Journal of Advanced College of Engineering and Management

Satellite Attitude Control using Reinforcement Learning and State Space Model

[1]Aryan Kafle, [2]Ajita Kharel, [3]Alisha Shah, [4]Aagya Khati Chhetri, [5]Damodar Pokhrel
[1,2,3,4]*Department of Electronics and Computer Engineering, Advanced College of Engineering and Management, Kathmandu, Nepal*
[5]*Nepal Academy of Science and Technology, Lalitpur, Nepal*
Email: aryan.078bct014@acem.edu.np

DOI: 10.3126/jacem.v12i01.93930

Abstract

This paper presents a reinforcement learning (RL) based framework for satellite attitude control using a state-space model. The Proximal Policy Optimization (PPO) algorithm is used to train an agent for three-axis satellite reorientation in a simulation environment governed by Euler's rotational equations and quaternion kinematics. Real-world Inertial Measurement Unit (IMU) data collected from Micro-Electro-Mechanical Systems (MEMS) Accelerometer and Gyro sensors was used to characterize noise parameters and validate simulation fidelity. The trained PPO agent was evaluated against an untrained baseline and a cascade Proportional-Integral-Derivative (PID) controller over 500 randomized episodes. The trained RL agent achieved a 96% success rate with 18.3° mean pointing error and 98.8% alignment score, closely competitive with PID which achieved 100% success rate, 26.0° mean pointing error, and 99.8% alignment score, while using comparable control effort 276 Rate Per Minute (RPM) for RL agent vs 287 RPM for PID. A 3D interactive visualization system was developed for real-time trajectory inspection. Results confirm the feasibility of RL-based attitude control and identify clear directions for improvement.

Keywords*—Reinforcement Learning, Satellite Attitude Control, Proximal Policy Optimization, Quaternion Kinematics, IMU Sensor Fusion, Madgwick Filter, Mahony Filter, State Space Model*

1. INTRODUCTION

Satellite attitude control is a critical component of space mission engineering, determining the precise orientation of a satellite in orbit to satisfy mission requirements including Earth observation, communication link, and data acquisition. Any deviation from the desired orientation directly affects payload alignment, solar panel efficiency, and antenna pointing, all of which can impact mission success.

Traditionally, attitude control has relied on model-based methods such as Proportional-Integral-Derivative (PID) controllers. While effective when accurate system models are available, these controllers degrade in performance as satellites undergo mass property changes during missions, such as fuel depletion, thermal expansion, and

micro-vibrations. In highly nonlinear and uncertain environments, model-based controllers lack the adaptability needed for continued reliable performance.

Reinforcement Learning (RL) has emerged as a promising model-free alternative. RL agents learn optimal control policies directly from environment interaction without requiring an explicit analytical model, making them inherently adaptable to changing dynamics. The Proximal Policy Optimization (PPO) algorithm has demonstrated strong performance on continuous control tasks and is well suited for satellite attitude applications [3].

This paper presents the design, implementation, and evaluation of a PPO-based RL controller for three-axis satellite attitude control. The system is grounded in real-world IMU data from MEMS sensors and is complemented by a 3D interactive visualization module for real-time trajectory inspection.

The main contributions are: **(1)** a practical data-driven PPO framework trained with real-world IMU noise parameters; **(2)** a comparative evaluation of Madgwick and Mahony filters, Attitude and Heading Reference System (AHRS) Systems; **(3)** a rigorous statistical comparison of trained PPO, untrained PPO, and cascade PID over 500 randomized episodes; and **(4)** a 3D interactive visualization module for diagnostic trajectory inspection.

2. LITERATURE REVIEW

This section reviews prior work relevant to satellite attitude control, reinforcement learning-based control, model-based approaches, and sensor fusion techniques, as these form the theoretical and practical foundations of the proposed system.

Cai et al. [1] addressed multi-constraint satellite formation attitude control using a phased priority RL (PPRL) strategy with Deep Deterministic Policy Gradient (DDPG). DDPG is an off-policy actor-critic algorithm designed for continuous action spaces, combining deterministic policy gradients with experience replay and target networks for stability. Their results showed RL-based controllers outperform traditional methods in adaptability and constraint satisfaction in dynamic multi-satellite scenarios, directly motivating the use of RL in this work.

Esit et al. [2] investigated Model Predictive Control (MPC) for three-axis attitude control of small satellites using only magnetorquers. MPC is an optimization-based control technique that solves a finite-horizon optimal control problem at each time step using a predictive model of the system. Their system achieved attitude accuracy within 10 degrees over approximately 10 orbits from a tumbling initial state, but highlighted the fundamental limitation of model-based controllers in handling unexpected disturbances without frequent model updates.

Elkins et al. [3] developed and tested a deep RL framework using PPO for autonomous spacecraft attitude control, targeting time-optimal high-accuracy maneuvers without

detailed system models. PPO is a policy gradient algorithm that constrains policy updates via a clipped surrogate objective, ensuring stable and sample-efficient learning in continuous control domains. Training with randomized initial conditions, their agent achieved sub-millidegree pointing accuracy across thousands of test episodes, directly validating PPO as a viable algorithm for satellite attitude tasks.

Peterson [4] derived exact analytical solutions to Euler's equations for rigid-body motion with application to satellite detumbling using Jacobi elliptic functions. These solutions provide rigorous mathematical benchmarks for evaluating simulation environments and learning-based controllers under known dynamic conditions.

El Hariry et al. [5] investigated custom PPO policies for fully actuated and underactuated satellite scenarios involving reaction wheel failures, demonstrating successful sim-to-real policy transfer and robustness to actuator degradation, supporting the practical viability of RL-based controllers in fault-tolerant operations.

Narkiewicz et al. [6] proposed a generic model of a satellite attitude control system covering all principal functional blocks, providing a comprehensive framework that facilitates the development and verification of new control strategies across a range of mission profiles.

Gao et al. [7] applied deep RL to satellite attitude control and demonstrated competitive performance relative to classical controllers, further validating the RL-based approach for onboard autonomous control.

Regarding sensor fusion and orientation estimation, the Madgwick filter [8] applies gradient descent optimization on a quaternion-based orientation representation to minimize the difference between measured and predicted accelerometer and magnetometer readings, offering computationally efficient AHRS suitable for embedded systems. The Mahony filter [9] employs a proportional-integral (PI) feedback mechanism to correct gyroscope integration drift using accelerometer and magnetometer measurements, providing robust orientation estimates even at low computational budgets. Both filters are widely employed in MEMS-based IMU systems for real-time orientation estimation.

Regarding IMU sensor characterization, Aggarwal et al. [10] provide a comprehensive treatment of MEMS-based IMU error modeling and noise characterization, including methods for estimating sensor bias, random walk, and scale factor errors from static measurements — directly applicable to the noise parameter extraction methodology used in this work.

Despite these advances, a clear gap remains in the availability of accessible, real-world-inspired datasets for training and validation. Most prior works rely on fully simulated or proprietary environments. This work addresses that gap using consumer-grade IMU sensors embedded in a smartphone, making the approach reproducible and accessible to researchers without specialized hardware.

3. METHODOLOGY

A. System Overview

The proposed system consists of four integrated stages: (1) real-world sensor data collection and preprocessing from a IMU sensor; (2) state-space model identification and Gymnasium simulation environment construction; (3) PPO-based RL agent training using Stable-Baselines3; and (4) performance evaluation with 3D visualization. The simulation environment encodes rigid-body rotational dynamics and serves as the training ground for the RL agent, while the real-world IMU data configures realistic noise parameters and validates the dynamics model.

B. Dataset Collection and Processing

Real-world motion data was collected via the built-in IMU sensors, capturing 3-axis accelerometer, gyroscope, and magnetometer readings at approximately 120 Hz in CSV format. Six motion profiles were recorded: inclined, declined, circular, linear, stable, and north-aligned orientations, each in 10-minute sessions to ensure diverse dynamic coverage.

Preprocessing involved standardizing column headers, interpolating missing values, detecting outliers via statistical thresholds, and synchronizing timestamps. Sensor noise standard deviation and bias per axis were estimated from the first 30 seconds of static recording. The extracted noise parameters — including accelerometer and gyroscope bias and standard deviation — were subsequently used to configure simulation-level noise injection, following established MEMS IMU characterization methodologies [10]. Extracted maximum angular velocities and accelerations were also used to scale simulation parameters, ensuring the training environment reflects realistic sensor behavior.

C. Sensor Fusion and Orientation Estimation

Two complementary AHRS methods were implemented for orientation estimation from the collected IMU data.

The Mahony filter [9] is a nonlinear complementary filter that estimates orientation by minimizing the error between body-frame sensor measurement vectors (from accelerometers and magnetometers) and navigation-frame reference vectors. Correction is applied via proportional-integral (PI) feedback, where the proportional gain (K□ = 0.5) controls the speed of error correction and the integral gain ($K_i = 0.0$) compensates for gyroscope bias drift. The filter fuses gyroscope, accelerometer, and magnetometer data to produce a quaternion-based orientation estimate at low computational cost, making it suitable for real-time embedded applications.

The Madgwick filter [8] applies gradient descent optimization on the quaternion orientation representation. At each time step, the algorithm minimizes the objective function representing the difference between the measured accelerometer and

magnetometer vectors and those predicted by the current orientation estimate. The filter gain $\beta = 0.1$ controls the convergence rate of the gradient descent step, trading off between gyroscope noise and accelerometer/magnetometer noise. A lower β relies more heavily on gyroscope integration and is less susceptible to accelerometer disturbances, while a higher β provides faster correction at the cost of increased noise sensitivity.

Both filters were validated using three standardized tests: (1) static stability under zero-motion input; (2) known rotation of 0.1 rad/s about the Z-axis for 10 s producing 1.0 rad total within 0.01 rad tolerance; (3) gravity correction recovering a 30° tilt error to within 5° in 20 s using only accelerometer feedback. Both filters passed all three tests, confirming correct implementation.

D. Satellite Dynamics Model

The simulation environment models satellite rotational dynamics using Euler's equations for a rigid body. With principal moments of inertia I_1, I_2, $I_3 = 1.0$ kg·m², angular velocities ω_1, ω_2, ω_3, control torques τ_1, τ_2, τ_3, and damping coefficient $d = 0.001$:

$$I_1(d\omega_1/dt) = (I_2 - I_3)\omega_2\omega_3 + \tau_1 - d\cdot\omega_1 \quad (1)$$

$$I_2(d\omega_2/dt) = (I_3 - I_1)\omega_3\omega_1 + \tau_2 - d\cdot\omega_2 \quad (2)$$

$$I_3(d\omega_3/dt) = (I_1 - I_2)\omega_1\omega_2 + \tau_3 - d\cdot\omega_3 \quad (3)$$

Satellite orientation is propagated using quaternion kinematics. Let $q = [q_0, q_1, q_2, q_3]^T$ be the unit quaternion:

$$dq/dt = \tfrac{1}{2} \cdot q \otimes \omega\_q \quad (4)$$

where $\omega\_q = [0, \omega_x, \omega_\gamma, \omega\_z]^T$ is the angular velocity as a pure quaternion. Euler integration at $\Delta t = 0.05$ s is applied followed by renormalization, avoiding the gimbal lock singularities inherent to Euler angle representations.

E. Reinforcement Learning Framework

The RL framework employs PPO implemented via Stable-Baselines3. The Actor network maps the 12-dimensional state vector $s = [q, \omega, q\_target, \theta\_error] \in \mathbb{R}^{12}$ to a 3-dimensional continuous torque command through two hidden layers of 256 neurons with Tanh activation. The Critic shares the same architecture and outputs a scalar value estimate. Key hyperparameters: learning rate 1×10^{-4}, batch size 64, epochs per update 20, Generalized Advantage Estimation (GAE) $\lambda = 0.95$, discount $\gamma = 0.99$, clip range 0.2, entropy coefficient 0.01.

The action vector $a \in [-1, 1]^3$ represents normalized torques scaled by $\tau\_max = 1.0$ N·m. Angular error between current and target orientations is computed via quaternion algebra:

$$q\_error = q\_target^{-1} \otimes q\_current \quad (5)$$

$$\theta\_error = 2 \cdot \cos^{-1}(|q\_error,w|) \quad (6)$$

where |q_error,w| is the absolute scalar component ensuring shortest-path rotation selection. The composite reward function is:

$$R = \exp(-5.5 \cdot \theta\_error) - 0.001\|\omega\| - 0.001\|a\|^2 - 0.005\|a - a\_prev\| \quad (7)$$

Terminal rewards of +10.0 are given when $\theta\_error < 1^\circ$ and $\|\omega\| < 0.01$ rad/s (success), and −10.0 when $\theta\_error > 120^\circ$ (failure). Training used 4 parallel environments with VecNormalize observation and reward scaling, running for 5 million timesteps.

F. Validation

The angular error formula was validated with four analytical tests: identity test (zero error for identical quaternions), 90º rotation test, 180º rotation test, and shortest-path verification where a 270º command correctly resolved to 90º error. All four tests passed within 10^{-6} radians tolerance. Controller evaluation used 500 randomized episodes per controller with uniformly sampled initial orientations, angular velocities within ±0.1 rad/s, and random target orientations. Episodes were capped at 500 steps (25s) and success was defined as final angular error below 5º.

4. RESULTS AND DISCUSSION

A. Training Convergence

The PPO agent was trained for 5,000,000 timesteps across 4 parallel environments. Figure 1 shows the comprehensive training and evaluation dashboard. The training progress curve (bottom-left panel) shows validation error steadily decreasing from approximately 70º at the start to a stable mean of 14.4º beyond 500k steps, with a dashed red line marking this convergence threshold. Control effort reduced dramatically from 1441 RPM for the untrained baseline to 276 RPM for the trained agent, comparable to the PID's 287 RPM, demonstrating that training also produced an energy-efficient policy.

Controller	Mean Error	Alignment	Success	Effort
Untrained RL	88.9°	46.1%	0.0%	1441 RPM
Trained RL	18.3°	98.8%	96.0%	276 RPM
PID	26.0°	99.8%	100.0%	287 RPM

Figure 1: Comprehensive 9-panel evaluation dashboard.

B. Controller Performance Comparison

Three controllers were evaluated over 500 randomized episodes each. Table I summarizes aggregate performance metrics extracted from the evaluation results. The trained PPO agent achieved a 96% success rate with 18.3º mean pointing error and a 98.8% alignment score. The cascade PID achieved 100% success rate, 26.0º mean pointing error, and 99.8% alignment score. Notably, the trained RL agent outperformed PID in mean pointing error and nearly matched it in alignment score, while using comparable control effort (276 vs 287 RPM).

Table I: Aggregate Controller Performance over 500 Episodes

Metric	Untrained PPO	Trained PPO	Cascade PID
Mean Pointing Error (º)	88.9	18.3	26.0
Alignment Score (%)	46.1	98.8	99.8
Success Rate (< 5º)	0.0%	96.0%	100.0%
Avg. Control Effort (RPM)	1441	276	287
Settling Time to 1º (s)	24	~21	~6
Mean Episode Reward	~0	~280	~95

Figure 2 shows angular error trajectories for a representative single episode alongside final error and reward bar charts. The cascade PID (cyan) achieves smooth convergence from 73º to near-zero within 100 steps. The trained PPO (brown) rapidly reduces error from 150º to approximately 20º in the first 50 steps but exhibits residual oscillations. The untrained PPO (blue) shows no systematic convergence, oscillating between 125 - 180º. The final error bar chart confirms untrained PPO ends at approximately 120º while both trained PPO and PID reach near-zero. Despite the trained PPO achieving higher total reward ~385 vs PID ~130, PID achieves superior final precision.

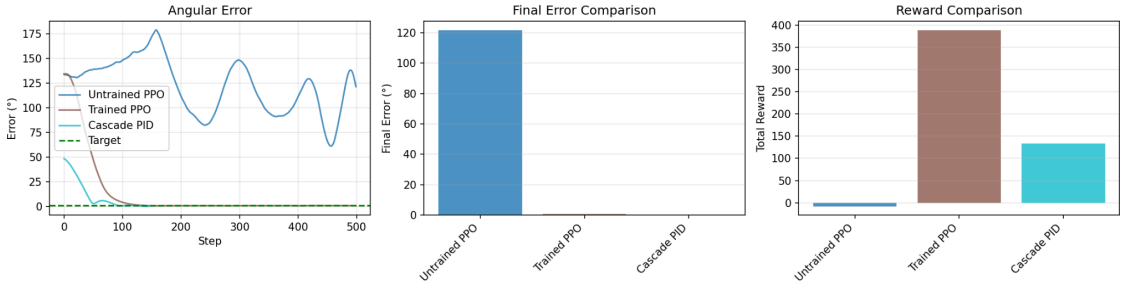

Figure 2: Single-episode controller comparison

Figure 3 presents performance distributions across all 500 episodes. The final error distribution shows the untrained PPO with median ~95º and wide range (8 - 175º); the trained PPO with median ~3º and a tight interquartile range with a few outliers up to 25º; and the cascade PID with median ~0.5º and very tight range (0 - 5º). The reward distribution shows trained PPO achieving median ~340 with larger spread (150 - 500+), while PID is tightly distributed around median ~125 (range 80 - 130). These distributions confirm the trained RL agent is consistent but with higher episode-to-episode variability than PID.

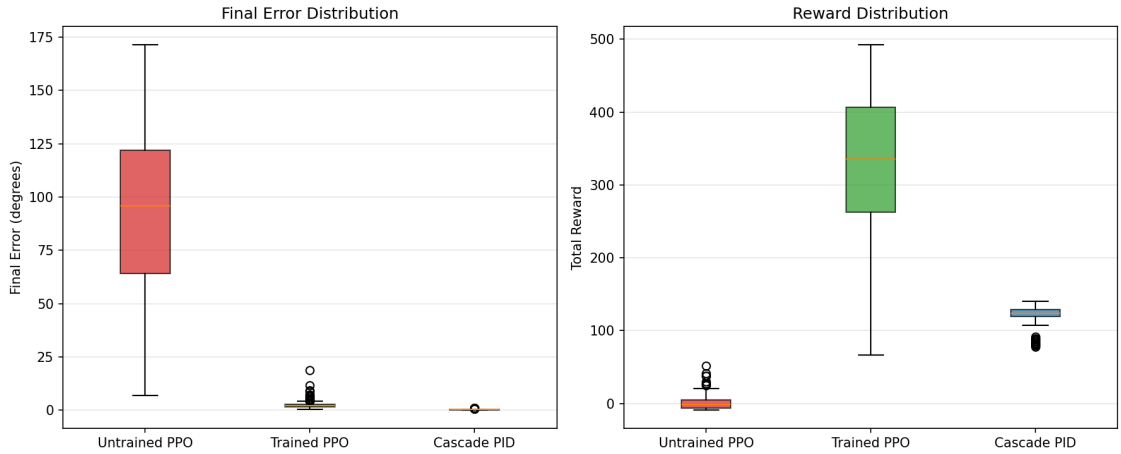

Figure 3: Performance distributions across 500 evaluation episodes

Figure 4 shows a detailed single-episode comparison between PID and trained RL. Both controllers start from a 170º initial error and both successfully converge (marked Success: Yes). PID achieves final error 0.3º in 207 steps with peak angular velocity 0.68 rad/s decaying smoothly. The RL agent achieves final error 0.5º but uses the full 500 steps, with peak angular velocity 1.17 rad/s and sustained low-level residual actuation. The RL agent's total reward (336.5) is substantially higher than PID's (115.0) despite the longer episode, while mean error over the episode is lower for RL.

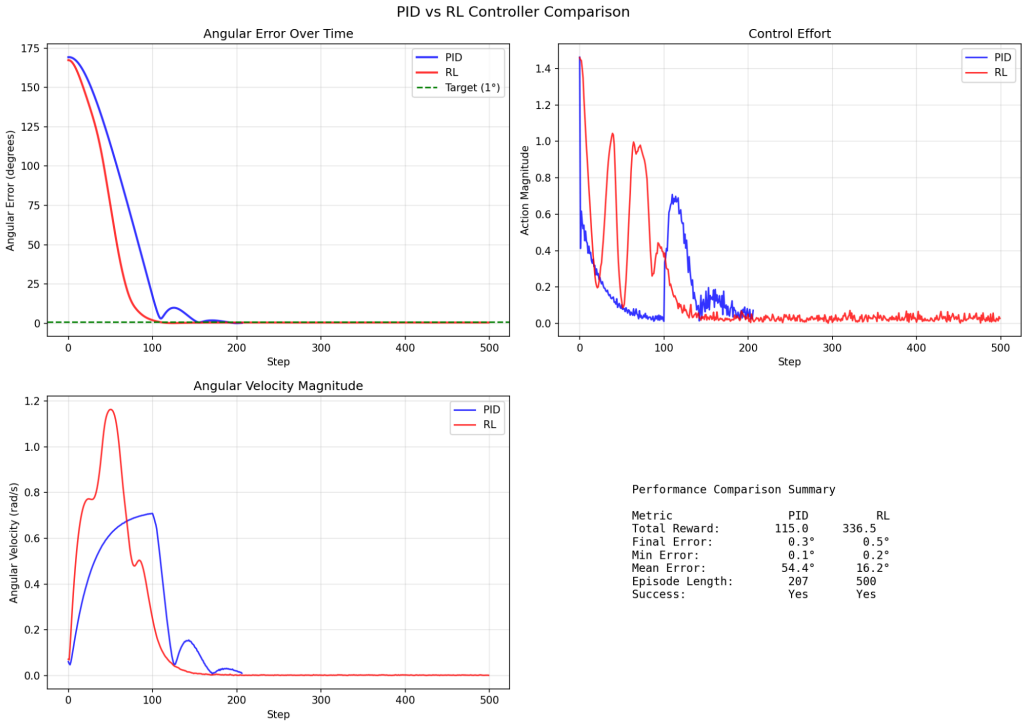

Figure 4: Detailed single-episode PID vs. trained RL comparison

Figure 5 shows the cascade PID final state 3D visualization and attitude error convergence curve. The 3D render confirms the satellite body axes (X=red, Y=green, Z=blue pointing axis) converging to the target orientation (orange). The attitude error curve shows convergence from approximately 45º, crossing below the 1º target threshold (dashed green line) at around step 90 with a minor overshoot near step 60 (~6º), before finally settling near 0º by step 140.

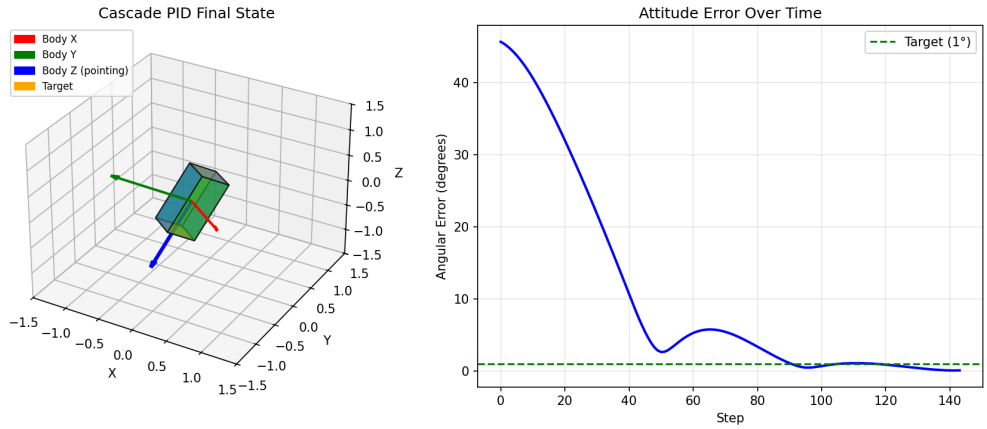

Figure 5: Cascade PID controller final state visualization.

C. Analysis and Discussion

The results reveal a nuanced picture. While cascade PID achieves 100% success rate and tighter final precision in aggregate, the trained PPO agent is closely competitive: 96% success rate, lower mean pointing error (18.3º vs 26.0º), near-equal alignment score (98.8% vs 99.8%), and comparable control effort 276 RPM for RL agent vs 287 RPM for PID. This is a strong result for a model-free agent trained entirely through simulation.

The primary remaining weakness is settling time. PID reaches below 1º in approximately 6 s while trained PPO requires approximately 21 s. This is reflected in episode length - RL consistently uses the full 500 steps. The persistent low-level oscillation is attributed to insufficient reward penalty weights on action jerk (0.005) and magnitude (0.001), allowing the agent to accumulate alignment reward through approximate convergence rather than tight settling.

The higher cumulative reward achieved by RL (~280 vs ~95 for PID) reflects the reward function rewarding sustained near-target behavior over 500 steps, while PID terminates earlier after converging precisely. This reward-accuracy mismatch is a known challenge in reward engineering for precision control and is the primary direction for future improvement.

5. CONCLUSION

This paper presented a reinforcement learning-based framework for satellite attitude control using a state-space model of rigid-body rotational dynamics. The system integrated real-world IMU data from smartphone sensors, Madgwick and Mahony orientation filters, a PPO-based RL agent trained using Stable-Baselines3, and a 3D interactive visualization module.

The trained PPO agent achieved a 96% success rate with 18.3º mean pointing error and 98.8% alignment score across 500 randomized episodes, closely competitive with the cascade PID controller (100% success, 26.0º mean error, 99.8% alignment score, 287 RPM). Control effort was comparable (276 RPM for RL vs 287 RPM for PID). The primary gap is settling time: PID converges to below 1º in ~6 s compared to ~21 s for the trained RL agent, due to residual oscillation near the target.

Future work will focus on: (1) increasing penalty weights on action jerk and magnitude to eliminate residual oscillation; (2) extending training beyond 5 million timesteps toward the 20–50 million timestep range reported for comparable spacecraft control tasks [3][5]; (3) evaluating both controllers under sensor noise and model uncertainty to expose the adaptability advantages of RL; and (4) incorporating curriculum learning to guide early training toward faster convergence.

This work demonstrates that a practical RL-based attitude control framework built on consumer-grade smartphone IMU data can achieve performance competitive with classical PID control, providing a reproducible and accessible foundation for future adaptive satellite control systems.

ACKNOWLEDGMENTS

The authors express sincere gratitude to Er. Ramesh Sharma (Academic Project Coordinator), Er. Roshani Ghimire (Head of Department), and Er. Navaraj Banstola and Dr. Rakshya Dangol (Deputy Heads of Department) of Electronics and Computer Engineering, Advanced College of Engineering and Management, for their invaluable guidance, support, and encouragement throughout this project.

SUGGESTIONS AND RECOMMENDATION

Extended training is strongly recommended. The current 5 million timestep budget is comparatively modest for continuous precision control tasks. Literature on comparable spacecraft control problems reports effective policies emerging between 20 and 50 million timesteps [3][5].

REFERENCES

[1] Y. Cai, K.-S. Low, and Z. Wang, "Reinforcement Learning-Based Satellite Formation Attitude Control Under Multi-Constraint," Advances in Space Research, 2024.

[2] M. Esit, H. E. Soken, and C. Hajiyev, "A Model Predictive Control Based Magnetorquer-only Attitude Control Approach for a Small Satellite," IFAC PapersOnLine, 2023.

[3] J. G. Elkins, R. Sood, and C. Rumpf, "Autonomous Spacecraft Attitude Control Using Deep Reinforcement Learning," 71st International Astronautical Congress (IAC), 2020.

[4] C. Peterson, "Exact solutions to Euler's equations for rigid body motion with application to detumbling satellites," arXiv:2301.10220, 2022.

[5] M. El Hariry, A. Cini, G. Mellone, and A. Balossino, "Deep Reinforcement Learning Policies for Underactuated Satellite Attitude Control," arXiv:2505.00165, 2025.

[6] J. Narkiewicz, M. Sochacki, and B. Zakrzewski, "Generic model of a satellite attitude control system," International Journal of Aerospace Engineering, vol. 2020, 2020.

[7] D. Gao, X. Gao, H. Zhang, and C. Li, "Satellite Attitude Control with Deep Reinforcement Learning," 2020 Chinese Automation Congress (CAC), pp. 4095–4102, 2020.

[8] S. O. H. Madgwick, A. J. L. Harrison, and R. Vaidyanathan, "Estimation of IMU and MARG orientation using a gradient descent algorithm," 2011 IEEE International Conference on Rehabilitation Robotics (ICORR), pp. 1–7, 2011.

[9] R. Mahony, T. Hamel, and J.-M. Pflimlin, "Nonlinear complementary filters on the special orthogonal group," IEEE Transactions on Automatic Control, vol. 53, no. 5, pp. 1203–1218, 2008.

[10] P. Aggarwal, Z. Syed, X. Niu, and N. El-Sheimy, "A Standard Testing and Calibration Procedure for Low Cost MEMS Inertial Sensors and Units," Journal of Navigation, vol. 61, no. 2, pp. 323–336, 2008. doi: 10.1117/12.2296568