2022-2023 End of Year Project Report

Yuval Sharifi
Itai Carmeli
Ilaie Nadejde
Daniel Cohen
Gil Sason
Adi Aber
Capucine Cohen
Aya Rudek
Michal Broder
Saar Levi
Shachar Deyi
Shachar Hill
Shaked Levi
Niv Baumel Noiman
Ilay Lazarovich
Amit Malka
Dr. Revital Huber-Shalem
Shay Gurevitch

Contents

1 Introduction

Originated in the IAI, JERICCO (Joint-EffoRt Innovative Cubesat for Cislunar Orbit) is a student designed lunar CubeSat-satellite. It is being developed in a cooperative effort between the Technion- Israel Institute of Technology and IAI - Israel Aerospace Industries. The project is expected to launch to low lunar orbit in the coming years and execute a scientific mission. In this section, we will outlay the work of the teams throughout the winter semester and spring semester of 2022/2023.

1.1 Background

Final projects are unique because they are occupied with a new group each passing year; as the second group to work on project JERICCO, the teams worked hard to advance the project as much as possible.

The teams started the year with aligning with the work done in the first year and then continued with the development of the various systems and the simulation of the project.

In addition, this year we emphasized the clarity of the codes and the documents we produced in order to facilitate the replacement of teams next year. Our group is built from eight teams, and each team is working on a different part. The main achievements of each team are:

The Payload Team (Yuval Sharifi)- researched and searched for appropriate payload for the project. The team also searched for a similar mission to ours in order to design the specific features of our payload.

The Systems Engineering Team (Itai Carmeli, Ilaie Nadejde)- consolidated all system requirements provided by all other project teams into a detailed and easily understood document. In addition, the team worked on documents related to the satellite's states and modes throughout its lifecycle.

The Mission and Orbit Team (Daniel Cohen)- proved using Industrial tools (GMAT) that the assigned fuel amount is sufficient for orbit maintenance under the set orbit tolerances for the desired mission duration.

The Structure Team (Gil Sason, Adi Aber)-created a fully defined CAD model design of the satellite, which was verified with modal analyses.

The Thermal Team (Capucine Cohen, Aya Rudek)- calculated the overall ionized radiation with the "OMERE" software and made sure that our satellite is capable of withstanding it. The team included the Q_{in} (internal dissipation) in our heat analysis, and we made it even more accurate with the new data we got from the Tharmika software. We found out the minimum size of the radiators that need.

The Control and Propulsion Team (Shaked Levi, Niv Baumel Noiman)- found a suitable control law for detumbling using reaction wheels, and implemented it in the simulation. In addition, a model using engine for detumbling mode was planned and developed.

The Avionics Team (Michal Broder, Saar Levi, Shachar Deyi, Shachar Hill)- designed power and data systems for the satellite components by conducting a link budget analysis. In addition, the team checked for the ground station options and the solar panels ideal configuration with the help of MMA Design Company.

The Simulation Team (Ilay Lazarovich, Amit Malka)- rewrote the code in a more efficient and modular way, achieved better documentation, and implemented the control and orbital simulations fully.

2 Systems Engineering

2.1 Abstract

The JERICCO satellite is a complex engineering project that requires careful coordination between the different teams. The system engineering team plays a critical role in ensuring that the satellite is designed to operate as intended throughout its lifetime while securing all the mission goals. The system engineering team's job is to set concrete goals and procedures that correspond with every part of the satellite and to translate those goals into tasks and requirements that are processed and followed by the other teams.

The system engineering team coordinates between the groups to ensure the project's efficiency and success. This report showcases the team's progress this year. The team continued the work of the previous year's team. This report summarizes the literature review of existing material of the satellite building process and data gathered from the different groups. The system engineering team used this data to develop a priority algorithm, sequencing document, a logic flow for the JERICCO satellite systems, and create a list of demands from and for each team following the ConOps and Logic document from the previous year. Also created is a table of all the system states and modes which is updated actively, focusing on detumbling and communications for this semester. Additionally, the team worked closely with the simulation team to achieve a full working simulation that accurately reflects the demands of all the other teams and to acquire the models from the other teams. Throughout the process, proper documentation is adopted and formulated for the present and future teams, focusing on choosing a clear and comprehensive format and methods to present and sort the data efficiently.

2.2 Role and Objectives

As we approach the end of the year, it is essential to reflect on the accomplishments and contributions of the system engineering team in the project life cycle. While different groups within the project may have diverse goals and achievements, there are several critical roles that the system engineering team has fulfilled, and should guide it in the next years, which deserve emphasis. These roles serve as guiding principles for our ongoing and future work, setting the path for our success.

- Requirements – An up-to-date list of requirements is crucial for project success. This year, we worked on a comprehensive system requirements list, which we recommend the future team to adopt and continue. The list should detail project requirements and their relationships with each team.
- ConOps - This document captures the vision of the client and teams regarding the project, shedding light on its essential operations, requirements, and goals. It serves as a detailed roadmap, guiding throughout the project.
- Logic Document – This document provides an overview of the JERICCO satellite system, outlining its structure and interactions between different subsystems. It explains the logical

flow of operations, demonstrating the intricacies and relationships among components that enable the satellite's functionality.

- Reliability – A document aimed at illuminating the reliability of each system, subsystem and its components, along with the overall reliability of different operations.
- Risk Management - The document explains how we handle risks in the project. It includes a list of potential risks, details the risk assessment process, identifies responsible parties, and outlines the frequency of risk planning.
- Gantt - The Gantt chart is a visual project management tool that displays the tasks, timelines, and progress of the project. It helps plan, track, and manage the project efficiently.

2.2.1 Last Year's Progress

The following chapter provides a concise overview of the project's accomplishments in its initial year. This summary aim to give the future teams a brief yet informative understanding of the project's starting point. For more information, consult the "End of Year Report [1]" for Year 1.

- The System - JERICCO is a 12U satellite with a versatile bus capable of accommodating various internal configurations of components and actuators. It features a propulsion system for orbit maintenance and momentum unload, along with reaction wheels to control attitude. The power unit is charged from solar panels, supporting all systems throughout the mission. Other essential components, such as Gyro, CSS, and Star trackers, contribute to the mission's success, leading to a robust design.
- In its first year, the system engineering team dedicated efforts to conceptualizing the project's name and logo, developing the project's website, and overseeing project coordination.
- State Machine - During the first year, an initial state machine was developed, outlining the satellite's detumbling process, starting from its wake-up phase, transmission, solar panel deployment, and starting the mission.

- Control logic – A first draft of the control logic of the satellite was made, showing each of its states and modes.
- Logic Document and ConOps - In the first year, a comprehensive document was produced, providing a detailed description of the satellite's logic, operational modes, and associated requirements. This year's team continued this work, and it is essential to keep updating this document in the future. It serves as one of the core responsibilities of the system engineering team and guides the project's success. See the relevant documents for further detail.

It is important to highlight that a significant portion of the system engineering team's time is dedicated to coordinating between the different teams. This aspect of management is vital for both the team's success and the overall project. The system engineer should possess a good understanding of each team's work and progress to ensure seamless collaboration and project advancement.

2.3 This Year's Progress

This year, the system engineering team underwent a mid-year staff change, prompting us to divide the report into two sections: activities carried out in the winter semester by Daniel Cohen, and those conducted in the spring by our team. Building upon the accomplishments of the previous year, our primary focus was to enhance the state machine, ConOps, and logic document. Furthermore, we aimed to initiate the development of comprehensive documents for system requirements, risk management, and reliability – areas that were previously unaddressed in the first year.

2.3.1 Winter semester

During the winter semester, Daniel made significant progress in the communications aspect of the mission and the detumbling phase. Specifically, he established requirements and logic for these procedures.

2.3.1.1 Detumbling state

The satellite is ejected from its carrier at an uncertain orbit around the moon. the satellite has limited but unknown angular velocity. It is required at this point to stabilize the satellite, determine the sun's direction, and correct its orbit such as at the end it will be at the nominal orbit with solar panels pointing to the sun. There are 2 possible procedures, if the solar panels have a sun detector while in the closed position it will be possible to search the sun and then deploy the solar panel. If the solar panels sun detector can only be operational when the panels are open, then first the satellite will need to open the solar panels and then search for the sun. After the solar panel is chosen, then the procedure can be determined, for now , both are mentioned.

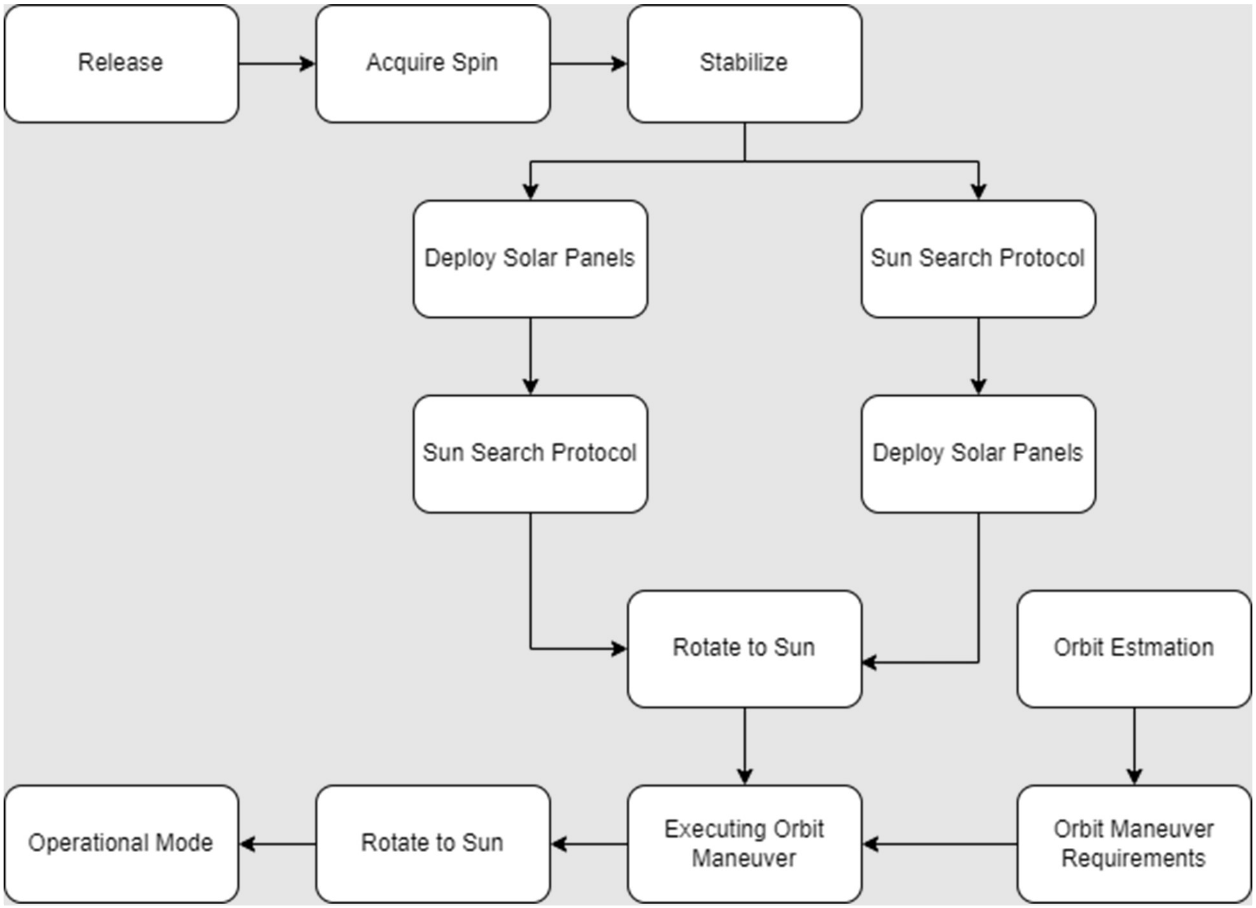


Figure 1 - Detumbling Protocol

Initial conditions:

- Spinning rate of up to 30° per second (TBD), relative to any possible body vector.
- Initial attitude is unknown.
- The satellite is released at the ascending node of the nominal orbit with zero eccentricity.
- The variance in the orbital elements is:

$$\Delta h = \pm 10\text{km (TBD)},\ \Delta i = \pm 2°\text{(TBD)}$$

- Mission starts at 19/09/2024 at 10:00 UTC. (This date is subject to change as the project advances every year. Relevant orbit calculations should be revisited for each new launch date. It is recommended to have several back up launch windows)
- Solar panels deployment is in the order of 10 seconds (TBD).

Attitude Control Requirements

- Design stabilization control to stop the rotation of the satellite under 5 minutes.
  1. For nominal cases consider spin around body principal axis.
  2. Find extreme cases for which time to destabilizing is exceeding the required time or when the usage of propulsion system is required.
- Design sun search control, required time to located the sun is up to 10 minutes, use only reaction wheels.
- Study the possible effects of solar panels deployment on the sun search control:
  1. How does the open state versus the close state of the solar panels affect the requirements from the reaction wheels?
  2. Does deploying the solar panels increase the locating time?

For both controls find extreme cases for which time to completion is exceeding the required time or when the usage of propulsion system is required.

- Design orbit maneuver control, input is defined as a set of vectors at different times for which the propulsion system is required to be aligned to.
  1. Assume initial attitude is sun pointing, the error from this value will be the input to the control scheme.
  2. After maneuver execution return to sun pointing attitude.
  3. Report any limitations such as excess use of reaction wheels, propulsion usage for attitude control and more.
  4. Calculate the required fuel needed to execute the maneuver.

Orbital Control Requirements

- Calculate orbit correcting maneuvers:
  1. Consider the cases for minimum time to execute or minimum ΔV required.
  2. The propulsion is continuous, no impulses are available.
  3. Provide the vector of thrust overtime required.
  4. The initial time window for executing the maneuver is 5 laps after release.

At the end, we desire to simulate various scenario of the mission, and the Detumbling State is one of them. At first it is alright to report with some MATLAB scripts and a word document, but to understand further variances of the initial conditions on the results, a MATLAB app is the best option, this need to be achieved by the future teams.

2.3.1.2 Communications Mode Requirements

Communications with a satellite is a vital part of the mission, it is the only way to monitor and operate its systems, and to receive the mission data recorded by it. Our satellite is equipped with 2 antennas, S band for (mainly) receiving data and X band for (mainly) transmitting data, and the (current) payload design is an optical type. We would like to be able to transmit (and receive) as much data as possible "in on go" (meaning in one data transmission session) in order to maximize data acquiring from the payload (data overwrite itself when full) and so it is a key to maximizing the efficiency of data transmission session. These are the key features determining the communication mode efficiency:

- Earth visible line of sight time span.
- Stored data files size.
- Data transmission speeds.
- Available power.

Several points to notice:

- We assume that the bulk data is payload data, so designing for its transmissions should be enough for other data communications.
- There is a difference between the day and night communications as transmitting data at the X band is much more power demanding than the S band.
- Temperature management is also a consideration. Transmitting during the daytime may not align with the satellite's optimal attitude and could potentially lead to overheating.
- Most likely there will be time that not all data could be send or received in one go, and therefore we need to prepare for efficient data acquiring and storage of the satellite's systems.

General teamwork requirements:

- Orbit Team will be tasked with creating data about the earth and the sun position locations relative to the satellite (two pointing vectors).
- Attitude Control Team will be asked to provide a controller to keep the satellite pointing to earth for the required time.

- The Avionics Team will be requested to produce a paper, document, or simulation (TBD) outlining the extent and type of data that can be transmitted, considering communication time spans and available power levels.
- Space Condition Team will be tasked to determine based on sun positioning and the satellite attitude if there is any thermal risk while communication is done.
- System Engineering will be tasks to determine the communications protocol for the various possible scenarios. In addition, it required to make a prioritization list for data storage and transfer, meaning what data is prioritized for transmission and at what conditions, and what data is prioritized to be stored when the memory is full.

2.3.1.2.1 The simulation

A basic ODE45 for the Kepler problem was done for 3 bodies:

- The Earth around the Sun.
- The Moon around the Earth.
- The Satellite around the Moon.

All orbits were moved to a single frame of reference. The lines of sight from the satellite to the Moon, Earth and Sun were calculated. The angle between each line of sight was calculated. If one body blocked the line of sight to another, a flag was recorded. If and only if there is a clear line of sight to the sun and the earth, communication is possible. In reality, this is not true, we need only to see the earth, but then we may lose power quickly. The Max, Min and Average time of possible communication was calculated.

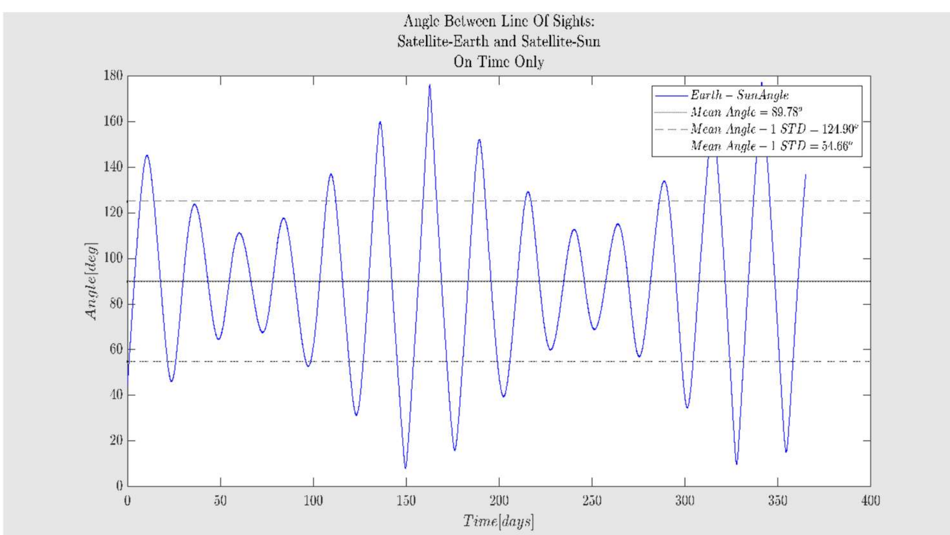


Figure 2 - COMMS Angle between line of sight

2.3.1.2.2 Conclusions from COMMS simulation

- Shortest Period of Available Communication: **0.21 [min]**
- Longest Period of Available Communication: **7687 [min]**
- Average Period of Available Communication: **79.55 [min]**
- Shortest Period of Lack of Communication: **0.44 [min]**
- Longest Period of Lack of Communication: **59.10 [min]**
- Average Period of Lack of Communication: **24.85 [min]**

This means that the average time period for communications is relatively long, around 79 minutes which is about 60% of the orbital period.

There are several long periods of time (in the order of days) that we can have communications, we should prepare for these days to make the most out of them.

The lines of sight angle vary from 0 to 180 degrees, but on average it is around 90 degrees, the current position of the antennas seems okay.

2.3.2 Spring Semester

2.3.2.1 System Requirements List

14	Frame of Reference Used	Body Frame	Systems Engineering
15	Epoch Used	19/09/2024 at 10:00 UTC	Systems Engineering
16	Maximum Spin rate on one axis	30 degrees per second	Systems Engineering
17	Time for Detumbling	At most 5 minutes	Systems Engineering
18	Max time to locate sun	10 minutes	Systems Engineering
19	pointing tollerance of antenna to earth	5 degrees	Systems Engineering
20	pointing tollerance of solar panel to sun	7 degrees	Systems Engineering
21	pointing tollerance of radiator to given command	7 degrees	Systems Engineering

Figure 3: Sample of System Requirements List

Throughout the semester, we created a comprehensive system requirements document, fulfilling a goal established by our predecessors. We did this by having tough talks with the different teams. This document provides an exhaustive overview of the project's system requirements, encompassing both quantitative and general aspects. It delineates specific requirements, such as tolerance for inclination, altitude, and attitude. Additionally, it attributes each requirement to the relevant team, specifies the originating team, and tracks its progress. This inclusive list encompasses all requirements from small to big, and thus makes it easier to translate requirements into tasks for the system engineer.

The system requirements list holds the following:

- Requirement – A clear explanation of the specific requirement.
- Value of requirement – If it exists, the quantifiable value associated with the requirement (e.g., tolerance).

- Team that brought it up – The team responsible for suggesting the requirement.
- Team that it is relevant for – The team accountable for fulfilling or addressing the requirement.
- Stage of life – The current status of the requirement (e.g., pending action, in progress, completed).
- Where it stands – The necessary actions to achieve the requirement.
- Category – The requirement's classification, such as Power, Fuel Budget, Integration, States and Modes, Communications, Sensors, or Orbit. The list can be expanded.
- Other team that it is relevant for – Additional teams involved in addressing the requirement or requiring its information.

The complete list is accessible in the project drive, in the Systems Engineering Folder, in the section Systems Requirements. We strongly advise maintaining, expanding, and updating it throughout the project's future progression.

2.3.2.2 Coordinate Reference Frames

During the semester, confusion arose concerning the choice of a coordinate system. Each team proceeded with their individual definitions, leading the system engineering team to collaborate with the simulation team to work on a uniform coordinate system. The objective was to establish a consistent and mutually agreed-upon coordinate system. This uniformity is crucial to ensure that the simulation team can receive inputs from other teams in a certain coordinate system and present results that everyone can interpret, reducing the likelihood of unnecessary errors. Transitions between coordinate systems can be effectively achieved using the correct combination of cosine matrices.

2.3.2.2.1 Other Teams Coordinate System

A short overview of part of the utilized coordinate systems by different teams; for in-depth information, refer to the respective team's chapter.

Structure:

Center in the geometrical center of the satellite.

- Y axis – Towards the payload
- Z axis – Towards the solar panels
- X axis – Oriented as such to create a right-hand coordinate system

Control & Propulsion:

Center in the center of mass.

- X axis - Towards the solar panels
- Z axis – towards the payload
- Y axis - Oriented as such to create a right-hand coordinate system

Thermodynamics:

Center In the geometrical center

- X axis – towards the payload
- Z axis – towards the panels
- Y axis - oriented as such to create a right-hand coordinate system

2.3.2.2.2 Agreed upon, chosen coordinate systems

The center of the coordinate system is chosen to be on the geometrical center of the satellite where:

- X axis – towards the solar panels
- Z axis – towards the payload
- Y axis - oriented as such to create a right-hand coordinate system

Each team will transfer from their system to this using the correct cosine matrix transformation.

LVLH:

- X axis – towards the satellite specific angular momentum vector.
- Z axis – towards the moon
- Y axis - oriented as such to create a right-hand coordinate system

TOD:

Non-inertial, turning with the moon coordinate system, whose center is in the geometric center of the moon.

- X axis – towards the closest to Earth point of the moon's equator.
- Z axis – towards the geometric north pole of the moon
- Y axis - oriented as such to create a right-hand coordinate system

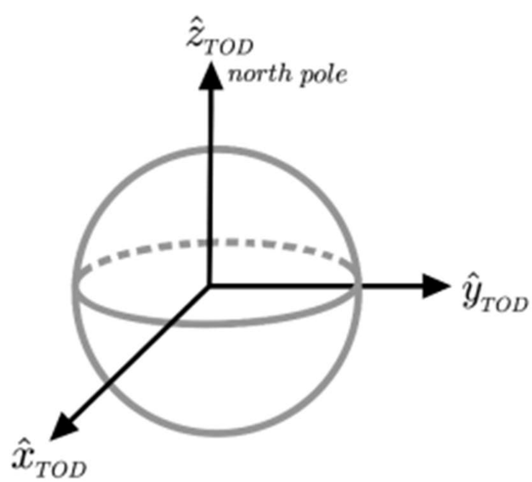


Figure 4: TOD Coordinate System

The coordinate system detailed above serves as the project's default coordinates unless otherwise determined. It is essential for every team to acknowledge and understand this established system, and to understand how their work translates into other coordinate systems and the default one.

2.3.2.3 Logic Flow Algorithm

Upon reviewing the previous year's work, the spring semester's Systems Engineering team found it difficult to understand and comprehend the logic flow of the overall system of systems planned for the JERICCO CubeSat. That is to say, the various definitions of states, modes and flags were not cohesively represented in all the documents provided by previous teams. Hence, it became evident that an easily understood logic tree needs to be built, by the systems engineering team, but for all the other teams. This logic tree would incorporate every stage for the CubeSat's system, what its inputs at each stage are, what the decision points between different modes are, and what the requirements that were researched through the requirements list are. The creation of this logic tree tries to incorporate everything that previous years have done and set up a method for future teams to work that will be transparent for all those needed to understand the information at each stage of the satellite's lifespan.

With that, it is important to understand several definitions:

Stage: It is the point in the lifespan of the satellite. These are the general descriptions of what the satellite is doing throughout its 1.5 year mission plan.

Sub Stage: This is a category that describes an action being done by the satellite at a specific stage. These substages are a more specific description of what the satellite is doing in each stage. The substages should, in the future, incorporate general procedures to follow order by order and shall be represented in the logic tree as a whole.

The following are the stages and substages mapped out by this year's team, in the order of algorithmic importance:

1. Pre Launch (from mothership)
2. Launch (from mothership)
    a. Detumbling
    b. Acquire Sun
    c. Establish Contact
    d. Downlink
    e. Uplink
    f. Instructions (Testing and corrections)
3. Cruise
    a. Enter Orbit
    b. Charge
    c. Mission
    d. Uplink
    e. Downlink
4. Extension of Mission
5. Decommission

Mode: Modes are intended to describe the specific objectives of the satellite and its system parameters during cruise. As seen in the stage above, the Cruise part of the CubeSat's life includes many different substages. Each substage requires different systems to work differently depending on the environment in which it works. Hence, a night mode and day mode. There is also a communication mode for when there is a line of sight with Earth, and a charging mode for when the battery is low enough and a LOS with the sun is available. There is also battery saving mode for when it is necessary for the ground team to run tests or when the satellite senses any potential dangers. These modes were not developed this semester; however it is highly recommended that the logic tree be continued next semester to include them.

Flags: These are coding signals that are changed when a sensor determines a specific system or environmental change. For example, the battery depletion flag entails that the battery is too low. The Sun acquired flag indicated that the sun sensor acquired the sun and that the solar panels are in optimal position to receive sunlight. If these two flags are up, then the system should move to the Recharging mode. A list of all possible flags that should be incorporated into the CubeSat's system was determined by this year's team:

- Battery Power
- Data Storage
- Earth Contact
- Voltage/Current of each component
- Angular velocity of satellite over tolerance for sun sensor
- Angular Momentum of each reaction wheel is under saturation point
- Wheels 1-4 are working
- Engine Operational
- Fuel Remaining
- Sun Acquired
- Night/Day
- Star tracker Acquired
- Each Side of satellite < Heat tolerance
- Orbit determination is out of date
- Payload Directions
- Received Instructions
- Mission (autonomous or linked, or none)

The stages, substages, modes and flags are all incorporated by the state machine of the simulation team. The parameters worked on by the systems engineering team this semester were all designed in tandem and constant communication with the Simulation team, as is recommended to do so in the following years. This is because the simulation team would build the code for the simulation that is identical to the code that will be in the satellite system itself. If the simulation team sees any problem with the state machine, it will undoubtably be a problem for the satellite and hence the two teams need to work together, as we did this semester.

This year's progress on the logic tree can be seen though the following link:

https://lucid.app/lucidchart/1b47d1d9-22b0-49d1-8ed9-78ef46596dff/edit?viewport_loc=347%2C338%2C1710%2C723%2C0_0&invitationId=inv_294666d8-4261-4517-afab-19e452f1e9d4

(It is required to have a sign in to Lucid for this link to work, yet it is free and can be connected via a Technion email account)

An example of it can be seen in the figures below:

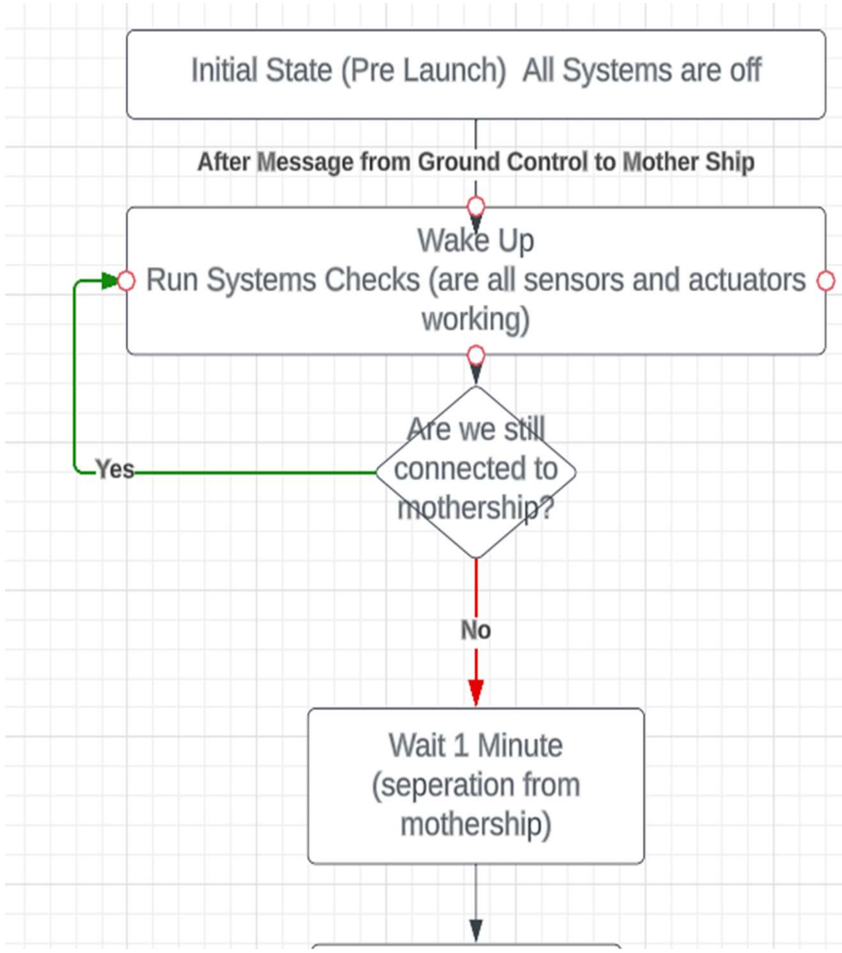


Figure 5: Beginning of Logic Tree

2.3.2.4 Documentation

One of the biggest challenges of this kind of project is the constant turnover. Every year, each team changes personnel, and hence, documentation is super important. It allows future teams to more easily understand not only the past of the project, but also the future tasks necessary to complete. This semester, the Systems Engineering team helped create end of year report format that would be uniform not only across all teams, but also across all years. This would be helpful to catch up on the work at the beginning of each semester. This semester we also put in effort to create GANTTS that would be available for all teams and for all years to see the progress of the project throughout the years. We implemented a group Monday account to for all teams to better list out their work schedule and any future tasks they might think are relevant. We highly recommend that future years use tools like these to record their progress and implement efficiency tactics. This also allowed for constant communication and updates between the teams themselves.

Furthermore, with the introduction of a new team and fresh perspectives, the existing project documents will require updating. It's evident that as the payload and mission selection process continues, these documents will undergo further revisions. Hence, they are regarded as 'dynamic documents'.

2.4 Future Work

A lot of work was done by all teams in all years on this project, yet a lot remains to be done. It is important to recognize the lessons learned from each year so that future teams can implement them and have better progress in the future creation of the JERICCO CubeSat. With that, this section is designed to reflect ideas to better improve efficiency as well as outline assignments that still need to be done.

The first lesson we learned is that each year, and each team, needs to have a schedule of their planned assignments. We recognize the reality of a student project, which changes personnel every year, and that student's schedules are not very free. With that reality, it is important to breakdown goals for each team for each semester, and that all teams are aware of those goals for the whole project. This is the job of the systems engineering team, to consult with all project members as well as the design consultants on what those goals should be. Additionally, the schedule and timeline planning should be reflective of a larger GAANT that has the mission as a whole scheduled and planned out. That means every year knows the plan for what each year after is supposed to do, and changes that plan according to their progress during their work on the project in the school year. So, the systems engineering team should make sure that each team has a set of goals in the beginning of each semester, and that they keep updating the schedule of the satellite as whole.

Another lesson learned is the importance of organization. Each team should be organized with their tasks moving forward, as well as all previous work done on the project that is pertinent to them. To house and transfer files between teams and years, we used Google Drive. To organize assignments for each team, we used Monday.com. These are not necessarily the best methods to stay organized, but they are a method, and it is important that each year uses an organization method. To save time it is recommended to use the platforms of previous years.

It is also important to document the progress of the project throughout the work itself, and not the end. By the end of each semester, students are not focused on the project and can't remember everything they did to work on the project. So, it is recommended that each team submits a report of what they did on the project each week. It is important to have this report written so that for the end of year report, an analysis of these reports can increase efficiency and accuracy.

One of the most important aspects of this project is communication. Communication between the different teams as well as between the members of each team themselves. Each team should have a basic understanding of what all the other teams are working on, what their goals are, and any points of impact that they might have to work together. The systems engineering team is responsible for implementing this communication and encourages methods that will be helpful for the entirety of the project.

After applying these lessons to your work plan, there are several tasks remaining to continue the work for the JERICCO project:

2.4.1 Pick Payload and Mission

The true goal of this project is to teach students how to work together and build a satellite which incorporates several different teams and departments. However, most satellites built are designed with a specific science or engineering mission in mind: communications, observation, scientific... These mission definitions define the parameters and requirements for the whole design of the satellite. The main problem that JERICCO has is that there is no such mission to guide the teams in determining its design. A mission has a strong connection to the payload being carried by the satellite. Without a payload there is no mission. Without a mission, teams will not know how to finalize their designs, and some critical tasks and requirements cannot be undertaken until such a payload is determined. Therefore, as its first and highest priority, the project team next year must decide on a mission and payload.

It is recommended to introduce payload options offered by faculty members in other departments, such as electrical engineering or physics, so that a broader cohesion of Technion students would be involved.

2.4.2 Project GANTT

To have a better idea of the progress of the JERICCO project, we shall provide a draft for the plan for the development of the satellite. This is subject to change once the payload and mission is defined, and so we recommend that a redo of this schedule should be done at the end of the first semester next year. Additionally, it is recommended that the schedule in subsequent years is determined between the winter and spring semesters, due to the fact that most students in the project will finish their studies at the end of the spring semester and it is harder to consolidate everyone's notes once they graduated.

A more visual friendly GANTT can be produced and seen on the project's Monday account, yet here we shall write out the main points for each semester. It is important to note that these objectives are general to the entirety of the project, and that induvial goals for each team will be detailed in each team's chapter. Likewise, this plan underscores the reality of students designing a long-term project, with limited time windows to commence the actual work and a turnover of personnel every year. Though the GANTT should be adhered to, it is recommended that it be updated every semester.

Year 2023-2024

Semester A

1) Determine Payload and Define Mission
2) Adjust calculations to fit chosen payload and mission
3) Determine changes required to systems, algorithms and components based on chosen payload

Semester B

1) Finish algorithms and calculations up until cruise state
2) Finish analyzing all risks for entirety of mission
3) Finalize updated components and suppliers list

Year 2024-2025

Semester A

1) Finish algorithms and calculations for cruise state

2) Finish full scale simulation up until cruise state

3) Contact all suppliers for all components

Semester B

1) Finish algorithms for end of life of satellite

2) Finish Risks analysis

3) Create design and build schedule for physical satellite

4) Obtain budget for building the satellite

Year 2025-2026

Semester A

1) Finish full scale simulation up to end of life

2) Finish redundancy algorithms

3) Start ordering parts and building the satellite

4) Work on budget for project

5) Contact launcher

Semester B

1) Build the satellite

2) Test all individual components delivered

3) Communicate with launcher

4) Finish simulation that includes redundancy and unpredictable problems

Year 2026-2027

Semester A

1) Build the satellite

2) Test entirety of the satellite

3) Communicate with launcher

4) Simulate potential problems throughout the mission

5) Work on budget for launch

Semester B

1) Finish building the satellite

2) Test entirety of the satellite

3) Communicate with launcher

4) Plan for launch

2.4.3 Individual Teams Project GANTT

It is also important that each team will have its own GANTT at the beginning of each year so that they can better stick to the general plan and communicate any issues with that plan as the work progresses. Using the above outline for a basis of the goals for the project as a whole and the goals reached by each team this year, here is a basic outline for the goals for each team for the next 2 years:

	2023-2024 Semester A	2023-2024 Semester B	2024-2025 Semester A	2024-2025 Semester B
Systems Engineering	-Communicate requirements between all teams stemming from new payload -update logic flow and algorithm	-Organize all modes, flags and states for the duration of the satellite mission -Consolidate risks from all teams regarding the components and mission	-Coordinate all requirements from all teams with an emphasis on building a simulator	-Update all documents (logic document, ConOps, suppliers, requirements list)
Payload	-Decide on Mission payload -Write out mission plan	-work on payload requirements and limitations	-Design and build payload	- Design and build payload
Control and Propulsion	-Fine tune the finished control loop to fit the new satellite properties -Develop Sun Search Algorithm - Implement the engine's limitations	- Implement the combined wheel-engine control loop - Determine the optimal thruster tilt angles	- Develop an altitude control loop architecture and simulation	- Analyze fuel consumption for the mission and implementation
Structure	-Adjust CAD design to include payload -Fine tune the vibrational analysis to include payload	-Test structure resiliency	-Contact suppliers for parts	-Start building Structure
Simulation	-Implement GUI -Update all requirements and limitations	-Create enhanced thermal module integration	-Final State machine implementation	-Incorporate redundancies algorithms into simulations

	determined by new payload	-Payload module implementation		-Create list of problems to simulate
Mission and Orbit	-Update orbit data using new payload parameters -Create method of orbit determination	-Final procedure for orbit determination throughout the mission -Calculate Day and Night time during throughout the mission	-Create orbit simulation -Create alternative orbits -Plan for any anomalies in day and night and LOS times (eclipses...)	-Create simulation of varying orbital problems
Avionics	-determine final configuration of solar panels -Determine final method of communications (piggyback or LOS)	- Monitor power balance for each orbit - Characterize the satellite antenna	-Input all electrical component requirements to simulations	-Order parts for first stage of building and circuiting
Thermal	-Solve the optimal radiator size when the components act separately - Perform a material analysis with the Structure team on Solid Works	-Analyse if the satellite can withstand cold cases - Improve the calculations of the internal heat	-create thermal simulation problems and potential solutions	-test components ordered from suppliers for thermal resilience

Table 1: GANTT by team for 2 years

It is important to note that these are teams that were designated in the first year of the project and continued into its second year. In the future, it is more than likely that new teams will be needed, such as Ground Control, Software Engineers, Materials Testing, Assembly Team... Future project members should be flexible at the beginning of each year to create new teams and recommend new teams for each subsequent year.

2.4.4 Specific Future Work for System Engineering

There are components of the work done by the Systems Engineering team that require further detail. It is important to understand, for each task, what has been done up until now and what is recommended to do in the future.

2.4.4.1 Update and Combine Logic Flow for all States and Modes

In year 1 and year 2 of the JERICCO project, a logic flow was designed. These were logic flows that were designed without a payload in mind. With that, we hope that in the next coming semester a payload and mission will be defined, and it is recommended that the logic flow be rewritten to fit the designated mission. It is also important to consolidate all the information of the algorithmic process that the satellite will ensure in its lifetime into an easy to follow flow that parallels the lifespan of the CubeSat. The entire lifecycle of the satellite is described below:

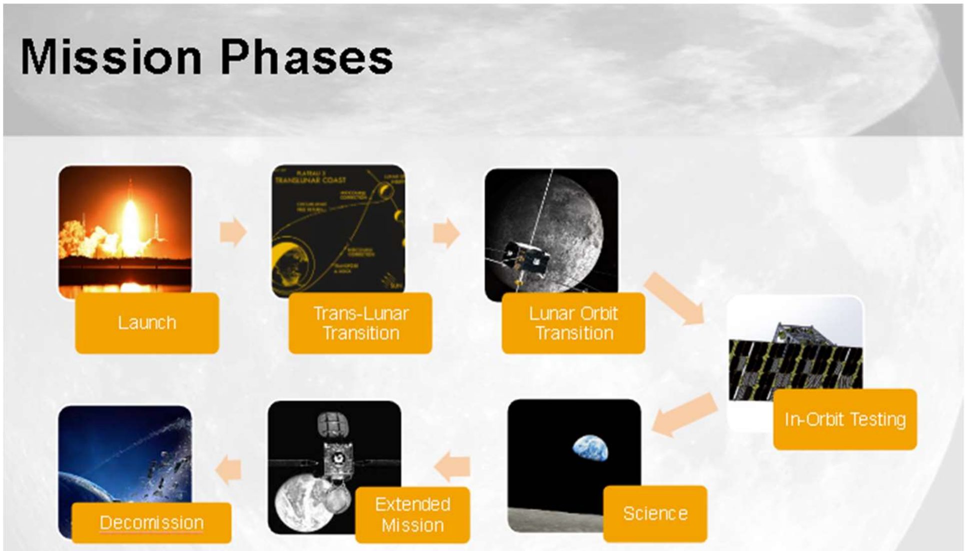


Figure 6 – Lifecycle of the JERICCO Satellite

For each phase of the mission, the potential states, modes and flags must be listed in an orderly fashion, and the transition between each of these subcategories must be explained in detail. This year we worked on the algorithm up until the cruise section in the In-Orbit testing phase. So, to continue, a review of the previous logic flow must be done and a continuation of the logic flow until decommission must be written out and saved.

Additionally, a review of the requirements for night and day mode operations must be done. Maybe the satellite payload will need to use energy at night. Even if not, a calculation of the charge time for each orbit must be done so that we can know how often the CubeSat can use the payload. This is in communications with the Mission and Orbit Team, the Avionics team and the Payload Team.

Furthermore, a comprehensive list of possible redundancy systems must be written to help ascertain any potential solutions to probable problems during the mission. Extra components, changing missions and orbits, using only one component (solar panel for example) are all ideas that could help save the CubeSat mission when it is already in orbit and so a comprehensive list of such readiness functions should be written.

2.4.5 Updating Documents

With the eventual designation of a mission and payload, it is important to update and finalize all pertinent documents. The original Logic and ConOps documents were written in the first year of the project. Many things have changed since then and so an update by the end of next year is necessary for these documents. The new updates shall include the system restrictions and new parameters for those restrictions determined by the individual teams handling the components of those systems.

Additionally, it is necessary to start working on creating formal documents with new information that needs to be presented, such as the Risk Assessment document and the Reliability Document.

The Risk Assessment document demonstrates positions that might cause trouble for the satellite during its mission as well as in its assembly, as well as steps suggested to take in order to mitigate those risks. The Reliability Document is a document that assesses each component used in the CubeSat itself and assesses the probability of their failures.

2.4.6 Communication to Simulation

There are two teams that are in constant communication with all the other teams and can see the project as a whole: Systems Engineering and Simulation Teams. Where the Systems Engineering helps to communicate the requirements of different systems and components to the different teams, the Simulation Team takes those parameters and uses them to design a simulation of the mission. The logic flow that the Systems Engineering team designs for the CubeSat is the same logic flow that will be implemented in the simulation. Hence, constant communication between the two teams is necessary. Going forward, it is recommended to create weekly meetings between the two teams, and to update each other on the respective progress. The finalized requirements and logic flow designed by the systems engineering team should be easily understood and accessible to the simulation team, and vice versa, any simulation code written by the simulation team should be transferred to the systems engineering team as well. Potential issues with different requirements can only be determined once they've been simulated, and hence these updates are crucial for detecting problems early on and relaying those issues to the relevant teams so that they can work on solutions.

2.5 Conclusions

The Systems Engineering team plays a pivotal role in the design and assembly of the JERICCO CubeSat, serving as the orchestrators of a complex and multifaceted project. Responsible for the holistic design, integration, and operation of the satellite, the team brings together the diverse expertise of each team, spanning various disciplines in order to create a cohesive and smoothly run system of systems. Their mandate encompasses defining mission objectives, allocating resources efficiently, managing technical trade-offs, and ensuring seamless collaboration among subsystems and their designs. The JERICCO mission is one that holds many obstacles and challenges, and the systems engineers should know every aspect of the satellite that is affected by those issues and how the teams are planning to tackle them. By becoming the nucleus of all relevant information, it is easy to spot points of intervention. For example, this past semester, we realized that each team was working in a different reference system. Once we realized this, we immediately set aside a full project meeting where we decided on a single unifying set of reference frames for all teams to use. Each team would not have realized this was necessary on their own, and hence the systems engineering team is crucial for the integration and eventual launch of the satellite. By meticulously addressing challenges such as thermal management, power distribution, communication protocols, and navigation systems, the systems engineering team ensures that the satellite functions as an integrated, cohesive unit in the demanding space environment.

2.6 References

[1] Michael Malka , Tom Shneor , May Alon , Yotam Granov, , Alon Binyamin , Omer Zakar , Meshi Blum , Tom Itzhaki , Oren Amber , Yarden Arad , Beni Muchnik , Tobi Weinberg , Geffen Aharoni , Tamar Alperin , and Dan Unter, " Project JERICCO End of year progress report" , Projects google Drive

3 Payload Team

3.1 Introduction

This year was the first year of the payload team. The payload team researched and searched for appropriate payload for the project. The team also searched for a similar mission to ours in order to design the specific features of our payload.

In Section II we present our Payload constraints. Section III deals with our Call for Scientific Payloads. In Section IV we present some Mission Objectives proposals and compare them. Section V summarizes and concludes our work.

3.2 Payload constraints

At the beginning of the year, no scientific mission or payload was selected.
Therefore, we defined the maximum constraints of the payload in order to facilitate the work of the other teams:

Max mass	1.5 [kg]
Max power	30 [watt]
Envelope Dimensions	D: 110 [mm], L: 150 [mm]
Min Temperature	0 [°C]
Max Temperature	70 [°C]

Table 2: Payload constraints

These constraints can change according to the requirement after payload is selected. In addition, we have defined the position of the payload in the satellite:

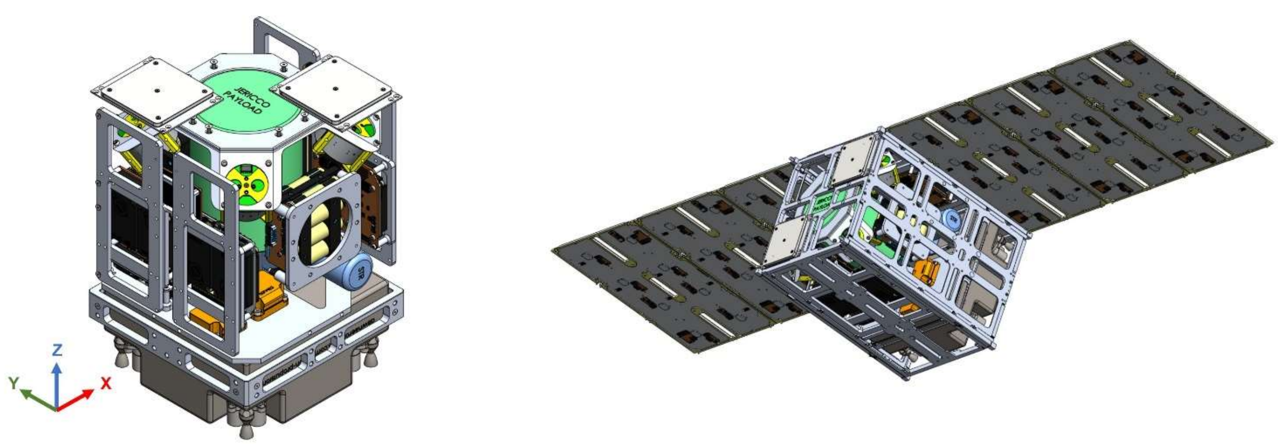


Figure 7: Position of the payload

3.3 Call for Scientific Payloads

After defining the constraints, we published a call for scientific payloads in order to receive proposals for collaborations. Published with the help of Aerospace Faculty professors and it also reached other faculties at the Technion.

JERICCO

Call for Scientific Payloads Onboard JERICCO Mission

Overview

JERICCO (**J**OINT-**E**FFO**R**T **I**NNOVATIVE **C**UBESAT FOR **C**ISLUNAR **O**RBIT) is a student-designed lunar satellite being developed jointly by the Aerospace Department of the Technion (Israel Institute of Technology) and IAI (Israel Aerospace Industries), expected to be launched to Low-Lunar-Orbit in the coming years for scientific missions.

Cooperation Opportunitis

The lunar orbiter provides a great opportunity to conduct scientific research and commercial endeavors. The JERICCO satellite will be equipped with a payload that will define the scope of its mission. The planned lunar orbit is 200 km altitude, 86° inclination, and a flight duration of 18 months.

If necessary, it will be possible to examine certain changes in the lunar orbit parameters.

Design Constraints on Payloads onboard lunar

- Total Mass shall be **1.5 kg Max**.
- Total Power consumption shall be **30 Watt Max**.
- Available Size of **∅110mm x 150mm (cylinder shell)**.
- Withstanding in a Temperature Range of **0°C to 70°C**.

Call for Payload Proposals

We are looking for a partner interested in using our platform for Lunar scientific missions. Please describe the suggested payload and identify a contact person.

Contact us

Yuval Sharifi - yuvalsharifi@campus.technion.ac.il

Dr. Oded Golan - oded.golan@technion.ac.il

Figure 8: Call for Scientific Payloads

3.4 Mission Objectives proposals

In this section we will present Mission Objectives proposals.

3.4.1 SCS- Selfie Camera of Space

The SCS is a system that enhances the mission's ability to perform remote supervision and inspection and improves the way we control electronic systems on deployed satellites, UAVs, or planes. The SCS is a mission flight proven TRL9 Space heritage system that captures both high resolution images and HD video streams. A control card can operate simultaneously up to 3 cameras. A lighting system is integrated in each camera, allowing capturing images in any light condition, including complete darkness.

Figure 9: SCS camera

Camera data:

Sensor	
Sensor Type	VIS
Sensor Technology	CMOS
Resolution	10M Pixel
Active Pixels	3664 X 2748
Lens	
Focal Length [mm]	3.8
FOV (Horizontal)	75°
FOV (Vertical)	64°
Distortion	<2.8%
Lightning	
Viewing Angle	120°
CCT [K]	4000
Electronic	
Power Consumption Standby mode [W]	1W
Power Consumption Acquisition Mode with LED [W]	1.85W
Power Requirements	3.3VDC
Dimensions	
Camera [mm]	36 X 21 X 33
Camera Weight [grams]	56
Control Card [mm]	58 X 92

Table 3: SCS data.

It can be seen that the resolution of the camera does not match our requirements, however, the desired resolution depends on the task, and it may be possible to increase the optical apeture, depending on the desired image quality setting.

3.4.2 LUMIO Cam

Lunar Meteoroid Impacts Observer:
LUMIO is a mission designed to observe, quantify, and characterize the meteoroid impacts by detecting their flashes on the lunar far side. Earth-based lunar observations are restricted by weather, geometric, and illumination conditions, while a lunar orbiter can improve the detection rate of lunar meteoroid impact flashes, as it would allow for longer monitoring periods.

LUMIO Cam:
The main payload of LUMIO. Its purpose is to observe the light flashes produced by meteoroid impacts on the Moon far-side. The impact flashes on the Moon can be modelled as black body emissions with temperatures between 2700 and 6000 K, and durations greater than 30 m/s. The lowest impact energies correspond to apparent magnitudes higher than +6 as seen from Earth.

Data about the camera:

Detector The baseline detector is the CCD201 of e2V L3Vision™	
Image area	13.3 x 13.3 mm
Active pixels	1024 x 1024
Pixel size	13.3 x 13.3 µm
Storage area	13.3 x 13.3 µm
Low noise gain	1-1000
Readout frequency	~15 MHz
Charge handling capacity	80 Ke$^-$/pixel
Readout noise	<1 e-rms
Optics	
field of view of 5.6º is necessary to always have the Moon full disk view. To compensate for pointing errors, a 6º FOV is considered with a 127 mm focal length. The LUMIO-Cam optics features are: FOV=6.0º, focal length = 127 mm, aperture= 55 mm.	
Mechanical layout	
The mechanical layout includes a mechanical barrel supporting five lenses, an entrance baffle for out-of-field straylight reduction, a focal plane assembly, a proximity electronics box, and an external box for mechanical protection.	

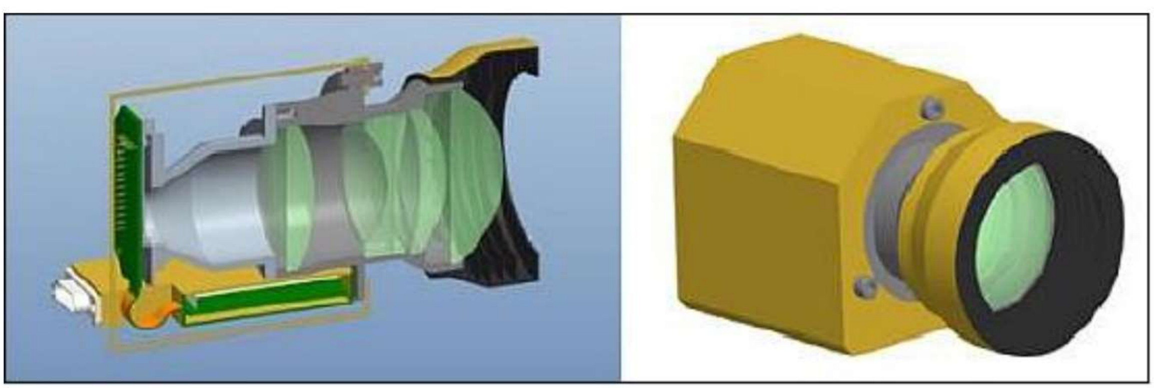


Figure 10: Mechanical layout

On-board payload data processing
• For an acquisition rate of 1.8 MB images at 15 fps, the data products of the payload would be around 2.4 TB/day of science acquisitions. • The on-board payload data processing (OBPDP) detects flashes in the images and stores only the images with scientific relevance. Also, the OBPDP cuts everything outside an area around the flash. • In this way, from 35.7 TB gathered during a LUMIO orbit period (~14.7 days), 13 MB of data needs to be stored.

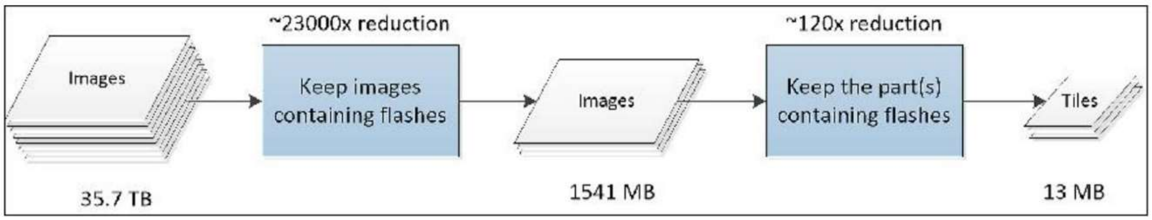


Figure 11: Data amount reduction

Table 4: LUMIO- Cam data.

In addition, we wanted to check the budget of the LUMIO- Cam and compare it to our constraints:

	Payload constraints	LUMIO- Cam
Max mass	1.5 [kg]	1.3 [kg]
Max power	30 [watt]	3.5 [watt]
Envelope Dimensions	D: 110 [mm], L: 150 [mm]	
Min Temperature	0 [°C]	
Max Temperature	70 [°C]	

Table 5: LUMIO- Cam VS our payload.

As you can see, some data is missing. We tried to reach out to LUMIO members but did not get an answer.
It is recommended that the third-year team try to contact the LUMIO members again[1].

3.4.3 Estimation Using Blurred Star Images

In this method we are not looking for a unique payload for our satellite but use the star tracker selected for the satellite.

[1] LUMIO website: https://www.lumio.com/contact/

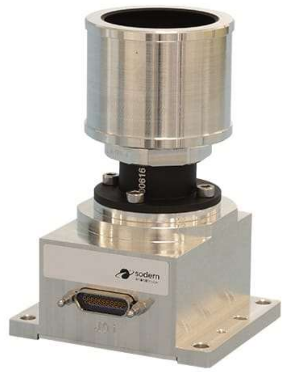


Figure 12: Our star tracker- AURIGA CP.

Possible mission:
Estimation Using Blurred Star Images is a mission designed to provide clear star images for star point centroid determination, star pattern recognition, and angular velocity estimation.

In the article dealing with this method, the requirements of the star tracker were defined. The comparison of the star tracker in the article with our star tracker is as follows:

	From the article	Sodern Auriga-CP
Pixel image plane	1024x1024	
The focal length	43.6 [mm]	
FOV	20x20 [deg]	
The sampling freq.	2 [Hz]	10 [Hz]
the sensitivity limit of the star sensor	5.5 [Mv]	

Table 6: Sodern Auriga- CP Known Elements.

As you can see, some data is missing. We tried to reach Sodern but did not get an answer.
It is recommended that the third year team try to contact the developers again[2].

Pay attention, this method may not require a payload, but it may require a stronger computer, and this should be taken into account if we choose this method.

3.4.4 Hybrid Optical Navigation by Crater Detection for Lunar Pin-point Landing

Trajectories from Helicopter Flight Tests
Accurate autonomous navigation capabilities are essential for future lunar robotic landing missions with a pin-point landing requirement, since in the absence of direct line of sight to ground control during critical approach and landing phases, or when facing long signal delays the herein before mentioned capability is needed to establish a guidance solution to reach the landing site reliably.

The crater navigation method is the only source of absolute position measurements supporting the navigation filter during these flight experiments. Crater detections within images of a navigation

[2] Sodern contact: https://satsearch.co/

camera, matched to a database of known craters within the world reference frame, allow the straightforward solution for the pose of the capturing camera w.r.t. the global reference frame and, by extension, of the vehicle to which it is fixed. Most crater navigation schemes suggested in the literature function in a prediction-match-update configuration, where the burden of matching detected image craters to known database craters is removed by providing prior knowledge of the vehicle pose, enabling direct visual matching within the image space: Prior pose knowledge allows projecting the crater database into the image, where proximity and size matching of the projections to the detections can then be performed.

Trajectory and flight apparatus- payload
The job of transporting the navigation payload was performed by an unmanned SwissDrones (SDO 50 V2) helicopter. This platform is capable of autonomous, assisted and remote-controlled flight and it offers a payload capability of approximately 50 kg (fuel plus experiment equipment). All sensors were integrated on one single platform.
A Tactical-grade IMU (iMAR iTraceRT-F400-Q-E) was used for acquiring velocity and angle increments. It operated at 100 Hz.
Capturing of images was performed by a monocular, monochromatic camera (Prosilica GT1380). its resolution was set to 1024 px x 1024 px. The camera images served as input to two algorithms, a high-framerate feature tracker and the lower-framerate CNav algorithm.
For supplying the high-framerate task, the camera was triggered at 20 Hz. The asynchronous crater navigation was always provided the most recent image at a rate ranging from 3 to 5 Hz, depending on the individual image processing load.
A laser altimeter was installed adjacent to the camera, pointing in parallel with the camera boresight. Measurements from this sensor, although not included in the hybrid navigation set-up here proposed (nor in the ground truth generation).

3.5 Future Work

Future work of the payload team is to choose both the scientific mission of our satellite and the payload with which we will perform it. After selecting the payload, the team will have to check if it is necessary to change the constraints of the payload. If so, it should be examined whether it is possible to change to the required constraints and, if so, how to do it. In addition, it is recommended that the third-year team try to contact the LUMIO members and Sodern in order to get the missing information about the above tasks.

3.6 References

[1] SCS, Microgic™: https://www.microgic.com/scs/.
[2] LUMIO (Lunar Meteoroid Impact Observer), EoPortal: https://www.eoportal.org/satellite-missions/lumio#lumio-lunar-meteoroid-impact-observer.
[3] Xiaolin Ning, Pingping Chen, Jiancheng Fang and Weiren Wu, "Angular Velocity Estimation Method Using Blurred Star Images for Spacecraft", Feb 2019: https://arc.aiaa.org/doi/10.2514/1.G003777.
[4] Guilherme F. Trigo, Bolko Maass, Hans Kr¨uger, Stephan Theil, "Hybrid Optical Navigation by Crater Detection for Lunar Pin-point Landing: Trajectories from Helicopter Flight Tests", https://link.springer.com/article/10.1007/s12567-017-0188-y.

4 Mission and Orbit

4.1 Nomenclature

The following are the convention symbols and their meanings that will be used in this chapter.

a– semi-major axis.

e – eccentricity.

i – inclination.

Ω – RAAN – right ascension of ascending node.

ω – argument of periapsis.

f – true anomaly.

θ – orbit angle from line of nodes ($\theta = f + \omega$).

μ - Standard gravitational parameter.

R_p – Planet radius.

Ω_p – Planet rotational rate $\left(\frac{2\pi}{planetery\ "day"}\right)$.

$J_n, C_{n,m}, S_{n,m}$ – Harmonic perturbation parameter.

4.2 Abstract

Mission and Orbit team (also known as FDS – flight dynamics in the industry) purpose is to design the journey that the satellite will take in space throughout its lifetime, from the separation stage from the carrier spacecraft, onto the main mission phase and to the final moments before the satellite is decommissioned.

With the mission dictated by the payload in mind, the MO team designed the most suitable orbit for the mission, with emphasis on fuel requirements, since the bulk of the fuel will be used for orbit maintenance and executing maneuvers.

In this work we present how we developed orbit maintenance methods using professional tools such as GMAT. Additional work was done to create relatively accurate orbit propagator for the complete satellite system simulation.

4.3 Introduction

Mission and Orbit team is responsible for the design of the satellite trajectory throughout its lifetime, with a focus on the design and maintenance methods of the orbit for the main mission phase.

This year the team focused on two aspects of the mission orbit:

- Orbit analysis using professional tools.

- High accuracy orbit simulation for system simulation.

We used GMAT, a free to use satellite simulation tool developed by NASA, to study how the mass perturbation of the Moon, and third body perturbations from the Earth and Sun affect the orbit of the satellite. We implemented an orbit maintenance method into GMAT to study how the orbit tolerances affect the fuel consumption and overall possible mission phase time. Additionally, using GMAT, we extruded data that can be used on other simulations like Sun and Earth positions for attitude control, or ground station contact simulation for communication time analysis.

We implemented methods of producing equations of motion using Hamiltonian mechanics and Poisson brackets to create short term high accuracy simulation for system simulation. We used the harmonic geopotential model to create the mass perturbation Hamiltonians, and we considered the rotation of the moon by introducing time itself into state space. We used Maple to calculate the Hamiltonians and the equations of motion.

We provide with the report itself the GMAT scripts and Maple files that were developed throughout the year. Additionally, we provided a beginners guide for the GMAT scripts written this year.

4.4 Requirements

The orbit of the satellite is defined by the mission's payload itself. For example, should an optic camera payload be selected, there is an optimal altitude for surveying a specific area with some specific resolution. Since a payload is yet to be determined, the chosen orbit is not final, but nonetheless, the work is done as if it was finalized, and the calculations could be used as a basis for when the payload is to be chosen.

The following table contains the orbit definition and requirements.

Number	Description	Values	Notes
1	Inertial frame of reference	T.O.D	True of Date
2	Nominal altitude	$200\ [km]$	Circular orbit
3	Nominal inclination	$86°$	-
4	Nominal RAAN	$25°$	Not final
5	Altitude tolerance	$\pm 20\ [km]$	Not final
6	Inclination tolerance	$\pm 5°$	Not final
7	RAAN tolerance	-	No tolerance was set

Table 7 – Orbit requirements

Usually, there are some restrictions on total ΔV required to maintain the orbit. However, since the orbit tolerance affects the required ΔV, and it was not finalized, there is no restriction on the total ΔV. Some restriction that could be listed are limitations of the available fuel resulted from maneuvers that places the satellite into orbit in first place (may consume relatively large amounts of fuel), or limitations that are due to the finalized propulsion system. In any case, as stated before, the calculations made assume that the current requirements are used as basis for future analysis.

4.5 Orbit Maintenance

Using GMAT as will be explained later, we explored how the orbit of the satellite evolves naturally under the influence of the gravity of the Moon, the Earth, and the Sun, with the Moon considering up to order 165 of mass perturbation. The results are shown in the following figures:

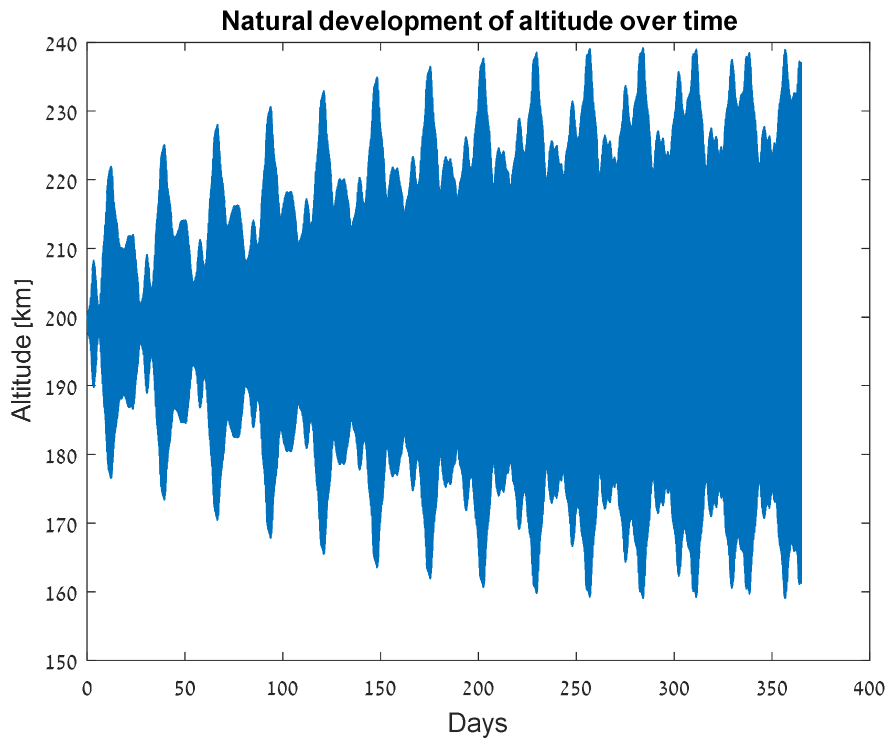


Figure 13 – Natural development of the altitude over one year

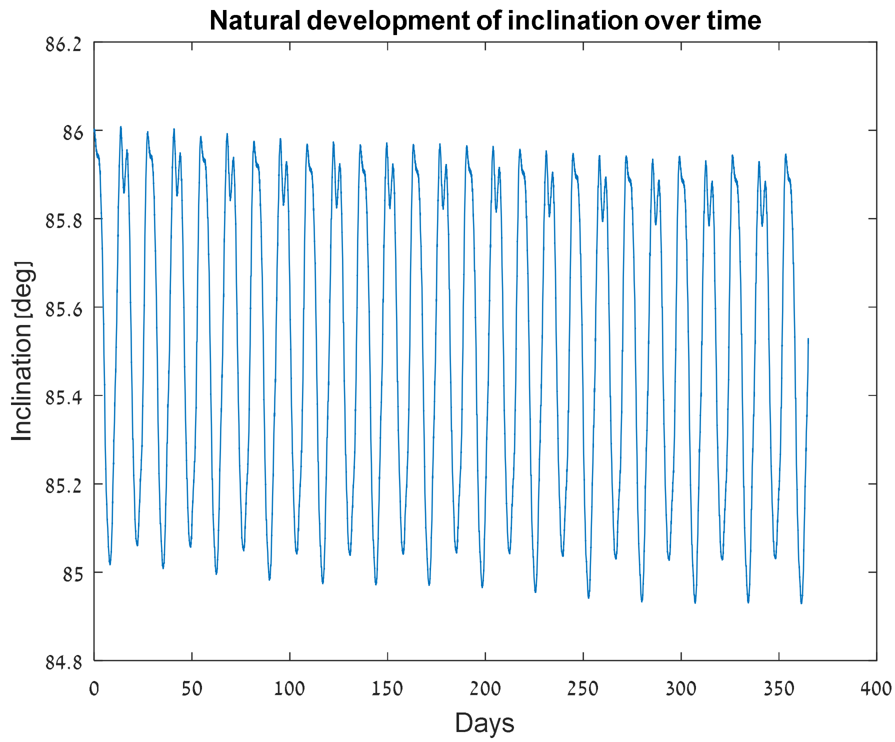


Figure 14– Natural development of the inclination over one year

As can be seen by these figures, the chosen orbit mostly maintains its inclination angle. However, the altitude varies symmetrically, meaning while the semi-major axis remains as is, the eccentricity overall increases. This behavior needs to be controlled via orbit maintenance maneuvers.

4.5.1 Altitude Maintenance

Altitude maintenance is done using Hohmann maneuver (as we learned on Fundamentals of Space Engineering course). Assume that at some time, the orbit deviates from its circular shape by the following:

- Δh_{min} – lower altitude deviation, meaning that the altitude of the periapsis is

  $h_{peri} = R_0 + \Delta h_{min}$, with $\Delta h_{min} < 0$.
- Δh_{max} – upper altitude deviation, meaning that the altitude of the periapsis is

  $h_{apo} = R_0 + \Delta h_{max}$, with $\Delta h_{max} > 0$.

The maneuver will be executed by two pulses, executed first at the periapsis, and then at the apoapsis, with the following values, respectively:

$$\Delta V_{periapsis} = \sqrt{\frac{2\mu}{R_0 + \Delta h_{min}}}\left(\sqrt{\frac{R_0}{2R_0 + \Delta h_{min}}} - \sqrt{\frac{R_0 + \Delta h_{max}}{2R_0 + \Delta h_{min} + \Delta h_{max}}}\right) \qquad (1)$$

This will negate the deviation of the apoapsis, meaning having $h_{apo} = R_0$.

$$\Delta V_{apoapsis} = \sqrt{\frac{2\mu}{R_0}}\left(\sqrt{\frac{1}{2}} - \sqrt{\frac{R_0 + \Delta h_{min}}{2R_0 + \Delta h_{min}}}\right) \quad (2)$$

This will negate the deviation of the periapsis, meaning having $h_{peri} = R_0$.

And by that, we have transformed the elliptic orbit back into circular.

The following flow chart depicts the altitude maintenance method:

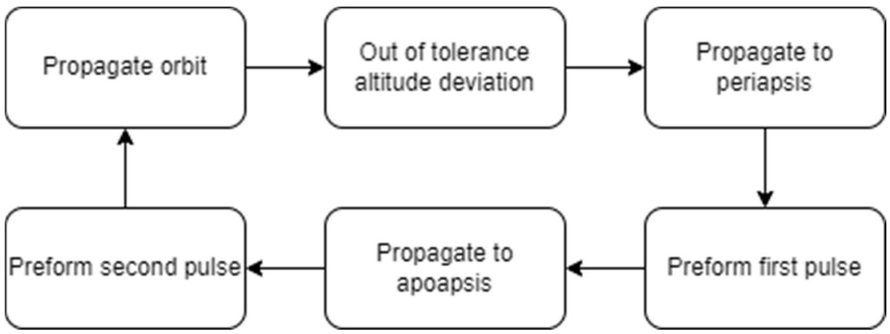


Figure 15 – Altitude maintenance flow chart

Based on equations (1) and (2) and under the assumption that only the eccentricity changes, meaning $|\Delta h_{min}| = |\Delta h_{max}|$, the following graph shows the total ΔV required for a single altitude correction maneuver. Specifically for altitude of $200\ [km]$ and tolerance of $20\ [km]$, the required ΔV is around 8.2 $\left[\frac{m}{s}\right]$ per maneuver.

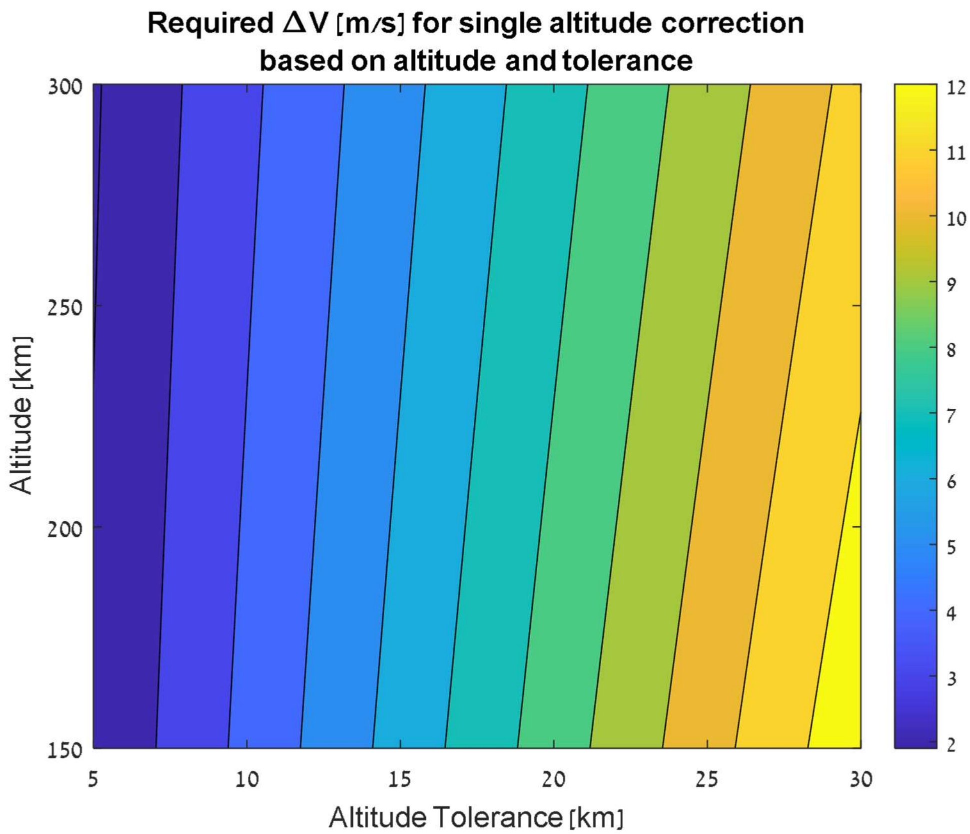


Figure 16 – Required ΔV for a single altitude correction based on altitude and tolerance.

A simulation for possible mission time was made in GMAT. Using the logic shown in Figure 15 and equations (1) and (2), and the following mass and propulsion data:

- Dry mass of 13.2 kg.
- Fuel mass of 1.5 kg.
- ISP of 285 sec.

The simulation ran a mission until total fuel consumption. The following graph shows the effect of altitude tolerance on mission time:

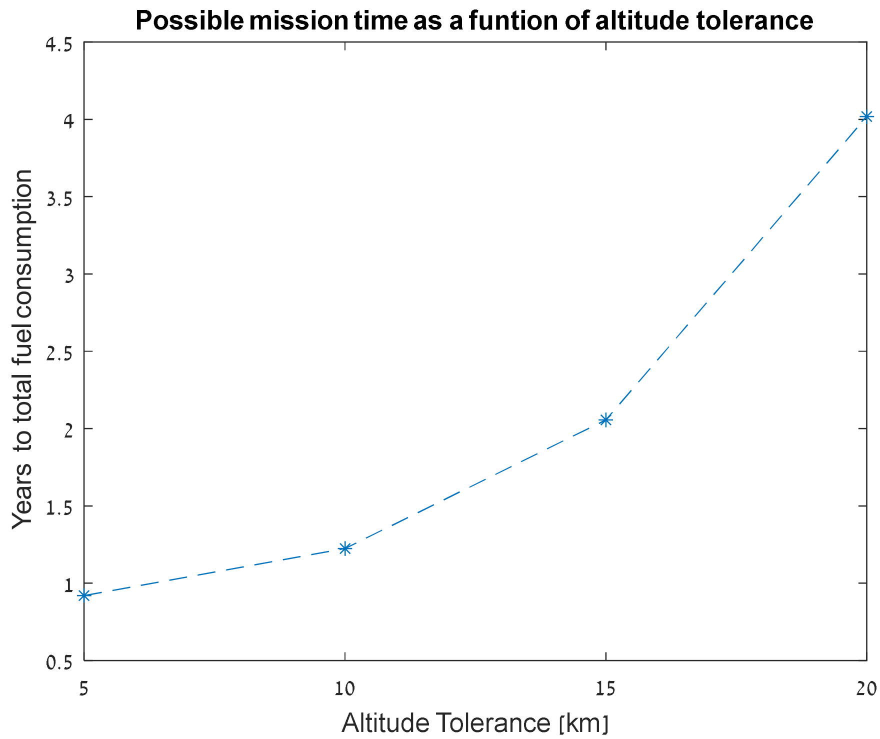


Figure 17 – Mission time as a function of altitude tolerance

It is clear that even though higher tolerance requires higher ΔV (as shown in Figure 16), and by that higher ISP propulsion, it is preferred for the total fuel economic.

4.5.2 Inclination Maintenance

Even though, as it was shown, the inclination angle is relatively stable, there is the possibility of needing to perform a single pulse inclination correction maneuver. In that case, the Gauss Variational Equation (GVE) provide the required relation between single pulse maneuver and the change in orbital elements, specifically for the case of the inclination:

$$\Delta i = \Delta V_h \frac{r}{h} \cos\theta \qquad (3)$$

Where here h is the magnitude of the angular momentum and ΔV_h is a pulse given in the direction of the angular momentum. It is clear that in order to minimize the use of fuel (minimum ΔV_h) a pulse needs to be applied at the nodes where $\cos\theta = \pm 1$.

For the worst-case scenario, we will consider the case that the orbit is in its circular shape but on a different plane, shifted by 5° (max tolerance), in that case, the required ΔV for the maneuver is:

$$\Delta i = \Delta V_h \sqrt{\frac{R_0}{\mu}} \rightarrow \Delta V_h = \Delta i \sqrt{\frac{\mu}{R_0}} \rightarrow \Delta V_h = 54.8 \left[\frac{m}{s}\right] \qquad (4)$$

4.5.3 RAAN Maintenance

The RAAN angle is not final, even for the chosen orbit.

The first reason we decided not to maintain it as of now is that the natural drift of it allows more coverage of moon surface and as byproduct also more fuel economic. By means of simulation we saw that even one month period is probably enough to cover the whole moon surface with some kind of camera payload.

The second reason for not studying maintaining or controlling the RAAN angle is for communication purposes. It is possible that we will use another satellite that orbits the Moon, and for maximum communication efficiency, the orbital planes of the satellites need to be co-planar. Meaning either letting them drift naturally somewhat the same or maintaining them the same all the time.

Overall, up until a requirement is made, the RAAN angle will not be maintained and will evolve naturally.

4.6 Orbit Simulation Using GMAT

This year we have decided to simulate the orbit using GMAT by NASA. 3 simulation scripts were created:

1. Orbit development – orbit propagation program without orbit maintenance, used to study how the orbit develops naturally.
2. Orbit maintenance - orbit propagation program with orbit maintenance, used to study orbit maintenance methods.
3. Earth, Sun location – used only to create Earth and Sun locations (vectors) relative to the inertial frame for the use of other project groups.

A more detailed explanation of each of the simulations, and basics tips for using GMAT for beginners can be found on the corresponding appendix.

4.7 Orbit Simulation In MATLAB

Even though we use professional tools like GMAT to simulate the orbit, we cannot (as of now) incorporate in them the complete spacecraft with all of its subsystems and features, therefor part of our team is dedicated to developing as fully as possible system simulation.

To this effort Mission and Orbit team need to provide the equations of motion of the satellite. This part of the report will explain how we did it.

DISCLAIMER: The method discussed in this part is taught on the Astrodynamics course, therefor the assumption is that the reader is familiar with Hamiltonian mechanics, Poisson brackets and canonical variables.

4.7.1 The harmonic geopotential model

Given the latitude angle ϕ and the longitude angle ζ the mass perturbation potential can be represented by:

$$V = -\frac{\mu}{r}\sum_{n=2}^{\infty}\left[\left(\frac{R_p}{r}\right)^n J_n P_n^0(\sin\phi)\right] - \frac{\mu}{r}\sum_{n=2}^{\infty}\sum_{m=1}^{n}\left[\left(\frac{R_p}{r}\right)^n P_n^m(\sin\phi)\left(C_n^m\cos(m\zeta) + S_n^m\sin(m\zeta)\right)\right] \tag{5}$$

Where:

- μ, R_p are the gravitational parameter and planet equatorial radius.
- J_n, C_n^m, S_n^m are the harmonic coefficients.
- P_n^0 is the Legendre polynomial function.

  Can be approximated by:

$$P_n^0(x) = \frac{1}{2^n \cdot n!}\frac{d^n}{dx^n}\left((x^2-1)^n\right) \tag{6}$$

- P_n^m is the associated Legendre polynomial function.

  Can be approximated by:

$$P_n^m(x) = (1-x^2)^{\frac{m}{2}}\frac{d^m}{dx^m}\left(P_n^0(x)\right) \tag{7}$$

This potential will be used to define the orbit governing Hamiltonian.

4.7.2 Polar Nodal Variables

We will define the orbital state space by the following variables:

- r – radial distance.
- θ – the orbital angle from the line of nodes (also known as $\omega + f$).
- ν – the angle of rising node.
- R – radial velocity.
- Θ – the magnitude of angular momentum.
- N – the projection of angular momentum on Z axis.

These of course are the Polar Nodal variables, which are known to be canonical.

The following are useful relations between them and the Cartesian coordinates:

$$[\boldsymbol{R}] = [\boldsymbol{Z}](-\nu) \cdot [\boldsymbol{X}](-i) \cdot [\boldsymbol{Z}](-\theta) \tag{8}$$

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = [\boldsymbol{R}] \begin{bmatrix} r \\ 0 \\ 0 \end{bmatrix} , \quad \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = [\boldsymbol{R}] \begin{bmatrix} R \\ \frac{\Theta}{r} \\ 0 \end{bmatrix} \tag{9}$$

Where:

- $[\boldsymbol{X}](x)$ is rotation around X axis by x degrees.
- $[r \quad 0 \quad 0]^T$ is the position in the PN reference frame and $[R \quad \Theta/r \quad 0]^T$ is the velocity.

For polar nodal coordinates here are some angle relations:

$$\cos(i) = \frac{\mathrm{N}}{\Theta} \triangleq c \rightarrow \sin(i) = \sqrt{1 - \frac{\mathrm{N}^2}{\Theta^2}} \triangleq s \tag{10}$$

$$\sin(\phi) = s \cdot \sin(\theta) \tag{11}$$

For the longitude angle it is more complicated, since the earth's rotation itself changes the longitude angle even for "stationary" satellite, we need to consider the rotation itself.

$$\zeta = \zeta_0 - \Omega_p(t - t_0) \tag{12}$$

Where:

- ζ_0 is the longitude angle in the inertial frame, calculated by $\zeta_0 = \arctan(y, x)$.
- Ω_p is the planet rotational rate.
- t_0 is the epoch time.

From here we can deduce that the mass perturbation potential showed in equation (5) is time dependent, a modification to the State Space is in need.

We will modify the state space to include time by introducing:

- t – time itself.
- T – the "momentum" counterpart of time for canonicity purposes.

And the complete state space will become:

$$\vec{X} = [t \quad r \quad \theta \quad \mathrm{N} \quad T \quad R \quad \Theta \quad \mathrm{N}]^T$$

4.7.3 The equation of motions

The Keplerian motion is governed by the following Hamiltonian (with time in state space):

$$\mathcal{H}_{KP} = \frac{1}{2}\left(R^2 + \frac{\Theta^2}{r^2}\right) - \frac{\mu}{r} + T \tag{13}$$

And so the complete Hamiltonian is:

$$\mathcal{H} = \frac{1}{2}\left(R^2 + \frac{\Theta^2}{r^2}\right) - \frac{\mu}{r} + T - \frac{\mu}{r}\sum_{n=2}^{\infty}\left[\left(\frac{R_p}{r}\right)^n J_n P_n^0(s \cdot \sin\theta)\right] - \frac{\mu}{r}\sum_{n=2}^{\infty}\sum_{m=1}^{n}\left[\left(\frac{R_p}{r}\right)^n P_n^m(s \cdot \sin\theta)(C_n^m \cos(m\zeta) + S_n^m \sin(m\zeta))\right] \tag{14}$$

And the equations of motion are given by:

$$\dot{\vec{X}} = \{\vec{X}, \mathcal{H}\}$$

For the practical approach the equations of motion were developed using MAPLE and were transferred to MATLAB using CodeGeneration feature.

The equations were developed for perturbation of up to 5th order, meaning that $n_{max} = 5$.

Harmonic coefficients were taken from GMAT source files (specifically for the moon we took the LP165 model), however in the source file there were written in their normalized from:

$$J_n/C_n^m/S_n^m = \sqrt{\frac{(n-m)! \cdot (2n+1) \cdot \delta_{0,n}}{(n+m)!}}\bar{J}_n/\bar{C}_n^m/\bar{S}_n^m \tag{15}$$

Where the $\bar{\ }$ is the data given by GMAT source file.

4.8 Appendix – GMAT Scripts and tips

This appendix will explain some of the settings done in the various GMAT scripts made throughout the year and will provide some tips on how to use GMAT in a more user-friendly and efficient way.

GMAT does provide with the program itself a long and detailed user manual with some teaching cases, and even has several YouTube instruction videos. However, **the first tip** is not to read the entire manual end to end or watch the entire tutorial videos, read and watch only about the basic settings and then search for how to do the specific thing you want to do.

The appendix will be divided into each of the scripts, starting with the Orbit Maintenance script which covers all of the settings, and the other scripts will just note the difference with respect to it.

4.8.1 Orbit Maintenance Script

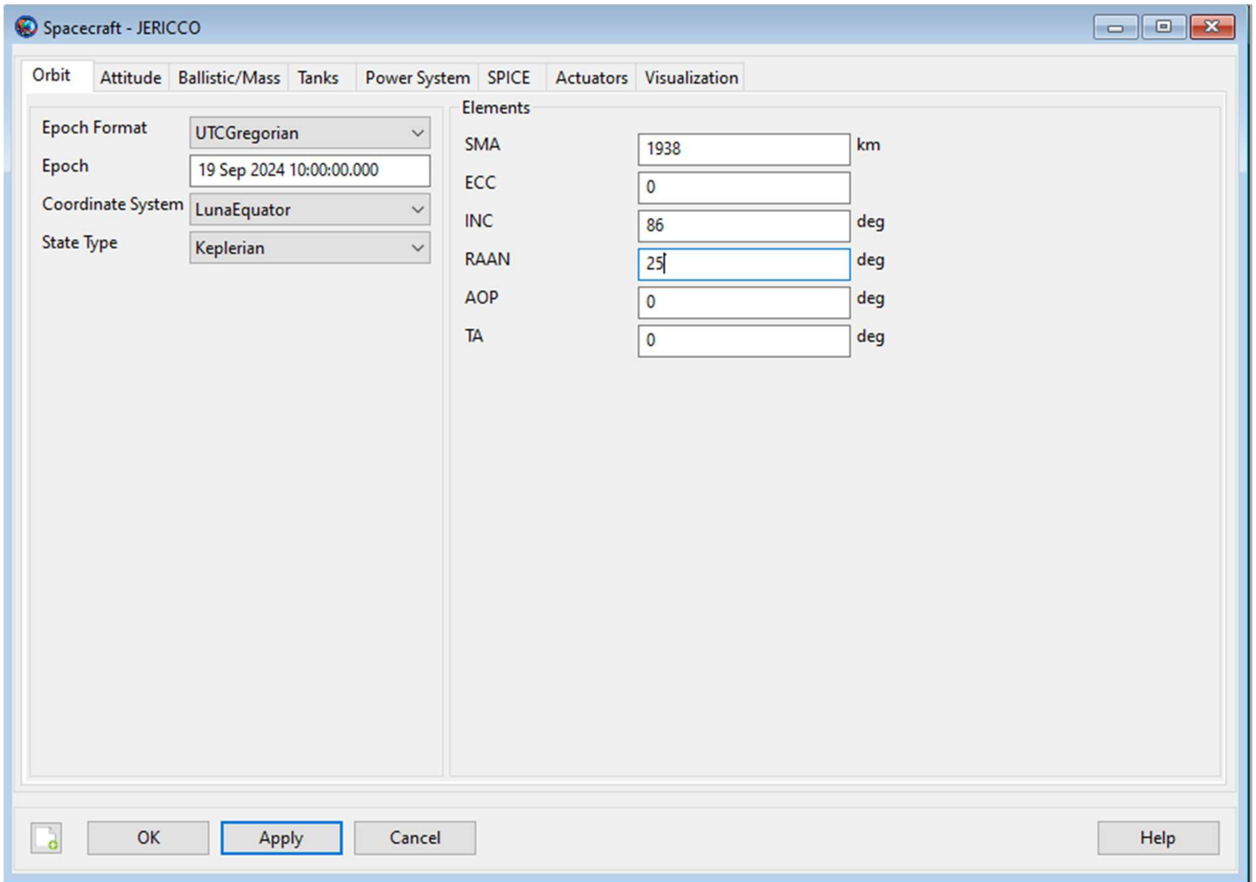


Figure 18 – Spacecraft settings window – Orbit tab

- Epoch Format – sets the epoch time format.
- Epoch – sets the epoch time, for now it is 19.09.2024 10: 00.
- Coordinate System – Chose the coordinate system that relative to it we initialize the orbit.
- State Type – Chose the type of initial condition, we use the Keplerian (Classical orbital elements).
- SMA – semi-major axis.
- ECC – eccentricity.
- INC – inclination.
- RAAN – right ascension of ascending node.
- AOP – argument of periapsis.

- TA – true anomaly.

Note that after you set your elements close the window, when reloading it the elements **will** change, it seems that the program does not allow circular orbits. Do not worry, it will set the eccentricity to be numerically zero (magnitude of about 10^{-15}).

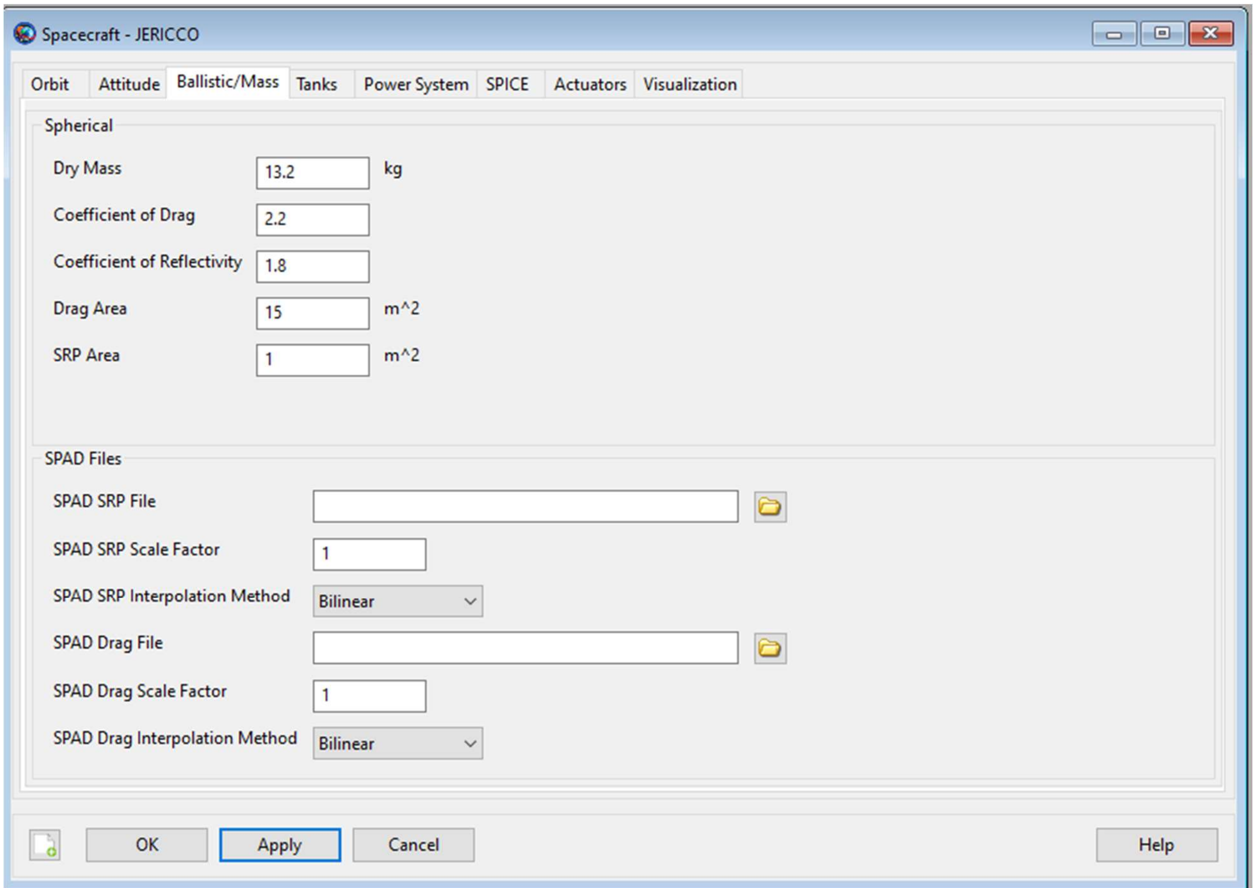


Figure 19 - Spacecraft settings window – Ballistics/Mass tab

The only required definition is Dry mass (mass without fuel), the other settings where not used.

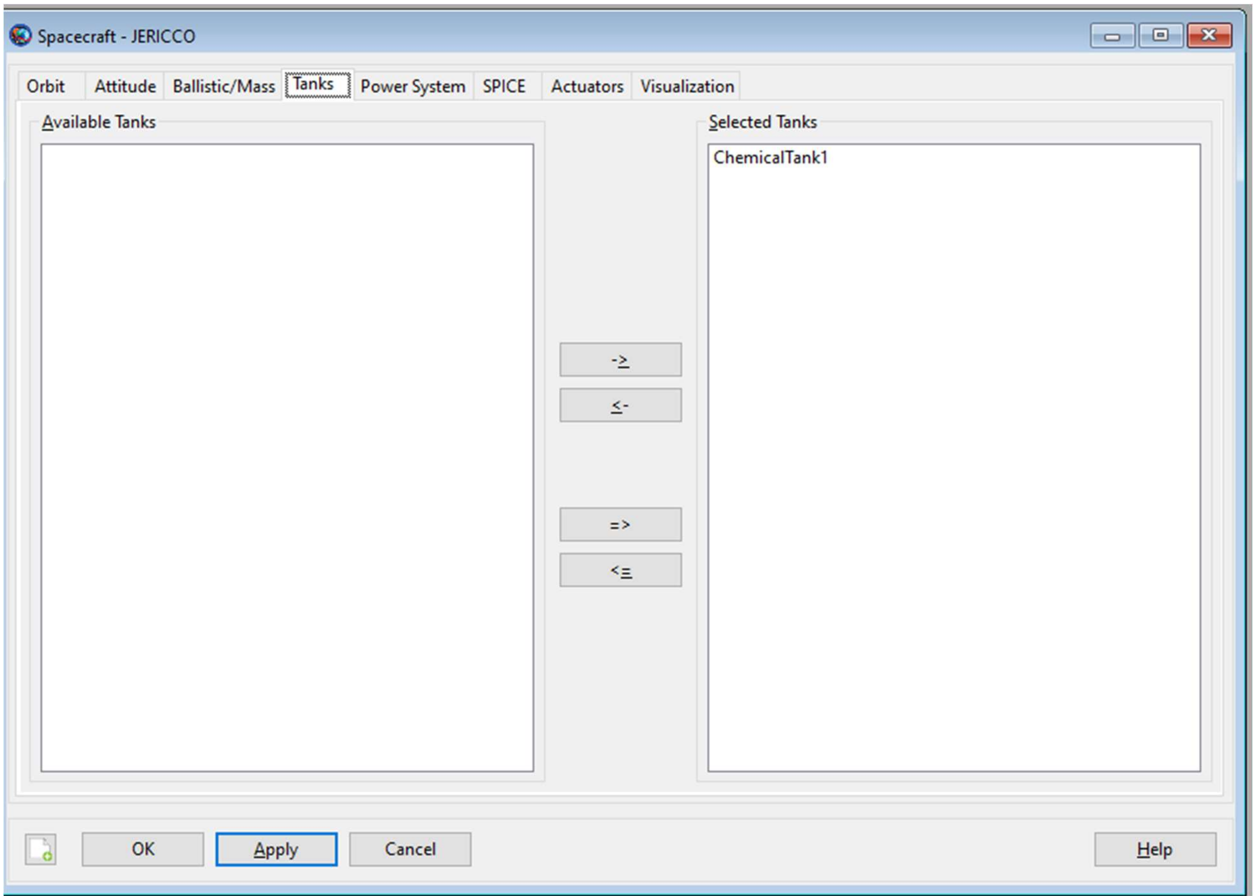


Figure 20 - Spacecraft settings window – Tanks tab

Select the fuel tanks to be used by the spacecraft. Will be available only after the fuel tanks are defined.

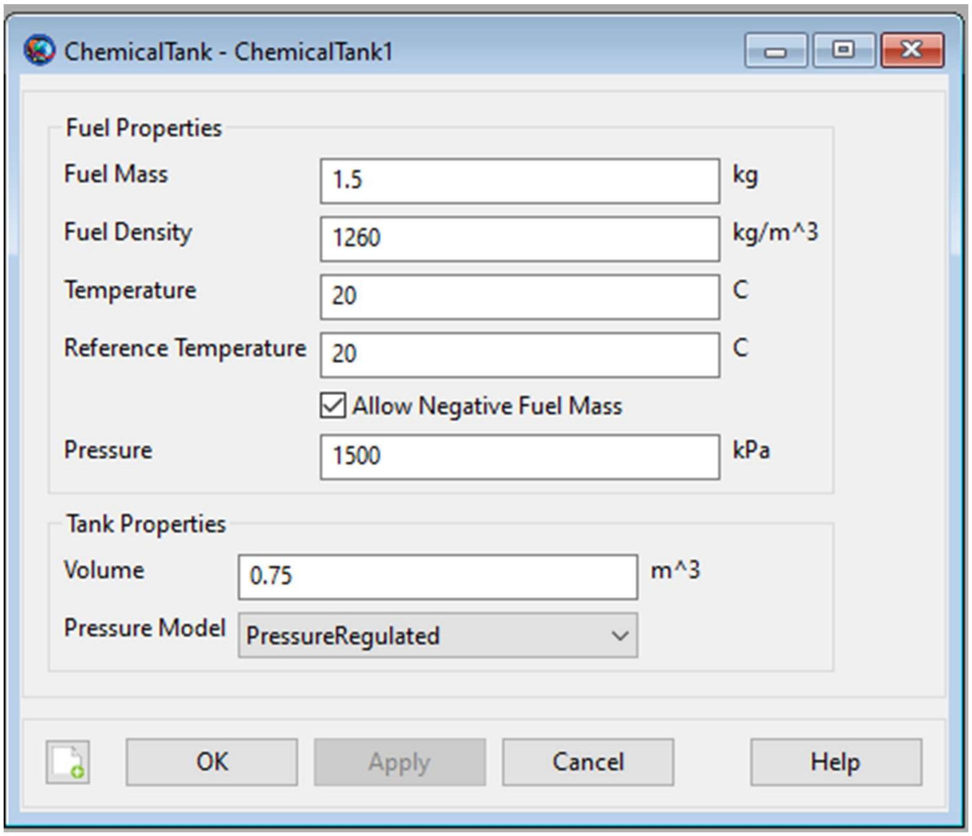


Figure 21 – Hardware setting window – ChemicalTank

When adding a new fuel tank this window opens. As of now only the Fuel Mass setting is required since we simulate ΔV pulses and not more complex maneuvers.

Select also the "Allow Negative Fuel Mass" for the logic of the simulation later shown.

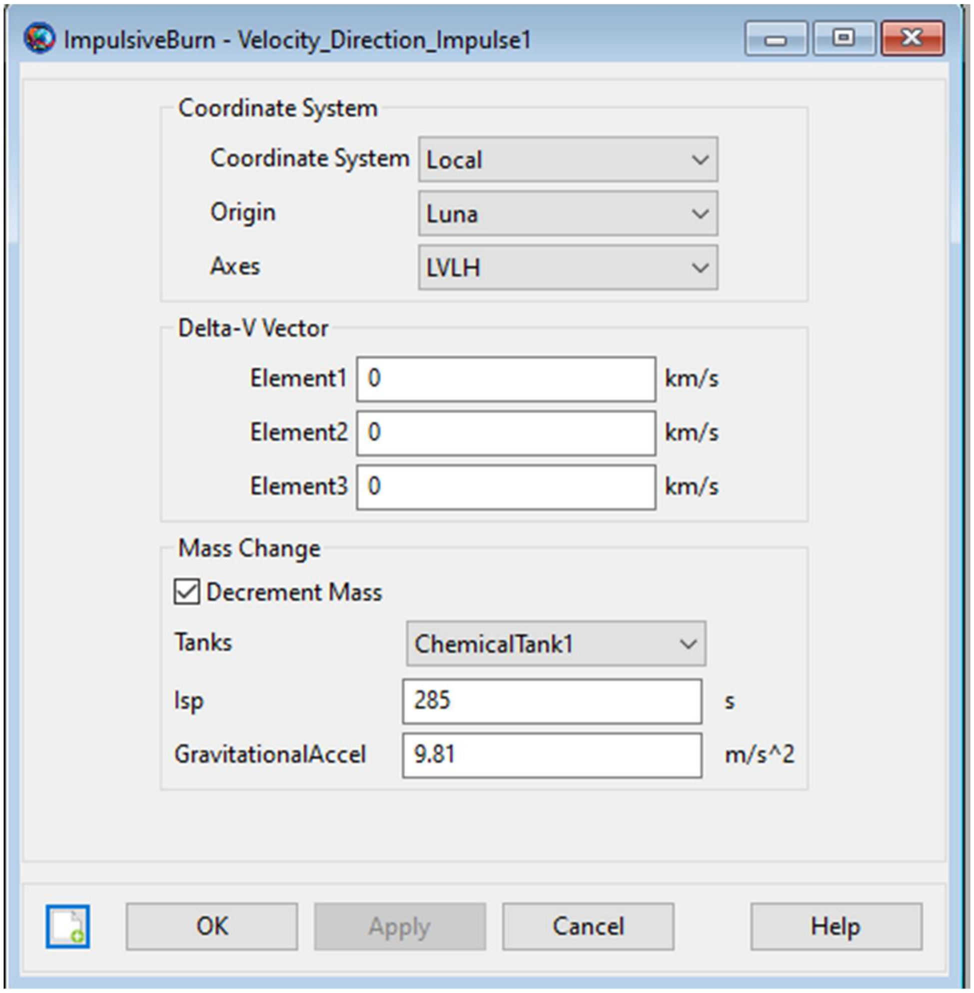


Figure 22 – Burns setting window – ImpulsiveBurn

- Coordinate System – chose how to define the burn (ΔV). For the orbit maintenance we choose local – Luna – LVLH.
- Delta-V Vector – we define the elements in the scripts itself, set them to zero.
- Mass Change – choose "Decrement Mass", the tank that is consumed and ISP and g_0.

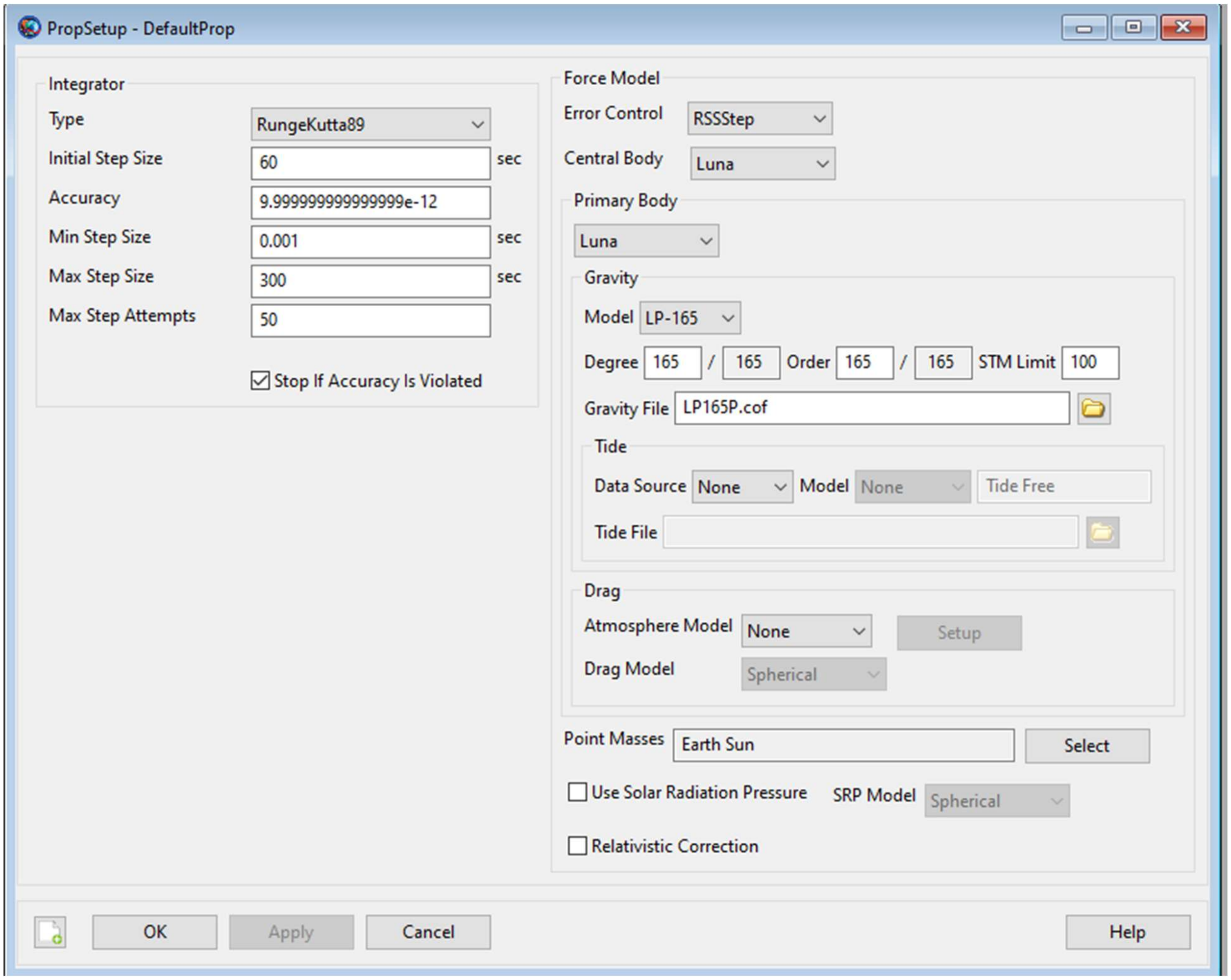


Figure 23 – Propagator setting window.

- Integrator – other than the max step, which is recommended to be lowered since we want to monitor the altitude carefully, leave the defaults as they are.
- Central body and Primary body should be the same.
- Gravity Model – choose any model other than "None" to simulate mass distribution model. Degree and Order should be the same.
- Point Masses – add Earth and Sun for multiple body effects.
- This year we ignored drag and SRP, if it is to be simulated, verify the settings of cross section areas of the satellite and reflecting index.

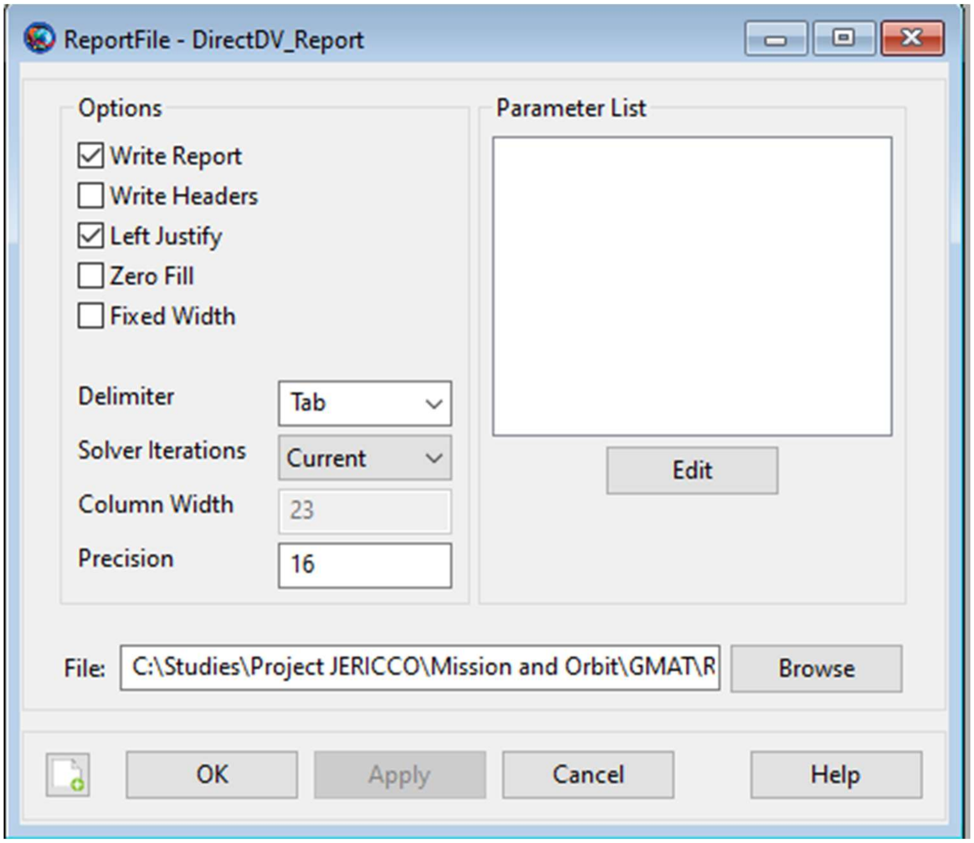


Figure 24 – Report file settings

Report Files are exported to selected directory, so change it when using the script in your own computer.

Keep the option as they are.

The Parameter List is empty because we load them from the script itself with our own variables. If you chose to create a report with predetermined program variables, choose them here.

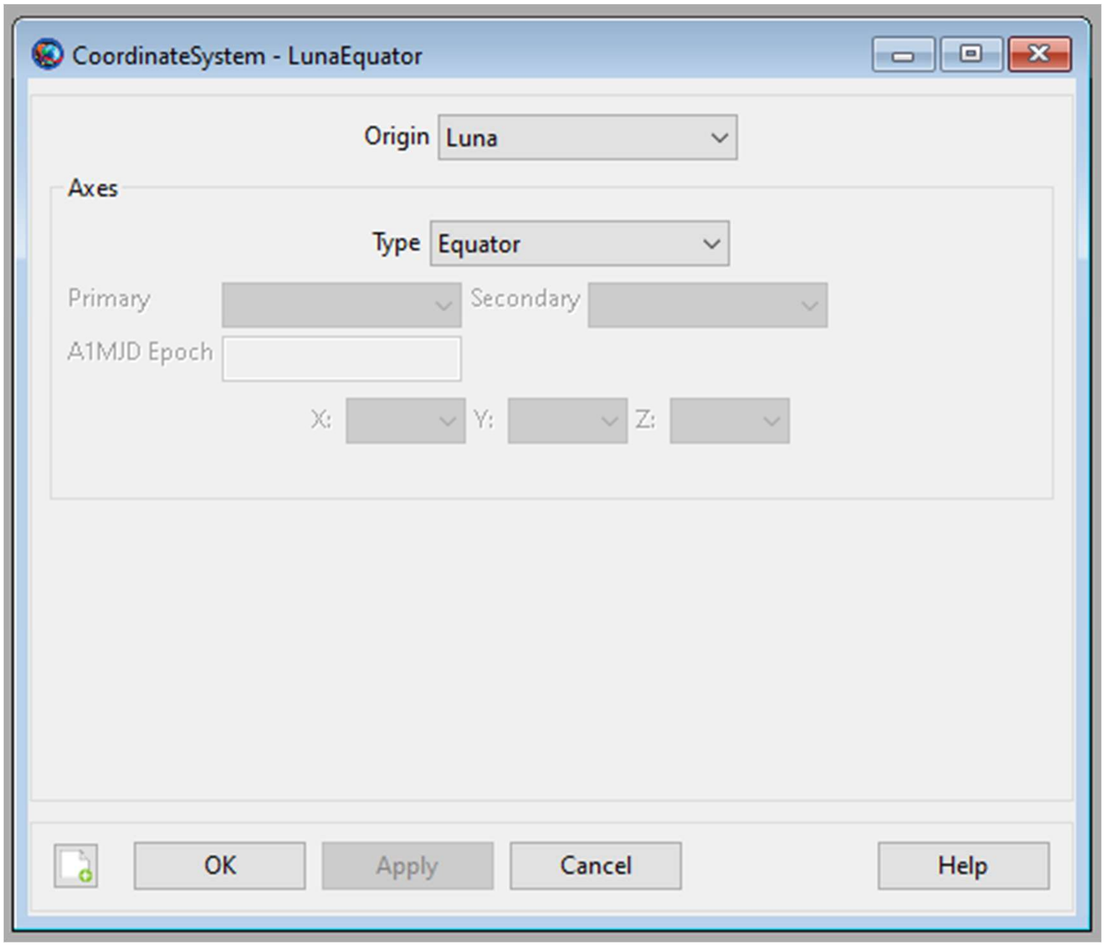


Figure 25 – Coordinate System Setting

The default coordinate systems are Earth oriented. To create the TOD coordinate system, select the origin as Luna and the type as Equator.

Up to now we have created all the required settings in the Resources tab, now we will create the mission simulation itself and the variables mostly directly in the script itself.

First create some random variable in the "Variables/Arrays/Strings" folder in "Resources" tab. After that press the save bottom and open the script. Go to the variables area.

```
%----------------------------------------
%---------- Arrays, Variables, Strings
%----------------------------------------
Create Variable MaxAltitudeDeviation MinAltitudeDeviation AltitudeTolerance SimDayTime DV1tot DV2tot;
Create Variable MoonMu PeriRad ApoRad PeriDh ApoDh NominalSMA Maneuver1_Time Maneuver2_Time FuelMass;
Create String Maneuver1_UTC Maneuver2_UTC Maneuver1_Pos Maneuver2_Pos;
GMAT AltitudeTolerance = 5;
GMAT SimDayTime = 30;
GMAT MoonMu = 4904.8695;
```

Figure 26

We have added the following variables and strings:

- Max/MinAltitudeDeviation – used to set the maximum and minimum allowed altitude.

- AltitudeTolerance – The required altitude tolerance (symmetric).
- SimDayTime – Sets the maximum simulation time in days (tough not used in the shown configuration).
- DV1tot / DV2tot – calculates the magnitude of the pulses.
- MoonMu – the μ of the moon, can be looked up from LP165 file.
- PeriRad/ApoRad – radius of periapsis and apoapsis.
- PeriDh/ApoDh – altitude deviation of periapsis and apoapsis.
- NominalSMA – the initial SMA value.
- Maneuver1/2_Time – the time in seconds from the beginning of the simulation of the pulse.
- FuelMass – current fuel mass.
- Maneuver1/2_UTC – the UTC time of the maneuver.
- Maneuver1/2_Pos – "periapsis" / "apoapsis"

After creating the variables create the general scheme of the simulation in the "Mission" tab. After that go back to the script itself to edit it to the following configuration:

```
%----------------------------------------
%---------- Mission Sequence
%----------------------------------------
BeginMissionSequence;

GMAT Velocity_Direction_Impulse1.Element1 = 0;
GMAT Velocity_Direction_Impulse1.Element3 = 0;
GMAT Velocity_Direction_Impulse2.Element1 = 0;
GMAT Velocity_Direction_Impulse2.Element3 = 0;

GMAT NominalSMA = JERICCO.Luna.SMA;
GMAT MaxAltitudeDeviation = NominalSMA - 1738.2 + AltitudeTolerance;
GMAT MinAltitudeDeviation = NominalSMA - 1738.2 - AltitudeTolerance;

Write MaxAltitudeDeviation MinAltitudeDeviation { Style = Concise, LogFile = false, MessageWindow = true }
```

Figure 27 – Mission script part 1

At the beginning we verify that the elements of the pulses that are not to be used are zero.

After that we calculate the maximum and minimum altitude based on the initial SMA and tolerance.

Next, we print into the message window the maximum and minimum altitudes.

```
While 'Simulating' JERICCO.ChemicalTank1.FuelMass >= 0 %& JERICCO.ElapsedDays <= SimDayTime
   Propagate 'Propagate to max altitude deviation' DefaultProp(JERICCO) {JERICCO.Luna.Altitude = MaxAltitudeDeviation, JERICCO.Luna.Altitude = MinAltitudeDeviation};
   If 'Altitude Deviation' JERICCO.Luna.Altitude >= MaxAltitudeDeviation | JERICCO.Luna.Altitude <= MinAltitudeDeviation

      Propagate 'Propagate to periapsis' DefaultProp(JERICCO) {JERICCO.Luna.Periapsis};
      GMAT PeriRad = JERICCO.Luna.RadPer;
      GMAT ApoRad = JERICCO.Luna.RadApo;
      GMAT PeriDh = PeriRad - NominalSMA;
      GMAT ApoDh = ApoRad - NominalSMA;
```

Figure 28 – Mission script part 2

We enter a while loop to continue the mission as long as we have fuel, that's why we allowed negative fuel mass. Note that the SimDayTime condition is commented, which does not limit the simulation total time.

Next, we propagate the orbit as long as we are in the altitude tolerance defined by the maximum and minimum altitudes. For a more complex mission simulation other conditions may be added.

The ״if״ statement is unnecessary but used for ״good״ measure.

As for the altitude method explained in the report, we propagate to the periapsis and calculate the deviations in altitude.

```
GMAT Velocity_Direction_Impulse1.Element2 = sqrt(2*MoonMu/(NominalSMA+PeriDh))*(sqrt(NominalSMA/(2*NominalSMA+PeriDh))-sqrt((NominalSMA+ApoDh)/(2*NominalSMA+ApoDh+PeriDh)));
GMAT Velocity_Direction_Impulse2.Element2 = sqrt(2*MoonMu/NominalSMA)*(sqrt(1/2)-sqrt((NominalSMA+PeriDh)/(2*NominalSMA+PeriDh)));

GMAT 'Record DV1' DV1tot = sqrt((Velocity_Direction_Impulse1.Element1)^2+(Velocity_Direction_Impulse1.Element2)^2+(Velocity_Direction_Impulse1.Element3)^2);
GMAT 'Record DV2' DV2tot = sqrt((Velocity_Direction_Impulse2.Element1)^2+(Velocity_Direction_Impulse2.Element2)^2+(Velocity_Direction_Impulse2.Element3)^2);
```

Figure 29 – Mission script part 3

We calculate the required DV for each pulse and the ״total״ DV (for this case in practice it is the absolute value).

```
Maneuver 'DV1 Maneuver' Velocity_Direction_Impulse1(JERICCO);
GMAT 'Record DV1 UTC Greogorian' Maneuver1_UTC = JERICCO.UTCGregorian;
GMAT 'Record DV1 Elapsed Seconds' Maneuver1_Time = JERICCO.ElapsedSecs;
GMAT 'Record DV1 Location' Maneuver1_Pos = 'Periapsis';
GMAT 'Record Remaining Fuel Mass' FuelMass = JERICCO.ChemicalTank1.FuelMass;

Report 'Report DV1 recorded data' DirectDV_Report Maneuver1_Pos Maneuver1_Time Maneuver1_UTC PeriRad ApoRad Velocity_Direction_Impulse1.Element2 FuelMass;

Propagate 'Propagate to apoapsis' DefaultProp(JERICCO) {JERICCO.Luna.Apoapsis};
Maneuver 'DV1 Maneuver' Velocity_Direction_Impulse2(JERICCO);
GMAT 'Record DV2 UTC Greogorian' Maneuver2_UTC = JERICCO.UTCGregorian;
GMAT 'Record DV2 Elapsed Seconds' Maneuver2_Time = JERICCO.ElapsedSecs;
GMAT 'Record DV2 Location' Maneuver2_Pos = 'Apoapsis';
GMAT 'Record Remaining Fuel Mass' FuelMass = JERICCO.ChemicalTank1.FuelMass;

Report 'Report DV2 recorded data' DirectDV_Report Maneuver2_Pos Maneuver2_Time Maneuver2_UTC PeriRad ApoRad Velocity_Direction_Impulse2.Element2 FuelMass;
```

Figure 30 – Mission script part 4

We execute the first pulse and record the time and fuel left, then report everything into the report file.

Next, we propagate to the periapsis, execute the second report, record time and fuel left and report everything into the report file.

And thus the cycle continues until there is no fuel left in the tanks.

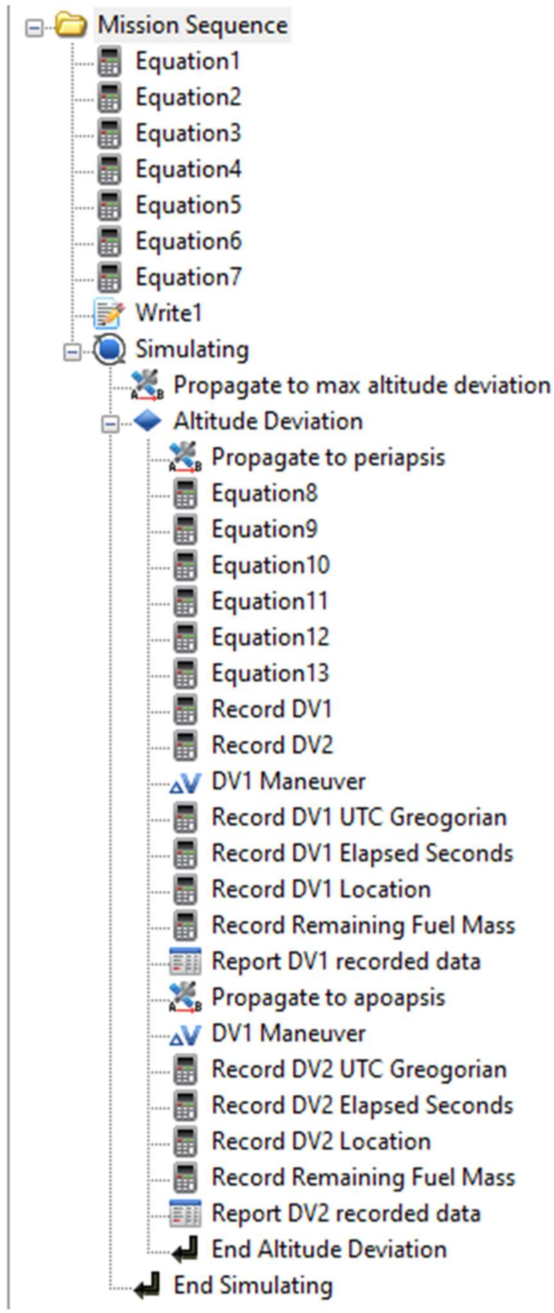


Figure 31 – Mission configuration

In the Mission tab of the main program the mission sequence looks like that.

4.8.2 Earth and Sun location

For this simulation we care only about the Earth and the Sun locations, so the propagator is degenerate.

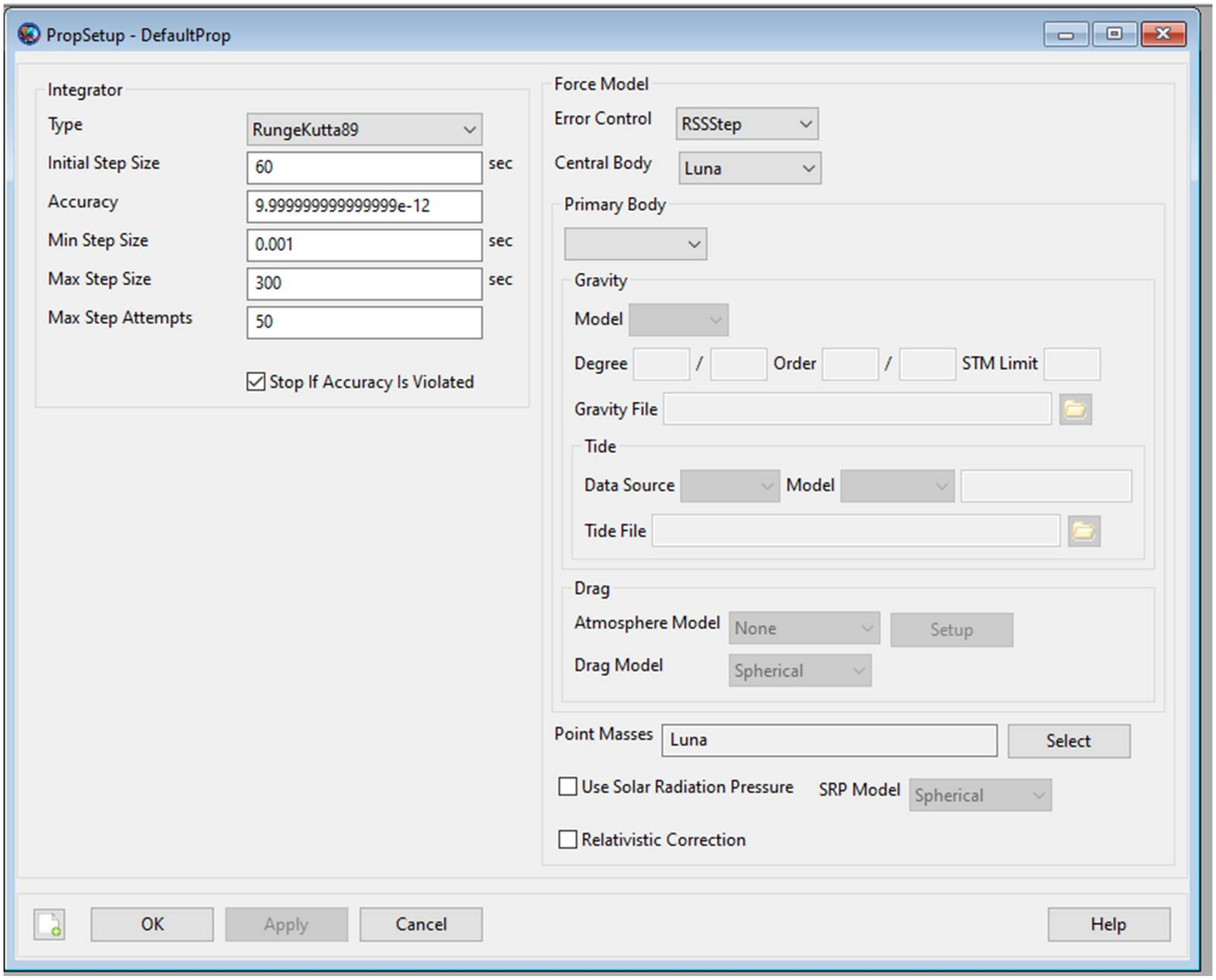


Figure 32 – Degenerate orbit propagator settings.

And the report file are as follows:

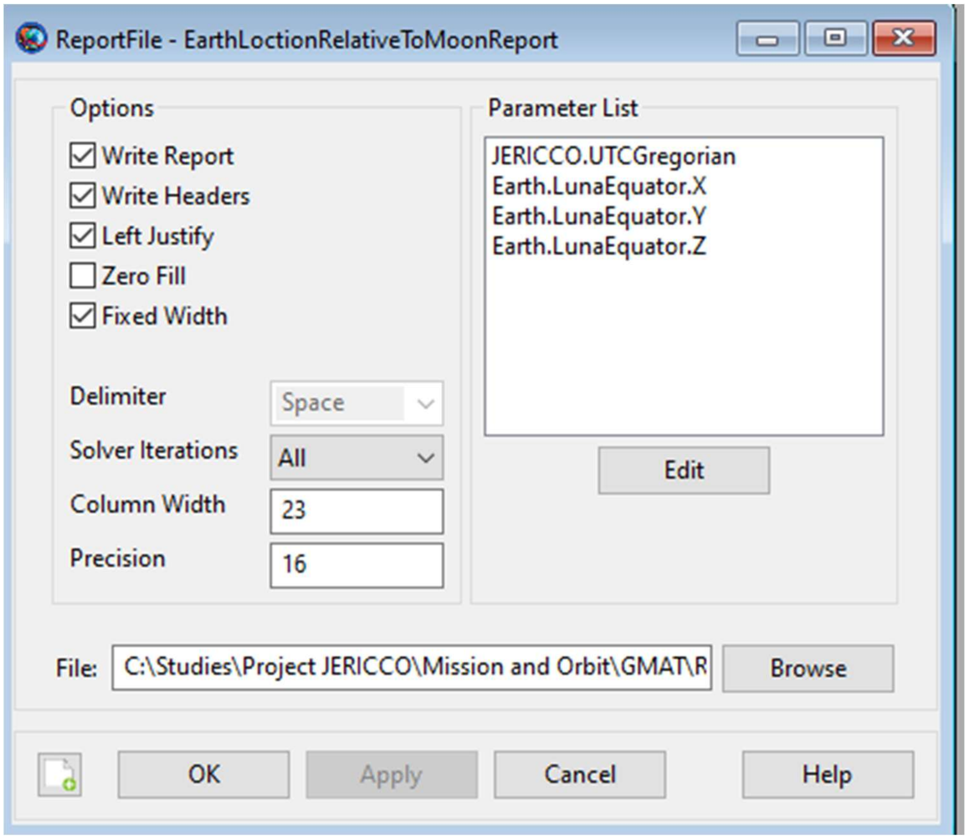


Figure 33 – Earth location report.

4.8.3 Orbit Development Script

For the study of orbit development, we simply propagate the orbit for certain number of days without any pulses.

```
%----------------------------------------
%---------- Mission Sequence
%----------------------------------------

BeginMissionSequence;
Propagate DefaultProp(JERICCO) {JERICCO.ElapsedDays = 365};
```

Figure 34 – Mission sequence for simple propagation

5 Design and Structure Team

5.1 Nomenclature

The following are the convention symbols and their meanings that will be used in this chapter.

$\boldsymbol{1U}$ – Cubic Volume of 10cm x 10cm x 10cm

$\boldsymbol{CAD}$ – Computer Aided Design

$\boldsymbol{COG}$ – Center Of Geometry

$\boldsymbol{COM}$ – Center Of Mass

$\boldsymbol{EPS}$ – Electrical Power System

$\boldsymbol{IMU}$ – Inertial Measurement Unit

$\boldsymbol{MLI}$ – Multi-Layer Insulation

$\boldsymbol{MP}$ – Mass Participation

$\boldsymbol{NDA}$ – Non-Disclosure Agreement

$\boldsymbol{OBC}$ – On Board Computer

$\boldsymbol{PN}$ – Part Number

$\boldsymbol{ROD}$ – Review Of Design

$\boldsymbol{RW}$ – Reaction Wheels

$\boldsymbol{S/X-BAND-A}$ – S/X-Band-Antenna

$\boldsymbol{S/X-BAND-T}$ – S/X-Band-Transmitter

$\boldsymbol{SP}$ – Solar Panels

$\boldsymbol{SS}$ – Sun Sensor

$\boldsymbol{STR}$ – Star Tracker

5.2 Abstract

Beginning with a detailed review of the previous year's work, a thorough market review was conducted between several options in order to select the BUS supplier. The BUS in our case has a major impact on the project, as it is the main structure that defines the base size, volume, and shape of the satellite. In the end, Spacemind's SM-12 BUS was selected, being the best fit for our needs according to last year's team. After the BUS selection, a mass budget of all components was outlined, here the first estimate of the total satellite mass was about 12 Kg. Finally, a basic CAD model of the satellite was made with the available CAD models of the various parts of the satellite. This CAD model was very much a work in progress, mainly because there was no definition on how each of the components within the satellite interacted with the BUS frame. This issue instructed us to our first goal of fully defining the satellite's design.

Beyond the definition of the satellite's design to ensure the success of the mission, we must also validate our design using industry standard tools. Our first tool for validation is modal analysis. The purpose of this analysis, the methodology, results, and conclusions will be elaborated upon in the modal analysis chapter. Secondly, we shall present a printed 3D model of the satellite, that we created in order to further validate our design in the real world. Finally, we shall discuss the risks regarding the satellite structure and the steps we took throughout the year to mitigate said risks.

5.3 Introduction

In order to create and send a satellite into space, a physical satellite must be designed from the ground up, starting with outlining the requirements needed of the satellite structure for the completion of the mission, including load or vibration requirements and more. With the requirements in mind, we will outline the goals we set for this year. The main goals are finalizing the satellite design and validating our design using industry standard tools while conducting constant part tracking.

Part tracking plays an essential role within the design team. It involves monitoring all the current parts utilized in the satellite, along with gathering pertinent information about these parts. This information includes the commercial name of the parts, their mass, volume, manufacturer details, and more. To finalize and validate the satellite design we used CAD or "Computer Aided Design", which is a central tool in any structure & design task, allowing us to model our satellite in detail digitally – exploring proper placement of all parts as well as advance studies of the response of the satellite structure to various mechanical loads, thermal conditions and more. Using CAD software "Solidworks" we will detail our method of mechanical design of the satellite, including collection of part models, placement of parts in the main structure frame (BUS), bracket design, the detailed satellite assembly, mass properties and more.

Our first critical validation tool is modal analysis, this analysis is carried out to assess the eigen frequencies of the mechanical design. Every structure has the tendency to vibrate at certain frequencies, called natural or resonant frequencies. Each natural frequency is associated with a certain shape, called mode shape, that the model tends to assume when vibrating at that frequency. The natural frequencies and corresponding mode shapes depend on the geometry, material properties, and support conditions. When a structure is properly excited by a dynamic load with a frequency that coincides with one of its natural frequencies, the structure undergoes large displacements and stresses. This phenomenon is known as resonance. For undamped systems, resonance theoretically causes infinite motion. Damping, however, puts a limit on the response of the structures due to resonant loads. Frequency studies can help avoid resonance and design vibration isolation systems. They also form the basis for evaluating the response of linear dynamic systems where the response of a system to a dynamic environment is assumed to be equal to the summation of the contributions of the modes considered in the analysis.

An additional tool for validation of our design is 3D printing. 3D printing is the process of constructing a 3D object from a digital 3D model. The 3D model can be created using CAD (Computer Aided Design) software, a 3D scanner or with a digital camera using photogrammetry software. These models can be saved in a STL file format which stores the surfaces of the CAD models using triangulation. The model then needs to be processed in specialized software, which slices the model into thin layers and produces a G-code file that contains the instruction for a 3D printer to print these layers, layer by layer until the full model is formed. Many commercial 3D printers utilize a heated nozzle extruder capable of three-axis motions. These printers typically use ABS plastic, which melts as it's fed into the heated nozzle. Once the melted plastic is accurately placed, it rapidly cools and solidifies. Using this method, we printed a 3D model of the satellite in order to validate our computer design in real life. Lastly we shall address the risks under our management, structural integrity, and parts fitment.

5.4 Requirements

In this section, we will outline the primary requirements and constraints that the design and construction team adhered to during their work. The following table contains the orbit definition and requirements. The requirements were determined based on varied factors, including the launcher's needs, the mission's objectives, the various teams' requirements, and engineering constraints related to design and structure.

Num.	Description	Validation Method
1	Maximum Load Factor (G) for the satellite structure due to spin shall be reported.	ANLYSIS
2	Maximum Load Factor (G) for the satellite structure due to pulse force shall be reported.	ANLYSIS
3	Maximum Load Factor (G) for the satellite structure due to launch phase's loads shall be reported.	ANLYSIS
4	X band Antenna Vector (COM & COG) shall be reported.	ROD
5	S band Antenna Vector (COM & COG) shall be reported.	ROD
6	Reaction wheels Vectors (COM & COG) shall be reported.	ROD
7	Moments of inertia matrix (COM & COG) shall be reported.	ROD
8	Satellite weight shall be max. 15 kg.	ROD
9	Frequencies Scan shall be performed in the range of 5Hz – 2000Hz. This requirement comes from the *Falcon Users Guide* [1].	ANALYSIS
10	Satellite detailed Mass properties document shall be reported.	ROD

Table 8 – Design & Structure requirements

Certain requirements will be validated through one or more of the following methods: Experimentation, Analysis, Calculation, or Review of Design.

5.5 Objectives

The subsequent objectives were recognized as the primary goals for the design and construction team:

1. Adhering to the requirements and objectives set.
2. Component documentation management
3. Conducting Modal Analysis while obtaining clear results.
4. Creating a physical model using 3D printing.
5. Establishing well-defined procedural steps for continued development.

5.6 Components Tracking & Documentation

Apart from the mechanical design tasks, regular monitoring of the components is necessary. A rigorous ongoing documentation process was conducted weekly, encompassing a list of components, and providing detailed information about their primary parameters. The information included component weights, dimensions, CAD model availability, and more. Moreover, any modifications to the model or content were documented and preserved as distinct versions. The latest documentation is shown in Table 9:

#	PN	Qty.	Description	$Mass_{gr}$	$L_{mm} \times W_{mm} \times H_{mm}$	Team	CAD
COMPONENTS							
1	J-A011	1	Thruster	4000	$200 \times 200 \times 100$	C&P	✓
2	J-A021	2	Battery	500	$93 \times 86 \times 41$	Avionics	✓
3	J-A031	2	EPS	191	$92 \times 89 \times 24$	Avionics	✓
4	J-A041	4	RW	185	$50 \times 50 \times 40$	C&P	✓
5	J-A051	2	OBC	24	$65 \times 40 \times 8$	Avionics	✓
6	J-A061	2	IMU	55	$39 \times 45 \times 22$	C&P	✓
7	J-A071	2	S-BAND-T	271	$92 \times 89 \times 25$	Avionics	✓
8	J-A081	1	S-BAND-A	55	$101 \times 83 \times 17$	Avionics	✓
9	J-A091	2	SS	6.5	$43 \times 14 \times 6$	C&P	✓
10	J-A101	1	X-BAND-T	270	✗	Avionics	✗
11	J-A111	1	X-BAND-A	150	✗	Avionics	✗
12	J-A121	7	Solar Panels	✗	✗	Avionics	✗

13	J-A131	1	Payload	1500	✗	Payload	✗
14	J-A141	1	Heater	✗	✗	DSC	✗
15	J-A151	4	MLI	✗	✗	DSC	✗
16	J-A161	2	STR	225	✗	C&P	✓
PARTS-BRACKETS (Al7075-T6)							
1	J-A001	1	BUS	1750	225 × 225 × 327	Structure	✓
2	J-B012	1	Thruster-B	350	217 × 215 × 15	Structure	✓
3	J-A022	2	Battery-B	88	100 × 100 × 23	Structure	✓
4	J-A042	1	RW-B	232	220 × 220 × 35	Structure	✓
5	J-A052	2	OBC-B	16.7	64 × 39 × 9	Structure	✓
6	J-A072	5	S-Band-B	52.8	98 × 93 × 21	Structure	✓
7	J-A162	1	STR-B	260.4	40 × 40 × 15	Structure	✓
8	J-A171	1	Vertical-B	116	203 × 86 × 5	Structure	✓
9	J-A181	1	Horizontal-B	504	150 × 100 × 5	Structure	✓

Table 9: Components Tracking & Documentation

✗ ***Missing Data***

✓ ***Existing Data***

- Table 9 lists the most important and relevant details and properties of each component in the satellite, the full table can be found in the main project drive as a csv file.

5.7 Mechanical Design

5.7.1 Gathering of Component CAD Models

Prior to commencing the mechanical design phase, it is imperative to gather the component models based on the selections made by each respective team. This task is not straightforward. While some companies readily provide CAD models for their components, others necessitate direct contact for model distribution. There are also instances where only a drawing is given without a CAD model. Certain companies are even unwilling to furnish any data prior to the execution of a non-disclosure agreement (which was unfeasible during this project stage).

Consequently, we have acquired all obtainable models that do not require an NDA commitment. For components lacking a CAD model but possessing a drawing, we created a CAD model based on the provided drawing. Unfortunately, components lacking both models and drawings were excluded from the mechanical design process, creating a notable gap that requires subsequent resolution. A summary of available component models can be found in Table 9. The CAD models we created according to the provided drawings are shown in figure 2. Left: IMU, right: STR.

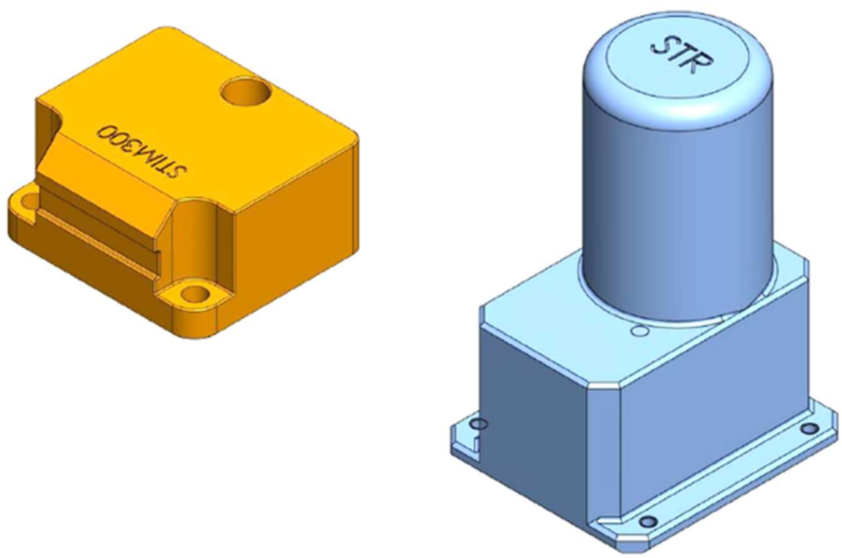


Figure 35: Components Modeling

5.7.2 Primary Considerations in the Design process

5.7.2.1 Appropriate Material Selection

When two different metals are in contact, particularly when exposed to an electrolyte such as a liquid or humid environment, an electrochemical process known as galvanic corrosion can ensue. This occurs due to the disparate electrical potentials of the metals, resulting in accelerated degradation of one or both metals at the point of contact. The main structure (BUS) is made of aluminum 7075-T6, therefore the brackets that are physically attached to it should also be made of aluminum to prevent this phenomenon. In addition, steel screws will be employed within the structure. To mitigate this concern, it is feasible to integrate surface treatments into the fabrication process of the affixed materials. For steel, the suitable treatment is passivation, while unsealed chromic anodize coating is the appropriate option for aluminum. This consideration holds significant importance when designing components crucial for achieving high reliability, especially in applications like space missions.

5.7.2.2 Tool Accessibility

While positioning the components within the assembly, it is imperative to ensure compatibility with a variety of standard tools, including screwdrivers, wrenches, torque wrenches, Allen keys, and more. To address this concern, we have incorporated models of the actual tools that will be employed in the physical assembly process within the assembly model itself. This step allows us to verify and confirm the absence of any physical constraints that might hinder their effective utilization.

5.7.2.3 Components Constrains

Certain components, such as the IMU, exhibit sensitivity to magnetic fields, necessitating their placement at a considerable distance from magnetic sources like reaction wheels. Furthermore, the battery, which is temperature-sensitive, was repositioned away from the solar panels, following an analysis of heat transfer that identified the solar panels as the highest heat producer. Another layer of constraints pertains to location and orientation, exemplified by the specific angles and distances governing the placement of the RW. These RWs must conform to predetermined angles as stipulated by the control team and be positioned at the base of a pyramid, which apex corresponds to the satellite's center of mass. These constraints were realized through the utilization of the RW's bracket, ensuring precise angle replication and approximate distances guided by the center of mass-to-geometric center distance (the pyramid apex aligns with the geometric center). Despite a slight one-centimeter deviation from the constraint degree, this adjustment was communicated to the control team for integration into their modular model, thus aligning with the RW accurate direction.

5.7.3 Coordinate System

This section will provide a comprehensive explanation of the two body axis systems defined for the satellite and been used throughout the entire work process. The systems are identical by their axis orientation but different only by their origin location. One of them has an origin located at the satellite's center of mass, while the other system's origin is located at the geometric center of the satellite. The idea behind defining two systems is to control the final satellite's center of mass location. When constructing the CAD model using an axis system with its origin in the geometric center, component placement can be carried out while continuously accounting for the center of mass's position (which changes with component adjustments or additions/removals). This facilitates control over the center of mass's location, aiming to achieve its proximity to the geometric center, thus striving to achieve balanced distribution across all sides of the satellite.

5.7.3.1 Center of Geometry Coordinate-System

This Coordinate-System is called "COG" and is defined as shown in Table 10:

	Definition
Origin Location	Center of Geometry
X-Axis	Normal Towards the Solar Panel's plain
Y-Axis	Supplementary to a right-handed axis system
Z-Axis	Aligns with the axis of the Payload and towards it
Origin Coordinate in COG	$(0mm \quad 0mm \quad 0mm)$

Table 10: COG Coordinate System Definition

5.7.3.2 Center of Mass Coordinate-System

This Coordinate-System is called "COM" and is defined as shown in Table 11:

	Definition
Origin Location	Center of Mass
X-Axis	Normal Towards the Solar Panel's plain
Y-Axis	Supplementary to a right-handed axis system
Z-Axis	Aligns with the axis of the Payload and towards it
Coordinate Point in COG	$(14.45mm \quad 0.57mm \quad -3.95mm)$

Table 11: COM Coordinate System Definition

5.7.3.3 Coordinate System's Visualization

The COG coordinate system is shown in Figure 36 in red and the COM coordinate system's origin is shown in green (both have identical axis orientations).

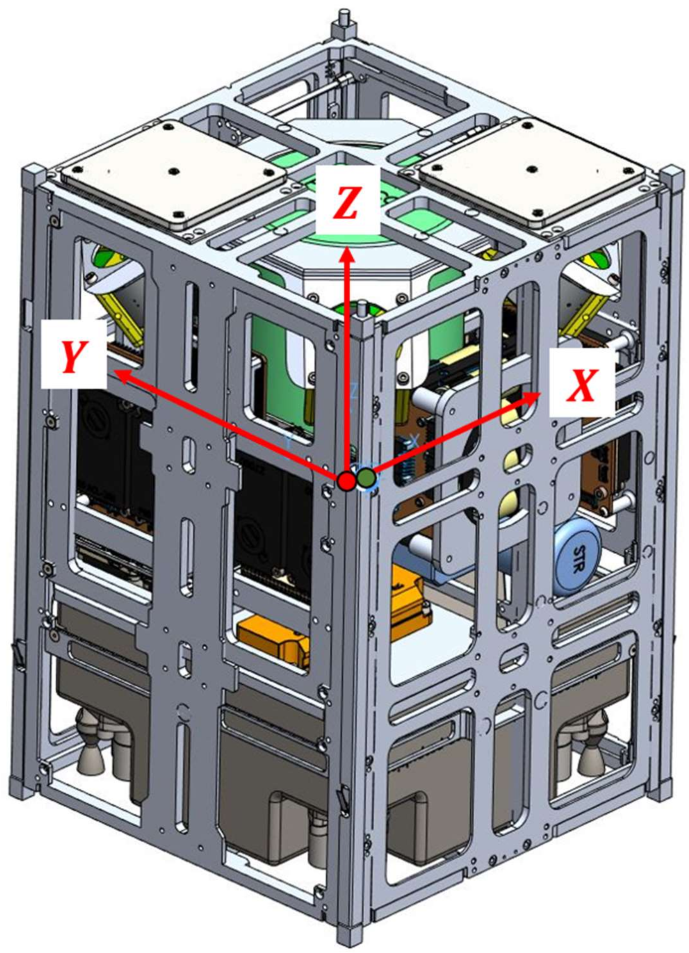


Figure 36: Coordinate System Visualization

5.7.4 Payload Physical Allocation

The Payload team has stipulated both a maximum volume and mass limit for the payload. This stipulation was meticulously factored into the satellite assembly process. In adherence to this, a CAD model was created within the designated spatial envelope, with the model's mass conforming to the predetermined maximum weight outlined in the software. This model serves as the most stringent scenario for Payload selection, as it currently remains unselected. It is seamlessly integrated into analyses, calculations, and the entirety of the assembly procedure. The allocated envelope takes the form of a cylinder with a diameter of 11 cm and a length of 15 cm (nearly 1.5U). The payload's utmost permissible mass has been determined at 1.5 kg. A visual representation of this model can be observed in Figure 37.

Figure 37: Payload Model

5.7.5 Brackets Design

Each component possesses a distinct mechanical interface, which is not directly aligned with the default mechanical interface of the satellite's general structure (BUS). Consequently, the design of dedicated brackets is imperative for each component, serving as adapters to bridge the gap between the component and the BUS mechanical interfaces. As outlined in Figure 38, all brackets shall be made of Al7075-T6 alloy with Unsealed Chromic Anodize coating to prevent galvanic corrosion. A visual representation of the bracket models can be shown in Figure 38. Additional data and information about the brackets are listed in Table 9.

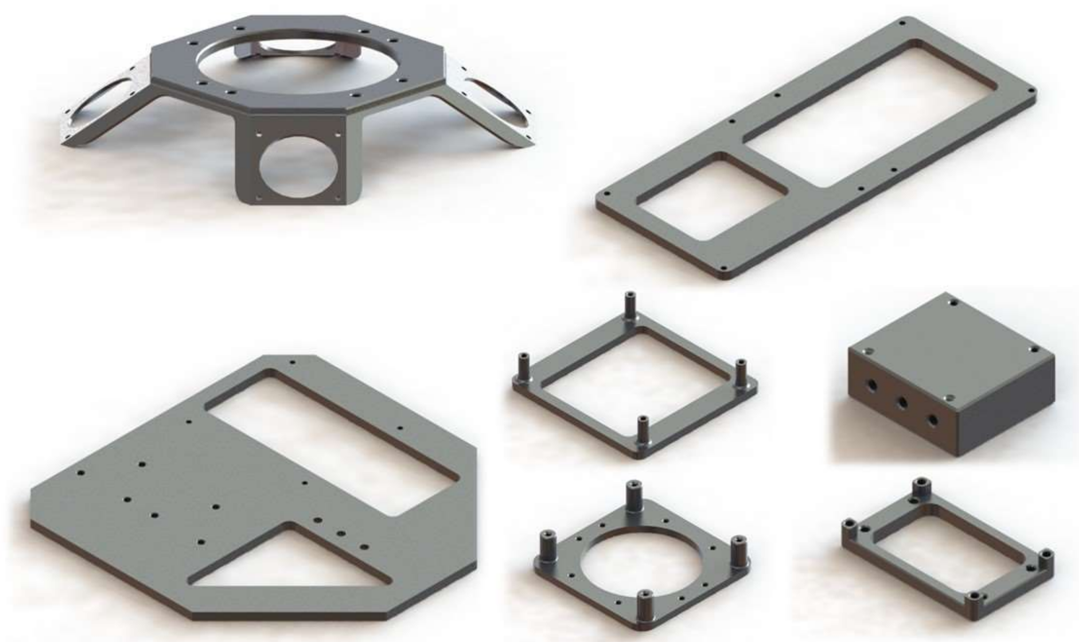


Figure 38: Bracket Models

5.7.6 Complete Satellite Design

The complete assembly is shown in Figure 39.

Figure 39: Complete Design of JERICCO

5.7.7 Satellite Mass Properties

5.7.7.1 COG System

- **Mass**: $14986\ [gr]$
- **Center of Mass**: $(14.45\ [mm] \quad 0.57\ [mm] \quad -3.95\ [mm])$
- **Moments of Inertia Matrix** $[\boldsymbol{gr \cdot mm^2}]$:

$I_{XX} = 376646168$	$I_{XY} = -450433$	$I_{XZ} = 555314$
$I_{YX} = -450433$	$I_{YY} = 205844635$	$I_{YZ} = -360593$
$I_{ZX} = 555314$	$I_{ZY} = -360593$	$I_{ZZ} = 317577481$

5.7.7.2 COM System

- **Mass**: $14986\ [gr]$
- **Center of Mass**: $(0\ [mm] \quad 0\ [mm] \quad 0\ [mm])$
- **Moments of Inertia Matrix** $[\boldsymbol{gr \cdot mm^2}]$:

$L_{XX} = 376411863$	$L_{XY} = -571839$	$L_{XZ} = 1394291$
$L_{YX} = -571839$	$L_{YY} = 202548101$	$L_{YZ} = -327383$
$L_{ZX} = 1394291$	$L_{ZY} = -327383$	$L_{ZZ} = 314505648$

5.7.8 Reaction Wheels Positions

The wheels are symmetrically positioned at 90-degree intervals around the Z-axis:

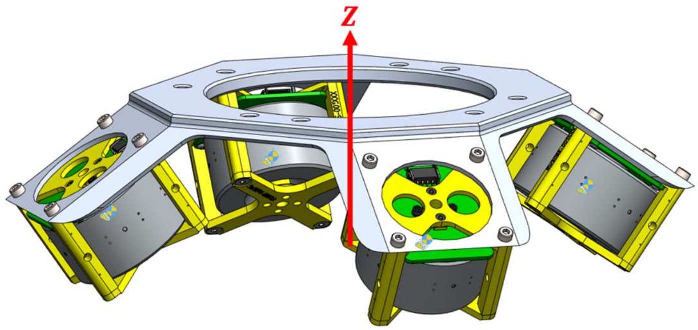


Figure 40: Reaction Wheels Positions

5.7.8.1 COG System

- All values listed below are in $[mm]$.

$$\boldsymbol{RW\ \#1}: (-67.58 \quad 67.58 \quad 115.75) \quad \boldsymbol{Distance\ from\ origin} = 150.11$$

$$\boldsymbol{RW\ \#2}: (\ \ 67.58 \quad 67.58 \quad 115.75) \quad \boldsymbol{Distance\ from\ origin} = 150.11$$

$$\boldsymbol{RW\ \#3}: (\ \ 67.58 \quad -67.58 \quad 115.75) \quad \boldsymbol{Distance\ from\ origin} = 150.11$$

$$\boldsymbol{RW\ \#4}: (-67.58 \quad -67.58 \quad 115.75) \quad \boldsymbol{Distance\ from\ origin} = 150.11$$

5.7.8.2 COM System

- All values listed below are in $[mm]$.

$$\boldsymbol{RW\ \#1}: (-82.04 \quad 67.01 \quad 119.70) \quad \boldsymbol{Distance\ from\ origin} = 159.84$$

$$\boldsymbol{RW\ \#2}: (\ \ 53.13 \quad 67.01 \quad 119.70) \quad \boldsymbol{Distance\ from\ origin} = 147.11$$

$$\boldsymbol{RW\ \#3}: (\ \ 53.13 \quad -68.16 \quad 119.70) \quad \boldsymbol{Distance\ from\ origin} = 147.64$$

$$\boldsymbol{RW\ \#4}: (-82.04 \quad -68.16 \quad 119.70) \quad \boldsymbol{Distance\ from\ origin} = 160.33$$

5.7.9 Propulsion System Position

The propulsion system comprises four nozzles positioned at each corner of it. The thruster's center of mass is situated at $(\mathbf{0}[\boldsymbol{mm}] \quad \mathbf{0}[\boldsymbol{mm}] \quad \mathbf{-156.12}[\boldsymbol{mm}])$ **in COG System**, and at $(\mathbf{-14.45}[\boldsymbol{mm}] \quad \mathbf{-0.57}[\boldsymbol{mm}] \quad \mathbf{-152.16}[\boldsymbol{mm}])$ **in COM System**.

In Figure 41:

- Red values: position vector of each nozzle of the thruster, in COG system.
  - The distances from the origin in this system are identical for each: 197.5mm.
- Green values: position vector of each nozzle of the thruster, in COM system.
- All values listed below are in $[mm]$.

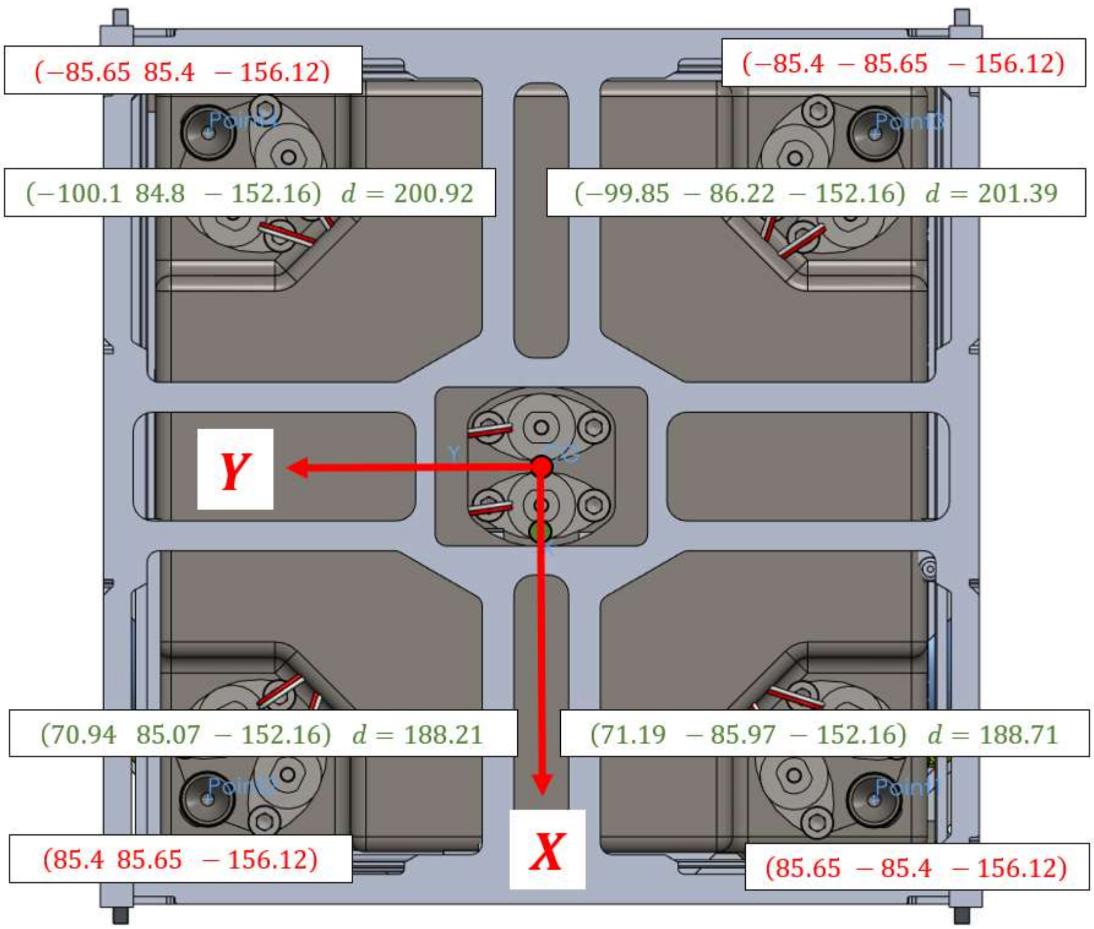


Figure 41: Propulsion System Position

5.8 Modal Analysis

5.8.1 Structure Description

5.8.1.1 Functional Components

Table 12 shows the components that were included in the analysis:

Num.	PN	Description	Qty.
1	J-A011	Thruster	1
2	J-A021	Battery	2
3	J-A031	EPS	1
4	J-A041	Reaction wheel	4
5	J-A051	OBC	2
6	J-A061	IMU	2
7	J-A071	S-Band / X-Band	4
8	J-A081	Antenna	2
9	J-A131	Payload	1
10	J-A161	STR	2

Table 12: List of functional components included in the analysis

All the components mentioned in the table above were simulated as non-structural masses but as point masses, keeping their center of mass location and moments of inertia ("Remote rigid mass"). Table 13 shows the components that were excluded from the analysis:

Num.	PN	Description	Reason	Qty.
1	J-A091	Sun Sensor	*	2
2	J-A121	Solar Panel	No CAD model	6

Table 13: List of functional components excluded in the analysis

* Component was excluded from the analysis due to low impact on the structure: its mass is less than 7 $[gr]$ and its moments of inertia are less than the magnitude of 10^{-7} $[kg \cdot m^2]$. These values are significantly smaller than the other components mass properties values and assumed to be negligible.

5.8.1.2 Mounting Parts

Table 14 shows the mounting parts that were included in the analysis:

Num.	PN	Description	Qty.
1	J-A001	BUS	1

2	J-A012	Thruster-B	1
3	J-A042	RW-B	1
4	J-A022	Battery-B	2
5	J-A072	S-BAND-B	2
6	J-A072	X-BAND-B	2
7	J-A072	EPS-B	1
8	J-A052	OBC-B	2
9	J-A162	STR-B	1
10	J-A171	Vertical-B	4
11	J-A181	Horizontal-B	1

Table 14: List of mounting parts included in the analysis

All the parts mentioned in this section were simulated as structural masses (original CADs) with the proper material properties.

5.8.1.3 Material Properties

Shows the following material properties were used in the simulation:

Num.	$\boldsymbol{Material}$	$\boldsymbol{\rho}\ \left[\frac{\boldsymbol{kg}}{\boldsymbol{m^3}}\right]$	$\boldsymbol{E}\ [\boldsymbol{GPa}]$	$\boldsymbol{\nu}$	$\boldsymbol{\sigma_y}\ [\boldsymbol{MPa}]$	$\boldsymbol{\sigma_{UTS}}\ [\boldsymbol{MPa}]$
1	Al 7075	2810	72	0.33	505	570

Table 15: List of material properties used in the analysis

- The values in the table were taken from SolidWorks 2021 – Student Version.
- All the structural masses (parts) are made from the same material above.

5.8.2 Connections

Connections between parts were modeled using rigid connection elements (pins). These elements are used to model screw connections between the parts with the ability to get reaction forces at the connection.

5.8.2.1 Screw – Spring Model

In order to obtain more realistic results, all the screws were modeled as springs with a torsional constant K_T and linear constant K_L. These constants were calculated according to theoretical formulas shown below, with the proper geometric and material properties. The following equations are the formulas of calculating the spring constants:

$$K_L = \frac{A \cdot E}{L} \qquad K_T = \frac{J \cdot G}{L} \tag{16}$$

Where:

- K_L $[N/m]$ is the linear spring constant.
- K_T $[N \cdot m]$ is the torsional spring constant.
- A $[m^2]$ is the cross-section area of the screw's neck.
- E $[Pa]$ is the elastic modulus of the screw's material.
- G $[Pa]$ is the shear modulus of the screw's material.
- L $[m]$ is the length of the screw (without its head).
- J $[m^4]$ is the polar moment of inertia screw's neck cross-section.

All the screws used in the model are made of Stainless-Steel 304. Table 16 shows the material properties taken to compute the spring constants:

Num.	***Material***	$E\ [GPa]$	$G\ [GPa]$	ν	$\sigma_y\ [MPa]$	$\sigma_{UTS}\ [MPa]$
1	*AISI* 304	190	75	0.29	207	517

Table 16: Screw Material Properties

Table 17 shows the calculated constants values per each screw, which were applied in the analysis:

Num.	$Screw$	$L\ [mm]$	$K_L\ [N/m]$	$K_T\ [N \cdot m]$
1	4-40	5.0	240718882	94
2	8-32	6.3	411884115	347
3	10-32	10	351218715	400
4	M3	15	89535391	39
5	M4	6.0	397935069	308

Table 17: Screw-Spring Model Constants Values

5.8.3 External Mechanical Interface

At the launch phase, the satellite is contained in a dedicated case which acts as the physical interface between each of the satellite's corner tracks and the launcher. Therefore, a 'Fixed Geometry' was defined over each corner in the simulation. Figure 42 shows the fixtures definition in the program:

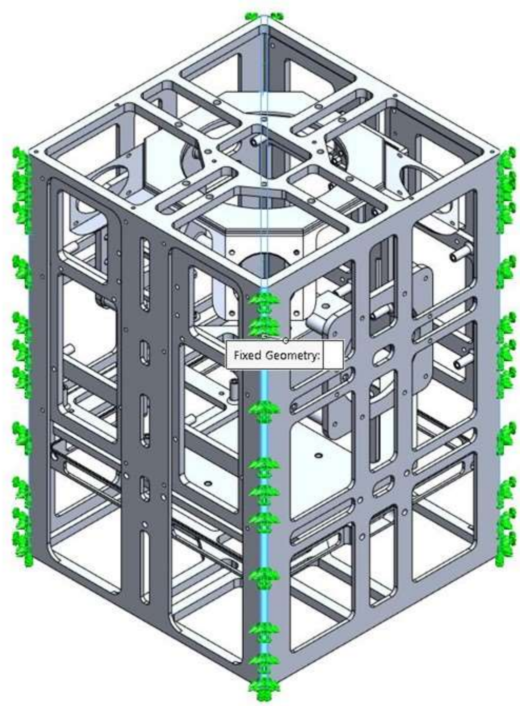


Figure 42: Exterior mechanical interface – analysis definition

- The blue highlight regions are the physical interaction with the launch case.
- The green arrows represent reaction forces in all directions (fixture).
- These four rails will not be affected or deformed in the analysis.

5.8.4 Analysis Coordinate System

In the analysis program, the coordinate system is defined as shown in Figure 43:

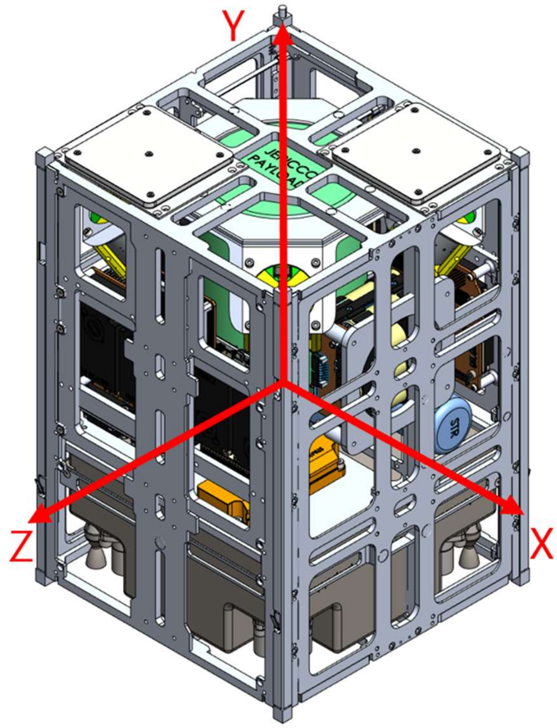


Figure 43: Coordinate System in the Analysis

- All the results corresponded with this coordinate system and NOT the two which were defined earlier (COG/COM).

5.8.5 The Finite Element Model

A real model has an infinite number of natural frequencies. However, a finite element model has a finite number of natural frequencies that is equal to the number of degrees of freedom considered in the model. The analysis was performed with Solidworks Simulations® student version 2021, a commercial general-purpose FE (Finite Elements).

5.8.5.1 Simplified Parts

To be able to run the mesh process it was essential to re-model the satellite's structure part (BUS) due to complicated geometry that failed the mesh process. Figure 44 shows the original model (left) and the simplified model (right):

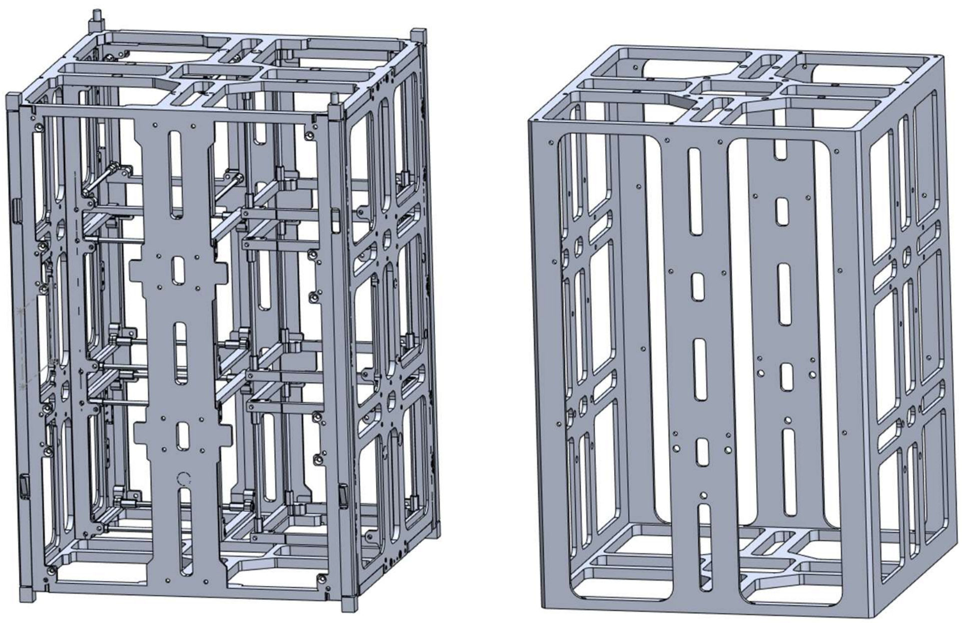


Figure 44: Original BUS model (left) and simplified BUS model (right)

This part replaced the original part in the analysis. All the original properties were applied to the simplified model – material properties (Al7075), mass, moments of inertia and center of mass.

5.8.5.2 Analysis Model

The analysis model is shown in Figure 45:

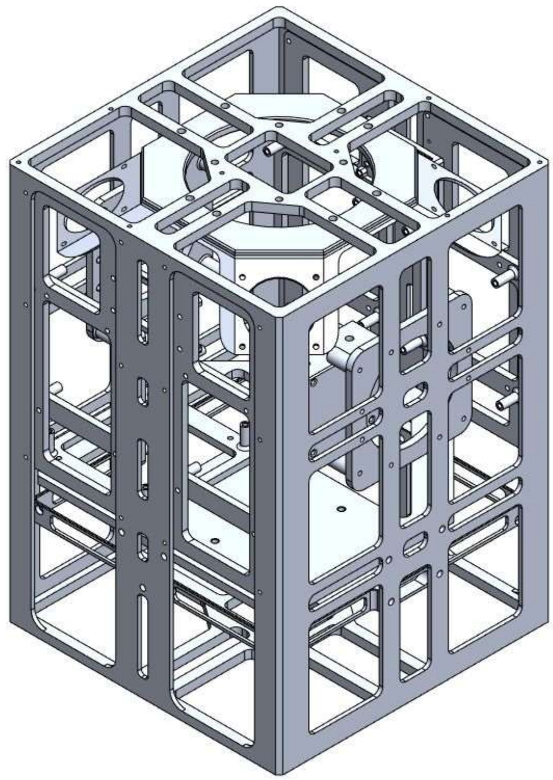
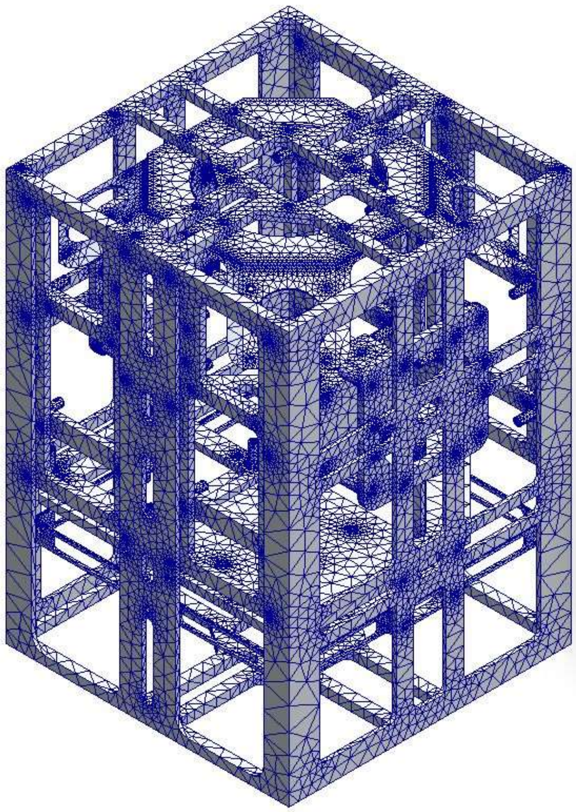


Figure 45: Analysis Model

5.8.5.3 Mesh

A general view of the unit structure FE mesh is shown in Figure 46:



Mesh Details

Study name	MA1 (-Default-)
Mesh type	Solid Mesh
Mesher Used	Blended curvature-based mesh
Jacobian points for High quality mesh	16 points
Max Element Size	20.4875 mm
Min Element Size	1.02438 mm
Mesh quality	High
Total nodes	535149
Total elements	303031
Maximum Aspect Ratio	80.813
Percentage of elements with Aspect Ratio < 3	94.1
Percentage of elements with Aspect Ratio > 10	0.28
Percentage of distorted elements	0
Number of distorted elements	0
Remesh failed parts independently	Off
Time to complete mesh(hh:mm:ss)	00:01:02
Computer name	GIL

Figure 46: Mesh Details

The structural parts were meshed using second order fully integrated 10-noded tetra solid element. It is used to mesh irregular geometry 3D CNC machined parts. This kind of element allows greater detail in describing the physical interaction and propagation of displacements and deformations.

5.8.6 First Iteration Results

The modal analysis performs an Eigen-value extraction to calculate the natural frequencies and the corresponding mode shapes of the assembly. This includes additional stiffness effects due to preloads since geometric nonlinearity is accounted for. In this iteration, our goal was to scan the first 100 mode shapes. This quantity is relatively large, chosen to effectively cover the desired frequency range (5Hz-2000Hz).

5.8.6.1 Natural Frequencies & Mass Participation

Each mode shape has its natural frequency and the mass participation (%) in each direction, which is the partial mass of the structure that is affected in the mode. The natural frequencies, mode shapes and mass participation description are listed in Table 18 (green = 5% to 10%, blue = more than 10%):

Mode No.	Freq (Hz)	X direction	Y direction	Z direction	SUM
1	173.07	4.92%	0.00%	3.89%	8.80%
2	187.47	0.32%	0.00%	0.33%	0.64%
3	189.48	0.43%	0.07%	1.40%	1.90%
4	193.78	0.10%	1.42%	0.05%	1.56%
5	208.65	0.04%	0.00%	0.43%	0.47%
6	214.03	6.84%	0.01%	0.10%	6.96%
7	214.69	6.08%	0.01%	0.07%	6.16%
8	288.03	0.65%	0.03%	3.96%	4.65%
9	326.12	0.01%	0.00%	1.37%	1.39%
10	326.25	0.02%	0.00%	3.51%	3.53%
11	332.12	2.95%	0.57%	0.09%	3.61%
12	387.84	0.36%	0.30%	0.92%	1.58%
13	392.05	0.00%	10.98%	0.02%	11.00%
14	494.51	0.04%	8.12%	0.04%	8.21%
15	496.53	0.00%	1.35%	0.00%	1.36%
16	497.59	0.00%	1.64%	0.00%	1.64%
17	498.19	0.00%	0.02%	0.00%	0.02%
18	505.64	0.16%	31.87%	0.15%	32.18%
19	545.72	0.00%	0.04%	0.03%	0.07%

20	554.73	1.22%	0.06%	4.07%	5.35%
21	567.87	0.85%	3.73%	0.07%	4.65%
22	572.97	0.07%	0.30%	0.14%	0.51%
23	577.31	0.36%	0.51%	0.28%	1.16%
24	619.75	2.76%	2.43%	0.11%	5.30%
25	690.91	35.63%	0.15%	0.00%	35.78%
26	701.38	0.40%	0.04%	10.11%	10.55%
27	752.72	1.16%	0.00%	20.37%	21.53%
28	766.01	0.04%	0.01%	2.15%	2.20%
29	777.92	0.08%	0.12%	9.82%	10.02%
30	782.97	0.06%	0.03%	0.04%	0.12%
31	794.57	0.32%	0.03%	2.24%	2.58%
32	816.36	0.24%	0.05%	0.01%	0.31%
33	849.90	0.41%	0.07%	7.40%	7.88%
34	889.50	0.19%	0.17%	0.11%	0.48%
35	937.26	1.39%	4.01%	0.27%	5.68%
36	946.66	1.04%	0.50%	0.21%	1.75%
37	997.33	0.25%	0.00%	0.01%	0.26%
38	1003.00	0.03%	0.00%	0.00%	0.03%
39	1004.20	0.01%	0.00%	0.01%	0.02%
40	1006.20	0.11%	0.01%	0.04%	0.16%
41	1008.00	0.00%	0.00%	0.15%	0.16%
42	1275.70	0.01%	14.10%	0.00%	14.11%
43	1303.70	0.00%	1.22%	0.01%	1.23%
44	1356.20	0.16%	0.21%	0.00%	0.37%
45	1440.00	0.00%	0.00%	0.00%	0.00%
46	1466.50	0.00%	0.02%	0.01%	0.03%
47	1608.20	0.00%	0.00%	0.14%	0.14%

48	1657.00	0.00%	0.43%	0.45%	0.87%
49	1803.70	0.00%	0.00%	0.09%	0.09%
50	1818.00	0.00%	0.00%	0.00%	0.00%
51	1822.00	0.00%	0.00%	0.00%	0.00%
52	1827.40	0.00%	0.00%	0.00%	0.00%
53	1827.70	0.00%	0.00%	0.00%	0.00%
54	1869.40	0.79%	0.00%	0.01%	0.80%
55	1872.50	0.01%	0.00%	0.00%	0.01%
56	1874.80	0.06%	0.00%	0.00%	0.06%
57	1877.70	0.19%	0.00%	0.01%	0.20%
58	1879.10	0.00%	0.00%	0.00%	0.00%
59	1891.60	0.28%	0.00%	0.01%	0.29%
60	2135.20	0.48%	0.00%	0.05%	0.53%
61	2264.10	0.00%	0.01%	0.00%	0.01%
62	2345.10	1.66%	0.00%	0.05%	1.71%
63	2347.40	0.10%	0.00%	0.00%	0.10%
64	2402.30	2.62%	0.01%	0.62%	3.25%
65	2410.80	0.60%	0.12%	0.20%	0.92%
66	2411.40	0.00%	0.00%	0.00%	0.00%
67	2459.20	1.18%	0.16%	1.98%	3.31%
68	2502.70	0.02%	0.00%	5.68%	5.70%
69	2523.50	0.25%	0.03%	1.60%	1.88%
70	2538.40	0.00%	0.00%	0.03%	0.04%
71	2563.10	0.18%	0.00%	0.03%	0.21%
72	2564.50	0.09%	0.00%	0.02%	0.11%
73	2585.70	0.16%	0.00%	0.01%	0.17%
74	2589.50	0.01%	0.00%	0.00%	0.01%
75	2594.50	0.02%	0.00%	0.00%	0.02%

76	2606.80	0.00%	0.00%	0.00%	0.00%
77	2609.10	0.00%	0.00%	0.01%	0.01%
78	2672.60	0.00%	0.00%	0.00%	0.00%
79	2680.20	0.00%	0.00%	0.00%	0.00%
80	2697.50	0.00%	0.00%	0.00%	0.00%
81	2700.70	0.00%	0.00%	0.00%	0.00%
82	2704.40	0.00%	0.00%	0.00%	0.00%
83	2723.90	0.02%	0.01%	0.00%	0.03%
84	2727.00	0.01%	0.01%	0.00%	0.02%
85	2731.40	0.07%	0.00%	0.00%	0.08%
86	2741.60	0.00%	0.00%	0.00%	0.00%
87	2803.20	0.00%	0.00%	0.00%	0.00%
88	2806.70	0.00%	0.00%	0.00%	0.00%
89	2819.30	0.00%	0.00%	0.00%	0.00%
90	2820.50	0.00%	0.00%	0.00%	0.00%
91	2904.00	0.00%	0.00%	0.00%	0.00%
92	2911.50	0.00%	0.00%	0.00%	0.00%
93	3016.40	0.00%	0.00%	0.00%	0.00%
94	3045.40	1.82%	0.00%	1.21%	3.03%
95	3059.00	0.01%	0.00%	0.00%	0.01%
96	3080.90	0.38%	0.01%	0.39%	0.78%
97	3089.80	0.60%	0.00%	0.49%	1.08%
98	3143.20	0.00%	0.00%	0.02%	0.02%
99	3170.40	0.00%	0.00%	0.49%	0.49%
100	3210.80	0.00%	0.01%	0.05%	0.06%
SUM	-	81.36%	85.02%	87.64%	-

Table 18: Mass Participation in each Mode Shape

Table 18 contains the numerical values per each mode. In this analysis it is important to verify that the scanned frequency range was wide enough to have the sum of mass participation in each direction close to 100%. This means that all of the significant modes and natural frequencies of the structure are found. A mode with less than 5% mass participation (the sum in all directions) is assumed to be negligible due to low effect. The total mass participation for this iteration was approximately 85% in each direction (considering the first 100 mode shapes). However, the more significant aspect is that the final mode shape within the desired range (5Hz-2000Hz) is the 59th out of 100. As a result, achieving 100% mass participation in this scenario becomes less crucial.

Figure 47 shows a sum of the mass participation in all directions as a function of the natural frequency:

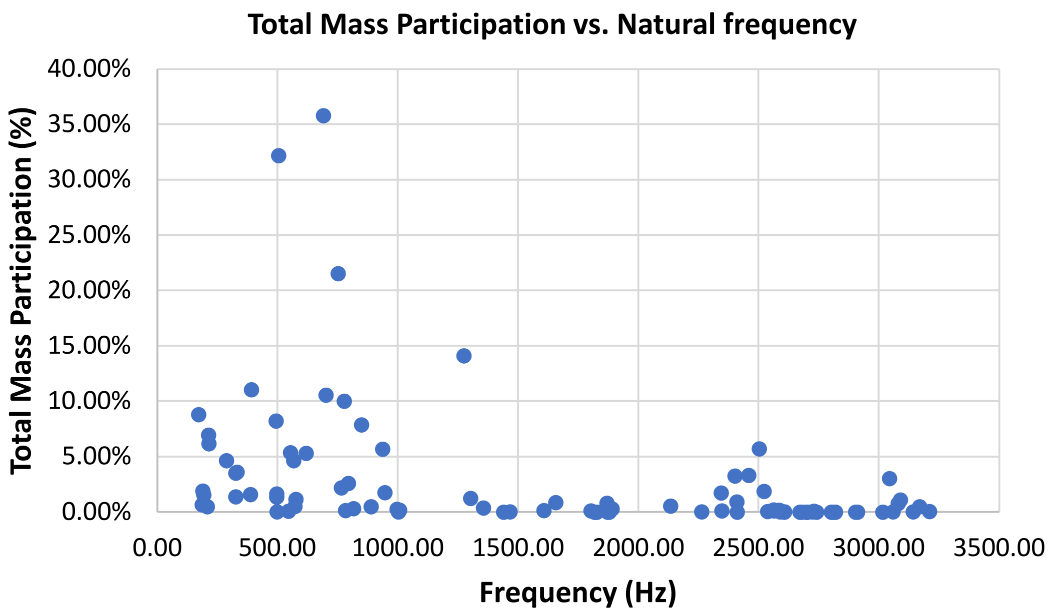


Figure 47: Total Mass Participation vs. Natural frequency

From the graph it can be seen that the most affected region is in the frequencies range of 400Hz to 1275Hz.

Figure 48 shows the natural frequency (Hz) as a function of the mode number (#). This can be helpful with a launch-dedicated frequencies range.

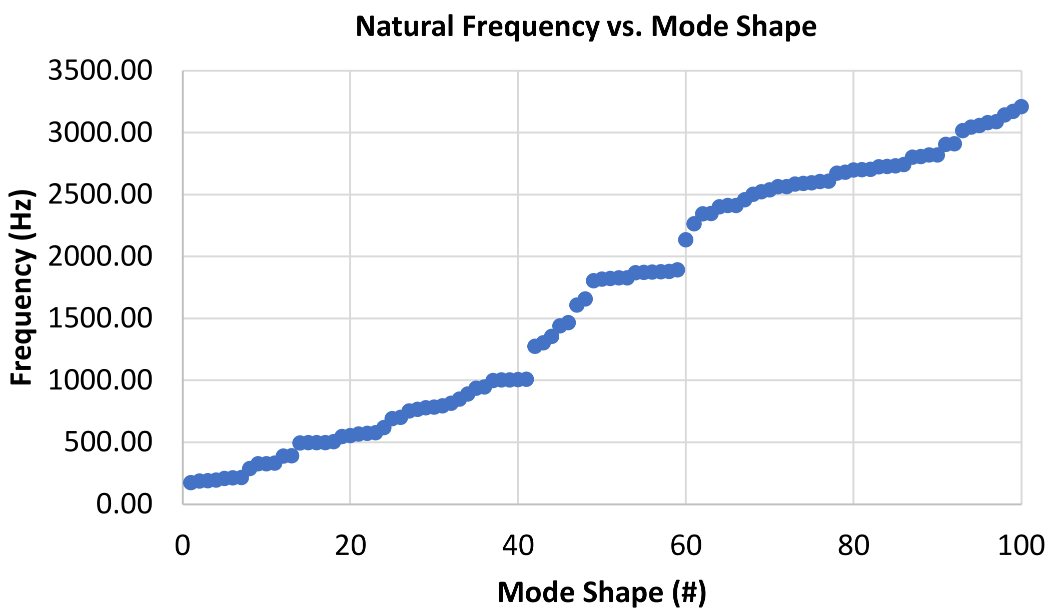


Figure 48: Natural Frequency vs. Mode Shape

Figure 49 shows the mass participation in each direction vs. frequency:

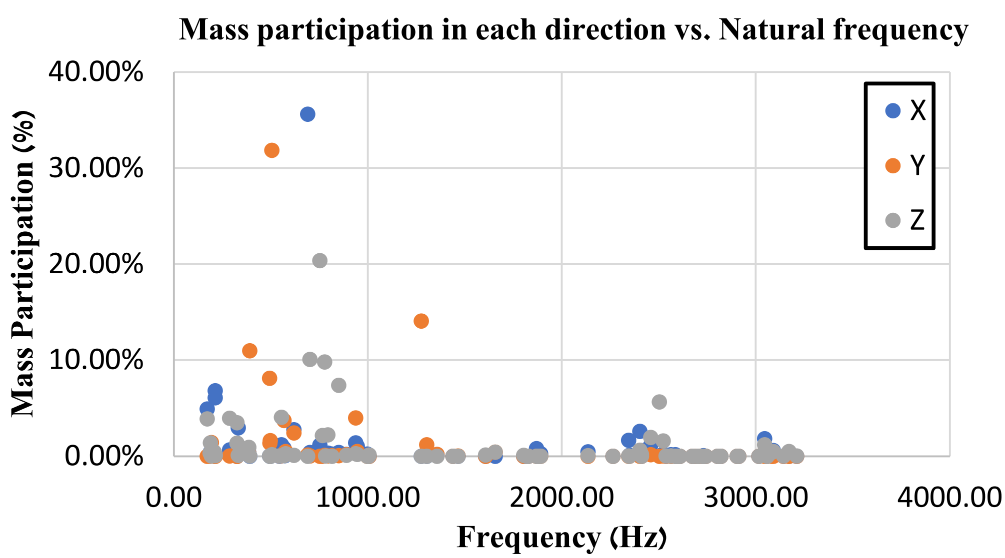


Figure 49: Mass participation in each direction vs. Natural frequency

5.8.6.2 Visual Results

Visual demonstration of the mode shapes helps us to understand the physical shapes of vibrations and the most affected regions in the structure for each natural frequency. Each figure represents a single mode shape, associated with the corresponding natural frequency presented in the upper left side text box. In the right side in each figure there is a colorized scale bar that associates colors with intensity of the displacements in arbitrary unit "AMPRES" of resultant relative vibrational amplitude. AMPRES shows the relative vibration of different parts of the assembly. This gives a sense of the relative displacement of regions of the body in each mode of vibration. The following figures show a visual demonstration of the mode shapes with more than 10% total mass participation (modes 13, 18, 25, 26, 27, 29, 42).

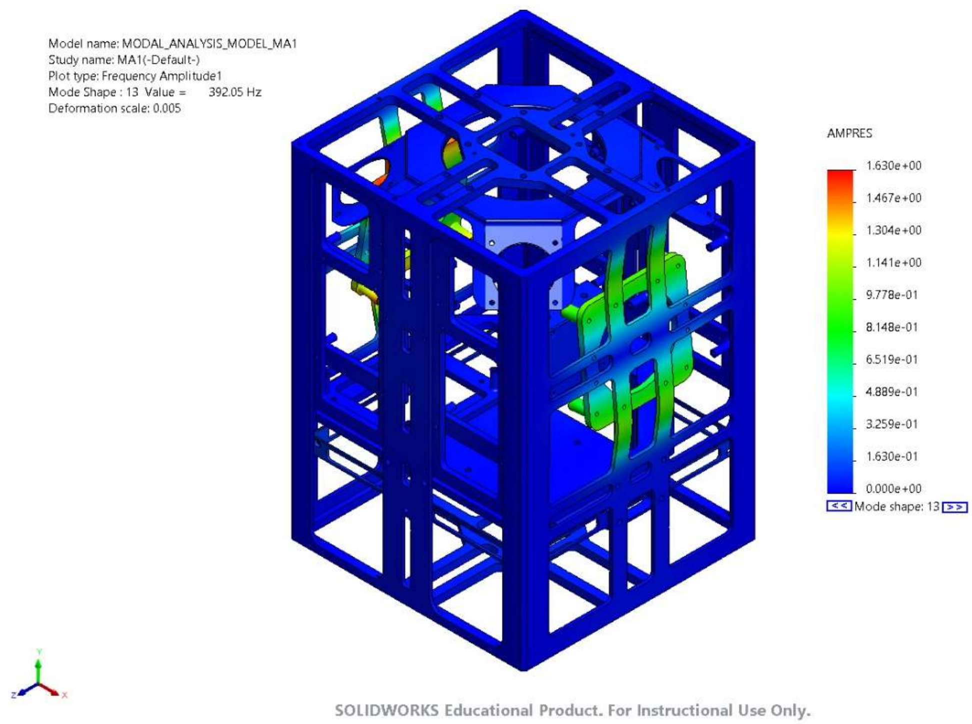


Figure 50: Mode Shape #13

- Mode: 13
- Natural frequency: 392 Hz
- Most affected region: Batteries
- Mass participation: 0% (X), 11% (Y), 0% (Z)
- Highest AMPRES: 1.63

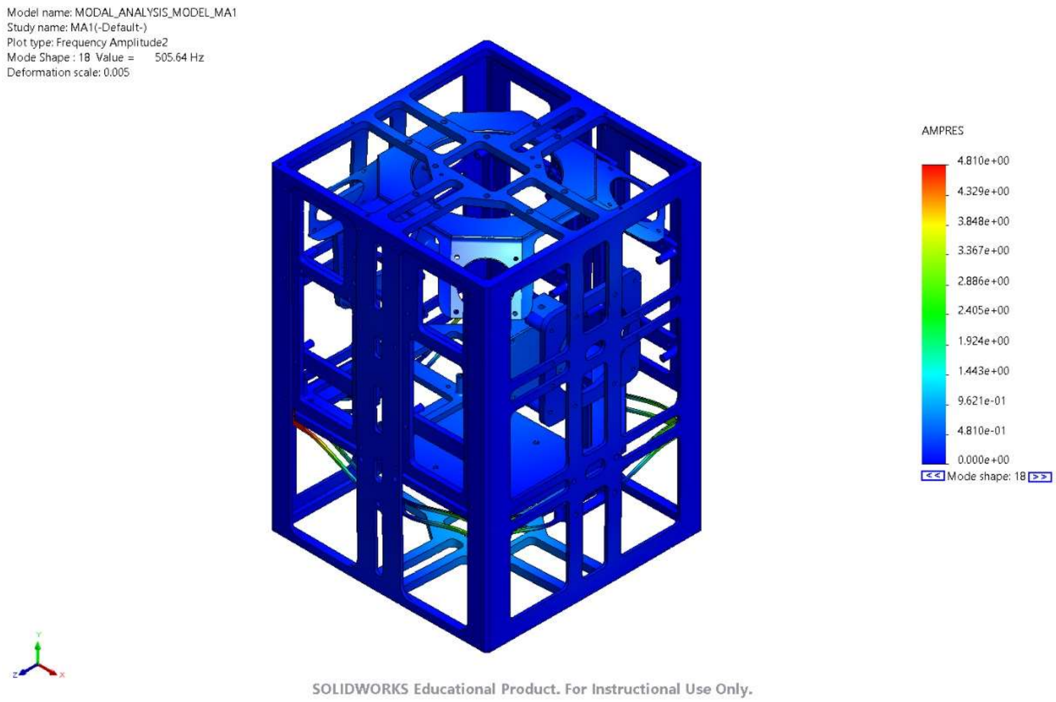


Figure 51: Mode Shape #18

- Mode: 18
- Natural frequency: 505.6 Hz
- Most affected region: Thruster
- Mass participation: 0% (X), 32% (Y), 0% (Z)
- Highest AMPRES: 4.81

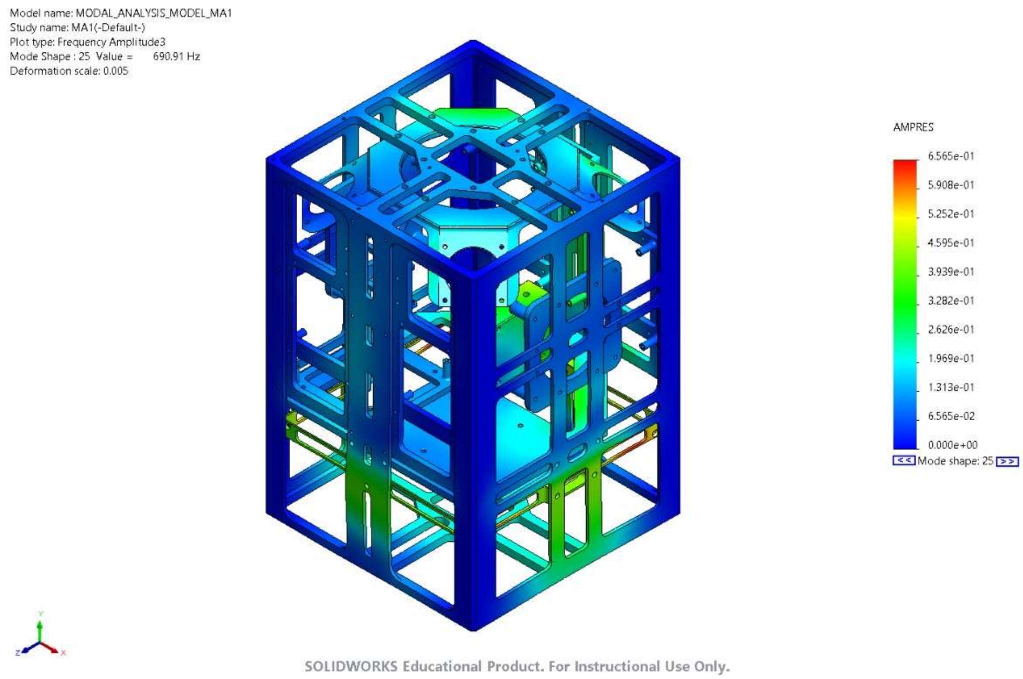


Figure 52: Mode Shape #25

- Mode: 25
- Natural frequency: 691 Hz
- Most affected region: Thruster, RWs, STR
- Mass participation: 35.6% (X), 0% (Y), 0% (Z)
- Highest AMPRES: 0.66

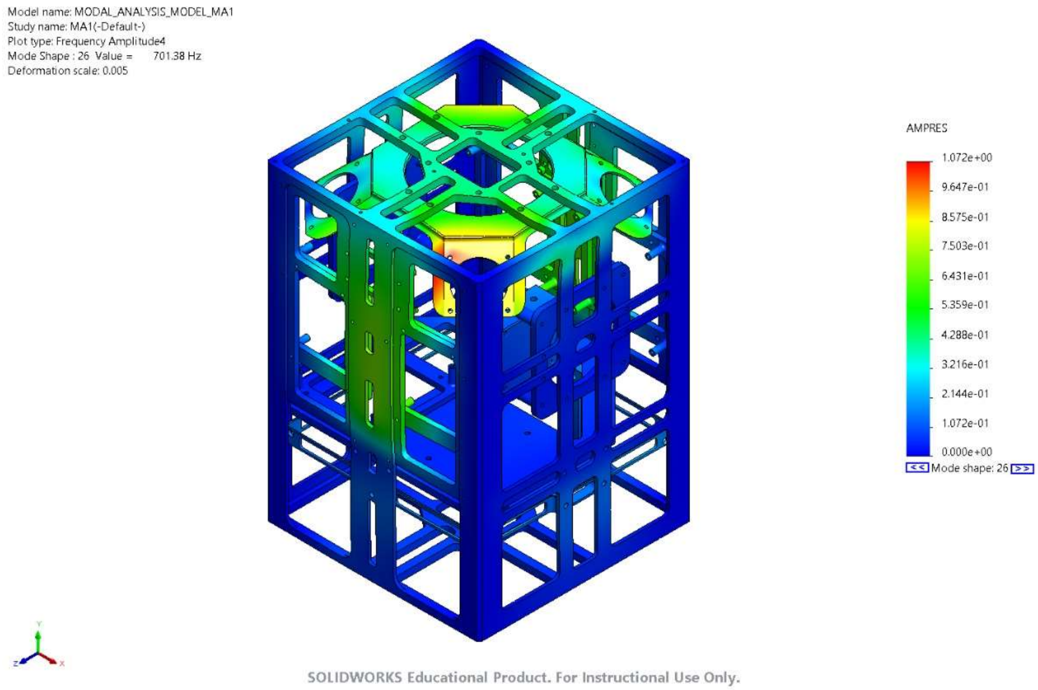


Figure 53: Mode shape #26

- Mode: 26
- Natural frequency: 701.4 Hz
- Most affected region: RWs
- Mass participation: 0% (X), 0% (Y), 10% (Z)
- Highest AMPRES: 1.07

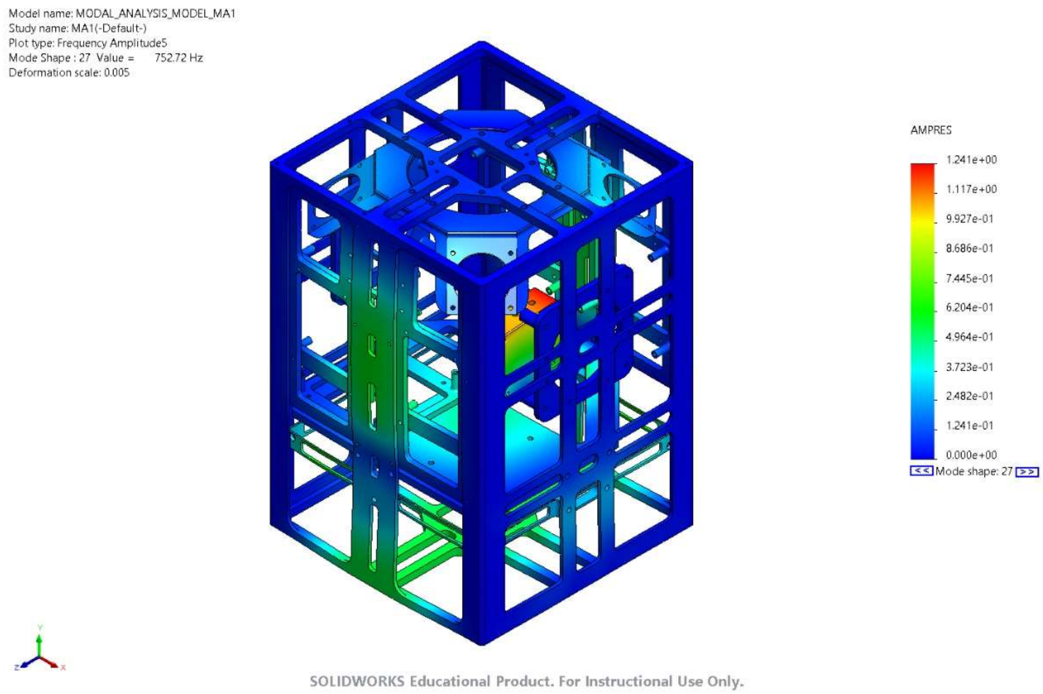


Figure 54: Mode shape #27

- Mode: 27
- Natural frequency: 752.7 Hz
- Most affected region: STR, Thruster
- Mass participation: 1.2% (X), 0% (Y), 20.4% (Z)
- Highest AMPRES: 1.24

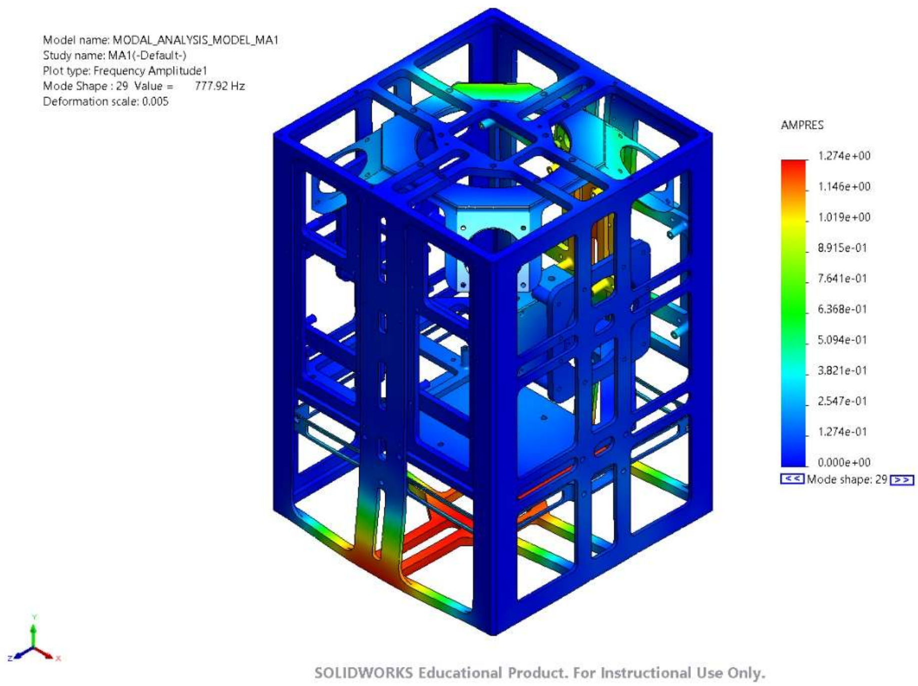


Figure 55: Mode shape #29

- Mode: 29
- Natural frequency: 778 Hz
- Most affected region: Thruster, RWs
- Mass participation: 0% (X), 0% (Y), 10% (Z)
- Highest AMPRES: 1.27

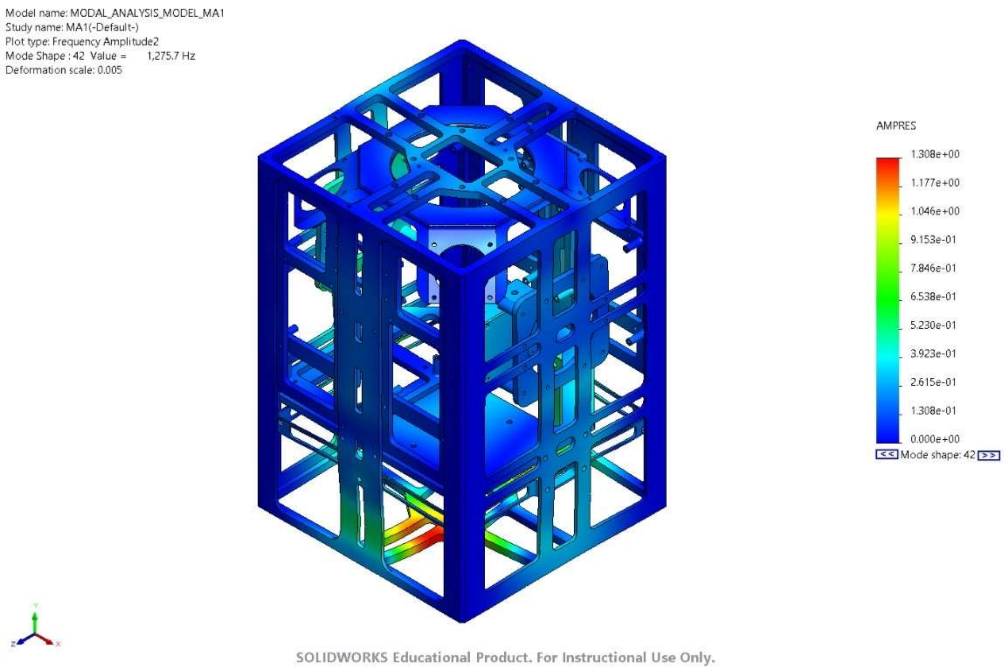


Figure 56: Mode shape #42

- Mode: 42
- Natural frequency: 1275.7 Hz
- Most affected region: Thruster
- Mass participation: 0% (X), 14% (Y), 0% (Z)
- Highest AMPRES: 1.31

5.8.6.3 Structural Summary & Conclusion

- The modal analysis definition successfully simulated the interactions between parts and components, which produced physical and clear results.
- The most effective range of frequencies is between 400Hz and 1275Hz regarding Figure 47.
- From Figure 51 we can see the effect of the propulsion system (PS) on the structure at 505Hz. There are 32% of mass participation stemming from the PS causing significant gain and displacements to its bracket. Therefore, a PS bracket redesign is essential. Figure 52 confirms this observation that the root cause is the PS bracket's unoptimized design.
- All other mode shapes above 10% mass participation either stem from the structure skeleton itself or result in low gain.
- Above the 1275Hz frequency (mode shape #42) there are no significant mode shapes with high mass participation.
- With a dedicated range of frequencies, using the information from Figure 47 and Figure 48, it is possible to minimize the mode shapes of interest.
- After another model revision a modal analysis iteration will be performed.

5.8.7 Structure Design Revision

One important conclusion from the first modal analysis iteration was to re-design the Thruster bracket due to significant mode shape with high mass participation. The old design is shown in Figure 57:

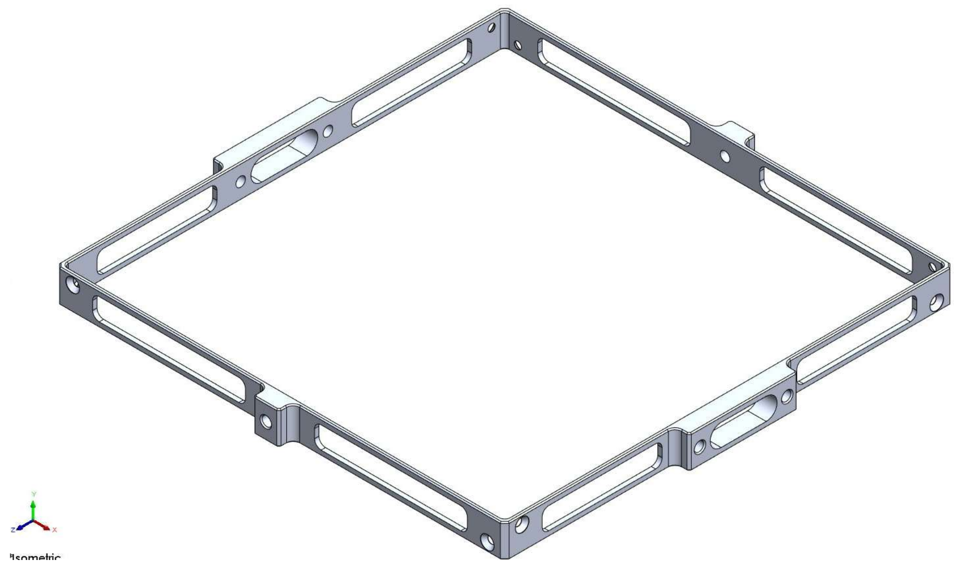


Figure 57: Old Thruster's Bracket Design

This bracket design has two major problems:

- The mounting screws (the mechanical interface to the satellite structure) are located in the middle of each edges instead of in each of the frame corners.
- The frame is too thin, and its mass is 58 [gr], compared to the thruster itself which is 4000 [gr].

Therefore, a new bracket design was needed to reduce the mass participation in low frequencies.

Figure 58 shows the new Thruster's Bracket design:

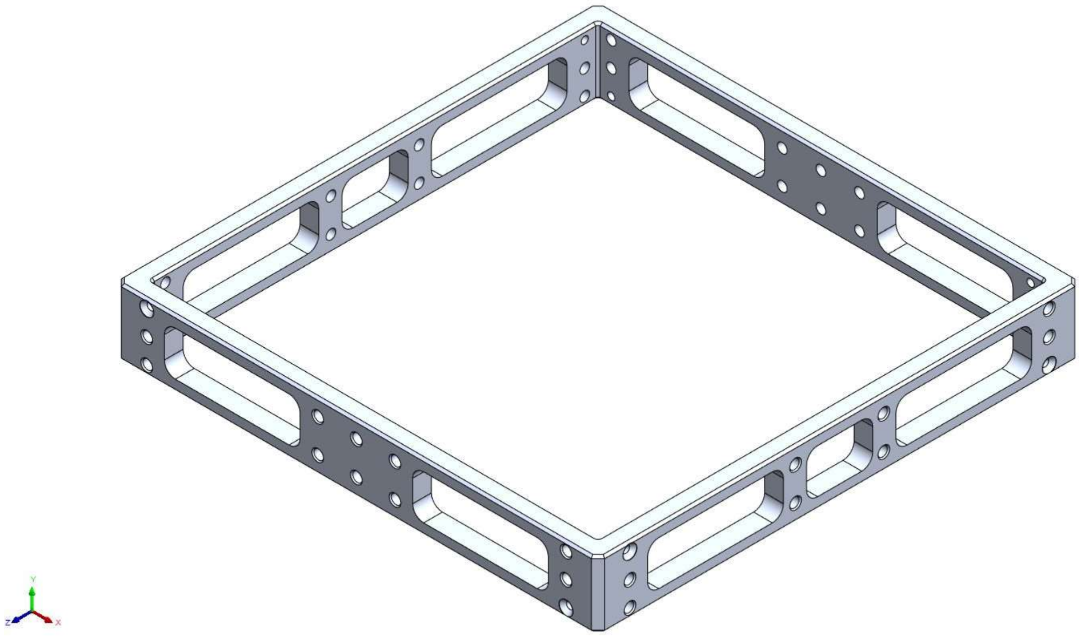


Figure 58: New Thruster's Bracket Design

This bracket has 28 mounting screws to the satellite structure located in the center and in the corners of the frame. In addition, the frame is thicker than the old one, the mass of the new bracket is 360 [gr], which is less than 10% of the thruster mass.

After the replacement of the brackets, a second modal analysis was performed.

5.8.8 Second Iteration Results

The analysis was defined in the same way as the first iteration. The single difference is the replacement of new thruster bracket by the old one. The frequency range that needs to be scanned is according to requirement #9 in Table 8. Therefore, this modal analysis iteration will discuss this range of frequencies.

5.8.8.1 Natural Frequencies & Mass Participation

The natural frequencies, mode shapes and mass participation description are listed in Table 19 (green = 5% to 10%, blue = more than 10%):

Mode No.	Freq (Hz)	X direction	Y direction	Z direction	SUM
1	173.11	4.85%	0.00%	3.81%	8.66%
2	187.49	0.32%	0.00%	0.31%	0.63%
3	189.50	0.45%	0.08%	1.36%	1.89%
4	193.74	0.10%	1.45%	0.05%	1.59%
5	208.63	0.02%	0.00%	0.41%	0.44%
6	213.60	5.01%	0.01%	0.09%	5.10%
7	214.42	7.51%	0.00%	0.09%	7.61%
8	292.71	1.32%	0.02%	4.18%	5.52%
9	325.99	0.10%	0.01%	0.09%	0.20%
10	326.12	0.00%	0.00%	4.69%	4.69%
11	329.31	2.48%	0.39%	0.05%	2.92%
12	367.67	0.00%	20.06%	0.00%	20.07%
13	388.22	0.19%	0.07%	0.82%	1.09%
14	463.95	0.09%	19.76%	0.12%	19.96%
15	540.72	0.00%	0.04%	0.02%	0.06%
16	554.76	1.42%	0.00%	3.86%	5.28%
17	565.29	0.28%	1.83%	0.04%	2.14%
18	572.80	0.40%	0.06%	0.17%	0.62%
19	576.75	0.16%	0.02%	0.26%	0.44%
20	616.33	0.16%	1.39%	0.02%	1.57%

21	702.54	1.20%	0.00%	6.25%	7.45%
22	757.40	0.02%	0.17%	2.88%	3.07%
23	766.61	0.01%	0.01%	0.26%	0.28%
24	781.39	0.28%	0.02%	3.74%	4.03%
25	786.58	0.01%	0.00%	2.92%	2.93%
26	797.31	0.03%	0.04%	1.33%	1.40%
27	817.51	0.14%	0.02%	0.20%	0.35%
28	884.79	2.06%	0.00%	0.00%	2.06%
29	946.35	1.53%	0.01%	0.04%	1.59%
30	955.84	0.01%	0.29%	0.10%	0.39%
31	1232.80	0.00%	0.00%	0.25%	0.26%
32	1307.30	0.00%	0.21%	0.12%	0.33%
33	1371.10	0.32%	0.07%	0.06%	0.45%
34	1396.00	0.01%	0.00%	0.03%	0.04%
35	1440.50	0.00%	0.00%	0.00%	0.00%
36	1548.90	0.01%	0.01%	0.25%	0.28%
37	1572.90	0.00%	0.00%	0.00%	0.00%
38	1612.10	0.00%	0.00%	0.10%	0.10%
39	1697.70	0.01%	0.06%	0.29%	0.36%
40	1726.60	0.22%	0.26%	0.59%	1.08%
41	1754.80	1.29%	0.02%	0.03%	1.35%
42	1840.40	2.83%	0.00%	0.01%	2.83%
43	1881.30	1.25%	0.00%	0.02%	1.27%
SUM	-	36.11%	46.38%	39.88%	-

Table 19: Mass Participation in the Second Iteration

Figure 59 shows the total mass participation (%) Vs. frequency (Hz) for the first iteration and the second iteration from 0Hz to 2000Hz:

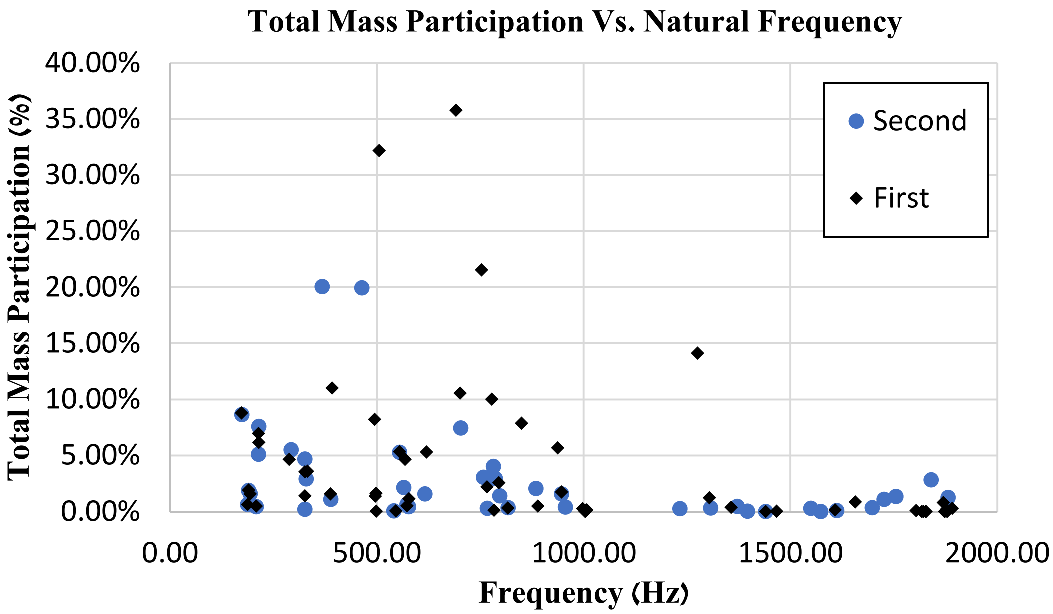


Figure 59: Total Mass Participation vs. Natural Frequencies Comparison

- First, the highest mass participation is 20%, compared to the first iteration which had several modes with more than 30% mass participation.
- Second, the number of mode shapes with more than 10% mass participation was reduced from 7 (first iteration) to 2 (second iteration).
- Third, the most affected region is in the frequencies range of 370Hz to 470Hz, compared to the first analysis (400Hz to 1275Hz). In general, we aim for mode shapes at higher frequencies. In this case, the iteration lowered the significant mode shape from 400Hz to 370Hz at the minimum frequency. However, many significant mode shapes were omitted – all the mode shapes between 470Hz and 1275Hz disappeared after the second iteration. Therefore, the reduction in the minimum mode shape from 400Hz to 370Hz proves beneficial when considering the omitted modes.

Figure 60 shows the mass participation in each direction (%) Vs. Frequency (Hz):

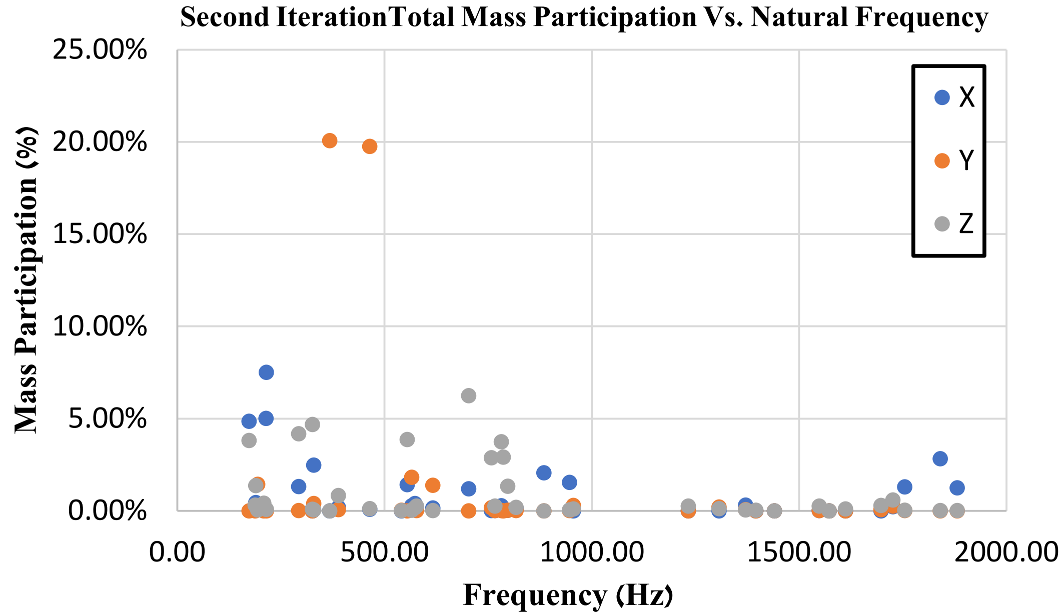


Figure 60: Second Iteration Mass Participation vs. Natural Frequencies

The similarity between this graph and the previous one can tell that most of the masses do not overlap with each other in a single mode shape. In simple words, the masses (parts/components) move in a single direction for most of the mode shapes.

Figure 61 shows the Frequency (Hz) Vs. Mode (#) for the first and second iterations:

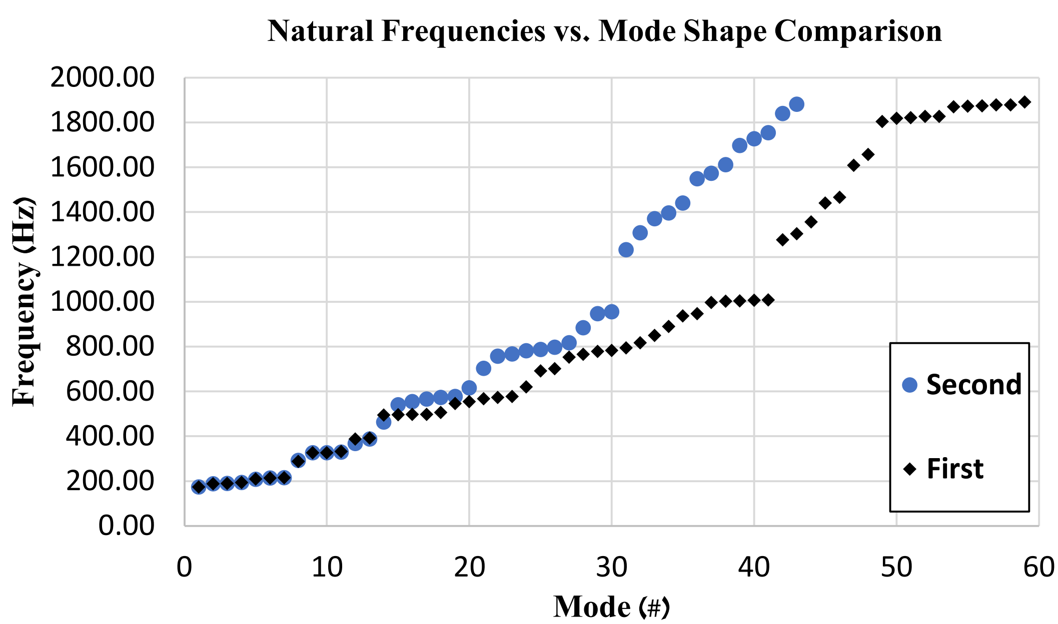


Figure 61: Natural Frequencies vs. Mode Shape Comparison

- First, there are 59 modes under 2000Hz in the first iteration, compared to the second iteration with only 43 modes under 2000Hz. The decreased number of the modes is better for the satellite environment loads withstanding.
- Second, there is a similarity between the iterations' behavior (modes number): from 0Hz to 500Hz the number of modes is similar, then the first iteration starting to have more modes up to 1000Hz, then from 1000Hz to 1200Hz there are no modes in both of the iterations and from 1200Hz to 2000Hz there are more modes in the first iteration.

5.8.8.2 Visual Results

Figure 62 and Figure 63 show a visual demonstration of the mode shapes with more than 10% total mass participation (modes #12 and #14):

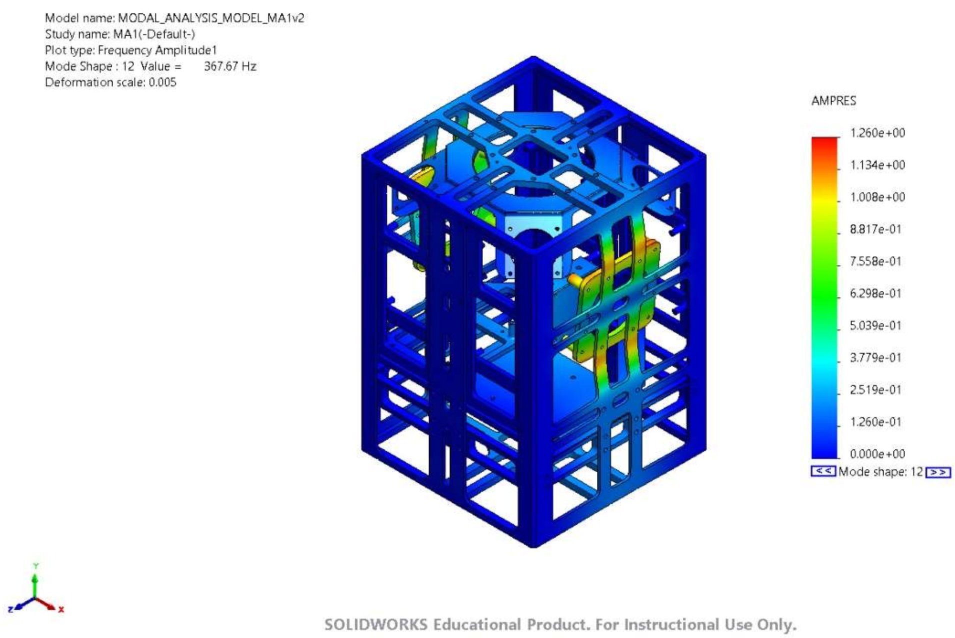


Figure 62: Second Iteration – Mode Shape #12

- Mode: 12
- Natural frequency: 367.7 Hz
- Most affected region: Batteries
- Mass participation: 0% (X), 20% (Y), 0% (Z)
- Highest AMPRES: 1.26

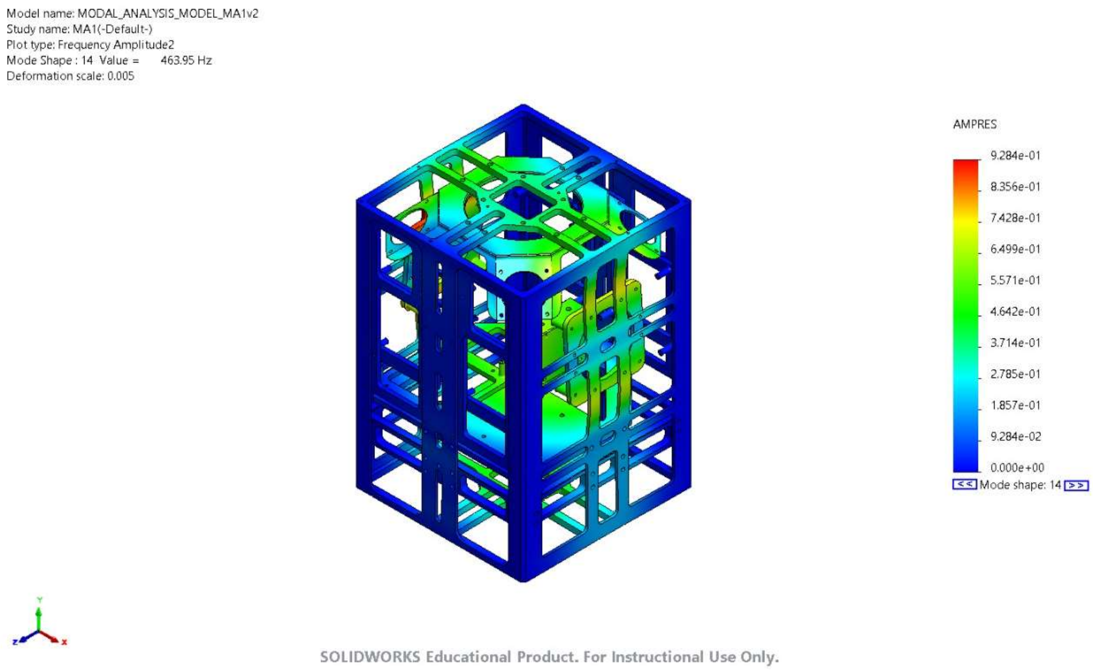


Figure 63: Second Iteration – Mode Shape #14

- Mode: 14
- Natural frequency: 464 Hz
- Most affected region: Batteries, RWs, horizontal plate, Thruster
- Mass participation: 0% (X), 20% (Y), 0% (Z)
- Highest AMPRES: 0.93

5.8.8.3 Structural Summary & Conclusion

- The second modal analysis iteration was successfully performed and simulated the physical structure elements, parts, components, their physical connections, and the mechanical interfaces between them.
- The new model revision, made as a result of the modal analysis campaign, is the updated version of the JERICCO satellite structure. This version will be tested in the forward analysis.
- The change in the thruster bracket caused a significant decrease in the mode shape number, and the mass participation in the significant modes.
- In the range of 0Hz to 2000Hz there are 43 mode shapes, where only 2 of them have more than 10% mass participation (both of them have 20% MP, and in a single direction Y).
- There is no significant mode shape focused on the thruster region, therefore the re-design was successful and has achieved its purpose.
- It is possible to continue with the updated model to the strength analysis campaign – constant load, vibrations, and shock loads.

5.9 3D Print Model

A fundamental objective established early in the mechanical design process was the fabrication of a 3D-printed model representing the final satellite design. This pursuit holds considerable significance as it empowers us to enhance our design intuition by interacting with a physical model, aiding in the identification of subtle issues that might evade detection solely through virtual simulations. Furthermore, the practical assembly of components utilizing the standard tools not only verifies design precision but also assesses real-world assembly feasibility. As part of our ongoing efforts, we've already realized this objective by 3D-printing models for certain components, thereby enabling the construction of a partial, 1:1 scale physical satellite model. The model can be shown in Figure 64 and Figure 65:

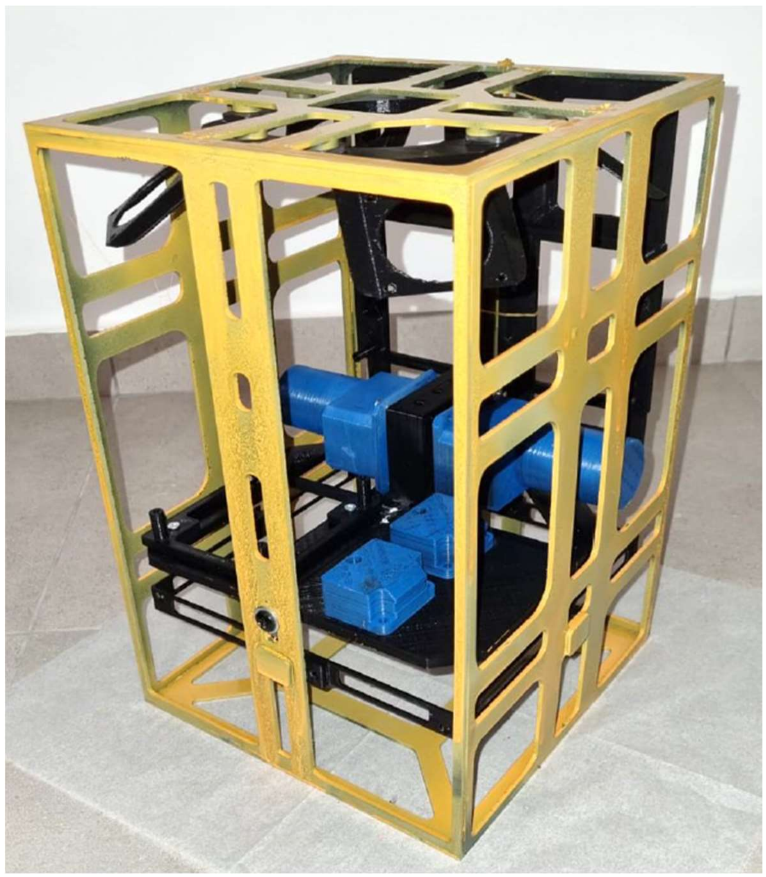


Figure 64: Partial 3D Print Model of JERICCO

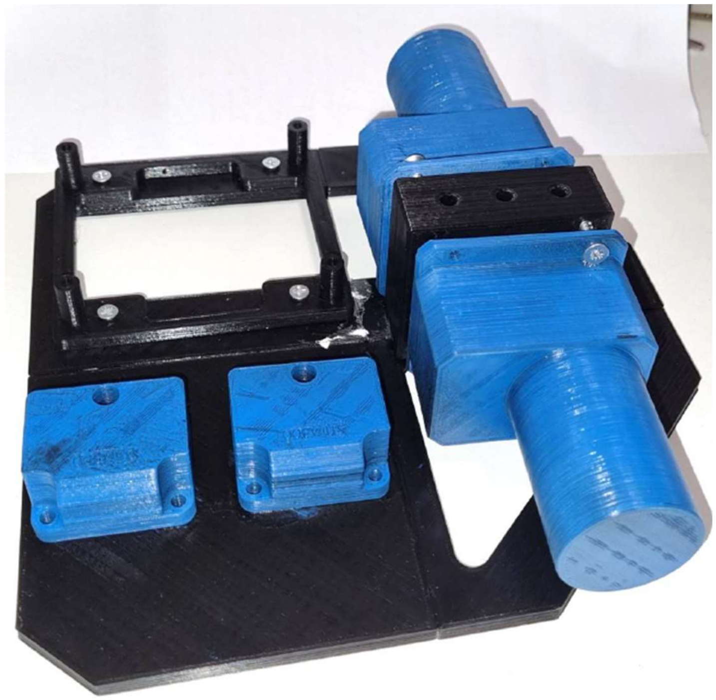


Figure 65: Horizontal plate, 2 IMU, 2 STR 3D Print models

5.10 Risk Management

The main risks are presented in Table 20:

Num.	Description	Probability of realization (1-5)	impact of implementation (1-5)	Risk Level
1	Inability to satisfy the launcher's strength criteria	2	4	**8**
2	Incompatibility during the physical assembly of the satellite	4	5	**20**

Table 20: Risk Management

Strategies to mitigate risk #1:

- Conducting strength analyses.
- Performing environmental testing campaign including random vibrations (simulates launch loads), sine vibrations (simulates launch loads), shocks (simulates the separation from launcher phase), and constant force using a centrifuge (simulates every phase of the mission where acceleration is involved).

Strategies to mitigate risk #2:

- Printing a 3D model to validate the feasibility of the assembly process.
- Integrating and practice tool CAD models into the software environment to simulate their utilization within the assembly procedure.

5.11 Further Work

The main tasks needed to be accomplished in the near future are:

- Generating production drawings encompassing all individual components.
  - **Estimated Time:** 50 hours
- Generating assembly drawings.
  - **Estimated Time:** 35 hours
- Create Assembly Procedure document.
  - **Estimated Time:** 25 hours
- Fabricating an enhanced and more intricate 3D model through printing.
  - **Estimated Time:** 10 hours (+printing time)
- Performing constant force analysis.
  - **Estimated Time:** 30 hours
- Performing shock analysis.
  - **Estimated Time:** 30 hours
- Performing steady sine waves vibration analysis.
  - **Estimated Time:** 35 hours
- Performing random vibrations analysis adhering to launcher or customer specifications.
  - **Estimated Time:** 40 hours
- Analyzing the collected data and compiling comprehensive documentation within the report.
  - **Estimated Time:** 70 hours

Total estimated time to complete all tasks:

325 hours (times were estimated per one student) – about 36 full workdays.

5.12 Summary & Conclusion

Guiding Principles in the Mechanical Design:

Our approach to mechanical design was anchored in fundamental engineering principles. We meticulously adhered to industry standards, considering factors such as structural integrity, material selection, and fabrication techniques. Attention to detail and precision governed our design process, ensuring the satellite's robustness and functionality in its intended environment.

Component Tracking & Documentation:

We adopted a systematic method for tracking and documenting components, meticulously recording their properties and modifications. This method is results in a comprehensive representation of the system, consistency, enhancing analysis accuracy, and establishing a dependable reference for future stages.

Modal Analysis

Our investigation spanned two modal analysis iterations, each meticulously orchestrated to unveil the structural intricacies of the JERICCO satellite. With a focus on accuracy, we dissected mode shapes, natural frequencies, and mass participation across the major axes of the satellite. A pivotal insight emerged from the redesign of the thruster bracket, resulting in optimized mass participation and overall performance enhancement.

Future Prospects

In our efforts, we have established a foundational framework for the ongoing development and mechanical design of the satellite. The upcoming tasks encompass generating production drawings, assembly procedures, and fabricating an enhanced 3D model. Additionally, a series of environmental testing campaigns, including strength analysis, constant force, shocks, and vibrations. Then performing data analysis and meticulous documentation along with a final report.

5.13 References

All of the reference documents are listed below along with a web-link attached to each of them. Additionally, these documents can be found in the main project drive.

[1] Falcon Users Guide 2021-09 (.pdf)

[2] Modal analysis - JERICCO - raw results (.csv)

[3] Screw - Spring model (.csv)

6 Thermal Team

6.1 Abstract

Passive thermal control maintains the satellite's component temperature without using powered equipment, to reduce power and weight. It consists of establishing the necessary thermal parameters involved in the process of heat transfer by radiation and conduction. In this work, we investigate a one and a half years lunar mission. This work deals with the equilibrium thermal problem that determines the range of temperatures to which the different components will be exposed.

One of our main efforts is to improve the accuracy of the possible range of temperature that our components can handle, therefore we are working on improving this by performing and computing analysis of the surrounding conditions.

Our research methods include the algebraic mathematical approach for steady-state and transient analysis that are implemented in MATLAB scripts, for which the output is the temperature range on each face separately. We showed the characteristics of a lunar orbit: the solar, albedo and moon radiation and the radiated heat generated by the inner components. Moreover, the heat flux emitted by the radiator surfaces to outer space was examined.

Overall, our role is essential to the JERICCO project since if the satellite's components cannot support the total radiation, the mission fails.

6.2 Nomenclature

- Solar Particle Events (SPE)
- Low Earth Orbit (LEO)
- Global Cosmic Radiation (GCR)
- Overall Multi-Layer Insulation (MLI)

6.3 Introduction

The Thermal Team's work focuses on the Environmental Conditions during the JERICCO mission. Thermally, the satellite will face extreme conditions resulting in a temperature range of $-130[°C]$ to $120[°C]$. (1)

Radiation in these environmental conditions is treated as a superposition of several effects:

• **SPE** - Highly energetic accelerated particles, mainly protons,

emitted by the sun. These particles are posing a significant radiation hazard to spacecrafts outside of LEO regions.

• **GCR** - Radiation type like SPE in its concept. Highly energetic protons, nuclei and other particles originating in faraway sources and events occurred outside of our solar system, yet primarily inside the milky way, such as supernovae or other solar system's emitted SPE.

• **Albedo Radiation** - Albedo is the fraction of radiation reflected from a radiated surface.

In our case, the earth and the moon reflect radiation back at our spacecraft. Earth's average albedo is 0.31, meaning an average of 31% of the sunlight is reflected back from earth's surface. The moon's average albedo is 0.14, thus it reflects 14% of the sunlight that reaches it.

• **Van Allen radiation belts** - They usually are two donut shaped regions (the two are often combined to one larger belt) surrounding earth in which highly energetic particles are trapped due to earth's magnetic field. This concealed radiation poses a hazardous risk to some space missions.

To achieve a successful mission, JERICCO satellite should withstand the mentioned above conditions throughout the mission's lifetime.

The challenge of meeting this requirement is dealt with by the following means:

• **MLI Coating** - The satellite will be coated in a holed thermal insulator to preserve interior heat and create a first shielding layer against penetrating radiation.

• **Heaters** - Heaters will be considered when thermal analyses are performed. They are high power consumers which we shall avoid using if unnecessary. If used, they will be applied to specific sensitive areas inside the satellite such as propulsion system's propellant or batteries to keep those at their operational temperature ranges.

• **Radiator** - In addition to the overall MLI coating, a highly reflective material surface will be placed on a strategically chosen side of the satellite to reflect reaching radiation and clearing out the heat. This will block some of the radiation from penetrating and prevent the satellite from overheating. It is necessary to make sure not to blind the payload and star trackers while choosing the radiator's location.

According to NASA documents regarding the radiation protection of the Apollo mission astronauts, the doses measured while passing the Van Allen Belts were negligible. (2) For the thermal analysis of our satellite, we needed to confirm this assumption by calculating the Ionized radiation caused by the alpha & beta particles, gamma rays and X-rays, with the help of the "OMERE" software.

To be able to define the size of the radiators, we needed to perform a heat analysis using the Heat Equation for each of the satellite's faces. We started our study with initial assumptions that were made previously for simplicity. We tried to improve the accuracy of the analysis this year by including the internal heat Q_{in}, caused by the electronic components inside the satellite, that was neglected last year. We also calculated the Cumulative Ionizing Radiation (as mentioned above).

Finally, we used very precise data for the heat fluxes where F_{solar}, F_{albedo} and F_{IR} were calculated every minute for full lunar orbit thanks to the powerful "Thermica" software. This analysis was done for all different acting modes of the missions.

6.4 Requirements

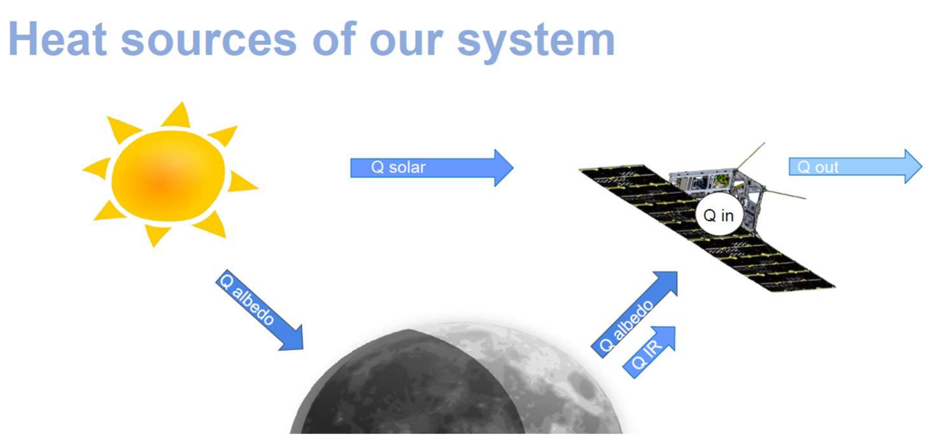


Figure 66 - Heat Sources of our system

All components of the CubeSat have a temperature tolerance range within which they must be kept to meet the operational and survivability criteria outlined for each phase of the mission. The amount of heat that is taken into, stored within, and released from the spacecraft is what determines its temperature. The flow of heat into and out of an orbiting satellite is shown in a simplified drawing shown in **Error! Reference source not found.**.

The following table depicts the operating temperature ranges for each component. The final goal of our Analysis is to compare the acting temperatures to the operating ones. This allows us to check if the components withstand the Deep Space Conditions.

Components	Operating Temperature [C]		Storage Temperature [C]
IMU	[-40 85]		-
STR	[-20 40]		-
Reaction Wheel	[-40 75]		[-40 85]
Battery	Charge	[0 45]	[-20 20]
	Discharge	[-20 60]	
EPS	[-40 85]		[-50 100]
OBC	[-25 65]		-
Transmitter	[-30 57]		-
Receiver	[-40 75]		-
Sun Sensor	[-30 85]		-
Propulsion System	[-10 30]		[-30 65]
Payload	[0 70]		-

Table 21 – The satellite's components and their operating temperature

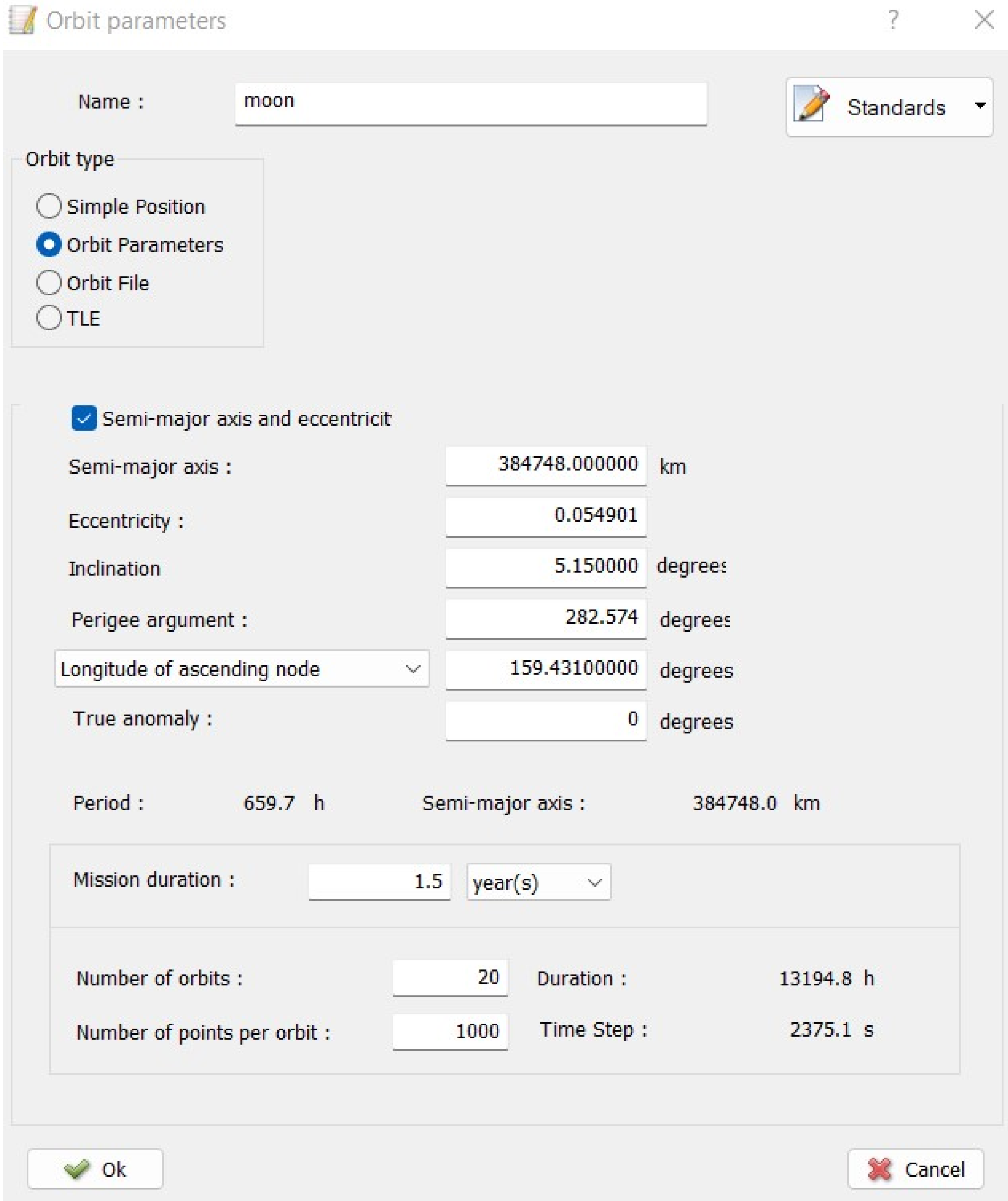
Overall, the objective of the Thermal Team was to ensure the feasibility of the mission and the ability to cope with withstanding space conditions.

6.5 Thermal Analysis

6.5.1 Ionized Radiations

Ionizing radiation is a form of energy that removes electrons from atoms and molecules of materials that include air, water, and living tissue. Ionizing radiation can travel unseen and pass through these materials. Thus, we needed to make sure that the Cumulative Ionized Radiations were not harming the components inside the satellite throughout the mission.

To perform this analysis, we used the software "Omere" that simulated the satellite's orbit and calculates the radiation over time. We chose to model the moon's orbit around Earth since the satellite's orbit is very close to the moon's surface (Low Lunar Orbit) as we can see in Figure 67 – Omere software.


Orbit parameters
Name :
moon
Standards
Orbit type
Simple Position
Orbit Parameters
Orbit File
TLE
Semi-major axis and eccentricit
Semi-major axis :
384748.000000
km
Eccentricity :
0.054901
Inclination
5.150000
degrees
Perigee argument :
282.574
degrees
Longitude of ascending node
159.43100000
degrees
True anomaly :
0
degrees
Period :
659.7 h
Semi-major axis :
384748.0 km
Mission duration :
1.5
year(s)
Number of orbits :
20
Duration :
13194.8 h
Number of points per orbit :
1000
Time Step :
2375.1 s
Ok
Cancel

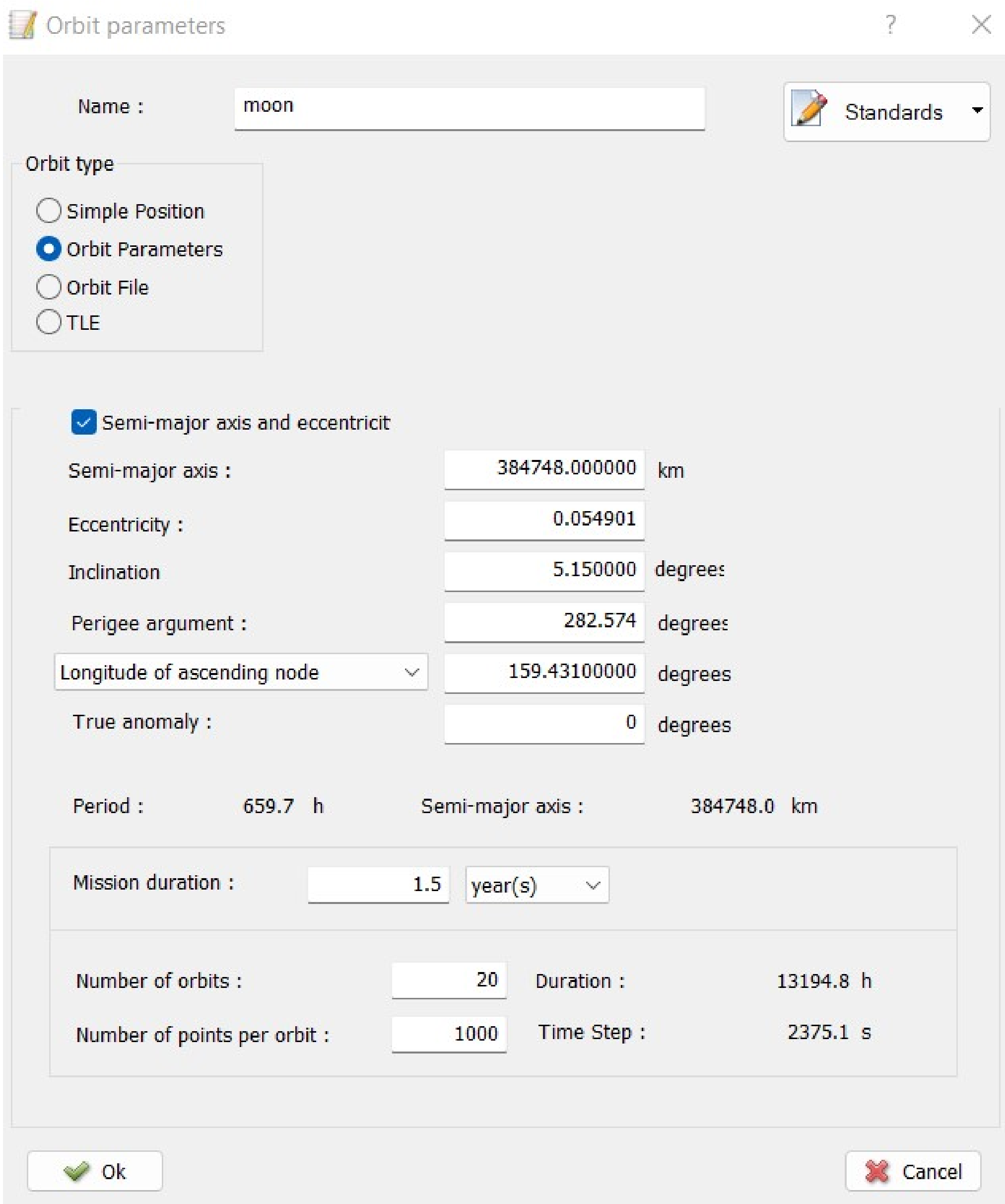


Figure 67 – Omere software

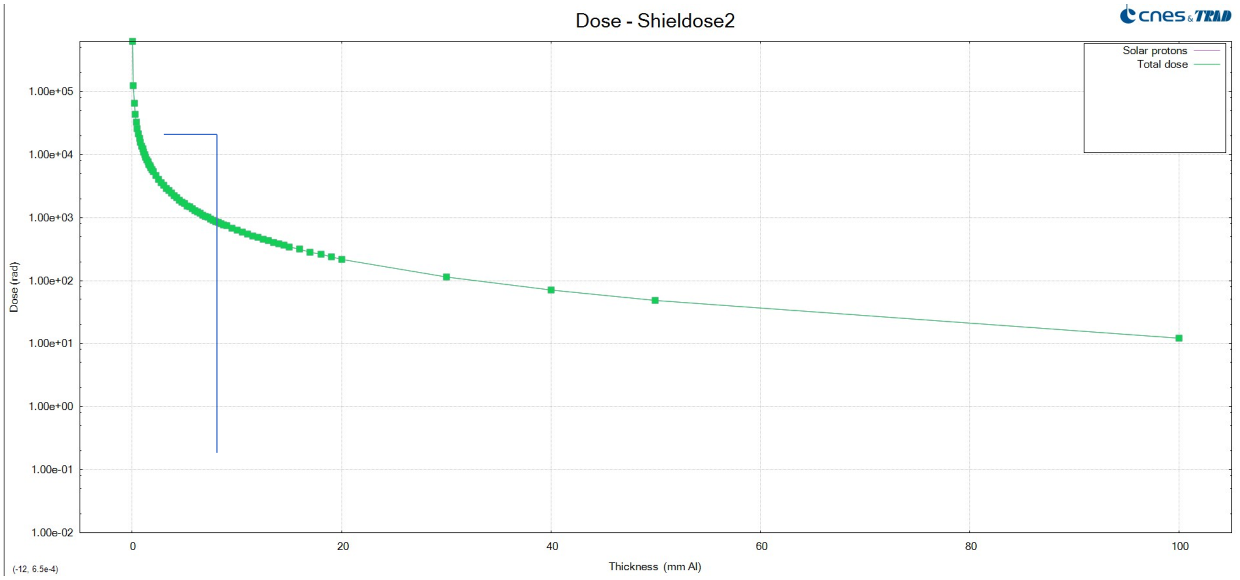


Figure 68 – Dose Depth Curve

In Figure 68, the Dose Depth Curve (DDC) represents the Cumulative Ionizing Radiation that is absorbed by each component inside the satellite for a 1.5-year mission. Knowing that the PBC* is 1.57mm (about 0.06 in) and the box is 0.5mm (about 0.02 in), our wall thickness is equal to 2.7mm (about 0.11 in). Indeed, we can read on the graph that the maximum DDC that our satellite will face is slightly less than 10 [krad]. And since our satellite can support up to 10 krad, we do not cross the limit, so we do not have any issues regarding the ionized radiation.

6.5.2 Internal Heat

Calculating the Internal Heat Dissipation of the electric components is essential to an accurate Thermal Analysis.

Firstly, we can recall from last year [1] the faces of the satellite:

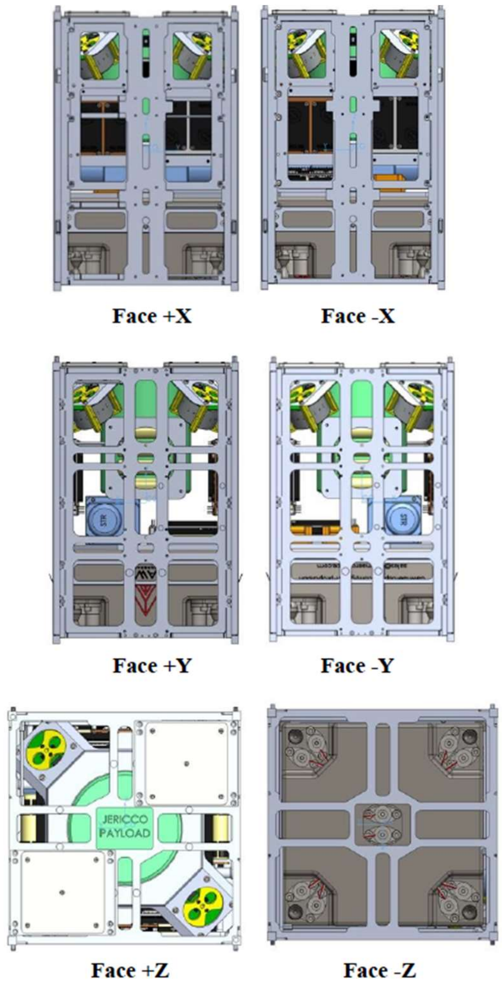


Figure 69 – Satellite's faces

Secondly, together with the avionics team, we built a table of every component and their total power emission for the different operational modes. There are 5 distinct modes that can occur during the mission: Day Cruise, Night Cruise, Mission, Communication and Altitude Control. Then, we computed their action on each face considering their location in the satellite. For example, one EPS is acting on the Face +x, so the total power of the EPS will be considered on this face. However, the Reaction Wheel (considered as one entity) is evenly arranged on top of Face -z in the satellite which implies that 1/6 of its total power is acting on Faces +x, -x, +y, -y and 1/3 on Face -z.

Preliminary Power Budget
02-06-23

	Component		Mode				
	Name	Power [W]	Day cruise	Night Cruise	Mission	Communication	Altitude Control
+x	EPS	0.6	on	on	on	on	on
	Sun Sensor	0.0759	on	off	on	on	on
	1/6 Propulstion System	3.33	off	off	off	off	on
	1/4 2 x IMU	1.00	on	on	on	on	on
	1/4 S band transmitter	0.55	off	off	off	on	off
	1/4 2 x STR	0.55	on	on	on	on	on
	S band receiver	2.20	on	on	on	off	on
	OBC	1	on	on	on	on	on
	Total Power [W]	**9.3092333**	**5.43**	**5.35**	**5.43**	**3.23**	**7.76**
-x	EPS	0.6	on	on	on	on	on
	Sun Sensor	0.0759	on	off	on	on	on
	1/6 Propulstion System	3.33	off	off	off	off	on
	1/4 2 x IMU	1.00	on	on	on	on	on
	1/4 S band transmitter	0.55	off	off	off	on	off
	1/4 2 x STR	0.55	on	on	on	on	on
	S band receiver	2.20	on	on	on	off	on
	OBC	1	on	on	on	on	on
	Total Power [W]	**9.3092333**	**5.43**	**5.35**	**5.43**	**3.78**	**8.76**
+y	battery	6.064	on	on	on	on	on
	1/6 Propulstion System	3.3333333	off	off	off	off	on
	1/4 2 x IMU	1.00	on	on	on	on	on
	1/4 S band transmitter	0.55	off	off	off	on	off
	1/4 2 x STR	0.55	on	on	on	on	on
	Total Power [W]	**11.497333**	**7.61**	**7.61**	**7.61**	**8.16**	**10.95**
-y	battery	6.064	on	on	on	on	on
	1/6 Propulstion System	3.3333333	off	off	off	off	on
	1/4 2 x IMU	1.00	on	on	on	on	on
	1/4 S band transmitter	0.55	off	off	off	on	off
	1/4 2 x STR	0.55	on	on	on	on	on
	Total Power [W]	**11.497333**	**7.61**	**7.61**	**7.61**	**8.16**	**10.95**
+z	payload-camera	3.5	off	off	on	off	on
	4 x Reaction Wheels	7.36	on	on	on	on	on
	2 x S band antena	4.40	on	on	on	on	on
	Total Power [W]	**15.26**	**11.76**	**11.76**	**15.26**	**11.76**	**15.26**
-z	1/3 Propulstion System	6.6666667	off	off	off	off	on
	Total Power [W]	**6.6666667**	**0**	**0**	**0**	**0**	**6.666666667**

Table 22 – Preliminary Power Budget

With this table we obtained the internal heat denoted as Q_{in} for each face of the satellite for each mode of the mission. This value is meaningful since it is used in the Heat equation stated in eq.1. This table should be reviewed with the new structure team and recalculated as there may be errors with the X band receiver and transmitter since they were not taken into account.

6.5.2.1 Assumptions

1. **Heat transfer into the satellite**

We assumed that the heat fluxes transfer into the satellite via the radiator's surfaces and the rest of the faces are completely isolated. As a result, the thermal balance will be executed on the radiators.

2. **Heat Capacity**

We assumed that the heat capacity of the whole satellite is equal to the heat capacity of aluminum which is: Cp = 921 $\left[{}^{J}/_{Kg\,\cdot\,K}\right]$.

3. **The mass of the satellite**

We assumed that the current value of the satellite's mass was equal to 15.3 [kg]. This value does not include the brackets built by the Design team.

4. **The View Factor**

We obtained the view factor's data values from our mentor (Zeev Sherman) for the thermal analysis. For each iteration, we got the $F_{Solar}, F_{Albedo}, F_{IR}$.

5. **The Radiator emissivity and absorptivity**

We assumed in our calculations that the average values of emissivity and absorptivity are as follows: ε = 0.82, α = 0.08.

6. **The initial guess for the temperature**

We assumed $T_{Initial} = 30°C$. It was chosen based on the component's requirements.

6.5.2.2 The Heat equation

$$\frac{dT}{dt} = \frac{1}{mC_p}\left(A_{rad}\left(\alpha F_{solar} + \alpha F_{albedo} + \epsilon F_{IR}\right) - \sigma\epsilon T^4 + Q_{in}\right) \tag{17}$$

T - Temperature [K]

t - Time [sec]

m - Mass of the satellite $[kg]$

Cp – Specific heat capacity of the satellite $\left[{}^{\mathrm{J}}/_{\mathrm{Kg}\cdot\mathrm{K}}\right]$

α - Absorptivity of the radiator surface

ϵ - Emissivity of the radiator surface

$\mathbf{Arad, j}$ - The radiator area for the j face $[\mathrm{m}^2]$

$\mathbf{F_i}$ - View factor for the irradiation $\left[{}^{\mathrm{W}}/_{\mathrm{m}^2}\right]$

$\boldsymbol{\sigma}_{\mathbf{rad}}$ – Stefan Boltzmann's constant $\left[{}^{\mathrm{W}}/_{\mathrm{m}^2\cdot\mathrm{K}^4}\right]$

$\mathbf{Q_{in}}$ - Internal Heat rate of the satellite $[W]$

6.5.3 Analytical & Numerical Calculations

The heat equation stated above was used to perform the heat analysis of the satellite. We computed twice the equation for two different situations: when all the components are acting in the satellite and when these components act separately relative to the mission mode.

The point of the analysis was to check if the satellite can withstand deep space conditions.

6.5.3.1 General mode analysis - All components are acting

When all the components are acting, we obtained from Table 22 – Preliminary Power Budget, an internal heat matrix for each face such that:

$$\mathrm{Q_{in}} = [9.3092, 9.3092, 11.4973, 11.4973, 15.26, 6.667]\ W.$$

We input this matrix into the heat equation and ran the analysis first on Excel and then on MATLAB.

6.5.3.2 Excel Analysis

The Excel analysis was running for a single orbit of the satellite $(\mathbf{7650}\ [\mathbf{sec}])$ and helped us determine the radiator's area needed for each face. We received precise data for the area absorbing the different radiation per minute per face. The excel file looked like the following:

alpha=0.08	epsilon=0.82	area_rad=0.077[m^2]	direction: +x	Q=9.31[w]	T=293 [k] initial assumption				
		0.077							
Albedo flux [W/M^2]	Albedo absorbed [W]	Solar flux [W/m^2]	Solar absorbed [W]	IR flux [W/m^2]	IR absorbed [W]	TOTAL IN WAT	TOTAL OUT WAT	NET WAT	T [k]
39.47918	0.243191749	0	0	330.2705	20.85327937	30.40647112	-26.38506193	4.021409189	293
43.13667	0.265721887	0	0	360.1978	22.74288909	32.31861098	-26.39122961	5.927381371	293.0171212
46.7388	0.287911008	0.00E+00	0	390.0822	24.62979011	34.22770112	-26.40032247	7.827378647	293.042357
50.14545	0.308895972	0	0	418.8919	26.44883457	36.06773054	-26.41233361	9.655396931	293.0756821
53.46782	0.329361771	0.00E+00	0	447.6956	28.26750018	37.90686196	-26.42715549	11.47970646	293.11679
56.6934	0.349231344	0	0	476.502	30.08633628	39.74556762	-26.44478597	13.30078166	293.1656648
59.73442	0.367964027	0	0	504.697	31.86656858	41.54453261	-26.46522427	15.07930834	293.222293

Table 23 – Excel Analysis

Afterwards, we computed the heat equation for one lunar orbit, and we plotted the temperature as a function of time. For example, face +x and face -z of the satellite had the plots depicted below:

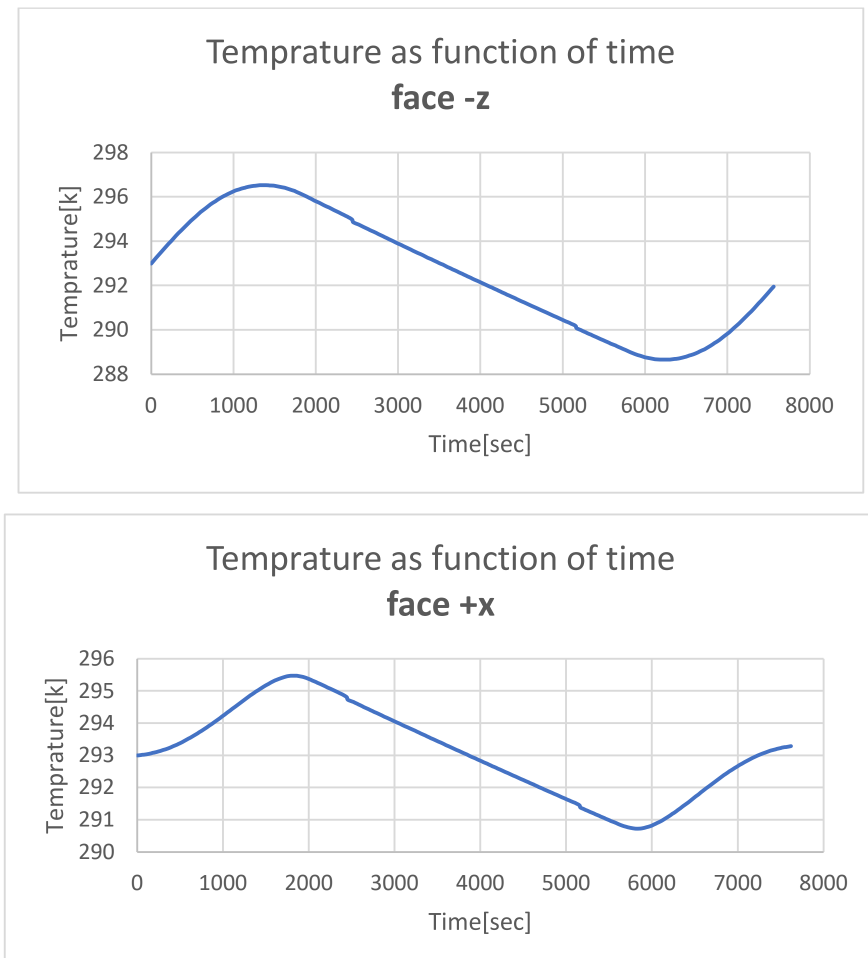


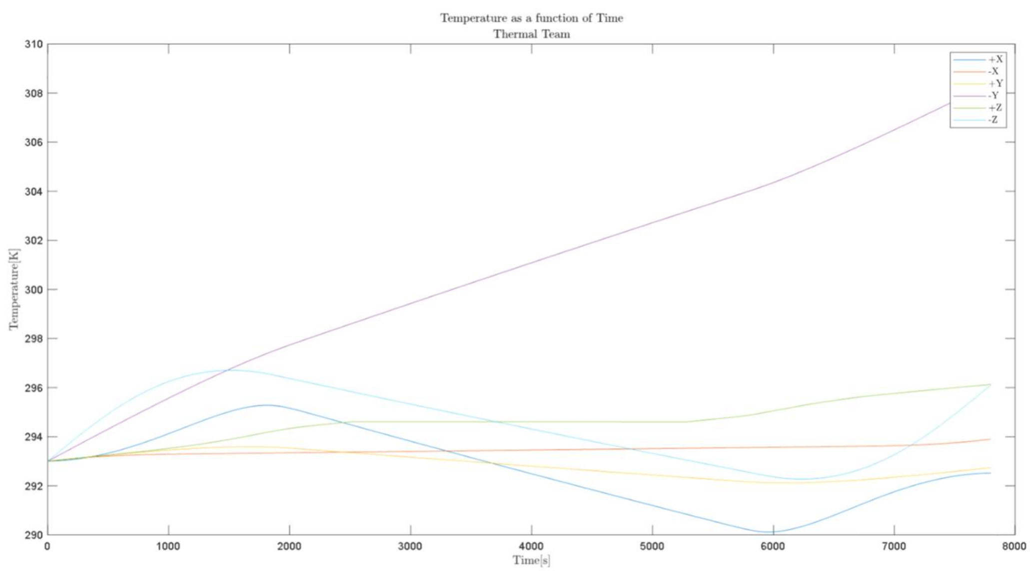
Figure 70 – Plots of the temperature as a function of time

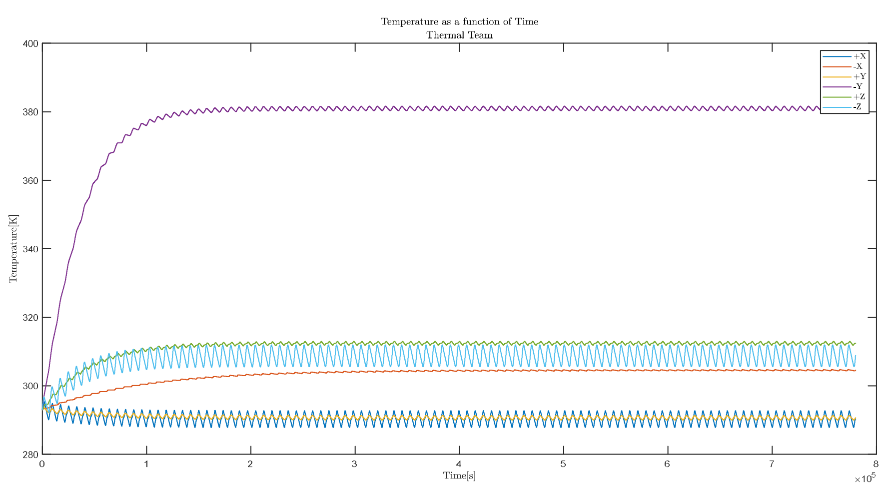
With the help of the obtained plots, we could try different radiator's area until we saw the function converging to a reasonable temperature. We repeated this step for all faces until we obtained the final matrix: $\mathrm{A_{rad}} = [0.077, 0.0295, 0.047, 0.0465, 0.061, 0.077]\ m^2$

6.5.3.3 MATLAB Analysis

We added the two matrices we obtained before $(\mathrm{Q_{in}}, \mathrm{A_{rad}})$, and inputted them in a Matlab file for 1 run. That allowed us to compare to the excel results and for 100 runs to have a more general view of the behavior of these functions.

We plotted the results on the same graphs as before, temperature as a function of time, and we obtained the following results:





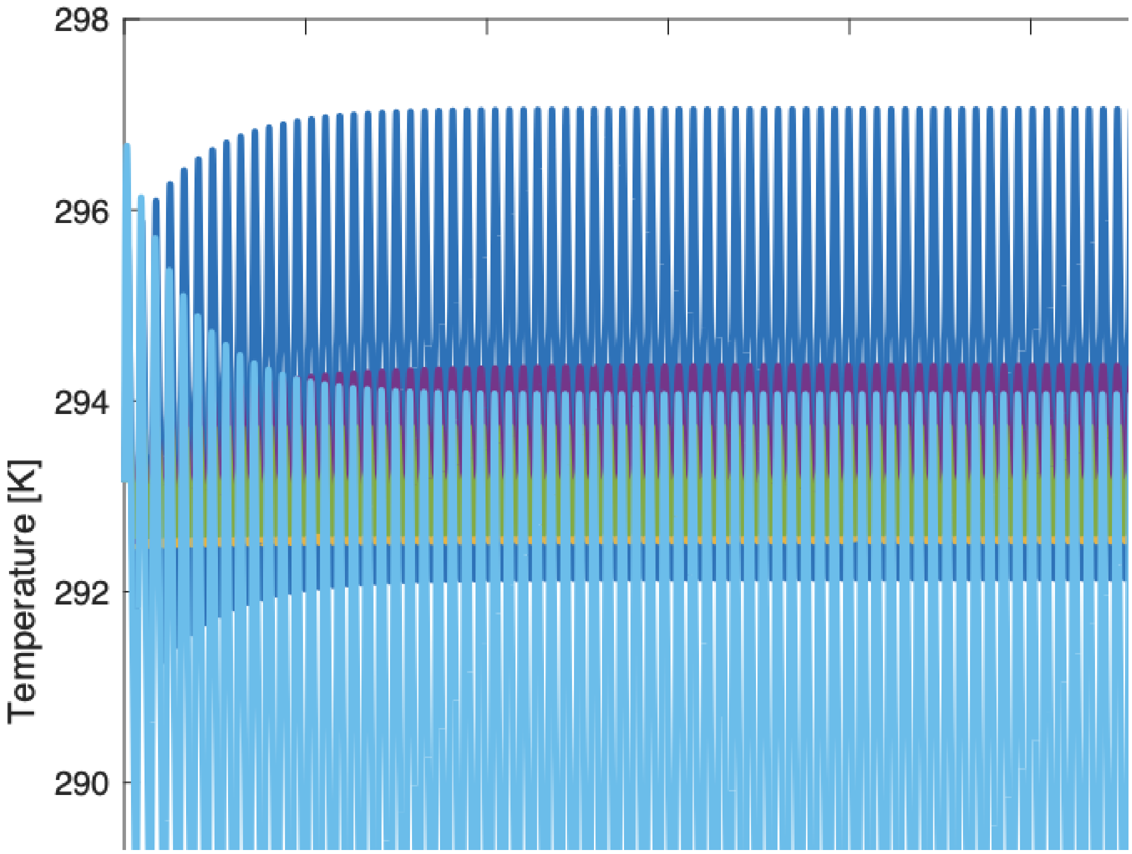


Figure 71 – Matlab Analysis

6.5.3.4 Separated modes analysis

As was mentioned above, together with the avionics team we made a table, Table 22 – Preliminary Power Budget, of acting components per mode and their total power on each face. We wanted to consider the idea that the components don't always work together. Therefore, we constructed a 5x6 matrix for the internal heat of each face for each state such that:

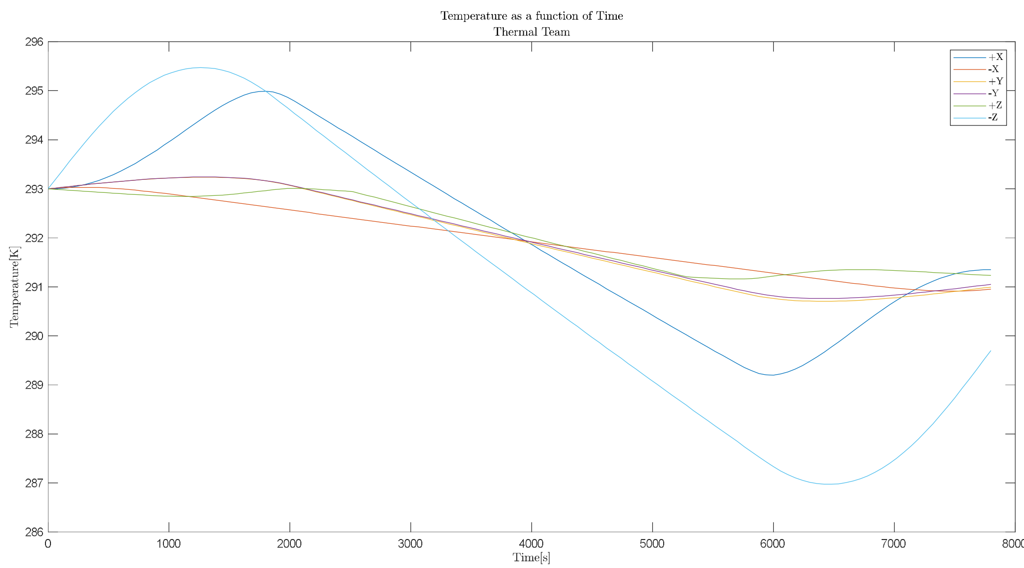
$$Q_{in_{new}} = \begin{bmatrix} \text{mode 1, face 1} & \cdots & \text{mode 1, face 6} \\ \vdots & \ddots & \vdots \\ \text{mode 5, face 1} & \cdots & \text{mode 5, face 6} \end{bmatrix} = \begin{bmatrix} 5.43 & 5.43 & 7.61 & 7.61 & 11.76 & 0 \\ 5.43 & 5.35 & 7.61 & 7.61 & 11.76 & 0 \\ 5.43 & 5.43 & 7.61 & 7.61 & 15.26 & 0 \\ 3.23 & 3.78 & 8.16 & 8.16 & 11.76 & 0 \\ 7.76 & 8.76 & 10.95 & 10.95 & 15.26 & 6.67 \end{bmatrix}$$

$$A_{rad} = [0.077, 0.0295, 0.047, 0.0465, 0.061, 0.077]$$

Where rows are total heat rate at a specific face for each mode and columns are total heat rate for a specific mode for each face.

6.5.3.5 MATLAB Analysis

We added the two matrices we obtained; $Q_{in_{new}}, A_{rad_{new}}$, and inputted them in a Matlab file for 1 orbit and 100 orbits. This allowed us to have a general view of the behaviour of these functions. We plotted the results as temperature as a function of time graph, and we obtained the following results:



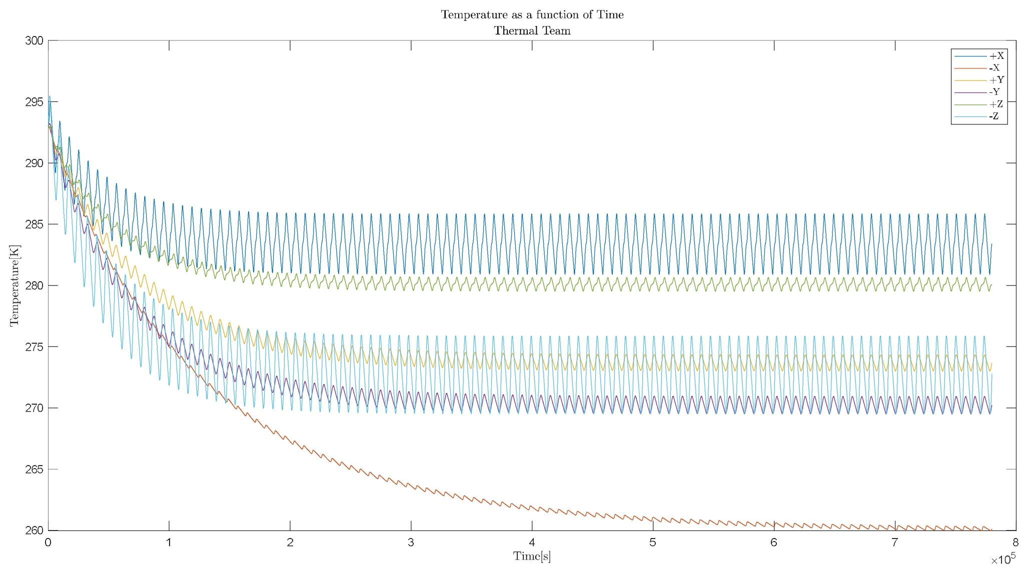
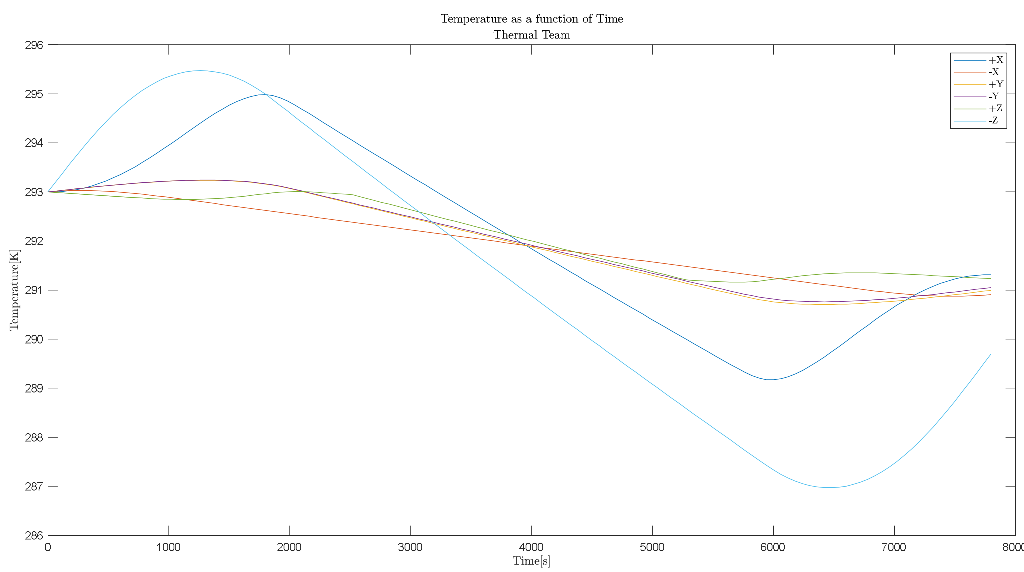


Figure 72 – Day Cruise



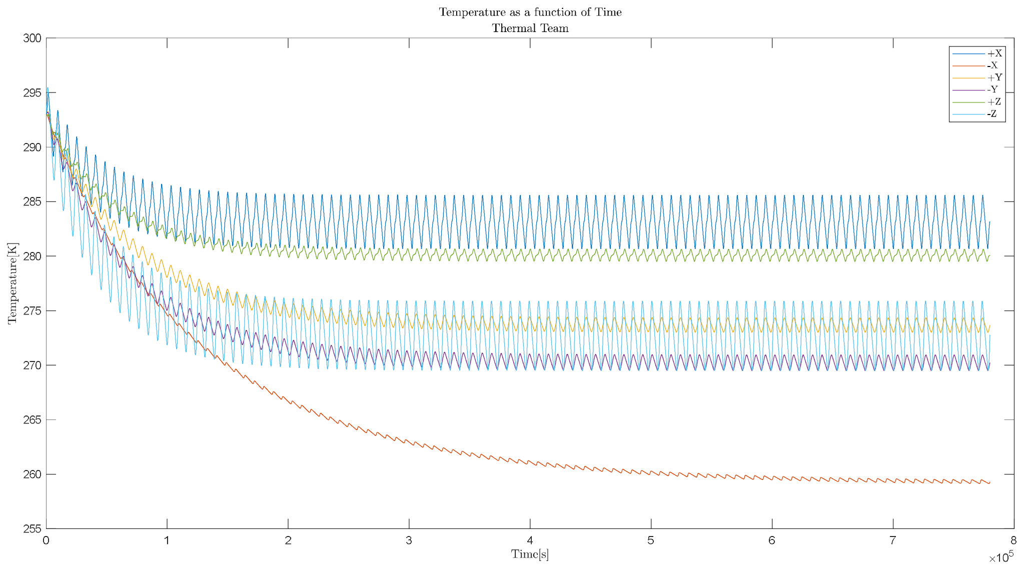
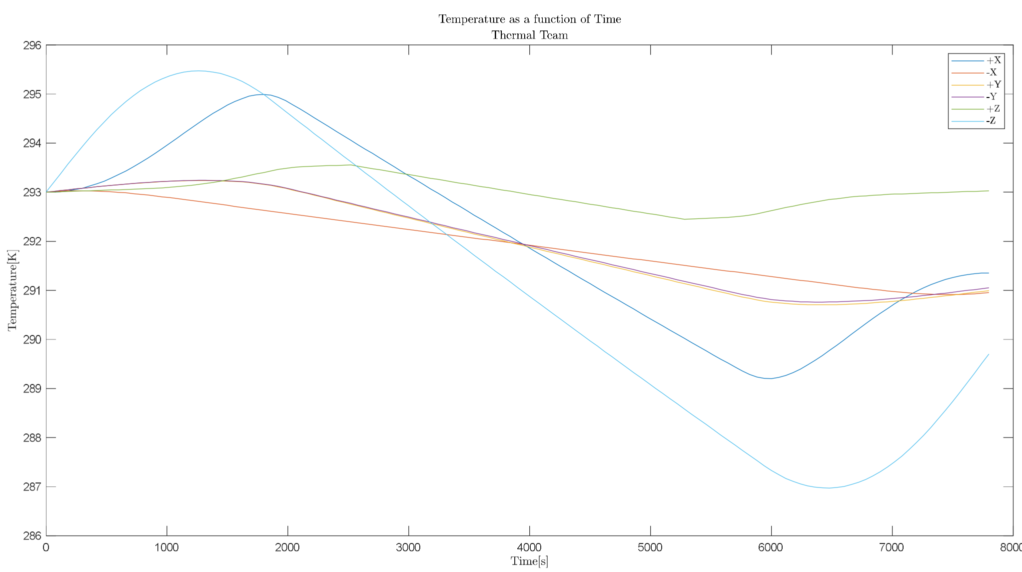


Figure 73 – Night Cruise



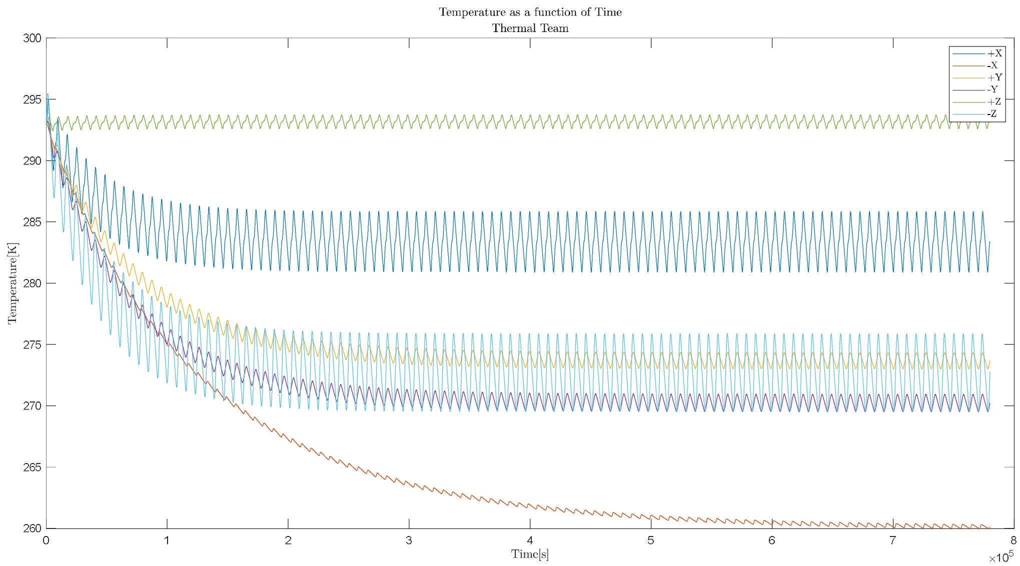
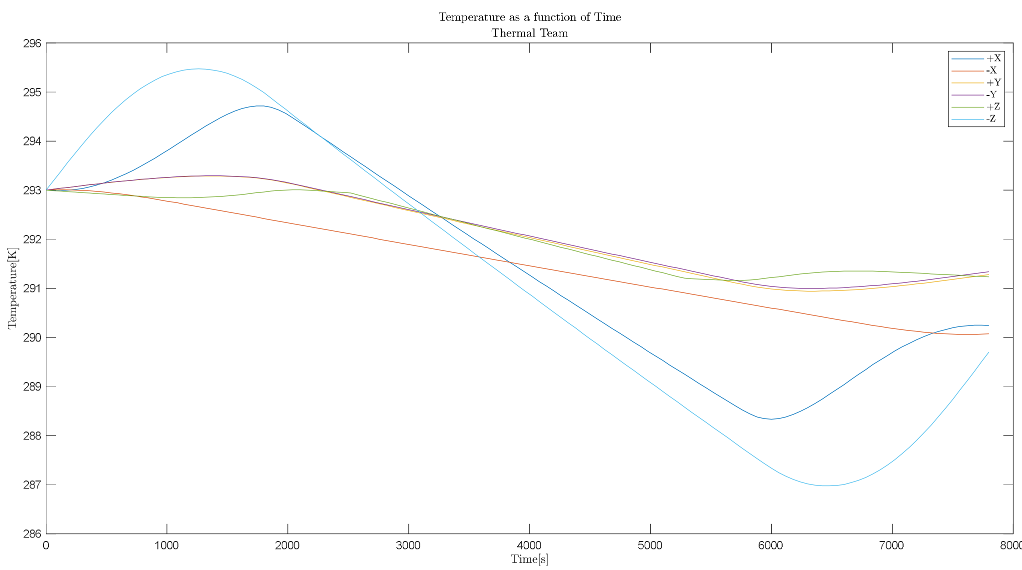


Figure 74 – Mission



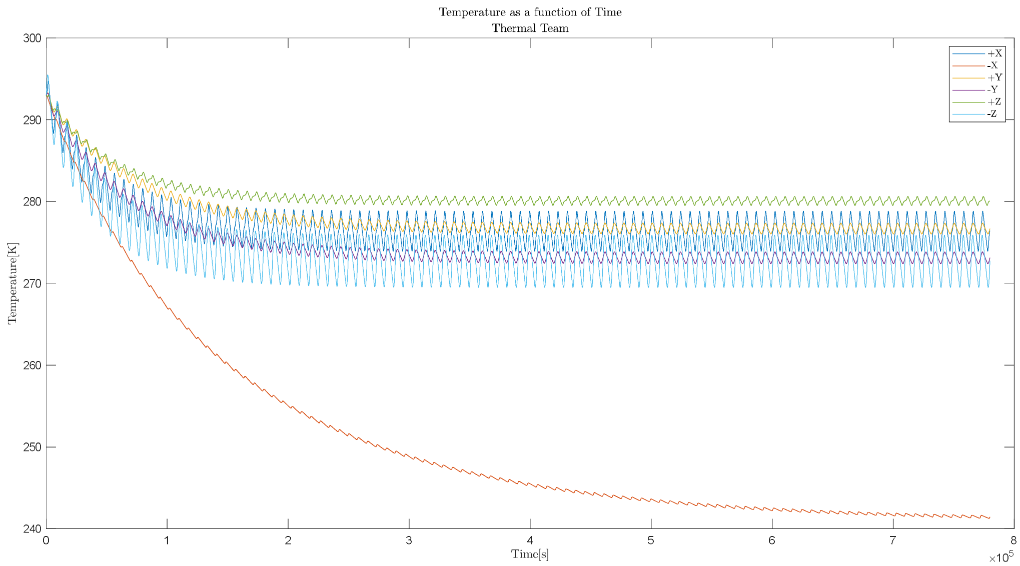
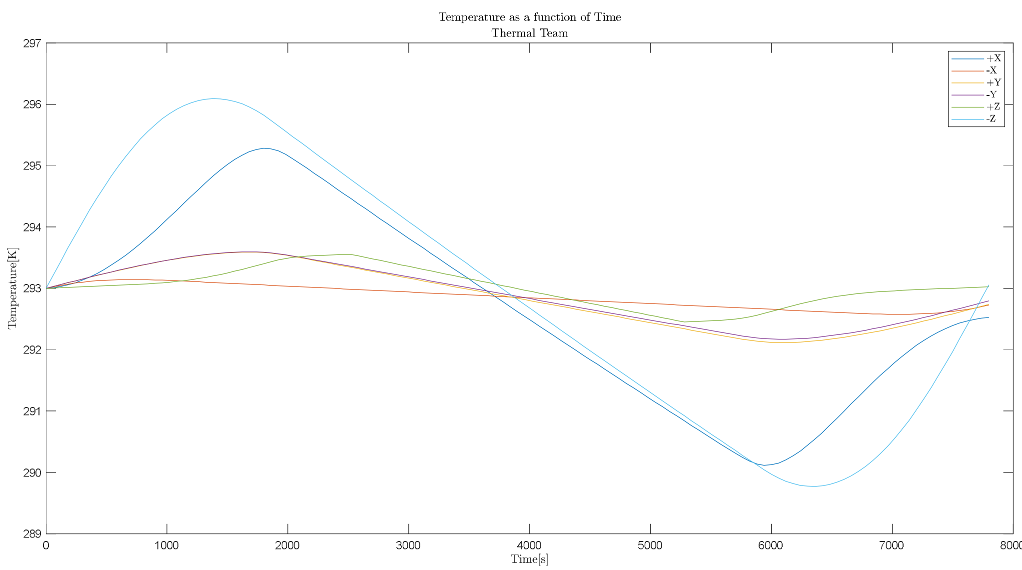


Figure 75 – Communication



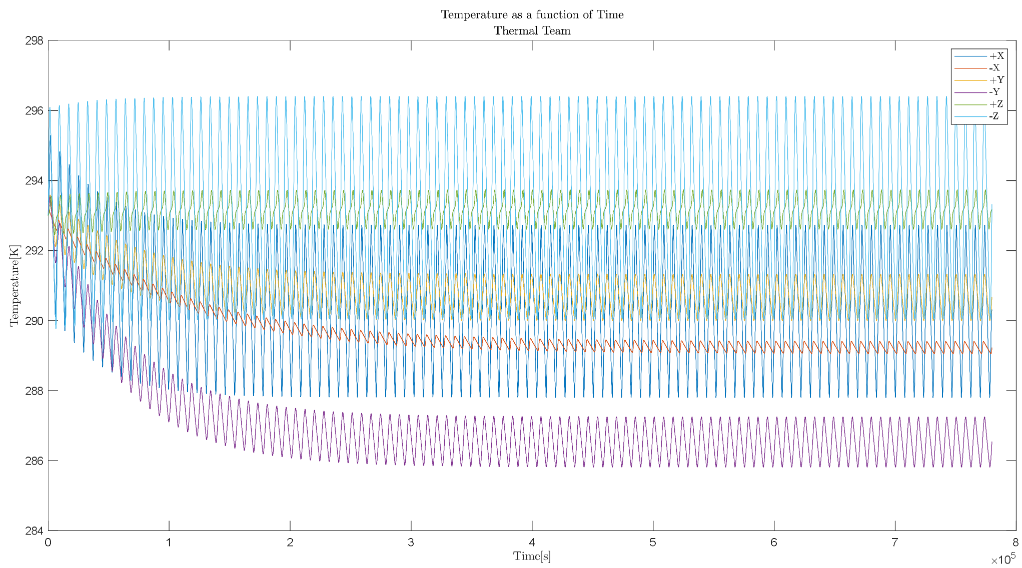


Figure 76 – Attitude Control

6.5.4 Results

In Figure 71, it can be seen that when all the components are acting, the range of values for all the faces is satisfactory and that there is convergence and non-divergence, accept from face -y which is overcooling because it is facing heat sources and its radiator is not able to reduce this heat.

A possible solution when the faces may overcool or overheat, as we can see in Figure 72-Figure 76, is adding heater to specific face to heat it or think about heat transfer pipes to conduct the heat to other faces and thus cooling a desired face. The main reason for this phenomenon that we see in these plots can be the fact that our satellite's face -x is not facing heat sources like the sun or the moon and therefore overcooling.

6.6 Conclusions

We can conclude from this current work that the ionizing radiation does not affect our satellite components and that they can withstand it. Furthermore, our thermal analysis is satisfactory and successful.

However, it is important to mention that our current results depend on the specific power we've chosen for our payload, which is currently minimized (3.5W). If we change it, we have to perform a new analysis and find new radiator areas.

Moreover, we will refer to the risks of changing our payload:

- If the payload power is maximized ($30[W]$):

Only face +z changes for which we modified the radiator area to $0.167[m^2]$ instead of $0.061[m^2]$. However, the size of this face of the satellite is $0.077\ [m^2]$, therefore we cannot go above this size. So, a $30[W]$ payload is way too high because we want the temperature of the plot to converge to the same value as the starting point.

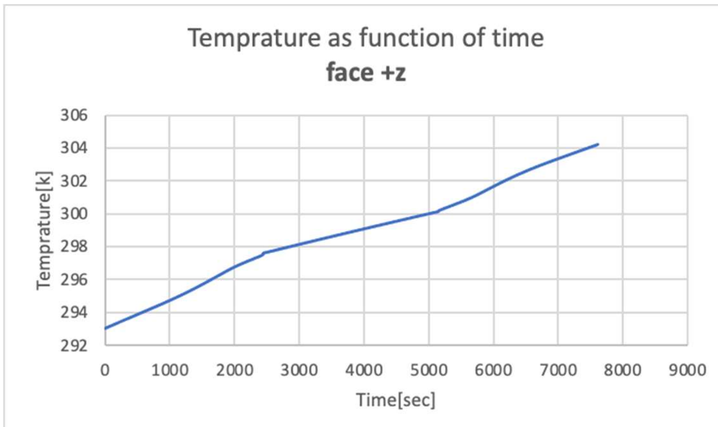


Figure 77 – $A_{rad} = 0.077[m^2]$ *and* $p_{payload} = 30[W]$

Therefore, the maximum total power possible acting on face +z when all the components are acting and if we take a radiator that is the size of the satellite's face, is $19.5[W]$. Therefore, we cannot use a payload with a higher power than $7.74[W]$. We can recall the power budget table for Face +z: $19.5 - 7.36 - 4.4 = 7.74[W]$.

+z	payload-camera	
	4 x Reaction Wheels	7.36
	2 x S band antena	4.40
Total Power [W]		19.5

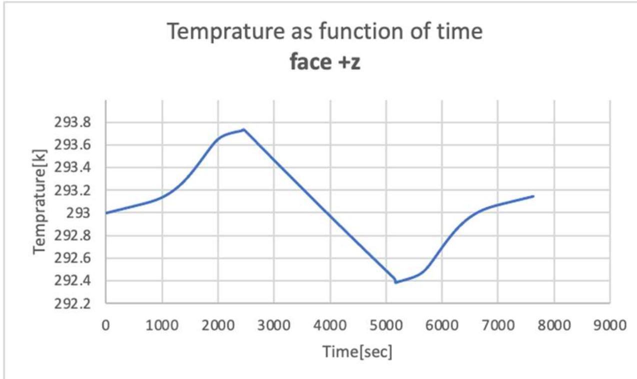


Figure 78 - $A_{rad} = 0.077[m^2]$ *and* $p_{payload} = 7.74[W]$

- If the payload power is minimized $(3.5[W])$:

We have the same results as our analysis which was a radiator area of $0.061[m^2]$.

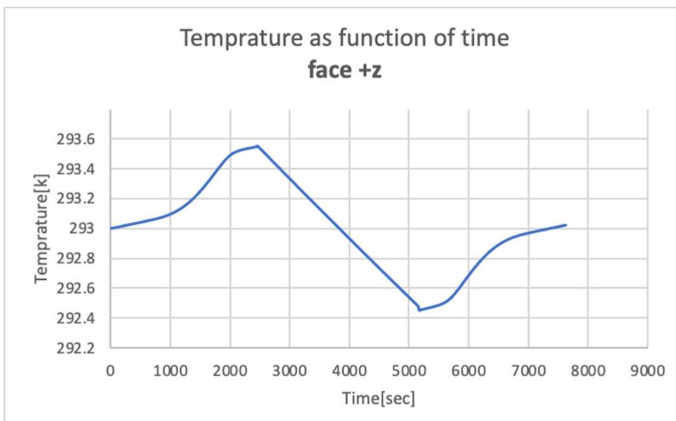


Figure 79 – $A_{rad} = 0.061[m^2]$ *and* $p_{payload} = 3.5[W]$

6.7 Future Work

For the future work of the next team, we suggest they:

- Solve the optimal radiator size when the components act separately: during the upcoming winter semester (12 weeks).
- Analyze if the satellite can withstand cold cases: 3 weeks in the next spring semester.
- Improve the calculations of the internal heat: continuous work throughout the year at every step of the calculations. We always try to improve the accuracy of the data.
- Perform a material analysis with the Structure team on Solid Works: 6 weeks.

References

1. Final Year Report, FDR - JERICCO Project 2021/2022
2. https://www.nasa.gov/sites/default/files/files/SMIII_Problem7.pdf

7 Control and propulsion

7.1 Nomenclature

• ADCS- Attitude Determination and Control System

• STR- Star Tracker

7.2 Abstract

The satellite's control system is responsible for the stabilization and orientation of the satellite, as well as orbital elements determination. The satellite's propulsion system performs as the actuator alongside the reaction wheels. Our team is working on analyzing, simulating and integrating the propulsion system into the control architecture. This abstract summarizes the work done by Jerrico's previous Control and Propulsion team and focuses on the advancements made by this year's team over the past year. The work done on this year's project was achieved mainly thanks to self-learning, aided by external resources and professionals in the relevant areas of expertise. At first, we focused on optimizing the cruise control loop performance, resulting in much improved settling times. Then we studied the chosen propulsion system in depth and designed a propulsion-system-based architecture for altitude control for maneuver mode. Meanwhile, we managed to convert the cruise mode wheel-based loop to fit the requirements of the detumbling control loop. Lastly, we began the design of the sun search algorithm, which is an integral part of our control architecture and will help lay the foundations for future work on unintegrated ADCS components. We have managed to integrate the propulsion system in the cruise and detumbling controllers, and optimize the cruise control loop performance. We also designed a schematic base for a hybrid control loop which is both wheel-based and thrusters-based, to ensure optimal fuel consumption. Additionally, we fulfilled some of the system engineering requirements, which helped focus our objectives and control designs. Also, some unclear documentation and Simulink annotation were improved by the team over the year.

7.3 Introduction

The satellite's control system is responsible for the stabilization and orientation of the satellite, as well as orbital elements determination and corrections. The satellite's propulsion system performs as the actuator alongside the reaction wheels. Our team is working on analyzing, simulating and integrating the propulsion system into the control architecture.

The previous year's work included a trade study of various control and propulsion components. Additional significant achievements were the design of a reaction-wheel-based architecture for cruise mode attitude control, development of kinematics and dynamics equations for the satellite, and determination of the optimal reaction wheel positioning configuration.[1]

This year's work will be detailed in the following chapters. Optimization of the cruise control loop will be presented in chapter 7.4. Additionally, in chapter 7.5 we will present adaptation, analysis and development of the wheel-based loop for detumbling control. Our work in chapter 7.5.3, integrates the propulsion system, improves control performance, and provides essential tools for future ADCS work. Furthermore, in chapters 7.5.4 and 7.6 we will propose our ideas for future work.

7.4 Optimizing Cruise Control

7.4.1 Cruise Control- Old Approach

Last year's team built the infrastructure for cruise control, which is based on reaction wheels. They applied the following control law (18):

$$\boldsymbol{u(t) = -K_p \cdot sign\big(\delta q_4(t)\big)\, \delta q_{1:3} - K_d \cdot \omega(t)} \tag{18}$$

Where K_p and K_d denote the proportional and derivative gains respectively, and δq denotes the error quaternion, a further explanation and the full derivation can be found in previous year's report [1].

The gains for the control law above were chosen by the following formula (19):

$$\begin{cases} K_{p_i} = 2\omega_n^2 \cdot J_{ii} \\ K_{d_i} = 2\zeta\omega_n J_{ii} \end{cases} \tag{19}$$

While J_{ii} is the primary moment of inertia for each corresponding axis. This is the result of an attitude change from $[0,0,0]$ to $[15°, 15°, 15°]$ and $[30°, 30°, 30°]$ with the former controller:

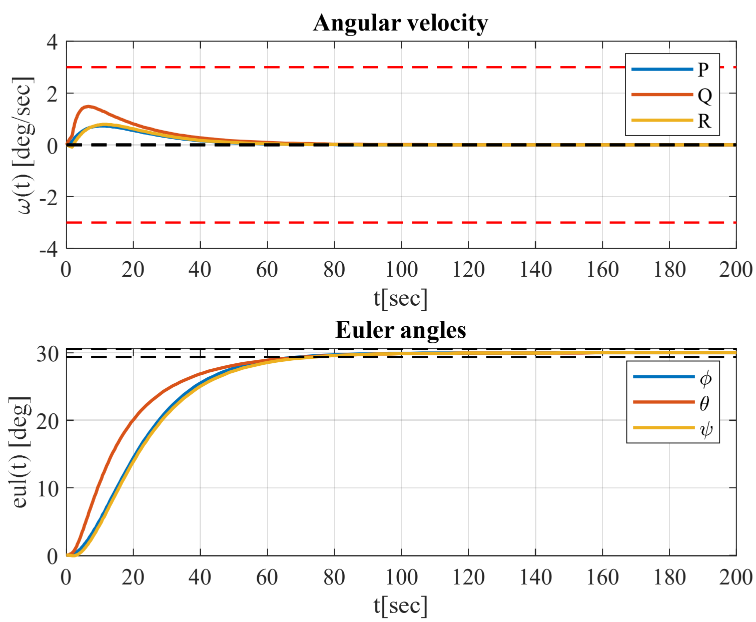


Figure 80-Cruise Controller Pre-Tuning

7.4.2 Cruise Control- New Approach

In continuation for the last year's work, the new requirement for the cruise control loop was to reach from one attitude to another in the quickest way possible. We started by analyzing the performance of the existing control loop. The controller worked, but slowly. We decided that the settling time could, and should be improved by applying the following changes:

Increasing the derivative gain K_d

Using the corresponding primary moments of inertia, instead of taking the largest primary moment of inertia and applying it to the other axes

The following results were obtained:

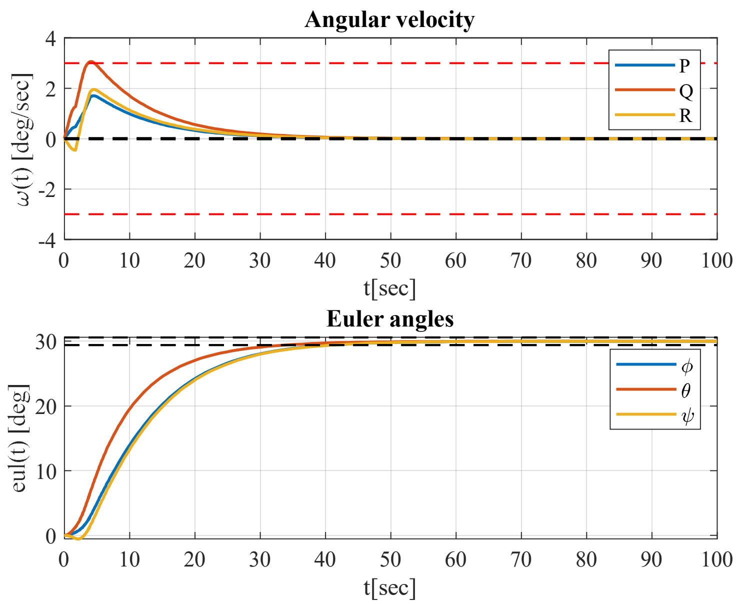


Figure 81-Cruise Controller Post-Tuning

Detailed optimized results:

	$Settling\ Time_{pre}$	$Settling\ Time_{new}$	$Improvement$
$[30°, 30°, 30°]$	$74.58\ [sec]$	$42.11\ [sec]$	-43.53%

By examining the different gains and comparing them, the following gains were finally chosen:

$$\boldsymbol{K_d = 20 \cdot \zeta\omega_n \cdot J_{ii} \ , K_p = 2 \cdot \omega_n^2 \cdot J_{ii}} \quad \textbf{(20)}$$

Where

$$\begin{cases} \boldsymbol{\zeta = 0.225} \\ \boldsymbol{\omega_n = 0.375[Hz]} \end{cases} \quad \textbf{(21)}$$

Throughout the performance analysis we verified that the angular momentum limitations, caused by the reaction wheels' mechanical thresholds, were not exceeded.

7.5 Detumbling Control

As the satellite detaches from the mothership, it will receive a random angular velocity vector and begin to rotate around a random axis. Here, the detumbling controller comes into play, the role of this controller is to stop said rotation, demanding 0 total angular velocity at the end of the detumbling phase.

7.5.1 Detumbling Control Loop-Simulink

After analyzing the existing control architecture model, built in Simulink, we decided that it can serve as a good basis for the detumbling controller, although some critical changes are required. The detumbling control loop Simulink model is below:

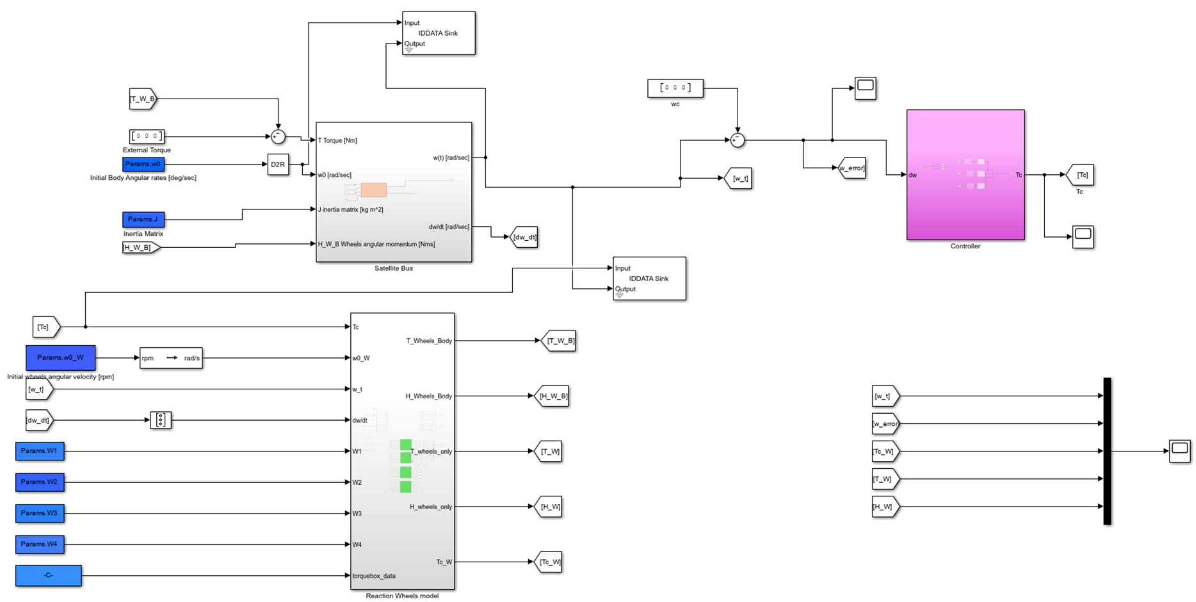


Figure 82-Detumbling Controller Model

The "Satellite Bus" block is responsible for estimating the angular velocity and moment of the satellite at any given time during the simulation, the "Controller" block is the detumbling controller, inside this block are 3 controllers, one for each axis (X, Y, Z). The "Reaction Wheels Model" block represents the reaction wheels on board of the satellite. It receives a certain commanded torque which is the controller's output. As a result, the reaction wheels start to spin in a certain speed, in order to produce the commanded torque.

7.5.2 Control Law

During the work on the controller, we made a logical error, we based the controller on the same control law that was used for cruise mode, but with different gains. We ran into a problem: the control law for the cruise loop required an initial quaternion, something that cannot necessarily be assumed or obtained, as the rotation inflicted on the satellite will render our STR unusable due to its limitations. The STR has a certain angular velocity threshold for which the images it captures become blurry and unreadable. As a result, it can't obtain the satellite's attitude.

To circumvent this issue, we had to look for a new control law based solely on the angular velocity measurement, assuming ideal measurements, for now.

Our first attempt was a PI controller, as was taught in the control theory course:

$$T_c(\omega_{error}) = K_P \cdot \omega_{error} + K_I \cdot \int \omega_{error}\, dt \tag{22}$$

Where:

$$\begin{cases} K_{P_i} = 2\omega_n^2 \cdot J_{ii} \\ K_{I_i} = 2\zeta\omega_n \cdot J_{ii} \end{cases} \tag{23}$$

7.5.2.1 Performance- PI Controller

We obtained the following results:

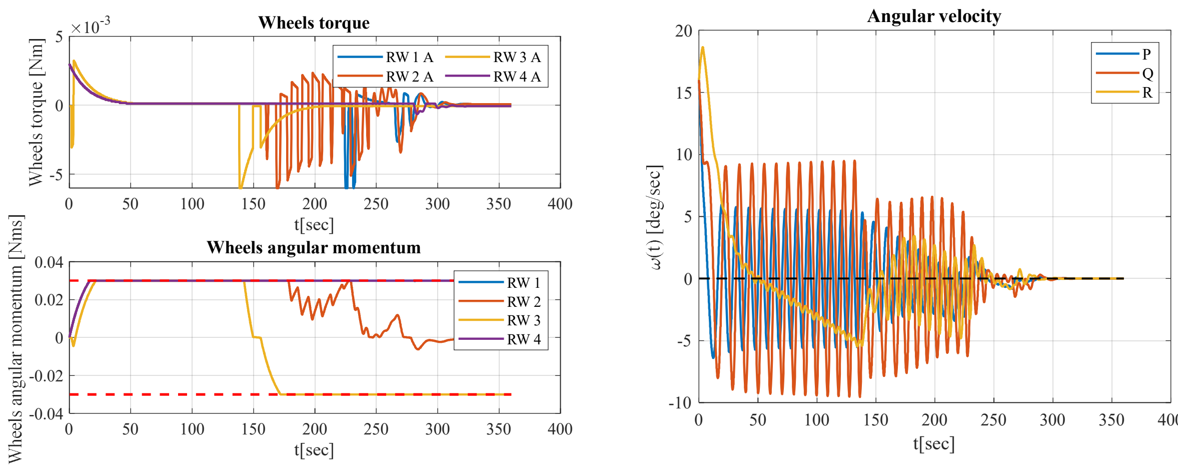


Figure 83-PI Controller Results

We can see that the controller almost complies with the system's engineering team's requirements: rotation with a magnitude of up to 30 degrees around any random axis, and settling time of under 5 minutes. The wheels are in saturation and the satellite still has some angular velocity after 5 minutes. We needed a faster and improved controller.

7.5.2.2 PID Controller

7.5.2.2.1 Method

The new control law is based on the paper "Probably the Best Simple PID Tuning Rules in the World" [2]. This paper includes a very in-depth explanation of the control law and the gains tuning method. We will now go through this process, step-by-step.

The new control law, as presented in the paper, is described as a cascade form PID controller [2]:

$$c(s) = K_c \cdot \frac{\tau_I s + 1}{\tau_I s} \cdot (\tau_D s + 1) \tag{24}$$

Where the controller gain, K_c is defined as:

$$\boldsymbol{K_c = \frac{1}{k}\cdot\frac{\tau_I}{\tau_c+\theta} = \frac{1}{k'}\frac{1}{\tau_c+\theta}, \qquad k' = \frac{\tau_I}{k}} \tag{25}$$

The derivative and integral times, τ_I and τ_D, are defined as:

$$\tau_I = \min\left\{\tau_1, \frac{4}{k'K_c}\right\} = \min\{\tau_1, 4(\tau_c+\theta)\}$$
$$\tau_D = \tau_2 \tag{26}$$

And θ denotes the system's time delay which we assumed as $3[sec]$, . This is bound to change, as there is no data regarding the delay of the future system.

We mainly assumed $\tau_I = \tau_1$, for more efficient calculations. The results can be optimized even more when calculating with the minimum operator shown (26), although the calculation will take a longer time, but with this assumption we still get good and accurate results.

7.5.2.2.2 Transfer Function Estimation

With these parameters established, we need to derive the system's transfer function. By using the System Identification application in MATLAB, we can derive the transfer function from the simulation alone and obtain $\tau_{1,2}$.

To obtain $\tau_{1,2}$ we start in Simulink by connecting the system identification block (highlighted in red), the name of the block is "IDDATA Sink" as shown below:

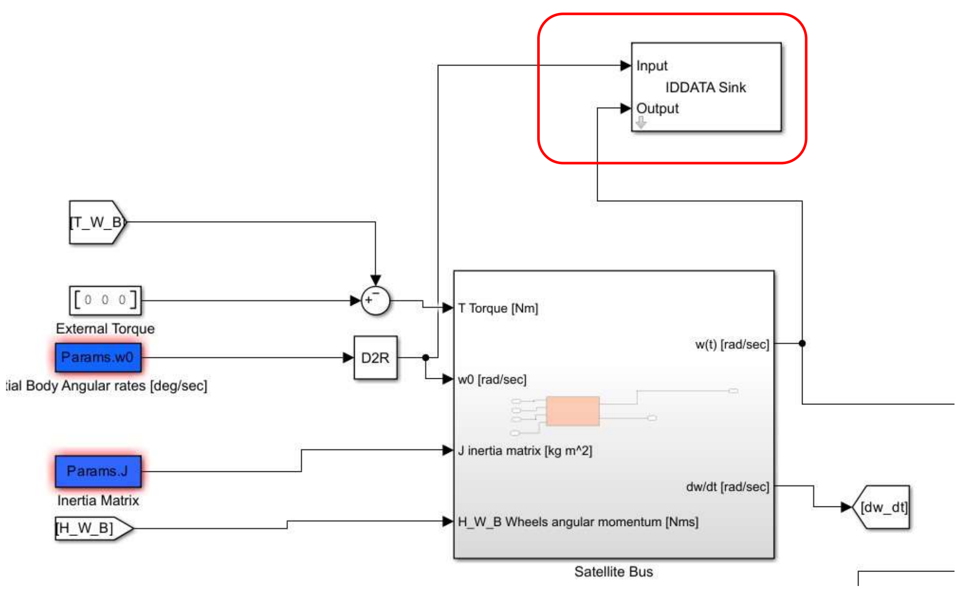


Figure 84-System Identification Block in Simulink

Here we connected the initial angular rates to the input, and the calculated angular rates as the output, which will provide us with the transfer function of the Satellite Bus block.

Running a clean simulation with either a proportional controller with a gain of 1 or just bypassing the controller seemed to provide the best results.

As the simulation ends, head to MATLAB. You can now see an IDDATA object when clicking on "Outsim" variable in the workspace window:

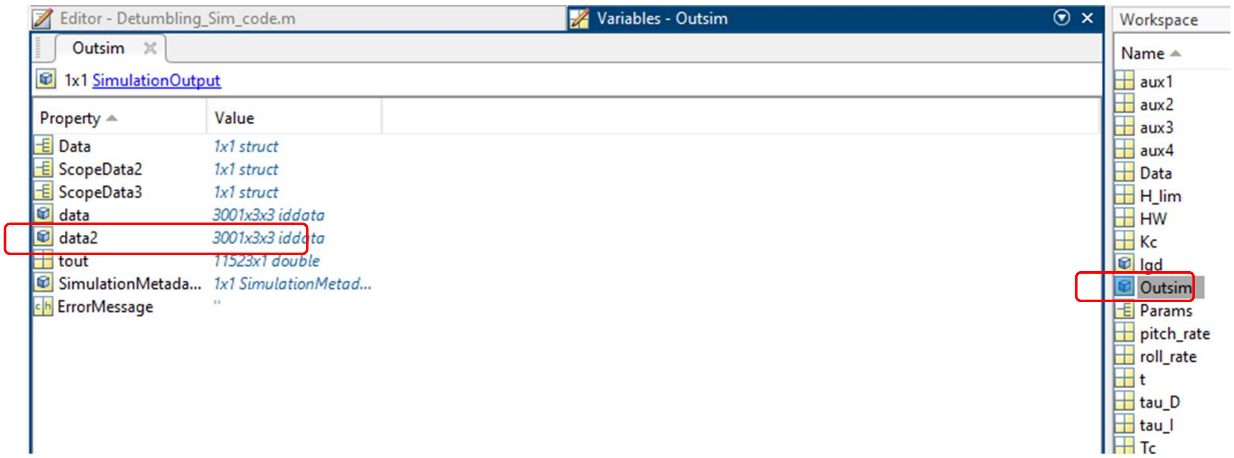


Figure 85-IDDATA File in MATLAB

Open the System Identification application:

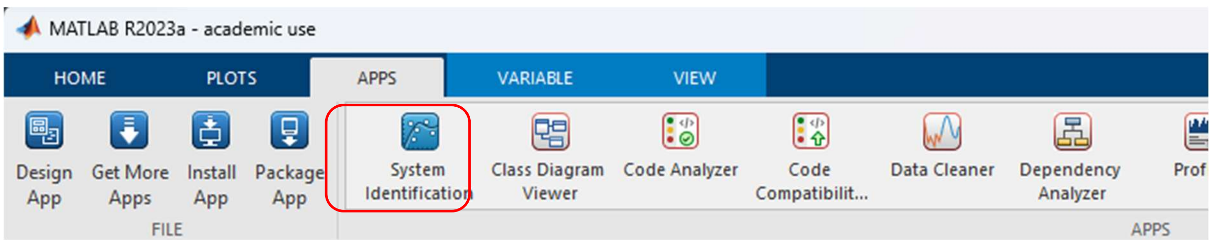


Figure 86-System Identification Application

After opening the application, a new window will open, and will appear as follows:

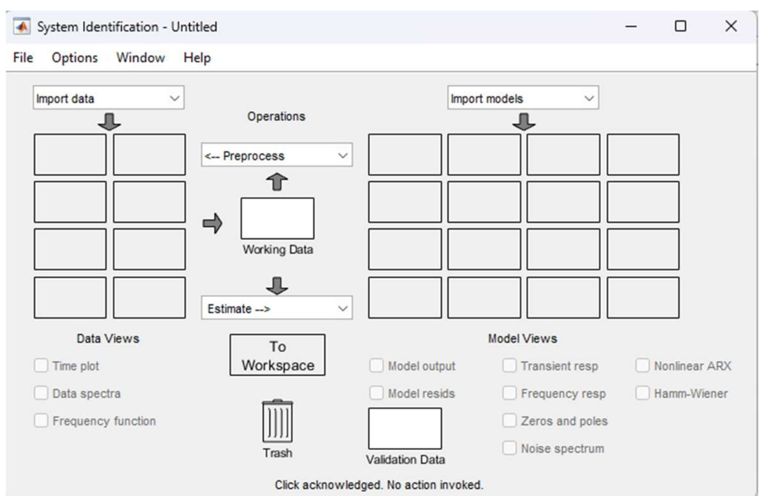


Figure 87-System Identification Application Window

We will now import said IDDATA object, click on "Import data" and choose "Data object…", a window will open. Under "Workspace Variable" enter the IDDATA object's name, in our case it is data2. Therefore, we would enter "Outsim.data2" and click "Import".

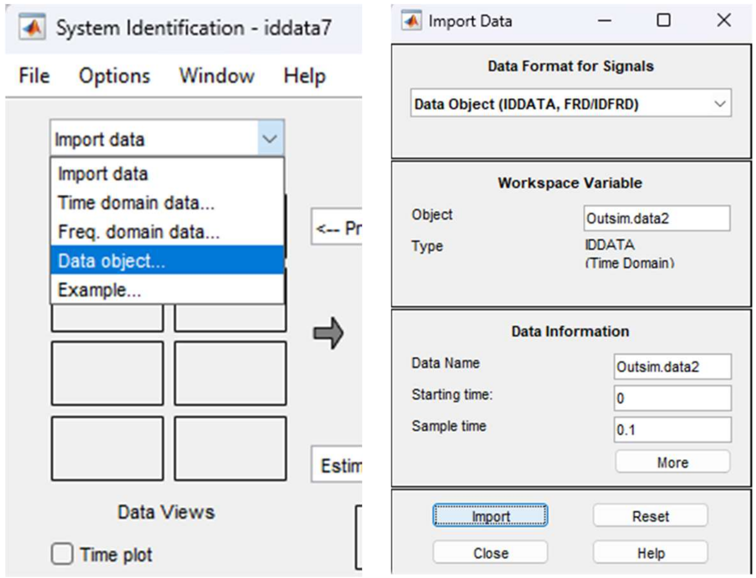


Figure 88-Importing Data

Now, we can see “Outsim.data2” on the top-left cell. Drag it to the blank “Working Data” cell:

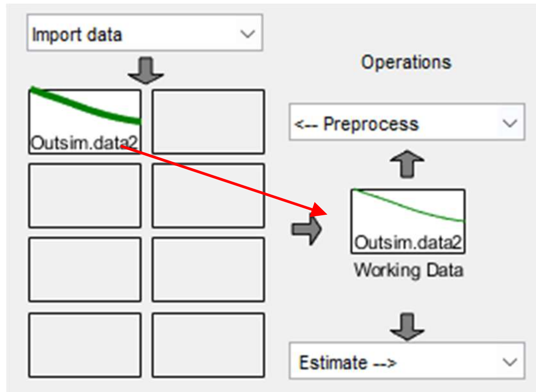


Figure 89-System Identification Process

From the “Estimate” drop-down menu, choose “Transfer Function Models…”, and a new window will open:

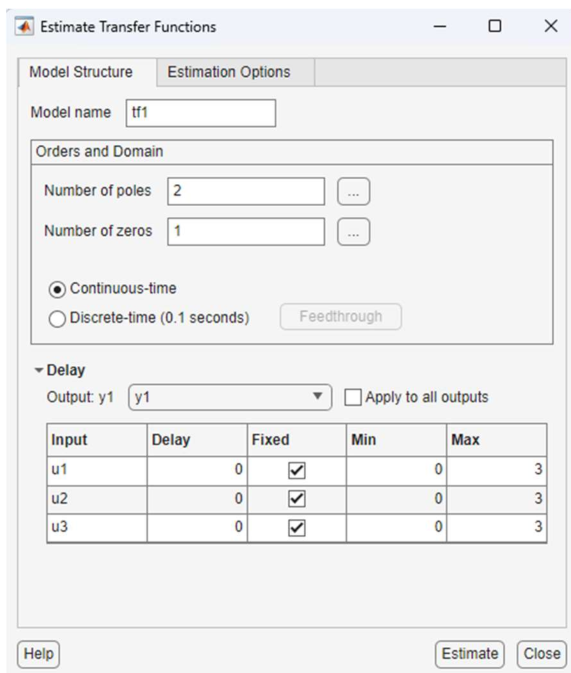


Figure 90-Transfer Function Estimation

For this example only, we set the number of poles to 2, and the number of zeros to 1, and most importantly, we set the delay to 3 on u1, u2 and u3 for all y1, y2 and y3. Click “Estimate” and another window will appear, wait for it to finish and close it.

Now, there is a transfer function object:

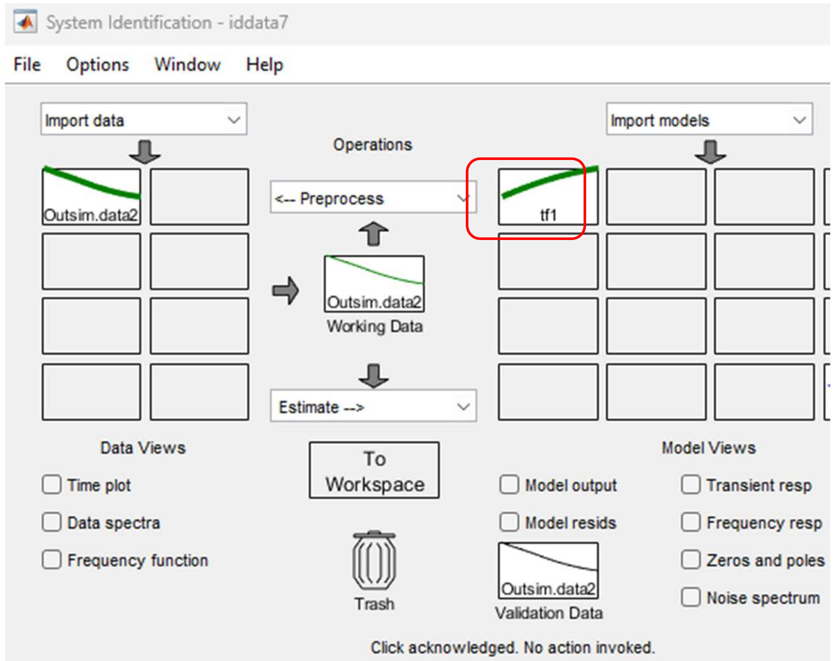


Figure 91-System Identification Window

Click the transfer function object twice and a new window will open, showing every transfer function for each input and output.

Once we have obtained all the given transfer functions, we can now calculate $\tau_{1,2}$. From the paper, we need to get to the following transfer function form:

$$\boldsymbol{g(s) = k\frac{1}{(\tau_1 s+1)(\tau_2 s+1)} \cdot e^{-\theta s} = k'\frac{1}{(s+1/\tau_1)(s+1/\tau_2)} \cdot e^{-\theta s}} \tag{27}$$

We can see that for finding the values of $\tau_{1,2}$ we need to estimate a transfer function of 2 poles and 0 zeros, transform the denominator to the following form, and then substitute into equations (24), (25) and (26). From there it is easy to obtain the required gains. In our case, these are the transfer functions we have obtained and the variables we calculated for $\theta = 3$:

$H(s)$	X	Y	Z
X	$\boldsymbol{\frac{0.9491}{1+0.4419s-0.01417s^2}}$	$\frac{0.002704}{1-1.823s+0.823s^2}$	$\frac{0.003685}{1-1.998s+s^2}$
Y	$\frac{0.003532}{1-1.767s+0.7763s^2}$	$\frac{0.9666}{1+0.9179s+0.2892s^2}$	$\frac{-0.0006766}{1-1.858s+0.8662s^2}$
Z	$\frac{0.006052}{1-1.999s+s^2}$	$\frac{0.006746}{1-1.974s+0.9808s^2}$	$\frac{1.226}{1+0.8567s+0.2577s^2}$

Figure 92-Transfer Functions

Then we can calculate:

$\theta = 3$	τ_1	τ_2
$\boldsymbol{H_{xx}}$	$\mathbf{2.1191}$	$\mathbf{-33.3038}$
H_{xy}	-1	-1.215
H_{xz}	$-$	$-$
H_{yx}	-1.0561	-1.2197
H_{yy}	$-$	$-$
H_{yz}	$-$	$-$
H_{zx}	$-$	$-$
H_{zy}	$-$	$-$
H_{zz}	$-$	$-$

Figure 93-Calculation of $\tau_{1,2}$

Cells with "-" represent complex values, we cannot use them to calculate the required values.

Therefore:

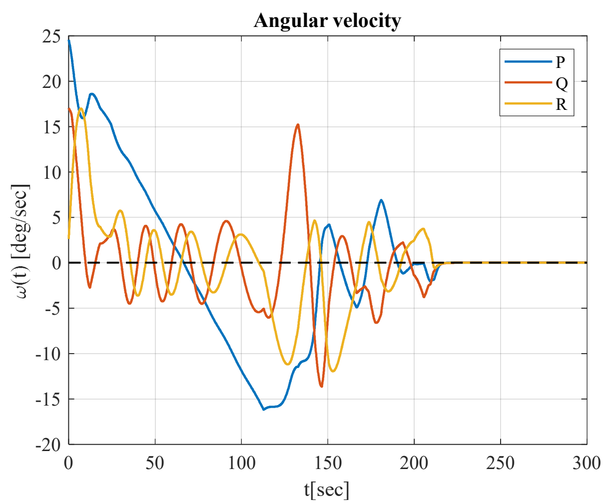
Kc	$\boldsymbol{\tau_I}$	$\boldsymbol{\tau_D}$
0.7202	2.1191	−33.3038

Figure 94-Calculated Variables

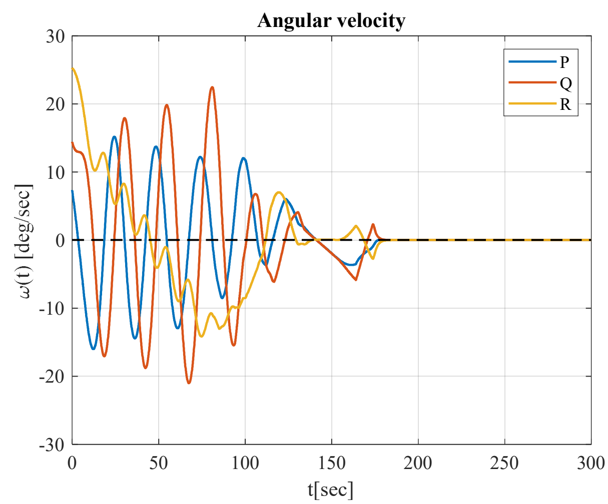
7.5.2.2.3 Performance Results

Below are the performance results, following the system engineering requirement to comply with a rotation of 30 degrees per second (magnitude) around a random axis:

[24.57,17.01,2.62]



[7.34,14.42,25.25]



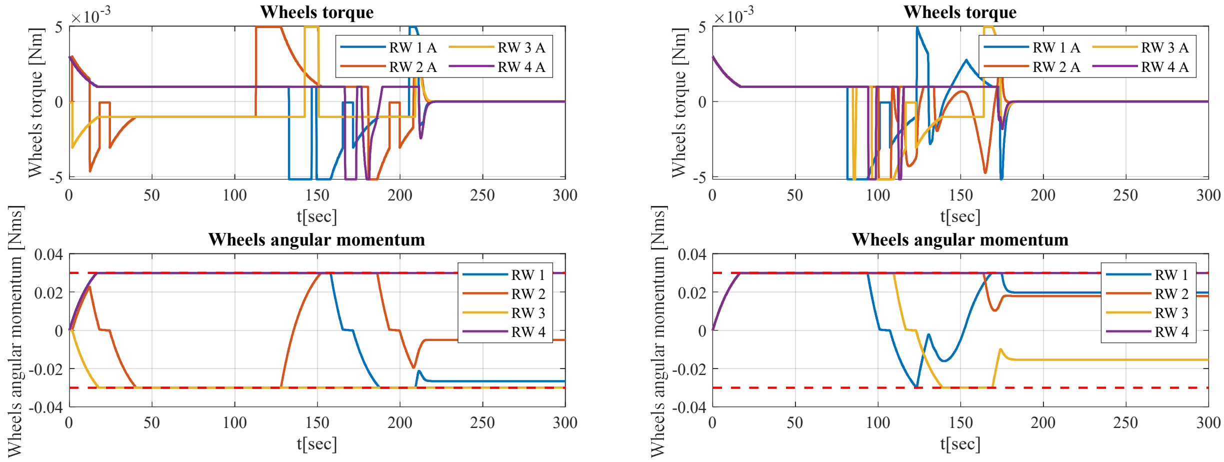


Figure 95-Cascade PID Controller Results

These results are based on a function that generates a random angular velocity vector, up to a magnitude of 30 degrees per second.

7.5.2.3 Detumbling Control Conclusion

The controller partially meets the requirements set by the system engineering team. While the controller does stop the rotation, there is a problem which is rooted in the limitations dictated by the reaction wheels, notice the "Wheels Angular Momentum" plots in Figure 95-Cascade **PID Controller Results**. The red broken line represents the saturation of the wheels, and any higher than the red value, which is $\pm 0.03[N \cdot m \cdot s]$, will cause damage to the wheels and compromise the mission. It is apparent that the only solution for the initial detumbling phase is engine-based detumbling control. The controller shown can be used to control rotations that exceed the limit of the STR, which is $3\left[\frac{deg}{sec}\right]$, during the satellite's cruise phase.

7.5.3 Engine Based Detumbling

Upon conferring with an aerospace industry expert specializing in control systems working in IAI, it has become evident that a strategic integration of thrusters is not only highly advisable but also imperative. This is particularly significant during the transition into cruise mode, once the angular velocity decreases and the satellite has the correct orbit elements. The purpose here is to prevent the accumulation of momentum within the reaction wheels, thus helping unimpeded satellite reorientation in the desired direction. Put differently, any momentum amassed in the reaction wheels during the detumbling phase must subsequently be dissipated by using the adequate thrusters.

7.5.3.1 The Engine

The previous year's team chose one of "Dawn Aerospace" engine following a thorough market survey. This engine offers two distinct operational modes: cold gas and bi-propellant. Detailed information from the manufacturer is available for future work, outlining its specifications. The engine comprises four thrusters positioned at the vertices, each accompanied by an oxygen tank and a fuel tank. Each of the thrusters is able to be directed according to a predetermined tilt angle, more about this later.

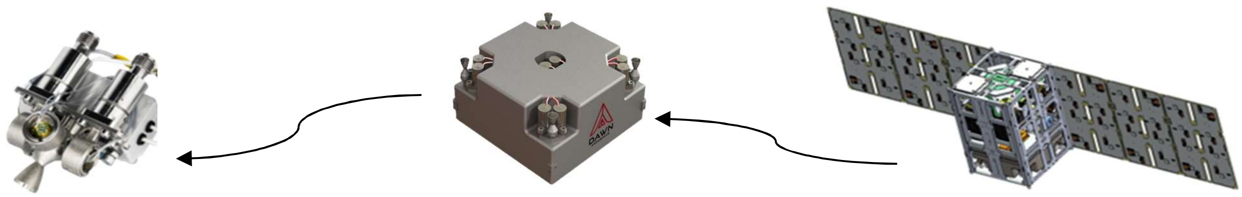


Figure 96-macro To Micro - Satellite's Engine

7.5.3.2 Controller

After examination, it was chosen to use a proportional controller for the angular speed control loop with the help of a motor. Our main emphasis was on constructing the simulation and laying out the logic upon which it would be built. The ongoing task involves selecting the optimal controller value for each mode within this logic.

7.5.3.3 Simulation

We began working on a simulation model that utilized the engine's thrusters to control the satellite's angular velocity around its body axes. We opted for this approach initially to keep things simple, without combining other constrains yet.

To initiate the development of the simulation logic, we investigated the prescribed manner in which the thrusters should function. This manner is supposed to generate a pure moment along each of the satellite's axes. The analysis considers both the intended satellite geometry and some of the engine properties, for the sake of simplicity. The following equation, using MATLAB, helped us receive the adequate gains that represent the strength in which each thruster will need to operate, in order to receive pure moment around one body-axis-

$$M = a \cdot \overline{m}_{thruster_1} + b \cdot \overline{m}_{thruster_2} + c \cdot \overline{m}_{thruster_3} + d \cdot \overline{m}_{thruster_4} \tag{28}$$

For example, in order to receive a pure positive moment around the x-axis, we had to demand $M = [1,0,0]$. Afterwards, we used the 'solve' tool in MATLAB, and got the suitable a, b, c, d . It's important to note that the coefficients were constrained to be positive. Furthermore, these coefficients will require normalization based on the engine's mode (Bi-prop/Cold gas, details are in manufacture's data page), which dictates the applicable thrust that can be generated. These constraints have not been incorporated into the simulation yet, and are intended for next year's endeavors.

In addition, during this deliberation, it was determined to intentionally tilt differently each two thrusters. Specifically, one pair of opposing thrusters would undergo a 7° tilt, while the other pair would tilt in 10°. This angle of tilt is demonstrated below in the figure as γ.

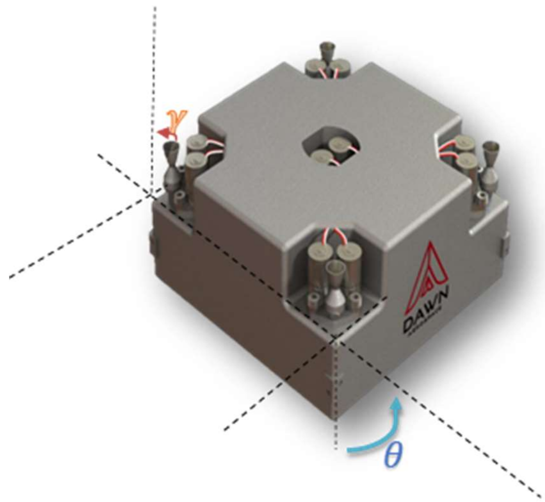


Figure 97- Engine Dawn Aerospace

7.5.3.3.1 Simulink- Engine's Model

The primary focus in setting up the new Simulink model, which incorporates the engine's thrusters as actuators, revolved around the Engine block depicted below:

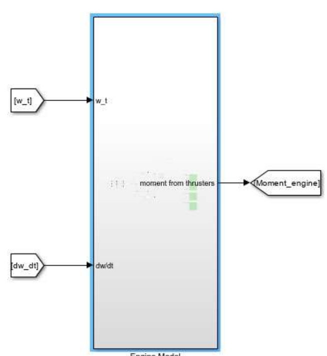


Figure 98- Engine's Model Block

The angular velocity is continuously monitored. The initial step involves prioritizing the body axis with the most substantial impact on the satellite. Meaning, that the body axis with the highest angular velocity is the current target.

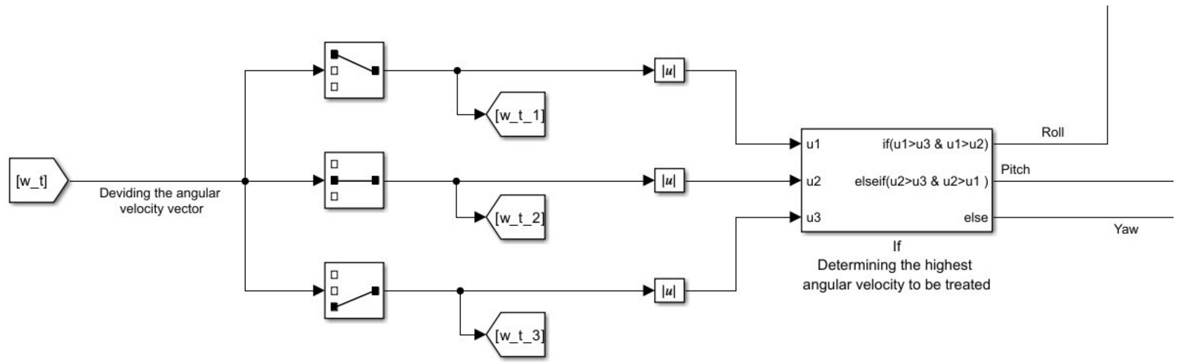


Figure 99- Prioritization Of Body Axes

Once we've determined whether the angular velocity is in the positive or negative direction, we proceed to engage the corresponding thrusters. This is done to generate a pure moment in the opposite direction as needed. The specific thrust produced by each thruster is determined based on the analysis conducted earlier, as outlined in the preceding section.

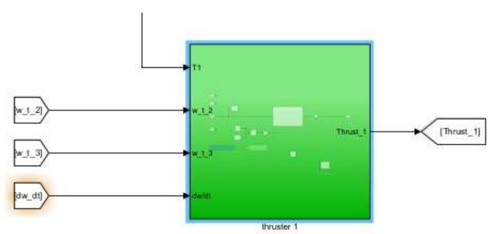


Figure 100- Thruster Block

Lastly, the total moments acting on the satellite are computed. These calculations rely on the orientation of each thruster.

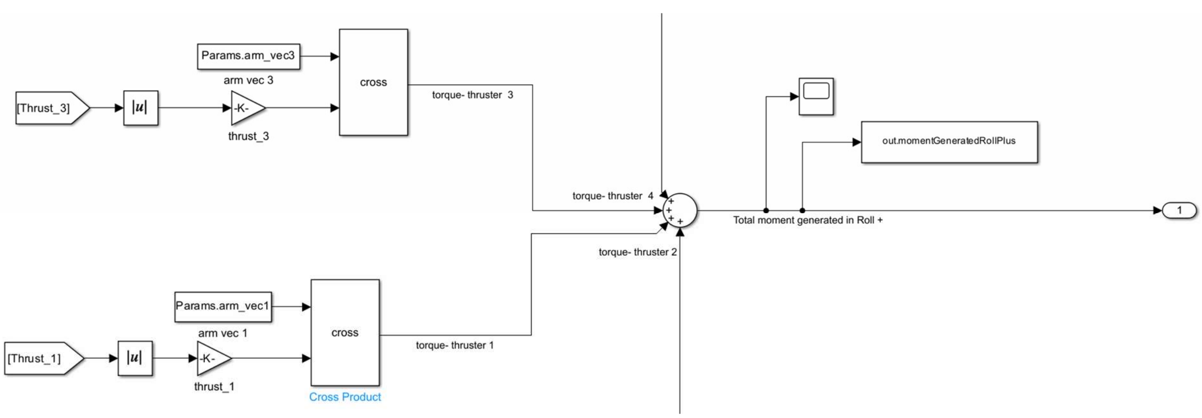


Figure 101-Calculation Of Total Moments

7.5.3.3.2 Performance

Here is an example of the control loop performance (detumbling based engine):

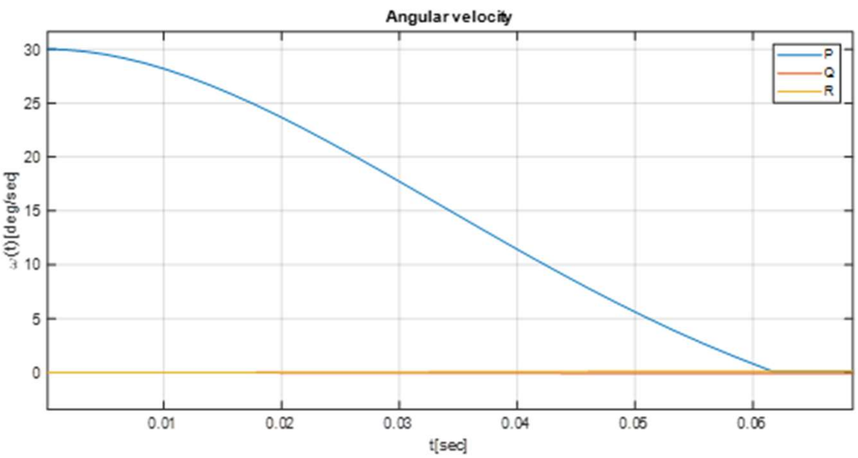


Figure 102-Satellite Angular Velocity

The initial angular velocity in this case was $\omega_0 = [30,0,0]\ \frac{deg}{sec}$, following the requirement of the system engineering team.

Here we can observe the engine's torque around each one of the body axes:

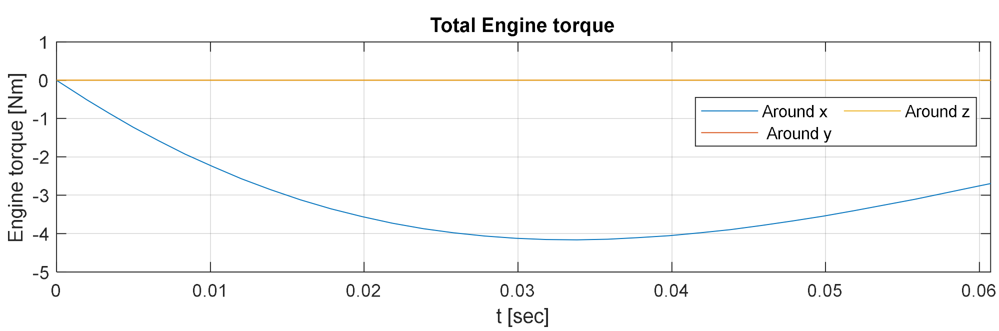


Figure 103- Engine's Torque

It can be clearly seen that the detumbling of the satellite is performed in a significantly shorter time compared to the use of the reaction wheels. Naturally, there is also no accumulating momentum at all that we need to take care for after.

As an additional remark, this simulation generates control inputs whenever the angular velocity is being monitored. Although this might not be wise considering the computational effort required. Consequently, the subsequent step will be discretizing the control loop, defining Δt as the time interval for a specific set of control inputs. This choice is made due to the perceived greater influence of the thrusters compared to the reaction wheels.

7.5.4 Hybrid Detumbling Control Architecture

We have thought of implementing a hybrid control loop, that integrates use of thrusters together with reaction wheels. The new control architecture consists 'Switch' block that chooses actuator according to angular velocity magnitude at the moment. In this way, we will enable control over the balance between fuel waste and momentum storage in the reaction wheels. Furthermore, we have planned inserting a limiter monitoring the angular velocity. Therefore, we could prevent unnecessary control inputs, and reduce calculations, once the angular velocity is small enough.

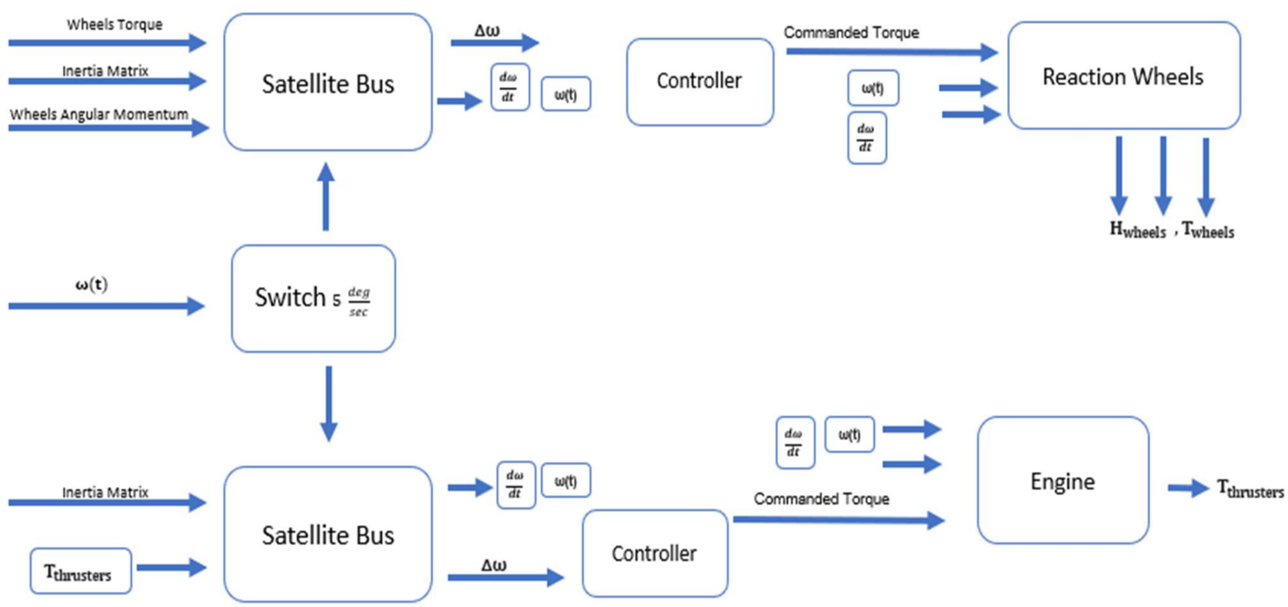


Figure 104 - Concept Control Arcitecture

Relying on consultation with specialist from AIA, the chosen value for the 'Switch' block is $5\left[\frac{deg}{sec}\right]$. This value is subject to change based on fuel consumption considerations, but it provides a reasonable initial estimate.

7.6 Future Work

There are a few tasks which should be considered by next year's team:

Develop and implement a sun search algorithm – A sun search algorithm provides the most efficient way for the satellite to locate the sun. This task should be of highest priority

Implement the engine's limitations, as provided in the data sheet, in the engine-based control loop simulation

Determine the optimal thruster tilt angles

Analyze fuel consumption for the mission and implementation

After the payload has been chosen, fine tune the finished control loop to fit the new satellite properties

Implement the combined wheel-engine control loop

Develop an altitude control loop architecture and simulation

The estimated task times can be found on the work times Excel table on the drive.

7.7 References

[1] M. Malka, T. Shneor, M. Alon, Y. Granov, A. Binyamin, O. Zakar, M. Blum, T. Itzhaki, O. Amber, Y. Arad, B. Muchnik, Y. Weinberg, G. Aharoni, T. Alperin, D. Unter. (2022). Project Jericco – Progress Report.

[2] Skogestad, S. (2001), *Probably the best simple PID tuning rules in the world.*

8 Avionics Team

8.1 Nomenclature

The following are the convention symbols and their meanings that will be used in this chapter.

EPS – Electric Power System

GCS – Ground Control Station

C&DH – Command and Data Handling

OBC – On-Board Computer

LUMIO – Lunar Meteoroid Impact Observer

PDU – Power Distribution Unit

ACU – Active Current Unit

STR – Star Tracker

IMU – Inertial Measurement Unit

RW – Reaction Wheels

LOS – Line Of Sight

8.2 Abstract

Avionic is the development and production of electronic instruments for use in aviation and astronautics. The purpose of this part is to introduce the Avionic systems that have been designed for the JERICCO satellite. The Power system, which acts as the primary energy source for the satellite, is responsible for generating, supplying, and storing power throughout the satellite's lifespan. Solar panels are used for power generation, while batteries and an Electric Power System (EPS) manage power storage and routing, respectively. These systems consist of various subsystems, each with its unique function.

The communication system is responsible for connecting the satellite to the ground control station (GCS) and fulfilling all mission objectives, requirements, and radio regulations. Additionally, the avionics system oversees the command and data handling (C&DH), which includes the On-Board Computer (OBC) and a data storage unit.

In this section of the report, we will discuss the process of constructing the block diagram and characterizing the interface lines between different systems. Additionally, present a comprehensive overview of the solar panels that were chosen, including the considerations that informed the selection. Lastly, in the avionic future work, we will map out the required components.

8.3 Introduction

In this article, we will explore the avionic systems designed for the JERICCO satellite, which are essential components that enable its proper functioning in orbit. The avionic systems include the Power System, Communication System, and Command and Data Handling (C&DH) system.

The Block Diagram is a crucial aspect of the avionics system design. It allows for the visual representation of the different subsystems and their interface lines. In this section, we will discuss the process of constructing the block diagram and characterizing the interface lines between different systems. As done in previous missions that have similarity to ours [2] (Power System Architecture Diagram (from LUMIO- Lunar Meteoroid Impact Observer)), it was important to ensure that all the components were able to interface with each other in a way that promised good performance and could handle all the different scenarios, as analyzed previously.

The Power System serves as the primary energy source for the satellite. It is responsible for generating, supplying, and storing power throughout the satellite's lifespan. This system utilizes solar panels for power generation, while batteries and an **Electric Power System (EPS)** manage power storage and routing, respectively.

The Solar Panels capture the energy from the sun and convert it into electrical energy, which is then used to power the satellite's subsystems. The size and type of solar panels used depend on factors such as the amount of power required and the available space on the satellite. During the year, issues with the selected panels have been identified, so we have been in touch with MMA design [1] to obtain a model more suitable for our mission. More details will be provided later.

Batteries, on the other hand, provide backup power for the satellite when it is not receiving energy from the sun, such as during eclipse periods. They also help regulate the power output from the solar panels, ensuring that the satellite receives a consistent flow of power. The type and number of batteries used depend on factors such as the mission duration and the required power level. Both solar panels and batteries are essential components of the JERICCO satellite's Power

System, working in harmony to ensure that the satellite's subsystems have the energy they need to function correctly.

The Communication System is responsible for connecting the JERICCO satellite to **the ground control station (GCS)** and ensuring that all mission objectives, requirements, and radio regulations are met. In addition, the avionic system oversees the Command and Data Handling (C&DH) subsystem, which includes the **On-Board Computer (OBC)** and a **data storage unit**. For our mission, we have chosen the NanoPower P60 EPS and NanoMind A3200 OBC by GOMSpace. More information about the reasons for this choice can be found in last year's report.

8.4 The Block Diagram

One of the key components for the mission success is the Avionics Block Diagram, a vital roadmap that outlines the satellite's intricate electronic systems and functionalities. Let's delve into the details of this cutting-edge avionics' architecture, unveiling the remarkable technologies poised to enable JERICCO's journey beyond our planet:

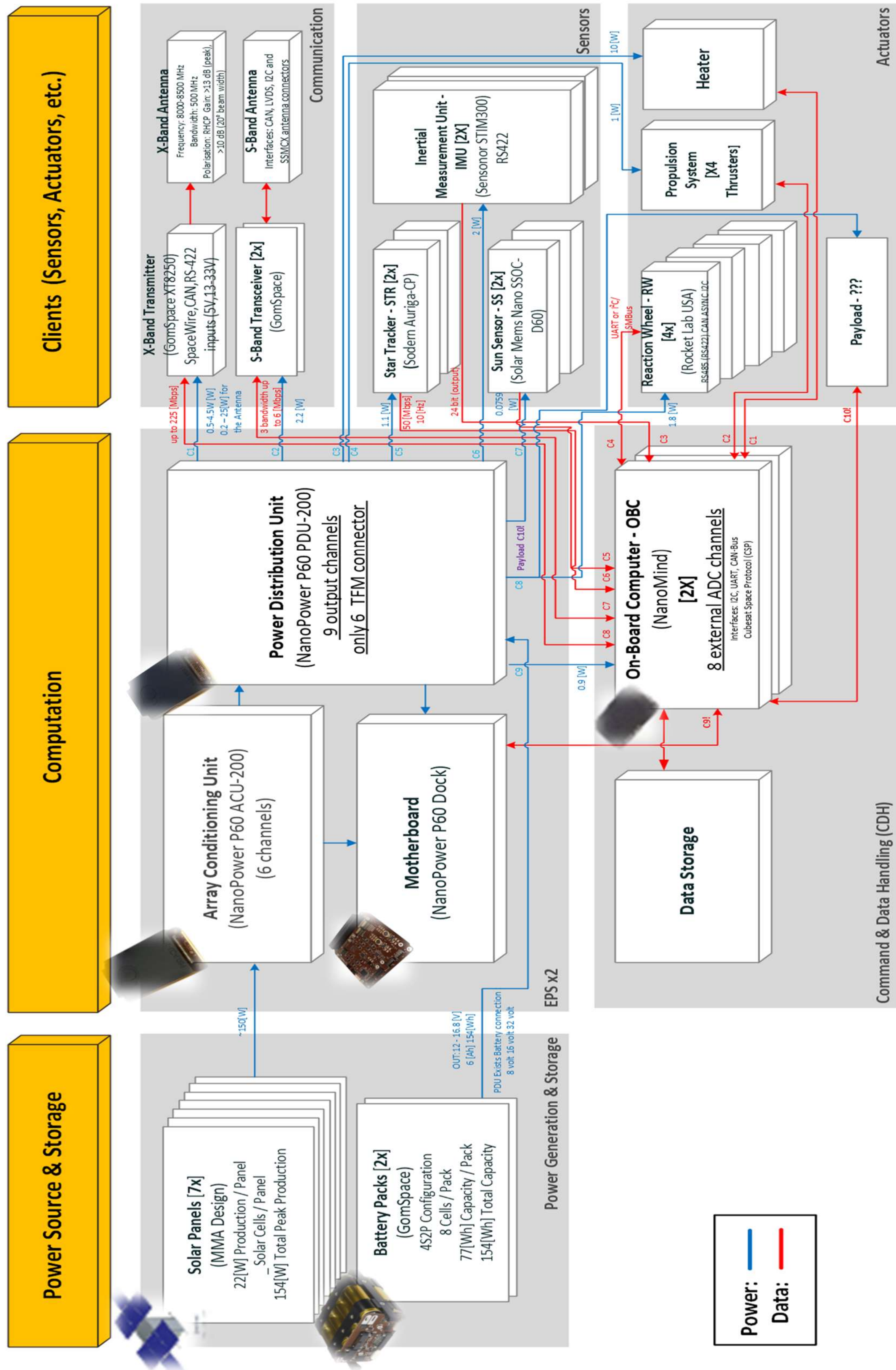


Figure 105: Block diagram

Power and Storage:

a. Solar Panels (MMA) - These solar panels are responsible for generating electrical power from sunlight.

b. Battery (Gom Space) - The battery stores excess power generated by the solar panels to provide power during eclipses or when the satellite is in the shadow of the Moon.

Computation:

a. Electrical Power System (EPS) - This includes the ACU (Active Current Unit) and PDU (Power Distribution Unit) motherboard based on the NanoMind technology. The EPS manages and distributes electrical power to all the subsystems.

b. On-Board Computer (OBC) - There are two NanoMind-based OBCs responsible for general computation, data processing, and overall satellite control.

c. Data Storage - This component is responsible for storing mission data and other relevant information.

Client Communication:

a. X-band and S-band Antennas (Gom Space) - These antennas facilitate communication with Earth-based ground stations. The X-band provides higher data rates, while the S-band offers reliable communication for various mission phases.

Sensor:

a. Star Trackers (STR) - There are two Sodem Aulga star trackers used for precise attitude determination by observing the stars.

b. Inertial Measurement Unit (IMU) - There are two Sensor STIM 300 IMUs that provide information about the satellite's orientation and motion in space.

c. Sun Sensors - There are two Solar MEMS nano sun sensors that detect the direction of sunlight incident on the satellite's surface.

Actuators:

a. Reaction Wheels (RW) - There are four Rocket Lab reaction wheels used for attitude control and stabilization of the satellite.

b. Heater - The heater helps in maintaining the temperature of critical components within their operational range in the harsh lunar environment.

c. Propulsion System - Responsible for orbital maneuvers and maintaining the satellite's position in the lunar orbit.

Unknown Payload:

The block diagram mentions an unknown payload, which could be any additional equipment or instruments onboard.

Overall, the avionics block diagram provides insights into the technological capabilities and design considerations of our CubeSat. It showcases a sophisticated system with advanced computation, communication, and control capabilities, enabling the satellite to carry out its scientific mission effectively in the challenging environment of lunar orbit.

In constructing the block diagram, one crucial aspect we emphasized was establishing the interfaces between the different components. It is noteworthy that when it comes to interfaces, there exist several options for each component, providing flexibility and catering to diverse requirements.

Among the various interfaces utilized in the design, we have:

On-Board Computer	Reaction Wheels	X-Band Communication	S-Band Communication	Star Tracker	IMU	Sun Sensor	EPS
I2C UART CAN BUS CSP	RS422 CAN ASYNC I2C	SPACEWIRE CAN RS422	LVDS CAN I2C	SPACEWIRE	RS422	UART I2C SPI	CAN BUS I2C KISS

Table 24: CubeSat's Components

8.5 The Power System

The power system in our CubeSat plays a critical role in ensuring the satellite's continuous operation and successful execution of its scientific mission. The power system is designed to harness solar energy and efficiently distribute it to the various subsystems while also storing excess energy for use during eclipses or periods of reduced solar exposure.

Coordination between Solar Panels and Batteries:

The coordination between the solar panels and batteries is crucial to maintain the satellite's power balance and extend its operational capabilities. When the solar panels receive sunlight, they generate electrical power, which is then supplied to the EPS for distribution to the satellite's components and for charging the batteries. During periods when the satellite enters lunar shadows or experiences eclipses, the batteries take over and supply power to the CubeSat to ensure continuous operation.

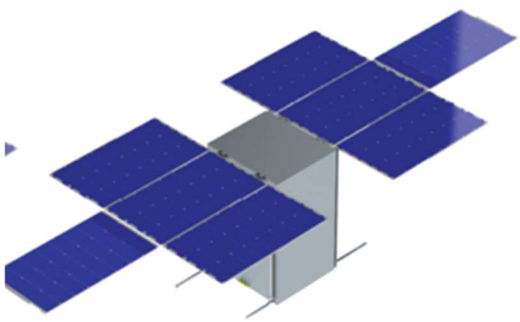
8.5.1 Solar panels

Throughout the year, we faced the challenge of choosing the solar panel configuration for the satellite. Following in-depth discussions with the MMA company, we have finalized the selected configuration, as depicted below [1]:

eHaWK Solar Arrays during final inspection at MMA.

Figure 106: Solar Panel

The chosen configuration resembles a T-shape, with the panels closing in on the satellite's ends. In terms of power capabilities, the configuration offers the following power:



Quad Panel
6U/12U: 178W
8U/16U: 207W

Figure 107: Solar Panel Power

After receiving feedback from the MMA company, we conducted additional research on the implications of selecting such panels for our mission. The results indicated that this configuration is not only effective but also reliable for various tasks. [The whole information we got from MMA is located at our JERRICO drive under Avionic team]

The T-shaped configuration of the solar panels for the satellite offers several potential advantages, including:

Optimal Sunlight Exposure: The T shape allows for a larger surface area of solar panels to be exposed to sunlight, maximizing power generation during the satellite's orbit around the Moon.

Increased Power Output: With more solar panels arranged in the T configuration, the satellite can generate higher power levels, providing ample energy for its scientific mission and subsystems.

Redundancy and Robustness: The T-shaped arrangement provides redundancy in case of partial shading or panel failure, ensuring continuous power generation and operational reliability.

Ease of Deployment: The configuration's design simplifies the deployment process, allowing for the efficient unfolding of the panels once the satellite reaches its intended lunar orbit.

8.6 Communication

To enable communication between the CubeSat and the ground station for necessary adjustments, two options are available: direct and indirect communication. In the direct communication approach, the CubeSat's antenna serves as the sole means for sending and receiving signals. Alternatively, the indirect communication approach involves utilizing a relay satellite. This method enhances the communication window for each orbit but depends on the specific relay satellite chosen. The third approach combines both options to leverage their individual advantages.

These approaches will determine the antenna length, the achievable data rate for sending and receiving information, and the duration of the transmission. Therefore, the mission's payload will be constrained by those approaches and their following decisions.

8.6.1 Direct

Direct communication from a CubeSat to the ground station involves the transmission of signals directly between the satellite's onboard antenna and the receiving antenna at the ground station. This approach relies on a clear line of sight (LOS) between the CubeSat and the ground station, allowing for a direct, unobstructed communication link.

Advantages:

Simplicity: Direct communication offers a straightforward and simple communication setup. There are no intermediate satellites or additional complexities, which can streamline the system design and reduce potential points of failure.

Control: Having a direct link allows for more direct control over the communication parameters and performance.

Disadvantages:

Limited Coverage: Direct communication is dependent on the CubeSat's position relative to the ground station. As the satellite orbits the moon, the communication window is constrained to when the CubeSat is in view of the ground station. This can limit the frequency and duration of communication opportunities.

Ground Station Network: To maintain continuous coverage, a network of ground stations may be required. This can add to the overall project cost and complexity, especially for global coverage.

Limited Data Rate at Longer Distances: As the CubeSat moves away from the ground station, the signal strength weakens, which can lead to reduced data rates. The CubeSat's payload cannot exceed this limit in effort to maintain a successful mission.

In summary, direct communication offers a straightforward and efficient means of establishing a link between a satellite, such as our CubeSat orbiting the moon, and a ground station. This approach provides direct control over the communication parameters and can serve as a reliable means of data transmission between lunar missions and Earth-based ground stations. However, it is essential to consider the limited coverage during the satellite's orbit around the moon and potential signal attenuation at larger distances during the system design and mission planning.

8.6.2 Indirect

Indirect communication, involving the use of a relay satellite, offers an alternative approach for our CubeSat orbiting the moon to establish communication with Earth-based ground stations. By employing an intermediate relay satellite, this method can extend the communication window and enable more frequent and stable connections. It provides the advantage of broader coverage during the satellite's orbit around the moon, improving the overall reliability of data transmission. However, careful consideration should be given to the selection of the appropriate relay satellite and the potential signal attenuation that can occur during data transfer over greater distances. Despite these challenges, the relay satellite option remains a viable and valuable solution for enhancing communication capabilities during lunar missions.

For this approach, a couple of options arise as an optional relay satellite, each having different advantages and disadvantages.

The following requirements will be considered while searching for an ideal relay satellite:

- Power Supply and Consumption
- Coverage and Link Budget
- Frequency Bands
- Data Rate and Bandwidth
- Antenna System
- Reliability and Redundancy
- Compatibility and Integration
- LOS between the satellites
- Cost and Availability

LunaNet (NASA) – Networking

Typically, when missions launch into space, their communication down to Earth is reliant on pre-scheduled links with either a space relay or a ground-based antenna. With multiple missions journeying to the Moon, the reliance on pre-scheduled links could limit communications opportunities and efficiencies. LunaNet offers a network approach similar to the internet on Earth, where users maintain connections with the larger network and do not need to schedule data transference in advance. [2]

LunaNet offers a network of satellites that connect with each other to facilitate signal transfer between the ground station, the CubeSat orbiting the moon, and back. Thanks to the success of the NASA network, we can theoretically assume the possibility of the relay satellite in our project.

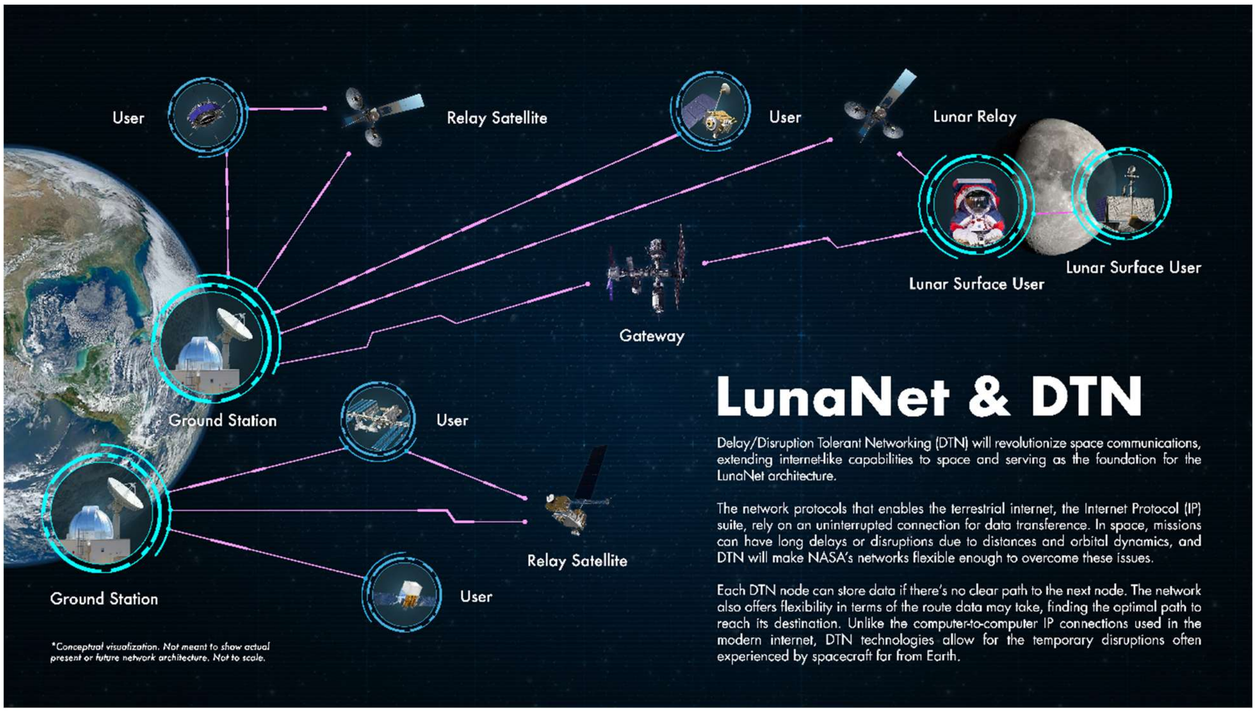


Figure 108: NASA Network

Lockheed Martin

One of the significant contributions of Lockheed Martin in the space domain is its development of relay satellites. These relay satellites play a critical role in facilitating communication between spacecraft, ground stations, and other satellites. By acting as intermediaries, relay satellites help extend the reach and efficiency of communication systems, particularly in remote or challenging environments. [3]

As a leader in the aerospace industry, Lockheed Martin's contributions to relay satellite technology have significantly advanced space communication capabilities, laying the groundwork for future missions and discoveries beyond our planet.

In compare with NASA, Lockheed Martin is a privet company and therefore will be more open to co-work with outside projects like our Jericco's CubeSat.

OHB

A small European geostationary platform (SmallGEO) for communications applications is being developed under OHB's lead management. [4]

With this company our CubeSat can communicate through a GEO communication Satellite. This option can increase the window we have between the ground station and the CubeSat but will have all the disadvantages we discussed before such as relying on another satellite transmitting the information.

The first satellite utilizing the SmallGEO platform was placed in orbit in 2017. Further SmallGEO satellites are being developed and/or manufactured.

Beresheet 2

Beresheet2 is Israel's second spacecraft whose development will shortly commence at SpaceIL. [5] The Beresheet program includes 2 landers and 1 Orbiter. The Jericco CubeSat can be masked as one of the landers while the orbiter does not connect to its landers and use it as the relay satellite.

Due to budget limitations, the Beresheet mission might not launch, yet we remain hopeful and stay updated on SpaceIL's news.

With this option of relay satellite our constrains on the payload are defined in two areas, LOS and the data download rate.

- Data download rate is between 5-60 [kb/sec]
- Direct contact with the Orbiter is at least 123.1 [sec] and at the most 1083.8 [sec] per orbit.

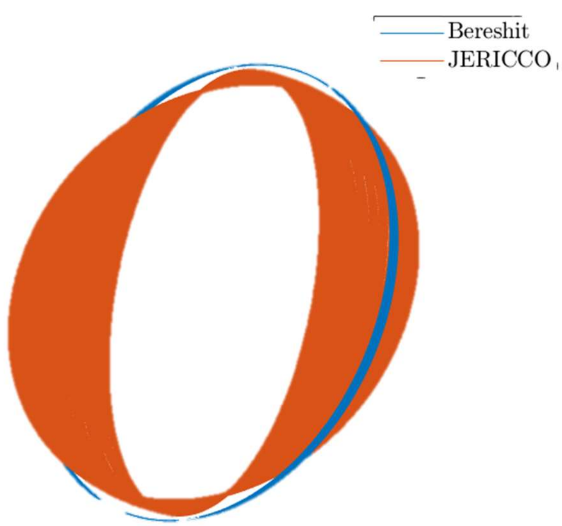


Figure 109 Beresheet and Jericco Orbits

To establish the LOS window between these two satellites, we employed the GMAT program, using the orbital elements of each satellite. More information about the GMAT program can be found in the 'Mission and Orbit' section.

Summarize:

In the search for an optimal relay satellite to enable indirect communication for our CubeSat orbiting the moon, the options above were carefully evaluated, each with its unique advantages and disadvantages. By employing an intermediate relay satellite, this approach extends the communication window, enhancing the overall reliability of data transmission during lunar missions. However, to ensure the most suitable choice, the requirements that were discussed previously took into consideration. In the following table we have collected all the current information we have.

	Number of satellites	**Launch date**	**Contact**
LunaNet (NASA)	Network	2022	Designated
Lockheed Martin	Network	2025	TBD
OHB - smallGEO	1 in GEO	Since 2017	TBD
Beresheet 2	1 Orbiter	2025	Established

Table 25: RELAY SATELLITE SUMMARIZE

8.7 Risk Management

The main risks are presented:

Num.	Description	Probability of realization (1-5)	impact of implementation (1-5)	Risk Level
1	over-expected allocation to the scheduled supply, leading to deviations in the allocation of resources among all components.	2	3	**6**
2	low information transfer rate under conditions of direct/indirect communication, potentially resulting in the overwriting of existing information with new data.	4	5	**20**
3	associated with a communication method that incorporates a relay satellite, where dependence on its activity may lead to changes in satellite control.	3	4	**12**
4	incompatibility between the current solar panel configuration and the satellite's structure.	3	5	**15**
5	insufficient battery charging during prolonged periods of solar eclipse.	2	5	**10**

Table 26: Risk Management

8.8 Future Work

For our next Avionics team future work, we suggest the following:

- **Power**
  - Monitoring power balance for each orbit. (All year long due to adjustments)
  - Determining the final configuration of the solar panels. (3 weeks)
  - To continue bringing improvements to the code built for the Power simulation in collaboration with the simulation team. (All year long due to adjustments)
- **Communication**
  - Establishing communication via direct contact or relay satellite with the ground station. (5 weeks)
  - Characterizing the satellite antenna in a manner that aligns with the chosen communication method. (3 weeks)
  - Ensure that the components in the selected communication configuration know how to operate accordingly and, if necessary, use adaptors. (3 weeks)
  - After the team selects the payload, evaluate how much memory is required to perform the task and ensure that our components meet the requirements.(5 weeks)

8.9 References

[1] MMA Solar Panels

[2] LunaNet: Empowering Artemis with Communications and Navigation Interoperability

[3] Lockheed Martin

[4] OHB

[5] Beresheet 2

[6] MMA Appendix

9 Simulation

9.1 Abstract

Simulation is the means for testing and verifying the design choices of the different teams working on the product.

In every engineering project there is a need to verify the design decisions by reviewing the effect of those decisions. As such, it is useful to create a simulation which encompasses the entirety of the product and its environment and simulates them with high precision for accurate results.

This is true especially in the development of a satellite for deep space conditions, since mimicking those conditions here on earth is expensive and, in many cases, impossible.

In a typical project, there are different teams, each with their own subject of development (such as control of the satellite, thermal balancing, etc). The system simulation is a tool which integrates those different disciplines into one simulation, with the purpose of showing the effects of different sub-systems on one another. The simulation is a tool used mainly by the system engineers, to review and approve the operation of the satellite in accordance with the requirements.

Our system simulation is written in MATLAB. Each sub-system has a mathematical model, which is in the form of a state space ($\dot{x} = f(x, u)$ where x is a vector of variables and u is some controller for the system). Each of those state spaces is propagated through time. An infrastructure was built to link each of them to the rest. For example, the attitude control affects the area exposed to the sun which, in turn, affects the satellite temperature and power production. Another example is of the satellite orbit correction, which the satellite does using the thrusters, generating heat in the process. The simulation will show us if, for a given satellite configuration, the maneuver can be done while also keeping temperature inside operational boundaries.

In our system simulation, we have integrated the orbit simulation, attitude control, thermal analyses, and power system, in such a way that they affect one another. An example would be that moving along its orbit, the relative direction from the sun to the satellite changes, and the attitude control needs to keep the solar panels facing the sun. We also accounted for the placement of the earth and sun relative to the satellite. This allows us to know when the satellite is at night (the night is of the satellite, meaning when it is not illuminated by the sun directly), and when we are able to transmit data back to the ground station on earth. In our modeling of the earth and sun, we took into consideration the approximate years the satellite will be in service (2024-2026).

Every engineering project involves challenges and risk managements. In our project, the risk is increased, since the people working on it change every year, with all the expected communicatory problems it brings. In our case, the code we received from last year's simulation team was lacking in modularity, documentation and was inefficient. Therefore, our main goal during winter semester, was rewriting the entire code base, making it as coherent, modular, and efficient as possible, making future changes easier and helping future teams understand how the entire simulation works in minimal time. For spring semester, the focus was on integrating updated modules, implementing different control laws (e.g., cruise control, detumbling etc) and writing documentation for the benefit of future teams.

9.2 Requirements

The process of setting clear and well-structured requirements holds immense value for our simulation development team. Requirements serve as the guiding principles that align our collective efforts towards a common goal. By explicitly defining the scope, functionalities, and expected outcomes of our simulation project, we establish a shared understanding among team members, ensuring that our work is synchronized, and purpose driven. As we navigate the complexities of simulation development, having established requirements provides a roadmap for decision-making, ensuring that our enhancements, integrations, and optimizations are aligned with the overarching project vision. These requirements not only streamline our development process but also facilitate clear communication with interested parties, enabling us to showcase our progress and effectively manage expectations. Here are the following Requirements we established.

9.2.1 Accurate Simulation

The simulation must accurately model the behavior of a satellite orbiting the Moon, considering the specified orbital parameters, lunar dynamics, and gravitational forces.

9.2.2 Subsystem Integration

The simulation should seamlessly integrate various subsystems, including orbit propagation, attitude control, thermal analysis, power system, and more, to accurately reflect the interactions between different aspects of the satellite.

9.2.3 Modularity

The simulation code should be modular, allowing different teams to work on individual components independently while ensuring compatibility with the overall system.

9.2.4 Efficiency

The simulation should be efficient in terms of computational resources and runtime, allowing for timely analysis and iterative testing.

9.2.5 Control Logic

The simulation should implement control logic that manages the satellite's attitude and orbit according to different scenarios and control laws, demonstrating the satellite's responsiveness to external factors.

9.2.6 Documentation

The simulation codebase must be well-documented, including clear explanations of variables, functions, algorithms, and design decisions. This documentation should aid current and future team members in understanding and extending the simulation.

9.2.7 Usability

The simulation should be user-friendly, allowing team members to easily configure simulation parameters, execute simulations, and interpret results without requiring extensive expertise in all areas of satellite engineering.

9.2.8 Scalability

The simulation should be designed to handle increasing complexity, allowing for the addition of new modules, control laws, and scenarios as the project evolves.

9.2.9 Error Handling

The simulation should handle exceptional cases gracefully and provide clear error messages or warnings in case of unexpected input or conditions.

9.2.10 Version Control

The codebase should be managed using version control tools (e.g., Git), allowing for collaborative development and effective tracking of changes across different iterations and team members.

9.2.11 Compliance

The simulation should adhere to coding standards and best practices to ensure maintainability, readability, and consistency across different parts of the codebase.

9.3 Risk Management

9.3.1 Team Turnover

Risk: The project experiences turnover of team members each year, leading to knowledge gaps, communication challenges, and potential disruptions.

Management: Implement thorough documentation and knowledge transfer processes to ensure that new team members can quickly understand the codebase and project context. Establish mentoring or onboarding procedures to help new members get up to speed.

9.3.2 Codebase Complexity

Risk: The simulation codebase can become complex and difficult to maintain over time, leading to inefficiencies, bugs, and difficulties in making changes.

Management: Emphasize code modularity and maintainability during development. Regularly review and refactor the code to improve its structure and eliminate redundancies. Enforce coding standards and practices that promote readability and consistency.

9.3.3 Accuracy and Validation

Risk: The simulation might produce inaccurate results if the models or parameters used do not accurately represent the real-world conditions.

Management: Validate the simulation against theoretical calculations, benchmark scenarios, or real data when available. Document assumptions and limitations of the models used. Conduct sensitivity analyses to understand the impact of parameter variations on results.

9.3.4 Integration Challenges

Risk: Integrating various subsystems (orbit, attitude control, thermal analysis, etc.) may lead to unexpected conflicts or issues that affect the overall simulation.

Management: Use a modular approach that isolates subsystems and clearly defines their interfaces. Implement integration testing to identify and address any compatibility or communication problems between modules.

9.3.5 Communication Breakdown

Risk: Miscommunication between team members could lead to misunderstandings, delays, and errors in the simulation's development.

Management: Establish regular communication channels and meetings to ensure that all team members are aligned on project goals, progress, and challenges. Use project management tools to track tasks, assignments, and deadlines.

9.3.6 Changing Requirements

Risk: Requirements for the simulation may change as the project progresses, potentially impacting the simulation's scope and goals.

Management: Maintain a flexible development approach that allows for iterative updates and adaptations to changing requirements. Prioritize requirements based on their criticality to the project's success.

9.3.7 Software Bugs and Errors

Risk: Undetected software bugs or errors in the simulation code could lead to incorrect results and unreliable simulations.

Management: Implement thorough testing procedures, including testing, integration testing, and validation against known scenarios. Encourage peer code reviews to identify and rectify issues early in the development process.

9.4 Objectives

9.4.1 Codebase Rewriting and Enhancement

- **Codebase Coherence and Structure**: The primary objective was to comprehensively rewrite the existing simulation codebase, transforming it into a coherent and structured framework. This involved reorganizing code sections, introducing modularization, and optimizing the logic flow for improved readability and maintainability.
- **Efficiency Optimization**: Our aim was to enhance the code's efficiency by identifying and addressing performance bottlenecks. The focus was on optimizing algorithms, minimizing redundant computations, and streamlining execution paths, ultimately leading to reduced processing times.
- **Input File Implementation**: A key objective was the integration of an input file mechanism. This involved creating a centralized JSON input file to store simulation parameters, allowing for easy configuration adjustments without manual code modifications. This addition streamlined parameter management and improved code maintainability.
- **Documentation Enhancement:** We aimed to significantly enhance code documentation. This included introducing comprehensive comments to functions, algorithms, and variables, as well as documenting design decisions and assumptions. Improved documentation aimed to ensure code understanding and facilitate collaboration among team members.
- **Preparation for Module Integration**: Another objective was to prepare the codebase for the seamless integration of additional modules, such as attitude control, thermal analysis, power management, and payload simulation. This involved establishing a well-structured foundation that could accommodate these modules without causing disruptions.

By achieving these objectives, our team aimed to transform the initial codebase into a highly functional, well-organized, and efficient simulation framework. Unfortunately, not all the objectives were met as intended. These unfulfilled objectives will be elaborated upon in the section dedicated to future work.

9.4.2 State Machine Implementation

- **Design and Implement State Machine Logic**: Develop a robust state machine, collaboratively designed by the system engineering team, to govern the satellite's behavior throughout different operational phases.
- **Integration with Existing Modules**: Integrate the state machine with existing modules, such as attitude control, thermal analysis, and power management. Ensure seamless coordination between the state machine and these modules to reflect real-world scenarios accurately.
- **Responsive Simulation Behavior**: Ensure that the state machine-driven simulation responds dynamically to changes in operational states. Implement logic that triggers appropriate control actions, module activations, and data recording based on the satellite's current state.
- **Enhance Realism and Scenario Testing**: Leverage the state machine to simulate different mission scenarios and operational conditions. Improve the simulation's realism by modeling the satellite's behavior under different operational constraints, emergencies, and transitions.

9.5 Methodology

9.5.1 Problem Formulation

9.5.1.1 Orbital Dynamics Differential Equations

The orbital dynamics are being modeled using the orbit_StateSpace function. The Differential Equations describe the motion of the satellite in its orbit around the moon, accounting for gravitational perturbations (J2, J3, J4, J5 terms) and other relevant factors.

9.5.1.2 Attitude Control Dynamics Differential Equations

The attitude dynamics are being modeled using the Jstep_State_Space function. This involves the behavior of the satellite's attitude control system, including its angular position, angular velocity, and other related parameters. The control gains and torques are used to determine how the satellite adjusts its orientation.

9.5.2 Numerical Methodology

The simulation's numerical integration is a crucial aspect of accurately capturing the satellite's behavior over time. To achieve this, the ode45 solver, a Runge-Kutta method, is employed. Unlike simple methods like the Euler method, ode45 adapts its time step size dynamically to maintain precision throughout the simulation. This solver uses a combination of fourth and fifth order Runge-Kutta formulas to estimate the solution at each step. The adaptive nature of ode45 contributes to the simulation's accuracy and stability, allowing us to model the intricate interactions of various subsystems under different conditions.

9.5.2.1 Solver Invocation

The ode45 solver is invoked to numerically approximate the solution of these ODEs. Its adaptive step-size algorithm ensures both accuracy and efficiency by dynamically adjusting step sizes to accommodate variations in the system's behavior.

9.5.2.2 Integration Process

During each iteration of the numerical simulation loop, the solver's execution commences. It progressively integrates the system's state variables over a prescribed time interval, employing the ODE functions to compute the changes in position, velocity, attitude, angular velocity, and control inputs.

9.5.2.3 Integration Step

The solver advances the simulation by iteratively integrating the state variables using the selected ODE functions, culminating in a comprehensive representation of the satellite's state at the end of the interval.

9.5.2.4 Accurate Approximation

The solver's adaptive nature, coupled with its integration algorithm, ensures that the simulated state remains close to the actual behavior of the system, even in scenarios with complex and nonlinear dynamics.

9.6 Flow Chart

The flowchart presented in this section visually outlines the sequence of operations and interactions within the developed simulation code. This flowchart serves as a roadmap to understanding the different modules and components of the code, highlighting the key stages of simulation, data processing, and decision-making.

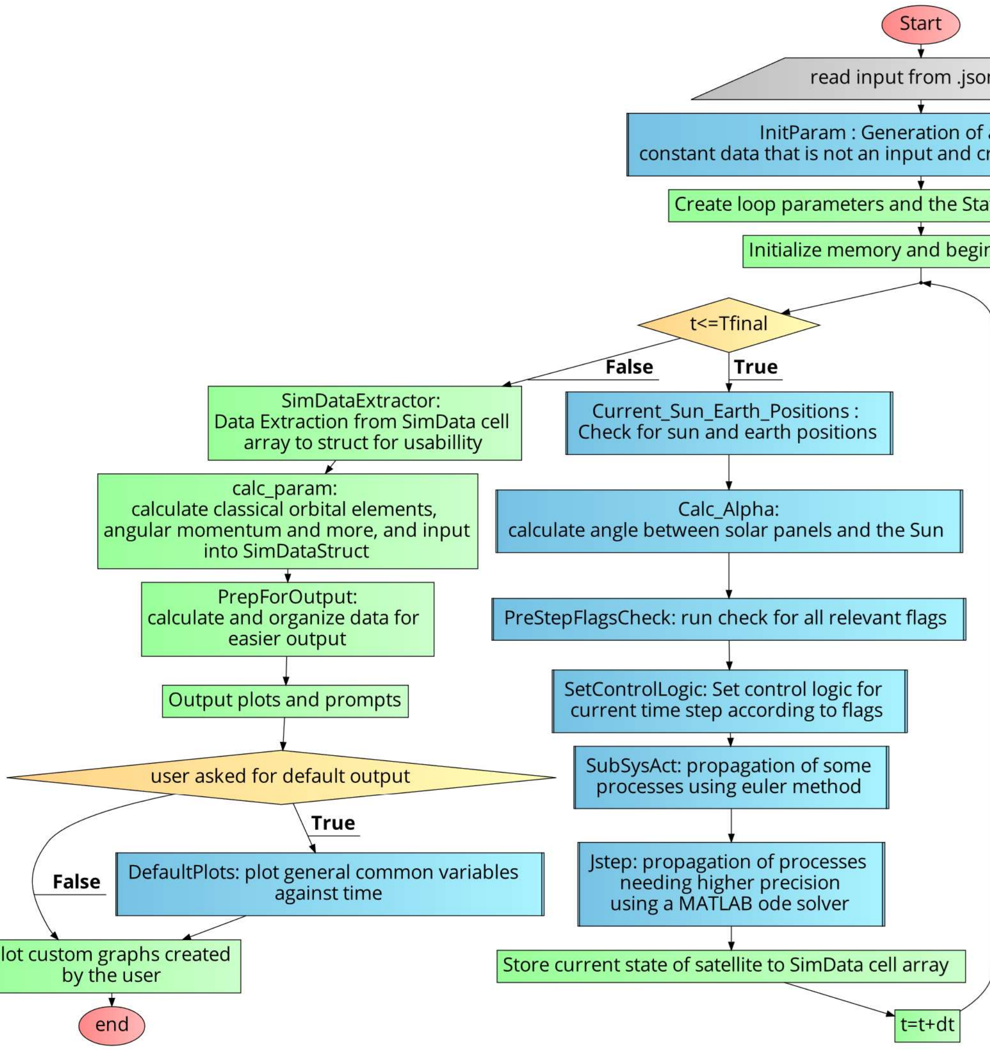


Figure 110: Main code flow chart.

9.7 Future Work

9.7.1 Graphical User Interface (GUI) Implementation

Implementing a Graphical User Interface (GUI) offers a user-friendly way to interact with and visualize the satellite simulation. It allows for parameter adjustments, visualization of simulation outputs, and intuitive control. Given that the team comprises students and considering the learning curve associated with GUI development, this task may require approximately 100-150 work hours.

9.7.2 Enhanced Thermal Module Integration

Incorporating an updated thermal module is crucial for accurate simulation results, especially in space conditions. This involves integrating advanced thermodynamics models, considering radiation and conduction effects. Given the complexity of thermal simulations and the need for meticulous testing, allocating around 150-200 work hours is reasonable.

9.7.3 Payload Module Implementation

Once the payload module is developed, integrating it into the simulation will provide insights into its impact on the satellite's behavior. The complexity of the payload's behavior and interactions with other subsystems will influence the effort required. A preliminary estimate could range from 100 to 200 work hours, depending on the intricacy of the payload's functionalities.

9.7.4 State Machine Implementation

Creating a robust and functional state machine from scratch is a significant endeavor. This involves designing the state transitions, handling various operational phases, and ensuring smooth coordination between different subsystems. Given the complexity and the need for careful planning, the implementation of the state machine could require approximately 200-250 work hours.

9.8 Conclusion

Reflecting on the second year of the JERICCO project, the role of our simulation team has been pivotal. We've revamped the code, integrated essential features such as heat management and power allocation, and laid the groundwork for the forthcoming payload system. We've delved deep into understanding how the satellite behaves in space using tools like the ode45 solver. This has enabled us and the other teams to make informed decisions, swiftly address challenges, and plan strategically. Overall, our simulation team has been instrumental in guiding the JERICCO project towards successful navigation in the realm of space exploration.

9.9 References

1. Final Year Report, FDR - JERICCO Project 2021/2022
2. Authors: Amit Malka., Ilay Lazarovich

   Title: JERICCO Simulation Code

   Version: 2.0

   Year: 2023

   Code and Documentation Location (Project Drive): Project JERICCO 2022/23 >

   Teams Folder > Simulation Folder