(Preprint) AAS 25-682

RBFNN-BASED ADAPTIVE ATTITUDE CONTROLLER WITH REACTION WHEEL HEALTH ESTIMATION

Morokot Sakal[*], George Nehma[*], Camilo Riano-Rios[†], and Madhur Tiwari[†]

ABSTRACT

Precise attitude control system is critical for the success of space missions. Reaction wheels, which are the most commonly used actuators in spacecraft, degrade over time due to the harsh conditions in space. We present an adaptive controller that simultaneously estimates the time-varying RW health degradation level while maintaining attitude tracking. We lift the assumption of constant or slowly varying health degradation made in our earlier work by approximating the RWs nonlinear health degradation using Radial Basis Function Neural Networks (RBFNN). We propose using a Concurrent Learning (CL)-based adaptation law to ensure the convergence of the RBFNN weights and biases, and thereby the health degradation levels, to their true values while maintaining attitude tracking. Simulation results demonstrate that the learned health parameters also provide the controller with the ability to reduce the control effort allocated on the degraded wheel, helping prevent its winding temperature from rising.

INTRODUCTION

Spacecraft attitude control systems play an important role in ensuring the success of space missions. Reaction wheels (RWs) are often chosen as the actuators due to their ability to precisely apply control torques to the spacecraft. However, RWs are susceptible to failure due to their moving parts and the harsh conditions of space environment.[1–5] A range of solutions to address uncertainties and faults in spacecraft have been explored, including the sliding-mode control,[6] observer-based,[7] adaptive controllers,[8–10] and neural network-based methods.[11] The advantage of adaptive controllers is their ability to compensate for uncertainties in the system dynamics, whilst maintaining control over the system in a computationally inexpensive manner. Neural networks are capable of learning uncertain parameters, but their increased size, complexity, and learning based on iterative algorithms often make it difficult to integrate them into stability analyses and implement on real-time spacecraft hardware. As such, methods utilizing Neural Networks should account for this limitation in their development.

In our prior work,[10] we designed an adaptive controller that can simultaneously estimate and compensate for the degradation level of the RWs while maintaining accurate attitude tracking when there are RW failures or degradation. Under the assumption of a constant or slowly varying health factor, our method involved a Lyapunov-based adaptive control system with an integral concurrent

[*]Ph.D. Student, Aerospace, Physics and Space Sciences Department, Florida Institute of Technology
[†]Assistant Professor, Aerospace, Physics and Space Sciences Department, Florida Institute of Technology

learning (ICL)-based update law that ensures convergence of the estimated RW health once a sufficient Finite Excitation (FE) condition is met. Despite promising results, the main limitation was the constant or slowly varying health parameters assumption. This assumption limited the controller's uncertain parameter estimation performance in scenarios with a time-varying health change. To mitigate the effect of this assumption, a regular reset of the ICL term was proposed to allow the controller collect new data periodically and provide a more accurate fault estimation.

This work addresses the aforementioned problem by proposing the use of Radial Basis Function Neural Networks (RBFNN) to approximate a nonlinear function that describes the RWs' health degradation level, relaxing the constant health parameters assumption whilst still maintaining stable attitude tracking. Using RBFNN, provides the controller's with the ability to learn more complex fault profiles influenced by factors such as wheel speed, spacecraft angular velocity, time, and internal winding temperature.[1]

This paper is outlined as follows. The first section derives the spacecraft dynamics and RBFNN-based fault model. The next two sections present the design of the proposed RBFNN-based controller and the stability analysis. The following section discusses the simulation results. Finally, the conclusion and future work are described.

SPACECRAFT ATTITUDE DYNAMICS

Given the attitude equations of motions (EoMs) for a Spacecraft with N RWs

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

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

where $\boldsymbol{\omega} \in \mathbb{R}^3$ is the spacecraft inertial angular velocity expressed in the body frame, $\boldsymbol{\sigma} \in \mathbb{R}^{\mathbf{3}}$ are the Modified Rodrigues Parameters (MRP) representing the attitude of the spacecraft with respect to the inertial frame, $\boldsymbol{\Omega} \in \mathbb{R}^N$ contains the angular velocity of the N reaction wheels, $J \in \mathbb{R}^{3\times3}$ is the total inertia matrix, $J_{RW} \in \mathbb{R}_{>0}$ is the inertia of the flywheels about their spin axis, $\Phi = \text{diag}\{\phi_1,\ \phi_2,\ \cdots,\ \phi_N\} \in \mathbb{R}^{N\times N}$ is the uncertain RW health matrix, $G \in \mathbb{R}^{3\times N}$ is the RW array configuration matrix.

From the Equation (1), we rewrite the term $-J_{RW}G\Phi\dot{\boldsymbol{\Omega}}$ as

$$-J_{RW}G\Phi\dot{\boldsymbol{\Omega}} = -G\ \text{diag}\left(\boldsymbol{\tau_{RW}}\right)\boldsymbol{\Phi_{vec}}, \tag{3}$$

where $\boldsymbol{\tau_{RW}} = J_{RW}\dot{\boldsymbol{\Omega}}$, and $\boldsymbol{\Phi_{vec}} = [\phi_1, \phi_2, \cdots, \phi_N] \in \mathbb{R}^N$ is the vector of reaction wheel "health" parameters.

Instead of assuming that $\boldsymbol{\Phi_{vec}}$ is constant, we apply the universal function approximation theorem of Neural Networks[12] to approximate $\boldsymbol{\Phi_{vec}}$ with RBFNNs. We treat each individual wheel-health factor ϕ_i as an unknown nonlinear function and estimate it with its own RBFNN due to the fact that approximating the nonlinear health profiles of all wheels with one RBFNN model exponentially increases the required number of RBF's to span the entire feature space. As each health profile is independent of one another and only dependent on the characteristics of the corresponding wheel, creating one RBFNN for each RW instead of coupling reduces the required number of RBF's

from M^m RBF's to $M \cdot m$. Where $M \in \mathbb{Z}_{>0}$ is the chosen number of RBFs to span each input feature's range, and $m \in \mathbb{Z}_{>0}$ is the number of features that affect a RW's health.

The choice of using an RBFNN over other Neural Network (NN) structures, such as a Multilayer Perceptron (MLP), is due to a key component in CL-based adaptation law design that requires the uncertain terms to be expressed in a form that is linear in the uncertain parameters. The structure of RBFNNs is suitable for this purpose when weights and biases are their only tunable parameters. The use of a MLP would result in some tunable parameters confined within nonlinear activation functions, complicating the derivation of formal convergence guarantees of the uncertain NN parameters.

Each RW-health factor ϕ_i can then be expressed as

$$\phi_i \;=\; W_i^T\, S_i(\bar{x}_i) + \varepsilon_i, \qquad i = 1, \ldots, N \tag{4}$$

where ε_i is the function approximation error (unknown, bounded and in general smaller as the size of the network increases), $\bar{\boldsymbol{x}}_{\boldsymbol{i}} = [\boldsymbol{x_i}^T, 1]^T$, $\boldsymbol{x_i} \in \mathbb{R}^d$ is the vector of inputs (features) to the NN, $S_i(\bar{x}_i) \in \mathbb{R}^{M_i+1}$ is the vector of nonlinear activation functions (including the appended unity to account for the bias terms), $M_i \in \mathbb{Z}_{>0}$ is the number of neurons assigned to wheel i, and $W_i \in \mathbb{R}^{M_i+1}$ contains weights and biases of the i-th RBFNN.

For our purposes, the term $W_i^T S_i(\bar{x}_i)$ represents an RBFNN assigned to RW i with M_i neurons, where the vector W_i contains the NN's weights and biases between the hidden and output layer. The vector of RBFs activation functions $S_i(\bar{x}_i)$ with its j^{th} entry defined as

$$S_{i,j}(\bar{\boldsymbol{x}}) = \exp\left(-\frac{(\bar{\boldsymbol{x}}_{\boldsymbol{i}} - \mu_{i,j})^T\,(\bar{\boldsymbol{x}}_{\boldsymbol{i}} - \mu_{i,j})}{\eta_{i,j}^2}\right), \tag{5}$$

where $j \in \mathbb{Z}_{>0}$ denotes the j^{th} neuron index for wheel i, $\mu_{i,j} \in \mathbb{R}^{d+1}$ is the "position" of the hidden node or neuron's RBF in the feature/input space, and $\eta_{i,j} \in \mathbb{R}_{>0}$ is the "width" of the j^{th} RBF.

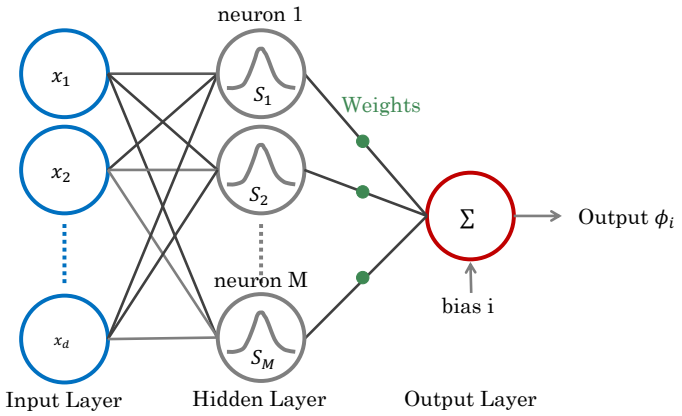


Figure 1: RBFNN with Gaussian activation centered at $\mu_{i,j}$ and width $\eta_{i,j}$.

For the i^{th} wheel, the term $W_i^T S_i$ is a scalar, and can therefore be transposed as

$$W_i^T S_i(\bar{x}_i) \;=\; S_i^T(\bar{x}_i)\, W_i, \qquad i = 1, \ldots, N, \tag{6}$$

Stacking all outputs into a vector, we obtain $\mathbf{\Phi}_{\text{vec}}$ as

$$\mathbf{\Phi}_{\text{vec}} = \begin{bmatrix} S_1^T W_1 \\ \vdots \\ S_N^T W_N \end{bmatrix} + \boldsymbol{\epsilon} = \begin{bmatrix} S_1^T & 0 & \cdots & 0 \\ 0 & S_2^T & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & S_N^T \end{bmatrix} \begin{bmatrix} W_1 \\ W_2 \\ \vdots \\ W_N \end{bmatrix} + \boldsymbol{\epsilon} \tag{7}$$

where the i^{th} contains the transposed activation vector $S_i^T = [\, S_{i,1}(\bar{x}_i) \;\; \ldots \;\; S_{i,M_i}(\bar{x}_i) \;\; 1 \,]$ and zeros elsewhere.

Equation (7) can be further expressed with the simplified short-hand notation

$$\mathbf{\Phi}_{\text{vec}} = \widetilde{\text{blkdiag}}\big(S_1^T, \ldots, S_N^T\big)\; W + \boldsymbol{\epsilon}, \tag{8}$$

where $W = \big[\, \mathbf{W_1}^T \; \mathbf{W_2}^T \;\; \ldots \;\; \mathbf{W_N}^T \big]^T$ and $\boldsymbol{\epsilon} = [\varepsilon_1, \ldots, \varepsilon_N]^T$

By plugging-in Equation (8) into Equation (3) yields

$$-J_{RW}\, G\, \Phi\, \dot{\Omega} = \; -G\, \text{diag}\big(\boldsymbol{\tau}_{RW}\big) \left(\widetilde{\text{blkdiag}}\big(S_1^T, \ldots, S_N^T\big)\, \mathbf{W} + \boldsymbol{\epsilon} \right), \tag{9}$$

where $\boldsymbol{\tau}_{RW} \in \mathbb{R}^N$.

We define the regressor matrix $\Psi \in \mathbb{R}^{3\times(M+1)N}$ as

$$\Psi = -G\, \text{diag}\big(\boldsymbol{\tau}_{RW}\big)\, \widetilde{\text{blkdiag}}\big(S_1^T, \ldots, S_N^T\big), \tag{10}$$

Equation (9) can be rewritten as

$$-J_{RW} G \Phi \dot{\mathbf{\Omega}} = \Psi\; \mathbf{W} - G\text{diag}(\boldsymbol{\tau_{RW}})\boldsymbol{\epsilon} = \Psi\; \mathbf{W} + G\text{diag}(\boldsymbol{\tau_{RW}})\boldsymbol{\epsilon} \tag{11}$$

where the term involving $\boldsymbol{\epsilon}$ is rewritten with the sign flipped, so the residual remains positive.

CONTROLLER DESIGN

Control Objective

The objective is to design an adaptive controller that guarantees attitude tracking and convergence of estimates of the RBFNN's weights/biases $\hat{\mathbf{W}}^T$ to their true values $\mathbf{W}^T$. As opposed to directly learning the RWs health parameters $\phi_1, \phi_2, \cdots, \phi_N$, we propose a Concurrent Learning (CL)-based adaptation law to ensure convergence of the RBFNN's weights and biases.

Using Equation (11), the dynamics in Equation (1) can be expressed as

$$J\dot{\boldsymbol{\omega}} = -\boldsymbol{\omega} \times (J\boldsymbol{\omega} + J_{RW} G \mathbf{\Omega}) + \Psi\; \mathbf{W} + G\text{diag}(\boldsymbol{\tau_{RW}})\boldsymbol{\epsilon} \tag{12}$$

$$\dot{\boldsymbol{\sigma}} = \frac{1}{4}\left[\big(1 - \boldsymbol{\sigma^T \sigma}\big)\, I_3 + 2\sigma^{\times} + 2\boldsymbol{\sigma}\boldsymbol{\sigma^T} \right] \boldsymbol{\omega} \tag{13}$$

Let us define the weights/biases estimation error as $\tilde{\mathbf{W}} = \mathbf{W} - \hat{\mathbf{W}}$, and the EoMs can be rewritten as

$$J\dot{\boldsymbol{\omega}} = -\boldsymbol{\omega} \times (J\boldsymbol{\omega} + J_{RW} G\boldsymbol{\Omega}) + \Psi\, \tilde{\mathbf{W}} + \Psi\, \hat{\mathbf{W}} + G\mathrm{diag}(\boldsymbol{\tau_{RW}})\boldsymbol{\epsilon} \tag{14}$$

$$\dot{\boldsymbol{\sigma}} = \frac{1}{4}\left[\left(1 - \boldsymbol{\sigma}^{\boldsymbol{T}}\boldsymbol{\sigma}\right) I_3 + 2\sigma^{\times} + 2\boldsymbol{\sigma}\boldsymbol{\sigma}^{\boldsymbol{T}}\right]\boldsymbol{\omega} \tag{15}$$

Assumption 1. The spacecraft carries an attitude-determination system capable of delivering the angular velocity $\boldsymbol{\omega}$ and attitude $\boldsymbol{\sigma}$ measurements/estimates in real-time.

Let us define the attitude error MRP $\boldsymbol{\sigma}_e = \boldsymbol{\sigma} - \boldsymbol{\sigma}_d \in \mathbb{R}^3$ and the relative angular velocity $\tilde{\boldsymbol{\omega}} = \boldsymbol{\omega} - \tilde{R}\,\boldsymbol{\omega}_d \in \mathbb{R}^3$ where $\tilde{R}$ represents the rotation matrix between body and the desired frames and is defined as

$$\tilde{R} = I_3 + \frac{8\boldsymbol{\sigma_e}^{\times}\boldsymbol{\sigma_e}^{\times} - 4(1 - \boldsymbol{\sigma_e}^{T}\boldsymbol{\sigma_e})\,\boldsymbol{\sigma_e}^{\times}}{(1 + \boldsymbol{\sigma_e}^{T}\boldsymbol{\sigma_e})^2}. \tag{16}$$

The attitude error and the relative angular velocity obeys the following kinematics equation[13]

$$\dot{\boldsymbol{\sigma}}_e = \tfrac{1}{4} B\,\tilde{\boldsymbol{\omega}}, \tag{17}$$

with $B = \left(1 - \boldsymbol{\sigma_e}^{T}\boldsymbol{\sigma_e}\right) I_3 + 2\boldsymbol{\sigma_e}^{\times} + 2\boldsymbol{\sigma_e}\boldsymbol{\sigma_e}^{T}$.

The attitude control objective is achieved when

$$\|\boldsymbol{\sigma_e}\| \to 0, \;\text{ and }\; \|\tilde{\boldsymbol{\omega}}\| \to 0 \;\Rightarrow\; \|\dot{\boldsymbol{\sigma}}_{\boldsymbol{e}}\| \to 0. \tag{18}$$

which implies that

$$\tilde{R} \to I_3, \quad \text{as } t \to \infty. \tag{19}$$

Control Development

We introduce the auxiliary error $\boldsymbol{r} \in \mathbb{R}^3$ as

$$\boldsymbol{r} = \dot{\boldsymbol{\sigma}}_{\boldsymbol{e}} + \alpha\boldsymbol{\sigma_e}, \tag{20}$$

where $\alpha \in \mathbb{R}^{3\times 3}$ is a symmetric, positive definite control gain matrix. The time derivative of the Equation (20) can be obtained as,

$$\dot{\boldsymbol{r}} = \frac{1}{4}\dot{B}\tilde{\boldsymbol{\omega}} + \frac{1}{4}B\left(\dot{\boldsymbol{\omega}} - \tilde{R}\dot{\boldsymbol{\omega}}_{\boldsymbol{d}} - \dot{\tilde{R}}\boldsymbol{\omega_d}\right) + \alpha\dot{\boldsymbol{\sigma}}_{\boldsymbol{e}}, \tag{21}$$

with $\dot{B} = \left[-2\boldsymbol{\sigma}_{\boldsymbol{e}}^{\boldsymbol{T}}\dot{\boldsymbol{\sigma}}_{\boldsymbol{e}} I_3 + 2\dot{\sigma}_e^{\times} + 4\left(\dot{\boldsymbol{\sigma}}_{\boldsymbol{e}}\boldsymbol{\sigma}_{\boldsymbol{e}}^{\boldsymbol{T}}\right)\right]$.[13]

Using the dynamics in Eq. (15), and the fact that $\dot{\tilde{R}} = -\tilde{\omega}^{\times}\tilde{R}$, we obtain

$$\dot{r} = \frac{1}{4}\dot{B}\,\tilde{\omega} + \frac{1}{4}B\big[J^{-1}(-\omega \times (J\omega + J_{RW}G\Omega) + \Psi\,\tilde{\mathbf{W}} + \Psi\,\hat{\mathbf{W}} + G\mathrm{diag}(\tau_{RW})\,\boldsymbol{\epsilon} - \tilde{R}\dot{\omega}_d + \tilde{\omega}^{\times}\tilde{R}\omega_d\big] + \alpha\dot{\sigma}_e. \tag{22}$$

Since the term $\Psi\,\mathbf{W}$ contains the control input $\boldsymbol{\tau_{RW}}$, i.e. RW torques, we design the the auxiliary control signal $\boldsymbol{u_d} \in \mathbb{R}^3$ as

$$\boldsymbol{u_d} = \boldsymbol{\omega} \times (J\boldsymbol{\omega} + J_{RW} G\boldsymbol{\Omega}) + J\tilde{R}\dot{\boldsymbol{\omega}}_{\boldsymbol{d}} - J\tilde{\omega}^{\times}\tilde{R}\boldsymbol{\omega_d} + 4JB^{-1}\left[-\tfrac{1}{4}\dot{B}\tilde{\boldsymbol{\omega}} - \alpha\dot{\boldsymbol{\sigma}}_{\boldsymbol{e}} - K\boldsymbol{r} - \beta\boldsymbol{\sigma_e}\right], \quad (23)$$

where $K \in \mathbb{R}^{3\times3}$ and $\beta \in \mathbb{R}_{>0}$ are constant, symmetric, positive definite control gains.

Based on the Equation (22) and (23), we propose an adaptation law with a gradient-based term and a CL term that collects input-output data.[14] The adaptation law for the RBFNN weight and bias estimates is proposed as

$$\dot{\hat{\mathbf{W}}} \;=\; \mathrm{proj}\Big\{\tfrac{1}{4}\Gamma\,\Psi^T(J^{-1})^T B^T\boldsymbol{r} + \Gamma K_{CL}\sum_{i=1}^{N_s}\Psi_i^T\left(J\dot{\boldsymbol{\omega}}_i + \boldsymbol{\omega}_i \times \left(J\boldsymbol{\omega}_i + J_{RW}G\boldsymbol{\Omega}_i\right) - \Psi_i\,\hat{\mathbf{W}}\right)\Big\} \quad (24)$$

where $\Gamma,\ K_{CL} \in \mathbb{R}^{(M+1)N\times(M+1)N}$ are symmetric, positive-definite adaptation gain matrices. $\mathrm{proj}\{\cdot\}$ is a projection algorithm used to confine the $\hat{\mathbf{W}}$ to a user-defined bounds.[15]

The commanded torque in the body frame $\boldsymbol{u_d}$ is mapped to the torque produced by the reaction wheels through the following relationship

$$G\,\hat{\Phi}\,\boldsymbol{\tau}_{RW} \;=\; \boldsymbol{u}_d, \quad (25)$$

where $\hat{\Phi} = \mathrm{diag}(\hat{\boldsymbol{\Phi}}_{vec})$ is the estimate of the actual wheel-health factor Φ.

We can obtain the actual RW control input $\boldsymbol{\tau}_{RW}$ as

$$\boldsymbol{\tau}_{RW} = \left(G\hat{\Phi}\right)^{\dagger}\boldsymbol{u}_d \quad (26)$$

where $(\cdot)^{\dagger}$ is the Moore–Penrose pseudoinverse.

Assumption 2. There exists a finite time $T > 0$ such that

$$\lambda_{\min}\Big(\sum_{i=1}^{N_s}\Psi_i^T\Psi_i\Big) \;\geq\; \bar{\lambda}, \quad (27)$$

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

STABILITY ANALYSIS

Let us define the composite state vector $\boldsymbol{z} = [\boldsymbol{r}^T, \boldsymbol{\sigma}_e^T, \tilde{W}]^T \in \mathbb{R}^{\,6+(M+1)N}$ and divide the analysis into two parts: the first part shows the stability of the closed-loop dynamics at time $t < T$, where

the finite excitation condition is not yet satisfied; and the second part considers the stability analysis for time $t \geq T$.

Consider the following Lyapunov candidate function

$$V(\boldsymbol{z}) = \frac{1}{2}\,\boldsymbol{r}^T\boldsymbol{r} + \frac{\beta}{2}\,\boldsymbol{\sigma}_e^T\boldsymbol{\sigma}_e + \frac{1}{2}\,\tilde{W}^T\,\Gamma^{-1}\,\tilde{W} \tag{28}$$

There exist two positive constants $\underline{\kappa}, \bar{\kappa}$ such that,

$$\underline{\kappa}\|\boldsymbol{z}\|^2 \leq V(\boldsymbol{z}) \leq \overline{\kappa}\|\boldsymbol{z}\|^2 \tag{29}$$

Using the time-derivative of the modified state vector $\dot{\boldsymbol{r}}$, and the auxiliary control law $\boldsymbol{u_d}$, we get

$$\dot{V}(\boldsymbol{z}) = \boldsymbol{r}^T\Big(\tfrac{1}{4}\,B\,J^{-1}\,(\Psi\,\tilde{W} + G\mathrm{diag}(\boldsymbol{\tau_{RW}})\boldsymbol{\epsilon}) - K\,\boldsymbol{r}\Big) - \beta\,\boldsymbol{\sigma}_e^T\,\alpha\,\boldsymbol{\sigma}_e - \tilde{W}^T\,\Gamma^{-1}\,\dot{\tilde{W}}. \tag{30}$$

Using Equations (13) and (24), the adaptation law can be expressed in its equivalent analytical form

$$\dot{\tilde{\boldsymbol{W}}} = \mathrm{proj}\Big\{\tfrac{1}{4}\,\Gamma\,\Psi^T(J^{-1})^T B^T\,\boldsymbol{r} + \Gamma\,K_{CL}\sum_{i=1}^{N_s}\Psi_i^T\,\Psi_i\,\tilde{\boldsymbol{W}} + \Psi_i^T G\mathrm{diag}(\boldsymbol{\tau_{RW,i}})\boldsymbol{\epsilon}_i\Big\}. \tag{31}$$

Plugging it into Equation (30), we obtain the expression for $\dot{V}$ as

$$\begin{aligned}\dot{V}(\boldsymbol{z}) = &-\boldsymbol{r}^T K\,\boldsymbol{r} - \beta\,\boldsymbol{\sigma}_e^T\alpha\,\boldsymbol{\sigma}_e - \tilde{\boldsymbol{W}}^T K_{CL}\sum_{i=1}^{N_s}\Psi_i^T\Psi_i\tilde{\boldsymbol{W}} \\ &- \tilde{\boldsymbol{W}}^{\boldsymbol{T}} K_{CL}\sum_{i=1}^{N_s}\Psi_i^T G\mathrm{diag}(\boldsymbol{\tau_{RW,i}})\boldsymbol{\epsilon}_i + \boldsymbol{r}^T\big(\tfrac{1}{4}\,B\,J^{-1}\,G\mathrm{diag}(\boldsymbol{\tau_{RW}})\boldsymbol{\epsilon}\big).\end{aligned} \tag{32}$$

Part I: Pre-finite-excitation phase

Theorem 1. If the FE condition is not yet satisfied (i.e., $0 \leq t < T$), the tracking-error state $\boldsymbol{y} = [\boldsymbol{r}^T,\, \boldsymbol{\sigma_e}]^T \in \mathbb{R}^6$ is ultimately bounded such that

$$\|\boldsymbol{y}(t)\| \;\leq\; \sqrt{\frac{\overline{\gamma}}{\underline{\gamma}}}\,\|\boldsymbol{y}(0)\|\,\exp\Big(-\frac{\eta}{2\,\overline{\gamma}}\,t\Big) \;+\; \sqrt{\frac{\overline{b}-\underline{b}}{\underline{\gamma}}} \;+\; \sqrt{\frac{c\,\overline{\gamma}}{\eta\,\underline{\gamma}}} \quad 0 \leq t < T. \tag{33}$$

Proof. Consider the following tracking-error states $\boldsymbol{y} = [\boldsymbol{r}^T,\, \boldsymbol{\sigma_e}]^T$. There exists bounding constants $\underline{\gamma}, \overline{\gamma}, \underline{b},\ \overline{b} \in \mathbb{R}_{>0}$ such that

$$\underline{\gamma}\,\|\boldsymbol{y}\|^2 + \underline{b} \;\leq\; V(\boldsymbol{y}) \;\leq\; \overline{\gamma}\,\|\boldsymbol{y}\|^2 + \overline{b}. \tag{34}$$

Before time $t = T$, the Equation (32) can be upper bounded

$$\dot{V}(\boldsymbol{y}) \;\leq\; -\lambda_{\min}\{K\}\|\boldsymbol{r}\|^2 - \beta\lambda_{\min}\{\alpha\}\,\|\boldsymbol{\sigma}_e\|^2 + \|\boldsymbol{r}\|\,\bar{\epsilon}_1 + \bar{\epsilon}_2. \tag{35}$$

where $\bar{\epsilon}_1,\ \bar{\epsilon}_2 \in \mathbb{R}_{>0}$ are constant bounds for therms involving the RBFNN approximation error such that

$$\left\|\tfrac{1}{4}B\,J^{-1}G\mathrm{diag}(\boldsymbol{\tau_{RW}})\boldsymbol{\epsilon}\right\| \;\leq\; \bar{\epsilon}_1, \quad , \left\|\tilde{\boldsymbol{W}}^T K_{CL}\sum_{i=1}^{N_s}\Psi_i^T G\mathrm{diag}(\boldsymbol{\tau_{RW,i}})\boldsymbol{\epsilon}_i\right\| \leq \bar{\epsilon}_2, \tag{36}$$

where the following facts were used: $\hat{\boldsymbol{W}}$ is bounded due to the projection algorithm in Equation (31), the applied RW torques $\boldsymbol{\tau}_{RW}$ are bounded due to the spacecraft+RWs being a closed system exchanging angular momentum, and RBF activation functions in Ψ are bounded by definition.

Applying Young's inequality as $\|\boldsymbol{r}\|\,\bar{\epsilon}_1 \leq \frac{1}{2}\|\boldsymbol{r}\|^2 + \frac{1}{2}\bar{\epsilon}_1^2$, yields the expression for $\dot{V}(\boldsymbol{y})$

$$\begin{aligned}\dot{V}(\boldsymbol{y}) \;&\leq\; -\lambda_{\min}\{K\}\|\boldsymbol{r}\|^2 \;-\; \beta\,\lambda_{\min}\{\alpha\}\,\|\boldsymbol{\sigma}_e\|^2 \;+\; \tfrac{1}{2}\,\|\boldsymbol{r}\|^2 \;+\; \tfrac{1}{2}\,\bar{\epsilon}_1^2 + \bar{\epsilon}_2 &(37)\\ &\leq\; -\big(\lambda_{\min}\{K\} - \tfrac{1}{2}\big)\|\boldsymbol{r}\|^2 \;-\; \beta\,\lambda_{\min}\{\alpha\}\,\|\boldsymbol{\sigma}_e\|^2 + c &(38)\end{aligned}$$

with $c = \;\frac{1}{2}\,\bar{\epsilon}_1^2 + \bar{\epsilon}_2$.

Here, let us define the following constants to simplify the expression

$$c_1 = \lambda_{\min}\{K\} - \tfrac{1}{2} > 0,\ c_2 = \beta\,\lambda_{\min}\{\alpha\} > 0\ . \tag{39}$$

Then, $\dot{V}(\boldsymbol{y})$ can be expressed as,

$$\dot{V}(\boldsymbol{y}) \leq -\eta\,\|\boldsymbol{y}\|^2 + c, \tag{40}$$

with $\eta = \min\{c_1, c_2\} > 0$. Using the upper bound of the Equation (34), we replace $\|\boldsymbol{y}\|^2$ with $(V(\boldsymbol{y}) - \bar{b})/\overline{\gamma}$, and rewrite $\dot{V}(\boldsymbol{y})$ as

$$\dot{V}(\boldsymbol{y}) \leq -\frac{\eta}{\overline{\gamma}}V(\boldsymbol{y}) + \left(\frac{\eta\bar{b}}{\overline{\gamma}} + c\right) \tag{41}$$

Using the comparison Lemma, obtain the following

$$V(\boldsymbol{y}(t)) \leq \left(V(\boldsymbol{y}(0)) - \bar{b} - \frac{c\overline{\gamma}}{\eta}\right)\exp\Big(-\frac{\eta}{\overline{\gamma}}t\Big) + \bar{b} + \frac{c\overline{\gamma}}{\eta} \quad 0 \leq t < T. \tag{42}$$

Using the lower quadratic bound in Equation (34), yields the upper bound for $\|y(t)\|$ in Equation (33), where the tracking error vector $\boldsymbol{y}$ decays with the rate of $\eta/(2\overline{\gamma})$ until it reaches the bound $\sqrt{(\bar{b} - \underline{b})/\underline{\gamma}} + \sqrt{c\,\overline{\gamma}/(\eta\,\underline{\gamma})}$.

Part II: Post-finite-excitation phase

Theorem 2. Once the FE condition is satisfied, the composite error state $\boldsymbol{z}(t)$ converges exponentially until it reaches a bound whose radius depends on the RBFNN approximation error. $\boldsymbol{z}(t)$ is expressed as

$$\|\boldsymbol{z}(t)\| \;\le\; \sqrt{\frac{\overline{\kappa}}{\underline{\kappa}}}\,\|\boldsymbol{z}(T)\|\,\exp\!\Big(-\frac{\zeta}{2\,\overline{\kappa}}\,(t-T)\Big) \;+\; \sqrt{\frac{c_3\,\overline{\kappa}}{\zeta\,\underline{\kappa}}} \quad t \ge T. \tag{43}$$

where $\zeta = \min\{\lambda_{\min}(K)-\frac{1}{2},\, \beta\lambda_{\min}(\alpha),\, \lambda_{\min}(K_{CL}\sum_{i=1}^{N_s}\Psi_i^T\Psi_i)-\frac{1}{2}\}$, $\Big\|K_{CL}\sum_{i=1}^{N_s}\Psi_i^T G\mathrm{diag}(\boldsymbol{\tau_{RW,i}})\boldsymbol{\epsilon}_i\Big\| \le \bar{\epsilon}_3$, and $c_3 = \frac{1}{2}(\bar{\epsilon}_1^2+\bar{\epsilon}_3^2)$.

Proof. With the finite excitation condition satisfied ($t \ge T$), the matrix $\sum_{i=1}^{N_s}\Psi_i^T\Psi_i$ becomes positive definite. The Equation (32) can be rewritten as

$$\begin{aligned}\dot{V}(\boldsymbol{z}) \;\le\; -\big(\lambda_{\min}(K)-\tfrac{1}{2}\big)\,\|\boldsymbol{r}\|^2 \;-\; \beta\,\lambda_{\min}(\alpha)\,\|\boldsymbol{\sigma}_e\|^2 \\ -\;\left(\lambda_{\min}\Big(K_{CL}\sum_{i=1}^{N_s}\Psi_i^T\Psi_i\Big)-\frac{1}{2}\right)\|\tilde{W}\|^2 \;+\; \frac{1}{2}(\bar{\epsilon}_1^2+\bar{\epsilon}_3^2).\end{aligned} \tag{44}$$

The expression for $\dot{V}(\boldsymbol{z})$ can be simplified as

$$\dot{V}(\boldsymbol{z}) \le -\zeta\|\boldsymbol{z}\|^2 + c_3. \tag{45}$$

Using the upper bound from the Equation (29), we rewrite the expression in terms of $V(\boldsymbol{z})$

$$\dot{V}(\boldsymbol{z}) \le -\frac{\zeta}{\overline{\kappa}}V(\boldsymbol{z}) + c_3. \tag{46}$$

Similarly, by comparison Lemma, we obtain the expression

$$V(\boldsymbol{z}(t)) \le \left(V(\boldsymbol{z}(T) - \frac{c_3\overline{\kappa}}{\zeta}\right)\exp\!\Big(-\frac{\zeta}{\overline{\kappa}}(t-T)\Big) + \frac{c_3\overline{\kappa}}{\zeta} \qquad t \ge T. \tag{47}$$

Applying the quadratic bounds from Equation (29), for $t \ge T$, we obtain Equation (43). The finite excitation results in the exponential decay of both tracking and weight-estimation error to reach the bound $\sqrt{c_3\,\overline{\kappa}/(\zeta\,\underline{\kappa})}$

SIMULATION

The following section describes the simulation setup to validate the proposed controller's capabilities of maintaining the attitude tracking while simultaneously learning the weight-bias of the RBFNN, resulting in the estimation of the health factor of each individual wheel. The setup includes the construction of the temperature-driven health model, where the RW health factor is modeled to depend on the winding temperature of the wheel, and we assume that the controller has access to measurements of the winding temperature of each RW. The RW winding temperature is modeled

with a first-order energy-balance ODE adopted from Basilisk's motorThermal module documentation[16]

$$\dot{T}_{w,i} \;=\; -\lambda_{T,i}\,(T_{w,i} - T_{\text{env}}) \;+\; \gamma_{T,i}\,\|\tau_i\,\Omega_i\|, \tag{48}$$

where λ_T is the cooling rate, γ_T is the heating rate, τ is the torque applied to the wheel, Ω is the angular velocity of the wheel, T_{env} is the ambient temperature. The coefficients λ_T and γ_T are obtained based on the thermal properties from the datasheet of the Maxon EC 60 flat brushless motor.[17] To simulate the faulty wheel, the degraded RW is assigned a lower, less effective cooling rate and a higher heating rate (i.e, having lower efficiency). The ambient temperature T_{env} is modeled as a sine wave with a 90-minute period that mimics the orbital day-night cycling, which is standard for first-order lumped spacecraft thermal models.[18] The bias temperature $34°\text{C}$ was chosen as a common operating bus temperature of the spacecraft, and the amplitude $\pm 20°\text{C}$ was selected to create some variation of temperature for the healthy wheel that is within their safe operating range.[19]

Each wheel-health factor is modeled to depend on the winding temperature of the RW through the following expression,

$$\phi_i = \exp\!\big[-\alpha\, z_i^2\big], \qquad z_i = \frac{\max\big(T_{w,i} - T_{\text{nom}},\, 0\big)}{T_{\max} - T_{\text{nom}}}, \tag{49}$$

and is modeled based on the physics-of-failure (PoF) reliability handbooks[20,21] and related study.[1] Table 1 shows the parameters chosen to build the wheel-health factor expression. As the wheel overheats, it drives the health factor ϕ close to zero, which lowers the effective torque available from the wheel. The health-map gain α controls how much health is left when the temperature reaches the hot-case limit. The gain α is set to three so that it outputs the wheel-health factor of only 5% when the winding temperature reaches the limit. The maximum temperature is chosen as $120\ ^\circ\text{C}$ based on datasheet[17] and an experimental study.[22]

Table 1: Configuration parameters of wheel-health factor model.

Parameters	Nominal	Degraded	Units
Cooling rate, λ_T	2.6×10^{-2}	1.25×10^{-3}	s^{-1}
Heating rate, γ_T	2×10^{-2}	4×10^{-1}	$\text{K}\,(\text{N m})^{-1}$
Nominal temperature, T_{nom}	34	34	$^\circ\text{C}$
Maximum temperature, $T_{\max}$	120	120	$^\circ\text{C}$
Health-map gain, α	3	3	N/A

For the health estimation, the health of each individual wheel is approximated using a single-input RBF neural network. In this setup, the input is the normalized winding temperature. The architecture of each RBFNN is an 11-node network consisting of 10 RBFs with uniformly spaced centers (i.e., $\mu_{i,j}$) between 0.05 and 0.95 with a common width (i.e., $\eta_{i,j}$) of 0.12, and one bias. All four wheels are assumed to be initially healthy at the beginning of the simulation. To ensure that the initial health estimate is set to healthy ($\phi \approx 1$), the hidden weights are randomly initialized close to zero, and the bias is initialized close to one. With four wheels, the RBF network resulted in 44 unknown parameters to be learned by the controller. Configuration parameters are summarized in Table 2.

Table 2: RBFNN configurations.

Parameters	Value	Units
$T_{\min}, T_{max}$ (nominal)	20,60	$^{\circ}$C
$T_{min}, T_{\max}$ (degraded)	20,120	$^{\circ}$C
Basis functions	10 RBFs	–
Centers, μ_k	0.05 … 0.95 (uniform)	–
Width, η	0.12	–
Weight init.	weights [-0.1,0.1], bias [0.8,1.0]	–

In all simulation scenarios, the spacecraft follows a 12-minute alternative attitude reference between inertial and Nadir pointing throughout the test, which lasts for 40,000 seconds. Spacecraft configuration parameters are listed in Table 3, and the gains for each scenario are listed in Table 4.

Table 3: 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
J_{RW}	5.7296×10^5	$\mathrm{kg\,m^2}$
Max RW Torque	20×10^{-3}	$\mathrm{N\,m}$
Max Ω (for all RWs)	1.0472×10^3	$\mathrm{rad\,s^{-1}}$

Table 4: Gain matrices/parameters used in the test scenarios.

Gain	Scenario A	Scenario B
K	$0.1I_3$	$0.1I_3$
α	$3{\times}10^{-2}I_3$	$3{\times}10^{-2}I_3$
β	$5{\times}10^{-3}$	$5{\times}10^{-3}$
Γ	$10^{-1}I_{44}$	$100I_4$
K_{CL}	$2 \times 10^3 I_{44}$	0 (disabled)
$\bar{\lambda}$	1×10^{-9}	N/A

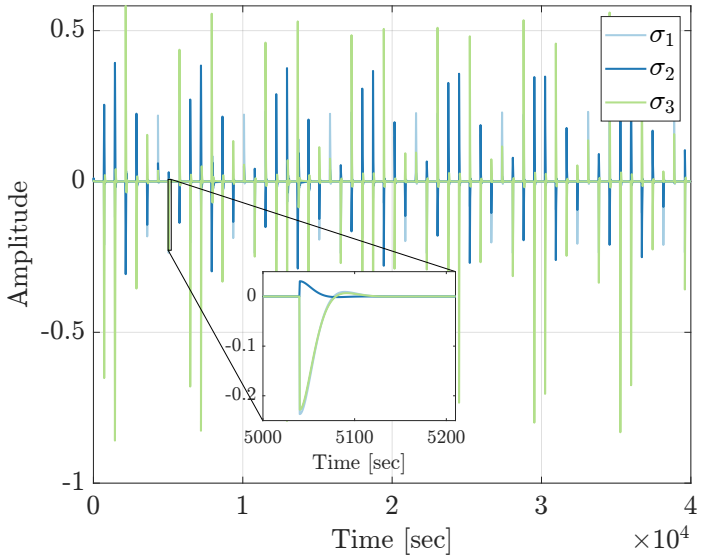
The health model setup for all scenarios is identical, in which RW4 suffers a poor thermal condition, and its winding temperature keeps rising during the operation. For Scenario A, the objective is to validate the performance of the new controller with the new health profile, which is a nonlinear, time-varying, and temperature-dependent function. This scenario also demonstrates the important CL terms in driving the convergence of the estimated health to its true value once the excitation con-

dition is satisfied. Scenario B aims to compare the advantages of the new controller by comparing it with the previously developed controller.

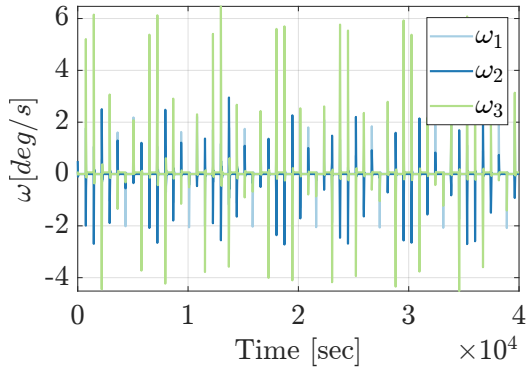
RESULTS

Scenario A: RBFNN+CL Controller

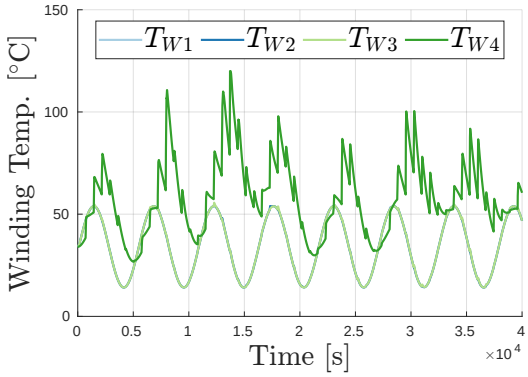
Figure 2 shows the simulation result for this case. The error MRP times series in Figure 2a, and body angular velocity times series in Figure 2b illustrate that the spacecraft achieved the attitude tracking. Because the simulation duration is long, the spikes indicate the attitude responses that occur when the spacecraft alternates between inertial hold and Nadir pointing. As seen in the zoomed-in plot, the attitude response settles to its steady-state, confirming the accurate tracking. Figure 2c plots the winding temperature of all four wheels, and it can be seen that the winding temperature of RW#4 is higher than the others due to its poor thermal management, indicating the faulty condition.



(a) Error MRP



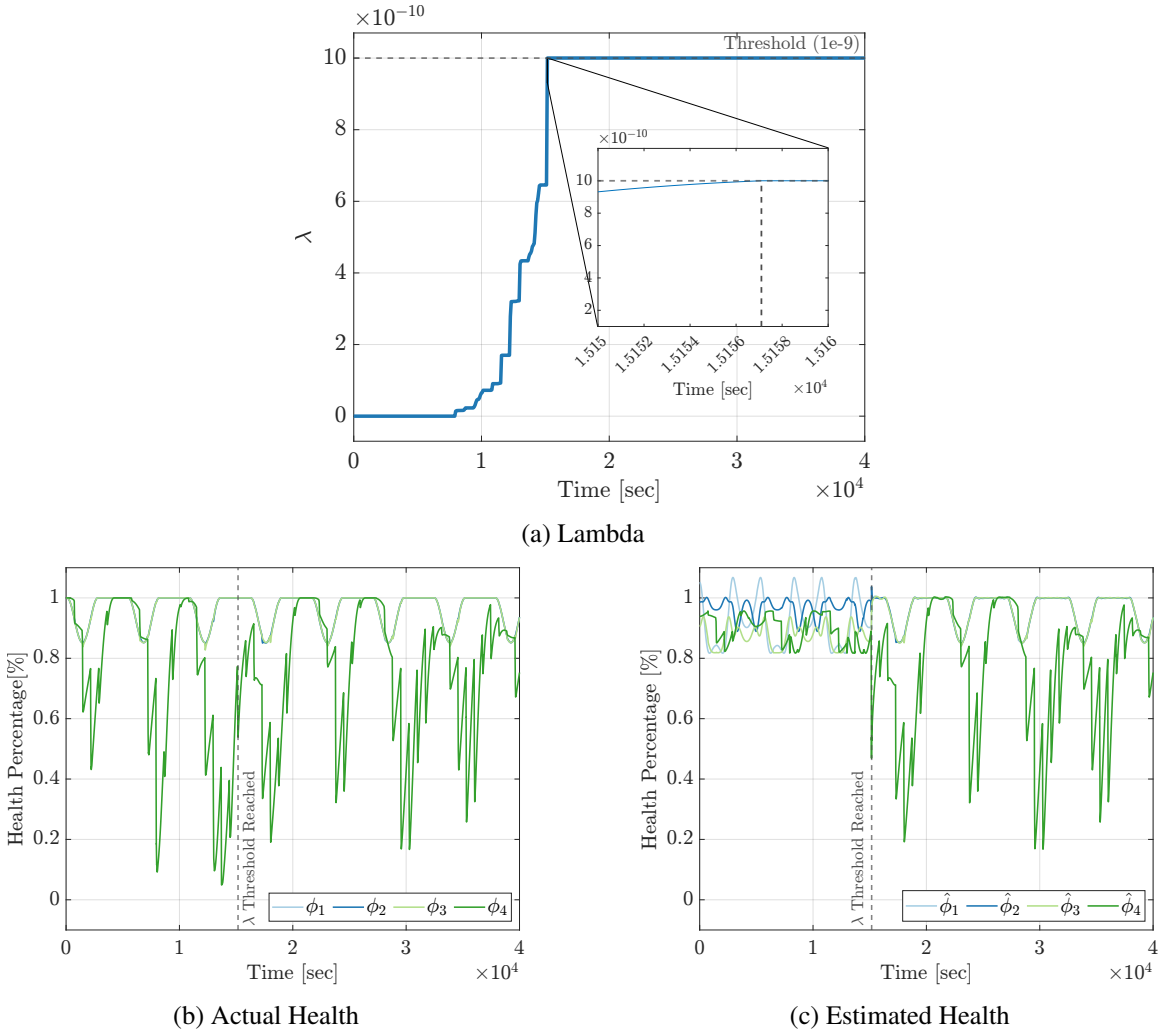
(b) Body Angular Velocity



(c) Winding Temperature

Figure 2: Scenario A: 4 RWs with CL Term Activated.

Figure 3 demonstrates the importance of the CL term for the convergence of the estimated health-factor to its true values. Figure 3a shows the plot of the lambda λ with the threshold $\bar{\lambda} = 10^{-9}$, which is user-defined, and serves as an indication of when the controller has gathered enough information about the system. Figure 3b and 3c shows the true health factors ϕ_i and their estimates $\hat{\phi}_i$, respectively. Before the $\bar{\lambda}$ threshold, the RW health estimated hovered around one and could not respond to the rapid change of the true health factors. Once λ passes the threshold $\bar{\lambda} = 10^{-9}$, the CL term is activated. As evident in Figure 3c, the estimated RW health closely tracks its true value as demonstrated in the stability analysis section.



(a) Lambda

(b) Actual Health

(c) Estimated Health

Figure 3: Reaction Wheel Health Estimation with RBF+CL Controller.

Figure 4 compares the winding temperature of RW#4 when the CL term is activated versus when it is deactivated. The health factor is modeled to depend on RW winding temperature, and the winding temperature depends on the amount of torque and angular velocity applied. Since the CL term is not activated, the controller was not able to accurately estimate the health of the RW#4. The controller continues to apply torque to RW#4, causing its temperature to rise, further degrading its health factor. With the CL term activated, the controller is able to estimate the health factor of

degraded torque, and it can effectively allocate more torque to those non-degraded wheels than to the degraded ones. Therefore, with the help of the CL term, the controller was able to prevent the RW#4 from overheating as seen in the Figure 4.

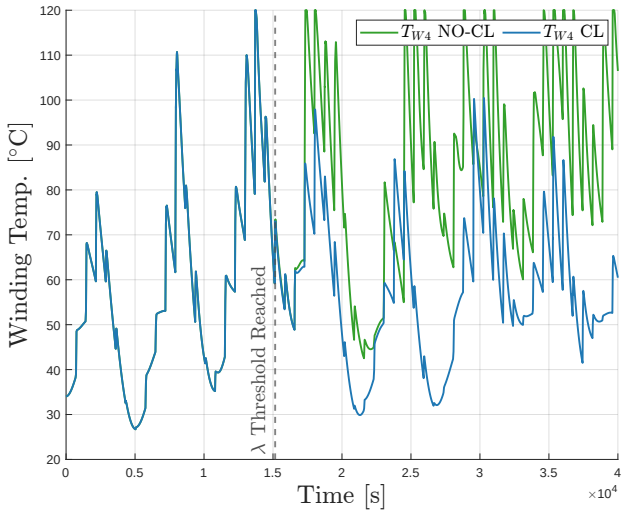
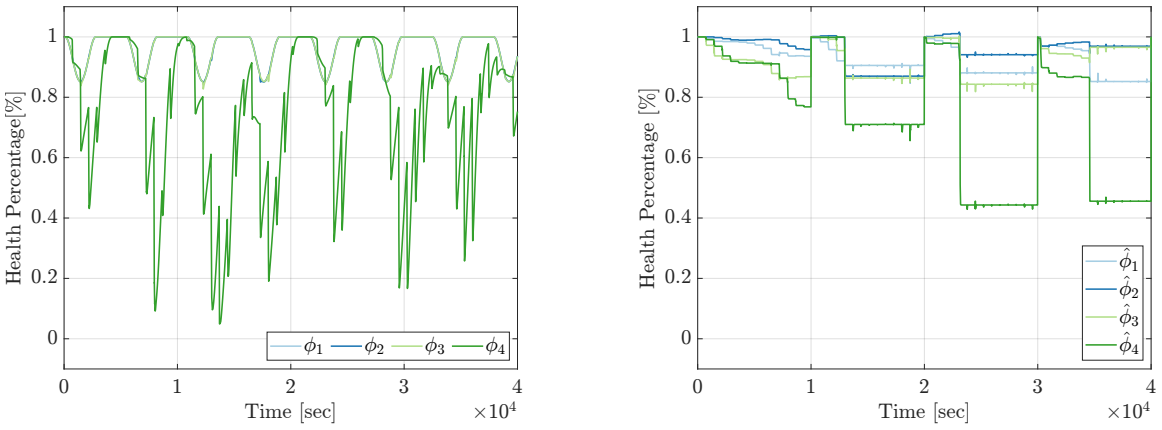


Figure 4: Comparison of RW#4's Winding Temperature with and without Activation of CL Term

Scenario B: ICL controller performance with realistic, nonlinear health model

In our previous work,[10] we developed an adaptive controller with an Integral Concurrent Learning (ICL) term that was able to learn the health parameters of the reaction wheels, given the assumption that the health parameters were constant or slowly time-varying. Although successful under this assumption, this did not allow us to properly model the complex, nonlinear health profile that realistic RWs experience. In this scenario, we apply this more realistic, nonlinear health model to our old adaptive controller to emphasize the capability of our new model to adequately learn such a nonlinear function. Figure 5 compares the true healths of the wheels during the simulation (a), and the estimated healths estimated by the ICL controller (b). It is evident that the controller, albeit activating the ICL term within periodic resets, is unable to accurately learn the health of the RWs. The constant health assumption is quite strict on this model and is hence why it struggles to capture the behavior of this health profile. It is able to recognize the fact that the wheel has degraded, but not accurately.

For this scenario, all health profiles based on coil winding temperature and guidance commands were kept the same as Scenario A to ensure a fair comparison between the two models. Not shown here, are the RWs temperatures during the simulation as they are similar to the No-CL profiles in Figure 4. The ICL controller was not able to effectively reduce the temperature, and hence wear on the degraded RW like it is able to with the CL controller.



(a) Actual Health

(b) Estimated Health

Figure 5: Reaction Wheel Health Estimation with ICL Controller.

CONCLUSION

In this paper, we proposed an adaptive controller that guarantees attitude tracking and reaction wheel health estimation by using Radial Basis Function Neural Network (RBFNN) and a Concurrent Learning (CL) adaptive control framework.

The new controller retains the Lyapunov-based stability guarantees of the original controller while being capable of identifying nonlinear, time-varying reaction-wheel (RW) health degradation that depends on winding temperature, wheel angular speed, and spacecraft angular velocity. Simulation results show that, compared with the previous approach, the new RBFNN + CL controller still provides accurate attitude tracking while being capable of learning the true health of a degraded wheel more accurately, and also helps reduce the wheel winding temperature from overheating.

Future work will focus on improving the wheel-health factor model using more accurate temperature data, validating the controller on a hardware-in-the-loop attitude testbed, and exploring the use of other, more complex NN architectures while retaining stability guarantees. This work serves as a stepping stone to use more complex NN architectures such as fully connected single-layer NNs (with weights/biases also in the input-to-hidden layer connections), and Deep NNs.

REFERENCES

[1] M. Alidadi and A. Rahimi, "Fault Diagnosis of Lubrication Decay in Reaction Wheels Using Temperature Estimation and Forecasting via Enhanced Adaptive Particle Filter," *Sensors*, Vol. 23, Jan. 2023.

[2] W. Bialke and E. Hansell, "A NEWLY DISCOVERED BRANCH OF THE FAULT TREE EXPLAINING SYSTEMIC REACTION WHEEL FAILURES AND ANOMALIES,"

[3] M. J. Dube, J. Fisher, S. Loewenthal, and P. Ward, "Recovery and Operational Best Practices for Reaction Wheel Bearings,"

[4] J. Kampmeier, R. Larsen, L. F. Migliorini, and K. A. Larson, "Reaction Wheel Performance Characterization Using the Kepler Spacecraft as a Case Study," *2018 SpaceOps Conference*, Marseille, France, American Institute of Aeronautics and Astronautics, May 2018, 10.2514/6.2018-2563.

[5] H. J. Park, S. Kim, J. Lee, N. H. Kim, and J.-H. Choi, "System-level prognostics approach for failure prediction of reaction wheel motor in satellites," *Adv. Space Res.*, Vol. 71, Mar. 2023, pp. 2691–2701.

[6] A. Li, M. Liu, X. Cao, and R. Liu, "Adaptive quantized sliding mode attitude tracking control for flexible spacecraft with input dead-zone via Takagi-Sugeno fuzzy approach," *Inf. Sci.*, Vol. 587, Mar. 2022, pp. 746–773.

[7] X. Liang, Q. Wang, C. Hu, and C. Dong, "Observer-based H∞ fault-tolerant attitude control for satellite with actuator and sensor faults," *Aerosp. Sci. Technol.*, Vol. 95, Dec. 2019, p. 105424.

[8] C. Riano-Rios, R. Bevilacqua, and W. E. Dixon, "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/10.1016/j.actaastro.2020.04.059.

[9] R. Sun, C. Riano-Rios, R. Bevilacqua, N. G. Fitz-Coy, and W. E. Dixon, "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, 10.2514/1.G005515.

[10] C. Riano-Rios, G. Nehma, and M. Tiwari, "Adaptive Controller For Simultaneous Spacecraft Attitude Tracking And Reaction Wheel Fault Detection," 2025. Version Number: 1, 10.48550/ARXIV.2504.12124.

[11] S. Mokhtari, A. Abbaspour, K. K. Yen, and A. Sargolzaei, "Neural Network-Based Active Fault-Tolerant Control Design for Unmanned Helicopter with Additive Faults," *Remote Sensing*, Vol. 13, No. 12, 2021, 10.3390/rs13122396.

[12] J. Park and I. W. Sandberg, "Universal Approximation Using Radial-Basis-Function Networks," *Neural Computation*, Vol. 3, June 1991, pp. 246–257, 10.1162/neco.1991.3.2.246.

[13] H. Schaub and J. Junkins, *Analytical Mechanics of Space Systems*. American Institute of Aeronautics and Astronautics, 4th ed., 2018.

[14] G. Chowdhary, T. Yucelen, M. Mühlegg, and E. N. Johnson, "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/10.1002/acs.2297.

[15] W. E. Dixon, A. Behal, D. M. Dawson, and S. P. Nagarkatti, *Nonlinear control of engineering systems: a Lyapunov-based approach*. Springer Science & Business Media, 2003.

[16] Autonomous Vehicle Systems Laboratory, "Basilisk Documentation: Module *motorThermal*," `https://avslab.github.io/basilisk/Documentation/simulation/thermal/motorThermal/motorThermal.html`, 2024. Version 2.7.17, accessed 17 July 2025.

[17] maxon motor ag, "EC 60 flat Ø60 mm, brushless, 100 W — Datasheet (Part No. 647691)," `https://www.maxongroup.com/maxon/view/product/647691`, 2025. Thermal data accessed 17 July 2025.

[18] I. Pérez-Grande, A. Sanz-Andrés, C. Guerra, and G. Alonso, "Analytical study of the thermal behaviour and stability of a small satellite," *Applied Thermal Engineering*, Vol. 29, No. 11, 2009, pp. 2567–2573, https://doi.org/10.1016/j.applthermaleng.2008.12.038.

[19] Rocket Lab USA, "RW-0.06 Reaction Wheel — 60 mN·s Momentum Storage," `https://www.rocketlabusa.com/assets/Uploads/RL-RW-0.06-Data-Sheet.pdf`, 2021. Datasheet, Version 1, 14 July 2021.

[20] M. White and J. B. Bernstein, "Microelectronics Reliability: Physics-of-Failure Based Modeling and Lifetime Evaluation," jpl publication 08-5, Jet Propulsion Laboratory, NASA Electronic Parts and Packaging (NEPP) Program, Pasadena, CA, 2008.

[21] NASA, "Methodology for Physics-of-Failure Based Reliability," nasa/tp-2024-000000, National Aeronautics and Space Administration, 2024.

[22] D. Czerwinski, J. Gęca, and K. Kolano, "Machine Learning for Sensorless Temperature Estimation of a BLDC Motor," *Sensors*, Vol. 21, No. 14, 2021, 10.3390/s21144655.