ADAPTIVE CONTROLLER FOR SIMULTANEOUS SPACECRAFT ATTITUDE TRACKING AND REACTION WHEEL FAULT DETECTION

George M. Nehma * and Camilo Riano-Rios † and Morokot Sakal ‡ and Madhur Tiwari†
Florida Institute of Technology, 150 W. University Blvd., Melbourne, FL, 32901

The attitude control of a spacecraft is integral to achieving mission success. However, failures in actuators such as reaction wheels are detrimental and can often lead to an early end of mission. We propose a Lyapunov-based adaptive controller that can estimate and compensate for reaction wheels degradation simultaneously. The controller incorporates an adaptive update control law with a gradient-based term and an integral concurrent learning term that collects input-output data for online estimation of uncertain parameters. The proposed controller guarantees attitude tracking and minimal loss of control authority under faulty reaction wheel performance. The controllers performance is tested through a variety of numerical simulations as well as a Software-In-Loop test, where the adaptive controller is run in Python via ROS2 whilst the environment is simulated in Simulink.

Nomenclature

$\boldsymbol{\omega}$	=	spacecraft angular velocity
$\boldsymbol{\sigma}$	=	modified rodrigues parameters (MRP)
$\boldsymbol{\Omega}$	=	RW angular velocities
$\boldsymbol{u}$	=	control input
G	=	reaction wheel configuration matrix
Φ	=	RW health matrix
J	=	total inertia matrix
$K,\ \alpha$	=	control gain matrices
$K_1,\ \Gamma$	=	adaptation gain matrices
$\bar{\lambda}$	=	finite excitation threshold
$\boldsymbol{\theta}$	=	vector of uncertain parameters

*Ph.D. Student, Aerospace, Physics and Space Sciences Department; (Corresponding Author) gnehma2020@my.fit.edu
†Assistant Professor, Aerospace, Physics and Space Sciences Department
‡Ph.D. Student, Aerospace, Physics and Space Sciences Department
Presented at the AAS/AIAA Astrodynamics Specialist Conference, August 11-15 2024, Broomfield, CO. AAS 24-440

I. Introduction

The spacecraft's ability to accurately modify its attitude is one of the most critical features that must be guaranteed in any space mission. Reaction Wheels (RW) are the most commonly used actuators for this purpose due to their relatively small size and torque accuracy. However, their moving parts and the harsh conditions in space make these actuators prone to failure. The Attitude Control System (ACS) of a spacecraft is a highly critical system, hence the failure of this system often means the end of the mission. Redundant Reaction Wheel Arrays (RWA) with more than three RWs are often used to maintain control authority for three-axis attitude control, and for momentum management using the RWA null space [1].

Designing controllers for attitude control requires that they be resilient to disturbances and uncertainties. Current methods of control to address these issues include, sliding mode control (SMC) [2], adaptive control [3], observers (SMO) [4–6], multiple model filers [7] and neural networks [8]. Often due to the lack of accuracy and difficulty in proving stability, neural networks are rejected as viable solutions, whilst observers are used at the cost of more computational burden. Hence, of particular importance is the development of sliding mode-based strategies, which are powerful yet simple controllers that unfortunately can often suffer from chattering due to their inherent discontinuities. One of the disadvantages of SMO is the requirement to have the dynamics in a specific form [5, 6], which limits its applicability to a broader class of nonlinear systems. The work in [9] develops an adaptive, fault tolerant sliding mode that does not generate chattering through an adaptive control protocol. Adaptive controllers are also a prominent solution because of their innate ability to handle uncertainties in the nonlinear dynamics [9–11]. The authors in [12] demonstrate the capabilities for a fuzzy logic controller to perform state estimate under partial state observation, while [13] uses a fuzzy bilateral cooperative tracking controller estimate unknown nonlinear functions with formal guarantees on convergence, demonstrating the wide applicable nature of adaptive controllers within the domain of control systems.

Fault-tolerant controllers have been developed to compensate for the reduction in performance of the ACS actuators. In [14], a backstepping sliding mode controller was designed to maneuver the spacecraft's attitude despite faulty actuators, which were a combination of RWs and thrusters. [15] develops a sliding mode combined with disturbance observer fault tolerant controller that can quickly and efficiently handle actuator saturation and misalignment. Sensor failures were also detected and isolated using Unscented Kalman filters (UKF). Authors in [16] designed a RW failure detection strategy that employs a pair of long short-term memory (LSTM) neural networks, one to approximate the RW dynamics and the other to train an adaptive threshold detector based on the difference between RW measurements and the first network.

A fault diagnosis strategy was designed in [17] using a particle filter to predict the non-measurable bearing temperature lubricant in a RW, which has been identified as one of the main indicators for potential failures in this type of actuator. A similar approach was proposed in [18] to predict the remaining useful life of RWs by using an adaptive extended Kalman filter. In [19] a Lyapunov-based terminal sliding-mode observer was proposed to reconstruct RW

faults and disturbances, and a compensation control law was designed to account for them.

Adaptive control strategies are often used to compensate for parametric uncertainties in the system model by proposing gradient-based adaptation laws that compute estimates of the uncertain parameters online [20]. These adaptation laws are often sufficient to prove convergence of the error states. However, convergence of the model parameters to their real values can only be guaranteed under the assumption of Persistent Excitation (PE), i.e., the system is persistently excited over an infinite time integral, which is difficult to guarantee and verify in practice [21, 22]. An adaptation strategy, called Concurrent Learning (CL), was proposed to relax the PE condition by introducing in the adaptation law a term that collects input and output data under the assumption of measurable higher order states [23]. The CL adaptation strategy guarantees exponential convergence of the system parameter estimates after a verifiable Finite Excitation (FE) condition is satisfied, which can be evaluated online, and concurrent to the control execution, by computing the eigenvalue of a matrix built with collected input-output data. Recent efforts have relaxed the assumption of measurable higher order states by using their integrals in the so-called Integral Concurrent Learning (ICL) [24], which has been used in a wide range of control applications [25–27].

Compensation for actuator fault detection and detection or monitoring of actuator degradation are often considered as separate blocks in the spacecraft ACS, in the control and estimation blocks, respectively. This paper introduces a Lyapunov-based adaptive controller that performs both compensation and fault detection simultaneously, and achieves global exponential attitude tracking in the presence of RW failure or degradation. An actuator health matrix is included in the satellite attitude dynamics and considered uncertain. An ICL-based adaptive update law is designed to achieve convergence of the RW health parameters once the FE condition is satisfied. The controller is validated in simulation with different numbers of RWs along with varying types of degradation levels.

The paper is organized as follows: A first section presents the satellite attitude dynamic model, the next two sections describe the controller design and corresponding stability analysis. Finally, sections with performed numerical and SIL simulations, and conclusions are presented.

II. Spacecraft Attitude Dynamics

Three reference frames and their corresponding coordinate systems will be considered throughout this paper, and are depicted in Fig. 1. The Body reference frame $\mathcal{B}$, attached to the spacecraft body is defined with origin at its center of mass, and the unit vectors $\hat{\boldsymbol{b}}_1, \hat{\boldsymbol{b}}_2$ and $\hat{\boldsymbol{b}}_3 \in \mathbb{R}^3$ along its longitudinal, lateral and vertical axes, respectively. The coordinate system associated with the orbital reference frame $\mathcal{O}$ has origin at the spacecraft center of mass, and is defined by the unit vectors $\hat{\boldsymbol{o}}_1, \hat{\boldsymbol{o}}_2$ and $\hat{\boldsymbol{o}}_3 \in \mathbb{R}^3$, with $\hat{\boldsymbol{o}}_3$ aligned with the zenith direction, $\hat{\boldsymbol{o}}_2$ the orbital angular momentum vector, and $\hat{\boldsymbol{o}}_1$ aligned with the $(\hat{\boldsymbol{o}}_2 \times \hat{\boldsymbol{o}}_3)$ direction. The Earth-Centered-Inertial (ECI) coordinate system, centered at the Earth's center of mass and with basis $\{\hat{\boldsymbol{x}}, \hat{\boldsymbol{y}}, \hat{\boldsymbol{z}}\}$, is associated with the Inertial reference frame $\mathcal{N}$.

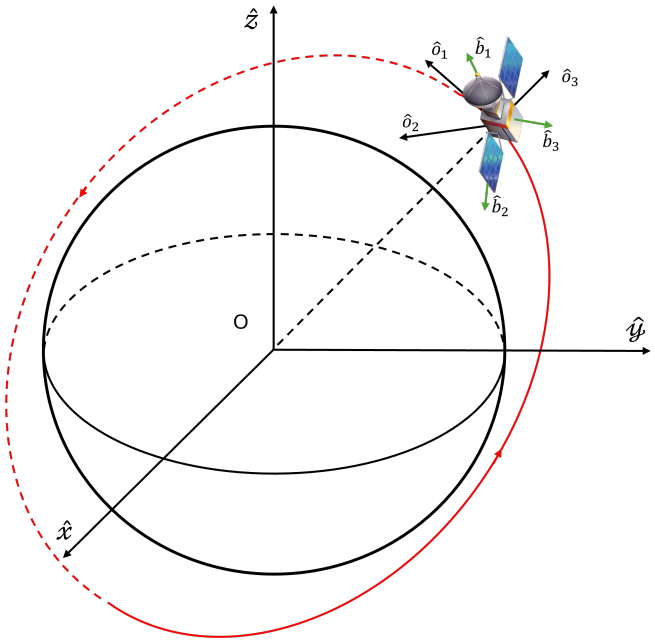


Fig. 1 Coordinate Systems

A. Equation of Motion

The attitude motion of a spacecraft with $N \in \mathbb{Z}_{>0}$ reaction wheels is governed by the following equations of motion:

$$J\dot{\boldsymbol{\omega}} = -\boldsymbol{\omega} \times (J\boldsymbol{\omega} + J_{RW} G\boldsymbol{\Omega}) + G\Phi\boldsymbol{u} + \boldsymbol{d} \tag{1}$$

$$\dot{\boldsymbol{\sigma}} = \frac{1}{4}\left[\left(1 - \boldsymbol{\sigma}^{\boldsymbol{T}}\boldsymbol{\sigma}\right) I_3 + 2\sigma^{\times} + 2\boldsymbol{\sigma}\boldsymbol{\sigma}^{\boldsymbol{T}}\right]\boldsymbol{\omega}, \tag{2}$$

where $\boldsymbol{\omega} \in \mathbb{R}^3$ is the spacecraft angular velocity expressed in the body coordinate system, $\boldsymbol{\sigma} \in \mathbb{R}^3$ is the vector of Modified Rodrigues Parameters (MRP) that represent the orientation of the spacecraft with respect to the inertial frame [28], $\boldsymbol{\Omega} = [\Omega_1, \Omega_2, \cdots, \Omega_N]^T \in \mathbb{R}^N$ is a vector containing the N RW angular velocities, $\boldsymbol{u} = -J_{RW}\dot{\boldsymbol{\Omega}} = [u_1, u_2, \cdots, u_N]^T \in \mathbb{R}^N$ is the control input that represents the torque applied by each RW, $J \in \mathbb{R}^{3\times3}$ is the total inertia matrix, $J_{RW} \in \mathbb{R}_{>0}$ is the inertia of the flywheels about their spin axis, $\Phi = diag\{\phi_1, \phi_2, \cdots, \phi_N\} \in \mathbb{R}^{N\times N}$ is the uncertain RW health matrix, $G = \{\hat{\boldsymbol{s}}_1, \hat{\boldsymbol{s}}_2, \cdots, \hat{\boldsymbol{s}}_N\} \in \mathbb{R}^{3\times N}$ is the RWA configuration matrix, and $\hat{\boldsymbol{s}}_i \in \mathbb{R}^3$ is the direction of the i^{th} RW's spin axis expressed in the body coordinate system, $\boldsymbol{d} \in \mathbb{R}^3$ represents external disturbances acting on spacecraft, and $\|\boldsymbol{d}\| \leq \delta$ with $\delta \in \mathbb{R}_{>0}$. The matrix $I_m \in \mathbb{R}^{m\times m}$ represents an identity matrix of dimension $m \times m$, and the skew-symmetric matrix $a^{\times} \in \mathbb{R}^{3\times3}$ for a vector $\boldsymbol{a} = [a_1, a_2, a_3]^T \in \mathbb{R}^3$ is defined as

$$a^{\times} = \begin{bmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{bmatrix}. \tag{3}$$

III. Control Design

A. Control Objective

The objective is to design an adaptive controller that achieves attitude tracking despite failures in one or more redundant reaction wheels. Besides compensating for these failures, the controller must also provide an online estimate of the health matrix Φ for actuators performance assessment.

Assumption 1. When failures occur to some of the RWs, the combined control effort of the remaining healthy RWs and the faulty RWs are sufficient maintain control authority along all body axes. This is a result of the redundant nature of RW configuration arrays. The underactuated case is not considered further in this paper.

Assumption 2. Changes or degradation in performance of RWs are slow enough, as compared to attitude maneuver times, for the Φ matrix to be considered constant.

The spacecraft attitude dynamics in Eq. (1) can be rewritten as

$$\dot{\boldsymbol{\omega}} = J^{-1}\left(-\boldsymbol{\omega} \times (J\boldsymbol{\omega} + J_{RW} G \boldsymbol{\Omega}) + Y\boldsymbol{\theta} + \boldsymbol{d}\right), \tag{4}$$

where the term $Y\boldsymbol{\theta} = G\Phi\boldsymbol{u} \in \mathbb{R}^3$ is a linear parameterization with respect to the uncertain diagonal entries of the health matrix Φ. $Y \in \mathbb{R}^{3\times N}$ is a measurable regression matrix defined as

$$Y = G\left(diag\{u_1, u_2, \cdots, u_N\}\right), \tag{5}$$

and the vector of uncertain parameters $\boldsymbol{\theta} \in \mathbb{R}^N$ as

$$\boldsymbol{\theta} = [\phi_1,\ \phi_2,\ \cdots,\ \phi_N]^T \tag{6}$$

Assumption 3. The spacecraft is equipped with an attitude determination system capable of providing the controller with angular velocity $\boldsymbol{\omega}$ and attitude $\boldsymbol{\sigma}$ measurements.

Let us introduce the error MRP $\boldsymbol{\sigma_e} \in \mathbb{R}^3$, representing the attitude mismatch between the body and desired frames, which can be computed using the spacecraft attitude $\boldsymbol{\sigma}$ and desired, bounded attitude trajectories $\boldsymbol{\sigma_d}, \boldsymbol{\omega_d} \in \mathbb{R}^3$, and

obeys the following kinematic equation [28]:

$$\dot{\boldsymbol{\sigma}}_{\boldsymbol{e}} = \frac{1}{4} B \tilde{\boldsymbol{\omega}}, \tag{7}$$

where the matrix $B \in \mathbb{R}^{3\times3}$ is defined as

$$B = \left(1 - \boldsymbol{\sigma}_{\boldsymbol{e}}^{\boldsymbol{T}} \boldsymbol{\sigma}_{\boldsymbol{e}}\right) I_3 + 2\sigma_e^{\times} + 2\boldsymbol{\sigma}_{\boldsymbol{e}} \boldsymbol{\sigma}_{\boldsymbol{e}}^{\boldsymbol{T}}, \tag{8}$$

$\tilde{\boldsymbol{\omega}} = \boldsymbol{\omega} - \tilde{R}\boldsymbol{\omega}_d \in \mathbb{R}^3$ is the relative angular velocity and $\tilde{R} \in \mathbb{R}^{3\times3}$, defined as

$$\tilde{R} = I_3 + \frac{8\left(\boldsymbol{\sigma}_{\boldsymbol{e}}^{\times}\right)^2 - 4\left(1 - \boldsymbol{\sigma}_{\boldsymbol{e}}^{\boldsymbol{T}} \boldsymbol{\sigma}_{\boldsymbol{e}}\right)\boldsymbol{\sigma}_{\boldsymbol{e}}^{\times}}{\left(1 + \boldsymbol{\sigma}_{\boldsymbol{e}}^{\boldsymbol{T}} \boldsymbol{\sigma}_{\boldsymbol{e}}\right)^2}, \tag{9}$$

represents the rotation matrix between the body and desired frames.

The attitude control objective can be established as

$$\tilde{R} \to I_3, \quad \text{as } t \to \infty, \tag{10}$$

which will be achieved if

$$\|\boldsymbol{\sigma}_{\boldsymbol{e}}\| \to 0, \text{ and } \|\tilde{\boldsymbol{\omega}}\| \to 0 \Rightarrow \|\dot{\boldsymbol{\sigma}}_{\boldsymbol{e}}\| \to 0. \tag{11}$$

B. Control Development

Let us define a modified state vector $\boldsymbol{r} \in \mathbb{R}^3$ as

$$\boldsymbol{r} = \dot{\boldsymbol{\sigma}}_{\boldsymbol{e}} + \alpha \boldsymbol{\sigma}_{\boldsymbol{e}}, \tag{12}$$

where $\alpha \in \mathbb{R}^{3\times3}$ is a symmetric, positive definite control gain matrix. Taking the time derivative of Eq. (12) yields

$$\dot{\boldsymbol{r}} = \frac{1}{4}\dot{B}\tilde{\boldsymbol{\omega}} + \frac{1}{4}B\left(\dot{\boldsymbol{\omega}} - \tilde{R}\dot{\boldsymbol{\omega}}_{\boldsymbol{d}} - \dot{\tilde{R}}\boldsymbol{\omega}_{\boldsymbol{d}}\right) + \alpha\dot{\boldsymbol{\sigma}}_{\boldsymbol{e}}, \tag{13}$$

with $\dot{B} = \left[-2\boldsymbol{\sigma}_{\boldsymbol{e}}^{\boldsymbol{T}}\dot{\boldsymbol{\sigma}}_{\boldsymbol{e}} I_3 + 2\dot{\sigma}_e^{\times} + 4\left(\dot{\boldsymbol{\sigma}}_{\boldsymbol{e}}\boldsymbol{\sigma}_{\boldsymbol{e}}^{\boldsymbol{T}}\right)\right]$.

Using the dynamics in Eq. (4), and the fact that $\dot{\tilde{R}} = -\tilde{\omega}^{\times}\tilde{R}$, we obtain

$$\dot{\boldsymbol{r}} = \frac{1}{4}\dot{B}\tilde{\boldsymbol{\omega}} + \frac{1}{4}B\left(J^{-1}\left(-\boldsymbol{\omega}\times(J\boldsymbol{\omega} + J_{RW}G\boldsymbol{\Omega}) + Y\tilde{\boldsymbol{\theta}} + Y\hat{\boldsymbol{\theta}} + \boldsymbol{d}\right) - \tilde{R}\dot{\boldsymbol{\omega}}_{\boldsymbol{d}} + \tilde{\omega}^{\times}\tilde{R}\boldsymbol{\omega}_{\boldsymbol{d}}\right) + \alpha\dot{\boldsymbol{\sigma}}_{\boldsymbol{e}}, \tag{14}$$

where the estimation error is defined as $\tilde{\boldsymbol{\theta}} = \boldsymbol{\theta} - \hat{\boldsymbol{\theta}} \in \mathbb{R}^N$, and therefore the fact that $Y\boldsymbol{\theta} = Y\tilde{\boldsymbol{\theta}} + Y\hat{\boldsymbol{\theta}}$ was used.

Since the control input $\boldsymbol{u}$, i,e., RW torques, is embedded in the $Y\boldsymbol{\theta}$ term, let the auxiliary control signal $\boldsymbol{u_d} \in \mathbb{R}^3$ be designed as

$$\boldsymbol{u_d} = \boldsymbol{\omega} \times (J\boldsymbol{\omega} + J_{RW}G\boldsymbol{\Omega}) + J\tilde{R}\dot{\boldsymbol{\omega}}_{\boldsymbol{d}} - J\tilde{\omega}^{\times}\tilde{R}\boldsymbol{\omega_d} + 4JB^{-1}\left[-\frac{1}{4}\dot{B}\tilde{\boldsymbol{\omega}} - \alpha\dot{\boldsymbol{\sigma}}_{\boldsymbol{e}} - K\boldsymbol{r} - \beta\boldsymbol{\sigma_e}\right], \tag{15}$$

where $\beta \in \mathbb{R}_{>0}$ is a constant control gain.

Based on Eqs. (14) and (15), the adaptation law is designed as

$$\dot{\hat{\boldsymbol{\theta}}} = \text{proj}\left\{\frac{1}{4}\Gamma Y^T\left(J^{-1}\right)^T B^T\boldsymbol{r} + \Gamma K_1\sum_{i=1}^{N_s}\mathcal{Y}_i^T\left(J\boldsymbol{\omega}(t) - J\boldsymbol{\omega}(t-\Delta t) + \boldsymbol{\mathcal{U}_i} - \mathcal{Y}_i\hat{\boldsymbol{\theta}}\right)\right\}, \tag{16}$$

where $\Gamma,\ K_1,\ \in \mathbb{R}^{N\times N}$ are constant, positive definite adaptation gain matrices, $\hat{\boldsymbol{\theta}} \in \mathbb{R}^N$ is the estimate of $\boldsymbol{\theta}$, $\Delta t \in \mathbb{R}_{>0}$ is the time between samples, proj$\{\cdot\}$ represents a projection algorithm to keep $\hat{\boldsymbol{\theta}}$ within known, user-defined bounds [29], $N_s \in \mathbb{Z}_{>0}$ is the number of input-output data pairs used for online parameter estimation, and the terms $\boldsymbol{\mathcal{U}_i} \in \mathbb{R}^3$, and $\mathcal{Y}_i \in \mathbb{R}^{3\times N}$ are defined as

$$\boldsymbol{\mathcal{U}_i}(\Delta t, t_i) = \int_{t_i-\Delta t}^{t_i}\left(\boldsymbol{\omega}(\tau) \times (J\boldsymbol{\omega}(\tau) + J_{RW}G\boldsymbol{\Omega}(\tau))\right)d\tau, \tag{17}$$

$$\mathcal{Y}_i(\Delta t, t_i) = \int_{t_i-\Delta t}^{t_i} Y(\tau)d\tau. \tag{18}$$

The input-output data pairs in Eq. (16) refer to the expressions in Eq. (17) and Eq. (18) that contain states (output) information, and control input information, since the regressor Y depends on the applied control input. The data pairs are collected via the summation without requiring memory buffers and the number of samples N_s is truncated once the subsequent FE condition is satisfied.

Based on the definition of $Y\boldsymbol{\theta}$, the term $Y\hat{\boldsymbol{\theta}}$ can be written as $Y\hat{\boldsymbol{\theta}} = G\hat{\Phi}\boldsymbol{u}$, where $\hat{\Phi}$ is the estimate of Φ. Note that the actual control input is $\boldsymbol{u}$. Then, we can set

$$Y\hat{\boldsymbol{\theta}} = \boldsymbol{u_d}, \tag{19}$$

and recover $\boldsymbol{u}$ as

$$\boldsymbol{u} = \left(G\hat{\Phi}\right)^{\dagger}\boldsymbol{u_d}, \tag{20}$$

where $\hat{\Phi}$ can be obtained by numerically integrating Eq. (16), and $(\cdot)^{\dagger}$ is the Moore-Penrose pseudo-inverse of $(\cdot)$.

In solving Eq. (20) for a system which has degraded RW performance, it is ideal for the controller to not only identify

which RW(s) are experiencing performance loss, but to then reallocate the required control torques to the remaining operational RWs so that the full control authority of the vehicle is maintained. Since the system of equations in Eq. (20) is underdetermined, using the Moore-Penrose pseudo-inverse ensures finding the solution that minimizes the 2-norm of $\boldsymbol{u}$, allocating less torque on the degraded RW(s) to keep the 2-norm low [1, 30]. The proof for the minimization of the 2-norm of the matrix as a result of using the Moore-Penrose pseudo-inverse is given in [1].

Assumption 4. The system is sufficiently excited over a finite duration of time. Then, there exists a finite time $T \in \mathbb{R}_{>0}$ such that

$$\lambda_{min}\left\{\sum_{i=1}^{N_s} \boldsymbol{\mathcal{Y}}_i^T \boldsymbol{\mathcal{Y}}_i\right\} \geq \bar{\lambda}, \tag{21}$$

where $\lambda_{min}\{\cdot\}$ denotes the minimum eigenvalue of the finite excitation condition matrix $\{\cdot\}$, and $\bar{\lambda} \in \mathbb{R}_{>0}$ is a user-defined threshold.

Prior to the FE condition being satisfied, the controller, through the second term (i.e., ICL term) in Eq. 16 collects the input-output data pairs to learn and increase λ_{min} in Eq. 21, but only the first term (i.e., gradient-based term) is effectively used to update the parameter estimates. Once the FE condition is met, the ICL term is enabled and data is no longer appended to Eq. 16. The amount of data needed to estimate the uncertain model parameters requires tuning of the $\bar{\lambda}$ threshold which is used to indicate when the system has been sufficiently excited. The selection of $\bar{\lambda}$ must be done in such a way that the system collects enough data from regions of the state space that are relevant for operation.

IV. Stability Analysis

To facilitate the subsequent stability analysis, let us define the composite state vectors $\boldsymbol{y} = \left[\boldsymbol{r^T},\ \boldsymbol{\sigma_e^T}\right]^T \in \mathbb{R}^6$, and $\boldsymbol{\eta} = [\boldsymbol{r^T},\ \boldsymbol{\sigma_e^T},\ \boldsymbol{\tilde{\theta}^T}]^T \in \mathbb{R}^{6+N}$, the known bounding constants $\underline{\gamma},\ \bar{\gamma},\ \bar{\epsilon}_1,\ \bar{\epsilon}_2,\ \underline{\kappa},\ \bar{\kappa} \in \mathbb{R}_{>0}$, and the convergence rate constants $\mu = \min\left(\lambda_{\min}\{K\} - \frac{1}{2},\ \beta\lambda_{\min}\{\alpha\}\right) \in \mathbb{R}_{>0}$, and $\zeta = \min\left(\mu,\ \lambda_{min}\left\{K_1 \sum_{i=1}^{N_s} \boldsymbol{\mathcal{Y}}_i^T \boldsymbol{\mathcal{Y}}_i\right\}\right) \in \mathbb{R}_{>0}$.

The stability analysis is divided into two theorems: the first theorem considers the stability of the closed-loop system before the finite excitation condition in Assumption 4 is satisfied, i.e., $t < T$; and the second theorem presents the stability of the closed-loop system after t=T.

Theorem 1. *Given the spacecraft attitude dynamics in Eqs. (1) and (2), along with the adaptive update law in Eq. (16), the controller proposed in Eq. (15) ensures the attitude tracking error is globally ultimately bounded such that*

$$\|\boldsymbol{y}(t)\| \leq \sqrt{\frac{\bar{\gamma}}{\underline{\gamma}}}\|\boldsymbol{y}(\boldsymbol{0})\| \exp\!\left(-\frac{\mu}{2\bar{\gamma}}(t - t_0)\right) + \sqrt{\frac{\bar{b}\,\underline{b}}{\underline{\gamma}}} + \sqrt{\frac{c\bar{\gamma}}{\mu\underline{\gamma}}} \quad t_0 \leq t < T. \tag{22}$$

Proof. Consider the Lyapunov candidate function $V : \mathbb{R}^{6+N} \rightarrow \mathbb{R}_{>0}$

$$V(t) = \frac{1}{2}\boldsymbol{r}^{\boldsymbol{T}}\boldsymbol{r} + \frac{\beta}{2}\boldsymbol{\sigma}_{\boldsymbol{e}}^{\boldsymbol{T}}\boldsymbol{\sigma}_{\boldsymbol{e}} + \frac{1}{2}\tilde{\boldsymbol{\theta}}^T\Gamma^{-1}\tilde{\boldsymbol{\theta}}, \tag{23}$$

bounded as

$$\underline{\gamma}\|\boldsymbol{y}\|^2 + \underline{b} \leq V(t) \leq \overline{\gamma}\|\boldsymbol{y}\|^2 + \overline{b}. \tag{24}$$

Taking the time derivative of Eq. (23), substituting Eqs. (12), (15) and (19), we get

$$\dot{V}(t) = \boldsymbol{r}^T\left(\frac{1}{4}BJ^{-1}Y\tilde{\boldsymbol{\theta}} - K\boldsymbol{r} + \frac{1}{4}BJ^{-1}\boldsymbol{d}\right) - \beta\boldsymbol{\sigma}_{\boldsymbol{e}}^T\alpha\boldsymbol{\sigma}_{\boldsymbol{e}} - \tilde{\boldsymbol{\theta}}^T\Gamma^{-1}\dot{\hat{\boldsymbol{\theta}}}. \tag{25}$$

The expression in Eq. (16) can be represented in the non-implementable -but useful for the stability analysis- form

$$\dot{\hat{\boldsymbol{\theta}}} = \text{proj}\left\{\frac{1}{4}\Gamma Y^T\left(J^{-1}\right)^T B^T\boldsymbol{r} + \Gamma K_1\sum_{i=1}^{N_s}\mathcal{Y}_i^T\left(\mathcal{D}_i + \mathcal{Y}_i\tilde{\boldsymbol{\theta}}\right)\right\}, \tag{26}$$

where $\mathcal{D}_i = \int_{t_i-\Delta t}^{t_i}\boldsymbol{d}(\tau)d\tau$. Eq. (26) can be plugged in Eq. (25) to obtain

$$\dot{V}(t) = -\boldsymbol{r}^T K\boldsymbol{r} - \beta\boldsymbol{\sigma}_{\boldsymbol{e}}^T\alpha\boldsymbol{\sigma}_{\boldsymbol{e}} + \frac{1}{4}\boldsymbol{r}^T BJ^{-1}\boldsymbol{d} - \tilde{\boldsymbol{\theta}}^T K_1\sum_{i=1}^{N_s}\mathcal{Y}_i^T\mathcal{Y}_i\tilde{\boldsymbol{\theta}} - \tilde{\theta}^T K_1\sum_{i=1}^{N_s}\mathcal{Y}_i^T\mathcal{D}_i. \tag{27}$$

Under Assumption 4, and before $t = T$, Eq. (27) can be upper bounded as

$$\dot{V}(t) \leq -\lambda_{min}\{K\}\|\boldsymbol{r}\|^2 - \beta\lambda_{min}\{\alpha\}\|\boldsymbol{\sigma}_{\boldsymbol{e}}\|^2 + \bar{\epsilon}_1 + \bar{\epsilon}_2\|\boldsymbol{r}\|, \tag{28}$$

where $\left\|\tilde{\boldsymbol{\theta}}^T K_1\sum_{i=1}^{N_s}\mathcal{Y}_i^T\mathcal{D}_i\right\| \leq \bar{\epsilon}_1$, $\left\|\frac{1}{4}\boldsymbol{r}^T BJ^{-1}\boldsymbol{d}\right\| \leq \bar{\epsilon}_2\|\boldsymbol{r}\|$. The preceding bounds come from the use of a smooth projection algorithm in Eq. (16) and the following facts: From the definition of the regression matrix in Eq. (5), $\mathcal{Y}_i$ is evaluated from the torques that each RW generates, thus due to physical limitations of the RW's, the torques and regression matrix are bounded. Since the disturbance $\boldsymbol{d}$ is bounded by δ, then $\mathcal{D}_i$ is bounded, as N_s is finite. J^{-1} and $\boldsymbol{d}$ are bounded by constants by definition, and B is also bounded by a constant due to the use of the MRP short rotation, i.e., $\|\boldsymbol{\sigma}_{\boldsymbol{e}}\| \leq 1$ [28, 31].

Applying Young's inequality as $\bar{\epsilon}_2\|\boldsymbol{r}\| \leq \frac{1}{2}\|\boldsymbol{r}\|^2 + \frac{1}{2}\bar{\epsilon}_2^2$, yields the expression

$$\dot{V}(t) \leq -\mu\|\boldsymbol{y}\|^2 + c, \tag{29}$$

with $c = \bar{\epsilon}_1 + \frac{1}{2}\bar{\epsilon}_2^2$. Using the upper bound in the Eq. (24), the expression can be rewritten as

$$\dot{V}(t) \leq -\frac{\mu}{\overline{\gamma}}V(t) + \left(\frac{\mu\bar{b}}{\overline{\gamma}} + c\right) \tag{30}$$

Using the comparison Lemma, we obtain the following

$$V(t) \leq \left(V(t_0) - \bar{b} - \frac{c\overline{\gamma}}{\mu}\right) \exp\left(-\frac{\mu}{\overline{\gamma}}(t - t_0)\right) + \bar{b} + \frac{c\overline{\gamma}}{\mu} \quad t_0 \leq t < T. \tag{31}$$

Using the quadratic bounds in Eq. (24), yields the upper bound for $\|\boldsymbol{y}(t)\|$ in Eq. (22), where the tracking error decays with rate $\mu/(2\overline{\gamma})$ until it reaches the bound $\sqrt{(\bar{b} - \underline{b})/\underline{\gamma}} + \sqrt{c\,\overline{\gamma}/(\mu\,\underline{\gamma})}$.

□

The subsequent theorem, assumes that the finite excitation condition in Eq. (21) is satisfied, which will allow online estimation of the RW health matrix Φ. The purpose of estimating Φ is to obtain information about the RWs performance. For instance, in the case of total failure on the i^{th} RW the spacecraft will not experience torques coming from that RW. This will force the estimation of $\Phi's\ i^{th}$ diagonal entry to approach zero provided guaranteed convergence of $\tilde{\boldsymbol{\theta}}$, so that no torques from this RW influence the attitude dynamics.

Theorem 2. *For the spacecraft attitude dynamics described in Eqs. (1) and (2), the controller in Eq. (15) and the adaptation law in Eq. (16) ensure that the tracking errors and health estimates are globally ultimately bunded such that*

$$\|\boldsymbol{\eta}(t)\| \leq \sqrt{\frac{\bar{\kappa}}{\underline{\kappa}}}\,\|\boldsymbol{\eta}(T)\| \exp\left(-\frac{\zeta}{2\bar{\kappa}}(t - T)\right) + \sqrt{\frac{c\bar{\kappa}}{\zeta\underline{\kappa}}} \;\; \forall\, t \geq T. \tag{32}$$

Proof. The bounds for the Lyapunov function in Eq. (23) can be expressed as

$$\underline{\kappa}\,\|\boldsymbol{\eta}\|^2 \leq V(t) \leq \bar{\kappa}\,\|\boldsymbol{\eta}\|^2, \tag{33}$$

Since the matrix $\sum_{i=1}^{N_s} \boldsymbol{\mathcal{Y}}_i^T \boldsymbol{\mathcal{Y}}_i$ is positive definite after $t = T$, i.e., the finite excitation condition is satisfied, then the upper bound for $\dot{V}$ can be rewritten in a compact form as

$$\dot{V}(t) \leq -\zeta\,\|\boldsymbol{\eta}\|^2 + c. \tag{34}$$

Applying the comparison lemma [20] and using the bounds in Eq. (33) yields

$$V(t) \leq \left(V(T) - \frac{c\bar{\kappa}}{\zeta}\right) \exp\left(-\frac{\zeta}{\bar{\kappa}}(t - T)\right) + \frac{c\bar{\kappa}}{\zeta} \;\; \forall\, t \geq T. \tag{35}$$

Using the bounds in Eq. (33) to express Eq. (35) in terms of $\|\boldsymbol{\eta}(t)\|$ and $\|\boldsymbol{\eta}(T)\|$ yields the result in Eq. (32). The finite excitation results in the exponential decay of the tracking error to reach the bound $\sqrt{(c\bar{\kappa})\,/\,(\zeta\underline{\kappa})}$.

□

V. Simulation

To validate the controller's capability of estimating the uncertain health parameters of the RWs, we devise 5 separate scenarios. In all simulation scenarios, the same attitude reference commands are used. In each of the first 4 scenarios, the attitude reference is initially set to align the body and ECI frames to then alternate between the initial configuration and Nadir pointing three times before settling with Nadir pointing. The change of attitude reference occurs every 12 minutes with the final Nadir pointing being set after 2000 seconds of simulation. The final scenario is a long time frame test aiming to evaluate the controller's performance under slow time-varying degradation. For this scenario, the attitude reference is continuously switched every 12 minutes for the entire duration of the simulation. The first two scenarios, both with 4 RWs, aim to demonstrate the health estimation capability by running the same scenario with and without the ICL term enabled.

The next pair of scenarios illustrate the capability of the controller to identify the degradation level of a certain pair of RWs. Both simulations contain six RWs with two failing to different levels. In the first scenario, the two RWs are degraded to 0% performance, i.e., complete failure, whilst in the second, RW 1 is degraded to 0% performance and RW 2 is degraded to 30% performance. The gains used for all scenarios are listed in Table 1 and other simulation parameters are listed in Table 2.

Table 1 Table of gains for the adaptive controller for all 4 Cases.

Gain	Case 1	Case 2	Case 3	Case 4
K_1	$10\ I_4$	0	$10\ I_6$	$10\ I_6$
K	$5 \times 10^{-1}\ I_3$	$5 \times 10^{-1}\ I_3$	$5 \times 10^{-2}\ I_3$	$5 \times 10^{-2}\ I_3$
α	$3 \times 10^{-2}\ I_3$	$3 \times 10^{-2}\ I_3$	$3 \times 10^{-2}\ I_3$	$3 \times 10^{-2}\ I_3$
β	5×10^{-3}	5×10^{-3}	5×10^{-3}	5×10^{-3}
Γ	$100\ I_4$	$100\ I_4$	$300\ I_6$	$300\ I_6$
$\bar{\lambda}$	1×10^{-7}	N/A	8×10^{-9}	8×10^{-9}
Φ	$diag\{1, 1, 0, 1\}$	$diag\{1, 1, 0, 1\}$	$diag\{0, 0, 1, 1, 1, 1\}$	$diag\{0, 0.3, 1, 1, 1, 1\}$

The final scenario is designed to test a limitation of the proposed controller, the assumption of constant health parameters defined in Assumption III.A. Set up similar to the first scenario, where RW 3 experiences performance degradation, it is run however for a total of 100,000 seconds. Unlike scenario 1, the degradation of RW 3 follows a logarithmic-like decay from $\sim 60\%$ to $\sim 50\%$ for the first 65,000 seconds to mimic realistic RW performance decay [32–34]. After that point, it experiences an abrupt loss in performance to 20%. Concurrently, we truncate the learning of the controller once FE condition is met by setting the upper limit in the summation, N_s in Eq. (16) to be the number of samples required to reach the verifiable FE condition. This allowed using only the necessary samples to make the estimation to reduce susceptibility to the time-varying health matrix Φ. Lastly, to further improve the estimation given the challenging scenario, the ICL term and estimates were reset every 10,000 seconds, to allow for the controller to

Table 2 Table of configuration parameters for the RWs for each simulation.

Parameter	Value	Units
Satellite Mass	25	kg
J	$diag\{0.4333, 0.7042, 0.7042\}$	$\mathrm{kg\,m^2}$
IC $(a, e, i, \omega, \Omega, \theta)$	$\begin{bmatrix}6878 & 0 & 0.8901 & 05236 & 0.3491 & 0.7854\end{bmatrix}$	$(km, N/A, \mathrm{rad})$
IC (q, ω_b)	$\left(\begin{bmatrix}1 & 0 & 0 & 0\end{bmatrix}, \begin{bmatrix}0.0017 & 0.0087 & 0.0017\end{bmatrix}\right)$	$\left(N/A\ , \mathrm{rad\,s^{-1}}\right)$
G (4 RWs)	$\begin{bmatrix}0.5774 & -0.5774 & 0.5774 & -0.5774\\ 0.5774 & 0.5774 & -0.5774 & -0.5774\\ 0.5774 & 0.5774 & 0.5774 & 0.5774\end{bmatrix}$	N/A
G (6 RWs)	$\begin{bmatrix}0.5 & 0.5 & 0.5 & 0.5 & 0.5 & 0.5\\ 0 & 0.75 & 0.75 & 0 & -0.75 & -0.75\\ 0.866 & 0.433 & -0.433 & -0.866 & -0.433 & 0.433\end{bmatrix}$	N/A
J_{RW}	5.7296×10^5	$\mathrm{kg\,m^2}$
Max RW Torque	20×10^{-3}	N m
Max Ω (for all RWs)	1.0472×10^3	$\mathrm{rad\,s^{-1}}$

learn from the most recent batch of data. By resetting the ICL term and estimates, the entire estimation process is essentially restarted, allowing for a more accurate estimate of the health focused around the reset time. For a profile that is not constant, the old data, with a much higher health parameter, will cause a bias in the estimate of the current health, so resetting mitigates this issue, although inducing the setback that the FE condition is required to be met again for the estimates to converge. The RW health decay is in clear violation of Assumption 2. However, the decay profile occurs slowly enough so that the controller is still able to provide a reasonable estimate of the health after each reset. The benefit of the ICL method is evident in this simulation, as the relaxed FE condition, as opposed to the Persistent Excitation condition that many other adaptive controllers require [21, 22], can quickly be satisfied and verified online (in time to converge before the reset), resulting in a more updated estimate of the health parameters.

A. Cases 1 & 2: ICL term influence

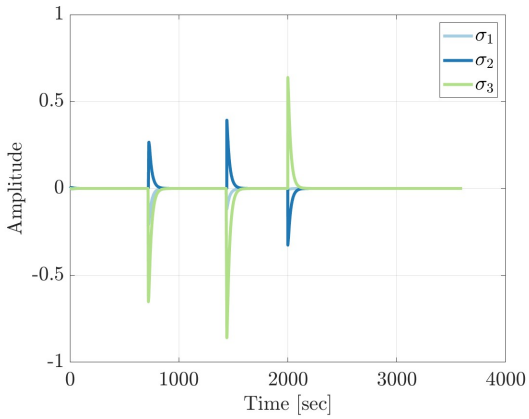
The ICL term is responsible for the input-output data collection and helps achieving convergence of the uncertain parameters' estimation error. Hence, the following two simulations illustrate the importance of this term in the controller's ability to estimate the health of the system's actuators; in this case, the health of all the RWs.

For both simulations, the time history plots for the error MRP, body angular velocity and RW angular velocity, are presented in Figs. 2a & 4a , 2b & 4b and 2c & 4c for Cases 1 & 2, respectively, to verify that the controller is accurately tracking the attitude command. The failing RW can be observed in the plot of the RWs angular velocities presented in Figs. 2c & 4c.

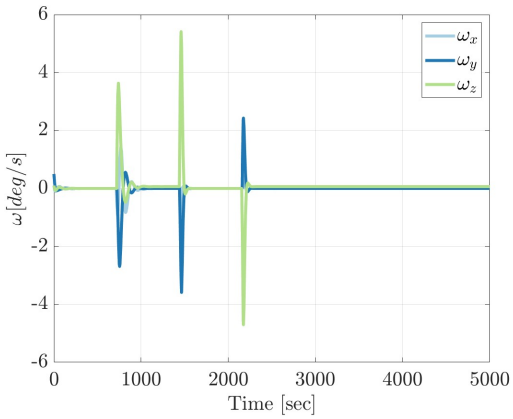
For Case 1, when the ICL term is active, we include Fig. 3a, a plot of λ, which is the minimum eigenvalue of the finite excitation condition matrix explained in Assumption 4. Because there is an user-defined threshold $\bar{\lambda}$, which can be

verified during execution, when this threshold is met it means that the system is sufficiently excited and exponential convergence of the estimation error can be guaranteed. The estimation of the uncertain parameters is presented in Figs. 3b & 4d for both simulations. Because the ICL term is deactivated in Case 2, no plot of λ is necessary.

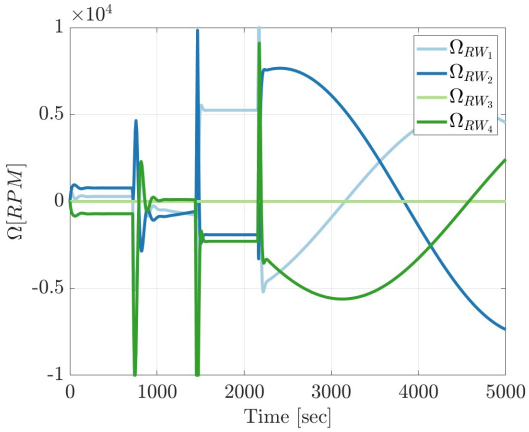
With Assumption 4 we can say that there is a time at which the FE condition is satisfied and thus the controlled has learned and can effectively allocate the correct torques to the system as required by the control law. This is evident in Fig. 5 which illustrates the applied control components in the body frame with and without ICL. Only with ICL after FE, do the allocated torques closely match the desired torques. It is also noticeable from Table 3 that once the health estimates have converged, the amount of torque applied to the failing RW is significantly lower, helping to save the health of that wheel.



(a) Error MRP

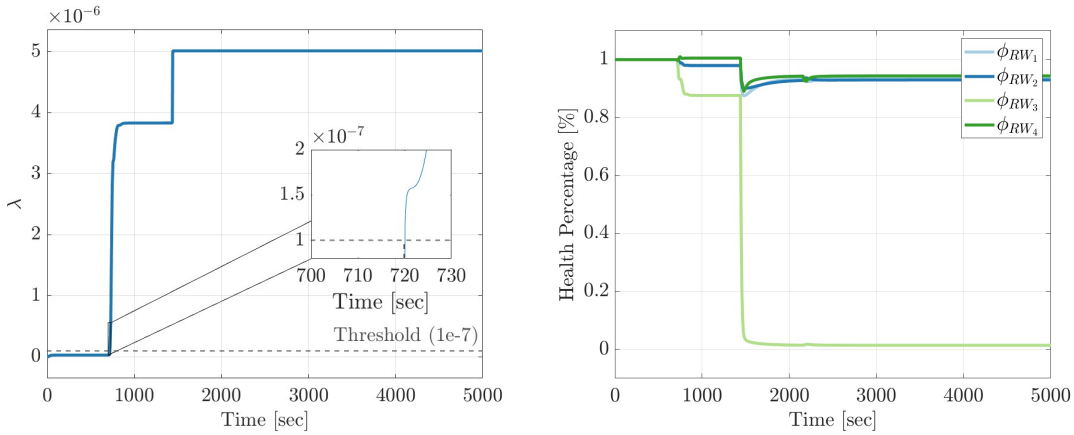


(b) Body Angular Velocity



(c) Reaction Wheel Angular Velocity

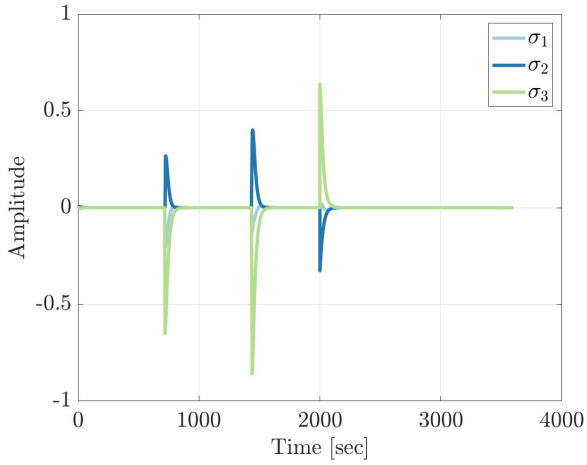
Fig. 2 Case 1: 4 RW's with ICL term activated.



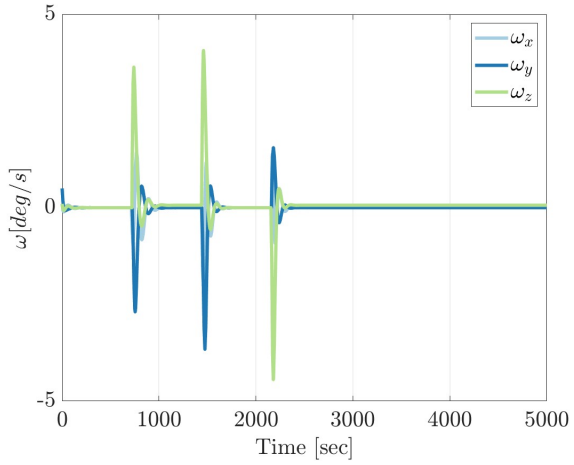
(a) Minimum Eigenvalue of Finite Excitation Condition Matrix w/ Threshold for Learning

(b) Estimated Parameters

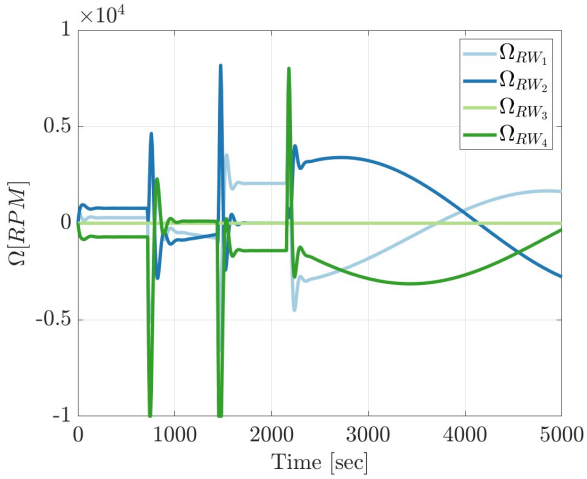
Fig. 3 Case 1: 4 RW's with ICL term activated.



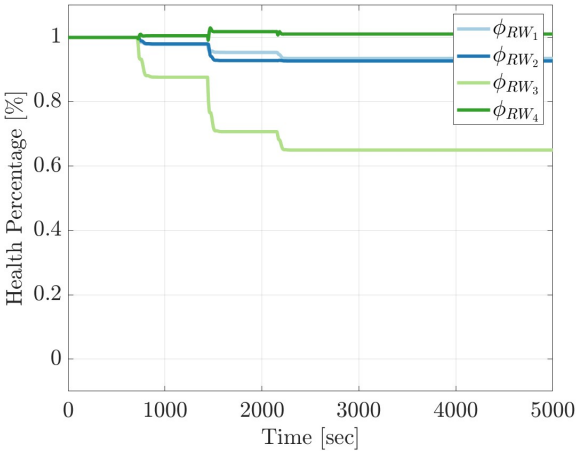
(a) Error MRP



(b) Body Angular Velocity



(c) Reaction Wheel Angular Velocity



(d) Estimated Parameters

Fig. 4 Case 2: 4 RW's with no ICL term activated.

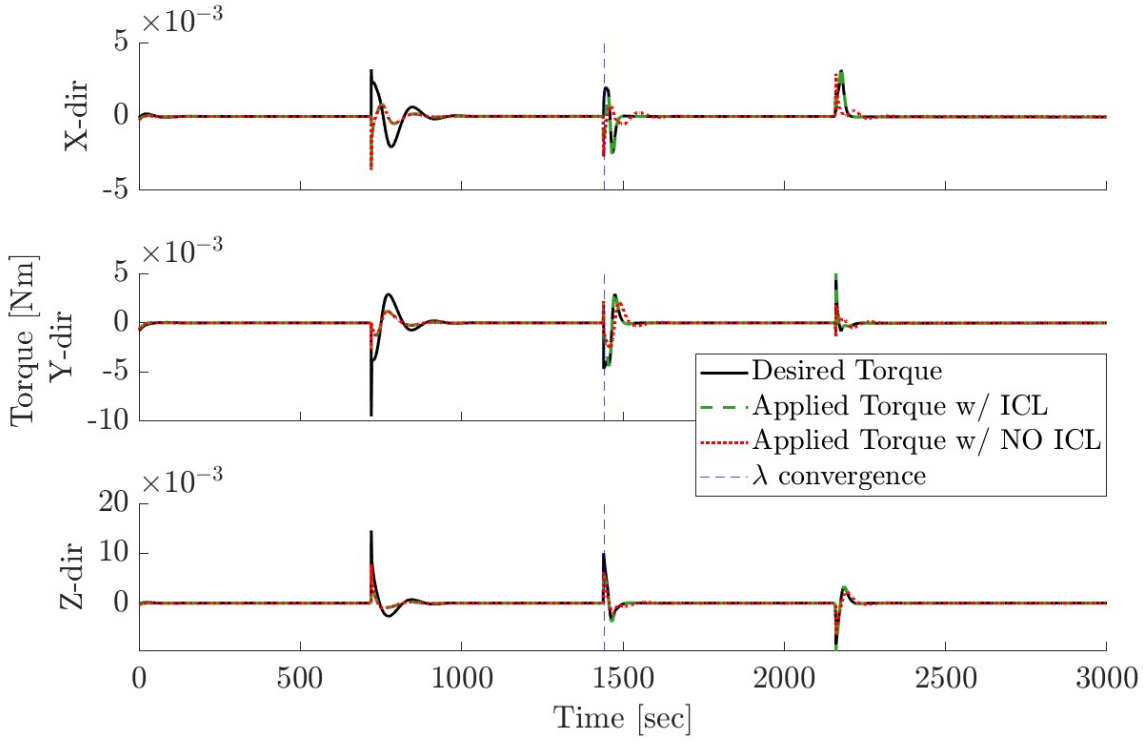
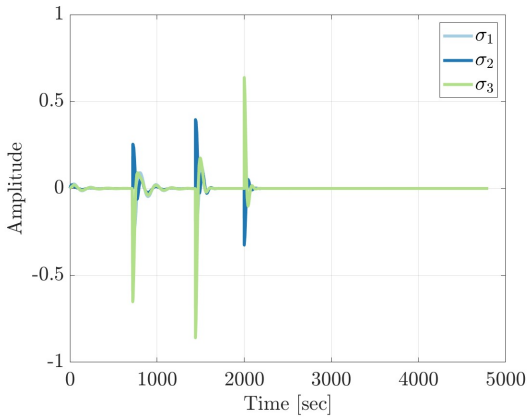


Fig. 5 Actual vs. Desired Torques with and without the ICL term.

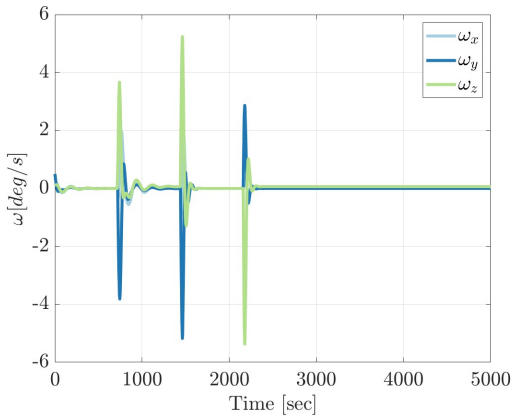
B. Cases 3 & 4: More RWs and partial degradation

As mentioned, Cases 3 & 4 are designed to test the capability of the proposed adaptive controller to perform in multiple scenarios and still deliver the expected results. Since Cases 1 & 2 were designed around a system with 4 RWs, these next two simulations contain a system with 6 RWs. The 6 RWs are arranged in a vertical, symmetric, hexagon such that there are multiple redundancies built into the system. Along with the increased number of RWs, the system in both simulations will experience the failure of two RWs, to test the ability of the controller to handle multiple failures. In Case 3, RWs 1 and 2 are set to experience total failure, whilst in Case 4 the same RWs are set to experience degraded performance, only able to output 0 and 30% of their nominal torque preformance, respectively. Like Cases 1 & 2, for both scenarios we display the time history of their error MRP in Figs. 6a & 8a, the body angular velocity in Figs. 6b & 8b and the RW angular velocities in Figs. 6c & 8c. These present the overall performance of the controller and ensure that it behaves as expected.

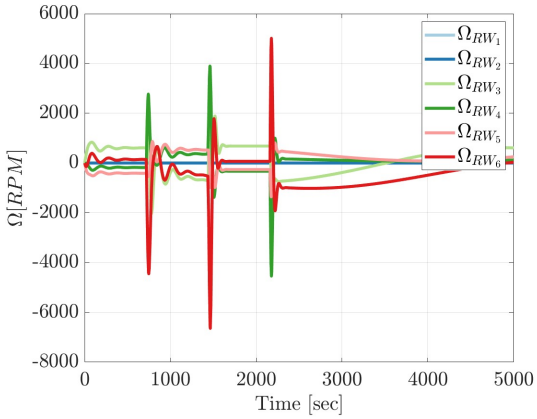
As both simulations have the ICL term active, Figs. 7a & 9a display the value of λ over time, whilst Figs. 7b & 9b show the values of the estimated uncertain RW health parameters.



(a) Error MRP

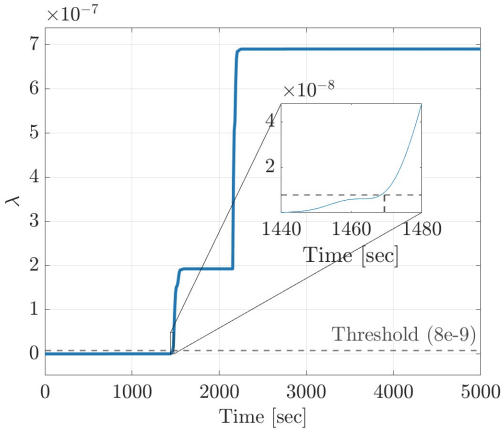


(b) Body Angular Velocity

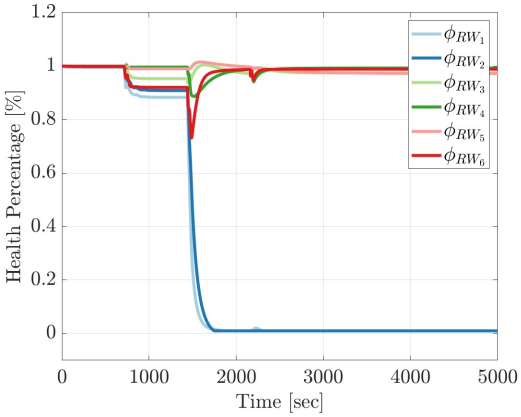


(c) Reaction Wheel Angular Velocity

Fig. 6 Case 3: 6 RW's with failure of RW's 1 & 2 set to 0%.

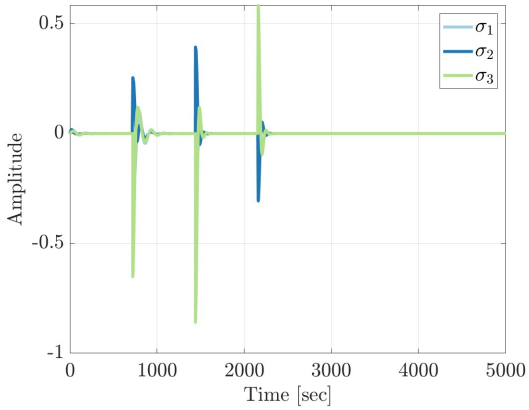


(a) Minimum Eigenvalue of Finite Excitation Condition Matrix w/ Threshold for Learning

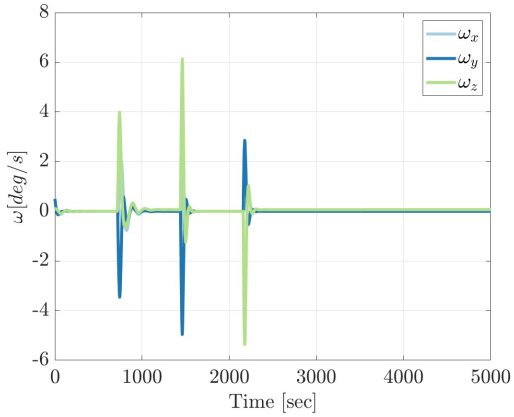


(b) Estimated Parameters

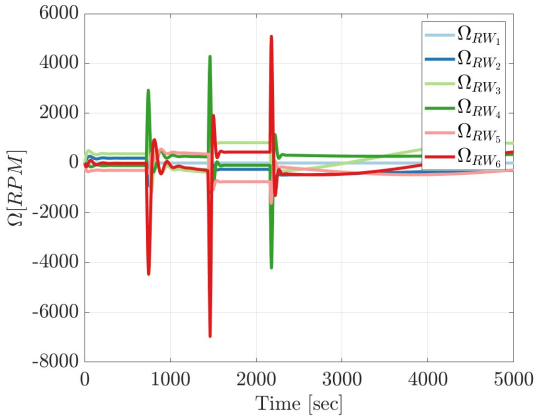
Fig. 7 Case 3: 6 RW's with failure of RW's 1 & 2 set to 0%.



(a) Error MRP

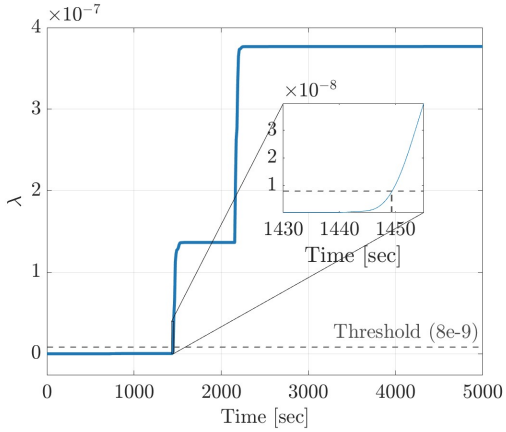


(b) Body Angular Velocity

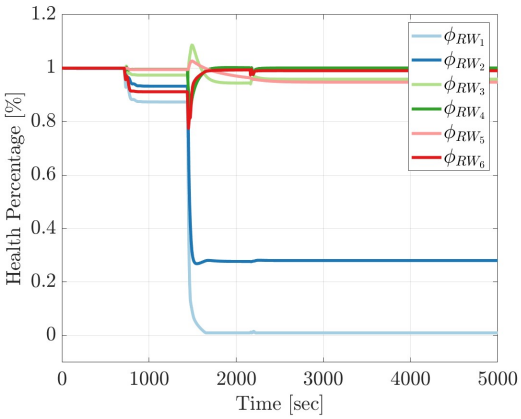


(c) Reaction Wheel Angular Velocity

Fig. 8 Case 4: 6 RW's with performance of RW 1 set to 0% and RW 2 set to 30%.



(a) Minimum Eigenvalue of Finite Excitation Condition Matrix w/ Threshold for Learning



(b) Estimated Parameters

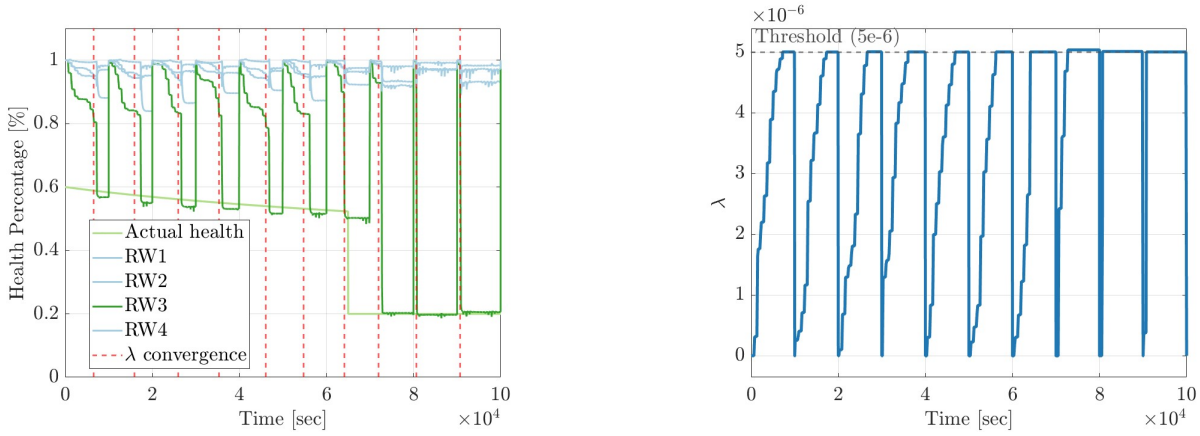
Fig. 9 Case 4: 6 RW's with performance of RW 1 set to 0% and RW 2 set to 30%.

C. Case 5: time-varying health degradation and ICL term reset

Due to the long time scale nature alternation between attitude commands, we present the results of this simulation slightly differently to the previous simulations for clarity and to also emphasize the key aspects we aim to test in this simulation. Like case 1, RW 3 experiences the performance loss and begins with a slow, gradual decay before abruptly losing a majority of its performance. The loss is modeled with the following piecewise function:

$$\theta_{RW4} = \begin{cases} -\frac{\ln(t+50000)}{\ln(50000)} + 1.6 & 0 \leq t \leq 65000 \\ 0.2 & t \geq 65000 \end{cases}$$

Figure 10 presents the estimated health of all the reaction wheels along with the true performance profile of RW 3. The red dashed lines represent the times when the FE condition is satisfied and λ has converged. When this condition is met, no more data will be added to the second term in Eq. (16) and the learning is truncated. The parameters of the controller responsible for learning are also reset every 10,000 seconds, as shown by the jump in all RW estimated parameters to 100%.



(a) Estimated health parameters of the 4 RW. **(b) ICL term convergence.**

Fig. 10 Simulation of a realistic log-like health degradation of RW3, with sudden performance loss at 65,000 seconds.

A summary of the five simulated cases is presented in Table 3. The metrics for comparison between the simulations include, whether ICL was active or not, the degradation profile of the RW and the number of RWs in the simulation setup. The next metric is the the maximum torque allocated to the degraded wheel during the third maneuver, after convergence of the ICL term (if applicable) as a percentage of what would have been the allocated torque by the same controller without any adaptation or learning. That is all healths are initialized to 1 and setting the adaptation law Eq. 16 to zero. The final two metrics are the mean errors in approximation of both the degraded and healthy wheels separately. This value is calculated analogous to a steady state error, thus is only captured once the estimation has converged.

Because of the resetting of the ICL terms in Case 5, no mean error or maximum torque for this case is presented.

Table 3 Table of simulation and error metrics for all 5 Cases

Metrics	Case 1	Case 2	Case 3	Case 4	Case 5
Degraded RW	3	3	1, 2	1, 2	3
ICL	✓	X	✓	✓	✓
Degradation Profile	0%	0%	0%, 0%	0%, 30%	Log Decay
No. of RW	4	4	6	6	4
Max. torque	6.46%	112.67%	3.66%, 4.94%	6.75%, 56.32%	N/A
Mean Error degraded RW	1.38%	64.98%	1.00%	1.48%	N/A
Mean Error healthy RW	6.38%	4.27%	1.80%	2.56%	N/A

D. Software-In-the-Loop (SIL)

To validate that the controller can be deployed and performs reliably on real hardware, we conducted a Software-in-the-Loop (SIL) simulation using an NVIDIA Jetson Nano as a target hardware platform. The objective is to assess the controller's accuracy, timing behavior, and resource utilization when it is deployed on a realistic onboard hardware.

The SIL simulation architecture includes two main components. First, the satellite simulator, developed with MATLAB/Simulink, models the spacecraft dynamics and provides satellite state data. Second, the adaptive control algorithm, implemented in Python, runs on the separate onboard computer and is the main focus of SIL testing. To facilitate communication in the SIL setup, the Robot Operating System 2 (ROS2) framework was chosen as the communication interface due to its robust communication model that ensures efficient data transfer over a shared network [35]. Additionally, ROS2 offers scalability, making it easy to add or remove components while maintaining the overall architecture. The SIL setup is available at the Space Vehicle and Robotics laboratory at the Florida Institute of Technology.

In the setup configuration shown in Fig. 11, we employ a publisher-subscriber communication model. The satellite simulator and the control algorithm act as ROS2 nodes that function as publishers and subscribers, creating a closed-loop system. ROS2 handles the data exchange between the satellite simulator and the control algorithm through predefined topics:

- `SatState`: Contains the satellite's position, velocity, quaternion, angular velocity, and reaction wheel angular velocity.
- `DesState`: Contains desired quaternions, angular velocity, and acceleration.
- `SatConfig`: Provides configuration settings, including the satellite's moment of inertia, flywheel moment of inertia, and reaction wheel configurations.
- `AdptCtrlOut`: Contains computed torques, the estimated RW health parameters, and the ICL term from the adaptive algorithm to the satellite simulator.

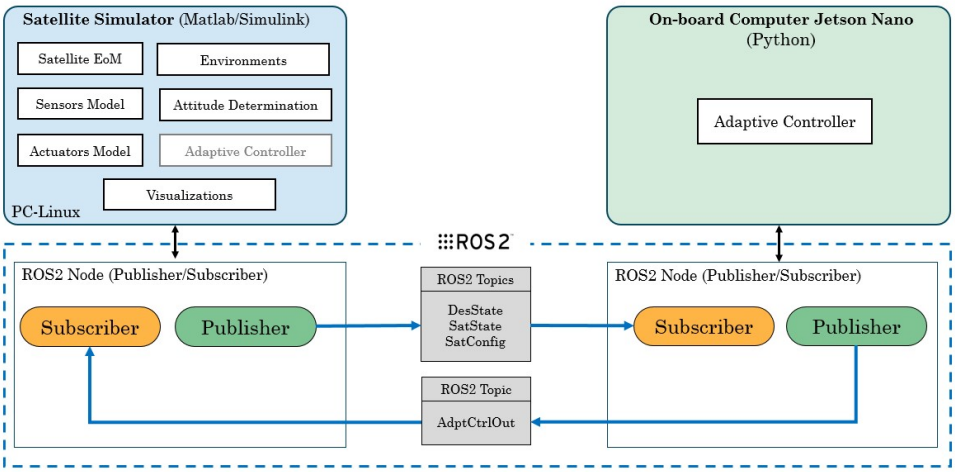


Fig. 11 Software-in-the-loop (SIL) simulation architecture.

The flow of the SIL simulation is as follows. First, the satellite simulator generates the current state of the simulated satellite based on initial conditions or previous state data. This state data is published to the relevant ROS2 topics (`SatState`, `DesState`, and `SatConfig`). Next, the controller algorithm, running on the Jetson Nano as another ROS2 node, subscribes to these topics, and processes the incoming data to compute the necessary control torques, the estimated RW health parameters, and the λ term. The computed controller outputs are published to the `AdptCtrlOut` topic. The satellite simulator, which subscribes to the `AdptCtrlOut` topic, receives the control inputs, which are then applied to update the state of the satellite in the next simulation cycle. This closed-loop process repeats with continuous interaction between the simulation and the control algorithms.

Test Case 5 was used as the representative test scenario for the SIL test and was conducted over 27.8 hours of wall-clock time (100,000 seconds), without any simulation time acceleration. The controller code was implemented in Python by translating the original MATLAB/Simulink code. The controller runs concurrently with the satellite simulator at a rate of 10 Hz. Euler integration method was used to update the estimate $\hat{\boldsymbol{\theta}}$. During the SIL test, three performance metrics, including execution time (milliseconds), CPU load (%), and memory usage (%), were recorded. The execution time was measured for every control loop step while CPU load (%), and memory usage were logged with a separate script running at 1 Hz.

E. Results and Discussion

The simulations performed in Cases 1 & 2 illustrate the importance of the ICL term. Looking at Figs. 3b and 4d it can be observed that when the ICL term is activated, the controller is able to effectively estimate the correct health value for all RWs, specifically the fact that RW 3 loses all of its effective performance. In the simulataion with no ICL term,

the estimated RW health parameters in $\hat{\boldsymbol{\theta}}$ do not converge to their real values. This is expected because the stability result from theorem 1 does not ensure convergence of the estimation error $\tilde{\boldsymbol{\theta}}$. The periods of increased growth in λ at 720, 1440, 2000 seconds are indicative of the system being excited due to the change in pointing reference. During these periods where the RWs are being more active to alter the pointing of the satellite, the system is able to gather input-output data through the ICL term until the threshold is reached and accurate estimation is guaranteed thereafter. This is evident in Fig. 3b as the exponential decay to 0% health occurs when the threshold is reached, as expected from theorem 2.

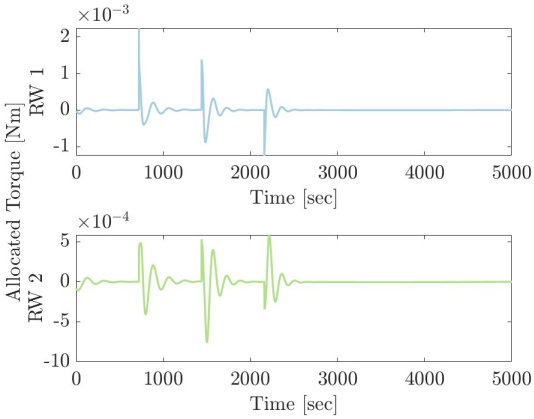
In both scenarios, it is important to note that the controller was still able to achieve accurate tracking of the attitude guidance. However, without the ICL term, the controller was unable to learn the uncertain parameters of the system. In both scenarios, the remaining RWs did not saturate with the failure of the third RW.

Cases 3 & 4 investigate the capabilities of the controller to handle a more complex system with an increased number of RWs as well as multiple failures at different levels. In Case 3, both RWs 1 and 2 were set to completely fail and this was correctly detected by the adaptive controller as seen in Fig. 7b. Once the threshold was crossed, and the system was sufficiently excited, the estimation of the health of the RWs was correctly estimated. Likewise, in Case 4, where the RWs 1 and 2 were set to be degraded to 0 and 30% performance, respectively, the controller was able to accurately determine the health and estimate these uncertainties. An important note for both Case 3 & 4, is that although two of the six RWs failed, the other four RWs did not saturate, a critical factor in satellite attitude control, showing that no excessive control effort was added once the FE condition was satisfied.

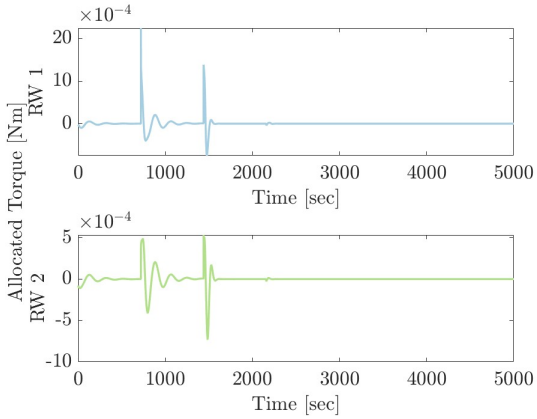
As in Cases 1 & 2, both simulations were able to accurately track the attitude guidance, showing that despite multiple failures, the system was still able to perform its control task and provide correct estimations. However, one of the key issues in the simulation scenarios, primarily evident in Case 1, was where the estimation of the health of the non-degraded RWs is not as accurate as the degraded RW. This could be explained by a number of reasons, primarily because the Simulink simulations include external disturbances and nonlinearities not accounted for in the design of the controller. These disturbances are the same upper bounded disturbances in Eq. 1. A refinement of the design along with further tuning of the key parameters outlined in Table 1 could also improve the performance of the estimation.

An interesting observation of the performance of the adaptive controller can be noted when running the same simulation, with and without the ICL term. Figs. 12a and 12b show the torque allocated by the adaptive controller to both RW 1 and 2 in each scenario with 6 total RWs. In both simulations, all parameters, but the ICL term, were kept equal, whilst RW 1 and 2 were degraded to 0% performance. The key difference between these simulations is that in Fig. 12a the ICL term is not active, whereas in Fig. 12b the ICL term is active. We can clearly see that when the ICL term is active, and the learning threshold is reached ($\sim$ 1500sec), the controller has correctly estimated that RW 1 and 2 are completely degraded, hence does not allocate any control torque to them. Figure 12a shows the contrary, where the even though the RW's are degraded, the adaptive controller does not learn the estimated health parameters and

continues trying to allocate control torques to these reaction wheels.



(a) Controller allocated torque to RW's 1 and 2 for the simulation with no ICL term active.



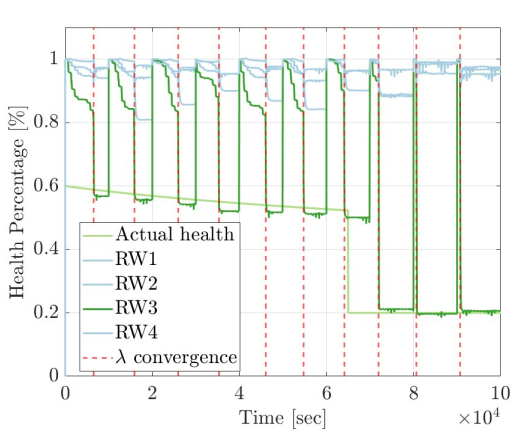
(b) Controller allocated torque to RW's 1 and 2 for the simulation with the ICL term active.

Fig. 12 Comparison of allocated torques on the degraded RW's when the ICL term is active or not.

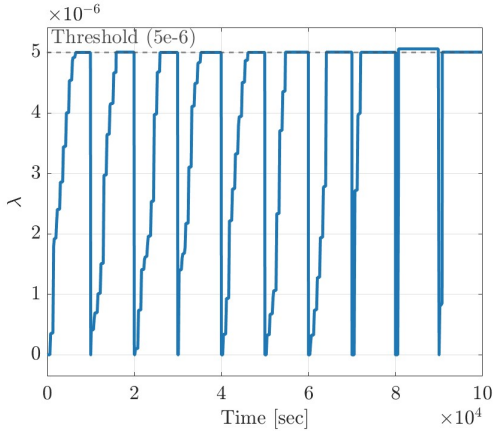
Case 5 provides insight into the performance of the controller in long-term deployment scenarios as well as how it reacts to realistic performance degradation profiles. As noted prior, the resetting of the learning parameters of the controller every 10,000 seconds allows the controller to only use the most recent data collected in order to learn the estimated health of the RWs. Figure 10 highlights that once converged, each time after the reset the estimated health of RW 3 closely follows the true degradation level, whilst the remaining fully operational RWs remain close to 100%. It is also clear that once the performance experiences the instantaneous drop to 20% and remains constant thereafter, the estimation of all RW healths significantly improves, demonstrating the importance of Assumption 3 in the development of the controller. We note that although the purpose of this case was to determine the capability of the estimation to handle tougher profiles, the attitude tracking was verified to be achieved after each change it reference.

Whilst for a relatively slow degradation profile the controller is still able to closely follow the profile, this is aided by regular resetting of the learning terms. It is evident that by resetting the learning parameters shortly after convergence of the λ term is achieved, a more accurate estimation can be achieved, as only the nearest data is utilized. Likewise, truncating the learning once the convergence is achieved, evidenced by the constant prediction values in Fig. 10, reduces the impact of the time varying θ and improves performance. It is worth noting that the other RWs are also closely estimated during this more complex simulation and the ICL resetting had no noticeable influence on the attitude tracking performance of the controller.

Figure 13 shows the result from the SIL test of the same case 5 of the simulation. It was confirmed that SIL test achieved a similar result with the simulation as shown in Figure 13a and 13b.



(a) Estimated health parameters of the 4 RW in the SIL test on the Jetson Nano.



(b) ICL term convergence for the SIL test.

Fig. 13 SIL test result of the same simulation as Case 5.

Of particular note is the reset and learning during and immediately after the degradation to 20%, around 65,000 seconds. Due to the truncated learning once the ICL term has converged, and the fact that this occurs slightly before the sudden loss of performance, it is clear that the controller has no indication of this performance loss until the next reset. Once the reset is completed and new data is used to converge the ICL term again, only then is the change in performance accounted for. It is also noted that the further from the true performance the estimates are, the faster the convergence of the ICL term, as with the 3 resets post the sudden drop in performance. This highlights the limitations of the controller and the balance that must be struck between resetting the estimate and allowing the controller to converge and give the true estimates.

Table 3 shows some key metrics that highlight the efficacy and benefits of this adaptive controller. For the Maximum torque metric, which is compared to the exact same controller minus the adaptive learning component, we can see that for Cases 1, 3 and 4 where the ICL term is active and the adaptive learning can occur, the utilization of the degraded RW is significantly lower when the health is degraded to zero, whereas in Case 4 where RW 2 is degraded to 30% performance, the controller allocates less torque relative to the degradation of the wheel. Likewise, it is clear that in Cases 1, 3 and 4 the estimation error of both the healthy and degraded RWs are low compared to the large error exhibited in Case 2 without ICL. As mentioned previously, small differences in the dynamics of the problem that are included in the simulation but not the controller development are the cause for the small errors in the converged health estimates.

Table 4 summarizes key performance metrics obtained from the SIL test. It shows that the controller performs reliably under the hardware constraints of the NVIDIA Jetson Nano. The controller executes each loop on an average time of 5.25 ms, with a standard deviation of 1.16 ms and maximum spikes reaching 40.2 ms. The execution time is well below the 100 ms time limit of the 10 Hz control loop. Even after accounting for the extra delay from Simulink-ROS2 message passing and other task schedules within the end-to-end control loop, the average execution time remains within

the 100-ms sampling period. Because the SIL result is closely aligned with the numerical simulations, it indicates that communication latency does not significantly impact control accuracy in this study. On average, the CPU load averaged over all four cores of the Jetson Nano remains below 9%, and only a peak of 25% occurs at the start of the SIL test, which is expected. Due to this low overall CPU usage by the control algorithm, a significant part of the memory is still available for running auxiliary computations and tasks that may be required by other subsystems onboard. The memory consumption is stable at an average of 17.7%, with only minor fluctuation (standard deviation of 0.04%) observed throughout the SIL simulation.

Table 4 Performance metrics of the controller from SIL test.

Metric	Execution Time (ms)	CPU Load (%)	Memory Usage (%)
Average	5.25	8.97	17.7
Maximum	40.2	25.02	17.8
Std Dev	1.16	1.38	0.04

VI. Conclusion

In this work, an ICL-based adaptive controller that can simultaneously estimate and compensate for the degradation of reaction wheels was designed. Through the implementation of an adaptive update law that incorporates integral concurrent learning, the controller successfully estimated the uncertain parameters, namely the RWs health. The finite excitation condition was verified online and the uncertain parameters estimation converged exponentially afterwards, as expected from the presented Lyapunov-based stability analysis. Posterior to the excitation condition being reached, the adaptive controller shows that it uses the learned estimated health parameters to better allocate control torques on those wheels still operable and not to those with degraded performance.

Five different scenarios were performed and the controller was able to ensure attitude tracking without saturation of the RWs, as well as estimation of the varying types of degradation level for all RWs. Although the controller was able to operate to auspiciously achieve global exponential tracking despite the multiple RW failures induced in simulation, future work could include the integration of robustness to additive disturbances in this type of controller, since these are expected to affect the stability result for both state and estimation errors.

SIL simulation results confirm that the controller can be deployed on an NVIDIA Jetson Nano. We demonstrate that the controller meets the 10 Hz execution-rate requirement while maintaining minimal resource utilization throughout the SIL simulation. These promising results set the stage for the next step, which is to further validate the controller in the hardware-in-the-loop (HIL) simulation, where a more comprehensive analysis of communication latency will be carried out.

References

[1] Evers, M., "Null Space Control Allocation and Configuration Optimization of Reaction Wheels," Master's thesis, TU Delft, 2017. URL https://resolver.tudelft.nl/uuid:9f87221b-9b27-438b-ad06-565a6ca38c54.

[2] Li, A., Liu, M., Cao, X., and Liu, R., "Adaptive quantized sliding mode attitude tracking control for flexible spacecraft with input dead-zone via Takagi-Sugeno fuzzy approach," *Information Sciences*, Vol. 587, 2022, pp. 746–773. https://doi.org/https://doi.org/10.1016/j.ins.2021.11.002, URL https://www.sciencedirect.com/science/article/pii/S0020025521011129.

[3] Xiao, Y., Ruiter, A. d., Ye, D., and Sun, Z., "Adaptive Fault-Tolerant Attitude Tracking Control for Flexible Spacecraft With Guaranteed Performance Bounds," *IEEE Transactions on Aerospace and Electronic Systems*, Vol. 58, No. 3, 2022, pp. 1922–1940. https://doi.org/10.1109/TAES.2021.3123295.

[4] Liang, X., Wang, Q., Hu, C., and Dong, C., "Observer-based H∞ fault-tolerant attitude control for satellite with actuator and sensor faults," *Aerospace Science and Technology*, Vol. 95, 2019, p. 105424. https://doi.org/https://doi.org/10.1016/j.ast.2019.105424, URL https://www.sciencedirect.com/science/article/pii/S1270963819300021.

[5] Xiao, B., Yin, S., and Gao, H., "Reconfigurable Tolerant Control of Uncertain Mechanical Systems With Actuator Faults: A Sliding Mode Observer-Based Approach," *IEEE Transactions on Control Systems Technology*, Vol. 26, No. 4, 2018, pp. 1249–1258. https://doi.org/10.1109/TCST.2017.2707333.

[6] Tan, C. P., and Edwards, C., "Sliding mode observers for robust detection and reconstruction of actuator and sensor faults," *International Journal of Robust and Nonlinear Control*, Vol. 13, No. 5, 2003, pp. 443–463. https://doi.org/https://doi.org/10.1002/rnc.723, URL https://onlinelibrary.wiley.com/doi/abs/10.1002/rnc.723.

[7] Sun, X., Wang, X., and Lin, S., "Multi-Fault Diagnosis Approach Based on Updated Interacting Multiple Model for Aviation Hydraulic Actuator," *Information*, Vol. 11, No. 9, 2020. https://doi.org/10.3390/info11090410, URL https://www.mdpi.com/2078-2489/11/9/410.

[8] Mokhtari, S., Abbaspour, A., Yen, K. K., and Sargolzaei, A., "Neural Network-Based Active Fault-Tolerant Control Design for Unmanned Helicopter with Additive Faults," *Remote Sensing*, Vol. 13, No. 12, 2021. https://doi.org/10.3390/rs13122396, URL https://www.mdpi.com/2072-4292/13/12/2396.

[9] M. Sadigh, S., Kashaninia, A., and Dehghan, S. M. M., "Adaptive sliding mode fault-tolerant control for satellite attitude tracking system," *Advances in Space Research*, Vol. 71, No. 3, 2023, pp. 1784–1805. https://doi.org/https://doi.org/10.1016/j.asr.2022.09.064, URL https://www.sciencedirect.com/science/article/pii/S0273117722009206.

[10] Xie, X., Sheng, T., Zhang, Y., Wang, J., and Chen, X., "Adaptive Fault-Tolerant Attitude Control for Rigid Spacecraft With Disturbances and Uncertainties," *2023 China Automation Congress (CAC)*, 2023, pp. 596–600. https://doi.org/10.1109/CAC59555.2023.10452054.

[11] Xiao, Y., Ruiter, A. d., Ye, D., and Sun, Z., "Adaptive Fault-Tolerant Attitude Tracking Control for Flexible Spacecraft With Guaranteed Performance Bounds," *IEEE Transactions on Aerospace and Electronic Systems*, Vol. 58, No. 3, 2022, pp. 1922–1940. https://doi.org/10.1109/TAES.2021.3123295.

[12] Zhao, Z., Wang, T., Peng, F., and Yu, J., "Adaptive Fuzzy Resilient Decentralized Control for Nonlinear Large-Scale CPSs Under DoS Attacks," *IEEE Transactions on Fuzzy Systems*, Vol. 32, No. 10, 2024, pp. 5899–5909. https://doi.org/10.1109/TFUZZ.2024.3434726.

[13] Zhao, Z., Wang, T., Yu, J., and Basin, M. V., "Bilateral Cooperative Control of Nonlinear Multiagent Systems With State and Output Quantification," *IEEE Transactions on Cybernetics*, Vol. 55, No. 6, 2025, pp. 2949–2957. https://doi.org/10.1109/TCYB.2025.3545144.

[14] Fazlyab, A. R., Saberi, F. F., and Kabganian, M., "Fault-tolerant attitude control of the satellite in the presence of simultaneous actuator and sensor faults," *Scientific Reports*, Vol. 13, 2023, p. 20802. https://doi.org/10.1038/s41598-023-48243-w, URL https://doi.org/10.1038/s41598-023-48243-w.

[15] Mei, Y., Luo, D., Gong, K., and Liao, Y., "Disturbance observer-based event-triggered fault tolerant control for coupled spacecraft on SE(3) with actuator saturation and misalignment," *Nonlinear Dynamics*, Vol. 111, 2023, pp. 17987–18011. https://doi.org/10.1007/s11071-023-08751-w.

[16] Chen, Z., "Satellite Reaction Wheel Fault Detection Based on Adaptive Threshold Observer," *2021 Global Reliability and Prognostics and Health Management (PHM-Nanjing)*, 2021, pp. 1–6. https://doi.org/10.1109/PHM-Nanjing52125.2021.9612876.

[17] Alidadi, M., and Rahimi, A., "Fault Diagnosis of Lubrication Decay in Reaction Wheels Using Temperature Estimation and Forecasting via Enhanced Adaptive Particle Filter," *Sensors*, Vol. 23, No. 3, 2023. https://doi.org/10.3390/s23031474, URL https://www.mdpi.com/1424-8220/23/3/1474.

[18] Park, H. J., Kim, S., Lee, J., Kim, N. H., and Choi, J.-H., "System-level prognostics approach for failure prediction of reaction wheel motor in satellites," *Advances in Space Research*, Vol. 71, No. 6, 2023, pp. 2691–2701. https://doi.org/https://doi.org/10.1016/j.asr.2022.11.028, URL https://www.sciencedirect.com/science/article/pii/S027311772201050X.

[19] Xiao, B., Hu, Q., Singhose, W., and Huo, X., "Reaction Wheel Fault Compensation and Disturbance Rejection for Spacecraft Attitude Tracking," *Journal of Guidance, Control, and Dynamics*, Vol. 36, No. 6, 2013, pp. 1565–1575. https://doi.org/10.2514/1.59839, URL https://doi.org/10.2514/1.59839.

[20] Khalil, H., *Nonlinear Systems*, Pearson Education, Prentice Hall, 2002. URL https://books.google.com/books?id=t_d1QgAACAAJ.

[21] Ioannou, P., and Sun, J., *Robust Adaptive Control*, Dover Books on Electrical Engineering Series, Dover Publications, Incorporated, 2012. URL https://books.google.com/books?id=pXWFY_vbg1MC.

[22] Sastry, S., and Bodson, M., *Adaptive Control: Stability, Convergence and Robustness*, Dover Books on Electrical Engineering Series, Dover Publications, 2011. URL https://books.google.com/books?id=-cOviBa9pR8C.

[23] Chowdhary, G., Yucelen, T., Mühlegg, M., and Johnson, E. N., "Concurrent learning adaptive control of linear systems with exponentially convergent bounds," *International Journal of Adaptive Control and Signal Processing*, Vol. 27, No. 4, 2013, pp. 280–301. https://doi.org/https://doi.org/10.1002/acs.2297, URL https://onlinelibrary.wiley.com/doi/abs/10.1002/acs.2297.

[24] Parikh, A., Kamalapurkar, R., and Dixon, W. E., "Integral concurrent learning: Adaptive control with parameter convergence using finite excitation," *International Journal of Adaptive Control and Signal Processing*, Vol. 33, No. 12, 2019, pp. 1775–1787. https://doi.org/https://doi.org/10.1002/acs.2945, URL https://onlinelibrary.wiley.com/doi/abs/10.1002/acs.2945.

[25] Bell, Z. I., Nezvadovitz, J., Parikh, A., Schwartz, E. M., and Dixon, W. E., "Global Exponential Tracking Control for an Autonomous Surface Vessel: An Integral Concurrent Learning Approach," *IEEE Journal of Oceanic Engineering*, Vol. 45, No. 2, 2020, pp. 362–370. https://doi.org/10.1109/JOE.2018.2880622.

[26] Riano-Rios, C., Bevilacqua, R., and Dixon, W. E., "Differential drag-based multiple spacecraft maneuvering and on-line parameter estimation using integral concurrent learning," *Acta Astronautica*, Vol. 174, 2020, pp. 189–203. https://doi.org/https://doi.org/10.1016/j.actaastro.2020.04.059, URL https://www.sciencedirect.com/science/article/pii/S0094576520302745.

[27] Sun, R., Riano-Rios, C., Bevilacqua, R., Fitz-Coy, N. G., and Dixon, W. E., "CubeSat Adaptive Attitude Control with Uncertain Drag Coefficient and Atmospheric Density," *Journal of Guidance, Control, and Dynamics*, Vol. 44, No. 2, 2021, pp. 379–388. https://doi.org/10.2514/1.G005515, URL https://doi.org/10.2514/1.G005515.

[28] Schaub, H., and Junkins, J. L., *Analytical Mechanics of Space Systems*, 4$^{\text{th}}$ ed., AIAA Education Series, Reston, VA, 2018. https://doi.org/10.2514/4.105210.

[29] Dixon, W. E., Behal, A., Dawson, D. M., and Nagarkatti, S. P., *Nonlinear Control of Engineering Systems: A Lyapunov-Based Approach*, Control Engineering, Springer Science+Business Media, 2003. https://doi.org/10.1007/978-1-4612-0031-4, URL https://link.springer.com/book/10.1007/978-1-4612-0031-4.

[30] Markley, F. L., and Crassidis, J. L., *Fundamentals of Spacecraft Attitude Determination and Control*, Space Technology Library, Vol. 36, Springer, 2014. https://doi.org/10.1007/978-1-4614-6435-8, URL https://link.springer.com/book/10.1007/978-1-4614-6435-8.

[31] Karlgaard, C., and Schaub, H., "Nonsingular Attitude Filtering Using Modified Rodrigues Parameters," *The Journal of the Astronautical Sciences*, Vol. 57, 2009. https://doi.org/10.1007/BF03321529.

[32] Rahimi, A., Kumar, K. D., and Alighanbari, H., "Failure Prognosis for Satellite Reaction Wheels Using Kalman Filter and Particle Filter," *Journal of Guidance, Control, and Dynamics*, Vol. 43, No. 3, 2020, pp. 585–588. https://doi.org/10.2514/1.G004616, URL https://doi.org/10.2514/1.G004616.

[33] Sirajul Islam, M., and Rahimi, A., "Fault Prognosis of Satellite Reaction Wheels Using A Two-Step LSTM Network," *2021 IEEE International Conference on Prognostics and Health Management (ICPHM)*, 2021, pp. 1–7. https://doi.org/10.1109/ICPHM51084.2021.9486655.

[34] Talebi, H., and Patel, R., "An intelligent Fault Detection and Recovery scheme for reaction wheel actuator of satellite attitude control systems," *2006 IEEE Conference on Computer Aided Control System Design, 2006 IEEE International Conference on Control Applications, 2006 IEEE International Symposium on Intelligent Control*, 2006, pp. 3282–3287. https://doi.org/10.1109/CACSD-CCA-ISIC.2006.4777164.

[35] Macenski, S., Foote, T., Gerkey, B., Lalancette, C., and Woodall, W., "Robot Operating System 2: Design, architecture, and uses in the wild," *Science Robotics*, Vol. 7, No. 66, 2022, p. eabm6074. https://doi.org/10.1126/scirobotics.abm6074, URL https://www.science.org/doi/abs/10.1126/scirobotics.abm6074.