



UNIVERSIDAD DE CONCEPCIÓN
DIRECCIÓN DE POSTGRADO

Facultad de Ingeniería - Programa de Magíster en Ciencias de la ingeniería mención ingeniería
mecánica

**Diseño e implementación de una suite de optimización
preliminar del sistema ADCS en CubeSats de
observación terrestre en órbitas bajas.**

Tesis para optar al grado de
Magíster en Ciencias de la Ingeniería mención Ingeniería Mecánica

Matías Ignacio Tacul Vargas
CONCEPCIÓN - CHILE
Agosto - 2025

Profesor Guía: Dr. Félix Leaman W.
Dpto. de Ingeniería Mecánica
Universidad de Concepción

Resumen Ejecutivo

Los CubeSats, nanosatélites de bajo costo, se han convertido en una herramienta ampliamente utilizada en diversas aplicaciones espaciales, especialmente en misiones de observación terrestre. En este tipo de misiones, es fundamental orientar con precisión la carga útil hacia la superficie terrestre para garantizar la calidad de los datos recolectados. Sin embargo, tanto el rendimiento como el costo de estos satélites dependen en gran medida de las características de los sensores y actuadores que los componen. En este contexto, el presente estudio aborda el desafío de identificar el conjunto óptimo de componentes del sistema de determinación y control de actitud (ADCS), ajustándose a los requisitos específicos de cada misión.

Con este propósito, se desarrolló una suite de simulación que modela tanto la dinámica orbital como la actitud del CubeSat, integrando sensores y actuadores en el Subsistema de Determinación y Control de Actitud (ADCS). Esta suite incorpora algoritmos de estimación, como una variante del filtro de Kalman extendido, y controladores LQR o PD según requerimiento. Se diseñó una suite de optimización capaz de encontrar un equilibrio entre costo y rendimiento, considerando las restricciones impuestas por los System Engineering envelopes (como masa y potencia) y los Measures of Performance (MoP) definidos por el usuario.

Dado que el problema de optimización no es convexo y presenta múltiples mínimos locales, se evaluaron diversos algoritmos, destacando el método de Powell como el más robusto. Este algoritmo, implementado a través del módulo `scipy.optimize` en Python, demostró una alta capacidad para alcanzar mínimos globales sin requerir derivadas explícitas. Como demostración, se ejecutó un caso práctico de optimización utilizando magnetorquers, donde se seleccionaron componentes físicos óptimos bajo restricciones técnicas representativas de la misión SUCHAI-3. El estudio reveló que para métricas como la agilidad, la autoridad del actuador tiene un impacto crítico, mientras que para métricas como el jitter o la exactitud, la precisión de los sensores es más determinante.

Los resultados indicaron que el diseño óptimo de un ADCS debe balancear cuidadosamente los SE Envelopes y los MoP según el objetivo de la misión, siendo posible, mediante esta metodología, seleccionar componentes específicos en función de su contribución al rendimiento general del sistema. Esta herramienta será evaluada a futuro como un optimizador no solo aplicable a CubeSats, sino también, con las adaptaciones necesarias, como una solución para otros tipos de vehículos aéreos, como los RPAS, incorporando cambios o mejoras al algoritmo de optimización según los requerimientos de cada plataforma.

Abstract

CubeSats, low-cost nanosatellites, have become widely used tools in various space applications, particularly in Earth observation missions. In such missions, accurately pointing the payload toward Earth is essential to ensure the quality of the collected data. However, both the performance and cost of these satellites largely depend on the characteristics of their sensors and actuators. In this context, the present study addresses the challenge of identifying the optimal set of components for the Attitude Determination and Control System (ADCS), tailored to the specific requirements of each mission.

To this end, a simulation suite was developed to model both the orbital and attitude dynamics of the CubeSat, integrating sensors and actuators within the ADCS. This suite incorporates estimation algorithms, such as a variant of the Extended Kalman Filter, as well as LQR or PD controllers, depending on the requirements. An optimization framework was designed to balance cost and performance, considering constraints imposed by System Engineering (SE) envelopes—such as mass and power—and user-defined Measures of Performance (MoP).

Since the optimization problem is non-convex and presents multiple local minima, several algorithms were tested, with the Powell method standing out as the most robust. This algorithm, implemented via the `scipy.optimize` module in Python, demonstrated strong capabilities for finding global minima without requiring explicit derivatives. As a demonstration, a practical optimization case was executed using magnetorquers, where optimal physical components were selected under technical constraints representative of the SUCHAI-3 mission. The study revealed that actuator authority plays a critical role in agility-related metrics, while sensor accuracy is more influential for metrics such as jitter and pointing accuracy.

The results indicated that the optimal design of an ADCS must carefully balance SE envelopes and MoP according to the mission objective. Using this methodology, it is possible to select specific components based on their contribution to the overall system performance. This tool will be evaluated in the future not only as an optimizer for CubeSats, but also—after the necessary adaptations—as a solution for other types of aerial vehicles, such as RPAS, incorporating algorithmic improvements as required by each platform.

Contenidos

Lista de Figuras	1
Lista de Tablas	4
Nomenclatura	6
1 Introducción	7
1.1 Hipótesis	9
1.2 Objetivos	10
1.3 Limitaciones de diseño	11
1.4 Metodología	11
1.5 Carta Gantt	12
2 Marco Teórico	13
2.1 Sistemas de referencia	13
2.2 Dinámica orbital	14
2.2.1 Parámetros orbitales	15
2.2.2 Perturbaciones presentes en LEO	16
2.3 Cinemática y dinámica de actitud	17
2.3.1 Actitud de un satélite y sus representaciones	17
2.3.2 Cuaterniones y cinemática de cuaterniones	19
2.3.3 Dinámica de actitud	20
2.4 Subsistema de determinación y control de actitud	20
2.4.1 Navigation (Sensores)	21
2.4.2 Guidance (Algoritmos)	21
2.4.3 Control (Controladores y actuadores)	24
2.5 System Engineering Envelopes	25
2.6 Controlabilidad y observabilidad de un sistema de control	27
2.7 Optimización Matemática	27
2.7.1 Clasificación de problemas de optimización	28
2.7.2 Optimizadores en Python	29
3 Estado del Arte	30
3.1 Spacecraft Control Toolbox	30
3.2 Ansys Systems Tool Kit (STK)	32
3.3 Aerospace Blockset	32
3.4 Valispace	35
3.5 Basilisk	35
3.6 Resumen de los simuladores disponibles	36
3.7 Métodos para la selección de componentes del ADCS	37
4 Diseño de la suite de simulación	39
4.1 Marco general del simulador	39
4.2 Propagador orbital	40
4.3 Modelación de sensores	41

4.3.1	Vector sol	41
4.3.2	Fuerzas geomagnéticas de la tierra	44
4.3.3	Cambio en sistema de referencia y modelación	45
4.4	Algoritmos de estimación y control satelital	48
4.4.1	Controladores y actuadores	48
4.4.2	Utilización del Filtro de Kalman Extendido (EKF)	52
4.5	Estructura general de la suite de simulación de actitud	54
5	Verificación de la suite de simulación	56
5.1	Resultados de estimación y control	56
5.1.1	Validación de la estimación con el SEKF	56
5.1.2	Comparación entre referencia y estimación en control	57
5.2	Cuantificación de los MoP de Apuntamiento	58
5.3	Condiciones y parámetros de simulación	60
5.4	Resultados suite de simulación	61
5.4.1	Resultados tipos de actuadores	61
5.4.2	Resultados de los controladores	62
5.4.3	Resultados niveles de sensores	64
5.4.4	Resultados niveles de actuadores	66
5.5	Resumen sobre análisis realizados	68
6	Modelo de optimización	70
6.1	Funciones de optimización para los parámetros de rendimiento en actitud	70
6.2	Funciones de optimización para recursos físicos	71
6.3	Selección de línea base mediante matriz de decisión	73
6.4	Pruebas de convexidad en scipy.optimize	74
6.4.1	Análisis caso de optimización simple.	74
6.4.2	Análisis caso de optimización complejo.	77
6.4.3	Análisis de convexidad y elección del método de optimización	79
6.5	Ejemplo de optimización utilizando magnetorquer	80
7	Conclusiones	85
	Anexo A Carta Gantt	93
	Anexo B Torques externos debido a las perturbaciones	95
	Anexo C Cambios en los sistemas de referencia	97
	Anexo D Control lineal Magnetorquer	98
	Anexo E Dinámica de seis estados del sistema con ruedas de reacción	101
	Anexo F Componentes utilizados para Capitulo 5	105
F.1	Giroscopios	105
F.2	Magnetometros	105
F.3	Sensores de sol	106

F.4	Magnetorquers	107
F.5	Ruedas de reacción	107
Anexo G	Implementación de los resultados de los MoP de apuntamiento	109
Anexo H	Código en Python	114

Lista de Figuras

Figura 1.1	Cantidad de nanosatélites lanzados a través de los años [1].	7
Figura 1.2	Representación gráfica de la relación entre aspectos de la misión respecto a costo y rendimiento (Elaboración propia).	8
Figura 1.3	Solución propuesta para la suite de simulación optimizada (Elaboración propia).	10
Figura 2.1	Marco de referencia del cuerpo (Elaboración propia).	13
Figura 2.2	Marco de referencia inercial ECI [29].	14
Figura 2.3	Marco de referencia inercial RPY [30].	14
Figura 2.4	Elementos keplerianos [31].	15
Figura 2.5	Representación gráfica de la primera fila de la matriz de cosenos directores [34].	17
Figura 2.6	Representación de un cambio de orientación en Euler axis/angle de x [32].	18
Figura 2.7	Secuencia clásica de Euler de tres rotaciones que transforman xyz en x'y'z' [29].	18
Figura 3.1	Resumen de la dinámica orbital y de actitud del satélite en CubeSat Toolbox.	31
Figura 3.2	Ventana de control del simulador de un CubeSat en CubeSat Toolbox.	31
Figura 3.3	Imagen referencial de STK para simulación de un satélite.	32
Figura 3.4	Plantilla de Simulink para el diseño de un CubeSat.	33
Figura 3.5	Plantilla de Simulink para la modelación y simulación del CubeSat en base a la visualización en Simulink 3D.	33
Figura 3.6	Visualización del MBSE de CubeSat.	34
Figura 3.7	Perfil de CubeSat según los SE envelopes configurables.	34
Figura 3.8	Simulación de apuntamiento de la Tierra hacia la estación terrestre.	35
Figura 4.1	Esquema general de la base del simulador (Elaboración propia).	40
Figura 4.2	Propagación del SUCHAI-3 durante una órbita con fecha de inicio 01/11/23.	42
Figura 4.3	Plano eclíptico y marco de referencia ECI, con representación de la oblicuidad terrestre y equinoccios. Adaptado de [58].	43
Figura 4.4	Componentes del vector sol en marco de referencia ECI simulados durante un año (Elaboración propia).	44
Figura 4.5	Componentes de las fuerzas magnéticas respecto a ECI que afectan al SUCHAI-3 con fecha inicial 01/11/2023 (Elaboración propia).	46
Figura 4.6	Estructura general de la suite de simulación de actitud.	55
Figura 5.1	Comparación entre filtro de kalman lineal y modelo dinámico no lineal.	57
Figura 5.2	Diferencia a través del tiempo del control LQR utilizando kalman respecto a objetivo de control.	58

Figura 5.3	Ángulos de Euler LVLH y body para dos actuadores distintos (Elaboración propia).	62
Figura 5.4	Ángulos de Euler LVLH y body para dos controladores distintos (Elaboración propia).	63
Figura 5.5	Ángulos de Euler LVLH y body para niveles de sensor distintos con magnetorquer (Elaboración propia).	64
Figura 5.6	Ángulos de Euler LVLH y body para niveles de sensor distintos con rueda de reacción (Elaboración propia).	65
Figura 5.7	Ángulos de Euler LVLH y body para niveles de magnetorquer distintos con sensores nivel 2 (Elaboración propia).	67
Figura 5.8	Ángulos de Euler LVLH y body para niveles de rueda de reacción distintos con sensores nivel 2 (Elaboración propia).	68
Figura 6.1	Modelos de regresión no lineal ajustados a datos empíricos que relacionan variables técnicas de componentes con masa para magnetorquers (arriba-izquierda), ruedas de reaccion (arriba-derecha), sensores solares (abajo-izquierda) y magnetómetros (abajo-derecha). Cada curva verde corresponde a un ajuste de tipo ley de potencias.	72
Figura 6.2	Modelos de regresión no lineal ajustados a datos empíricos que relacionan variables técnicas de componentes con consumo promedio de potencia ara magnetorquers(arriba-izquierda), Ruedas de reaccion (arriba-derecha), sensores solares (abajo-izquierda) y magnetómetros (abajo-derecha). Cada curva verde corresponde a un ajuste de tipo ley de potencias.	72
Figura 6.3	Gráfico radial normalizado de las funciones de costo f_i para cada nivel de componentes (Bajo, Medio, Alto).	74
Figura 6.4	Grilla 20x20 caso de estudio optimizando exactitud de apuntamiento con los SE envelopes de los sensores.	75
Figura 6.5	Grilla 20x20x6 caso de estudio utilizando todas las funciones de costo para la optimización.	77
Figura 6.6	Primer caso de estudio: optimización de componentes físicos seleccionando la exactitud como MoP.	81
Figura 6.7	Segundo caso de estudio: optimización de componentes físicos seleccionando la agilidad como MoP.	81
Figura 6.8	Tercer caso de estudio: optimización de componentes físicos seleccionando el jitter como MoP.	82
Figura 6.9	Gráfico de radar normalizado de las funciones de costo f_i para cada MoP de los componentes, respecto a componentes de nivel bajo y alto; Arriba a la izquierda: optimización de exactitud de apuntamiento, Arriba a la derecha: optimización de tiempo de asentamiento y Abajo: optimización de jitter.	84

Figura A.1 Carta Gantt parte 1.	93
Figura A.2 Carta Gantt parte 2.	94
Figura G.1 Filtro pasa alto de 10 Hz en la respuesta al sistema de sensores y actuadores de nivel bajo.	109
Figura G.2 Densidad espectro potencia en ancho de banda seleccionada para Roll.	110
Figura G.3 Densidad espectro potencia en ancho de banda seleccionada para Pitch.	110
Figura G.4 Densidad espectro potencia en ancho de banda seleccionada para Yaw.	111
Figura G.5 Banda de asentamiento Roll en modelo dinámico.	112
Figura G.6 Banda de asentamiento Pitch en modelo dinámico.	112
Figura G.7 Banda de asentamiento Yaw en modelo dinámico.	113

Lista de Tablas

Tabla 1.1	Cronograma de hitos y avances	12
Tabla 2.1	Sensores utilizados en CubeSat [6, 36].	21
Tabla 2.2	Descripción de las triadas de referencia y de observación.	22
Tabla 2.3	Actuadores utilizados en CubeSat [6].	26
Tabla 3.1	Comparativa de software/simuladores para CubeSats.	37
Tabla 3.2	Comparación entre métodos clásicos y modernos para la selección de componentes ADCS	38
Tabla 5.1	Error cuadrático medio (MSE) entre modelo no lineal y filtro SEKF	57
Tabla 5.2	Parámetros del sistema	61
Tabla 5.3	Componentes clasificados por nivel de rendimiento	61
Tabla 5.4	Rendimiento y costo para rueda de reaccion y magnetorquer en mismas condiciones.	63
Tabla 5.5	Rendimiento y costo para controlador PD y LQR en mismas condiciones. . .	64
Tabla 5.6	Rendimiento y costo para niveles de sensor con nivel 2 de magnetorquer y LQR. .	65
Tabla 5.7	Rendimiento y costo para niveles de sensor con nivel 2 de rueda de reacción y LQR.	66
Tabla 5.8	Rendimiento y costo para niveles de magnetorquer con nivel 2 de sensor y LQR. .	67
Tabla 5.9	Rendimiento y costo para niveles de rueda de reacción con nivel 2 de sensor y LQR.	68
Tabla 6.1	Costo de MoPs de apuntamiento para niveles de componentes seleccionados por MCDM.	73
Tabla 6.2	Costo de SE envelopes para niveles de componentes seleccionados por MCDM. .	73
Tabla 6.3	Resultados de optimización con distintos solvers y condiciones de incertidumbre intermedias.	76
Tabla 6.4	Resultados de optimización con distintos solvers y condiciones de incertidumbre extremas.	76
Tabla 6.5	Resultados de optimización compleja con distintos solvers y condiciones de incertidumbre intermedias.	78
Tabla 6.6	Resultados de optimización compleja con distintos solvers y condiciones de incertidumbre extremas.	78
Tabla 6.7	Parámetros de optimización	80
Tabla F.1	Tabla de giroscopios con sus respectivas características de costo.	105
Tabla F.2	Tabla de giroscopios con sus respectivas características de performance. . . .	105

Tabla F.3	Tabla de magnetómetros con sus respectivas características de costo.	105
Tabla F.4	Tabla de magnetómetros con sus respectivas características de performance. .	106
Tabla F.5	Tabla de sensores de sol con sus respectivas características de costo.	106
Tabla F.6	Tabla de sensores de sol con sus respectivas características de performance. .	106
Tabla F.7	Tabla de magnetorquers con sus respectivas características de costo.	107
Tabla F.8	Tabla de magnetorquers con sus respectivas características de performance. .	107
Tabla F.9	Tabla de ruedas de reacción con sus respectivas características de costo. . . .	107
Tabla F.10	Tabla de ruedas de reacción con sus respectivas características de performance.	108

Nomenclatura

μ	Constante gravitacional de la Tierra
ϕ	Roll
θ	Pitch
ψ	Yaw
$B(r, \theta, \varphi, t)$	Campo Geomagnético
ADCS	Subsistema de Determinación y Control de actitud
COTS	Commercial of the shelf
ECI	Earth Centered Inertial
EE	Espacio Estado
EKF	Filtro de Kalman Extendido
GPS	Global Position System
h	Momento Angular
I	Momentos principales de Inercia
IGRF	International Geomagnetic Reference Field
K	matriz de ganancia
LEO	Órbita Terrestre Baja
LQR	Linear Quadratic Regulator
LVLH	Local Vertical Local Horizontal
MBSE	Model Based System Engineering
MoP	Measures of Performance
NCEI	National Centers for Environmental Information
PD	Proporcional-Derivativo
PID	Proporcional-Integrativo-Derivativo
PSD	Densidad Espectro Potencia
q	Cuaternión
r	posición
SGP4	Simplified General Perturbations 4
STK	Systems Tool Kit
TLE	Two Line Elements
v	velocidad

1 Introducción

En las últimas décadas, la revolución tecnológica ha facilitado el desarrollo e implementación de satélites en órbitas bajas (Órbita Terrestre Baja (LEO)) a una escala sin precedentes [1]. Entre estos, los CubeSats han emergido como una solución eficiente y versátil para una amplia gama de aplicaciones espaciales. Los CubeSats son nanosatélites basados en un estándar que define sus dimensiones y arquitectura modular. Estos satélites se presentan en formatos como 1U, 3U, 6U y 12U, donde la unidad “U” equivale a un cubo de $10 \times 10 \times 11.35 \text{ cm}^3$ [2]. La Figura 1.1 evidencia el creciente interés por estos dispositivos, mostrando la cantidad de lanzamientos realizados y proyectados en el tiempo.

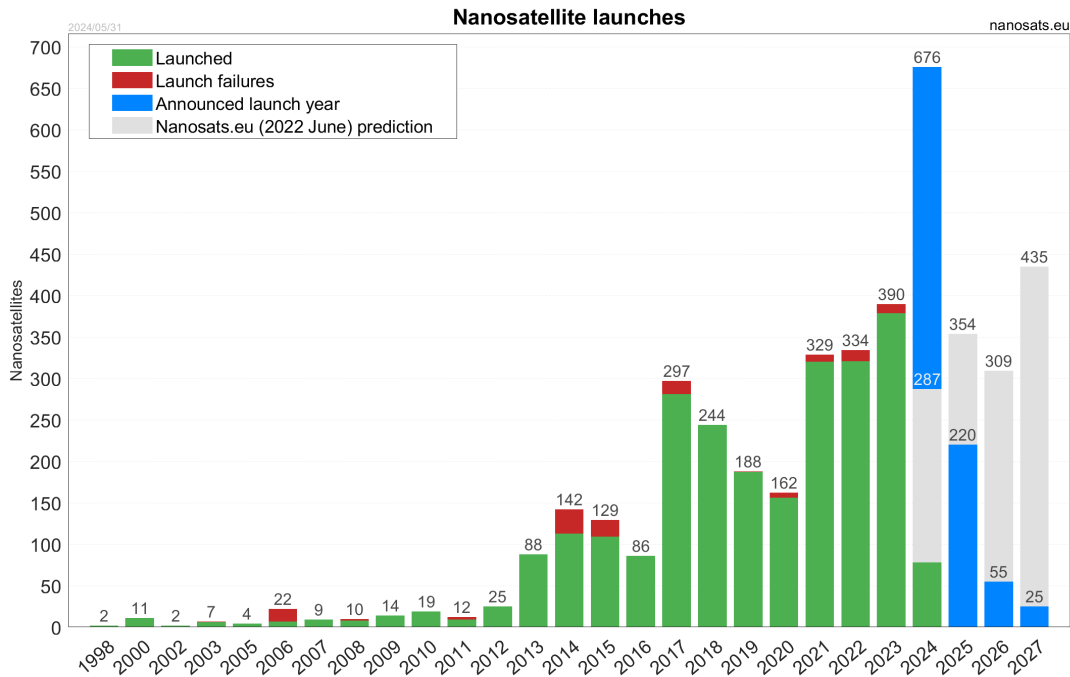


Figura 1.1 Cantidad de nanosatélites lanzados a través de los años [1].

Una de las aplicaciones comerciales más relevantes de los CubeSats es la observación terrestre mediante sensores ópticos [3]. Esta consiste en capturar imágenes de la superficie terrestre en diferentes bandas del espectro electromagnético, generalmente visible e infrarrojo cercano [4].

Para cumplir con los requisitos de calidad y resolución en este tipo de misiones, es fundamental que el satélite posea una adecuada capacidad de apuntamiento. El apuntamiento se define como la capacidad del satélite para ajustar su actitud, orientándose hacia un objetivo determinado [5]. Esta capacidad se caracteriza a través de los Measures of Performance (MoP) (Measures of Performance) relevantes para la misión, los cuales dependen tanto de las capacidades físicas del sistema como del desempeño de los algoritmos de estimación y control que conforman el Subsistema de Determinación y Control de actitud (ADCS).

Según la literatura [5, 6, 7, 8], los MoP de apuntamiento más relevantes para misiones de observación terrestre son:

- **Exactitud de apuntamiento** [5, 7]: Error absoluto entre la orientación requerida y la real.
- **Drift** [5]: Desviación acumulada del eje de apuntamiento en el tiempo, expresada en $^{\circ}/\text{hr}$.
- **Jitter** [9, 10]: Vibraciones de alta frecuencia que afectan la estabilidad visual de las imágenes, representadas mediante la Densidad Espectro Potencia (PSD) del filtro pasa alto aplicado a la señal de estabilización.
- **Agilidad** [11]: Tiempo de asentamiento para alcanzar una nueva orientación dentro de una banda de error establecida (generalmente 5%).

Estos parámetros de rendimiento deben ser analizados considerando las restricciones impuestas por los *System Engineering (SE) envelopes*, que agrupan limitaciones técnicas fundamentales como potencia eléctrica, masa, volumen y costo. Estas restricciones están directamente condicionadas por la carga útil del satélite, que suele ser el punto de partida para definir el diseño del ADCS [8].

Dado que los CubeSats poseen limitaciones estrictas de masa, volumen y consumo energético, resulta crucial establecer un balance adecuado entre los requerimientos de apuntamiento y las restricciones del sistema. Este balance debe lograrse en función del tipo de misión, buscando maximizar el rendimiento sin sobrepasar las capacidades físicas y presupuestarias.

En este contexto, es esencial definir claramente los parámetros orbitales, la dinámica del entorno, la geometría del satélite y las capacidades del ADCS. De esta manera, es posible estimar tanto el desempeño como el costo de la misión en términos de apuntamiento. La Figura 1.2 muestra una representación gráfica de las interacciones entre estos aspectos.

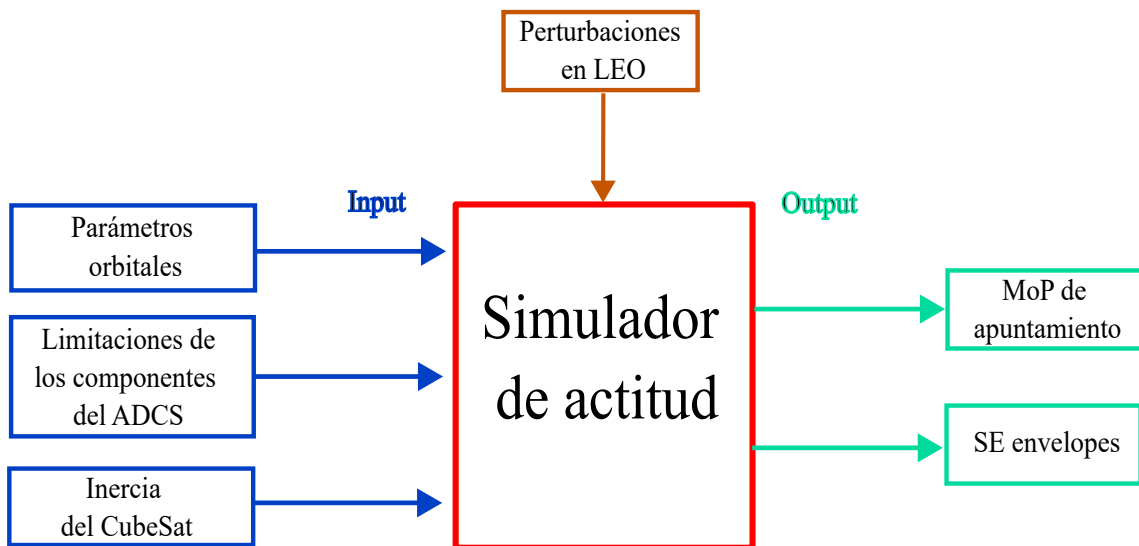


Figura 1.2 Representación gráfica de la relación entre aspectos de la misión respecto a costo y rendimiento (Elaboración propia).

Diversos estudios han abordado la simulación de componentes del ADCS [12], incluyendo sen-

sores, actuadores, algoritmos de estimación [13] y controladores [14]. Asimismo, existen trabajos que modelan y evalúan el ADCS completo de un CubeSat, desde su diseño hasta la obtención de métricas de rendimiento [15, 16, 17].

En cuanto a herramientas disponibles, destacan plataformas como el *Spacecraft Control Toolbox* [18] y el *Aerospace Blockset* de MATLAB [19], que permiten modelar la dinámica orbital y de actitud, visualizar resultados y cuantificar algunos MoP. También se encuentran entornos integrales como Valispace [20], orientados a la gestión de budgets de ingeniería. En el ámbito académico, el entorno de simulación *Basilisk*, desarrollado por la Universidad de Colorado, representa una alternativa robusta y gratuita para modelar dinámicas complejas y evaluar controladores espaciales [21].

A partir de esta revisión, se propone desarrollar una suite de simulación en Python, de acceso libre y basada en componentes comerciales actuales. Esta herramienta permitirá analizar el rendimiento del ADCS de un CubeSat y estimar el costo de distintas configuraciones en función de los requerimientos definidos por el usuario.

Una característica distintiva del simulador propuesto es la incorporación de un proceso de selección automatizada de componentes del ADCS, optimizando simultáneamente el rendimiento de apuntamiento y las restricciones impuestas por los SE envelopes. Para ello, se requiere un enfoque robusto de evaluación de alternativas tecnológicas.

La literatura ofrece diversos métodos para abordar este tipo de problema, que se pueden clasificar en dos grupos:

- **Métodos clásicos:** Incluyen técnicas como el análisis jerárquico de procesos (AHP), lógica difusa y análisis de Pareto, ampliamente utilizados en etapas tempranas del diseño [22, 23, 24].
- **Métodos modernos:** Se basan en simulaciones dinámicas y técnicas de aprendizaje automático, tales como árboles de decisión, análisis de sensibilidad y algoritmos de optimización no lineal [25, 26, 27].

Este trabajo se enmarca dentro de los métodos modernos, utilizando algoritmos de optimización acoplados a simulaciones dinámicas para identificar configuraciones óptimas de sensores y actuadores. La herramienta no solo permitirá encontrar soluciones específicas según restricciones de costo o desempeño, sino también entregar al usuario configuraciones recomendadas para cumplir objetivos definidos.

Un resumen visual del funcionamiento propuesto de esta suite se presenta en la Figura 1.3.

1.1 Hipótesis

Se postula que el diseño e implementación de una suite de simulación que modele el ambiente espacial y el ADCS, en conjunto con un algoritmo capaz de optimizar los parámetros de rendimiento y los SE envelopes, mejorará la eficacia operativa de las misiones CubeSat al seleccionar el subconjunto óptimo de componentes físicos necesarios.

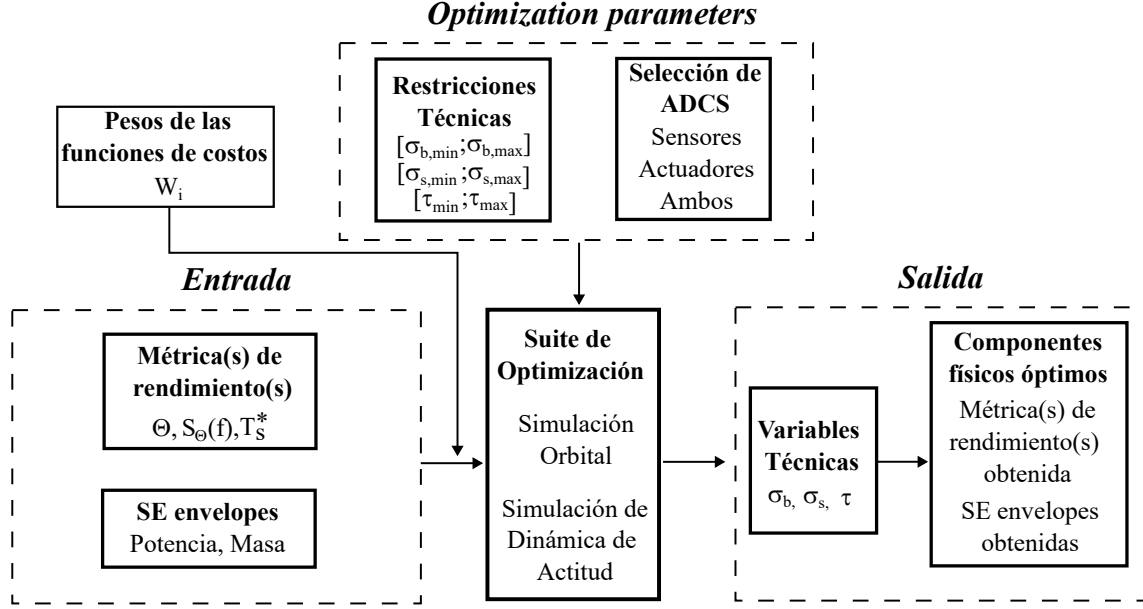


Figura 1.3 Solución propuesta para la suite de simulación optimizada (Elaboración propia).

1.2 Objetivos

Objetivo General: Diseñar e implementar una suite de simulación que modele el ambiente espacial y el ADCS de un CubeSat comercial en órbitas bajas de observación terrestre, capaz de optimizar el rendimiento (mediante los MoP de apuntamiento) y/o el costo (considerando los SE envelopes) definidos por el usuario, con el fin de seleccionar los componentes físicos del ADCS necesarios para la misión.

Objetivos Específicos:

- OE1: Desarrollar un marco teórico robusto sobre el modelamiento del ambiente espacial y del ADCS de un CubeSat, enfocado en aplicaciones de observación terrestre, basadas en referencias publicadas después del año 2010.
- OE2: Desarrollar un modelo de ambiente espacial que permita calcular los vectores de posición y velocidad del CubeSat, junto con un modelo de actitud que incluya al menos dos tipos de sensores y dos tipos de actuadores del ADCS.
- OE3: Implementar un algoritmo de estimación para el modelo dinámico del CubeSat con un margen de error inferior al 5%.
- OE4: Diseñar al menos un controlador que permita control de actitud en función de los actuadores implementados en la suite de simulación.
- OE5: Seleccionar e implementar un algoritmo de optimización no lineal capaz de optimizar los MoP de apuntamiento en función de los SE envelopes en la suite de simulación.
- OE6: Verificar cuantitativamente la suite de optimización mediante la comparación entre los resultados obtenidos en la simulación dinámica de actitud, configurada con componentes físicos en el CubeSat SUCHAI-3, y los resultados entregados por el algoritmo de

optimización, evaluando la consistencia en la selección de componentes y en el desempeño de los MoP de apuntamiento.

1.3 Limitaciones de diseño

Para el desarrollo del presente proyecto, se establecieron ciertas limitaciones de diseño que condicionan las decisiones metodológicas y técnicas tomadas a lo largo del trabajo. Una de las principales decisiones corresponde a la elección del lenguaje de programación. Esta elección debía considerar criterios de libre acceso, versatilidad, compatibilidad con futuras extensiones del simulador, y una amplia disponibilidad de bibliotecas científicas y herramientas de visualización.

Bajo estas consideraciones, se selecciona *Python* como el lenguaje base para la implementación de la suite de simulación y optimización. Python destaca por ser un lenguaje de alto nivel, de código abierto y ampliamente utilizado en la comunidad científica y espacial. Posee además una robusta colección de bibliotecas especializadas, tales como `NumPy`, `SciPy`, `Matplotlib`, `AstroPy`, `CVXPY`, y `Pyomo`, entre otras, que permiten abordar de manera eficiente problemas de modelación, control, visualización y optimización.

Otra limitación relevante corresponde al enfoque adoptado para la modelación física del sistema, el cual se basa en modelos simplificados pero representativos, con el objetivo de equilibrar precisión y tiempo de simulación. Asimismo, se considera como restricción el uso exclusivo de datos disponibles públicamente, sin acceso a bancos de datos operacionales de satélites reales. Esto motiva la validación cruzada entre componentes del simulador y su optimizador interno, en lugar de una validación empírica directa con misiones reales.

Finalmente, se establece como restricción que toda la suite sea autocontenida y modular, de modo que permita futuras integraciones con interfaces gráficas u otros entornos de simulación más complejos, como Basilisk o GMAT, en caso de ser necesario.

1.4 Metodología

OE1: Desarrollar el marco teórico relacionado con el ADCS y el ambiente espacial en el simulador mediante la revisión de trabajos previos. Se recopilará información de artículos y estudios sobre la determinación y control de actitud en CubeSats, junto con el estado del arte y las especificaciones técnicas de componentes disponibles en sitios web de fabricantes, para verificar la correcta implementación de los modelos y su similitud con estudios anteriores.

OE2: Implementar el propagador SGP4 para calcular los vectores de posición y velocidad del CubeSat, teniendo en cuenta las perturbaciones orbitales en órbitas bajas. En cuanto a los modelos dinámicos de actitud, se simularán las fuerzas magnéticas y el vector solar para representar el comportamiento de sensores como el magnetómetro y el sensor solar. Los actuadores considerados en los modelos serán el magnetorquer y la rueda de reacción en los ejes del CubeSat.

OE3: Estimar los cuaterniones y las velocidades angulares del CubeSat utilizando un estimador recursivo. Este estimador será aplicado al modelo dinámico lineal discreto, integrando el magnetorquer y las ruedas de reacción. Se emplearán las mediciones simuladas de los sensores para validar las estimaciones y se evaluará la precisión de las mismas mediante el cálculo del error cuadrático medio (MSE).

OE4: Diseñar un controlador PD o LQR dentro de la suite de simulación. Una vez implementados ambos controladores, se seleccionará el de mejor rendimiento en función de los MoP de apuntamiento bajo las mismas condiciones de simulación.

OE5: Realizar una revisión de optimizadores no lineales convexos y no convexos disponibles en Python, con el objetivo de encontrar una solución óptima que maximice el rendimiento y minimice el costo en función de las entradas y requisitos del usuario. Se evaluarán herramientas como `scipy.optimize` para determinar el optimizador más adecuado.

OE6: Para verificar la validez de la suite de optimización, se simulará el desempeño del sistema de actitud utilizando variados componentes físicos. Se obtendrán los MoP de apuntamiento a partir de esta simulación dinámica. Luego, se ejecutará el algoritmo de optimización bajo los mismos SE envelopes (restricciones de masa, potencia y costo) asociados a dichos componentes. Finalmente, se compararán los resultados entregados por la optimización con los obtenidos en la simulación dinámica, evaluando si el algoritmo logra recomendar un conjunto de componentes físico equivalente o similar al configurado inicialmente, y si el desempeño de los MoP es coherente entre ambos enfoques.

1.5 Carta Gantt

El proyecto sigue una estructura definida contenida en la carta Gantt mostrada en el Anexo A. El resumen en base a los entregables se muestra a continuación en la Tabla 1.1

Tabla 1.1 Cronograma de hitos y avances

Resultado	Objetivos es- pecíficos asociados	Hito de avance	Fecha de entrega
Finalización de la suite de simulación	OE1, OE2, OE3, OE4	Avance I: Pre-sentación	Semana 19 (29-02 de agosto de 2024)
Implementación de la optimización de parámetros en la suite de simulación	OE5	Avance II: Informe de avance	Semana 31 (21-27 de octubre de 2024)
Verificación de la suite de optimización y Redacción de artículo científico	OE6	Avance III: Entrega con manuscrito de artículo científico	Semana 22 (26 de mayo 2025-01 de junio de 2025)
Entrega final	-	Final: Presentación e informe final	Semana 31 (27 julio 2025-03 de agosto 2025)

2 Marco Teórico

2.1 Sistemas de referencia

Para describir la dinámica orbital, de actitud y el diseño del ADCS, es necesario la definición de los marcos de referencia a utilizar. Su selección obedece a criterios descritos a continuación.

- **Sistema de referencia body o del cuerpo [28]:** La dinámica relativa de actitud se describe con respecto al marco de referencia del cuerpo en el CubeSat, desde la cual se realizan las mediciones, debido a que los sensores de actitud están fijados a su cuerpo. Para los marcos de referencia del cuerpo, el eje z apunta en la dirección del momento de inercia más alto, y los ejes x e y son paralelos a los vectores de área de las caras de la nave espacial, apuntando todos en las direcciones principales del satélite, como se observa gráficamente en la Figura 2.1.

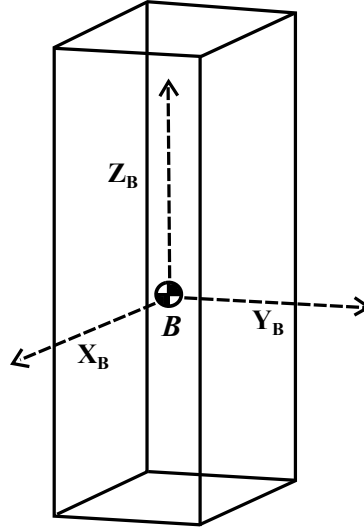


Figura 2.1 Marco de referencia del cuerpo (Elaboración propia).

- **Sistema de referencia inercial [28, 29]:** El Sistema de referencia inercial utilizado es el Earth Centered Inertial (ECI) debido a la necesidad de obtener vectores respecto a un marco de referencia no rotativo (asumiendo problema de dos cuerpos entre la Tierra y el satélite). El eje X apunta en la dirección del equinoccio de primavera. El plano XY es el plano ecuatorial de la Tierra, y el eje Z coincide con el eje de rotación de la Tierra y apunta hacia el norte. Este sistema de referencia se puede apreciar en la Figura 2.2.

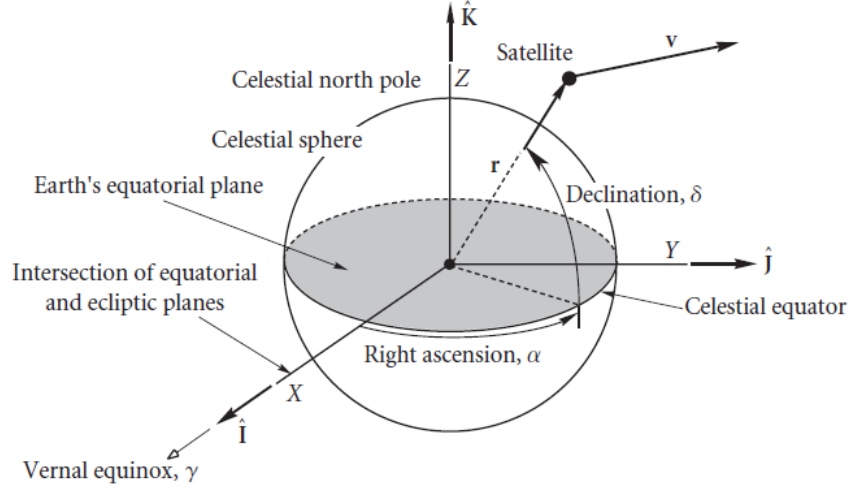


Figura 2.2 Marco de referencia inercial ECI [29].

- **Sistema de referencia LVLH o Roll-Pitch-Yaw [30]:** Se define un sistema coordenado que mantiene su orientación relativa a la Tierra a medida que la nave espacial se mueve en su órbita. Estas coordenadas son conocidas como roll, pitch y yaw (RPY), Local Vertical Local Horizontal (LVLH) u orbital como también será llamada en este trabajo y se ilustra en la Figura 2.3. En este sistema, el eje yaw se dirige hacia el nadir (es decir, hacia el centro de la Tierra), el eje pitch se dirige hacia la normal negativa de la órbita, y el eje roll es perpendicular a los otros dos, tal y como se muestra en (1). Se utilizará este Sistema de referencia para notar la posición ideal de la carga útil.

$$\hat{R} = \hat{P} \times \hat{Y} \quad (1)$$

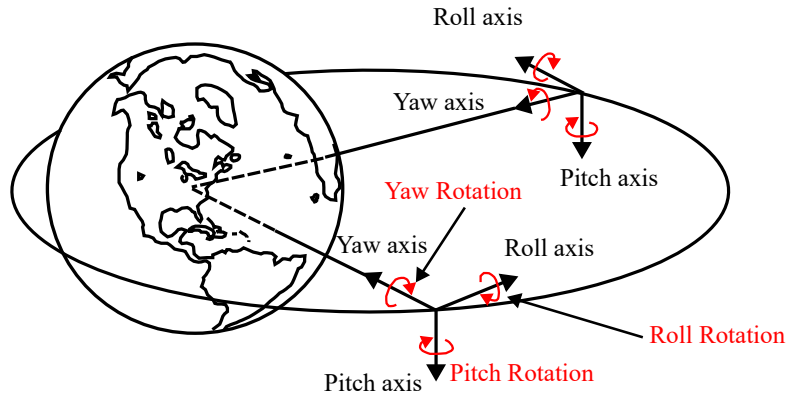


Figura 2.3 Marco de referencia inercial RPY [30].

2.2 Dinámica orbital

Para el modelamiento de la suite de simulación se debe conocer el significado de los parámetros orbitales que se entregara como entrada, así como las ecuaciones que gobiernan el movimiento del satélite alrededor de la Tierra y las perturbaciones presentes a baja altura.

2.2.1 Parámetros orbitales

Si la masa de un satélite se considera insignificante en comparación con la masa de la Tierra, y bajo el supuesto de que la Tierra es esféricamente simétrica, la aceleración $\ddot{\mathbf{r}}$ de un satélite está dado por la ley de gravedad de Newton descrita a continuación:

$$\ddot{\mathbf{r}} = -\frac{GM_E}{r^3}\mathbf{r} \quad (2)$$

Donde $\mathbf{r}$ es el vector posición entre el satélite y la Tierra y GM_E es conocida como la Constante gravitacional de la Tierra (μ) y está dada por:

$$\mu = GM_E = 398600,4418 \left[\frac{km^3}{s^2} \right] \quad (3)$$

Al resolver (2), se obtiene la posición y la velocidad del del satélite respecto a la Tierra en cualquier instante de tiempo dependiendo del sistema de referencia a utilizar. Si bien se tiene una cuantificación del posicionamiento y el movimiento del satélite, generalmente se utiliza otra caracterización para definir la órbita, utilizando los elementos keplerianos que se presentan en la Figura 2.4.

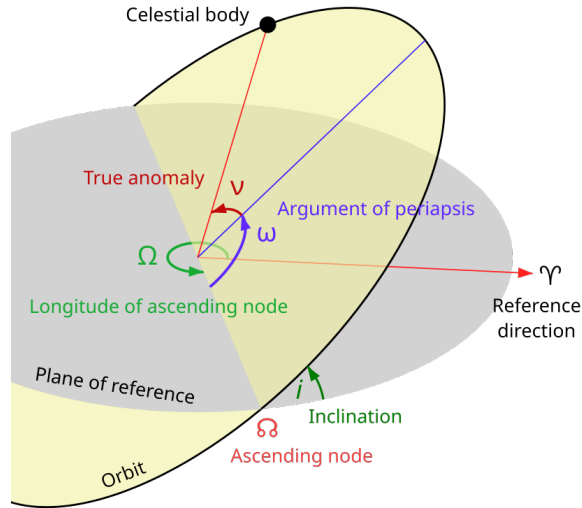


Figura 2.4 Elementos keplerianos [31].

Dichos elementos se definen brevemente a continuación, los que se pueden determinar desde la posición y la velocidad del satélite respecto a la Tierra mediante relaciones matemáticas obtenidas en [29].

- **Excentricidad (e):** Describe el alargamiento de la órbita. Si presenta valores entre 0 y 1 tendrá forma de elipse. Si es igual a 0 representa una órbita circular, mientras que si es igual a 1 tiene forma de parábola. Para casos mayores a 1 se presentan orbitas de trayectoria hipérbolica.

- **Semieje mayor (a):** Es la distancia entre el periapsis (distancia más cercana entre el satélite y la Tierra) y el apoapsis (distancia más lejana entre el satélite y la Tierra) dividido por 2. Representa el radio para orbitas circulares.
- **Inclinación (i):** Inclinación vertical de la elipse con respecto al plano de referencia (plano ecuatorial).
- **Right Ascension of the Ascending Node (RAAN Ω):** Inclinación vertical de la elipse con respecto al plano de referencia (plano ecuatorial).
- **Argumento del periapsis (ω):** Se define la orientación de la elipse en el plano orbital, como un ángulo medido desde el nodo ascendente a la periapsis.
- **Anomalía verdadera (ν):** Define la posición del cuerpo orbitante a lo largo de la elipse en un tiempo específico.

2.2.2 Perturbaciones presentes en LEO

Existen perturbaciones en el espacio que afectan a los satélites en órbita, de las cuales algunas tienen más relevancia a bajas altura respecto de la Tierra. Estas se muestran a continuación:

Gravedad no esférica [32]: La Tierra no es una esfera perfecta y la masa se distribuye de manera no uniforme. A diferencia de las simplificaciones que se aplican a menudo en órbitas altas, donde la influencia de la Tierra se aproxima a una esfera, en LEO la distribución irregular de la masa terrestre y las variaciones en la altitud pueden generar perturbaciones significativas en las trayectorias de los satélites. Por lo tanto, como la fuerza de gravedad depende directamente de la masa, el campo gravitatorio reflejará esta falta de uniformidad.

Para lograr modelar la gravedad no esférica se utiliza una expansión armónica esférica, con modelos como el geopotencial que descompone el campo gravitatorio terrestre en una serie de términos, cada uno correspondiente a una armonía esférica y su respectiva magnitud. Dentro de los coeficientes utilizados dentro del modelo recién mencionado están los “J”, siendo el J2 el principal para modelar el achatamiento de la Tierra. Otros J como el J3, J4, etc., modelan a mayor detalle la distribución másica de la Tierra.

Efectos atmosféricos [33]: Los efectos del arrastre y el oxígeno atómico (O) tienen implicancias para los satélites de baja altura (menor a 600 km). El arrastre se define como una fuerza resistiva que actúa sobre un objeto en movimiento a través de un fluido y tiende a disminuir su velocidad, cuyas implicancias son que acorta la vida útil del satélite. El arrastre depende de la densidad, la velocidad y también variará según cómo cambie la atmósfera (se expanda o se contraiga) debido a la variación en la actividad solar.

Por otro lado, debido a que en la atmosfera superior existe una mayor radiación, esto hace que se disocien los átomos de O_2 a O , los cuales son muy reactivos y potencialmente dañinos, degradando las superficies del CubeSat e interfiriendo con los sensores para la determinación de actitud.

2.3 Cinemática y dinámica de actitud

Para la cinemática y dinámica de actitud, el satélite ya no se asume como una partícula perturbada (como en el caso de la dinámica orbital), sino como un cuerpo rígido con masa. Con esto aparecen conceptos que serán definidos en esta sección.

2.3.1 Actitud de un satélite y sus representaciones

La actitud de un satélite se refiere a la orientación o posición que mantiene en el espacio mientras órbita alrededor de la Tierra u otro cuerpo celeste. Para describir esta actitud, se utilizan diversas representaciones matemáticas que permiten definir de manera precisa su orientación en el marco de referencia del cuerpo respecto al inercial, las cuales se presentan a continuación [30, 32]:

- **Direction Cosine Matrix (DCM):** Esta parametrización utiliza una matriz 3x3 para representar la orientación del satélite en relación con un sistema de referencia fijo. La matriz contiene nueve elementos que son los cosenos directores de los ejes del satélite en relación con los ejes de referencia. Es una representación matemáticamente precisa pero no es tan intuitiva como otras. Dicha representación se visualiza en la Figura 2.5 [34].

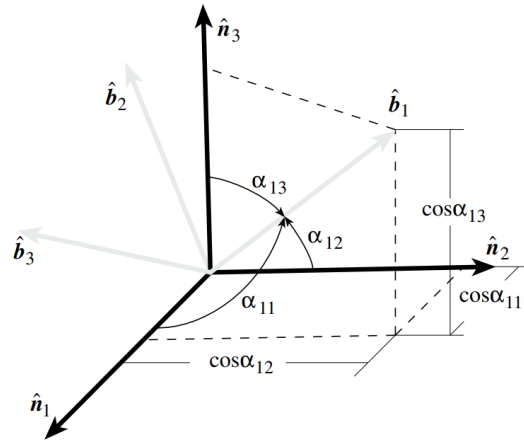


Figura 2.5 Representación gráfica de la primera fila de la matriz de cosenos directores [34].

- **Euler axis/angle:** En esta parametrización, se utiliza un vector tridimensional (el eje de Euler) junto con un ángulo para describir la orientación. El vector de Euler define el eje de rotación, mientras que el ángulo especifica la magnitud de la rotación alrededor de ese eje, como se muestra en la Figura 2.6. Es útil para representar giros simples y es intuitiva.

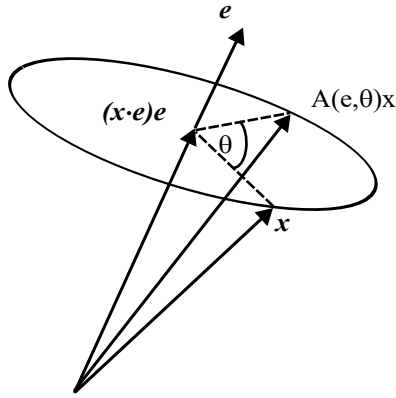


Figura 2.6 Representación de un cambio de orientación en Euler axis/angle de x [32].

- **Euler angles:** Esta parametrización describe la orientación mediante tres ángulos, generalmente llamados Roll (ϕ), Pitch (θ) y Yaw (ψ), que representan las rotaciones en torno a los ejes específicos (por ejemplo, X, Y y Z). Se presenta en la Figura 2.7 las rotaciones realizadas por esta parametrización.

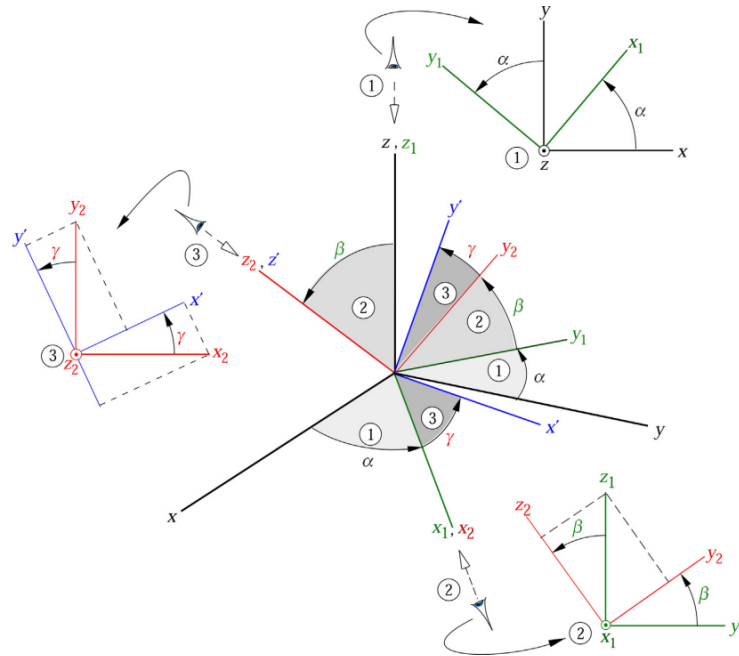


Figura 2.7 Secuencia clásica de Euler de tres rotaciones que transforman xyz en $x'y'z'$ [29].

- **Cuaternión (q):** Esta parametrización utiliza cuatro parámetros para representar la orientación, siendo tres de estas vectoriales y una escalar. Se discutirá más a fondo en la siguiente sección [32].

2.3.2 Cuaterniones y cinemática de cuaterniones

En este trabajo, la parametrización seleccionada para la descripción de la actitud es el cuaternión. Los cuaterniones tienen múltiples ventajas en comparación con otras parametrizaciones de actitud. Por ejemplo, en la parametrización de ángulos de Euler, la propagación de la actitud no es suave. Sin embargo, esta suavidad es fundamental para el correcto funcionamiento de métodos de estimación como el Filtro de Kalman [30].

Por otro lado, la desventaja de la parametrización de la matriz de cosenos directores es que conduce a una descripción de la actitud utilizando nueve elementos no independientes, y cumple con seis restricciones impuestas por la ortogonalidad de la matriz de actitud que son redundantes. La cantidad mínima de elementos que se pueden utilizar para describir la actitud sin singularidades son cuatro. A partir del hecho de que cualquier rotación puede describirse utilizando un solo eje de rotación y un ángulo que describe la rotación alrededor de este eje, el cuaternión se define mediante 4 elementos, con una parte vectorial y una parte escalar como se muestra a continuación [29]

$$q = \begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \end{bmatrix} = \begin{bmatrix} \mathbf{q} \\ q_3 \end{bmatrix} = \begin{bmatrix} \sin\left(\frac{\theta}{2}\right) \hat{u} \\ \cos\left(\frac{\theta}{2}\right) \end{bmatrix}. \quad (4)$$

La expresión $\mathbf{q}$ es la parte vectorial $\mathbf{q} = q_0\hat{i} + q_1\hat{j} + q_2\hat{k}$, y q_3 es la parte escalar. La representación mostrada corresponde a un cuaternión pasivo, el cual se utiliza para rotar el sistema de coordenadas en sí (sin rotar el vector). Si se desea rotar el vector sin rotar el sistema de coordenadas, se utiliza un cuaternión activo, y se obtiene cambiando la componente escalar al inicio, tal y como se muestra en la siguiente expresión:

$$q = \begin{bmatrix} q_3 \\ q_0 \\ q_1 \\ q_2 \end{bmatrix} = \begin{bmatrix} q_3 \\ \mathbf{q} \end{bmatrix}$$

Sabiendo esto, se puede definir la cinemática de la actitud del satélite utilizando cuaterniones mediante (5), sabiendo que ω_0 , ω_1 y ω_2 son las velocidades angulares en el marco de referencia cuerpo del satélite:

$$\frac{dq}{dt} = \frac{1}{2} \begin{bmatrix} 0 & -\omega_2 & \omega_1 & -\omega_0 \\ \omega_2 & 0 & -\omega_0 & -\omega_1 \\ -\omega_1 & \omega_0 & 0 & -\omega_2 \\ \omega_0 & \omega_1 & \omega_2 & 0 \end{bmatrix} \hat{q} \quad (5)$$

Además, se tiene la multiplicación de cuaterniones, la cual será útil para representar las rotaciones entre los sistemas de referencia y aproximaciones discretas que utilizan esta operación, la cual se representa en (6):

$$q \cdot r = \begin{bmatrix} q_3 r_0 + q_0 r_3 + q_1 r_2 - q_2 r_1 \\ q_3 r_1 + q_1 r_3 + q_2 r_0 - q_0 r_2 \\ q_3 r_2 + q_2 r_3 + q_0 r_1 - q_1 r_0 \\ q_3 r_3 - q_0 r_0 - q_1 r_1 - q_2 r_2 \end{bmatrix} \quad (6)$$

Si bien esta parametrización no tiene una representación física obvia, se puede describir la rotación del satélite a través del tiempo mediante un integrador numérico, sabiendo la condición inicial tanto del cuaternión como de la velocidad angular.

2.3.3 Dinámica de actitud

La ecuación que describe la variación del vector Momento Angular (h) a través del tiempo para un torque aplicado en un marco de referencia del cuerpo representa la dinámica de actitud y se muestra en (7) [29]

$$I \frac{d\omega}{dt} = -\omega \times I\omega + \tau, \quad (7)$$

sabiendo que ω es el vector de velocidad angular instantánea en el cuerpo, I son los momentos principales de inercia y τ son los torques aplicados, esta ecuación se puede representar también según sus componentes en i, j y k

$$\begin{aligned} \dot{\omega}_0 &= \frac{\omega_1 \omega_2 (I_y - I_z)}{I_x} + \frac{\tau_x}{I_x}, \\ \dot{\omega}_1 &= \frac{\omega_0 \omega_2 (I_x - I_z)}{I_y} + \frac{\tau_y}{I_y}, \\ \dot{\omega}_2 &= \frac{\omega_0 \omega_1 (I_x - I_y)}{I_z} + \frac{\tau_z}{I_z}. \end{aligned}$$

Los torques externos son los provocados por los actuadores y por perturbaciones externas en orbitas de baja altura. Las ultimas mencionadas generalmente se simplifican a relaciones para el peor de los casos analizados. Estas perturbaciones que afectan a la dinámica rotacional del satélite son cuatro y se discuten a fondo en el Anexo B.

2.4 Subsistema de determinación y control de actitud

El subsistema de determinación y control de actitud de un CubeSat es el responsable de determinar su orientación en el espacio y controlarla respecto a un objetivo durante un periodo específico. También se requiere para sobrevivir en el entorno espacial, controlando el satélite en orientaciones tales que se genere energía apuntando las celdas fotovoltaicas hacia el sol.

El proceso para poder describir el movimiento del satélite se describe en tres pasos o procesos los cuales son llamados Guidance, Navigation and Control (GNC). Cada uno se describe según los componentes del ADCS, como se muestra a continuación [35]:

- **Navigation:** Es el primer paso por seguir, y determina la actitud del satélite y la tasa de rotación mediante los sensores tanto inerciales como de medición externa. Responde a la pregunta: ¿Dónde está el satélite?
- **Guidance:** Al determinar la actitud con los sensores, se utilizan algoritmos de determinación de actitud para estimar la orientación tanto inicial como a través del tiempo, para posteriormente utilizar algoritmos de determinación de actitud para estimar la orientación del satélite. Responde a la pregunta: ¿Hacia dónde quiere ir el satélite?
- **Control:** Ya conocido hacia donde quiere ir el satélite, se genera el cambio de actitud mediante la implementación de torques utilizando controladores en conjunto con actuadores. Responde a la pregunta: ¿Cómo dirijo el satélite hacia allá?

2.4.1 Navigation (Sensores)

Para la determinación de actitud se utilizan componentes físicos llamados sensores, los cuales miden su orientación en base a la inercia del satélite como también observando las estrellas circundantes/cuerpos celestes o midiendo fuerzas representativas de una posición en particular [6]. Los sensores comúnmente utilizados en CubeSat se presentan en la Tabla 2.1, en conjunto con su descripción.

Tabla 2.1 Sensores utilizados en CubeSat [6, 36].

Sensores	Descripción
Giroscopios	Los giroscopios miden la tasa de cambio de la orientación angular respecto al marco inercial del satélite [°/s]. Los giroscopios proporcionan información sobre los movimientos de rotación en los tres ejes (roll, pitch y yaw).
Sensores de sol	Los sensores solares se utilizan para estimar la dirección del Sol en el marco de referencia del cuerpo del satélite.
Magnetómetros	Los magnetómetros determinan el campo magnético de la Tierra, midiendo su dirección y su fuerza en [nT].
Global Position System (GPS)	Mediante el receptor GPS se obtiene la posición tridimensional (latitud, longitud y altitud) con alta precisión. Esto permite al CubeSat conocer su ubicación en la órbita terrestre.
Star Tracker	Un Star Tracker o contador de estrellas es un sistema de sensores cuya función principal es determinar con exactitud la actitud del CubeSat utilizando las estrellas circundantes como referencia.

2.4.2 Guidance (Algoritmos)

En esta sección se proporciona una descripción de los diferentes métodos de determinación de actitud que se consideran para el simulador. En primera instancia, existen dos tipos principales de métodos de determinación de actitud, siendo el primero de ellos el método determinístico que

utiliza la información de las lecturas de los sensores a lo largo de la misión y las comparan con modelos informáticos para calcular la actitud actual [15]. Por otro lado, existen los estimadores recursivos que procesan las lecturas de sensores actuales y las compara con la última estimación de actitud para crear una nueva estimación.

Enfoque determinista de la determinación de actitud

Los algoritmos deterministas resuelven el problema de determinación de actitud a partir de observaciones directas, sin estimación probabilística. A continuación, se presentan tres de los métodos más utilizados.

Método TRIAD

La solución Tri-axial Attitude Determination (TRIAD) requiere dos conjuntos de vectores: un vector de observación de cada uno de los dos sensores ubicados en el satélite ($\mathbf{V}_1$ y $\mathbf{V}_2$), y un vector de referencia para cada observación en términos de su dirección de referencia inercial ($\mathbf{W}_1$ y $\mathbf{W}_2$). Con estos vectores se crean triadas de referencia ($\mathbf{M}_{\text{ref}}$) y de observación ($\mathbf{M}_{\text{obs}}$) descritas en la Tabla 2.2 [15].

Tabla 2.2 Descripción de las triadas de referencia y de observación.

Triada	Componentes de la triada
$\mathbf{M}_{\text{obs}} = [\hat{r}_1 \quad \hat{r}_2 \quad \hat{r}_3]$	$\hat{r}_1 = \hat{V}_1; \hat{r}_2 = \frac{\hat{V}_1 \times \hat{V}_2}{ \hat{V}_1 \times \hat{V}_2 }; \hat{r}_3 = \hat{r}_1 \times \hat{r}_2$
$\mathbf{M}_{\text{ref}} = [\hat{s}_1 \quad \hat{s}_2 \quad \hat{s}_3]$	$\hat{s}_1 = \hat{W}_1; \hat{s}_2 = \frac{\hat{W}_1 \times \hat{W}_2}{ \hat{W}_1 \times \hat{W}_2 }; \hat{s}_3 = \hat{s}_1 \times \hat{s}_2$

Una vez que se calculan las triadas de observación y referencia, se puede encontrar la solución TRIAD. Esta solución es la matriz de cosenos directores A que se define de la siguiente manera:

$$A = M_{\text{obs}}(M_{\text{ref}})^T$$

Esta matriz representa la rotación desde el marco de referencia del cuerpo del satélite al marco de referencia inercial fijo a la Tierra. Una vez que se conoce esta matriz, la actitud del satélite se puede expresar en términos del marco de referencia inercial fijo a la Tierra. Cuando se utiliza el método TRIAD, el sensor más exacto debe elegirse siempre como V_1 con su correspondiente vector de referencia W_1 .

q-method

El q-method resuelve el problema de Wahba mediante la minimización de una función de pérdida cuadrática, construyendo una matriz de actitud a partir del eigenvector asociado al mayor autovalor de una matriz K . Este método es más robusto que TRIAD, ya que considera múltiples pares de vectores ponderados por su confiabilidad [15].

Método QUEST

QUEST (QUaternion ESTimator) es una solución eficiente al problema de Wahba mediante la parametrización en cuaterniones. Es computacionalmente más liviano que el q-method completo y es ampliamente usado en sistemas embebidos por su eficiencia [15].

Enfoque recursivo de la estimación de actitud

Aunque se requiere un método determinista para la adquisición inicial de la actitud, los métodos recursivos resultan más eficientes para su mantenimiento a lo largo del tiempo. A diferencia de los enfoques deterministas, los métodos recursivos actualizan la actitud estimada en cada instante utilizando únicamente la medición actual del sensor y la estimación previa del estado. El más común de estos métodos es el Filtro de Kalman [32].

Este filtro constituye un estimador recursivo óptimo para sistemas dinámicos lineales sujetos a ruido, y opera en dos fases: predicción y actualización. En la fase de predicción, se estima el estado del sistema a partir de un modelo dinámico conocido. Luego, en la fase de actualización, esta predicción se corrige utilizando las mediciones disponibles, ajustando así la estimación del estado actual del sistema [37].

El filtro trabaja en tiempo discreto, utilizando un índice de tiempo k , lo cual es adecuado para CubeSats, dado que el intervalo de muestreo coincide con la frecuencia de lectura de los sensores. No obstante, debido a que el modelo dinámico del satélite es inherentemente no lineal, se emplea una extensión del filtro conocida como *Filtro de Kalman Extendido* (*Extended Kalman Filter*, EKF), que adapta el mismo esquema recursivo mediante una linealización local del sistema.

Este proceso de linealización se realiza a través de matrices Jacobianas, obtenidas mediante derivadas parciales de primer orden de las funciones que describen el sistema dinámico y el modelo de medición. Estas matrices permiten aproximar el comportamiento no lineal alrededor del estado estimado actual, y deben ser recalculadas en cada paso de tiempo [38].

El modelo dinámico del sistema y la ecuación de medición se expresan como

$$x_{k+1} = f_k(x_k, u_k) + w_k, \quad z_k = h_k(x_k) + v_k,$$

donde x_k representa el estado del sistema en el instante k , u_k es la entrada de control, z_k es la medición del sistema, f_k describe la dinámica no lineal, y h_k el modelo de medición. Los términos w_k y v_k corresponden al ruido del proceso y al ruido de medición, respectivamente.

Las matrices Jacobianas se obtienen como

$$A(\hat{x}, t) = \left. \frac{\partial f}{\partial x} \right|_{\hat{x}}, \quad H(\hat{x}, t) = \left. \frac{\partial h}{\partial x} \right|_{\hat{x}},$$

siendo A la derivada parcial de la dinámica con respecto al estado, y H la derivada parcial del modelo de medición, evaluadas en el estado estimado actual $\hat{x}$.

La estimación a priori del estado se calcula mediante la función de predicción del modelo no lineal,

$$\hat{x}_k^- = f_k(\hat{x}_{k-1}, u_{k-1}),$$

y la matriz de covarianza del error de predicción se obtiene como

$$P_k^- = A_k P_{k-1} A_k^T + Q_k,$$

donde Q_k representa la matriz de covarianza del ruido del proceso.

La ganancia de Kalman se determina aplicando la siguiente expresión:

$$K_k = P_k^- H_k^T (H_k P_k^- H_k^T + R_k)^{-1},$$

en la que R_k es la matriz de covarianza del ruido de medición.

Finalmente, se realiza la corrección del estado y de la covarianza del error a través de las siguientes ecuaciones,

$$\hat{x}_k = \hat{x}_k^- + K_k (z_k - h_k(\hat{x}_k^-)), \quad P_k = (I - K_k H_k) P_k^-. \quad (8)$$

2.4.3 Control (Controladores y actuadores)

Para el control del satélite, se requiere la orden del controlador para generar un torque mediante los actuadores.

Controladores

Controlador Proporcional-Derivativo (PD) y Proporcional-Integrativo-Derivativo (PID) [30]: Es un tipo de controlador ampliamente utilizado en automatización y control de procesos para regular sistemas dinámicos y mantener una variable de proceso en un valor deseado o setpoint. A continuación, se explicará cada uno de los componentes y cómo funcionan juntos para controlar el sistema:

- **Componente Proporcional (P):** El término proporcional es la parte principal del controlador PID. Su función es proporcionar una respuesta inmediata a las desviaciones actuales entre la variable controlada y el valor deseado (error). La salida del término proporcional (P) es directamente proporcional al error actual, por lo que cuanto mayor sea el error, mayor será la corrección aplicada.
- **Componente Integrativo (I):** El término integral es responsable de acumular el error a lo largo del tiempo y compensar errores persistentes o a largo plazo. El término integral responde a la acumulación de errores pasados, por lo que tiende a eliminar errores persistentes o sistemáticos. Esta componente ayuda a reducir el error constante (offset) y garantiza que el sistema alcance el setpoint. Sin el componente integral, el controlador podría quedarse con un error constante incluso si el controlador proporcional es capaz de mantenerlo bajo control.
- **Componente Derivativo (D):** El término derivativo es sensible a la tasa de cambio del error. Se encarga de prevenir oscilaciones y estabilizar el sistema. La acción derivativa es capaz de prever la dirección en la que el error se está moviendo y disminuir la velocidad a la que se acerca al setpoint. Ayuda a suavizar las respuestas del sistema y evita que el controlador reaccione de manera brusca ante cambios repentinos en el error.

La salida del controlador se representa mediante la siguiente ecuación:

$$U = k_p P + k_i I + k_d D$$

Siendo k_p , k_i y k_d constantes de ajuste de ganancias proporcional, integral y derivativo, respectivamente, determinando la magnitud de la contribución de cada término de control en general.

Controlador Linear Quadratic Regulator (LQR) [39]: Este es un método de control óptimo utilizado en sistemas dinámicos lineales y de tiempo continuo. Su objetivo es encontrar la ley de control lineal que minimiza una función de costo cuadrática, teniendo en cuenta tanto

el estado del sistema como la entrada de control. Para su uso, se debe modelar el sistema según (9), donde $\mathbf{x}$ es el vector de estado, $\mathbf{u}$ es el vector de entrada de control, $\mathbf{A}$ es la matriz de estado y $\mathbf{B}$ es la matriz de entrada

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u}. \quad (9)$$

Posteriormente, se requiere definir una función de costo cuadrática que será minimizada para optimizar el desempeño del sistema. Esta función típicamente incluye dos términos principales: uno que penaliza la desviación o error en el estado del sistema y otro que penaliza el esfuerzo o consumo asociado a la acción de control. Las matrices $\mathbf{Q}$ y $\mathbf{R}$ corresponden a matrices de ponderación que asignan la importancia relativa a cada uno de estos términos.

En detalle, la matriz $\mathbf{Q}$ pondera el costo asociado a los errores en las variables de estado, reflejando la prioridad de mantener la actitud cercana al objetivo deseado. Por otro lado, la matriz $\mathbf{R}$ penaliza el esfuerzo del control, es decir, la magnitud o la energía utilizada por los actuadores para corregir la actitud. Ajustar estas matrices permite balancear entre la precisión del apuntamiento y el consumo energético o desgaste de los actuadores, influyendo directamente en el desempeño y la eficiencia del controlador. De esta forma, la función de costo se expresa como

$$J = \int_0^\infty (\mathbf{x}^T \mathbf{Q} \mathbf{x} + \mathbf{u}^T \mathbf{R} \mathbf{u}) dt = \sum_{k=0}^\infty (\mathbf{x}_k^T \mathbf{Q} \mathbf{x}_k + \mathbf{u}_k^T \mathbf{R} \mathbf{u}_k).$$

El objetivo es encontrar $\mathbf{u} = -\mathbf{K}\mathbf{x}$ que minimiza la función de costo, siendo $\mathbf{K}$ la ganancia del controlador. Esta matriz $\mathbf{K}$ se obtiene al resolver la ecuación de Riccati, donde $\mathbf{P}$ es la matriz simétrica definida positiva asociada con la solución continua en (10) o discreta en (11)

$$\mathbf{A}^T \mathbf{P} + \mathbf{P} \mathbf{A} - \mathbf{B} \mathbf{R}^{-1} \mathbf{B}^T \mathbf{P} + \mathbf{Q} = 0 \quad (10)$$

$$\mathbf{A}^T \mathbf{P} \mathbf{A} - (\mathbf{A}^T \mathbf{P} \mathbf{B})(\mathbf{R} + \mathbf{B}^T \mathbf{P} \mathbf{B})^{-1}(\mathbf{B}^T \mathbf{P} \mathbf{A}) + \mathbf{Q} - \mathbf{P} = 0 \quad (11)$$

Ya encontrada la matriz $\mathbf{P}$, la ley de control óptimo se obtiene como $\mathbf{u} = -\mathbf{K}\mathbf{x}$ con $\mathbf{K} = \mathbf{R}^{-1} \mathbf{B}^T \mathbf{P}$ en el caso continuo y $\mathbf{K} = (\mathbf{R} + \mathbf{B}^T \mathbf{P} \mathbf{B})^{-1}(\mathbf{B}^T \mathbf{P} \mathbf{A})$ en el caso discreto. Esta ley se implementa en el sistema dinámico para estabilizarlo y minimizar la función de costo a lo largo del tiempo. El controlador LQR es particularmente eficaz para sistemas lineales y proporciona un enfoque sistemático para el diseño de controladores óptimos.

Actuadores

Los actuadores son los que generan el torque necesario para el control del satélite. Los actuadores comúnmente utilizados en CubeSat se muestra en la Tabla 2.3, en conjunto con una breve descripción y un ejemplo utilizado:

2.5 System Engineering Envelopes

Los *Systems Engineering Envelopes* (SE Envelopes) definen las restricciones técnicas, físicas y operativas que orientan el diseño, desarrollo y operación de un CubeSat a lo largo de toda su vida útil [5]. Estas restricciones permiten asegurar que todos los subsistemas se mantengan

Tabla 2.3 Actuadores utilizados en CubeSat [6].

Actuadores	Descripción
Magnetorquer	Los magnetorquers son dispositivos de control de actitud construidos utilizando bobinas electromagnéticas, que generan un torque a través de interacciones entre el campo magnético ambiental y dipolos magnéticos generados por este actuador.
Rueda de reacción	Una rueda de reacción es un motor acoplado a un disco de alta inercia que gira a gran velocidad a lo largo de un eje fijo del satélite. Este funciona aplicando un torque T en el disco, provocando un aumento en su Momento Angular (h). Por conservación de h y al haber ausencia de fuerzas externas se genera un torque de igual magnitud, pero en sentido contrario, que es aplicado en el CubeSat.

dentro de márgenes factibles, maximizando la eficiencia en el cumplimiento de los objetivos de misión.

En este estudio, se consideran exclusivamente dos parámetros críticos dentro de los SE Envelopes: **masa** y **potencia**. Ambos representan restricciones fundamentales en el diseño de CubeSats, debido a sus limitaciones inherentes como plataformas de tamaño y capacidad reducida [40].

La **masa** está estrictamente limitada por los factores de forma estandarizados del formato CubeSat, donde incluso pequeñas variaciones pueden afectar tanto el cumplimiento de los requisitos de lanzamiento como la dinámica orbital del satélite [41]. Por su parte, la **potencia** disponible se encuentra restringida por la limitada superficie para instalar paneles solares y la reducida capacidad de almacenamiento energético en órbita [42]. Estas limitaciones energéticas condicionan directamente la operación simultánea de múltiples subsistemas, especialmente en satélites pequeños.

La elección de masa y potencia como parámetros principales no solo responde a su impacto en la factibilidad física del diseño, sino también a su influencia directa en la selección de componentes del sistema de determinación y control de actitud (ADCS). La mayoría de los sensores y actuadores disponibles comercialmente (*Commercial Off-The-Shelf*, COTS) presentan una amplia variabilidad en estos dos aspectos, por lo que cualquier decisión de diseño implica un compromiso entre rendimiento y consumo de recursos [43].

En consecuencia, este trabajo se centra en analizar las configuraciones del ADCS desde una perspectiva de *trade-off* entre masa y potencia, con el objetivo de determinar cuál combinación de sensores y actuadores permite lograr un apuntamiento preciso de la carga útil hacia la Tierra, utilizando la menor cantidad posible de recursos disponibles. Esta estrategia permite reducir la complejidad del problema de optimización sin perder relevancia práctica, asegurando un enfoque técnicamente viable para futuras misiones de observación terrestre con CubeSats.

2.6 Controlabilidad y observabilidad de un sistema de control

Un sistema dinámico se considera controlable si se pueden aplicar señales de control que accionen cualquier estado del sistema dentro de una cantidad de tiempo finita. Esta característica también se denomina accesibilidad. Por otro lado, se considera observable si todos sus estados pueden conocerse a partir de la salida del sistema.

Si se tiene un modelo de espacio de estados de tiempo continuo con N_x estados, N_y salidas y N_u entradas como se muestra a continuación:

$$\dot{x} = Ax + Bu$$

$$y = Cx + Du$$

Donde:

- x : Estados del sistema
- u : Entradas del sistema
- y : Salidas del sistema
- A, B, C y D : Las matrices de espacio de estados con tamaños $N_x \times N_x$, $N_x \times N_u$, $N_y \times N_x$ y $N_y \times N_u$ respectivamente de valores reales o complejos.

El sistema es controlable y observable si la matriz de controlabilidad $Co = [B, AB, A^2B, \dots, A^{n-1}B]$ y la matriz de observabilidad $Ob = [C, CA, CA^2, \dots, CA^{n-1}]$ tienen un rango total, es decir, el rango es igual al número de estados del modelo de espacio de estados.

Es relevante tener en cuenta estos conceptos, ya que, para utilizar algunos algoritmos de estimación de actitud, es necesario que el sistema sea observable, como es el caso del EKF. Mismo caso para el uso de controladores, que dependen de si el sistema logra ser controlable para accionar el torque necesario para el sistema satelital.

2.7 Optimización Matemática

La optimización matemática es una rama de las matemáticas aplicadas que se enfoca en encontrar la mejor solución posible para un problema, generalmente formulado como la minimización de una función objetivo sujeta a ciertas restricciones [44]. El problema general se expresa como

$$\min_x f(x),$$

sujeito a las restricciones

$$g_i(x) \leq b_i, \quad i = 1, \dots, m,$$

donde $x = (x_1, \dots, x_n)$ representa el vector de variables de optimización, $f : R^n \rightarrow R$ es la función objetivo, y cada $g_i : R^n \rightarrow R$ junto con el valor escalar $b_i \in R$ define las restricciones que deben cumplir las variables para ser consideradas válidas.

2.7.1 Clasificación de problemas de optimización

Los problemas de optimización pueden clasificarse en distintas categorías dependiendo de las características de la función objetivo y las restricciones. A continuación se listan y definen algunos de ellos [45]:

Optimización Convexa y no Convexa: Un problema es convexo si la función objetivo y las restricciones forman un conjunto convexo, lo que asegura que cualquier mínimo local es también el mínimo global. Por otro lado, los problemas no convexos no cumplen con la propiedad de convexidad, lo que significa que pueden existir múltiples mínimos locales. La optimización no convexa es más compleja y suele requerir métodos numéricos avanzados.

Luego, se clasifican en base a la linealidad del problema, donde se encuentran las siguientes opciones:

Optimización Lineal (LP): Un problema de programación lineal busca minimizar o maximizar una función objetivo lineal sujeta a restricciones lineales. Tanto la función objetivo como las restricciones son expresadas como combinaciones lineales de variables.

- Función objetivo: $\min c^T x$ o $\max c^T x$
- Restricciones: $Ax \leq b$ (donde A es una matriz de coeficientes, x es el vector de variables, y b es un vector de valores límites)

Mixed Integer Linear Programming (MILP): En programación lineal entera mixta, algunas de las variables son continuas (pueden tomar cualquier valor real) y otras son enteras (solo pueden tomar valores enteros). Es una extensión de LP.

- Función objetivo y restricciones son similares a LP, pero algunas variables x_i deben ser enteras.

Quadratic Programming (QP): La programación cuadrática es una extensión de la programación lineal donde la función objetivo es cuadrática (tiene términos de segundo orden), pero las restricciones siguen siendo lineales.

- Función objetivo: $\min x^T Qx + c^T x$ o $\max x^T Qx + c^T x$
- Restricciones: $Ax \leq b$

Mixed Integer Quadratic Programming (MIQP): La programación cuadrática entera mixta combina la programación cuadrática con variables enteras. Algunas de las variables pueden ser continuas, y otras deben ser enteras, como en MILP.

- Función objetivo: cuadrática
- Restricciones: lineales
- Algunas variables deben ser enteras

Second-Order Cone Programming (SOCP): La programación de cono de segundo orden es una clase de problemas convexos donde la función objetivo es lineal, pero las restricciones incluyen desigualdades de tipo cónico, es decir, involucran normas Euclidianas o distancias cuadráticas.

- Función objetivo: lineal
- Restricciones: Involucran normas del tipo $\|A_i x + b_i\|_2^2 \leq c_i^T x + d_i$

Mixed Integer Second-Order Cone Programming (MISOCP): En la programación de cono de segundo orden entera mixta, algunas variables del problema SOCP deben ser enteras. Esto combina la complejidad de SOCP con la dificultad adicional de tener variables discretas.

Semidefinite Programming (SDP): En programación semidefinida, la función objetivo es lineal, pero las restricciones involucran matrices semidefinidas positivas. Un problema semidefinido involucra encontrar una matriz que minimice una función objetivo lineal bajo la condición de que la matriz sea semidefinida positiva.

- Función objetivo: lineal en una matriz X
- Restricciones: $X \succeq 0$ (la matriz X debe ser semidefinida positiva)

General Nonlinear Programming (NLP) : En la programación no lineal, tanto la función objetivo como las restricciones pueden ser no lineales. Este es un tipo de optimización más general y difícil de resolver, ya que puede haber múltiples óptimos locales y el problema puede no ser convexo.

- Función objetivo: $f(x)$ (no lineal)
- Restricciones: $g(x) \leq 0$ y $h(x) = 0$ (no lineales)

2.7.2 Optimizadores en Python

Para resolver problemas de optimización en el control, se utilizan diversas herramientas y optimizadores disponibles en Python. A continuación, se presentan dos:

- `Scipy.optimize.minimize` [46]: El módulo `scipy.optimize.minimize` proporciona una variedad de métodos para resolver problemas de optimización. Este optimizador permite la minimización de una función objetivo definida por el usuario, utilizando diferentes algoritmos de optimización, para así encontrar el mínimo local del problema a analizar. Este módulo es útil para problemas de optimización de funciones objetivo tanto lineales como no lineales y permite la implementación de restricciones y condiciones específicas según el problema.
- `Pyomo` [47, 48]: `Pyomo` es una biblioteca de Python para el modelado y solución de problemas de optimización matemática. A diferencia de `scipy.optimize`, `pyomo` permite definir problemas de optimización de manera más estructurada y flexible, especialmente para problemas complejos que involucran programación lineal, no lineal, entera y estocástica. `Pyomo` permite definir problemas de optimización mediante un enfoque declarativo, facilitando la expresión de restricciones y objetivos. Además, soporta una amplia gama de solucionadores, desde optimizadores simples hasta solvers avanzados.

3 Estado del Arte

En este capítulo se presentan diversas herramientas y metodologías utilizadas en la simulación de la dinámica orbital y de actitud de satélites, incluyendo su visualización a través de interfaces gráficas. Además de ofrecer métricas de desempeño (MoP) para el control de actitud, algunos softwares especializados facilitan el análisis de presupuestos de ingeniería, como los System Engineering envelopes (SE envelopes). Asimismo, se incluyen métodos y criterios para la selección y optimización de los componentes del subsistema de determinación y control de actitud (ADCS), fundamentales para el diseño eficiente de CubeSats y otras plataformas satelitales.

3.1 Spacecraft Control Toolbox

Spacecraft Control Toolbox (SCT) para MATLAB le permite diseñar, analizar y simular naves espaciales. Este producto es utilizado en todo el mundo por organizaciones líderes en investigación y desarrollo y fabricantes de naves espaciales. Se proporcionan más de dos mil funciones para dinámica, simulación, análisis y diseño de actitud y órbita. Puedes construir un satélite utilizando las herramientas gráficas CAD; diseñar y analizar los sistemas de control; realizar análisis de perturbaciones y pruebe el sistema de control en una simulación de seis grados de libertad, todo en el lenguaje de programación MATLAB [18]. Existen tres módulos disponibles, los cuales se muestran a continuación:

- *CubeSat Toolbox*: Es el producto más básico disponible, a un costo de 495 dólares. Está diseñado para CubeSat específicamente y sus principales características son el modelamiento de la dinámica y control de actitud (perturbaciones, controladores PID en conjunto con actuadores), propagación de órbita, cambios de marco de referencia, visualización (2D y 3D) y planeamiento de la misión. En la Figura 3.1, se presenta un resumen de los datos obtenidos por el simulador una vez creado el diseño 3D en CAD. Por otro lado, en la Figura 3.2, se muestra en específico la ventana de control del simulador, mostrando tanto los cuaterniones de rotación, como los torques y las razones de cambio de la rueda de reacción montada en el CubeSat.

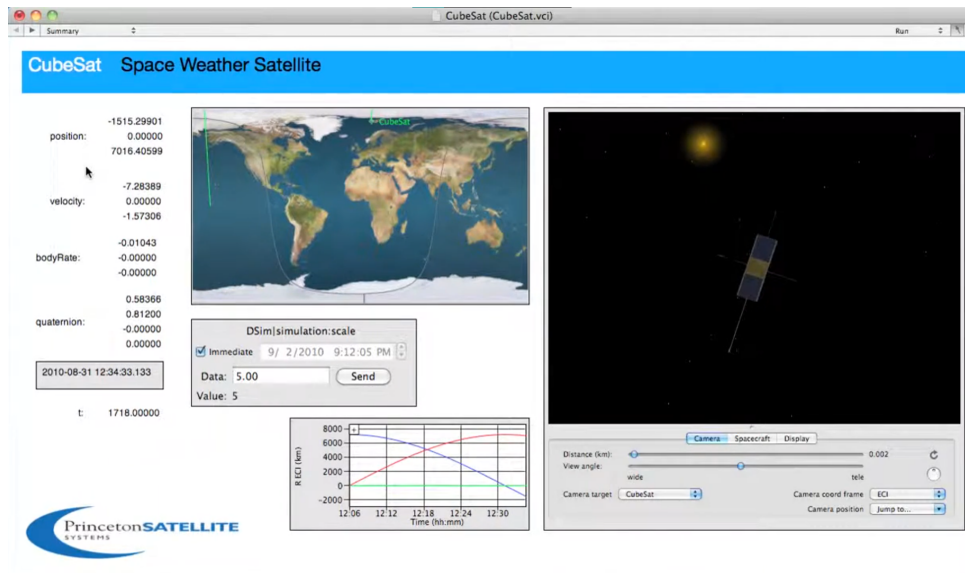


Figura 3.1 Resumen de la dinámica orbital y de actitud del satélite en CubeSat Toolbox.

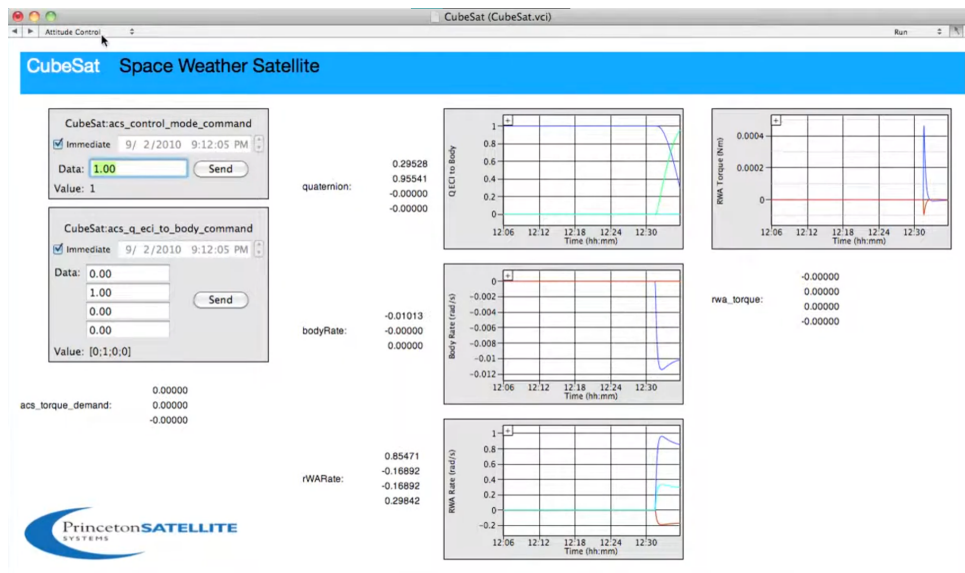


Figura 3.2 Ventana de control del simulador de un CubeSat en CubeSat Toolbox.

- *Spacecraft Control Toolbox versión académica:* La edición académica de la caja de herramientas contiene gran parte del software profesional Spacecraft Control Toolbox, incluidas las herramientas CAD, las funciones de diseño de control y dinámica de actitud, la mecánica orbital y el módulo CubeSat. Además, incluye esquemas de apuntamiento según marcos de referencia LVLH, sun-nadir pointing y latitude/longitude, y también otorga los pointing budgets más relevantes como la exactitud de apuntamiento. Esta versión cuesta 3995 dólares su adquisición y presenta un breve tutorial en el canal de YouTube de Princeton.
- *Spacecraft Control Toolbox versión profesional:* La edición profesional de Spacecraft Con-

trol Toolbox ofrece todo lo que hay en las versiones CubeSat y Academic, con la adición de herramientas avanzadas para la estimación de actitud y órbita, modelado de sensores y actuadores y análisis de subsistemas. Las aplicaciones de la caja de herramientas incluyen diseño de sistemas de control, simulación no lineal, análisis de órbitas y planificación de misiones, incluidas trayectorias interplanetarias, diseño y disposición de naves espaciales, estudios comerciales y visualización de actitudes y órbitas. La versión profesional cuesta 11995 dólares, siendo esta la más completa y a la vez más cara entre las tres mencionadas.

3.2 Ansys Systems Tool Kit (STK)

STK es una plataforma de software líder en el mercado diseñada para el modelado y análisis de sistemas complejos y sus interacciones en una variedad de dominios, incluyendo espacio, defensa y aplicaciones aeroespaciales. Esta herramienta proporciona una serie de capacidades avanzadas que permiten a los ingenieros, científicos y analistas modelar, simular y visualizar sistemas dinámicos de manera efectiva [49].

Es relevante ya que es capaz de modelar un satélite mediante su diseño 3D, mostrando gráficamente como este se propaga a través de la órbita. También es capaz de simular los cambios de actitud del satélite a través del tiempo y mostrar si efectivamente se está apuntando hacia la zona requerida de la tierra (exactitud de apuntamiento). En la Figura 3.3 se aprecia una imagen referencial de la simulación de un satélite en 3D y sus marcos de referencia cuerpo y LVLH. El costo de la licencia investigativa de este software es de alrededor de 4527 dólares por un año, que sería la versión útil para la realización del proyecto.



Figura 3.3 Imagen referencial de STK para simulación de un satélite.

3.3 Aerospace Blockset

Dentro de este módulo en MATLAB, existen librerías capaces de modelar, simular y analizar CubeSats con facilidad [19]. Algunas de las capacidades clave incluyen:

- *Modelado de CubeSats*: El modelo de plantilla es un ejemplo listo para simular que contiene un bloque de vehículo CubeSat. MATLAB permite a los usuarios modelar CubeSat

para proporcionar una opción de planificación de misión de alto nivel/creación rápida de prototipos para modelar y propagar rápidamente órbitas de satélites, un satélite a la vez. En este bloque se debe especificar el estado orbital inicial y modo de control de actitud (apuntamiento hacia el sol, nadir Tierra o personalizado), como se observa en la Figura 3.4.

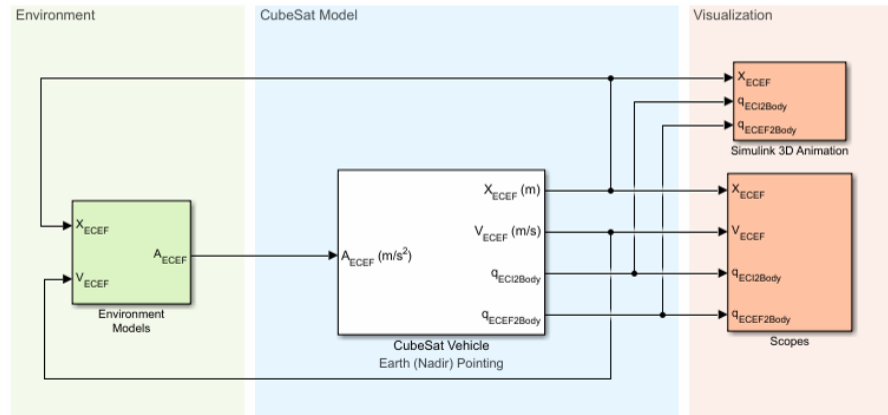


Figura 3.4 Plantilla de Simulink para el diseño de un CubeSat.

- **Simulación de Misiones:** El proyecto es un ejemplo listo para simular con visualización utilizando Simulink 3D Animation, como se muestra en la Figura 3.5. El ejemplo utiliza un subsistema de modelo de vehículo por defecto, pero también se puede utilizar el modelo de CubeSat diseñado en el paso anterior. Con el módulo, es posible simular misiones completas de CubeSats, lo que incluye la evaluación de la trayectoria, la dinámica orbital con modelos de perturbaciones incluidas y las operaciones a bordo. Esto es esencial para prever el comportamiento del CubeSat en el espacio y optimizar su rendimiento.

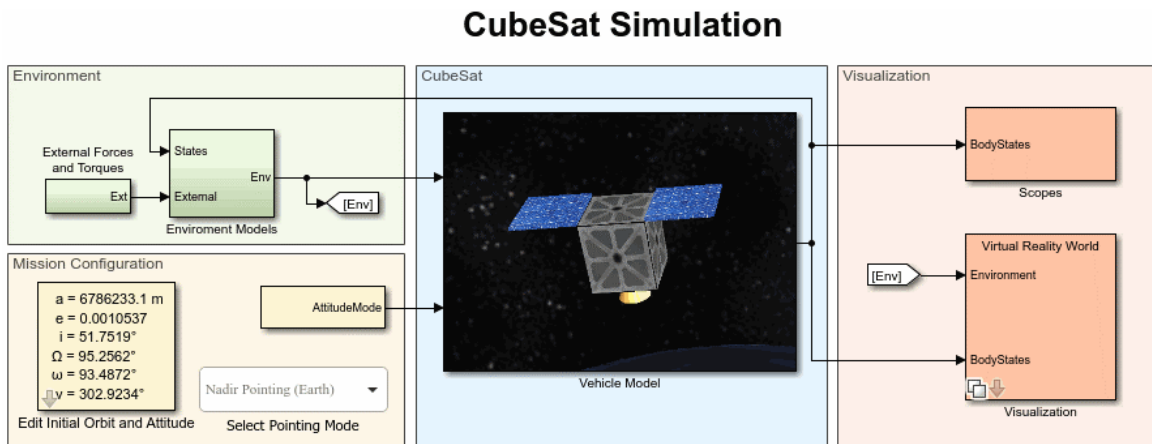


Figura 3.5 Plantilla de Simulink para la modelación y simulación del CubeSat en base a la visualización en Simulink 3D.

- **Proyecto utilizando Model Based System Engineering (MBSE) en CubeSat:** Es un ejemplo listo para simulación que muestra cómo modelar la arquitectura de una misión espacial

con System Composer y Aerospace Blockset. El proyecto hace referencia al Proyecto de simulación CubeSat para reutilizar modelos de subsistema, luego agrega una capa de arquitectura System Composer, vincula los requisitos del sistema con los componentes de la arquitectura y verifica los requisitos de la misión de nivel superior con Simulink Test™. El proyecto visualiza los resultados utilizando Simulink 3D Animation, escenarios satelitales de Aerospace Toolbox y Mapping Toolbox™. Todo esto se muestra según los bloques mostrados en la Figura 3.6.

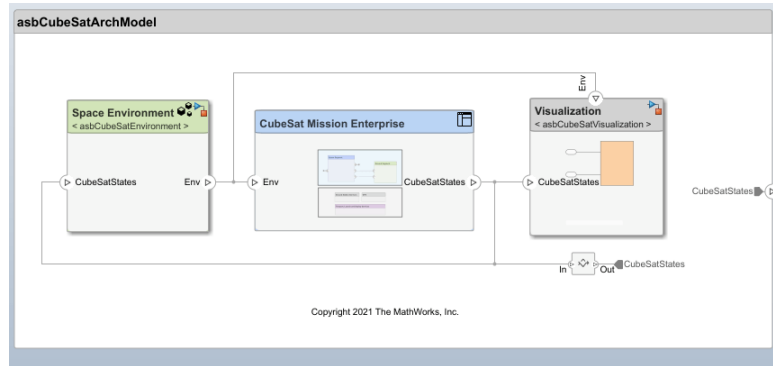


Figura 3.6 Visualización del MBSE de CubeSat.

Mediante la última librería mencionada se puede obtener la observación terrestre de un CubeSat, además de visualizar los budgets de costo como lo es el precio, la masa, la potencia tanto peak como promedio y el tamaño. También presenta en una interfaz gráfica los errores de determinación de actitud y de apuntamiento, además de los torques ejercidos por los actuadores (sean modelados o por defecto dentro del programa), como se muestra en la Figura 3.7. Así, se puede visualizar el resultado como muestra la Figura 3.8.

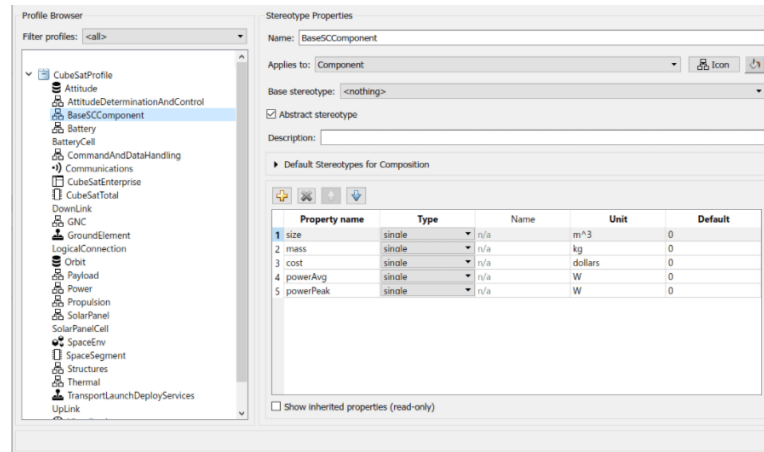


Figura 3.7 Perfil de CubeSat según los SE envelopes configurables.

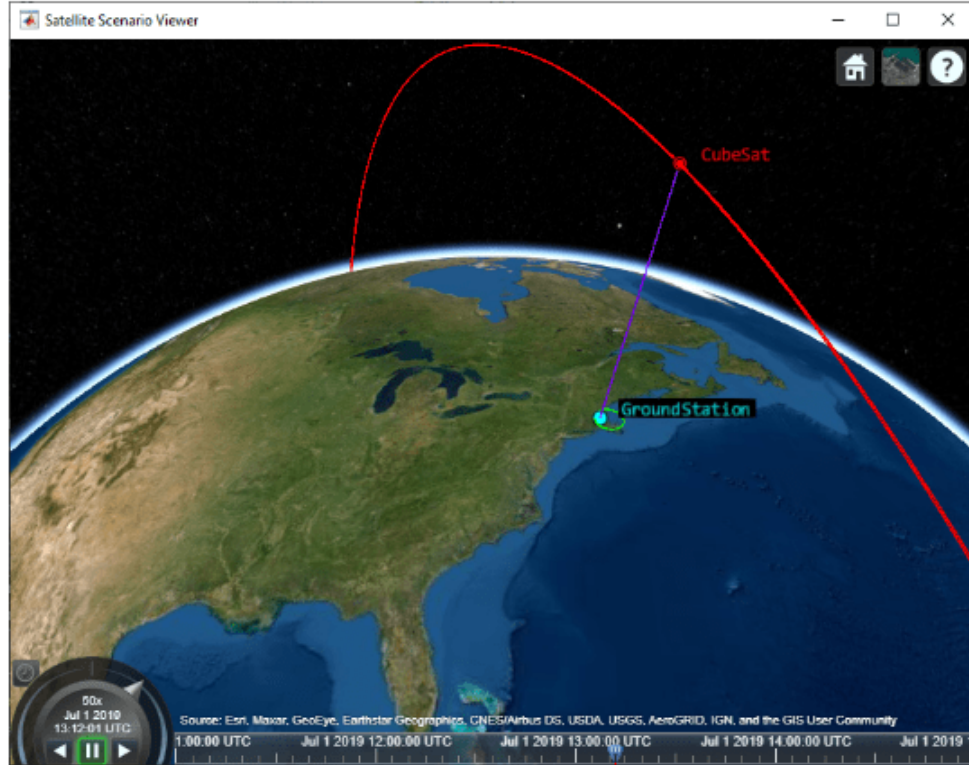


Figura 3.8 Simulación de apuntamiento de la Tierra hacia la estación terrestre.

3.4 Valispace

Valispace es una plataforma integral de ingeniería y gestión de proyectos diseñada para simplificar y optimizar el proceso de desarrollo de productos y sistemas, especialmente en industrias como la aeroespacial. La plataforma ofrece una variedad de herramientas y características poderosas que permiten a los equipos de ingeniería colaborar eficientemente, gestionar requisitos y parámetros críticos, y tomar decisiones informadas en tiempo real [50].

Las características claves son la gestión de datos en tiempo real, permitiendo el acceso en tiempo real y la colaboración de los miembros del equipo de ingeniería de forma actualizada. También permite a los equipos diseñar, analizar y optimizar sistemas de ingeniería complejos en base a requisitos que se imponen en el mismo programa al inicio del proyecto. Además, Valispace facilita el cálculo de parámetros críticos como costos, masa, potencia y tamaño, lo que es esencial en proyectos de ingeniería. Los resultados se pueden calcular y actualizar automáticamente a medida que se realizan cambios en el diseño.

3.5 Basilisk

El simulador Basilisk es un entorno de simulación avanzado, modular y extensible, desarrollado principalmente por el Laboratorio de Sistemas de Vehículos Espaciales de la Universidad de Colorado, Boulder. Su principal objetivo es facilitar la simulación de sistemas de naves espaciales, con un enfoque en la dinámica y el control de actitud. Este simulador ha sido ampliamente utilizado en investigaciones académicas y proyectos que requieren la modelación precisa de la

dinámica y control de vehículos espaciales [21].

Una de las principales ventajas de Basilisk es su capacidad para simular sistemas multi-plataforma y multi-cuerpo, permitiendo la modelación de la dinámica orbital y de actitud de diversos cuerpos en el espacio. Esto incluye la simulación de perturbaciones ambientales, así como el comportamiento de subsistemas complejos como el ADCS. Estas capacidades hacen que Basilisk sea especialmente útil para misiones que involucran satélites pequeños o CubeSats, donde las dinámicas precisas y los ajustes de actitud son críticos.

Otra de las fortalezas del simulador radica en su arquitectura modular. Cada subsistema —como los sensores, actuadores, o la dinámica orbital— está diseñado de manera independiente, lo que permite no solo la fácil personalización del código, sino también la posibilidad de integrar nuevos módulos o realizar simulaciones específicas de determinados componentes. Esta flexibilidad es clave para proyectos de investigación que requieren ajustes finos y configuraciones personalizadas.

El simulador también ofrece la capacidad de realizar simulaciones tanto en tiempo real como en tiempo no real, lo que resulta útil para diferentes tipos de aplicaciones. Las simulaciones en tiempo real permiten la implementación de pruebas de hardware-in-the-loop, mientras que las simulaciones en tiempo no real ofrecen mayor fidelidad para estudios más detallados.

3.6 Resumen de los simuladores disponibles

Para evaluar la viabilidad de utilizar algunos de los simuladores disponibles en el estado del arte para la implementación de la suite de simulación, se realizó una comparación basada en criterios clave como el costo, la dificultad de uso, la capacidad para entregar métricas de desempeño (MoP) de apuntamiento y el soporte a los System Engineering envelopes (SE envelopes), tal como se resume en la Tabla 3.1.

En términos de costo, las opciones más accesibles son CubeSat Toolbox y Aerospace Blockset, considerando que la suscripción académica de MATLAB está disponible para la Universidad de Concepción. No obstante, CubeSat Toolbox carece de análisis de MoP de apuntamiento, un requisito fundamental para el presente estudio. Además, dado que está desarrollado exclusivamente para MATLAB y la suite objetivo se pretende implementar en Python, su adaptación implicaría un esfuerzo considerable de reprogramación, lo que limita su conveniencia.

Por otro lado, Aerospace Blockset, aunque ofrece funcionalidades avanzadas para analizar métricas de apuntamiento, está basado en un entorno de Model-Based Systems Engineering (MBSE) que demanda un nivel medio-alto de conocimientos en ingeniería de sistemas. Esto, sumado a la rigidez del entorno que dificulta la implementación de lazos cerrados externos, restringe la flexibilidad requerida para integrar módulos de optimización personalizados.

Las opciones comerciales de mayor costo, como SCT académico y profesional, y Systems ToolKit, proveen capacidades robustas para análisis de apuntamiento y gestión de SE envelopes, pero su elevado precio y complejidad técnica las hacen menos accesibles para este proyecto.

Finalmente, Basilisk representa una alternativa gratuita con capacidades avanzadas para simular dinámicas orbitales y de actitud considerando diferentes configuraciones de hardware ADCS. Sin embargo, su elevada curva de aprendizaje, documentación técnica limitada y falta de un

enfoque nativo en optimización de sistemas, junto con la necesidad de configurar manualmente los parámetros de costo y rendimiento, presentan barreras significativas para su uso efectivo dentro del tiempo y recursos disponibles.

Tabla 3.1 Comparativa de software/simuladores para CubeSats.

Software/ simulador	Precio [USD]	Dificultad	MoP de apuntamiento	SE envelopes
<i>CubeSat Toolbox</i>	495 + Matlab	Media	No presenta	Potencia
SCT académico	3995 + Matlab	Alta	Exactitud de apuntamiento	Potencia y masa
SCT profesional	11995 + Matlab	Alta	Exactitud de apuntamiento/agilidad	Potencia, masa y tamaño
<i>Aerospace Blockset</i>	Matlab	Muy alta	Exactitud de apuntamiento	Todos, con imposición de requisitos
Systems ToolKit	4527 por año	Media	Exactitud de apuntamiento/agilidad	Potencia
Valispace	1295 por año	Media	No presenta	Todos
Basilisk	Gratis	Muy Alta	Configurable	Configurable

En resumen, aunque existen diversas herramientas que aportan funcionalidades valiosas para la simulación y análisis de CubeSats, ninguna de ellas satisface completamente los requerimientos específicos de integración, flexibilidad y optimización que demanda esta investigación. Por ello, se concluye que el desarrollo de una suite propia de simulación con capacidades de optimización en Python es la opción más adecuada, permitiendo un control total sobre los modelos, algoritmos y criterios de selección, y facilitando la adaptación a futuras necesidades y mejoras.

3.7 Métodos para la selección de componentes del ADCS

Además de las herramientas de simulación y análisis presentadas en este capítulo, la literatura especializada ofrece diversos enfoques metodológicos para la **selección óptima de componentes del sistema de determinación y control de actitud (ADCS)**. Estos métodos permiten identificar combinaciones eficientes de sensores y actuadores considerando criterios de desempeño, restricciones físicas (masa, volumen, potencia) y costos asociados. Se pueden clasificar en dos categorías principales: métodos clásicos y métodos modernos basados en técnicas computacionales.

Métodos clásicos

- **Matriz de decisión multicriterio (MCDM):** Técnica utilizada en el diseño conceptual para evaluar múltiples alternativas en función de distintos criterios. Herramientas como el Proceso Analítico Jerárquico (AHP) permiten asignar ponderaciones a criterios como la masa, costo, consumo energético y precisión de medición [22, 51].
- **Lógica difusa:** Útil para incorporar incertidumbre o ambigüedad en la evaluación de componentes. Se basa en reglas del tipo “si el consumo es bajo y la precisión alta, entonces

el componente es adecuado”. Ha sido aplicada en proyectos donde los requerimientos son cualitativos o difíciles de cuantificar [23, 52].

- **Análisis de Pareto:** Identifica conjuntos de soluciones que representan compromisos óptimos entre criterios en conflicto, como minimizar masa y maximizar exactitud de apuntamiento [24].

Métodos modernos

- **Feature importance y árboles de decisión:** Algunas investigaciones aplican algoritmos de clasificación supervisada para determinar qué variables (como dimensiones, precisión, o costo) son más relevantes en el cumplimiento de requisitos de actitud. Estos modelos permiten clasificar o predecir la idoneidad de componentes ADCS con base en datos históricos [25, 53].
- **Optimización basada en simulación:** Se integran modelos de simulación orbital y de actitud con algoritmos de optimización (como algoritmos genéticos o `scipy.optimize`) para seleccionar configuraciones que maximizan métricas de rendimiento bajo restricciones SE (masa, energía, costo) [26, 54].
- **Sistemas expertos e inferencia bayesiana:** Se han desarrollado sistemas que utilizan reglas de expertos o inferencia estadística para seleccionar componentes robustos frente a incertidumbre en condiciones orbitales y restricciones de misión [27, 55].

Tabla 3.2 Comparación entre métodos clásicos y modernos para la selección de componentes ADCS

Criterio	Métodos clásicos	Métodos modernos
Enfoque	Determinístico, basado en juicio experto	Basado en datos y simulación
Manejo de incertidumbre	Limitado, lógica difusa parcial	Elevado, modelos probabilísticos o de inferencia
Requiere simulación dinámica	No necesariamente	Sí, en la mayoría de los casos
Flexibilidad ante cambios en misión	Baja, depende de reglas fijas	Alta, modelos adaptables o aprendidos
Ejemplos	AHP, lógica difusa, análisis de Pareto	Árboles de decisión, optimización simulada, sistemas bayesianos
Aplicaciones típicas	Diseño conceptual temprano	Evaluación de configuraciones en fase de simulación

4 Diseño de la suite de simulación

4.1 Marco general del simulador

Para la construcción de la suite de simulación, se debe tener en cuenta tanto la dinámica orbital como la dinámica de actitud, para así tener conocimiento del posicionamiento y la velocidad del CubeSat a través del espacio, y de la actitud de este en su desplazamiento alrededor de la Tierra. Para ello se debe considerar los siguientes elementos como base:

- **Propagación orbital:** En primera instancia, el simulador busca conocer la posición del satélite alrededor de la Tierra, por lo que es necesario el uso de un propagador orbital adecuado capaz de entregar información sobre el vector posición y velocidad del satélite respecto a la Tierra. Además, se debe considerar las perturbaciones correspondientes para la altura del satélite analizado, que en este caso se acotará a órbitas bajas, con el objetivo de hacer el movimiento traslacional del satélite realista según las condiciones espaciales presentes.
- **Modelos orbitales:** Los modelos orbitales son necesarios para obtener la orientación del CubeSat. Con ello se obtendrán las representaciones necesarias de los sensores en el marco de referencia inercial.
- **Sensores:** Para determinar la actitud del satélite, se seleccionan sensores capaces de entregar información relevante sobre la inercia del satélite, el comportamiento físico del ambiente espacial o estrellas circundantes. Es relevante tanto para la estimación inicial como para el conocimiento de la actitud a través del tiempo y se simularán según los modelos orbitales analizados, teniendo en cuenta una rotación del marco de referencia inercial al marco de referencia del cuerpo y el ruido en la lectura de los componentes utilizados.
- **Algoritmos de estimación de actitud:** Estos son necesarios para trabajar los vectores entregados por los sensores y los modelos orbitales. Con estos se obtienen los cuaterniones que representan las rotaciones del satélite a través del tiempo. Se busca utilizar el EKF para la estimación de los estados posteriores, además de una mitigación del ruido en la obtención de la orientación del satélite.
- **Controladores y actuadores:** Ya obtenida la actitud del satélite a través del tiempo, se desea que el CubeSat apunte a una dirección en específico. Para ello se implementa el uso de un controlador capaz de realizar el control del satélite mediante la entrega del modelo dinámico y del actuador. Esto va de la mano con la selección del actuador a utilizar, para tener en cuenta las limitaciones de la acción de control al apuntar el satélite hacia la dirección deseada.

Teniendo en cuenta estos factores, la Figura 4.1 presenta un diagrama que resume la estructura de la suite de simulación, utilizada para obtener los resultados de los MoP de apuntamiento y los SE envelopes correspondientes. En primer lugar, se obtiene el vector de posición $\vec{r}$ del CubeSat mediante el propagador orbital, seguido por el uso de modelos orbitales para calcular $\vec{V}_{ref}$, el vector en el sistema de referencia inercial. Simultáneamente, a través de la lectura de los sensores, se obtienen los vectores $\vec{V}_{obs}$ en el marco de referencia del cuerpo. Posteriormente, se realiza el cambio de sistema de referencia de $\vec{V}_{ref}$ a LVLH (como se explicará en las secciones

siguientes), utilizando ambos vectores para la estimación del cuaternión q . Finalmente, se ejecuta la acción de control mediante el controlador seleccionado generando una señal o corriente I , considerando la restricción del torque τ ejercido por el actuador. En las próximas secciones se detallará el diseño y selección de cada componente de la suite de simulación.

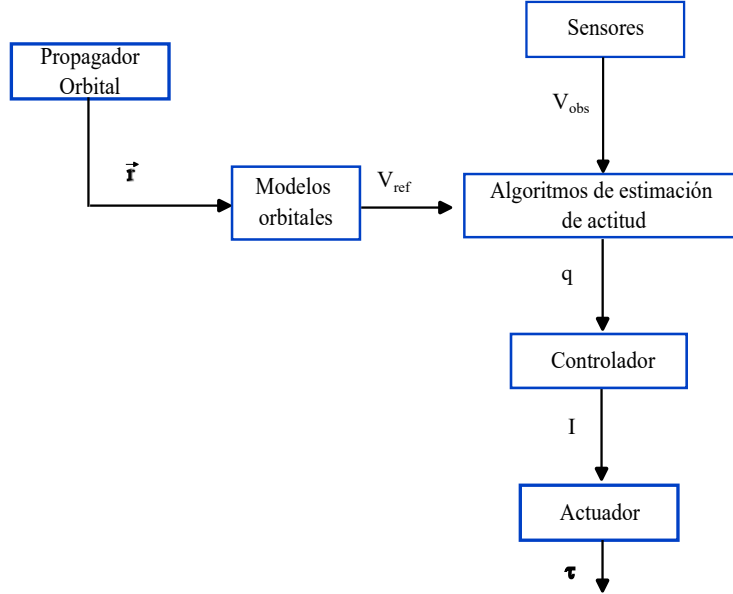


Figura 4.1 Esquema general de la base del simulador (Elaboración propia).

4.2 Propagador orbital

Para la simulación de la dinámica orbital, se debe considerar la implementación de un propagador capaz de entregar la posición y velocidad del satélite respecto a la Tierra incluyendo las perturbaciones relevantes a baja altura. Para ello se proponen las siguientes opciones:

Simplified General Perturbations 4 (SGP4) [56]: Es un modelo matemático que se utiliza ampliamente para predecir la posición y velocidad de los satélites. Utiliza las ecuaciones de movimiento de dos cuerpos, que describen cómo un satélite se mueve bajo la influencia de la gravedad de la Tierra, y luego aplica una serie de perturbaciones para tener en cuenta factores como la forma no esférica de la Tierra, el arrastre atmosférico, la radiación solar y los efectos gravitacionales de la interacción con otros cuerpos como el Sol o la Luna. El SGP4 utiliza los Two Line Elements (TLE) como entrada, que es un formato estándar utilizado para describir la órbita de un satélite en dos líneas, en conjunto con la fecha de inicio y el tiempo de propagación, para después aplicar una serie de ecuaciones y algoritmos que calculan los elementos orbitales futuros del satélite en función del tiempo.

FreeFlyer [57]: Es un programa de simulación espacial diseñado para visualizar y modelar varios escenarios, incluyendo, pero no limitándose a la propagación y maniobras de naves espaciales, análisis de cobertura y contacto, análisis interplanetario y la generación de diversos recursos visuales. Es capaz de propagar el movimiento del satélite a través del tiempo e incluir perturbaciones como J_2 , arrastre atmosférico, efecto multi-cuerpo y radiación solar.

Systems Tool Kit (STK) [49]: Es una plataforma ampliamente utilizada en el ámbito tanto aeronáutico como espacial. El STK es un simulador de Ansys utilizado para el diseño, planificación y simulación de misiones. Una de sus herramientas es la propagación del satélite a través del tiempo con una interfaz gráfica, incluyendo todas las perturbaciones presentes en el espacio.

Se descartan las opciones de los softwares FreeFlyer y STK debido a su alto costo monetario para la implementación del simulador solo para su uso en la dinámica orbital. Por lo tanto, se elige el SGP4 no solo por ser una opción gratuita, sino porque es fácil de implementar a cualquier satélite que se conozca su TLE. Esta entrega el movimiento traslacional del satélite a través del tiempo considerando las perturbaciones más importantes a bajas alturas en el marco de referencia TEME.

Para observar el uso del propagador SGP4, se utilizó como ejemplo un TLE del SUCHAI-3 obtenido de CelesTrak y se propagó durante una órbita tomando como fecha de inicio 1 de noviembre del 2023 a las 12 del mediodía. Se observa en la Figura 4.2a y 4.2b la posición y la velocidad respecto al marco de referencia ECI, al generar el cambio en el sistema de referencia TEME-ECI internamente utilizando la librería Astropy de Python.

4.3 Modelación de sensores

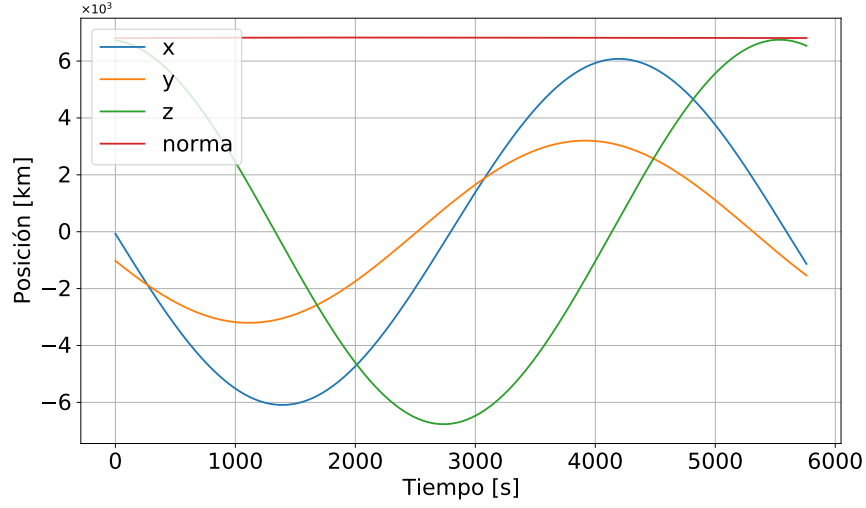
La determinación de la actitud del satélite requiere del uso de sensores capaces de proporcionar vectores de referencia confiables, los cuales se emplean para estimar los cuaterniones que describen la orientación del cuerpo. Para ello, se parte por modelar fenómenos físicos conocidos, como la posición del Sol y el campo magnético terrestre, con el fin de obtener los vectores correspondientes en el marco de referencia inercial (ECI). Estos vectores se transforman posteriormente al marco orbital local (LVLH) mediante rotaciones apropiadas.

Una vez definidos los vectores de referencia, se procede a generar los vectores de observación, que corresponden a las mediciones simuladas por los sensores del satélite. Estos se obtienen aplicando una rotación adicional desde el marco LVLH al sistema de referencia del cuerpo (Body), de acuerdo con la actitud instantánea del satélite. De esta forma, se simula el comportamiento de sensores como magnetómetros y sensores solares, los cuales entregan lecturas en el sistema de coordenadas del satélite, necesarias para los algoritmos de estimación de actitud.

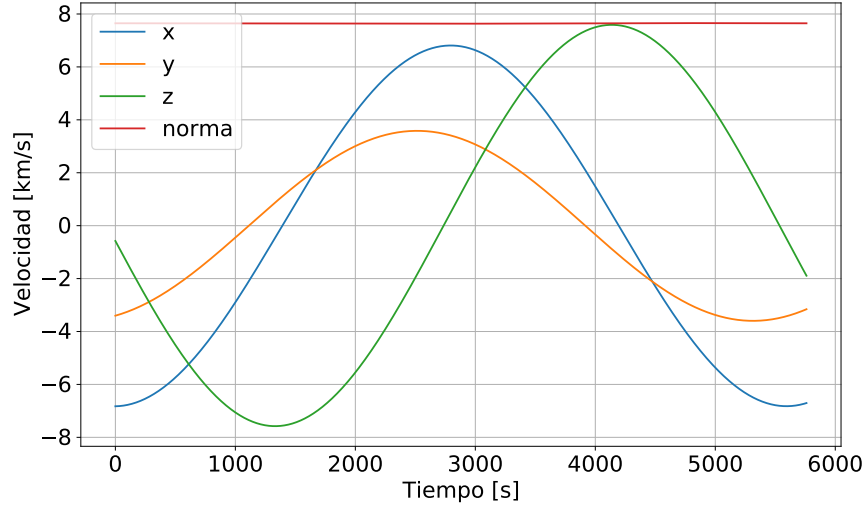
4.3.1 Vector sol

El modelo se basa principalmente en el movimiento del Sol respecto al sistema de referencia ECI. En la Figura 4.3 se muestra la relación entre el plano de la eclíptica, el ecuador celeste y los elementos orbitales fundamentales. Específicamente, se destaca la oblicuidad de la eclíptica (ϵ), que representa el ángulo entre el plano ecuatorial de la Tierra (referencia del marco ECI) y el plano orbital del Sol (eclíptica), así como la posición del equinoccio vernal, desde donde se mide la longitud eclíptica del Sol.

Lo primero a tener en cuenta en el modelo solar es la anomalía media del Sol, la cual se calcula según muestra la Ecuación 12. Esta corresponde al ángulo medido desde el perigeo, que describe la posición del Sol en su trayectoria orbital y depende únicamente del tiempo [15].



(a) Vector posición del satélite en ECI.



(b) Vector velocidad del satélite en ECI.

Figura 4.2 Propagación del SUCHAI-3 durante una órbita con fecha de inicio 01/11/23.

$$M_{\text{sun}} = M_{\text{sunEpoch}} + n_{\text{sun}}JD_{2000} = 357.528^\circ + 0.9856003^\circ \cdot JD_{2000} \quad (12)$$

En la ecuación anterior, M_{sunEpoch} es la anomalía media del Sol el 1 de enero del año 2000 (época J2000) al mediodía UTC, y n_{sun} es el movimiento medio del Sol. A partir de esta anomalía media, se puede calcular la longitud eclíptica del Sol λ_{sun} , definida en el plano de la eclíptica con respecto al equinoccio vernal:

$$\lambda_{\text{sun}} = 280.461^\circ + 0.9856474^\circ \cdot JD_{2000} + 1.915^\circ \sin(M_{\text{sun}}) + 0.020^\circ \sin(2M_{\text{sun}}) \quad (13)$$

Para proyectar esta posición sobre el marco ECI tridimensional, se debe considerar la oblicuidad de la eclíptica, definida como:

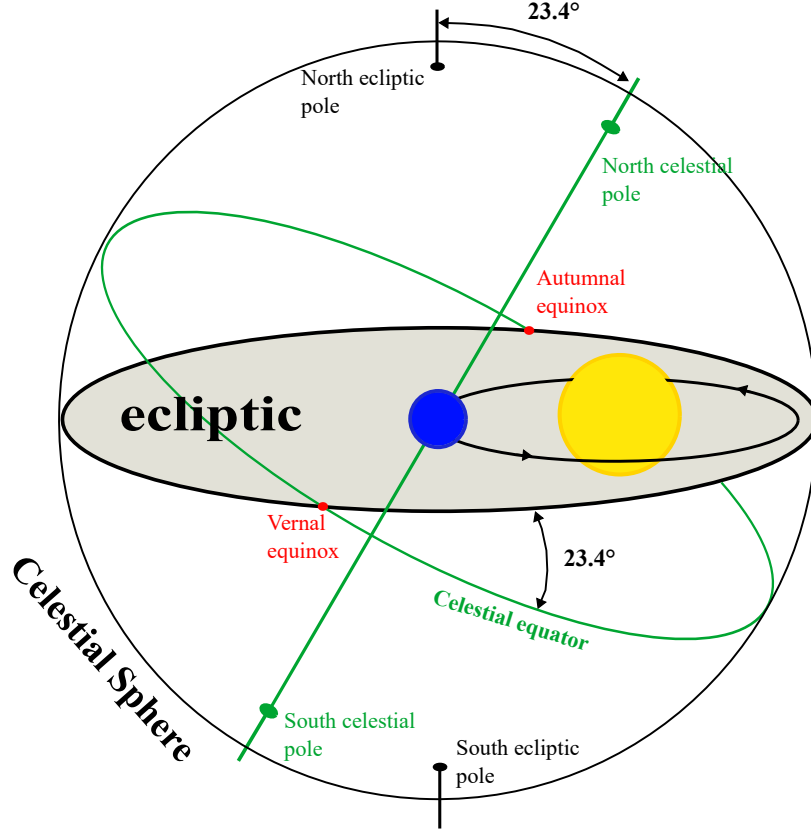


Figura 4.3 Plano eclíptico y marco de referencia ECI, con representación de la oblicuidad terrestre y equinoccios. Adaptado de [58].

$$\epsilon = 23.4393^\circ + 0.0000004^\circ \cdot JD_{2000} \quad (14)$$

Conociendo los ángulos λ_{sun} y ϵ , es posible obtener el vector unitario que representa la dirección del Sol en coordenadas ECI, según las siguientes componentes:

$$X_{\text{sun}} = \cos(\lambda_{\text{sun}}) \quad (15)$$

$$Y_{\text{sun}} = \cos(\epsilon) \sin(\lambda_{\text{sun}}) \quad (16)$$

$$Z_{\text{sun}} = \sin(\epsilon) \sin(\lambda_{\text{sun}}) \quad (17)$$

Este modelo del vector sol se observa en la Figura 4.4, en la cual se simula propagando por un año desde el 01 de noviembre del 2023, el cual en días julianos corresponde al día 2460250. Se observa en dicha gráfica un comportamiento oscilatorio, notándose un ciclo entero del movimiento del sol recién al término de la simulación.

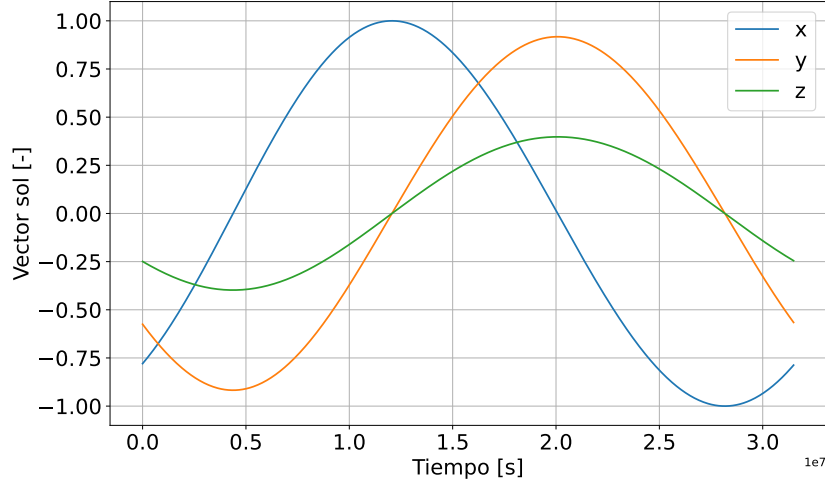


Figura 4.4 Componentes del vector sol en marco de referencia ECI simulados durante un año (Elaboración propia).

4.3.2 Fuerzas geomagnéticas de la tierra

Para el modelamiento de las fuerzas geomagnéticas de la tierra en distintas posiciones de la órbita de un satélite se requiere el uso del International Geomagnetic Reference Field (IGRF). Este campo de referencia consiste en un conjunto de coeficientes armónicos esféricos que pueden introducirse en un modelo matemático, para así describir la porción a gran escala y variable en el tiempo del campo magnético interno de la Tierra entre las épocas 1900 d.C. y el presente. El IGRF utilizado para la realización del simulador es el de decima tercera generación y se ha derivado de observaciones registradas por satélites, observatorios terrestres y estudios magnéticos [59].

El IGRF describe el principal Campo Geomagnético ($B(r, \theta, \varphi, t)$) que es producido por fuentes internas, principalmente dentro del núcleo de la Tierra. El IGRF es válido dentro y alrededor de la superficie de la Tierra, donde el campo geomagnético principal puede describirse como el gradiente de un potencial escalar, $B = -\nabla V$, y la función potencial $V(r, \theta, \varphi, t)$ se representa en la Ecuación 18 como una expansión en serie finita en términos de coeficientes armónicos esféricos, g_n^m, h_n^m , también conocidos como coeficientes de Gauss:

$$V(r, \theta, \varphi, t) = a \sum_{n=1}^N \sum_{m=0}^n \left(\frac{a}{r}\right)^{n+1} [g_n^m(t) \cos(m\varphi) + h_n^m(t) \sin(m\varphi)] P_n^m(\cos(\theta)) \quad (18)$$

Aquí, r, θ, φ se refieren a coordenadas en un sistema de coordenadas esféricas geocéntricas o geodésicas, siendo r la distancia radial desde el centro de la Tierra o la altura desde su superficie dependiendo de si son geocéntricas o geodésicas respectivamente, y θ, φ son la co-latitud y longitud geocéntricas respectivamente.

Las $P_n^m(\cos(\theta))$ son funciones de Legendre asociadas seminormalizadas de Schmidt de grado n y orden m . El parámetro N especifica el grado máximo de expansión armónica esférica y se eligió que fuera 10 hasta la época 1995 inclusive, después de lo cual aumenta a 13. Por otro lado, los

coeficientes de Gauss cambian en el tiempo y se proporcionan en unidades de nanoTesla (nT) en IGRF-13 en intervalos de época de 5 años. La dependencia temporal de estos parámetros se modela como lineal por partes y viene dada por:

$$g_n^m(t) = g_n^m(T_t) + (t - T_t)\dot{g}_n^m(T_t) \quad (19)$$

$$h_n^m(t) = h_n^m(T_t) + (t - T_t)\dot{h}_n^m(T_t) \quad (20)$$

Donde $g_n^m(T_t)$, $h_n^m(T_t)$ son los coeficientes de Gauss en la época T_t , que precede inmediatamente al tiempo t . Las épocas del modelo en IGRF-13 se proporcionan en múltiplos exactos de 5 años comenzando en 1900 y terminando en 2020, de modo que $T_t < t < T_t + 5$. Para $T_t < 2020$, los parámetros $\dot{g}_n^m(T_t)$, $\dot{h}_n^m(T_t)$ representan la aproximación lineal al cambio en los coeficientes de Gauss durante el intervalo de 5 años que abarca $[T_t, T_t + 5]$. Pueden calcularse en unidades de nanoTesla por año (nT/año) como:

$$\dot{g}_n^m(T_t) = \frac{1}{5} (g_n^m(T_t + 5) - g_n^m(T_t)) \quad (21)$$

$$\dot{h}_n^m(T_t) = \frac{1}{5} (h_n^m(T_t + 5) - h_n^m(T_t)) \quad (22)$$

Este procedimiento en Python viene entregado por la National Centers for Environmental Information (NCEI) [60] bajo el nombre de paquete `py.igrf` y se implementa en el simulador para el conocimiento de las fuerzas magnéticas ejercidas sobre la Tierra. Se requiere para el modelo insertar como parámetros de entrada las coordenadas geodésicas (r, θ, φ) en conjunto con el año de análisis, para así obtener siete salidas (declinación [°], inclinación [°], intensidad horizontal [nT], componente norte [nT], componente este [nT], componente vertical [nT] e intensidad total [nT]), siendo las componentes 4, 5 y 6 las fuerzas magnéticas en el marco de referencia NED.

Para conocer las coordenadas geodésicas y simular un receptor GPS sin ruido dentro del satélite, se hizo un cambio de sistema de referencia desde NED a ECI y luego a geodésicas. Utilizando la librería Skyfield de Python y conociendo la posición en la cual está el satélite respecto a la Tierra gracias al propagador SGP4, se conocen las coordenadas GPS en cada instante del movimiento traslacional del satélite. Para analizar el funcionamiento del modelo IGRF entregado en Python, se utilizó los vectores posición y velocidad de la sección 5.3.1 correspondiente al SUCHAI-3 para hacer el cambio de sistema ECI a geodésicas, obteniendo en la Figura 4.5 las fuerzas magnéticas del satélite a través del tiempo.

4.3.3 Cambio en sistema de referencia y modelación

Ya obtenidos los vectores del Sol y las fuerzas magnéticas en el sistema inercial (ECI), se procede a transformarlos al marco de referencia orbital local (LVLH). Para ello, se utiliza una matriz de rotación que relaciona el sistema ECI con LVLH, construida a partir de los vectores de posición (r) y velocidad (v) del satélite, obtenidos mediante el modelo SGP4 descrito en la Sección 4.2. Esta matriz de rotación se define como [9]:

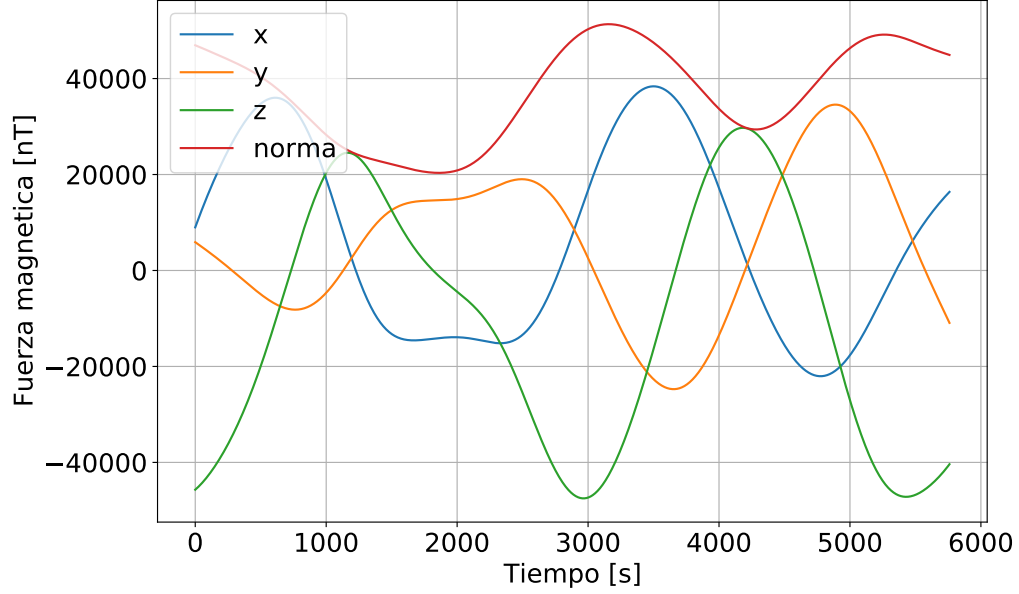


Figura 4.5 Componentes de las fuerzas magnéticas respecto a ECI que afectan al SUCHAI-3 con fecha inicial 01/11/2023 (Elaboración propia).

$$\vec{Z}_O = \frac{\vec{r}}{\|\vec{r}\|}, \quad \vec{X}_O = \frac{\vec{v} \times \vec{Z}_O}{\|\vec{v} \times \vec{Z}_O\|}, \quad \vec{Y}_O = \vec{Z}_O \times \vec{X}_O$$

$$R_{\text{ECI}}^O = \begin{bmatrix} \vec{X}_O^T \\ \vec{Y}_O^T \\ \vec{Z}_O^T \end{bmatrix}$$

Esta matriz permite transformar vectores desde ECI al sistema orbital. Si además se conoce el cuaternión que representa la actitud del cuerpo respecto al sistema inercial, denotado como $R_{\text{ECI}}^{\text{body}}$, entonces es posible obtener la actitud del cuerpo en relación al marco orbital mediante:

$$R_O^{\text{body}} = (R_{\text{ECI}}^O)^T \cdot R_{\text{ECI}}^{\text{body}}$$

La matriz R_O^{body} puede transformarse a un cuaternión q_O^{body} , lo que permite aplicar control sobre el sistema en el marco de referencia adecuado, como se detallará en secciones posteriores. Este cuaternión se actualiza dinámicamente en cada instante de tiempo, y es utilizado para transformar vectores desde el sistema orbital al cuerpo, tales como el campo magnético $\vec{B}_O$ y el vector solar $\vec{v}_{\text{sun},O}$:

$$(q_O^{\text{body}})^{-1} = [-q_0, -q_1, -q_2, q_3]$$

$$\vec{B}_{\text{body}} = (q_O^{\text{body}})^{-1} \cdot \vec{B}_O \cdot q_O^{\text{body}}, \quad \vec{v}_{\text{sun,body}} = (q_O^{\text{body}})^{-1} \cdot \vec{v}_{\text{sun},O} \cdot q_O^{\text{body}}$$

Los vectores $\vec{B}_{\text{body}}$ y $vecv_{\text{sun,body}}$ ya rotados permiten simular la lectura de sensores del sistema ADCS, bajo el supuesto de una calibración ideal. En esta investigación, tanto el magnetómetro como el sensor solar son modelados incorporando ruido gaussiano aditivo, permitiendo evaluar la influencia de la incertidumbre sensorial en el desempeño de los algoritmos de estimación y control.

El modelo de medición del magnetómetro se deriva de la siguiente ecuación general [32]:

$$\mathbf{B}_k = (I_3 + D^{\text{true}})^{-1} (\mathcal{O}^T A_k^{\text{true}} \mathbf{R}_k + \mathbf{bias}^{\text{true}} + \epsilon_k(\sigma_b)), \quad k = 1, 2, \dots, N \quad (23)$$

donde $\mathbf{B}_k$ representa la medición del campo magnético a tiempo t_k en el marco del cuerpo, y $\mathbf{R}_k$ corresponde al vector del campo magnético geomagnético en el marco inercial (ECI). La matriz A_k^{true} es la matriz de actitud real del cuerpo en el instante k , que permite transformar el vector de referencia desde el marco ECI al marco del cuerpo.

La matriz D^{true} modela errores de escala (en los elementos diagonales) y de no ortogonalidad (en los elementos fuera de la diagonal), mientras que $\mathcal{O}$ es una matriz ortogonal que representa una rotación fija asociada a una posible mala alineación del sensor respecto al cuerpo. El vector $\mathbf{bias}^{\text{true}}$ representa el sesgo estático del magnetómetro, y el término $\epsilon_k(\sigma_b)$ corresponde al ruido gaussiano de media cero y desviación estándar σ_b , que se asume con distribución normal multivariada.

Sin pérdida de generalidad, se puede asumir que D^{true} es simétrica, ya que cualquier componente antisimétrica puede ser absorbida por $\mathcal{O}$.

Para simplificar el análisis en esta investigación, se asume un magnetómetro idealmente calibrado, es decir:

$$D^{\text{true}} = 0, \quad \mathcal{O} = I_3, \quad \mathbf{bias}^{\text{true}} = 0$$

bajo estas condiciones, el modelo se reduce a:

$$\mathbf{B}_k = \vec{B}_{\text{body}} + \epsilon_k(\sigma_b) \quad (24)$$

Esta simplificación implica que las únicas fuentes de error en la medición provienen del ruido, permitiendo estudiar de manera directa el impacto de la incertidumbre en las mediciones sobre la estimación de actitud.

En este contexto, la desviación estándar del ruido del sensor, σ_b , se convierte en la variable clave a ser optimizada dentro del proceso de simulación. Esta representa la incertidumbre esperada en las lecturas del magnetómetro y es sistemáticamente variada para evaluar su impacto sobre el desempeño del algoritmo de estimación de actitud.

Por su parte, En este trabajo el **sensor solar** adopta un modelo simplificado pero realista, que permite representar diversas configuraciones de sensores (finos, gruesos o híbridos), manteniendo flexibilidad en el entorno de simulación y facilitando la evaluación comparativa de componentes con distintos parámetros de masa, consumo y precisión. Se modela como una medición directa del vector solar en el marco del cuerpo, afectada también por ruido gaussiano:

$$\vec{v}_{\text{sun},k} = \vec{v}_{\text{sun,body},k} + \epsilon_k(\sigma_s) \quad (25)$$

Los sensores solares son ampliamente utilizados en CubeSats para estimar orientación, ya sea mediante fotodiodos o diseños ópticos tipo estenopeico, los cuales permiten obtener direcciones solares con diferentes niveles de precisión y complejidad [61, 62].

4.4 Algoritmos de estimación y control satelital

Una vez obtenidas las mediciones de los sensores a bordo del satélite, es necesario procesar dicha información para estimar su estado actual —específicamente, su orientación y velocidad angular— con respecto a un marco de referencia inercial u orbital. Esta tarea se lleva a cabo mediante algoritmos de estimación que combinan modelos dinámicos y mediciones ruidosas para proporcionar una estimación precisa de la actitud.

A partir de esta estimación, se implementa un sistema de control automático que determina las acciones necesarias para alinear el satélite con una actitud objetivo, usualmente definida en función del marco de referencia orbital local (LVLH). En conjunto, los algoritmos de estimación y control permiten mantener la orientación deseada del satélite frente a perturbaciones externas y errores acumulados, asegurando así el cumplimiento de los requerimientos de la misión.

4.4.1 Controladores y actuadores

Se ha optado por utilizar el controlador Proporcional-Derivativo (PD) debido a su simplicidad y a su capacidad para orientar el satélite hacia una posición específica. Adicionalmente, se ha simulado el controlador Linear Quadratic Regulator (LQR) para obtener una matriz de ganancia que garantice un control estable en ambos actuadores simulados.

Para utilizar un controlador, es fundamental comprender la linealización del sistema, un proceso útil para simplificar modelos no lineales como en (5) y (7) y facilitar el diseño de estrategias de control. Este proceso se basa en la aproximación de Taylor [63], que consiste en linealizar las ecuaciones alrededor de un punto de equilibrio.

Esto implica calcular las derivadas parciales de las funciones de estado con respecto a las variables de estado y de control. Las matrices resultantes, $A = \frac{\partial f(x,u)}{\partial x}$ y $B = \frac{\partial f(x,u)}{\partial u}$, representan la dinámica linealizada del sistema, generando así el modelo Espacio Estado (EE) mostrado a continuación.

$$\dot{x} = Ax + Bu$$

Es importante destacar que la linealización proporciona una descripción local del comportamiento del sistema alrededor del punto de equilibrio, y por lo tanto, la validez de este enfoque depende de la proximidad del estado actual del sistema al punto de equilibrio.

Además, se optan por el magnetorquer [14] y la rueda de reacción [64, 65] como actuadores, debido a la documentación encontrada sobre su utilización y su uso mayoritario en nanosatélites. Para conocer el álgebra y las consideraciones del diseño del control lineal a mayor detalle para cada actuador, se recomienda revisar el Anexo D y E respectivamente.

Magnetorquer

El punto de equilibrio para el caso del modelo dinamico con magnetorquer se muestra a continuación, cuyos valores representan que no existe diferencia de orientación entre los marcos de referencia del cuerpo y orbital, además de lograr que se minimice tanto la acción de control como la velocidad angular [14].

$$q_{co} = [0, 0, 0, 1] \quad u = [0, 0, 0] \quad \omega_{co} = [0, 0, 0]$$

Los torques de control en sus tres componentes se representan en (26), (27) y (28).

$$\tau_{x,mt} = \left(\frac{b_x m_z - b_z m_x}{\|b\|} \right) b_z - \left(\frac{b_y m_x - b_x m_y}{\|b\|} \right) b_y \quad (26)$$

$$\tau_{y,mt} = - \left(\frac{b_z m_y - b_y m_z}{\|b\|} \right) b_z + \left(\frac{b_y m_x - b_x m_y}{\|b\|} \right) b_x \quad (27)$$

$$\tau_{z,mt} = \left(\frac{b_z m_y - b_y m_z}{\|b\|} \right) b_y - \left(\frac{b_x m_z - b_z m_x}{\|b\|} \right) b_x \quad (28)$$

Donde b son las fuerzas magnéticas en el marco de referencia del cuerpo y m es el momento dipolar del magnetorquer. Evaluando los valores deseados de q_{co} y ω_{co} en las matrices A y B mencionadas en el Anexo D, se obtiene:

$$A = \begin{bmatrix} 0 & 0 & 0 & 0.5 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.5 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.5 \\ 6\omega_{(0,o)}^2[I_z - I_y] & 0 & 0 & 0 & 0 & 0 \\ 0 & 6\omega_{(0,o)}^2[I_z - I_x] & 0 & 0 & 0 & \frac{\omega_{(0,o)}(I_x - I_y)}{I_z} - \omega_{(0,o)} \\ 0 & 0 & 0 & 0 & \frac{\omega_{(0,o)}(I_x - I_z)}{I_y} + \omega_{(0,o)} & 0 \end{bmatrix}$$

$$\bar{B} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ \frac{-b_z^2 - b_y^2}{\|b\|I_x} & \frac{b_x b_y}{\|b\|I_y} & \frac{b_x b_z}{\|b\|I_z} \\ \frac{b_x b_y}{\|b\|I_x} & \frac{-b_z^2 - b_x^2}{\|b\|I_y} & \frac{b_y b_z}{\|b\|I_z} \\ \frac{b_x b_z}{\|b\|I_x} & \frac{b_y b_z}{\|b\|I_y} & \frac{-b_y^2 - b_x^2}{\|b\|I_z} \end{bmatrix}$$

Debido a que las fuerzas magnéticas de la Tierra varían en el tiempo según la posición del satélite en su órbita, la matriz $\bar{B}(t)$, que relaciona el momento dipolar del magnetorquer con el torque ejercido sobre el cuerpo, también varía a lo largo de la simulación. Sin embargo, como

el diseño del controlador requiere una matriz constante en el modelo linealizado, se adopta una aproximación promediada de $\bar{B}(t)$ sobre un intervalo de tiempo representativo T . Esta aproximación se expresa como:

$$B = \frac{1}{T} \int_{t_0}^{t_0+T} \bar{B}(t) dt$$

La elección de este intervalo de integración está asociada al período orbital, donde se espera una evolución representativa del vector campo magnético. Esta técnica ha sido utilizada en estudios previos donde se busca un equilibrio entre simplicidad computacional y robustez del controlador frente a perturbaciones temporales [14, 66].

Cabe destacar que esta matriz promediada B se emplea tanto en el diseño del controlador como en la implementación del estimador. Esta decisión responde a la necesidad de coherencia entre el modelo dinámico utilizado en ambos algoritmos, y permite una implementación más eficiente al evitar la recalibración de matrices dinámicas en cada instante de tiempo.

Para el caso del controlador LQR, con las matrices A y B y las matrices Q y R entregadas por el diseñador del control, se logra obtener la matriz de ganancia (K) y ejercer el control en el modelo dinámico.

Por otro lado, para el controlador PD, se debe representar el modelo espacio estado de la siguiente manera:

$$\dot{x} = \hat{A}x$$

Con:

$$\hat{A} = A + BK$$

$$K = \begin{bmatrix} K_{p,x} & 0 & 0 & K_{d,x} & 0 & 0 \\ 0 & K_{p,y} & 0 & 0 & K_{d,y} & 0 \\ 0 & 0 & K_{p,z} & 0 & 0 & K_{d,z} \end{bmatrix}$$

Las constantes K_p y K_d son las constantes proporcionales y derivativas del controlador, y desempeñan un papel crucial en la determinación de la estabilidad del sistema. Estas constantes se ajustan de manera que los valores propios de la matriz $A+BK$ sean negativos, lo que garantiza la estabilidad del sistema en lazo cerrado.

Los valores propios negativos son indicativos de que el sistema es asintóticamente estable, lo que significa que las respuestas transitorias del sistema convergen hacia el estado de equilibrio deseado sin oscilaciones ni divergencias. En otras palabras, los valores propios negativos aseguran que el sistema retorne a su estado de equilibrio después de cualquier perturbación, lo que es fundamental para el funcionamiento adecuado del sistema de control, con un rendimiento estable y robusto [67].

Ruedas de reacción

El punto de equilibrio para el caso del modelo dinamico con rueda de reacción se muestra a continuación, cuyos valores representan que no existe diferencia de orientación entre los marcos de referencia del cuerpo y orbital, además de lograr que se minimice tanto la acción de control como la velocidad angular del satélite [64, 65, 68].

$$q_{co} = [0, 0, 0, 1] \quad u = [0, 0, 0] \quad \omega_{co} = [0, 0, 0]$$

Los torques de control son ejercidos por cada rueda de reacción de manera directa, siendo colocados en los tres ejes de referencia del marco de referencia del cuerpo del CubeSat por simplicidad. De esta manera, la rueda modelada en el eje x se representa mediante $\tau_{x,rw}$. Mismo caso para $\tau_{y,rw}$ y $\tau_{z,rw}$

Como se observa para los puntos de equilibrio, en este caso son seis estados que se reemplazan en la matriz de estado A y B obtenida en el Anexo E, generando como resultado las siguientes matrices:

$$A = \begin{bmatrix} 0 & 0 & 0 & 0.5 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.5 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.5 \\ 6\omega_{(0,o)}^2[J_z - J_y] & 0 & 0 & 0 & 0 & 0 \\ 0 & 6\omega_{(0,o)}^2[J_z - J_x] & 0 & 0 & 0 & \frac{\omega_{(0,o)}(J_x - J_y)}{J_z} - \omega_{(0,o)} \\ 0 & 0 & 0 & 0 & \frac{\omega_{(0,o)}(J_x - J_z)}{J_y} + \omega_{(0,o)} & 0 \end{bmatrix}$$

$$B = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ \frac{1}{J_x} & 0 & 0 \\ 0 & \frac{1}{J_y} & 0 \\ 0 & 0 & \frac{1}{J_z} \end{bmatrix}$$

Una vez obtenidas las matrices A y B, es posible utilizar un controlador LQR para guiar el sistema hacia el punto de equilibrio. Para ello, se debe diseñar el controlador seleccionando adecuadamente las matrices de ponderación Q y R, las cuales permiten ajustar el compromiso entre el rendimiento y el esfuerzo de control.

Es importante señalar que para ambos actuadores, la acción de control u se determina utilizando el vector de estado estimado, lo que permite aplicar torques realistas basados en las estimaciones obtenidas a través del EKF. Además, el estado 'real' es aquel que simulará las mediciones de los sensores, mientras que el 'estimado' sera el que se utiliza en cada paso de tiempo del EKF.

4.4.2 Utilización del EKF

Dado que los algoritmos de control requieren conocer el estado actual del satélite —en particular, su orientación—, se hace necesario estimar estas variables a partir de las mediciones de los sensores disponibles. Para este propósito, la suite de simulación implementa un algoritmo de estimación de actitud basado en el Filtro de Kalman Semiextendido (SEKF), que permite aproximar el estado del sistema dinámico de forma recursiva en cada instante de tiempo.

Para iniciar el proceso de estimación, se entregan dos condiciones iniciales: un cuaternión “real” que describe la orientación verdadera del satélite, y un cuaternión “estimado”, que representa el conocimiento inicial aproximado del sistema. La evolución de ambos permite cuantificar el error de estimación y evaluar la efectividad del filtro. Para una mejor interpretación física de la actitud, tanto la simulación como el análisis de resultados se complementan con la conversión de cuaterniones a ángulos de Euler (vista en el Anexo C).

El vector de estado adoptado considera la parte vectorial del cuaternión y los componentes de la velocidad angular, eliminando el componente escalar del cuaternión al ser dependiente de los demás mediante

$$q_3 = \sqrt{1 - q_0^2 - q_1^2 - q_2^2}.$$

Esta formulación, tomada del Filtro de Kalman Extendido Truncado (TEKF) [38], permite mantener la observabilidad del sistema, reduciendo la dimensión del vector de estado de siete a seis

$$x = \begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ \omega_0 \\ \omega_1 \\ \omega_2 \end{bmatrix}.$$

La dinámica de actitud del satélite, de naturaleza no lineal, se expresa en espacio de estados como

$$\dot{x} = f(x, u, t) + w, \quad y = h(x, t) + v,$$

donde u es la entrada de control, w y v son los ruidos de proceso y medición respectivamente, f representa el modelo dinámico, y h el modelo de medición.

El filtro fue implementado de manera recursiva utilizando una versión discretizada de los modelos dinámicos del satélite, diseñada para ser compatible con ambos tipos de actuadores simulados. En cada caso, se definieron matrices de estado $\mathbf{A}$ y control $\mathbf{B}$ específicas, evaluadas en el punto de equilibrio, las cuales permanecen invariantes en el tiempo,

$$\dot{x} = Ax_k + Bu_k + w, \quad y = h(x_k) + v.$$

Sin embargo, las mediciones de sensores, en particular del magnetómetro, varían en el tiempo dependiendo de la posición del satélite (el cual se asume conocida para evitar mas estados) respecto al campo magnético terrestre. Por lo tanto, la matriz de observación $\mathbf{C}_k$ (o H_k) debe actualizarse en cada iteración mediante la linealización del modelo de medición, caracterizando al algoritmo como un Filtro de Kalman Semiextendido (SEKF). La matriz de covarianza del proceso se define como,

$$Q = \begin{bmatrix} \sigma_\omega^2 \Delta t + \frac{1}{3} \sigma_\beta^2 \Delta t^3 \cdot I_{3 \times 3} & -\frac{1}{2} \sigma_\beta^2 \Delta t^2 \cdot I_{3 \times 3} \\ -\frac{1}{2} \sigma_\beta^2 \Delta t^2 \cdot I_{3 \times 3} & \sigma_\beta^2 \Delta t \cdot I_{3 \times 3} \end{bmatrix}$$

y la predicción de la covarianza a priori se calcula como,

$$P_{k+1}^- = A P_k^+ A^T + Q.$$

Para la actualización con medición, se utiliza la matriz de covarianza R y la matriz de observación H_k , calculada con base en los vectores medidos de campo magnético (b^B) y solar (s^B),

$$R = \begin{bmatrix} \sigma_b^2 I_{3 \times 3} & 0_{3 \times 3} \\ 0_{3 \times 3} & \sigma_s^2 I_{3 \times 3} \end{bmatrix}$$

$$H_k = \begin{bmatrix} -2S(b^B) & 0_{3 \times 3} \\ -2S(s^B) & 0_{3 \times 3} \end{bmatrix}$$

Donde las matrices $S(\cdot)$ representan la forma antisimétrica del vector medido.

$$S(v) = \begin{bmatrix} 0 & -v_z & v_y \\ v_z & 0 & -v_x \\ -v_y & v_x & 0 \end{bmatrix}$$

La matriz de ganancia de Kalman se calcula como,

$$K_k = P_k^- H_k^T (H_k P_k^- H_k^T + R)^{-1}$$

La corrección del estado estimado es:

$$\delta x = \begin{bmatrix} \delta q \\ \delta \omega \end{bmatrix} = K_k \left(\begin{bmatrix} b_k^B \\ s_k^B \end{bmatrix} - \begin{bmatrix} \hat{b}_k^B \\ \hat{s}_k^B \end{bmatrix} \right)$$

Donde b_k^B y s_k^B son las mediciones de los sensores, y $\hat{b}_k^B$ y $\hat{s}_k^B$ son las estimaciones de los vectores campo magnético y solar, respectivamente, rotados al sistema de referencia del cuerpo mediante los cuaterniones estimado y que generalmente es representado como $h(x)$

Para conservar la unidad del cuaternión, se adopta una corrección multiplicativa (MEKF) en lugar de aditiva, mientras que para la velocidad angular se mantiene la corrección original. Las estimaciones a posteriori se definen a continuación,

$$q_k^+ = \left[\frac{\delta q_k}{\sqrt{1 - \delta q_k^T \cdot \delta q_k}} \right] \otimes q_k^- \quad (29)$$

$$\omega_k^+ = \omega_k^- + \delta \omega_k \quad (30)$$

Finalmente, la covarianza posterior se actualiza como:

$$P_k^+ = (I - K_k H_k) P_k^-$$

Esta matriz P_k^+ se utiliza como condición inicial para el siguiente paso de tiempo. En conjunto, esta implementación híbrida SEKF/TEKF/MEKF permite una estimación robusta y consistente de la actitud y velocidades angulares del CubeSat, respetando tanto la estructura del sistema dinámico como las propiedades geométricas del cuaternión, y adaptándose a las variaciones en los sensores simulados durante la misión.

4.5 Estructura general de la suite de simulación de actitud

La Figura 4.6 muestra la estructura general de la suite de simulación de actitud, la cual integra el propagador orbital, el modelo de dinámica de actitud y los módulos de estimación y control. La simulación comienza con el cálculo de los vectores de posición $\mathbf{r}$ y velocidad $\mathbf{v}$ del CubeSat mediante el propagador SGP4, que recibe como entrada el conjunto TLE del satélite, la fecha y hora de simulación, y el tiempo de propagación t .

A partir de la posición orbital propagada, se calculan el vector del campo magnético terrestre $\mathbf{b}_e$ y el vector solar $\mathbf{s}_e$ en el sistema de referencia inercial (ECI). Estos vectores se transforman luego al sistema de referencia orbital, obteniéndose $\mathbf{b}_o$ y $\mathbf{s}_o$. Finalmente, utilizando el cuaternión de actitud $\mathbf{q}(t)$, los vectores son rotados al sistema de referencia del cuerpo del satélite, resultando en $\mathbf{b}_b$ y $\mathbf{s}_b$, que representan las mediciones reales del campo magnético y del vector solar por parte de los magnetómetros y sensores solares a bordo.

Para simular el proceso de medición a bordo, se aplican modelos de sensores a estos vectores en el sistema del cuerpo, introduciendo ruido para generar las mediciones observadas $\bar{\mathbf{b}}_b$ y $\bar{\mathbf{s}}_b$. Estas mediciones ruidosas se utilizan como entradas en el filtro de Kalman, que estima el vector de estado $\mathbf{x}_k$, compuesto por la velocidad angular $\boldsymbol{\omega}(t)$ y el cuaternión de actitud $\mathbf{q}(t)$.

El estado estimado $\mathbf{x}_k$ se emplea para calcular la acción de control $\mathbf{u}_k = \mathbf{K}\mathbf{x}_k$ mediante un controlador, donde $\mathbf{K}$ es la matriz de ganancias del controlador. Esta señal de control se retroalimenta tanto al modelo dinámico como al filtro de Kalman, permitiendo avanzar el estado del sistema y mejorar la estimación en los pasos de tiempo siguientes.

En la figura, los bloques y flechas en rojo representan la propagación de la dinámica de actitud verdadera y la generación de mediciones ideales. Los bloques azules corresponden al lazo de estimación y control, mientras que las conexiones en verde indican la transformación de los vectores ambientales desde el entorno orbital al sistema de referencia del satélite.

Los subíndices utilizados indican el sistema de referencia asociado a cada vector: e se refiere al sistema inercial (ECI), o al sistema orbital, y b al sistema del cuerpo del satélite. Por ejemplo,

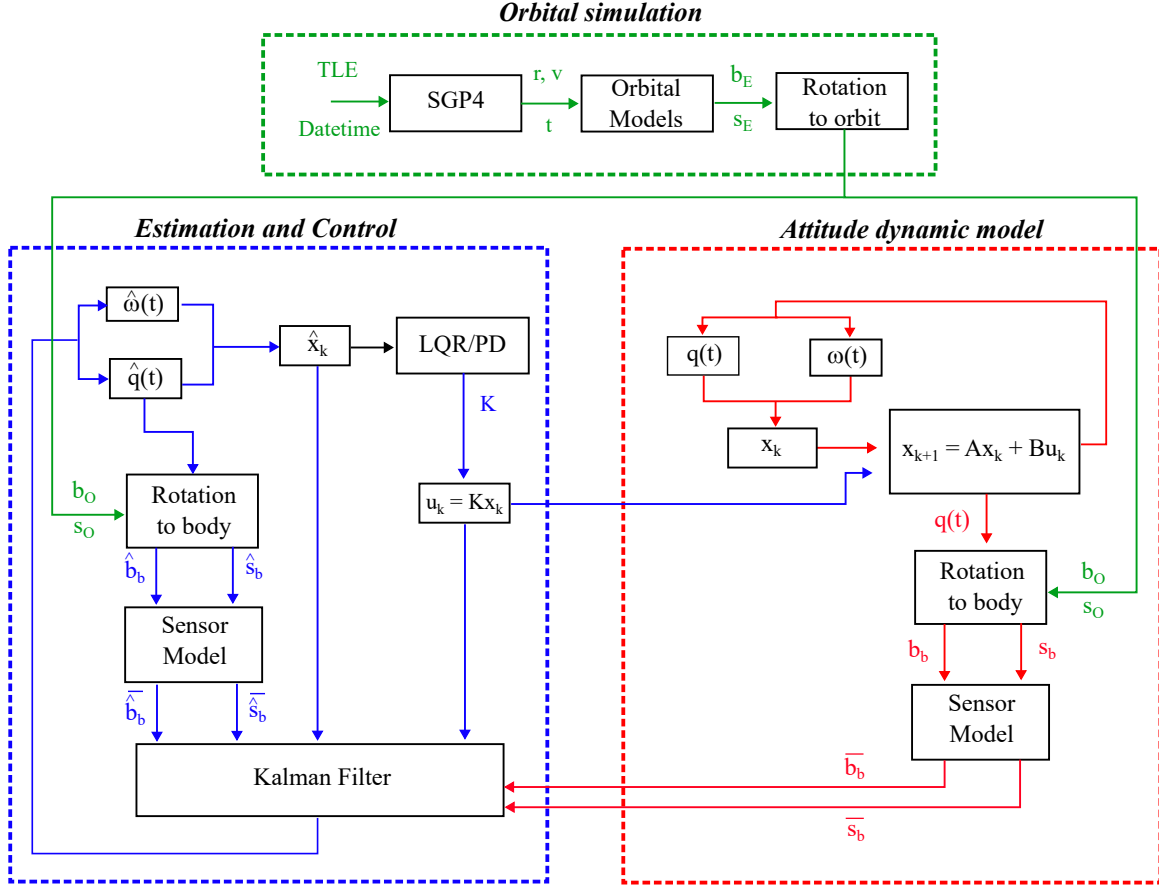


Figura 4.6 Estructura general de la suite de simulación de actitud.

$\mathbf{b}_e$ representa el vector del campo magnético en el sistema ECI, mientras que $\bar{\mathbf{b}}_b$ representa la medición ruidosa del campo magnético en el sistema del cuerpo. Esta notación resulta esencial para distinguir entre los marcos de referencia involucrados en el proceso de estimación y control.

Mediante la iteración de este ciclo en cada paso temporal, la suite permite reproducir el comportamiento acoplado entre la dinámica del CubeSat, su respuesta de control, y el desempeño de su sistema de estimación a bordo.

5 Verificación de la suite de simulación

En este capítulo se presentan resultados de prueba para validar el funcionamiento de la suite de simulación del CubeSat. Con este fin, se simularon distintos escenarios de hardware (configuraciones de nivel bajo, medio y alto), definidos de manera representativa en base a parámetros realistas de un CubeSat operativo. Estos escenarios permiten observar cómo varían los MoP de apuntamiento y los costos asociados cuando se modifican las prestaciones de sensores y actuadores, asegurando que la suite responde de manera consistente ante cambios en las entradas.

Cabe destacar que, en esta etapa, las configuraciones de hardware se emplean únicamente como casos de verificación de la herramienta. Una selección más rigurosa y sistemática de configuraciones de referencia se presenta posteriormente en el Capítulo 6, mediante el uso de una matriz de decisión multicriterio (MCDM), que servirá como línea base para el análisis de optimización.

5.1 Resultados de estimación y control

5.1.1 Validación de la estimación con el SEKF

Con el objetivo de validar cuantitativamente el desempeño del Filtro de Kalman Semiextendido (SEKF), se compararon sus estimaciones con la dinámica no lineal del satélite obtenida mediante integración numérica (RK4). Para ello, se aplicó una acción de control constante de $u = [0.15, 0.15, 0.15]$ durante 100 s, permitiendo evaluar la capacidad del filtro para reconstruir la actitud y las velocidades angulares a partir de mediciones ruidosas.

La Figura 5.1 muestra la comparación entre la simulación no lineal y las estimaciones del SEKF. Se observa que el filtro es capaz de seguir adecuadamente la evolución de los cuaterniones y atenuar el ruido en las velocidades angulares, logrando una concordancia satisfactoria con la dinámica real.

Para cuantificar esta similitud, se empleó la métrica del Error Cuadrático Medio (MSE), definida como:

$$\text{MSE} = \frac{1}{N} \sum_{k=1}^N (x_k^{\text{SEKF}} - x_k^{\text{NL}})^2 \quad (31)$$

donde x_k^{SEKF} corresponde al estado estimado, x_k^{NL} al estado real obtenido del modelo no lineal, y N al número total de muestras.

Los resultados se resumen en la Tabla 5.1, donde se aprecia que los errores son bajos en todas las variables, validando la correcta implementación del estimador.

Se destaca que las componentes q_1 y q_2 presentan los mayores valores de error relativo, lo cual puede asociarse a la sensibilidad de estas variables frente a perturbaciones iniciales. Aun así, los errores globales se mantienen dentro del umbral aceptable establecido (5%), confirmando que el SEKF permite estimar con alta precisión la actitud y las velocidades angulares del satélite.

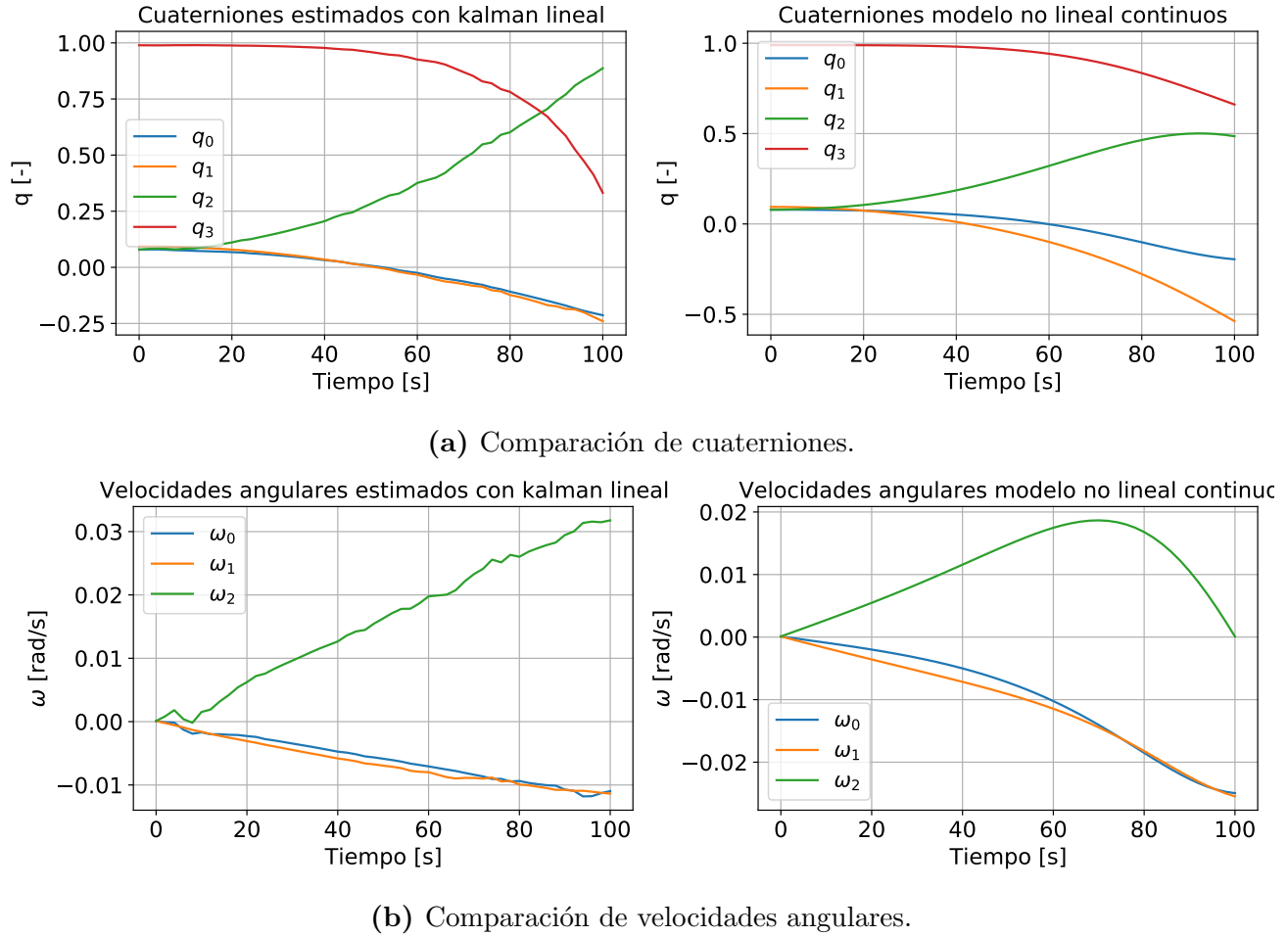


Figura 5.1 Comparación entre filtro de kalman lineal y modelo dinámico no lineal.

Tabla 5.1 Error cuadrático medio (MSE) entre modelo no lineal y filtro SEKF

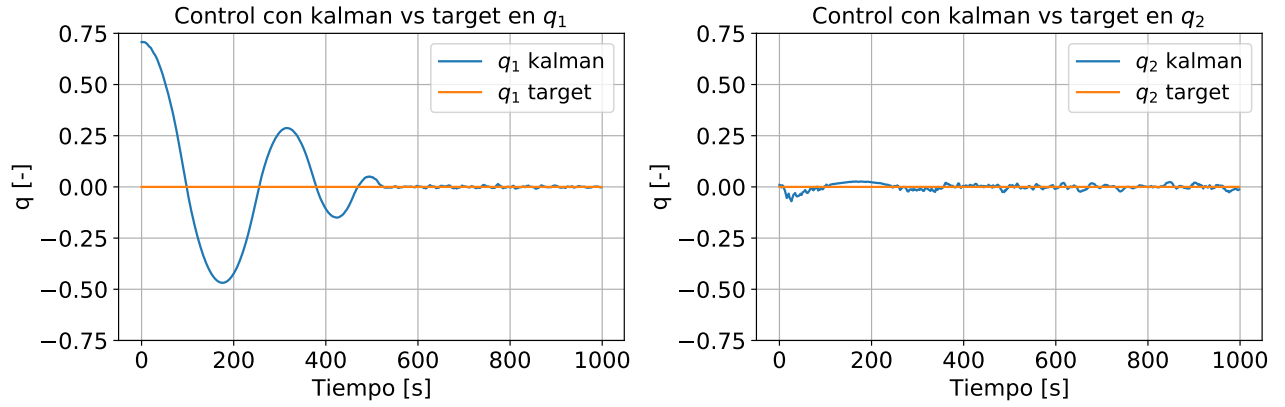
Variable	MSE
$q_0[-]$	2.22×10^{-4}
$q_1[-]$	1.42×10^{-2}
$q_2[-]$	1.73×10^{-2}
$q_3[-]$	6.80×10^{-3}
$\omega_0 [rad/s]$	3.71×10^{-5}
$\omega_1 [rad/s]$	3.63×10^{-5}
$\omega_2 [rad/s]$	1.00×10^{-4}

5.1.2 Comparación entre referencia y estimación en control

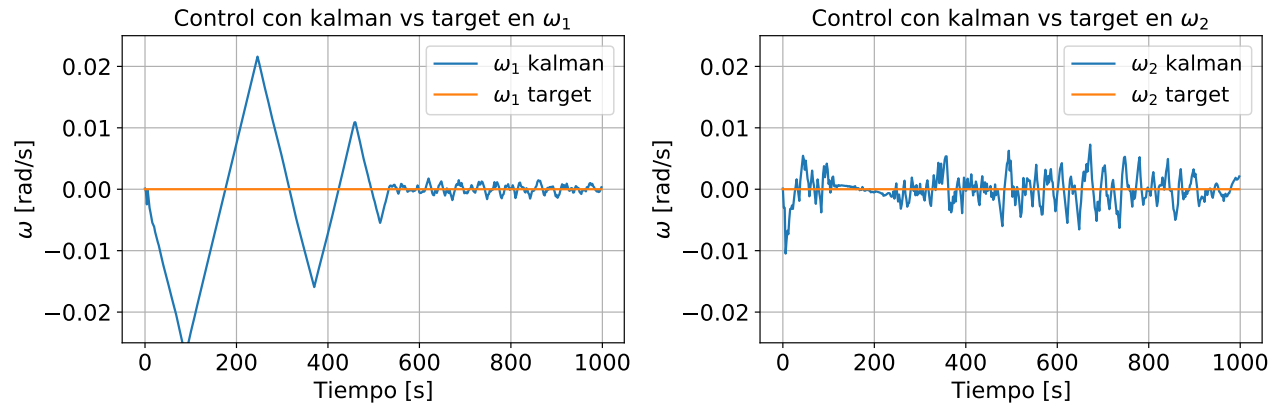
Una vez validado el desempeño del estimador, se evaluó su influencia en el esquema de control de actitud. Para ello, se comparó la referencia (vector objetivo de orientación) con la señal estimada por el SEKF, utilizada como retroalimentación en el controlador LQR.

En la Figura 5.2 se presenta la diferencia entre la referencia y la estimación en algunos componentes de cuaternión y velocidad angular a lo largo del tiempo. Como se esperaba, al inicio

de la simulación existe un desfase debido a las condiciones iniciales impuestas, pero el sistema logra converger rápidamente pese a la condición inicial lejana, mostrando que el filtro entrega una estimación suficientemente precisa para no comprometer el desempeño del controlador.



(a) Target vs estimado en q_1 y q_2 .



(b) Target vs estimado en ω_1 y ω_2 .

Figura 5.2 Diferencia a través del tiempo del control LQR utilizando kalman respecto a objetivo de control.

Estos resultados evidencian que el SEKF cumple un doble rol: por un lado, provee una estimación robusta y precisa de los estados no directamente medibles; por otro, garantiza que el esquema de control basado en LQR opere de manera estable y eficiente al disponer de información confiable para la retroalimentación.

5.2 Cuantificación de los MoP de Apuntamiento

Para asegurar un desempeño adecuado en misiones de observación terrestre, es necesario cuantificar los MoP asociados al apuntamiento del CubeSat. Cada uno de estos índices caracteriza aspectos fundamentales del desempeño del sistema de determinación y control de actitud (ADCS), influenciado por sensores, actuadores y algoritmos de control. En esta sección se detallan los principales MoP empleados en la suite de simulación, junto con su formulación cuantitativa.

Exactitud de apuntamiento [5, 7]

La *exactitud de apuntamiento* es uno de los indicadores más relevantes para misiones de observación terrestre, pues representa la capacidad del CubeSat de mantener su carga útil orientada hacia una región específica de la superficie terrestre. Se define como el error absoluto de apuntamiento, cuantificado en este trabajo mediante el valor medio de la norma del vector de ángulos de Euler $\boldsymbol{\eta} = [\phi, \theta, \psi]^\top$, derivado de la representación de actitud en cuaterniones. Asumiendo que la orientación deseada se encuentra alineada con el marco de referencia (es decir, $\boldsymbol{\eta} = \mathbf{0}$), la exactitud de apuntamiento se expresa como:

$$\Theta = \frac{1}{N} \sum_{i=1}^N \|\boldsymbol{\eta}^{(i)}\|_2, \quad (32)$$

donde N es el número de muestras de actitud consideradas dentro de la ventana de tiempo definida por el tiempo de asentamiento. Este valor se expresa en grados ($^\circ$) o radianes (rad). Una menor Θ implica un mejor desempeño del sistema de control de actitud.

Jitter [5, 9, 10]

El *jitter* en la línea de visión (LoS) se refiere a las oscilaciones de alta frecuencia que afectan la orientación del satélite, generando imágenes borrosas o inestabilidad en la adquisición de datos. Estas fluctuaciones provienen de interacciones dinámicas entre la estructura del satélite y componentes físicos (como magnetorquers), y se manifiestan en frecuencias superiores al ancho de banda del sistema de control.

Para cuantificar el jitter, se analiza la *densidad espectral de potencia* (PSD) del error angular $\Theta(t)$ (la norma temporal del vector de ángulos de Euler), después de aplicar un filtro pasa alto con corte a 10 Hz. La PSD se estima utilizando el método de Welch:

$$\hat{S}_\Theta(f) = \frac{1}{K} \sum_{k=0}^{K-1} \frac{|X_k(f)|^2}{U}, \quad (33)$$

donde $X_k(f)$ es la Transformada Discreta de Fourier (DFT) del segmento k de la señal $\Theta(t)$, y U es el factor de normalización de potencia de la ventana utilizada. Un mayor valor de $\hat{S}_\Theta(f)$ en las altas frecuencias indica mayor jitter y, por ende, menor calidad en la estabilidad de apuntamiento.

Agilidad [5, 11]

La *agilidad* corresponde a la capacidad del CubeSat de realizar maniobras rápidas de apuntamiento, permitiéndole reorientarse eficientemente hacia objetivos dinámicos o múltiples dentro de una misma órbita. Este parámetro se cuantifica mediante el *tiempo de asentamiento* (T_s), que define el tiempo requerido para que la respuesta de actitud permanezca dentro de una banda de tolerancia establecida posterior a una maniobra o perturbación.

Para cada ángulo de Euler $w(t) \in \{\phi(t), \theta(t), \psi(t)\}$, el tiempo de asentamiento individual se define como:

$$T_{s,w} = \min \{t \mid |w(\tau) - w_{\text{final}}| \leq \varepsilon, \forall \tau \geq t\}, \quad (34)$$

donde ε es la banda de tolerancia y w_{final} es el valor de referencia deseado. El tiempo de asentamiento global se obtiene como:

$$T_s = \max \{T_{s,\phi}, T_{s,\theta}, T_{s,\psi}\}. \quad (35)$$

Para permitir comparaciones normalizadas entre distintos sistemas ADCS, se define un índice adimensional de agilidad T_s^* como:

$$T_s^* = \frac{T_s}{\omega_n}, \quad (36)$$

donde ω_n es la frecuencia natural del sistema, obtenida en los análisis de las funciones de transferencia utilizando las matrices espacio estado A de los actuadores. Un menor valor de T_s^* implica mayor agilidad. Este índice es particularmente relevante para misiones que requieren cobertura de múltiples objetivos o respuesta rápida a eventos dinámicos.

Drift [5]

El *drift* o deriva se refiere a la desviación angular gradual del satélite con el tiempo, y es un parámetro clave en aplicaciones que requieren mantener una orientación constante durante largos periodos. Se expresa en grados por hora [$^\circ/\text{hr}$]. Aunque su impacto es considerable, en el presente trabajo solo se menciona como referencia y no se incluye en la evaluación cuantitativa.

5.3 Condiciones y parámetros de simulación

Para obtener resultados de rendimiento, se utilizaron los datos de TLE del SUCHAI-3 con fecha de inicio del 01 de noviembre del 2023. Se debe tener en cuenta que estas condiciones, en conjunto con otros parámetros son dados para todos los niveles de sensores y actuadores impuestos dentro de la suite de simulación, y sirven para tener un parámetro comparativo entre los componentes físicos del ADCS.

En la Tabla 5.2 se muestran los valores de los parámetros recién mencionados utilizados en la simulación, que son basados en literatura [14, 69, 70] o por elaboración propia. Cabe mencionar que q_i y ω_i representan la orientación y velocidad angular inicial real.

Además, con el objetivo de visualizar en el simulador el rendimiento de apuntamiento y su costo asociado, se decide agrupar los componentes físico en niveles según el Commercial off the shelf (COTS) disponible en CubeSat [71]. Esto se hace debido a que existen una variedad considerable de componentes capaces de ser usado en este tipo de nanosatélites, al ser de poco costo tanto energético como monetario. Caracterizar todos estos se escapa de los objetivos del trabajo, ya que su consideración genera un análisis extenso e innecesario respecto a la obtención de los costos en base a los SE envelopes. Por ello, se presenta en la Tabla 5.3 los componentes COTS seleccionados en conjunto con su proveedor para analizar tanto el rendimiento (MoP de

Tabla 5.2 Parámetros del sistema

Parámetro	Valor
$[I_x, I_y, I_z] [kg \cdot m^2]$	[0.037, 0.036, 0.006]
$[I_{s0}, I_{s1}, I_{s2}] [kg \cdot m^2]$	[0.005, 0.005, 0.004]
$[b_0, b_1, b_2] [m]$	[0.05, 0.05, 0.015]
$\omega_{0,0} [rad/s]$	0.00163
$T_{prop} [s]$	5760 [s] (1 órbita)
LVLH to Body $q_i [-]$	[0.1771, 0.3063, 0.1771, 0.9202]
LVLH to Body Euler angles $(\phi, \theta, \psi) [^\circ]$	[30, 30, 30]
$\omega_i [rad/s]$	[0.0001, 0.0001, 0.0001]
$P_{i,MT}$	diag(0.25, 0.25, 0.25, 0.01, 0.01, 0.01)

apuntamiento) como el costo utilizado por cada uno de ellos. Mas detalles de cada uno de los sensores y actuadores elegidos en el Anexo F.

Tabla 5.3 Componentes clasificados por nivel de rendimiento

Componente	Nivel bajo	Nivel medio	Nivel alto
Giroscopio	CRH03 - 200 (Silicon Sensing Systems)	CRH03 - 010 (Silicon Sensing Systems)	NSGY-001 (NewSpace Systems)
Magnetómetro	MM200 (AAC Clyde Space)	MAG-3-2 (AAC Clyde Space)	MAG-3 (AAC Clyde Space)
Sun Sensor	CSS-01, CSS-02 (Space Micro)	MSS-01 (Space Micro)	FSS (Bradford Space)
Magnetorquer	MT0.5-1 (ZARM Technik)	NCTR-M012 (NewSpace Systems)	MT15-1 (ZARM Technik)
Rueda de reacción	RWP500 (Blue Canyon Technologies)	RW1 (Blue Canyon Technologies)	RW4 (Blue Canyon Technologies)

Es relevante mencionar que en el Anexo G se muestra un ejemplo a detalle sobre la obtención de los MoP de apuntamiento, siendo estos los pasos a seguir para los casos de estudio que se mencionarán en las siguientes secciones.

5.4 Resultados suite de simulación

En esta sección se presentarán análisis respecto a los resultados obtenidos utilizando parámetros de la sección anterior. Se compararan los diferentes niveles de sensores y actuadores, los tipos de actuadores y los controladores utilizados para conocer la performance y costo de cada caso.

5.4.1 Resultados tipos de actuadores

A continuación, se presenta un análisis sobre el uso de los magnetorquers y las ruedas de reacción, destacando sus diferencias en cuanto a la performance de apuntamiento y el costo asociado.

En la Figura 5.3, se muestran la norma de los ángulos de Euler a lo largo del tiempo utilizando sensores y actuadores de nivel medio (nivel 2). A la izquierda se observa el control ejercido por las ruedas de reacción, mientras que a la derecha, el control realizado por los magnetorquers. La Tabla 5.4 resume los resultados de la norma de los MoP de apuntamiento, junto con la diferencia en el consumo de potencia y la masa entre ambos tipos de actuadores.

La gráfica revela que las ruedas de reacción son capaces de generar torques de mayor magnitud, lo que permite alcanzar más rápidamente la orientación deseada en comparación con los magnetorquers. En ambos casos, se observa un comportamiento cíclico del ruido una vez que la orientación se estabiliza cerca de cero, lo cual se debe a la repetición de vectores de posición similares en cada órbita. Esto provoca que el campo magnético terrestre ejerza una fuerza de magnitud constante sobre el CubeSat, la cual es medida por el magnetómetro.

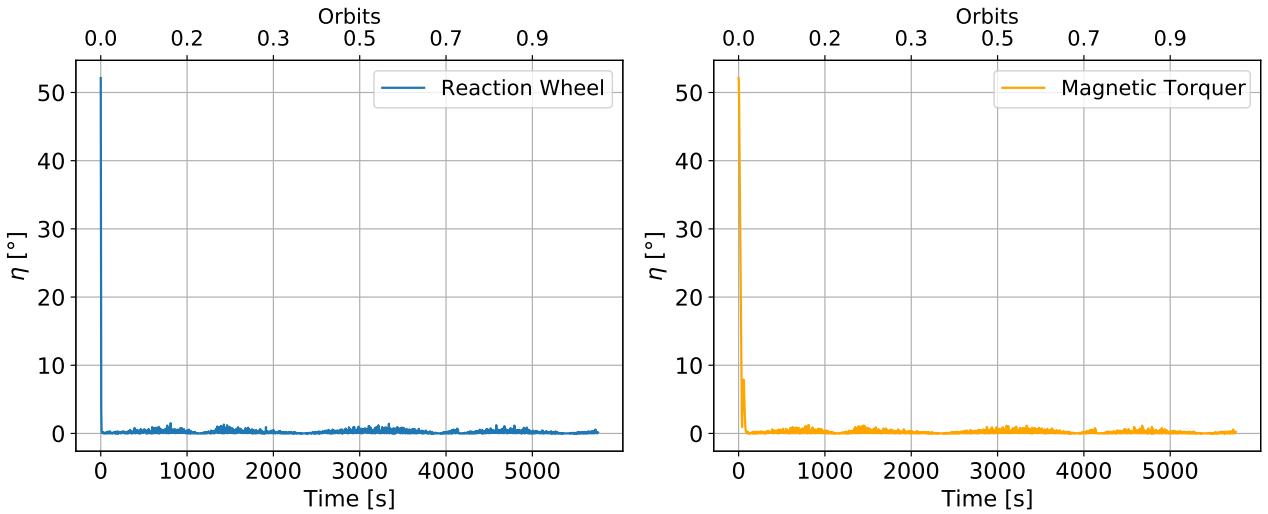


Figura 5.3 Ángulos de Euler LVLH y body para dos actuadores distintos (Elaboración propia).

Además, en la Tabla 5.4 se observa de forma cuantitativa que las ruedas de reacción ofrecen mejores parámetros de rendimiento en términos de la agilidad y jitter, a cambio de un mayor consumo energético y una mayor masa en comparación con el uso de un magnetorquer, el cual presenta mejor métrica de exactitud de apuntamiento. Por lo tanto, según los requisitos de la misión, es fundamental que cada usuario evalúe qué opción resulta más adecuada en función de sus necesidades específicas. Cabe destacar que la métrica de esfuerzo de control $\sum u^T R u$ confirma el carácter “continuo pero débil” del magnetorquer: al requerir torque durante mucho más tiempo, su valor es varios órdenes de magnitud mayor que el de la rueda de reacción, la cual aplica impulsos breves y por ello acumula un costo energético muy bajo.

5.4.2 Resultados de los controladores

A continuación, se presenta un análisis comparativo entre el controlador PD y el controlador LQR aplicado al magnetorquer, con el objetivo de determinar cuál de estas opciones es más adecuada para su uso en la optimización.

En la Figura 5.4, se muestran los resultados de los ángulos de Euler llevados al equilibrio,

Tabla 5.4 Rendimiento y costo para rueda de reaccion y magnetorquer en mismas condiciones.

Tipo de actuador	Potencia promedio [W]	Masa [kg]	Θ [°]	$\hat{S}_\Theta(f)$ [W/Hz]	T_s [s]	$\sum u^T Ru$
Rueda de reacción	9	0.95	0.20	0.056	11.7	7.37×10^{-5}
Magnetorquer	0.8	0.053	0.18	0.061	74	211.9

utilizando sensores y actuadores de nivel medio (nivel 2). A la izquierda, se observa el control ejercido por el controlador PD, mientras que a la derecha, se presenta el control realizado por el controlador LQR. La Tabla 5.5 resume los resultados de la norma de los MoP de apuntamiento.

La gráfica muestra que el controlador PD tarda más en estabilizar el sistema en el punto de equilibrio, utilizando los mismos componentes físicos (sensor de nivel medio y magnetorquer). Sin embargo, al estabilizarse más lentamente, se espera una mayor precisión en el apuntamiento, ya que el sistema permanece más tiempo alejado del punto de equilibrio dentro de la banda de asentamiento.

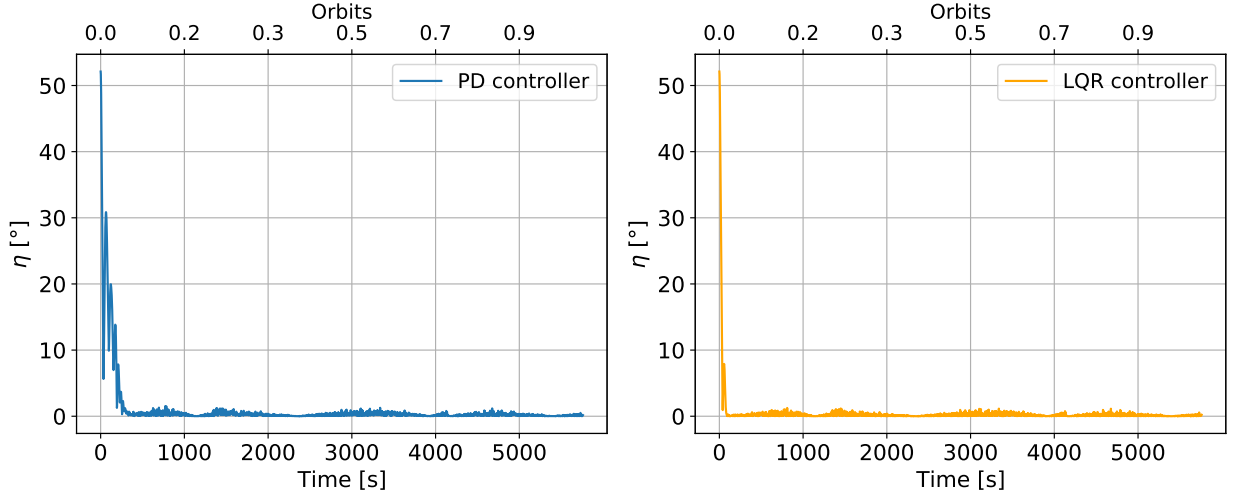


Figura 5.4 Ángulos de Euler LVLH y body para dos controladores distintos (Elaboración propia).

Además, en la Tabla 5.5 se confirma que, como se mencionó anteriormente, ambos controladores utilizan los mismos componentes, lo que implica el mismo consumo energético según la potencia máxima y la masa de los magnetorquers de nivel medio. Sin embargo, en términos de rendimiento, el controlador LQR ofrece un mejor tiempo de asentamiento y mayor precisión en el apuntamiento. Por esta razón, tanto para las ruedas de reacción como para los magnetorquers, se selecciona el controlador LQR para su implementación en la optimización dentro de la suite de simulación.

El valor de $\sum u^T Ru$ muestra que el LQR logra la misma tarea con cerca de un 80 % menos de energía de control que el PD, reflejando la menor sobre-corrección y la política óptima inherente

Tabla 5.5 Rendimiento y costo para controlador PD y LQR en mismas condiciones.

Controlador	Potencia promedio [W]	Masa [kg]	Θ [°]	$\hat{S}_\Theta(f)$ [W/Hz]	T_s [s]	$\sum u^T R u$
PD	0.8	0.053	0.23	0.081	248	1087
LQR	0.8	0.053	0.18	0.061	74	211.9

al diseño LQR.

5.4.3 Resultados niveles de sensores

A continuación, se presenta un análisis sobre los tres niveles de sensores, utilizando actuadores de nivel 2 (medio), con el fin de evaluar cómo la estimación de la actitud del satélite, basada en los diferentes niveles de sensores, afecta los MoP de apuntamiento y su costo asociado.

En la Figura 5.5, se muestran tres columnas de gráficas que representan el control de los ángulos de Euler hacia el equilibrio. En la primera columna, a la izquierda, se observa la simulación de actitud basada en los ángulos de Roll, Pitch y Yaw utilizando sensores de nivel 1 (bajo). La segunda columna muestra el caso con sensores de nivel 2 (medio), mientras que la tercera columna, a la derecha, corresponde a sensores de nivel 3 (alto). La Tabla 5.6 resume los resultados de los MoP de apuntamiento junto con los costos asociados a los distintos niveles de magnetómetro y sensor solar.

En la gráfica, se puede observar que, con sensores de nivel bajo, existe mayor ruido y dispersión en los ángulos de Euler una vez estabilizados en el equilibrio. Este comportamiento mejora conforme se incrementa el nivel de los sensores, mostrando una menor dispersión en los ángulos de Euler, lo que resulta en una mejora de la precisión de apuntamiento y una reducción del jitter.

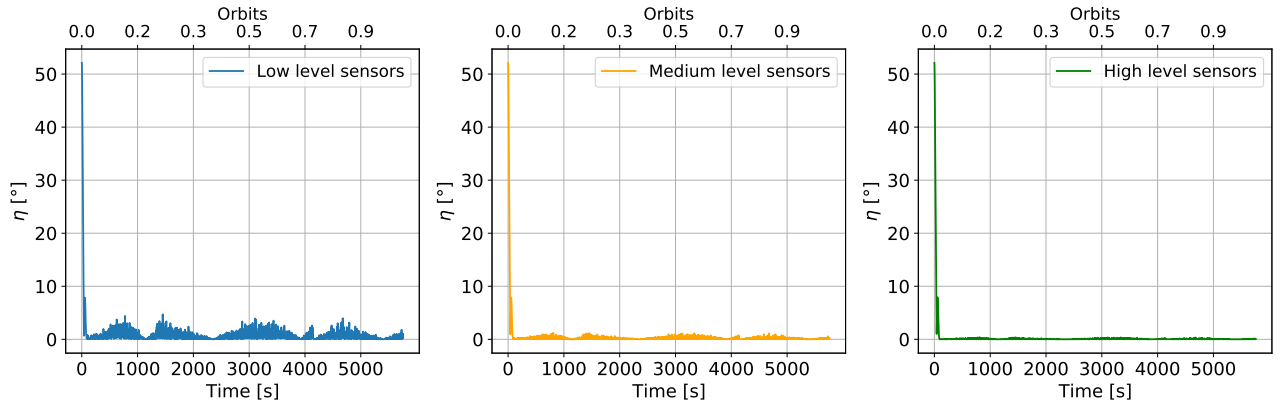


Figura 5.5 Ángulos de Euler LVLH y body para niveles de sensor distintos con magnetorquer (Elaboración propia).

Además, en la Tabla 5.6 se confirma que la mejora en el nivel de los sensores contribuye significa-

tivamente a un mejor rendimiento en la precisión de apuntamiento y en la reducción del jitter. Sin embargo, una observación importante es que, aunque el paso de sensores de nivel medio a nivel alto mejoró la exactitud de apuntamiento, no tuvo un impacto notable en la reducción del jitter. En cuanto a la agilidad, el efecto de los sensores en la mejora del tiempo de asentamiento fue mínimo, lo que indica que este componente físico no está directamente relacionado con dicho MoP de apuntamiento. A medida que mejora la calidad de los sensores, $\sum u^T Ru$ descende drásticamente: se necesitan menos maniobras correctivas porque la estimación de la actitud es más precisa y el controlador aplica torques más pequeños.

Tabla 5.6 Rendimiento y costo para niveles de sensor con nivel 2 de magnetorquer y LQR.

Nivel de sensor	Potencia promedio [W]	Masa [kg]	Θ [°]	$\hat{S}_\Theta(f)$ [W/Hz]	T_s [s]	$\sum u^T Ru$
Nivel 1 [$\sigma_s = 0.833^\circ$, $\sigma_b = 1.18nT$]	0.15	0.067	0.62	0.56	72	1023
Nivel 2 [$\sigma_s = 0.167^\circ$, $\sigma_b = 0.1nT$]	0.3	0.181	0.18	0.061	74	211.9
Nivel 3 [$\sigma_s = 0.05^\circ$, $\sigma_b = 0.012nT$]	0.75	0.530	0.056	0.022	74	104.3

En la Figura 5.6 se presenta el mismo análisis de sensores, pero en este caso utilizando las ruedas de reacción como actuador de nivel 2. También se muestran los resultados de los MoP de apuntamiento y el costo asociado a los sensores en la Tabla 5.7.

En la gráfica, al igual que con los magnetorquers, se puede observar un mayor ruido y dispersión en los datos de los ángulos de Euler cuando se utilizan sensores de nivel bajo. Esta dispersión disminuye a medida que se incrementa la calidad de los magnetómetros y los sensores solares, evidenciándose una menor variabilidad en los datos una vez que se alcanza el equilibrio.

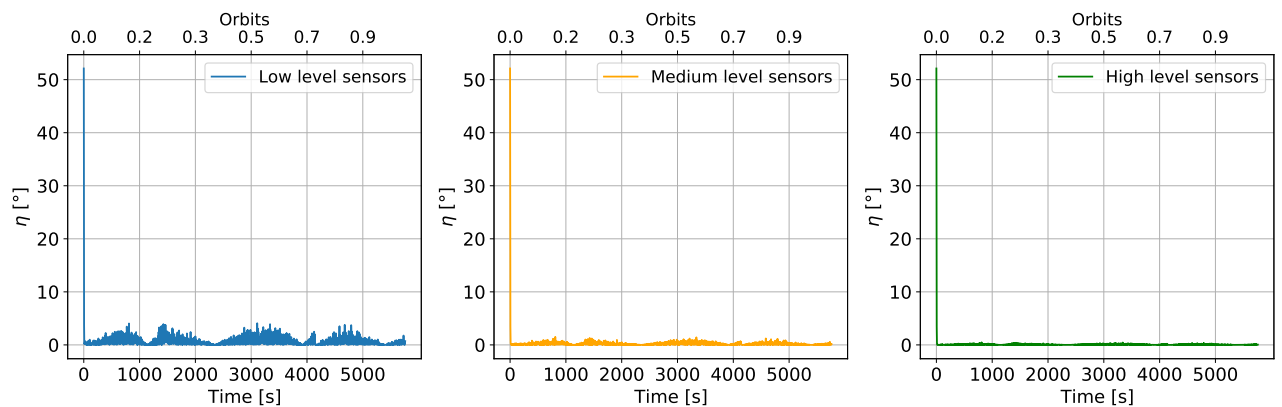


Figura 5.6 Ángulos de Euler LVLH y body para niveles de sensor distintos con rueda de reacción (Elaboración propia).

En la Tabla 5.7, se confirma lo mencionado anteriormente: se observa una mejora notable en la exactitud de apuntamiento y en el jitter al pasar de sensores de nivel bajo a medio, aunque

esto conlleva un incremento significativo en el consumo de potencia y energía debido al uso de sensores de mayor calidad.

Un aspecto particular del caso de las ruedas de reacción es que, a diferencia de lo observado con los magnetorquers, en cada nivel se manifiesta una mejora en el jitter. En el caso de los magnetorquers, no se notó esta mejora al pasar del nivel 2 al nivel 3, donde los valores se mantuvieron constantes. Además, no se observa una mejora en la agilidad al incrementar el nivel de los sensores, como se evidencia en la tabla, donde los tiempos de asentamiento permanecen sin variaciones a pesar de la mejora en el componente físico. En las ruedas de reacción se observa que mejores sensores reducen $\sum u^T Ru$, aunque el descenso es menos pronunciado.

Tabla 5.7 Rendimiento y costo para niveles de sensor con nivel 2 de rueda de reacción y LQR.

Nivel de sensor	Potencia promedio [W]	Masa [kg]	Θ [°]	$\hat{S}_\Theta(f)$ [W/Hz]	T_s [s]	$\sum u^T Ru$
Nivel 1 [$\sigma_s = 0.833^\circ$, $\sigma_b = 1.18nT$]	0.15	0.067	0.62	0.55	780	1.45×10^{-4}
Nivel 2 [$\sigma_s = 0.167^\circ$, $\sigma_b = 0.1nT$]	0.3	0.181	0.20	0.056	8	7.37×10^{-5}
Nivel 3 [$\sigma_s = 0.05^\circ$, $\sigma_b = 0.012nT$]	0.75	0.530	0.061	0.0060	8	6.29×10^{-5}

5.4.4 Resultados niveles de actuadores

A continuación, se presentan análisis sobre los tres niveles de actuadores, utilizando sensores de nivel 2 (medio), para evaluar cómo la estimación de la actitud del satélite, en función de los diferentes niveles de actuadores, afecta los MoP de apuntamiento y su costo asociado.

En la Figura 5.7, se muestran tres columnas de gráficas que representan el control de los ángulos de Euler hacia el equilibrio. En la primera columna, a la izquierda, se presenta la simulación de actitud basada en los ángulos de Roll, Pitch y Yaw, utilizando magnetorquers de nivel 1. Las segunda y tercera columnas muestran el mismo análisis para los niveles 2 y 3, respectivamente. La Tabla 5.8 resume los resultados de los MoP de apuntamiento y el costo asociado a los magnetorquers.

En la gráfica, se puede observar que, con actuadores de nivel bajo, el control es más suave, ya que el sistema tarda más tiempo en asentarse en el equilibrio. Sin embargo, esta situación mejora al utilizar magnetorquers de mayor calidad, como se evidencia en las otras dos columnas. Este comportamiento de mejora en el control hacia el equilibrio se aprecia con mayor claridad en la gráfica correspondiente al Yaw (verde).

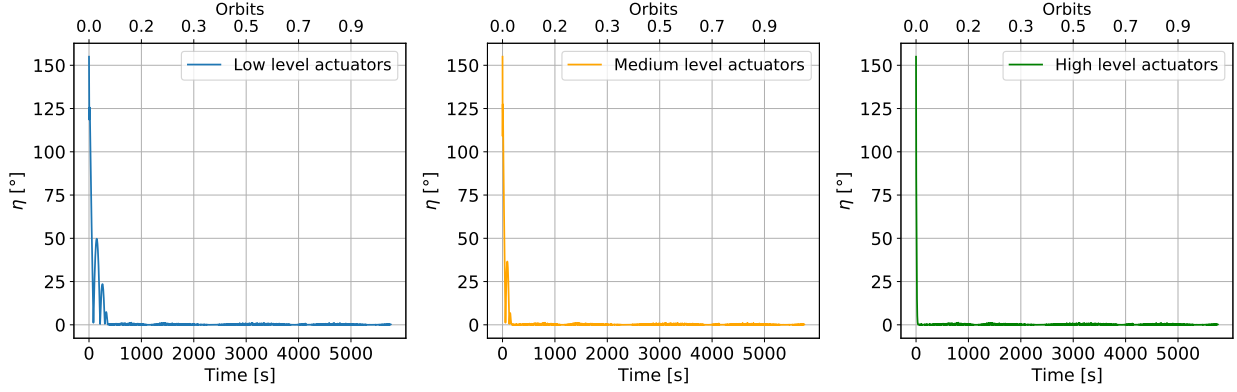


Figura 5.7 Ángulos de Euler LVLH y body para niveles de magnetorquer distintos con sensores nivel 2 (Elaboración propia).

En la Tabla 5.8, se observa que, a medida que se incrementa el nivel de los magnetorquers, la norma de la exactitud de apuntamiento mejora del nivel 1 al nivel 2, pero se deteriora al alcanzar el nivel 3. También se nota una disminución en el tiempo de asentamiento, un aspecto que se representa de manera más clara en la gráfica.

Sin embargo, al aumentar el nivel del magnetorquer, se evidencia un incremento en el ruido, reflejado por el jitter. Este fenómeno se debe a que un mayor torque debido al campo magnético, genera vibraciones mecánicas que afectan la densidad espectral de potencia. A partir de los resultados, se puede concluir que existe una tendencia clara que indica que los magnetorquers están directamente relacionados con la mejora del tiempo de asentamiento. Sin embargo, es importante destacar que, dependiendo del caso, este aumento en el nivel del actuador puede no necesariamente traducirse en una mejora de la exactitud de apuntamiento, lo que sugiere que no hay una relación directa entre ambas variables en todas las situaciones. Para los magnetorquers, subir de nivel reduce el tiempo de asentamiento pero, a partir del nivel 2, eleva fuertemente $\sum u^T Ru$; los torques mayores generan vibraciones y el controlador compensa con más acción, encareciendo la energía de control.

Tabla 5.8 Rendimiento y costo para niveles de magnetorquer con nivel 2 de sensor y LQR.

Nivel de actuador	Potencia promedio [W]	Masa [kg]	Θ [°]	$\hat{S}_{\Theta}(f)$ [W/Hz]	T_s [s]	$\sum u^T Ru$
Nivel 1 [$\tau = 0.5Am^2$]	0.275	0.03	0.18	0.077	166	150.6
Nivel 2 [$\tau = 1.19Am^2$]	0.8	0.053	0.18	0.061	74	211.9
Nivel 3 [$\tau = 15Am^2$]	1.11	0.43	0.17	0.05	22	923.4

En la Figura 5.8, se presenta el mismo análisis aplicado a las ruedas de reacción, utilizando nuevamente sensores de nivel 2. Además, los resultados de los MoP de apuntamiento y el costo asociado a los actuadores se detallan en la Tabla 5.9.

En la gráfica, se observa un comportamiento similar al observado en el caso de los magnetorquers. A medida que se mejora el nivel del actuador, se evidencia una mejora en el control hacia el tiempo de asentamiento en los ángulos de Euler. Las gráficas muestran leves diferencias en la dispersión de datos y el ruido, lo que sugiere que, a simple vista, estos factores no afectan significativamente los demás MoP de apuntamiento analizados.

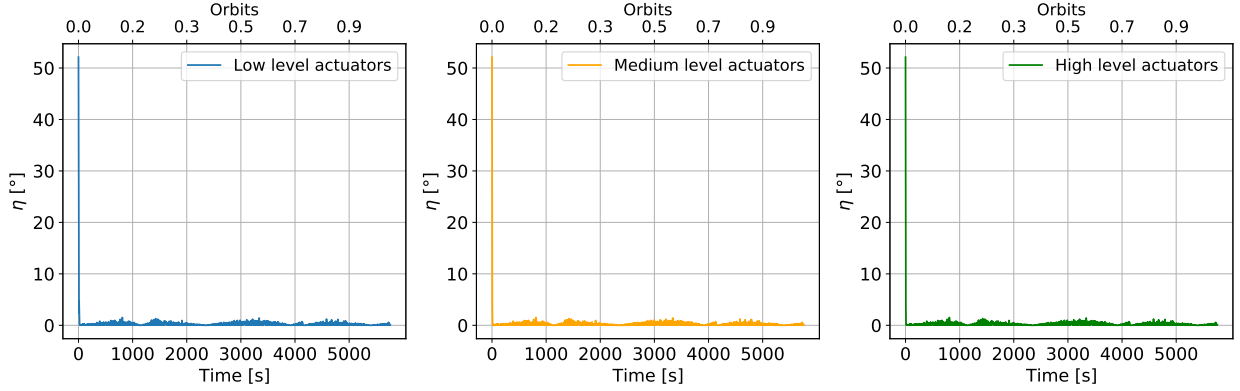


Figura 5.8 Ángulos de Euler LVLH y body para niveles de rueda de reacción distintos con sensores nivel 2 (Elaboración propia).

En la Tabla 5.9, se puede observar que una mejora en el nivel de las ruedas de reacción implica un costo significativamente mayor, especialmente en comparación con los magnetorquers. Este aumento de costo se traduce en una mejora en el tiempo de asentamiento, así como en una mejora del jitter y, en menor medida, en la exactitud de apuntamiento. Sin embargo, es importante señalar que, a pesar de la mejora en el nivel de las ruedas de reacción, el impacto en la exactitud de apuntamiento no es significativo, lo cual se refleja también en las gráficas. En ruedas de reacción, el esfuerzo de control crece del nivel 1 al 2, pero se estanca después; la autoridad extra no se traduce en ahorro adicional de energía, lo que sugiere un punto de rendimientos decrecientes más allá del nivel 2.

Tabla 5.9 Rendimiento y costo para niveles de rueda de reacción con nivel 2 de sensor y LQR.

Nivel de actuador	Potencia promedio [W]	Masa [kg]	Θ [°]	$\hat{S}_{\Theta}(f)$ [W/Hz]	T_s [s]	$\sum u^T R u$
Nivel 1 [$\tau = 1 \times 10^{-3} Nm$]	6	0.75	0.20	0.057	16	2.36×10^{-5}
Nivel 2 [$\tau = 0.025 Nm$]	9	0.95	0.20	0.056	8	7.37×10^{-5}
Nivel 3 [$\tau = 0.25 Nm$]	10	3.2	0.20	0.056	8	7.37×10^{-5}

5.5 Resumen sobre análisis realizados

Al analizar los resultados obtenidos con la suite de simulación, se pueden establecer las siguientes conclusiones:

- **Rueda de reacción vs Magnetorquer:** A partir de los análisis realizados y el examen detallado de ambos actuadores, se concluye que las ruedas de reacción ofrecen una mejor reducción del tiempo de asentamiento. Por otro lado, se recomienda el uso de magnetorquers para misiones con menos restricciones en el rendimiento o con mayores limitaciones en alguno de los SE envelopes, ya que su costo es inferior. En los demás parámetros presentan valores similares.
- **PD vs LQR:** Este análisis se llevó a cabo para determinar la viabilidad de utilizar un controlador PD en comparación con el LQR. Los resultados mostraron que el LQR siempre superó al PD en términos de exactitud de apuntamiento y tiempo de asentamiento. Además, se observó que, para un mismo actuador, el controlador LQR es más eficiente que el PD, logrando estabilizar el sistema con menor esfuerzo acumulado, por lo que se optó por la segunda opción.
- **Análisis del nivel de sensores:** Se buscó entender cómo el tipo de sensores impacta en los parámetros de rendimiento para asignar un peso específico a las variables de optimización. Se observó que los sensores mejoran significativamente la exactitud de apuntamiento y reducen el jitter, aunque no hay una relación clara con la agilidad, que mostró poca variación. La mejora en la calidad de medición permitió una reducción significativa del esfuerzo de control, al disminuir la cantidad de correcciones necesarias. Esta tendencia es consistente tanto para las ruedas de reacción como para los magnetorquers.
- **Análisis del nivel de actuadores:** Similar al análisis de sensores, se buscó establecer una relación entre el nivel de los actuadores y los parámetros de rendimiento. Se encontró que ambos actuadores ofrecían mejores valores de agilidad, con un tiempo de asentamiento reducido a medida que mejoraba el actuador. No se observó un cambio relevante entre la exactitud de apuntamiento y los niveles evaluados. En ambos actuadores se observa una menor cantidad de jitter a medida que mejora la calidad del componente, aunque en magnetorquers es más notorio. El aumento de nivel suele implicar un mayor esfuerzo de control, particularmente en los magnetorquers, donde un mayor torque genera vibraciones que deben ser corregidas por el controlador.

6 Modelo de optimización

Este trabajo aborda el problema de seleccionar un conjunto óptimo de componentes físicos para el ADCS de un CubeSat, sujeto a requerimientos de rendimiento y restricciones impuestas por los SE envelopes relacionados con masa y potencia. Tal como se muestra en la Fig. 1.3, la solución propuesta integra una suite de simulación que evalúa los parámetros de rendimiento en actitud y computa la configuración más adecuada de sensores y actuadores.

Las variables de decisión de la optimización son las especificaciones técnicas de los componentes seleccionados: las desviaciones estándar del sensor solar σ_s y del magnetómetro σ_b , junto con el torque máximo τ que puede entregar el actuador. Estas variables están acotadas dentro de intervalos de factibilidad definidos, tales que: $\sigma_s \in [\sigma_{s,\min}, \sigma_{s,\max}]$, $\sigma_b \in [\sigma_{b,\min}, \sigma_{b,\max}]$, y $\tau \in [\tau_{\min}, \tau_{\max}]$.

El problema de optimización se formula como la minimización de una función objetivo escalar definida como la suma ponderada de nueve funciones de costo:

$$f(x) = \sum_{i=1}^9 W_i f_i(x), \quad (37)$$

donde cada $f_i(x)$ cuantifica el rendimiento de los MoP de apuntamiento o el uso de recursos en base a la potencia y masa de cada componente físico, y W_i representa el peso relativo asignado según las prioridades de la misión. Esta formulación corresponde a un problema de programación no lineal general (NLP), dado que la dependencia entre las funciones de costo y las variables de decisión es no lineal y potencialmente no convexa.

Adicionalmente, se imponen restricciones que aseguran el cumplimiento de los SE envelopes, tales que la masa y el consumo promedio de potencia de la configuración seleccionada deben satisfacer: $Masa(x) \leq m_{\max}$ y $Potencia(x) \leq p_{\max}$. Para cada solución candidata x , la suite de simulación ejecuta modelos dinámicos orbitales y de actitud que calculan los indicadores de rendimiento relevantes: la exactitud de apuntamiento Θ , el cálculo del jitter $\mathcal{S}_\Theta(f)$ y el tiempo de asentamiento máximo T_s^* . Estos valores son posteriormente utilizados para evaluar las funciones de costo. El proceso de optimización retorna la configuración que minimiza la función objetivo respetando todas las restricciones técnicas y operacionales impuestas.

6.1 Funciones de optimización para los parámetros de rendimiento en actitud

Para formular las funciones de costo asociadas a los MoP de apuntamiento, se utiliza la suite de simulación de actitud. Esta se inicializa con condiciones predefinidas y se ejecuta sobre un horizonte de tiempo especificado. La evolución resultante en actitud del CubeSat es luego procesada para convertir la representación en cuaterniones a ángulos de Euler, lo cual permite una interpretación más intuitiva del comportamiento rotacional del satélite.

A partir de estos ángulos de Euler se calculan tres MoPs escalarizadas y definidas previamente: exactitud de apuntamiento, jitter y agilidad. Cada una se evalúa sobre la norma de los tres ejes principales de rotación —Roll, Pitch y Yaw— proporcionando una caracterización completa del desempeño del control de actitud.

Estas tres métricas de rendimiento conforman indicadores clave en el marco de optimización. Las funciones de costo correspondientes se definen como:

$$f_{MOP} = \|\text{MoP}(\sigma_m, \sigma_s, \tau) - \text{MoP}\|^2 \quad (38)$$

Donde $MOP(\sigma_m, \sigma_s, \tau)$ es la métrica de rendimiento obtenida en cada paso de simulación y MoP es el valor de la métrica objetivo.

6.2 Funciones de optimización para recursos físicos

Las funciones de costo restantes están asociadas al uso de recursos físicos de los sensores y actuadores considerados. Para cada componente, se comparan la masa y el consumo de potencia con los valores objetivo definidos por el usuario, tal como se detalla a continuación:

$$f_{SE} = \|\text{SE}(\sigma_m, \sigma_s, \tau) - \text{SE}\|^2 \quad (39)$$

donde $SE(\sigma_m, \sigma_s, \tau)$ representa la potencia o masa promedio obtenida en cada paso de la simulación de optimización, y SE corresponde al costo objetivo.

Para definir estas funciones de costo asociadas a los SE envelopes seleccionado, se empleó una metodología basada en datos existentes que permitió identificar relaciones funcionales entre estas métricas de desempeño y las variables técnicas involucradas en la optimización. El análisis se sustentó en especificaciones técnicas de sensores solares, magnetómetros y magnetorquers disponibles comercialmente, los cuales han sido integrados en misiones previas tipo CubeSat, como se documenta en la literatura y hojas de datos de fabricantes [36, 72].

Para cada categoría de componente se seleccionó un conjunto de 7 a 8 dispositivos, considerando su uso y disponibilidad en el mercado CubeSat. En el caso de los sensores solares, solo se identificaron ocho dispositivos con datos completos y confiables. Aunque el tamaño de muestra puede parecer limitado, cubre todo el rango de diseños de vanguardia actualmente utilizados en plataformas satelitales pequeñas, lo cual entrega una base representativa para la formulación de leyes de escala a nivel de componente.

Para caracterizar las dependencias entre las variables técnicas y las SE envelopes, se realizaron regresiones no lineales. En particular, se ajustó un modelo de ley de potencias de la forma:

$$y = a \cdot x^b, \quad (40)$$

donde y representa el consumo de potencia o la masa estimada, x es la variable técnica correspondiente (como la desviación estándar del sensor solar, el nivel de ruido del magnetómetro, o el momento dipolar magnético en el caso de magnetorquers), y a , b son los coeficientes de regresión. Por ejemplo, para sensores solares, la relación entre la resolución angular σ y el consumo de potencia fue aproximada por el modelo $y = 6.27 \times 10^{-1} x^{-0.34}$, mientras que la masa siguió $y = 9.58 \times 10^{-2} x^{-0.68}$. Relaciones similares se derivaron para los otros componentes.

Las Figuras 6.1 y 6.2 ilustran las regresiones ajustadas mediante leyes de potencia para los cuatro tipos de componentes, mostrando los datos empíricos junto a las curvas de regresión. Estas expresiones se utilizaron para construir modelos sustitutos que relacionan las especificaciones técnicas de cada componente con su penalización en masa y consumo, integrándolos como términos de costo dentro del esquema de optimización multiobjetivo.

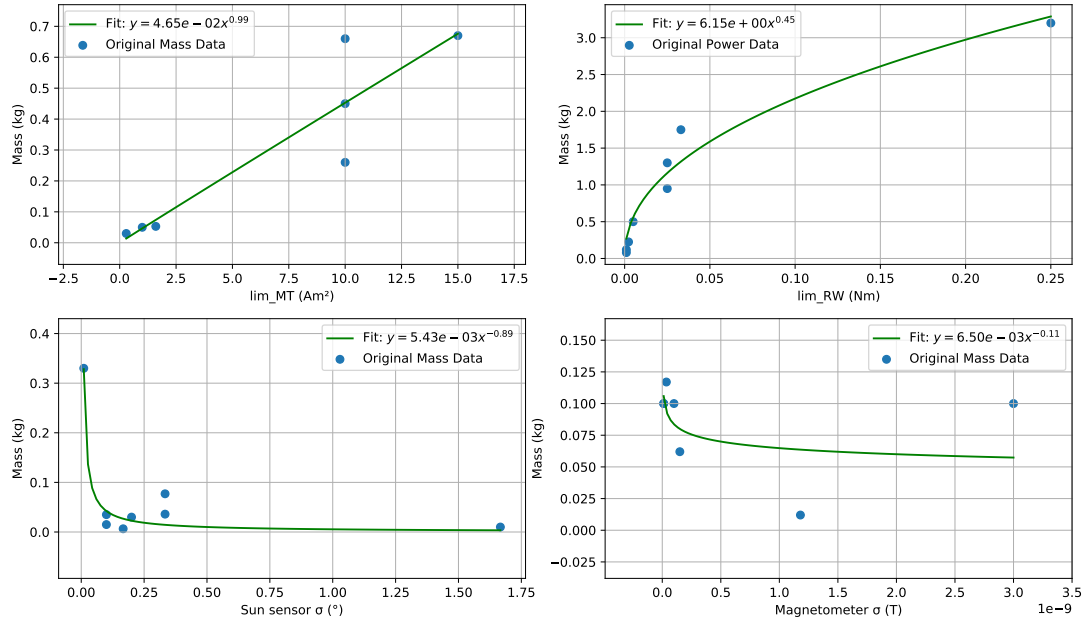


Figura 6.1 Modelos de regresión no lineal ajustados a datos empíricos que relacionan variables técnicas de componentes con masa para magnetorquers (arriba-izquierda), ruedas de reaccion (arriba-derecha), sensores solares (abajo-izquierda) y magnetómetros (abajo-derecha). Cada curva verde corresponde a un ajuste de tipo ley de potencias.

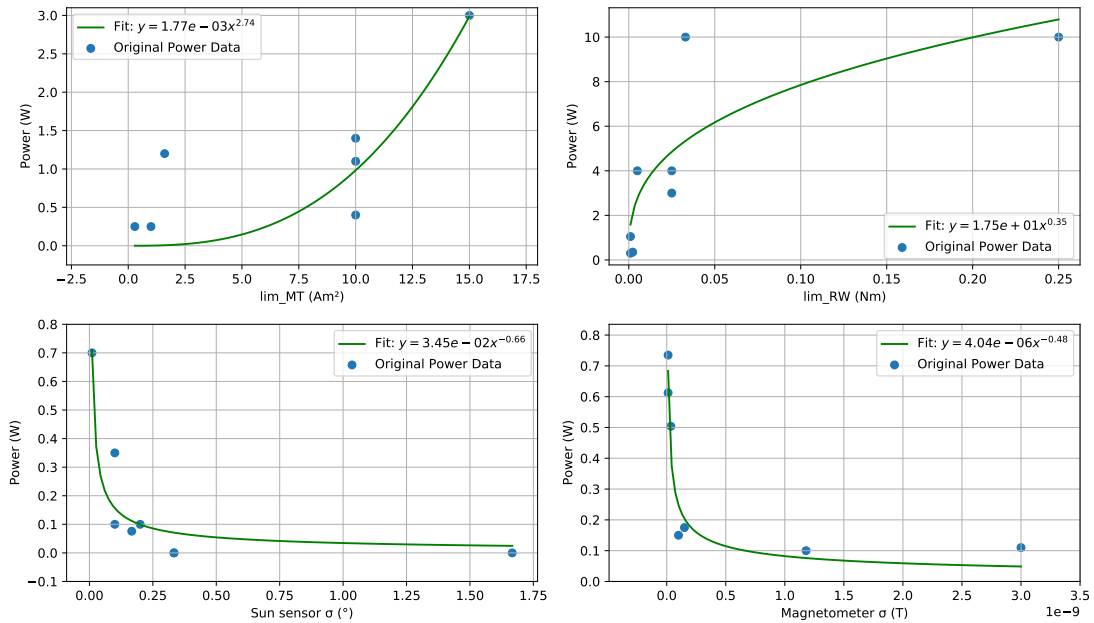


Figura 6.2 Modelos de regresión no lineal ajustados a datos empíricos que relacionan variables técnicas de componentes con consumo promedio de potencia para magnetorquers (arriba-izquierda), Ruedas de reaccion (arriba-derecha), sensores solares (abajo-izquierda) y magnetómetros (abajo-derecha). Cada curva verde corresponde a un ajuste de tipo ley de potencias.

Finalmente, es importante señalar que el usuario podrá seleccionar libremente qué componentes físicos y métricas de apuntamiento incluir en la optimización. Para ello, simplemente se excluyen las funciones f_i correspondientes, adaptando la formulación del problema a los requisitos específicos de cada misión.

6.3 Selección de línea base mediante matriz de decisión

A diferencia del Capítulo 5, donde las configuraciones de hardware bajo, medio y alto se utilizaron de forma representativa para verificar la suite de simulación, en esta sección dichas configuraciones se establecen formalmente como líneas base mediante un método convencional de toma de decisiones multicriterio (MCDM).

La matriz de decisión evaluó sensores y actuadores disponibles comercialmente en función de los MoPs de apuntamiento y los SE envelopes considerados en la función de costo (37). De este modo, se definieron tres configuraciones representativas de referencia (nivel bajo, medio y alto), que permiten comparar posteriormente los resultados de la optimización.

Las Tablas 6.1 y 6.2 resumen el desempeño de apuntamiento y los costos asociados a los SE envelopes para las tres configuraciones seleccionadas mediante MCDM. Los resultados muestran una mejora consistente en precisión, agilidad y *jitter* a medida que aumenta la calidad de sensores y actuadores. Por ejemplo, la precisión de apuntamiento mejora en un 99.98% al pasar de la configuración de nivel bajo a la de nivel alto. De manera similar, el *jitter* disminuye en 7 órdenes de magnitud.

Sin embargo, estas mejoras en el desempeño se obtienen a expensas de un mayor consumo de recursos. La Tabla 6.2 muestra que tanto el consumo del sensor solar como la masa del magnetorquer aumentan en 3 órdenes de magnitud. Estas tendencias evidencian el compromiso fundamental entre el desempeño de apuntamiento y las restricciones de los SE en el diseño del ADCS de CubeSats.

Tabla 6.1 Costo de MoPs de apuntamiento para niveles de componentes seleccionados por MCDM.

MoP de apuntamiento	f_i Nivel bajo	f_i Nivel medio	f_i Nivel alto
Θ [°]	0.9350	3.1618×10^{-2}	1.4104×10^{-4}
T_s^* [−]	0.02105	1.21×10^{-6}	4.9×10^{-7}
J [W/Hz]	1.9523	2.2089×10^{-3}	5.3436×10^{-7}

Tabla 6.2 Costo de SE envelopes para niveles de componentes seleccionados por MCDM.

SE envelope	f_i Nivel bajo	f_i Nivel medio	f_i Nivel alto
Potencia promedio Magnetómetro [W]	2.4815×10^{-3}	6.4977×10^{-2}	0.4974
Masa Magnetómetro [kg]	3.1685×10^{-3}	6.6962×10^{-3}	1.0676×10^{-2}
Potencia promedio Sensor solar [W]	6.0485×10^{-4}	1.2637×10^{-2}	0.5196
Masa Sensor solar [kg]	1.1834×10^{-5}	7.1312×10^{-4}	0.1070
Potencia promedio Magnetorquer [W]	3.5472×10^{-9}	8.1273×10^{-6}	8.728
Masa Magnetorquer [kg]	1.8640×10^{-4}	3.0515×10^{-3}	0.4609

La Figura 6.3 entrega una visualización de estos resultados mediante un gráfico radial para las tres configuraciones seleccionadas por MCDM. Cada eje en el gráfico corresponde a un componente de la función de costo, y los valores han sido normalizados usando *min-max scaling* entre los tres niveles de hardware, con el fin de permitir una comparación visual significativa. El gráfico muestra claramente que el hardware de nivel alto minimiza las métricas asociadas al desempeño, pero maximiza el consumo de potencia y masa. Por el contrario, la configuración de nivel bajo minimiza el uso de recursos, pero degrada severamente el desempeño de apuntamiento.

Esta comparación visual refuerza la necesidad de un enfoque sistemático que equilibre objetivos contrapuestos, más allá de lo que una matriz de decisión estática puede lograr.

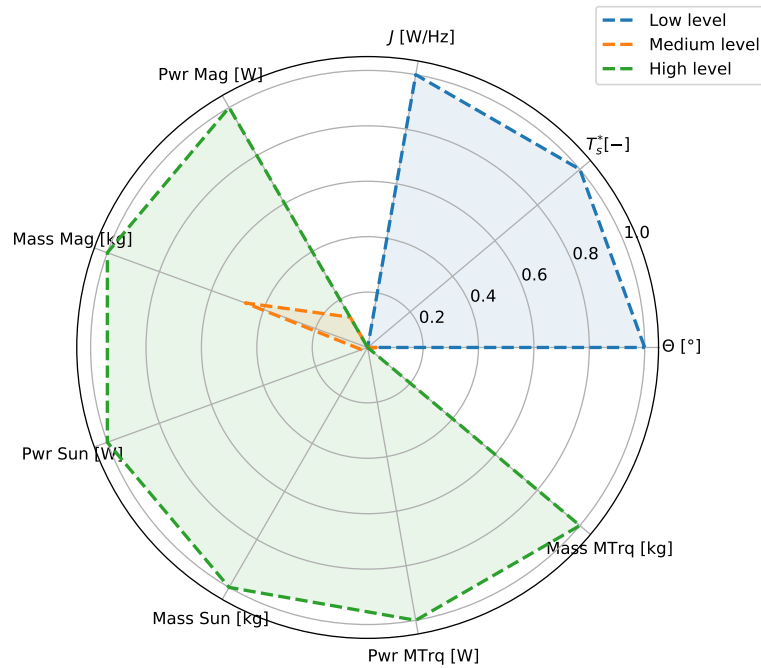


Figura 6.3 Gráfico radial normalizado de las funciones de costo f_i para cada nivel de componentes (Bajo, Medio, Alto).

6.4 Pruebas de convexidad en scipy.optimize

Para determinar si el problema es convexo, se utiliza la librería `scipy.optimize.minimize`, especializada en la búsqueda de mínimos locales. Estos serían suficientes si la suite de simulación resultara ser convexa, y para ello se realizan las pruebas correspondientes para visualizar si en distintos casos logra llegar a un mínimo global

6.4.1 Análisis caso de optimización simple.

El primer caso de estudio consiste en optimizar únicamente el MoP de exactitud de apuntamiento (accuracy), dándole prioridad a la optimización a los sensores para los SE envelopes y utilizando

magnetorquers como actuadores. Esto significa que solo se incluyen en la función de costo global la función de la exactitud de apuntamiento para los f_{MOP} y la potencia/masa de los magnetómetros y sensores sol para los f_{SE} .

Para iniciar el análisis de convexidad, se configuró una malla de 20x20 puntos, evaluando 20 valores dentro de los límites inferiores y superiores de las desviaciones estándar del magnetómetro y del sensor solar, según los valores y parámetros del Capítulo 5. Esta grilla permite visualizar gráficamente la aproximación de la ubicación del mínimo global al considerar las funciones previamente descritas. La Figura 6.4 muestra esta grilla, donde los ejes x e y representan las desviaciones estándar evaluadas para el sensor solar y el magnetómetro, respectivamente, y el eje z corresponde al valor de la función de costo, acompañado por un mapa de colores que también representa esta función objetivo. A partir de estas evaluaciones, se logra minimizar la función a un valor de $f(x) = 3.4928$, obtenido en el par $x = (0.641 \text{ [nT]}, 1.41[^\circ])$.

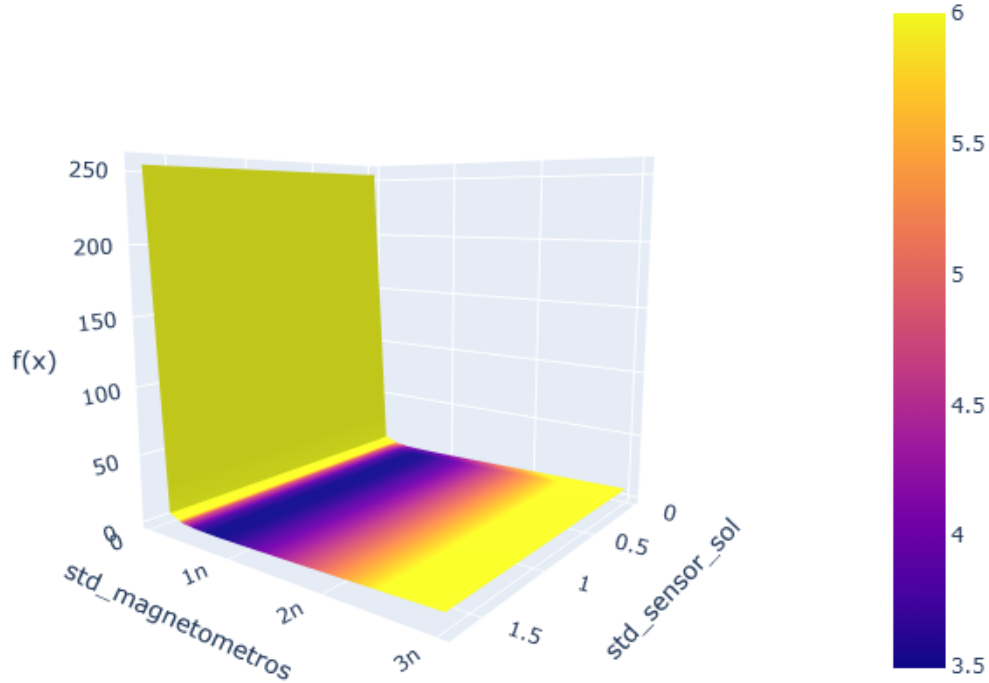


Figura 6.4 Grilla 20x20 caso de estudio optimizando exactitud de apuntamiento con los SE envelopes de los sensores.

Con esta base, se realizó un análisis utilizando diferentes condiciones iniciales para la optimización con minimize, empleando diversos solvers. Se eligieron aquellos solvers que no requieren el cálculo del Hessiano, como L-BFGS-B, Nelder-Mead y Powell, debido a la dificultad de obtener la doble derivada de la suite de optimización en relación con las desviaciones estándar de los sensores.

Los resultados de la minimización para las condiciones iniciales de 1.5 [nT] y 0.5 [°] en los solvers mencionados se presentan en la Tabla 6.3. Los valores obtenidos para la función objetivo son muy similares entre sí, incluso mostrando resultados inferiores al mínimo encontrado en la

evaluación previa con la grilla.

Adicionalmente, se evaluaron diferentes condiciones iniciales para los sensores, utilizando 0.5 [nT] y 0.8 [°]. Los resultados obtenidos para los mismos solvers se muestran en la Tabla 6.3, revelando valores prácticamente idénticos a los obtenidos con las primeras condiciones. El único solver que muestra ligeras diferencias es Nelder-Mead, aunque estas discrepancias son mínimas y pueden considerarse insignificantes.

Tabla 6.3 Resultados de optimización con distintos solvers y condiciones de incertidumbre intermedias.

Solver	$c.i$ [0.5°, 1.5 nT]		$c.i$ [0.8°, 0.5 nT]	
	$[\sigma_{s,opt}[^{\circ}], \sigma_{b,opt}[nT]]$	f_{opt}	$[\sigma_{s,opt}, \sigma_{b,opt}]$	f_{opt}
L-BFGS-B	$[3.97 \times 10^{-1}, 3.0]$	3.475	$[3.96 \times 10^{-1}, 3.0]$	3.475
Nelder-Mead	$[3.97 \times 10^{-1}, 1.97]$	3.476	$[3.99 \times 10^{-1}, 1.08]$	3.479
Powell	$[4.03 \times 10^{-1}, 1.15]$	3.480	$[4.03 \times 10^{-1}, 1.15]$	3.480

Por otro lado, se presentan condiciones iniciales que se encuentran en los extremos de las restricciones, las cuales se muestran en la Tabla 6.4. En todos los casos, se muestra una variación de al menos 3 milésimas en cada solver, al igual que los casos de estudio anteriores, siendo cualquier opción útil en caso de querer optimizar solo un MoP de apuntamiento, con un componente físico.

Tabla 6.4 Resultados de optimización con distintos solvers y condiciones de incertidumbre extremas.

Solver	$c.i$ [1.6°, 3.0 nT]		$c.i$ [0.05°, 0.015 nT]	
	$[\sigma_{s,opt}[^{\circ}], \sigma_{b,opt}[nT]]$	f_{opt}	$[\sigma_{s,opt}, \sigma_{b,opt}]$	f_{opt}
L-BFGS-B	$[3.97 \times 10^{-1}, 3.0]$	3.475	$[3.96 \times 10^{-1}, 3.0]$	3.475
Nelder-Mead	$[3.97 \times 10^{-1}, 3.0]$	3.475	$[3.96 \times 10^{-1}, 2.33]$	3.475
Powell	$[4.03e - 01, 1.15]$	3.480	$[4.03 \times 10^{-1}, 1.15]$	3.480

Una vez presentados los resultados numéricos, se procede a cuantificar la consistencia entre soluciones obtenidas por los distintos *solvers*. Se observa que, en todos los casos, las soluciones óptimas de los pares $[\sigma_{b,opt}, \sigma_{s,opt}]$ difieren en menos de 0.07 unidades, y los valores de la función objetivo presentan variaciones inferiores a 0.005 unidades. Estas diferencias corresponden a menos del 0.15% respecto al valor mínimo encontrado, lo que indica que los algoritmos convergen hacia soluciones prácticamente idénticas.

Este análisis respalda que el problema de optimización simple es suficientemente suave y bien condicionado para ser abordado con los métodos utilizados. En particular, la función objetivo muestra un comportamiento regular dentro del espacio de búsqueda definido, lo cual permite a

todos los *solvers* considerados alcanzar soluciones cercanas al mínimo global con independencia de la condición inicial.

6.4.2 Análisis caso de optimización complejo.

El segundo caso analiza la optimización de todos los MoP de apuntamiento, considerando tanto sensores como actuadores del tipo magnetorquer, lo que significa que se utilizan las nueve funciones de costo para la optimización.

En este caso, se generó una malla de 20x20x6 puntos para evaluar las desviaciones estándar de los sensores y el límite de torque del magnetorquer dentro de un rango determinado, similar al análisis previo. En esta grilla, los ejes x, y y z representan las variables de optimización, mientras que el mapa de colores refleja la variación en los resultados de la función objetivo, permitiendo identificar la posible ubicación de un mínimo global. La grilla se presenta en la Figura 6.5, donde se obtiene un valor mínimo global de 4.2171, alcanzado en los valores óptimos de 2.21 [nT], 0.36 [°] y 14.23 [Am²].

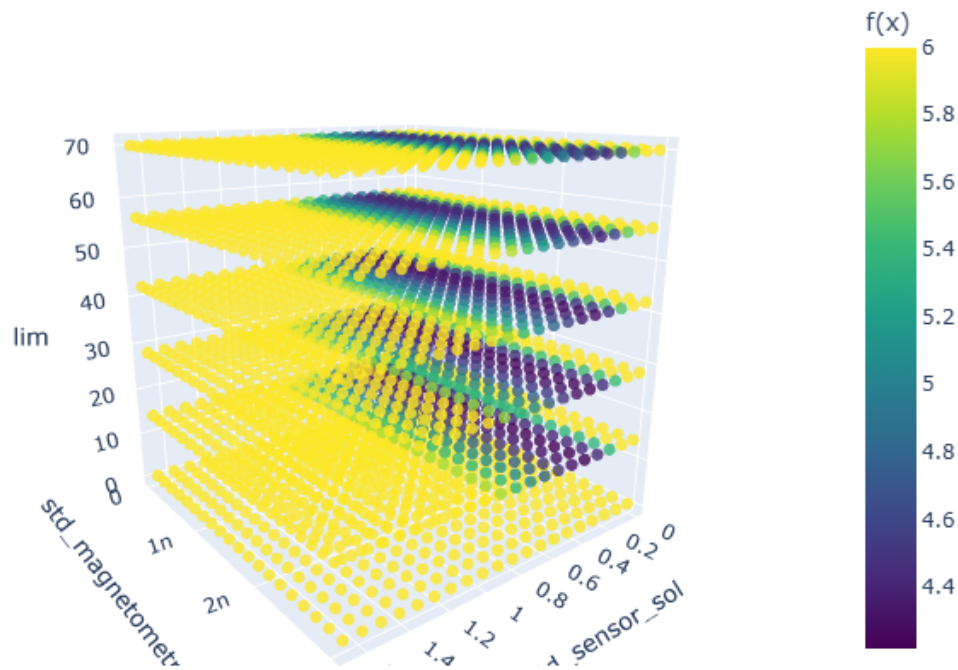


Figura 6.5 Grilla 20x20x6 caso de estudio utilizando todas las funciones de costo para la optimización.

Al igual que en el caso de optimización simple, aquí se probaron cuatro condiciones iniciales para evaluar si los solvers son capaces de aproximarse al valor mínimo global obtenido en la grilla.

Los resultados para los primeros casos de condiciones iniciales se presentan en las Tabla 6.5. En la primera figura, se observa que los solvers L-BFGS-B y Powell logran valores cercanos al

mínimo global, mientras que Nelder-Mead muestra una diferencia de 0.6 respecto al objetivo. En el segundo caso, Nelder-Mead alcanza el mismo mínimo que en el primer intento, mientras que L-BFGS-B se queda atrapado en otro mínimo local, distante del objetivo deseado. Por otro lado, Powell logra nuevamente alcanzar el mínimo esperado, lo que lo posiciona como un solver más confiable para este tipo de problema.

Tabla 6.5 Resultados de optimización compleja con distintos solvers y condiciones de incertidumbre intermedias.

Solver	$c.i$ $[0.5^\circ, 1.5 \text{ nT}, 1.0 \text{ Am}^2]$		$c.i$ $[0.8^\circ, 0.5 \text{ nT}, 1.0 \text{ Am}^2]$	
	$[\sigma_{s,opt}[^{\circ}], \sigma_{b,opt}[\text{nT}], \tau[\text{Am}^2]]$	f_{opt}	$[\sigma_{s,opt}[^{\circ}], \sigma_{b,opt}[\text{nT}], \tau[\text{Am}^2]]$	f_{opt}
L-BFGS-B	$[3.73 \times 10^{-1}, 3.0, 6.53]$	4.251	$[5.23 \times 10^{-1}, 3.0, 7.46 \times 10^{-1}]$	5.418
Nelder-Mead	$[3.95 \times 10^{-1}, 1.83, 9.23 \times 10^{-1}]$	4.844	$[3.94 \times 10^{-1}, 0.71, 9.18 \times 10^{-1}]$	4.851
Powell	$[3.75 \times 10^{-1}, 1.16, 3.60]$	4.216	$[3.70 \times 10^{-1}, 1.15, 3.69]$	4.216

Además, se probaron dos casos adicionales con condiciones iniciales extremas, es decir, cercanas a los límites inferiores y superiores de las restricciones de las variables de optimización. Los resultados se presentan en la Tabla 6.6. En el primer caso extremo, tanto L-BFGS-B como Nelder-Mead quedaron atrapados nuevamente en mínimos locales, más alejados del objetivo deseado, con valores de 11 y 6.2, respectivamente. Sin embargo, Powell logró alcanzar nuevamente el valor mínimo esperado. Para el segundo caso extremo, Nelder-Mead, al igual que Powell, logró llegar al mínimo global identificado en la grilla. Estos resultados refuerzan la fiabilidad de Powell para este tipo de problema, incluso en condiciones iniciales más extremas.

Tabla 6.6 Resultados de optimización compleja con distintos solvers y condiciones de incertidumbre extremas.

Solver	$c.i$ $[1.6^\circ, 3.0 \text{ nT}, 0.5 \text{ Am}^2]$		$c.i$ $[0.05^\circ, 0.015 \text{ nT}, 40 \text{ Am}^2]$	
	$[\sigma_{s,opt}[^{\circ}], \sigma_{b,opt}[\text{nT}], \tau[\text{Am}^2]]$	f_{opt}	$[\sigma_{s,opt}[^{\circ}], \sigma_{b,opt}[\text{nT}], \tau[\text{Am}^2]]$	f_{opt}
L-BFGS-B	$[1.60, 3.0, 5.0 \times 10^{-1}]$	11.05	$[6.87 \times 10^{-1}, 3.0, 3.22]$	5.001
Nelder-Mead	$[3.514 \times 10^{-1}, 3.0, 4.76 \times 10^{-1}]$	6.228	$[3.62 \times 10^{-1}, 1.08, 3.35]$	4.216
Powell	$[3.66 \times 10^{-1}, 1.15, 3.60]$	4.215	$[3.72 \times 10^{-1}, 1.15, 3.61]$	4.215

Una vez obtenidos los resultados de las Tablas 6.5 y 6.6, se cuantifica la variación en las soluciones entregadas por los diferentes *solvers*. Se observa que la función objetivo presenta una variación máxima de 6.834 unidades entre las soluciones obtenidas, correspondiente al caso extremo con L-BFGS-B. Sin embargo, en las soluciones más cercanas al mínimo global, la diferencia entre valores óptimos es inferior a 0.04 unidades, lo que representa menos del 1% de variación relativa respecto al mínimo global identificado ($f = 4.2171$). Esto demuestra que, aunque la presencia de múltiples variables y funciones complica la superficie de la función objetivo, algunos *solvers* como Powell logran converger de forma robusta y consistente a soluciones óptimas, incluso con condiciones iniciales extremas. La comparación también permite concluir que el problema

mantiene características locales suaves, pero puede presentar múltiples mínimos locales según el punto de partida.

6.4.3 Análisis de convexidad y elección del método de optimización

Los resultados obtenidos en los dos casos de estudio permiten caracterizar la superficie de la función objetivo y evaluar la idoneidad de los distintos *solvers* empleados (L-BFGS-B, Nelder–Mead y Powell), todos ellos adecuados para problemas en los que no se dispone del Hessiano explícito.

Caso simple (un solo MoP). La variación máxima entre los valores óptimos entregados por los tres *solvers* fue $\Delta f_{\max} = 5 \times 10^{-3}$, equivalente al 0.14 % respecto al valor mínimo global ($f = 3.478$). La dispersión en la desviación estándar del magnetómetro no superó el 1.7 % y, aunque la variable del sensor solar mostró diferencias más amplias —asociadas a convergencia en el límite superior permitido—, la función objetivo permaneció prácticamente inalterada. Estos resultados indican que, para este subproblema reducido, la superficie es localmente suave y todos los algoritmos alcanzan soluciones casi idénticas.

Caso complejo (todos los MoP). Cuando se incluyeron las nueve funciones de costo y el límite de torque del magnetorquer, la complejidad de la superficie aumentó. Entre los 12 experimentos realizados (4 condiciones iniciales $\times$ 3 *solvers*), la diferencia global más alta fue $\Delta f_{\max} = 6.834$ unidades, observada con L-BFGS-B en un caso extremo. No obstante, en las soluciones cercanas al mínimo global ($f = 4.217$) la discrepancia se redujo a $\Delta f = 0.04$ (menos del 1 %). Powell convergió al mínimo global en los 4 intentos; Nelder–Mead y L-BFGS-B quedaron atrapados en mínimos locales en al menos uno de los escenarios.

Implicaciones sobre la convexidad. Si la función objetivo fuera estrictamente convexa, los tres *solvers* habrían convergido siempre al mismo punto óptimo con independencia de la condición inicial. La presencia de múltiples mínimos locales y la sensibilidad a la semilla inicial demuestran que el problema pertenece a la clase *General Non-linear Problem* (GNP) y, por tanto, no es convexo. Ello se debe a la combinación de funciones de costo no lineales y a la interacción no trivial entre sensores y actuadores dentro de los *System-Engineering envelopes*.

Elección del método. Considerando la variación inferior al 1 % entre sus mejores soluciones y la capacidad demostrada para evitar mínimos locales aun con condiciones iniciales extremas, el algoritmo de Powell se perfila como la alternativa más robusta para este tipo de problemas no convexos. Su consistencia en la convergencia y la ausencia de requerimientos sobre gradientes de segundo orden lo convierten en el *solver* preferente para la implementación final de la suite de optimización.

Descarte del uso de la librería *Pyomo*. Se evaluó inicialmente el uso de la librería *Pyomo* por su capacidad de formular y resolver problemas de optimización no lineal de forma robusta, especialmente en contextos de optimización global. Sin embargo, su integración con la suite desarrollada resultó inviable. Esto se debe a que *Pyomo* emplea una estructura de modelado declarativa basada en variables simbólicas propias, lo cual entra en conflicto con librerías ampliamente utilizadas como *numpy*, *scipy* y *matplotlib*, que operan sobre estructuras numéricas explícitas. Esta incompatibilidad dificultó la evaluación directa de funciones definidas previa-

mente, por lo que se optó por utilizar `scipy.optimize.minimize`, que permite una integración más fluida con el entorno de trabajo utilizado.

6.5 Ejemplo de optimización utilizando magnetorquer

Dado el análisis de convexidad realizado en la sección anterior, se concluyó que el problema de optimización no es convexo y presenta múltiples mínimos locales. En ese contexto, el método de Powell demostró ser el más robusto y consistente, alcanzando el mínimo global en todos los casos evaluados, incluso bajo condiciones iniciales extremas. El método de Powell es un algoritmo de optimización no lineal sin derivadas basado en búsquedas por direcciones conjugadas [73], implementado a través del módulo de optimización `SciPy` en Python.

Esta sección presenta la integración de dicha suite de simulación de actitud del satélite con un solucionador de optimización no lineal para determinar la combinación óptima de componentes físicos—específicamente sensores y magnetorquers—bajo restricciones específicas de misión. El enfoque aprovecha el modelo de desempeño dinámico para guiar la selección de componentes dentro de límites técnicos predefinidos, con el objetivo de equilibrar el rendimiento de apuntamiento con SE envelopes, tales como masa y potencia.

La simulación se realiza utilizando los parámetros orbitales y geométricos definidos para la misión SUCHAI-3, mientras que los parámetros y restricciones de optimización correspondientes se resumen en la Tabla 6.7. Para evaluar la robustez del enfoque, se consideran múltiples condiciones iniciales dentro de los límites impuestos, lo que permite al solucionador explorar el espacio de diseño y converger hacia una configuración óptima de hardware. Estos límites, detallados en la Tabla 6.7, corresponden a los mismos rangos de parámetros utilizados para construir las regresiones de ley potencial para las funciones de costo basadas en los envelopes SE, asegurando la consistencia entre el marco de optimización y los modelos subyacentes de los componentes.

Tabla 6.7 Parámetros de optimización

Parámetro	Valor
Límites de σ_b [T]	$[0.012 \times 10^{-9}, 3 \times 10^{-9}]$
Límites de σ_s [°]	[0.01, 1.67]
Límites de τ [$A \cdot m^2$]	[0.29, 15]

Los casos de estudio están estructurados mediante la optimización de todos los componentes físicos incluidos en la función de costo, lo que significa que los términos de costo asociados a los SE envelopes (potencia y masa) están siempre activos y asignados con un peso constante de $W_i = 1$. Cada caso se enfoca en una única métrica de desempeño de apuntamiento (MoP), lo que se logra asignando un peso $W_i = 1$ a la métrica seleccionada y cero a las demás, aislando así su influencia en el proceso de optimización.

Aunque las funciones de costo no fueron normalizadas explícitamente, un análisis preliminar de sus magnitudes a través de las simulaciones a nivel de componente (bajo, medio y alto) reveló que todos los términos de costo están en una escala comparable dentro del dominio de interés.

Esto justificó el uso de pesos uniformes ($W_i = 1$) en la formulación multiobjetivo sin introducir sesgos hacia una métrica en particular.

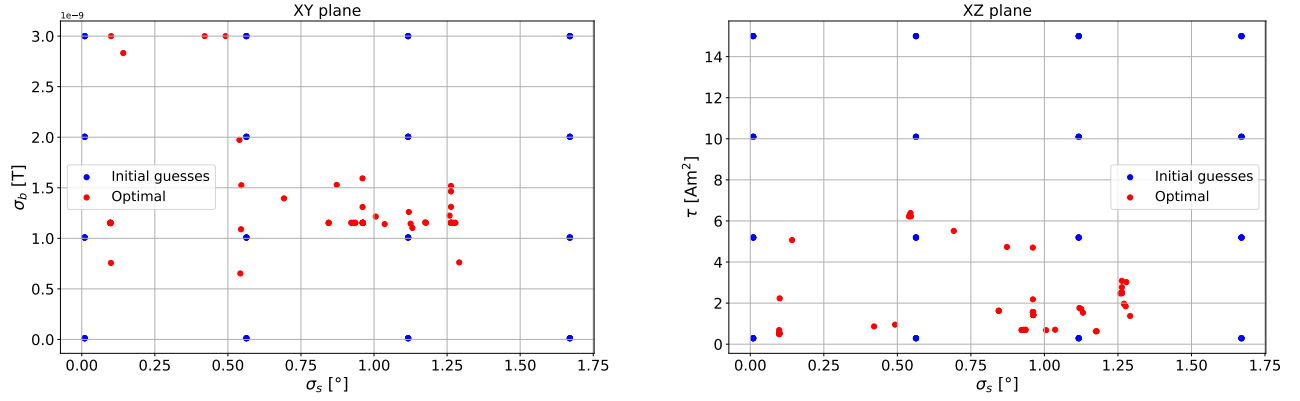


Figura 6.6 Primer caso de estudio: optimización de componentes físicos seleccionando la exactitud como MoP.

La Figura 6.6 presenta los resultados de la optimización de la accuracy mediante dos planos del espacio 3D compuesto por σ_b , σ_s y τ . Los resultados revelan que diferentes condiciones iniciales (azul) conducen a óptimos distintos (rojo), lo que indica la existencia de múltiples mínimos locales. En el plano xz , los valores óptimos para el límite de torque del actuador τ se encuentran de manera consistente cerca del límite inferior, lo que sugiere un beneficio limitado al aumentar la autoridad del actuador para esta métrica. Mientras tanto, el plano xy muestra que las desviaciones estándar óptimas de los sensores, en particular σ_s , tienden a agruparse en valores intermedios, con σ_b mostrando una dispersión moderada. Esta distribución implica que la precisión de apuntamiento es más sensible a la precisión de los sensores —especialmente del sensor solar— que a la potencia del actuador.

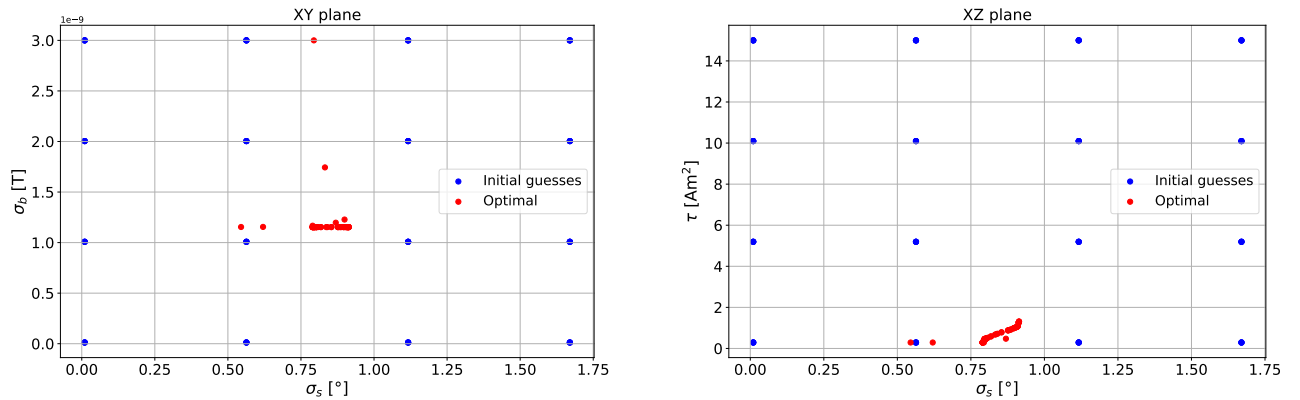


Figura 6.7 Segundo caso de estudio: optimización de componentes físicos seleccionando la agilidad como MoP.

El segundo caso de optimización busca mejorar la agilidad del sistema minimizando el tiempo de establecimiento. Como se muestra en la Figura 6.7, las soluciones óptimas en el plano xz se

concentran en la región central inferior, donde el torque τ es bajo y σ_s es moderado. Esto sugiere que alcanzar respuestas rápidas no requiere un torque máximo del actuador. El optimizador selecciona de manera consistente valores de τ apenas por debajo de $6.4 \text{ A}\cdot\text{m}^2$, coincidiendo con el torque máximo observado en las simulaciones—más allá del cual un mayor torque no ofrece mejoras en el rendimiento.

En el plano xy , las soluciones se agrupan simétricamente en torno a niveles moderados de ruido en los sensores, lo que indica que tanto σ_s como σ_b se benefician de una configuración equilibrada e intermedia. El optimizador prefiere diseños balanceados por sobre configuraciones extremas de sensores, ya que sensores extremadamente precisos o excesivamente ruidosos aportan un beneficio mínimo en el tiempo de establecimiento y son penalizados por consumo de potencia o masa adicional. Estos hallazgos sugieren que, para misiones orientadas a la agilidad, un diseño con fidelidad sensorial moderada y una fuerza de actuador suficiente—pero no excesiva—ofrece el mejor compromiso.

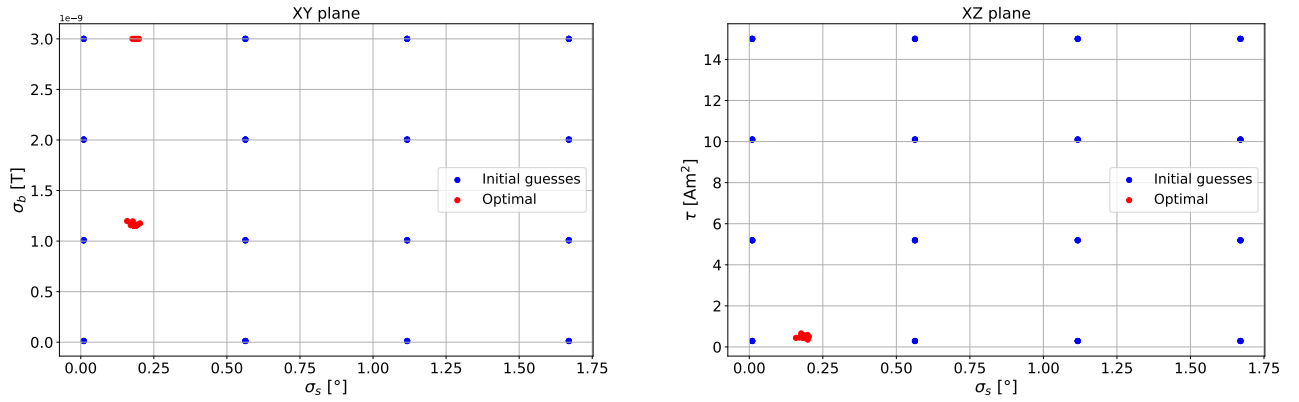


Figura 6.8 Tercer caso de estudio: optimización de componentes físicos seleccionando el jitter como MoP.

El tercer caso de estudio, mostrado en la Figura 6.8, se enfoca en la minimización del jitter, definido a través de la densidad espectral de potencia (PSD) de la señal de actitud η . El plano xy revela una clara concentración en la región central izquierda, favoreciendo valores bajos de σ_s y valores moderados de σ_b . Esto indica una fuerte sensibilidad del jitter a la precisión del sensor solar, probablemente debido a su papel dominante en la retroalimentación de alta frecuencia.

En el plano xz , las configuraciones óptimas se agrupan en la esquina inferior izquierda, donde tanto τ como σ_s se encuentran minimizados. Esto sugiere que reducir la autoridad del actuador, junto con el uso de sensores solares precisos, contribuye a amortiguar las oscilaciones de alta frecuencia. Límites de torque más bajos probablemente generan acciones de control más suaves, disminuyendo la probabilidad de correcciones agresivas que incrementen el jitter.

Al igual que en los casos anteriores, la forma de la función de costo—que penaliza masa y potencia—desincentiva capacidades excesivas de actuadores o sensores ultra-precisos. El optimizador, por lo tanto, favorece configuraciones que reducen el jitter manteniendo la eficiencia. Estos resultados destacan la importancia del co-diseño del hardware de sensores y actuadores considerando la dinámica de control, en particular cuando el objetivo es el desempeño en el dominio frecuencial.

Para reforzar el análisis comparativo de los resultados de optimización, la Figura 6.9 presenta un conjunto de gráficos radiales normalizados que visualizan los valores de las funciones de costo f_i en todos los casos de optimización, incluyendo tanto configuraciones de hardware de bajo como de alto rendimiento.

Estos gráficos radiales muestran cómo cada estrategia de optimización mejora la métrica de desempeño objetivo, al mismo tiempo que respeta las restricciones realistas de masa y potencia. En la mayoría de los casos, las configuraciones optimizadas se agrupan cerca del centro, reflejando los compromisos impuestos por la función de costo y las limitaciones a nivel de sistema. Cabe destacar que muchas de estas soluciones igualan o incluso superan el rendimiento de configuraciones con componentes de alta gama, pero con demandas de recursos significativamente menores. Esto pone en evidencia las limitaciones de los enfoques tradicionales de toma de decisiones multicriterio (MCDM), los cuales entregan únicamente un ranking estático y pasan por alto el acoplamiento dinámico entre sensores, actuadores y algoritmos de control. En contraste, el optimizador basado en simulación propuesto en este trabajo explora un espacio de compromisos más rico, guiado por la dinámica del ADCS, logrando un desempeño superior o equivalente con menores costos de sistema.

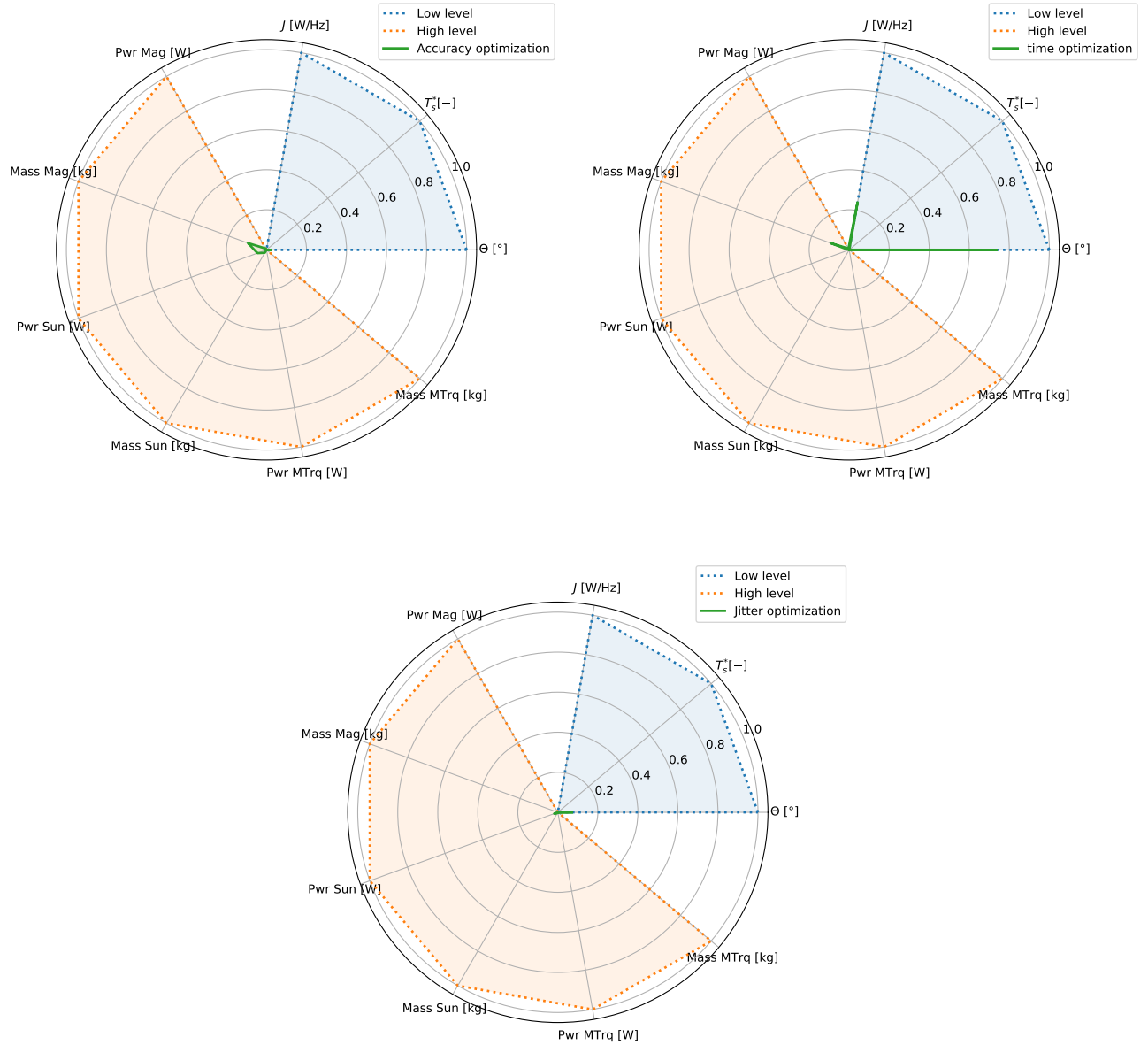


Figura 6.9 Gráfico de radar normalizado de las funciones de costo f_i para cada MoP de los componentes, respecto a componentes de nivel bajo y alto; Arriba a la izquierda: optimización de exactitud de apuntamiento, Arriba a la derecha: optimización de tiempo de asentamiento y Abajo: optimización de jitter.

7 Conclusiones

Se ha desarrollado un marco teórico robusto para el modelado del ambiente espacial y del sistema de determinación y control de actitud (ADCS) en CubeSats destinados a misiones de observación terrestre, utilizando literatura especializada de la última década. Este marco permitió validar los modelos empleados, garantizando su coherencia con investigaciones recientes y con especificaciones de componentes comerciales, y brindó una base sólida para la implementación de modelos de simulación realistas.

Para el ambiente espacial, se utilizó el propagador SGP4, logrando representar de forma precisa las posiciones y velocidades orbitales del CubeSat bajo perturbaciones típicas de órbitas bajas. Asimismo, se modelaron sensores (magnetómetros y sensores solares) y actuadores (magnetorquers y ruedas de reacción), lo que permitió simular la dinámica de actitud del CubeSat en condiciones operativas representativas.

En cuanto a la estimación de estado, se implementó con éxito un filtro de Kalman semiextendido, capaz de estimar cuaterniones y velocidades angulares con un margen de error inferior al 5% respecto al modelo dinámico. Esta herramienta resultó eficaz para integrar las mediciones de múltiples sensores y actualizar continuamente el estado del sistema, aspecto clave para las tareas de apuntamiento y control.

Para el control de actitud, se probaron los controladores PD y LQR en escenarios equivalentes. El controlador LQR fue seleccionado por su mejor desempeño en términos de estabilidad y precisión del apuntamiento, optimizando los parámetros del sistema de forma más efectiva.

En el ámbito de la optimización, se desarrolló y validó un marco basado en simulaciones que permite vincular directamente las especificaciones de sensores y actuadores con métricas de desempeño de apuntamiento (MoP). A diferencia de aproximaciones heurísticas o métodos de decisión multicriterio, este enfoque habilita un análisis sistemático de compromisos entre exactitud de apuntamiento, jitter y agilidad, considerando simultáneamente restricciones de masa y potencia de los componentes comerciales.

Los resultados muestran que la exactitud de apuntamiento y el jitter están principalmente determinados por la calidad de los sensores, destacando la relevancia de magnetómetros y sensores solares de baja desviación estándar. Por su parte, la agilidad depende de la capacidad de los actuadores, aunque un exceso en la autoridad de control presenta rendimientos decrecientes bajo limitaciones estrictas de recursos. Asimismo, las ruedas de reacción demostraron un desempeño superior en comparación con los magnetorquers, si bien a un costo mayor en términos de consumo energético y complejidad.

Entre los métodos evaluados en `scipy.optimize`, el algoritmo de Powell se destacó por su capacidad para abordar problemas no convexos sin requerir derivadas, encontrando soluciones cercanas al mínimo global de manera eficiente. La verificación cuantitativa realizada con datos del CubeSat SUCHAI-3 evidenció una concordancia significativa entre los resultados dinámicos y los obtenidos por el optimizador, confirmando la robustez del marco propuesto.

Este enfoque ofrece una arquitectura práctica y escalable para el diseño de ADCS en CubeSats, con potencial de extensión hacia plataformas de mayor fidelidad —como Basilisk o entornos hardware-in-the-loop— e integración futura de objetivos múltiples que incluyan restricciones

térmicas o de comunicaciones. Con ello, se valida el cumplimiento de los seis objetivos específicos planteados, consolidando la suite como una herramienta efectiva para la selección óptima de componentes de ADCS en misiones de observación terrestre.

Referencias

- [1] nanosats.eu, “Nanosats database,” 2024. <https://www.nanosats.eu/>.
- [2] SatCatalog, “Cubesat launch costs,” 2022.
- [3] nanosats.eu, “Cubesat costs,” 2024. <https://www.nanosats.eu/tables>.
- [4] D. Odonohue, “Optical satellites,” 2024.
- [5] J. Wertz and W. Larson, *Space mission analysis and design*. Space technology Series, 1999.
- [6] M. Tacul, “Identificación y caracterización de índices de performance en el apuntamiento de payload en cubesats,” 2023. Accedido el: 16 de septiembre de 2024.
- [7] J. P. Mason, M. Baumgart, B. Rogler, C. Downs, M. Williams, T. N. Woods, S. Palo, P. C. Chamberlin, S. Solomon, A. Jones, X. Li, R. Kohnert, and A. Caspi, “Minxss-1 cubesat on-orbit pointing and power performance: The first flight of the blue canyon technologies xact 3-axis attitude determination and control system,” *Journal of Small Satellite Missions*, vol. 10, no. 3, pp. 651–662, 2017. <https://www.researchgate.net/publication/317672613>.
- [8] S.-G. Kim and J.-S. Chae, “Attitude control system performance estimation for next generation small satellite 1,” *World Congress on Aeronautics, Nano, Bio, Robotics and Energy (ANBRE15)*, vol. 1, p. Incheon, August 2015.
- [9] Z. Ye, Y. Xu, S. Zheng, X. Tong, X. Xu, S. Liu, H. Xie, S. Liu, C. Wei, and U. Stilla, “Resolving time-varying attitude jitter of an optical remote sensing satellite based on a time-frequency analysis,” *Optics Express*, vol. 28, p. 15805, May 2020.
- [10] C. Dennehy and O. S. Alvarez-Salazar, “A survey of the spacecraft line-of-sight jitter problem,” *NASA Technical Reports*, 2020.
- [11] G. Lavezzi, M. E. Grottey, and M. Ciarcia, “Attitude control strategies for an imaging cubesat,” *2019 IEEE International Conference on Electro Information Technology (EIT)*, vol. 1, pp. 149–155, May 2019.
- [12] R. Votel and D. Sinclair, “Comparison of control moment gyros and reaction wheels for small earth-observing satellites,” in *Conference on Small Satellites*, 2012.
- [13] E. P. Babcock, E. P. Babcock, and T. Bretl, “Cubesat attitude determination via kalman filtering of magnetometer and solar cell data,” in *25 th Annual AIAA/USU Conference on Small Satellites*, 2011.
- [14] D. Torczynski, R. Amini, and P. Massioni, “Magnetorquer based attitude control for a nanosatellite testplatform,” in *AIAA Infotech@Aerospace 2010*, American Institute of Aeronautics and Astronautics, 4 2010.
- [15] D. M. Vélez, E. I. Dawson, and N. E. Nassiff, “Attitude determination and control subsystem design for a cubesat,” 2012. <https://digitalcommons.wpi.edu/mqp-all/3322>.
- [16] A. Annenkova, S. Biktimirov, K. Latyshev, A. Mahfouz, P. Mukhachev, and D. Pritykin, “Cubesat adcs model for preliminary design procedures within a concurrent design approach,” 2019.

- [17] A. Rassõlkin, T. Vaimann, P. Org, A. Leibak, R. Gordon, and E. Priidel, “Adcs development for student cubesat satellites – taltech case study,” *Proceedings of the Estonian Academy of Sciences*, vol. 70, p. 268, 2021.
- [18] P. S. System, “Spacecraft control toolbox,” 2017. <http://support.psatellite.com//sct/index.php>.
- [19] Mathworks, “Model and simulate cubesats,” 2023. <https://la.mathworks.com/help/aeroblks/model-and-simulate-cubesats.html>.
- [20] Valispace, “Fan tutorials.” <https://docs.valispace.com/vhd/fan-tutorials>.
- [21] U. of Colorado, L. for Atmospheric, and S. Physics, “Basilisk,” 2024. <https://hanspeterschaub.info/basilisk/index.html>.
- [22] I. H. Gondal and M. T. Islam, “Ahp-based selection of sensors and actuators for small satellite attitude control system,” *Advances in Space Research*, vol. 60, no. 6, pp. 1273–1283, 2017.
- [23] M. Pereira and J. Sousa, “Fuzzy logic based component selection method for nanosatellite subsystems,” in *2014 IEEE Aerospace Conference*, pp. 1–7, 2014.
- [24] A. Kaya and M. Yildiz, “Multi-objective design optimization of a satellite attitude control system using pareto analysis,” *Engineering Optimization*, vol. 51, no. 5, pp. 791–808, 2019.
- [25] M. Ali and A. Kumar, “Application of machine learning for component selection in spacecraft adcs,” *Acta Astronautica*, vol. 177, pp. 360–368, 2020.
- [26] M. Toghraie and M. Azizi, “Simulation-based optimization of cubesat adcs design using genetic algorithms,” *Journal of Aerospace Engineering*, vol. 35, no. 2, 2022.
- [27] D. Izzo and F. Biscani, “A novel bayesian approach for robust spacecraft subsystem selection,” *IEEE Transactions on Aerospace and Electronic Systems*, vol. 50, no. 1, pp. 164–175, 2014.
- [28] A. Chaves, “On the coupling of orbit and attitude determination of satellite formations from atmospheric drag,” 2020.
- [29] H. Curtis, *Orbital Mechanics for Engineering Students*. Elsevier, 2014.
- [30] J. Wertz, *Spacecraft Attitude Determination and Control*, vol. 73. Springer Netherlands, 1978.
- [31] AcademiaLab, “Elementos orbitales.” <https://academia-lab.com/enciclopedia/elementos-orbitales/>.
- [32] F. L. Markley and J. L. Crassidis, *Fundamentals of Spacecraft Attitude Determination and Control*. Springer New York, 2014.
- [33] J. Sellers, *Understanding Space: An Introduction to Astronautics*. Space technology series, McGraw-Hill, 2000.
- [34] H. Schaub and J. L. Junkins, “Analytical mechanics of space systems,” 2003. <https://api.semanticscholar.org/CorpusID:118562512>.
- [35] T. Kuwahara, “Introduction to cubesat attitude control system,” 2021.

- [36] S. V. Weston, C. D. Burkhard, J. M. Stupl, R. L. Ticknor, B. D. Yost, R. A. Austin, P. Galchenko, L. K. Newman, and L. S. Soto, “State-of-the-art small spacecraft technology,” Technical Publication NASA/TP–20250000142, NASA, Moffett Field and Greenbelt, Maryland, 2025. Millennium Engineering and Integration Services, Ames Research Center, Goddard Space Flight Center.
- [37] Q. Li, L. Ranyang, K. Ji, and W. Dai, “Kalman filter and its application,” pp. 74–77, 11 2015.
- [38] E. Lefferts, L. Markley, and M. SHUSTER, “Kalman filtering for spacecraft attitude estimation,” 01 1982.
- [39] R. Tedrake, “Linear quadratic regulators,” 2024. <https://underactuated.mit.edu/lqr.html>.
- [40] J. R. Wertz, D. F. Everett, and J. J. Puschell, *Space Mission Engineering: The New SMAD*. Hawthorne, California: Microcosm Press, 2011.
- [41] S. M. Spremo, A. R. Crocker, and T. L. Panontin, “Small spacecraft overview,” *NASA Ames Research Center*, 09 2019.
- [42] J. F. Wall and C. P. Bridges, “Solving the cubesat power paradox,” in *Small Satellite Conference*, 2016.
- [43] J. Puig-Suari, R. J. Twiggs, C. Turner, and W. A. Marshall, “Cubesat: The development and launch support infrastructure for eighteen cubesats,” in *AIAA/USU Conference on Small Satellites*, (Logan, Utah), AIAA/USU, 2009.
- [44] D. Kirk, *Optimal Control Theory: An Introduction*. Dover Books on Electrical Engineering Series, Dover Publications, 2004.
- [45] C.-Y. R. Huang, C.-Y. Lai, and K.-T. T. Cheng, “Fundamentals of algorithms,” in *Electronic Design Automation*.
- [46] P. Virtanen, R. Gommers, T. E. Oliphant, M. Haberland, T. Reddy, D. Cournapeau, E. Burovski, P. Peterson, W. Weckesser, J. Bright, S. J. van der Walt, M. Brett, J. Wilson, K. J. Millman, N. Mayorov, A. R. J. Nelson, E. Jones, R. Kern, E. Larson, C. J. Carey, Í. Polat, Y. Feng, E. W. Moore, J. VanderPlas, D. Laxalde, J. Perktold, R. Cimrman, I. Henriksen, E. A. Quintero, C. R. Harris, A. M. Archibald, A. H. Ribeiro, F. Pedregosa, P. van Mulbregt, and S. . Contributors, “SciPy 1.0: Fundamental Algorithms for Scientific Computing in Python,” *Nature Methods*, vol. 17, pp. 261–272, 2020.
- [47] M. L. Bynum, G. A. Hackebeil, W. E. Hart, C. D. Laird, B. L. Nicholson, J. D. Siirola, J.-P. Watson, and D. L. Woodruff, *Pyomo—optimization modeling in python*, vol. 67. Springer Science & Business Media, third ed., 2021.
- [48] W. E. Hart, J.-P. Watson, and D. L. Woodruff, “Pyomo: modeling and solving mathematical programs in python,” *Mathematical Programming Computation*, vol. 3, no. 3, pp. 219–260, 2011.
- [49] Ansys, “Systems tool kit,” 2024.
- [50] Valispace, “Valispace,” 2024. <https://www.valispace.com/>.

- [51] O. Montenbruck and E. Gill, *Satellite Orbits: Models, Methods and Applications*. Springer, 2012.
- [52] I. Gokasar and A. Akgun, “Multi-criteria decision making for sensor selection in satellite missions,” *Journal of Aerospace Information Systems*, vol. 13, no. 2, pp. 75–84, 2016.
- [53] M. A. Makrani and K. Salari, “Decision tree-based evaluation of satellite subsystem trade-offs,” *IEEE Access*, vol. 10, pp. 45810–45820, 2022.
- [54] A. Klesh and C. Kitts, “Design optimization of a nanosatellite attitude control subsystem,” *Acta Astronautica*, vol. 66, no. 7–8, pp. 1102–1110, 2010.
- [55] M. Muehlbauer and M. Prater, “Expert system for satellite adcs hardware selection,” in *Proceedings of the 32nd Annual AIAA/USU Conference on Small Satellites*, 2018.
- [56] D. Vallado and P. Crawford, “Sgp4 orbit determination,” 08 2008.
- [57] a.i solutions, “Freeflyer software,” 2023.
- [58] IIT Bombay Student Satellite Program, “Determination of sun vector using sun sensors.” https://www.aero.iitb.ac.in/satelliteWiki/index.php/Determination_of_sun_vector_using_sun_sensors, 2023.
- [59] P. Alken, E. Thébaud, C. D. Beggan, H. Amit, J. Aubert, J. Baerenzung, T. N. Bondar, W. J. Brown, S. Califf, A. Chambodut, A. Chulliat, G. A. Cox, C. C. Finlay, A. Fournier, N. Gillet, A. Grayver, M. D. Hammer, M. Holschneider, L. Huder, G. Hulot, T. Jager, C. Kloss, M. Korte, W. Kuang, A. Kuvshinov, B. Langlais, J.-M. L  ger, V. Lesur, P. W. Livermore, F. J. Lowes, S. Macmillan, W. Magnes, M. Manda, S. Marsal, J. Matzka, M. C. Metman, T. Minami, A. Morschhauser, J. E. Mound, M. Nair, S. Nakano, N. Olsen, F. J. Pav  n-Carrasco, V. G. Petrov, G. Ropp, M. Rother, T. J. Sabaka, S. Sanchez, D. Saturnino, N. R. Schnepf, X. Shen, C. Stolle, A. Tangborn, L. T  ffner-Clausen, H. Toh, J. M. Torta, J. Varner, F. Vervelidou, P. Vigneron, I. Wardinski, J. Wicht, A. Woods, Y. Yang, Z. Zeren, and B. Zhou, “International geomagnetic reference field: the thirteenth generation,” *Earth, Planets and Space*, vol. 73, p. 49, 12 2021.
- [60] N. C. for Enviromental Information, “International geomagnetic reference field (igrf),” 2021.
- [61] O. Egeland and J. T. Gravdahl, “Sun sensor for small satellite attitude determination,” *IFAC-PapersOnLine*, vol. 55, no. 29, pp. 264–269, 2022.
- [62] P. S. I. (PSI), “Miniature sun sensor for cubesats.” <https://www.ps-inc.com/sun-sensors>, 2023. Accessed: 2025-04-17.
- [63] J. Stewart, *Calculus: Early Transcendentals*. Cengage Learning, 8 ed., 2015.
- [64] M. Blanke and M. B. Larsen, *Satellite Dynamics and Control in a Quaternion Formulation*. Technical University of Denmark, Department of Electrical Engineering, 2nd ed., 2010.
- [65] D. Kolosa, “Implementing a linear quadratic spacecraft attitude control system,” master’s thesis, Western Michigan University, 2015.
- [66] M. L. Psiaki, “Magnetometer bias calibration via kalman filtering,” *Journal of Guidance, Control, and Dynamics*, vol. 24, no. 2, pp. 386–394, 2001.

- [67] A. Bacciotti, “Stability and control of linear systems,” 2019.
- [68] J. Narkiewicz, M. Sochacki, and B. Zakrzewski, “Generic model of a satellite attitude control system,” *Journal of Sensors*, 2020.
- [69] R. Bauer, “Attitude dynamics model of a cubesat with reaction wheels for an extended kalman filter,” in *Progress in Canadian Mechanical Engineering. Volume 4*, University of Prince Edward Island. Robertson Library, 6 2021.
- [70] P. Baranwal, K. Batta, and T. Kaushik, “Comparative study of classical and fuzzy pid attitude control system with extended kalman filter feedback for nanosatellites,” 10 2018.
- [71] NASA, “State of the art of small spacecraft technology,” 2023.
- [72] SatCatalog, “Cubesat acds subsystem components,” 2025. Accessed: 2025-04-19.
- [73] M. J. D. Powell, “An efficient method for finding the minimum of a function of several variables without calculating derivatives,” *The Computer Journal*, vol. 7, pp. 155–162, 01 1964.
- [74] S. S. Systems, “Giroscopio crh03 – 200,” 2024. <https://siliconsensing.com/product/crh03>.
- [75] S. S. Systems, “Giroscopio crh03 – 10,” 2024. <https://siliconsensing.com/product/crh03>.
- [76] N. System, “Giroscopio nsgy-001,” 2024. <https://satsearch.co/products/newspace-systems-nsgy-001-stellar-gyro>.
- [77] A. C. Space, “Magnetómetro mm200-1,” 2024. <https://www.aac-clyde.space/what-we-do/space-products-components/adcs/mm200>.
- [78] A. C. Space, “Magnetómetro mag-3-2,” 2024. <https://www.aac-clyde.space/what-we-do/space-products-components/adcs/mag-3>.
- [79] A. C. Space, “Magnetómetro mag-3,” 2024. <https://www.aac-clyde.space/what-we-do/space-products-components/adcs/mag-3>.
- [80] S. Micro, “Sensor sol css-01, css-02,” 2024. <https://www.satcatalog.com/component/css-01-02/>.
- [81] S. Micro, “Sensor sol mss-01,” 2024. <https://www.satcatalog.com/component/mss-01-02/>.
- [82] B. Space, “Sensor sol fss,” 2024. <https://satsearch.co/products/bradford-fine-sun-sensor>.
- [83] Z. Technik, “Magnetorquer mt0.5-1,” 2024. <https://www.satcatalog.com/component/mt05-1-po/>.
- [84] N. systems, “Magnetorquer nctr-m012,” 2024. <https://www.satcatalog.com/component/nctr-m012/>.
- [85] Z. Technik, “Magnetorquer mt15-1,” 2024. <https://www.satcatalog.com/component/mt15-1/>.
- [86] B. C. Technologies, “Rueda de reacción rwp500,” 2024. <https://www.satcatalog.com/component/rwp500/>.

- [87] B. C. Technologies, “Rueda de reacción rw1,” 2024.
<https://www.satcatalog.com/component/rw1/>.
- [88] B. C. Technologies, “Rueda de reacción rw4,” 2024.
<https://www.satcatalog.com/component/rw4/>.

A Carta Gantt

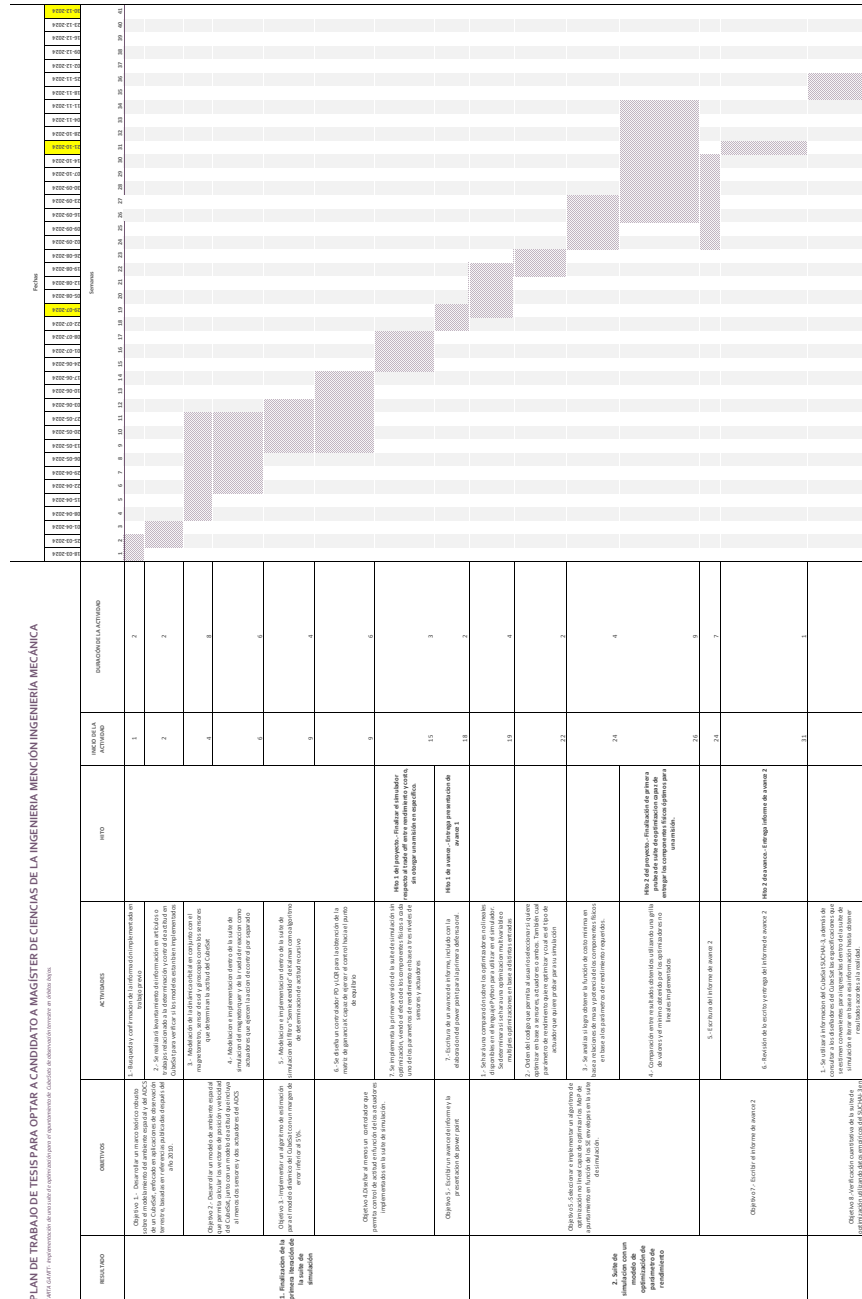


Figura A.1 Carta Gantt parte 1.

B Torques externos debido a las perturbaciones

Gradiente de gravedad: La órbita alrededor de la Tierra es posible gracias a la caída libre, que consiste en que la fuerza gravitacional hace caer al satélite constantemente sobre él. Sin embargo, debido a la velocidad horizontal que presenta la nave espacial en órbita, se mueve lo suficientemente rápido como para que su trayectoria curva coincida con la curvatura de la Tierra.

Si bien esta es la perturbación clave para que funcionen las órbitas en los satélites, también presenta consecuencias negativas para estos. Una de ellas es que la variación en el campo gravitacional de la Tierra a lo largo del CubeSat puede generar un gradiente gravitacional, causando un torque que se debe considerar al momento de analizar la dinámica de actitud del satélite. La ecuación que representa el torque se puede representar de la siguiente manera según [5], donde se utilizan los Momentos principales de Inercia (I) del satélite en cada uno de sus ejes:

$$T_g = \frac{3\mu}{2\|R\|^3} \left(\frac{R}{\|R\|} \times \begin{bmatrix} I_x & 0 & 0 \\ 0 & I_y & 0 \\ 0 & 0 & I_z \end{bmatrix} \frac{R}{\|R\|} \right) \approx 6\omega_{0,o}^2 \begin{bmatrix} (I_z - I_y)\delta q_0 \\ (I_z - I_x)\delta q_1 \\ 0 \end{bmatrix}$$

Donde:

- T_g : Es el torque de gravedad máximo.
- μ : Es la constante de gravedad de la Tierra ($3.988 \times 10^{14} \text{ [m}^3/\text{s}^2]$).
- R : Posición de la órbita desde el centro de la Tierra.
- I_x, I_y, I_z : Momentos de inercia en los ejes x, y, z respectivamente [$\text{kg} \cdot \text{m}^2$].

Torque debido al arrastre atmosférico: La fuerza de arrastre en órbitas bajas genera torques externos que pueden afectar la dinámica de actitud. Esta tiene una ecuación que se representa a continuación [?]:

$$T_a = F(c_{pa} - c_g) = FL$$

$$F = \frac{1}{2}\rho c_d AV^2$$

Donde:

- T_a : Torque ejercido por la fuerza de arrastre [Nm].
- L : Offset del centro de masa respecto al centro de presión [m].
- F : Fuerza de arrastre [N].
- ρ : Densidad atmosférica [kg/m^3].
- c_d : Coeficiente de arrastre [-].

- A : Superficie enfrentada a la fuerza de arrastre $[\text{m}^2]$.
- V : Velocidad del satélite $[\text{m/s}]$.

Presión debido a la radiación solar [5]: Esta depende en gran medida del tipo de superficie que está siendo iluminada. Una superficie puede ser transparente, absorbente o reflectante, pero la mayoría de las superficies son una combinación de las tres. Los reflectores se clasifican como difusos o especulares. En general, los conjuntos solares son absorbentes y el cuerpo de la nave espacial es un reflector. El peor caso de par de torsión debido a la radiación solar es:

$$T_{sp} = F(c_{ps} - cg)$$

$$F = \frac{F_s}{c} A_s (1 + q) \cos(i)$$

Donde:

- F_s : Constante solar ($1.367 [\text{W/m}^2]$).
- c : Velocidad de la luz ($3 \times 10^8 [\text{m/s}]$).
- A_s : Área de la superficie $[\text{m}^2]$.
- q : Factor de reflectancia (varía de 0 a 1, comúnmente se usa 0.6).
- i : Ángulo de incidencia del sol $[\circ]$.
- c_{ps} : Ubicación del centro de presión solar $[\text{mm}]$.
- cg : Centro de gravedad.

Campo magnético: El campo magnético ejerce un torque en el satélite, el cual se modela como se muestra a continuación, sabiendo que el campo magnético de la Tierra se puede aproximar tanto para una órbita polar como para una ecuatorial.

$$T_m = DB$$

$$B = \frac{2M}{R^3} (\text{Polar}); \quad B = \frac{M}{R^3} (\text{Ecuatorial})$$

Donde:

- D : Dipolo residual del satélite $[\text{A} \cdot \text{m}^2]$.
- B : Campo magnético de la Tierra $[\text{T}]$.
- M : Momento magnético de la Tierra ($7.96 \times 10^{15} [\text{T} \cdot \text{m}^3]$).
- R : Distancia desde el centro del dipolo (la Tierra) al satélite $[\text{m}]$.

C Cambios en los sistemas de referencia

De cuaternión a ángulos de Euler:

$$\text{Roll} = \arctan \left(\frac{2(q_3q_0 + q_1q_2)}{1 - 2(q_0^2 + q_1^2)} \right)$$

$$\text{Pitch} = \begin{cases} \arcsin(1.0), & \text{si } 2(q_3q_1 - q_2q_0) > 1.0 \\ \arcsin(-1.0), & \text{si } 2(q_3q_1 - q_2q_0) < -1.0 \\ \arcsin(2(q_3q_1 - q_2q_0)), & \text{en otro caso} \end{cases}$$

$$\text{Yaw} = \arctan \left(\frac{2(q_3q_2 + q_0q_1)}{1 - 2(q_1^2 + q_2^2)} \right)$$

De cuaterniones a DCM y viceversa:

$$\text{DCM} = \begin{bmatrix} q_3^2 + q_0^2 - q_2^2 - q_1^2 & 2(q_0q_1 + q_2q_3) & 2(q_0q_2 - q_1q_3) \\ 2(q_0q_1 - q_2q_3) & q_3^2 - q_0^2 + q_1^2 - q_2^2 & 2(q_1q_2 + q_0q_3) \\ 2(q_0q_2 + q_1q_3) & 2(q_1q_2 - q_0q_3) & q_3^2 - q_0^2 - q_1^2 + q_2^2 \end{bmatrix} = \begin{bmatrix} C_{11} & C_{12} & C_{13} \\ C_{21} & C_{22} & C_{23} \\ C_{31} & C_{32} & C_{33} \end{bmatrix}$$

$$q_0^2 = \frac{1}{4}(1 + C_{11} - C_{22} - C_{33})$$

$$q_1^2 = \frac{1}{4}(1 - C_{11} + C_{22} - C_{33})$$

$$q_2^2 = \frac{1}{4}(1 - C_{11} - C_{22} + C_{33})$$

$$q_3^2 = \frac{1}{4}(1 + C_{11} + C_{22} + C_{33})$$

D Control lineal Magnetorquer

Modelo no lineal del sistema

Se plantean primero las ecuaciones de la cinemática del cuaternión utilizando la componente escalar q_3 como dependiente de las componentes vectoriales, aprovechando que la norma del cuaternión es siempre igual a 1:

$$q_3 = \sqrt{1 - q_0^2 - q_1^2 - q_2^2}$$

Esto permite escribir las ecuaciones de estado del sistema dinámico como:

$$\dot{q}_0 = \frac{1}{2} \left(\omega_{2_co} q_1 - \omega_{1_co} q_2 + \omega_{0_co} \sqrt{1 - q_0^2 - q_1^2 - q_2^2} \right) \quad (41)$$

$$\dot{q}_1 = \frac{1}{2} \left(-\omega_{2_co} q_0 + \omega_{0_co} q_2 + \omega_{1_co} \sqrt{1 - q_0^2 - q_1^2 - q_2^2} \right) \quad (42)$$

$$\dot{q}_2 = \frac{1}{2} \left(\omega_{1_co} q_0 - \omega_{0_co} q_1 + \omega_{2_co} \sqrt{1 - q_0^2 - q_1^2 - q_2^2} \right) \quad (43)$$

Las ecuaciones de movimiento rotacional se expresan como:

$$\dot{\omega}_{0,ci} I_x = \omega_{1,ci} \omega_{2,ci} (I_y - I_z) + \tau_{x,ctrl} + \tau_{x,pert} \quad (44)$$

$$\dot{\omega}_{1,ci} I_y = \omega_{0,ci} \omega_{2,ci} (I_x - I_z) + \tau_{y,ctrl} + \tau_{y,pert} \quad (45)$$

$$\dot{\omega}_{2,ci} I_z = \omega_{0,ci} \omega_{1,ci} (I_x - I_y) + \tau_{z,ctrl} + \tau_{z,pert} \quad (46)$$

Modelo del actuador

El torque de control generado por los magnetorquers se modela como:

$$\tau_{ctrl} = \frac{\vec{m} \times \vec{b}}{\|\vec{b}\|} \times \vec{b} \quad (47)$$

Donde $\vec{b}$ representa el campo magnético en el marco del cuerpo, y $\vec{m}$ el momento dipolar generado por el magnetorquer. Este modelo se descompone en componentes:

$$\begin{aligned} \tau_{x,ctrl} &= \left(\frac{b_x m_z - b_z m_x}{\|\vec{b}\|} \right) b_z - \left(\frac{b_y m_x - b_x m_y}{\|\vec{b}\|} \right) b_y \\ \tau_{y,ctrl} &= - \left(\frac{b_z m_y - b_y m_z}{\|\vec{b}\|} \right) b_z + \left(\frac{b_y m_x - b_x m_y}{\|\vec{b}\|} \right) b_x \\ \tau_{z,ctrl} &= \left(\frac{b_z m_y - b_y m_z}{\|\vec{b}\|} \right) b_y - \left(\frac{b_x m_z - b_z m_x}{\|\vec{b}\|} \right) b_x \end{aligned}$$

Cambio de sistema de referencia

Para llevar todas las ecuaciones al sistema de referencia cuerpo vs órbita, se implementa el cambio de variables propuesto por Torczynski [14], utilizando el periodo orbital ω_{0_o} :

$$\vec{\omega}_{ci} = \vec{\omega}_{co} + \mathbf{Q}(q_0, q_1, q_2) \cdot \begin{bmatrix} \omega_{0_o} \\ 0 \\ 0 \end{bmatrix}$$

Donde $\mathbf{Q}(q)$ es la matriz de rotación asociada al cuaternión.

Ecuaciones modificadas con cambio de variable

Las ecuaciones dinámicas con el cambio de referencia aplicado se expresan como:

$$\begin{aligned} \dot{\omega}_{0_{co}} &= \left(\omega_{1_{co}} + \omega_{0_o} \left[2 \left(q_0 q_1 - q_2 \sqrt{1 - q_0^2 - q_1^2 - q_2^2} \right) \right] \right) \cdot \left(\omega_{2_{co}} + \omega_{0_o} \left[2 \left(q_0 q_2 + q_1 \sqrt{1 - q_0^2 - q_1^2 - q_2^2} \right) \right] \right) \cdot \left(\frac{I_x - I_z}{I_y} \right) \\ &\quad - \omega_{0_o} I_x [-4q_1 \dot{q}_1 - 4q_2 \dot{q}_2] + \frac{\tau_{x_{ctrl}}}{I_x} + \frac{\tau_{x_{pert}}}{I_x} \\ \dot{\omega}_{1_{co}} &= (\omega_{0_{co}} + \omega_{0_o}(1 - 2q_1^2 - 2q_2^2)) \cdot \left(\omega_{2_{co}} + \omega_{0_o} \left[2(q_0 q_2 + q_1 \sqrt{1 - q_0^2 - q_1^2 - q_2^2}) \right] \right) \cdot \frac{(I_x - I_z)}{I_y} \\ &\quad - \omega_{0_o} I_y \left[2(\dot{q}_0 q_1 + q_0 \dot{q}_1 - \dot{q}_2 \sqrt{1 - q_0^2 - q_1^2 - q_2^2} + q_2 \frac{(q_0 \dot{q}_0 + q_1 \dot{q}_1 + q_2 \dot{q}_2)}{\sqrt{1 - q_0^2 - q_1^2 - q_2^2}}) \right] + \frac{\tau_{y_{ctrl}}}{I_y} + \frac{\tau_{y_{pert}}}{I_y} \\ \dot{\omega}_{2_{co}} &= (\omega_{0_{co}} + \omega_{0_o}(1 - 2q_1^2 - 2q_2^2)) \cdot \left(\omega_{1_{co}} + \omega_{0_o} \left[2(q_0 q_1 - q_2 \sqrt{1 - q_0^2 - q_1^2 - q_2^2}) \right] \right) \cdot \frac{(I_x - I_z)}{I_z} \\ &\quad - \omega_{0_o} I_z \left[2(\dot{q}_0 q_2 + q_0 \dot{q}_2 + \dot{q}_1 \sqrt{1 - q_0^2 - q_1^2 - q_2^2} - q_1 \frac{(q_0 \dot{q}_0 + q_1 \dot{q}_1 + q_2 \dot{q}_2)}{\sqrt{1 - q_0^2 - q_1^2 - q_2^2}}) \right] + \frac{\tau_{z_{ctrl}}}{I_z} + \frac{\tau_{z_{pert}}}{I_z} \end{aligned}$$

Linealización del sistema

Evalando en el punto de equilibrio $x = \vec{0}$ y $u = \vec{0}$, se obtiene la matriz Jacobiana del sistema:

$$A = \begin{bmatrix} 0 & 0 & 0 & 0.5 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.5 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.5 \\ 6\omega_{(0,o)}^2(I_z - I_y) & 0 & 0 & 0 & 0 & 0 \\ 0 & 6\omega_{(0,o)}^2(I_z - I_x) & 0 & 0 & 0 & \frac{\omega_{(0,o)}(I_x - I_y)}{I_z} - \omega_{(0,o)} \\ 0 & 0 & 0 & 0 & \frac{\omega_{(0,o)}(I_x - I_z)}{I_y} + \omega_{(0,o)} & 0 \end{bmatrix}$$

Y la matriz de entrada $\bar{B}$ considerando los torques generados por los magnetorquers:

$$\bar{B} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ \frac{-b_z^2 - b_y^2}{\|b\|I_x} & \frac{b_x b_y}{\|b\|I_y} & \frac{b_x b_z}{\|b\|I_z} \\ \frac{b_x b_y}{\|b\|I_x} & \frac{-b_z^2 - b_x^2}{\|b\|I_y} & \frac{b_y b_z}{\|b\|I_z} \\ \frac{b_x b_z}{\|b\|I_x} & \frac{b_y b_z}{\|b\|I_y} & \frac{-b_y^2 - b_x^2}{\|b\|I_z} \end{bmatrix}$$

E Dinámica de seis estados del sistema con ruedas de reacción

En este anexo se presenta una formulación reducida de seis estados, en la cual las ruedas de reacción se representan como generadoras de torque de control aplicado al satélite. No obstante, el modelo completo incluye las velocidades angulares de las ruedas y su momento angular acoplado a la dinámica del cuerpo, lo cual permite capturar los efectos cruzados entre ejes.

Cinemática del cuaternión

Se plantea la cinemática utilizando las tres primeras componentes del cuaternión como estados, asumiendo que la norma del cuaternión es siempre unidad. De este modo, se define la componente escalar como dependiente de las otras:

$$q_3 = \sqrt{1 - q_0^2 - q_1^2 - q_2^2}$$

Las ecuaciones cinemáticas del cuaternión quedan como:

$$\dot{q}_0 = \frac{1}{2} (\omega_{2co} q_1 - \omega_{1co} q_2 + \omega_{0co} q_3) \quad (48)$$

$$\dot{q}_1 = \frac{1}{2} (-\omega_{2co} q_0 + \omega_{0co} q_2 + \omega_{1co} q_3) \quad (49)$$

$$\dot{q}_2 = \frac{1}{2} (\omega_{1co} q_0 - \omega_{0co} q_1 + \omega_{2co} q_3) \quad (50)$$

Dinámica del cuerpo rígido

La dinámica del satélite se describe con las siguientes ecuaciones, considerando que los torques de control provienen directamente de las ruedas de reacción:

$$\dot{\omega}_{0ci} J_x = \omega_{1ci} \omega_{2ci} (J_y - J_z) + \tau_{x,ctrl} + \tau_{x,pert} \quad (51)$$

$$\dot{\omega}_{1ci} J_y = \omega_{0ci} \omega_{2ci} (J_x - J_z) + \tau_{y,ctrl} + \tau_{y,pert} \quad (52)$$

$$\dot{\omega}_{2ci} J_z = \omega_{0ci} \omega_{1ci} (J_x - J_y) + \tau_{z,ctrl} + \tau_{z,pert} \quad (53)$$

Es importante destacar que las ecuaciones generales son con 9 estados, las cuales se representan a continuación, y son difíciles de aplicar al también estimar el vector estado con el filtro de kalman, y se requiere más variables a incluir para lograrlo, como se muestra en [69].

$$\dot{\omega}_{0,ci}(J_x - I_{s0}) = \omega_{1,ci}\omega_{2,ci}(J_y - J_z) + \omega_{1s,ci}\omega_{2,ci}I_{s1} - \omega_{1,ci}\omega_{2s,ci}I_{s2} + \tau_{x,ctrl} + \tau_{x,pert} \quad (54)$$

$$\dot{\omega}_{1,ci}(J_y - I_{s1}) = \omega_{0,ci}\omega_{2,ci}(J_x - J_z) + \omega_{0s,ci}\omega_{2,ci}I_{s0} - \omega_{0,ci}\omega_{2s,ci}I_{s2} + \tau_{y,ctrl} + \tau_{y,pert} \quad (55)$$

$$\dot{\omega}_{2,ci}(J_z - I_{s2}) = \omega_{0,ci}\omega_{1,ci}(J_x - J_y) + \omega_{0s,ci}\omega_{1,ci}I_{s0} - \omega_{0,ci}\omega_{1s,ci}I_{s1} + \tau_{z,ctrl} + \tau_{z,pert} \quad (56)$$

$$\dot{\omega}_{0s,ci}I_{s0} = -I_{s0}\dot{\omega}_{0,ci} + \tau_{s0,ctrl} + \tau_{s0,pert} \quad (57)$$

$$\dot{\omega}_{1s,ci}I_{s1} = -I_{s1}\dot{\omega}_{1,ci} + \tau_{s1,ctrl} + \tau_{s1,pert} \quad (58)$$

$$\dot{\omega}_{2s,ci}I_{s2} = -I_{s2}\dot{\omega}_{2,ci} + \tau_{s2,ctrl} + \tau_{s2,pert} \quad (59)$$

Momento de inercia efectivo

Se incorpora el segundo momento de inercia total del sistema, considerando la masa m y geometría (o distancia entre centros de masa b) de las ruedas de reacción mediante el teorema de los ejes paralelos:

$$J_x = I_x + I_{s0} + I_{s1} + m_1 b_1^2 + I_{s2} + m_2 b_2^2$$

$$J_y = I_y + I_{s1} + I_{s0} + m_0 b_0^2 + I_{s2} + m_2 b_2^2$$

$$J_z = I_z + I_{s2} + I_{s0} + m_0 b_0^2 + I_{s1} + m_1 b_1^2$$

Cambio de referencia inercial a orbital

Para que todas las ecuaciones estén en el marco de referencia control-órbita, se aplica la transformación de velocidades angulares propuesta por Torczynski [14], considerando la velocidad angular de la órbita ω_{0_o} :

$$\vec{\omega}_{ci} = \vec{\omega}_{co} + R(q) \cdot \begin{bmatrix} \omega_{0_o} \\ 0 \\ 0 \end{bmatrix}$$

donde $R(q)$ es la matriz de rotación derivada del cuaternión (definida completamente en la ecuación del cuerpo principal).

Ecuaciones dinámicas transformadas

Aplicando la transformación anterior, las ecuaciones de dinámica angular se expresan en términos de las velocidades angulares relativas $\vec{\omega}_{co}$ y los cuaterniones:

$$\begin{aligned}\dot{\omega}_{0_{co}} = & (\omega_{1_{co}} + \omega_{0_o} \cdot 2(q_0 q_1 - q_2 q_3)) \cdot (\omega_{2_{co}} + \omega_{0_o} \cdot 2(q_0 q_2 + q_1 q_3)) \cdot \frac{J_y - J_z}{J_x} \\ & - \omega_{0_o} J_x \cdot 4(q_1 \dot{q}_1 + q_2 \dot{q}_2) + \frac{\tau_{x,ctrl}}{J_x} + \frac{\tau_{x,pert}}{J_x}\end{aligned}\quad (60)$$

$$\begin{aligned}\dot{\omega}_{1_{co}} = & (\omega_{0_{co}} + \omega_{0_o}(1 - 2q_1^2 - 2q_2^2)) \cdot (\omega_{2_{co}} + \omega_{0_o} \cdot 2(q_0 q_2 + q_1 q_3)) \cdot \frac{J_x - J_z}{J_y} \\ & - \omega_{0_o} J_y \cdot 2 \left(\dot{q}_0 q_1 + q_0 \dot{q}_1 - \dot{q}_2 q_3 + q_2 \frac{q_0 \dot{q}_0 + q_1 \dot{q}_1 + q_2 \dot{q}_2}{q_3} \right) \\ & + \frac{\tau_{y,ctrl}}{J_y} + \frac{\tau_{y,pert}}{J_y}\end{aligned}\quad (61)$$

$$\begin{aligned}\dot{\omega}_{2_{co}} = & (\omega_{0_{co}} + \omega_{0_o}(1 - 2q_1^2 - 2q_2^2)) \cdot (\omega_{1_{co}} + \omega_{0_o} \cdot 2(q_0 q_1 - q_2 q_3)) \cdot \frac{J_x - J_y}{J_z} \\ & - \omega_{0_o} J_z \cdot 2 \left(\dot{q}_0 q_2 + q_0 \dot{q}_2 + \dot{q}_1 q_3 - q_1 \frac{q_0 \dot{q}_0 + q_1 \dot{q}_1 + q_2 \dot{q}_2}{q_3} \right) \\ & + \frac{\tau_{z,ctrl}}{J_z} + \frac{\tau_{z,pert}}{J_z}\end{aligned}\quad (62)$$

Modelo linealizado

A partir del modelo no lineal, se realiza la linealización alrededor del equilibrio para obtener las matrices A y B , útiles tanto para el diseño del controlador lineal como para la estimación con el Filtro de Kalman Extendido (EKF):

$$x = [q_0 \quad q_1 \quad q_2 \quad \omega_{0_{co}} \quad \omega_{1_{co}} \quad \omega_{2_{co}}]^T, \quad u = [\tau_x \quad \tau_y \quad \tau_z]^T$$

$$A = \begin{bmatrix} 0 & 0 & 0 & 0.5 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0.5 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.5 \\ a_{40} & 0 & 0 & 0 & 0 & 0 \\ 0 & a_{51} & 0 & 0 & 0 & a_{56} \\ 0 & 0 & 0 & 0 & a_{65} & 0 \end{bmatrix}$$

donde:

$$a_{40} = 6\omega_{0_o}^2 \cdot \frac{J_z - J_y}{J_x}, \quad a_{51} = 6\omega_{0_o}^2 \cdot \frac{J_z - J_x}{J_y}, \quad a_{56} = \frac{\omega_{0_o}(J_x - J_y)}{J_z} - \omega_{0_o}, \quad a_{65} = \frac{\omega_{0_o}(J_x - J_z)}{J_y} + \omega_{0_o}$$

$$B = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ \frac{1}{J_x} & 0 & 0 \\ 0 & \frac{1}{J_y} & 0 \\ 0 & 0 & \frac{1}{J_z} \end{bmatrix}$$

F Componentes utilizados para Capitulo 5

F.1 Giroscopios

Tabla F.1 Tabla de giroscopios con sus respectivas características de costo.

Nivel	Modelo	Masa [kg]	Potencia [W]	Tamaño [mm]
Bajo	CRH03 – 200 [74] (Silicon Sensing Systems)	0.045	0.15	47 x 33.5 x 25.4
Medio	CRH03 – 010 [75] (Silicon Sensing Systems)	0.045	0.15	47 x 33.5 x 25.4
Alto	NSGY-001 [76] (NewSpace System)	0.055	0.2	37.0 x 35.5 x 49.0

Tabla F.2 Tabla de giroscopios con sus respectivas características de performance.

Nivel	Modelo	Estabilidad Bias [°/hr]	Ruido/ σ [°/s]
Bajo	CRH03 – 200 (Silicon Sensing Systems)	0.05	0.12
Medio	CRH03 – 010 (Silicon Sensing Systems)	0.03	0.050
Alto	NSGY-001 (NewSpace System)	-	0.033

F.2 Magnetómetros

Tabla F.3 Tabla de magnetómetros con sus respectivas características de costo.

Nivel	Modelo	Masa [kg]	Potencia [W]	Tamaño [mm]
Bajo	MM200-1 [77] (AAC Clyde Space)	0.012	0.1	33 x 20 x 113
Medio	MAG-3-2 [78] (AAC Clyde Space)	0.15	0.1	3.51 x 3.23 x 8.26
Alto	MAG-3 [79] (AAC Clyde Space)	0.1	0.735	3.51 x 3.23 x 8.26

Tabla F.4 Tabla de magnetómetros con sus respectivas características de performance.

Nivel	Modelo	Ruido máximo [nT/Hz]
Bajo	MM200-1 (AAC Clyde Space)	1.18
Medio	MAG-3-2 (AAC Clyde Space)	0.1
Alto	MAG-3 (AAC Clyde Space)	0.012

F.3 Sensores de sol

Tabla F.5 Tabla de sensores de sol con sus respectivas características de costo.

Nivel	Modelo	Masa [kg]	Potencia [W]	Tamaño [mm]
Bajo	CSS-01, CSS-02 [80] (Space Micro)	0.010	0	12.7 ϕ x 9
Medio	MSS-01 [81] (Space Micro)	0.036	0	34.9 ϕ x 24.3
Alto	FSS [82] (Bradford Space)	0.375	0.25	108 x 108 x 52.5

Tabla F.6 Tabla de sensores de sol con sus respectivas características de performance.

Nivel	Modelo	Exactitud [°]/ Desviación Estándar [°]
Bajo	CSS-01, CSS-02 (Space Micro)	5/0.833
Medio	MSS-01 (Space Micro)	1/0.167
Alto	FSS (Bradford Space)	0.3/0.05

F.4 Magnetorquers

Tabla F.7 Tabla de magnetorquers con sus respectivas características de costo.

Nivel	Modelo	Masa [kg]	Potencia [W]	Tamaño [mm]
Bajo	MT0.5-1 [83] (ZARM Technik)	0.03	0.275	9ø x 100
Medio	NCTR-M012 [84] (NewSpace systems)	0.053	0.8	94 x 15 x 13
Alto	MT15-1 [85] (ZARM Technik)	0.43	1.11	17ø x 329.5

Tabla F.8 Tabla de magnetorquers con sus respectivas características de performance.

Nivel	Modelo	Peak dipole [Am^2]
Bajo	MT0.5-1 (ZARM Technik)	0.5
Medio	NCTR-M012 (NewSpace systems)	1.19
Alto	MT15-1 (ZARM Technik)	15

F.5 Ruedas de reacción

Tabla F.9 Tabla de ruedas de reacción con sus respectivas características de costo.

Nivel	Modelo	Masa [kg]	Potencia [W]	Tamaño [mm]
Bajo	RWP500 [86] (Blue Canyon Technologies)	0.75	6	110 x 110 x 38
Medio	RW1 [87] (Blue Canyon Technologies)	0.95	9	110 x 110 x 38
Alto	RW4 [88] (Blue Canyon Technologies)	3.2	10	108 x 108 x 52.5

Tabla F.10 Tabla de ruedas de reacción con sus respectivas características de performance.

Nivel	Modelo	Peak torque $[Nm]$/ Capacidad de momento $[Nms]$
Bajo	RWP500 (Blue Canyon Technologies)	0.025/0.50
Medio	RW1 (Blue Canyon Technologies)	0.1/1.0
Alto	RW4 (Blue Canyon Technologies)	0.25/4.0

G Implementación de los resultados de los MoP de apuntamiento

Para mostrar el paso a paso de la cuantificación de los MoP de apuntamiento, se utilizará como ejemplo la simulación del SUCHAI-3 con los datos utilizados en el Capítulo 4, con los componentes físicos de nivel bajo en magnetorquer.

Jitter

Para la cuantificación de este MoP de apuntamiento, en primera instancia se aplica el filtro pasa alto con una frecuencia tope de 10 Hz, mostrando solo las señales mayores a dicho valor. Con esto se observa la Figura G.1 asentándose la señal siempre en el cero con el ruido de las vibraciones del sistema.

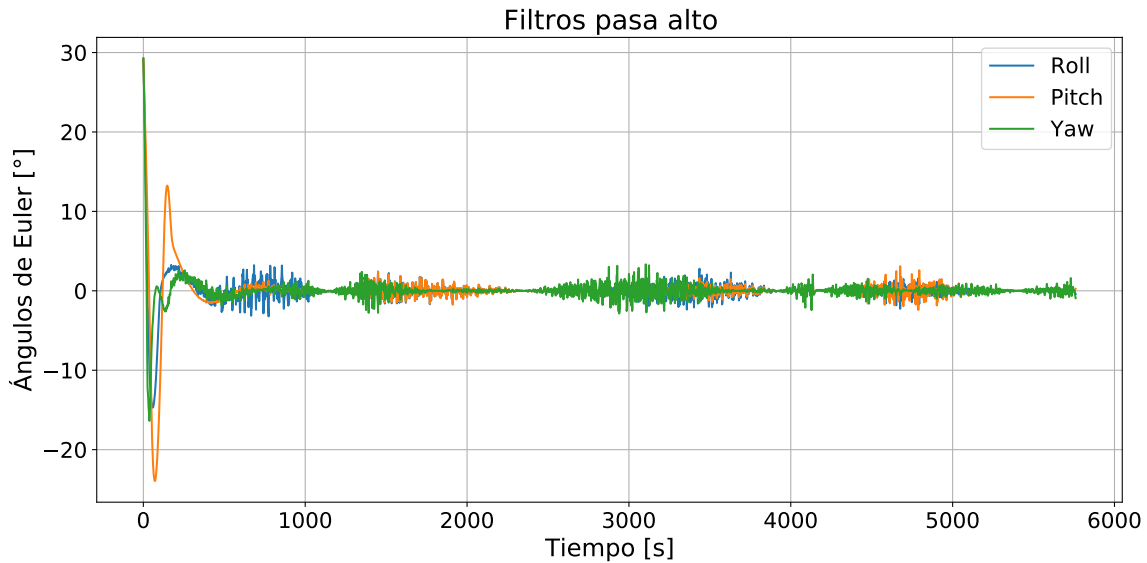


Figura G.1 Filtro pasa alto de 10 Hz en la respuesta al sistema de sensores y actuadores de nivel bajo.

Con esto, se obtienen las frecuencias y sus densidades espectro potencias aproximadas mediante la función `welch` de `scipy.signal` disponible en Python. Posteriormente, se selecciona todo el ancho de banda, para obtener el PSD que representa el jitter del sistema. Esto se observa en los ángulos de Euler en las Figura G.2, Figura G.3 y Figura G.4 para cada uno, presentado mayores valores a menores frecuencias. Se observa en el área pintada la integración dentro del ancho de banda realizada.

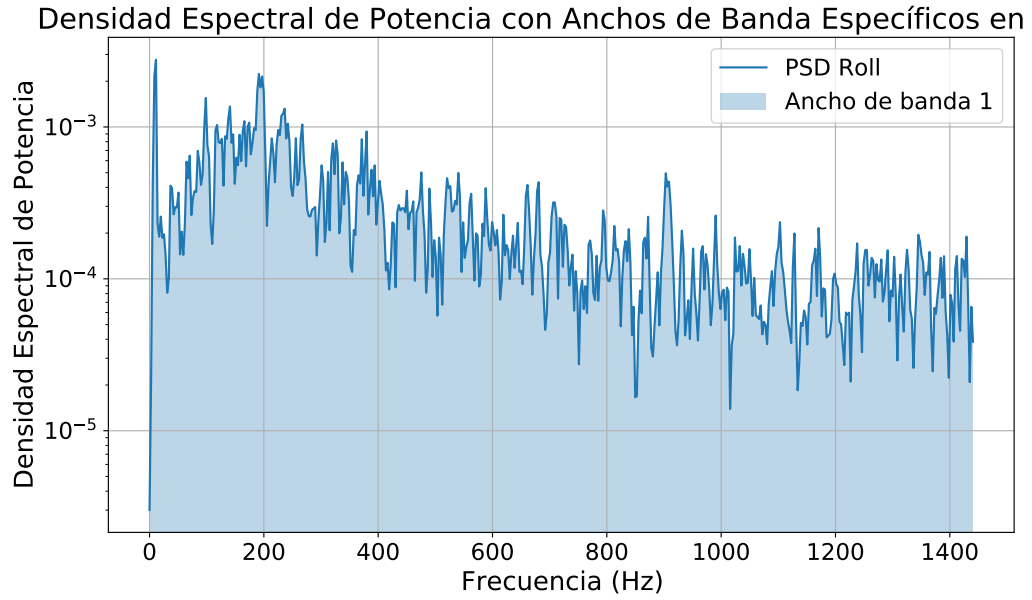


Figura G.2 Densidad espectro potencia en ancho de banda seleccionada para Roll.

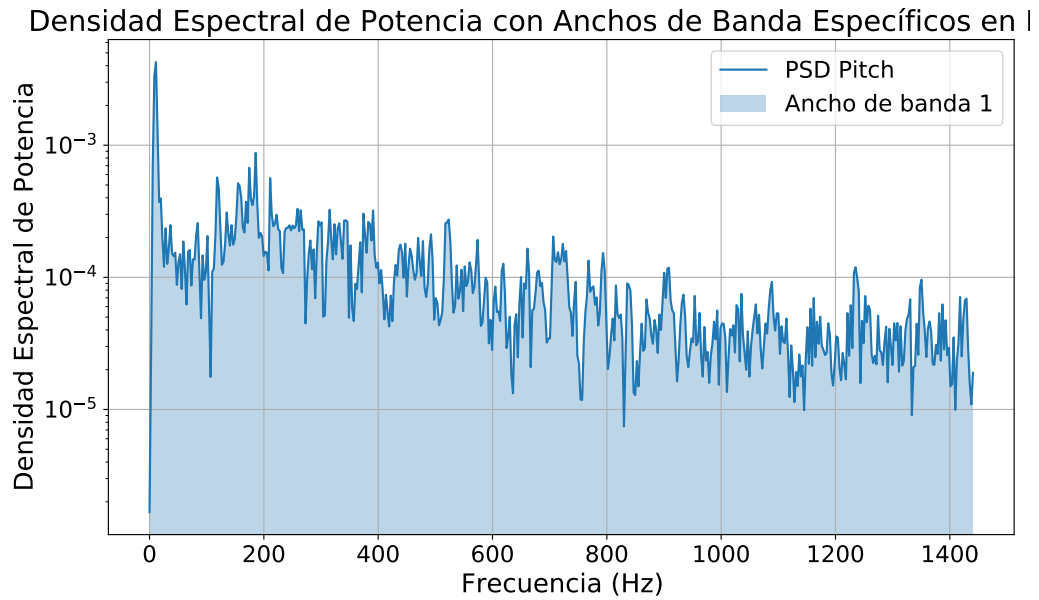


Figura G.3 Densidad espectro potencia en ancho de banda seleccionada para Pitch.

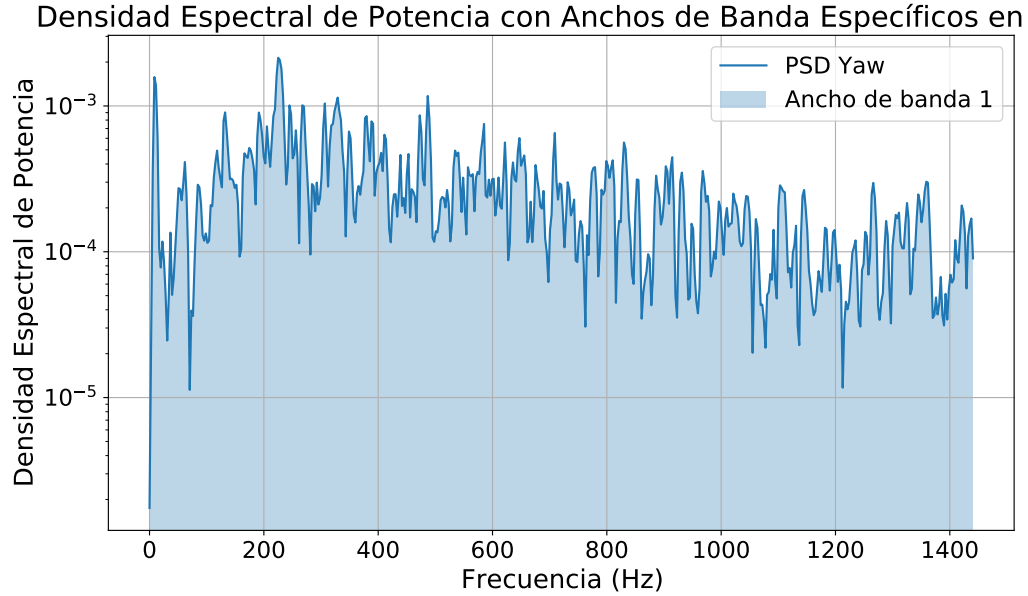


Figura G.4 Densidad espectro potencia en ancho de banda seleccionada para Yaw.

Agilidad

Por otro lado, para la cuantificación de la agilidad, se utilizan los valores de los ángulos de Euler reales obtenidos por el modelo. Esto se hace con el objetivo de conocer lo más preciso posible el tiempo de asentamiento con una banda cercana al objetivo (entre -2.5 y 2.5 grados de los ángulos de Euler) sin las perturbaciones causadas por el ruido de los sensores.

En las Figura G.5, Figura G.6 y Figura G.7 se muestran los ángulos de Euler Roll, Pitch y Yaw reales, en conjunto con una gráfica acercada en la zona de asentamiento, mostrando además los límites superior e inferior dadas por la banda de asentamiento. Una vez que la orientación del satélite está presente dentro de la banda de asentamiento, el tiempo inicial para el ingreso sin retirarse es el tiempo de asentamiento con el cual se cuantifica la agilidad. En el caso de la optimización, se utiliza el mayor valor de los ángulos de Euler antes de escaparse de la banda de asentamiento como el tiempo de asentamiento.

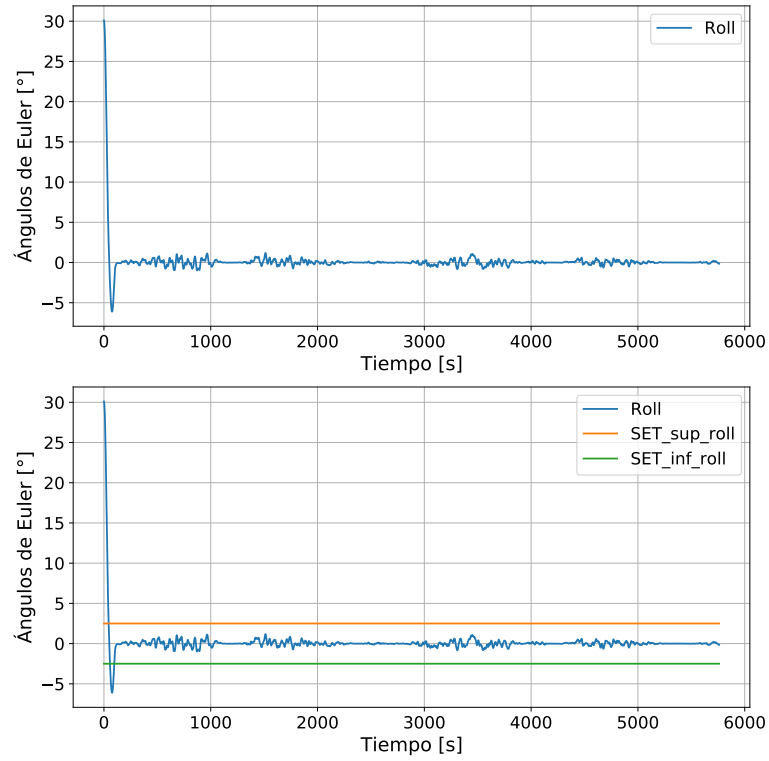


Figura G.5 Banda de asentamiento Roll en modelo dinámico.

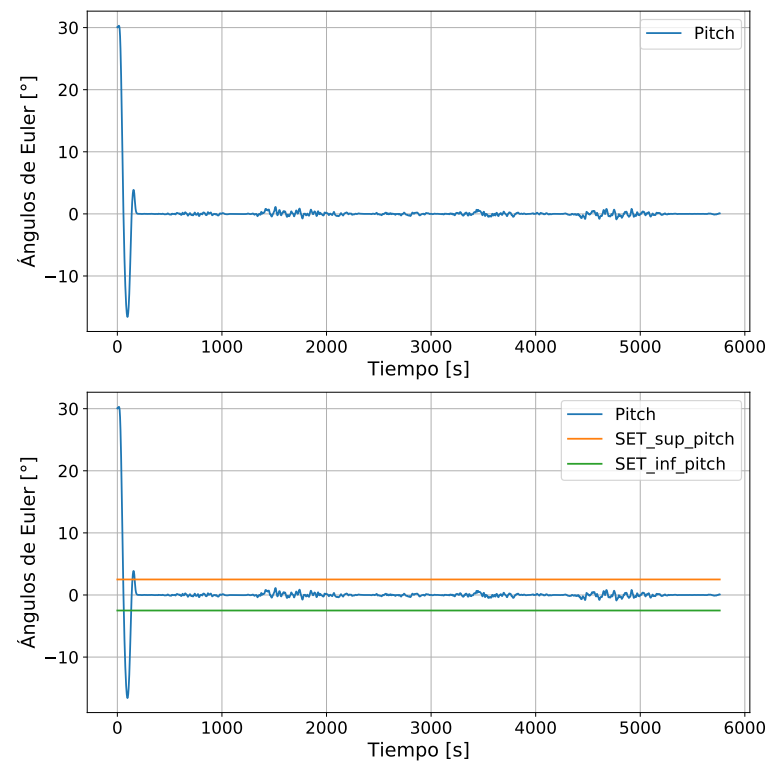


Figura G.6 Banda de asentamiento Pitch en modelo dinámico.

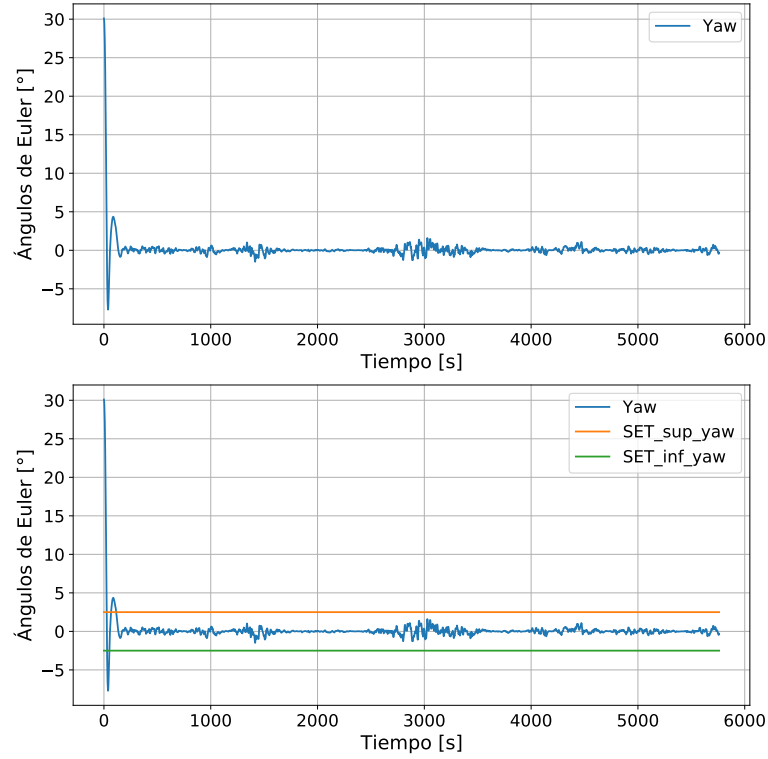


Figura G.7 Banda de asentamiento Yaw en modelo dinámico.

Exactitud de apuntamiento

Para evaluar la exactitud de apuntamiento en la norma de los ángulos de Euler, se utilizan los valores de los ángulos de Euler estimados, obtenidos tras el tiempo de asentamiento. A continuación, se calcula la media dentro de ese periodo de tiempo para obtener el resultado a utilizar para evaluar rendimiento y optimización.

H Código en Python

El código con el informe en LaTeX y toda la suite de simulación y optimización se encuentra en el siguiente link de GitHub:

https://github.com/mtacul/thesis_postgrade