University of Minho
School of Engineering

Pedro Nuno Reis Andrade

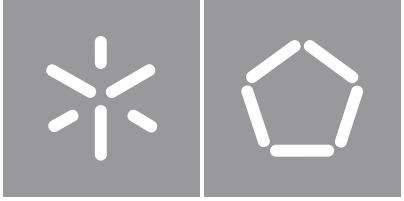
Development and integration of sun sensors pyramidal arrays on PROMETHEUS-1 for attitude determination

July 2025

Development and integration of sun sensors pyramidal arrays on PROMETHEUS-1 for attitude determination

Pedro Nuno Reis Andrade

UMinho | 2025



University of Minho
School of Engineering

Pedro Nuno Reis Andrade

Development and integration of sun sensors pyramidal arrays on PROMETHEUS-1 for attitude determination

Master's dissertation
Master's degree in Aerospace Engineering
Specialisation in Space Systems

Work supervised by
Prof. Dr. Alexandre Ferreira da Silva
and
Prof. Dr. Estela Bicho

July 2025

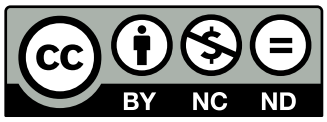
COPYRIGHT AND TERMS OF USE OF THIS WORK BY A THIRD PARTY

This is academic work that can be used by third parties as long as internationally accepted rules and good practices regarding copyright and related rights are respected.

Accordingly, this work may be used under the license provided below.

If the user needs permission to make use of the work under conditions not provided for in the indicated licensing, they should contact the author through the RepositóriUM of University of Minho.

License granted to the users of this work



Creative Commons Attribution-NonCommercial-NoDerivatives 4.0 International

CC BY-NC-ND 4.0

https://creativecommons.org/licenses/by-nc-nd/4.0/deed.en

STATEMENT OF INTEGRITY

I hereby declare having conducted this academic work with integrity. I confirm that I have not used plagiarism or any form of undue use of information or falsification of results along the process leading to its elaboration.

I further declare that I have fully acknowledged the Code of Ethical Conduct of the University of Minho.

To my parents, brother and significant other.

Acknowledgements

First and foremost, I would like to express my sincere gratitude to my dissertation supervisors — Prof. Dr. Alexandre Ferreira da Silva and Prof. Dr. Estela Bicho. Over the past two years, they provided me with the freedom to explore my research interests while consistently offering thoughtful advice and constructive criticism. Their trust in my judgment and continued support played a crucial role in shaping both the direction of this dissertation and my development as a researcher.

I would also like to acknowledge the professors, laboratory groups, and technical assistants of the Industrial Electronics Department, whose expertise and assistance were essential to the practical aspects of my work.

I am extremely grateful to my mentor, Carla Sepúlveda, for her insightful guidance and unwavering encouragement. Although our paths crossed only briefly, the impact was lasting. Your work ethic inspired me deeply, and I genuinely appreciated your concern for both me and my family.

I could not have undertaken this journey without the friends who walked with me — Gaybotas, Zombie Pitt, Águas, Ativista, Baljeet, Bocks, Cuco, Dias, Formigas, Greg, Joca, Maze, Parkour, Peter Babs, Pito, Reverse, Sérgio, Sherlock, Splinter, Trucker, Uapinha, and Xanax. From legendary stories to dawn-lit study sessions, your presence made this journey far more bearable. I am also grateful to those who joined halfway and left their mark — António, João, Jorge, Bárbara, and David.

To my friends outside the academic world — David, Emmanuel, Zé, Fábio, Érica, Nélson, and Veiga — thank you for being there through it all. Whether it was a night out or just the usual nonsense, you helped me step back, laugh, and breathe. You reminded me that there is more to life than deadlines.

This endeavour would not have been possible without my significant other, Maria. Thank you for being my steady anchor. Your love, patience, and unwavering belief gave me the strength to keep going, especially when that meant sacrificing time together. I am endlessly grateful for your presence over these past five years and for making me the happiest man alive.

Finally, I would like to express my deepest gratitude to my family. To my parents — thank you for your

unconditional love, support, and sacrifices. Your belief in the value of work and education laid the foundation for everything I have accomplished. My success is as much yours as it is mine. Your encouragement was always there, even when I did not call as often as you would have liked. To my brother — thank you for putting up with me through this rollercoaster. I hope, in some way, this journey sets a good example or at least gives you slightly fewer reasons to tease me. To my grandparents, aunts, uncles, and cousins — thank you for your care, for your curiosity about what I have been doing all this time, and for helping shape who I have been since day one.

Completing this dissertation has been the most challenging experience of my academic journey. It would not have been possible without the support, guidance, and encouragement of all these people. As Isaac Newton once said, *"If I have seen further, it is by standing upon the shoulders of giants"*. I could not agree more.

Pedro Nuno Reis Andrade

Keep your eyes to the sky, never glued to your shoes.

Mac Miller

Abstract

Development and integration of sun sensors pyramidal arrays on PROMETHEUS-1 for attitude determination

PocketQubes suffer from limited functionality due to volume constraints, which also impact accuracy and redundancy in satellite systems, such as the Attitude Determination and Control System (ADCS). This dissertation addresses the development, integration, and validation of a novel Coarse Sun Sensor Pyramidal Array (CSSPA) designed for the PROMETHEUS-1 PocketQube. A flexible, low-cost, and miniaturised circuit leveraging Commercial Off-The-Shelf (COTS) components was proposed to enhance Sun vector observability without compromising performance, manufacturability, and compatibility with the current satellite architecture.

The methodology followed an iterative strategy, starting with a proof-of-concept to validate geometric and functional principles, progressing to an advanced prototype with optimised topology and manufacturing, and culminating in a final, miniaturised, flight-ready prototype for integration. A comprehensive validation was conducted by assessing the Sun vector determination across the entire Field-Of-View (FOV) of the sensor in laboratory settings after following a complete calibration framework for Sun sensors. In parallel, software simulations analysed attitude estimation using a Multiplicative Extended Kalman Filter (MEKF), following orbital dynamics, Space environment and hardware sensors models.

Results showed a CSSPA accuracy of 1.61° (4.13° 3σ) in the Sun vector problem, and a FOV of 5 st ($\approx$170x170°). Resilient attitude estimation was achieved, with an accuracy of 0.95° (2.39° 3σ) and filter convergence under 600 s, in both nominal and extreme dynamic and noise conditions. Key contributions include the design of this state-of-the-art CSSPA system tailored to nanosatellite constraints, as well as the creation of a reusable simulation framework for ADCS validation on the PROMETHEUS satellite platform. This research further strengthens the capability and viability of the PocketQube standard.

Keywords: Sun sensor array, attitude determination, PocketQube, software simulation

Resumo

Desenvolvimento e integração de vetores piramidais de sensores solares no PROMETHEUS-1 para determinação de atitude

PocketQubes têm funcionalidade limitada devido a restrições de volume, afetando a exatidão e redundância de certos sistemas, como o de Determinação e Controlo de Atitude (SDCA). Esta dissertação aborda o desenvolvimento, integração e validação de um inovador vetor piramidal de sensores solares concebido para o PocketQube PROMETHEUS-1. Foi proposto um circuito flexível, de baixo custo e miniaturizado, utilizando componentes comerciais de uso genérico para melhorar a observabilidade do vetor solar sem comprometer o desempenho, capacidade de fabrico e compatibilidade com a arquitetura do satélite.

A metodologia seguiu uma estratégia iterativa, começando com uma prova de conceito para validar os princípios geométricos e funcionais, progredindo para um protótipo avançado com topologia e fabrico optimizados, e culminando num protótipo final miniaturizado pronto para voo. Foi efetuada uma avaliação exaustiva da determinação do vetor solar em todo o campo de visão do sensor, em ambiente laboratorial, após seguido um procedimento de calibração completo para sensores solares. Em paralelo, simulações de software analisaram a estimativa de atitude utilizando um filtro de Kalman extendido multiplicativo, com modelos de dinâmica orbital, ambiente espacial e sensores físicos.

Os resultados do sensor mostraram uma exatidão de 1.61° (4.13° 3σ) no problema do vetor solar, e um campo de visão de 5 st (≈170x170°). A estimativa de atitude foi resiliente, com uma exatidão de 0.95° (2.39° 3σ) e convergência do filtro em 600 s, tanto em condições dinâmicas e de ruído nominais como extremas. As principais contribuições incluem a conceção de um sistema de vetores solares de última geração adaptado às restrições dos nanosatélites, bem como a criação de uma ferramenta de simulação reutilizável para validação do SDCA na plataforma de satélites PROMETHEUS. Esta investigação reforça a capacidade e a viabilidade dos PocketQubes.

Palavras-chave: vetor de sensores solares, determinação de atitude, PocketQube, simulação de software

Contents

List of Figures **xiv**

List of Tables **xix**

List of Algorithms **xx**

Acronyms **xxi**

1 Introduction **1**

1.1 Framework 1

1.2 Problem statement 5

1.3 Objectives 6

1.4 Research questions 7

1.5 Methodology 7

1.6 Contributions 10

1.7 Document outline 11

2 Literature review **12**

2.1 Theoretical background 12

2.1.1 Attitude Determination Subsystem 13

2.1.2 Attitude parametrization 22

2.1.3 Reference frames 26

2.1.4 Spacecraft mechanics 29

2.1.5 Sun sensors 31

2.2 State-of-the-art 33

2.2.1 Sun sensor arrays 33

2.2.2 Sun vector determination and calibration . . . 38
2.2.3 Spacecraft attitude determination . . . 44

3 Proof-of-concept prototype 50

3.1 Introduction . . . 50
3.2 Photoelectric transducer selection . . . 51
3.3 Array system design . . . 56
3.4 Sun vector determination methodology . . . 61
3.5 Sun vector determination results analysis . . . 66
3.6 Summary . . . 71

4 Advanced prototype 72

4.1 Introduction . . . 72
4.2 Array topology optimisation . . . 73
4.3 Miniaturized array manufacturing . . . 79
4.4 Sun vector determination methodology . . . 88
4.5 Sun vector determination results analysis . . . 95
4.5.1 Single pyramid case study . . . 97
4.5.2 Combined pyramids case study . . . 113
4.6 Summary . . . 115

5 Final prototype 117

5.1 Introduction . . . 117
5.2 Array modifications . . . 118
5.3 Satellite modifications . . . 124
5.4 Summary . . . 128

6 Attitude determination software simulation 129

6.1 Introduction . . . 129
6.2 Space environment . . . 130
6.3 Satellite model . . . 139
6.4 MEKF implementation . . . 146
6.5 Attitude determination methodology . . . 149

6.6 Attitude determination results analysis 151
6.6.1 Nominal case study 152
6.6.2 Extreme case study 154
6.7 Summary 157

7 Conclusion **159**
7.1 Concluding remarks 159
7.2 Research answers 161
7.3 Future directions 163

Bibliography **166**

Appendices **176**

A1 From radiometry to photometry **176**

A2 Advanced prototype remaining illuminance results **179**

List of Figures

Figure 1 PROMETHEUS-1 exploded view. 4
Figure 2 PROMETHEUS-1 functional architecture. 5
Figure 3 Research methodology fluxogram. 9
Figure 4 Typical ADCS block diagram. 13
Figure 5 TRIAD frame constructed by two reference directions. 15
Figure 6 Kalman filter simplified operation diagram. 18
Figure 7 Definition of necessary reference frames (adapted from [30]). 27
Figure 8 Sun sensor main properties (adapted from [34]). 32
Figure 9 Different types of array assemblies, with transducers in red (adapted from [39]). . . 34
Figure 10 Relevant definitions of pyramidal arrays (in a 2D-sliced view). 35
Figure 11 Geometric representation of the Sun vector determination problem. 39
Figure 12 Fresnel effect on cosine transducers for different refractive indices. 42
Figure 13 OPT4001DTSR hardware (adapted from [67]). 53
Figure 14 OPT4001 response plots (adapted from [67]). 56
Figure 15 Proof-of-concept prototype support. 57
Figure 16 OPT4001 breakout PCB. 59
Figure 17 Proof-of-concept prototype assembly. 59
Figure 18 Feather M4 board pinout (adapted from [68]). 60
Figure 19 Cube assembly for proof-of-concept testing. 62
Figure 20 Proof-of-concept test setup parts. 63
Figure 21 Proof-of-concept test setup assembled. 64
Figure 22 Temporary frames defined for calculating the true incidence angle for each transducer. 65
Figure 23 Illuminance measured by the four transducers of the proof-of-concept prototype. . . 66

Figure 24 Normalised illuminance comparison between dataset and models (cosine/polyfitted). 68
Figure 25 Normalised Sun vector determination comparison between dataset and models. . . 69
Figure 26 Sun vector determination angular error of the proof-of-concept prototype. 70

Figure 27 Geometry regarding a thirty-degree inclination of transducers. 73
Figure 28 Different approaches to discretise the surface of a sphere. 74
Figure 29 Geometry of the proposed sensor configurations. 75
Figure 30 Three-component Sun coverage across the attitude sphere for multiple configurations with failing transducers. 77
Figure 31 Number of transducers with visibility across the 2D-projected attitude sphere for each configuration without failures. 78
Figure 32 Advanced prototype support iterations. 81
Figure 33 Advanced prototype support dimensions. 82
Figure 34 Advanced prototype circuits' schematics. 83
Figure 35 2D sketch of the FPC 84
Figure 36 Designed layer stackup of the FPC. 84
Figure 37 Advanced prototype flexible printed circuit. 86
Figure 38 Advanced prototype interface board. 87
Figure 39 Advanced prototype assembly. 88
Figure 40 Advanced prototype test setup parts. 89
Figure 41 Advanced prototype test setup. 91
Figure 42 Sensor alignment with the respective axes. 92
Figure 43 Advanced prototype combined assembly. 93
Figure 44 Illuminance evolution over time during static sampling. 96
Figure 45 Illuminance measured by the four transducers of the advanced prototype (first sensor). 98
Figure 46 Uncalibrated illuminance dependence on the incidence angle (first sensor). 99
Figure 47 Illuminance dependence on the incidence angle after light evolution calibration (first sensor). 101
Figure 48 Illuminance dependence on the incidence angle after normal vectors calibration (first sensor). 102
Figure 49 Comparison between transducers' normal vectors (first sensor). 103

Figure 50 Illuminance dependence on the incidence angle after characteristics calibration (first sensor). 104
Figure 51 Illuminance dependence on the incidence angle after iterative transmittance calibration with r_t=1.5 (first sensor). 105
Figure 52 Illuminance dependence on the incidence angle after iterative transmittance calibration with r_t=1.02 (first sensor). 106
Figure 53 Sun vector determination angular error of the advanced prototype (first sensor). . . 107
Figure 54 Comparison between transducers' normal vectors (second and third sensor). 108
Figure 55 Sun vector determination angular error of the advanced prototype (second and third sensor). 109
Figure 56 Sun vector determination coverage of the advanced prototype. 112
Figure 57 Sun vector determination angular error of the advanced prototype (three sensors combined). 114
Figure 58 Sun vector determination coverage of the advanced prototype (three sensors combined). 114

Figure 59 Available region for the CSSPA in the satellite's solar panels. 118
Figure 60 Updated stackup for the PCB accounting blind slots. 119
Figure 61 Final prototype support. 120
Figure 62 Final prototype circuit schematics. 122
Figure 63 Final prototype flexible printed circuit. 122
Figure 64 FPC design rules constraints and adopted dimensions. 123
Figure 65 3D rendering of the final prototype on top of a one cent coin. 124
Figure 66 Proposed schematic changes on PROMETHEUS-1. 126
Figure 67 Proposed layout changes on PROMETHEUS-1 main board. 127
Figure 68 Proposed layout changes on PROMETHEUS-1 solar boards. 127

Figure 69 Main blocks of Simulink implementation and its connections. 130
Figure 70 Space environment subsystem. 131
Figure 71 Attitude propagator blocks. 132
Figure 72 Space environment block. 132
Figure 73 Geometry for the Sun vector determination (adapted from [79]). 133
Figure 74 Different space environment models selected for implementation. 135

Figure 75 Geometry for the Earth's albedo determination (adapted from [33]). 136
Figure 76 2D scenarios where the satellite is not eclipsed (adapted from [83]). 138
Figure 77 Satellite model subsystem. 139
Figure 78 Simulated hardware measurements block. 140
Figure 79 Gyroscope hardware model. 141
Figure 80 Magnetometer hardware model. 142
Figure 81 Sun sensor hardware model. 143
Figure 82 Light vector composition. 143
Figure 83 Relative error distribution of the magnetic field intensity with a four-degree IGRF. . . 145
Figure 84 Expected sensor measurements from constrained models. 145
Figure 85 MEKF implementation subsystem. 146
Figure 86 Attitude error evolution on the nominal study case. 154
Figure 87 Gyro bias error evolution on the nominal study case. 154
Figure 88 Attitude error evolution on the extreme study case. 156
Figure 89 Gyro bias error evolution on the extreme study case. 157
Figure 90 Luminous efficiency and solar irradiance evolution over the visible spectrum. 177
Figure 91 Uncalibrated illuminance dependence on the incidence angle (second sensor). . . . 180
Figure 92 Illuminance dependence on the incidence angle after light evolution calibration (second sensor). 181
Figure 93 Illuminance dependence on the incidence angle after normal vectors calibration (second sensor). 182
Figure 94 Illuminance dependence on the incidence angle after characteristics calibration (second sensor). 183
Figure 95 Illuminance dependence on the incidence angle after iterative transmittance calibration with r_t=1.02 (second sensor). 184
Figure 96 Uncalibrated illuminance dependence on the incidence angle (third sensor). 185
Figure 97 Illuminance dependence on the incidence angle after light evolution calibration (third sensor). 186
Figure 98 Illuminance dependence on the incidence angle after normal vectors calibration (third sensor). 187

Figure 99 Illuminance dependence on the incidence angle after characteristics calibration (third sensor). 188

Figure 100 Illuminance dependence on the incidence angle after iterative transmittance calibration with r_t=1.02 (third sensor). 189

List of Tables

Table 1 Typical specifications of sensors used for attitude determination (adapted from [22]). 22

Table 2 Hamilton vs. JPL quaternion conventions (adapted from [26]). 25

Table 3 Literature review on pyramidal arrays, with key drivers compiled. 36

Table 4 Properties comparison of available ALS. 52

Table 5 Available I2C addresses on OPT4001 (adapted from [67]). 53

Table 6 Effective illuminance resolution of OPT4001, for each conversion time and light range scenario (adapted from [67]). 55

Table 7 Calibration parameters estimated during the framework. 110

Table 8 Accuracy performance results after each calibration phase. 110

Table 9 Accuracy performance results with the three combined sensors. 113

Table 10 Properties differences between OPT4001 and OPT4003-Q1. 121

Table 11 MEKF selected standard deviation noises. 147

Table 12 Simulation general initialisation parameters. 150

Table 13 MEKF initialization parameters. 150

Table 14 Satellite initialization constraints. 151

Table 15 State error performance results on the nominal study case. 153

Table 16 State error performance results on the extreme study case. 156

Table 17 Radiometric quantities and its photometric equivalents (adapted from [96]). 177

List of Algorithms

Algorithm 1 Singular value decomposition solution of Wahba's problem. 16

Algorithm 2 Discrete-time Kalman filter. 19

Algorithm 3 Discrete-time extended Kalman filter with Joseph's form. 20

Algorithm 4 Discrete-time invariant multiplicative extended Kalman filter with Joseph's form. . 48

Algorithm 5 OPT4001 conversion from a measurement digital representation to its illuminance value. 61

Algorithm 6 Sun coverage across the attitude sphere for multiple configurations with failing transducers. 76

Algorithm 7 Photoelectric transducers calibration (adapted from [36]). 95

Algorithm 8 Sun vector estimation with an iterative least squares (retrieved from [36]). . . . 95

Algorithm 9 Sun vector direction and magnitude algorithm [30]. 133

Algorithm 10 Earth's total albedo model from TOMS reflectivity data [33]. 137

Algorithm 11 Eclipse determination by the ray-sphere intersection problem [83]. 137

Algorithm 12 Discrete-time invariant multiplicative extended Kalman filter with Joseph's form, including magnetic field and Sun unit vectors as measurement updates. 148

Acronyms

2-DOF two degrees of freedom

2D two-dimensional

3-DOF three degrees of freedom

3D three-dimensional

A-FOV Array's Field-Of-View

ADC Analog-to-Digital Converter

ADCS Attitude Determination and Control System

ADS Attitude Determination Subystem

AEKF Additive Extended Kalman Filter

ALS Ambient Light Sensor

AM0 Air Mass Zero

ARW Angle Random Walk

CC Constant Current

CCD Charge-Coupled Device

CDC Communications Device Class

CDR Critical Design Review

CIE International Commission on Illumination

CMOS Complementary Metal-Oxide-Semiconductor

COCOM Coordinating Committee for Multilateral Export Controls

COTS Commercial Off-The-Shelf

CRC Cyclic Redundancy Check

CSS Coarse Sun Sensor

CSSPA Coarse Sun Sensor Pyramidal Array

CSV Comma-Separated Values

DCM Direction Cosine Matrix

ECEF Earth-Centered Earth-Fixed

ECI Earth-Centered Inertial

EKF Extended Kalman Filter

EMC Eletromagnetic Compatibility

EMI Eletromagnetic Interference

ENIG Electroless Nickel Immersion Gold

ESOQ EStimators of the Optimal Quaternion

FOAM Fast Optimal Attitude Matrix

FOV Field-Of-View

FPC Flexible Printed Circuit

FSS Fine Sun Sensor

GNSS Global Navigation Satellite System

GPIO General Purpose Input/Output

HASL Hot Air Solder Leveling

HITL Hardware-In-The-Loop

HPA Half-Power Angle

I2C Inter-Integrated Circuit

IC Integrated Circuit

IGRF International Geomagnetic Reference Field

IMU Inertial Measurement Unit

JD Julian Date

JPL Jet Propulsion Laboratory

LED Light-Emitting Diode

LEO Low Earth Orbit

LOS Line Of Sight

LS Least Squares

LSB Least Significant Bit

MAPE Mean Absolute Percentage Error

MCU Microcontroller Unit

MEKF Multiplicative Extended Kalman Filter

MEMS Microelectromechanical System

MRP Modified Rodrigues Parameters

MSB Most Significant Bit

MSE Mean Square Error

NED North-East-Down

NOAA National Oceanic and Atmospheric Administration

OBC On-Board Computer

PCB Printed Circuit Board

PLA Polylactic Acid

PMMA Poly(methyl methacrylate)

QUEST QUaternion ESTimator

RAAN Right Ascension of the Ascending Node

RMSE Root Mean Square Error

RRW Rate Random Walk

RTC Real Time Clock

RTD Resistance Temperature Detector

SITL Software-In-The-Loop

SLS Selective Laser Sintering

SMD Surface Mount Device

SNR Signal-to-Noise Ratio

SSO Sun-Synchronous Orbit

st.dev. standard deviation

SVD Singular Value Decomposition

T-FOV Transducer's Field-Of-View

TDB Barycentric Dynamical Time

TLE Two-Line Element set

TOMS Total Ozone Mapping Spectrometer

TRIAD TRI-axial Attitude Determination

UF2 Universal Serial Bus Flashing Format

USB Universal Serial Bus

UT1 Universal Time 1

UTC Coordinated Universal Time

V-FOV Vehicle's Field-Of-View

w.r.t. with respect to

Chapter 1

Introduction

1.1 Framework

The New Space economy, characterised by the increasing commercialisation of space exploration, has exhibited a continuous rise in global activity since 2005 [1], and this momentum is expected to persist [2]. In turn, numerous benefits have emerged in recent years, most notably a substantial reduction in the cost of space access [3]. This shift has enabled an unprecedented number of players, particularly private companies, startups, and academic institutions, to participate in space missions, contribute to innovation, and pursue applications ranging from Earth observation to scientific research.

In parallel, satellite technology has also seen a reduction of size and weight, fueled by the adoption of cheaper Commercial Off-The-Shelf (COTS) components. Although not as well-performing as space-graded components, they rely on flight heritage and radiation evaluation to behave adequately in a space environment [4]. Furthermore, standardised satellite form factors, such as CubeSats and PocketQubes, provide quicker development phases and more compatibility with smaller budgets [5], with multiple open-source projects available as knowledge providers. Overall, this environment paved the way for a broad democratisation of space access.

Despite both standards having limited volume and weight, they have been growing steadily in popularity in the past years [6], with CubeSats being the preferred option for satellite missions. From a total of 2,714 registered launches until September 30th, 2024, 92% of them used this standardised platform, whereas only 3% were PocketQubes [7].

Today, PocketQubes are still primarily used for researching the limits of satellite miniaturisation and demonstrating small satellite technology. Although it may prove to be a cheaper solution, the sustainability

of PocketQubes requires a clear advantage in its long-term cost-effectiveness compared to larger satellites [8]. They exhibit reduced functionality relative to CubeSats due to their limited physical constraints, typically lacking, for example, any form of Attitude Determination and Control System (ADCS) [5], which restricts satellite operation to a few minimal tasks.

With all this in mind, the PROMETHEUS project, a PocketQube framework designed for research and educational access to space, was developed to provide an open-source end-to-end pipeline for deploying a 1P PocketQube satellite, encompassing its design, assembly, licensing, certification, launch and operation, with all required documentation and guiding information released and explained[1]. The project addresses the primary challenges for small satellite missions, including funding constraints, limited expertise, tight schedules, limited launch opportunities, and testing and validation procedures. Fundamentally, it provides a low-cost, accessible, and customisable PocketQube hardware platform for others to learn and replicate. Furthermore, with its compact form factor of 5x5x5 cm and 250 grams, a PROMETHEUS satellite still features most of the classical spacecraft subsystems, including an active ADCS, which tackles one of the constraints raised above and aims to establish the PocketQube standard on the same level of popularity as the CubeSat.

Succinctly, PROMETHEUS satellites are composed of two main hardware parts: the internal stack, where the avionics (main), payload and battery boards are placed horizontally, and the external solar panels, on each of the cube's faces, screwed to the stack. This arrangement can be perceived in Figure 1. Software-wise, above the hardware layer, there is a firmware one implementing a CircuitPython environment, followed by the low-level code state machine and the device drivers, topped by the desired application, where specific tasks (or routines) are defined, such as the ones running during testing or flight mode. Figure 2 shows its functional architecture, detailing each subsystem main components and interfaces.

PROMETHEUS-1, the 1st Portuguese PocketQube, launched on January 14th 2025, onboard SpaceX's Falcon 9 during the Transporter-12 Rideshare mission, represented a significant milestone in the rapidly evolving Portuguese space ecosystem. Its primary mission objective was to validate the satellite development platform, together with its ADCS and payload: a small camera and a dedicated Microcontroller Unit (MCU) with machine-learning image classification capabilities. As of now, it has achieved orbit and initial communication, receiving telemetry and images during its first pass.

[1]The PROMETHEUS project was supported by the Portuguese Foundation for Science and Technology under the International Partnership CMU Portugal Program, in a consortium between Universidade do Minho, Instituto Superior Técnico and Carnegie Mellon University. Its public knowledge repository can be accessed on Github with `https://github.com/AFS-pt/PROMETHEUS-1`

Since February 2023, the author has had the opportunity to research PROMETHEUS-1, mainly regarding assembly, validation, documentation and operation of the flight unit, but also its ground segment, while simultaneously learning from CMU colleagues on how each satellite subsystem works. Later that year, under the CMU Portugal Visiting Students Program, another opportunity arose to work with them in Pittsburgh. The central research theme was the development of a low-cost Sun sensor array for nanosatellites to improve attitude determination performance and redundancy. The research culminated in the manufacturing and assessment of a proof-of-concept pyramidal prototype, from which improving and integrating it on PROMETHEUS-1 provided a follow-on continuation of the work.

This theme emerged as the Attitude Determination Subystem (ADS) integrates data from various sensors to estimate this orientation, enabling payload pointing, power optimisation via solar panel alignment, and stable communication links. Sun sensors, and in particular Coarse Sun Sensors (CSSs), have emerged as fundamental components of the ADSs in small satellites due to their simplicity, low power consumption, and reliability. However, individual CSSs offer limited Field-Of-View (FOV) and redundancy. This motivated the exploration of sensor arrays, which can combine multiple CSSs in a structured geometry to enhance spatial coverage and fault tolerance while maintaining simple manufacturing, reduced size and good performance without compromising the cost and modularity.

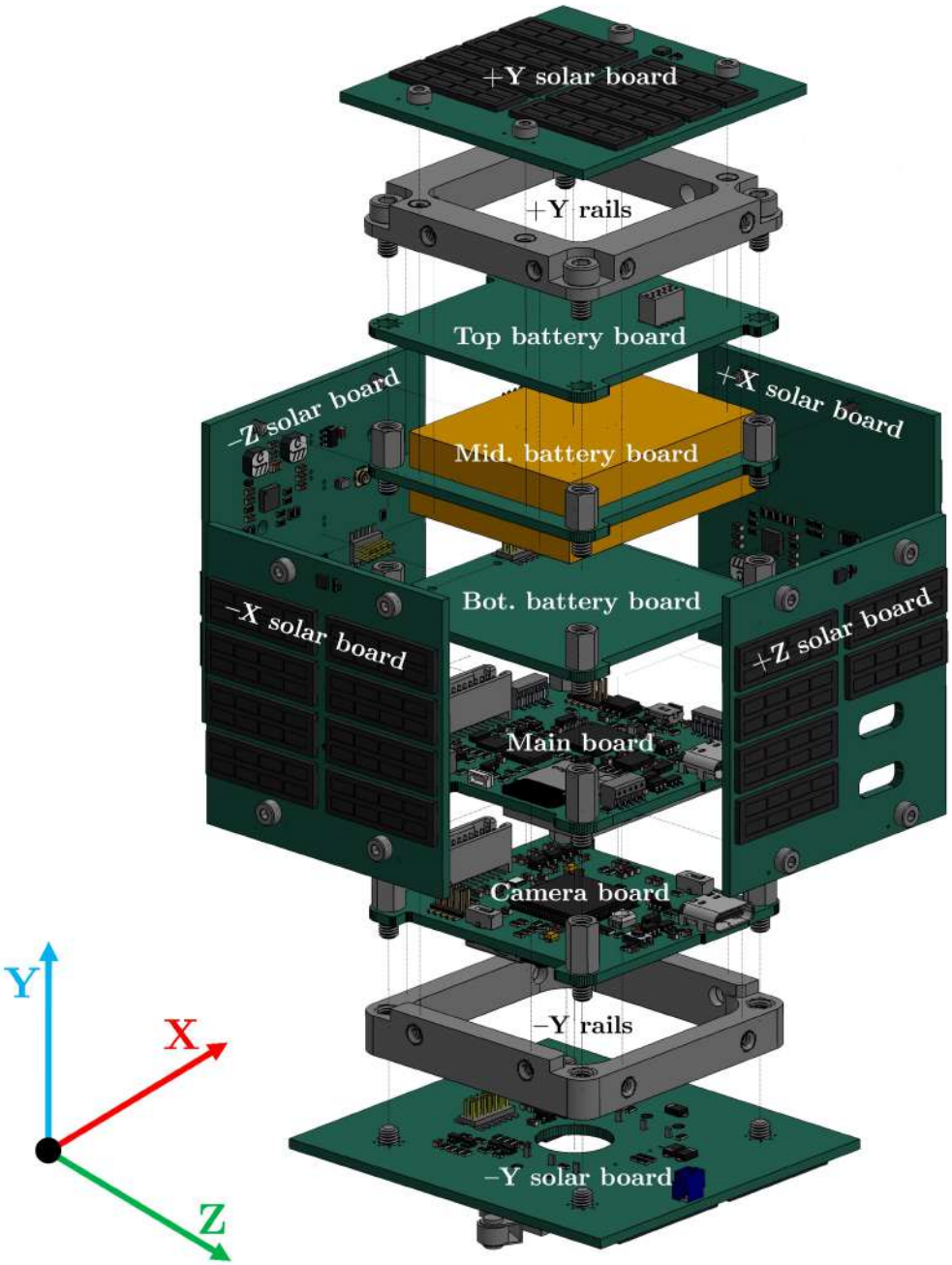


Figure 1: PROMETHEUS-1 exploded view.

Currently, the PROMETHEUS-1 satellite ADCS features embedded magnetorquers on the solar panels, which are the only actuators. Additionally, on the sensor side, which will be the focus of this dissertation, a gyroscope and magnetometer are used, together with one CSS for each solar panel.

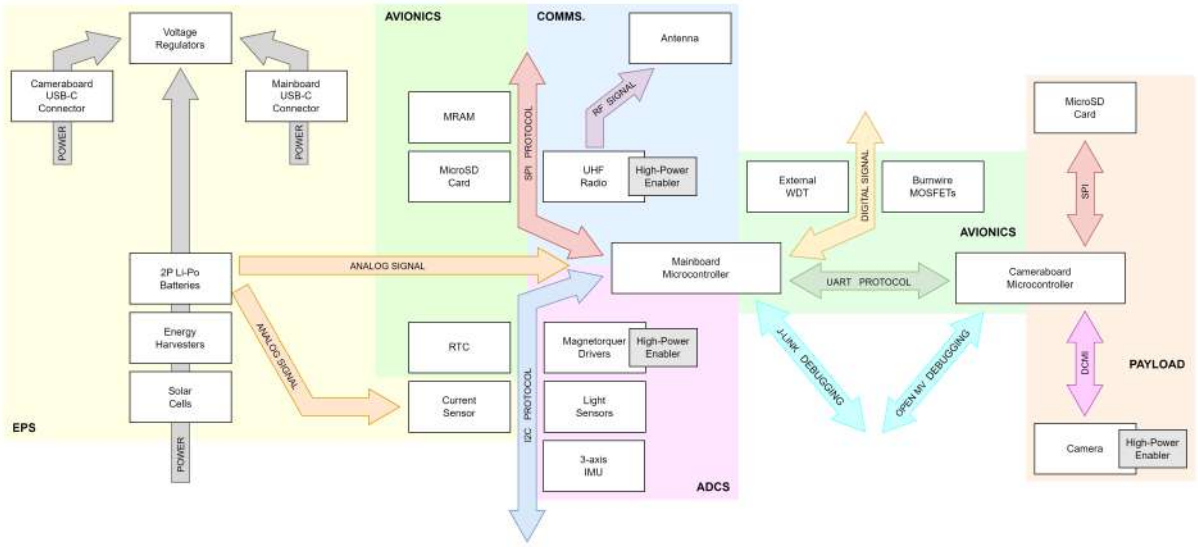


Figure 2: PROMETHEUS-1 functional architecture.

1.2 Problem statement

Currently, the Sun vector direction cannot be geometrically determined across almost the entirety of the satellite, as it relies on a flat and dispersed arrangement of the transducers. This, however, could be mitigated via software, namely by the employed Multiplicative Extended Kalman Filter (MEKF), which recursively estimates the attitude assuming noise on this vector. Nonetheless, further noise sources such as the albedo irradiance or even sensor misbehaviour are not taken into account, which undermines the attitude accuracy. Since no Global Navigation Satellite System (GNSS) receiver is implemented onboard, if these issues are not resolved, errors can accumulate to amounts that distrust the estimated attitude.

Moreover, the precise estimation of the Sun vector often requires advanced calibration methods of the sensors employed, which, although time and resource-consuming, should be conducted to assess and improve performance. The current alternative is to rely on proprietary calibrated sensors sold at expensive prices, usually on the order of thousands of Euros.

Furthermore, no structured testing platform is currently developed for the satellite's ADCS, neither in software nor hardware, which would be a valuable improvement for researching and validating purposes. As such, three specific research problems were identified and are addressed in this dissertation:

1. the absence of a modular, miniaturised, well-documented, and low-cost Sun sensor array capable of delivering high accuracy, good observability, and robust performance in the space environment;

2. the challenge of integrating such a system into an already-developed satellite platform without compromising existing functionalities or subsystem compatibility;

3. the need for a comprehensive validation framework of the sensor and ADS, ensuring its performance under realistic operational conditions and demonstrating flight-readiness.

1.3 Objectives

Fundamentally, this dissertation seeks to bridge this gap by designing, building, and evaluating an innovative Sun sensor system for nanosatellites that can be manufactured using COTS components and additive techniques, tested in laboratory settings, and validated through empirical and simulated frameworks tailored to PROMETHEUS-1's architecture. Although not ready in time for its scheduled flight, the proposed solution could be added in future satellite iterations.

The specific objectives of this dissertation are:

01 Review existing literature examples of Sun sensors for space applications and classify their performance for comparison;

02 Design and manufacture modular, miniaturised, and low-cost Sun sensor configuration that maximises field-of-view and redundancy;

03 Select, integrate, and calibrate transducers capable of delivering reliable and high-resolution measurements under varying light conditions;

04 Assemble a rigorous testing setup and detail testing procedures to be conducted in laboratory settings for Sun vector determination;

05 Experimentally validate the sensor array, achieving at least 1 degree of angular accuracy on Sun vector determination and a full observability coverage on the satellite;

06 Develop a simulation environment for attitude estimation on PROMETHEUS-1, and thus validate the simulated performance of the sensor on the ADS, achieving at least 1 degree of angular accuracy.

1.4 Research questions

The following research questions guided the developed work:

Q1 What are the critical design considerations for an array of Sun sensors optimised for accuracy and redundancy while keeping costs low and remaining easily manufactured?

Q2 How can the performance of the developed sensor array be validated under various operating conditions?

Q3 What calibration methods are necessary to ensure reliable Sun vector determination using the sensor array?

Q4 How does the performance of the developed sensor array compare with other solutions?

Q5 How can the sensor array be interfaced to achieve seamless integration into the already implemented satellite's ADS?

Q6 What characterises high-fidelity simulation environments for attitude estimation?

1.5 Methodology

To achieve the defined objectives (**O***) and answer the research questions (**Q***) that guided this dissertation, a structured and multi-phased research methodology was adopted.

The research started with an extensive literature review to understand current practices, limitations and performances in Sun sensors for spacecrafts. Particular emphasis was placed on identifying compact, low-cost, and modular Sun sensor configurations, while comparing properties and design choices. This approach directly informed the answers to **Q1**, and partially to **Q4**, while laying the foundation for **O1**.

State-of-the-art calibration frameworks, tailored for Sun sensors, were also identified during this phase, answering **Q2** and **Q3**. Attitude determination methods were also reviewed to aid on the implementation of the simulation, partially explaining **Q6**.

The core development of the Coarse Sun Sensor Pyramidal Array (CSSPA) followed a iterative prototyping strategy, successively improving objectives **O2** and **O3**. A proof-of-concept prototype was designed and tested to validate fundamental geometric principles of sun vector determination using pyramidal sensor arrangements. This was followed by an advanced prototype, where array topology and manufacturing

techniques were refined for miniaturisation and performance optimisation. Finally, a flight-ready prototype was designed to ensure compatibility with the PROMETHEUS-1 platform which, by evaluating electrical, mechanical and environmental constraints, answered **Q5**.

To access the performance and observability of the developed CSSPA on laboratory, calibration and testing frameworks were employed and iterated, while also upgrading the testing setup hardware. This approach successively improved objectives **04** and **05**, while providing the information required for answering **Q4**. **05** was only partially achieved as the obtained accuracy was 0.61 degrees above desired.

In-orbit performance of the updated ADS was evaluated with a detailed software simulation environment on MATLAB Simulink. It was improved regarding environmental and hardware behaviour with some specific literature examples, successfully achieving objective **06** and answering **Q6**.

Figure 3 shows a clear fluxogram representing this process.

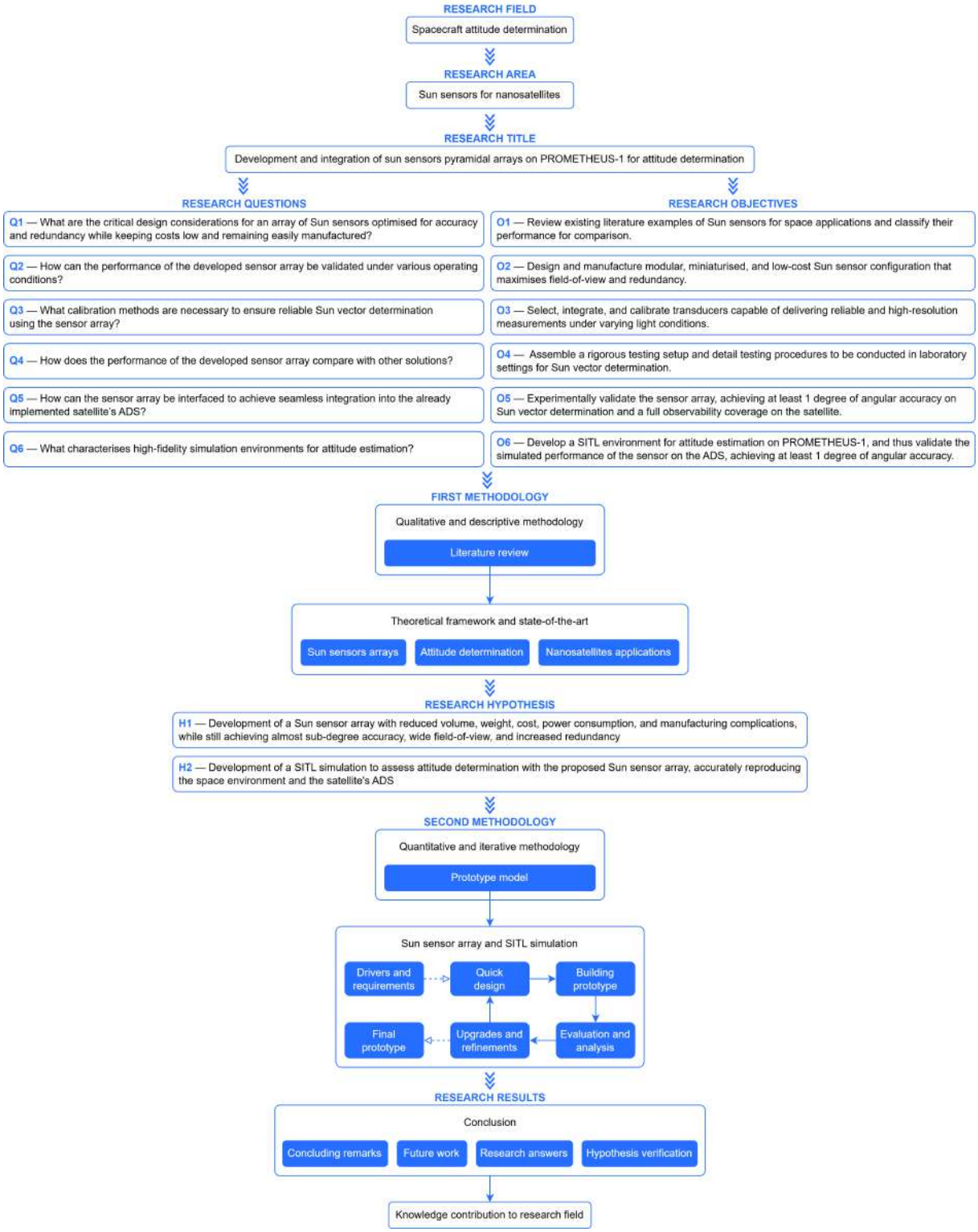


Figure 3: Research methodology fluxogram.

1.6 Contributions

This dissertation research contributes to the advancement of small satellite ADS and Sun sensor design overall, but also particularly to the PROMETHEUS project, through the following:

1. Design and implementation of a Flexible Printed Circuit (FPC)-based CSSPA design for nanosatellites, with reduced volume, weight, cost, power consumption, and manufacturing complications, while still achieving almost sub-degree accuracy, 5 steradians FOV, and increased redundancy;

2. Implementation of a simulation reproducing the space environment and the satellite's ADS, using MATLAB's Simulink, allowing to replicate the on-orbit performance of the employed MEKF for attitude determination with the integrated CSSPAs, and showing sub-degree convergence and high resilience across several dynamic scenarios, thus providing a reusable and accurate framework for PROMETHEUS.

In parallel, the author actively contributed to the broader PROMETHEUS research project, which led to several academic outputs. These works, while not entirely focused on the Sun sensor, addressed key aspects of the satellite's development and validation. They were presented at national and international venues, supporting the dissemination of the project's findings and reinforcing the applied research impact of the project. The following publications were made:

1. **P. R. Andrade**, A. F. Silva, "PROMETHEUS-1: A Small Satellite With A Big Responsibility For Portugal's Space Education", *2025 PocketQube Conference*, Glasgow, Scotland, March 25-27, 2025;

2. **P. R. Andrade**, A. F. Silva, "PROMETHEUS-1: the 1st Portuguese PocketQube", *1st Space Education Summit*, Porto, Portugal, April 16, 2025;

3. P. R. M. Fisch, **P. R. Andrade**, N. Khera, D. Luis, R. M. Rocha, R. Ventura, Z. Manchester, A. F. Silva, "PROMETHEUS-1: A Satellite Framework for Research and Educational Access to Space", in *Proceedings of the AIAA/USU Conference on Small Satellites*, Year in Review — Research & Academia Track, SSC25-RAIV-06, Utah, USA, August 10-13, 2025;

4. **P. R. Andrade**, P. R. M. Fisch, N. Khera, D. Luis, R. M. Rocha, R. Ventura, Z. Manchester, A. F. Silva, "Open-source Low-cost Architecture for Educational PocketQube Missions: PROMETHEUS Satellite and Ground Station", under review in *IEEE Transactions on Aerospace and Electronic Systems*;

In addition, the author was awarded for best oral communication of the *1st Space Education Summit* conference.

1.7 Document outline

The structure of the document is organised as follows:

Chapter 1 — Introduction Presents the research context and problem statement, proposed objectives and research questions, methodology used to guide the work and obtained contributions;

Chapter 2 — Literature review Covers the theoretical background of attitude determination, Sun sensors, and relevant estimation methods alongside recent state-of-the-art developments;

Chapter 3 — Proof-of-concept prototype Describes the initial prototype's design, testing, and performance evaluation, identifying challenges and improvement areas;

Chapter 4 — Advanced prototype Details the optimised sensor array, manufacturing improvements, refined testing, and enhanced calibration results from single and combined array trials;

Chapter 5 — Final prototype Describes the design for the miniaturised, flight-ready sensor design and proposed integration steps into PROMETHEUS-1;

Chapter 6 — Attitude determination software simulation Implements and assesses the updated ADS using a realistic orbital simulation and hardware models;

Chapter 7 — Conclusion Summarises the research outcomes, answers the proposed research questions, and outlines future work.

Chapter 2

Literature review

This chapter performs a subject and a literature review, with a clear distinction between the theoretical background and state-of-the-art developments. Theoretical background is essential for framing the study, as it defines the principles and theories that underpin the research scope. This section will briefly explore already established concepts and paradigms that shape the discourse within each domain. The state-of-the-art section will present a synthesis of the most recent advancements and findings relevant within the research field. This review showcases the trajectory of research concerning the field, highlighting the current landscape.

2.1 Theoretical background

In this section, a summary of the key concepts of attitude determination and Sun sensoring is provided. Section 2.1.1 deals with the ADS, exploring its holistic context and purpose as a satellite system. It also overviews both deterministic and recursive methods employed and the sensors usually involved in the process. Section 2.1.2 describes the several mathematical forms in which spacecraft orientation can be represented, and Section 2.1.3 presents the frames of reference standard to the field and particularly subject of this dissertation. Lastly, Section 2.1.4 will delve into the basic principles that govern the spacecraft motion in space, and Section 2.1.5 explores the behaviour of Sun sensors in their various types and arrangements.

2.1.1 Attitude Determination Subsystem

Satellite design is ultimately divided into several distinct subsystems, such as structural & thermal, energy & power, telecommunications, command & data handling, and guidance, navigation & control. The last is partially composed by the ADCS, which fundamentally serves to provide pointing accuracy and stability for the payload or other critical components in order to fulfil the mission objective.

The ADCS can also be divided into two parts: the attitude determination and the attitude control subsystems. As the name tells, the first is solely responsible for finding the spacecraft orientation given some specific method and sensor inputs, while the second has to converge this orientation to a desired one, also with a specific algorithm and some form of actuation. This separation creates a closed loop, as shown in Figure 4, with a dynamics block integrated within, along with additional details such as examples of these methods and hardware components. This dissertation scope will be limited to the ADS and the strictly necessary dynamics since the conducted research also encompasses the development of the proposed Sun sensor.

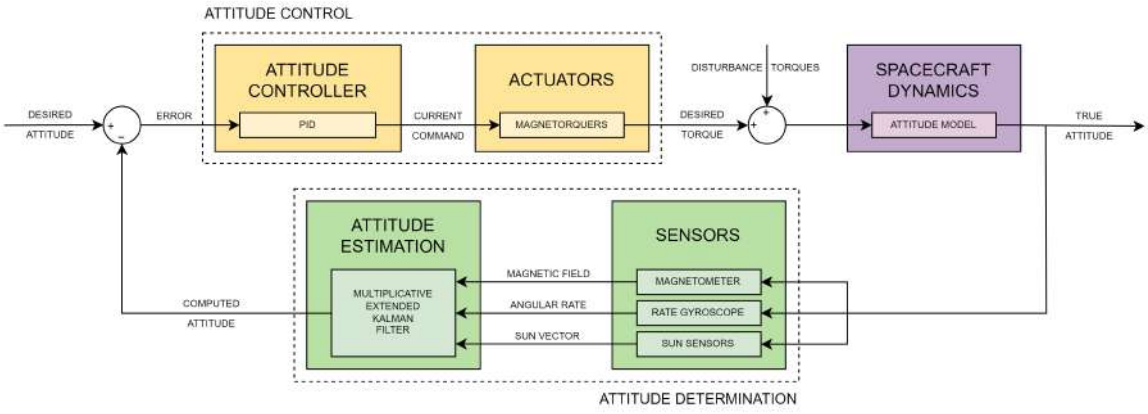


Figure 4: Typical ADCS block diagram.

The attitude of a spacecraft is a set of three independent quantities that indicate the rotational position of its body frame relative to an inertial reference frame. In a three-dimensional (3D) problem, at least two observation vectors will be required to satisfy this condition, analogically interpreted as one required to lock an axis direction and the other to define the rotation about that axis [9].

Attitude determination methods can roughly be divided into static and dynamic. Static methods are time-independent, meaning that all measurements are taken simultaneously or close enough in time that the spacecraft's motion between observations can be ignored or compensated for. This approach is deterministic and does not require information about past states. The challenge, therefore, is to solve the

geometry of these measurements in the body reference frame and compare them with their corresponding known descriptions in the inertial reference frame [9].

Dynamic attitude determination methods, on the other hand, consider the spacecraft's movement and are thus time-dependent and recursive. They treat measurements as prone to random noise rather than following a deterministic process, requiring a statistical approach. Filtering techniques, such as Kalman filters, are employed to process information from past and present measurements, which is crucial for understanding the spacecraft's motion and potential errors in the system's dynamic model. This behaviour is why such an approach is often called attitude "estimation" rather than "determination" due to the statistical nature of these methods [9].

Recursive attitude determination has two advantages over the static methods. First, the inclusion of information from the past generally provides more accurate solutions. Secondly, it can provide estimates when less than two measurements are available, which would be insufficient otherwise. Nonetheless, static methods tend to be simpler and faster.

2.1.1.1 Static methods

The TRI-axial Attitude Determination (TRIAD) algorithm was proposed in 1964 by Black [10] and was the first published method to determine the satellite attitude. Assuming the existence of two linearly independent reference directions $\check{r}_1$ and $\check{r}_2$ in the inertial frame $\mathcal{I}$, also known in body frame $\mathcal{B}$ as $\check{b}_1$ and $\check{b}_2$, respectively, there is one matrix A that rotates the first frame to the second [9].

This method begins by defining three orthogonal vectors (a triad) in both frames, given by

$$\check{t}_1 = \check{v}_1 \tag{1a}$$

$$\check{t}_2 = \frac{\check{v}_1 \times \check{v}_2}{\|\check{v}_1 \times \check{v}_2\|} \tag{1b}$$

$$\check{t}_3 = \check{t}_1 \times \check{t}_2 \tag{1c}$$

where $\check{t}_1$ will have the same direction as the most accurate of the two observation vectors, $\check{t}_2$ is the orthogonal vector to the plane defined by both observations, and $\check{t}_3$ is orthogonal to the first and second vectors [9], as depicted in Figure 5. Note that $\check{v}$ is substituted by $\check{b}$ and $\check{r}$ depending on the frame, enabling the definition of two matrices T given by

$${}^{\mathcal{B}}T = \begin{bmatrix} {}^{\mathcal{B}}\check{t}_1 & {}^{\mathcal{B}}\check{t}_2 & {}^{\mathcal{B}}\check{t}_3 \end{bmatrix} \tag{2a}$$

$${}^{\mathcal{I}}T = \begin{bmatrix} {}^{\mathcal{I}}\check{t}_1 & {}^{\mathcal{I}}\check{t}_2 & {}^{\mathcal{I}}\check{t}_3 \end{bmatrix} \tag{2b}$$

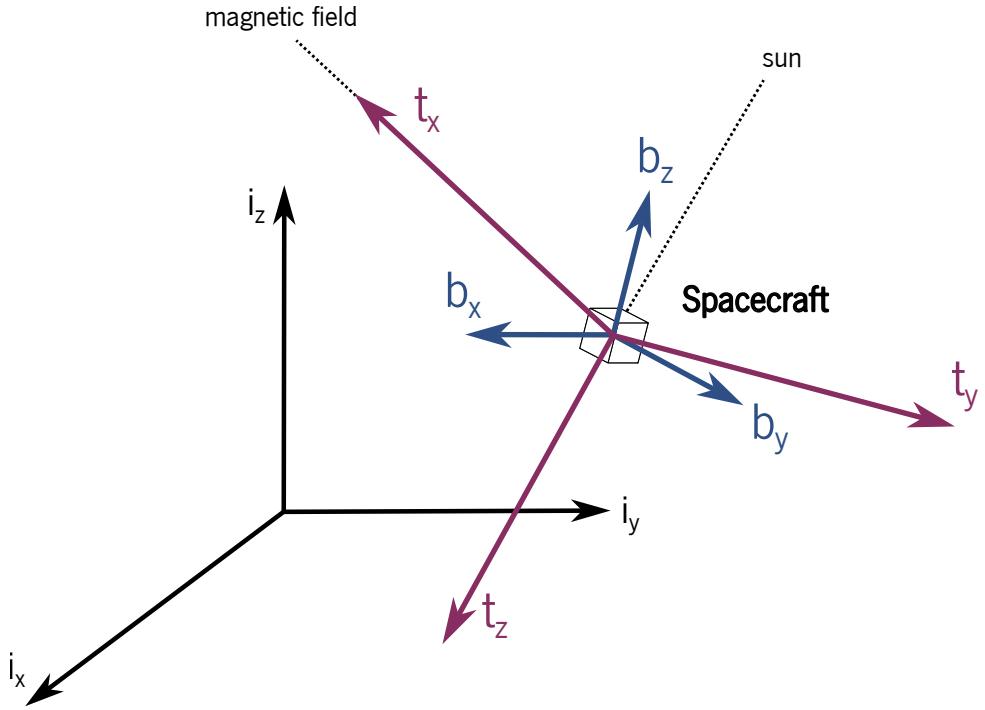


Figure 5: TRIAD frame constructed by two reference directions.

Since the TRIAD matrices are orthogonal, they define an intermediate frame between body and inertial frames [9], aiding the determination of attitude matrix with

$$A = {}^{\mathcal{B}}T\,[{}^{I}T]^{T} \tag{3}$$

The TRIAD algorithm is straightforward and intuitive but only allows two vectors to be used, with more weight given to the information from the first vector than to the second [9].

Grace Wahba proposed a problem in 1965 [11] where these issues could be fixed. He formulated the weighted Least Squares (LS) function J given by

$$J(A) = \sum_{n=1}^{N} a_n \|b_n - Ar_n\|^2 \tag{4}$$

where N is the total number of directions, n is the reference direction number and a_n is its associated non-negative weight, later conveniently proposed to be inversely proportional to the sensor variances. The attitude matrix A, vectors b and r are the same as the TRIAD method. By minimizing this loss function, one would obtain the most likely attitude matrix from all the available information.

Several solutions were offered almost immediately for solving this problem, none of which was suitable at the time for practical spacecraft applications [12]. Over the following decades, multiple remarkable proposals arose, such as Davenport's q-method in 1977 [13], the QUaternion ESTimator (QUEST) in 1981 [14],

Singular Value Decomposition (SVD) in 1988 [15], Fast Optimal Attitude Matrix (FOAM) in 1993 [16] and EStimators of the Optimal Quaternion (ESOQ) in 1997 [17].

The SVD method provides a simple yet very robust solution to the problem [15]. It starts by rewriting the loss function as

$$J(A) = \sum_{n=1}^{N} a_n - \mathrm{tr}(AB^T) \tag{5a}$$

$$B \equiv \sum_{n=1}^{N} a_n b_n r_n^T \tag{5b}$$

where $\mathrm{tr}(\square)$ is the matrix trace operator [15]. It is clear that $J(A)$ is minimized when the trace is maximized [15]. The SVD of the matrix B is then given by

$$B = U\Sigma V^T = U\,\mathrm{diag}[\Sigma_{11}\ \ \Sigma_{22}\ \ \Sigma_{33}]V^T \tag{6}$$

where U and V are orthogonal matrices, $\mathrm{diag}[\square]$ is the diagonal matrix operator, and the singular values obey the inequalities $\Sigma_{11} \geq \Sigma_{22} \geq \Sigma_{33} \geq 0$ [15]. The trace is invariant under cyclic permutation of its argument [15], so

$$\mathrm{tr}(AB^T) = \mathrm{tr}(U^T AV\,\mathrm{diag}[\Sigma_{11}\ \ \Sigma_{22}\ \ \Sigma_{33}]) \tag{7}$$

Therefore, the presented trace is maximized by the optimal attitude matrix [15] given by

$$A = U\,\mathrm{diag}[1\ \ 1\ \ \det(U)\det(V)]V^T \tag{8}$$

where $\det(\square)$ is the matrix determinant operator. Equation (8) ensures that $\det(A) = 1$, consistent with proper orthogonal rotation matrices [15]. Algorithm 1 describes all the steps mentioned above to find the optimal attitude matrix.

Algorithm 1 Singular value decomposition solution of Wahba's problem.

for all timesteps k **do**
  $B_k \leftarrow 0$ ▷ reset
  for all directions n **do**
    $B_k \leftarrow B_k + a_n b_n r_n^T$ ▷ iterate for all reference directions
  end for
  $\begin{bmatrix} U & \Sigma & V \end{bmatrix} \leftarrow \mathrm{svd}(B_k)$ ▷ decompose matrix singular values
  $d \leftarrow \det(U)\det(V)$ ▷ calculate determinants product
  $A_k \leftarrow U\,\mathrm{diag}[1\ \ 1\ \ d]V^T$ ▷ calculate optimal solution
end for
return A_k

2.1.1.2 Dynamic methods

Rudolph Kalman presented the Kalman filter in 1960 [18] and provides the optimal, Mean Square Error (MSE) estimate for a linear system under Gaussian noise [19], with implementations in distinct fields for numerous applications.

It recursively estimates the evolution of the system as a function of the current state estimate, control inputs and measurements, while keeping track of the state uncertainty [19]. This filter assumes that the system can be described by a linear model that is disturbed by two types of noise: process noise and measurement noise, both to be white (meaning their values are uncorrelated over time) and Gaussian (meaning they follow a normal distribution). Although true white noise does not exist in nature, adding many independent random effects typically results in a normal distribution, which simplifies the problem [19].

The optimal estimate of the true state x is given by

$$\hat{x} \equiv E\{x\} \tag{9}$$

where $E\{\square\}$ is the expectation operator, which roughly translates to the value of a function it should be expected to get by averaging it many random samples. The caret $\hat{\square}$ denotes an estimated value, regardless of the underlying probability distribution [19]. The estimate $\hat{x}$ is unbiased, meaning its expected value equals the true value being estimated, and it has the minimum variance among all linear unbiased estimates [19].

The variable x has an associated covariance matrix P, defined as

$$P \equiv E\{(x - \hat{x})(x - \hat{x})^T\} \tag{10}$$

which is symmetric and positive semi-definite. Its diagonal elements are equal to the variance of the elements of x, and its trace gives the state MSE [19].

Consider the discrete-time linear system model given by

$$x_{k+1} = \Phi_k x_k + \Gamma_k u_k + w_k, \quad w_k \sim N(0, Q_k) \tag{11a}$$

$$\tilde{y}_k = H_k x_k + v_k, \quad v_k \sim N(0, R_k) \tag{11b}$$

where k represents a specific time step, Φ_k is the state matrix, Γ_k is the input matrix, u_k is the input, H_k is the output matrix, w_k and v_k are uncorrelated zero-mean Gaussian white noise processes with Q_k and R_k covariances [19], respectively, here described using the normal distribution function N.

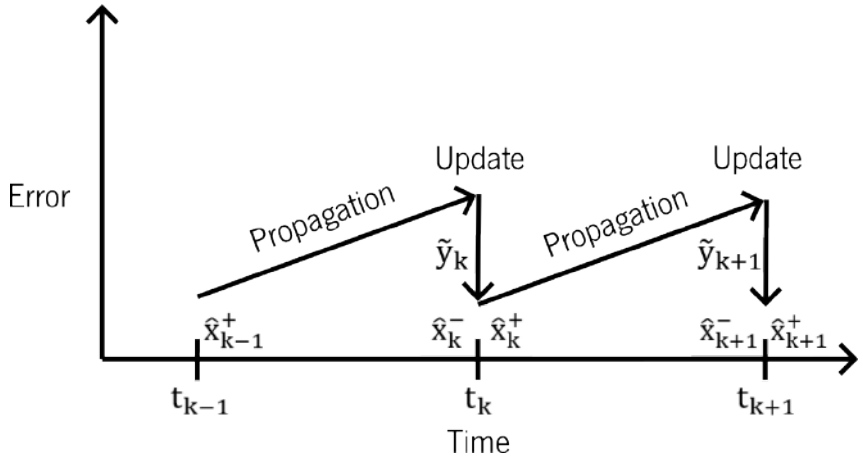


Figure 6: Kalman filter simplified operation diagram.

The filter operation is simplified in Figure 6 and can be described as the following: it begins with an initialization stage, where an initial state $\hat{x}_0$, and its corresponding covariance matrix P_0 are established. After this, the filter operates recursively in two alternating stages. First is the propagation (or prediction) step, where the estimate is projected from time t_{k-1} to t_k, yielding the predicted state $\hat{x}_k^-$ and covariance P_k^- based on the process model. Next comes the update (or correction) step, during which the *a priori* estimate and its covariance are refined using a new measurement $\tilde{y}_k$. In the so-called innovation process, a residual z_k between the real measurement and its predicted value is computed and used to update the *a posteriori* state $\hat{x}_k^+$ and covariance P_k^+ for time t_k. For clarity, the superscripts $\square^-$ and $\square^+$ denote values computed before and after this update, respectively. These estimates are then fed back into the first step to predict the *a priori* values for the next time step t_{k+1}. Since $\hat{x}$ minimizes the trace of P at each iteration, in the calculation of the Kalman gain K_k, $\hat{x}$ is the optimal estimate of the variables contained in x [19].

The complete and thorough derivation of the filter can be observed in many sources, such as the one made by Crassidis and Junkins in [20], but the required equations to design the filter as described previously are presented in Algorithm 2.

Algorithm 2 Discrete-time Kalman filter.

$\hat{x}_0^+, P_0^+$ ▷ initialize state estimate and covariance
for all timesteps k **do**
  $\hat{x}_k^- \leftarrow \Phi_{k-1}\hat{x}_{k-1}^+ + \Gamma_{k-1}u_{k-1}$ ▷ predict state estimate
  $P_k^- \leftarrow \Phi_{k-1}P_{k-1}^+\Phi_{k-1}^T + Q_{k-1}$ ▷ predict estimate covariance
  $z_k \leftarrow \tilde{y}_k - H_k\hat{x}_k^-$ ▷ calculate measurement residual
  $K_k \leftarrow P_k^- H_k^T(H_kP_k^-H_k^T + R_k)$ ▷ calculate Kalman gain
  $\Delta x_k^+ \leftarrow K_k z_k$ ▷ calculate estimation error
  $\hat{x}_k^+ \leftarrow \hat{x}_k^- + \Delta x_k^+$ ▷ correct state estimate
  $P_k^+ \leftarrow (I - K_kH_k)P_k^-$ ▷ correct estimate covariance
end for
return $\hat{x}_k^+, P_k^+$

This time, consider the discrete-time *nonlinear* system model given by

$$x_{k+1} = f_k(x_k, u_k) + w_k, \quad w_k \sim N(0, Q_k) \tag{12a}$$

$$\tilde{y}_k = h_k(x_k) + v_k, \quad v_k \sim N(0, R_k) \tag{12b}$$

where $f_k(x_k, u_k)$ and $h_k(x_k)$ are assumed to be continuously differentiable nonlinear functions representing the state and output behaviour, respectively. As stated by Crassidis and Junkins in [20], the problem with this nonlinear model is that a Gaussian input does not necessarily produce a Gaussian output, unlike the linear case. To solve this, the Extended Kalman Filter (EKF) surfaced around 1962 [21] and assumes that the true state is sufficiently close to the estimated state [20]. Therefore, the error dynamics can be represented fairly accurately by a linearized first-order Taylor series expansion about the estimate $\hat{x}_k$ [20] given by

$$f_k(x_k, u_k) \approx f_k(\hat{x}_k, u_k) + \left.\frac{\partial f_k}{\partial x}\right|_{\hat{x}_k, u_k}(x_k - \hat{x}_k) \tag{13a}$$

$$h_k(x_k) \approx h_k(\hat{x}_k) + \left.\frac{\partial h_k}{\partial x}\right|_{\hat{x}_k}(x_k - \hat{x}_k) \tag{13b}$$

where $\partial\square/\partial x|_{\hat{x}_k}$ represents a partial derivative with respect to (w.r.t.). x evaluated at $\hat{x}_k$. After some manipulation, the system model is now linear in terms of the state error Δx, as well as the measurement residual z [20], both given by

$$\Delta x_{k+1} = F_k\Delta x_k + w_k, \quad F_k \equiv \left.\frac{\partial f_k}{\partial x}\right|_{\hat{x}_k, u_k} \tag{14a}$$

$$z_k = H_k\Delta x_k + v_k, \quad H_k \equiv \left.\frac{\partial h_k}{\partial x}\right|_{\hat{x}_k} \tag{14b}$$

where $\Delta x_k = x_k - \hat{x}_k$ and $z_k = \tilde{y}_k - h_k(\hat{x}_k)$, F_k and H_k are the Jacobian matrices of the state and output functions, respectively, w.r.t.. x evaluated at $\hat{x}_k$.

Assuming that Δx_k obeys the Equation (14b), the theory from the linear Kalman filter can be applied, although formulated to estimate the small error deviations from a nominal state, rather than the full state itself [20]. This formulation implies that the nominal state ignores noise and imperfections, causing errors to accumulate, and only at the correction step are these deviations incorporated, as shown on Algorithm 3. This time, the covariance matrix is updated using Joseph's form on line 8, ensuring symmetry and positive definiteness, thus improving numerical stability [20].

Algorithm 3 Discrete-time extended Kalman filter with Joseph's form.

1 $\hat{x}_0^+, P_0^+$ ▷ initialize state estimate and covariance
2 **for all** timesteps k **do**
3 $\quad \hat{x}_k^- \leftarrow f_{k-1}(\hat{x}_{k-1}^+, u_{k-1})$ ▷ predict state estimate
4 $\quad P_k^- \leftarrow F_{k-1} P_{k-1}^+ F_{k-1}^T + Q_{k-1}$ ▷ predict estimate covariance
5 $\quad z_k \leftarrow \tilde{y}_k - h(\hat{x}_k^-)$ ▷ calculate measurement residual
6 $\quad K_k \leftarrow P_k^- H_k^T (H_k P_k^- H_k^T + R_k)$ ▷ calculate Kalman gain
7 $\quad \Delta x_k^+ \leftarrow K_k z_k$ ▷ calculate estimation error
8 $\quad \hat{x}_k^+ \leftarrow \hat{x}_k^- + \Delta x_k^+$ ▷ correct state estimate
9 $\quad P_k^+ \leftarrow (I - K_k H_k) P_k^- (I - K_k H_k)^T + K_k R_k K_k^T$ ▷ correct estimate covariance
10 **end for**
11 **return** $\hat{x}_k^+, P_k^+$

Considering the above, the EKF formulation for attitude is further discussed in Section 2.2.3.

2.1.1.3 Sensors

Attitude determination relies on enough observable information obtained from measurements of the spacecraft environment, collected from a wide range of sensors. Commonly, the Earth's magnetic field and the Sun vectors are employed to provide a good, cost-effective attitude, but other sensors are often used because of tighter pointing requirements or system constraints.

As detailed on Wertz's Satellite Mission Analysis and Design book [22], specifically on Chapter 11.1, sun sensors are visible-light or infrared detectors that measure angles between incident sunlight and their mounting base. They are widely used due to their reliability and accuracy, often below 0.01 degrees, but they require an unobstructed FOV and cannot function during eclipse periods. They are often mounted at vehicle ends to avoid obstructions, though structural bending may affect precision. CSSs, which employ

small solar cells or photodiodes, are commonly used for redundancy and low-power fault recovery, with reduced accuracy. Moreover, it is also common to use solar panels for this effort.

Still as mentioned on [22], star sensors have become the most common high-accuracy sensors, functioning as either trackers or scanners. Star trackers, which use Charge-Coupled Devices (CCDs) or advanced photosystems imaging to match observed star fields with catalogues of expected visible stars, provide highly accurate three-axis attitude data. However, they require clear visibility and a stable spacecraft. To mitigate limitations, they are often paired with gyroscopes, which help during occlusions or interference.

Horizon sensors detect the contrast between Earth's atmosphere and deep space to determine attitude [22]. They are generally imaging sensors where an infrared image is post-processed to differentiate the radiation coming from Earth from the surrounding atmosphere [23]. These are often used on spinning spacecraft, with scanning horizon sensors providing nadir-pointing stability. While effective, they require clear fields of view and may need correction for seasonal or oblateness variations.

Magnetometers measure Earth's magnetic field to infer spacecraft orientation. They are lightweight and low-power but less accurate than star or horizon sensors. While complex magnetic field sensors based on the Josephson junction have been built, the simpler versions rely on Faraday's law effects on induction coils [23] or magnetoresistance variations. Their usage is limited to Low Earth Orbit (LEO) since the magnetic field strength decreases significantly with the altitude [22]. As expected, Eletromagnetic Interference (EMI) from other parts of the spacecraft can perturb the measurements and extensible booms are often used to mitigate this. Microelectromechanical System (MEMS)-based sensors are also available.

GNSS receivers, while primarily used for navigation, can also determine attitude if multiple antennas are placed with sufficient separation. Orientation can be determined based on the phase differences between signals received by the different antennas [23]. They are a possible alternative for LEO missions, achieving accuracies of 0.1 to 0.5 degrees, but can be either tedious to obtain, due to Coordinating Committee for Multilateral Export Controls (COCOM) regulations [24, 25], or expensive, as its use license can be purchased but not at a low cost.

Gyroscopes, including mechanical and MEMS-based versions, provide high-frequency inertial reference data. MEMS usually employ rate gyros that measure angular rate instead of orientation by measuring Coriolis-induced displacements of a vibrating mass. However, they suffer from several types of biases, requiring periodic calibration [22].

Overall, an effective attitude determination system often integrates multiple sensors to compensate for individual limitations, ensuring robustness and accuracy across various mission conditions. Table 1

summarizes performance, mass, power and cost requirements for each type.

Table 1: Typical specifications of sensors used for attitude determination (adapted from [22]).

Type	Performance	Mass (kg)	Power (W)	Cost
Sun sensor	0.005° - 3° [a]	<0.01 - 2	<0.1 - 3	COTS components are cheap
Magnetometer	0.5° - 3° [a]	<0.01 - 1.2	<0.1 - 1	COTS components are cheap
Gyroscope	0.003°/hr - 1°/hr [b]	<0.01 - 15	<0.1 - 200	COTS components are cheap
GNSS receiver	0.1° - 1° [a]	<0.01 - 3	0.1 - 10	Overall moderate
Star tracker	0.0003° - 0.001° [a]	2 - 5	5 - 20	Overall expensive
Horizon sensor	<0.1° - 1° [a]	0.5 - 4	0.3 - 10	Overall expensive

[a] angular deviation
[b] drift rate

2.1.2 Attitude parametrization

It is clear by now that spacecraft attitude is simply a rotation. However, this simplicity quickly vanishes once one starts to define it. In this matter, three basic choices can be made: the operator, the function and the direction of rotations [26].

There are several ways to represent rotations. The rotation operators can be matrices, vectors, or other more complex notations, such as quaternions or Modified Rodrigues Parameters (MRP) [27]. These are mainly characterized by how many parameters have, and consequently the number of constraints, but also their singularities. It is obvious that one would want the least of both. The most common representations will be presented in the remainder of the section, with some important algebraic rules.

The function of the rotation can be distinguished into two types: active (*alibi*) transformations, which change the physical position of vectors relative to a fixed frame of reference; or passive (*alias*), which leave vectors fixed but change the frame of reference to which they are described [28]. It is trivial to note that the rotational effect is similar, but their meaning is different. One can also transform one into the other by inverting the operator. The passive interpretation is most natural when knowledge of the direction of successive rotations is known in the local frame and when describing the motion of an object by its own, which is the case in attitude determination [28].

Inherent to passive transformations, a second source of interpretation is related to the direction in which the operation happens, either converting from local to global[1] frames or the opposite [26]. Again, one rotation is the inverse of the other. Regarding this binary choice, both sides have popularity in this field. It seems reasonable to preserve each author's preference when describing their contributions. For

[1] Other common denominations for the {local, global} frames are {body, inertial}

example, in Section 2.1.1.1, it is clear that attitude is defined as a matrix rotating from global to local frames. However, the opposite convention is preferred by the author of this dissertation, which will be discussed in the section.

2.1.2.1 Rotation matrices

The most fundamental representation of attitude is a 3x3 orthogonal matrix, whose inverse is equal to its transpose [27]. Such a matrix is also referred to as *proper* because its determinant is +1. These matrices form the special orthogonal group SO(3), which defines all possible rotations about the origin in a 3D Euclidean space [27].

The Direction Cosine Matrix (DCM) is a rotation matrix that specifically describes the orientation of one coordinate system relative to another, thus being a passive transformation [26]. It is constructed with Equation (15), where each component c_{ij} is the cosine of the angle between the axis i in the source frame $\mathcal{S}$ and the axis j in the target frame $\mathcal{T}$ [26]. The superscript notation was added to matrix A for clarification.

$$ {}^{\mathcal{T}}A^{\mathcal{S}} = \begin{bmatrix} c_{xx} & c_{yx} & c_{zx} \\ c_{xy} & c_{yy} & c_{zy} \\ c_{xz} & c_{yz} & c_{zz} \end{bmatrix} \tag{15} $$

As said before, the opposite rotation, from frame $\mathcal{T}$ to frame $\mathcal{S}$, is given by the matrix transpose. An additional rotation through an intermediate frame $\mathcal{U}$ is given by matrix multiplication, which is not commutative [26].

$$ {}^{\mathcal{T}}A^{\mathcal{S}} = \left[{}^{\mathcal{S}}A^{\mathcal{T}} \right]^{T} = {}^{\mathcal{T}}A^{\mathcal{U}} \left[{}^{\mathcal{U}}A^{\mathcal{S}} \right] \tag{16} $$

The DCM has no singularities and facilitates the combination of multiple rotations through the product of matrices. However, it is clearly redundant and inefficient since it requires nine parameters to represent a rotation with three degrees of freedom (3-DOF).

2.1.2.2 Rotation vectors

Euler's Theorem states that the most general motion of a rigid body with one point fixed is a rotation by an angle about some axis [27]. This axis is specified by a unit vector u and combines the angle θ into a rotation vector so that $\psi \equiv \theta u$ [27]. This parametrization is also called the axis-angle vector if both terms are treated as separate. The rotation matrix can be formulated as a function of the rotation vector with the

Rodrigues rotation formula or the exponential map [26] given by

$$A(\theta, u) = I_3 + (\sin\theta)[u]_\times + (1 - \cos\theta)[u]_\times^2 = \exp([\theta u]_\times) \tag{17}$$

where I_3 is the 3x3 identity matrix and $[\square]_\times$ is the skew matrix operator where

$$[v]_\times = \begin{bmatrix} 0 & -v_3 & v_2 \\ v_3 & 0 & -v_1 \\ -v_2 & v_1 & 0 \end{bmatrix} \tag{18}$$

Despite being very intuitive to use and only having one redundant parameter, it is not useful for large rotations, mainly because it obscures the periodicity of the attitude matrix as a function of the rotation vector [27]. Additionally, it is singular when $\sin\theta = 0$ because the rotation axis becomes undefined [27].

2.1.2.3 Quaternions

A quaternion $q \in \mathbb{H}$ is a mathematical construct consisting of a scalar part and a vector part [26]. In addition to the choices outlined earlier in this section, two more conventions must be specified to proceed with its definition: the order of its elements and its handedness [26].

The order determines whether the scalar part appears first or last in the quaternion representation. While this choice is not fundamentally significant, maintaining consistency is crucial to prevent implementation errors [26].

The handedness refers to the convention used for quaternion multiplication and rotation, analogous to left-handed and right-handed coordinate systems in 3D space. This choice affects how quaternions represent rotations and interact with vectors, making it essential for ensuring correct transformations [26]. The function and the direction of rotations obviously apply to quaternions as well.

Regarding this set of choices, there are nowadays two popular conventions: the Hamilton convention, which describes the quaternion as it was originally defined and is common across many scientific fields; and the Jet Propulsion Laboratory (JPL) convention, which became popular in the aerospace domain [26]. Table 2 resumes the differences.

Table 2: Hamilton vs. JPL quaternion conventions (adapted from [26]).

Parameter	Convention	
	Hamilton	JPL
Order	Scalar first	Scalar last
Handedness	Right-handed	Left-handed
Function	Passive	Passive
Direction	Local-to-global	Global-to-local

Folowing the Hamilton convention, which is arguably the best practise [29], the quarternion is defined in the form of $q = q_s + q_x i + q_y j + q_z k$, where $\{q_s, q_x, q_y, q_z\} \in \mathbb{R}$ and $\{i, j, k\}$ are three imaginary unit numbers that $i^2 = j^2 = k^2 = ijk = -1$ [26]. As mentioned before, the quaternion is usually represented with its scalar $q_s \in \mathbb{R}$ and vector $q_v \in \mathbb{R}^3$ parts, that form the vector of Equation (19) in $\mathbb{R}^4$ [26].

$$q \equiv \begin{bmatrix} q_s \\ q_v \end{bmatrix} \equiv \begin{bmatrix} q_s \\ q_x \\ q_y \\ q_z \end{bmatrix} \tag{19}$$

Unit quaternions reside on the 3D sphere $\mathbb{S}^3$ embedded in $\mathbb{R}^4$ and present a double-cover of the rotation group $SO(3)$, since q and $-q$ represent the same rotation, thus expressing partial isomorphism [27].

Quaternion multiplication is generally not commutative, following the rotation matrices algebraic rules, and it is usually represented with the $\otimes$ operator, defined with quaternion product matrices [26] as

$$q_1 \otimes q_2 = [q_1]_L \, q_2 \tag{20a}$$

$$[q]_L \equiv \begin{bmatrix} q_s & -q_v^T \\ q_v & q_s I_3 + [q_v]_\times \end{bmatrix} \tag{20b}$$

The unit quaternions $\in \mathbb{S}^3$ have an exponential map from rotation vectors $\in \mathbb{R}^3$, extended from the Euler formula, which is given by Equation (21) and can be converted to rotation matrices with Equation (22) [26].

$$q(\theta, u) = \begin{bmatrix} \cos\frac{\theta}{2} \\ u \sin\frac{\theta}{2} \end{bmatrix} \tag{21}$$

$$A(q) = (q_s^2 - q_v^T q_v) I_3 + 2 q_v q_v^T + 2 q_s [q_v]_\times \tag{22}$$

As it will be clear later, sometimes it is helpful to create a quaternion with zero scalar part from a vector $v \in \mathbb{R}^3$ [26], denoted with $\check{\square}$ operator, which gives

$$\check{v} = \begin{bmatrix} 0 \\ \\ I_3 \end{bmatrix} v \tag{23}$$

The quaternion is an ideal global attitude representation since it varies continuously over $\mathbb{S}^3$ as the attitude changes, avoiding the jumps required by some three-dimensional parameterizations. It has four parameters, leaving it with only one constraint [27].

2.1.2.4 Modified Rodrigues Parameters

The MRP are viewed as a three-dimensional stereographic projection of the four-dimensional quaternion sphere [27]. Its defining vector p is given by Equation (24), and its conversion to rotation matrix is given by Equation (25).

$$p = \frac{q_v}{1 + q_s} = \frac{u \sin(\theta/2)}{1 + \cos(\theta/2)} \tag{24}$$

$$A(p) = I + 4(1 + p^2)^{-2}[(1 - p^2)[p]_\times - 2[p]_\times^2] \tag{25}$$

The MRPs provide a continuous representation of rotations if the angle is bounded and less than 2π [27]. Compared to the Euler axis-angle vector, the Modified Rodrigues vector also has three parameters but lacks intuition.

2.1.3 Reference frames

As discussed previously, reference frames are inevitable in the context of attitude determination and by now, only two were mentioned: the body frame and the inertial frame, which are fundamental. However, others are recommended for more specific purposes, such as fixed or orbit-defined frames. The reference frames needed for this dissertation are presented in this section and are represented in Figure 7.

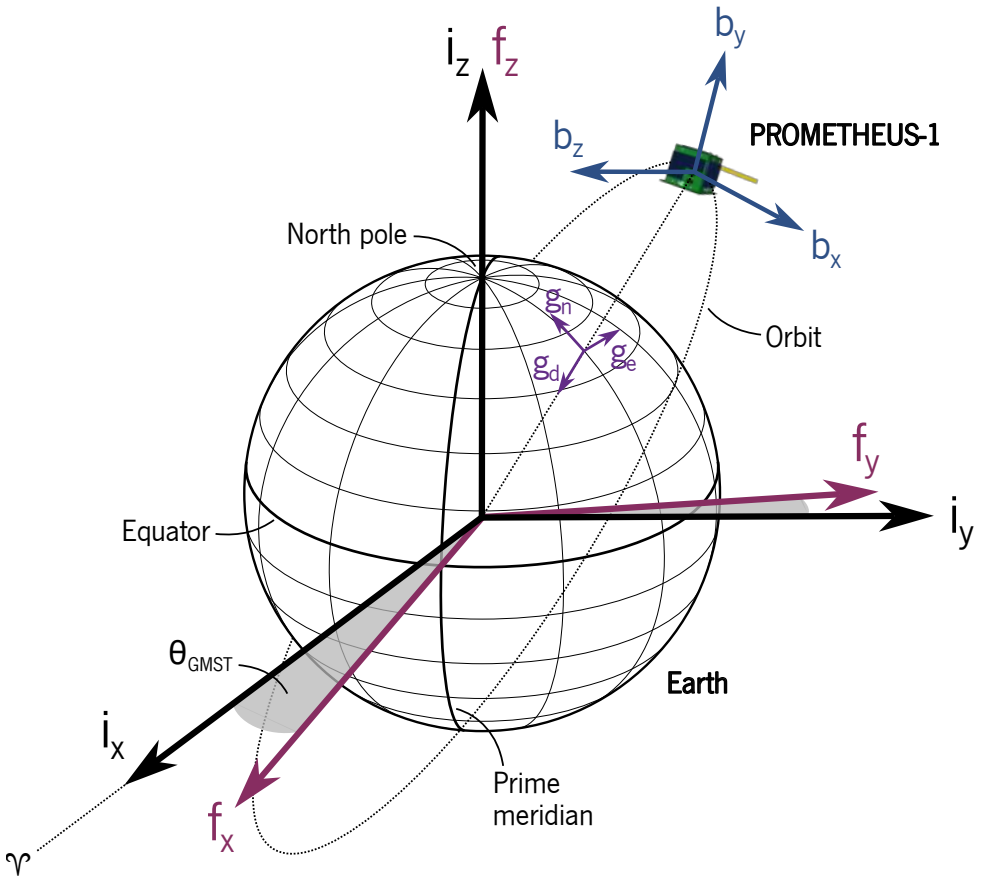


Figure 7: Definition of necessary reference frames (adapted from [30]).

2.1.3.1 Spacecraft body frame

The spacecraft body frame is designated by $\mathcal{B} = \{b_x, b_y, b_z\}$. Its origin is typically fixed at the centre of mass of the spacecraft, and the axes rotate with it [30]. The convention orientation depends on the spacecraft manufacturer, but on PROMETHEUS-1, the b_z points in the opposite direction of the antenna boresight, b_y points in the opposite direction of the camera boresight, and b_x completes the right-handed coordinate system. This triad is also represented on Figure 1.

2.1.3.2 Earth-Centered Inertial frame

The Earth-Centered Inertial (ECI) reference frame, denoted as $\mathcal{I} = \{i_x, i_y, i_z\}$, is the most common non-rotating frame for attitude. The axis i_x is aligned with the vernal equinox direction, which represents the intersection of the Earth's equatorial plane and the plane of its orbit around the Sun [30]. This direction corresponds to the Sun's position relative to the Earth at the start of spring. The i_z axis points toward the Earth's North Pole, while the i_y axis is defined to complete the right-handed orthogonal triad, which also lands on the equator [30]. Since neither the polar axis nor the vernal equinox direction remains truly inertial

over time, the ECI frame is conventionally defined based on their mean orientations at a fixed epoch [30]. The standard epoch currently in use is J2000 [30] and was also adopted in this dissertation.

Defining an inertial reference frame is essential because it simplifies the equations of motion governing orbital dynamics [30]. Although the ECI frame is not perfectly inertial, since the Earth's centre also experiences acceleration in its orbit around the Sun, it is often treated as inertial for practical purposes [30]. In most applications, this assumption introduces negligible error.

2.1.3.3 Earth-Centered Earth-Fixed frame

The Earth-Centered Earth-Fixed (ECEF) reference frame, denoted as $\mathcal{F} = \{f_x, f_y, f_z\}$, is a coordinate system with its origin fixed at the centre of the Earth [30]. Unlike the ECI frame, the ECEF frame rotates with the Earth, making it suitable for describing positions relative to the planet's surface [30]. Similarly to ECI, the f_z axis is aligned with the Earth's North pole, f_x points in the direction of Earth's prime meridian, and f_y completes the right-handed orthogonal triad [30]. The rotation is quantified by the Greenwich Mean Sidereal Time angle θ_{GMST} [30].

The rotation angle was calculated based on IERS 2003 Convention [31] and is given by

$$\theta_{GMST} = \mathrm{mod}\{2\pi(\mathrm{rem}\{t, 1.0\} + 0.7790572732640 + 0.00273781191135448t), 2\pi\} \tag{26}$$

where $\mathrm{mod}\{\square, \square\}$ and $\mathrm{rem}\{\square, \square\}$ are the modulo and remainder operators, respectively, and t is the time elapsed since J2000 epoch in Julian Universal Time 1 (UT1) format. This model discards precession, nutation and polar motion, which introduces an error in the order of microradians. Additionally, Coordinated Universal Time (UTC) was used for simplicity due to varying differences of below one second.

One can then go from ECI to ECEF frames by rotating around their z axis, with a rotation matrix given by

$$^{\mathcal{F}}A^{\mathcal{I}} = \begin{bmatrix} \cos\theta_{GMST} & \sin\theta_{GMST} & 0 \\ -\sin\theta_{GMST} & \cos\theta_{GMST} & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{27}$$

2.1.3.4 North-East-Down frame

When working with Earth's magnetic field models, there is a need to associate a vector with each point in a spherical coordinate system [32]. To do so, it is convenient to define an orthogonal coordinate system whose three axes are each parallel to the change in one of the three spherical coordinates at each point

in space [32]. Such systems are called local tangents since the reference plane at any point is always tangent to the sphere centred on the origin of the system and passing through the point in question [32].

The North-East-Down (NED) frame, denoted as $\mathcal{G} = \{g_n, g_e, g_d\}$, is defined with: the north axis in the direction of increasing latitude λ; east axis in the direction of increasing longitude ϕ; and nadir axis in the direction of decreasing altitude; at some arbitrary point on an Earth's geodetic model [32], such as the WGS84.

The rotation from NED to ECEF is then given by

$$ {}^{\mathcal{F}}A^{\mathcal{G}} = \begin{bmatrix} -\sin\lambda\cos\phi & -\sin\phi & -\cos\lambda\cos\phi \\ -\sin\lambda\sin\phi & \cos\phi & -\cos\lambda\sin\phi \\ \cos\lambda & 0 & -\sin\lambda \end{bmatrix} \tag{28} $$

2.1.4 Spacecraft mechanics

Spacecraft mechanics relates to the study of the motion of spacecraft operating in an environment subject to forces and torques. It encompasses the principles of attitude dynamics, attitude kinematics, orbital mechanics, propulsion and perturbations acting on a spacecraft [30]. This dissertation scope will only concern the first two subjects related to attitude, detailed in the following sections. The distinction between attitude kinematics and dynamics is that kinematics covers those aspects of motion that can be analyzed without consideration of forces or torques. The realm of dynamics begins when forces and torques are introduced [30].

2.1.4.1 Attitude kinematics

The time dependence of the attitude matrix expressing the rotation from source frame $\mathcal{S}$ to a target frame $\mathcal{T}$ is given by the fundamental definition of a derivative [30] as

$$ {}^{\mathcal{T}}\dot{A}^{\mathcal{S}}(t) = \lim_{\Delta t \to 0} \frac{{}^{\mathcal{T}}A^{\mathcal{S}}(t+\Delta t) - {}^{\mathcal{T}}A^{\mathcal{S}}(t)}{\Delta t} = \lim_{\Delta t \to 0} \frac{{}^{\mathcal{T}}A^{\mathcal{S}}(t+\Delta t)\,{}^{\mathcal{S}}A^{\mathcal{T}}(t) - I_3}{\Delta t}\,{}^{\mathcal{T}}A^{\mathcal{S}}(t) \tag{29} $$

As Δt goes to zero, the product ${}^{\mathcal{T}}A^{\mathcal{S}}(t+\Delta t)\,{}^{\mathcal{S}}A^{\mathcal{T}}(t)$ differs from the identity matrix by a small rotation, represented by a rotation vector $\Delta^{\mathcal{T}}\psi^{\mathcal{S}}_{\mathcal{T}}(t)$ in target frame $\mathcal{T}$ [30], so that

$$ {}^{\mathcal{T}}A^{\mathcal{S}}(t+\Delta t)\,{}^{\mathcal{S}}A^{\mathcal{T}}(t) = \exp(-[\Delta^{\mathcal{T}}\psi^{\mathcal{S}}_{\mathcal{T}}(t)]_\times) \approx I_3 - [\Delta^{\mathcal{T}}\psi^{\mathcal{S}}_{\mathcal{T}}(t)]_\times \tag{30} $$

Substituting Equation (30), where the approximation comes from the residual impact of higher-order terms in small rotations, into Equation (29), the fundamental equation of rigid-body attitude kinematics is

obtained [30], given by

$$ {}^{\mathcal{T}}\dot{A}^{\mathcal{S}}(t) = \lim_{\Delta t \to 0} \frac{-[\Delta {}^{\mathcal{T}}\psi^{\mathcal{S}}_{\mathcal{T}}(t)]_\times}{\Delta t} {}^{\mathcal{T}}A^{\mathcal{S}}(t) = -[{}^{\mathcal{T}}\omega^{\mathcal{S}}_{\mathcal{T}}(t)]_\times {}^{\mathcal{T}}A^{\mathcal{S}}(t) \tag{31} $$

where ${}^{\mathcal{T}}\omega^{\mathcal{S}}_{\mathcal{T}}(t)$ is the angular velocity vector, defined by the limit of $\Delta {}^{\mathcal{T}}\psi^{\mathcal{S}}_{\mathcal{T}}(t)/\Delta t$, as Δt approaches zero.

Similarly, for quaternions, the attitude kinematics is given by Equation (32) [30]. Replacing these frames, by a rotation from body frame $\mathcal{B}$ to inertial frame $\mathcal{I}$, as described in Section 2.1.2.3, and omitting the time arguments [26], the expression becomes

$$ {}^{\mathcal{I}}\dot{q}^{\mathcal{B}} = \frac{1}{2} \begin{bmatrix} 0 \\ {}^{\mathcal{I}}\omega^{\mathcal{B}}_{\mathcal{I}} \end{bmatrix} \otimes {}^{\mathcal{I}}q^{\mathcal{B}} \tag{32} $$

However, since the angular velocity will be obtained in the origin frame $\mathcal{B}$, the order of terms must be switched [26]. Additionally simplifying with Equation (23) yields

$$ \dot{q} = \frac{1}{2} q \otimes \check{\omega}_{\mathcal{B}} \tag{33} $$

2.1.4.2 Attitude dynamics

Consider the representations of the same vector v in two different frames, given by Equation (34). Its time derivative can be derived as given by Equation (35) [30], which is further simplified by Equation (31).

$$ v_{\mathcal{T}} = {}^{\mathcal{T}}A^{\mathcal{S}} v_{\mathcal{S}} \tag{34} $$

$$ \dot{v}_{\mathcal{T}} = {}^{\mathcal{T}}A^{\mathcal{S}} \dot{v}_{\mathcal{S}} + {}^{\mathcal{T}}\dot{A}^{\mathcal{S}} v_{\mathcal{S}} = {}^{\mathcal{T}}A^{\mathcal{S}} \dot{v}_{\mathcal{S}} - [{}^{\mathcal{T}}\omega^{\mathcal{S}}_{\mathcal{T}}]_\times {}^{\mathcal{T}}A^{\mathcal{S}} v_{\mathcal{S}} = {}^{\mathcal{T}}A^{\mathcal{S}} \dot{v}_{\mathcal{S}} - {}^{\mathcal{T}}\omega^{\mathcal{S}}_{\mathcal{T}} \times v_{\mathcal{T}} \tag{35} $$

Euler's second law states that the time rate of change of angular momentum h of a rigid body rotating about any axis is equal to the external torques τ applied to the body about that axis, in a fixed frame of reference [30], which translates to

$$ \dot{h} = \tau \tag{36} $$

Rewriting the angular momentum in terms of Equation (35), and replacing the frames by the body frame $\mathcal{B}$ and the inertial frame $\mathcal{I}$ [30], one can elaborate that

$$ \dot{h}_{\mathcal{B}} = {}^{\mathcal{B}}A^{\mathcal{I}} \dot{h}_{\mathcal{I}} - {}^{\mathcal{B}}\omega^{\mathcal{I}}_{\mathcal{B}} \times h_{\mathcal{B}} \tag{37} $$

A single expression for angular momentum h can be deduced w.r.t. the moment of inertia tensor J and angular velocity ω, all defined in the same frame, and it is given by Equation (38), while its derivative by Equation (39) [30].

$$ h = J\omega \tag{38} $$

$$\dot{h} = \dot{J}\omega + J\dot{\omega} = J\dot{\omega} \tag{39}$$

Next, starting from Equation (37), Equations (34), (36), (38) and (39) can be sequentially plugged together, w.r.t. to the same frame [30], which results to

$$J_{\mathcal{B}}\dot{{}^{\mathcal{B}}\omega^{\mathcal{I}}_{\mathcal{B}}} = \tau_{\mathcal{B}} - {}^{\mathcal{B}}\omega^{\mathcal{I}}_{\mathcal{B}} \times J_{\mathcal{B}}{}^{\mathcal{B}}\omega^{\mathcal{I}}_{\mathcal{B}} \tag{40}$$

Writing the equation in terms of the time rate of angular velocity in body frame $\mathcal{B}$ [30], finally results in

$$\dot{\omega}_{\mathcal{B}} = J^{-1}(\tau_{\mathcal{B}} - \omega_{\mathcal{B}} \times J\omega_{\mathcal{B}}) \tag{41}$$

2.1.5 Sun sensors

Sun sensors are the most common input used in satellites for attitude determination [32]. The Sun is easy to detect without expensive equipment due to its high brightness compared with other celestial bodies, which in turn offers a reliable reference direction for attitude determination [32]. As described earlier in Section 2.1.1.3, Sun sensors present some limitations: they require an unobstructed field of view, either by other components present in the spacecraft or by the Earth during the eclipse. Furthermore, Earth's albedo irradiance, which can be up to 40% of the solar irradiance [33], also corrupts the measurements.

Before proceeding, a fundamental distinction must be made between what is a photoelectric transducer and a Sun sensor. The first is defined as the component that directly interacts with the incoming radiation and converts it into an electrical signal. At the same time, the second is a composition of one or more transducers designed to provide comprehensive information about the Sun's orientation. For example, a photodiode is a very common transducer used as a Sun sensor. However, multiple photodiodes are often grouped in the same structure, which is also called a Sun sensor, sometimes without the due mention of being an array. This separation helps in clarifying the topic, but also in defining each element's specific properties and those that can be shared.

Sun sensors can be categorized by several properties, some of which are closely related but still worth separation since the meaning is distinct. These properties were compiled in Figure 8 and are divided by their signal behaviour, their accuracy level, the type of transducer they employ, their structure typology, and vector determination [34]. Other properties, such as the number of transducers, data transmission types, or manufacturing, could also be observed but were discarded because of lower relevance.

Regarding their behaviour, digital Sun sensors produce a discrete output signal, often a binary number representing the Sun's position based on which detector element is illuminated [35]. Analog sensors, on

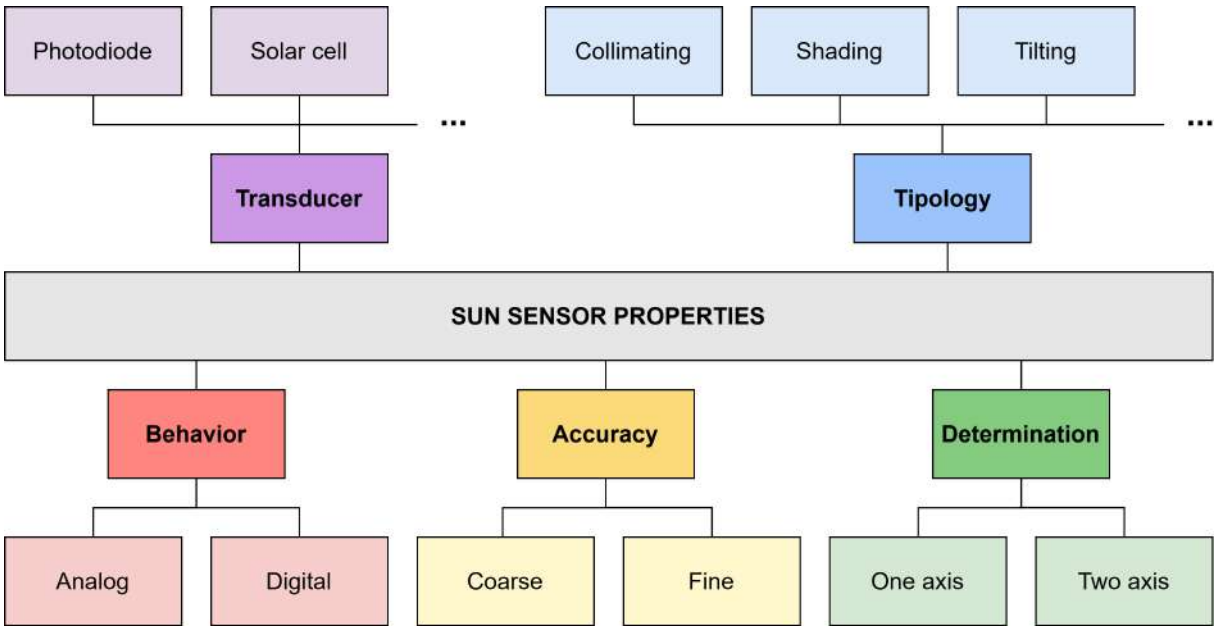


Figure 8: Sun sensor main properties (adapted from [34]).

the other hand, produce a continuous output signal and are frequently called cosine detectors. The energy deposited in a transducer, and consequently its measured quantity i, is proportional to the cosine of the angle of incidence of the solar radiation θ, as expressed in Lambert's law of Equation (42). This angle cosine can also be expressed by the dot product between the transducer's normal unit vector $\breve{n}$ and the Sun direction $\breve{s}$. Note the breve $\breve{\square}$ added on both vectors to denote this property.

$$i = i_{max} \cos\theta = i_{max}(\breve{n}^T \breve{s}) \tag{42}$$

Transmission losses due to the Fresnel effect can cause a change in the considered cosine curve when the incident angle is high enough [36]. Adding a multiplicative coefficient in Equation (42) to reflect this deviation from the ideal model is sometimes employed [36], together with an additive noise bias.

Regarding accuracy, Sun sensors can be broadly classified as coarse or fine. CSSs are popular due to their simplicity and low cost and are often used for initial attitude acquisition or safe-mode operations [37]. CSS designs aim to maximize the FOV, while still achieving degree accuracies [37]. Fine Sun Sensors (FSSs) generally employ precise optics and complex assemblies to determine the Sun's direction, achieving sub-degree accuracy [37], often in the arcseconds range.

There are many types of photoelectric transducers, such as photovoltaic cells, photodiodes, phototubes, photomultipliers or photoconducters, which vary in operation principle and consequently the affected electrical quantity and reacting wavelength range [38]. More recently, Complementary Metal-Oxide-Semiconductor (CMOS) and CCD technology have also been used for this purpose [36].

Regarding determination, single-axis sensors only lock in one direction. In contrast, two-axis sensors measure the Sun's position in azimuth and elevation, providing a complete orientation of the Sun relative to the sensor [34].

At last, Sun sensors are designed based on different geometric and operational principles. Besides single transducers on a flat surface, the most prominent types are collimating, shading and tilting sensors [34], and as the names suggest, they require specific structures beside the transducer. These are detailed in the next section, where the state-of-the-art on this topic is shared.

2.2 State-of-the-art

In this section, recent developments relevant to the scope of this dissertation are provided. Section 2.2.1 will focus on what characterises a Sun sensor array and how it performs nowadays, with several literature examples collected and analysed. From these, Section 2.2.2 will resume the most prominent methods to determine the Sun orientation and to calibrate Sun sensors. Section 2.2.3 will explore one of the most common algorithms for spacecraft attitude determination and some short notions of its stability.

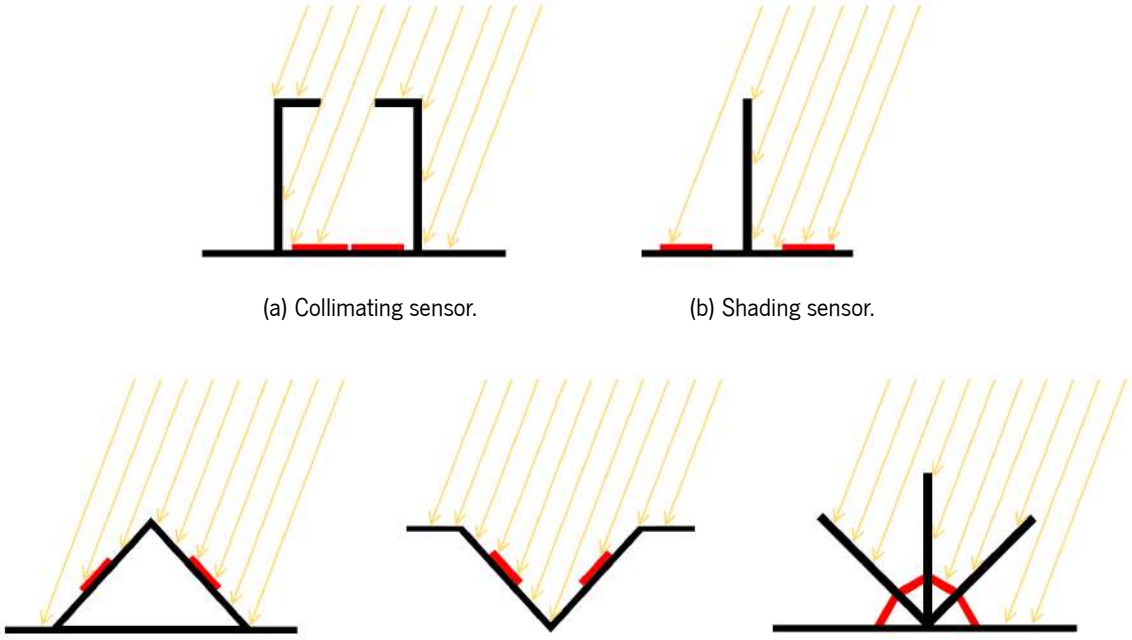
2.2.1 Sun sensor arrays

As discussed, a variety of array systems have been conceptualised by researchers, each with its unique operational principles.

Collimating sensors, for instance, employ a precision-focused structure, such as a tube with a pinhole or multiple apertures, to concentrate the Sun's rays onto the sensor [34]. The exact position of the light spot on the detector serves as a reliable indicator of the Sun's direction. These sensors can be of the straightforward single detection or four-quadrant detection types, further enhancing their precision [34].

Shading sensors, on the other hand, are designed to assess the interplay of light and shade on photodetectors, a result of an obstruction, to determine the Sun's apparent position [34]. The adaptability of these sensors to different shading patterns ensures their functionality in a variety of conditions, and the differences in the output of the detectors due to the shading pattern are effectively used to calculate the Sun's direction [34].

Tilting sensors, a simpler alternative, consist of two or more photodetectors mounted on inclined planes, often in a pyramidal form [34]. The Sun's position is determined by comparing the intensity of



(a) Collimating sensor. (b) Shading sensor.

(c) Tilting sensor (regular). (d) Tilting sensor (inverted). (e) Hybrid sensor (shading & tilting).

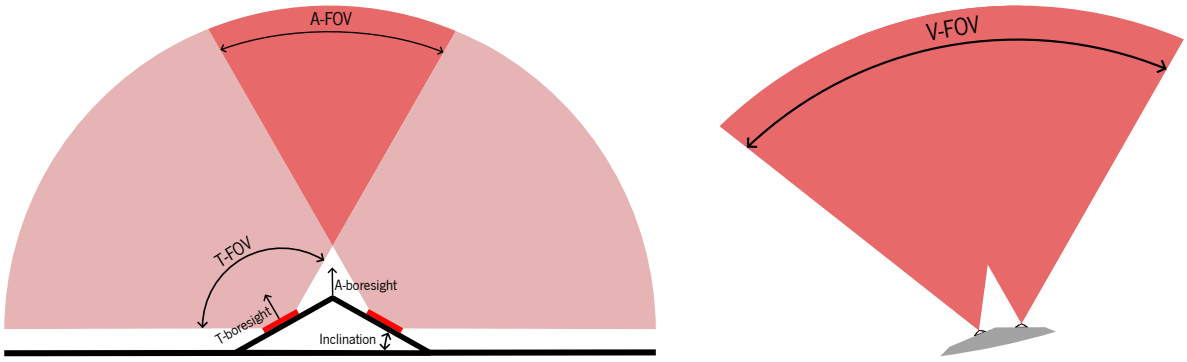
Figure 9: Different types of array assemblies, with transducers in red (adapted from [39]).

light received by each detector or with transducer models, making these sensors also easy to understand and use [34].

Other sensors could also be classified as hybrid, combining the principles above. These four types of assemblies were pictured in Figure 9.

Collimating sensors provide very high angular accuracy because their narrow FOV minimises stray light. However, their optical complexity and need for precise alignment increase both their design complexity and cost [34]. Shading sensors have a simple, robust design with wide FOV. Their simplicity comes at the expense of precision because the response is less sharply defined [34]. Tilting sensors are valued for their ease of integration into spacecraft surfaces. They extend the FOV by compensating for non-flat mounting geometries. However, their non-linear response often requires additional compensation to achieve the desired accuracy [34]. Overall, the main problems in the Sun sensor design, independently of type, are sensitivity and calibration, the damage of optical components by radiation and the operating temperature range of the Sun sensor [34].

The defined objectives of full Sun coverage of the satellite and relatively good accuracy must not be at the expense of cost-effectiveness, modularity and simplicity of the sensor, together with size and weight constraints characteristic of space applications. The tilting Sun sensor was therefore selected, specifically



(a) Zoom-in of a single pyramidal array.

(b) Zoom-out of a vehicle with multiple arrays.

Figure 10: Relevant definitions of pyramidal arrays (in a 2D-sliced view).

the regular pyramidal array (Figure 9c), because it is the simplest and less expensive to manufacture yet provides a rigid support for the sensors. Their working principle relies less on the geometric dimensions and allows for a wide FOV. On the other hand, it has high exposure to radiation, albedo, and pre-flight accidental damage, additionally requiring proper calibration. In order to mitigate this, radiation-tested components with flight heritage should be selected if possible, evaluate the influence of errors in measurements, minimise undue hardware handling, and conduct calibration testing campaigns, respectively.

Some researchers indeed focused their efforts on developing and improving CSSPAs, some even for satellite applications. A review of several devices was performed, with some key drivers evaluated regarding the transducer and pyramid: the transducer type and output were observed, together with the number of sides and inclination of the pyramidal array. Furthermore, the accuracy of the CSSPA was analysed. Three distinct FOV values were collected when possible: the singular Transducer's Field-Of-View (T-FOV), defined as the two-dimensional (2D) or 3D angle from where light can reach the transducer, however, some authors reduce this operation range, either physically with covers, or logically, by associating it with the Half-Power Angle (HPA); the Array's Field-Of-View (A-FOV), defined as the region where the CSSPA can uniquely determine the Sun position, by either three or, more commonly, all employed transducers on the array; and the Vehicle's Field-Of-View (V-FOV), defined as the regions in the vehicle where the Sun can be uniquely determined, sometimes also called as coverage. These three types are pictured in Figure 10, which shows a 4-sided CSSPA and its 2D-sliced regions, along with other important definitions. As will be explained in the next section, determining the Sun position requires at least three T-FOVs to overlap, thus creating the A-FOV region above the boresight where all four intersect. Note that the V-FOV is also the union of all A-FOVs spread across the vehicle. It is important to know that a FOV of 120 degrees means the same as a FOV of ±60 degrees, also known as full and half FOVs, respectively.

The properties described in the previous paragraph were compiled in Table 3. Other properties such as cost, power, size, weight, and material were taken into account, but their frequency was very low. Furthermore, general considerations, such as the Sun vector calculation method and chosen array placement, were collected from each example but expressed and analysed in the next section.

Table 3: Literature review on pyramidal arrays, with key drivers compiled.

Reference	Photoelectric transducer		FOV	Accuracy	Pyramidal array	
	Output	Type			Sides	Inclination
[37]	Analog	Photodiode	2π st [a]	1° - 2°	8	{20°, 50°}
[40]	Analog	Photodiode	4π st [v], 70° [t]	1°	17 [i]	{10°, 20°}
[41]	Analog	Solar panel	127.4° [a]	1° - 2°	16	26.4°
[42]	Digital	Ambient light sensor	140° [a]	0.1° - 5°	4	20°
[43]	Analog	Pinhole quadrant photodiode	120° [t]	0.5°	4	30°
[44]	Analog	Photodiode	120° [t]	N/A	4	30°
[45]	Digital	Ambient light sensor	4π st [v], 2π st [a], 100° [t]	N/A	4	[41°, 66°]
[46]	Analog	Solar cell	100° [a]	1°	4	40°
[47]	Analog	Photoresistor	90° [a]	1.7°	3	45°
[48]	Analog	Solar cell	4π st [v], 100° [a], 160° [t]	1.8°	4 [e]	{30°, 90°}
[49]	Digital	Light-to-frequency converter	2π st [a]	0.12° - 5°	8 [e]	{45°, 90°}
[50]	Analog	Photodiode	2π st [v]	1°	13 [i]	{0°, 45°, 90°}
[51]	Analog	Photodiode	90° [a], 110° [t]	N/A	4	45°
[52]	Analog	MEMS photodiode	109.4° [a]	1°	4	54.7°

[v] V-FOV
[a] A-FOV
[t] T-FOV
[i] individual tilted transducers on dispersed locations
[e] with one extra transducer on truncated top vertex

RAX-2 satellite [40] employed Osram's SFH-2430, an analog photodiode with a HPA of 60 degrees, to provide Sun measurements. In total, seventeen were placed on tilted surfaces, with their normal azimuth and elevation optimised to minimise Sun vector angular uncertainty. The selected photodiode inclination was 10 and 20 degrees, depending on the placement. However, on ±Z Printed Circuit Boards (PCBs), multiple redundant photodiodes, placed without tilting, are used to avoid possible shadowing from the antennas [40]. The tilted surfaces seem to be made from 3D-printed material, and the sensors were fixed in place with epoxy and connected with wire harnesses. The author also mentioned that photodiodes suffer degradation from radiation, which significantly affects the scale factor of their behaviour, thus adding a coverglass layer above to mitigate this, similar to what is used on solar cells. Despite the transducer's HPA being 60 degrees, the author mentions that its T-FOV is around 70 degrees since it maintains the relationship of Lambert's law up until that angle.

PW-Sat2 is a CubeSat developed by Warsaw University. In their experimental Sun sensor Critical Design Review (CDR) [42], the requirements were set for small size, accuracy between 0.1 and 5 degrees, A-FOV of 120 to 160 degrees on two axes; easy and cheap to manufacture structure with low power COTS sensors. They employed multiple ROHM's BH1730 Ambient Light Sensor (ALS) Integrated Circuit (IC), similar to photodiodes but with a built-in current-integrating amplifier and an Analog-to-Digital Converter (ADC) [42]. The structure used was a square pyramid with 20 degrees tilted faces, with a PCB with three ALS on each side, locked in place with an aluminium frame [42]. The temperature was measured with a Resistance Temperature Detector (RTD), and calibration curves were obtained to evaluate light-temperature dependence. An independent MCU interfaces all data readings from sensors to the main On-Board Computer (OBC) that computes the solar orientation with a 3-layer neural network [42].

Saborio [44] characterised the performance and errors of a "split pyramid" prototype for EXACT or SOCRATES missions, with photodiodes on a 3D-printed support [44]. Due to Lambert's cosine law, there is less sensitivity to light changes in the sensor's normal direction, so the author suggests titling them 30 degrees.

Gulmammadov et al. [37] mentioned that an A-FOV broader than 180 degrees is not desirable because of mounting restrictions and light reflections. They conducted a Monte Carlo simulation to evaluate accuracy and redundancy with noise present and proved that increasing the number of pyramid faces increases both metrics. Errors also increase along with increasing the inclination of the pyramid faces. The authors also noted that photodiode response to light near the boresight and the edges could negatively impact Sun determination due to low variation and high SNR, respectively. Therefore, they limited the photodiode T-FOV to [10, 80] degrees of incidence angles - or around 98% and 14% of normalised response, respectively - discarding the usual HPA assumption. The proposed pyramidal sensor in [37] has eight faces with one analog photodiode and respective ADC populating each PCB along its multiple wires, symmetrically distributed concerning the azimuth angles. To improve coverage, the authors used an optimisation algorithm to find the best pair of inclination angles for the sensors, which resulted in dividing them into symmetrical groups of 20 and 55 degrees of inclination.

The Sun sensors used in the NanoSat-1B satellite are constructed with a square pyramid, with 30 degrees of tilting, achieving a A-FOV of 160 degrees. The structure has the top vertex truncated to incorporate one extra transducer, colinear with the sensor boresight [48]. The employed solar cells are made from gallium arsenide, covered with a protective coverglass [48].

2.2.2 Sun vector determination and calibration

The principle of the complete Sun vector determination using a CSSPA is usually based on the transducer's cosine behaviour, ideally given by Equation (42) and here augmented to reflect uncertainty and noise [49]. If the sampling period is sufficiently fast, such that the motion of the Sun and the sensor will have a negligible effect on all M measurements, they can be approximated as constants during sampling [49]. Equation (42) can then be stacked into a matrix form given by

$$\check{I} = \check{N}\check{s} + \varepsilon \tag{43}$$

where $\check{I}$ is the Mx1 vector of all normalized measurements, composed by i_j/i_{max} for each j-th transducer, $\check{N}$ is the$M$x3 matrix formulated by all normal unit vectors n_j of each j-th transducer, $\check{s}$ is the 3x1 Sun direction vector, and ε is a Mx1 measurement error vector [49].

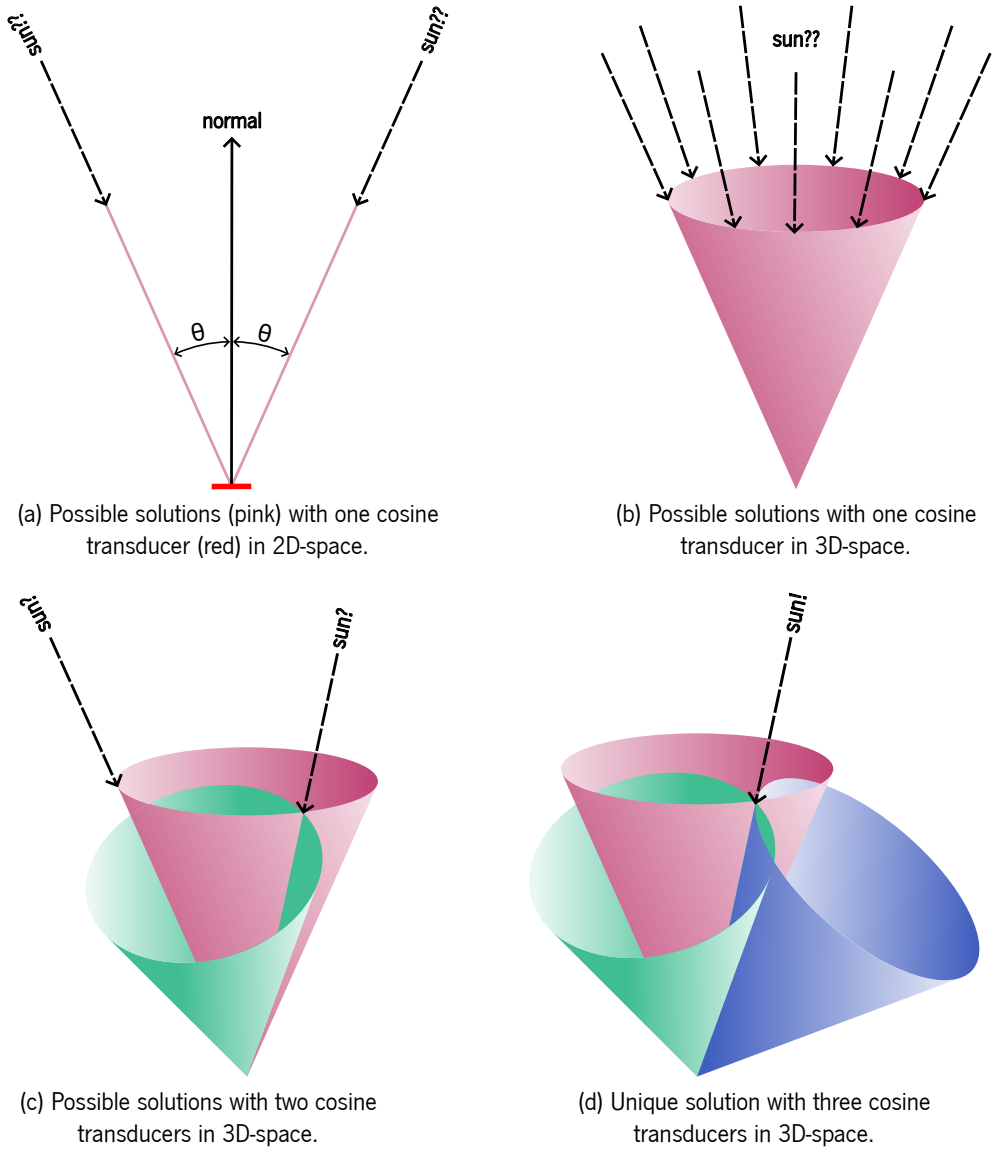
From the Gauss-Markov Theorem, the best linear estimate of the Sun vector $\hat{s}$ is given by

$$\hat{s} = (\check{N}^T\check{N})^{-1}\check{N}^T\check{I} \tag{44}$$

if measurement error has zero mean and finite variance [49]. Additionally, the covariance matrix of the error is proportional to $1/M$, meaning that increasing the number of available measurements, besides increasing redundancy, also increases accuracy [49].

However, the lower bound on the number of measurements needed to compute the Sun vector must also be found. It is clear that the solution given by Equation (44) only exists when the matrix $(\check{N}^T\check{N})$ is invertible, or in other words, when it has non-zero determinant or rank three [49]. For this to be achieved, at least three linearly independent measurements are needed, preferably with low noise or interferences [49]. One can think of the Sun vector as having three components, and each transducer provides one. Once more than three components (measurements) are provided, the LS estimator minimises the influence of the disturbances on the overdetermined solution.

A geometric approach could also be used to describe this phenomenon. Starting with a 2D problem to determine the Sun vector, a single transducer measurement following Equation (42) would give us two possible solutions represented with the pink lines in Figure 11a since both are at θ degrees from the normal vector. Translating this to a 3D-space, a measurement would create a cone of possible solutions by revolving those pink lines around the normal vector, as seen in Figure 11b. By sequentially increasing the measurements, represented by the other cones on Figures 11c and 11d, the unique solution to the problem can be narrowed down by intersecting them [32]. Note that the linear independence of measurements, in



(a) Possible solutions (pink) with one cosine transducer (red) in 2D-space.

(b) Possible solutions with one cosine transducer in 3D-space.

(c) Possible solutions with two cosine transducers in 3D-space.

(d) Unique solution with three cosine transducers in 3D-space.

Figure 11: Geometric representation of the Sun vector determination problem.

this scenario represented by the different pointing cones, is fundamental to achieving a solution with three measurements.

The sensor specifications dictate the error associated with each measurement and obviously impact the Sun vector solution. Wang et al. [41] outlined that non-planar Sun sensors' errors in solar orientation determination arise from internal and external interferences. Regarding the CSSPAs, internal interference can be originated from three types of sources: the scale factor degradation, either caused by imperfect circuitry or by radiation over time [40], mechanical misalignment, either of the array structure itself or the transducers in each pyramid face [36]; and nonlinear behaviour of the transducer [36]. These relate to each component that formulates Equation (42). Studies suggest that besides increasing the number of

illuminated photodiodes, uniform and symmetric configurations can mitigate this type of interference [41], but proper calibration would substantially improve results.

External interference arises from environmental factors such as scattered and reflected sunlight, as well as infrared radiation sources. Wang et al. provided a mathematical formulation for this issue and concluded that it remains largely independent of the number of sensors, predominantly contingent upon orientation, and it is impossible to be completely suppressed [41]. However, the main source of scattered and reflected sunlight might come from Earth's albedo, and therefore, modelling it would prove useful to mitigation.

The scale factor relates the measured output to the incident irrandiance [40]. Usually, the transducer measures current or voltage but can also measure illuminance for more advanced components. The scale factor is commonly discovered by pointing a calibrated light source directly to the transducer, which in turn will output the maximum intensity measurement. However, this proportionality will differ from transducer to transducer because each might have different circuitry and surroundings [40]. Temperature and radiation will also affect sensor behaviour. However, while the scale factor might cycle during orbit due to temperature dependence, radiation will continuously degrade the transducer condition [40].

Mafi et al. [36] explained that mechanical errors produce misalignments on transducers' normal vector, causing a mismatching between the supposed mathematical model and where it is oriented in reality. These can arise during multiple stages, including the structure manufacturing, transducer installation or sensor placement on the spacecraft [40].

The cosine curve is considered the ideal model of most common photoelectric transducers. These commonly have a protective cover made of glass [53]. The incidence angle affects the amount of solar radiation transmitted through the cover and, therefore, converted to electricity [53]. However, as the incidence angle increases, the amount of radiation reflected from the cover also increases [53]. This reflection creates the nonlinear behaviour mentioned above, most noticeable for incidence angles greater than 60 degrees [36], also known as the Fresnel effect [36].

This effect can be expressed as a function of incidence angle, in terms of the transmittance T, defined as the ratio of the transmitted to the incident power [54] and given by

$$T(\theta_i) = 1 - \frac{1}{2}\left[\frac{\sin^2(\theta_i - \theta_t)}{\sin^2(\theta_i + \theta_t)} + \frac{\tan^2(\theta_i - \theta_t)}{\tan^2(\theta_i + \theta_t)}\right] \tag{45a}$$

$$\theta_t = \sin^{-1}[(r_i/r_t)\sin\theta_i] \tag{45b}$$

where θ_i is the incidence angle and θ_t is the refraction angle given by Snell's law [54], following Equation (45b). The absolute refractive indices r_i and r_t depend on the incident and transmitted medium properties, respectively, and their relative index is given by its ratio. The absolute refractive index for air and glass are 1 and 1.526 [53], respectively, therefore the relative refractive index for an air-to-glass propagation is $r_i/r_t = 0.655$.

Paying attention to Equation (45a), this equation is undefined when $\theta_i = 0$ [54]. For this instance, Equation (46) is defined, and one interesting property can be observed: even when the incident angle is null, there is some residual power that is not transmitted [54]. For the air-to-glass example, only 96% is effectively transmitted, and the remaining is reflected, as seen on Figure 12a, where other indices are analysed.

$$T(0) = \frac{4r_t r_i}{(r_t + r_i)^2} \tag{46}$$

However, in order to add this effect to the transducer model, transmittance needs to be normalised so that the measured intensity remains normalized. This originates the transmittance coefficient τ [36] given by

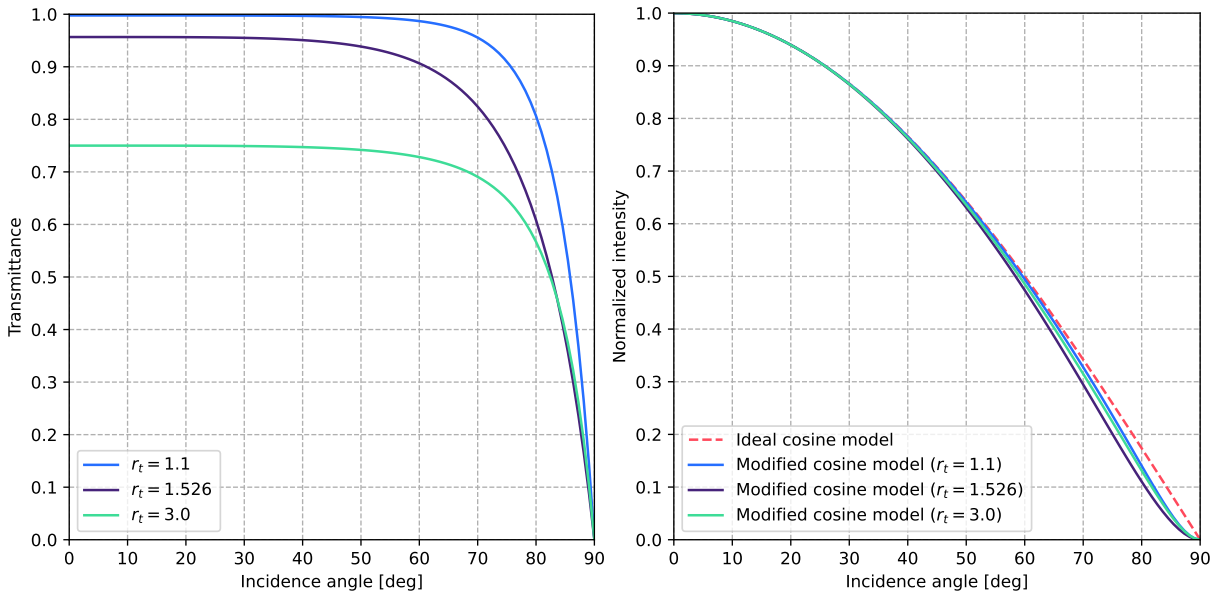
$$\tau(\theta_i) = T(\theta_i)/T(0) \tag{47}$$

Adding this multiplicative coefficient in Equation (42) to reflect the Fresnel effect results in the modified cosine curve of Equation (48) [36]. A comparison between the ideal and modified model is illustrated in Figure 12b. As the incidence angle increases, the modified curves deviate more from the ideal one because the transmittance rapidly decreases at large angles.

$$i = i_{max}\tau(\theta)\cos\theta = i_{max}\tau(\theta)(\check{n}^T\check{s}) \tag{48}$$

After understanding the behaviour and errors, calibration can be pursued for these cosine transducers. Mafi et al. [36] provided a framework that can be applied in either pre-flight or on-orbit for CSSPAs. The first step necessary for calibration is to obtain a dataset. It must be composed of distributed measurements of all transducers across the A-FOV and the true light source direction for each point [36].

Starting by calibrating the misalignments, the real normal vectors for each transducer need to be discovered, preferably with the linear estimator shown in Equation (44), due to its simplicity [36]. To do so, a region where the cosine model is approximately linear must be defined [36]. Observing Figure 12, when the incidence angle is between 0 and 30 degrees, the modified model is very close to the ideal one, therefore, it is valid [36]. Then, one can rearrange Equation (43) so that the solution to the problem returns



(a) Transmittance.

(b) Transducer response.

Figure 12: Fresnel effect on cosine transducers for different refractive indices.

the optimal normal vector estimate $\hat{n}$, instead of the Sun vector [36]. This rearrangement gives

$$\hat{n}_t = (\check{S}^T \check{S})^{-1} \check{S}^T \check{I}_t \tag{49}$$

where $\check{I}$ is the Mx1 vector of all normalised measurements M from the dataset that respect the constraint $0° \leq \theta \leq 30°$ and $\check{S}$ is the Mx3 matrix formulated by the known Sun vectors from the dataset for each measurement [36]. This step must be done separately for each transducer, as denoted above by the subscript $\square_t$.

Regarding the nonlinearity introduced in the measurement model, an issue with the implementation appears. The Sun vector determination algorithm, which was initially based on the ideal linear behaviour of the transducer, now will have worse performance when the Fresnel effect is more noticeable [36], namely when the Sun is close to the T-FOV limits. To prevent this, two options arise: either a nonlinear LS method is used to estimate the Sun vector; or a new linear model is developed in the hope of somehow representing these new nonlinear characteristics and minimise the error across the entire T-FOV [36]. However, there is no dilemma here, and the second option can be the starting point to get a fast and reasonable estimate of the Sun vector, and then hopefully interate to a more accurate solution using the first option, at the expense of more computational load and time [36].

The new linear model describes a new cosine behaviour given by

$$i = \alpha \cos\theta + \kappa = \alpha(\check{n}^T \check{s}) + \kappa \tag{50}$$

where α and κ are gain and bias parameters that need to be calibrated to fit the curve to the data points [36]. Again, the LS method is applied to estimate the optimal values of both with

$$\begin{bmatrix} \hat{\alpha}_t \\ \\ \hat{\kappa}_t \end{bmatrix} = (C^T C)^{-1} C^T I_t \tag{51}$$

where I is the $M \times 1$ vector of all measurements from the dataset and C is the $M \times 2$ matrix of the indicence angle and a constant, formulated by $C = [\hat{N}_t^T \check{S} \quad 1]$ [36]. This step must be done for each transducer as well.

The iterative process mentioned earlier begins when the transducers take new measurements. The Sun vector is estimated using the linear LS of Equation (44). Then, the measurements are divided by the transmittance coefficients, calculated for each transducer using Equation (47) [36]. These last two steps are repeated until new measurements are available or the estimated Sun vector converges, possibly quantified by a threshold. The starting estimate must be reasonably close to the optimal estimate; otherwise, the solution could not converge.

With this framework, Mafi et al. obtained a simulated improvement from 6.2 to 0.9 degrees [36]. Pita et al. calibrated their CSSPA in laboratory, effectively reducing the error from 7.5 to 1.8 degrees with a similar non-iterative approach [48].

Calibrating for radiation and temperature requires characterisation curves and new modified models. Radiation impact is expensive and difficult to assess empirically, but temperature dependence could be evaluated similarly to PW-Sat2 [42], as described in Section 2.2.1.

Performing the calibration requires a testing setup that points a stable and calibrated light source to the Sun sensor from multiple directions. This setup can be achieved with a two degrees of freedom (2-DOF) test stand, where the sensor is placed and rotated while the beam is fixed at some known reference direction [48]. This allows for the measurement dataset to be obtained while simultaneously knowing where the light is coming from. The range and steps of rotation can vary, depending on the stand specifications. The above modelling assumed that the incident light was a point source at infinity, which is a reasonable assumption for a Sun sensor in orbit. However, this assumption must not be broken during calibration.

The experimental setup of Pita et al. [48] included a two-axis rotation system and a collimated light beam with Air Mass Zero (AM0) irradiance, identical to what could be expected in orbit. Pre-flight testing in AM0 light is important to design the sensor to maximise sensor resolution while minimising the chance of saturation [55], but these simulators are quite expensive. PW-Sat2 team tried to approximate the Sun

spectrum at least using a COTS metal-halide lamp combined with a collimator [42]. A motorised two-axis rotation table allowed collecting measurements at 0.5 degrees steps for almost all the hemisphere above the sensor [42]. Besides calibration procedures, vacuum, vibration, and thermal tests were also proposed [42]. Saborio [44] created one simple model made of wood and rotated it manually at a 5 degrees step, with a spotlight fixed above the CSSPA [44]. Nurgizat [56] rotated the sensor on a three-axis holder, at 5 degrees steps, placed horizontally at 32 cm from a 300 W halogen spotlight.

Now that all pre-flight calibration and static Sun vector determination are detailed, some authors explored an alternative to achieve on-orbit calibration and dynamic determination. A common attitude estimation problem involves propagating the state dynamics and correcting that estimate using a direct measurement of the attitude, as it will be presented in the next section [57]. Thus, instead of solving the Sun vector determination problem, sensor measurements can be used to correct a propagated estimate of the vector with a Kalman filter [57]. While a single transducer measurement cannot fully determine the Sun direction, a series of measurements coupled with the differential rotation information from a rate gyro can be enough [57]. On top of this, on-orbit calibration of the transducer characteristics can also be achieved via the Kalman filter by augmenting the filter state to estimate the measurement model parameters [40]. This approach could mean lower costs and development times since it requires almost no calibration hardware and testing procedures before flight [40].

2.2.3 Spacecraft attitude determination

The EKF is the *de facto* standard for real-time spacecraft attitude estimation [58]. It is a simple and flexible tool for nonlinear dynamic systems with extended heritage and proven success [58]. However, it is not without its challenges. Poor performance or divergence from the linearization and initial conditions can occur if not carefully designed and tuned [58]. While other filters have been recently suggested to address these issues, such as the unscented version, they often come with increased computational cost and complexity [58], which is undesirable for this application. The attitude EKF has several implementations, such as the Additive Extended Kalman Filter (AEKF) [59] or the MEKF [60, 61].

Regarding the attitude representation form, since the SO(3) group has dimension three, most attitude EKFs use lower-dimensional representations than the common nine-parameter rotation matrix. On the other hand, all three-parameter representations of the 3D rotation group are singular or discontinuous for certain attitudes. Thus, its representation must be either singular or redundant [58]. The unit quaternion was selected because it has the lowest dimension of any globally non-singular parameterization [27], and

it is trivial to restore normalization that may be lost due to numerical errors [58] while simultaneously improving filter convergence [62].

Regarding the filter implementation, the AEKF applies an additive error update to its state, whereas the MEKF employs a multiplicative error formulation that better preserves rotation constraints [58]. Additionally, the AEKF considers each quaternion component as independent (which is not physically true) and requires excessive covariance elements (which is undesirable) [58].

Concerning the above paragraph, the MEKF was chosen for attitude estimation and the axis-angle vector for error representation because of simpler intuition. As described in Section 2.1.2.2, this representation has a singularity at 180 degrees, but it is not expected to rotate that much between successive time steps. It is also ill-defined at null rotations, but that can easily be accounted for in code [63].

The MEKF estimates a local 3D vector of an attitude error, while the correctly normalized four-component quaternion provides a globally non-singular attitude representation [27]. This separation is necessary to avoid working with rank-deficient matrices [27], besides the considerations explained in Section 2.1.1.2 about the EKF. This results in the filter formulation having two states, explained later in the section, and the true attitude being represented as the quaternion product given by

$$q = q_n \otimes \Delta q(\psi) \tag{52}$$

where q_n is some unit nominal quaternion and Δq a unit quaternion, parameterized by the error axis-angle 3D vector ψ, representing the small rotation from q_n to the true attitude q [27]. The mapping between the axis-angle vector and the unit quaternion is given by Equation (21).

Many implementations incorporate gyro data directly into the dynamic model rather than using it solely for measurement updates. This approach has two key advantages: first, gyro measurements are often more accurate than the available rotational dynamics and torque models; second, it requires considerably less computation [30].

Farrenkopf's rate-integrating gyro dynamic model [64] can be employed, which assumes that the angular velocity measurement is corrupted by a bias drift and Gaussian noise. This bias is also slowly time-varying but can be modelled as a random walk process (integral of white Gaussian noise). This behaviour translates to

$$\tilde{\omega}_k = \omega_k + \beta_{\omega_k} + \eta_{v_k} \tag{53a}$$

$$\beta_{\omega_{k+1}} = \beta_{\omega_k} + \eta_{u_k} \tag{53b}$$

where $\tilde{\omega}$ is the measured rate, ω is the true rate, β_ω is the gyro bias drift, η_v and η_u are independent zero-mean Gaussian white noise processes, with σ_v^2 and σ_u^2 variances, respectively, as already seen on Section 2.1.1.2.

Assuming that the spacecraft gyro sample rate is much faster than the dynamics, in order to assume angular velocity and the bias as constant over a sample period Δt, the attitude state propagation can then rely only on the quaternion kinematics [26] given by Equation (33), and here discretized to

$$q_{k+1} = q_k \otimes \Delta q(\omega_k \Delta t) \tag{54}$$

Gaussian noise influence will be small for good gyros, but the bias drift can be too large to ignore, which corrupts the filter. This influence can be avoided by augmenting the state and estimating it [30]. Let then the nominal state x be composed by Equation (55a) and the error state Δx by Equation (55b).

$$x = \begin{bmatrix} q \\ \beta_\omega \end{bmatrix} \begin{matrix} : \textit{ attitude quaternion in inertial frame} \in \mathbb{S}^3 \\ : \textit{ gyro bias drift in body frame} \in \mathbb{R}^3 \end{matrix} \tag{55a}$$

$$\Delta x = \begin{bmatrix} \Delta\psi \\ \Delta\beta_\omega \end{bmatrix} \begin{matrix} : \textit{ error attitude axis-angle vector in body frame} \in \mathbb{R}^3 \\ : \textit{ error gyro bias drift in body frame} \in \mathbb{R}^3 \end{matrix} \tag{55b}$$

The angular velocity can be represented in axis-vector form with Equation (56), allowing Equations (53), (21) and (54) to formulate the state propagation [63], simplified in Equation (57). Notice that the angular velocity ω is treated as a control input.

$$\omega_k = \tilde{\omega}_k - \hat{\beta}_{\omega_k} \tag{56a}$$

$$\theta = \|\omega_k\| \Delta t \tag{56b}$$

$$u = \frac{\omega_k}{\|\omega_k\|} \tag{56c}$$

$$\hat{x}_k^- = f(\hat{x}_{k-1}^+, \omega_k) = \begin{bmatrix} \hat{q}_{k-1}^+ \otimes \begin{bmatrix} \cos\frac{\theta}{2} \\ u \sin\frac{\theta}{2} \end{bmatrix} \\ \hat{\beta}_{\omega_{k-1}}^+ \end{bmatrix} \tag{57}$$

As explained in Section 2.1.1.2, an EKF is formulated to estimate the small error deviations from the nominal state. This means that the covariance matrix must be propagated reflecting the Jacobian matrix

of the error states [63], given by

$$F_{k-1} = \begin{bmatrix} \frac{\partial \psi_k}{\partial \psi_{k-1}} & \frac{\partial \psi_k}{\partial \beta_{\omega_{k-1}}} \\ \frac{\partial \beta_{\omega_k}}{\partial \psi_{k-1}} & \frac{\partial \beta_{\omega_k}}{\partial \beta_{\omega_{k-1}}} \end{bmatrix} = \begin{bmatrix} {}^{\mathcal{B}_k}R^{\mathcal{B}_{k-1}} & -I_3 \Delta t \\ 0_3 & I_3 \end{bmatrix} \tag{58a}$$

$${}^{\mathcal{B}_k}R^{\mathcal{B}_{k-1}} = \left[I_3 + (\sin\theta)[u]_\times + (1-\cos\theta)[u]_\times^2 \right]^T \tag{58b}$$

The first partial derivative is the rotation matrix between the two vectors from successive time steps on the body frame, given by Equation (58b) [26]. The second also has a dependence on angular velocity that can be approximated to the negative time interval between each step, since rotation angles are small [26].

Regarding the measurements, only one vector observation was considered, but several sensors can provide useful data to update the attitude, either as measured quantities retrieved directly from the sensor, such as the gyroscope case, or as unit vector directions. The Line Of Sight (LOS) measurement model is a common implementation [65] and is given by

$$\tilde{y}_k = A(q_k) r_k + \varsigma_k \tag{59}$$

where r_k is a known reference direction in the inertial frame, $A(q_k)$ is the rotation matrix from inertial to body frames and ς_k is the sensor zero-mean Gaussian white noise process, with σ_r^2 variance. Notice that it is equivalent to the EKF measurement model. The process and measurement noise matrices Q_k and R_k are then given by

$$Q_k = \begin{bmatrix} \sigma_v^2 I_3 & 0_3 \\ 0_3 & \sigma_u^2 I_3 \end{bmatrix} \tag{60a}$$

$$R_k = \sigma_r^2 I_3 \tag{60b}$$

The covariance matrix requires the Jacobian matrix of the measurement functions for the update step [65], and it is given by

$$H_k = \begin{bmatrix} [\tilde{y}_k]_\times & 0_3 \end{bmatrix} \tag{61}$$

The nominal state is updated with the error state by distinct operations. The quaternion is updated with the product defined in Equation (52) and the gyro bias with an addition. In theory, the quaternion would already respect its unit norm, but due to possible numerical errors, it is good practice to normalize it [26]. After this, the error state vector is then reset, computationally setting the variable to zero or with a new calculation overriding the current value. This step is especially relevant for the orientation part, as

the new orientation error will be expressed locally concerning the orientation frame of the new nominal state [26].

All filter steps were resumed in Algorithm 4. This formulation is called the invariant MEKF. It differs from the traditional MEKF by making the measurement independent from the global attitude estimate on Equation (61), which considerably improves performance with large initialization errors [65]. It is also relatively efficient and light due to the simplifications assumed, which is crucial for this application. Other variations of the MEKF exist, and Chang [65] recently explored some of them, providing a well-founded comparison of implementations and results.

Algorithm 4 Discrete-time invariant multiplicative extended Kalman filter with Joseph's form.

$\hat{q}_0^+, \hat{\beta}_{\omega_0}^+, P_0^+$ ▷ initialize state estimates and covariance

for all timesteps k **do**

$\omega_k \leftarrow \tilde{\omega}_k - \hat{\beta}_{\omega_k}$ ▷ calculate true angular velocity

$\theta \leftarrow \|\omega_k\| \Delta t$ ▷ calculate angle of rotation

$u \leftarrow \omega_k / \|\omega_k\|$ ▷ calculate axis of rotation

$\hat{q}_k^- \leftarrow \hat{q}_{k-1}^+ \otimes \begin{bmatrix} \cos(\theta/2) & ; & u\sin(\theta/2) \end{bmatrix}$ ▷ predict attitude estimate

$\hat{\beta}_{\omega_k}^- \leftarrow \hat{\beta}_{\omega_{k-1}}^+$ ▷ predict gyro bias estimate

$F_{k-1} \leftarrow \begin{bmatrix} \left[I_3 + (\sin\theta)[u]_\times + (1-\cos\theta)[u]_\times^2\right]^T & -I_3\Delta t & ; & 0_3 & I_3 \end{bmatrix}$

$P_k^- \leftarrow F_{k-1} P_{k-1}^+ F_{k-1}^T + Q_{k-1}$ ▷ calculate state Jacobian and predict estimate covariance

$z_k \leftarrow \tilde{y}_k - A(\hat{q}_k^-) r_k$ ▷ calculate measurement residual

$H_k \leftarrow \begin{bmatrix} [\tilde{y}_k]_\times & 0_3 \end{bmatrix}$ ▷ calculate measurement Jacobian

$K_k \leftarrow P_k^- H_k^T (H_k P_k^- H_k^T + R_k)$ ▷ calculate Kalman gain

$\Delta x_k^+ \leftarrow K_k z_k$ ▷ calculate estimation error

$\hat{q}_k^+ \leftarrow \hat{q}_k^- \otimes \begin{bmatrix} \cos(\|\Delta\psi_k^+\|/2) & ; & (\Delta\psi_k^+/\|\Delta\psi_k^+\|)\sin(\|\Delta\psi_k^+\|/2) \end{bmatrix}$

$\hat{\beta}_{\omega_k}^+ \leftarrow \hat{\beta}_{\omega_k}^- + \Delta\hat{\beta}_{\omega_k}^+$ ▷ correct attitude and gyro bias estimate

$P_k^+ \leftarrow (I - K_k H_k) P_k^- (I - K_k H_k)^T + K_k R_k K_k^T$ ▷ correct estimate covariance

$\hat{q}_k^+ \leftarrow \hat{q}_k^+ / \|\hat{q}_k^+\|$ ▷ normalize attitude estimate

end for

return $\hat{q}_k^+, \hat{\beta}_{\omega_k}^+, P_k^+$

One last thing to consider is the filter stability, and prooving it on a stochastic estimator is not straightforward. Tønne [66] provides some extensive research done on the stability of the EKF. In summary, the estimation error for a stochastic discrete-time system is bounded if it satisfies the criteria below [66]:

1. The solution of the Riccati equation remains positive definite and bounded;
2. The initial estimation error is small;

3. The noise terms are small;
4. The nonlinearities of the system are continuous;
5. The Jacobian matrix of the output must be bounded.

Numerical stability is simpler to evaluate and is usually accessed by verifying if the lowest of all inverted real parts of the state Jacobian eigenvalue is a lot higher than the discrete-time step interval. Nonetheless, demonstrating the stability of the EKF is usually done per application and relies on empirical proof.

Chapter 3

Proof-of-concept prototype

3.1 Introduction

This chapter presents the design, development and assessment of a proof-of-concept prototype designed to demonstrate the feasibility and effectiveness of CSSPAs, tailored to this application. The primary objective of this prototype is to validate the theoretical concepts and methodologies discussed in the previous chapter through practical and iterative implementation and testing. This is achieved by creating a low-cost, modular, and oversized model that can be easily scaled to smaller dimensions while still performing with reasonable accuracy and A-FOV.

The proof-of-concept prototype is a critical step in bridging the gap between theory and application, allowing us to design and evaluate the basics of the Sun sensor, such as the selected light transducers, array configuration, and testing setup and procedures. This approach helps to identify potential challenges and limitations in the following versions.

This chapter is structured as follows. Section 3.2 begins with the selection process for the photoelectric transducer, detailing the criteria and considerations that guided its choice. Section 3.3 describes the array system's design, highlighting the initial CSSPA's configuration and layout. Following this, the methodology and metrics for determining the Sun vector from sensor data and the testing setup are outlined in Section 3.4, with the results analysed in Section 3.5. Finally, Section 3.6 summarises the key findings, discusses their implications and provides future improvements.

3.2 Photoelectric transducer selection

The design process began by choosing the photoelectric transducer, as it is one of the most important elements of the CSSPA. After available components were grasped and selected, the design evolved around that.

The first two properties defined to narrow down the thousands of components possible were availability and price. The selected component must be classified as COTS, which means that it is mass-produced and sold in common resellers such as Digikey. It is also good practice to select components that are in active production, rather than obsolete, to create a design that can be easily reproduced in the near future. When discussing COTS components, they are typically inexpensive or at least less expensive than proprietary alternatives. For COTS photoelectric transducers, a quick search yielded prices between cents to a few dozen Euros.

Simplicity, modularity, lightness, and tinyness can be achieved by selecting a specific type of photoelectric transducer: the ALS. This light sensor measures the illuminance of visible light, in lux, and converts it to a digital value, due to its embedded ADC, truly becoming an all-in-one IC. They are very small, typically measured in millimetres, so several devices can be stacked side by side, as required by this application. On Digikey, a few thousand are available, with the majority having an Inter-Integrated Circuit (I2C) protocol interface as well, which facilitates communication with the transducer.

Next, attention was given to how well the transducer performs. The resolution, range, and HPA were the main characteristics analysed in an ALS, without forgetting the size and cost, as discussed earlier. Other properties, such as supply voltage, active current, sampling period, interface protocol and operating temperature, were also taken into account. A small batch of six different options was composed, all of which share good values for the properties described as desired. These were compiled on Table 4, separated by primary and secondary properties to fit the page width. The best value for each property was highlighted.

Table 4: Properties comparison of available ALS.

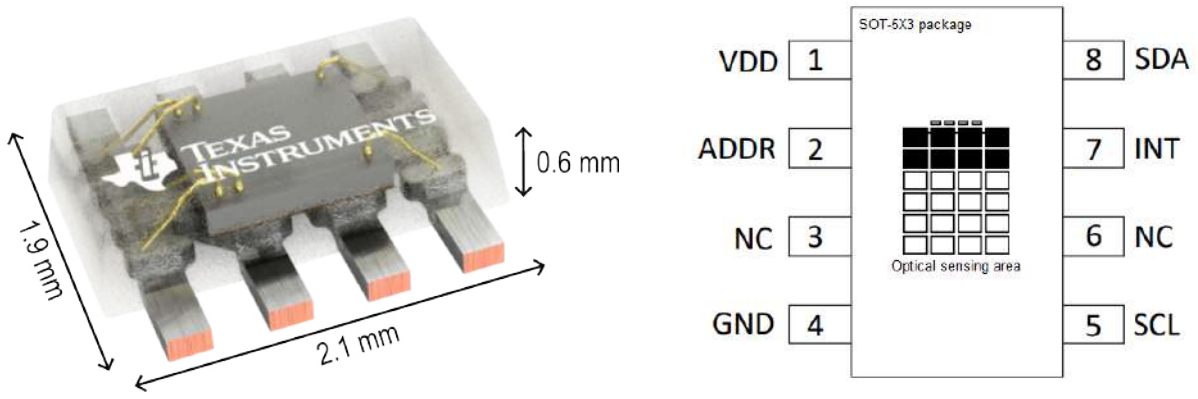
Category	**Manufacturer part number** (first row) — **Manufacturer** (second row) — **Properties** (remaining rows)					
	BH1730FVC	BH1750FVI	OPT3001DNPR	OPT4001DTSR	TSL25911FN	VEML7700-TR
	ROMH Semiconductor	ROMH Semiconductor	Texas Instruments	Texas Instruments	ams OSRAM	Vishay Semiconductors
Range (lx)	0.001 — 100k	0.11 — 100k	0.01 — 83k	~0 — 117k	~0 — 88k	**~0 — 167k**
Resolution (bits)	16	16	23	28	**29**	16
HPA (deg)	N/A	50	47	**60**	**60**	55
Size (mm)	3.0×1.6×0.75	2.6×1.8×0.75	2.0×2.0×0.65	**2.1×1.9×0.6**	2.2×2.0×0.6	6.8×2.4×3.0
Cost (€)	3.07	3.03	1.25	1.32	1.54	**1.14**
Voltage (V)	2.4 — 3.6	2.4 — 3.6	**1.6 — 3.6**	**1.6 — 3.6**	2.7 — 3.6	2.5 — 3.6
Current (µA)	150	120	**3.7**	30	275	45
Sampling (ms)	104.6	16 — 180	100 — 800	**0.6 — 800**	100 — 600	25 — 800
Interface	I2C	I2C	I2C	I2C	I2C	I2C
Temperature (°C)	-40 — 85	-40 — 85	-40 — 85	-40 — 85	-40 — 85	-40 — 85
Package	WSOF6	WSOF6I	USON6	SOT-583	6-WFDFN	4-SMD

The bottom three transducers exhibited better properties, with each sensor having at least one primary property superior to the others. Thus, the top half was discarded. All had an I2C interface and the same operating temperature. A trade-off study was then conducted to select the best candidate among the remaining options.

The VEML7700 presented the best value in terms of range and cost, but it severely lacked in resolution and size compared to the remaining three, which could not be tolerated. The TSL2591 offered the best resolution but had a worse range, size and cost than the OPT4001. The latter had only one less bit than the TSL2591, but had around 30% more range and almost all of the best secondary properties. This combination of factors made the OPT4001 the most reasonable choice for use as a transducer, which may be the same reason it was already employed on PROMETHEUS-1.

Two packages of the sensor were available, but the SOT-5X3 variant (OPT4001DTSR) was selected due to its better properties, as displayed in Table 4. This variant features an 8-pin package: two for power input; two for I2C communication; one for I2C address selection; one for interrupt features; and the remaining two without any electrical connection, as displayed in Figure 13. It also presents some useful functionalities such as configurable I2C addresses, programmable sampling (or conversion) time, programmable or automatic range selection, and several operating modes, including low-power functioning [67]. These will be discussed in the next paragraphs. It is also easier to handle than the PICOSTAR variant.

Regarding the I2C protocol, the OPT4001 supports three transmission speeds: the standard mode of 100 kHz, the fast mode up to 400 kHz and the high-speed mode up to 2.6 MHz [67]. The standard was selected since it is initialised as the default and is sufficient for this purpose. Four I2C addresses are available by connecting the ADDR pin to one of four pins: GND, VDD, SDA, or SCL. The state of the



(a) Dimensions. (b) Pinout.

Figure 13: OPT4001DTSR hardware (adapted from [67]).

ADDR pin must be connected to the desired level before any activity on the interface occurs [67]. Table 5 summarises the possible addresses with the corresponding ADDR pin configuration. The address selection will be explored in the next section.

Table 5: Available I2C addresses on OPT4001 (adapted from [67]).

ADDR pin connection	Assigned I2C address	
	Binary	Hexadecimal
GND pin	0b01000100	0x44
VDD pin	0b01000101	0x45
SDA pin	0b01000110	0x46
SCL pin	0b01000111	0x47

Three available modes of operation can be set via register configuration: the power-down mode; the continuous mode; and the one-shot mode [67]. The power-down mode sets the device to a low-power standby phase, with 95% less consumption, and can be awakened by any I2C transaction. The continuous mode measures and updates the output registers non-stop, in intervals determined by the conversion time. For every successful conversion, a hardware interrupt is generated. However, the device is continuously kept active to minimise the interval between measurements. The one-shot mode allows the device to remain in standby mode, and measurement is triggered either by a register write command via I2C or by a hardware interrupt [67]. This duality means that, depending on the operation mode, the interrupt pin can act as either an input or output.

Several operational light ranges are available, from 459 lx up to 117,441 lx, incrementing in binary multiples [67]. Nevertheless, as the range increases, the resolution within that range decreases since the sensor's 28-bit effective resolution is distributed across a broader span of light intensities. Internally, each

measurement is encoded using 24 bits, with 20 bits for the mantissa and 4 bits for the exponent, but the combination of this semi-logarithmic encoding across multiple ranges results in an effective resolution equivalent to 28 bits [67]. In other words, the mantissa part represents the linear ADC bits, which are proportional to the measured light within a given range, and the exponent part represents the current full-scale range selected.

Since sunlight incidence under AM0 conditions produces around 133,531 lx on this transducer, as derived in Appendix A1, this implies that the upper end of the sensor's measurable light range would be fully used. The OPT4001 includes an automatic full-scale range setting mode that selects the most appropriate range based on current and previous light measurements [67]. This feature significantly enhances ease of use and ensures optimal resolution by dynamically selecting the most suitable range, particularly when the full span is not necessary.

The device's effective resolution is also influenced by the conversion time setting, which defines the integration duration for each measurement [67]. While the register width for mantissa and exponent remains fixed, the number of useful bits within those fields varies with the conversion time, as shown in Table 6. Any unused lower bits are padded with zeros, which effectively reduces resolution for faster measurements but allows for significant power savings or higher-speed data acquisition when needed.

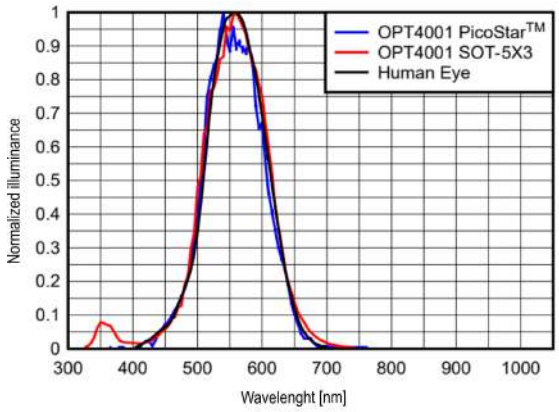
The continuous mode was selected for three reasons: it is the simplest to implement in software; it would be complicated to wire each interrupt line when multiple transducers are used; and since there are many conversion times available, the ADS task running can be set to one of the options to increase efficiency. Although consumption is already low in continuous mode, the device could be driven into power-down mode during an eclipse, for example, and bring it back on once the light is incident. A conversion time of 100 ms was selected as a reasonable compromise between speed and accuracy, with the corresponding effective resolution highlighted in Table 6.

Other features are also available, such as the output register Cyclic Redundancy Check (CRC) and counter, as well as I2C Burst Read Mode and General-Call Reset Command [67], but were not explored.

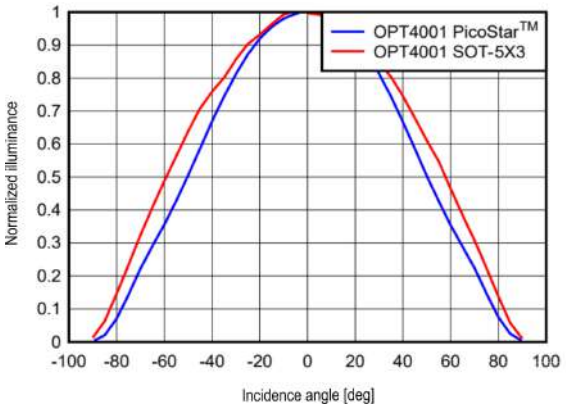
Table 6: Effective illuminance resolution of OPT4001, for each conversion time and light range scenario (adapted from [67]).

Conversion time	**Mantessa effective bits**	**Light range** (first row) — **Exponent** (second row) — **Resolution** (remaining rows)								
		459 lx	918 lx	1835 lx	3670 lx	7340 lx	14680 lx	29360 lx	58720 lx	117441 lx
		0	1	2	3	4	5	6	7	8
600 µs	9 bits	896 mlx	1.792 lx	3.584 lx	7.168 lx	14.336 lx	28.672 lx	47.344 lx	114.688 lx	229.376 lx
1 ms	10 bits	448 mlx	896 mlx	1.792 lx	3.584 lx	7.168 lx	14.336 lx	28.672 lx	47.344 lx	114.688 lx
1.8 ms	11 bits	224 mlx	448 mlx	896 mlx	1.792 lx	3.584 lx	7.168 lx	14.336 lx	28.672 lx	47.344 lx
3.4 ms	12 bits	112 mlx	224 mlx	448 mlx	896 mlx	1.792 lx	3.584 lx	7.168 lx	14.336 lx	28.672 lx
6.5 ms	13 bits	56 mlx	112 mlx	224 mlx	448 mlx	896 mlx	1.792 lx	3.584 lx	7.168 lx	14.336 lx
12.7 ms	14 bits	28 mlx	56 mlx	112 mlx	224 mlx	448 mlx	896 mlx	1.792 lx	3.584 lx	7.168 lx
25 ms	15 bits	14 mlx	28 mlx	56 mlx	112 mlx	224 mlx	448 mlx	896 mlx	1.792 lx	3.584 lx
50 ms	16 bits	7 mlx	14 mlx	28 mlx	56 mlx	112 mlx	224 mlx	448 mlx	896 mlx	1.792 lx
100 ms	**17 bits**	**3.5 mlx**	**7 mlx**	**14 mlx**	**28 mlx**	**56 mlx**	**112 mlx**	**224 mlx**	**448 mlx**	**896 mlx**
200 ms	18 bits	1.75 mlx	3.5 mlx	7 mlx	14 mlx	28 mlx	56 mlx	112 mlx	224 mlx	448 mlx
400 ms	19 bits	0.875 mlx	1.75 mlx	3.5 mlx	7 mlx	14 mlx	28 mlx	56 mlx	112 mlx	224 mlx
800 ms	20 bits	0.4375 mlx	0.875 mlx	1.75 mlx	3.5 mlx	7 mlx	14 mlx	28 mlx	56 mlx	112 mlx

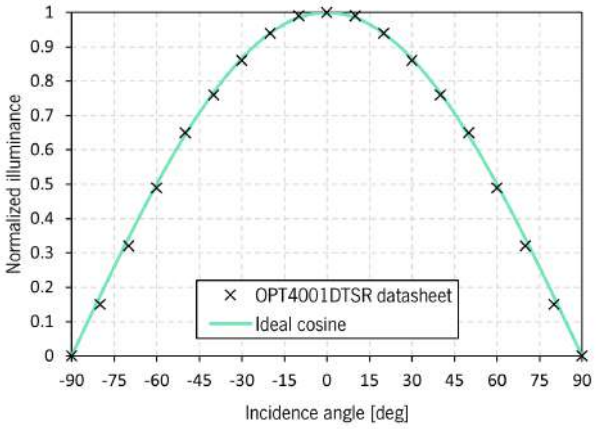
Analysing the OPT4001's behaviour regarding wavelength, it presents a spectral response close to the human eye peaking around 550 nm [67], as shown in Figure 14a. This behaviour is common in most ALSs. Regarding incidence angle, the device response differs noticeably between variants [67], as perceived in Figure 14b. However, the SOT-5X3 variant possesses a very similar response curve to the ideal cosine, as seen in Figure 14c, with a Root Mean Square Error (RMSE) of 0.012.



(a) Spectral response.



(b) Angular response.



(c) Cosine-like behaviour.

Figure 14: OPT4001 response plots (adapted from [67]).

3.3 Array system design

This prototype design began with the selection of key properties of the pyramid, including the number of sides and their inclination. Looking back at the literature review performed on this topic, most CSSPA possessed four sides, which seemed a reasonable compromise between the volume needed and theoretical accuracy. Moreover, as seen in the previous section, the OPT4001 has four configurable I2C addresses, meaning that all four transducers placed on each CSSPA could all be connected to the same I2C bus. Regarding the inclination of said pyramid sides, the literature presented several distinct values. At the time, no rationale was placed on selecting it to be of 45 degrees, besides being in the half of the set of possible values, which seemed enough to validate the CSSPA, and facilitate the Sun vector determination, since sine and cosine values are the same. Therefore, with this inclination, a total A-FOV of 90x90 degrees was achieved.

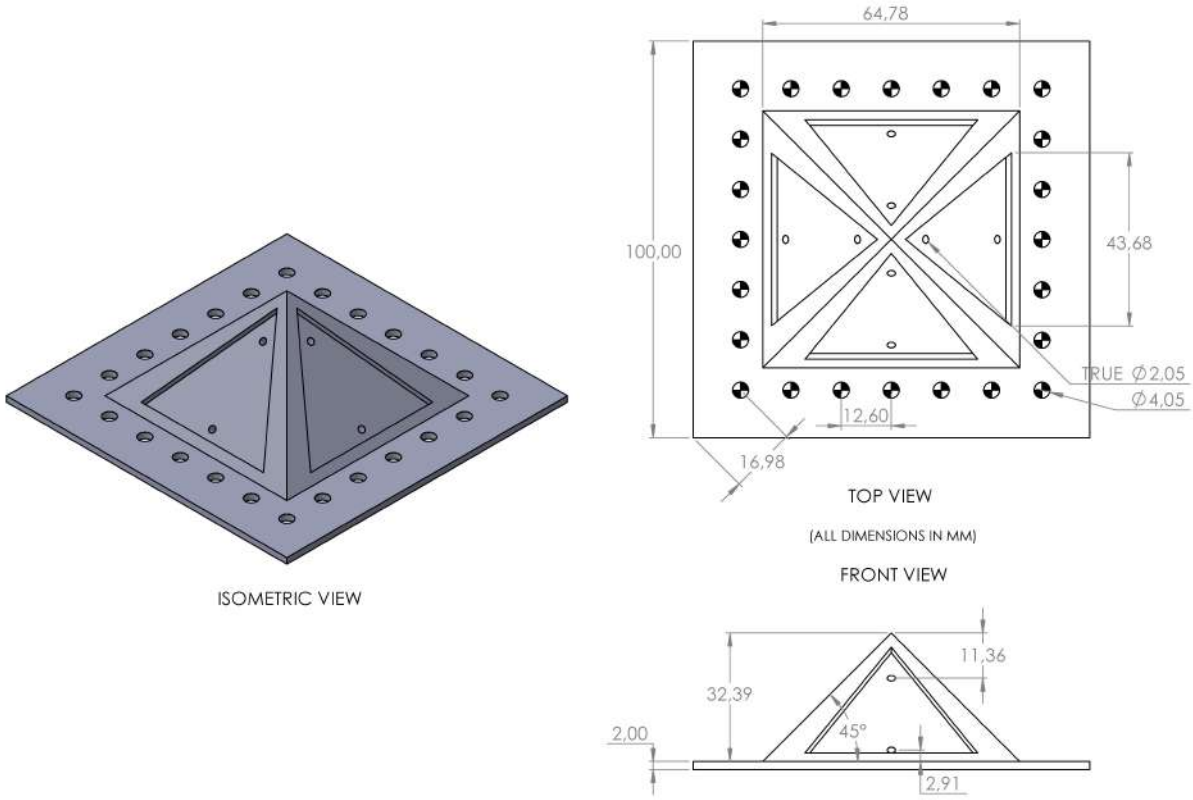


Figure 15: Proof-of-concept prototype support.

The pyramid model was decided to have a 65 mm edge. To fulfil the 45-degree constraint of the faces, its height was calculated to be approximately 33 mm. Its base was added to a 10x10 cm plate, with 4 mm holes to allow for cable management and screw fixing. On each face, 2 mm cutouts were made to allow each OPT4001 breakout board to be placed together with 2 mm holes to align and secure them. Figure 15 shows the design dimensions as explained. This model was designed from scratch using SolidWorks, and once ready, it was 3D-printed with Polylactic Acid (PLA) because of easy handling, residual cost and quick manufacturing.

The next step was to design the OPT4001 breakout PCBs. This process was achieved in two phases: first, the electrical schematic of the PCB was drawn, with all components wired as intended; and then, the physical disposition of these elements was arranged on the desired PCB layers.

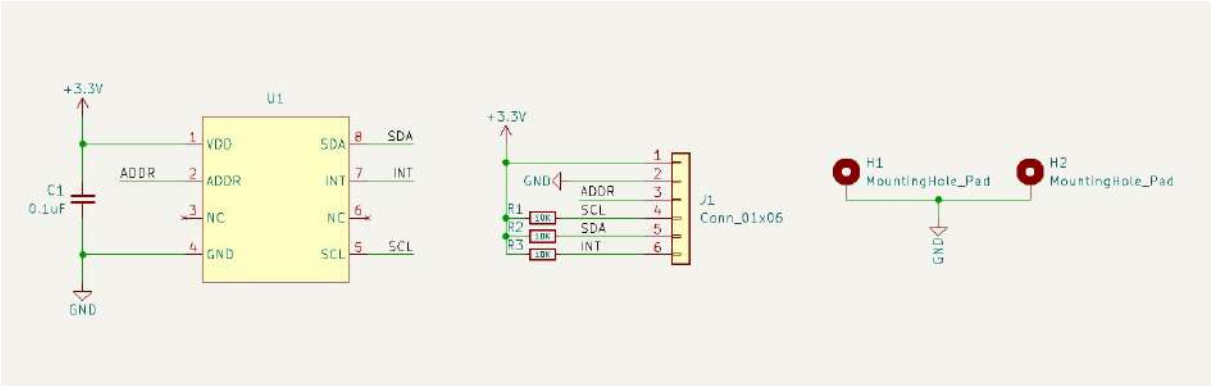
The circuit schematic was very simple. It started by importing the OPT4001 symbol, together with a decoupling capacitor and some pull-up resistors for the I2C and interrupt lines, as recommended by the datasheet [67]. All pins from the sensor were wired to a 1x6 pin header connector for an interface, allowing the I2C address to be set later. Lastly, the two mounting holes required were also incorporated, with their pads grounded. This resulted in Figure 16a.

The PCB design usually starts by defining a set of rules and properties, such as trace width and spacing, via diameter and spacing, number and type of layers, and many others. These depend on several factors, including the electrical requirements of the circuit, the manufacturing capabilities of the PCB manufacturer, thermal and mechanical constraints, and signal integrity needs. For this simple PCB, manufacturing was the most important property, so a standard 0.25 mm trace width on a two-layer copper board was selected.

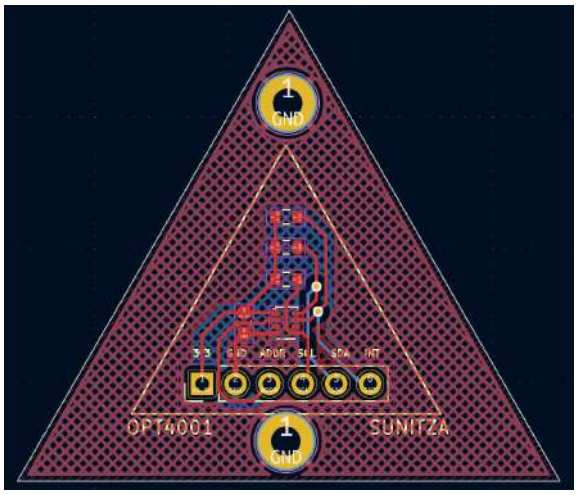
The PCB dimensions were selected based on the pyramidal support, with the edges reduced by 2 mm, to ensure the board fits inside the support, even with relaxed tolerances. Then with the properly defined holes, the PCB could be screwed in place. In KiCad, the OPT4001 footprint was imported and placed in the centre of the front plane, with the decoupling capacitor positioned relatively close to its power pads, following good practice but without obstructing the sensor. This disadvantage is mitigated by selecting all passive components in the small 0603 Surface Mount Device (SMD) form factor. Finally, the remaining unused space in each copper layer was filled with ground planes to improve signal integrity. Some information was added to the silkscreen layer, such as pin labels and the project's name "SUNITZA". The result is presented in Figure 16b, with its 3D rendering in Figure 16c. After exporting the Gerber files, they were sent to the PCB manufacturer.

Once ready, these parts were assembled into the designed prototype. First, all electronic components were soldered into their respective position using a reflow oven, followed by some wires being manually soldered into the pin connectors vias for power, communication and I2C address setting. Then, each PCB was secured in its respective position with M2 screws. The final result is pictured in Figure 17a.

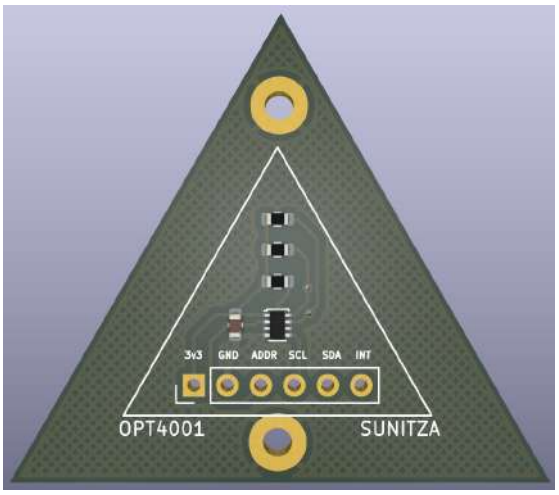
A MCU was required for interfacing the OPT4001 transducers. During the testing procedures, it continuously reads new light measurements from each sensor and sends them over Universal Serial Bus (USB) to the computer, where they can be displayed and processed. An Adafruit Feather M4 Express development board was used, which features an ATSAMD51J19 Cortex-M4 32-bit chip, capable of high performance with low power consumption [68]. The most important features include USB native support, with an USB bootloader and serial port debugging, six SERCOM ports over 21 General Purpose Input/Output (GPIO) pins, and floating-point support [68]. It also supports CircuitPython, an open-source embedded language with Python syntax, targeted toward educational purposes, which simplifies programming [68]. This MCU and language are featured in multiple satellites, such as PROMETHEUS-1.



(a) Circuit schematic.

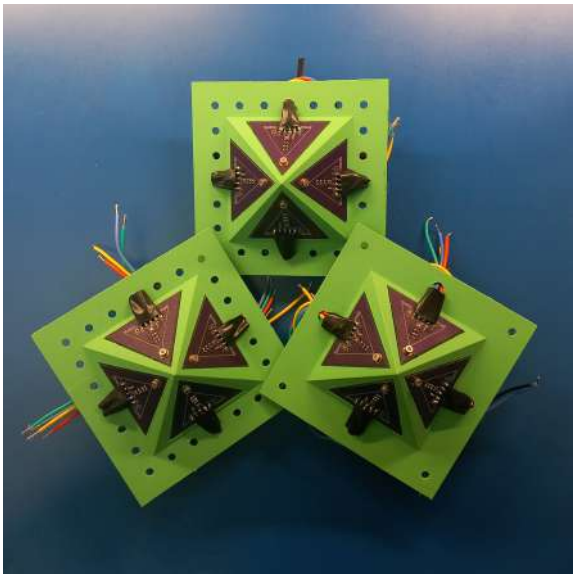


(b) PCB layout.

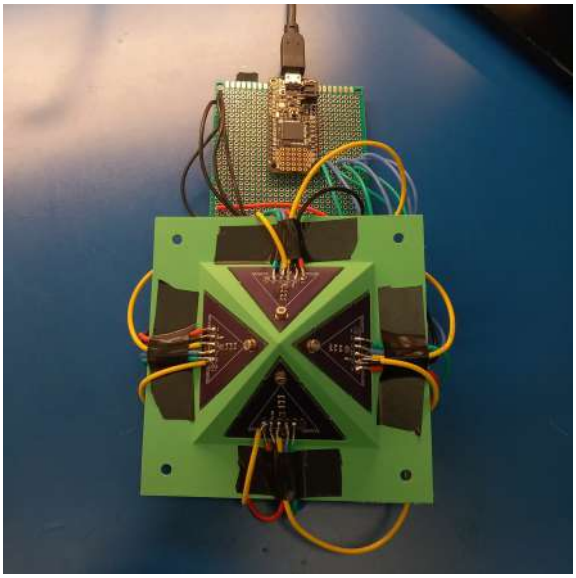


(c) 3D rendering.

Figure 16: OPT4001 breakout PCB.



(a) Three prototypes finished.



(b) Connection with the MCU.

Figure 17: Proof-of-concept prototype assembly.

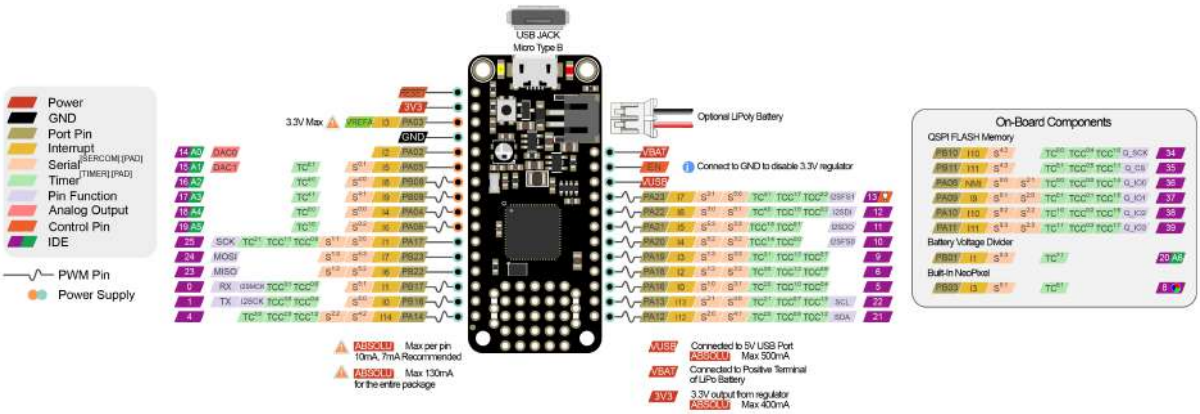


Figure 18: Feather M4 board pinout (adapted from [68]).

Examining the Feather M4 pinout on Figure 18, the main I2C bus is connected to pins PA12 and PA13, corresponding to the SDA and SCL pins, respectively. However, other pin pairs could be employed, such as PA04 and PA05, or PB08 and PB09. Additionally, this board can supply a 3.3 V output from a linear regulator up to 400 mA, which is more than enough to power the transducers. Based on this, the PCB wires were soldered to the respective pins on the Feather M4 board, which was held in a protoboard, as shown in Figure 17b.

Looking at the firmware side, the Feather M4 comes pre-loaded with a Universal Serial Bus Flashing Format (UF2) bootloader, which appears as a USB storage key once plugged into the computer. The bootloader was updated to its latest version by consulting the CircuitPython website, downloading the most recent safe file compatible with the Feather M4, and dragging it to the board's USB drive, which automatically updated it. This simple process was repeated to flash the board with CircuitPython's firmware.

Regarding software, an OPT4001 CircuitPython driver was required to configure the transducer and take measurements. Fortunately, an almost complete one was already available [69], which defined an OPT4001 class whose attributes were the main settings presented in the previous section: the I2C bus where the sensor is connected and its chosen I2C address, the sensor variant, range, conversion time and operating mode, but also some of the remaining features not used. This class implemented some low-level code to read and write to the OPT4001 designated registers via I2C, as well as to convert the digital values in the output registers to the measured illuminance in lux [69].

Particularly in the latter, this process starts by retrieving the measurement parts: the exponent located at bits 15-12 of register 0x00; the mantissa Most Significant Bit (MSB) at bits 11-0 of the same register; and mantissa Least Significant Bit (LSB) at bits 15-8 of register 0x01. Then, with Algorithm 5, the measurement value in lux can be calculated from this representation [67].

Algorithm 5 OPT4001 conversion from a measurement digital representation to its illuminance value.

1 $mantissa \leftarrow (mantissa_msb << 8) + mantissa_lsb$ ▷ formulate the 20-bit mantissa
2 $adc \leftarrow mantissa << exponent$ ▷ formulate the 28-bit representation of the measurement
3 $lux \leftarrow adc \times 0.0004375$ ▷ convert it to the illuminance value
4 **return** lux

A custom class named Pyramid was implemented to manage four instances of the OPT4001 class, one for each transducer on the prototype pyramid. This new class inherited the same attributes as the OPT4001 class, allowing it to be initialised with the same settings, except for the address, which now requires an array containing all four possible values for each transducer. Additionally, each transducer's normal vector, represented by its azimuth and elevation angles, can be provided, allowing one method to calculate the Sun's orientation, if necessary. This class also has a method to retrieve all four measurements, which is the most relevant, as well as others, to determine the true light direction based on Euler angles and to set all transducers to low-power mode. Most attributes have default parameters, but they can be initialised differently as desired.

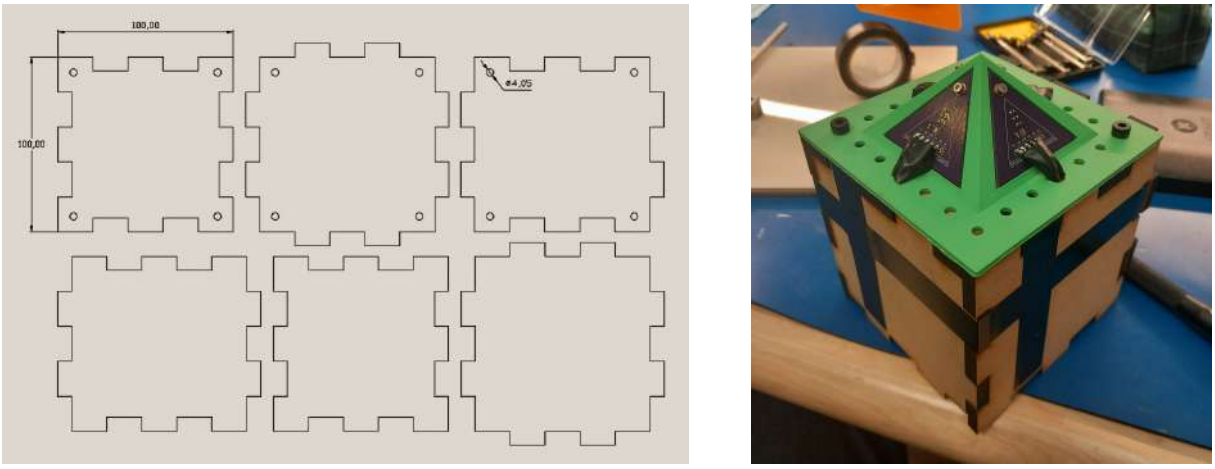
Since onboard storage is reduced, the collected data was sent to a computer where it could be post-processed. To do this, the pre-installed CircuitPython USB Communications Device Class (CDC) driver was enabled during the MCU boot so that a serial connection could be established between the two devices. On the computer side, a small Python script was developed using the same driver to receive the data, timestamp it, and save it to a Comma-Separated Values (CSV) file.

3.4 Sun vector determination methodology

The testing methodology to assess the accuracy of Sun vector determination consisted of rotating the assembled CSSPA in one axis while a stationary light source pointed to its centre. This approach was considerably simpler than those reviewed in the literature, which involves two axes rotating. Additionally, since the light source direction is known, at each set of measurements, the calculated Sun vector can be compared to its expected direction, and register the system accuracy.

The testing setup was designed from scratch, as it was the only feasible option due to time constraints, and modelled using AutoCAD and SolidWorks. First, the pyramid was attached to a wooden cube with a 10 cm edge to facilitate handling. Each cube face was 2D-sketched, depicted in Figure 19a, and later cut into 5 mm plywood sheets using a laser-cutting machine. Before assembling it, one CSSPA was screwed to one of the faces using M4 screws, and the Feather M4 was placed on the opposite side, with the remaining

length of the wires that crossed the sheet routed through added holes. The cube was held in place with tape for a smooth finish, as seen in Figure 19b.



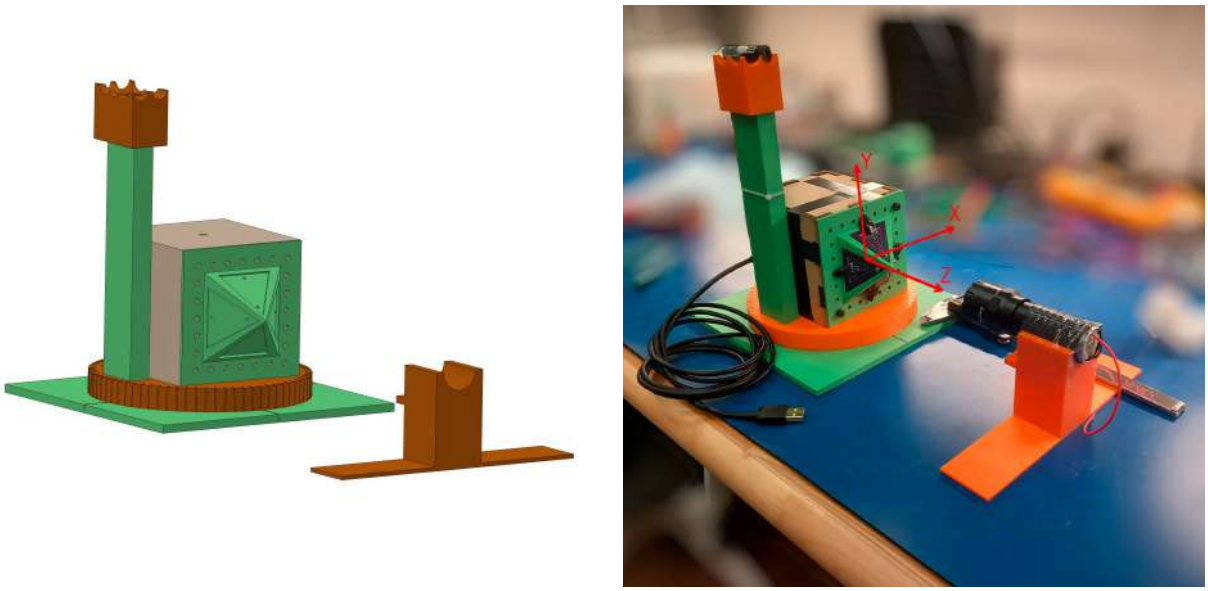
(a) 2D sketch of each face.

(b) Prototype on a face.

Figure 19: Cube assembly for proof-of-concept testing.

Secondly, a spinning disc table was designed to rotate the CSSPA. It was composed by four separate parts: a laser holder; its elevation bar; the rotating disc that also held in place the cube, the pyramid and the bar; and the static base. The base cutout for the disc was designed with tight tolerances to prevent translation and included a marking dent that served as the reference direction. Together with the multiple dents made on the disc, five degrees apart from each other, they allowed for coarse rotation measurements. However, this design did not have sufficient resolution to quantify the angle rotated from each measurement reliably, hence the need for a laser. By pointing it at a distant surface, smaller angle changes can be measured by observing its displacement on the surface instead. The farther the distance, the bigger the displacement, however, to obtain a large range of disc rotations, the measurement set needed to be divided due to room size constraints. In other words, it was planned that the laser would point to the start of the wall, and then the disc would rotate until the laser was pointing at the end of the wall. Finally, without moving the disc, the laser would be re-aligned with the start, allowing another set of measurements to take place. Several cutouts, spaced 35 degrees apart, were made at the laser holder to facilitate this procedure. The elevation bar was needed so that the laser could point to the wall without obstructions.

Lastly, it was planned to simulate the Sun with a flashlight, providing a cheap and nearly collimated light source. The Nightstick NSP-420 flashlight was selected due to its adjustable beam, traceable LED selection, and ANSI FL1 standard compliance, which provides useful data on the light source, including



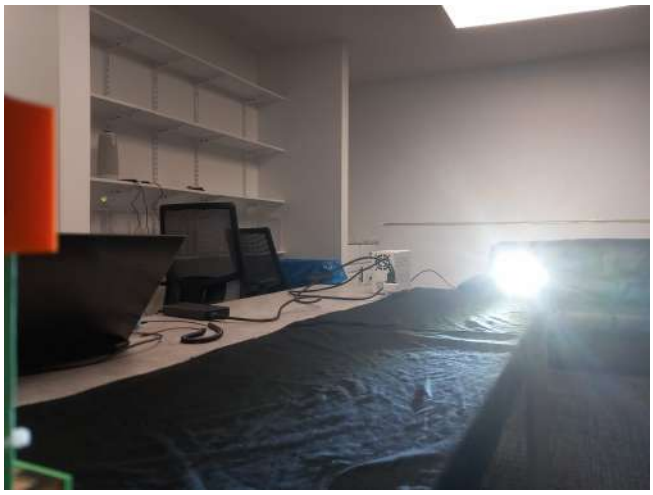
(a) 3D model design. (b) Printed assembly with axes definition.

Figure 20: Proof-of-concept test setup parts.

its luminous flux of 275 lm and peak beam intensity of 4445 cd [70]. This model is normally supplied with three AAA batteries; however, the internal circuit has been modified to be powered by an external power supply, allowing for continuous and constant illumination. A static support was designed so that the flashlight was colinear with the array boresight when the latter was in its initial position.

All parts were 3D-printed with PLA, resulting in Figure 20. The visible coordinate frame was defined on the pyramid to serve two purposes: to define the rotations and to properly align the normal vectors of the transducers. The Z-axis pointed to the light source in the initial position, the Y-axis was the axis of rotation, and the remaining X-axis completed the triad. To help identify the transducers, each label followed the ±X/Y convention. For example, the +X device pointed to the first quadrant of the plane formed by the X and Z axes. With this formulation, it was expected that the ±X transducers would measure an opposite change in illuminance upon rotating the disc. In contrast, the ±Y devices would have detected the same illuminance. It was also defined that the I2C addresses of the OPT4001 would be set in ascending anti-clockwise order: 0x44 address to +X transducer; 0x45 to +Y; 0x46 to -X; and 0x47 to -Y.

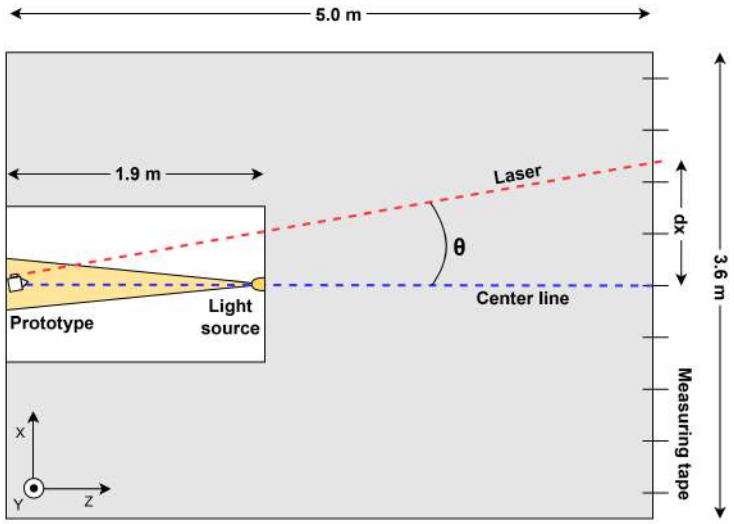
The experimental setup was arranged in a dark room, as illustrated in Figures 21a and 21b, following the schematic and dimensions provided in Figure 21c. The testing procedure began by supplying the flashlight with 3.3 W of power and aligning the disc position using a laser and marked dents. The disc was rotated from -85 to +85 degrees, with measurements taken in six sets. In each set, ten measurements were recorded at each position to allow for averaging, and the data was logged on a computer. Between



(a) Laser pointing to measuring tape.



(b) Light pointing to prototype.



(c) Schematic and dimensions.

Figure 21: Proof-of-concept test setup assembled.

measurements, the disc was rotated to achieve an approximate 20 mm displacement on the measuring tape. The exact distance at which the laser was pointed was also recorded to determine the true incidence angles for each measurement. With the specified dimensions, this displacement corresponded to an approximate rotation of 0.2 degrees.

Assessing the accuracy of the Sun vector determination involved calculating the angular error between the Sun vector estimated from the measurements and the expected vector for each point in the retrieved dataset within the A-FOV. The measured Sun vector was estimated using the LS method described in Equation (44) of Section 2.2.2. Subsequently, the expected Sun vector needed to be calculated. Due to calendar constraints, this prototype was tested before the literature review was complete. As a result, two approaches for the expected behaviour were initially developed, which proved to be more complex than

those found in [36].

The first approach, termed “cosine”, involved calculating the expected illuminance for each sensor using the ideal cosine model, similar to what is presented by OPT4001 in the previous section, while correcting for potential non-point-source light effects. The second approach, called “polyfitted”, involved fitting each sensor response curve to a fifth-degree polynomial. After determining the expected illuminances using both methods, the Sun vector was estimated using the same LS method. The rationale was to develop two distinct methods for estimating the expected Sun vector: one based on theoretical principles and the other on practical considerations with some calibration, and then compare their results.

Delving more closely to the first approach, the correction mentioned determines the true incidence angle for each transducer. In a point source light model, the +X transducer’s incidence angle, for example, is simply 45 degrees added to the incidence angle on the CSSPA. In other words, the transducer’s normal vector is just tilted according to the face inclination, because the light is very far and one can assume the transducer is placed on the rotation axis. However, if the distance between the axis and the transducer is considerable, around 70 mm in this prototype, then the assumption breaks because there is a noticeable translation compared to the source, thereby inducing errors. To solve this, a forward kinematics problem was formulated with specific coordinate frames: one defined on the light source aligned with its boresight; one on the disc centre aligned with the rotation axis; and one on each transducer aligned with the normal vector, as shown in Figure 22. Then, concerning the setup dimensions and definitions, the transformation matrices between frames were constructed and used to determine the normal vector represented in the light source frame. From here, the true incidence angle of each transducer could be easily discovered.

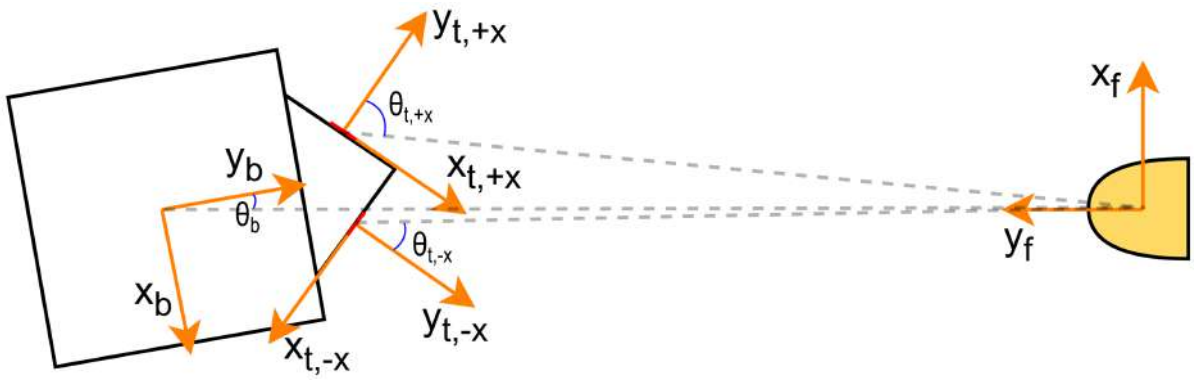


Figure 22: Temporary frames defined for calculating the true incidence angle for each transducer.

3.5 Sun vector determination results analysis

After finishing all six sets, the post-processing step began by merging all the measurements into a single, uniform dataset. The incidence angle of the CSSPA was calculated based on the laser's position relative to the measuring tape and added to each data point. The illuminance results were plotted in Figure 23.

Overall, on the X axis, the maximum intensity is around ±38 degrees, which is a considerable shift from the expected 45. This behaviour could be explained by the oversized pyramid, which manifested some additional translation, as foreseen earlier, or by setup imprecisions. Manufacturing misalignments could also cause the PCBs or transducers to tilt during printing, soldering, or assembly. Another noticeable effect was that beyond the ±35 degrees interval, reflected light coming from the nearest wall created a non-zero measurement. Consequently, the A-FOV also shrunk.

Some discontinuities between sets were present, caused by the manual reset of the setup. The maximum values for the ±X transducers were slightly different, indicating variations in each device's response.

Regarding the ±Y transducers, the measured illuminance was very similar, except beyond the ±65 degrees interval, where the reflection phenomenon started to appear. The maximum intensity is significantly lower than that of the ±X transducers, as the light source was never colinear with the normal vectors.

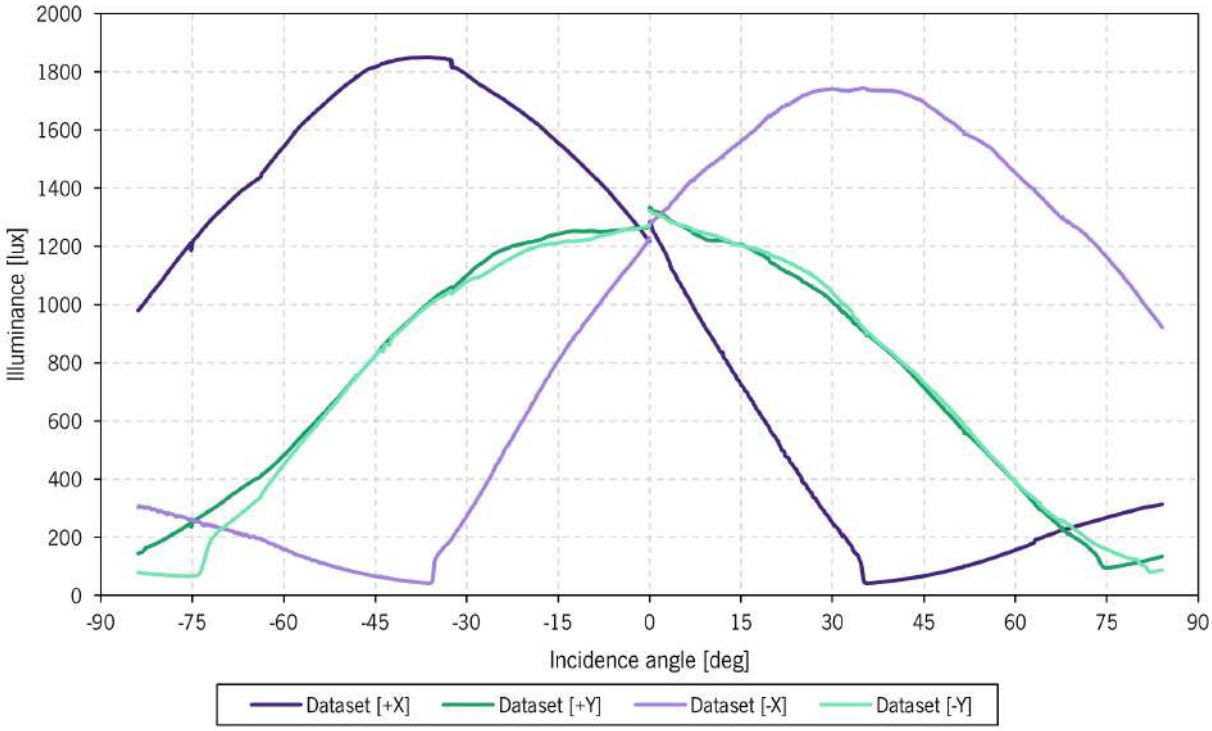


Figure 23: Illuminance measured by the four transducers of the proof-of-concept prototype.

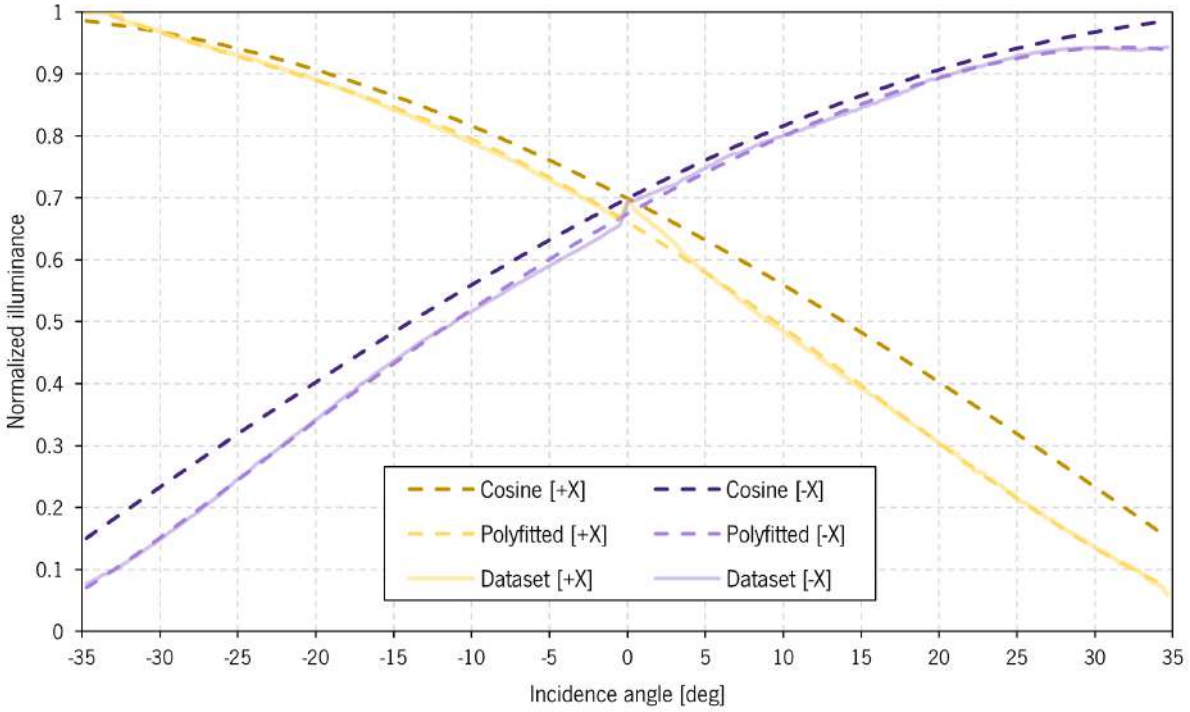
Looking more closely at the domain between ±35 degrees, after normalisation, shown in Figure 24,

the measured data was slightly off the cosine model, but once fitted to a fifth-degree polynomial, there was a clear approximation between the dataset and its expected values, in both axes. This likeness suggests that the data was overfitted to the model, which could lead to poor generalisation and inaccurate new predictions. This assumption is at least partially true because the transducer's translation was not explicitly defined, unlike in the cosine model. Furthermore, it is also clear that the divergence between models was more pronounced as the incidence angle approached the T-FOV edges, where the ideal cosine model tended to underperform in terms of non-linearity.

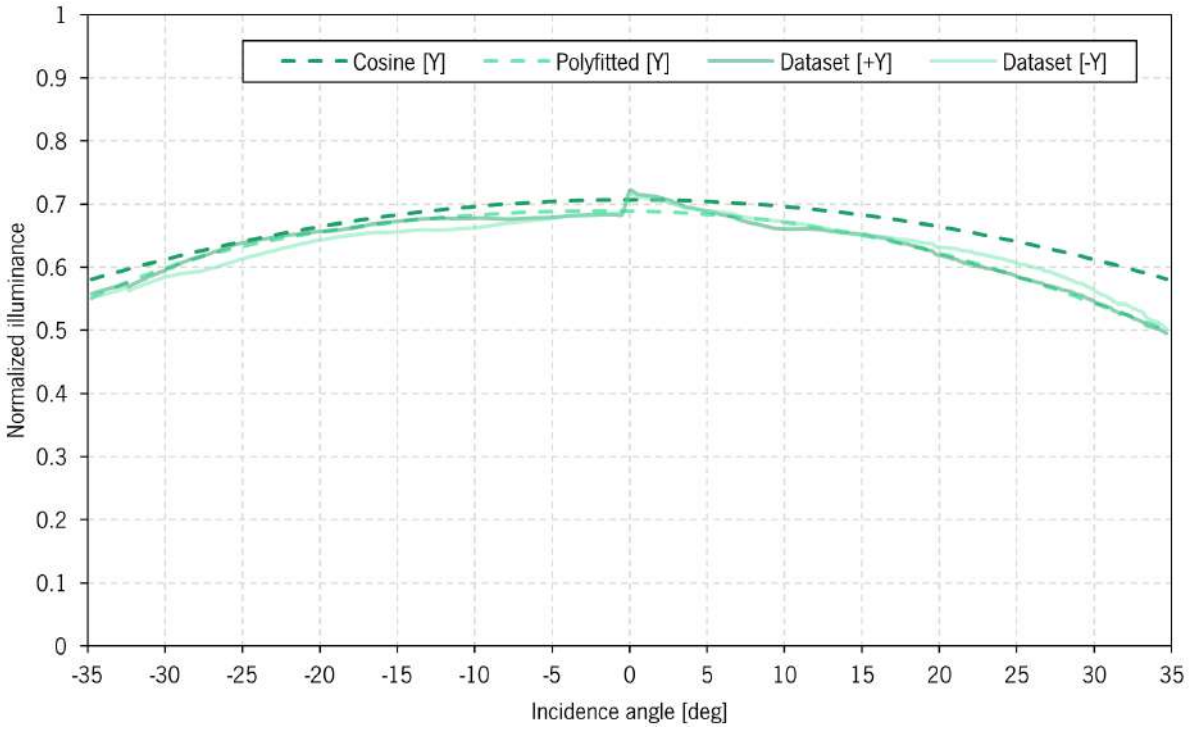
This illuminance behaviour disparity was also reflected in the Sun vector determination, as expected, specifically in the X and Z components of the vector, as depicted in Figure 25. The Y component remained both equal and null for both models because the ±Y transducers' measurements, which are solely responsible for defining this component's output, were equal across the domain.

Analysing the angular error of Sun vector determination, or in other words, the angle between the measured and expected Sun vectors, the cosine model performed with an acceptable accuracy only in a very short interval, steadily increasing up to 6 degrees, as portraited on Figure 26a. This evolution was explained by the fact that while one X transducer's normal tends to become collinear with the light source, which reduces the illuminance error, the other moves to a zone with a considerable one. On the other hand, for the polyfitted model, the error was relatively bounded, with values almost below 1 degree. Looking closely at the distribution of the error on the said model, depicted on Figures 26b and 26c, only 19 out of 300 were above the 1-degree limit. However, on the cosine model, the error is uniformly distributed within the range, with approximately 40 to 70 samples in each interval.

In general, the cosine model presented a mean error of 3.34 degrees, with a standard deviation of 1.66 degrees. The polyfitted model improved these values to 0.60 and 0.27 degrees, respectively.



(a) X-axis transducers.



(b) Y-axis transducers.

Figure 24: Normalised illuminance comparison between dataset and models (cosine/polyfitted).

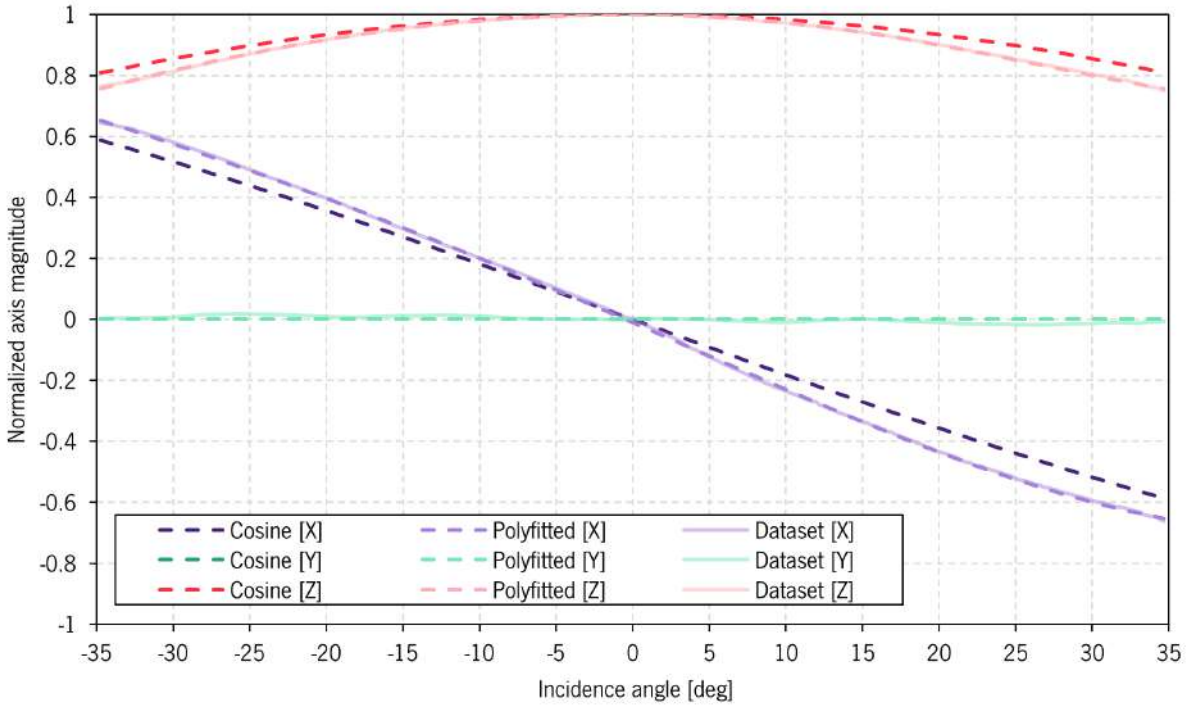
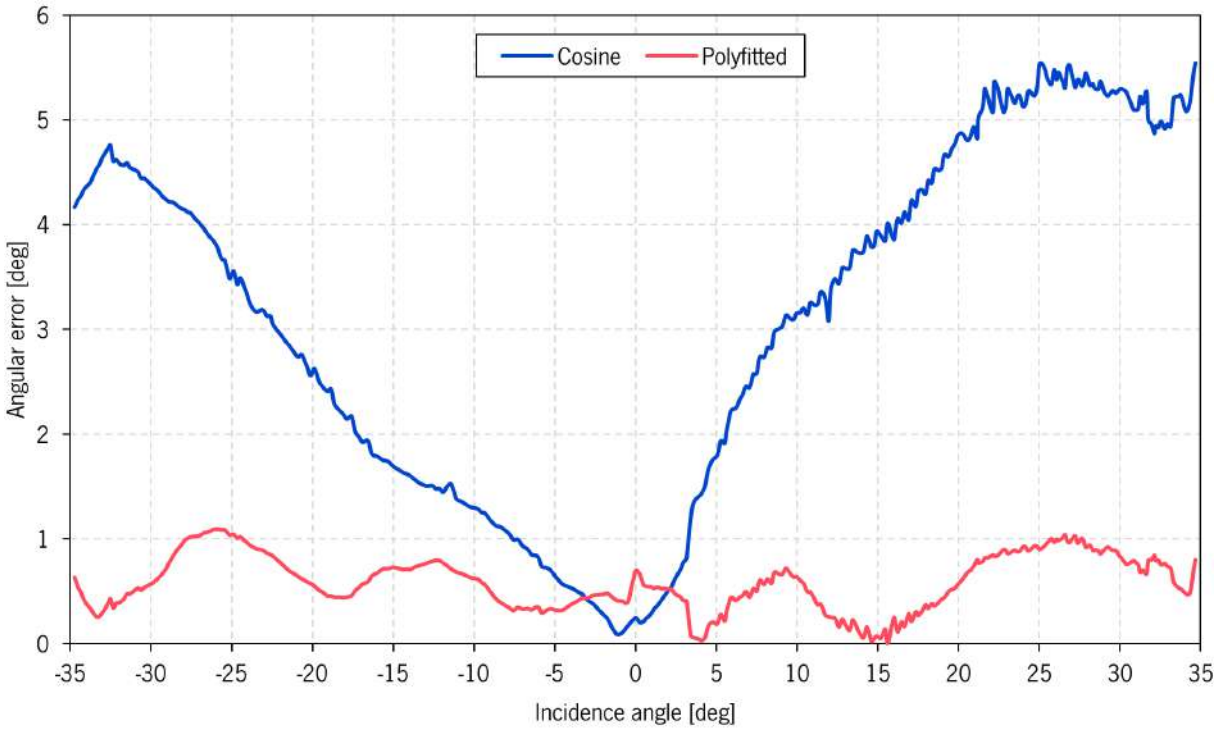
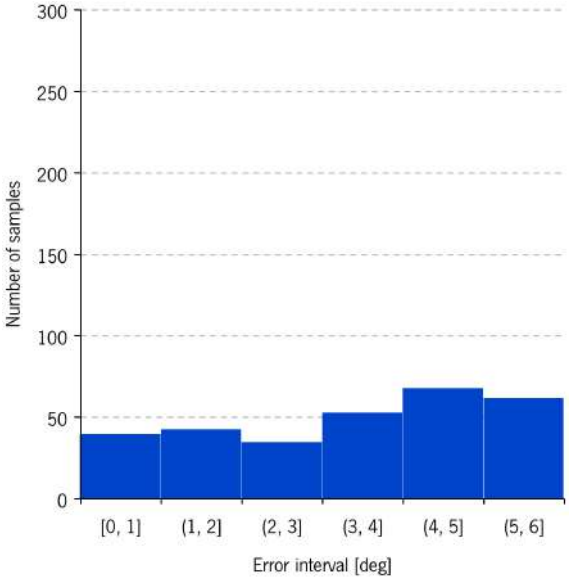


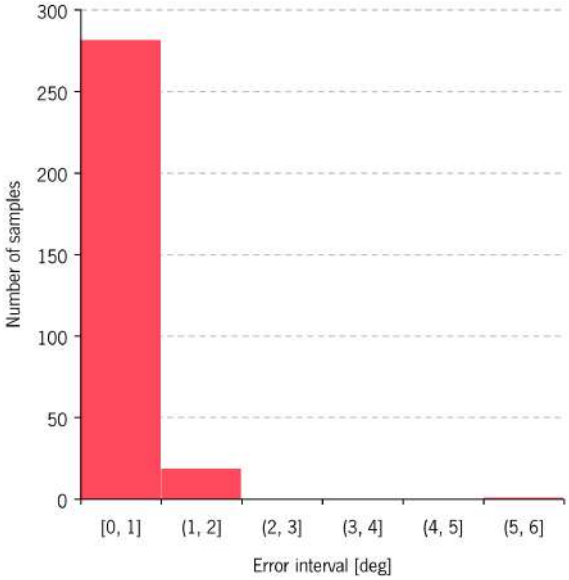
Figure 25: Normalised Sun vector determination comparison between dataset and models.



(a) Error evolution with incidence angle.



(b) Cosine model error histogram.



(c) Polyfitted model error histogram.

Figure 26: Sun vector determination angular error of the proof-of-concept prototype.

3.6 Summary

The design process began with the selection of the OPT4001 sensor (Section 3.2), chosen for its balance of high resolution, illuminance range, compact size, cosine-like response, and operating features. Its compatibility with the I2C protocol and versatile configuration options made it particularly well-suited for integration into a multi-sensor array.

In the array design phase (Section 3.3), a four-sided pyramid configuration was implemented to match the number of available I2C addresses, with sensors mounted on custom-designed breakout PCBs. The prototype was fabricated using 3D-printing and integrated with a Feather M4 development board, enabling real-time data acquisition. Following the typology presented in the previous chapter, the CSSPA would thus be classified as a two-axis, coarse Sun sensor with ALSs providing an analog behaviour.

The experimental setup (Section 3.4) employed a custom-built testbed with single-axis rotation and a collimated light source to simulate illumination. Measurements were conducted across varying incidence angles, with angular displacement determined via a laser pointing system. While this approach provided valuable insights, it constrained the evaluation to a single axis and was associated with doubtful accuracy, thereby limiting the comprehensive assessment of the array's performance.

In the data analysis (Section 3.5), Sun vector determination was evaluated using both a theoretical cosine response model and a polynomial-fitted empirical model. The cosine model yielded a mean angular error of 3.34 degrees, with a standard deviation of 1.66 degrees, while the polynomial fit reduced this to 0.60 and 0.27 degrees, respectively, albeit with risks of overfitting. These results underscored the potential of the CSSPA concept when accompanied by careful design and calibration.

A few major improvements were drafted for the next version of the prototype. Its oversized geometry contributed to translation and orientation errors, particularly w.r.t. the light source. A considerable reduction in the pyramid's size was recommended to minimise such discrepancies during testing, as well as for the final version to be placed on the satellite. The experimental setup allowed for light interference and was subject to manual control, resulting in measurement inconsistencies. Following experiments should employ an enclosed, light-tight, and automated setup, ensuring the stable fixation of components to maintain measurement fidelity. Future calibration should be grounded in typical transducer behaviour and conducted individually for each sensor to account for inherent manufacturing variances.

In conclusion, the CSSPA proof-of-concept prototype validated the core design concept and offered a foundation for further refinement. Through the suggested improvements and extended testing, this system can evolve into a robust, low-cost and feasible solution.

Chapter 4

Advanced prototype

4.1 Introduction

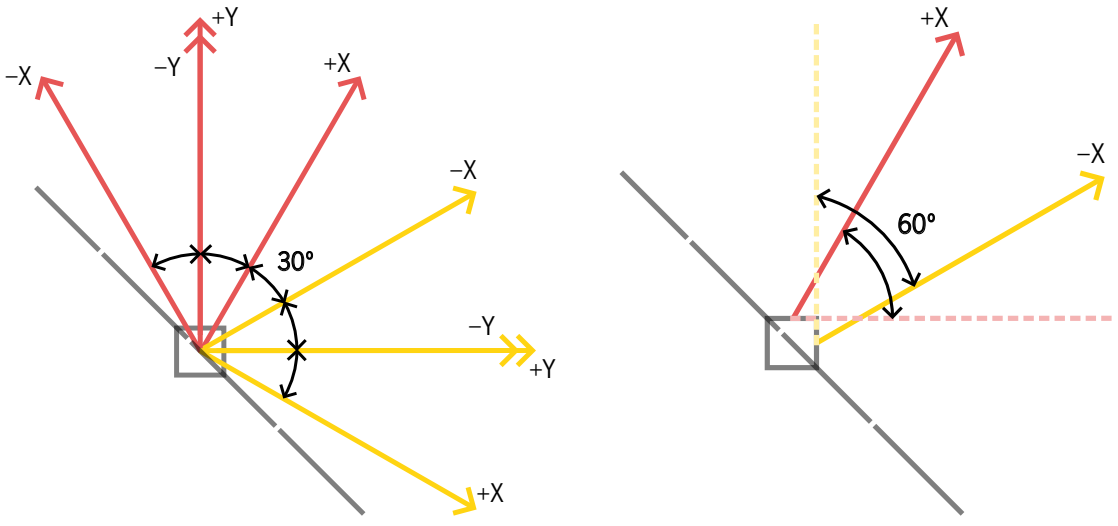
Building on the insights and challenges encountered with the proof-of-concept prototype, this chapter presents the development of an advanced prototype designed to address the limitations and enhance the performance of the proposed CSSPA in terms of accuracy and A-FOV. Fundamentally, the primary objective addressed during this chapter was to improve the sensor design and manufacturability, essentially converging the prototype to an almost ready-to-flight version without compromising its functionality and capabilities. This advanced model also sought to incorporate lessons learned from the previous prototype, calibration setup, and employed procedures, which were almost entirely redefined and upgraded.

The upcoming sections outline key developments in the advanced prototype. Section 4.2 discusses array topology optimisation, focusing on strategic adjustments to enhance symmetry and coverage of transducers, as well as the necessary procedures to validate them. Section 4.3 explores the miniaturised array manufacturing process, detailing the materials and designs used to achieve compactness without sacrificing functionality. Flexible circuits were employed to mitigate some of the complications that arose from reducing the sensor's size. Section 4.4 revisits the Sun vector determination methodology, incorporating improvements that stemmed from previous results and analyses. Lastly, Section 4.5 evaluates the array's performance through testing and calibration, with Section 4.6 providing a summary of the findings, setting the stage for final prototype development in the subsequent chapter.

4.2 Array topology optimisation

This section focuses on optimising the array topology by evaluating the spatial arrangement, orientation and quantity of transducers within the array to maximise coverage for the Sun vector determination. By strategically optimising the CSSPA layout, a complete V-FOV was aimed to be achieved across the attitude sphere of the satellite, thereby enabling the determination of the Sun's position in all possible attitudes.

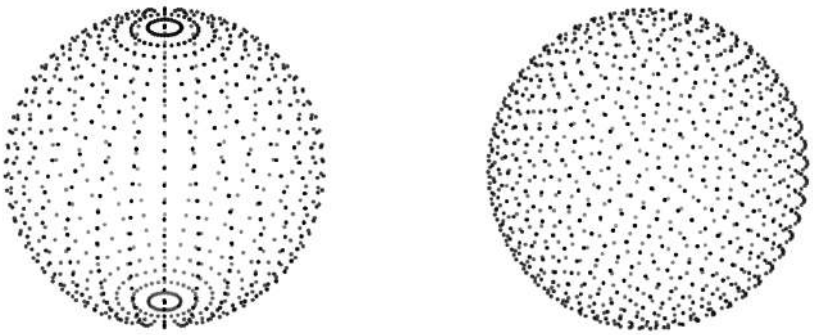
Starting with the evaluation of the pyramid's inclination, several key considerations were immediately apparent. For example, the smaller the inclination, the shorter the pyramid w.r.t. its height. This reduction is important since the CSSPA must be placed on the external faces of the satellite without exceeding the allowed volume. In the scenario similar to the previous chapter, where a squared pyramid with all faces sharing the same inclination is employed, and assuming the T-FOV is constrained to the HPA region of ±60 degrees, the best possible inclination regarding a complete coverage and uniform distribution of transducers is 30 degrees. Two reasons derive this solution: first, it is the only angle that makes transducers evenly spaced in this configuration, as pictured in Figure 27a, where each coloured set of arrows represents the normal vectors present in each pyramid; and second, the outward HPA limit of each transducer is parallel to the face plane, aligning with this physical barrier of the satellite, as shown in Figure 27b. In other words, this inclination enabled a full union region over the plane while maximising the intersection region of transducers. This inclination angle was also commonly reported in multiple literature sources, as noted in Chapter 2.



(a) Even spacing.

(b) Parallel alignment.

Figure 27: Geometry regarding a thirty-degree inclination of transducers.



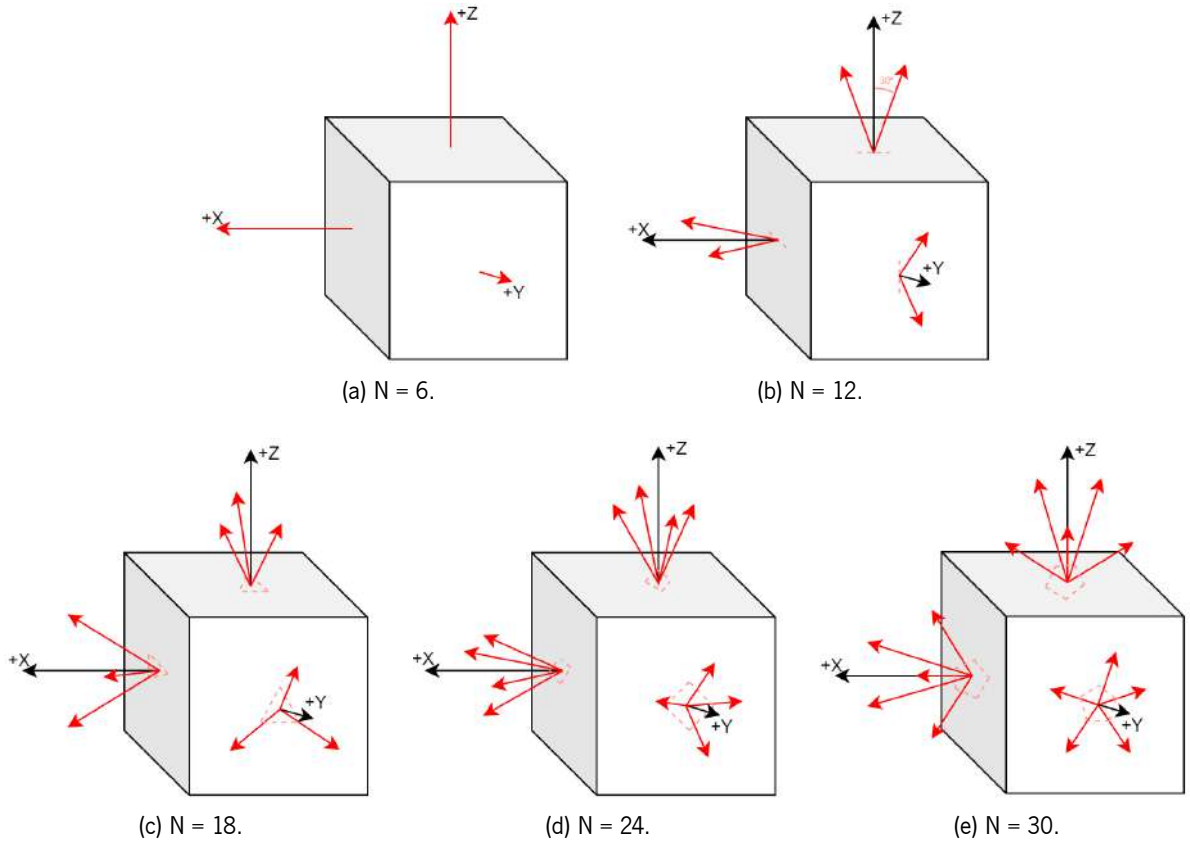
(a) Azimuth/elevation sphere. (b) Fibonacci sphere.

Figure 28: Different approaches to discretise the surface of a sphere.

Although the inclination is an important property of the CSSPA, the number of sides is simultaneously influential. In the proof-of-concept prototype, setting it at four sides was logical. However, the coverage must be evaluated, specifically considering the selected inclination, which, in theory, appears to be the right choice but could be proven otherwise.

The concept of the attitude sphere shall be introduced, which is simply a unit sphere that can be used to represent all possible directions from a spacecraft to objects of interest within its body-fixed frame. To make this useful for practical applications, such as determining whether the Sun direction can be determined for each configuration, it is necessary to discretise (or pixelate) the surface of the sphere, where each pixel corresponds to a unique direction. However, care must be taken in the method of pixelisation to ensure uniform angular resolution. Classical methods, such as evenly varying azimuth and elevation, cluster more points near the poles, leading to uneven weighting across the sphere, as observed in Figure 28a. Two methods for uniformly distributing points appear: the icosahedron-based and the Fibonacci sphere approaches. The first involves inscribing an icosahedron within the unit sphere, subdividing each of its triangular faces evenly, and projecting the resulting points onto the sphere's surface. The second relies on incrementally placing points according to the golden angle, distributing them spirally from pole to pole. This latter process yields an approximately even distribution of points without the need for complex operations, as shown in Figure 28b, so it was therefore chosen.

Five different configurations were proposed to evaluate the satellite's V-FOV, all of which aimed to preserve the uniform and symmetrical distribution of transducers. All configurations shared the same number of CSSPAs employed, with one on each face of the spacecraft, as seen in Figure 29. It is obvious that the hidden faces of the cube also shared this setup. Each configuration differed on the number of transducers used in each CSSPA, from one to five.



(a) N = 6. (b) N = 12.

(c) N = 18. (d) N = 24. (e) N = 30.

Figure 29: Geometry of the proposed sensor configurations.

Note that the first and second configurations, with only one and two transducers, respectively, were not pyramids since they lacked the number of faces to form one geometrically. Additionally, in the first configuration, the transducer was not tilted and, therefore, remained coplanar with the satellite face. This configuration is currently employed in PROMETHEUS-1. Nevertheless, the second one, despite not being a pyramid, had its two transducers tilted at 30 degrees, which was also shared by the remaining three configurations, which were composed of regular triangular, square, and pentagonal pyramids. Each configuration was identified by the total number of transducers used, N, which ranged from 6 to 30.

After calculating the normal vectors of all transducers, the evaluation algorithm for coverage was implemented as follows in Python. For each configuration, the number of failed transducers was successively incremented to assess the satellite coverage, as explained earlier, and also its robustness against sensor damage, primarily induced by radiation. Specifically, for a given total number of failed transducers, multiple random combinations were generated with a Monte Carlo simulation. In each trial, for every direction in the attitude sphere, the algorithm counted the number of transducers that could detect the Sun. The

selected performance metric was the percentage of directions where at least three transducers could detect the target simultaneously. For example, during the evaluation of configuration $N = 24$ with 7 failed transducers, 10,000 random positional combinations for the remaining 17 operational transducers were generated. Next, for each one of those combinations, the algorithm changed the Sun direction to one of the 5,000 possible pixels and counted how many transducers would have the Sun within its HPA. This simulation process was repeated for each scenario to obtain reliable, statistical results. The algorithm saved each trial's parameters and results to a file, including the number of failed sensors by configuration, the calculated coverage percentage, and the number of simulations to facilitate subsequent analysis. Additionally, coverage maps were generated to illustrate how each sensor configuration performed under ideal conditions. Algorithm 6 simplifies the number of loops required to evaluate all scenarios.

Algorithm 6 Sun coverage across the attitude sphere for multiple configurations with failing transducers.

```
1  for all defined configurations do
2      for all number of failed transducers possible do
3          for all Monte Carlo combinations of operational transducers do
4              for all attitude sphere directions do
5                  for all instanced operational transducers do
6                      if the Sun is visible then
7                          increment the counter of transducers for this iteration
8                      end if
9                  end for
10             end for
11         end for
12     end for
13 end for
```

After completing approximately 500,000 iterations in total, over 41 hours of continuous computing were spent on an Intel Core i7-8750H, and the extracted results were compiled in Figure 30. Each configuration was represented by a different set of colours, which were separately identified in the plot legend. Starting with the first configuration, it is clear that the Sun vector can only be determined in a few spots on the attitude sphere, more precisely around 2.6% of its area. As the transducers started to fail, this percentage was reduced by half for each failure. When more than three transducers fail, the Sun cannot be fully determined, so this area becomes nonexistent. This undercoverage underlies the current problem of PROMETHEUS-1, where only a tiny portion of the attitude sphere can fully determine the Sun vector. In the second configuration, although the increase in transducers also increases the percentage of coverage, it still does not provide a complete one, achieving around 94%. Again, once the transducers start to

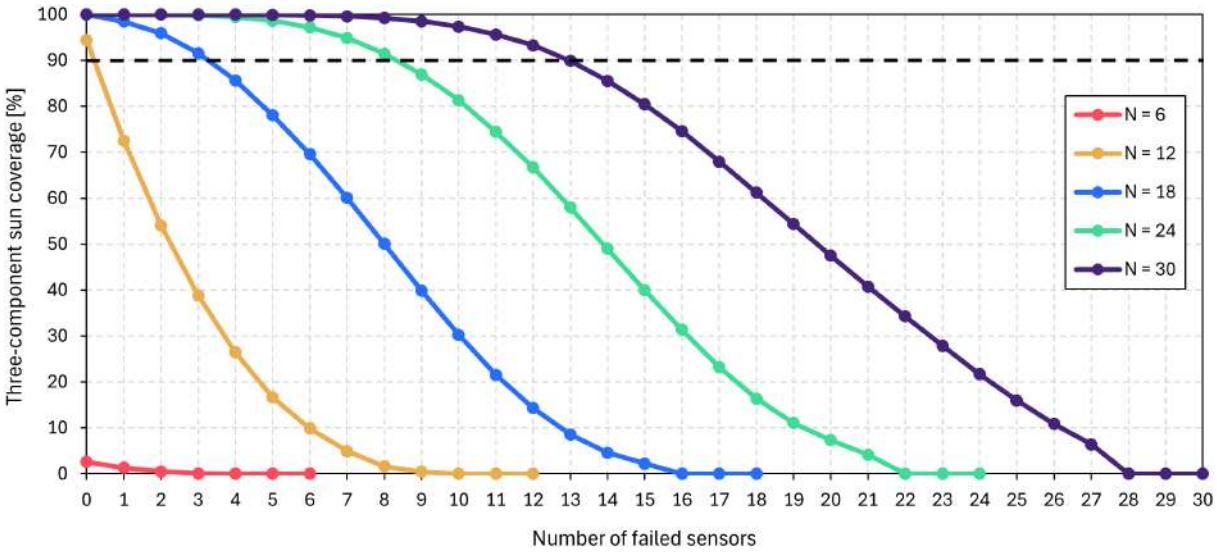
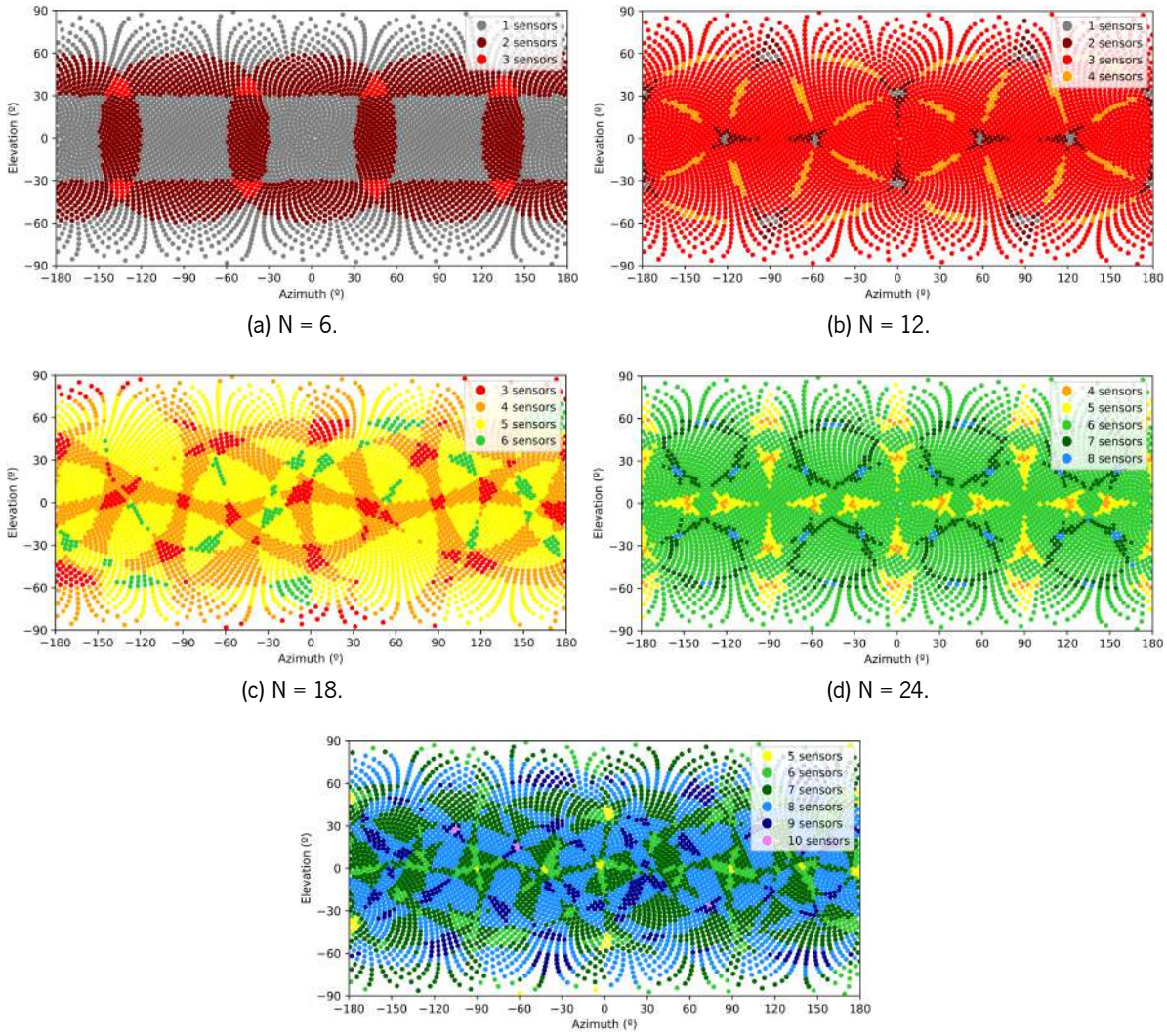


Figure 30: Three-component Sun coverage across the attitude sphere for multiple configurations with failing transducers.

fail, this value decreases exponentially, with more than half of the coverage lost after three failures. The increase in the transducer's density in the following configurations mitigates these behaviours. With three transducers per pyramid, complete coverage is achieved if all sensors are operational. However, once they start to fail, the coverage decreases, managing to stay within the 90% coverage acceptance threshold until three failures and 50% coverage with eight failures. The fourth and fifth configurations present robust and redundant results. The fourth configuration achieves 100% coverage with three or fewer failures, above 90% coverage with up to eight failures, and 50% coverage with at least 13 failures. These results are further improved in the fifth configuration, with full coverage achieved with seven or fewer failures, above 90% coverage with up to 13 failures, and half the coverage possible with at least 19 failures. Note that these results are based on an average, as if the transducers' failures occur in a more localised region, the coverage percentage could drop more quickly, since observability would be lost in specific regions. On the other hand, if the failures are spread uniformly across the attitude sphere, the coverage could remain intact.

Figure 31 shows the number of transducers that would have the Sun vector within its HPA, for each direction of the attitude sphere, here represented by its 2D projection, and for each proposed configuration without transducer failures. It essentially provides a distribution map of the results of Figure 30, specifically for the first data point of each configuration. In the first, Figure 31a shows that the three-component Sun coverage is provided at the vertices of the satellite. However, in the next configuration, the areas without coverage are located in the middle of the edges of the cube, as portrayed in Figure 31b. These two specific

zones are evident as the areas where there is a greater and lesser intersection of transducers' T-FOV, respectively. With three sensors per pyramid would suffice if redundancy was not cared for. Additionally, with one base edge aligning with a face edge, a uniform distribution of transducers is not possible, as pictured in Figure 31c, which is an additional disadvantage of triangular pyramids for this application. However, there is some degree of symmetry, a behaviour also shared with pentagonal pyramids, as seen in Figure 31e. Therefore, the 30-degree inclined square pyramid filled all criteria to be selected, providing a complete and redundant coverage of the spacecraft attitude sphere for Sun vector determination, but also a symmetric and almost uniform one, as depicted in Figure 31d. It was possible to achieve these results with fewer transducers, for example, by selecting specific orientations for each; however, the modularity and simplicity provided by this CSSPA design would be lost.



(a) N = 6.

(b) N = 12.

(c) N = 18.

(d) N = 24.

(e) N = 30.

Figure 31: Number of transducers with visibility across the 2D-projected attitude sphere for each configuration without failures.

4.3 Miniaturized array manufacturing

With the key properties of the new prototype evaluated and defined, the design and manufacturing process was initiated. As discussed in the previous chapter, the pyramid size needed to be considerably reduced to enhance its performance credibility and to fit on the satellite as intended. The latter objective was decided to be postponed to the final version of the prototype, as the focus was given to validating the CSSPA performance with a new type of manufacturing using flexible circuits.

Traditional rigid PCBs present significant limitations in confined volumes that require 3D circuit routing and simultaneous operation under harsh environmental conditions. Flexible circuits have emerged as a vital technology to overcome these limitations. Manufactured from durable polymer substrates, most notably polyimide, also known commercially as Kapton, they support copper interconnections that can bend without substantially compromising electrical performance.

Kapton is one of the most frequently used materials in the space industry due to its lightweight nature, thermal stability, chemical resistance, and versatile mechanical properties, including vibration absorption. It can even be used at very wide temperature intervals and presents low outgassing rates and high radiation tolerance. However, polyimides are usually more expensive than the FR-4 materials used in traditional PCBs. Additionally, flexible circuits are more complex to design and manufacture. Furthermore, in applications where the circuit needs to flex repeatedly, there is a risk of mechanical fatigue, leading to cracked traces and delamination, which are exponentially accentuated with bending.

Two approaches are available with this technology: the rigid-flex PCB and the FPC. The first is a hybrid solution, utilising rigid boards where electronic components are soldered, and leveraging flexible circuits only as a wiring substitute. The second is entirely composed of flexible parts. The FPC is preferable in this application since the CSSPA circuitry becomes significantly simplified, especially during assembly. Rigid-flex PCBs also require validating the thermal expansion compatibility of the different materials, which is disadvantageous.

Similarly to the traditional PCBs, the FPCs is composed of multiple layers arranged in a stackup, each serving a specific purpose. The stackup design is critical to the performance, durability and functionality of the circuit. It typically consists of a base substrate, conductive layers, adhesive layers, protective coverlay, and optional layers such as stiffeners and coatings [71].

The base layer of a FPC, often referred to as the substrate, is the foundation upon which the entire structure is built. Typically composed of polyimide, the substrate provides the necessary flexibility while also serving as an insulating layer [71].

Directly above the substrate is the copper foil layer, the essential conductive pathway that carries electric signals across the circuit. The copper foil can be applied using two main methods: electro-deposited or rolled annealed copper. The first is economical and suitable for static applications where bending is minimal, while the second offers greater flexibility and fatigue resistance [71].

To secure the copper layer to the substrate, an adhesive layer is used. Adhesives such as epoxy are commonly selected based on their bonding strength and flexibility. In some advanced designs, adhesiveless laminates are used, where the copper is directly bonded to the substrate without the need for an additional adhesive layer [71].

The coverlay layer protects the conductive traces from environmental exposure and mechanical damage. Made of polyimide or liquid photoimageable materials, the coverlay is laminated over the copper traces. Its primary function is to insulate the copper and prevent corrosion while allowing necessary access points for soldering through pre-cut openings. The coverlay also contributes to the overall durability of the FPC [71].

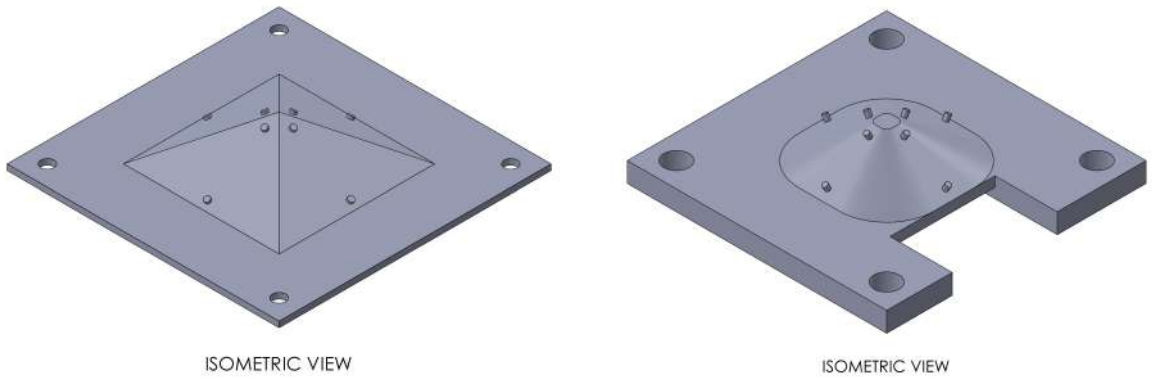
In areas where the circuit requires enhanced structural support, stiffener layers are incorporated to provide additional strength. Made from materials such as FR-4 or additional polyimide, they are most requested at connector interfaces or mounting points. Stiffeners help maintain the circuit's shape and prevent deformation during handling or installation. In some cases, metal stiffeners are employed to reinforce sections that will bear heavy components or experience mechanical stress [71].

To safeguard against environmental hazards, a protective coating can be applied to the FPC surface. Conformal coatings and encapsulants provide moisture resistance, chemical protection, and insulation. These coatings are especially useful in harsh environments, where the FPC may be exposed to dust, chemicals, or fluctuating temperatures [71].

Finally, the exposed copper areas, especially the solder pads, are often coated with a surface finish to enhance solderability and prevent oxidation. Common finishes include Electroless Nickel Immersion Gold (ENIG), which provides a smooth, flat surface ideal for soldering and wire bonding, or Hot Air Solder Leveling (HASL), which is more economical but less suitable for fine-pitch components [71].

The advanced prototype assembly consisted of three main parts: a FPC, its structural support and an interface PCB to aid the connection between the FPC and the MCU.

The process began by designing the pyramidal support with its faces at a 30-degree inclination. The new pyramid support was based on the previous prototype, but some relevant modifications were made. The most significant change was in its size, which was reduced by approximately a factor of 2.5 in terms



(a) First iteration. (b) Last iteration.

Figure 32: Advanced prototype support iterations.

of the pyramid base edge, ultimately reaching approximately 27 mm. Obviously, its height was also updated to approximately 8 mm, regarding this reduction, as well as the updated inclination of the faces. The extrusions in each face were removed because of the desired placement of the FPC on top of the pyramid. Marker poles substituted the holes to ensure the proper orientation of the transducers while fixing the circuit in place. This modification was made to assess the removal of the screws, which would occupy a considerable volume in the following prototype. The base plate size was also reduced by half to approximately 52 mm, with 5 mm holes only in its four corners, matching those present in the testing setup table, as will be shown in the following section. This support, as shown in Figure 32a, was iterated upon until it reached its final design, as seen in Figure 32b, where three key features are clearly noticeable. The first and most important is the filleting of all lateral edges, almost blending the pyramid into a cone, to prevent the FPC bending and, therefore, the damage of conductive traces. Note that the cone is not ideal, and there are some straight zones where the transducers can be placed, delimited by the dashed lines, thereby maintaining a pyramid shape. Next, the pyramid's top vertex was truncated by 1 mm to reduce the total height even further while also aiding the flexing of the FPC around the pyramid. Finally, the base plate was elevated to 4 mm, and a 2 mm extrusion on the bottom was added so that the interface PCB could be placed underneath.

After this design was completed, the unfolding surface geometry of the pyramid was sketched, ensuring that the FPC would fit exactly on the pyramid when laid coplanarly against each support face. In other words, the lateral surface area of the pyramid was deconstructed from the 3D model into a 2D sheet in order to manufacture the FPC effectively. The result is pictured in Figure 35a. Notice three relevant considerations: there is an additional incomplete face that is intended to overlap the first, specifically in

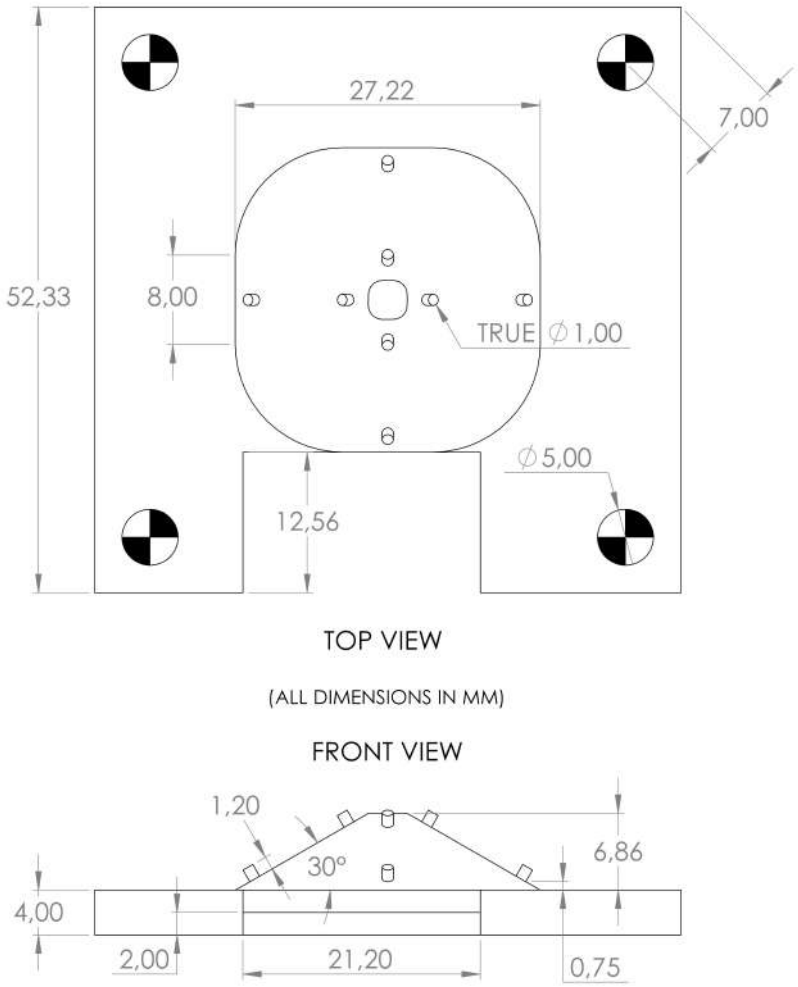
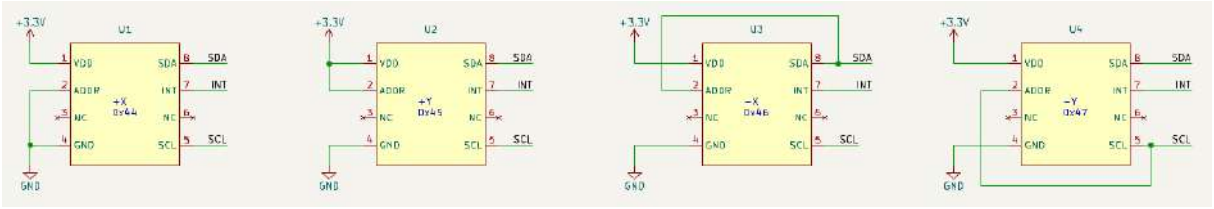


Figure 33: Advanced prototype support dimensions.

the poles zone, in order to ensure a uniform tension distribution across the pyramid; a small tail was added, in which the interface PCB will electrically connect to the FPC; and some fillets were introduced to prevent the accidental bending of corners.

The FPC circuit schematic was even simpler than the previous prototype. It was composed of the four transducers connected to the designated power and I2C lines, with the respective addresses hard-wired, as shown in Figure 34a. The interrupt lines of all transducers were connected together, allowing for simultaneous triggering of measurements if desired. To reduce possible interference in the T-FOVs, the remaining passive components were placed on the interface board, which also features a specific connector for the FPC and a mounting hole to secure it in place, as depicted in Figure 34b.

It is good practice to start the FPC design process by selecting the connector and then adapting the circuit to match its specifications. MOLEX is one of the most well-known sellers in this specific field, and a quick search narrowed down the MOLEX 52746-0671, a slider-type connector with six gold-plated contacts



(a) Flexible printed circuit.

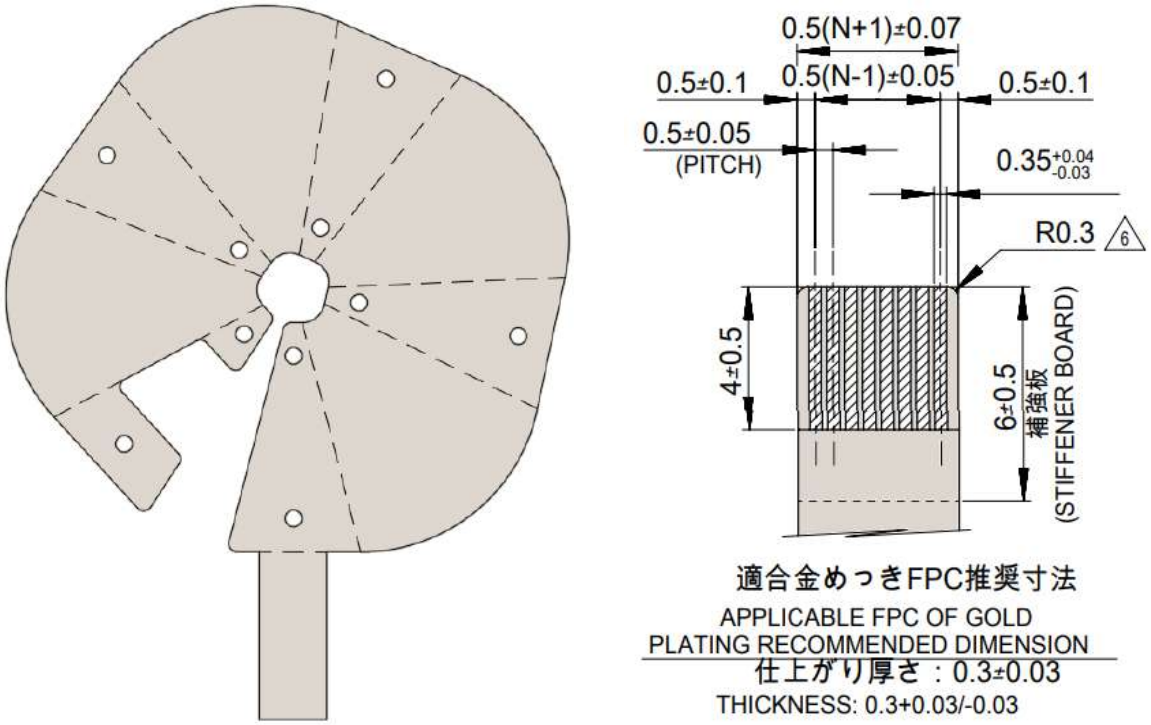


(b) Interface printed circuit board.

Figure 34: Advanced prototype circuits' schematics.

and a reduced height of 2 mm. The small form factor was desirable since it would be easier to fit in the satellite, if intended. Its datasheet presents very detailed information regarding the recommended dimensions of the FPC connector zone, which is shared on Figure 35b. A thickness of 0.3±0.03 mm was therefore imperative on that zone, in pair with a 4 mm exposure length of the conductive fingers and a 6 mm stiffener length.

A standard 2-layer FPC, sufficient for this application, is composed of a symmetric stackup of a single core, followed by the copper layers, thermosetting adhesive, and the coverlay, as described earlier; however, their thickness may vary. Fortunately, manufacturers also provide their recommendations on this dimension, where the copper is usually 12 µm, the adhesive is 15 µm, and the core is 25 µm. To verify whether the FPC recommended layer thickness was compatible with the selected connector thickness, a simple stackup table was formulated and presented in Figure 36. This table is typically requested by the manufacturer so that it can accurately follow the design choices. Adding a standard 0.2 mm stiffener and the required adhesive layer below was sufficient to ensure the thickness compatibility of the connector, totalling 0.29 mm in the exposed pad zone. Overall, the remaining FPC thickness is around 0.11 mm, which is approximately 15 times thinner than a regular PCB. The thickness of the copper and core layers can be set in KiCad, ensuring this information is embedded in the exported files. However, the coverlay and stiffener locations must be added as customary layers on the design since the design program does not provide a specific type for them. Fortunately, the manufacturer interprets them easily.



(a) Unfolded lateral surface of the pyramid.

(b) Edge connector zone (retrieved from [72]).

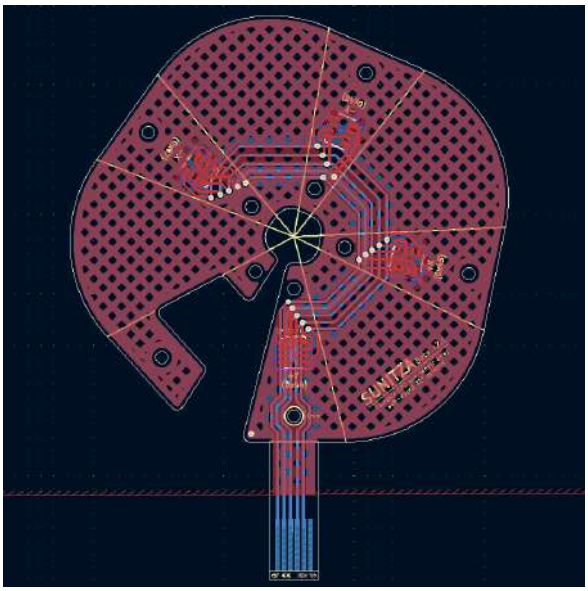
Figure 35: 2D sketch of the FPC

ALL FPC	EDGE CONNECTOR									
	<---------- 6 mm ---------->									
		<---------- 4 mm ---------->								
	STIFFENER (POLYIMIDE)			0.200	mm	-	-	0.200	mm
	THERMOSETTING ADHESIVE			0.015	mm	-	-	0.015	mm
COVERLAY (POLYIMIDE)				0.013	mm	0.013	mm	0.013	mm
THERMOSETTING ADHESIVE				0.015	mm	0.015	mm	0.015	mm
COPPER FOIL				0.012	mm	0.012	mm	0.012	mm
CORE (POLYIMIDE)				0.025	mm	0.025	mm	0.025	mm
COPPER FOIL				0.012	mm	0.012	mm	0.012	mm
THERMOSETTING ADHESIVE				-	-	0.015	mm	0.015	mm
COVERLAY (POLYIMIDE)				-	-	0.013	mm	0.013	mm
			TOTAL	**0.292**	**mm**	**0.104**	**mm**	**0.319**	**mm**

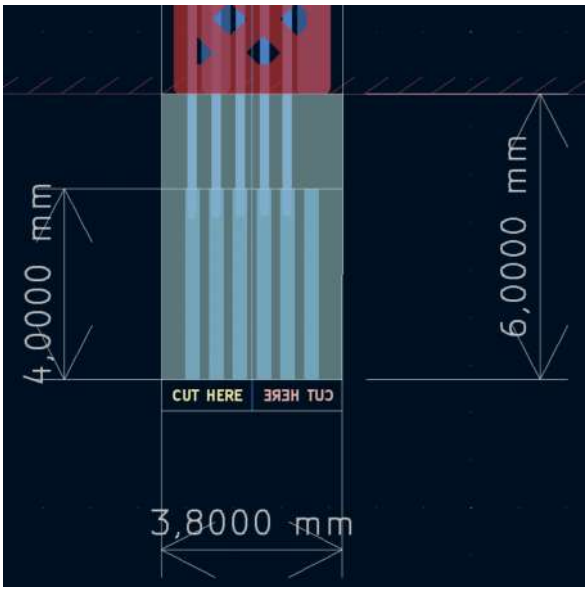
Figure 36: Designed layer stackup of the FPC.

The 2D sketch of the FPC of Figure 35a was imported into KiCad and set as the edge cuts layer. Then, the OPT4001 footprints were placed on the top copper layer, arranged in the centre of each designated side, while the power and data buses were routed via the bottom, with 0.2 mm traces, as seen in Figure 37a. The remaining free space was filled with the usual copper plane, in this case especially hatched to improve flexibility. On the edge connector zone, the connection fingers were drawn on the bottom copper and solder mask layers, as dictated by the connector dimensions of Figure 35b. The bottom coverlay was not added there, and the stiffener was added at the top, as shown in Figure 37b. Note that a post-manufacturing cut zone is implemented on the tip to overcome the manufacturing constraint of the clearance required between the FPC edge and the laser cut while simultaneously respecting the dimensions needed for the connector. The 3D rendering of the circuit is shown in Figures 37c and 37d. After exporting the Gerber files, they were sent to the PCB manufacturer, together with the stackup table.

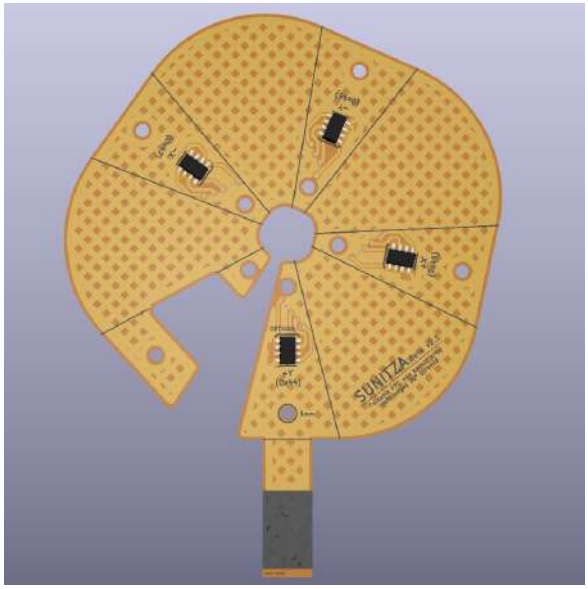
The FPC copper was electro-deposited, and its surface was finished with ENIG. The gold finger plating was approximately 25 nm thick. An EMI shielding layer with 18 µm was not opted for, but it is something to be considered in the final version. These were additional options selected when the order was placed. Furthermore, the manufacturer, in this case JLCPCB, was also requested to fulfil the lacking information regarding the specific materials of the FPC. Its composition was reported to be as follows: the polyimide core and adhesiveless, flexible copper double-clad laminate are from the LONGLITE Flex 300 series; the coverlay and epoxy adhesive are from the R/Flex A series; and the polyimide stiffener is from the R/Flex AN00 series, all fabricated by the Rogers Corporation. The specific polyimide brand employed is not DuPont's Kapton, but Kaneka's APICAL NP film for the coverlay and their PIXEO film for the core. All have similar performances, with APICAL presenting better mechanical properties [73, 74].



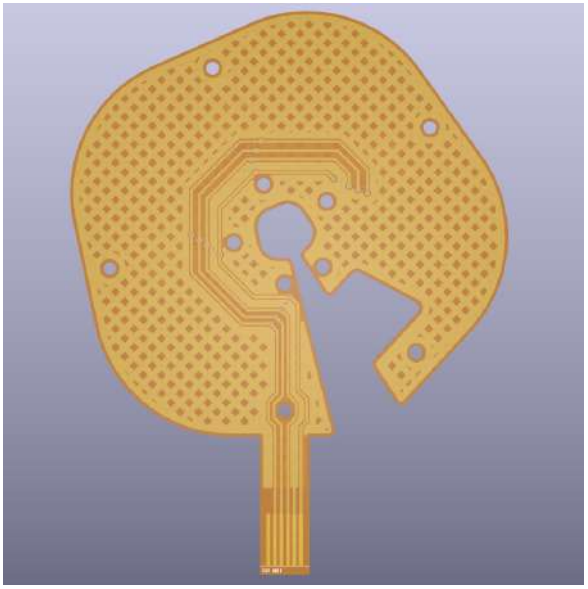
(a) FPC layout.



(b) Edge connector zone.



(c) 3D rendering (front side).

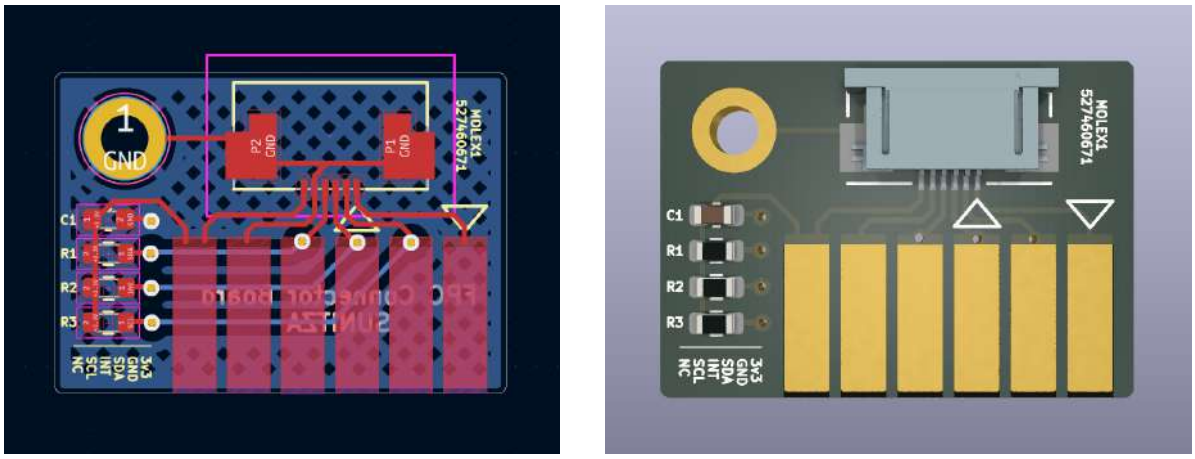


(d) 3D rendering (back side).

Figure 37: Advanced prototype flexible printed circuit.

Regarding the design of the interface PCB, it is also a traditional 2-layered board, where the traces originating from the FPC connector are routed to larger pads for horizontal connections, as shown in Figure 38a. This choice was made instead of the previous vias type to allow a reduced profile. The 3D rendering is depicted in Figure 38b.

Once manufactured, the CSSPA was assembled following similar steps as before. The transducers were soldered using a reflow oven and carefully inspected using a microscope, as pictured in Figure 39a. Then, the FPC was laid onto the 3D-printed support, sequentially inserting its holes into each face's marker



(a) PCB layout.

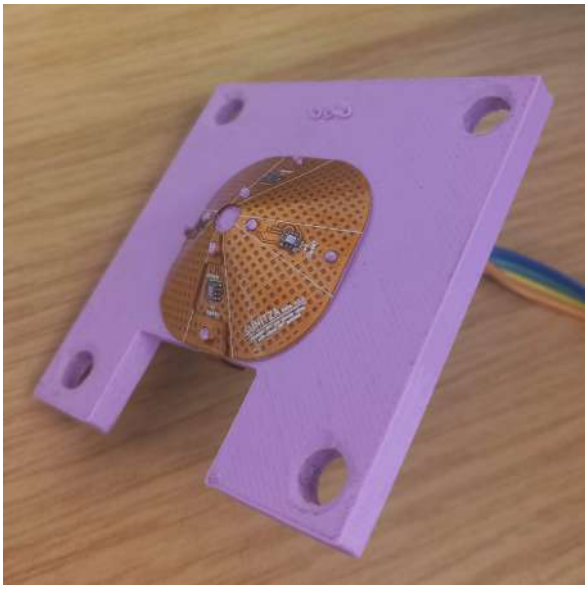
(b) 3D rendering.

Figure 38: Advanced prototype interface board.

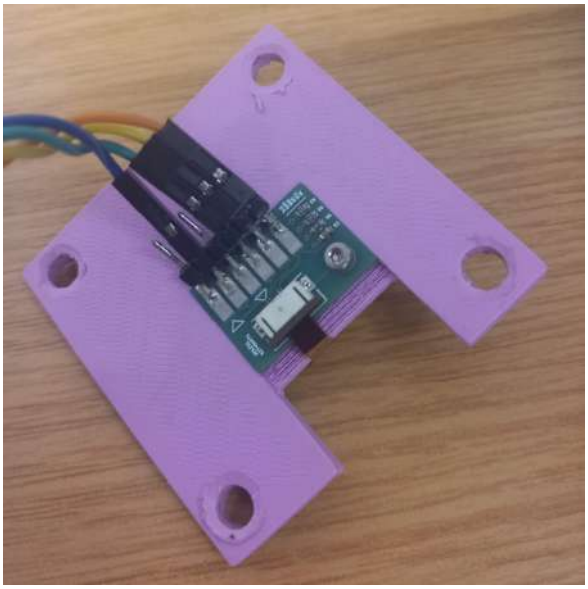
poles, resulting in Figure 39b. No adhesive was added between the two parts, allowing the circuit to be removed later for other assemblies. This is another aspect to consider in the following prototype. Then, the FPC fingers were inserted in the respective connector present at the interface PCB, which in turn was secured at the bottom of the support with an M2 screw, as shown in Figure 39c. Wires were also soldered to establish a connection with the MCU board, the same one used in the previous prototype. Its firmware and software remain the same as before, as its purpose has not changed.



(a) Microscope inspection of the FPC.



(b) Finished prototype (front side).

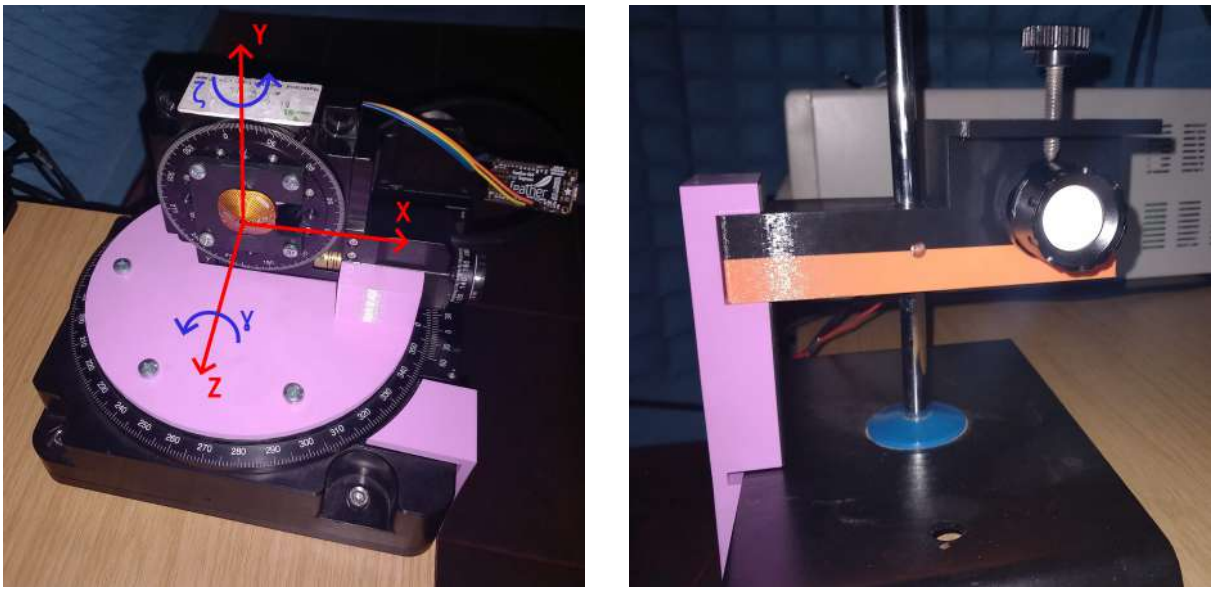


(c) Finished prototype (back side).

Figure 39: Advanced prototype assembly.

4.4 Sun vector determination methodology

As recommended in the previous chapter, the experiments should conduct a two-axis evaluation of the CSSPA performance, with an automated setup and stable fixture of components. To do so, the testing setup was upgraded, and two high-precision rotation stages replaced the 3D-printed spinning disc table. Both were made of aluminium and hardened steel bodies, with stepper motors featuring negligible backlash. They could travel continuously with a 0.01-degree resolution. The main difference between models is



(a) Rotation tables with axes and angles definition.

(b) Flashlight support.

Figure 40: Advanced prototype test setup parts.

their size. The Standa 8MR190-90-59 is 20x20x5 cm, while the Standa 8MR190-2-28 is 10x10x3 cm, therefore the first was nicknamed as “big”, while the second as “small”. Each table required a dedicated controller and specific Windows drivers to be installed in the controlling computer. Additionally, some low-level programming drivers were required to communicate with the controllers via the serial port, including functions to rotate to the desired degrees and retrieve the disc position. Fortunately, these were provided in Python by the laboratory team that borrowed the equipment.

This time, the pyramid base was placed directly on the rotation disc of the small table and secured in place with four M4 screws, eliminating the need for the wooden contraption of the previous setup. A 3D-printed adapter was fixed on the rotation disc of the big table with five M4 screws, allowing the small one to be placed sideways on top, as shown in Figure 40a, and thus enabling the two axis rotation of the CSSPA. A few parts were also modelled to fix the lower stage in place.

Unfortunately, the fixed light source employed was the same as before. However, some modifications were made to secure and adjust the flashlight. Its support relied on two separate parts held in place by two M5 screws, which fixed the flashlight handle tightly. A third, located at the edge, allowed for small adjustments. This assembly was slid through a vertical pole to adjust its height relative to the CSSPA, as seen in Figure 40b. A separate 3D-printed part was added to ensure the proper alignment between the light vector and the pyramid boresight.

A general rule of thumb for approximating a light source to a point source was found to be distancing

the target at least five times farther than the largest dimension of the source. As stated by Ryer [75], this dimension may be either the length of the filament or the diameter of the bulb, depending on the light collimation. In this scenario, the latter was chosen as the driving dimension, thereby selecting a somewhat worst-case scenario. After this distance, the error from assuming an inverse square behaviour is below 1%, whereas closer to the source, its geometry and emission matter and the law does not hold accurately. This knowledge was helpful in logically evaluating whether the distance between the light source and the sensor was sufficient to simulate the Sun's behaviour in orbit as closely as possible within the available time and cost budgets. The diameter of the flashlight was measured to be 33 mm, so a distance of at least 16.5 cm should be granted. Regarding the available test bench for this setup, a maximum distance of 82 cm between the light and sensor was possible and selected.

After conducting more thorough research on the flashlight composition, especially its Light-Emitting Diode (LED), it was estimated to be the XP-G4-B3 reference from Cree, due to the same advertised luminous flux and similar size. This information allowed us to discover the LED nominal electrical characteristics at which the luminous flux and intensity are specified. The nominal current of this LED is 700 mA, and its typical forward voltage is 2.8 V, which translates to a power consumption of around 2 W. Still, some luminous flux variation was expected, as also reminded in the datasheet, due to the temperature increase by the Joule effect.

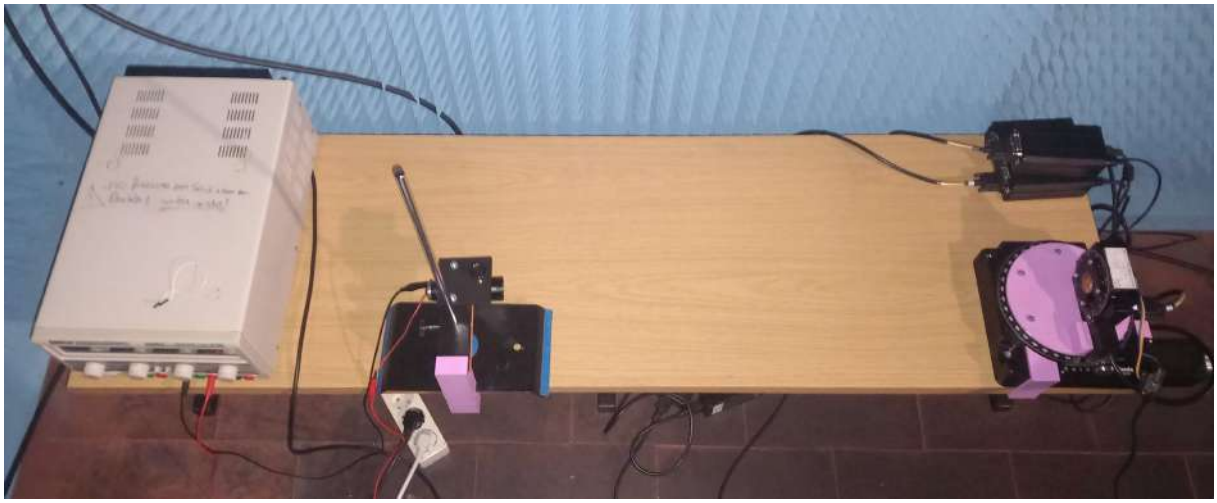
Another improvement made in the testing setup, addressing a concern from the previous chapter, was to minimise the translation when rotating the CSSPA relative to the light source. The reduction of the pyramid size was one of the solutions to this issue, as explained in the previous section; however, another approach could have been employed in this phase. Note that with the current setup, the only point in space that does not translate is the intersection of the two rotation axes. Therefore, it is impossible to prevent the translation of all transducers simultaneously. However, by placing this interception point in the middle of all transducers, the translation is minimal and equally distributed.

The setup was modelled in SolidWorks, as pictured in Figure 41a, and after defining the rotation axes, all the 3D parts were adjusted and printed so that the alignment of the light source and the rotation axes are respected, as shown in Figure 42, respectively. The setup was mounted in a dark room, this time inside an anechoic chamber, to mitigate external light sources and internal light reflections, as also recommended in the previous chapter. The final result is pictured in Figure 41b.

During the previous testing campaign, only one CSSPA was rotated. This time, to fully evaluate the



(a) High-fidelity software model.

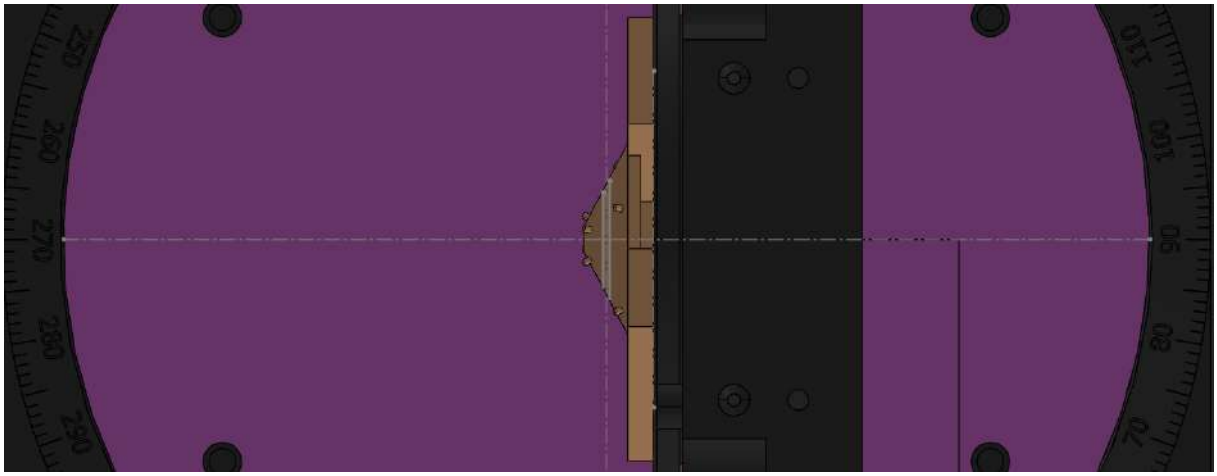


(b) Anechoic chamber implementation.

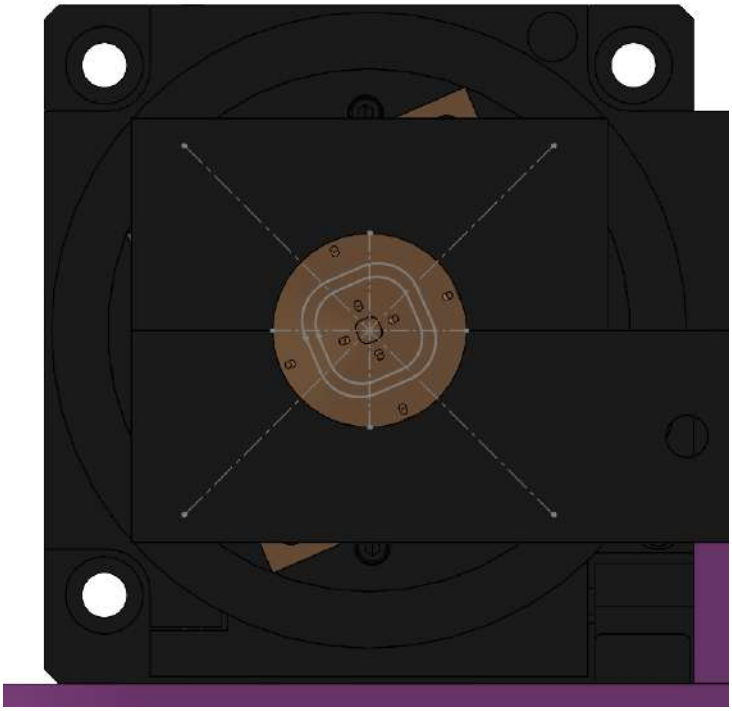
Figure 41: Advanced prototype test setup.

system performance, four trials were performed in total, where the testing procedure was repeated individually for the three advanced prototypes assembled, and then a final one with all three CSSPAs combined. This way, a scenario in which numerous transducers were available could be tested, similar to what it is expected in orbit. Note that three CSSPAs were enough due to the symmetry of the determination problem.

To perform the combined trial, some minor modifications were needed to both the pyramid support and the test setup. Considering the satellite form, the three pyramids would form half the cube's faces. However, with the current design, this cube would be relatively large, at least with the 53 mm edge of the pyramid plate. To reduce its size, a separate four-part assembly was designed, composed of three optimised pyramid supports and a new plate. The optimised support shares the same pyramid structure but removes the holes and reduces the plate size to the necessary, approximately 34 mm. Additionally, puzzle-like connections were added to each, enabling the three supports to be easily assembled without the need for screws. The fourth part, the new base, provided a coplanar face with the rotating table, some screw holes for fixing as before, and a circular gap for cable management.

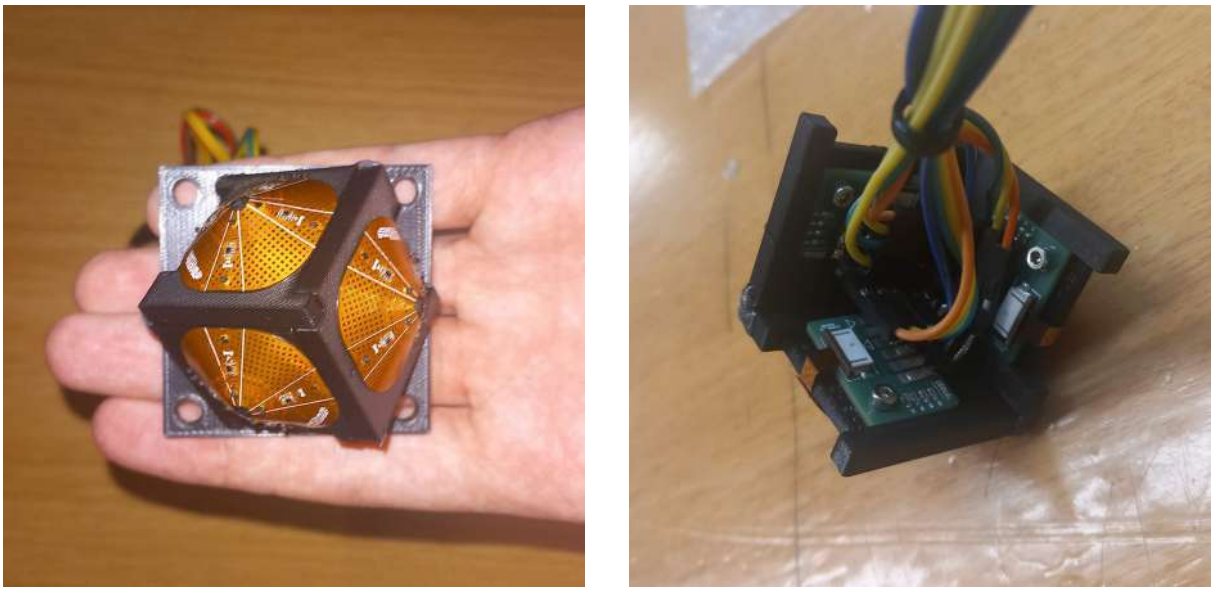


(a) Top view of big table, with its rotation axis (Y) intersecting transducers centroid.



(b) Front view of light source support, with its certer aligned with small table rotation axis (Z) and pyramid boresight.

Figure 42: Sensor alignment with the respective axes.



(a) Updated prototype (front side).

(b) Updated prototype (back side).

Figure 43: Advanced prototype combined assembly.

Figure 43 shows the assembly result. Note that this separation into different pieces improves both printing quality and assembly ease. Regarding the test setup, since the support dimensions changed and the rotation axes alignment was lost. Therefore, other 3D parts, such as the stages adapter, required modifications to ensure the best possible alignment in this scenario.

The testing procedure was similar during all trials. It began resetting both tables to their starting position via code, and visually confirming it with the engraved dents on each disc, followed by supplying the flashlight with its nominal current. The power supply was adjusted to operate in its Constant Current (CC) mode, setting a maximum current limit at 700 mA and increasing the voltage limit to the maximum possible, ensuring this operation during all testing. With two tables, the easiest rotation path follows a conical manoeuvre, close to the one proposed by Mafi et al. [36]. In this setup, the small table rotated 360 degrees in 2-degree increments, and then the big table took 2-degree increments. This step was repeated until the big table had rotated 90 degrees in total. Since it is a conical manoeuvre, in reality the 180x180 degrees region above the CSSPA is being collected, or in other words, the attitude sphere hemisphere of 2π steradians. The increment was increased compared to the previous chapter to balance the number of data points and the time needed to complete each trial.

The rotation tables allowed for a quasi-automated collection of measurements. The recording algorithm was designed to enable the computer to play a central role in the communication between devices. It started with the computer sending the rotation command to the stages. Once completed, it sent five

separate sampling commands to the MCU via the serial port. Then, the computer averaged the results of each transducer measurement and logged them, along with the timestamp and the rotation angles of the tables, into a CSV file, as done previously. The measurements are not fully automated due to the fact that this setup does not have a slip ring for the wiring, so they need to be disconnected and untwisted several times per trial.

As explained in the previous chapter, assessing the accuracy of the Sun vector determination involved calculating the angular error between the Sun vector estimated from the measurements and the expected vector for each point in the collected dataset. However, this time, the estimation and calibration procedures were conducted in accordance with the literature review.

The expected Sun vector was obtained by applying the rotation matrices, which map the Sun vector from its initial direction at the pyramid boresight to its current direction, depending on the rotation angles of the tables. With this high-precision setup, no translation was assumed, and the two sequential rotations were sufficient to calculate the direction of the light source. Note that the rotation angles of the tables described the CSSPA movement, so to describe the light rotation, they needed to be mirrored. The incidence angle was obtained by the dot product between the calculated light direction and each normal vector, which, by this problem formulation, remained inertial.

The measured Sun vector was estimated using the LS method; however, this vector was calibrated several times, following the framework provided by Mafi et al. [36] and explained in Section 2.2.2. In short, the first step was to estimate the measured Sun vector without calibration, serving as a baseline. Then, as explained in the next section, the flashlight intensity decay was calibrated. After that, three calibration steps were performed: correcting the misalignments of the normal vectors; addressing the non-linear characteristics of the curve; and adjusting the transmittance coefficient through an iterative process [36]. After each calibration step was completed for all transducers, the Sun vector was re-estimated to assess how performance evolved over time. This process followed a pipeline structure so that each downstream step calibrated data that the upstream steps had already processed. Algorithm 7 summarises the calibration procedure applied to each transducer after the dataset was collected and the light vector determined for each direction of the attitude sphere. Next, Algorithm 8 details the method for estimating the Sun vector using calibrated transducers once new measurements are provided. Note that the iterative transmittance calibration is only performed during this phase.

Algorithm 7 Photoelectric transducers calibration (adapted from [36]).

$I_t, \check{S}$ ▷ collect measurements dataset and calculate Sun vector ground truth
$I_t \leftarrow I_t/d$ ▷ calculate linear decay factor of intensity and update measurements
if $\check{I}_t > 0.87$ **then** ▷ riddle dataset unaffected by Fresnel effects for each transducer
    $I_t^* \leftarrow I_t$
end if
$\hat{n}_t \leftarrow (\check{S}^T\check{S})^{-1}\check{S}^T\check{I}_t^*$ ▷ estimate normal vectors for each transducer with riddled dataset
$C_t \leftarrow \left[\begin{array}{ccc} \check{S} \cdot \hat{n}_t & ; & 1 \end{array} \right]$ ▷ calculate cosines matrix for each transducer
$\left[\begin{array}{ccc} \hat{\alpha}_t & ; & \hat{\kappa}_t \end{array} \right] \leftarrow (C_t^T C_t)^{-1} C_t^T I_t$ ▷ estimate cosine curve characteristics for each transducer
return $\hat{n}_t, \hat{\alpha}_t, \hat{\kappa}_t$

Algorithm 8 Sun vector estimation with an iterative least squares (retrieved from [36]).

$i_t, \hat{n}_t, \hat{\alpha}_t, \hat{\kappa}_t$ ▷ get new measurement and calibration coefficients of each transducer
$\check{i}_t \leftarrow (i_t - \kappa_t)/\alpha_t$ ▷ calibrate new measurements with cosine characteristics coefficients
$\hat{s} \leftarrow (\hat{N}^T\hat{N})^{-1}\hat{N}^T\check{I}$ ▷ estimate Sun vector with calibrated measurement and normal vector
while Sun vector has not converged **do** ▷ verify with either by a threshold or number of iterations
    $\tau_t \leftarrow T(\theta_t)/T(0)$ ▷ calculate transmittance coefficient for each transducer
    $\check{i}_t \leftarrow \check{i}_t/\tau_t$ ▷ calibrate measurements of each transducer
    $\hat{s} \leftarrow (\hat{N}^T\hat{N})^{-1}\hat{N}^T\check{I}$ ▷ estimate Sun vector with new calibrated measurements
end while
return $\hat{s}$

Furthermore, after assessing the determination accuracy and the illuminance response, the sensor coverage was also evaluated. This methodology was conducted during the individual trials of each CSSPA. Once the three sensors were calibrated, the discovered coefficients were saved and introduced into the combined trial procedure, during which only the measured Sun vector estimation was conducted and its accuracy and coverage assessed. These are the two case studies analysed in the next section.

4.5 Sun vector determination results analysis

After collecting the dataset for each trial, which consisted of measurements and rotation angles, the performance was evaluated in terms of the angular error across the A-FOV. Each dataset consisted of 8,280 points across the CSSPA, collected for approximately five hours each.

Motivated by the concerns regarding the luminous flux dependence on temperature, as mentioned earlier, a static collection of measurements was performed before starting each trial. Figure 44 shows the

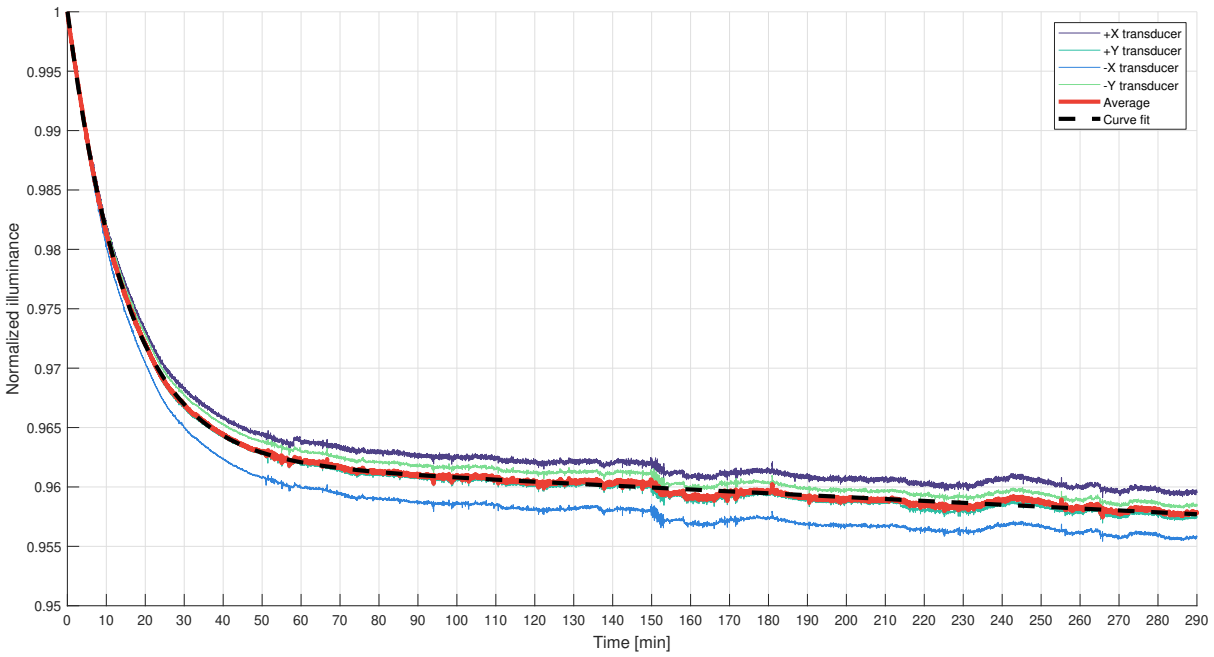


Figure 44: Illuminance evolution over time during static sampling.

normalised illuminance measured over time for each transducer, in pair with their average and a fitted curve. The results are close to expected. As the current flowed through the LED, its junction temperature increased, and the luminous efficacy of the LED dropped. Over time, the temperature approached a thermal equilibrium exponentially, and inversely, the illuminance decayed exponentially. During the first hour, the illuminance decayed by approximately 3.8%, and over the next four hours, it decreased by around 0.5% on average, following an almost linear trend. The average curve was successfully fitted to a two-term exponential, with a RMSE of 0.0002. The calculated coefficients defined the funtion $\check{E}(t) = 0.03775\exp(-1.115 \times 10^{-3}t) + 0.9624\exp(-2.789 \times 10^{-7}t)$, where $\check{E}$ is the normalized illuminance and t is the elapsed time in seconds.

This initial fast decay was problematic because it would have introduced another source of non-linear error in the dataset. To mitigate this, the static collection of measurements was performed before every trial for approximately two hours. This way, enough time was allowed for the flashlight to reach a linear behaviour and to collect enough measurements to predict its evolution over the following hours during the dataset collection.

Upon closer examination of the plot, two patterns are immediately apparent. The first pattern is that each transducer pair, namely the ±X, which is more noticeable than the ±Y pair, exhibited almost equally opposing deviations from the average, although very short, around 0.2%. This behaviour could indicate some slight misalignment regarding the test setup parts. The second pattern is that all measurement

curves shared the same spikes and valleys, predominantly after the 50th minute mark, meaning that there was some additional noise affecting the light source, most likely in the power supply.

4.5.1 Single pyramid case study

Before delving into the accuracy evaluation, some calculations were performed to verify whether the measured illuminance was within expectation. Since the light source was assumed to be a point source, as explained in the previous section, the only variable affecting the measurement was the distance between the transducer and the source. Knowing that the flashlight produces a 4,445 cd beam, this converts to an illuminance of 4,445 lm/m^2 at 1 m. Therefore, at 82 cm, the transducers should have measured 6,611 lx, following the inverse square law, which is approximately 5% of the expected value of AM0. From the collected datasets, the maximum values measured were averaged to 6,340 lx, translating to a 4% decrease compared to the theoretical result. This difference was an acceptable margin due to the initial static period, which reduced the overall flashlight intensity, as well as other sources of discrepancy in the setup.

Starting with the first assembly of the advanced prototype, the obtained raw measurements are presented in Figure 45, where the illuminance is shown as a colourmap in a polar plot, where the two rotation angles, gamma, γ, and zeta, ζ, are its angular and radial components, respectively, similarly to azimuth and co-elevation, defined in Figure 40a. This plot can also be interpreted as the 2D-projection of the attitude sphere above the CSSPA onto a plane whose normal vector is collinear with the pyramid's boresight and is defined at the origin. As expected, the maximum values are obtained at angles close to 30 degrees of zeta, with 90-degree variations of gamma for each transducer. In other words, the sensors are roughly tilted by 30 degrees and rotated 90 degrees apart. At first sight, the measurements also evolve smoothly, from 5 lx, which can be perceived as only noise, up to 6404 lx, the maximum value of the -Y transducer.

However, the illuminance can also be presented in terms of incidence angle, which its dependence is quite more graspable. To perform this transformation, since the dataset had three dimensions, 3D planes needed to be defined to cluster a variable, so a 2D plot could be constructed afterwards. The clusters were defined to group common gamma values, each one separated by 45 degrees. Although considerable information was lost, as only eight clusters were formed, it helped to visualise the evolution across the interest region. This specific step can be understood by observing Figure 45 and grouping the measurements intersected by each radial line, starting from the origin of the circumference to each defined cluster. After this, the independent variable of each became zeta. Nevertheless, the incidence angle, θ,

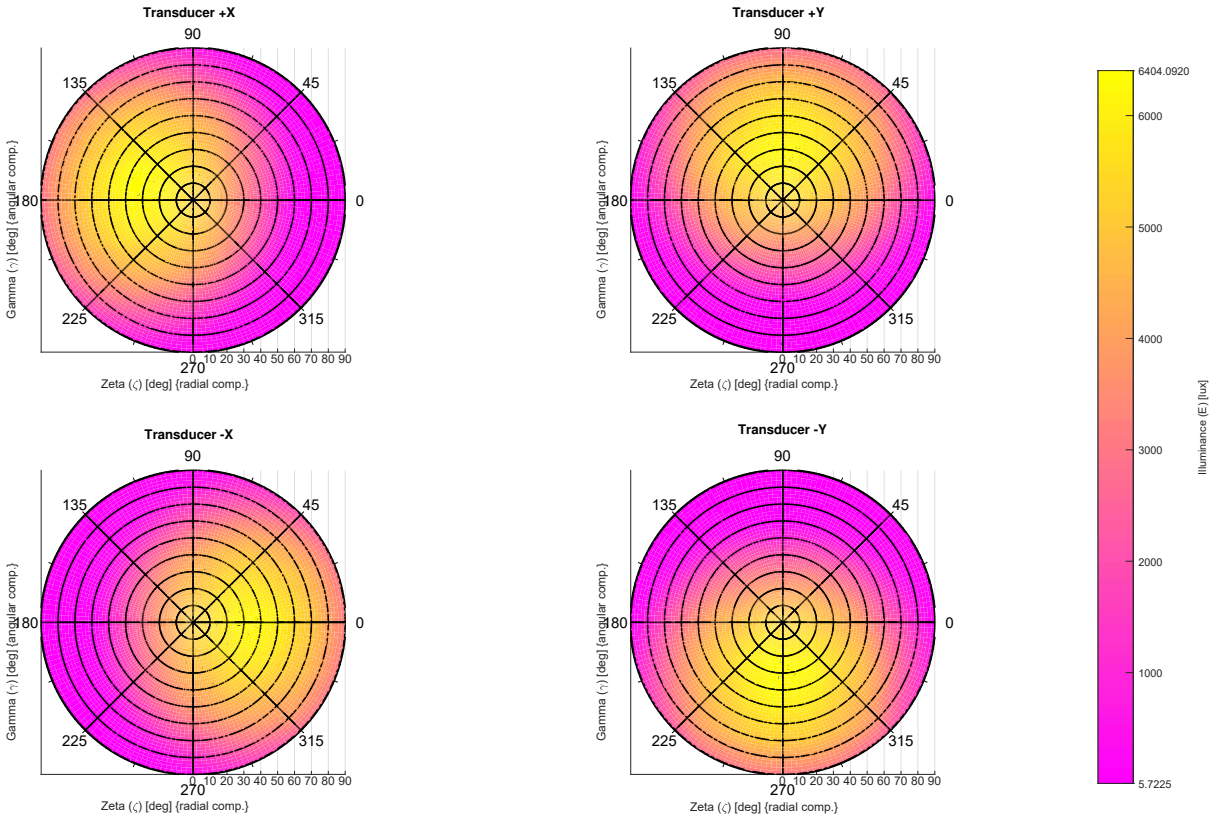


Figure 45: Illuminance measured by the four transducers of the advanced prototype (first sensor).

was calculated using the two rotation angles and the transducer normals and thus rearranged the data. Note that the opposite procedure, selecting the zeta as the cluster variable and gamma as the independent variable, could also be done.

Figure 46 shows the evolution for each transducer, compared to the expected value obtained by the cosine law. The results were truncated to the A-FOV region where all transducers intersect, as conducted in the original calibration framework. The subplots display the relative error of the measurements from each cluster and the evolution of the incidence angle as zeta is varied. Overall, the illuminance followed the tendency of the cosine law. However, discrepancies arose when the incidence angle increased, and consequently, the Signal-to-Noise Ratio (SNR) decreased. The +X transducer had the worst performance of the four, with the relative error reaching 30% around $\theta = 60^{\circ}$, and even larger once the illuminance was very low. The remaining transducers behaved similarly with less expressive error, usually half as previously stated. The ripple observed in the three clusters with smaller incidence angles across all transducers, while $\theta < 30^{\circ}$, is explained by the fact that during this interval, there were two data points for approximately the same incidence angle, as seen in the bottom-left subplot. This effect occurred because of some setup misalignment that prevented it from having the same result.

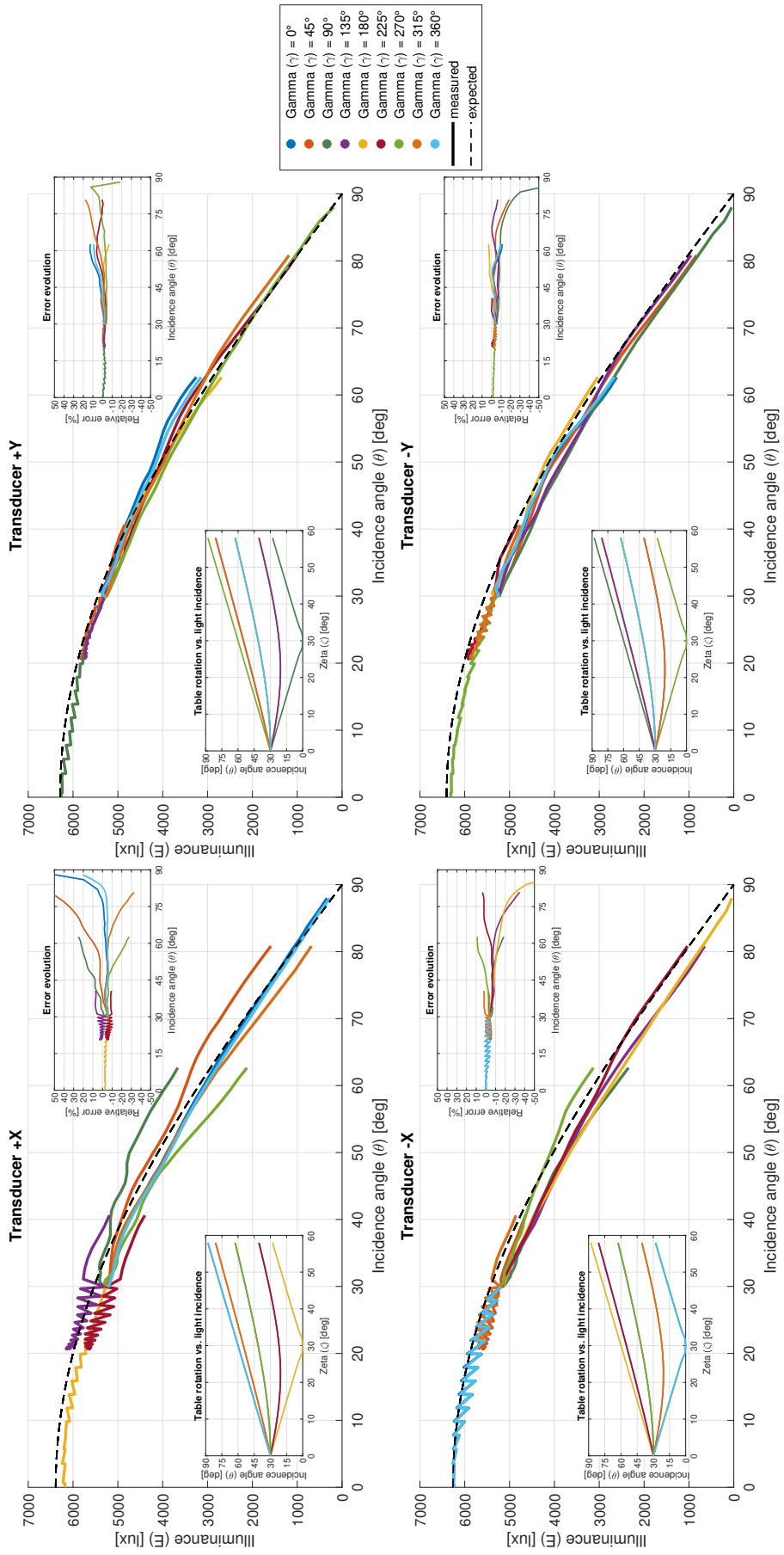


Figure 46: Uncalibrated illuminance dependence on the incidence angle (first sensor).

After performing the two-hour wait period during the static collection of measurements, the intensity variation over time was insignificant and remained fairly constant, since the plots of the uncalibrated measurements and the measurements with the calibrated linear decaying of the intensity were identical, as perceived in Figure 47.

Next, the normal vector calibration results are shown in Figure 48. As can be seen, this step significantly reduced the illuminance error, approximating each estimated transducer boresight to the actual one. Furthermore, the existing discrepancies were more bounded than before, with most data points having an error of less than 10%. Note that with this calibration, the selected clusters did not have a true zero incidence angle, clearly observed in the +X transducer, which had the first point at around $\theta = 7^{\circ}$. This result indicates that the transducer was significantly misaligned on its pyramid face; however, this discrepancy could be compensated during calibration. The primary possible causes were the detachment of the FPC from the pyramid support or incorrect soldering of the transducer. The most visible error during this calibration step was present on the ±X pair at higher incidence angles, which could also mean a setup misalignment over that axis in either the big table or the flashlight support, which was substantially more susceptible than the Y axis, since it was more difficult to align. The normal vectors of the new transducers were compared with those of the previous ones in Figure 49, along with the angle between them. As expected, there were deviations between the transducers' normals and the pyramid faces' normals, most noticeably regarding the +X transducer, which presented a considerable change of 7.3 degrees, followed by the -X, with 4.7 degrees. This calibration step also revealed a degree of asymmetry, particularly observed in the bottom-left subplots, where the complementary pairs of curves are now proportionally separated.

The transducers' characteristics calibration step further improved the results, as seen in Figure 50, especially in the low incidence region. However, once the incidence angle reached zero, the error grew exponentially due to the asymptote on the relative error function. Excluding this behaviour, the 10% error was maintained across the entire dataset, with most regions falling below the 5% mark. However, once the transmittance calibration was applied, this specific region was affected, and the error rose again, reversing the calibration improvements, as shown in Figure 51. Fundamentally, this step is characterised by iteratively converging the measured illuminance to a new value, with 50 iterations, successively updated by the incidence angle and the transmittance coefficient, because both terms are cross-dependent. However, this coefficient relies on the absolute refractive indices of the incident and transmitted mediums. The first is the air, from which the flashlight photons are travelling, and the second is assumed to be the clear epoxy-based package that protects the transducer sensing element.

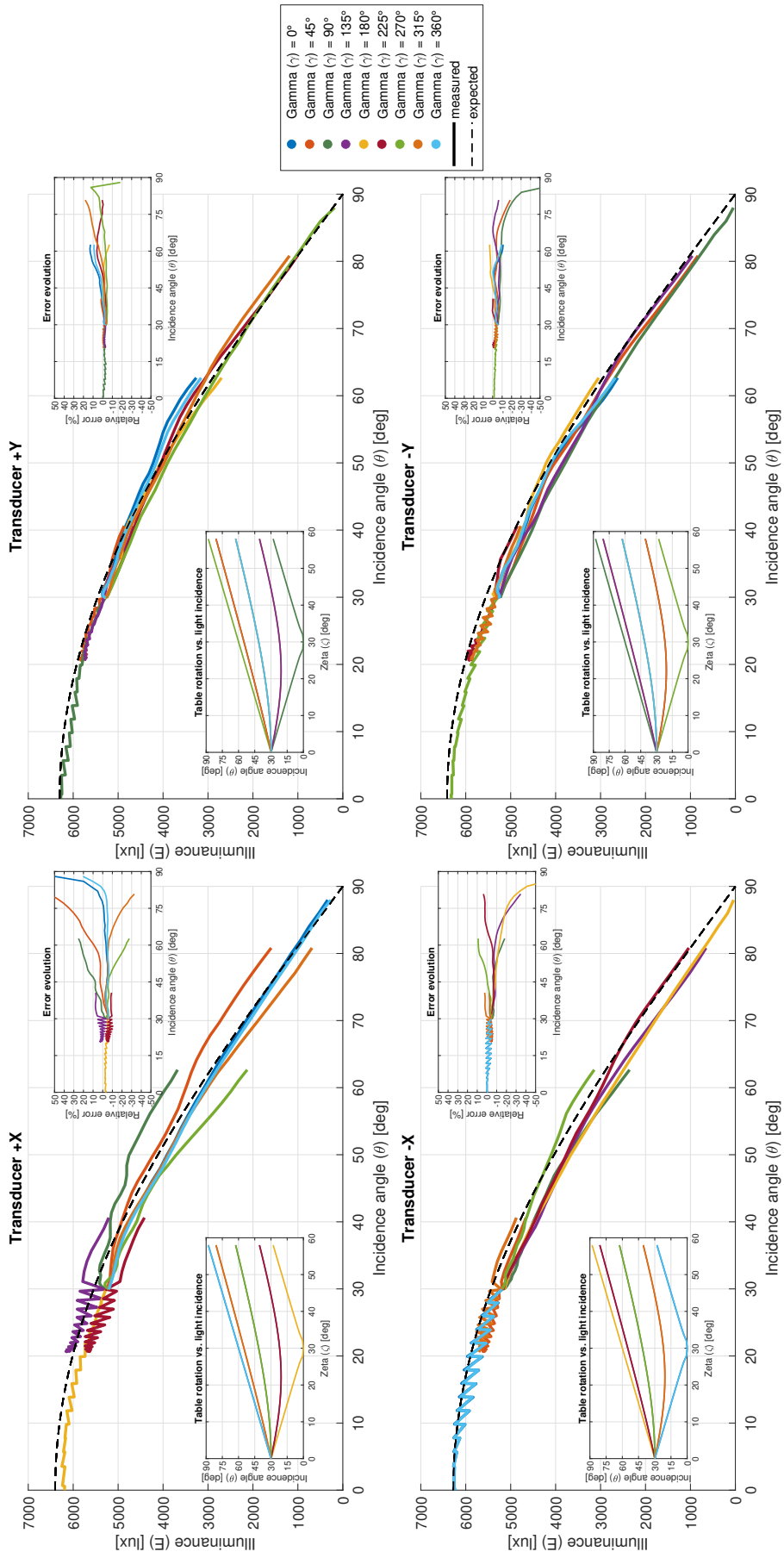


Figure 47: Illuminance dependence on the incidence angle after light evolution calibration (first sensor).

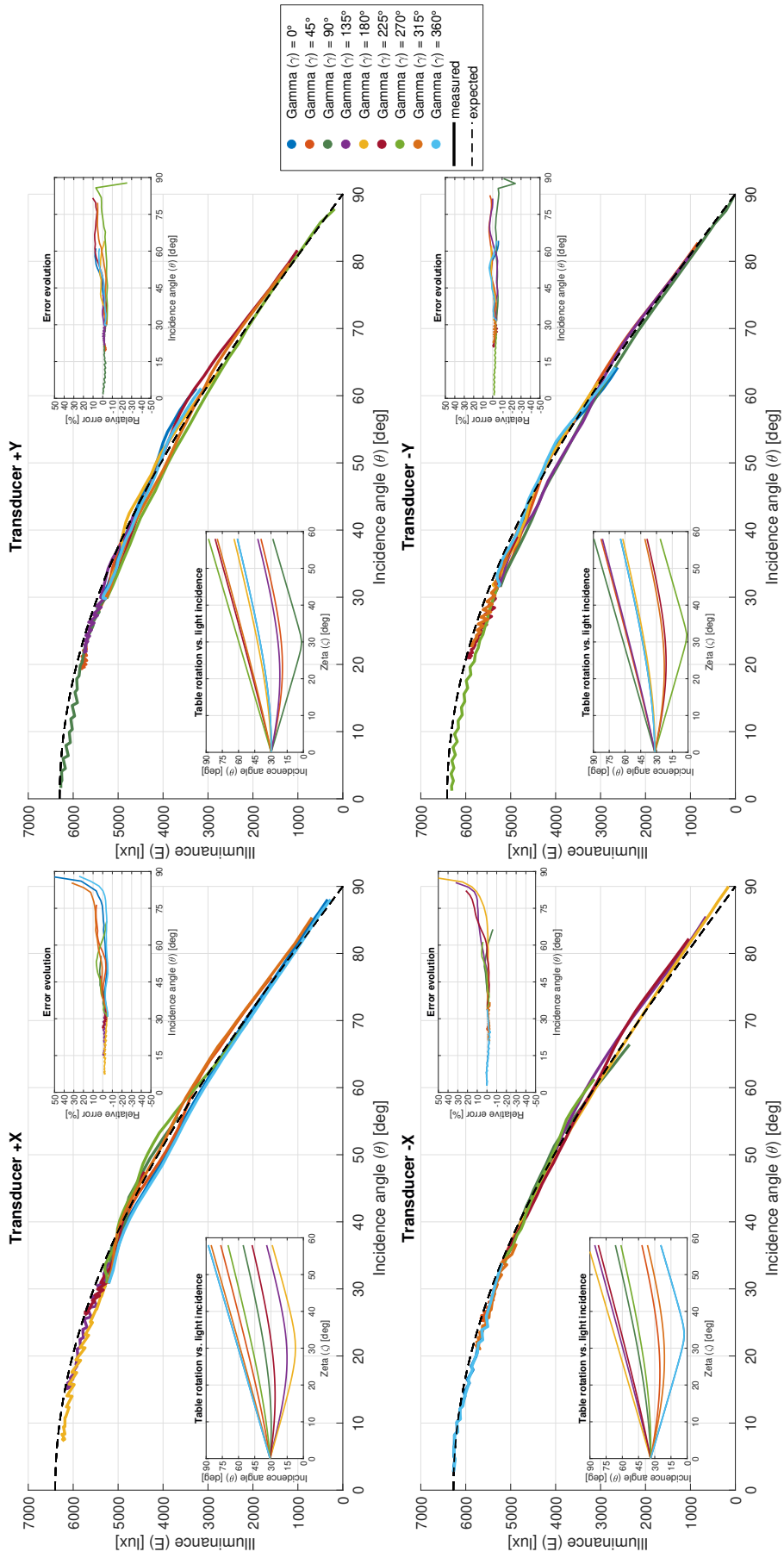


Figure 48: Illuminance dependence on the incidence angle after normal vectors calibration (first sensor).

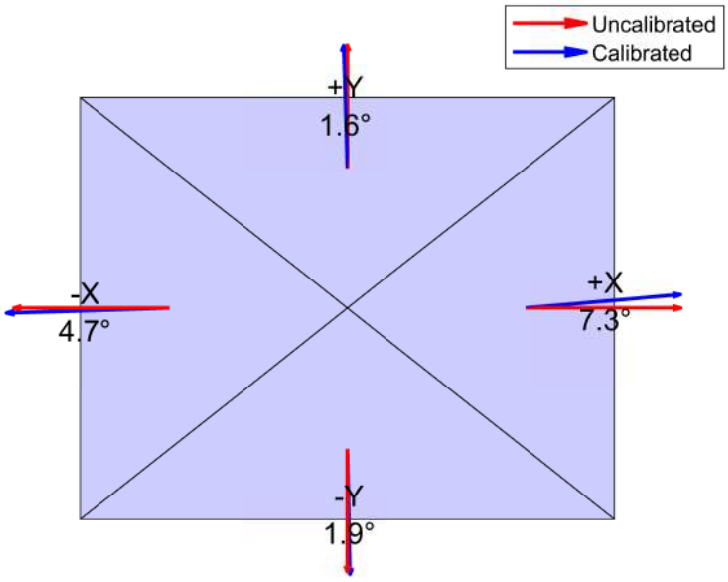


Figure 49: Comparison between transducers' normal vectors (first sensor).

As this error increase was unexpected, a reverse approach was employed by fitting the illuminance data to Equation (48), which models the non-linear behaviour, so a better estimate of the index could be calculated. Using the Levenberg-Marquardt algorithm, it was observed that an almost perfect fit is achieved, with an average value of $r_t = 1.02$ for all transducers, which is significantly different from the epoxy's $r_t \approx 1.5$. This deviation could mean that the sensing element was protected by an uncommon type of material or even not protected at all. This new index was updated, and the calibration step was restarted. Figure 52 shows a clear improvement in the error, where at the low incidence regions, the measurement curve overlapped the expected. As defined earlier for simplicity, the expected curve was based solely on the cosine law. However, this curve did not deviate from the non-linear one, as with this index of refraction, the transmittance impact was practically null.

After this careful analysis of the illuminance evolution across the calibration framework, the accuracy of the Sun vector determination was assessed. As explained in the previous section, this metric was verified after each cumulative calibration step.

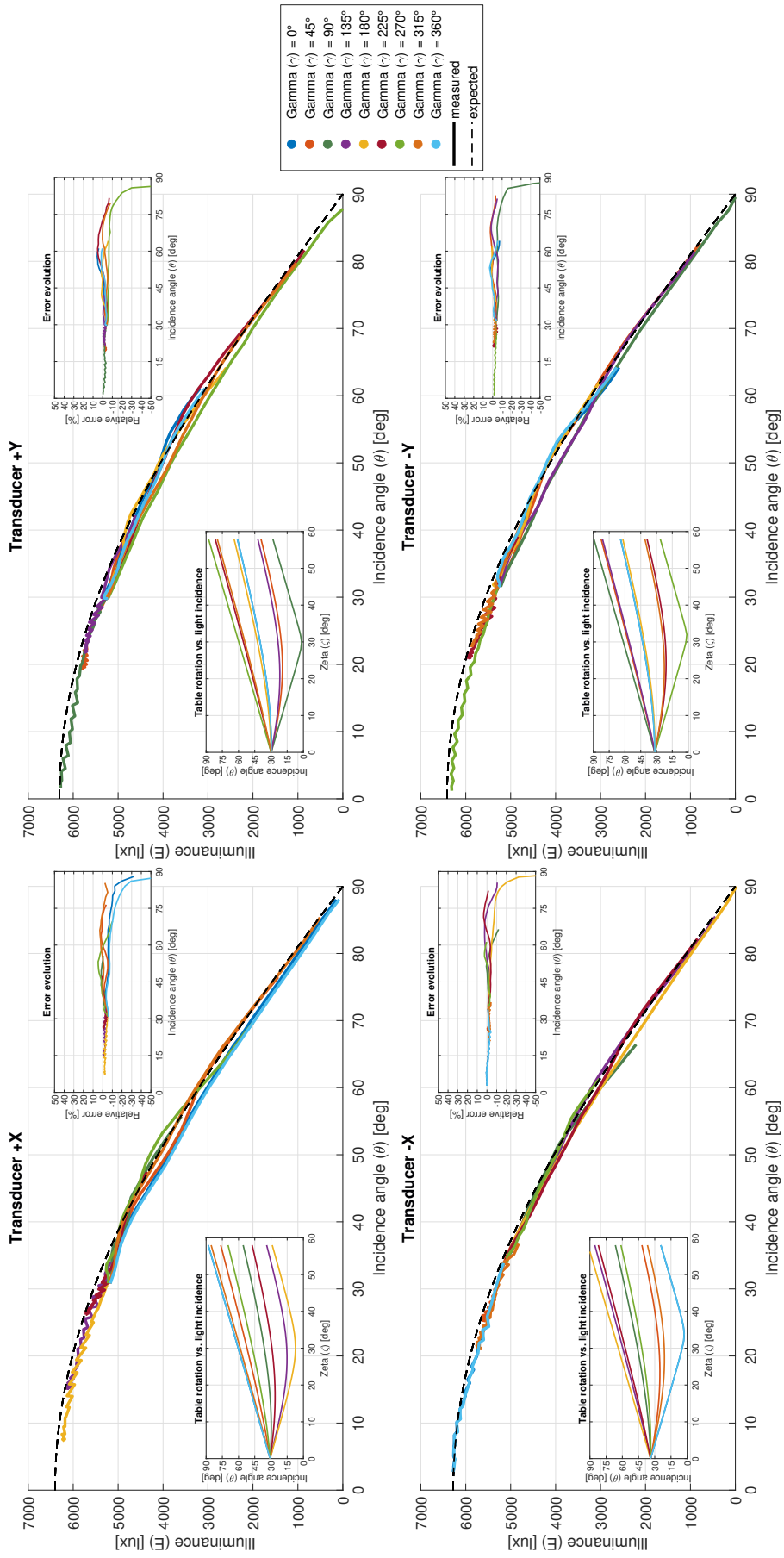


Figure 50: Illuminance dependence on the incidence angle after characteristics calibration (first sensor).

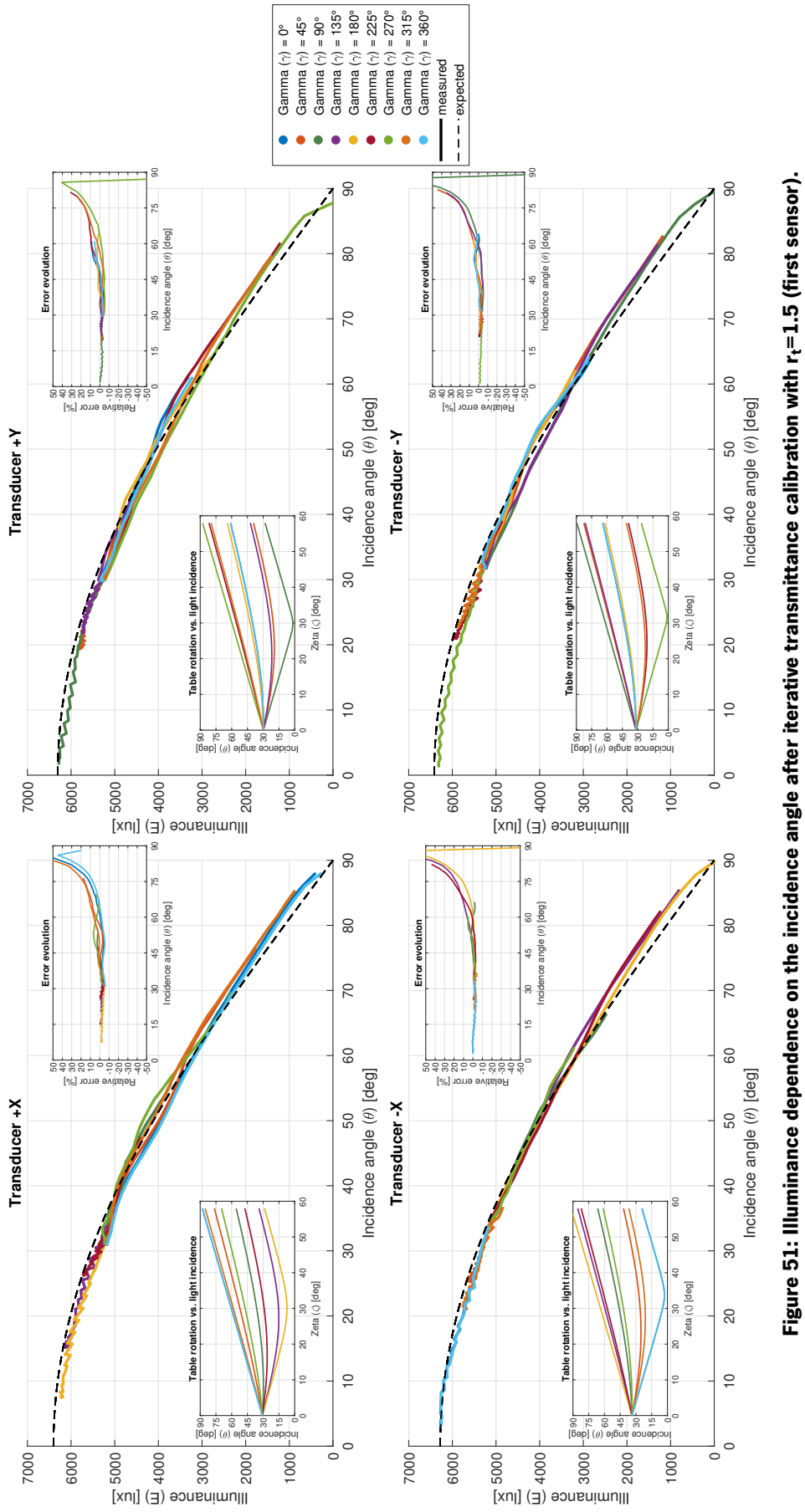


Figure 51: Illuminance dependence on the incidence angle after iterative transmittance calibration with r_t=1.5 (first sensor).

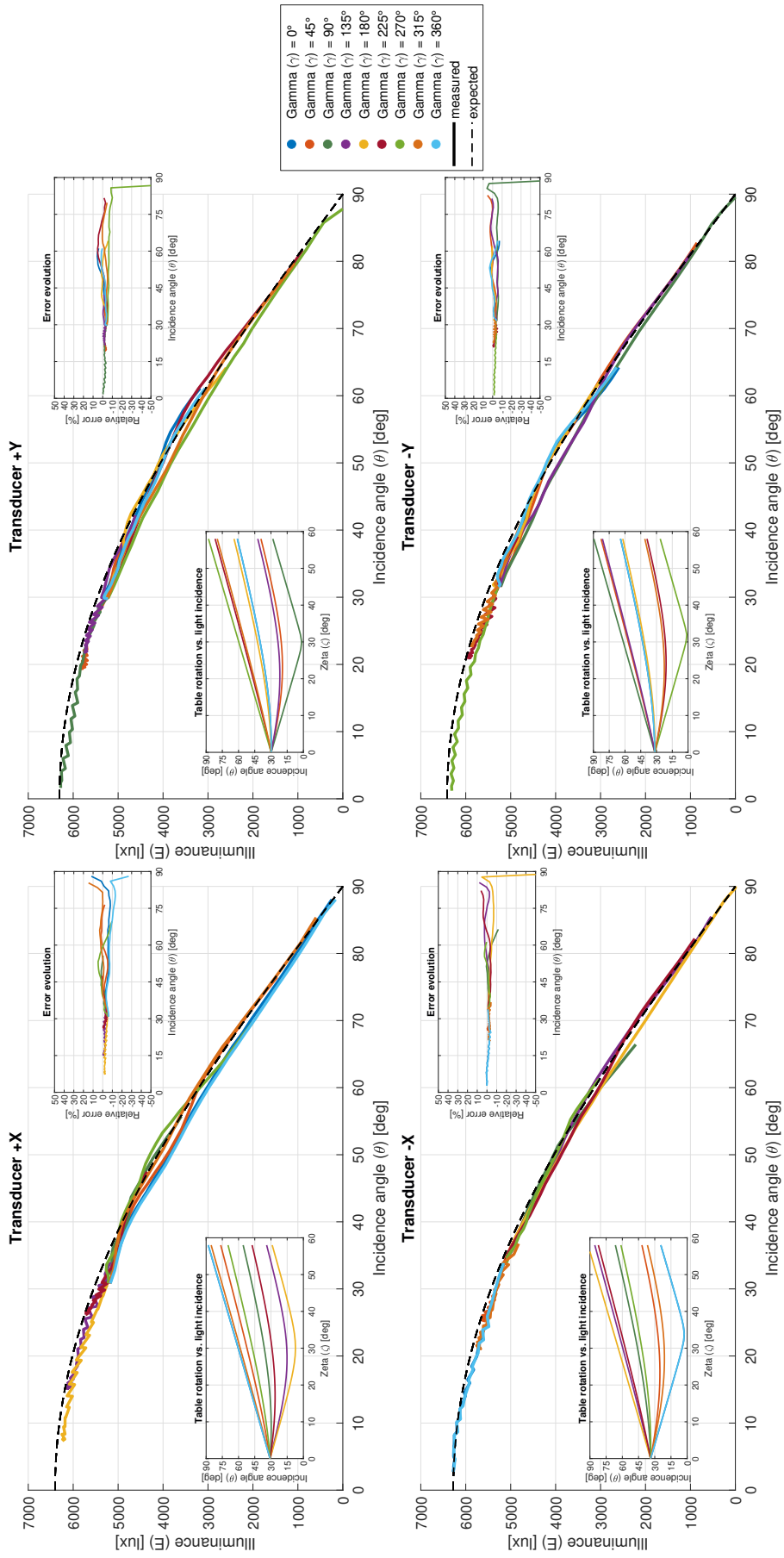


Figure 52: Illuminance dependence on the incidence angle after iterative transmittance calibration with r_t=1.02 (first sensor).

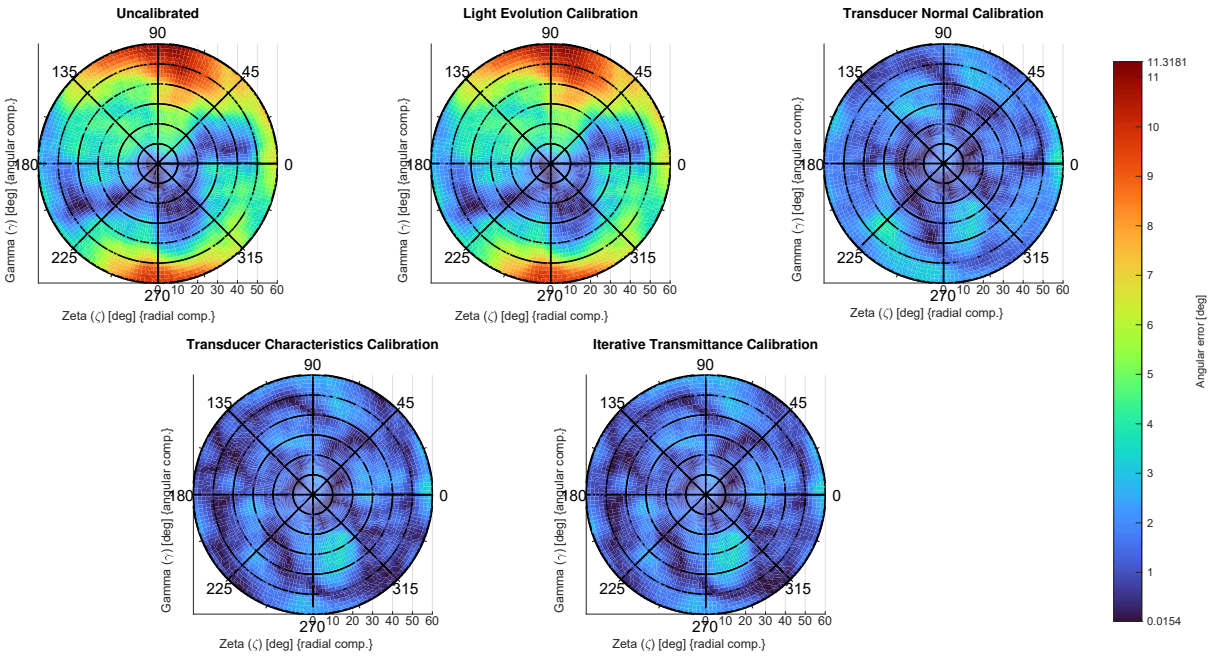


Figure 53: Sun vector determination angular error of the advanced prototype (first sensor).

As perceived in Figure 53, the uncalibrated data had the worst performance, as expected, with a mean error of 3.59 degrees and a standard deviation of 2.44 degrees. Furthermore, the error was not evenly distributed across the A-FOV, as the north and south plot zones, with $\zeta > 40^{\circ}$ together with $45^{\circ} < \gamma < 135^{\circ}$ and $250^{\circ} < \gamma < 300^{\circ}$, respectively, reached errors above 11 degrees. Looking back to Figure 46, these gamma clusters are mostly represented by the green, orange and purple curves, which tended to deviate more from the expected as theta increased, and consequently, as zeta increased above that threshold. This behaviour is evident in all transducers, but especially on +X. Since the determination solution is geometrically linked to the difference between the pairs of transducers, the ±X difference in those regions is accentuated, thereby increasing the error.

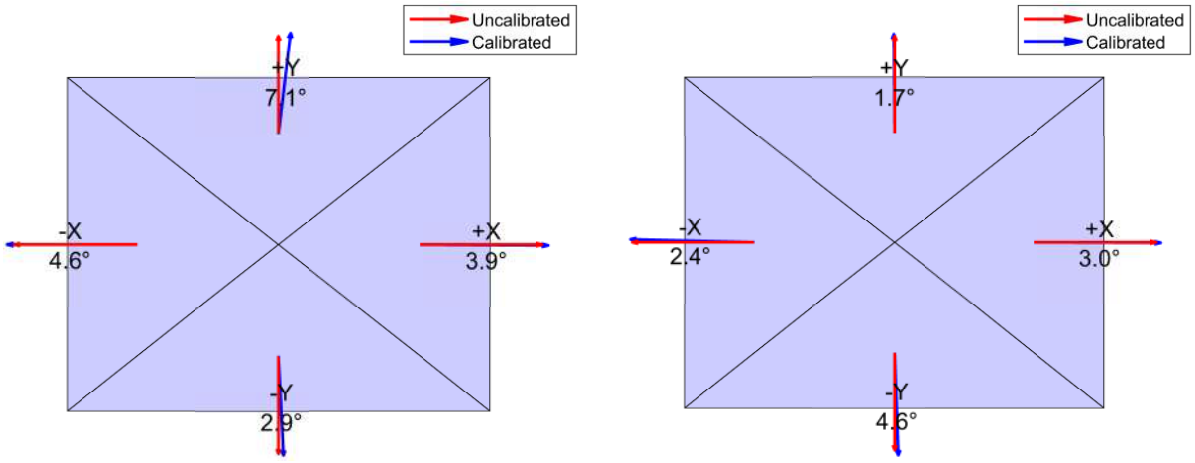
The light evolution calibration maintained the error at the same level, as expected, whereas the normals' calibration showed larger improvements. Together with reducing the error mean to 1.59 degrees and the standard deviation to 0.75 degrees, it also reduced the maximum error to 4.11 degrees. Overall, the error was also more uniformly distributed.

The characteristics calibration further improved the accuracy to 1.39 degrees with a 0.68 degrees deviation. The maximum error was reduced by half a degree and was approximately concentrated in the $(\gamma, \zeta) \approx (300^{\circ}, 30^{\circ})$ region. Notably, this was one of the regions with the least error across the A-FOV until the normal vectors were updated.

After the last calibration step, the accuracy remained approximately the same, with a slight increase

in the average, which could be attributed to the fact that the refractive index selected for calibration was defined as the average of all transducers rather than using individual parameters for each.

The illuminance measured by the second and third assemblies of the prototype yielded results very close to those of the first. In addition, after each calibration step, the 2D plots of illuminance versus incidence angle also repeated most of the patterns observed on the first, both in the same and in different regions of the A-FOV and can be consulted in Appendix A2. Figure 54 shows the correction made for the normal vectors that, for the second assembly, also shows a 7-degree correction, this time on the +Y transducer. At the same time, on the third, the calibration was moderate, with a maximum deviation of 4.6 degrees on the -Y transducer.



(a) Second sensor.

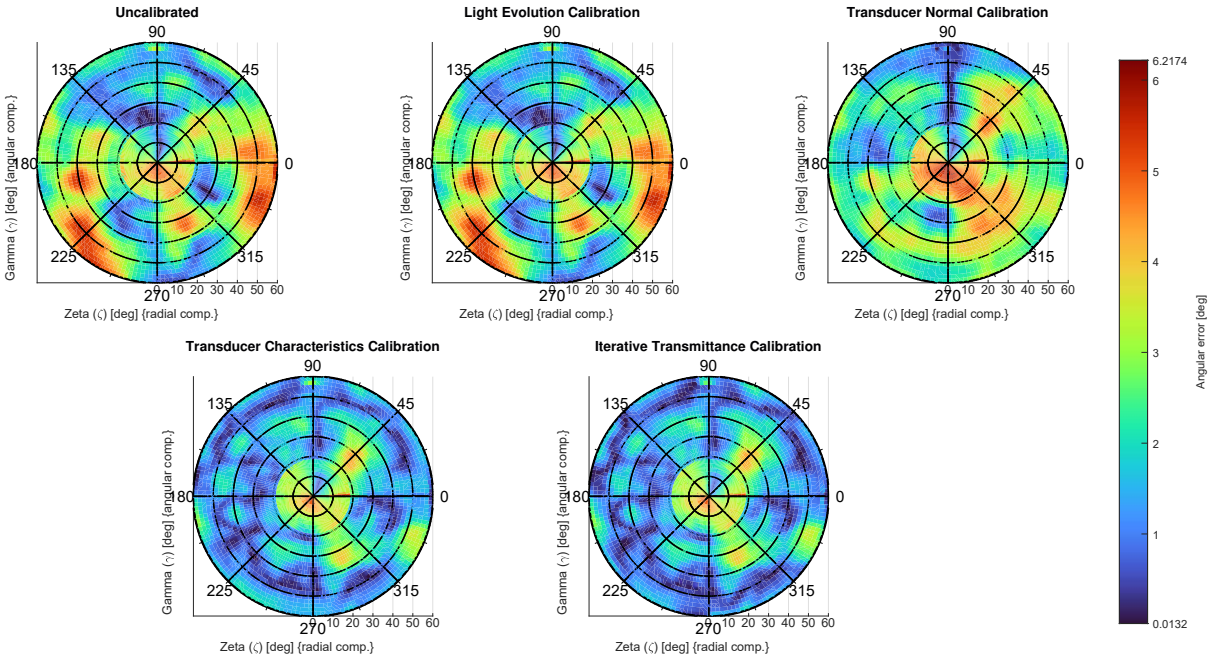
(b) Third sensor.

Figure 54: Comparison between transducers' normal vectors (second and third sensor).

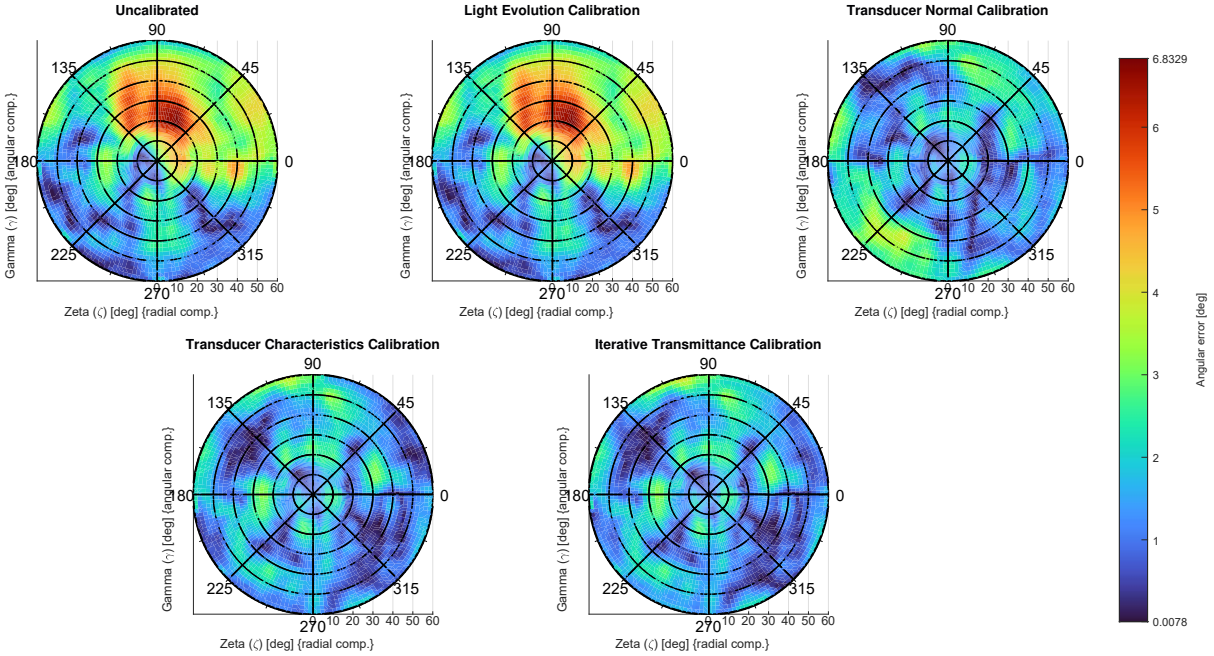
Analysing the accuracy of these assemblies in determining the light, the maximum error was around 6 degrees in both, which was half of what it was before. The second assembly had its error initially distributed around the origin and towards $\gamma \approx 225°$ and $\gamma \approx 340°$, but the normal calibration reduced the discrepancies around the A-FOV except on the central region, at the expense of a slight mean error increase. However, the characteristics step substantially reduced the error.

On the other hand, the third CSSPA had its error concentrated very much at $\gamma \approx 90°$, which was improved during the normal calibration. The characteristics step reduces the error primarily across the edges. Figures 55a and 55b picture the accuracy distribution of these sensors, respectively.

The calibration parameters obtained across the procedure for each sensor were compiled on Table 7. In addition, the mean, standard deviation, and amplitude of accuracy over the A-FOV were calculated between each step and presented in Table 8, showing their evolution over the calibration framework.



(a) Second sensor.



(b) Third sensor.

Figure 55: Sun vector determination angular error of the advanced prototype (second and third sensor).

Table 7: Calibration parameters estimated during the framework.

Pyramid ID	Transducer ID	Calibration parameters			
		Normal vector ($\check{n}$)	Curve gain (α)	Curve bias (κ)	Refraction index (r_t)
1	+X	[0.496; 0.127; 0.859]	6078.4	194.4	1.02
	+Y	[-0.028; 0.497; 0.868]	5975.2	181.3	1.02
	-X	[-0.555; -0.048; 0.830]	5928.3	227.1	1.02
	-Y	[0.019; -0.523; 0.852]	6217.8	54.2	1.02
2	+X	[0.556; -0.010; 0.831]	5714.9	456.5	1.02
	+Y	[0.115; 0.535; 0.837]	5839.1	286.5	1.02
	-X	[-0.568; 0.000; 0.823]	6133.2	347.6	1.02
	-Y	[0.047; -0.518; 0.854]	5951.6	104.2	1.02
3	+X	[0.544; -0.006; 0.839]	5,913.8	167.1	1.02
	+Y	[-0.008; 0.524; 0.852]	6,084.0	170.9	1.02
	-X	[-0.518; 0.036; 0.855]	5,988.6	198.1	1.02
	-Y	[0.033; -0.561; 0.827]	6,129.2	86.8	1.02

Table 8: Accuracy performance results after each calibration phase.

Pyramid ID	Metrics	Calibration step					Total difference	
		Uncalibrated	Light evo.	Normal vec.	Characteristics	Transmittance	Absolute	Relative
1	Mean	3.59°	3.59°	1.59°	1.39°	1.41°	-2.18°	-60.7%
	St.dev.	2.44°	2.44°	0.75°	0.68°	0.68°	-1.76°	-72.1%
	Max.	11.32°	11.32°	4.11°	3.61°	3.60°	-7.72°	-68.2%
	Min.	0.04°	0.05°	0.03°	0.02°	0.04°	0.00°	0.0%
2	Mean	2.72°	2.72°	2.93°	1.98°	1.94°	-0.78°	-28.7%
	St.dev.	1.25°	1.25°	1.24°	1.13°	1.16°	-0.09°	-7.2%
	Max.	5.84°	5.84°	6.22°	5.41°	5.41°	-0.43°	-7.4%
	Min.	0.02°	0.01°	0.06°	0.02°	0.02°	0.00°	0.0%
3	Mean	2.60°	2.60°	1.52°	1.46°	1.47°	-1.13°	-43.5%
	St.dev.	1.47°	1.47°	0.74°	0.67°	0.67°	-0.80°	-54.4%
	Max.	6.82°	6.83°	3.93°	3.97°	4.19°	-2.63°	-38.6%
	Min.	0.04°	0.05°	0.01°	0.02°	0.01°	-0.03°	-75.0%
Avg.	**Mean**	**2.97º**	**2.97º**	**2.01º**	**1.61º**	**1.61º**	**-1.36º**	**-45.9%**
	St.dev.	**1.72º**	**1.72º**	**0.91º**	**0.83º**	**0.84º**	**-0.88º**	**-51.4%**
	Max.	**7.99º**	**8.00º**	**4.75º**	**4.33º**	**4.40º**	**-3.59º**	**-45.0%**
	Min.	**0.03º**	**0.04º**	**0.03º**	**0.02º**	**0.02º**	**-0.01º**	**-30.0%**

Overall, without calibration, the developed CSSPA had twice the angular accuracy compared to similar examples in the literature under the same conditions. After calibration, although it yielded substantial improvements, it performed as well as the other sensors. Averaged across all three pyramids, the mean angular error decreased from 2.97 degrees in the uncalibrated state to 1.61 degrees after full calibration,

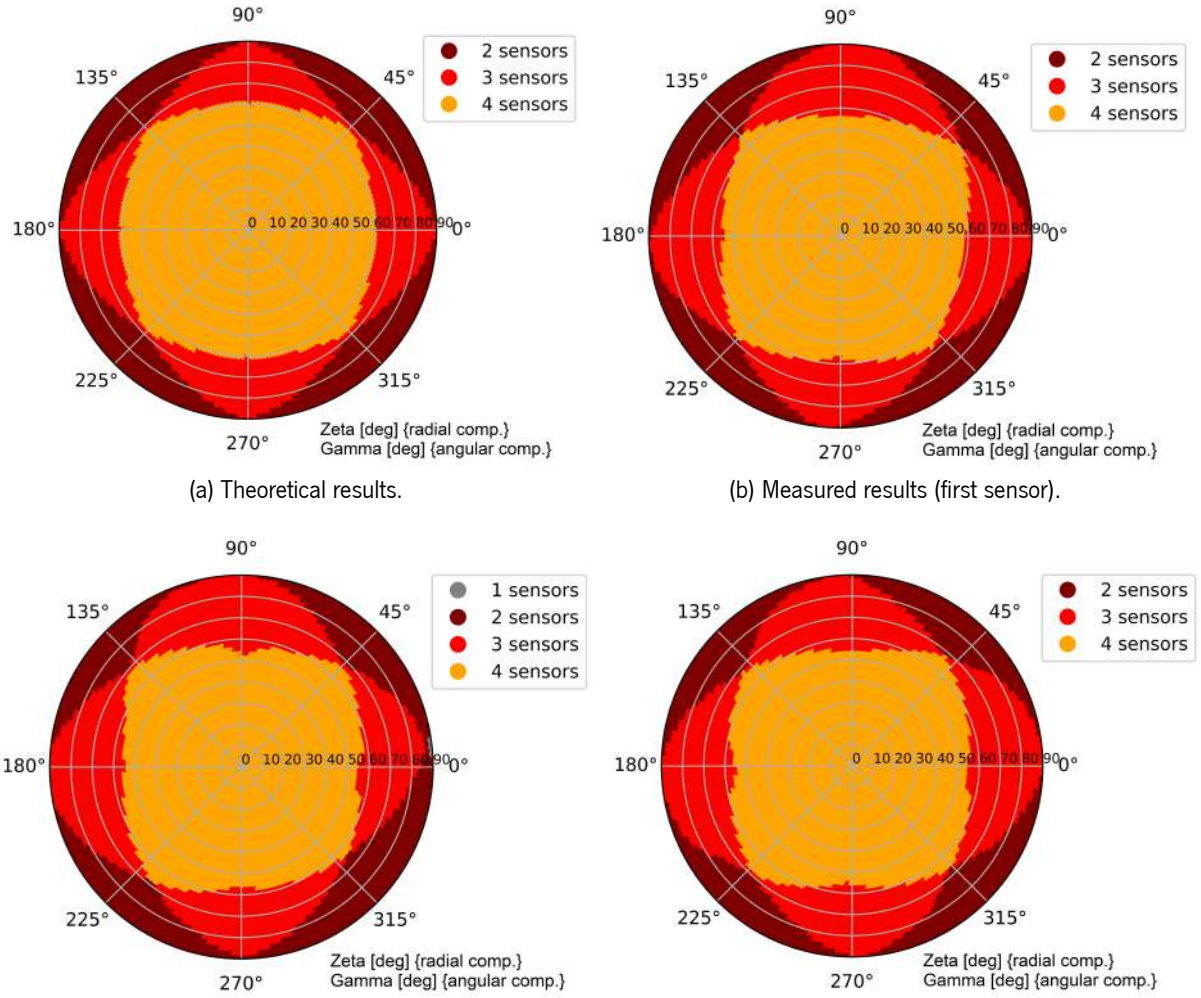
corresponding to an absolute improvement of 1.36 degrees and a relative reduction of 45.9%. Notably, the standard deviation was also significantly reduced, from 1.72 to 0.88 degrees on average, indicating improved consistency.

The first CSSPA demonstrated the most pronounced improvement, with the mean error dropping from 3.59 to 1.41 degrees, yielding an absolute improvement of 2.18 degrees and a relative reduction of 60.7%. The maximum error also decreased markedly, from 11.32 to 3.60 degrees, highlighting the robustness of the calibration process in this case. In contrast, the second assembly exhibits the least improvement, with a mean error reduction of only 0.78 degrees. The third pyramid shows moderate improvement, with its mean error decreasing from 2.60 to 1.08 degrees, equating to a 58.3% relative reduction.

Examining the contributions of individual calibration stages, the lighting evolution step did not alter the error metrics and was therefore ineffective once the flashlight intensity was stabilised, as perceived earlier. Significant gains were achieved during the normal vector and characteristic adjustment phases. These steps consistently reduced the mean error across all pyramids, with the normal vector refinement being particularly impactful. The final transmittance correction step provided only marginal additional benefits, suggesting diminishing returns at this stage of calibration, as this particular non-linear behaviour was not as pronounced as initially expected.

Lastly, a brief analysis of the CSSPA coverage was also conducted. Again, to make the employed concepts clear, the coverage was defined as the region where the Sun can be uniquely determined or, in other words, where three or more T-FOVs intersect. The A-FOV, as explained in the literature review, is sometimes addressed the same as the coverage, or more commonly, as the region where *all* T-FOVs intersect, as assumed during this section. Obviously, with these definitions, since the A-FOV requires more intersections, it is a shorter region with better accuracy. However, the coverage is also important, as it is wider and still achieves its purpose. Following a similar approach described in Section 4.2, the theoretical number of transducers with visibility to the light source was calculated for a single CSSPA, this time extending each T-FOV to 90 degrees and limiting the attitude sphere to the top hemisphere. In parallel, and for each assembled prototype, the same number was calculated based on the uncalibrated illuminance measured at each point of the same hemisphere, excluding noisy measurements with less than 200 lx, value based on OPT4001 datasheet.

Figure 56 shows the obtained results for each sensor, where three plot differences are present compared with Figure 31. The first is clear, as the plots were arranged in polar coordinates so that coverage was represented in the same format as accuracy. The second, not so obvious, is that since the dataset



(a) Theoretical results.

(b) Measured results (first sensor).

(c) Measured results (second sensor).

(d) Measured results (third sensor).

Figure 56: Sun vector determination coverage of the advanced prototype.

collection proceeded with equal steps in both rotation tables, the attitude sphere was therefore discretised as an azimuth/elevation sphere. Although not ideal, this decision was made to ensure that the resumption of sampling between each pause for wire untwisting could be performed correctly, visually confirming the current rotation angles before taking the next measurements. As a result, explained in Section 4.2, there was a higher point density at the pyramid boresight; however, due to the high number of samples, the coverage limits could still be determined. Lastly, the axes were constructed in terms of the table rotation angles rather than azimuth and elevation, which have equivalent meanings.

The measured results were remarkably similar to the theoretical ones despite the misalignments of the transducers. In this discretisation, while the first displays an 85.9% coverage, the assembled pyramids achieved 88.6%, 88.0% and 89.0%, respectively. The slight increment could be attributed to noise still present in the low-incidence measurements, particularly near the threshold and at the edges of the T-FOVs.

For reference, the theoretical coverage for a Fibonacci sphere dropped to 79.4%, which can be converted to approximately 4.99 steradians of the 6.28 steradians possible of a full hemisphere, or approximatelly 170x170 degrees. This was a pleasing result and in line with the current literature.

4.5.2 Combined pyramids case study

After loading the calibration coefficients displayed in Table 7 and defining the new sensor normals to match this setup's orientation, the LS estimator was used to determine the Sun vector using the combined trial illuminance measurements. Since the number of measurements tripled, the 200 lx threshold from earlier was adopted as a simple noise filter, while the remaining available measurements were combined for the estimation in each direction. Similarly, as in the previous scenarios, this was performed at each calibration step. However, as the light evolution and iterative transmittance calibrations proved to be rarely effective, they were excluded from this specific case.

Table 9 shows the results obtained. Before applying the calibration coefficients, the data presented a 2.65 degrees mean accuracy with a 1.28 degrees deviation, which is a slight improvement from the average of the three pyramids when separated. However, after introducing the calibration coefficients discovered earlier, these metrics became worse by approximately 40%. As unexpected as it may seem, this behaviour was justified by the fact that the FPC had to be removed from its initial pyramid, where the transducers were initially calibrated, to this new assembly. While the previous calibration could mitigate the transducer's misalignments relative to the FPC, the misalignments of the FPC relative to the structure were redefined. In addition, the transducer characteristics calibration process, which is dependent on calibrated normals, increments the issue.

Table 9: Accuracy performance results with the three combined sensors.

Metrics	Calibration step			Total difference	
	Uncalibrated	Normal vec.	Characteristics	Absolute	Relative
Mean	2.65°	2.98°	3.74°	1.09°	41.1%
St.dev.	1.28°	1.37°	1.75°	0.47°	36.7%
Max.	7.28°	6.94°	8.52°	1.24°	17.0%
Min.	0.03°	0.01°	0.04°	0.01°	33.3%

The error distribution was also plotted, as pictured in Figure 57, which shows that the error accumulated at specific regions, namely $\gamma \approx 90°$, $\gamma \approx 215°$, $\gamma \approx 330°$ at $\zeta > 60°$. These regions had fewer visible transducers, and the distribution of transducers per pyramid was unbalanced, with four transducers from

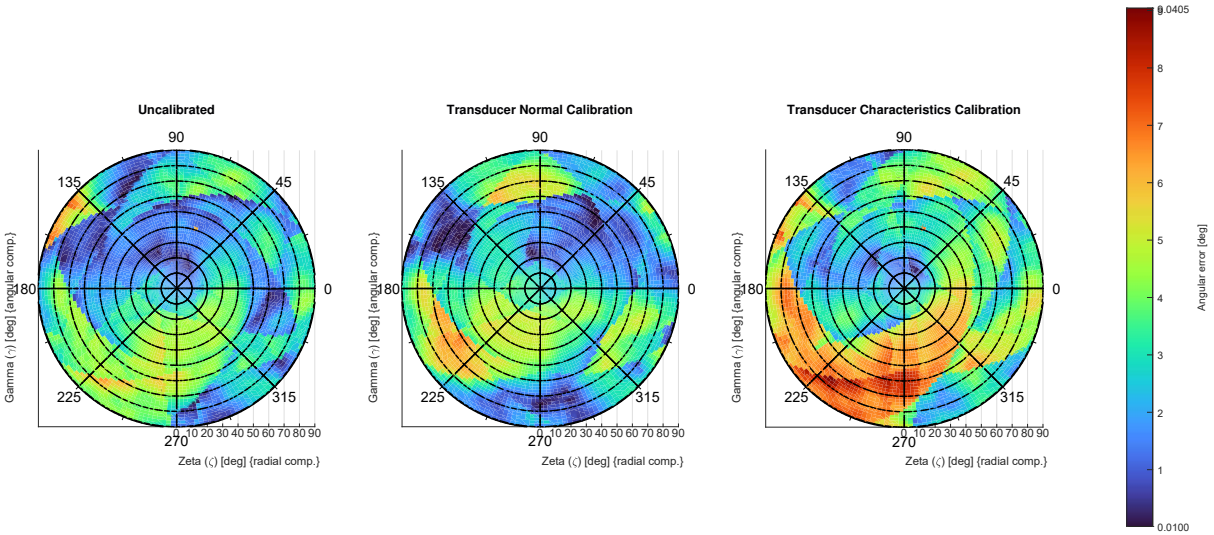
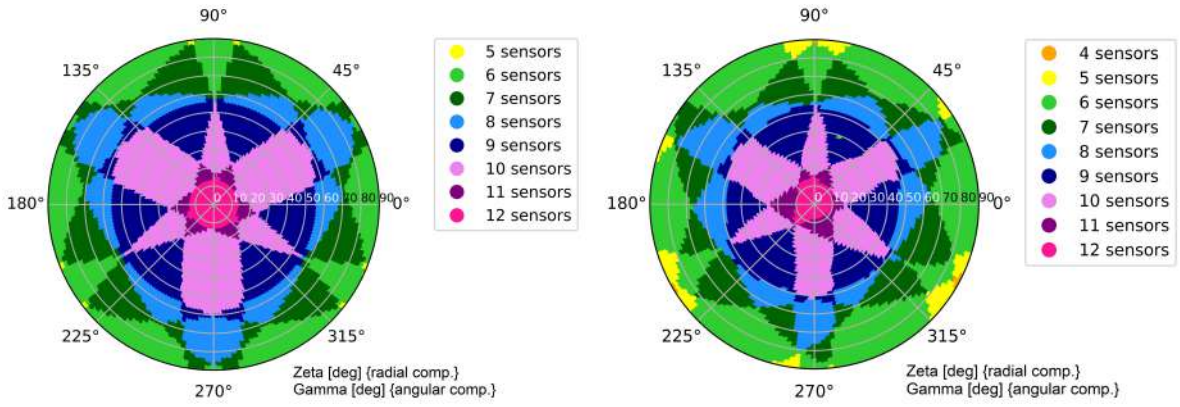


Figure 57: Sun vector determination angular error of the advanced prototype (three sensors combined).

the same pyramid and one or two from the remaining pyramids. This condition was the most likely to create the erroneous behaviour.

Figure 58 introduces the measured combined coverage, and the boundaries of these regions are clearly visible. The theoretical coverage had a similar distribution but with more symmetry, as expected. Again, note that the density of available transducers was higher than the one shown in Section 4.2 because this time, the T-FOV was set to 90 degrees, while previously it was limited to the transducer's HPA, as the analysis was more conservative and applicable to different transducers, even with FOV cutoffs. Although the regions with fewer transducers were larger than expected, they were also less available, yet the Sun can still be estimated from any position within the hemisphere. Thus, 4π steradians V-FOV on PROMETHEUS-1 with incremented redundancy can be achieved.



(a) Theoretical results.

(b) Measured results.

Figure 58: Sun vector determination coverage of the advanced prototype (three sensors combined).

4.6 Summary

This chapter detailed the development of the advanced prototype for the proposed CSSPA, focusing on design refinement, manufacturing innovation, and comprehensive testing.

In the array topology optimisation (Section 4.2), five CSSPAs configurations were evaluated through Monte Carlo simulations over a discretised attitude sphere using a Fibonacci grid. The square pyramid with four sensors tilted at 30 degrees proved optimal, achieving 100% hemispherical coverage and high resilience to sensor failure, maintaining over 90% coverage with up to eight transducer faults, and substantially outperforming the configuration currently flying on PROMETHEUS-1.

In the miniaturisation phase (Section 4.3), a FPC manufactured by space-grade polyimide was introduced, drastically reducing volume while enabling a more mechanically adaptive assembly. A new structural support with softened edge transitions and modular interfaces was developed to accommodate FPC bending without damaging traces. The final assembly integrated three identical prototypes via a compact, reconfigurable platform, with simplified electrical interfacing to the MCU via a custom PCB.

The experimental methodology (Section 4.4) incorporated a dual-axis automated rotation setup inside an anechoic chamber. Illumination stability was modelled and compensated using a fitted exponential decay curve, while a conical rotation path allowed complete sampling over the visible attitude sphere. Careful alignment through modelling minimised translational error, and high measurement precision was enabled by the high-resolution rotation stages with no backlash.

In the results analysis (Section 4.5), Sun vector estimation was calibrated in three sequential stages: transducer orientation, sensor characteristics, and iterative transmittance. The method followed a pipeline framework from literature for these types of sensors and successfully reduced Sun vector angular error by approximately 50% to below 5 degrees across most of the 5 steradians coverage, with an average of 1.61 degrees and a standard deviation of 0.84 degrees. This performance range is typical for similar sensors, but without calibration, it achieved at least twice the accuracy. Despite overall remarkable performance, several error sources limited the prototype's accuracy. Three main contributors were identified: mechanical misalignments, inconsistencies in the light source, and transducer underperformance. Misalignments were possible across multiple components, including the rotation stages, supports, pyramids, and even the transducers themselves. The flashlight used as a light source exhibited poor beam collimation, unstable power delivery, and temperature-dependent output. At the same time, nearby reflections and shadows from the pyramid's marker poles may have introduced additional noise. Lastly, the transducers may have exhibited behavioural changes or optical deformities caused during manufacturing or handling,

either within internal components or, more likely, in the sensing element itself, which could feature an uncommon, defective or even absent optical cover.

Compared to state-of-the-art alternatives, it offers a compelling trade-off between performance, redundancy, and manufacturability. Nonetheless, several improvements and insights were identified for the next iteration of sensor development and testing. These include further reducing the pyramid size, the addition of shielding layers to the FPC, and improving its mechanical fixation to the support to prevent misalignments or detachment. Additionally, its location on PROMETHEUS-1 must be evaluated, and an integration plan must be drafted to ensure readiness for flight on future satellite iterations. Regarding the testing setup, due to the misalignments encountered during the calibration process, an optical table, a calibrated AM0 light source, and metal adapters should be employed to objectively assess the sensor accuracy prior to flight. An on-orbit routine could also be developed to perform in-situ calibration, as the original framework was designed for such scenarios.

In conclusion, the advanced prototype successfully demonstrated robust and accurate Sun vector determination, full-sky observability, and fault-tolerant performance in a small, modular form factor.

Chapter 5

Final prototype

5.1 Introduction

Following the development and evaluation of the advanced prototype, this chapter introduces the final design iteration of the CSSPA, intended for full integration into the PROMETHEUS-1 satellite. This version incorporated lessons learned from previous prototypes and testing campaigns, addressing the mechanical, electrical, and environmental constraints necessary for flight readiness.

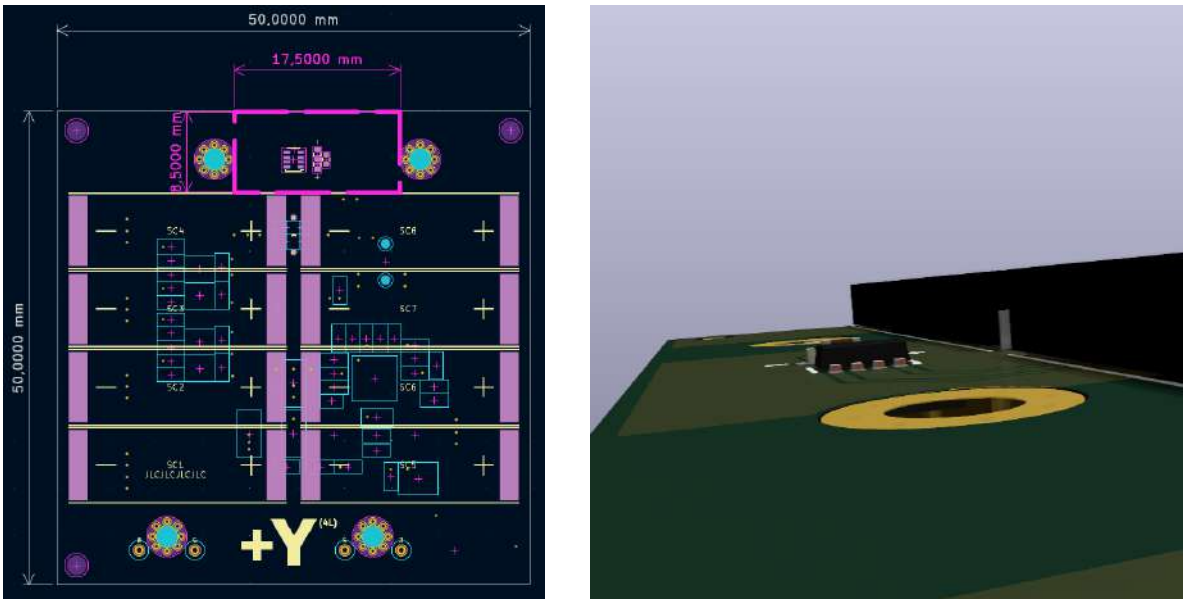
Key motivations for the redesign included compliance with the PocketQube volume standard, avoidance of T-FOVs obstructions, and improved sensor performance under expected illumination and thermal conditions. The integration strategy extended to interface compatibility with the satellite's current architecture.

However, only the detailed design and initial validation stages were completed. The actual hardware assembly and onboard integration were not pursued due to calendar constraints. Nevertheless, almost all critical aspects necessary for future implementation, from the usual design considerations to assembly procedures, were thoroughly addressed.

Section 5.2 outlines the structural redesign using space-grade materials, high-density FPC layout and sensor upgrades, while Section 5.3 focus on the proposed satellite hardware and software modifications to support the full integration of the CSSPA. Section 5.4 recaps the usual key findings.

5.2 Array modifications

As pointed out in the last chapter, a significant size reduction was necessary to fit the developed CSSPA into the satellite. To achieve this, all major hardware constraints were drafted and rigorously evaluated. Starting from the pyramid's maximum allowed height, it was found that the envelope around a PocketQube should be no more than 7 mm, as defined by the PocketQube standard [76]. Estimating the edge size of a 7 mm height pyramid with 30-degree tilted faces results in a squared base of approximately 24 mm in edge length. In perspective, this represents approximately one-quarter of the total surface area of the external panels, which is unavailable due to the solar cells. Looking closely at the current available space on the satellite's PCBs, the largest region consists of a 17.5 mm by 8.5 mm rectangle on the top of the board, where the Sun sensor is currently placed. This area is shown in Figure 59a, specifically on the +Y solar panel, but it is the same across all other boards, except for the -Y board, which, due to the panel's different layout, is slightly shifted towards the centre. Defining a 1 mm total margin, the pyramid's base was constrained to 7.5 mm edges, which translated to an approximate height of 2.17 mm, without truncation of the top vertex, thus satisfying the maximum height limit with a safe margin.



(a) Location and dimension, in purple.

(b) 3D rendering showing obstructions by solar cells.

Figure 59: Available region for the CSSPA in the satellite's solar panels.

A simple 3D rendering of the currently available region on the solar panels shows that the OPT4001 has some T-FOV obstruction by the solar cells, as portrayed in Figure 59b. This was inevitable as the transducer must be soldered directly to the PCB. However, since the FPC can physically extend the distance between

REMAINING PCB	H. MARGIN	BLIND SLOT	H. MARGIN				
	<- 0.5 mm ->	<---------- 7.5 mm ---------->	<- 0.5 mm ->				
SOLDER MASK					-	-	0.007 mm
COPPER FOIL					-	-	0.035 mm
PREPREG (FR4)				V. MARGIN	0.200 mm		0.210 mm
COPPER FOIL					0.015 mm		0.015 mm
CORE (FR4)					1.065 mm		1.065 mm
COPPER FOIL					0.015 mm		0.015 mm
PREPREG (FR4)					0.210 mm		0.210 mm
COPPER FOIL					0.035 mm		0.035 mm
SOLDER MASK					0.007 mm		0.007 mm
				TOTAL	**1.541 mm**		**1.600 mm**
				DIFF.	**0.06 mm**		

Figure 60: Updated stackup for the PCB accounting blind slots.

the transducers and the connection pads, the height of the pyramidal support can also be increased to avoid shadows. An approximate height of 2 mm was measured on the solar cells, allowing the support to be raised above the nearby components, while still complying with the PocketQube standard.

Another key insight obtained from the last chapter is that misalignments are very often present. To ensure that the support is correctly oriented in the PCBs, blind slots can be added. These are precisely machined regions with a specific depth drilled into the board. However, manufacturers typically have constraints on adding these. For example, JLCPCB imposes a minimum slot width of 1 mm and a minimum distance to copper traces of 0.2 mm, either vertically or horizontally. Figure 60 shows the stackup of the satellite's 4-layer PCBs, together with the addition of a blind slot square with a 7.5 mm edge, the same as the pyramid's base. Note that the 1 mm horizontal margin previously defined was divided between the solar cell pads and the edge of the board, and still complied with the horizontal requirements. Additionally, a 0.2 mm vertical margin to the embedded magnetorquers present in the inner layers was achieved. This spacing leaves a maximum drill depth of 0.06 mm for the blind slot, which usually requires at least 0.2 mm. However, since this is a machining procedure, it might still be achievable, but it requires discussion with the manufacturer. Note that this depth must also be added to the base of the pyramid. Alternatively, if this approach is not possible, the two near screw holes can provide a precise reference to orient the pyramid.

The CSSPA support was then modelled w.r.t. all these constraints, following the same design principles as the previous prototype, as pictured in Figure 61. The necessary area for the transducers was accommodated on each face, and then the top vertex was truncated, resulting in a pyramid with a 1.2 mm height. This time, the base plate was replaced by a squared extrusion of 2.06 mm following the same layout. Only two marker poles were added because of the reduced space available. After this, the 2D layout of the

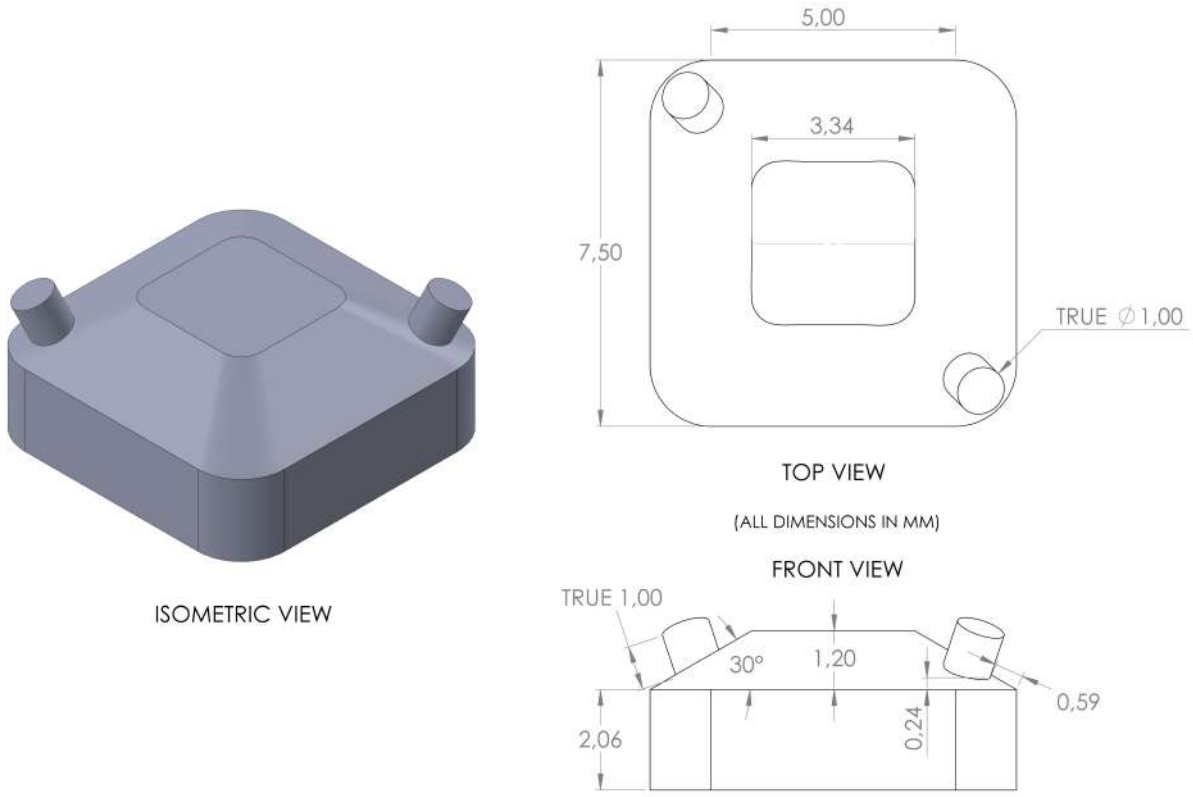


Figure 61: Final prototype support.

surface was exported in order to design the FPC, as already described.

While the pyramidal support was made of generic PLA in the previous prototypes, the flight version of the sensor requires more specialised materials to resist the launch phase, characterised by high vibrations, and endure the space environment, characterised by its harsh thermal cycles and absence of pressure. Windform's XT 2.0 was therefore selected, a polyamide-based carbon fibre-reinforced composite, specifically due to its high strength-to-weight ratio, stiffness, temperature resistance, and spaceproof compliance, as demonstrated by NASA and ESA outgassing tests. Although made using 3D-printing, a special technique called Selective Laser Sintering (SLS) is employed, enabling the creation of complex geometries with small tolerances. After contacting CRP Technology, ten supports were quoted at € 210.00, with 0.5 mm tolerances.

Regarding the electronic part of the sensor, the most significant limitation of the previously selected transducer is that it may saturate if the incidence angle is too high, as the Sun's irradiance would translate to an illuminance of 133,531 lx, as derived in Appendix A1. In contrast, the transducer range is limited to 117,000 lx. Fortunately, Texas Instruments recently released a new transducer of the same family. The OPT4003-Q1 offers a wider temperature and illuminance range, and features automotive AEC-Q100

certification, which indicates that it has undergone more rigorous qualification. Additionally, it includes a secondary near-infrared channel, providing another embedded source of data. A specific comparison between the differences between OPT4001 and OPT4003-Q1 was compiled in Table 10. Nevertheless, most operational features overviewed in Section 3.2 are still present on the latter. As one can see, the new package is different, which requires some rearrangement of the transducer footprints, but the size is almost identical. The component price has doubled, but it remains cost-effective.

Table 10: Properties differences between OPT4001 and OPT4003-Q1.

Part number	Range	Size (mm)	Cost (€)	Temperature (ºC)	Package
OPT4001DTSR	~0 – 117 klx [v]	2.1×1.9×0.6	1.32	-40 – 85	SOT-583
OPT4003DNPRQ1	~0 – 143 klx [v] ~0 – 27.45 mW/cm^2 [i]	2.0×2.0×0.6	2.71	-40 – 105	USON6

[v] Visible channel
[i] Infrared channel

As the height of the pyramid is relatively low compared to the available envelope, the addition of a fifth sensor at its truncated vertex was studied. As another photoelectric transducer would be too redundant, the most feasible and useful sensor would be a temperature one. This way, data would be available in a very close region to the transducers, enabling reliable temperature calibration, and also close to the solar cells, allowing for improved real-time energy production estimates. After some research, the Texas Instruments' TMP117, a high-precision, low-power digital temperature sensor, providing 16-bit measurements with an accuracy of ±0.3 °C, was selected. It can operate from -55 °C to 150 °C, similar thermal intervals that one would expect during a Sun-Synchronous Orbit (SSO) trajectory [77]. The TMP117 shares some similarities with OPT4003-Q1: has an I2C interface with configurable addresses (0x48 - 0x4B); selectable conversion times (15 ms - 16 s); an interrupt-like pin (ALERT pin); a similar package (WSON6); size (2 mm×2 mm×0.6 mm); and reasonable cost (€ 1.84). Note that the available I2C addresses on the sensor differ from those of the OPT4003-Q1, allowing them to be connected to the same bus. Other models from the TMP11x family have better performance but were discarded as their package is more difficult to handle.

The five components were added in KiCad, and the schematic was drawn, as shown in Figure 62. The multiple OPT4003-Q1 were assigned the same addresses as before, and the 0x49 address was selected in TMP117 due to its proximity to the ADDR pin.

The FPC design was conducted as already explained in the previous chapter, with Figure 63 showing the final result. The considerable reduction in size forced the FPC design to adopt almost the minimum

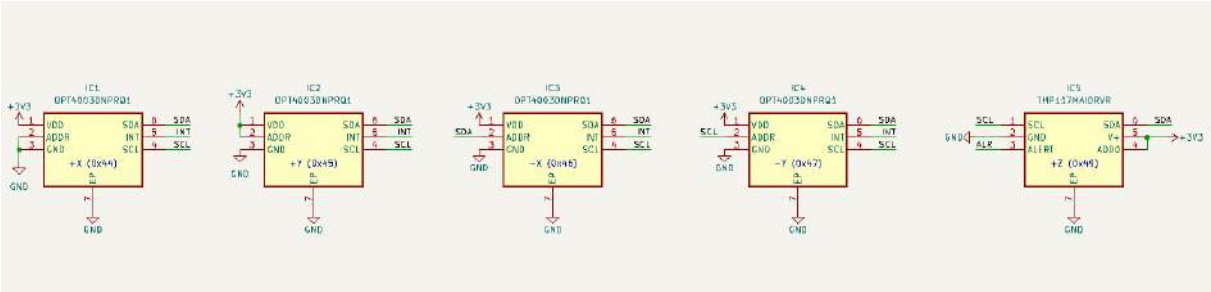
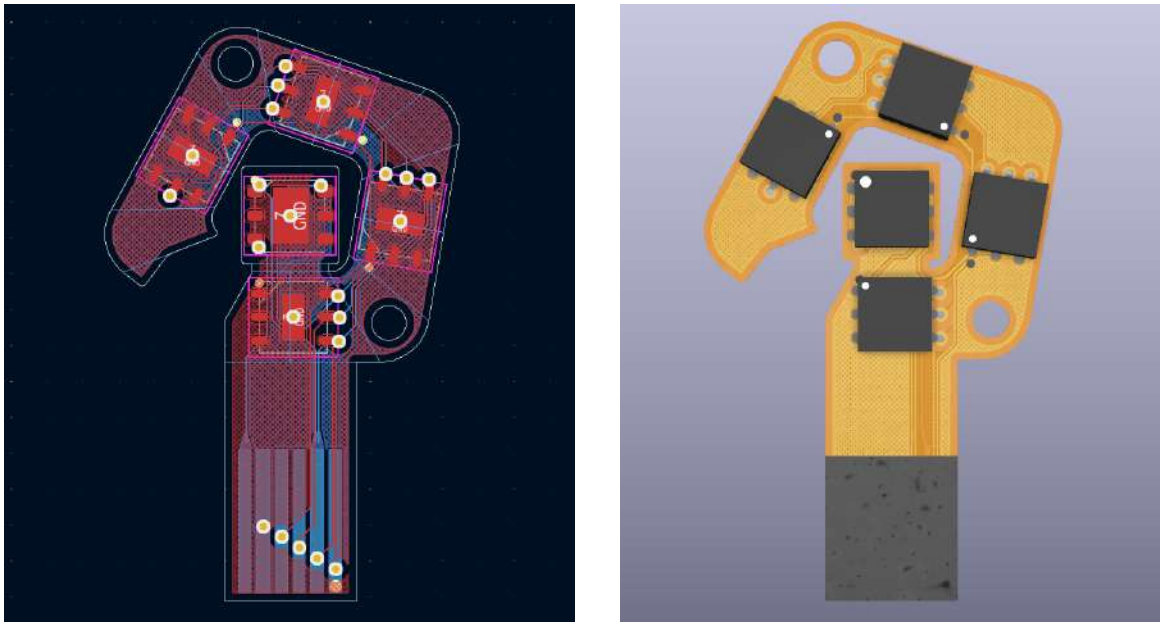


Figure 62: Final prototype circuit schematics.



(a) FPC layout. (b) 3D rendering.

Figure 63: Final prototype flexible printed circuit.

possible dimensions by the manufacturer. For example, while the advanced prototype had 0.2 mm traces, this time, 0.05 mm traces were required to successfully wire all the pads as intended. Additionally, only the first pins of each footprint were identified with small dots, as more elaborate text would be printed outside the cuts. Figure 64 illustrates the constraints imposed by the manufacturer, with the adopted values listed at the bottom for comparison.

On the other hand, the FPC stackup was identical, as were the gold finger dimensions, allowing the CSSPA behaviour to be assessed using the interface PCB of the previous prototype before integrating it into the satellite. Since the pyramid is elevated, the edge connector will flex 90 degrees. To mitigate any damage to the traces, the minimum bend radius was calculated. For a 0.11 mm thickness FPC, the conservative value is 12 times the thickness, resulting in a 1.32 mm radius.

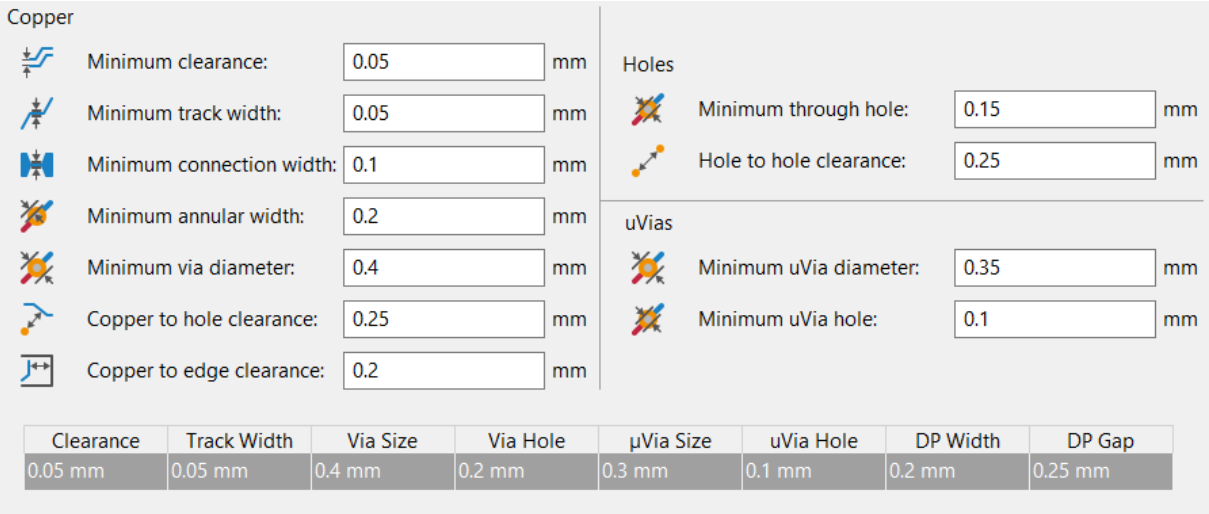


Figure 64: FPC design rules constraints and adopted dimensions.

Despite the copper-to-edge clearance being violated in the temperature sensor zone, the manufacturer replied, ensuring that it would not be an obstacle. Contacting JLCPCB, ten FPCs were quoted at € 65.19, with 18 µm EMI shielding on both sides, as the satellite's solar panels also possess energy harvesting systems that can harm Eletromagnetic Compatibility (EMC). Additionally, 0.05 mm improved tolerances were specified, as demanded by the reduced size.

Combining the total costs of the FPCs, supports, and components, each CSSPA comes at a reduced price of € 40.20, which is an extremely competitive value w.r.t. its performance and compared to other proprietary solutions on the market. It was estimated that each CSSPA weighs between 0.2 and 1 gram. Since the satellite currently weighs 241 grams, all six sensors can be added without exceeding the mass budget. Figure 65 shows a 3D-rendering of the final prototype assembled, on top of a one cent coin, emphasising its tiny size.

Unfortunately, the assembly was not conducted due to time constraints. However, further research was conducted to ensure the proper assembly of the CSSPA. Using a reflow oven, all the transducers shall be soldered using tiny amounts of liquid solder and following the temperature curves displayed in its datasheets. Microscopic inspection should be conducted to verify that all electrical connections are established as desired, so that the interface PCB can be used to assess the sensor's behaviour. Once this step is clear, the CSSPA is permanently fixed onto the pyramid. 3M's DP2216 epoxy was selected for similar reasons to the pyramid's composite: excellent bonding with plastics and metals, resistance to extreme vibration, adequacy for parts subject to thermal cycling, and compliance with NASA's outgassing tests. The epoxy shall be handled and cured w.r.t. to its datasheet, ensuring that each face is fully cured

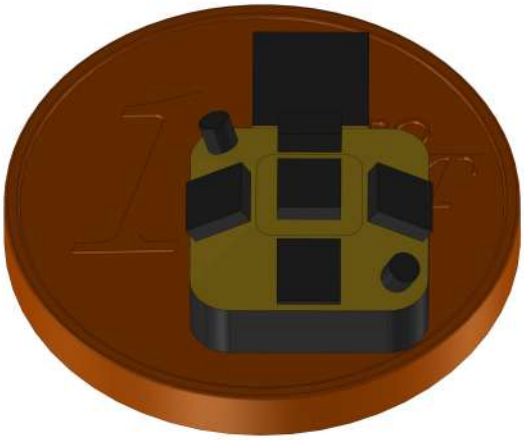


Figure 65: 3D rendering of the final prototype on top of a one cent coin.

before proceeding to the next. After this, each CSSPAs can be glued on the blind slots of the satellite's PCB. Finally, the marker poles can be removed to avoid obstructions of T-FOVs and a pre-flight calibration can be conducted, as described in the previous chapter.

5.3 Satellite modifications

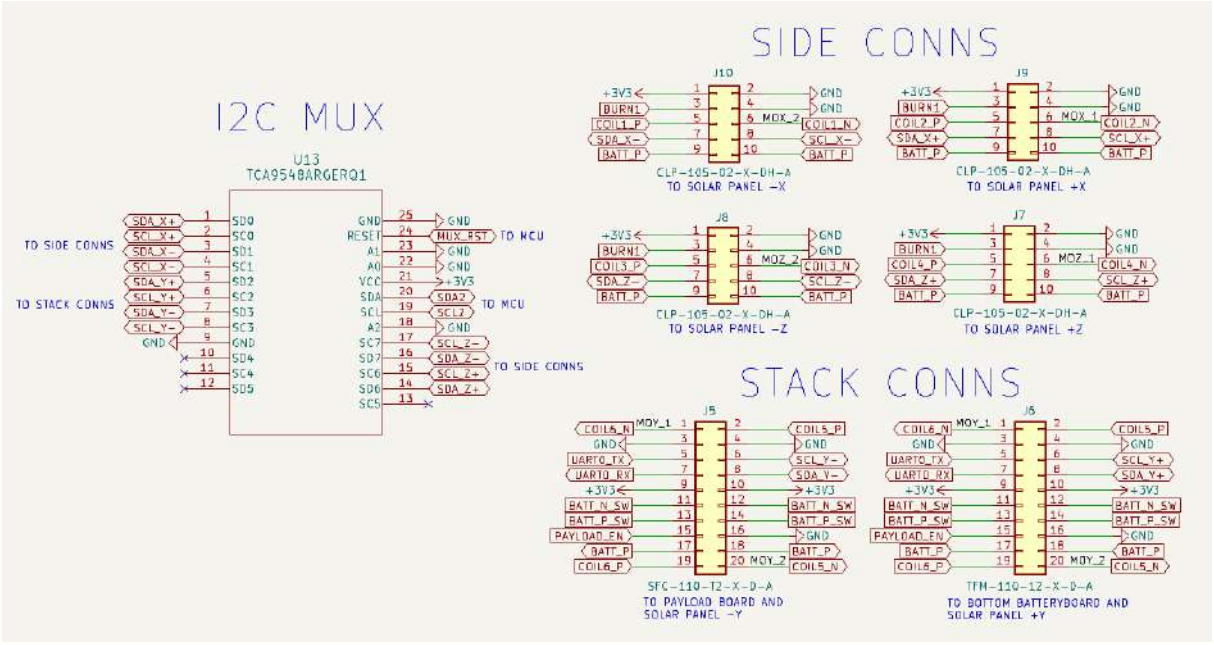
Some modifications must also be made to the satellite to accommodate the CSSPAs integration. Despite the current MCU having multiple connectivity ports, during the PY4 satellite mission, it was found that the ATSAMD51 exhibits a surprisingly high sensitivity to radiation, most likely caused by recent manufacturing changes [78]. In future iterations, it may be replaced by another MCU, which might not have the same connectivity features currently available. To address this concern, an I2C multiplexer approach was adopted, enabling a single I2C master, in this case, the MCU, to communicate with multiple I2C buses, where all the sensors will be connected. To fulfil this purpose, Texas Instruments' TCA9548A-Q1 was chosen, which provides up to eight selectable downstream I2C buses, supporting standard and fast modes up to 400 kHz, thus being compatible with OPT4003-Q1 and TMP117 features. Its low resistance and minimal propagation delay ensure signal integrity, and the availability of CircuitPython drivers simplifies implementation. It also features an AEC-Q100 automotive qualification and selectable I2C addresses (0x70 - 0x77). This component shall be placed in the satellite's main board, where the OBC is hosted and where all other PCBs are interfaced.

Another important aspect regarding the integration is the hardware wiring of the said I2C buses, which must be established from the main board to all solar panels, where each of all six CSSPAs will be mounted.

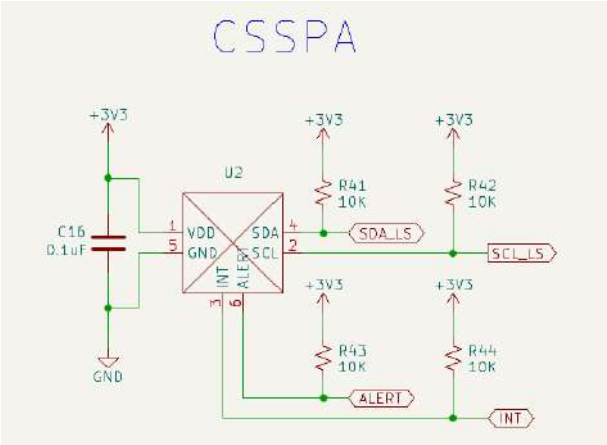
Fortunately, since the satellite already has one OPT4001 in each solar panel, the physical I2C lines are already routed. However, opposite pairs of panels are currently sharing the same bus, which must be logically separated, such as by defining each panel to have its separate bus. Another issue is that there are no pins available on most connectors. This way, the additional ALERT and INT pins cannot be routed; therefore, increasing the connector's pin count should be conducted in future PCB iterations. Note that although the satellite employs vertical stack connectors, these are SMD-type instead of through-hole, meaning that no electrical connection is established directly between the top and bottom layers of the main board, allowing for said readjustment.

The main proposed schematic changes were compiled on the main board and in one of the solar panels, specifically the +Y board. Figure 66a shows the renaming of the I2C buses on the connectors on the main board, matching the same assignment of the multiplexer, while Figure 66b pictures the CSSPA wiring on the solar panel. Furthermore, the footprint location of the multiplexer and the CSSPA were updated in the design, as shown in Figures 67 and 68, which displays the before and after states of each PCB.

On the software side, the required modifications are simple. The new sensors' CircuitPython drivers already available need to be added to the satellite main class. After that, the Sun vector estimation algorithm needs to be updated to perform the LS estimate of all valid measurements, as conducted in the previous chapter. However, this procedure is currently based on the difference of transducer pairs. For now, as the current onboard MEKF implementation takes the Sun vector as input, only the previous routine needs changes. Regarding the temperature part, a simple measurement routine could be implemented for data collection. Later, once the OPT4003-Q1 dependence on temperature is fully characterised, the illuminance measurements could be calibrated before each LS estimation.

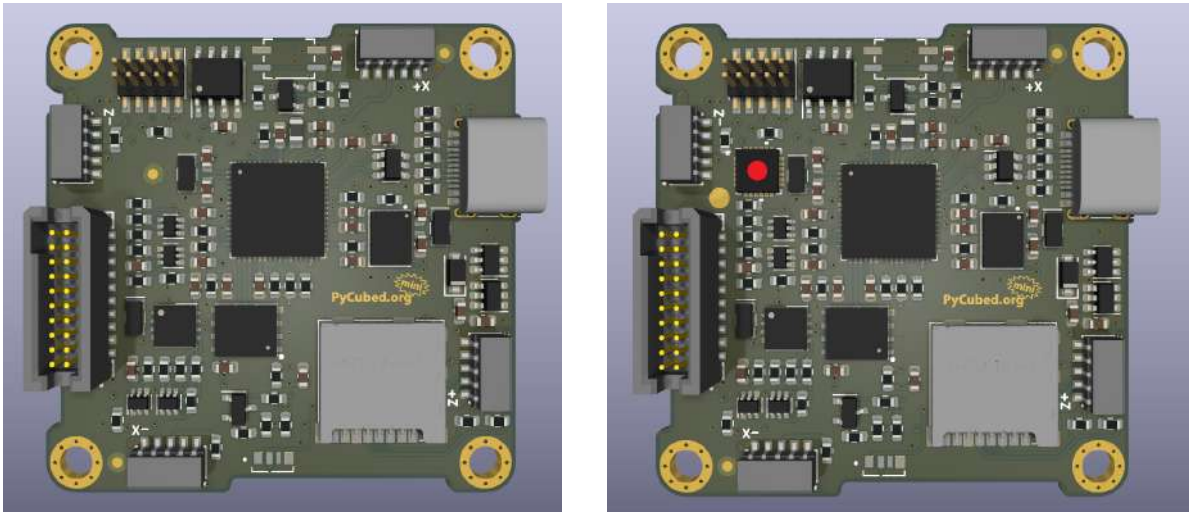


(a) Main board.



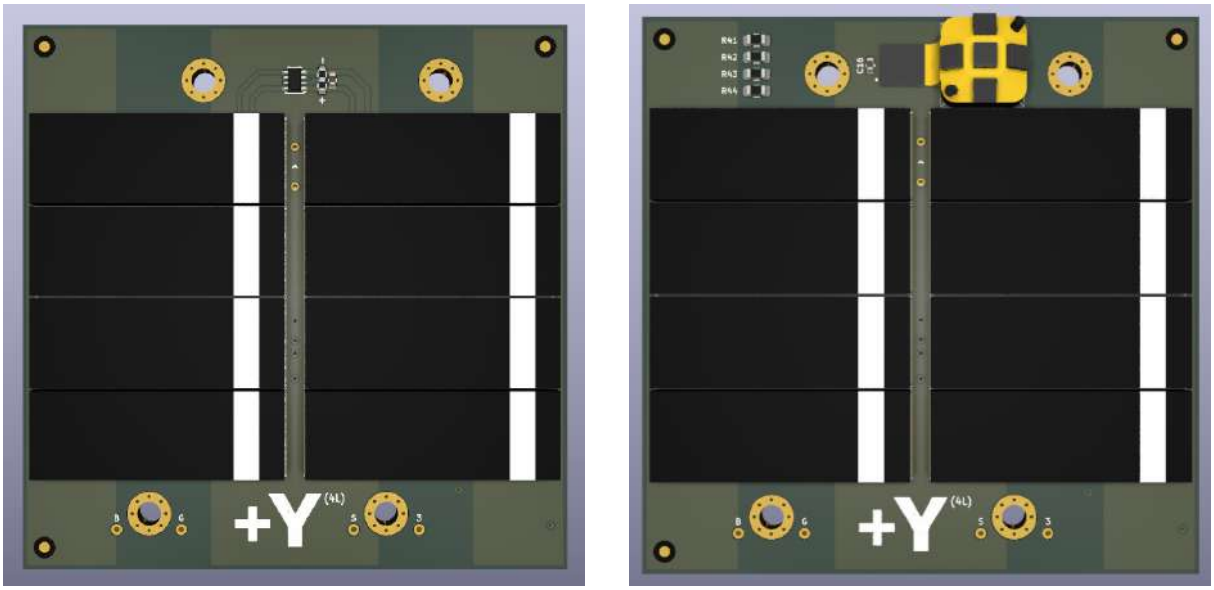
(b) Solar panels.

Figure 66: Proposed schematic changes on PROMETHEUS-1.



(a) Current design. (b) Addition of the multiplexer on the centre-left, in red.

Figure 67: Proposed layout changes on PROMETHEUS-1 main board.



(a) Current design. (b) Addition of the CSSPA on the top.

Figure 68: Proposed layout changes on PROMETHEUS-1 solar boards.

5.4 Summary

The final prototype of the CSSPA addressed all critical integration constraints identified in earlier development stages. A redesigned pyramidal support, manufactured from a space-compliant composite material, was engineered to comply with the PocketQube height envelope while minimising T-FOV obstructions. The FPC was also re-engineered to meet stringent spatial and electrical routing constraints, including EMI shielding and bend radius limitations.

Hardware upgrades included the replacement of the OPT4001 with the enhanced OPT4003-Q1 transducer and the integration of a TMP117 temperature sensor. These changes provide an increased measurement range, a more robust operation, and onboard temperature calibration. A total cost of € 40.20 per CSSPA was obtained, which results in an exceptional price point for the given performance. Provisional circuit schematics and PCB layouts were developed to support electrical integration into the PROMETHEUS-1 satellite.

In terms of integration, both hardware and software adaptations were addressed. Electrically, an I2C multiplexer was proposed to handle multiple buses within the constraints of the satellite architecture, as well as current and future MCUs. Mechanically-wise, sensor placement was optimised on the solar panels, and software modifications were outlined to incorporate new sensor drivers and LS-based Sun vector estimation.

Although only the design and initial validation stages were completed, this chapter fulfils the objective related to the miniaturisation and modularisation of the sensor system while keeping manufacturing costs low for a flight-compatible version of the CSSPA. All necessary steps for physical implementation were covered, and as such, this final prototype represents a viable and well-documented path towards the adoption of the developed CSSPA in future nanosatellite missions.

Chapter 6

Attitude determination software simulation

6.1 Introduction

The software simulation testing is a crucial step in developing and validating the satellite's ADS. By simulating real-world conditions and integrating software models of the sensors, it provides a dynamic and iterative platform for evaluating the performance and reliability of attitude determination algorithms. Software simulation is usually performed as an early validation phase, laying the foundation for Software-In-The-Loop (SITL) and Hardware-In-The-Loop (HITL) testing, providing a cost-effective solution to evaluate the ADS performance.

In the context of this dissertation, the absence of a HITL setup led to a focus on only the simulation part. The software simulation for this project was developed using MATLAB's Simulink, a versatile and widely-used simulation tool that facilitates the modelling and testing of complex systems. Simulink's modular block interface allows for easy integration and modification of system components. It also provides a graphical visualisation of the satellite in orbit, which facilitates grasping the evolution of the simulation. Due to its flexibility, the simulation setup can also be adapted for other similar projects, providing a reusable framework for future developments on the PROMETHEUS satellite platform. It also provides an easy tool to export code to Python, enabling SITL testing on the satellite, which was not performed during the dissertation because of calendar constraints.

The simulation is essentially composed of three main blocks: the space environment, the satellite model and the MEKF implementation, as seen in Figure 69. The ADS would be the latter two.

The subsequent sections of this chapter will provide a comprehensive understanding of each separate block in Sections 6.2 to 6.4, followed by conducted tests on Section 6.5 and obtained results on

Section 6.6. Section 6.7 provides a summary of the chapter's key aspects.

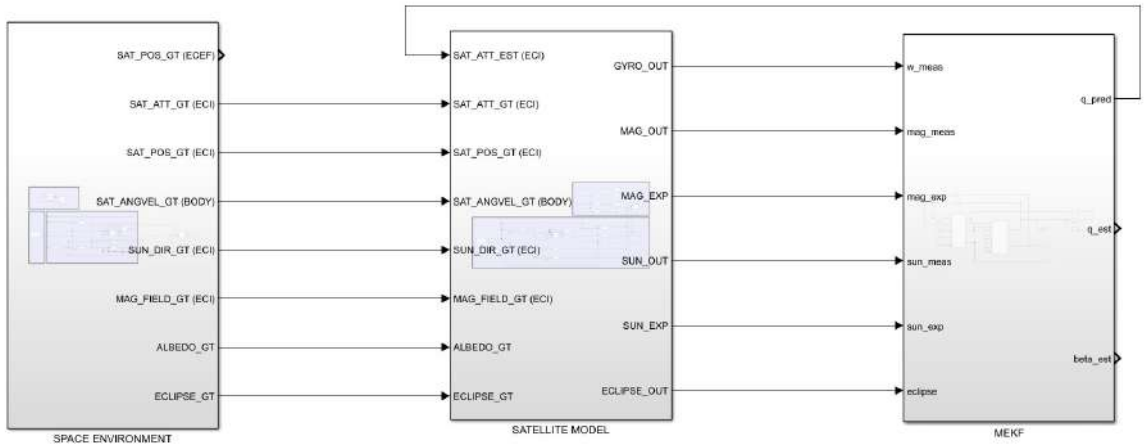


Figure 69: Main blocks of Simulink implementation and its connections.

6.2 Space environment

The first subsystem was responsible for simulating the orbital and attitude dynamics of the satellite, as if it were in space orbiting the Earth, providing the satellite's position, orientation, and linear and angular velocities, all in the respective frame. Furthermore, it simulated environmental behaviour, specifically the Earth's magnetic field flux and albedo reflectance, the Sun's position, and the satellite eclipse. Fundamentally, this block represented the "ground truth" for the simulation, which required being modelled with enough accuracy to replicate the space accurately. Figure 70 shows its implementation.

Regarding the attitude dynamics and kinematics, these were implemented using the quaternion formulation presented in Section 2.1.4, which outputted the satellite's angular velocity and orientation evolution over time, respectively, as shown in Figure 71. The angular velocity propagator had as inputs the applied torque, the body inertia tensor, and initial angular velocity, and outputted the angular velocity at each time instant. This velocity was fed to the attitude propagator, together with the initial attitude, which in turn outputted the orientation.

Considering the orbital dynamics, these were implemented using the Orbit Propagator block from Simulink's Aerospace Toolbox, which was formulated using a Kepler propagator rather than other high-precision numerical methods, such as the available SGP-4. This choice was made to prioritise speed over accuracy, as the orbital behaviour was not within the scope of this dissertation. From the block output, the inertial position of the spacecraft and the UTC time in Julian Date (JD) format were the base inputs for

most of the downstream models.

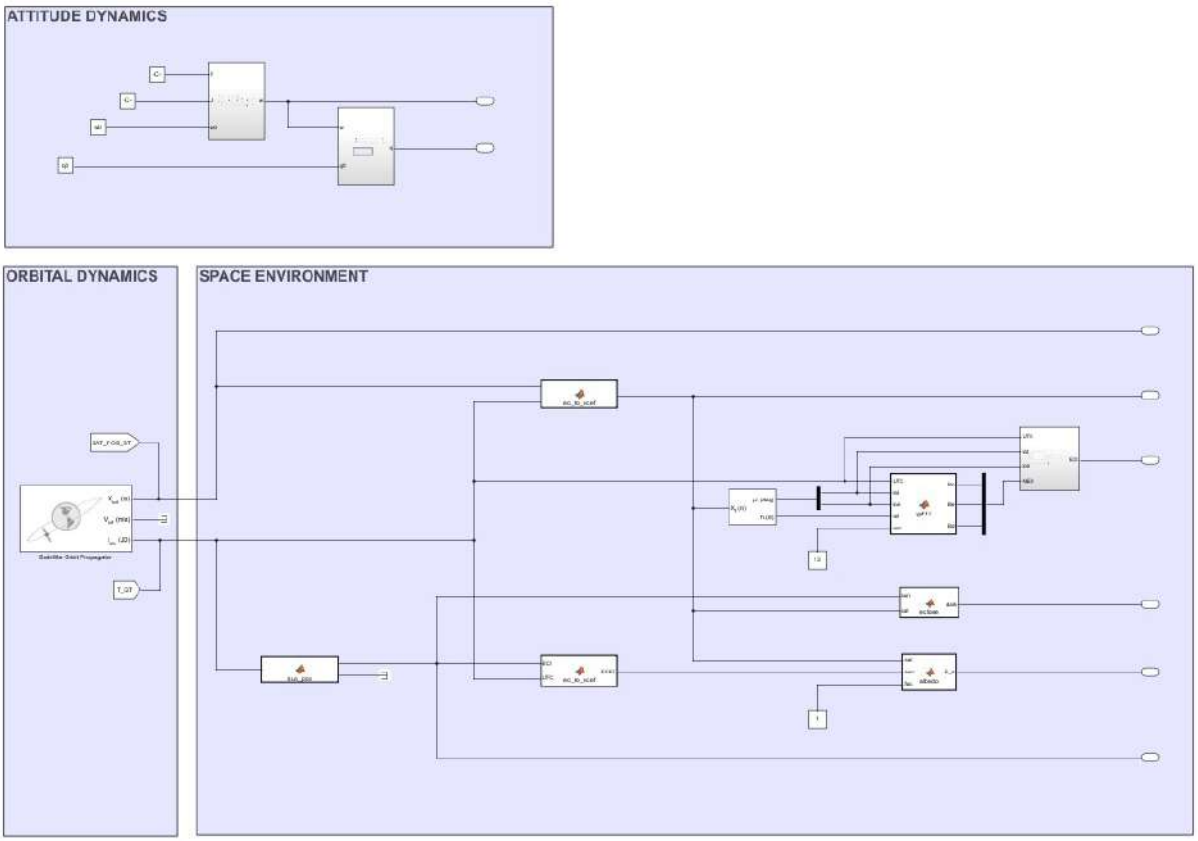


Figure 70: Space environment subsystem.

The remaining area of the subsystem was dedicated to simulating space environment phenomena, such as the Sun's position and magnetic field, both in ECI, as well as Earth's albedo magnitude and eclipse status.

Very precise ephemerides of the Sun w.r.t. the Earth were available. However, a slightly less precise technique was employed to improve speed and simplicity. A method from the Astronomical Almanac [30, 79] was used to determine the Sun vector with an accuracy of 0.01 degrees in ECI, valid up to 2050. The solution rested on determining the ecliptic longitude, ϕ_{eclp}, and obliquity, ε, of Earth relative to its orbit around the Sun, as shown in Figure 73.

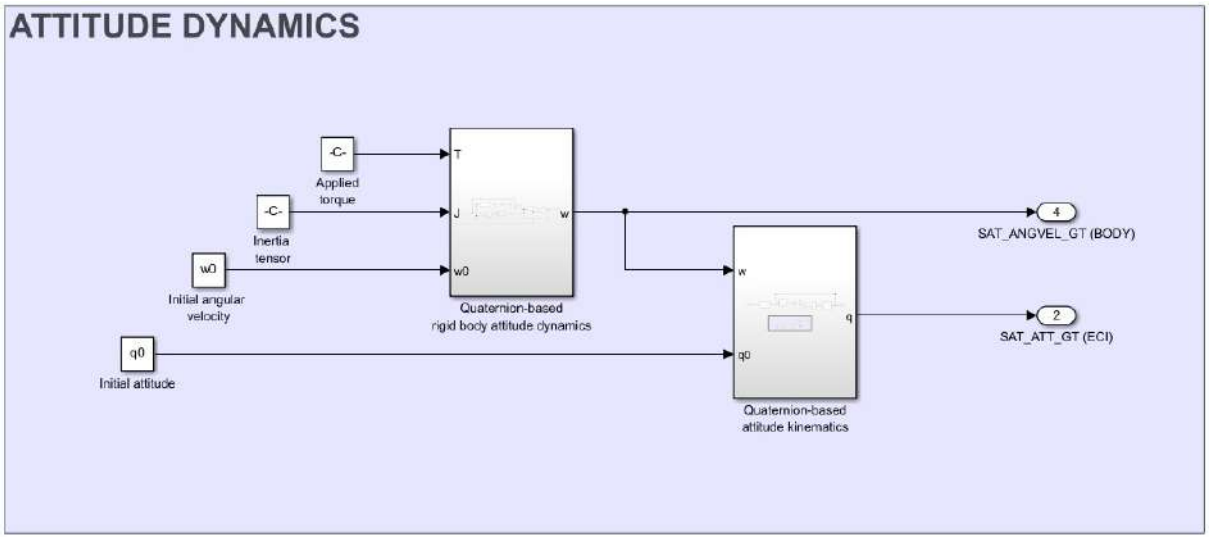


Figure 71: Attitude propagator blocks.

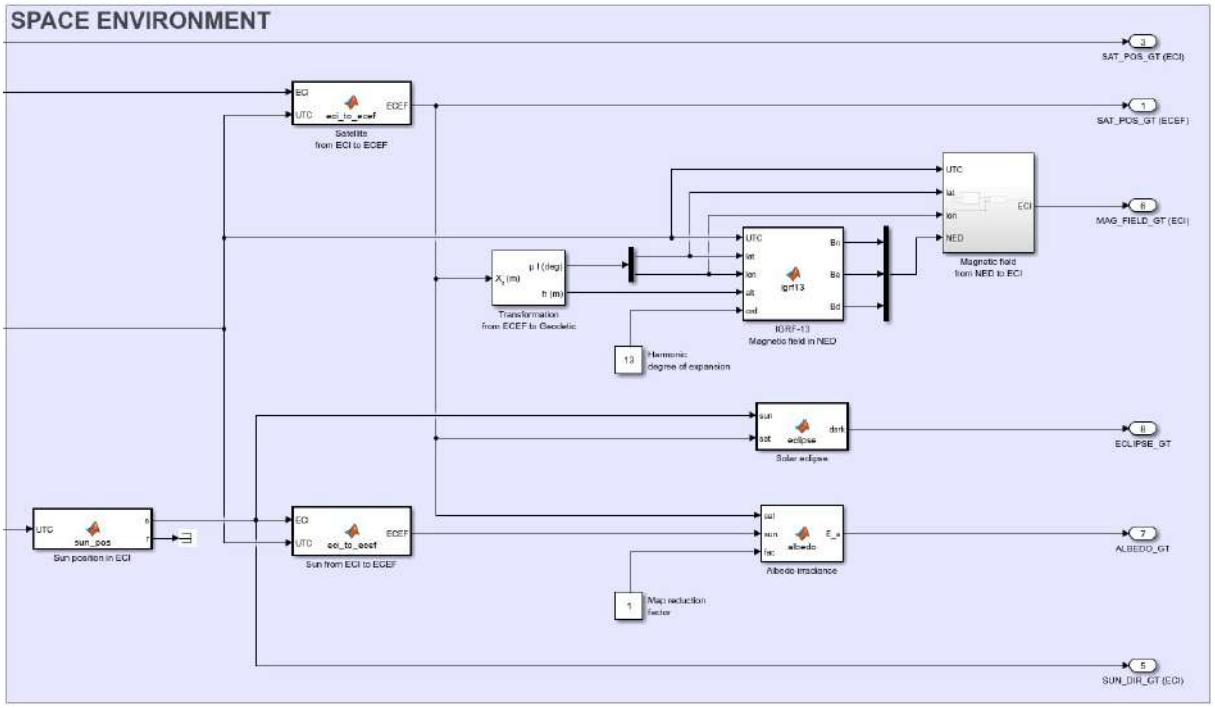


Figure 72: Space environment block.

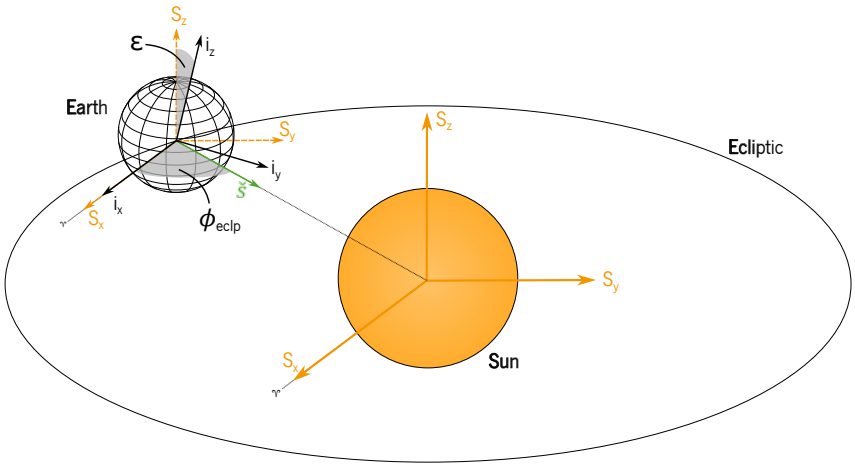


Figure 73: Geometry for the Sun vector determination (adapted from [79]).

Algorithm 9 describes the method used. After determining the centuries elapsed since J2000 epoch in JD UT1 format, T, the mean longitude of the Sun, $\phi_\odot$, was calculated and, approximating the Barycentric Dynamical Time (TDB) time to UT1 time, the mean anomaly of the Sun, $M_\odot$, could be determined [79]. For the same reasons explained in Section 2.1.3, UTC time was employed instead. These two were needed to determine the ecliptic longitude, under the assumption that the true anomaly was close to the longitude of the Sun and the orbit eccentricity was constant and near circular [79], which was generally true. The expression for the obliquity only relied on time and was the only variable left to determine the Sun vector direction $\check{s}$. All the formulas were originally derived in degrees, but they were converted to radians to be consistent with the trigonometric implementations in Simulink. An additional expression was derived for the vector magnitude r but was not necessary for the application. This Sun vector effectively indicated its position relative to Earth. However, since the distance was 300,000 times greater than the distance between the Earth and the satellite, the vector on the satellite was assumed to be parallel.

Algorithm 9 Sun vector direction and magnitude algorithm [30].

$T \leftarrow (T_{UTC} - 2451545.0)/36525$ ▷ Calculate the centuries elapsed

$\phi_\odot \leftarrow \text{rem}\{280.460 + 36000.771T, 360\}\pi/180$ ▷ Calculate the Sun's mean longitude

$M_\odot \leftarrow \text{rem}\{357.5277233 + 35999.05034T, 360\}\pi/180$ ▷ Calculate the Sun's mean anomaly

$\phi_{eclp} \leftarrow [M_\odot + 1.914666471\sin(\phi_\odot) + 0.019994643\sin(2\phi_\odot)]\pi/180$

$\varepsilon \leftarrow (23.439291 - 0.0130042T)\pi/180$ ▷ Calculate the ecliptic's longitude and obliquity

$\check{s} \leftarrow \begin{bmatrix} \cos(\phi_{eclp}) & ; & \cos(\varepsilon)\sin(\phi_{eclp}) & ; & \sin(\varepsilon)\sin(\phi_{eclp}) \end{bmatrix}$

$r \leftarrow 1.000140612 - 0.016708617\cos(M_\odot) - 0.000139589\cos(2M_\odot)$ ▷ Calculate the Sun vector direction and magnitude

return $\check{s}, r$

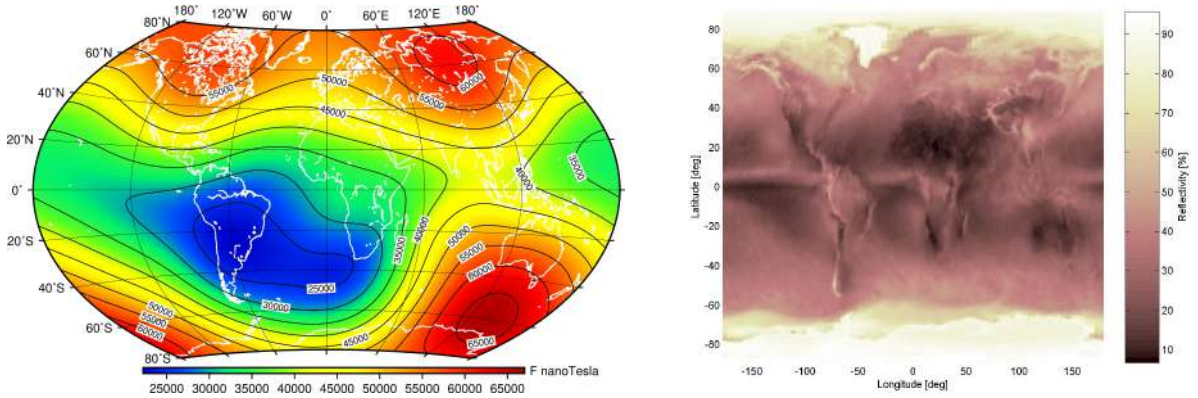
Regarding the magnetic field, the International Geomagnetic Reference Field (IGRF) model was used to determine the magnetic field direction and intensity at each point in orbit. The geomagnetic field is produced by internal sources, primarily inside Earth's core, which changes continuously and unpredictably on timescales ranging from months to millions of years [80]. To account for this, the IGRF is regularly revised, typically every five years.

The geomagnetic field is described as the gradient of a scalar potential, $B = -\nabla V$, and the potential function V is represented as a finite series expansion in terms of spherical harmonic coefficients, also known as the Gauss coefficients, given by

$$V = a \sum_{n=1}^{N} \sum_{m=0}^{n} \left(\frac{a}{r}\right)^{n+1} \left[g_n^m \cos(m\phi) + h_n^m \sin(m\phi)\right] P_n^m \cos\theta \tag{62}$$

where r, θ, ϕ refer to coordinates in a geocentric spherical coordinate system, such as the NED frame described in Section 2.1.3, with r being the radial distance from the centre of the Earth, and θ, ϕ representing geocentric co-latitude and longitude, respectively [80]. A reference radius a was chosen to be the mean Earth radius. The $P_n^m \cos\theta$ terms are Schmidt semi-normalised associated Legendre functions of degree n and order m. The parameter N specifies the maximum spherical harmonic degree of expansion, which is up to 13 in this generation of the model [80].

IGRF-13 provides the definitive Gauss coefficients for epochs prior to 2015, the estimated coefficients for the 2020 epoch, and a predictive secular variation for these coefficients from 2020 to 2025 [80]. The employed algorithm was a conversion from the original National Oceanic and Atmospheric Administration (NOAA)'s FORTRAN routines to MATLAB, made by W. Brown [81], which was approximately 2.5 times faster, on average, compared to the one provided by the Aerospace Toolbox set, without compromising accuracy.



(a) Magnetic field intensity of IGRF-13 model (retrieved from [82]).

(b) Annual mean reflectivity of TOMS 2001 data (retrieved from [33]).

Figure 74: Different space environment models selected for implementation.

Next, the Earth's albedo model estimated the irradiance reflected from Earth that reached the spacecraft. Developed by Bhanderi [33], the model of the Earth albedo is based on the reflectivity data measured by the Earth Probe Satellite of the Total Ozone Mapping Spectrometer (TOMS) project. This data is given in a 1-degree latitude by a 1.25-degree longitude surface grid around the globe's surface [33]. The selected data pack was collected throughout 2004 and then averaged.

As discussed in previous chapters, the incident irradiance on each grid cell is equal to the solar AM0 irradiance ε_{AM0} multiplied by a cosine term dependent on the incident angle α_{in}, in this scenario, the angle between the cell normal n_c and the Sun LOS vector from the grid point s_c [33]. Additionally, the amount of energy incident on a cell at each grid point depends on the cell area A_c, which is calculated using the Surface of Revolutions expression [33]. The reflected radiant flux density by each cell P_c is the incident radiant flux density P_r multiplied by the reflectivity coefficient ρ, assuming the Earth's surface is Lambertian [33].

The amount of Earth's albedo irradiance from a single cell, seen from the satellite, ε_c, depends on: the distance to the satellite r_{sc}; the angle between the cell normal and the satellite LOS vector from the grid point, α_{out}; and ultimately, the reflected irradiance [33]. The latter is given by the reflected radiant flux density P_c divided by π [33]. The Inverse Square Law states that the intensity of this irradiance decreases with the square of the distance from the grid point to the satellite [33]. Figure 75 shows the geometric construction of the problem.

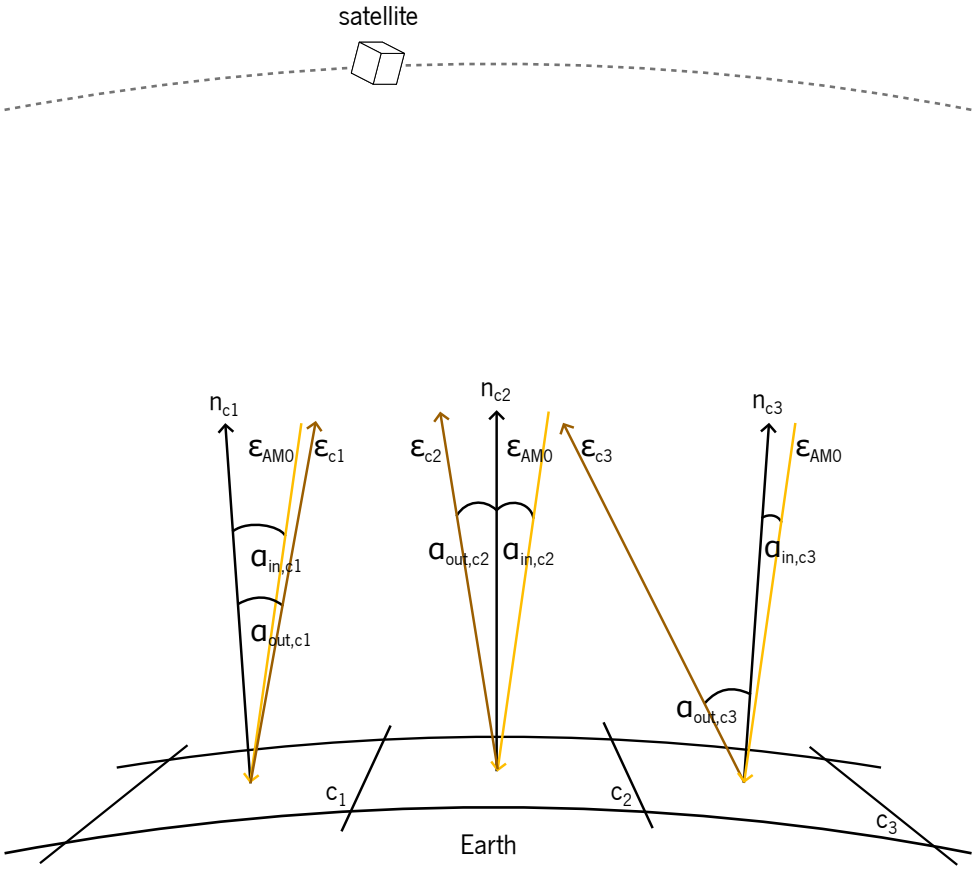


Figure 75: Geometry for the Earth's albedo determination (adapted from [33]).

Obviously, from these 51,840 grid points, only those who were sunlit and simultaneously visible by the satellite mattered. Therefore, the Earth's total albedo ε_a was the sum of all these small pieces of irradiance coming from each cell [33]. Algorithm 10 represents all the required steps to calculate the albedo magnitude. However, since the mentioned grid cells could still be thousands, it was assumed that this magnitude reached the satellite in the anti-parallel direction to the Earth, which simplified the Sun sensor measurement equation [33]. Similarly to the IGRF model, the coefficients were saved in a separate file and loaded the first time the algorithm ran. The Earth Albedo Toolbox, also developed by Bhanderi, was already available in Simulink; however, an efficiency-driven reimplementation was conducted, resulting in an average computation speed of around 4.4 times faster.

Algorithm 10 Earth's total albedo model from TOMS reflectivity data [33].

$\varepsilon_a \leftarrow 0$ ▷ Reset albedo irradiance
for all grid cells c **do**
  $\cos\alpha_{in} \leftarrow \check{s}_c \cdot \check{n}_c$ ▷ Calculate Sun incidence angle cosine on the cell
  $\cos\alpha_{out} \leftarrow \check{r}_{sc} \cdot \check{n}_c$ ▷ Calculate the angle cosine of the cell normal and the satellite
  if $\cos\alpha_{in} > 0$ **and** $\cos\alpha_{out} > 0$ **then** ▷ Verify if both angles are between 0 and 90 degrees
    $A_c \leftarrow \theta_c R_E^2 \left[\cos(\phi_c - \Delta\phi/2) - \cos(\phi_c + \Delta\phi/2)\right]$ ▷ Calculate the cell area
    $P_c \leftarrow \varepsilon_{AM0} A_c \cos\alpha_{in}$ ▷ Calculate the incident radiant flux density
    $P_r \leftarrow \rho P_c$ ▷ Calculate the reflected radiant flux density
    $\varepsilon_c \leftarrow P_r \cos\alpha_{out} / (\pi \|r_{sc}\|^2)$ ▷ Calculate the cell reflected irradiance reaching the satellite
    $\varepsilon_a \leftarrow \varepsilon_a + \varepsilon_c$ ▷ Sum to the total albedo irradiance
  end if
end for
return ε_a

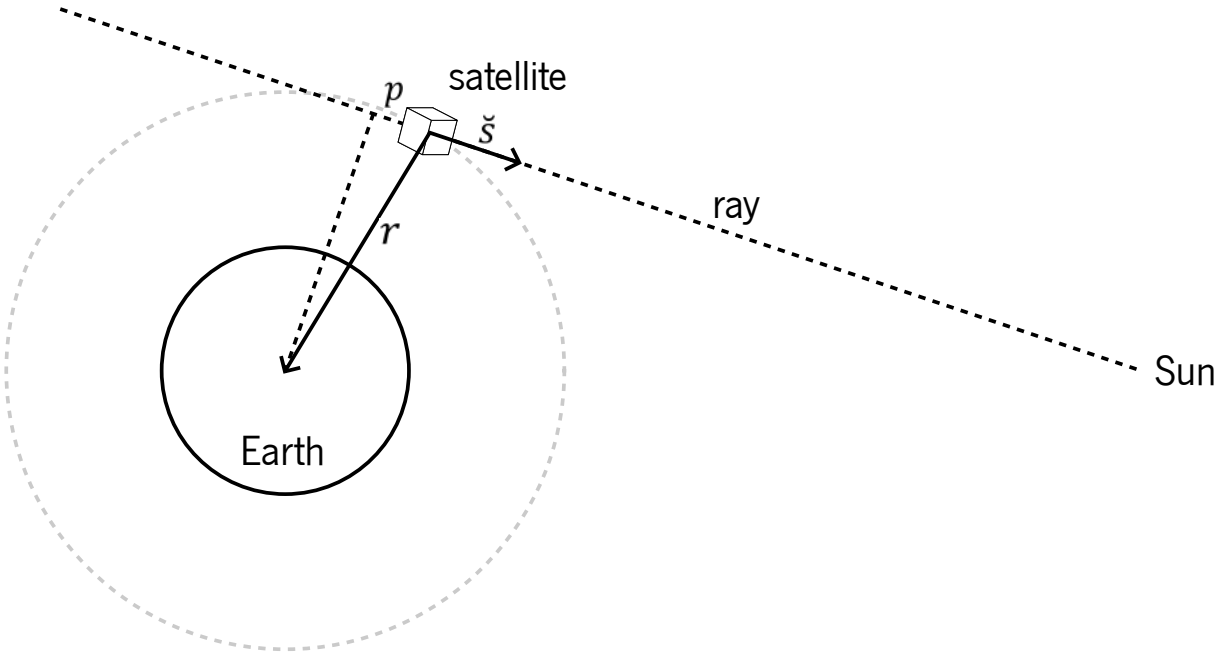
Lastly, the eclipse algorithm was based on the ray-sphere intersection problem [83]; therefore, this model assumes the eclipse to be cylindrical instead of a dual cone. Given a ray, defined by a unit vector originating from a point, and a sphere, the problem consists of determining whether they intersect or not. In this scenario, the ray was represented by the unit Sun vector $\check{s}$ with its origin on the spacecraft, and the sphere represented the Earth, with r as a vector pointing to its centre from the satellite.

With this in mind, the spacecraft was not eclipsed in one of two scenarios [83]: if the sphere was behind the ray origin, therefore it could never block the ray, shown in Figure 76a; or if the sphere was in front, but there was still an unobstructed path, depicted in Figure 76b.

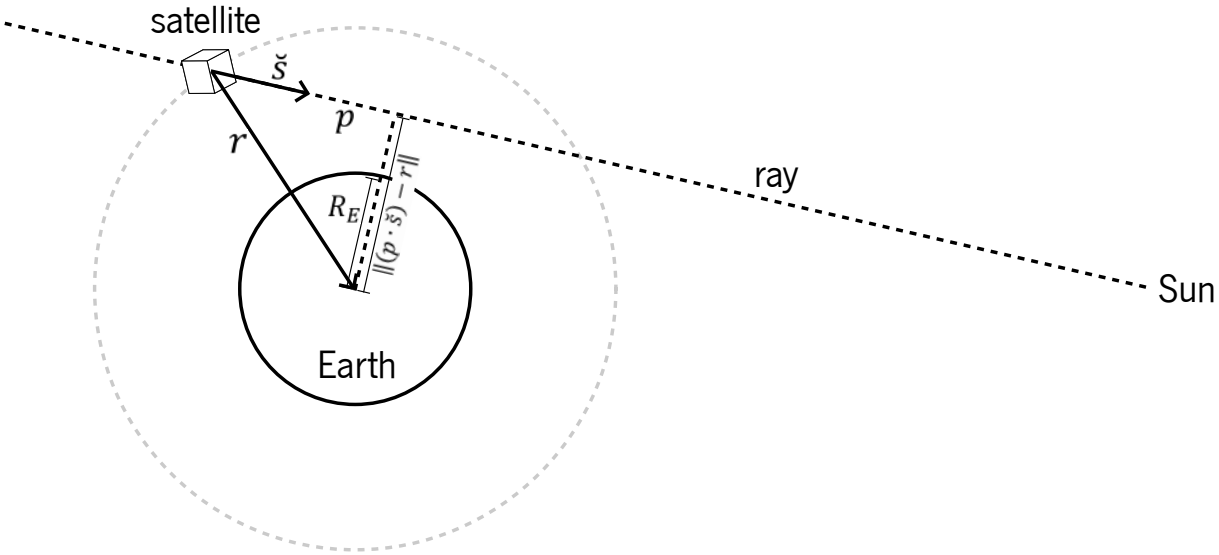
The dot product between the ray direction and the sphere direction indicated whether the sphere laid ahead or behind. Then, if it was ahead, the distance between the sphere centre and its projection into the ray could be determined. If this distance was higher than the sphere radius, in this case, the Earth's mean radius R_E, there was no intersection [83]. The binary flag "dark", represented with d, outputs the result of Algorithm 11.

Algorithm 11 Eclipse determination by the ray-sphere intersection problem [83].

$p \leftarrow \check{s} \cdot r$ ▷ Calculate the direction of the sphere relative to the ray
if $p \leq 0$ **or** $\|(p \cdot \check{s}) - r\| > R_E$ **then** ▷ Verify if any of the no-eclipse scenarios occur
  $d \leftarrow 0$ ▷ Eclipse is false
else
  $d \leftarrow 1$ ▷ Eclipse is true
end if
return d



(a) Sphere is behind the ray origin.



(b) Sphere is in front of the ray, but there is an unobstructed path.

Figure 76: 2D scenarios where the satellite is not eclipsed (adapted from [83]).

6.3 Satellite model

The second block was responsible for simulating the satellite's operational behaviour in orbit. This block was further divided into two main parts: the measurements obtained by the onboard sensors, which need to be simulated; and the expected measurements obtained by the implemented models, as shown in Figure 77. The latter models are the very similar to the ones described in the previous section, as their purpose is very similar: to provide a reference in this simulation and the onboard filter.

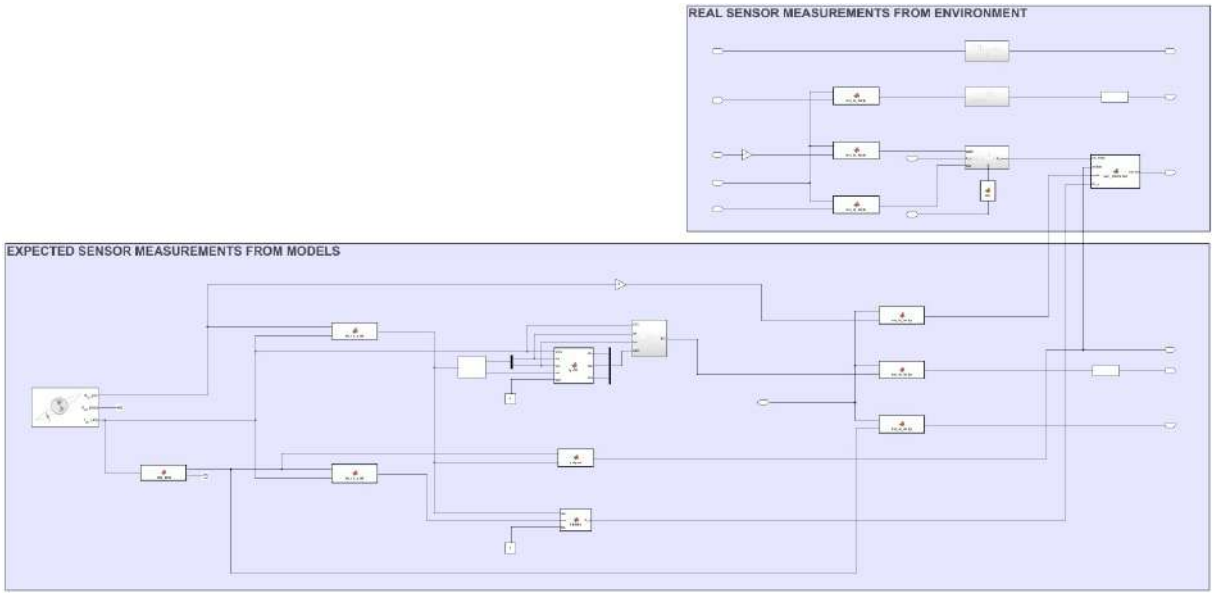


Figure 77: Satellite model subsystem.

As explained previously, PROMETHEUS-1 has three types of useful available sensors: a rate-integrating gyroscope, which will be used to propagate the MEKF state, as well as a magnetometer and Sun sensors, which will update it. The magnetometer measures the local magnetic field flux $\tilde{B}$, and the Sun sensors, dispersed around the spacecraft, measure light illuminance $\tilde{E}$ used to estimate the Sun vector $\tilde{s}$. Therefore, hardware models were formulated for these components, w.r.t. the sensor specifications such as noise, saturation and sampling, for example. The gyro and magnetometer frame were assumed to be the same as the body frame, and the six Sun sensors were oriented as described in the previous chapter.

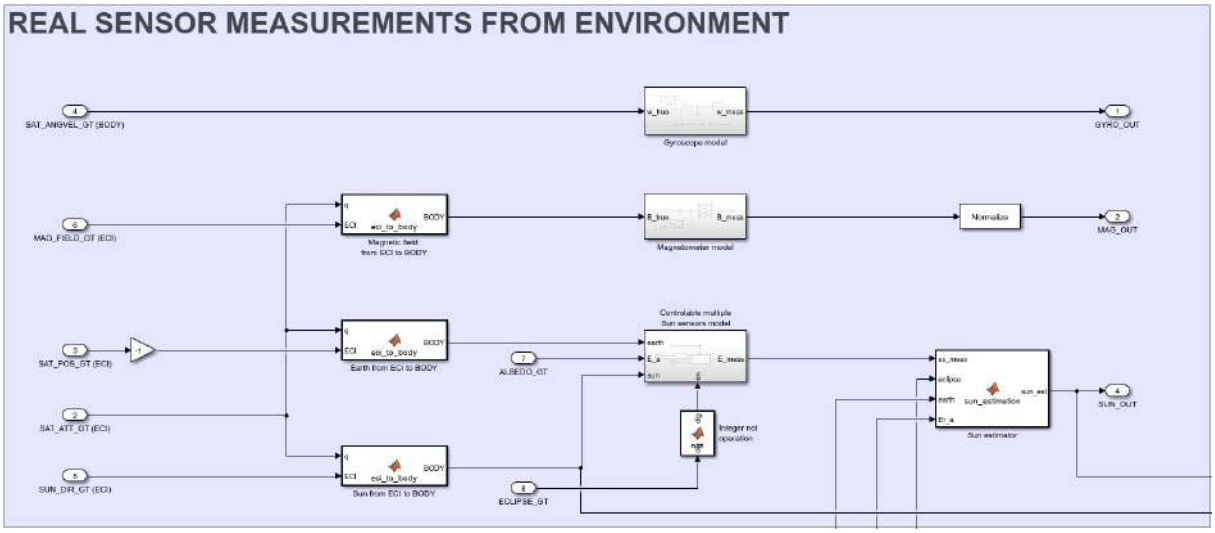


Figure 78: Simulated hardware measurements block.

Gyro measurements are compromised by various types of noise, which can be that could be separated into five categories [84]:

Scale factor error Proportionality constant between the input physical quantity and the sensor output;

Misalignment error Sensing axes that are not perfectly orthogonal or are not perfectly aligned with the defined sensor frame;

Static bias Sonstant offset in the measurement, independent of time or motion;

Rate Random Walk bias Low-frequency random drift in the rate measurements, commonly modelled as a random walk process, reflecting slowly varying biases due to factors such as temperature changes or ageing;

Angle Random Walk bias High-frequency stochastic fluctuations in sensor output due to inherent electrical and mechanical noise, often modelled as white Gaussian noise.

The first two are very trivial to calibrate, but to analyse and quantify the latter three, the Allan variance is applied, a time-domain statistical technique widely adopted as a standard for characterising inertial sensor noise [85, 86]. The Allan variance is computed by segmenting a time series of sensor data into clusters of length, calculating the mean of each cluster, and then computing the variance of the differences between adjacent cluster means. Each noise source has a different correlation time [85]. By choosing the correct correlation time or cluster length, the desired noise source variance can be calculated. Lam et al. [84] thoroughly explore this topic.

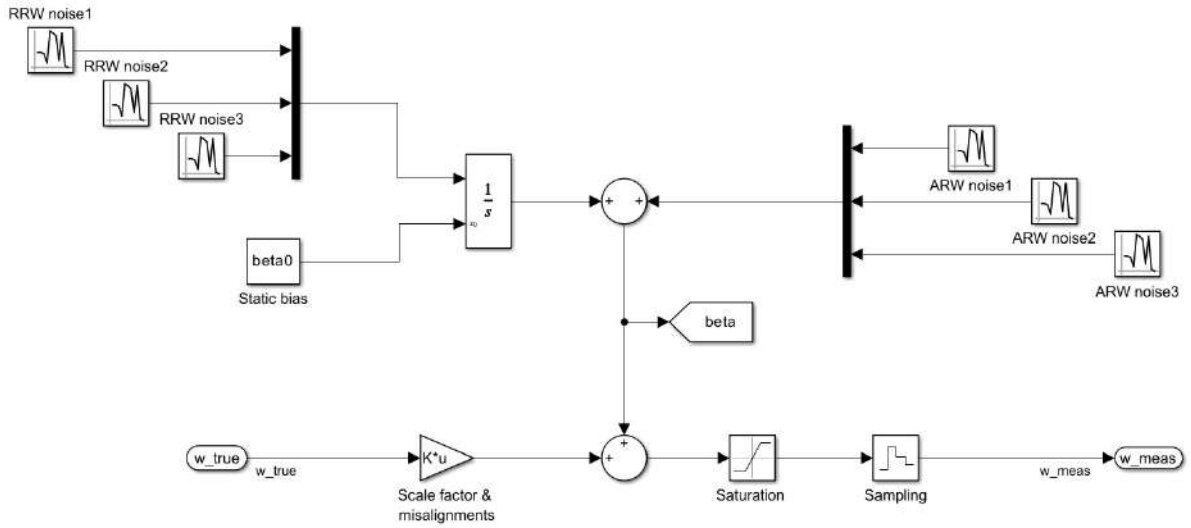


Figure 79: Gyroscope hardware model.

Angle Random Walk (ARW) bias, as η_v, and Rate Random Walk (RRW) bias, as η_u, are zero-mean Gaussian white noises, the latter integrated over time. Furthermore, the static bias β_ω is obviously constant, as explained in Sections 2.2.3 and 6.4. The measurement model was adapted from Equation (53) to a single expression given by

$$\tilde{\omega} = M_\omega \omega + \beta_\omega + \eta_v + \sum \eta_u \Delta t \tag{63}$$

Each axis had a separate block with a different random seed to make the generated values independent. Scale factors and misalignments formed a 3x3 matrix M_ω that multiplied the readings. The sensor's saturation introduced a generic quantisation error, which could be later extended to include other types, such as sensitivity, data type conversion, and scale conversion [87]. Lastly, a zero-order hold block was added to simulate the sensor's sampling frequency [85]. Figure 79 shows the implemented gyro model.

For the employed BMG160 gyro, embedded in the BNO085 9-axis Inertial Measurement Unit (IMU), all biases are further discussed and presented in Sections 6.4 and 6.5. The sensor saturates at ±2000 degrees per second, and its sampling frequency was set to match the simulation execution. No scale factor and misalignment errors were assumed for the time being.

The true magnetic field flux density vector was subject to scale factor and alignment errors, represented by matrix M_B. It was corrupted by a zero-mean Gaussian white noise process with a given standard deviation and by a static bias, defined as η_B and β_B, respectively [40]. Equation (65) gives the measurement function of the sensor.

$$\tilde{B} = M_B B + \beta_B + \eta_B \tag{64}$$

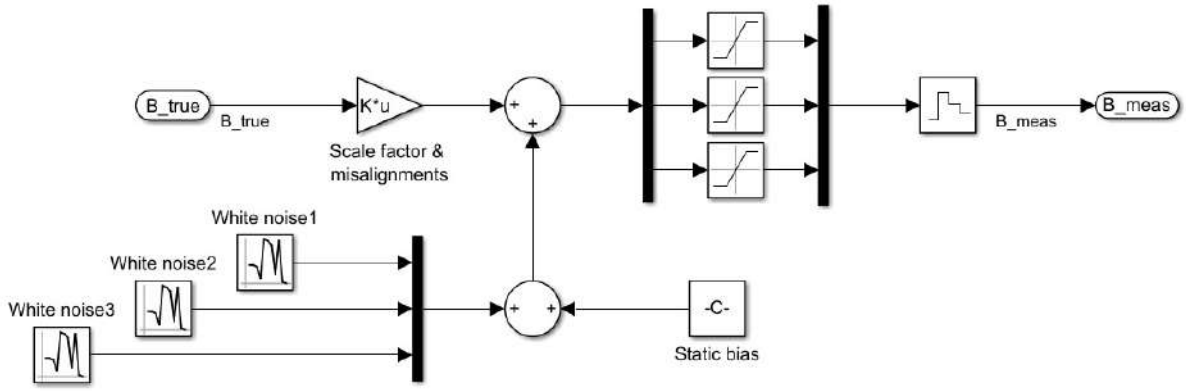


Figure 80: Magnetometer hardware model.

In this case, the saturation block was separated by axes to comply with the sensor specifications. The BMM150, also embedded in the BNO085, limits measurements to ± 1300 µT in the X and Y axes and extends the range to ± 2000 µT in the Z axis. In its optimal mode, it has a noise standard deviation of 300 nT. The static bias, scale factor and misalignment errors were not considered at the time.

Section 2.2.2 presented some measurement models for the photoelectric transducers. However, they had not yet incorporated albedo interference, as pre-flight calibration typically does not consider it. The extended measurement model [40] was then reformulated as

$$\tilde{E}_j = C_j \left[\check{n}^T \check{s} + \frac{\varepsilon_{aj}}{\varepsilon_{AM0}} (\check{n}^T \check{r}) \right] + \eta_E \tag{65}$$

where $\tilde{E}$ is the measured illuminance, ε_a and ε_{AM0} are the albedo and Sun irradiances, respectively. $C = E_{max}\varepsilon_{AM0}/\varepsilon_{cal}$ is the sensor scale factor, effectively calibrating and converting irradiance to illuminance, $\check{n}$ represents the sensor's normal vector, eventually including misalignments errors, $\check{s}$ and $\check{r}$ are the Sun and Earth directions, respectively. Lastly, η_E is the usual zero-mean Gaussian noise with a standard deviation of 67 lx, as calculated from OPT4001's datasheet. Since there are multiple Sun sensors on the satellite, identified with the subscript j, the device only needed to be modelled once. Then, measurements were taken iteratively by modifying the normal vector and scale factor to the respective one. The zero-hold blocks placed in the middle of the model of Figure 81 were required to ensure that the cosine values were positive or, in other words, the light was coming from within the T-FOV.

The algorithm to determine the measured Sun vector was based on the LS estimator of Equation (44), with a threshold to exclude measurements below 1% of the maximum illuminance, aiming to remove low SNR values in a simply manner. Two key aspects must be highlighted regarding this specific implementation. First, all transducers were treated as independent, instead of each CSSPA estimating the Sun vector. This approach, as explained in Section 2.2.2, increased the number of measurements on the LS estimation

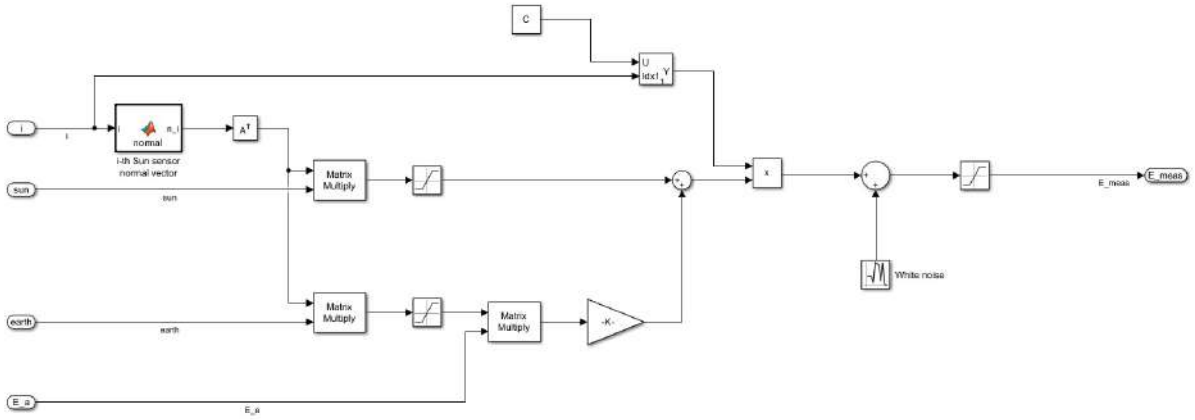


Figure 81: Sun sensor hardware model.

process, effectively reducing its error. Secondly, the obtained vector from the algorithm was influenced by both Sun irradiance ε_{AM0} with direction $\check{s}$, and albedo reflectance ε_a with direction $\check{a}$. This light unit vector $\check{l}$ was approximately given by the vector sum $l = \check{s}\varepsilon_{AM0} + \check{a}\varepsilon_a$ [33], as depicted in Figure 82. Rearranging the expression, the measured Sun vector direction was therefore approximately given by

$$\check{\check{s}} = \frac{\check{l}\sqrt{\varepsilon_{AM0}^2 + \varepsilon_a^2} - \check{a}\varepsilon_a}{\varepsilon_{AM0}} \tag{66}$$

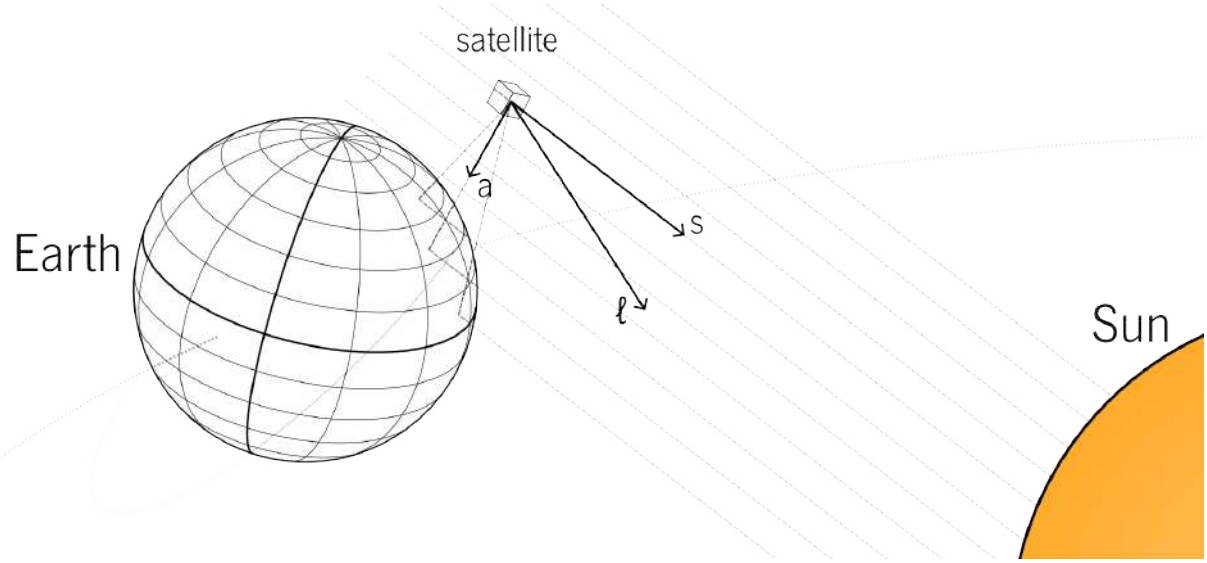


Figure 82: Light vector composition.

Although the calibration of the previous chapter provided a useful and valuable procedure to characterise the performance of the developed Sun sensors, the absence of an AM0 light source proved to be an obstacle to using the calibrated scale factor as the initial value because the emission spectrum was different, and the incident illuminance was only about 5% of the maximum expected. Several methods

can be employed in the future to overcome this without the expensive setup availability. However, for this simulation, a theoretical value of $E_{max} = 133,531$ lx was defined for an irradiance of ε_{AM0} reaching the OPT4001 transducer, as derived in Appendix A1, together with a starting calibration factor $\varepsilon_{cal} = 1$. Each transducer's normal vector was initially set to be perfectly oriented, but slight deviations could be added in the future to evaluate performance.

As explained earlier, specifically in Section 2.2.3, during the MEKF update step, the measurements taken onboard were compared to their expected values to estimate how the attitude evolved over time. The measured values were obtained through the above implementation, which aimed to replicate the sensor's behaviour in orbit. In parallel, as mentioned at the beginning, the expected measurements are essentially provided by the same models in Section 6.2. However, some modifications were made, primarily concerning speed and memory usage, as they will be running on a resource-constrained device.

The magnetometer measurement function requires the satellite's position and the current time to predict the magnetic field vector in the inertial frame, as seen above. The strength of the spherical harmonic of the n-th order decreases with the inverse (n+1)-th power of the distance from its center [32]. So, higher-order harmonics can sometimes be omitted in order to reduce computation time at the expense of accuracy [32]. For the onboard model, it was selected to include harmonics up to degree four, which resulted in a Mean Absolute Percentage Error (MAPE) of around 2% at satellite altitude and reduced computation time by 70%, compared to the full model. Figure 83 shows the error distribution, which can reach up to 10% in some regions. Calculating the model terms recursively, besides improving memory usage, also reduced computation time by 7% [32]. The expected magnetic field vector needed to be rotated to the body frame using a rotation matrix parameterised with the last predicted quaternion, following Equation (22).

Regarding the Sun sensors, the Sun vector in the inertial frame only requires the current time, as seen previously. The satellite's and Sun's positions were used to calculate the eclipse, which determined whether the filter should consider or discard the Sun measurement update. These two models remained equal as described in Section 6.2.

For the onboard model of albedo, due to the high computational load, the TOMS reflectivity data resolution was reduced by a factor of four, as recommended by the model author [33], effectively increasing the data angular resolution to 4x5 degrees. This reduction was achieved by applying a low-pass Finite Impulse Response filter and linear interpolation to the data. While the RMSE increased negligibly, it reduced computation time by 93% [33]. The expected Sun vector was also rotated to the body frame.

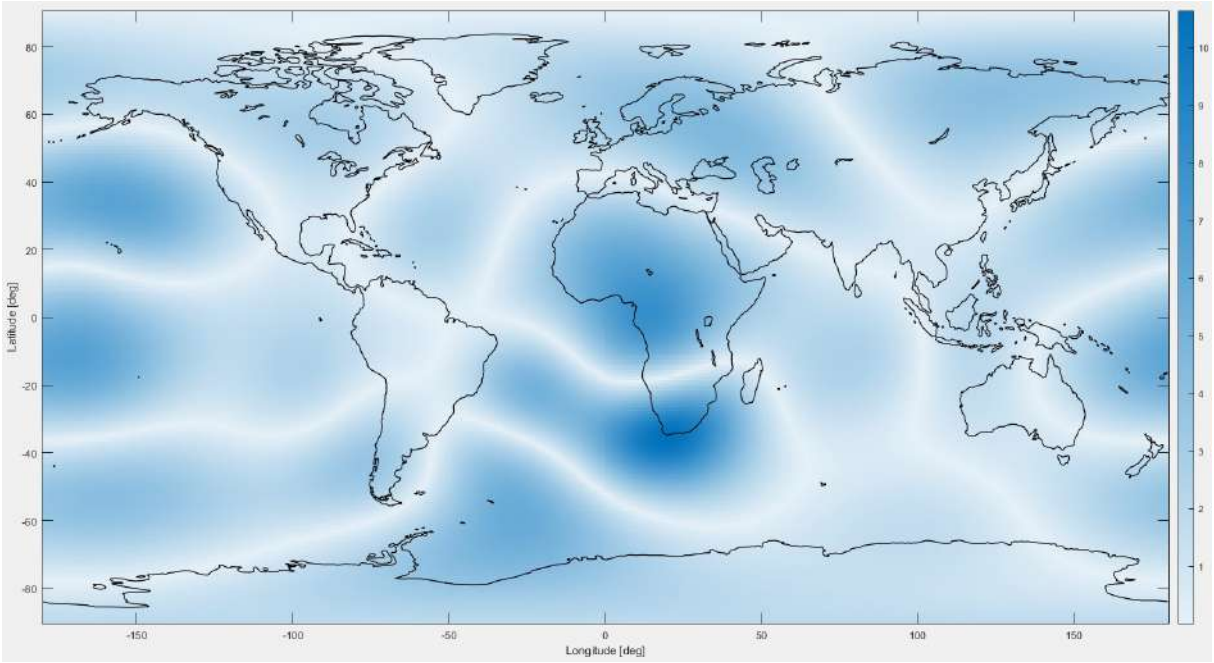


Figure 83: Relative error distribution of the magnetic field intensity with a four-degree IGRF.

Due to the absence of a GNSS receiver on PROMETHEUS-1, the satellite's position was calculated using an orbital propagator, such as the one described in Section 6.2, which could be continuously updated with the most recent Two-Line Element set (TLE) when in orbit. In that scenario, the current time is tracked by the Real Time Clock (RTC) but can also be updated via a telemetry command. In the simulation, these variables were obtained from an Orbit Propagator block. Figure 84 shows the final result.

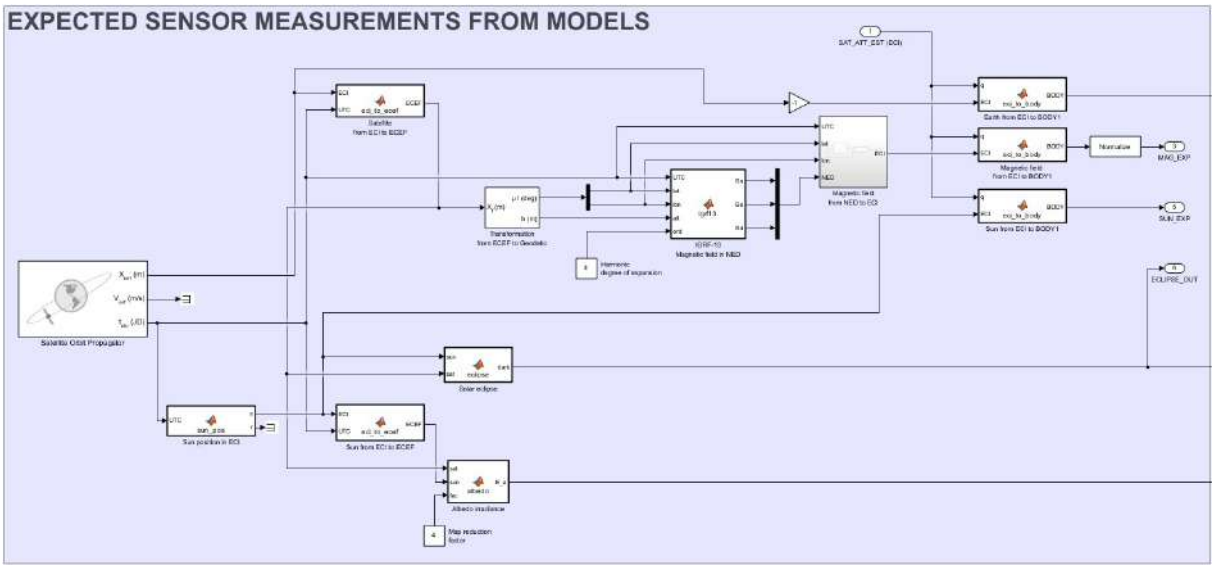


Figure 84: Expected sensor measurements from constrained models.

6.4 MEKF implementation

The filter implementation followed Algorithm 4, and all that remained was to define the measurement residuals, which were determined by the pair of measurements and expected values provided by the previous block, along with the covariance matrices specific to this application. The simulation blocks were divided into two main steps: prediction and correction, as pictured in Figure 85.

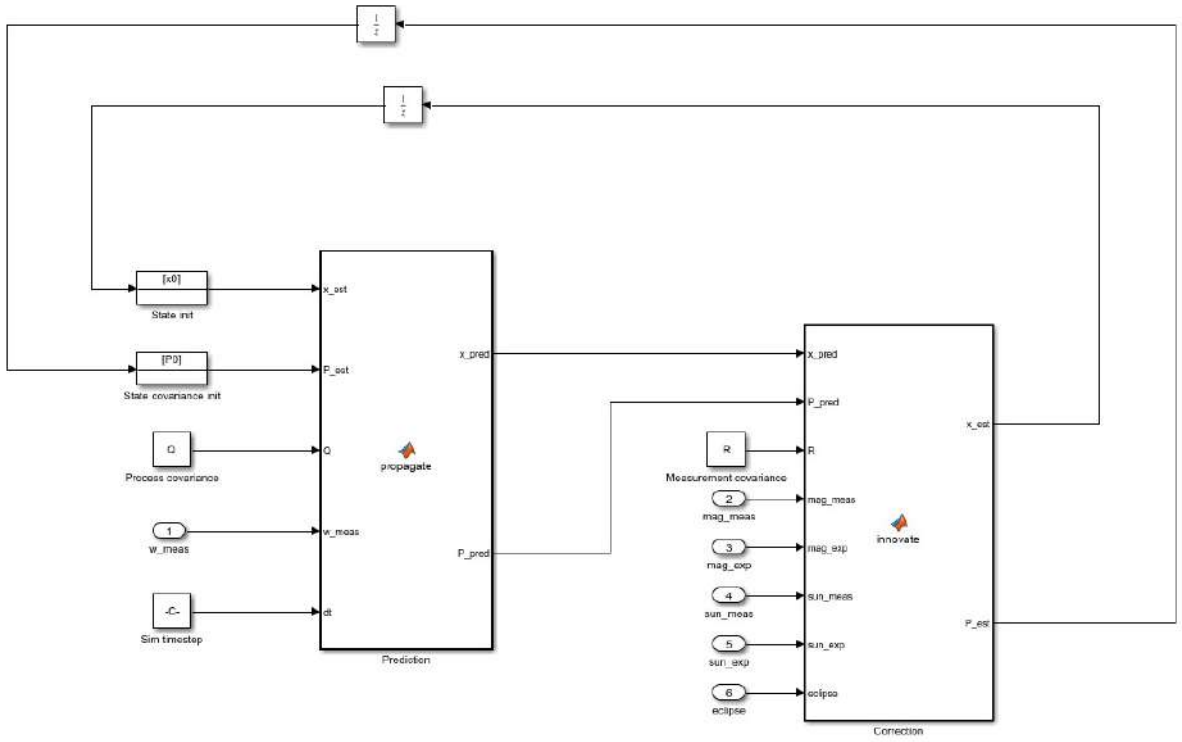


Figure 85: MEKF implementation subsystem.

Both measured and expected vectors were normalised vectors, and their measurement residual z was given by

$$z_{B_k} = \tilde{\tilde{B}}_k - A(\hat{q}_k^-)\breve{B}_k(p_k, t_k) \tag{67a}$$

$$z_{s_k} = \tilde{\tilde{s}}_k - A(\hat{q}_k^-)\breve{s}_k(\hat{q}_k^-, p_k, t_k) \tag{67b}$$

where p is the satellite's position, t is the current time, $\hat{q}^-$ is the predicted quaternion attitude, A is the rotation matrix from inertial to body frames. B and s are the magnetic field and Sun vectors, respectively.

As seen on Section 2.2.3, the process covariance matrix Q was formulated with σ_v and σ_u, the ARW standard deviation (st.dev.) and RRW st.dev., respectively. These values were obtained from [63] since the same sensor was used, and they were not present on the sensor's datasheet. The measurement covariance matrix R was adjusted to Equation (68). It required the standard deviation of the magnetic field

$\sigma_{\check{B}}$ and Sun $\sigma_{\check{s}}$ unit vectors instead of the sensors' performance, although being correlated. These were ultimately selected using a trial-and-error approach with two key considerations in mind: maintaining a low attitude error, even during eclipses and with high albedo interference, and maintaining stability during the light-dark transitions.

The four noise parameters were compiled in Table 11 and remained the same during estimation.

$$R = \begin{bmatrix} \sigma_{\check{B}}^2 I_3 & 0_3 \\ 0_3 & \sigma_{\check{s}}^2 I_3 \end{bmatrix} \tag{68}$$

Table 11: MEKF selected standard deviation noises.

Parameter	Symbol	Value	Unit
Gyro ARW st.dev.	σ_v	2.73×10^{-4}	rad/s
Gyro RRW st.dev.	σ_u	1.45×10^{-5}	rad/s^2
Magnetic field vector st.dev.	$\sigma_{\check{B}}$	8.00×10^{-2}	-
Sun vector st.dev.	$\sigma_{\check{s}}$	8.00×10^{-1}	-

Algorithm 12 reflects these changes from Algorithm 4 and, therefore, was the complete and developed method for PROMETHEUS-1.

Algorithm 12 Discrete-time invariant multiplicative extended Kalman filter with Joseph's form, including magnetic field and Sun unit vectors as measurement updates.

1 $\hat{q}_0^+, \hat{\beta}_{\omega_0}^+, P_0^+$ ▷ initialize state estimates and covariance
2 **for all** timesteps k **do**
3 $\quad \omega_k \leftarrow \tilde{\omega}_k - \hat{\beta}_{\omega_k}$ ▷ calculate true angular velocity
4 $\quad \theta \leftarrow \|\omega_k\| \Delta t$ ▷ calculate angle of rotation
5 $\quad u \leftarrow \omega_k / \|\omega_k\|$ ▷ calculate axis of rotation
6 $\quad \hat{q}_k^- \leftarrow \hat{q}_{k-1}^+ \otimes \begin{bmatrix} \cos(\theta/2) & ; & u \sin(\theta/2) \end{bmatrix}$ ▷ predict attitude estimate
7 $\quad \hat{\beta}_{\omega_k}^- \leftarrow \hat{\beta}_{\omega_{k-1}}^+$ ▷ predict gyro bias estimate
8 $\quad F_{k-1} \leftarrow \begin{bmatrix} \left[I_3 + (\sin\theta)[u]_\times + (1-\cos\theta)[u]_\times^2\right]^T & -I_3 \Delta t & ; & 0_3 & I_3 \end{bmatrix}$
9 $\quad P_k^- \leftarrow F_{k-1} P_{k-1}^+ F_{k-1}^T + Q_{k-1}$ ▷ calculate state Jacobian and predict estimate covariance
10 $\quad z_k \leftarrow \begin{bmatrix} \breve{\vec{B}}_k - A(\hat{q}_k^-) \breve{B}_k(p_k, t_k) & ; & \breve{\vec{s}}_k - A(\hat{q}_k^-) \breve{s}_k(\hat{q}_k^-, p_k, t_k) \end{bmatrix}$
11 $\quad H_k \leftarrow \begin{bmatrix} [\breve{\vec{B}}_k]_\times & 0_3 & ; & [\breve{\vec{s}}_k]_\times & 0_3 \end{bmatrix}$ ▷ calculate measurement residual and Jacobian
12 $\quad K_k \leftarrow P_k^- H_k^T (H_k P_k^- H_k^T + R_k)$ ▷ calculate Kalman gain
13 $\quad \Delta x_k^+ \leftarrow K_k z_k$ ▷ calculate estimation error
14 $\quad \hat{q}_k^+ \leftarrow \hat{q}_k^- \otimes \begin{bmatrix} \cos(\|\Delta\psi_k^+\|/2) & ; & (\Delta\psi_k^+ / \|\Delta\psi_k^+\|) \sin(\|\Delta\psi_k^+\|/2) \end{bmatrix}$
15 $\quad \hat{\beta}_{\omega_k}^+ \leftarrow \hat{\beta}_{\omega_k}^- + \Delta\hat{\beta}_{\omega_k}^+$ ▷ correct attitude and gyro bias estimate
16 $\quad P_k^+ \leftarrow (I - K_k H_k) P_k^- (I - K_k H_k)^T + K_k R_k K_k^T$ ▷ correct estimate covariance
17 $\quad \hat{q}_k^+ \leftarrow \hat{q}_k^+ / \|\hat{q}_k^+\|$ ▷ normalize attitude estimate
18 **end for**
19 **return** $\hat{q}_k^+, \hat{\beta}_{\omega_k}^+, P_k^+$

Last but not least, the filter stability was pre-analysed regarding the criteria established in Section 2.2.3:

The solution of the Riccati equation remains positive definite and bounded This was always true since it is formulated using Joseph's form that guaranteed it;

The initial estimation error is small This could be false; however, a deterministic method could be used to estimate a better initial state than a random one;

The noise terms are small This could be false but restricted to the current satellite hardware and models;

The nonlinearities of the system are continuous This was often false since there were discontinuities in Sun vector measurement during the eclipse transitions;

The Jacobian matrix of the output must be bounded This could be false, especially according to the previous item, but was tracked and verified during simulation;

The smallest inverted real part of the Jacobian eigenvalues greatly exceeds the discrete–time step This could be false but was tracked and verified during simulation.

Despite this, these assumptions do not have to be strictly held for the MEKF to work well [63], which is simultaneously worrying and fortunate.

6.5 Attitude determination methodology

When a satellite is deployed, it often has a high spin rate [88]. The ADS would start operating once its angular velocity reaches a reduced threshold after detumbling operations due to the filter's reliance on small-angle approximations. This scenario means that the initial angular velocity and attitude are difficult to define and can be treated as arbitrary.

In order to confidently evaluate the ADS performance in this multitude of possible scenarios, two major considerations were derived for the methodology:

1. two predominant study cases were selected, one where the satellite behaviour was considered to be nominal and another extreme;
2. a Monte Carlo approach was employed, aiming to achieve a robust and statistically validated evaluation.

These considerations enabled the assessment of filter reliability and accuracy under different conditions and sensor noises, two concerns described in the previous section, thereby ensuring its readiness and resilience for real-world space operations.

Some parameters remained constant in both study cases, such as spacecraft and orbit characteristics, as detailed in Table 12. The inertia tensor reflected a uniform distribution of the satellite's weight, approximately 250 grams, across its cubic shape. The selected orbit was an approximate one that the satellite took upon its launch. However, the start epoch was set to one year prior to proceeding with the IGRF-13 model implementation, whose coefficients expired last December.

Table 12: Simulation general initialisation parameters.

Parameter	Symbol	Value	Unit
Epoch	t_0	$14/01/2024, 19{:}09{:}00$	UTC
Orbit semi-major axis	a	6894	km
Orbit eccentricity	ϵ	1.541×10^{-3}	-
Orbit inclination	i	97.48	deg
Orbit Right Ascension of the Ascending Node (RAAN)	Ω	152.8	deg
Orbit argument of perigee	ϖ	67.24	deg
Orbit initial true anomaly	θ_0	0	deg
Orbit period	T_{orb}	5688	s
Satellite mass	M	0.250	kg
Satellite inertia tensor	J	$1.042 \times 10^{-4} I_3$	kg m^2

Furthermore, the initial filter state was arbitrarily set to the identity quaternion, and the gyro bias was set to null. The covariance matrix, however, was proposed to be initialised with Equation (69), following several authors recommendations [89].

$$P_0 = diag((\hat{x}_0 - x_0)^T (\hat{x}_0 - x_0)) \tag{69}$$

Considering the worst-case scenario where the true attitude is in the opposite direction of the one initialised, and the bias is twice the maximum value of the datasheet, the initial uncertainty was defined by the matrix expressed in Table 13.

Table 13: MEKF initialization parameters.

Parameter	Symbol	Value	Unit
Initial attitude quaternion	$\hat{q}_0^+$	$[1; 0; 0; 0]$	-
Initial gyro bias	$\hat{\beta}_{\omega_0}^+$	$[0; 0; 0]$	deg/s
Initial state covariance matrix	P_0^+	$diag([180 I_3 \;\; 2 I_3])$	[deg deg/s]

The first study case was considered nominal because the angular velocity of the satellite was close to the expected value after detumbling [87]. Moreover, the gyro bias was set w.r.t. the offset bounds expressed in its datasheet, and the attitude to be around half the possible angular displacement.

The second case study was described as extreme because the angular velocity was ten times greater than before, which could occur if detumbling is not successful as intended, either due to an actuation problem or high sensor offsets, for example. The true orientation was set to be opposite of the estimate, and the initial gyro bias to be twice as high as before.

Despite these fixed constraints, detailed in Table 14 for each scenario, each vector component was generated randomly with a uniform distribution at the start of each trial.

Table 14: Satellite initialization constraints.

Parameter	Symbol	Value		Unit
		Nominal case	Extreme case	
Attitude displacement	Δq	90	180	deg
Gyro bias bounds	β_ω	$[-1 \;\; 1]$	$[-2 \;\; 2]$	deg/s
Angular velocity norm	$\|\omega\|$	0.25	2.5	deg/s

All simulations ran over three orbits at 100 ms fixed-timestep increments. Each study case ran for twenty trials, each with different random seeds defined. In each trial, the error between the true state and the filter estimate was collected at all time steps.

The error calculation for the attitude quaternion differed from the gyro bias. The first was obtained by the 3D angular displacement between the truth and the estimate, while the latter was calculated with the norm of the residual. The filter performance was evaluated w.r.t. five criteria:

1. the filter convergence time;
2. the error mean, after convergence;
3. the error standard deviation, after convergence;
4. the error maximum value, after convergence;
5. the error minimum value, after convergence;

The convergence was defined as achieved once the st.dev. of a 500-element sliding window was below 0.1. After all trials were finished, each metric was also averaged for each study case to simplify the results.

6.6 Attitude determination results analysis

After collecting the data of all 40 trials, with over 47 hours of continuous computing with a Intel Core i7-8750H, the performance criteria for each case study were evaluated as described. The results are summarised in a table for both the attitude quaternion and the gyro bias, and the convergence time is reported as a normalised value relative to the orbital period for each type of error separately. In addition, a plot is provided, showing the evolution of each error across the first three orbits upon initialisation. The main line represents the mean error of the trials, with the minimum and maximum value bounds represented as a light shadow. The eclipse region, where the Sun measurement is lost, is also shown. A

zoomed region of the plot after convergence is provided. Items 5 and 6 of the stability criteria defined in Section 6.4 were successfully achieved after numerical verification of the majority of trials.

6.6.1 Nominal case study

The results are shown in Table 15. The attitude quaternion converged rapidly, with an average convergence time of just 5.3% of the orbital period, equivalent to approximately five minutes. This fast response is particularly notable given that no external deterministic initialisation was used. Furthermore, the narrow spread of convergence times indicates that the filter was robust to variations in initial conditions or measurement noise in the nominal regime.

The mean attitude estimation error settled at 0.946 degrees, with a standard deviation of 0.476 degrees. These numbers highlight the high accuracy and repeatability of the solution, especially considering that the maximum observed error remained below 2.87 degrees in all cases, and the average minimum error was 0.03 degrees. The low dispersion between trials, as seen in the min/max bounds that closely followed the mean trajectory in Figure 86, suggests that the residual error was largely deterministic, likely dominated by model imperfections in either the hardware or environment blocks.

The zoomed-in plot further reveals that the attitude error increased in the later stages of the eclipse when measurement availability was limited and prolonged. These local maxima were quickly suppressed when Sun vector data became available again. Crucially, the filter's performance did not degrade severely in shadowed conditions. This stability could be a direct consequence of appropriately inflated covariance of the magnetic field vector, which down-weighted their influence during uncertain periods. Instead, the filter relied more heavily on gyroscope data, which was continuously available and corrected for time-varying bias in real time.

Regarding gyroscope bias estimation, the filter converged even faster than the attitude solution, with a mean convergence time of only 0.4% of the orbit, which translates to roughly 23 seconds. After convergence, the average bias error was 0.031 degrees per second, with a trial-to-trial deviation of 0.015 degrees per second, indicating a satisfactory performance. The maximum error observed across all runs remained below 0.603 degrees per second.

On the zoomed plot of Figure 87, the error mean evolution stayed nearly constant, showing that the MEKF performed well regarding the time-varying nature of the ARW and RRW biases, especially the latter. The bounds were more noticeable due to scale, since the mean to st.dev. ratio was approximately the same as in the attitude error. The eclipse behaviour was also similar to the one described before, as expected.

Overall, these results validated that the implemented filter architecture was not only fast to converge but also highly stable and precise under nominal operating conditions. This regime established a strong baseline for evaluating its performance under more demanding scenarios, such as those involving high angular velocity, as seen in the next section.

Table 15: State error performance results on the nominal study case.

Trial ID	Attitude quaternion error					Gyro bias error				
	Convergence time (t/T_{orb})	Mean [deg]	St.dev. [deg]	Max. [deg]	Min. [deg]	Convergence time (t/T_{orb})	Mean [deg/s]	St.dev. [deg/s]	Max. [deg/s]	Min. [deg/s]
1	0.052	0.946	0.476	2.771	0.006	0.004	0.031	0.015	0.603	0.0009
2	0.053	0.950	0.484	2.776	0.037	0.005	0.031	0.016	0.556	0.0002
3	0.054	0.943	0.474	2.762	0.039	0.004	0.031	0.015	0.507	0.0003
4	0.054	0.947	0.474	2.730	0.051	0.004	0.031	0.015	0.502	0.0007
5	0.055	0.941	0.479	2.801	0.030	0.004	0.030	0.015	0.572	0.0006
6	0.053	0.946	0.476	2.744	0.015	0.004	0.031	0.015	0.589	0.0005
7	0.052	0.945	0.479	2.733	0.038	0.004	0.030	0.015	0.567	0.0006
8	0.053	0.945	0.471	2.630	0.009	0.004	0.031	0.015	0.596	0.0005
9	0.052	0.943	0.477	2.833	0.028	0.005	0.031	0.015	0.560	0.0001
10	0.053	0.946	0.471	2.646	0.022	0.004	0.031	0.015	0.540	0.0005
11	0.054	0.947	0.475	2.709	0.028	0.004	0.031	0.015	0.512	0.0004
12	0.052	0.945	0.476	2.744	0.016	0.004	0.031	0.015	0.548	0.0003
13	0.053	0.952	0.476	2.790	0.025	0.004	0.031	0.015	0.617	0.0002
14	0.053	0.948	0.472	2.712	0.022	0.004	0.031	0.015	0.549	0.0006
15	0.052	0.946	0.477	2.683	0.025	0.004	0.030	0.015	0.509	0.0004
16	0.056	0.945	0.476	2.781	0.015	0.005	0.031	0.015	0.554	0.0006
17	0.052	0.949	0.479	2.800	0.044	0.005	0.030	0.015	0.583	0.0006
18	0.053	0.944	0.474	2.739	0.005	0.004	0.031	0.015	0.589	0.0001
19	0.054	0.947	0.477	2.865	0.026	0.004	0.031	0.015	0.563	0.0005
20	0.056	0.947	0.478	2.730	0.036	0.004	0.031	0.015	0.566	0.0006
Avg.	**0.053**	**0.946**	**0.476**	**2.749**	**0.026**	**0.004**	**0.031**	**0.015**	**0.559**	**0.0005**

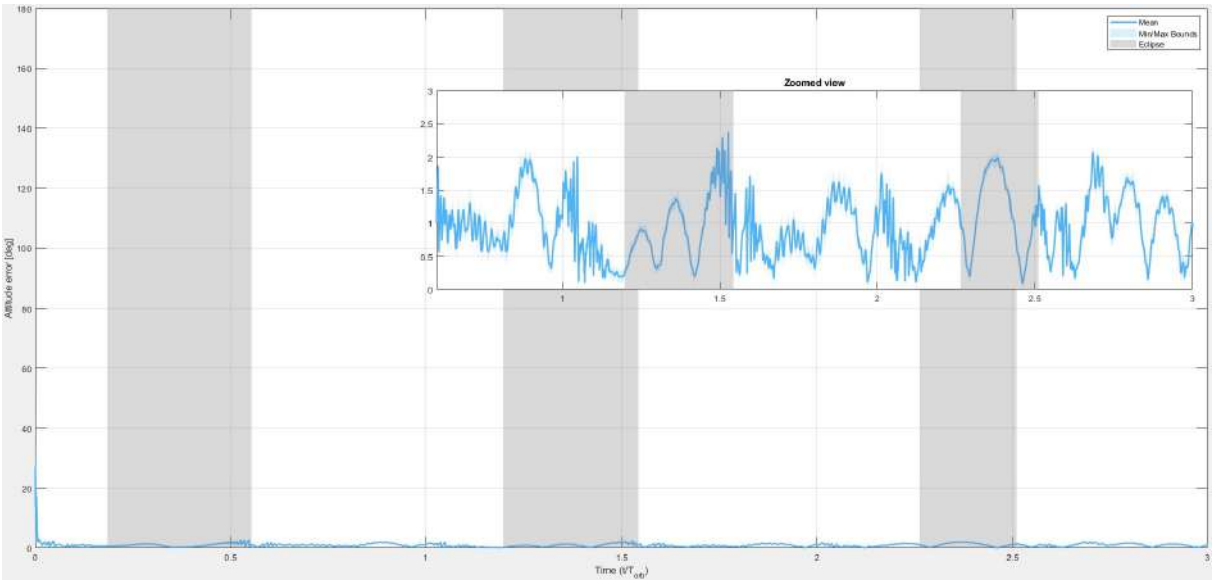


Figure 86: Attitude error evolution on the nominal study case.

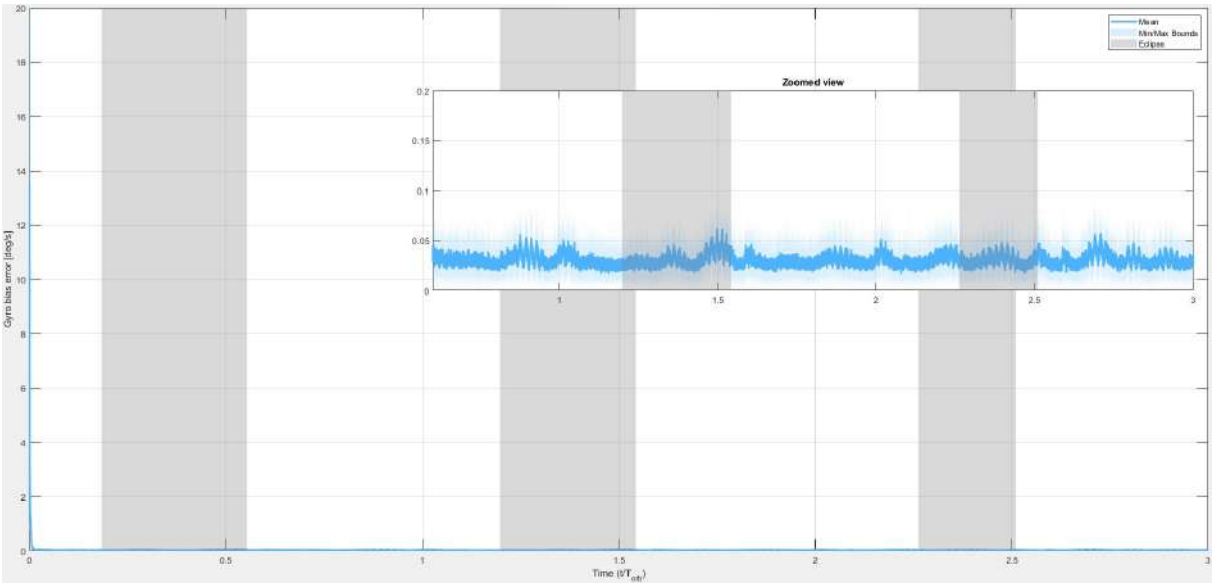


Figure 87: Gyro bias error evolution on the nominal study case.

6.6.2 Extreme case study

In the extreme case scenario, characterised by significantly higher angular velocities, the filter continued to perform robustly, with attitude and bias estimation errors remaining well within acceptable bounds, as shown in Table 16.

On average, the attitude quaternion converged in approximately 10.1% of the orbital period, which, while roughly doubled the nominal case, remained well within optimal values. This increase was expected due to the more aggressive motion. Nevertheless, it still represented a rapid response, considering the increased

dynamic influence and the appearance of some weighty outliers. The most noticeable is trial 16, which exhibited a convergence time of 61.6% of the orbit, whereas it remained severely divergent until then. This isolated outlier likely resulted from unfavourable initial conditions, such as poor alignment between the gyro bias and the true attitude trajectory, or from temporary degradation in measurement quality at initialisation. While rare, such behaviour underscored the importance of initialising state covariances conservatively.

The mean attitude error across all trials remained low at 0.868 degrees, with a standard deviation of 0.446 degrees, representing a slight improvement over the results observed under nominal conditions. Although counterintuitive, the key driver of this lower mean error was the most likely enhanced observability: under higher angular velocity, the Sun and magnetic field vector measurements swept through a broader range of orientations between samples, enriching the information content of each update. In Kalman filter terms, this stronger excitation enabled the filter to distinguish more effectively between true state changes and measurement noise, thereby increasing the update gain and reducing the a posteriori covariance, which in turn decreased the associated estimation error. At the same time, the small angle assumption remained valid, as imperative, because the filter linearised only the incremental error over each very short sampling interval, during which even fast spin rates translated to small rotations.

The maximum attitude error remained under 2.70 degrees, and the minimum values frequently dropped below 0.05 degrees. During eclipse periods, attitude error growth was slightly more restrained compared to the nominal case but performed similarly to previous trials, such as its first transition from dark to light, which occurred at the instant when trial 16 converged. The min/max bounds increased slightly compared to the first case despite the reduced standard deviation. This behaviour means that a few trials may have produced more extreme values while the majority became more consistent.

The mean bias error settled at 0.033 degrees per second, with a low standard deviation of 0.023 degrees per second, slightly above the nominal case. The maximum bias error observed was 0.598 degrees per second, with most trials maintaining peaks below 0.5 degrees per second. However, the average convergence time was 7.0% of the orbit, approximately 17 times longer than in the nominal case, but still considered fast in absolute terms. While a few excursions near 0.8 degrees per second occurred, notably in trial 16, the error distribution remained compact but slightly higher than in the nominal case. This consistency suggests that the filter's bias modelling remained valid even under high-speed dynamics.

Table 16: State error performance results on the extreme study case.

Trial ID	Attitude quaternion error					Gyro bias error				
	Convergence time (t/T_{orb})	Mean [deg]	St.dev. [deg]	Max. [deg]	Min. [deg]	Convergence time (t/T_{orb})	Mean [deg/s]	St.dev. [deg/s]	Max. [deg/s]	Min. [deg/s]
1	0.086	0.865	0.448	2.327	0.023	0.057	0.033	0.029	0.701	0.0007
2	0.108	0.879	0.450	2.694	0.053	0.066	0.034	0.023	0.651	0.0004
3	0.051	0.865	0.441	2.295	0.066	0.028	0.033	0.018	0.500	0.0002
4	0.071	0.862	0.448	2.292	0.025	0.044	0.033	0.025	0.628	0.0003
5	0.054	0.872	0.442	2.386	0.012	0.032	0.033	0.019	0.496	0.0006
6	0.053	0.875	0.445	2.464	0.051	0.030	0.033	0.021	0.493	0.0003
7	0.108	0.874	0.449	2.414	0.040	0.062	0.033	0.020	0.497	0.0006
8	0.050	0.861	0.447	2.205	0.026	0.017	0.033	0.024	0.667	0.0007
9	0.069	0.862	0.445	2.336	0.045	0.041	0.033	0.024	0.680	0.0007
10	0.098	0.867	0.444	2.296	0.014	0.065	0.033	0.020	0.480	0.0004
11	0.058	0.856	0.446	2.388	0.013	0.041	0.032	0.018	0.431	0.0005
12	0.050	0.866	0.440	2.167	0.024	0.017	0.034	0.024	0.559	0.0008
13	0.051	0.872	0.444	2.470	0.026	0.017	0.033	0.019	0.590	0.0006
14	0.088	0.872	0.448	2.377	0.026	0.042	0.034	0.024	0.626	0.0004
15	0.055	0.873	0.451	2.385	0.038	0.033	0.033	0.018	0.519	0.0004
16	0.616	0.872	0.463	2.543	0.030	0.565	0.032	0.030	0.794	0.0003
17	0.107	0.875	0.443	2.661	0.049	0.084	0.034	0.027	0.813	0.0004
18	0.050	0.863	0.443	2.259	0.006	0.018	0.033	0.020	0.566	0.0005
19	0.081	0.877	0.440	2.415	0.030	0.061	0.034	0.022	0.562	0.0007
20	0.114	0.864	0.447	2.279	0.007	0.080	0.034	0.029	0.703	0.0005
Avg.	**0.101**	**0.868**	**0.446**	**2.383**	**0.030**	**0.070**	**0.033**	**0.023**	**0.598**	**0.0005**

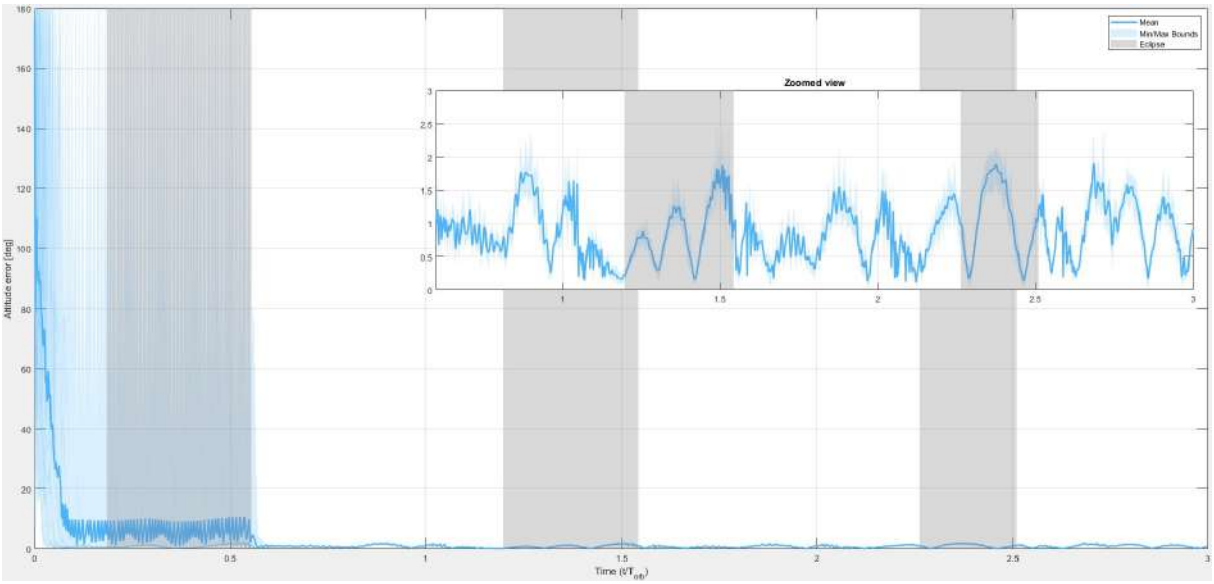


Figure 88: Attitude error evolution on the extreme study case.

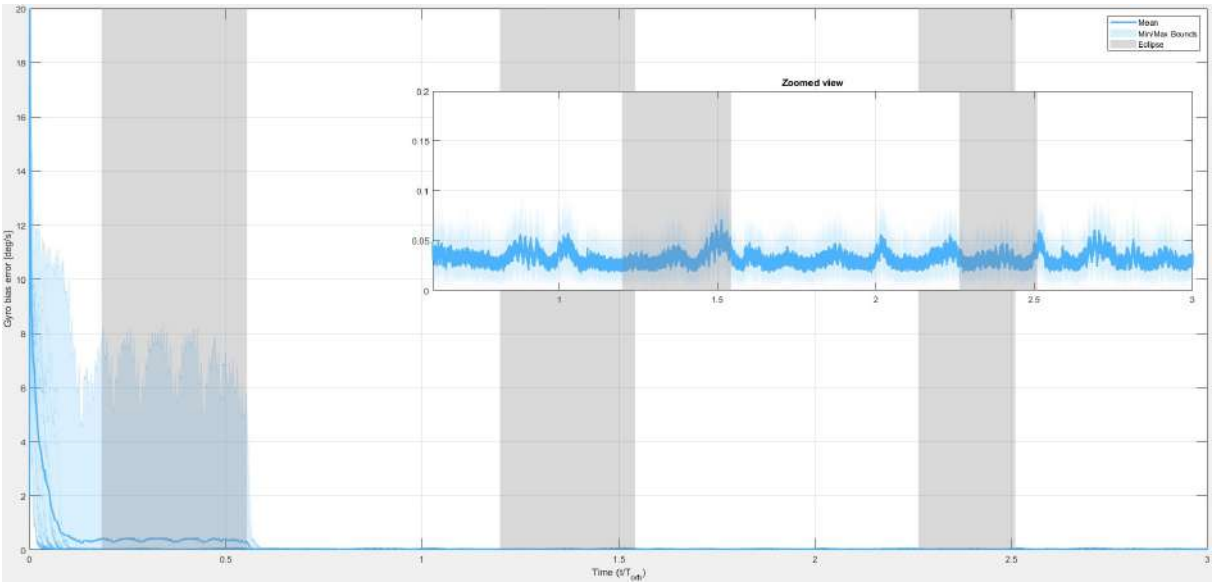


Figure 89: Gyro bias error evolution on the extreme study case.

6.7 Summary

This chapter presented a software simulation framework for the PROMETHEUS satellite. A high-fidelity orbital and sensor simulator was coupled to a quaternion-based invariant MEKF that estimated both the spacecraft attitude and the tri-axial gyro bias. The truth model of Section 6.2 propagated the orbit taken by PROMETHEUS-1, with detail given to attitude and environmental models of the magnetic field, Sun position, albedo reflectance and eclipse status. Sensor blocks of Section 6.3 reproduced the behaviour of the MEMS-based gyro, with special attention to its noise, a MEMS-based magnetometer, from the same IMU unit, and the developed CSSPAs model. The same section detailed the modifications required for the models to be implemented on a resource-constrained device, necessary for determining the expected values of each measurement, as specified by the MEKF implementation described in Section 6.4.

Two Monte Carlo study cases were proposed, one reflecting normal motion conditions after detumbling. At the same time, the second exposed the satellite to a more aggressive behaviour, roughly an order of magnitude more, essentially to probe the limits of the filter. The initialisation parameters and evaluation metrics were derived in Section 6.5, and the results were analysed in Section 6.6.

Succinctly, in the nominal set and on average, the quaternion error converged after 5.3% of the orbit, achieving a mean of 0.946 degrees, with a standard deviation of 0.476 degrees. In contrast, the gyro bias converged at 0.4%, with a mean of 0.031 degrees per second and a standard deviation of 0.015 degrees per second. These results demonstrate both rapid convergence and highly stable, precise performance.

In the extreme set, the quaternion error converged on average after 10.1% of the orbit, achieving a mean of 0.868 degrees with a standard deviation of 0.446 degrees. The gyro bias converged at 7%, with a mean of 0.033 degrees per second and a standard deviation of 0.023 degrees per second. The worse convergence was mostly explained by an outlier trial resulting from unfavourable initial conditions. However, the attitude error metrics slightly improved, most likely explained by enhanced observability. These results append robustness and reliability to the implementation.

Several technical insights emerged. First, the MEKF maintained accuracy and stability throughout all test cases despite being initialised at the worst-credible error bounds, which proved its reliability as an on-board attitude estimation method. This success can be partially attributed to the invariant implementation of the filter, which specifically addresses this issue. Second, the lone outlier underscored the importance of conservative covariance initialisation paired with deterministic attitude methods for state initialisation. Third, faster dynamics could be beneficial rather than detrimental, ensuring the filter runs fast enough to keep incremental rotations within the small-angle assumption. Lastly, the 10 Hertz sample rate offered a comfortable margin for estimating attitude, as the defined extreme scenario achieved significant success, thus providing a safety net for onboard implementation.

Overall, the simulation study with the current implementation showed that the proposed Sun-sensor- and magnetometer-aided invariant MEKF for PROMETHEUS-1 consistently delivered sub-degree steady-state accuracy and rapid convergence across a broad dynamic spectrum. However, further investigation is required to fully evaluate its performance, such as describing the sensor noise sources in more detail, and also implementing it on flight hardware, preferably during HITL testing.

Chapter 7

Conclusion

7.1 Concluding remarks

This dissertation addressed the design, development, and validation of a modular CSSPA tailored for attitude determination in nanosatellites, with a specific focus on integration into PROMETHEUS-1. From conceptual formulation to near-flight readiness, each chapter contributed to demonstrating that low-cost, compact, and fault-tolerant Sun sensing systems can be achieved using COTS components, without compromising accuracy or reliability.

The work began with a comprehensive literature review, outlined in Chapter 2, which provided the theoretical foundations for attitude determination and spacecraft sensors. It analysed sensor typologies, with emphasis on the role and limitations of CSSs compared to more complex sensors. State-of-the-art array sensors were reviewed, revealing trade-offs in accuracy, FOV, redundancy, and cost. It was found that while many designs offer solid performance, none address the unique challenges of modularity, manufacturability, and integration simplicity, especially for constrained platforms like PocketQubes. Sun vector determination and calibration procedures were also reviewed.

In Chapter 3, a proof-of-concept prototype was developed using 3D-printed supports and OPT4001 illuminance transducers, extensively compared to other COTS alternatives, arranged in a four-sided pyramid. The prototype was tested on a single-axis fast-prototyped testbed using an almost collimated flashlight as the light source. A polynomial-based model calibration outperformed the traditional cosine one in estimating the Sun vector, reducing the angular error from 3.34 to 0.60 degrees on average. Despite limitations in the test environment and hardware, this phase validated the feasibility of the CSSPA concept and highlighted key improvements for further iterations, particularly the need for smaller geometry, reduced light

source interference, and more accurate sensor calibration procedures.

Chapter 4 detailed the development of the advanced prototype for the CSSPA, focusing on design refinement, manufacturing innovation, and comprehensive testing. An array topology optimisation was conducted, with multiple sensor configurations evaluated through Monte Carlo simulations over a discretised attitude sphere. A square pyramid with four transducers tilted at 30 degrees proved optimal, achieving 100% hemispherical coverage and high resilience to sensor failure, while maintaining over 90% coverage even with up to eight transducer faults. This configuration substantially outperforms the one currently flying on PROMETHEUS-1. In the miniaturisation phase, an FPC design made of space-grade polyimide was introduced, drastically reducing volume while enabling a more mechanically adaptive assembly. A new structural support with softened edge transitions was developed to accommodate the FPC bending without damaging electrical traces. The experimental methodology employed a dual-axis automated rotation stage inside an anechoic chamber. Illumination stability was modelled and compensated, while rotation paths allowed complete hemisphere sampling, and careful alignment minimised translational error. Sun vector estimation was calibrated in three sequential stages: transducer orientation, sensor response linearity, and light transmittance, resulting in an average angular error reduction from 2.97 to 1.61 degrees, which is slightly above the defined objective. Combined array testing demonstrated superior spatial uniformity and redundancy, confirming the system's capability to operate with minimal performance degradation due to enumerated plausible error sources. Compared to state-of-the-art alternatives, it offers a compelling trade-off between performance, redundancy and manufacturability. This prototype successfully demonstrated robust and accurate Sun vector determination, full-sky observability, and fault-tolerant performance in a compact, modular form factor, while also establishing the required guidelines for the final prototype.

Chapter 5 focused on the final prototype design, addressing the physical, electrical, and thermal constraints for integration into PROMETHEUS-1. A new mechanical support made from space-compliant Windform XT 2.0 composite was designed to meet PocketQube dimensional restrictions and comply with the harsh space environment. The flexible circuit was redesigned with tighter trace widths and EMI shielding. At the same time, the OPT4001 was replaced with the similar OPT4003-Q1, offering an enhanced light range, robustness, and near-infrared sensing capabilities. A TMP117 temperature sensor was added to enable in-situ calibration and thermal sensing. Satellite integration was carefully planned through schematic and PCB modifications, I2C bus management via a multiplexer, and simple software updates. However, due to time constraints, only the design was completed. Nevertheless, this chapter established a pre-validated, flight-ready sensor blueprint for future implementation.

Chapter 6 introduced a software simulation to assess system performance under realistic orbital dynamics. Orbital and sensor models were coupled to the employed invariant MEKF formulation, estimating both attitude and gyroscope bias from simulated corrupted Sun and magnetic field vector measurements. Results from two Monte Carlo scenarios, nominal and extreme, demonstrated sub-degree steady-state accuracy, rapid convergence, and filter robustness, even under fast dynamic regimes. These findings assessed the CSSPA performance on the PROMETHEUS-1 ADS, providing a useful tool for validating such systems on future missions.

All proposed research objectives were achieved.

7.2 Research answers

Through the development of the dissertation, the proposed research questions were answered accordingly:

Q1 — What are the critical design considerations for an array of Sun sensors optimised for accuracy and redundancy while keeping costs low and remaining easily manufactured? The chosen design leverages a pyramid-shaped configuration using high-performance COTS transducers mounted on a FPC, thereby minimising cost, weight, and manufacturing complexity. Accuracy and redundancy were maximised through optimised transducer orientation, wide coverage, and enhanced calibration techniques targeting the most common sources of error of photoelectric transducers. Additive manufacturing and simplified electronics integration allow straightforward scaling and assembly, making the solution simple and modular.

Q2 — How can the performance of the developed sensor array be validated under various operating conditions? Laboratory validation was conducted through multiple trials using precise rotation tables under partially light-controlled conditions. The setup simulated illumination from all possible directions, allowing robust evaluation of measurement repeatability and Sun vector accuracy across the multiple developed sensor units. The simulated performance of attitude determination was further validated in a software simulation framework using realistic environmental models. Temperature and radiation dependence should also be evaluated in the future; however, the required instrumentation could be expensive.

Q3 — What calibration methods are necessary to ensure reliable Sun vector determination using the sensor array? Calibration included compensating for sensor misalignment via normal vector estimation, illuminance modelling using cosine-based fits, and iterative transmittance correction. Custom methods were also developed to mitigate light source non-uniformity and experimental setup misalignments. Better calibration is also feasible using methods that rely on known spectral light sources, providing a more robust alternative to costly laboratory equipment. Nevertheless, online calibration is the most recommended method for future implementations, either by following the proposed framework or by augmenting the filter state with the calibration coefficients.

Q4 — How does the performance of the developed sensor array compare with other solutions? Uncalibrated, the developed CSSPA demonstrated roughly twice the accuracy of similar literature examples under the same conditions. After calibration, its mean angular accuracy of 1.6 degrees and a standard deviation of 0.84 degrees matched the performance of other similar implementations, while preserving a significantly smaller form factor, reduced cost, and a simpler assembly process. Therefore, while the raw performance is comparable post-calibration, the CSSPA's value proposition lies in its compactness, cost-effectiveness, modularity, manufacturability and documentation.

Q5 — How can the sensor array be interfaced to achieve seamless integration into the already implemented satellite's ADS? Mechanically, sensors are mounted on raised supports with alignment features to ensure proper orientation without blocking their view. Electrically, an I2C multiplexer manages communication between the MCU and the multiple CSSPAs, requiring some simple connector redesigns. Software changes include adding drivers for new transducers and upgrading the Sun vector estimation algorithm. Although the integration plan is well-documented, physical testing and validation remain to be completed.

Q6 — What characterizes high-fidelity simulation environments for attitude estimation? They include realistic orbital and environmental modelling, incorporating detailed orbital propagators, Earth's magnetic field, the Sun's position and albedo models, as well as eclipse detection. Additionally, it provides accurate sensor behaviour, with realistic noise profiles, non-linearities, and potential failure modes that mimic those found in actual hardware. They are modular and scalable, supporting integration with onboard SITL and HITL testing. These environments enable validation

under both nominal and extreme conditions, providing a reliable platform to test the ADS and open to include other satellite subsystems.

7.3 Future directions

The work developed in this dissertation opens up several opportunities for future research and improvement, either in the sensor or in the satellite's ADCS.

In the event of integrating the proposed CSSPA into a flying satellite, several improvements should be made to the testing setup so that its performance can be assessed more reliably. Starting by ensuring that misalignments are not present is critical, which would be considerably improved by operating in an optical table with mechanical supports. The most advanced photoelectric transducer calibration techniques found in the literature are based on the usage of AM0 spectrum light sources, and they should be employed if possible. However, this requires specific instrumentation, which is usually costly for a university project. Fortunately, it is possible, to some extent, to characterise the in-flight performance of the photodiode without one if it is tested with a source of light whose spectrum is fully known [90]. The testing setup should include a slip ring for the connection between the CSSPAs and the MCU, allowing for uninterrupted performance. This modification enables the calibration procedure to employ the Fibonacci sphere distribution when collecting measurements from all possible directions rather than using fixed-size steps on both rotation stages. Ultimately, while pre-flight calibration is helpful to assess the sensor's initial performance, on-orbit calibration is more useful because it estimates the evolution of calibration coefficients during the mission. As explained below, this applies to all types of sensors. Additionally, the specific framework proposed by Mafi et al. [36] could also be performed in parallel.

Temperature and radiation calibration were not conducted; however, this could lead to an integrated, state-of-the-art framework for the complete calibration of Sun sensors. For increased functionality, the transducers can also help with antenna deployment verification [40]. This author also explores iterative methods of optimising sensors' orientation that could be evaluated for different configurations of the CSSPA. Finally, the total illuminance in solar AM0 irradiance should be recalculated, tackling the limitations raised in Appendix A1.

Prolonged radiation exposure can cause degradation of the transducer. Using acrylics such as Poly(methyl methacrylate) (PMMA) can effectively block harmful ultraviolet wavelengths without attenuating visible light [91], and could be an option for a future, more complex design of the CSSPA. Simpler improvements,

such as hexagonal copper hatching and teardrop-shaped finger pads and vias, can significantly enhance the flexibility and durability of the FPC [71].

The shared bus architecture of the I2C protocol means that a single device failure can compromise the entire bus. As more devices are added, the probability of failure increases exponentially, making reliability a critical concern in aerospace applications [92]. To address this, automatic isolation circuits specifically designed for I2C are available in the literature [92], allowing faulty devices to be isolated without requiring additional processing overhead. These circuits have been integrated into some I2C devices in PROMETHEUS-1 to ensure robust and fault-tolerant communication; however, the current CSSPA design does not yet implement them. Although its redundancy was planned and accounted for, a failure of this nature could damage four photoelectric transducers at a time. As the FPC is already densely packed with components and traces, the solution requires increasing the number of layers to route each transducer's I2C pins to the edge connector. This way, the isolation circuit could be implemented in the solar panel PCB for each transducer. Consequently, the cost will increase, but it is a necessary improvement for full protection.

Finally, although it was estimated that all six final CSSPAs might not exceed the satellite's mass budget, it shall be precisely weighed to ensure it.

Several improvements could be performed on the current PROMETHEUS-1 ADS. Implementing Murrell's version of the MEKF should be a priority, as it drastically improves speed and memory without affecting the results. Square root methods guarantee positive definiteness and improve convergence and precision, but typically require 50-150% more computational time than that required by the traditional versions [30]. Regarding the current measurement update form, it was chosen to take unit vectors as input, which, although it simplifies the problem, constraints on-orbit calibration of the sensors and reduces the impact of albedo influence on the Sun sensors. To improve this topic, the filter state should be augmented to include desired calibration coefficients and error sources, such as gains and biases. The measurement equations and covariances should be updated to reflect the raw measurements. These changes can be adapted from Jensen [63] or Springmann [40]. If not pursued, some other methods should be implemented to improve the Sun vector estimation problem further: a weighted LS to trust sensors with more intensity [93]; add more transient-friendly angle-based thresholds for low-incidence expected measurements [93]; and geometrically project measurements between transducers to evaluate its trustworthiness [48, 93].

The MEKF would also benefit from concrete stability prooving, such as the one suggested by Tønne [66],

and from deterministic methods to estimate a better initial state, or to reset the filter once convergence cannot be achieved after a long period.

The simulation methodology could be improved by simulating with more case studies, such as different initial conditions for all sensors, evaluating the successive failures of the photoelectric transducers, or even evaluating the effects of orbital position error accumulation on the attitude filter. The obtained results could also be compared to the ADS version that flew on PROMETHEUS-1 to assess the improvement. In addition, SITL testing should be conducted, as most required parts are already developed, followed by HITL testing, once the necessary infrastructure is built.

The satellite ADS also lacks an orbital propagator, which is mandatory for successfully running the MEKF over time, as the satellite position is required for estimating measurements from onboard models. The current propagator used in Simulink could serve as a starting point. However, to achieve good performance, it should be replaced by a numerical one that incorporates perturbations, such as atmospheric drag, third-body influence, solar radiation pressure, and gravity gradients. The same applies to the simulation environment in Simulink. Baumgarte stabilisation methods could be studied to prevent issues on this matter. The satellite position could also be added to the filter state, together with the orbital propagator as the predictor, and eventually corrected via TLEs from the ground station. The communication link could also provide some positional awareness, but ultimately, adding a GNSS receiver would robustly improve the ADS performance while contributing to a state-of-the-art implementation on PocketQubes.

The albedo data was based on TOMS project, which is two decades old. More recent data was collected by multiple satellites and can be found on Copernicus data store [94]. In addition, the IGRF should be updated to the latest 14^{th} version. The hardware models of the sensors could be improved to reflect more realistically the sensors' errors, such as the data conversion loss common in all, but more particularly the Allan variance gyro and magnetometer noise sources. Rondão [87] provides a well-developed baseline for each employed sensor. The scaling and misalignment errors were not subject of this dissertation, but should be evaluated and added to the implementation.

Lastly, the biggest gap currently in PROMETHEUS-1 ADCS is the lack of an attitude control routine besides detumbling. Although not belonging to this dissertation scope, the satellite would benefit deeply from the addition of different pointing schemes, such as to the Sun or Earth's nadir, but also the current simulation implementation, as adding model actuators and its interactions with the environment would allow a complete validation of an entire subsystem of the satellite, and increment the features of the PROMETHEUS project.

Bibliography

[1] Space Foundation Editorial Team. *Space industry growth: Where are the opportunities in 2022?* https://cie.spacefoundation.org/space-industry-growth-where-are-the-opportunities-in-2022/. Accessed: 2025-05-10. 2023.

[2] Morgan Stanley. *Space: Investing in the Final Frontier*. https://www.morganstanley.com/ideas/investing-in-space. Accessed: 2025-05-10. 2020.

[3] Our World in Data. *Cost of space launches to low Earth Orbit*. https://ourworldindata.org/grapher/cost-space-launches-low-earth-orbit. Accessed: 2025-05-10. 2022.

[4] D. Sinclair and J. Dyer. "Radiation Effects and COTS Parts in SmallSats." In: 2013. url: https://api.semanticscholar.org/CorpusID:108002146.

[5] N. Khera. "PyCubed-Mini: A Low-Cost, Open-Source Satellite Research Platform." Master's thesis. Pittsburgh, PA: Carnegie Mellon University, 2023. url: https://www.ri.cmu.edu/publications/pycubed-mini-a-low-cost-open-source-satellite-research-platform/.

[6] S. Speretta, T. P. Soriano, J. Bouwmeester, J. Carvajal-Godinez, A. Menicucci, T. Watts, P. Sundaramoorthy, J. Guo, and E. Gill. "CubeSats to PocketQubes: Opportunities and Challenges." In: vol. 67. ResearchGate, 2016. url: https://www.researchgate.net/publication/308777632.

[7] E. Kulu. "CubeSats Nanosatellites - 2024 Statistics, Forecast and Reliability." In: vol. 75. 2024. url: https://www.nanosats.eu/assets/CubeSats-Nanosatellites-2024_ErikKulu_IAC2024.pdf.

[8] J. Bouwmeester, S. Radu, M. S. Uludag, N. Chronas, S. Speretta, A. Menicucci, and E. K. Gill. "Utility and constraints of PocketQubes." In: *CEAS Space Journal* 12 (4 Dec. 2020), pp. 573–586. issn: 18682510. doi: 10.1007/s12567-020-00300-0.

[9] E. W. Enger. *Spacecraft Attitude Determination Methods in an Educational Context*. 2019. url: https://api.semanticscholar.org/CorpusID:210902300.

[10] H. D. Black. "A passive system for determining the attitude of a satellite." In: *AIAA Journal* 2.7 (1964), pp. 1350–1351. doi: 10.2514/3.2555.

[11] G. Wahba. "A Least Squares Estimate of Satellite Attitude." In: *SIAM Review* 7.3 (1965), pp. 409–409. doi: 10.1137/1007077.

[12] M. D. Shuster. "The generalized Wahba problem." In: *Journal of the Astronautical Sciences* 54.2 (2006), pp. 245–259. doi: 10.1007/BF03256485.

[13] J. E. Keat. "Analysis of Least-Squares Attitude Determination Routine DOAOP." In: (1977). doi: 10.5281/zenodo.32291.

[14] M. D. Shuster and S. D. Oh. "Three-axis attitude determination from vector observations." In: *Journal of Guidance and Control* 4.1 (1981), pp. 70–77. doi: 10.2514/3.19717.

[15] F. L. Markley. "Attitude Determination using Vector Observations and the Singular Value Decomposition." In: *Journal of the Astronautical Sciences* 36.3 (1988), pp. 254–258. url: https://www.researchgate.net/publication/243753921_Attitude_Determination_Using_Vector_Observations_and_the_Singular_Value_Decomposition.

[16] F. L. Markley. "Attitude Determination Using Vector Observations: a Fast Optimal Matrix Algorithm." In: *Journal of the Astronautical Sciences* 41.2 (1993), pp. 261–280. url: https://www.researchgate.net/publication/4718386_Attitude_Determination_from_Vector_Observations_A_Fast_Optimal_Matrix_Algorithm.

[17] D. Mortari. "ESOQ: A Closed-Form Solution to the Wahba Problem." In: *Journal of the Astronautical Sciences* 45.2 (1997), pp. 195–204. doi: 10.1007/BF03546376.

[18] R. E. Kalman. "A New Approach to Linear Filtering and Prediction Problems." In: *Journal of Basic Engineering* 82.1 (1960), pp. 35–45. issn: 0021-9223. doi: 10.1115/1.3662552.

[19] P. S. Maybeck. *Stochastic Models, Estimation, and Control*. Academic Press, 1979. isbn: 9780124807020.

[20] J. L. Crassidis and J. L. Junkins. *Optimal Estimation of Dynamic Systems*. Chapman and Hall/CRC, 2011. isbn: 9780429105609. doi: 10.1201/b11154.

[21] G. L. Smith, S. F. Schmidt, and L. A. McGee. *Application of Statistical Filter Theory to the Optimal Estimation of Position and Velocity on Board a Circumlunar Vehicle*. Tech. rep. 1962. url: https://ntrs.nasa.gov/citations/19620006857.

[22] J. R. Wertz, D. F. Everett, and J. J. Puschell. *Space Mission Engineering: The New SMAD*. Vol. 28. Microcosm Press, 2011. isbn: 9781881883159.

[23] R. Vepa. *Dynamics and Control of Autonomous Space Vehicles and Robotics*. Cambridge University Press, 2019. isbn: 9781108525404. doi: 10.1017/9781108525404.

[24] *Equipment, Software and Technology Annex*. Tech. rep. Missile Technology Control Regime, 2024. url: https://www.mtcr.info/en/mtcr-annex.

[25] Wikipedia. *Coordinating Committee for Multilateral Export Control*. Accessed: 2025-03-05. url: https://en.wikipedia.org/wiki/Coordinating_Committee_for_Multilateral_Export_Controls#GPS.

[26] J. Solà. *Quaternion kinematics for the error-state Kalman filter*. 2017. arXiv: 1711.02508 [cs.RO].

[27] F. L. Markley. "Attitude Error Representations for Kalman Filtering." In: *Journal of Guidance, Control, and Dynamics* 26 (2 2003), pp. 311–317. issn: 0731-5090. doi: 10.2514/2.5048.

[28] R. Zanneti. "Rotations, Transformations, Left Quaternions, Right Quaternions?" In: *Journal of the Astronautical Sciences* 66.3 (2019), pp. 361–381. doi: 10.1007/s40295-018-00151-2.

[29] H. Sommer, I. Gilitschenski, M. Bloesch, S. Weiss, R. Siegwart, and J. Nieto. *Why and How to Avoid the Flipped Quaternion Multiplication*. 2018. arXiv: 1801.07478 [cs.RO].

[30] F. L. Markley and J. L. Crassidis. *Fundamentals of Spacecraft Attitude Determination and Control*. Springer New York, 2014. isbn: 978-1-4939-0801-1. doi: 10.1007/978-1-4939-0802-8.

[31] D. D. McCarthy and G. Petit. "IERS Conventions (2003)." In: *IERS Technical Note* 32 (2004), p. 1. url: https://iers-conventions.obspm.fr/archive/2003/tn32.pdf.

[32] J. R. Wertz. *Spacecraft Attitude Determination and Control*. Ed. by J. R. Wertz. Vol. 73. Springer Netherlands, 1978. isbn: 978-90-277-1204-2. doi: 10.1007/978-94-009-9907-7.

[33] D. Bhanderi. "Spacecraft Attitude Determination with Earth Albedo Corrected Sun Sensor Measurements." Doctoral dissertation. 2005. isbn: 8790664264. doi: 10.13140/RG.2.2.15490.53441.

[34] L. Salgado-Conrado. “A review on sun position sensors used in solar applications.” In: *Renewable and Sustainable Energy Reviews* 82 (2018), pp. 2128–2146. doi: 10.1016/j.rser.2017.08.040.

[35] NASA. *Spacecraft Sun sensors - NASA space vehicle design criteria*. Tech. rep. 1962. url: https://ntrs.nasa.gov/citations/19620006857.

[36] M. Mafi, S. Saki, and H. Bolandi. “In-orbit calibration of the pyramid coarse sun sensor using Lyapunov-based conical maneuvers of the satellite: A new framework.” In: *Transactions of the Institute of Measurement and Control* 45.14 (2023), pp. 2712–2724. doi: 10.1177/01423312231160583.

[37] F. Gulmammadov, B. Akyazi, and E. Karmşçl. “Field of View and Accuracy Improvement of a Coarse Sun Sensor.” In: *2023 10th International Conference on Recent Advances in Air and Space Technologies (RAST)*. 2023, pp. 1–6. doi: 10.1109/RAST57548.2023.10197654.

[38] *Radiation Transducers*. Accessed: 2025-03-27. 2022. url: https://chem.libretexts.org/@go/page/366613.

[39] I. Luque-Heredia, G. Quéméré, R. Cervantes, O. Laurent, E. Chiappori, and J. Y. Chong. “The Sun Tracker in Concentrator Photovoltaics.” In: *Next Generation of Photovoltaics: New Concepts*. Ed. by A. B. Cristóbal López, A. Martí Vega, and A. Luque López. Springer, 2012, pp. 61–93. isbn: 978-3-642-23369-2. doi: 10.1007/978-3-642-23369-2_3.

[40] J. C. Springmann. “Satellite Attitude Determination with Low-Cost Sensors.” Doctoral dissertation. University of Michigan, 2013. url: https://deepblue.lib.umich.edu/handle/2027.42/102312.

[41] J Wang, X Wang, and J. Y. Yang. “A Sun Sensor Based on Regular Pyramid Sensor Arrays.” In: *Journal of Physics: Conference Series* 1207 (1 Apr. 2019), p. 012010. doi: 10.1088/1742-6596/1207/1/012010.

[42] W. U. of Technology. *PW-Sat2 Sun Sensor Critical Design Review*. 2016. url: https://pw-sat.pl/en/documentation/.

[43] H. E. Soken, S. N. Sozen, M. Gokce, C. Yavuzyilmaz, and F. Gulmammadov. “Analog sun sensor measurement correction using deep neural network.” In: *Acta Astronautica* 211 (Oct. 2023), pp. 808–817. doi: 10.1016/j.actaastro.2023.07.013.

[44] R. J. Saborio. *Characterizing Performance and Errors of Coarse Sun Sensors*. 2017. url: https://hdl.handle.net/11299/191874.

[45] A. Adnane, M. E. A. Cheriet, E. Benfriha, A. Bellar, M. A. S. Mohammed, and Y. Kebbati. "Sun Sensor Design for Full Field of View Coverage." In: Institute of Electrical and Electronics Engineers Inc., 2022, pp. 85–90. doi: 10.1109/IC_ASET53395.2022.9765945.

[46] S. Rhee, H. Lee, M. Nam, and D. Park. "Development of Pyramidal Type 2-Axes Analog Sun Sensor." In: *Journal of Astronomy and Space Sciences* (2000). url: https://www.researchgate.net/publication/252897570_Development_of_Pyramidal_Type_2AXES_Analog_Sun_Sensor.

[47] Y. Away and M. Ikhsan. "Dual-axis sun tracker sensor based on tetrahedron geometry." In: *Automation in Construction* 73 (Jan. 2017), pp. 175–183. doi: 10.1016/j.autcon.2016.10.009.

[48] L. C. Pita, S. E. S. Roman, J. M. Giron-Sierra, J. R. Barriga, P. D. D. Vicente, and M. A. Jerez. "Getting More Performance from INTA NanoSat-1B Truncated Pyramid Sun Sensors." In: *IEEE Sensors Journal* 14 (6 2014), pp. 1867–1877. doi: 10.1109/JSEN.2014.2305471.

[49] J. Barnes, C. Liu, and K. Ariyur. "A hemispherical sun sensor for orientation and geolocation." In: *IEEE Sensors Journal* 14 (12 2014), pp. 4423–4433. doi: 10.1109/JSEN.2014.2331319.

[50] A. S. Clair et al. "Sun Determination of the DustBUSTER CubeSat." In: (). url: https://www.colorado.edu/aerospace/sites/default/files/attached-files/dust_buster_aiaa.pdf.

[51] M. Taraba, C. Rayburn, A. Tsuda, and C. MacGillivray. "Boeing's CubeSat TestBed 1 Attitude Determination Design and On-Orbit Experience." In: 2009. url: https://digitalcommons.usu.edu/smallsat/2009/all2009/64.

[52] D. H. Zhuo, A. Rai, S. Vosoogh-Grayli, G. W. Leach, and B. Bahreyni. "A micromachined vector light sensor." In: *Sensors and Actuators, A: Physical* 311 (2020). doi: 10.1016/j.sna.2020.112045.

[53] W. De Soto, S. Klein, and W. Beckman. "Improvement and validation of a model for photovoltaic array performance." In: *Solar Energy* 80.1 (2006), pp. 78–88. doi: 10.1016/j.solener.2005.06.010.

[54] E. Hecht. *Optics, 5th edition*. Pearson Education Ltd, 2016. isbn: 9781292096933.

[55] J. C. Springmann, A. J. Sloboda, A. T. Klesh, M. W. Bennett, and J. W. Cutler. "The attitude determination system of the RAX satellite." In: *Acta Astronautica* 75 (2012), pp. 120–135. doi: 10.1016/j.actaastro.2012.02.001.

[56] Y. Nurgizat, G. Balbayev, and D. Galayko. "Solar sensor for Cubesat attitude determination." In: *2021 28th IEEE International Conference on Electronics, Circuits, and Systems, ICECS 2021 - Proceedings*. Institute of Electrical and Electronics Engineers Inc., 2021. isbn: 9781728182810. doi: 10.1109/ICECS53924.2021.9665513.

[57] S. A. O'Keefe and H. Schaub. "Sun-direction estimation using a partially underdetermined set of coarse sun sensors." In: *Journal of the Astronautical Sciences* 61 (1 2014), pp. 85–106. doi: 10.1007/s40295-015-0058-9.

[58] F. L. Markley, J. Crassidis, and Y. Cheng. "Nonlinear Attitude Filtering Methods." In: *AIAA Guidance, Navigation, and Control Conference and Exhibit*. American Institute of Aeronautics and Astronautics, 2005. isbn: 978-1-62410-056-7. doi: 10.2514/6.2005-5927.

[59] I. Y. Bar-Itzhack and Y. Oshman. "Attitude Determination from Vector Observations: Quaternion Estimation." In: *IEEE Transactions on Aerospace and Electronic Systems* Vol. AES-21 (1985), pp. 128–136. doi: 10.1109/TAES.1985.310546.

[60] D. C. Paulson, D. B. Jackson, and C. D. Brown. "SPARS Algorithms and Simulation Results." In: vol. 1. 1969, pp. 293–317. url: https://malcolmdshuster.com/HTF_Paulsson-Jackson_Brown_SPARS_1969_FLM.pdf.

[61] N. F. Toda, J. L. Heiss, and F. H. Schlee. "SPARS: the System, Algorithm, and Test Results." In: vol. 1. 1969, pp. 361–370. url: https://malcolmdshuster.com/HTF_Toda-Heiss-Schlee_SPARS_1969_FLM.pdf.

[62] J. Deutschmann, L. Markley, and I. Bar-Itzhack. *Quaternion normalization in spacecraft attitude determination*. Tech. rep. 1993. url: https://ntrs.nasa.gov/citations/19930015541.

[63] B. Jensen. "A Low-Cost Attitude Determination and Control System and Hardware-in-the-Loop Testbed for CubeSats." Master's thesis. Pittsburgh, PA: Carnegie Mellon University, 2022. url: https://www.ri.cmu.edu/publications/a-low-cost-attitude-determination-and-control-system-and-hardware-in-the-loop-testbed-for-cubesats/.

[64] R. L. Farrenkopf. “Analytic Steady-State Accuracy Solutions for Two Common Spacecraft Attitude Estimators.” In: *Journal of Guidance, Control, and Dynamics* 1 (4 1978), pp. 282–284. issn: 0731-5090. doi: 10.2514/3.55779.

[65] L. Chang. “Multiplicative extended Kalman filter ignoring initial conditions for attitude estimation using vector observations.” In: *Journal of Navigation* 76.1 (2023), pp. 62–76. issn: 1469-7785. doi: 10.1017/s0373463322000443.

[66] K. K. Tønne. “Stability Analysis of EKF-based Attitude Determination and Control.” Master’s thesis. Trondheim, Norway: Norwegian University of Science and Technology, 2007. url: https://ntnuopen.ntnu.no/ntnu-xmlui/handle/11250/259608.

[67] Texas Instruments. *OPT4001 High Speed, High Precision, Digital Ambient Light Sensor*. https://www.ti.com/lit/ds/symlink/opt4001.pdf. Datasheet, Revision A, Accessed: 2025-04-24. 2022.

[68] Adafruit. *Adafruit Feather M4 Express*. https://cdn-learn.adafruit.com/downloads/pdf/adafruit-feather-m4-express-atsamd51.pdf. Datasheet, Accessed: 2025-04-24. 2025.

[69] PyCubed-Mini. *CircuitPython driver for the OPT4001 Ambient Light Sensor*. https://github.com/PyCubed-Mini/CircuitPython_OPT4001. Github Repository, Accessed: 2025-04-24. 2023.

[70] Night Stick. *NSP-420 Adjustable Beam Flashlight*. https://docs.baycoproducts.com/images/PDF/ProductDataSheets/PDS_NSP-420.pdf. Datasheet, Accessed: 2025-04-24.

[71] Altium. *Rigid-Flex PCB Design Guidelines*. Tech. rep. Altium LLC, 2022. url: https://resources.altium.com/p/altium-rigid-flex-guidebook.

[72] MOLEX. *Easy-On FFC/FPC Connector, 0.50mm Pitch, Slider Series, Right-Angle, Bottom Contact, 2.00mm Height, 6 Circuits, Gold Plating*. https://www.molex.com/content/dam/molex/molex-dot-com/products/automated/en-us/salesdrawingpdf/527/52746/527460671_sd.pdf. Datasheet, Revision J, Accessed: 2025-05-10. 2014.

[73] H. Furutan and H. Nagano. “Novel Thermoplastic Polyimide Adhesives; KANEKA PIXEO.” In: *Japan Polyimide Conference*. 1998. url: http://fibertech.or.jp/JPIC/pdf/1999_21.pdf.

[74] D. J. McClure. “Polyimide Film as a Vacuum Coating Substrate.” In: *53rd Annual Technical Conference Proceedings*. Society of Vacuum Coaters. Orlando, FL, USA, 2010. url: https://www.svc.org/clientuploads/directory/resource_library/10_608.pdf.

[75] A. Ryer. *The Light Measurement Handbook*. Tech. rep. International Light Technologies, 1997. url: https://cgvr.informatik.uni-bremen.de/teaching/cg_literatur/ILT-Light-Measurement-Handbook.pdf.

[76] S. Radu, M. Uludag, S. Speretta, J. Bouwmeester, A. Dunn, T. Walkinshaw, P. K. D. Cas, and C. Cappelletti. *The PocketQube Standard*. https://www.albaorbital.com/pocketqube-standard. Standard Report, Accessed: 2025-05-24. 2018.

[77] J. P. Mason, B. Lamprecht, T. N. Woods, and C. Downs. “CubeSat On-Orbit Temperature Comparison to Thermal-Balance-Tuned-Model Predictions.” In: *Journal of Thermophysics and Heat Transfer* 32.1 (2018), pp. 237–255. doi: 10.2514/1.T5169.

[78] M. Holliday. *PyCubed GitHub repository forums*. https://github.com/pycubed/forums/discussions/87. Accessed: 2025-05-24. 2024.

[79] D. Vallado. *Fundamentals of Astrodynamics and Applications, 4th edition*. Space technology library. Microcosm Press, 2013. isbn: 978-1881883180.

[80] P. Alken et al. “International Geomagnetic Reference Field: the thirteenth generation.” In: *Earth, Planets and Space* 73.1 (2021), p. 49. doi: 10.1186/s40623-020-01288-x.

[81] W. Brown. *IGRF model for Matlab*. https://github.com/wb-bgs/m_IGRF. Github Repository, Accessed: 2025-04-24.

[82] British Geological Survey. *International Geomagnetic Reference Field*. https://geomag.bgs.ac.uk/research/modelling/IGRF.html. Accessed: 2024-06-03. 2024.

[83] Lighthouse3d. *Ray-Sphere Intersection*. http://www.lighthouse3d.com/tutorials/maths/ray-sphere-intersection/. Website, Accessed: 2025-04-24.

[84] Q. Lam, N. Stamatakos, C. Woodruff, and S. Ashton. “Gyro Modeling and Estimation of Its Random Noise Sources.” In: *AIAA Guidance, Navigation, and Control Conference and Exhibit*. 2003. doi: 10.2514/6.2003-5562.

[85] J. Tuthill. “Design and Simulation of a Nano-satellite Attitude Determination System.” Master’s thesis. Naval Postgraduate School, Monterey, California, 2009.

[86] MathWorks. *Inertial Sensor Noise Analysis Using Allan Variance*. Accessed: 2025-03-05. n.d. url: https://www.mathworks.com/help/nav/ug/inertial-sensor-noise-analysis-using-allan-variance.html.

[87] D. Rondao. "Modeling and Simulation of the ECOSat-III Attitude Determination and Control System." Master's thesis. Instituto Superior Técnico, 2016. url: https://api.semanticscholar.org/CorpusID:212414103.

[88] G. Avanzini and F. Giulietti. "Magnetic detumbling of a rigid spacecraft." In: *Journal of Guidance, Control, and Dynamics* 35 (4 2012), pp. 1326–1334. issn: 15333884. doi: 10.2514/1.53074.

[89] R. Schneider and C. Georgakis. "How To NOT Make the Extended Kalman Filter Fail." In: *Industrial & Engineering Chemistry Research* 52.9 (2013), pp. 3354–3362. doi: 10.1021/ie300415d.

[90] A. Porras-Hermoso, D. Alfonso-Corcuera, J. Piqueras, E. Roibás-Millán, J. Cubas, J. Pérez-Álvarez, and S. Pindado. "Design, Ground Testing and On-Orbit Performance of a Sun Sensor Based on COTS Photodiodes for the UPMSat-2 Satellite." In: *Sensors* 21.14 (2021). doi: 10.3390/s21144905.

[91] Texas Instruments. *OPT3001: Ambient Light Sensor Application Guide*. https://www.ti.com/lit/an/sbea002a/sbea002a.pdf?ts=1748715535462. Application Report, Revision A, Accessed: 2025-04-24. 2017.

[92] M. Holliday, Z. Manchester, and D. G. Senesky. "On-Orbit Implementation of Discrete Isolation Schemes for Improved Reliability of Serial Communication Buses." In: *IEEE Transactions on Aerospace and Electronic Systems* 58.4 (2022), pp. 2973–2982. doi: 10.1109/TAES.2022.3142713.

[93] H. R. Haave. "Simulating Sun Vector Estimation and Finding Gyroscopes for the NUTS Project." Master's thesis. Trondheim, Norway: Norwegian University of Science and Technology, 2016. url: https://ntnuopen.ntnu.no/ntnu-xmlui/handle/11250/2399222.

[94] Copernicus Climate Change Service, Climate Data Store. *Surface albedo 10-daily gridded data from 1981 to present*. Accessed: 2025-05-24. 2018. doi: 10.24381/cds.ea87ed30.

[95] Y. Ohno. *OSA Handbook of Optics*. 2nd ed. Vol. 3. The McGraw-Hill Companies, Inc., 2001. Chap. 14. isbn: 9780071498913.

[96] F. L. Pedrotti, L. M. Pedrotti, and L. S. Pedrotti. *Introduction to Optics*. 3rd ed. Cambridge University Press, 2017. Chap. 19. isbn: 9781108563321.

[97] P. R. Michael, D. E. Johnston, and W. Moreno. “A conversion guide: solar irradiance and lux illuminance.” In: *Journal of Measurements in Engineering* 8.4 (2020), pp. 153–166. doi: 10.21595/jme.2020.21667.

Appendix A1

From radiometry to photometry

This appendix was added to clarify the difference between radiometry and photometry and, based on those distinct concepts, determine the expected illuminance response of OPT4001 transducers in orbit to the Sun's AM0 irradiance of 1366.1 W/m^2.

Radiometry is the measurement of electromagnetic radiation, which ranges from 10 nm to 1000 m wavelengths, and includes the regions commonly referred to as the ultraviolet, the visible, and the infrared. Photometry, on the other hand, is the measurement of light, which is defined as electromagnetic radiation detectable by the human eye. It is thus restricted to the visible region of the spectrum, with wavelengths ranging from 360 nm to 830 nm, and all the quantities are weighted by the spectral response of the eye [95]. Whereas radiometry involves a purely physical measurement, photometry involves a stimulus response to radiant energy [96]. Photometry uses either optical radiation detectors constructed to mimic the spectral response of the eye, as is the selected transducer, or spectroradiometry coupled with appropriate calculations for weighting by the spectral response of the eye [95].

A summary of the radiometric quantities and units, along with their corresponding photometric counterparts, is given in Table 17. Note that all the photometric quantities are weighted by the spectral response of the eye. For example, light whose wavelength is well outside the visible spectrum has a luminous energy of zero, regardless of the amount of radiant energy present.

Table 17: Radiometric quantities and its photometric equivalents (adapted from [96]).

Radiometric quantity	Photometric quantity	Description
Radiant energy [J *or* W.s]	Luminous energy [talbot *or* lm.s]	Total amount of light energy emitted or received over time.
Radiant flux [W]	Luminous flux [lm]	Total rate of light energy emitted, transferred, or received.
Radiant exitance [W/m^2]	Luminous exitance [lm/m^2]	Light power or brightness arriving at a surface from all directions.
Irradiance [W/m^2]	Illuminance [lm/m^2 *or* lx]	Light power or brightness leaving a surface in all directions.
Radiant intensity [W/sr]	Luminous intensity [lm/sr *or* cd]	Light flux emitted per unit solid angle in a specific direction.
Radiance [W/m^2.sr]	Luminance [cd/m^2]	Represents how bright a surface appears from a direction.

Since not all human eyes are identical, a standard response has been determined by the International Commission on Illumination (CIE). It is reproduced in Figure 90, which shows the established luminous efficiency of the eye for photopic vision, in other words, the relative sensation of brightness plotted against wavelength [96]. Radiometric quantities are related to photometric quantities through this curve.

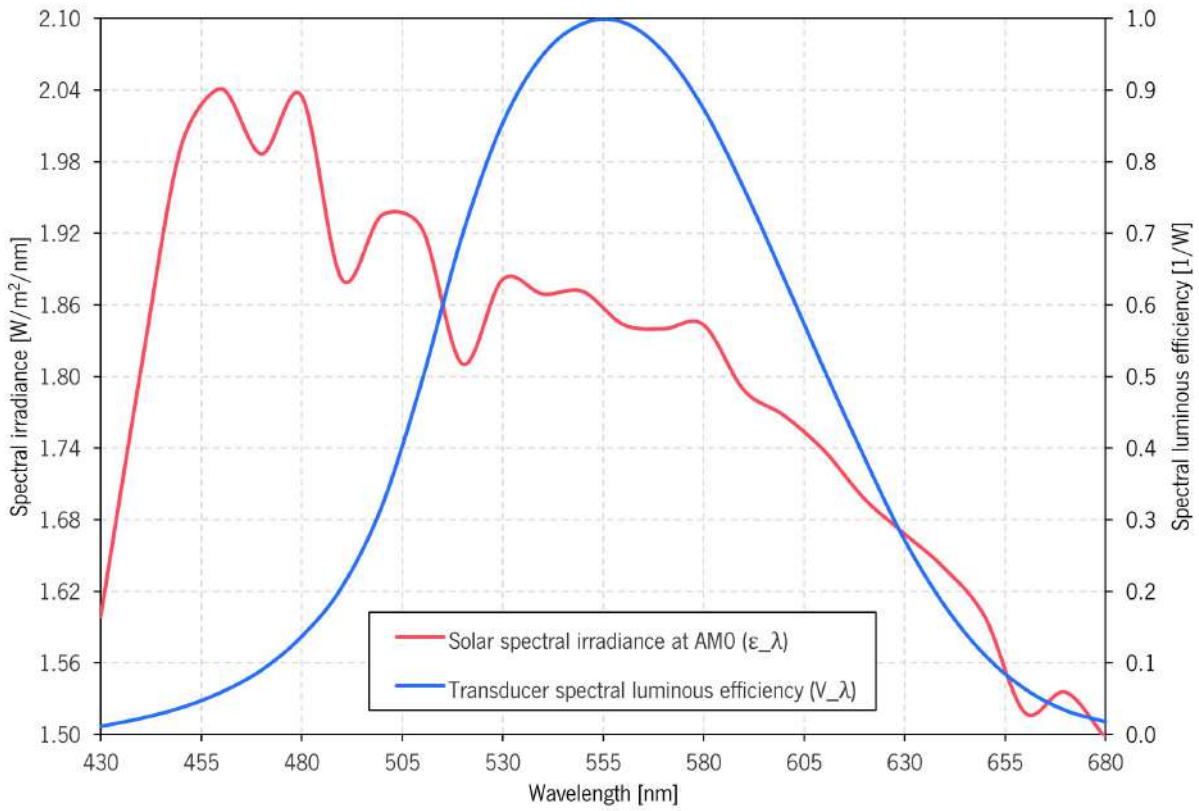


Figure 90: Luminous efficiency and solar irradiance evolution over the visible spectrum.

In general, for monochromatic radiation of wavelength λ, analogous units are related by

$$ph = K(\lambda) \cdot rd \tag{70a}$$

$$K(\lambda) = 685 \cdot V(\lambda) \tag{70b}$$

where ph and rd are any photometric and radiometric quantities, respectively, $K(\lambda)$ is luminous efficacy determined with the luminous efficiency $V(\lambda)$, obtained from the CIE curve [96]. The proportionality constant in Equation (70b) is defined such that a radiant flux of 1 W at the wavelength of 555 nm, where the luminous efficiency reaches its maximum, corresponds to a luminous flux of 685 lm [96].

When the radiation consists of a spread of wavelengths, such as the one emitted by the Sun, the radiometric and photometric terms are functions of wavelength [96]. This dependence is indicated by preceding the term with the word "spectral" and by using a subscript $\square_\lambda$. The total photometric quantity is then determined by integration over the wavelength region of interest with

$$ph = \int_{\lambda_1}^{\lambda_2} ph_\lambda(\lambda) d\lambda \tag{71}$$

As stated at the beginning, the goal was to estimate the illuminance measured by the selected transducer when AM0 irradiance is incident. To achieve this, Equations (70) and (71) can be reformulated to this purpose [97], which give

$$E = \int_{\lambda_1}^{\lambda_2} E_\lambda(\lambda) d\lambda \tag{72a}$$

$$E_\lambda(\lambda) = 685 \cdot V_\lambda(\lambda) \cdot \varepsilon_\lambda(\lambda) \tag{72b}$$

where E is the total illuminance, E_λ is the spectral illuminance and ε_λ is the spectral Sun's irradiance at AM0.

The coefficients needed for Equation (72b) were collected from two standard documents. The solar spectral AM0 irradiance at each wavelength, ε_λ, was gathered from ASTM E-490, while the spectral luminous efficiency, V_λ, was retrieved from ISO/CIE 23539.2005. Again, while the latter is related to the human eye, the OPT4001 follows a very similar response. These coefficients were plotted in Figure 90.

Equation (72a) was performed numerically over the wavelength range from 430 nm to 680 nm, where the efficiency exceeds 1%, using a 10 nm step average. This approach is an approximation since: the spectral luminous efficiency curve is slightly different from the OPT4001 response on certain regions, as shown in Figure 14a; the solar spectral irradiance coefficients represent an average value of the Sun's intensity, not accounting for its dynamic events; and the selected standards provide coefficients up to 1 nm intervals, so their resolution could be increased.

Finally, the obtained illuminance was 133,531 lx.

Appendix A2

Advanced prototype remaining illuminance results

This appendix displays the remaining plots not present in Section 4.5 due to formatting constraints. These are specifically the illuminance evolution over different incidence angles of the second and third assembly of the advanced prototype.

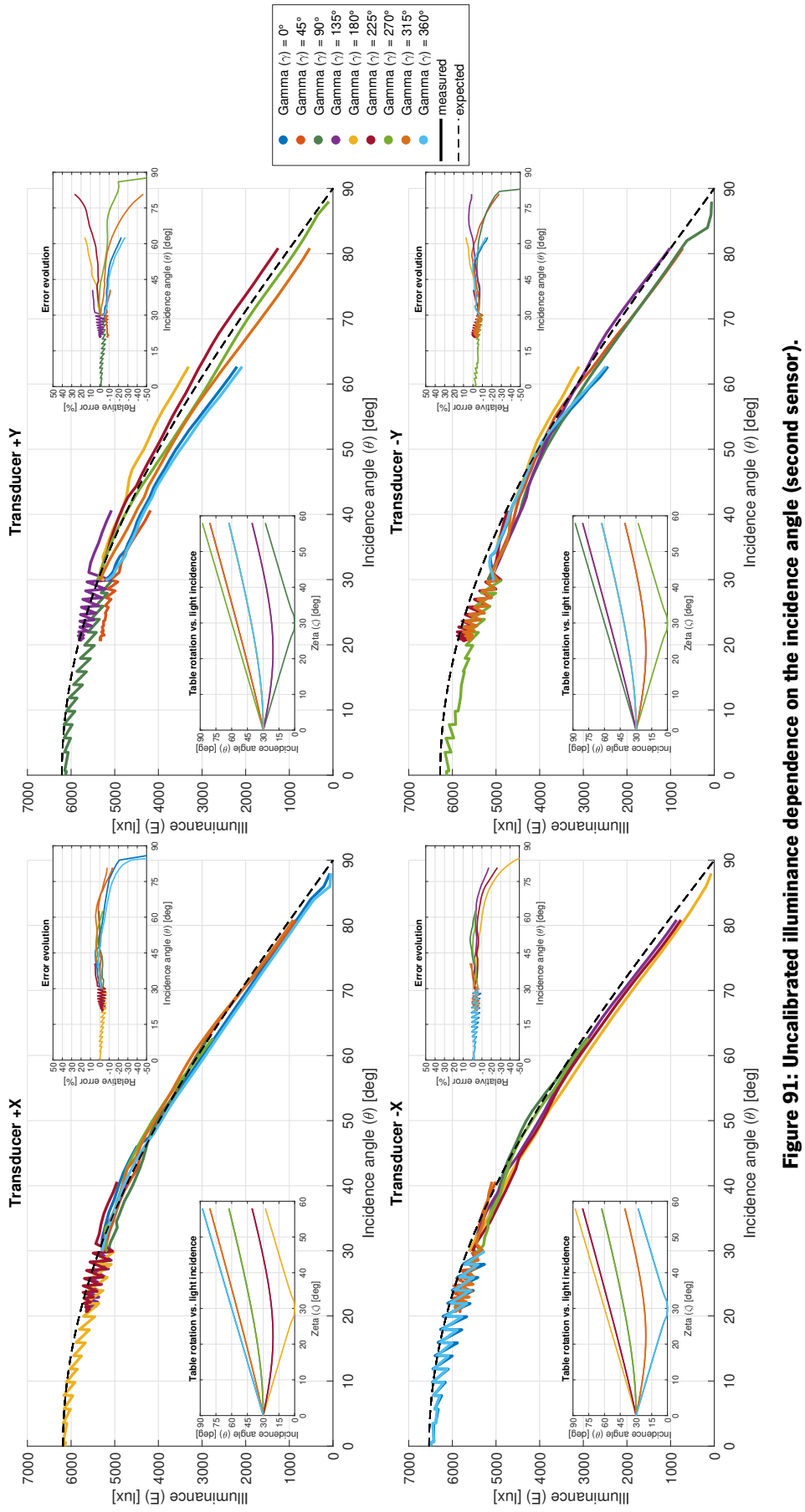


Figure 91: Uncalibrated illuminance dependence on the incidence angle (second sensor).

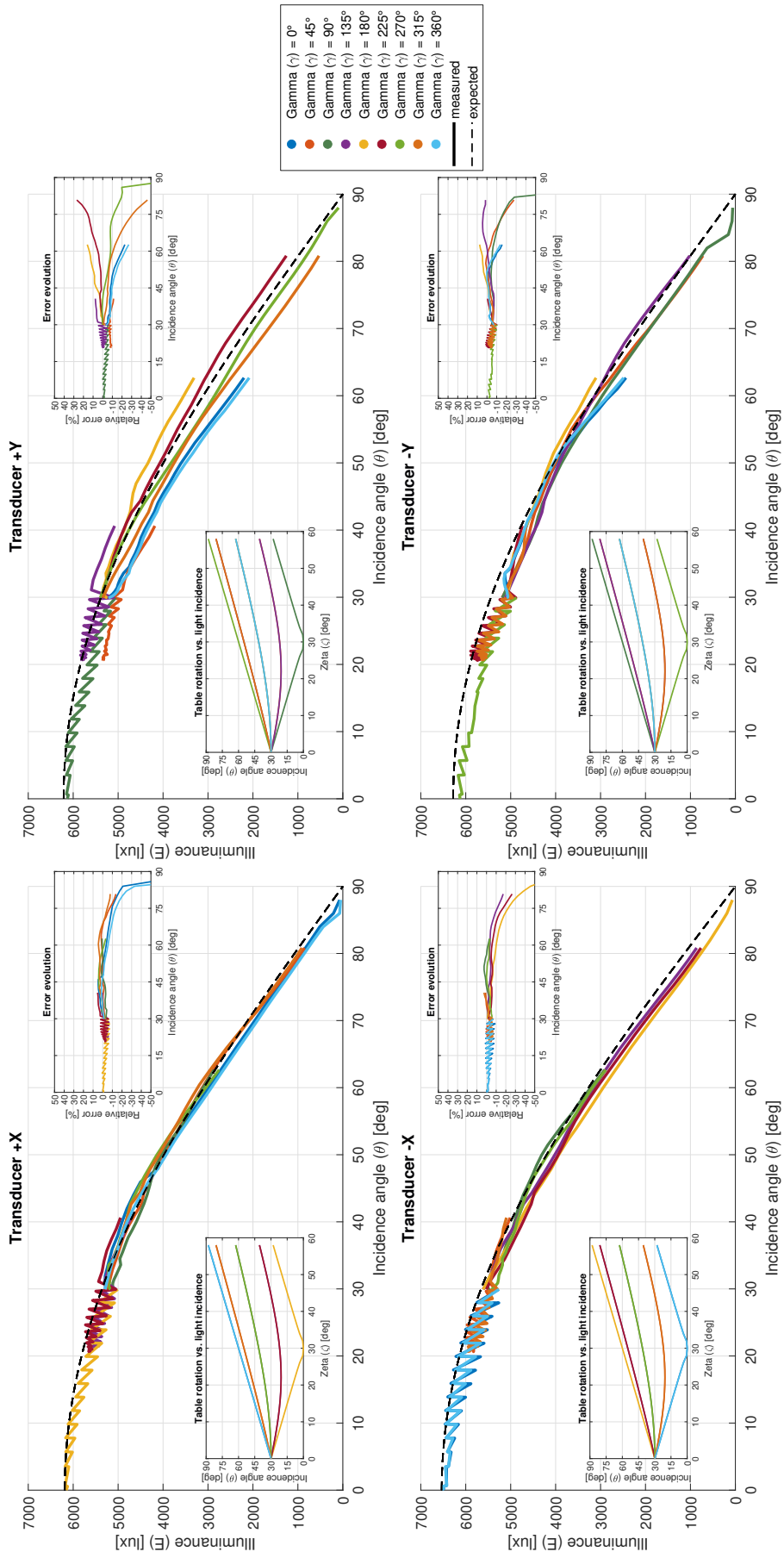


Figure 92: Illuminance dependence on the incidence angle after light evolution calibration (second sensor).

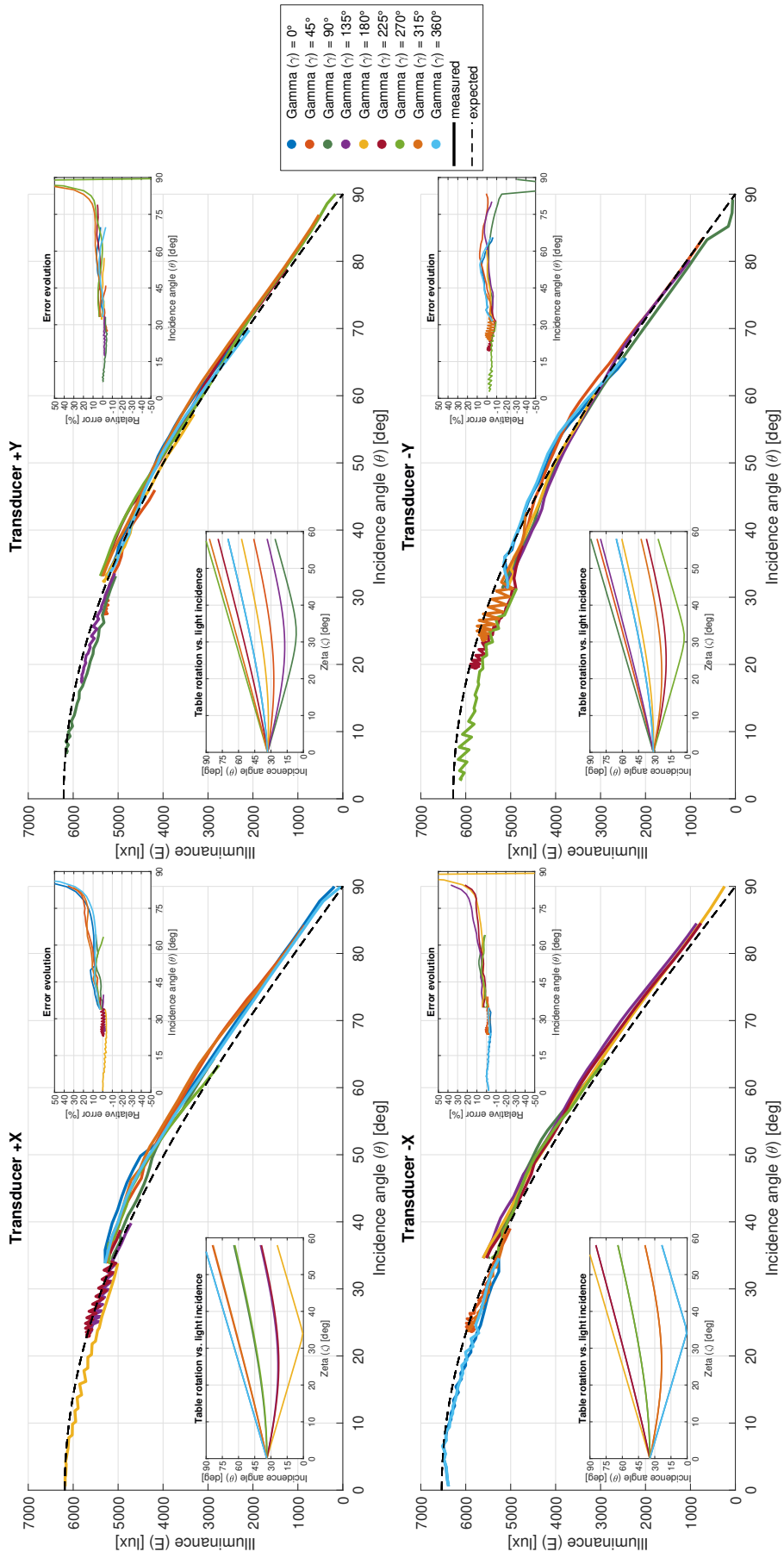


Figure 93: Illuminance dependence on the incidence angle after normal vectors calibration (second sensor).

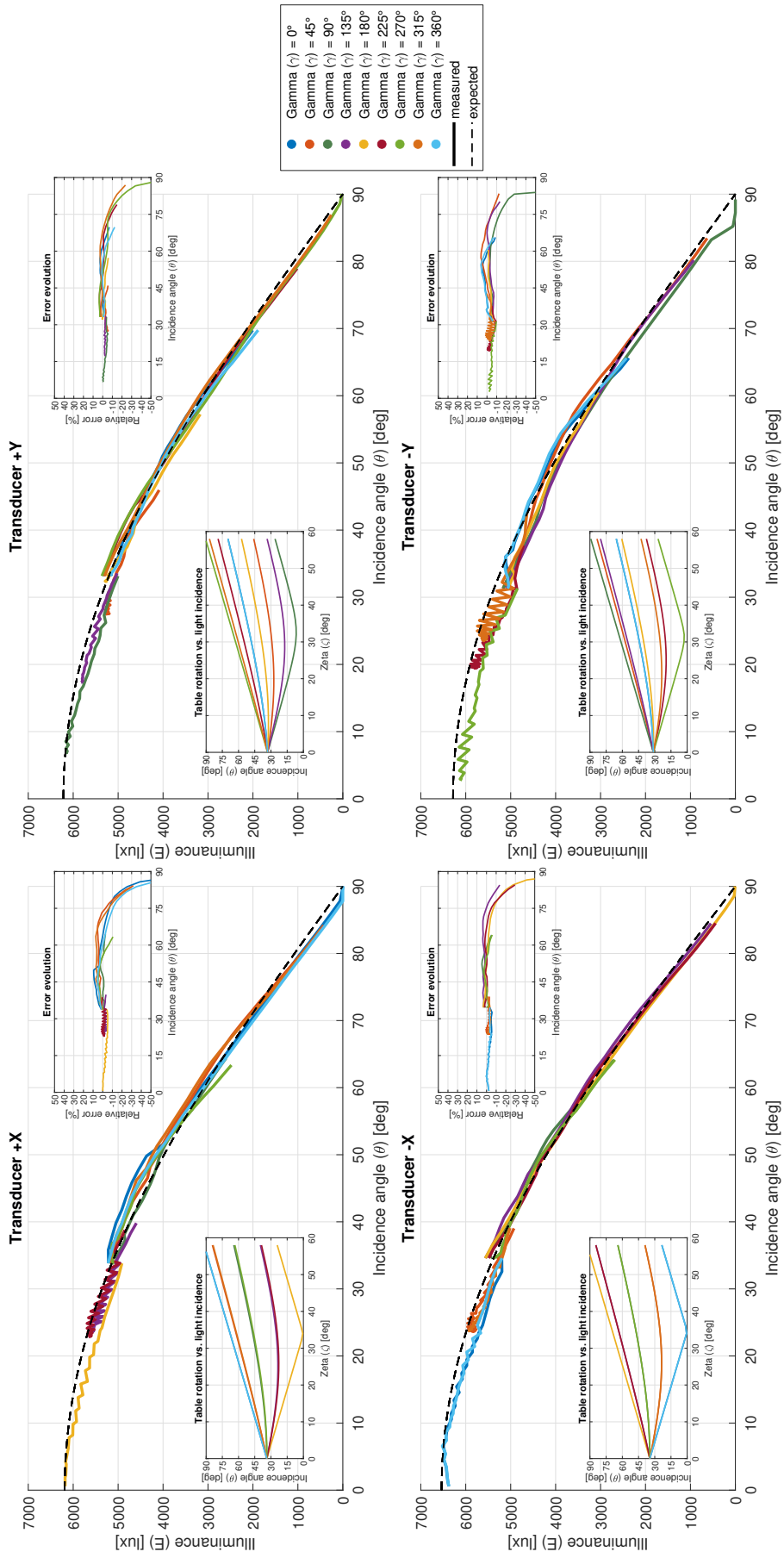


Figure 94: Illuminance dependence on the incidence angle after characteristics calibration (second sensor).

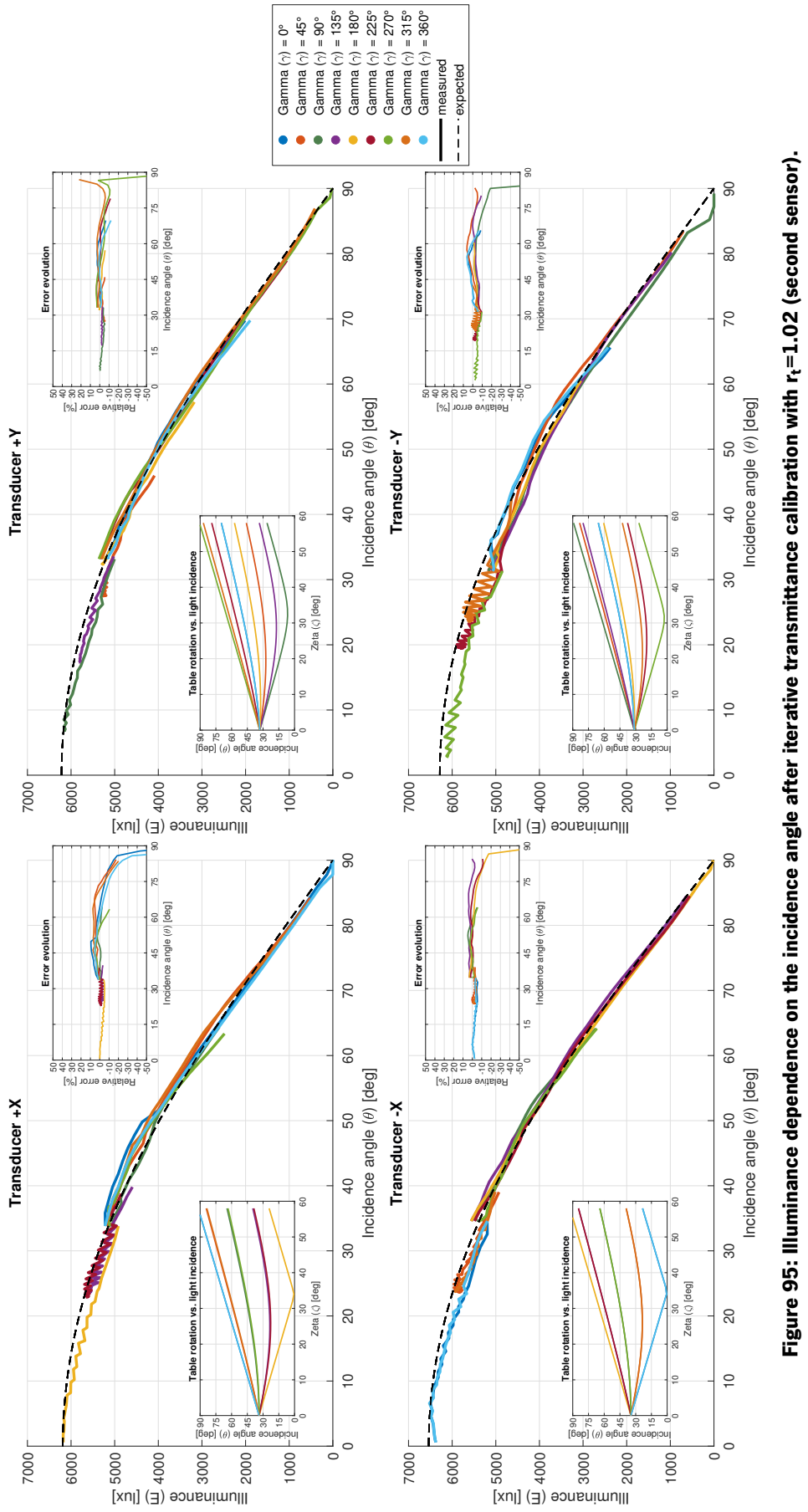


Figure 95: Illuminance dependence on the incidence angle after iterative transmittance calibration with r_t=1.02 (second sensor).

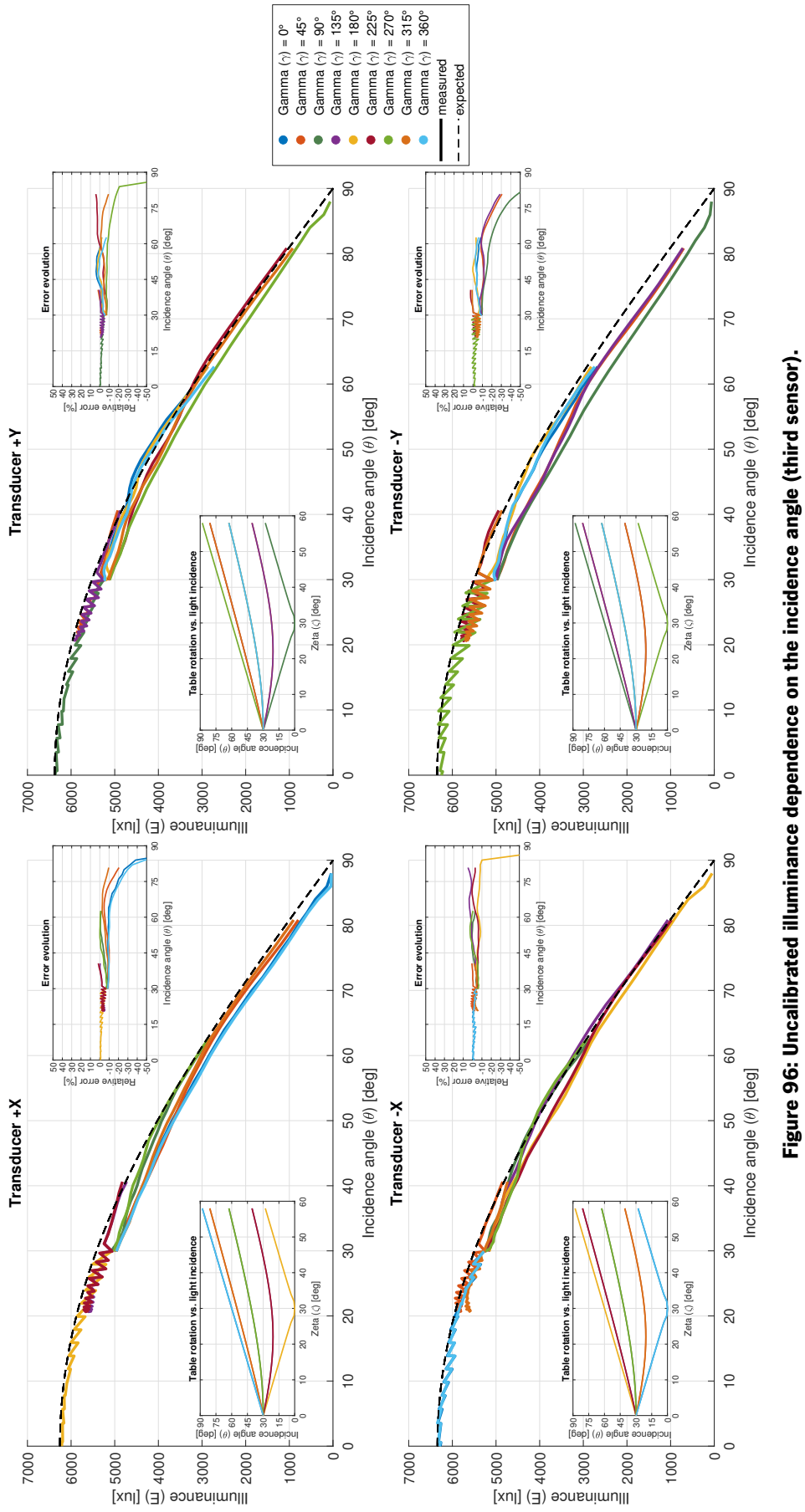


Figure 96: Uncalibrated illuminance dependence on the incidence angle (third sensor).

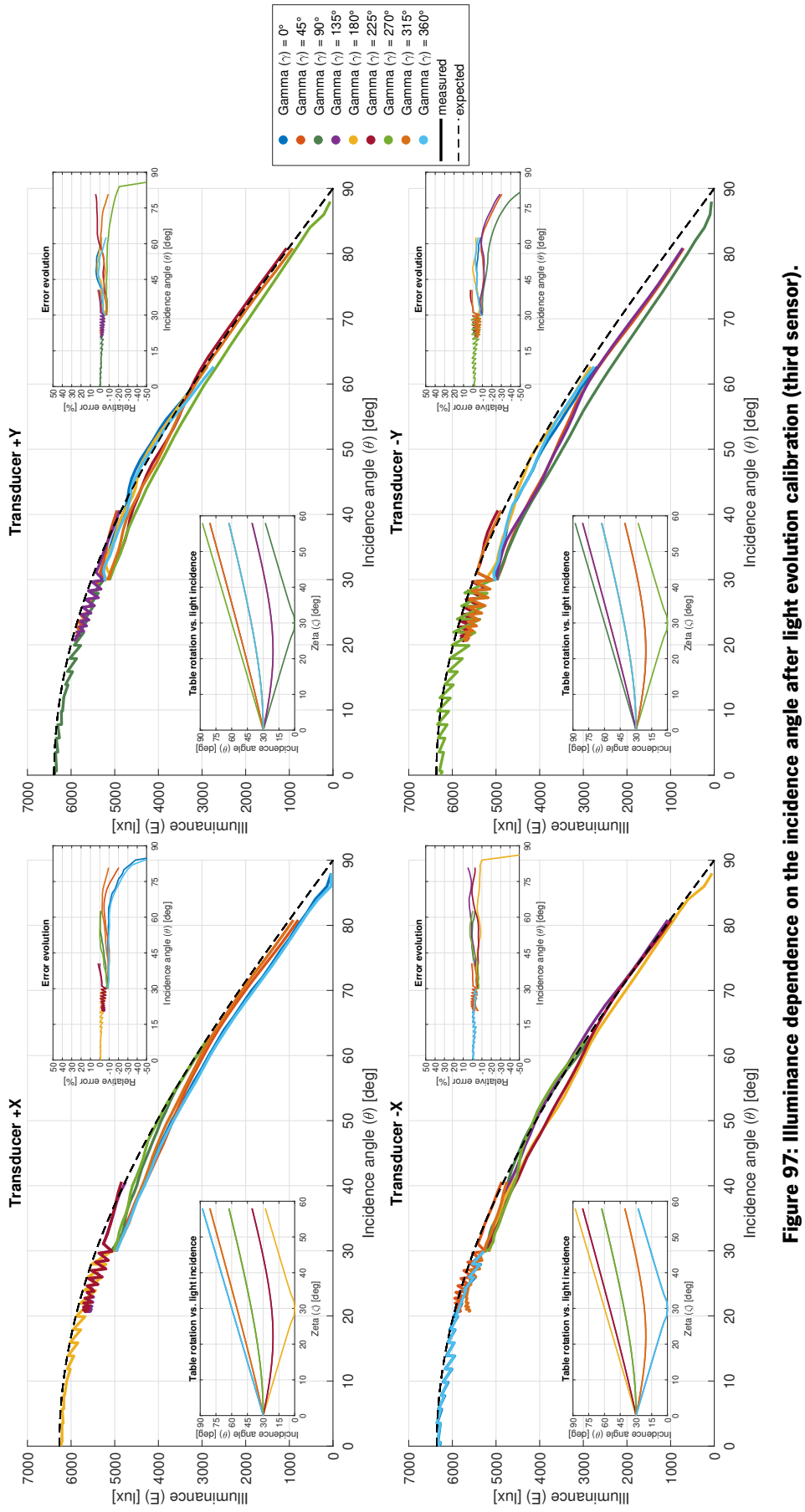


Figure 97: Illuminance dependence on the incidence angle after light evolution calibration (third sensor).

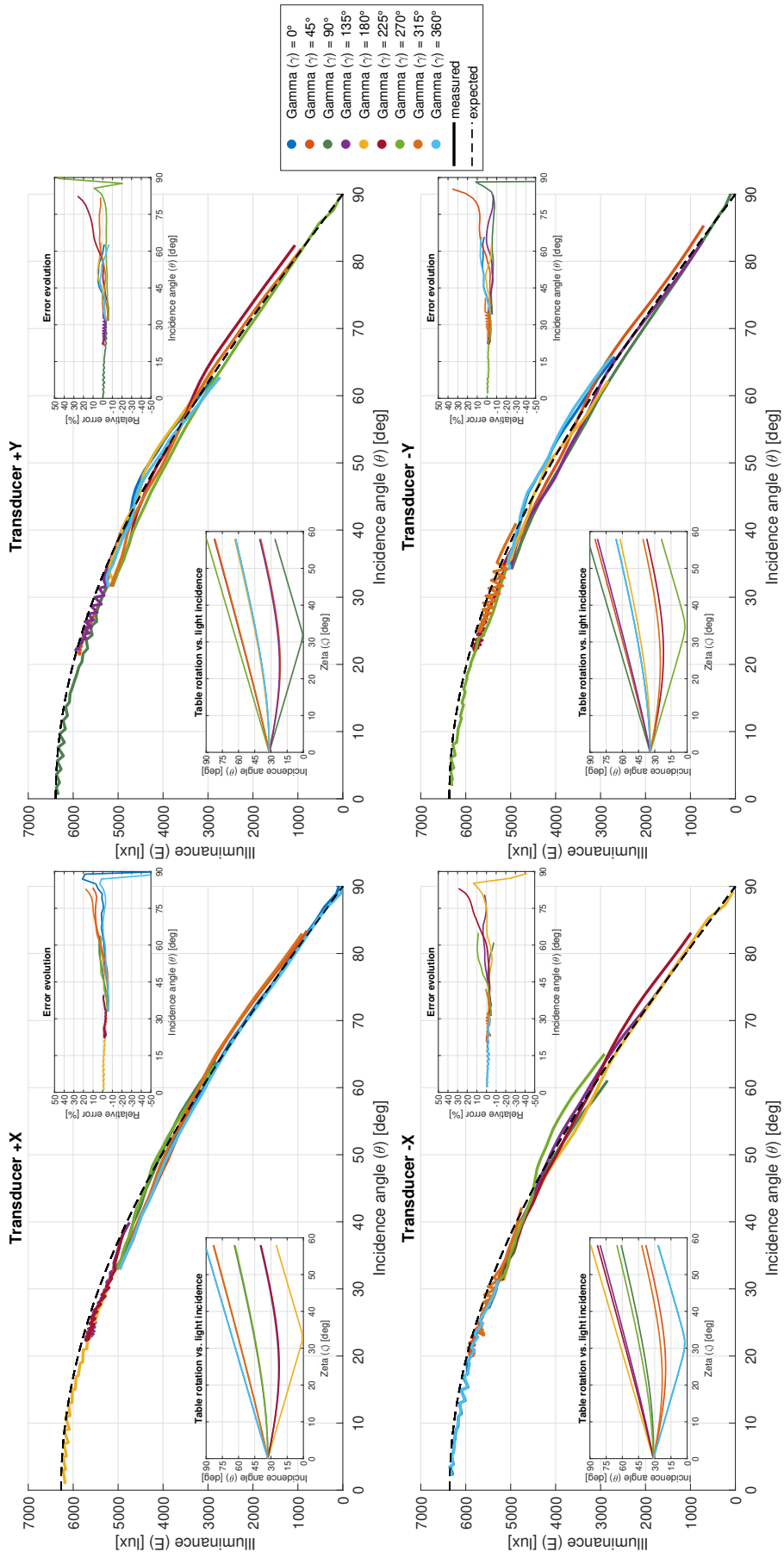


Figure 98: Illuminance dependence on the incidence angle after normal vectors calibration (third sensor).

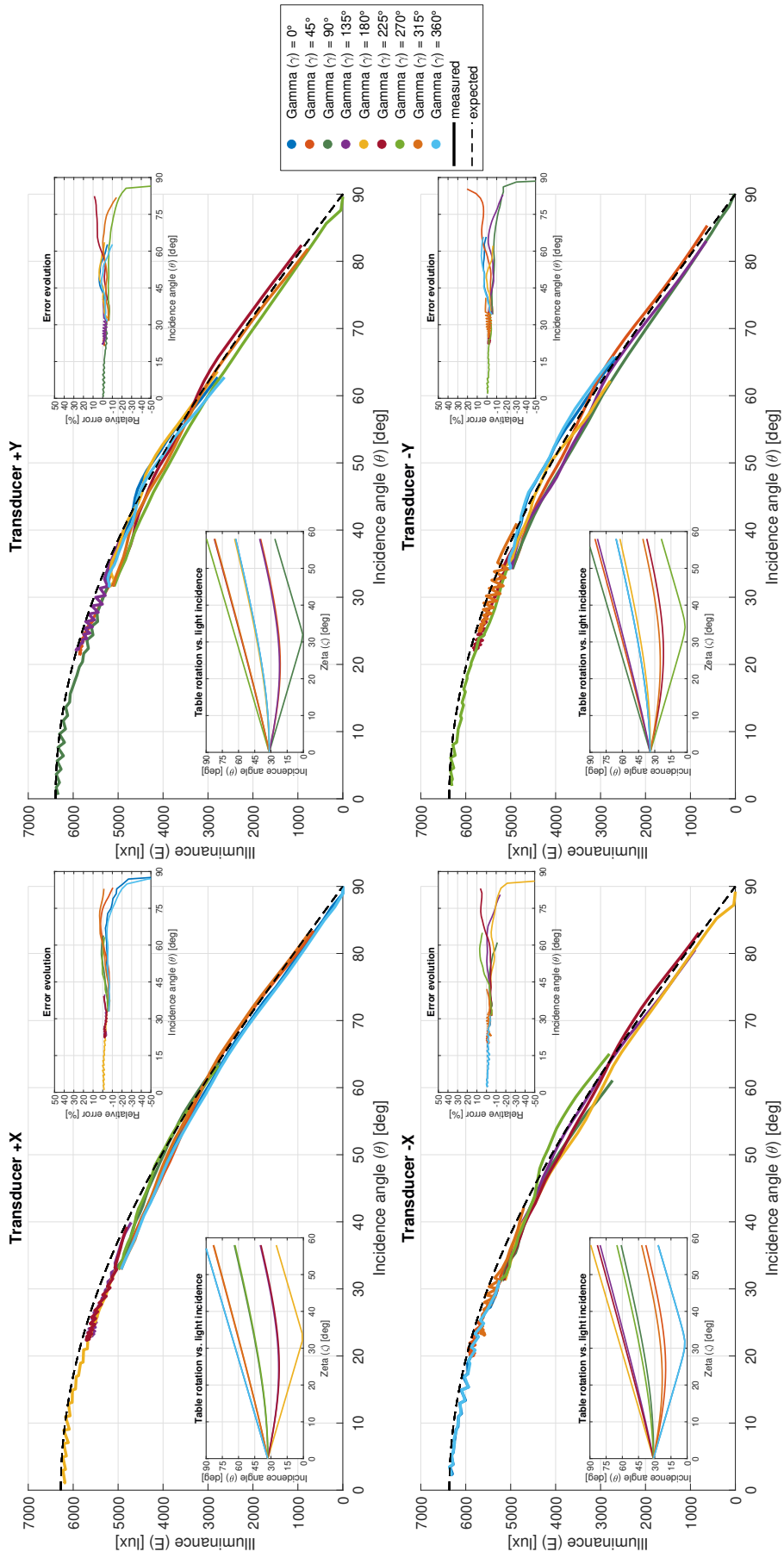


Figure 99: Illuminance dependence on the incidence angle after characteristics calibration (third sensor).

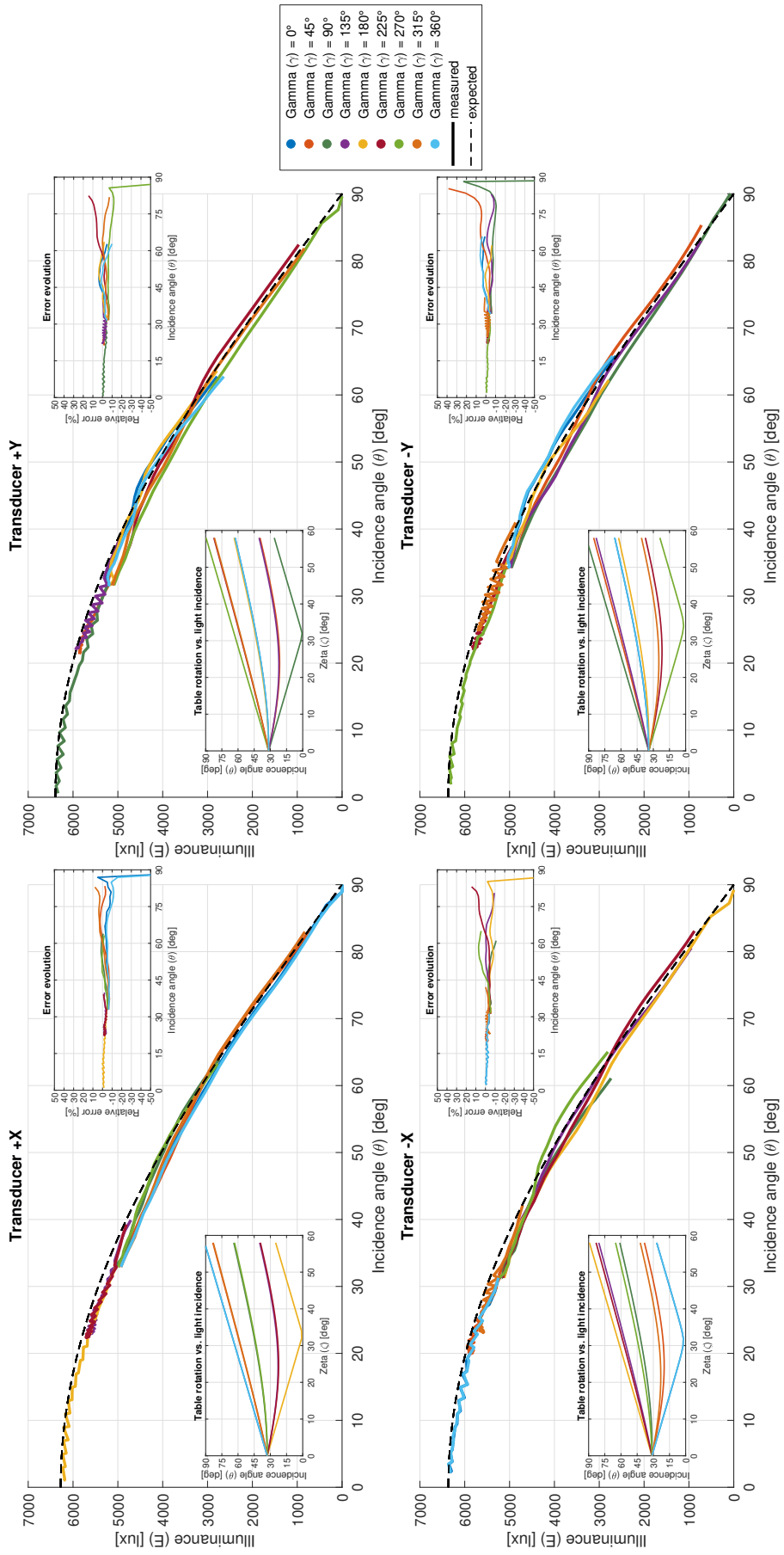


Figure 100: Illuminance dependence on the incidence angle after iterative transmittance calibration with r_t=1.02 (third sensor).