73rd International Astronautical Congress (IAC), Paris, France, 18-22 September 2022.

IAC-22,E2,3-GTS.4,13,x74358

STRATHcube: A CubeSat Against Space Debris

Lewis Gray*, Ewan Leitch, Julie Graham, Andrew R. Wilson, Massimiliano Vasile

Aerospace Centre of Excellence, Department of Mechanical & Aerospace Engineering, University of Strathclyde, 75 Montrose Street, Glasgow, G1 1XJ, United Kingdom
*Corresponding Author: lewig1999@hotmail.co.uk

Abstract

The responsible management of space debris is critical for the continued use of space. The STRATHcube project purposes a CubeSat which focusses on two issues of space debris - detection and removal. There is an increasing need to detect, track and catalogue debris in the Low Earth Orbit (LEO).

The first payload tracks orbital debris by using Passive Bistatic Radar (PBR). The project purposes to launch the CubeSat into LEO as a PBR technology demonstrator, where a signal processor algorithm developed at the University of Strathclyde to detect debris will be tested. If debris were to pass between the CubeSat and the transmitting satellite, this signal would be disturbed, indicating the presence of debris. If adopted in industry, this method can be upscaled to provide data at increased accuracy, reduced cost, and higher availability.

As a secondary payload, the STRATHcube project aims to provide data on fragmentation of solar panels upon re-entry into the atmosphere. To reduce the volume of debris in low orbits, the Design for Demise (D4D) initiative champions removal of debris via uncontrolled atmospheric re-entry in which satellites completely demise. Current D4D analysis tools under-predict the effectiveness of break up upon re-entry due to a lack of re-entry data – in particular fragmentation data. With this flight data, STRATHcube aims to provide greater validation and verification of satellite re-entry modelling tools that currently exist, as well as providing the framework for future fragmentation studies.

To verify the life cycle of the CubeSat, an Integrated Systems Tool (IST) has been developed. Using MATLAB code, the IST creates a digital twin of the CubeSat which provides a high-fidelity simulation of the propagation as well as interlinking subsystems of STRATHcube (mission analysis, AOCS, power, thermal). This provides an important initial step to verify the component and orbit selection prior to product procurement and launch of the STRATHcube mission. The IST uses a Runge-Kutta 4th Order (RK4) numerical integrator, which currently includes the core mechanics (idealised control, actuator control) and determination (Wahba's problem with two separate methods) of the orientation of the CubeSat, with the power profile developed, but not integrated. Some results have been confirmed with additional resources to verify their accuracy, and any future results are recommended to have validation. Notable considerations for further development include integrating the power profile, developing the thermal model, and creating a GUI.

Keywords: Debris Detection; CubeSat; Radar; Passive Bistatic Radar, Simulation

Nomenclature

Symbol(s)	Variable(s)	Unit
$[\bar{B}N]$	Rotation matrix from the inertial to estimated body frame	N/D
$e_1, e_2, e_3, \hat{\boldsymbol{e}}$	PRV x, y, z components, and unit vector	km
$h_x, h_y, h_z, \hat{\boldsymbol{h}}$	Components, and unit vector of orbit angular momentum	$\frac{kg\ m^2}{s}$
$\bar{\boldsymbol{L}}$	Estimated torque vector for idealized control	Nm
L, M, N	Magnetorquer torque x, y, z components	Nm
$\hat{n}_{Ri}$	Unit direction of reaction wheel in body frame	N/D
$[NT]$	Rotation matrix from T to N frame	N/D
q_0, q_1, q_2, q_3	Quaternion components	0123
q_1, q_2, q_3, q_4	Quaternion components	1234
$\dot{q}_0, \dot{q}_1, \dot{q}_2, \dot{q}_3$	Quaternion rate components	1234
$q_{c1}, q_{c2}, q_{c3}, q_{c4}$	Quaternion components – Commanded orientation	1234
$\bar{\boldsymbol{q}}, \boldsymbol{q}, \boldsymbol{q}_c$	Estimated, actual, and commanded unit quaternion	1234 Or 0123

${}^{\mathcal{N}}\hat{\boldsymbol{r}}_{measured}$	Zenith sensor model position	km
${}^{\mathcal{N}}\hat{\boldsymbol{r}}_{position}$	Simulation radial vector position in Inertial frame	km
$R_1(\theta)$, $R_2(\theta), R_3(\theta)$	Rotation matrix of the x, y, z axis	N/D
${}^{\mathcal{B}}\hat{\boldsymbol{v}}_k$ ${}^{\mathrm{N}}\hat{\mathrm{V}}_{\mathrm{k}}$	Measured vector from the sensor in the body frame (has error applied)	Sensor dependent
${}^{\mathcal{N}}\hat{\boldsymbol{v}}_k$ ${}^{\mathrm{B}}\hat{\mathrm{V}}_{\mathrm{k}}$	Measured vector from the sensor in the inertial frame (no error applied)	Sensor dependent
$\boldsymbol{w}_k$	Weights vector of the sensors	N/D
$\hat{\boldsymbol{x}}_{em}$	x-axis error vector with the 90degree offset	km
$\hat{\boldsymbol{x}}_t$	x-axis true vector	km
$\hat{\boldsymbol{x}}_{tm}$	x-axis error vector with the 90degree offset	km
$x_x, x_y, x_z, \hat{\boldsymbol{x}}$	Components, and unit vector of y axis	km
$y_x, y_y, y_z, \hat{\boldsymbol{y}}$	Components, and unit vector of y axis	km
$z_x, z_y, z_z, \hat{\boldsymbol{z}}$	Components, and unit vector of z axis	km
$\beta_x, \beta_y, \beta_z$	Magnetic field x, y, z components	T
$\boldsymbol{\beta}$	CRP vector	N/D
$\delta\boldsymbol{q}_{1:3}, \delta q_4$	Error quaternion	1234
$\theta, \theta_t, \theta_e$	2nd sequential rotation, true vector, error vector	Rad
Φ	PRV rotation angle	Degree
ϕ, ϕ_t, ϕ_e	3rd sequential rotation, true vector, error vector	Rad
ψ, ψ_t, ψ_e	1st sequential rotation, true vector, error vector	Rad
$\omega_x, \omega_y, \omega_z$	Angular velocity x, y, z components	Rad/s
$\boldsymbol{\omega}_B, \dot{\boldsymbol{\omega}}_B$	Angular velocity and angular velocity rate in body frame	Rad/s
$\boldsymbol{\omega}$	Arbitrary angular velocity	Rad/s

Acronyms/Abbreviations

ADCS	Attitude Determination and Control System
AOCS	Attitude and Orbit Control System
CONOPS	Concept of operations
CRP	Classical Rodrigues Parameter
DRAMA	Debris Risk Assessment and Mitigation Analysis
EOL	Design for Demise
FOV	Field of View
ESA	End of Life
GUI	Graphical User Interface
IMU	Inertial Measurement Unit
ISS	International Space Station
IST	Integrated Simulation Tool
LEO	Low Earth Orbit
LOS	Line of Sight
N/D	Non-Dimensional
NR	Number of Reaction Wheels
PBR	Passive Bistatic Radar
PD	Proportional-Derivative
PRV	Principal Rotation Vector
PSD	Power Spectral Density
RK4	Runge Kutta 4th Order
RW	Reaction Wheel
Sat	Satellite

1 Introduction

Since the launch of the first satellite in 1957, the number of objects being launched into orbit, in particular Low Earth Orbit (LEO), has increased exponentially. Consequently, the level of space debris present within LEO is rapidly approaching catastrophic levels due to the ever-increasing chances of a Kessler Syndrome event occurring in which a collision, even amongst small pieces of debris, escalates via a domino effect of the resulting shrapnel from the collision. As a result, potentially thousands of operational satellites could be critically damaged and, furthermore, the possibility for future launches into LEO could be severely diminished. Henceforth, the need to detect, catalogue and track debris in LEO is essential.

Currently, it is estimated that there are over 19,000 objects currently in LEO around the Earth and over 52% of those objects are thought to be considered space debris [1]. This is a stark increase from 2020 when only 14,000 bodies were recorded in orbit. This trend highlights the increasing number of satellites that are being launched into LEO, partially due to private companies providing more regular and affordable access to launches. To date there have been four known collisions between artificial satellites [2], the largest of them occurring in 2007 when a collision between Iridium 33 and Cosmos 2251 generated over 1,500 fragments of debris.

Currently, orbital debris and meteoroids in LEO are detected via telescopes, radar and laser systems, all of which are ground-based. These methods are limited as optical detection of satellites of debris can only be made during short periods each day and furthermore dependent on weather conditions, radar technologies are very energy demanding and laser systems, although highly accurate, lack consistency [3][4].

The STRATHcube projects offers a novel solution to debris detection by proposing the first ever in-orbit debris detection system using Passive Bistatic Radar (PBR). As a technology demonstration, STRATHcube will hopefully prove the potential to provide high resolution data on space debris in LEO via the use of PBR, whilst bypassing the severe energy demands, inconsistencies and time constraints posed by current methods. This will be the primary payload of the STRATHcube satellite.

To avoid the further cluttering of LEO, artificial satellites launched into orbit must have a strategy to remove themselves from LEO after End of Life (EOL). There are two options for moving a satellite out of LEO; either moving it to a graveyard orbit or letting it re-enter the atmosphere in a controlled or uncontrolled fashion. To manoeuvre a satellite from LEO to a graveyard orbit (over 40,000km altitude) requires an immense amount of fuel which would be infeasible for most missions [5]. Likewise, controlled re-entries where satellites are manoeuvred to land in an ocean and later recovered can require major fuel masses and add further complexities to missions. Therefore, the most feasible solution is uncontrolled re-entry for satellites.

However, uncontrolled re-entry introduces the problem of ensuring objects re-entering the atmosphere are fully disintegrated before reaching ground level. The Design for Demise (D4D) initiative was introduced to minimise the risk involved in re-entry by encouraging responsible material selection and the use of re-entry models that predict the position and timing of re-entry. In addition to this analysis tools are used to estimate how the satellite will fragment upon re-entry and if there is any risk of components surviving. Despite this, a 2013 study into re-entry predictions for uncontrolled satellites found that relative error in predictions can often reach up to 20% [6].

The secondary payload onboard STRATHcube aims to help address the unpredictability in re-entry models by sending flight data throughout its re-entry through the atmosphere. This involves tracking the fragmentation of the solar panels that will burn up prior to the chassis of the CubeSat. A thermal imaging camera paired with mechanical and thermal sensors are used to capture detail the fragmentation process of the solar panels and provide data to contribute to the D4D program.

To verify the design of the STRATHcube mission, a digital twin is being created to simulate the lifespan of the CubeSat. This is known as the Integrated Systems Tool (IST). The IST currently simulates the Attitude, Orbit and Control System (AOCS) of the satellite using quaternions to simulate the behaviour of control methods on board the spacecraft.

Ultimately, the IST will be able to verify the feasibility of the STRATHcube mission with regards to power and thermal requirements given any orbit parameters, which is essential as it is uncertain exactly what orbit STRATHcube will be launched into.

1.1 STRATHcube Mission

The concept of operations (CONOPS) for the STRATHcube mission can be seen in Fig 1.

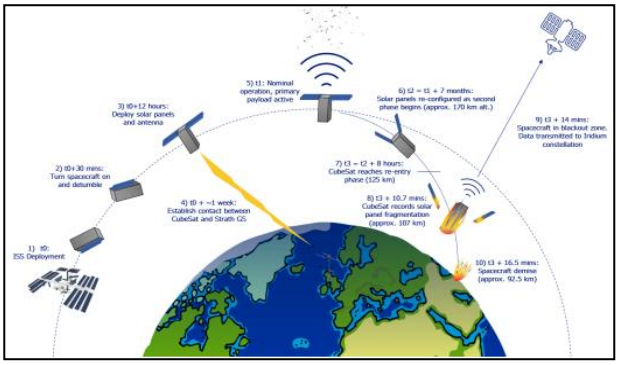

Fig 1. STRATHcube CONOPS [7]

STRATHcube will be launched into LEO, currently the mission is modelled around a launch from the International Space Station (ISS), however the mission is designed to be flexible to accommodate any launch provider. After the detumble phase is completed, the solar panels are deployed into their nominal operation position at 90 degrees to the body of the spacecraft and a week later the primary payload is activated. From this point onwards, STRATHcube will be able to detect, track and catalogue debris.

After 7 months of operation of the primary payload, STRATHcube's orbit will have naturally decayed to an altitude where it is approaching re-entry, at which point the spacecraft prepares for operation of the secondary payload. To do so, the solar panels are reconfigured to a shuttlecock position to allow for a controlled orientation during re-entry. As the CubeSat begins re-entry, it broadcasts data from the thermal and mechanical sensors as well as images from the thermal camera to the Iridium constellation, detailing the aerothermodynamic behaviour during re-entry as well as providing data of the fragmentation process.

After 16 minutes of broadcasting, the satellite will no longer be able to broadcast data to the Iridium constellation and will proceed to demise, signalling the successful end of the STRATHcube mission.

1.2 Primary Payload – Passive Bistatic Radar (PBR)

A bistatic radar describes a radar system in which the transmitter and receiver are sufficiently distanced so that the ranges or angles are adequately different. Persico et al. [8] explored the concept of Spaceborne PBR which offers a solution to in orbit debris tracking which can bypass the attenuation experienced during receiving signals through the Earth's atmosphere. Using an algorithm developed at the University of Strathclyde, the spacecraft will be able to detect, track and catalogue debris. Figure 2 shows how the forward scatter region created by an intercepting object passing through the line of sight of the transmitter and receiver

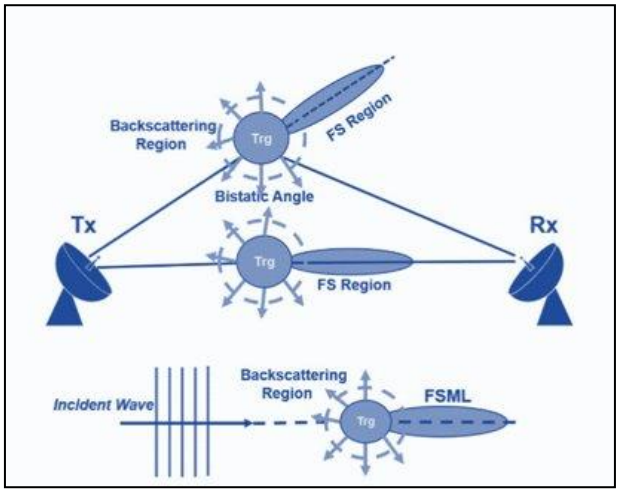

Fig 2. PBR Demonstration

Jenkins et al.[7] conceptualised how to adapt the technology to be suitable for applications in CubeSats. To do so the spacecraft had to be able to receive signals from the transmitting satellite, known as the illuminator of opportunity. The Iridium constellation was chosen due to its reliability. By choosing a compact patch antenna that was compatible with the Iridium constellation broadcasted signal, the technology was able to be incorporated into the STRATHcube design conveniently and affordably. A radar receiver is still to be developed to complete the technology.

1.3 Secondary Payload – Fragmentation Analysis of Solar Panels Upon Re-Entry

The secondary payload of STRATHcube looks to collect data of the satellites aerothermodynamic properties during re-entry. Graham et al. [9] designed a secondary payload which, through the use of thermal and mechanical sensors, will collect data from the fragmentation of the spacecraft during re-entry. This data can then be used to contribute to the D4D initiative.

In order to control the orientation of the CubeSat at the beginning of re-entry, a passive stabilisation system is implemented. By reorientating the solar panels to behind the body of the satellite, it is possible to manipulate the centre of pressure behind the centre of gravity, creating stability during re-entry.

Due to the severe conditions experienced upon re-entry, it is impossible to broadcast during this phase to a ground station on Earth. Therefore, during re-entry STRATHcube will broadcast data throughout its descent to the Iridium constellation and downlink the data later via dial up connection. To verify the feasibility of STRATHcube's secondary payload, a link budget was generated using the currently selected components used for transmission during re-entry (Maxtena and Iridium Certus modem) and properties of the Iridium constellation.

Table 1. Link Budget for STRATHcube's Secondary Payload

Uplink CubeSat Tx Iridium Constellation		
	Range (km)	
	655	**700**
Frequency (GHz)	1.626	1.626
Antenna Size	0.019	0.019
Ptx	2.5	2.5
Gtx	2.8	2.8
Lfs	152.9872368	153.5643716
Lm	10	10
Lc	2	2
Grx	24.9	24.9
EIRP (dBW)	9	9
Prx (W)	-134.787237	-135.3643716
System Temperature [10]	13	13
Boltzmann's constant	-228.599167	-228.5991672
Data Rate (b/s)	88000	88000
Eb/N0	31.36710365	30.78996885
Eb/No Required	9.9	9.9
Link Margin	21.46710365	20.88996885

The link budget of Table 1 details two scenarios: transmission from the CubeSat to the Iridium constellation at the beginning of re-entry (655km) and the transmission from the CubeSat to the Iridium constellation at the estimated end of its re-entry (700.05km). As can be seen, even in the worst-case scenario where the distance between transmitter and receiver is at a maximum, the link margin is substantially above the recommended value of 3dB for a successful transmission. However, assumes the signal is travelling

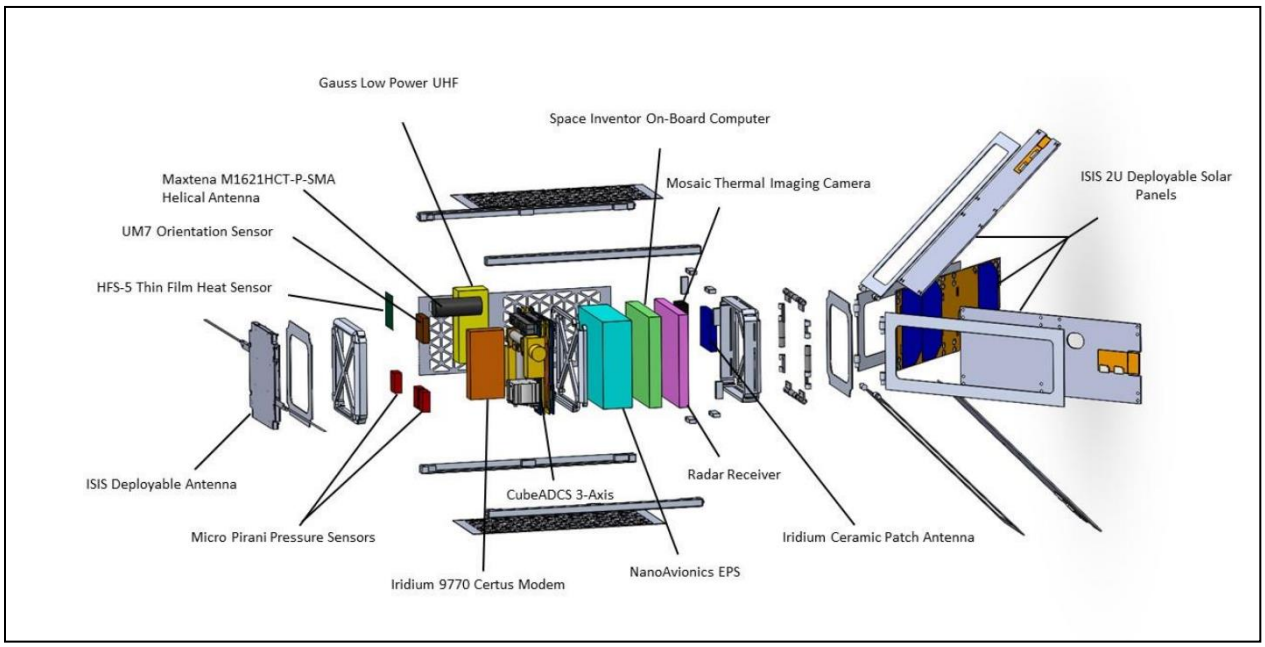

Fig 3. Exploded View of STRATHcube

through the vacuum of space which will not be the case as the satellite re-enters the atmosphere. Regardless, the link budget margin remains >9dB for both cases for when full atmosphere attenuation is accounted for [11] demonstrating the capability of the selected components.

1.4 STRATHcube Design

STRATHcube is a 2U CubeSat designed from Aluminium 7075-T6. The structure is custom designed to accommodate for deployables that extrude from the body and has been subjected to virtual modal testing, verifying its compliance with ESA FYS! launch standards. The qualification and acceptance test profiles can be seen as power spectral density (PSD) against frequency shown in Figure 3.

The design encompasses four 2U deployable solar panels which use a step geared motor to control the deployment of the panels. The solar panels are required to change orientation dependent on the phase of the mission (as illustrated in Figure 1). Furthermore, the possibility of manoeuvring the solar panels in order to maximise power generation is also being explored, however a further risk analysis must be conducted to explore the feasibility.

The satellite is equipped with a high data rate antenna (250 kbps) to allow for successful data transmission to the James Weir ground station, located at the University of Strathclyde, Glasgow. The high data rate is required to combat the short daily view times experienced during the operation of the primary payload.

A full breakdown of the STRATHcube design can be seen in Figure 3.

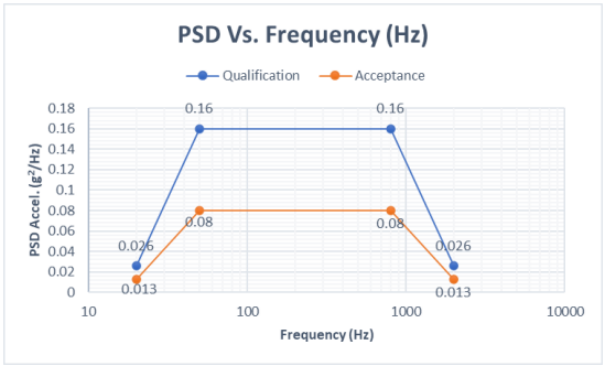

Figure 4 Acceptance/Qualification Test Profiles

2 IST Background

The IST is used for the purpose of verifying the design of the CubeSat and future alterations so that components can be bought with confidence for further testing. It is an integrated simulation which currently combines the Mission Analysis, and Attitude Determination and Control System (ADCS), with the power profile completed but as an independent script. It uses a numerical integration method called RK4 [12]. Previous work has included development of the thermal subsystem; however, this has not been integrated. Visualisation has been setup for more intuitive understanding of the data which uses numerous orientation parameters such as; Euler Angles (easier intuition), Rotation Matrix (non-singular, tracking), Quaternions (non-singular transformations at speed), and Classical Rodrigues Parameters (determination).

2.1 Orientation Parameters

The CubeSat will need to describe its orientation in space. To do this particular parameters have been used throughout the development of the IST which has a specific use for each section. Orientation is being able to define three orthonormal vectors (rotated frame) relative to a separate frame.

2.1.1 Euler Angles

Euler angles are a more complicated version of spherical coordinates which instead of describing a single point in space, represents the orientation of three orthonormal vectors using angles. The Euler angles are sequential rotations, which come in 12 different sets [13]. In Fig 5, a 3-2-3 set can be seen.

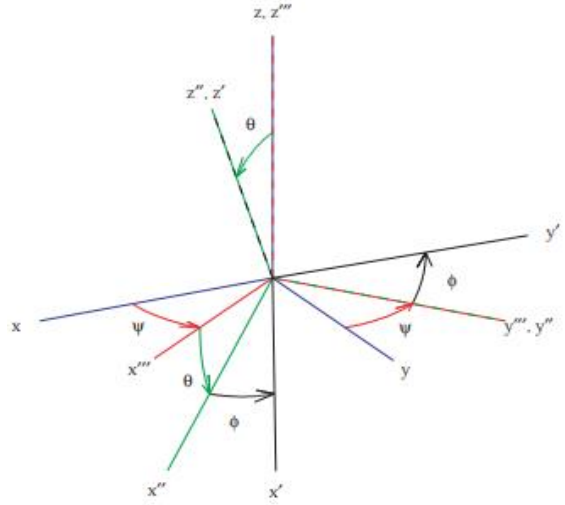

Fig 5. 3-2-3 Euler Angle Rotations [13]

2.1.2 Principal Rotation Vector (PRV)

The PRV is a more compact way of representing the Euler angles, which instead of having three sequential fixed axis to rotate round, there is a single alterable axis. The rotation angle is round the changeable axis, which can represent a single angle offset between two different frames. This is useful for error comparison of separate orientations [14].

2.1.3 Rotation Matrix

This variable is a 3*3 matrix which describes rotations round specific axis. These specific axis can be related to the Euler angles and combined into the different sets [13]. The individual rotations can be seen in Equations (1)-(3), where θ is an arbitrary rotation.

$$R_1(\theta) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\theta & \sin\theta \\ 0 & -\sin\theta & \cos\theta \end{bmatrix} \quad (1)$$

$$R_2(\theta) = \begin{bmatrix} \cos\theta & 0 & -\sin\theta \\ 0 & 1 & 0 \\ \sin\theta & 0 & \cos\theta \end{bmatrix} \quad (2)$$

$$R_3(\theta) = \begin{bmatrix} \cos\theta & \sin\theta & 0 \\ -\sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (3)$$

By relating the individual rotations, a full Euler set can be obtained as seen in Equation (4) for a 3-2-1 set.

$$R_{321}(\psi, \theta, \phi) = R_1(\phi)R_2(\theta)R_3(\psi) \quad (4)$$

Fundamentally the rotation matrix will describe the position of the rotated frame relative to the original frame, which can be applied to the Track Frame (Equation (5)) for the first payload operation.

$$[NT] = \begin{bmatrix} -h_x & -h_y & -h_z \\ y_x & y_y & z_z \\ z_x & z_y & z_z \end{bmatrix} \quad (5)$$

This format allows the use of the orbit angular momentum vector (Equation (6)), the zenith vector (Equation (7)) and the last vector can be found using the cross product (Equation (8)) to uphold the orthonormal constraints. The reason h is negative is to have the y vector pointing towards the direction of the velocity, which was initially for easier understanding of simulations.

$$\hat{\boldsymbol{h}} = [h_x, h_y, h_z]^T \quad (6)$$

$$\hat{\boldsymbol{z}} = [z_x, z_y, z_z]^T \quad (7)$$

$$\hat{\boldsymbol{y}} = \hat{\boldsymbol{z}} \times (-\hat{\boldsymbol{h}}) \quad (8)$$

2.1.4 Quaternions

A quaternion is a 4D number which describes the orientation by a point on the surface of a hypersphere. This is for mathematical reasons which can circumvent singularity issues related to the Euler angles but be more efficient than the Rotation Matrix. They also come in two different sets, 1234, and 0123, which are both used within the simulation so they will specified on each occasion. For a 0123 set, the quaternion is related to a PRV through Equation (9) - (12) which highlights the real (0) and vector (123) component of the unit quaternion through the components of $\hat{\boldsymbol{e}}$ [14].

$$q_0 = \cos\left(\frac{\Phi}{2}\right) \quad (9)$$

$$q_1 = e_1 \sin\left(\frac{\Phi}{2}\right) \quad (10)$$

$$q_2 = e_2 \sin\left(\frac{\Phi}{2}\right) \quad (11)$$

$$q_3 = e_3 \sin\left(\frac{\Phi}{2}\right) \quad (12)$$

2.1.5 Classical Rodrigues Parameter (CRP)

A CRP is the stereographic projection of a 4D unit quaternion into 3D space, similar to how the 3D spherical surface of Earth is projected into a 2D map. This parameter, in this case, is only used for orientation determination purposes. Where the CRP converting to a unit quaternion is shown in Equation (13) [14].

$$\bar{\boldsymbol{q}} = \frac{1}{1+\boldsymbol{\beta}^T\boldsymbol{\beta}}\begin{bmatrix}1\\ \boldsymbol{\beta}\end{bmatrix} \tag{13}$$

2.2 Mechanics

2.2.1 Kinematics

Kinematics describes how a body moves without external forces, which is shown in Equation (14) [14].

$$\begin{pmatrix}\dot{q}_0\\ \dot{q}_1\\ \dot{q}_2\\ \dot{q}_3\end{pmatrix} = \frac{1}{2}\begin{bmatrix}0 & -\omega_x & -\omega_y & -\omega_z\\ \omega_x & 0 & \omega_z & -\omega_y\\ \omega_y & -\omega_z & 0 & \omega_x\\ \omega_z & \omega_y & -\omega_x & 0\end{bmatrix}\begin{pmatrix}q_0\\ q_1\\ q_2\\ q_3\end{pmatrix} \tag{14}$$

Where $\omega_B = \left[\omega_x, \omega_y, \omega_z\right]^T$ represents the angular rate vector in the orbit reference frame.

2.2.2 Dynamics

Dynamics considers forces acting on the object as seen in Equation (15).

$$I_{sc}\dot{\boldsymbol{\omega}}_B = \boldsymbol{T_c} + \boldsymbol{T}_d - \boldsymbol{\omega}_B(I_{sc}\boldsymbol{\omega}_B + \boldsymbol{h}_w) \tag{15}$$

I_{sc} is the satellite inertia matrix in the body frame, $\boldsymbol{T}_C\mathrm{T_c}$ is the control vector, $\boldsymbol{T}_d$ is the sum of disturbances, and $\boldsymbol{h}_w$ is the reaction wheel angular momentum vector. The disturbances could potentially include drag, magnetic field strength, solar radiation pressure, and gravity gradient, however these are not included in the ADCS model. Although, the orbit perturbations include J2, drag, and 3rd body effects from the Sun and Moon. Thus the ADCS perturbations are recommended for integration to have more realistic results.

2.2.3 Idealised Control

The ADCS platform requires two main aims to be met, detumbling and zenith orientation tracking. The detumbling is primarily at initial ejection from the launch vehicle into space. This is trying to achieve the CubeSat angular velocity to tend towards zero. The tracking case will have the CubeSat orientating away from the surface of Earth to point in the direction of the Iridium constellation which will enable first payload operation and debris detection. A non-tracking controller is used as an ideal case (non-actuator specific) to prove the functionality of the IST and allows further development of other subsystems. This uses a Linear Controller (Equation (16)), which is used for verification later in this report, Equation 7.17a in [15].

$$\bar{\boldsymbol{L}} = -k_p sign(\delta q_4)\delta\boldsymbol{q}_{1:3} - k_d\boldsymbol{\omega} \tag{16}$$

Where the components of Equation (16) are shown in Equation (17) - (19) [15].

$$\delta\boldsymbol{q}_{1:3} = \Xi^T(\boldsymbol{q}_c)\boldsymbol{q} \tag{17}$$

$$\delta q_4 = \boldsymbol{q}^T\boldsymbol{q}_c \tag{18}$$

$$\Xi(\boldsymbol{q}_c) = \begin{bmatrix}q_{c_4} & -q_{c_3} & q_{c_2}\\ q_{c_3} & q_{c_4} & -q_{c_1}\\ -q_{c_2} & q_{c_1} & q_{c_4}\\ -q_{c_1} & -q_{c_2} & -q_{c_3}\end{bmatrix} \tag{19}$$

The IST can be used with Equation (20), a nonlinear control law from Equation 7.17b [15]. It is thought that due to its nonlinearity it is more robust than Equation (16).

$$\begin{aligned}\bar{\boldsymbol{L}} = &-k_p sign(\delta q_4)\delta\boldsymbol{q}_{1:3}\\ &-k_d(1 \pm \delta\boldsymbol{q}_{1:3}^T\delta\boldsymbol{q}_{1:3})\boldsymbol{\omega}\end{aligned} \tag{20}$$

2.2.4 Magnetorquer

The detumble manoeuvre can be completed using a magnetorquer. The B-dot control law is shown in Equation (21), which uses 3 magnetorquers on each axis, and a gain value of k [16].

$$\begin{bmatrix}L\\ M\\ N\end{bmatrix} = -k\begin{bmatrix}\beta_y^2+\beta_z^2 & -\beta_x\beta_y & -\beta_x\beta_z\\ -\beta_x\beta_y & \beta_x^2+\beta_z^2 & -\beta_y\beta_z\\ -\beta_x\beta_z & -\beta_y\beta_z & \beta_x^2+\beta_z^2\end{bmatrix}\begin{bmatrix}\omega_x\\ \omega_y\\ \omega_z\end{bmatrix} \tag{21}$$

The magnetorquers have different properties on each axis, which will need to be accounted for in the simulation as future work.

2.2.5 Reaction Wheel

Whilst the magnetorquers are useful for detumble, the reaction wheels are beneficial for accurate 3-axis control. An inertia matrix for an individual reaction wheel is in Equation (22), where the subscript R refers the centre of mass of the reaction wheel, and the superscript R refers to the frame of reference [16].

$$I_{Ri}^R = \begin{bmatrix}\frac{m_{RW}r_{RW}^2}{2} & 0 & 0\\ 0 & \frac{m_{RW}}{12}(3r_{RW}^2 + h_{RW}^2) & 0\\ 0 & 0 & \frac{m_{RW}}{12}(3r_{RW}^2 + h_{RW}^2)\end{bmatrix} \tag{22}$$

The proportional-derivative (PD) control law is shown in Equation (23) which has the current ($\vec{\epsilon}_i$) and desired ($\vec{\epsilon}_{desired}$) Euler angles (321 set), with the current ($\vec{\omega}_i$) and desired ($\vec{\omega}_{desired}$) angular velocity. The desired

torque ($\vec{M}_{desired}$) is influenced by the proportional (k_p) and derivative (k_d) gains.

$$\vec{M}_{desired} = -k_p(\vec{\epsilon}_i - \vec{\epsilon}_{desired}) - k_d(\vec{\omega}_i - \vec{\omega}_{desired}) \quad (23)$$

Through the process in [16], the inertia of the reaction wheels ($\boldsymbol{J}$), helps to find the desired angular acceleration of the reaction wheels.

$$\vec{\alpha} = \boldsymbol{J}^T(\boldsymbol{J}\boldsymbol{J}^T)^{-1}\vec{M}_{desired} \quad (24)$$

The angular acceleration is limited based on saturation conditions, which leads to finding the true torque in Equation (25), which refers to the direction of the reaction wheels in the body frame [16].

$$\vec{M}_{desired} = \sum_{i=1}^{NR} I_{Ri}^{B}\alpha_{Ri}\hat{n}_{Ri} \quad (25)$$

2.3 Wahba

For the CubeSat to know where to point in space it must be able to predict its current orientation with the sensor data. This is calculated using Equation (26), known as Wahba's problem [14].

$$J([\bar{B}N]) = \frac{1}{2}\sum_{k=1}^{N} \boldsymbol{w}_k \left| {}^{\mathcal{B}}\hat{\boldsymbol{v}}_k - [\bar{B}N]\,{}^{\mathcal{N}}\hat{\boldsymbol{v}}_k \right|^2 \quad (26)$$

The equation uses the sensor weights of Table 2 which depend on the sensor accuracy and is then related to a range of 0 to 1.

Table 2. Weights in relation to the sensor measurements onboard the CubeSat [17][18].

Sensor	Accuracy	'FOV'	w_k
Magnetometer	< 50nT	25,000 nT	0.556
Sun/Nadir Sensor	< 0.2°	180°	1
Coarse Sun Sensor	< 10°	180°	0.02

Wahba's problem takes in known readings from the Sun, Magnetometer, and Nadir sensor on board the CubeSat to then estimate the orientation. The sun measurement uses the ephemerides of the planets, the magnetometer is taken from the IGRF model in MATLAB [19], and the Nadir sensor reverses the position vector (Equation (27)) from the simulation.

$${}^{\mathcal{N}}\hat{\boldsymbol{r}}_{measured} = -{}^{\mathcal{N}}\hat{\boldsymbol{r}}_{position} \quad (27)$$

2.3.1 Error Model

The ADCS platform on the CubeSat is taken from [17], which has the errors of the Sun and Nadir measurement as a single angle offset [18], geometrically it is seen in Fig 6. This is applied within the body frame vector of Equation (26).

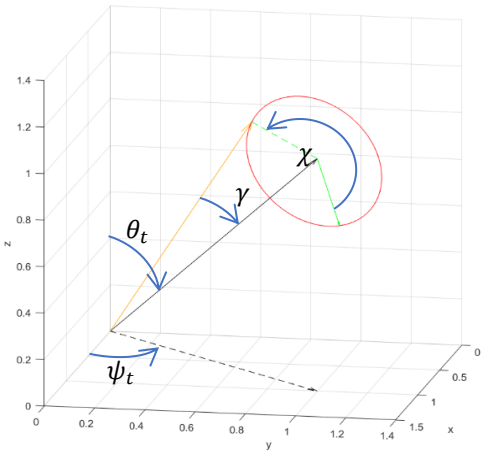

Fig 6. Angle Geometry of the Error Model, Solid Black line – measured data, Orange line – Output Error data, Red line – bounding limit

In Fig 6 the true black line is taken directly from the environment data and is related to spherical coordinates. When the single angle γ offset and χ (chosen randomly) error is applied, the orange line is obtained. This is shown in Equation (28).

$$\begin{aligned}\hat{\boldsymbol{x}}_{em} = &\left(c_{(\gamma\, s_\chi)}s_{(\theta_t+\gamma\, c_\chi)}c_{\psi_t} - s_{(\gamma\, s_\chi)}s_{\psi_t}\right)\boldsymbol{i} \\ &+\left(c_{(\gamma\, s_\chi)}s_{(\theta_t+\gamma\, c_\chi)}s_{\psi_t} + s_{(\gamma\, s_\chi)}c_{\psi_t}\right)\boldsymbol{j} \\ &+\left(c_{(\gamma\, s_\chi)}c_{(\theta_t+\gamma\, c_\chi)}\right)\boldsymbol{k}\end{aligned} \quad (28)$$

Currently the same error is applied through the orbit; it is suggested to update the error based on environment. This would mean changing the value of γ depending on the orientation of the CubeSat. The Nadir sensor won't be able to determine its position accurately if pointing away from Earth. Similarly, the sun sensor won't be as accurate if it is in eclipse or is shadowed by the solar panels of the CubeSat. Further investigation is recommended for determining line of sight with the sensors and the required environment data.

2.3.2 Devenport-q

This method solves Wahba's problem through the use of an eigen value-eigen vector problem (Equation (29)), where the resulting estimated attitude is from the largest eigen value (λ). From the derivation [14], the largest eigen value minimises the error, meaning that $[\bar{B}N]$of Equation (26) will be better approximated.

$$[K]\bar{\boldsymbol{q}} = \lambda\bar{\boldsymbol{q}} \tag{29}$$

The components to solve Equation (29) are shown in Equation (30) - (34), where I_{3x3} is a 3x3 identity matrix.

$$[K] = \begin{bmatrix} \sigma & Z^T \\ Z & S - \sigma I_{3x3} \end{bmatrix} \tag{30}$$

$$Z = \begin{bmatrix} B_{23} - B_{32} \\ B_{31} - B_{13} \\ B_{12} - B_{21} \end{bmatrix} \tag{31}$$

$$S = B + B^T \tag{32}$$

$$\sigma = tr(B) \tag{33}$$

$$B = \sum_{k=1}^{N} \boldsymbol{w}_k \, {}^{\mathcal{B}}\hat{\boldsymbol{v}}_k \, {}^{\mathcal{N}}\hat{\boldsymbol{v}}_k^T \tag{34}$$

2.3.3 *QUEST*

This method solves the eigen value-eigen vector problem (Equation (29)) with a Newton Raphson method. The first iteration is the sum of the weights applied to each measurement (Equation (35)), which is then iterated through Equation (36) [14], until a maximum is reached.

$$\lambda_0 = \sum_{k=1}^{N} w_k \tag{35}$$

$$\lambda_i = \lambda_{i-1} - \frac{f(\lambda_{i-1})}{f'(\lambda_{i-1})} \tag{36}$$

Equation (37) is taken from [14] and Equation (38) is found using Jacobi's Formula [20].

$$f(\lambda) = \det([K] - \lambda\,[I_{4*4}]) \tag{37}$$

$$f'(\lambda) = f(\lambda) * tr\left(\left[-([K] - \lambda\,[I_{4*4}])[I_{4*4}]\right]^{-1}\right) \tag{38}$$

When the max eigenvalue is reached, it is related to a CRP (Equation (39) [14]) and is converted to the relevant quaternion orientation (Equation (13)).

$$\bar{\boldsymbol{\beta}} = \left((\lambda_{max} + \sigma)[I_{3x3}] - S\right)^{-1} Z \tag{39}$$

It is expected that Devenport-q is the more reliable method, however QUEST is anticipated to be faster, and is implemented more regularly on spacecraft during operation . Despite this, timings have not been compared for the methods and this report is limited to the accuracy of the determination. The QUEST method has an inverse in matrix form which can slow down the algorithm (Equation (39)). If this is removed then the method could be faster. Alternatively other methods such as ESOQ and OLAE could be included, which would enable a more rigorous timing comparison [14].

3 Results

The results consist of visualisation, mechanics verification, detumble with actuators, and idealized tracking control. This gives an overview for the core mechanics and user functionality related to the IST. The power profile has been developed but is not integrated, and the thermal model is tasked for future work.

3.1 *Visualisation*

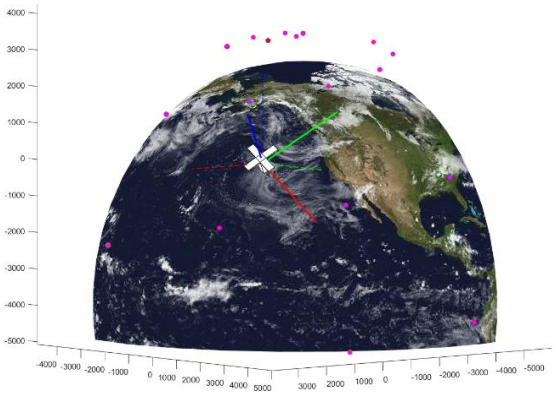

Fig 7. Orientation Visualisation: r-g-b Dotted Lines are Inertial Frame x-y-z Axis; r-g-b Solid Lines are Track Frame x-y-z axis; Purple Markers are Iridium Satellites

3.2 *Mechanics Verification*

Table 3. Initialisation parameters for mechanics verification simulation (Example 7.1 of [15])

Name	Symbol	Value	Unit
Satellite Mass properties			
x-axis moment of Inertia	I_{xx}	10 000	$kg\ m^2$
y-axis moment of Inertia	I_{yy}	9 000	$kg\ m^2$
z-axis moment of Inertia	I_{zz}	12 000	$kg\ m^2$
Initial Orientation properties			
Initial quaternion orientation	$\boldsymbol{q}(t_0)$	$\begin{bmatrix} 0.6853 \\ 0.6953 \\ 0.1531 \\ 0.1531 \end{bmatrix}$	1234
Initial rotation	$\boldsymbol{\omega}(t_0)$	$\begin{bmatrix} 0.5300 \\ 0.5300 \\ 0.0530 \end{bmatrix}$	$\frac{\circ}{s}$
Controller Parameters			
Commanded quaternion orientation	$\boldsymbol{q}_c$	$\begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}$	1234
Proportional Gain	k_p	50	-
Derivative Gain	k_d	500	-

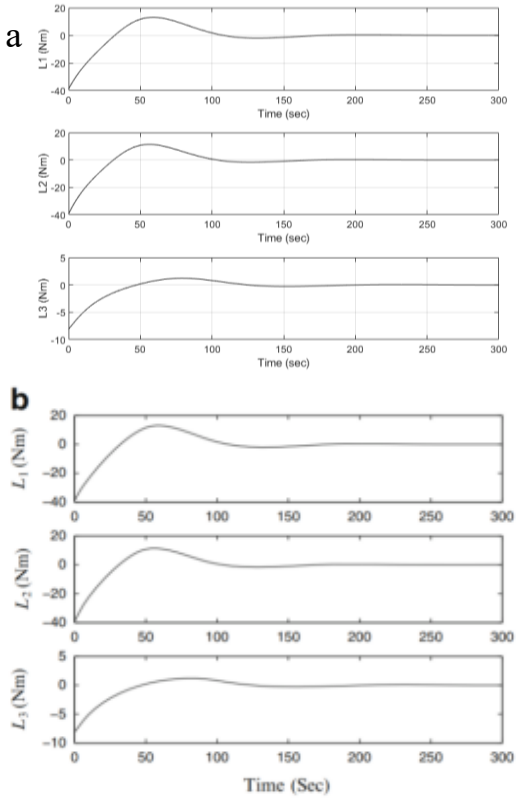

Fig 8. Torque Applied by the Satellite for the Initialisation in Table 3, a – STRATHcube Software Results, b – External Results [15].

Detumble

Table 4. Initialisation for the Detumble Simulation

Name	Symbol	Value	Unit
CubeSat mass properties			
Moment of Inertia – x	I_{xx}	0.00962	kg.m^2
Moment of Inertia – y	I_{yy}	0.0102	kg.m^2
Moment of Inertia – z	I_{zz}	0.00445	kg.m^2
RW CoG offset - x	x_{cm}	0.00127	m
RW CoG offset - y	y_{cm}	0.00029	m
RW CoG offset - z	z_{cm}	0.00798	m
Mass of Satellite	m	2.41486	kg
Reaction Wheel Properties			
Mass of Wheel	m_{rw}	0.06	kg
Height of Wheel	h_{rw}	26.1	mm
Radius of Wheel	r_{rw}	14	mm
RW Angular velocity	$\boldsymbol{\omega}_{rw}$	0	rad/s
Other Properties			
Sat. x angular velocity	p_0	0.8	rad/s
Sat. y angular velocity	q_0	-0.2	rad/s
Sat. z angular velocity	r_0	0.3	rad/s
Commanded quaternion	$\mathbf{q_c}$	$[1\ 0\ 0\ 0]^T$	0123

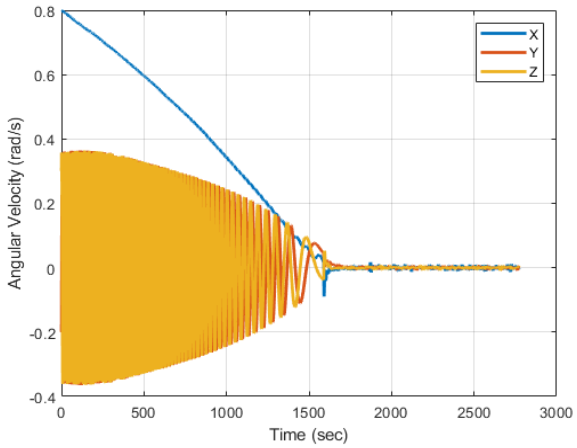

Fig 9. Angular Velocity of CubeSat During Detumble

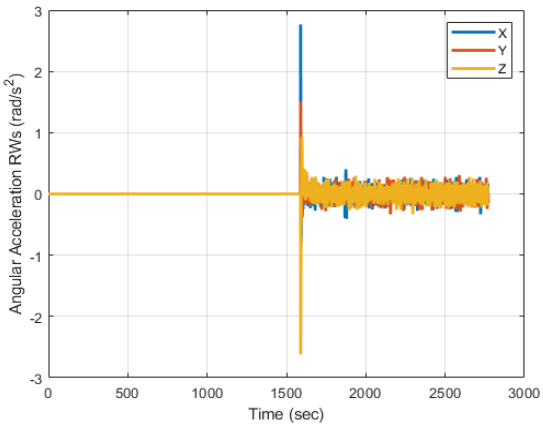

Fig 10. Angular Acceleration of CubeSat Reaction Wheels

Idealised tracking control

Table 5. Initialisation for the Idealised Tracking Control Simulation (Other values are from Table 4)

Name	Symbol	Value	Unit
Other Properties			
Sat. x angular velocity	p_0	0.8	rad/s
Sat. y angular velocity	q_0	1.09	rad/s
Sat. z angular velocity	r_0	-0.8	rad/s
Commanded quaternion	$\mathbf{q_c}$	Track Frame	-

The reason this section has idealised control is linked to the control law. This uses Equation (20), which is the nonlinear controller, and is not related to the dynamics of the reaction wheel and magnetorquer. This created a stepping stone for finding bugs in the software before further pushing forward with actuator control for the full lifecycle.

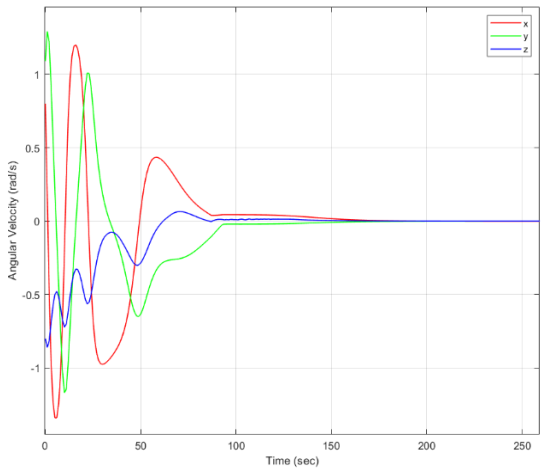

Fig 11. Angular Velocity of STRATHcube During the Idealised Tracking Control Simulation

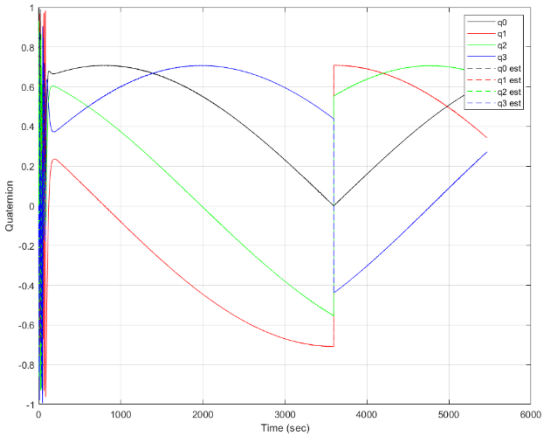

Fig 12. Quaternion Orientation of the CubeSat for the Simulation ran With Initialisation in Table 5

3.3 Determination Comparison

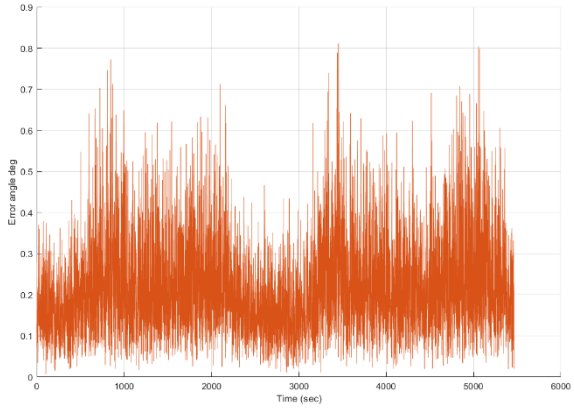

Fig 13. QUEST method error angle results of solving Wahba's problem for Table 5 Initialisation

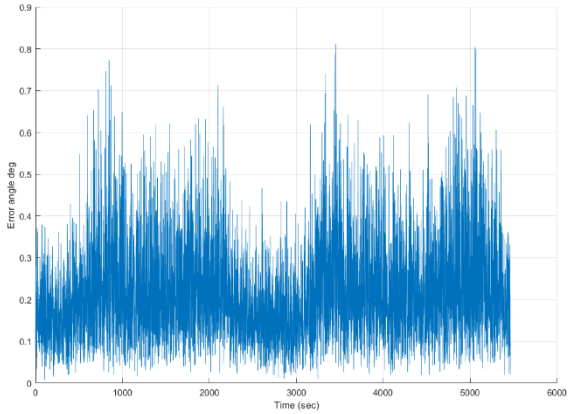

Fig 14. Devenport-q error angle results of solving Wahba's problem for Table 5 initialisation

Table 6. Determination statistics comparison of solving Wahba's problem

Error angle comparison	Devenport-q method (°)	QUEST (°)
Standard Deviation	0.1201	0.1201
Mean	0.2147	0.2147
Max	0.8116	0.8116

4 Discussion

Within the visualisation the IST developed from a 2D static plot in polar coordinates to an animated 3D cartesian simulation of the core mechanics. In Fig 7 the CubeSat is centred at (0,0,0) thoughout the simulation, which enables easier understanding of the pointing direction of the CubeSat. It is a pseudo orbit frame as only translation is accounted for, not rotation, and has the same axis as the inertial frame. Further development can include making a Graphical User Interface (GUI), so that the IST is more user friendly.

The Iridium satellites position is based on the orbital parameters a (777.5 km), e (0), i (90°), Ω (0), ω (0) and θ (0), which have 6 evenly spaced planes of 11 satellites. The Iridium constellation does not have any perturbations applied and is considered ideal. Real ephemerides are recommended for further analysis. Unless stated otherwise the CubeSat has orbital parameters of altitude (401km), e (0.01), i (51.6°), Ω (0), ω (0) and θ (0) when initialised. This is due to the ISS as the baseline orbit, which was related to ESA FYS!

For the IST to accurately confirm potential hardware, the software needs to be verified against external results. From [15], a simulation is completed using a linear controller (Equation (16)), which can be used as a comparison for the fundamental mechanics. Due to a generic control example, it is not orbit specific.

The initialisation is seen in Table 3, which highlights that the simulation is not tracking a slew frame (as is the

case for STRATHcube) but is a constant unit quaternion. The inertia matrix has zeros on the diagonals.

From the results in Fig 8a and Fig 8b it is clear the output is qualitatively identical. The shape and numerical magnitude of the graphs is very similar; however it has not been compared quantitively. Despite this it is thought the STRATHcube software is mechanically accurate due to the simulation being an accurate qualitative match.

The detumble simulation includes using the magnetorquer to detumble, then using the reaction wheels to control the CubeSat to the constant unit quaternion orientation with an angular velocity of zero (Fig 9, Fig 10). Due to simulation issues the CubeSat could not be maintained at a slew frame with actuators, and only a constant frame could be maintained. This simulation also uses Equation (23), which has Euler angles within it and potentially causes some of the problems. Other reasoning can be due to the gain selection which is currently manually estimated.

The CubeSat initialises with a total rotation of 0.877 rad/s (50.3 deg/s) and completes the detumble (reaction wheels turn on) at roughly 1600s. This is using the CubeADCS platform from [17]. There is also a large kick when the reaction wheels turn on (Fig 10), potentially meaning a relatively large proportional gain causing overshoot. Despite the results being able to track a stationary frame, the actuators are currently not verified externally, which is recommended for future work alongside the slew frame developments.

In idealised tracking the initialised parameters are very similar to the detumble simulation; however it is initialised with a larger value of angular velocity for a more rigorous detumble, as seen in Table 5. The detumble is substantially faster than the reaction wheels, where it finishes at ~100s (reaction wheels would turn on when < 0.1rad/s) in Fig 11. Once the detumble is complete, the CubeSat then positions itself for the track frame orientation (Fig 7, Fig 12). This then has the quaternion orientation varying throughout its orbit. The graph appears discontinuous at roughly 3500s, which is due to the quaternions representing the short rotation. This allows a comparison with the estimated quaternion from determination. If the short rotation is not accounted for the estimated quaternion can oscillate between the short and long rotation, creating a noisy and unclear appearance.

The determination is completed using the two methods described in 2.3.2 Devenport-q and 2.3.3 QUEST. By using two separate methods, it is thought to verify the output by comparing the separate methods. This gives a level of confidence for using the software and further parts verification. Currently it assumes the maximum error limits applied to the environment data does not change, however in practice this is not true.

From the hardware description only a single error angle is given, however with the Sun Sensor, it is unlikely to be operating with the same accuracy for eclipse and direct sunlight which also needs to consider shadowing of the body and solar panels.

The QUEST method in Fig 13 and Devenport-q method in Fig 14 show a very similar qualitative resemblence. The graphs represent the PRV error angle between the true quaternion orientation, and the estimated quaternion orientation. The reason for the shapes of the graph relate to how the environment data interacts. As the magnetic field, zenith vector, and sun vector become near parallel the orientation becomes more uncertain therefore there are sections of larger inaccuracies.

The data is not only qualitatively similar but quantitatively identical. By comparing each value the error angle data (Fig 13, Fig 14) ends up as an exact match to 5 significant figures, where some statistics are represented in Table 6 which shows a quantitative comparison. The error angle is related to a Principal Rotation Vector, which enables comparing two separate frames, but with a single angle. It highlights that the error for determining the pointing is relatively low for the mission requirements. This is before filtering has been applied, which could reduce the error angle further. However, the max error angle is expected to increase once the eclipse is accounted for, as this will cause one of the main sensors to be erroneous.

5 Conclusion

STRATHcube presents a novel solution to combat debris in LEO through its two payloads, the first providing detection, tracking, and cataloguing of debris capabilities through PBR, and the second generating data in the aid for the D4D program by collecting structural and thermal behaviour data of the spacecraft as it re-enters the atmosphere.

Throughout the project, development has consisted of software and hardware. The structures have been updated, including modal analysis of the spacecraft, and link budgets have confirmed the feasibility of transmission during the secondary payload. The IST has managed to implement visualisation, orbital mechanics, and ADCS simulations, whilst having a basic verification process. The power has been developed but is not integrated with the IST. Several points are still to be met with recommendations for further development:

- Thermal model
- Integrate power
- ADCS perturbations
- Verify actuators
- Filtering
- Account for environment with the sensor errors (Sun sensor in eclipse)
- GUI

Acknowledgements

This works was sponsored by the University of Strathclyde Alumni Fund, Royal Aeronautical Society, and the Institute of Mechanical Engineers. The authors would like to also thank the individuals of Andrew Gibb, Ciaran Jenkins, David Peacock, David Polley, Iain Hall, and Maria Kardassi for their support and guidance. The University of Strathclyde Mechanical and Aerospace Engineering (MAE) Department and the Aerospace Centre for Excellence (ACE) has enabled substantial development of this project, and we are grateful for their help and support.

6 References

[1] "Space Environment Statistics · Space Debris User Portal." https://sdup.esoc.esa.int/discosweb/statistics/ (accessed Aug. 23, 2022).

[2] D. J. Kessler, N. L. Johnson, J.-C. Liou, and M. Matney, "The Kessler Syndrome: Implications to Future Space operations".

[3] "ESA - Using lasers to track space debris." https://www.esa.int/Enabling_Support/Space_Engineering_Technology/Shaping_the_Future/Using_lasers_to_track_space_debris (accessed Sep. 01, 2022).

[4] D. Mehrholz, L. Leushacke, W. Flury, R. Jehn, H. Klinkrad, and M. Landgraf, "Detecting, Tracking and Imaging Space Debris," 2002.

[5] "ESA - Mitigation scenarios: Graveyard orbit 300 km above GEO." https://www.esa.int/ESA_Multimedia/Images/2008/03/Mitigation_scenarios_Graveyard_orbit_300_km_above_GEO (accessed Sep. 01, 2022).

[6] L. Anselmo, "RE-ENTRY PREDICTIONS FOR UNCONTROLLED SATELLITES: RESULTS AND CHALLENGES".

[7] C. Jenkins, R. Hope, A. R. Wilson, and A. Berquand, "STRATHcube : The design of a student CubeSat using concurrent engineering methods," *Uniwersytet śląski*, pp. 343–354, Oct. 2020, doi: 10.2/JQUERY.MIN.JS.

[8] A. R. Persico, P. Kirkland, C. Clemente, J. J. Soraghan, and M. Vasile, "CubeSat-based passive bistatic radar for space situational awareness: A feasibility study," *IEEE Trans Aerosp Electron Syst*, vol. 55, no. 1, pp. 476–485, Feb. 2019, doi: 10.1109/TAES.2018.2848340.

[9] J. Graham, L. Creed, C. Jenkins, A. R. Wilson, and M. Vasile, "The design of a fragmentation experiment for a CubeSat during atmospheric re-entry," *Uniwersytet śląski*, pp. 343–354, Oct. 2021, doi: 10.2/JQUERY.MIN.JS.

[10] S. McKenzie-Picot and P. Gavigan, "Communication Link Analysis for a Nanosatellite Constellation in Low Earth Orbit ," 2013. https://publications.gc.ca/collections/collection_2016/rddc-drdc/D68-6-068-2013-eng.pdf (accessed Sep. 01, 2022).

[11] S. Dey Asst Professor, D. Kumar Mohapatra AsstProfessor, and S. Devashish Rajanandini Pummy Archana, "An Approach to calculate the Performance and Link Budget of LEO Satellite (Iridium) For Communication Operated at frequency Range (1650-1550) MHz".

[12] M. Mehdizadeh Khalsaraei, "Positivity of an explicit Runge–Kutta method," *Ain Shams Engineering Journal*, vol. 6, no. 4, pp. 1217–1223, Dec. 2015, doi: 10.1016/J.ASEJ.2015.05.018.

[13] J. Diebel, "Representing Attitude: Euler Angles, Unit Quaternions, and Rotation Vectors," 2006.

[14] H. Schaub and L. J. Junkins, *Analytical Mechanics of Space Systems - Hanspeter Schaub, John L. Junkins - Google Books*, 4th ed. 2018.

[15] F. Landis Markley and J. L. Crassidis, *Space Technology Library Fundamentals of Spacecraft Attitude Determination and Control*. Accessed: Sep. 01, 2022. [Online]. Available: http://www.springer.com/series/6575

[16] "LaTeX/aerospace_mechanics.pdf at master · cmontalvo251/LaTeX · GitHub." https://github.com/cmontalvo251/LaTeX/blob/master/Aerospace_Mechanics/aerospace_mechanics.pdf (accessed Sep. 01, 2022).

[17] "CubeADCS Gen 1 | CubeSpace Satellite Systems." https://www.cubespace.co.za/products/gen-1/integrated-adcs/cubeadcs/ (accessed Sep. 01, 2022).

[18] "CubeSense Gen 1 | CubeSpace Satellite Systems." https://www.cubespace.co.za/products/gen-1/sensors/cubesense/ (accessed Sep. 01, 2022).

[19] "International Geomagnetic Reference Field (IGRF) Model - File Exchange - MATLAB Central." https://uk.mathworks.com/matlabcentral/fileexchange/34388-international-geomagnetic-reference-field-igrf-model (accessed Sep. 01, 2022).

[20] J.R Magnus and H. Neudecker, "Matrix Differential Calculus with Applications in Statistics and Econometrics," John Wiley & Sons, 2019.