Parameter Estimation and Control of an Automatic Balancing System for CubeSat Research and Applications

Alex McCafferty-Leroux[a], Andrew Newton[a], and S. Andrew Gadsden[a]

[a]McMaster University, Hamilton, Canada

ABSTRACT

Deployed for purposes of GPS, defense, atmospheric and space research, environmental monitoring, broadcasting, and communication, Earth observation satellites are complex systems that require the design of highly reliable control and estimation algorithms. A satellite's attitude determination and control system (ADCS) must be able to operate accurately, in a robust manner against unexpected conditions, especially in missions that demand more intricate tasks. The desire for optimal and robust performance in satellites has been the driving factor behind decades of attitude control research. With computers, the performance of spacecraft subject to some mission can be simulated to test new control methods, but the availability of real satellites to researchers for testing these algorithms is very limited. To solve this issue, attitude control simulators have been developed, such that algorithms and hardware can be tested inexpensively in a lab environment, while maintaining a high level of accuracy to the environment it emulates. The Nanosatellite Attitude Control Simulator (NACS) has been developed at McMaster University for this purpose. Consisting of a mock 1U CubeSat, an air-bearing configuration, and an automatic balancing system (ABS), rotational attitude control experiments are conducted in-lab without deployment, simulating the zero-gravity of space. The mechanism responsible for environment simulation is the ABS, which minimizes residual torque due to gravity by influencing the center of mass (CoM) of the system, thereby improving control performance and efficiency. The performance of the ABS in a balancing task is presented, where system parameters of inertia and CoM are estimated from response data. Three filtering strategies are investigated for this purpose, providing varying degrees of accuracy and computational cost.

Keywords: Attitude control, satellite, nanosatellite simulator, control theory, estimation theory, dynamics

1. INTRODUCTION

The prevalence of satellite technologies in the modern age is undeniable and growing. According to the European Space Agency's (ESA) annual space environment report, there are approximately 11 500 satellites still in space, where almost 50% were launched in the last 5 years.[1] Responsible for numerous Earth observation tasks, such as GPS tracking, defense, communication and broadcasting, and environmental monitoring, satellites have demonstrated themselves to be essential in every day life. For instance, Starlink aims to launch hundreds of satellites this decade to reduce broadband service costs with their large constellations. Satellites have additional utility in exploration beyond Earth's atmosphere, as in the Moon,[2] Mars,[3] and heliocentric[4] missions involved in observing, analyzing, and exploring the solar system. Considering these applications, satellites typically focus on a single or multiple targets, and as such must have very accurate pointing capabilities. The onboard subsystem most responsible for this functionality is the attitude control and determination system (ADCS), which unifies control and estimation algorithms for precise sensing and actuation. The body of literature in control and estimation strategy development is considerably large and growing, with a portion of it dedicated to autonomous spacecraft.

Generally, there are many internal or external disturbances that can result in strain or even failure towards a satellite's ADCS, having monetary or data misplacement consequences. Object collision, environmental interferences (e.g., EMI, charged particles), or mission complexity can be contributing factors to these failures. To counteract the harsh space environment or handle complex tasks, advanced ADCS strategies, decision making,

Further author information: (Send correspondence to A. McCafferty-Leroux or S. A. Gadsden)
A.M.L.: E-mail: mccaffea@mcmaster.ca
S.A.G.: E-mail: gadsden@mcmaster.ca

Sensors and Systems for Space Applications XVII, edited by Genshe Chen,
Khanh D. Pham, Proc. of SPIE Vol. 13062, 130620M · © 2024 SPIE
0277-786X · doi: 10.1117/12.3013732

and precision equipment is often required. In designing such safeguards, engineers employ simulation software to validate key components such has structures, control laws, and sensors. The degree of complexity can range from Matlab dynamical simulations and finite element analysis (FEA) to state of the art simulators. However, deployment environments are unpredictable and hardware, algorithms, etc., do not always function as planned. A recent example is from India's Small Satellite Launch Vehicle (SSLV), where satellite deployment resulted in failure due to unplanned vibrations in orbit that saturated the rocket's inertial measurement unit (IMU).[5] Considering the cost and unservicability of satellites, it is most beneficial to validate newly proposed methods in physical experiments as well as simulations. In-lab testing provides an inexpensive and safe alternative to untested deployment, being able to diagnose and fix problems in a controlled environment. This aspect is especially crucial in missions where maximum autonomy is desired and machine learning methods are to be implemented, such as in new endeavors from MDA[6] and the ESA.[7]

In recent years, there has been an academic focus on the development of attitude control simulators (ACS) for small satellites. This is most likely the result of the establishment of the CubeSat, making satellite research more accessible to small labs and universities. CubeSats are a modular standard for fabricating economical and easily deployable satellites,[8] and ACSs are appealing for testing these, able to be fabricated with similar off-the-shelf components. In this work, our primary focus is rotational air-bearing simulators. While air-bearing ACSs have become popular in the last 20 years, they were first used by space agencies, with the earliest known example coming out of the NASA Ames Research Center.[9]

R. da Silva *et al.* discusses these simulators[10] in a comprehensive manner, reviewing modeling, filtering/identification, and active control methods applied in these simulators. Hardware arrangements for balancing and their associated performance is also evaluated.[10] Schwartz and Hall alternatively focus on the comparison of the parameter identification strategies in ACS.[11] Although da Silva focuses on air-bearing ACSs, air tables are also utilized for docking/capturing maneuver experiments.[12] There are also simulators that combine the two methods, making a five or six degree-of-freedom (DoF) ACS.[13–15] Where few simulators use manually tuned masses[16] or control moment gyros,[17] the majority of rotational air-bearing ACSs achieve balance in three DoF through automated movable mass units (MMUs). The xy plane is first balanced through some control algorithm, then parameter identification is applied to tune the z mass through the dynamic response of the system. The Naval Postgraduate School ACS[18–20] applies the batch least squares (BLS) method outlined by Young[21] and nonlinear Kalman filters for this purpose. At the Space Research Laboratory (SRL), Sharifi and Zabihian[22] compare three separate filtering strategies in their dumbbell-shaped simulator, achieving the highest accuracy with Levenberg-Marquardt filtering. Bahu and Modenini out of the University of Bologna present a novel planar balancing control law, utilizing BLS for system identification and balancing.[23] Researchers from the University of Brazil,[24, 25] the Brazil National Institute for Space Research,[26] and the University of Science and Technology of Hanoi[27] also investigate this class of ACS for nanosatellite research purposes.

In this paper, we aim to design an air-bearing-based ACS for in-lab control, estimation, and machine learning research. We take a relatively unique approach to this problem,[28] making the balancing mechanism and the mock satellite unit distinct from one another. To emulate a space environment as accurately as possible and minimize passive forces due to gravity, the balancing process is automated. In achieving accurate balancing, the validity of proposed methods are enhanced. Utilizing mass control and recursive system identification methods, the system can be adaptively balanced such that a variety of nanosatellite configurations can be theoretically tested without major system modifications. For example, varying sizes of satellites, or satellites that use different actuation or sensing techniques can be fixed to the system and evaluated. Additionally, the system identification will enable further understanding of the system, resulting in more accurate modeling. The applied method of center of mass tuning is through translatable masses.

The remainder of the publication develops the system model and the mathematics of its function. Section 2 derives the ABS dynamics and Section 3 the proposed control and parameter estimation strategies. Section 4 defines the parameters of the experiment and system. Section 5 discusses the experimental results and their implications, and Section 6 concludes the analysis, providing a summary of results and proposals for future research avenues.

2. ABS MODEL DERIVATION

The kinematic and dynamic relationships are established for the ABS in the following section. The system is modeled as a rigid body capable of movement in three rotational DoF, in terms of a body frame B and the inertial frame I, represented in Figure 1. The frame I is fixed in North East Down (NED) configuration at the center of rotation (CoR) of the air-bearing and the frame B is centered at the system center of mass (CoM). The kinematic equations consider the quaternion for attitude and evolve based on the angular velocity of the system, The ABS dynamics are derived considering angular momentum $\mathbf{H}(t)$ and the system response due to the translatable mass (or MMU) actuators, providing some control input $\mathbf{u}(t)$. Vectors are time-varying, t for $t \geq 0$.

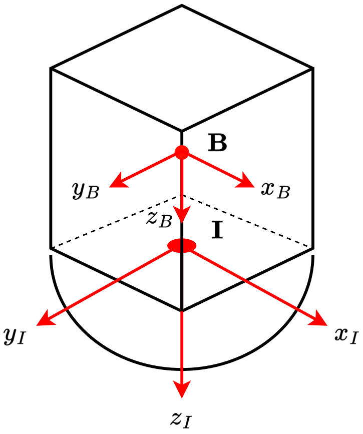


Figure 1: NACS Coordinate System

2.1 Kinematics

The ABS attitude is defined by the quaternion $\mathbf{q}(t) \in \mathbb{R}^4$, making the attitude expressible without trigonometric functions or singularities. The rate of change equation for the quaternion is presented as Equation 1, where the adopted convention expresses its identity as $\mathbf{q}_I = [0\ 0\ 0\ 1]^T$.[29]

$$\dot{\mathbf{q}}(t) = \frac{1}{2}\ \left[\omega_B^{\mathrm{BI}} \otimes\right] \mathbf{q}(t) \tag{1}$$

Where $\omega(t) \in \mathbb{R}^3$ is the angular velocity of the system about B with respect to I. The skew symmetric matrices required for preserving the quaternion norm are also defined in Equations 2 and 3.

$$\left[\omega_B^{\mathrm{BI}} \otimes\right] = \begin{bmatrix} -\left[\omega \times\right] & \omega(t) \\ -\omega(t)^T & 0 \end{bmatrix} \tag{2}$$

$$[\omega \times] = \begin{bmatrix} 0 & -\omega_z(t) & \omega_y(t) \\ \omega_z(t) & 0 & -\omega_x(t) \\ -\omega_y(t) & \omega_x(t) & 0 \end{bmatrix} \tag{3}$$

2.2 Dynamics

The dynamics if the rigid body due to applied and inertial forces are formulated considering the net angular momentum $\mathbf{H}(t) \in \mathbb{R}^3$, as applied by others.[19, 23] The angular velocity $\omega(t)$ dynamics are obtained from this relationship. The net momenta around the CoM c is most generally parameterized by Equation 4. Its derivative is then presented as Equation 5, with respect to both the angular velocity and the angular momentum.

$$\mathbf{H}^{c}_{B}(t) = J^{c}_{B}(t)\omega^{\mathrm{BI}}_{B}(t) \tag{4}$$

$$\dot{\mathbf{H}}^{c}_{B}(t) = \mathbf{L}^{c}_{B}(t) - \omega^{\mathrm{BI}}_{B}(t) \times \mathbf{H}^{c}_{B}(t) - \dot{J}^{c}_{B}(t)\, \omega^{\mathrm{BI}}_{B}(t) \;= J^{c}_{B}(t)\, \dot{\omega}^{\mathrm{BI}}_{B}(t) \tag{5}$$

Where $\mathbf{L}^{c}_{B}(t) \in \mathbb{R}^3$ is representative of both internal and external torques acting on the system. The total moment of inertia (MOI) matrix of the ABS is time varying due to the variable translatable mass positions, expressed as $J^{c}_{B}(t) \in \mathbb{R}^{3\times 3}$.

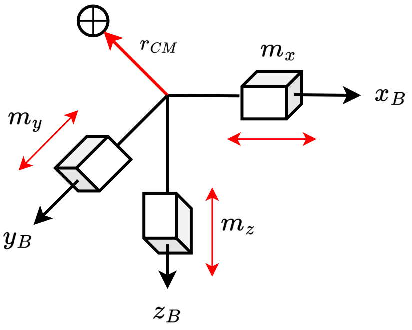


Figure 2: A Visualization of the Offset Between Coordinate Systems

When considering the ACS under study, the influence of gravity must be accounted for as a disturbance. In the unbalanced state, the CoM is misaligned with the air-bearing CoR, and the gravitational force applies a torque to the system proportional to this offset. This concept is illustrated in Figure 2. The purpose of the ABS is to utilize the translatable masses to tune the CoM to the CoR, making the offset between the two points, $\mathbf{r}_{\mathbf{CM}}(t) \in \mathbb{R}^3$, arbitrarily small and the gravitational torque negligible. We can redefine the system as Equation 6, considering the external and internal portions of $\mathbf{L}^{c}_{B}(t)$, $\mathbf{L}_g(t) \in \mathbb{R}^3$ and $\mathbf{L}_u(t) \in \mathbb{R}^3$ respectively.

$$J^{c}_{B}\left(t\right)\ \dot{\omega}^{\mathrm{BI}}_{B}\left(t\right) = \mathbf{L}_g\left(t\right) + \mathbf{L}_u\left(t\right) - \omega^{\mathrm{BI}}_{B}\left(t\right) \times J^{c}_{B}\left(t\right)\ \omega^{\mathrm{BI}}_{B}\left(t\right) - \dot{J}^{c}_{B}\left(t\right)\ \omega^{\mathrm{BI}}_{B}\left(t\right) \tag{6}$$

It is typically considered that the rate of change of the MOI is small due to the speed capabilities of the stepper motors in active balancing, and is therefore neglected. The MOI is defined as a function of the inertia of the system without the masses, $J_o \in \mathbb{R}^{3\times 3}$, and the inertia of the masses in relation to the system CoM. The system MOI is presented as Equation 7 below.

$$J^{c}_{B}\left(t\right) = J_o - \sum_{i=1}^{3} m_{m,i}\ \left[\mathbf{r}_i(t)\times\right]\left[\mathbf{r}_i(t)\times\right] \tag{7}$$

Where m_i is the i^{th} translatable mass, and $i \in \{1, 2, 3\}$ corresponds to the x, y, and z axes of B, respectively. The vector $\mathbf{r}_i \in \mathbb{R}^3$ is the absolute position of the i^{th} mass' CoM in B, of the form $\mathbf{r}_i = [r_{x,i}\ \ r_{y,i}\ \ r_{z,i}]^T$. For m_i, the i^{th} element of $\mathbf{r}_i$ is variable, where the other positions are constant and known, since a mass moves along a single Cartesian axis. The disturbance torque on the system due to gravity is then defined as Equation 8. Note how this term will go to zero once CoM/CoR alignment is achieved.

$$\mathbf{L}_g\left(t\right) = \mathbf{r}_{\mathrm{CM}}\left(t\right) \times m_{\mathrm{tot}}\ \mathbf{g}_B\left(t\right) \tag{8}$$

The constant m_{tot} is the total mass of the system, i.e., $m_{tot} = m_{sys} + 3m_i$, where m_{sys} is the mass of the system without the masses. $\mathbf{g}_B(t) \in \mathbb{R}^3$ is the gravity vector in the body frame B, expressed as the following:

$$\mathbf{g}_B\left(t\right) = R^{I}_{B}\left(t\right)\ g\ \hat{\mathbf{k}}_I \tag{9}$$

The constant g is the assumed gravitational acceleration of $-9.81m/s^2$ and $\hat{\mathbf{k}}_I$ is the unit vector in the positive z direction of the inertial frame I. To obtain the gravity in the body frame, the transformation matrix $R_B^I(t)$ is applied to relate I and B. The transformation is defined as Equation 10, defined by Markley and Crassidis[29] as a function of the quaternion $\mathbf{q}(t)$. Note the quaternion elements of $R_B^I(t)$ are time-varying, excluded for brevity.

$$R_B^I\,(t) = \begin{bmatrix} q_1^2 - q_2^2 - q_3^2 + q_4^2 & 2\,(q_1\;q_2 + q_4\;q_3) & 2\,(q_1\;q_3 - q_4\;q_2) \\ 2\,(q_1\;q_2 - q_4\;q_3) & -q_1^2 + q_2^2 - q_3^2 + q_4^2 & 2\,(q_2\;q_3 + q_4\;q_1) \\ 2\,(q_1\;q_3 + q_4\;q_2) & 2\,(q_2\;q_3 - q_4\;q_1) & -q_1^2 - q_2^2 + q_3^2 + q_4^2 \end{bmatrix} \tag{10}$$

The offset vector $\mathbf{r}_{CM}(t)$ is the distance between the CoR and CoM at time t. The system CoM is defined in reference to the origin of the frame I and can generally be described as Equation 11 (for modeling). The influence of the mass motion on the offset vector is also derived from 11, as Equation 12.[23]

$$\mathbf{r}_{\text{CM}} = \frac{1}{m_{\text{tot}}} \sum_{j=1}^{4} m_j\;\mathbf{r}_j \tag{11}$$

Where $\mathbf{r}_j$ is the absolute position of each body's CoM in the system relative to I and $j \in \{1, 2, 3, 4\}$. The bodies in this case are the three translating masses ($m_{m,1}$, $m_{m,2}$, and $m_{m,3}$) and the rest of the system m_{sys}. The vector $\mathbf{r}_m(t)$ is the variable position of the translatable mass in B.

$$\Delta\mathbf{r}_{\text{CM}} = \frac{m_m}{m_{\text{tot}}}\;\Delta\mathbf{r}_m \tag{12}$$

The internal torque $\mathbf{L}_u(t)$ can represent the ABS or satellite actuators in this application. With active mass movement for balancing as demonstrated in,[19,23] a torque can be generated from their movement. For the proposed estimation methods, however, the masses are not active and this term is zero. The utility of this term will be demonstrated in Section 3.2 and 3.3, where the influence of the reaction wheels is considered.

3. PARAMETER ESTIMATION AND CONTROL

For the balancing of the system, there are multiple avenues that can be explored. Of the methods featured in literature, balance in these systems is most commonly achieved actively or semi-actively. Active methods utilize the xy plane balancing and subsequent online estimation and tuning of the z axis, whereas more passive methods analyze a controlled or free response of the data, estimate the parameters, and move the masses to the next iteration if off balance.

The following section will explore a few of these methods mathematically and apply them in Section 5. The first method is an iterated process used by both Young[21] and da Silva[30] that determines and tunes $\mathbf{r}_{CM}(t)$ from free response data with a linear model-based Kalman filter (KF). The second method operates in the same manner, instead utilizing controlled response data and producing inertial component estimation using a linearized nonlinear model, and the extended Kalman filter (EKF).[20,31] The third method estimates parameters using a fully nonlinear model and the unscented Kalman filter (UKF),[19,30] using controlled response data as with the EKF.

3.1 $\mathbf{r}_{CM}(t)$ KF Estimation from Free-Response Data

The method outlined by Young[21] proposes a linear least squares approach to estimating the offset vector from free-response data. However, in testing this method, it was observed that estimates were unreliable for large system angular velocities, and the perturbations due to the air-bearing flow. Instead, we apply the KF to this model, where to achieve knowledge of $\mathbf{r}_{CM}$ from free response data, we must have sufficient knowledge of the angular position and velocity. The diagonal inertial components are also required for computation, which can immediately be noted as a disadvantage to applying this method. The linear model-based method is is defined by Equation 13, the simplified equations of motion.

$$\dot{\omega}\left(t\right)=\left[\begin{array}{c} \frac{mg}{J_{\text{xx}}}\ \left(-r_{\ \text{CM},y}\ \cos\varphi\ \cos\theta+r_{\ \text{CM},z}\ \sin\varphi\ \cos\theta\right) \\ \frac{mg}{J_{\text{yy}}}\ \left(-r_{\ \text{CM},x}\ \cos\varphi\ \cos\theta+r_{\ \text{CM},z}\ \sin\theta\right) \\ \frac{mg}{J_{\text{zz}}}\ \left(-r_{\ \text{CM},x}\ \sin\varphi\ \cos\theta-r_{\ \text{CM},y}\ \sin\theta\right) \end{array}\right] \tag{13}$$

$$\omega_{k+1}-\omega_k=\left[\begin{array}{c} \frac{mgT_s}{J_{\text{xx}}}\ \left(-r_{\ \text{CM},y}\ \left((c\varphi\ c\theta)_{k+1}+(c\varphi\ c\theta)_k\right)+r_{\ \text{CM},z}\left(\ (s\varphi\ c\theta)_{k+1}+(s\varphi\ c\theta)_k\right)\right) \\ \frac{mgT_s}{J_{\text{yy}}}\ \left(-r_{\ \text{CM},x}\ \left((c\varphi\ c\theta)_{k+1}+(c\varphi\ c\theta)_k\right)+r_{\ \text{CM},z}\ \left(s\theta_{k+1}+s\theta_k\right)\right) \\ \frac{mgT_s}{J_{\text{zz}}}\ \left(-r_{\ \text{CM},x}\ \left((s\varphi\ c\theta)_{k+1}+(s\varphi\ c\theta)_k\right)-r_{\ \text{CM},y}\ \left(s\theta_{k+1}+s\theta_k\right)\right) \end{array}\right] \tag{14}$$

Where $\dot{\omega}(t)\in\mathbb{R}^3$ is the angular velocity of the ABS, and φ and θ are the time-varying roll and pitch, respectively. The total mass of the system m_{tot} is represented here by m. The left side of the equation can be discretized with the trapezoidal rule and expressed as Equation 14. The sampling interval is represented by T_s, where for real applications with varying sampling, it is more correctly expressed as $\Delta T_k=t_{k+1}-t_k$. The right side of the equation can be expressed linearly as $\omega_{k+1}=\omega_k+\phi\ \mathbf{r}_{CM}$, with the full equation as 15, and the elements of the skew-symmetric matrix ϕ as 16.

$$\left[\begin{array}{c}\omega_x^{k+1}\\ \omega_y^{k+1}\\ \omega_z^{k+1}\end{array}\right]=\left[\begin{array}{c}\omega_x^{k}\\ \omega_y^{k}\\ \omega_z^{k}\end{array}\right]+\left[\begin{array}{ccc}0&\phi_{12}^k&\phi_{13}^k\\ \phi_{21}^k&0&\phi_{23}^k\\ \phi_{31}^k&\phi_{32}^k&0\end{array}\right]\left[\begin{array}{c}r_{\ \text{CM},x}\\ r_{\ \text{CM},y}\\ r_{\ \text{CM},z}\end{array}\right] \tag{15}$$

$$\begin{array}{ll} \phi_{12}^k=\frac{-mgT}{2\ J_{\text{xx}}}\left[(c\varphi\ c\theta)_{k+1}+(c\varphi\ c\theta)_k\right] & \phi_{13}^k=\frac{mgT}{2\ J_{\text{xx}}}\left[(s\varphi\ c\theta)_{k+1}+(s\varphi\ c\theta)_k\right] \\ \phi_{21}^k=\frac{mgT}{2\ J_{\text{yy}}}\left[(c\varphi\ c\theta)_{k+1}+(c\varphi\ c\theta)_k\right] & \phi_{23}^k=\frac{mgT}{2\ J_{\text{yy}}}\left[(s\theta)_{k+1}+(s\theta)_k\right] \\ \phi_{31}^k=\frac{-mgT}{2\ J_{zz}}\left[(s\varphi\ c\theta)_{k+1}+(s\varphi\ c\theta)_k\right] & \phi_{32}^k=\frac{-mgT}{2\ J_{zz}}\left[(s\theta)_{k+1}+(s\theta)_k\right] \end{array} \tag{16}$$

For estimation of $\mathbf{r}_{CM}$, we apply the KF algorithm to this model. A discrete linear model is instead derived of the form $\mathbf{x}_{k+1}=F\mathbf{x}_k$, where the KF can estimate $\mathbf{r}_{CM}$ without input. The system matrix $F\in\mathbb{R}^{n\times n}$ is defined in Equation 17, which models the system motion and enables the offset vector to be estimated. The estimated state vector $\hat{\mathbf{x}}_k\in\mathbb{R}^n$ is composed of the angular velocity and offset vector $\hat{\mathbf{x}}_k=[\omega_x^k\ \ \omega_y^k\ \ \omega_z^k\ \ r_{CM}^x\ \ r_{CM}^y\ \ r_{CM}^z]$, where $n=6$. Since $\mathbf{r}_{CM}$ is not directly measurable, the measurement vector is represented as $\mathbf{z}_k\in\mathbb{R}^m$, with $m=3$. Note then that the constant measurement matrix $C\in\mathbb{R}^{m\times n}$ is represented as $[I_{3\times3}\ \ \ 0_{3\times3}]$.

$$\mathbf{x}_{k+1}=\left[\begin{array}{cc}I_{3\times3}&\phi\\ 0_{3\times3}&I_{3\times3}\end{array}\right]\mathbf{x}_k \tag{17}$$

The KF process is outlined thoroughly in Section 3.2 (Equations 21-28). The time-varying linearized system matrix A_k is instead represented as F for this application, and the *a priori* estimate is computed with Equation 17, not as a nonlinear function considering input. The method can be applied to the system in linear time-invariant (LTI) F or linear time-varying (LTV) F_k form. In the LTI system, the ϕ-matrix of Equation 17 is linearized around the initial condition. This assumption is valid for small angular velocities ω, though there is a trade-off between accuracy and computational speed. For balancing, the masses are iteratively moved after the $\mathbf{r}_{CM}$ estimation, governed by Equation 18. This equation is applied for each method. It was noted by Young[21] and demonstrated by da Silva[30] that increasing the data collection window allows for estimates to converge.

$$\Delta\mathbf{r}_m=-\frac{m_{\text{tot}}}{m_m}\ \mathbf{r}_{\text{CM}} \tag{18}$$

3.2 $\mathbf{r}_{CM}(t)$ and J EKF Estimation from Controlled-Response Data

Kim *et al.*[20] alternatively proposed utilizing momentum exchange devices to estimate the entries of the MOI matrix J_B in addition to the CoM location w.r.t the CoR. This method overcomes the issue of having to have knowledge of the system inertia, which changes every time the masses move. The method is computable having knowledge of the reaction wheel velocity ω_W and inertia J_W, and the NACS angular position and velocity from the IMU. Considering the off-balance ABS with momentum wheel actuation, we obtain Equation 19.

$$J_B\, \dot{\omega}\,(t) + \omega\,(t) \times J_B\, \omega\,(t) = -\dot{\mathbf{h}}\,(t) - \omega\,(t) \times \mathbf{h}\,(t) + \mathbf{r}_{\text{CM}} \times m_{\text{tot}}\, \mathbf{g}_B\,(t) \tag{19}$$

In this case, since the masses are not moving during data collection, J_B and $\mathbf{r}_{CM}$ are constant. The momentum contribution due to actuation $\mathbf{h}(t) \in \mathbb{R}^3$ is a function of the reaction wheel inertia J_w, velocity $\omega_w(t)$, and geometry mapping W_4, as represented by Equation 20. The mapping is required when the RWA are not aligned with principal axes and redundant configurations. The NACS utilizes a pyramidal configuration, where the geometry was defined for the setup by Hill *et al.*[32] and Newton *et al.*[33,34]

$$\mathbf{h}(t) = W_4\, J_w\, \omega_w(t) \tag{20}$$

Where Kim *et al.* apply a least squares method, Xu *et al.*[31] approach $\mathbf{r}_{CM}$ and J parameter estimation through their EKF method. Validated in,[30,31] the estimation of immeasurable parameters in this manner was proposed as an alternative to linear filtering or batch approaches. The standard recursive EKF algorithm is applied, where the nonlinear model $f(\mathbf{x}(t), \mathbf{u}(t))$ is utilized to determine the *a priori* state $\hat{\mathbf{x}}_{k+1|k} \in \mathbb{R}^n$ and state error covariance $P_{k+1|k} \in \mathbb{R}^{n\times n}$. For the application, $n = 12$, as the selected state vector is $\mathbf{x} = [\mathbf{x}_1\ \ \mathbf{x}_2]^T$, where $\mathbf{x}_1 = [\omega_x\ \ \omega_y\ \ \omega_z]^T$ and $\mathbf{x}_2 = [J_{xx}\ \ J_{yy}\ \ J_{zz}\ \ J_{xy}\ \ J_{xz}\ \ J_{yz}\ \ m\mathbf{r}_{CM,x}\ \ m\mathbf{r}_{CM,y}\ \ m\mathbf{r}_{CM,z}]^T$. Recall the total mass of the system m_{tot} is denoted here for brevity as the constant parameter m and $\omega(t) \in \mathbb{R}^3$ is the rotational velocity of NACS in the frame B. Equations 21 and 22 outline the *a priori* process.

$$\hat{\mathbf{x}}_{k+1|k} = f\left(\hat{\mathbf{x}}_{k|k}\ , \mathbf{u}_k\right) \tag{21}$$

$$P_{k+1|k} = A_k\, P_{k|k}\, A_k^T + Q_k \tag{22}$$

The system noise covariance $Q \in \mathbb{R}^{n\times n}$ is assumed constant and known. The time-varying linearized discrete system $A_k \in \mathbb{R}^{n\times n}$ is determined as the first-order Jacobian of the nonlinear function $\partial F\,(\mathbf{x}, \mathbf{u})\,/\partial \mathbf{x}$. The nonlinear model of the unbalanced system is presented as Equation 19, augmented in 23, linearized in 24.[31]

$$\begin{bmatrix} \dot{\mathbf{x}}_1\,(t) \\ \dot{\mathbf{x}}_2\,(t) \end{bmatrix} = \begin{bmatrix} f\,(\mathbf{x}\,(t)\ \ , \mathbf{u}\,(t)) \\ 0 \end{bmatrix} = F\,(\mathbf{x}\,(t)\ \ , \mathbf{u}\,(t)) \tag{23}$$

$$F_k = \begin{bmatrix} T_s \left[\frac{\partial}{\partial \omega_k} f_k\right]_{3\times 3} & T_s \left[\frac{\partial}{\partial J_k} f_k\right]_{3\times 6} & T_s \left[\frac{\partial}{\partial m\ \mathbf{r}_k} f_k\right]_{3\times 3} \\ 0_{6\times 3} & I_{6\times 6} & 0_{6\times 3} \\ 0_{3\times 3} & 0_{3\times 6} & I_{3\times 3} \end{bmatrix} \tag{24}$$

The linearization step can be performed with either a diagonal or complete MOI matrix J_B. The trade-off between computation time and accuracy is again considered, where their difference is analyzed in Section 5. The correction process of the EKF is outlined in the following Equations. The *a posteriori* portion updates the estimates based on the measurements with the gain $K_{k+1} \in \mathbb{R}^{n\times m}$, decreasing the estimation variance.

$$S_{k+1} = C_{k+1}\, P_{k+1|k}\, C_{k+1}^T + R_{k+1} \tag{25}$$

$$K_{k+1} = P_{k+1|k}\, C_{k+1}^T\, S_{k+1}^{-1} \tag{26}$$

$$\hat{\mathbf{x}}_{k+1|k+1} = \hat{\mathbf{x}}_{k+1|k} + K_{k+1}\ \left(\mathbf{z}_{k+1} - C\hat{\mathbf{x}}_{k+1|k}\right) \tag{27}$$

$$P_{k+1|k+1} = \left(I - K_{k+1}\, C_{k+1}\right)\, P_{k+1|k}\, \left(I - K_{k+1}\, C_{k+1}\right)^T + K_{k+1}\, R_{k+1}\, K_{k+1}^T \tag{28}$$

The *a posteriori* estimated state and state error covariances are denoted $\hat{\mathbf{x}}_{k+1|k+1}$ and $P_{k+1|k+1}$, respectively. The innovation and measurement noise covariances are $S_{k+1} \in \mathbb{R}^{m \times m}$ and $R_{k+1} \in \mathbb{R}^{m \times m}$, where m is the measurement vector $\mathbf{z}_{k+1} \in \mathbb{R}^m$ dimension, $m = 3$. The measurable states are the angular velocity $\omega(t)$ via IMU. Since immeasurable parameters are being estimated, the measurement matrix $C \in \mathbb{R}^{m \times n}$ is not identity and is instead represented by $C = \begin{bmatrix} I_{3\times 3} & 0_{3\times 9} \end{bmatrix}$.

3.3 $\mathbf{r}_{CM}(t)$ and J UKF Estimation from Controlled-Response Data

Using the EKF, the linearized time-varying system is applied for the *a priori* state error covariance. This assumption is usually valid for satellites and provides accurate results, as the nonlinearities of small-scale satellites due to non-diagonal inertia matrices are minimal. However, in estimating very small quantities, implementing a nonlinear model into the estimation process should result in increased accuracy. da Silva[30] implements the UKF in their research because of this, augmenting the recursive EKF process outlined in Section 3.2. The UKF is applied based on the process outlined in,[35] which linearizes the nonlinear model statistically. Sigma points are derived from the projected probability distribution of the state, which are then propagated through the nonlinear model to obtain an estimate of the mean and covariance of the estimate.

The UKF assumes that the n-dimensional random variable x with mean $\hat{x}$ and covariance P can be approximated from $2n+1$ sigma points.[35] The sigma points $X \in \mathbb{R}^{n \times 2n+1}$ and their weights $W \in \mathbb{R}^{1 \times 2n+1}$ must be generated at the start of each time step based on the current state estimate $\hat{\mathbf{x}}_{k|k}$ and state error covariance $P_{k|k}$. Recall their dimensionality from Section 3.2. The initial points are determined from Equation 29 and 30.

$$X_{1,k|k} = \hat{\mathbf{x}}_{k|k} \tag{29}$$

$$W_1 = \frac{\kappa}{n+\kappa} \tag{30}$$

Where the tunable parameter $\kappa \in (0,1)$. The first n number of sigma points ($i = 2$ to $i = n+1$) are defined from Equations 31 and 33. Note that $P_{i,k|k}$ is the i^{th} column of $P_{k|k}$ and W_i is the weight of the i^{th} sample point. The remaining n sigma points ($i = n+2$ to $i = 2n+1$) is then defined with Equation 32, where the weight equation is unchanged.

$$X_{i,k|k} = \hat{\mathbf{x}}_{k|k} + \sqrt{(n+\kappa)\, P_{i,k|k}} \tag{31}$$

$$X_{i+n,k|k} = \hat{\mathbf{x}}_{k|k} - \sqrt{(n+\kappa)\, P_{i,k|k}} \tag{32}$$

$$W_i = \frac{1}{2\,(n+\kappa)} \tag{33}$$

The sigma points at the next time step $\hat{X} \in \mathbb{R}^{n \times 2n+1}$ are then estimated by propagation through the nonlinear model, with Equation 34, and the weights are used for *a priori* state estimation, governed by Equation 35. The state error covariance can then be determined by Equation 36.

$$\hat{X}_{i,k+1|k} = f\left(X_{i,k|k}\ , u_k\right) \tag{34}$$

$$\hat{\mathbf{x}}_{k+1|k} = \sum_{i=0}^{2n} W_i\ \hat{X}_{i,k+1|k} \tag{35}$$

$$P_{k+1|k} = \sum_{i=0}^{2n} W_i\ \left(\hat{X}_{i,k+1|k} - \hat{\mathbf{x}}_{k+1|k}\right)\left(\hat{X}_{i,k+1|k} - \hat{\mathbf{x}}_{k+1|k}\right)^T + Q_k \tag{36}$$

To predict the measurements, the sigma points are then propagated through the measurement equation $\mathbf{z}_{k+1} = C\hat{\mathbf{x}}_{k+1|k}$, which is for our application linear. Equation 37 and 38 compute this as follows.

$$\hat{Z}_{i,k+1|k} = C\ \hat{X}_{i,k+1|k} \tag{37}$$

$$\hat{\mathbf{z}}_{k+1|k} = \sum_{i=0}^{2n} W_i\ \hat{Z}_{i,k+1|k} \tag{38}$$

Recall the C matrix defined in Section 3.2. The innovation covariance $P_{zz} \in \mathbb{R}^{m\times m}$ is then determined by Equation 39.

$$P_{\mathrm{zz},k+1|k} = \sum_{i=0}^{2n} W_i\ \left(\hat{Z}_{i,k+1|k} - \hat{\mathbf{z}}_{k+1|k}\right)\left(\hat{Z}_{i,k+1|k} - \hat{\mathbf{z}}_{k+1|k}\right)^T + R_{k+1} \tag{39}$$

The cross-covariance $P_{xz} \in \mathbb{R}^{n\times m}$ can then be computed as Equation 40, representing the covariance between the estimate and measurement.

$$P_{\mathrm{xz},k+1|k} = \sum_{i=0}^{2n} W_i\ \left(\hat{X}_{i,k+1|k} - \hat{\mathbf{x}}_{k+1|k}\right)\left(\hat{Z}_{i,k+1|k} - \hat{\mathbf{z}}_{k+1|k}\right)^T \tag{40}$$

The Kalman gain K_{k+1} is then determined with the cross and innovation covariances as follows, with the *a posteriori* equations as Equations 42 and 43.

$$K_{k+1} = P_{\mathrm{xz},k+1|k}\ P^{-1}_{\mathrm{zz},k+1|k} \tag{41}$$

$$\hat{\mathbf{x}}_{k+1|k+1} = \hat{\mathbf{x}}_{k+1|k} + K_{k+1}\left(\mathbf{z}_{k+1} - \hat{\mathbf{z}}_{k+1|k}\right) \tag{42}$$

$$P_{k+1|k+1} = P_{k+1|k} - K_{k+1} P_{\mathrm{zz},k+1|k}\ K^T_{k+1} \tag{43}$$

The same state vector for estimation is applied as in Section 3.2, where the nonlinear model $F\left(\mathbf{x}\left(t\right)\ ,\mathbf{u}\left(t\right)\right)$ is defined in with Equations 19 and 23.

4. METHODOLOGY

For the validation of balancing methods outlined in Section 3, we utilize the NACS experimental setup presented in Figure 3. For the parameter estimation methods that require control input to the system, a trajectory-tracking attitude control maneuver is applied. In having a time-varying trajectory the system must track, there is an internal change in momentum due to the actuators, and excitation due to a disturbance is not required.

The NACS experimental setup is divided into two main subsystems: the mock CubeSat unit and the ABS. The stand can also be considered as a key system, though currently only provides remote controllability to the air valve, and is not part of the analysis. The system is an air-bearing simulator with a range of motion of ±30° in the pitch and roll axes and ±360° in the yaw. The air pressure is regulated to approximately 14.5 psi loaded, experimentally determined to be the minimum pressure that does not cause binding or excessive planar perturbations. The CubeSat unit is fixed to the ABS and able to be removed. With this mechanism, the performance of different units can be evaluated in a modular fashion.

Previously discussed, the purpose of the ABS is to use its translatable masses to iteratively tune the system center of mass to be near coincident with the center of rotation, minimizing the occurrence of external torques due to gravity on the system. The masses are driven along a lead screw, where their position is bounded by their fixtures. An open-loop scheme is utilized for position control, where limit switches are applied for hard stops. Implementing encoders into the design is not necessary for iterative balancing, where motors are being moved a predetermined number of steps per cycle, bu would be crucial for an active balancing procedure. A NEMA 8 stepper motor rotates the screw, controlled by a TMC 2209 driver and Arduino. Figure 4 presents the MMU module, mounted on each principal axis. The electronics of the ABS are configured on a custom PCB. The ABS receives motor direction and pulse commands via Bluetooth from the on-board computer (OBC) of

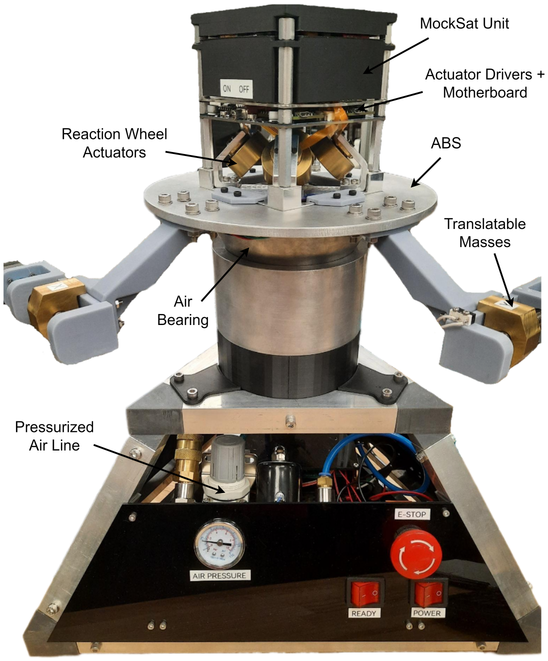


Figure 3: Nanosatellite Attitude Control Simulator

the CubeSat unit, calculated as a function of the estimated $\mathbf{r}_{CM}$ (Equation 18), based on the data obtained in an experiment. Note that for the results presented in this paper, this computation was executed on another PC for simplicity in extracting and processing data incrementally. The system configured to operate in the proposed manner, offering increased efficiency.

The mock CubeSat is responsible for performing the attitude control maneuver. Two LiPo batteries supply power to the OBC and actuators, which are in this case reaction wheels. The ESCON 24/2 motor drivers controls and senses the velocity of the four reaction wheels. The inertia of the reaction wheels was also estimated by the ESCON software, discussed in.[36] For attitude and motion feedback, the VN-100 IMU is utilized, and control operations are computed by the Raspberry Pi 3A+ OBC. In Figure 5, a high-level diagram of the interaction between the two systems is provided, visualizing the flow of power and information.

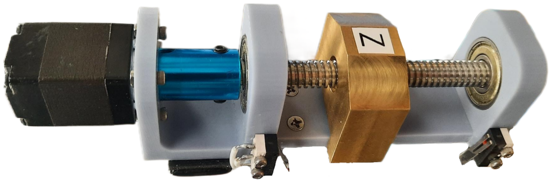


Figure 4: MMU Sub-System

For the experiments, the system was balanced manually as accurately as possible through observing its stationary elevation before the balancing trials were conducted, reducing the total trials. The attitude control routine is then performed , where the estimation of the offset vector is used in Equation 18 to determine mass movements. The process is iterated until the misalignment vector converges to a predetermined magnitude. In literature,[23] the residual torque was desired to be reduced to $1 \times 10^{-6} Nm$, with the associated stopping point of $\mathbf{r}_{CM} = 3 \times 10^{-7} m$. This is theoretically almost obtainable, where one pulse of the stepper motor is equivalent to

$1 \times 10^{-5} m$ movement, or a change in $\mathbf{r}_{CM}$ of $4 \times 10^{-7} m$. Sensor/process noise and imperfections in the estimation might affect this however. The stopping point for the experiment is when the calculated pulses reach zero, where the calculated movement falls below the 1×10^{-5} threshold (rounding down). The experiment procedure is repeated for each estimation method. Before data collection, the system must be stabilized sufficiently, where calculations assume no initial input torque. This is most crucial for the KF method of estimating $\mathbf{r}_{CM}$, as the method was demonstrated as ineffective for a non-zero initial condition in velocity and pitch/roll.

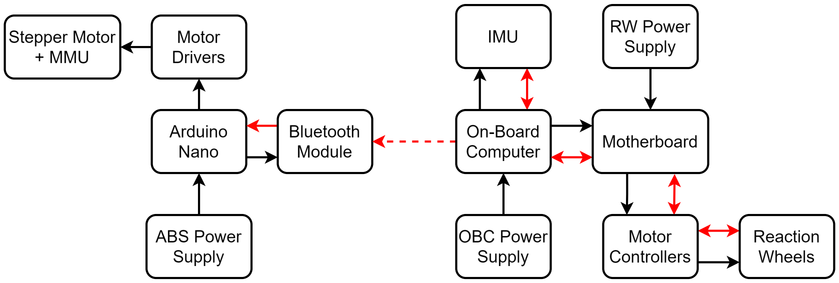


Figure 5: NACS High-Level Electronics Diagram

For the parameter estimation methods, the mass of the system and wheel inertias must be known, determined as 3.3852 kg and $J_w = [2.17 \quad 1.58 \quad 1.55 \quad 2.06]^T \times 10^{-5}\ kgm^2$, respectively. For the KF method, the system inertia J_B must also be known, which contributes to its inaccuracies. For attitude control $\mathbf{u}(t)$, a proportional-derivative (PD) controller (Equation 44) is implemented, with experimentally determined gains of $K_p = 0.4$ and $K_d = 0.075$.

$$\mathbf{u}\left(t\right) = -K_p \operatorname{sign}\left(q_4\left(t\right)\right)\ q_{1:3}\left(t\right) - K_d\ \omega_B^{\mathrm{BI}}\left(t\right) \tag{44}$$

The measurement and system noise covariance matrices R and Q were assumed constant for the EKF. For R the diagonal entries were estimated to be the variance values reported in the VN-100 IMU and ESCON 24/2 data sheets. The output voltage variance of $2.42 \times 10^{-3}\ V$ was applied for the reaction wheels, and for the quaternion and body angular velocity, the gyroscope variance values of $3 \times 10^{-6}\ rad^2/s^2$ were implemented. Choosing Q was completed by trial and error, separately for simulation and experiment datasets. The attitude control experiments were run for 4800 iterations, approximately 480 seconds.

5. EXPERIMENTAL RESULTS

5.1 Simulation Estimation Results

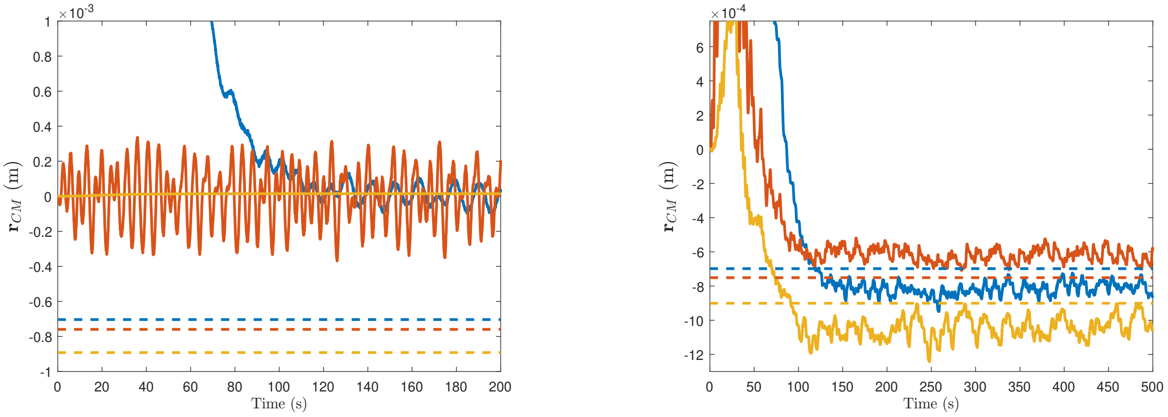
To verify the effectiveness of the methods outlined in Section 3, datasets of controlled and un-controlled system responses are generated. For the simulation, the desired parameters J_B and $\mathbf{r}_{CM}$ are known, where convergence to these values will verify the method for utilization in the balancing procedure. The simulations were programmed in Matlab at a 0.1 second sampling rate for 10000 iterations to observe convergence. The gyroscope variances described in Section 4 were applied for R. The total inertia J_B was determined from Equation 7, where J was approximated with the Solidworks model. The offset vector was also determined from this model if the masses were placed at the zero of their respective axes. System noise covariances were determined through trial and error, and are broken up by the angular velocity $Q_\omega \in \mathbb{R}^{3\times3}$, principal inertia $Q_J \in \mathbb{R}^{3\times3}$, off-diagonal inertia $Q_{J\times} \in \mathbb{R}^{3\times3}$, and offset vector $Q_{mr} \in \mathbb{R}^{3\times3}$. For the KF method, instead of Q_{mr}, Q_r is applied. These parameters are presented below.

$$\mathbf{r}_{\mathrm{CM}} = [-7.0381 \ \ -7.5656 \ \ -9.3773] \times 10^{-4}\ m \tag{45}$$

$$\begin{array}{lll} Q_\omega = 3 \times 10^{-9}\ I_{3\times3}\ \ rad^2/s^2 & Q_{J\times} = 3 \times 10^{-9}\ I_{3\times3}\ \ kg^2\ m^4 & Q_r = 5 \times 10^{-12}\ I_{3\times3}\ \ kg^2\ m^4 \\ Q_J = 3 \times 10^{-9}\ I_{3\times3}\ \ kg^2\ m^4 & Q_{\mathrm{mr}} = 3 \times 10^{-12}\ I_{3\times3}\ \ kg^2\ m^2 & \end{array} \tag{46}$$

$$J = \begin{bmatrix} 0.0218 & -0.0023 & -2.62 \times 10^{-4} \\ -0.0023 & 0.0316 & -4.05 \times 10^{-4} \\ -2.62 \times 10^{-4} & -4.05 \times 10^{-4} & 0.1729 \end{bmatrix} \text{ kg } m^2 \tag{47}$$

For the KF method with the linear model, we can either consider the LTI or LTV case, where both are plotted as Figure 6. The dotted lines represent the constant true value. The KF method does not consider external input, so free response data is utilized with stable initial conditions and small angular velocities. The LTI KF does not converge to the true $\mathbf{r}_{CM}$, where the z component is constrained to its initial condition, the y component oscillates heavily about its initial condition, and the x component converges to zero with slight oscillation. The LTI assumption is therefore not suitable for the application, resulting in significant error. Figure 6b shows the LTV KF simulation results, which show stochastic convergence to a value with an average error of approximately ±0.1 mm. Note for the plots, the blue, red, and yellow data lines represent the x, y, and z components of $\mathbf{r}_{CM}$, respectively.



(a) LTI KF Estimation

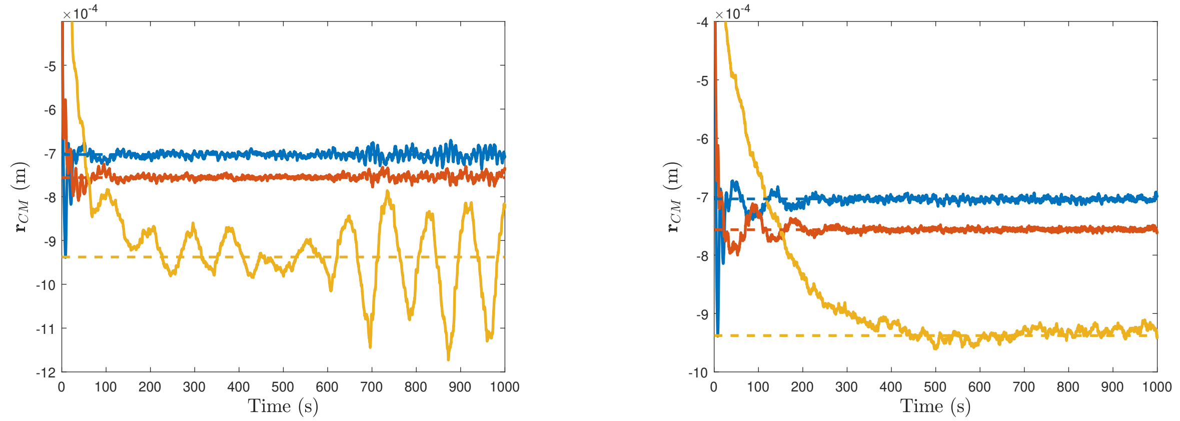
(b) LTV KF Estimation

Figure 6: Kalman Filter $\mathbf{r}_{CM}$ Estimation

For the EKF method, we linearize a nonlinear equation that describes the motion of the system with some imbalance. With this method as well, we can consider either the simplified or complex linearization, where there is a tradeoff between accuracy and computation. A controlled response is simulated, and the data from the trajectory tracking attitude control scenario is used for validation of this method. The same data set is used for the UKF as well. Figure 7 and 8 presents the performance of each respective method, for the estimation of $\mathbf{r}_{CM}$ and J. Note for the plots, the blue, red, and yellow data lines represent the x, y, and z components of both $\mathbf{r}_{CM}$ and principal J elements. It was observed that the simplified EKF produced unreliable estimates, where its process was sensitive to noise and in some cases resulted in unstable behaviour on one or more axes. The off-diagonal element estimation of the simplified EKF was also unsuccessful, where its change was due to noise. The complex EKF was demonstrated to be much more effective, estimating parameters with minimal error.

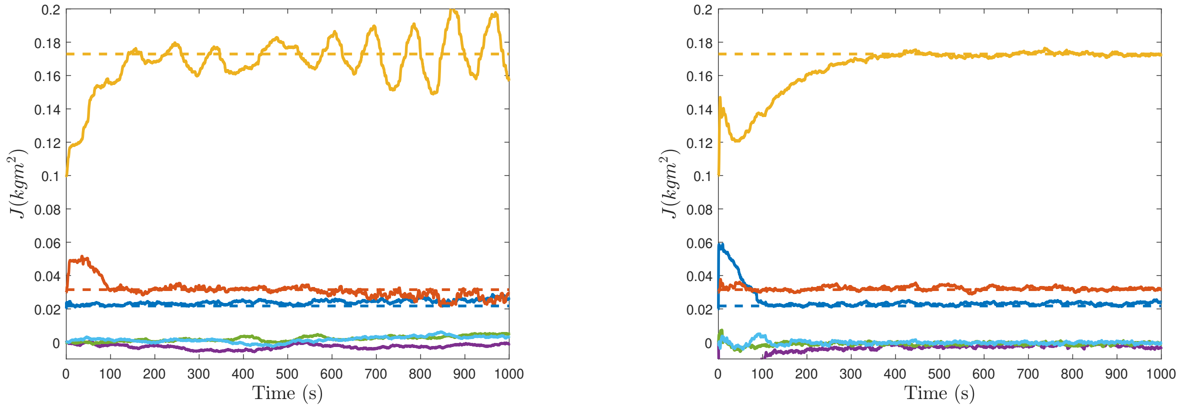
The UKF method does not require linearization, such that the full nonlinear equation can be leveraged. The performance of the UKF is presented in Figure 9 for $\mathbf{r}_{CM}$ and J. The UKF is demonstrated to be similar in performance when compared to the complete EKF, where convergence occurs slightly faster and overshoot is not as prevalent in the estimation of diagonal J elements. The difference in accuracy of convergence to the true value is essentially negligible. The similarity is due in part to the linearization with respect to all J in the EKF so they are related to the output, resulting in near matched dynamics. The computation time of the EKF method was observed to be slightly faster than the UKF for the simulated data.

Tables 1 and 2 highlight the results of the simulations for the simplified and complex EKFs and the UKF. The average value that was converged to is recorded for the offset vector and MOI elements, along with the sum of the mean absolute error (MAE) and the average percent error of the converged values. The estimated parameters



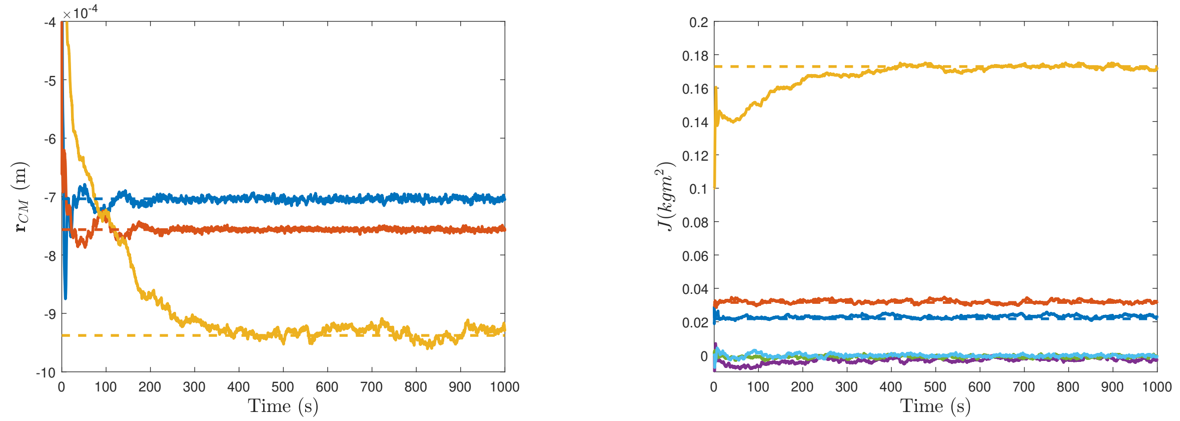
(a) Simplified EKF Estimation (b) Full EKF Estimation

Figure 7: Extended Kalman Filter $\mathbf{r}_{CM}$ Estimation



(a) Simplified EKF Estimation (b) Full EKF Estimation

Figure 8: Extended Kalman Filter J Estimation



(a) $\mathbf{r}_{CM}$ Estimation (b) J Estimation

Figure 9: Unscented Kalman Filter $\mathbf{r}_{CM}$ and J Estimation

were determined by the average of values after convergence, at approximately 500 seconds. All results were the average of 10 trials, where results differed in each simulation due to noise.

Table 1: $\mathbf{r_{CM}}$ Estimation of Simulated Data Results (m)

	$\mathbf{r}_{CM,x}$	$\mathbf{r}_{CM,y}$	$\mathbf{r}_{CM,z}$	Σ_{MAE}	% Error
3J-EKF	-0.0007036034397	-0.0007538757215	-0.0009577965876	0.006147272212	0.84123
6J-EKF	-0.0007042708127	-0.0007563962197	-0.0009332632154	0.005586199715	0.19707
UKF	-0.0007045854933	-0.0007564524897	-0.0009334177078	0.005497240483	0.18518

Table 2: J Estimation of Simulated Data Results (kgm^2)

	J_x	J_y	J_z	J_{xy}	J_{xz}	J_{zy}	Σ_{MAE}	% Diag. Error	% Off. Error
3J-EKF	0.0260002	0.0253788	0.1766550	0.0003517	0.0025317	0.0053324	0.0468479	13.71022	992.42589
6J-EKF	0.0230196	0.0321004	0.1724882	-0.0023685	-0.0009641	-0.0002785	0.0156387	2.56214	49.10934
UKF	0.0229490	0.0320774	0.1724407	-0.0027390	-0.0008392	-0.0002998	0.00992007	2.42072	46.88868

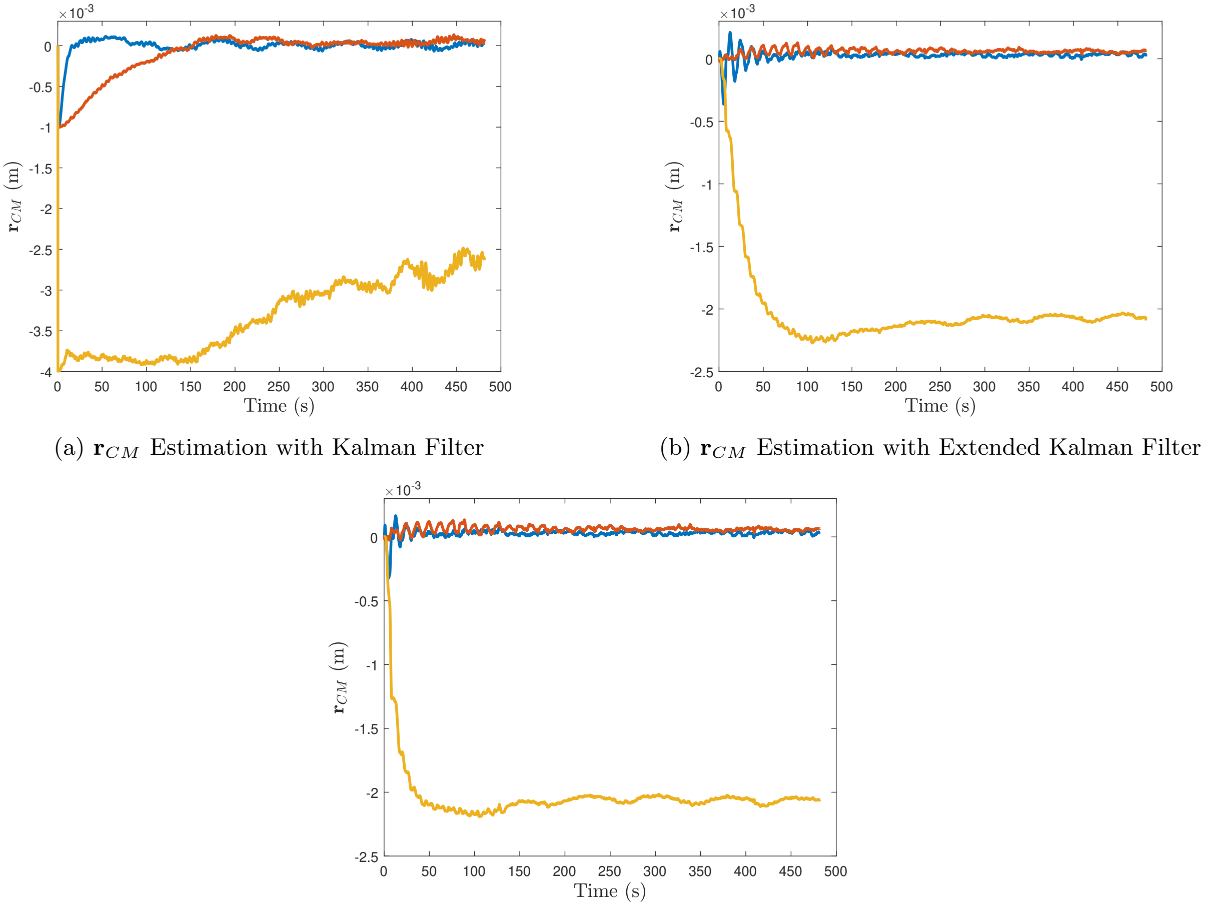
5.2 Experiment Estimation Results

For experiments, the LTV KF, full EKF, and UKF are applied for parameter estimation in the balancing process, based on their demonstrated effectiveness in Section 5.1. To further validate these methods, they can be applied to free and controlled response data sets. Since the true values are not known, the methods can be categorized as successful if they converge to values for varying initial P and a bounded range of Q and R, and they are consistent with each other. The NACS was balanced manually to a satisfactory degree, and data was collected for some unknown mass positions. The data was processed in Matlab, using similar scripts as in Section 5.1. Observing the parameter estimation effectiveness on real data beforehand is also important in determining the system noise covariance for possible online estimation. The determined Q values for the experiments are presented below. To ensure convergence, experiments were performed for 4800 iterations at a sampling rate of approximately 0.1 seconds. Other model parameters and general operation are described in Section 4.

$$\begin{array}{lll} Q_\omega = 3\times 10^{-9}\ I_{3\times 3}\ \ rad^2/s^2 & Q_{J\times} = 3\times 10^{-10}\ I_{3\times 3}\ \ kg^2\ m^4 & Q_r = 3\times 10^{-12}\ I_{3\times 3}\ \ kg^2\ m^4 \\ Q_J = 3\times 10^{-10}\ I_{3\times 3}\ \ kg^2\ m^4 & Q_{\mathrm{mr}} = 3\times 10^{-12}\ I_{3\times 3}\ \ kg^2\ m^2 & \end{array} \tag{48}$$

The results suggest the system's $\mathbf{r}_{CM}$ components are very small in the x and y axes, where the z component is small but sufficiently unbalanced. The magnitude of the planar values is expected considering the fine manual balancing conducted beforehand. The relatively large z component is also expected, since manual tuning of this axis can only be performed after observing the error in a trajectory tracking scenario, or skillful observation of pendulum behaviour. The utility of an automatic balancing system is reinforced by this aspect. The estimation of J is difficult to verify, though it can be concluded that the converged values are reasonably accurate based on similarity with model values and intuition. In choosing Q, convergence with minimal noise only occurred for the Q_J, $Q_{J\times}$, and Q_{mr} presented as Equation 48. Q_ω determined the magnitude of converged principal values, where off-diagonal elements were generally consistent for the stable range. From simulations and the form of the NACS, it should be expected that J_{xx} and J_{yy} should be similar, with J_{zz} much larger. The chosen Q_ω lied within the stable range and exhibited this behaviour, where stable values differing by a factor of 10 did not. The experimental J is close to the simulated value, expected to be significantly larger in x and y due to an update to the arm lengths. The $\mathbf{r}_{CM}$ and J estimation of the NACS are presented in Figure 10 and 11.

Overall, all methods estimated x and y components of the offset vector with comparable accuracy, featuring slight oscillations. The KF was not able to converge to the true z component reliably, where the UKF and EKF performed equally as well. The insufficient performance of the KF could be the result of neglecting nonlinearities in the model or a slight acceleration in the data observed in the z axis rotation. As previously mentioned, the initial condition also affects performance, were large velocities or inclinations result in unstable behaviour. Generally, the UKF converges slightly faster than the EKF.

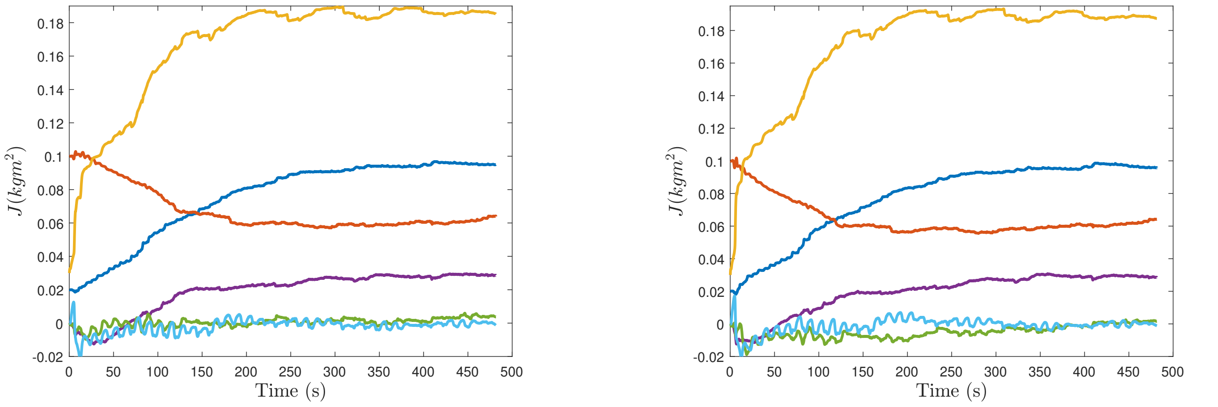


(a) $\mathbf{r}_{CM}$ Estimation with Kalman Filter

(b) $\mathbf{r}_{CM}$ Estimation with Extended Kalman Filter

(c) $\mathbf{r}_{CM}$ Estimation with Unscented Kalman Filter

Figure 10: $\mathbf{r}_{CM}$ Estimation Results



(a) J Estimation with Extended Kalman Filter

(b) J Estimation with Unscented Kalman Filter

Figure 11: J Estimation Results

5.3 Experiment Balancing Results

With the parameter estimation methods validated using experimental data, the balancing procedure can be conducted. From Tables 1 and 2, and Figure 10, it can be observed that the difference in performance between the most accurate methods, the Full-J EKF and UKF, is minimal. Therefore, for the balancing process, only the UKF will be implemented. The initial iteration of the system was roughly balanced, and the movement of the center of mass across iterations is visualized in Figure 12.

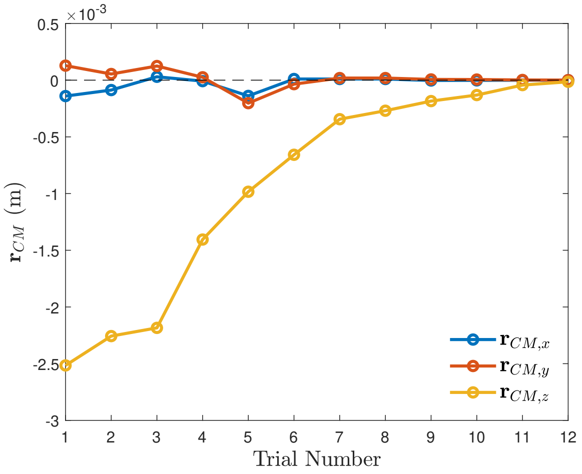


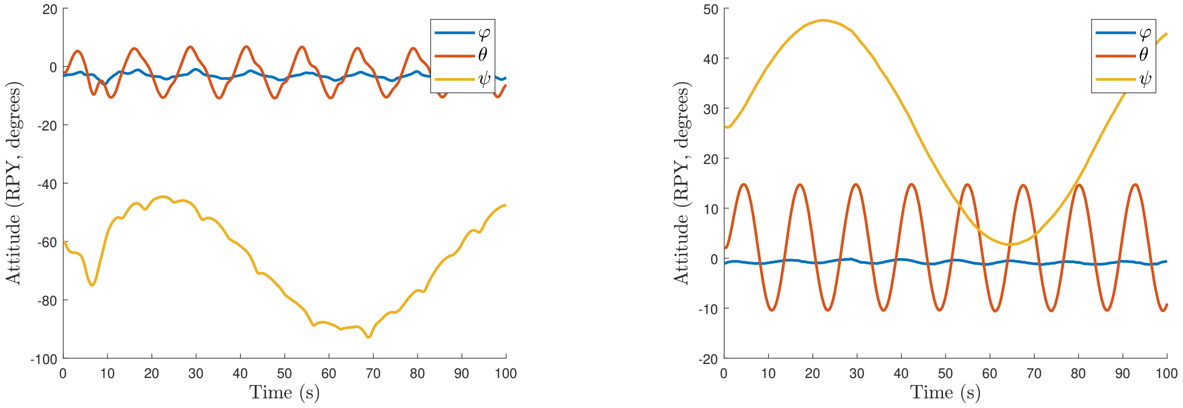
Figure 12: $\mathbf{r}_{CM}$ Across Balancing Iterations

It was observed from the experimental data that the prediction of J did not always converge to expected values each iteration, where the offset vector estimation did not experience this effect. Increasing Q_ω assisted in reducing noise and maintaining J convergence as the trials increased. However, due to the actuator saturation in later tests, the final predicted MOI is inconclusive. It was generally observed that the principal values appeared to lie around the model and initial estimates. The iterative balancing process took 12 trials, with a final achieved $\mathbf{r}_{CM}$ magnitude of 1.43×10^{-5} m, resulting in a residual torque of 4.84×10^{-5} Nm.

Because the filter converges to estimated values for large offsets (greater than 2 mm) after approximately 400 seconds, this method of balancing is not the most efficient in practice. As the system becomes more balanced, saturation of the actuators was also observed occasionally around this point in the experiment, resulting in decaying estimation quality. Although, as the system becomes more balanced, time for convergence decreases as well. Having the CoM lower than the center of rotation induces oscillations, but also reduces the torque that the actuators must produce. Regardless of inefficiency, the method is quite effective in tuning z, which is difficult to do visually, and fine tuning x and y. Increasing the sampling rate of the system could also result in faster convergence of the filter.

The performance of the system after the balancing procedure that is achieved is greatly improved from an attitude control perspective. This is exemplified in Figure 13, where the controlled response on the left shows heavy oscillations around the trajectory, as a result of a variable torque with the system's changing inclination. The system here is reasonably balanced in x and y, but offset by about 2.5 mm in positive z. Offsets in this direction are stable but induce tracking error, where offsets in negative z produce an unstable inverted pendulum effect. In contrast, the balanced system's response features almost no oscillation, tracking the desired trajectory tightly. Note the trajectory waveforms are the same in ψ, where the difference observed is due to initial condition.

Additionally, the period of oscillation in free responses can be analyzed to further validate the reduction in $\mathbf{r}_{CM}$. Young[21] uses the pendulum analogy, where Chesi[19] uses energy methods. Here, we employ Young's method and compare the oscillations before and after balancing, presented in Figure 14. Using Equation 49, we can roughly determine the magnitude of the offset vector, with some knowledge of the principal inertia in the i^{th} axis and system mass, by disturbing the system on the i^{th} axis. Recall that m represents the mass of the



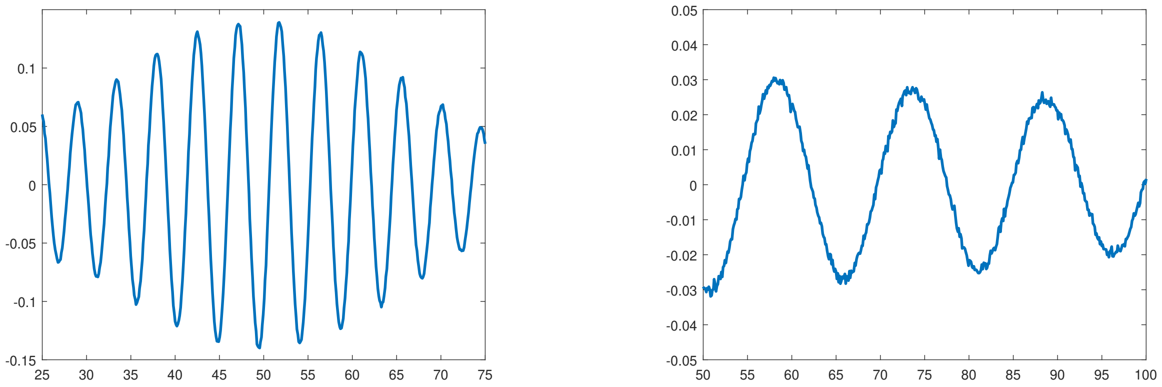
(a) Unbalanced Attitude Control Performance

(b) Balanced Attitude Control Performance

Figure 13: Balanced and Unbalanced System Attitude Control Performance

system, g is is the constant gravitational acceleration, and the period of oscillation is T in seconds. Choosing the y axis, Figure 14 indicates the improvement, increasing the period of oscillation from 4.5 to 16 seconds after disturbance, which correspond to approximate $\mathbf{r}_{CM}$ magnitudes of 1×10^{-3} m and 7.9×10^{-5} m. This is an approximation, due to the motion not being purely on a single axis.

$$T = 2\pi \left[\frac{mg \ |\mathbf{r}_{\text{CM}}|}{J_{ii}} \right]^{-1/2} \rightarrow \ |\mathbf{r}_{\text{CM}}| = \frac{4\pi^2 \ J_{ii}}{mg \ T^2} \tag{49}$$



(a) Unbalanced Free Response

(b) Balanced Free Response

Figure 14: Balanced and Unbalanced Period of Oscillation

6. CONCLUSION

In this paper, several estimation strategies were applied to a nanosatellite attitude control simulator for accurate estimation of unmeasureable parameters, quantifying setup uncertainties. The utility of in-lab attitude control simulation was highlighted as being useful in system and algorithm design, mitigating performance issues in uncertain environments. The air-bearing class of simulator was thoroughly discussed, where each use different methods for determining the system inertia, and CoM relative to the CoR. The satellite system model with actuation and unbalanced was outlined, along with the KF, EKF, and UKF under analysis.

The identification of simulator system parameters was crucial for determining the level of unbalance in the system, such that it can be corrected. The unbalance of the system is directly related to the controller performance and efficiency, as well as the congruency between the simulator and a real satellite environment. Determining these parameters also improves modelling accuracy for computational simulations. Parameter estimation with the KF, EKF, and UKF was validated first by simulated data, where convergence to the known true parameters is easily verifiable. Different formulations of the KF and EKF were tested, where less generalized models performed better. The effective methods were then tested using attitude and motion data from an unbalanced system, where the complex EKF and UKF were able to successfully estimate the offset vector and MOI, with unknown accuracy. The difference in performance between the two was determined to be negligible. The UKF was then applied for the balancing process, where an initially unbalanced system was iteratively balanced through parameter identification and mass movement between data collection trials. The system was determined to be balanced at the end of the experiment from the estimation, which validated by analyzing the improvement of the controlled and free response of the simulator. The final offset vector was estimated to be approximately 1.43×10^{-5} m in magnitude, resulting in a residual torque of 4.84×10^{-5} Nm. The final results of J were inconclusive.

The methods presented in this paper were effective in balancing and estimating parameters, but the length of the experiments required for convergence were limiting. Future research can involve the active balancing methods explored by Chesi[19] and Bahu,[23] where planar balancing in the x and y axes is instead performed online through attitude feedback control of the masses. This would theoretically streamline experiments. The investigation of other nonlinear estimation strategies, such as particle filters or cubature Kalman filters, could also be a possible avenue, along with identification of other parameters, such as reaction wheel inertia. Machine learning methods focused around system identification, such as physics informed neural networks (PINNs), could also be investigated for the application. In addition, closed loop mass position control is a necessary inclusion towards planar balancing, and testing the applicability for varying satellite sizes would serve as a proof of concept for the design.

REFERENCES

[1] Agency, E. S., "Esa's annual space environment report 2023," (December 2023). Accessed March 17, 2024.

[2] Mahanti, P., Robinson, M., Wagner, R., Estes, N., and Humm, D., "First look, first results - comparing secondary illumination at lunar permanently shadowed regions from the first shadowcam image and topography based simulation," in [*IGARSS 2023 - 2023 IEEE International Geoscience and Remote Sensing Symposium*], 4162–4165 (2023).

[3] Zou, Y., Zhu, Y., Bai, Y., Wang, L., Jia, Y., Shen, W., Fan, Y., Liu, Y., Wang, C., Zhang, A., Yu, G., Dong, J., Shu, R., He, Z., Zhang, T., Du, A., Fan, M., Yang, J., Zhou, B., Wang, Y., and Peng, Y., "Scientific objectives and payloads of tianwen-1, china's first mars exploration mission," *Advances in Space Research* **67**(2), 812–823 (2021).

[4] Dotto, E., Della Corte, V., Amoroso, M., Bertini, I., Brucato, J., Capannolo, A., Cotugno, B., Cremonese, G., Di Tana, V., Gai, I., Ieva, S., Impresario, G., Ivanovski, S., Lavagna, M., Lucchetti, A., Mazzotta Epifani, E., Meneghin, A., Miglioretti, F., Modenini, D., Pajola, M., Palumbo, P., Perna, D., Pirrotta, S., Poggiali, G., Rossi, A., Simioni, E., Simonetti, S., Tortora, P., Zannoni, M., Zanotti, G., Zinzi, A., Cheng, A., Rivkin, A., Adams, E., Reynolds, E., and Fretz, K., "Liciacube - the light italian cubesat for imaging of asteroids in support of the nasa dart mission towards asteroid (65803) didymos," *Planetary and Space Science* **199**, 105185 (2021).

[5] Wall, M., "'Vibration Disturbance' caused failure of new indian rocket, ISRO says," (2023). Accessed March 12, 2024.

[6] Picard, M., "An overview of the csa recent activities in space robotics," (2019).

[7] Melega, N., Marchese, V., Paškevičiūtė-Kidron, A., Carnicero, B., Dominguez, N. L., Casaled, O. A., Babkina, I., Marin, A., Steyn, H., Buckley, L., et al., "The esa ϕsat-2 mission: An ai enhanced multispectral cubesat for earth observation," *Annual Small Satellite Conference* **37** (2023).

[8] Poghosyan, A. and Golkar, A., "Cubesat evolution: Analyzing cubesat capabilities for conducting science missions," *Progress in Aerospace Sciences* **88**, 59–83 (2017).

[9] Schwartz, J., Peck, M., and Hall, C., "Historical review of air-bearing spacecraft simulators," *Journal of Guidance, Control, and Dynamics* **26** (05 2003).

[10] da Silva, R. C., Borges, R. A., Battistini, S., and Cappelletti, C., "A review of balancing methods for satellite simulators," *Acta Astronautica* **187**, 537–545 (2021).

[11] Schwartz, J. and Hall, C., "Comparison of system identification techniques for a spherical air-bearing spacecraft simulator," (01 2004).

[12] Rybus, T., Seweryn, K., Oles, J., Basmadji, F. L., Tarenko, K., Moczydlowski, R., Barcinski, T., Kindracki, J., Mezyk, L., Paszkiewicz, P., and Wolanski, P., "Application of a planar air-bearing microgravity simulator for demonstration of operations required for an orbital capture with a manipulator," *Acta Astronautica* **155**, 211–229 (2019).

[13] Xie, Y., Yao, W., Li, X., Wang, N., Zheng, X., and Chen, X., "Hybrid digital twin for satellite temperature field perception and attitude control," *Advanced Engineering Informatics* **60**, 102405 (2024).

[14] Qi, N., Xu, Z., Chen, Y., Wang, G., and Wu, F., "System design and attitude control experiment of a 5-dof spacecraft simulator," in [*2014 International Conference on Mechatronics and Control (ICMC)*], 876–880 (2014).

[15] Yan, W., Cheng, H., and Lin, C. X., "Automatic mass balancing and design of a six degrees-of-freedom air bearing spacecraft simulator," in [*2015 34th Chinese Control Conference (CCC)*], 5696–5700 (2015).

[16] Jovanovic, N., Pearce, J., and Praks, J., "Design and testing of a low-cost, open source, 3-d printed air-bearing-based attitude simulator for cubesat satellites," *Journal of Small Satellites* **8**, 859–880 (01 2019).

[17] Jung, D. and Tsiotras, P., "A 3-dof experimental test-bed for integrated attitude dynamics and control research," in [*AIAA Guidance, Navigation, and Control Conference and Exhibit*], (08 2003).

[18] Meissner, D. M., "A three degrees of freedom test-bed for nanosatellite and cubesat attitude dynamics, determination, and control," (2009).

[19] Chesi, S., Gong, Q., Pellegrini, V., Cristi, R., and Romano, M., "Automatic mass balancing of a spacecraft three-axis simulator: Analysis and experimentation," *Journal of Guidance, Control, and Dynamics* **37**, 197–206 (01 2014).

[20] Kim, J. J. and Agrawal, B., "Automatic mass balancing of air-bearing-based three-axis rotational spacecraft simulator," *Journal of Guidance Control and Dynamics - J GUID CONTROL DYNAM* **32**, 1005–1017 (05 2009).

[21] Young, J., "Balancing of a small satellite attitude control simulator on an air bearing," (1998).

[22] Sharifi, G. and Zabihian, E., "An effective approach to identify the mass properties of a satellite attitude dynamics simulator," *Australian Journal of Mechanical Engineering* **18**(3), 245–254 (2020).

[23] Bahu, A. and Modenini, D., "Automatic mass balancing system for a dynamic cubesat attitude simulator: development and experimental validation," *CEAS Space Journal* **12**, 597–611 (Dec 2020).

[24] Cardoso da Silva, R., Rodrigues, U., Borges, R., Sampaio, M., Beghelli, P., Costa, S., Popov, B., Battistini, S., and Cappelletti, C., "A test-bed for attitude determination and control of spacecrafts," in [*II Latin American IAA CubeSat Workshop*], (02 2016).

[25] Cardoso da Silva, R., Borges, R., Battistini, S., and Cappelletti, C., "Center of mass compensation of a nanosatellite testbed based on the extended kalman filter," in [*4th IAA Conference on University Satellite Missions and CubeSat Workshop*], (12 2017).

[26] Carrara, V., Oliveira, A., and Kuga, H., "A gas bearing platform attitude control for assessment of aocs systems," *Advances in the Astronautical Sciences* **152** (01 2014).

[27] Xuan, H., Chemori, A., Anh, T., Xuan, H., Hoai, T., and Viet, P., "From pid to l1 adaptive control for automatic balancing of a spacecraft three-axis simulator," *International Journal of Emerging Technology and Advanced Engineering* **06**, 77–86 (01 2016).

[28] McCafferty-Leroux, A., Newton, A., and Gadsden, S. A., "An improved nanosatellite attitude control simulator for experimental research," in [*Sensors and Systems for Space Applications XVI*], **12546**, 43–52, SPIE (2023).

[29] Markley, F. L. and Crassidis, J. L., [*Fundamentals of spacecraft attitude determination and control*], vol. 1286, Springer (2014).

[30] Cardoso da Silva, R., *Filtering and adaptive control for balancing a nanosatellite testbed*, PhD thesis, University of Brasilia (July 2018).

[31] Xu, Z., Qi, N., and Chen, Y., "Parameter estimation of a three-axis spacecraft simulator using recursive least-squares approach with tracking differentiator and extended kalman filter," *Acta Astronautica* **117**, 254–262 (2015).

[32] Hill, E., Newton, A., Gadsden, S. A., and Biglarbegian, M., "Tube-based robust model predictive control for fault tolerance," *Mechatronics* **95**, 103051 (2023).

[33] Newton, A., Hill, E., Gadsden, S., Biglarbegian, M., and Yang, S., "Investigating reaction wheel configuration and control law pairings for cubesats in the presence of faults," (06 2020).

[34] Newton, A., Hill, E. D., Gadsden, S. A., and Biglarbegian, M., "Development of a nanosatellite attitude control simulator for ground-based research," in [*Sensors and Systems for Space Applications XV*], Chen, G. and Pham, K. D., eds., **12121**, 1212106, International Society for Optics and Photonics, SPIE (2022).

[35] Julier, S. and Uhlmann, J., "Unscented filtering and nonlinear estimation," *Proceedings of the IEEE* **92**(3), 401–422 (2004).

[36] Newton, A., *Design, Development, and Experimental Validation of a Nanosatellite Attitude Control Simulator*, PhD thesis, University of Guelph (2021).