



UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

CENTRO DE FÍSICA APLICADA Y TECNOLOGÍA AVANZADA

IMPLEMENTACIÓN DE UN MODELO VIRTUAL DE
UN NANOSATÉLITE MEDIANTE TÉCNICAS SIL
COMO MEDIO DE VERIFICACIÓN DE SISTEMAS DE
CONTROL DE ORIENTACIÓN MAGNÉTICOS

T E S I S

QUE PARA OBTENER EL TÍTULO DE:

Licenciado en Tecnología

PRESENTA:

Héctor Antonio Belmont Hernández

TUTOR:

Dr. Rafael Guadalupe Chávez Moreno



Juriquilla, Querétaro, México, 2021



Universidad Nacional
Autónoma de México

Dirección General de Bibliotecas de la UNAM

Biblioteca Central

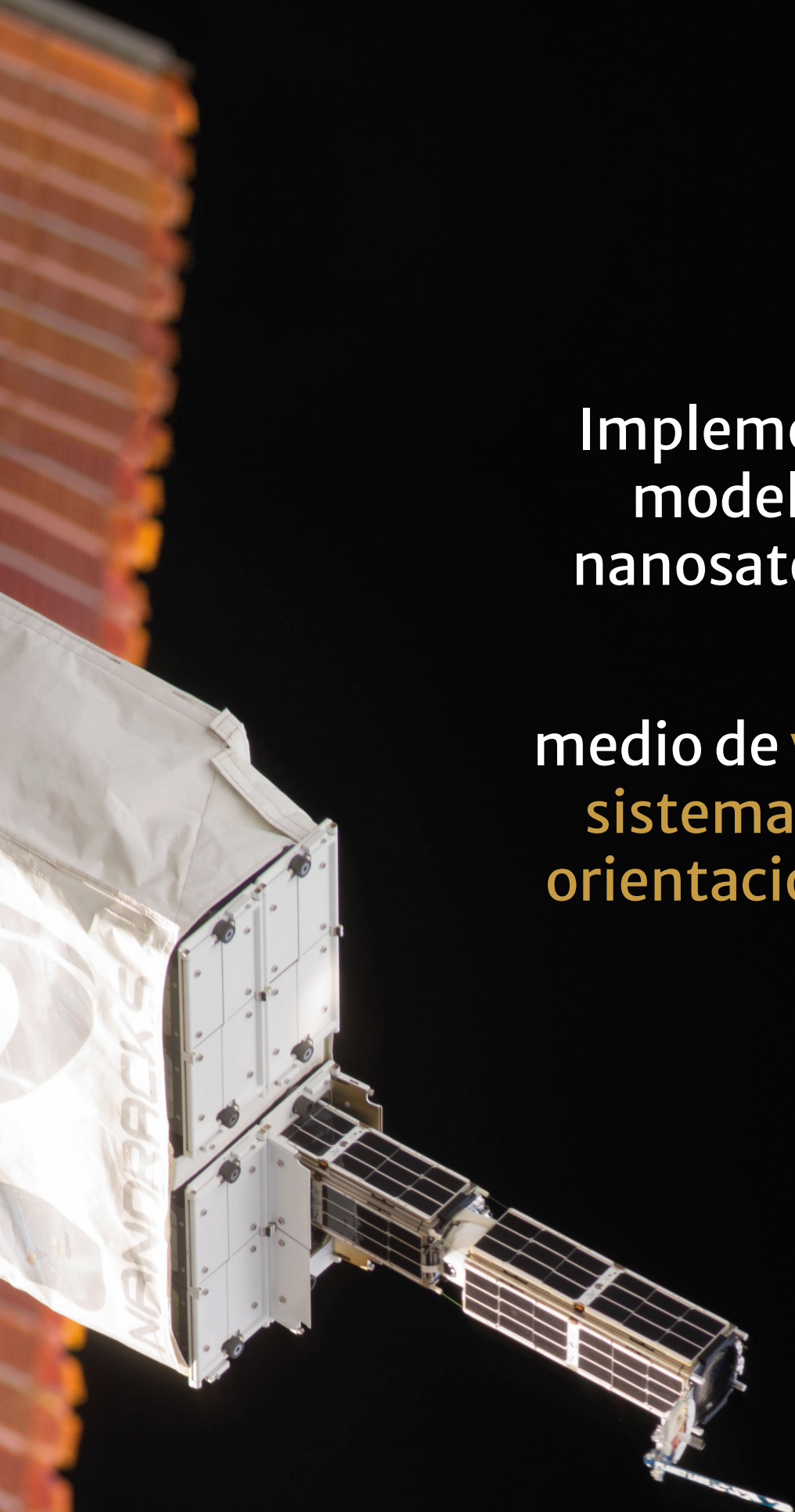


UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.



Implementación de un
modelo virtual de un
nanosatélite mediante
técnicas SIL
como
medio de **verificación de
sistemas de control de
orientación magnéticos**

Héctor Antonio Belmont Hernández
bajo dirección del Dr. Rafael Chávez Moreno

A mi familia

Agradecimientos

A SEDESU a través del programa PEDETI, así como al proyecto PAPIIT TA102220 por el apoyo para el desarrollo del proyecto.

Al equipo ADCS de K'oto por su colaboración y retroalimentación directa en este trabajo, así como al equipo completo de K'oto por la oportunidad de participar en el fortalecimiento del área espacial nacional.

A mis asesores de estancia y profesores de la carrera por enseñarme el significado de la investigación científica.

Resumen

En el diseño de un sistema espacial las pruebas de validación y verificación toman una importancia adicional debido a factores como la incapacidad de interactuar físicamente con el sistema después del lanzamiento. Existen herramientas comerciales enfocadas hacia la parte de pruebas, pero factores como la necesidad de licencias, complejidad de uso y falta de transparencia en los modelos suelen traer contratiempos adicionales durante el desarrollo.

En este trabajo se presenta la elaboración de una herramienta de simulación que aborda estos problemas, enfocada hacia la verificación del subsistema de determinación y control de orientación de un nanosatélite. Para ello se hizo uso del enfoque de diseño basado en modelos, en donde a cada elemento involucrado con el subsistema como componentes electrónicos, algoritmos o factores ambientales; le corresponde un modelo, los cuales pueden interactuar entre sí para simular el subsistema completamente. La referencia para el diseño de la simulación fue un nanosatélite de control de orientación magnético, en donde es de interés conocer el desempeño de los actuadores.

Como caso de estudio se usó a K'oto, un satélite mexicano en estado de desarrollo. Los resultados de la herramienta permitieron identificar factores de riesgo, puesto que se determinó que revisiones son necesarias en el diseño del subsistema de control de orientación para poder satisfacer los requerimientos.

Índice general

Índice de figuras	XI
Índice de tablas	XIII
1. Introducción	1
1.1. Contexto	1
1.1.1. Los satélites y la sustentabilidad	4
1.1.2. El estándar CubeSat	5
1.1.3. El área espacial en México	6
1.1.3.1. Historia del área espacial en México	7
1.1.3.2. Misiones satelitales desarrolladas en México	9
1.1.3.3. El papel de la industria aeroespacial	10
1.1.3.4. Querétaro: Su estado actual y potencial	11
1.2. Planteamiento del problema	12
1.2.1. El software en las misiones espaciales	13
1.2.2. La importancia de las simulaciones	14
1.2.3. La necesidad de herramientas de verificación en el desarrollo de un ADCS	15
1.3. Objetivos	16
1.3.1. Objetivos particulares	16
1.4. Hipótesis	16
1.5. Antecedentes	16
1.5.1. Diseño basado en modelos	18
1.5.1.1. Modelo en V	19
1.5.1.2. Técnicas de verificación en los ADCS	21
1.5.2. Software de vuelo	23
1.5.3. Software de análisis de misiones	26
1.5.3.1. Software existente para el análisis de misiones	26
1.5.3.2. Software de verificación desarrollado por universidades	28
1.5.3.3. Funcionalidades relacionadas en Simulink	29
1.5.4. Gemelos digitales y hermanos digitales	29
1.6. Justificación	30
2. Marco teórico	33
2.1. Los nanosatélites y sus características	33
2.1.1. Subsistemas de un satélite	34
2.2. Subsistema de determinación y control de orientación	35
2.2.1. Sensores	37
2.2.1.1. Sensores solares	38

2.2.1.2.	Magnetómetros	38
2.2.1.3.	Giroscopios	39
2.2.1.4.	Sensores de horizonte	39
2.2.1.5.	Sensores de estrellas	40
2.2.2.	Sistemas de control de orientación	40
2.2.2.1.	Pasivos	40
2.2.2.2.	Activos	41
2.3.	Órbitas y marcos de referencia	42
2.3.1.	Órbitas	42
2.3.1.1.	Leyes de Kepler	43
2.3.1.2.	Elementos keplerianos	43
2.3.2.	Marcos de referencia	44
2.3.2.1.	Marco de referencia fijo al cuerpo	45
2.3.2.2.	Marco de referencia fijo al sensor	45
2.3.2.3.	Marco de referencia Roll–Pitch–Yaw (RPY)	46
2.3.2.4.	LVLH (Local Vertical, Local Horizontal):	46
2.3.2.5.	Earth-Centered Inertial (ECI):	47
2.3.2.6.	Earth-Centered Earth Fixed (ECEF):	47
2.4.	Representaciones de orientación	48
2.4.1.	Matrices de rotación	48
2.4.2.	Ángulos de Euler	50
2.4.3.	Cuaterniones	50
2.4.3.1.	Propiedades de los cuaterniones	51
2.5.	Dinámica y cinemática del satélite	53
2.5.1.	Cinemática	54
2.5.2.	Dinámica	54
3.	Metodología	57
3.1.	Organización del modelo virtual	57
3.2.	Configuración del nanosatélite simulado	59
3.2.1.	Evaluación de requerimientos	60
3.3.	Obtención de parámetros mediante STK	60
3.3.1.	Flujo de obtención de datos	61
3.4.	Subsistema de visualización	62
4.	Modelado	65
4.1.	Perturbaciones del ambiente espacial	65
4.1.1.	Par por gradiente de gravedad	66
4.1.2.	Par por presión de radiación solar	66
4.1.3.	Par aerodinámico	67
4.2.	Modelos atmosféricos	68
4.2.1.	Ejemplos de modelos de densidad atmosférica	70
4.2.1.1.	Modelos de Jacchia	72
4.2.1.2.	Modelos MSIS	74
4.3.	Algoritmos de determinación de orientación	74

4.3.1.	Generalidades	74
4.3.1.1.	El problema de la determinación de orientación . . .	75
4.3.2.	Modelos inerciales de referencia	76
4.3.2.1.	Modelo inercial del sol	76
4.3.3.	Algoritmos de determinación estáticos	78
4.3.3.1.	TRIAD	78
4.3.3.2.	El problema de Wahba	80
4.3.3.3.	Davenport Q	81
4.3.3.4.	QUEST	84
4.4.	Estimación de estado mediante filtros de Kalman	86
4.4.1.	Sistemas lineales	86
4.4.2.	Filtro de Kalman lineal	87
4.4.3.	Filtro de Kalman extendido	87
4.4.4.	Filtros de Kalman en satélites	88
4.4.5.	Implementación del filtro de Kalman extendido en el modelo virtual	89
4.5.	Algoritmos de control de orientación	91
4.5.1.	B-dot	91
4.5.2.	Apuntamiento a nadir	92
4.6.	Componentes electrónicos	93
4.6.1.	Sensores	94
4.6.1.1.	Sensores solares	94
4.6.1.2.	Magnetómetro	96
4.6.1.3.	Giroscopio	98
4.6.1.4.	GPS	98
4.6.2.	Actuadores: Magnetorcas	100
5.	Resultados	103
5.1.	Características del modelo virtual	103
5.2.	Caso de estudio: K'oto	105
5.2.1.	Sistema de determinación y control de orientación de K'oto .	105
5.3.	Simulaciones	105
5.3.1.	Condiciones iniciales de la simulación	106
5.3.2.	Elección de modelo de densidad	108
5.3.3.	Cinemática y dinámica	109
5.3.4.	Determinación de orientación	112
5.3.5.	Filtro de Kalman extendido	113
5.3.6.	Verificación de requerimientos	116
5.3.6.1.	Requerimientos en el caso de estudio K'oto	117
5.3.7.	Optimización de parámetros de control	119
6.	Conclusiones	121
6.1.	Trabajo a futuro	123
	Referencias	125

Índice de figuras

1.1. Actividades en donde participan los satélites.	2
1.2. Las órbitas LEO, HEO y geosíncrona son comunes en satélites enfocados a la Tierra.	3
1.3. Las 17 metas de desarrollo sustentable de la ONU apoyadas por el espacio.	4
1.4. Ejemplos de distintas configuraciones de CubeSats.	5
1.5. Imágenes retratando la trayectoria de México en el área espacial.	8
1.6. Misiones mexicanas por parte de universidades y centros de investigación.	11
1.7. Flujo seguido por la metodología del modelo en V.	20
1.8. Ejemplo un análisis de órbita mediante los propagadores de STK.	27
1.9. Simulación de un CubeSat como parte de los ejemplos predefinidos de Simulink.	29
2.1. Subsistemas que componen a un satélite.	36
2.2. Elementos que conforman un ADCS y su relación.	37
2.3. Ejemplo de un sensor solar digital.	39
2.4. Elementos keplerianos que conforman una órbita.	44
2.5. Definición de marco de referencia fijo al cuerpo del satélite.	46
2.6. Definición del marco de referencia LVLH u orbital.	47
2.7. Definición del marco de referencia ECI.	48
2.8. Definición del marco de referencia ECEF.	49
3.1. Diagrama representando el flujo de desarrollo de un ADCS siguiendo el modelo en V y el MBD.	58
3.2. Relación entre los distintos bloques de Simulink en el modelo virtual.	59
3.3. Ejemplo de reportes obtenidos mediante la interfaz gráfica de STK.	62
3.4. Diagrama del proceso de obtención de datos de STK a Simulink.	63
3.5. Ejemplo de la interfaz VRML en Simulink.	63
3.6. Flujo de información para el subsistema de visualización en Simulink.	64
4.1. Modelo implementado de los sensores solares en Simulink.	96
4.2. Primer modelo variante de los sensores solares, en donde el procesamiento ocurre a nivel de corriente.	97
4.3. Segundo modelo variante de los sensores solares, en donde el procesamiento ocurre a nivel de ángulos.	97
4.4. Modelo implementado del magnetómetro en Simulink.	98
4.5. Modelo implementado del giroscopio en Simulink.	98
4.6. Modelo implementado del GPS en Simulink.	99
4.7. Modelo simplificado del GPS en Simulink.	100

4.8. Modelo implementado de las magnetorcas en Simulink.	101
5.1. Comparación de los resultados de los modelos de densidad.	108
5.2. Velocidad angular real del satélite en la simulación con respecto al tiempo.	109
5.3. Par de control generado por las magnetorcas del satélite en la simulación con respecto al tiempo.	110
5.4. Momento magnético generado por las magnetorcas del satélite en la simulación con respecto al tiempo.	110
5.5. Potencia de las magnetorcas del satélite simulado con respecto al tiempo.	111
5.6. Orientación en cuaterniones del satélite simulado con respecto al tiempo en el marco de referencia orbital.	112
5.7. Error de la orientación resultante con respecto a la orientación objetivo (marco de referencia orbital). Se utiliza la representación de ángulos de Euler.	112
5.8. Captura proveniente del subsistema de visualización.	113
5.9. Error de la ecuación de Wahba para el algoritmo TRIAD en el satélite simulado.	114
5.10. Error de la velocidad angular medida con respecto a la velocidad angular real.	115
5.11. Error de la velocidad angular filtrada con respecto a la velocidad angular real.	115
5.12. Error de la orientación medida con respecto a la orientación real. . .	116
5.13. Error absoluto de la orientación filtrada con respecto a la orientación real.	116
5.14. Resultado de la ejecución de una prueba unitaria en la simulación. .	118
5.15. Resultado de la verificación de requerimientos de acuerdo a las pruebas unitarias en la simulación.	119
5.16. Resultado de optimización de las constantes de control para el algoritmo B-dot.	120

Índice de tablas

1.1. Clasificación de los satélites con respecto a su masa.	3
4.1. Características de las capas que componen la atmósfera.	69
4.2. Comparación de las características de los métodos de determinación de orientación estáticos.	85
5.1. Ventajas de la herramienta desarrollada durante este trabajo.	104
5.2. Elementos keplerianos de la ISS en el tiempo $t = 0$ de la simulación. .	106
5.3. Condiciones iniciales de la simulación.	107
5.4. Características de los sensores simulados.	107
5.5. Parámetros relacionados con el control de orientación.	108
5.6. Estadística descriptiva de los modelos de densidad seleccionados: Jacchia 1960 y Jacchia-Bowman 2008	109
5.7. Parámetros de inicialización del filtro de Kalman Extendido.	113
5.8. Evaluación del filtro de Kalman extendido.	117
5.9. Resultados de la verificación de requerimientos en el caso de estudio K'oto.	118



"A bright blue, South Atlantic Ocean", NASA (2020)

1

Introducción

LOS SATÉLITES son parte de la infraestructura que permite la existencia de varios servicios y actividades hoy en día como son las telecomunicaciones, los sistemas de navegación (GNSS), investigación, meteorología, solo por mencionar algunas actividades. Estas aplicaciones son posibles gracias a las ventajas que otorgan los sistemas espaciales. Por ejemplo, la cobertura global en sistemas de navegación es posible debido a las constelaciones de satélites que se encuentran distribuidas alrededor de la Tierra. Los satélites también otorgan una vista aérea sobre de regiones de interés, lo que permite su monitoreo de forma remota. En cuanto a la investigación, la cercanía de los satélites hacia los objetos de estudio localizados en el espacio permite la adquisición de datos sin las interferencias y dificultades que poseen los medios terrestres. En la Figura 1.1 se resumen estas y otras aplicaciones de los satélites.

Las aplicaciones de los satélites están fuertemente ligadas tanto a características técnicas como la órbita y el tamaño del satélite, así como a factores sociales como los antecedentes de los países que los desarrollan. En la siguiente sección se exploran algunos de estos conceptos con el fin de dar un mayor contexto al problema planteado por esta tesis.

1.1. Contexto

Formalmente los satélites se definen como cuerpos que se encuentran en órbita en torno a un cuerpo celeste, como lo son los planetas y estrellas [1]. Los satélites artificiales son aquellos que son fabricados por el hombre y para los cuales se usará el término satélite de ahora en adelante.

Las características de un satélite suelen estar ligadas con su tamaño, por lo que pueden clasificarse en términos de su masa (Tabla 1.1) [2]. En los últimos años han sido los satélites pequeños los que se han popularizado alrededor del mundo gracias a su bajo costo, abriendo la opción de impulsar paradigmas como las constelaciones, las cuales antes eran limitadas debido a los presupuestos.

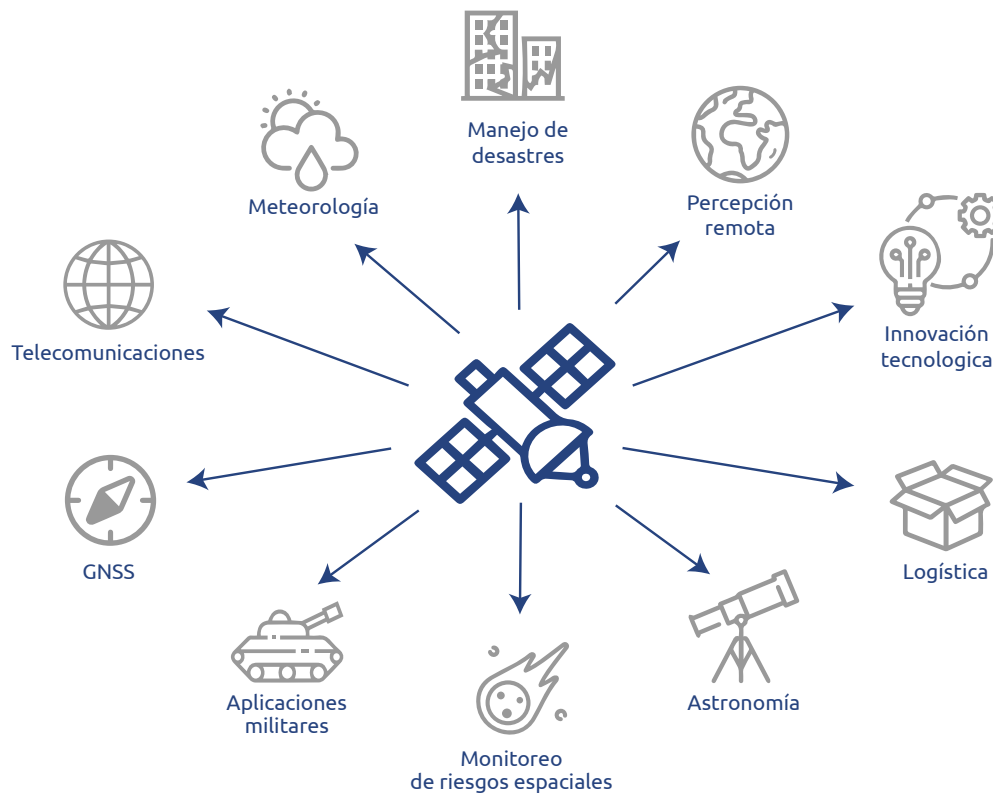


Figura 1.1: Actividades en donde participan los satélites.

Las órbitas constituyen la trayectoria periódica de un satélite a lo largo de su vida útil. La elección de la órbita depende de los objetivos de la misión, y pueden variar en cuanto a la orientación del plano orbital, la excentricidad (la circularidad de la órbita) y la distancia hacia el cuerpo hacia el cual orbitan. A continuación se listan algunos ejemplos de órbitas para satélites con aplicaciones enfocadas a la Tierra, así como sus representaciones gráficas (Figura 1.2).

Órbita Terrestre Baja (LEO). Los satélites en esta órbita se encuentran a una altura de 160 a 500 km sobre la superficie terrestre [1]. Es de las órbitas con más cercanía a la Tierra, lo que la hace una órbita accesible especialmente para satélites pequeños. Además, esta altitud la hace ideal para aplicaciones de percepción remota debido a una mayor resolución en las imágenes. El periodo orbital de estos satélites es corto, así que un satélite en LEO circulará por la Tierra varias veces por día. Por esta razón se suele hacer uso de constelaciones si se desean aplicaciones relacionadas con telecomunicaciones.

Órbitas altamente elípticas (HEO). Órbitas con un valor alto de excentricidad e inclinación orbital. El resultado es que el satélite transcurre gran parte de su órbita en línea de visión a un solo hemisferio, mientras que el tiempo transcurrido en el

Tabla 1.1: Clasificación de los satélites con respecto a su masa [2].

CLASIFICACIÓN	MASA (kg)
Satélite grande	$\geq 1,000$
Satélite mediano	≥ 500 y $<1,000$
Satélite pequeño	<500
Minisatélite	≥ 100 y <500
Microsatélite	≥ 10 y <100
Nanosatélite	≥ 1 y <10
Picosatélite	≥ 0.1 y <1
Femtosatélite	≤ 0.1

hemisferio opuesto será mínimo. El satélite aumentará su velocidad en el perigeo, con lo cual se reduce el tiempo fuera del hemisferio de interés. Debido a las razones anteriores, ha sido empleada históricamente para fines de telecomunicaciones, particularmente por Rusia.

Órbita geosíncrona. Órbita en donde el periodo orbital del satélite coincide con la rotación de la Tierra. Como consecuencia la vista del satélite permanece fija con respecto al área de interés. Para que un satélite mantenga esta órbita se requiere una altitud de 35,786 km con el fin de llegar a la velocidad orbital apropiada. Ejemplos de aplicaciones incluyen las telecomunicaciones y la vigilancia con fines militares.

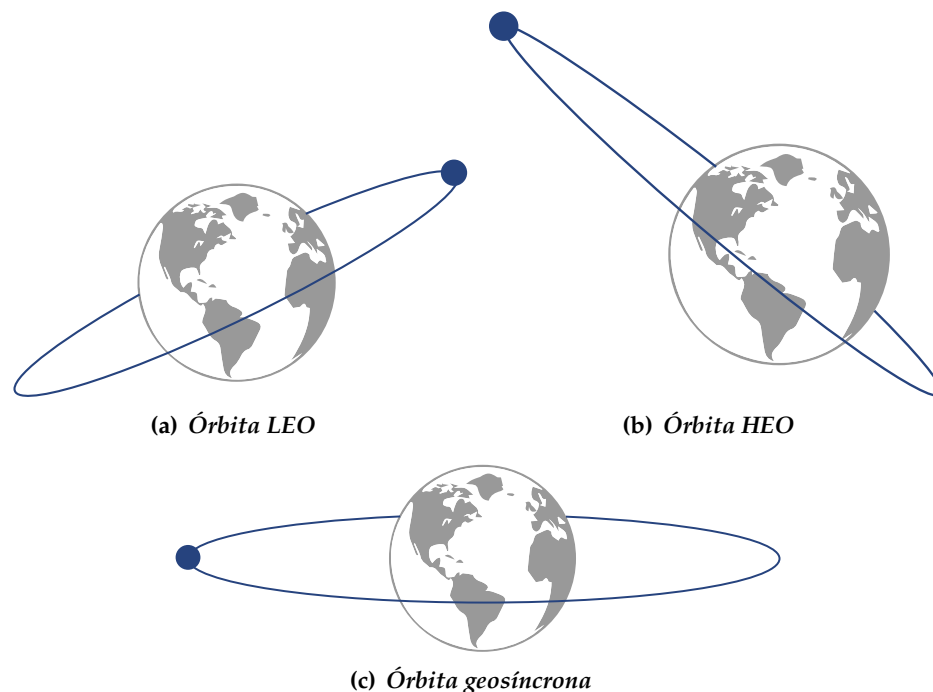


Figura 1.2: Las órbitas LEO, HEO y geosíncrona son comunes en satélites enfocados a la Tierra.

1.1.1. Los satélites y la sustentabilidad

La tecnología espacial es el soporte de varias de las tecnologías actuales, pero tiene el potencial apoyar otros rubros para aportar en temas de sociedad y sustentabilidad. Un ejemplo de este tipo de aplicaciones es el monitoreo en cuanto al cambio climático, en donde los satélites pueden ser usados para analizar los patrones del clima y como recursos de acción ante manejo de desastres naturales. El uso de medios espaciales enfocados en el cambio climático es un área que ha sido abordada por numerosos centros de investigación, pero existen otras áreas con un menor nivel de madurez en las que las aplicaciones del espacio también pueden contribuir significativamente. Con base en esta área de oportunidad, la ONU propuso 17 metas en las cuales se plantea el uso de tecnología espacial para cumplir la Agenda de Desarrollo Sustentable de 2030 [3]. Esta iniciativa también se conoce como *Space4SDGs* y sus objetivos se muestran en la Figura 1.3.



Figura 1.3: Las 17 metas de desarrollo sustentable de la ONU apoyadas por el espacio [3].

Algunos de estos objetivos involucran una aplicación directa de la tecnología, como el uso de percepción remota para el monitoreo de ecosistemas y la identificación de pesca ilegal, mientras que otros objetivos le conciernen a la formación de recursos y creación de oportunidades dentro del área espacial, como lo es la meta de igualdad de género.

En el caso de un país en vías de desarrollo, los satélites proveen entonces una forma de cumplir dos objetivos a largo plazo: el desarrollo mismo de la tecnología espacial y su aplicación para resolver problemas sociales y ambientales.

1.1.2. El estándar CubeSat

El estándar CubeSat es una propuesta de diseño de nanosatélites nacida en 1999 en la *California Polytechnic State University* de San Luis Obispo. Su objetivo fue el de facilitar el acceso al espacio a la comunidad científica mediante la estandarización de las especificaciones del nanosatélite, lo cual conlleva a una reducción de costo y tiempo de desarrollo. Un nanosatélite CubeSat típico adopta la forma de un cubo de 10 cm x 10 cm x 10 cm dentro del rango de 1-2 kg, donde a una unidad con estas características se le denomina «1U». Los CubeSat pueden componerse de varias unidades, llegando hasta los 12U. Ejemplos de las diferentes configuraciones se muestran en la Figura 1.4.

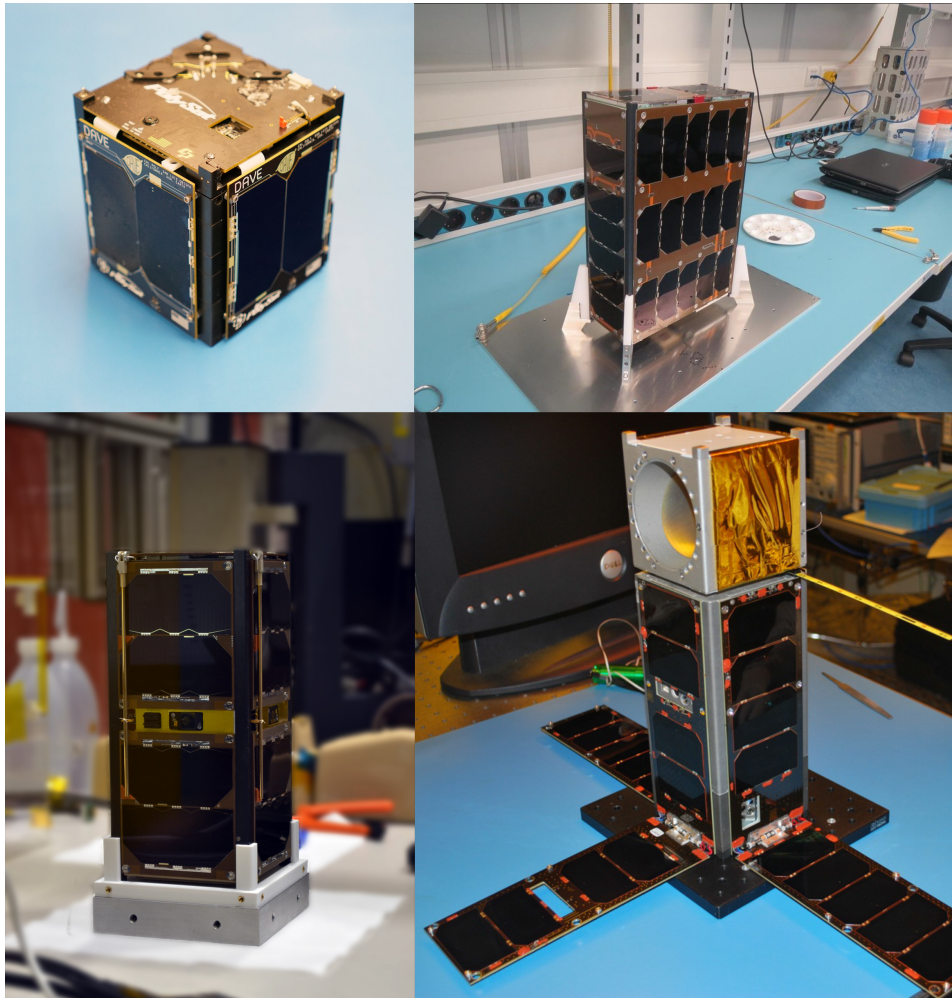


Figura 1.4: Ejemplos de distintas configuraciones de CubeSats. De la esquina superior izquierda, en sentido de las manecillas del reloj: DAVE, ITASAT-1, AISTECHSAT (DANU) 2U, MicroMAS-2 [4, 5, 6, 7].

El documento *CubeSat Design Specification* desarrollado por el programa CubeSat [8] define algunas consideraciones adicionales que se deben tomar al desarrollar un nanosatélite de este tipo y entre ellas se hace referencia a las especificaciones mecánicas, eléctricas y de operación.

En el lanzamiento, los CubeSats son desplegados a órbita por medio de un dispensador adjunto al vehículo de lanzamiento. El dispensador puede variar según las especificaciones de este vehículo, pero uno de los más comunes es el P-POD, el cual funciona por medio de rieles y tiene una capacidad de 3U [8].

La estandarización del CubeSat tiene como consecuencia que los fabricantes de material espacial puedan producir componentes comerciales, también conocidos como COTS (*Commercial Off-The-Shelf*), que se acoplen a cualquier satélite de este formato. Esto en turno se ve reflejado como una modularidad entre los componentes, de tal forma que en una misión CubeSat se pueda enfocar en el desarrollo de componentes o tecnología experimental, mientras que para el resto se pueden usar COTS.

Uno de los principales beneficios del estándar CubeSat es el costo, el cual también se ve derivado del uso de COTS, además de que se cuentan de programas como la *CubeSat Launch Initiative* de la NASA que ofrece llevar CubeSats a bordo de sus vehículos como parte de la carga auxiliar de mayor escala, reduciendo el costo de lanzamiento [9]. Con todas estas características, los CubeSats han incrementado la accesibilidad al espacio para la industria y la comunidad científica, abriendo la posibilidad de probar nuevos paradigmas y técnicas experimentales que antes no eran factibles en misiones de mayor escala. Los CubeSats también son un medio por el cual formar a los estudiantes en proyectos del área espacial, ya que estos trabajan directamente en su desarrollo lo que permite una experiencia de primera mano en cuanto a las aplicaciones de la teoría, la normatividad y las metodologías usadas en este tipo de proyectos.

En este trabajo se tomará como caso de estudio al nanosatélite K'oto, el cual es un satélite mexicano bajo el estándar CubeSat. Más información sobre esta misión se detalla en la Sección 5.2.

1.1.3. El área espacial en México

Existen precedentes en cuanto a la participación de Latinoamérica en el espacio, ya que los gobiernos de Latinoamérica a lo largo de su historia han colaborado en proyectos en conjunto con agencias internacionales ya establecidas como la NASA [10, 11]. Los objetivos de dichas misiones han variado de acuerdo a las necesidades y objetivos de cada país, pero todas comparten la visión de lograr la independencia hacia los recursos de otras naciones mediante el desarrollo de las agencias espaciales nacionales. Estas agencias representan una forma de impulsar la tecnología y de capacitar a los profesionistas locales en cuanto a proyectos del área espacial. Un efecto secundario de ello es el desarrollo de infraestructura que pueda apoyar estas actividades, lo que representa en sí mismo un desarrollo de estas capacidades.

En México se cuenta con la Agencia Espacial Mexicana, pero a diferencia de otros países, las misiones que se han llevado a través de su apoyo han sido en su

mayoría misiones universitarias. Estos tipos de misiones llevadas y por llevarse a cabo recientemente en los años siguientes quizás se encuentren dentro las únicas memorias que la mayoría de la población tenga sobre los avances relacionados con la ingeniería espacial nacional. El resultado es la existencia de la percepción de que México se encuentra en sus primeros pasos hacia el espacio. Sin embargo, la historia de México en esta materia no es algo reciente y en realidad se remonta a mediados del siglo XX.

1.1.3.1 HISTORIA DEL ÁREA ESPACIAL EN MÉXICO

Uno de los primeros acercamientos formales de México al desarrollo de su industria espacial fue la construcción de cohetes. A partir de 1957, instituciones como la Universidad de San Luis Potosí diseñaron y lanzaron cohetes de mediana escala, donde la capacidad de carga útil rondaba los 8 kg. El objetivo de estos vehículos se enfocaba completamente en fines educativos y de investigación, enfocándose en el estudio de la atmósfera [12]. La característica compartida por los proyectos de este periodo es la independencia y aislamiento. No existía una visión compartida entre las instituciones involucradas que fomentara la colaboración o el cumplimiento de objetivos a largo plazo, factores que son reconocidos como fundamentales dentro de este rubro hoy en días.

No fue hasta 1962, durante el mandato de Adolfo López Mateos, cuando se creó la Comisión Nacional del Espacio Exterior (CNEE) en respuesta a estas inquietudes. El objetivo de la comisión consolidó lo que los pioneros mexicanos buscaban: la investigación y exploración del espacio; además de promover estas actividades. Este objetivo también se vio reflejado en la postura del país en los años siguientes. En gran contraste con las agencias estadounidense y soviética, México no entró en una carrera espacial con fines de competencia, por lo que el principal propulsor de su agencia era simplemente el desarrollo científico y tecnológico con fines completamente pacíficos.

En 1967 México se integró al sistema satelital Intelsat y al requerir de infraestructura para recibir los datos, se construyó la primera estación terrena de México en Tulancingo. En 1968 se transmiten los juegos olímpicos que se llevaron a cabo en el país por medio de ella.

Con varios hitos alcanzados y proyectos cada vez más ambiciosos como la serie de cohetes MITL donde se alcanzó llegar al espacio exterior, México tenía por delante un futuro prometedor [12]. Sin embargo, la crisis económica durante el mandato de José López Portillo estremeció varios sectores del país incluido el espacial, donde se tomó la decisión de desaparecer la CNEE.

En la Figura 1.5 se resumen algunos de los acontecimientos mencionados hasta ahora, previo la desaparición de esta organización.

1. INTRODUCCIÓN

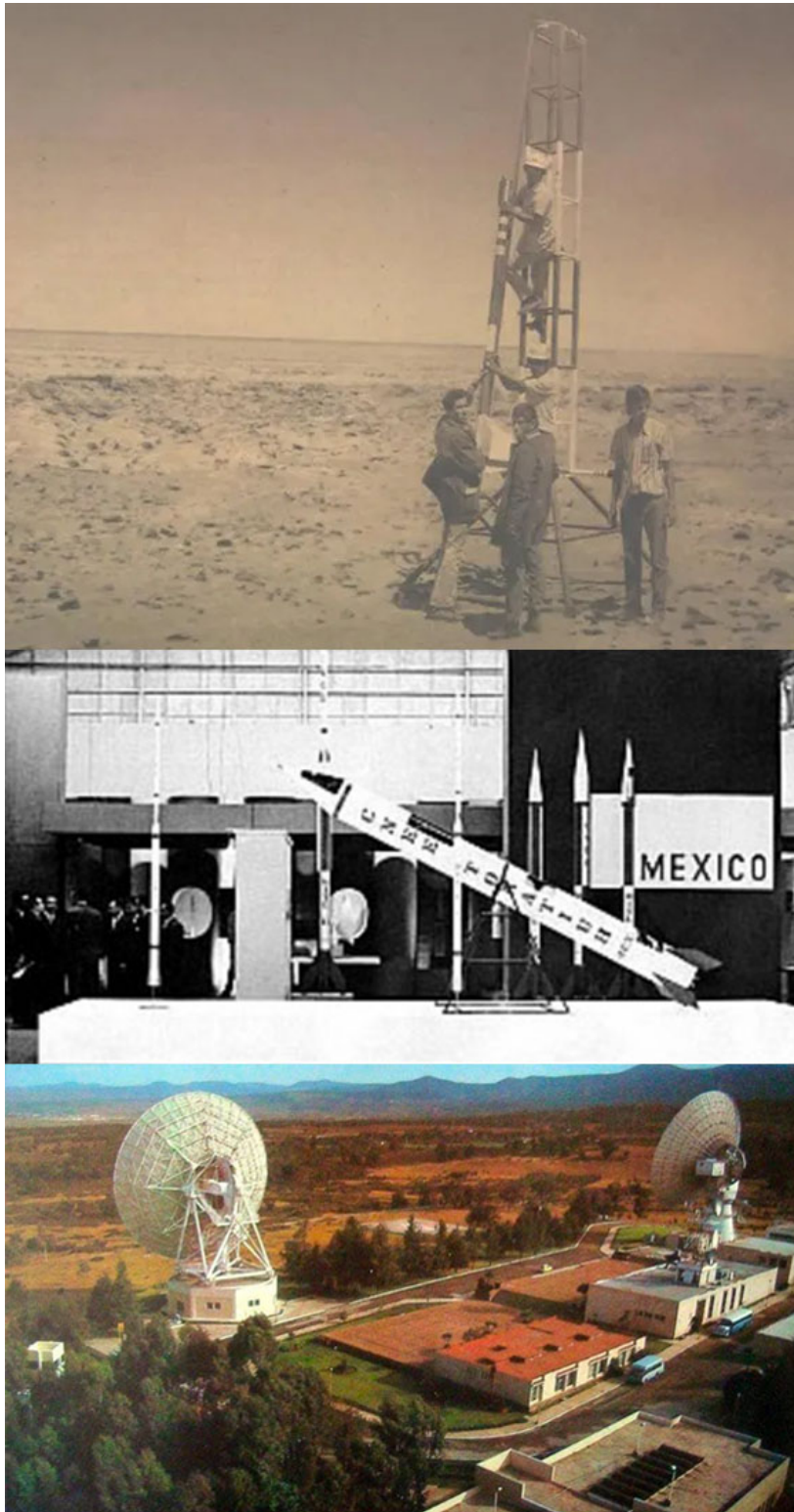


Figura 1.5: Imágenes retratando el desarrollo de México en el área espacial. De arriba hacia abajo: Lanzamiento de Filoctetes II, de la Universidad de San Luis Potosí; exposición de cohetes; estación terrena de Tulancingo [12, 13, 14].

Los años siguientes aún tuvieron varios éxitos como el lanzamiento de los sistemas de telecomunicaciones Morelos I y Morelos II en los ochenta, así como de Solidaridad I y II en los noventa. Por su parte las universidades continuaron con sus proyectos propios y cada vez más de ellas se involucraron más en el desarrollo de pequeños satélites estudiantiles, destacando la UNAM con el Programa Espacial Universitario (PEU). Sin embargo, la tendencia que comparten en común todos estos proyectos es nuevamente el aislamiento que se tenía antes de la fundación de la CNEE, carente de una visión a largo plazo en cuanto al país [15].

No fue sino hasta la vuelta del siglo, en el 2010 donde surgió de nuevo un interés por parte del gobierno en el sector espacial con la fundación de la Agencia Espacial Mexicana.

Su creación representó uno de los primeros avances del país en el siglo presente por impulsar la ciencia y tecnología espacial bajo una visión nacional. Fue en 2012, en conjunto con ProMéxico y la Secretaría de Comunicaciones y Transportes, cuando se publicó el documento “Plan de órbita” en el cual se establece un análisis del panorama de la industria, identificando jugadores clave tanto a nivel nacional como internacional, y se plantean una serie de objetivos que se desean lograr en los años siguientes. Una segunda versión, “Plan de órbita 2.0” fue publicada en 2017 con actualizaciones basadas en los cambios ocurridos durante ese periodo, tanto a nivel nacional como internacional [16].

El plan de órbita 2.0 contiene un análisis FODA de la industria espacial mexicana, en donde como debilidades resaltan el escaso conocimiento del sector y recursos humanos con capacidades poco desarrolladas, mientras que la dependencia tecnológica se considera una amenaza. Estos tres factores tienen una relación estrecha entre sí. Por un lado, México es un generador de capital humano capacitado, pero la escasa cantidad de proyectos tiene como consecuencia un déficit de especialistas al limitar la cantidad de personas que pudieran ser capacitadas en el área.

En un estudio donde se analizó el estado de la industria aeroespacial en Sonora en 2018 [17], se encontró que los principales retos que enfrenta el estado es la mejora del capital humano y de la infraestructura. Hacen hincapié en la falta de experiencia a pesar de contar con personal altamente calificado, contactos e instituciones educativas vinculadas a la industria. Es una tendencia que se extiende en todo el país. Esto a su vez resulta en que el sector sea virtualmente inexplorado bajo la perspectiva nacional, generando una dependencia tecnológica hacia países más avanzados. La clave para impulsar la industria, es entonces, el desarrollo de proyectos propios.

1.1.3.2 MISIONES SATELITALES DESARROLLADAS EN MÉXICO

Las universidades y centros de investigación representan una de las principales fuentes para la generación de nuevos proyectos y han sido el origen de misiones satelitales desarrolladas en el país y que fueron lanzadas con éxito. En esta sección se enumeran algunas de estas misiones (Figura 1.6).

UNAMSAT-B. Microsatélite cúbico de 23 cm por lado, con el objetivo de medir la velocidad de entrada de los meteoritos a la atmósfera terrestre. Fue desarrollado por

la Facultad de Ingeniería de la UNAM y su lanzamiento se llevó a cabo con éxito el 5 de septiembre de 1996. El UNAMSAT-B surgió a raíz del considerado el primer satélite universitario, el UNAMSAT-1, el cual fue destruido antes de alcanzar su órbita debido a un fallo en el vehículo de lanzamiento [18].

Aztechsat-1. Este proyecto fue desarrollado por estudiantes de la Universidad Popular Autónoma del Estado de Puebla (UPAEP) en colaboración con NASA, CONACYT, el Clúster Espacial MX Space y la Universidad Autónoma de Chihuahua [19]. La misión de AztechSat-1 fue «la intercomunicación satelital con la constelación GlobalStar, con el fin de desarrollar, implementar e integrar un subsistema en el nanosatélite, que permita mejorar la comunicación en las misiones futuras de los nanosatélites de clase CubeSat» [20]. Fue lanzado el 5 de diciembre de 2019 desde Cabo Cañaveral a bordo de un Falcon 9. Al 4 de octubre de 2020, AztechSat-1 cumplió con éxito su misión en la Estación Espacial Internacional, donde realizó un enlace intersatelital con la constelación GlobalStar.

NanoConnect-2. Desarrollado por el Laboratorio de Instrumentación Espacial LINX del Instituto de Ciencias Nucleares (ICN) de la UNAM. Este CubeSat 2U fue lanzado el 28 de febrero de 2021 a través de la Agencia India de Investigación Espacial (ISRO) [21] y fue colocado a 504 km de altitud [22]. Es un satélite que fue diseñado en su totalidad en México, en colaboración con el gobierno de Hidalgo [23] Su objetivo fue el de probar la instrumentación y subsistemas desarrollados, así como su enlace con la estación terrena [21].

Painani-I. Proyecto desarrollado por el CICESE con apoyo de la Secretaría de la Defensa Nacional [24]. El Painani-I es un CubeSat 3U que fue puesto en órbita el 28 de junio de 2019 desde Nueva Zelanda por medio de la empresa Rocket Lab [25]. Cuenta con un módulo de captura de imágenes, del cual se deseó probar las capacidades de los sensores que lo constituyen. Uno de sus objetivos fue el de ser plataforma de aprendizaje para la Universidad del Ejército y Fuerza Aérea Mexicana [26].

1.1.3.3 EL PAPEL DE LA INDUSTRIA AEROESPACIAL

México se ha abierto paso en los últimos años en la industria aeroespacial, llegando a ser el décimo cuarto productor a nivel mundial y el séptimo proveedor de la industria aeroespacial en Estados Unidos [16]. Esto ha sido posible gracias a la llegada de empresas transnacionales, las cuales han establecido sus centros de manufactura e invertido en algunos estados de la república. Las empresas nacionales también han jugado un papel importante, puesto que en 2020 el número de empresas nacionales subió a 450 [27], indicando un crecimiento en el área. Para el año 2025 se tiene como meta colocar al país dentro de los primeros diez lugares internacionales en materia de exportación [27].

Según la secretaría de Economía, en el 2015 los estados con un mayor número de empresas dentro del sector espacial fueron Baja California, Sonora, Chihuahua,

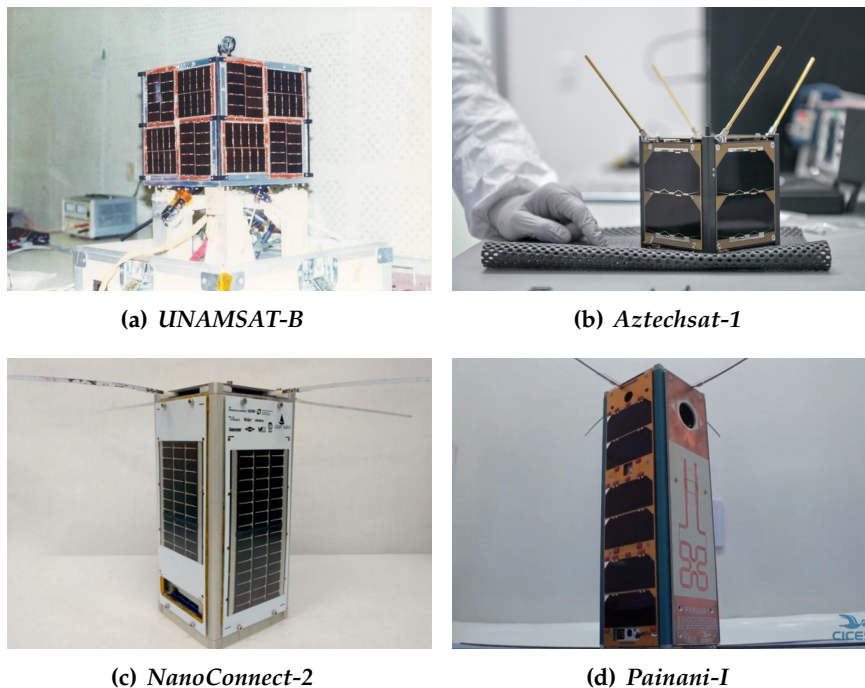


Figura 1.6: Misiones mexicanas por parte de universidades y centros de investigación.

Querétaro y Nuevo León. El estado más involucrado en el desarrollo de nueva tecnología es Baja California mediante la ingeniería y diseño, e instrumentación; lo cual se puede atribuir a su cercanía geográfica con las empresas estadounidenses de la frontera. Por otra parte, los demás estados se dedican en su mayoría a la manufactura. En el caso de Querétaro, el enfoque se encuentra en partes de fuselaje, trenes de aterrizaje y maquinados de precisión.

La producción aeroespacial nacional se puede dividir en tres rubros, donde del total 79 % se dedica a manufactura, 11 % mantenimiento y revisión y 10 % al diseño e investigación. Todos ellos componen los pasos de la fabricación de una aeronave y por tanto son dependientes entre sí [17]. Sin embargo, la mayoría de las labores dentro del país se concentran en la manufactura debido a los intereses de las empresas extranjeras, y como resultado la producción de diseños e investigaciones originales es baja. Por el otro lado, el ambiente proporcionado por la manufactura al generar la infraestructura, capacitación de recursos y su lugar en la cadena de producción proporciona una base fuerte para un desarrollo potencial de la industria aeroespacial nacional en los demás rubros, lo cual resultará en beneficios al área espacial.

1.1.3.4 QUERÉTARO: SU ESTADO ACTUAL Y POTENCIAL

En Querétaro se estableció el llamado clúster aeroespacial, también conocido como el Aeroclúster Querétaro, en el cual se han concentrado empresas y cadenas de

producción en esta materia. Previo a la pandemia se reportaban 12,000 empleos del área espacial en el estado, dentro de los cuales 9,500 estaban asociados al Aeroclúster.

La agrupación en clústeres es uno de los objetivos buscados por la industria aeroespacial independientemente del país o región. La cercanía entre empresas permite la división de tareas y la retroalimentación continua entre cada una de las partes involucradas en cuanto a productos, servicios y estándares de calidad. Las industrias aeronáutica y espacial están entrelazadas entre sí en términos de infraestructura y cadenas de producción, por lo que los recursos existentes de la industria aeronáutica pueden ser beneficiosos para la industria espacial, particularmente en Querétaro. Los beneficios de los clústeres en la cadena productiva son atractivos no solo para las empresas, sino también para los estados que las alojan [17], por lo que existe un beneficio mutuo tanto para la industria aeroespacial como para la economía del Estado.

En cuanto al área académica, por la parte de aeronáutica se tiene a la Universidad Aeronáutica en Querétaro mientras que por el área espacial se tiene a la Unidad de Alta Tecnología de la UNAM campus Juriquilla. Es en esta última en donde se propone el desarrollo de proyectos para generar capital humano capacitado en el área espacial, tanto para beneficio del país y de la industria. Un ejemplo de ello es el proyecto K'oto: un nanosatélite cuyo propósito es percepción remota del territorio nacional, desarrollado por la UNAM con el apoyo de la Secretaría de Desarrollo Sustentable del Estado de Querétaro (SEDESU).

El crecimiento de las capacidades de Querétaro tanto en industria como academia se relacionan con el análisis FODA de la industria espacial Mexicana establecido en el plan de órbita 2.0. En él se establecen como debilidades el bajo nivel de desarrollo de proveedores y de emprendimiento basado en desarrollos tecnológicos. Este es un aspecto en el cual el Aeroclúster otorga su aportación, mediante la colaboración conjunta entre diferentes nichos del sector que llevará a un desarrollo y exploración de nuevas aportaciones según la definición de un clúster. Específicamente estos avances de la industria se enfocan en el eje temático de «Impulso a la cadena de valor del sector e impulso al desarrollo industrial». Por otra parte, la academia se enfoca en el desarrollo de proyectos con tecnología y medios localizados en el estado y el país, promoviendo la colaboración entre distintos estados de la república. Este aspecto además implica el ganar experiencia en proyectos de este tipo, lo cual también es un aspecto buscado en el plan de órbita 2.0.

En resumen, Querétaro juega un papel importante en el cumplimiento de objetivos para impulsar el sector espacial en México y presenta un área de oportunidad para ser un jugador clave en los siguientes años.

1.2. Planteamiento del problema

El software de una misión espacial es caracterizado por tener altos requerimientos de seguridad y confiabilidad (*reliability*). Esta característica es reflejada en el software embebido en el satélite —conocido como software de vuelo—, debido a que la

falta de acceso físico a su electrónica tras el lanzamiento impone cierto rigor en el desarrollo de modelos y código para minimizar la posibilidad de accidentes. Por consiguiente, tareas críticas para el cumplimiento de la misión como el flujo de datos, la prioridad de comandos y el control de actuadores deben implementarse mediante metodologías de tolerancia de fallos que, si bien son rígidas, aumentan la probabilidad de éxito.

El enfoque anterior no se limita al software de vuelo, puesto que también es parte esencial del software destinado a ser usado en Tierra —en producción y en operaciones, por ejemplo— como los simuladores, el software de telecomunicaciones y el software de validación y verificación. En otras palabras, las características de confiabilidad y seguridad se hacen presentes en todos los ámbitos del software espacial. Es por estas necesidades por las cuales surgieron una serie de estándares y guías que sirven para aumentar el rigor durante el proceso de desarrollo.

1.2.1. El software en las misiones espaciales

De forma general, los estándares de software presentan una serie de recomendaciones, metodologías de desarrollo y metodologías de revisión con el objetivo de crear sistemas robustos y a prueba de falla. Las recomendaciones van desde la parte de administración de proyectos —con un gran énfasis en la documentación— hasta la parte técnica donde se alude al uso de recursión y asignación de memoria.

Como ejemplo, la NASA ha empleado el *JPL Institutional Coding Standard for the C programming language* como parte de sus misiones. Este estándar es de naturaleza técnica y consiste en un conjunto de guías empleadas en el desarrollo de sistemas en donde la seguridad sea un aspecto crítico, y no solo ha sido usado en el ámbito espacial, sino también en otras áreas como la automotriz. Otro conjunto de guías con el mismo objetivo es MISRA C. Por otra parte, uno de los estándares exclusivos del área aeroespacial es TOR-2004(3909)-3537, el estándar de desarrollo para sistemas espaciales. Todos estos estándares son una buena herramienta para mejorar la calidad del software desarrollado e incluso se complementan entre sí, pero tienen un aspecto importante a considerar: son completamente opcionales.

El seguimiento o no de los estándares de software queda a criterio del equipo de desarrolladores, ya que no existen certificaciones o cualificaciones espaciales directamente asociadas a ellos. El código no será examinado, dictaminado o puesto a prueba por un tercero como es el caso de otros aspectos del satélite en pruebas como compatibilidad electromagnética, termovaciación y vibraciones. Esto es una tendencia general en cuanto a hardware contra software. Incluso los estándares de uso general como *Launch Services Program (LSP)* de la NASA y *CubeSat Design Specifications (CDS)* del programa CubeSat hacen mención de las pruebas mencionadas anteriormente relacionadas con el hardware, pero no se hace mención directa del software.

El hecho de que no haya certificaciones obligatorias asociadas al software, de que no existan convenciones que forzosamente deban adoptarse en los códigos y que los estándares más usados no lo subrayen como un área prioritaria dan lugar a una conclusión: los desarrolladores tienen el control total de cómo desarrollar

su software, pero esa libertad implica la responsabilidad adicional de someter los entregables a pruebas y metodologías más rigurosas.

El caso de los CubeSats podría aparentar ser una excepción a lo anterior. Características que definen a una misión de este tipo como el bajo costo, corto tiempo de desarrollo y reducido número de integrantes podrían hacer parecer los esfuerzos de revisión de software como algo de más cuando podrían ser reubicados a áreas más prioritarias como aquellas mencionadas en los estándares generales [28]. La realidad es que dichos esfuerzos son igual de necesarios en las misiones CubeSat.

Es común que los CubeSats sean el primer contacto de varias universidades en el área espacial, y por ende el éxito o no de estos proyectos puede definir como se tratarán los proyectos subsecuentes. Además, las CubeSat suelen usarse a manera de prueba de concepto, por lo que está dentro del mejor interés de todas las partes involucradas que el satélite funcione y así se pueda recolectar la información de estos enfoques experimentales.

Desarrollar software con altos requerimientos de seguridad y confiabilidad es un medio para cumplir estos objetivos, lo que da lugar a la necesidad de herramientas auxiliares para apoyar este proceso.

1.2.2. La importancia de las simulaciones

Existe otro elemento clave que va de la mano con el software de vuelo, y que es indispensable en su desarrollo y despliegue: las simulaciones. El código dentro del satélite puede seguir todas recomendaciones de los estándares de software, tener el mínimo número de *bugs* permisibles y funcionar sin interrupción en el espacio, pero no se garantiza el funcionamiento al nivel de cumplimiento de objetivos si no se tomaron en cuenta simulaciones y las implicaciones de sus resultados durante su desarrollo.

Un ejemplo de lo anterior podría darse cuando el software de vuelo se diseñó con un comportamiento en mente —por ejemplo, que el satélite se limitará a un rango de orientaciones—, pero las simulaciones indican que si bien, ese comportamiento sucederá en la mayoría de ocasiones, existen condiciones en las que el satélite puede salir de dicho rango. Son esos momentos los que pueden llevar a un estado crítico y significar la diferencia entre el éxito y el fracaso, y son las simulaciones las que ayudan a determinarlos.

Además de su utilidad durante la fase de diseño, las simulaciones también son empleadas en las fases de validación y verificación. Mediante procesos como Hardware-in-the-Loop es posible poner a prueba el funcionamiento de la electrónica de los subsistemas al enviar señales provenientes de las simulaciones. Los resultados pueden traer a la luz detalles finos que deben ser atendidos al nivel de hardware. Es con base en lo anterior que las simulaciones no únicamente se limitan a una predicción, sino también como una forma de comprobación.

Se ha mencionado la función de las simulaciones durante las fases preliminares y posteriores del desarrollo, pero su verdadera fortaleza radica en su relevancia durante todo el proceso. Los modelos pueden proveer claridad a la hora de tomar decisiones. Desde diseños conceptuales hasta la colocación de componentes en

diferentes posiciones, todos ellos son puntos que implican la toma de una decisión dentro de un conjunto de alternativas que definirán a su vez más decisiones en el futuro. El uso de modelos sirve como un referente cuantitativo en el cual basarse a la hora de comparar distintos casos y que además puede usarse como una forma de llegar a acuerdos cuando se busca satisfacer los requerimientos simultáneos entre subsistemas. Con esto en consideración, las simulaciones representan una herramienta que soporta gran parte del ecosistema del desarrollo de una misión y con las cuales se puede reducir el factor de incertidumbre.

1.2.3. La necesidad de herramientas de verificación en el desarrollo de un ADCS

Un nanosatélite se compone de subsistemas, dentro de los cuales destaca el sistema de determinación y control de orientación o ADCS por sus siglas en inglés. Este subsistema se encarga de estabilizar al satélite y de regular su orientación a lo largo de su vida útil. El funcionamiento de la carga útil es comúnmente dependiente de la orientación, por lo que el ADCS toma una importancia que lo liga directamente al éxito de la misión.

El diseño del controlador de un ADCS empieza por predecir cómo el satélite se orientará en el espacio con el paso del tiempo, en otras palabras, el dominio de la cinemática y dinámica. Este hecho tiene implicaciones no solo en el objetivo principal, sino también en los demás subsistemas. Por ejemplo, en el enlace de comunicación con la estación terrena para el subsistema de telecomunicaciones o la orientación de los paneles solares para el subsistema de potencia. Una falla en el control de orientación implicaría una disrupción en estos subsistemas, lo cual es otro argumento a favor de los requerimientos de confiabilidad en el software además de simulaciones que ayuden a comprobar la funcionalidad del subsistema.

Las herramientas usadas para el desarrollo del ADCS sirven como un punto de partida para tomar en consideración ambos aspectos. En este caso, todo empieza con una herramienta con la cual poder predecir la dinámica del satélite al probar distintas configuraciones en las simulaciones y someterlo a las condiciones del espacio. El desarrollo de dicha herramienta es el objetivo de esta tesis. Sus resultados auxiliarán y ayudarán a respaldar decisiones relacionadas con el diseño del sistema de control de orientación y subsecuentemente lo evaluarán durante el proceso de validación y verificación. Las funciones y modelos de esta herramienta podrán expandirse para incluir otros elementos pertenecientes al ADCS como los sensores y actuadores.

El uso de software comercial es una propuesta ya establecida para solucionar esta necesidad. Con el respaldo de la herencia de vuelo, equipos dedicados de soporte técnico y los elevados estándares requeridos de un producto comercial, este tipo de software tiene características que lo respaldan como un medio de satisfacer los requerimientos necesarios de las simulaciones. Además, traen consigo funcionalidades adicionales relacionadas con los demás subsistemas. Sin embargo, el software comercial tiene la desventaja de requerir licencias, las cuales no solo implican otro costo a un presupuesto ya limitado, sino que les otorga a los proveedores la

capacidad revocarlas a su criterio. Acción que, de llevarse a cabo, pondrá en juego tanto la misión presente y como planes de misiones futuras.

El punto anterior se relaciona con los objetivos planteados en el plan de órbita 2.0, en particular el de la independencia tecnológica. Se necesitan desarrollar herramientas propias no solo por razones económicas sino también tecnológicas, puesto que las primeras versiones de estas herramientas pueden servir como punto de partida para futuros proyectos que se desarrollen dentro del país.

1.3. Objetivos

Implementar un modelo virtual de un nanosatélite en órbita baja mediante técnicas Software-in-the-Loop (SIL) para verificar el funcionamiento de un sistema de control de orientación magnético de un nanosatélite que cumpla con los requerimientos de la misión.

1.3.1. Objetivos particulares

- Implementar las ecuaciones de cinemática y dinámica de un nanosatélite en órbita terrestre baja (LEO) para obtener un modelo virtual mediante Simulink.
- Obtener los parámetros ambientales y de la órbita definida en la misión simulada mediante STK.
- Utilizar técnicas SIL para verificar que el campo magnético producido por las bobinas sea el adecuado según los requerimientos.

1.4. Hipótesis

Es posible implementar un modelo virtual de un nanosatélite y simular las condiciones ambientales a las que será expuesto en la trayectoria durante su órbita, lo que permitirá evaluar el rendimiento del sistema de ADCS al usar actuadores magnéticos (bobinas magnéticas) y estimar el cumplimiento de los requerimientos establecidos en la misión.

1.5. Antecedentes

El sistema de determinación y control de orientación (ADCS) está compuesto por sus dos partes homónimas: el sistema de determinación de orientación—cuyo propósito es obtener la orientación del satélite en el espacio mediante la información de sensores—y el sistema de control de orientación, con la tarea de accionar los actuadores para alcanzar la orientación deseada. El sistema de control depende del sistema de determinación para obtener información sobre el estado del sistema y

el sistema de determinación arrojará resultados que a su vez están basados en las acciones realizadas por el sistema de control.

Las nuevas tecnologías de ADCS actualmente se direccionan hacia la miniaturización de sensores y actuadores, buscando lograr una mayor precisión a la vez que se reduce la masa del satélite, lo que conlleva a una reducción de costos. Pero a pesar de estos avances, los CubeSats han adoptado en su gran mayoría estrategias de control de orientación clásicas, poniendo gran importancia a la confiabilidad otorgada por la herencia de vuelo. Entre los métodos de control más usados por CubeSats se encuentran las ruedas de reacción, control magnético pasivo y control magnético activo [29]. Las ruedas de reacción son favorecidas cuando se requiere de un control de orientación fino mientras que los esquemas basados en magnetismo se favorecen cuando se busca tener una solución de bajo costo.

Las características de una gran parte de los nanosatélites se alinean con el segundo enfoque, en particular mediante el uso del control de orientación magnético activo. Dentro de esta categoría se destaca el uso de bobinas magnéticas o magnetorcas, las cuales además de su bajo costo, tienen la ventaja de no utilizar mecanismos, a diferencia de las ruedas de reacción, que causen desgaste mecánico e incrementen las posibilidades de falla; y a diferencia de un control de orientación pasivo, otorgan una mayor flexibilidad en cuanto al rango de orientaciones disponibles.

Las magnetorcas han sido usadas también por picosatélites como COMPASS-1 [30]. El principio de funcionamiento de las magnetorcas se basa en la generación de un campo magnético en las bobinas cuando una corriente se hace circular sobre ellas. El campo magnético generado interactúa con el campo magnético de la Tierra, generando un par que puede ser usado para rotar el satélite hasta la posición deseada [31].

Gracias a la popularización de las misiones tipo CubeSat, hoy en día las magnetorcas, así como otros componentes, pueden encontrarse como COTS que van desde placas que únicamente comprenden los actuadores hasta módulos que cubren el ADCS en su totalidad. La adquisición de estos últimos representa un mecanismo de acelerar el desarrollo de la misión, respaldados por herencia de vuelo y certificaciones. Algunos ejemplos de COTS de ADCS son el *Arcus ADCS* de la universidad KU Leuven y el *CubeADCS* de CubeSpace.

En el caso del desarrollo de algunos nanosatélites, el uso de COTS para el ADCS se restringe a componentes individuales y no a todo el sistema como los productos mencionados anteriormente. Parte de la razón se puede atribuir al objetivo desarrollar tecnología propia. Con esta decisión se está haciendo un *trade-off* sobre la tecnología que se podrá desarrollar en el futuro contra la certeza que traen los COTS de ADCS que ya fueron probados y puestos en órbita. La incertidumbre que trae la experimentación de nuevos enfoques conlleva a la necesidad de adoptar una metodología de diseño y desarrollo rígida que ayude a minimizar los riesgos. Tal metodología es el diseño basado en modelos.

1.5.1. Diseño basado en modelos

El diseño basado en modelos (*Model-based design* o MBD) es una metodología usada comúnmente en el área de control. El MBD consiste en la elaboración de modelos como el principal método de trabajo durante el ciclo de desarrollo. Esto significa que los modelos pueden ejecutarse y actualizarse durante cualquier fase, por lo que es posible evaluar continuamente el comportamiento del sistema representado contra sus requerimientos [32]. El hecho de que los modelos sean el principal método de trabajo quita la necesidad de usar herramientas diferentes para cada fase, y en su lugar se promueve la iteración y mejora subsecuente de los modelos.

El diseño basado en modelos está pensado para sistemas embebidos. Parte de la razón por la que surge es para agilizar el proceso de desarrollo en el uso de estos componentes. En lugar de elaborar el código, implementarlo y hacer pruebas en el hardware; el hardware puede abstraerse mediante modelos y así formar parte de las simulaciones. Por ejemplo, el comportamiento de un motor puede reducirse a un conjunto de modelos matemáticos y sus características técnicas a valores de parámetros. Esta abstracción permite que esta metodología pueda escalar a proyectos más complejos como del área médica, aeronáutica y espacial.

El proceso general seguido en esta metodología consiste en modelar la planta del sistema, con lo cual se puede ver el comportamiento del sistema al variar los parámetros de su modelo matemático. En segundo lugar, se modela el controlador que puede incluir elementos eléctricos, mecánicos, hidráulicos, etc. El modelo del controlador interactuará con la planta dentro del mismo ambiente de simulación. Los modelos se van mejorando progresivamente y son sujetos a diferentes técnicas, desde simulaciones fuera de línea hasta simulaciones de tiempo real. El código que se implementará en los sistemas embebidos puede ser generado automáticamente a partir de los modelos.

El diseño basado en modelos difiere de las metodologías tradicionales de diseño en que gran parte del trabajo puede ser desarrollado únicamente desde el entorno de simulación. Los casos que pueden implicar un riesgo como la manipulación de componentes delicados o las situaciones en donde no se tenga acceso a cierto hardware, representarían un retraso o inconvenientes en las actividades de las metodologías tradicionales, pero en el MBD simplemente pueden ser sustituidos por simulaciones. Simulaciones que pueden ejecutarse bajo distintas condiciones y parámetros, que pueden ir más allá de los físicamente disponibles para considerar diseños teóricos para la evaluación de conceptos.

Una de las características que definen al MBD es su entorno visual. Enfoques como Simulink, Xcos y SysML hacen una segmentación mediante bloques. Cada bloque puede representar un conjunto de funciones con el mismo propósito, por ejemplo, actuadores, o un conjunto de objetos representables el mundo físico, como por ejemplo motores. El enfoque basado en bloques gráficos confiere dos ventajas a simple vista: la facilidad de comprensión y la modularidad.

La facilidad de comprensión surge debido a que es posible observar fácilmente la relación que guardan los bloques entre sí, de tal manera que se puede identificar a primera vista que elementos serán afectados al modificar ciertos parámetros. Este

aspecto también involucra la comunicación entre miembros del equipo, ya que facilita la comunicación de ideas y conceptos, además de que el aspecto visual es atractivo en fines de documentación.

El aspecto de modularidad se encuentra en que cada bloque y su independencia del sistema que lo alberga, ya que solo están definidos en términos de entradas y salidas. Como consecuencia los modelos son escalables y los bloques pueden reusarse en distintos proyectos. En contraste, desarrollo de código tradicional implicaría asignar mayores esfuerzos a la modificación de código existente para poder reutilizarlo. Otra ventaja de la modularidad yace en que facilita el trabajo multidisciplinario. Al usar las mismas herramientas y entorno, los especialistas pueden enfocarse en la realización de los bloques pertenecientes a su dominio. La problemática para integrar las distintas partes es manejada por los programas.

Otras virtudes a resaltar del MBD es una reducción de los esfuerzos relacionados con la elaboración de software. El código es generado automáticamente a partir de los modelos, lo cual ahorra el tiempo que se dedicaría a un desarrollo manual, corrección de bugs e implementación. Con ello se permite enfocarse más en los aspectos de diseño. Además, la generación de código automática cesa los errores de origen humano, proporcionando una mayor seguridad al código resultante.

1.5.1.1 MODELO EN V

El diseño basado en modelos se relaciona íntimamente con el modelo en V.

El modelo en V es una metodología más general, cuyo uso se extiende más allá de la ingeniería de control, siendo usado en proyectos de ingeniería de software y desarrollo de productos. El modelo en V sigue una estructura lineal, la cual como su nombre indica, sigue la forma de una V (Figura 1.7).

A grandes rasgos la parte izquierda de la V engloba los pasos relacionados con el diseño y la especificación de requerimientos, mientras que la parte derecha engloba la parte de pruebas e integración. Dentro de cada una de las etapas del modelo en V se puede encontrar un análogo en el diseño basado en modelos, por ejemplo, la implementación en el modelo V se convertiría en la generación de código en el MBD.

La linealidad del modelo en V implica que no se puede avanzar con las siguientes fases sin haber completado la anterior. Esta restricción limita la flexibilidad de esta metodología, pero resulta ser un aspecto positivo porque no se pueden omitir elementos de algunas fases para pasar a las otras; aspectos que pueden ser críticos a largo plazo. Por este motivo el modelo en V puede no ser el flujo de trabajo ideal para algunas áreas, pero para el área espacial caracterizada por su rigidez, documentación extensa, requerimientos claros y fechas de entrega definidas; el modelo en V resulta ser un modelo adecuado.

Tanto el modelo en V como el MBD ponen gran importancia a la verificación y validación. La NASA define a la verificación el proceso que demuestra que un producto satisface los requerimientos con los que fue diseñado. En contraste, la validación se encarga de demostrar que el producto llevará a cabo su función destinada en el ambiente para el que fue diseñado [33]. Dentro del contexto de las simulaciones, la verificación son las pruebas con respecto a las especificaciones de

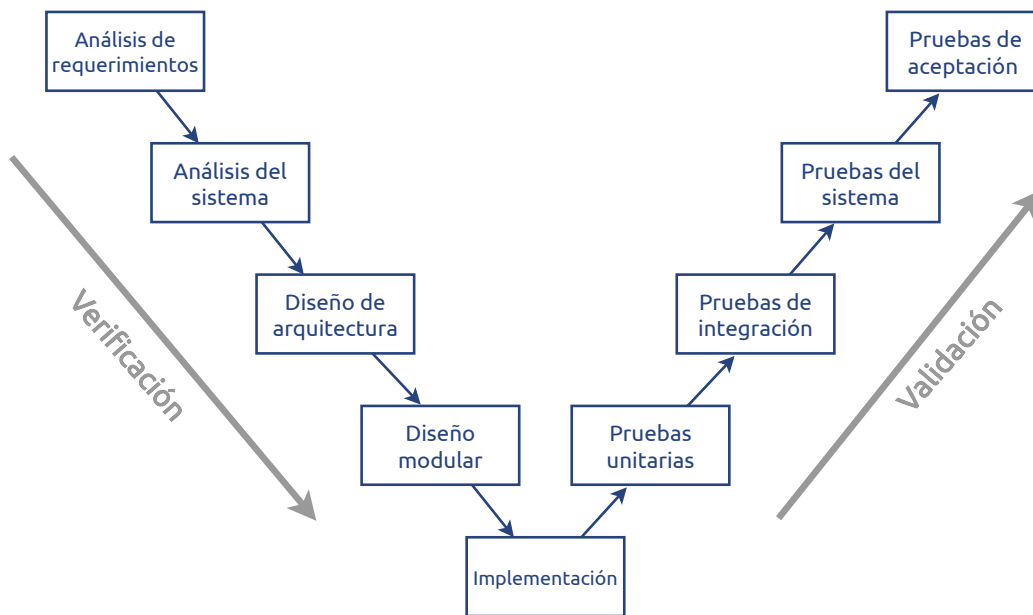


Figura 1.7: Flujo seguido por la metodología del modelo en V.

diseño del modelo; mientras que en la validación se hacen pruebas con respecto al referente, el sistema representado por el modelo.

La verificación y la validación son aspectos importantes que definirán el estado final de los productos y modelos desarrollados. Es a partir de estas técnicas con las que se pueden contestar las preguntas relacionadas a si satisfacen los requerimientos de seguridad y en el caso de los satélites, si tendrán la capacidad de cumplir su misión. En el MBD los procesos de verificación y validación se llevan a cabo durante las simulaciones, lo cual permite darle una mayor importancia a los aspectos a los que se dirigen estas pruebas. Mediante la verificación y validación continua, los modelos pueden modificarse con base a las necesidades de cada etapa y se pueden hacer modificaciones mayores con facilidad en caso de necesitarlas en etapas tardías, cosa que no es práctica en el diseño tradicional.

La continuidad en el esquema MBD en modelo en V significa entonces, que la validación y verificación se toman en consideración a lo largo de todo el proceso, en lugar de delimitarlo a una sola etapa. Una ventaja nacida de ello y que es clave en el uso de estas metodologías, es que los errores se pueden identificar en etapas tempranas del desarrollo y así evitar que se propaguen al avanzar. Al usarse en conjunto, el diseño basado en modelos y el modelo en V representan metodologías de trabajo que se complementan entre sí en la búsqueda de la minimización de riesgos y que además otorgan un flujo de trabajo que se alinea con las necesidades de las misiones CubeSat.

Si bien el modelo en V puede utilizarse para los distintos subsistemas que conforman el satélite, el MBD se asocia frecuentemente al subsistema ADCS. Los sensores, actuadores y el ambiente espacial son elementos que cuentan con modelos matemáticos y que por tanto pueden incorporarse en simulaciones. Las simulaciones,

sin embargo, no se limitan solo a los aspectos virtuales, puesto que dentro de las pruebas utilizadas para la verificación de sistemas ADCS existen dos familias:

- Simular el ambiente espacial mediante bobinas de Helmholtz, baleros de aire, simuladores de luz solar, etc. con el fin de que sean los sensores y hardware reales los que reciban y procesen las entradas enviadas por este ambiente simulado.
- Simular los sensores junto con otras piezas de hardware numéricamente, de tal manera que emulen las señales generadas por sus contrapartes reales y sean enviadas a una computadora de tiempo real [34].

El uso de uno u otro depende en gran medida de la infraestructura disponible para el equipo de desarrollo. Sin embargo, ambos toman elementos de los cuatro métodos generales de verificación que forman parte del MBD: Model-in-the-Loop, Software-in-the-Loop, Processor-in-the-Loop y Hardware-in-the-Loop.

1.5.1.2 TÉCNICAS DE VERIFICACIÓN EN LOS ADCS

Model-in-the-Loop (MIL). El primero de todos estos procesos. Es aquí en donde se lleva a cabo el desarrollo del ambiente de simulación. Los elementos de hardware o del ambiente se modelan en una planta, y se evalúa su interacción con el modelo del controlador. Esta etapa se enfoca en identificar las entradas y salidas clave del modelo, así como de analizar y evaluar la lógica del controlador. En otras palabras, esta es la etapa de verificación que se enfoca más en el lado teórico de los algoritmos de control.

Software-in-the-Loop (SIL). En esta técnica se genera código fuente a partir del modelo y se ejecuta dentro de la misma computadora. Con este enfoque se tiene una mayor flexibilidad para analizar el código de diferentes conceptos y prototipos, pero por otro lado estas pruebas no consideran los aspectos del hardware. El código es compilado y ejecutado en una arquitectura diferente a la del hardware final, por lo que aún se requerirán más pruebas para obtener resultados que satisfagan los requerimientos.

Las pruebas mediante SIL permiten detectar fallos en los algoritmos antes de insertar los factores relacionados con el hardware [35]. La principal ventaja de SIL se relaciona a este hecho, y es que al no requerir hardware, es un método de verificación que puede usarse en etapas tempranas del desarrollo. Además, cabe mencionar que Model-in-the-Loop puede agruparse como parte de «técnicas SIL» cuando se hace referencia a la diferencia de verificación basada en software contra aquella basada en hardware.

Software-in-the-Loop requiere que el modelo contenga tanto la planta como el controlador, de forma que el ambiente de simulación refleje aquellos elementos presentes en la realidad. Esto puede incluir el modelado de sensores. La inclusión de estos elementos no solo ayuda a mejorar los resultados, sino que también se sientan las bases para las pruebas de verificación subsecuentes como Hardware-in-the-Loop.

SIL permite hacer comparaciones entre dos controladores para hacer estimaciones sobre cuál tendrá un mejor rendimiento teórico. SIL puede emparejarse con las técnicas Monte Carlo para evaluar el rendimiento al someterse a diferentes condiciones iniciales, como la velocidad angular [36].

Processor-in-the-Loop (PIL). La primera técnica de verificación que incluye elementos de hardware. El código generado correspondiente al controlador se implementa en un procesador dedicado, mientras que el modelo de la planta permanece en el entorno de simulación [37]. Ambos elementos son interconectados mediante una interfaz. El principal objetivo de esta fase es evaluar la respuesta del controlador cuando se encuentra en el hardware dedicado. Las simulaciones no ocurren en tiempo real, por lo que solo es el primer paso en términos de las pruebas de hardware.

Hardware-in-the-Loop (HIL). En Hardware-in-the-Loop el modelo de la planta se ejecuta por medio de una computadora de tiempo real, la cual estará enlazada al procesador con el software del controlador. En ciertos casos, también participan otros elementos correspondientes a la planta mediante la fabricación de modelos físicos. HIL es considerado como un método eficiente para probar la funcionalidad de un software en un periodo de tiempo corto, comparado con otras técnicas de verificación de hardware.

Un ejemplo real de la aplicación de técnicas HIL en una misión real fue en el CubeSat MOVE-II de la Universidad Técnica de Múnich, lanzada en 2018. Fueron utilizadas para la verificación del ADCS y enfocándose hacia los presupuestos de energía. La simulación se implementó en Simulink. Para simular los sensores se incorporaron errores característicos en las señales simuladas [34].

Todas estas técnicas de verificación forman parte del diseño basado en modelos. Sus usos no se limitan a una disciplina específica, ya que pueden usarse en muchos tipos de proyectos, que pueden ir desde brazos robóticos hasta naves espaciales. En el caso particular de los satélites, enfoque de este proyecto, existen técnicas adicionales de verificación específicas a este dominio que es necesario mencionar debido a su relación subsecuente con los métodos previamente mencionados.

Una de dichas técnicas se presenta en las etapas tardías del proyecto y surge a raíz de la necesidad de hacer pruebas en donde la electrónica, el software y elementos de los otros subsistemas estén integrados: los FlatSats. Los FlatSats son definidos por la NASA como unidades de prueba en donde todos los componentes del CubeSat están interconectados, más no ensamblados dentro de la estructura del satélite. Su nombre hace referencia al acomodo de los componentes, los cuales comúnmente se montan en una superficie plana como una mesa. En la configuración FlatSat los subsistemas están interconectados y funcionan en conjunto, lo que permite hallar y corregir errores que no se manifiestan en las pruebas unitarias, además de evitar la necesidad de desarmar y volver a ensamblar el CubeSat de la estructura constantemente.

Otro procedimiento de prueba donde existe una integración de componentes es el *Day-in-the-life Testing* o DITL que consiste en hacer pruebas relacionadas con el día a día del satélite. Para ello se simula físicamente que la liberación del

CubeSat de su dispensador y se reporta que los temporizadores, mecanismos de transmisión, interruptores y otras acciones que se lleven a cabo tras la eyección funcionen adecuadamente.

Ambas técnicas son pruebas de integración y, por tanto, son de las pruebas que ocurren en las últimas etapas del proyecto. Por lo mismo, son ellas las que determinarán finalmente si el satélite será apto para cumplir su misión o no. El resultado aparentemente solamente recae en esta etapa de integración, pero se debe tomar en cuenta que esta no ocurre de forma aislada, puesto que los resultados de las pruebas anteriores van incluidos de forma implícita. El uso de las técnicas de verificación y validación durante todo el proceso de desarrollo según lo establecido en el diseño basado en modelos confiere entonces, un apoyo adicional cuyos resultados son evidentes durante las últimas etapas del proyecto.

El desarrollo del subsistema ADCS de un nanosatélite puede beneficiarse del uso del MBD y las técnicas de verificación in-the-Loop para el desarrollo general de algoritmos y componentes como actuadores, siendo uno de los aspectos más beneficiados el relacionado con el software de vuelo o (FSW).

1.5.2. Software de vuelo

El software de vuelo, también como FSW (*Flight Software*), se trata del conjunto de programas implementados en un sistema destinado al vuelo. Esto incluye aviones, cohetes, satélites y globos meteorológicos. En el contexto de los satélites, el software de vuelo puede estar implementado tanto en la computadora abordo como en los dispositivos periféricos como la instrumentación, en otras palabras, en la aviónica del satélite. La NASA destaca tres características clave que definen al software de vuelo:

- **Es un tipo de software embebido:** Diseñado en torno a un cierto hardware en particular con todas las limitaciones que esto implica, como en memoria.
- **Debe trabajar en tiempo real:** Debe ser capaz de otorgar una respuesta antes de un tiempo límite, de lo contrario puede traer consecuencias catastróficas que dependiendo de la misión pueden ir desde la pérdida de la nave de la espacial hasta la pérdida de vidas humanas. Los sistemas de tiempo real están en interacción continua con su ambiente, por lo que sus intervalos de respuesta hacia los estímulos están dictados por este [38].
- **Es crítico para la misión:** Debe mantener a la nave espacial segura ante anomalías y debe ser capaz de ser autónomo en ausencia de comunicación con el centro de control.

En otras palabras, el FSW debe para asegurar el cumplimiento de los objetivos aún con la presencia de imprevistos.

Las tareas del software de vuelo se relacionan a la administración de las actividades de la computadora a bordo, el procesamiento de datos, así como la salud y seguridad de la nave espacial [39]. En tiempos más recientes, las tareas del FSW también incluyen tareas relacionadas con sistemas autónomos y constelaciones.

El software de vuelo será entonces el que definirá el comportamiento del satélite en el espacio y la vía por la cual se podrán cumplir los objetivos de la misión, pero desde el otro punto de vista, la falla de un componente tan esencial se puede equiparar al fallo de la misión.

Una de las razones por las cuales se le pone tanto énfasis al software de vuelo se debe a la inhabilidad de acceder al hardware una vez que este ha sido lanzado, lo que mantiene al software de vuelo en un estado de alto riesgo. Como consecuencia, gran parte de los esfuerzos relacionados con la realización de pruebas, seguimiento de estándares de software y procesos de validación y verificación son enfocados a este aspecto.

Por estos motivos las misiones que no buscan desarrollar un software de vuelo propio pueden optar por usar soluciones preexistentes. En general, están diseñados para ser adecuados a cualquier misión. Algunas características deseables en la elección de un software de vuelo preexistente son un soporte a largo plazo por parte del desarrollador, la reusabilidad, la modularidad y la herencia de vuelo. A continuación, se presentan algunos de ellos y sus características.

cFS. Desarrollado en *Goddard Space Flight Center* de la NASA, este software de vuelo cuenta con herencia de vuelo. Utiliza una *Operating System Abstraction Layer (OSAL)*, la cual aísla el software embebido del sistema operativo en tiempo real [40]. Esto permite la ejecución de cFS en simuladores y computadores personales, además de permitir que las aplicaciones se ejecuten independientemente del sistema operativo y el hardware. Fue desarrollado para ser aplicable en diferentes misiones mediante la configuración de parámetros. Dentro de sus aplicaciones se encuentra la revisión de integridad de memoria, un administrador de archivos, planificador de actividades y comandos, y *housekeeping*. Nace a raíz del problema de partir de misiones con herencia de vuelo y tener que hacer modificaciones extensas y tediosas para adaptarlas a la misión presente. No solo es un conjunto de software, sino que también representa una metodología en el desarrollo de software de vuelo.

Algunas de las misiones que hicieron uso de este software de vuelo son *Lunar Reconnaissance Orbiter (LRO)*, *Lunar Atmosphere and Dust Environment Explorer (LADEE)*, *Global Precipitation Measurement Mission (GPM)* y *Magnetospheric Multiscale Mission (MMS)* [41].

NASA Operational Simulator for Small Satellites (NOS3). *Testbed* que únicamente hace uso de software, sin necesidad de hardware. Se enlaza fácilmente con cFS. Fue desarrollado para la misión STF-1, cuyo objetivo principal fue la demostración de la utilidad de simulaciones sin hardware. Simula hardware basado en COTS. Mediante la imitación de los protocolos que usarían los componentes de hardware al ser conectados al procesador, cFS no tiene la capacidad de discernir que dichos componentes son simulados [40].

Radiant Flight Software. Software de vuelo desarrollado por el laboratorio de dinámica espacial de la Universidad Estatal de Utah. Se dirige hacia «naves espaciales

pequeñas». Su arquitectura descentralizada entabla el uso de los distintos módulos que lo conforman. Requiere de un ambiente de tiempo real en Linux. No se limita a una selección de hardware específica. Entre sus módulos se encuentra el procesamiento de telemetría, administración de modos de operación y monitoreo de salud [42].

KubOS. Es un software de vuelo desarrollado con los CubeSats en mente. KubOs está diseñado a partir de una distribución de Linux que si bien no es inherentemente un sistema operativo en tiempo real (RTOS), llega a satisfacer la mayoría de requerimientos relacionados con los tiempos de respuesta [43]. La arquitectura de una misión que hace uso de KubOs está compuesta de procesos independientes llamados microservicios. La separación de estos componentes permite preservar la herencia de vuelo de los componentes no modificados durante una actualización. Esto permite al usuario enfocarse en las aplicaciones específicas a su misión. El código abierto está disponible en GitHub.

GERICOS. *GEneRIC Onboard Software*. Desarrollado por el Laboratorio de estudios espaciales y de instrumentación en astrofísica (LESIA) del Observatorio de París. Escrito en C++, cuenta con tres capas: CORE, BLOCKS y DRIVERS. CORE alberga funciones relacionadas con sistemas embebidos y de tiempo real como interrupciones. BLOCKS consiste en funciones genéricas frecuentes en software de vuelo como administración de telemetría que sirven como bloques para armar la funcionalidad deseada. No cuenta con código abierto y la documentación que contiene más información sobre su funcionamiento está escrita en francés. Cuenta con herencia de vuelo [39].

CORDET. Nace a partir de una iniciativa de ESA, la Agencia Espacial Europea. Dentro de los frameworks de software de vuelo más completos con respecto a documentación. Dentro de su herencia de vuelo se encuentra la CHEOPS (*CHaracterising ExOPlanets Satellite*), lanzado en diciembre de 2019. Cuenta con soporte de ESA. Reconocido por la estandarización CDSS. Implementado en C. El código abierto se encuentra disponible en GitHub.

Los software de vuelo preexistentes cuentan con el respaldo de la herencia de vuelo, pero independientemente de si la misión eligió uno de ellos o si se eligió desarrollar un software de vuelo propio, los desarrolladores aún tienen que someterlo a los procesos de validación y verificación. Uno particular para los FSW es el denominado *Test as you fly/Test like you fly*.

En esta metodología se busca reproducir tanto como sea posible las condiciones de operación del satélite con el fin de que las fases de integración y pruebas sean de alta fidelidad. Cómo indica su nombre, *Test as you fly*, consiste en realizar pruebas tratando al satélite como si estuviera volando. En ocasiones no se tiene acceso a los sensores por lo que lo se recurre a su emulación [34].

Mediante este enfoque la relación del software de vuelo con el MBD sea hace aparente. Los requerimientos del software de vuelo pueden ser validados y

verificados a través de los modelos generados mediante el MBD. Una vez comprobado su funcionamiento, el código que será embebido en el satélite puede ser automáticamente generado a partir de los modelos. Este código puede entonces ser integrado en un software de vuelo COTS como cFS y después ser evaluado mediante Processor-in-the-Loop y Hardware-in-the-Loop en tiempo real [44].

1.5.3. Software de análisis de misiones

Un aspecto que resulta de interés es la simulación del ambiente espacial, ya que este influirá en la dinámica rotacional del satélite y por consecuencia en el sistema de control de orientación. Y al igual que existen opciones comerciales para el software de vuelo, existen opciones comerciales para este tipo de software, comúnmente conocidos como herramientas de análisis de misiones. El objetivo de estos programas es proveer un ambiente de simulación con el cual generar pruebas de concepto de misiones y escenarios hipotéticos a analizar. Entre sus características generales se encuentran la simulación de la mecánica orbital, la generación de reportes y la visualización mediante animaciones, pero pueden tener características adicionales dependiendo el caso.

1.5.3.1 SOFTWARE EXISTENTE PARA EL ANÁLISIS DE MISIONES

STK. Systems ToolKit (STK) cuenta con una amplia selección de funcionalidades, las cuales incluyen simulaciones no solo para naves espaciales sino también vehículos como aeronaves y embarcaciones. Cuenta con una amplia selección de propagadores como SGP4 con los cuales es posible simular la órbita (Figura 1.8). Con el conocimiento de variables como la posición y velocidad inerciales del satélite, se pueden hacer uso de los modelos atmosféricos y astronómicos para calcular las magnitudes de las perturbaciones que afectan la orientación del satélite. Pero al ser un software con licencias, el acceso a estos modelos y funcionalidades es limitado en su versión gratuita.

FreeFlyer Astrodynamics Software. Freeflyer comparte varias características y funcionalidades con STK al ser también un software con licencia. FreeFlyer es un software diseñado para ser utilizado durante todo el ciclo de vida de la misión, incluyendo su diseño, análisis y operaciones. Sus modelos de dinámica de vuelo cuentan con una herencia de vuelo de más de 225 misiones. Cuenta con herramientas relacionadas al análisis de orientación, así como integración con interfaces externas [46].

Orbiter. Un software gratuito cuya principal característica es la visualización de gráficos 3D. Posee la habilidad de recrear misiones históricas como la llegada a la Luna. Los usos de este simulador se limitan a la demostración de conceptos de misiones y a fines educativos. Se le suele clasificar como un videojuego a pesar de que sus modelos sí consideran la física de los sistemas involucrados. Cuenta con

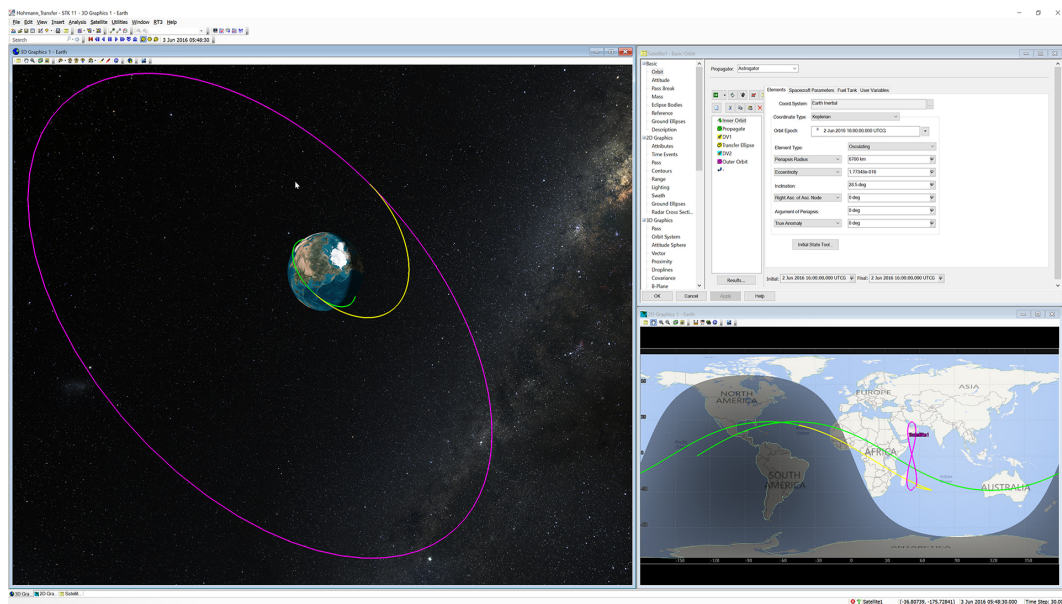


Figura 1.8: Ejemplo un análisis de órbita mediante los propagadores de STK [45].

módulos realizados por la comunidad, además de la habilidad de implementar módulos propios.

General Mission Analysis Tool (GMAT). Herramienta de código abierto desarrollada por la NASA en colaboración con contribuidores tanto públicos como privados. Fue usada como parte de las misiones *Lunar Reconnaissance Orbiter (LRO)* y *Transiting Exoplanet Survey Satellite (TESS)*. Se le ha dado soporte recientemente, siendo su última actualización en 2020 [47]. Se enfoca en el análisis de trayectorias de naves espaciales.

Nasa 42. Es un simulador de código abierto de la dinámica orbital y rotacional de naves espaciales desarrollado por NASA. Sus aplicaciones van desde las fases iniciales con la realización de prototipos hasta la verificación de código en las fases finales. Cuenta con una variedad de modelos ambientales que abarcan varias posiciones a lo largo del sistema solar. Cuenta con la capacidad de usar dinámica de cuerpo flexible. Usa el modelo atmosférico Jacchia-Roberts. No es *plug-and-play*, por lo que se requiere analizar del código para determinar qué cambios son necesarios para adaptarlo para la misión en la que se desea usar [36].

Al igual que en el caso de los software de vuelo, los desarrolladores de la misión tienen la elección de usar estas herramientas existentes o desarrollar las propias para una mejor adecuación a sus casos particulares. Relacionado con este punto, un enfoque usado por las universidades viene en la forma de módulos y librerías de lenguajes como Python.

Estos carecen de una interfaz gráfica, pero en contraste pueden interactuar fácilmente con otras librerías. Un ejemplo de este tipo de software es TU Delft Astrodynamic Toolbox, desarrollado por la universidad TU Delft. Algunas de las funcionalidades que ofrece es la selección de varios integradores, propagadores traslacionales y rotacionales, modelos ambientales y modelos de estimación. Ha sido usado en varios proyectos de investigación [48].

Por otro lado, varias universidades han desarrollado sus propios simuladores tanto para usos educativos como para el análisis de misiones reales. Estos simuladores son comúnmente desarrollados usando el paradigma de diseño basado en modelos, lo que soporta el argumento de que es un método efectivo para el desarrollo de misiones espaciales. A continuación, se hace mención de algunos ejemplos que hacen énfasis en la parte de verificación de ADCS.

1.5.3.2 SOFTWARE DE VERIFICACIÓN DESARROLLADO POR UNIVERSIDADES

Universidad de Aalborg. Un ambiente de simulación en Simulink fue desarrollado para la misión AAUSAT3 [49]. Incluye el propagador SGP4, modelos de eclipse y albedo, así como el modelo de campo magnético IGRF. Modelos de giroscopios, sensores de sol y magnetómetros, así como los algoritmos de determinación y control de orientación, también fueron implementados dentro de Simulink. Para la estimación se implementó un filtro de Kalman Unscented (UKF).

Universidad Politécnica del Estado de California. Simulaciones desarrolladas para la misión ExoCube [50]. Se modelaron los pares provenientes de perturbaciones ambientales como el gradiente de gravedad, la presión de radiación solar, el par aerodinámico y el par magnético. La configuración de esta misión incluye ruedas de reacción. Entre los algoritmos utilizados destaca el B-dot. No se realizó Hardware-in-the-Loop.

Instituto Superior Técnico de la Universidad de Lisboa. Desarrollado para ECOSat-III [51]. Modela las perturbaciones ambientales, así como los sensores y actuadores. Las simulaciones fueron implementadas en Simulink. La misión hizo uso de magnetómetros, sensores solares y giroscopios. Se hizo una comparación utilizando estimadores distintos como el filtro de Kalman extendido (EKF) y el UKF.

Estos son tan solo algunos ejemplos de herramientas de verificación para el software de vuelo desarrolladas por universidades. Todos ellos comparten el mismo objetivo más están diseñados de acuerdo a la configuración de sus respectivas misiones —en su mayoría CubeSats— y por tanto difieren en el tipo de sensores y actuadores usados, sus características técnicas, los modelos de estimación, entre otros aspectos. Pero en general todos hacen uso del diseño basado en modelos o un enfoque similar. Así mismo, Simulink es el ambiente de desarrollo predilecto por características como su modularidad y capacidad de integración con las técnicas in-the-Loop.

1.5.3.3 FUNCIONALIDADES RELACIONADAS EN SIMULINK

Además de las características anteriores, Simulink también cuenta con algunas funcionalidades nativas enfocadas al área aeroespacial que lo hacen una plataforma atractiva para solucionar el problema de la verificación de un ADCS. En particular, las nuevas versiones incluyen recursos enfocados al uso de CubeSats, donde se incluyen bloques, plantillas y ejemplos predefinidos. Por ejemplo, se incluyen bloques para simular el ambiente, los algoritmos y modos de control, la dinámica del CubeSat y la visualización mediante animaciones. El gran atractivo de estas funcionalidades es que el desarrollador puede hacer uso de estos bloques preestablecidos para acelerar el desarrollo y enfocarse únicamente en desarrollar aquellos que le conciernen directamente a la misión. Sin embargo, esto representa únicamente un punto de entrada, puesto que el modelo resultante debe complementarse con otras funcionalidades para que sea adecuado a las características de la misión.

Un ejemplo de como todos estos elementos trabajan en conjunto dentro de una simulación se puede encontrar dentro de los ejemplos incluidos por defecto en Simulink (Figura 1.9).

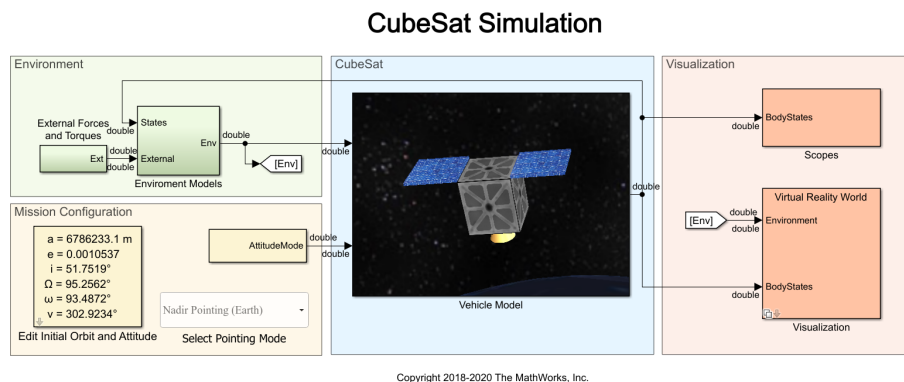


Figura 1.9: Simulación de un CubeSat como parte de los ejemplos predefinidos de Simulink.

Los bloques que componen esta simulación son en su mayoría aquellos que se encuentran dentro de la *Aerospace Toolbox*, pero su funcionalidad puede ser expandida a través de otros bloques, tanto de otras librerías como personalizados.

1.5.4. Gemelos digitales y hermanos digitales

Uno de los objetivos detrás de modelar un sistema mediante el MBD es poder consolidar los modelos con la implementación. Un concepto relacionado es el de los gemelos digitales. Un gemelo digital se define como una representación virtual de un componente físico, en donde el modelo virtual recibe datos del componente físico en todo momento, con el fin de que los estados sean los más parecidos posibles. Para esto los modelos pueden realizarse a partir de modelos físicos, o estadísticos basados

en datos. Los gemelos digitales están pensados para incluir el uso de modelos más allá de la fase de diseño, al incluirlos también en la fase de operaciones.

El uso de gemelos digitales no solo proporciona la información del estado actual del sistema, sino que también puede ser un medio de predecir como el sistema se comportará en el futuro. Otros beneficios incluyen el poder hacer detección de anomalías, mantenimiento predictivo y optimización de los procesos [52]. Además, el concepto de Hardware-in-the-Loop presenta una compatibilidad con el enfoque de los gemelos digitales [53]. En el caso de análisis de escenarios hipotéticos, en donde la simulación no se ejecuta en tiempo real, se usa el término hermano digital [53].

Si bien los bloques para CubeSats que vienen integrados en Simulink pueden tener los mismos resultados que bloques personalizados al hablar en términos de entradas y salidas, el enfoque detrás de ellos es diferente. Por una parte los bloques predefinidos tienen el objetivo de agilizar el desarrollo, mientras el de los bloques personalizados se asemeja al de los hermanos digitales al buscar que los algoritmos incluidos en el modelo sean los mismos que se incluirán dentro del satélite. Puesto que MATLAB es de código cerrado, no es posible contar con los detalles de los algoritmos que componen los bloques predefinidos, por lo que no será posible aprovechar con las ventajas de los hermanos digitales en cuanto a encontrar posibles condiciones de falla en ellos.

De este punto emerge la necesidad de poder realizar simulaciones para la verificación del ADCS de forma similar a la Figura 1.9, pero utilizando modelos que reflejen adecuadamente la contraparte física del satélite. Esta también es una de las razones por las cuales las universidades que han usado Simulink como plataforma de verificación han desarrollado sus propias implementaciones. Cabe destacar que una de las ventajas de usar Simulink para resolver este problema es que no excluye necesariamente el uso de bloques predefinidos, puesto que pueden ser eventualmente incluidos a manera de complementación.

1.6. Justificación

El desarrollo de un ADCS para un nanosatélite requiere de una herramienta que haga uso de simulaciones con fines de validación y verificación para reducir sus riesgos asociados. La implementación de modelos virtuales en dicha herramienta significará un medio de poder replicar los componentes del subsistema dentro de un modelo, de tal forma que bajo una única simulación se podrán considerar aspectos como la dinámica, los algoritmos y componentes que conforman al ADCS, así como el ambiente espacial.

El uso de la metodología de diseño basado en modelos para conseguirlo es un enfoque prometedor, puesto que sus ventajas lo han llevado a ser usado en otras misiones con éxito. El usarlo para la plataforma de simulación resultará benéfico no solo para el proyecto que se esté trabajando en el momento, sino que también para futuros proyectos. Debido a que el modelado es la principal forma de trabajo,

se tendrán distintas maneras de personalización en cuanto a la complejidad y forma de los modelos, lo que permitirá la capacidad de un prototipado concurrente a la implementación a diferencia del enfoque de diseño tradicional. El producto final siendo una plataforma de simulaciones que se adecue al flujo de trabajo del equipo desarrollo.

El MBD también permite la generación automática de código a partir de los modelos, siendo este aspecto en donde entra la estandarización de software. La generación de código sigue un único estilo en cuanto a la sintaxis, con lo cual se facilitará su legibilidad, así como la corrección de errores. Este código será eventualmente embebido en un microcontrolador, convirtiéndolo en software de vuelo. El proceso de verificación se hará por medio de SIL, para lo cual se requiere el adecuado funcionamiento del modelo principal. Con esto será posible hacer una evaluación del sistema en términos de requerimientos, lo cual será usado para añadir el factor de confiabilidad al diseño del ADCS.

Además, se busca que la herramienta resultante sea reutilizable y expandible. La primera característica hace referencia a su capacidad de poder adaptarse a misiones distintas a la que fue desarrollado sin modificaciones mayores. Para ello se busca un enfoque modular con el cual se permita insertar o quitar nuevos modelos o variantes de estos para afinar las características de la simulación. Por otro lado, ser expandible implica que características y funciones adicionales a las originales pueden ser adicionadas en el futuro de acuerdo con las necesidades específicas de la misión. Estas funcionalidades pueden ir desde los aspectos técnicos mediante una mayor gama de opciones de modelos ambientales hasta aspectos de la experiencia del usuario, como la habilidad de importar modelos 3D para las animaciones.

La herramienta será para uso interno, por lo que eventualmente no dependerá de licencias a diferencia de los software de análisis de misiones comerciales. Además, a diferencia de ellos, se centrará por el momento en el ADCS, facilitando la experiencia de usuario al no incluir funcionalidades ajenas a esta área. Sin embargo, para las primeras versiones sí se planea el uso de software comercial debido a que para simular un ADCS se requieren modelos que están fuera de su alcance y que pertenecen al dominio de otros subsistemas como es el caso de la órbita. La sustitución de esta funcionalidad a favor de modelos realizados internamente es una funcionalidad que se dejará para subsecuentes versiones.

La elaboración de este trabajo también es respaldada por los objetivos del Plan de órbita 2.0, ya que constituye una pieza de tecnología elaborada en México destinada a eliminar la dependencia hacia programas elaborados por otros países. El proceso de desarrollo implica, además, una formación de recursos humanos en cuanto a conocimiento del área aeroespacial.

2

Marco teórico

UNA FORMA de clasificar a los satélites es mediante su peso (véase la Tabla 1.1). Los satélites más pequeños pueden llegar a ser menores a 0.1 kg, mientras que los más grandes pueden llegar a superar los 1000 kg [2]. De forma general los satélites de mayor tamaño poseen funciones más complejas y sofisticadas, con la desventaja de que su costo aumenta exponencialmente, mientras que en los satélites pequeños la tendencia es opuesta. Sin embargo, una miniaturización extrema puede llevar igualmente a un aumento en los costos. Un equilibrio entre costos y funcionalidad se puede encontrar en los nanosatélites.

2.1. Los nanosatélites y sus características

Un nanosatélite es un satélite cuyo peso está dentro del rango de 1 a 10 kg. Estos satélites fueron desarrollados por primera vez durante la carrera espacial, como una forma de abordar el problema de la capacidad limitada de la carga útil de los vehículos de lanzamiento de la época. Pero tras la llegada del nuevo milenio, el paradigma de los nanosatélites cambió de una forma de resolver limitaciones a una forma de aprovechar activamente las ventajas que conllevan las características de su tamaño y desarrollo [54].

El tamaño de los nanosatélites les otorga una buena relación desempeño-masa, pero sin duda la mayor ventaja se encuentra en sus costos, los cuales son órdenes de magnitud menores que satélites de mayor tamaño. Ambas características llevaron al desarrollo del estándar CubeSat en 1999. Desde entonces el desarrollo de nanosatélites se ha popularizado en las universidades —la mayoría siguiendo este estándar— y su tendencia va en aumento.

En promedio la duración planeada de una misión de nanosatélites es de 8 meses, pero la duración real puede verse afectada ante la presencia de fallos en cualquiera de las etapas durante la vida útil. La mayoría de los nanosatélites se lanzan una órbita terrestre baja siendo la órbita predilecta la solar síncrona, aunque se han

reportado otros tipos de órbitas [54]. Estas características concuerdan con los tipos de objetivos que suelen tener los nanosatélites.

Un estudio en 2010 [54] hizo un análisis de los nanosatélites lanzados hasta el 1 de septiembre de 2009, y se encontró que las universidades permanecen siendo los principales desarrolladores de nanosatélites, así como de picosatélites. Dentro del mismo estudio se determinó que los principales objetivos de las misiones con satélites de este tipo son una mezcla de fines educativos y de demostración de tecnología, reduciéndose drásticamente a un 14 % si únicamente se considera la demostración de tecnología. Es gracias a las características de los nanosatélites mediante las cuales las universidades pueden integrar a sus alumnos en un proyecto espacial real, además de desarrollar técnicas experimentales, sin las desventajas asociadas al desarrollo de satélites de mayor tamaño.

En respuesta al aumento de la popularidad de los nanosatélites la tecnología relacionada también ha sufrido avances. Actualmente los desarrollos tecnológicos en cuanto a nanosatélites se encuentran en dos vertientes: las constelaciones y la miniaturización. En la primera el bajo costo de los nanosatélites permitirá no solamente hacer posible la creación de constelaciones, sino que también facilita el reemplazo de unidades defectuosas o que se acercan al final de su vida útil, todo gracias a que las características de los nanosatélites promueven la reemplazabilidad. En cuanto a la miniaturización, los avances en la electrónica permitirán encoger sistemas para que puedan ser implementables en un nanosatélite, de tal forma que se aumente la funcionalidad sin perder al mejor la relación desempeño-masa.

2.1.1. Subsistemas de un satélite

Los nanosatélites, al igual que todos los satélites, se componen por una serie de subsistemas. Cada subsistema se encarga de una función vital para el satélite como el monitoreo de salud o la protección contra el ambiente espacial. Algunos componentes de los subsistemas se muestran en la Figura 2.1. A pesar de realizar tareas distintas, todos los subsistemas actúan como un conjunto y las operaciones de uno pueden depender del funcionamiento de otro. Es por esta razón por la cual los subsistemas no se desarrollan de forma aislada, para incorporar las necesidades de los demás subsistemas como parte de los requerimientos.

A continuación se enumeran los subsistemas junto con su descripción general.

1. **Subsistema de potencia (EPS):** Provee a los componentes de los demás subsistemas la energía necesaria para que lleven a cabo sus operaciones durante la vida útil del satélite. El subsistema de potencia se encarga de almacenar, monitorear, controlar y generar la energía eléctrica [55]. En los CubeSats la fuente de energía primaria en la mayoría de los casos es mediante paneles solares [56].
2. **Subsistema de determinación y control de orientación (ADCS):** Subsistema encargado de estabilizar la orientación del satélite. Algunos satélites pueden carecer de un ADCS.

3. **Subsistema de carga útil:** Contiene a la carga útil de la misión, que es aquel hardware o software que interactúa con el exterior del satélite [55]. La carga útil depende de los objetivos, y puede tomar la forma de cámaras o instrumentos de medición.
4. **Subsistema de telecomunicaciones (COM):** Entabla la comunicación entre el satélite y la estación terrena mediante un *beacon*. Su principal responsabilidad es enviar telemetría y recibir comandos provenientes de Tierra. Este subsistema es el que se encarga de transmitir los datos provenientes de la carga útil a la estación terrena [56]. La telemetría sirve para monitorear la salud de los subsistemas del satélite [1].
5. **Subsistema de estructura:** Consiste en la estructura que alberga a los demás subsistemas y les brinda protección contra el ambiente espacial. La estructura debe contener mecanismos que permitan el despliegue desde los dispensadores. El estándar CubeSat especifica la forma y características que debe poseer.
6. **Subsistema de comando y manejo de datos (C&DH):** Ejecuta el sistema operativo en la computadora abordo (OBC) y controla los demás subsistemas mediante el manejo de comandos [57]. Este subsistema engloba a los subsistemas de OBC, telecomunicaciones y bus de datos [55].
7. **Subsistema de control térmico:** El ambiente espacial presenta variaciones de temperaturas extremas en el satélite, además de que este también está sujeto a la radiación térmica proveniente del sol a lo largo de su órbita. Este subsistema se encarga de asegurar que los componentes y demás subsistemas del satélite permanezcan dentro de los rangos de temperaturas óptimos de operación [56].
8. **Subsistema de control de órbita:** Las perturbaciones que afectan a un satélite modifican las características de su órbita a lo largo del tiempo, por lo que se suelen usar propulsores para su corrección y realización de maniobras. Debido a las limitaciones de dimensiones en los nanosatélites, este subsistema suele ser raramente incluido.

2.2. Subsistema de determinación y control de orientación

El sistema de determinación y control de orientación o ADCS (*Attitude Determination and Control System*), es el sujeto principal de este proyecto, razón por la cual se dará una descripción de los conceptos relevantes a él.

El objetivo general del ADCS se puede resumir como el mantener y estabilizar la orientación del satélite a lo largo de su vida útil. Esto se puede dividir en tres subtareas [58]:

- **Determinación:** Calcular la orientación de la nave relativa al marco de referencia inercial o algún punto de interés como la Tierra.

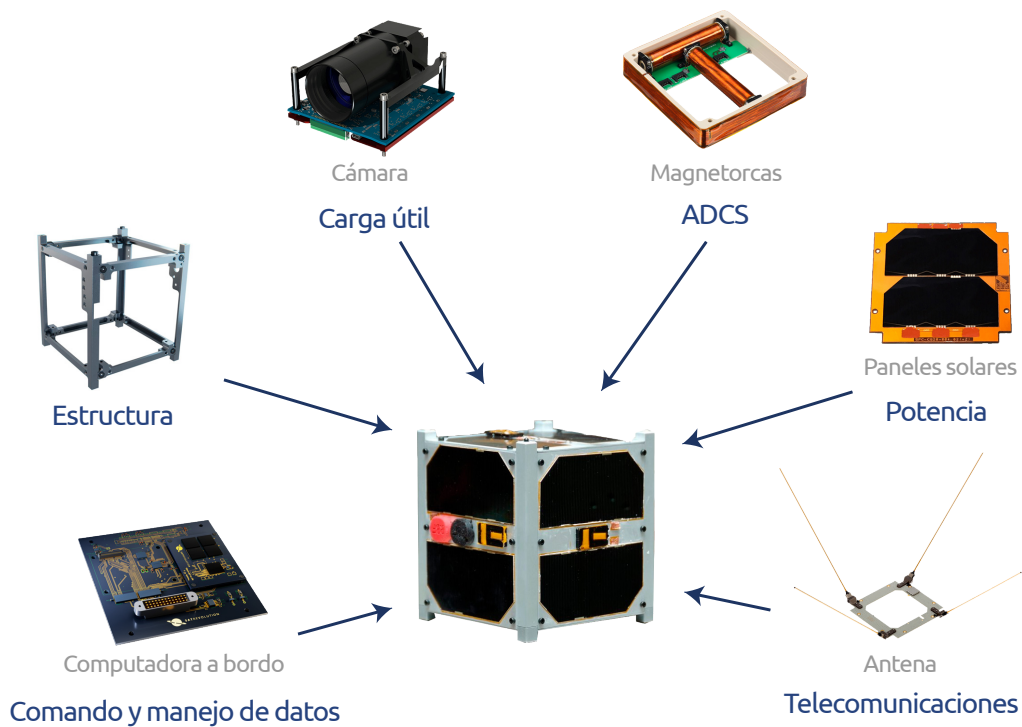


Figura 2.1: Subsistemas que componen a un satélite.

- **Control:** Orientar la nave espacial hacia la orientación deseada. Esto implica mantener dicha orientación una vez alcanzada, y la realización de maniobras para reorientar la nave hacia otra orientación.
- **Predicción:** Hacer uso de modelos matemáticos para propagar la orientación a partir de la orientación conocida para determinar el estado futuro.

En otras palabras el satélite necesita determinar su orientación en un momento dado, realizar acciones basadas en la información anterior para llegar a la orientación objetivo y predecir una nueva orientación con base en la información conocida.

Un concepto fundamental en el control de orientación son los pares de torsión, los cuales se pueden dividir en dos categorías:

- **Pares de perturbación:** Resultado del ambiente espacial. No pueden ser eliminados, por lo que se requieren pares de control provenientes del ADCS para contrarrestarlos. Ejemplos de este tipo de pares son el par aerodinámico y el par de presión de radiación solar.
- **Pares de control:** Se generan intencionalmente para controlar la orientación. Los actuadores son los encargados de proveer estos pares, por ejemplo por medio jets de gas o bobinas magnéticas [58].

El ADCS juega un papel importante en el satélite, ya que muchas tareas críticas para la misión requieren algún tipo de apuntamiento, por ejemplo, la alineación de las antenas hacia la Tierra. En otros casos la carga útil debe estar en línea de

visión con el objeto de estudio para la recolección de datos. Por lo tanto, fallas en el ADCS pueden llevar a una gran cantidad de problemas que pueden poner en riesgo el estado de la misión, como problemas en las telecomunicaciones, una deriva con respecto a la trayectoria de referencia o una alineación indebida de los paneles solares, resultando en una recarga ineficiente de la batería [59]. A pesar de esto, la necesidad de un sistema ADCS en el satélite está íntimamente ligada al tipo de misión. Los objetivos pueden requerir desde una estabilización y apuntamiento precisos hasta no requerirlos por completo.

En términos de la implementación, un sistema de control de orientación típico se conforma de tres elementos: sensores, el controlador y actuadores.

- El sensor de orientación localiza referencias como el sol o la Tierra para poder determinar la orientación.
- El controlador o ley de control determinan el par que se necesita generar para llegar a la orientación deseada.
- El actuador es el mecanismo que proporciona el par de control [58].

La relación entre estos elementos se muestra en la Figura 2.2, donde cabe resaltar que se tratan de los medios por los cuales cumplir las tareas de determinación, control y predicción.

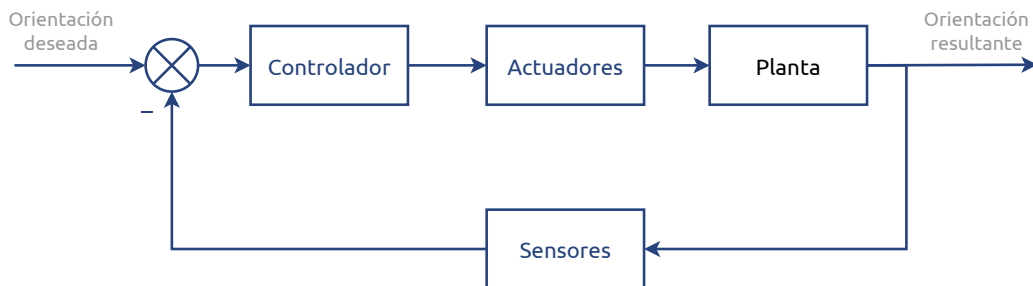


Figura 2.2: Elementos que conforman un ADCS y su relación.

En el caso de CubeSats la elección del método de control en cuanto a los elementos anteriores es más limitada con respecto a los satélites más grandes debido a las restricciones de tamaño y masa [60].

2.2.1. Sensores

Los sensores son el medio con el cual satélite recibe la información necesaria para determinar la orientación. Puesto que esto es un acto continuo, los sensores están diseñados para detectar entidades o referencias con alto nivel de disponibilidad. Por lo general son cuerpos celestes con características fácilmente diferenciables de otros cuerpos, como el sol y su luminosidad. Sensores comúnmente usados para la determinación de la orientación son los sensores solares, magnetómetros, giroscopios y seguidores de estrellas [31]. A continuación se presentan los tipos de sensores más comunes en nanosatélites.

2.2.1.1 SENSORES SOLARES

Miden la posición relativa del sol al satélite para determinar la orientación. El sol es el objeto más brillante en el espacio, lo que hace le otorga confiabilidad a su detección. Un ejemplo de un sensor solar se muestra en la Figura 2.3. Los sensores solares operan por medio del efecto fotoeléctrico, hacen uso de fotoceldas y tienen un bajo consumo energético. Cuando dos sensores se colocan perpendicularmente entre sí, la dirección del sol con respecto a los ejes de los sensores puede ser calculada dado el azimut y la elevación del vector solar.

Una ventaja de su uso es que la mayoría de los satélites usan energía solar, así que alinear los paneles hacia la dirección del sol resulta algo natural. Sin embargo, una desventaja surge del mismo fenómeno, ya que no operan en eclipse. Además el vector solar no es suficiente para poder determinar la orientación en su totalidad, por lo que se requiere de otro tipo de sensor adicional.

Los tipos de sensores solares son:

- **Sensor analógico:** Regresan una señal analógica.
 - **Baja precisión:** Miden la salida de corriente, la cual es proporcional al coseno del ángulo entre el sol y la normal a la fotocelda [61]. Utiliza un único fotodiodo. Su precisión disminuye en zenit llegando a 5° y aumentando a 3° en ángulos del rango 10-50°. En general su precisión es baja y son sensibles a señales de albedo y reflexiones proveniente de las partes del satélite.
 - **Fino:** Sensores que usan una apertura. La luz entra por medio de una rejilla creando un spot en el sensor y generando corrientes en cuatro diodos independientes (fotodiodos de cuadrantes). A partir de esta información se puede calcular la altitud del sol en dos direcciones.
- **Sensor digital:** Producen una señal discreta y usan procesamiento de señales para diferenciar entre luz solar directa y luz reflejada. Están compuestos por arreglos de fotoceldas, las cuales pueden estar organizadas en distintos patrones [61]. De acuerdo a las fotoceldas iluminadas se genera una señal digital, a partir de la cual se puede obtener el ángulo de incidencia [62].
- **Presencia del sol:** Da una señal escalonada sí el sol se encuentra dentro del rango de visión.

En los primeros CubeSats la combinación de la corriente generada por los fotodiodos en el satélite o en su caso, los paneles solares montados, daba una aproximación del vector solar. Este método aún es utilizado cuando solo se quiere una aproximación cruda [64].

2.2.1.2 MAGNETÓMETROS

Un magnetómetro determina la orientación de un satélite al medir el campo magnético en el marco fijo al cuerpo del satélite y compararlo al campo magnético



Figura 2.3: Ejemplo de un sensor solar digital [63].

del marco orbital [65]. Los magnetómetros miden la dirección y magnitud del campo magnético.

Dentro de sus ventajas se encuentran que son sensores ligeros y con bajo consumo de energía. Sin embargo, su principal desventaja es su relativa baja precisión comparada con otros sensores de orientación como los sensores de estrellas, además de que las lecturas pueden ser impactadas por la misma electrónica del satélite.

El uso de magnetómetros requiere la implementación de un modelo matemático del campo magnético para poder determinar la orientación, además de que se necesita de conocer otro vector de referencia, como es el caso del vector solar. Las mediciones del campo magnético siempre están disponibles debido a la persistencia del campo magnético terrestre.

2.2.1.3 GIROSCOPIOS

Instrumento que usa una masa que gira rápidamente para identificar y responder a cambios en la orientación inercial de su eje de giro.

Miden la rotación angular de una nave espacial en los tres ejes, midiendo el movimiento con respecto al marco inercial, por lo que los giroscopios son sensores inerciales. En los CubeSats es común el uso de giroscopios MEMS [66]. El uso de giroscopios permite propagar la orientación, lo cual es útil en métodos de determinación de orientación dinámicos como los filtros de Kalman.

Los giroscopios padecen de la generación de una deriva o *drift* con el tiempo, alterando las mediciones progresivamente, por lo que se deben tomar medidas para mitigarlo. Los giroscopios generalmente se dividen en dos clases: los *rate gyros* que devuelven tasas angulares y los *rate-integrating gyros* los cuales integran dichas tasas para devolver desplazamientos angulares [67].

2.2.1.4 SENSORES DE HORIZONTE

Usados para orientar la posición del satélite con respecto a la Tierra. Debido a que la Tierra es el segundo objeto más brillante para un satélite LEO y que esta permanece dentro de su campo de visión de forma constante, los sensores de horizonte son una

buena opción en cuanto a disponibilidad, más deben de ser más sofisticados que un simple sensor de detección de presencia. Por ello es necesaria la incorporación de un sistema de discriminación hacia el sol.

En ocasiones el horizonte no está bien definido para cuerpos con atmósfera o variaciones en la intensidad de su radiación, así que el diseño de los sensores se hace tomando esta perspectiva.

Sus partes están compuestas de un mecanismo de escaneo, un sistema óptico, un detector de radiancia y electrónica de procesamiento de señales. Se encuentran en modalidades como: fijo al cuerpo del satélite, junto a una rueda de momento, y en forma de rejilla. Una de sus principales ventajas es que operan independientemente de las condiciones de eclipse.

2.2.1.5 SENSORES DE ESTRELLAS

Miden las coordenadas de las estrellas en el marco fijo al satélite y las comparan con un catálogo de estrellas para obtener información de la orientación. Se componen de dos partes principales: una cámara digital con la cual toman imágenes que contengan estrellas, y una unidad de procesamiento.

El uso de sensores de estrellas es independiente a la órbita debido a la disponibilidad de las estrellas en el espacio. Son de los sensores más precisos, llegando al rango de arco-segundos. Son pesados, caros, requieren de mayor potencia y recursos de procesamiento, además de que son sensibles a la interferencia de objetos brillantes.

2.2.2. Sistemas de control de orientación

Los objetivos del ADCS se pueden resumir en dos: estabilizar la orientación del satélite con respecto a la orientación de referencia y la ejecución de maniobras de orientación [62]. Existen dos tipos de sistema de control en satélites para lograr lo anterior: sistemas pasivos y sistemas activos.

A grandes rasgos los sistemas de control pasivos hacen uso de los pares ambientales para mantener la orientación deseada, mientras que los sistemas de control activos involucran una constante realización de decisiones y operación de hardware para generar pares de control. La elección de uno u otro depende de la naturaleza, componentes y requerimientos de la misión.

2.2.2.1 PASIVOS

Los sistemas de control de orientación pasivos alcanzan la estabilización al interactuar con el ambiente. El ambiente espacial proporciona ciertos puntos de equilibrio para la dinámica y cinemática que pueden ser alcanzados si el satélite cumple con determinadas condiciones. El diseño del satélite en cuanto a su distribución de masa y forma busca facilitar la satisfacción de estas condiciones para la adquisición de los puntos de equilibrio. El satélite permanecerá en el punto de equilibrio una vez que lo haya alcanzado siempre y cuando no sea sujeto a perturbaciones de gran magnitud.

Una estrategia para corregir esto es el uso de un control de orientación activo, únicamente para cuando ocurran estas situaciones [62]. Este enfoque tiene la ventaja contra un sistema completamente activo en que si este último falla, la orientación permanecerá estable [68]. Debido a que la implementación de los sistemas pasivos está enfocada en la estructura, no hay un consumo energético.

Algunos ejemplos de control pasivo son:

Estabilización de giro (*spin stabilization*). El satélite se hace girar para que su vector de momento angular permanezca fijo en el espacio inercial. De esta forma el satélite gira de forma continua a través de un único eje, normalmente el de simetría. Tiene la ventaja de que no requiere de partes móviles, pero esta técnica no puede ser usada en todos los casos, ya que su uso está directamente ligado a los objetivos de la misión. La estabilización de giro puede realizarse de forma pasiva o activa. En el giro doble o *dual-spin* el satélite está segmentado de tal manera que una de las partes se mantenga fija mientras la otra gira para estabilizar. Este diseño es usado para situaciones que requieran el rastrear un objetivo [60].

Estabilización por gradiente de gravedad. La fuerza de gravedad actúa en un satélite asimétrico forzando el eje menor, aquel con el menor momento de inercia, a ser perpendicular al equipotencial de gravedad, por lo que el eje menor se alinearía hacia nadir. El requerimiento para que se lleve a cabo es que el par de gradiente de gravedad sea mayor que los demás pares ambientales.

Requieren de alguna forma de amortiguación para la nutación [58]. La implementación de este método requiere un cuidadoso análisis en la distribución de masa del satélite.

2.2.2.2 ACTIVOS

Los sistemas de control de orientación activos ejercen pares de forma intencionada para corregir y mantener la orientación del satélite. La principal ventaja del control activo con respecto al pasivo es que pueden tratar con orientaciones arbitrarias en los tres ejes, las cuales no están restringidas por las condiciones impuestas por el ambiente espacial y sus puntos de equilibrio limitados.

Este tipo de control hace uso de sensores de orientación y actuadores. Las mediciones de los sensores se utilizan para determinar la orientación del satélite y a partir del resultado los actuadores proveen los pares de control necesarios para disminuir la diferencia entre la orientación actual y la orientación deseada [68].

Los sistemas de control activos son más precisos, rápidos y otorgan una mayor flexibilidad que los pasivos. Sin embargo, su implementación requiere de una fuente de poder y de definir la lógica involucrada en los modos de operación. En algunos casos los sistemas activos requieren de consumibles que no pueden ser reemplazados una vez lanzado el satélite, como combustible [58].

Sistemas de expulsión de masa. Hacen uso de propulsores, los cuales se basan en la tercera ley de Newton para generar una aceleración en dirección opuesta a la

masa expulsada. Su costo es alto, requieren de un sistema de tuberías y su vida útil está limitada por el combustible abordo. Estas razones no hacen viable su uso en nanosatélites. Para un sistema estabilizado por tres ejes se requiere un mínimo de seis propulsores.

Estabilización por ruedas de reacción. Las ruedas de reacción son dispositivos que giran independientemente del resto de la nave espacial, por lo que tienen un momento angular propio. La rueda reacciona cuando un par actúa en el satélite a lo largo de un eje, absorbiendo el par mediante un cambio en la velocidad de giro, el cual ocurre en dirección contraria a la dirección deseada. Este tipo de estabilización permite mantener la orientación del satélite mediante el intercambio de momento angular entre el satélite y la rueda gracias a la conservación de momento. Es usado cuando los pares de perturbación son cíclicos. Se pierde el control de la orientación cuando la rueda alcanza su velocidad de saturación y se vuelve a retomar una vez que la velocidad disminuye. Por esta razón se requiere de algún mecanismo para disipar el momento adicional [60].

Magnetorcas. Las magnetorcas son un tipo de bobinas magnéticas, por lo que producen un campo magnético cuando corriente circula por ellas. Este campo magnético interactúa con el campo magnético terrestre para producir un par con el cual se puede controlar al satélite. Son sistemas ligeros, sin partes móviles o hardware complejo, lo que los hace opciones atractivas en nanosatélites. Sin embargo, requieren de un suministro de energía considerable. En consecuencia sus maniobras pueden llegar a ser lentas debido a la restricción de energía, y su operación depende del estado del campo magnético [58]. Las magnetorcas representan un sistema de control magnético activo.

2.3. Órbitas y marcos de referencia

Para modelar el sistema de determinación y control de orientación se necesitan conocer las características de la órbita del satélite, así como los marcos de referencia que se utilizarán para los análisis de dinámica. En esta sección se introducen conceptos de órbitas y marcos de referencia.

2.3.1. Órbitas

Las órbitas de los satélites suelen describirse como órbitas de Kepler. Las órbitas de Kepler son aquellas órbitas que toman la forma de cónicas bidimensionales en un espacio tridimensional y que se definen a partir de las tres leyes de Kepler. Las órbitas keplerianas se basan en un modelo ideal, ya que en la realidad las perturbaciones orbitales causan desviaciones y cambios en los parámetros keplerianos con el tiempo. Algunos ejemplos de originadores de perturbaciones son las fuerzas del sol, la

luna y la forma esferoide oblata de la Tierra. Las simulaciones realizadas mediante propagadores toman en cuenta estos fenómenos.

Dentro de las suposiciones que toman las órbitas keplerianas se encuentran [69]:

- La gravedad es la única fuerza que actúa en los dos cuerpos del sistema
- La masa del cuerpo primario es mucho mayor que la del cuerpo secundario
- El cuerpo primario es una esfera perfecta

En una órbita kepleriana la energía y el momento angular se conservan.

2.3.1.1 LEYES DE KEPLER

Primera ley. Si dos objetos en el espacio interactúan gravitacionalmente, cada uno describirá una órbita en la forma de una sección cónica, con el centro de masa en uno de los focos. Si los cuerpos están permanentemente asociados, las órbitas serán hipérbolas. Los círculos y las parábolas solo ocurren bajo excentricidades específicas, por lo que representan casos específicos dentro del rango de posibles excentricidades. Por esa razón no ocurren en la naturaleza a diferencia de las órbitas elípticas e hiperbólicas.

Segunda ley. Si dos objetos en el espacio interactúan gravitacionalmente (se muevan o no en una órbita elíptica cerrada), la línea que los une barrerá áreas iguales en intervalos de tiempo iguales. Es una reformulación de la conservación de momento angular debido a que es proporcional a la magnitud del vector de radio multiplicado por el componente perpendicular de la velocidad.

Tercera ley. Si dos objetos giran entre sí debido a su atracción gravitacional mutua, la suma de sus masas (M para el primario y m para el secundario) multiplicada por el cuadrado de su periodo de revolución mutua P es proporcional al cubo de la distancia media entre ellos. Esta distancia es el semieje mayor a .

La expresión matemática está dada por:

$$(m + M)P^2 = \frac{4\pi^2}{G}a^3, \quad (2.1)$$

donde G es la constante de gravitación universal. Solo aplica para órbitas elípticas y relaciona el periodo orbital con el semieje mayor. En varios casos se puede ignorar la masa del secundario.

2.3.1.2 ELEMENTOS KEPLERIANOS

Las órbitas keplerianas pueden describirse a partir de una serie de seis parámetros llamados los elementos keplerianos. Se definen con un tiempo de referencia o época, ya que existe una variación con respecto al tiempo de forma lenta pero significativa. Son usados en órbitas elípticas porque para órbitas hiperbólicas y circulares, parámetros como el perigeo no están bien definidos.

Los elementos keplerianos son:

1. **Semieje mayor** (a): Es el eje que pasa a través del centro de la órbita hasta su borde exterior, pasando por uno de los focos. Define el tamaño de la órbita.
2. **Excentricidad** (e): Define la forma de la órbita. Si $e = 0$ la órbita será circular. Si $0 < e < 1$, entonces la órbita será elíptica. Para $e = 1$ el resultado es una órbita parabólica. Por último, si $e > 1$, entonces la órbita será una hipérbola.
3. **Anomalía verdadera** (v): Ángulo que representa la posición del satélite en la órbita, medido desde el perigeo.
4. **Longitud de nodo ascendente** (RAAN, Ω): Ángulo medido desde el punto de Aries hasta el nodo ascendente, medido en sentido opuesto a las manecillas del reloj.
5. **Inclinación de la órbita** (i): Ángulo entre el plano de referencia (por ejemplo el plano ecuatorial de la Tierra) y el plano orbital.
6. **Argumento de perigeo** (ω): Ángulo entre el perigeo y nodo ascendente.

Se requieren dos elementos para describir la forma de la órbita, dos para la orientación del plano orbital en el espacio, uno para la orientación de la órbita dentro del plano orbital y uno para la posición del satélite [69]. Algunos de los elementos keplerianos se muestran en la Figura 2.4.

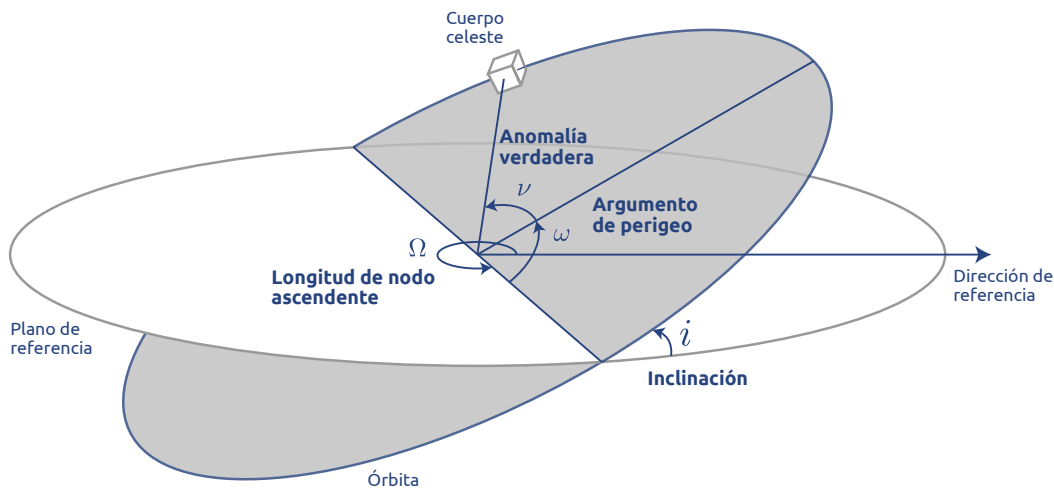


Figura 2.4: Elementos keplerianos que conforman una órbita, los cuales se resaltan en negrita. Adaptado de [70].

2.3.2. Marcos de referencia

Un marco de referencia es un sistema de coordenadas cartesianas en tres dimensiones, consistiendo de tres vectores unitarios perpendiculares entre sí [49]. Existen

diferentes formas de definir un marco de referencia. Por lo general, se elige que los primeros dos ejes sean colineales a vectores con un significado relevante al objeto de estudio, un ejemplo son las referencias astronómicas. Finalmente el eje restante, normalmente \hat{z} , se obtiene mediante la regla de la mano derecha.

Cuando un marco de referencia se mueve a una velocidad constante o es fijo, se tiene un marco de referencia inercial. Un marco es no inercial cuando tiene una aceleración asociada. En el caso específico de análisis de orientación se pueden encontrar dos tipos de marcos de referencia: los centrados en el espacio inercial y los centrados en la nave espacial.

Los marcos de referencia inerciales se usan principalmente para obtener vectores de referencia como el del campo magnético o para posicionar vectores a objetos vistos por la nave en el espacio inercial. Debido a que usan cuerpos celestes como puntos de referencia se necesita especificar una fecha o época para considerar efectos como la precesión. En cambio, los marcos de referencia centrados a la nave se usan para definir la orientación del hardware de determinación y control de orientación. Las medidas de los sensores se hacen a través de este tipo de marco.

A continuación se describen algunos tipos de marcos de referencia utilizados en los análisis de orientación.

2.3.2.1 MARCO DE REFERENCIA FIJO AL CUERPO

Este marco de referencia está fijo al cuerpo que se está analizando, en este caso el satélite. Los ejes se pueden ubicar de forma arbitraria, pero por lo general se eligen los ejes principales de inercia, con origen en el centro de masa del satélite. Como consecuencia los productos de inercia se reducen a cero, facilitando las simulaciones de dinámica. Un ejemplo de este marco de referencia se muestra en la Figura 2.5, tomando en cuenta que para poder definirlo con la convención anterior se necesita previamente la matriz de inercia del satélite para identificar los ejes principales.

Otra forma de definir los ejes es en torno a la geometría del satélite [61], por ejemplo, al asignar cada eje de forma colineal a las aristas de un CubeSat.

Puede ser inercial o no inercial dependiendo del movimiento del cuerpo. El marco de referencia fijo al cuerpo junto con el ECI juegan un papel importante para la determinación de orientación.

2.3.2.2 MARCO DE REFERENCIA FIJO AL SENSOR

El satélite recibe información del exterior mediante sus sensores, por lo que suele ser conveniente definir un marco de referencia centrado en ellos y después hacer las conversiones a marcos de referencia más generales para los análisis posteriores. Puesto que hay variaciones en el posicionamiento de los sensores de satélite a satélite, no existe una convención definida en cuanto a la dirección de los ejes. Un ejemplo de este marco de referencia es asignar dos ejes en dirección ortogonal y paralela respectivamente a la superficie de detección de un sensor [66]. El origen corresponde al centro del componente o de su superficie de detección.

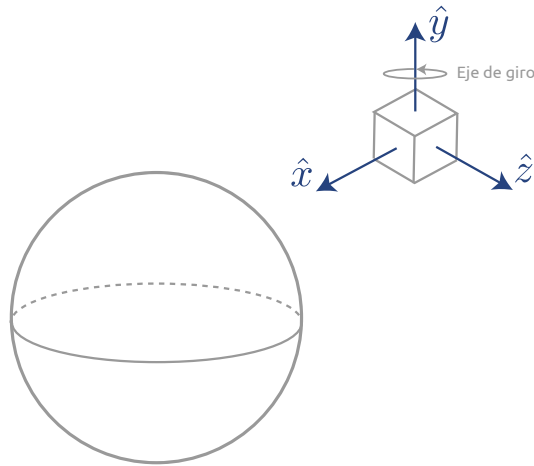


Figura 2.5: Definición de marco de referencia fijo al cuerpo del satélite.

En el caso de que los ejes elegidos para el componente se alineen con los del satélite, resulta más conveniente usar el marco de referencia fijo al cuerpo. De caso contrario se tendrían que hacer dos rotaciones en las simulaciones en lugar de una: $Sensor \rightleftharpoons Cuerpo \rightleftharpoons ECI$, en lugar de $Cuerpo \rightleftharpoons ECI$.

2.3.2.3 MARCO DE REFERENCIA ROLL-PITCH-YAW (RPY)

Este sistema mantiene su orientación relativa a la Tierra cuando la nave se desplaza en su órbita, por lo que se puede considerar un marco orbital.

Este marco de referencia está basado en secuencias de ángulos de Euler por lo que requiere de definir el orden de las secuencias, por ejemplo 123 [66]. Más información sobre los ángulos de Euler se presenta en la Sección 2.4.2.

El eje Yaw (correspondiente al eje \hat{z}) está dirigido hacia nadir, el eje de Pitch (\hat{y}) hacia la normal negativa de la órbita y el eje de Roll (\hat{x}) definido de acuerdo a la regla de la mano derecha [58]. El marco de referencia RPY se usa comúnmente para naves orientadas en dirección hacia la Tierra.

2.3.2.4 LVLH (LOCAL VERTICAL, LOCAL HORIZONTAL):

Tiene su origen en el centro de la nave espacial, y su definición varía según su contexto. Por ejemplo, en una definición el eje \hat{z} apunta hacia nadir y el \hat{x} hacia la dirección de la órbita (el vector de velocidad o la horizontal local), el eje \hat{y} es perpendicular al plano de la órbita [71]. Por otra parte en STK, el eje \hat{x} va en dirección del vector de posición, \hat{z} a la normal de la órbita y \hat{y} hacia el vector de velocidad [72]. Esta última convención se muestra en la Figura 2.6. Debido a que está rotando se considera un marco no inercial. Es un caso particular de un marco orbital y en este trabajo se utilizará este al marco de referencia LVLH como sinónimo del marco orbital.

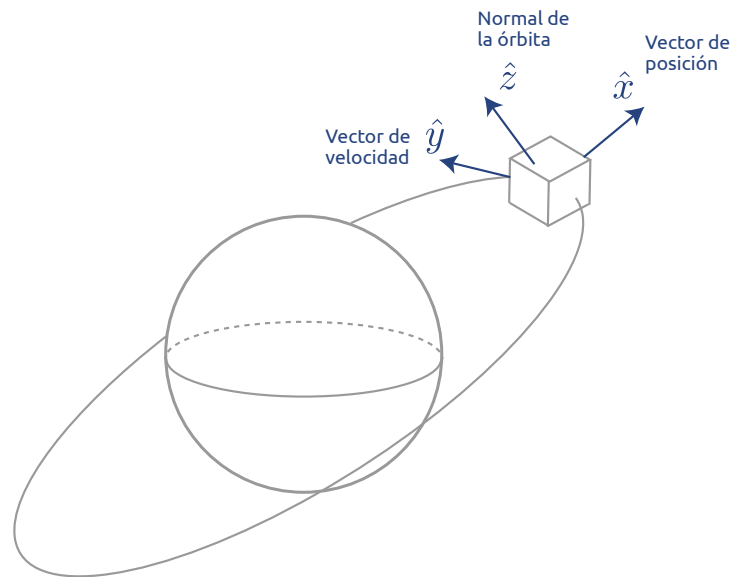


Figura 2.6: Definición del marco de referencia LVLH u orbital.

2.3.2.5 EARTH-CENTERED INERTIAL (ECI):

Es un marco de referencia que no rota, lo que quiere decir que los ejes permanecen fijos en el espacio inercial con el transcurso del tiempo. Tiene su origen en el centro de la Tierra. El eje \hat{x} apunta hacia el punto de Aries y el eje \hat{z} en la dirección del eje de rotación a través del polo norte geográfico (Figura 2.7).

Los efectos de la rotación de la Tierra en torno a su propio eje y en su órbita con respecto al sol resultan en aceleraciones centrípetas. Como consecuencia, el ECI no es un marco de referencia inercial perfecto. Aunque estos efectos pueden ser ignorados [49]. Bajo esta suposición, las leyes de Newton son válidas en este marco de referencia [73]. Además, es útil para tratar con las mediciones de sensores inerciales como las IMU [69].

2.3.2.6 EARTH-CENTERED EARTH FIXED (ECEF):

Tiene su origen en el centro de la Tierra. El eje \hat{x} apunta a la intersección del meridiano cero y el ecuador, es decir longitud 0° latitud 0° . El eje \hat{z} apunta en la dirección de rotación de la Tierra (Figura 2.8). Es un marco de referencia en rotación, por lo que está fijo relativo a la superficie de la Tierra y rota con la velocidad la misma velocidad. De esta forma un punto geográfico en la Tierra estará siempre representado por las mismas coordenadas en este marco de referencia. Es conveniente para definir puntos de interés localizados en la Tierra como las estaciones terrenas.

Otro marco de referencia, el geocéntrico, usa los mismos ejes, pero está expresado en coordenadas polares en función de la latitud, longitud y radio (LLR) [50].

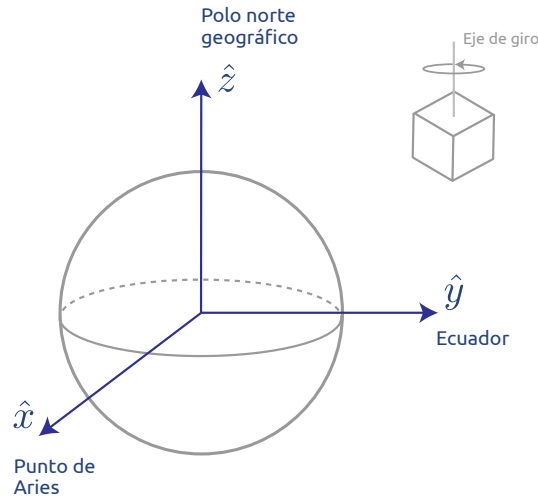


Figura 2.7: Definición del marco de referencia ECI.

2.4. Representaciones de orientación

Existen varias formas de expresar una orientación matemáticamente. Una de las principales diferencias entre las diferentes parametrizaciones se encuentra en el número de elementos necesarios para describir una orientación dada. Como mínimo se necesitan al menos tres elementos independientes, pero en ese caso se tendrán singularidades bajo ciertas condiciones que dependen de la parametrización.

Conforme sube el número de elementos se reduce la presencia de singularidades con la desventaja de tratar con elementos redundantes. Entonces, la elección de una parametrización de la orientación tiene el dilema de elegir entre la posibilidad de singularidades o de parámetros redundantes [51].

A continuación se describen las parametrizaciones usadas para la realización de los modelos: las matrices de rotación, los ángulos de Euler y los cuaterniones.

2.4.1. Matrices de rotación

Las matrices de rotación también conocidas como matrices de dirección de cosenos o DCM, son matrices que permiten pasar de una orientación a otra de tal forma que

$$\mathbf{x}_b = \mathbf{R}_{a2b} \mathbf{x}_a, \quad (2.2)$$

donde \mathbf{R}_{a2b} es la matriz de rotación que rota al vector \mathbf{x} del marco de referencia de a al marco de referencia de b . En tres dimensiones son matrices de 3×3 . El nombre de matrices de dirección de cosenos surge de los cosenos que componen a cada uno de los nueve elementos que la conforman. Estos describen la proyección de la base del marco de referencia objetivo en término de la base del marco de referencia inicial [50].

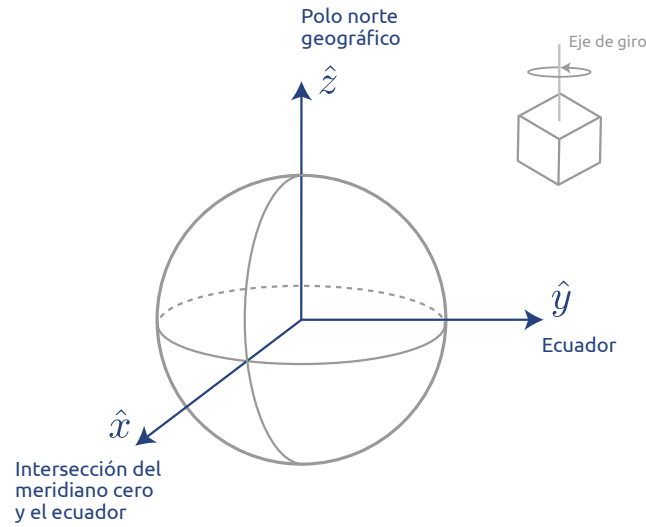


Figura 2.8: Definición del marco de referencia ECEF.

Las matrices de rotación deben satisfacer ciertas condiciones, puesto que forman parte del llamado grupo $SO(3)$. La matriz perteneciente al $SO(3)$ deberá ser ortogonal y tener un determinante de uno. Esto se resume en las siguientes ecuaciones:

$$\mathbf{R}^T \mathbf{R} = \mathbf{I}, \quad (2.3)$$

$$\det(\mathbf{R}) = 1, \quad (2.4)$$

donde \mathbf{I} es la matriz identidad [49].

La ortogonalidad tiene como consecuencia en que la transpuesta de una matriz de rotación sea igual a su inversa:

$$\mathbf{R}^T = \mathbf{R}^{-1}. \quad (2.5)$$

Es posible definir secuencias de rotaciones al multiplicar las DCM de las rotaciones involucradas, por ejemplo

$$\mathbf{R}_{a2c} = \mathbf{R}_{b2c} \mathbf{R}_{a2b}, \quad (2.6)$$

donde \mathbf{R}_{a2c} es la matriz que realiza la rotación de a a c directamente. Cabe mencionar que las matrices de rotación no son conmutativas.

La matriz de rotación cuenta con nueve parámetros, pero solo tres de ellos son independientes, representando los grados de libertad. Los seis parámetros restantes son restricciones. Estos parámetros redundantes eliminan la aparición de singularidades en la representación de matrices de rotación. Sin embargo, el uso de más elementos que los mínimos necesarios hace que sea más ineficiente computacionalmente hacer cálculos con esta representación. Además, los errores

numéricos pueden alterar la ortonormalidad de la matriz y las operaciones necesarias para restaurarla pueden poner una carga adicional en los recursos computacionales.

2.4.2. Ángulos de Euler

Los ángulos de Euler son considerados como la parametrización de orientación más sencilla de interpretar por humanos. Los ángulos de Euler describen una rotación en términos de tres elementos: Roll (ϕ), Pitch (θ) y Yaw (ψ) [66]. Estos son los mismos ángulos utilizados por el marco de referencia RPY.

Las matrices de rotación para rotaciones de un solo eje, en términos de los ángulos de Euler, se describen a continuación [68]:

$$\mathbf{R}_x(\phi) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\phi) & \sin(\phi) \\ 0 & -\sin(\phi) & \cos(\phi) \end{bmatrix}, \quad (2.7)$$

$$\mathbf{R}_y(\theta) = \begin{bmatrix} \cos(\theta) & 0 & -\sin(\theta) \\ 0 & 1 & 0 \\ \sin(\theta) & 0 & \cos(\theta) \end{bmatrix}, \quad (2.8)$$

$$\mathbf{R}_z(\psi) = \begin{bmatrix} \cos(\psi) & \sin(\psi) & 0 \\ -\sin(\psi) & \cos(\psi) & 0 \\ 0 & 0 & 1 \end{bmatrix}. \quad (2.9)$$

La multiplicación de las matrices anteriores para construir una secuencia de rotaciones tiene resultados distintos según el orden especificado. Razón por la cual se especifican secuencias como 123, en donde cada número corresponde a su respectivo eje. La secuencia no debe tener rotaciones sucesivas bajo el mismo eje aunque sí se puede repetir si no son contiguas. De esta forma 113 no es válida, pero 131 sí lo es. Existen un total de 12 secuencias de ángulos de Euler, las cuales se pueden clasificar en conjuntos simétricos y conjuntos asimétricos. En los simétricos dos de las tres rotaciones ocurren bajo un mismo eje, mientras que en los asimétricos las tres rotaciones ocurren en ejes diferentes [74].

Al ser una parametrización de tres elementos, los ángulos de Euler cuentan con el mínimo número de elementos para describir una orientación. Pero por esta misma razón está sujeta a singularidades, cuyo origen se puede encontrar en las funciones trigonométricas que utilizan. En los conjuntos asimétricos las singularidades ocurren cuando $\theta = \pm\frac{\pi}{2}$, mientras que los simétricos cuando $\theta = 0$ o $\theta = \pm\pi$.

2.4.3. Cuaterniones

Los cuaterniones son números hipercomplejos compuestos por un escalar q_0 y un vector $\hat{q} = (q_1, q_2, q_3)$ [75]. Un cuaternión \mathbf{q} se define como

$$\mathbf{q} = \begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \end{bmatrix} = q_0 + q_1i + q_2j + q_3k, \quad (2.10)$$

donde se deben satisfacer las siguientes condiciones:

$$i^2 = j^2 = k^2 = ijk = -1, \quad (2.11)$$

$$ij = k = -ji, \quad (2.12)$$

$$jk = i = -kj, \quad (2.13)$$

$$ki = j = -ik. \quad (2.14)$$

El concepto de los cuaterniones surge a partir del teorema de rotación de Euler, el cual establece que cualquier secuencia de rotaciones puede representarse como una única rotación a través de un eje conocido como el eje de Euler.

Los cuaterniones no involucran funciones trigonométricas en el cálculo de la cinemática, además de que proveen una forma conveniente de calcular rotaciones sucesivas. Una de las mayores ventajas de usar cuaterniones es el hecho de que carecen de singularidades.

2.4.3.1 PROPIEDADES DE LOS CUATERNIONES

Cuaternión unitario. Un cuaternión unitario se define como

$$\mathbf{q} = u \sin\left(\frac{\theta}{2}\right) + \cos\left(\frac{\theta}{2}\right), \quad (2.15)$$

donde u representa el eje de Euler y θ el ángulo de rotación a través de este eje.

Norma del cuaternión. La norma de un cuaternión es

$$|\mathbf{q}| = \sqrt{\mathbf{q}^* \mathbf{q}} = \sqrt{q_0^2 + q_1^2 + q_2^2 + q_3^2}, \quad (2.16)$$

donde \mathbf{q}^* es el conjugado del cuaternión.

Conjugado del cuaternión. En el conjugado de un cuaternión se invierten los signos de los componentes de la parte vectorial; de tal forma que

$$\mathbf{q}^* = q_0 - q_1i - q_2j - q_3k. \quad (2.17)$$

Suma. La suma de los cuaterniones sigue las reglas de asociatividad y conmutatividad mediante la adición de sus componentes. Sean \mathbf{q} y \mathbf{p} dos cuaterniones,

entonces

$$\mathbf{q} + \mathbf{p} = (q_0 + p_0) + i(q_1 + p_1) + j(q_2 + p_2) + k(q_3 + p_3). \quad (2.18)$$

Multiplicación. La multiplicación mediante cuaterniones considera las condiciones dadas por las ecuaciones (2.11) a (2.14), y puede expresarse como

$$\mathbf{p} \otimes \mathbf{q} = \mathbf{r} = r_0 + r_1i + r_2j + r_3k, \quad (2.19)$$

$$r_0 = p_0q_0 - p_1q_1 - p_2q_2 - p_3q_3, \quad (2.20)$$

$$r_1 = p_0q_1 + p_1q_0 + p_2q_3 - p_3q_2, \quad (2.21)$$

$$r_2 = p_0q_2 - p_1q_3 + p_2q_0 + p_3q_1, \quad (2.22)$$

$$r_3 = p_0q_3 + p_1q_2 - p_2q_1 + p_3q_0. \quad (2.23)$$

Alternativamente la forma matricial

$$\begin{bmatrix} r_1 \\ r_2 \\ r_3 \\ r_4 \end{bmatrix} = \begin{bmatrix} p_0 & -p_1 & -p_2 & -p_3 \\ p_1 & p_0 & -p_3 & p_2 \\ p_2 & p_3 & p_0 & -p_1 \\ p_3 & -p_2 & p_1 & p_0 \end{bmatrix} \begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \end{bmatrix}, \quad (2.24)$$

resulta más conveniente en la implementación

Rotaciones. Para realizar una rotación por medio de cuaterniones se requiere que el vector v a ser rotado tenga el mismo número de dimensiones que un cuaternión. Esto se logra al agregar un cero en el lugar correspondiente al escalar del cuaternión, en este caso q_0 , por lo que se define al cuaternión ν como $\nu = [0 \ v]$.

El resultado de rotar ν a través del cuaternión \mathbf{q} con un ángulo θ se puede definir entonces mediante el cuaternión \mathbf{w} :

$$\mathbf{w} = \mathbf{q} \otimes \nu \otimes \mathbf{q}^*. \quad (2.25)$$

Alternativamente se tiene caso de que el marco de referencia sea el que rota relativo al vector v . Entonces se intercambia el orden de \mathbf{q} y \mathbf{q}^* [76]:

$$\mathbf{w} = \mathbf{q}^* \otimes \nu \otimes \mathbf{q}. \quad (2.26)$$

Inverso del cuaternión. El inverso del cuaternión se define como

$$\mathbf{q}^{-1} = \frac{\mathbf{q}^*}{|\mathbf{q}|^2}. \quad (2.27)$$

Considerando un cuaternión unitario con $|\mathbf{q}| = 1$, la expresión anterior se reduce a

$$\mathbf{q}^{-1} = \mathbf{q}^*. \quad (2.28)$$

El inverso de un cuaternión unitario es por lo tanto su conjugado, con lo cual se puede explicar el comportamiento de las Ecuaciones (2.25) y (2.26). Esencialmente, la inversa representa la misma rotación que su cuaternión inicial pero con sentido opuesto.

Conversión a matriz de rotación. Finalmente, la relación entre las matrices de rotación y los cuaterniones se da mediante [75]

$$\mathbf{R}(\mathbf{q}) = \begin{bmatrix} 2q_0^2 - 1 + 2q_1^2 & 2(q_1q_2 - q_0q_3) & 2(q_1q_3 + q_0q_2) \\ 2(q_1q_2 + q_0q_3) & 2q_0^2 - 1 + 2q_2^2 & 2(q_2q_3 - q_0q_1) \\ 2(q_1q_3 - q_0q_2) & 2(q_2q_3 + q_0q_1) & 2q_0^2 - 1 + 2q_3^2 \end{bmatrix}. \quad (2.29)$$

En el contexto de los cuaterniones como operadores de rotación, la norma se restringe a $|\mathbf{q}| = 1$. Las operaciones computacionales con cuaterniones suelen introducir un factor de error que altera esta restricción, en especial con las integraciones. Por esta razón se realizan normalizaciones en cada iteración.

2.5. Dinámica y cinemática del satélite

Las ecuaciones de cinemática y dinámica de un cuerpo rígido permiten modelar el comportamiento del satélite en el espacio. Es a partir de ellas con la cual se encuentra la respuesta de como el ambiente espacial afectará la orientación del satélite.

La ecuación de la cinemática se utiliza para la propagación de estado de la orientación. Solo tiene como entrada la velocidad angular neta y no los factores que le dieron origen. En cambio la ecuación de la dinámica toma como entrada los pares presentes en el satélite, tanto de perturbación como de control, y puede integrarse para determinar las velocidades angulares, más no consideran directamente la orientación.

Tanto la ecuación de dinámica como la de cinemática deben usarse en conjunto para resolver el problema de modelar la orientación del satélite y sus cambios a lo largo de la órbita, para así proceder a diseñar el ADCS.

2.5.1. Cinemática

La ecuación de cinemática se define como

$$\frac{d}{dt} \mathbf{q} = \frac{1}{2} \Omega \mathbf{q}, \quad (2.30)$$

donde Ω es una matriz antisimétrica con los componentes de la velocidad angular $\boldsymbol{\omega}$ del satélite en el marco de referencia fijo al cuerpo del satélite. La construcción de Ω depende de la convención usada para definir los cuaterniones [77] y en este caso está dada por

$$\Omega = \begin{bmatrix} 0 & -\omega_x & -\omega_y & -\omega_z \\ \omega_x & 0 & \omega_z & -\omega_y \\ \omega_y & -\omega_z & 0 & \omega_x \\ \omega_z & \omega_y & -\omega_x & 0 \end{bmatrix}. \quad (2.31)$$

De la Ecuación (2.30) se observa que la descripción de la cinemática únicamente relaciona la orientación con la velocidad angular.

Al hacer uso de cuaterniones se evita la presencia de singularidades en los cálculos por lo que se evita la necesidad de agregar excepciones que manejen estos casos en la hora de la implementación.

2.5.2. Dinámica

Un satélite se puede modelar como un cuerpo rígido, en donde se asume que no hay deformaciones y que la posición de las partículas del sistema permanecen en la misma posición unas con respecto a otras. Para un cuerpo rígido el momento angular L con respecto al marco de referencia fijo al cuerpo es proporcional a su velocidad angular:

$$L = I\boldsymbol{\omega}. \quad (2.32)$$

El tensor de inercia I es una propiedad del cuerpo relacionada con la distribución de su masa. Por lo tanto es influenciada por la geometría del cuerpo y de los materiales de los que esté hecho. La expresión para el tensor de inercia consta de nueve elementos [78]:

$$I = \begin{bmatrix} I_{xx} & I_{xy} & I_{xz} \\ I_{yx} & I_{yy} & I_{yz} \\ I_{zx} & I_{zy} & I_{zz} \end{bmatrix}. \quad (2.33)$$

Los elementos fuera de la diagonal son llamados los productos de inercia. Si el cuerpo rígido es perfectamente simétrico, el tensor de inercia adquiere la forma de una matriz diagonal. En este caso el momento angular para cada eje consiste en la velocidad angular para cada eje, multiplicada por una escalar. Al elegir el marco de referencia fijo al cuerpo los productos de inercia desaparecen, resultando en el caso anterior sin importar la forma del satélite. Esto simplifica los cálculos subsecuentes, lo que explica la elección de este marco de referencia.

Otra expresión del momento angular se relaciona sobre su tasa de cambio, la cual está dada por el par neto τ que actúa sobre el cuerpo:

$$\dot{L} = \tau, \quad (2.34)$$

de tal forma que un $\tau \neq 0$ aplicado sobre el cuerpo ocasionará un cambio en el momento angular, lo cual tendrá en consecuencia un cambio en la velocidad angular.

La ecuación que describe la dinámica de un cuerpo rígido es conocida como la ecuación de cuerpo rígido de Euler. Se puede derivar a partir de las ecuaciones (2.32) y (2.34), por lo que involucra el momento angular, la velocidad angular y el par neto. En su forma general se define como

$$\frac{d}{dt} I\omega = I\omega \times \omega + \tau. \quad (2.35)$$

En el contexto de los satélites el par neto τ es la suma de los pares de perturbación y los de control. Es decir, los pares que las perturbaciones del ambiente espacial generan sobre el satélite y los pares generados por el sistema de control de orientación del satélite para contrarrestarlos.

En varios satélites no ocurre una redistribución de masa con respecto al tiempo. Eso se debe a que pueden no contar con un sistema de propulsión con un flujo de masa o que no tengan componentes flexibles como paneles desplegables. Para dichas situaciones el tensor de inercia permanece constante. La ecuación que describe estos casos es entonces [79]:

$$\frac{d}{dt} \omega = I^{-1}(I\omega \times \omega + \tau). \quad (2.36)$$

3

Metodología

LA HERRAMIENTA propuesta para este trabajo se desarrollará por medio del ambiente de MATLAB, específicamente en Simulink. Además de los antecedentes y funcionalidades mencionadas en la Sección 1.5.3, la elección de Simulink también otorga ventajas como el permitir enfocarse directamente en el modelado y no en la implementación de utilerías para conseguirlo. Siguiendo esta línea, en Simulink no es necesario la implementación de métodos de integración o demás métodos numéricos porque son incorporados automáticamente.

Mediante el enfoque del diseño basado en modelos a cada entidad que compone al satélite —sensores, actuadores, algoritmos, así como el mismo satélite—, le corresponderá un modelo matemático. En Simulink cada uno de estos modelos se manifiestan como bloques, los cuales tienen una separación lógica entre ellos y únicamente se comunican con los demás por medio de entradas y salidas. Las conexiones entre los distintos bloques permiten modelar subsistemas más complejos, que en este caso, serían las entidades que conforman el subsistema de determinación y control de orientación. Las características anteriores corresponden a las técnicas de SIL, específicamente Model-in-the-Loop, en donde se generarán los modelos que serán usados en las etapas posteriores.

El flujo general de la metodología que se seguirá para la realización del modelo virtual, siguiendo los principios del diseño basado en modelos y el modelo en V, se presenta en la Figura 3.1. Los bloques resaltados corresponden a los pasos abordados por este trabajo, mientras que los demás representan los pasos que siguen en el desarrollo de un ADCS.

3.1. Organización del modelo virtual

Los modelos que se elaborarán en la fase de Model-in-the-Loop se agruparán por medio de categorías de acuerdo al propósito de cada modelo y su relación con el ADCS. Estos grupos son lo que en Simulink se denominan como subsistemas. Los subsistemas permiten abstraer aún más la simulación al reducir el número de señales

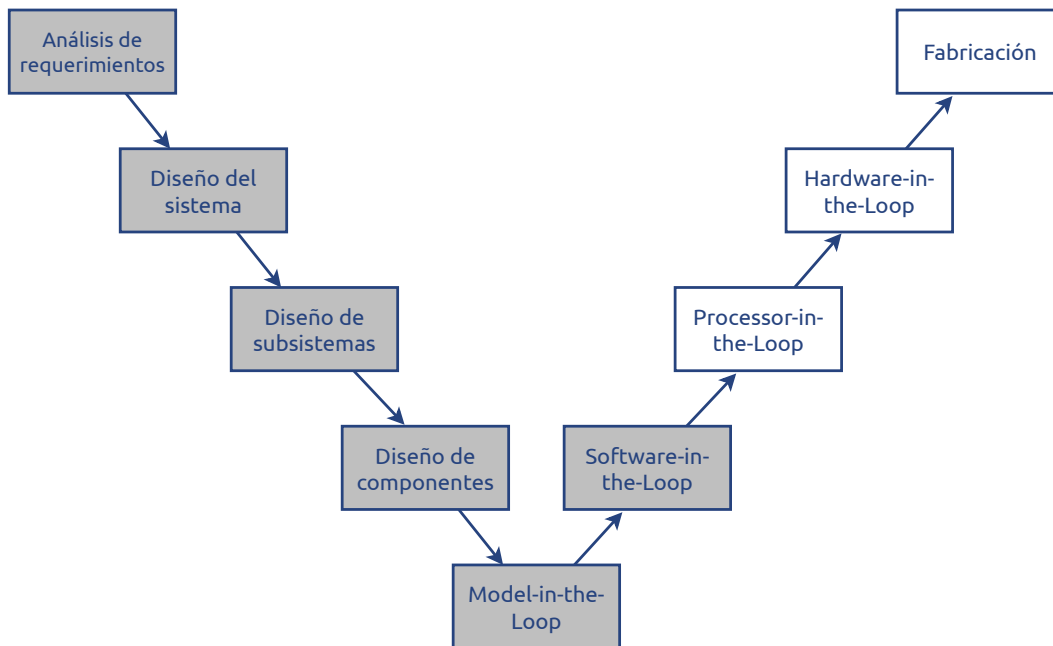


Figura 3.1: Diagrama representando el flujo de desarrollo de un ADCS siguiendo el modelo en V y el MBD. Las partes que abarca la herramienta de simulación son resaltadas.

de entrada y de salida a un alto nivel. Los subsistemas presentes en el modelo virtual se describen a continuación y la relación entre ellos se presenta en la Figura 3.2.

- **Dinámica y cinemática:** Define y propaga el estado verdadero del satélite (orientación y velocidad angular) basado en el tensor de inercia del satélite y los pares a los que este está sujeto: la suma de los pares producidos por el modelo de perturbaciones y de actuadores.
- **Modelo de perturbaciones:** Contiene los pares de perturbación del ambiente que tienen un efecto en la orientación. Se considera el par aerodinámico, por presión de radiación solar y gravitacional.
- **Determinación de orientación:** Determina la orientación del satélite a partir de las mediciones de los sensores simulados y los algoritmos que calculan sus contrapartes inerciales. En el mejor de los casos la orientación determinada sería igual al estado verdadero, calculado por el modelo de cinemática.
- **Modelo orbital y ambiental:** Devuelve los parámetros que dependen de la órbita como la posición del satélite, así como parámetros del ambiente que no dependen de la orientación como el campo magnético inercial. Este subsistema cumple la función de un propagador, además de que constituye la fuente de las señales de estado verdadero que utilizarán los sensores.
- **Modelo temporal:** Proporciona la fecha inicial. Además lleva un seguimiento del tiempo transcurrido mediante un contador, el cual puede usarse para determinar la fecha en un momento dado.

- **Estimación de estado:** Utiliza un filtro de Kalman para procesar las mediciones de los sensores junto con un modelo de extrapolación para generar una estimación sobre el estado verdadero del sistema.
- **Algoritmos de control:** Definen el modo de control del satélite, así como la cantidad de voltaje que se le debe proporcionar a los actuadores.
- **Actuadores:** Modelo de los actuadores, los cuales imponen restricciones a la señal producida por la ley de control.
- **Sensores:** Modelo de los sensores que imponen restricciones en las señales medidas como al agregar ruido.
- **Visualización:** Genera animaciones a partir de los resultados de las simulaciones. Véase la Sección 3.4 para mayor información.

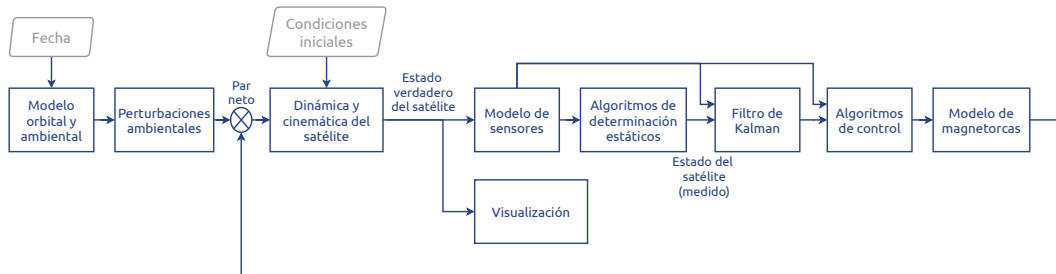


Figura 3.2: Relación entre los distintos bloques de Simulink en el modelo virtual.

Durante la ejecución del modelo se manejarán dos tipos de señales. Por una parte se tendrán las señales que representan el estado verdadero del sistema y por la otra las señales que representan las mediciones realizadas por los modelos de los sensores. Estas últimas serán sujetas a ruido, muestreo y otras limitaciones relacionadas a las características de los sensores con el fin de simular el tipo de señales que recibirían los algoritmos como entrada en la realidad.

Como consecuencia de lo anterior, los modelos del estado verdadero proporcionan señales continuas, mientras que los modelos de los algoritmos regresarán señales discretas basadas en el tiempo de muestreo definido. Un ejemplo de modelo de estado verdadero es el subsistema de dinámica y cinemática, mientras que en la otra categoría se encuentran el subsistema de sensores y el subsistema de actuadores, el cual cuenta con un enfoque similar.

El objetivo del subsistema de estimación de estado es el poder estimar el estado verdadero a través de la información limitada otorgada por las señales devueltas por los sensores.

3.2. Configuración del nanosatélite simulado

Cada uno de los subsistemas de la Figura 3.2 consiste en la abstracción de los componentes del ADCS o de su ambiente. En el primer caso se cuenta con la capacidad de implementar variantes en cuanto a la configuración del nanosatélite

simulado, pudiendo tener modelos para distintos algoritmos, sensores y actuadores. En otras palabras, la estructura del modelo virtual propuesto refleja aquella de un ADCS genérico y como resultado, será posible seleccionar la configuración del nanosatélite que se quiera simular, lo cual será de utilidad en la fase de exploración de concepto.

Para este trabajo el enfoque se dirige hacia sistemas de control de orientación magnéticos activos, por lo que la actuación del satélite simulado se hará por medio de magnetorcas. Las magnetorcas requieren de conocer el campo magnético, por lo que se requieren magnetómetros. Sin embargo, el vector de campo magnético es insuficiente para determinar la orientación, así que se requiere un tipo adicional de sensores. Los sensores solares suelen ser usados frecuentemente junto a los magnetómetros para afrontar este problema, razón por la cual se incluirán dentro del nanosatélite simulado. La elección de algoritmos de determinación y control de orientación está fuertemente ligada a la configuración del satélite y como resultado se considerarán algoritmos basándose en la configuración descrita anteriormente.

3.2.1. Evaluación de requerimientos

El cumplimiento de los requerimientos es el medio por el cual se comprobará si el ADCS cumplirá con sus objetivos propuestos, por lo que es necesario someterlo a un proceso de verificación. El modelo virtual tendrá la capacidad de evaluar las respuestas producidas por el nanosatélite simulado contra sus requerimientos asociados. Esto será posible mediante el uso de *Simulink Test*, en donde se proveen herramientas para generar pruebas unitarias y de integración para todos los subsistemas que componen al modelo virtual. El resultado de dichas pruebas podrá determinar si el requerimiento se ha satisfecho.

En el caso de un nanosatélite de control de orientación magnético activo, la capacidad de las magnetorcas para producir el campo magnético necesario para satisfacer los requerimientos está fuertemente ligada a las constantes elegidas para el controlador. Dentro del modelo virtual se ofrecerá la funcionalidad de parametrizar estas constantes con el fin de comparar las respuestas producidas por diferentes valores, lo cual ayudará al diseñador a seleccionar los valores adecuados.

3.3. Obtención de parámetros mediante STK

El modelo orbital y ambiental se encarga de proporcionar información al resto del modelo sobre variables relacionadas con la órbita del satélite, la cual se tratará de una órbita LEO, y al ambiente que lo rodea. Estas variables son independientes de la orientación y por lo tanto requieren modelos que no forman parte del ADCS. Por ejemplo, la información referente a la órbita puede ser manejada por otro subsistema en el satélite como el subsistema de control de órbita, mientras que los análisis relacionados con el ambiente espacial requieren de modelos más sofisticados, así que se suelen hacer desde Tierra. El desarrollo de modelos de esta naturaleza se

encuentra fuera del alcance de esta tesis, por lo que en su lugar se hará uso de STK para obtener estos datos.

Los parámetros que devuelve el modelo orbital y ambiental, y que son usados por el resto de los subsistemas son los siguientes:

- Posición del satélite dada en el marco de referencia ECI
- Coordenadas LLA
- Elementos keplerianos
- Vector de campo magnético en coordenadas inerciales
- Vector solar en coordenadas inerciales
- Densidad atmosférica

De estos parámetros se puede hacer una distinción en dos grupos, donde los primeros tres elementos hacen referencia a la posición del satélite o a su órbita, mientras que los tres restantes a los modelos ambientales y de referencia.

En el caso de estos últimos, como parte del subsistema de determinación de orientación, se poseerán algoritmos para determinar tanto el vector de campo magnético como el vector solar en coordenadas inerciales. Como resultado se tendrán dos subsistemas en el modelo que proveerán información sobre las mismas variables, pero la diferencia entre los resultados de ambos es que los del modelo ambiental representan el estado verdadero, mientras que los del subsistema de determinación de orientación están sujetos a las restricciones de la implementación y algoritmos utilizados, lo que resulta en una aproximación al estado verdadero.

Para determinar la densidad atmosférica se tiene una variedad de modelos a elegir dentro de STK, donde cada modelo tiene sus particularidades en cuanto a las suposiciones que realizan, los casos de uso y los métodos que se usan para determinar los valores. Estos modelos estarán disponibles para su selección dentro del modelo virtual.

3.3.1. Flujo de obtención de datos

Los datos obtenidos desde STK se pueden exportar en formato CSV por medio de reportes a través de la interfaz gráfica (3.3) o una integración programática. Para ello se debe generar un escenario de STK que refleje las condiciones iniciales de la simulación en cuanto a tiempo y elementos keplerianos de la misión.

En el modelo virtual este proceso se hará a partir de código dentro del mismo entorno de simulación, lo que ayudará a automatizar la obtención de datos, pero además servirá como una forma de reproducibilidad al siempre seguir los pasos impuestos en el código. El modelo orbital y ambiental importará estos datos y aplicará las transformaciones necesarias para generar señales compatibles con la simulación. La Figura 3.4 ilustra la secuencia de estos pasos.

Mediante este método también se asegurará que los datos de STK siempre estén actualizados a sus valores más recientes.

3. METODOLOGÍA

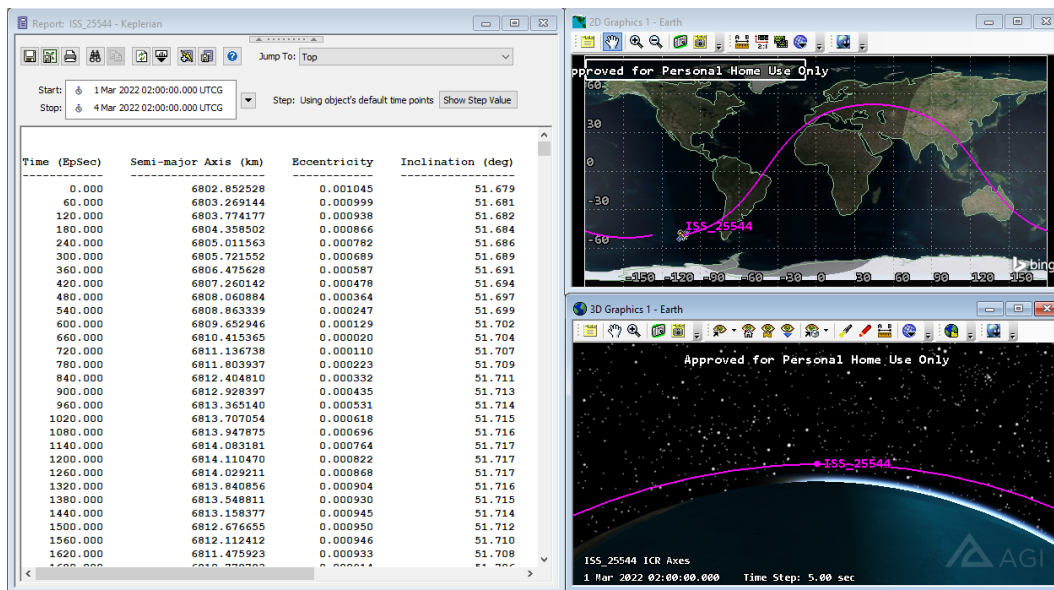


Figura 3.3: Ejemplo de reportes obtenidos mediante la interfaz gráfica de STK.

3.4. Subsistema de visualización

La visualización juega una parte importante en los análisis que involucran cambios de orientación. No solo es una forma de verificar que los resultados sean coherentes, sino que también resulta una herramienta poderosa en el desarrollo de los algoritmos, ya que sirven como una forma de depuración: la visualización puede proporcionar cierta intuición de porque sucede el comportamiento observado. Por ejemplo, se puede observar si las rotaciones se han realizado correctamente mediante la comparación de la posición de los ejes contra las referencias del marco de referencia elegido. En el contexto específico del análisis de una misión, la visualización se utiliza en otras áreas además de la orientación, por ejemplo, para seguir el recorrido del satélite a lo largo de su órbita e identificar las regiones de enlace.

Con referencia a este trabajo, se tienen tres áreas de interés a las cuales incorporar la visualización:

- **Orientación del satélite:** Para observar los efectos de las perturbaciones en el satélite y los efectos de los algoritmos de control de orientación.
- **Visualización de marcos de referencia:** Para evaluar que las rotaciones empleadas se hayan realizado correctamente.
- **Posición orbital:** Para identificar donde se encuentra el satélite con respecto al tiempo simulado.

El ambiente Simulink-MATLAB permite exportar los resultados de las simulaciones, con lo cual pueden ser cargados en herramientas de visualización externas como el propio STK. Sin embargo, Simulink también cuenta con herramientas nativas para generar animaciones, con lo cual se quitará la dependencia hacia programas

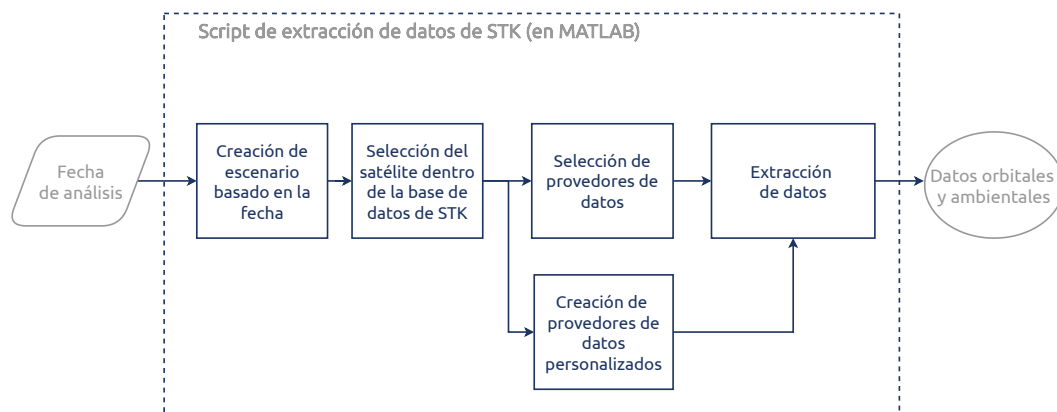


Figura 3.4: Diagrama del proceso de obtención de datos de STK a Simulink.

externos que además requerirían un procesamiento de datos adicional para asegurar la compatibilidad de formato. Una de estas herramientas es el lenguaje VRML (*Virtual Reality Modeling Language*), un formato con el cual se pueden generar modelos y animaciones en 3D, y que se manifiesta directamente en Simulink como un bloque configurable. Una de las principales ventajas de usar VRML dentro de Simulink surge de este hecho, ya que las animaciones pueden interactuar con los resultados de las simulaciones y reflejar los cambios del fenómeno que representan conforme transcurre la simulación. Dentro de las propiedades de los modelos 3D que se pueden modificar están el tamaño, posición, propiedades de iluminación y, lo que más concierne a este trabajo, la orientación. En la Figura 3.5 se muestra la interfaz para editar VRML dentro de Simulink.

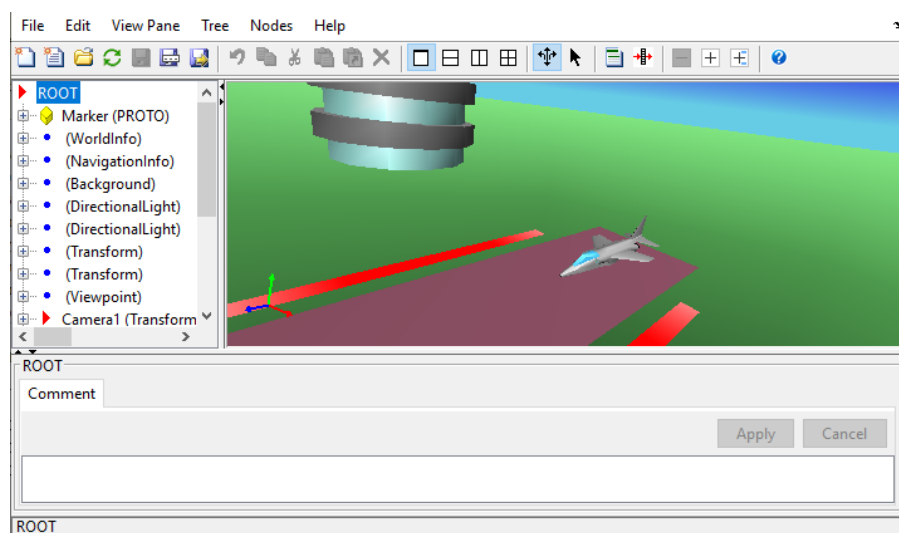


Figura 3.5: Ejemplo de la interfaz VRML en Simulink.

Los archivos resultantes de VRML están basados en texto, por lo que se pueden integrar fácilmente a un control de versiones además de que resultan en archivos

3. METODOLOGÍA

ligeros. Teniendo en cuenta que la presente herramienta se utilizará para fines de desarrollo del satélite y no en sus operaciones, no es necesario tener la visualización en forma paralela a la ejecución del modelo, sino que puede ejecutarse posteriormente a obtener los resultados para guardar recursos computacionales. Sin embargo es posible incorporar esa opción con facilidad mediante la incorporación de un bloque variante de Simulink. El flujo de información para el subsistema de visualización mediante VRML se muestra en la Figura 3.6

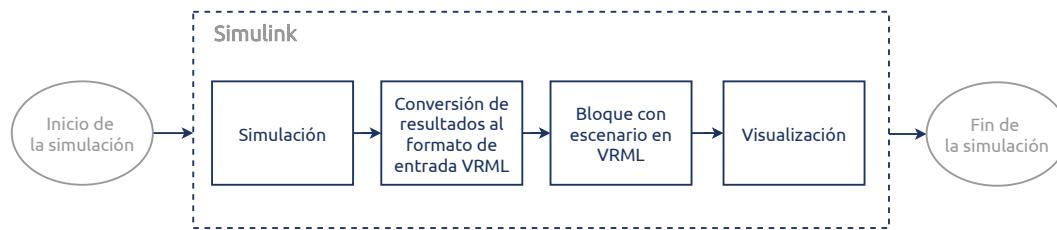


Figura 3.6: Flujo de información para el subsistema de visualización en Simulink.

4

Modelado

EN ESTA SECCIÓN se describirán los detalles de los modelos que compondrán los subsistemas del modelo virtual incluyendo: el ambiente espacial y las perturbaciones que afectan la orientación del satélite, los algoritmos de determinación de orientación, el algoritmo de estimación de estado, los algoritmos de control de orientación y los modelos de los componentes electrónicos. Esto últimos incluyen los sensores y actuadores que corresponden a la configuración del nanosatélite descrito en la Sección 3.2.

4.1. Perturbaciones del ambiente espacial

Para un satélite en el espacio, el ambiente espacial es la principal fuente de pares de perturbación. Otras fuentes como el mismo satélite y sus componentes internos (expulsión de masa, componentes flexibles, movimiento de combustible) generan pares órdenes de magnitud menores, al grado que suelen ser despreciables durante los análisis.

En el modelo virtual o modelo principal, el ambiente espacial será simulado a través del subsistema orbital y ambiental, en donde se proporcionarán las condiciones que darán origen a las perturbaciones en el satélite simulado. Las perturbaciones serán tratadas por medio del subsistema de perturbaciones, en donde se considerará el par por gradiente de gravedad, el par por presión de radiación solar y el par aerodinámico. Estos pares se detallan a continuación.

Cabe destacar que el par por campo magnético es otra perturbación de orden de magnitud similar a los anteriores, pero en este caso no se considera como tal puesto que un nanosatélite con control de orientación magnético usa este principio como un par de control. En el caso de un satélite sin control magnético, este tipo de par sí debe ser considerado como una perturbación.

4.1.1. Par por gradiente de gravedad

La Tierra posee un campo gravitatorio no uniforme y como consecuencia cualquier objeto no simétrico en una órbita alrededor de ella experimenta un par. Las fuerzas asociadas a este par son mayores en las regiones del satélite más cercanas a la Tierra mientras que son menores en las regiones más lejanas. Esta diferencia se conoce como el gradiente de gravedad. El par por gradiente de gravedad tiende a alinear el eje más largo del satélite hacia nadir y permanecerá en esa posición a menos de que actúen otras perturbaciones. Este es el principio en el que se basa el control de orientación por estabilización de gradiente de gravedad, pero en otros tipos de control el ADCS debe compensar los efectos de esta perturbación.

El par por gradiente de gravedad τ_g se puede describir mediante

$$\tau_g = \frac{3\mu}{|\mathbf{R}_s|^3} [\hat{\mathbf{R}}_s \times (I\hat{\mathbf{R}}_s)], \quad (4.1)$$

donde I es el tensor de inercia, μ es la constante gravitacional de la Tierra y \mathbf{R}_s es el vector de posición visto desde el marco de referencia fijo al cuerpo. La formulación anterior asume que el centro geométrico coincide con el centro de masa y que la Tierra es esférica [58].

4.1.2. Par por presión de radiación solar

Las fuentes principales de radiación electromagnética en el satélite son la proveniente directamente del sol, el albedo —la radiación solar reflejada por la Tierra y su atmósfera— y la radiación emitida directamente por la Tierra [49]. Generalmente solo se toma en consideración la primera debido a que tiene la mayor magnitud y que las demás disminuyen considerablemente con la altitud.

El satélite es golpeado constantemente por fotones cuando tiene línea de visión al sol. La radiación incidente en la superficie del satélite produce una fuerza que resulta en un par en torno al centro de masa. Este proceso se detiene cuando los fotones dejan de interactuar con el satélite, por lo que el par de presión de radiación solar desaparece cuando el satélite se encuentra en eclipse.

La radiación puede ser absorbida, reflejada especularmente o reflejada difusamente por la superficie en el satélite, lo cual depende de sus materiales. Las fuerzas diferenciales para cada uno de estos fenómenos en un elemento de área dA con normal $\hat{\mathbf{N}}$ se muestran a continuación:

$$df_{absorbida} = -\phi C_a \cos(\theta) \hat{\mathbf{S}} dA, \quad (4.2)$$

$$df_{especular} = -2\phi C_s \cos^2(\theta) \hat{\mathbf{N}} dA, \quad (4.3)$$

$$df_{difusa} = \phi C_d \left(-\frac{2}{3} \cos(\theta) \hat{\mathbf{N}} - \cos(\theta) \hat{\mathbf{S}} \right) dA, \quad (4.4)$$

donde C_a , C_s y C_d son los coeficientes de absorción, reflexión especular y reflexión difusa respectivamente, los cuales representan la fracción de la radiación incidente en la que ocurre cada uno de ellos [58]. Se relacionan mediante $C_a + C_s + C_d = 1$. Por otra parte \hat{S} es el vector hacia la dirección del sol medido desde el marco fijo al cuerpo, y θ es el ángulo entre \hat{N} y \hat{S} .

El flujo de momento promedio ϕ actuando sobre la superficie normal a la radiación solar se da por:

$$\phi = \frac{F_e}{c}, \quad (4.5)$$

donde F_e es la constante solar con un valor promedio de $1366 [W/m^2]$ y c es la velocidad de la luz.

La fuerza solar total f es la suma de cada uno de las fuerzas anteriores. Las ecuaciones son válidas para $(0^\circ \leq \theta \leq 90^\circ)$. Si $\cos(\theta)$ es negativo, la superficie no está iluminada y por lo tanto no experimentará fuerza solar. Esta situación ocurre en las caras opuestas a aquellas que están siendo iluminadas. Entonces la fuerza en una cara i está dada por

$$f_i = -\phi[(1 - C_s)\hat{S} + 2(C_s \cos(\theta) + \frac{1}{3}C_d)\hat{N}] \cos(\theta) A_i. \quad (4.6)$$

El par se calcula a partir de la ecuación anterior, considerando que el vector r_i en la que actúa la fuerza es la diferencia entre el centro de masa del satélite y el centro de presión solar del satélite. El par de presión de radiación solar total τ_{solar} en el satélite es la suma de los pares en las caras [79]:

$$\tau_{solar} = \sum_i (r_i \times f_i). \quad (4.7)$$

4.1.3. Par aerodinámico

El satélite estará sujeto a un arrastre aerodinámico siempre que se encuentre bajo influencia de la atmósfera. Esto es debido a la interacción de las moléculas de la atmósfera con la superficie del satélite. La densidad ρ de estas moléculas disminuye conforme la altitud aumenta, por lo que la fuerza de arrastre será menor a grandes altitudes. La expresión que define a la fuerza de arrastre f_i para una cara i es:

$$f_i = \frac{1}{2} C_D \rho v_{rel}^2 A_i, \quad (4.8)$$

donde C_D es el coeficiente de arrastre, A es el área en la que actúa la fuerza y v_{rel} es la velocidad relativa del satélite con respecto a la atmósfera [79]. El coeficiente de arrastre puede considerarse como $C_D = 2$ si no se cuenta con un valor disponible. Típicamente los valores se encuentran entre 1 y 2 [49].

Asumiendo que la atmósfera rota junto con la Tierra como un cuerpo rígido, la velocidad relativa del satélite se calcula mediante

$$v_{rel} = v - v_{atm}, \quad (4.9)$$

donde la velocidad del satélite v y la velocidad de la atmósfera v_{atm} están definidas en un sistema de referencia inercial [80]. En una forma más general se considerarían las variaciones en el viento, pero debido a que este tipo de información no es comúnmente disponible, se usa la expresión simplificada en la Ecuación (4.9) [81].

Al igual que el par de presión de radiación solar, el par aerodinámico para una cara i se calcula a partir de su respectiva fuerza f_i y un r_i , que en este caso es la diferencia entre el centro de masa del satélite y el centro de presión del satélite. La expresión resultante es:

$$\tau_{aero} = \sum_i (r_i \times f_i). \quad (4.10)$$

Como en el caso del par solar, el cálculo del par aerodinámico solo toma en consideración las caras en las cuales el ángulo entre la normal de la cara y el vector de velocidad este en $[0^\circ, 90^\circ]$.

Para satélites debajo de 400 km, el par aerodinámico es el par de perturbación ambiental dominante y su efecto principal es la disminución de la energía mecánica en el satélite. El arrastre es mayor en el perigeo, ya que la velocidad del satélite y la densidad de la atmósfera alcanzan sus valores máximos, lo que implica que la pérdida de energía mecánica es mayor en este punto. Esto tiene como consecuencia la reducción de la altitud del apogeo, disminuyendo la excentricidad y el semieje mayor de la órbita, con lo cual sin un sistema de maniobras de control de altitud, el satélite eventualmente reingresaría a la Tierra [80].

La sustentación o *lift* es otro tipo de fuerza de origen aerodinámico, la cual actúa perpendicularmente al vector de velocidad relativa. Sin embargo, en la mayoría de los casos es despreciable [82].

4.2. Modelos atmosféricos

La densidad de la atmósfera influye directamente en el cálculo del arrastre de acuerdo a la Ecuación (4.8), y en general disminuye con la altitud. Sin embargo, es uno de los parámetros más complejos de determinar debido a la multitud de variables que la afectan simultáneamente, requiriendo conocimientos de termodinámica, química molecular y electromagnetismo para obtener una buena aproximación.

La atmósfera está dividida en una serie de capas, donde cada una cuenta con su propia composición química y distribución de temperatura [80]. Por consiguiente, el modelo usado para describir una capa puede ser inadecuado para describir otra, razón por la cual algunos modelos atmosféricos toman en cuenta sus características individuales. En la Tabla 4.1 se describen algunas de las capas atmosféricas.

Tabla 4.1: Características de las capas que componen la atmósfera [83].

CAPA	RANGO	CARACTERÍSTICAS
Troposfera	Desde la superficie de la Tierra hasta 14.5 km.	Contiene la mayor cantidad de gases y vapor existente en la atmósfera. Es la capa en donde transcurren los vuelos comerciales. La mayoría de los fenómenos meteorológicos ocurren aquí.
Estratosfera	14.5 km a 50 km.	Ubicación de la capa de ozono. A diferencia de la troposfera la temperatura aumenta gradualmente con la altitud.
Mesosfera	50 km a 85 km.	En esta capa la temperatura disminuye una vez más con la altitud. La mayoría de los meteoritos se desintegran en esta capa.
Termosfera	85 km a 600 km.	La radiación solar y la baja densidad de la atmósfera resultan en un aumento de la temperatura. En esta capa es en donde la mayoría de satélites orbitan, incluyendo la ISS.
Exosfera	Empieza a partir de los 600 km.	Después de este rango se entraría al espacio interplanetario. Los átomos y moléculas son libres de escapar hacia el espacio, además de que la baja densidad resulta en una carencia de colisiones entre ellos.

En las alturas de LEO la baja densidad del aire tiene como consecuencia que modelos basados en el medio continuo de la atmósfera no sean representativos de la realidad, por lo que la interacción entre la atmósfera y la nave espacial debe tratarse al nivel molecular en los modelos atmosféricos [68].

Una clasificación general de los modelos atmosféricos los divide en estáticos o variantes en el tiempo. En los modelos estáticos se asume que los parámetros atmosféricos como la composición química permanecen constantes, lo que le confiere a este tipo de modelos una mayor simplicidad. Sin embargo, existen ciertos aspectos que sí se deben ser considerados como las variaciones latitudinales y longitudinales, debido a los efectos del ecuador durante la órbita para la latitud y las diferencias de terreno (por ejemplo montañas, mar abierto o llanura) que se pueden encontrar para una misma latitud pero a distinta longitud [81]. En otras palabras, la ubicación geográfica influye en el cálculo de la densidad.

En el caso de los modelos variantes en el tiempo se deben tomar consideraciones adicionales para tener una mejor aproximación de la realidad, aumentando su complejidad [62] [78] [81]. Algunas de ellas son:

- **Variación diurna:** Variaciones de la densidad de la atmósfera en el ciclo día-noche ocasionadas por el calentamiento debido a la radiación ultravioleta solar. Depende de la latitud. El valor máximo ocurre alrededor de 14:00 en el Tiempo Solar Local (LST) mientras el mínimo ocurre alrededor de las 04:00.

- **Variaciones semianuales:** Variaciones que duran seis meses y están relacionadas con la variación de la distancia de la Tierra al sol y la latitud del sol a lo largo del año. Este tipo de variaciones son usualmente pequeñas.
- **Ciclo de rotación solar de 27 días:** Efecto del periodo rotacional del sol, y correlacionado con el flujo solar. Algunos cambios irregulares en el flujo solar están relacionados con el crecimiento y decaimiento de regiones solares activas. Estos patrones son difíciles de predecir por lo que su incertidumbre tiene un efecto en la predicción del flujo solar.
- **Variaciones por la actividad geomagnética:** Las tormentas geomagnéticas pueden incrementar la temperatura, así como la densidad y la composición química de la termosfera en una escala de uno a dos días [82]. Su intensidad es medida mediante el índice geomagnético medido en nT.
- **Atmósfera en rotación:** La atmósfera rota con respecto a la Tierra, lo cual genera variaciones de la densidad con el tiempo.
- **Viento:** Causa variaciones en la temperatura de la atmósfera, lo cual ocasiona cambios en la densidad. El viento y demás fenómenos meteorológicos son complejos y difíciles de predecir.
- **Mareas:** Variaciones muy pequeñas en la densidad atmosférica.
- **Variaciones estación-latitud del helio:** Resultantes de la migración del helio hacia los polos en invierno.

Tomando en consideración los puntos anteriores, se puede decir que la densidad de la atmósfera cambia principalmente debido a la interacción de tres parámetros: su composición molecular, el flujo solar incidente y las interacciones geomagnéticas [81]. Un mayor flujo solar y actividad geomagnética incrementan la densidad al aumentar las colisiones entre partículas debido al calentamiento que provocan en ellas.

Si bien la complejidad de calcular la densidad debido a estas dependencias tiene como consecuencia que el modelo atmosférico no se incorpore directamente en el controlador, el efecto del par aerodinámico puede ser superado con un buen sistema de rechazo de perturbaciones siempre y cuando se hagan verificaciones en Tierra usando un modelo que sí incorpore la variación en la densidad de la atmósfera.

4.2.1. Ejemplos de modelos de densidad atmosférica

El subsistema ambiental y orbital dentro del modelo principal tendrá la capacidad de seleccionar de entre varios modelos de densidad, tanto estáticos como variantes en el tiempo. El proveedor de estos datos será STK a partir del método descrito previamente en la Figura 3.4. Los modelos atmosféricos disponibles a partir de esta manera se describen a continuación.

Modelo exponencial. El modelo más simple para calcular la densidad atmosférica es el modelo exponencial. Se caracteriza por ser un modelo completamente estático, por lo que los valores de densidad calculados son independientes del tiempo. En este modelo la densidad disminuye exponencialmente con la altitud. El modelo está

descrito por:

$$\rho = \rho_0 \exp\left(-\frac{(h - h_0)}{H}\right), \quad (4.11)$$

donde ρ_0 y h_0 son la densidad y altura de referencia respectivamente, h es la altitud y H es el factor de cambio de la densidad con la altura. H está medida en km. Existen tablas de referencia en donde se pueden encontrar valores de ρ , h_0 y H .

La simplicidad del modelo y su bajo consumo de recursos computacionales lo hace una herramienta útil en simulaciones preliminares [67]. El rango válido va de 0 a 1000 km de altitud [81].

Harris–Priester. Toma en cuenta el nivel de flujo solar F10.7 cm. El rango válido reportado en la implementación de STK es 0 km a 1000 km [84], mientras que en la literatura es de 120 a 2000 km [81].

El modelo Harris-Priester fue uno de los primeros intentos de modelar la densidad atmosférica en términos de la temperatura. La densidad numérica es calculada mediante la integración de las ecuaciones de difusión y conducción de calor para cada tipo de moléculas que constituyen la atmósfera. La densidad total de la atmósfera es calculada mediante la ley de gas ideal. La temperatura se obtiene mediante la integración de la ecuación de conducción de calor.

Si bien representó un avance para modelar los fenómenos que se llevan a cabo en la atmósfera de una manera más completa, el modelo de Harris-Priester no toma en cuenta otros fenómenos como las variaciones latitudinales, geomagnéticas o estacionarias, las cuales son importantes en análisis de largo plazo [67]. Es un modelo esencialmente estático, pero contiene tablas en donde se consideran las densidades que fluctúan con el ciclo solar. Hace uso de interpolación para determinar la densidad en un momento dado. No modela los efectos de la rotación solar de 27 días [81].

Una comparación con el modelo de Jacchia 1971 muestra una desviación promedio de los valores de la densidad de aproximadamente 40 % en condiciones de flujo solar promedio, la cual incrementa a 60 % para actividad solar máxima [82].

US Standard Atmosphere 1976. También llamado 1976 Standard, fue desarrollado por NOAA, NASA y la fuerza aérea de Estados Unidos. Este modelo busca valores en tablas a partir de la altitud del satélite, y su rango válido es de 86 a 1000 km [84]. Divide la atmósfera en un total de seis capas, cada una definida por su propio gradiente de temperatura. Asume un valor constante de la gravedad, además de un aire seco, pero la diferencia en densidad con respecto a un aire con humedad es menor a 1 % por lo que es despreciable [80]. Es un modelo ideal que asume una latitud de 45° y actividad solar moderada. Su intención es la de ser un modelo representativo para todo el año y ser usado en situaciones como diagramas meteorológicos, así como diseño de aeronaves y naves espaciales. Se asume que el aire obedece la ecuación de gas ideal y la ecuación hidrostática [81].

CIRA 1972. Modelo empírico de temperatura y densidades atmosféricas según lo recomendado por la Comisión de Investigaciones Espaciales (COSPAR). Toma el

mismo principio que el modelo Jacchia 1971, pero utiliza la integración numérica en lugar de interpolar polinomios para algunas cantidades. Sus datos vienen principalmente de mediciones del arrastre en satélites y mediciones en Tierra. El límite inferior de altitud es de 90 km [84].

DTM 2012. El DTM 2012 (*Drag Temperature Model*) es un modelo semiempírico que calcula la temperatura, la densidad y la composición de la termosfera.

El modelo DTM usa armónicos esféricos para incorporar datos del arrastre de un satélite en dos ciclos solares completos y de temperatura de la exosfera. Tiene la ventaja utilizar menos código que el modelo de Jacchia-Roberts, sin embargo, este último aún puede tener un mejor desempeño en algunas ocasiones a pesar de que el modelo DTM sea más reciente. El modelo DTM es más sensible a cambios de flujo solar [81]. El rango válido es de 120 km a 1500 km [84].

4.2.1.1 MODELOS DE JACCHIA

Los modelos atmosféricos de Jacchia son una serie de modelos realizados entre los años sesenta y los años setenta por el astrónomo Luigi G. Jacchia. A diferencia del modelo de Harris-Priester, los modelos de Jacchia usan un perfil de temperatura derivado empíricamente para integrar la ecuación de difusión para cada componente de la atmósfera. Para considerar los efectos de la actividad solar, los modelos de Jacchia agregan correcciones de temperatura y densidad posteriormente, en lugar de tomarlos en cuenta en la resolución de la ecuación de conducción.

Los modelos de Jacchia emplearon los datos obtenidos directamente de naves espaciales para realizar un mejor ajuste en ciertos fenómenos del modelo. El uso de datos reales lo vuelve un modelo más fidedigno. Una de sus desventajas es que no existe una serie de tablas estandarizadas, por lo que se requieren los métodos numéricos para determinar la temperatura y densidad para un momento deseado [67]. Los modelos de Jacchia utilizan la temperatura de la exosfera como uno de los parámetros básicos para calcular la densidad atmosférica.

Jacchia 1971. Calcula la densidad atmosférica de acuerdo la composición de la atmósfera, la cual depende de la altitud del satélite, así como las variaciones estacionales. Rango válido de 100 km a 2500 km [84]. A diferencia de sus antecesores, calcula la densidad en función del tiempo y no solamente la temperatura y la altura. El proceso para calcular la densidad en el modelo Jacchia 1971 es [82]:

- Calcular la temperatura de la exosfera a partir de los datos de actividad solar e índices geomagnéticos.
- Un perfil de temperatura se asume al conocer la temperatura de la exosfera, con lo cual se puede usar la ecuación barométrica o de difusión para calcular la densidad, según sea el caso.
- Se aplican correcciones para tomar en cuenta las variaciones de la densidad.

Jacchia 1970. El antecesor del modelo Jacchia 1971. El rango válido es de 90 km a 2500 km [84].

Jacchia 1960. Un modelo anterior que usa el ciclo solar para predecir un valor para el flujo solar F10.7 cm y explica los efectos del abultamiento diurno. El límite inferior de altitud es de 0 km [84].

Jacchia-Roberts. Usa el mismo principio que Jacchia 1971, pero utiliza métodos analíticos de integración para las ecuaciones barométrica y de difusión. Utiliza el mismo perfil de temperatura que los modelos previos de Jacchia en el rango de 90 a 125 km, después del cual se hacen uso de las soluciones analíticas. El límite inferior de altitud es de 90 km [84]. Contiene expresiones analíticas para determinar la temperatura de la exosfera como función de la posición, tiempo, actividad solar y actividad geomagnética. Una vez conseguida la temperatura, la densidad se obtiene empíricamente a partir de perfiles de temperatura o de la ecuación de difusión.

Este modelo cuenta con diferencias relativas de 1 % y 3 % con respecto a la densidad promedio y máxima del modelo Jacchia 1971. Es uno de los modelos atmosféricos más utilizados. Sus resultados prometedores causaron que la NASA implementara el modelo en el *Goddard Trajectory Determination System* (GTDS).

Jacchia-Bowman 2008. Utiliza datos del arrastre de una nave espacial y nuevos índices solares para realizar correcciones al modelo Jacchia 1970. Dentro de los cambios realizados se encuentran:

- La incorporación de nuevos índices solares además del índice estándar F10.7 cm para calcular la temperatura de la exosfera.
- Correcciones semianuales, donde se considera cómo varía la densidad a lo largo del año. La densidad máxima ocurre en abril y octubre mientras la mínima ocurre en enero y julio.
- Correcciones estacionales y latitudinales. Es común que para determinada altitud se asuma una temperatura constante e idéntica en todas las regiones del mundo cuando en realidad, la temperatura para cierta altura varía tanto con las estaciones del año como a la latitud en la que se encuentre.
- Correcciones para altitudes grandes. Mediante un factor de corrección obtenido tras el análisis de 25 naves espaciales con altitudes orbitales de entre 1500 y 4000 km.
- Medios para considerar la actividad geomagnética.

El modelo de Jacchia-Bowman usa la función de temperatura contenida en el modelo original y no la analítica de Roberts, por lo que se requiere integración numérica para encontrar el perfil de densidad. El modelo es lento, pero las correcciones hechas por Bowman reducen sustancialmente el error en el cálculo de la densidad a comparación de otros modelos. Los resultados de alta calidad del modelo hicieron que Jacchia-Bowman fuera adoptado como el modelo por defecto en el *Joint Space Operations Center* (JSpOC) cuando se trata de rastrear objetos [67].

Para altitudes de entre 90 y 105 km basta con integrar la ecuación barométrica para determinar la densidad atmosférica. Asume que la variación de la presión de la atmósfera con la altitud únicamente se debe al efecto de la gravedad. Por arriba del rango de 105 km, se asume que la atmósfera está en equilibrio difuso, en donde la densidad se calcula al determinar la densidad numérica de cada molécula que compone la atmósfera: nitrógeno, argón, helio, hidrógeno y oxígeno molecular y atómico (O y O₂) [67]. El hidrógeno no influye en el cálculo de la densidad para altitudes menor a 500 km, por lo que la atmósfera se segmenta en regiones de acuerdo a ello.

4.2.1.2 MODELOS MSIS

Las siglas de los modelos MSIS significan *Mass Spectrometer and Incoherent Scatter Radar*. Están basados en datos recopilados directamente a partir de satélites.

MSIS 1986 y MSISE 1990. Estos modelos encuentran la densidad total al considerar la contribución de N₂, O, O₂, He, Ar y H. Para MSIS 1986 el rango válido es de 85 a 1000 km, mientras que para MSISE 1990 es de 0 a 1000 km [84].

NRLMSISE 2000. Modelo empírico de densidad desarrollado por el Laboratorio de Investigación Naval de EE. UU. (NRL). Encuentra la densidad total al considerar la contribución de N, N₂, O, O₂, He, Ar y H. Incluye oxígeno anómalo. El rango válido es de 0 a 1000 km [84].

4.3. Algoritmos de determinación de orientación

Para controlar un sistema se requiere del conocimiento de su estado actual. Los satélites no son la excepción a lo anterior, y por consecuencia si se desea controlar la orientación, primero es necesario tener un medio por el cual se pueda determinar. Una vez conocida la orientación del satélite, el sistema de control de orientación podrá calcular las acciones necesarias que ejecutarán los actuadores para llegar a la orientación deseada [78].

Sensores que son montados en el satélite con el propósito de conocer la orientación en el espacio, como los sensores solares y magnetómetros, no regresan por sí solos una representación de orientación como ángulos de Euler o cuaterniones. En su lugar, los datos producidos por estos sensores son enviados a una serie de algoritmos denominados de *determinación de orientación* que extraen la información relevante para calcular la orientación en una de las representaciones mencionadas con anterioridad.

4.3.1. Generalidades

Los algoritmos de determinación de orientación se enfrentan a problemas de naturaleza matemática y de implementación para definir una orientación con la menor cantidad de información posible, así como las peculiaridades de cada

sistema de representación como la complejidad computacional o la presencia de singularidades.

Lo anterior puede o no representar una serie de restricciones a las misiones que se enfrenten al problema de determinación de orientación, por lo que como resultado, se han desarrollado una variedad de algoritmos que sobresalen en los distintos escenarios en los que se puede encontrar una misión según sus recursos disponibles.

En el problema de determinación de orientación existen dos clases de métodos: los estáticos y las variantes en el tiempo. La diferencia entre ambos es que en los estáticos la diferencia entre el tiempo de adquisición de datos entre los sensores involucrados es muy pequeña o despreciable, por lo que se asume que el satélite permanece en la misma orientación durante el tiempo de muestreo. Los variantes en el tiempo hacen uso de filtros y de conocimiento previo sobre la orientación de la nave espacial para hacer su determinación [67]. Por consiguiente los modelos estáticos pueden determinar la orientación en un momento instantáneo, mientras que los modelos variantes en el tiempo requieren de conocer el estado previo de la orientación.

Otra diferencia es que los modelos variantes en el tiempo requieren de un mayor uso de memoria y recursos computacionales por lo que se debe evaluar su uso mediante una solución de compromiso o *trade-off* entre exactitud y uso de recursos.

4.3.1.1 EL PROBLEMA DE LA DETERMINACIÓN DE ORIENTACIÓN

La definición de la orientación en un momento dado requiere el conocimiento de tres variables independientes. Sin embargo la medición de un vector como el solar o el del campo magnético, consiste en realidad de dos variables correspondientes a la dirección del vector (dos ángulos), ya que la tercera variable involucrada, su magnitud; tiene la restricción del vector unitario además de no ser un parámetro involucrado en la definición de una rotación.

Lo anterior quiere decir que si únicamente se tienen las mediciones de un solo vector, entonces el problema de determinación de orientación no estará suficientemente definido al tener dos de las tres variables necesarias, pero por el contrario si se tienen las mediciones de dos o más vectores, el problema estará sobredeterminado [85].

La idea general de la determinación de orientación es obtener la posición angular del satélite, de tal forma que la matriz de rotación determinada \mathbf{R} pueda transformar los componentes inerciales a los del marco fijo al cuerpo o viceversa.

Para el nanosatélite de control de orientación magnético activo a simular se harán uso de sensores solares y magnetómetros, así que, si se trabaja con el vector solar s en los marcos de referencia inercial y fijo al cuerpo, s_i y s_b respectivamente, y el vector del campo magnético B con B_i y B_b ; la matriz de rotación \mathbf{R}_{i2b} (inercial a

marco fijo al cuerpo) debería ser tal que:

$$\mathbf{s}_b = \mathbf{R}_{i2b}\mathbf{s}_i, \quad (4.12)$$

$$\mathbf{B}_b = \mathbf{R}_{i2b}\mathbf{B}_i. \quad (4.13)$$

Sin embargo, la sobredeterminación del problema tiene como consecuencia que no se pueda determinar una \mathbf{R}_{i2b} única capaz de satisfacer ambas transformaciones de forma exacta [85].

La determinación de la matriz \mathbf{R}_{i2b} en las Ecuaciones (4.12) y (4.13) requiere el conocimiento de los vectores en cada uno de los marcos de referencia con los que se esté trabajando. Los vectores en el marco inercial se obtienen a partir de modelos matemáticos o utilizando un propagador como STK, mientras que los valores del marco fijo al cuerpo se obtienen directamente desde los sensores montados en el satélite y de hacer las rotaciones necesarias en el caso de que no se encuentren alineados con los ejes principales.

4.3.2. Modelos inerciales de referencia

Los modelos inerciales son los modelos usados para calcular los vectores de referencia vistos desde el marco de referencia ECI. En la implementación son embebidos a bordo del satélite con el fin de determinar la orientación para un tiempo arbitrario. Otra alternativa es incluir únicamente los resultados de estos modelos a manera de tablas.

Dentro del subsistema de determinación de orientación de la simulación se incluyen los modelos de referencia inercial para obtener el vector solar y el vector de campo magnético. Como parte del proceso de elaboración del modelo virtual se desarrolló el modelo para el vector solar, el cual se describe en la siguiente sección. Para el modelo del campo magnético se utilizó el modelo IGRF (*International Geomagnetic Reference Field*), el cual hace uso de armónicos esféricos para determinar el potencial escalar magnético y así determinar el campo magnético. Para mayor información del modelo se hace referencia a [86].

4.3.2.1 MODELO INERCIAL DEL SOL

A diferencia de los modelos usados para calcular el campo electromagnético, los modelos para el sol no requieren de tablas de datos o coeficientes obtenidos experimentalmente y únicamente se requiere el uso de geometría gracias a la predictibilidad de la órbita del sol. Por esa razón, resulta viable implementar este modelo dentro del satélite en lugar de calcular los datos en Tierra y almacenar los resultados en la memoria interna.

El cálculo de la posición inercial del sol únicamente requiere como entrada la Fecha Juliana (JD), la cual se puede calcular a partir de la fecha gregoriana y los datos de la hora en un instante dado (en formato de 24 horas) mediante la siguiente

ecuación [85]

$$JD = 367(año) - INT \left\{ \frac{7[año + INT \frac{mes+9}{12}]}{4} \right\} + INT \frac{275(mes)}{9} + día + 1721013.5 + \frac{hora}{24} + \frac{minuto}{1440} + \frac{segundos}{86400}, \quad (4.14)$$

donde INT hace referencia al tipo de dato entero.

Para calcular el vector solar se deben calcular como pasos intermedios la longitud promedio del sol ($\lambda_{M_{Sun}}$), la anomalía promedio del sol (M_{Sun}) y la longitud eclíptica del sol ($\lambda_{ecliptic}$).

Para calcular $\lambda_{M_{Sun}}$ se requiere conocer la fecha en formato J2000 T_{UT1} . Esto se puede calcular mediante

$$T_{UT1} = \frac{JD - 2451545}{36525}, \quad (4.15)$$

además de que el valor de la longitud promedio del sol debe ser ajustado apropiadamente mediante

$$\lambda_{M_{Sun}} = 280.4603484deg + 36000.77005361(T_{UT1}), \quad (4.16)$$

donde la notación deg significa que las unidades se dan en grados. Se debe hacer la suposición de que el Tiempo Dinámico Baricéntrico (T_{TDB}) satisface $T_{TDB} \approx T_{UT1}$ para poder calcular la anomalía promedio del sol mediante la Ecuación (4.17) y así calcular la longitud eclíptica mediante la Ecuación (4.18) [78]:

$$M_{Sun} = 357.5277233deg + 35999.05034(T_{TDB}), \quad (4.17)$$

$$\lambda_{ecliptic} = \lambda_{M_{Sun}} + 1.914666471deg[\sin(M_{Sun})] + 0.019994643[\sin(2M_{Sun})]. \quad (4.18)$$

La oblicuidad de la eclíptica se calcula mediante:

$$\epsilon = 23.439291deg - 0.0130042(T_{TDB}). \quad (4.19)$$

Finalmente, para obtener el vector solar inercial únicamente se hace uso de la oblicuidad de la eclíptica y la longitud eclíptica ajustada (Ecuaciones (4.19) y (4.18) respectivamente):

$$\mathbf{s}_i = \begin{bmatrix} \cos \lambda_{ecliptic} \\ \cos \epsilon \sin \lambda_{ecliptic} \\ \sin \epsilon \sin \lambda_{ecliptic} \end{bmatrix}. \quad (4.20)$$

A partir de la anomalía promedio del sol también se puede calcular la distancia hacia el sol en unidades astronómicas (AU). La distancia d entre la Tierra y el sol está dada por

$$d = 1.000140612 - 0.016708617[\cos(M_{Sun})] - 0.000139589[\cos(2M_{Sun})], \quad (4.21)$$

donde el resultado puede utilizarse para calcular la distancia de la nave espacial al sol [67].

4.3.3. Algoritmos de determinación estáticos

Los algoritmos de determinación de orientación que se propusieron para su implementación en el modelo fueron TRIAD, Davenport Q y QUEST. Todos son estáticos y requieren de al menos dos vectores como entrada. La diferencia entre ellos radica en características como la representación de orientación que utilizan, sus suposiciones y sus limitaciones.

4.3.3.1 TRIAD

El algoritmo TRIAD fue el primer algoritmo de determinación de orientación publicado donde se usaron observaciones en el marco inercial y el marco fijo al cuerpo [67]. Fue ideado en 1964 y es considerado como el método determinista más sencillo. Sus siglas significan *TRI*axial *A*ttitude *D*etermination, aunque también hace referencia a la palabra «tríada» en la cual basa su concepto [67].

TRIAD consiste en construir dos tríadas de vectores ortogonales a partir de la información de los vectores de entrada: una tríada a partir de los vectores inerciales y la otra a partir de los vectores del marco fijo al cuerpo. Requiere como entrada de conocer las mediciones de dos vectores tanto en el marco inercial como en el marco fijo al cuerpo, para un total de cuatro entradas. TRIAD tiene la restricción de que los vectores no deben ser paralelos.

Ambas tríadas se construyen usando el mismo procedimiento [85]. Sea t_b la tríada construida a partir de los vectores en el marco fijo al cuerpo t_{nb} . La obtención de estos vectores se muestra en la Ecuación (4.22).

$$\begin{aligned} t_{1b} &= s_b, \\ t_{2b} &= \frac{s_b \times B_b}{|s_b \times B_b|}, \\ t_{3b} &= t_{1b} \times t_{2b}. \end{aligned} \quad (4.22)$$

Para la tríada inercial t_i , los vectores inerciales t_{ni} que la componen se obtienen a través de la Ecuación (4.23):

$$\begin{aligned} t_{1i} &= s_i, \\ t_{2i} &= \frac{s_i \times B_i}{|s_i \times B_i|}, \\ t_{3i} &= t_{1i} \times t_{2i}. \end{aligned} \quad (4.23)$$

A continuación se da una interpretación para el procedimiento.

- El primer vector que compone la tríada corresponde a uno de los vectores de entrada, donde el vector elegido es aquel en donde existe menos incertidumbre. En el ejemplo se eligió al vector solar.
- El segundo vector de la tríada se obtiene al calcular el producto cruz entre los dos vectores de entrada, en este caso s y B , para asegurar la ortogonalidad con respecto a cada uno de ellos.
- La obtención del tercer vector se da a partir del producto cruz entre el primer y segundo vector de la tríada. Por lo tanto, los tres vectores que componen la tríada son ortogonales, independientemente del vector medido que se eligió para iniciar la secuencia.

En el caso de cada tríada es posible construir una matriz de 3x3 al usar los respectivos vectores t_n como columnas. El resultado son dos matrices de 3x3 construidas de la siguiente forma:

$$t_b = [t_{1b} \quad t_{2b} \quad t_{3b}], \quad (4.24)$$

$$t_i = [t_{1i} \quad t_{2i} \quad t_{3i}]. \quad (4.25)$$

La matriz R_{i2b} se puede encontrar a partir de las dos matrices anteriores donde

$$R_{i2b} = [t_{1b} \quad t_{2b} \quad t_{3b}] [t_{1i} \quad t_{2i} \quad t_{3i}]^T, \quad (4.26)$$

de tal forma que para un vector v con componentes conocidos en el marco inercial y el marco fijo al cuerpo, se cumpla la Ecuación (4.27) en concordancia con las Ecuaciones (4.12) y (4.13).

$$v_b = R_{i2b} v_i. \quad (4.27)$$

El algoritmo TRIAD tiene como salida la matriz de rotación definida por (4.26) y dependiendo de si se desea encontrar la transformación de inercial a marco fijo al cuerpo o viceversa solo basta aplicar la traspuesta según sea el caso.

Este algoritmo se basa en la suposición de que uno de los sensores es más preciso que el otro, y a partir de este primero se empieza a construir las tríadas. En las Ecuaciones (4.22) y (4.23) se asume que la medición del vector solar es más precisa que la del campo magnético, por lo que la inicialización empieza con esa información.

Lo anterior garantiza que al usar la Ecuación (4.27) para transformar el vector «más preciso» de un marco de referencia a otro, se tenga como resultado exactamente su contraparte en el otro marco de referencia. Sin embargo, esto no ocurre al realizar la transformación del vector «menos preciso» (en este caso el del campo magnético), ya que la matriz de rotación produce solamente una estimación debido al problema de sobredeterminación.

La presencia del error en la segunda transformación es una de las consecuencias de la naturaleza determinista de este modelo, así como la necesidad de descartar una pieza de información.

Uno de los inconvenientes a los que se enfrenta TRIAD es que no cuenta con un mecanismo para reducir este error, por lo que no hay garantía de que la orientación regresada por el algoritmo sea aquella que realmente describa la orientación real bajo un margen de error aceptable. Por otra parte, su simplicidad lo hace una buena herramienta en misiones con recursos limitados.

4.3.3.2 EL PROBLEMA DE WAHBA

Otra clase de algoritmos de determinación de orientación estáticos, además de los deterministas, son los métodos estadísticos. A diferencia de TRIAD, en estos métodos es posible tener más de dos conjuntos de vectores (cada uno con mediciones en el marco inercial y en el marco de referencia fijo al cuerpo) hasta contar con N vectores. La principal ventaja de los métodos estadísticos es que hacen uso de toda la información disponible para determinar la orientación. Los dos algoritmos de determinación de orientación restantes en el modelo virtual, Davenport Q y QUEST, son métodos estadísticos.

La sobredeterminación implica que no existirá una matriz de rotación capaz de transformar todas las mediciones de un marco de referencia a otro sin presencia de error. Este es el llamado problema de Wahba.

El objetivo de los métodos estadísticos es minimizar el error de la matriz de rotación para todos los vectores y así evitar que existan situaciones como en TRIAD, donde una rotación puede tener un error significativo mientras que la otra un error de cero. Los métodos estadísticos en otras palabras, buscan minimizar la función de pérdida [85]

$$J(\mathbf{R}_{\mathbf{i2b}}) = \frac{1}{2} \sum_{k=1}^N w_k |v_{kb} - \mathbf{R}_{\mathbf{i2b}} v_{ki}|^2. \quad (4.28)$$

El problema de Wahba consiste entonces en encontrar una matriz ortogonal \mathbf{R} con determinante +1, una matriz de rotación, que minimice la Ecuación (4.28).

En la función de pérdida se lleva a cabo una suma de los errores cuadrados de cada vector con respecto a la rotación resultante de aplicar $\mathbf{R}_{\mathbf{i2b}}$. Ambos vectores estarían en el mismo marco de referencia. Cada error cuadrado es multiplicado por w_k , un peso entre cero y uno asociado a la precisión o importancia del vector v_k . Una regla de dedo es asignar los pesos w_k de acuerdo a la varianza de las mediciones, $w_k = 1/\sigma_k^2$. De esta forma una medición con baja incertidumbre tendrá un mayor peso que una con alta incertidumbre [68].

En el caso ideal $J = 0$, pero debido a que siempre habrá errores en las mediciones y en la determinación de la matriz de orientación, casi siempre se tendrá $J > 0$ [85].

4.3.3.3 DAVENPORT Q

El método Davenport Q o simplemente el método Q, empieza por expandir la función de pérdida del problema de Wahba (Ecuación (4.28)).

$$J = \frac{1}{2} \sum w_k (\mathbf{v}_{kb} - \mathbf{R}_{i2b} \mathbf{v}_{ki})^T (\mathbf{v}_{kb} - \mathbf{R}_{i2b} \mathbf{v}_{ki}), \quad (4.29)$$

$$= \frac{1}{2} \sum w_k (\mathbf{v}_{kb}^T \mathbf{v}_{kb} + \mathbf{v}_{ki}^T \mathbf{v}_{ki} - 2 \mathbf{v}_{kb}^T \mathbf{R}_{i2b} \mathbf{v}_{ki}). \quad (4.30)$$

Los vectores \mathbf{v}_{kb} y \mathbf{v}_{ki} están normalizados porque únicamente es de interés su dirección y no su magnitud, y por lo tanto su norma es uno. Haciendo uso de esta simplificación se obtiene

$$J = \sum w_k (1 - \mathbf{v}_{kb}^T \mathbf{R}_{i2b} \mathbf{v}_{ki}). \quad (4.31)$$

Minimizar la Ecuación (4.31) es equivalente a maximizar su contraparte, la función de ganancia, la cual está definida como:

$$g(\mathbf{R}_{i2b}) = \sum w_k (\mathbf{v}_{kb}^T \mathbf{R}_{i2b} \mathbf{v}_{ki}). \quad (4.32)$$

Se pueden hacer uso de identidades para descomponer la matriz \mathbf{R} en términos de cuaterniones [67] [68]. Con esto la función de ganancia se simplifica a

$$g(\mathbf{q}) = \mathbf{q}^T K \mathbf{q}, \quad (4.33)$$

considerando que se hace uso de la convención en donde q_4 es el escalar, y K se define como

$$K = \begin{bmatrix} S - \sigma \mathbf{I}_{3 \times 3} & Z \\ Z^T & \sigma \end{bmatrix}, \quad (4.34)$$

donde

$$B = \sum_{k=1}^N w_k (\mathbf{v}_{kb} \mathbf{v}_{ki}^T), \quad (4.35)$$

$$S = B + B^T, \quad (4.36)$$

$$Z = \begin{bmatrix} B_{23} - B_{32} \\ B_{31} - B_{13} \\ B_{12} - B_{21} \end{bmatrix} = \sum_{k=1}^N w_k (\mathbf{v}_{kb} \times \mathbf{v}_{ki}), \quad (4.37)$$

$$\sigma = \text{tr}[B]. \quad (4.38)$$

K es una matriz simétrica con traza cero, por lo que se puede realizar la descomposición de eigenvalores λ y eigenvectores x de la siguiente manera:

$$K = \sum_{k=1}^N \lambda_k x_k x_k^T. \quad (4.39)$$

Además, se tiene que K es diagonalizable por lo que su traza es la suma de sus eigenvalores, que en este caso suman a cero,

$$\text{tr}[K] = \sum_{k=1}^4 \lambda_k = 0. \quad (4.40)$$

Al sustituir la Ecuación (4.39) en la Ecuación (4.33), se obtiene la siguiente forma:

$$g(\mathbf{q}) = \mathbf{q}^T K \mathbf{q}, \quad (4.41)$$

$$= \mathbf{q}^T \sum_{k=1}^N \lambda_k x_k x_k^T \mathbf{q}, \quad (4.42)$$

$$= \sum_{k=1}^N \lambda_k (\mathbf{q}^T x_k) (x_k^T \mathbf{q}), \quad (4.43)$$

$$= \sum_{k=1}^N \lambda_k (\mathbf{q}^T x_k) (\mathbf{q}^T x_k), \quad (4.44)$$

$$= \sum_{k=1}^N \lambda_k (\mathbf{q}^T x_k)^2. \quad (4.45)$$

Manipulaciones adicionales pueden realizarse haciendo uso de la ortonormalidad de los eigenvectores para llegar a la forma de

$$g(\mathbf{q}) = \lambda_{max} - \sum_{i=2}^4 (\lambda_{max} - \lambda_i) (\mathbf{q}^T x_i)^2. \quad (4.46)$$

En otras palabras, la función de ganancia es maximizada cuando el cuaternión \mathbf{q} es ortogonal a los eigenvectores x_2, x_3 y x_4 . Es decir $\mathbf{q} = x_1$. Entonces el cuaternión óptimo, aquel que corresponde a la orientación determinada, es el eigenvector de K que corresponde al eigenvalor más grande [67].

Un método alternativo para encontrar el \mathbf{q} que maximiza la Ecuación (4.33) es tener en cuenta la restricción del cuaternión unitario, $\mathbf{q}^T \mathbf{q} = 1$. Para encontrar una solución que cumpla con esta restricción se usa un multiplicador de Lagrange:

$$\mathcal{L}(\mathbf{q}, \lambda_{\mathcal{L}}) = \mathbf{q}^T K \mathbf{q} - \lambda_{\mathcal{L}} (\mathbf{q}^T \mathbf{q} - 1). \quad (4.47)$$

Derivando la Ecuación (4.47) da como resultado la Ecuación (4.48) [78], en donde se cumple la restricción de que $\mathbf{q}^T \mathbf{q} = 1$, ya que los componentes del cuaternión no son independientes.

$$K \mathbf{q} = \lambda \mathbf{q}. \quad (4.48)$$

La Ecuación (4.48) es un problema de eigenvalores y por tanto la orientación óptima es un eigenvector de la matriz K . Sin embargo hay cuatro eigenvalores, cada uno con diferentes eigenvectores. Sustituyendo la Ecuación (4.48) en la función de ganancia (Ecuación (4.41)):

$$g(\mathbf{q}) = \mathbf{q}^T K \mathbf{q}, \quad (4.49)$$

$$= \mathbf{q}^T \lambda \mathbf{q}, \quad (4.50)$$

$$= \lambda \mathbf{q}^T \mathbf{q}, \quad (4.51)$$

$$= \lambda. \quad (4.52)$$

La función de ganancia es entonces maximizada cuando λ toma su valor más grande, es decir con el eigenvalor más grande de K , resultando que su eigenvector correspondiente sea el cuaternión de la orientación óptima.

El método Q implica resolver el problema de eigenvectores (4.48) directamente y después determinar cuál es el eigenvalor más grande. Este es un proceso sencillo si se cuenta con una herramienta como MATLAB, pero de lo contrario representa un obstáculo debido a la cantidad de recursos computacionales necesarios para hallar la solución. Sin embargo, la ventaja es que el método Q es uno de los algoritmos que produce mejores resultados.

El algoritmo Davenport Q no tiene una solución única si los dos eigenvalores más grandes de K son iguales. Esto no es un fallo por parte del algoritmo, sino que se carece suficiente información para tener una solución única de la orientación [67].

4.3.3.4 QUEST

El método Q puede llegar a ser difícil de implementar dadas las limitaciones de memoria y procesamiento en la computadora de abordo. El método QUEST (QUaternion ESTimator) ofrece una opción más eficiente al hacer algunas suposiciones. Fue desarrollado por Malcolm Shuster en 1978 [68].

Al sustituir la función de ganancia (4.33) en la función de pérdida (4.31) y después reordenar tras usar la Ecuación (4.52) se obtiene una expresión simplificada para la función de pérdida:

$$\lambda_{opt} = \sum w_k - J. \quad (4.53)$$

El usar el eigenvalor óptimo asegura que la función de pérdida adquiera un valor muy pequeño, por lo que se puede despreciar resultando en

$$\lambda_{opt} \approx \sum w_k. \quad (4.54)$$

El método QUEST en realidad involucra una serie de pasos usando el método de Newton para calcular λ_{opt} [85]. El resultado converge a un valor muy cercano a $\sum w_k$, por lo que en la práctica se omite el cálculo al asumir la Ecuación (4.54) desde el inicio.

Una vez conocido el eigenvalor óptimo se procede a calcular el eigenvector correspondiente, el cual representa el cuaternión de la orientación determinada. Una forma de obtener el cuaternión es a partir del vector de Gibbs [87]. La relación entre el cuaternión \mathbf{q} y el vector de Gibbs \mathbf{p} es:

$$\mathbf{p} = \frac{\hat{\mathbf{q}}}{q_4} = \text{atan} \frac{\phi}{2}. \quad (4.55)$$

El problema de eigenvectores de la Ecuación (4.48) se puede reescribir como el siguiente sistema de ecuaciones:

$$(S - \sigma \mathbf{I}_{3 \times 3}) \hat{\mathbf{q}} + Z q_4 = \lambda \hat{\mathbf{q}}, \quad (4.56)$$

$$Z^T \hat{\mathbf{q}} + \sigma q_4 = \lambda q_4. \quad (4.57)$$

Al dividir la Ecuación (4.56) entre q_4 , haciendo uso de la Ecuación (4.55) y despejando para \mathbf{p} se obtiene

$$\mathbf{p} = [(\lambda_{opt} + \sigma) \mathbf{I} - S]^{-1} Z. \quad (4.58)$$

Existen varias formas de evitar la necesidad de calcular la inversa [68]. Una de ellas es hacer uso de eliminación gaussiana para resolver la ecuación

$$Z = [(\lambda_{opt} + \sigma)\mathbf{I} - S]\mathbf{p}. \quad (4.59)$$

Para convertir el vector de Gibbs de vuelta en un cuaternión se usa:

$$\mathbf{q} = \frac{1}{\sqrt{1 + \mathbf{p}^T \mathbf{p}}} \begin{bmatrix} \mathbf{p} \\ 1 \end{bmatrix}. \quad (4.60)$$

Un problema con este método es que el vector de Gibbs encuentra una singularidad cuando la rotación es de π radianes [85]. Una de las formas en las que el método QUEST evita este problema es mediante el uso de rotaciones secuenciales [87].

Aún con la implementación de medidas que previenen el encuentro con singularidades, el método QUEST representa una forma más eficiente de calcular el cuaternión óptimo con respecto al método Q. La desventaja se encuentra en que el error producido por la función de pérdida es en la mayoría de los casos mayor. Por lo tanto la elección entre el método Q y QUEST depende de las características y objetivos de la misión.

Las características principales de los tres métodos de determinación de orientación presentados se resumen en la Tabla 4.2.

Tabla 4.2: Comparación de las características de los métodos de determinación de orientación estáticos.

	TRIAD	DAVENPORT Q	QUEST
Tipo	Determinista	Estadístico	Estadístico
Representación de orientación	Matriz de rotación	Cuaternión	Cuaternión
No. de vectores de entrada	2	2 o más	2 o más
Principal ventaja	El algoritmo más sencillo de implementar	Produce el resultado más preciso	Más eficiente que Davenport Q computacionalmente
Principal desventaja	No resuelve el problema de Wahba	Computacionalmente demandante	Requiere de medidas para prevenir singularidades
Suposiciones	Un sensor es más preciso que otro	-	La función de pérdida ≈ 0 con el cuaternión óptimo

4.4. Estimación de estado mediante filtros de Kalman

Un estimador de estado realiza una estimación de las variables que definen el estado de un sistema cuando el estado real no puede ser medido directamente. En el caso de un satélite, una variable de interés con estas características suele ser la orientación. Usar únicamente un algoritmo de determinación de orientación estático puede resultar en una divergencia en el resultado si para un instante dado el error de los sensores es significativo. Para contrarrestar esto, los estimadores de estado hacen uso de la información sobre el estado previo, considerándose como métodos de determinación variantes en el tiempo, para obtener una estimación que tenga concordancia con el estado anterior.

La implementación del subsistema de estimación de estado en el modelo virtual hace uso de un filtro de Kalman extendido. Un filtro de Kalman se trata de un estimador óptimo, lo que quiere decir que minimiza la varianza. El filtro de Kalman combina la información recibida de las mediciones de sensores y el modelo de la dinámica conocida del sistema, así como sus incertidumbres asociadas.

4.4.1. Sistemas lineales

Un sistema dinámico lineal y discreto puede describirse como

$$x_k = \mathbf{F}_{k-1}x_{k-1} + \mathbf{G}_{k-1}u_{k-1} + w_{k-1}, \quad (4.61)$$

$$z_k = \mathbf{H}_k x_k + v_k, \quad (4.62)$$

donde x_k representa las variables de estado del sistema en el instante k y z_k las mediciones realizadas a dicho sistema en el mismo instante. \mathbf{F} es la matriz de transición de estado, \mathbf{G} la matriz de control-entrada, la cual mapea el vector de control u , y \mathbf{H} es la matriz de medición [88], en donde se hace el mapeo de las variables de estado a aquellas que realmente se pueden medir a través de los sensores.

El proceso y las mediciones tienen un ruido asociado, w_n y v_n respectivamente, los cuales se asumen son ruidos gaussianos con media cero y sin correlación, por lo que las variables son independientes. Las covarianzas asociadas a ellos son

$$\mathbf{Q} = E(w_n w_n^T), \quad (4.63)$$

$$\mathbf{R} = E(v_n v_n^T), \quad (4.64)$$

donde \mathbf{Q} es la covarianza del ruido del proceso y \mathbf{R} la covarianza del ruido de las mediciones. Además v_n y w_n no tienen correlación entre sí, de tal forma que $E\{v_n w_n^T\} = 0$.

4.4.2. Filtro de Kalman lineal

El filtro de Kalman se puede separar en dos etapas principales. La primera se conoce como la etapa de predicción, extrapolación o *a priori*. Aquí el estado conocido del sistema se propaga mediante un modelo que se asume describe la dinámica del sistema. Dicha acción conlleva a un aumento en la incertidumbre de la estimación, puesto que se desconoce que tan bien el modelo predice el comportamiento de la realidad. Las ecuaciones para la extrapolación de estado \hat{x}_k y extrapolación de la covarianza \mathbf{P} son:

$$\hat{x}_k = \mathbf{F}_{k-1}\hat{x}_{k-1} + \mathbf{G}_{k-1}\mathbf{u}_{k-1}, \quad (4.65)$$

$$\mathbf{P}_k = \mathbf{F}_{k-1}\mathbf{P}_{k-1}\mathbf{F}_{k-1}^T + \mathbf{Q}_{k-1}. \quad (4.66)$$

La segunda etapa es la de corrección o actualización (*a posteriori*). En este paso se actualiza la estimación hecha en la etapa de predicción con la información adquirida mediante las mediciones. Esta información mejora la calidad de la estimación, pero a la vez puede traer incertidumbre adicional debido a la varianza de las mediciones de los sensores. La ganancia de Kalman \mathbf{K} es el parámetro que determina el peso que tienen las mediciones en la estimación. La ganancia se calcula con

$$\mathbf{K}_k = \frac{\mathbf{P}_k\mathbf{H}_k^T}{\mathbf{H}_k\mathbf{P}_k\mathbf{H}_k^T + \mathbf{R}_k}. \quad (4.67)$$

De la Ecuación (4.67) se infiere que entre más grande sea la covarianza de las mediciones, la ganancia adquirirá un valor más pequeño. En consecuencia los valores de las mediciones no tendrán tanta influencia en la estimación subsecuente. En la etapa de corrección, el estado y la covarianza se calculan mediante las Ecuaciones (4.68) y (4.69).

$$\hat{x}_k = \hat{x}_k + \mathbf{K}_k(z_k - \mathbf{H}_k\hat{x}_k), \quad (4.68)$$

$$\mathbf{P}_k = (\mathbf{I} - \mathbf{K}_k\mathbf{H}_k)\mathbf{P}_k(\mathbf{I} - \mathbf{K}_k\mathbf{H}_k)^T + \mathbf{K}_k\mathbf{R}_k\mathbf{K}_k^T. \quad (4.69)$$

En la práctica \mathbf{Q} y \mathbf{R} sirven como parámetros de ajuste para cambiar el comportamiento del filtro, por ejemplo, en la velocidad de convergencia. Es tarea del diseñador elegir los valores que produzcan una respuesta que satisfaga los requerimientos.

4.4.3. Filtro de Kalman extendido

El filtro de Kalman descrito por las ecuaciones de la subsección anterior está pensado para sistemas lineales. Esto se puede ver en las Ecuaciones (4.61) y (4.62), las cuales sirven como punto de partida para las ecuaciones del filtro lineal. En cambio

un sistema no lineal puede ser descrito mediante las siguientes ecuaciones:

$$\mathbf{x}_k = f_{k-1}(\mathbf{x}_{k-1}, \mathbf{u}_{k-1}, \mathbf{w}_{k-1}), \quad (4.70)$$

$$\mathbf{z}_k = h_k(\mathbf{x}_k, \mathbf{v}_k), \quad (4.71)$$

donde f_k y h_k son funciones no lineales [88]. Por lo tanto en sistemas no lineales el uso de un filtro de Kalman lineal no es apropiado. En respuesta a esto se han propuesto varias soluciones, una de ellas siendo el filtro de Kalman extendido o EKF, uno de los filtros de este tipo más utilizados en aplicaciones espaciales.

El filtro de Kalman extendido toma esencialmente la misma estructura que el filtro de Kalman lineal. Las diferencias entre ellos se ven reflejadas en las ecuaciones relacionadas a las variables de estado, tanto en la predicción (Ecuación (4.72)) como en la corrección (Ecuación (4.73)):

$$\hat{\mathbf{x}}_k = f_{k-1}(\hat{\mathbf{x}}_{k-1}, \mathbf{u}_{k-1}, 0), \quad (4.72)$$

$$\hat{\mathbf{x}}_k = \hat{\mathbf{x}}_k + \mathbf{K}_k(\mathbf{z}_k - h_k(\hat{\mathbf{x}}_k, 0)). \quad (4.73)$$

Los cambios con respecto a las ecuaciones equivalentes en el filtro lineal, las Ecuaciones (4.65) y (4.68), se deben a las funciones no lineales mencionadas anteriormente. Las ecuaciones restantes son las mismas que en el filtro de Kalman lineal. El EKF esencialmente linealiza las funciones no lineales durante cada iteración tal que:

$$\mathbf{F}_{k-1} = \left. \frac{\partial f_{k-1}}{\partial \mathbf{x}} \right|_{\hat{\mathbf{x}}_{k-1}}, \quad (4.74)$$

$$\mathbf{H}_k = \left. \frac{\partial h_k}{\partial \mathbf{x}} \right|_{\hat{\mathbf{x}}_k}. \quad (4.75)$$

El filtro de Kalman no es un estimador óptimo a diferencia del filtro de Kalman lineal por lo que puede llegar a divergir debido a estas linealizaciones. Esta es una de las principales limitaciones del filtro, por lo que se tiene que poner especial consideración en su diseño. Pero a pesar de esto permanece como uno de los tipos de filtros más utilizado en aplicaciones espaciales incluyendo la estimación de la orientación [89].

4.4.4. Filtros de Kalman en satélites

Si los algoritmos de determinación de orientación estáticos se usaran por sí mismos en un satélite, solo se estaría usando la información proveniente de los vectores medidos para cada tiempo dado, lo cual aumenta la probabilidad de un resultado erróneo al sumar factores como el ruido de los sensores y las imperfecciones del algoritmo.

El uso de filtros es útil en este caso, puesto que también se toma información de las mediciones previas y se incorpora en la estimación de la orientación. Otra de

las ventajas de usar filtros es que incorporan modelos de la dinámica del sistema, lo cual es una herramienta útil cuando se cuenta con sensores con un bajo nivel de precisión. Sin embargo los algoritmos de determinación estáticos también cuentan con ventajas, razón por la cual es común que ambos métodos se complementen en la implementación [90, 91, 92, 93].

Los filtros de Kalman pueden adoptar varias formas dependiendo del tipo de filtro (lineal, extendido, etc.), las variables de estado, los sensores utilizados, el modelo de predicción de estado y las limitaciones impuestas por los modos de operación en el control de orientación. La gran cantidad de opciones se expande aún más en la literatura, en donde además se proponen técnicas y restricciones teóricas que difieren de las usadas habitualmente en la práctica. Como resultado no existe una implementación del filtro de Kalman estándar, ya que la elección y diseño del filtro a emplear resulta en la consideración de todas las variables anteriores.

Las características de la misión otorgan los primeros indicios del filtro a emplear mediante la imposición de restricciones. Un ejemplo son los sensores que dispondrá el satélite. Hay varios modelos que pueden acomodar el uso de determinados sensores mediante la cuidadosa elección de la matriz \mathbf{H} , pero otros pueden ser descartados en base de los sensores que no serán empleados.

Otro de los factores en la elección del filtro y uno de los que tienen más importancia en la toma de la decisión es la herencia de vuelo. Algunos filtros empleados en artículos científicos pueden llegar a reportar mejores resultados que otros métodos, pero son resultados permanecen en la teoría. Los riesgos que conlleva el lanzamiento del espacio requieren del uso de soluciones que hayan demostrado ser exitosas en misiones anteriores.

Las restricciones del filtro que demanda el nanosatélite propuesto para la simulación son el uso de magnetómetros, sensores solares, GPS y giroscopios. Por lo tanto, las posibles opciones de filtros de Kalman se limitan a usar los tipos de sensores anteriores.

4.4.5. Implementación del filtro de Kalman extendido en el modelo virtual

Dentro de la simulación se incluye un filtro de Kalman extendido dentro del subsistema de estimación de estado. Las variables de estado de interés para el nanosatélite simulado son la velocidad angular y la orientación, ya que ambas variables están directamente relacionadas con los algoritmos para un control de

orientación magnético. El vector de estado sería

$$\mathbf{x} = \begin{bmatrix} \omega_1 \\ \omega_2 \\ \omega_3 \\ q_0 \\ q_1 \\ q_2 \\ q_3 \end{bmatrix}. \quad (4.76)$$

El modelo de transición de estado (Ecuación (4.72)) consiste en la Ecuación de dinámica (2.36) y la Ecuación de cinemática (2.30) respectivamente:

$$\begin{bmatrix} \dot{\boldsymbol{\omega}} \\ \dot{\mathbf{q}} \end{bmatrix} = \begin{bmatrix} \mathbf{I}^{-1}(\mathbf{I}\boldsymbol{\omega} \times \boldsymbol{\omega}) \\ \frac{1}{2}\boldsymbol{\Omega}\mathbf{q} \end{bmatrix}. \quad (4.77)$$

Los pares se omiten del modelo de dinámica, puesto que se desconoce el valor neto, el cual no es constante y tampoco es una variable de estado. Los pares de control, cuyo valor es conocido, se pueden incluir bajo la parte de control como parte del término \mathbf{u}_k , sin embargo se omiten por simplicidad. Como resultado, los valores de la incertidumbre asociada a la transición de estado deben ser actualizados apropiadamente para reflejar esta falta de conocimiento.

El Jacobiano del modelo de transición se calcula con respecto a las variables que componen $\boldsymbol{\omega}$ y \mathbf{q} usando la Ecuación (4.74). Su valor se calculó mediante la herramienta *Symbolic Math Toolbox* de MATLAB.

Para calcular la transición de estado mediante la Ecuación (4.77) se pueden hacer uso de métodos de integración como Euler o Runge-Kutta para una mayor precisión, aunque se puede utilizar el modelo linealizado (Ecuación (4.65)) utilizando el Jacobiano (Ecuación (4.74)), método empleado dentro del modelo. En cualquier caso se tiene que realizar una normalización del cuaternión en cada iteración para evitar errores numéricos.

El filtro de Kalman actuará después de que el algoritmo de determinación estático haya calculado la orientación, por lo que su resultado sirve como entrada al filtro. Por el otro lado, los giroscopios miden la velocidad angular directamente y sus valores podrán ser obtenidos a partir de su respectivo modelo. En ambos casos las variables medidas son las mismas que las variables de estado, por lo que el modelo de medición es la matriz de identidad

$$h(\mathbf{w}, \mathbf{q}) = \mathbf{I}_{7 \times 7}. \quad (4.78)$$

La matriz h es lineal por lo que calcular el Jacobiano no es necesario. En ese caso $\mathbf{H} = h$ en las Ecuaciones (4.69) y (4.67), teniendo en cuenta que no es necesario recalcularla en cada iteración, puesto que su valor permanece constante.

4.5. Algoritmos de control de orientación

Para un nanosatélite de control de orientación magnético destacan dos algoritmos de control: B-dot y apuntamiento a nadir. Ambos corresponden a diferentes modos de operación en los que puede alternar un satélite durante su vida útil para alcanzar objetivos como el de percepción remota. Los bloques para estos algoritmos proporcionados por el equipo de desarrollo de K'oto y fueron implementados como parte del subsistema de control de orientación en el modelo virtual.

4.5.1. B-dot

Después de que un satélite ha sido lanzado del dispensador, este empezará a girar de una forma no controlada, lo cual se conoce como *tumbling*. B-dot es un algoritmo de *detumbling*, lo que quiere decir que su objetivo es reducir la velocidad angular (idealmente a cero), lo cual es necesario antes de controlar la orientación y así poder proceder a otros modos de operación como el apuntamiento [94].

B-dot corresponde a un control de orientación magnético activo, por lo que hace uso del principio del par magnético para proveer control mediante magnetorcas. El par magnético τ_B depende del momento de dipolo magnético m generado por las magnetorcas y el campo magnético de la Tierra B de tal forma que

$$\tau_B = m \times B. \quad (4.79)$$

Por lo tanto, el uso de este algoritmo no requiere de información sobre la orientación, ya que únicamente utiliza mediciones del magnetómetro, mientras que el único tipo de actuadores que se requieren son las magnetorcas [95].

B-dot se encarga de calcular el m necesario en las magnetorcas para generar un par neto que conlleve a la estabilización del satélite. El nombre del algoritmo hace referencia a que utiliza la derivada del campo magnético \dot{B} para conseguirlo. La ley de control se define como

$$m = -k\dot{B}, \quad (4.80)$$

donde k es la ganancia de control. Una versión modificada fue desarrollada por [96] en donde se reemplaza a la ganancia estática por una parametrizada para «usar eficientemente el ciclo de trabajo de las magnetorcas a lo largo de la fase de *detumbling*». En esta versión el momento de dipolo magnético se calcula como

$$m = -\frac{k_b(p)}{|B|} \hat{B}, \quad (4.81)$$

en donde k_b es la ganancia parametrizada en términos del parámetro de *tumble* p . Esta nueva ganancia se calcula como

$$k_b = \frac{k_b^*}{\phi p + \epsilon}, \quad (4.82)$$

donde k_b^* es una ganancia estática subóptima que puede ser calculada a partir de la tasa orbital, la inclinación y el momento de inercia mínimo. ϕ es el factor de tasa y cumple $0 \leq \phi < \infty$; ϵ es el parámetro de ajuste de tal forma que $0 < \phi \leq 1$. Mientras que p cumple $0 \leq p \leq 1$ y puede calculado mediante la Ecuación (4.83). Para más información sobre estas variables se dirige hacia [96].

$$p \approx \frac{|\omega|}{\sqrt{3}|\omega_{max}|}. \quad (4.83)$$

\dot{B} se puede calcular a partir de la diferencia en las mediciones del magnetómetro, sin embargo también es posible calcularla si se conoce la velocidad angular en el marco fijo al cuerpo mediante la Ecuación (4.84) [97]:

$$\dot{B} = -B \times \omega. \quad (4.84)$$

4.5.2. Apuntamiento a nadir

Una vez estabilizado el satélite, será posible cambiar al modo de operación de apuntamiento. El objetivo del algoritmo de apuntamiento es orientar al satélite de tal forma que siempre se esté apuntando hacia una dirección en particular, en este caso a nadir, con lo cual la carga útil podrá alinearse con un objeto de estudio como la Tierra. Mediante el algoritmo de apuntamiento se podrá mantener esta orientación a lo largo de la órbita a pesar de las perturbaciones del ambiente.

En su versión inercial, uno de los algoritmos propuestos consiste en un control PD con dos modos de estabilización: retroalimentación de estado completo, el cual considera la orientación y la velocidad angular; y retroalimentación de estado parcial, la cual solo considera la orientación [98]. Una versión para el caso de apuntamiento a nadir fue desarrollada a partir de este apuntamiento inercial por [71]. En el caso de retroalimentación de estado completo el vector de control u se define como

$$u = -[\epsilon^2 k_1 \hat{q}_o + \epsilon k_2 (\omega_o)], \quad (4.85)$$

donde q_o se define con respecto al marco de referencia orbital, ω_o es la velocidad angular definida también en el marco orbital, k_1 y k_2 son ganancias positivas, y ϵ es un parámetro positivo suficientemente pequeño [99], el cual es diferente al usado en

el B-dot. El momento de dipolo magnético puede ser calculado a partir de \mathbf{u} :

$$\mathbf{m} = \mathbf{B} \times \mathbf{u}. \quad (4.86)$$

Esta operación también puede llevarse a cabo al usar la matriz antisimétrica de \mathbf{B} para mayor conveniencia.

El apuntamiento a nadir requiere de conocer la orientación en el marco de referencia orbital para calcular el vector de control conforme a la Ecuación (4.85). Sin embargo, los modelos de determinación estáticos dentro del modelo virtual solo permiten la obtención de la orientación con respecto al marco de referencia ECI o el marco de referencia fijo al cuerpo. Por ello se requieren rotaciones para llegar al marco de referencia orbital a partir de estos marcos de referencia iniciales. La matriz de rotación usada para rotar el marco de referencia ECI al marco de referencia orbital se define como

$$\mathbf{R}_{i2o} = \begin{bmatrix} \cos(\Omega)\cos(u) - \sin(\Omega)\cos(i)\sin(u) & \sin(\Omega)\cos(u) + \cos(\Omega)\cos(i)\sin(u) & \sin(i)\sin(u) \\ -\cos(\Omega)\sin(u) - \sin(\Omega)\cos(i)\cos(u) & -\sin(\Omega)\sin(u) + \cos(\Omega)\cos(i)\cos(u) & \sin(i)\cos(u) \\ \sin(\Omega)\sin(i) & -\cos(\Omega)\sin(i) & \cos(i) \end{bmatrix}, \quad (4.87)$$

donde Ω es el RAAN de la órbita, i la inclinación y u el argumento de altitud, el cual se puede calcular a partir de la anomalía verdadera y el argumento de perigeo. Estos datos pueden ser obtenidos mediante un propagador, y en el modelo virtual se manejan dentro del subsistema ambiental y orbital. El uso de la Ecuación (4.87) asume que el marco orbital se define con el eje \hat{x} en dirección radial de la órbita, el eje \hat{y} en dirección de la velocidad tangencial de la órbita y el eje \hat{z} perpendicular al plano orbital [100]. Debido a la diferencia en la definición del marco orbital/LVLH en algunos propagadores y software de análisis de misión, se deben hacer ajustes en los datos de salida para que uno de los ejes esté definido en dirección de nadir. La derivación de la Ecuación (4.87) se describe en [75, 100, 101, 102].

El subsistema de control de orientación de la simulación incluye el B-dot y el apuntamiento a nadir. La ejecución y lógica de cambio entre los modos de operación que incluyen dichos algoritmos están definidas dentro de este subsistema. Cuando la velocidad angular es mayor al umbral definido, la ley de control será B-dot. Una vez que la velocidad angular se encuentre por debajo de dicho valor se ejecutará el apuntamiento.

4.6. Componentes electrónicos

El modelado de componentes electrónicos también es necesario para otorgar mayor precisión a las simulaciones. Si bien los algoritmos se pueden ejecutar sin considerar estos modelos, sus resultados dependen en gran medida de las entradas que les son alimentados. En la vida real dichas entradas son proporcionadas por los sensores que a su vez están limitados por factores como el ruido, frecuencia de

muestreo y saturación. En el caso de los actuadores se aplica la misma lógica debido a que no son capaces de proporcionar una salida ideal debido a las limitaciones en el hardware. Como consecuencia, modelar los sensores y actuadores permite imponer restricciones a las simulaciones para tener una mayor similitud con el comportamiento real del satélite.

Dentro del modelo virtual, el modelado de componentes electrónicos afectará directamente los subsistemas de determinación de orientación, control de orientación y estimación de estado. Los modelos dentro de esta categoría se dividen en sensores y actuadores.

4.6.1. Sensores

Como parte de la configuración del nanosatélite simulado se considera que los sensores principales son los magnetómetros y sensores solares, ya que son la fuente principal de la información requerida para la determinación y control de orientación. Sin embargo, también se considerarán GPS y giroscopios. El propósito de estos sensores adicionales es el de redundancia y el proveer información auxiliar para los algoritmos. A continuación se describirán los modelos para los cuatro sensores.

4.6.1.1 SENSORES SOLARES

Los sensores solares se modelaron como sensores solares analógicos. En un sensor solar analógico la corriente de salida I es proporcional al coseno del ángulo de incidencia θ de la radiación solar, de tal forma que

$$I = I_{max} \cos(\theta), \quad (4.88)$$

donde el ángulo θ es el ángulo entre la normal de la superficie del sensor \mathbf{n} y el vector solar \mathbf{s} a determinar. I_{max} es la corriente máxima que puede pasar por los sensores (indicando que el sol se encuentra a 0° del respectivo sensor). La Ecuación (4.88) se puede reescribir en términos de \mathbf{n} y \mathbf{s} :

$$I = I_{max}(\mathbf{n} \cdot \mathbf{s}). \quad (4.89)$$

Entonces, un sensor analógico tiene como salida un ángulo en el cual el vector solar puede encontrarse con respecto a la normal del sensor. Siendo este ángulo la única información que se tiene sobre el vector solar, se tiene como resultado un cono —con la normal en su centro— representando las posibles ubicaciones del vector solar. Al recibir lecturas de dos sensores distintos se tienen dos conos, uno para cada sensor, donde la intersección de ambos conos reduce las opciones a dos posibles vectores solares.

Se hace uso de la Ecuación (4.89) para las dos lecturas, junto con la restricción sobre la esfera unitaria para obtener las dos soluciones. Sin embargo, se necesita más

información para poder determinar cuál de ellas es el vector solar verdadero, por lo que se suele hacer una predicción con base en el valor previo [78].

En el caso de tres sensores, la intersección entre los tres conos resulta en un vector solar único asumiendo que el error de medición es cero [68]. La solución se puede encontrar directamente a partir del sistema de ecuaciones definido por la Ecuación (4.89). Por lo tanto, se requieren como mínimo dos sensores analógicos para determinar el vector solar, sin embargo se suelen usar más para garantizar la presencia de la señal, puesto que únicamente tres caras del satélite son iluminadas en un momento dado por la luz directa del sol.

Asumiendo que existen seis sensores analógicos en el satélite y cada uno se encuentra en una cara distinta, de tal forma que tres sensores tengan la normal negativa de los tres restantes (por ejemplo $\mathbf{n}_i = -\mathbf{n}_i$), se tiene el siguiente sistema

$$\begin{bmatrix} I_i - I_{-i} \\ I_j - I_{-j} \\ I_k - I_{-k} \end{bmatrix} = I_{max} \begin{bmatrix} \mathbf{n}_i \cdot \mathbf{s} \\ \mathbf{n}_j \cdot \mathbf{s} \\ \mathbf{n}_k \cdot \mathbf{s} \end{bmatrix} = I_{max} \begin{bmatrix} \mathbf{n}_i^T \\ \mathbf{n}_j^T \\ \mathbf{n}_k^T \end{bmatrix} \mathbf{s}. \quad (4.90)$$

Cada sensor solo tiene como salida una corriente cuando el ángulo entre el vector solar y la normal está en el rango $(-90^\circ, 90^\circ)$, en los demás casos la salida es cero. Al tomar la diferencia entre las corrientes de cada par de sensores se asegura una salida de corriente para todos los ángulos. El vector solar puede encontrarse de la siguiente forma siempre y cuando las normales no sean coplanares [67]:

$$\mathbf{s} = \frac{1}{I_{max}} \begin{bmatrix} \mathbf{n}_i^T \\ \mathbf{n}_j^T \\ \mathbf{n}_k^T \end{bmatrix}^{-1} \begin{bmatrix} I_i - I_{-i} \\ I_j - I_{-j} \\ I_k - I_{-k} \end{bmatrix}. \quad (4.91)$$

La simulación de los sensores solares requiere subsecuentemente de conocer las corrientes, por lo que se necesitaría realizar una simulación al nivel de electrónica que considere a su vez un modelo de iluminación que incorpore elementos de la geometría del satélite y la órbita. Sin embargo, un enfoque más práctico consiste en determinar un vector solar «verdadero» obtenido mediante un propagador o software afín y posteriormente implementar las características del sensor en un modelo para obtener una salida lo más parecido a la realidad.

La implementación del modelo del sensor solar para este trabajo consistió en dos variantes: trabajar a nivel de corrientes y trabajar a nivel de los ángulos de los sensores. En la primera variante se parte del vector solar verdadero y se utilizó la Ecuación (4.90) para obtener las corrientes correspondientes. Después se añade un ruido para simular la variación al nivel de corrientes y posteriormente se aplica la Ecuación (4.91) para obtener el vector solar nuevamente.

La segunda variante consiste en añadir las restricciones impuestas por los sensores, donde se tomó como referencia el trabajo de [49], [51] y [69]. Este enfoque

asume sensores solares analógicos de tipo de celda doble, con lo cual la salida de los sensores previo a la conversión al vector solar serían dos ángulos. Para esta variante primero se transforma el vector solar verdadero de coordenadas cartesianas a esféricas, dando como resultado la obtención directa estos ángulos. Se procede a añadir un ruido blanco con la varianza correspondiente a las características del sensor. Puesto que los sensores solares solo miden la dirección del vector (tratándose de un vector unitario) se desprecia el componente radial de las coordenadas esféricas. El modelo del sensor devolvería entonces dos ángulos, y para obtener el vector solar se tendría que transformar de nuevo a coordenadas cartesianas, simulando el procesamiento llevado a cabo en la computadora a bordo. Por último, se evalúa mediante una señal si el satélite está en condición de eclipse o no. En el caso de eclipse no se regresa una señal mientras que en el caso opuesto se regresa el vector calculado. Posterior a esto se hace un muestreo de la señal.

La desventaja de ambas variantes es la dependencia a tener un vector solar como entrada en el modelo, lo cual dificulta la posterior implementación del algoritmo hacia la computadora de a bordo. Una de las diferencias entre ambas es que la segunda variante corresponde a sensores solares de celda doble, comparado con los sensores de celda única que asume la Ecuación (4.91). A pesar de ser dos enfoques que corresponden a hardware distinto, la implementación de ambos es de utilidad en el modelo, puesto que se proporcionan ambos tipos de incertidumbre, a nivel de mediciones de corriente y a nivel de mediciones de ángulos, lo cual ayudará a demostrar el efecto del filtro extendido de Kalman, además de añadir flexibilidad a la simulación al proporcionar más opciones.

La implementación del modelo en Simulink se muestra en la Figura 4.1. Las variantes se muestran en las Figuras 4.2 y 4.3.

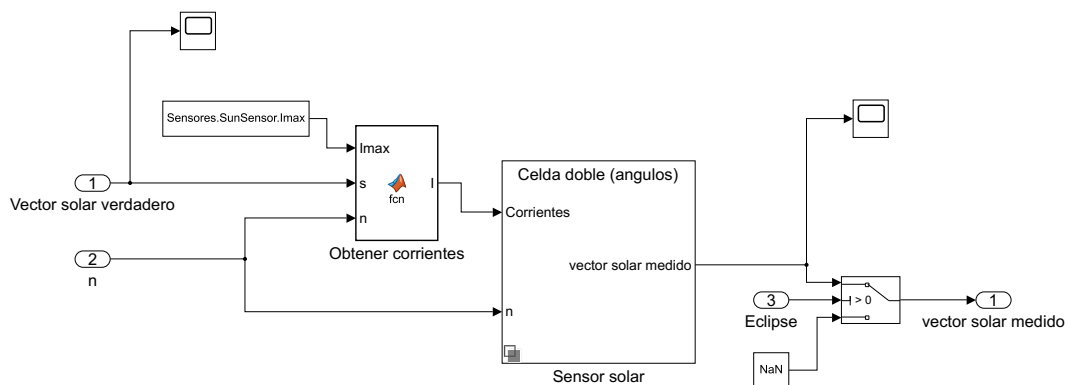


Figura 4.1: Modelo implementado de los sensores solares en Simulink. En este nivel es posible seleccionar el modelo variante a utilizar.

4.6.1.2 MAGNETÓMETRO

Los magnetómetros son sensores con los cuales se puede obtener un vector al proveer la dirección y magnitud del campo magnético. Son de uso confiable, ligeros,

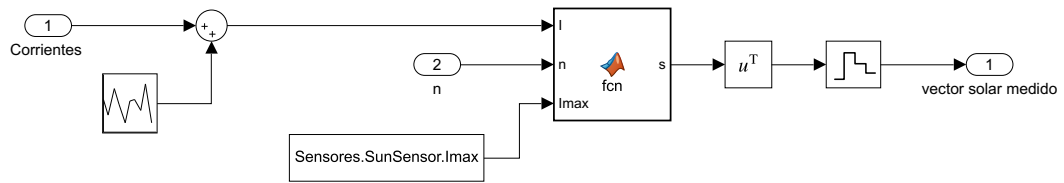


Figura 4.2: Primer modelo variante de los sensores solares, en donde el procesamiento ocurre a nivel de corriente.

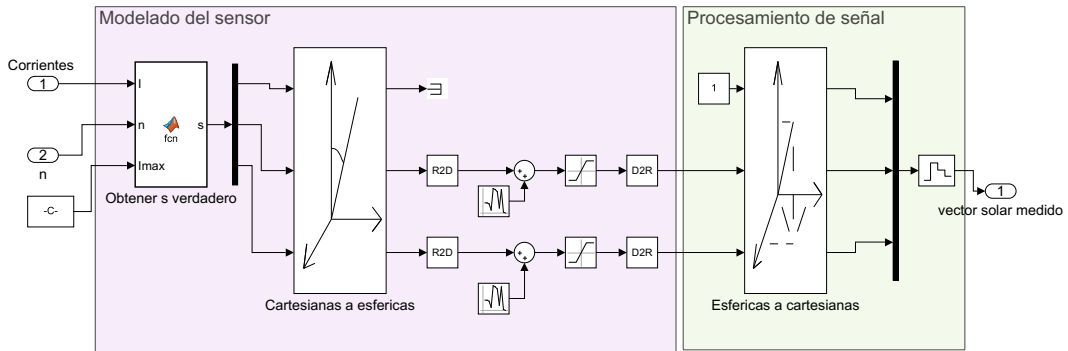


Figura 4.3: Segundo modelo variante de los sensores solares, en donde el procesamiento ocurre a nivel de ángulos.

tienen bajos requerimientos de energía, operan en un rango amplio de temperatura y no tienen partes móviles. Puesto que se habla de un nanosatélite de control de orientación magnético, el magnetómetro permitirá obtener los valores necesarios del campo magnético para poder generar la señal de control. El algoritmo B-dot depende directamente de las mediciones del magnetómetro para poder calcular la derivada del campo magnético, mientras que para el apuntamiento se utiliza para calcular el momento de dipolo magnético necesario. Además, el magnetómetro también se utiliza para la determinación de orientación.

La implementación del modelo del magnetómetro se basa en el mismo enfoque que el de los sensores solares, en donde se parte de una señal verdadera y se modifica para simular la respuesta de un sensor real. La señal verdadera está definida con respecto al marco de referencia ECI, por lo que para simular las mediciones con respecto al marco de referencia fijo al cuerpo se hace una rotación utilizando la orientación verdadera que se calcula a lo largo de la simulación.

Pero a diferencia de los sensores solares, el magnetómetro regresa el campo magnético directamente sin la necesidad de cálculos adicionales. Entre las características que se incorporaron a la señal verdadera se encuentran un sesgo (*bias*), ruido y saturación. La implementación se muestra en la Figura 4.4.

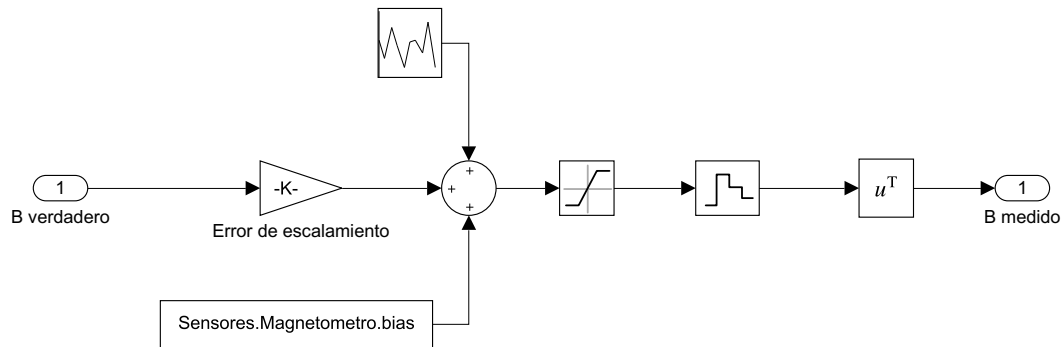


Figura 4.4: Modelo implementado del magnetómetro en Simulink.

4.6.1.3 GIROSCOPIO

El giroscopio juega un papel importante en los algoritmos de control de orientación, ya que es mediante los valores de las velocidades angulares con las cuales se determinan los modos de control, por ejemplo en la ejecución del algoritmo B-dot cuando las velocidades angulares son lo suficientemente altas.

En las simulaciones, el valor verdadero de la velocidad angular se obtiene a partir del modelo de la dinámica, el cual se usa internamente para calcular cantidades que dependen del estado verdadero del satélite. El modelo del giroscopio (Figura 4.5) toma esta velocidad angular verdadera y posteriormente aplica los bloques necesarios para simular las características del sensor. Por ejemplo se añade un ruido blanco, lo que permite que las mediciones de cada eje permanezcan independientes entre sí, y se aplica el muestreo correspondiente.

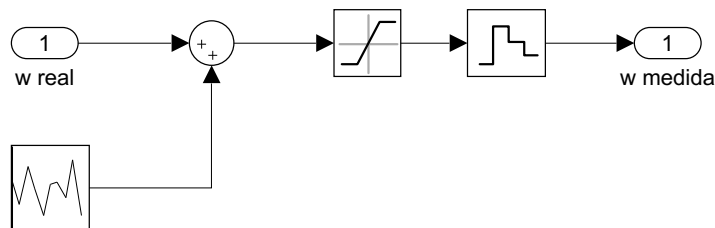


Figura 4.5: Modelo implementado del giroscopio en Simulink.

Un problema que padecen los giroscopios es la presencia de deriva o *drift*, con el cual el error en uno de los ejes se acumula progresivamente y por lo tanto las mediciones empiezan a divergir sustancialmente de los valores reales. Para tomar en cuenta este fenómeno se incrementó la magnitud del ruido, aunque es recomendable modelarlo explícitamente en una futura iteración.

4.6.1.4 GPS

El GPS permite determinar la posición del satélite a lo largo de su órbita. Mediante sus datos será posible calcular el campo magnético inercial mediante el modelo IGRF.

El modelo utilizado para el GPS fue adaptado de [69], trabajo en donde se utilizó el GPS OEM615. El GPS del nanosatélite simulado para este trabajo se modeló con base en un GPS de la familia OEM7, el cual no solo comparte varias características con el OEM615, sino que también comparte varias de sus limitaciones. Según la documentación del firmware de la familia OEM7 el error vertical es al menos 1.5 mayor que el horizontal, hecho que también está presente en la documentación de la familia OEM6 [103, 104]. El GPS tiene como salida las coordenadas LLA, sin embargo la hoja de datos especifica características como el error en término de metros. Esto combinado con la diferencia entre el error vertical y horizontal permiten generar un modelo para el error. Cabe mencionar que el error depende en gran medida del proveedor de servicio seleccionado. La implementación de este modelo se muestra en la Figura 4.6.

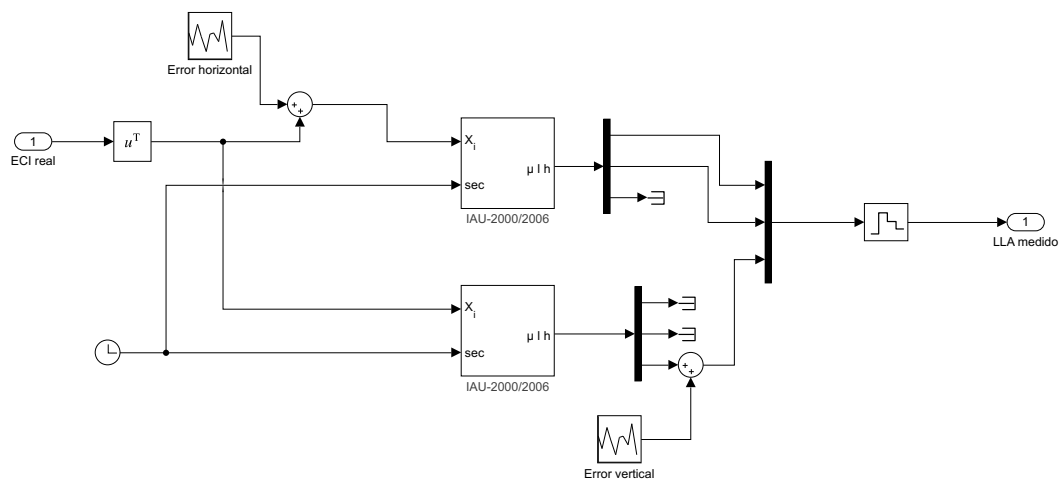


Figura 4.6: Modelo implementado del GPS en Simulink.

El modelo del GPS recibe como entrada las coordenadas de posición reales en ECI, con lo cual se puede agregar el ruido en términos de metros. Se hace una separación en dos grupos, lo cual corresponderá a la latitud y longitud (componentes horizontales) y la altitud (componente vertical), al momento de convertir hacia coordenadas LLA para mantener un error independiente entre ambos. Por último, a la señal resultante al combinar ambos resultados se le aplica un bloque de muestreo.

Una desventaja de este modelo se encuentra en el uso de dos bloques de conversión de coordenadas ECI a LLA, ya que se trata de un cálculo complejo que puede generar un cuello de botella en la simulación. Por esta razón se incluyó un modelo variante con un enfoque similar al de la Figura 4.2, el cual podrá ser usado como una alternativa con menor uso de recursos computacionales. El procesamiento para este modelo (Figura 4.7) se realiza en las mismas unidades de LLA, grados y metros respectivamente.

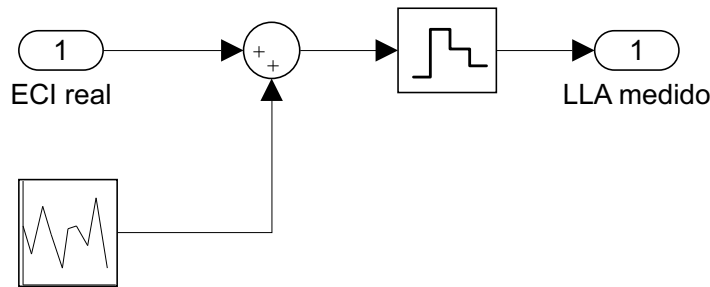


Figura 4.7: Modelo simplificado del GPS en Simulink.

4.6.2. Actuadores: Magnetorcas

El único tipo de actuadores dentro del satélite simulado son las magnetorcas. El momento magnético, el cual definirá el par de control mediante la Ecuación (4.79), es proporcional a la corriente que se haga circular a través de las magnetorcas. Esta relación se resume en la siguiente ecuación

$$m = NIA, \quad (4.92)$$

donde m es el momento magnético, N es el número de vueltas, I es la corriente que pasa en el lazo de la bobina y A es el vector de área. En la implementación, los algoritmos de control de orientación indicarán la cantidad de voltaje a producir mediante un pulso PWM, con lo cual se hará llegar una corriente a las magnetorcas. El par producido por las magnetorcas diferirá del par teórico calculado por los algoritmos de control debido a las restricciones que imponen las características de las magnetorcas y la señal PWM.

Algunos de los parámetros que pueden ser ajustados para el modelo de las magnetorcas incluyen el área y el número de vueltas. El modelo implementado incluye dos tipos de magnetorcas, las cuales difieren en estas características. El modelo se muestra en la Figura 4.8.

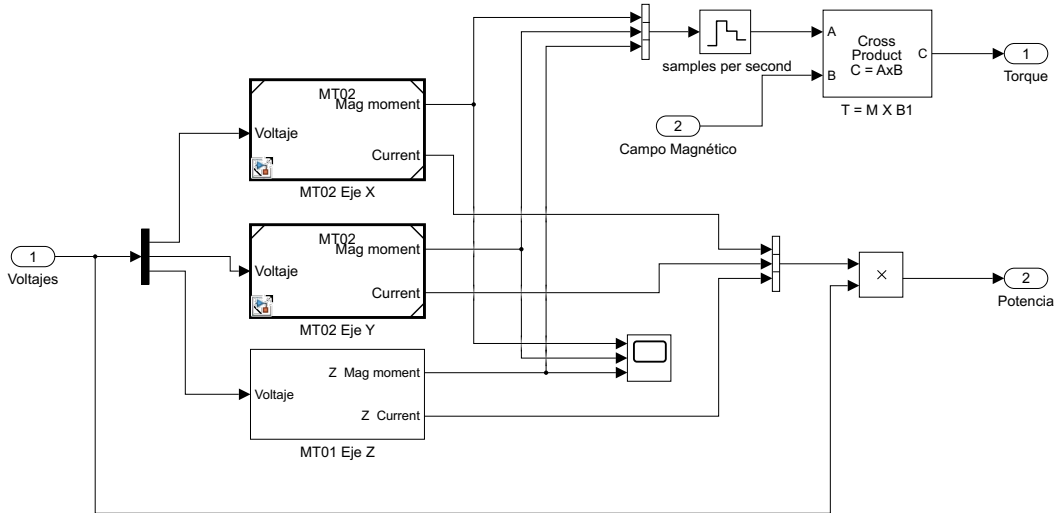


Figura 4.8: Modelo implementado de las magnetorcas en Simulink.

5

Resultados

5.1. Características del modelo virtual

LA HERRAMIENTA desarrollada durante este proyecto integra exitosamente los modelos mencionados a lo largo de las secciones anteriores. El enfoque bajo el diseño basado en modelos permitió el desarrollo independientemente de cada uno de ellos, con lo cual después fue posible enlazarlos bajo una simulación principal que corresponde a la representación de un ADCS en su totalidad. Algunas de las características que definen a la herramienta se presentan en la Tabla 5.1.

Uno de los principales atractivos del modelo virtual es que sus funcionalidades están enfocadas específicamente hacia los ADCS. Esto permite un desarrollo más rápido en esta área a comparación de una herramienta más general, en donde debe haber una búsqueda y selección más exhaustiva para encontrar las funcionalidades que corresponden al análisis de un ADCS. Otras de las ventajas se enfocan en su uso futuro, ya que la modularidad y la reusabilidad permiten adaptar la herramienta a otros proyectos mediante un cambio de parámetros o en los mismos subsistemas. Además, es posible parametrizar las características de interés de los casos de estudio, por ejemplo en las características de los sensores o las condiciones iniciales, para observar sus diferentes efectos en las simulaciones.

Tabla 5.1: Ventajas de la herramienta desarrollada durante este trabajo.

CARACTERÍSTICA	DESCRIPCIÓN
Enfocada a un problema	El enfoque principal se encuentra en el análisis del ADCS de un satélite, lo que permite centrarse en las funcionalidades directamente relacionadas a él en contraste con un enfoque más general, el cual puede generar ruido al usuario.
Control y modos de operación	Es posible implementar modos de operación en el controlador para incorporarlos dentro la simulación, característica ausente en software comerciales.
Modularidad	Cada subsistema es un módulo que puede conectarse con el resto de la simulación en términos de entradas y salidas. Es posible usar modelos variantes sin modificar el resto de los módulos.
Reusabilidad	Los parámetros de la herramienta pueden modificarse de acuerdo a las características del caso a analizar. Las capacidades de la reusabilidad aumentan con la modularidad.
Validación y verificación	Es posible generar código a partir de los modelos para realizar las pruebas de Hardware-in-the-Loop en el microcontrolador. Además se pueden incorporar pruebas unitarias para la parte de verificación.
Documentación	El entorno gráfico y las relaciones entre componentes sirven como una forma de documentación. Simulink posee funcionalidades para generar documentación estandarizada a partir de los modelos.
Integración con herramientas externas	Los resultados de las simulaciones pueden enlazarse con otras aplicaciones compatibles con MATLAB y Simulink tanto elaboradas propiamente como elaboradas por terceros.
Menor curva de aprendizaje	El ambiente de MATLAB/Simulink representa un ambiente familiar para los estudiantes, por lo que la curva de aprendizaje se disminuye en contraste con los software comerciales.
Control de versiones	Otros software existentes utilizan formato binario para sus archivos, lo cual dificulta la integración de un control de versiones. En la herramienta desarrollada el control de versiones es una funcionalidad nativa, lo cual apoya la trazabilidad.

5.2. Caso de estudio: K'oto

El caso de estudio utilizado para demostrar las funcionalidades de la herramienta es la misión de K'oto, el cual es un nanosatélite cuyo nombre en otomí significa chapulín o saltamontes y que se plantea que sea puesto en órbita el primer semestre del 2022. El proyecto está a cargo de la Unidad de Alta Tecnología (UAT) de la UNAM y tiene como misión percepción remota: la toma fotografías de resolución baja-mediana del territorio mexicano y transmitir las a tierra [105].

Además, se tiene como objetivo adicional el fomentar la formación de recursos humanos en el área espacial y cerrar la brecha tecnológica en el desarrollo de satélites pequeños en México al desarrollar un proyecto de esta índole. El satélite será puesto en órbita desde la Estación Espacial Internacional (ISS por sus siglas en inglés) y se cuenta con la colaboración del Instituto de Tecnología de Kyushu, el cual será el encargado de realizar las pruebas de certificación de vuelo.

5.2.1. Sistema de determinación y control de orientación de K'oto

Para nano y picosatélites en Órbita Terrestre Baja (LEO), los sistemas de control de orientación más usados son los magnéticos [54]. K'oto no es la excepción, ya que será puesto en una órbita tipo LEO y hará uso de un sistema de control de orientación activo mediante magnetorcas para regular su orientación. Un sistema de control de esta naturaleza puede ser implementado a través de tres magnetorcas y tres magnetómetros colocados ortogonalmente entre sí.

Además de los magnetómetros, se hará uso de sensores solares para llevar a cabo la determinación de orientación. Como forma de redundancia se tendrán sensores adicionales como giroscopios y GPS, los cuales además pueden proveer información adicional para algoritmos como los de control.

En cuanto a los algoritmos de determinación de orientación, se hará uso de un algoritmo de determinación estático para obtener una orientación a partir de las lecturas de los sensores. Este valor, junto con las lecturas de la velocidad angular de los giroscopios se procesarán mediante un filtro de Kalman con el fin de realizar una estimación de estado. Para el control de orientación se busca tener al menos dos modos, uno de reducción de la velocidad angular para estabilizar el satélite y otro de apuntamiento con el fin de que el satélite llegue a la orientación deseada para realizar su misión (percepción remota).

En resumen, la configuración del sistema de determinación y control de orientación de K'oto coincide con los modelos desarrollados durante el Capítulo 4.

5.3. Simulaciones

Los subsistemas de la simulación están interconectados bajo un mismo modelo, por lo que la órbita elegida tendrá repercusiones directas en los subsistemas que dependen de ella, y estos a su vez afectarán a sus dependientes aunque sus entradas

no estén directamente relacionadas con la órbita. La órbita de K'oto será la misma que la de la ISS, así que para la obtención de parámetros orbitales se seleccionó la órbita de la ISS directamente desde la base de datos de STK. Con fines de la simulación se asumió que el satélite será puesto en órbita el primero de marzo de 2022, a las 2 a.m. UTCG. Las condiciones iniciales de la órbita de acuerdo a esta fecha se muestran en la Tabla 5.2.

Tabla 5.2: Elementos keplerianos de la ISS en el momento $t = 0$ de la simulación.

ELEMENTO KEPLERIANO	VALOR
Semieje mayor	6791.384 km
Excentricidad	0.000129
Inclinación	51.732°
RAAN	147.6160°
Argumento de perigeo	132.3326°
Anomalía verdadera	47.8284°

Para la obtención de los demás datos, obtenidos también mediante STK, que utilizará el subsistema orbital y ambiental del modelo virtual se utilizó la misma fecha inicio y una duración de tres días. Por lo que el rango de datos disponibles corresponde a 01 Mar 2022 02:00:00.000 - 04 Mar 2022 02:00:00.000. El muestreo de estos datos se definió como 1 segundo. Para obtener datos a una frecuencia mayor, correspondiendo a la que necesitarán la mayoría de los bloques, se hizo uso del comportamiento de interpolación de Simulink. El bloque usado para importar los datos de STK es «From Workspace», el cual ofrece una interpolación lineal usando los valores adyacentes conocidos para calcular los datos faltantes. Por otra parte, el no usar la interpolación resulta en que se use el último valor conocido [106]. La interpolación lineal fue usada durante las simulaciones.

5.3.1. Condiciones iniciales de la simulación

Las condiciones iniciales de la simulación se muestran en la Tabla 5.3. Dentro de estos valores se incluyen las condiciones de cinemática, así como características del satélite como la matriz de inercia, la cual permanece constante dado que se asume un cuerpo rígido. La fecha de inicio corresponde con el inicio del rango de los datos de STK, mientras que no se define una fecha de finalización, puesto que la simulación está definida en términos de duración. En este caso se eligió una duración equivalente a 1.5 días lo cual equivale a aproximadamente 24 órbitas de la ISS.

Los valores de la velocidad angular se definieron como $[0.35 \quad -0.35 \quad 0.35]$ rads/s, los cuales representan una aproximación de lo ocurrido en la fase del *tumbling* tras la liberación del satélite en el dispensador. La orientación inicial se eligió arbitrariamente, puesto que se desconoce la orientación tras la liberación. Para el control de orientación se tiene la opción de definir un tiempo de retraso y para este caso, se definió que el control se iniciara a la par con el inicio de la simulación.

Las simulaciones hicieron uso del solucionador de ecuaciones de Simulink ode4, el cual es un solucionador de paso fijo que hace uso del método Runge-Kutta. Para

Tabla 5.3: Condiciones iniciales de la simulación.

CARACTERÍSTICA	VALOR
Matriz de inercia	$\begin{bmatrix} 5.0548 & -3.2915 & -0.043565 \\ -3.2915 & 4.4637 & 0.056644 \\ -0.043565 & 0.056644 & 7.7289 \end{bmatrix} \times 10^{-3} \text{ kg m}^2$
Orientación inicial	$[0 \ 0 \ 0 \ 1]$
Velocidad angular inicial	$[0.35 \ -0.35 \ 0.35] \text{ rad/s}$
Fecha de inicio de la simulación	1 de marzo de 2022 02:00:00
Tiempo total de simulación	129600 s
Solucionador	ode4
Paso del solucionador	0.0001 s
Tiempo en el que se activa el control	0 s

tener la capacidad de generar código y ejecutarlo dentro de un sistema de tiempo real, como es el caso de las técnicas de verificación de ADCS en el MBD, se debe elegir un solucionador de este tipo, ya que los pasos variables no son compatibles con los relojes de tiempo real [107].

Los valores elegidos para los parámetros de los sensores se muestran en la Tabla 5.4, mientras que los parámetros relacionados con el control de orientación se muestran en la Tabla 5.5. En esta última también se incluyen las características de los actuadores. Para los sensores solares se eligió el modelo variante que trabaja en términos de corrientes, mientras que se asumió una ausencia de eclipse. En el caso del GPS se optó por el modelo simplificado, buscando que los parámetros generaran una respuesta similar al modelo completo. Los valores iniciales del filtro de Kalman bajo el subsistema de estimación de estado se discuten en la Sección 5.3.5.

Tabla 5.4: Características de los sensores simulados.

SENSOR	CARACTERÍSTICA	VALOR
Sensores solares	Varianza de ruido	$[10 \ 10 \ 10] \text{ mA}^2$
	Corriente máxima	500 mA
	Muestreo	0.25 s
Magnetómetro	Sesgo	$[0 \ 0 \ 0] \text{ nT}$
	Saturación	80000 nT
	Factor de error de escalamiento	$[1.046 \ 1.125 \ 1.161]$
	Varianza de ruido	$[20 \ 20 \ 20] \times 10^5 \text{ nT}^2$
	Muestreo	0.25 s
Giroscopio	Saturación	-8.72 a 8.72 rad/s
	Varianza de ruido	$[0.001 \ 0.001 \ 0.001] \text{ rad}^2/\text{s}^2$
	Muestreo	0.25 s
GPS	Varianza de ruido (latitud y longitud)	$[1 \ 1] \text{ deg}^2$
	Varianza de ruido (altitud)	20 m ²
	Muestreo	0.25 s

Tabla 5.5: Parámetros relacionados con el control de orientación.

ÁREA	PARÁMETRO	VALOR
B-dot	k_b	8.6593×10^{-6}
	ϕ	12
	ϵ	0.2
Apuntamiento	k_1	3
	k_2	10
	ϵ	0.001
Magnetorcas	N	[49 -49 212]
	A	$[3.12 \ 3.12 \ 1.733] \times 10^{-3} \text{ m}^2$
	r	$[7.2 \ 7.2 \ 4.7] \ \Omega$

5.3.2. Elección de modelo de densidad

La elección del modelo de densidad usado para calcular el par aerodinámico depende de las características de la misión conforme lo descrito en la Sección 4.2. Estas características son conocidas previo a la ejecución de las simulaciones, por lo que el modelo de densidad se debe elegir también con anterioridad. Una comparación de los resultados de todos los modelos de densidad incluidos en el modelo virtual, bajo el rango de tiempo definido previamente y de acuerdo a la órbita de K'oto, se muestran en la Figura 5.1.

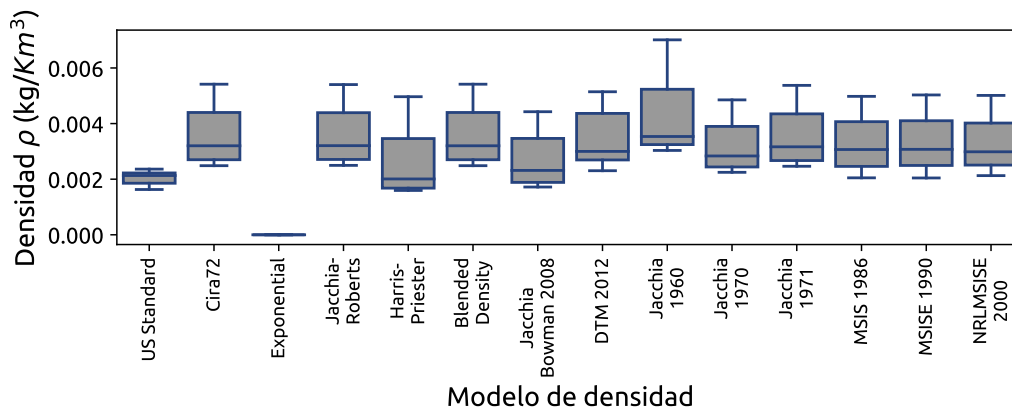


Figura 5.1: Comparación de los resultados de los modelos de densidad.

Se pueden tener varios criterios para la selección del modelo de densidad, el primero siendo la varianza. Para la verificación de un sistema se desea comprobar que los requerimientos sean satisfechos aun cuando existan variaciones del entorno, por lo que es conveniente hacer las simulaciones en aquellos casos en donde los parámetros presenten una mayor varianza. Siguiendo este criterio, se selecciona el modelo Jacchia 1960.

Otro criterio a utilizar es la aparición de los modelos en la literatura, haciendo analogía con la herencia de vuelo. El elegir un modelo de densidad con antecedentes permite una mayor facilidad al comparar resultados con lo ya reportado, además de que existirán más recursos en cuanto a herramientas y referencias durante la implementación de dicho modelo de densidad, lo cual será útil a la hora de alejarse por completo de software propietario como STK para la obtención de estos datos. De acuerdo a lo anterior se selecciona el modelo Jacchia-Bowman 2008.

La estadística descriptiva de ambos modelos se presenta en la Tabla 5.6. En los resultados por presentar se escogió el modelo Jacchia-Bowman 2008.

Tabla 5.6: Estadística descriptiva de los modelos de densidad seleccionados: Jacchia 1960 y Jacchia-Bowman 2008. Unidades son kg km^{-3} .

	JACCHIA 1960	JACCHIA-BOWMAN 2008
Media	4.21×10^{-3}	2.66×10^{-3}
Desviación estándar	1.21×10^{-3}	0.862×10^{-3}
Mínimo	2.12×10^{-3}	1.71×10^{-3}
Máximo	5.01×10^{-3}	4.42×10^{-3}

5.3.3. Cinemática y dinámica

La velocidad angular real obtenida tras la simulación se muestra en la Figura 5.2. A partir de ahora se hará uso de grados en lugar de radianes por simplicidad. En los primeros momentos la velocidad angular comienza con un comportamiento oscilatorio, característico previo a los efectos del controlador. El primer modo de operación en ser activado es el B-dot, cuyo efecto deseado es la reducción de la velocidad angular. La transición hacia el apuntamiento se lleva a cabo una vez que la norma de la velocidad angular se encuentra por debajo del umbral definido, en este caso $5^\circ/\text{s}$. Esto ocurre después de las primeras tres horas.

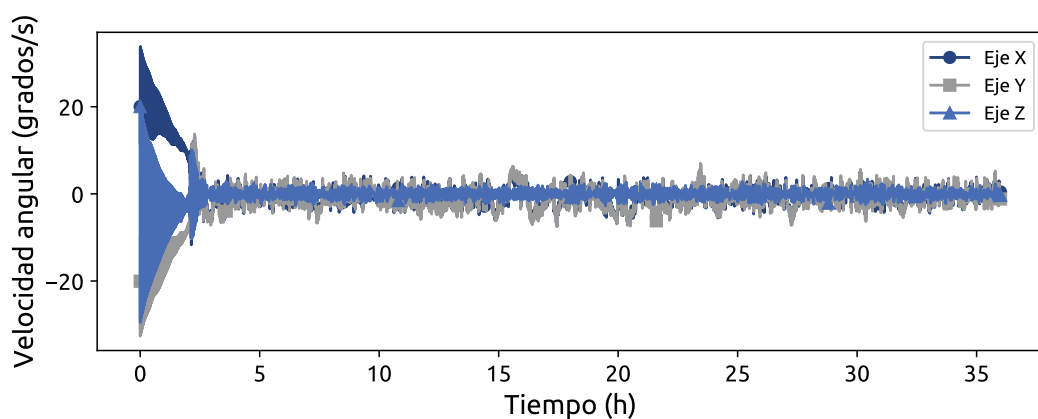


Figura 5.2: Velocidad angular real del satélite en la simulación con respecto al tiempo.

5. RESULTADOS

La señal alcanza el valor ideal de $0^\circ/\text{s}$, pero no lo mantiene por largos periodos de tiempo. Esto entra dentro del comportamiento esperado, el cual se debe a la generación de par constante a través de las magnetorcas. Las perturbaciones del ambiente espacial siempre estarán actuando sobre el satélite, por lo que las magnetorcas estarán aplicando un par de control de forma continua para contrarrestar los efectos de estas perturbaciones. El algoritmo de apuntamiento logra exitosamente lo anterior al mantener la velocidad angular cercana a cero.

En la Figura 5.3 se presenta una gráfica donde se aprecia el comportamiento del par de las magnetorcas contra el tiempo. El caso del momento magnético y la potencia se muestran en las Figuras 5.4 y 5.5 respectivamente.

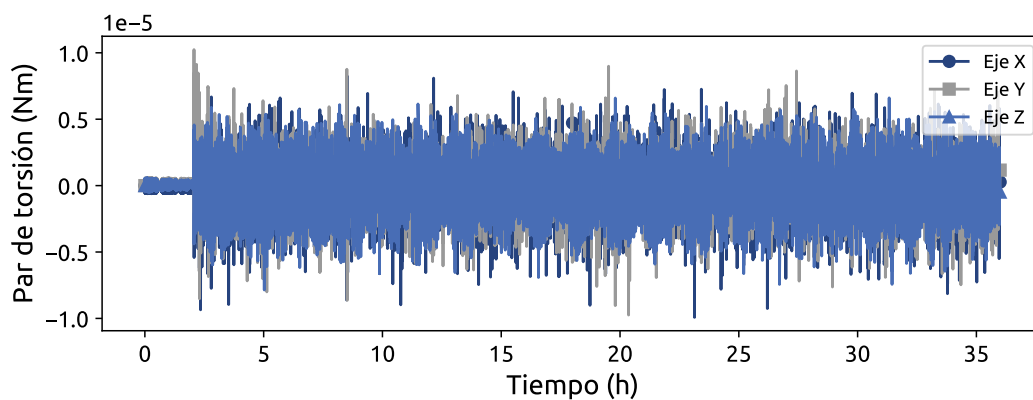


Figura 5.3: Par de control generado por las magnetorcas del satélite en la simulación con respecto al tiempo.

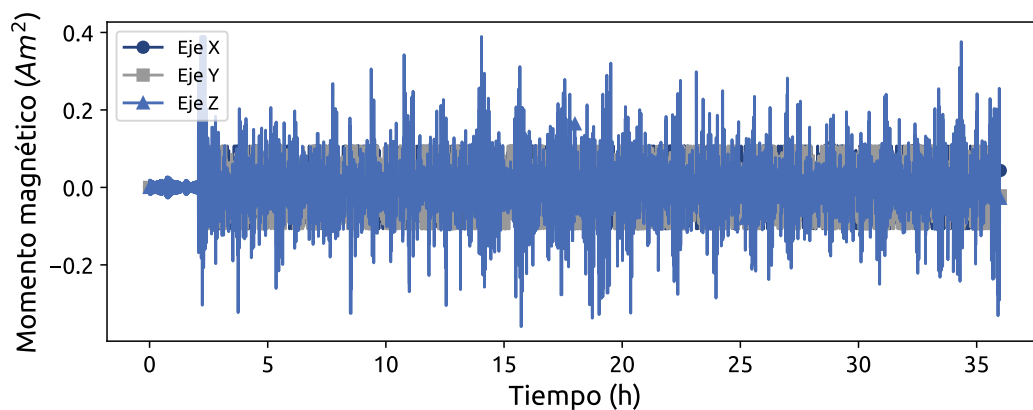


Figura 5.4: Momento magnético generado por las magnetorcas del satélite en la simulación con respecto al tiempo.

El par de control, el momento magnético y la potencia siguen un patrón similar, en donde la respuesta crece en magnitud al transcurrir las tres primeras horas, igual que en el caso anterior. Este comportamiento coincide con el inicio del modo

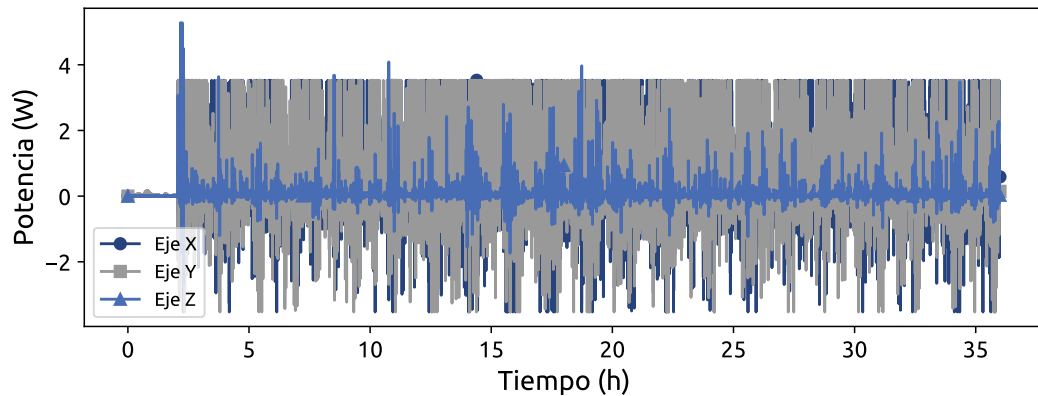


Figura 5.5: Potencia de las magnetorcas del satélite simulado con respecto al tiempo.

de apuntamiento, el cual no se encuentra activo de forma constante, sino que lo hace en periodos que corresponden a la situación en donde el satélite se encuentra sobre el territorio de interés. Debido a la deriva en la velocidad angular durante los periodos de inactividad, el par, el momento magnético y la potencia subsiguientes generadas por las magnetorcas deberán ser mayores durante los cortos periodos de activación. En contraste, el B-dot es activado de forma continua, además de contar en la implementación con un mecanismo para limitar la potencia bajo un umbral definido.

La orientación del satélite se da como resultado de su velocidad angular en un instante específico, la cual estará definida por los efectos del controlador. En la Figura 5.6 se muestra la orientación en cuaterniones con respecto al tiempo en el marco de referencia orbital. El comportamiento esperado consistiría en grandes cambios en la orientación instantes después de la liberación del satélite desde el dispensador (*tumbling*). Después, una vez iniciado el algoritmo B-dot, la orientación sufriría cambios más lentos gracias a la reducción en la velocidad angular (la orientación no tendería a un valor específico). Finalmente cuando el apuntamiento está activo se esperaría que la orientación tienda al cuaternión $[1 \ 0 \ 0 \ 0]$ con respecto al marco orbital en la convención adoptada, indicando que el satélite está apuntando hacia nadir. Sin embargo, este comportamiento no ocurre y no se observa ninguna tendencia en la orientación. Esto indica que la orientación no fue controlada, posiblemente atribuido a problemas de estabilidad o de una elección errónea de los parámetros de control en el apuntamiento. El error con respecto a la orientación de referencia a nadir se muestra en la Figura 5.7.

Por otra parte, en la Figura 5.8 se muestra una captura obtenida de la animación del subsistema de visualización, lo cual representa una forma más interactiva de observar los resultados.

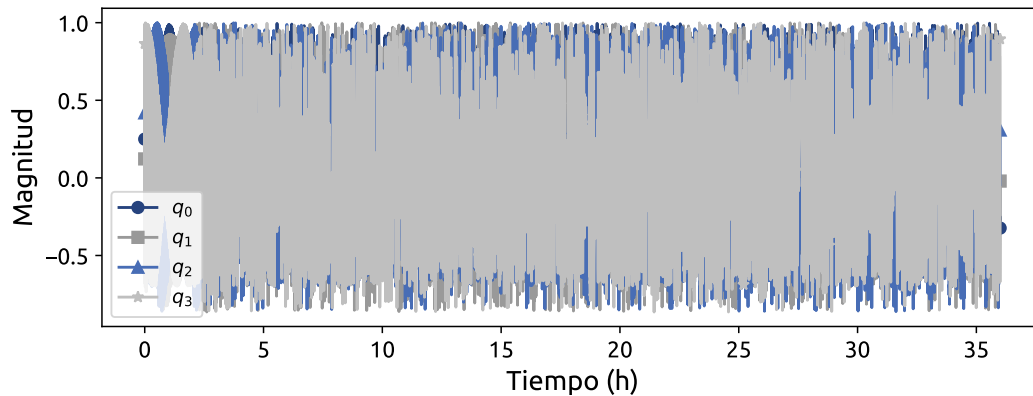


Figura 5.6: Orientación en cuaterniones del satélite simulado con respecto al tiempo en el marco de referencia orbital.

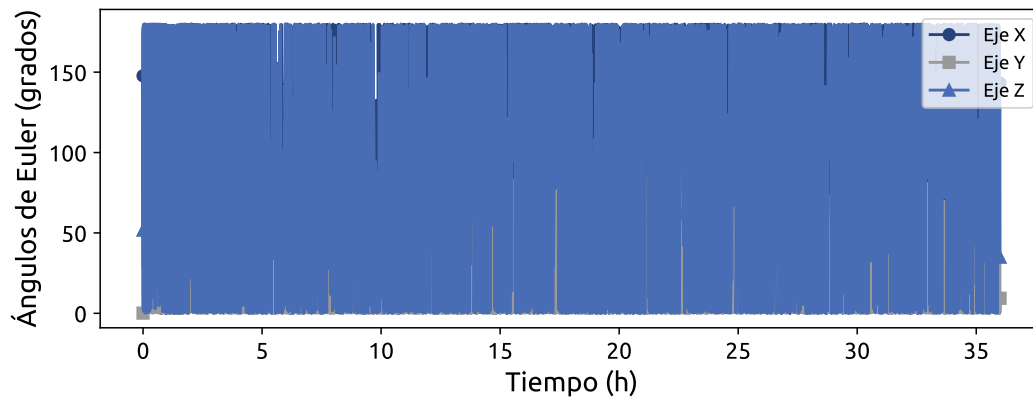


Figura 5.7: Error de la orientación resultante con respecto a la orientación objetivo (marco de referencia orbital). Se utiliza la representación de ángulos de Euler.

5.3.4. Determinación de orientación

Dentro del subsistema de determinación de orientación del modelo virtual, el algoritmo Davenport Q no pudo ser implementado directamente debido a las limitaciones que existen en Simulink con respecto a las funciones para tratar eigenvalores. Sin embargo, los algoritmos TRIAD y QUEST se implementaron con éxito. Debido a las especificaciones de K'oto se eligió al algoritmo TRIAD como el algoritmo de determinación estático a utilizar durante la misión. Una forma de evaluar los algoritmos de determinación estáticos es mediante la ecuación de Wahba. En la Figura 5.9 se grafican sus valores con respecto al tiempo.

En la ecuación de Wahba los resultados son adimensionales, donde un valor que tiende a cero representa que la orientación obtenida incorpora adecuadamente la información de los vectores proporcionados, provenientes de marcos de referencia distintos. La ecuación de Wahba representa una métrica para evaluar el algoritmo

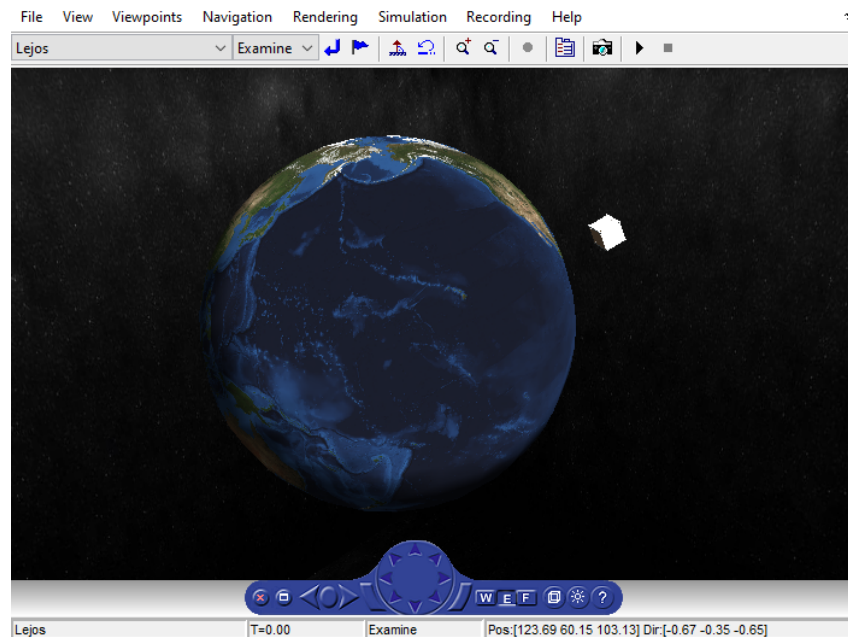


Figura 5.8: Captura proveniente del subsistema de visualización.

seleccionado, sin embargo para cuantificar el error con respecto a la orientación real se deben emplear otros métodos. Un ejemplo de ello se discute en la Sección 5.3.5.

La fluctuación en la Figura 5.9 era de esperarse considerando las variaciones que pueden existir en los vectores de referencia, a los cuales se les ha implementado un ruido blanco, obtenido por medio de los modelos de los sensores. Sin embargo, el orden de magnitud permanece constante y para esta métrica se considera elevado, por lo que se considera que los resultados de la determinación de orientación son imprecisos en contraste con resultados que tiendan a cero. Esto se debe a una combinación de los factores de ruido de los sensores y las desventajas del mismo algoritmo. Este resultado sustenta la necesidad de un estimador de estado en el sistema.

5.3.5. Filtro de Kalman extendido

El filtro de Kalman extendido se inicializó con los parámetros de la Tabla 5.7.

Tabla 5.7: Parámetros de inicialización del filtro de Kalman Extendido.

CARACTERÍSTICA	VALOR
Estimación inicial de la covarianza	$0.001\mathbf{I}_{7 \times 7}$
Q	$200\mathbf{I}_{7 \times 7}$
R	$200\mathbf{I}_{7 \times 7}$

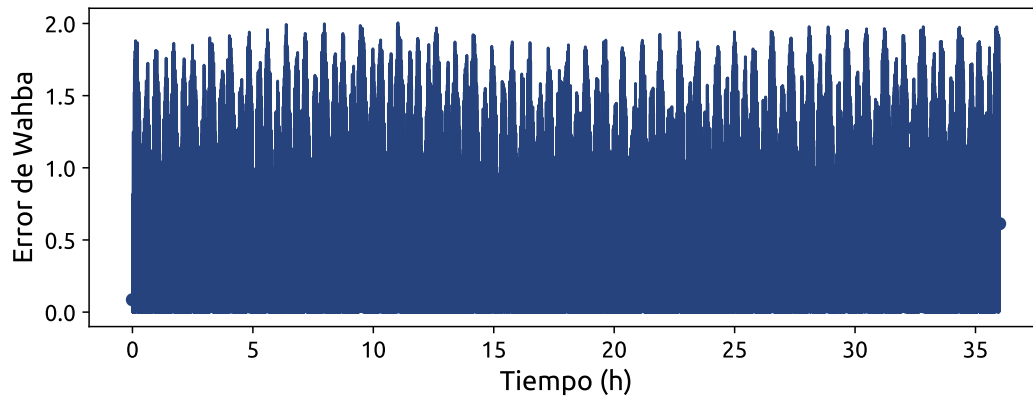


Figura 5.9: Error de la ecuación de Wahba para el algoritmo TRIAD en el satélite simulado.

La inicialización del filtro juega un papel importante en su desempeño futuro. La elección de valores definirá en gran medida la velocidad de convergencia, pero en el caso particular del filtro de Kalman extendido también determinará si este divergirá eventualmente. Esto se debe a la linealización del filtro y el hecho de que no es un estimador óptimo. Debido a que los valores de la velocidad angular y la orientación serán definidos por las condiciones impuestas por el dispensador, las cuales son desconocidas al momento de diseñar el filtro, las estimaciones iniciales se definieron como las primeras lecturas que devuelven los sensores y el algoritmo de determinación respectivamente. Es decir, las velocidades angulares devueltas por el modelo de giroscopio y la orientación determinada por TRIAD. Esto garantiza que los valores de inicialización no difieran en gran magnitud del estado verdadero.

De igual forma, las matrices \mathbf{Q} y \mathbf{R} se definen durante la fase de diseño y determinarán el comportamiento del filtro. La obtención de estas matrices se dio a través de varias iteraciones de la simulación para conseguir aquellas que dieran la mejor respuesta. Durante este proceso se encontró que se deben favorecer los datos provenientes de los modelos de transición en ambos casos debido al ruido elegido para los sensores. Este es el caso especialmente en el caso de la orientación medida.

El error de las mediciones contra el valor real y de las estimaciones contra el valor real se muestra a continuación para ambas variables de estado. Las Figuras 5.10 y 5.11 corresponden a la velocidad angular, mientras que las Figuras 5.12 y 5.13 corresponden a la orientación. Para evaluar el filtro en términos de la orientación se eligió la representación de ángulos de Euler debido a que facilitan la interpretación. Además, se usa el marco de referencia inercial para la comparación, puesto que la determinación de orientación se hace por medio de este marco.

Una comparación de la señal filtrada contra la señal medida, se muestra en la Tabla 5.8 para la orientación y la velocidad angular. La métrica elegida fue la raíz del error cuadrático medio (RMSE), en donde se hace una evaluación de cada señal estimada contra las señales reales. Resultados cercanos a cero implican una cercanía con estos valores reales y por lo tanto un mejor rendimiento del estimador.

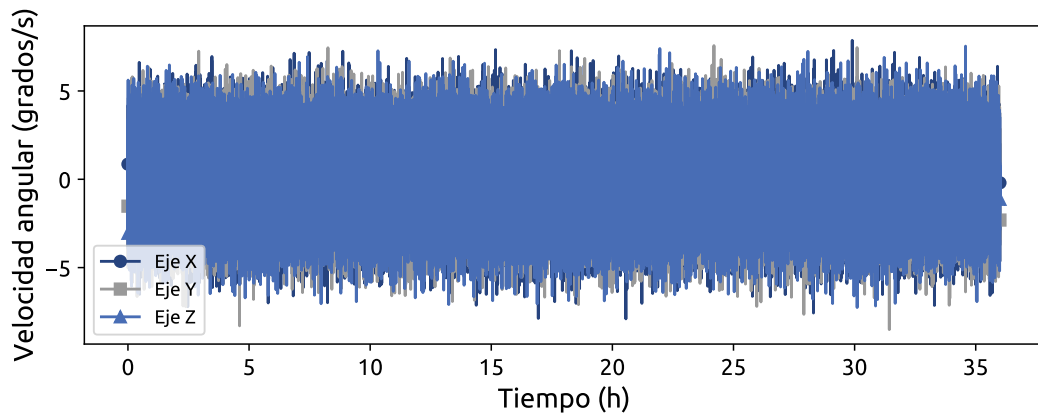


Figura 5.10: Error de la velocidad angular medida con respecto a la velocidad angular real.

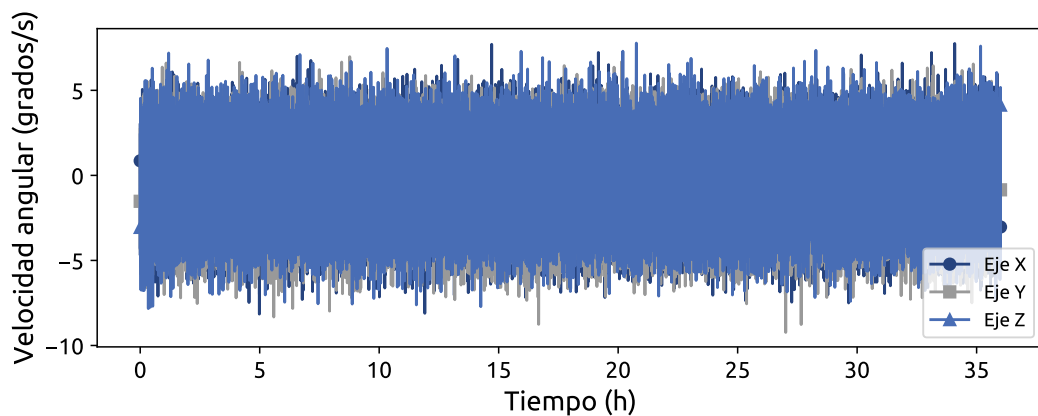


Figura 5.11: Error de la velocidad angular filtrada con respecto a la velocidad angular real.

Los valores filtrados resultaron en una reducción del RMSE con respecto a los valores medidos de la orientación, pero un ligero aumento con respecto a la velocidad angular, lo cual indica un funcionamiento parcial del filtro. En cuanto a la orientación se observa una mejora significativa, implicando que el filtro de Kalman es una buena herramienta para contrarrestar los resultados imprecisos del algoritmo TRIAD conforme a lo mostrado en la Figura 5.9. En el caso del error de la orientación filtrada (Figura 5.13), se puede observar que el error se incrementa drásticamente en ciertos puntos, sin embargo este es un comportamiento esperado puesto que se tratan de orientaciones equivalentes de acuerdo a la representación de ángulos de Euler. Es posible introducir una función de ajuste para quitar estas ambigüedades y así reducir más el RMSE.

Para la velocidad angular la diferencia entre los RMSE es ligera, por lo que se le puede atribuir en primer lugar a una inadecuada elección de los parámetros del filtro.

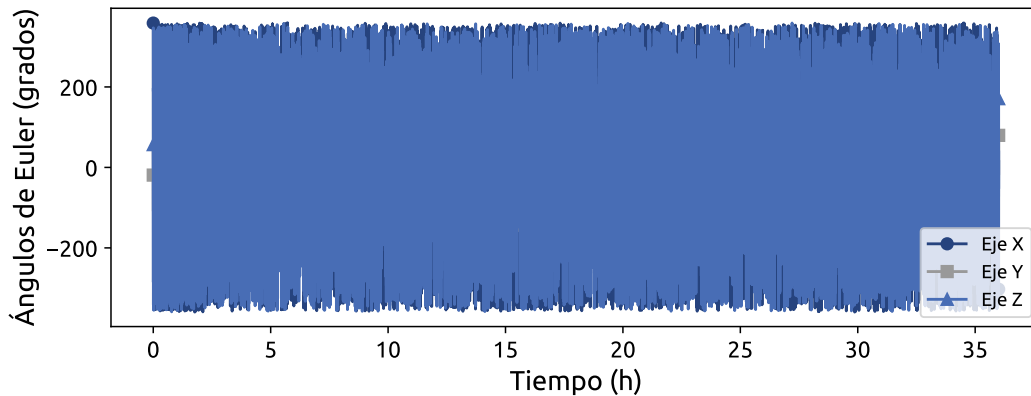


Figura 5.12: Error de la orientación medida con respecto a la orientación real. La orientación se muestra en ángulos de Euler.

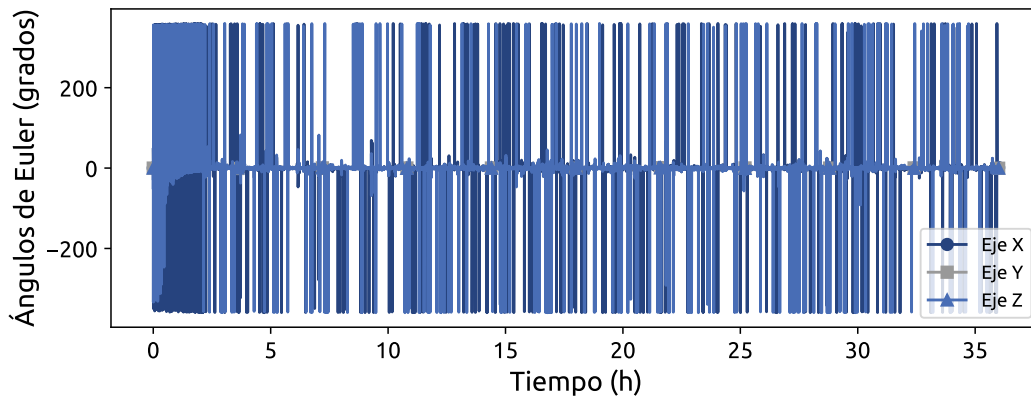


Figura 5.13: Error de la orientación filtrada con respecto a la orientación real. La orientación se muestra en ángulos de Euler.

Un ajuste de este tipo debe realizarse antes de modificar los modelos de transición de estado o los modelos de medición dentro del filtro.

5.3.6. Verificación de requerimientos

Como parte de la realización del modelo virtual se incluyó una funcionalidad para la verificación de los requerimientos, en donde se hizo uso de *Simulink Test*. Mediante este *toolbox* es posible crear pruebas con respecto a la simulación al establecer aserciones que deben satisfacerse a lo largo de la ejecución. Estas aserciones pueden ser de tipo lógico, las cuales indican los límites permitidos de una señal dada, o de tipo temporal donde se puede evaluar la ocurrencia de eventos después de una señal de disparo. Tras la ejecución de las pruebas se obtiene un resultado binario, el cual depende de la satisfacción de todas las aserciones involucradas. En otras palabras *Simulink Test* es una forma de realizar pruebas unitarias y de integración entre subsistemas de la simulación. Mediante este método se quitan las ambigüedades

Tabla 5.8: Evaluación del filtro de Kalman extendido usando la raíz del error cuadrático medio.

	VELOCIDAD ANGULAR ($^{\circ}/s$)	ORIENTACIÓN ($^{\circ}$)
Valores medidos	[1.813 1.811 1.813]	[153.933 56.452 146.346]
Filtro de Kalman	[1.859 1.902 1.839]	[20.091 0.831 18.691]

existentes en la etapa de verificación, las cuales si no se tiene una metodología definida, pueden estar sujetas a la subjetividad.

Adicionalmente, Simulink cuenta con una función para insertar los requerimientos directamente en el modelo, lo cual puede realizarse ya sea directamente desde la interfaz de MATLAB o mediante la importación de un documento preexistente. La inclusión de requerimientos dentro del modelo permite su trazabilidad al asociar cada uno de ellos a la respectiva parte del modelo, función o subsistema que lo implementan. Pero una de las principales ventajas de incluir los requerimientos mediante este enfoque es la capacidad de enlazarlos directamente con las pruebas de Simulink Test, con lo cual el resultado de la prueba determinará si el requerimiento se satisface o no.

La unión de ambos elementos, los requerimientos y las pruebas, permite focalizar la etapa de verificación del satélite bajo una misma interfaz y facilita el encontrar aquellos parámetros o aspectos del diseño que deben ser modificados para poder cumplir con todos los requerimientos.

5.3.6.1 REQUERIMIENTOS EN EL CASO DE ESTUDIO K'OTO

Las funciones de verificación y de optimización de parámetros (descrita en la sección posterior) requieren de ejecutar simulaciones en serie, por lo que para demostrar estas funcionalidades se utilizó una configuración distinta a la simulación principal. El solucionador se definió como ode45, el cual es un método de Runge-Kutta de paso variable, mientras que la duración se definió como 1 hora. La opción de usar la misma configuración a la simulación principal (paso fijo) fue descartada debido al excesivo tiempo de cómputo requerido.

En la Tabla 5.6 se muestran los requerimientos definidos como parte de la misión y el resultado obtenido tras las pruebas de verificación usando el método anterior. En K'oto varios de los requerimientos están enfocados a evaluar la salida de las magnetorcas.

A manera de ejemplo para demostrar la funcionalidad de verificación, uno de los requerimientos es que la magnitud de la velocidad angular durante toda la operación se encuentre por debajo de $30^{\circ}/s$. El resultado de una de las pruebas unitarias relacionadas con este requerimiento se presenta en la Figura 5.14. En este caso en particular la prueba devolvió un resultado negativo, por lo que el requerimiento no se cumplió. La razón de ello se debe a las condiciones iniciales, lo cual indica que se debe hacer una reformulación de los requerimientos para considerar estos casos o añadir condiciones más específicas que los excuyan.

5. RESULTADOS

Tabla 5.9: Resultados de la verificación de requerimientos en el caso de estudio K'oto.

REQUERIMIENTO	RESULTADO
La velocidad angular neta se mantendrá por debajo de 30 °/s	Fallido
El apuntamiento a nadir se logrará en menos de 5 minutos una vez iniciado el modo.	Fallido
El error de la orientación durante el modo de apuntamiento a nadir será menor a 20°	Fallido
La velocidad angular neta durante el apuntamiento a nadir se mantendrá por debajo de 4°	Aprobado

Dentro de la interfaz es posible ver el estado de la verificación con respecto al resultado de las pruebas (Figura 5.15).

The screenshot shows a software interface with a 'SUMMARY' section. It contains a table with the following data:

Name	Umbral de velocidad angular de operación
Outcome	1 ✖
Start Time	09/15/2021 22:27:52
End Time	09/15/2021 22:29:24
Type	Baseline Test
Test File Location	E:\Proyectos\Koto\adcs-koto-simulink\Pruebas\Tes...
Test Case Definition	📄
Rerun Test Case	▶
Tags	
Cause of Failure	Failed criteria: Assessments
▶ Simulation Metadata	

Below the summary is a 'TEST REQUIREMENTS' section with a single entry:

4: [La velocidad angular neta se mantendrá <30 °s \(Requerimientos#4\)](#)

Figura 5.14: Resultado de la ejecución de una prueba unitaria en la simulación. El resultado es negativo, por lo que el requerimiento no se cumplió.

La mayoría de los requerimientos no fueron aprobados, en especial aquellos relacionados con el apuntamiento. Una conclusión que se puede hacer a partir de estos resultados es la necesidad de ajustar los parámetros de control para mejorar el desempeño. Además se recomienda definir un protocolo de verificación para definir las pruebas y criterios de la verificación con el fin de evitar falsos positivos. Este pudo ser el caso del cuarto requerimiento, el cual aparentemente fue aprobado en esta demostración, pero falta de comprobar al ejecutar la simulación con paso fijo y duración extendida.

The screenshot shows a software interface for requirements management. At the top, there is a 'View: Requirements' dropdown and a search box. Below is a table with columns: Index, ID, Summary, Implemented, and Verified. The table lists four requirements: 1 (Estabilización), 2 (Apuntamiento), 3 (Operación), and 4 (#20 Justifications). Requirement 2 is highlighted, and a tooltip shows its verification statistics: Passed: 1, Justified: 4, Failed: 3, Unexecuted: 0, None: 0, Total: 8. Progress bars are shown for the 'Implemented' and 'Verified' columns for each requirement.

Index	ID	Summary	Implemented	Verified
1	A	Estabilización	<div style="width: 100%; background-color: blue;"></div>	<div style="width: 100%; background-color: green;"></div>
2	B	Apuntamiento	<div style="width: 0%; background-color: blue;"></div>	<div style="width: 100%; background-color: green;"></div>
3	C	Operación	<div style="width: 100%; background-color: blue;"></div>	<div style="width: 100%; background-color: green;"></div>
4	#20	Justifications		

Figura 5.15: Resultado de la verificación de requerimientos de acuerdo a las pruebas unitarias en la simulación. Además de incluir las pruebas aprobadas y falladas, también se incluye la opción de especificar los requerimientos sin pruebas.

5.3.7. Optimización de parámetros de control

La parametrización de características como las del controlador hizo posible el desarrollo de otra funcionalidad, la optimización de parámetros. Gracias al modelo virtual es posible realizar una optimización de parámetros como la ganancia de control para encontrar valores ideales dadas ciertas restricciones impuestas. Una optimización de este tipo permite refinar el controlador antes de la fase de pruebas con componentes reales, además de que se tiene una metodología más formalizada a comparación de ensayos de prueba y error.

Esta funcionalidad es ofrecida a partir de la aplicación de Simulink *Sensitivity Analysis*, con la cual se pueden ejecutar diferentes iteraciones de la simulación utilizando diferentes valores de los parámetros. Es posible agregar restricciones a estos valores en cuanto al valor mínimo y máximo, además de que es posible definirlos en términos de distribuciones de probabilidad como gaussianas.

Sensitivity Analysis calcula después las combinaciones para los diferentes parámetros, lo cual corresponde a la configuración de cada iteración. La herramienta procede a ejecutar la simulación con las diferentes combinaciones y se guarda el resultado de acuerdo a la métrica elegida y su requerimiento básico, por ejemplo, la velocidad angular y su minimización hacia cero. Al final de la ejecución se producen gráficas como la de la Figura 5.16, en donde se muestra como los diferentes valores de los parámetros afectan la salida del modelo de acuerdo a la métrica elegida.

El diseñador puede elegir entonces los valores apropiados considerando los *trade-offs*. Esta funcionalidad ofrece un respaldo con evidencia estadística para la toma de esta decisión, puesto que además es posible generar reportes estadísticos a partir de los resultados.

Sin embargo como se mencionó anteriormente, la combinación de ejecutar una serie de simulaciones con haber especificado un paso fijo resultan en un tiempo total de simulación largo, el cual está en el orden de magnitud de semanas. Por lo tanto el uso de esta funcionalidad también debe considerar las restricciones de tiempo del proyecto.

5. RESULTADOS

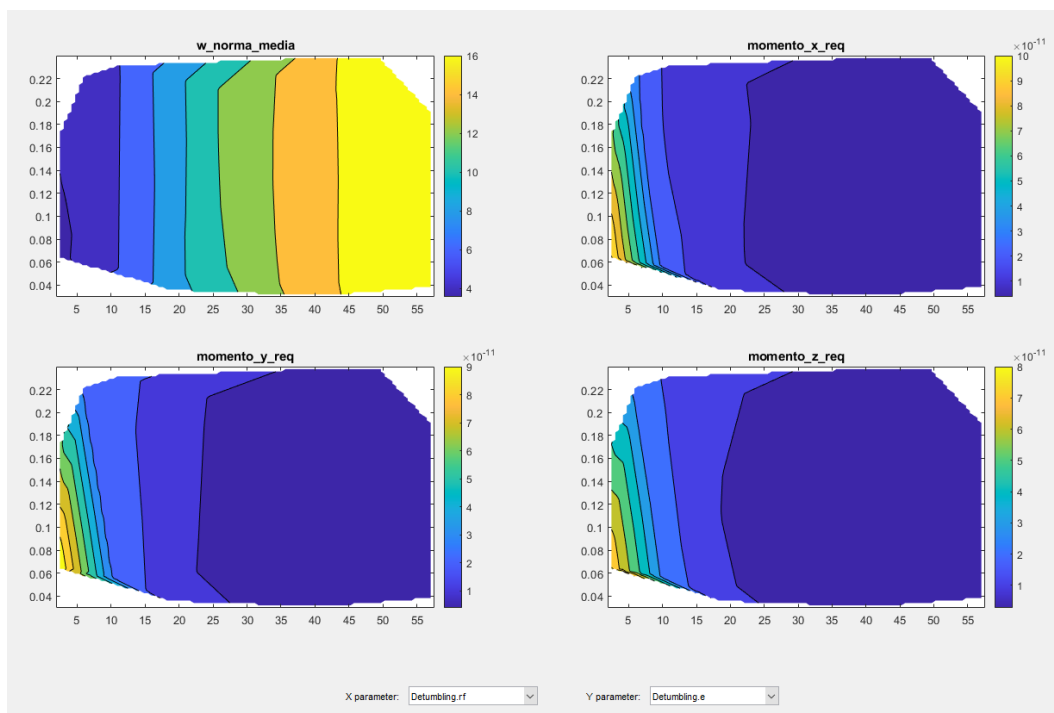


Figura 5.16: Resultado de optimización de las constantes de control para el algoritmo B-dot. Se muestra el efecto de variar los parámetros entre sí con respecto a las métricas seleccionadas. En este caso se eligió la norma de la velocidad angular y los valores máximos del momento magnético en cada eje generados por las magnetorcas.

6

Conclusiones

EN ESTE TRABAJO se desarrolló una herramienta para simular la cinemática y dinámica de un satélite con el objetivo de validar y verificar un subsistema de determinación y control de orientación.

Para esto se incluyeron los elementos del ambiente espacial que producen perturbaciones en el satélite, lo cual corresponde al factor de interferencia no deseada al que se debe enfrentar el controlador. Además, también se incluyeron modelos que representan los algoritmos embebidos dentro del satélite y que con los cuales este podrá obtener datos del exterior para poder determinar las acciones de control apropiadas. Estos algoritmos son los de determinación de orientación, estimación de estado y control de orientación, donde los primeros dos corresponden a los desarrollados durante esta tesis. Un modelo de visualización también fue incluido, con el cual se muestran los resultados mediante una animación en 3D realizada desde el mismo entorno de Simulink. En conjunto, estos subsistemas representaron el modelo virtual del ADCS de un nanosatélite.

La herramienta se desarrolló con un nanosatélite con control de orientación magnético en mente y para evaluar su funcionamiento se utilizó el caso de K'oto, un nanosatélite que comparte estas características. Los resultados obtenidos de las simulaciones indicaron que los algoritmos de determinación son capaces de calcular la orientación a partir de los vectores de referencia devueltos por los modelos de sensores, considerando el ruido que estos últimos agregan. Sin embargo se descubrió que el uso de un estimador de estado, el cual se consideró originalmente como una medida de respaldo, se convirtió en una necesidad debido a los resultados imprecisos.

Un filtro de Kalman extendido fue propuesto para incorporar la información obtenida a través de sensores así como la obtenida a partir de modelos matemáticos con el objetivo de mejorar la determinación de orientación.

El diseño del filtro se basó en la configuración del nanosatélite simulado. Se encontró que el rendimiento del filtro depende en gran medida de los parámetros ajustables \mathbf{Q} y \mathbf{R} , por lo que el ajuste deberá hacerse con respecto a la implementación en el microcontrolador. Esto se debe a que los valores definidos para la

simulación pueden traer resultados distintos cuando el filtro se ejecuta a través del microcontrolador. En particular este es el caso para la velocidad angular.

En cuanto a la parte de control de orientación, los algoritmos proporcionados reducen exitosamente la velocidad angular, la cual tiende a cero a pesar de las perturbaciones presentes. Sin embargo, el apuntamiento no proporcionó los resultados esperados para la orientación, ya que se observó que esta no llega a un estado estacionario ni a los valores necesarios para apuntar a nadir. Esto pudo deberse a la elección de los parámetros del controlador o posibles problemas de estabilidad en algoritmo, lo cual requiere de una investigación más profunda antes de la implementación en el microcontrolador. Fue posible identificar este problema y la falta de cumplimiento en los requerimientos prematuramente gracias al modelo virtual desarrollado; y además se ofrecen métodos para apoyar la decisión que será tomada al respecto.

En el caso de que se desee probar otros parámetros de control será posible usar la funcionalidad de optimización para poder guiar la decisión, mientras que si se desea probar otro algoritmo de control se tiene la opción de implementarlo como un modelo variante para comparar su desempeño. Mediante la herramienta de verificación es posible determinar si estos cambios satisfarán los requerimientos o si más cambios serán necesarios.

La metodología que se adoptó para el desarrollo del proyecto, el diseño basado en modelos (MBD), proporcionó varias ventajas como la naturaleza modular de los modelos mediante la cual se permite incorporar nuevas funcionalidades sin romper el funcionamiento del resto del sistema. Adicionalmente se tiene la capacidad de incorporar modelos variantes, con lo cual se pueden intentar enfoques diferentes de un mismo subsistema en cuanto al modelado mismo, los valores de sus parámetros o su fuente de origen. Esto no solo es de utilidad en la fase de prototipado, sino que también aumenta la reusabilidad en otros proyectos. Otra de las ventajas de usar el MBD se encontró en términos de la implementación, ya que es posible generar código para el microcontrolador a partir de los modelos con el fin de aplicar técnicas HIL. De esta forma tanto las simulaciones como la implementación existen en un ecosistema donde existe retroalimentación entre ambos en lugar de que el desarrollo sea aislado.

Para un nanosatélite como K'oto esta herramienta representaría una forma de poder dar seguimiento a las fases de desarrollo e integración. Es posible proceder con el resto de técnicas in-the-Loop, ya que los principales componentes que definirán el comportamiento de K'oto ya han sido modelados. El resultado de usar estos métodos se verá reflejado en el software de vuelo relacionado con el control de orientación, donde las aportaciones del MBD consisten en la reducción de riesgos, aumento de trazabilidad y agilización de integración.

Por otra parte, la misma elaboración de este trabajo busco alinearse con lo establecido en el Plan de órbita 2.0. Con respecto a la formación de recursos humanos, se adquirieron varias de las competencias involucradas en el trato de un proyecto aeroespacial relacionadas con el desarrollo, administración y documentación.

En cuanto al desarrollo tecnológico nacional se buscó elaborar una herramienta de análisis de misión que no dependiera del uso de programas externos que puedan

frenar el progreso de la misión por cuestiones administrativas. El uso de STK, el cual depende de licencias, se limitó únicamente a proveer los datos de aquellos modelos que se encontraron fuera del alcance de la tesis. En el futuro se espera que estos modelos sean desarrollados e implementados dentro del modelo virtual y así reducir la dependencia hacia herramientas externas por completo. Esto resultará en un impulso adicional para la tecnología aeroespacial desarrollada en México.

6.1. Trabajo a futuro

Con base en el trabajo realizado y en las lecciones aprendidas a lo largo de este se presentan las siguientes recomendaciones para las subsecuentes iteraciones de la herramienta desarrollada:

- Para el caso de estudio de K'oto se recomienda hacer uso de las herramientas de optimización y verificación de forma iterativa con el fin de mejorar los algoritmos de control de orientación y así añadir robustez al diseño. Es recomendable realizar este proceso antes de considerar cambiar los algoritmos.
- En cuanto a la implementación, el siguiente paso es realizar los análisis faltantes de Software-in-the-Loop, para la generación de código automática a partir de los modelos. La verificación para la implementación se deberá hacer mediante Hardware-in-the-Loop para involucrar al hardware en las evaluaciones.
- Para el modelo orbital y atmosférico actualmente se emplearon datos extraídos desde STK. Este fue el caso particularmente para los modelos de densidad atmosférica. Siguiendo el objetivo de independencia tecnológica es deseable implementar estos modelos dentro de la simulación como subsistemas propios para no depender por completo de STK, aunque ambas opciones pueden coexistir en el modelo gracias a los bloques variantes.
- Se asumieron los valores para algunos parámetros como el coeficiente de arrastre y los coeficientes de absorción y reflexión basados en valores de la literatura. Se recomienda caracterizar estos valores basados en los componentes reales del satélite para otorgar una mayor precisión al modelo.
- Para la estimación se propuso el uso de un Filtro de Kalman Extendido. Existen variaciones en cuanto a este filtro como el EKF aditivo o multiplicativo que surgieron como una mejora para tratar con cuaterniones, por lo que se recomienda indagar al respecto. Alternativamente se puede implementar otro tipo de filtro como el Filtro de Kalman Unscented para comparar su desempeño con respecto al EKF.

Referencias

- [1] Anil Kumar Maini y Varsha Agrawal. *Satellite Technology: Principles and Applications*. 2nd ed. Chichester, West Sussex, U.K. ; Hoboken, N.J: Wiley, 2011. 674 págs. ISBN: 978-0-470-66024-9.
- [2] Giuseppe Di Mauro, M. Lawn y Riccardo Bevilacqua. «Survey on Guidance Navigation and Control Requirements for Spacecraft Formation-Flying Missions». En: *Journal of Guidance, Control, and Dynamics* 41 (5 de dic. de 2017), págs. 1-22. DOI: [10.2514/1.G002868](https://doi.org/10.2514/1.G002868).
- [3] United Nations. *Space4SDGs: How Space Can Be Used in Support of the 2030 Agenda for Sustainable Development*. 2021. URL: <https://www.unoosa.org/oosa/en/ourwork/space4sdgs/index.html> (visitado 26-05-2021).
- [4] Ashley Ladin. *Student-Built CubeSat DAVE Successfully Launches Aboard Delta II*. Mustang News. 20 de sep. de 2018. URL: <https://mustangnews.net/cubesat-dave-successfully-launches-aboard-delta-ii/> (visitado 02-09-2021).
- [5] Erik Kulu. *ITASAT 1 @ Nanosats Database*. Nanosats Database. 2020. URL: <https://www.nanosats.eu/sat/itasat-1.html> (visitado 02-09-2021).
- [6] Erik Kulu. *AISTECHSAT (DANU) 2U @ Nanosats Database*. Nanosats Database. 2021. URL: <https://www.nanosats.eu/sat/aistechsat.html> (visitado 02-09-2021).
- [7] Erik Kulu. *MicroMAS-2 @ Nanosats Database*. Nanosats Database. 2020. URL: <https://www.nanosats.eu/sat/micromas-2.html> (visitado 02-09-2021).
- [8] California Polytechnic State University. *CubeSat Design Specification*. 2020.
- [9] NASA. «CubeSat 101: Basic Concepts and Processes for First-Time CubeSat Developers». En: (2017), pág. 96.
- [10] Bruno Victorino Sarli y col. «Review of Space Activities in South America». En: (2018), pág. 25.
- [11] Johanna Mendelson Forman y col. *Toward the Heavens: Latin America's Emerging Space Programs*. CENTER FOR STRATEGIC & INTERNATIONAL STUDIES, 2009. URL: https://csis-website-prod.s3.amazonaws.com/s3fs-public/legacy_files/files/publication/090730_Mendelson_TowardHeavens_Web.pdf (visitado 07-07-2021).
- [12] Alejandro Montaña Barbosa. *La trayectoria de México en la exploración espacial*. México Ciencia y Tecnología. 16 de dic. de 2015. URL: <http://www.cienciamx.com/index.php/ciencia/universo/4714-historia-de-la-astronautica-en-mexico-del-sputnik-i-a-la-agencia-espacial-mexicana> (visitado 13-11-2020).
- [13] *La trayectoria de México en la exploración espacial*. México Ciencia y Tecnología. URL: <http://www.cienciamx.com/index.php/ciencia/universo/4714-historia-de-la-astronautica-en-mexico-del-sputnik-i-a-la-agencia-espacial-mexicana> (visitado 07-07-2021).
- [14] Viridiana Fernández Vera. *Gobierno estatal trabaja en proyecto de la estación terrena de Tulancingo*. Síntesis Hidalgo. 12 de oct. de 2018. URL: <https://sintesis.com.mx/hidalgo/2018/10/12/gobierno-proyecto-estacion-terrena-tulancingo/> (visitado 02-09-2021).

REFERENCIAS

- [15] Rodrigo Nava Amezcua. «Historia de La Industria Aeroespacial En México y Su Vínculo Con La Aeronáutica». En: *Ciencia UANL* 81 (Septiembre-octubre 2016 2016).
- [16] ProMéxico y col. *Plan de Órbita 2.0: Mapa de Ruta Del Sector Espacial Mexicano*. 2017.
- [17] Miguel Ángel Vázquez Ruiz y Carmen Bocanegra. «La Industria Aeroespacial En México: Características y Retos En Sonora». En: *Problemas del Desarrollo. Revista Latinoamericana de Economía* 49.195 (26 de sep. de 2018). ISSN: 2007-8951, 0301-7036. DOI: 10.22201/iiiec.20078951e.2018.195.63183. URL: <https://www.probdes.iiiec.unam.mx/index.php/pde/article/view/63183> (visitado 09-11-2020).
- [18] Gaceta UNAM. *El satélite universitario UAMSAT-B fue puesto en órbita*. 1996. URL: <https://www.dint.unam.mx/blog/index.php/item/3201-el-satelite-universitario-uamsat-b-fue-puesto-exitosamente-en-orbita> (visitado 04-06-2021).
- [19] Agencia Espacial Mexicana. *Cumple con éxito su misión el primer nanosatélite mexicano Aztechsat-1*. gob.mx. URL: <http://www.gob.mx/aem/articulos/cumple-con-exito-su-mision-el-primer-nanosatelite-mexicano-aztechsat-1-253907?idiom=es> (visitado 10-11-2020).
- [20] UPAEP. *AzTechSat-1*. 2019. URL: <https://upaep.mx/aztechsat> (visitado 17-11-2020).
- [21] Javier López Casarín. *Satélite SAI-I 'NanoConnect-2', Misión Colaborativa*. El Economista. 2021. URL: <https://www.economista.com.mx/opinion/Satelite-SAI-I-NanoConnect-2-Mision-Colaborativa-20210304-0037.html> (visitado 17-06-2021).
- [22] Editorial Demos y La Jornada. *La Jornada - Lanzarán al espacio Nanosatélite desarrollado por la UNAM*. 2021. URL: <https://www.jornada.com.mx> (visitado 17-06-2021).
- [23] Fernanda Avila | El Sol de México. *¡México Llega al Espacio! Lanzan Con Éxito Nanosatélite Desarrollado Por La UNAM*. El Sol de México | Noticias, Deportes, Gossip, Columnas. 2021. URL: <https://www.elsoldemexico.com.mx/doble-via/ciencia/agencia-india-lanzamiento-nanoconnect-2-nanosatelite-mexicano-unam-espacio-orbita-6419522.html> (visitado 17-06-2021).
- [24] Jorge Flores Villa. *Painani-1 Nanosatélite mexicano desplegado en el espacio*. México Ciencia y Tecnología. 2019. URL: <http://www.cienciamx.com/index.php/tecnologia/nanotecnologia/26142-painani-1-nanosatelite-mexicano-desplegado-en-el-espacio> (visitado 17-06-2021).
- [25] Erik Kulu. *Painani-1 @ Nanosats Database*. Nanosats Database. s.f. URL: <https://www.nanosats.eu/sat/painani-1.html> (visitado 17-06-2021).
- [26] Antimio Cruz. *Llega a su órbita Painani-I, el primer nanosatélite mexicano*. La Crónica de Hoy. 2019. URL: https://www.cronica.com.mx/notas-llega_a_su_orbita_painani_i_el_primer_nanosatelite_mexicano-1133287-2019 (visitado 17-06-2021).
- [27] Viviana Estrella. *Querétaro, Cuarto Estado Con Más Empresas Aeroespaciales*. El Economista. 2020. URL: <https://www.economista.com.mx/estados/Queretaro-cuarto-estado-con-mas-empresas-aeroespaciales-20201201-0143.html> (visitado 18-06-2021).

- [28] Noah Weitz. «Analysis of Verification and Validation Techniques for Educational CubeSat Programs». San Luis Obispo, California: California Polytechnic State University, 1 de mayo de 2018. DOI: [10.15368/theses.2018.27](https://doi.org/10.15368/theses.2018.27). URL: <https://digitalcommons.calpoly.edu/theses/1854> (visitado 26-11-2020).
- [29] Xiwang Xia y col. «NanoSats/CubeSats ADCS Survey». En: *2017 29th Chinese Control And Decision Conference (CCDC)*. 2017 29th Chinese Control And Decision Conference (CCDC). Chongqing, China: IEEE, mayo de 2017, págs. 5151-5158. ISBN: 978-1-5090-4657-7. DOI: [10.1109/CCDC.2017.7979410](https://doi.org/10.1109/CCDC.2017.7979410). URL: <http://ieeexplore.ieee.org/document/7979410/> (visitado 15-06-2020).
- [30] A. Aydinlioglu y M. Hammer. «COMPASS-1 Pico Satellite: Magnetic Coils for Attitude Control». En: *Proceedings of 2nd International Conference on Recent Advances in Space Technologies, 2005. RAST 2005*. 2nd International Conference on Recent Advances in Space Technologies, 2005. RAST 2005. Istanbul, Turkey: IEEE, 2005, págs. 90-93. ISBN: 978-0-7803-8977-9. DOI: [10.1109/RAST.2005.1512541](https://doi.org/10.1109/RAST.2005.1512541). URL: <http://ieeexplore.ieee.org/document/1512541/> (visitado 24-04-2020).
- [31] Danil Ivanov y Dmitry Roldugin. «Nanosatellite Three-Axis Attitude Control and Determination Using Two Magnetorquers Only». En: *2019 9th International Conference on Recent Advances in Space Technologies (RAST)*. 2019 9th International Conference on Recent Advances in Space Technologies (RAST). Istanbul, Turkey: IEEE, jun. de 2019, págs. 761-768. ISBN: 978-1-5386-9448-0. DOI: [10.1109/RAST.2019.8767828](https://doi.org/10.1109/RAST.2019.8767828). URL: <https://ieeexplore.ieee.org/document/8767828/> (visitado 17-04-2020).
- [32] Arno Bergmann. «Benefits and Drawbacks of Model-Based Design». En: *KMUTNB International Journal of Applied Science and Technology* 7 (30 de sep. de 2014), págs. 15-19. DOI: [10.14416/j.ijast.2014.04.004](https://doi.org/10.14416/j.ijast.2014.04.004).
- [33] NASA. «NASA Systems Engineering Handbook». En: (2020), pág. 297.
- [34] Jonis Kiesbye y col. «Hardware-In-The-Loop and Software-In-The-Loop Testing of the MOVE-II CubeSat». En: *Aerospace* 6.12 (1 de dic. de 2019), pág. 130. ISSN: 2226-4310. DOI: [10.3390/aerospace6120130](https://doi.org/10.3390/aerospace6120130). URL: <https://www.mdpi.com/2226-4310/6/12/130> (visitado 15-12-2020).
- [35] Stephanie Wegner y col. «Methodology for Software-in-the-Loop Testing of Low-Cost Attitude Determination Systems». En: 2017.
- [36] Andrew Richard Sorensen. «Hardware-in-the-Loop Simulation for Verification of CubeSat Attitude Determination and Control Subsystem». San Luis Obispo, California: California Polytechnic State University, 1 de mar. de 2018. DOI: [10.15368/theses.2018.4](https://doi.org/10.15368/theses.2018.4). URL: <https://digitalcommons.calpoly.edu/theses/1812> (visitado 17-12-2020).
- [37] J. Mina y col. «Processor-in-the-Loop and Hardware-in-the-Loop Simulation of Electric Systems Based in FPGA». En: *2016 13th International Conference on Power Electronics (CIEP)*. 2016 13th International Conference on Power Electronics (CIEP). Jun. de 2016, págs. 172-177. DOI: [10.1109/CIEP.2016.7530751](https://doi.org/10.1109/CIEP.2016.7530751).
- [38] Kanaka Juvva. *Real-Time Systems*. Carnegie Mellon University. 1998. URL: https://users.ece.cmu.edu/~koopman/des_s99/real_time/ (visitado 17-12-2020).

- [39] Danilo José Franzim Miranda y col. «A Comparative Survey on Flight Software Frameworks for 'New Space' Nanosatellite Missions». En: *Journal of Aerospace Technology and Management* (24 de oct. de 2019), e4619. ISSN: 2175-9146. DOI: 10.5028/jatm.v11.1081. URL: http://www.scielo.br/scielo.php?script=sci_arttext&pid=S2175-91462019000100341&lng=en&nrm=iso&tlng=en (visitado 15-12-2020).
- [40] Justin Morris y col. «Simulation-to-Flight 1 (STF-1): A Mission to Enable CubeSat Software-Based Verification and Validation». En: *54th AIAA Aerospace Sciences Meeting*. 54th AIAA Aerospace Sciences Meeting. San Diego, California, USA: American Institute of Aeronautics and Astronautics, 4 de ene. de 2016. ISBN: 978-1-62410-393-3. DOI: 10.2514/6.2016-1464. URL: <http://arc.aiaa.org/doi/10.2514/6.2016-1464> (visitado 15-12-2020).
- [41] NASA. *Core Flight System*. 2020. URL: <https://cfs.gsfc.nasa.gov/Introduction.html> (visitado 29-12-2020).
- [42] Mb Utah State University. *Radiant Flight Software*.
- [43] Kubos Corporation. *KubOS Design — Kubos 1.21.0 Documentation*. 2020. URL: <https://docs.kubos.com/1.21.0/kubos-design.html> (visitado 29-12-2020).
- [44] Howard Cannon y Karen Gundy-Burlet. «Software Cost Estimation for the LADEE Mission». En: *2015 IEEE Aerospace Conference*. 2015 IEEE Aerospace Conference. Big Sky, MT: IEEE, mar. de 2015, págs. 1-8. ISBN: 978-1-4799-5379-0 978-1-4799-5380-6. DOI: 10.1109/AERO.2015.7119109. URL: <http://ieeexplore.ieee.org/document/7119109/> (visitado 15-12-2020).
- [45] AGI. *Systems Tool Kit (STK)*. Agi. 2021. URL: <https://www.agi.com/products/stk> (visitado 21-08-2021).
- [46] a.i. solutions. *FreeFlyer® Software*. a.i. solutions. 2020. URL: <https://ai-solutions.com/freelyer-astrodynamic-software/> (visitado 15-12-2020).
- [47] NASA. *GMAT Wiki - General Mission Analysis Tool (GMAT)*. 2020. URL: <https://gmat.atlassian.net/wiki/spaces/GW/overview?mode=global> (visitado 16-12-2020).
- [48] Delft University of Technology. *TU Delft Astrodynamic Toolbox Documentation*. 2018. URL: <http://tudat.tudelft.nl/started/index.html> (visitado 16-12-2020).
- [49] Kasper Fuglsang Jensen y Kasper Vinther. «Attitude Determination and Control System for AAUSAT3». Dinamarca: Aalborg University, 2010.
- [50] Ryan J Sellers. «A Gravity Gradient, Momentum-Biased Attitude Control System for A CubeSat». San Luis Obispo, California: California Polytechnic State University, 1 de mar. de 2013. DOI: 10.15368/theses.2013.23. URL: <http://digitalcommons.calpoly.edu/theses/974> (visitado 24-06-2020).
- [51] Duarte Otero de Morais Alves Rondão. «Modeling and Simulation of the ECOSat-III Attitude Determination and Control System». Portugal: Técnico Lisboa, 2016.
- [52] MathWorks. *What Is a Digital Twin?* URL: <https://www.mathworks.com/discovery/digital-twin.html> (visitado 03-07-2021).
- [53] Adil Rasheed, Omer San y Trond Kvamsdal. «Digital Twin: Values, Challenges and Enablers From a Modeling Perspective». En: *IEEE Access* 8 (2020), págs. 21980-22012. ISSN: 2169-3536. DOI: 10.1109/ACCESS.2020.2970143.

- [54] J. Bouwmeester y J. Guo. «Survey of Worldwide Pico- and Nanosatellite Missions, Distributions and Subsystem Technology». En: *Acta Astronautica* 67.7-8 (oct. de 2010), págs. 854-862. ISSN: 00945765. DOI: 10.1016/j.actaastro.2010.06.004. URL: <https://linkinghub.elsevier.com/retrieve/pii/S0094576510001955> (visitado 12-03-2021).
- [55] Olman D. Quiros-Jimenez y Duncan D'Hemecourt. «Development of a Flight Software Framework for Student CubeSat Missions». En: *Revista Tecnología en Marcha* (20 de dic. de 2019). ISSN: 2215-3241, 0379-3982. DOI: 10.18845/tm.v32i8.4992. URL: https://revistas.tec.ac.cr/index.php/tec_marcha/article/view/4992 (visitado 15-12-2020).
- [56] Pedro Filipe Rodrigues Coelho. «ISTNanosat-1 Quality Assurance, Risk Management and Assembly, Integration and Verification Planning». Portugal: Técnico Lisboa, 2016.
- [57] Frederik Bajersvej. «Attitude Control System for AAU CubeSat». Aalborg University, 2002.
- [58] James Richard Wertz y Computer Sciences Corporation, eds. *Spacecraft Attitude Determination and Control*. Astrophysics and Space Science Library v. 73. Dordrecht ; Boston: Reidel, 1978. 858 págs. ISBN: 978-90-277-0959-2.
- [59] Rômulo Fernandes da Costa, Osamu Saotome y Elvira Rafikova. «Efficacy Comparison of Satellite Attitude Control Systems Using a Spherical Air Bearing». En: *Proceedings of the 4th Brazilian Technology Symposium (BTSym'18)*. Ed. por Yuzo Iano y col. Vol. 140. Smart Innovation, Systems and Technologies. Cham: Springer International Publishing, 2019, págs. 55-64. ISBN: 978-3-030-16052-4 978-3-030-16053-1. DOI: 10.1007/978-3-030-16053-1_6. URL: http://link.springer.com/10.1007/978-3-030-16053-1_6 (visitado 10-03-2021).
- [60] Alexander Joseph Westfall. «Design of an Attitude Control System for Spin-Axis Control of a 3U CubeSat». Master of Science. San Jose, CA, USA: San Jose State University, 1 de dic. de 2015. DOI: 10.31979/etd.u6fr-6t4m. URL: https://scholarworks.sjsu.edu/etd_theses/4672 (visitado 10-03-2021).
- [61] Svein Tohami El Moussaoüi Brembo. «Sensor Modeling, Attitude Determination and Control for Micro-Satellite». Noruega: Norwegian University of Science and Technology, 2005.
- [62] Enrico Canuto y col. *Spacecraft Dynamics and Control: The Embedded Model Control Approach*. 2018. ISBN: 978-0-08-101795-1. URL: <https://app.knovel.com/hotlink/toc/id:kpSDCTEMC5/spacecraft-dynamics-control/spacecraft-dynamics-control> (visitado 09-07-2020).
- [63] CubeSatShop. *NSS Fine Sun Sensor*. 2021. URL: <https://www.cubesatshop.com/product/digital-fine-sun-sensor/> (visitado 21-08-2021).
- [64] M. Sumathi y col. «Performance Analysis of Sun Sensors for Satellite Systems». En: *2013 International Conference on Advanced Electronic Systems (ICAES)*. 2013 International Conference on Advanced Electronic Systems (ICAES). PILANI, India: IEEE, sep. de 2013, págs. 10-14. ISBN: 978-1-4799-1441-8 978-1-4799-1439-5 978-1-4799-1440-1. DOI: 10.1109/ICAES.2013.6659351. URL: <http://ieeexplore.ieee.org/document/6659351/> (visitado 20-05-2020).
- [65] Elizabeth Marie Keil. «Kalman Filter Implementation to Determine Orbit and Attitude of a Satellite in a Molniya Orbit». Virginia Tech, 2014.

REFERENCIAS

- [66] Maxwell Bezold. «An Attitude Determination System with MEMs Gyroscope Drift Compensation for Small Satellites». United States: University of Kentucky, 2013.
- [67] F. Landis Markley y John L. Crassidis. *Fundamentals of Spacecraft Attitude Determination and Control*. Space Technology Library 33. New York: Springer, 2014. 486 págs. ISBN: 978-1-4939-0801-1.
- [68] Anton H. J. De Ruiter, Christopher Damaren y James R. Forbes. *Spacecraft Dynamics and Control: An Introduction*. Chichester, West Sussex, UK: Wiley, 2013. 1 pág. ISBN: 978-1-118-40332-7 978-1-118-40330-3 978-1-118-40331-0.
- [69] João Henrique Figueiredo Freitas. «Model-in-the-Loop Simulation of the ECOSat-III Attitude Determination and Control System». Portugal, 2016.
- [70] Wikipedia. *Elementos orbitales*. En: *Wikipedia, la enciclopedia libre*. 22 de nov. de 2019. URL: https://es.wikipedia.org/w/index.php?title=Elementos_orbitales&oldid=121504192 (visitado 30-08-2021).
- [71] Mahmut Reyhanoglu, Chau Ton y Sergey Drakunov. «Attitude Stabilization of a Nadir-Pointing Small Satellite Using Only Magnetic Actuators». En: *IFAC Proceedings Volumes*. 2nd IFAC Conference on Intelligent Control Systems and Signal Processing 42.19 (1 de ene. de 2009), págs. 292-297. ISSN: 1474-6670. DOI: [10.3182/20090921-3-TR-3005.00052](https://doi.org/10.3182/20090921-3-TR-3005.00052). URL: <https://www.sciencedirect.com/science/article/pii/S1474667015308491> (visitado 15-07-2021).
- [72] AGI. *Technical Notes - Vehicle Local Coordinate Axes*. 2016. URL: <https://help.agi.com/stk/11.0.1/Content/gator/eq-coordsys.htm> (visitado 02-09-2021).
- [73] Bernt Ove Sunde. «Sensor Modelling and Attitude Determination for Micro-Satellite». Noruega: Norwegian University of Science and Technology, 2005.
- [74] Puneet Singla, D. Mortari y John Junkins. «How to Avoid Singularity When Using Euler Angles?» En: *Advances in the Astronautical Sciences* 119 (1 de ene. de 2005), págs. 1409-1426.
- [75] Jack B. Kuipers. *Quaternions and Rotation Sequences: A Primer with Applications to Orbits, Aerospace, and Virtual Reality*. Princeton, N.J: Princeton University Press, 2002. 371 págs. ISBN: 978-0-691-10298-6.
- [76] Yan-Bin Jia. *Quaternions and Rotations*. 2013.
- [77] Karsten Grojckatthöfer y Zizung Yo on. *Introduction into Quaternions for Spacecraft Attitude Representation*. 2012. URL: <http://www.tu-berlin.de/fileadmin/fg169/miscellaneous/Quaternions.pdf> (visitado 23-07-2021).
- [78] Yaguang Yang. *Spacecraft Modeling, Attitude Determination, and Control: Quaternion-Based Approach*. Boca Raton, FL: CRC Press, 2019. ISBN: 978-1-138-33150-1.
- [79] Lucian-Florentin Barbulescu y col. «Estimation of the Attitude of a CubeSat under Environmental Torques». En: *2017 21st International Conference on System Theory, Control and Computing (ICSTCC)*. 2017 21st International Conference on System Theory, Control and Computing (ICSTCC). Sinaia: IEEE, oct. de 2017, págs. 291-296. ISBN: 978-1-5386-3842-2. DOI: [10.1109/ICSTCC.2017.8107049](https://doi.org/10.1109/ICSTCC.2017.8107049). URL: <http://ieeexplore.ieee.org/document/8107049/> (visitado 15-06-2020).

- [80] Alessandro de Iaco Veris. *Practical Astrodynamics*. Springer Aerospace Technology. Cham: Springer International Publishing, 2018. ISBN: 978-3-319-62219-4 978-3-319-62220-0. DOI: [10.1007/978-3-319-62220-0](https://doi.org/10.1007/978-3-319-62220-0). URL: <http://link.springer.com/10.1007/978-3-319-62220-0> (visitado 09-07-2020).
- [81] David A Vallado y Wayne D McClain. *Fundamentals of Astrodynamics and Applications*. New York: McGraw-Hill, 1997. ISBN: 978-0-07-066829-4 978-0-07-066834-8.
- [82] Oliver Montenbruck y Gill Eberhard. *Satellite Orbits: Models, Methods, and Applications*. Berlin : New York: Springer, 2000. 369 págs. ISBN: 978-3-540-67280-7.
- [83] Elena Odriozola Olavarria y Guillermo Ramos Hernandez. «Atmospheric Models Evaluation for Space Applications». En: (), pág. 85.
- [84] AGI. *HPOP Force Models: Drag*. STK Help. 2016. URL: <https://help.agi.com/stk/index.htm#hpop/hpopDrag.htm#MarsGRAM> (visitado 03-07-2020).
- [85] Christopher D. Hall. *Spacecraft Attitude Dynamics and Control*. 12 de ene. de 2003.
- [86] Patrick Alken. *IAGA V-MOD Geomagnetic Field Modeling: International Geomagnetic Reference Field IGRF-13*. 2019. URL: <https://www.ngdc.noaa.gov/IAGA/vmod/igrf.html> (visitado 30-07-2021).
- [87] M. D. Shuster y S. D. Oh. «Three-Axis Attitude Determination from Vector Observations». En: *Journal of Guidance and Control* 4.1 (ene. de 1981), págs. 70-77. ISSN: 0162-3192. DOI: [10.2514/3.19717](https://doi.org/10.2514/3.19717). URL: <https://arc.aiaa.org/doi/10.2514/3.19717> (visitado 21-10-2020).
- [88] Dan Simon. *Optimal State Estimation: Kalman, H [Infinity] and Nonlinear Approaches*. Hoboken, N.J: Wiley-Interscience, 2006. 526 págs. ISBN: 978-0-471-70858-2.
- [89] Halil Ersin Soken, Chingiz Hajiyev y Shin-ichiro Sakai. «Robust Kalman Filtering for Small Satellite Attitude Estimation in the Presence of Measurement Faults». En: *European Journal of Control* 20.2 (mar. de 2014), págs. 64-72. ISSN: 09473580. DOI: [10.1016/j.ejcon.2013.12.002](https://doi.org/10.1016/j.ejcon.2013.12.002). URL: <https://linkinghub.elsevier.com/retrieve/pii/S0947358013002094> (visitado 18-08-2021).
- [90] M. D. Pham y col. «Gain-Scheduled Extended Kalman Filter for Nanosatellite Attitude Determination System». En: *IEEE Transactions on Aerospace and Electronic Systems* 51.2 (abr. de 2015), págs. 1017-1028. ISSN: 0018-9251. DOI: [10.1109/TAES.2014.130204](https://doi.org/10.1109/TAES.2014.130204). URL: <http://ieeexplore.ieee.org/document/7126161/> (visitado 28-12-2020).
- [91] Mohamed Temam Nasri y Witold Kinsner. «Extended Kalman Filtering for Pico-Satellites Attitude Determination». En: *2013 26th IEEE Canadian Conference on Electrical and Computer Engineering (CCECE)*. 2013 26th IEEE Canadian Conference on Electrical and Computer Engineering (CCECE). Regina, SK, Canada: IEEE, mayo de 2013, págs. 1-5. ISBN: 978-1-4799-0033-6 978-1-4799-0031-2 978-1-4799-0032-9. DOI: [10.1109/CCECE.2013.6567843](https://doi.org/10.1109/CCECE.2013.6567843). URL: <http://ieeexplore.ieee.org/document/6567843/> (visitado 18-08-2021).
- [92] Chingiz Hajiyev, Demet Cilden y Yevgeny Somov. «Gyro-Free Attitude and Rate Estimation for a Small Satellite Using SVD and EKF». En: *Aerospace Science and Technology* 55 (ago. de 2016), págs. 324-331. ISSN: 12709638. DOI: [10.1016/j.ast.2016.06.004](https://doi.org/10.1016/j.ast.2016.06.004). URL: <http://linkinghub.elsevier.com/retrieve/pii/S1270963816302139> (visitado 18-08-2021).

REFERENCIAS

- [93] Cesar A Castellanos y Lilia Edith Aparicio Pico. «Design and Simulation of an Attitude Determination System Based on the Extended Kalman Filter for Cube-Sat Colombia I». En: (2014), pág. 9.
- [94] Marco Lovera. «Magnetic Satellite Detumbling: The b-Dot Algorithm Revisited». En: *2015 American Control Conference (ACC)*. 2015 American Control Conference (ACC). Chicago, IL, USA: IEEE, jul. de 2015, págs. 1867-1872. ISBN: 978-1-4799-8684-2. DOI: [10.1109/ACC.2015.7171005](https://doi.org/10.1109/ACC.2015.7171005). URL: <http://ieeexplore.ieee.org/document/7171005/> (visitado 18-08-2021).
- [95] Grzegorz Juchnikowski, Tomasz Barcinski y Jakub Lisowski. «Optimal Control Gain for Satellite Detumbling Using B-Dot Algorithm». En: (2013), pág. 12.
- [96] Robert Fonod y Eberhard Gill. «Magnetic Detumbling of Fast-Tumbling Picosatellites». En: 1 de oct. de 2018.
- [97] Ho-Nien Shou, Jia-Shing Sheu y Jyh-Haw Wang. «Micro-Satellite Detumbling Mode Attitude Determination and Control : UKF Approach». En: *IEEE ICCA 2010*. 2010 8th IEEE International Conference on Control and Automation (ICCA). Xiamen, China: IEEE, jun. de 2010, págs. 673-678. ISBN: 978-1-4244-5195-1. DOI: [10.1109/ICCA.2010.5524162](https://doi.org/10.1109/ICCA.2010.5524162). URL: <http://ieeexplore.ieee.org/document/5524162/> (visitado 18-08-2021).
- [98] M. Lovera y A. Astolfi. «Spacecraft Attitude Control Using Magnetic Actuators». En: *Automatica* 40.8 (1 de ago. de 2004), págs. 1405-1414. ISSN: 0005-1098. DOI: [10.1016/j.automatica.2004.02.022](https://doi.org/10.1016/j.automatica.2004.02.022). URL: <https://www.sciencedirect.com/science/article/pii/S0005109804000767> (visitado 13-07-2021).
- [99] Chau Ton. «Magnetic Stabilization of Nadir-Pointing Small Satellites». Embry-Riddle Aeronautical University, 2009.
- [100] Simone Battistini, Chantal Cappelletti y Filippo Graziani. «Results of the Attitude Reconstruction for the UniSat-6 Microsatellite Using in-Orbit Data». En: *Acta Astronautica* 127 (1 de mayo de 2016). DOI: [10.1016/j.actaastro.2016.05.020](https://doi.org/10.1016/j.actaastro.2016.05.020).
- [101] Fabio Celani. «Gain Selection for Attitude Stabilization of Earth-Pointing Spacecraft Using Magnetorquers». En: *Aerotecnica Missili & Spazio* 100.1 (1 de mar. de 2021), págs. 15-24. ISSN: 2524-6968. DOI: [10.1007/s42496-020-00062-2](https://doi.org/10.1007/s42496-020-00062-2). URL: <https://doi.org/10.1007/s42496-020-00062-2> (visitado 15-07-2021).
- [102] Pedro A. Capó-Lugo y Peter M. Bainum. «4 - Frame Rotations and Quaternions». En: *Orbital Mechanics and Formation Flying*. Ed. por Pedro A. Capó-Lugo y Peter M. Bainum. Woodhead Publishing, 1 de ene. de 2011, págs. 75-91. ISBN: 978-0-85709-054-6. DOI: [10.1533/9780857093875.75](https://doi.org/10.1533/9780857093875.75). URL: <https://www.sciencedirect.com/science/article/pii/B9780857090546500040> (visitado 19-08-2021).
- [103] NovAtel. «OEM7 Commands and Logs Reference Manual». En: (2021), pág. 1192.
- [104] NovAtel. *OEM6® Family Firmware Reference Manual*. 2017. URL: <https://hexagondownloads.blob.core.windows.net/public/Novatel/assets/Documents/Manuals/om-20000129/om-20000129.pdf> (visitado 20-08-2021).
- [105] UAT FI UNAM. *Proyecto Koto*. 2021. URL: <https://www.ingenieria.unam.mx/k-oto/> (visitado 16-04-2021).

- [106] MathWorks. *Load Signal Data from Workspace into Simulink Model - Simulink*. 2021. URL: https://www.mathworks.com/help/simulink/slref/fromworkspace.html#mw_cbb1cbe6-510f-4f49-924f-f9f86c15f257_head (visitado 25-08-2021).
- [107] MathWorks. *Compare Solvers - MATLAB & Simulink*. 2021. URL: <https://www.mathworks.com/help/simulink/ug/compare-solvers.html> (visitado 27-08-2021).



"Spacewalkers Thomas Pesquet and Akihiko Hoshide", NASA (2021)

