**POLITECNICO**
MILANO 1863

SCUOLA DI INGEGNERIA INDUSTRIALE
E DELL'INFORMAZIONE

# Development and Testing of STASIS Balancing Algorithm

TESI DI LAUREA MAGISTRALE IN
SPACE ENGINEERING - INGEGNERIA SPAZIALE

Author: **Niccolò Giannone**

Student ID: 990628
Advisor: Prof. Francesco Topputo
Co-advisors: Gianfranco Di Domenico, Dr. Gianmario Merisio
Academic Year: 2022-23

# Abstract

In the rapidly evolving field of space exploration, CubeSats are becoming increasingly prominent due to their cost-effectiveness and operational efficiency. However, the current satellite communication paradigm struggles to keep up with this growth. The ERC-funded EXTREMA project aims to address this gap by enhancing CubeSats with autonomous capabilities, eliminating certain communication steps. To realize this visionary project, rigorous ground testing is imperative to manage the associated risks. Thus, STASIS, the platform crafted at the DART laboratory in Polytechnic of Milan, was designed to ensure reliable validation of autonomous GNC algorithms. However, to establish a dependable basis for simulations, STASIS shall accurately replicate a micro-gravity environment.

The thesis delineates the development and testing of an automatic mass balancing algorithm for STASIS, aimed to generate a micro-gravity environment for the simulator, pivotal for the success of the EXTREMA project. The balancing process employs a set of shifting masses, wireless controlled to modify the system's internal mass distribution until the offset between the center of mass and the center of rotation is canceled. These masses generate a torque within a plane orthogonal to the gravity vector, introducing an under-actuated control challenge. Therefore, a two-step procedure is proposed.

The first step involves achieving planar balancing through the application of a non-linear feedback control technique. Following, the residual vertical offset is estimated using a constrained batch least squares method. This estimation feeds the platform's inertia parameters to a Kalman filter, potentially refining the accuracy of the balancing procedure. Comprehensive simulations on the platform's digital twin authenticated these algorithms, paving the way for real-world tests. Results proved how reduced is the spectrum of viable strategies to achieve unbiased balancing. Hence, this step was pivotal, pinpointing and addressing possible concerns, and bolstering the algorithms resilience.

Experimental results demonstrate a $10^{-5}$ m precision for automatic planar balancing.

**Keywords:** Automatic balancing, Hardware-in-the-loop, Satellite simulator, Experimental testing, Micro-gravity, Autonomous GNC.

# Sommario

Nel campo dell'esplorazione spaziale, i CubeSat stanno diventando sempre più importanti grazie alla loro economicità ed efficienza operativa. Tuttavia, l'attuale paradigma di comunicazione satellitare fatica a tenere il passo con questa crescita. Il progetto EXTREMA mira a colmare questa lacuna potenziando i CubeSat con capacità autonome, eliminando alcune fasi di comunicazione. Per realizzare questo progetto innovativo, sono indispensabili rigorosi test a terra atti a mitigare i rischi. Con tale fine, STASIS è stato sviluppato presso il laboratorio DART, nel Politecnico di Milano. Per stabilire una base valida per le simulazioni, STASIS deve replicare accuratamente un ambiente di microgravità.

A tale scopo, la tesi presenta l'implementazione e i test sperimentali di un algoritmo di bilanciamento per STASIS. Il processo di bilanciamento impiega un insieme di masse mobili, controllate via wireless per modificare la distribuzione di massa del sistema fino a compensare la differenza tra centro di massa e centro di rotazione. Tuttavia, le masse generano una coppia necessariamente ortogonale al vettore gravitazionale, introducendo un problema di sotto-attuazione, che viene risolto tramite una procedura a due fasi.

La prima fase prevede il conseguimento del bilanciamento planare attraverso l'applicazione di una tecnica di controllo non lineare. In seguito, l'offset verticale residuo viene stimato utilizzando un metodo dei minimi quadrati vincolato. Questa stima, avendo come risultato l'inerzia del simulatore, permette di un'osservazione del sistema basata sulla teoria di Kalman, potenzialmente affinando l'accuratezza della procedura di bilanciamento.

Numerose simulazioni sul gemello digitale della piattaforma hanno autenticato questi algoritmi, aprendo la strada ai test nel mondo reale. I risultati delle simulazioni hanno dimostrato quanto sia ridotto lo spettro delle strategie praticabili per ottenere risultati robusti. Pertanto, questo passaggio è stato fondamentale, permettendo di individuare e affrontare possibili criticità e rafforzando, quindi, la resilienza degli algoritmi.

I risultati sperimentali dimostrano una precisione di $10^{-5}$ m per il bilanciamento planare.

**Parole chiave:** Bilanciamento automatico, Hardware-in-the-loop, Simulatore di assetto, Test sperimentali, Micro-gravità, GNC autonomo.

# Acknowledgements

In the culmination of this enriching academic journey, my heart is imbued with gratitude for those who have illuminated my path with their wisdom, support, and kindness.

First and foremost, my deepest appreciation is extended to Prof. Francesco Topputo, whose unwavering commitment to excellence and scholarship has been a beacon of inspiration, motivating me to strive for my utmost best. His guidance has been a compass in the realms of uncertainty, for which I am eternally grateful.

Profound gratitude is also due to my esteemed co-advisors, Gianfranco Di Domenico and Dr. Gianmario Merisio. Their invaluable technical support and insightful advice have been pivotal in my research. The contours of this thesis are shaped significantly by their contributions; if anything within these pages proves useful, it is largely a testament to their sagacious counsel. Beyond the realm of academic guidance, getting to know them on a personal level has been a profound pleasure. Their warmth and genuineness have made the strenuous journey of research a more human and heartfelt experience, and for this, I owe them a debt of sincere appreciation. Heartfelt thanks to Fabio Ornati, a continuous help against the challenges encountered during experimentation. His constant availability was paramount during this research, and for this, I extend my profound gratitude to him. To my family, whose constant encouragement and unwavering support have been my foundation throughout this journey, I owe an immeasurable amount of thanks. To my mother and father, for their endless love and sacrifice, I hope this thesis is a reflection of the values and dedication you have instilled in me. A special note of thanks to my sister Edvige, whose innate ability to understand and communicate with me has always been extraordinarily genuine and pure. In the realm of friendship, my gratitude is boundless. To my friends in Milan, whose names are too numerous to list but not forgotten, I am thankful for the laughter and respite provided. To Gianmarco and Davide, my lifetime companions, your support from afar has been as solid as ever.

And to Sonia, my dear girlfriend, whose sweetness and sensitivity have been the solace in my stress, the patience in my impatience, I extend my deepest thanks. Your presence has been a gentle reminder of the joys beyond work, and for that, I am eternally grateful.

# Contents

# 1 | Introduction

In the ever-evolving realm of space technology, CubeSats represent a significant paradigm shift, emblematic of the modern age's penchant for miniaturization and cost-effectiveness. Originally conceptualized to provide affordable access to space for university students, CubeSats have transcended their educational origins to become pivotal assets in both commercial and scientific arenas.

Indeed, the advent of advanced microelectronics and miniaturized subsystems has allowed for the creation of robust satellite capabilities in considerably smaller form factors. In conjunction with streamlined satellite structures, this ensures that CubeSats are not just affordable but also efficient.

Furthermore, the standardization of interfaces between launchers and payloads has opened up a new world of launch opportunities. Standardization reduces complexities involved in launch configurations, making it more feasible for multiple CubeSats to hitch a ride on a single rocket[1]. This shared approach to launching not only minimizes costs but also accelerates the frequency of launches, ensuring rapid deployment of satellite constellations. As a result, CubeSats are now at the forefront of space exploration and observation, democratizing space access and catalyzing innovations in various domains, from Earth observation to deep space exploration [1].

## 1.1. The paradigm shift: towards self-driving interplanetary CubeSats

The exploration of distant celestial bodies, while captivating to both public and private sectors, presents notable operational challenges. Reference [1] highlights that the actual standard procedure mostly utilizes the Deep Space Network (DSN) to communicate with distant spacecrafts, emphasizing the limitations inherent in this approach. Indeed, the DSN is designed to employ the minimum number of ground stations to ensure that at least one station is in line of sight with any point in space at any time instant [1] (Fig-

---

[1]European Space Agency, "CubeSats", `https://www.esa.int/Enabling_Support/Preparing_for_the_Future/Discovery_and_Preparation/CubeSats`, last accessed: 9/11/2023.

ure 1.1). Consequently, the DSN operates on a highly regimented timetable and may not be equipped to handle the rapid expansion similar to what has been observed in more proximate applications. Additionally, the vast distances separating Earth from deep-space missions probes may lead to asynchronous communication. This, in turn, escalates operational expenses due to human personnel and the need for specialized hardware.
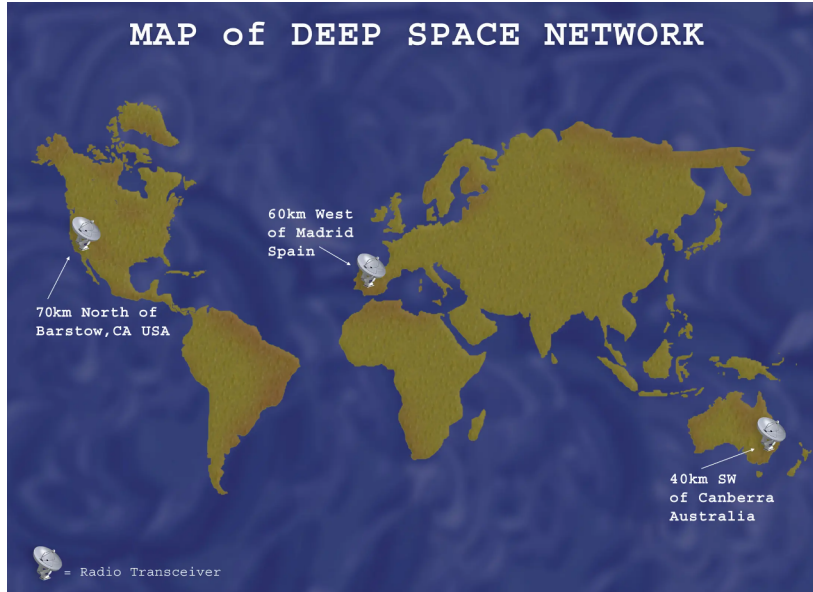


Figure 1.1: DSN ground stations locations.

In this scenario, the European Research Council (ERC)-funded Engineering Extremely Rare Events in Astrodynamics for Deep-Space Missions in Autonomy (EXTREMA) project, seeks to revolutionize current practices [2]. By enhancing interplanetary CubeSats with autonomous capabilities, the project aims to address the limitations posed by the limited access to ground stations and drastically reduce the operational expenses associated with deep-space probes. To realize this captivating and ambitious vision, the EXTREMA project is anchored on three fundamental pillars:

1. **Autonomous Navigation**. The development of CubeSats-safe navigation algorithms enabling autonomous reconstruction of their position in the deep-space by exploiting the information from the surrounding environment [3–6].

2. **Autonomous Guidance and Control**. The implementation of lightweight guidance algorithms efficiently operating given the on-board computational constraints, enabling the computation of a time-definite thrust profile [7–9].

3. **Ballistic Capture**. The exploitation of the multi-body dynamics of the Solar System to maintain prolonged proximity to a celestial body [10–12].

For the success of this complex and ambitious project, ensuring reliability for the Guidance, Navigation and Control (GNC) algorithms is pivotal. As such, rigorous performance and robustness testing of these autonomous algorithms is imperatively required.

## 1.2. Spacecraft attitude simulators

Spacecraft simulators play a pivotal role in the field of aerospace engineering. They provide a physical environment to test and validate the design, performance, and operational capabilities of a spacecraft before its actual deployment. By replicating the conditions of outer space, these simulators offer invaluable insights into potential challenges and contingencies, thereby enhancing the safety and success of space missions.

Their integration into space mission design has gained significant prominence in recent decades, paralleling the marked escalation in the complexity of GNC algorithms. Indeed, complex GNC strategies, while augmenting the autonomy and efficiency of space missions, may result in elongated simulations that might not consistently mirror real-world performances. Thus, the pre-flight performance evaluation via comprehensive hardware and software in-the-loop ground testing is of utmost importance, resulting in the rising popularity of the spacecraft simulators in recent decades [13].

Reference [14] provides a throughout of the technological solutions until the year 2003. The paper represents an historical review of the precedent 50 years development of air-bearing supported facilities, which have the advantage of minimizing the friction with respect to other mechanical arrangements for spacecraft simulators. Based on specific simulation needs and fidelity criteria, various configurations can be integrated, each with different Degree Of Freedoms (DOFs).

- Planar systems, enabling 2 DOFs planar motion, presented in Figure 1.2a.

- Rotational systems, granting a 3 DOFs rotational dynamics, Figure 1.2b.

- Hybrid systems, 5 DOFs systems which incorporate both the previous systems, as illustrated in Figure 1.2c.

The last category is employed when high-fidelity simulations of complex algorithms shall be performed. Furthermore, if necessary, the $6^{th}$ DOFs can be introduced into the system by incorporating a counterbalancing mass propelled by electric motors, enabling vertical motion, therefore culminating in a comprehensive representation of spacecrafts dynamics.

(a) Planar 2 DOFs.              (b) Spherical 3 DOFs.              (c) Hybrid 5 DOFs.

Figure 1.2: Simulator configurations.

Concerning the 3 DOFs rotational systems, which necessitate meticulous facility design to attain alignment with the actual spacecraft dynamics, they are categorized into two primary groups as highlighted in [14] (Figure 1.3).

- Tabletops and umbrellas, providing full freedom in yaw, while roll and pitch motion are bounded (Figures 1.3a and 1.3b).

- Dumbbells, providing full freedom for both yaw and roll axis, therefore bounding only the pitch rotation (Figure 1.3c).



(a) Tabletop.                    (b) Umbrella.                    (c) Dumbbell.
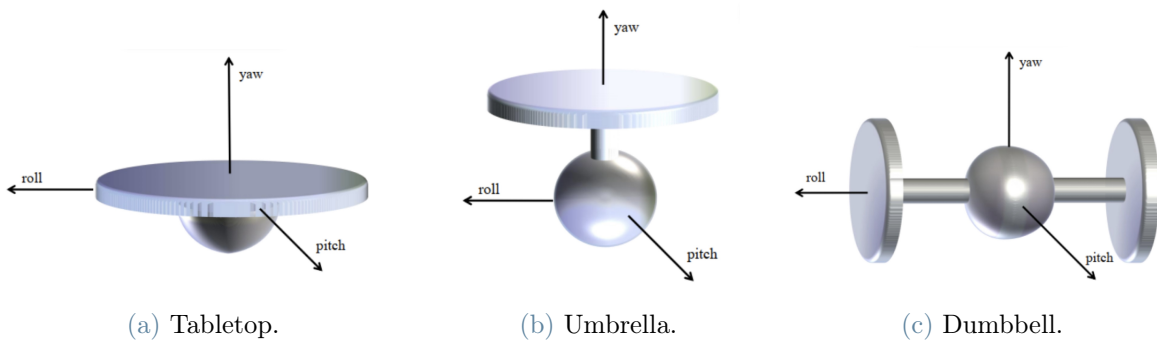
Figure 1.3: Attitude simulator configurations.

Furthermore, the work [14] discusses the main disturbance torques which may affect the motion, with reference to the classification presented by Smith in a conference paper in

1964 [15]. Hence, facility enhancements are proposed through some pragmatical recommendations and disturbance minimization strategies.

Moving to more recent test-beds, reference [16] represents a modern approach for the replication of spacecrafts rotational dynamics. Besides the great progresses in terms of hardware and software, the paper offers a valid demonstration of the recent years trend of the spacecraft simulators: the higher and higher penchant to simulate nano-satellite scale spacecrafts. References [17, 18] confirm this temporal trend, also highlighting the major advantages to have a 6 DOFs platform for the testing of formation guidance, relative navigation and control algorithms.

Even though published in different decades, all of the aforementioned references share a commonality, that is underscoring the imperative necessity to compensate for the offset between the Centre of Mass (CM) and the Centre of Rotation (CR) in order to accurately simulate a micro-gravity environment (i.e., the balancing problem).

## 1.2.1. The balancing problem

The 3 DOFs satellite simulator cannot replicate micro-gravity in its exact form and necessitates a meticulous design to satisfy this achievement. Indeed, when the system is mounted on the air-bearing, there is no assurance that it remains free from torques, which are virtually non-existent in space.

Reference [15] elucidates the predominant disturbance torques inherent to the air bearing platform, as exhibited within the Table 1.1. It is imperative to note that achieving precise compensation for all torques remains a challenge. However, Table 1.1 offers a concise representation, encapsulating the complexity of the balancing procedure. As a consequence, the spacecraft simulator design involves numerous critical decisions to guarantee that the vast spectrum of disturbances does not substantially impact the process. Indeed, as demonstrated in future chapters, if even a few of these disturbances surpass the negligible threshold, the entire procedure might be jeopardized.

Table 1.1: Torque catalogue.

| Source | Torques | Remarks |
|---|---|---|
| Platform | • Static unbalance<br>• Dynamic unbalance<br>• Anisoelasticity<br>• Material instability<br>• Equipment motion<br>• Gravity gradient | Torques arising from the CR-CM offset. |
| Air-bearing | • Aerodynamic turbine effect<br>• Exhaust air impingement | Non-symmetrical airflow and imperfect scavenging may be sources of torques. |
| Environment | • Aerodynamic Damping<br>• Air currents<br>• Magnetic field<br>• Vibrations<br>• Radiation pressure | Most troublesome category of torques to minimize in a simulator configuration. |
| Test system | • Electrical wires to base<br>• Mass shift in bearings and loose fits<br>• Battery discharge<br>• Reaction jet supply discharge<br>• Replacement of components | Torques arising from the test system configuration. |

Within this vast array of disturbance effects, one particular torque stands out for its magnitude, being notably greater than the others, i.e. the torque resulting from the CM-CR offset [17, 18]. Under the effect of this torque, the spacecraft simulators oscillates around the center of rotation, mirroring the behavior of a 3D pendulum (Figure 1.4).
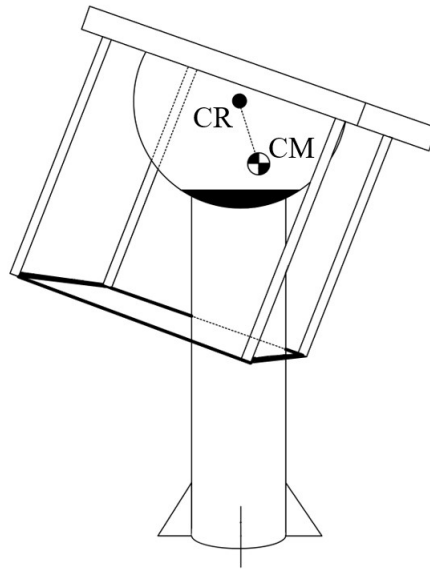
Figure 1.4: The 3D pendulum effect for a spacecraft simulator.

Hence, literature historically focused on the compensation of this effect, approaching the problem through two distinguished methodologies:

1. **Mechanical compensation**. A set of movable masses is employed to modify the internal mass distribution of the system, aiming to compensate the offset between CR and CM [16, 19].

2. **Active control compensation**. Using actuators (e.g. control moment gyroscopes, thrusters), external torque is applied to the system to achieve continuous compensation of the disturbance [18].

In general, the trade-off between these methodologies is performed considering the desired level of complexity of the simulator, the target fidelity of the simulations and the specific system requirements (e.g., hardware availability or need for distinct balancing and control hardware). Table 1.3 presents a comparison of the methodologies, emphasizing benefits and drawbacks of each of them. Captured the primary features of the two methodologies, in a design process specific weighting factors to their key performance indicators shall be applied. These weights should reflect the practical challenges and targeted objectives of a particular spacecraft simulator, enabling a contingent comparison between the two approaches.

Table 1.3: Compared mechanical and active control methods.

| Method | Advantages | Drawbacks |
|---|---|---|
| Mechanical | **Passivity**: no power needed to maintain micro-gravity conditions. **Consistency**: unless physically altered, consistent behaviour among simulations is achieved. **Conceptual simplicity**: the complexity of micro-gravity simulations is translated in the cancellation of a physical vector. | **Static reconfiguration**: balancing is performed for any change in satellite mock-up. **Thrusters incompatibility**: CM shift are not allowed during attitude dynamics simulations. **Potential for drift**: periodical re-calibrations are recommended to withstand CM drift. |
| Active control | **Flexibility**: fast adaptation to different scenarios. **Major simulation capability**: can mimic a broader range of spacecraft dynamics. **Dynamic re-balancing**: potentiality for a real-time compensation of a time-variant disturbance. | **Complexity**: sophisticated control systems required. **Power Consumption**: actuators are in continuous operation. **Potential for Errors**: incorrect compensation may compromise the GNC algorithm validation. **Wear and Tear**: potentiality for hardware malfunctioning over time. **Saturation**: risk for saturation during simulations. |

## 1.3.  The EXTREMA Simulation Hub

The EXTREMA Simulation Hub (ESH) is an integrated infrastructure in the Deep-space Astrodynamics Research and Technology (DART) laboratory, designed for dynamic simulations that focus on interactions between deep space probes and their environment [20, 21]. This ensures precise testing of autonomous GNC systems tailored for CubeSats in deep-space scenarios. Within this experimental setup, individual subsystems are isolated and tested independently to circumvent the complexities associated with integration. For this reason, a comprehensive experiment involving all the subsystems is required. At the core of the facility is the guidance unit, responsible for executing the GNC algo-

rithms and managing the software tailored to identify and target ballistic capture events. This unit relies on high-definition optical scene projections, which serve as inputs for the navigation camera. A cold gas thruster interprets and executes commands from the guidance unit, with a force transducer gauging thrust magnitude. Furthermore, there is also an attitude simulation component: the guidance unit will be affixed to a platform, namely SpacecrafT Attitude SImulation System (STASIS). Inputs such as thrust vector, magnitude, and external disturbances are then directed to EXTREMA SPace Environment SImulator (SPESI), an orbital propagator connected to a rendering unit. The whole setup, depicted in Figure 1.5, effectively emulates the conditions and visuals of deep-space.
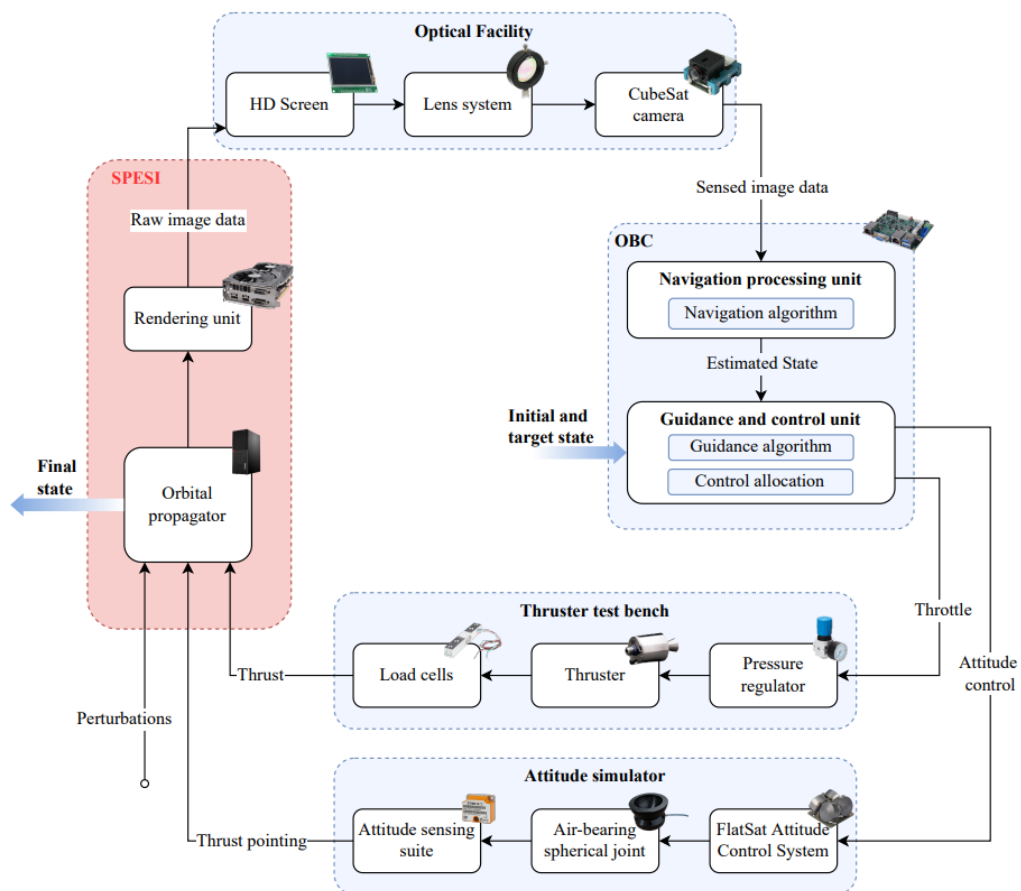


Figure 1.5: ESH functional breakdown [21].

It is noteworthy that simulating GNC algorithms for deep-space probes inherently poses difficulties. Specifically, interplanetary transfers can span years, rendering such duration impractical for simulation purposes. To address this challenge, dynamic similarity is employed to condense real-world time scales. Such a strategy involves a set of scaling parameters to translate conditions from the target world to the laboratory setting. Moreover, these parameters are fine-tuned during different phases to enhance fidelity during

critical mission phases. The underlying concept of this time-condensation approach is encapsulated in Figure 1.6, which presents the simulation path employed in ESH.
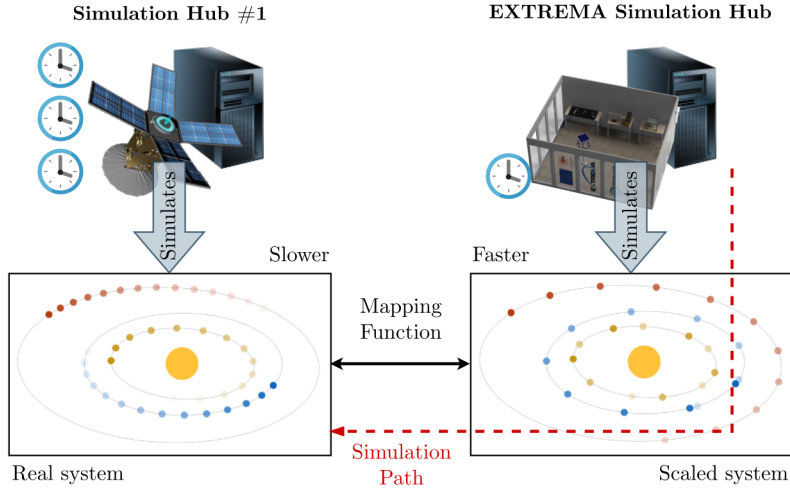


Figure 1.6: ESH simulation path [19].

During the simulations, the capabilities of the attitude control shall be rigorously verified. Indeed, in the closed-loop guidance approach proposed in the EXTREMA project, implies two primary requirements on the spacecrafts attitude control [19]:

1. In the cruising phase, minimal discrepancy between the thruster pointing direction and nominal thrust vector direction, determined by the guidance algorithms, shall be imperatively guaranteed.

2. In the navigation phase, the capability of performing a sequence of slew maneuvers is required. Indeed, in this phase, the spacecraft attitude orientation shall guarantee the target planets to appear in the navigation camera Field Of View (FOV), to retrieve optimal information for the autonomous navigation algorithms.

The validation of these high-level requirements is the primary role of STASIS in the EXTREMA project.

### 1.3.1. STASIS

STASIS [19] is the satellite simulator assembled in the ESH of the DART laboratory, with the primary objective of simulating the rotational dynamics inherent to deep-space probes. Operating based on the principles of air-bearing simulators, STASIS specifically classifies within the tabletop category. In this configuration, it permits unrestricted rotation around

the yaw axis, imposing constraints on the pitch and roll, limiting them to an angle of ±30°. STASIS is planned to be balanced with the mechanical compensation strategy, a design choice strictly related to the simulation of deep-space GNC algorithms inherent to the EXTREMA project. Specifically, actuators' compensation presents inadmissible limitations when considering deep-space autonomous GNC algorithms, as the extended duration required for the simulations ensures the likelihood of saturation. It is noteworthy the the problem of potential gargantuan simulation times, briefly reported in Section 1.3, have been rigorously engineered and addressed through parameters adimensionalization, allowing to compress the simulations time-scales [19–21]. Nonetheless, simulations and experiments might span several days, which is still impractical for an active compensation technique. Hence, STASIS configuration is presented in Figure 1.7.
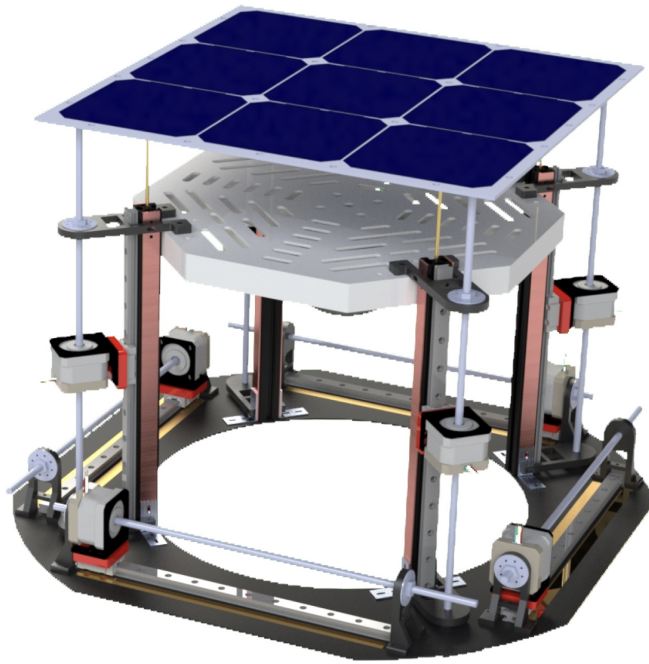


Figure 1.7: STASIS CAD model.

While the platform remains in continuous development, by the designated period of experimentation, it is projected to be constituted of:

1. **Attitude Determination and Control Subsystem (ADCS)**: STASIS is equipped with a LED-based attitude reconstruction system, namely ground truth, which employs an optical camera to sense the position of an array of LEDs mounted on the upper face of the platform. Figure 1.8 presents schematically the LED system, which is mounted in a axial symmetric configuration, except for a reference LED which is employed to remove any ambiguity on the body-fixed frame orientation.

The captured images are processed to determine the attitude of the simulator referring to a known arrangement, therefore the reconstruction is totally unaffected by bias instability. However, being an off-board system, the attitude determination algorithm is heavy. Hypothetically angular rates may be derived from differentiation of attitude knowledge. Nonetheless, this procedure may result in extremely noisy measurements. For this reason, STASIS is also equipped with onboard ADCS hardware, the inertial measurement unit Bosch BNO055[2]. The platform, currently under development, is intended to be outfitted with Reaction Wheels (RWs) for future attitude control. However, at the time of the experiment's execution, only a single wheel was available.
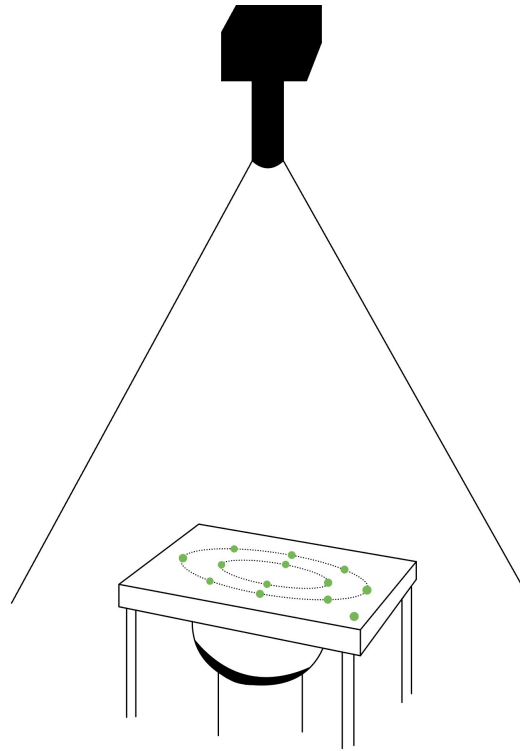


Figure 1.8: Schematic representation of the ground truth system, with a reference LED on one edge to remove ambiguities on attitude reconstruction.

2. **Movable Mass Units (MMUs)**: The set of MMUs represents the specific hardware to modify the internal mass distribution of STASIS [19, 22]. The complete set is composed of 8 masses, 4 horizontal masses, devoted to planar compensation of the offset, and 4 out-of-plane vertical masses (Figure 1.9).

---

[2]BOSCH, "Smart Sensor BNO055" https://www.bosch-sensortec.com/products/smart-sensors/bno055/, accessed: 19/09/2023
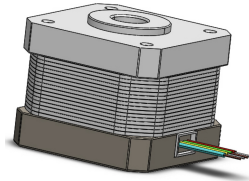
Figure 1.9: MMU CAD model.

Each mass is moved by a permanent magnet stepper motor, adopting screw driven transmission to convert the rotational into translational motion. The received command contains the number of steps and the buffering time between following steps. The command is processed by the motor controller, devoted to feed the Enable port voltage (ENA) and Reverse port voltage (REV) of the driver. The driver toggles the phase winding in a sequence to move the rotor step by step. As a result, the driver can determine the direction, rotational speed, and angular position of the motor based on the input provided by the control unit. The discussed control chain of the STASIS MMUs can be schematized as presented in [22] (Figure 1.10).
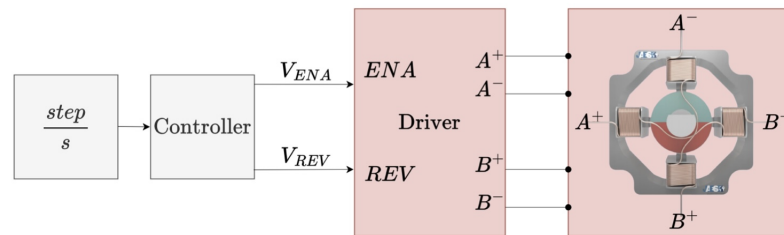


Figure 1.10: Stepper motors control chain [22].

3. **Power Generation and Management System (PGMS)**: Solar panel equipment is planned for the future developments of the platform, ensuring energetic autonomy of the system to enable operations of all the platform apparatuses.
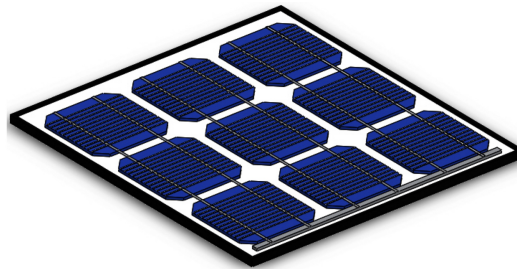


Figure 1.11: Solar panel CAD model.

## 1.4.    Motivation and possible outcomes

The development and testing of an automatic balancing procedure for STASIS is of paramount importance in the spacecraft subsystems design of the EXTREMA project. As a matter of fact, within the framework of autonomous GNC simulations, a balanced satellite simulator, facilitating the simulation of micro-gravity conditions, emerges as an invaluable asset for the purposes of:

- **Research and development advancements**: A simulated environment aids in refining ADCS and GNC algorithms, enabling iterative development. Upon trained knowledge on system behavior, engineers can promptly operate with refinement procedures in real-time.

- **Risk mitigation**: Launching a satellite with untested autonomous systems might result in catastrophic failures. By testing in a micro-gravity environment, the potentiality of pre-flight errors rectification is introduced in the design process, boosting mission success and preventing financial losses.

Simulations of micro-gravity condition are expected to have a very relevant impact on the EXTREMA project, enabling to bolster the chances of highly innovative missions success. Indeed, potential results of the implementation of an automatic balancing procedure are:

- **Enhanced reliability**: Ground testing is expected to result in higher reliability, lower number of in-flight correction and, consequently, longer mission lifetimes.

- **Cost efficiency**: The anomalies addressing in the pre-flight phase is a potential for reducing the costs related to in-flight adjustments.

- **Performances improvement**: An hardware-in-the-loop iterative design enables the refinement of the algorithms, enhancing the mission objective achievement.

- **Accelerated design and deployment**: Confidence in the system functionality is expected to reduce considerably the decision-making times.

In summary, in paving the way for advanced deep-space mission, it is still essential to address practical challenges and ensure unwavering reliability.

## 1.5.    Research questions and objectives

To provide a valid aid in the design of STASIS ADCS the research question that shall be answered is the following:

1. **Main research question**: *To what extent an automatic balancing procedure can provide a micro-gravity environment for an air-bearing CubeSat simulator?*

That implicitly requires to face different operative research questions:

2. **Operative research question**: *What is the impact of the other disturbances (e.g., aerodynamic drag, structural sagging) on the balancing performances for a nano-satellite dimensions simulator?*

3. **Operative research question**: *How attitude sensors noise can affect the balancing performance?*

4. **Operative research question**: *What is the most efficient strategy to achieve an unbiased platform balancing?*

5. **Operative research question**: *What are the advantages of having 4 planes of symmetry in the balancing hardware configuration?*

The goals that this work shall achieve to answer the questions can be outlined as follows:

- **Objective 1.** *Developing an automatic balancing algorithm that maximizes robustness to unmodeled dynamics and STASIS sensors noise.*

- **Objective 2.** *Testing the automatic balancing algorithm on STASIS and assess the attainable performances.*

At this point, considering the primary goals and the specific features of STASIS, some high level requirements can be assessed.

Table 1.5: High-level compensation requirements.

| ID | Type | Text | Rationale | Verification |
|------|------|------|-----------|--------------|
| R001 | O | The algorithm shall be flexible with respect to changes in satellite mock-up. | A crucial feature for a work-in-progress platform is the capability to perform balancing under different conditions and phases. | Inspection |

Table 1.5 – *Continued from previous page*

| ID | Type | Text | Rationale | Verification |
|---|---|---|---|---|
| R002 | F | The algorithm shall be capable of accomplish balancing without using RWs. | Uncertain availability. Even if they were delivered promptly, mounting challenges arise. Placing them on the top could compromise attitude knowledge, while bottom placement could exacerbate vibration effects due to the increased distance from the rotation center. | Inspection |
| R003 | F | The algorithm shall avoid error propagation of preliminary estimations. | Independence from any preliminary estimation on the problem parameters is crucial in an experimental procedure. | Inspection |
| R004 | F | The algorithm shall operate with the available ADCS hardware precision. | Convergence of the algorithm shall be guaranteed with the available sensors noise and MMU to prevent from further design bottlenecks. | Analysis |
| R005 | F | The algorithm shall be robust against measurements noise. | To maximize performances, the trade-off between different strategies must be performed considering that STASIS has very accurate off-board attitude reconstruction system. | Analysis |
| R006 | O | The compensation shall guarantee brevity in time. | Minimized time window ensures minimal impact of unmodeled dynamics and disturbances. | Analysis |
| R007 | O | The compensation shall exploit the symmetrical configuration of the MMU. | The performance guaranteed by a symmetrical MMU setup shall be assessed to answer the $5^{th}$ operative research question. | Analysis |
| R008 | O | The compensation strategy should guarantee fast repeatability. | The complexity of the problem may cause divergence also in proper algorithms, therefore the strategy should minimize the restart time. | RoD |

Table 1.5 – *Continued from previous page*

| ID | Type | Text | Rationale | Verification |
|----|------|------|-----------|--------------|
| R009 | V | The performance assessment shall not alter the physical arrangement of the system. | Performances shall be gauged through non-invasive procedures to prevent from alteration of the results. | Inspection |

Table 1.5 enumerates the initial requirements that have been delineated to ensure compatibility with the evolving FlatSat platform, without introducing additional design constraints. Consequently, the devised algorithm is mandated to compensate STASIS without relying on RWs. Nonetheless, RWs are not entirely ruled out, their utilization would be reserved for potential enhancement or fine-tuning of the outcomes.

It should be emphasized that, for effectively addressing the associated research question, the algorithm is designed to leverage the symmetrical configuration of the MMU. This preference is translated on preferring algorithms that segregate the problem into planar and vertical balancing stages, given the minimal interference of the second stage on the outcomes of the first if the MMU configuration is symmetrical.

## 1.6. Thesis outline

The thesis is organized as follows:

- **Chapter 1**: the thesis begins with an introduction to the balancing problem, situates STASIS within the scope of the EXTREMA project, and delineates the initial requirements guiding this study.

- **Chapter 2**: this chapter delineates the theoretical foundation required to institute a standardization and a rigorous scientific methodology pertaining to the problem at hand.

- **Chapter 3**: mathematical proofs underpinning the entire automatic balancing procedure are elucidated, with a comprehensive exposition of the rationale for each adopted strategy.

- **Chapter 4**: numerical simulations outcomes are detailed, highlighting both the constraints and potentialities intrinsic to each strategy.

- **Chapter 5**: the experimental phase concerning the platform is delineated, eluci-

dating the rationale for the devised strategy, which is contingent upon the findings from the simulations.

- **Chapter 6**: this chapter presents the results procured from the experimental evaluations on STASIS.

- **Chapter 7**: a recapitulation of the accomplishments of this research is provided, accompanied by recommendations for prospective advancements.

# 2 | Theoretical foundation

In this chapter, the bedrocks upon which the entirety of this thesis stands are presented. Herein, an unambiguous definition of the reference frames essential for capturing the dynamics is proposed. The mathematical model is detailed, leveraging the Euler rigid body equations to describe the behavior of the rotational dynamics.

Furthermore, partial observability, which stems from the orthogonality between the soliciting torque and the gravity vector, is mathematically proven. This phenomenon, coupled with its implications, is dissected in detail to offer readers an in-depth understanding of its significance and inherent challenges.

Finally, a comprehensive overview of the state-of-the-art solutions traditionally employed for automatic balancing procedures is provided. This chapter, therefore, serves not merely as a mere introduction but as a beacon, illuminating the intricacies of the problem at hand, standardizing its description, and spotlighting the immediate challenges, such as partial observability. Such foundational insights are indispensable for the design and execution of a consistent experiment.

## 2.1. Reference frames and kinematics

Prior to delving into the dynamic modeling of the simulator, it is imperative to establish a standardization of the problem. Hence, reference frames must be delineated with precision. In the scope of this study, two distinct frames are stipulated: an inertial and a body-fixed frame.

- The inertial frame has its z-axis along the local vertical direction, whereas the x-y plane is parallel to ground.

- The body frame, rigidly affixed to the moving body, has axes parallel to the MMUs screws. Specifically, the z-axis corresponds to the nominal direction of the vertical MMUs, oriented upwards, while the x and y-axes align with the horizontal MMUs.

Both the frames are centred in the Centre of Rotation of the platform (Figure 2.1).
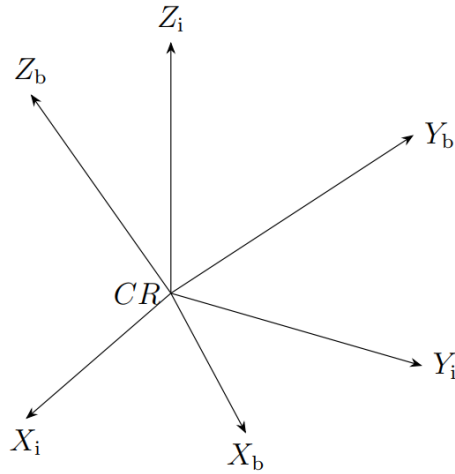
Figure 2.1: Inertial (i) and body-fixed (b) frames.

The orientation of the inertial frame axes is designed for a straightforward representation of the gravity vector, as encapsulated in Equation (2.1), which is pivotal, allowing to lighten the mathematical manipulations in subsequent chapters.

$$\mathbf{g} = \begin{bmatrix} 0 \\ 0 \\ -g \end{bmatrix}. \tag{2.1}$$

Concurrently, the definition of the body frame is very convenient for an automatic balancing procedure, as it obviates the need for rotational transformations between the body and the motion directions of the MMUs. This alignment strategy, which superimposes the observation axes upon the actuation axes, significantly streamlines the compensation process, and potentially reduces the computational load of the algorithms.

Regarding the kinematic representation of the rotation between the frames, multiple mathematical forms are used (e.g., direction cosine matrix, quaternions, Euler angles), depending on the convenience in each case. For instance, for a PID control application, a kinematics represented in the form of Euler angles, may result very direct and simple to implement, since the angles are the errors to feed the control with. Furthermore, this attitude representation proves especially beneficial for a tabletop attitude simulator. Given that the roll and pitch angles are confined to ±30°, the Euler angles in the 321 sequence remain non-singular, eliminating the need for any switching. Conversely, the direction cosine matrix is employed for quick and efficient vector rotations between the frames, while quaternions are used to seamlessly interface with attitude sensors.

## 2.2.  Dynamic model

The dynamic model adopted for this work is the one of a 3D-pendulum-like motion, whose physical model is represented in Figure 2.2.
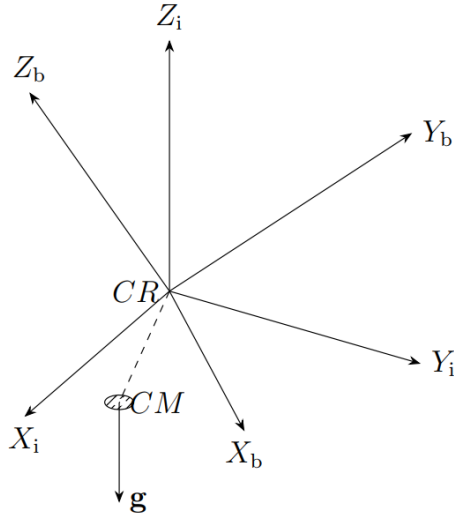


Figure 2.2: 3D pendulum physical model.

Hence, the mathematical model is represented by the Euler equation Of Motion (EOM) centred in the CR of the simulator, where the external torque considered is only the one due to the CR-CM offset, as reported in Equation (2.2).

$$
\begin{aligned}
\frac{d\mathbf{H}_{CR}}{dt} &= \mathbf{M}_{CR} \\
\dot{\mathbf{H}}_{CR} &= J\dot{\boldsymbol{\omega}} + \boldsymbol{\omega} \times J\boldsymbol{\omega} \\
\mathbf{M}_{CR} &= \mathbf{r} \times \mathbf{F}_g = \mathbf{r} \times m\mathbf{g},
\end{aligned}
\tag{2.2}
$$

where $\mathbf{H}_{CR}$ is the angular momentum, $\mathbf{M}_{CR}$ is the external momentum, $J$ is the platform inertia matrix, $\boldsymbol{\omega}$ is the platform angular rate, and $\mathbf{r}$ is the CR-CM offset.

However, to achieve faster manipulation, the same equation can be arranged in the form:

$$
J\dot{\boldsymbol{\omega}} + \boldsymbol{\omega} \times J\boldsymbol{\omega} = -m[\mathbf{g}\times]\mathbf{r},
\tag{2.3}
$$

where $[\mathbf{g}\times]$ is a skew-symmetric matrix, given by:

$$[\mathbf{g}\times] = \begin{bmatrix} 0 & -g_z & g_y \\ g_z & 0 & -g_x \\ -g_y & g_x & 0 \end{bmatrix}. \tag{2.4}$$

It is worth noticing that, as reported in literature, drag is usually the most significant disturbance [23], therefore, an higher fidelity mathematical model of the dynamics is:

$$J\dot{\boldsymbol{\omega}} + \boldsymbol{\omega} \times J\boldsymbol{\omega} = -m[\mathbf{g}\times]\mathbf{r} - \varepsilon\Lambda J\boldsymbol{\omega}, \tag{2.5}$$

wherein the aerodynamic damping effect is shaped proportional to the platform angular momentum. Indeed, in modeling drag, it is imperative to consider the influence of the platform's mass distribution. By simply modeling drag as directly proportional to the angular rate, the nuances introduced by varying mass distribution would be irrelevant. For instance, even if the platform rotates swiftly around the z-axis, if the mass distributed around this axis is minimal, the resulting drag will be relatively small. Therefore, to capture this intricacy, drag is modeled in this work as proportional to the rotational angular momentum of the platform, which inherently embeds both angular velocity and the distribution of mass.

Although a comprehensive model would also factor in the cross-sectional area exposed to the flow, for STASIS, such intricacies are mitigated by the uniformity of the density distribution of the platform structure. Consequently, the mass distribution is utterly representative of the volumetric distribution of STASIS. As a result, the proposed model is assessed to remain valid for the purposes of this work.

At the moment of the experiment, still there is not an accurate estimation of drag coefficients for the STASIS, therefore drag is considered as an unknown disturbance effect, taken into account to achieve simulations maximally congruent with reality.

## 2.3.   Observability analysis

The rationale behind a balancing procedure is the achievement of accurate knowledge of the offset vector, as visible in Equation (2.3). However, obtaining this may be markedly intricate in a real-world environment.

As discussed in [24], in a parameters identification scenario, one of the first steps to design an experiment is the observability assessment. In the case of a 3D pendulum-like motion,

the full observability is guaranteed without any external solicitation only if the inertia matrix is known. However, the inertia knowledge is a very strong assumption, potentially leading to significantly skewed outcomes. Indeed, this approach demands a high-fidelity CAD assembly, not only introducing susceptibility to error, but also proving impractical for a work-in-progress platform, given the modifications continuously introduced.

When this assumption is released, the system identification is affected by a *scaling problem* [19]. Indeed, being $\mathbf{x} = [\omega_x \ \omega_y \ \omega_z \ J_x \ J_y \ J_z \ J_{xy} \ J_{xz} \ J_{yz} \ r_x \ r_y \ r_z]^T$ the solution vector, any scaling $\alpha\mathbf{x}$ with $\alpha \in \mathbb{R}$, still represents a solution for the problem. As a consequence, an observability discussion is fundamental for the planning of a consistent experiment, unaffected by this ambiguity.

### 2.3.1. Z-observability

In the non-linear dynamics case, as for the 3D pendulum, the observability of the system is not determined as directly as in the linear case. For this reason, the *Z-observability* concept is introduced, referring to [24, 25] definitions.

**Proposition 2.1. *Z-observability.*** *Considering a general non linear control system,*

$$\dot{\epsilon} = f(t, \epsilon, u), \ \epsilon \in \mathbb{R}^n, \ u \in \mathbb{R}^m$$
$$Y = h(t, \epsilon, u), \tag{2.6}$$

*where $\epsilon$ and $Y$ are respectively the state and the output. Let $U \in \mathbb{R} \times \mathbb{R}^n \times \mathbb{R}^m$ be an open set in the time-state-control space. The function z=Z(t,x,u) is said to be Z-observable in U with respect to the system, if for any two trajectories, $(t, \epsilon^i(t), u^i(t))$, i=1,2, in U defined on a same interval $[t_0, t_1]$, the equality*

$$h(\epsilon^1(t), u^1(t)) = h(\epsilon^2(t), u^2(t)), \ a.e. \ in \ [t_0, t_1] \tag{2.7}$$

*implies*

$$Z(t, \epsilon^1(t), u^1(t)) = Z(t, \epsilon^2(t), u^2(t)), \ a.e. \ in \ [t_0, t_1] \tag{2.8}$$

*almost everywhere in $[t_0, t_1]$. Suppose that there exist $U_1 \in U$ such that $(t, \epsilon(t), u(t)) \in U_1$ and $Z(t, \epsilon(t), u(t))$ is observable in $U_1$. Then, the function $z = Z(t, \epsilon, u)$ is said to be locally observable in U.*

This mathematical framework is substantially resuming observability as a condition wherein, identical measurement histories are generated if and only if the trajectories are identical as well. Reference [25] also proposes an observability lemma, which is very powerful to

assess the observability of complex systems.

**Proposition 2.2. *Z-observability (lemma).*** *Considering a system without control*

$$\dot{\epsilon} = f(t, \epsilon), \ \epsilon \in \mathbb{R}^n,$$
$$Y = h(t, \epsilon). \tag{2.9}$$

*Let $U \in \mathbb{R} \times \mathbb{R}^n$ be an open set in the time-state space. Consider*

$$\boldsymbol{V} = (\boldsymbol{Y}^T, D\,\boldsymbol{Y}^T, ..., D^{l-1}\,\boldsymbol{Y}^T)^T \tag{2.10}$$

*for some $l>0$, where $D$ is the differentiation operator. If*

$$rank\left(\frac{\partial \boldsymbol{V}}{\partial \epsilon}\right) = n \tag{2.11}$$

*for $(t, \epsilon) \in U$, then $z = Z(t, \epsilon)$ is locally observable in U.*

After establishing the theoretical framework, *Z-observability* shall be tailored to the specific problem at hand. First, it is important to highlight that the presented observability lemma has been streamlined to be directly applicable to the problem [24], whereas the form presented in [25] presents further intricacies.

At this point the relevant quantities specifically for the 3D-pendulum motion shall be identified. It should be evident to the reader that $\mathbf{Y}$ represents the measurement vector, that is the system output. The next consideration is determining the extent of its derivative in the observability demonstration. That is, identifying the value of $n$ for the 3D-pendulum motion observation. The underlying theory of *Z-observability* is intricate, and for a deeper understanding, readers can refer to [25]. However, when applied to observing the specific dynamics of the 3D pendulum, the theory can be streamlined. Substantially, the requirement is for $z = Z(t, \epsilon)$ to be a function of $(\mathbf{Y}^T, D\mathbf{Y}^T, ..., D^{l-1}\mathbf{Y}^T)^T$. Given that the Euler rigid body equations encompass only $\boldsymbol{\omega}$ and $\dot{\boldsymbol{\omega}}$, it can be conclusively determined that $\mathbf{Y} = \boldsymbol{\omega}$, $D\mathbf{Y} = \dot{\boldsymbol{\omega}}$. Therefore, $n = 6$, independently from any assumption on the knowledge of the inertia matrix of the platform.

Hence, the two parameters identifications, differing for whether or not including the inertia parameters in the estimation procedures, can be studied separately. In the case that the inertia matrix is supposed to be known, the state is $\mathbf{x} = [\boldsymbol{\omega} \ \mathbf{r}]^T$, hence:

$$\frac{\partial \mathbf{V}}{\partial [\boldsymbol{\omega} \ \mathbf{r}]} = \begin{bmatrix} I_{3\times3} & 0_{3\times3} \\ \frac{\partial D\mathbf{Y}}{\partial \boldsymbol{\omega}} & \frac{\partial D\mathbf{Y}}{\partial \mathbf{r}} \end{bmatrix}. \tag{2.12}$$

Being rank$(I_{3\times3}) = 3$, to prove that the rank of the matrix is equal to 6, it is sufficient to demonstrate that rank$(\frac{\partial D\mathbf{Y}}{\partial \mathbf{r}}) = 3$.

$$\frac{\partial D\mathbf{Y}}{\partial \mathbf{r}} = \frac{\partial}{\partial \mathbf{r}} J^{-1}(\mathbf{r} \times m\mathbf{g}) = J^{-1}m(-[\mathbf{g}\times]I_{3\times3}). \tag{2.13}$$

Hence, the system is observable if all the components of the gravity vector are non null in the body frame. Essentially, to ensure observability, it is sufficient for the platform to undergo a tumbling motion during the observation window.

As the hypothesis of exactly knowing the inertia parameters is released, the state of the problem shall include also the inertia parameters, therefore $\mathbf{x} = [\boldsymbol{\omega}\ \tilde{\mathbf{J}}\ \mathbf{r}]^T$. Where $\tilde{\mathbf{J}}$ is an array containing the inertia values.

$$\tilde{\mathbf{J}} = [J_x\ J_y\ J_z\ J_{xy}\ J_{xz}\ J_{yz}]^T. \tag{2.14}$$

In this case, an external control is required to achieve an unambiguous identification. Indeed, the core hypothesis underpinning the subsequent analysis is:

$$\text{rank}\left(\frac{\partial \mathbf{V}}{\partial[\boldsymbol{\omega}\ \tilde{\mathbf{J}}\ \mathbf{r}]}\right) = 5 < n. \tag{2.15}$$

To validate this rank assertion, 3 methodologies can be exploited:

1. **Linear dependency analysis**: Demonstrate that a row (or column) within the matrix can be constructed as a linear blend of other rows (or columns). If such a dependency exists, it implies redundancy in information, effectively resulting in reduced matrix rank.

2. **Determinant inspection**: Establish that every $6 \times 6$ sub-matrix holds a determinant of zero, while at least one $5 \times 5$ sub-matrix exhibits a non-zero determinant.

3. **System equivalence argument**: Highlight the rank deficiency by demonstrating that a linear combination of the system parameters can yield the same output.

The third approach, based on system equivalence, is especially insightful. The initial two methodologies demand strict assumptions on the platform's inertia matrix. Even with these assumptions, the analytical computations remain complex. Conversely, the system equivalence logic is straightforward: given any scalar $\alpha \in \mathbb{R}$, a system characterized by

scaled parameters $\alpha J, \alpha \mathbf{r}$ produces an acceleration described as:

$$
\begin{aligned}
\dot{\boldsymbol{\omega}} &= \frac{J^{-1}}{\alpha}(-\boldsymbol{\omega} \times \alpha J\boldsymbol{\omega} - m[\mathbf{g}\times]\alpha\mathbf{r}) \\
&= \frac{J^{-1}}{\alpha}\alpha(-\boldsymbol{\omega} \times J\boldsymbol{\omega} - m[\mathbf{g}\times]\mathbf{r}) \\
&= J^{-1}(-\boldsymbol{\omega} \times J\boldsymbol{\omega} - m[\mathbf{g}\times]\mathbf{r})
\end{aligned}
\tag{2.16}
$$

that is, the same acceleration of the non-scaled system. The scaling invariance proves the need to apply an external control to provide observability, being related to a reduced rank of the observability matrix.

Hence, as highlighted in [24, 25], the control is appended to the state as following:

$$
\frac{\partial \mathbf{V}}{\partial [\boldsymbol{\omega}\ \tilde{\mathbf{J}}\ \mathbf{r}\ \mathbf{u}]} = \begin{bmatrix} I_{3\times 3} & 0_{3\times 9} & 0_{3\times 3} \\ \frac{\partial D\mathbf{Y}}{\partial \boldsymbol{\omega}} & \frac{\partial D\mathbf{Y}}{\partial [\tilde{\mathbf{J}}\ \mathbf{r}]} & \frac{\partial D\mathbf{Y}}{\partial \mathbf{u}} \end{bmatrix}.
\tag{2.17}
$$

Equation (2.17) is fundamental to understand the high effectiveness of the concept of *Z-observability* in the case of study. Indeed, the control variable is treated as a state variable, which is convenient in the case that is the control itself to guarantee observability.

Hence, in the case that the control is provided by the variation of the angular momentum of RWs, the control variable assumes the form:

$$
\mathbf{u} = [\dot{h}_{wx}\ \dot{h}_{wy}\ \dot{h}_{wz}]^T.
\tag{2.18}
$$

As in the previous case, to infer that the rank of the matrix is equal to 6, it is sufficient to prove that $\text{rank}(\frac{\partial D\mathbf{Y}}{\partial \mathbf{u}}) = 3$. Therefore, considering that

$$
\frac{\partial D\mathbf{Y}}{\partial \mathbf{u}} = -\frac{\partial}{\partial \dot{\mathbf{h}}_w} J^{-1}(\dot{\mathbf{h}}_w + \boldsymbol{\omega} \times \int \dot{\mathbf{h}}_w\ dt),
\tag{2.19}
$$

and conjecturing sinusoidal RWs trajectory,

$$
\frac{\partial D\mathbf{Y}}{\partial \mathbf{u}} = -J^{-1}(I_{3\times 3} + [\boldsymbol{\omega}\times](1 - \cos t)I_{3\times 3}).
\tag{2.20}
$$

Consequently, the system is observable if all the components of the torque provided by RWs are non-null. As a result, to ensure continuous observability on the time window, RWs trajectories shall be characterized by same frequency but different phases, to avoid multiple zero-crossings in the same instant.

## 2.4. State of the art

Provided the major exigence of minimizing the disturbance torque due to the CM displacement, several recent publications have discussed different methodologies to approach the problem of balancing automatically a three-axis spacecraft simulator.

Reference [19] recognizes 3 different categories of algorithms for platform balancing:

- the *open loop* approach, which is based on CAD models to properly place the masses and eliminate the gravity torque. Despite the conceptual simplicity of the approach, its primary limitations are its susceptibility to errors and the challenges associated with implementing a high-fidelity CAD model.

- the *observe and compensate* approach, which relies on the full observation of the motion of the platform to deduce the inertia properties, according to a faithful mathematical model of the platform.

- the *closed loop* approach, which relies on control laws of the moving masses to cancel the disturbance torque due to the CM-CR offset. In this case, the trade-off for achieving high accuracy and robustness against unmodeled effects is the significant computational and implementation demands.

Generally, the first approach is chosen for a preliminary rough estimation, primarily due to the inherent constraints regarding the fidelity of a CAD model, particularly for complex platforms. Furthermore, factors like production tolerances and the presence of non-rigid components (e.g., cables) introduce additional errors. Subsequently, one of the alternative approaches is employed to enhance the precision of the offset compensation.

### 2.4.1. Observe and compensate approach

Within this category, distinct sub-categories can be identified. Reference [26] reports a complete review of the balancing techniques historically considered. The significance of this work also stems from the comprehensive analysis of the advantages and disadvantages of the various techniques, therefore constituting a baseline for any trade-off on the batch estimations of a spacecraft simulator parameters.

### Least squares methods

This category has the advantage of low implementation effort while achieving fast convergence, however, it requires typically computational demanding operations (e.g., high-

dimension matrix inversions), necessitates substantial memory allocation (due to the accumulation of all batch data) and exhibits pronounced sensitivity to modeling inaccuracies. References [27–29] report examples of the direct offset computation through the discretization of the dynamics equations. Nonetheless, the attainable accuracy is significantly limited by the propagation of errors related to the inertia characteristics, as in these instances, they are directly derived from the CAD models.

To overcome this limit, the *torque method* strategy can be adopted, which effectively treats the inertia characteristics as additional unknowns of the problem. Consequently, besides the advantage of flexibility to any change on inertia matrix due modifications performed on the platform, unbiased estimations are obtained. References [30–32] outline the mathematical manipulation of the equations of motion to compute simultaneously both the inertia parameters and the CM-CR offset. Reference [33] highlights that the method may achieve unsatisfactory results on the products of inertia due to measurements noise, therefore a novel methodology based on modified law of conservation of angular momentum is proposed, and the higher performances are demonstrated.

It is worth noticing that the state vector can also be computed recursively, through the Recursive Least Squares Method (RLS), as in the case of [31, 34]. This last work is particularly significant, since numerical Tracking Differentiator (TD) is exploited to filter the noisy angular acceleration. Typically the equations of motion of the *torque method* are integrated, in order to prevent the noisy angular accelerations (computed numerically) from affecting the final estimation. However, the integrated form is more sensitive to the effect of unmodeled disturbance torques, as their cumulative effect becomes greater and greater. Therefore, a double TD is adopted to filter the angular accelerations, resulting in higher accuracy with respect to the integrated form. Finally, the paper shows the great number of aspects that shall be considered when facing an automatic balancing problem. Reference [35] presents a further application of the RLS for the automatic balancing of a spacecraft simulator. In this case an external torque is provided by moving the movable masses randomly for a preliminary estimation, then an innovative technique based on RLS is presented. The novelty of the technique is due to the parametrization of the influence matrix, which is the matrix containing the directions of the sliding masses. The procedure separates the nominal direction from the misalignment terms, and also considers the uncertainties on the movable masses positions. Both these sources of error are estimated in the RLS algorithm, which is declared to be fundamental in achieving the final accuracy. Finally, the residual offset in the horizontal plane is compensated.

## Classical Levenberg Marquardt estimation

Classical Levenberg Marquardt (CLM), is adopted to determine the minimum of a multivariate function, which in the case of [31] is the sum of the squares error between real and expected data. This algorithm has the main advantage of being very flexible with respect to the difference between the initial guess of parameters and the optimal values. Specifically, the algorithm works similarly to the Gradient Descend method when the initial guesses are far from the effective values, and as the Gauss–Newton algorithm in the case that the initial guesses lie in the neighbourhood of the real values.

Reference [31] enables to conclude the higher accuracy of the method with respect to RLS, and lower sensitivity to disturbances and uncertainties.

## Filtering methods

When the process is highly affected by noise the Kalman filtering and its extensions may provide optimal solutions. A great benefit of these techniques is memory saving, since data are not gathered, being recursive methodologies. Additionally, computational demanding operations, as high-dimensions matrix inversion, may be avoided.

Reference [24] offers an interesting insight on the mathematical model adopted for the Extended Kalman Filter (EKF) and the Unscented Kalman Filter (UKF), and the augmented dynamics, including also the inertia terms, is presented for the EKF. In this case the full observability of the 9 parameters can be guaranteed through the innovative definition of *Z-observability* reported in [25].

Regarding the EKF, the implicit linearization of the dynamics around the current estimate, may lead to disappointing results, as observed in [34]. In this case the problem is addressed to low acquisition frequency of sensors, which reduces the accuracy of the linearization. Another drawback of the EKF is the need to calculate the Jacobians, whose analytical computation may be so complex that reference [24] proposes the use of Complex Step Differentiator, to obtain it numerically. Reference [36] highlights that the numerical Jacobian computation may increase the computational effort of the procedure. To overcome this limit, reference [37] approaches the computation with a closed-form solution, which is declared to be more accurate and computationally efficient than a finite difference method approximation. The paper develops different applications of the Kalman filtering, making use of dual-filtering procedures and joint-filters, in which the set of parameters to be estimated are appended to the state vector. These procedures are applied for both the EKF and the Iterated Extended Kalman Filter (IEKF), whose element of novelty is that the Jacobian of the measurements is computed around the *new* estimate of the state vector, instead of the *old*. Consequently, the Kalman gain computation is an implicit

equation and shall be cycled at each step until convergence is reached. It is worth noticing that the convergence properties of the filter can be enhanced through a pre-filtering procedure, as presented in [38], where a Savitzky-Golay filter allows to retrieve precisely inertia parameters of the platform.

On the other hand, UKF does not require any linearization around the current estimate, resulting not only in lower proneness to error propagation, but also in lower time for convergence and higher accuracy. This is due to the strong non-linearities that may affect the platform dynamics (e.g., significant off-diagonal terms of the inertia matrix, magnetic actuation [39]). It is worth noticing that the convergence performances of this methodology can be improved through a proper design of the momentum trajectory, as presented in [40], where an Optimal Control Problem (OCP) is resolved to keep the matrix $\mathbf{\Phi}$ well-conditioned, therefore minimizing its conditioning number. Where $\mathbf{\Phi}$ is the matrix mapping the augmented state, made of inertia parameters and the CM-CR offset, into the external torque.

$$\mathbf{\Phi} \begin{bmatrix} J_x & J_y & J_z & J_{xy} & J_{xz} & J_{yz} & mr_x & mr_y & mr_z \end{bmatrix}^T = \mathbf{T}_{ext}. \tag{2.21}$$

Nonetheless, the UKF requires higher computational effort than the EKF due to the computation of the $\sigma$-*points*, which are then propagated through the non-linear model of the platform to update the state estimate and covariance. For this reason, reference [26] suggests as future development the implementation of complexity reduction techniques for the UKF, as reported in [41], where the structure of the state vector and measurements models are exploited to reduce the computational load.

### 2.4.2.   Closed-loop approaches

This category of techniques consists of a balancing accomplishment through the use of the mounted actuators, to compensate the torque generated by the CM-CR offset. Typically the actuators involve a set of sliding masses, movable in three orthogonal directions to ensure a fully controllable system.

A very simple methodology is the manual balancing, where an operator recursively adjusts movable masses positions until a satisfactory residual torque is reached. Despite its tempting conceptual simplicity, manual balancing has the drawback of very limited accuracy. Specifically the published works in literature demonstrate a reachable value of residual torque of 0.01 N m [17, 42]. A further drawback is that manual balancing requires hours of work of the operator, which is particularly demanding as the procedure shall be repeated every time a change in the satellite mock-up is performed.

Discharged the human-in-the-loop solutions, the active control techniques may be divided into 2 categories:

- Under-actuated control techniques, in which the actuators adopted for the balancing procedure are only the movable masses.

- Fully actuated control techniques, where another set of actuators (e.g., CMGs, RWs) is involved in the gravity torque compensation.

The nomenclature arises from the system's inherent dynamics. Specifically, when the torque is produced by the group of shifting masses, it operates within a plane that is perpendicular to the direction of gravitational force, meaning that the control problem is under-actuated. To withstand the non full controllability of the system, in many works in literature a two-step procedure is proposed [13, 43]. The first step is devoted to an x-y plane CM offset compensation, while the vertical compensation is achieved in the second step, involving a batch estimation through the techniques discussed previously.

The main advantage of this technique is the simplicity of the system design, as no further actuators are required besides the movable masses. This feature is particularly effective when the system is susceptible to the disturbance that may arise from the vibrations of the actuators, which is declared to be one of the main issues for an accurate estimation in [31]. On the other hand, a fully actuated control technique, which requires a set of actuators capable to generate any three-dimensional torque vector, may overcome the other methods with a similar hardware (e.g., sensors). Moreover, this approach achieves balance in a single step, eliminating the need for batch estimations for vertical compensation.

It is worth noticing that various control methodologies within this category can be utilized to address the balancing problem. Presently, the methods implemented include PID, non-linear, and adaptive controls.

The central premise underlying all the aforementioned methodologies is that the CM-CR offset stands as the primary disturbance in the system. Thus, designing a control capable to stabilize the system, intrinsically cancels the offset. Therefore, the balancing is treated as a classical non linear attitude control problem, deeply described in [44, 45], where the convergence implies a full or partial offset compensation.

## PID control

PID control is the most diffused method for solving an automatic balancing problem. The widespread use primarily stems from the ease of implementation and the commendable accuracy, which is on par with that achieved by the latest techniques (e.g., non-linear, L1

adaptive controls). The working principle essentially involves applying a control torque composed of three distinct contributions: the first is proportional to the attitude error, the second to the cumulative attitude error, and the third to the angular rate [46]. For the under-actuated technique, the attitude error is defined by the discrepancy between the current kinematics and a configuration with null pitch and roll, allowing the probe to freely spin around the yaw axis. Conversely, in the fully actuated approach, all three rotational motions are nullified. Concerning the pragmatic control setup, the control gains can be determined through the Ziegler–Nichols method, followed by a fine tuning adjustment [47], or directly through trial and error.

It is worth noticing that within the same loop also the effect of disturbances with known frequency can be compensated, as reported in [48], through the use of a Disturbance Rejection Filter (DRF). The paper, once assessed the low robustness of the DRF to frequency uncertainties, presents two different strategies to identify the disturbance frequencies in the system dynamics.

## Non-linear control

Besides the PID control, literature is rich of novel nonlinear feedback laws. These latter have typically higher performances with respect to the linear control, which may result completely ineffective if the stabilizing effect of the control is overcome by large non-linearities. Furthermore, in the PID controls, typically low gains must be selected to ensure stable control, due to the unmodeled dynamics effects. The consequent result are an higher closed-loop time constant and an amplified effect of non-linearities.

A distinctive characteristic of these nonlinear control laws is the introduction of a multi-variate function, namely the Lyapunov function. This function is zero at the desired state and positive elsewhere. Hence, the control torque is designed to assure the time-derivative of the Lyapunov function to be semi-definite negative, and null in correspondence of the desired kinematics. As in the case of PID, the target kinematics differs between under-actuated case, characterized by a rotation around yaw as target state, and fully actuated case, where all rotations are canceled. Reference [13] introduces one of the most compelling techniques, proposing in the first step of a dual-stage procedure the Lyapunov function:

$$V = \frac{1}{2}\boldsymbol{\omega}^T J \boldsymbol{\omega} + \frac{1}{2}K_P(\hat{\mathbf{g}} - \hat{\mathbf{z}})^T(\hat{\mathbf{g}} - \hat{\mathbf{z}}), \tag{2.22}$$

where $\hat{\mathbf{g}}$ is the local vertical direction and $\hat{\mathbf{z}}$ is the longitudinal direction in the body frame. Thus, the Lyapunov function implicitly embeds the kinetic energy and the error between current and target kinematics. The work, as an admissible control feedback is

proposed, offers a mathematical proof of the asymptotic stability of the control. The designed control is tuned heuristically, demonstrating superior performance compared to a PID control. Furthermore, the only kinematic data needed by the control is the gravity vector in the body frame, differently from the PID control, which requires a comprehensive set of kinematic parameters.

The second step of the balancing algorithm is a Least Squares (LSQ) estimation of the residual offset, which is used for the further cycles of the plane balancing. This is a key feature of the strategy in achieving high performances, as, after the first iteration, the closed-loop algorithm relies on accurate information of the inertia characteristics of the platform. In this step, the Savitzki-Golay filtering of the angular rate measurements is assessed to be of paramount importance, allowing to obtain better results of the integrated form of the LSQ method.

References [43, 49, 50] put forth a Lyapunov function incorporating both the energetic term and the estimation error $\tilde{\mathbf{r}}(t) = \mathbf{r} - \hat{\mathbf{r}}(t)$, representing the discrepancy between the actual and estimated offset vector. The primary distinction among these works is that both [43, 50] incorporate a kinematic term as illustrated in Equation (2.23), wherein $\mathbf{q}$ is the quaternion vector. However, this term is nullified in subsequent calculations.

$$V = \frac{1}{2}\boldsymbol{\omega}^T J \boldsymbol{\omega} + \frac{1}{2}\tilde{\mathbf{r}}^T \tilde{\mathbf{r}} + \frac{1}{2}\mathbf{q}^T \mathbf{q}. \tag{2.23}$$

This balancing approach leans on the conservation of angular momentum. It formulates a control law adjusting the movable masses until the platform's angular momentum derivative becomes zero. Hence, as proved in [43, 50], the time-derivative of the Lyapunov function can be expressed as $\dot{V} = -K_P ||\boldsymbol{\omega}_\perp||^2$. Therefore, being it negative semi-definite, the proof of the asymptotic stability of the system is provided. Being $\boldsymbol{\omega}_\perp$ the transverse angular velocity, the control law drives the system to a rotation around the vertical axis. Furthermore, utilizing the LaSalle Invariance Principle, references [43, 50] prove that the system converges to a state such that $\boldsymbol{\omega}_\perp \to 0$, i.e., exact compensation of the horizontal offset in absence of disturbances. The second step consists in a Kalman filtering of the pendulum-like motion to retrieve the information on the vertical offset. It is pivotal to note that in this phase, for ensuring complete system observability, the platform undergoes tumbling motion, preventing the condition of $[g_x \ g_y] = [0 \ 0]$ in the body frame.

Finally, reference [30] provides a fully actuated strategy for the automatic mass balancing through a set of Control Moment Gyros (CMGs). In this case the candidate Lyapunov function is

$$V = \frac{1}{2}\mathbf{H}^T\mathbf{H} + \frac{1}{2}\mathbf{r}^T\boldsymbol{\Gamma}^{-1}\mathbf{r}, \tag{2.24}$$

where $\mathbf{H}$ is the angular momentum of the platform, $\mathbf{r}$ is the offset, and $\boldsymbol{\Gamma}$ is a positive

definite matrix. Nevertheless, a control derived from this function might result in a state where the angular momentum is preserved even if the CM does not align with the CR. Indeed, the zero-vector solution is introduced, meaning the CM is displaced from the CR along the local vertical direction. This indicates that the proposed Lyapunov function is not suitable for use. Differently from concurrent works, in [30] the issue of partial observability is not addressed using a two-step procedure. The proposed solution to this problem is to generate a desired spacecraft momentum trajectory $\mathbf{H}_d$ leading to complete offset compensation

$$V = \frac{1}{2}(\mathbf{H} - \mathbf{H}_d)^T(\mathbf{H} - \mathbf{H}_d) + \frac{1}{2}\mathbf{r}^T\mathbf{\Gamma}^{-1}\mathbf{r}. \tag{2.25}$$

## L1 adaptive control

L1 adaptive control is a novel methodology able to achieve a fast adaptation without sacrificing the robustness [51]. The key feature of this type of control is the decoupling of the adaptation loop from the control loop. Specifically, the rate of the adaptation loop can be set arbitrarily high, subject only to hardware limitations (e.g., computational power), enabling to achieve fast compensation of undesirable effects without affecting convergence. Reference [52] presents a clear L1 adaptive control algorithm flow, whose key features are the state predictor step, whose dynamics can be prescribed according to a reference model, and the utilization of projection techniques in the adaptive law.

References [47, 52] compare the results of an adaptive scheme with those attainable from a PID controller, proving the higher performances of the first. Specifically, in [47] both the PID and L1 control schemes are compared in nominal operation, in case of perturbed values of mass and inertia, and in case of sudden external disturbances. In all these scenarios the adaptive scheme achieves equal or better results than the PID controller, both for the settling time and steady state error. In particular, the L1 control maintains the same results on accuracy in all the different scenarios, thereby proving its high robustness to uncertainties and external disturbance rejection.

### 2.4.3. Disturbances and performance assessment

The significance of the literature is further underscored by the extensive list of disturbances presented [15], and also the empirical methods used to attribute sub-optimal performances to a specific disturbance. For instance, reference [30] suggests that if the momentum error has peaks in correspondence of the maximum or minimum attitude, the main source of error is due to structural sagging, which can be minimized performing a

calibration and sensitivity test, so relating the CM shift with the platform tilt; reference [31] instead, suggests to seek for any match between the high error and the high RWs velocity, since in that case the vibrations are expected to be the main disturbance; and reference [43] suggests to monitor the angular rate around the vertical direction, this to inspect the effect of aerodynamic drag.

Modeling these perturbations may be very challenging as, an effective emulator of the disturbance shall be designed without affecting excessively the computational effort of the simulations. Reference [23] offers a simple model for the aerodynamic drag effect on a rotational dynamics, considering the body surrounded by a linearly resisting medium. The main assumption of the work is that the diagonal viscous damping torque tensor is proportional to the moment of inertia tensor, implying a resisting torque proportional to the angular momentum

$$\mathbf{M}_{drag} = -\varepsilon \Lambda J \boldsymbol{\omega}, \tag{2.26}$$

where $\varepsilon$ is a small number and $\Lambda$ is a coefficient accounting for medium properties. On the other hand, reference [29] models drag as a vector proportional to the square of the angular rate

$$\mathbf{M}_{drag} = - \begin{bmatrix} B_x \omega_x^2 \\ B_y \omega_y^2 \\ B_z \omega_z^2 \end{bmatrix}, \tag{2.27}$$

where $B_x, B_y, B_z$ are aerodynamic drag coefficients. Literature also provides insights into the disturbances impacting performance. In numerous references, an accurate modeling of these perturbations is highlighted as the primary area for further enhancements.

Furthermore, the mentioned studies introduce a variety of methods for performance assessment. Since the objectives of the balancing processes vary, it might be challenging to establish a universal performance metric. Nevertheless, there are some shared criteria for evaluating performance. For instance, computational effort can be quantitatively gauged across all balancing algorithms. On the other hand, accuracy and robustness can be evaluated only qualitatively, since a precise assessment would require the exact knowledge of the centre of mass position.

In references [35, 53] to evaluate the final distance between the CM and the CR the oscillation period of the test-bed is measured. In the first case the oscillation period is retrieved from the frequency spectrum of the roll and pitch angles signals, therefore involving a Fourier transform for the signal decomposition, whereas in the second work it is deducted with a proper LSQ fitting of the pendulum motion, to filter out the undesired effects. The

offset is therefore evaluated through the following relationship

$$T_p = 2\pi \sqrt{\frac{J_s}{mg||\delta\mathbf{r}||}}, \tag{2.28}$$

where $T_p$ is the period, $J_s$ the inertia of the axis of rotation, $\delta\mathbf{r}$ the residual offset.

Reference [43] presents a further indirect technique to estimate the offset, which involves the kinetic energy. In the hypothesis of negligibility of disturbances, the exact compensation implies constant gravitational potential energy, and consequently constant kinetic energy. Although the presence of dissipative effects cause a kinetic energy secular reduction, the effectiveness of the technique is not compromised. Indeed, the performance index is the amplitude of the oscillations, independently on the effective value. Specifically, a properly executed balancing procedure, should result in significantly reduced amplitude of the kinetic energy oscillations.

Finally, reference [13] presents a direct estimation of the residual gravity torque simply considering the Euler Equations of Motion for a rigid body [44, 54], therefore equating the time-derivative of the angular momentum to the external torque. However, as this technique utilizes the angular rate $\boldsymbol{\omega}$ and the angular acceleration $\dot{\boldsymbol{\omega}}$, a preliminary filtering of the measurements shall be performed to achieve significant results.

It is imperative to note that these methodologies are not definitive benchmarks for comparing different balancing algorithms. The outcomes are multifaceted, influenced by a myriad of factors including platform dimensions, sensor noise and bias, and external torque excitation. Hence, while they provide valuable insights, they should be viewed in the context of the broader parameters influencing the final results.

# 3 | Automatic balancing algorithms

In this chapter, different algorithms are individually presented to assess attainable performances of each of them. The rationale behind developing multiple algorithms stems from the overarching objective of integrating them. This integration seeks to capture the distinct advantages of each algorithm, culminating in a unified automatic balancing procedure, capable to transcend the inherent limitations of each standalone method. This synergistic approach is designed to yield a comprehensive balancing procedure, specifically optimized for the operational exigencies of the STASIS platform.

## 3.1. STASIS digital twin

Prior to delving into the intricacies of the balancing algorithms, this study elucidates the digital simulation framework employed. The formulation and refinement of the balancing algorithm have been facilitated by the utilization of the platform's digital twin, which is substantially a digital model of a real cyber-physical system for its monitoring and functional improvement [55]. Reference [22] deeply describes the procedure behind the multi-physics modeling of STASIS. The work acknowledges the intricate nature of pinpointing a clear-cut definition for a digital twin. Nevertheless, its core attributes are identified as reliability, the capability of supporting a continuous stream of data, and adaptability. The inherent challenge, then, lies in crafting an accurate model that not only embodies these features but is also characterized by a well-organized code architecture, ensuring ease of future developments.

The work, following an initial evaluation of potential software options, favors the SimScape[1] environment over alternatives like Simulink[2] or OpenModelica[3] [22]. This decision

---

[1] Mathworks, "SimScape Multibody", https://it.mathworks.com/products/simscape-multibody.html, last accessed: 16/05/2023.

[2] Mathworks, "Simulink", https://it.mathworks.com/products/simulink.html, last accessed: 16/05/2023.

[3] OpenModelica, "OpenModelica", https://openmodelica.org/, last accessed: 16/05/2023.

aligns with the goal of executing an *A-causal* model of the platform. Such a model empha-sizes the representation of the physical connections between blocks, rather than detailing the computational procedures linking them, which is characteristic of *causal* modeling instead [22, 56–58]. MathWorks[4] synthesizes the difference between the 2 approaches for a simple mechanical system, as visible in Figure 3.1.



(a) Simulink model.



(b) SimScape model.
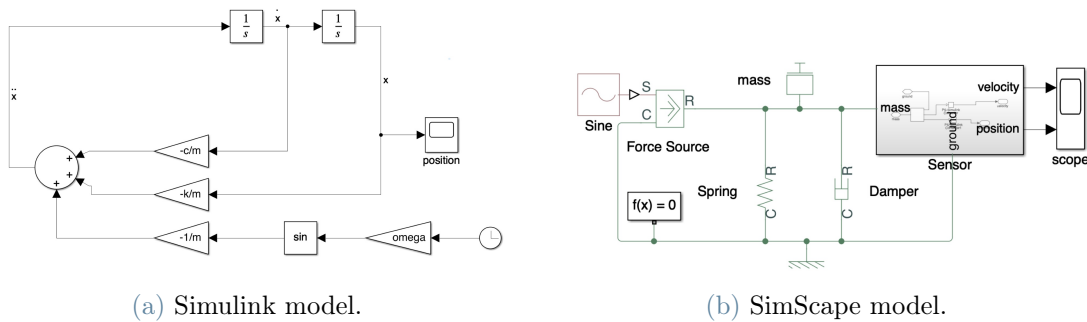
Figure 3.1: *Causal* (left) and *A-causal* (right) modeling of a Mass–Spring–Damper system [22].

Opting for *A-causal* modeling offers significant benefits in terms of re-configurability and composability, attributes that become paramount when dealing with a sophisticated work-in-progress platform.

Moreover, an invaluable feature of SimScape is its ability to directly import CAD assem-blies, complete with mass and inertia properties. This provides an intuitive visual inter-face for observing the system's dynamic responses. In addition, the SimScape Multibody library facilitates the inclusion of blocks representing joints, bearings, and screws, thus further elevating the authenticity and the expeditiousness of implementation (Figure 3.2).

---

[4]MathWorks, "Mass-Spring-Damper in Simulink and Simscape", `https://uk.mathworks.com/help/simscape/ug/mass-spring-damper-in-simulink-and-simscape.html`, last accessed: 16/05/2023

(a) CAD model.

(b) SimScape CAD assembly.
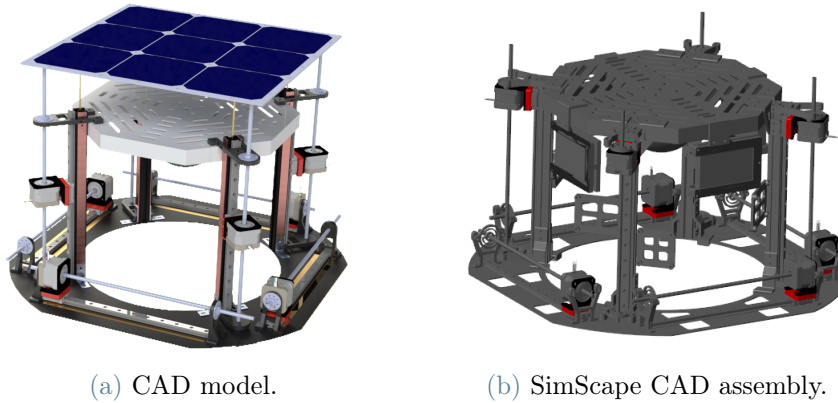
Figure 3.2: CAD comparison.

It is important to acknowledge that multi-domain problems can be plagued by computational stiffness, potentially leading to numerical instability unless an exceedingly small step size is employed. As outlined in [59], while the basic concept is intuitive to mathematicians, a definite metric to establish whether a system is stiff or not is impossible. Nonetheless, the stiffness index, presented in Equation (3.1), can provide a quantifiable parameter to judge the degree of stiffness in a system.

$$S = \frac{max|Re(\lambda_i)|}{min|Re(\lambda_i)|}. \tag{3.1}$$

In common practices, $S$ is established to be "very high" if in the order of $10^6$ or higher [59]. Being $\lambda_i$ the eigenvalues of the Jacobian, it is implied that this definition demands the system to be linearized around a stable equilibrium point. However, this simple definition hints that, in a multi-domain system stiffness arises when different components of the state exhibit significantly diverse timescales. Indeed, this would be related to a significant disparity in the magnitudes of eigenvalues [59, 60].

To address this challenge, the algorithms are crafted within a modified digital twin, wherein the transient response of the stepper motor is neglected. This essentially assumes the motor's dynamics to be infinitely faster than the platform's dynamic response. The rationale behind this approach is to sidestep potential bottlenecks during the developmental phase of the algorithm.

## 3.1.1. Model simplification

The primary aim of model block reconfiguration is to ensure accurate results during the algorithm's developmental stage without stalling due to the extensive computational time

demanded by stiff systems. As such, the stepper motor block's simplicity became pivotal, even at the cost of overlooking its transient phase.

In a first attempt, modifications to the digital twin were minimal, therefore modeling the system as a torque imparted to a lead screw to mobilize the mass. The objective was to maintain a consistent angular rate, thus, the torque was initially conceptualized as a Dirac $\delta$ function. However, given SimScape limitations in handling mathematical limits, narrow square waves were used instead. This yielded a trapezoidal angular rate pattern, which more closely resembled the desired behavior as the wave width approached zero. The drawback of this method was the trial and error tuning needed to align with reference [22]. Furthermore, the masses in this model were subjected to external forces (e.g., gravity), causing a potential disparity between the desired and actual mass' shifts.

Consequently, a more rigorous modeling have been approached, in which the motion command is directly translated to an angular position on the screw, while the torque is automatically computed to satisfy the command by the revolute screw block (Figure 3.3).
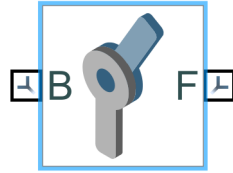


Figure 3.3: Revolute screw SimScape Multibody block.

In this configuration, also the first and second derivative of the motion shall be provided to the SimScape revolute screw block. Table 3.1 reports the performances of the simplified model compared to the original model [22], for a 10 seconds dynamics simulation, proving the utmost importance of the model reconfiguration. Indeed, in this setup, the algorithm design is considerably accelerated.

Table 3.1: Simplified model performances.

|              | Model | Simplified model |
| ------------ | ----- | ---------------- |
| **Time [s]** | 98.7  | 2.8              |

## 3.1.2.   Sensor implementation

In the STASIS digital twin, angular rates and attitude are directly outputted without modifications. However, for a more authentic simulation mirroring a real-world experi-

ment, it is essential to incorporate sensors behaviour.

The reference IMU for the simulation is the Bosch BNO055[5] sensor, already available in the ESH. The Power Spectral Density (PSD) of the IMU noise, whose value is presented in Equation (3.2), is reported on its datasheet.

$$S(f) = 0.014°/s \sqrt{Hz}. \tag{3.2}$$

The PSD allows to compute the variance of the noise, as reported in Equation (3.3),

$$\sigma_\omega^2 = \int_{f_{min}}^{f_{max}} S(f)^2 \, df \tag{3.3}$$

where $f_{min}$ and $f_{max}$ define the bandwidth of the noise.

For white noise, the PSD remains constant across all frequency values:

$$\sigma_\omega^2 = S(f)^2 (f_{max} - f_{min}). \tag{3.4}$$

Therefore, to introduce the noise in the simulations, it is necessary to know the bandwidth. As the adopted sensor operates at 100 Hz, it is inferred that the Nyquist frequency is 50 Hz, that is half of the sampling rate. This means that the sensor can correctly capture frequency components up to 50 Hz, and hence the relevant frequency band for noise would be from 0 Hz to 50 Hz

$$\begin{aligned} \sigma_\omega &= 0.014°/s \sqrt{Hz} \sqrt{(50-0) \, Hz} \\ &= 0.09899°/s \\ &\approx 0.1°/s. \end{aligned} \tag{3.5}$$

It should be noted that white noise does not represent the sole source of error. Measurement quantization can also introduce considerable inaccuracies.

Considering that the selected IMU sensor has a resolution of 16 bits in the gyroscope mode, then there are $2^{16}$ distinct levels that the sensor's output can assume. Therefore, considering the nominal range of $125°/s$, the following quantization interval have been incorporated in the simulations,

$$Q_I = \frac{125°/s \frac{\pi}{180°}}{2^{16}} \, rad/s. \tag{3.6}$$

Coherently, attitude parameters have been corrupted adding random noise on the Euler angles knowledge. This approach was based on preliminary studies conducted at the

---

[5]BOSCH, "Smart Sensor BNO055" https://www.bosch-sensortec.com/products/smart-sensors/bno055/, last accessed: 19/09/2023.

DART laboratory by doctoral students. Following their recommendations, the variances for the ground truth attitude reconstruction system were established as $\sigma_\phi = \sigma_\theta = 15$ arcsec and $\sigma_\psi = 1$ arcsec. The attitude reconstruction does not rely on any previous measurements, being directly retrieved at each time instant from the spacial distribution of the LEDs placed on the platform. As a consequence, bias instability is not incorporated in the digital twin simulations.

The rationale behind different approaches for the models of attitude knowledge, wherein the error variance is directly introduced, and IMU, for which a prior analysis of the PSD is conducted, is the higher criticality of the noise of this latter. Indeed, iterative tuning of the IMU acquisition frequency may be developed to achieve good performances. Notably, within this tuning process, boundaries shall be posed to acquire consistent measurements. As a matter of fact, reducing the acquisition frequency may procure aliasing, escalated bias instability, drift or non-linearities of the system dynamics to become more pronounced. Additionally, in a real-time procedure, latency issues may arise.

### 3.1.3.   Reaction wheels implementation

The RWs provide a torque given as:

$$\mathbf{T}_w = J_w\dot{\boldsymbol{\omega}}_w + \boldsymbol{\omega} \times J_w\boldsymbol{\omega}_w. \tag{3.7}$$

However, considering the intricacies of these systems, it is imperative to conduct a thorough analysis to achieve accurate simulations [61]. To this end, various error sources were assessed, based on a notable market reference[6].

Literature offers analytical representations of the primary dynamic disturbances within the RWs [62, 63]. Vibrations arising from both static and dynamic imbalances have been meticulously analyzed, and models have been affirmed through experimental procedures. The vibrations are evaluated in the frequency domain, noting that the predominant disturbance aligns with the rotational frequency of the RWs. However, disturbances at higher frequencies were also detected.

Furthermore, reference [64] delves into the micro-vibrations resulting from single-point contact in the wheel's mechanical structure, investigated using the Hertz contact theory. Nonetheless, integrating these insights necessitates a profound knowledge of the reaction wheel's internal configuration. Hereafter, the incorporated sources of error are detailed.

- **Quantization error.** Both the wheels angular accelerations and angular rates are

---

[6]"Faulhaber, DC Brushless Motors BX4" `https://www.faulhaber.com/it/prodotti/serie/3242bx4/`, last accessed: 01/06/2023.

quantized based on a 14-bit resolution.

- **Non-linearities.** Saturation is introduced to simulate the non-linear behaviour of the component at high velocities.

- **Friction.** The output is damped with respect to the command, according to the available dynamic friction coefficient $c_V$.

$$\mathbf{T}_f = -c_V\boldsymbol{\omega}_w. \tag{3.8}$$

- **Mass imbalances.** Vibration are introduced, with the same frequency of the wheel dynamics and with a magnitude proportional to the square of the wheel rate.

$$\mathbf{T}_v = \mathbf{T}_{max}\boldsymbol{\omega}_w^2\sin(2\pi f_w). \tag{3.9}$$

- **Gyroscopic effect.** The noise on the angular rates of the platform is propagated to the term $\boldsymbol{\omega} \times J_w\boldsymbol{\omega}_w$ in the torque knowledge.

- **Backlash.** A backlash behaviour is incorporated to simulate the dynamic behaviour during zero-crossings.

- **Delay.** A delay is introduced from command to actuation.

## 3.2.   Least squares estimation

As presented in [30–32], the EOM can be manipulated to obtain through a Least Square estimation not only the unbalance vector, but also the inertia parameters of the platform. This choice introduces the scaling problem discussed in Section 2.3, however allows to retrieve a system identification independent from the assumptions performed on the inertia characteristics of the platform. In the simulations, the full observability have been accomplished through the employment of RWs. Therefore the EOM become:

$$J\dot{\boldsymbol{\omega}} + \boldsymbol{\omega} \times J\boldsymbol{\omega} = \mathbf{T}_{ext} = \mathbf{T}_w - [\mathbf{g}\times]m\mathbf{r}. \tag{3.10}$$

Hence, considering the parameters vector to be estimated is $\mathbf{x} = [J_x\ J_y\ J_z\ J_{xy}\ J_{xz}\ J_{yz}\ mr_x\ mr_y\ mr_z]^T$, Equation (3.10) may be rearranged as follows

$$[\dot{\boldsymbol{\Omega}} + \boldsymbol{\omega} \times \boldsymbol{\Omega}\ \ [\mathbf{g}\times]] \begin{bmatrix} \tilde{\mathbf{J}} \\ m\mathbf{r} \end{bmatrix} = \mathbf{T}_w, \tag{3.11}$$

wherein,

$$\mathbf{\Omega} = \begin{bmatrix} \omega_1 & 0 & 0 & \omega_2 & \omega_3 & 0 \\ 0 & \omega_2 & 0 & \omega_1 & 0 & \omega_3 \\ 0 & 0 & \omega_3 & 0 & \omega_1 & \omega_2 \end{bmatrix} \tag{3.12}$$

$$\tilde{\mathbf{J}} = [J_x \; J_y \; J_z \; J_{xy} \; J_{xz} \; J_{yz}] \tag{3.13}$$

$$[\mathbf{g}\times]] = g \begin{bmatrix} 0 & -\cos\phi\cos\theta & \sin\phi\cos\theta \\ \cos\phi\cos\theta & 0 & \sin\theta \\ -\sin\phi\cos\theta & -\sin\theta & 0 \end{bmatrix}. \tag{3.14}$$

It is worth noticing that Equation (3.11) involves angular acceleration, which are generally affected by high noise, introduced by the numerical differentiation. To avoid relying on them, Equation (3.11) can be integrated in the time window of dynamic observation,

$$\underbrace{[\mathbf{\Omega} + \int_{t_0}^{t} \boldsymbol{\omega} \times \mathbf{\Omega}\, dt \quad \int_{t_0}^{t} [\mathbf{g}\times]\, dt]}_{\mathbf{\Phi}} \underbrace{\begin{bmatrix} \tilde{\mathbf{J}} \\ m\mathbf{r} \end{bmatrix}}_{\mathbf{x}} = \underbrace{\int_{t_0}^{t} \mathbf{T}_w\, dt}_{\mathbf{H}}. \tag{3.15}$$

To apply the LSQ, $n$ samples are considered,

$$\mathbf{\Phi}_{aug} = \begin{bmatrix} \mathbf{\Phi}_0 \\ \mathbf{\Phi}_1 \\ \vdots \\ \mathbf{\Phi}_n \end{bmatrix}_{3n \times 9} , \text{ with } \mathbf{\Phi}_k = [\mathbf{\Omega} + \int_{t_0}^{t_k} \boldsymbol{\omega} \times \mathbf{\Omega}\, dt \quad \int_{t_0}^{t_k} [\mathbf{g}\times]\, dt] \;\; k = 1, 2...n \tag{3.16}$$

$$\mathbf{H}_{aug} = \begin{bmatrix} \mathbf{H}_0 \\ \mathbf{H}_1 \\ \vdots \\ \mathbf{H}_n \end{bmatrix}_{3n \times 1} , \text{ with } \mathbf{H}_k = \int_{t_0}^{t_k} \mathbf{T}_w\, dt \;\; k = 1, 2...n. \tag{3.17}$$

Therefore, the LSQ solution is

$$\mathbf{x} = (\mathbf{\Phi}_{aug}^T \mathbf{\Phi}_{aug})^{-1} \mathbf{\Phi}_{aug}^T \mathbf{H}_{aug}. \tag{3.18}$$

An alternative approach to address the scaling problem is to impose a mathematical constraint on the final solution vector $\mathbf{x}$. This is feasible when certain components of $\mathbf{x}$ can be ascertained through a distinct procedure. For instance, given a preliminary compensation in the x-y plane, thus determining $r_x$ and $r_y$, the problem illustrated in

Equation (3.15) can be reformulated as follows

$$\mathbf{\Phi}\mathbf{x} = 0, \text{ such that } B\mathbf{x} = \mathbf{c}. \tag{3.19}$$

As highlighted in [13], in the study case, the constraint determination is straightforward:

$$B = \begin{bmatrix} 0_{1\times6} & 1 & 0 & 0 \\ 0_{1\times6} & 0 & 1 & 0 \end{bmatrix}, \; \mathbf{c} = \begin{bmatrix} mr_x \\ mr_y \end{bmatrix}. \tag{3.20}$$

This procedure is expected to be prone to propagation of errors, since $r_x$ and $r_y$ are estimations, however it allows to exclude from the methodology the RWs, which is a key feature in the case of STASIS, as detailed in Section 1.5.

## 3.2.1.  Filtering of measurements

Upon rigorous implementation of the IMU sensor in the digital framework, the numerically determined angular accelerations turned out to be indecipherable. To address this, the Savitzky-Golay filter was employed, following the recommendation in [13].
The Savitzky-Golay filter is a digital filter that can be applied to a set of digital data points in order to smooth the data, that is, to increase the precision of the data without distorting the signal tendency. This is achieved by fitting successive subsets of adjacent data points with a low-degree polynomial, typically using linear least squares. For a given window size $n$ and polynomial order $m$, the local segment of data is approximated as

$$f(x) = a_0 + a_1 x + a_2 x^2 + \cdots + a_m x^m. \tag{3.21}$$

Therefore, the Savitzky-Golay coefficients $c_i$ are computed by performing a least squares fit of the data within the window to a polynomial of order $m$. As reported in [65], the smoothed value $y_i^{SG}$ at position $i$ is given by

$$y_i^{SG} = \sum_{j=-k}^{k} c_j y_{i+j} \tag{3.22}$$

where $k$ is $(n-1)/2$ and the coefficients $c_j$ depend on both the window size $n$ and polynomial order $m$. As proposed in [38], the filter can be also used for differentiation.

The derivative is computed using a similar convolution with different coefficients,

$$\dot{y}_i^{SG} = \sum_{j=-k}^{k} g_j y_{i+j}.$$ (3.23)

Therefore, to compute the variable at the current instant, the filter uses the future acquisitions. Hence, its implementation on SimScape environment requires a buffering of data, in order to collect all the required measurements. Algorithm 3.1 presents the procedure for the input variable `omega`. It is worth noticing that, to achieve a consistent set of LSQ

---

**Algorithm 3.1** Real Time Savitzky-Golay Filtering

---
Create a persistent variable `buffered_data`
**if** `if-isempty(buffered_data)` **then**
　　Initialize buffered data as null matrix
**end if**
Slide data to make room for new input
**for** `i = 1:length(omega)` **do**
　　Data filtering
**end for**

---

matrices, also the external torque and the quaternion vector shall be artificially delayed of the same steps of the filtered data. The results of Savitzky-Golay filtering have been compared to a low-pass filtering procedure (Figure 3.5). To design such a filter, the platform dynamics is analyzed in the frequency domain. The analysis is performed through the Fast Fourier Transform (FFT), through the MATLAB command `fft`, leading to the results reported in Figure 3.4. Upon detection of both dynamics and disturbance frequencies, the filter is designed through the Butterworth technique, to achieve a maximally flat frequency response in the passband [66, 67]. To achieve a limited delay, a $4^{th}$ order filter is proposed. The cut-off frequency is set to 2.5 Hz, such as the dynamics response is not attenuated from the filter. This value is set after the imposition of margins on the dynamics frequency of the platform, and it is therefore normalized on the Nyquist frequency (i.e., 50 Hz in this case) to generate a discrete-time filter through the MATLAB function `butter`. On the other hand, the numerical filter is configured with $m = 2$ and $n = 6$, while the angular accelerations are processed according to the approach in [38].

The inherent delay in low-pass filtering causes the Savitzky-Golay filtering to yield superior outcomes, as depicted in Figure 3.5. Nonetheless, implementing a low-pass filter is an interesting back-up strategy. Indeed, if the inherent delay is strategically cancelled, its superior smoothing capability could result in highly accurate estimations. Specifically, in the constrained case the LSQ equation is given as $\mathbf{\Phi x} = 0$. Since the equation is set to a null vector, the delay from a low-pass filter can be effortlessly offset. In fact, if the same
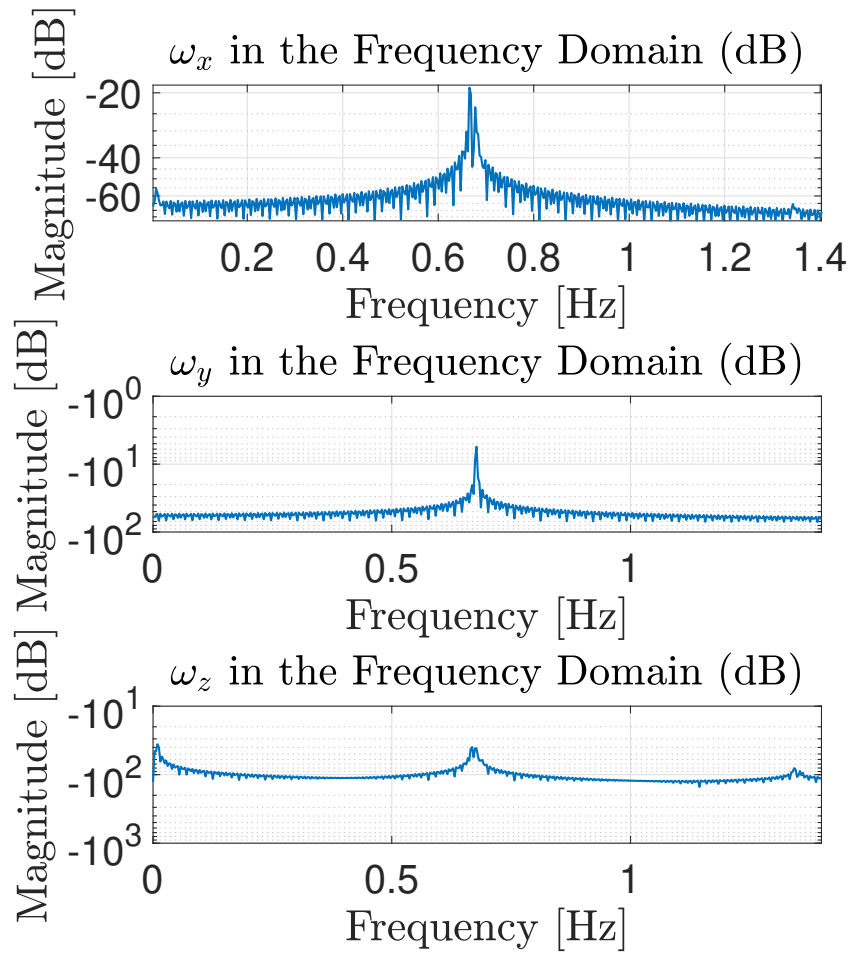
Figure 3.4: Angular rates in the frequency domain.

filter is applied to all state parameters and given that both the attitude and angular rates operate at the same frequency, applying the same filters to both quaternions and angular rates will result in the delay being cancelled.
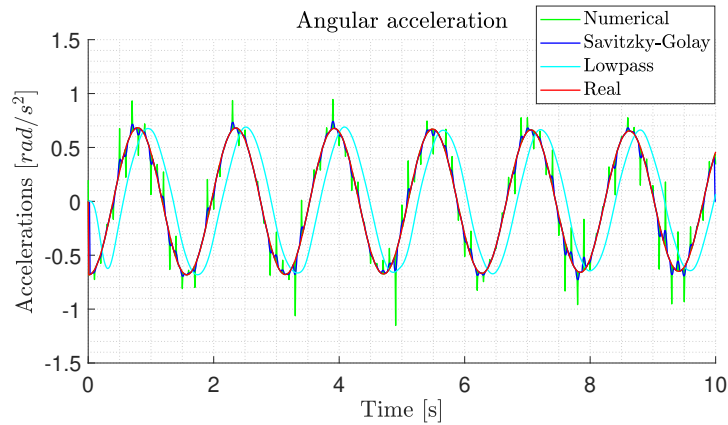
(a) Angular accelerations on x-axis, wherein the "Real" is the nominal trajectory and "Numerical" is attained from numerical differentiation of the noisy angular velocities. "Lowpass" and "Savitzky-Golay" refer to the filters which take as input "Numerical".
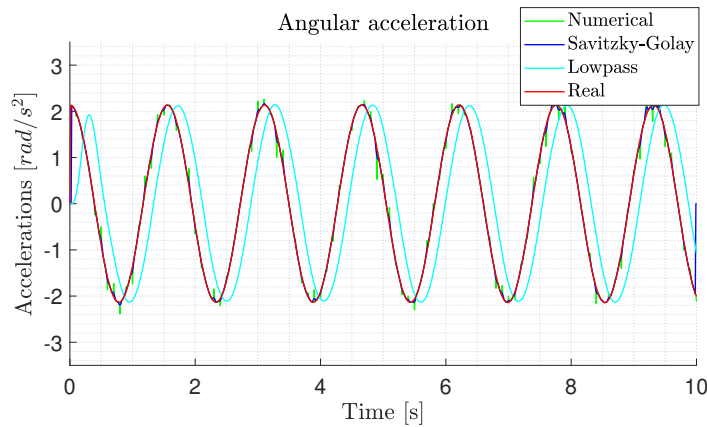


(b) Angular accelerations on y-axis.
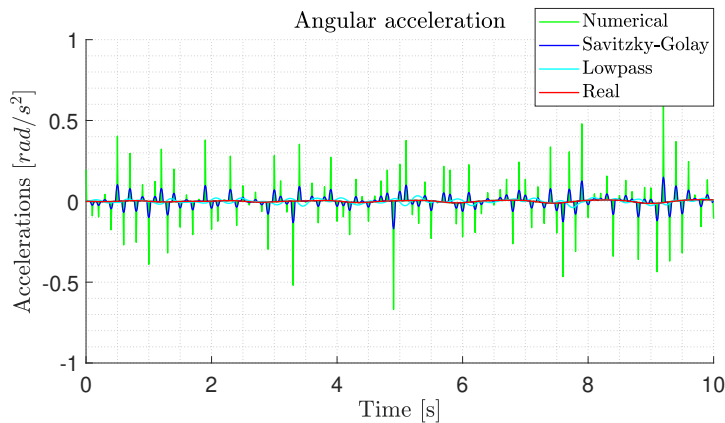


(c) Angular accelerations on z-axis.

Figure 3.5: Filtering techniques comparison.

## 3.3.    Active control techniques

In this category of methods, the MMUs accomplish the offset compensation through the actuation of an active control technique. The rationale underlying the methodology is to devise a control law that steers the MMUs towards a configuration in which the offset is neutralized. This achievement can be realized through direct estimation of the offset, or imposing a condition (e.g., null angular momentum), wherein intrinsically the offset must have settled to zero.

### 3.3.1.    PID control

The simplest control strategy is PID, requiring only the Euler Angles of the platform and its angular rates. It is worth noticing that in the small angles approximation the roll and pitch angles may be approximated by the components of the gravity vector in the body frame, therefore, as reported in [13], the error becomes

$$\mathbf{e} = [\phi, \theta]^T = [-\frac{g_y}{\mathbf{g}}, \frac{g_x}{\mathbf{g}}]^T. \tag{3.24}$$

The different signs are assigned considering that a positive roll angle results in negative y-component of the gravity vector in the body frame and, conversely, a positive pitch implies a positive x-component of the gravity vector. The non-full controllability is managed with null error around yaw axis, as follows:

$$\mathbf{e}_{PID} = [-\frac{g_y}{\mathbf{g}}, \frac{g_x}{\mathbf{g}}, 0]^T. \tag{3.25}$$

The resulting control torque is therefore

$$\boldsymbol{\tau} = -K_P \mathbf{e}_{PID} - K_I \int_{t_0}^{t} \mathbf{e}_{PID} \, dt - K_D \boldsymbol{\omega}, \tag{3.26}$$

where the control gains $K_P, K_I, K_D$ are tuned by trial and error.

### 3.3.2.    Non-linear control based on gravity vector

Despite the theoretical efficacy of PID control, there is a significant susceptibility to instability during practical implementation, particularly when nonlinear effects surpass the control torque. Consequently, a nonlinear control strategy has also been executed, as advocated in [13]. This approach leverages the platform's attitude information to produce,

under the hypothesis of diagonal inertia tensor, a torque that orients the yaw axis towards the local vertical direction. In line with this approach, a candidate Lyapunov function is introduced as follows:

$$V = \frac{1}{2}\boldsymbol{\omega}^T J\boldsymbol{\omega} + \frac{1}{2}K_P(\hat{\mathbf{g}} - \hat{\mathbf{z}})^T(\hat{\mathbf{g}} - \hat{\mathbf{z}}) \tag{3.27}$$

wherein $\hat{\mathbf{z}}$ is the yaw-axis direction and $\hat{\mathbf{g}}$ is the local vertical direction, that is the opposite of the gravity versor,

$$\hat{\mathbf{g}} = -\frac{\mathbf{g}}{||\mathbf{g}||}. \tag{3.28}$$

Therefore, the time-derivative of the Lyapunov function becomes:

$$
\begin{aligned}
\dot{V} &= \boldsymbol{\omega}^T J\dot{\boldsymbol{\omega}} - K_P\hat{\mathbf{z}}^T\dot{\hat{\mathbf{g}}} \\
&= \boldsymbol{\omega}^T(\boldsymbol{\tau}_u - \boldsymbol{\omega} \times J\boldsymbol{\omega}) + K_P\hat{\mathbf{z}}^T\boldsymbol{\omega} \times \hat{\mathbf{g}} \\
&= \boldsymbol{\omega}^T\boldsymbol{\tau}_u + K_P\boldsymbol{\omega}^T\hat{\mathbf{g}} \times \hat{\mathbf{z}} \\
&= \boldsymbol{\omega}^T(\boldsymbol{\tau}_u + K_P\hat{\mathbf{g}} \times \hat{\mathbf{z}}),
\end{aligned}
\tag{3.29}
$$

where the local vertical versor is seamlessly propagated as a constant vector in a rotating frame $\dot{\hat{\mathbf{g}}} = -\boldsymbol{\omega} \times \hat{\mathbf{g}}$. The control torque shall be generated such as the time-derivative of the Lyapunov function is semi-definite negative. Hence,

$$\boldsymbol{\tau}_u = -K_P\hat{\mathbf{g}} \times \hat{\mathbf{z}} - K_D\boldsymbol{\omega}_\perp \tag{3.30}$$

satisfies the asymptotic stability criterion. Indeed, given that $\boldsymbol{\omega}_\perp$ is the component of the angular velocity orthogonal to the gravity vector, the time derivative of the candidate Lyapunov function becomes

$$
\begin{aligned}
\dot{V} &= -K_D\boldsymbol{\omega}^T\boldsymbol{\omega}_\perp \\
&= -K_D\boldsymbol{\omega}_\perp^T\boldsymbol{\omega}_\perp \\
&= -K_D||\boldsymbol{\omega}_\perp||^2.
\end{aligned}
\tag{3.31}
$$

The control strategy is fundamentally designed to steer the system towards a state of pure rotation around the vector $\hat{\mathbf{g}}$. This method remains robust against aerodynamic drag disturbances, which is a primary reason for devising an active control mechanism for planar balancing. Aerodynamic momentum typically introduces bias in estimation processes, such as filtering techniques or batch estimation methods. Instead, within the active control approach, the system is simply guided toward an alternative equilibrium state. To illustrate this, if the drag's dissipative term $-\varepsilon\Lambda J\boldsymbol{\omega}$ is incorporated into the mathematical derivations found in Equation (3.29), it is possible to show that the Lyapunov function's

derivative maintains a negative semi-definite characteristic,

$$\dot{V} = \boldsymbol{\omega}^T(\boldsymbol{\tau}_u + K_P \hat{\mathbf{g}} \times \hat{\mathbf{z}} - \varepsilon \Lambda J \boldsymbol{\omega}). \tag{3.32}$$

Therefore, without any modification to the control torque,

$$\begin{aligned} \dot{V} &= -K_D ||\boldsymbol{\omega}_\perp||^2 - \varepsilon \Lambda \boldsymbol{\omega}^T J \boldsymbol{\omega} \\ &= -K_D ||\boldsymbol{\omega}_\perp||^2 - \varepsilon \Lambda E_{kin} \end{aligned} \tag{3.33}$$

where $E_{kin}$ represents the kinetic energy of the platform, which is a quadratic form. Hence, the platform is driven to a condition wherein no rotation is performed, instead of pure rotation around $\hat{\mathbf{g}}$. Nonetheless, the control theoretically maintains its capabilities.

The mathematical manipulations depend uniquely on the model assumed for the aerodynamic drag. Indeed, no assumption are performed on the shape of the inertia matrix of the platform, being it positive semi-definite by definition.

The formulated control torque, rather than being directly applied, incorporates an integral element into the control schema, bolstering both stability and control effectiveness. Such an approach effectively dampens steady-state discrepancies, fortifying the system's resilience against perturbations. As expounded in [13, 68], this strategy proves especially beneficial when the disruptive torque is confined within certain bounds, analogous to the scenario of a 3D pendulum motion. Consequently,

$$\boldsymbol{\tau}_u = -K_P \hat{\mathbf{g}} \times \hat{\mathbf{z}} - K_D \boldsymbol{\omega}_\perp - K_I \int_{t_0}^{t} \hat{\mathbf{g}} \times \hat{\mathbf{z}} \tag{3.34}$$

is the definitive control law. The designed strategy is underpinned by the principle that when the integral term converges to a bounded value, it is a sufficient indicator that the cross product of the vectors $\hat{\mathbf{g}}$ and $\hat{\mathbf{z}}$ has stabilized to zero. This is a critical aspect of the control law, as it implies that the platform's z-axis has aligned with the local gravitational vector, signifying a successful attainment of the desired orientation.

### 3.3.3.   Non-linear control based on angular momentum

Although certain control laws are not currently intended for deployment on STASIS due to hardware constraints or insufficient data, they have been incorporated into the digital twin as a preliminary stage for prospective advancements. For example, when RWs will be installed on STASIS, a preliminary LSQ could be executed to assess platform parameters. Following, control strategies based on a comprehensive knowledge of the inertia matrix or

the offset could be examined, potentially leading to a refined compensation mechanism. Hence, an adapted version of the control law detailed in [30] has been implemented. Under the assumption that the torque generated by the offset between the CM and the CR is the sole acting on the simulator, the EOM can be expressed as follows:

$$\dot{\mathbf{H}} + [\boldsymbol{\omega}\times]\mathbf{H} = m\mathbf{r} \times \mathbf{g}. \tag{3.35}$$

Therefore, a control able to minimize the angular momentum, concurrently shall minimize the offset as well. To compute the angular momentum at each step, the knowledge of the inertia parameters shall be available. Hence, conjecturing the knowledge of platform inertia, the control can be implemented updating at each step $J(t)$. Thus, considering the MMUs as lumped masses, the expression of the inertia matrix is the following:

$$J(t) = J_p + \sum_{i=1}^{8}(-m_i[\mathbf{R}_i(t)\times][\mathbf{R}_i(t)\times]), \tag{3.36}$$

where $m_i$ and $\mathbf{R}_i(t)$ are respectively mass and position of each MMU with respect to the platform centre of rotation. Hence, decomposing the offset respectively in initial offset and offset shift due to the control,

$$\mathbf{r} = \mathbf{r}_0 + \delta\mathbf{r} \tag{3.37}$$

the following Lyapunov function is proposed:

$$V(\mathbf{H}, \delta\mathbf{r}) = \frac{1}{2}\mathbf{H}^T\mathbf{H} + \frac{1}{2}(\mathbf{r}_0 + \delta\mathbf{r})^T\boldsymbol{\Gamma}^{-1}(\mathbf{r}_0 + \delta\mathbf{r}) \tag{3.38}$$

where $\boldsymbol{\Gamma}$ is a symmetric positive definite matrix. Ergo, the time-derivative of the candidate Lyapunov can be seamlessly computed,

$$\begin{aligned}
\dot{V} &= \mathbf{H}^T\dot{\mathbf{H}} + (\mathbf{r}_0 + \delta\mathbf{r})^T\boldsymbol{\Gamma}^{-1}\dot{\delta\mathbf{r}} \\
&= \mathbf{H}^T(-[\boldsymbol{\omega}\times]\mathbf{H} - m[\mathbf{g}\times](\mathbf{r}_0 + \delta\mathbf{r})) + (\mathbf{r}_0 + \delta\mathbf{r})^T\boldsymbol{\Gamma}^{-1}\dot{\delta\mathbf{r}},
\end{aligned} \tag{3.39}$$

where $\dot{\mathbf{H}}$ is substituted through Equation (3.35). Hence, the proposing adaptation law

$$\dot{\delta\mathbf{r}} = m\boldsymbol{\Gamma}[\mathbf{g}\times]^T\mathbf{H} \tag{3.40}$$

the control is capable to attain a semi-definite negative derivative of the Lyapunov function. Indeed, substituting Equation (3.40) in Equation (3.39), it is possible to demon-

strate, upon cancellation of the last terms, that

$$\dot{V} = -\mathbf{H}^T[\boldsymbol{\omega}\times]\mathbf{H}, \tag{3.41}$$

thus satisfying the asymptotic stability criterion. It is important to recognize that the theoretical convergence of the system is not impacted by the aerodynamic effect. In fact, when incorporating the dissipative term $-\varepsilon\Lambda J\boldsymbol{\omega}$ into the EOM, and then substituting this into Equation (3.39),

$$\dot{V} = \mathbf{H}^T(-[\boldsymbol{\omega}\times]\mathbf{H} - m[\mathbf{g}\times](\mathbf{r}_0 + \delta\mathbf{r}) - \varepsilon\Lambda J\boldsymbol{\omega}) + (\mathbf{r}_0 + \delta\mathbf{r})^T\boldsymbol{\Gamma}^{-1}\dot{\delta\mathbf{r}}, \tag{3.42}$$

which still results in a negative semi-definite function. Indeed, considering the adaptation law proposed in Equation (3.40), the time-derivative of the Lyapunov function becomes:

$$\dot{V} = -\mathbf{H}^T([\boldsymbol{\omega}\times] + \varepsilon\Lambda\mathbf{I}_{3\times3})\mathbf{H}, \tag{3.43}$$

proving the environmental robustness of the control. As in the previous case, an integral term is incorporated to the proposed control law to accelerate convergence and enhance the stability of the control.

$$\dot{\delta\mathbf{r}} = m\boldsymbol{\Gamma}[\mathbf{g}\times]^T\mathbf{H} + K_I\int_{t_0}^t m\boldsymbol{\Gamma}[\mathbf{g}\times]^T\mathbf{H}. \tag{3.44}$$

Equation (3.44) can be directly integrated at each step to feed the MMUs with a position information, without any intermediate step passing through a control torque generation. Hence, the control loop is capable to output directly the MMUs positions at each step without further conversions, enhancing the computational efficiency.

### 3.3.4. Non-linear control based on offset estimation

An alternative to the aforementioned control strategy involves offset compensation based on its prior estimation. The underlying principle remains the conservation of angular momentum. However, the approach here entails directly producing a torque that neutralizes the torque arising from the CR-CM offset. Indeed, as evident from EOM,

$$J\dot{\boldsymbol{\omega}} + \boldsymbol{\omega} \times J\boldsymbol{\omega} = m\mathbf{r} \times \mathbf{g} + \boldsymbol{\tau}_u \tag{3.45}$$

if $\boldsymbol{\tau}_u$ is shaped to cancel the $m\mathbf{r}\times\mathbf{g}$ term, the angular momentum is necessarily conserved. To actuate a direct compensation, the real offset value would be required. Instead, con-

sidering the impossibility of an exact estimation of the offset, let $\tilde{\mathbf{r}}(t)$ be the error between the real offset $\mathbf{r}$ and the real time estimate $\hat{\mathbf{r}}(t)$.

$$\tilde{\mathbf{r}}(t) = \mathbf{r} - \hat{\mathbf{r}}(t). \tag{3.46}$$

In order to expedite the manipulation of equations, the matrix $\mathbf{\Psi}$ is defined as

$$\mathbf{\Psi} = -m[\mathbf{g}\times], \tag{3.47}$$

lightening the mathematical notation in the following demonstrations. Hence, the candidate Lyapunov function is proposed to embed both the kinetic energy and the error in Equation (3.46). Moreover, a kinematic component is incorporated by including the quaternion vector $\mathbf{q}$,

$$V(\mathbf{q}, \boldsymbol{\omega}, \tilde{\mathbf{r}}) = \frac{1}{2}\boldsymbol{\omega}^T J \boldsymbol{\omega} + \frac{1}{2}\tilde{\mathbf{r}}^T \tilde{\mathbf{r}} + \frac{1}{2}\mathbf{q}^T \mathbf{q}. \tag{3.48}$$

Thus, substituting Equation (3.45) and conjecturing constant inertia matrix, the time-derivative of the Lyapunov function is:

$$
\begin{aligned}
\dot{V} &= \boldsymbol{\omega}^T J \dot{\boldsymbol{\omega}} + \tilde{\mathbf{r}}^T \dot{\tilde{\mathbf{r}}} + \mathbf{q}^T \dot{\mathbf{q}} \\
&= \boldsymbol{\omega}^T (-\boldsymbol{\omega} \times J\boldsymbol{\omega} + \mathbf{\Psi}\mathbf{r} + \boldsymbol{\tau}_u) + \tilde{\mathbf{r}}^T \dot{\tilde{\mathbf{r}}} \\
&= \boldsymbol{\omega}^T \mathbf{\Psi}\mathbf{r} + \boldsymbol{\omega}^T \boldsymbol{\tau}_u + \tilde{\mathbf{r}}^T \dot{\tilde{\mathbf{r}}}
\end{aligned}
\tag{3.49}
$$

where the kinematic contribution of the quaternion vector is canceled as $\mathbf{q}^T \dot{\mathbf{q}} = 0$ [50]. Subsequently, Equation (3.49) is rearranged using the definition of the estimation error in Equation (3.46). As a result, the time-derivative of the Lyapunov function can be further developed as:

$$\dot{V} = \boldsymbol{\omega}^T \mathbf{\Psi}\hat{\mathbf{r}} + \boldsymbol{\omega}^T \boldsymbol{\tau}_u + \tilde{\mathbf{r}}^T (\dot{\hat{\mathbf{r}}} + \dot{\tilde{\mathbf{r}}}) \tag{3.50}$$

where $\dot{\tilde{\mathbf{r}}} = -\mathbf{\Psi}^T \boldsymbol{\omega}$ [43, 50]. Hence, accurate selection of the adaptive law of the offset estimation and proper shaping of the control torque result in an asymptotic stability of the control. Indeed, letting the adaptive law for the estimated parameter be

$$\dot{\hat{\mathbf{r}}} = \mathbf{\Psi}^T \boldsymbol{\omega} \tag{3.51}$$

and the control torque,

$$\boldsymbol{\tau}_u = -\mathbf{\Psi}\hat{\mathbf{r}} - K_P \boldsymbol{\omega}_\perp \tag{3.52}$$

the time-derivative of the candidate Lyapunov function becomes:

$$
\begin{aligned}
\dot{V} &= \boldsymbol{\omega}^T \boldsymbol{\Psi}\hat{\mathbf{r}} - \boldsymbol{\omega}^T \boldsymbol{\Psi}\hat{\mathbf{r}} - K_P \boldsymbol{\omega}^T \boldsymbol{\omega}_\perp \\
&= -K_P \boldsymbol{\omega}^T \boldsymbol{\omega}_\perp \\
&= -K_P ||\boldsymbol{\omega}_\perp||^2,
\end{aligned}
\tag{3.53}
$$

proving the stability of the designed control, being negative semi-definite. Again, the control is robust with respect to unmodeled aerodynamic effect. To prove this inference, it is sufficient to incorporate the dissipative aerodynamic momentum $-\varepsilon\Lambda J\boldsymbol{\omega}$ in Equation (3.53) and demonstrate that the resulting function remains negative semi-definite.

$$
\dot{V} = \boldsymbol{\omega}^T(\boldsymbol{\Psi}\hat{\mathbf{r}} + \boldsymbol{\tau}_u - \varepsilon\Lambda J\boldsymbol{\omega}) + \tilde{\mathbf{r}}^T\dot{\tilde{\mathbf{r}}}.
\tag{3.54}
$$

Therefore, executing the same passages reported in Equation (3.53):

$$
\begin{aligned}
\dot{V} &= -K_D||\boldsymbol{\omega}_\perp||^2 - \varepsilon\Lambda\boldsymbol{\omega}^T J\boldsymbol{\omega} \\
&= -K_D||\boldsymbol{\omega}_\perp||^2 - \varepsilon\Lambda E_{kin}.
\end{aligned}
\tag{3.55}
$$

Hence, upon incorporation of drag, the system is driven to a different equilibrium point, which is the null rotation. As in the previous cases, an integral contribution is embodied in the control law, to improve the convergence properties of the offset compensation. As such, generating a torque given by

$$
\boldsymbol{\tau}_u = -\boldsymbol{\Psi}\hat{\mathbf{r}} - K_D\boldsymbol{\omega}_\perp - K_I \int_{t_0}^{t} \boldsymbol{\Psi}\hat{\mathbf{r}} \, dt
\tag{3.56}
$$

is theoretically functional to the purpose of compensating the platform.

### 3.3.5. Control torque generation

Upon implementation of the control law, the torque $\boldsymbol{\tau}_u$ shall be physically generated by using the MMUs system. Thus, a mapping function from control torque to mass shift shall be formulated. To fulfill this objective, the torque generated by the individual *i-th* mass is inspected,

$$
\boldsymbol{\tau}_u = m_i(-\mathbf{g} \times \mathbf{R}_i).
\tag{3.57}
$$

A proper mapping function shall consider consistency in the transformation. Specifically, it shall be mentioned that the MMUs cannot generate a torque in the tridimensional space. Indeed, as discussed in Section 2.4, the problem is under-actuated. This information is

encapsulated in the cross product in Equation (3.57), constraining the torque on a plane orthogonal to the gravity vector. Consequently, given the output of the control $\boldsymbol{\tau}_u$, to generate consistent command to the MMUs it is necessary to observe that the control output is always orthogonal to the gravity vector. Coherently, all the control torques designed in Section 3.3, are orthogonal to $\mathbf{g}$, which is pivotal for a consistent actuation. Upon this verification, the mapping function can be consistently implemented. However, it is important to acknowledge that Equation (3.57) cannot be inverted to directly retrieve the command, because matrix $[-\mathbf{g}\times]$ is always singular. Nonetheless, since all the control torques are consistently generated, they are guaranteed to be in the range of $[-\mathbf{g}\times]$. Hence, it is possible to prove that there always exist a mass position $\mathbf{R}_i$ capable to satisfy Equation (3.57). Indeed, a solution is given by:

$$\mathbf{R}_i = \frac{\mathbf{g} \times \boldsymbol{\tau}_u}{||\mathbf{g}||^2 m_i}. \tag{3.58}$$

To verify this, it is sufficient to substitute Equation (3.58) in Equation (3.57).

$$
\begin{aligned}
\boldsymbol{\tau}_u =& -m_i\left(\frac{\mathbf{g} \times \mathbf{g} \times \boldsymbol{\tau}_u}{||\mathbf{g}||^2 m_i}\right) \\
=& -m_i\left(\frac{(\mathbf{g} \cdot \boldsymbol{\tau}_u)\mathbf{g} - (\mathbf{g} \cdot \mathbf{g})\boldsymbol{\tau}_u}{||\mathbf{g}||^2 m_i}\right) \\
=& \boldsymbol{\tau}_u.
\end{aligned}
\tag{3.59}
$$

Summarily, if the control law is designed with consistency with respect to the under-actuation problem, it is always possible to generate a consistent command to the MMUs.

## 3.4.   Kalman filtering methods

In Section 3.2, the LSQ method is introduced as a mean to estimate the unbalance vector, operating under an implicit linearity assumption, which is evident in Equation (3.18). This stringent assumption could potentially lead to sub-optimal outcomes due to the inherent non-linear dynamics of the 3D pendulum. Consequently, demonstrating a non-linear technique for estimation becomes notably pertinent. To address this, EKF and UKF are incorporated into this work, aimed at developing a dependable estimator for experimental application.

### 3.4.1. Extended Kalman filtering

Considering the myriad of challenges presented by non-linear systems, the Extended Kalman Filter (EKF) was devised as an enhancement to the traditional Kalman filtering methodology. Even though it inherently operates by linearizing about the current state, the EKF is specifically tailored to address non-linear applications effectively.

To settle the basis for the filter's implementation, it is essential to first represent the model, embedding both state and measurements, in the state-space form:

$$
\begin{aligned}
\mathbf{x}_k &= \mathbf{f}_{k-1}(\boldsymbol{\omega}_{k-1}, \mathbf{r}_{k-1}, \mathbf{w}_{k-1}), \\
\mathbf{y}_k &= \mathbf{h}_k(\boldsymbol{\omega}_k, \mathbf{r}_k, \mathbf{v}_k),
\end{aligned}
\tag{3.60}
$$

wherein the state vector contains angular velocities and offset $\mathbf{x} = [\boldsymbol{\omega}\ \mathbf{r}]^T$ and measurements are the measured angular velocities, obtained integrating white noise to the real values. The vectorial function $\mathbf{f}(\boldsymbol{\omega}, \mathbf{r}, \mathbf{w})$ is retrieved through a linearization of the EOM around the current estimate of the state:

$$
\begin{cases}
f_1^{k-1} = \omega_{x,k} = \omega_x + \dot{\omega}_x \Delta T = \omega_x + [\omega_z \omega_y(\frac{J_y - J_z}{J_x}) + \frac{1}{J_x}(mg_z r_y - mg_y r_z)]\Delta T \\
f_2^{k-1} = \omega_{y,k} = \omega_y + \dot{\omega}_y \Delta T = \omega_y + [\omega_x \omega_z(\frac{J_z - J_x}{J_y}) + \frac{1}{J_y}(mg_x r_z - mg_z r_x)]\Delta T \\
f_3^{k-1} = \omega_{z,k} = \omega_z + \dot{\omega}_z \Delta T = \omega_z + [\omega_x \omega_y(\frac{J_x - J_y}{J_z}) + \frac{1}{J_z}(mg_y r_x - mg_x r_y)]\Delta T \\
f_4^{k-1} = r_x^k = r_x^{k-1} \\
f_5^{k-1} = r_y^k = r_y^{k-1} \\
f_6^{k-1} = r_z^k = r_z^{k-1}
\end{cases}
\tag{3.61}
$$

It shall be noticed that to adeptly update the covariance matrix at each step, the EKF necessitates the computation of the Jacobian of $\mathbf{f}(\boldsymbol{\omega}, \mathbf{r}, \mathbf{w})$. This can be straightforwardly derived from Equation (3.61). Specifically, deriving the function of the dynamics with respect to the state vector,

$$
\mathbf{F}_{k-1} =
\begin{bmatrix}
\frac{\partial f_1^{k-1}}{\partial x_1^{k-1}} & \cdots & \frac{\partial f_1^{k-1}}{\partial x_6^{k-1}} \\
\vdots & & \vdots \\
\frac{\partial f_6^{k-1}}{\partial x_1^{k-1}} & \cdots & \frac{\partial f_6^{k-1}}{\partial x_6^{k-1}}
\end{bmatrix}
\tag{3.62}
$$

the Jacobian can be fully analytically compiled. To lighten the computation of the 36 derivatives involved, let notice that some rows are straightforward to implement. Indeed,

given that the offset is independent of the angular rate, the Jacobian matrix $\mathbf{F}_{k-1}$ assumes the following structure:

$$\mathbf{F}_{k-1} = \begin{bmatrix} [\frac{\partial f_{1:3}^{k-1}}{\partial \mathbf{x}_{k-1}}]_{3\times6} \\ 0_{3\times3} \; I_{3\times3} \end{bmatrix}. \tag{3.63}$$

Performed the linearization around the current state, the filter can be seamlessly implemented. Indeed, with the state and the Jacobian in hand, the EKF can be executed in the subsequent loop:

1. Prediction step: The state is predicted through a dynamics propagation of the previous state estimate $\hat{\mathbf{x}}_{k-1}^+$. Covariance matrix $\mathbf{P}_{k-1}^+$ is propagated through the computed Jacobian. In this step also the measurements $\mathbf{y}_k$ are predicted through the measurement equation.

$$\begin{aligned} \hat{\mathbf{x}}_k^- &= \mathbf{f}_{k-1}(\hat{\mathbf{x}}_{k-1}^+, \mathbf{w}_{k-1}), \\ \mathbf{P}_k^- &= \mathbf{F}_{k-1}\mathbf{P}_{k-1}^+\mathbf{F}_{k-1}^T, \\ \hat{\mathbf{y}}_k^- &= \mathbf{h}_k(\hat{\mathbf{x}}_k^-, \mathbf{v}_k). \end{aligned} \tag{3.64}$$

2. Measuremets Update: The state estimation ad covariance matrix are updated according the acquired measurements $\mathbf{y}_k$.

$$\begin{aligned} \mathbf{K}_k &= \mathbf{P}_k^- \mathscr{H}_k^T (\mathscr{H}_k \mathbf{P}_k^- \mathscr{H}_k^T + \mathbf{R})^{-1}, \\ \hat{\mathbf{x}}_k^+ &= \hat{\mathbf{x}}_k^- + \mathbf{K}_k(\mathbf{y}_k - \hat{\mathbf{y}}_k^-), \\ \mathbf{P}_k^+ &= (\mathbf{I} - \mathbf{K}_k\mathscr{H}_k)\mathbf{P}_k^-. \end{aligned} \tag{3.65}$$

Where $\mathscr{H}_k$ is the Jacobian of the measurements.

$$\mathscr{H}_k = \frac{\partial \mathbf{h}_k(\hat{\mathbf{x}}_k^-, \mathbf{v}_k)}{\partial \mathbf{x}}\Big|_{\mathbf{x}_k^-}. \tag{3.66}$$

However, given that the measurements are integrated into the state, $\mathscr{H}_k$ is simply expressed as:

$$\mathscr{H}_k = [\mathbf{I}_{3\times3} \; \mathbf{0}_{3\times3}]. \tag{3.67}$$

Therefore, the filter at each step updates its estimates through a weighted difference of the acquired measurements and their analytical prediction, retrieved through dynamics propagation of the previous state estimation.

### 3.4.2. Augmented Extended Kalman filter

Interestingly, within the context of the EKF, linearity might not be its most stringent assumption. Indeed, the procedure presupposes precise knowledge of the platform's inertia parameters, a condition that is unattainable in real-world experiments. This could notably limit the accuracy of the state-observer. Hence, to release this stringent assumption, let $J$ matrix be part of the estimation process. Operationally, this is augmenting the state including the inertia parameters, as follows:

$$\mathbf{x} = [\omega_x\ \omega_y\ \omega_z\ J_x\ J_y\ J_z\ J_{xy}\ J_{xz}\ J_{yz}\ r_x\ r_y\ r_z]^T. \tag{3.68}$$

However, as detailed in Section 2.3, this necessitates the use of Reaction Wheels (RWs), or any equivalent control, to reestablish the observability of the state. In this study, RWs are employed in alignment with the anticipated future advancements of STASIS. Incorporating RWs introduces a notable challenge, primarily due to the intricate nature of analytically computing the Jacobian $\mathbf{F}_{k-1}$. As a solution, this study employs a numerical method to determine the Jacobian: the Complex Step Differentiator (CSD) [24]. The CSD stands out as a highly precise algorithmic differentiation tool, exhibiting a second-order convergence relative to the step size. The key characteristics of the CSD are detailed in [69] and succinctly encapsulated in Figure 3.6.
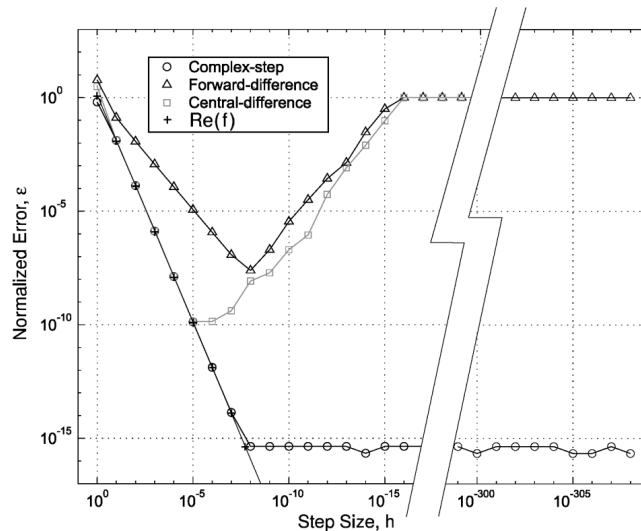


Figure 3.6: CSD performances [69].

Distinctively, CSD, unlike finite difference methods, is not subjected to cancellation error, since no subtracting operations are involved. This feature enables to achieve quadratic

convergence also for very small step sizes, reaching practically the Machine $\varepsilon$ degree of accuracy. Specifically, the principle behind the CSD can be summarized as:

- Firstly, the function is added of an imaginary step of size $h$.

- Then, the imaginary part of the computed function is isolated.

- Finally, the result is divided by $h$.

For instance, at the instant k-1 the derivative of the angular acceleration with respect to the angular rate can be seamlessly computed:

$$
\left[\frac{\partial \dot{\boldsymbol{\omega}}_{k-1}}{\partial \boldsymbol{\omega}_{k-1}}\right]_{CSD} = \left[ \frac{\Im\left[\dot{\boldsymbol{\omega}}_{k-1}\begin{pmatrix} \omega_{x,k-1} + h \cdot i \\ \omega_{y,k-1} \\ \omega_{z,k-1} \end{pmatrix}\right]}{h} \quad \cdots \right].
\tag{3.69}
$$

Hence, the (1:3,1:3) sub-matrix of the Jacobian $\mathbf{F}_{k-1}$ becomes:

$$
\left[\frac{\partial f^{k-1}_{1:3}}{\partial \boldsymbol{\omega}_{k-1}}\right]_{3\times3} = \mathbf{I}_{3\times3} + \Delta T \left[\frac{\partial \dot{\boldsymbol{\omega}}_{k-1}}{\partial \boldsymbol{\omega}_{k-1}}\right]_{CSD}.
\tag{3.70}
$$

By executing a similar procedure for each component of the state vector, the Jacobian is comprehensively constructed, paving the way for the EKF implementation. Beyond the problem's dimension, which is doubled, the architectural framework of the augmented EKF implementation mirrors the simple EKF case. However, in this case, the prediction step is slightly modified to include the dynamic effect of RWs solicitation. As a consequence, considering $\mathbf{u}$ the control torque generated by the RWs apparatus, the state-space form shall be implemented as

$$
\begin{aligned}
\mathbf{x}_k &= \mathbf{f}_{k-1}(\boldsymbol{\omega}_{k-1}, \mathbf{r}_{k-1}, \mathbf{u}_{k-1}, \mathbf{w}_{k-1}), \\
\mathbf{y}_k &= \mathbf{h}_k(\boldsymbol{\omega}_k, \mathbf{r}_k, \mathbf{v}_k).
\end{aligned}
\tag{3.71}
$$

### 3.4.3.  Unscented Kalman filtering

Given the inherent non-linearity of the problem, a non-linear filtering approach could potentially yield superior results. Consequently, the UKF has been integrated into the study [70–72]. In this filter, there is no endeavour to determine analytically the probability distribution of the problem. Instead, it is shaped through the propagation of optimal points, the $\sigma$ points. Following, the UKF algorithm can thus be structured in the subsequent manner:

1. Generation of $\sigma$ *points*: set $\alpha = 1e - 3$, $\beta = 2$, $\kappa = 0$, the $\sigma$ *points* are computed from the previous estimated state.

$$
\begin{aligned}
\boldsymbol{\chi}_{0,k-1} &= \hat{\mathbf{x}}_{k-1}^{+}, \\
\boldsymbol{\chi}_{i,k-1} &= \hat{\mathbf{x}}_{k-1}^{+} + \Delta \mathbf{x}_i \quad for \ i = 1...2n, \\
\Delta \mathbf{x}_i &= \sqrt{c\mathbf{P}_{k-1}^{+}} \quad for \ i = 1...n, \\
\Delta \mathbf{x}_i &= \sqrt{c\mathbf{P}_{k-1}^{+}} \quad for \ i = n+1...2n, \\
c &= \alpha^2(n + \kappa).
\end{aligned}
\tag{3.72}
$$

2. Prediction step: $\sigma$ *points* are propagated through the $\mathbf{f}$ function and the respective measurement are computed.

$$
\begin{aligned}
\boldsymbol{\chi}_{i,k} &= \mathbf{f}(\boldsymbol{\chi}_{i,k-1}, \mathbf{w}_{k-1}), \\
\boldsymbol{\gamma}_{i,k} &= \mathbf{h}(\boldsymbol{\chi}_{i,k}, \mathbf{v}_k).
\end{aligned}
\tag{3.73}
$$

Hence, the Unscented Transform weights are introduced,

$$
\begin{aligned}
W_0^{(m)} &= 1 - \frac{n}{\alpha^2(n+\kappa)}, \\
W_i^{(m)} &= \frac{1}{2\alpha^2(n+\kappa)}, \quad for \quad i = 1, \dots, 2n, \\
W_0^{(c)} &= 2 - \alpha^2 + \beta - \frac{n}{\alpha^2(n+\kappa)}, \\
W_i^{(c)} &= \frac{1}{2\alpha^2(n+\kappa)}, \quad for \quad i = 1, \dots, 2n.
\end{aligned}
\tag{3.74}
$$

paving the way for computing the estimated means and covariances

$$
\begin{aligned}
\hat{\mathbf{x}}_k^- &= \sum_{i=0}^{2n} W_i^{(m)} \boldsymbol{\chi}_i, \\
\hat{\mathbf{y}}_k^- &= \sum_{i=0}^{2n} W_i^{(m)} \boldsymbol{\gamma}_i, \\
\mathbf{P}_k^- &= \sum_{i=0}^{2n} W_i^{(c)} (\boldsymbol{\chi}_i - \hat{\mathbf{x}}_k^-)(\boldsymbol{\chi}_i - \hat{\mathbf{x}}_k^-)^T, \\
\mathbf{P}_{ee,k} &= \sum_{i=0}^{2n} W_i^{(c)} (\boldsymbol{\gamma}_i - \hat{\mathbf{y}}_k^-)(\boldsymbol{\gamma}_i - \hat{\mathbf{y}}_k^-)^T + \mathbf{R}_k, \\
\mathbf{P}_{xy,k} &= \sum_{i=0}^{2n} W_i^{(c)} (\boldsymbol{\chi}_i - \hat{\mathbf{x}}_k^-)(\boldsymbol{\gamma}_i - \hat{\mathbf{y}}_k^-)^T.
\end{aligned}
\tag{3.75}
$$

3. Measuremets Update: The state estimation ad covariance matrix are updated according the acquired measurements $\mathbf{y}_k$.

$$
\begin{aligned}
\mathbf{K}_k &= \mathbf{P}_{xy,k}^- \mathbf{P}_{ee,k}^{-1}, \\
\hat{\mathbf{x}}_k^+ &= \hat{\mathbf{x}}_k^- + \mathbf{K}_k(\mathbf{y}_k - \hat{\mathbf{y}}_k^-), \\
\mathbf{P}_k^+ &= \mathbf{P}_k^- - \mathbf{K}_k \mathbf{P}_{ee,k} \mathbf{K}_k^T.
\end{aligned}
\tag{3.76}
$$

Within this framework, the primary objective is to engineer a filter robust enough to contend with significant dynamic non-linearities, inherent in the rotational dynamics described by EOM. By forgoing the EKF linearity assumption, the system potentially benefits from an expedited rate of convergence, a crucial factor when aiming to reject the influence of unforeseen disturbances. Indeed, the impact of unmodeled dynamics is more relevant the longer is the time window of observation.

Furthermore, even in instances where initial guesses are far from accurate values, the inherent flexibility of the UKF assures its convergence, guaranteeing robustness to the estimation mechanism. In synthesis, potentially the UKF could stand as a optimal solution in the intricate realm of automatic balancing algorithms.

### 3.4.4.    Augmented Unscented Kalman filtering

It seems incongruous to develop such an intricate and efficient algorithm like the UKF, only to leave it vulnerable to errors stemming from the assumption of precise knowledge of the inertia matrix. To counteract this potential weak point, similar to the EKF approach, the state can be augmented to incorporate inertia parameters into the estimation process. This augmentation bolsters the robustness of the UKF, requiring however the integration of RWs to address the partial observability of the problem.

While this approach effectively doubles the problem's dimension, the structural foundation of the control remains similar. However, it is crucial to note, that the inclusion of RWs does impart an influence on the system dynamics, modifying the state-space form, as reported hereafter,

$$
\begin{aligned}
\mathbf{x}_k &= \mathbf{f}_{k-1}(\boldsymbol{\omega}_{k-1}, \mathbf{r}_{k-1}, \mathbf{u}_{k-1}, \mathbf{w}_{k-1}), \\
\mathbf{y}_k &= \mathbf{h}_k(\boldsymbol{\omega}_k, \mathbf{r}_k, \mathbf{v}_k).
\end{aligned}
\tag{3.77}
$$

## 3.5.    Methodologies remarks

Given the methodological analysis for automatic mass balancing on satellite simulators, it is imperative to elucidate the inherent limitations associated with each approach. This

understanding is crucial for the meticulous planning and organization of an efficient experiment within the STASIS framework. Table 3.2 summarizes the main aspects of each methodology, focusing on the features required to design a consistent experiment.

In structuring an automatic balancing procedure, an astute assessment of the individual strengths and vulnerabilities of each methodology is pivotal. Given the constraints in the frequency of experiment repetition, an approach grounded in diligence and precision becomes indispensable. Ideally, the proficient goal would be to architect a strategy that amalgamates the salient features of individual methodologies while circumventing their inherent limitations. In the framework of STASIS, wherein RWs are omitted from the

Table 3.2: Methodologies remarks, wherein both general and specific features of Kalman filters are explored.

| Methodologies | Remarks |
|---|---|
| Least square | **Model independence**: versatile due to lack of reliance on a specific hypothesis or dynamic model. <br> **Constraint necessity**: requires external mechanisms like RWs or mathematical constraints to determine offset. |
| Active control | **Operational complexity**: necessitates online control mechanisms for balance. <br> **Environmental robustness**: maintains performance even in the presence of external disturbances like drag. <br> **Observational independence**: no prior information or control is needed to compensate. <br> **Bi-axial compensation**: limits compensation to two vector components only. |
| Kalman filters | **Model dependency**: relies on a predefined state-space model for system state estimation. <br> **Inertia sensitivity**: performance are contingent to precise inertia knowledge. <br> **Noise attenuation**: capable of attenuating measurement and process noise through prediction-correction mechanism. |
| EKF | **Need for Jacobian**: inherent simplifications or uncertainties on inertia tensor may lead to skewed results. |
| UKF | **High computational demand**: requires the computation and integration of $\sigma$ points. <br> **Robustness**: more equipped to reject the effects of uncertainties on the filtering process. |

procedure, the essential pillar of an experiment is to find a consistent information on the state vector to feed the following observer with. Therefore, a preliminary experiment chain can be assessed posing the active control as basis to determine the mathematical

constraint $B\mathbf{x} = \mathbf{c}$ required by the LSQ. As the method outputs the inertia parameters, the Kalman filter could be adopted to refine the estimations. Figure 3.7 encapsulates this preliminary strategy.
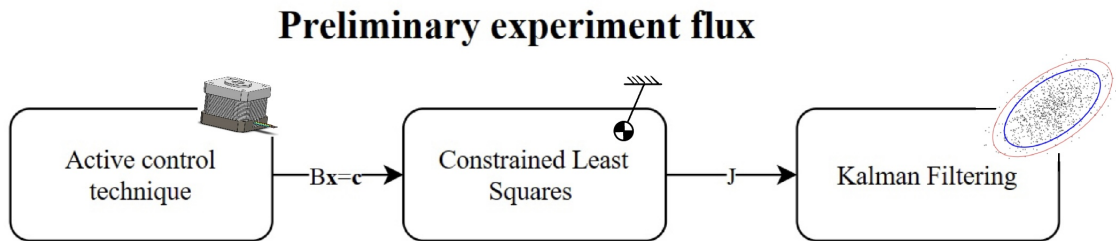


Figure 3.7: Preliminary flux diagram.

# 4 | Simulation results

This chapter delves into the results derived from extensive simulations carried out using MATLAB and Simscape Multibody, all operating within the framework of the digital twin of the platform. As such, it stands as a pivotal foundation for methodical and coherent experiment planning. Notably, any techniques identified that pose pragmatic challenges or fail to show convergence in the presence of disturbances are promptly discarded. This proactive approach ensures that no valuable time is wasted in attempting to actualize them in the real-world experiments. As a result, readers will gain a comprehensive understanding of which methodologies prove promising and which ones are set aside for the scope of this work. In the following sections the solver adopted is automatically set by SimScape, however, it is constrained to be a fixed-step type to keep uniformity with a real world environment, wherein both sensors and actuators work at a set frequency.

## 4.1. Least squares estimation

The simulation are performed setting a time step coherent with the acquisition frequency of the BNO055 sensor of 100 Hz. Table 4.1 presents the performance outcomes of the LSQ method when the external solicitation is sourced from RWs. The RWs performance metrics are configured based on specific, market-available models[1].

Furthermore, simulations were conducted addressing the *scaling problem* by employing the mathematical constraint $B\mathbf{x} = \mathbf{c}$, detailed in Section 3.2. It is pivotal to note that under this condition, where the offset torque is constrained to a plane orthogonal to the local vertical, the z-axis remains unsolicited. As a result, even if the *scaling problem* is solved, $J_z$ can theoretically assume any value, since no torque directly acts on z. Nonetheless, these outcomes are deemed acceptable, as rotations around the yaw axis are not expected in subsequent Kalman filtering. Thus, a potentially imprecise estimation of $J_z$ does not jeopardize the overall experimental integrity. Eventually, for a more precise estimation, a viable approach is to employ one wheel, oriented with rotational axis aligned with yaw.

---

[1]"Faulhaber, DC Brushless Motors BX4" `https://www.faulhaber.com/it/prodotti/serie/3242bx4/`, accessed: 01/06/2023

It is imperative to recognize that, although the outcomes in Table 4.1 represent idealized performances (assuming no drag, sensor noise, and perfectly behaving RWs), the inertia's exact values are not achieved. This discrepancy is attributed to the bad-conditioning of the $\Phi$ matrix. Essentially, performance across all axes is not uniformly optimal due to the differential magnitudes of inertia moments, with the cross inertia slightly skewed by numerical inaccuracies.

Table 4.1: Ideal LSQ results, wherein the relative error with respect to the reference parameters is reported in brackets.

|          | Real state | RWs esteem        | Constrained esteem | Constraint + RW     |
|----------|------------|-------------------|--------------------|---------------------|
| $J_x$    | 0.4142     | 0.4149 (+0.17%)   | 0.4036 (-2.56%)    | 0.4110 (-0.77%)     |
| $J_y$    | 0.4042     | 0.4058 (+0.40%)   | 0.4052 (+0.25%)    | 0.4060 (+0.44%)     |
| $J_z$    | 0.4546     | 0.4534 (-0.26%)   | 0 (-100%)          | 0.4518 (-0.61%)     |
| $J_{xy}$ | 2.84e-6    | -4.43e-4 (-15592%)| -2.25e-04 (-7912%) | -0.0030 (-105633%)  |
| $J_{xz}$ | 0.0052     | 0.0040 (-23.08%)  | 0.0066 (+26.92%)   | -0.0162 (-411.54%)  |
| $J_{yz}$ | 0.0013     | 7.55e-4 (-41.92%) | 0.0072 (+453.85%)  | 0.0274 (+2008.46%)  |
| $mr_x$   | 0.0225     | 0.0224 (-0.44%)   | 0.0225 (0%)        | 0.0225 (0%)         |
| $mr_y$   | 0.0058     | 0.0059 (+1.72%)   | 0.0058 (0%)        | 0.0058 (0%)         |
| $mr_z$   | -0.5201    | -0.5211 (-0.19%)  | -0.5211 (-0.19%)   | -0.5248 (-0.90%)    |

The extremely high relative errors on the cross inertia parameters are due to a combination of two factors. Firstly, very small values of the real parameters cause the relative error to escalate due to the nature of its calculation, wherein, as evident from Equation (4.1),

$$\text{Relative error} = \frac{\text{Estimated value} - \text{Real value}}{\text{Real value}} \cdot 100\% \qquad (4.1)$$

the real value is the denominator in the computation. Secondly, significant deviations of cross inertia are detected, potentially attributable to the mathematics behind the LSQ technique. The underlying principle of the method is a square error minimization. Considering that the principal inertia values have a greater influence the overall error due to their greater magnitude, the LSQ necessarily provides more accurate estimates for them.

While initial observations may suggest that the estimation executed using RWs yields superior accuracy, it is crucial to underline that these are results under ideal conditions. When factors like noises and disturbances are incorporated, the performance significantly degrades. As presented in Table 4.2, even with the employment of Savitzky-Golay filtering, the outcomes deviate considerably from the ideal benchmarks.

Table 4.2: Effective LSQ results upon incorporation of noise and disturbances and Savitzky-Golay pre-filtering, with relative error in brackets.

|  | Real state | RWs esteem | Constrained esteem | Constraint + RW |
|---|---|---|---|---|
| $J_x$ | 0.4142 | 0.3916 (-5.46%) | 0.4017 (-3.02%) | 0.3992 (-3.62%) |
| $J_y$ | 0.4042 | 0.4055 (+0.32%) | 0.4015 (-0.67%) | 0.4046 (+0.10%) |
| $J_z$ | 0.4546 | 0.4642 (+2.11%) | 0.0351 (-92.28%) | 0.5085 (+11.85%) |
| $J_{xy}$ | 2.848e-6 | -0.0080 (-280734%) | -1.33e-4 (-4573%) | -0.0039 (-136718%) |
| $J_{xz}$ | 0.0052 | 0.0806 (+1450%) | -0.0337 (-748%) | 0.0231 (+344%) |
| $J_{yz}$ | 0.0013 | -0.0368 (-2931%) | 0.0098 (+654%) | 0.0049 (+277%) |
| $mr_x$ | 0.0225 | 0.0225 (0%) | 0.0225 (0%) | 0.0225 (0%) |
| $mr_y$ | 0.0058 | 0.0058 (0%) | 0.0058 (0%) | 0.0058 (0%) |
| $mr_z$ | -0.5201 | -0.5228 (-0.52%) | -0.5210 (-0.17%) | -0.5210 (-0.17%) |

Table 4.2 highlights that the estimations in this case typically underestimate the principal inertia, while overestimating the cross-inertia terms. The observed effect arises because the noise introduced is perceived by the least square procedure as a rapid dynamic. Hence, the method tries to allocate this fast dynamics to the inertia parameters. The consequent effect is the diminished estimation of principal inertia, as faster dynamics are typically linked to reduced inertia. Concurrently, there is an amplification in the estimation of cross-inertia. This is attributed to the noise being perceived as rapid and pronounced energy exchanges between the rotational axes. On the contrary, drag diminishes the amplitude of the oscillations, increasing the values of the estimated inertia. Hence, the two main disturbances have subtracting effects. However, the impact of noise predominates over the influence of aerodynamic drag due to the intentionally minimized time window, which

aims to mitigate the effects of unmodeled dynamics.

It is worth noticing that RWs estimation is more significantly affected by noise (Table 4.2). This phenomenon arises because the solicitation from the RWs is so light that the sensor noise exhibits a magnitude comparable to the angular rates.

## 4.2.    Active control techniques

This section offers an insight into the results achieved from the various active control techniques. At the core of these techniques is a two-fold process: a torque generation, and a subsequent manipulation to retrieve a command to send to the MMU.

Initially, the best-case scenario is presented, where outcomes under ideal conditions are showcased. This provides a benchmark against which real-world results can be gauged. Subsequently, the complexities of real-world experiment are incorporated, specifically the noise from sensors and the drag forces. Such considerations lead to more authentic, real-world representative performances, allowing for a thorough evaluation of the techniques. More importantly, by juxtaposing these results against the ideal, we can accurately pinpoint and address the root causes behind any eventual subpar performance. Through this structured presentation, readers will not only grasp the efficacy of the control strategies but also discern the tangible impacts of real-world impediments on system performance.

### 4.2.1.   PID results

The method's efficacy is demonstrated via a simulation spanning 600 seconds, within which the time history of the offset and the profile of angular velocities are key metrics indicating the control's success. These performance metrics are illustrated and further discussed in Figure 4.1. The results are obtained placing the MMUs 3 mm, from the centre of the leadscrew, therefore with initial offset of $[5.29e\text{-}4, 2.64e\text{-}4, -0.08525]^T$ m. The initial condition on the state are assigned as null yaw, pitch and roll angles, and null angular velocities. Hence, the platform to oscillate under the unique effect of its offset.

(a) Angular rates.
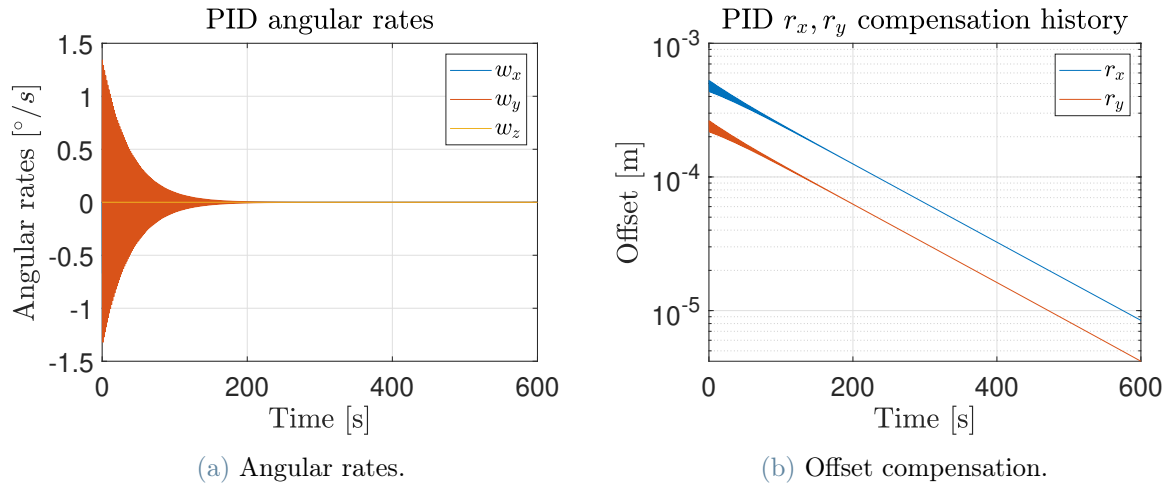
(b) Offset compensation.

Figure 4.1: Ideal PID performances.

However, these performances do not incorporate the aerodynamic disturbance and measurements noise, which are introduced as detailed in Section 3.1, resulting in the performances encapsulated in Figure 4.2.
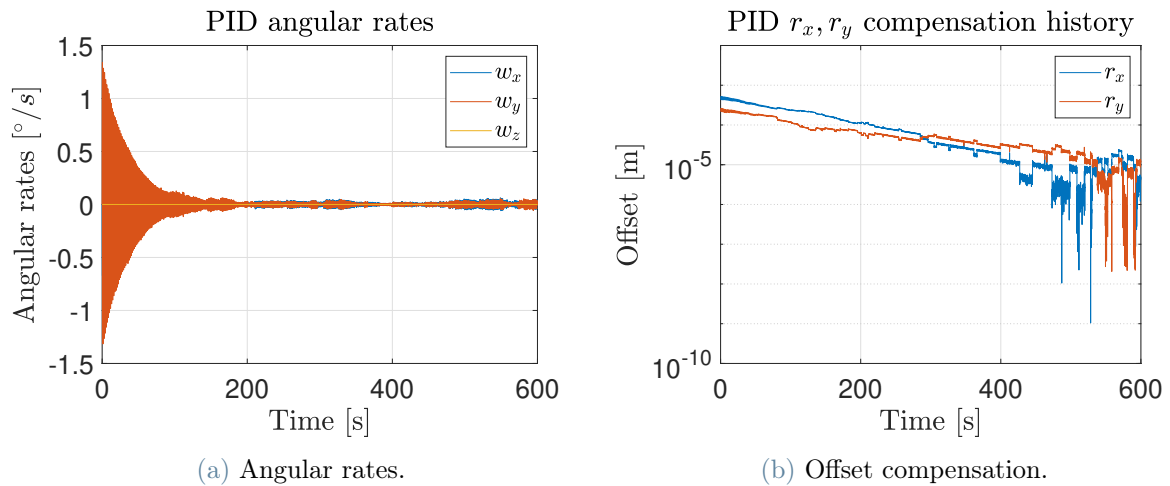


(a) Angular rates.

(b) Offset compensation.

Figure 4.2: PID performances upon incorporation of noise and disturbances.

Table 4.3: PID compensation performances.

|  | $r_x$ [m] | $r_y$ [m] | $r_z$ [m] |
|---|---|---|---|
| **Ideal PID** | 8.45e-06 | 4.18e-06 | -0.08525 |
| **PID** | 4.82e-06 | 1.25e-05 | -0.08525 |

Table 4.3 resumes the results of the ideal and realistic cases of the methodology. When disturbances and noises are incorporated, there is a notable degradation in performance, with the error amplifying by roughly an order of magnitude. Nonetheless, the method continues to exhibit commendable convergence capabilities. This is largely attributed to the PID methodology's predominant reliance on attitude knowledge, which, in the context of STASIS platform, is exceptionally precise.

### 4.2.2. Non-linear control based on gravity vector

The efficacy of the method, namely $\hat{\mathbf{g}}$ control, is assessed under the same conditions as previously, to ensure result comparability, both in the ideal scenario and in the more realistic setting where noises and aerodynamic disturbances are incorporated.



(a) Angular rates.                         (b) Offset compensation.
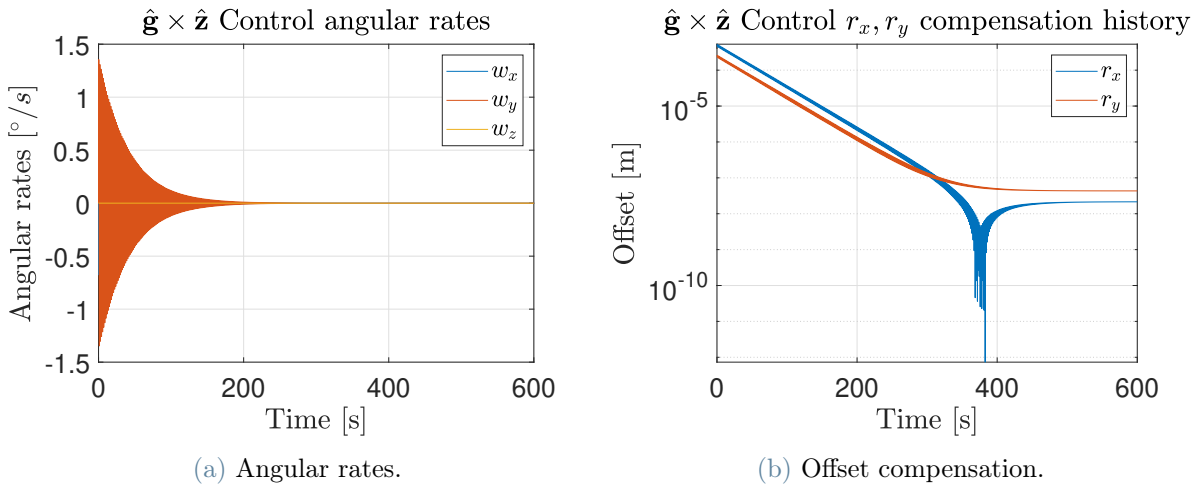
Figure 4.3: Ideal $\hat{\mathbf{g}}$ control performances.

(a) Angular rates.
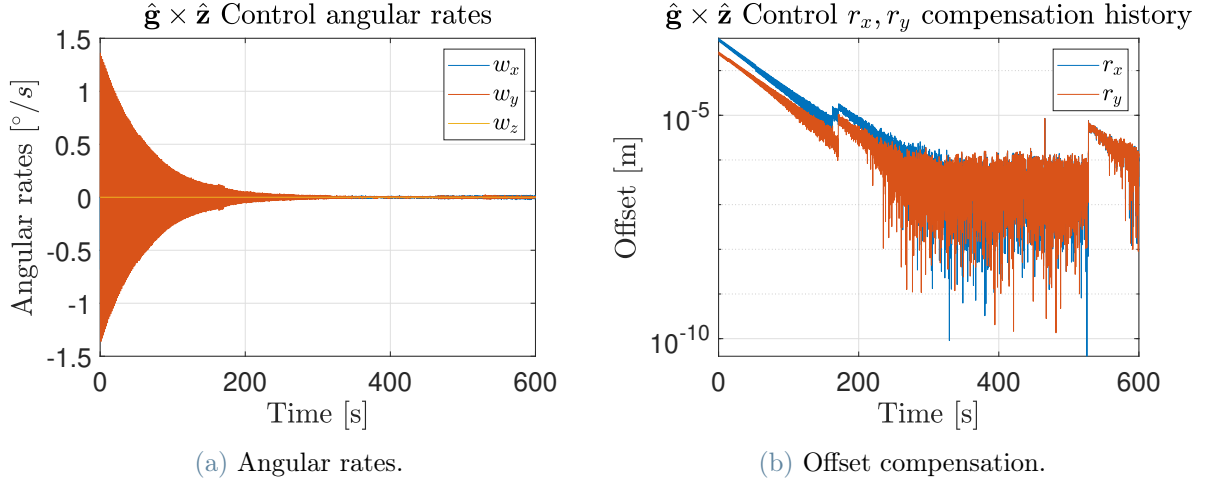
(b) Offset compensation.

Figure 4.4: $\hat{\mathbf{g}}$ control performances upon incorporation of noise and disturbances.

Table 4.4: $\hat{\mathbf{g}}$ control compensation performances.

| | $r_x$ [m] | $r_y$ [m] | $r_z$ [m] |
|---|---|---|---|
| **Ideal $\hat{\mathbf{g}}$ control** | -2.14e-08 | 4.32e-08 | -0.08525 |
| **$\hat{\mathbf{g}}$ control** | -2.54e-07 | 4.17e-07 | -0.08525 |

Table 4.4 demonstrates, as anticipated, that the nonlinear compensation significantly outperforms the PID. Moreover, similar to the previous instance, the strong dependence on accurate attitude knowledge ensures satisfactory convergence even in scenarios with incorporated disturbances.

## 4.2.3.   Non-linear control based on angular momentum

Unlike the other methods, this nonlinear control is highly dependent on the noisy angular rates, given that the platform's angular momentum is computed at each step. Consequently, when aerodynamic disturbances and measurement noise are introduced, a significant decline in its performance is expected.
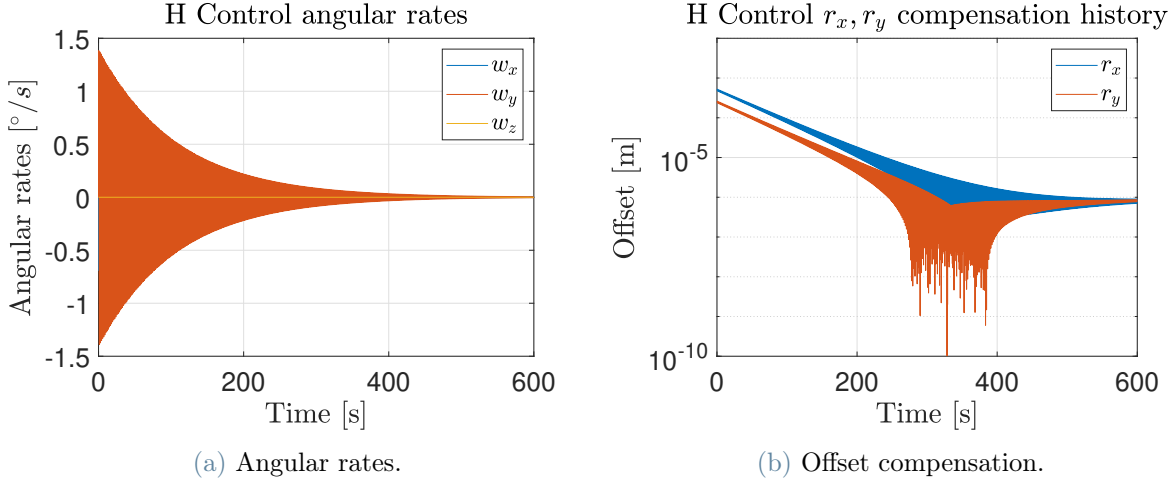
(a) Angular rates.

(b) Offset compensation.

Figure 4.5: Ideal non-linear **H** control performances.



(a) Angular rates.

(b) Offset compensation.
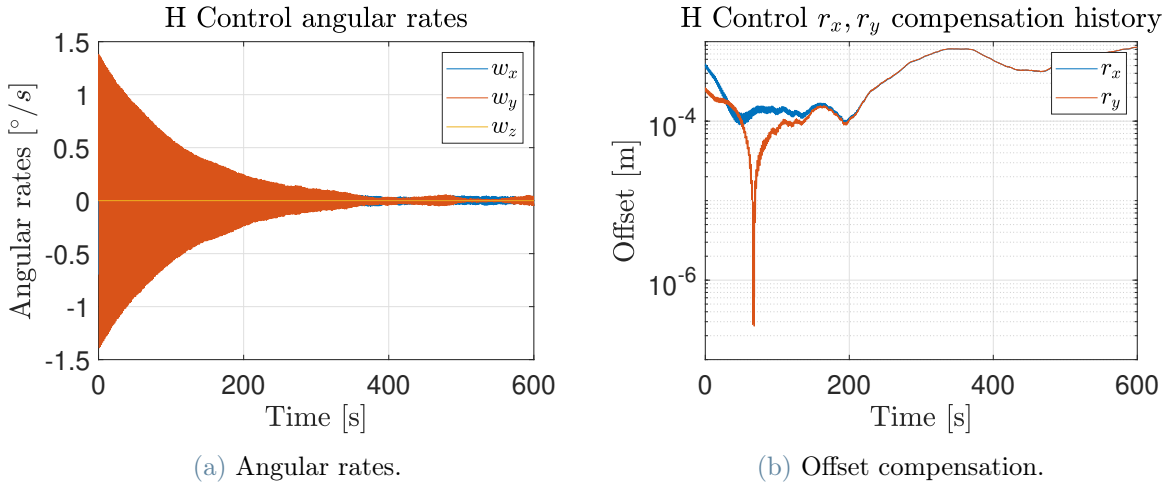
Figure 4.6: Non-linear **H** control performances upon incorporation of noise and distur-bances.

As anticipated and evident from the Figure 4.6, the method's performance deteriorates significantly in a more realistic scenario. In fact, the method cannot be conclusively deemed to achieve full compensation. This degradation arises because the control relies on the cross product of the gravity vector and the angular momentum, therefore on a composition of two corrupted measurements, as visible hereafter,

$$\dot{\delta \mathbf{r}} = m\mathbf{\Gamma}[\mathbf{g}\times]^T\mathbf{H} + K_I \int_{t_0}^{t} m\mathbf{\Gamma}[\mathbf{g}\times]^T\mathbf{H}. \tag{4.2}$$

Indeed, both the gravity vector and angular momentum are skewed due to imprecise

ground truth and BNO055 measurements. Therefore, the two primary compromised measurements compound, resulting in amplified errors. Nonetheless, as depicted in Figure 4.6a, the angular rates do converge to zero, likely attributable to the aerodynamic drag incorporated in the model.

Table 4.5: **H** control compensation performances.

|  | $r_x$ [m] | $r_y$ [m] | $r_z$ [m] |
|---|---|---|---|
| **Ideal H control** | 9.11e-07 | -7.66e-07 | -0.08525 |
| **H control** | 8.40e-04 | -8.44e-04 | -0.08525 |

## 4.2.4. Non-linear control based on offset estimation

Given its conceptual similarity to the preceding control, one would anticipate comparable performance in this instance. Indeed, the underlying principle of this control is still angular momentum compensation, also if in a different arrangement. Indeed, the momentum cancellation does not derive from direct knowledge of the platform angular momentum but rather from a time-integrated estimation of the CR-CM offset. Figure 4.7 and Figure 4.8 showcase the results in the ideal and corrupted measurement cases, highlighting the significant degradation in compensation capabilities upon the incorporation in the simulations of noise and disturbances.
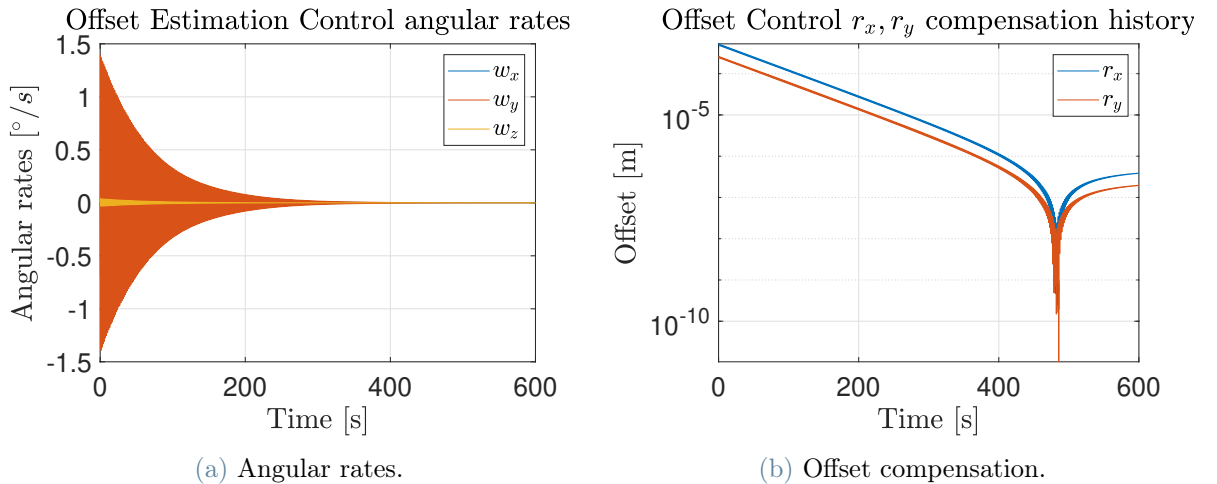


(a) Angular rates.

(b) Offset compensation.

Figure 4.7: Ideal non-linear $\tilde{\mathbf{r}}$ control performances.

Offset Estimation Control angular rates

Offset Control $r_x, r_y$ compensation history

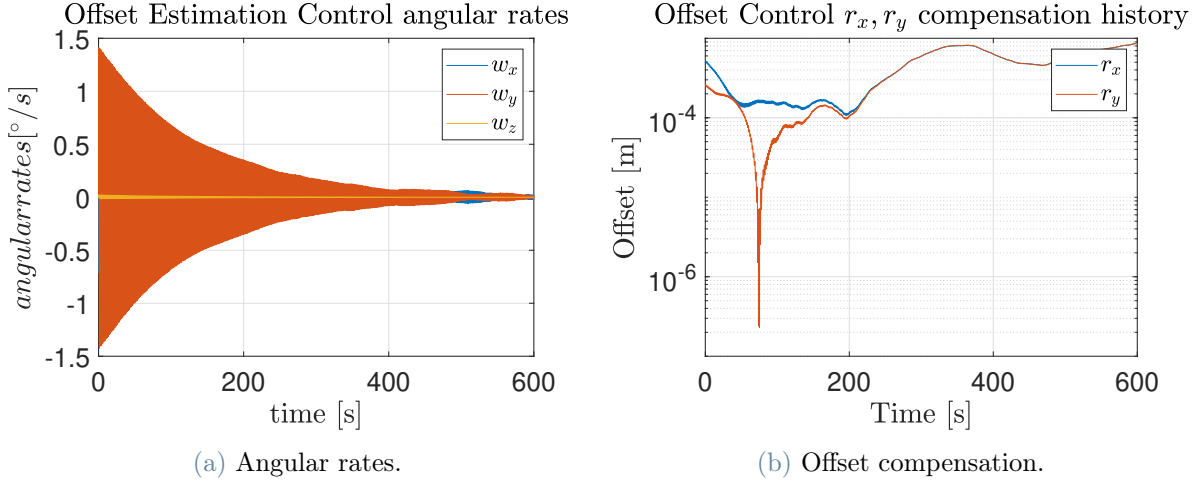(a) Angular rates.

(b) Offset compensation.

Figure 4.8: Non-linear $\tilde{\mathbf{r}}$ control performances upon incorporation of noise and disturbances.

Once again, the imprecise angular rates measurements lead the control to diverge, failing to fully compensate for the offset. As with the prior instance, albeit in a different arrangement, error accumulates through corrupted measurement composition, indeed, implementing the control adaptation law

$$\dot{\hat{\mathbf{r}}} = \mathbf{\Psi}^T \boldsymbol{\omega}, \tag{4.3}$$

two flawed measurements are employed, resulting in a significant error escalation. However, due in part to the dampening effect of aerodynamic drag on the angular rates, the control successfully nullifies them.

Table 4.6: $\tilde{\mathbf{r}}$ control compensation performances.

|  | $r_x$ [m] | $r_y$ [m] | $r_z$ [m] |
|---|---|---|---|
| **Ideal $\tilde{\mathbf{r}}$ control** | -3.86e-07 | -1.95e-07 | -0.08525 |
| **$\tilde{\mathbf{r}}$ control** | 8.56e-04 | -8.57e-04 | -0.08525 |

## 4.3.   Kalman filtering

Within the realm of Kalman filtering, the results are bifurcated into two classifications: one achieving solely the offset estimation, and another that integrates inertia into the estimation framework. Despite these variations, the methodological foundation remains

consistent, focusing on the ideal results and, therefore, gauging the effects of measurements corruption and disturbances. This approach facilitates an evaluation of the filter's robustness against environmental disturbances and noise, therefore proving crucial for comprehensively addressing the operative research questions. This section underscores the importance of meticulous filter initialization, which is instrumental in ensuring optimal outcomes. Specifically, all requisite covariance matrices are evaluated based on plausible preliminary offset knowledge and noise variance metrics. Notably, the adopted methodology exhibits a pronounced dependency on the mathematical model. Thus, unlike the active control and least square methods, wherein noise is the principal source of error, dynamic disturbances can potentially represent the major cause of degradation in performances in this case.

### 4.3.1. 6-state Kalman filters

The efficacy of the developed filter is evaluated through simulations conducted in the MATLAB environment over a span of 20 seconds. This concise time window is selected aiming to curtail the influence of any unmodeled dynamics within the simulation framework. The covariance matrix is initialized conjecturing a poor knowledge of both the angular rates and the offset:

$$\mathbf{P}_0 = diag([1e-1, 1e-1, 1e-1, 1e-4, 1e-4, 1e-4]). \tag{4.4}$$

On the other hand, the matrix $\mathbf{R}_k$ is mathematically derived from the BNO055 IMU sensor datasheet, according to the ESH availability. The primary source of error in this sensor is attributed to the white noise, which amounts to 0.099°/s. Therefore, in the hypothesis of null cross-correlation between the noises of different axes, the measurement covariance matrix can be set to:

$$\mathbf{R}_k = diag([3.1e-6, 3.1e-6, 3.1e-6]) \text{ rad}^2/\text{s}^2. \tag{4.5}$$

In the simulation the measurements noise $\mathbf{v}_k$ is generated through the *mvnrnd* MATLAB function. A set of random vectors from the multivariate normal distribution is generated with zero expected value and $0.099°/s$ standard deviation, consistently with the assumptions performed on the IMU sensor.

The process noise covariance $\mathbf{Q}_k$ should be set according to the expected effect of the dynamics' uncertainties. However, due to the absence of a definitive aerodynamic coefficient value for STASIS, establishing the process noise matrix with a rigorous criterion becomes

unfeasible. Consequently, the matrix values are derived from existing literature, adjusted upwards by an order of magnitude to account for the procedure's preliminary nature.

$$\mathbf{Q}_k = diag([1e-5, 1e-5, 1e-5, 1e-10, 1e-10, 1e-10]). \tag{4.6}$$

As detailed in Section 2.2, unmodeled effects are incorporated in the simulations, with an unknown damping effect $-\varepsilon \Lambda J \boldsymbol{\omega}$. However, this effect is not integrated in the prediction step. Indeed, in the current configuration of STASIS, the absence of an estimation for damping effects is a recognized limitation, slated for future enhancement. Incorporating damping effects in the prediction step, at this juncture, could compromise the verisimilitude of the simulations relative to empirical experiments. Conversely, the goal at this stage is to architect a filter exhibiting maximal robustness, even when driven by a simplified dynamic model. Nonetheless, forthcoming iterations should prioritize the precise quantification and integration of dissipative aerodynamic effects to bolster the system's realism and efficacy.

## Theoretical results

Simulations are conducted with uniform matrices applied across both filters, and a time step is configured at 0.01 s, aligning with the peak acquisition frequency of 100 Hz of the BNO055 sensor. As delineated in Table 4.7, the superior performance characteristics of the UKF become evident, primarily attributable to its ability to bypass the constraints of linearity assumptions.

Table 4.7: EKF and UKF ideal results.

| Filter | error [m] |
|--------|-----------|
| **EKF** | $[5.94e-07, 1.56e-07, 6.25e-06]$ |
| **UKF** | $[3.45e-07, 2.31e-07, 3.68e-06]$ |

Yet, the simulation has not incorporated the aerodynamic disturbance, a critical component for aligning with real-world experimental conditions.

## Effective results

When the aerodynamic effect is integrated into the model, the filters' performance diminishes, with the residual error increasing approximately by an order of magnitude compared to the ideal scenario, as illustrated in Table 4.8.

Table 4.8: EKF and UKF results upon integration of the aerodynamic effect.

| Filter | error [m] |
|--------|-----------|
| EKF | $[6.78e - 06, 2.27e - 06, 4.41e - 05]$ |
| UKF | $[3.91e - 06, 9.91e - 07, 2.09e - 05]$ |

The outcomes prove to be commendable to achieve simulations conducted under microgravity conditions. Notably, they effectively reduce the CM-CR offset torque to levels comparable with other inherent disturbances.

As a matter of fact, owing to the extremely brief observation time window, the unmodeled effects do not lead to divergence in the procedure. Figure 4.9 and 4.10 present the behaviour of the errors on the state during the simulation. The associated $3\sigma$ value, retrieved at each step from the state covariance matrix, is reported as well.
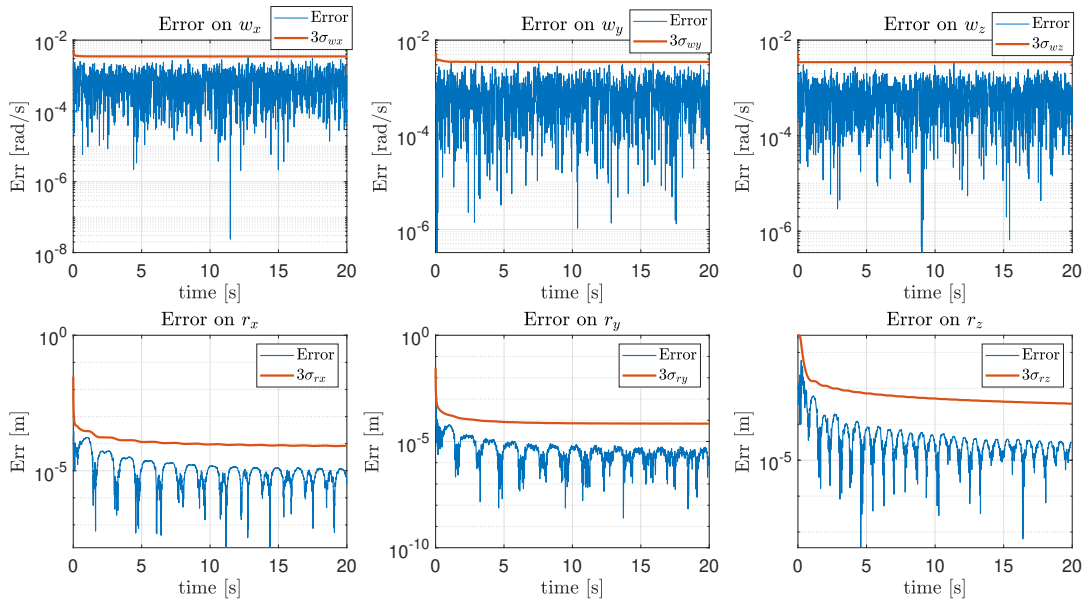


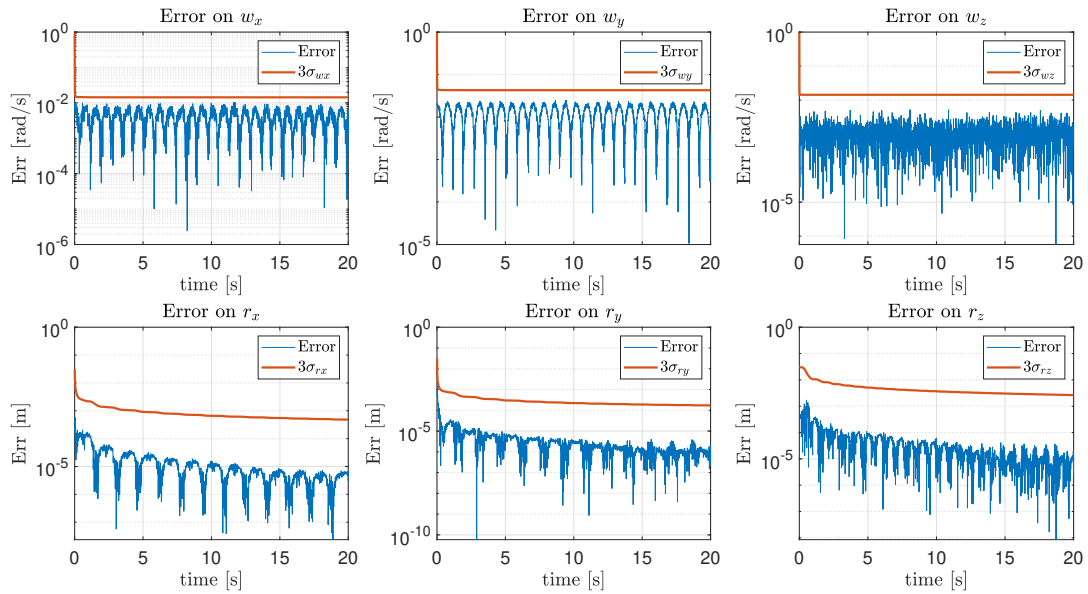Figure 4.9: 6-state EKF upon incorporation of the aerodynamic momentum.

Figure 4.10: 6-state UKF upon incorporation of the aerodynamic momentum.

## 4.3.2.  Augmented Kalman filters

To demonstrate the sensitivity of the aforementioned procedure to inaccuracies on the inertia matrix, the EKF was iteratively tested with progressively larger error introduced on the inertia parameters.
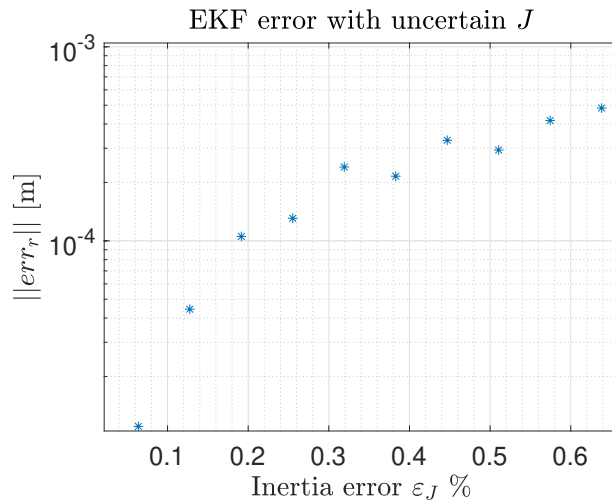


Figure 4.11: EKF error norm upon integration of increasing uncertainty on the inertia parameters. The percentage is respective to the maximum inertia.

As depicted in Figure 4.11, even minor deviations in the inertia parameters lead to an

exponential surge in error, especially when close to the true solution. Beyond this initial zone, the error tends to rise moderately with increasing perturbation of inertia parameters. Specifically, the pattern emerging is a great error escalation at low relative inertia errors levels, rapidly increasing the EKF error of more than one order of magnitude, followed by a more gradual linear-like increase for higher errors on the inertia. As consequence, integrating an augmented procedure is of utmost interest in this application.

The filter's configuration remains consistent with the previous scenario. However, with the inclusion of inertia parameters in the state, both the state covariance matrix and the process noise covariance matrix have been coherently augmented. As anticipated in Section 2.3, RWs shall be utilized to rigorously address the problem of partial observability inherent to an estimation incorporating both the inertia parameters and the offset vector. The RWs reference is the same for the LSQ estimation, to ensure a consistent basis for comparing results.

## Theoretical results

Avoiding strict assumptions on the inertia parameters, the procedure allows to obtain satisfactory results on the offset estimation. The results are reported in Table 4.9.
.

Table 4.9: Augmented EKF and UKF ideal results.

| Filter | error [m] |
|--------|-----------|
| EKF | $[2.77e-06, 9.84e-07, 1.16e-05]$ |
| UKF | $[2.98e-06, 8.89e-07, 1.09e-05]$ |

In general, simulations proved that this approach may lead to optimal results, whenever the available RWs have high degree of control on the platform. Moreover, Table 4.9 highlights that there is no significant advantage between augmented UKF and EKF, as the final errors in the estimation are comparable. The reasons behind these results include:

1. Very high accuracy of CSD in computing the Jacobian in the EKF.

2. MATLAB inner algorithms, which typically adopt different procedures when the complexity of the problem increases, in order to optimize the run time.

3. The higher complexity of the augmented case may cause that the linear assumption inherent in the EKF is not the major source of error.

## Effective results

Figures 4.12–4.14 display the outcomes from a 20-second simulation. It is evident that the method does not converge when aerodynamic disturbance is introduced. This occurrence can be attributed to the fact that the wheel solicitation has comparable magnitude to the effects of the unmodeled dynamics, preventing the method from discerning the dynamic responses due to wheel solicitation from the drag effect. Consequently, the methodology could be executed with a RWs actuation, wherein the control trajectory is designed to embed quiescent phases, aimed to filter out the aerodynamic effect from the observation.
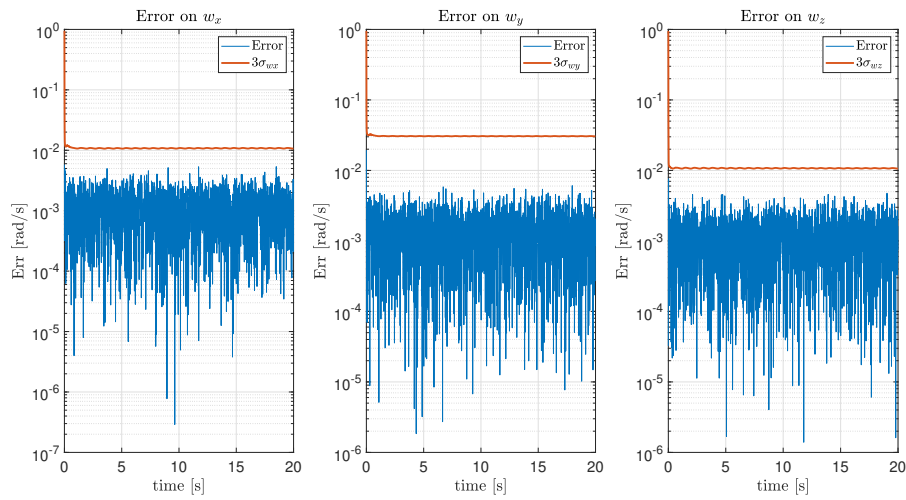


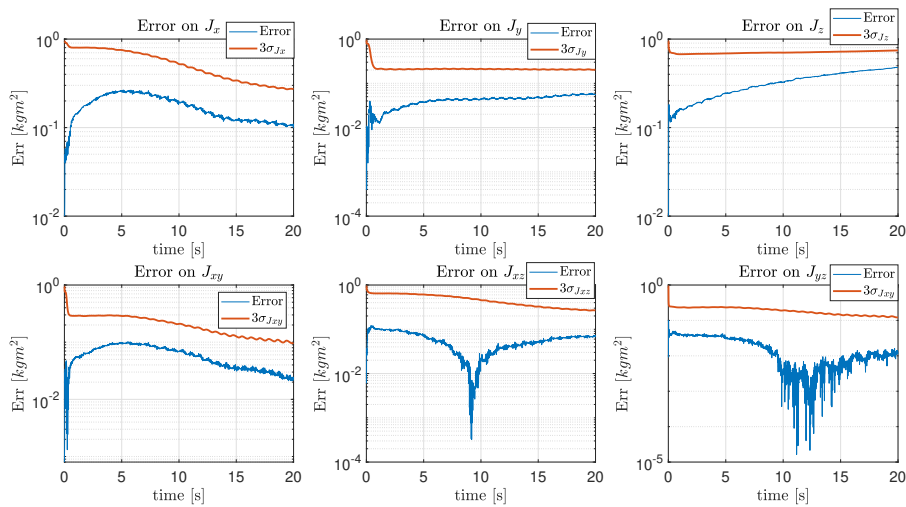Figure 4.12: Augmented EKF angular rates error.
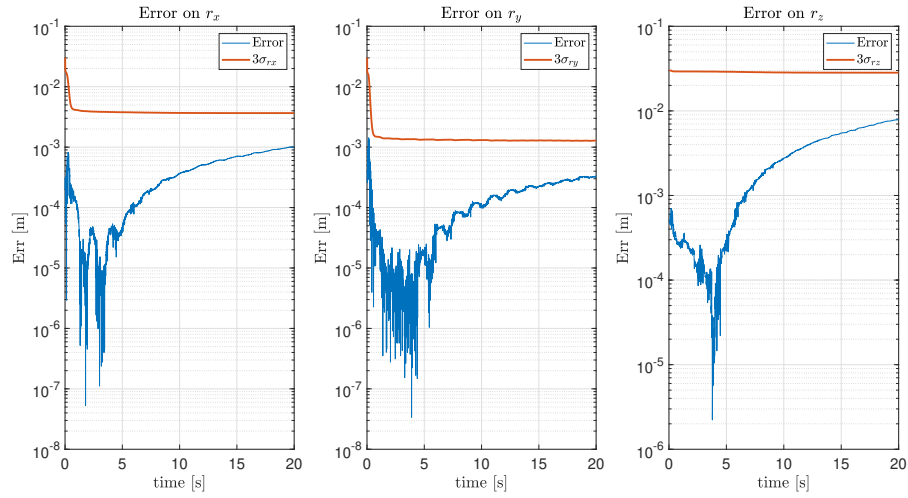


Figure 4.13: Augmented EKF inertia error.

Figure 4.14: Augmented EKF offset error.

Figures 4.15–4.17 illustrate that the unscented Kalman filter falls short in its effectiveness as well, leading to the conclusion that the drag effect neglecting is a rather stringent assumption for these approaches.
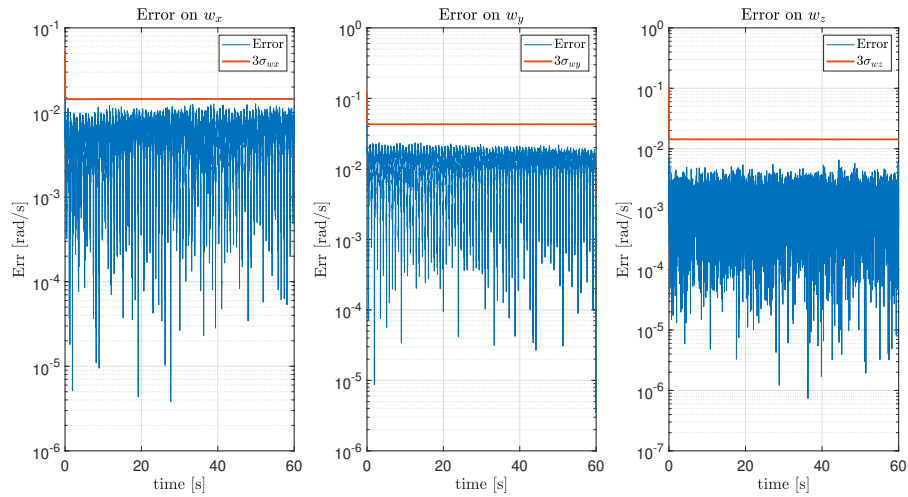


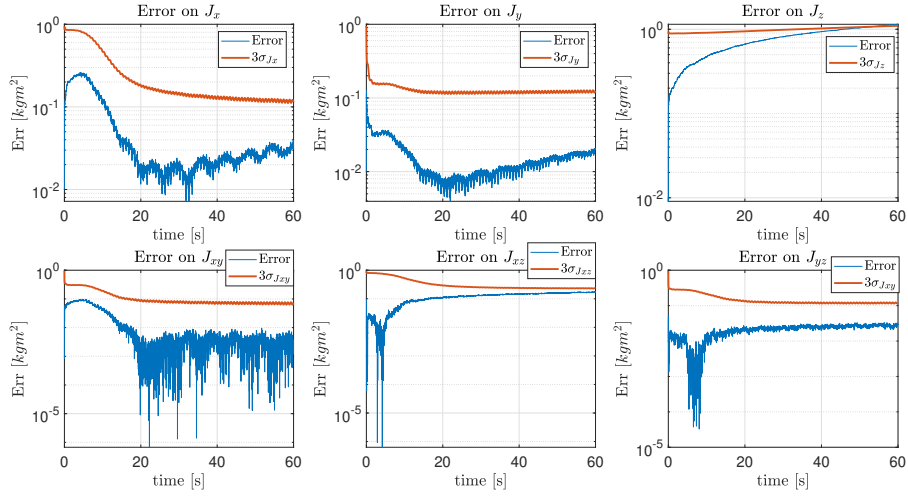Figure 4.15: Augmented UKF angular rates error.
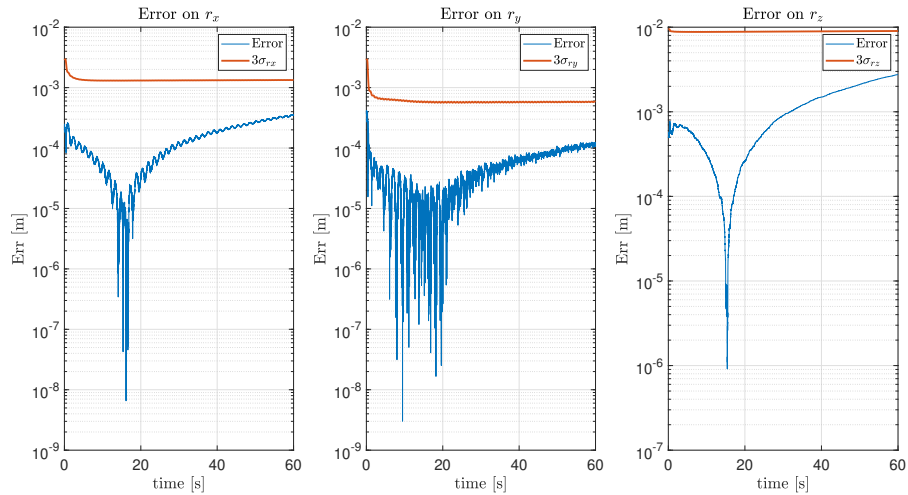
Figure 4.16: Augmented UKF inertia error.



Figure 4.17: Augmented UKF offset error.

In summary, the filters can reconstruct the dynamic behavior and distinguish it from noise. However, they fall short in attributing this dynamic response to the 3D pendulum-like motion. Consequently, these procedures are recommended exclusively when a precise estimation of the drag coefficients is available, or in the case that the available torque provided by RWs is order of magnitude above the drag damping effect.

## 4.4.    Final remarks

Following the development and implementation on STASIS digital twin, it is imperative to outline several key aspects contingent to the initial strategy design. The ensuing

discussion will deliberately bypass the augmented filters. Although these have been implemented as a foundation for imminent advancements, they contravene the requirement of excluding RWs involvement from the experiment.

The residuals shall be computed considering that, in absence of RWs, the LSQ method shall rely on $r_x$ and $r_y$, given by an active control method, to output $r_z$. Therefore, the Kalman filters could be fed with the inertia matrix to refine the estimations. Table 4.10 reports the attainable order of magnitude of the residual for the selected methodologies.

Table 4.10: Residual evaluation across different methods.

| Methods | $r_x$ [m] | $r_y$ [m] | $r_z$ [m] |
|---------|-----------|-----------|-----------|
| Least squares | - | - | $\sim 10^{-4}$ |
| PID | $\sim 10^{-6}$ | $\sim 10^{-6}$ | - |
| $\hat{\mathbf{g}} \times \hat{\mathbf{z}}$ control | $\sim 10^{-7}$ | $\sim 10^{-7}$ | - |
| $\mathbf{H}$ control | $\sim 10^{-4}$ | $\sim 10^{-4}$ | - |
| $\tilde{\mathbf{r}}$ control | $\sim 10^{-4}$ | $\sim 10^{-4}$ | - |
| EKF | $\sim 10^{-6}$ | $\sim 10^{-6}$ | $\sim 10^{-5}$ |
| UKF | $\sim 10^{-6}$ | $\sim 10^{-7}$ | $\sim 10^{-5}$ |

A critical assessment is imperative at this juncture. In pursuit of eliminating the involvement of RWs, an experiment with a sequential structure has been crafted, consecutively setting one phase output as input of the following one. The active control technique provides the basis to build the mathematical constraint $B\mathbf{x} = \mathbf{c}$; then, the LSQ outputs the inertia parameters for the Kalman filter (Figure 3.7). While such a design inherently opens avenues for error propagation, it is essential to underscore that the inherent compensatory capabilities remain theoretically unaltered. In fact, the active control methodology boasts a precision that surpasses the LSQ by several orders of magnitude. Consequently, any propagated errors are anticipated to have a negligible impact on the final outcomes.

Contrarily, the LSQ method emerges as the limiting factor in the pursuit of superior accuracy. Directly feeding the inertia parameters into a Kalman filter under these circumstances could potentially instigate divergence. However, once the LSQ phase is terminated, the complete offset knowledge becomes accessible. As such, the propagated error is likely to influence only the refinement of the results, rather than the core computations. In summary, taking into account the intrinsic challenges associated with a sequential structure experiment, the preliminary evaluation posits that the initial experimental design retains its robustness to the myriad of complexities inherent to a real-world experiment.

Indeed, it is projected to adeptly facilitate comprehensive CR-CM offset compensation. The initial experimental design was virtually modeled to provide a mathematical basis of the performed dissertation. The deviations in the parameters $r_x, r_y$ are propagated to the LSQ, attaining null effect due to the very high accuracy of the $\hat{\mathbf{g}} \times \hat{\mathbf{z}}$ control. Hence, the resulting inertia is inputted to the Kalman filtering techniques. Table 4.11 demonstrates, through an error on $r_z$ reduced of the 53% with respect to EKF's, the major capability of UKF to reject uncertainties effects, a crucial feature for experiments with sequential structure.

Table 4.11: EKF and UKF of the simulated experiment, fed with the inertia parameters attained through the constraint+RW LSQ fitting, detailed in Section 4.1.

| Filter | error [m] |
|---|---|
| **EKF** | $[1.96e - 05, 8.88e - 06, 1.68e - 04]$ |
| **UKF** | $[2.14e - 05, 1.71e - 05, 7.85e - 05]$ |

As a consequence, the designed balancing procedure is theoretically capable to reach $10^{-7}$ m accuracy for planar balancing, and $10^{-5}$ m for the vertical offset compensation. Since these results show an improvement over the previous LSQ estimate on $r_z$, the initial design of the experiment is corroborated. Nevertheless, these findings do not constitute a mathematical proof of the procedure's convergence in practical applications. Real-world complexities can substantially alter outcomes, potentially leading to divergence in the UKF performance.

# 5 | Hardware-in-the-loop experimental procedure

The experimental procedure is meticulously designed to align with the STASIS attitude determination system, maximizing the benefits of low-noise measurements while minimizing the impact of the most noise-infused measures. However, the true challenge of the experiment is to ensure the procedure's effectiveness, even with the myriad of complexities inherent in a real-world test. As a consequence, a crucial feature of the approach is its ability to operate without cables, due to the challenges associated to the modeling of their influence on STASIS attitude dynamics.
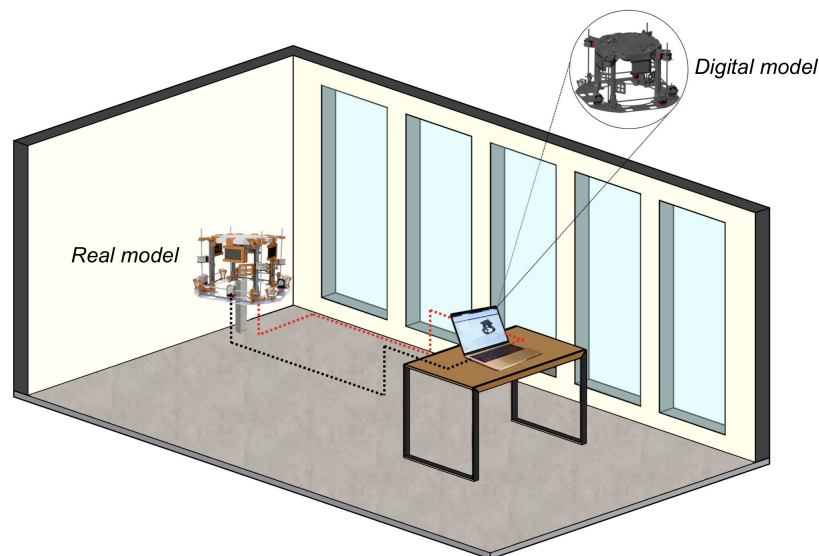


Figure 5.1: Real-time wireless interface [22].

Hence, requests and commands with the attitude sensors, the IMU and the actuators are carried out through either the User Datagram Protocol (UDP), Transmission Control Protocol (TCP) and Hypertext Transfer Protocol (HTTP) communication protocols [73], briefly presented in Table 5.1.

Table 5.1: Comparison of UDP, TCP, and HTTP protocols.

| | UDP | TCP | HTTP |
|---|---|---|---|
| **Key Features** | *Fast*: No handshake. *No Guarantees*: No error checking or acknowledgment. *Lightweight*: No connection establishment. | *Connection-oriented*: Requires handshake. *Reliable*: Ensures data integrity and delivery. *Flow control*: Prevents network congestion. | *Request-Response* mechanism. *Stateless*: Without session persistence. *Reliable*: Ensures data integrity and order. |
| **Reliability** | Fast but data might be lost. | Ensures data delivery through acknowledgments and re-transmissions. | Ensures data delivery but might be slower due to the inclusion of the header in the message. |
| **Complexity** | Simple to implement. | More complex due to reliability and flow control. | Higher level, more intricate setup and handling. |
| **Use** | Streaming media. | Web browsing, emails, file transfers. | Web browsing, data transfer in web applications. |

In essence, the UDP protocol offers a lightweight solution without assurances for data integrity or sequence maintenance. In contrast, both TCP and HTTP protocols, despite their higher computational overhead, provide the advantage of ensuring the integrity of the data received.

## 5.1.  The experiment

In adherence with the conduced research, the experiment was meticulously designed to ensure unbiased balancing (i.e., independent from CAD data), while ensuring compatibility with the hardware available in ESH. In this context, a crucial feature of the compensation methodology is its reliance solely on attitude data and angular velocity measurements, abstaining from the need for additional information. This element of the strategy ensures its general applicability, maintaining its efficacy regardless of potential future modifica-

tions to the simulator's mock-up, that may invalidate any prior hypothesis on inertia or CR-CM offset. Furthermore, by not depending on prior knowledge of inertia parameters or offset estimations, the method prevents the possibility of error accumulation in the balancing experiment.

### 5.1.1. The experiment phases

The design of the experiment is fundamentally underpinned by two primary considerations. First, it shall adhere scrupulously to all the specifications delineated in Section 1.5. Second, it shall be driven by an ambition to realize peak performance, a goal that is substantiated through the critical analysis provided in Chapter 4.

Due to STASIS hardware availability constraints, the RWs are omitted from the experimental procedure. Hence, being the torque generated by the displacement of a set of movable masses, control is implicitly constrained to a plane orthogonal to the gravity vector, making the problem under-actuated. To withstand the non-full controllability of the system, a two-step experimental procedure is proposed.

- **$1^{st}$ step**: Planar balancing is executed through the MMUs. Providing the values of $r_x$ and $r_y$, a pivotal mathematical constraint to address partial observability. Notably, the spectrum of viable strategies to perform unbiased balancing is reduced to PID and $\hat{\mathbf{g}} \times \hat{\mathbf{z}}$ control methodologies, as encapsulated in Chapter 3.

- **$2^{nd}$ step**: The free motion of the platform is observed and analyzed to solve the constrained problem detailed in Section 3.2.

$$\mathbf{\Phi}\mathbf{x} = 0, \text{ such that } B\mathbf{x} = \mathbf{c}. \tag{5.1}$$

The availability of the inertia parameters is therefore exploited for eventual results refinement, through Kalman filtering.

To yield satisfactory outcomes, the methodology necessitates highly accurate information regarding both the MMU and the platform masses, whose involvement is implicit in both the steps. For instance, in computing the command to the masses in the active control technique, as illustrated in Equation (5.2),

$$\mathbf{R}_i = \frac{\mathbf{g} \times \boldsymbol{\tau}_u}{||\mathbf{g}||^2 m_i}, \tag{5.2}$$

the knowledge of the MMU mass is required. Additionally, after the vertical offset is retrieved, it shall be mapped into a command for the vertical masses. Assumed $m_z$ to be

the sum of the vertical MMU, the command is expressed as:

$$\Delta R_z = \frac{m}{m_z \Delta r_z},\qquad(5.3)$$

requiring therefore the knowledge of the platform-MMU mass ratio. A further example involves the errors propagated through the mathematical constraint $B\mathbf{x} = \mathbf{c}$. Indeed, while $\mathbf{c} = [mr_x \ mr_y]^T$, the active control technique provides the knowledge of $[r_x \ r_y]^T$, rendering the procedure vulnerable to any uncertainty on the platform mass. Thus, prior to procedure execution, the MMU and the platform shall be necessarily weighed.

It is noteworthy that a two-step procedure aligns seamlessly with the experimental design requirements, effectively reducing disturbance impacts and obviating the need for RWs, while simultaneously leveraging the symmetrical arrangement of the MMU. Specifically, the symmetry inherent in the MMU setup ensures that the vertical compensation's influence on the preceding phase is minimized, given that the center of gravity of the vertical MMU configuration is anticipated to be proximal to the CR to the greatest extent possible. This proximity strengthens the theoretical justification for segregating the problem into distinct planar and vertical balancing tasks, thereby enhancing the methodological robustness of the approach.

## $1^{st}$ step - planar balancing

The procedure is performed synchronously through UDP and HTTP communication as receiver from the attitude sensors and as sender with the MMUs. In particular, this step proceeds according the following phases:

1. **Facility mock-up**. During this initial setup phase, equipment is installed on STASIS in a manner that mirrors the configuration present in the digital twin. This similarity ensures that the simulated control gains are already closely aligned with their optimal values. This phase is crucial as it allows for the evaluation of result consistency against the digital framework.

2. **Experiment initialization**. The stepper motors are interconnected to the control software via a wireless setup. Subsequently, a concise examination of the movable masses control is conducted, scrutinizing several aspects: adherence to the issued commands, the effect of the lag between command dispatch and actuation, and the precision in the recorded command sequence for each stepper motor. Paramount to this process is ascertaining the positive direction of motion within each MMU frame, a critical step for maintaining consistency throughout the experiment. Furthermore, special emphasis is placed on verifying the integrity of the sensor readings, especially

focusing on the zero-offset calibration of the IMU.

3. **Initial conditions imposition**. Initial conditions on attitude and angular rates are imposed by randomly moving the set of movable masses. Being an under-actuated problem, under the negligible cross-inertias assumption, no rotation around yaw is expected in this phase. Therefore, measuring this rotation may give an idea of the consistency of the main hypothesis underneath the procedure, that is the negligibility of the cross-inertia terms.

4. **Active control balancing**. At the core of the process lies the activation of the closed-loop control. Real-time data on attitude orientation and angular velocities are transmitted via HTTP and TCP. Then, they are decoded and analyzed to generate a UDP directive dispatched to the MMUs. This phase unfolds iteratively, continuing until the command value converges on a predetermined threshold, denoted as $\varepsilon_A$. In adherence with the peak demonstrated performances, the selected methodology is the $\hat{\mathbf{g}} \times \hat{\mathbf{z}}$ control. Figure 5.2 illustrates the logical sequence of the control phase.
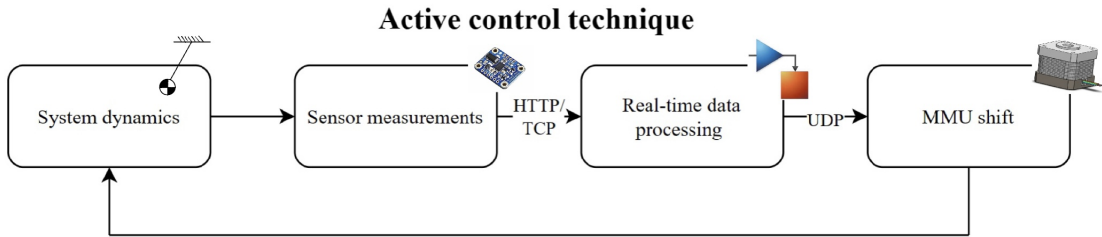


Figure 5.2: Active control scheme.

5. **Data collection**. To complete the $1^{st}$ step, some relevant data, such as the exact final position of the MMUs and the attitude orientation, are collected for further analysis.

The rigor in executing this $1^{st}$ step is of paramount importance, as any error is propagated to the following step through the mathematical constraint. Therefore, in this first half of the procedure the checklist should be severely observed, in particular regarding phase 2, wherein the performance of each controlled MMU is assessed separately.

## $2^{nd}$ step - observe and compensate

In the $2^{nd}$ step the MMUs are not actively controlled. Instead, STASIS dynamics is observed with no interference, collecting data transmitted by sensors via TCP and HTTP protocols. Then, MMUs position adjustment is performed to cancel the vertical offset.

6. **Initial conditions imposition**. This phase is conceptually identical to phase 3. However, in this case, the final MMUs position is not randomic. Contrarily, they are placed with a precisely known distance from the point of offset compensation, possibly maximizing it, to retrieve an accurate mathematical constraint $B\mathbf{x} = \mathbf{c}$. It should be noted that a significant MMUs shift maximizes the vector $\mathbf{c} = m[r_x \ r_y]^T$, enabling to reduce at minimum the relative error.

7. **Batch estimation**. Data from IMU are collected to perform the batch estimation of the augmented state. According to literature, 500 measurements are sufficient to retrieve a proper estimation [26, 53]. In this phase the Savitzky-Golay filter cleans angular rates from noise. Once this phase is concluded, a preliminary estimation including both the inertia parameters and the offset is available.

8. **Batch estimation + Kalman filtering.** Once an initial estimation of inertia properties is at hand, the procedure advances into a loop consisting of batch estimation coupled with the filtering of angular measurements through a 6-state Kalman filter. This batch estimation, taking the filtered angular rates as input, periodically (e.g., every 20 seconds) yields a refined inertia estimation for the Kalman filter. The logic underpinning this approach is twofold. Firstly, as proved in Chapter 4, batch estimation, while robust against unmodeled effects, typically does not deliver exceptionally precise results. Conversely, the Kalman filter, though capable of attaining high accuracy levels, exhibits considerable sensitivity to unmodeled effects and any inaccuracies in the inertia information. The process continues iteratively until a stable value of $\varepsilon_B$ is reached, which is determined by the normalized difference between two consecutive estimations. Once this settled threshold is achieved, indicating minimal variation between estimations, the procedure concludes.
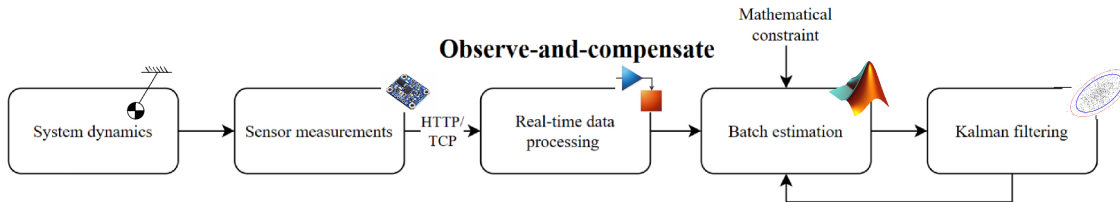


Figure 5.3: Observe-and-compensate scheme.

9. **MMUs adjustment**. After the estimation, vertical MMUs are shifted consistently with the vertical offset estimation. Conversely, horizontal MMUs are positioned in the final position of phase 4, that is, re-compensating the planar offset.

10. **Performance assessment**. The residual offset $||\delta\mathbf{r}||$ is finally estimated through direct (e.g., computing the derivative of the angular momentum) or indirect techniques (e.g., measuring the period of the pendulum, kinetic energy oscillations).

The main advantage of this approach lies in its ability to inherently generate a back-up estimation, despite the intrinsic complexity of the system and the inability to evaluate convergence properties a priori. Specifically, if unmodeled effects lead to divergence during phase 8, the preliminary batch estimation from phase 7 stands ready as an alternative, directly applicable for adjusting the MMUs' positions. This redundancy ensures that, even when faced with unforeseen modeling discrepancies or analytical challenges, the procedure retains the capacity to proceed with corrective actions based on the most reliable data.

## 5.1.2. Maintaining experimental consistency amidst procedural condensation

In the practical execution of the experiment, a strategic condensation can be introduced to mitigate potential risks associated with the re-initialization of mass positions post the initial balancing phase. The foundational basis of such a condensation is the high similarity of phase 3 in the planar balancing and phase 6 in the observe-and-compensate. This approach, besides the reduced operational efficacy, since two identical phases are performed separately, also lets the experiment vulnerable to an untenable risk. Indeed, in the procedure presented in Section 5.1, inducing a non-zero offset, required to initialize the observe-and-compensate step, necessitates shifting the masses from their equilibrium positions. Such an operation is characterized by the risk of not being able to return them to the exact prior configuration, a critical concern in sensitive procedures as for balancing. Let consider the initialization of the observe-and-compensate step, involving, for instance, moving one MMU 60000 steps toward its positive direction. In theory, reversing this movement by 60000 steps should reestablish the equilibrium. However, conjecturing that 50 steps are missed during this process, an error approximately on the order of $10^{-5}$ m would be introduced in the planar balancing. Such an error significantly exceeds the residual observed in the simulations presented in Chapter 4, proving the necessity of managing the associated risk. In summary, the risk can be resumed as the impossibility to ascertain the MMUs exact adherence to the command.

To circumvent this issue, the experimental framework can be restructured to obviate the need for re-initialization, while preserving conceptual integrity. The essential idea of this procedural condensation lays in designing an experiment where a unique initialization stands for both the subsequent steps. Therefore, once the initial conditions are imposed moving the MMUs, prior to any control activation, a 30 seconds window of dynamics

observation is performed. In absence of the mathematical constraint $B\mathbf{x} = \mathbf{c}$, the estimation cannot be directly performed, it is instead let to post-processing. Hence, the control is activated paving the way for the observe-and-compensate step. Essentially, while the conceptual flux diagram remains unaltered, the system dynamics observation of the observe-and-compensate step is anticipated to the planar balancing. This adaptation ensures that a univocal enforcement of the initial condition is pertinent to both the steps, thereby enhancing procedural efficiency without compromising the experiment's foundational principles. This paralleling strategy is illustrated in Figure 5.4.
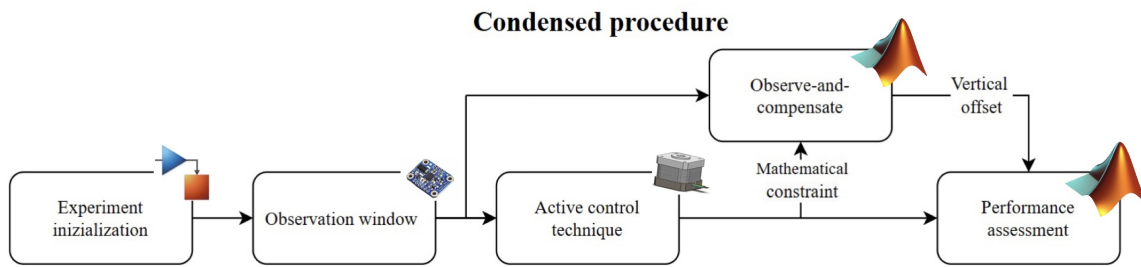


Figure 5.4: Experiment breakdown.

This modification allows to better fulfill the experiment requirements, including the fast repeatability of the process, enhanced by the procedural condensation.

## 5.2. Hardware interface

Establishing real-time communication with both sensors and shifting masses is facilitated through the Simulink software. This approach offers significant advantages, including the possibility to aggregate sensor readings into an array for subsequent analysis, as well as enabling real-time control of mass adjustments. However, operating a real-time experimental compensation in Simulink introduces a higher procedural intricacy, increasing the susceptibility to errors and data loss.

Furthermore, managing communication within this software presents specific challenges, primarily because numerous functions in Simulink are not compatible with code generation. This limitation necessitates their replacement with iterative algorithmic loops that can be more time-consuming and less efficient (e.g., Algorithm 5.1).

In summary, while the real-time data acquisition and control introduces enhanced compensation capabilities, a meticulous oversight to mitigate the risks of data fidelity compromise and capability to navigate the constraints imposed by the Simulink environment's code generation capabilities are required.

## 5.2.1.  Sensor reading

Communication with attitude sensors and IMU is established in real-time via the TCP and HTTP communication protocols. A distinctive feature of these protocols is the request-response model, meaning that sensors transmit data only upon receipt of explicit requests. To facilitate this communication within the system architecture, two primary TCP blocks are employed:

- A TCP Send block, tasked with issuing requests.

- A TCP Receive block, responsible for acquiring the corresponding responses.

The utilization of TCP blocks also for the HTTP communication is necessitated by the absence of dedicated HTTP communication blocks within the Simulink environment. Despite this, the TCP blocks suffice in managing the data exchange process, ensuring the system's capability to perform real-time data retrieval and processing tasks essential for operational efficacy.

The comprehensive subsystem shall also accomplish a conversion of the received message into the desired format, specifically a double-precision vector, a paramount aspect in the command generation chain. Specifically, in the communication with ground truth system, the server's response for attitude data appears as follows:

```
HTTP/1.1 200 OK
Access-Control-Allow-Origin: *
Content-Length: [PayloadSize]
Server: Crow/master
Date: [Time]


[Quaternion]    [Residual]    [ValidityFlag]
```

Where `[PayloadSize]` is the size of the data payload, `[Time]` is the timestamp of the response, `[Quaternion]` represents the quaternion array, `[Residual]` denotes the residual value, and `[ValidityFlag]` indicates the integrity of the data received. The response is prefaced with a char message containing metadata, followed by the effective payload in the `uint8` data type. Therefore, data parsing and processing operations are required to extract and convert this information into a usable format. Algorithm 5.1 details the proposed structure for handling data. Though it appears as a unified procedure, its functions are executed separately within the Simulink environment, according to the following structure:

- Data transmission initiation → TCP Send block.

- TCP Receive block → Data reception and processing.

---

**Algorithm 5.1** Procedure for Attitude Orientation Data Acquisition in Real-Time

---

**Data transmission initiation**:
**for** each predetermined temporal interval **do**
    Create an HTTP 'GET /raw' request
    Construct a formatted string utilizing the `sprintf` function
    Encode the formatted string into `uint8` binary format
**end for**
**Data reception and processing**:
Isolate the terminal line of the data stream
**for** each element within the isolated data array **do**
    Employ the `typecast` function to transmute data into the target type
**end for**

---

As prefigured, the algorithmic complexity escalates within the Simulink environment, due to incompatibility of the `webread` MATLAB function with the code generation capabilities of the software. In fact, `webread` would inherently simplify the process by obviating the necessity for specific request generation, explicit string segmentation and data transmutation.

Conversely, the IMU is programmed to interface through the TCP protocol. While the block configuration remains analogous, featuring both a TCP Send and a TCP Receive as for the previous case, the request-response mechanism implementation, diverges slightly. Specifically, responses are not utf-encoded char messages but arrays of bytes directly, encompassing a complete set of different outputs. Accordingly, the viable requests command structures are the following:

```
GET /quaternion
GET /gyro
GET /acc
GET /grav
GET /all
```

The request depends on the specific data one aims to extract, whether it is solely the quaternion, the angular rates, the linear acceleration, the gravity vector, or an all-encompassing unique 26-bytes array that includes all these elements, with each component comprising 2 bytes. This array structure hints that response handling is tasked with dissecting the array components, converting them into 16-bit integer values, and scaling them based on the Least Significant Bit (LSB) value documented in the datasheet for each measurement. Algorithm 5.2 reports the structure of the request-response mechanism. An

---

**Algorithm 5.2** Real-time IMU Data Acquisition

---

**Data transmission initiation**:
**for** each predefined time instant **do**
    Initiate the request using 'GET/all'
    Construct a structured string with `sprintf`
    Convert the structured string to `uint8` format
**end for**
**Data reception and processing**:
Separate bytes corresponding to each vector
**for** each component of the extracted vector **do**
    Rearrange assuming little-endian order using `bitshift` and `bitor`
    **if** most significant bit of rearranged bytes is 1 **then**
        Apply bitwise NOT operation using `bitxor`
        Add 1 to result
        Convert to negative int16 format
    **else**
        Convert to int16 format
    **end if**
**end for**
Adjust according to the LSB value

---

inner function is developed to manage negative outputs according the two-complement logic: the most significant bit is detected and the two-complement inversion of the bit array is performed to enable proper reading. These operations lead to separate information of both sign and value of the reading, enabling seamless detection.

## 5.2.2.  Control command generation

The stepper motor controller is configured to accept commands in JavaScript Object Notation (JSON) format, a data interchange standard derived from the JavaScript programming language [74, 75]. An example of such a command is shown hereafter:

```
{
    "type": "move",
    "data": {
        "steps": 16000,
        "delay": 120,
        "microsteps": 16,
        "dir": 1
    }
}
```

This command essentially constitutes a structured entity specifying the command type, accompanied by additional pertinent details. Specifically, it includes the number of steps the motor is to execute, the interval between steps measured in microseconds, the quantity of micro-steps encompassed within each individual step, and the intended direction of motion. The latter is represented in binary form, with possible values being 0 and 1, respectively indicating reverse and forward motion.

The initial critical observation regarding this input is that it clearly processes a specified number of steps to be executed within a certain time interval, that is a $\Delta$steps in a $\Delta$t. This necessitates the primary operation of ensuring compatibility with the control command, which is, conversely, position-oriented. Therefore, the MMU position at each step is subtracted by the precedent position, and the difference is fed as command. This method, if the $\Delta$t (i.e., the step delay) is priory set, facilitates the conversion of a position-based command to a dual information embedding the required steps and duration, aligning with the stepper motor controller's operational parameters.

Upon the successful generation of the appropriate mass shift, its magnitude is channeled into the command generation algorithm, dictating the number of steps to be executed. Concurrently, the mass shift's sign is processed to yield binary information, indicative of the direction of motion. The real-time command generation is executed iteratively, adhering to the subsequent procedural outline:

---

**Algorithm 5.3** Generation of Real-Time Command

---

Employ `int32` for the formulation of an integer-based command.
Utilize `sprintf` to instantiate a temporally-variable JSON command while maintaining a fixed dimension.
Invoke `blanks` to establish a consistently dimensioned array composed of spaces.
Apply `uint8` for the transformation of the constructed command into a byte array.

---

The process encompasses the conversion of command specifications into an integer format, the dynamic generation of JSON command strings with uniform dimensions, and the conversion of the command into a byte array, optimizing it for transmission.

The generation of the control command for the batch procedure stands distinct, intrinsically requiring the system's center of mass displacement, $\Delta\mathbf{r}$. Specifically, executing a mass command mandates the utilization of the following relationship:

$$\Delta R_i = \frac{m}{m_i \Delta r_i} \ i \in \{x, y, z\} \tag{5.4}$$

Provided the information of $\Delta\mathbf{r}$, the steps to perform are unequivocally defined. However, this ostensibly straightforward concept conceals multiple underlying uncertainties. Equa-

tion (5.4) necessitates an accurate knowledge of both the MMU and the platform's mass attributes. Further processing is required when considering that a single step equates to an angular displacement of 1.8°, and a complete revolution corresponds to a linear displacement of 0.1 mm, contingent upon the screw thread's specifications, which are potential sources of error as well. In summary, the vertical balancing procedure requires meticulous knowledge of the system's physical properties and an adept handling of the associated operative uncertainties.

## 5.3.   Performance assessment

In estimating the residual offset, it is crucial to acknowledge the complexity of evaluating a two-step procedure. Specifically, challenges may arise in attempting to gauge the 2 steps performance separately. In fact, the techniques proposed in literature and detailed in Section 2.4, may be impractical to evaluate distinctly the residuals $\delta r_x, \delta r_y$ and $\delta r_z$. The LSQ fitting of Euler equations, while providing separate evaluation, demonstrated in Chapter 4 its limited accuracy; and the method employing the period of the pendulum offers an evaluation on the total residual magnitude $||\delta \mathbf{r}||$.

However, a comprehensive estimation of the residual offset $||\delta \mathbf{r}||$, without segregating the performance assessment for planar and vertical balancing, is inadmissible for the procedure. Indeed, it poses a significant risks, potentially highlighting the weaker procedure and thereby undermining the overall experiment's validity, even if one stage is conducted proficiently. For instance, if the achieved residuals are comparable to the precision levels observed in simulations detailed in Chapter 4:

$$
\begin{aligned}
\delta r_x, \delta r_y &\sim 10^{-7} \text{ m,} \\
\delta r_z &\sim 10^{-4} \text{ m,}
\end{aligned}
\tag{5.5}
$$

the total residual necessarily becomes:

$$
||\delta \mathbf{r}|| = \sqrt{\delta r_x^2 + \delta r_y^2 + \delta r_z^2} \sim 10^{-4} \text{ m.}
\tag{5.6}
$$

In this case, the notably larger residual in the vertical component $\delta r_z$ effectively overshadows the proficient planar compensation. Furthermore, separate performance evaluation is more suitable in dealing with challenges presented by real systems, such as the potential non-exact orthogonality of the MMU axes. Specifically, any significant uncertainty in MMUs position and screw direction might jeopardize the results of the planar balancing, as the in-plane residual might increase following the vertical compensation.

Given these intricacies, the performance assessment plan necessitates a methodology that is both simple, in terms of being easily comprehensible and implementable, and distinct, in its capacity to independently evaluate the unique aspects of each step while considering their inter-dependencies.

The challenge lies in the scaling problem, which renders the segregation of evaluation metrics particularly complex. Specifically, it is not possible to establish a planar offset residual, without disposing of the vertical offset information. For instance, relying on a static methodology to infer the residual planar offset, the problem becomes indeterminate. This is because there exists an infinite array of solutions corresponding to different vertical offsets, all of which can result in the same platform inclination $\alpha$. Indeed, with the unique information of $\alpha$, the method is not able to distinguish the CM from any other point lying on the local vertical direction (the gray line in Figure 5.5)
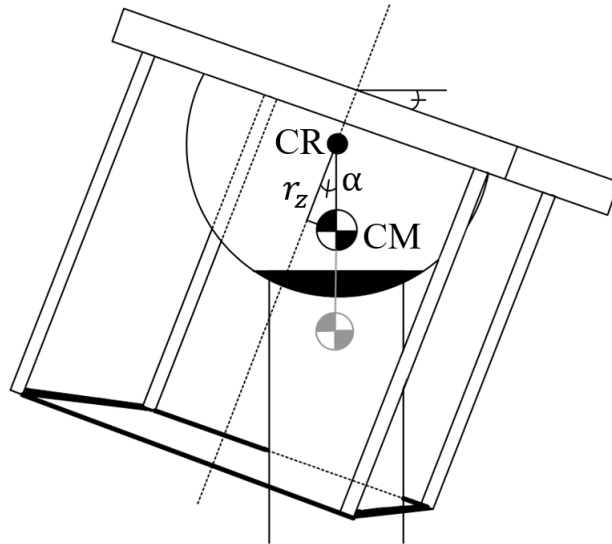


Figure 5.5: Scaling problem effect on performance assessment.

Therefore, the proposed performance evaluation strategy focuses on utilizing the offset computed in the batch estimation phase to address the scaling problem in the static methodology. This balanced approach ensures a comprehensive evaluation of the planar balancing without unjustly discrediting the experimental accuracy and success. Hence, to obtain $\delta r_x$, $\delta r_y$ the residual planar offsets,

$$\delta r_x = r_z \tan\theta, \tag{5.7}$$

$$\delta r_y = r_z \tan\phi, \tag{5.8}$$

where $\phi, \theta$ are directly retrieved from the last readings of the attitude sensors in the experiment. The procedure is non-invasive, eliminating the necessity to artificially induce motion within the satellite simulator. While it is acknowledged that the process is susceptible to error propagation via $r_z$, the performance assessment does not demand exceedingly high precision levels. Instead, the focus is on capturing the order of magnitude, which suffices for the intended evaluative objectives. This approach maintains the integrity of the system while providing a reasonable approximation of performance metrics, aligning with the overarching experimental criteria.

Ultimately, an indirect methodology is utilized for computing the magnitude of the whole offset vector. The chosen strategy utilizes the period of the 3D pendulum to extract the information on the magnitude of the residual $\delta\mathbf{r}$.

$$T_p = 2\pi \sqrt{\frac{J_s}{mg||\delta\mathbf{r}||}}. \tag{5.9}$$

Hence, the method requires to enforce the motion externally and acquire the platform angular velocities during the oscillation. Equation (5.9) is therefore inverted to retrieve the residual norm. Finally, the acquired data pertaining to the planar offset facilitates the evaluation of the vertical compensation's efficacy inverting the relationship

$$||\delta\mathbf{r}|| = \sqrt{\delta r_x^2 + \delta r_y^2 + \delta r_z^2}, \tag{5.10}$$

therefore culminating in an individual steps performance assessment through the separate knowledge of $\delta r_x$, $\delta r_y$ and $\delta r_z$. The discussed strategy is favored over the application of LSQ fitting to the Euler equations of rigid body motion. Evident from the results attained in Chapter 4, while least squares fitting is theoretically capable of delineating the residual components along x,y and z axes distinctly, its practical application for extremely precise estimations is constrained by limited accuracy levels, $\sim 10^{-4}$ m. Indeed, as anticipated in Section 4.4, the residual for planar balancing is expected to be in the order of $10^{-6}$ or $10^{-7}$ m. Consequently, adopting LSQ fitting for performance evaluation would introduce an error magnitude that surpasses the tolerable bounds of the planar balancing expected error, rendering it an unsuitable method for an accurate performance assessment. Conversely, the segregated assessment strategy, as outlined in Table 5.3, demonstrates a increased sensitivity to detect even microscopic residuals. Specifically, with the incorporation of a ground truth system error of 15 arcseconds, and conjecturing, with reference to CAD models and simulation results, a maximum vertical offset of 5 cm, the residual can be discerned with a precision in performance evaluation reaching $3.635 \cdot 10^{-6}$ m. This precision underscores the efficacy of the proposed method in accurately identifying and quantifying

offsets that would be otherwise imperceptible with less sensitive evaluation techniques.

Table 5.3: Comparison of unique and segregated performance assessment.

| | Integrated | Separated |
|---|---|---|
| **Objective** | Evaluates overall system performance holistically. | Evaluates planar and vertical balancing steps separately. |
| **Features** | Straightforward but can mask the performance of individual steps due to the interdependency of vertical and planar offsets. | Addresses the complexity by assessing each step based on its unique parameters and challenges. |
| **Challenges** | Difficulty in isolating the impact of each balancing step on the overall performance. | More equipped to reject real system sources of error (e.g., non-orthogonality of axes) but requires addressing the partial observability of the 3D pendulum. |
| **Evaluation Method** | *Qualitatively*, observing the reduced kinetic energy oscillations; or *quantitatively*, LSQ fitting of EOM or measuring the period of the 3D pendulum to retrieve $||\delta\mathbf{r}||$. | *Quantitatively*, exploiting sensor measurements for direct estimation of $\delta r_x$ and $\delta r_y$, followed by $\delta r_z$ estimation from separate knowledge of $||\delta\mathbf{r}||, \delta r_x, \delta r_y$. |
| **Mathematical Approach** | $||\delta\mathbf{r}|| = \frac{4\pi^2 J_s}{mgT_p^2}$. | $\delta r_x = r_z \tan\theta,$ $\delta r_y = r_z \tan\phi,$ $||\delta\mathbf{r}|| = \frac{4\pi^2 J_s}{mgT_p^2},$ $\delta r_z^2 = ||\delta\mathbf{r}||^2 - \delta r_x^2 - \delta r_y^2.$ |
| **Error Propagation** | Uncertainties on inertia parameters propagate on the final result. | More subjected to error propagation due to the reliance on the batch estimation. |
| **Outcome** | Provides a holistic overview but with less actionable insight for stepwise improvement. | Yields detailed, actionable insights for each step, enhancing the potential for precise adjustments. |

## 5.4. Requirements fulfillment

Following the comprehensive planning of the experiment, it is imperative to verify the fulfillment of requirements to evaluate the rigor and validity of the presented research.

Table 5.4: High-level compensation requirements fulfillment.

| ID | Type | Text | Fulfillment |
|---|---|---|---|
| R001 | O | The algorithm shall be flexible with respect to changes in satellite mock-up. | The approach is agnostic to any particular configuration of STASIS and does not mandate specific constraints on its setup. |
| R002 | F | The algorithm shall be capable of accomplish balancing without using RWs. | Strategies involving the utilization of RWs have been excluded from consideration. |
| R003 | F | The algorithm shall avoid error propagation of preliminary estimations. | The strategy does not require prior information on the CR-CM offset or the inertia parameters. |
| R004 | F | The algorithm shall operate with the available ADCS hardware precision. | Through simulations, various strategies were evaluated, confirming those approaches that exhibited compensation capabilities even in the presence of noise and aerodynamic drag. |
| R005 | F | The algorithm shall be robust against measurements noise. | The gains for active control techniques have been tailored to prioritize reliance on attitude knowledge over angular rates. Moreover, the LSQ methodology is enhanced through Savitzky-Golay and Kalman filters. |
| R006 | O | The compensation shall guarantee brevity in time. | Both the Kalman filters and LSQ methods have demonstrated effectiveness within an observation window of 1 minute. Active control techniques achieve convergence within a span of 10 minutes. |

Table 5.4 – *Continued from previous page*

| ID | Type | Text | Fulfillment |
|------|------|------|-------------|
| R007 | O | The compensation shall exploit the symmetrical configuration of the MMU. | The algorithm leverages the entirety of the MMUs system. Moreover, the division into two steps is argued to be more efficient when the MMUs system exhibits symmetry. |
| R008 | O | The compensation strategy should guarantee fast repeatability. | Beyond ensuring the experiment's efficiency, its condensation in a unique step also promotes a shorter restart time. |
| R009 | V | The performance assessment shall not alter the physical arrangement of the system. | Neither the evaluation of the planar compensation nor the vertical's necessitate shifting the MMUs from their position. |

Following the detailed planning of the experiment and extensive simulations, there remains one requirement that poses a challenge for unambiguous fulfillment assessment, R006. Initially, the requirement was conceptualized to favor a procedure that could compensate the platform before disturbances led to divergence or significant errors. However, in the absence of studies on aerodynamic coefficients for STASIS, the "brevity in time" criterion remains qualitative and cannot be quantitatively assessed. This evaluation forms a focal point for future development endeavors. The experience accrued during the thesis development suggests that a duration of 20 minutes is reasonable. Consequently, the finalized strategy is tailored to balance the platform within 10 minutes during simulations. Nevertheless, an essential consideration emerges at this juncture. While it might be theoretically feasible to adopt more aggressive gain settings, conservative gains were chosen. This decision aims to enhance the probability of achieving convergence, given the intricacies inherent to real-world experiments, and importantly, to prevent command throttling during wireless control operations.

# 6 | Hardware-in-the-loop experiment results

Upon several months of dedicated research, the automatic balancing experiment of STASIS is approached. The outcome, whether positive or negative, holds significant implications for the EXTREMA project and will dictate the subsequent course of action. Given this importance, detailed performance metrics and analysis of the planar offset residual's magnitude is provided. It is crucial to note that due to specific time and power constraints, the opportunity to repeat this experiment is limited. The time constraint was primarily due to a delay caused by the commercial unavailability of motor controllers, which significantly postponed the commencement of the experiment. Power limitations were also a factor, as the intended power-beaming system for continuous energy supply to the platform had not been installed yet. This absence necessitated reliance on battery power, which restricted the number of possible experiment runs due to the extended recharge periods. Given these intricacies, meticulous attention to detail and precision during the procedure's executions was paramount.

## 6.1. Experiment setup

During the experimental phase, the MMUs system was undergoing a prototypical transition, following a substantial upgrade of the platform. The upgraded MMUs system was a response to issues noted in the previous version, including high power consumption and overheating problems. Consequently, the use of newly integrated, yet untested components, was anticipated to introduce unexpected behaviors in the experiment.

Furthermore, the platform exhibited recognized areas for improvement, including the flexibility of its lower plate and the presence of intermittent gaps in the power rails. These gaps were known to cause sporadic disconnections of the masses from the power line.

A minor constraint was the unavailability of the whole set of MMUs. Specifically, as depicted in Figure 6.1, on the day of the experiment, STASIS was equipped with only two out of the eight MMUs that were originally planned in the design. This configuration

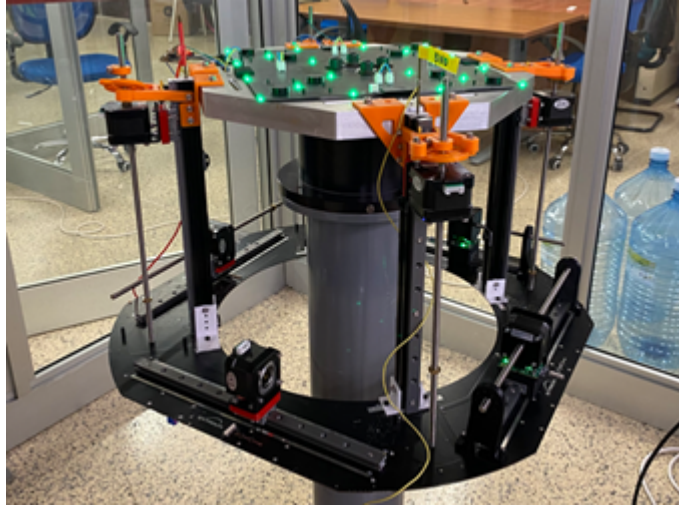negated the capability to achieve three-dimensional compensation.



Figure 6.1: STASIS at the time of the experiment, wherein the light-emitting masses are the active ones.

In light of the limited number of MMUs available during the experiment, the focus was exclusively on planar balancing. Venturing into vertical compensation would have been impractical without the requisite masses for vertical movement, negating the possibility to evaluate the vertical offset residual. Moreover, given the high operational complexity and error susceptibility of the active control phase, priority was given to laying a robust foundation for its future implementation. This approach was chosen over conducting a less rigorous, complete experiment. Concentrating on the most complex aspect allowed for a more efficient and detailed exploration, setting the stage for further improvements and refinement of initial results. Consequently, while the results might not exhibit extreme accuracy, the primary objective is to demonstrate the viability of the proposed procedure for STASIS. This focus is critical in establishing a solid foundation for the subsequent development of the platform's balancing.

## 6.2.   PID balancing results

Upon proper calibration of the ground truth system, performed through the acquisition of 5000 images in different attitude configurations, and observed the convergence of the reconstruction algorithm, the PID control was tested on STASIS. The control methodology is configured to operate at two distinct frequencies, stemming from the varying operational frequencies of the hardware components responsible for attitude knowledge and those for attitude control. Initial investigations have determined that a frequency of

10 Hz is optimal for attitude reconstruction, effectively circumventing request throttling. On the other hand, preliminary tests indicated that a lower frequency of 1 Hz is preferable for actuation, ensuring that each MMU command is executed within the constraints of command latency, maintaining smooth operational flow.

Additionally, post preliminary actuator tests, a further measure to mitigate the risk of command throttling was adopted. The control strategy artificially saturates the command, bounding the actuator steps at a limit of 500 steps/s. This approach offers significant advantages in active control systems, which are often marked by initial high-intensity actuation that subsequently tapers off. Such moderation of actuator responses is particularly advantageous in scenarios where precise pointing accuracy is critical, even sacrificing the settling time, as in the case of platform balancing.

The experiment sustained operation for an approximate duration of 650 seconds, as visible from Figure 6.2. At this point, the depletion of the batteries resulted in the shutdown of the entire CubeSat simulator[1].



(a) Angular rates.

(b) Steps command.
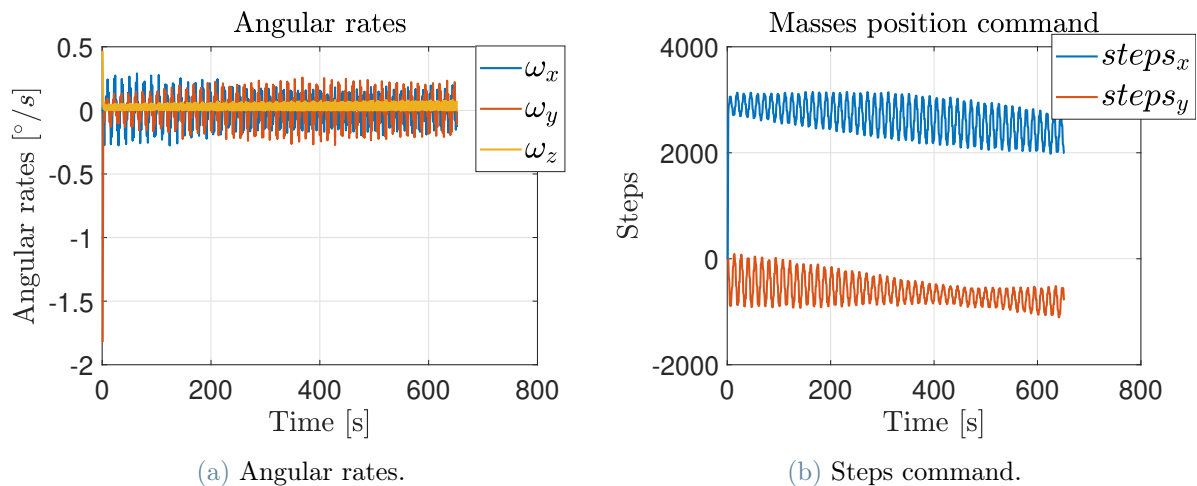
Figure 6.2: Results from the first PID control experiment.

Analysis of the Euler angles trajectory, as shown in Figure 6.3, indicates that the control parameters were appropriately selected, evidenced by the system's gradual progression towards a state with negligible roll and pitch angles. However, due to the experiment's truncated execution, full convergence was not realized, implying the necessity for a revision to select higher gains.

---

[1]It is important to note that the batteries' capacity exceeds 650 seconds under normal conditions. However, for this experiment, they were already in a state of less-than-optimal charge.
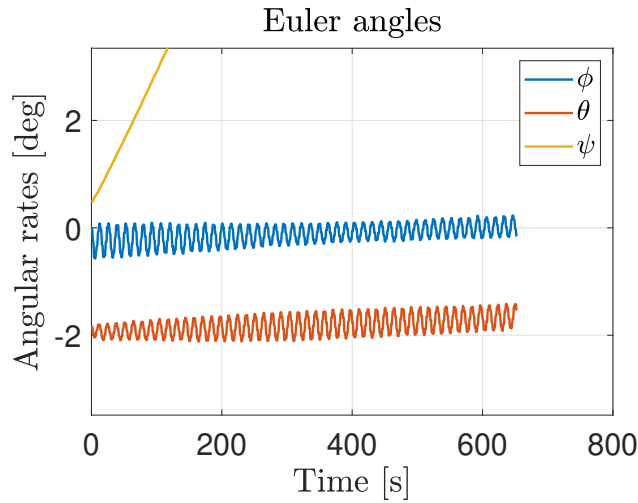
Figure 6.3: Euler angles history during the first PID actuation.

The initial experimental setback proved to be a valuable learning experience. Consequently, the PID control experiment was reconducted with increased gain settings, preceded by a thorough verification of the batteries status. In this second iteration, however, the wireless connection to the ground truth server was lost after approximately 820 seconds, as evident in Figure 6.4. The cause of this disruption remains undetermined, though likely attributable to generic connection issues, potentially leading to network congestion.



(a) Angular rates.



(b) Steps command.
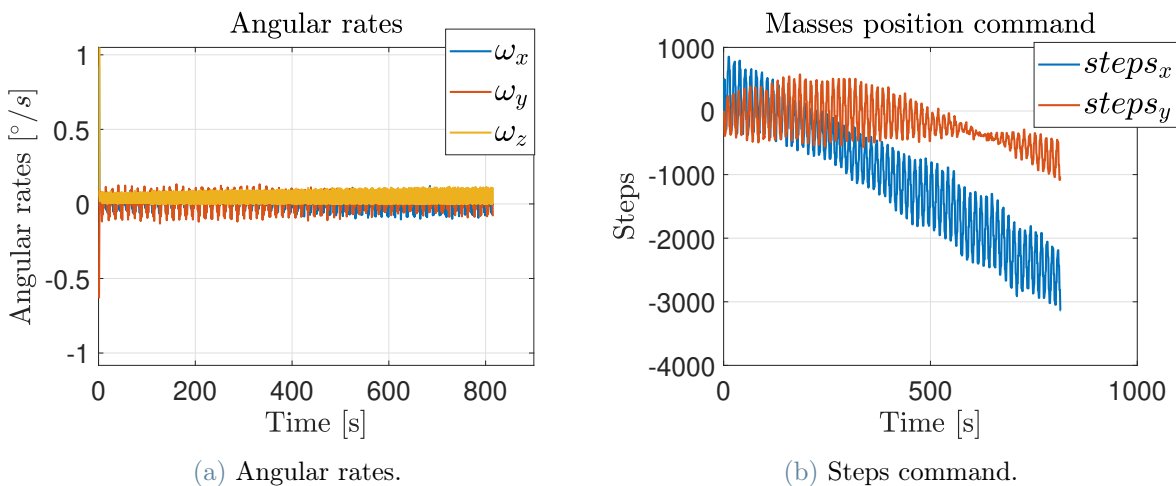
Figure 6.4: Data from the repeated PID control trial.

Despite the interruption, the data acquired during this period were encouraging. The Euler angles residuals, as shown in Figure 6.5, were considerably reduced, indicating a successful application of the control within the given operational time-frame.
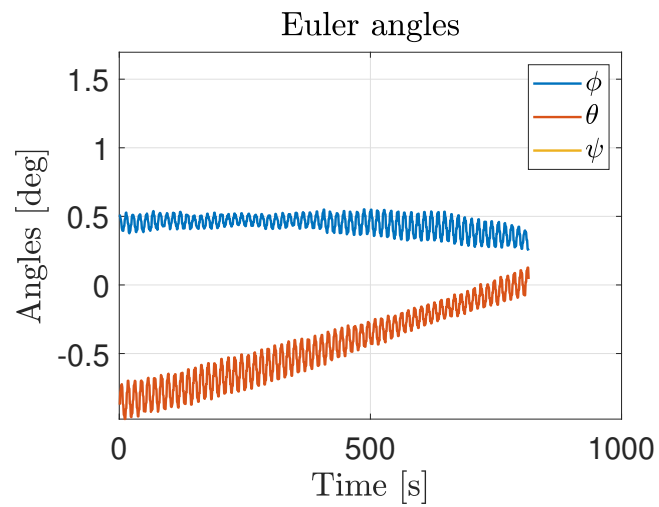
Figure 6.5: Euler angles history during the second PID actuation.

Figure 6.6 provides a visual demonstration of the procedure's effectiveness. It shows that the platform achieves effective planar balance, as indicated by the near-perfect horizontal alignment of its upper face.
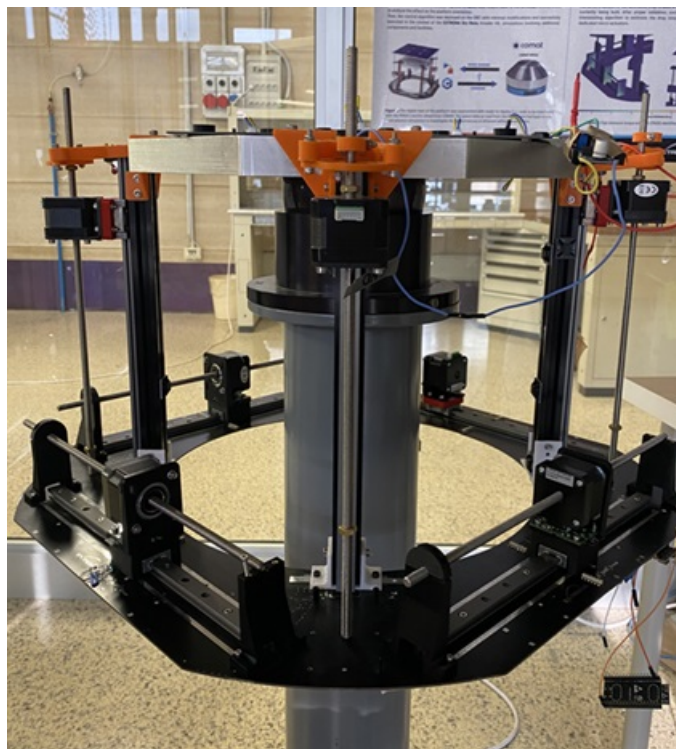


Figure 6.6: STASIS, concluded the second PID actuation.

## 6.3.  Non linear balancing results

Following the experimental testing of PID, the facility underwent testing with the non-linear $\hat{\mathbf{g}}$ control method, aiming to achieve improved outcomes. Unfortunately, in the course of the attitude control phase, the pitch-oriented MMU encountered a malfunction, remaining immobilized on its thread. This issue originated from a damage on one out of the four wheels whereon the MMU is mounted. Consequently, the actuator became jammed, causing continuous transmission of a single command without achieving convergence to a definitive value. Hence, the experiment was abruptly interrupted as presented in Figure 6.7. Despite the challenges, the outcomes are deemed positive in the terms of future developments. The MMU that functioned properly demonstrated a clear convergence to a specific position, underscoring the validity of the applied methodology. This inference is further supported by the final position of the MMU, aligning closely with the mid-point of the screw, as anticipated from the digital twin analysis.
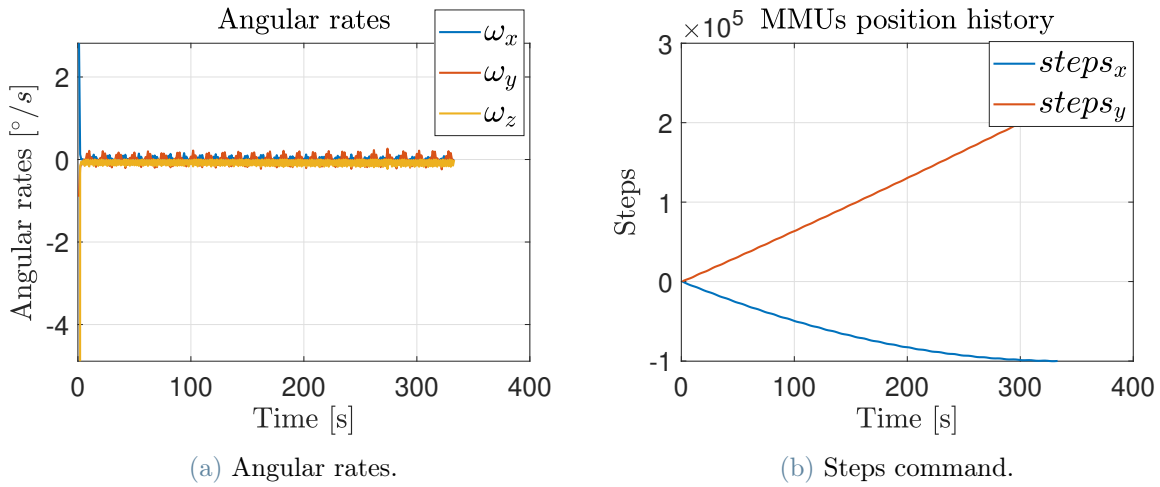


(a) Angular rates.                          (b) Steps command.

Figure 6.7: Non-linear $\hat{\mathbf{g}}$ control experimental results.

## 6.4.  Performance assessment

The absence of the experiment's second step precludes an unbiased evaluation of the methodology's performances. Nonetheless, as the aim of this phase is primarily to ascertain the residuals order of magnitude, utilizing CAD data is deemed a justifiable approximation. In the current MMUs configuration, the digital twin indicates a vertical offset of $r_z = 2.55$ cm, placing $r_z$ order of magnitude at $10^{-2}$ m. Consequently, with the known residuals of the Euler angles, a rough estimate of the CR-CM offset can be achieved.

Table 6.1: Static estimation of residuals.

|  | Euler Angles [rad] | Residuals [m] |
|---|---|---|
| First PID | $\phi = -0.0006$ | $r_x \sim 10^{-4}$ |
|  | $\theta = -0.0286$ | $r_y \sim 10^{-6}$ |
| Second PID | $\phi = 0.0056$ | $r_x \sim 10^{-6}$ |
|  | $\theta = 0.0005$ | $r_y \sim 10^{-5}$ |

To evaluate the effectiveness of the procedure, these residuals shall be evaluated with respect to a reference framework, providing a quantitative comparison base to properly gauge them. Hence, three distinct approaches have been conceptualized:

- A methodology based on control torque, involving a comparison with the torque generated by the actuators that STASIS is planned to employ specifically to counterbalance the residual offset. This approach, prior to any assessment on the experiment efficacy, shall include specific considerations on actuators' saturation.

- An approach founded on a comparison with the aerodynamic drag effect, aimed at measuring the residual relative to the predicted drag disturbance torque at a specific angular velocity.

- An examination of MMUs precision, assessing the minimal CM shifting provided by the stepper-motor driven masses, to determine the control execution performances.

The first two approaches are not suitable for an evolving platform such as STASIS, as they necessitate comprehensive system knowledge. Presently, the final actuators for the residual offset compensation have not been determined, and accurate drag coefficient estimates are unavailable. Conversely, the third approach aligns with the currently available data. The maximum precision of the system actuators can be effectively evaluated by considering their minimal impact on the overall CR-CM offset. Given the mass ratio between MMU and platform, and the smallest possible shift for an MMU (equivalent to one step), the actuators precision can be computed as:

$$\Delta r_{\min} = \frac{1}{2}\frac{m_i}{m}\Delta R_{\min} \approx 1.33 \cdot 10^{-7} \text{ m},$$ (6.1)

which is two orders of magnitude below to what has been achieved, suggesting that additional enhancements are necessary to fully leverage the actuators' high precision.

In summary, despite being aware of the numerous complexities associated with real-world experimentation, it is essential to complement the commendable design efforts in the

balancing hardware of STASIS with a more proficient algorithm. Therefore, further development of the closed-loop strategy, potentially incorporating the non-linear approach, is strongly advised.

## 6.5.   Lessons learnt

The control experiments conducted as part of the EXTREMA project has yielded significant findings. Besides the identification of key operational parameters such as working frequencies and control saturation, it has also laid a foundational framework for future enhancements. Specifically, non-invasive modifications to STASIS are advisable to rectify the observed structural anomalies. Notably, during various experimental runs and tests, the MMUs frequently disengaged around the mid-point of the screw. Moreover, disconnections were observed at points other than the known structural singularity of STASIS, suggesting that the screws supporting the MMUs may be experiencing mechanical bending. Evident from Figure 6.8, this bending might generate a vertical force on the actuators, leading to their loss of electrical connectivity with the linear guide.
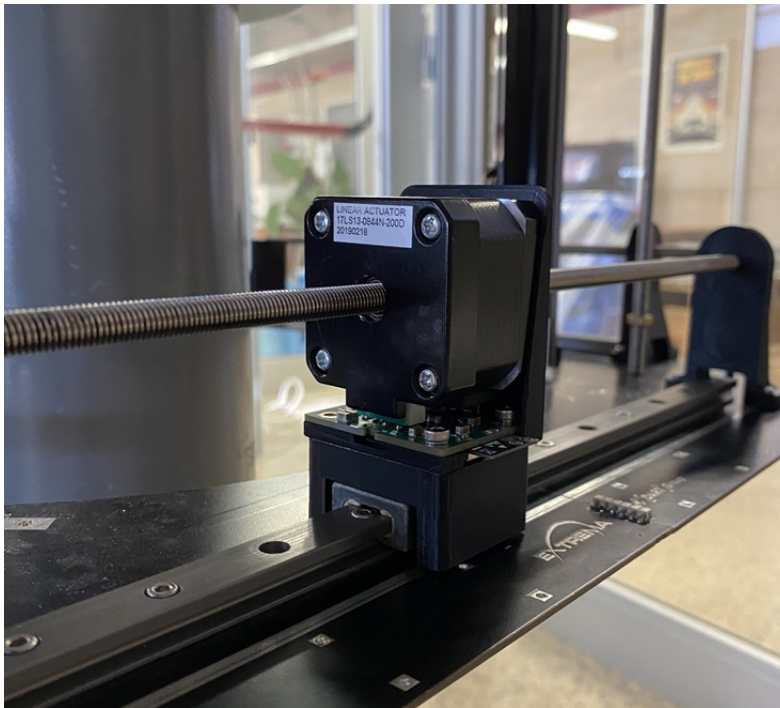


Figure 6.8: Real STASIS movable mass unit.

Furthermore, the experimental phase, employing diverse communication protocols simultaneously, highlighted the risk of overloading the local router. Notably, during real-time experiments, the host server was observed to reject connections after approximately 10

minutes of actuation. This duration is acceptable for a balancing procedure, but might cause a significant slowdown for the EXTREMA project, which intrinsically requires the execution of simulation that may span several days. Finally, as underscored in [19] and evident from the first experimental PID execution, enhancing the FlatSat with an energetic autonomy constitutes a significant advancement. This advancement is pivotal for enabling rapid experiment repetition, which is instrumental in expedited parameter adjustment and debugging processes.

# 7 | Conclusions

This thesis delves into the simulation, design, and testing of an automatic balancing protocol for a nano-satellite dimension simulator. The intrinsic complexity of the project necessitated scrupulous planning to guarantee consistent balancing process, rendering the actual experimental execution a comparatively minor component of this project. The introduction of unforeseen variables into the simulation markedly constricts the range of viable strategies, underscoring the procedure's susceptibility to errors. Specifically, given the simulator's reduced dimensions, the potential for disturbances such as structural sagging and vibrations, may significantly skew the final outcomes.

Acknowledging these complexities, even the simulation results are not exceptionally accurate, exhibiting a residual error on the scale of $10^{-4}$ m. This accuracy, however, could potentially be enhanced through the application of a state observer. Despite these challenges, the research underscores the critical nuances and careful orchestration required in balancing algorithms for nano-satellite simulators, especially due to the pronounced impact of minor disturbances.

## 7.1. Addressing the research question

The comprehensive efforts dedicated to the STASIS digital twin and the EXTREMA Simulation Hub have facilitated a thorough response to the study's research questions, encompassing both theoretical achievements discerned from the simulations and the tangible challenges faced during experimental processes. The discussion initially focuses on resolving operative research questions, addressed through the comprehensive simulations performed on the platform digital twin. Hence, an in-depth exploration of the primary research question, which constitutes the essence of this study, is provided, focusing on the main outcomes of the experimental activities.

2. **Operative research question**: *What is the impact of the other disturbances on the balancing performances for a nano-satellite dimensions simulator?*
   Consistent simulations have unequivocally demonstrated the criticality of disturbances in the balancing process. Specifically, incorporating aerodynamic drag into

the automated balancing simulations markedly diminishes performance outcomes. This phenomenon has minor impact on batch estimation, primarily because it does not rely on specific assumption on platform dynamics. However, state observers, such as extended and unscented Kalman filters, are exposed to the risk of divergence. This is predominantly due to their intrinsic dynamic prediction phase, which can become ineffective if there is a lack of comprehensive understanding of the platform's consistent dynamic behavior. Consequently, it is advisable to employ state observers only when there is an in-depth, precise model of platform dynamics, possibly combined with RWs with very high torque capability.

3. **Operative research question**: *How attitude sensors noise can affect the balancing performance?*
An intricate and precise characterization of sensor noise has highlighted the profound influence that sensor inaccuracies exert on balancing performance. In this scenario, it is not just the observe-and-compensate methodologies suffering significantly, leading to highly skewed inertia estimations, but also certain theoretically efficient active controls, potentially tending toward divergence when these factors are included. This outcome is attributable not only to their dependence on the most noise-afflicted measurements but also to certain stages in the control implementation where noise-infused measurements aggregate, thereby culminating in compounded errors. In essence, the introduction of both dynamic disturbances and sensor errors drastically reduces the array of viable strategies that can be effectively employed.

4. **Operative research question**: *What is the best strategy to achieve an unbiased platform balancing?*
Upon integrating the primary error sources and considering the practical constraints of the procedure, the spectrum of feasible strategies significantly contracts, paving the way for the implementation of an effective approach. Simulations' analyses indicate that the optimal path for platform balancing hinges on a compensation strategy primarily anchored to the most accurate sensor available. Additionally, the selected approach shall avoid the amalgamation of noise-affected readings and steer clear of any initial assumptions regarding platform inertia or gravitational torque.
The remaining vertical offset can be proficiently ascertained using a robust batch estimation technique. The strength of this method lies in its lack of stringent presuppositions, although this same factor contributes to its less-than-ideal precision. The accuracy of this technique can be enhanced through the employment of an additional Kalman observer. However, while this tool can yield precise outcomes, it also exhibits an increased susceptibility to divergence, attributable to the multitude

of underlying assumptions.

5. **Operative research question**: *What are the advantages of having 4 planes of symmetry in the balancing hardware configuration?*

The utilization of continuous and rapidly reproducible simulations via the digital twin facilitated an accelerated understanding of the balancing problem, encompassing all its complex and technical aspects. Specifically, the benefits of employing a symmetric balancing hardware configuration were fully realized. This symmetry presents both technical and practical advantages. From a technical standpoint, the dual-actuation system allows for the use of lower control gains, thereby reducing the propensity for divergence and minimizing step commands to the stepper motors, which subsequently decreases the likelihood of request throttling. Moreover, a symmetric configuration facilitates the proximity of the actuator's center of mass to that of the platform. This proximity is crucial when segregating the procedure into planar and vertical balancing, as it significantly mitigates the influence of vertical balancing on planar compensation. By ensuring that the actuator's center of mass is closely aligned with the platform's, the process enhances the precision of each balancing action, making the overall procedure more efficient and reducing potential interference between the two phases.

On the practical side, there is an inherent advantage, especially in processes where the procurement and setup of micro-controllers are bottlenecks of the procedure. The ability to deploy multiple controllers enhances resilience against motor malfunctions, as it permits the continuation of experiments by simply installing the operational MMU on the simulator, accelerating the balancing algorithm execution and providing robustness against pragmatic setbacks.

1. **Main research question**: *To what extent an automatic balancing procedure can provide a micro-gravity environment for an air-bearing CubeSat simulator?*

Upon months of dedicated research, the primary research question of this thesis can be effectively addressed. The design of a balancing procedure for a CubeSat simulator presents substantial challenges, mainly due to escalated disturbances and uncertainties effects with decreasing dimensions of the platform. Hence, designing a procedure that inherently possesses robustness against environmental variables and production uncertainties is pivotal. Despite the numerous complexities associated with real-time empirical methodologies, this work has demonstrated the possibility of achieving a $10^{-5}$ m compensation in the planar CR-CM offset. Additionally, there is optimism for attaining higher precision with further refinement of this method, as evidenced by the roll axis reaching a precision of $10^{-6}$ m in the second PID experi-

mental execution. This achievement provides accurate data of STASIS, essential for progressing to a consistent state observation, unbiased by CAD information. Considering that observe-and-compensate methods typically yield less precise outcomes, the active control's error propagation is judged to be minimal. This factor renders it a robust solution for the automatic balancing of the platform, effectively addressing the technical challenges posed by the compensation of a partially observable system.

## 7.2. Recommendations for future developments

Given the procedure's elevated complexity and susceptibility to errors, expanding the array of available hardware could facilitate the implementation of a technique with enhanced robustness in facing the real-world variables.

While this work demonstrates only virtually the outcomes achievable with RWs, future advancements could leverage them, especially as they are planned to be mounted on STASIS to simulate complex GNC algorithms performances. Hence, the restored observability would facilitate the design of a balancing process executed in a singular phase.

It is noteworthy that utilizing RWs does not inherently preclude the implementation of active control strategies. In fact, RWs serve as instrumental components in re-establishing observability and can be integrated into an active control framework, thereby enabling comprehensive, multi-directional balancing executions in a singular operational phase.

Subsequent recommendations also advocate for a more rigorous examination of STASIS disturbances, hence, the potential integration of high-fidelity aerodynamic effects and the structure's anisoelasticity. Such incorporation could be crucial for offset compensation through an augmented Kalman filter that concurrently estimates inertia parameters and offset in a singular step. This filter has demonstrated a precise capacity for compensating simulator offset but necessitates an exceptionally accurate dynamic model and three-dimensional actuation to ensure complete observability.

Additionally, considering that balancing procedures focus on the cancellation of the major disturbance only, delving into these advanced disturbances and incorporating them into the simulation model would significantly contribute to the development of more reliable and robust GNC algorithms for future deep-space probes missions.

The experimental endeavors provided valuable insights, from which actionable recommendations for the future development of the platform can be discerned. To tackle the jamming of the pitch-oriented MMU, caused by structural discontinuities, a revision of the actuator assembly, including the re-installation of the damaged wheel, is proposed. In addressing the structural bending of the leadscrew, a redesign of the rigid joint at the terminus of the thread is suggested, potentially mitigating the internal stress when the

MMU is positioned centrally on the leadscrew. Furthermore, the implementation of a router appositely tailored for the ESH is advisable. While the current continuous connection duration might suffice for balancing procedures, for GNC simulations extending over several days, inherent to the EXTREMA project, the risk of disconnection is untenable. Finally, the interruption of simulations and tests due to battery depletion reaffirms the need for STASIS to achieve energetic autonomy, aligning with the planned design advancements of the platform.

# Bibliography

[1] Gianfranco Di Domenico. Development of a hardware-in-the-loop simulation framework for interplanetary transfers on smaller timescales. Master's thesis, Politecnico di Milano, 2019.

[2] Gianfranco Di Domenico, Eleonora Andreis, Andrea Carlo Morelli, Gianmario Merisio, Vittorio Franzese, Carmine Giordano, Alessandro Morselli, Paolo Panicucci, Fabio Ferrari, Francesco Topputo, et al. Toward self-driving interplanetary CubeSats: The ERC-funded project EXTREMA. In *72nd International Astronautical Congress (IAC 2021)*, pages 1–11, 2021.

[3] Eleonora Andreis, Vittorio Franzese, Francesco Topputo, et al. An overview of autonomous optical navigation for deep-space CubeSats. In *INTERNATIONAL ASTRONAUTICAL CONGRESS: IAC PROCEEDINGS*, pages 1–11, 2021.

[4] P Panicucci, E Andreis, V Franzese, F Topputo, et al. An Overview of the EXTREMA Deep-Space Optical Navigation Experiment. In *3rd Space Imaging Workshop*, pages 1–3, 2022.

[5] E Andreis, P Panicucci, V Franzese, F Topputo, et al. A Robust Image Processing Pipeline for Planets Line-Of-Sight Extraction for Deep-Space Autonomous Cubesats Navigation. In *44th AAS Guidance, Navigation and Control Conference*, pages 1–19, 2022.

[6] Eleonora Andreis, Paolo Panicucci, Fabio Ornati, Francesco Topputo, et al. Towards Validation and Verification of Autonomous Vision-Based Navigation for Interplanetary Spacecraft. In *12th International Conference on Guidance, Navigation & Control Systems (GNC) and 9th International Conference on Astrodynamics Tools and Techniques (ICATT)*, pages 1–14, 2023.

[7] Alessandro Morselli, Andrea Carlo Morelli, Francesco Topputo, et al. ETHILE: A Thruster-In-The-Loop Facility to Enable Autonomous Guidance and Control of Autonomous Interplanetary CubeSat. In *73rd International Astronautical Congress (IAC 2022)*, pages 1–10, 2022.

[8] Christian Hofmann, Andrea C Morelli, and Francesco Topputo. Performance Assessment of Convex Low-Thrust Trajectory Optimization Methods. *Journal of Spacecraft and Rockets*, 60(1):299–314, 2023. doi: 10.2514/1.A35461.

[9] Christian Hofmann, Andrea C Morelli, and Francesco Topputo. On the performance of discretization and trust-region methods for on-board convex low-thrust trajectory optimization. In *AIAA Scitech 2022 Forum*, page 1892, 2022. doi: 10.2514/6.2022-1892.

[10] Gianmario Merisio, Francesco Topputo, et al. Characterization of ballistic capture corridors aiming at autonomous ballistic capture at Mars. In *2021 AAS/AIAA Astrodynamics Specialist Conference*, pages 1–21, 2022.

[11] Gianmario Merisio, Francesco Topputo, et al. Backbone of Ballistic Capture Set. In *33rd AAS/AIAA Space Flight Mechanics Meeting*, pages 1–22, 2023.

[12] Martina Liotta, Gianmario Merisio, Carmine Giordano, and Francesco Topputo. Synthesis of Ballistic Capture Corridors at Mars via Polynomial Chaos Expansion. *arXiv*, 2023. doi: 10.48550/arXiv.2309.01678.

[13] Anton Bahu and Dario Modenini. Automatic mass balancing system for a dynamic CubeSat attitude simulator: development and experimental validation. *CEAS Space Journal*, 03 2020. doi: 10.1007/s12567-020-00309-5.

[14] Jana Schwartz, Mason Peck, and Christopher Hall. Historical Review of Air-Bearing Spacecraft Simulators. *Journal of Guidance Control and Dynamics*, 26:513–522, 05 2003. doi: 10.2514/2.5085.

[15] G. A. Smith. Dynamic simulators for test of space vehicle attitude control systems. *Va. Polytech. Inst. Proc. od the Conf. on the role of simulation in space technol.*, pages 1–30, 1964.

[16] Simone Chesi, Octavio Perez, and Marcello Romano. A Dynamic, Hardware-in-the-Loop, Three-Axis Simulator of Spacecraft Attitude Maneuvering with Nanosatellite Dimensions. *Journal of small satellites*, 2015.

[17] Kelsey Saulnier, David Duran Perez, R. C. Huang, Daniele Gallardo, Grace Tilton, and Riccardo Bevilacqua. A six-degree-of-freedom hardware-in-the-loop simulator for small spacecraft. *Acta Astronautica*, 105:444–462, 2014. doi: 10.1016/j.actaastro.2014.10.027.

[18] Yashwanth Kumar Nakka, Rebecca Foust, Elena Sorina Lupu, David B. Elliott, Irene S. Crowell, Soon-Jo Chung, and Fred Y. Hadaegh. A Six Degree-of-Freedom

Spacecraft Dynamics Simulator for Formation Control Research. *AIAA Astrodynamics Specialist Conference*, 2018.

[19] G. Di Domenico and F. Topputo. STASIS: An Attitude Testbed for Hardware-in-the-Loop Simulations of Autonomous Guidance, Navigation, and Control Systems. In *73rd International Astronautical Congress (IAC 2022)*, pages 1–20, 2022.

[20] Alessandro Morselli, Gianfranco Di Domenico, Eleonora Andreis, Andrea Carlo Morelli, Gianmario Merisio, Vittorio Franzese, Carmine Giordano, Paolo Panicucci, Fabio Ferrari, Francesco Topputo, et al. The EXTREMA orbital simulation hub: A facility for GNC testing of autonomous interplanetary CubeSat. In *4S Symposium*, pages 1–13, 2022.

[21] Carmine Giordano, Francesco Topputo, et al. SPESI: a Real-Time Space Environment Simulator for the EXTREMA Project. In *33rd AAS/AIAA Space Flight Mechanics Meeting*, pages 1–13, 2023.

[22] Marco Schingaro. Multi-physics modeling of an attitude simulation platform: a virtual prototype for STASIS' digital twin. Master's thesis, Politecnico di Milano, 2022.

[23] Ya S Zinkevich. Quasi-optimal deceleration of rotational motion of a dynamically symmetric rigid body in a resisting medium. *Mechanics of Solids*, 51(2):156–160, 2016. doi: 10.3103/S0025654416020035.

[24] Rodrigo Cardoso da Silva. *Filtering and adaptive control for balancing a nanosatellite testbed*. PhD thesis, University of Brasìlia, 07 2018.

[25] W. Kang and J-P Barbot. Discussion on Observability and Invertibility. *IFAC Proceedings Volumes*, 40(12):426–431, 2007. doi: 10.3182/20070822-3-ZA-2920.00070. 7th IFAC Symposium on Nonlinear Control Systems.

[26] Rodrigo Cardoso da Silva, Renato Alves Borges, Simone Battistini, and Chantal Cappelletti. A review of balancing methods for satellite simulators. *Acta Astronautica*, 187:537–545, 2021. ISSN 0094-5765. doi: 10.1016/j.actaastro.2021.05.037.

[27] Yang Liu, Li Li, Zhenxian Fu, Jiubin Tan, and Kang Li. Automatic mass balancing of a spacecraft simulator based on non-orthogonal structure. In *2016 UKACC 11th International Conference on Control*, pages 1–6, 2016. doi: 10.1109/CONTROL. 2016.7737579.

[28] Yanbin Li and Youhua Gao. Equations of motion for the automatic balancing system of 3-DOF spacecraft attitude control simulator. In *2010 3rd International Symposium*

on Systems and Control in Aeronautics and Astronautics, pages 248–251, 2010. doi: 10.1109/ISSCAA.2010.5633650.

[29] Jeff S Young. Balancing of a small satellite attitude control simulator on an air bearing. Master's thesis, Utah State University, 1998.

[30] Jae Jun Kim and Brij N. Agrawal. Automatic Mass Balancing of Air-Bearing-Based Three-Axis Rotational Spacecraft Simulator. Journal of Guidance, Control, and Dynamics, 32(3):1005–1017, 2009. doi: 10.2514/1.34437.

[31] Ghasem Sharifi, Mehran Mirshams, and Hamed Shahmohamadi Ousaloo. Mass properties identification and automatic mass balancing system for satellite attitude dynamics simulator. Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering, 233:896 – 907, 2019. doi: 10.1177/0954410017742932.

[32] Jason A Keim, A Behcet Acikmese, and Joel F Shields. Spacecraft inertia estimation via constrained least squares. In 2006 IEEE Aerospace Conference. IEEE, 2006. doi: 10.1109/AERO.2006.1655995.

[33] Dong Hoon Kim, Dae-Gyun Choi, and Hwa-Suk Oh. Inertia estimation of spacecraft based on modified law of conservation of angular momentum. Journal of Astronomy and Space Sciences, 27(4):353–357, 2010. doi: 10.5140/JASS.2010.27.4.353.

[34] Zheyao Xu, Naiming Qi, and Yukun Chen. 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. ISSN 0094-5765. doi: 10.1016/j.actaastro.2015.08.010.

[35] Takeya Shima, Mitsunori Saito, Kazuhiko Fukushima, and Katsuhiko Yamada. Automatic balancing for a three-axis spacecraft simulator. Transactions of the Japan society for aeronautical and space sciences, aerospace technology Japan, 8(ists27): 15–22, 2010.

[36] Sam Wright. Parameter estimation of a spacecraft simulator using parameter-adaptive control. MS Project, Aerospace and Ocean Engineering Department, Virginia Polytechnic Institute and State University, Blacksburg, VA, 2006.

[37] Jana Schwartz and Christopher Hall. Comparison of System Identification Techniques for a Spherical Air-Bearing Spacecraft Simulator. Advances in the Astronautical Sciences, 01 2004.

[38] Donghoon Kim, Sungwook Yang, and Sangchul Lee. Rigid body inertia estimation

using extended Kalman and Savitzky-Golay filters. *Mathematical Problems in Engineering*, 2016, 2016. doi: 10.1155/2016/2962671.

[39] Pooya Sekhavat, Qi Gong, and I Michael Ross. NPSAT1 Parameter Estimation using Unscented Kalman Filtering. In *2007 American Control Conference*, pages 4445–4451. IEEE, 2007. doi: 10.1109/ACC.2007.4283031.

[40] Pooya Sekhavat, Mark Karpenko, and Isaac Ross. Ukf-based spacecraft parameter estimation using optimal excitation. In *AIAA Guidance, Navigation, and Control Conference*, page 5786, 2009. doi: 10.2514/6.2009-5786.

[41] Matti Raitoharju and Robert Piché. On Computational Complexity Reduction Methods for Kalman Filter Extensions. *IEEE Aerospace and Electronic Systems Magazine*, 34(10):2–19, 2019. doi: 10.1109/MAES.2019.2927898.

[42] Marcello Romano and Brij N. Agrawal. Acquisition, tracking and pointing control of the Bifocal Relay Mirror spacecraft. *Acta Astronautica*, 53(4):509–519, 2003. ISSN 0094-5765. doi: 10.1016/S0094-5765(03)80011-5. The New Face of Space Selected Proceedings of the 53rd International Astronautical Federation Congress.

[43] Simone Chesi, Qi Gong, Veronica Pellegrini, Roberto Cristi, and Marcello Romano. Automatic Mass Balancing of a Spacecraft Three-Axis Simulator: Analysis and Experimentation. *Journal of Guidance, Control, and Dynamics*, 37(1):197–206, 2014. doi: 10.2514/1.60380.

[44] F Landis Markley and John L Crassidis. *Fundamentals of spacecraft attitude determination and control*, volume 1286. Springer, 2014. doi: 10.1007/978-1-4939-0802-8.

[45] F. Bernelli Zazzera. *Spacecraft Attitude Dynamics, course notes part 3: Introduction to state-space control and attitude control*. Politecnico di Milano, 2021.

[46] Yeon Choi. Automatic mass balancing system for the 5-DOF spacecraft simulator. Master's thesis, Georgia Institute of technology, 08 2016.

[47] Hung Truong Xuan, Ahmed Chemori, Tuan Pham Anh, Huy Le Xuan, Thu Phan Hoai, and Phuong Vu Viet. From PID to L1 adaptive control for automatic balancing of a spacecraft three-axis simulator. *Int. J. Emerg. Technol. Adv. Eng*, 6(1):77–86, 2016.

[48] Jimmy Lau, Sanjay S Joshi, Brij N Agrawal, and Jong-Woo Kim. Investigation of periodic-disturbance identification and rejection in spacecraft. *Journal of guidance, control, and dynamics*, 29(4):792–798, 2006. doi: 10.2514/1.17341.

[49] Wang Yan, Huang Cheng, and Chen Xing Lin. Automatic mass balancing and design of a six degrees-of-freedom air bearing spacecraft simulator. In *2015 34th Chinese Control Conference (CCC)*, pages 5696–5700, 2015. doi: 10.1109/ChiCC.2015. 7260529.

[50] Keith Hudson, Andrew J Lingenfelter, and Joshuah A Hess. Dynamic Mass Balancing of a Spacecraft Test Platform. In *AIAA Scitech 2019 Forum*, 2019. doi: 10.2514/6. 2019-0941.

[51] Naira Hovakimyan and Chengyu Cao. *L1 Adaptive Control theory: Guaranteed robustness with fast adaptation.* SIAM, 2010. doi: 10.1137/1.9780898719376.

[52] Bing Hua, Lin Chen, Yunhua Wu, and Zhiming Chen. A study of PID and L1 adaptive control for automatic balancing of a spacecraft three-axis simulator. *International Journal of Intelligent Computing and Cybernetics*, 11(2):269–284, 2018. doi: 10.1108/IJICC-07-2017-0084.

[53] Rodrigo Cardoso da Silva, Fernando Cardoso Guimarães, João Victor Lopes de Loiola, Renato Alves Borges, Simone Battistini, and Chantal Cappelletti. Tabletop testbed for attitude determination and control of nanosatellites. *Journal of Aerospace Engineering*, 32(1):04018122, 2019. doi: 10.1061/(ASCE)AS.1943-5525.0000952.

[54] F. Bernelli Zazzera. *Spacecraft Attitude Dynamics, course notes part 1: Attitude dynamics and kinematics.* Politecnico di Milano, 2021.

[55] Christian Stary, Matthes Elstermann, Albert Fleischmann, and Werner Schmidt. Behavior-Centered Digital-Twin Design for Dynamic Cyber-Physical System Development. *Complex Systems Informatics and Modeling Quarterly*, pages 31–52, 04 2022. doi: 10.7250/csimq.2022-30.02.

[56] J Kofránek, M Mateják, P Privitzer, and M Tribula. Causal or acausal modeling: labour for humans or labour for machines. *Technical computing prague*, pages 1–16, 2008.

[57] G. Schweiger, H. Nilsson, J. Schoeggl, W. Birk, and A. Posch. Modeling and simulation of large-scale systems: A systematic comparison of modeling paradigms. *Applied Mathematics and Computation*, 365:124713, 2020. ISSN 0096-3003. doi: 10.1016/j.amc.2019.124713.

[58] Tomáš Kulhánek, Filip Ježek, Marek Mateják, Jan Šilar, and Jří Kofránek. Experiences in teaching of modeling and simulation with emphasize on equation-based and acausal modeling techniques. In *2015 37th Annual International Conference of the*

*IEEE Engineering in Medicine and Biology Society (EMBC)*, pages 3683–3686, 2015. doi: 10.1109/EMBC.2015.7319192.

[59] Jirı Kunovskỳ, Milan Pindryc, Václav Šátek, and FV Zboril. Stiff Systems in Theory and Practice. In *Proceedings of the 6th EUROSIM Congress on Modelling and Simulation*, page 6, 2007.

[60] T.D. Bui and T.R. Bui. Numerical methods for extremely stiff systems of ordinary differential equations. *Applied Mathematical Modelling*, 3(5):355–358, 1979. ISSN 0307-904X. doi: 10.1016/S0307-904X(79)80042-6.

[61] Marcel J Sidi. *Spacecraft dynamics and control: a practical engineering approach*, volume 7. Cambridge university press, 1997. doi: 10.1017/CBO9780511815652.

[62] Rebecca Ann Masterson. *Development and validation of empirical and analytical reaction wheel disturbance models*. PhD thesis, Massachusetts Institute of Technology, 1999.

[63] R.A. Masterton, D.W. Miller, and R.L. Grogan. Development and validation of reaction wheel disturbance model: Empirical model. *Journal of Sound and Vibration*, 249(3):575–598, 2002. ISSN 0022-460X. doi: 10.1006/jsvi.2001.3868.

[64] Hassan Alkomy and Jinjun Shan. Modeling and validation of reaction wheel microvibrations considering imbalances and bearing disturbances. *Journal of Sound and Vibration*, 492:115766, 2021. ISSN 0022-460X. doi: 10.1016/j.jsv.2020.115766.

[65] Abraham Savitzky and Marcel JE Golay. Smoothing and differentiation of data by simplified least squares procedures. *Analytical chemistry*, 36(8):1627–1639, 1964. doi: 10.1021/ac60214a047.

[66] S.D. Roy. On maximally flat sharp cutoff low-pass filters. *IEEE Transactions on Audio and Electroacoustics*, 19(1):58–63, 1971. doi: 10.1109/TAU.1971.1162160.

[67] I.W. Selesnick and C.S. Burrus. Generalized digital Butterworth filter design. *IEEE Transactions on Signal Processing*, 46(6):1688–1694, 1998. doi: 10.1109/78.678493.

[68] Hanspeter Schaub and John L Junkins. Analytical Mechanics of Space Systems. American Institute of Aeronautics and Astronautics. *Inc., Reston, VA*, page 581, 2003. doi: 10.2514/4.105210.

[69] Joaquim Martins, Peter Sturdza, and Juan Alonso. The Complex-Step Derivative Approximation. *ACM Transactions on Mathematical Software*, 29:245–262, 09 2003. doi: 10.1145/838250.838251.

[70] E.A. Wan and R. Van Der Merwe. The unscented Kalman filter for nonlinear estimation. In *Proceedings of the IEEE 2000 Adaptive Systems for Signal Processing, Communications, and Control Symposium (Cat. No.00EX373)*, pages 153–158, 2000. doi: 10.1109/ASSPCC.2000.882463.

[71] Gabriel A Terejanu. Unscented Kalman filter tutorial. *University at Buffalo, Buffalo*, 2011.

[72] Rambabu Kandepu, Bjarne Foss, and Lars Imsland. Applying the unscented Kalman filter for nonlinear state estimation. *Journal of Process Control*, 18(7):753–768, 2008. ISSN 0959-1524. doi: 10.1016/j.jprocont.2007.11.004.

[73] Christine Jardak, Elena Meshkova, Janne Riihijarvi, Krisakorn Rerkrai, and Petri Mahonen. Implementation and Performance Evaluation of nanoIP Protocols: Simplified Versions of TCP, UDP,HTTP and SLP for Wireless Sensor Networks. In *2008 IEEE Wireless Communications and Networking Conference*, pages 2474–2479, 2008. doi: 10.1109/WCNC.2008.435.

[74] Charles Severance. Discovering JavaScript Object Notation. *Computer*, 45(4):6–8, 2012. doi: 10.1109/MC.2012.132.

[75] Lindsay Bassett. *Introduction to JavaScript object notation: a to-the-point guide to JSON*. " O'Reilly Media, Inc.", 2015.

# A | IMU validation

Comprehensive review and adherence to the IMU datasheet were conducted to guarantee accurate data capture and appropriate measurement scaling. Preliminary tests were executed on the IMU sensor to verify the uniformity of the readings. The sensor underwent specific motions to promptly ascertain result consistency. Initially, a rotation around the yaw axis was applied, with the anticipation of a steady gravity vector reading.



(a) Angular rates.  (b) Linear accelerations.  (c) Gravity vector.
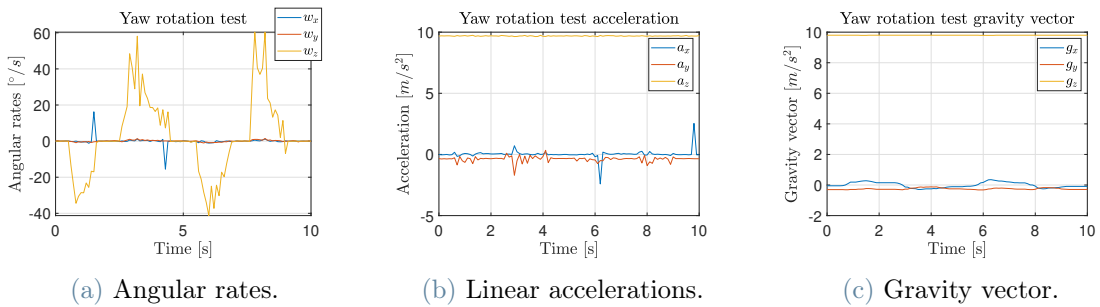
Figure A.1: Yaw rotation IMU test.

Subsequently, a rotation around the pitch axis was introduced to assess the proper inversion of the gravity vector.



(a) Angular rates.  (b) Linear accelerations.  (c) Gravity vector.
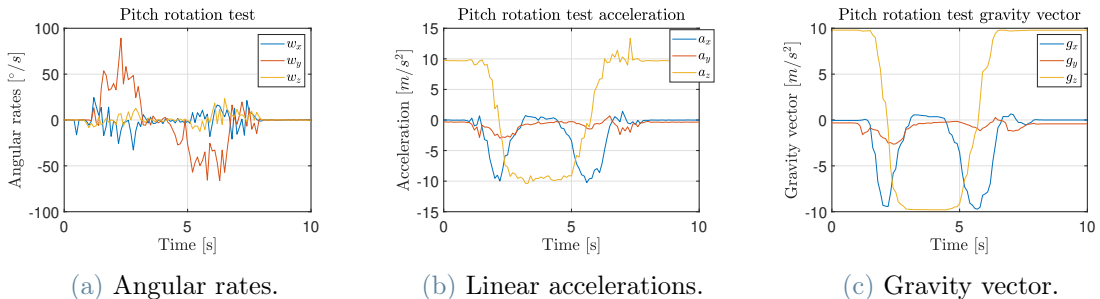
Figure A.2: Pitch rotation IMU test.

Evident from the accompanying Figures A.1 and A.2, the sensor readings align precisely with the expected outcomes, thereby confirming the precision of the sensor's configuration.

# List of Figures

# List of Tables

# List of Acronyms

**ADCS** Attitude Determination and Control Subsystem

**CLM** Classical Levenberg Marquardt

**CM** Centre of Mass

**CMG** Control Moment Gyro

**CR** Centre of Rotation

**CSD** Complex Step Differentiator

**DART** Deep-space Astrodynamics Research and Technology

**DOF** Degree Of Freedom

**DRF** Disturbance Rejection Filter

**DSN** Deep Space Network

**EKF** Extended Kalman Filter

**ENA** Enable port voltage

**EOM** Euler equation Of Motion

**ERC** European Research Council

**ESH** EXTREMA Simulation Hub

**EXTREMA** Engineering Extremely Rare Events in Astrodynamics for Deep-Space Missions in Autonomy

**FFT** Fast Fourier Transform

**FOV** Field Of View

**GNC** Guidance, Navigation and Control

**HTTP** Hypertext Transfer Protocol

**IEKF** Iterated Extended Kalman Filter

**JSON** JavaScript Object Notation

**LSB** Least Significant Bit

**LSQ** Least Squares

**MMU** Movable Mass Unit

**OCP** Optimal Control Problem

**PGMS** Power Generation and Management System

**PSD** Power Spectral Density

**REV** Reverse port voltage

**RLS** Recursive Least Squares Method

**RW** Reaction Wheel

**SPESI** EXTREMA SPace Environment SImulator

**STASIS** SpacecrafT Attitude SImulation System

**TCP** Transmission Control Protocol

**TD** Tracking Differentiator

**UDP** User Datagram Protocol

**UKF** Unscented Kalman Filter

# List of Symbols

| Variable | Description | SI unit |
|---|---|---|
| $B$ | Mathematical constraint matrix | - |
| $B_x$ | Roll drag coefficient | N m s$^2$ |
| $B_y$ | Pitch drag coefficient | N m s$^2$ |
| $B_z$ | Yaw drag coefficient | N m s$^2$ |
| $\mathbf{c}$ | Constraint vector | m |
| $D$ | Differentiation operator | - |
| $\mathbf{e}$ | Attitude error | - |
| $\mathbf{e}_{PID}$ | PID error input | - |
| $\mathbf{F}_g$ | Gravity force | N |
| $\mathbf{F}$ | Jacobian of state dynamics | - |
| $\mathbf{f}$ | Dynamics function | - |
| $\mathbf{g}$ | Gravity vector | N/kg |
| $\hat{\mathbf{g}}$ | Local Gravity direction | - |
| $g$ | Gravity vector norm | N/kg |
| $\mathscr{H}$ | Jacobian of Measurements | - |
| $\mathbf{H}$ | Platform angular momentum | kg m$^2$/s |
| $\mathbf{H}_d$ | Desired angular momentum | kg m$^2$/s |
| $\mathbf{h}$ | Measurements function | - |
| $\mathbf{h}_w$ | RWs angular momentum | kg m$^2$/s |
| $I$ | Identity matrix | - |
| $\tilde{\mathbf{J}}$ | Vector of inertia parameters | kg m$^2$ |
| $J$ | Inertia matrix | kg m$^2$ |
| $J_p$ | Inertia matrix without MMU | kg m$^2$ |
| $J_s$ | Inertia of solicited axis | kg m$^2$ |
| $\mathbf{K}$ | Kalman gain | - |

| | | |
|---|---|---|
| $K_D$ | Derivative gain | - |
| $K_I$ | Integral gain | - |
| $K_P$ | Proportional gain | - |
| $\mathbf{M}_{drag}$ | Drag momentum | N m |
| $m$ | Platform mass | kg |
| $m_i$ | i-th MMU mass | kg |
| $n$ | State dimension | - |
| $\mathbf{P}$ | State Covariance matrix | - |
| $\mathbf{P}_0$ | Initial State Covariance matrix | - |
| $\mathbf{P}_{ee}$ | Measurements Covariance matrix | - |
| $\mathbf{P}_{xy}$ | Cross Covariance matrix | - |
| $\mathbf{q}$ | Quaternion vector | - |
| $\mathbf{R}_i$ | i-th MMU position | m |
| $\mathbf{R}_k$ | Noise matrix | rad$^2$/s$^2$ |
| $\mathbf{r}$ | CR-CM offset | m |
| $\mathbf{r}_0$ | Initial offset | m |
| $\hat{\mathbf{r}}$ | Offset estimation | m |
| $\tilde{\mathbf{r}}$ | Offset estimation error | m |
| $r$ | Offset vector norm | m |
| $S$ | Stiffness index | - |
| $\mathbf{T}_{ext}$ | External Torque | N m |
| $\mathbf{T}_w$ | RWs Torque | N m |
| $T_p$ | Period of 3D pendulum | s |
| $\mathbf{u}$ | Control | N m |
| $\mathbf{V}$ | Z-observability matrix | - |
| $\mathbf{v}$ | Measurements noise | rad/s |
| $V$ | Lyapunov function | - |
| $\mathbf{w}$ | Unmodeled effects error | - |
| $W_i^{(c)}$ | Covariances weights | - |
| $W_i^{(m)}$ | Mean value weights | - |
| $\mathbf{x}$ | State vector | - |
| $\hat{\mathbf{x}}$ | State estimate vector | - |
| $\mathbf{Y}$ | Measurements | rad/s |

| | | |
|---|---|---|
| $\mathbf{y}$ | Measurements | rad/s |
| $\hat{\mathbf{y}}$ | Measurements prediction | rad/s |
| $\hat{\mathbf{z}}$ | Longitudinal Body frame direction | - |
| $\alpha$ | Sigma points generation coefficient | - |
| $\beta$ | Sigma points generation coefficient | - |
| $\boldsymbol{\Gamma}$ | Positive definite matrix of gains | - |
| $\boldsymbol{\gamma}_i$ | UKF measurements prediction | - |
| $\varepsilon$ | Drag coefficient | - |
| $\theta$ | Pitch angle | rad |
| $\kappa$ | Sigma points generation coefficient | - |
| $\Lambda$ | Medium properties coefficient | - |
| $\lambda_i$ | Eigenvalues of the problem | - |
| $\sigma$ | Inertia variance | kg m$^2$ |
| $\boldsymbol{\tau}_u$ | Control torque | N m |
| $\boldsymbol{\Phi}$ | Torque-state mapping matrix | - |
| $\phi$ | Roll angle | rad |
| $\boldsymbol{\chi}_i$ | Sigma points | - |
| $\boldsymbol{\Psi}$ | Anti-symmetric matrix | - |
| $\psi$ | Yaw angle | rad |
| $\boldsymbol{\Omega}$ | Matrix for LSQ | - |
| $\boldsymbol{\omega}$ | Angular velocity | rad/s |