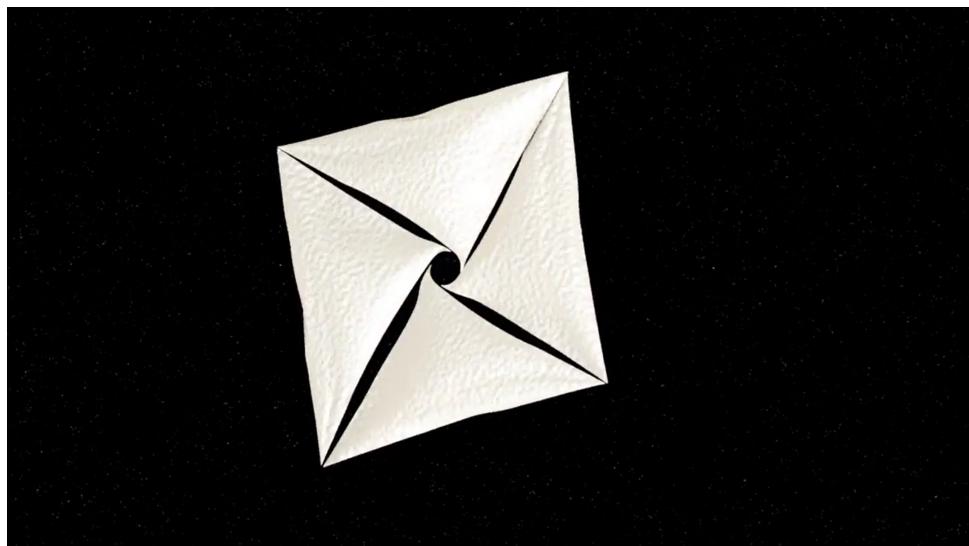


Preliminary Design of a Solar Sail Propelled Spacecraft for Extrasolar Trajectories



Screenshot from animation of sailcraft deployment by ICSS [1].

Individual Final Year Project

Student: Tanmay Ubgade
CID: 01503854

Academic supervisor: Dr Aaron Knoll
Second marker: Dr Davide Amato
Department: Department of Aeronautics
Course: H415 MEng in Aeronautics
with Spacecraft Engineering
Academic year: 2021-2022
Date of submission: 06/06/2022

Department of Aeronautics
South Kensington Campus
Imperial College London
London SW7 2AZ
U.K.

Abstract

This thesis is a continuation of a project started in 2018, aiming to design a solar sailcraft for extrasolar trajectories in a CubeSat configuration. This iteration expands on the sailcraft design proposed in the last works, implementing an active tracking system as well as a balanced and validated subsystem design in a 3U CubeSat specification with a reviewed mass budget. An attitude sensitivity study was conducted on the updated sailcraft design which lead to the conclusion that any practical and feasible designs of CubeSat fall outside the extremely tight bounds of center of mass offset of ± 0.1 mm from the origin. As the current mission concept of a passive gyroscopically spin-stabilised sailcraft was proven to be unfeasible, suggestions were given for future study. In particular, an active attitude control systems was deemed necessary to stabilise the sailcraft during its trajectory. Further studies and investigations are needed to advance the proposed mission towards feasibility and materialisation.

Acknowledgements

First and foremost, I would like to give massive thanks towards my friends who have continually supported me throughout the duration of my final year and particularly towards the completion of this thesis. I would like to particularly give appreciation to the following amazing people: Jenny, Tonye, Giulio, Ren, Palash, Ryo, Oscar, Haganta, Ray, Daniele, Hayley, Agnese, Zan and Aoi. Much love to you guys for everything! I'd also like to thank my parents for their unfaltering support and love throughout my life, without which I would not have been able to get to this point.

Of course, I would also like to thank Dr Knoll for his continual support and guidance throughout the process to help lead this project towards a practical and achievable future. I hope to see this mission come to fruition one day in the future, not only for the joy of knowing my efforts helped to partly lead us there, but also that we all get to learn just a bit more of this crazy universe and how it works. Seriously, light pushing a hunk of metal out of the solar system, that is certifiably insane and amazing.

Contents

Abstract	iii
Table of Contents	iii
List of Figures	iv
List of Tables	vi
List of Algorithms	vi
List of Symbols	viii
1 Introduction	1
1.1 Motivations	1
1.2 Mission objectives	1
1.3 Previous work	2
1.3.1 Trajectory characteristics	3
1.3.2 Sailcraft characteristics	3
1.3.3 Areas to explore	4
1.4 Thesis objectives	4
2 Sailcraft design	5
2.1 Mass Sensitivity Study	5
2.2 Hub sizing	7
2.2.1 CubeSat specification choice	7
2.2.1.1 Internal layout	7
2.2.2 Folding schemes	7
2.2.3 Wrapping of sail	9
2.2.4 Tip weight	9
2.3 Subsystem sizing	10
2.3.1 Attitude determination and control	10
2.3.2 Command and data handling	11
2.3.3 Telemetry, tracking, and command	11
2.3.4 Power	14
2.3.4.1 Consumption	14
2.3.4.2 Batteries	15
2.3.4.3 Solar panels	15
2.3.4.4 Thin-film solar cells	17
2.3.4.5 Alternative power sources	18
2.3.5 Energy capacity balance of overall system	18
2.4 Sailcraft mass budget	20

3 Attitude sensitivity study	21
3.1 Problem setup	21
3.1.1 Theory	21
3.1.1.1 Solar radiation pressure	22
3.1.2 Assumptions	24
3.1.3 Reference frames	25
3.1.3.1 Rotations to sail	25
3.1.4 Moment of inertia	26
3.1.5 Numerical implementation	27
3.1.6 Method validation	27
3.2 Center of gravity offset study	28
3.2.1 Results	28
3.2.2 Discussion	31
4 Mission scenario	33
4.1 Initial reference case	33
4.2 Suggestions for future of project	34
5 Conclusion	35
5.1 Future work	35
A MATLAB functions and scripts	40
A.1 Functions	40
A.1.1 Importing GMAT Data	40
A.1.2 Solar Intensity	42
A.1.3 Temperature of Sail	42
A.1.4 Direct Cosine Rotation Matrix	42
A.1.5 Moment of Inertia	43
A.1.6 Angular acceleration from torque (Euler equations)	43
A.1.7 Torque on sail (ideal case)	44
A.1.8 Torque on sail (CoG offset case)	45
A.2 Scripts	46
A.2.1 Hub sizing	46
A.2.2 Power Transmission	48
A.2.3 Energy capacity balancing	50
A.2.4 Earth-Sail graphs	55
A.2.5 Attitude sensitivity - ideal case	56
A.2.6 Attitude sensitivity - CoG offset case	58

List of Figures

1	Overhead view of trajectory simulations of all tested masses of the sailcraft, with a sail size of 28.6 m^2	6
2	Visualisation of standard CubeSat specifications [2].	7
3	Proposed internal layout of the bottom CubeSat unit, Top view. Dimensions in mm.	8
4	Deployment process of IKAROS. (a) Tip mass separation and first stage deployment. (b) Second stage deployment. [3]	8
5	Folding patterns for a sail membrane quadrant. (a) Entirely-parallel z-folds. (b) Half-parallel z-folds. (c) Partial fan fold. [4].	9
6	Representation of how the folded layers the membrane quadrant stack with entirely-parallel z-folds (above) and half parallel z-folds or partial fan folds (below) [4].	9
7	Distance between Earth and Sailcraft over the duration of the active mission.	12
8	Link budget design equation variation to decide optimal values.	13
9	Bit Error Probability (BER) as a function of E_b/N_o [5].	13
10	Pie chart showing distribution of power by subsystem.	14
11	Power consumption of the system with daily, weekly, and monthly tracking rates.	15
12	Diagram showcasing the minimum shadow angle from which the entire solar panel is visible to Sun and able to generate power at full capacity. Diagram not to scale.	16
13	Cumulative power generation of the solar panels over the duration of the active mission.	16
14	Cumulative power generation of the thin-film solar cells over the duration of the active mission.	17
15	Energy generation of entire system during the mission.	19
16	Energy capacity balance over the duration of the active mission.	19
17	Distance between the sailcraft and Sun with indication of death of the system due to loss of energy capacity over the duration of the active mission.	19
18	Pie chart showing mass distribution by subsystem.	20
19	Diagram showcasing the variation of SRP across the sail. The negative region of x is closer to the Sun and positive region is further.	23
20	SRP variation over sail in different reference frames at $r = 0.13 \text{ AU}$	23
21	Relevant reference frames in attitude sensitivity study.	25
22	CAD of sailcraft used in calculating the moment of inertia about the origin.	26
23	Torque variation about principal axes at $r_{sail} = 0.12 \text{ AU}$, $\alpha = 90^\circ$	27
24	Rotation change over the mission based on varied CoG offset.	29
25	Angular velocity change over the mission based on varied CoG offset.	30
26	Angular acceleration change over the mission based on varied CoG offset.	31
27	Rotation change over the mission for -0.1 to 0.1 mm CoG offset.	32
28	Orientation of sailcraft over one orbit. Trajectory does not accounting for change in eccentricity due to sail. Diagram not to scale.	33
29	Distance from Sun over the mission.	33

30	Magnitude of velocity in Sun inertial frame over the mission.	33
31	Temperature over the duration of the active mission.	34

List of Tables

1	Orbital elements for the initial reference case, defined with an initial epoch of 27 Feb 2027 10:00:00.000 [6].	3
2	Design envelope for key parameters of the sailcraft from Geragidis' study [6]. . .	3
3	GMAT simulation results from the mass sensitivity study to investigate the mission timeline and trajectory as a result of the mass of the sailcraft.	6
4	Properties of a Miniature Spinning Sun Sensor by Redwire Space [7].	11
5	Properties of a generic on-board computer (iOBC) for space missions [8].	11
6	Relevant properties of chosen TT&C subsystem [9].	11
7	Power budget of the sailcraft system.	14
8	Mass budget breakdown by each subsystem to give total mass of sailcraft.	20
9	The distribution of mass and volume to give an averaged effective uniform density of the CubeSat and the sail to be used for computing \mathbf{I} in Fusion 360.	26
10	Rotation change about X and Y axes varied by CoG offset.	30

List of Algorithms

1	An algorithm used to compute wrapping thickness of sail in the spool. Dimensions of all variables are in mm.	10
---	--	----

List of Symbols

Acronyms

ADCS Attitude Determination and Control

AOP Argument of perigee

BER Bit error probability

C&DH Command and Data Handling

CoG Center of Gravity

ECC Eccentricity

EIRP Effective isotropic radiated power [dB]

INC Inclination

RAAN Right ascension of the ascending node

RTG Radioisotope Thermoelectric Generators

SMA Semi-major axis

SRP	Solar Radiation Pressure
TA	True anomaly
TT&C	Telemetry, Tracking and Command

Physics Constants

c	Speed of light in a vacuum	$299\,792\,458 \text{ m s}^{-1}$
k	Stefan-Boltzmann constant	$5.67\text{e-}8 \text{ W}\cdot\text{m}^{-2}\text{K}^{-4}$
L_{\odot}	Solar luminosity	$3.828\text{e}26 \text{ W}$

Variables

α	Angle of attack of sail to sunlight [°]
χ	Euler angles [°]
ω	Angular velocity [$\text{rads}\cdot\text{s}^{-1}$]
\mathbf{H}	Total angular momentum [$\text{rads}\cdot\text{s}^{-1}$]
\mathbf{I}	Angular momentum [$\text{kg}\cdot\text{m}^2$]
\mathbf{L}	Angular momentum [rads]
\mathbf{T}	External torque [Nm]
η	Receiving antenna efficiency [-]
γ	Anti-clockwise rotation about spin-axis (Z-axis) [°]
μ	Efficiency [-]
ω_{sc}	Angular velocity of sailcraft about spin-axis [RPM]
ρ_e	Reflectivity of emissive side [-]
ρ_{Kapton}	Density of Kapton [$\text{kg}\cdot\text{m}^{-3}$]
ρ_{Mylar}	Density of Mylar [$\text{kg}\cdot\text{m}^{-3}$]
ρ_r	Reflectivity of reflective side [-]
θ	True anomaly [°]
θ_x	Major axis of half-power beamwidth angle [°]
θ_y	Minor axis of half-power beamwidth angle [°]
A_{sc}	Area of sail [m^2]
A_{solar}	Area of respective solar power cells [m^2]
c_p	Center of pressure
$circ$	Circumference based on layers wrapped [mm]
D	Receiving antenna diameter [m]

E_b/N_o	Ratio of received energy-per-bit to noise [dB]
f	Frequency [Hz]
$F_{SRP,tot}$	Total force imparted by solar radiation pressure on sail [N]
F_{SRP}	Force distribution imparted by solar radiation pressure [N]
G_r	Receiver antenna gain [dB]
G_t	Transmitter antenna gain [dB]
L_α	Transmission path loss [dB]
L_l	Transmitter-to-antenna line loss [dB]
L_S	Space loss [dB]
L_d	Lifetime degradation [-]
l_{rem}	Remaining length of folded sail being wrapped [mm]
l_{side}	Length of side of sail [m]
m_{sail}	Mass of sail [kg]
m_{tip}	Mass of tip weight [kg]
P_t	Transmitter power [dBW]
P_{gen}	Power generated [W]
P_{SRP}	Pressure imparted by solar radiation pressure [μPa]
R_{data}	Data rate [bps]
r_{sail}	Radius of Sun-Sail vector [m]
r_s	Axis of Sun-Sail vector [m]
S	Path length [m]
T_1	Torque about 1st principal axis [Nm]
T_2	Torque about 2nd principal axis [Nm]
t_e	Time elapsed [s]
T_g	Glass transition temperature [K]
T_s	System noise temperature [K]
t_{wrap}	Thickness of wrapped layers of sail [mm]
x_{rot}	x-coordinates of rotated sail in body-inertial coordinates [m]
y_{rot}	y-coordinates of rotated sail in body-inertial coordinates [m]

1 Introduction

1.1 Motivations

Humanity always strives to explore and understand the unknown, continually pushing the frontier. In the late 1900s, big strides were made in exploring celestial bodies within the Solar System, even passing the edge of our local system [10]. To date, there are 5 spacecrafts that mankind has launched that will be leaving the Solar System: Pioneer 10, Pioneer 11, Voyager 1, Voyager 2, and New Horizons [11, 10, 12, 13]. Of these, all but New Horizons have escaped the solar system, with only Voyager 1 and 2 being still operational until 2025. Each of these probes were placed into their escape trajectory by multistage rockets which used solid fuel, in combination with gravity assist manoeuvres [14]. Of these missions, New Horizons is the only spacecraft that was launched directly into an extrasolar trajectory without need for a gravity assist, but it came at a large cost of \$565 million in development and launch costs [15].

A cheaper and lighter alternative to such mission scenarios can be achieved by developing the technology of Solar Sails, which utilise solar radiation pressure (SRP) [16]. SRP is a phenomenon that effects all spacecraft, where it can displace trajectories and change orientation of the spacecraft [17]. This effect can be used advantageously with the use of Solar Sails. Some notable Solar Sail missions are IKAROS by JAXA, the first successful Solar Sail mission deployed on a voyage towards Venus, and LightSail projects by the Planetary Society, deployed in Earth orbit in a 3U CubeSat [18, 19]. These Solar Sail missions cost approximately 90% less than previous interstellar missions as they are much lighter and take up minimal space pre-deployment [20]. While the scopes of the aforementioned Solar Sail missions and extrasolar missions are drastically different in scale, the majority of the cost lies towards the launch and propulsion rather than research and development.

This thesis is a continuation of a project that started in 2018, aiming to produce a CubeSat probe to reach interstellar space. Next, the mission objectives are stated, followed by a summary of the contributions of the previous works, which consist of a feasibility study, a preliminary design, and preliminary structural design.

1.2 Mission objectives

To reach interstellar space, the sailcraft needs to pass the boundary of the solar system known as the heliopause, which lies at a distance of 123 AU. This is the region where the flux of matter emanating from the Sun (solar winds) is stopped by flux from the interstellar medium (stellar winds). [21, 22]. This definition can be considered the interstellar equivalent of the von Karman line.

The primary objectives of the mission are:

1. To launch a solar sailcraft to exit the Solar System within 100 years to maximise its scientific utility
2. To carry out measurements of the sailcraft's trajectory and produce a method of designing larger-scale extrasolar missions

In the grander context of the project, there are a number of secondary objectives, which are:

1. To produce a high-TRL solar sailcraft

2. To advance CubeSat capabilities
3. To attract funding for CubeSat development in subsequent years
4. To establish Imperial College London as leader for CubeSats within UK and an innovator in space technologies worldwide

1.3 Previous work

This section will summarise the relevant findings from the 3 prior works on this project to build an initial reference case of the trajectory and sailcraft design to provide a foundation for what needs to be investigated going forward. The areas explored in this particular thesis will be explained in Section 1.4.

The three previous works are as follow: a feasibility study of the mission by Moore, a preliminary design by Geragidis, and a preliminary structural design by Montete. Moore's paper successfully proved the feasibility of an extrasolar mission using a passive gyroscopically stabilised solar sail. A novel trajectory was suggested: rather than utilising a constant cone angle between the vector to the Sun and the sail's normal like in previous solar sail proposals, this concept recommends a constant angle in the inertial reference frame. This mission concept results in a low complexity, mass, and cost design where the orbit's periapsis decays and apoapsis rises on each passing due to deceleration when moving towards the Sun and acceleration when moving away from the Sun respectively. Suggestions were given to investigate utilising a Mars transfer trajectory as a secondary piggyback payload with a kick motor, where the sail is deployed using gyroscopic forces without mechanical booms for more mass savings [23].

Both the preliminary design and preliminary structural design papers were done concurrently, thus based on the work of Moore's paper from 2018. As a result, the structural design analysed in Montete's paper is outdated due to advancements made in Geragidis' paper, however the methodology employed will be useful for future studies. Additionally, there was useful finding regarding the tracking of the satellite. Moore's and Filippou's work proposed utilising reflectors, either at the hub or corners) for tracking the satellite with laser detection and ranging (LADAR or radar). Unfortunately, there was an arithmetic error which when accounted for, resulted in the reflectors needing to be kilometers in size rather than a few meters, thus deemed unfeasible. Consequently, Montete investigated the implementation of active tracking and power systems on the previously simple sailcraft [24].

Geragidis' paper explored orbit propagation using GMAT and sailcraft design. Regarding the trajectory, Geragidis demonstrated the feasibility of using a Mars transfer orbit without a kick-burn as the candidate mission scenario [6]. In terms of sailcraft design, a preliminary design study was conducted to provide a carrier sizing to CubeSat specifications and mass budgets based on primary components. Again, this analysis still assumed the use of reflectors for tracking, thus no subsystems were included in this sizing. At the end of the study there was a design envelope study to size down the sailcraft from the original 454 m^2 sail to a more practical size for CubeSat missions. The relevant findings are summarised in Table 2.

The next two sections will summarise the main trajectory and sailcraft characteristics that will be considered for the exploration in this thesis.

1.3.1 Trajectory characteristics

The candidate mission proposed by Geragidis has an initialisation of orbital element parameters that is tabulated in Table 1. This candidate trajectory is of a sailcraft deployed at the apoapsis of a Mars transfer orbit, where the sail size is 454 m^2 . This is to be scaled down during the exploration of this thesis.

Table 1: Orbital elements for the initial reference case, defined with an initial epoch of 27 Feb 2027 10:00:00.000 [6].

SMA (km)	ECC (-)	INC ($^\circ$)	RAAN ($^\circ$)	AOP ($^\circ$)	TA ($^\circ$)
1.89e8	0.2076	0	0	360	180

1.3.2 Sailcraft characteristics

As per a design envelope study done by Geragidis, the sailcraft has been significantly sized down from the feasibility studies of the mission concept, and the relevant parameters are tabulated in Table 2.

Table 2: Design envelope for key parameters of the sailcraft from Geragidis' study [6].

Parameter	Minimum	Design	Maximum	Units
A_{sc}	7.6	12.4	28.6	m^2
l_{side}	2.739	3.493	5.320	m
m_{sail}	21.584	35.216	81.224	g
ω_{sc}	0.32	0.49	0.63	RPM

The sail itself has a thickness of $2 \mu\text{m}$ and is made of one of two candidate materials: DuPont Kapton (HN polyimide) where $\rho_{Kapton} = 1420 \text{ kg}\cdot\text{m}^{-3}$ and $T_{g,Kapton} = 633 - 683K$; or Mylar (polyethylene terephthalate) where $\rho_{Mylar} = 1390 \text{ kg}\cdot\text{m}^{-3}$ and $T_{g,Mylar} = 443K$ [25, 26]. Mylar was considered as the restrictive material due to its lower glass transition temperature during Geragidis and Moore' analysis, but this investigation will consider Kapton as the sail material for its higher glass transition temperature due to potentially closer proximity to the Sun at cost of marginally higher mass.

Montete's paper did propose the use of a boom design due to structural considerations such as wrinkling and spin-up dynamics, the analysis was conducted at a much larger scale and the arguments towards the boom design weren't definitively conclusive due to the conclusions (design envelope) from Geragidis' report [24]. As a scaled down version of the gyroscopically stabilised sail is to be investigated, the boom design will not be explored in this thesis, but should be explored further as a concurrent alternative mission design in a future study.

Furthermore, as Montete's paper highlighted, there is a need for additional subsystems for active tracking [24]. The investigation particularly explored the communications subsystem and the power subsystem. Regarding the communications system, it was modelled approximately off other interstellar probes like Voyager for an initial estimate, thus further investigation is required for CubeSat specification. Then regarding power subsystem, it was concluded that solely using batteries would be far too heavy, therefore requiring power generation. The proposed solutionw as to utilise thin-film solar cells, particularly copper indium gallium diselenide (CIGS). CIGS is the most promising material compared to the likes of hydrogenated amorphous silicon and

multi-junction p-i-n structures [27]. CIGS cells have efficiency potentials of upto 15%, making them very attractive for space applications. For solar sail applications, two TSFC (flexible CIGS thin-film solar cells) devices are selected. These are Ti foil substrate ($118 \text{ g}\cdot\text{m}^{-2}$ and $\mu = 13.5\%$) and PI substrate ($41 \text{ g}\cdot\text{m}^{-2}$ and $\mu = 10.4\%$). Due to its higher considerably higher specific power, PI substrates are favoured. It is possible to increase the efficiency of the cells to 14.1% by application of a Na post-treatment method [28].

1.3.3 Areas to explore

- Comprehensive subsystem design followed by component-level design
 - ADCS, TT&C, structural and power subsystem studies
- Update of reference structural design
- Study of folding and deployment schemes, using Finite Element Analysis
- Second iteration of trajectory analysis with a list of optimum trajectories, leading to a new reference case
- Conducting an attitude sensitivity study, and constructing an integrated trajectory simulator
- Design of the carrier spacecraft

1.4 Thesis objectives

As seen in Section 1.3.3, there are several avenues to explore regarding the design of the solar sailcraft, but there are a few vital ones that need to be tackled urgently. The most important area to explore is the attitude sensitivity analysis. So far, the assumption has been the passive gyroscopically stabilised sailcraft about the spin-axis will remain planar in the Sun inertial reference frame, but this assumption needs to be verified, else the sailcraft design will have to drastically change. To conduct this analysis, an accurate mass and size estimate is needed on the sailcraft's design; and as such, a preliminary design will also need to be conducted on the sailcraft to size the structure and its subsystems in a CubeSat specification. Therefore, the thesis objectives are as follow:

1. Further-refine the solar sailcraft's preliminary design
 - (a) Validate the usage of a standard CubeSat specification
 - (b) Defining the requirements and sizing of each subsystem
 - (c) Update power and mass budget
2. Explore the solar sailcraft's attitude stability over the mission

The first objective is investigated in Section 2 and the second objective is investigated in Section 3. Thereafter, Section 4 contains the details of the reference case utilised for analysis in the studies of this thesis followed by a discussion of the implication of the results of the attitude sensitivity study. Finally, Section 5 summarises the findings and conclusions of the report with a list of recommended future work in Section 5.1.

2 Sailcraft design

It is vital to gain more clarity on the intricacies of the proposed design of the sailcraft before any further analysis is done it, as they may be rendered invalid if the scope of the project changes. Therefore, this section will aim to further-refine the sailcraft's preliminary design.

In the previous iteration of the project, Geragidis proposed a design envelope for the sailcraft's and carrier's design, which will be utilised for the bounds of exploration in this section [6]. It was found by varying parameters in GMAT simulations that the main driving factor in the results of the trajectory analysis was the mass of the sailcraft, which is explored in Section 2.1. With mass as the design driver, this section will explore the hub sizing to define a particular CubeSat specification for the sailcraft, and the subsystem sizing to provide an updated sailcraft mass budget. The resultant design will then be validated in Section 3. Lastly, the reference trajectory used for the analyses in this section is detailed in Section 4.1.

2.1 Mass Sensitivity Study

A study of the trajectory was done using GMAT to vary parameters provided in the design envelope from the previous work by Geragidis found in Table 2). The two parameters identified to explore within the bounds of the design envelope are the area of the sail and the mass of the sailcraft.

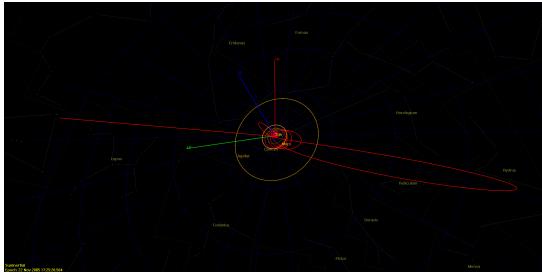
In these studies, the prescribed reference case parameters (Table 1) in Section 1.3 will be the baseline used, only varying the area of sail and mass of sailcraft. As the fidelity of the sailcraft model is unchanged from the last study, the error tolerance remains to be 1e-5 [6]. As GMAT does not account accurately represent the shape of the sailcraft, the attitude rates were set to 0 and the orientation was set for ideal deployment at a true anomaly (θ) of 180° (sail is normal to orbital plane and facing the Sun). Additionally, a sail size of 28.6 m² was chosen as it was the largest value within the design envelope. A larger sail size minimises the time to heliopause as it imparts a greater acceleration for a small increase in mass. This reduction in timescale of the mission is beneficial in subsystem design as it can reduce the required mass of the power systems.

A mass range of [2, 3, 4, 5, 6, 8] kg was considered for the GMAT simulations. The study aimed to find the optimal trajectory where mass can be minimised with the shortest time to the heliopause. In addition, attention was given towards the time till and radius of the maximum apoapsis achieved before reaching its escape trajectory. These values will help inform the sizing of the power systems as the sailcraft needs to be able to communicate with the ground station on Earth until reaching its escape trajectory to relay the mission's data. The point of transition towards its escape trajectory can be considered to be past this maximum apoapsis, and officially on the escape path after the closest approach. Furthermore, the closest approach radius was noted as this can cause the sailcraft to exceed its thermal operating limit, 633-683 K for Kapton [25].

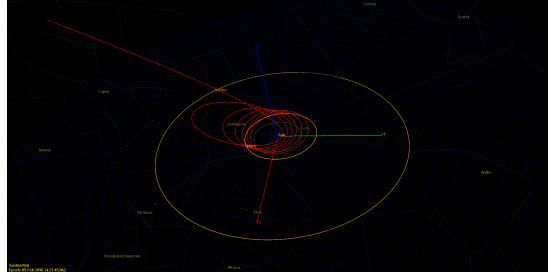
The results of the mass sensitivity study can be found in Table 3, and the resultant trajectories can be seen in Figure 1. From the perspective of closest approach, the mass sizes of 2 kg, 5 kg and 6 kg can be put at lower priority due to having a closest approach to the Sun of under 0.10 AU. Specifically 2 kg can be eliminated since it would be very hard to design a sailcraft with various subsystems to such a low mass limit, in addition to its highly irregular escape pattern seen in Figure 1a. In comparison, Figures 1b-1f follow the expected trajectory pattern of this

Table 3: GMAT simulation results from the mass sensitivity study to investigate the mission timeline and trajectory as a result of the mass of the sailcraft.

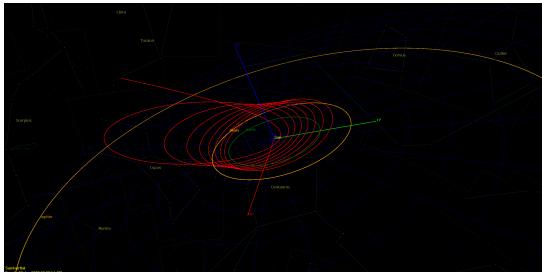
Mass (kg)	Maximum Apoapsis (AU)	Time to Apoapsis (years)	Closest Approach (AU)	Time to 121 AU (years)
2	23.99	27.94	0.07	58.65
3	4.85	10.05	0.18	62.04
4	6.97	15.61	0.13	52.70
5	11.01	23.62	0.09	55.70
6	21.43	40.30	0.04	72.96
8	9.17	32.27	0.14	98.60



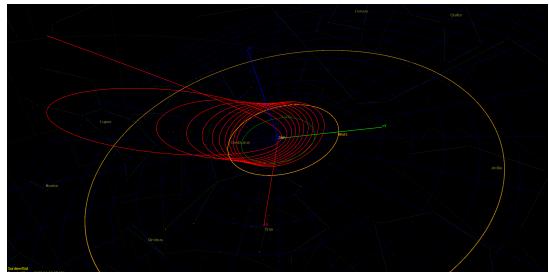
(a) Sailcraft mass of 2 kg.



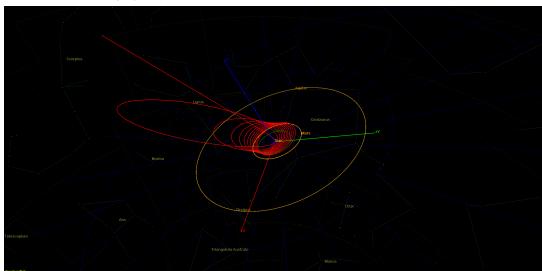
(b) Sailcraft mass of 3 kg.



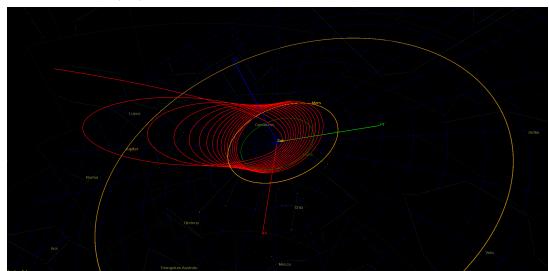
(c) Sailcraft mass of 4 kg.



(d) Sailcraft mass of 5 kg.



(e) Sailcraft mass of 6 kg.



(f) Sailcraft mass of 8 kg.

Figure 1: Overhead view of trajectory simulations of all tested masses of the sailcraft, with a sail size of 28.6 m^2 .

mission concept. However, considering the average lifespan of batteries in space to be about 20 years, 8 kg can be discarded [5]. This leaves the mass range of 3-4 kg as the ideal goal mass of the sailcraft. Accounting for the need of various subsystems, the mass limit is set to 3.99 kg to fit within the bounds of a 3U CubeSat specification [29]. This specification is discussed further in Section 2.2.1.

2.2 Hub sizing

The hub refers to the housing structure of the sail and other subsystems of the sailcraft. As the goal of this mission is to design a sailcraft in a CubeSat configuration, there are standard size specifications that must be met. As discussed in Section 2.1, a sail size of 28.6 m^2 is considered in analysis. In this section, a prospective CubeSat size will be defined in Section 2.2.1, then the folding and wrapping of the sail with a tip weight is investigated to validate this choice.

2.2.1 CubeSat specification choice

As per the secondary goals of this project, the sailcraft is to be modeled as a CubeSat. A unit of a CubeSat is of the size $10 \times 10 \times 10 \text{ cm}$, and each has a mass limit of 1.33 kg [29]. CubeSats have standard size configurations such as 3U and 12U CubeSats like can be seen in Figure 2 [2]. 16U and 27U size CubeSat specification can also be used. The aforementioned CubeSat configurations allow for a base dimensions with a square geometry to effectively fit a wrapped sail around a spool. As the ideal mass range of the sailcraft was found to be 3-4 kg in Section 2.1, this would correspond to a 3U CubeSat or larger to be used. In the interest of minimising volume for efficiency, the design in this section will aim to validate the use of a 3U CubeSat.

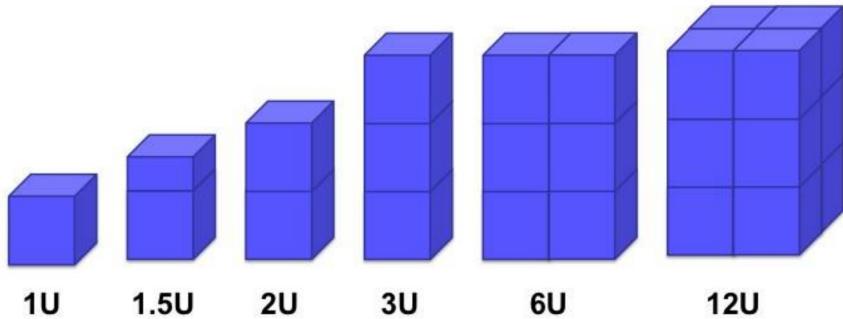


Figure 2: Visualisation of standard CubeSat specifications [2].

2.2.1.1 Internal layout

Within the 3U CubeSat, the stowed sail is to be placed in the bottom unit around a spool. A proposed schematic of the internal layout for the bottom unit is shown in Figure 3. The structure should be made of a lightweight material such as Aluminium 7075, a common material for space applications [30]. The spool around which the sail is to be wrapped is assumed to have a central core with a diameter of 1 cm , and an outer flange diameter of 9 cm .

Since the sail needs to be deployed laterally from this CubeSat unit, there needs to be an unconventional unique structure connecting this unit to the top of the CubeSat. A proposed solution is to connect the structure through the center core of the spool, while the walls will come off during deployment as deployable solar panels (discussed further in Section 2.3.4.3). As such, the sail, which is split into four quadrants, can be pulled out by tip weights similar to how it was achieved with IKAROS, as seen in Figure 4 [3]. This simulates the effect of boom deployment but is induced by the spin of the sailcraft causing centrifugal force to unfold the sail.

2.2.2 Folding schemes

In several solar sail missions, the deployment method tends to use boom deployment methods which tend to use bi-axial folding patterns to shorten the membrane [31]. Since this project's

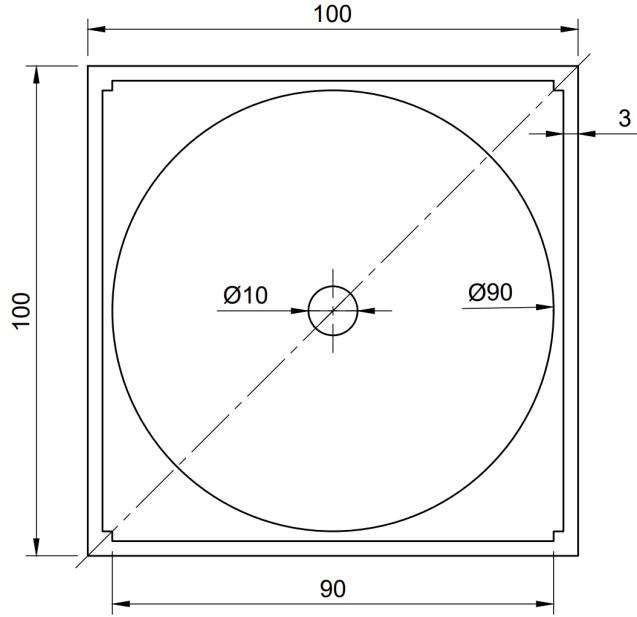


Figure 3: Proposed internal layout of the bottom CubeSat unit, Top view. Dimensions in mm.



Figure 4: Deployment process of IKAROS. (a) Tip mass separation and first stage deployment. (b) Second stage deployment. [3]

focus is on spin-deployment due to mass concerns of implementing a boom, the proposed approach is wrapping the folded sail around the core of the spool, which will be able to rotate due to implementation of bearings. A recent study by NASA on low-cost manufacturing and folding of solar sail membranes for use in CubeSats does the aforementioned with boom deployment, but can be replicated with centrifugal force. The deployment dynamics proposed should be investigated in a future study.

For a design where the folded sail is to be wrapped, the best candidates for folding patterns can be seen in Figure 5. Entirely-parallel z-folds are the simplest folding pattern to implement while half-parallel z-folds are the hardest due to the folds along the centerline. Moreover, it can be seen from Figure 6 entirely-parallel z-folds are volume inefficient at the base. Consequently, it can be concluded that a partial fan-fold be the chosen folding choice of the sail.

To determine the thickness of the folded sail, and overestimation was made for the sake of a

safety margin. The thickness of the sail is $2 \mu\text{m}$, and it was decided that the quadrant would be folded 50 times, leading to a folded thickness, t_{fold} , of 0.1 mm. This value will be used in the next section to calculate the wrapping thickness.

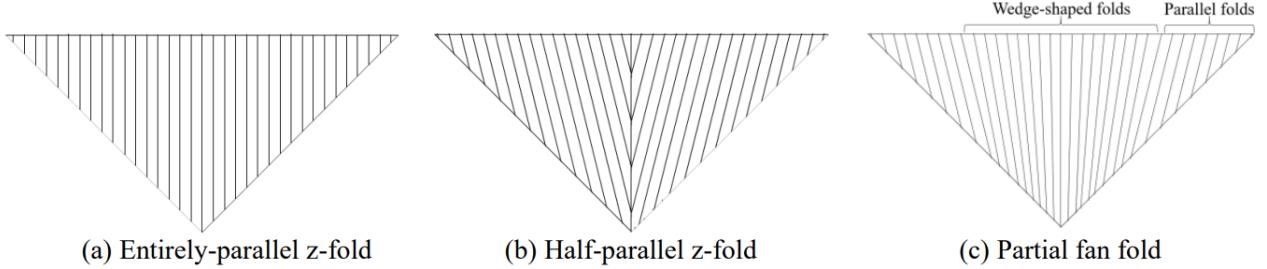


Figure 5: Folding patterns for a sail membrane quadrant. (a) Entirely-parallel z-folds. (b) Half-parallel z-folds. (c) Partial fan fold. [4].

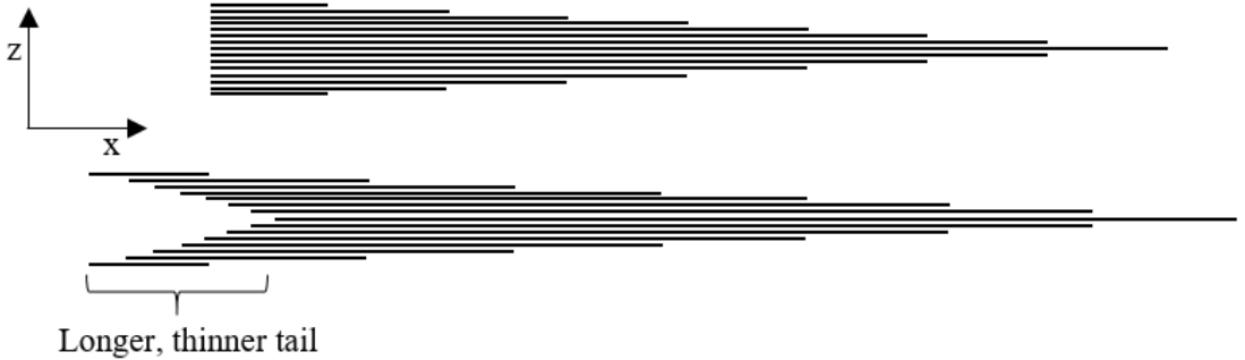


Figure 6: Representation of how the folded layers the membrane quadrant stack with entirely-parallel z-folds (above) and half parallel z-folds or partial fan folds (below) [4].

At this stage of the design, the choices made regarding the folding method is satisfactory. Admittedly, it would be encouraged to perform a detailed study in the future to explore the most optimal folding pattern for stowage efficiency and deployment dynamics.

2.2.3 Wrapping of sail

The base of each quadrant is to be tethered to the core of the spool, equally distributed around its circumference. In interest of applying a safety margin, it will be assumed the entire folded sail has a uniform thickness of the base of 0.1 mm as calculated previously. In reality, the sail will have a variable thickness allowing for more efficient stowage. The thickness around the spool was calculated using Algorithm 1 seen below:

The wrapping thickness for the 28.6 m^2 sail is found to be 62.8 mm. Next, the addition of a tip weight needs to be accounted for to validate the hub sizing choices made.

2.2.4 Tip weight

The tip weight is necessary to pull out the sail due to centrifugal force after spin-up of the sailcraft. The required mass of each tip weight is $m_{tip} = 0.0789 \times m_{sail} = 6.4 \text{ g}$ [6]. The material of this tip weight doesn't need to be durable to space environment in the long term as it is only needed during deployment. Therefore, considering volume and cost efficiency, a dense and

Algorithm 1 An algorithm used to compute wrapping thickness of sail in the spool. Dimensions of all variables are in mm.

```

 $t = 0.1$                                  $\triangleright$  Thickness of folded sail
 $layer = 0$                                 $\triangleright$  Layer count
 $l_{rem} = l_{side}$                           $\triangleright$  Initialise  $l_{rem}$ 
while  $l_{rem} > 0$  do
     $circ = 2\pi(r + layer \times t)$ 
     $l_{rem} = l_{rem} - \frac{circ}{4}$             $\triangleright$  Subtract length of sail wrapped until next layer
     $layer = layer + 1$ 
end while
 $t_{wrap} = 2(layer \times t + r)$   $\triangleright$  Distance between spool core and outer diameter of wrapped sail

```

cheap material should be used, thus best candidate is lead with a density of 11.29 g/cm³ [32]. The best shape for the tip weight in terms of volume would be a sphere, which for 6.4 g has a diameter of 10.27 mm.

The result of the hub sizing analysis provides a remaining distance of 6.93 mm from any sail material to the outer diameter of the flange of the spool. Given the analysis used a large margin of safety factors, it can be assumed that even with imperfect packing, the sail fits comfortably within the spool, validating the choice of a 3U CubeSat for this mission.

2.3 Subsystem sizing

As discussed in Section 1.3, the discovery of a miscalculation in the sizing of the previously proposed corner reflectors lead to the need for active tracking of the sailcraft [24]. There is a need for tracking the sailcraft’s position throughout the mission to verify the progress of the mission and compare with simulations. Since the prospect of active tracking has been added, further instrumentation can be put on board to tracking the sailcraft’s attitude during the journey as well. It follows that for active tracking, there needs to be attitude determination and control subsystem (ADCS); computer and data handling (C&DH); telemetry, tracking, and command (TT&C); and power subsystems. This section aims to provide speculative values for mass and power consumption of these subsystems to be utilised for a power budget (Section 2.3.4.1) and energy capacity balance (Section 2.3.5) to validate the system design. Finally, an overall mass budget of the sailcraft is tabulated in Section 2.4.

During the analysis, the upper limit mass of 3.99 kg was used as the initial mass value for the reference trajectory until the final mass was converged upon in Section 2.4, after which the studies were ran again for verifying the converged values.

2.3.1 Attitude determination and control

Generally, ADCS is used to stabilise the vehicle and orient it in the desired direction during the mission despite external disturbance torques [5]. However, this mission concept relies on a passively stable sailcraft due to spin-stability about the spin-axis (height-wise axis of a 3U CubeSat). Certainly, it is important to consider the effects of imparted disturbance torques about the plane of the sail, and this is investigated in Section 3.

The design of the ADCS for this sailcraft is inspired from IKAROS, whose attitude determination is realised by the Sun and the Earth angle measurements. The Sun angle measurements are acquired from a Spin Sun Aspect Sensor, which provides both the Sun angle and spin rate

of the sailcraft. The Earth angle measurements are acquired from Doppler modulation of the downlink radio frequency (RF) caused by the sailcraft's rotation [18]. A similar approach is taken for this sailcraft, utilising a pair of miniature spinning sun sensors. The relevant details of a product from the market by Redwire Space is detailed in Table 4 as a reference for ADCS [7].

Table 4: Properties of a Miniature Spinning Sun Sensor by Redwire Space [7].

Mass (kg)	Power rating (W)	Field of view (°)	Radiation tolerance (krad)
< 0.25	< 0.5	±87.5	100

2.3.2 Command and data handling

The C&DH subsystem receives commands and distributes to other spacecraft systems, and gathers, processes, and formats spacecraft housekeeping and mission data for downlink or an onboard computer [5]. Given this sailcraft is to operate passively and only relay data, the main function of the C&DH is to store mission data from the ADCS and Power subsystems, timekeeping, computer health monitoring and processing data to be relayed by the TT&C subsystem. The properties of a generic on-board computer for space missions (iOBC by ISISPACE) is tabulated in Table 5 as a reference for the C&DH subsystem [8].

Table 5: Properties of a generic on-board computer (iOBC) for space missions [8].

Mass (kg)	Power rating (W)	Data Storage (kB)
0.1	0.4	512

2.3.3 Telemetry, tracking, and command

The TT&C subsystem provides the interface between the spacecraft and the ground systems. The housekeeping, mission, and system health monitoring data is passed from this system back to the ground station to operators [5]. For this sailcraft, all operations are passive thus there is no need for an uplink communication, just downlink transmission of data. The combination of transmission and timekeeping allows this subsystem to range (determine position of) the satellite.

A prospective component, X-Band patch antenna by EnduroSat, was chosen for its small form factor and mass, making it suitable for CubeSat missions. The relevant properties are tabulated in Table 6 [9].

Table 6: Relevant properties of chosen TT&C subsystem [9].

Mass (g)	RF Output Power (W)	Frequency Range (MHz)	Half Power Beam Width (°)
2.2	≤ 4	8025-8400	74

Link design equation in decibel scale is as seen in Equation (1). The definition of each variable can be found in the nomenclature () [5, 33].

$$\frac{E_b}{N_o} = P_t + L_l + G_t + L_S + L_\alpha + G_r + 228.6 + 10 \log T_s - 10 \log R_{data} \quad (1)$$

where $T_s = 135$ K for downlink, and $L_\alpha = 0.04$ dB, acquired from literature [5].

$$G_t \approx 44.3 - 10 \log(\theta_x \theta_y) = 6.9154 \quad (2)$$

where θ_x and θ_y are the half power beam width found in Table 6.

$$EIRP = P_t + L_l + G_t = 11.9360 \quad (3)$$

where $P_t = 4\text{W} = 6.0206$ from Table 6, and L_l is typically -1 dB.

$$L_S = 147.55 - 20 \log S - 20 \log f \quad (4)$$

where S is the path length, which is the distance between the sail and ground station on Earth, and $f = 8.4$ GHz as the upper end limit of X-band frequencies.

$$G_r = -159.59 + 20 \log D + 20 \log f + 10 \log \eta \quad (5)$$

where $D = 35$ m, and $\eta = 0.663$ [34].

The maximum path length, S , and the data rate, R_{data} , are unknown values. Firstly, to decide the tracking limit by distance, the relationship between S and the time elapsed of the mission was examined in Figure 7. It can be seen that the maximum apoapsis before escape results in a distance of 7 AU and embarks on the escape trajectory after the closest approach at 19 years. Since the maximum lifetime of batteries in a space environment is about 20 years and the fact the sailcraft will be on an uninterrupted path to extrasolar space, there is no need for tracking at this stage of the mission. Therefore, the maximum tracking distance is set to 7 AU due to being the furthest crucial tracking distance before escape, and minimum required satellite lifetime to 19 years.

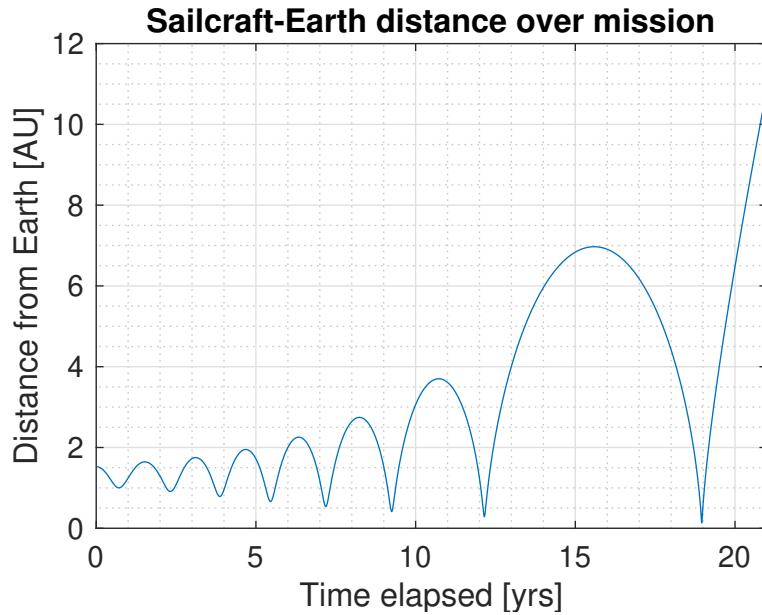
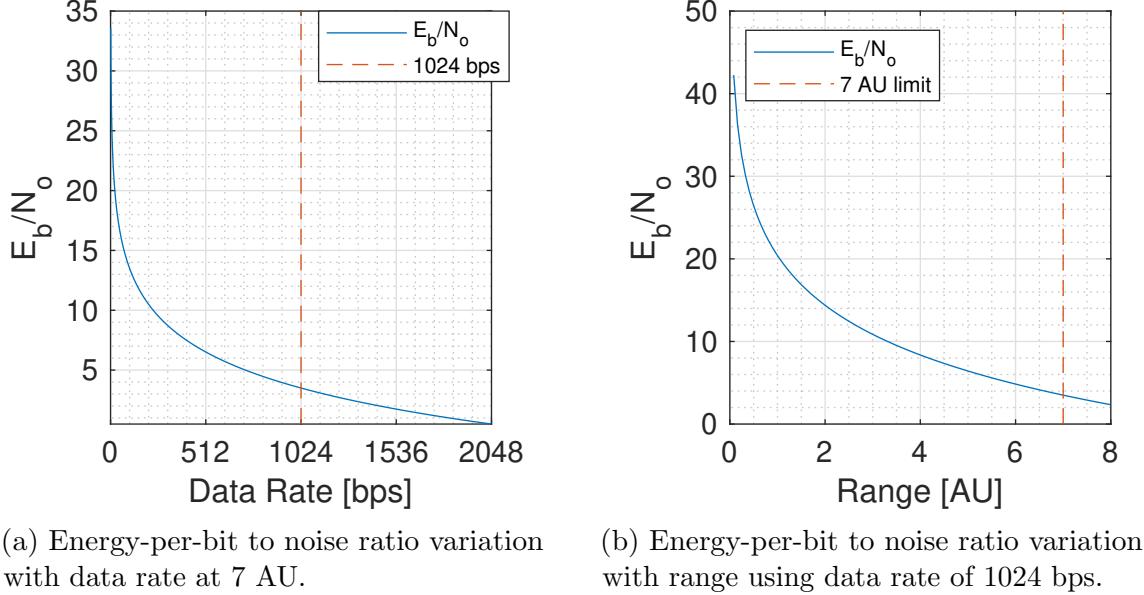


Figure 7: Distance between Earth and Sailcraft over the duration of the active mission.

Utilising this maximum tracking limit as for the path length, S , the link design equation (1) is plotted with a variation of $R_{data} = [0, 2048]$ bps in Figure 8a. Looking at the performance of several modulation and coding schemes in terms of bit error probability (BER) as a function of E_b/N_o , the best performing scheme is BPSK Reed-Solomon Plus R-1/2, K=7 Viterbi decoding [5]. This modulation scheme has a lower limit of just under $E_b/N_o = 3$ dB. Selecting a $R_{data} = 1024$ bps at the worst case distance of 7 AU results in $E_b/N_o = 3.5$ dB, within the

range of the selected modulation scheme. For closer distances, the E_b/N_o only becomes greater as seen in Figure 8b, therefore the selected modulation scheme is validated.

Finally, the selected $R_{data} = 1024$ bps allows for a downlink lasting 30 minutes to transfer 1.758 Mb or 225 kB of data, which is about half the storage capacity of the C&DH. Even accounting for coding schemes that lengthen data to catch erroneous bits, such a large transmission would allow for all mission data of long intervals to be transferred to ground station in this time-frame.



(a) Energy-per-bit to noise ratio variation with data rate at 7 AU.

(b) Energy-per-bit to noise ratio variation with range using data rate of 1024 bps.

Figure 8: Link budget design equation variation to decide optimal values.

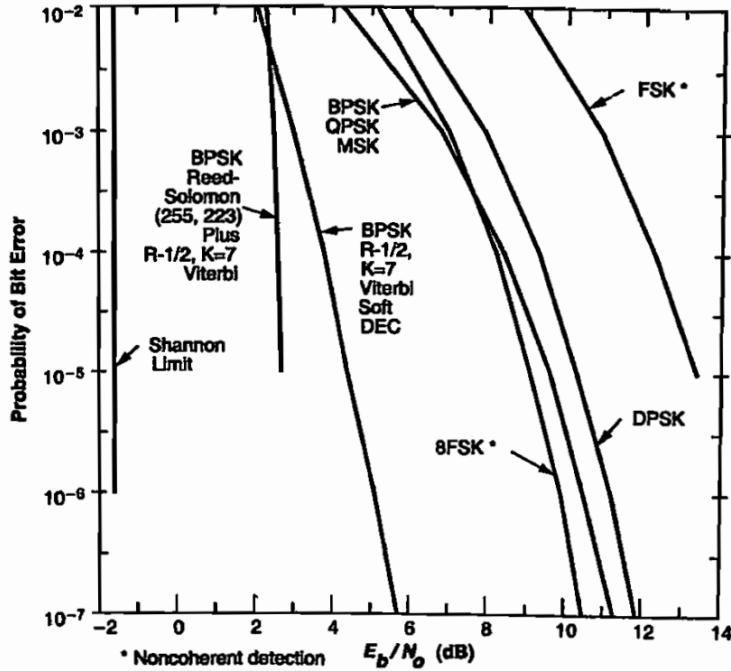


Figure 9: Bit Error Probability (BER) as a function of E_b/N_o [5].

2.3.4 Power

As there has been an implementation of ADCS, C&DH, and TT&C subsystems, there is a necessity for a power supply system. This section will compute the power consumption (Section 2.3.4.1) over the active tracking portion of the mission (20 years) and use it as a guideline to size the required power system. The specific parameters of the power system were sized according to the analysis in Section 2.3.5, validating the power and overall subsystem design of the sailcraft.

The power generation methods considered are solar panels (Section 2.3.4.3) and thin-film solar cells (Section 2.3.4.4), and alternative methods that were considered are discussed in Section 2.3.4.5.

2.3.4.1 Consumption

By compiling the power consumption of all subsystems from Tables 4, 5, 6 into Table 7, a power budget of the system can be realised by applying a margin of 10%. In terms of distribution by each subsystem, that can be seen in Figure 10.

Table 7: Power budget of the sailcraft system.

Component	Power (W)
Propulsion	0
ADC	0.25
CDH	0.4
TTC	4
Sum	4.9
Margin	10%
Total	5.12

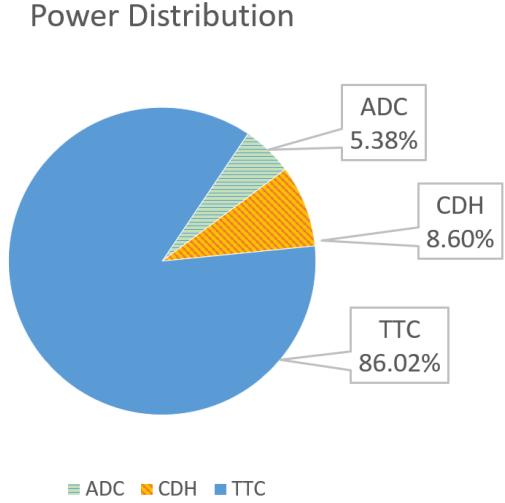


Figure 10: Pie chart showing distribution of power by subsystem.

To compute the power consumption over the duration of the active tracking portion of the mission, which is set to be 20 years, a passive and maximum consumption rate need to be defined. For passive consumption, it's assumed to utilise an ADCS Sun sensor and the C&DH operating on lower power mode. Therefore, passive consumption is approximated to 0.45 W. The power consumption is maximised during the downlink transmission, which is taken from Table 7 as 5.12 W. These consumption parameters are used to vary the power consumption based on daily, weekly, and monthly (every 28 days) tracking rates, which is visualised in Figure 11.

As seen from Figure 11, the total cumulative energy consumption is as follows: Daily - 96.9 kWh, Weekly - 82.2 kWh, and Monthly - 80.7 kWh. Given this mission's tracking lifetime is 20 years, weekly tracking would be sufficient to relay all mission data for monitoring without risk of filling up the storage capacity, letting us discard the use of monthly tracking. Additionally, daily tracking uses 17.9% more energy than weekly tracking, thus will not be used going forward.

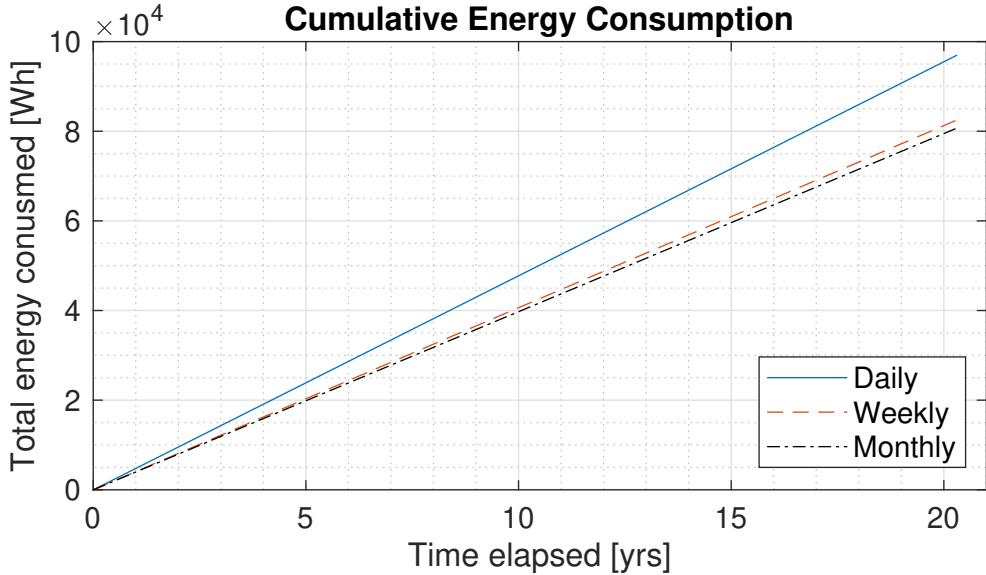


Figure 11: Power consumption of the system with daily, weekly, and monthly tracking rates.

2.3.4.2 Batteries

Utilising the work done by Montete, the ideal choice for energy storage is using Rayovac Li/CF_x cells with a high specific energy of 640 Wh/kg at $0^\circ C$ [24, 35]. However the efficiency of the cells are highly dependant on temperature, and the inclusion of a thermal system for cooling/heating has been neglected due to mass concerns at this design iteration. As such, the specific energy will be taken to be approximately $300 \text{ Wh}\cdot\text{kg}^{-1}$. If these batteries were the only source of power, that would result in a battery mass of 274 kg, well out of the scope of a 3U CubeSat. Consequently, this reinforces the need for power generation sources which are discussed in the following sections. It was concluded that based on analysis done in Section 2.3.5 that the minimum battery mass required is 1 kg (300 Wh capacity). Moreover, according to Section 2.4, the final battery mass in the power subsystem was increased to 1.2 kg (360 Wh capacity) as a safety margin in operations whilst keeping within the 3.99 kg mass limit of a 3U CubeSat.

2.3.4.3 Solar panels

To calculate the power output of solar panels, first the intensity of light at the location position needs to be calculated as shown in Equation (6).

$$I(r_{sail}) = \frac{L_\odot}{4\pi r_{sail}^2} \quad (6)$$

where L_\odot is the solar luminosity equal to $3.828 \times 10^{26} \text{ W}$, and r_{sail} is the position of the satellite relative to the Sun [36].

In terms of material choice, multi-junction solar arrays were chosen for their relative high efficiency (μ) of 22% compared to gallium arsenide (GaAs: 18.5%) and silicon (Si: 14.8%) [5]. While it does have a higher density per area, there is only a specified area which can be utilised for the solar panel placements, the four opening doors on the side of the 3U CubeSat. Furthermore, a degradation rate of 3.75% per year was accounted for, which is used to calculate the lifetime degradation as seen in Equation (7).

$$L_d(t_e) = (1 - degradation)^{\frac{t_e}{year}} \quad (7)$$

This leads to being able to calculate the power generated by the solar panels as seen in Equation (8).

$$P_{gen}(r_{sail}, t_e) = I(r_{sail})\mu_{mj}A_{solar}L_d(t_e) \quad (8)$$

Followingly, the non-operational regions of the orbit must be defined. This is when the solar panels are eclipsed by the sail resulting in the power generation dropping to zero. This can be found by observing the geometry of the sailcraft as shown in Figure 12. Therefore, the angle β is found to be 8.05° . This in combination with the assumption that the sailcraft is deployed perfectly as described in Section 4.1 equivalently means that the solar panels are considered operational between true anomaly range of $98.05^\circ \leq \theta \leq 261.95^\circ$.

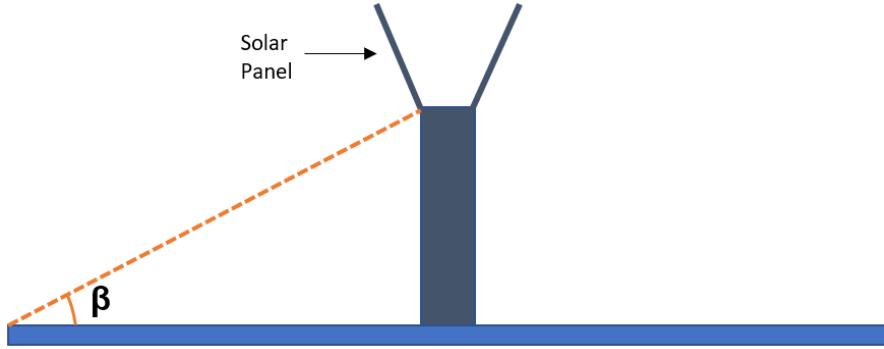


Figure 12: Diagram showcasing the minimum shadow angle from which the entire solar panel is visible to Sun and able to generate power at full capacity. Diagram not to scale.

As the sailcraft is always spinning, not all solar panels will be continually operational. At this stage of analysis, a simplification was made in computation that only one solar panel would be continually operational as if it were perfectly facing the Sun, barring its angle of incidence due to true anomaly. Accordingly, the cumulative energy generated by the solar panels over the duration of the mission is showcased in Figure 14.

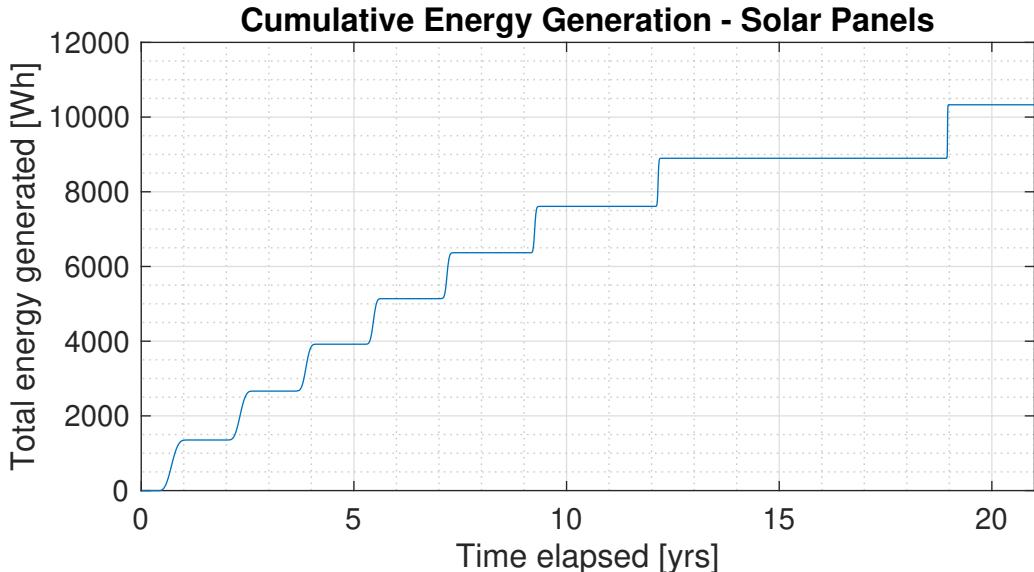


Figure 13: Cumulative power generation of the solar panels over the duration of the active mission.

The cumulative energy generated by the solar panels is 8-10x less than the total power consumption, which showcases that solar panels alone won't be sufficient for the sailcraft to

survive. The additional energy required can be solved by implementing thin-film solar film cells as discussed in the next section. While the solar panels generated a magnitude less of the required energy consumption, even disregarding for the long periods of zero-to-low energy generation, they are in fact a necessary component to the sailcraft's survival. It was found during the analysis in Section 2.3.5 that if there were no solar panels, the battery capacity would deplete after just 1 year of the mission. The solar panels are essential to generate power when the sail surfaces have a very small incidence angle to the Sun, therefore being the only viable source of power. It can be said with certainty that solar panels are vital to the power subsystem.

2.3.4.4 Thin-film solar cells

As explained in previous work, the chosen material for thin-film solar cells is CIGS on PI substrate with a Na post-treatment method which results in a mass per unit area of 41 g/m^2 and efficiency of 14.1% [27, 28].

As the sailcraft is assumed to be deployed perfectly as discussed in Section 4.1, one side will be facing the Sun between $90^\circ < \theta < 270^\circ$, and the other side for the remained. The former will be referred to as the bottom-facing side and the latter as the top-facing side. The bottom-facing side is active on approach towards and away from the rising apoapsis while the top-facing side is active on approach towards and away from the falling periapsis. It follows that the bottom-facing side needs a larger coverage of solar-film cells to generate sufficient power. Conversely, the top-facing side needs a much smaller coverage of solar-film cells. The area of coverage of the thin-film solar cells, which were converged upon during the analysis done in Section 2.3.5, will be used in the following analysis and is as follows:

$$A_{solar} = \begin{cases} 0.1 \text{ m}^2 & 0^\circ < \theta < 90^\circ, 270^\circ < \theta < 360^\circ \\ 14.9704 \text{ m}^2 & 90^\circ < \theta < 270^\circ \end{cases}$$

The calculations for this section follow the same method described in the previous section using equations (6)-(8) but with the following parameters changed: $\mu_{CIGS} = 14\%$, $degradation = 10.4\%$ [28]. With these changes accounted for, the cumulative energy generated by the thin-film solar cell configuration over the duration of the mission can be seen in Figure 14.

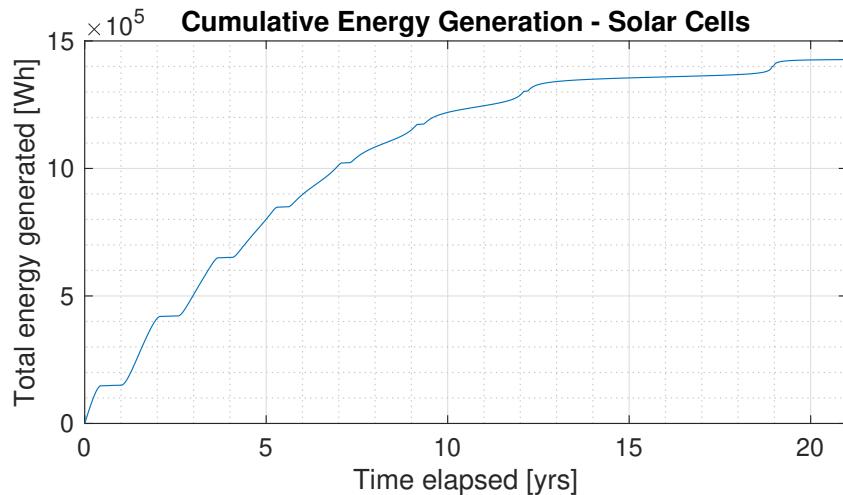


Figure 14: Cumulative power generation of the thin-film solar cells over the duration of the active mission.

It can be seen that the energy generated is almost twice the amount that is consumed in

magnitude, but due to limited capacity to store energy, it isn't all useful. As a result, further analysis needs to be done on the balance of consumption and generation of energy, which is explored in Section 2.3.5 and validates the design of the power subsystem.

It is important to consider that implementing thin-film solar cells has a non-negligible effect on the sail's trajectory due to wrinkling of the membrane causing a deviation of the overall force orientation vector from the ideal normal [37]. Thereby, there is a need for a detailed wrinkling analysis using FEA followed by a trajectory simulation that accounts for the wrinkling effect.

2.3.4.5 Alternative power sources

This section will briefly discuss alternative power sources that can be investigated in a future design stage. The discussed power sources in this section would require detailed studies or a different sailcraft design to accommodate for their implementation.

Typically, spacecrafts that travel far from the Sun lead to solar power being impractical. While in this mission concept there is sufficient power generated by solar power sources due to a decreasing periapsis, there are prolonged dips in energy generation due to distance from the Sun as seen in Figures 13-15. In every interstellar mission, radioisotope thermoelectric generators (RTGs) have been used as a power source [38]. These are nuclear batteries that use thermocouples to convert heat from radioactive decay into electricity by principle of the Peltier-Seebeck effect [39]. Small scale versions of RTGs would be very effective for use in CubeSats as they are mass and volume efficient with a very high specific power and long lifespan, ideal for an extrasolar space mission. Unfortunately, this option has been currently sidelined due to potential safety concerns and political push-back due to potential for environmental harm if an accident were to occur during launch [40].

Another interesting area to explore for an energy source would be utilising a thermoelectric couple to take advantage of the sharp temperature gradient of the sail [5]. This couple is a basic converter that uses the temperature gradient between the p-n junction of individual thermoelectric cells connected in a series-parallel arrangement to generate an electric output under principle of the Peltier-Seebeck effect [39]. The efficiency of such a thermoelectric couple would typically be 5-8% [5]. While this couldn't act as a primary source of energy, considering one side of the sail will face the Sun for a long period, the temperature gradient between the Sun-facing and Sun-eclipsed side can be taken advantage of for some extra power generation. This secondary power source would especially be beneficial when the sail is closer to the sun during the periapsis passes. Further investigation is encouraged for this power source.

2.3.5 Energy capacity balance of overall system

The method to validate the subsystem design of this sailcraft is to simulate the energy capacity balance over the duration of the mission to observe if the system survives till the set target lifetime of 20 years. The sizing of the batteries, solar panels and solar film cells were tweaked during the process of analysing this section until convergence on a working design. The resultant parameters are 1.2 kg of batteries (360 Wh capacity), 14.97 m^2 of thin-film solar cells on the bottom-side of the sail and 0.1 m^2 on the top-side, and four $10 \times 30 \text{ cm}$ panels on the inside of each side door.

The energy generation by both power sources is showcased in Figure 15. It should be noted that while at 19 years elapsed the sail is at the closest position to the Sun, the power

generated is too high thus it is opted to switch off the power generation for 10 days to not overload the system.

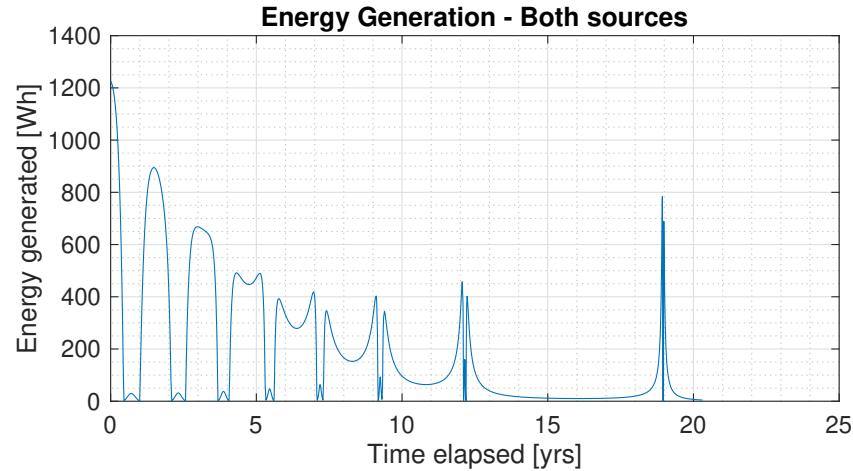


Figure 15: Energy generation of entire system during the mission.

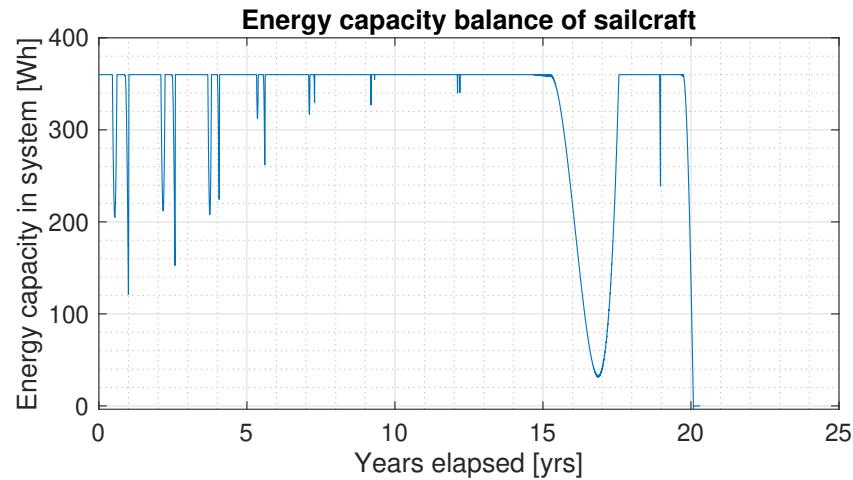


Figure 16: Energy capacity balance over the duration of the active mission.

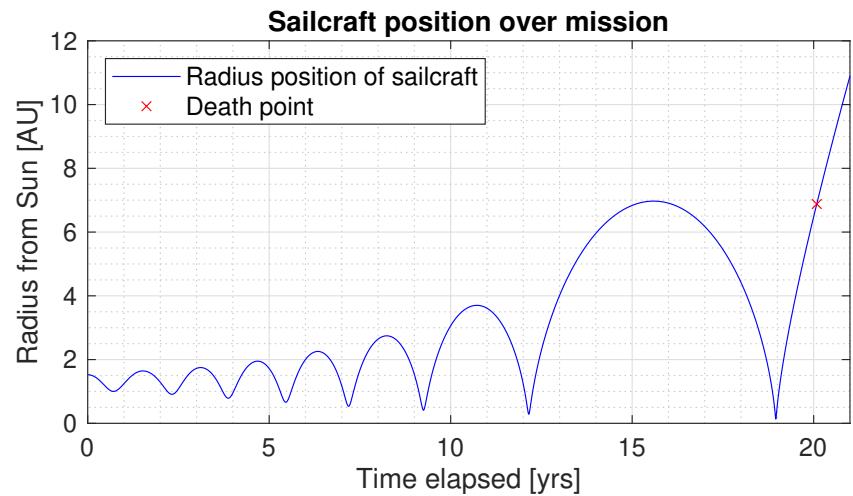


Figure 17: Distance between the sailcraft and Sun with indication of death of the system due to loss of energy capacity over the duration of the active mission.

Next, the energy capacity balance of the sailcraft system is showcased in Figure 16. In early stage of analysis, the minimum battery capacity was found to be 300 Wh to allow the sailcraft to not run out of power during the dip in capacity seen between 15-18 years elapsed. This was raised to 360 Wh (1.2 kg mass) as a safety margin on the power system while keeping within the maximum mass limit of a 3U CubeSat. Before the battery mass was raised, this was also used to size the minimum required areas of thin-film solar cells on each side of the sail. Finally, the range of the sailcraft is plotted against the duration of the mission in Figure 17. Here, the death point at 20.09 years proves the subsystem design results in the sailcraft meeting the target mission lifetime of 20 years successfully, validating the sailcraft design.

2.4 Sailcraft mass budget

The last remaining validation of the sailcraft design to tabulate and verify that the mass of the system does not exceed the limits of a 3U CubeSat. The sailcraft was originally 0.2 kg under the 3.99 kg limit, so this additionally remaining mass was allocated towards the batteries in the power subsystem which was designed to the minimum specification. This increase allowed a higher safety margin for the power subsystem's capacity. This design is validated by the mass budget in Table 8 meeting the mass limit requirement. Finally, the mass distribution of this sailcraft (Figure 18) is unique, with most mass going towards power due to longevity of the active tracking of the mission, and satellite being propelled by a light solar sail makes the propulsion subsystem minuscule relative to common interstellar probes.

Table 8: Mass budget breakdown by each subsystem to give total mass of sailcraft.

Component	Mass (kg)	Percentage of Mass (%)
Structures	0.852	21.36
Power	2.554	64.02
ADCS	0.500	12.53
TT&C	0.002	0.05
Propulsion (Sail)	0.081	2.04
Total	3.989	100

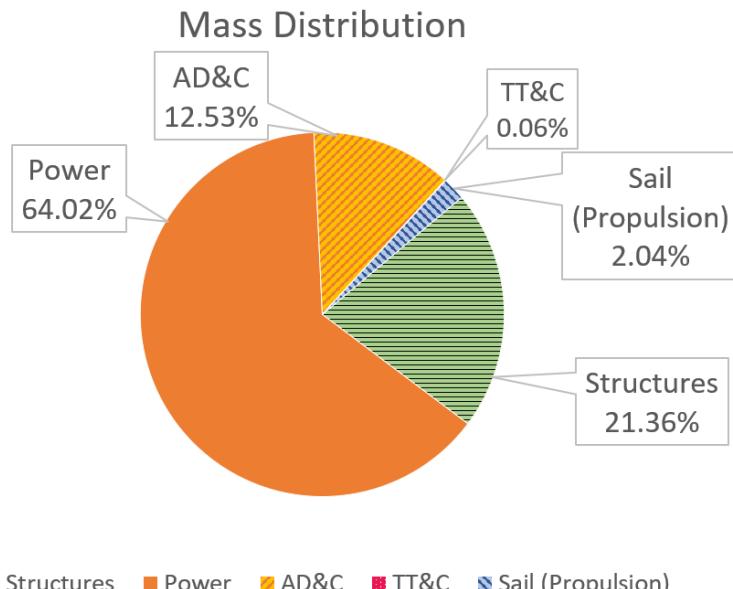


Figure 18: Pie chart showing mass distribution by subsystem.

3 Attitude sensitivity study

Generally, a spacecraft is said to be statically stable when its center of mass lies between the Sun and the center of pressure due to SRP. When a statically stable sailcraft rotates away from the neutral Sun-pointing orientation, a restoring torque is generated, analogous to dynamic behaviour observed of a gravity-gradient-stabilised satellite. However, when the center of pressure lies between the Sun and the center of mass, a destabilizing torque is generated when the sailcraft rotates away from the trimmed orientation [41].

The solar sailcraft of this project operates as a passive gyroscopically stabilised satellite about the spin-axis and so far has been assumed to remain planar in the the Sun inertial reference frame [23]. As per the above discussion, unless the center of mass lies between inside the shape mapped by the center of pressure, a destabilising torque will be imparted. If a consistent and large enough destabilising torque is accumulated on the sailcraft, it will rotate off the ideal orientation.

The aim of this section is to investigate if there is a destabilising torque induced on the passively stable sailcraft and the extent to which it affects the attitude of the sailcraft over the mission. Hereafter, the center of the mass of sail will be referred to as center of gravity (CoG).

3.1 Problem setup

First, the background theory will be explored in Section 3.1.1, followed by defining the calculation of SRP in Section 3.1.1.1. Next, the approach towards the problem will be contextualised in Section 3.1.3 and 3.1.2 then explored in Section 3.1.5. The calculation for the moment of inertia (\mathbf{I}) is done in Section 3.1.4. The proposed approach is then validated in Section 3.1.6 by simulating a scenario where the CoG offset is zero (CoG lies on origin). Following this section, the proposed approach is applied towards practical CoG offset values of the sailcraft for analysis.

3.1.1 Theory

The reference frames mentioned are defined in Section 3.1.3. The derivations in this section are referenced from Dr Jason Forshaw's Advanced Guidance Navigation and Control lecture slides [42]. The angular velocity ($\boldsymbol{\omega}$) is the rotation rate of the body frame of the satellite with respect to a fixed inertial frame and expressed in body frame coordinates. Next the angular momentum (\mathbf{L}) of a satellite in body frame is given in Equation (9)

$$\mathbf{L} = \mathbf{I}\boldsymbol{\omega} \quad (9)$$

where the moment of inertia is defined as seen in (10) and simplified by designing to principal axes.

$$\begin{bmatrix} I_x & -I_{xy} & -I_{xz} \\ -I_{yx} & I_y & -I_{yz} \\ -I_{zx} & -I_{zy} & I_z \end{bmatrix} \approx \begin{bmatrix} I_1 & 0 & 0 \\ 0 & I_2 & 0 \\ 0 & 0 & I_3 \end{bmatrix} \quad (10)$$

For a disturbance-free attitude problem, the total angular momentum of the inertial frame is $\dot{\mathbf{H}} = 0$. $\dot{\mathbf{H}}$ can be linked to the time derivative of momentum in the body frame $\dot{\mathbf{L}}$ as shown in Equation (11). Since the satellite has no reaction wheels, $\mathbf{L} = \mathbf{I}\boldsymbol{\omega}$, thus this value can be substituted into Equation 11 to give Euler's dynamic equation of motion shown in Equation 12.

$$\dot{\mathbf{H}} = \dot{\mathbf{L}} + \boldsymbol{\omega} \times \mathbf{L} = 0 \quad (11)$$

$$\mathbf{I}\dot{\boldsymbol{\omega}} + \boldsymbol{\omega} \times \mathbf{I}\boldsymbol{\omega} = 0 \quad (12)$$

When the satellite is under effect of external torques, such as SRP with a CoG offset, the rate change in total angular momentum is equal to the external torque, \mathbf{T} . This makes Euler's dynamic equation of motion into the form seen in Equation (13).

$$\mathbf{I}\dot{\boldsymbol{\omega}} + \boldsymbol{\omega} \times \mathbf{I}\boldsymbol{\omega} = \mathbf{T} \quad (13)$$

Applying the moment of inertia using principal axes, the Euler's equations can be expressed as done in Equations (14)

$$I_1\dot{\omega}_1 + (I_3 - I_2)\omega_2\omega_3 = T_1 \quad (14a)$$

$$I_2\dot{\omega}_2 + (I_1 - I_3)\omega_1\omega_3 = T_2 \quad (14b)$$

$$I_3\dot{\omega}_3 + (I_2 - I_1)\omega_1\omega_2 = T_3 \quad (14c)$$

By the definition of the body frame given in Section 3.1.3, the principal axes [1,2,3] correspond to [X,Y,Z] of the body frame where Z is the out-of-sail-plane axis, and can be used interchangeably in this analysis. Given this sailcraft is spin-stabilised about the Z-axis, it can be said that $\omega_1 = \omega_2 = 0$. Additionally, due to axisymmetric geometry of the sail and CubeSat configuration, $I_1 = I_2 = I_t$, which is proven later in Section 3.1.4. It's also assumed for the case of this study that $T_3 \approx 0$ due to the small cross-sectional area of the CubeSat imparting a relatively negligible torque. Therefore, since the angular velocities are expected to be negligible early on and along with the above assumptions, the Euler equations can be further simplified to the form seen in Equations (15).

$$I_1\dot{\omega}_1 + (I_3 - I_2)\omega_2\omega_3 = T_1 \rightarrow I_1\dot{\omega}_1 = T_1 \quad (15a)$$

$$I_2\dot{\omega}_2 + (I_1 - I_3)\omega_1\omega_3 = T_2 \rightarrow I_2\dot{\omega}_2 = T_2 \quad (15b)$$

$$I_3\dot{\omega}_3 + (I_2 - I_1)\omega_1\omega_2 = T_3 \rightarrow I_3\dot{\omega}_3 = 0 \quad (15c)$$

Finally, the imparted angular acceleration can be expressed as shown in Equations 16.

$$\dot{\omega}_1 = \frac{T_1}{I_1} \quad (16a)$$

$$\dot{\omega}_2 = \frac{T_2}{I_2} \quad (16b)$$

$$\dot{\omega}_3 = 0 \quad (16c)$$

Next, the torque needs to be calculated, which is a result of the solar radiation pressure.

3.1.1.1 Solar radiation pressure

The force that results in the external torque on the sailcraft comes from the solar radiation pressure that acts on the sail [17]. The pressure imparted by SRP varies by distance from the Sun, which is shown in Equation 17.

$$P_{SRP}(r_{sail}) = \frac{I(r_{sail})}{c} \quad (17)$$

where c is the speed of light.

Unless the sail is perfectly oriented to the Sun and perpendicular to the orbital plane, which would only happen at $\theta = 0^\circ, 180^\circ$, there will be a variation of SRP across the sail as shown in Figure 19.

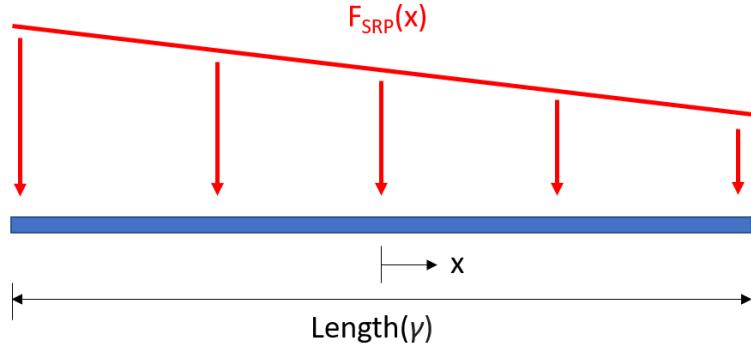
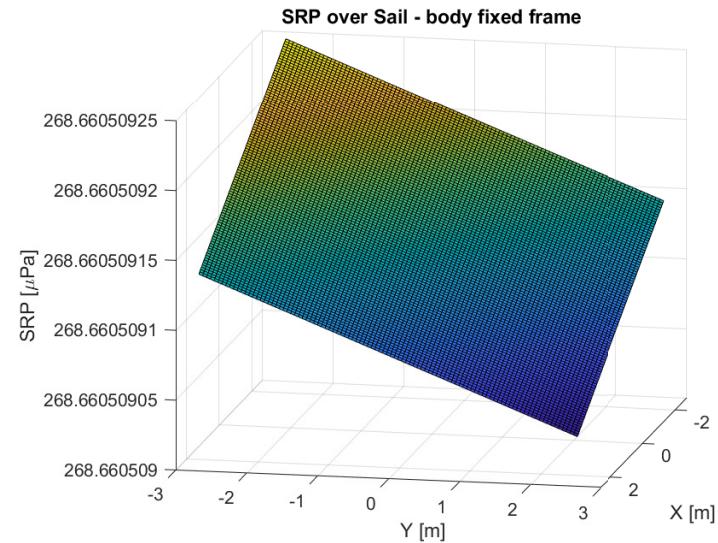
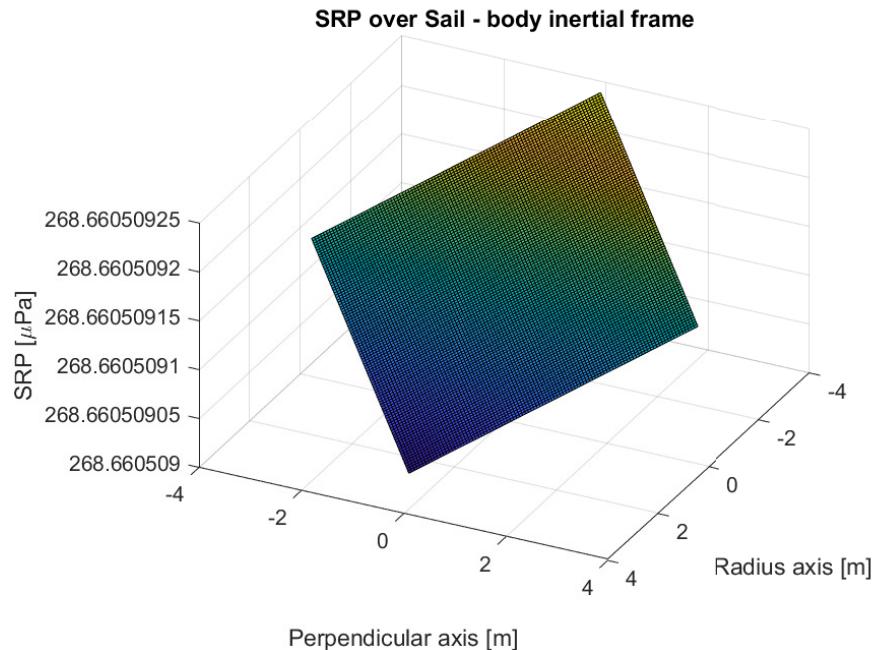


Figure 19: Diagram showcasing the variation of SRP across the sail. The negative region of x is closer to the Sun and positive region is further.



(a) SRP Variation in body fixed frame where orientation is $\gamma = 45^\circ$ and $\alpha = 45^\circ$.



(b) SRP Variation in body inertial frame where orientation is $\gamma = 45^\circ$ and $\alpha = 45^\circ$.

Figure 20: SRP variation over sail in different reference frames at $r = 0.13$ AU.

As the magnitude of distance from the Sun is quite large, SRP is only considered to vary in the radial axis (closest region to furthest region of sail). This can be seen in Figure 20, where in the body inertial frame the SRP remains constant across each value of the radius axis despite change in perpendicular axis. Furthermore, the variation of SRP, even on the μPa scale is minuscule, therefore the resultant center of pressure will be only slightly offset from the origin. The numerical integration of the SRP distribution over the sail at any spin-axis orientation and resultant force is discussed further in Section 3.1.5.

The center of pressure of the SRP can be calculated by calculating the X-axis and Y-axis centroids (1st moment of area) using the expressions in Equations (18).

$$x_{cp} = \frac{\int_{x_{min}}^{x_{max}} x F_{SRP}(x) dx}{F_{SRP,tot}} \quad (18a)$$

$$y_{cp} = \frac{\int_{y_{min}}^{y_{max}} y F_{SRP}(y) dy}{F_{SRP,tot}} \quad (18b)$$

where $F_{SRP,tot}$ is the net force acting on the sail, which can be calculated using Equation (19).

$$F_{SRP,tot} = \int_A F_{SRP} dA \quad (19)$$

Finally, the formula for calculating torque about each principal axis on the sail is expressed in Equations (20).

$$T_x = F_{SRP,tot}(x_{cp} - x_{cg}) \quad (20a)$$

$$T_y = F_{SRP,tot}(y_{cp} - y_{cg}) \quad (20b)$$

These values can be used to calculate the angular acceleration from Euler's dynamic equation of motion in Equations (16). These can then be calculated at every time instance and integrated over time to obtain the resultant angular velocity, and integrated again to obtain the resultant rotational change about each axis.

3.1.2 Assumptions

The following major assumptions are being made in this analysis:

1. Sail is the only body in consideration during attitude calculations.
 - The CubeSat structure is ignored during this analysis as it is only a preliminary approach to determine the effect SRP on the sail's orientation over time.
 - A detailed implementation and attitude study should be conducted at a later stage for a thoroughly designed sailcraft.
2. Perfect deployment as described in Section 4.1.
3. The starting orientation of the sail is perpendicular to the orbital plane and parallel to the radial axis from the Sun, considering an angle of attack (α) to compute the magnitude of force normal to the surface.
 - $\alpha = 0^\circ$ when sailcraft is at $\theta = [90^\circ, 270^\circ]$.
 - $\alpha = 90^\circ$ when sailcraft is at $\theta = [0^\circ, 180^\circ]$.

4. Magnitude of induced angular acceleration by external torques is negligible in short term, resulting in no change in orientation about the principal axes.
- Rather than utilising the direct cosine matrix for rotation, a different approach is taken, described in Section 3.1.3.
 - Change in orientation would also affect trajectory, and current solution isn't an integrated solver thus not a viable implementation. If a divergence of orientation is noted, the mission is considered a failure.

3.1.3 Reference frames

There are three main reference frames to consider: body-fixed frame (axis of [X,Y,Z]), body-inertial frame (axis of [x,y,z], and the Sun-inertial frame (radial axis of r_s). The body-fixed frame coinciding with the body-inertial frame is seen in Figure 21a. The body-inertial frame with a rotated body-fixed frame about the z-axis (spin-stabilised axis) by an angle γ is seen in Figure 21b. Finally, the Sun-inertial frame is seen in Figure 21c with the sailcraft at an angle of attack of α .

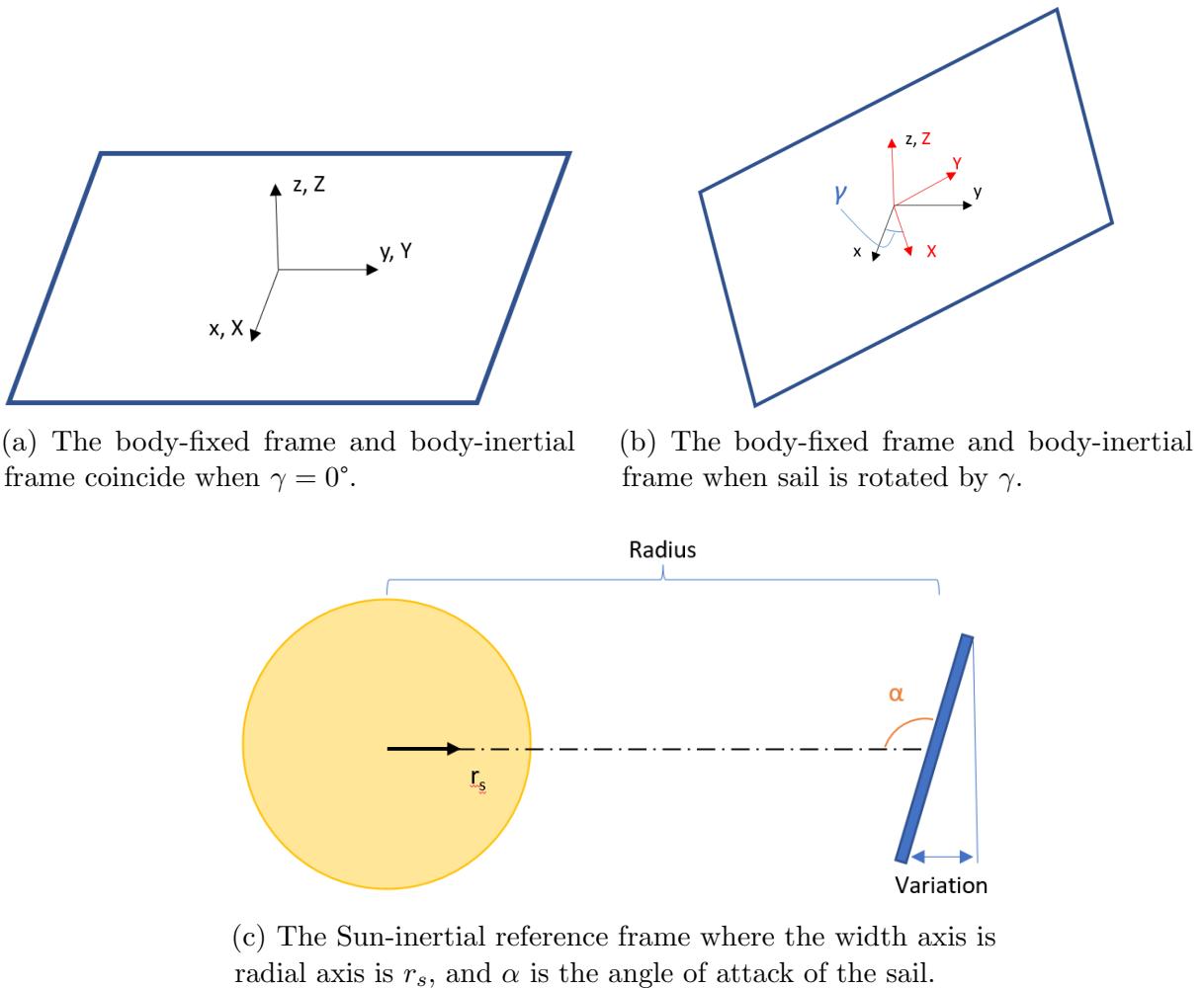


Figure 21: Relevant reference frames in attitude sensitivity study.

3.1.3.1 Rotations to sail

Generally to apply rotations a Direct Cosine Matrix is used, where $\mathbf{R} = \mathbf{R}_3(\gamma)\mathbf{R}_1(\phi)\mathbf{R}_3(\psi)$ [42]. However, as per the 4th assumption in Section 3.1.2, full orientation control is not required.

As such a different rotation order is proposed for numerical implementation, using the 3rd assumption in Section 3.1.2. The computational approach is detailed in Section 3.1.5. A description of the rotations done are as follows:

1. The sail is oriented such that the X-axis is aligned the r_s axis, Y-axis is perpendicular to the orbital plane, and Z-axis is in the direction of motion.
2. The sail is rotated about the Z-axis (spin-axis) by an angle γ , which is dependant on time.
3. The sail is rotated about the y-axis (in inertial frame) to obtain the correct α , which is a function of θ .

3.1.4 Moment of inertia

To calculate the moment of inertia of the sailcraft, an assumption was made to consider the density to be uniform across the CubeSat structure and the sail. The CubeSat structure would combine the mass of the structure, ADCS, TT&C, and power (exc. thin-film solar cells) subsystems. On the other hand, the sail mass consists of the sail itself (propulsion subsystem) and the thin-film solar cells from the power subsystem. The values are tabulated in Table 9 to be utilised in a simple CAD model built in Fusion 360 to compute the moment of inertia about the origin, visualised in Figure 22.

Table 9: The distribution of mass and volume to give an averaged effective uniform density of the CubeSat and the sail to be used for computing \mathbf{I} in Fusion 360.

Body	Mass (kg)	Volume (m^3)	Effective Density ($\text{kg}\cdot\text{m}^{-3}$)
CubeSat	3.290	0.003	1096.713
Sail	0.699	5.72e-5	12222.448

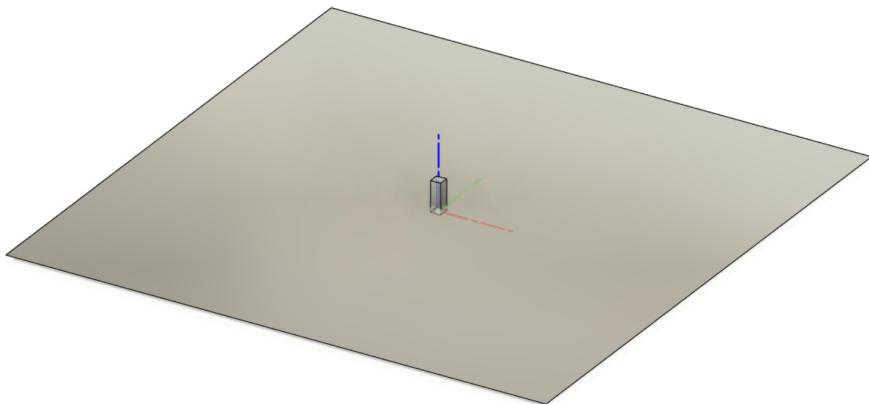


Figure 22: CAD of sailcraft used in calculating the moment of inertia about the origin.

The CAD file provided the moment of inertia about the origin of the CubeSat, $\mathbf{I}_{\text{CubeSat}}$, and the sail, \mathbf{I}_{sail} . These matrices can be summed, $\mathbf{I} = \mathbf{I}_{\text{CubeSat}} + \mathbf{I}_{\text{sail}}$ to give a moment of inertia of the sailcraft of:

$$\begin{bmatrix} 3.3452 & 0 & 0 \\ 0 & 3.3452 & 0 \\ 0 & 0 & 6.6646 \end{bmatrix} 10^{13} \text{ kg} \cdot \text{m}^2 \quad (21)$$

3.1.5 Numerical implementation

The relevant codes utilised in this analysis can be found in Appendix A under the ‘torque on sail’ functions and ‘attitude sensitivity’ scripts. The numerical implementation will be summarised in this section.

First, with the origin placed at the center of the sail, the sail is discretised into equidistant nodes each given an x and y coordinate. The rotation of the sail about the spin-axis is considered and a rotation is applied to the sail coordinates using Equations (22) to obtain the rotated coordinates in the inertial body frame coordinate system [43].

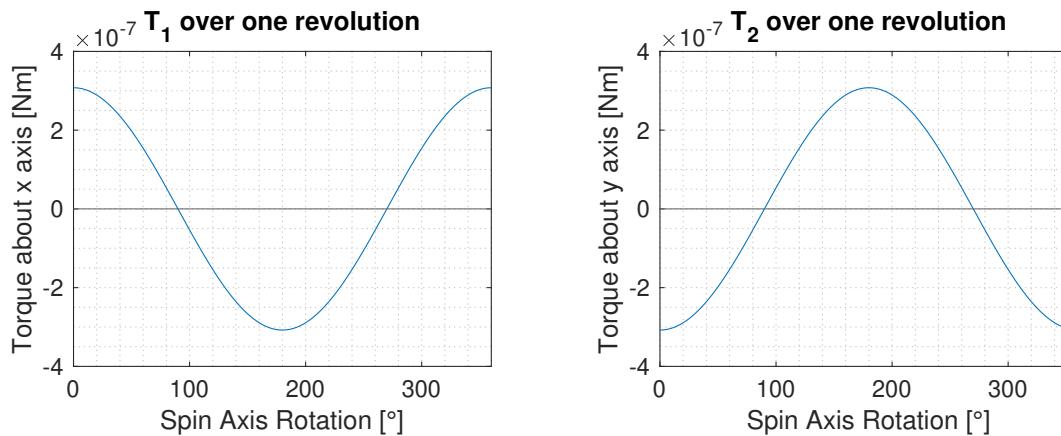
$$x_{rot} = x \cos \gamma - y \sin \gamma \quad (22a)$$

$$y_{rot} = y \cos \gamma + x \sin \gamma \quad (22b)$$

Next, a variable called r_s is computed to give coordinates along the radial axis from the Sun to sail. This is utilised to compute the radial axis variation of SRP across the sail, which can then be mapped onto the sail as has been done in Figure 20. Since the sail was discretised into nodes, they are then averaged to obtain centroids for both coordinates and the SRP between each 4 centroids that represent a square element. The local force is calculated on each element and they are summed to provide the total force, giving the discretised equivalent result of Equation (19). The centroid location of each force is calculated using Equation (18) but with discretised elements, thus using the `trapz` function in MATLAB. Followingly, the torque is calculated using the relation in Equation (20), from which the angular acceleration is obtained used Equation (16). The angular velocity can be computed over a time domain of results using `trapz`, and this can be done again on angular velocity to obtain the resultant rotation.

3.1.6 Method validation

To validate the method and implementation, the scenario where the CoG offset is set to zero, implying the CoG lies perfectly on the origin of the body axes, is investigated. The angular velocity about the spin axis results in a revolution period of 95.24 s, which in the scope of the orbit around the Sun is quite small that the change in position (range from Sun) is negligible. It is expected for the center of pressure to remain equidistant from the origin due to the axisymmetric design of the Sail, passing over the axis to map a circular path. As a result, it would follow that after an spin-axis rotation of 180°, the torque imparted would be equal and opposite, and would follow for every angle. Therefore, the expected result is for the torque about each axis to form a sine or cosine wave.



(a) Torque variation about x-axis over 1 revolution.

(b) Torque variation about y-axis over 1 revolution.

Figure 23: Torque variation about principal axes at $r_{sail} = 0.12$ AU, $\alpha = 90^\circ$.

The above scenario was tested at every instance of $\theta = [0^\circ, 45^\circ, 135^\circ, 180^\circ, 225^\circ, 315^\circ]$, as these would result in $\alpha = 90^\circ$ for $\theta = [0^\circ, 180^\circ]$, and $\alpha = 45^\circ$ for the rest. These were chosen as $\alpha = 90^\circ$ gives the largest magnitude of force while $\alpha = 45^\circ$ gives the largest center of pressure offset. The case with the highest resultant torque was at $\theta = 0^\circ$ during the final closest approach before going on the escape trajectory. While a destabilisation at this instance wouldn't matter, the largest magnitude of forces would showcase if the methodology is robust. As such, the resultant torque variation about both axes can be seen in Figure 23, where the spin-axis orientation was discretised by a step 0.5° over one revolution.

The c_p offset is found at maximum to be about 0.03 mm with a $F_{SRP,tot} = 2.3$ mN, resulting in a maximum torque of 6.9×10^{-7} Nm. Given the angular acceleration would be computed by dividing this torque by the much larger moment of inertia, the immediate effect on the orientation about the X or Y axis is negligible. Now, looking at the trend seen in Figure 23, this fits exactly the desired result, proving that methodology is robust and validating the approach. The resultant net torque acquired by using `trapz` integration over the revolution was on the order of magnitude of 10^{-23} Nm, further verifying the hypothesised result as the orientation effects would be completely negligible. Another simulation was ran over 100 revolutions (approx 2.5 hours), and this showcased resulted in the net torque equalling to 0. In conclusion, this idealised case serves as a strong verification of the methodology proposed for the attitude sensitivity study.

While these results are promising for the performance of the solar sail, they are not possible to achieve in reality. Due to manufacturing limitations and tolerancing, it is impossible to guarantee the CoG to lie within 0.03 mm of the origin as this would require immeasurable precision in production of each part regarding both their mass and dimensions. As a result, a destabilising torque would be imparted about each axis and accumulate over time. This effect will be investigated in Section 3.2.

3.2 Center of gravity offset study

As the methodology has been validated, cases with realistic CoG offset can be investigated. The range of CoG offset that will be investigated are the following: [-10, -5, -1, -0.1, -0.05, 0, 0.05, 0.1, 1, 5, 10] mm. A CoG offset of 0 was considered as a reference value since it was utilised in the validation, not because it is a viable choice. A maximum CoG offset of 10 mm is quite extreme for a 3U CubeSat, being 10% body width away from the origin, but this extreme was considered as a worst-case upper bound. Values of 0.05 and 0.1 mm were added in during a later stage of analysis to find a minimum bound of relatively acceptable results from attitude deviation. An acceptable result will be considered to be around 6° rotation offset. Since the torque accumulates over the mission, the rotation changes more rapidly towards the end. As such, while rotation may have a large effect on the resultant trajectory, if it mostly happens at a late stage of the mission, extrasolar escape may still be achieved.

The torque analysis was conducted on all the offset cases above over the reference case trajectory detailed in Section 4.1. If too large a deviation in orientation occurs, the CoG offset will be considered out of acceptable bounds. The trajectories are analysed until 21 years since the active mission lifetime is just above 20 years as per Section 2.

3.2.1 Results

The results of the analysis are found in the following figures: rotation results are in Figure 24 with end of mission values tabulated in Table 10, angular velocity results are in Figure 25,

and angular acceleration results are in Figure 26.

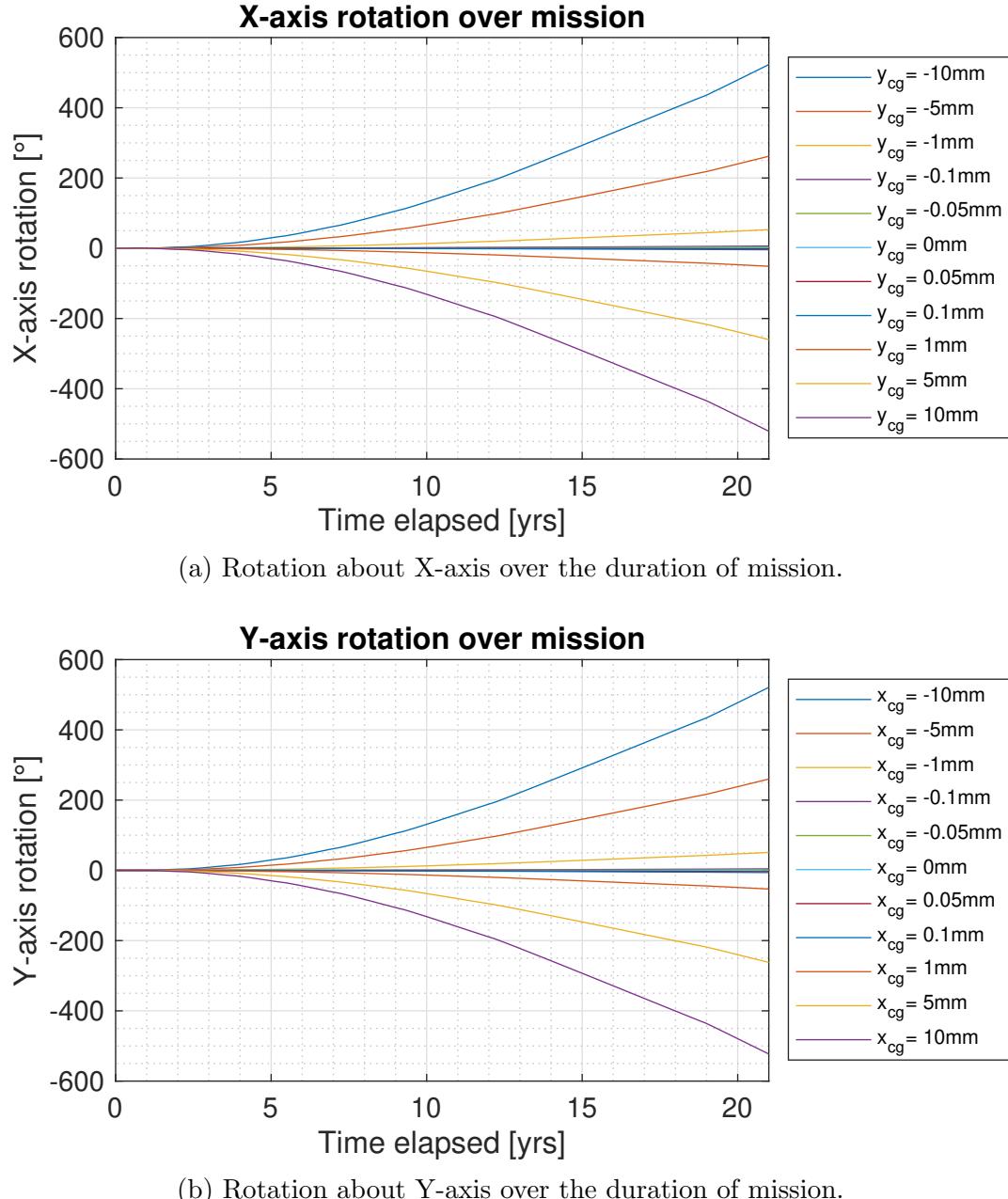
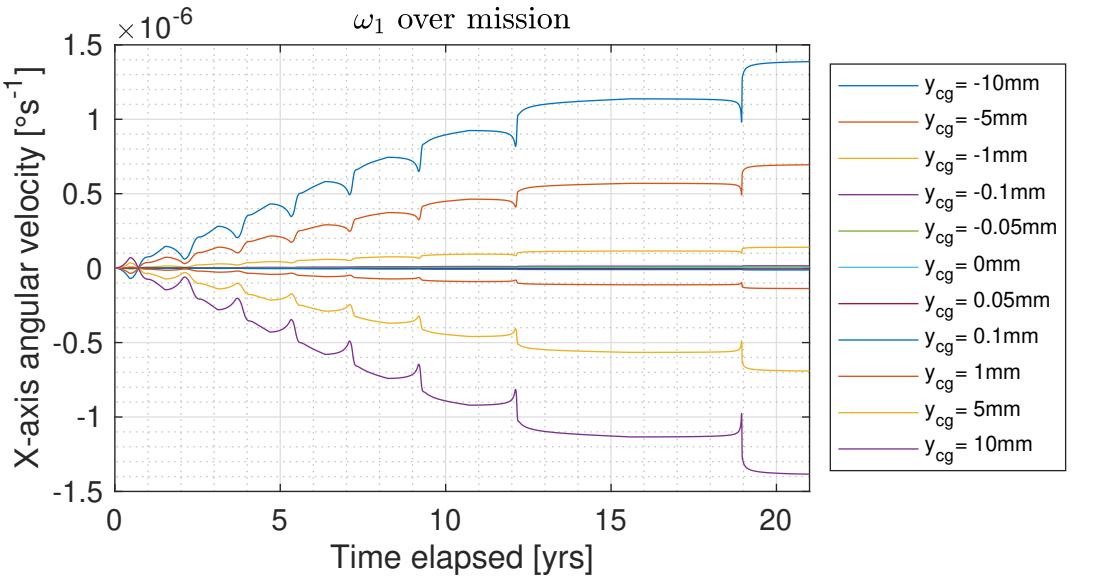


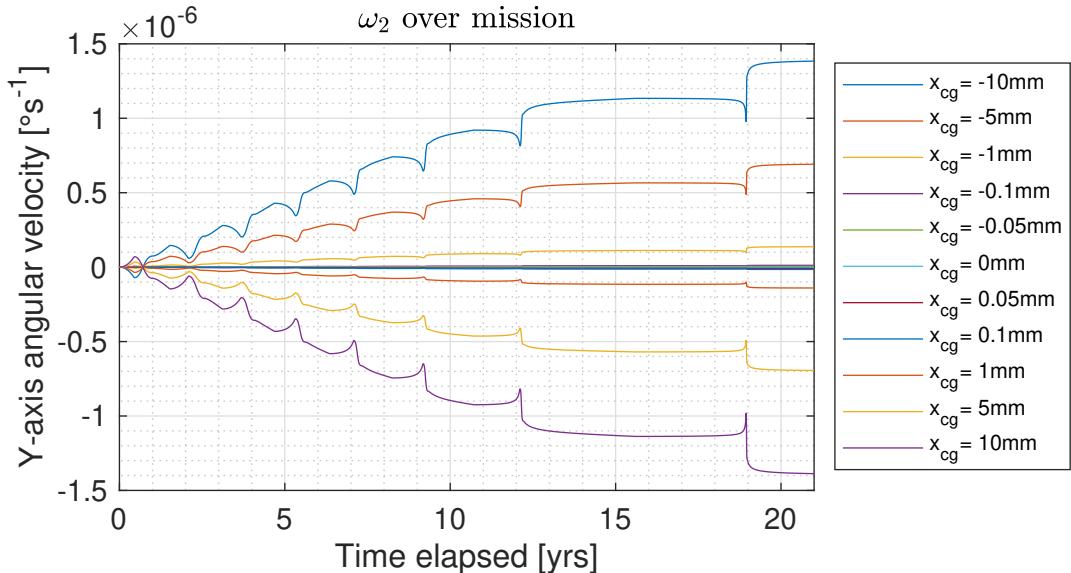
Figure 24: Rotation change over the mission based on varied CoG offset.

Table 10: Rotation change about X and Y axes varied by CoG offset.

CoG offset (mm)	X-axis rotation (°)	Y-axis rotation (°)
-10	520.87	522.82
-5	259.95	261.90
-1	51.21	53.16
-0.1	4.24	6.19
-0.05	1.64	3.58
0	-0.97	0.97
0.05	-3.58	-1.64
0.1	-6.19	-4.24
1	-53.16	-51.21
5	-261.90	-259.95
10	-522.82	-520.87

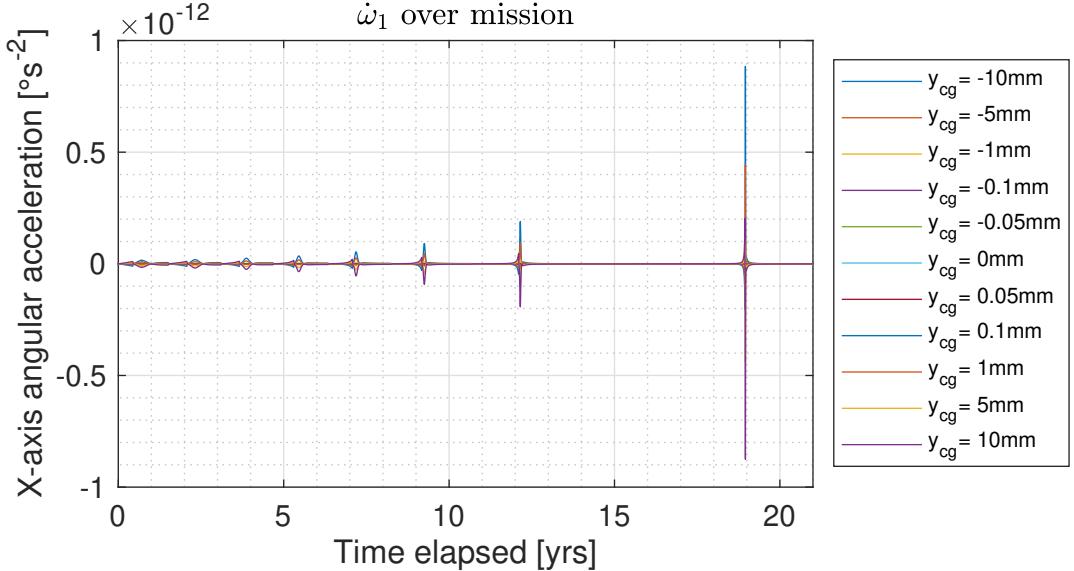


(a) Angular velocity about X-axis over the duration of mission.

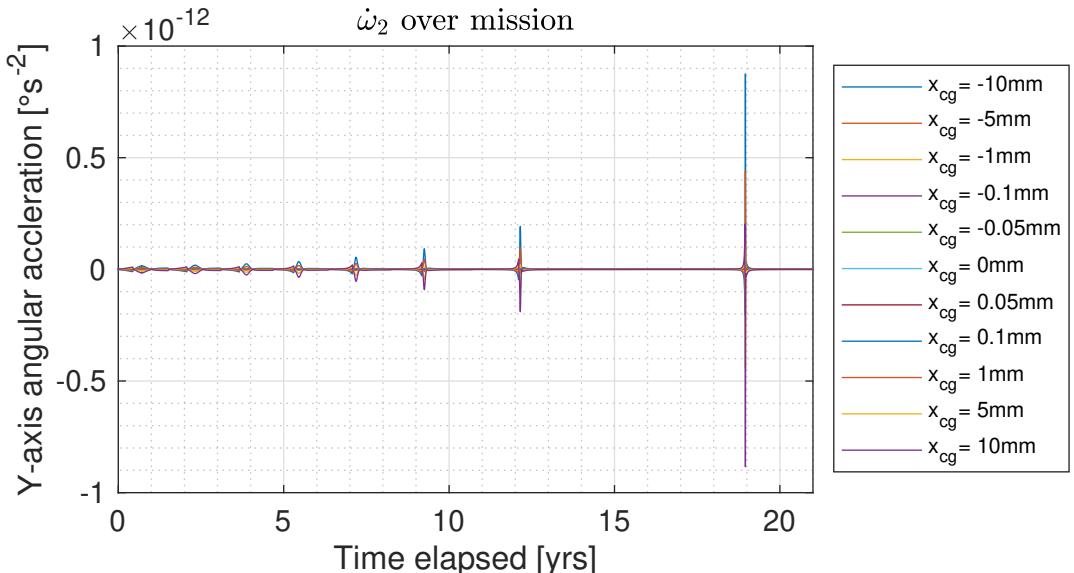


(b) Angular velocity about Y-axis over the duration of mission.

Figure 25: Angular velocity change over the mission based on varied CoG offset.



(a) Angular acceleration about X-axis over the duration of mission.



(b) Angular acceleration about Y-axis over the duration of mission.

Figure 26: Angular acceleration change over the mission based on varied CoG offset.

3.2.2 Discussion

Firstly, the trends in the results should be noted. With negative CoG offset, the resultant torque, angular acceleration, angular velocity and rotation were positive, and vice-versa. It's also interesting to note that while the order of magnitude of results for each CoG offset is same regardless of axis, the specific values appear in flipped order and with a sign change as seen in 10. Next, as seen in Figure 26, the peaks in angular acceleration appear to take place on during the closest approach to the Sun of each pass, increasing in magnitude greatly as the sailcraft gets closer to the Sun. As discussed in Section 3.1.6, the torque about each axis will maintain its sinusoidal pattern, but due to the CoG offset, this wave will not be centered around 0, resulting in an accumulating torque. Consequently, the angular velocity (Figure 25) continually increases with spikes on passbys to the Sun. The relative dips in these figures are likely an artefact of numerical integration where each time step is over 10 days, thus a sharp change in values occur when passing by the periapsis each orbit. Finally, observing the magnitude of change in rotation, once the CoG offset is greater than 0.1 mm, the attitude significantly deviates from the ideal,

resulting in what is considered a mission failure. A closer look at the “acceptable” CoG offset range results are showcased in Figure 27.

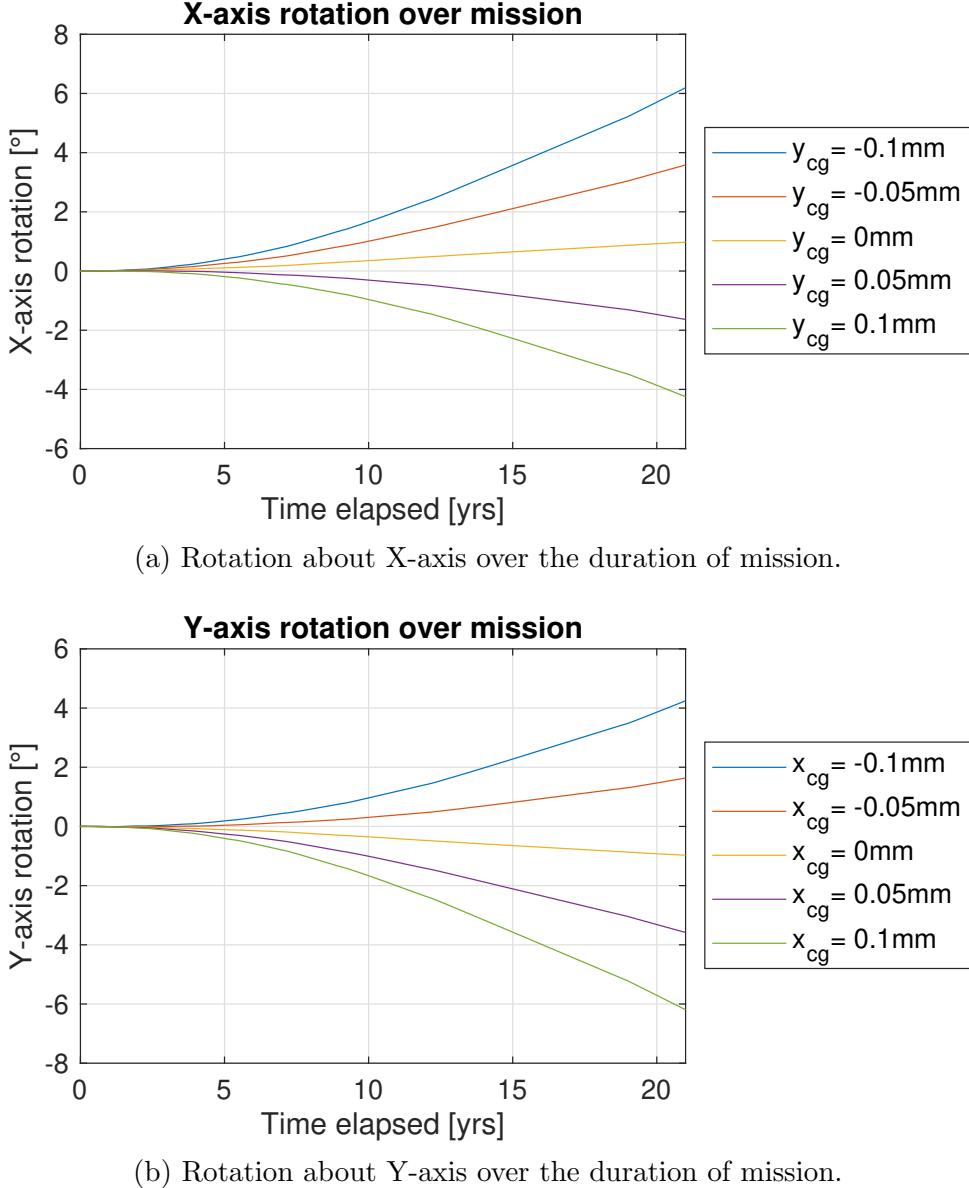


Figure 27: Rotation change over the mission for -0.1 to 0.1 mm CoG offset.

As stated in the assumptions, the sail’s change in attitude isn’t accounted for in this analysis. Thus, the torque and the subsequently processed data are no longer an accurate representation of the real scenario for CoG offsets greater than 0.1 mm. However, a large enough deviation earlier in the mission will reduce the magnitude of force on the sail; change the direction for imparted force; and without any correction — since this is a passively stabilised sailcraft — the trajectory of the orbit is completely altered. Despite this, the current analysis still draws the useful conclusion that CoG offset of greater than 0.1 mm results in too large of a deviation in the rotation of the X and Y axes without active attitude control. For a successful mission, the CoG offset needs to be guaranteed to be within a tolerance of 0.1 mm from the origin, which is highly improbable due to the extreme precision required to guarantee such a tight tolerance. This tolerance is unachievable at a reasonable cost due to requiring extremely precise tolerancing on the mass and dimensions of every component. Therefore, as per the attitude sensitivity study, this iteration of the sailcraft design is considered a failure, leading a requirement of a new sailcraft design to be proposed. This discussion will be explored in Section 4.2.

4 Mission scenario

This section contains the details of the reference case utilised for the studies conducted in Sections 2-3, and discussion regarding the future of the extrasolar sailcraft project.

4.1 Initial reference case

The simulations for the trajectory were ran in the same setup and script as Geragidis' study, with relevant parameters updated from his last work [6]. As per the completed design of the sailcraft in Section 2 and specifically the mass budget analysis in Section 2.4, the mass of the sailcraft used in GMAT trajectory simulations is 3.989 kg. The sailcraft is assumed to have been perfectly deployed at the apoapsis of the Mars transfer orbit with orientation (Euler angles) of $\chi = [0, 0, 0]^T$. A diagram showcasing the orientation over the trajectory can be seen in Figure 28. Next, the angular velocity of the sailcraft is: $\omega = [0, 0, 0]^T$. There is no rotation rate imparted about the spin axis since there is not a detailed model to utilise for the GMAT simulations at the time of this study. Hence, the attitude propagation was investigated in Section 3. Finally, the orbital elements and timing of the initial epoch of the simulation are summarised in 1 in Section 1.3.1.

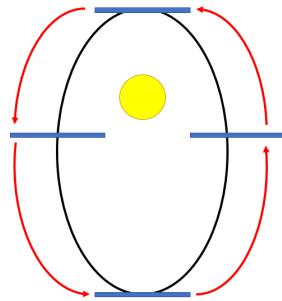


Figure 28: Orientation of sailcraft over one orbit. Trajectory does not account for change in eccentricity due to sail. Diagram not to scale.

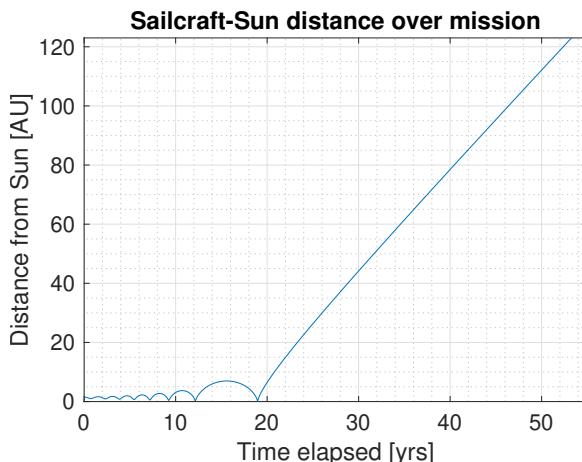


Figure 29: Distance from Sun over the mission.

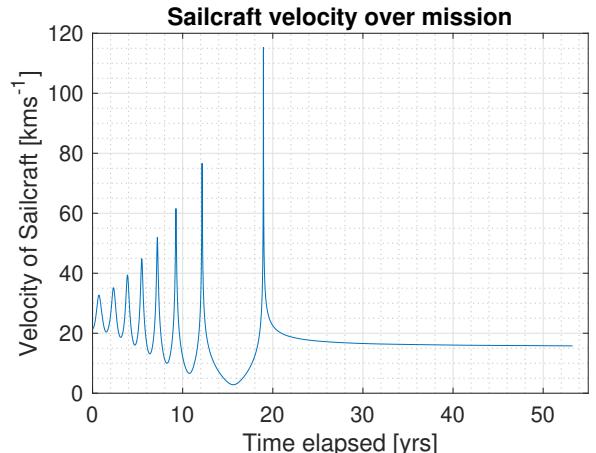


Figure 30: Magnitude of velocity in Sun inertial frame over the mission.

With the aforementioned parameters considered, the GMAT simulation was done to provide the reference case trajectory that was utilised in prior analyses, with relevant plots found in Figure 29-31. Figure 29 showcases the distance of the sailcraft from the Sun until heliopause and figure 30 showcases the magnitude of velocity until heliopause. Finally, figure 31 showcases the variation of temperature with a focus on the active tracking of the mission. This was calculated from the trajectory data by considering the radiation of the Sun, emissivity and orientation of the sailcraft. The sail temperature in kelvin is calculated using Equation (23) [44].

$$T(r_{sail}) = \left[\frac{I(r_{sail})[1 - \rho_r]|\sin \chi_2 \sin \chi_3|}{[2 - \rho_r - \rho_e]k} \right]^{\frac{1}{4}} \quad (23)$$

where $\rho_r = 0.91$ and $\rho_e = 0.16$ are the reflectivity coefficient of the reflective and emissive sides of the sail, χ_2 and χ_3 are the 2nd and 3rd Euler angles in the body inertial frame, and $k = 5.67 \times 10^{-8} \text{ W}\cdot\text{m}^{-2}\cdot\text{K}^{-4}$ is the Stefan-Boltzmann constant. As the temperature stays under 350K throughout the crucial stages of the mission only to decrease thereafter, it validates the choice of Kapton ($T_{g,\text{Kapton}} = 633 - 683 \text{ K}$) for the sail material while still leaving Mylar ($T_{g,\text{Mylar}} = 433 \text{ K}$) as an option.

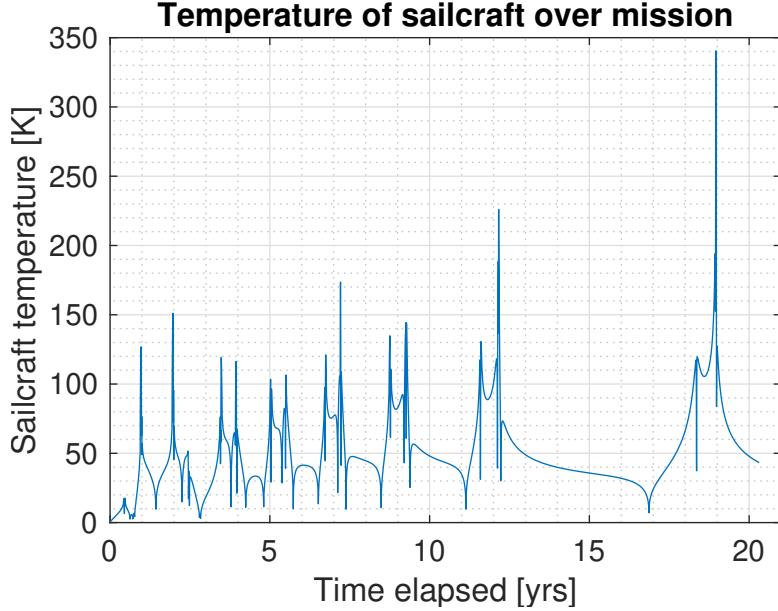


Figure 31: Temperature over the duration of the active mission.

4.2 Suggestions for future of project

The attitude sensitivity study proved that this sailcraft design is unfeasible in terms of maintaining its attitude stability to follow the prescribed trajectory. Therefore, there is a need to reassess the mission concept to allow the goals of the project to be met in the future. The following design suggestions are proposed for a future iteration:

- Implement an active attitude control system
 - Consider utilising LCD lights along the edge of the sail — like in IKAROS — to counter the torque imparted by SRP [18].
 - Consider implementing small thrusters on the CubeSat or at the corners of the sail if considering boom deployment concept.
- Significantly increase the size of the solar sail and sailcraft
 - Increasing sailcraft size allows for implementing further ADCS as suggested above, including additional useful instrumentation to monitor the sailcraft, such as cameras.
 - Increasing the size of the sail will greatly increase the moment of inertia, which could reduce the effect that torque has on the sail over the mission. This requires further investigation.
 - Utilise 12U, 16U or 27U CubeSats (square base dimensions to effectively stow and deploy sail) and conduct a study to determine optimal of CubeSat configuration.

Alongside the above suggestions, an integrated solver of attitude and orbit propagation should be created for detailed analysis of sailcraft trajectories.

The author would like to suggest possible alternate mission objectives for this project. While utilising solar sails for interstellar probes is a novel concept and a heavy engineering challenge, there are planetary bodies in proximity that can be further studied with the use of this sailcraft concept. Such investigations would advance the understanding and capabilities of accurately modelling and tuning sailcraft trajectories. As one of the secondary goals is to advance CubeSat capabilities and attracting funding, this can be achieved by developing a sailcraft in CubeSat specification to cheaply study bodies within the Solar System like the asteroid belt, the moons of gas giants or planets like Mercury, Venus or Mars [45, 20]. For example, using a fleet of sailcrafts to form a satellite constellation around another planet such as Mars, or similarly ambitious mission concepts can be achieved at significantly lower costs than current technology can achieve.

5 Conclusion

This thesis has presented robust methodologies for the sailcraft design and attitude sensitivity analysis that can be utilised well in future studies of the extrasolar sailcraft mission. The sailcraft design was initially validated with an updated subsystem design, successfully implementing active tracking alongside relevant subsystems to build a viable study from the mission. However, the attitude sensitivity found that the currently passive gyroscopically spin-stabilised sailcraft will be destabilised about its X and Y axes due to an inevitable CoG offset as a result of manufacturing and cost limitations. Hence, suggestions were provided towards a new design iteration, including but not limited to: implementing active attitude control systems as part of ADCS, and increasing the size of the sailcraft to larger CubeSat specifications. Alternatively, suggestions were given to expand the scope of the project to investigate planetary missions alongside the extrasolar mission to attract CubeSat funding and initiate developments towards solar sailing CubeSat missions.

5.1 Future work

While it is unfortunate that the findings of this thesis result in the scrapping of this design iteration, hope remains that future works will bring the project closer to materialisation with a more robust and infallible design. The following topics are suggested for future works in order of importance:

- Designing and testing an active attitude control system as part of ADCS
- Updated reference case trajectory with new design iteration
- Construction of an integrated attitude and orbit propagation simulator for accurate trajectory simulations
- Construction of a FEA model of the sail and detailed CAD of sailcraft to import into trajectory simulations
- Revised structural analysis on sail with thin-film solar cells implemented on the sail
- Design of the carrier spacecraft and custom CubeSat deployment system
- Component level design of each subsystems
- Considerations and exploration of alternate mission concepts as previously suggested

References

- [1] Imperial College Space Society. Daedalus Sail Animation. URL <https://imgur.com/a/6zejyKK>.
- [2] Elizabeth Mabrouk. What are SmallSats and CubeSats?, February 26 2015. URL <https://www.nasa.gov/content/what-are-smallsats-and-cubesats>.
- [3] Jiang Zhao, Qiang Tian, and Hai-Yan Hu. Deployment dynamics of a simplified spinning IKAROS solar sail via absolute coordinate based method. *Acta mechanica Sinica*, 29(1):132–142, Mar 7, 2013. URL <https://link.springer.com/article/10.1007/s10409-013-0002-9>.
- [4] Olive R. Stohlman, Juan M. Fernandez, Gregory Dean, Nigel Schneider, Jin Ho Kang, Randall Barfield, Taylor Herndon, and Peter Stokes. *Advances in Low-Cost Manufacturing and Folding of Solar Sail Membranes*. AIAA Scitech 2020 Forum.
- [5] Wiley J. Larson. *Space Mission Analysis and Design*. Kluwer Academic Publ, Dordrecht [u.a.], 3. ed., 6. print. edition, 2004. ISBN 1881883108.
- [6] Filippos V. Geragidis. *Preliminary design of a solar sail propelled spacecraft for extrasolar trajectories*. PhD thesis, 06/01/ 2021.
- [7] Miniature Spinning Sun Sensor ($\pm 87.5^\circ$) , 22/11/ 2022. URL <https://satsearch.co/products/redwirespace-miniature-spinning-sun-sensor>.
- [8] ISIS On Board Computer, 2021. URL <https://www.isispace.nl/product/on-board-computer/>.
- [9] DATASHEET X-Band Single Patch Antenna. Technical report, EnduroSat, 04/03/ 2021. URL <https://www.endurosat.com/cubesat-store/cubesat-antennas/x-band-patch-antenna/>.
- [10] Brook Barnes. In a Breathtaking First, NASA’s Voyager 1 Exits the Solar System, September 12 2013. URL <https://www.nytimes.com/2013/09/13/science/in-a-breathtaking-first-nasa-craft-exits-the-solar-system.html>.
- [11] The Pioneer Missions, March 26 2007. URL <https://www.nasa.gov/centers/ames/missions/archive/pioneer.html>.
- [12] Victoria Gill. Nasa’s Voyager 2 probe ‘leaves the Solar System’, 10 December 2018. URL <https://www.bbc.co.uk/news/science-environment-46502820>.
- [13] Tony Greicius. New Horizons Receives Mission Extension to Kuiper Belt, Dawn to Remain at Ceres, July 1 2016. URL <https://www.nasa.gov/feature/new-horizons-receives-mission-extension-to-kuiper-belt-dawn-to-remain-at-ceres>.
- [14] Alan Stern and Yanping Guo. Where Is the New Horizons Centaur Stage?, 10/28/ 2010. URL <http://pluto.jhuapl.edu/News-Center/News-Article.php?page=20101028>.
- [15] Cost of New Horizons. URL <https://www.planetary.org/space-policy/cost-of-new-horizons>.

- [16] Colin R. McInnes. *Solar sailing: technology, dynamics and mission applications*. Astronomy and Planetary Sciences. Springer, London, 1999. ISBN 185233102X. URL <http://cds.cern.ch/record/1663677>.
- [17] Robert Hewson and Davide Amato. Orbital Mechanics Notes, 2022.
- [18] Yuichi Tsuda, Osamu Mori, Ryu Funase, Hirotaka Sawada, Takayuki Yamamoto, Takanao Saiki, Tatsuya Endo, Katsuhide Yonekura, Hirokazu Hoshino, and Jun'ichiro Kawaguchi. Achievement of IKAROS — Japanese deep space solar sail demonstration mission. *Acta astronautica*, 82(2):183–188, Feb 2013. URL <https://dx.doi.org/10.1016/j.actaastro.2012.03.032>.
- [19] David A. Spencer, Bruce Betts, John M. Bellardo, Alex Diaz, Barbara Plante, and Justin R. Mansell. The LightSail 2 solar sailing technology demonstration. *Advances in space research*, 67(9):2878–2889, May 1, 2021. URL <https://dx.doi.org/10.1016/j.asr.2020.06.029>.
- [20] Malcolm Macdonald. *Solar Sailing : Applications and Technology Advancement*. IntechOpen, Jan 01, 2011. ISBN 9533075511. URL <https://openresearchlibrary.org/viewer/f0dac12b-2ed6-4c12-911d-41e2273e5d7c>.
- [21] NASA’s Voyager 1 Spacecraft Nearing Edge of the Solar System , 14/12/ 2010. URL <https://www.space.com/10482-nasa-voyager-1-spacecraft-nearing-edge-solar-system.html>.
- [22] Exploration — Beyond Our Solar System, 26/06/ 2019. URL https://solarsystem.nasa.gov/solar-system/beyond/exploration/?page=0&per_page=10&order=launch_date+desc%2Ctitle+asc&search=&tags=Beyond+Our+Solar+System&category=33.
- [23] Esme H. Moore. *Feasibility analysis of a solar sail propelled spacecraft for extrasolar trajectories*. PhD thesis, -05 2019.
- [24] Pauline Montete. *Preliminary structural design of a solar sail propelled spacecraft for extrasolar trajectories*. PhD thesis, 09/24/ 2021.
- [25] DuPont Kapton HN - Datasheet. Technical report, DuPont, September 2019. URL <https://www.dupont.com/content/dam/dupont/amer/us/en/products/ei-transformation/documents/DEC-Kapton-HN-datasheet.pdf>.
- [26] Mylar polyester film. Technical report, DuPont Teijin Films. URL https://usa.dupontteijinfilms.com/wp-content/uploads/2017/01/Mylar_Physical_Properties.pdf.
- [27] S. Deb. Recent developments in high-efficiency PV cells. United States, May 22, 2000. URL <https://www.osti.gov/servlets/purl/756334>.
- [28] K. Otte, L. Makhova, A. Braun, and I. Konovalov. Flexible Cu(In,Ga)Se₂ thin-film solar cells for space application. *Thin solid films*, 511-512(Complete):613–622, Jul 26, 2006. URL <https://search.proquest.com/docview/28005009>.
- [29] Poly -San,Cal and Obispo,Luis. Cubesat design specification rev. 14 the cubesat program, cal poly slo cubesat design specification (1u -12u) rev 14 cp-cds-r14.
- [30] Aluminium 7075 Technical Datasheet. Technical report, Smiths Metal Centers, 2022. URL <https://www.smithmetal.com/pdf/aluminium/7xxx/7075.pdf>.

- [31] Koryo Miura. Method of Packaging and Deployment of Large Membranes in Space. *The Institute of Space and Astronautical Science report*, 618:1–9, Dec 1, 1985. URL https://explore.openaire.eu/search/publication?articleId=jairo______:_a2cb4b76a5bf5f2ecaab5ecbff0b7bc0.
- [32] William M. Haynes, David R. Lide, and Thomas J. Bruno. *CRC handbook of chemistry and physics*. CRC Press, Boca Raton, Florida, 2016-2017, 97th edition edition, 2016. ISBN 9781498754286.
- [33] Joseph H. Yuen. *Deep Space Telecommunications Systems Engineering*. Plenum Pr, New York u.a, 1. print. edition, 1983. ISBN 0306414899.
- [34] Jim Taylor. *The Deep Space Network*, pages 15–35. Deep Space Communications. Wiley, Hoboken, NJ, USA, 1 edition, 2016. ISBN 9781119169024. URL <http://ieeexplore.ieee.org/xpl/articleDetails.jsp?reload=true&arnumber=8042853>.
- [35] Frederick C. Krause, John-Paul Jones, Simon C. Jones, Jasmina Pasalic, Keith J. Billings, William C. West, Marshall C. Smart, Ratnakumar V. Bugga, Erik J. Brandon, and Mario Destephen. High Specific Energy Lithium Primary Batteries as Power Sources for Deep Space Exploration. *Journal of the Electrochemical Society*, 165(10):A2312–A2320, Jan 2018. URL <https://iopscience.iop.org/article/10.1149/2.1061810jes>.
- [36] RESOLUTION B1 The IAU Strategic Plan 2010-2020: Astronomy for Development. Technical report, International Astronomical Union (IAU), 2015. URL https://www.iau.org/static/resolutions/IAU2015_English.pdf.
- [37] Giovanni Vulpetti, Davide Apponi, Xiangyuan Zeng, and Christian Circi. Wrinkling analysis of solar-photon sails. *Advances in space research*, 67(9):2669–2687, May 1, 2021. URL <https://dx.doi.org/10.1016/j.asr.2020.07.016>.
- [38] Ethan Siegel. NASA Doesn't Have Enough Nuclear Fuel For Its Deep Space Missions, December 13 2018. URL <https://www.forbes.com/sites/startswithabang/2018/12/13/nasa-doesnt-have-enough-nuclear-fuel-for-its-deep-space-missions/#47a2787d1c18>.
- [39] Kurt H. J. Buschow. *Encyclopedia of materials science and technology*. Pergamon Press, Oxford [u.a.], 2001. ISBN 9780080431529.
- [40] Rashid Alimov. Radioisotope Thermoelectric Generators, April 2005. URL <https://bellona.org/news/nuclear-issues/radioactive-waste-and-spent-nuclear-fuel/2005-04-radioisotope-thermoelectric-generators-2>.
- [41] Bong Wie. Solar Sail Attitude Control and Dynamics, Part 1. *Journal of guidance, control, and dynamics*, 27(4):526–535, Jul 2004. URL <http://arc.aiaa.org/doi/full/10.2514/1.11134>.
- [42] Jason Forshaw. Advanced Guidance Navigation and Control (Lecture 8), 2022.
- [43] George B. Arfken. *Mathematical Methods for Physicists*. Academic Press, Oct 22, 2013. ISBN 0120598205. URL <https://www.vlebooks.com/vleweb/product/openreader?id=none&isbn=9781483277820&uid=none>.
- [44] Bernd Dachwald. *Solar Sail Performance Requirements for Missions to the Outer Solar System and Beyond*. 55th International Astronautical Congress of the International Astronautical Federation, the International Academy of Astronautics, and the International Institute of Space Law.

- [45] Andrew J. Tang and Xiaofeng Wu. An Interplanetary Mission Design of a Solar Sailing CubeSat to Mars. *Journal of physics. Conference series*, 1509(1):12026, Apr 1, 2020. URL <https://iopscience.iop.org/article/10.1088/1742-6596/1509/1/012026>.

A MATLAB functions and scripts

This section will provide all relevant MATLAB scripts and functions utilised in the analyses of this thesis. If you would like to request the zip file of this thesis, please contact me at tanmay.ubgade@gmail.com or Dr Aaron Knoll of the Aeronautics Department at Imperial College London.

A.1 Functions

The following are the functions utilised in the scripts.

A.1.1 Importing GMAT Data

```
1 % Function to import GMAT simulation data, adapted from code made by
2 % Filippou and modified by Tanmay for importing extra data.
3 % Ensure to change the SPAD file in GMAT sim
4 % Tanmay | 220422
5
6 %% Inputs
7
8 %%
9
10 function [te, R, SMA, V, Ax, Ay, Az, E1, E2, E3, E1_dot, E2_dot, E3_dot, X, Y, Z,
chi2, chi3, G, Temp, Tmax, th, Ve, Amax, Rmin, R_earth] = ImportGMATData(i)
11 n = 1;
12 % AU to km
13 AU = 149597870.7; % [km]
14 % Years to days
15 yr = 365.2422; % [days]
16 % Solar constants
17 I0 = 1365.4; % Solar intensity [W/m^2]
18 Cr = [0.91,0.16]; % Reflectivity coefficient vector for the two
19 % Solar gravitational constant [m^3/s^2]
20 mu = 1.32712440018e20;
21 % sail surfaces: [reflective,emissive] [-]
22 SBc = 5.670374419e-8; % Stefan-Boltzmann constant [W/(m^2 K^4)]
23 % Perihelion radius set for simulations [AU]
24 mass = [2.27,3,5,10,1.5]; % initial yaw angle
25 % Generate filenames for each simulation
26 a = ['Sim',num2str(i), 'a.txt']; % Simxa.txt contains data on elapsed time, radial
    distance and SMA from Sun for sim number x
27 b = ['Sim',num2str(i), 'b.txt']; % Simxb.txt contains data on velocity, and 3-axis
    acceleration for sim number x
28 c = ['Sim',num2str(i), 'c.txt']; % Simxc.txt contains data on Euler angles for sim
    number x
29 d = ['Sim',num2str(i), 'd.txt']; % Simxd.txt contains data on Euler angle rates
    for sim number x
30 e = ['Sim',num2str(i), 'e.txt']; % Simxe.txt contains data on X, Y, and Z
    coordinates for sim number x
31 f = ['Sim',num2str(i), 'f.txt']; % Simxf.txt contains data on elapsed time, radial
    distance and SMA from Earth for sim number x
32 g = ['Sim',num2str(i), 'g.txt']; % Simxg.txt contains data on elasped time and
    true anomaly from Sun for sim number x
33
34 % Import data from files
35
36 % Open documents
37 aID = fopen(a,'r');
38 bID = fopen(b,'r');
39 cID = fopen(c,'r');
40 dID = fopen(d,'r');
41 eID = fopen(e,'r');
42 fID = fopen(f,'r');
43 gID = fopen(g,'r');
44
45 % Skip headers
```

```

46 fscanf(aID, '%s %s %s', [3,1]);
47 fscanf(bID, '%s %s %s %s', [4,1]);
48 fscanf(cID, '%s %s %s', [3,1]);
49 fscanf(dID, '%s %s %s', [3,1]);
50 fscanf(eID, '%s %s %s', [3,1]);
51 fscanf(fID, '%s %s %s', [3,1]);
52 fscanf(gID, '%s %s', [2,1]);
53
54 % Store data into file-specific matrices
55 A = transpose(fscanf(aID, '%f %f %f', [3,inf]));
56 B = transpose(fscanf(bID, '%f %f %f %f', [4,inf]));
57 C = transpose(fscanf(cID, '%f %f %f', [3,inf]));
58 D = transpose(fscanf(dID, '%f %f %f', [3,inf]));
59 E = transpose(fscanf(eID, '%f %f %f', [3,inf]));
60 F = transpose(fscanf(fID, '%f %f %f', [3,inf]));
61 G = transpose(fscanf(gID, '%f %f', [2,inf]));
62
63 % Close files
64 fclose(aID);
65 fclose(bID);
66 fclose(cID);
67 fclose(dID);
68 fclose(eID);
69 fclose(fID);
70 fclose(gID);
71
72 te(1:min([length(A),length(B),length(C)]),1) = A(1:min([length(A),length(B),
    length(C)]),1); % elapsed time [yrs] (converted from days)
73 R(1:min([length(A),length(B),length(C)]),1) = A(1:min([length(A),length(B),length(
    C)]),2); % orbital distance [AU] (converted from km)
74 SMA(1:min([length(A),length(B),length(C)]),1) = A(1:min([length(A),length(B),
    length(C)]),3); % Semi-major axis [AU] (converted from km)
75 V(1:min([length(A),length(B),length(C)]),1) = B(1:min([length(A),length(B),length(
    C)]),1); % Velocity [km/s]
76 Ax(1:min([length(A),length(B),length(C)]),1) = B(1:min([length(A),length(B),
    length(C)]),2)*1e3; % Normal acceleration [m/s]
77 Ay(1:min([length(A),length(B),length(C)]),1) = B(1:min([length(A),length(B),
    length(C)]),3)*1e3; % Lateral acceleration [m/s]
78 Az(1:min([length(A),length(B),length(C)]),1) = B(1:min([length(A),length(B),
    length(C)]),4)*1e3; % Vertical acceleration [m/s]
79 E1(1:min([length(A),length(B),length(C)]),1) = C(1:min([length(A),length(B),
    length(C)]),1); % Euler angle 1 [deg]
80 E2(1:min([length(A),length(B),length(C)]),1) = C(1:min([length(A),length(B),
    length(C)]),2); % Euler angle 2 [deg]
81 E3(1:min([length(A),length(B),length(C)]),1) = C(1:min([length(A),length(B),
    length(C)]),3); % Euler angle 3 [deg]
82 E1_dot(1:min([length(A),length(B),length(C)]),1) = D(1:min([length(A),length(B),
    length(C)]),1); % Euler angle rate 1 [deg]
83 E2_dot(1:min([length(A),length(B),length(C)]),1) = D(1:min([length(A),length(B),
    length(C)]),2); % Euler angle rate 2 [deg]
84 E3_dot(1:min([length(A),length(B),length(C)]),1) = D(1:min([length(A),length(B),
    length(C)]),3); % Euler angle rate 3 [deg]
85 X(1:min([length(A),length(B),length(C)]),1) = E(1:min([length(A),length(B),length(
    C)]),1); % X coordinate in Sun-fixed inertial coordinates [km]
86 Y(1:min([length(A),length(B),length(C)]),1) = E(1:min([length(A),length(B),length(
    C)]),2); % Y coordinate in Sun-fixed inertial coordinates [km]
87 Z(1:min([length(A),length(B),length(C)]),1) = E(1:min([length(A),length(B),length(
    C)]),3); % Z coordinate in Sun-fixed inertial coordinates [km]
88 R_earth(1:min([length(A),length(B),length(C)]),1) = A(1:min([length(A),length(B),
    length(C)]),2); % orbital distance [AU] (converted from km)
89 %TA(1:length(G),1) = G(:,2)';
90 % Calculate sun-spacecraft angles and instantaneous spacecraft
91 % temperatures
92 for m = 1:length(X) % for each data point
    chi2(m,1) = atand(Z(m,1)/X(m,1)) + E2(m,1); % inertial plane angle to body
        lateral plane
    chi3(m,1) = atand(Y(m,1)/X(m,1))+E3(m,1); % inertial lateral plane angle to
        body lateral plane
93 % determining whether the sun-facing side is reflective
94 % (front) or emissive (rear)
95 if chi2(m,1) < 90 && chi2(m,1) > -90 % front
    if chi3(m,1) < 90 && chi3(m,1) > -90 % front
        Cri = 1;
    else % rear

```

```

101         Cri = 2;
102     end
103 else % rear
104     if chi3(m,1) < 90 && chi3(m,1) > -90 % front
105         Cri = 2;
106     else % front
107         Cri = 1;
108     end
109 end
110 % emissive equilibrium temperature
111 Temp(m,1) = (((1-Cr(Cri))*I0*abs(sind(chi2(m,1))*sind(chi3(m,1)))))/((2-(Cr(1)
112 +Cr(2)))*SBc*AU*R(m,1)))^0.25;
113
114 % Overall orbit properties
115 Tmax(1,n) = max(Temp); % Maximum instantaneous temperature [K]
116 [qq, pp] = max(A(:,2));
117 th = A(pp,1); % Time to reach heliopause [yrs]
118 V_e = V(end,1); % Exit velocity [km/s]
119 Amax = max([max(abs(Ax)),max(abs(Ay)),max(abs(Az))]); % Max acceleration [m/s^2]
120 Rmin = min(R); % Minimum solar flyby distance [AU]
121 end

```

A.1.2 Solar Intensity

```

1 % Function to calculate value of solar intensity for position relative to
2 % the Sun. Radius from Sun input taken in m.
3 % Code by Tanmay | 220425
4
5 function I = SolarIntensity(r)
6
7 solar_luminosity = 3.828*10^26; % https://en.wikipedia.org/wiki/Solar_luminosity
8
9 I = solar_luminosity./(4*pi*r.^2);
10 end

```

A.1.3 Temperature of Sail

```

1 % Function that calculates temperature of the Sail based on emmisivity and
2 % orientation.
3
4 function T = Temp_sail(r,chi2,chi3,rho_r,rho_e)
5 %% Inputs
6 % r - Radius vector of Sun-Sail positions
7 % Chi2 -
8 % Chi3 -
9 % rho_r - reflectivity of reflective surface
10 % rho_e - reflectivity of emissive surface
11 SBc = 5.670374419e-8; % Stefan-Boltzmann constant
12
13 %% Calculations
14
15 T = ((1-rho_r)*SolarIntensity(r).*abs(sind(chi2).*sind(chi3))/((2-rho_r-rho_e)*SBc))
16 .^(1/4);
16 end

```

A.1.4 Direct Cosine Rotation Matrix

```

1 % Function to provide output of overall rotation matrix (Direction Cosine
2 % Matrix)
3 % Tanmay Ubgade | 220531
4
5 function [R] = RotZYX(phi,gam,psi)
6 %% Inputs
7 % phi - First rotation about Z axis
8 % gam - Second rotation about Y axis

```

```

9 % psi - Third rotation about Z axis
10
11 %% Rotation matrices
12 R3_phi = [cos(phi) sin(phi) 0;
13         -sin(phi) cos(phi) 0;
14             0         0         1];
15
16 R2_gam = [cos(gam) 0         sin(gam);
17             0         1         0;
18         -sin(gam) 0         cos(gam)];
19
20 R3_psi = [cos(psi) sin(psi) 0;
21         -sin(psi) cos(psi) 0;
22             0         0         1];
23
24 R = R3_phi*R2_gam*R3_psi;
25
26 end

```

A.1.5 Moment of Inertia

```

1 % Calculating the moment of interia of the Solar Sailcraft
2 % By Tanmay Ubgade 220529
3
4 function I_tot = MomOfInertia()
5
6 %% Moments of inertia calculated from Fusion 360
7
8 % Values taken about the origin in g/mm^2
9
10 % For Sail
11 Ixx = 3.335E+10;
12 Ixy = 1.138E-05;
13 Ixz = 1.349E-07;
14 Iyx = 1.138E-05;
15 Iyy = 3.335E+10;
16 Iyz = 1.349E-07;
17 Izx = 1.349E-07;
18 Izx = 1.349E-07;
19 Izz = 6.664E+10;
20
21 I_Sail = [Ixx, Ixy, Ixz; Iyx, Iyy, Iyz; Izx, Izx, Izy, Izz];
22 clear Ixx Ixy Ixz Iyx Iyy Iyz Izx Izx Izy Izz
23
24 % For CubeSat
25 Ixx = 1.017E+08;
26 Ixy = 0.00;
27 Ixz = 0.00;
28 Iyx = 0.00;
29 Iyy = 1.017E+08;
30 Iyz = 0.00;
31 Izx = 0.00;
32 Izx = 0.00;
33 Izy = 0.00;
34 Izz = 5.500E+06;
35
36 I_CubeSat = [Ixx, Ixy, Ixz; Iyx, Iyy, Iyz; Izx, Izx, Izy, Izz];
37
38 I_tot = (I_Sail + I_CubeSat)*(1000^2)/1000; % In kg/m^2
39 end

```

A.1.6 Angular acceleration from torque (Euler equations)

```

1 % Function that provides angular acceleration as a result of a specific
2 % torque.
3 % Tanmay Ubgade | 220531
4
5 function [omega_dot] = EulerEq12(T,I_tot)
6 %% Inputs

```

```

7 % Torque vector where T(1) is T1 (x-ax torque) and T(2) is T2 (y-ax torque)
8 % I is the 2nd moment of inertia of the sailcraft, acquired from the
9 % MomofInertia function
10
11 %% Processing
12 I = diag(I_tot);
13 It = I(1);
14 omega_dot = T/It;
15 end

```

A.1.7 Torque on sail (ideal case)

```

1 % Function that provides output for Torque on Sail
2 % Code by Tanmay Ubade | 220530
3
4 function [T] = TorqueOnSailIdeal(sideLength, spinAxRot, alpha, radius)
5 %% Inputs
6 % sideLength - Side length of sail in m
7 % spinAxRot - Anticlockwise rotation about spin axis in degrees
8 % alpha - Angle of Attack of the Sail in degrees
9 % radius - Position relative to the Sun in m
10
11 %% Constants
12 % Speed of Light
13 c = 299792458; % [m/s]
14
15 %% SRP over sail(2D Case)
16
17 % Forming discretised nodes of x and y axis over the sail
18 x = ones(100).*linspace(-sideLength/2,sideLength/2); % x-axis in body frame
19 y = ones(100).*(linspace(sideLength/2,-sideLength/2)'); % y-axis in body frame
20
21 % Rotating coordinates based on orientation of Sail about spin axis
22 x_new = x.*cosd(spinAxRot) - y.*sind(spinAxRot);
23 y_new = y.*cosd(spinAxRot) + x.*sind(spinAxRot);
24 % x and y axis is in same orientation as before, but nodes are mapped to
25 % their new position.
26
27 % Radius axis to get variation of sail at radius point
28 r = ones(100).*linspace(min(x_new,[],'all'),max(x_new,[],'all'))*cosd(alpha); % r in
    Sun-body frame
29 % Convert variation of length to acquire Sun radius coordinates
30 global_coords_r = radius + r;
31
32 % Compute SRP over each element
33 SRP_local = SolarIntensity(global_coords_r)*sind(alpha)./c; % SRP across length of
    sail on each element
34
35 %% Element Sizes
36 elem_size = x(1,2) - x(1,1);
37 elem_area = elem_size^2;
38
39 %% Computing centroids of each element
40 % Need to consider 90 degree case due to way array is formed. else the
41 % resultant centroids and torques become 0. Converting angles within +-45
42 % degrees of 90 and 270 degrees helps account for these cases as properly
43 % travelling across matrix in various directions will prove to be quite
44 % complicated
45
46 x_centroids = zeros(length(x_new)-1);
47 y_centroids = x_centroids;
48
49 for i = 1:length(x_centroids) % rows
50     for j = 1:length(x_centroids) % columns
51         x_centroids(i,j) = mean(x_new(i:i+1,j:j+1), 'all')/4;
52         y_centroids(i,j) = mean(y_new(i:i+1,j:j+1), 'all')/4;
53     end
54 end
55
56 %% Computing SRP on each element
57 SRP_local_elem = zeros(length(x_centroids));

```

```

58 for i = 1:length(x_centroids) % rows
59     for j = 1:length(x_centroids) % columns
60         SRP_local_elem(i,j) = mean(SRP_local(i:i+1,j:j+1), 'all')/4;
61     end
62 end
63 %% Force and force centroid computation
64 force_local = SRP_local_elem.*elem_area;
65 % Total force on sail accounting
66 force_tot = sum(force_local, 'all')*(1+0.6); % 0.6 is the reflectance factor
67
68 X_CEN_force_all = zeros(length(x_centroids),1);
69 Y_CEN_force_all = zeros(length(y_centroids),1);
70
71 for i = 1:length(x_centroids)
72     X_CEN_force_all(i) = trapz(x_centroids(i,:),force_local(i,:))/force_tot;
73     Y_CEN_force_all(i) = trapz(y_centroids(:,i),force_local(:,i))/force_tot;
74 end
75
76 X_CEN_force = mean(X_CEN_force_all);
77 Y_CEN_force = mean(Y_CEN_force_all);
78 X(CG) = 0;
79 Y(CG) = 0;
80
81 %% Torque or Moment on Sail about each axis
82 T1 = force_tot*(X_CEN_force-X(CG));
83 T2 = force_tot*(Y_CEN_force-Y(CG));
84 T = [T1, T2, 0]; % T = [T1, T2, T3], 1 is x, 2 is y, 3 is z (spin-axis)
85 end

```

A.1.8 Torque on sail (CoG offset case)

```

1 % Function that provides output for Torque on Sail
2 % Code by Tanmay Ubgade | 220530
3
4 function [T] = TorqueOnSailOffset(sideLength, spinAxRot, alpha, radius, offset)
5 %% Inputs
6 % sideLength - Side length of sail in m
7 % spinAxRot - Anticlockwise rotation about spin axis in degrees
8 % alpha - Angle of Attack of the Sail in degrees
9 % radius - Position relative to the Sun in m
10
11 %% Constants
12 % Speed of Light
13 c = 299792458; % [m/s]
14
15 %% SRP over sail(2D Case)
16
17 % Forming discretised nodes of x and y axis over the sail
18 x = ones(100).*linspace(-sideLength/2,sideLength/2); % x-axis in body frame
19 y = ones(100).*(linspace(sideLength/2,-sideLength/2)); % y-axis in body frame
20
21 % Rotating coordinates based on orientation of Sail about spin axis
22 x_new = x.*cosd(spinAxRot) - y.*sind(spinAxRot);
23 y_new = y.*cosd(spinAxRot) + x.*sind(spinAxRot);
24 % x and y axis is in same orientation as before, but nodes are mapped to
25 % their new position.
26
27 % Radius axis to get variation of sail at radius point
28 r = ones(100).*linspace(min(x_new,[],'all'),max(x_new,[],'all'))*cosd(alpha); % r in
    Sun-body frame
29 % Convert variation of length to acquire Sun radius coordinates
30 global_coords_r = radius + r;
31
32 % Compute SRP over each element
33 SRP_local = SolarIntensity(global_coords_r)*sind(alpha)./c; % SRP across length of
    sail on each element
34
35 %% Element Sizes
36 elem_size = x(1,2) - x(1,1);
37 elem_area = elem_size^2;
38

```

```

39 %% Computing centroids of each element
40 % Need to consider 90 degree case due to way array is formed. else the
41 % resultant centroids and torques become 0. Converting angles within +-45
42 % degrees of 90 and 270 degrees helps account for these cases as properly
43 % travelling across matrix in various directions will prove to be quite
44 % complicated
45
46 x_centroids = zeros(length(x_new)-1);
47 y_centroids = x_centroids;
48
49 for i = 1:length(x_centroids) % rows
50     for j = 1:length(x_centroids) % columns
51         x_centroids(i,j) = mean(x_new(i:i+1,j:j+1), 'all')/4;
52         y_centroids(i,j) = mean(y_new(i:i+1,j:j+1), 'all')/4;
53     end
54 end
55
56 %% Computing SRP on each element
57 SRP_local_elem = zeros(length(x_centroids));
58 for i = 1:length(x_centroids) % rows
59     for j = 1:length(x_centroids) % columns
60         SRP_local_elem(i,j) = mean(SRP_local(i:i+1,j:j+1), 'all')/4;
61     end
62 end
63 %% Force and force centroid computation
64 force_local = SRP_local_elem.*elem_area;
65 % Total force on sail accounting
66 force_tot = sum(force_local, 'all')*(1+0.6); % 0.6 is the reflectance factor
67
68 X_CEN_force_all = zeros(length(x_centroids),1);
69 Y_CEN_force_all = zeros(length(y_centroids),1);
70
71 for i = 1:length(x_centroids)
72     X_CEN_force_all(i) = trapz(x_centroids(i,:),force_local(i,:))/force_tot;
73     Y_CEN_force_all(i) = trapz(y_centroids(:,i),force_local(:,i))/force_tot;
74 end
75
76 X_CEN_force = mean(X_CEN_force_all);
77 Y_CEN_force = mean(Y_CEN_force_all);
78 X_CG = offset;
79 Y_CG = offset;
80
81 %% Torque or Moment on Sail about each axis
82 T1 = force_tot*(X_CEN_force-X_CG);
83 T2 = force_tot*(Y_CEN_force-Y_CG);
84 T = [T1, T2, 0]; % T = [T1, T2, T3], 1 is x, 2 is y, 3 is z (spin-axis)
85 end

```

A.2 Scripts

The following are the scripts utilised in the analyses of this thesis.

A.2.1 Hub sizing

```

1 % Code to verify that sail fits within CubeSat
2 % by Tanmay Ubgade | 220522
3
4 %% housekeeping
5 clear all
6 clc
7
8 %% Input parameters
9 areas = [7.5,12.2,28.3]; % areas in m^2
10
11 %% CubeSat variables in mm
12 length_CS = 100;
13 width_CS = 100;
14 height_CS = 300;
15 wall_thickness = 2;

```

```

16 corner_thickness = 5;
17 diag_length = sqrt(length_CS^2 + width_CS^2);
18
19 %% Folding values
20
21 sail_side = sqrt(areas); % Length of outmost part of sail in m
22 sail_edge = sqrt(2*sail_side.^2)*1000 - 10; % Edge of each triangle, converted to mm.
    Assuming 5mm spool centre
23
24 % 500 folds would be needed to make the sail thickness into 1mm thickness,
25 % let's assume 50 folds (excessive?) will be made, leading to 0.1mm
26 % thickness of the folded sail triangle
27 % This now is to be wrapped around the spool centre
28
29 %% Wrapping calculations
30 % Following calculations were made assuming that only one quadrant of the
31 % sail is being wrapped
32 %
33 % Spool diam is 9 cm, available area for wrapping is 8 cm.
34 wrap_thickness = [0,0,0];
35 r = 5; % 5 mm, 1 cm diam
36 t = 0.2;
37 for i = 1:3
38     l = sail_edge(i);
39     length_rem = l;
40     layers = 0;
41     while length_rem > 0
42         circ = 2*pi*(r + layers*t);
43         length_rem = length_rem - circ;
44         layers = layers + 1;
45     end
46     wrap_thickness(i) = 2*(layers*t + r);
47 end
48 empty_spool_space = 80 - wrap_thickness;
49 %
50
51 % If wrapping all quadrants at the same time, each quadrant wraps a
52 % quarter-circumference before an additional layer is to be considered
53
54 wrap_thickness = [0,0,0];
55 r = 5; % 5 mm, 1 cm diam of the center of spool
56 t = 0.1; % Of one layer of sail that is to be wrapped
57 for i = 1:3
58     l = sail_edge(i);
59     length_rem = l;
60     layers = 0;
61     while length_rem > 0
62         circ = 2*pi*(r + layers*t);
63         length_rem = length_rem - circ/4;
64         layers = layers + 1;
65     end
66     wrap_thickness(i) = 2*(layers*t + r);
67 end
68 % Spool outer diameter is distance between corners of CubeSat minus an
69 % additional 1 mm at each corner for clearance
70 spool_outer_diam = 90;
71 empty_spool_space = spool_outer_diam - (wrap_thickness + 2*r);
72
73 %% Corner masses
74 mass_ball = 0.0789*[0.069, 0.087, 0.081244];
75 lead_density = 11.29/(1000*1000);
76 volume_ball = mass_ball/lead_density;
77 diam = 2*((0.75*volume_ball/pi).^(1/3));
78 empty_spool_space = empty_spool_space - diam
79
80 %% Structural mass
81 topbot_vol = (wall_thickness+1)*length_CS*width_CS;
82 side_vol = wall_thickness*height_CS*width_CS;
83 support_vol = sqrt(5)*height_CS;
84 tot_vol = topbot_vol*2 + side_vol*4 + support_vol*4;
85 mass_structure = tot_vol*0.00000281 % vol * density in kg/mm^3

```

A.2.2 Power Transmission

```

1 % Script to calculate the total power requirements for sailcraft
2 % Tanmay Ubade 220321
3
4 %% housekeeping
5 clear
6 clc
7 close all
8
9 %% Constants
10
11 k = 1.38*10^-20; % Boltzmann constant
12 T_S = 28.5; % System equivalent noise temperature in K
13 NO = k*T_S; % Noise spectral density
14 c = 299792458; % Speed of light
15 AU = 1495978.70691; % AU to km conversion
16
17 %% Base parameters
18
19 % Signal
20 %lambda = 3.75e-2; % Wavelength in m (X-band is 3.75 - 2.4 cm)
21 freq = 8.4*10^9; % 8.4 GHZ
22 lambda = c/freq;
23 % Distance
24 r = 7*AU*10^3; % distance of sailcraft from Earth
25 A = 4*pi*r^2; % Area of sphere
26
27 % Transmitter
28
29 P_T = 4; % Power transmitted
30 %G_T = 4*pi*A_T/lambda^2; % Gain of transmitting antenna
31 G_T = 6.5*10^4; % Voyager gain at Jupiter
32 A_T = G_T*(lambda^2)/(4*pi);
33
34 % Receiver
35
36 rho = P_T*G_T/A; % Power flux density at receiver
37 mu = 0.9; % Receiving antenna efficiency
38 %A_R = 1; % Area of receiving antenna
39 %P_R = rho*mu*A_R; % Received signal
40 %G_R = 4*pi*A_R/lambda^2; % Receiving antenna gain
41 SNR = 5.48*10^5; %SNR for voyager at 8.4GHZ, SNR = P_R/(L * NO)
42 L = 0.7; % Approximate total loss
43 P_R = SNR*L*NO;
44 A_R = P_R/rho;
45 G_R = 4*pi*A_R/lambda^2;
46
47 RT_ratio = G_T*G_R*(lambda^2)/(4*pi*r)^2;
48 P_TTC = P_R/RT_ratio;
49
50 %% C&DH
51
52 P_CDH = 0.4;
53
54 %% Link Design Equation
55 % EIRP = P + L_l + G_t
56
57 P_db = 10*log10(4);
58 L_l = -1; % in dB
59 G_t = 44.3 - 10*log10(74^2);
60 EIRP = P_db + L_l + G_t;
61
62 % Changing data rate
63 %datarate = 1:1:2048; % in bps
64
65 % Changing range
66 r = linspace(0,8)*AU*1000;
67 datarate = 1024;
68
69
70 L_s = 147.55 - 20*log10(r)-20*log10(freq);
71 L_alp = 4*10^-2; % in dB taken from fig 13.10 in SMAD

```

```

72 G_r = -159.59 + 20*log10(35) + 20*log10(freq) + 10*log(0.663); % DSN antenna radius
    is 35m
73 T_s = 135; % 135 downlink, 614 uplink
74
75
76
77 EbNo = EIRP + L_s + L_alp + G_r + 228.6 - 10*log10(T_s) - 10*log10(datarate);
78
79 %}
80 fig1 = figure(1);
81 hold on
82 plot(datarate,EbNo)
83 plot([1024,1024],[0,45], '--')
84 xlabel('Data Rate [bps]')
85 ylabel('{E_b}/{N_o}')
86 box on
87 grid on
88 grid minor
89 ylim([min(EbNo) 35])
90 xticks([0 512 1024 1024+512 2048])
91 hold off
92
93 fig1.Units = 'inches';
94 fig1.Position(3) = 2.8;
95 fig1.Position(4) = 2.8;
96 set(fig1.Children, 'FontName', 'Arial', 'FontSize', 11);
97 legend('{E_b}/{N_o}', '1024 bps', 'Location', 'Northwest', 'FontSize', 8)
98 %print('LinkDesignDataRate', '-depsc')
99 %}
100
101 %
102 fig1 = figure(1);
103 hold on
104 plot(r/(1000*AU),EbNo)
105 plot([7,7],[0,55], '--')
106 hold off
107 xlabel('Range [AU]')
108 ylabel('{E_b}/{N_o}')
109 xlim([0 8])
110 ylim([0 50])
111 box on
112 grid on
113 grid minor
114 fig1.Units = 'inches';
115 fig1.Position(3) = 2.8;
116 fig1.Position(4) = 2.8;
117 set(fig1.Children, 'FontName', 'Arial', 'FontSize', 11);
118 legend('{E_b}/{N_o}', '7 AU limit', 'Location', 'Northwest', 'FontSize', 8)
119 print('LinkDesignRange', '-depsc')
120
121 %}
122
123
124 % boltzmann constant in dB scale = 228.6
125
126 %% Solar Panel Sizing
127
128 % CubeSat Panel assumptions
129 P_req = 5; % in W
130 P_sa = P_req/0.8;
131
132 % 19-20 years in Space
133
134 % Efficiencies: Si = 14.8%, GaAs = 18.5%, Multijunction = 22%
135 I_Earth = 1367; %W/m^2
136 I_limit = I_Earth/(2^2); % intensity at 8 AU, inverse square law
137
138 P0_SI = 0.148*I_limit;
139 P0_GaAs = 0.185*I_limit;
140 P0_mult = 0.22*I_limit;
141 P0 = [P0_SI, P0_GaAs,P0_mult]
142 P_BOL = P0*0.77*cosd(60); % Inherent degradation = 0.77
143 P_EOL = P_BOL.*([1 - [0.0375, 0.0275, 0.005]].^20);
144

```

```

145 area_sa = 1./(P_EOL./P_sa)
146 mass_sa1 = 0.04*PO

```

A.2.3 Energy capacity balancing

```

1 % Code to compute power generation vs power consumption rates over course
2 % of entire journey
3 % Code by Tanmay | 220424
4
5 %% housekeeping
6
7 clear all
8 close all
9 clc
10
11 %% Constants
12 % AU to km
13 AU = 149597870.7; % [km]
14 % Years to days
15 yr = 365.2422; % [days]
16 % hours in a day
17 day = 24;
18 % Solar constants
19 I0 = 1365.4; % Solar intensity [W/m^2]
20 Cr = [0.91,0.16]; % Reflectivity coefficient vector for the two
21 % Solar gravitational constant [m^3/s^2]
22 mu = 1.32712440018e20;
23 % sail surfaces: [reflective, emissive] [-]
24 SBc = 5.670374419e-8; % Stefan-Boltzmann constant [W/(m^2 K^4)]
25 % Perihelion radius set for simulations [AU]
26 mass = [2.27,3,5,10,1.5]; % initial yaw angle
27
28
29 %% Input data
30
31 % te in Days
32 % R, SMA in km
33 % V in km/s
34 % Acceleration in km/s^2
35 % E and Chi in deg
36 % X Y Z in km
37 % Temp in K
38
39 [te, R, SMA, V, Ax, Ay, Az, E1, E2, E3, E1_dot, E2_dot, E3_dot, X, Y, Z, chi2, chi3,
   G, Temp, Tmax, th, Ve, Amax, Rmin, R_earth] = ImportGMATData(1);
40 TA = G(:,2); % True anomaly in degrees
41
42 te_s = te*24*60*60; % Time elapsed in seconds
43 n = length(te);
44
45 % chi3 to be used for true anomaly
46
47 %% Power Consumption
48 %%%%%%%%%%%%%%%%
49 %%%%%%%%%%%%%%% POWER CONSUMPTION %%%%%%%%%%%%%%%
50 %%%%%%%%%%%%%%%%
51 %% Closest approach timing
52 closest_approach_index = find(R==Rmin);
53 R_closest = R(closest_approach_index);
54 te_closest = te(closest_approach_index);
55
56 %% Max tracking - need to be able to track until
57 R_maxtrack = 8*AU;
58 maxtrack_index = find(R<R_maxtrack, 1, 'last');
59 te_maxtrack = floor(te(maxtrack_index));
60
61 %% Consumption parameters
62 passive_consumption = 0.2+0.25; %Watts
63 % During transmission (over estimating length of transmission to 1h)
64 max_consumption = 5; % Watts
65

```

```

66 % Assuming datarate of 512 bps with transmission time of 30 minutes or 1h
67 % minute giving a file size of 0.9 mb or 1.8 mb
68 link_time_min = 30;
69 link_time_hr = link_time_min/60;
70
71 % Assume voltage of amplifier is 3.3V
72 link_energy_usage = max_consumption*link_time_hr;
73
74 % energy consumption per day
75 link_energy_day = max_consumption*link_time_hr + passive_consumption*(day-
    link_time_hr);
76 passive_energy_day = passive_consumption*day;
77
78 % initialise variables
79 t_tracking = 1:1:te_maxtrack; t_tracking = t_tracking';
80 weekly = 1:7:te_maxtrack;
81 monthly = 1:28:te_maxtrack;
82 energy_con = ones(te_maxtrack,3);
83 for j = 1:3
84     %% Energy consumption based on tracking rates
85     tracking_rate = j; % Pick tracking rate here
86
87     switch tracking_rate
88         % Daily - 1
89         case 1
90             energy_con(:,j) = energy_con(:,j)*link_energy_day;
91         % Weekly - 2
92         case 2
93             energy_con(:,j) = energy_con(:,j)*passive_energy_day;
94             energy_con(weekly,j) = link_energy_day;
95         % Monthly (every 28 days) - 3
96         case 3
97             energy_con(:,j) = energy_con(:,j)*passive_energy_day;
98             energy_con(monthly,j) = link_energy_day;
99     end
100 end
101 %% Cumulative energy consumption
102 energy_con_cum = cumtrapz(t_tracking,energy_con);
103 tot_energy_con = trapz(t_tracking,energy_con);
104 %% Plotting consumption
105 fig1 = figure(1);
106 hold on
107 plot(t_tracking/yr, energy_con_cum(:,1), '-')
108 plot(t_tracking/yr, energy_con_cum(:,2), '--')
109 plot(t_tracking/yr, energy_con_cum(:,3), '-.k')
110 grid on
111 grid minor
112 box on
113 title('Cumulative Energy Consumption')
114 xlabel('Time elapsed [yrs]')
115 ylabel('Total energy consumed [W*hrs]')
116 axis([0 21 0 10^5])
117 hold off
118 legend('Daily', 'Weekly', 'Monthly', 'Location', 'Southeast')
119
120 fig1.Units = 'inches';
121 fig1.Position(3) = 6;
122 fig1.Position(4) = 3;
123 set(fig1.Children, 'FontName', 'Arial', 'FontSize', 11);
124 %print('PowerConsumption', '-depsc')
125
126 %% Only battery power
127 LiCFx_energycdensity = 300; % Wh/kg
128 batteryonly_mass = tot_energy_con/LiCFx_energycdensity; % in kg
129
130 % These values are absurdly high and would require a SmallSat or larger,
131 % and would completely invalidate the use of Solar Sails as the mass would
132 % be too high to impart sufficient impulse to escape the solar system
133 % within the required time constraints.
134
135 %% Baseline battery capacity
136
137 % A battery capacity needs to be decided upon
138 batteryonly_mass = 1; % in kg

```

```

139 battery_capacity = batteryonly_mass*LiCFx_energydensity;
140
141
142 %% Power Generation
143 %%%%%%%%
144 %%%%%%% POWER GENERATION %%%%%%%
145 %%%%%%%
146 %% CubeSat Parameters
147 % Area (3U CubeSat)
148 dim_width = 0.1; % in m
149 dim_height = 0.3; % in m
150 area_sp = dim_width*dim_height*sqrt(2); % of each side panel in m^2
151
152 %% Solar Panels on side of CubeSat only
153
154 % Solar intensity at every calculated point
155 I = SolarIntensity(R*10^3);
156
157 % Will use Multi-Junction Solar Arrays for high efficiency making effective
158 % use of limited area available
159 eff_mj = 0.22;
160
161 % Degradation rate of 3.75%
162 degradation_sp = 3.75/100;
163 L_d_sp = (1 - degradation_sp).^(te/365.2422); % Lifetime degredation
164
165 %Power generated by solar array
166 gen_sp = eff_mj.*I.*area_sp.*L_d_sp;
167
168 % Can generate power between 261.95 -> 98.05 degrees, else turned off
169 solar_on1_sp = TA > 261.95;
170 solar_on2_sp = TA < 98.05;
171 solar_on_sp = solar_on1_sp + solar_on2_sp;
172
173 % When operational
174 gen_operational_sp = gen_sp.*solar_on_sp.*abs(cosd(TA));
175 energy_gen_cum_sp = cumtrapz(te,gen_operational_sp);
176
177 % Use figure 2 to showcase that power generated by solar panels on the
178 % doors is 1-2 orders of magnitude lower than needed for the total power
179 % consumption of the mission. As such, Thin Film Solar Cells must be
180 % utilised.
181
182 fig2 = figure(2);
183 hold on
184 plot(te/yr, energy_gen_cum_sp)
185 grid on
186 grid minor
187 box on
188 title('Cumulative Energy Generation - Solar Panels')
189 xlabel('Time elapsed [yrs]')
190 ylabel('Total energy generated [Wh]')
191 xlim([0 21])
192 hold off
193
194 fig2.Units = 'inches';
195 fig2.Position(3) = 6;
196 fig2.Position(4) = 3;
197 set(fig2.Children, 'FontName', 'Arial', 'FontSize', 11);
198 %print('SolarPanelGen', '-depsc')
199
200 %% Thin film solar cells in addition to solar panels
201
202 % CIGS on PI Substrate considered
203 CIGS_PI_mass_area = 41/1000; %kg/m^2
204
205 quad1 = TA < 90;
206 quad4 = TA > 270;
207 quad14 = quad1 + quad4; % Faces sun from 270-90 deg
208 quad23 = quad14 == 0; % Faces sun from 90-270 deg
209
210
211 area_sc1 = 0.1*quad14; % Faces sun from 270-90 deg
212 area_sc2 = 14.9704*quad23; % Faces sun from 90-270 deg

```

```

213 area_sc = area_sc1 + area_sc2; % m^2
214 mass_sc = (0.1 + 14.9704)*CIGS_PI_mass_area;
215 % Using efficiency of 14.1% considering Na post-treatment method
216 eff_CIGS = 0.14;
217
218 % Degradation rate of 10.4%
219 degradation_sc = 10.4/100;
220 L_d_sc = (1 - degradation_sc).^(te/365.2422); % Lifetime degredation
221
222 %Power generated by solar array
223 gen_sc = eff_CIGS.*I.*area_sc.*L_d_sc;
224
225 % If considering
226 %
227 % Can generate power between 90 -> 270 degrees, else turned off
228 solar_on_sc = TA>=90 & TA <= 270 ;
229
230 % When operational
231 gen_operational_sc = gen_sc.*solar_on_sc.*abs(cosd(TA));
232 energy_gen_cum_sc = cumtrapz(te,gen_operational_sc);
233 %
234 %
235 %
236 % When operational
237 gen_operational_sc = gen_sc.*abs(cosd(TA));
238 energy_gen_cum_sc = cumtrapz(te,gen_operational_sc);
239 %
240
241 fig3 = figure(3);
242 hold on
243 plot(te/yr, energy_gen_cum_sc)
244 grid on
245 grid minor
246 box on
247 title('Cumulative Energy Generation - Solar Cells')
248 xlabel('Time elapsed [yrs]')
249 ylabel('Total energy generated [Wh]')
250 xlim([0 21])
251 hold off
252
253 fig3.Units = 'inches';
254 fig3.Position(3) = 6;
255 fig3.Position(4) = 3;
256 set(fig3.Children, 'FontName', 'Arial', 'FontSize', 11);
257 %print('SolarCellGen', '-depsc')
258
259 %% Energy balance
260
261 % total energy generation
262 tot_energy_gen = gen_operational_sc + gen_operational_sp;
263
264 % 2 kg of battery -> 600 Whrs of capacity stored
265 battery_mass = 1.2; % in kg
266
267 capacity = zeros(te_maxtrack,1);
268 capacity(1) = battery_mass * LiCFx_energydensity;
269
270 % interpolate energy generation values to get values for each day
271 daily_tot_energy_gen = interp1q(te,tot_energy_gen,t_tracking);
272 daily_tot_energy_gen(6919:6929) = 0;
273
274 fig4 = figure(4);
275 hold on
276 plot(t_tracking/yr,daily_tot_energy_gen)
277 title('Energy Generation - Both sources')
278 grid on
279 grid minor
280 box on
281 xlabel('Time elapsed [yrs]')
282 ylabel('Energy generated [Wh]')
283 hold off
284
285 fig4.Units = 'inches';
286 fig4.Position(3) = 6;

```

```

287 fig4.Position(4) = 3;
288 set(fig4.Children, 'FontName', 'Arial', 'FontSize', 11);
289 %print('AllCumGen', '-depsc')
290
291 % Energy balance time!!! Woooohoooooooo!!! please make this end
292 for i = 2:te_maxtrack
293     vari = capacity(i-1) + daily_tot_energy_gen(i) - energy_con(i,2);
294     if vari >= capacity(1)
295         capacity(i) = capacity(1);
296     else
297         capacity(i) = vari;
298     end
299     if capacity(i) <= 0
300         %%print(['Out of energy! Happened on day',i,'.'])
301         index_fail = i;
302         break
303     end
304 end
305
306 fig5 = figure(5);
307 hold on
308 plot(t_tracking/yr, capacity)
309 grid on
310 grid minor
311 box on
312 title('Energy capacity balance of sailcraft')
313 xlabel('Years elapsed [yrs]')
314 ylabel('Energy capacity in system [Wh]')
315 hold off
316
317 fig5.Units = 'inches';
318 fig5.Position(3) = 6;
319 fig5.Position(4) = 3;
320 set(fig5.Children, 'FontName', 'Arial', 'FontSize', 11);
321 %print('EnergyBalance', '-depsc')
322
323 battery_death_radius = interp1q(te,R,t_tracking(index_fail))/AU;
324
325 fig6 = figure(6);
326 hold on
327 plot(te/yr,R/AU, '-b')
328 plot(t_tracking(index_fail)/yr,battery_death_radius, 'xr')
329 grid on
330 grid minor
331 box on
332 title('Sailcraft position over mission')
333 xlabel('Time elapsed [yrs]')
334 ylabel('Radius from Sun [AU]')
335 legend('Radius position of sailcraft', 'Death point', 'Location', 'Northwest')
336 xlim([0 21])
337 hold off
338
339 fig6.Units = 'inches';
340 fig6.Position(3) = 6;
341 fig6.Position(4) = 3;
342 set(fig6.Children, 'FontName', 'Arial', 'FontSize', 11);
343 %print('DeathPoint', '-depsc')
344
345 %% Temperature balance
346 rho_r = 0.91; % Reflectivity of reflective surface
347 rho_e = 0.16; % Reflectivity of emissive surface
348 SBc = 5.670374419e-8; % Stefan boltzmann constant
349 R_tracking = interp1q(te,R,t_tracking);
350 chi2_daily = interp1q(te,chi2,t_tracking);
351 chi3_daily = interp1q(te,chi3,t_tracking);
352 Temp_sail_daily = Temp_sail(R_tracking*1000,chi2_daily,chi3_daily,rho_r,rho_e);
353
354
355 fig7 = figure(7);
356 hold on
357 plot(t_tracking/yr,Temp_sail_daily, '-b')
358 grid on
359 grid minor
360 box on

```

```

361 title('Temperature of sailcraft over mission')
362 xlabel('Time elapsed [yrs]')
363 ylabel('Sailcraft temperature [K]')
364 xlim([0 21])
365 hold off
366
367 fig7.Units = 'inches';
368 fig7.Position(3) = 6;
369 fig7.Position(4) = 3;
370 set(fig7.Children, 'FontName', 'Arial', 'FontSize', 11);
371 %print('TempTime', '-depsc')

```

A.2.4 Earth-Sail graphs

```

1 % Code to generate radius from Earth Graph
2 %% housekeeping
3
4 clear all
5 close all
6 clc
7
8 %% Constants
9 % AU to km
10 AU = 149597870.7; % [km]
11 % Years to days
12 yr = 365.2422; % [days]
13 % hours in a day
14 day = 24;
15 % Solar constants
16 I0 = 1365.4; % Solar intensity [W/m^2]
17 Cr = [0.91,0.16]; % Reflectivity coefficient vector for the two
18 % Solar gravitational constant [m^3/s^2]
19 mu = 1.32712440018e20;
20 % sail surfaces: [reflective, emissive] [-]
21 SBc = 5.670374419e-8; % Stefan-Boltzmann constant [W/(m^2 K^4)]
22 % Perihelion radius set for simulations [AU]
23 mass = [2.27,3,5,10,1.5]; % initial yaw angle
24
25 %% Import data
26 [te, R, SMA, V, Ax, Ay, Az, E1, E2, E3, E1_dot, E2_dot, E3_dot, X, Y, Z, chi2, chi3,
   G, Temp, Tmax, th, Ve, Amax, Rmin, R_earth] = ImportGMATData(1);
27
28 %% Calculating temperature
29
30 % Max tracking - need to be able to track until
31 R_maxtrack = 8*AU;
32 maxtrack_index = find(R < R_maxtrack, 1, 'last');
33 te_maxtrack = floor(te(maxtrack_index));
34 t_tracking = 1:1:te_maxtrack; t_tracking = t_tracking';
35
36 rho_r = 0.91; % Reflectivity of reflective surface
37 rho_e = 0.16; % Reflectivity of emissive surface
38 SBc = 5.670374419e-8; % Stefan boltzmann constant
39 R_tracking = interp1q(te,R,t_tracking);
40 chi2_daily = interp1q(te,chi2,t_tracking);
41 chi3_daily = interp1q(te,chi3,t_tracking);
42 Temp_sail_daily = Temp_sail(R_tracking*1000,chi2_daily,chi3_daily,rho_r,rho_e);
43
44
45 %% Plotting
46
47 limit_21yr = length(find((te < 21*yr)));
48
49 fig1 = figure(1);
50 plot(te/yr, R_earth/AU)
51 title('Sailcraft-Earth distance over mission')
52 xlabel('Time elapsed [yrs]')
53 xlim([0 te(limit_21yr)/yr])
54 ylabel('Distance from Earth [AU]')
55 box on
56 grid on

```

```

57 grid minor
58
59 fig1.Units = 'inches';
60 fig1.Position(3) = 4.5;
61 fig1.Position(4) = 3.2;
62 set(fig1.Children, 'FontName', 'Arial', 'FontSize', 11);
63 %print('EarthSailRadius', '-depsc')
64
65 fig2 = figure(2);
66 plot(te/yr, R/AU)
67 title('Sailcraft-Sun distance over mission')
68 xlabel('Time elapsed [yrs]')
69 ylabel('Distance from Sun [AU]')
70 box on
71 grid on
72 grid minor
73 ylim([0 123])
74 xlim([0 55])
75
76 fig2.Units = 'inches';
77 fig2.Position(3) = 4.5;
78 fig2.Position(4) = 3.2;
79 set(fig2.Children, 'FontName', 'Arial', 'FontSize', 11);
80 print('SunSailRadius', '-depsc')
81
82 fig3 = figure(3);
83 plot(te/yr, V)
84 title('Sailcraft velocity over mission')
85 xlabel('Time elapsed [yrs]')
86 ylabel('Velocity of Sailcraft [AU]')
87 box on
88 grid on
89 grid minor
90 xlim([0 55])
91
92 fig3.Units = 'inches';
93 fig3.Position(3) = 4.5;
94 fig3.Position(4) = 3.2;
95 set(fig3.Children, 'FontName', 'Arial', 'FontSize', 11);
96 print('SunSailVelocity', '-depsc')
97
98 fig4 = figure(4);
99 hold on
100 plot(t_tracking/yr,Temp_sail_daily)
101 grid on
102 grid minor
103 box on
104 title('Temperature of sailcraft over mission')
105 xlabel('Time elapsed [yrs]')
106 ylabel('Sailcraft temperature [K]')
107 xlim([0 21])
108 hold off
109
110 fig4.Units = 'inches';
111 fig4.Position(3) = 4.5;
112 fig4.Position(4) = 3.2;
113 set(fig4.Children, 'FontName', 'Arial', 'FontSize', 11);
114 print('TempTime', '-depsc')

```

A.2.5 Attitude sensitivity - ideal case

```

1 % Code that showcases the sailcraft remains spin-stabilised
2 % By Tanmay Ubgade | 220531
3
4 %% housekeeping
5 clear all
6 close all
7 clc
8
9 tic
10 %% Constants

```

```

11 % AU to km
12 AU = 149597870.7; % [km]
13 % Speed of Light
14 c = 299792458; % [m/s]
15 % Years to days
16 yr = 365.2422; % [days]
17 % Hours in a day
18 day = 24;
19
20 %% Inputs
21 side = sqrt(28.6); % Side length of sail in m
22 spinAxRot_initial = 0; % Rotation about spin-axis in degrees
23 alpha = 90; % Angle of Attack in degrees
24 I_tot = MomOfInertia(); % Moment of inertia of sail in kg/m^2
25
26 %% Acquiring positions of Sail when at alpha of 45 degrees
27 [te, R, ~, ~, ~, ~, ~, ~, ~, ~, ~, ~, ~, ~, ~, ~, ~, ~, G, ~, ~, ~, ~, ~, ~] =
    ImportGMATData(1);
28 TA = G(:,2); % True anomaly in degrees
29
30 R_maxtrack = 8*AU;
31 maxtrack_index = find(R<R_maxtrack, 1, 'last');
32 te_maxtrack = floor(te(maxtrack_index));
33 t_tracking = 1:1:te_maxtrack; t_tracking = t_tracking';
34 R_tracking = interp1q(te,R,t_tracking);
35 TA_tracking = interp1q(te,TA,t_tracking);
36
37 AoAofI = [45,135,225,315]; % Angles of Attack of interest
38 % Index of numbers before the stated angle, interpolate with index after
39 deg45 = [303,886,1443,2015,2640,3388,4444,6925];
40 deg135 = [451,1018,1565,2128,2743,2477,4541,6955];
41 deg225 = [91,667,1264,1863,2514,3287,4372,6896];
42 deg315 = [216,810,1385,1971,2609,3368,4432,6922];
43 deg0 = [260,853,1416,2002,2631,3380,4441,6926]; % Go backwards
44 deg180 = [2,556,1137,1711,2317,3010,3917,5695]; % Go backwards
45
46 R_ofI = zeros(length(deg45),6);
47 for i = 1:length(deg45)
48     R_ofI(i,1) = interpolator(45,TA_tracking(deg45(i)),...
49                             TA_tracking(deg45(i)+1),...
50                             R_tracking(deg45(i)),...
51                             R_tracking(deg45(i)+1)); % 45 degrees
52     R_ofI(i,2) = interpolator(135,TA_tracking(deg135(i)),...
53                             TA_tracking(deg135(i)+1),...
54                             R_tracking(deg135(i)),...
55                             R_tracking(deg135(i)+1)); % 135 degrees
56     R_ofI(i,3) = interpolator(225,TA_tracking(deg225(i)),...
57                             TA_tracking(deg225(i)+1),...
58                             R_tracking(deg225(i)),...
59                             R_tracking(deg225(i)+1)); % 225 degrees
60     R_ofI(i,4) = interpolator(315,TA_tracking(deg315(i)),...
61                             TA_tracking(deg315(i)+1),...
62                             R_tracking(deg315(i)),...
63                             R_tracking(deg315(i)+1)); % 315 degrees
64     R_ofI(i,5) = interpolator(0,TA_tracking(deg0(i)-1),...
65                             TA_tracking(deg0(i)),...
66                             R_tracking(deg0(i)-1),...
67                             R_tracking(deg0(i))); % 0 degrees
68     R_ofI(i,6) = interpolator(0,TA_tracking(deg180(i)-1),...
69                             TA_tracking(deg180(i)),...
70                             R_tracking(deg180(i)-1),...
71                             R_tracking(deg180(i))); % 0 degrees
72 end
73 R_ofI = R_ofI*1000; % Needs to be in meters
74 [row, col] = size(R_ofI);
75 %
76 %% Propagate attitude over 1 revolution
77 T1 = zeros(row, col); T2 = T1; %T3 = T1;
78
79 spinAxRot = 0:10:359.9;
80
81 % Choose which TorqueOnSail function to use depending on if it is offset
82 % case or idealised case
83
```

```

84 for i = 1:row
85     % 1 is 45 deg, 2 is 135 deg, 3 is 225 deg, 4 is 315 deg
86     for j = 5:6%5:col
87         for k = 1:length(spinAxRot)
88             Torque = TorqueOnSailOffset(side, spinAxRot(k), alpha, R_ofI(i,j));
89             T1(i,j,k) = Torque(1);
90             T2(i,j,k) = Torque(2);
91             %T3(i,j,k) = Torque(3);
92             clear Torque
93         end
94     end
95 end
96 NetTorque1 = zeros(row, col); NetTorque2 = zeros(row, col);
97 %}
98 RPM = 0.63; t_rev = 60/RPM;
99 t_rot = linspace(0,t_rev,length(spinAxRot));
100 PlotT1 = zeros(1,length(spinAxRot)); PlotT2 = PlotT1;
101 for i = 1:row
102     % 1 is 45 deg, 2 is 135 deg, 3 is 225 deg, 4 is 315 deg
103     for j = 5:6%5:col
104         for k = 1:length(spinAxRot)
105             PlotT1(k) = T1(i,j,k);
106             PlotT2(k) = T2(i,j,k);
107         end
108         NetTorque1(i,j) = trapz(t_rot,PlotT1);
109         NetTorque2(i,j) = trapz(t_rot,PlotT2);
110         figure(1)
111         hold on
112         plot(spinAxRot, PlotT1);
113         xlabel('Spin Axis Rotation [\degree]')
114         ylabel('Torque about x axis [Nm]')
115         title('T_1 over one revolution')
116         xlim([0,360])
117         hold off
118         % T2
119         figure(2)
120         hold on
121         plot(spinAxRot, PlotT2);
122         xlabel('Spin Axis Rotation [\degree]')
123         ylabel('Torque about y axis [Nm]')
124         title('T_2 over one revolution')
125         xlim([0,360])
126         hold off
127     end
128 end
129
130 toc
131 %% Functions
132
133 function [y] = interpolator(x,x1,x2,y1,y2)
134     y = y1 + (x-x1)*(y2-y1)/(x2-x1);
135 end

```

A.2.6 Attitude sensitivity - CoG offset case

```

1 % Code that simulates the accumulation and rotation of spin over the
2 % mission
3 % Tanmay Ubgaade | 220601
4 %% housekeeping
5 clear all
6 close all
7 clc
8
9 %% Constants
10 % AU to km
11 AU = 149597870.7; % [km]
12 % Speed of Light
13 c = 299792458; % [m/s]
14 % Years to days
15 yrs = 365.2422; % [days]
16 % Hours in a day

```

```

17 day = 24;
18
19
20 %% Inputs
21 side = sqrt(28.6); % Side length of sail in m
22 I_tot = MomOfInertia(); % Moment of inertia of sail in kg/m^2
23 [te, R, ~, ~, ~, ~, ~, ~, ~, ~, ~, ~, ~, ~, ~, ~, ~, ~, ~, ~, ~, ~] =
    ImportGMATData(1);
24 TA = G(:,2); % True anomaly in ',char(176),'rees
25 te_s = te*day*60*60; % Time elapsed in seconds
26
27 % Spin-axis rotation
28 RPM = 0.63;
29 t_rev = 60/RPM;
30 %% Torque Calculation loop
31 limit_21yrs = length(find((te < 21*yrs)));
32 T1 = zeros(1,limit_21yrs); T2 = T1; omega_dot_1 = T1; omega_dot_2 = T1;
33
34 offset = [-10, -5, -1, -0.1, -0.05, 0, 0.05, 0.1, 1, 5, 10]/1000; % CG offset value
    in meters
35
36 rot1 = zeros(1,length(offset)); rot2 = rot1; omega1 = rot1; omega2 = rot1;
37 count = 1;
38
39 for j = offset
40     tic
41         for i = 1:limit_21yrs
42             spinAxRot = rem(te_s(i),t_rev)*360;
43             Torque = TorqueOnSailOffset(side,spinAxRot,AoA(TA(i)),R(i),j);
44             T1(i) = Torque(1);
45             T2(i) = Torque(2);
46             omega_dot = EulerEq12(Torque,I_tot);
47             omega_dot_1(i) = omega_dot(1);
48             omega_dot_2(i) = omega_dot(2);
49             clear Torque omega_dot
50         end
51
52 %% Rotation rate
53 omega1 = trapz(te_s(1:limit_21yrs), omega_dot_1);
54 omega1_array = cumtrapz(te_s(1:limit_21yrs), omega_dot_1);
55
56 omega2 = trapz(te_s(1:limit_21yrs), omega_dot_2);
57 omega2_array = cumtrapz(te_s(1:limit_21yrs), omega_dot_2);
58
59 %% Rotations
60 rot1(count) = trapz(te_s(1:limit_21yrs), omega1_array);
61 rot1_array = cumtrapz(te_s(1:limit_21yrs), omega1_array);
62
63 rot2(count) = trapz(te_s(1:limit_21yrs), omega2_array);
64 rot2_array = cumtrapz(te_s(1:limit_21yrs), omega2_array);
65
66 %% Plots
67
68 fig1 = figure(1);
69 hold on
70 plot(te_s(1:limit_21yrs)/(day*yrs*60*60),rot1_array)
71 title('X-axis rotation over mission')
72 xlabel('Time elapsed [yrs]')
73 ylabel(['X-axis rotation [',char(176),'s]'])
74 grid on
75 box on
76 xlim([0,21])
77 hold off
78
79 fig2 = figure(2);
80 hold on
81 plot(te_s(1:limit_21yrs)/(day*yrs*60*60),rot2_array)
82 title('Y-axis rotation over mission')
83 xlabel('Time elapsed [yrs]')
84 ylabel(['Y-axis rotation [',char(176),'s]'])
85 grid on
86 box on
87 xlim([0,21])
88 hold off

```

```

89
90 fig3 = figure(3);
91 hold on
92 plot(te_s(1:limit_21yrs)/(day*yrs*60*60),omegai_array)
93 title('$\omega_1$ over mission','interpreter','latex')
94 xlabel('Time elapsed [yrs]')
95 ylabel(['X-axis angular velocity [',char(176),'s^{-1}]]')
96 grid on
97 box on
98 xlim([0,21])
99 hold off
100
101 fig4 = figure(4);
102 hold on
103 plot(te_s(1:limit_21yrs)/(day*yrs*60*60),omega2_array)
104 title('$\omega_2$ over mission','interpreter','latex')
105 xlabel('Time elapsed [yrs]')
106 ylabel(['Y-axis angular velocity [',char(176),'s^{-1}]]')
107 grid on
108 box on
109 xlim([0,21])
110 hold off
111
112 fig5 = figure(5);
113 hold on
114 plot(te_s(1:limit_21yrs)/(day*yrs*60*60),omega_dot_1)
115 title('$\dot{\omega}_1$ over mission','interpreter','latex')
116 xlabel('Time elapsed [yrs]')
117 ylabel(['X-axis angular acceleration [',char(176),'s^{-2}]]')
118 grid on
119 box on
120 xlim([0,21])
121 hold off
122
123 fig6 = figure(6);
124 hold on
125 plot(te_s(1:limit_21yrs)/(day*yrs*60*60),omega_dot_2)
126 title('$\dot{\omega}_2$ over mission','interpreter','latex')
127 xlabel('Time elapsed [yrs]')
128 ylabel(['Y-axis angular acceleration [',char(176),'s^{-2}]]')
129 grid on
130 box on
131 xlim([0,21])
132 hold off
133
134
135 clear omega1_array omega2_array omega_dot_1 omega_dot_2 ...
136 rot1_array rot2_array
137 count = count + 1;
138 toc
139 end
140
141
142
143 fig1.Units = 'inches';
144 fig1.Position(3) = 6;
145 fig1.Position(4) = 3;
146 set(fig1.Children, 'FontName', 'Arial', 'FontSize', 11);
147
148 fig2.Units = 'inches';
149 fig2.Position(3) = 6;
150 fig2.Position(4) = 3;
151 set(fig2.Children, 'FontName', 'Arial', 'FontSize', 11);
152
153 fig3.Units = 'inches';
154 fig3.Position(3) = 6;
155 fig3.Position(4) = 3;
156 set(fig3.Children, 'FontName', 'Arial', 'FontSize', 11);
157
158 fig4.Units = 'inches';
159 fig4.Position(3) = 6;
160 fig4.Position(4) = 3;
161 set(fig4.Children, 'FontName', 'Arial', 'FontSize', 11);
162

```

```

163 fig5.Units = 'inches';
164 fig5.Position(3) = 6;
165 fig5.Position(4) = 3;
166 set(fig5.Children, 'FontName', 'Arial', 'FontSize', 11);
167
168 fig6.Units = 'inches';
169 fig6.Position(3) = 6;
170 fig6.Position(4) = 3;
171 set(fig6.Children, 'FontName', 'Arial', 'FontSize', 11);
172
173
174
175 figure(1)
176 legend('x_{cg}= -10mm', 'x_{cg}= -5mm', 'x_{cg}= -1mm', 'x_{cg}= -0.1mm',...
177     'x_{cg}= -0.05mm', 'x_{cg}= 0mm', 'x_{cg}= 0.05mm', 'x_{cg}= 0.1mm',...
178     'x_{cg}= 1mm', 'x_{cg}= 5mm', 'x_{cg}= 10mm',...
179     'Location', 'eastoutside', 'FontSize', 8)
180 figure(2)
181 legend('x_{cg}= -10mm', 'x_{cg}= -5mm', 'x_{cg}= -1mm', 'x_{cg}= -0.1mm',...
182     'x_{cg}= -0.05mm', 'x_{cg}= 0mm', 'x_{cg}= 0.05mm', 'x_{cg}= 0.1mm',...
183     'x_{cg}= 1mm', 'x_{cg}= 5mm', 'x_{cg}= 10mm',...
184     'Location', 'eastoutside', 'FontSize', 8)
185 figure(3)
186 legend('x_{cg}= -10mm', 'x_{cg}= -5mm', 'x_{cg}= -1mm', 'x_{cg}= -0.1mm',...
187     'x_{cg}= -0.05mm', 'x_{cg}= 0mm', 'x_{cg}= 0.05mm', 'x_{cg}= 0.1mm',...
188     'x_{cg}= 1mm', 'x_{cg}= 5mm', 'x_{cg}= 10mm',...
189     'Location', 'eastoutside', 'FontSize', 8)
190 figure(4)
191 legend('x_{cg}= -10mm', 'x_{cg}= -5mm', 'x_{cg}= -1mm', 'x_{cg}= -0.1mm',...
192     'x_{cg}= -0.05mm', 'x_{cg}= 0mm', 'x_{cg}= 0.05mm', 'x_{cg}= 0.1mm',...
193     'x_{cg}= 1mm', 'x_{cg}= 5mm', 'x_{cg}= 10mm',...
194     'Location', 'eastoutside', 'FontSize', 8)
195 figure(5)
196 legend('x_{cg}= -10mm', 'x_{cg}= -5mm', 'x_{cg}= -1mm', 'x_{cg}= -0.1mm',...
197     'x_{cg}= -0.05mm', 'x_{cg}= 0mm', 'x_{cg}= 0.05mm', 'x_{cg}= 0.1mm',...
198     'x_{cg}= 1mm', 'x_{cg}= 5mm', 'x_{cg}= 10mm',...
199     'Location', 'eastoutside', 'FontSize', 8)
200 figure(6)
201 legend('x_{cg}= -10mm', 'x_{cg}= -5mm', 'x_{cg}= -1mm', 'x_{cg}= -0.1mm',...
202     'x_{cg}= -0.05mm', 'x_{cg}= 0mm', 'x_{cg}= 0.05mm', 'x_{cg}= 0.1mm',...
203     'x_{cg}= 1mm', 'x_{cg}= 5mm', 'x_{cg}= 10mm',...
204     'Location', 'eastoutside', 'FontSize', 8)
205
206
207 %% Functions
208 function alpha = AoA(TA)
209     if (TA > 0) & (TA < 90)
210         alpha = 90 - TA;
211     elseif (TA > 90) & (TA < 180)
212         alpha = TA - 90;
213     elseif (TA > 180) & (TA < 270)
214         alpha = 180 - TA;
215     else
216         alpha = TA - 270;
217     end
218 end

```