CEAS Space Journal (2020) 12:597–611
https://doi.org/10.1007/s12567-020-00309-5

ORIGINAL PAPER

Automatic mass balancing system for a dynamic CubeSat attitude simulator: development and experimental validation

Anton Bahu[1] · Dario Modenini[1]

Received: 16 January 2020 / Revised: 16 March 2020 / Accepted: 17 March 2020 / Published online: 15 April 2020

Abstract

This paper describes the automatic balancing and inertia identification system for three degrees of freedom CubeSat attitude simulator testbed. For a reliable verification of the attitude determination and control subsystem, the on-orbit environment shall be simulated within the testbed, minimizing the external disturbances acting on the satellite mock-up. The gravity torque is expected to be the largest among the disturbances, and an automatic balancing procedure can largely reduce the time necessary for tuning the platform and minimize the residual torque. The automatic balancing system adopted in this work employs three sliding masses independently actuated by three electric motors using a two-step procedure. In the first step, a feedback control is employed for a plane balancing. The inertia parameters and the remaining offset component are then estimated by collecting free oscillating platform data. This two-step procedure is iterated towards increasingly finer balancing until no further improvement is obtained. For the planar balancing, a control law based on linearized equations and a newly developed nonlinear feedback law is implemented and compared, showing the superior performance of the latter. The unbalance offset vector component along the local vertical and inertia tensor are estimated by a constrained batch least squares filter. Experimental results show the effectiveness of the implemented approach, which leads to a residual disturbance torque acting on the balanced platform smaller than 5×10^{-5} Nm.

1 Introduction

The growing interest for the development of highly capable nanosatellites for demanding scientific missions [1–3] is justified by much lower cost, low communication latency, low energy consumption, and high fault tolerance (several nanosatellites can be employed at the same cost to achieve redundancy) [4, 5]. Components miniaturization within nanosatellites is particularly challenging in case of the Attitude and Determination Control Subsystem (ADCS). ADCS stabilizes and orients the spacecraft despite external disturbance torques acting on it and low performance and reliability of the ADCS are potentially limiting and risk factors for future nanosatellite missions [6–8]. As part of the development and verification of an ADCS, careful pre-flight performance assessment through extensive hardware and software in-the-loop ground testing is of paramount importance [9, 10]. As a result, low-cost simulators for the attitude motion of CubeSat platforms, especially in the size-range from 1 to 3 U [11, 12], are becoming increasingly popular [13].

To simulate the on-orbit environment, the ground test facility shall allow for disturbance-free rotational dynamic of the satellite mock-up under test. A class of simulators that offers nearly torque-free three-degrees of freedom motion is the one based on articulated stands with air bearings [14–19], therefore, they have been used extensively in the past decades (see [20] for a historical review of the existing air-bearing based solutions). For such facilities, only few attempts to show ground-to-flight simulations traceability are reported [21, 22], highlighting the challenges for achieving reliable on-ground simulations. Indeed, the external torque acting on a nanosatellite in Low Earth Orbit can be as low as 10^{-6}Nm [23, 24]: reducing the overall torque down to this range is perhaps the biggest challenge for a dynamic ADCS simulator targeted to this satellite class.

Disturbance torques in air-bearing based simulators have different sources: the environment (aerodynamic, gravitational), the air bearing itself (friction), the supporting platform (anisoelastic) and torques arising from the systems

✉ Dario Modenini
dario.modenini@unibo.it

Anton Bahu
anton.bahu2@unibo.it

[1] Department of Industrial Engineering, University of Bologna, Via Fontanelle 40, 47121 Forlì, Italy

installed on the platform [17]. One of the largest disturbance torques affecting spherical air-bearing test benches is the one due to gravity [14, 16, 25, 26]. It arises due to the offset between the center of mass (CM) and the center of rotation (CR), which must be minimized. The simplest way to address this disturbance is by manual balancing: two horizontal axes are balanced first, then the centre of mass is raised along the vertical direction until the period of the pendulum-like motion becomes very large [27]. This method is potentially time consuming and has limited applicability because of the rotational constraints affecting the air bearing in one or two degrees of freedom (depending on the specific implementation [20]). In addition, manual balancing has limited accuracy when compared to automatic balancing systems [15, 16, 28, 29]. Indeed, for a lightweight simulator with a mass in the order of ~ 10 kg, automatic balancing can improve the offset compensation obtained manually (≈ 500 µm [28]) by up to two orders of magnitude [16, 30, 31]. At the same time, it allows saving volume and mass, at the cost of implementing a control system.

Automatic balancing involves estimation of the CM offset, usually by processing samples of angular motion of the testbed, and its compensation by actuated shifting masses providing three-dimensional mass displacement. If additional actuators are available, such as momentum exchange devices, the unbalance can be estimated through input–output data processing, by measuring the angular rate response of the system to known torques [16, 18, 19, 32]. In case no adjustable control torque input is available, the solution to the system balancing has been approached whether through a least-squares formulation based on the dynamic model [25, 26] or using a two-stage procedure. In the latter case, a first step consists of in-plane balancing through a feedback loop which compensates two out of three offset vector components [16, 33, 34]. To this end, linear [16], adaptive [34], and non-linear [14] control laws were proposed. The residual vertical unbalance is then identified and compensated in a second step by sampling the platform free oscillations [34].

Platform balancing is intrinsically related to the estimation of system inertia since this last is required to be known for balancing. Several solutions have been proposed for inertia identification, which employs constrained least squares [35], nonlinear Kalman Filters [36], conservation of angular momentum [37] or input–output data processing [18, 32].

Experimental verification of the balanced platform, when available, shows limited efficiency in terms of disturbance torque reduction. Even when employing high-performance sensors and a stiff mechanical structure, reported CM residual offsets are in the order of 5–10 µm [16, 30, 31]. For nanosatellite testing, where the total rotating mass is in the order of 10 kg, such offset would lead to disturbance torques up to 10^{-3} Nm, which are unacceptably high. These results indicate that an extensive verification campaign is mandatory for assessing the a-posteriori effectiveness of any balancing system.

Structure sagging and estimation errors are often ascribed as the main causes limiting the performance of automatic balancing systems [17]. Estimation of CM offset and system inertia normally requires the availability of angular acceleration [17–19, 25], which is obtained by numerical differentiation of angular rates measurements, thus being greatly affected by errors such as white noise, drift, and random walk affecting gyroscope data. Derivative free formulations have been also developed [26, 32], which guarantee some degree of noise filtering thanks to the integration of data, but exhibit poor performance in case of unmodelled disturbances torques. Note that measurement errors limit the balancing accuracy regardless of the estimation algorithm: comparison between EKF and batch least-squares method shows similar performance [38]. To reduce the impact of data noise, processing techniques such as Savitzky–Golay Filtering (SGF) [39] and tracking differentiator (TD) [32], have been successfully employed. Finally, parameters estimation is further degraded by the constraints to the testbed rotation and due to actuators errors. These were also shown to lead to inconsistent estimates when using different identification algorithms throughout the experiments [19].

A dynamic attitude testbed for nanosatellites is currently under commissioning at the University of Bologna: this manuscript reports on the development and experimental testing of the automatic balancing system for this facility. Strict budget constraints led our design choices towards reduced complexity, making use of low-end off-the-shelf hardware. Inspired by existing solutions, the proposed automatic balancing system employs three sliding masses whose locations are independently controlled and adjusted to eliminate the unknown offset between the center of rotation and the center of mass. This is achieved through a two-step procedure, with the first step devoted to in-plane balancing, and the second one to dynamics parameters identification. Both steps require attitude quaternions and angular velocity values, which are made available thanks to an inertial measurement unit (IMU) mounted on the platform.

Since the torque that can be generated by the balance masses is physically confined in the direction perpendicular to the gravity field, the disturbance torque acting on the same subspace can be compensated by a feedback law. To this end, a nonlinear control law is proposed for in-plane balancing, and compared to classic linear feedback. Afterwards, the residual unbalance along the local vertical is estimated, jointly with the inertia parameters of the platform, by sampling its free oscillations. Sampled data are processed offline through a batch least squares (LSQ) algorithm, implemented in MATLAB environment. The entire procedure is then iterated to incrementally refine the unbalance compensation, assuming the inertia known from the first iteration estimate. The effectiveness of the developed procedure is evaluated by estimating the disturbance torque acting on the free oscillating platform after balancing, from inspection of the angular momentum variation.

As the main contribution, this work provides a cost-effective solution to the balancing of nanosatellite class air bearing attitude simulators, which is capable to reduce the residual torque to less than 5×10^{-5}Nm. This objective is reached by comparing, through extensive numerical and experimental verifications, alternative control techniques and data processing strategies, and by combining them to maximize the performances, given the available hardware. As a result, this study provides guidelines that will (hopefully) prove useful for future cost-driven implementations of similar systems.

The manuscript is organized as follows. Automatic balancing system hardware and mechanical design overview are given in Sect. 2. Mathematical model of the system, control and estimation algorithms is outlined in the Sect. 3. The performance of the in-plane balancing and of the estimation algorithms are analyzed in Sect. 4, alone and combined in an iterative procedure. The developed balancing system, as implemented on the testbed platform and the results of the experimental verification, are provided in Sect. 5. Conclusions are stated in Sect. 6.

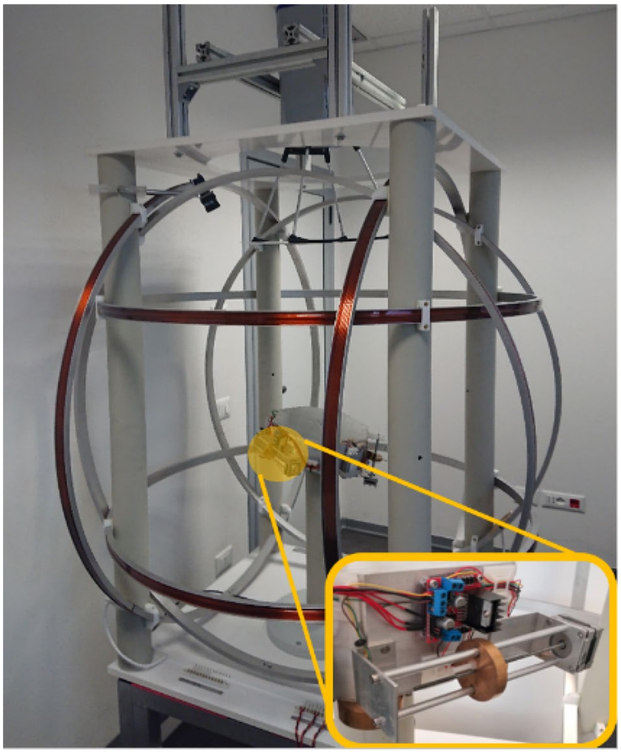


Fig. 1 Three-degree of freedom simulator testbed and automatic balancing system

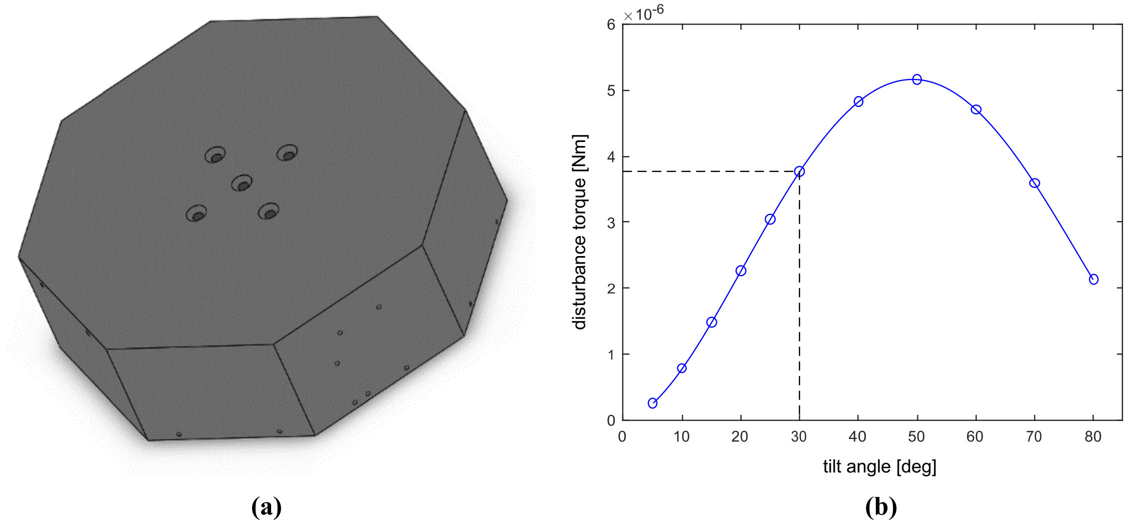
2 Testbed platform and automatic balancing system design

The automatic balancing system (ABS) is implemented as part of a three-degrees of freedom simulator testbed for ADCS verification of 3U CubeSats (see Fig. 1), developed at the Microsatellite and Space Microsystems Lab of the University of Bologna [40]. The core of the testbed consists of a table-top air-bearing platform, with custom design, whose function is to hold the nanosatellite mock-up under test; the automatic balancing system with shifting masses actuated by linear motors are installed on the platform. Other subsystems include a Helmholtz cage for geomagnetic field simulation, a Sun simulator, and a metrology vision system for ground-truth attitude measurement. The tabletop air bearing design provides three rotational degrees of freedom with angular excursions of 360° in yaw and maximum allowed tilt angle of $\approx 30°$.

The unbalance torque due to the offset between the center of mass and the center of rotation can be cancelled only partially by design and, even if static unbalance torque is successfully reduced, sagging gives origin to deformation of the platform which may lead to a significant disturbance torque [16, 22]. The structural deformation of the rotating platform due to its own weight is challenging to compensate, since making a structure stiffer implies a heavier structure. Therefore, the first step towards the development of the ABS for our facility was the mechanical design of the rotating platform, which supports the satellite mock-up and all the components necessary for the testbed operations.

The design was driven by a structural analysis aimed at minimizing the platform weight and inertia while limiting the torque due to sagging to the order of 10^{-6}Nm: the outcome of such process led to an aluminum tabletop platform, made of a thin octagonal plate with sidewalls, and equipped with radial elements to enhance flexural stiffness, see Fig. 2a).

The selected design was then verified through FEM simulations. The numerical model included the octagonal geometry with the stiffening elements, the payload modelled as 3U CubeSat with uniform mass distribution placed on the top of platform and balancing system components, modelled as point masses located at the bottom of the sidewalls, as this is the most conservative assumption. Simulations were repeated at various tilt angles, computing the center of the mass shift of the structure due to deformation. The anisoelastic torque acting on the platfrom with respect the tilt angle estimated through the FEM simulations is shown in Fig. 2b): the maximum torque is expected to be lower than 4×10^{-6}Nm at the maximum allowable tilt angle. This value is in line with the target requirement set by the expected disturbance torques encountered by nanosatellites in LEO.



(a) (b)

Fig. 2 **a** 3D CAD model of the platform; **b** sagging torque as a function of the tilt angle

Table 1 Balancing system parameters

m_{tot} (including 3U CubeSat mock-up)	11 kg
$m_{b,i}$	0.11 kg
$\Delta d_{i,\min}$	0.002 mm
$\Delta d_{i,\max}$	150 mm (110 mm along vertical)

2.1 Mass balancing system design

The amount of shifting balance masses $m_{b,1}, m_{b,2}, m_{b,3}$ and their maximum displacement $\Delta \boldsymbol{d}_{\max}$ determine the maximum CR to CM offset, $\boldsymbol{r}_{\text{CM,max}}$, that can be compensated: the ABS system is sized to compensate a residual maximum unbalance of $\|\boldsymbol{r}_{\text{CM,max}}\| \cong 2\text{mm}$ for a 4 kg payload; any possibly larger initial unbalance, is compensated manually using counterweights. Conversely, the resolution of the balance masses displacement, $\Delta \boldsymbol{d}_{\min}$, defines the lower bound of the offset vector the system can compensate:

$$\Delta \boldsymbol{r}_{\text{CM},i} = \frac{m_{b,i} \times \Delta \boldsymbol{d}_{\min}}{m_{\text{tot}}} \tag{1}$$

where m_{tot} is the mass of the whole platform and $\Delta \boldsymbol{r}_{\text{CM},i}$ is the smallest displacement that $\boldsymbol{r}_{\text{CM}}$ may undergo along the i th mass shifting direction. The balance masses displacement directions are, in turn, assumed to be mutually orthogonal. The design parameters are provided in Table 1. Note that in case of perfect balancing, i.e., CR-to-CM offset reduced down to Eq. (1), the residual gravity torque magnitude would be $\|\boldsymbol{T}_{D,\max}\| = \|\Delta \boldsymbol{r}_{\text{CM}} \times m_{\text{tot}} \boldsymbol{g}\| < 3.7 \cdot 10^{-6}\text{Nm}$. This is, of course, an ideal lower limit: backlash in the mechanical system, axes misalignment, and measurement errors unavoidably deteriorate the system performance, so that the expected residual torque after balancing will be higher than the above limit.

2.2 Automatic balancing control system design

The core of the ABS is built upon low-cost Arduino components (see Fig. 3). Three stepper motors are employed to control the position of the balancing masses. The motors are driven by an Arduino-based control system, with a dedicated power supply. The controller algorithms run on an Arduino Due board, which exchange data with a PC server through a Wi-Fi connection. Measurements are provided by

Fig. 3 Architectural diagram of the automatic balancing system [40]

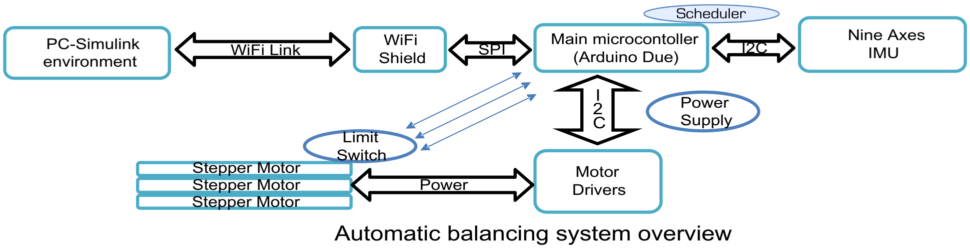


Table 2 IMU technical data

Accelerometer	
0-g offset temperature drift	3.5 mg/K
0-g offset supply volt. drift	2.5 mg/V
Output noise density	190 µg/Hz$^{0.5}$
Sensitivity	1 mg
Cross axis sensitivity (max)	2%
Gyroscope	
0-Ω offset change over temperature	0.03°/s per K
0-Ω offset supply volt. drift	0.1°/s per V
Output noise rms, BW = 47 Hz	0.3°/s
Sensitivity	0.0625°/s
Cross axis sensitivity (max)	3%

an Arduino 9-Axes IMU based on the Bosch BNO055 Absolute Orientation Sensor (see Table 2).

The IMU embeds proprietary filtering algorithms whose output is the attitude represented whether through a Roll–Pitch–Yaw (RPY) rotation sequence or as a quaternion. As will be discussed in Sect. 3, in-plane balancing requires whether the feedback of the tilt angles (in case of a linearized feedback) or the gravity vector in the body reference frame (in case of nonlinear feedback). The gravity vector is provided by filtered accelerometers measurements or, alternatively, can be computed from the estimated quaternion. According to the error sources listed in Table 2, a worst-case acceleration static measurement error is of 0.025 g, which would lead to tilt angles estimation errors of about 0.15°. On the other hand, quaternion accuracy is not stated in the datasheet: an experimental assessment showed static repeatability of quaternion readings of 0.7° RMS. Parameters estimation requires the body-frame gravity vector and gyroscope data measurements, which are also retrieved from the IMU.

3 Mathematical framework for automatic balancing and system identification

As a preliminary step towards the development of the algorithms for the ABS, we outline the mathematical framework for modelling the system dynamics, which is in common to other existing works [17, 26, 34]. The rotating platform is assimilated to a rigid body equipped with linearly moving point masses. The CR is fixed to a point in the inertial coordinate system $\mathcal{F}_i$, whose z axis is taken as the local vertical. The IMU is aligned with the platform reference frame $\mathcal{F}_\mathrm{b}$, centered at CR, and the balance masses can move only along a set of mutually orthogonal unit axes defined by $\mathcal{F}_\mathrm{a}$, fixed with respect to the platform body reference one [Fig. 4 Platform (Body) reference frame].

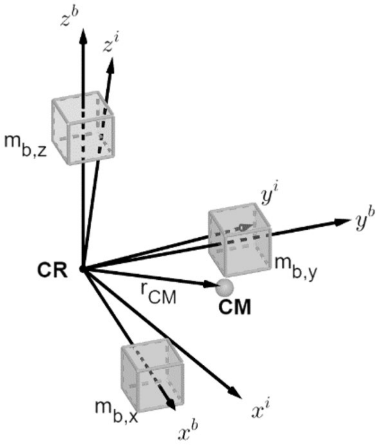


Fig. 4 Platform (body) reference frame

The relative orientation between frames is described by quaternion representation. The rotational kinematics of the platform are a function of the absolute angular velocity $\boldsymbol{\omega}^b = \left[\, \omega_x^b \;\; \omega_y^b \;\; \omega_z^b \,\right]^T$ and attitude describing quaternion vector $\boldsymbol{q}^b = \left[\, q_q^b \;\; q_2^b \;\; q_3^b \;\; q_4^b \,\right]^T$. In the following, with some abuse of notation, we will refer to these vectors as $\boldsymbol{q} = \left[\, q_1 \;\; q_2 \;\; q_3 \;\; q_4 \,\right]^T = \left[\, \boldsymbol{q}_e^T \;\; q_4 \,\right]^T$ and $\boldsymbol{\omega} = \left[\, \omega_x \;\; \omega_y \;\; \omega_z \,\right]^T$. The rotation matrix R_i^b, referred to simply as R in the following, represents the attitude of the body frame with respect to an inertial frame and can be expressed as a function of the quaternion:

$$R = \begin{bmatrix} 1 - 2\left(q_2^2 + q_3^2\right) & 2\left(q_1q_2 - q_4q_3\right) & 2\left(q_1q_3 + q_4q_2\right) \\ 2\left(q_1q_2 + q_4q_3\right) & 1 - 2\left(q_1^2 + q_3^2\right) & 2\left(q_2q_3 - q_4q_1\right) \\ 2\left(q_1q_3 - q_4q_2\right) & 2\left(q_2q_3 + q_4q_1\right) & 1 - 2\left(q_1^2 + q_2^2\right) \end{bmatrix} \tag{2}$$

The gravity vector in body reference frame $\boldsymbol{g}$, is expressed as:

$$\boldsymbol{g} = g\hat{\boldsymbol{g}} = gR\boldsymbol{e}_3 \tag{3}$$

where g is gravity vector magnitude and $\boldsymbol{e}_3 = \left[\, 0 \;\; 0 \;\; 1 \,\right]^T$. The rotational kinematics are:

$$\begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \dot{q}_3 \\ \dot{q}_4 \end{bmatrix} = \frac{1}{2} \begin{bmatrix} 0 & \omega_z & -\omega_y & \omega_x \\ -\omega_z & 0 & \omega_x & \omega_y \\ \omega_y & -\omega_x & 0 & \omega_z \\ -\omega_x & -\omega_y & -\omega_z & 0 \end{bmatrix} \begin{bmatrix} q_1 \\ q_2 \\ q_3 \\ q_4 \end{bmatrix} \quad \text{or} \quad \begin{array}{l} \dot{\boldsymbol{q}}_e = \frac{1}{2}\left(q_4\boldsymbol{\omega} + \boldsymbol{q}_e \times \boldsymbol{\omega}\right) \\ \dot{q}_4 = -\frac{1}{2}\boldsymbol{\omega}^T\boldsymbol{q}_e \end{array} \tag{4}$$

Following [17], we assume that the three shifting masses are equal $m_{b,1} = m_{b,2} = m_{b,3} = m_b$ can be independently actuated along with three mutually orthogonal directions, $\left[\boldsymbol{u}_1\boldsymbol{u}_2\boldsymbol{u}_3\right]$, which are aligned with the body frame axes. Let the $\boldsymbol{r}_{b,i}$ be the absolute position of mass $m_{b,i}$ with respect to CR. Due to the definition of $\boldsymbol{u}_i$, a displacement Δd_i of any of the three masses will induce a net shift of the overall platform CM equal to:

$$\Delta \boldsymbol{r}_{\mathrm{CM}} = \frac{1}{m_{\mathrm{tot}}} \sum_{i=1}^{3} m_{b,i} \Delta d_i \boldsymbol{u}_i = \frac{m_b}{m_{\mathrm{tot}}} \sum_{i=1}^{3} \Delta d_i \boldsymbol{u}_i = \frac{m_b}{m_{\mathrm{tot}}} \Delta \boldsymbol{r}_b \quad (5)$$

where $\boldsymbol{r}_b = \left[\boldsymbol{r}_{b,1} \ \boldsymbol{r}_{b,2} \ \boldsymbol{r}_{b,3} \right]^T$. The total inertia matrix J is the sum of the platform inertia with respect to the body reference frame without balance masses J_s plus the contribution due to the masses themselves, assumed to be punctiform:

$$J = J_S + \sum_{i=1}^{3} \left(-m_{b,i} \left[\boldsymbol{r}_{b,i} \times \right] \left[\boldsymbol{r}_{b,i} \times \right] \right) \quad (6)$$

where $[\boldsymbol{r}_{b,i}\times]$ is the cross-product matrix function of the position vector of the ith balance mass. Following [34] the rotational dynamics of the simulator about CR, subject to gravity torque can be expressed as following:

$$\begin{aligned} J\dot{\boldsymbol{\omega}} &= -\omega \times J\omega + m_{\mathrm{tot}} g \boldsymbol{r}_{\mathrm{CM}} \times R^T \boldsymbol{e}_3 + \boldsymbol{\tau}_u, \\ \tau_u &= m_b g \boldsymbol{r}_b \times R^T \boldsymbol{e}_3 \end{aligned} \quad (7)$$

Note that Eq. (7) does not account for the angular momentum variation due to the temporal derivative of J, which is assumed to be negligible, due to the small stepper motors displacement speed. The only control torque available is due to the balancing masses, which is clearly constrained to lie in the direction normal to both the masses position vector and to the gravitational field direction. Once a control law is designed which provides a stabilizing $\boldsymbol{\tau}_u$, the control mass displacement components $\boldsymbol{r}_b$ required to generate the desired torque can be computed according to [34]:

$$\boldsymbol{r}_b = \frac{\boldsymbol{g} \times \boldsymbol{\tau}_u}{\|\boldsymbol{g}\|^2 m_b} \quad (8)$$

Given the above mathematical model, our objective is to develop a) a feedback control law for in-plane horizontal balancing of the platform, and b) an identification strategy for estimating the residual vertical offset plus system inertia.

The goal of the in-plane balancing control problem is to align the body axis $\hat{z}$ to the local vertical $\hat{\boldsymbol{g}}$. If the body axes dynamics are sufficiently slow, the dynamic equations can be decoupled through linearization near the origin, and a controller can be designed by classical control tools, using either $\boldsymbol{g}$ or $\boldsymbol{q}$ as feedback variables: this is the path pursued in Sect. 3.1. To overcome the limits of the linearized model, a nonlinear control law is proposed which solves the underactuation by projecting the control law on the subspace orthogonal to the gravity direction, as discussed in Sect. 3.2.

3.1 Decoupled dynamics and linear control law

Let assume the inertia matrix is diagonal with respect to the chosen body reference frame, $J = \mathrm{diag}(J_x, J_y, J_z)$. The dynamics equations can now be rewritten as [41]:

$$\begin{aligned} J_x \dot{\omega}_x^b + \left(J_z - J_y\right) \omega_y^b \omega_z^b &= m_{\mathrm{tot}} \left(r_{\mathrm{CR},y} g_z^b - r_{\mathrm{CR},z} g_y^b \right) + \tau_{u,x} \\ J_y \dot{\omega}_y^b + \left(J_x - J_z\right) \omega_x^b \omega_z^b &= m_{\mathrm{tot}} \left(r_{\mathrm{CR},z} g_x^b - r_{\mathrm{CR},x} g_z^b \right) + \tau_{u,y} \\ J_z \dot{\omega}_z^b + \left(J_y - J_x\right) \omega_x^b \omega_y^b &= m_{\mathrm{tot}} \left(r_{\mathrm{CR},x} g_y^b - r_{\mathrm{CR},y} g_x^b \right) + \tau_{u,z} \end{aligned} \quad (9)$$

The disturbance torque due to $\boldsymbol{r}_{\mathrm{CM}}$ needs to be perfectly compensated by the controller and is considered to be slowly varying. Only the first two equations can be considered since $\tau_{u,z}$ is null in the target position. Let $\left[\phi \ \theta \ \psi \right]^T$ be the RPY attitude representation, where ϕ is the roll angle, θ is the pitch angle and ψ the yaw angle. We define the output error vector $\boldsymbol{y}_{\mathrm{e}} = \left[\phi \ \theta \right]^T$. The components of gravity vector can be computed as:

$$g_x^b = \boldsymbol{g}^b \cos\theta, \; g_y^b = \boldsymbol{g}^b \cos\phi, \; g_z^b = \sqrt{\boldsymbol{g}^{b2} - \left(g_x^b\right)^2 - \left(g_y^b\right)^2} \quad (10)$$

If small angles and rates are assumed, the nonlinear dynamics described by Eq. (9) can be linearized. In this case, the control problem can be solved by two controllers PID$_x$ and PID$_y$, designed to stabilize the following second-order dynamic equations:

$$\begin{aligned} J_x \ddot{\phi} &= m_{\mathrm{tot}} \left(r_{\mathrm{CM},y} g_z^b - r_{\mathrm{CM},z} g_y^b \right) + \mathrm{PID}_x(\phi) \\ J_y \ddot{\theta} &= m_{\mathrm{tot}} \left(r_{\mathrm{CM},z} g_x^b - r_{\mathrm{CM},x} g_z^b \right) + \mathrm{PID}_y(\theta) \end{aligned} \quad (11)$$

These equations can be asymptotically stabilized by PID controllers.

For small-angle approximation, $\theta = \frac{g_x^b}{\boldsymbol{g}^b}$ and $\phi = \frac{g_y^b}{\boldsymbol{g}^b}$. This means it is a reasonable assumption to use as feedback $\boldsymbol{y}_{\mathrm{e}} = \left[\frac{g_y^b}{\boldsymbol{g}^b} \ \frac{g_x^b}{\boldsymbol{g}^b} \right]^T$ for small tilt angles, due to limited angular excursions of the table-top design. Tilt angle estimation through acceleration feedback is expected to be more accurate, thus it will be employed in the plane balancing performance assessment in the following chapter.

At the equilibrium point, $g_x^b, g_y^b = 0$:

$$\begin{aligned} 0 &= \frac{1}{J_x} \left(m_{\mathrm{tot}} r_{\mathrm{CM},y} g_z^b + \mathrm{PID}_x \phi \right) \\ 0 &= \frac{1}{J_y} \left(-m_{\mathrm{tot}} r_{\mathrm{CM},x} g_z^b + \mathrm{PID}_y \theta \right) \end{aligned} \quad (12)$$

The above equations show that, in steady state, any plane unbalance $\left(r_{\mathrm{CM},x}, r_{\mathrm{CM},y} \right)$ results to a constant disturbance torque which must be compensated by the integral action.

Since the control action is done by balancing masses, the planar balancing is preserved until the balancing masses position holds.

3.2 Underactuated single-axis pointing for planar balancing

The main limitation of the linear approximation is its potential instability in case the feedback stabilizing effect is overcome by large nonlinearities [42]. Due to the slow actuators and unmodelled dynamics, low gains must be chosen to provide feasible requested control torque, leading possibly to large closed-loop time constant and great nonlinearities effect on the system trajectories. To overcome this issue, we propose a novel nonlinear control law, and we show that in case of an underactuated system, partial attitude control is still possible, with proved asymptotic stability in absence of disturbances. The practical usefulness in the presence of an unbalance torque will be then verified both through simulations and experiments.

Our objective is to drive the system to a pure spin condition with the z-body axis aligned to the local vertical direction. We will approach the problem as the one of a single-axis pointing, where the pointing direction is coincident with the under-actuated one.

We first analyze the disturbance-free single-axis pointing case, i.e., when $\boldsymbol{r}_{\mathrm{CM}} = 0$: from a physical standpoint, this means considering a balanced platform which is tilted with respect to the horizontal plane. To develop a stabilizing control law, we consider the candidate Lyapunov function:

$$V = \frac{1}{2}\boldsymbol{\omega}^T J\boldsymbol{\omega} + \frac{1}{2}K_{\mathrm{p}}(\hat{\mathbf{g}} - \hat{\mathbf{z}})^T(\hat{\mathbf{g}} - \hat{\mathbf{z}}) \tag{13}$$

where K_{p} is a positive scalar. V is positive definite and equate zero for the system at rest, with $\hat{\mathbf{z}}||\hat{\mathbf{g}}$. Along the trajectories of the system, the time derivative of the Lyapunov function is defined as follows:

$$\begin{aligned}\dot{V} &= \boldsymbol{\omega}^T J\dot{\boldsymbol{\omega}} - K_{\mathrm{p}}\hat{\mathbf{z}}^T\dot{\hat{\mathbf{g}}} = \boldsymbol{\omega}^T\left(\boldsymbol{\tau}_u - \boldsymbol{\omega}\times J\boldsymbol{\omega}\right) + K_{\mathrm{p}}\hat{\mathbf{z}}^T\boldsymbol{\omega}\times\hat{\mathbf{g}}\\ &= \boldsymbol{\omega}^T\boldsymbol{\tau}_u + K_{\mathrm{p}}\boldsymbol{\omega}^T\hat{\mathbf{g}}\times\hat{\mathbf{z}}\\ &= \boldsymbol{\omega}^T\left(\boldsymbol{\tau}_u + K_{\mathrm{p}}\hat{\mathbf{g}}\times\hat{\mathbf{z}}\right)\end{aligned} \tag{14}$$

In deriving Eq. (14), we made use of the vector kinematic equation $\dot{\hat{\mathbf{g}}} = -\boldsymbol{\omega}\times\hat{\mathbf{g}}$. Let us define a projection operator:

$$P_{\mathrm{p}} = \left[I - \hat{\mathbf{g}}\hat{\mathbf{g}}^{\mathrm{T}}\right] \tag{15}$$

It is then possible to decompose the angular speed vector by means of the projection operator as follows:

$$\boldsymbol{\omega}_{\mathrm{p}} = P_{\mathrm{p}}\boldsymbol{\omega},\ \boldsymbol{\omega}_{\mathrm{g}} = \boldsymbol{\omega} - \boldsymbol{\omega}_{\mathrm{p}} \tag{16}$$

We now consider the *admissible* control feedback:

$$\boldsymbol{\tau}_u = -K_{\mathrm{p}}\hat{\mathbf{g}}\times\hat{\mathbf{z}} - K_{\mathrm{d}}\boldsymbol{\omega}_{\mathrm{p}} \tag{17}$$

with K_d being a positive scalar. The control torque is defined on the plane orthogonal to $\hat{\mathbf{g}}$, so that the corresponding mass displacement $\boldsymbol{r}_b$ can always be computed through Eq. (8). The proposed feedback makes $\dot{V}$ semi-definite negative, according to:

$$\dot{V} = -K_{\mathrm{d}}\boldsymbol{\omega}^T\boldsymbol{\omega}_{\mathrm{p}} = -K_{\mathrm{d}}\left(\boldsymbol{\omega}_{\mathrm{p}} + \boldsymbol{\omega}_{\mathrm{g}}\right)^T\boldsymbol{\omega}_{\mathrm{p}} = -K_{\mathrm{d}}\boldsymbol{\omega}_{\mathrm{p}}^T\boldsymbol{\omega}_{\mathrm{p}} \tag{18}$$

Equation (18) indicates that the system will converge to the set $\boldsymbol{\omega}_{\mathrm{p}} = 0$, i.e. having angular velocity parallel to $\hat{\mathbf{g}}$. By inspection of the closed-loop dynamics, it can be shown that in the convergence set $\hat{\mathbf{g}}\times\hat{\mathbf{z}}$ must be null as well, which is the desired z axis pointing condition. A proof is given in Appendix A.

Equation (17) is suitable for reorienting a balanced platform, but when $\boldsymbol{r}_{\mathrm{CM}} \neq 0$, it fails to lead $\hat{\mathbf{g}}\times\hat{\mathbf{z}}$ to zero. It is well known that, when full actuation is available, a system subject to a bounded disturbance torque, such as the static unbalance we want to compensate, can be stabilized by adding an integral feedback [43]. To this end, we introduce the new state variable:

$$\boldsymbol{i}_{\mathrm{p}} = \int_0^t K_{\mathrm{p}}\hat{\mathbf{g}}\times\hat{\mathbf{z}}\mathrm{d}t \tag{19}$$

and modify the feedback law according to:

$$\boldsymbol{\tau}_u = -K_{\mathrm{p}}\hat{\mathbf{g}}\times\hat{\mathbf{z}} - K_{\mathrm{d}}\boldsymbol{\omega}_{\mathrm{p}} - K_{\mathrm{d}}K_{\mathrm{I}}\boldsymbol{i}_{\mathrm{p}} \tag{20}$$

The rationale lies in that if one can verify that the feedback control drives the integral variable to a finite limit, then $\hat{\mathbf{g}}\times\hat{\mathbf{z}}$ must have settled to zero, which in turn requires also $\boldsymbol{\omega}_{\mathrm{p}} = 0$ due to the kinematics of $\hat{\mathbf{g}}$. If the system converges to a pure spin around $\hat{\mathbf{z}}||\hat{\mathbf{g}}$, the only non-zero component of $\boldsymbol{r}_{\mathrm{CM}}$ must lie along the vertical axis, which means that planar balancing is achieved.

The validity of the above assumption and the practical effectiveness of Eq. (20) have been checked through simulations and experiments, as detailed in Sects. 4 and 5.

3.3 Inertia parameters and residual offset identification

Existing approaches for parameters identification in automatic balancing systems consider whether the joint estimate of the inertia matrix and CR-to-CM offset vector for a one-stage balancing [17, 26, 32] or the estimation of the vertical offset after planar balancing in a dual-stage procedure, assuming the inertia known [34]. Our application lies somehow in between those: after the balancing on the x–y plane is achieved, is possible to rely on the partial knowledge of the unbalance vector for a more accurate estimation of the last component of the offset vector and of the elements of the inertia tensor. Since in the ABS no external actuation is available other than the shifting masses, the approach used is based on sampling free oscillating rotations. The only torque acting on the system is, therefore, supposed to be the gravitational disturbance.

The identification problem can be easily cast in a linear least squares estimation framework such as:

$$H\boldsymbol{x} = \boldsymbol{b}\left(\boldsymbol{\tau}_{\text{ext}}\right) \tag{21}$$

where $\boldsymbol{\tau}_{\text{ext}}$ is the external torque, H is the observation matrix, and $\boldsymbol{x} = \left[\boldsymbol{j}^T\ \boldsymbol{r}_{\text{CM}}^T\right]^T = \left[J_{xx}\ J_{yy}\ J_{zz}\ J_{xy}\ J_{xz}\ J_{yz}\ r_{\text{CM},x}\ r_{\text{CM},y}\ r_{\text{CM},z}\right]^T$ is the vector of the dynamic parameters to be identified:$\boldsymbol{x}$ =. However, if no actuation is available, as for a freely oscillating platform, the right-hand side of Eq. (22) is identically zero, which would require solving equations for the null space of matrix H. This, in turn, would allow to estimate $\boldsymbol{x}$ only up to an unknown scaling of all its elements. A possible solution has been proposed in [34]: the authors show that such a drawback can be overcome when the dynamic parameters are computed with respect to a freely chosen point O, which differs from both CM and CR. Then, the six inertia parameters and three unbalance vector components are estimated jointly. This is not necessary, however, in our case, since we can rely on the partial knowledge of the unbalance vector provided by the planar balancing, after which $r_{\text{CM},x}r_{\text{CM},y}$ are ideally null. We thus developed a different solution to the identification problem, as follows. By applying a known shift to the x–y masses, some known $r_{\text{CM},x}r_{\text{CM},y}$ are generated according to Eq. (5), and the estimation problem can be reformulated as a constrained least-squares problem:

$$\begin{aligned} &H\boldsymbol{x} = \boldsymbol{0} \\ &\text{s.t.: } B\boldsymbol{x} = \boldsymbol{c} \end{aligned} \tag{22}$$

This way, the scaling ambiguity is removed. The system dynamics described by Eq. (7) can be written in the matrix framework of Eq. (22), as:

$$\left[\Omega(\dot{\boldsymbol{\omega}}) + [\tilde{\omega}]\Omega(\boldsymbol{\omega}) | m_{\text{tot}}[\tilde{g}]\right]\begin{bmatrix}\boldsymbol{j}\\ \boldsymbol{r}_{\text{CM}}\end{bmatrix} = \boldsymbol{0} \tag{23}$$

where Ω is a 3×6 matrix rearranging the elements of $\boldsymbol{\omega}$ or $\dot{\boldsymbol{\omega}}$ [17]. The constraints for the known values of $r_{\text{CM},x}r_{\text{CM},y}$ can be enforced by setting:

$$B = \begin{bmatrix} 0_{1\times 6} & 1 & 0 & 0 \\ 0_{1\times 6} & 0 & 1 & 0 \end{bmatrix},\ \boldsymbol{c} = \begin{bmatrix} r_{\text{CM},x} \\ r_{\text{CM},y} \end{bmatrix} \tag{24}$$

A drawback of this formulation lies in the need of the angular accelerations, which are computed from numerical differentiation of noisy angular rate samples, thus being potentially highly inaccurate. As the dynamics of the free-oscillating body is relatively slow, high-frequency random noise can be partially removed by filtering the data through e.g., a SGF.

As suggested in [17], a different formulation can be obtained by integrating Eq. (24). In this case, angular accelerations are no more required and data samples integration gives some degree of robustness with respect to the white noise:

$$\left[\Omega\left(\boldsymbol{\omega}^b\right) + \int [\tilde{\omega}]\Omega\left(\boldsymbol{\omega}^b\right) | m_{\text{tot}} \int [\tilde{g}]\right]\begin{bmatrix}\boldsymbol{j}\\ \boldsymbol{r}_{\text{CM}}\end{bmatrix} = 0 \tag{25}$$

In collecting the data for solving Eq. (25mulations adopting both the linear) one should trade-off between two counter-opposing requirements: on one side, collectmulations adopting both the linearing as many data as possible shall enhance the parameter observability. On the other hand, too long time frame may lead to a degraded estimate, due to the increasing impact of the unmodeled torques. The performance of two possible solutions will be compared through the simulations in the following section.

4 Simulation results

To evaluate the performance of the plane balancing and parameters estimation algorithms, and for tuning our experimental set-up, an accurate dynamic model of the testbed was developed in MATLAB environment. The simulator implements the mathematical model developed in Sect. 3. The ground-truth inertia matrix employed in the simulator is computed from the CAD model of the platform, using a dummy mass in place of the CubeSat mockup:

$$J_{\text{CR}} = \begin{bmatrix} 5.70 & 0.00 & 0.17 \\ 0.00 & 5.97 & 0.01 \\ 0.17 & 0.01 & 9.67 \end{bmatrix} \cdot 10^{-2}\ \text{kg/m}^2 \tag{26}$$

Angular rates, attitude and gravity data are corrupted by errors, according to the IMU characteristics in Table 2, before being fed to the control and estimation algorithms. The actuators are modelled by limiting the maximum rate and acceleration of the balancing mass displacement, in accordance to the limits experimentally found for the stepper motors. Mass displacement resolution is set equal to one motor step. Air friction is included as a disturbance torque (in addition to the gravity one) and assumed proportional to the square of the platform angular rate.

4.1 Plane balancing simulations

We performed simulations adopting both the linear feedback defined in Eq. (11), and the nonlinear feedback of Eq. (20). The initial platform angular speeds were assumed of magnitude up to about 30°/s; initial roll and pitch angles are supposed to be equal to 30°, which is the maximum allowable excursion.

The choice of the optimal gains for both controllers depend on the magnitude of the disturbance torque, here we report results of a fine balancing experiments starting from $\boldsymbol{r}_{\mathrm{CM}} = \left[6\ 4\ -7 \right] \cdot 10^{-5}$ m. The PID controller was tuned using a dedicated MATLAB built-in tool, by setting a bandwidth of 0.05 rad/s and a phase margin of 70°. The nonlinear controller was instead tuned heuristically. The adopted gains are summarized in Table 3.

The linear control law needs angular rates and tilt angles for balancing. Near the origin, where the inertial accelerations are negligible, more accurate accelerometer's readings can be employed as a feedback (see Sect. 2.2). For the nonlinear controller, only the gravity vector retrieved from the accelerometer is required. The attitude errors obtained with the nonlinear controller and the PID are depicted in Fig. 5.

The nonlinear controller guarantees faster convergence then the PID one, the steady state being reached in less than 100 s. Attempts to achieve faster convergence of the linear controller were made, by increasing gains magnitude, which, however, led to instability of the closed-loop system.

The offset components perpendicular to the $\hat{\mathbf{z}}$ axis, which is aligned with the $\hat{\mathbf{g}}$ axis at the steady state, are well compensated, with residual tilt angles in the order of 0.1° for both controllers. As a result, the unbalance vector can be effectively reduced down to 10^{-8} m.

Table 3 Controllers gains adopted for fine balancing

PID				Nonlinear controller	
K_{px}	$2.3{\cdot}10^{-5}$	K_{py}	$2.4{\cdot}10^{-5}$	K_p	$2{\cdot}10^{-2}$
K_{dx}	$2.6{\cdot}10^{-3}$	K_{dy}	$2.1{\cdot}10^{-3}$	K_d	$2{\cdot}10^{-2}$
K_{ix}	$1.5{\cdot}10^{-6}$	K_{iy}	$1.9{\cdot}10^{-6}$	K_i	$1{\cdot}10^{-1}$

4.2 Dynamics parameters estimation

The estimation of the residual unbalance vector component and inertia parameters relies on data collected from the simulation of free oscillations processed through a batch LSQ estimator. The two formulations introduced in Sect. 3.3, namely the standard formulation with the angular rate filtered through SGF, Eq. (23), and the derivative-free, integral formulation, Eq. (25), are compared. They are referred to as LSQ-SGF and LSQ-INT, respectively. In general, parameters observability from free oscillations is enhanced when the motion features high accelerations and angular rates.

Sample outputs from the two algorithms are depicted in Fig. 6, obtained assuming $r_{\mathrm{CM}} = 5 \cdot 10^{-5}$ m. $r_{\mathrm{CM},x}$ and $r_{\mathrm{CM},y}$ are arbitrary set after the plane balancing. The sampling time is of 0.05 s. Clearly, LSQ-SGF outperforms LSQ-INT: estimation errors on the principal inertia moments are around 1% for the former, while can be as high as 5% for the latter. Remarkably, both algorithms lead to an estimation error for $r_{\mathrm{CM},z}$ in the order of 10^{-6} m, which is not acceptable: an error of $\approx 10^{-8}$ m shall be obtained if we want to limit the residual disturbance down to 10^{-6} Nm. This result indicates that, due to the measurement errors and unmodelled dynamics, multiple iterations may be necessary before sufficiently accurate results can be achieved.

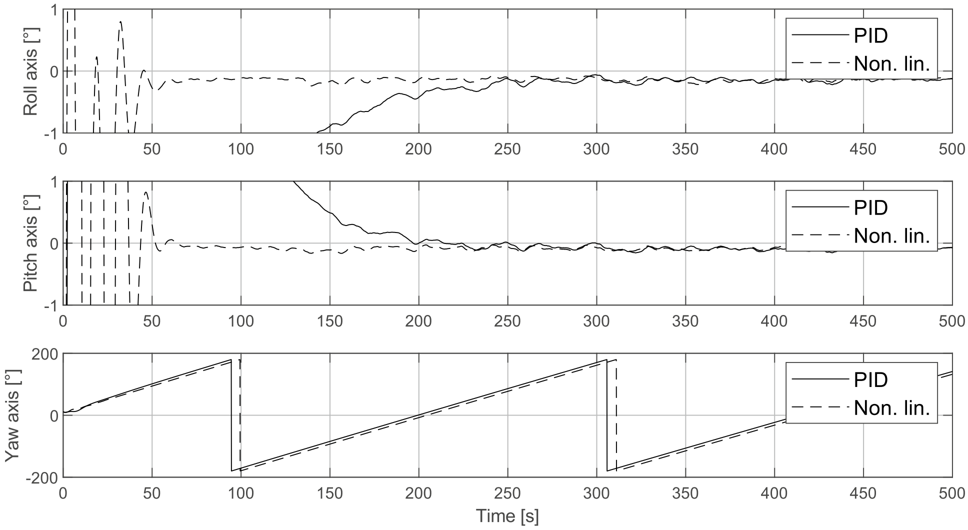


Fig. 5 Balancing control laws comparison: simulations

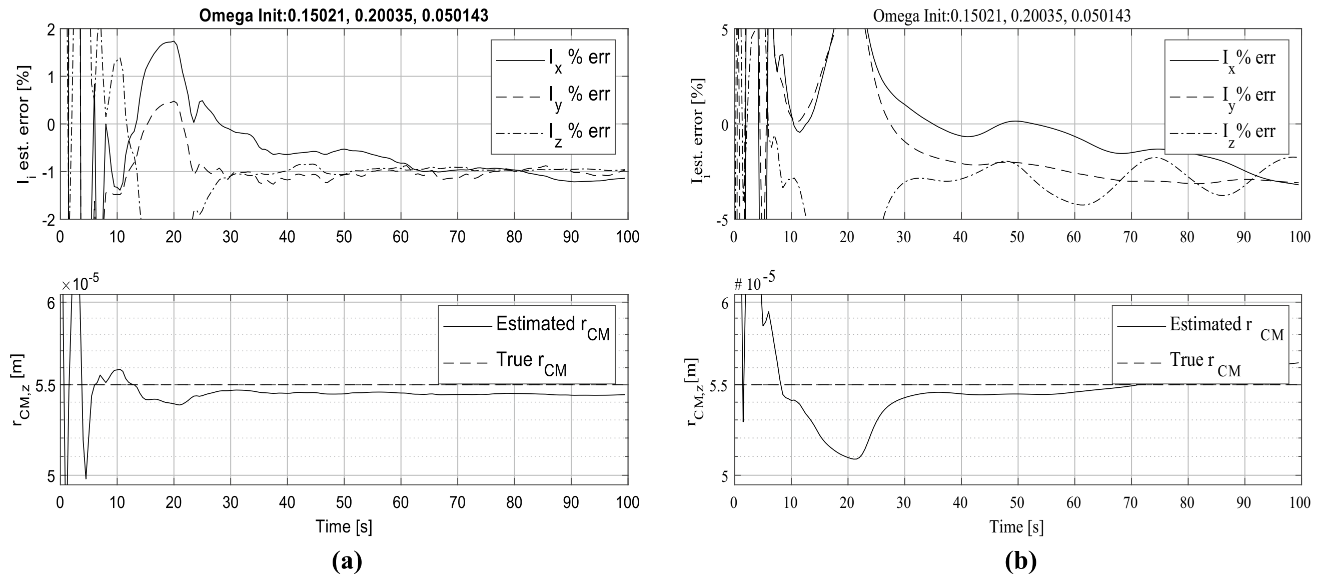


Fig. 6 LSQ estimation. **a** LSQ-SGF; **b** LSQ-INT (right panel)

Indeed, if the offset estimated with a first iteration is compensated by adjusting accordingly the position of the balancing masses, an additional, refined balancing iteration can be attempted. In doing so, however, we verified that the LSQ estimation as proposed in Eq. (23) and in Eq. (25) is no more effective when trying to estimate jointly the inertia and the residual unbalance. On the other hand, assuming the inertia known, additional constraints can be included to Eq. (24) for improving the estimate of the residual unbalance $r_{\mathrm{CM},z}$ only. An iterated parameter estimation stage can thus be envisaged: in a first iteration, the LSQ batch filter estimates both inertia and unbalance vector; for successive iterations, after the plane balancing is repeated, an LSQ fed with the inertia matrix estimated through the first stage can be employed for refined $r_{\mathrm{CM},z}$ estimation. The iterative approach improves the $r_{CM,z}$ estimation accuracy, with an error in the order of 10^{-8}m, as requested. The variation of the estimated $r_{\mathrm{CM},z}$ with respect to an increasingly longer data batch duration is shown in Fig. 7, where the inertia matrix is assumed to be known with an uncertainty of 2% and true $r_{\mathrm{CM},z}$ set to $5.5 \cdot 10^{-7}$ m.

In summary, the outcome of the simulations in Sects. 4.1 and 4.2 suggest that several balancing iterations, from coarser to finer, are necessary for achieving adequate performance: usually after three iterations the disturbance torque due to the unbalance is reduced down to $2 \cdot 10^{-6}$ Nm, with no further improvements.

4.3 Disturbance torque estimation

To verify the effectiveness of the balancing procedure, one can inspect the variation of the kinetic energy: indeed, for a perfectly balanced and dissipation-free system, the kinetic energy E_{kin} should be constant in time. Kinetic energy can be easily evaluated from the gyroscope measurements, provided the knowledge of the inertia matrix, through the formula $2E_{\mathrm{kin}} = \boldsymbol{\omega}^T \boldsymbol{J}\boldsymbol{\omega}$. The periodic variation due to the pendulum-like motion of the platform can be decoupled from the slow exponential decay due to the air-bearing friction. However, the disturbance torque due to the residual unbalanced cannot be directly estimated from the kinetic energy variance, and thus can be used only as additional tool for fast and reliable balancing accuracy evaluation.

The disturbance torque acting on the system can be estimated directly from the variation of the angular momentum, as per the dynamic equation:

$$\boldsymbol{\tau}_d = \boldsymbol{J}\dot{\boldsymbol{\omega}} + \boldsymbol{\omega} \times \boldsymbol{J}\boldsymbol{\omega} \tag{27}$$

For small angular speed, aerodynamic torque and torque due to air-bearing friction are negligible. The torque due to the interaction between the platform residual magnetic dipole and the Earth magnetic field can be compensated thanks to the available Helmholtz cage. In this case, the torque due to the unbalance can be considered as the main contributor to the overall $\boldsymbol{\tau}_d$.

Equation (27) requires differentiation of the angular rate measurements, which we performed using Savitzky–Golay filtering. The choice of data pre-filtering is justified by the very slow dynamics which is expected in case of small residual unbalance, so that high-frequency measurement noise can be effectively removed.

Numerical simulations showed this approach to be reliable for estimating torques down to the order of 10^{-6}Nm. In the case of smaller torques, the error due to the measurement

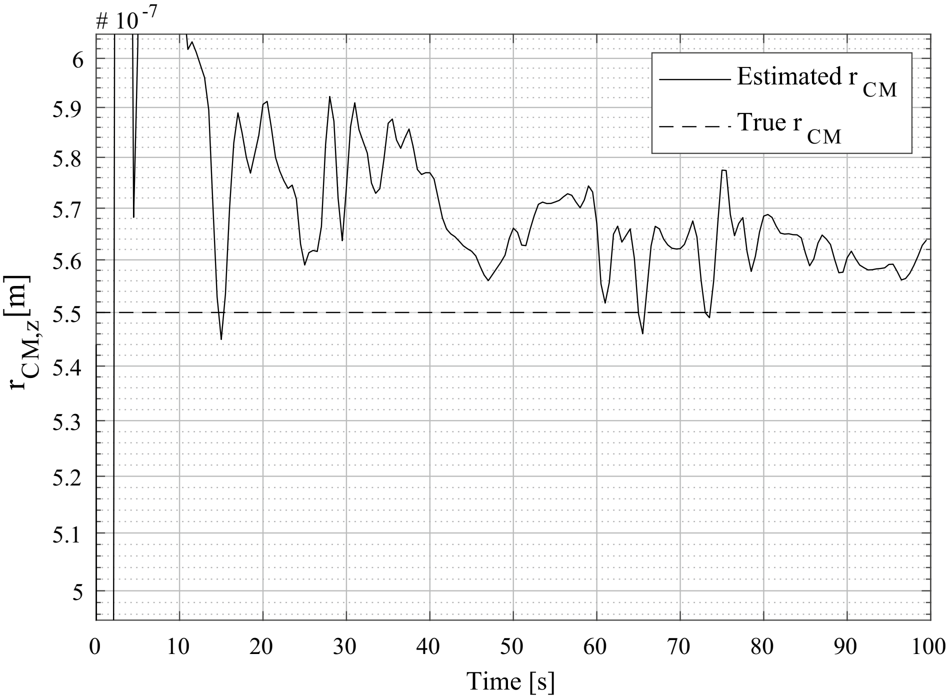


Fig. 7 Numerical simulation of the $r_{\mathrm{CM},z}$ estimation by constrained LSQ

errors and inertia matrix uncertainty renders the estimation unreliable. As an example, Fig. 8 compares the estimated disturbance torque computed according to Eq. (27) against the true one, for an assumed $\boldsymbol{r}_{\mathrm{CM}} = \begin{bmatrix} 4 & 4 & 14 \end{bmatrix} \cdot 10^{-8}$ m and with inertia moments uncertainty of 1%.

5 Experimental results

The proposed balancing procedure and disturbance torque estimation algorithm have been verified through experiments using the testbed platform described in Sect. 2. The first step, plane balancing, runs in real-time onboard the Arduino microcontroller. The parameter estimation, instead, is performed offline, by collecting the free-oscillation data through the IMU, with sampling frequency 20 Hz. Thanks to a Wi-Fi link, the data collected are sent to a desktop computer for processing.

As suggested by the numerical simulations, an iterative procedure has been found to be necessary to guarantee good balancing performance. In our experiments, each balancing iteration consists of an in-plane balancing, followed by a parameter estimation stage.

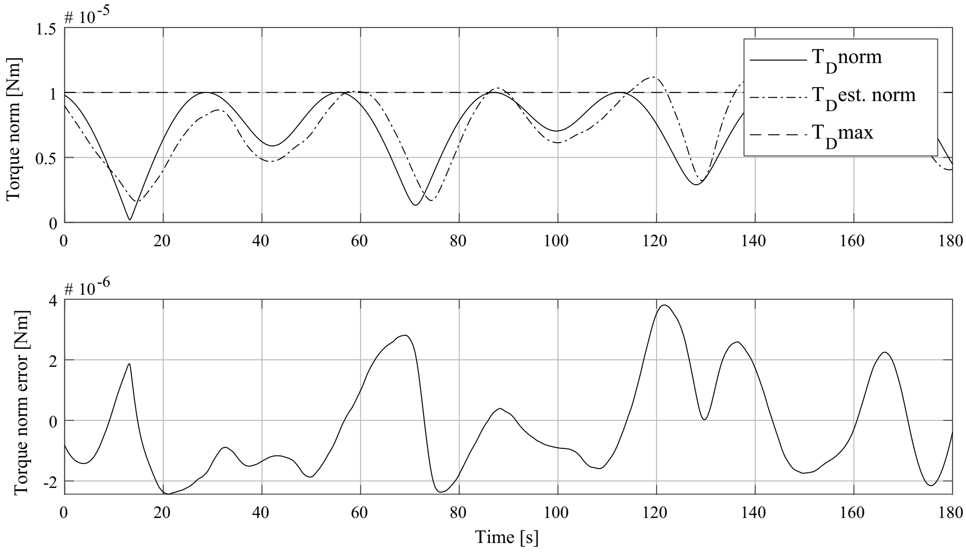


Fig. 8 Numerical simulation of the residual disturbance estimation after balancing

For the in-plane balancing, the nonlinear control law Eq. (20) is employed. Initial conditions for the experiment consist of the platform manually positioned at rest: the unbalance itself triggers the platform to tilt, thus exciting the feedback controller. The convergence time of the in-plane balancing control depends on the magnitude of the unbalance vector $\boldsymbol{r}_{\mathrm{CM}}$. In the first iteration, inertia parameters estimation is performed through the LSQ-SGF technique. As outlined in Sect. 3.3, this requires a known unbalance to be introduced in the *x*–*y* plane, which generates a constraint in the least squares homogeneous equation. Since inertia matrix identification benefits from a high unbalance torque, which excites wide oscillations, a large unbalance shall be used. Free oscillations of the platform are then sampled, starting from initial angular speeds applied manually to the platform. Collected data are processed through the LSQ filter, which generally guarantees convergence in less than 100 s. Figure 9 depicts the estimated principal moments of inertia, which are compared to their a-priori values computed from the CAD model, showing a mismatching of about 5%.

The unbalance vector components are compensated, and a new iteration begins. Plane balancing is repeated and residual $r_{\mathrm{CM},z}$ estimated, this time assuming the inertia known and set equal to the first iteration estimate. The procedure can be further iterated, as long as the *i*th $r_{\mathrm{CM},z}$ estimate is smaller than the $(i-1)$th one, meaning that the unbalance vector is being reduced. In case no significant difference in the estimated value is found, the balancing is ended, and the residual disturbance torque estimated to validate the balancing procedure. The overall process is summarized in Fig. 10.

It was found through experiments that after three balancing iterations the oscillation period becomes extremely long and the residual $r_{\mathrm{CM},z}$ estimation does not improve any further below a threshold of $\approx 7 \cdot 10^{-7} m$. A sample output of the $r_{\mathrm{CM},z}$ estimation during a fine balancing iteration is shown in Fig. 11, while the outcome of the

Fig. 9 Experimental LSQ estimation results

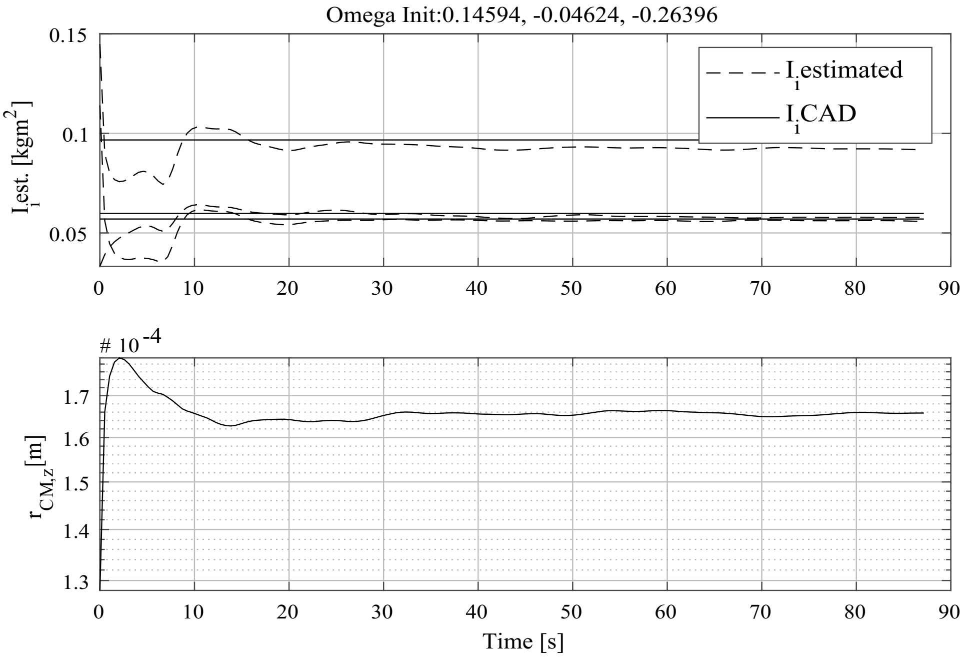
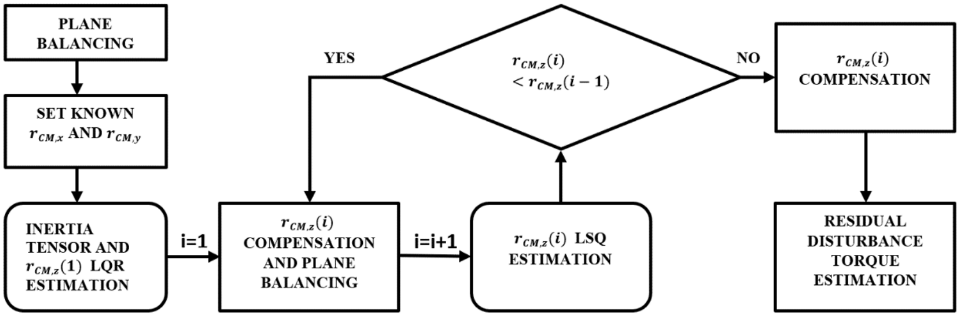


Fig. 10 Iterative balancing procedure


PLANE BALANCING
SET KNOWN $r_{CM,x}$ AND $r_{CM,y}$
INERTIA TENSOR AND $r_{CM,z}(1)$ LQR ESTIMATION
i=1
$r_{CM,z}(i)$ COMPENSATION AND PLANE BALANCING
i=i+1
$r_{CM,z}(i)$ LSQ ESTIMATION
$r_{CM,z}(i) < r_{CM,z}(i-1)$
YES
NO
$r_{CM,z}(i)$ COMPENSATION
RESIDUAL DISTURBANCE TORQUE ESTIMATION

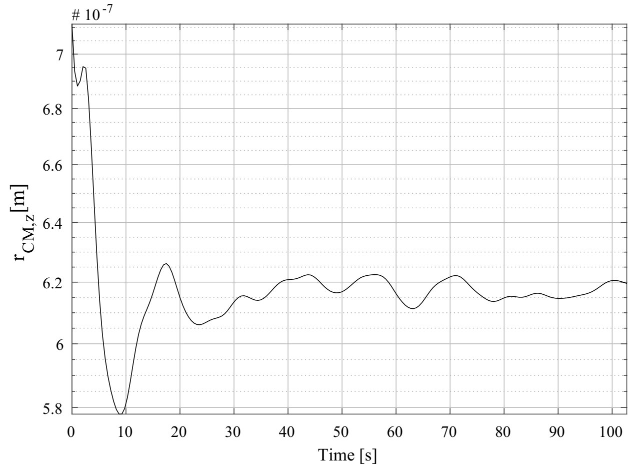


Fig. 11 Fine $r_{CM,z}$ estimation

Table 4 Convergence history of the parameters identification step

Iteration #	Principal inertia moments (kg m^2)	Vertical offset (m)
1	[5.57 5.78 9.17]$\cdot 10^{-2}$	$1.66\cdot 10^{-4}$
2		$2.50\cdot 10^{-6}$
3		$6.19\cdot 10^{-7}$

identification across the three iterations is summarized in Table 4.

The effectiveness of the balancing is checked first by comparing the kinetic energy variation during free platform oscillations before and after balancing. As shown in Fig. 12, three balancing iterations reduce the kinetic energy variation by about 98%. Nevertheless, a more convenient indicator is the magnitude of the residual disturbance torque acting on the system after balancing, which can be estimated with the method described in Sect. 4.3.

The outcome of the estimate is shown in Fig. 13. The upper bound of the torque magnitude was found to be about $5 \cdot 10^{-5}$Nm, while its root mean squared value is $2.5 \cdot 10^{-5}$Nm. These values, despite being one order of magnitude higher than the theoretical lower bound of the unbalance torque alone, set by the mass displacement resolution, Eq. (1), compares favorably with respect to the outcome from similar studies [16, 30, 31].

Results are, therefore, deemed significant, especially when considering that the overall cost of the ABS (sensors, actuators, control and masses) is lower than 500 Euro.

6 Conclusions

In this work, a two-step, iterative procedure for reducing the gravitational torque acting on an attitude dynamic testbed based on air bearing is presented and experimentally tested. Two iterations, at least, are envisaged: a first one, for large unbalance vector compensation and inertia parameters estimation, followed by a fine balancing iteration. Each iteration consists of two stages: a real-time in-plane balancing, followed by an estimation of dynamic parameters.

For the in-plane balancing, feedback control law can be employed. A novel nonlinear feedback control law for single-axis pointing, robust to unmodeled disturbances, is developed and compared to a linearized PID controller.

Once the plane balancing is performed, two components of the unbalance vector are compensated. After the first plane balancing iteration, the residual center of gravity offset, together with an estimate of the inertia matrix elements,

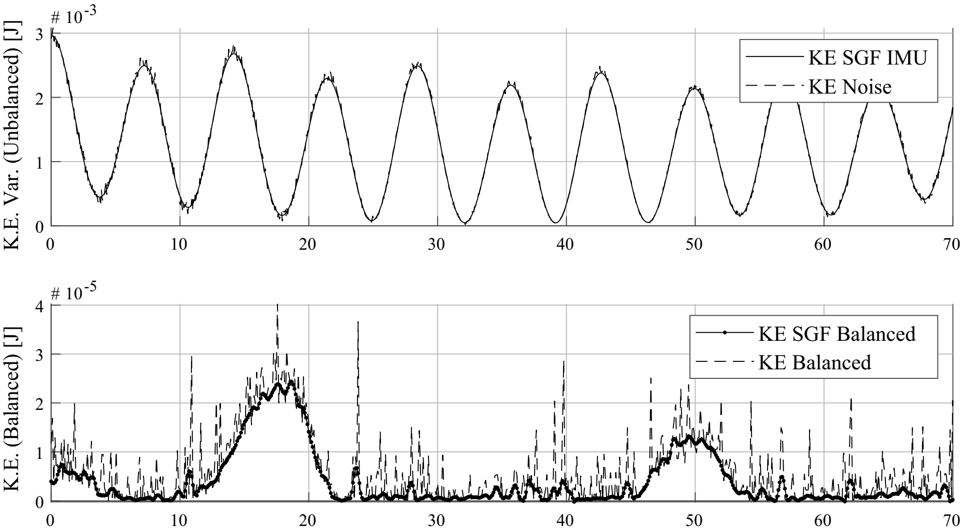


Fig. 12 Kinetic energy variation of the balanced platform

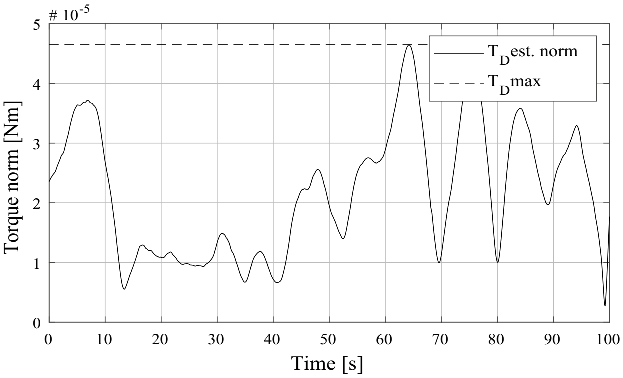


Fig. 13 External torque acting on the balanced platform

are obtained through a batch LSQ estimator. Pre-processing of the angular rate measurements using a Savitzki–Golay filter was found to be of paramount importance for achieving good estimates. For the following balancing iterations, inertia parameters are used as constraints for accurate residual vertical offset estimation.

The proposed balancing procedure was first analyzed on a numerical simulator. The simulator, which accounts for the main sources of measurement noise, system uncertainties, and unmodelled disturbances, allowed us identifying the algorithms combination leading to the most effective balancing. This was then implemented on low-cost hardware, requiring no actuators other than simple linear stages, and experimentally tested.

To validate the balancing procedure, kinetic energy and angular momentum variations were analyzed. A residual disturbance torque smaller than $5 \cdot 10^{-5}$ Nm was retrieved, with a root mean squared value of $2.5 \cdot 10^{-5}$ Nm. A further reduction of the residual torque might be achieved by employing higher-end sensors. Other identified areas of improvement are the structural optimization of the rotational support for reducing its weight without impacting the flexural rigidity, and the implementation of new control laws for planar balancing to enhance the robustness against system uncertainties.

The experimental results confirm that adequate performance for nanosatellite ADCS ground-based simulators can be reached even with low-cost hardware, with the automatic balancing system being effective in reducing the disturbance torque down to the hardware-dependent limit. The design solutions adopted herein and described throughout the manuscript may thus prove useful for other similar facilities.

Appendix A

The closed-loop dynamic under Eq. (17) and within the convergence set $\boldsymbol{\omega}_p = 0$ is:

$$J\dot{\boldsymbol{\omega}} = -K_p \hat{\mathbf{g}} \times \hat{\mathbf{z}} - \boldsymbol{\omega}_g \times J\boldsymbol{\omega}_g \tag{28}$$

The right-hand side of the equation is the sum of two vectors orthogonal to $\hat{\mathbf{g}}$, thus any non-null combination of them would lead to $J\dot{\boldsymbol{\omega}} \perp \boldsymbol{g}$. However, this would lead $\boldsymbol{\omega}$ out of the convergence set, as any non-null $\dot{\boldsymbol{\omega}}$ shall lie parallel to $\hat{\mathbf{g}}$. Thus, in the given set, $\dot{\boldsymbol{\omega}} = 0$. For this to hold, $\hat{\mathbf{g}} \times \hat{\mathbf{z}}$ must be parallel to $\boldsymbol{\omega}_g \times J\boldsymbol{\omega}_g$, that is, $\hat{\mathbf{z}} \parallel J\boldsymbol{\omega}_g$. If the chosen body axes are principle axes of inertia, J is diagonal, and $z \parallel J\boldsymbol{\omega}_g$ may occur only if $\boldsymbol{\omega}_g$ is parallel to $\hat{\mathbf{z}}$, leading to

$$\hat{\mathbf{g}} \times \hat{\mathbf{z}} = \boldsymbol{\omega}_g \times J\boldsymbol{\omega}_g = \mathbf{0} \tag{29}$$

which proves the desired pure spin around $\hat{\mathbf{z}} \parallel \hat{\mathbf{g}}$.

References

1. Perez, F., et al.: DustCube, a nanosatellite mission to binary asteroid 65803 Didymos as part of the ESA AIM mission. Adv. Space Res. **62**(12), 3335–3356 (2018). https://doi.org/10.1016/j.asr.2018.06.019
2. Speretta, S., et al.: LUMIO: an autonomous CubeSat for lunar exploration. In: Cruzen, C., Schmidhuber, M., Lee, Y., Pasquier, H. (eds.) Space operations: inspiring Humankind's future, pp. 103–134. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-11536-4
3. Viscio, M.A., et al.: Interplanetary CubeSats system for space weather evaluations and technology demonstration. Acta Astronaut. **104**(2), 516–525 (2014). https://doi.org/10.1016/j.actaastro.2014.06.005
4. Davoli, F., Kourogiorgas, C., Marchese, M., Panagopoulos, A., Patrone, F.: Small satellites and CubeSats: survey of structures, architectures, and protocols. Int. J. Satellite Commun. Netw. **37**, 343–359 (2019). https://doi.org/10.1002/sat.1277
5. Woellert, K., Ehrenfreund, P., Ricco, A.J., Hertzfeld, H.: Cubesats: cost-effective science and technology platforms for emerging and developing nations. Front. Microbiol. **47**(4), 663–684 (2011). https://doi.org/10.1016/j.asr.2010.10.009
6. Bouwmeester, J., Guo, J.: Survey of worldwide pico- and nanosatellite missions, distributions and subsystem technology. Acta Astronaut. **67**(7–8), 854–862 (2010). https://doi.org/10.1016/j.actaastro.2010.06.004
7. Selva, D., Krejci, D.: A survey and assessment of the capabilities of Cubesats for Earth observation. Acta Astronaut. **74**, 50–68 (2012)
8. Stesina, F., Corpino, S., Mozzillo, R., Rabasa, G.O.: Design of the active attitude determination and control system for the E-ST@R CUBESAT. Proc Int Astronaut Congress IAC **6**, 4585–4594 (2012)
9. Stesina, F., Corpino, S., Feruglio, L.: An in-the-loop simulator for the verification of small space platforms. Int. Rev. Aerosp. Eng. **10**(2), 50–60 (2017)
10. Kiesbye, J., Messmann, D., Preisinger, M., Reina, G., Nagy, D., Schummer, F., Mostad, M., Kale, T., Langer, M.: Hardware-in-the-loop and software-in-the-loop testing of the MOVE-II CubeSat. Aerospace **6**, 130 (2019)
11. Polat, H.C., Virgili-Llop, J., Romano, M.: Survey, statistical analysis and classification of launched CubeSat missions with emphasis on the attitude control method. J. Small Satellite **5**(3), 513–530 (2016)

12. Xia, X., Sun, G., Zhang, K., Wu, S., Wang, T.C., Xia, L., Liu, S.: NanoSats/CubeSats ADCS survey. In: 29th Chinese Control and Decision Conference (CCDC), pp. 5151–5158. (2017)
13. Jovanovic, N., Pearce, J.M., Praks, J.: Design and testing of a low-cost, open source, 3-D printed air-bearing-based. J. Small Satellites **8**(2), 859–880 (2019)
14. Kwan, T.H., Lee, K.M.B., Yan, J., et al.: An air bearing table for satellite attitude control simulation. In: Proc. IEEE 10th Conf. on Indus-trial Electronics and Applications (ICIEA), pp. 1420–1425 (2015)
15. Peck, M.A., Cavender, A.: An airbearing-based Testbed for momentum-control systems and spacecraft line of sight. In: Advances in the Astronautical Sciences (American Astronautical Science), pp. 427–446 (2003)
16. Prado, J., Bisiacchi, G., Reyes, L., Vicente, E., Contreras, F., Mesinas, M., Jurez, A.: Three-axis air-bearing based platform for small satellite attitude determination and control simulation. J. Appl. Res. Technol. **3**(3), 222–237 (2005). https://doi.org/10.22201/icat.16656423.2005.3.03.563
17. Kim, J.J., Agrawal, B.: Automatic mass balancing of air-bearing based three-axis rotational spacecraft simulator. J. Guidance Control Dyn. **32**(3), 1005–1017 (2009)
18. Schwartz, J.L. Hall, C.D.: Comparison of system identification techniques for a spherical air-bearing spacecraft simulator. In: AAS/AIAA Astrodynamics Specialist Conference (2003)
19. Schwartz, J.L., Hall, C.D.: System identification of a spherical air-bearing spacecraft simulator. In: AAS/AIAA Space Flight Mechanics Conference (2004)
20. Schwartz, L.J., Peck, A.M., Hall, D.C.: Historical review of air-bearing spacecraft simulators. J. Guidance Control Dyn. **26**(4), 513–522 (2003)
21. Sternberg, D.C., Pong, C., Filipe, N., Mohan, S., Johnson, S., Jones-Wilson, L.: Jet propulsion laboratory small satellite dynamics Testbed simulation: on-orbit performance model validation. J. Spacecr. Rockets **55**(2), 322–334 (2018)
22. Saenz Otero, A.: The SPHERES satellite formation flight Testbed: design and initial control. M.Sc. Thesis, Massachusset Institute of Technology, Cambridge, MA, USA (2000)
23. Cortiella, A., Vidal, D., Jané, J., Juan, E., Olivé, R.: CAT-2: attitude determination and control system for a GNSS-R Earth observation 6U CubeSat mission. Eur. J. Remote Sensing **49**(1), 759–776 (2016). https://doi.org/10.5721/EuJRS20164940
24. Sutherland, R., Kolmanovsky, I., Girard, A.R.: Attitude control of a 2U Cubesat by magnetic and air drag torques. IEEE Trans. Control Syst. Technol. **27**(3), 1047–1059 (2019). https://doi.org/10.1109/TCST.2018.2791979
25. Thomas, D., Wolosik, A.T., Black, J.: CubeSat attitude control simulator design, AIAA Modeling and Simulation Technologies Conference. Kissimmee, Florida. (2018). https://doi.org/10.2514/6.2018-1391
26. Gavrilovich, I., Krut, S., Gouttefarde, M., Pierrot, F., Dusseau, L.: Robotic test bench for CubeSat ground testing: concept and satellite dynamic parameter identification. In: IEEE/RSJ international conference on intelligent robots and systems (IROS). Hamburg, pp. 5447–5453 (2015)
27. Fullmer, R. R.: Dynamic ground testing of the skipper attitude control system. In: 34th AIAA Aerospace Science Meeting and Exhibit, AIAA Paper 1996-0103 (1996)
28. Young, J.: Balancing of a small satellite attitude control simulator on an air bearing. In: Proceedings of the Utah Space Grant Consortium Symposium, pp. 1–7, Salt Lake City, UT, USA, 19 June 1998
29. Romano, M., Agrawal, B.N.: Acquisition, tracking and pointing control of the bifocal relay mirror spacecraft. Acta Astronaut. **53**(4), 509–519 (2003). https://doi.org/10.1016/S0094-5765(03)80011-5
30. Liu, Y., Li, L., Fu, Z., Tan, J., Li, K.: Automatic mass balancing of a spacecraft simulator based on non-orthogonal structure. In: UKACC 11th international conference on control (CONTROL), pp. 1–6 (2016). https://doi.org/10.1109/CONTROL.2016.7737579
31. da Silva, R.C., Guimaraes, F.C., Loiola, J.V.L., Borges, R.A., Battistini, S., Cappelletti, C.: Tabletop testbed for attitude determination and control of nanosatellites. J. Aerosp. Eng. (2018). https://doi.org/10.1061/(ASCE)AS.1943-5525.0000952
32. Xu, Z., Qi, N., Chen, Y.: Parameter estimation of a three-axis spacecraft simulator using recursive least-squares approach with tracking differentiator and Extended Kalman Filter. Acta Astronaut. **117**, 254–262 (2015). https://doi.org/10.1016/j.actaastro.2015.08.010
33. Hatcher, N.M., Young, R.N.: Automatic balancing system for use on frictionlessly supported attitude-controlled test platforms (NASA TN-D-4426, Langley Research Center, 1968)
34. Chesi, S., Gong, Q., Pellegrini, V., Cristi, R., Romano, M.: Automatic mass balancing of a spacecraft three-axis simulator: analysis and experimentation. J. Guidance Control Dyn. **37**(1), 197–206 (2013). https://doi.org/10.2514/1.60380
35. Keim, J.A., Açıkmeşe, B., Shields, J.: Spacecraft inertia estimation via constrained least squares. In: IEEE aerospace conference (2006)
36. Vandyke, M., Schwartz, J., Hall, C.: Unscented Kalman filtering for spacecraft attitude state and parameter estimation. Adv. Astronaut. Sci. **119**, 217–228 (2004)
37. Kim, D., Choi, D., Oh, H.-S.: Inertia estimation of spacecraft based on modified law of conservation of angular momentum. J. Astron. Space Sci. **27**, 353–357 (2010). https://doi.org/10.5140/JASS.2010.27.4.353
38. Bellar, A.: Satellite inertia parameters estimation based on extended Kalman Filter. J. Aerosp. Technol. Manag. (2019). https://doi.org/10.5028/jatm.v11.1016
39. Kim, D., Yang, S., Lee, S.: Rigid body inertia estimation using extended Kalman and Savitzky–Golay filters. Math. Probl. Eng. (2016). https://doi.org/10.1155/2016/2962671
40. Modenini, D., Bahu, A., Curzi, G., Togni, A.: A dynamic testbed for nanosatellites attitude verification. Aerospace **7**(3), 31 (2020). https://doi.org/10.3390/aerospace7030031
41. de Ruiter, A., Damaren, C.J., Forbes, J.R.: Spacecraft dynamics and control. Wiley, New York (2013)
42. Khalil, H.: Nonlinear Systems, 3rd edn. Prentice Hall, Upper Saddle River (2002)
43. Schaub, H., Junkis, J.L.: Analytical Mechanics of Space Systems. American Institute of Aeronautics and Astronautics, Reston (2009)