IAC-12 C1.8.4

ROBUSTNESS ANALYSES OF ATTITUDE SLEW MANOEUVRES FOR SPINNING PENETRATOR SPACECRAFT

Robin Raus*
r.a.raus+iac2012@gmail.com

Yang Gao†*
yang.gao@surrey.ac.uk

Yunhua Wu*
yunhua.wu@surrey.ac.uk

Mark Watt‡
mark.watt@astrium.eads.net

Spin-stabilisation of spacecraft is considered a simple and relatively effective way of ensuring passive attitude stability and has therefore seen wide application. However, very few spin-stabilised spacecraft are prolate (long and slender), likely due to the fact that spin-stabilising a prolate spacecraft is dynamically unstable in the presence of energy dissipation due to e.g. fuel sloshing. Recently, interest in prolate spinning spacecraft has been rekindled because of mission proposals utilising a penetrator concept for exploration of atmosphere-less celestial bodies such as the Moon. A penetrator is a spacecraft/probe that is designed to impact the body and survive, thereby burying itself in the upper layer, yielding an attractive vantage point for subsurface sensors such as seismometers. A high-level analysis and comparison of attitude slew algorithms using only one thruster has been performed previously in a paper presented at IAC 2010.

This paper aims to give a summary of the comprehensive robustness analysis of seven different single-thruster slew manoeuvres for a rigid, highly prolate spin-stabilised axisymmetric penetrator spacecraft as designed in the MoonLITE mission proposal. This analysis has been performed using both analytical solutions and simulations, except for the Rhumb line algorithm where only simulations could be used. Five different parameters (spin axis moment of inertia, transverse moment of inertia, spin rate, thruster-on time, asymmetry) were perturbed one by one; the impact of different design parameters (moments of inertia, spin rate) on the slew error has also been investigated.

Comparing the analytical solutions with the simulation results it appears that the analytical solutions are quite accurate and require much less computation time. A further advantage of using analytical solutions is that they can be inverted and used to e.g. determine the minimum spin rate given a certain maximum slew time. With respect to the comparison between the slew algorithms, all slews derived from the Half-cone slew algorithm turn out to be highly susceptible to errors in spin rate for a spacecraft of this type; errors in moments of inertia have a much lower impact on the final accuracy. Of the Pulse-train algorithms, the Rhumb line is relatively impervious to a spin rate disturbance but the Spin-Synch is not. These results will benefit the design of a penetrator mission and spacecraft as they allow the designer to improve estimates of a mission's feasibility.

I INTRODUCTION

One proposed mission concept for subsurface investigations into celestial bodies is to use a penetrator-type spacecraft, a cylindrical projectile that is designed to survive an impact with a celestial body, burying itself into the subsurface. From this vantage point on-board sensors such as seismometers and thermometers can take measurements of the inner layer. Two proposed missions using this concept are the Japanese mission Lunar-A (unfortunately cancelled according to Shiraishi et al. [1]) and the British MoonLITE mission concept [2], both intended to study the lunar subsurface. Due to the lack of atmosphere on the Moon or other potential targets such as asteroids, attitude stabilisation of these penetrator spacecraft is usually performed by means of spin stabilisation as this is a relatively simple and low-cost means of stabilisation. The nature of the mission (i.e. tip-first impact) suggests spin stabilisation about a minor axis of inertia.

Previous papers investigated the motion of an axisymmetric, prolate, spinning semi-rigid penetrator

*Surrey Space Centre, United Kingdom
†Corresponding author
‡EADS Astrium Ltd, United Kingdom

spacecraft (Raus et al. [3] and Raus et al. [4]) and made a trade-off of slew algorithms [5]. This paper studies the robustness of each of the identified slew algorithms against perturbations in input parameters.

I.I Paper Structure

After this introduction, the next section will briefly introduce the assumptions made for the analyses. This is followed by a section on the analytical derivation of the methods used in the robustness analyses. The following section gives a selection of results to highlight the varied responses. The penultimate section summarises and compares the analysis results and the last section gives the conclusions of this paper.

II ASSUMPTIONS

The following assumptions are made in general:

- There are no external nor internal disturbance torques.
- The spacecraft is considered as a rigid body.
- The spacecraft is axisymmetric; its symmetry axis is defined as the body Z-axis.
- The initial condition is a pure spin around the spacecraft's symmetry axis.
- The thruster is ideal, meaning:
  - Torque perfectly aligned with the body Y-axis;
  - No transient behaviour during on/off cycling (perfect step response);
  - 'Impulsive shot' thrust pulse: the angular impulse is delivered instantaneously, similar to the impulsive shot-approach in orbit dynamics, where it is assumed the ΔV for one orbit change is delivered in one instant.

All the slew algorithms discussed in this section are described in Raus et al. [3], Raus et al. [4] and Wu et al. [5].

III GEOMETRIC APPROACH

III.I Half-cone Slew

For the purpose of this geometric analysis, the Half-cone slew (HC) is subdivided in three phases, determined by the angular momentum vector and illustrated in Figure 1. The Spacecraft-Fixed Body frame (SFB) is coincident with the Reference Inertial frame (RI) at time $t = 0$.

1. Initial phase: immediately before the initiation torque pulse, when $\boldsymbol{H} = \boldsymbol{H}_0$;
2. Intermediate phase: $\boldsymbol{H} = \boldsymbol{H}_{int}$ and the spacecraft is precessing around this axis, in-between the two torque pulses;
3. Final phase: $\boldsymbol{H} = \boldsymbol{H}_f$, immediately after the cancellation torque pulse.

Due to the assumptions that no disturbance torques are present and that the angular impulse is instantaneous, the angular momentum vector is also the Euler eigenaxis for the rotation in each phase. With this fixed eigenaxis, the kinematic equations for these phases can be set up using quaternions. The rotation starting from RI around $\boldsymbol{H}_0$, between $t = 0$ and the initiation pulse, is described by $\overline{\boldsymbol{q}}_{\mathrm{I}}$ in Equation (48). The rotation of the SFB Z-axis between the end of the initiation pulse and the start of the cancellation pulse is described by $\overline{\boldsymbol{q}}_{\mathrm{II}}$ in Equation (49), relative to the attitude at the end of the initiation pulse; $\overline{\boldsymbol{q}}_{21}$ is the corresponding absolute attitude relative to RI, defined in Equation (29). And finally, $\overline{\boldsymbol{q}}_{\mathrm{III}}$ is the rotation of the SFB frame around its Z-axis relative to the Z-H plane, defined in Equation (50), and $\overline{\boldsymbol{q}}_{321}$ is defined in Equation (32) and is its corresponding absolute attitude relative to RI.

The equations for the angular momenta are given below:

$$\boldsymbol{H}_0 = I\boldsymbol{\omega}_0 \tag{1}$$

$$\begin{aligned}\boldsymbol{H}_{int} &= \boldsymbol{H}_0 + \boldsymbol{T}t_{fd}\Big|_{init} \\ &= \boldsymbol{H}_0 + [\overline{\boldsymbol{q}}_{\mathrm{I}}]\boldsymbol{T}_b t_{fd}\end{aligned} \tag{2}$$

$$\begin{aligned}\boldsymbol{H}_f &= \boldsymbol{H}_{int} + \boldsymbol{T}t_{fd}\Big|_{cancell} \\ &= \boldsymbol{H}_{int} + [\overline{\boldsymbol{q}}_{\mathrm{III}} \otimes \overline{\boldsymbol{q}}_{\mathrm{II}} \otimes \overline{\boldsymbol{q}}_{\mathrm{I}}]\,\boldsymbol{T}_b t_{fd} \\ &= \boldsymbol{H}_{int} + [\overline{\boldsymbol{q}}_{321}]\,\boldsymbol{T}_b t_{fd}\end{aligned} \tag{3}$$

The angular momentum vector $\boldsymbol{H}$ is measured in the RI frame while the torque vector $\boldsymbol{T}_b$ is given in the

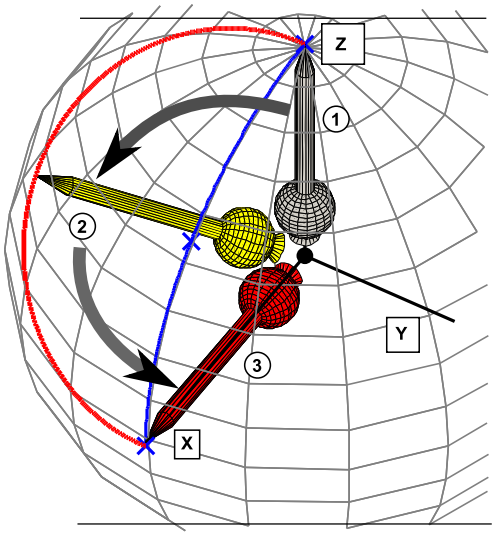

Fig. 1: Phases in Half-cone slew

SFB frame, hence the necessity to convert the angular impulse vector $\boldsymbol{T}_b t_{fd}$ into RI frame at both instants. The square brackets around the quaternions indicate they are transformed into their respective 3x3 Direction Cosine Matrix (DCM) using Equation (4), which can be found in Sidi [6], Equation A.4.15 in Appendix A.4.3.

$$[\overline{\boldsymbol{q}}] = DCM(\overline{\boldsymbol{q}}) = \begin{bmatrix} DCM_{11} & DCM_{12} & DCM_{13} \\ DCM_{21} & DCM_{22} & DCM_{23} \\ DCM_{31} & DCM_{32} & DCM_{33} \end{bmatrix} \tag{4}$$

$$DCM_{11} = q_1^2 - q_2^2 - q_3^2 + q_4^2 \tag{5}$$
$$DCM_{12} = 2(q_1 q_2 + q_3 q_4) \tag{6}$$
$$DCM_{13} = 2(q_1 q_3 - q_2 q_4) \tag{7}$$
$$DCM_{21} = 2(q_1 q_2 - q_3 q_4) \tag{8}$$
$$DCM_{22} = -q_1^2 + q_2^2 - q_3^2 + q_4^2 \tag{9}$$
$$DCM_{23} = 2(q_2 q_3 + q_1 q_4) \tag{10}$$
$$DCM_{31} = 2(q_1 q_3 + q_2 q_4) \tag{11}$$
$$DCM_{32} = 2(q_2 q_3 - q_1 q_4) \tag{12}$$
$$DCM_{33} = -q_1^2 - q_2^2 + q_3^2 + q_4^2 \tag{13}$$

III.II Dual-cone Slew

The Dual-cone slew (DC) is a concatenation of two Half-cone slews, where the azimuth angle α_{int}, controlling the direction of the second HC, is modified such that greater flexibility in potential slew angles is attained. The calculations behind this can be found in Raus et al. [4]. A second, less critical modification is that the nutation angle θ for each Half-cone needs to be forced to the larger angle, again as specified in the Dual-cone's definition. These two modifications do not require any other input than the initial conditions, which allows the use of the Multi-cone algorithm with an extension for the Dual-cone, triggered by an optional flag in the input arguments.

III.III Sector-Arc Slew

The Sector-Arc Slew (SAS) algorithm is similar to a Half-cone, but the main difference is that the intermediate angular momentum vector $\boldsymbol{H}_{int}$ is not in the plane spanned by the initial and target spin axis attitudes. As a result, an extra degree of freedom is obtained that greatly enhances flexibility. It is set up as described in Raus et al. [4], where both θ and k are considered inputs, and the slew angle δ is an output. This allows for easy reuse of the algorithm in the EHC algorithm, where both θ and k can be pre-calculated.

III.IV Extended Half-cone Slew

The Extended Half-cone slew (EHC) is a concatenation of two partial Half-cones as described in Wu et al. [5]. This requires a different treatment than the Dual-cone. Fortunately, the SAS algorithm can be reused such that the EHC algorithm resembles the Dual-cone algorithm but concatenates two SAS instead of two Half-cones, with a modification for the azimuth angle α of each SAS.

IV TAYLOR SERIES APPROXIMATION

In this section the sensitivity of the Half-cone's constitutive equations to errors in inertia, spin rate, torque is explored by setting up a Taylor series approximation around their nominal values. In theory a similar approach can be taken for the other Half-cone derived slew algorithms, but in practice this turned out to be too complex, especially in light of the low penalty in calculation time for using the relatively simple geometrical analysis as explained in Section III.

We assume the target nutation angle θ (the subscript 't' is dropped for clarity) is a pre-defined constant given by the target slew angle δ: $\theta = \delta/2$. Instead of θ the HC algorithm will have to calculate a

corresponding k value, which is depending on several parameters.

IV.I Summary

In this section the results of the derivations of first and second order partial derivatives for the second-order Taylor series approximation of the following three Half-cone quantities will be briefly summarised as a function of I_z, I_t and ω_z:

1. Intermediate angular momentum vector $\boldsymbol{H}_{int}$
2. Final angular momentum vector $\boldsymbol{H}_f$
3. Final Z-axis attitude $\boldsymbol{Z}_f$

These will be briefly introduced here with the equation numbers used in their respective derivations.

For $\boldsymbol{H}_{int}$:

$$\frac{\partial \boldsymbol{H}_{int}}{\partial I_z} = \boldsymbol{\omega}_z \tag{38}$$

$$\frac{\partial \boldsymbol{H}_{int}}{\partial I_t} = \boldsymbol{0} \tag{39}$$

$$\frac{\partial \boldsymbol{H}_{int}}{\partial \omega_z} = I_z \hat{\boldsymbol{\omega}}_z + t_{fd} \frac{\partial [\overline{\boldsymbol{q}}_{\mathrm{I}}]}{\partial \omega_z} \boldsymbol{T}_b \tag{40}$$

$$\frac{\partial \boldsymbol{H}_{int}}{\partial t_{fd}} = [\overline{\boldsymbol{q}}_{\mathrm{I}}] \boldsymbol{T}_b \tag{41}$$

$$\frac{\partial^2 \boldsymbol{H}_{int}}{\partial I_z^2} = \boldsymbol{0} \tag{42}$$

$$\frac{\partial^2 \boldsymbol{H}_{int}}{\partial I_t^2} = \boldsymbol{0} \tag{43}$$

$$\frac{\partial^2 \boldsymbol{H}_{int}}{\partial \omega_z^2} = t_{fd} \frac{\partial^2 [\overline{\boldsymbol{q}}_{\mathrm{I}}]}{\partial \omega_z^2} \boldsymbol{T}_b \tag{44}$$

$$\frac{\partial^2 \boldsymbol{H}_{int}}{\partial t_{fd}^2} = \boldsymbol{0} \tag{45}$$

where:

- $\frac{\partial [\overline{\boldsymbol{q}}_{\mathrm{I}}]}{\partial \omega_z} = M\left(\overline{\boldsymbol{q}}_{\mathrm{I}}, \frac{\partial \overline{\boldsymbol{q}}_{\mathrm{I}}}{\partial \xi}\right)$ is calculated by using equations (27), (48) and (51).
- $\frac{\partial^2 [\overline{\boldsymbol{q}}_{\mathrm{I}}]}{\partial \omega_z^2}$ is calculated by using equations (28), (48), (51) and (52).

The derivation of $\boldsymbol{H}_f$ starts from Equation (47) and is simplified using Equation (37):

$$\begin{aligned} \boldsymbol{H}_f =& \boldsymbol{H}_{int} + [\overline{\boldsymbol{q}}_{321}] \boldsymbol{T}_b t_{fd} \\ =& \boldsymbol{H}_0 + [\overline{\boldsymbol{q}}_{\mathrm{I}}] \boldsymbol{T}_b t_{fd} + [\overline{\boldsymbol{q}}_{321}] \boldsymbol{T}_b t_{fd} \\ =& \boldsymbol{H}_0 + ([\overline{\boldsymbol{q}}_{\mathrm{I}}] + [\overline{\boldsymbol{q}}_{321}]) \boldsymbol{T}_b t_{fd} \Leftrightarrow \end{aligned} \tag{14}$$

$$\begin{aligned} \frac{\partial \boldsymbol{H}_f}{\partial \xi} =& \frac{\partial \boldsymbol{H}_0}{\partial \xi} + \left(\frac{\partial [\overline{\boldsymbol{q}}_{\mathrm{I}}]}{\partial \xi} + \frac{\partial [\overline{\boldsymbol{q}}_{\mathrm{III}} \otimes \overline{\boldsymbol{q}}_{\mathrm{II}} \otimes \overline{\boldsymbol{q}}_{\mathrm{I}}]}{\partial \xi} \right) \boldsymbol{T}_b t_{fd} \\ =& \frac{\partial \boldsymbol{H}_0}{\partial \xi} + M\left(\overline{\boldsymbol{q}}_{\mathrm{I}}, \frac{\partial \overline{\boldsymbol{q}}_{\mathrm{I}}}{\partial \xi}\right) \boldsymbol{T}_b t_{fd} \\ &+ M\left(\overline{\boldsymbol{q}}_{321}, \frac{\partial \overline{\boldsymbol{q}}_{321}}{\partial \xi}\right) \boldsymbol{T}_b t_{fd} \end{aligned} \tag{15}$$

$$\begin{aligned} \frac{\partial^2 \boldsymbol{H}_f}{\partial \xi^2} =& \frac{\partial^2 \boldsymbol{H}_0}{\partial \xi^2} + \left(\frac{\partial^2 [\overline{\boldsymbol{q}}_{\mathrm{I}}]}{\partial \xi^2} + \frac{\partial^2 [\overline{\boldsymbol{q}}_{\mathrm{III}} \otimes \overline{\boldsymbol{q}}_{\mathrm{II}} \otimes \overline{\boldsymbol{q}}_{\mathrm{I}}]}{\partial \xi^2} \right) \boldsymbol{T}_b t_{fd} \\ =& \frac{\partial^2 \boldsymbol{H}_0}{\partial \xi^2} + M2\left(\overline{\boldsymbol{q}}_{\mathrm{I}}, \frac{\partial \overline{\boldsymbol{q}}_{\mathrm{I}}}{\partial \xi}, \frac{\partial^2 \overline{\boldsymbol{q}}_{\mathrm{I}}}{\partial \xi^2}\right) \boldsymbol{T}_b t_{fd} \\ &+ M2\left(\overline{\boldsymbol{q}}_{321}, \frac{\partial \overline{\boldsymbol{q}}_{321}}{\partial \xi}, \frac{\partial^2 \overline{\boldsymbol{q}}_{321}}{\partial \xi^2}\right) \boldsymbol{T}_b t_{fd} \end{aligned} \tag{16}$$

where $\overline{\boldsymbol{q}}_{321}$ is defined in Equation (32), $\frac{\partial \overline{\boldsymbol{q}}_{321}}{\partial \xi}$ is defined in Equation (33), and $\frac{\partial^2 \overline{\boldsymbol{q}}_{321}}{\partial \xi^2}$ is defined in Equation (34).

For $\boldsymbol{Z}_f$, from Equation (62):

$$\begin{aligned} \boldsymbol{Z}_f &= [\overline{\boldsymbol{q}}_{\mathrm{II}} \otimes \overline{\boldsymbol{q}}_{\mathrm{I}}] \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \\ &= [\overline{\boldsymbol{q}}_{21}] \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \Leftrightarrow \\ \frac{\partial \boldsymbol{Z}_f}{\partial \xi} &= M\left(\overline{\boldsymbol{q}}_{21}, \frac{\partial \overline{\boldsymbol{q}}_{21}}{\partial \xi}\right) \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \\ \frac{\partial^2 \boldsymbol{Z}_f}{\partial \xi^2} &= M2\left(\overline{\boldsymbol{q}}_{21}, \frac{\partial \overline{\boldsymbol{q}}_{21}}{\partial \xi}, \frac{\partial^2 \overline{\boldsymbol{q}}_{21}}{\partial \xi^2}\right) \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \end{aligned}$$

where $\overline{\boldsymbol{q}}_{21}$ is defined in Equation (29), $\frac{\partial \overline{\boldsymbol{q}}_{21}}{\partial \xi}$ is defined in Equation (30), and $\frac{\partial^2 \overline{\boldsymbol{q}}_{21}}{\partial \xi^2}$ is defined in Equation (31).

IV.II Preconditions

The Direction Cosine Matrix itself also needs to be derived as it is a function of its respective quaternion. Its partial derivative with respect to any variable ξ is derived from Equation (4) as follows:

$$\frac{\partial DCM(\overline{\boldsymbol{q}})}{\partial \xi} = \begin{bmatrix} M_1 & M_2 & M_3 \\ M_4 & M_5 & M_6 \\ M_7 & M_8 & M_9 \end{bmatrix} \tag{17}$$

Where:

$$M_1 = q_1\frac{\partial q_1}{\partial \xi} - q_2\frac{\partial q_2}{\partial \xi} - q_3\frac{\partial q_3}{\partial \xi} + q_4\frac{\partial q_4}{\partial \xi} \tag{18}$$

$$M_2 = q_2\frac{\partial q_1}{\partial \xi} + q_1\frac{\partial q_2}{\partial \xi} + q_4\frac{\partial q_3}{\partial \xi} + q_3\frac{\partial q_4}{\partial \xi} \tag{19}$$

$$M_3 = q_3\frac{\partial q_1}{\partial \xi} - q_4\frac{\partial q_2}{\partial \xi} + q_1\frac{\partial q_3}{\partial \xi} - q_2\frac{\partial q_4}{\partial \xi} \tag{20}$$

$$M_4 = q_2\frac{\partial q_1}{\partial \xi} + q_1\frac{\partial q_2}{\partial \xi} - q_4\frac{\partial q_3}{\partial \xi} - q_3\frac{\partial q_4}{\partial \xi} \tag{21}$$

$$M_5 = -q_1\frac{\partial q_1}{\partial \xi} + q_2\frac{\partial q_2}{\partial \xi} - q_3\frac{\partial q_3}{\partial \xi} + q_4\frac{\partial q_4}{\partial \xi} \tag{22}$$

$$M_6 = q_4\frac{\partial q_1}{\partial \xi} + q_3\frac{\partial q_2}{\partial \xi} + q_2\frac{\partial q_3}{\partial \xi} + q_1\frac{\partial q_4}{\partial \xi} \tag{23}$$

$$M_7 = q_3\frac{\partial q_1}{\partial \xi} + q_4\frac{\partial q_2}{\partial \xi} + q_1\frac{\partial q_3}{\partial \xi} + q_2\frac{\partial q_4}{\partial \xi} \tag{24}$$

$$M_8 = -q_4\frac{\partial q_1}{\partial \xi} + q_3\frac{\partial q_2}{\partial \xi} + q_2\frac{\partial q_3}{\partial \xi} - q_1\frac{\partial q_4}{\partial \xi} \tag{25}$$

$$M_9 = -q_1\frac{\partial q_1}{\partial \xi} - q_2\frac{\partial q_2}{\partial \xi} + q_3\frac{\partial q_3}{\partial \xi} + q_4\frac{\partial q_4}{\partial \xi} \tag{26}$$

Equations (18) to (26) can be written more clearly as a matrix-vector equation using the column vector $\partial\overline{\boldsymbol{q}}/\partial\xi$:

$$M\left(\overline{\boldsymbol{q}}, \frac{\partial \overline{\boldsymbol{q}}}{\partial \xi}\right) = 2\begin{bmatrix} q_1 & -q_2 & -q_3 & q_4 \\ q_2 & q_1 & q_4 & q_3 \\ q_3 & -q_4 & q_1 & -q_2 \\ q_2 & q_1 & -q_4 & -q_3 \\ -q_1 & q_2 & -q_3 & q_4 \\ q_4 & q_3 & q_2 & q_1 \\ q_3 & q_4 & q_1 & q_2 \\ -q_4 & q_3 & q_2 & -q_1 \\ -q_1 & -q_2 & q_3 & q_4 \end{bmatrix}\frac{\partial \overline{\boldsymbol{q}}}{\partial \xi} \tag{27}$$

Note that the result M from (27) is now a 9x1 column vector, but it can easily be converted back to the 3x3 matrix $\partial DCM(\overline{\boldsymbol{q}})/\partial\xi$ using Equation (17). If e.g. using MATLAB, the function `vec2mat` is perfectly suited for this.

The second-order partial derivative of the DCM will be required for a second-order approximation. Using Equation (17), we can see that this derivative can be translated to $\partial M/\partial\xi$, which is easier to derive starting from Equation (27) and resulting in Equation (28). Similar as for the first-order equation, the result $M2 = \partial M/\partial\xi$ is a 9-element column vector, which is converted to a 3x3 matrix in the same way as in Equation (17).

$$\begin{aligned} M2\left(\overline{\boldsymbol{q}}, \frac{\partial \overline{\boldsymbol{q}}}{\partial \xi}, \frac{\partial^2 \overline{\boldsymbol{q}}}{\partial \xi^2}\right) =& \frac{\partial M\left(\overline{\boldsymbol{q}}, \frac{\partial \overline{\boldsymbol{q}}}{\partial \xi}\right)}{\partial \xi} \\ =& M\left(\frac{\partial \overline{\boldsymbol{q}}}{\partial \xi}, \frac{\partial \overline{\boldsymbol{q}}}{\partial \xi}\right) + M\left(\overline{\boldsymbol{q}}, \frac{\partial^2 \overline{\boldsymbol{q}}}{\partial \xi^2}\right) \end{aligned} \tag{28}$$

Finally, to shorten certain equations, two additional quaternions $\overline{\boldsymbol{q}}_{21}$ and $\overline{\boldsymbol{q}}_{321}$ are defined as follows, with their partial derivatives:

$$\overline{\boldsymbol{q}}_{21} \equiv \overline{\boldsymbol{q}}_{\mathrm{II}} \otimes \overline{\boldsymbol{q}}_{\mathrm{I}} \Leftrightarrow \tag{29}$$

$$\frac{\partial \overline{\boldsymbol{q}}_{21}}{\partial \xi} = \frac{\partial \overline{\boldsymbol{q}}_{\mathrm{II}}}{\partial \xi} \otimes \overline{\boldsymbol{q}}_{\mathrm{I}} + \overline{\boldsymbol{q}}_{\mathrm{II}} \otimes \frac{\partial \overline{\boldsymbol{q}}_{\mathrm{I}}}{\partial \xi} \tag{30}$$

$$\frac{\partial^2 \overline{\boldsymbol{q}}_{21}}{\partial \xi^2} = \frac{\partial^2 \overline{\boldsymbol{q}}_{\mathrm{II}}}{\partial \xi^2} \otimes \overline{\boldsymbol{q}}_{\mathrm{I}} + 2\frac{\partial \overline{\boldsymbol{q}}_{\mathrm{II}}}{\partial \xi} \otimes \frac{\partial \overline{\boldsymbol{q}}_{\mathrm{I}}}{\partial \xi} + \overline{\boldsymbol{q}}_{\mathrm{II}} \otimes \frac{\partial^2 \overline{\boldsymbol{q}}_{\mathrm{I}}}{\partial \xi^2} \tag{31}$$

and

$$\overline{\boldsymbol{q}}_{321} \equiv \overline{\boldsymbol{q}}_{\mathrm{III}} \otimes \overline{\boldsymbol{q}}_{\mathrm{II}} \otimes \overline{\boldsymbol{q}}_{\mathrm{I}} = \overline{\boldsymbol{q}}_{\mathrm{III}} \otimes \overline{\boldsymbol{q}}_{21} \Leftrightarrow \tag{32}$$

$$\frac{\partial \overline{\boldsymbol{q}}_{321}}{\partial \xi} = \frac{\partial \overline{\boldsymbol{q}}_{\mathrm{III}}}{\partial \xi} \otimes \overline{\boldsymbol{q}}_{21} + \overline{\boldsymbol{q}}_{\mathrm{III}} \otimes \frac{\partial \overline{\boldsymbol{q}}_{21}}{\partial \xi} \tag{33}$$

$$\begin{aligned} \frac{\partial^2 \overline{\boldsymbol{q}}_{321}}{\partial \xi^2} =& \frac{\partial^2 \overline{\boldsymbol{q}}_{\mathrm{III}}}{\partial \xi^2} \otimes \overline{\boldsymbol{q}}_{21} \\ &+ 2\frac{\partial \overline{\boldsymbol{q}}_{\mathrm{III}}}{\partial \xi} \otimes \frac{\partial \overline{\boldsymbol{q}}_{21}}{\partial \xi} + \overline{\boldsymbol{q}}_{\mathrm{III}} \otimes \frac{\partial^2 \overline{\boldsymbol{q}}_{21}}{\partial \xi^2} \end{aligned} \tag{34}$$

where

- $\frac{\partial \overline{\boldsymbol{q}}_{\mathrm{I}}}{\partial \xi}$ will be defined in Equation (51) for ω_z and is zero otherwise;
- $\frac{\partial^2 \overline{\boldsymbol{q}}_{\mathrm{I}}}{\partial \xi^2}$ will be defined in Equation (52) for ω_z and is zero otherwise;
- $\frac{\partial \overline{\boldsymbol{q}}_{\mathrm{II}}}{\partial \xi}$ will be defined in Equation (53);

- $\frac{\partial^2 \overline{\boldsymbol{q}}_{\mathrm{II}}}{\partial \xi^2}$ is derived from Equation (53);
- $\frac{\partial \overline{\boldsymbol{q}}_{\mathrm{III}}}{\partial \xi}$ will be defined in Equation (54);
- $\frac{\partial^2 \overline{\boldsymbol{q}}_{\mathrm{III}}}{\partial \xi^2}$ will be defined in Equation (59).

The controller uses Equation (36) to calculate t_{fd} based on the nutation angle θ_C, which is in its turn calculated from the slew angle δ according to Equation (35).

$$\zeta = \text{round}\left[1/2\left(\cos(\delta/2)\frac{1-\lambda}{\lambda} - 1\right)\right]$$
$$\theta_C = \arccos\left[(1+2\zeta)\frac{\lambda}{1-\lambda}\right] \tag{35}$$
$$t_{fd} = I_z \omega_z \frac{\tan(\theta_C)}{\|\boldsymbol{T}\|} \tag{36}$$

IV.III Angular Momentum Error

For the Half-cone there are only three important points for the angular momentum vector $\boldsymbol{H}$: initial, intermediate and final position ($\boldsymbol{H}_0, \boldsymbol{H}_{int}, \boldsymbol{H}_f$ respectively). It is assumed the initial value is given at the start, so this is not affected by any parameter changes.

For $\boldsymbol{H}_{int}$:

$$\begin{aligned}\boldsymbol{H}_{int} &= \boldsymbol{H}_0 + \boldsymbol{T} t_{fd}\Big|_{init} \Leftrightarrow \\ &= \boldsymbol{H}_0 + [\overline{\boldsymbol{q}}_{\mathrm{I}}]\boldsymbol{T}_b t_{fd} \Leftrightarrow \end{aligned} \tag{37}$$
$$\begin{aligned}\frac{\partial \boldsymbol{H}_{int}}{\partial I_z} &= \frac{\partial \boldsymbol{H}_0}{\partial I_z} + [\overline{\boldsymbol{q}}_{\mathrm{I}}]\boldsymbol{T}_b \frac{\partial t_{fd}}{\partial I_z} \\ &= \boldsymbol{\omega}_z \end{aligned} \tag{38}$$
$$\frac{\partial \boldsymbol{H}_{int}}{\partial I_t} = \boldsymbol{0} \tag{39}$$
$$\begin{aligned}\frac{\partial \boldsymbol{H}_{int}}{\partial \omega_z} &= \frac{\partial \boldsymbol{H}_0}{\partial \omega_z} + \left[\boldsymbol{T}_b \frac{\partial t_{fd}}{\partial \omega_z} + t_{fd}\left(\frac{\partial [\overline{\boldsymbol{q}}_{\mathrm{I}}]}{\partial \omega_z}\boldsymbol{T}_b\right)\right] \\ &= I_z \hat{\boldsymbol{\omega}}_z + t_{fd}\frac{\partial [\overline{\boldsymbol{q}}_{\mathrm{I}}]}{\partial \omega_z}\boldsymbol{T}_b \end{aligned} \tag{40}$$
$$\frac{\partial \boldsymbol{H}_{int}}{\partial t_{fd}} = [\overline{\boldsymbol{q}}_{\mathrm{I}}]\boldsymbol{T}_b \tag{41}$$

The second-order derivatives are as follows:

$$\frac{\partial^2 \boldsymbol{H}_{int}}{\partial I_z^2} = \boldsymbol{0} \tag{42}$$
$$\frac{\partial^2 \boldsymbol{H}_{int}}{\partial I_t^2} = \boldsymbol{0} \tag{43}$$
$$\begin{aligned}\frac{\partial^2 \boldsymbol{H}_{int}}{\partial \omega_z^2} &= \frac{\partial\left(I_z \hat{\boldsymbol{\omega}}_z + t_{fd}\frac{\partial [\overline{\boldsymbol{q}}_{\mathrm{I}}]}{\partial \omega_z}\boldsymbol{T}_b\right)}{\partial \omega_z} \\ &= t_{fd}\frac{\partial^2 [\overline{\boldsymbol{q}}_{\mathrm{I}}]}{\partial \omega_z^2}\boldsymbol{T}_b \\ &= M2\left(\overline{\boldsymbol{q}}_{\mathrm{I}}, \frac{\partial \overline{\boldsymbol{q}}_{\mathrm{I}}}{\partial \omega_z}, \frac{\partial^2 \overline{\boldsymbol{q}}_{\mathrm{I}}}{\partial \omega_z^2}\right)\boldsymbol{T}_b t_{fd} \end{aligned} \tag{44}$$
$$\frac{\partial^2 \boldsymbol{H}_{int}}{\partial t_{fd}^2} = \boldsymbol{0} \tag{45}$$
$$\tag{46}$$

For $\boldsymbol{H}_f$ using Equation (3), with the *cancell* subscript denoting the cancellation pulse and $\overline{\boldsymbol{q}}_{321}$ defined in Equation (32):

$$\begin{aligned}\boldsymbol{H}_f &= \boldsymbol{H}_{int} + \boldsymbol{T} t_{fd}\Big|_{cancell} \\ &= \boldsymbol{H}_{int} + [\overline{\boldsymbol{q}}_{\mathrm{III}} \otimes \overline{\boldsymbol{q}}_{\mathrm{II}} \otimes \overline{\boldsymbol{q}}_{\mathrm{I}}]\boldsymbol{T}_b t_{fd} \\ &= \boldsymbol{H}_{int} + [\overline{\boldsymbol{q}}_{321}]\boldsymbol{T}_b t_{fd} \end{aligned} \tag{47}$$

Setting up the equations for $\overline{\boldsymbol{q}}$ as functions of time t:

$$\overline{\boldsymbol{q}}_{\mathrm{I}} = \begin{bmatrix} 0 \\ 0 \\ \sin(\alpha_0/2) \\ -\cos(\alpha_0/2) \end{bmatrix} \tag{48}$$
$$\overline{\boldsymbol{q}}_{\mathrm{II}} = \begin{bmatrix} \hat{\boldsymbol{H}}_{int}^{bI} \sin(\omega_H t/2) \\ -\cos(\omega_H t/2) \end{bmatrix} \tag{49}$$
$$\overline{\boldsymbol{q}}_{\mathrm{III}} = \begin{bmatrix} 0 \\ 0 \\ \sin(\omega_N t/2) \\ -\cos(\omega_N t/2) \end{bmatrix} \tag{50}$$

Note that the $\hat{\boldsymbol{H}}_{int}$ occurring in Equation (49) needs to be expressed in the SFB–I reference frame, not the RI frame, since $\overline{\boldsymbol{q}}_{\mathrm{II}}$ is specified relative to that frame. The minus sign in the fourth element is present to get the right conversion: SFB to RI instead of RI to SFB.

The derivative of $\overline{\boldsymbol{q}}_{\mathrm{I}}$ only depends on ω_z because of

α_0, as follows:

$$\frac{\partial \overline{\boldsymbol{q}}_{\mathrm{I}}}{\partial \omega_z} = \begin{bmatrix} 0 \\ 0 \\ \cos(\alpha_0/2) \\ \sin(\alpha_0/2) \end{bmatrix} \frac{\partial(\alpha_0/2)}{\partial \omega_z} = \begin{bmatrix} 0 \\ 0 \\ \cos(\alpha_0/2) \\ \sin(\alpha_0/2) \end{bmatrix} t_0/2 \tag{51}$$

This result has been used above in Equation (40). The second-order derivative is then given by:

$$\frac{\partial^2 \overline{\boldsymbol{q}}_{\mathrm{I}}}{\partial \omega_z^2} = \begin{bmatrix} 0 \\ 0 \\ -\sin(\alpha_0/2) \\ \cos(\alpha_0/2) \end{bmatrix} (t_0/2)^2 \tag{52}$$

For $\overline{\boldsymbol{q}}_{\mathrm{II}}$ chain rule derivation with respect to an arbitrary variable ξ yields Equation (53):

$$\frac{\partial \overline{\boldsymbol{q}}_{\mathrm{II}}}{\partial \xi} = \frac{\partial \begin{bmatrix} \hat{\boldsymbol{H}}_{int}^{b\mathrm{I}} \sin(\omega_H t_s^*/2) \\ -\cos(\omega_H t_s^*/2) \end{bmatrix}}{\partial \xi} = \begin{bmatrix} \frac{\partial \hat{\boldsymbol{H}}_{int}^{b\mathrm{I}}}{\partial \xi} \sin\frac{\omega_H t_s^*}{2} + \hat{\boldsymbol{H}}_{int}^{b\mathrm{I}} \frac{t_s^*}{2} \frac{\partial \omega_H}{\partial \xi} \cos\frac{\omega_H t_s^*}{2} \\ \frac{t_s^*}{2} \frac{\partial \omega_H}{\partial \xi} \sin\frac{\omega_H t_s^*}{2} \end{bmatrix} \tag{53}$$

Where t_s^* is the pre-calculated slew time and is independent of the perturbations as the controller does not know of these perturbations but uses precoded values. Second-order derivatives are derived from this equation.

For $\overline{\boldsymbol{q}}_{\mathrm{III}}$:

$$\frac{\partial \overline{\boldsymbol{q}}_{\mathrm{III}}}{\partial \xi} = \frac{\partial \begin{bmatrix} 0 \\ 0 \\ \sin(\omega_N t_s^*/2) \\ -\cos(\omega_N t_s^*/2) \end{bmatrix}}{\partial \xi} = \begin{bmatrix} 0 \\ 0 \\ \cos(\omega_N t_s^*/2) \\ \sin(\omega_N t_s^*/2) \end{bmatrix} t_s^*/2 \frac{\partial \omega_N}{\partial \xi} \tag{54}$$

$$= \begin{bmatrix} 0 \\ 0 \\ \sin(\pi/2 - \omega_N t_s^*/2) \\ \cos(\pi/2 - \omega_N t_s^*/2) \end{bmatrix} t_s^*/2 \frac{\partial \omega_N}{\partial \xi}$$

$$= \overline{\boldsymbol{q}}_{\mathrm{III}}(\pi/\omega_N - t_s^*) t_s^*/2 \frac{\partial \omega_N}{\partial \xi} \tag{55}$$

The corresponding partial derivatives of $\overline{\boldsymbol{q}}_{\mathrm{III}}$ are summarised here in equations (56), (57) and (58) below:

$$\frac{\partial \overline{\boldsymbol{q}}_{\mathrm{III}}}{\partial I_z} = \overline{\boldsymbol{q}}_{\mathrm{III}}(t_s^* + \pi/\omega_N) \frac{-\omega_z}{I_t} t_s^*/2 \tag{56}$$

$$\frac{\partial \overline{\boldsymbol{q}}_{\mathrm{III}}}{\partial I_t} = \overline{\boldsymbol{q}}_{\mathrm{III}}(t_s^* + \pi/\omega_N) \frac{\lambda \omega_z}{I_t} t_s^*/2 \tag{57}$$

$$\frac{\partial \overline{\boldsymbol{q}}_{\mathrm{III}}}{\partial \omega_z} = \overline{\boldsymbol{q}}_{\mathrm{III}}(t_s^* + \pi/\omega_N)(1-\lambda) t_s^*/2 \tag{58}$$

Second-order derivatives of Equation (54) are then as follows:

$$\frac{\partial^2 \overline{\boldsymbol{q}}_{\mathrm{III}}}{\partial \xi^2} = \begin{bmatrix} 0 \\ 0 \\ -\sin(\omega_N t_s^*/2) \\ \cos(\omega_N t_s^*/2) \end{bmatrix} \left(\frac{t_s^*}{2} \frac{\partial \omega_N}{\partial \xi} \right)^2 + \begin{bmatrix} 0 \\ 0 \\ \cos(\omega_N t_s^*/2) \\ \sin(\omega_N t_s^*/2) \end{bmatrix} \frac{t_s^*}{2} \frac{\partial^2 \omega_N}{\partial \xi^2} \tag{59}$$

$$\tag{60}$$

With the above equations, the values for all first-order derivatives are known. With these variables, an estimate for the error in $\boldsymbol{H}_{int}$ and $\boldsymbol{H}_f$ due to an error in I_z, I_t or ω_z can be estimated using a second-order Taylor series expansion, as done in Equation (61) for the case of an error in I_z.

$$\boldsymbol{H}_f \approx \boldsymbol{H}_{f,0} + \frac{\partial \boldsymbol{H}_f}{\partial I_z}(\boldsymbol{H}_{f,0}) \Delta I_z + \frac{\partial^2 \boldsymbol{H}_f}{\partial I_z^2}(\boldsymbol{H}_{f,0}) \frac{(\Delta I_z)^2}{2} \tag{61}$$

IV.IV Residual Nutation

The residual nutation error is determined by the distance between $\boldsymbol{Z}_f$ and $\boldsymbol{H}_f$. Of these two $\boldsymbol{H}_f$ is defined above, so this leaves finding an equation for the spin axis $\boldsymbol{Z}_f$. Fortunately the work done for $\boldsymbol{T}_{cancell}$ (see Equation (47)) can be reused as $\boldsymbol{Z}_f$ is defined in the same reference frame:

$$\begin{aligned}\boldsymbol{Z} = \boldsymbol{Z}_{RI} &= [\overline{\boldsymbol{q}}_{\mathrm{SF2RI}}]\boldsymbol{Z}_{SFB} \\ &= [\overline{\boldsymbol{q}}_{\mathrm{III}} \otimes \overline{\boldsymbol{q}}_{\mathrm{II}} \otimes \overline{\boldsymbol{q}}_{\mathrm{I}}]\,\boldsymbol{Z}_{SFB} \\ &= [\overline{\boldsymbol{q}}_{\mathrm{II}} \otimes \overline{\boldsymbol{q}}_{\mathrm{I}}] \begin{bmatrix}0\\0\\1\end{bmatrix}\end{aligned} \tag{62}$$

Compared with $\boldsymbol{T}$, this equation is actually simpler since the rotation of the spacecraft around its Z-axis given by $\overline{\boldsymbol{q}}_{\mathrm{III}}$ is not relevant for the orientation of this Z-axis in inertial space.

Going into more detail, the residual nutation error can be written as follows:

$$\Delta_\theta = \arccos(\boldsymbol{Z}_f \cdot \hat{\boldsymbol{H}}_f) \tag{63}$$

The dot-product of two vectors does not yield the enclosed angle but rather the cosine of this angle if both vectors are unit vectors, hence the need to transform $\boldsymbol{H}_f$ into its unit vector $\hat{\boldsymbol{H}}_f$. For simplicity, the following equations will focus on the dot-product, leaving out the arccosine.

Using the chain rule, the partial derivative of this dot-product relative to an arbitrary variable ξ can be written as follows:

$$\begin{aligned}\frac{\partial(\boldsymbol{Z}\cdot\hat{\boldsymbol{H}}_f)}{\partial\xi} =& \frac{1}{H_f}\left(\frac{\partial \boldsymbol{Z}}{\partial\xi}\cdot\boldsymbol{H_f} + \boldsymbol{Z}\cdot\frac{\partial \boldsymbol{H}_f}{\partial\xi}\right) \\ &+ \frac{\boldsymbol{Z}\cdot\boldsymbol{H}_f}{-H_f^2}\frac{\partial H_f}{\partial\xi}\end{aligned} \tag{64}$$

Note that it was decided to split $\hat{\boldsymbol{H}}_f = \boldsymbol{H}_f/H_f$ in order to facilitate reuse of formulas derived above; this does impose an additional chain rule term.

Concerning the partial derivatives occurring in the last Equation (64), $\partial\boldsymbol{H}_f/\partial\xi$ has been calculated in the previous section and $\partial\boldsymbol{Z}_f/\partial\xi$ can be obtained through taking the derivatives for the above derivation of $\boldsymbol{Z}_f$ in (62). This leaves $\partial H_f/\partial\xi$, which is derived in Equation (65) below:

$$\begin{aligned}\frac{\partial H_f}{\partial\xi} &= \frac{\partial\sqrt{\boldsymbol{H}_f\cdot\boldsymbol{H}_f}}{\partial\xi} \\ &= 1/H_f\left(\boldsymbol{H_f}\cdot\frac{\partial\boldsymbol{H}_f}{\partial\xi}\right) \\ &= \hat{\boldsymbol{H}}_f\cdot\frac{\partial\boldsymbol{H}_f}{\partial\xi}\end{aligned} \tag{65}$$

which is again composed of pre-derived quantities.

V ANALYSIS BY SIMULATIONS

In every case the 'predicted' behaviour (if any) of the spacecraft is compared with simulations. The simulator used is developed in Matlab Simulink, with a collection of M-files in supporting roles. The Simulink model only requires one M-file to set up initial values for one simulation run; the added value of the remaining M-files is mainly in enabling automation of chaining many simulation runs into batches and post-processing the results of each batch. One batch typically contains about 110 simulation runs and generates one pair of (Δ_H, Δ_Z)-figures in the next section VII.

The assumptions used in the simulation set-up are identical to the ones described in Section II, with one major exception: the 'impulsive shot' assumption (part of the 'ideal thruster' assumption) is no longer upheld in the simulations. Instead, the angular impulse is now spread over the thruster firing duration, resulting in a potentially significant aberration from the theoretical predictions as the combination of spin rate and thruster firing duration implies that the control torque is not continuously in the desired direction during the thruster firing duration.

Simulations are the only method used in this report for analysing the Rhumb line slew (RL). This algorithm is dependent on an external signal (the sun pulse), which is in its turn depending on the spacecraft's instantaneous attitude and precludes most attempts at analytical simplification.

Aside from this, simulations are also the only method used in this report when analysing every algorithm's robustness against a perturbation in the axisymmetric assumption, as this assumption is a very fundamental building block for the mathematical analyses discussed in this paper. However, it is not impossible (but still quite complex) to mathematically describe the spacecraft behaviour in case of small perturbations: see e.g. Livneh and Wie [7].

V.I Simulation Inputs

Simulations were made using a MATLAB Simulink simulator as described in Wu et al. [8]. The initial state of the spacecraft is a pure spin around its SFB Z-axis. The attitude thruster is modelled as an on-off thruster, meaning the thrust is either zero or maximum.

Input parameters	Value
I_x	$5.35416\,\mathrm{kg{\cdot}m^2}$
I_y	$5.35416\,\mathrm{kg{\cdot}m^2}$
I_z	$0.065\,\mathrm{kg{\cdot}m^2}$
Control torque $\boldsymbol{T}_b$	[0 0.1 0]N·m
Spin rate (around Z-axis)	0.2 rad/s
Initial Euler angles	[0 0 0]
Target slew angle δ_t	90°

Table 1: Input parameters for simulations

Table 1 defines the constants used during the simulation run. Each simulation will be compared on three main aspects: accuracy, time efficiency and energy efficiency.

Accuracy is measured by the angular difference (error) between the target attitude and the final attitude. The 'final' attitude is defined as the attitude immediately after the last thruster pulse. Any residual nutation angle at this point will induce a varying angular error, which is why the error is divided into two parts: the angular momentum error Δ_H, which is the angle between the target angular momentum vector $\boldsymbol{H}_t$ and its final value $\boldsymbol{H}_f$, and the residual nutation angle Δ_Z, which is the angle between the final spin axis $\boldsymbol{Z}_f$ and $\boldsymbol{H}_f$, see Figure 2. These two errors are time-independent in the absence of disturbance torques.

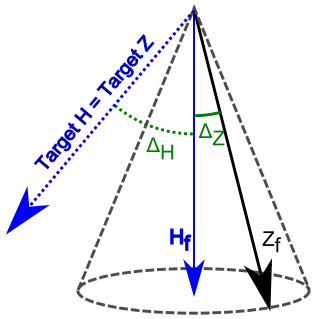

Fig. 2: Accuracy: Angular momentum error Δ_H and residual nutation angle Δ_Z

VI METHODOLOGY OVERVIEW

Table 2 gives an overview of the analysis method used for an algorithm, depending on the perturbation. As stated above, in summary this boils down to:

- Simulations are always used;
- The Taylor series approximation is only used for the Half-cone manoeuvre;
- All other manoeuvres use a geometric approach when available; and
- Asymmetric perturbations are only analysed through simulations.

Perturbation	I_z	I_t	ω_z	t_{fd}	I_y
HC	T,S	T,S	T,S	T,S	S
EHC	G,S	G,S	G,S	G,S	S
SAS	G,S	G,S	G,S	G,S	S
DC	G,S	G,S	G,S	G,S	S
RL	S	S	S	S	S

Table 2: Overview of analysis methods (**T**aylor, **G**eometric, **S**imulation) used per algorithm and perturbation

Legend:

- T = Taylor series approximation (see Section IV)
- G = Geometric method (see Section III)
- S = Simulations (see Section V)

VII ROBUSTNESS ANALYSIS RESULTS

For all the algorithms defined in the previous Section III, robustness analyses were performed for a perturbation of one of five different parameters:

1. Spin axis moment of inertia I_z
2. Transverse moment of inertia I_t
3. Spin rate ω_z
4. Thruster firing duration (also called 'thruster on-time') t_{fd}

5. Y-axis moment of inertia I_y, to create a non-axisymmetric spacecraft

These perturbations are in linear steps ranging from -1% to $+1\%$, except for t_{fd} which is perturbed in linear steps from -10ms to +10ms, as this was deemed more realistic.

In order to see the effect of a parameter range on robustness, additionally parameters can be varied in linear steps from -10% to $+10\%$. Due to space restrictions, in this paper only slew angle δ will be considered.

The Pulse-train manoeuvre analyses do not consider the slew angle as parameter because it has virtually no impact on the controller (only on controller cutoff time), in contrast to the Half-cone derived manoeuvres.

As previously done in Raus et al. [3], a slight modification was made to the initial parameters described in table 1 in order to enable representative Pulse-train slew manoeuvres (RL, SS): the spin axis moment of inertia (I_z) is increased tenfold to $0.65\,\mathrm{kg{\cdot}m^2}$. Furthermore, for these slews, the thruster firing duration t_{fd} is set constant to $0.05\,\mathrm{s}$.

VII.I Half-cone Slew

The behaviour of Δ_H relative to δ_t seems erratic at first, but this is caused by the integer round-off in the Half-cone's governing equations due to the integer number of attainable nutation angles. This integer round-off also causes the Δ_H behaviour relative to the perturbation: for a certain requested slew angle δ_t, Δ_H can become smaller when I_z is off-nominal as this I_z value actually enables a nutation angle θ closer to the ideal value of $\delta_t/2$. This behaviour returns in other Half-cone robustness analyses for transverse and spin axis moments of inertia, as these two parameters are also influencing the attainable Half-cone nutation angles.

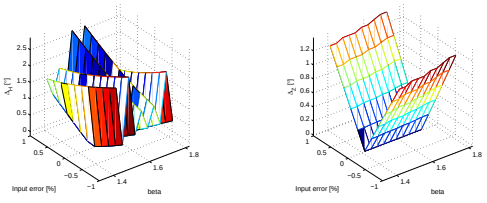

Fig. 3: Δ_H and Δ_Z as a function of δ_t and I_z perturbation for HC

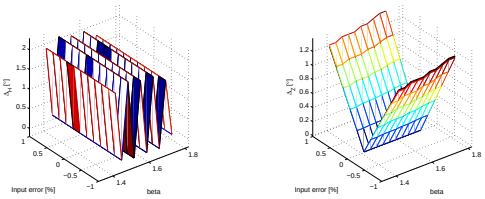

Fig. 4: Δ_H and Δ_Z as a function of δ_t and I_t perturbation for HC

The effects of a spin axis moment of inertia or transverse moment of inertia perturbation as sketched in Figures 3 and 4 are reasonable: at 1% perturbation, the final pointing error amounts to about 4° maximum. Furthermore, there seems to be little or no effect of an I_t perturbation on Δ_H.

In sharp contrast, the Half-cone manoeuvre is extremely sensitive to an error in spin rate as can be seen from Figure 5. The second-order predictions do overestimate the angular error, but by definition Taylor series predictions are only valid 'close' to the nominal point. Given that the simulated results already go up to an extreme of about 50° for each of the error angles, this is definitely out of the confidence interval of the second-order Taylor series approximation. This sensitivity to spin rate ω_z reappears in all Half-cone derived manoeuvres.

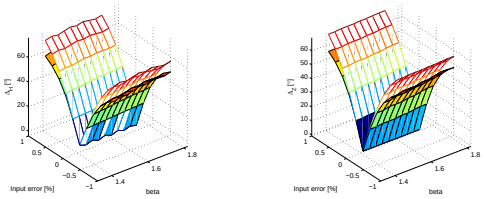

Fig. 5: Δ_H and Δ_Z as a function of δ_t and ω_z perturbation for HC

By inverting the second-order predictions, an initial estimate for the accuracy of the spin rate determination can be found if the pointing error is required to be less than a certain value. E.g. in this case, for a pointing accuracy in the order of 5° or less, the error in ω_z should preferably be less than 0.05%. This translates to an accuracy of 0.1 mrad/s on a spin rate of $\omega_z = 0.2\,\mathrm{rad/s}$.

The effects of a timing error in the thruster firing duration can be seen in Figure 6. Note that the X-axis is different in scale from before, ranging between

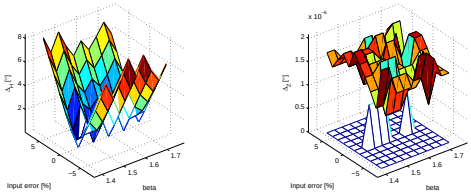

Fig. 6: Δ_H and Δ_Z as a function of δ_t and t_{fd} perturbation for HC

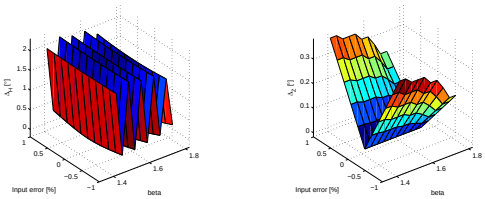

Fig. 7: Δ_H and Δ_Z as a function of δ_t and asymmetric perturbation for HC

-8% and $+8\%$, due to the way the error is inserted into the simulations: where previously the error was defined as a percentage of the nominal value, for a t_{fd} simulation it is defined as an absolute value (in seconds) and afterwards recalculated as a percentage of the nominal t_{fd}. This is done because in contrast to previous parameters, which were input constants, t_{fd} is calculated by the on-board controller and the error should therefore be inserted after this step. Since the value of t_{fd} is formally 'unknown' beforehand, the error cannot be defined as a percentage of t_{fd} but has to be defined in absolute value.

An error in t_{fd} appears to have no effect on the residual nutation angle Δ_Z when the scale of the Z-axis in this figure: 10^{-6} degrees is taken into account. This is because both the initiation and the cancellation pulse are modified with the same absolute error; since the timing of the cancellation pulse depends on ω_N, which is independent of t_{fd}, the thruster will still be in the correct position for a complete cancellation when the command to fire the cancellation pulse is given. This is true for all Half-cone derived algorithms.

Finally, a perturbation of the axisymmetric assumption has an effect on Δ_H very similar to that of an I_t perturbation, compare Figures 4 and 7. This seems to indicate the Half-cone manoeuvre is relatively robust against errors in the inertia matrix. The discontinuities resulting from a changing δ_t while the nutation angle is limited to a definite number of values is again apparent in both Δ_H and Δ_Z.

Δ_Z for an asymmetric perturbation is surprisingly small compared to e.g. the I_t perturbation and appears to decrease with δ_t.

VII.II Extended Half-cone Slew

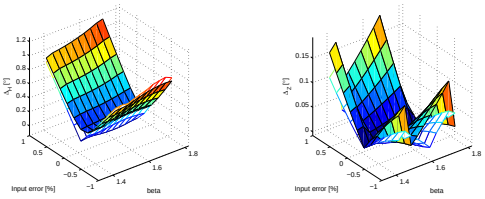

Fig. 8: Δ_H and Δ_Z as a function of δ_t and I_z perturbation for EHC

The trend for Δ_H seen in Figures 8 and 9 is to increase slightly with increasing slew angle for perturbations in I_t and I_z. This can be expected as the larger the total slew angle is, the larger the nutation angles of the constitutive Half-cones are, and therefore a perturbed slew can have a higher final error. The discontinuities in the Δ_Z graphs are caused by the switch from one k-value to a next one, which is obviously highly depending on slew angle δ_t.

The effects of a spin rate perturbation, visible in Figure 10, are again by far the most influential on the pointing errors. A t_{fd} perturbation plotted in Figure 11 has a moderate effect on Δ_H and a negligible effect on Δ_Z, as was also seen for the Half-cone manoeuvre in Section VII.I.

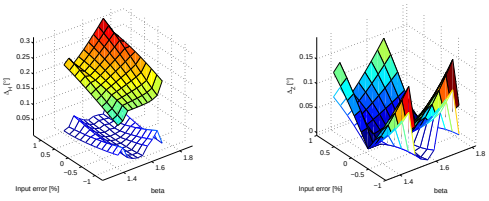

Fig. 9: Δ_H and Δ_Z as a function of δ_t and I_t perturbation for EHC

The effects of an asymmetric inertia matrix shown in Figure 12 appear quite similar to the effects of an

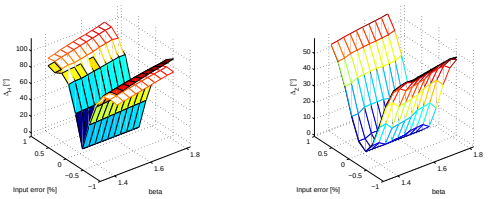

Fig. 10: Δ_H and Δ_Z as a function of δ_t and ω_z perturbation for EHC

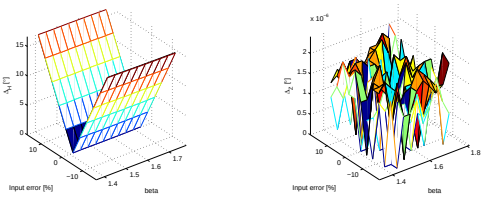

Fig. 11: Δ_H and Δ_Z as a function of δ_t and t_{fd} perturbation for EHC

I_t perturbation as in Figure 9, though with slightly higher values for Δ_H and lower values for Δ_Z. The effects of varying moments of inertia, spin rate or thrust are quite similar to their corresponding slew angle graphs and are therefore omitted for brevity.

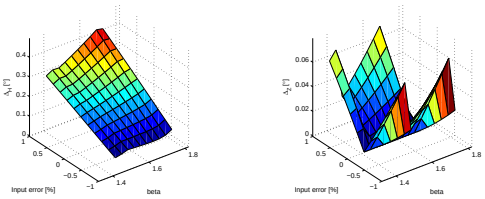

Fig. 12: Δ_H and Δ_Z as a function of δ_t and asymmetric perturbation for EHC

VII.III Sector-Arc Slew

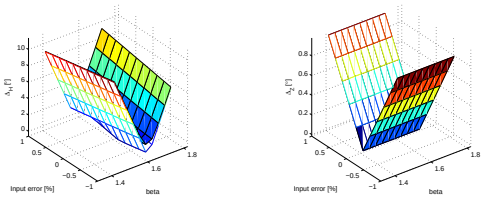

Fig. 13: Δ_H and Δ_Z as a function of δ_t and I_z perturbation for SAS

The variation in Δ_H with slew angle δ_t is due to the fact that the SAS parameters (nutation angle θ and k) are optimised for $\delta_t = 90°$. Δ_Z on the other hand does not seem to be influenced by δ_t at all. Again, similar to the Half-cone slew, the SAS is most sensitive to a perturbation in spin rate, with Δ_H and Δ_Z reaching comparable values of about 50°.

The effects of an asymmetric inertia matrix shown in Figure 17 appear quite similar to the effects of an I_t perturbation as in Figure 14, though with slightly higher values for Δ_H and lower values for Δ_Z. The effects of varying moments of inertia are quite similar to their corresponding slew angle graphs and are therefore omitted for brevity.

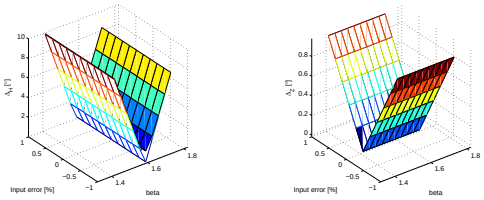

Fig. 14: Δ_H and Δ_Z as a function of δ_t and I_t perturbation for SAS

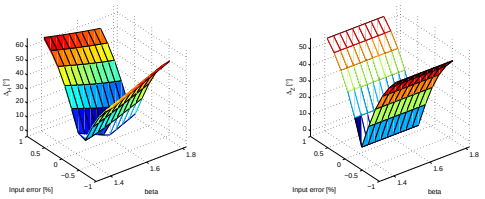

Fig. 15: Δ_H and Δ_Z as a function of δ_t and ω_z perturbation for SAS

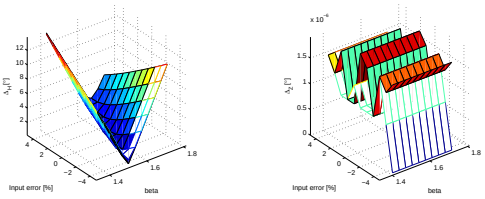

Fig. 16: Δ_H and Δ_Z as a function of δ_t and t_{fd} perturbation for SAS

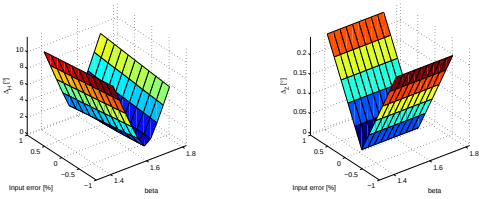

Fig. 17: Δ_H and Δ_Z as a function of δ_t and asymmetric perturbation for SAS

VII.IV Dual-cone Slew

There are a few characterising trends visible in the Dual-cone robustness graphs. The main improvement with respect to the HC is that the nominal values for both errors at zero perturbation are very close to zero – which was the idea behind the Dual-cone slew algorithm in the first place. The one discontinuity occurring with increasing δ_t is where the algorithm chooses a different k-value.

The effects of an asymmetric inertia matrix shown in Figure 22 appear quite similar to the effects of an I_t perturbation as in Figure 19, though with slightly higher values for Δ_H and lower values for Δ_Z. The effects of varying moments of inertia, spin rate or

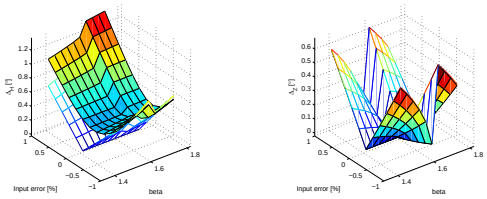

Fig. 18: Δ_H and Δ_Z as a function of δ_t and I_z perturbation for DC

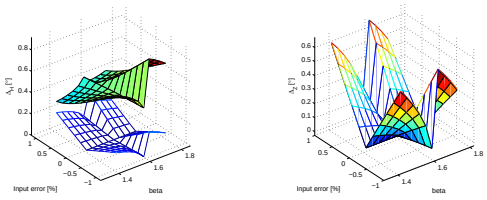

Fig. 19: Δ_H and Δ_Z as a function of δ_t and I_t perturbation for DC

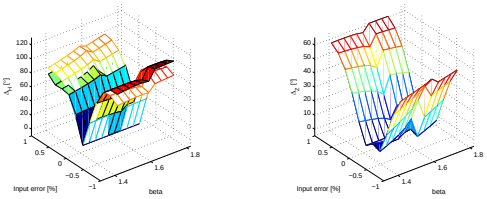

Fig. 20: Δ_H and Δ_Z as a function of δ_t and ω_z perturbation for DC

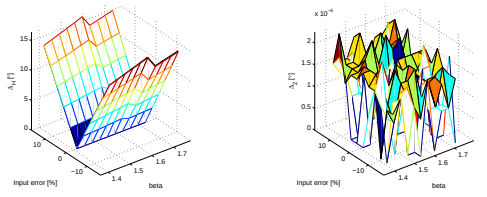

Fig. 21: Δ_H and Δ_Z as a function of δ_t and t_{fd} perturbation for DC

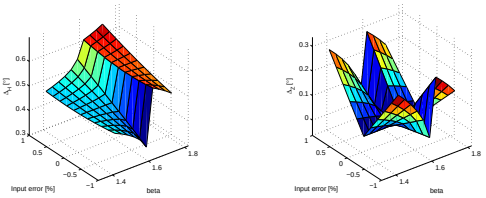

Fig. 22: Δ_H and Δ_Z as a function of δ_t and asymmetric perturbation for DC

thrust are quite similar to their corresponding slew angle graphs and are therefore omitted for brevity.

VII.V Rhumb Line Slew

For the Rhumb Line algorithm, a model of a sun sensor was implemented. This gives rise to additional initialisation data, in particular:

- Sun vector in RI: [0 1 0]
- Sun sensor normal vector in SFB: [-1 0 0]

Nearly all Δ_H figures appear to indicate the nominal value of I_t actually yields the worst result (highest Δ_H). The one exception to this observation seems to be the t_{fd} perturbation in Figure 26, though this may just be an effect of scale: for all other graphs the Δ_H range is at the most 1°, while for Figure 26 this range is 20°. This likely indicates that the nominal t_{fd} value is suboptimal for a 90° slew. Δ_Z is nearly the same for all considered perturbations with values ranging up to 6° and minimum around the nominal value for I_t, with the perturbation itself having a very small effect.

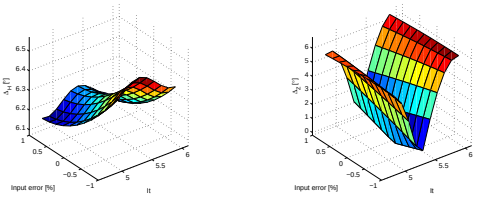

Fig. 23: Δ_H and Δ_Z as a function of I_t and I_z perturbation for RL

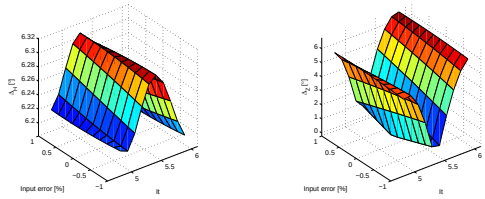

Fig. 24: Δ_H and Δ_Z as a function of I_t and I_t perturbation for RL

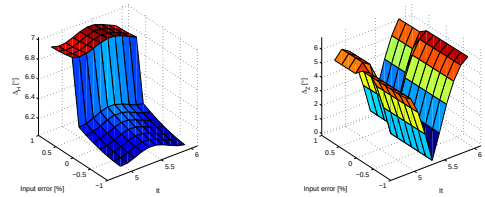

Fig. 25: Δ_H and Δ_Z as a function of I_t and ω_z perturbation for RL

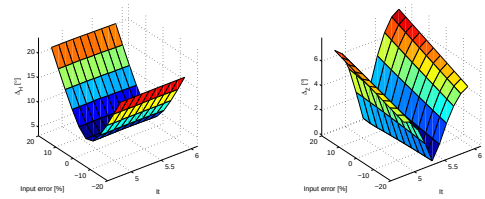

Fig. 26: Δ_H and Δ_Z as a function of I_t and t_{fd} perturbation for RL

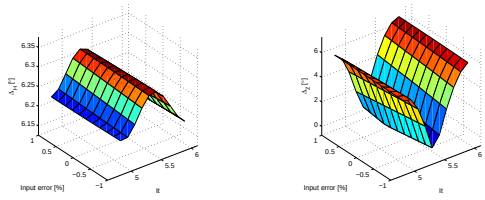

Fig. 27: Δ_H and Δ_Z as a function of I_t and asymmetric perturbation for RL

Approach	Pro	Con
Geometrical (Section III)	Fast	Very difficult for Rhumb Line Only one point per calculation
Taylor Series (Section IV)	Comprehensive: one calculation yields local solution behaviour Moderately fast	Complex Only valid in small region around nominal point
Simulations (Section V)	Accurate Possible for all slews	Slow Only one point per calculation

Table 3: Advantages and disadvantages of the studied approaches

Algorithm	I_z	I_t	ω_z	t_{fd}	Asymm
Half-cone family	moderate	moderate	extreme	minimal	comparable to I_t
Rhumb line	moderate	moderate	moderate	moderate	comparable to I_t

Table 4: Overview of algorithms versus robustness

VIII ROBUSTNESS ANALYSIS OVERVIEW

VIII.I Observations on Analytical Prediction Algorithms

For most algorithms, the values for Δ_H and Δ_Z predicted by the analytical approaches, whether Taylor series approximation or geometrical, is quite close to the simulated values.

One important advantage of the analytical solutions over simulations is that they can be inverted to e.g. determine what the minimum spin rate needs to be in order to bring the slew time down to a certain duration, or to determine the torque level for minimum Δ_H.

The largest discrepancies between simulations and analytical predictions appear to stem from the 'impulsive shot' assumption that is only applied in the predictions.

Table 3 summarises the advantages and disadvantages of the three methodologies used for the robustness analysis as defined in Sections III, IV and V.

VIII.II Slew Robustness Overview

The Half-cone derived slews are most sensitive to perturbations in the spin rate. For the nominal scenario of Table 1 a 1% perturbation in spin rate leads to extremely large values of Δ_H and Δ_Z, in the order of $50° - 80°$. The Rhumb Line slew is not nearly as sensitive to a perturbation in spin rate due to the use of a Sun sensor, though some dependency on spin rate remains in the conversion of heading angle and torque angle to time delays.

Table 5 gives a quantitative overview of the maximum values for Δ_H and Δ_Z, in degrees. The rows represent the maximum error angle for one specific perturbation for all scenarios.

VIII.III Confirmation of DC and EHC Design

Dual-cone and Extended Half-cone perform better when varying the slew angle than Half-cone or Multi-cone. These findings confirm the basic intention behind them that led to a definition especially crafted to lessen the fluctuations (discrete jumps) visible in e.g. the Half-cone results for slew angle variations (compare the Half-cone Figure 3 to the Dual-cone Figure 18).

IX CONCLUSIONS

A thorough robustness analysis was performed for several slew algorithms identified as usable for a spinning prolate spacecraft: the Half-cone derived slews Half-cone, Extended Half-cone, Sector-Arc Slew and Dual-cone, and the two Rhumb Line slew.

	HC		EHC		SAS		DC		RL	
	Δ_H	Δ_Z	Δ_H	Δ_Z	Δ_H	Δ_Z	Δ_H	Δ_Z	Δ_H	Δ_Z
I_z	2.5	1.2	1.2	0.2	3	1.2	1.2	0.6	12	7
I_t	2	1.2	0.3	0.15	2.5	1.2	0.8	0.6	12	7
ω_z	60	60	80	50	60	50	100	60	14	7
I_y (asymm)	2	0.4	0.4	0.07	4	0.3	0.7	0.3	13	7
t_{fd} (+/-1ms)	8	2e-06	16	2e-06	3.5	2e-06	16	2e-06	30	9

Table 5: Overview of maximum angular errors in degrees per algorithm and perturbation

IX.I Analytical and Numerical Approaches

As a first step potential analytical approaches for this robustness analysis were investigated and verified based on feasibility and complexity.

For the Half-cone slew, first-order and second-order Taylor series approximations were developed to calculate the final angular momentum vector and the final spin axis vector. After comparison with simulation data, the second-order (quadratic) approximation is deemed more precise than the first-order (linear) approximation, but in certain situations the first-order approximation is still quite accurate for analytical predictions. The main advantage of the Taylor series approximation lies in the fact that with one calculation, it can supply an approximation of the behaviour of the solution in the vicinity of the calculated point (usually the nominal point). However, the Taylor series approximation rapidly becomes too complex for other slew algorithms.

For all Half-cone derived slews a geometrical approach has been derived. This approach is less versatile than the Taylor series approximation since it only gives information about the point where it is calculated; as a result, it needs to be run many times in order to get an idea of the behaviour of the solution near the nominal point. However, it is easier to derive and can therefore be used not just for all Half-cone derived algorithms but also for the Spin-Synch slew.

Analytical approaches were not used in the following two cases:

1. For the Rhumb line algorithm, predicting the external trigger (time of sun detection) requires calculating the spacecraft's instantaneous attitude at each time instant, comparing it with the external trigger condition and acting upon the result of this condition. Analytically deriving this was deemed too complex for this thesis and has therefore not been attempted.

2. All analytical approaches are founded on several assumptions, among which the assumption that the spacecraft is axisymmetric. Introducing an asymmetric inertia matrix in these analytical equations is not impossible but too complex for the aim of this thesis, and has as a result been left out.

In all cases simulations are used as benchmark for the result of an analytical approach where applicable. The main difference in assumptions between the analytical approaches and the simulations is the 'impulsive shot' assumption: the analytical approach assumes the impulse generated by the control torque is delivered instantaneously while the simulator numerically integrates the effect torque over the short but not insignificant time period of the thruster firing duration t_{fd}.

IX.II Analysis Results

The most conspicuous result of the robustness analysis is that all Half-cone derived slews and the Spin-Synch slew are highly sensitive to perturbations in the spin rate. The Rhumb line is not nearly as sensitive due to the use of a Sun sensor, though some dependency on spin rate remains in the conversion of heading angle and torque angle to time delays.

The high sensitivity to spin rate perturbation of the Half-cone derived manoeuvres (and Spin-Synch) poses very stringent requirements on the attitude rate knowledge before starting the manoeuvre and likely necessitates the use of a high-accuracy angular rate sensor (e.g. rate gyro). Another option might be to have the spacecraft perform a predefined manoeuvre and use system identification techniques to

try and quantify the deviations in input parameters from their predefined values, but this is of course depending on available time and resources.

REFERENCES

[1] H Shiraishi, S Tanaka, A Fujimura, and H Hayakawa. The present status of the Japanese penetrator mission: Lunar-A. *Journal of Advances in Space Research*, 42:386–393, 2008. doi:10.1016/j.asr.2007.08.022.

[2] Yang Gao, Andy Phipps, Mark Taylor, Ian A. Crawford, Andrew J. Ball, Lionel Wilson, Alan Smith, Dave Parker, Martin Sweeting, Alex da Silva Curiel, and Phil Davies. Concepts and instruments of UK MoonLITE & Moonraker missions. In *IAC 2007 Proceedings*, 2007. IAC-07-A3.6B.01.

[3] Robin Raus, Yang Gao, and Mark Watt. Review and analysis of single-thruster attitude control techniques for spinning spacecraft. In *IAC 2010 Proceedings*, 2010. IAC-10-C1.3.7.

[4] Robin Raus, Yang Gao, Yunhua Wu, and Mark Watt. Analysis of state-of-the-art single-thruster attitude control techniques for spinning spacecraft. *Acta Astronautica*, 76:60–78, Jul-Aug 2012. doi:10.1016/j.actaastro.2012.02.014.

[5] Yunhua Wu, Yang Gao, Robin Raus, and Mark Watt. Tradeoff analysis of attitude-control slew algorithms for prolate spinner. *Journal of Guidance, Control, and Dynamics*, 35(4):1143–1157, Jul-Aug 2012. doi:10.2514/1.55619.

[6] Marcel J Sidi. *Spacecraft Dynamics and Control*. Cambridge University Press, 1997.

[7] Rafael Livneh and Bong Wie. Asymmetric body spinning motion with energy dissipation and constant body-fixed torques. *Journal of Guidance, Control, and Dynamics*, 22(2):322–328, Mar-Apr 1999. AIAA-4381-101.

[8] Yunhua Wu, Yang Gao, Robin Raus, and Mark Watt. Single thruster attitude control software simulator for spinning spacecraft. In *IEEE Aerospace 2012 proceedings*, 2012. doi:10.1109/AERO.2012.6187256.