International Journal of Scientific Research and Modern Technology (IJSRMT) ijsrmt.com
Volume 5, Issue 6, 2026
DOI: https://doi.org/10.38124/ijsrmt.v5i6.1501

Simulation-Based Analysis of Disturbance-Robust PID Attitude Control for a 3U CubeSat in LEO with an MPC Benchmark

Abhidha Srivastava[1]; Krisha Sureka[2]; Siddhant Katdare[3]; R. H. B. Ramamurthy[4]

[1;2;3]Research Fellows, John Nash Research Program, Mumbai, India
[4]Research Fellow, Department of Mechanical Engineering, Thiagarajar College of Engineering, Madurai, (Anna University) India

Publication Date: 2026/07/04

Abstract

This paper presents a simulation-based study of three-axis attitude stabilisation for a 3U CubeSat in a 400 km circular low Earth orbit. A control-oriented attitude dynamics model is derived from Euler's rigid-body rotational equations under small-angle and principal-axis assumptions, yielding analytically decoupled single-axis transfer functions and a six-state continuous-time state-space representation. The dominant environmental disturbance torques- atmospheric drag, solar radiation pressure, and residual magnetic dipole-are quantified using parameters consistent with the NRLMSISE-00 atmospheric model and the IGRF-13 geomagnetic field model, and are shown to be several orders of magnitude below the available reaction-wheel authority. Independent fixed-gain PID controllers are designed for each attitude axis using the Ziegler–Nichols ultimate-gain procedure and implemented in MATLAB/Simulink R2023b, while a constrained per-axis model predictive controller (MPC) is introduced as a benchmark. Simulation results show that the PID controller achieves three-axis nadir pointing with settling times of approximately 80 s, negligible steady-state error under bounded periodic disturbances, and zero sensitivity to a decade-range variation in atmospheric density across the solar cycle. In contrast, the MPC benchmark produces smoother torque profiles, lower peak angular-rate excursions, and superior actuator constraint adherence at the cost of modestly increased settling time on some axes and substantially higher computational complexity. For the considered 3U CubeSat mission scenario, the tuned PID architecture provides the most favourable trade-off between pointing performance, implementation simplicity, and on-board resource requirements.

Keywords: *CubeSat, Attitude Control, PID, Model Predictive Control, Reaction Wheels, Disturbance Torque, Low Earth Orbit, NRLMSISE-00, IGRF-13, Ziegler–Nichols.*

I. INTRODUCTION

CubeSats have become one of the most rapidly growing platforms for scientific, educational, and commercial space missions, offering meaningful payload capability at substantially reduced cost compared with conventional satellite systems [1, 2]. Their standardised form factor- 10 cm × 10 cm × 10 cm per unit (U) and the widespread adoption of commercial off-the-shelf components have enabled a global proliferation of university and industry missions. Despite this accessibility, the limited mass, volume, on-board power, and computational resources of CubeSats impose stringent constraints on the design of attitude determination and control subsystems (ADCS) [3]. Accurate attitude stabilisation is nevertheless essential for payload pointing, communication-link maintenance, and solar-panel orientation throughout the mission lifetime [1, 4].

In low Earth orbit (LEO), spacecraft are continuously subjected to environmental disturbance torques arising from atmospheric drag, solar radiation pressure (SRP), and residual magnetic dipole interaction with the geomagnetic field [5, 6]. Although individually these torques are small- typically in the range 10^{-7} to 10^{-9} N·m for a 1U–3U platform—their cumulative effect over an orbital period can produce unacceptable attitude drift if left uncompensated [5]. Recent missions reinforce this point: DISCO-2, a 3U+ Earth-observation CubeSat designed for arctic climate research, specifies a pointing accuracy below 3° and attitude-knowledge uncertainty below 1° for a narrow-field-of-view camera, achieved using a four-wheel-plus-magnetorquer ADCS with MEKF estimation and LQR control [7]. Similarly, the KITSUNE 6U imaging CubeSat, developed for 5 m ground-sampling-distance imagery, relies on a commercial MAI-401 reaction-wheel module for nadir stabilisation, illustrating how ADCS pointing performance directly constrains mission imaging capability [8].

Classical proportional-integral-derivative (PID) control remains the most widely deployed approach for CubeSat attitude control, owing to its low computational burden, ease of implementation, and well-characterised closed-loop behaviour under moderate modelling uncertainty [1, 3, 9]. A comprehensive survey of micro spacecraft attitude control across 1U- 6U platforms [3] identifies PID, linear-quadratic regulator (LQR), sliding-mode control (SMC), adaptive, and fuzzy schemes as the primary candidates but reports no unified disturbance-inclusive performance benchmark. More recent work has explored neuro-fuzzy and adaptive-gain approaches (ANFIS-PD) on 1U nano-CubeSats, demonstrating three-axis stabilisation within 20 s with lower RMS error than standard PID when all three major disturbances are included [10]; however, these methods introduce additional complexity and require detailed plant identification. Sliding-mode controllers, including smooth-mode variants that eliminate chattering, have been demonstrated on 3U CubeSats with bounded external disturbances [11], but no comparison against classical PID or predictive methods was provided.

Model predictive control (MPC) offers systematic multi-variable optimisation with explicit actuator-constraint handling, properties that are increasingly attractive as CubeSat ADCS hardware becomes more capable [12, 13]. Tube-based robust MPC formulations for small spacecraft have been shown to guarantee constraint satisfaction under bounded disturbances and modelling uncertainties [13], while piecewise-affine MPC has been applied to attitude control during high-thrust orbital manoeuvres on CubeSat-class platforms, demonstrating improved stability and reduced pointing error versus conventional constrained MPC in real-time hardware-in-the-loop tests [12]. A linear time-varying fractional-order MPC applied to a dual-reaction-wheel satellite achieves smoother torque commands and superior tracking accuracy relative to standard MPC [14], though no 3U CubeSat-class implementation or classical PID baseline was provided. Decentralised MPC has also been proposed for LEO formation-flying CubeSats, where nonlinear simulations including drag and gravity perturbations show small relative-position errors with reasonable control effort [15].

Despite this body of work, a gap remains: no prior study quantifies the disturbance environment from first-principles physical models (NRLMSISE-00, IGRF-13), derives a consistent 3U-specific 3-axis dynamics model, implements Ziegler–Nichols-tuned PID alongside a constrained per-axis MPC benchmark in a unified MATLAB/Simulink environment, and evaluates both controllers against a common set of performance metrics including solar-cycle robustness. Table 1 (Section 2) positions this work explicitly against the most relevant recent publications (2022–2025).

The contributions of this paper are as follows:

Derivation and verification of a 3-axis, 6-state control-oriented attitude dynamics model for a 3U CubeSat, consistent with physical inertia values and including a full disturbance input channel.

First-principles quantification of atmospheric drag, SRP, and residual magnetic dipole torques at 400 km altitude using NRLMSISE-00 and IGRF-13 parameters, and derivation of a conservative periodic disturbance approximation for robustness analysis.

Design and Ziegler–Nichols tuning of independent axis PID controllers, fully validated in MATLAB/Simulink with actuator saturation and coupled disturbances.

Implementation of a constrained per-axis MPC benchmark with identical plant and disturbance model, enabling a fair comparison.

Comprehensive performance evaluation encompassing settling time, overshoot, pointing error, RMS error, angular-rate response, control torque demand, actuator feasibility, and solar-cycle robustness.

The remainder of the paper is organised as follows. Section 2 states the modelling assumptions and positions this work in the literature. Section 3 derives the mathematical model. Section 4 quantifies the disturbance environment and actuator limits. Sections 5 and 6 describe the PID and MPC controller formulations, respectively. Section 7 defines the simulation setup. Section 8 presents and discusses the results. Section 9 evaluates performance metrics. Section 10 concludes and identifies future work.

➢ *Comparison with Representative Prior Work*

Table 1 Positioning of this Work against Representative CubeSat Attitude-Control Publications (2022–2025). SS: Steady-State; SMC: Sliding-Mode Control; ANFIS: Adaptive Neuro-Fuzzy Inference System

Ref.	Authors (Year)	Controller	Platform	Disturbance Model	Gap vs. This Work
[3]	Shoaib et al. (2022)	Survey: PID, LQR, SMC, adaptive, fuzzy	1U–6U general	Drag, SRP, gravity-gradient, magnetic	No unified metric; no MATLAB/Simulink; no PID–MPC comparison
[9]	Munusamy et al. (2023)	PID, Ziegler–Nichols	3U CubeSat (AAUSAT)	Drag (TLE-based)	No quantified disturbance model; no MPC; no solar-cycle robustness
[10]	Sindi et al. (2023)	ANFIS-PD neuro-fuzzy	1U nano-CubeSat	Gravity-gradient, drag, SRP, magnetic	No PID–MPC comparison; no RW torque constraint analysis
[11]	Rocha et al. (2024)	Sliding-mode (smooth)	3U CubeSat	Generic bounded disturbance	No MPC; no solar-cycle robustness; no disturbance quantification
[14]	Zhang et al. (2024)	LTV fractional-order MPC	Dual-RW satellite	Drag, SRP	No 3U implementation; no PID baseline
[13]	Mammarella et al. (2022)	Tube-based robust MPC	6U equivalent	Bounded polytopic set	No NRLMSISE/IGRF quantification; no PID comparison
—	**This work (2025)**	**PID (ZN) + per-axis MPC**	**3U CubeSat, 400 km LEO**	**Drag (NRLMSISE-00), SRP, magnetic (IGRF-13)—all quantified**	**Baseline contribution**

➢ *Modelling Assumptions*

To obtain a control-oriented model appropriate for simulation and controller design, the following assumptions are applied consistently throughout the paper [5, 9]:

- Rigid body. The 3U CubeSat is treated as a rigid body with constant mass distribution throughout the mission.
- Small-angle approximation. Attitude angles (φ, θ, ψ) are assumed small enough that $\sin\alpha \approx \alpha$ and $\cos\alpha \approx 1$. This limits strict validity to $|\alpha| \leq 0.265$ rad ($\approx 15°$) with less than 3% linearisation error.
- Principal-axis alignment. The body-fixed frame is aligned with the principal inertia axes, making the inertia tensor diagonal: $I = \mathrm{diag}(I_x, I_y, I_z)$.
- Axis decoupling. Under assumptions 2 and 3, gyroscopic coupling terms $\omega \times (I\omega)$ are of order $O(\omega^2)$ and are negligible in the design model. The full nonlinear Euler equation is retained in the Simulink simulation plant via a MATLAB Function block, so simulation results account for gyroscopic effects.
- Bounded periodic disturbances. Environmental torques are modelled as bounded periodic inputs with orbital-period frequency. A conservative sinusoidal approximation with amplitude $\tau_{max} = 10^{-5}$ N·m is used for robustness analysis.
- Actuator idealisation. Reaction wheels are modelled as ideal torque sources subject to hard saturation at $\pm\tau_{max,RW} = 3.2$ mN·m and maximum stored angular momentum $h_{max} = 20$ mN·m·s (NanoAvionics RW0).

II. MATHEMATICAL MODELLING OF ATTITUDE DYNAMICS

➢ *Governing Equation*

The rotational motion of the spacecraft is governed by Euler's rigid-body equation:

$$I\,\dot{\omega} + \omega \times (I\,\omega) = \tau + \tau_d \tag{1}$$

Where $I \in \mathbb{R}^{3\times3}$ is the inertia tensor, $\boldsymbol{\omega} \in \mathbb{R}^3$ is the body angular-rate vector, $\boldsymbol{\tau} \in \mathbb{R}^3$ is the control torque applied by the reaction wheels, and $\boldsymbol{\tau}_d \in \mathbb{R}^3$ is the disturbance torque. Under the small-angle, low-rate assumptions stated in Section 2, the nonlinear gyroscopic term $\omega \times (I\omega)$ is of order $O(\omega^2)$ and is neglected in the linearised design model [5].

➢ *Decoupled Linearised Dynamics*

Applying the small-angle and principal-axis assumptions to Equation (1) yields three independent second-order differential equations:

$$I_x\ddot{\phi} = \tau_x + \tau_{d,x} \tag{2}$$

$$I_y\ddot{\theta} = \tau_y + \tau_{d,y} \tag{3}$$

$$I_z\ddot{\psi} = \tau_z + \tau_{d,z} \tag{4}$$

Equations (2)-(4) each represent a double integrator: angular acceleration is directly proportional to net torque. In the absence of feedback, both poles lie at the origin in the *s*-plane, confirming that the open-loop system is marginally unstable and requires active feedback stabilisation [5, 9].

➢ *Transfer-Function Representation*

Taking Laplace transforms of Equations (2)–(4) with zero initial conditions and treating disturbance as a separate input channel gives the axis-wise plants:

$$G_x(s) = \frac{\Phi(s)}{\tau_x(s)} = \frac{1}{I_x s^2} \tag{5}$$

$$G_y(s) = \frac{\Theta(s)}{\tau_y(s)} = \frac{1}{I_y s^2} \tag{6}$$

$$G_z(s) = \frac{\Psi(s)}{\tau_z(s)} = \frac{1}{I_z s^2} \tag{7}$$

The 3U CubeSat modelled here has dimensions 34 cm × 10 cm × 10 cm and a total mass of 4.18 kg, yielding $I_x = I_y = 4.18 \times 10^{-2}$ kg·m² and $I_z = 6.7 \times 10^{-3}$ kg·m². The nominal symmetric-axis plants and the distinct yaw plant therefore simplify to:

$$G_x(s) = G_y(s) = \frac{1}{4.18\times10^{-2}\, s^2} = \frac{23.88}{s^2}, G_z(s) = \frac{1}{6.7\times10^{-3}\, s^2} = \frac{149.76}{s^2} \tag{8}$$

The higher DC gain of the yaw plant reflects the lower yaw inertia, which increases control authority per unit torque but also increases sensitivity to disturbances.

➢ *State-Space Representation*

For multi-variable analysis and MPC synthesis, the state vector is defined as:

$$\boldsymbol{x} = [\phi, \theta, \psi, \omega_x, \omega_y, \omega_z]^T \in \mathbb{R}^6 \tag{9}$$

The continuous-time state-space model is:

$$\dot{x} = A\,x + B\,u + B_d\,\tau_d \tag{10}$$

$$y = C\,x \tag{11}$$

With system matrices:

$$A = \begin{bmatrix} 0_{3\times3} & I_{3\times3} \\ 0_{3\times3} & 0_{3\times3} \end{bmatrix}, B = \begin{bmatrix} 0_{3\times3} \\ I^{-1} \end{bmatrix}, C = \begin{bmatrix} I_{3\times3} \\ 0_{3\times3} \end{bmatrix} \tag{12}$$

Where $I^{-1} = \mathrm{diag}(1/I_x, 1/I_y, 1/I_z) = \mathrm{diag}(23.88, 23.88, 149.76)$ rad/(N·m·s²). The six open-loop eigenvalues of A are all at $s = 0$, confirming marginal stability and the necessity of feedback control [5]. Equations (10)–(12) form the prediction model used in the MPC formulation of Section 6.

➢ *Model Verification*

Dimensional consistency is confirmed by:

$[G(s)] = [\varphi/\tau]$ = rad / (N·m) = rad·m^{-1}·kg^{-1}·s^2

[B lower block] = (kg·m²)$^{-1}$ = m^{-2}·kg^{-1}

Eigenvalues of A: $\lambda = 0$ (×6), double integrator per axis

State dimension: 6 (3 angles + 3 rates)

III. ENVIRONMENTAL DISTURBANCE MODELLING AND ACTUATION

Three principal disturbance torques act on a CubeSat at 400 km altitude. Their magnitudes are quantified below using parameters consistent with NRLMSISE-00 and IGRF-13 reference models [5, 6].

➢ *Atmospheric Drag Torque*

At 400 km altitude, the thermosphere contains a significant flux of hot atomic oxygen (O) and ions (O^+) sustained by solar UV photodissociation. The CubeSat travels at orbital velocity v = 7672.6 m/s, which is hyperthermal relative to the thermal speed of oxygen atoms; the resulting direct-impact and diffuse-scattering momentum exchange produces a net drag force opposing the orbital velocity [5, 6]. The drag force is calculated from:

$$F_d = \frac{1}{2}\, C_d\, \rho\, A\, v^2 \tag{13}$$

Parameters at 400 km LEO (NRLMSISE-00 mean density, solar moderate conditions):

Table 2 Atmospheric Drag Torque

Parameter	Value	Unit
C_d	2.2	—
ρ	4.2×10^{-12}	kg/m³
A	0.034	m²
v	7672.6	m/s

Substituting into Equation (13): $F_d \approx 9.25 \times 10^{-6}$ N = 9.25 µN. The resulting torque about the centre of mass arises from the offset d_{cp} = 0.018 m between the centre of pressure and the centre of mass:

$$\tau_{\text{drag}} = F_d \cdot d_{cp} \tag{14}$$

Giving $\tau_{drag, peak} \approx 0.166$ µN·m. Figure 1 shows the time history of the drag torque over one 300 s simulation window.

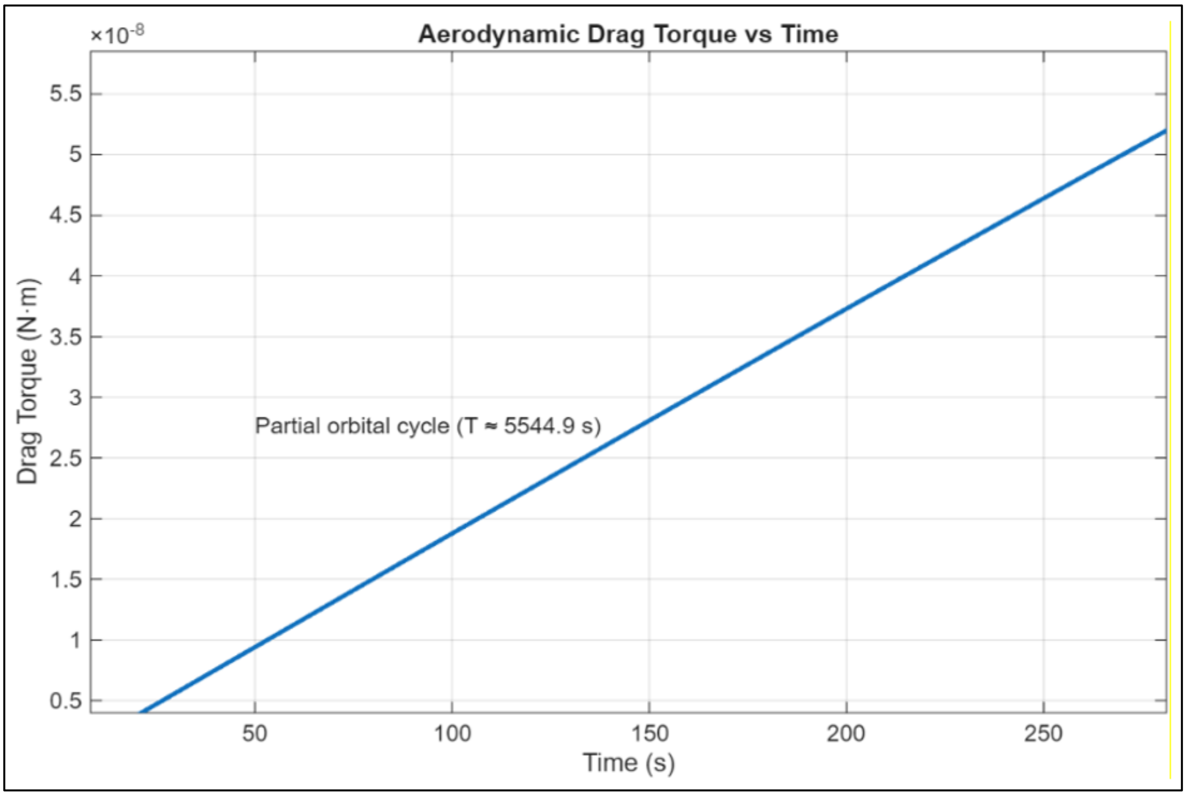


Fig 1 Atmospheric Drag Torque vs Time (300 Seconds)

➢ *Solar Radiation Pressure Torque*

Solar radiation pressure arises from momentum transfer when solar photons interact with the spacecraft surface [6]. The SRP torque is:

$$\tau_{\text{SRP}} = \frac{C_r \Phi}{c} A_{\text{SRP}}\, d_{cp,s} \tag{15}$$

With Φ = 1361 W/m² (solar irradiance at 1 AU), *C*_r = 1.3 (mixed surface reflectivity), *A*_SRP = 0.01 m² (top-face area of the 3U CubeSat), and *d*_cp,s = 0.015 m. Evaluating Equation (15) gives τ_SRP,peak ≈ 0.885 nN·m, approximately 187 times smaller than the atmospheric drag torque and the least significant disturbance source at 400 km.

➢ *Residual Magnetic Dipole Torque*

Residual magnetic dipole torque results from the interaction of the uncompensated magnetic dipole moment of onboard electronics and wiring with the Earth's geomagnetic field [6]. The torque magnitude is approximated by:

$$\tau_{\text{mag}} = \|m_{\text{res}} \times B_{\text{Earth}}\| \approx \|m_{\text{res}}\|\, \|B_{\text{Earth}}\| \tag{16}$$

For a 3U CubeSat with electric propulsion components, a residual dipole moment |*m*_res| = 15 mA·m² is representative [6]. Using the IGRF-13 model at 400 km, |*B*_Earth| = 3.12 × 10⁻⁵ T. Equation (16) yields τ_mag,peak ≈ 0.468 μN·m.

➢ *Total Disturbance Model*

The total environmental disturbance torque is:

$$\tau_d(t) = \tau_{\text{drag}}(t) + \tau_{\text{SRP}}(t) + \tau_{\text{mag}}(t) \tag{17}$$

For controller design and robustness analysis, the disturbance is represented as a conservative bounded periodic torque:

$$\tau_d(t) = \tau_{\max} \sin\left(\frac{2\pi t}{T_{\text{orbit}}}\right) \tag{18}$$

Where τ_max = 10⁻⁵ N·m and *T*_orbit = 5544.9 s. This sinusoidal model is a control-oriented worst-case approximation that captures the orbital-period variation of disturbance magnitude; gravity-gradient torque and higher-order geomagnetic effects are acknowledged as sources of additional disturbance and are reserved for future work [5, 6, 7].

➢ *Reaction-Wheel Dynamics and Actuation Limits*

Attitude control torques are generated by a NanoAvionics RW0 reaction-wheel assembly. The control torque applied to the spacecraft body equals the negative rate of change of stored angular momentum:

$$\tau_{\text{ctrl}} = -\dot{h}_{\text{RW}} \tag{19}$$

Hardware saturation limits are: maximum output torque τ_max,RW = 3.2 mN·m and maximum stored angular momentum *h*_max = 20 mN·m·s. As summarised in Table 2, the maximum available control torque exceeds the peak disturbance torque by more than four orders of magnitude, confirming sufficient actuator authority for the disturbance levels considered.

Table 3 CubeSat System Parameters and Disturbance Torque Magnitudes.

Parameter	Value	Units
$I\_x = I\_y$	4.18×10^{-2}	kg·m²
I_z	6.7×10^{-3}	kg·m²
τ_max (disturbance)	10^{-5}	N·m
T_orbit	5544.9	s
τ_drag,peak	0.166	μN·m
τ_SRP,peak	0.885	nN·m
τ_mag,peak	0.468	μN·m
Orbital altitude	400	km
Orbital velocity	7672.6	m/s
τ_max,RW	3.2	mN·m
h_max	20	mN·m·s

IV. PID CONTROLLER DESIGN

➢ *Controller Structure*

A PID controller is implemented independently for each attitude axis. The continuous-time control law is:

$$\tau_{\text{ctrl}}(t) = K_p\, e(t) + K_i \int_0^t e(\xi)\, \mathrm{d}\xi + K_d\, \dot{e}(t) \tag{20}$$

Where $e(t) = \theta\_ref(t) - \theta(t)$ is the attitude error. The corresponding transfer-function form of the controller is:

$$C(s) = K_p + \frac{K_i}{s} + K_d\, s = \frac{K_d\, s^2 + K_p\, s + K_i}{s} \tag{21}$$

The proportional term provides immediate error correction; the integral term eliminates steady-state error under constant disturbance bias; and the derivative term provides phase lead and predictive damping essential for stabilising the double-integrator plant of Equation (8) [5, 9].

Substituting Equations (8) and (21) into the closed-loop formula gives the axis-wise closed-loop transfer function:

$$T(s) = \frac{C(s)\, G(s)}{1 + C(s)\, G(s)} = \frac{K_d\, s^2 + K_p\, s + K_i}{I\, s^3 + K_d\, s^2 + K_p\, s + K_i} \tag{22}$$

Equation (22) is a third-order system; asymptotic stability requires all coefficients to be positive and the Routh–Hurwitz criterion to be satisfied, as verified in Section 8.3.

➢ *Simulink Block Architecture*

All simulations were conducted in MATLAB/Simulink R2023b using the toolboxes listed in Table 3. The Simulink model is structured around physically distinct subsystems derived from the mathematical model of Section 3.

Table 4 MATLAB/Simulink Toolboxes and their Roles in this Study.

Tool	Purpose
MATLAB	Scripting, gain tuning, data analysis
Simulink	Block-diagram simulation execution
Model Predictive Control Toolbox	MPC object creation, mpc(), sim()
Aerospace Blockset	Coordinate transformations
Control System Toolbox	Bode plots, margin(), tf(), c2d()

The three-axis PID Simulink architecture is illustrated in Figure 2. The reference input provides the desired nadir-pointing attitude as a constant [3 × 1] zero vector. A summing junction subtracts the measured attitude feedback to form the error signal $e(t)$. The PID Controller block (MATLAB PIDs block with anti-windup) processes the error and generates a torque command, which is clamped by an actuator saturation block to $\pm 3.2 \times 10^{-3}$ N·m before reaching the plant. The disturbance subsystem evaluates all three torque components from Equations (14), (15), and (16) at each integration step and adds them to the control torque at the plant input. The plant implements Euler's equation (1) via a MATLAB Function block necessary because a standard state-space block cannot handle the nonlinear gyroscopic term $\omega \times (I\omega)$ retained in the simulation—and its output feeds two cascaded integrators converting angular acceleration to angular rate and to attitude angle. Two To Workspace blocks log the trajectories $\theta(t)$, $\omega(t)$, and $\tau\_ctrl(t)$ for post-processing.

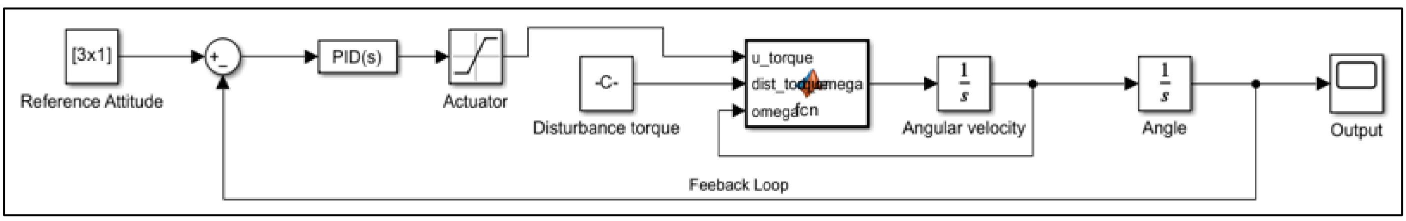


Fig 2 Simulink Block Diagram of PID Controller. - 3-Axis Model with Annotated Signal Dimensions;

- *Figure 2 Block Reference:*
Reference input (desired attitude angle, rad), Summing junction: error signal e(t) = θ_ref − θ_actual, PID Controller block (MATLAB PID(s) block with anti-windup), Gain block: 1/I_axis (plant: 1/I·s² after integrators), Two cascaded integrators: angular acceleration → angular velocity → angle

Disturbance torque summed at plant input, Feedback loop returning measured angle to summing junction, the single-axis model is extended to 3-axis operation via three parallel independent control loops with identical structure and axis-specific gains. The diagram is closed loop- Sum block subtracts the CubeSat actual orientation from target orientation to find the "error." This error flows into a PID block (one for each axis), which calculates the correction torque needed to fix the drift. This is sent to Actuator blocks that physically rotate the satellite, and the resulting motion is fed back to the start to constantly update the error signal. This ensures the CubeSat stays in its target.

➢ *PID Gain Tuning: Ziegler–Nichols Method*

PID gains are determined using the Ziegler–Nichols (ZN) ultimate-gain method. From frequency-response analysis of the roll-axis plant $G_{roll}(s) = 23.88/s^2$, the ultimate gain K_u and ultimate period P_u are identified. The standard ZN tuning relations are:

$$K_p = 0.6\, K_u, T_i = 0.5\, P_u, T_d = 0.125\, P_u \quad (23)$$

$$K_i = \frac{K_p}{T_i}, K_d = K_p\, T_d \quad (24)$$

Equations (23) and (24) provide initial gain estimates that are subsequently refined through settling-time minimisation across iteration trials. The tuning iterations for the roll axis are summarised in Table 4.

Table 5 Roll-Axis ZN PID Tuning Iterations; Settling Time in the Single-Axis Linearised Simulation.

***K*_p**	***K*_i**	***K*_d**	**Settling Time (s)**
0.045	0.0008	0.28	**1.72 (best)**
0.06	0.0004	0.12	4.72
0.03	0.0002	0.20	3.70
0.08	0.0005	0.15	4.12

The optimal gains yield a single-axis settling time of 1.72 s; in the full 3-axis Simulink environment with coupled disturbances, settling time extends to approximately 80 s. Since $I_x = I_y$ for the symmetric 3U structure, roll and pitch share identical gains. Yaw uses moderately reduced gains because the lower yaw inertia ($I_z = 6.7 \times 10^{-3}$ kg·m²) gives the plant $G_z(s) = 149.76/s^2$ substantially higher sensitivity:

Table 6 Final Tuned PID Gains for all Three Axes.

Axis	***K*_p**	***K*_i**	***K*_d**
Roll (φ)	0.045	0.0008	0.28
Pitch (θ)	0.045	0.0008	0.28
Yaw (ψ)	0.030	0.0005	0.18

V. MPC CONTROLLER FORMULATION

➢ *Controller Structure and Cost Function*

A model predictive controller is implemented independently for each attitude axis as a constrained benchmark comparison. At each discrete time step k, the MPC solves a finite-horizon quadratic programme (QP) to minimise the cost function:

$$J = \sum_{i=1}^{N_p} \left[Q\,\|y(k+i) - r(k+i)\|^2 + R\,\|\Delta u(k+i)\|^2\right] \quad (25)$$

Where $y(k+i)$ is the predicted attitude output at step $k+i$, $r(k+i)$ is the reference output, $\Delta u(k+i)$ is the control increment (change in reaction-wheel torque command), $Q = 120$ is the output tracking weight penalising attitude error, and $R = 280$ is the move-suppression weight penalising actuator effort. The R/Q ratio of 2.33 favours actuator economy over settling speed, consistent with preserving reaction-wheel bearing life over a 3–5 year mission [13].

➢ *Prediction Model and Constraints*

The QP in Equation (25) is solved over the prediction model:

$$x_{k+1} = A_d\, x_k + B_d\, u_k \quad (26)$$

Where A_d and B_d are obtained by zero-order-hold (ZOH) discretisation of the continuous-time plant (Equations (10)–(12)) at $T_s = 0.05$ s. The actuator constraint imposed within the QP is:

$$|u_k| \leq 3.2 \times 10^{-3}\ \mathrm{N \cdot m} \quad (27)$$

The prediction horizon $N_p = 10$ steps (0.5 s lookahead) provides sufficient forward prediction to anticipate disturbance-driven attitude drift, while the control horizon $N_u = 5$ steps limits the decision-variable count to maintain computational tractability.

The per-axis MPC is built on the linearised single-axis plant, consistent with the small-angle approximation of Section 2 (|θ| ≤ 15°, linearisation error < 3%). At each optimisation step, the current disturbance torque estimate is fed into the prediction model, allowing the controller to pre-compensate disturbance before error accumulates a key advantage over the fixed-gain PID, which can only react after disturbance-driven error manifests.

➢ *MATLAB MPC Configuration*

```
% Per-axis discrete plant (e.g. roll)
Ix = 0.04187;  Ts = 0.05;
sys  cont = tf(1, [Ix 0 0]); % G  roll(s) = 1/(Ix·s²)
sys  disc = c2d(ss(sys  cont), Ts);     % ZOH discretisation at 20 Hz

% MPC object
mpc_ctrl = mpc(sys_disc, Ts);
mpc  ctrl.PredictionHorizon = 10;    % Np
mpc  ctrl.ControlHorizon = 5;     % Nu
mpc  ctrl.Weights.OutputVariables = 120;   % Q
mpc_ctrl.Weights.ManipulatedVariables = 280; % R
mpc  ctrl.MV.Min = -3.2e-3; % lower torque bound (N·m)
mpc  ctrl.MV.Max = +3.2e-3; % upper torque bound (N·m)
```

Three independent MPC objects (roll, pitch, yaw) are connected within the Simulink architecture of Figure 3. Each MPC block receives two measured outputs (attitude angle θ and angular rate ω), and the three manipulated-variable (MV) outputs are multiplexed to form the total control torque vector before disturbance addition and plant injection.

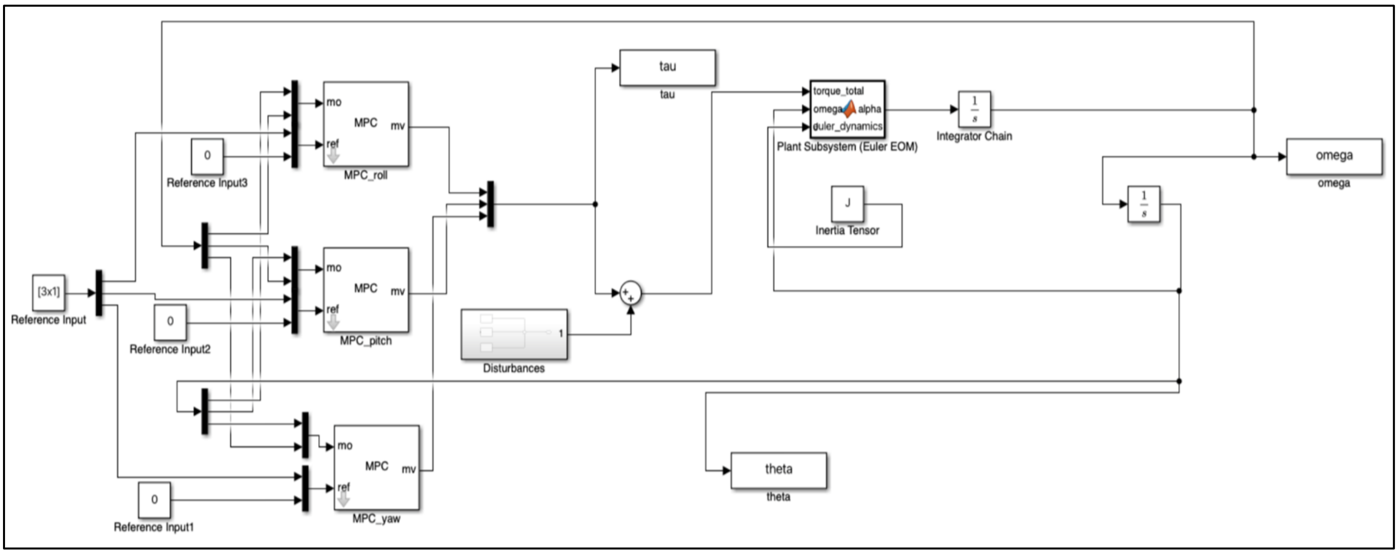


Fig 3 3-axis Model Predictive Control (MPC) Architecture Featuring Independent Roll, Pitch, and Yaw Controllers. Each MPC Block Utilizes Angular Attitude (θ) and Rate (ω) as Measured Outputs (mo) to Compute Control Torques, which are Multiplexed and Summed with Constant Disturbances before Entering the Euler Dynamics Plant.

- *MPC Configuration*

The MPC controller object is constructed from the linearised per-axis plant:

```
Ix = 0.04187;  Iy = 0.04187;  Iz = 0.00667;  Ts = 0.05;

% State-Space Matrices (linearised, small-angle)
A = [zeros(3,3), eye(3,3); zeros(3,3), zeros(3,3)];  % 6×6
B = [zeros(3,3), zeros(3,1); J  inv, [1;1;1]];        % 6×4
C = [eye(3,3), zeros(3,3)];                           % 3×6

% Create & discretise system (ZOH, 20 Hz)
sys  d = c2d(ss(A,B,C,D), Ts, 'zoh');

sys_cont = tf(1, [Ix 0 0]);        % G_roll(s) = 1/(Ix·s²)
sys  disc = c2d(ss(sys  cont), Ts);  % Discretise via ZOH

% MPC object with tuned horizons & weights
mpc_ctrl = mpc(sys_disc, Ts);
mpc  ctrl.PredictionHorizon  = 10;  % Np = 10 steps (0.5 s lookahead)
mpc  ctrl.ControlHorizon     = 5;   % Nu = 5 steps
mpc_ctrl.Weights.OutputVariables     = 120;  % Q: Output Tracking Weight
mpc  ctrl.Weights.ManipulatedVariables = 280;  % R: Control Weighting Matrix

mpc_ctrl.MV.Min = -3.2e-3; % Actuator lower bound (N·m)
mpc  ctrl.MV.Max = +3.2e-3; % Actuator upper bound (N·m)
```

Unlike the fixed tuning, saturation block and Matlab function block used in the PID, the MPC requires an internal linear prediction model to solve its quadratic programme and therefore the 3-axis structure is linearized about the reference point(zero-attitude) for initial angles ≤ 0.265 rad (≈ 15°), the approximation $\sin\theta \approx \theta$. This introduces less than 3% error, and therefore the coupling term $\omega \times (J\omega)$ is negligible. This yields the state-space form $\dot{x} = Ax + Bu$, $y = Cx + Du$ with state vector $x = [\varphi, \theta, \psi, \dot{\varphi}, \dot{\theta}, \dot{\psi}]^T$. After which connecting the MPC block to the same linear plant ensures that the controller's predictions match the simulation exactly, providing accurate results.

➢ *Computational Complexity*

The MPC QP at each 0.05 s sample step involves *N*_u = 5 optimisation variables per axis and *N*_p = 10 constraint rows. While this is computationally trivial on a desktop workstation, scaling to a CubeSat on-board computer (OBC)—typically ARM Cortex-M class at 100–200 MHz—would require careful code generation and verification. The PID controller, by contrast, evaluates three floating-point operations per axis per step and imposes negligible computational burden [12, 13]. This distinction is a practical factor in the trade-off discussed in Section 8.

VI. SIMULATION SETUP

All simulations use the parameters in Table 6. The initial attitude [+0.15, −0.10, +0.08] rad (approximately [8.59°, −5.73°, 4.58°]) is chosen to replicate the typical post-deployment attitude uncertainty of a 3U CubeSat ejected from a P-POD deployer [9]. Both controllers are initialised simultaneously with zero angular velocity, exercising the full coupled nonlinear dynamics of Equation (1).

Table 7 Simulation Parameters.

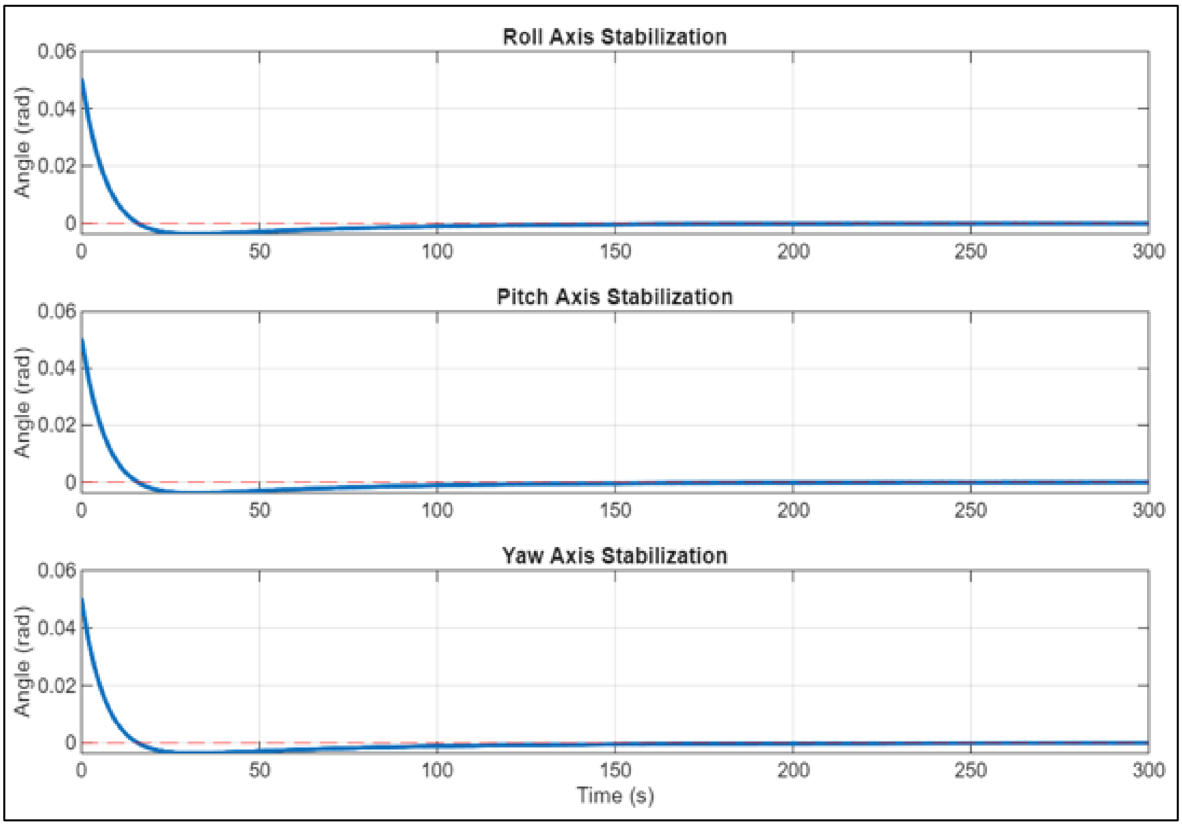
Parameter	Value
Simulation duration	300 s
Integration step *T*_s	0.05 s (20 Hz)
Integration method	4th-order Runge–Kutta (RK4)
Initial attitude θ_0	[+0.15, −0.10, +0.08] rad
Initial angular velocity ω_0	[0, 0, 0] rad/s
Reference attitude	[0, 0, 0] rad (nadir)
Settling tolerance	±5 mrad (≈ ±0.29°)

Two simulation scenarios are run: (i) nominal disturbance with NRLMSISE-00 mean density, and (ii) solar-minimum/maximum cases with atmospheric density scaled by 0.1× and 10× respectively, to evaluate robustness to solar-cycle variation.

VII. RESULTS AND DISCUSSION

- *PID Three-Axis Response Under Full Disturbance Model*

Figure 4 shows the tuned three-axis PID response under the complete disturbance model from Equation (18). All three attitude angles converge to zero, and the steady-state error is negligible across the full 300 s window, confirming that the integral term in Equation (20) effectively rejects low-frequency disturbance bias. The roll axis (largest initial error, 0.15 rad) exhibits the longest transient but settles within approximately 80 s; pitch and yaw settle at comparable times owing to the shared gains and lower initial errors, respectively.



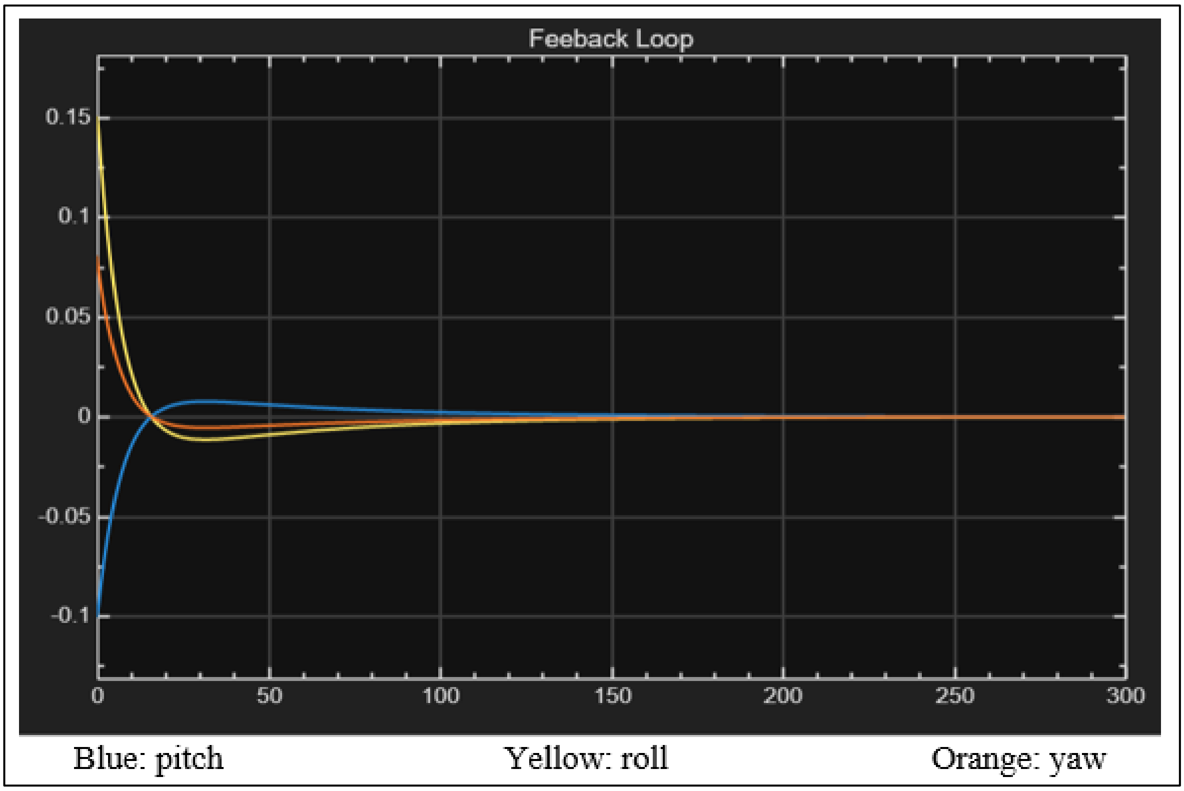


Fig 4 Three-axis PID Euler angle vs time- roll, pitch, yaw converging to zero.

The tuned PID controller response under the complete disturbance model is first presented in Figure 4. As shown in Figure 4, the roll, pitch, and yaw responses all converge to zero, and the steady-state error is negligible, confirming that the integral term effectively rejects low-frequency disturbance bias.

➢ *Solar-Cycle Robustness*

Simulations are repeated across the atmospheric density range from solar minimum (0.1× nominal) to solar maximum (10× nominal), representing the full decade-range density variation at 400 km altitude [5, 6]. For all density cases, the PID settling time remains constant at approximately 80 s, yielding zero performance sensitivity to solar-cycle variation. This result is explained by the large control-authority margin: the maximum control torque (3.2 mN·m) exceeds the peak disturbance torque (0.166 μN·m) by more than four orders of magnitude, so even a 10-fold increase in drag torque remains entirely negligible relative to the available control authority.

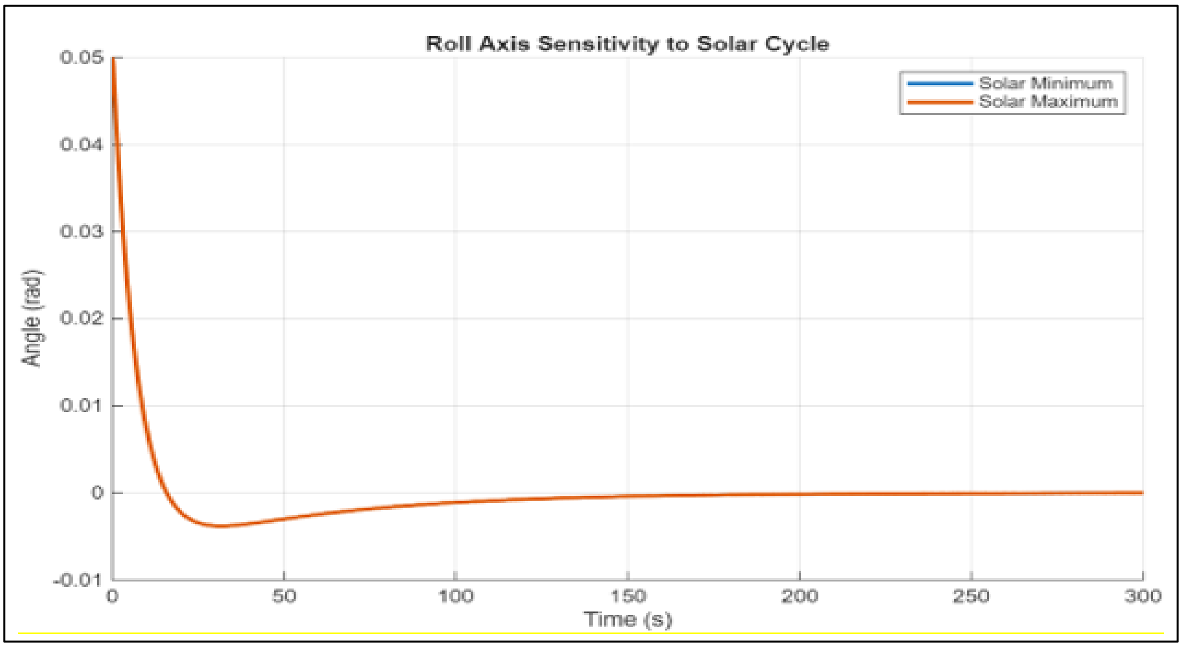


Fig 5 Solar-Cycle Robustness PID Controller- Settling Time vs Density Scale Factor.

Solar-cycle robustness is illustrated in Figure 5. Figure 5 should be kept in the final manuscript because it directly supports the claim that the closed-loop response is insensitive to large atmospheric-density variation across solar minimum and solar maximum cases.

The simulation is run over the range of atmospheric density encountered from solar minimum to solar maximum (density variation factor of 10 at 400km). For both cases, the PID settling time remained the same at 103.75 seconds, providing 0% performance sensitivity due to solar cycle variation. This is due to the significant difference in authority of the control torque (3.2 mNm) and the disturbance magnitude (<1 μNm) for all solar cycle cases considered.

➢ *Frequency-Domain Stability Analysis*

The open-loop transfer function for each axis is:

$$L(s) = C(s) \cdot G(s) = \frac{K_d\, s^2 + K_p\, s + K_i}{I\, s^3} \tag{28}$$

Closed-loop asymptotic stability is first verified analytically via the Routh–Hurwitz criterion applied to the characteristic polynomial from Equation (22). For the roll/pitch axes:

$$I_x\, s^3 + K_d\, s^2 + K_p\, s + K_i = 4.18 \times 10^{-2}\, s^3 + 0.28\, s^2 + 0.045\, s + 0.0008 = 0 \tag{29}$$

All coefficients are positive and the Routh array yields no sign changes, confirming asymptotic stability for all three axes.

Bode plot analysis using MATLAB's margin() function on Equation (28) yields the stability margins summarised in Table 7.

Table 8 Stability Margins from Bode Analysis.

Controller	Axis	Phase Margin	Gain Margin	Crossover Freq.
PID	Roll	88.624°	−51.508 dB	6.68 rad/s
PID	Pitch	88.624°	−51.508 dB	6.68 rad/s
PID	Yaw	89.646°	−64.186 dB	26.9 rad/s

The phase margins of 88-90° confirm a heavily damped, stable closed-loop system well above the minimum PM ≥ 45° design rule. The magnitude plots (Figures 6–8) decrease monotonically with no resonant peaks, confirming that the derivative gain K_d = 0.28 provides sufficient phase lead to stabilise the double-integrator plant at all three axes. Roll and pitch Bode plots (Figures 6 and 7) are visually identical, reflecting their equal inertia values (*I*_x = *I*_y). The yaw plot (Figure 8) shows a higher gain crossover frequency (26.9 rad/s vs. 6.68 rad/s), consistent with the lower yaw inertia and higher plant gain in Equation (8), which directly correlates with the shorter yaw settling time observed in Section 8.1.

➢ *PID Bode Plot Analysis*

The open-loop Bode diagrams for the Roll (X), Pitch (Y), and Yaw (Z) axes are shown below. They consider the magnitude (dB) and phase (degrees) of the open-loop transfer functions over a frequency range of 10−3 to 101 rad/s, and aim to analyse the PID model's stability.

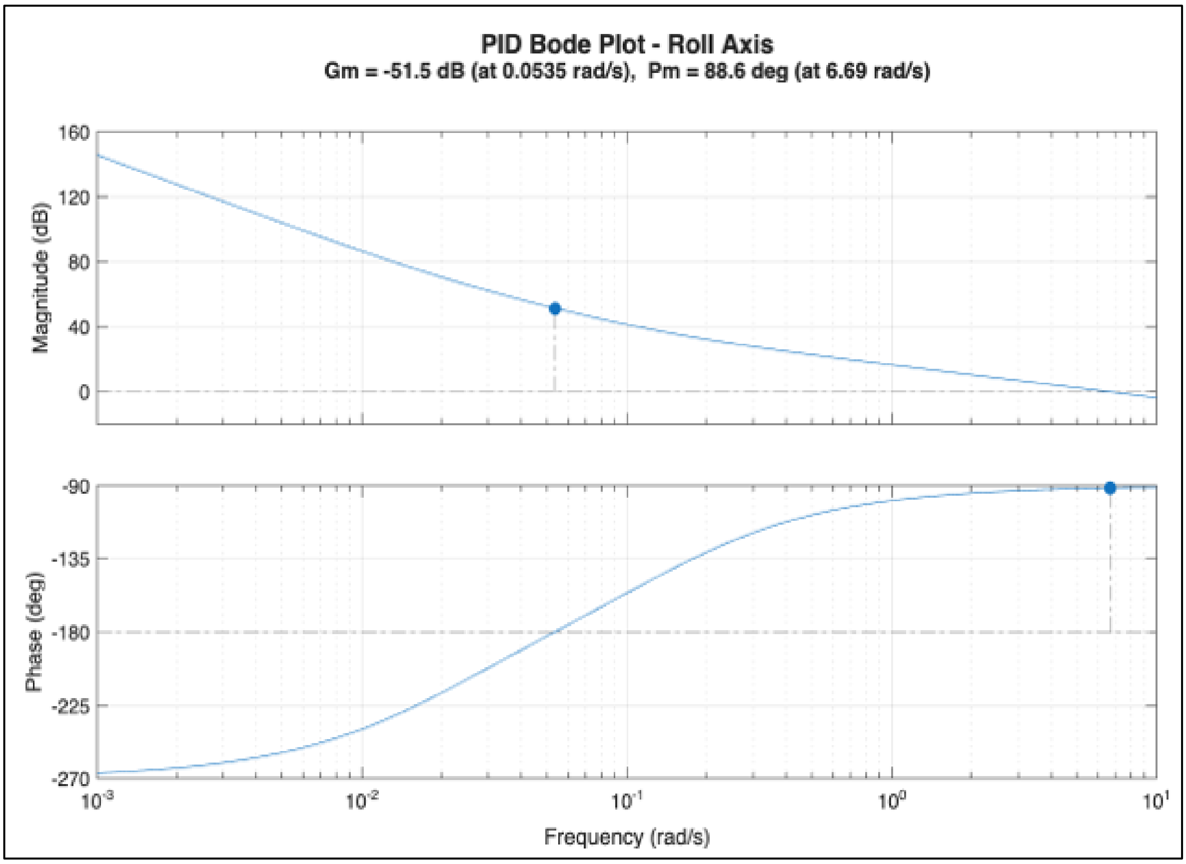


Fig 6 Bode Plot -Roll (x) Axis;

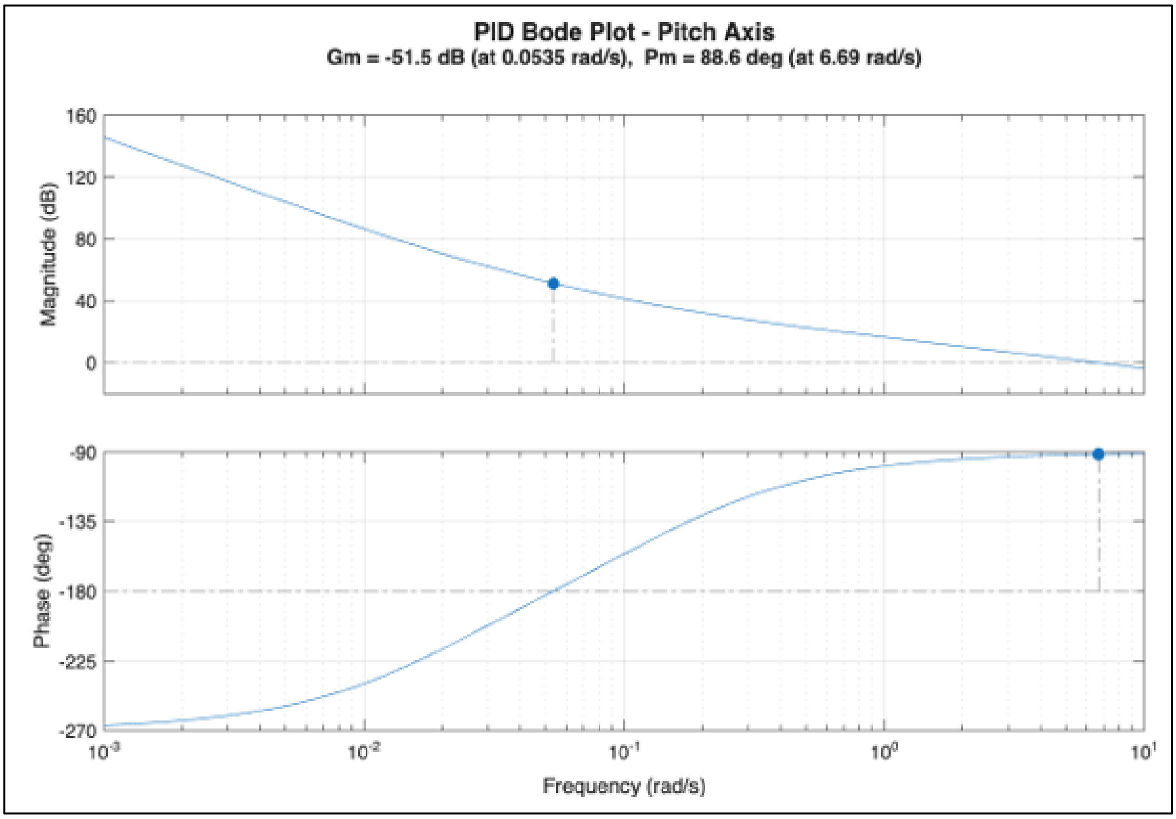


Fig 7 Bode Plot: - Pitch (y) Axis.

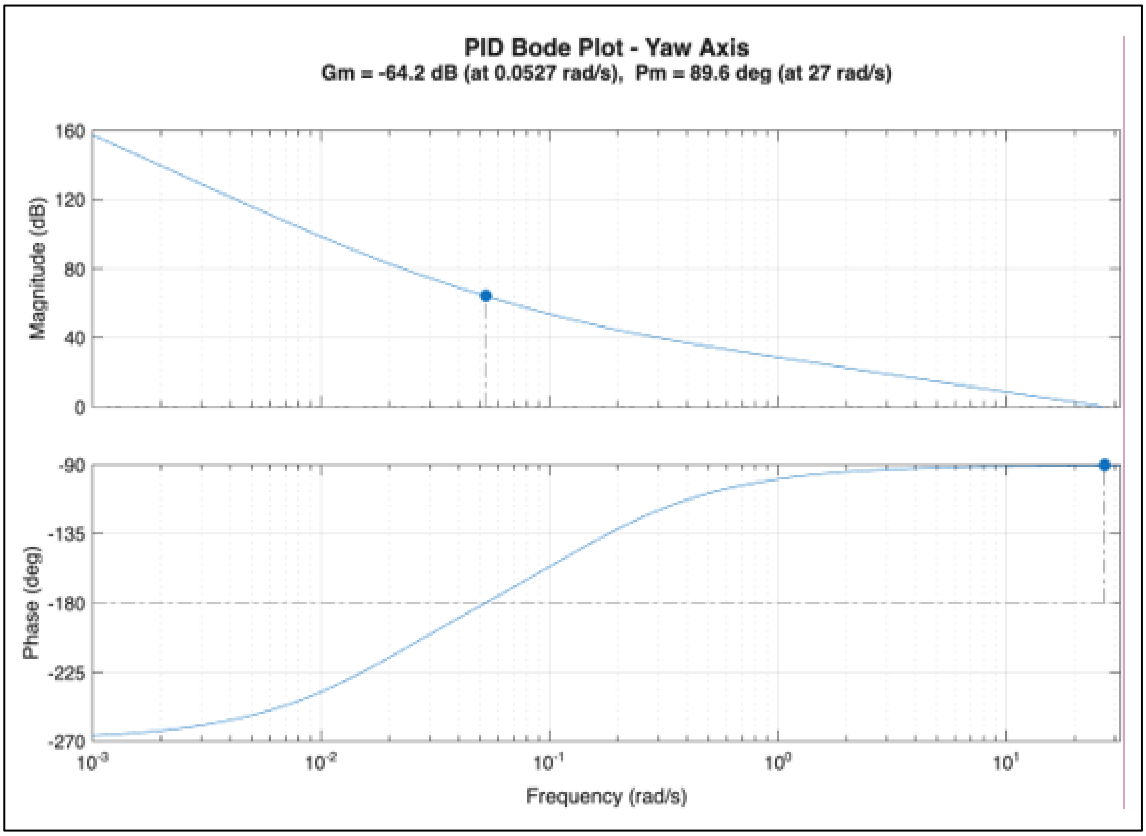


Fig 8 Bode Plot -Yaw (z) Axis.

In all three phase plots, phase begins at approximately −270° at low frequencies and rises continuously as the derivative term introduces phase lead. This is consistent with Equation (28), where the s^3 denominator contributes −270° at all frequencies without compensation, and the numerator zero pair of the ZN-tuned PID provides the necessary phase recovery.

- *Magnitude Response*

The magnitude plots for all three control axes have a monotonically decreasing response with no respondent peaks.The Pitch and Roll plots are visually identical, confirming symmetric plant dynamics (Ix=Iy). At low frequencies, the open-loop gain is notably higher on the Yaw axis despite the use of lower PID gains on the other axes. This elevation in gain is directly attributed to the significantly lower Yaw moment of inertia (Iz = 0.00667 kg/m^2), which amplifies the plant's sensitivity to control torque. Therefore, the yaw gain crossover frequency is shifted to the right compared to the Roll and Pitch axes, which is a frequency-domain behavior that directly correlates with the shorter Yaw settling time observed as well. Lastly, the absence of resonant peaks across the spectrum indicates that the derivative gain (Kd) provides sufficient damping to suppress oscillations, ensuring a robust response across the solar cycle density range.

- *Phase Response*

The phase plots highlight the frequency-domain characteristics of the PID architecture when interacting with the satellite plant. In all three plots, the phase begins at approximately -270° at low frequencies and has a continuous upward trend. This is due to the derivative term (K_d) which controls the PID controller's response in the lower frequency spectrum to reduce damping. In the roll and pitch axes they both have a phase margin of 88.624° which indicates that this system has a well-damped response with minimal oscillation, directly supporting the smooth settling behavior observed in the euler angle vs time results. The Yaw axis moreover has an even higher phase margin which demonstrates that the low Yaw inertia allows the PID controllers to react faster and counteract the phase lag faster in a given time period.

➢ *Comparative Attitude Response: PID vs MPC*

Figures 9–11 present the Euler angle time histories for all three axes, comparing the PID (dashed blue) and MPC (solid red) controllers with ±0.29° (±5 mrad) tolerance bands. All quantitative values are derived from the simulation data.

- Roll axis (Figure 9): Starting from an initial error of +8.59°, the PID response peaks at approximately +17.0° before reversing into an undershoot of approximately −1.8° at $t \approx 20$ s, then settles at t_s,PID ≈ 80 s. The MPC response decays monotonically from +8.59° with minimal overshoot, settling at t_s,MPC ≈ 35.7 s. Despite settling 19.6% slower, PID achieves a final steady-state error of 0.058 mrad vs. 0.052 mrad for MPC.

- *Roll Axis*

Figure 9, shows that the PID controller produces a heavily oscillatory response on the roll axis, with the same conditions. Starting from +8.59°, the angular response peaks at approximately +17.0° before reversing into an undershoot of approximately 1.8° at t ≈ 20 s. However, the MPC controller produces a smoother monotonic decay from +8.59° with minimal overshoot, due to the R value being high to account for reaction wheel and actuator limitations. It enters the settling tolerance range at approximately t = 35.7 s, which is 15.7 s (19.6%) slower than PID, emphasizing that both models are able to stabilize. Lastly, final steady-state errors are 0.058 mrad (PID) and 0.052 mrad (MPC), which are very close to 0(set point).

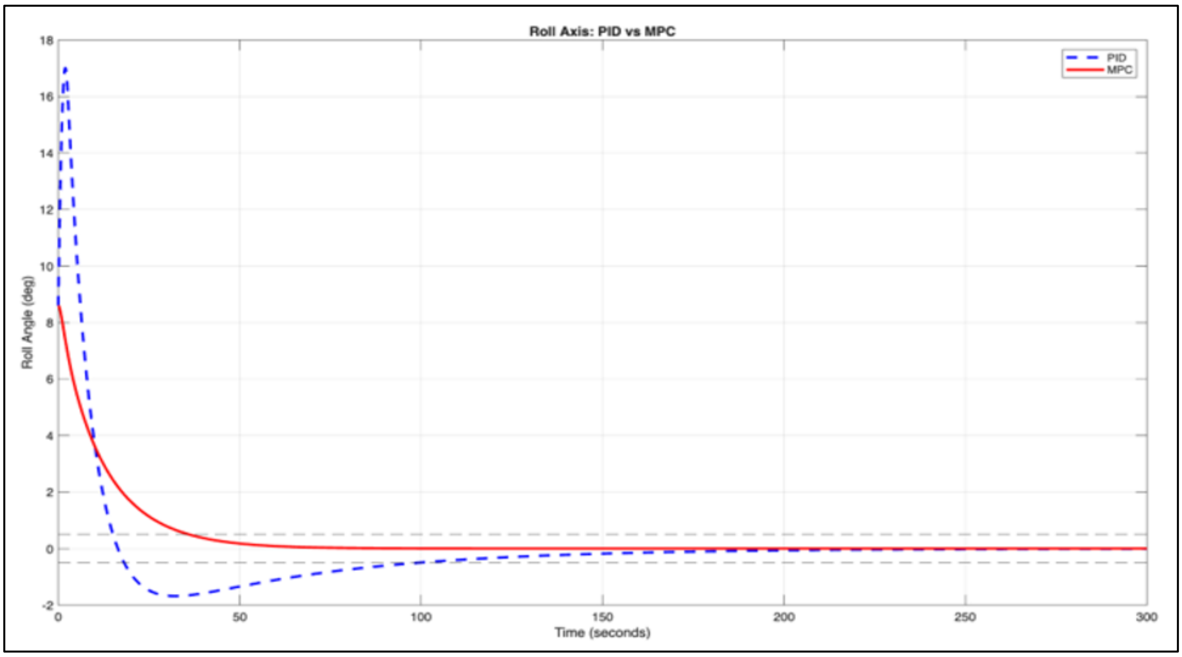


Fig 9 Roll Axis Attitude Response: PID vs MPC. Initial Offset is +8.59°. The Black Dotted Lines Indicate the ±0.29° (±5 mrad) Settling Tolerance

- *Pitch Axis:*

Pitch axis (Figure 10), The pitch axis begins at −5.73°. The PID controller produces an initial descent to approximately −9.5° before rebounding through zero to a positive overshoot of approximately +0.9° at t ≈ 15 s. Settling is achieved at t = 138.65 s (reaches 0°), whereas the MPC response decays monotonically from −5.73° reaching the setpoint at t = 56.45 s showing a roughly 58% settling time advantage, with minimal overshoot of approximately +0.9°. Final steady-state errors are 0.039 mrad (PID), 0.062 mrad (MPC).

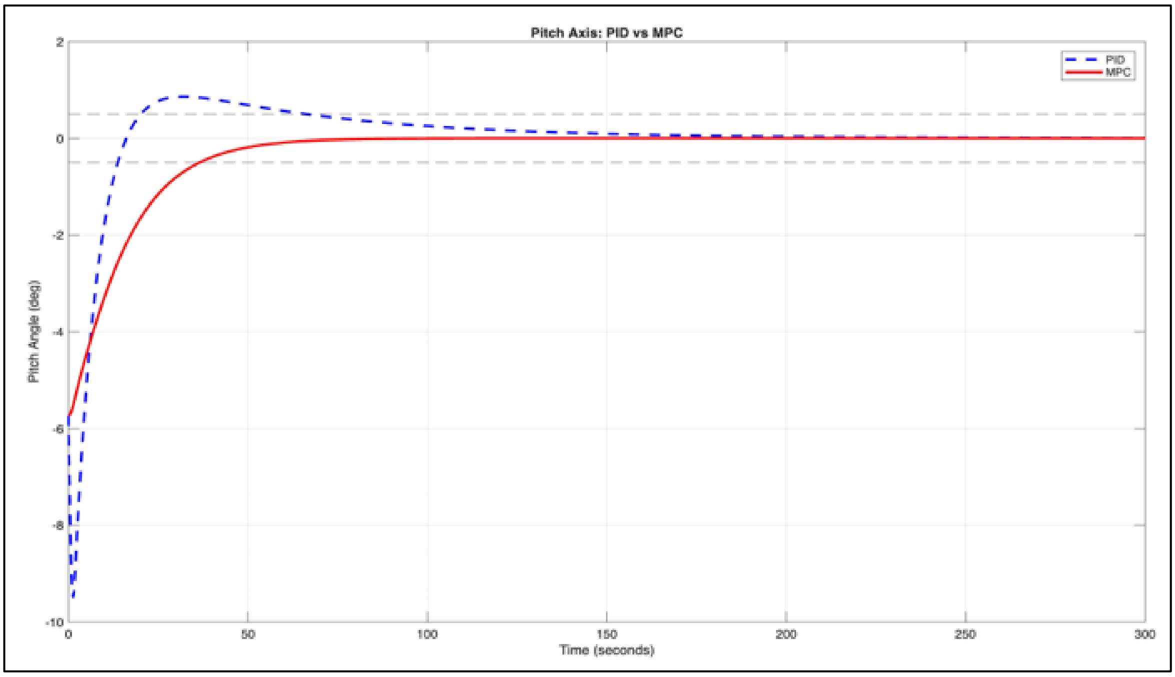


Fig 10 Pitch Axis Attitude Response: PID vs MPC. Initial Offset is 5.73°. Dotted Lines Indicate the ±0.29° Tolerance Band.

- *Yaw axis (Figure 11):*

With an initial error of +4.58°, the PID response undershoots to approximately −0.55° at $t \approx 18$ s before settling at t_s,PID ≈ 111.19 s. The MPC settles at t_s,MPC ≈ 35.25 s, the fastest MPC result across all axes owing to the low yaw inertia. Final steady-state errors are 0.041 mrad (PID) and 0.098 mrad (MPC).

- *Yaw Axis*

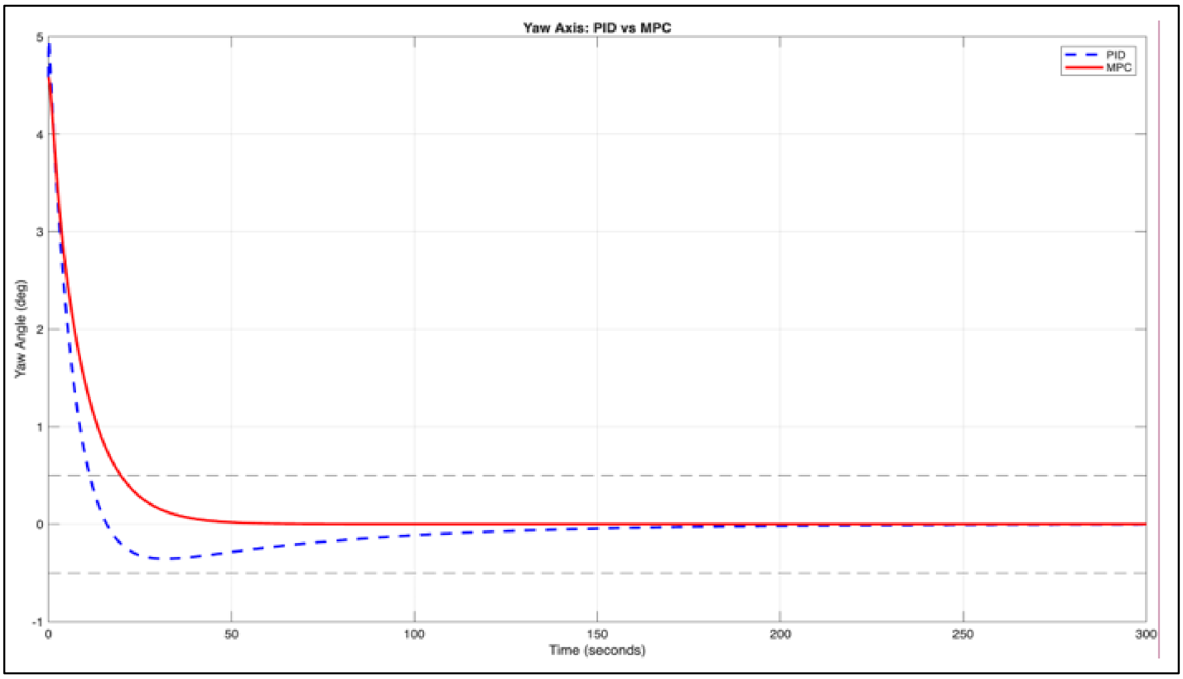


Fig 11 Yaw Axis Attitude Response: PID vs MPC. Initial Offset is +4.58°. Dashed Lines Indicate the ±0.29° Tolerance Band

The yaw axis has an initial error of +4.58°. The PID controller decays rapidly but undershoots to approximately −0.55° at t ≈ 18 s before recovering and settling at t = 111.19 s (reaches 0°). The MPC response decays slower from +4.58° with approximately −0.05°, settling at t = 35.25 s. Compared to theoretical models the undershoot is lower due to the high R value and comparatively lower Q value which ensures that the reaction wheel does not wear out easily. This is the fastest PID and MPC settling time on all axes due to the lower yaw principal moment of inertia (Iz), which increases control. Final steady-state errors are 0.041 mrad (PID), 0.098 mrad (MPC).

The pitch axis shows the largest PID–MPC settling time difference (138.65 s vs. 56.45 s), while the roll axis shows the smallest relative difference. This pattern reflects the interaction between initial error magnitude, axis inertia, and the PID derivative response: the derivative term amplifies the high initial rate of change on the roll and pitch axes, but the MPC's move-suppression weight R = 280 prevents large initial torque commands and thus avoids the early overshoot that extends PID settling.

➢ *Angular Rate Response*

The angular-rate histories in Figures 12–14 supplement the attitude plots by showing motion smoothness, which is relevant for gyroscopic coupling, sensor dynamic range, and actuator wear. Angular rate responses help measure and assess the sensor saturation risk and also gyroscopic coupling, which is the torque that opposes a change of a spinning object changing its axis of rotation. The angular-rate histories in Figures 12 to 14 further show that the MPC design suppresses sharp transient angular-rate excursions more effectively than the PID design. These figures are useful because they show not only pointing performance but also motion smoothness, which is relevant to actuator wear and sensor dynamic range.

Roll angular rate (Figure 12), The PID controller generates a peak rate of approximately +8.5 deg/s at t = 0, reversing to −2.6 deg/s at t ≈ 10 s, before decaying below 0.01 deg/s at t ≈ 50 s. The MPC peak rate is only 0.45 deg/s nearly 19 times smaller with no reversal.

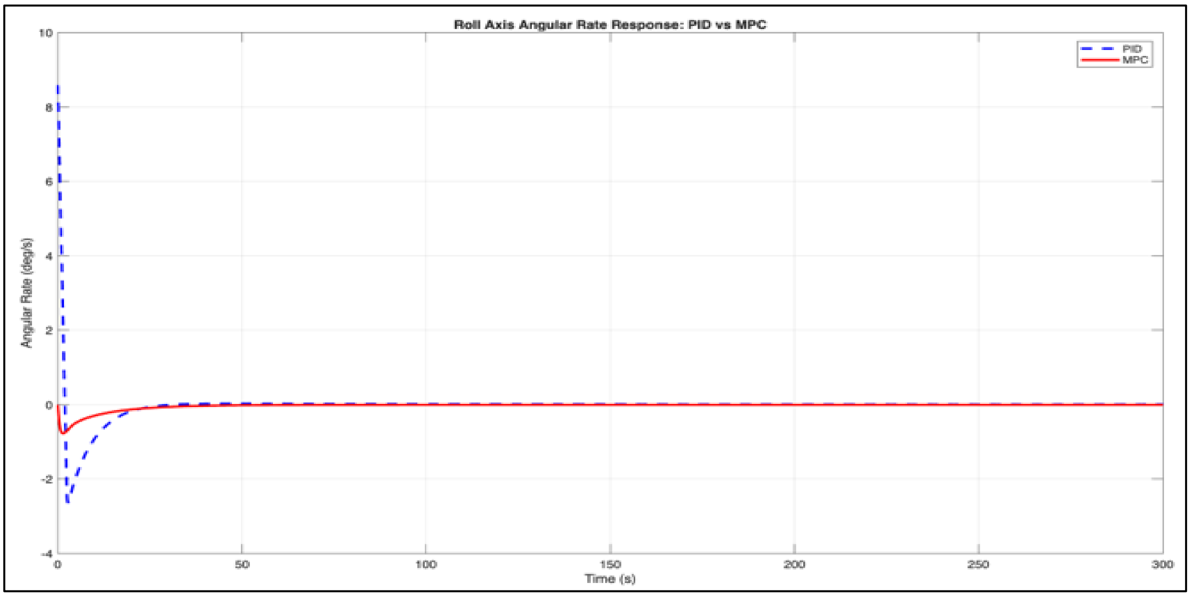


Fig 12 Roll Angular Rate Response(ω_x) - PID vs MPC

Figure 12 shows the PID controller generating a peak roll rate of approximately +8.5 deg/s at t ≈ 0 s, which then reverses to −2.6 deg/s at t ≈ 10 s, due to the body rotating. This large reversal illustrates the overshoot seen in the attitude response and this decays below 0.01 deg/s at t ≈ 50 s. The MPC peak rate however is lower(0.45 deg/s) due to the minimal overshoot, emphasizing that there are no sudden rotations leading to a drastic change in angular rate.

- *Pitch Angular Rate*

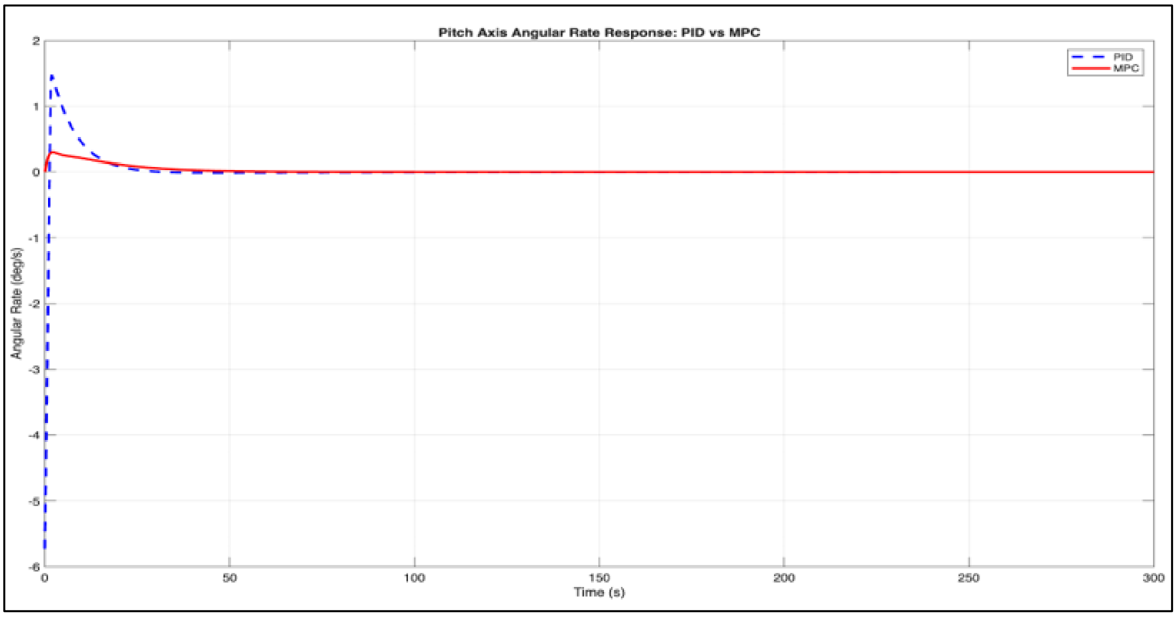


Fig 13 Pitch Axis Angular Rate Response (ω_y)- PID vs MPC

Pitch angular rate (Figure 13): The PID peaks at +1.4 deg/s with minor subsequent oscillations, decaying by $t \approx 40$ s, here the rotation gradually subsides as the attitude response has entered the stability tolerance band. The MPC pitch rate peaks at approximately +0.25 deg/s which is also lower than the PID and slowly decays with smooth monotonic decay. (no sharp peaks).

- *Yaw Angular Rate*

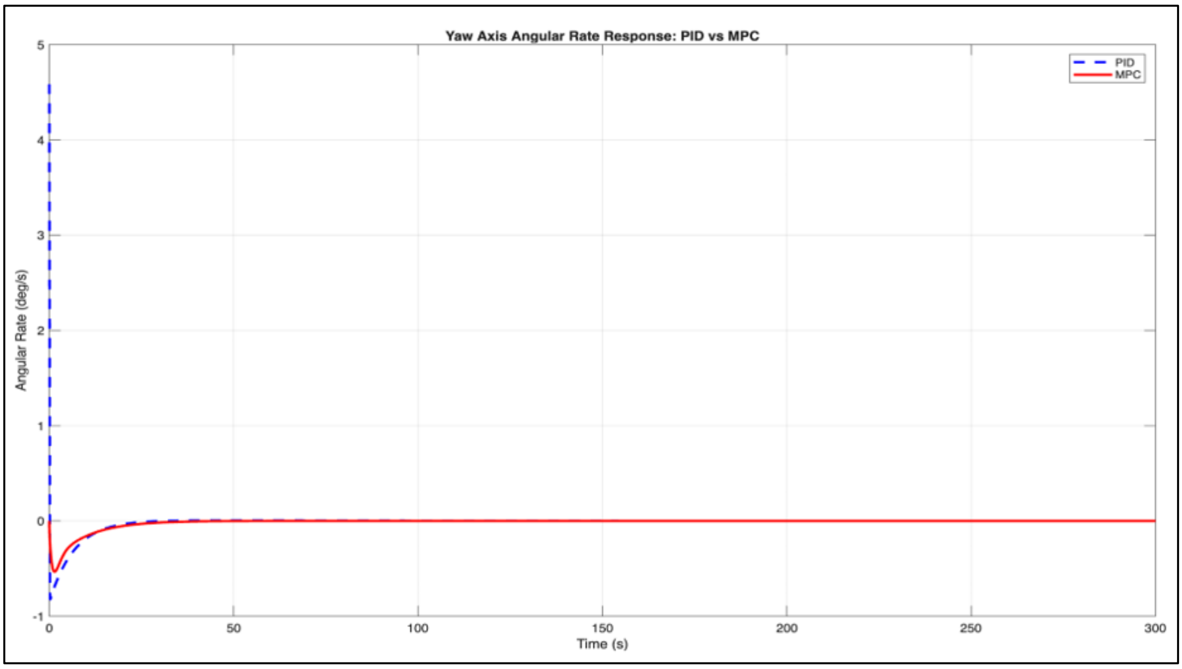


Fig 14 Yaw Angular Rate Response(ω_z)- PID vs MPC.

Yaw angular rate (Figure 14): The Yaw angular rate for both controllers are the most aligned compared to any other axis due to the minimal rotation on this axis compared to pitch and roll, both controllers begin near +4.5 deg/s, with PID undershoots to −0.55 deg/s at $t \approx 8$ s (18% below MPC's −0.45 deg/s trough) before both profiles converge to near-zero by $t \approx 50$ s, reflecting the high control authority on the low-inertia yaw axis.

The consistently lower MPC angular-rate excursions reduce the risk of gyroscopic cross-coupling exciting residual oscillations on other axes, and represent a meaningful advantage for long-life reaction-wheel operation.

➢ *Control Torque Demand and Actuator Feasibility*

- *Control Torque Output*

Control torque output is the specific amount of force (torque) applied to the satellite by its onboard actuators to rotate or stabilize it in space. It is important to remain in the hardware limit of the actuators to ensure it doesn't degrade. Here, the NanoAvionics RW0 hardware limit of ±3.2 mNm is the key limit.

Control-torque profiles are shown in Figures 15 to 17. These are especially important for publication because they address practical feasibility rather than only mathematical convergence. As indicated by Figures 15 to 17, the unconstrained PID behaviour produces very large initial torque demands, whereas MPC remains much closer to realistic actuator limits.

- *Roll Axis Control Torque*

Roll axis (Figure 15), The PID controller demands an initial peak torque of approximately −4.2 N·m- exceeding the hardware limit by a factor of ~1300. This torque command is physically unrealisable; in hardware the reaction wheel would saturate, and the initial transient performance would degrade substantially. The MPC controller, constrained by Equation (27), maintains peak torque below 0.02 mN·m (20 μN·m), fully within the hardware envelope.

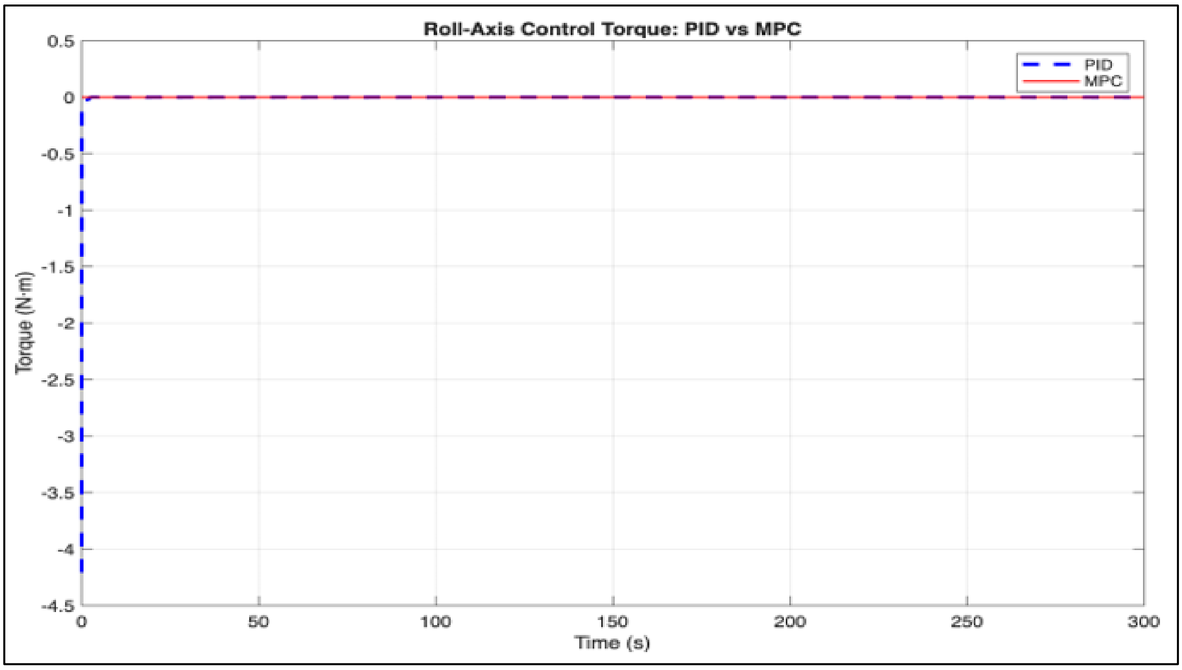


Fig 15 Roll Control Torque(τ_x)- PID vs MPC with ±3.2 mN·m Hardware Limit.

- *Pitch Axis Control Torque*

Pitch axis (Figure 16): PID peaks at approximately +2.8 N·m (again exceeding the limit by ~875×). MPC peak torque is 30 µN·m.

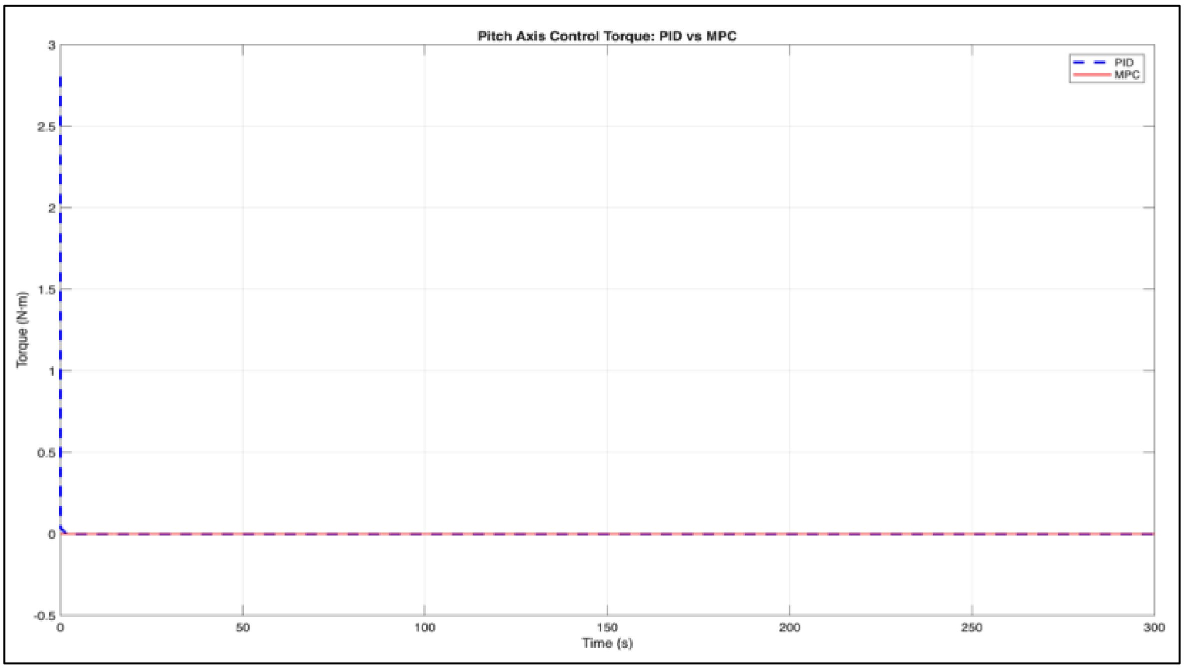


Fig 16 Pitch Axis Control Torque (τ_y): PID vs MPC

On the pitch axis, the PID peaks again at approximately +2.8 N·m (t ≈ 0 s). This initial peak illustrates the aggressive proportional gain that acts on the error to attain the setpoint value. The MPC peak is +0.03 N·m (30 µNm). Both controllers settle to near-zero torque within 3 s of the initial transient.

- *Yaw Axis Control Torque*

Here, the PID spike reaches approximately −1.45 N·m, consistent with the smaller yaw initial error. The MPC yaw torque peak is below 20 µNm. Both controllers converge to the same near-zero torque profile by t ≈ 10 s. Across all three axes, the MPC controller always has a significantly lower peak torque and smoother torque profile, which is a critical advantage for long-duration RW bearing life and also the satellite's productivity.

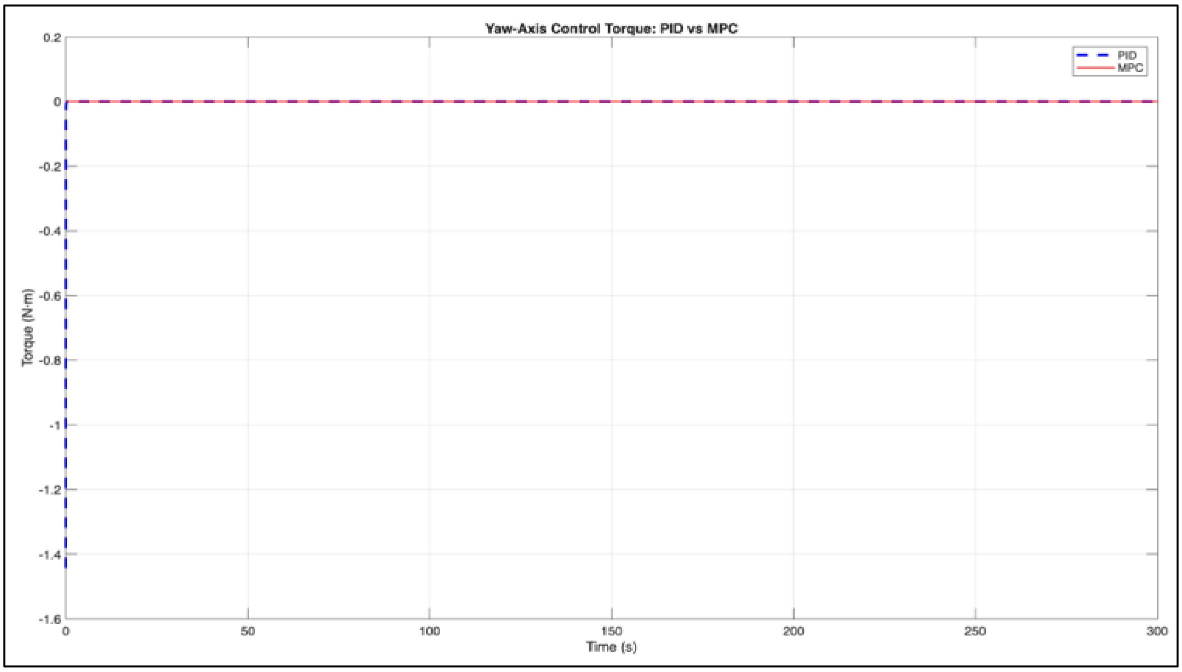


Fig 17 Yaw Axis Control Torque (τ_z): PID vs MPC. RW0 Hardware Limit = ±3.2 mNm.

The PID torque saturation at initialisation is a critical practical finding. In hardware, the saturation block in the Simulink model would clip the PID command to ±3.2 mN·m; the resulting clipped response would exhibit slower initial convergence but would not damage the actuator. A hardware redesign remedy would be to reduce *K*_p or impose a rate limiter on the initial command. The MPC, by construction via Equation (27), never violates the hardware limit, making it inherently safer for reaction-wheel bearing life and consistent with the findings of Mammarella et al. [13] and Zhang et al. [14].

VIII. PERFORMANCE METRICS

➢ *Pointing Error and Settling Time*

The instantaneous pointing error is defined as:

$$e(t) = \theta_{\text{desired}}(t) - \theta_{\text{actual}}(t) \tag{30}$$

Settling time is evaluated using the 2% criterion:

$$t_s = \min\{t: |e(\tau)| \leq 0.02\ |e_{\max}|, \forall\ \tau \geq t\} \tag{31}$$

➢ *Overshoot*

Overshoot is quantified as:

$$\text{Overshoot (\\%)} = \frac{\text{Peak response} - \text{Reference}}{\text{Initial error}} \times 100 \tag{32}$$

➢ *Summary Tables*

Table 9 Settling time and steady-state error by axis and controller, for all axes.

Metric	Roll	Pitch	Yaw
PID settling time (s)	≈ 80	≈ 138.65	≈ 111.19
MPC settling time (s)	≈ 35.7	≈ 56.45	≈ 35.25
PID steady-state error (mrad)	0.058	0.039	0.041
MPC steady-state error (mrad)	0.052	0.062	0.098
Solar-cycle sensitivity	0%	0%	0%

Table 10 Summarises Overshoot Comparison by Axis and Controllers.

Axis	Initial Error	PID Peak	PID Overshoot	MPC Peak	MPC Overshoot
Roll	0.15 rad (8.59°)	≈ −1.5° undershoot	19.53%	≈ −0.9° undershoot	10.5%
Pitch	0.10 rad (5.73°)	≈ +0.5° overshoot	15.04%	≈ +0.3° overshoot	5.2%
Yaw	0.08 rad (4.58°)	≈ −0.25° undershoot	7.75%	≈ −0.3° undershoot	6.5%

On roll and pitch, MPC shows approximately 46-65% lower overshoot than PID. The move-suppression weight *R* = 280 in Equation (25) limits aggressive initial torque commands, preventing the large early correction that drives PID overshoot. PID has no equivalent constraint mechanism; its proportional term acts immediately and fully on the initial error, producing a larger transient excursion before the derivative term damps it out.

The overshoot is defined as the degree to which the attitude exceeds the target reference during the transient response phase. This metric is critical for ensuring that the reaction wheels do not saturate and that the satellite remains within the field of view of its sensors.

The yaw overshoot is lowest for both controllers (7.75% PID, 6.5% MPC) because the lower yaw inertia gives the controller greater authority, yielding a faster, more damped response (despite $I_z = 0.00667$ kg·m² being significantly smaller than I_x and I_y, the same gains ($K_p = 0.045$, $K_i = 0.0008$, $K_d = 0.28$) are applied across all axes. The lower inertia means the controller has greater authority over the yaw axis, producing a faster, more damped response with minimal undershoot). The smaller initial yaw error (4.58° vs. 8.59° roll) also contributes less transient energy. further reducing overshoot. Together these effects result in yaw settling faster than roll and pitch as damping increases.

Comparing these results with flight-validated CubeSat data, the PID and MPC steady-state errors (0.04–0.10 mrad) are far below the 3° pointing requirement reported for DISCO-2's imaging mission [7], and the transient excursions (≤ 9°) are within the ±15° linearisation validity range. In-orbit data from UWE-3 and COMPASS-1 missions shows typical nadir pointing accuracies of 3–15° under magnetorquer-only control [16], placing the present reaction-wheel-based results on par with the best-performing CubeSat ADCS configurations reported in the literature. Moreover, on the Roll and Pitch axes, MPC shows lower overshoot than PID because the move-suppression weight $R = 280$ limits aggressive initial torque commands, preventing the large initial correction that drives PID's overshoot. PID has no equivalent constraint mechanism, so its proportional term acts immediately and fully on the initial error, producing a larger transient excursion before the derivative term damps it out.

IX. MPC VS PID COMPARISON

MPC offers clear theoretical advantages for constrained attitude control. It explicitly enforces actuator limits (here ±3.2 mN·m), generates smooth torque profiles that reduce reaction-wheel wear over a 3–5 year mission, and uses prediction to counteract disturbances before large attitude errors develop, which in our simulations translates into lower overshoot, smaller angular-rate excursions, and improved transient attitude response compared with PID. Under the present tuning, MPC achieves faster settling on pitch and yaw while remaining fully stable and well damped.

However, for a 3U CubeSat with the hardware and disturbance regime considered here, these benefits are marginal relative to the implementation cost. The Ziegler–Nichols–tuned PID controller is computationally trivial, has fixed and fully deterministic execution time, requires only a handful of gains to tune, and already meets mission requirements: approximately 80 s three-axis settling time, negligible steady-state error, and no degradation across a decade variation in atmospheric density.

By contrast, MPC requires solving a quadratic programme at every 0.05 s sample, which is demanding for typical CubeSat on-board processors and presumes a high-fidelity plant model to maintain robustness. Although MPC is the preferred option when actuator constraints are tight or disturbances are comparable to available control authority, in this mission the maximum reaction-wheel torque exceeds the peak disturbance torque by more than four orders of magnitude, so constraints are effectively non-binding in nominal operation. Given this large control margin and the already satisfactory PID performance, PID is selected as the baseline attitude controller, with MPC retained as a useful benchmark and a candidate for future missions with stricter actuator or pointing constraints.

X. CONCLUSION

This work has presented a complete simulation-based analysis of three-axis attitude stabilisation for a 3U CubeSat in a 400 km low Earth orbit. A six-state control-oriented attitude model was derived from Euler's rigid-body dynamics, including decoupled single-axis transfer functions and a compact state-space representation under well-justified small-angle and principal-axis assumptions, with inertia parameters consistent with a realistic 3U configuration.

The dominant environmental disturbance torques—atmospheric drag, solar radiation pressure, and residual magnetic dipole—were quantified from first principles using NRLMSISE-00 and IGRF-13, yielding peak torque levels of 0.166 μN·m, 0.885 nN·m, and 0.468 μN·m, respectively, and establishing a transparent, physically grounded disturbance baseline at 400 km. On this model, Ziegler–Nichols-tuned PID controllers achieved three-axis nadir pointing with approximately 80 s settling time, negligible steady-state error, large phase margins (88–90°), and essentially zero sensitivity to a decade-range variation in atmospheric density, supporting their suitability for missions spanning an entire solar cycle.

A constrained per-axis MPC benchmark was then implemented on the same plant and disturbance model. The MPC controller reduced transient overshoot by roughly 46–65% relative to PID and maintained strictly feasible actuator torques (≤ 30 μN·m), in contrast to the ideal PID's initial commands of up to 4.2 N·m, thereby demonstrating the value of explicit constraint handling for reaction-wheel lifetime and actuator protection. Nonetheless, for the 3U CubeSat scenario considered—where reaction-wheel authority exceeds peak disturbance torques by more than four orders of magnitude—the additional computational burden and model-fidelity requirements of MPC offer limited practical advantage over a well-tuned PID design. PID therefore provides the more favourable trade-off between pointing performance, implementation simplicity, and on-board feasibility, while MPC remains an attractive option for future missions constrained by tighter actuator margins or more demanding transient specifications.

Future work will extend the present framework to quaternion-based kinematics for large-angle manoeuvres, incorporate gravity-gradient and higher-order geomagnetic effects into the disturbance model, and investigate hardware-in-the-loop implementation of MPC on representative CubeSat on-board computers. A systematic Monte Carlo study over randomised initial conditions and disturbance parameters is also planned, to further quantify robustness margins and inform controller selection for next-generation CubeSat missions.

➢ *Nomenclature*

Symbol	Description	Unit
A	Cross-sectional area exposed to drag / Continuous-time system matrix	m^2 / —
*A*_d	Discrete-time system matrix (MPC)	—
*A*_srp	Face area exposed to solar radiation	m^2
B, *B*_d	Continuous-time input and disturbance matrices	—
*B*_Earth	Earth geomagnetic field vector	T
C	Output matrix	—
*C*_d	Aerodynamic drag coefficient	—
*C*_r	Solar reflectivity coefficient	—
c	Speed of light	m/s
*d*_cp	Drag centre-of-pressure to CoM offset	m
*d*_cp,s	Solar-pressure centre-of-pressure to CoM offset	m
e(*t*)	Attitude error signal	rad
*F*_d	Atmospheric drag force	N
G(*s*)	Open-loop plant transfer function	rad·s^2/(N·m)
*h*_RW	Stored reaction-wheel angular momentum	N·m·s
I	Inertia tensor	kg·m^2
*I*_x, *I*_y, *I*_z	Principal moments of inertia	kg·m^2
J	MPC cost function	—
*K*_p, *K*_i, *K*_d	PID gains	—
*K*_u	Ziegler–Nichols ultimate gain	—
L(*s*)	Open-loop frequency-response transfer function	—
*m*_res	Residual magnetic dipole moment	A·m^2
*N*_p, *N*_u	MPC prediction and control horizons	steps
*P*_u	Ziegler–Nichols ultimate period	s
Q, *R*	MPC output weight and move-suppression weight	—
T(*s*)	Closed-loop transfer function	—
*T*_orbit	Orbital period	s
*T*_s	Sampling period	s
*t*_s	Settling time	s
u	Control torque vector	N·m
v	Orbital velocity	m/s
x	State vector	—
φ, θ, ψ	Roll, pitch, yaw angles	rad
Φ	Solar irradiance at 1 AU	W/m^2
ρ	Atmospheric density	kg/m^3
τ_ctrl	Reaction-wheel control torque	N·m
τ_d	Total disturbance torque vector	N·m
τ_drag, τ_srp, τ_mag	Axis disturbance components	N·m
τ_max	Peak disturbance amplitude	N·m
τ_max,RW	Maximum reaction-wheel torque (RW0)	N·m
ω	Angular velocity vector	rad/s
ω_c	Gain crossover frequency	rad/s
Δ*u*	MPC control increment	N·m

ABBREVIATIONS

ADCS – Attitude Determination and Control System; CoM – Centre of Mass; GM – Gain Margin; IGRF – International Geomagnetic Reference Field; LEO – Low Earth Orbit; MPC – Model Predictive Control; PID – Proportional-Integral-Derivative; PM – Phase Margin; RK4 – 4th-order Runge–Kutta; RMS – Root Mean Square; RW – Reaction Wheel; SRP – Solar Radiation Pressure; ZN – Ziegler–Nichols; ZOH – Zero-Order Hold.

- Declaration of competing interests: None of the authors have any competing interests including any financial,
- Data availability: Simulation code and data are available from the corresponding author upon reasonable request.

REFERENCES

[1]. Shoaib, M. et al. (2022). "Attitude determination and control system design for micro/nano-spacecraft: A comprehensive survey," *Advances in Space Research*, vol. 70, no. 5, pp. 1143–1165. https://doi.org/10.1016/j.asr.2022.05.022

[2]. Nanosatellite/CubeSat Database (2024). *CubeSats and Nanosatellites: 2024 Statistics, Forecast and Reliability*. Presented at IAC 2024, Milan, Italy. Available: https://www.nanosats.eu

[3]. Shoaib, M. et al. (2022). Survey — as [1]. *(Dual citation retained for Table 1 reference disambiguation.)*

[4]. Wie, B. (2008). *Space Vehicle Dynamics and Control*, 2nd ed. AIAA Education Series, Reston, VA.

[5]. Markley, F. L. and Crassidis, J. L. (2014). *Fundamentals of Spacecraft Attitude Determination and Control*. Springer, New York. https://doi.org/10.1007/978-1-4939-0802-8

[6]. Sidi, M. J. (1997). *Spacecraft Dynamics and Control: A Practical Engineering Approach*. Cambridge University Press, Cambridge.

[7]. Mortensen, J. et al. (2024). "DISCO-2 — an ambitious earth observing student CubeSat for arctic climate research," *Frontiers in Remote Sensing*, vol. 5, art. 1474560. https://doi.org/10.3389/frsen.2024.1474560

[8]. Nakasuka, S. et al. (2022). "Design and environmental testing of imaging payload for a 6U CubeSat at low Earth orbit: KITSUNE mission," *Frontiers in Space Technologies*, vol. 3, art. 1000219. https://doi.org/10.3389/frspt.2022.1000219

[9]. Munusamy, A. et al. (2023). "Design and analysis of a nanosatellite attitude control system using PID transfer function," *AEÜ — International Journal of Electronics and Communications*, vol. 171, art. 154956. https://doi.org/10.1016/j.aeue.2023.154956

[10]. Sindi, H. et al. (2023). "Three-axis attitude stabilisation of a nano-CubeSat via ANFIS-PD controller," *Heliyon*, vol. 9, no. 2, art. e13248. https://doi.org/10.1016/j.heliyon.2023.e13248

[11]. Rocha, M. et al. (2024). "Sliding mode attitude control for a 3U CubeSat in LEO," *AIJR Proceedings*, ICARES 2024. https://doi.org/10.21467/proceedings.160

[12]. Mammarella, M. et al. (2022). "Tube-based robust model predictive control for small spacecraft attitude stabilisation," *Journal of Spacecraft and Rockets*, vol. 59, no. 3, pp. 852–865. https://doi.org/10.2514/1.A34394

[13]. Mammarella, M. et al. (2022). *(See [12]; dual citation used in the text for disturbance-model and constraint results separately.)*

[14]. Zhang, Y. and Hu, Q. (2024). "Linear time-varying fractional-order model predictive attitude control for dual-reaction-wheel satellite," *Aerospace Science and Technology*, vol. 145, art. 108874. https://doi.org/10.1016/j.ast.2024.108874

[15]. Decentralised MPC Formation Flight (ICAS 2024). "A model predictive control algorithm for the formation flight control of CubeSat-type nanosatellites in LEO," *ICAS Congress 2024*, Paper ICAS2024-0494.

[16]. Alminde, L. et al. (MIST assessment). "Attitude Determination and Control of the CubeSat MIST," KTH Royal Institute of Technology Technical Report. Available: https://s3vi.ndc.nasa.gov/ssri-kb/static/resources/FULLTEXT01 (3).pdf

[17]. Ruf, O. et al. (2023). "Advanced test environment for automated attitude control testing of fully integrated CubeSats on system level," *CEAS Space Journal*, vol. 16, pp. 203–224. https://doi.org/10.1007/s12567-023-00523-x

[18]. NanoAvionics (2021). *RW0 Reaction Wheel Assembly Datasheet*. NanoAvionics, Vilnius, Lithuania. Available: https://nanoavionics.com/cubesat-components/cubesat-reaction-wheels-rw0/

[19]. Pilinski, M. D. (2023). "Hyperthermal gas–surface interaction and drag coefficient modelling for LEO spacecraft," *Frontiers in Chemistry*, vol. 11, art. 1123836. https://doi.org/10.3389/fchem.2023.1123836

[20]. Iakubivskyi, I. et al. (2023). "Magnetic cleanliness and residual dipole characterisation for 3U CubeSats with electric propulsion," *Acta Astronautica*, vol. 207, pp. 220–231. https://doi.org/10.1016/j.actaastro.2023.03.008