

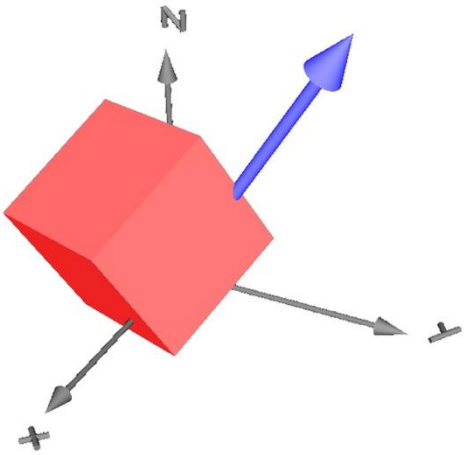
The satellite dynamics presented a more complex problem to model. The reason to this is due to the fact that the last instance of the satellite's orientation is needed to compute the new orientation, which reflects a torque applied on the satellite. This was not easy to model under OpenModelica, one reason was that if using Modelica libraries, that already have some components to represent rotations, torques and forces, they would not provide all the data needed to the simulation.

The solution found delays quite a lot the simulation calculation time and it's a simple approximation for the problem at hands.

In the dynamic models, the fundamental representation of the satellite orientation is obtained in quaternions. From quaternions, a direction cosine matrix (DCM) and Euler angles can be attained.

To present the results obtained from the dynamic model one timestamp was selected and the data available to represent the satellite orientation is presented and analyzed.

The orientation chosen at a random timestamp is represented visually on Picture 11. The cube represents the satellite and the arrow the direction vector.



Picture 11. Satellite orientation and direction vector.

Singularity-free, the quaternion data represents the orientation of the satellite and is presented on Table 2. It is a unit quaternion since its norm equals 1 (one).

Table 2. Unit quaternion.

Unit Quaternion	
w	0.832284
x	0.0492058
y	0.511685
z	0.207508

It is not intuitive to picture the satellite attitude from quaternions so Euler angles are converted from quaternions.

Euler angles are the easiest way to picture the object orientation, although singularities can occur. The Euler angles are presented on Table 3.

Table 3. Euler angles.

Euler Angles	
Roll (ϕ)	31.9677º (degrees)
Pitch (θ)	56.234º (degrees)
Yaw (ψ)	45.4032º (degrees)

Another representation that is mostly used on the 3D visualization is the direction cosine matrix. A three by three direction cosine matrix is presented on Table 4.

Table 4. Direction Cosine Matrix.

Direction Cosine Matrix		
0.390236	-0.295056	0.872157
0.395768	0.909038	0.130452
-0.831314	0.294264	0.471514

The conversion between these three representations has been verified and every representation denotes the satellite orientation.

7.2.3 Actuators

As there is no controller model to command reaction wheels, thrusters and magnetic torquers, they are controlled by a Boolean pulse, set to true at a pre-defined period. The changes on the satellite attitude are presented on Figure 17, being the attitude represented in quaternions.

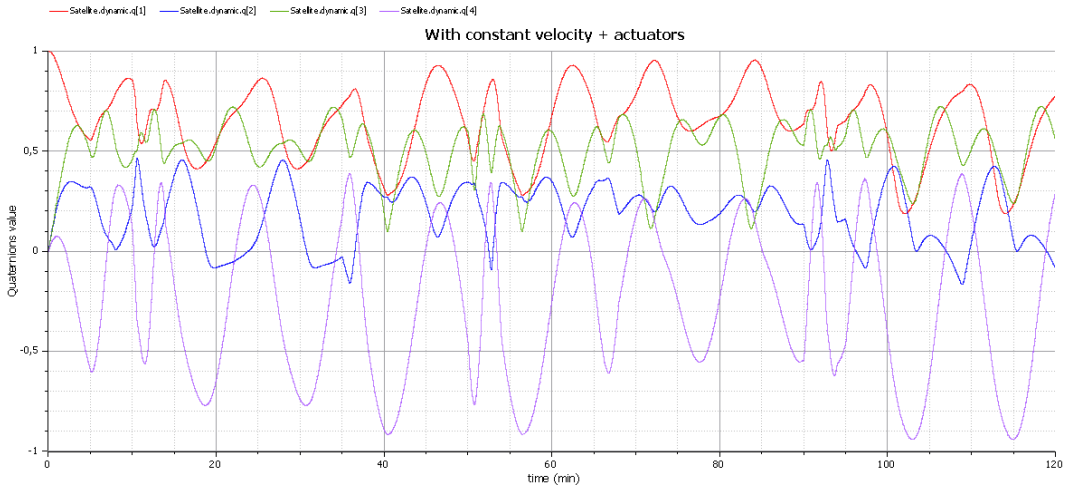


Figure 17. Satellite attitude with actuation.

The actuators are triggered at different times and employ different velocity changes onto the satellite. This produces the spikes on the plotted data. If a controller model were present, other tests could be performed. For example, to stabilize the satellite on an uncontrolled attitude after being launched from Earth. The algorithms of the controller model would activate the most suited actuators to perform the correction of attitude.

7.3 Simulated Sensors Results

The data gathered from the sensors is a fundamental piece of information. It can be used to verify the controller module and close the test loop.

7.3.1 CESS

The data from the six CESS simulated is divided on solar incident radiation and albedo incident radiation. On the example herein presented, the CESS heads are constantly

rotating with the satellite that has a constant angular velocity. The reason for this rotation, is to help validating the data, otherwise the angle between the CESS normal vector and the incident radiation vector wouldn't change and wouldn't present data variations over the course of a small simulation time.

The data is collected from each CESS's point of view, consequently, on each one's coordinate system. The data here presented was rotated to the satellite frame.

Sun radiation

As the satellite owns an angular velocity, it allows, over the course of one orbit, most of the six CESS to receive some solar radiation.

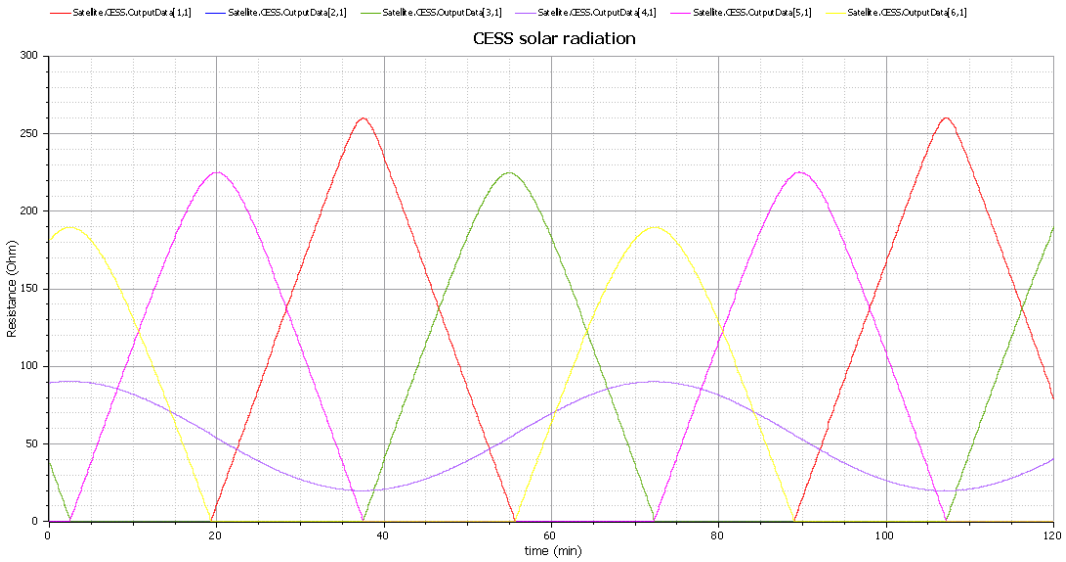


Figure 18. CESS solar radiation.

The maximum resistance for solar radiation has been defined to be 280 Ohms. As illustrated on Figure 18, only CESS number 1 peaks near that maximum resistance. As the satellite rotates, the output of each thermistor fluctuates based on the angle of the incident radiation.

Albedo radiation

The same logic from Sun radiation applies to the radiation reflected from Earth's surface. The major difference is that, as Earth's albedo is not constant across its surface, different altitudes correspond to different albedo values. This fact result in, as seen on Figure 19, the values to oscillate a like a stair-step pattern.

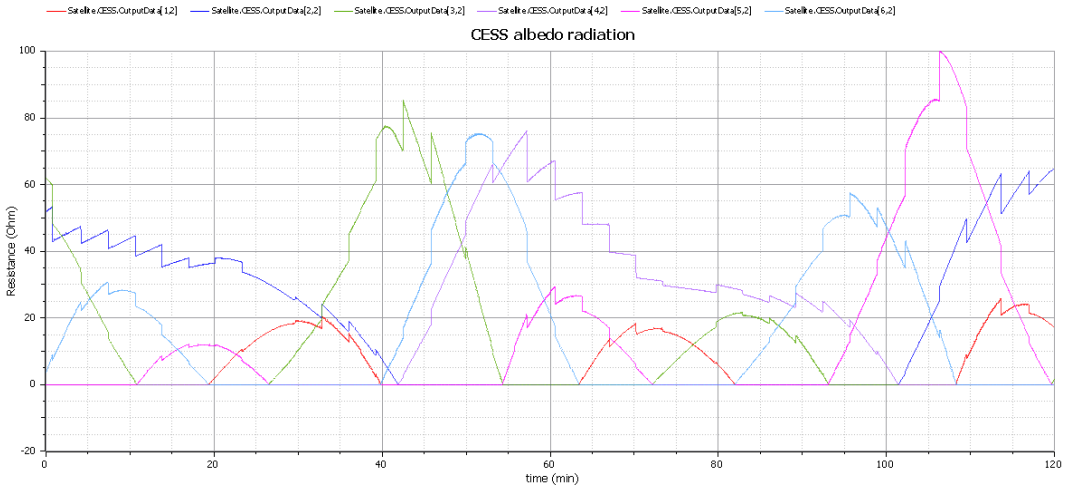


Figure 19. CESS albedo radiation.

Instead of having just a stair step pattern, it is also visible some curves that appear due to the satellite rotation. In this case, the maximum resistance was limited to 140 Ohms. One of the reasons for this maximum value not being outputted by the CESS is that, the satellite orbit is not completely passing over the poles, where the albedo presents its highest values.

7.3.2 Magnetometer

The magnetometer is measuring the strength of the gravitational field around the satellite. This data is presented as a component on each coordinated axis ($B_x, B_y\ and\ B_z$) and also as a total field intensity B, the unit used to represent them is nanoTeslas (nT).

The satellite orientation is not being taken into account. This orientation would affect the readings of the magnetometer, namely, the direction of the field.

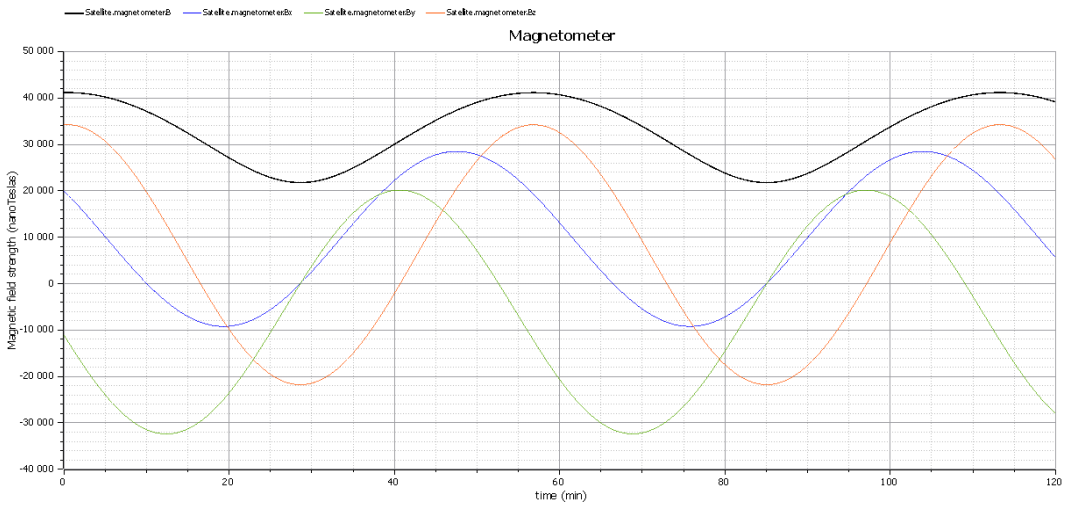


Figure 20. Magnetometer measurement.

The data presented on Figure 20 represents the readings of the simulated magnetometer.

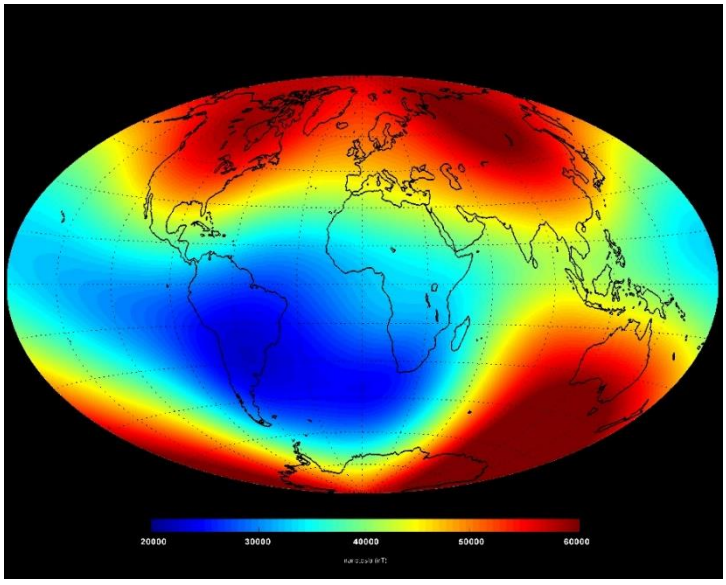


Figure 21. Magnetic field map (values in nT), June 2014 (ESA/DTU Space).

As the satellite is relatively close to Earth's surface, at about 795 km of altitude, the data collected is very close from values obtained from ESA's Swarm[8], a LEO satellite. That data is presented on Figure 21 that maps the intensity of Earth's magnetic field. As seen before, the satellite does not cross the poles, consequently, the data produced from

[8] Swarm mission is a LEO satellite with the objective of researching Earth's magnetic field

where the satellite makes passages range between 22.000 nT and 50.000 nT. The data produced by the simulation is a close approximation with values fluctuating between 20.000 nT and 42.000 nT.

7.3.3 IMU

The inertial measurement unit presents its data as differences between one timestamp and another, a delta.

These deltas, denote the satellite linear and angular velocities on each axis (x, y and z) and Euler angles. Like CESS, the IMU also makes its readings on its sensor's coordinate system and the data is then rotated to the satellite frame.

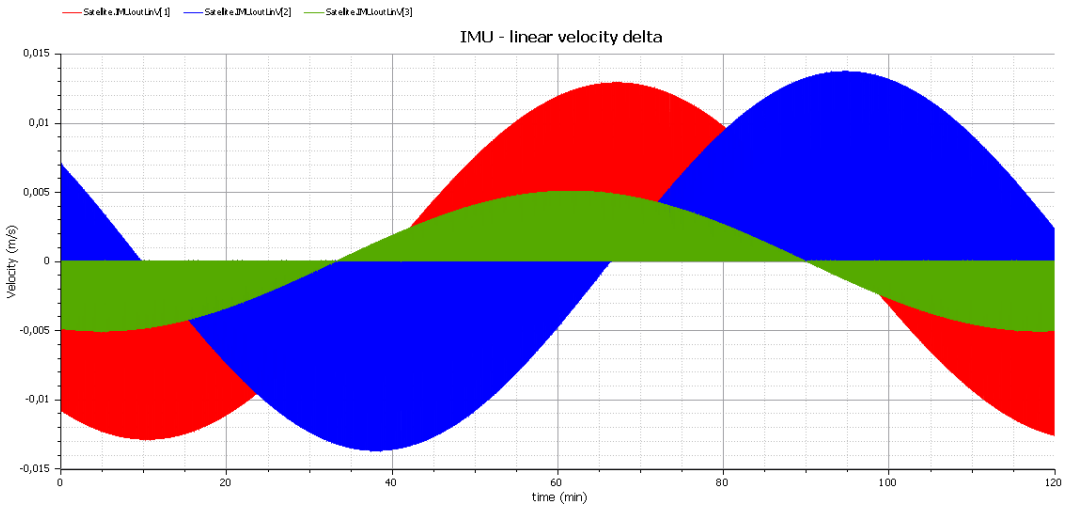


Figure 22. IMU linear velocity delta over 120 minutes.

The data is calculated every 1 minute and, when plotted, oscillates between the delta value and zero, creating, for instance, the plot that can be seen on Figure 22, the linear velocity deltas over the course of 120 minutes. The same behavior occurs to the angular velocity and for Euler angles deltas. This data would be used by AOCS making it harder to validate the produced data. As this is a subtraction of the current timestamp value by the previous timestamp value, it's assumed that the data is valid.

7.3.4 Star Tracker

The star tracker was not fully detailed and does not output the expected data. Instead, the star tracker outputs the star that he is currently pointing to (the Satellite dynamic is not being accounted for). The data expected from the star tracker is calculated from other models. For instance, the linear velocity on each coordinate axis is calculated from orbital elements. As the star tracker was not one of the key functionalities of the simulator, an approximation of its functionality was implemented.

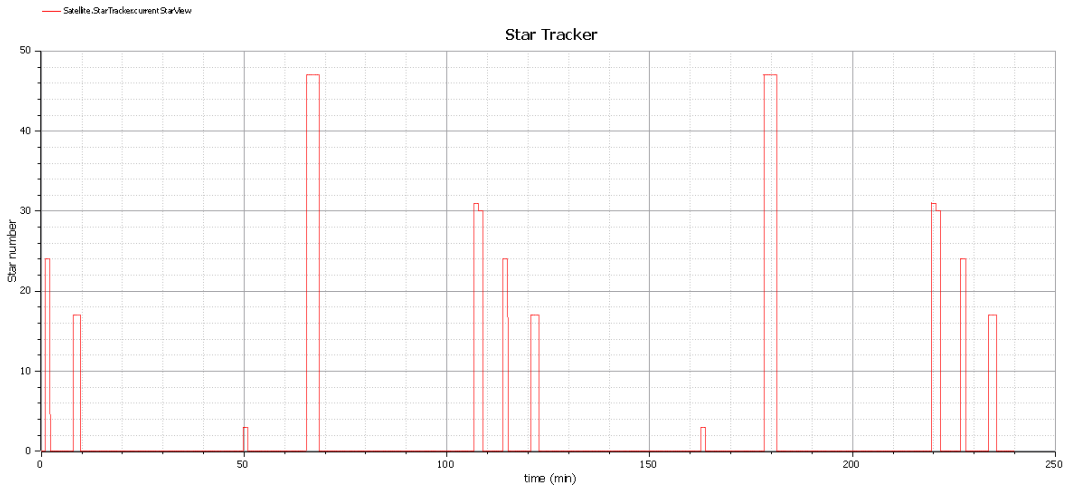


Figure 23. Start tracker plot.

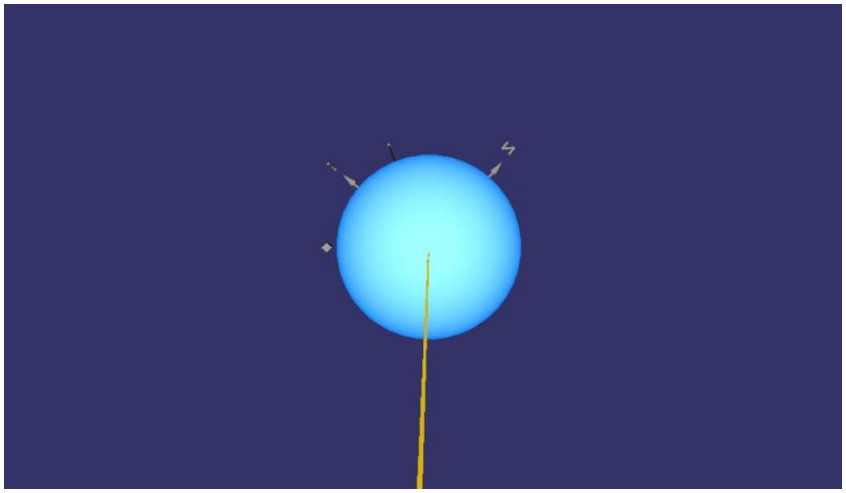
The results presented on Figure 23, reflect the star that the star tracker was pointing to. The plotted values represent the number of the star in question according to the star almanac.

7.4 3D visualization

A 3D visualization was created using a Modelica service that includes a Python server. The goal in mind was to develop a more appealing and easier to visualize way to present the simulation. It is an add-on to the work developed and all the other data is still available on OpenModelica plotting view.

This view, features the Earth as the central object, the satellite orbiting it and the Sun and Moon with their respective vectors to Earth. Those vectors were added to help locate the Sun and Moon with ease.

The greatest advantage of having a 3D visualization is that it creates a possibility to validate some of the calculations from simulation. For instance, in order to check the positions of the Moon and Sun, a specific date was chosen. On 16 to 17 September 2016 occurred a penumbral lunar eclipse[9]. This event is enough to test the calculations of orbits and the 3D visualization.



Picture 12. 3D visualization of the simulation during a lunar eclipse.

Although Picture 12 it's not an interactive 3D representation of the lunar eclipse, it's still visible that the Moon is right on Earth shadow causing a Lunar Eclipse.

The 3D visualization is a very useful supplement to the simulator. It brings interaction, visualization and helps validate the data that otherwise would be just a two-dimension plot.

7.5 Simulation Performance

The performance here denoted is the time that the simulation process takes to be calculated by OpenModelica, before it can plot the data.

A good performance is always desirable, long periods of simulation may be required and it's convenient that its calculation be within reasonable timings. As the quantity of

[9] A penumbral lunar eclipse occurs when the moon moves through the penumbral cone without entering Earth's umbra.

equations to be calculated, data to be processed and stored gets bigger, so does the simulation process time.

The simulation process time had a substantial increment, of about 500%, after introducing the dynamic and the 3D visualization. This is noticeable on long simulations and one of the reasons is that, those models trigger time events that need to be handled by OpenModelica consuming computation time. The time events represent about 50% of the total simulation time.

For small simulations, for example 120 minutes (slightly more than one satellite orbit), the process takes around 20 seconds and it's still a reasonable performance. If, longer simulation time is required, the 3D visualization can be disabled by commenting the object declaration. For instance, when simulating 1440 minutes, one Earth day or about 14.3 satellite revolutions, with the 3D visualization disabled, the process only takes 30 seconds. It's quite a reasonable process time and creates enough data to be analyzed.

8 FUTURE WORK

The simulation was designed to allow further implementations and improvements. Its architecture is flexible and compliant with improvements and new features.

If future work is desired, two paths can be followed, either both at the same time or one at a time. There can be improvements done, more detail and accuracy added to the models, or, new features can be added once attested useful to the simulation.

Some of the improvements that can be done are:

- Replacing the linear correlation of the sensors and actuators with a more realistic coefficient. For instance, on CESS, the relation between the incident light and resistance output can be calculated using a different coefficient instead of a linear correlation.
- Bowring's method could be used to improve the conversion from Cartesian to Geodetic coordinates (ECEF to LLA).
- Disturbances can be added to the dynamic model. Such as: gravity gradient, solar radiation pressure, atmospheric drag and magnetic field.
- The data from satellite dynamic could be integrated with the magnetometer, refining its data.
- The simulation definition of time can be improved, for example, assuring that minutes is used across all models, especially when using the latest OpenModelica software versions.
- The satellite orbital elements can be recalculated with the change of time. The same way that this is done for the Sun and Moon orbits, the satellite orbital elements can also be refined to maintain its sun-synchronous orbit.
- A more realistic sampling rate can be added to the IMU calculations.

New features that can be implemented are:

- Extraction of Functional Mock-up Interface (FMI). This will generate C code and can be used to close the test loop allowing the simulation to run separately from OpenModelica.
- The simulation of the satellite launched from a pre-defined location to its specified Sun-synchronous orbit.

- A solar power system can be modelled to simulate the satellite electrical system and power consumption.
- More interesting features implementation depends on what is expected from the simulation but, there is a wide range of subjects that can be implemented.

On both paths, the latest OpenModelica version should be used. A new promising version was released during the course of the simulator project execution but was not used because it was still a beta version and some errors were detected. This new version should increase the simulator performance and capabilities, like the simulation time unit and a 3D visualization already integrated with OpenModelica.

REFERENCES

Boain, R. J. (2004). A-B-Cs of Sun-Synchronous Orbit Mission Design. (p. 1). Pasadena, CA: Jet Propulsion Laboratory, National Aeronautics and Space Administration.

Costa, E. F. (2016). *Simulador Órbita Solar Síncrona.* Instituto Superior Engenharia do Porto.

ESA. ACT / ESA - Star Trackers. Consulted 19.12.2016 http://www.esa.int/gsp/ACT/ai/projects/star_trackers.html

GeoGebra. (2016). Consulted 09.12.2016 https://www.geogebra.org/home

Holst, R. (2014). *Using Magnetorquers with Magnetic Dipole Moment Cancellation.* Aalbord University.

Hughes, S. P., Qureshi, R. H., Cooley, D. S., Parker, J. J., & Grubb, T. G. (2014). Verification and Validation of the General Mission Analysis Tool (GMAT). *AIAA/AAS Astrodynamics Specialist Conference.* San Diego, CA.

Jet Propulsion Laboratory. (1995). *Thermal environments, JPL D-8160.* Pasadena, CA: National Aeronautics and Space Administration.

Kurjakov, A., Kurjakov, M., Mišković, D., & Carić, M. (2012). Electrical characteristics of thin film solar panels on a river boat under different microclimatic conditions. *Electronics and Energetics 25*(2), 151-160.

National Aeronautics and Space Administration. *GMAT*. Consulted 09.12.2016 https://gmat.gsfc.nasa.gov/

Nautical Almanac. (2016). Defense Dept., Navy, Nautical Almanac Office.

Schlyter, P. (2016). *How to compute planetary positions*. Consulted 2.12.2016 http://www.stjarnhimlen.se/comp/ppcomp.html#12

Sentinel 2A Satellite details. (n.d.). Consulted on 06.12.2016 https://www.n2yo.com/satellite/?s=40697

SpaceMath@Nasa. (n.d.). *Magnetic math.* National Aeronautics and Space Administration.

SpaceTech GmbH. (n.d.). *Coarse Earth Sun Sensor - CESS*. Consulted 22.12.2016 http://www.spacetech-i.com/images/products/satellite_equipment/cess/DataSheet_HCESS.pdf

Wertz, J. R. (1994). *Spacecraft attitude determination and control* (third edition ed.). Kluwer Academic.

Wikipedia. Orbital elements. (2016). Consulted 9.12.2016 https://en.wikipedia.org/wiki/Orbital_elements

Williams, D. D. (2016). Earth Fact Sheet. National Aeronautics and Space Administration. Consulted 2.12.2016 http://nssdc.gsfc.nasa.gov/planetary/factsheet/earthfact.html

Williams, D. D. (2016). Moon Fact Sheet. National Aeronautics and Space Administration. Consulted 2.12.2016 http://nssdc.gsfc.nasa.gov/planetary/factsheet/moonfact.html

Wolfram|Alpha. (2016). Consulted 9.12.2016 https://www.wolframalpha.com/