



ESCUELA DE INGENIERÍA DE FUENLABRADA

GRADO EN INGENIERÍA BIOMÉDICA

TRABAJO DE FIN DE GRADO

**ESTIMACIÓN DE ÁNGULOS DEL TREN INFERIOR MEDIANTE IMUs
Y FILTRO DE MADGWICK PARA SU APLICACIÓN EN
EXOESQUELETOS PARA REHABILITACIÓN**

AUTOR: PABLO DE MIGUEL PUEYO

TUTOR: SANTIAGO MURANO

CO-TUTOR: RUBÉN NIETO CAPUCHINO

CURSO ACADÉMICO: 2023/2024

©2024 Pablo de Miguel Pueyo
Algunos derechos reservados
Este documento se distribuye bajo la licencia
“Atribución 4.0 Internacional” de Creative Commons,
disponible en
<https://creativecommons.org/licenses/by-sa/4.0/deed.es>

Todo el código desarrollado específicamente para este proyecto estará disponible en el repositorio de GitHub: <https://github.com/pabdm>

Índice

Agradecimientos	1
Resumen	2
Índice de figuras	3
Lista de acrónimos y abreviaturas	6
Capítulo 1: Introducción y objetivos	7
1.1. Contexto y motivación	7
1.2. Objetivos	8
1.3. Metodología	9
1.4. Estructura de la memoria	11
Capítulo 2: Conceptos previos	13
2.1. Exoesqueleto	13
2.2. Inertial Measurement Unit (IMU)	15
2.3. Ángulos de Euler y matrices de rotación	17
2.4. Cuaterniones	20
Capítulo 3: Métodos de estimación	23
3.1. Filtro de Kalman	24
3.1.1. Filtro de Kalman Extendido (EFK)	26

3.1.2. Otros Filtros de Kalman	29
3.1.2.1. Error-state Kalman Filter (ErKF)	30
3.1.2.2. Filtro Kalman Unscented (UKF)	30
3.1.2.3. Filtro Kalman de Rango Variable (VKF)	30
3.1.2.4. Filtro Kalman Complementario (CKF)	30
3.2. Filtro de Madgwick	31
3.3. Ventajas del Filtro de Madgwick frente al Filtro de Kalman Extendido	41
Capítulo 4: Bases de datos	43
4.1. Primera opción: síntesis base de datos	43
4.2. Segunda opción: base de datos con datos de magnetómetro	52
4.3. Tercera opción: base de datos con IMUs propias	55
Capítulo 5: Experimentos y resultados	63
5.1. Experimentos con la primera base de datos	63
5.3. Experimentos con la segunda base de datos	65
5.3. Experimentos con la tercera base de datos	66
5.3.1. Desarrollo del código principal	66
5.3.2. Ajuste de parámetros	68
5.3.3. Resultados finales	71
5.4. Discusión	83
Capítulo 6: Conclusión	85
Referencias	87

Agradecimientos

Quiero expresar mi más sincero agradecimiento a todas las personas que han hecho posible que este proyecto salga adelante. Este trabajo es la síntesis de cuatro años de esfuerzo y el fin de una de las mejores etapas de mi vida.

En primer lugar, me gustaría agradecer a mis tutores Santiago y Rubén, por su orientación, paciencia y sobre todo por su apoyo constante durante todos estos meses. Desde el primer momento han estado encima y siempre a disposición para cualquier consulta.

También merece una mención especial mi amigo y compañero Daniel, con el que he colaborado en algunos partes y cuya constancia y perseverancia han sido fuente de inspiración para seguir incluso en los momentos más difíciles.

Gracias, por su puesto, a mis compañeros y amigos de la carrera, por hacer de este viaje una experiencia inolvidable, y sobre todo por hacerme ser quien soy. También a los amigos de *siempre*, por *siempre* estar ahí.

No puedo olvidar a mi familia, que con su amor y comprensión me han brindado el soporte emocional que necesitaba para enfrentar los desafíos de esta larga y corta etapa académica. A mis padres y mi hermano, por su confianza incondicional y por inculcarme los valores del esfuerzo y la dedicación.

Finalmente, agradezco a todas las personas y entidades que, de manera directa o indirecta, han contribuido a la culminación de este proyecto.

Gracias a todos.

Resumen

Este Trabajo de Fin de Grado (TFG) explora la aplicación del Filtro de Madgwick en la estimación de ángulos del tren inferior para la rehabilitación con exoesqueletos en pacientes en movimiento. La investigación se centra en mejorar la precisión y estabilidad en la estimación de la orientación de las extremidades inferiores, esencial para el desarrollo de exoesqueletos que asistan en la rehabilitación de personas con disfunciones motoras.

Se comparan los filtros Madgwick y Kalman en términos de precisión, robustez y eficiencia computacional. La metodología incluye una revisión de la literatura, la selección y síntesis de bases de datos de captura de movimiento, y la implementación y validación de los algoritmos de filtrado.

Los resultados demuestran que el Filtro de Madgwick, al utilizar cuaterniones para representar rotaciones, ofrece ventajas significativas en términos de eficiencia computacional y robustez frente a errores de sensores, superando al Filtro de Kalman extendido en estas áreas. Los experimentos muestran la efectividad del filtro Madgwick en entornos reales, destacando su potencial para su integración en sistemas de exoesqueletos en tiempo real.

Finalmente, se discuten las implicaciones de estos hallazgos para el diseño de exoesqueletos de rehabilitación y se proponen líneas futuras de investigación para optimizar y ampliar la aplicabilidad de los filtros de orientación en contextos clínicos y terapéuticos.

Índice de figuras

Figura 1: Diagrama de Gantt del Trabajo de Fin de Grado.	11
Figura 2: Fotografía del exoesqueleto Exo-H3 de la marca Technaid [12].	15
Figura 3: Representación de los ángulos de Euler en 3 dimensiones [23].	17
Figura 4: Representación gráfica del bloqueo de Cardán [27].	20
Figura 5: Representación de un cuaternión [28].	22
Figura 6: Diagrama del algoritmo del Filtro de Kalman [33].	26
Figura 7: Diagrama del algoritmo del Filtro de Kalman Extendido [35].	29
Figura 8: Diagrama del algoritmo del Filtro de Madgwick [45].	33
Figura 9: Estructura de los datos de orientación de la base sintetizada.	46
Figura 10: Estructura de los datos de aceleración de la base sintetizada.	46
Figura 11: Estructura de los datos de tipo de movimiento de la base sintetizada.	47
Figura 12: Uso de las herramientas de Aitviewer para la visualización de los vértices.	47
Figura 13: Uso de las herramienta de Aitviewer para la navegación sobre las articulaciones del modelo.	48
Figura 14: Visualización en Aitviewer de la IMU de referencia de la zona lumbar.	48
Figura 15: Visualización en Aitviewer de la IMU las IMUs virtuales.	49
Figura 16: Gráficas comparativas entre la aceleración sin ruido (azul) y con ruido gaussiano (rojo). ..	51
Figura 17: Gráficas comparativas entre la orientación sin ruido (azul) y con ruido gaussiano (rojo). ..	51
Figura 18: Disposición de los sensores del experimento [51].	54
Figura 19: Unidad de Medida Inercial de 3 ejes, Tech-IMU [52].	56
Figura 20: Modelo del Tech-HUB V3 [52].	56

Figura 21: Interfaz del software Tech MCS Studio [53]	57
Figura 22: Esquema del proceso interno del Tech-IMU para obtener la orientación 3D [52].	59
Figura 23: Accesorios elásticos para colocación de IMUs [52].	60
Figura 24: Sujeto de prueba con el sistema Tech-MCS V3 colocado.	61
Figura 25: Cinturón con el Tech-HUB V3.....	61
Figura 26: Representación en ángulos de Euler en grados de la primera IMU con datos sintetizados con ruido añadido.	64
Figura 27: Representación en ángulos de Euler en grados de la primera IMU con datos sintetizados con ruido añadido y con cuaternión próximo a los valores iniciales.....	64
Figura 28: Representación en ángulos de Euler en grados de la segunda base. Captura de IMU de la tibia derecha de sujeto caminando.	65
Figura 29: Representación en ángulos de Euler en grados calculados por los creadores de la segunda base. Captura de IMU de la tibia derecha de sujeto caminando.	66
Figura 30: Comparación del cuaternión q_W con diferentes combinaciones de ζ y β en primera prueba.	70
Figura 31: Comparación del cuaternión q_W con diferentes combinaciones de ζ y β en segunda prueba.	70
Figura 32: Comparación entre la señal filtrada con el Filtro de Madgwick y el Filtro de Kalman Extendido. Captura de IMU en el muslo derecho de muestra de marcha.	71
Figura 33: Diferencia entre la señal filtrada con el Filtro de Madgwick y el Filtro de Kalman Extendido. Captura de IMU en el muslo derecho de muestra de marcha.	72
Figura 34: Comparación entre la señal filtrada con el Filtro de Madgwick y el Filtro de Kalman Extendido. Captura de IMU en el pie derecho de muestra de marcha.....	73
Figura 35: Diferencia entre la señal filtrada con el Filtro de Madgwick y el Filtro de Kalman Extendido. Captura de IMU en el pie derecho de muestra de marcha.....	74

Figura 36: Comparación entre la señal filtrada con el Filtro de Madgwick y el Filtro de Kalman Extendido. Captura de IMU en el muslo derecho de muestra de sujeto sentándose.	75
Figura 37: Diferencia entre la señal filtrada con el Filtro de Madgwick y el Filtro de Kalman Extendido. Captura de IMU en el muslo derecho de muestra de sujeto sentándose.	76
Figura 38: Comparación entre la señal filtrada con el Filtro de Madgwick y el Filtro de Kalman Extendido. Captura de IMU en el muslo derecho de muestra de sujeto parado de pie.	77
Figura 39: Diferencia entre la señal filtrada con el Filtro de Madgwick y el Filtro de Kalman Extendido. Captura de IMU en el muslo derecho de muestra de sujeto parado de pie.	78
Figura 40: Representación de ángulos de Euler en radianes con problema de discontinuidades de salto. Captura de IMU en el muslo derecho de muestra de marcha.	79
Figura 41: Representación de los ángulos de Euler en grados. Captura de IMU en el muslo derecho de muestra de marcha.	80
Figura 42: Diferencia de los ángulos de Euler en grados. Captura de IMU en el muslo derecho de muestra de marcha.	81
Figura 43: Representación de los ángulos de Euler en grados. Captura de IMU en el muslo derecho de muestra de sujeto sentándose.	82
Figura 44: Diferencia de los ángulos de Euler en grados. Captura de IMU en el muslo derecho de muestra de sujeto sentándose.	83

Lista de acrónimos y abreviaturas

TFG: Trabajo de Fin de Grado

IMU: Inertial Measurement Unit

KF: Filtro de Kalman

EKF: Filtro de Kalman Extendido

ErKF: Filtro de Kalman de Error de Estado

UKF: Filtro de Kalman Unscented

VKF: Filtro de Kalman de Rango Variable

CKF: Filtro de Kalman Complementario

MF: Filtro de Madgwick

SMPL: Skinned Multi-Person Linear Model

Capítulo 1

Introducción y objetivos

En este capítulo se aborda la problemática de la rehabilitación en pacientes con disfunciones motoras, en particular, se enfoca en la estimación de ángulos del tren inferior mediante el uso de filtros Madgwick. Se contextualiza la importancia de esta área de investigación en el ámbito de la rehabilitación con exoesqueletos, destacando la necesidad de herramientas precisas y eficaces para mejorar la calidad de vida de los pacientes en movimiento. A continuación, se detallan los objetivos del proyecto, se describe la metodología empleada, respaldada por un diagrama de Gantt para una planificación eficiente, y se proporciona una visión general del contenido que se desarrollará en los siguientes apartados de la memoria.

1.1. Contexto y motivación

En la actualidad, aproximadamente el 9,49% de la población española, equivalente a unos 4,38 millones de individuos, enfrenta desafíos asociados con la discapacidad, lo que resalta la necesidad imperativa de contar con servicios de atención rehabilitadora efectivos y accesibles [1]. La atención rehabilitadora tradicional, mayormente proporcionada en entornos hospitalarios, se enfrenta a una demanda significativa de servicios, lo que a menudo se traduce en largos tiempos de espera y limitaciones en la disponibilidad de recursos, obstaculizando el acceso oportuno a la rehabilitación

para muchos individuos con discapacidades. En este contexto, se hace evidente la importancia de explorar alternativas que permitan abordar estas necesidades de manera más eficiente y efectiva [2].

La rehabilitación con exoesqueletos emerge como una de estas alternativas, ofreciendo nuevas perspectivas para mejorar la calidad de vida de pacientes con disfunciones motoras. Los exoesqueletos, dispositivos biomecánicos diseñados para asistir, mejorar o restaurar la función motora de las extremidades, juegan un papel crucial en la recuperación funcional de individuos afectados por lesiones medulares [3], accidentes cerebrovasculares [4] o enfermedades neurológicas [5].

Dentro de las necesidades técnicas de funcionamiento de los exoesqueletos, la estimación precisa de los ángulos de las articulaciones se vuelve fundamental para el diseño y la implementación efectiva de los sistemas de control que proveen las terapias de rehabilitación con exoesqueletos.

Los ángulos del tren inferior, que abarcan la cadera, la rodilla y el tobillo, son indicadores clave del movimiento humano, fundamentales para actividades motoras cotidianas como caminar, correr o subir escaleras. Por lo tanto, una estimación precisa de estos ángulos es esencial para garantizar una interacción segura y eficiente entre el usuario y el exoesqueleto. Los dispositivos que se emplean para obtener estas medidas son las Unidades de Medición Inercial (IMU).

Es en este contexto que los filtros de orientación, como los filtros Kalman o Madgwick, han surgido como una herramienta prometedora para mejorar la precisión y la estabilidad en la estimación de la orientación espacial de los segmentos corporales durante el movimiento. El filtro Madgwick destaca frente al Filtro de Kalman por ventajas significativas en términos de eficiencia computacional y capacidad para integrarse con sistemas de exoesqueletos en tiempo real.

1.2. Objetivos

El presente trabajo tiene como objetivo principal investigar y evaluar la aplicación de los filtros Madgwick en la estimación de ángulos del tren inferior para la rehabilitación con exoesqueletos en pacientes en movimiento. Con este propósito, se plantean los siguientes objetivos específicos:

- Seleccionar una base de datos adecuada que contenga registros de movimiento de pacientes con las variables físicas que necesita un exoesqueleto en entornos clínicos o terapéuticos. Esta base de datos se utilizará para realizar experimentos y evaluar el rendimiento de los filtros Madgwick en la estimación de ángulos del tren inferior durante el movimiento dinámico.
- Implementar y comparar los filtros Madgwick con los filtros Kalman en términos de precisión, robustez y eficiencia computacional en la estimación de los ángulos del tren inferior. Esta comparación se llevará a cabo utilizando la misma base de datos.
- Evaluar la capacidad de los filtros Madgwick para proporcionar estimaciones precisas y estables de los ángulos del tren inferior durante actividades motoras, como caminar.

1.3. Metodología

Para la realización de este proyecto se siguió una metodología estructurada en varias fases, cada una de las cuales se detalla a continuación.

La primera fase del proyecto consistió en una exploración exhaustiva de la literatura existente relacionada con el uso de sensores inerciales (IMUs), filtros de orientación (como el Filtro de Madgwick y el Filtro de Kalman Extendido), y bases de datos de captura de movimiento. Esta revisión bibliográfica fue importante para comprender los fundamentos teóricos y las aplicaciones prácticas de las tecnologías y métodos utilizados en el proyecto. Durante esta fase, se evaluaron diversas estrategias y enfoques para abordar el problema de la síntesis y análisis de datos de IMUs, permitiendo seleccionar las metodologías más adecuadas para el desarrollo del proyecto.

Una vez obtenida una comprensión sólida de los conceptos teóricos, se procedió a la búsqueda de bases de datos de captura de movimiento que incluyeran lecturas de acelerómetros, giroscopios y magnetómetros. Inicialmente, se exploraron varias opciones disponibles en la literatura y en repositorios públicos. También se llevó a cabo la síntesis de la base de datos, tanto de manera digital

como física. Las bases de datos seleccionadas fueron evaluadas en términos de calidad, variedad de movimientos capturados y la disposición de los sensores en el cuerpo humano.

Una parte fundamental del proyecto fue la implementación y validación del Filtro de Madgwick (MF) para estimar la orientación del cuerpo a partir de los datos de los sensores inerciales. Se desarrolló un código que integra el Filtro de Madgwick con los datos de acelerómetro, giroscopio y magnetómetro, permitiendo calcular cuaterniones y convertirlos en ángulos de Euler. Este proceso incluyó la verificación de los resultados obtenidos mediante comparaciones con los datos proporcionados por las propias IMUs.

Los datos sintéticos generados a partir de imágenes de video y los datos reales proporcionados por las IMUs fueron sometidos a un proceso de preprocesamiento usando distintas librerías de Python. Este paso incluyó la normalización de los datos y la preparación de los datos para su análisis. Además, se añadió ruido a los datos sintéticos, simulando las condiciones del mundo real y mejorando la robustez del modelo.

Se realizaron comparaciones detalladas entre los ángulos de Euler y los cuaterniones calculados por el filtro de Madgwick y los proporcionados por las IMUs. Estas comparaciones permitieron evaluar la precisión y la consistencia del filtro de Madgwick en diferentes condiciones. Se representaron de forma gráfica las diferencias entre los resultados, facilitando el análisis y la interpretación de los datos.

Finalmente, se evaluaron los resultados obtenidos en cada fase del proyecto. Se analizaron las ventajas y limitaciones de los métodos utilizados, se discutieron las posibles mejoras y se presentaron las conclusiones del proyecto. Esta evaluación incluyó la valoración de la precisión de los filtros de orientación, la calidad de los datos sintetizados y la efectividad de las técnicas de preprocesamiento y análisis implementadas.

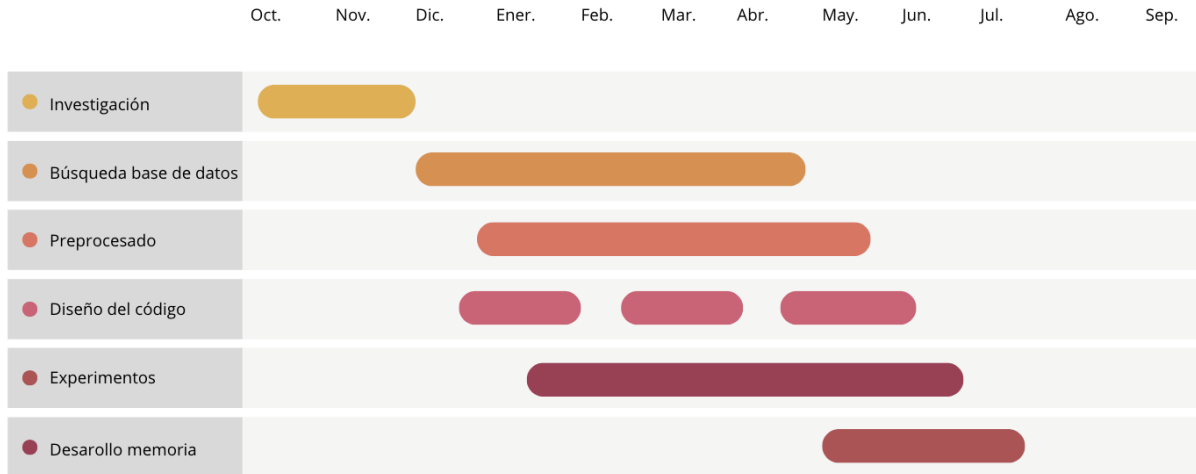


Figura 1: Diagrama de Gantt del Trabajo de Fin de Grado.

1.4. Estructura de la memoria

El contenido de la memoria se organizará de la siguiente manera:

- **Capítulo 1: Introducción y objetivos:** En este capítulo inicial, se proporciona una visión general del contexto y la motivación del estudio, así como una descripción detallada de los objetivos específicos que se abordarán a lo largo de la investigación.
- **Capítulo 2: Conceptos previos:** Aquí se presentan los conceptos teóricos y técnicos fundamentales relacionados con la rehabilitación con exoesqueletos, incluyendo aspectos biomecánicos, tecnológicos y clínicos relevantes para comprender el marco de trabajo del estudio.
- **Capítulo 3: Metodología:** Se describe en detalle la metodología utilizada en el estudio, incluyendo los procedimientos experimentales, los criterios de selección de la base de datos, la

implementación de los filtros Madgwick y Kalman, y cualquier otra técnica o herramienta metodológica utilizada en la investigación.

- **Capítulo 4: Base de datos:** Este capítulo se centra en la selección y caracterización de la base de datos utilizada en el estudio. Se detallan los criterios de selección, el proceso de adquisición de datos, el preprocesamiento realizado y cualquier otra consideración relevante sobre la base de datos utilizada.
- **Capítulo 5: Experimentos y resultados:** Aquí se presentan los experimentos llevados a cabo durante la investigación, así como los resultados obtenidos de dichos experimentos. Se incluyen análisis cuantitativos y cualitativos de los datos, así como visualizaciones y gráficos para respaldar la interpretación de los resultados.
- **Capítulo 6: Conclusiones y líneas futuras:** En este último capítulo, se presentan las conclusiones principales derivadas del estudio, así como posibles direcciones futuras de investigación basadas en los hallazgos obtenidos. Se discuten las implicaciones prácticas y teóricas de los resultados y se ofrecen recomendaciones para investigaciones posteriores.

Capítulo 2

Conceptos previos

2.1. Exoesqueleto

Los exoesqueletos son dispositivos biomecánicos diseñados para asistir, mejorar o restaurar la función motora de las extremidades. Estos dispositivos pueden ser utilizados en una variedad de contextos, que van desde aplicaciones industriales y militares hasta la rehabilitación médica. En el ámbito industrial [6, 7, 8], los exoesqueletos se utilizan para reducir la carga física [9] sobre los trabajadores y prevenir lesiones laborales. Sin embargo, el enfoque de este trabajo se centra en el uso de exoesqueletos para la rehabilitación de personas con discapacidades motoras.

En el contexto de la rehabilitación, los exoesqueletos se emplean para ayudar a pacientes con limitaciones motoras a recuperar la movilidad y mejorar su calidad de vida. Estos dispositivos son especialmente beneficiosos para individuos que han sufrido lesiones medulares, accidentes cerebrovasculares o padecen enfermedades neuromusculares. Al proporcionar soporte y asistencia en los movimientos, los exoesqueletos pueden facilitar el proceso de rehabilitación y permitir a los pacientes realizar ejercicios terapéuticos que serían difíciles o imposibles de llevar a cabo de manera autónoma [10].

Existen exoesqueletos diseñados para diferentes partes del cuerpo, incluyendo el tren superior (brazos y hombros) y el tren inferior (piernas y caderas). Este estudio se centrará específicamente en los exoesqueletos del tren inferior. Estos dispositivos están diseñados para asistir en movimientos

complejos de las extremidades inferiores, como caminar, correr, y subir escaleras. Los exoesqueletos del tren inferior pueden variar desde dispositivos que asisten parcialmente en ciertos movimientos hasta aquellos que proporcionan soporte completo, permitiendo a los usuarios realizar actividades motoras que de otro modo no podrían hacer.

El uso de exoesqueletos en la rehabilitación del tren inferior presenta varios beneficios. Ayudan a mejorar la fuerza muscular, la coordinación y el equilibrio de los pacientes. Además, permiten a los terapeutas controlar y personalizar las sesiones de rehabilitación de acuerdo con las necesidades específicas de cada paciente, monitoreando su progreso en tiempo real [11].

Sin embargo, para que los exoesqueletos sean verdaderamente efectivos en la rehabilitación, es crucial contar con sistemas precisos para la estimación y control de los movimientos. En este sentido, la precisión en la estimación de los ángulos del tren inferior, como la cadera, la rodilla o el tobillo son esenciales para asegurar que el exoesqueleto pueda adaptarse correctamente a los movimientos del usuario.

Este Trabajo de Fin de Grado (TFG) se enmarca dentro de un proyecto más amplio sobre el desarrollo y aplicación de un exoesqueleto de la Universidad Rey Juan Carlos. El proyecto general, llevado a cabo por un equipo multidisciplinar, tiene como objetivo principal precisamente la creación de una terapia de rehabilitación basada en el uso de un exoesqueleto avanzado que facilite la rehabilitación de pacientes con discapacidades motoras severas.

La terapia de rehabilitación en desarrollo en el proyecto NIMBLE ha sido diseñada específicamente para asistir en la rehabilitación del tren inferior. Este exoesqueleto se basa en avances recientes en neurociencia y tecnologías robóticas para optimizar la recuperación funcional de la marcha en pacientes que han sufrido lesiones del sistema nervioso central, como lesiones medulares y accidentes cerebrovasculares. El modelo usado fue el exoesqueleto Exo-H3 de la empresa Technaid (ver Figura 2).



Figura 2: Fotografía del exoesqueleto Exo-H3 de la marca Technaid [12].

El dispositivo cuenta con múltiples sensores inerciales que recopilan datos en tiempo real sobre los movimientos del usuario. Estos datos deben ser procesados mediante algoritmos avanzados, como los que se discutirán en este trabajo, para estimar con precisión, entre otras cosas, los ángulos de las articulaciones del tren inferior, incluyendo la cadera, la rodilla y el tobillo. La información obtenida debe permitir ajustar y personalizar la asistencia proporcionada por el exoesqueleto, asegurando que se adapte correctamente a los movimientos naturales del usuario y evitando posturas incómodas o movimientos forzados.

2.2. Inertial Measurement Unit (IMU)

Una Unidad de Medición Inercial (IMU, por sus siglas en inglés, Inertial Measurement Unit) es un dispositivo electrónico que mide y reporta la aceleración, la velocidad angular y, a veces, el campo magnético de un objeto al que está adherido [13]. Las IMU son fundamentales en una variedad de aplicaciones, como la navegación, vehículos [14], robótica [15], control de movimiento [16],

microcirugías [17] o rehabilitación [18]. Los componentes principales de una IMU son el acelerómetro, el giroscopio y, en algunos casos, el magnetómetro.

- **Acelerómetro:** Este sensor mide la aceleración lineal en tres ejes perpendiculares (x, y, z). Los acelerómetros pueden detectar tanto la aceleración debida al movimiento del dispositivo como la aceleración debida a la gravedad. Esta capacidad es importante para determinar la inclinación y la dirección de movimiento del dispositivo.
- **Giroscopio:** El giroscopio mide la velocidad angular en torno a los tres ejes (roll, pitch y yaw). Proporciona información sobre la tasa de cambio de la orientación del dispositivo, lo que es esencial para calcular la orientación precisa y para detectar rotaciones rápidas o movimientos bruscos.
- **Magnetómetro:** Algunos IMU incluyen un magnetómetro, que mide la intensidad y dirección del campo magnético terrestre en tres ejes. Esta información es útil para corregir el *drift* o deriva en las mediciones del giroscopio y para proporcionar una referencia absoluta de orientación con respecto al norte magnético.

La integración de estas tres fuentes de datos permite que una IMU proporcione una imagen completa y precisa de la orientación y el movimiento del dispositivo al que está adherida.

En el uso de exoesqueletos para la rehabilitación, estos dispositivos son fundamentales para monitorear y controlar los movimientos del usuario. Al colocarlas en diferentes segmentos del cuerpo, como el muslo, la pierna y el pie, es posible capturar la dinámica completa de los movimientos del tren inferior. Los datos obtenidos se procesan para estimar los ángulos de las articulaciones, que son esenciales para ajustar y controlar los actuadores del exoesqueleto en tiempo real.

Uno de los desafíos principales en el uso de IMU es la acumulación de errores a lo largo del tiempo, especialmente en los giroscopios, un fenómeno conocido como *drift* [13]. Para mitigar este problema, se utilizan técnicas de fusión de sensores, como los filtros Kalman [19] y Madgwick, que combinan los datos de los acelerómetros, giroscopios y magnetómetros para proporcionar estimaciones más precisas y estables de la orientación y el movimiento.

2.3. Ángulos de Euler y matrices de rotación

Los ángulos de Euler son una forma de representar la orientación de un cuerpo rígido en el espacio tridimensional. Se utilizan comúnmente en robótica [20], computación gráfica [21], aeronáutica [22] y en muchas aplicaciones de ingeniería para describir las rotaciones y la orientación de un objeto. Los ángulos de Euler consisten en tres valores angulares que corresponden a tres rotaciones consecutivas alrededor de los ejes de coordenadas.

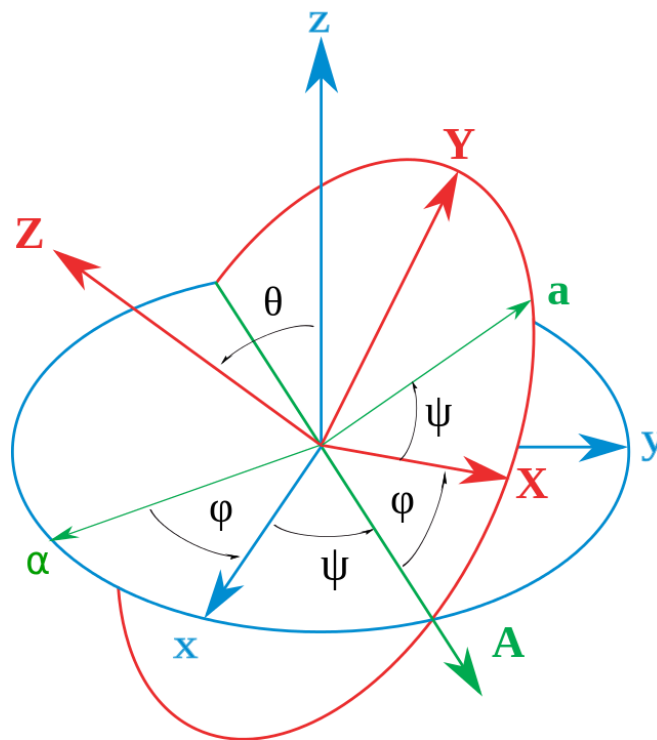


Figura 3: Representación de los ángulos de Euler en 3 dimensiones [23].

Los ángulos de Euler se definen mediante tres rotaciones sucesivas alrededor de los ejes de un sistema de coordenadas. Existen diferentes convenciones para estas rotaciones, pero una de las más comunes es la convención "zyx" [24, 25], que consiste en:

1. **Rotación alrededor del eje z (yaw, ψ):** La primera rotación se realiza alrededor del eje z del sistema de coordenadas inicial. Este ángulo se denomina yaw y define la orientación del objeto en el plano xy.
2. **Rotación alrededor del eje y (pitch, θ):** La segunda rotación se realiza alrededor del eje y del sistema de coordenadas, que ha sido previamente rotado por el primer ángulo. Este ángulo se denomina pitch y define la inclinación del objeto respecto al eje y.
3. **Rotación alrededor del eje x (roll, φ):** La tercera rotación se realiza alrededor del eje x del sistema de coordenadas, que ha sido rotado por las dos rotaciones anteriores. Este ángulo se denomina roll y define la rotación del objeto respecto al eje x.

Las rotaciones mediante ángulos de Euler se pueden representar utilizando matrices de rotación. Para la convención "zyx" [26], las matrices de rotación individuales son:

- Rotación alrededor del eje z:

$$R_z(\psi) = \begin{pmatrix} \cos(\psi) & \text{sen}(\psi) & 0 \\ -\text{sen}(\psi) & \cos(\psi) & 0 \\ 0 & 0 & 1 \end{pmatrix} \quad (1)$$

- Rotación alrededor del eje y:

$$R_y(\theta) = \begin{pmatrix} \cos(\theta) & 0 & -\text{sen}(\theta) \\ 0 & 1 & 0 \\ \text{sen}(\theta) & 0 & \cos(\theta) \end{pmatrix} \quad (2)$$

- Rotación alrededor del eje x:

$$R_x(\varphi) = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos(\varphi) & \text{sen}(\varphi) \\ 0 & -\text{sen}(\varphi) & \cos(\varphi) \end{pmatrix} \quad (3)$$

La matriz de rotación total R que describe la orientación del objeto es el producto de las tres matrices individuales:

$$\begin{aligned}
 R &= R_z(\psi)R_y(\theta)R_x(\varphi) = \tag{4} \\
 &= \begin{pmatrix} \cos(\psi) & \text{sen}(\psi) & 0 \\ -\text{sen}(\psi) & \cos(\psi) & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} \cos(\theta) & 0 & -\text{sen}(\theta) \\ 0 & 1 & 0 \\ \text{sen}(\theta) & 0 & \cos(\theta) \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & \cos(\varphi) & \text{sen}(\varphi) \\ 0 & -\text{sen}(\varphi) & \cos(\varphi) \end{pmatrix} = \\
 &= \begin{pmatrix} \cos(\psi) \cos(\theta) & \cos(\psi) \text{sen}(\theta) \text{sen}(\varphi) - \text{sen}(\psi) \cos(\varphi) & \cos(\psi) \text{sen}(\theta) \cos(\varphi) + \text{sen}(\psi) \text{sen}(\varphi) \\ \text{sen}(\psi) \cos(\theta) & \text{sen}(\psi) \text{sen}(\theta) \text{sen}(\varphi) + \cos(\psi) \cos(\varphi) & \text{sen}(\psi) \text{sen}(\theta) \cos(\varphi) - \cos(\psi) \text{sen}(\varphi) \\ -\text{sen}(\theta) & \cos(\theta) \text{sen}(\varphi) & \cos(\theta) \cos(\varphi) \end{pmatrix}
 \end{aligned}$$

Sin embargo, un problema conocido como el bloqueo de Cardan puede surgir cuando se utilizan ángulos de Euler para representar la orientación [27]. Este fenómeno ocurre cuando dos de los tres ejes de rotación se alinean, causando una pérdida de un grado de libertad en la rotación y, por lo tanto, imposibilitando describir ciertos movimientos. El bloqueo de Cardan se presenta específicamente cuando el ángulo de pitch es de ± 90 grados, alineando las rotaciones alrededor de los ejes x , y y z y haciendo indistinguible una rotación alrededor del eje z de una rotación alrededor del eje x (ver Figura 4).

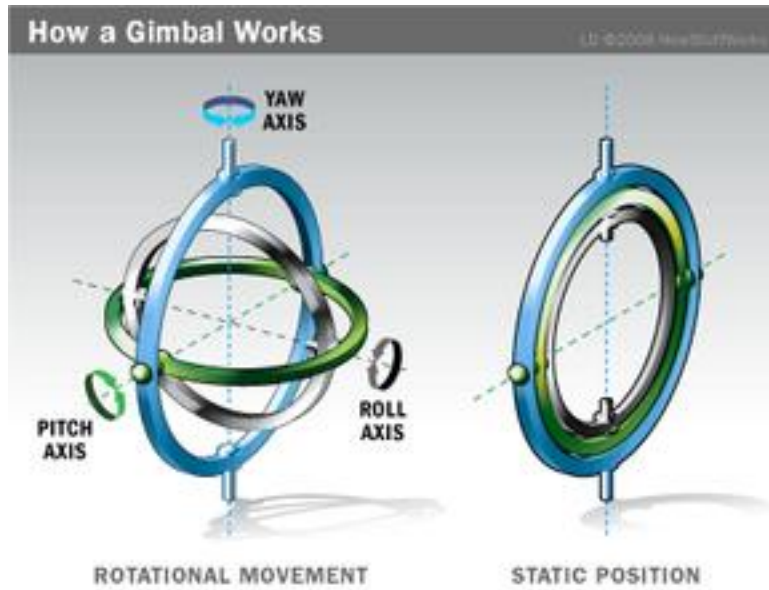


Figura 4: Representación gráfica del bloqueo de Cardán [27].

Para mitigar este problema, se pueden utilizar representaciones alternativas de la orientación, como los cuaterniones o las matrices de rotación, que no sufren de este fenómeno y proporcionan una representación continua y sin singularidades de la orientación.

2.4. Cuaterniones

Los cuaterniones son una extensión matemática de los números complejos y se utilizan para representar rotaciones en el espacio tridimensional [28]. Introducidos por William Rowan Hamilton en 1843, los cuaterniones han encontrado aplicaciones significativas en áreas como la robótica, la navegación y la computación gráfica debido a sus propiedades únicas que evitan problemas comunes asociados con otras representaciones de rotación, como los ángulos de Euler y las matrices de rotación [29].

Un cuaternión se puede expresar en la forma:

$$\mathbf{q} = w + xi + yj + zk \quad (5)$$

donde w , x , y y z son números reales, mientras que \mathbf{i} , \mathbf{j} y \mathbf{k} son las unidades imaginarias que cumplen con las siguientes relaciones [30]:

$$\mathbf{i}^2 = \mathbf{j}^2 = \mathbf{k}^2 = \mathbf{ijk} = -1 \quad (5)$$

Estas unidades imaginarias permiten describir rotaciones en un espacio tridimensional de manera más compacta y sin ambigüedades. Un cuaternión puede también representarse como un par ordenado (w, \mathbf{v}) , donde $\mathbf{v} = (x, y, z)$ es un vector tridimensional. Esta representación es útil para separar la parte escalar w de la parte vectorial \mathbf{v} .

Para aplicar una rotación a un vector tridimensional \mathbf{p} utilizando un cuaternión unitario q , se sigue el siguiente procedimiento:

1. Se representa el vector \mathbf{p} como un cuaternión puro $\mathbf{p} = (0, \mathbf{p})$.
2. Se calcula el cuaternión conjugado de q , denotado como q^*
3. La rotación del vector \mathbf{p} se obtiene mediante la operación de cuaterniones $\mathbf{p}' = q\mathbf{p}q^*$

Donde q y q^* son:

$$q = \cos(q/2) + \mathbf{u}\sin(q/2) \quad (6)$$

$$q^* = \cos(q/2) - \mathbf{u}\sin(q/2) \quad (7)$$

Aquí, θ es el ángulo de rotación y \mathbf{u} es el vector unitario alrededor del cual se realiza la rotación. En la Figura 5, se representa un cuaternión, donde en este caso el vector unitario queda definido por \mathbf{v} .

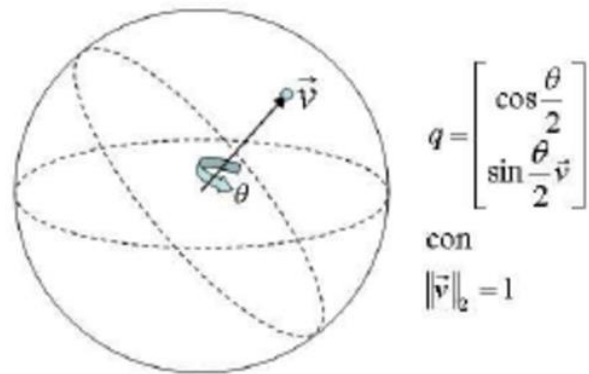


Figura 5: Representación de un cuaternión [28].

Los cuaterniones presentan varias ventajas frente a otras formas de representar rotaciones. En primer lugar, respecto a su eficiencia computacional, ya que las operaciones con cuaterniones son más eficientes y menos propensas a errores numéricos que las operaciones con matrices de rotación. Por otro lado, evitan las singularidades, ya que, a diferencia de los ángulos de Euler, no sufren bloqueos de Cardan, permitiendo una representación continua y sin ambigüedades de cualquier rotación. De esta manera son esenciales en el contexto de la estimación de ángulos para un exoesqueleto, ya que necesitamos realizar operaciones con la mayor velocidad posible sin perder precisión.

Capítulo 3

Métodos de estimación

En el ámbito de la ingeniería, la estimación precisa del estado de un sistema dinámico es fundamental para el control y la supervisión efectivos. Esta necesidad se hace aún más crítica en aplicaciones como la rehabilitación con exoesqueletos, donde la exactitud en la estimación de ángulos y movimientos es esencial para garantizar la eficacia de la terapia. Los filtros de estimación son algoritmos que permiten inferir valores desconocidos de un sistema a partir de mediciones ruidosas e incompletas, proporcionando una representación más precisa y robusta del estado del sistema.

La estimación en sistemas dinámicos se refiere al proceso de inferir el estado interno de un sistema a lo largo del tiempo, basándose en observaciones externas que suelen estar contaminadas por ruido. En la práctica, las mediciones disponibles rara vez son perfectas. Factores como el ruido de los sensores, las perturbaciones ambientales y las limitaciones de los modelos teóricos introducen errores que deben ser gestionados adecuadamente. Los filtros de estimación abordan estos desafíos mediante algoritmos que fusionan la información del modelo del sistema con las observaciones ruidosas, proporcionando estimaciones más precisas del estado real.

Como se ha mencionado anteriormente, en el contexto de la rehabilitación con exoesqueletos, la estimación precisa de los ángulos de las articulaciones es crucial. Los exoesqueletos dependen de esta información para adaptarse a los movimientos del usuario y garantizar la sincronización adecuada.

Existen diversos tipos de filtros de estimación, cada uno con sus características y aplicaciones específicas, y aunque en este trabajo se profundizará en el Filtro Madgwick, este no se entiende sin el clásico Filtro de Kalman. Por ello se hará una explicación previa sobre el Filtro de Kalman.

3.1. Filtro de Kalman

El Filtro de Kalman (KF), introducido por Rudolf E. Kalman en 1960, es un algoritmo de estimación recursiva que proporciona una solución óptima para sistemas lineales y gaussianos [31]. Este filtro es ampliamente utilizado en diversas aplicaciones debido a su eficiencia computacional y su capacidad para producir estimaciones precisas del estado de un sistema a partir de mediciones ruidosas.

El Filtro de Kalman estima el estado de un sistema dinámico en el tiempo, combinando el modelo de proceso del sistema con las observaciones ruidosas. Basado en la teoría de estimación óptima y utilizando un enfoque bayesiano, el filtro actualiza las estimaciones del estado del sistema a medida que se reciben nuevas mediciones.

El Filtro de Kalman se basa en un modelo lineal del sistema, descrito por las siguientes ecuaciones:

- **Modelo de proceso:**

$$\mathbf{x}_k = \mathbf{A}\mathbf{x}_{k-1} + \mathbf{B}\mathbf{u}_{k-1} + \mathbf{w}_{k-1} \quad (8)$$

donde \mathbf{x}_k es el estado del sistema en el tiempo k , \mathbf{u}_{k-1} es la entrada de control, y \mathbf{w}_{k-1} es el ruido del proceso, que se asume con distribución normal de media cero y covarianza \mathbf{Q} .

- **Modelo de observación:**

$$\mathbf{z}_k = \mathbf{H}\mathbf{x}_k + \mathbf{v}_k \quad (9)$$

donde \mathbf{z}_k es la observación en el tiempo k , \mathbf{H} es la matriz de observación, y \mathbf{v}_k es el ruido de medición, que se asume con distribución normal de media cero y covarianza \mathbf{R} .

El Filtro de Kalman opera en dos etapas principales: predicción y actualización [32].

Etapas de Predicción

En la etapa de predicción, el filtro proyecta la creencia actual del estado hacia adelante en el tiempo utilizando el modelo del proceso. Esto se realiza mediante las siguientes ecuaciones:

1. Proyección del estado hacia adelante:

$$\bar{\mathbf{x}}_t = \mathbf{A}_t \mathbf{X}_{t-1} + \mathbf{B}_t \mathbf{u}_t \quad (10)$$

2. Proyección de la covarianza del error hacia adelante:

$$\bar{\mathbf{P}}_t = \mathbf{A}_t \mathbf{P}_{t-1} \mathbf{A}_t^T + \mathbf{R}_t \quad (11)$$

Etapas de Actualización

En la etapa de actualización, el filtro ajusta la predicción del estado utilizando la nueva medición disponible. Las ecuaciones de esta etapa son:

3. Cómputo de la ganancia de Kalman:

$$\mathbf{K}_t = \bar{\mathbf{P}}_t \mathbf{C}_t^T (\mathbf{C}_t \bar{\mathbf{P}}_t \mathbf{C}_t^T + \mathbf{Q}_t)^{-1} \quad (12)$$

4. Actualización del estado con la medida (\mathbf{z}_t):

$$\mathbf{x}_t = \bar{\mathbf{x}}_t + \mathbf{K}_t (\mathbf{z}_t - \mathbf{C}_t \bar{\mathbf{x}}_t) \quad (13)$$

5. Actualización de la covarianza del error:

$$\mathbf{P}_t = (\mathbf{I} - \mathbf{K}_t \mathbf{C}_t) \bar{\mathbf{P}}_t \quad (14)$$

6. Devuelve $\mathbf{x}_t, \mathbf{P}_t$

donde:

- \mathbf{A}_t : Matriz $n \times n$ que relaciona el estado en el instante $t - 1$ con el estado en el instante t , en ausencia de señales de control.

- \mathbf{B}_t : Matriz $n \times l$ que relaciona las señales de control, opcionales, con el estado actual.
- \mathbf{R}_t : Matriz $n \times n$ que representa a la covarianza del ruido del proceso.
- \mathbf{C}_t : Matriz $m \times n$ que relaciona el estado actual con las observaciones del entorno.
- \mathbf{Q}_t : Matriz $n \times n$ que representa a la covarianza del ruido de las observaciones.
- \mathbf{K}_t : Matriz $n \times m$ que representa a la ganancia de Kalman.

La ganancia de Kalman \mathbf{K}_t ajusta la importancia de las observaciones frente a las predicciones del modelo. Si las observaciones son precisas y confiables, \mathbf{K}_t aumenta, dando más peso a las nuevas mediciones. Por el contrario, si las observaciones son ruidosas, \mathbf{K}_t disminuye, confiando más en el modelo de predicción. Se puede ver resumido el KF en un diagrama en la Figura 6.

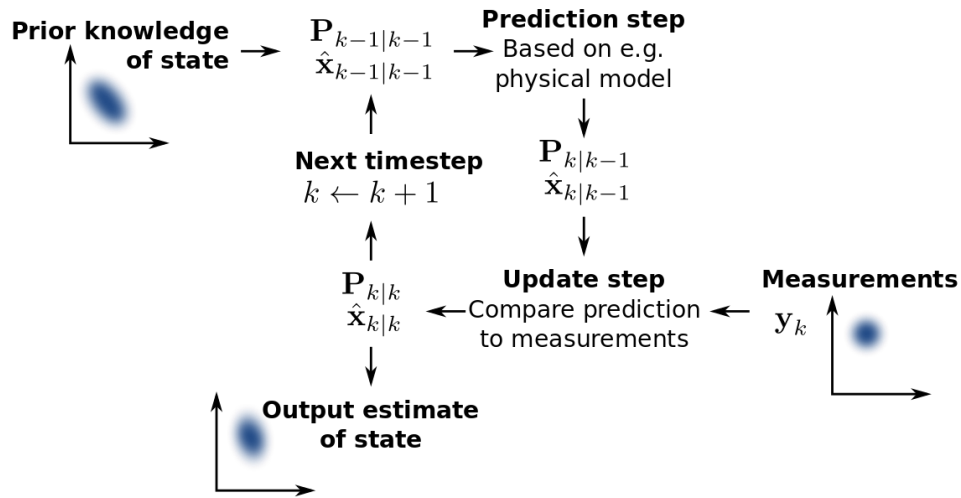


Figura 6: Diagrama del algoritmo del Filtro de Kalman [33].

3.1.1. Filtro de Kalman Extendido (EFK)

El Filtro de Kalman Extendido (EKF) es una variante del Filtro de Kalman diseñada para manejar sistemas no lineales. Mientras que el Filtro de Kalman original se aplica a sistemas lineales, el EKF extiende su aplicabilidad a sistemas donde las relaciones entre variables no pueden ser expresadas

linealmente [34]. El EKF logra esto mediante la linealización de las ecuaciones del sistema alrededor de la estimación actual del estado.

El EKF se basa en la misma idea fundamental que el Filtro de Kalman: la combinación óptima de predicciones del modelo y mediciones observadas. Sin embargo, en lugar de suponer relaciones lineales, el EKF utiliza aproximaciones lineales locales de las funciones no lineales que describen el sistema.

En el contexto del EKF, los modelos de proceso y observación se representan generalmente como funciones no lineales [32]:

- **Modelo de proceso:**

$$\mathbf{x}_k = \mathbf{f}(\mathbf{x}_{k-1}, \mathbf{u}_{k-1}) + \mathbf{w}_{k-1} \quad (15)$$

donde \mathbf{f} es una función no lineal que describe la evolución del estado del sistema.

- **Modelo de observación:**

$$\mathbf{z}_k = \mathbf{h}(\mathbf{x}_k) + \mathbf{v}_k \quad (16)$$

donde \mathbf{h} es una función no lineal que relaciona el estado del sistema con las observaciones.

Al igual que el Filtro de Kalman, el EKF opera en dos etapas: predicción y actualización. Sin embargo, estas etapas implican la linealización de las funciones no lineales utilizando aproximaciones de primer orden (expansión de Taylor).

También existen distintas publicaciones en las que se implementa el uso de cuaterniones en el EKF ³⁵En lugar de usar ángulos de Euler o matrices de rotación, los cuaterniones se utilizan para representar la orientación del sistema. La evolución del cuaternión se puede modelar utilizando las tasas de giro medidas por los giroscopios. Las mediciones del acelerómetro y el magnetómetro se pueden relacionar con la orientación estimada del sistema utilizando cuaterniones. Durante la etapa de predicción, el estado del cuaternión se actualiza utilizando el modelo de proceso, y en la etapa de

actualización, el EKF ajusta la estimación del cuaternión utilizando las mediciones observadas y la matriz de covarianza del error (ver Figura 7).

Por ejemplo, el cuaternión \mathbf{q} será el vector de estado, y la velocidad angular $\boldsymbol{\omega}$ (en radianes por segundo) será el vector de control:

$$\mathbf{x} \triangleq \mathbf{q} = [\mathbf{q}_w \ \mathbf{q}_v]^T = [\mathbf{q}_w \ \mathbf{q}_x \ \mathbf{q}_y \ \mathbf{q}_z]^T \quad (17)$$

$$\mathbf{u} \triangleq \boldsymbol{\omega} = [\boldsymbol{\omega}_x \ \boldsymbol{\omega}_y \ \boldsymbol{\omega}_z]^T \quad (18)$$

La etapa de predicción sería:

$$\hat{\mathbf{q}}_t = \mathbf{q}_{t-1} + \int_{t-1}^t \boldsymbol{\omega} \, dt \quad (19)$$

La etapa de actualización:

$$\mathbf{q}_t = \hat{\mathbf{q}}_t + \mathbf{K}_t(\mathbf{z}_t - \mathbf{h}(\mathbf{q}_t)) \quad (20)$$

Donde

- \mathbf{z}_t : es la medida actual
- \mathbf{h} : es medida predicha.
- \mathbf{K}_t : la ganancia de Kalman.

En cuanto a la linealización [36], los métodos de linealización polinomial de orden n se pueden construir a partir de series de Taylor de $q(t_n + \Delta t)$ alrededor del tiempo $t = t_n$:

$$\mathbf{q}_t = \mathbf{q}_{t-1} + \dot{\mathbf{q}}_{t-1} \Delta t + \frac{1}{2!} \ddot{\mathbf{q}}_{t-1} \Delta t^2 + \frac{1}{3!} \dddot{\mathbf{q}}_{t-1} \Delta t^3 + \dots \quad (21)$$

Donde $\dot{\mathbf{q}}$ es la primera derivada del cuaternión.

Por lo tanto, se utiliza para sistemas no lineales, permitiendo estimaciones en situaciones donde las relaciones entre las variables no son lineales o donde las distribuciones de error no son gaussianas. El

EKF aproxima las no linealidades a través de una serie de expansiones de Taylor, lo que lo hace más flexible en comparación con el Filtro de Kalman estándar. Sin embargo, el EKF puede ser más computacionalmente muy intensivo, provocando que, para cálculos en tiempo real, en los que se requiere mucha inmediatez, no sea óptimo. Además, puede arrastrar ruido en el proceso de linealización. Por ello puede encontrar desafíos en trabajos como los necesarios con un exosqueleto para rehabilitación.

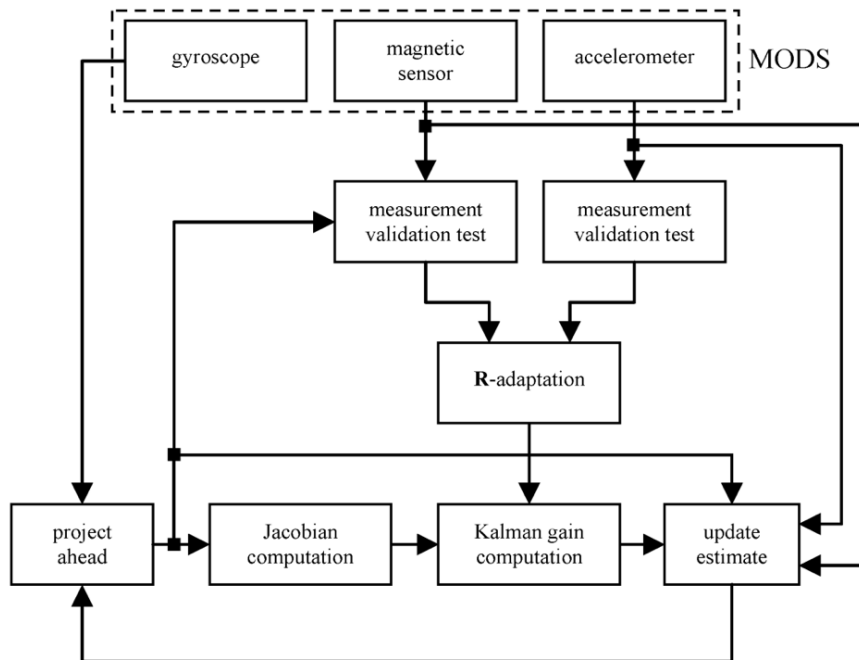


Figura 7: Diagrama del algoritmo del Filtro de Kalman Extendido [35].

3.1.2. Otros Filtros de Kalman

Además del Filtro de Kalman estándar y el Filtro de Kalman Extendido, existen varias otras variantes diseñadas para abordar diferentes desafíos en la estimación de estados en sistemas dinámicos. A continuación, se presentan algunos de estos filtros, resaltando sus características y aplicaciones particulares.

3.1.2.1. Error-state Kalman Filter (ErKF)

El Filtro de Kalman en el espacio de error (ErKF) es una variante del EKF que opera en el espacio de "error de estado" o "estado extendido". En lugar de linealizar el modelo en cada iteración, el ErKF trabaja con las diferencias entre el estado real y la estimación prevista. Este enfoque permite propagar el error de estado a través del modelo no lineal, manteniendo una representación más precisa de la no linealidad en comparación con la linealización típica del EKF. Aunque el ErKF puede ofrecer una mejor estabilidad y precisión en entornos altamente no lineales, requiere un mayor esfuerzo computacional debido a la manipulación del espacio de error [37, 38].

3.1.2.2. Filtro Kalman Unscented (UKF)

El Filtro Kalman Unscented (UKF) es una extensión del EKF que emplea la técnica de transformación *unscented* para propagar las distribuciones de probabilidad a través del sistema no lineal. A diferencia del EKF, que utiliza expansiones de Taylor para aproximar las no linealidades, el UKF utiliza un conjunto de puntos de muestreo (puntos sigma) que capturan la estructura de la distribución con mayor precisión. Esta técnica permite que el UKF sea más preciso que el EKF en sistemas altamente no lineales y, a menudo, requiere menos cálculos, mejorando la eficiencia computacional [39, 40].

3.1.2.3. Filtro Kalman de Rango Variable (VKF)

El Filtro Kalman de Rango Variable (VKF) es una variante diseñada para adaptarse a entornos donde la varianza del proceso o las mediciones varían con el tiempo. A diferencia del Filtro Kalman estándar, que asume constantes de varianza, el VKF permite que las varianzas se ajusten dinámicamente a las condiciones cambiantes del sistema. Esta adaptabilidad lo hace más robusto en situaciones donde la incertidumbre varía considerablemente a lo largo del tiempo, proporcionando estimaciones más precisas en entornos variables [41].

3.1.2.4. Filtro Kalman Complementario (CKF)

El Filtro Kalman Complementario (CKF) es una extensión del Filtro de Kalman estándar utilizada para la fusión de datos provenientes de múltiples fuentes. Este filtro combina dos filtros de Kalman

estándar en paralelo: uno optimizado para estimar el estado de baja frecuencia y otro para alta frecuencia. Luego, utiliza un proceso de combinación para obtener una estimación más precisa y robusta. El CKF es particularmente útil en aplicaciones donde se dispone de múltiples fuentes de datos que proporcionan información complementaria sobre el estado del sistema [42].

3.2. Filtro de Madgwick

El Filtro de Madgwick es una técnica iterativa para la estimación de la orientación tridimensional de un objeto en movimiento, utilizando datos de sensores inerciales como giroscopios y acelerómetros. Su principal objetivo es calcular la rotación del objeto respecto a un sistema de referencia fijo. A continuación, se describirán detalladamente los componentes y el funcionamiento del filtro de Madgwick, integrando conceptos previamente mencionados como los cuaterniones y la fusión de datos [43].

El Filtro de Madgwick se basa en una formulación matemática que integra las tasas de giro provenientes de los giroscopios y las aceleraciones lineales medidas por los acelerómetros. Este filtro también puede incorporar datos de magnetómetros para mejorar la precisión en presencia de campos magnéticos, aunque su uso es opcional. La integración de estos datos permite calcular la orientación del objeto en el espacio tridimensional de manera eficiente.

Como se explicó anteriormente, los cuaterniones son una herramienta matemática que extiende los números complejos y se utilizan para representar rotaciones en el espacio tridimensional sin las singularidades y ambigüedades asociadas con los ángulos de Euler. En el Capítulo 2 se indicó que un cuaternión se puede expresar en la forma $\mathbf{q} = w + x\mathbf{i} + y\mathbf{j} + z\mathbf{k}$, donde w es la parte escalar y x, y, z son las partes vectoriales.

Las medidas de los giroscopios, los acelerómetros y magnetómetros se combinan para obtener una estimación más precisa de la orientación. Cada uno desempeña una función dentro del proceso:

1. **Giroscopios:** sus datos se utilizan para actualizar continuamente el cuaternión de orientación mediante la integración de las tasas de cambio angular.
2. **Acelerómetros:** se utiliza para corregir los errores acumulativos de los giroscopios, especialmente la deriva a largo plazo.
3. **Magnetómetros:** corrige la orientación, utilizando la dirección del campo magnético terrestre como referencia. Es importante mencionar que es opcional, aunque su uso de manera adecuada mejora sustancialmente los resultados. Esto se verá en el Capítulo 5.

La combinación de estos datos permite mitigar la deriva acumulativa de los giroscopios y mantener la estimación de la orientación alineada con la gravedad, resultando en una estimación más precisa y estable a lo largo del tiempo.

El Filtro de Madgwick utiliza dos parámetros de control de ganancia conocidos como beta (β) y zeta (ζ), cada uno con un papel específico en el ajuste del comportamiento del filtro [44].

Por un lado, beta controla la velocidad a la que el Filtro de Madgwick converge hacia la solución. Un valor mayor de β resultará en una convergencia más rápida, pero podría introducir más ruido. Un valor menor de β dará una convergencia más lenta (mayor cantidad de iteraciones), aunque más suave y robusto frente al ruido. Generalmente, el valor de β se ajusta experimentalmente para adaptarse a las características específicas del sistema y las condiciones de operación.

Por otro lado, zeta controla la cantidad de "corrección" aplicada al giroscopio en el cálculo de la orientación. Un valor mayor de ζ dará más peso a las mediciones del giroscopio, lo que puede hacer que la salida sea más suave pero potencialmente menos precisa. Un valor menor de ζ dará más peso a las mediciones del acelerómetro y el magnetómetro, lo que puede hacer que la salida sea más sensible a la aceleración y la magnetización terrestre, pero potencialmente más precisa en condiciones estáticas.

La implementación del Filtro de Madgwick implica la integración de los datos de los sensores inerciales para calcular el cuaternión de orientación. El algoritmo de Madgwick utiliza una serie de ecuaciones diferenciales para actualizar los cuaterniones en función de las tasas de cambio angular y las aceleraciones medidas. Además, si se utilizan magnetómetros, se incorporan datos del campo

magnético para corregir la orientación. De forma esquemática se puede ver representado en la Figura 8.

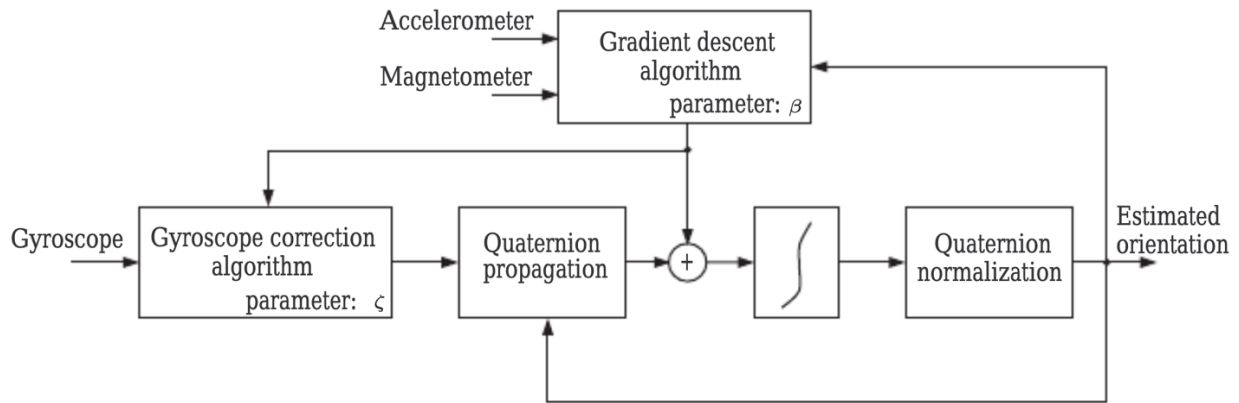


Figura 8: Diagrama del algoritmo del Filtro de Madgwick [45].

Para trabajar con el Filtro de Madgwick se empleó una versión desarrollada en Python creada por Jonas Böer y que se puede encontrar en el repositorio GitHub [46]. El código está adaptado tanto para IMUs con magnetómetro y sin él, pudiendo escoger entre dos funciones en el caso que convenga. El filtro está compuesto por dos archivos. El primero, MadgwickAHRS.py, contiene la implementación del filtro de Madgwick, utilizado para la estimación de la orientación tridimensional de un objeto en movimiento. Este archivo configura los parámetros iniciales del filtro, como el período de muestreo, el cuaternión inicial y los parámetros de ganancia beta (β) y zeta (ζ).

El filtro calcula la derivada del cuaternión utilizando las tasas de cambio angular del giroscopio y corrige el error de orientación basándose en los datos del acelerómetro y opcionalmente del magnetómetro. Después, el cuaternión se actualiza integrando la derivada calculada y aplicando las correcciones necesarias, seguido de una normalización para mantener su magnitud. Además, el archivo implementa métodos de integración para actualizar los estados del filtro en función de las lecturas de los sensores inerciales.

Por otro lado, el archivo Quaternion.py define una clase para manejar las operaciones con cuaterniones, fundamentales para el cálculo de rotaciones en tres dimensiones. Este archivo proporciona una función para inicializar cuaterniones con valores específicos y define operaciones aritméticas básicas como la multiplicación de cuaterniones. También incluye métodos para normalizar cuaterniones y asegurar que mantengan una magnitud de uno, además de funciones para convertir cuaterniones a otras representaciones de rotación, como matrices de rotación o ángulos de Euler. Además, contiene funciones auxiliares para facilitar las operaciones con cuaterniones, como el cálculo del conjugado y la magnitud de un cuaternión.

Ahora vamos a ver en detalle el funcionamiento del Filtro de Madgwick y las matemáticas empleadas en el proceso. Este se puede dividir en 5 fases.

1. Inicialización

El filtro comienza con la inicialización de los parámetros principales: el período de muestreo, el cuaternión inicial y los parámetros de ganancia beta (β) y zeta (ζ). Tanto beta como zeta deben estar normalizados, siendo valores entre 0 y 1.

2. Normalización de mediciones de acelerómetro y magnetómetro

Primero, las mediciones del acelerómetro y el magnetómetro se normalizan para evitar cualquier error debido a la magnitud variable de los datos. La normalización es esencial porque asegura que las mediciones se comparen correctamente con la orientación estimada, eliminando las diferencias de escala y permitiendo una fusión de datos precisa.

La normalización de un vector \mathbf{a} (acelerómetro) y un vector \mathbf{m} (magnetómetro) se realiza dividiendo cada componente del vector por su norma:

$$\mathbf{a}_{norm} = \frac{\mathbf{a}}{\|\mathbf{a}\|} \quad (22)$$

$$\mathbf{m}_{norm} = \frac{\mathbf{m}}{\|\mathbf{m}\|} \quad (23)$$

Donde $\| a \|$ y $\| m \|$ representan la magnitud de los vectores magnético a y m , respectivamente, calculada como:

$$\| a \| = \sqrt{a_x^2 + a_y^2 + a_z^2} \quad (24)$$

$$\| m \| = \sqrt{m_x^2 + m_y^2 + m_z^2} \quad (25)$$

Este paso es importante porque cualquier error en la magnitud de las mediciones podría llevar a errores significativos en la orientación estimada. La normalización asegura que todas las mediciones se utilicen en la misma escala.

3. Cálculo del vector de referencia del campo

Una vez normalizadas las mediciones del magnetómetro, se calcula el vector de referencia del campo magnético utilizando el cuaternión actual y las mediciones del magnetómetro. Este vector de referencia es esencial para definir la orientación del campo magnético en el marco de referencia de la tierra, lo cual es necesario para una correcta estimación de la orientación.

El vector de referencia del campo magnético h se calcula como:

$$h = q \otimes (0, m_x, m_y, m_z) \otimes q^* \quad (26)$$

Donde:

- q es el cuaternión actual.
- \otimes denota la multiplicación de cuaterniones.
- $(0, m_x, m_y, m_z)$ es el cuaternión que representa el vector del magnetómetro.
- q^* es el cuaternión conjugado de q .

Este cálculo transforma el vector del magnetómetro del marco de referencia del sensor al marco de referencia global utilizando la orientación actual del sensor representada por el cuaternión q .

A continuación, se extrae el vector b que representa el campo magnético proyectado en el marco de referencia global:

$$b = (0, \|h_{1:3}\|, 0, h_3) \quad (27)$$

Donde $\|h_{1:3}\|$ representa la magnitud del vector h en los ejes x e y , y h_3 es el componente en el eje z . Este vector de referencia es fundamental para la corrección del error de orientación utilizando las mediciones del magnetómetro.

4. Cálculo del error de orientación

El error de orientación se define como la diferencia entre la orientación estimada por el cuaternión y la orientación medida por los sensores inerciales (acelerómetro y magnetómetro). Este error se calcula para ajustar la orientación estimada y mejorar su precisión.

El vector de error f se define como:

$$f = \begin{bmatrix} 2(q_1q_3 - q_0q_2) - a_x \\ 2(q_0q_1 + q_2q_3) - a_y \\ 2(0.5 - q_1^2 - q_2^2) - a_x \\ 2b_1(0.5 - q_2^2 - q_3^2) + 2b_3(q_1q_3 - q_0q_2) - m_x \\ 2b_1(q_1q_2 - q_0q_3) + 2b_3(q_0q_1 + q_2q_3) - m_y \\ 2b_1(q_0q_2 + q_1q_3) + 2b_32b_1(0.5 - q_1^2 - q_2^2) - m_z \end{bmatrix} \quad (28)$$

Cada componente de este vector representa una diferencia entre una medición y la estimación correspondiente de la orientación. Los primeros tres términos representan la diferencia entre la gravedad medida (por el acelerómetro) y la gravedad estimada basada en el cuaternión. Los tres términos restantes representan la diferencia entre el campo magnético medido y el campo magnético estimado.

Este error se utiliza para ajustar la orientación estimada de manera que minimice las diferencias entre las mediciones reales y las estimaciones del modelo, mejorando así la precisión de la orientación estimada.

5. Cálculo del Jacobiano

Se calcula el Jacobiano de la función de error con respecto al cuaternión. El Jacobiano representa las derivadas parciales de la función de error con respecto a cada componente del cuaternión, y es crucial para el algoritmo de descenso de gradiente porque indica cómo cambia el error cuando se ajustan los componentes del cuaternión.

El Jacobiano se define como:

$$J = \begin{bmatrix} -2q_2 & 2q_3 & -2q_0 & 2q_1 \\ 2q_1 & 2q_0 & 2q_3 & 2q_2 \\ 0 & -4q_1 & -4q_2 & 0 \\ -2b_3q_2 & 2b_3q_3 & -4b_1q_2 - 2b_3q_0 & -4b_1q_3 + 2b_3q_1 \\ -2b_1q_3 + 2b_3q_1 & 2b_1q_2 + 2b_3q_0 & 2b_1q_1 + 2b_3q_3 & -2b_1q_0 + 2b_3q_2 \\ 2b_1q_2 & 2b_1q_3 - 4b_3q_1 & 2b_1q_0 - 4b_3q_2 & 2b_1q_1 \end{bmatrix} \quad (29)$$

Cada elemento de esta matriz representa la tasa de cambio de un componente del error con respecto a un componente del cuaternión. Por ejemplo, el elemento $-2q_2$ en la primera fila y primera columna indica cómo cambia el primer componente del error (relacionado con la gravedad en el eje x) cuando se ajusta el segundo componente del cuaternión (q_2).

Este Jacobiano se utiliza en el siguiente paso para calcular el gradiente del error y ajustar el cuaternión de manera que se minimice el error.

6. Cálculo del paso de corrección

El paso de corrección se calcula como el producto del Jacobiano transpuesto y la función de error. Este paso se normaliza para asegurar que la magnitud del ajuste sea consistente, lo que es esencial para

evitar que el algoritmo haga correcciones demasiado grandes que podrían desestabilizar la estimación de la orientación.

El paso de corrección se define como:

$$step = J^T f \quad (30)$$

Donde

- J^T es la transpuesta del Jacobiano calculado anteriormente.
- f es el vector de error de orientación.

Después de calcular el paso, se normaliza para asegurar que tenga una magnitud constante:

$$step_{norm} = \frac{step}{\|step\|} \quad (31)$$

La normalización del paso de corrección es importante para que el ajuste aplicado a la orientación estimada sea controlado y no dependa de la magnitud del error. Esto ayuda a mantener la estabilidad del algoritmo y evita ajustes excesivos que podrían introducir más errores.

7. Compensación de deriva del giroscopio

La deriva del giroscopio se compensa ajustando el cuaternión del giroscopio utilizando el paso de corrección y el factor de ganancia ζ . Este ajuste se hace para corregir cualquier deriva que pueda haberse acumulado en la estimación de la orientación debido a errores en las mediciones del giroscopio.

Primero, se define el cuaternión del giroscopio:

$$gyroscopeQuat = (0, \omega_x, \omega_y, \omega_z) \quad (32)$$

Donde ω_x , ω_y y ω_z son las tasas de giro medidas por el giroscopio en los ejes x , y y z , respectivamente.

Luego, se define el cuaternión de paso de corrección:

$$stepQuat = (step[0], step[1], step[2], step[3]) \quad (33)$$

La corrección se aplica al cuaternión del giroscopio de la siguiente manera:

$$gyroscopeQuat = gyroscopeQuat + (q^* \otimes stepQuat) \cdot 2 \cdot samplePeriod \cdot \zeta \cdot (-1) \quad (34)$$

Donde:

- q^* es el cuaternión conjugado.
- \otimes representa la multiplicación de cuaterniones.
- `samplePeriod` es el periodo de muestreo.
- ζ es el factor de ganancia que controla la cantidad de corrección aplicada para compensar la deriva del giroscopio.

Esta corrección ayuda a mantener la precisión de la orientación estimada al compensar cualquier error acumulado en las mediciones del giroscopio.

8. Cálculo de la tasa de cambio del cuaternión

La tasa de cambio del cuaternión se calcula utilizando la tasa de cambio de orientación medida por los giroscopios y la corrección del error. Esto se hace para actualizar la orientación estimada de manera que refleje tanto las mediciones actuales del giroscopio como las correcciones basadas en el error de orientación.

La tasa de cambio del cuaternión se define como:

$$\dot{q} = \frac{1}{2} q \otimes gyroscopeQuat - \beta \cdot step_{norm} \quad (35)$$

Donde:

- q^* es el cuaternión conjugado.
- `gyroscopeQuat` es el cuaternión del giroscopio corregido calculado anteriormente.
- β es el factor de ganancia que controla la rapidez de convergencia del filtro.

Este cálculo combina la información de las mediciones del giroscopio con las correcciones necesarias para minimizar el error de orientación, proporcionando una estimación precisa de la tasa de cambio de orientación.

9. Integración y normalización del cuaternión

Finalmente, el cuaternión se actualiza integrando la tasa de cambio calculada y normalizando el resultado para mantener la magnitud del cuaternión igual a uno. Este paso asegura que el cuaternión actualizado represente una rotación válida en el espacio tridimensional.

La actualización del cuaternión se define como:

$$q = q + \dot{q} \cdot \text{samplePeriod} \quad (36)$$

Después de actualizar el cuaternión, se normaliza para mantener su magnitud igual a uno:

$$q_{norm} = \frac{q}{\|q\|} \quad (37)$$

La normalización es crucial para asegurar que el cuaternión siga siendo una representación válida de la orientación en el espacio tridimensional. Esto evita que los errores numéricos acumulativos afecten la precisión de la estimación de la orientación.

3.3. Ventajas del Filtro de Madgwick frente al Filtro de Kalman Extendido

A continuación, se detallan algunas de las teóricas ventajas del Filtro de Madgwick frente al Filtro de Kalman, sobre todo al Filtro de Kalman Extendido, junto con una discusión sobre el uso de cuaterniones en ambos filtros.

1. Simplicidad y eficiencia computacional

Una de las principales ventajas del Filtro de Madgwick es su simplicidad y eficiencia computacional. El algoritmo de Madgwick está diseñado específicamente para aplicaciones de fusión de sensores en tiempo real y se caracteriza por tener menos requerimientos computacionales en comparación con el EKF. Esto se debe a que el MF utiliza una serie de ecuaciones diferenciales directas y actualizaciones basadas en tasas de cambio angular y mediciones de aceleración, evitando la complejidad de las derivadas parciales y la linearización que son necesarias en el EKF [47].

2. Implementación directa con cuaterniones

El filtro de Madgwick utiliza cuaterniones de forma nativa para representar la orientación, lo que simplifica significativamente las operaciones matemáticas y evita los problemas de singularidad y bloqueo de Cardan asociados con los ángulos de Euler. Aunque el EKF también puede utilizar cuaterniones para representar la orientación, su implementación es más compleja debido a la necesidad de manejar las derivadas parciales de las funciones de estado y observación, además de realizar la linearización en cada iteración del filtro [44].

3. Menor dependencia de un modelo de sistema preciso

El EKF requiere un modelo preciso del sistema dinámico y las relaciones de observación para funcionar correctamente. Cualquier error en estos modelos puede llevar a una degradación significativa en el rendimiento del filtro. En contraste, el Filtro de Madgwick es menos dependiente de un modelo preciso del sistema, ya que se basa principalmente en la integración de las tasas de

cambio angular y las correcciones de orientación basadas en acelerómetros y magnetómetros. Esto lo hace más sólido en escenarios donde el modelo del sistema es difícil de definir con precisión.

4. Robustez frente a errores de sensores

El Filtro de Madgwick incorpora mecanismos de corrección de error que utilizan datos del acelerómetro y el magnetómetro para corregir la orientación estimada. Este enfoque permite al filtro ser más robusto frente a errores de sensores y la deriva de los giroscopios. Aunque el EKF también puede manejar errores de sensores, su dependencia de un modelo preciso y la complejidad de su implementación pueden hacer que sea más susceptible a la acumulación de errores en ciertos escenarios.

5. Ajustabilidad de parámetros

El Filtro de Madgwick permite un ajuste relativamente sencillo de los parámetros de ganancia (beta y zeta), lo que facilita la optimización del filtro para diferentes aplicaciones y condiciones de operación. Este ajuste es menos intuitivo en el EKF, donde los parámetros de ganancia están implícitos en la matriz de covarianza del error y las ecuaciones de actualización del filtro, lo que puede complicar la sintonización del filtro para un rendimiento óptimo.

Capítulo 4

Bases de datos

Para poder aplicar los filtros se necesitaba una base de datos que fuese lo más parecida a una situación real con el exoesqueleto del proyecto. Para ello se necesitaba encontrar una base de datos que tuviese datos recogidos por IMUs con acelerómetro, giroscopio y magnetómetro. Además, la posición y el número de los sensores en el cuerpo debería ser lo más cercano a la situación real. Es decir, debería estar compuesto por 8 sensores IMUs, colocados uno en cada pie, tibia, muslo, además de uno en la cadera y otro en el pecho, de referencia. Asimismo, el formato de los datos debía ser específico para poder aplicar los filtros.

Por otro lado, era esencial encontrar movimientos lo más parecidos a los movimientos que podría realizar un paciente durante una sesión de rehabilitación. Por ejemplo, movimientos de andar hacia delante, estar parado o sentarse. Conocer la frecuencia de muestreo también era realmente importante para poder indicarlo en los filtros.

4.1. Primera opción: síntesis base de datos

Tras realizar una búsqueda intensiva de bases de datos con las características requeridas, no se encontró ninguna que cumpliera con todos los requisitos. La mayoría de ellas presentaban colocaciones de IMUs en zonas muy distintas a las planteadas (grabaciones de miembros del tren superior), los datos se mostraban en formatos no compatibles o los movimientos estaban muy alejados a los realizados en rehabilitación (como correr o la práctica de distintos deportes). Es por ello por lo

que se cambió el rumbo para diseñar y sintetizar una base de datos propia. La combinación de datos de sensores inerciales y modelos de cuerpo humano parametrizados, como el modelo SMPL (Skinned Multi-Person Linear Model), ofrece una herramienta para la captura y análisis del movimiento humano. El proceso de síntesis de una base de datos de aceleración y giroscopio utilizando estos datos, apoyándose en la base de datos AMASS (Archive of Motion Capture as Surface Shapes), permite la creación de una base de datos detallada, precisa y sobre todo personalizable, ya que permite grabar movimiento colocando los IMUs donde más convenga [48].

El primer paso fue la recopilación de datos. La base de datos AMASS es una colección extensa de secuencias de movimiento humano capturadas mediante diversos sistemas de captura de movimiento (MoCap). Este recurso integra datos de múltiples estudios y conjuntos de datos, unificados bajo un formato común que facilita la integración con otros modelos y herramientas de análisis, como el modelo SMPL. Por otro lado, las IMUs, que como ya se mencionó, miden la aceleración lineal y la velocidad angular en múltiples ejes, se colocan en puntos específicos del cuerpo, como las tobillos, los muslos y la cintura, para capturar movimientos precisos.

La base de datos de AMASS es muy extensa, y se compone a su vez de 24 bases de datos diferentes con todo tipo de movimientos. En este caso se usó la base de datos HumanEva, ya que está compuesta por movimientos de caminar principalmente.

El modelo SMPL es un modelo de cuerpo humano que utiliza parámetros para representar formas y poses. Este modelo permitió la parametrización de las poses mediante algoritmos de ajuste que mapearon las poses de AMASS a los parámetros de pose de SMPL. De igual manera, los parámetros de forma fueron ajustados para cada individuo basándose en las medidas capturadas. Utilizando estos parámetros, el modelo SMPL generó una malla tridimensional del cuerpo humano en cada fotograma de la secuencia, lo que permitió una representación visual y analítica detallada del movimiento.

El código utilizado se basa en el encontrado en el repositorio de GitHub [49] que está basado en el artículo [48] publicado por Huang et al. Sin embargo, este tuvo que ser modificado para que incluyese los datos de aceleración y giroscopio de todas las IMUs. Además, se automatizaron los procesos de

síntesis con diferentes scripts en Python. Más tarde, también se desarrollaron otros scripts para cambiar el formato de .npz a formato .pkl para un mejor manejo en el posterior procesado.

Para la síntesis de datos de aceleración y giroscopio se simuló la colocación de IMUs en puntos específicos de la malla generada por SMPL, como se ve en la Figura 13. La aceleración lineal fue obtenida mediante la derivada segunda de la posición de los puntos de colocación, ver fórmula (38). Por otro lado, la velocidad angular fue derivada de la rotación de los segmentos del cuerpo, calculado a partir de las rotaciones de transformación global.

$$a_t = \frac{p_{t-1} + p_{t+1} - 2p_t}{dt^2} \quad (38)$$

De esta manera, cada archivo estaba compuesto por:

- Un array para los datos de la orientación, el cual estaba compuesto por un array por cada instante. Para cada instante había 8 matrices de rotación que correspondían a cada IMU.
- En el array de aceleración había también un array para cada instante. Este último estaba compuesto por un vector con los datos en metros por segundo de cada eje por cada una de las 8 IMUs.
- Por último, un array que contenía la etiqueta del tipo de movimiento que se estaba realizando: “walk”, “stand”, “sit”, ...

Para ver la estructura de los datos de una forma más gráfica se pueden observar la Figura 9, 10 y 11.

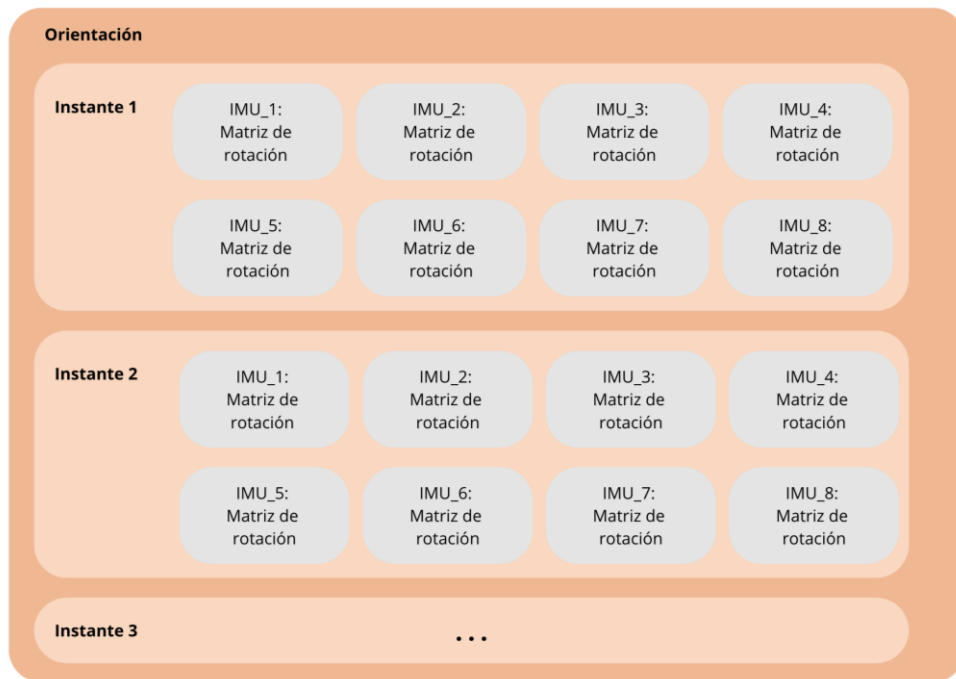


Figura 9: Estructura de los datos de orientación de la base sintetizada.

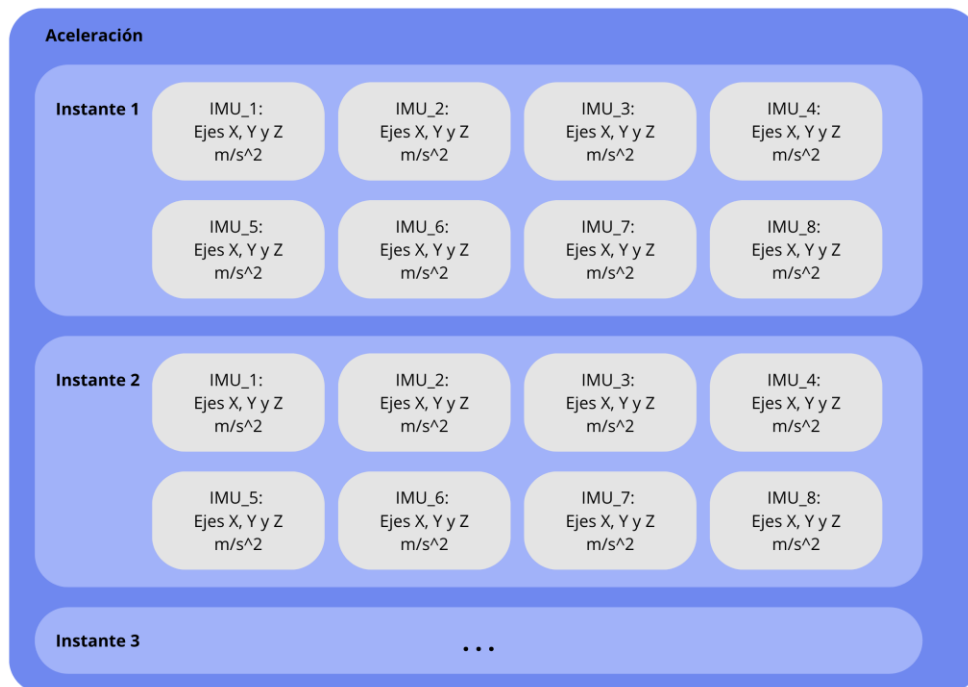


Figura 10: Estructura de los datos de aceleración de la base sintetizada.

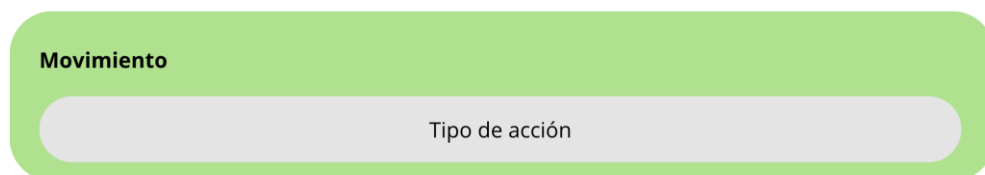


Figura 11: Estructura de los datos de tipo de movimiento de la base sintetizada.

Además, para la visualización de los datos y análisis más detallados, se utilizaron herramientas como Aitviewer. Esta herramienta permite la visualización interactiva de las mallas generadas por SMPL y las secuencias de movimiento, proporcionando una forma intuitiva de explorar y analizar los datos de movimiento humano, como se puede ver en la Figura 12. Con Aitviewer, se pueden rotar, ampliar y reproducir las secuencias de movimiento, facilitando la identificación de los datos.

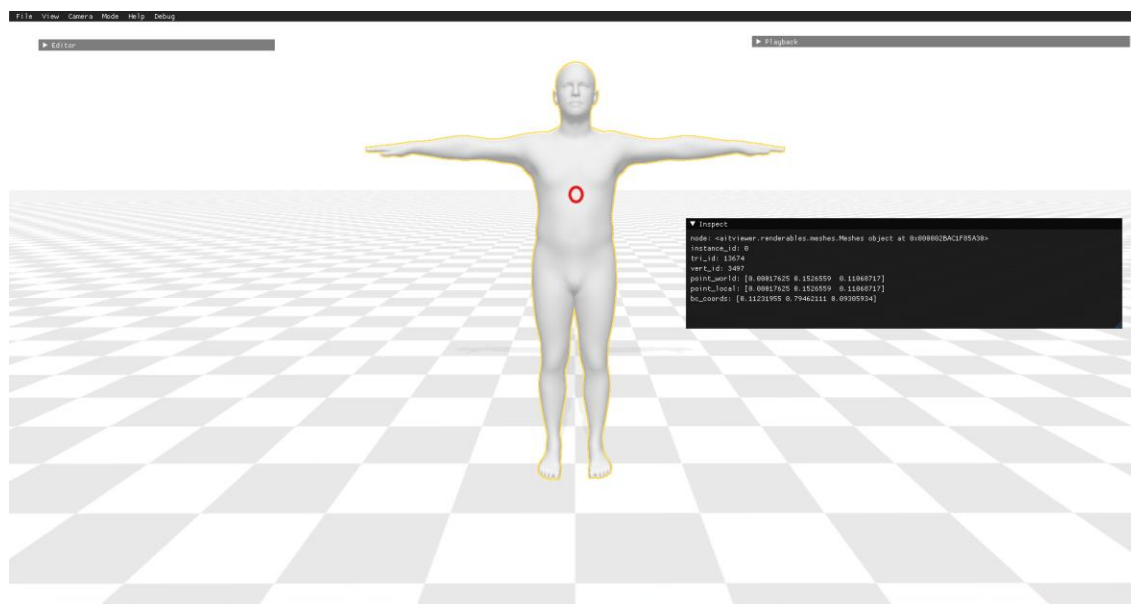


Figura 12: Uso de las herramientas de Aitviewer para la visualización de los vértices.

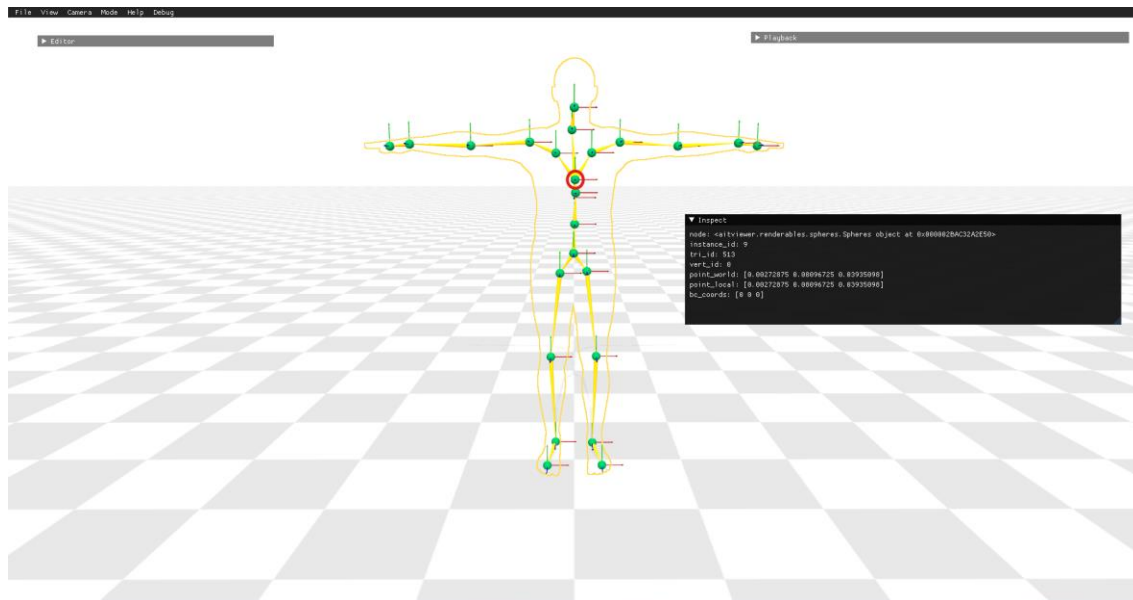


Figura 13: Uso de las herramientas de Aitviewer para la navegación sobre las articulaciones del modelo.

En la Figura 13 y 14 se muestran los lugares donde se colocaron las IMUs virtuales en el modelo.

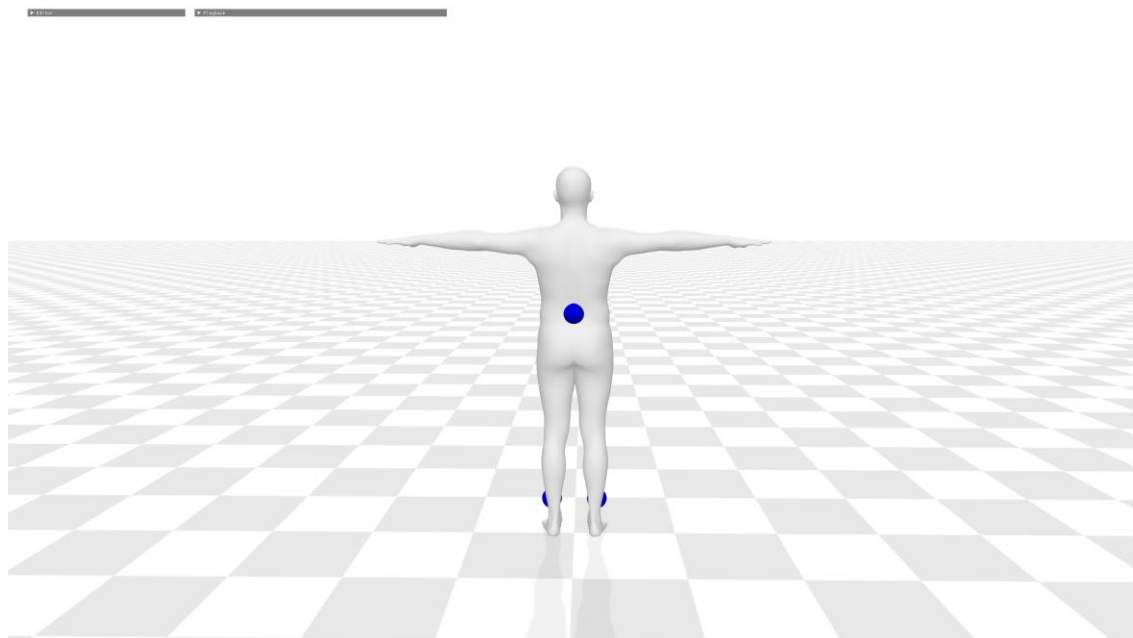


Figura 14: Visualización en Aitviewer de la IMU de referencia de la zona lumbar.

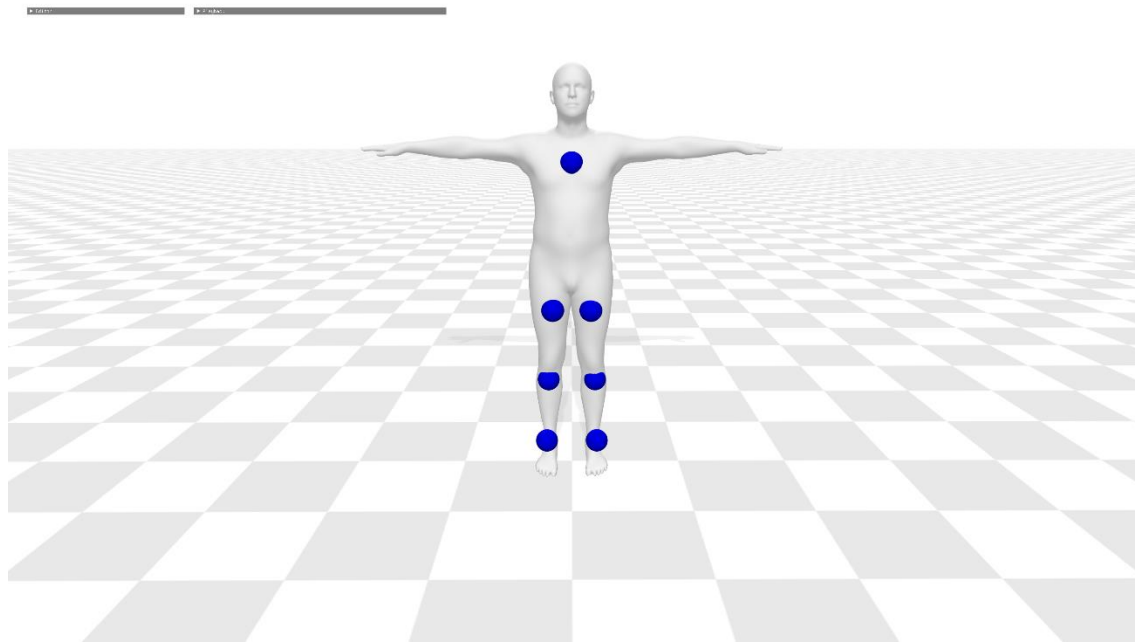


Figura 15: Visualización en Aitviewer de la IMU las IMUs virtuales.

Para poder usar estos datos con el Filtro de Madgwick seleccionado se necesitaban tener los parámetros de la siguiente manera:

- Acelerómetro: una matriz de tres elementos que contenga los datos del acelerómetro, pero puede ser cualquier unidad ya que se utiliza un valor normalizado.
- Giroscopio: una matriz de tres elementos que contenga los datos del giroscopio en radianes por segundo.
- Magnetómetro: una matriz de tres elementos que contenga los datos del magnetómetro, puede ser cualquier unidad ya que se utiliza un valor normalizado.

Por ello, los datos del giroscopio debían transformarse de matrices de rotación a velocidades angulares $(\omega_x, \omega_y, \omega_z)$ en radianes por segundo. Se diseñó un script de Python para automatizar el proceso basándose en los siguientes principios matemáticos [50].

Primero se definieron las matrices de rotación $R(t_0)$ y $R(t_1)$ en los instantes t_0 y t_1 respectivamente. Luego se calcula la derivada temporal de la matriz de rotación, \dot{R} , que se aproxima como la diferencia entre $R(t_0)$ y $R(t_1)$ dividida por el intervalo de tiempo Δt :

$$\dot{R} \approx \frac{R(t_1) - R(t_0)}{\Delta t} \quad (39)$$

A continuación, hay que calcular el producto de la transpuesta de la derivada y la matriz de rotación original, que tiene relación directa con las velocidades angulares:

$$\dot{R}^T R(t_0) = \begin{bmatrix} 0 & -\omega_z & \omega_y \\ \omega_z & 0 & -\omega_x \\ -\omega_y & \omega_x & 0 \end{bmatrix} \quad (40)$$

Por último, se pueden extraer las componentes de la velocidad angular de la siguiente manera:

$$\omega_x = R^T \dot{R}_{32} - R^T \dot{R}_{23} \quad (41)$$

$$\omega_y = R^T \dot{R}_{13} - R^T \dot{R}_{31} \quad (42)$$

$$\omega_z = R^T \dot{R}_{21} - R^T \dot{R}_{12} \quad (43)$$

Por otra parte, al ser datos sintetizados mediante la estimación del algoritmo, estos no poseían ruido. Por ello, era necesario introducir cierto ruido que simulase un entorno real en el que las IMUs no capturan la señal perfectamente limpia.

Se diseñó otro script en Python de manera que, para los datos de aceleración, generase ruido gaussiano y lo combinase con una constante multiplicada por un signo aleatorio, sumándolo a los valores originales. Para los datos de orientación, se sigue un proceso similar, aplicando ruido gaussiano y una constante multiplicada por un signo aleatorio a los valores originales. Finalmente, los datos modificados se guardan en nuevos archivos .pkl en la carpeta de destino, simulando así un entorno más realista con ruido en las mediciones.

Los resultados se pueden apreciar en las Figura 16 y 17.

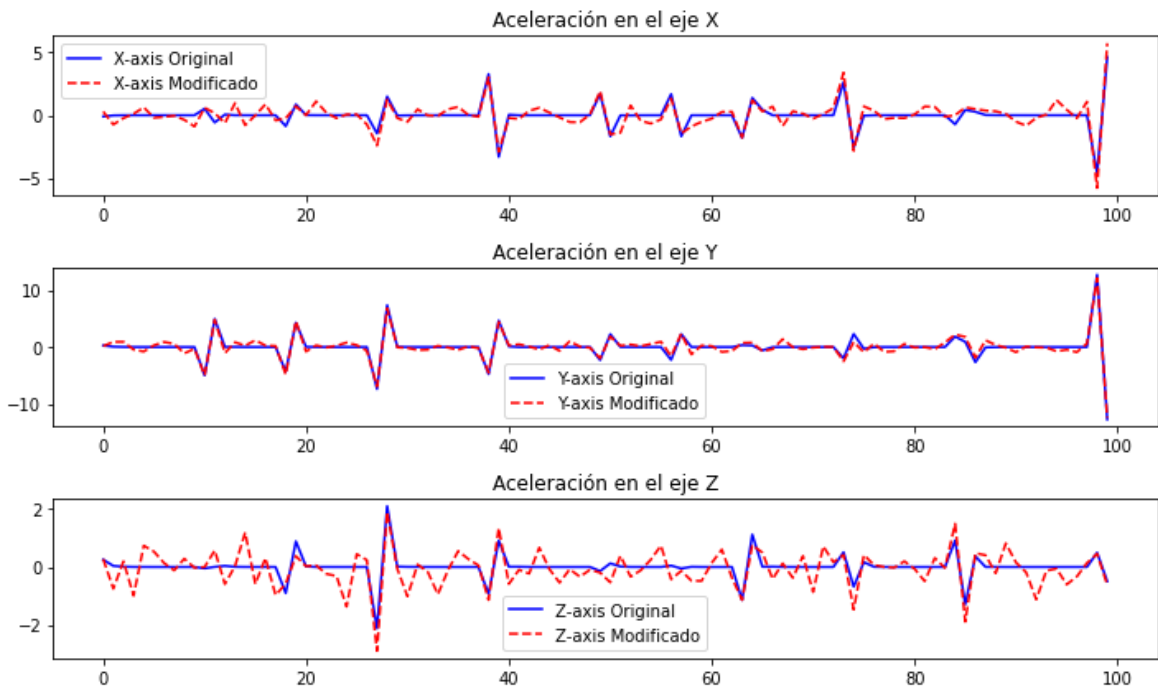


Figura 16: Gráficas comparativas entre la aceleración sin ruido (azul) y con ruido gaussiano (rojo).

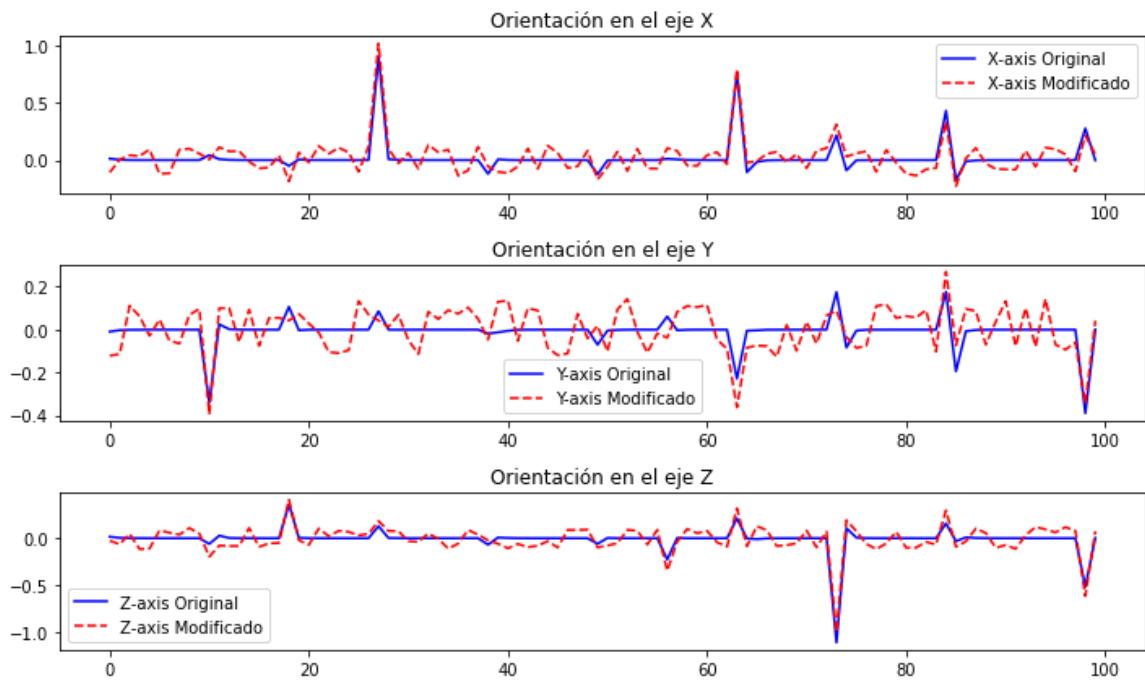


Figura 17: Gráficas comparativas entre la orientación sin ruido (azul) y con ruido gaussiano (rojo).

4.2. Segunda opción: base de datos con datos de magnetómetro

Tras realizar varias pruebas con los datos de la base sintetizada, que se mostrarán en mayor profundidad en el Capítulo 5, se decidió buscar una nueva base de datos, esta vez que incorporara datos también de magnetómetro. De esta manera, se podría conseguir un entorno más parecido al del exoesqueleto del proyecto, cuyas IMUs proporcionan datos de aceleración, velocidades angulares y campos magnéticos. Además, esta incorporación permitiría realizar una mejor estimación de los ángulos.

La nueva base de datos seleccionada proviene del estudio realizado por Luo et al. [51], que se centra en el análisis del rendimiento de la marcha humana en superficies irregulares y desiguales utilizando sensores portátiles. Esta base de datos proporciona un conjunto de datos recogidos de treinta participantes que caminaron sobre nueve superficies exteriores diferentes mientras llevaban seis sensores IMU, como se muestran en la Figura 18. Los sensores IMU utilizados (MTw Awinda Xsens) miden la aceleración, la velocidad angular y el campo magnético en tres dimensiones.

La base de datos incluye datos de treinta participantes (quince hombres y quince mujeres) con edades comprendidas entre 18 y 33 años. Los participantes caminaron sobre nueve superficies diferentes, que incluyen:

1. Superficie plana y uniforme (horizontal, pavimentada).
2. Escaleras ascendentes (cemento).
3. Escaleras descendentes (cemento).
4. Pendiente ascendente (cemento).
5. Pendiente descendente (cemento).
6. Césped.
7. Superficie inclinada hacia la izquierda (pavimentada).

8. Superficie inclinada hacia la derecha (pavimentada).
9. Adoquines irregulares.

Cada participante realizó seis pruebas en cada una de estas superficies, resultando en un conjunto de datos diversificado que refleja condiciones del mundo real y que son adecuadas para entornos de rehabilitación.

Los sensores IMU se colocaron en seis ubicaciones específicas del cuerpo (ver Figura 18) utilizando bandas proporcionadas por el fabricante para asegurar una sujeción adecuada. Las ubicaciones de los sensores fueron:

1. Centro de la muñeca en el antebrazo dorsal.
2. Centro de ambos muslos anteriores.
3. Centro a 5 cm por encima de los procesos óseos de ambos tobillos.
4. Región posterior al nivel de la articulación L5/S1.

Antes de cada prueba, se realizó un procedimiento de calibración que consistía en movimientos específicos del tronco, brazos y piernas para asegurar la correcta alineación y calibración de los sensores. Esta calibración se realizó tres veces antes de cada prueba experimental.

Aunque no fuesen exactamente los sensores que se buscaban podrían servir como base de datos útil para aplicar el Filtro de Madgwick.



Figura 18: Disposición de los sensores del experimento [51].

Durante cada prueba, los participantes caminaron a una velocidad autoseleccionada sin cambios de dirección, excepto en las pruebas realizadas en superficies planas, de césped y adoquines, donde se cambiaba de dirección cada dos pruebas. Cada prueba duró aproximadamente 16.4 segundos. Se proporcionaron descansos adecuados entre las pruebas para prevenir la fatiga.

Los datos se recopilaron utilizando el software MTw Awinda con una frecuencia de muestreo de 100 Hz. Los datos en bruto se sincronizaron y se exportaron en formato .txt para su posterior procesamiento. Además, se utilizó un filtro de paso bajo Butterworth de 2º orden con una frecuencia de corte de 6 Hz para suavizar las trayectorias de los sensores.

La estructura de los datos es detallada y bien organizada, facilitando el análisis y procesamiento posterior. Los archivos de datos en bruto están etiquetados sistemáticamente con un identificador que indica la condición de la superficie y la ubicación del sensor. Por ejemplo, el archivo '9-000_00B432CC.txt' corresponde a los datos del sensor del tronco en una superficie plana y uniforme para un participante específico.

Cada archivo contiene:

- Datos de aceleración en las tres direcciones tanto con gravedad como sin gravedad.

- Datos de giroscopio (velocidad angular) en las tres direcciones.
- Datos de magnetómetro (campo magnético) en las tres direcciones.
- Información adicional sobre la captura de datos, como el contador de paquetes, tiempo de muestra, tasa de actualización, perfil del filtro y versión del firmware.

Una de las ventajas de esta base de datos, en un principio, era que contenía datos de los ángulos de Euler ya calculados. De esta manera, se podría comprobar los resultados obtenidos con el Filtro de Madgwick. Sin embargo, como se mostrará en el Capítulo 5, al representar los datos de los ángulos de Euler calculados por los propios autores mediante un script sencillo de Python, las señales que se mostraban no parecían muy convincentes, provocando que no fuese un buen método para comprobar la eficacia del filtro propuesto.

4.3. Tercera opción: base de datos con IMUs propias

Finalmente, se tuvo acceso a las IMUs que se emplearán en el proyecto real del exoesqueleto y que se encuentran en el Campus de Móstoles de la Universidad Rey Juan Carlos. No se pudo contar con ellas antes por problemas logísticos.

Generar una base de datos utilizando IMUs propias permitía obtener datos más específicos y controlados, adaptados perfectamente a las necesidades del proyecto. El sistema de captura con el que se contaba era Tech-MCS V3 de Technaid. A continuación, se detallarán los elementos clave de este [52].

En primer lugar, las Tech-IMUs, sensores inerciales tridimensionales que están basados en tecnología MEMS (Micro Electro Mechanical Systems) e incluyen un acelerómetro, un giróscopo y un magnetómetro. Cada IMU mide aproximadamente 36x26x11 mm y pesa 10 gramos, lo que las hace extremadamente portátiles y fáciles de montar en diferentes partes del cuerpo, ver Figura 19.



Figura 19: Unidad de Medida Inercial de 3 ejes, Tech-IMU [52].

Por otro lado, el Tech-HUB V3, el dispositivo concentrador de datos que permite conectar y sincronizar hasta 16 Tech-IMUs simultáneamente, ver Figura 20: Modelo del Tech-HUB V3 [52]. Este hub puede comunicarse con un PC a través de USB o Bluetooth, y también puede almacenar datos en una tarjeta MicroSD de 8 GB para operaciones sin conexión. En este caso se empleó la modalidad de conexión por Bluetooth.

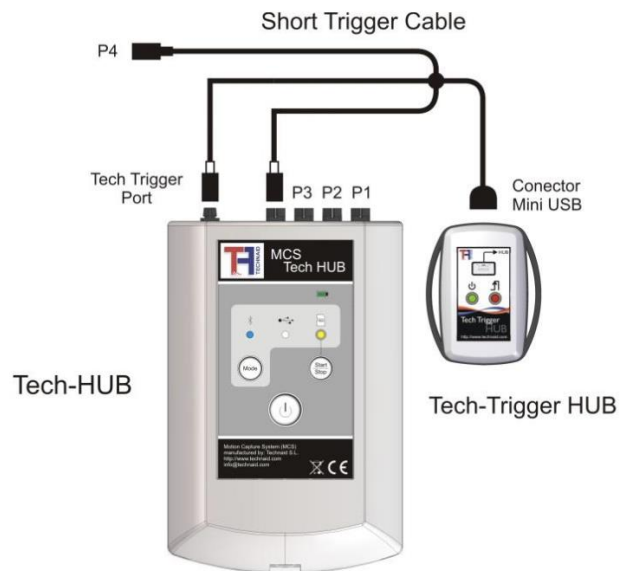


Figura 20: Modelo del Tech-HUB V3 [52].

Por último, el Software Tech MCS Studio. Esta aplicación permite la gestión y visualización de los datos de movimiento capturados por las IMUs, como se puede ver en la Figura 21. Ofrece funciones avanzadas como la segmentación automática de la marcha, la obtención de velocidades angulares y rangos articulares, y la generación de informes detallados en formato PDF. El software es compatible con Windows y proporciona herramientas para visualizar gráficamente en 3D los datos capturados.

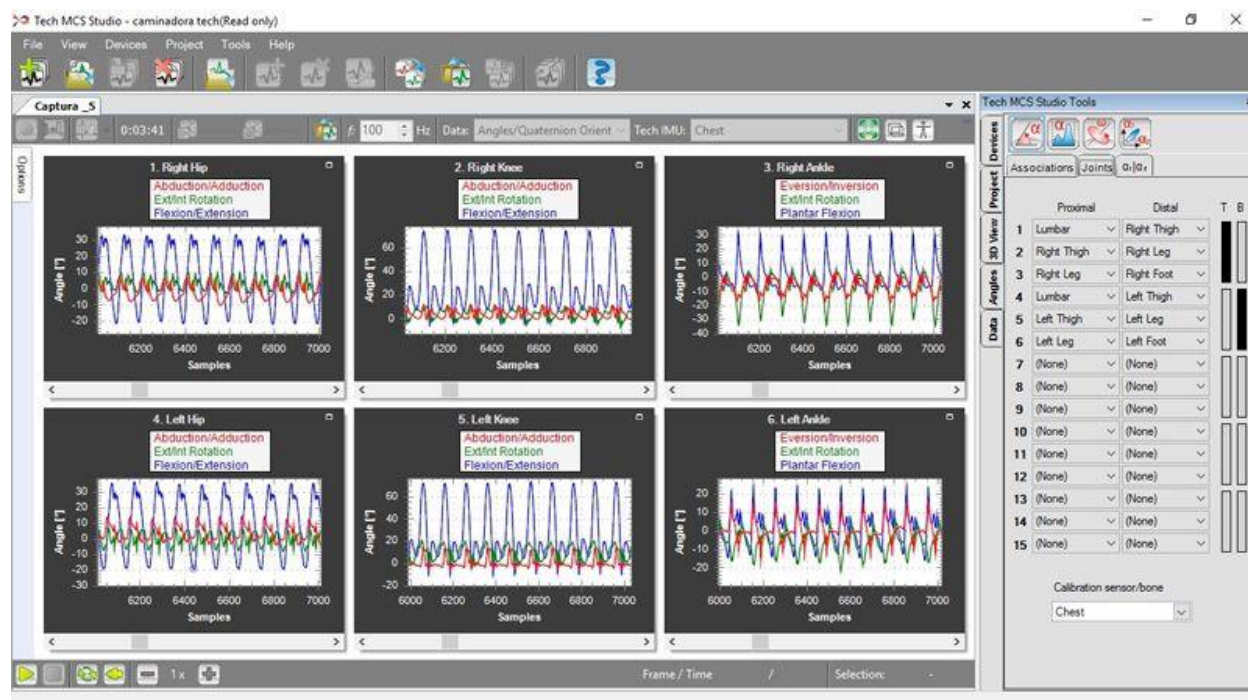


Figura 21: Interfaz del software Tech MCS Studio [53].

Los Tech_IMU V3 pueden recopilar datos de los movimientos en 4 tipos de formato diferentes. Estos reciben el nombre de: Digitales, Físicos, Orientación en DCM y Orientación en Cuaterniones. Se describen a continuación:

- Digitales: proporciona los datos brutos de los sensores del Tech-IMU, tal como los entrega el conversor analógico/digital (AD). Los datos de los giróscopos, acelerómetros, magnetómetros y termómetro se envían al Tech-HUB V3 sin modificaciones ni ajustes. Las señales del acelerómetro y giroscopio se digitalizan a 16 bits, y las del magnetómetro a 12 bits, con rangos de valores de 0 a 65535 y de 0 a 4095, respectivamente.

- Físicos: proporciona los datos de los sensores del Tech-IMU después de ser calibrados, correspondiendo a las magnitudes físicas medidas (aceleración, velocidad angular, campo magnético y temperatura) en formato float32.
- Orientación en DCM: proporciona la orientación representada mediante una matriz de rotación 3x3 que contiene los cosenos directores de los ejes del Tech-IMU respecto al sistema de coordenadas de referencia global. Los valores están en el rango de -1.0 a +1.0 en formato float32.
- Orientación en Cuaterniones: proporciona la orientación del Tech-IMU utilizando cuatro valores float32. Estos valores corresponden al cuaternión que representa la rotación de los ejes del sensor respecto al sistema de coordenadas de referencia. Los datos se presentan en el formato $[q_w, q_x, q_y, q_z]$, donde q_w es la parte real del cuaternión y q_x, q_y, q_z son las partes imaginarias. El rango de valores para los cuaterniones está entre -1.0 y +1.0.

Una ventaja de este sistema es que permite enviar datos en dos formatos distintos a la vez. Las opciones posibles son:

- Digitales
- Físicos
- Orientación es DCM
- Orientación en Cuaterniones
- Físicos y Orientación en Cuaterniones
- Cuaterniones/Ángulos articulares

De esta manera, aunque no permitía guardar la combinación óptima, datos físicos y los ángulos ya calculados, para comparar los ángulos estimados con Filtro de Madgwick con los del propio sistema Tech-MCS V3, se podía obtener los datos físicos y los cuaterniones. Por lo tanto, existen dos formas de comparar, por un lado, calcular los cuaterniones con FM, y por otro lado, transformar los cuaterniones de las IMUs a ángulos de Euler mediante un script. Esto se verá en mayor profundidad en el Capítulo 5.

Para realizar la estimación de la orientación a partir de las unidades inerciales el sistema Tech-MCS V3 emplea el Filtro de Kalman Extendido, que se ejecuta en la propia IMU. En la Figura 22 se muestra un diagrama de bloques del proceso. Esto es ideal para poder hacer una comparación precisa entre la estimación con Filtro de Madgwick y Filtro de Kalman Extendido.

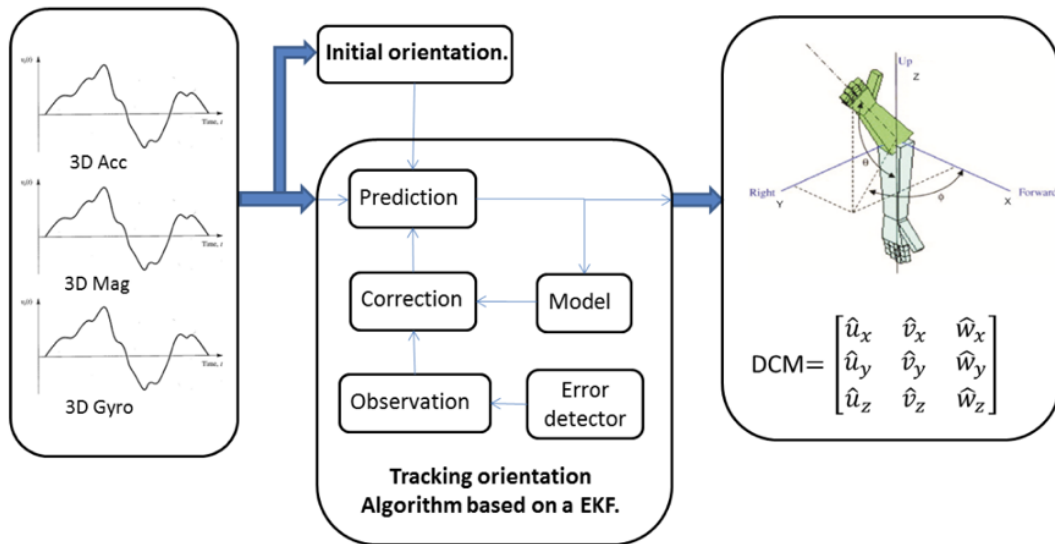


Figura 22: Esquema del proceso interno del Tech-IMU para obtener la orientación 3D [52].

Para realizar el proceso de captura de datos, lo primero que se hizo fue colocar las IMUs en el sujeto de prueba. El sistema de captura viene con un kit de accesorios elásticos (ver Figura 23) para colocar los sensores en las distintas partes del cuerpo.



Figura 23: Accesorios elásticos para colocación de IMUs [52].

En este caso se colocaron 8 IMUs en las siguientes posiciones, tal y como se puede observar en la Figura 24:

- 2 IMUs en los pies
- 2 IMUs en la zona tibial
- 2 IMUs en los muslos
- 1 IMU en la cadera
- 1 IMU en el pecho

Además, se colocó en la cintura con un cinturón especial el Tech-HUB V3 donde van conectadas todas las IMUs mediante cables, como se puede ver en la Figura 25.



Figura 24: Sujeto de prueba con el sistema Tech-MCS V3 colocado.

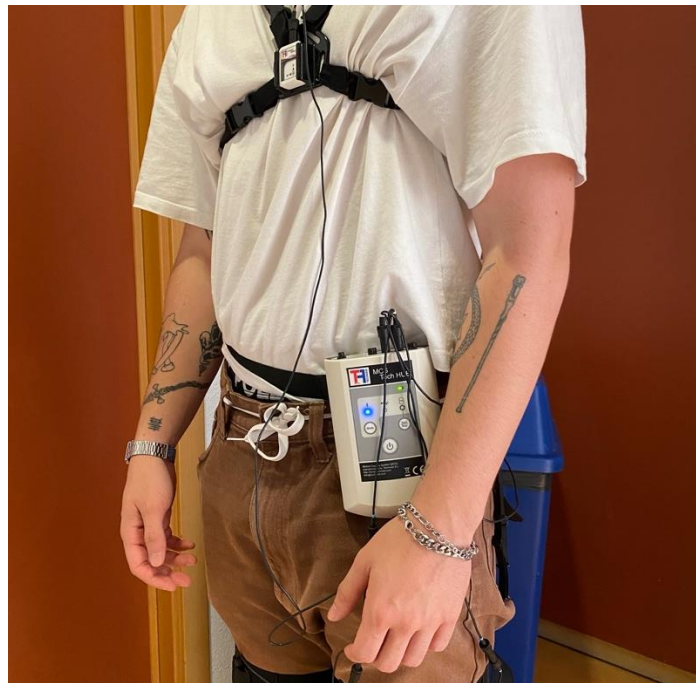


Figura 25: Cinturón con el Tech-HUB V3.

Se utilizó el software Tech MCS Studio para configurar los parámetros de captura, como la frecuencia de muestreo, y para calibrar los sensores. Los datos se capturaron a 60 Hz, siendo este el máximo valor permitido para la configuración Físicos / Orientación con Cuaterniones.

El siguiente paso es la calibración inicial para ajustar las IMUs a las condiciones específicas del entorno de medición. Esta se hacía automáticamente cada vez que se grababa un muestra. El sujeto debía permanecer quieto hasta dejar de escuchar un pitido.

Las sesiones de captura de los movimientos se podían dividir en 3 tipos: marcha, estar parado y sentarse. Los sensores registraban y enviaban la información al ordenador mediante Bluetooth en tiempo real. Se realizaron varias pruebas y grabaciones con cada tipo y de distintas duraciones (entre 10 y 30 segundos por muestra).

También se grabó una sesión de captura de movimiento en el que las IMUs se encontraban estáticas apoyadas en una mesa. Esto permite calcular los errores y derivas medias de las señales para calcular los parámetros beta y zeta. Esto se verá en mayor profundidad en el apartado 5.3.2.

Después, de ser procesados utilizando el software Tech MCS Studio, los archivos se pueden exportar en el ordenador. Para la configuración Físicos / Orientación con Cuaterniones hay dos tipos de archivos: formato .capp para los datos físicos y formato .capq para los datos en cuaterniones. Estos formatos no son usuales y no permiten una fácil lectura. Por ello se desarrollaron 2 scripts para pasar cada uno de los tipos de archivo a formato CSV.

Capítulo 5

Experimentos y resultados

En este apartado se expondrán los experimentos realizados con las bases de datos vistas en el capítulo anterior.

5.1. Experimentos con la primera base de datos

En primer lugar, se desarrolló un código principal que ejecutase el Filtro de Madgwick (MF) y representase gráficamente los resultados en ángulos de Euler en grados. En la Figura 26 se puede observar el resultado del MF para la primera IMU. Existen tres problemas que llevaron a desestimar esta base de datos:

- 1) No existen otros datos obtenidos por otros métodos con los que comparar el resultado.
- 2) Se desconoce la posición inicial de la IMU y, por tanto, la inicialización del cuaternión (se usó $[1, 0, 0, 0]$). Para visualizar mejor los resultados se usó un cuaternión inicial lo más próximo a los valores iniciales, como se puede observar en la Figura 27.
- 3) Esta base de datos no contenía datos de magnetómetro con los que mejorar la precisión.

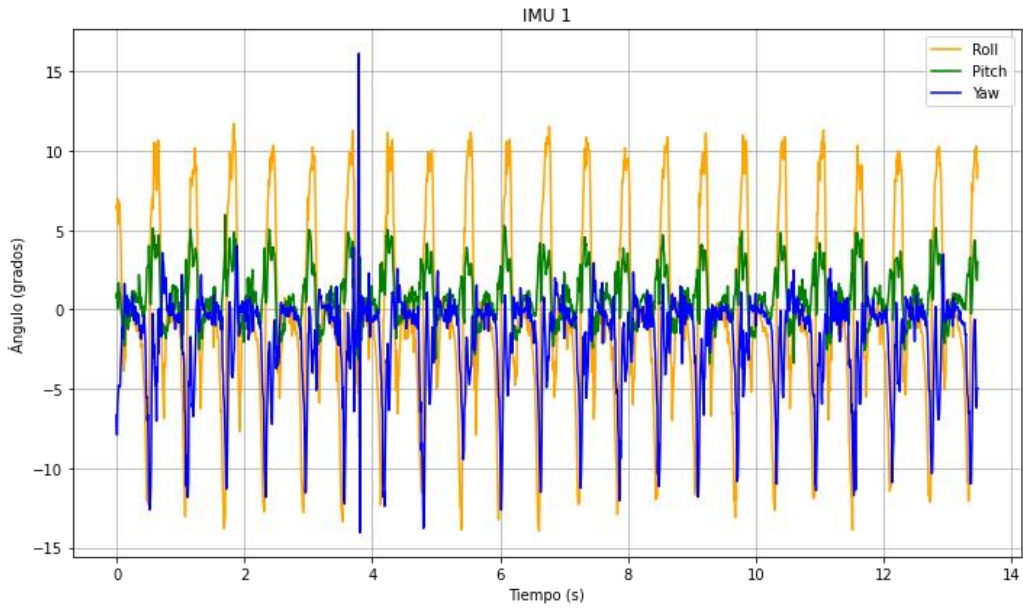


Figura 26: Representación en ángulos de Euler en grados de la primera IMU con datos sintetizados con ruido añadido.

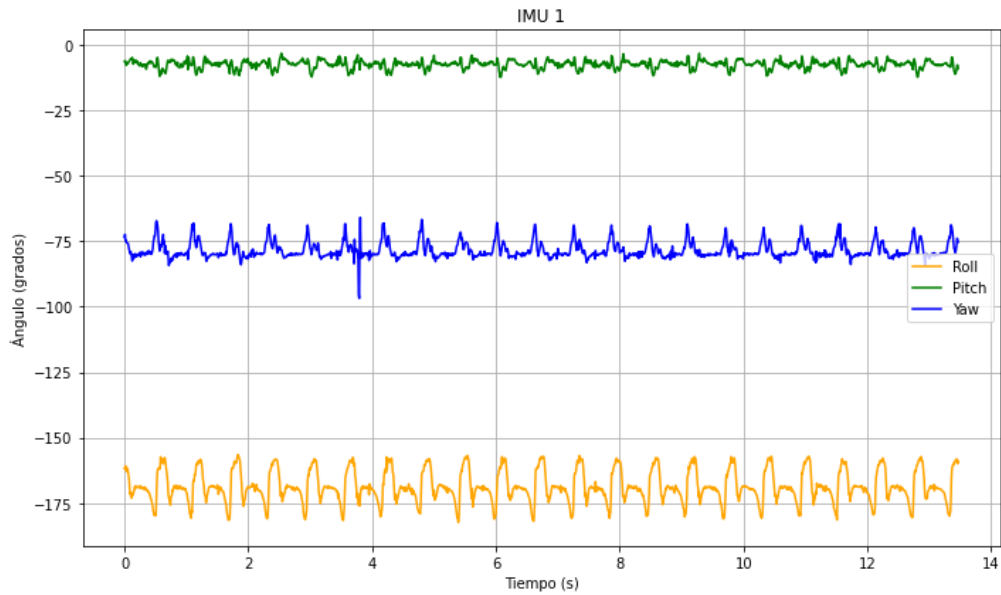


Figura 27: Representación en ángulos de Euler en grados de la primera IMU con datos sintetizados con ruido añadido y con cuaternión próximo a los valores iniciales.

5.2. Experimentos con la segunda base de datos

De nuevo creo otro código principal, adaptado al formato de este conjunto de datos. En este caso tampoco se disponía del cuaternión inicial, por lo que se puede observar que comienza la señal en el origen del eje y (ver Figura 28). Por otro lado, si se disponía de los datos de los ángulos de Euler en grados ya calculados por los creadores de la base, sin embargo, como se puede observar en la Figura 29, los ángulos no fueron bien estimados. Además, los creadores tampoco indicaban en el documento el tipo de algoritmo que se usó para ello. Por lo tanto, por no disponer de un buen criterio comparativo, también se decidió evitar esta base de datos.

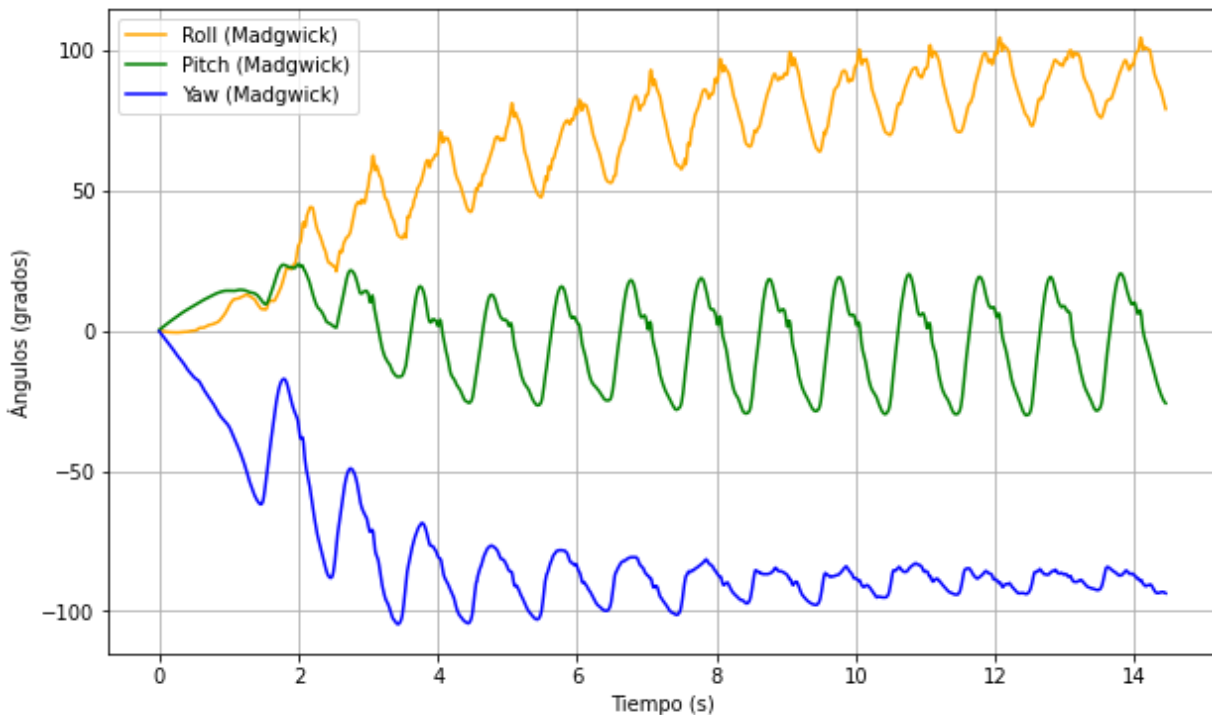


Figura 28: Representación en ángulos de Euler en grados de la segunda base. Captura de IMU de la tibia derecha de sujeto caminando.

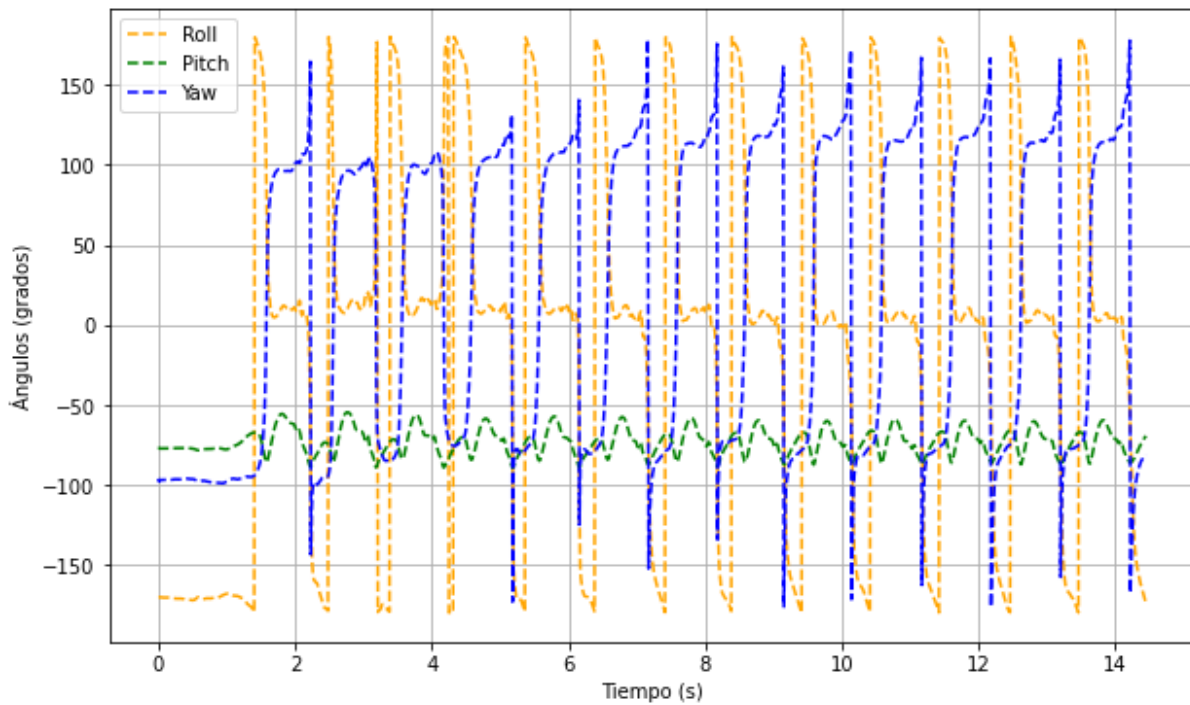


Figura 29: Representación en ángulos de Euler en grados calculados por los creadores de la segunda base. Captura de IMU de la tibia derecha de sujeto caminando.

5.3. Experimentos con la tercera base de datos

5.3.1. Desarrollo del código principal

Tras la selección de la base de datos más adecuada para este proyecto se procedió a la parte experimental. En primer lugar, se debía diseñar el código principal cuya finalidad es comparar los resultados obtenidos al aplicar el Filtro de Madgwick a los datos físicos con los calculados por las propias IMUs con el Filtro de Kalman Extendido de la base de datos vista en el apartado 4.3. Tercera opción: base de datos con IMUs propias Se compararán tanto los resultados obtenidos en cuaterniones como en ángulos de Euler. De esta manera es necesario diseñar dos tipos de código diferente. Por un lado, un script que inicialice el código del Filtro de Madgwick planteado en el apartado 3.2. una manera adaptada a la estructura de los datos del proyecto. Dentro de este archivo es necesario que devuelva

dos tipos de formato de salida: cuaterniones y ángulos de Euler. Por otro lado, es necesario otro código que permita pasar los cuaterniones calculados por las IMUs con el EKF a ángulos de Euler para hacer la comparación.

Así, el funcionamiento del código podría describirse de esta forma. El primer paso del proceso es la carga de datos desde un archivo CSV. Este archivo contiene lecturas de sensores inerciales. Se seleccionan específicamente las columnas que contienen datos del acelerómetro, giroscopio y magnetómetro. Mediante un ajuste inicial se asegura de que los nombres de las columnas no tengan espacios en blanco innecesarios. Los datos de estas columnas se extraen y se almacenan en matrices separadas para su uso posterior.

Una vez que los datos de los sensores están cargados, se utiliza el Filtro de Madgwick para calcular los cuaterniones. Se comienza con la inicialización del Filtro de Madgwick: se crea una instancia del filtro, que estará listo para procesar los datos de los sensores.

Luego, para cada conjunto de lecturas del acelerómetro, giroscopio y magnetómetro, el filtro de Madgwick se actualiza. Esta actualización ajusta la orientación estimada del objeto basándose en los nuevos datos de los sensores, como se vio de una forma más extendida en el apartado 3.2. Después de cada actualización, el cuaternión resultante, que representa la orientación del objeto, se almacena en una lista para su uso posterior.

Los cuaterniones obtenidos del MF se convierten luego en ángulos de Euler. Los ángulos de Euler (Roll, Pitch y Yaw) son una representación más intuitiva de la orientación que los cuaterniones, especialmente útil para análisis y visualización.

Finalmente, los ángulos de Euler se visualizan para proporcionar una representación gráfica de cómo cambia la orientación del objeto a lo largo del tiempo. Se usa la frecuencia de muestreo específica para los datos, y los ángulos de Euler se grafican en función del tiempo.

Para mejorar la adaptabilidad se modificó el archivo original “madgwickahrs.py” para que desde el archivo “main” se pudiese elegir los valores de beta y zeta al llamar a la función.

Por último, es importante indicar el primer cuaternión de la muestra para evitar que el sistema comience en el origen de coordenadas. El sistema Tech-MCS V3 dedica unos segundos a la calibración durante su inicialización, con el fin de obtener el valor del cuaternión inicial. Dado que este valor no puede obtenerse directamente de las IMUs para aplicar el Filtro de Madgwick, se utilizó el valor calculado por el sistema mediante el EKF. Esta es una limitación en el caso de que desconozca el cuaternión inicial.

5.3.2. Ajuste de parámetros

Una vez definido el código principal, el siguiente paso consistía en debía definir los parámetros iniciales requeridos vistos en el Capítulo 3. Como ya se comentó en la descripción de la tercera base de datos, la frecuencia con la que se tomaron las muestras fue de 60 Hz.

Para ajustar los parámetros ζ y β del Filtro de Madgwick, se utiliza una metodología basada en los errores de medición y deriva del giroscopio. A continuación, se explica este proceso con fórmulas y fundamentos teóricos [44].

En primer lugar, se deben tomar muestras en la que las IMUs estén apoyadas en una superficie plana sin movimiento, logrando así una señal lo más estable posible. Esto permite calcular el error medio de la señal.

$$\bar{\omega}_i = \frac{1}{N} \sum_{k=1}^N \omega_{i,k} \quad (44)$$

Donde $\omega_{i,k}$ es la lectura del giroscopio en el eje i (x, y, z) en el momento k , y N es el número total de muestras. Para calcular la desviación estándar se recurre a la siguiente ecuación:

$$\sigma_i = \sqrt{\frac{1}{N-1} \sum_{k=1}^N (\omega_{i,k} - \bar{\omega}_i)^2} \quad (45)$$

Por último, se calcula beta, como:

$$\beta = \sqrt{\frac{3}{4}} \times \sigma_i \quad (46)$$

Por otro lado, para calcular la deriva, estimamos la diferencia de la media de las lecturas entre dos intervalos consecutivos dividido por el tiempo entre ellos:

$$drift_i = \frac{\bar{\omega}_{i,t+1} - \bar{\omega}_{i,t}}{T} \quad (47)$$

Para calcular zeta se usa la siguiente ecuación:

$$\zeta = \sqrt{\frac{3}{4}} \times drift_i \quad (48)$$

En este caso, los valores de beta y zeta obtenidos fueron 0,35 y 0,07 respectivamente.

Existe otra posibilidad de hallar los valores óptimos de beta y zeta de forma empírica mediante la comparación con un valor de referencia. De esta manera, se desarrolló un código en el que mediante un bucle se obtenía la diferencia media entre la señal la estimada con el EKF y con el FM con distintos valores de beta y zeta.

Para ello, se hizo un barrido más amplio escogiendo valores desde el 0.1 hasta el 1 en ambos casos. Con una diferencia media de 0.11546 para los valores beta = 0.34 y zeta = 0.01. A continuación, se hizo una segunda prueba acotando más los valores a probar siendo entre 0.3 y 0.4 para beta y entre 0.01 y 0.1 para zeta. Finalmente, los valores óptimos alcanzados mediante este experimento fueron 0,33 y 0,05, proporcionando una diferencia media de 0.11535.

En la Figura 30 se puede ver una representación de diez segundos de la comparación de una selección de combinaciones de beta y zeta en un primer barrido. En la Figura 31 se puede observar la segunda prueba con los parámetros más ajustados, en la que se puede apreciar una diferencia mucho menor entre señales.

De esta manera, se comprobó de manera empírica que los valores de beta y zeta obtenidos con el primer método eran correctos y, por lo tanto, se procedió a usar los valores de beta = 0.35 y zeta = 0.07.

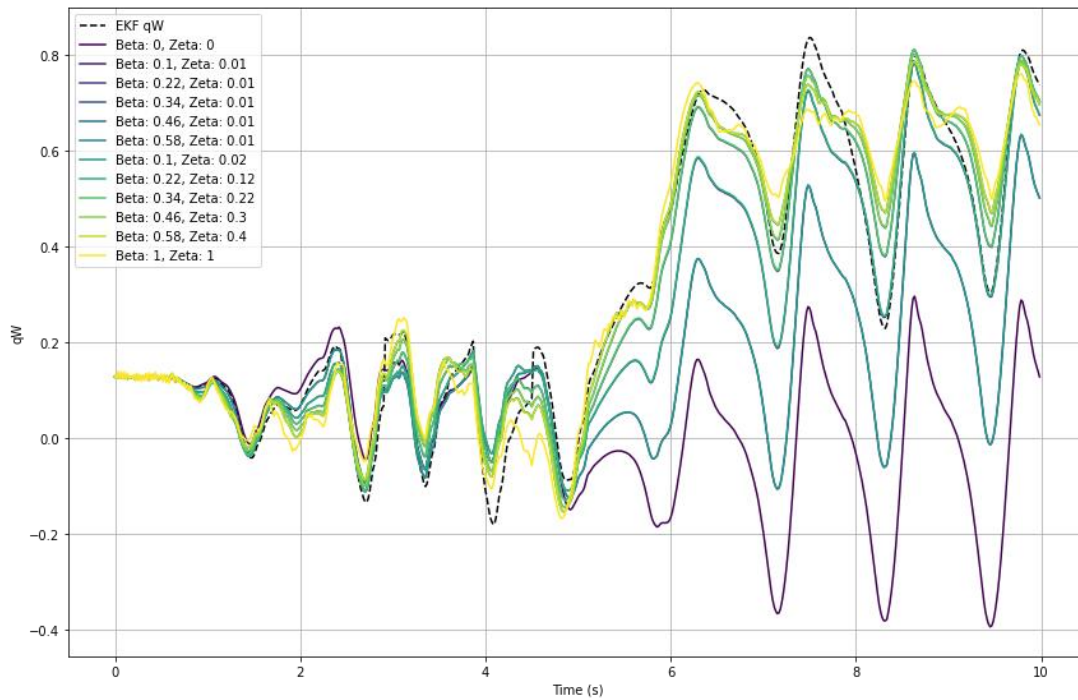


Figura 30: Comparación del cuaternión qW con diferentes combinaciones de ζ y β en primera prueba.

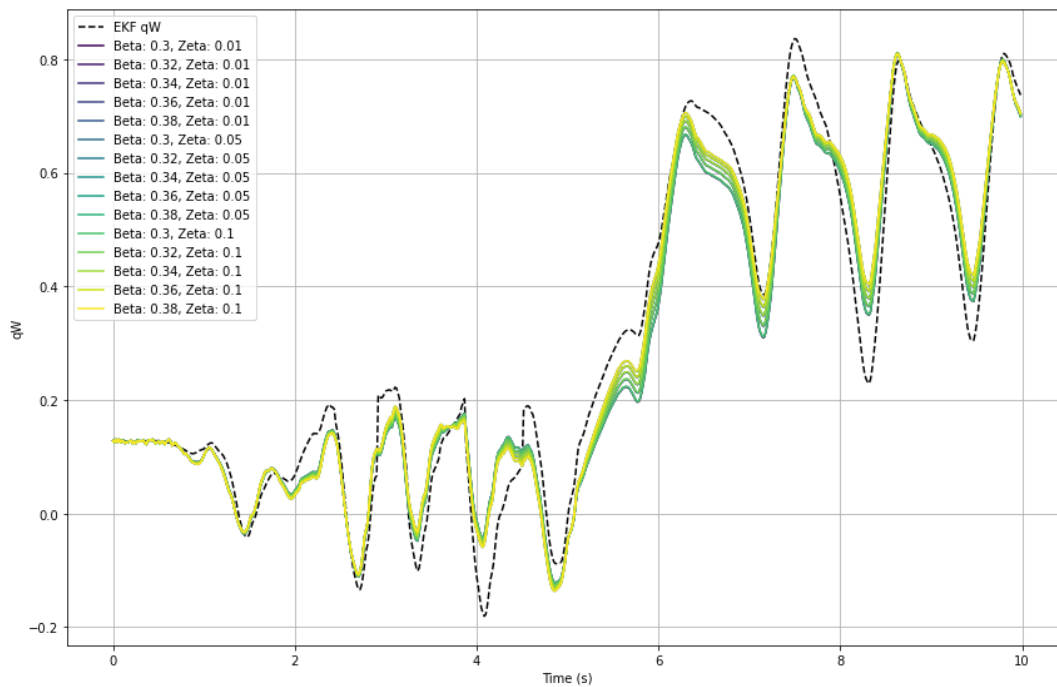


Figura 31: Comparación del cuaternión qW con diferentes combinaciones de ζ y β en segunda prueba.

5.3.3. Resultados finales

Tras haber conseguido los parámetros de ζ y β óptimos se procedió a realizar distintas pruebas con las diferentes muestras de la base de datos. Se comenzó por calcular únicamente los cuaterniones y mostrar los resultados de dos formas distintas. Por un lado, están las gráficas comparativas de las dos señales estimadas por ambos filtros. Por otro lado, se calcula la diferencia entre las dos señales a lo largo del tiempo y se muestran en una gráfica.

Para mostrar los resultados se ha decidido escoger capturas de IMU en el muslo y en el pie, donde la estimación es más complicada debido al movimiento. A continuación, se muestran diferentes ejemplos andando (Figura 32, 33, 34, 35), sentándose (Figura 36, 37) y estando parado de pie (Figura 38, 39).

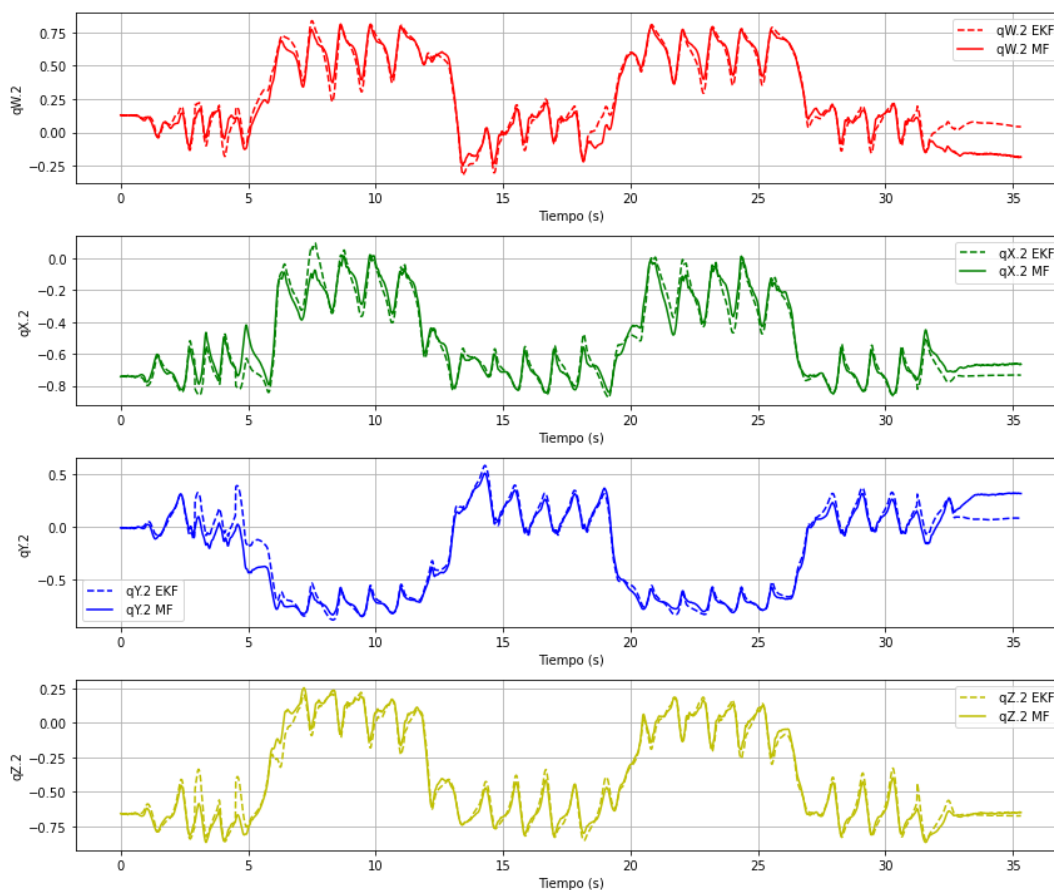


Figura 32: Comparación entre la señal filtrada con el Filtro de Madgwick y el Filtro de Kalman Extendido. Captura de IMU en el muslo derecho de muestra de marcha.

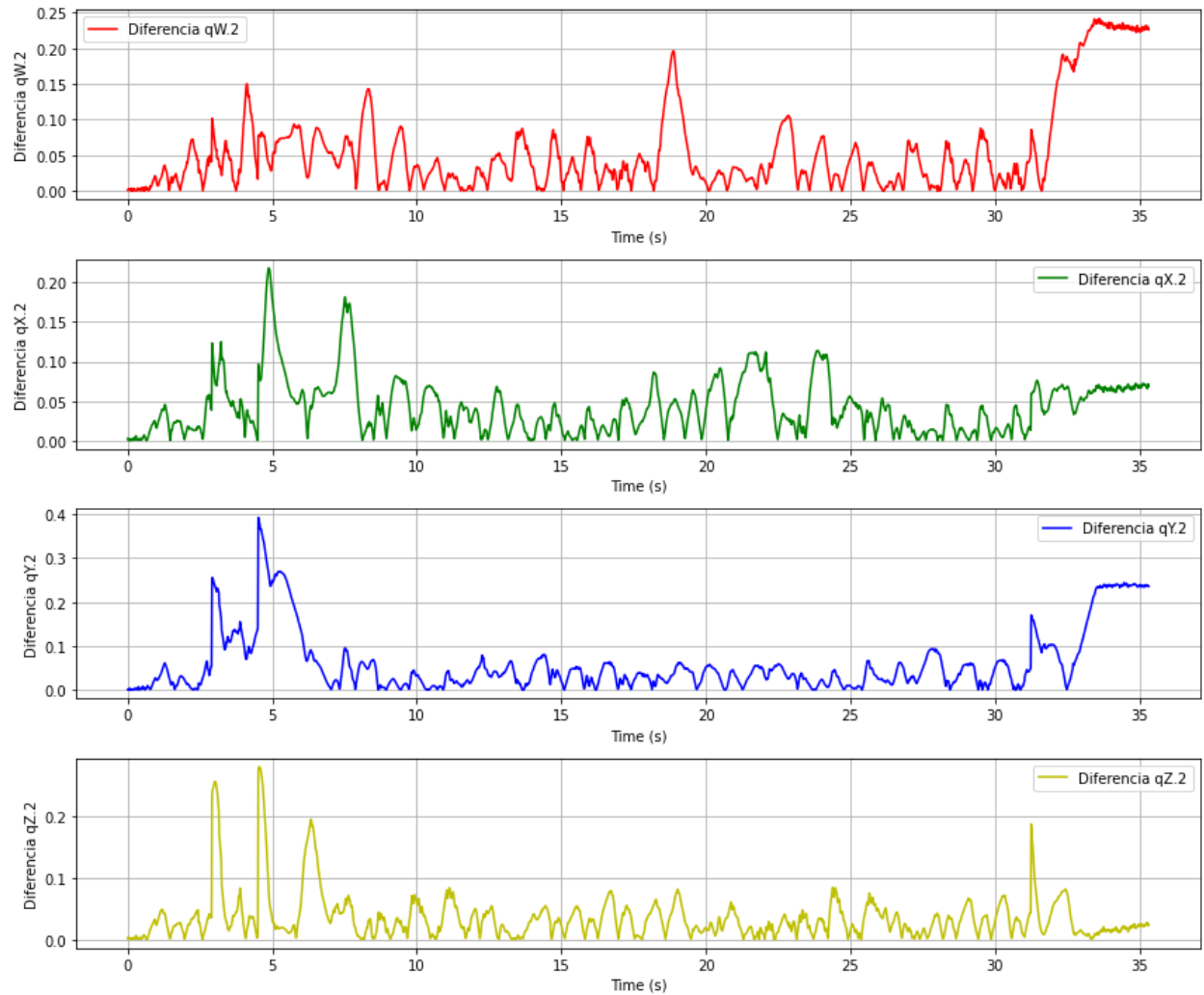


Figura 33: Diferencia entre la señal filtrada con el Filtro de Madgwick y el Filtro de Kalman Extendido. Captura de IMU en el muslo derecho de muestra de marcha.

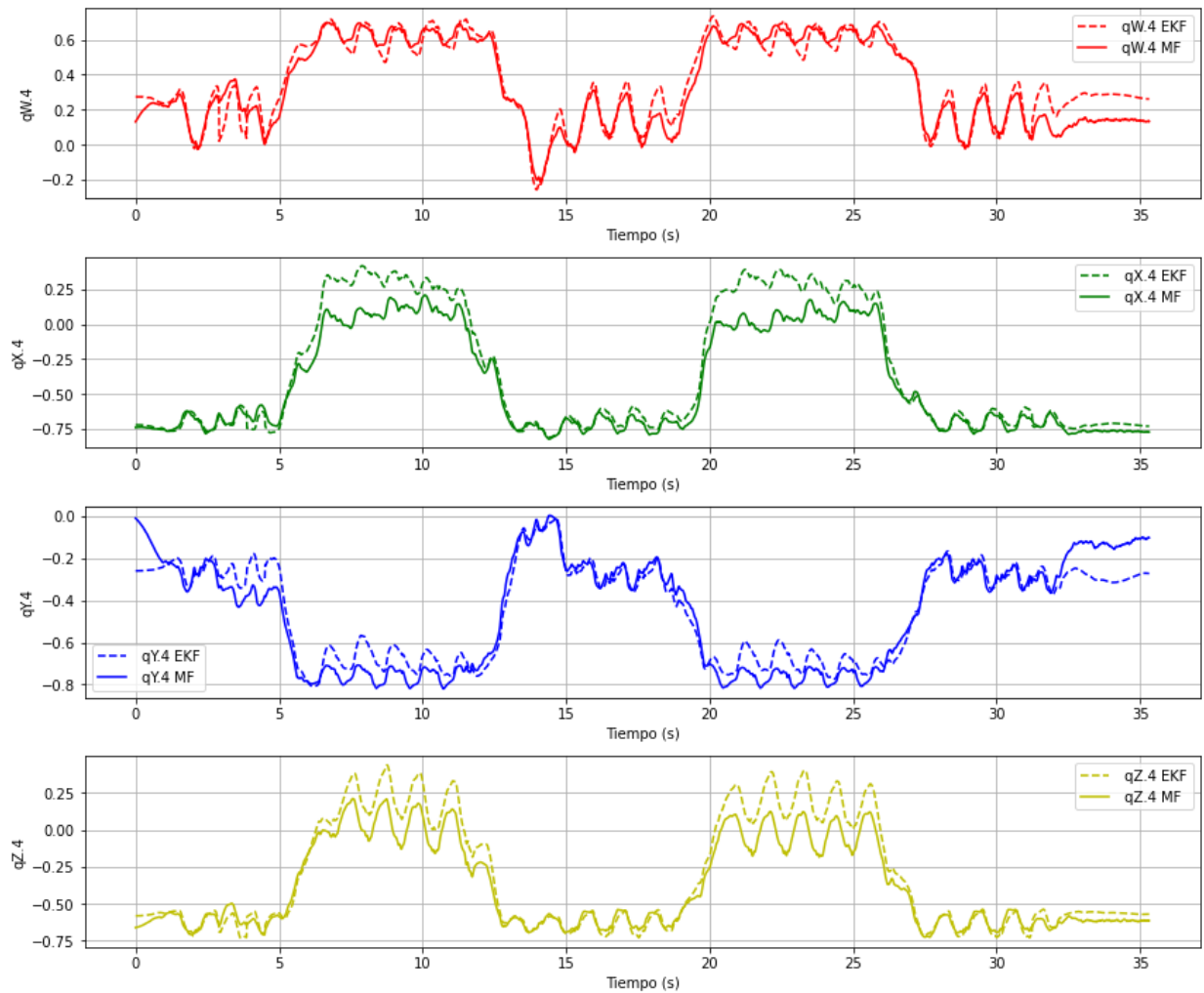


Figura 34: Comparación entre la señal filtrada con el Filtro de Madgwick y el Filtro de Kalman Extendido. Captura de IMU en el pie derecho de muestra de marcha.

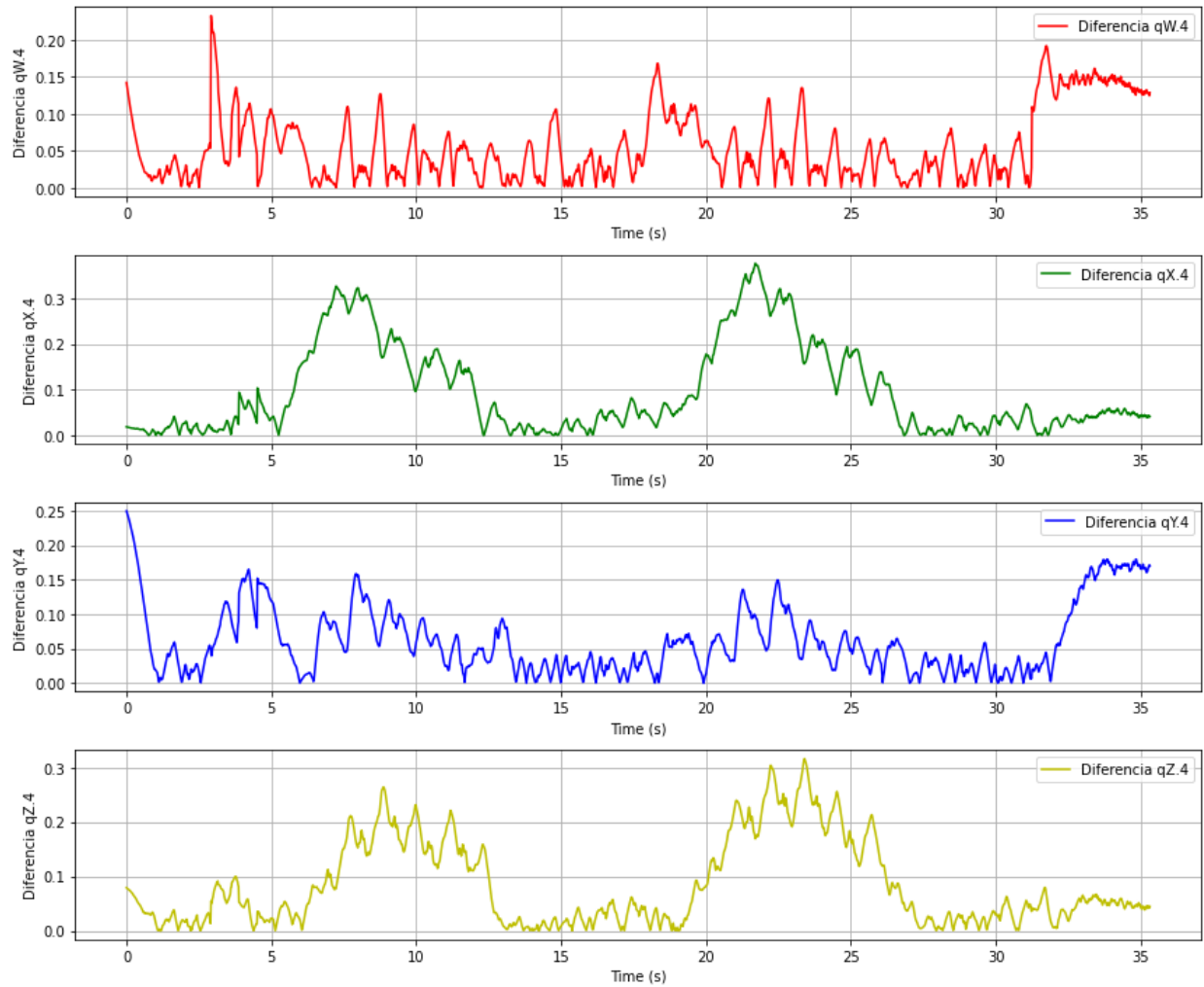


Figura 35: Diferencia entre la señal filtrada con el Filtro de Madgwick y el Filtro de Kalman Extendido. Captura de IMU en el pie derecho de muestra de marcha.

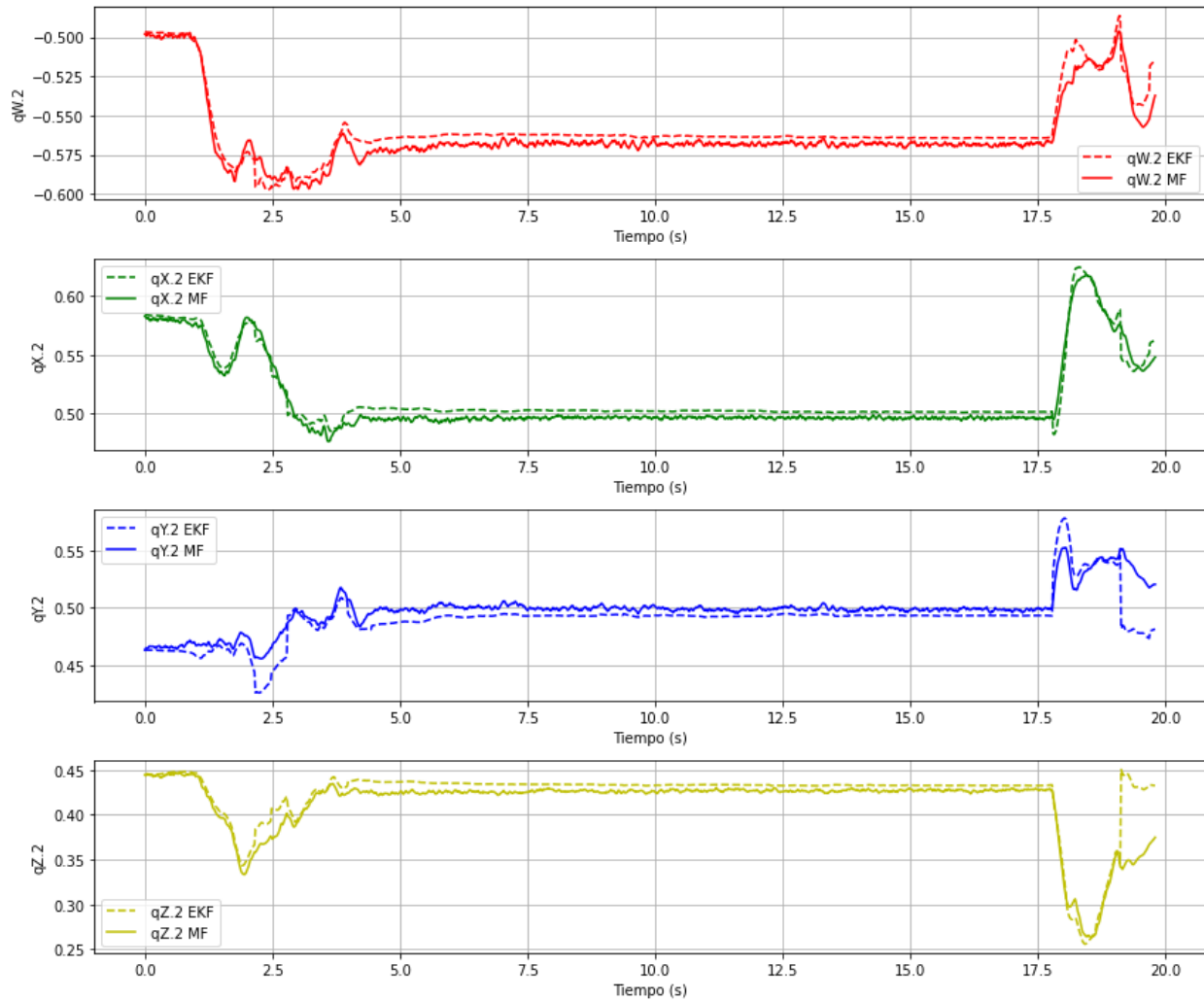


Figura 36: Comparación entre la señal filtrada con el Filtro de Madgwick y el Filtro de Kalman Extendido. Captura de IMU en el muslo derecho de muestra de sujeto sentándose.

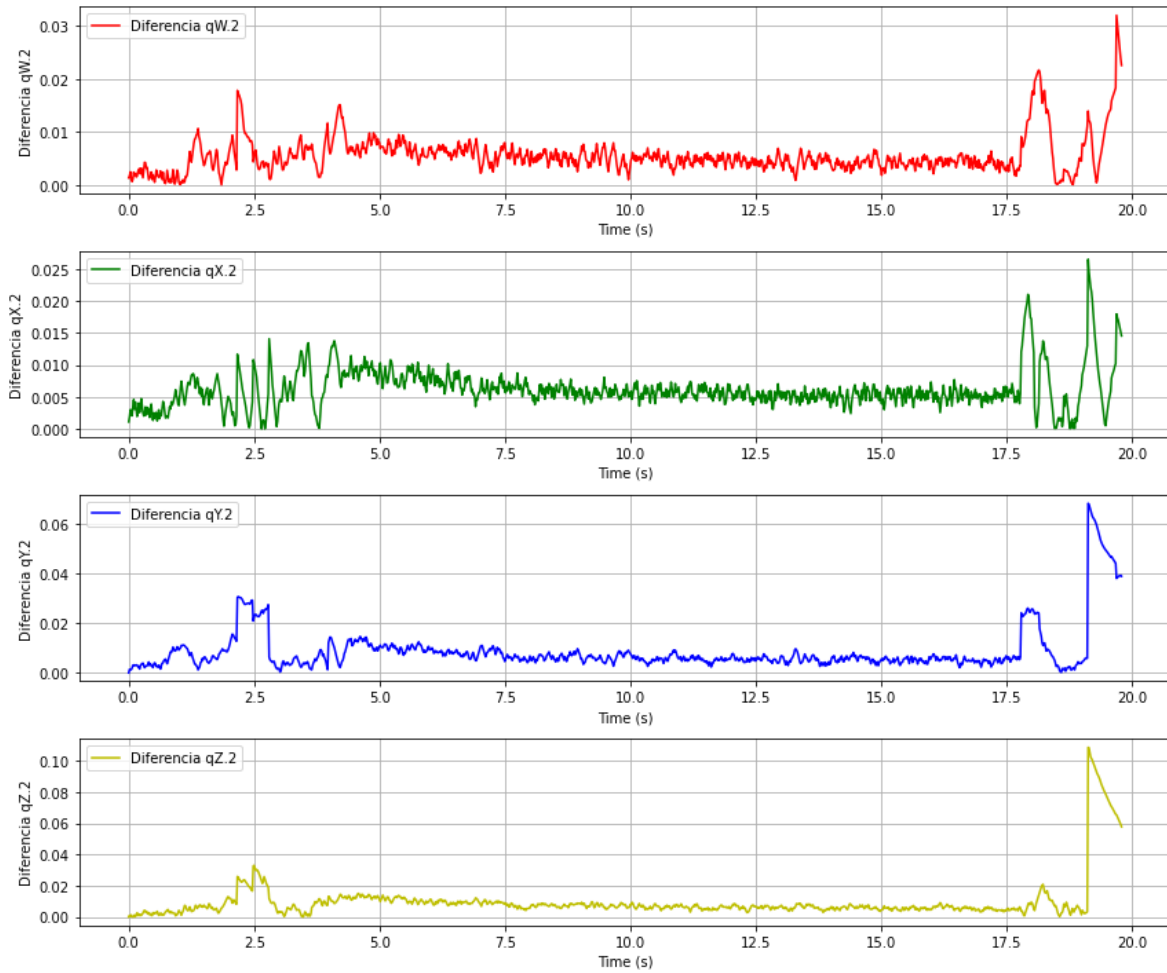


Figura 37: Diferencia entre la señal filtrada con el Filtro de Madgwick y el Filtro de Kalman Extendido. Captura de IMU en el muslo derecho de muestra de sujeto sentándose.

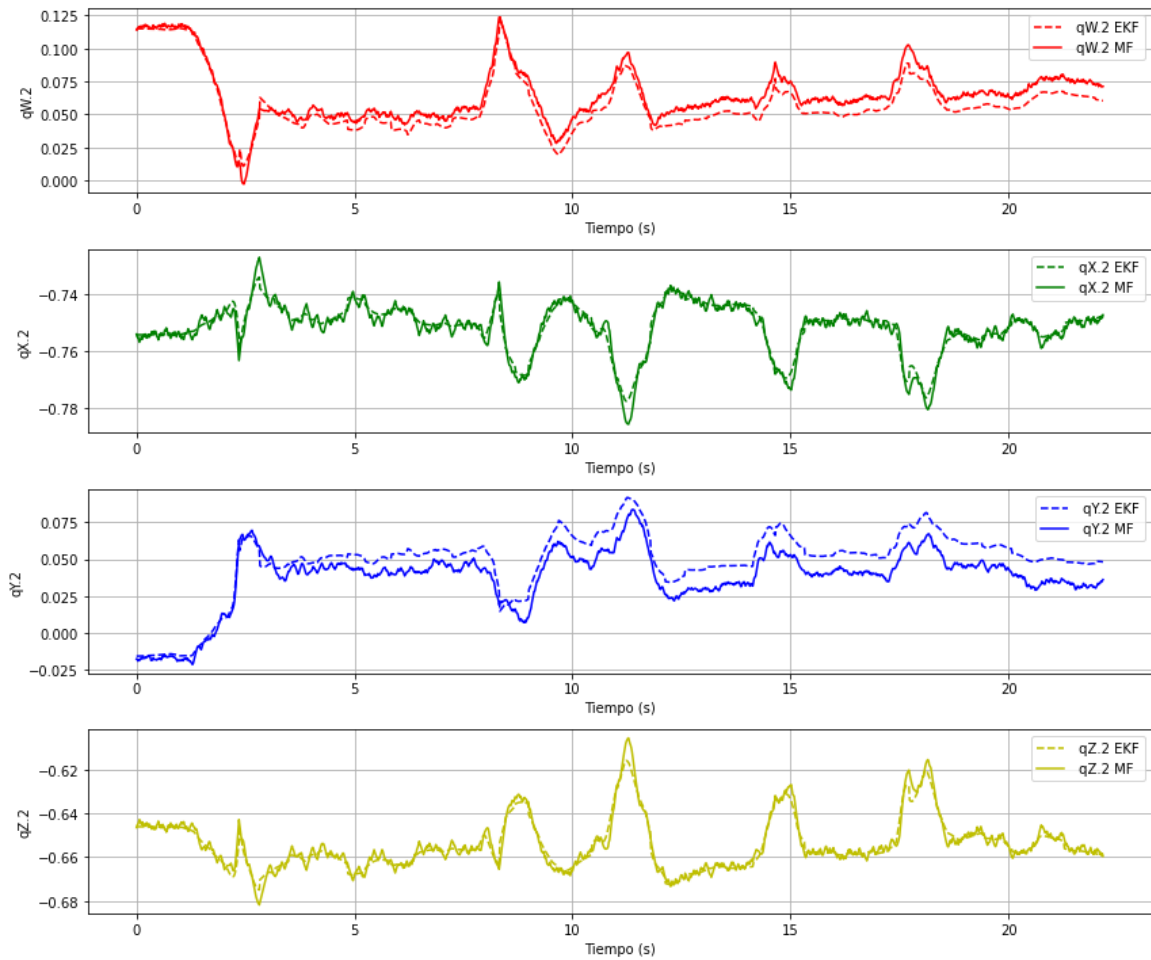


Figura 38: Comparación entre la señal filtrada con el Filtro de Madgwick y el Filtro de Kalman Extendido. Captura de IMU en el muslo derecho de muestra de sujeto parado de pie.

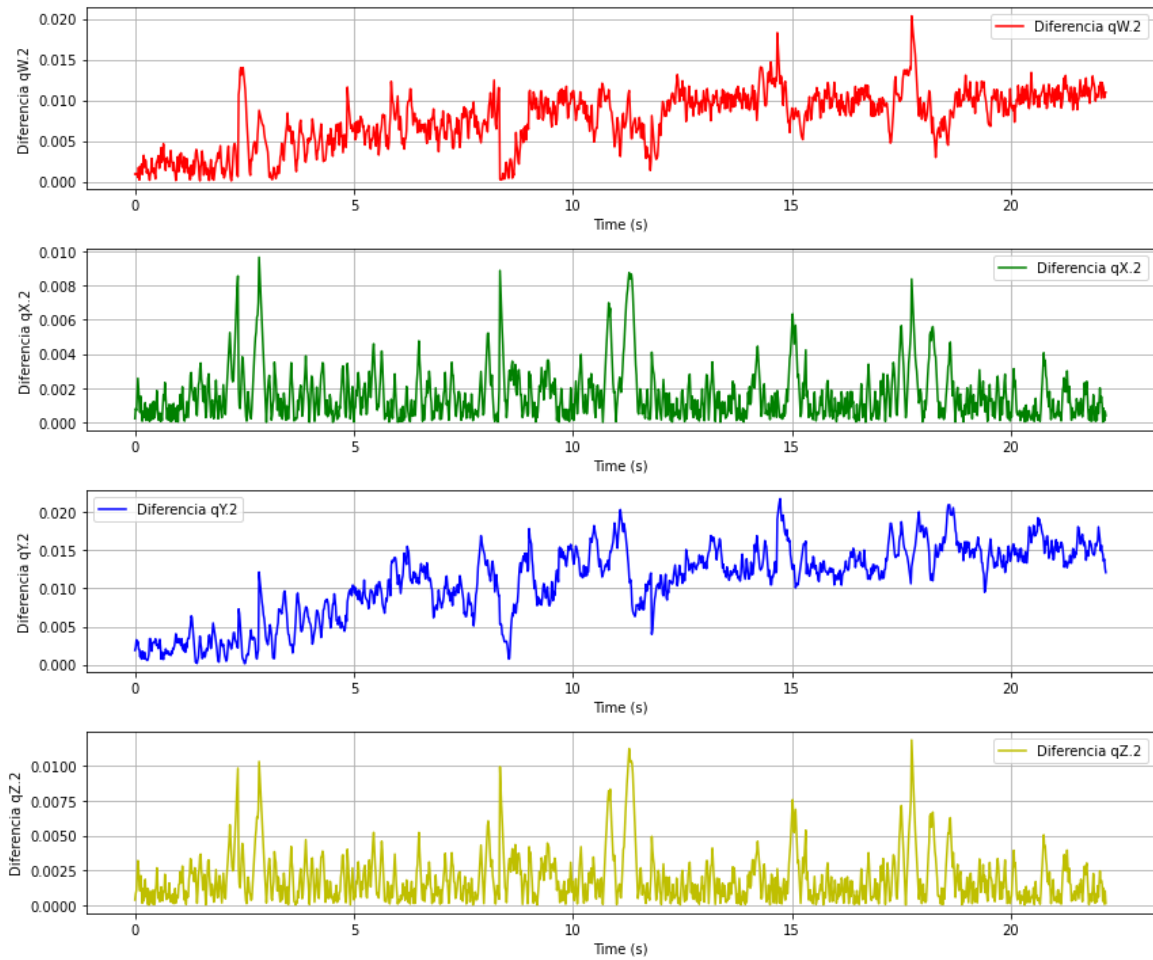


Figura 39: Diferencia entre la señal filtrada con el Filtro de Madgwick y el Filtro de Kalman Extendido. Captura de IMU en el muslo derecho de muestra de sujeto parado de pie.

El siguiente paso fue pasar los cuaterniones a ángulos de Euler. Para ello se usaron las funciones vistas en el apartado 3.2. Filtro de Madgwick Sin embargo, se tuvo que solucionar el problema de discontinuidades de salto que se pueden observar en la Figura 40.

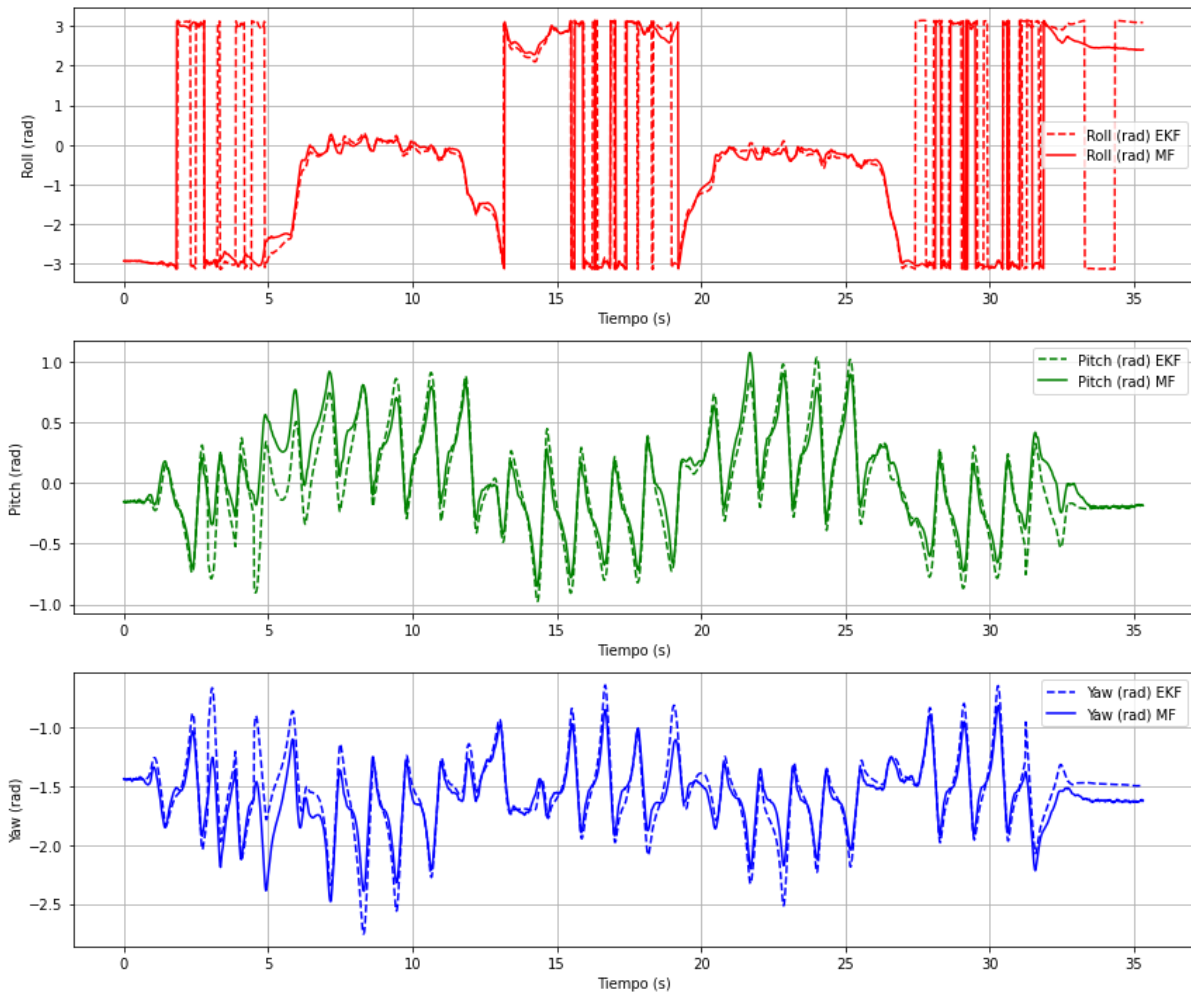


Figura 40: Representación de ángulos de Euler en radianes con problema de discontinuidades de salto. Captura de IMU en el muslo derecho de muestra de marcha.

Para solucionar este problema se usó la función *unwrap* de la librería Numpy de Python [54]. Estas discontinuidades, conocidas como "saltos de 2π ", se producen porque los ángulos en radianes se representan cíclicamente en el intervalo $[-\pi, \pi]$ o $[0, 2\pi]$. Por ejemplo, un ángulo que aumenta linealmente puede "saltarse" de $-\pi$ a π , o viceversa, debido a esta representación cíclica. La función *unwrap* ajusta estos saltos sumando o restando 2π cuando se detecta una discontinuidad mayor que π , suavizando así la transición entre los valores angulares. Al aplicar *unwrap*, se asegura que los ángulos cambien suavemente sin saltos bruscos, proporcionando una representación más precisa y comprensible de los datos angulares.

Además, se llevó a cabo la conversión de radianes a grados ya que estos son una unidad de medida más intuitiva para los ángulos. Se usó la función *degrees* de Numpy para hacerlo de forma automática. Los resultados finales se pueden ver en la Figura 41, 42, 43 y 44.

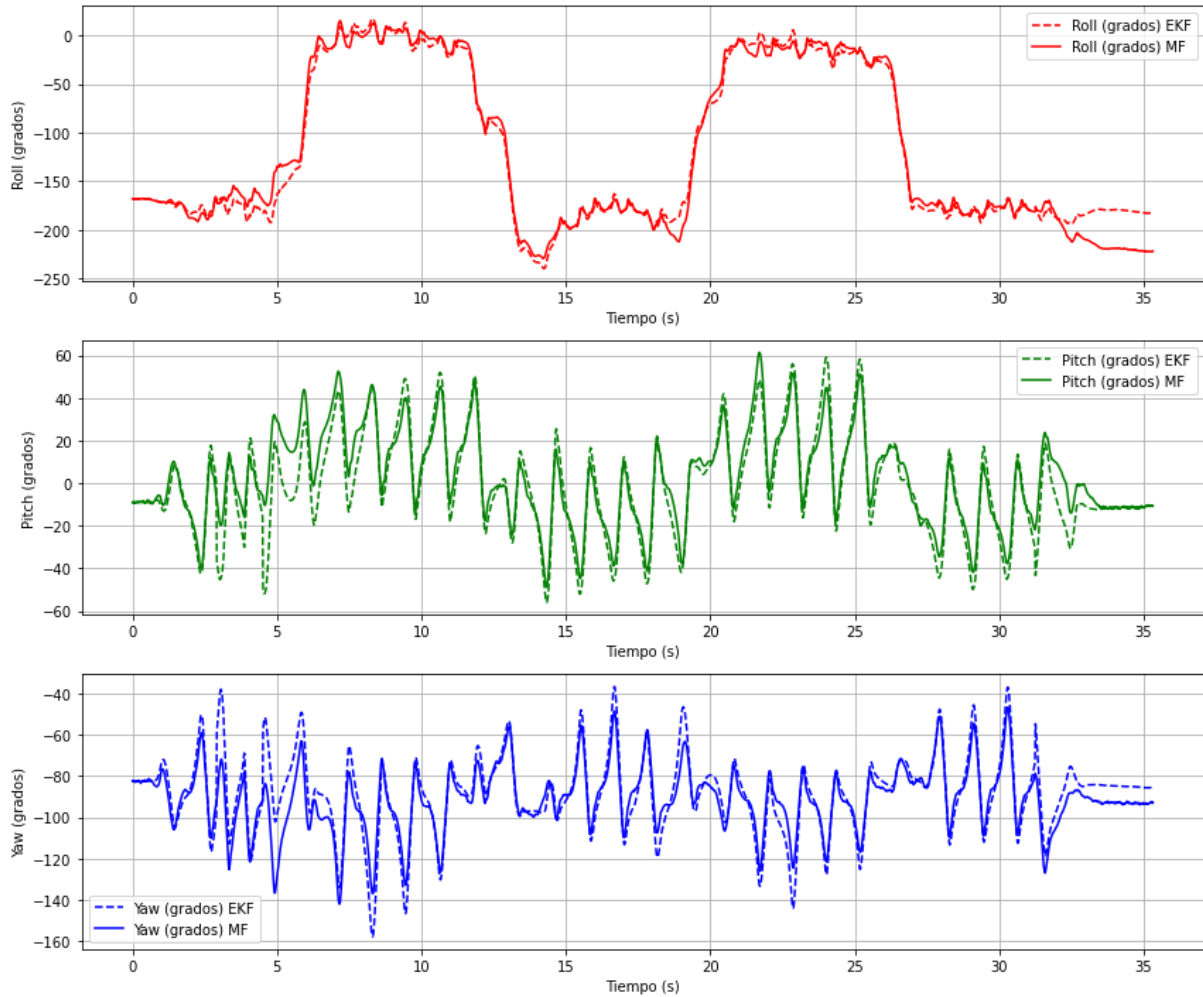


Figura 41: Representación de los ángulos de Euler en grados. Captura de IMU en el muslo derecho de muestra de marcha.



Figura 42: Diferencia de los ángulos de Euler en grados. Captura de IMU en el muslo derecho de muestra de marcha.

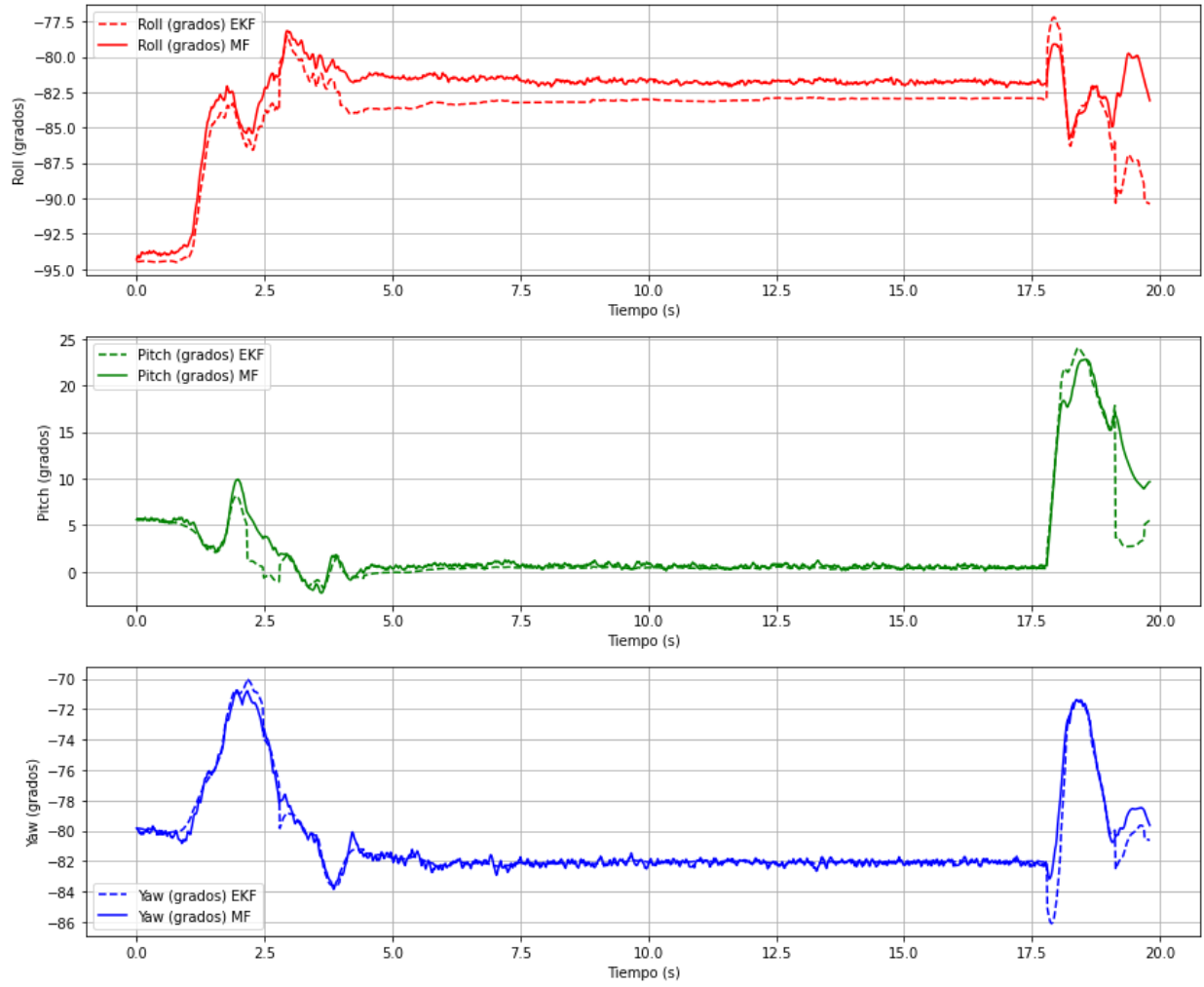


Figura 43: Representación de los ángulos de Euler en grados. Captura de IMU en el muslo derecho de muestra de sujeto sentándose.

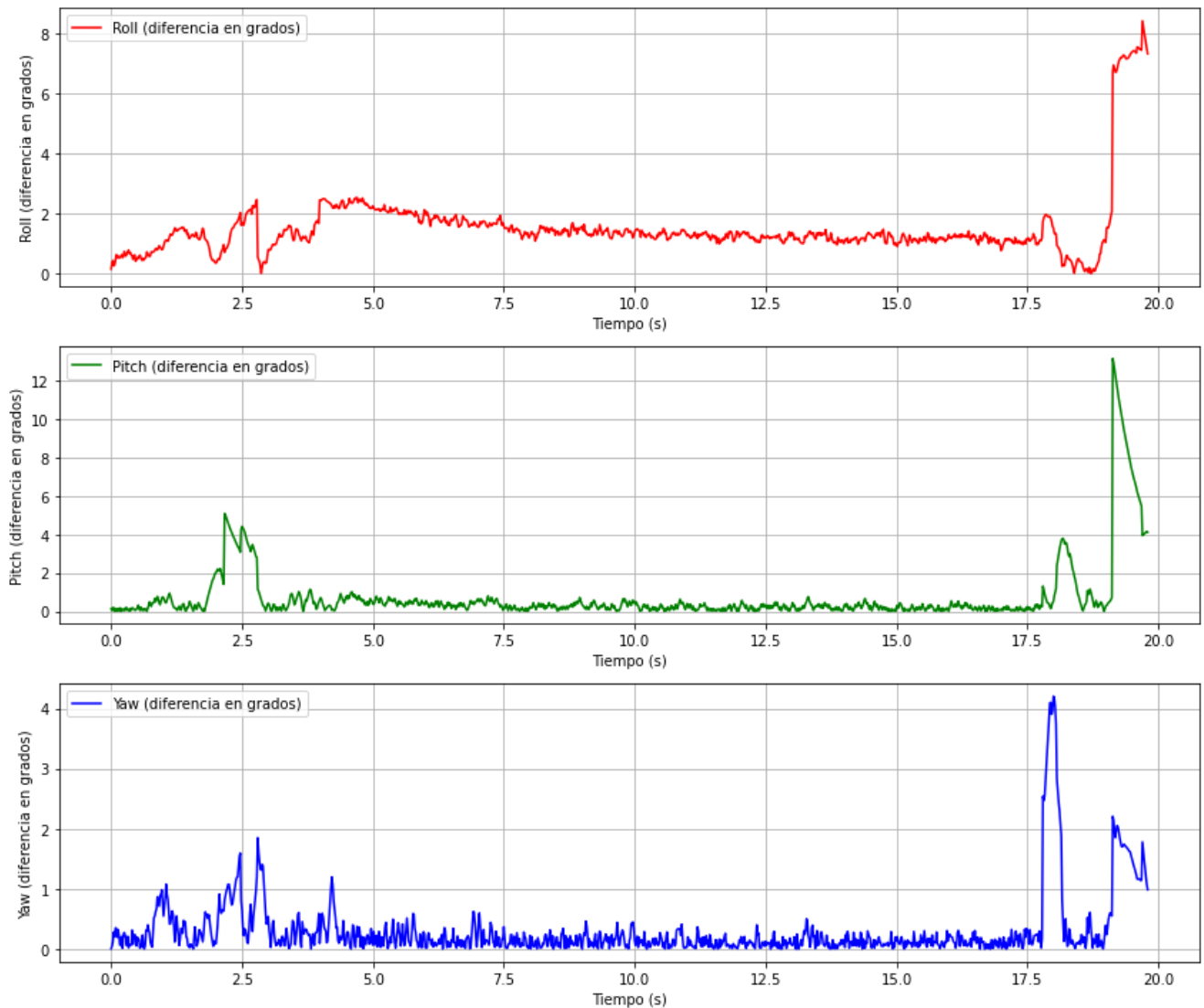


Figura 44: Diferencia de los ángulos de Euler en grados. Captura de IMU en el muslo derecho de muestra de sujeto sentándose.

5.4. Discusión

Uno de los aspectos más desafiantes de este proyecto ha sido lograr una comparación precisa de los resultados obtenidos con el Filtro de Madgwick y el Filtro de Kalman Extendido para evaluar la precisión de ambos. A lo largo de este estudio, los resultados indican que el Filtro de Madgwick es una

alternativa viable y prometedora al EKF en la estimación de ángulos para exoesqueletos de rehabilitación.

Una de las conclusiones más destacables es que el Filtro de Madgwick ha demostrado un desempeño comparable al del EKF en términos de precisión en la estimación de ángulos. Esto es significativo, dado que el EKF es un algoritmo bien establecido y ampliamente estudiado en aplicaciones que requieren alta precisión y robustez.

Si partimos de la premisa de que la estimación con el Filtro de Madgwick es al menos tan funcional como con el bien estudiado EKF, y sumamos las ventajas vistas en el apartado 3.3, como una teórica mejor velocidad de procesamiento y una menor exigencia en cuanto a recursos computacionales, se puede llegar a la conclusión preliminar de que el Filtro de Madgwick es realmente una alternativa viable y útil para su aplicación en dispositivos de menores prestaciones y de bajo coste, haciendo posible que sea más accesible en exoesqueletos de rehabilitación.

Respecto a los aspectos negativos, uno de los mayores desafíos ha sido obtener el cuaternión inicial necesario para asegurar una convergencia adecuada del sistema. Este cuaternión depende de la orientación inicial del sensor, un valor que es difícil de determinar sin un sistema de captura propio. Sin embargo, se anticipa que, en una situación real, con todo el equipo necesario, este valor pueda ser obtenido de manera más accesible.

También se observa que los mayores errores se encuentran principalmente en los resultados de los ángulos de Euler (ver Figura 42 y 44). Esto se debe a los errores acumulados durante la transformación de cuaterniones a ángulos de Euler. Sin embargo, esta diferencia no debería representar un problema significativo, ya que las estimaciones se realizan utilizando únicamente cuaterniones.

Capítulo 6

Conclusiones

En este trabajo se ha investigado y evaluado la aplicación del Filtro de Madgwick en la estimación de ángulos del tren inferior para la rehabilitación con exoesqueletos. A lo largo del estudio, se ha demostrado que el Filtro de Madgwick puede igualar la precisión del Filtro de Kalman Extendido mientras ofrece ventajas significativas en términos de eficiencia computacional y simplicidad de implementación. Esto es particularmente relevante en aplicaciones de tiempo real, donde la eficiencia y la capacidad de respuesta son esenciales.

Se ha logrado implementar un algoritmo que integra eficazmente los datos de acelerómetros, giroscopios y magnetómetros, proporcionando estimaciones precisas y estables de los ángulos del tren inferior. Esta capacidad es crucial para mejorar la interacción entre el exoesqueleto y el usuario, facilitando movimientos naturales y reduciendo el riesgo de posturas incómodas o movimientos forzados.

Para futuras investigaciones, se podrían explorar varias mejoras y extensiones. Por ejemplo, la integración de técnicas de aprendizaje automático podría mejorar aún más la precisión de la estimación de ángulos y la adaptabilidad del sistema a diferentes usuarios y condiciones. Además, se podría investigar la integración del cálculo del centro de gravedad para mejorar la estabilidad y el equilibrio del usuario, lo que es esencial en la rehabilitación motora.

Otra posible mejora sería la optimización de los parámetros del filtro en tiempo real, adaptándose dinámicamente a los cambios en las condiciones del entorno y en el estado del usuario. También sería interesante realizar estudios clínicos para validar el rendimiento del Filtro de Madgwick en escenarios de rehabilitación reales, evaluando su impacto en la recuperación funcional de los pacientes.

En resumen, este trabajo ha proporcionado una base para el uso del Filtro de Madgwick en la estimación de ángulos para exoesqueletos de rehabilitación, abriendo nuevas posibilidades para mejorar la calidad de vida de personas con discapacidades motoras. Las futuras investigaciones y desarrollos basados en estos hallazgos podrán seguir avanzando en esta dirección, contribuyendo al progreso de la rehabilitación asistida por tecnología.

Referencias

1. Encuesta de Discapacidad, Autonomía personal y Situaciones de Dependencia (EDAD) Principales resultados. Año 2020. Ine.es. https://www.ine.es/prensa/edad_2020_p.pdf
2. La atención rehabilitadora en régimen hospitalaria es una de las mayores carencias de la Sanidad española. Sermef.es. <https://www.sermef.es/la-atencion-rehabilitadora-en-regimen-hospitalaria-es-una-de-las-mayores-carencias-de-la-sanidad-espanola/>
3. Aach, M., Cruciger, O., Sczesny-Kaiser, M., Höffken, O., Meindl, R. C., Tegenthoff, M., Schwenkreis, P., Sankai, Y., & Schildhauer, T. A. (2014). Voluntary driven exoskeleton as a new tool for rehabilitation in chronic spinal cord injury: a pilot study. *The Spine Journal: Official Journal of the North American Spine Society*, 14(12), 2847–2853. <https://doi.org/10.1016/j.spinee.2014.03.042>
4. Iqbal, J., & Baizid, K. (2015). Stroke rehabilitation using exoskeleton-based robotic exercisers: mini review. *Biomedical Research*, 26(1), 197-201.
5. Fisahn, C., Aach, M., Jansen, O., Moisi, M., Mayadev, A., Pagarigan, K. T., Dettori, J. R., & Schildhauer, T. A. (2016). The effectiveness and safety of exoskeletons as assistive and rehabilitation devices in the treatment of neurologic gait disorders in patients with Spinal Cord Injury: A systematic review. *Global Spine Journal*, 6(8), 822–841. <https://doi.org/10.1055/s-0036-1593805>
6. Li, R. Y. M., & Ng, D. P. L. (2018). Wearable robotics, industrial robots and construction worker's safety and health. En *Advances in Intelligent Systems and Computing* (pp. 31–36). Springer International Publishing.
7. Industrial exoskeletons: What you're not hearing -- occupational health & safety. (s/f). *Occupational Health & Safety*. <https://ohsonline.com/Articles/2018/10/01/Industrial-Exoskeletons-What-Youre-Not-Hearing.aspx?Page=1>

8. de Looze, M. P., Bosch, T., Krause, F., Stadler, K. S., & O’Sullivan, L. (2017). Exoskeletons for industrial application and their potential effects on physical work load [dataset].
9. Bogue, R. (2022). Exoskeletons: a review of recent progress. *The Industrial Robot*, 49(5), 813–818. <https://doi.org/10.1108/ir-04-2022-0105>
10. Zhou, J., Yang, S., & Xue, Q. (2021). Lower limb rehabilitation exoskeleton robot: A review. *Advances in Mechanical Engineering*, 13(4), 168781402110118. <https://doi.org/10.1177/16878140211011862>
11. Shi, D., Zhang, W., Zhang, W., & Ding, X. (2019). A review on lower limb rehabilitation exoskeleton robots. *Chinese Journal of Mechanical Engineering*, 32(1). <https://doi.org/10.1186/s10033-019-0389-8>
12. Technaid. Exoesqueleto robótico – Exo-H3. <https://www.technaid.com/es/productos/robotic-exoskeleton-exo-h3/>
13. Lin, P.-C., Lu, J.-C., Tsai, C.-H., & Ho, C.-W. (2012). Design and implementation of a nine-axis inertial measurement unit. *IEEE/ASME transactions on mechatronics*, 17(4), 657–668. <https://doi.org/10.1109/tmech.2011.2111378>
14. Zhang, Z., Zhang, X., Pan, H., Salman, W., Rasim, Y., Liu, X., Wang, C., Yang, Y., & Li, X. (2017). A novel steering system for a space-saving 4WS4WD electric vehicle: Design, modeling, and road tests. *IEEE transactions on intelligent transportation systems: a publication of the IEEE Intelligent Transportation Systems Council*, 18(1), 114–127. <https://doi.org/10.1109/tits.2016.2561626>
15. Abdellatif, H., & Heimann, B. (2010). Advanced model-based control of a 6-DOF hexapod robot: A case study. *IEEE/ASME transactions on mechatronics*, 15(2), 269–279. <https://doi.org/10.1109/tmech.2009.2024682>

16. Tan, U.-X., Veluvolu, K. C., Latt, W. T., Shee, C. Y., Riviere, C. N., & Ang, W. T. (2008). Estimating displacement of periodic motion with inertial sensors. *IEEE sensors journal*, 8(8), 1385–1388. <https://doi.org/10.1109/jsen.2008.917488>
17. Shao, T., Wang, Y., Jiang, P., Wu, C., Li, G., Wang, L., & Du, S. (2021). A preliminary study on an IMU-based surgery assisting method for minimally invasive puncture of intracranial hematoma. 2021 IEEE International Conference on Real-time Computing and Robotics (RCAR), 1323–1328.
18. Alahakone, A. U., & Senanayake, S. M. N. A. (2010). A real-time system with assistive feedback for postural control in rehabilitation. *IEEE/ASME transactions on mechatronics*, 15(2), 226–233. <https://doi.org/10.1109/tmech.2010.2041030>
19. Zhao, J. (2018). A review of wearable IMU (inertial-measurement-unit)-based pose estimation and drift reduction technologies. *Journal of Physics. Conference Series*, 1087, 042003. <https://doi.org/10.1088/1742-6596/1087/4/042003>
20. Ojeda, L., & Rodríguez, A. (2022). Review of the state-of-the-art in bipedal exoskeletons for lower-limb rehabilitation. *Robotics*, 11(1), 15. <https://doi.org/10.3390/robotics11010015>
21. Rasmussen, D. H. (2019). Quaternion kinematics for the error-state Kalman filter (Technical Report No. NPS-2019-0009). Naval Postgraduate School. Recuperado de <https://calhoun.nps.edu/server/api/core/bitstreams/7f2ba116-fe11-410f-80aa-58c1d7e37222/content>
22. Euler Angles Introduction. Academic Flight.. Euler angles. <https://academicflight.com/articles/kinematics/rotation-formalisms/euler-angles/>
23. Teorema de rotación de Euler. <https://www.wikiwand.com/es/Teorema de rotación de Euler>
24. Goldman, R. (2011). Understanding quaternions. *Graphical Models*, 73(2), 21–49. <https://doi.org/10.1016/j.gmod.2010.10.004>
25. Mit.edu. <https://web.mit.edu/2.05/www/Handout/HO2.PDF>
26. Weisstein, E. W. . Rotation matrix. <https://mathworld.wolfram.com/RotationMatrix.html>

27. Strickland, J. (2023, noviembre 30). What is a gimbal -- and what does it have to do with NASA? HowStuffWorks. <https://science.howstuffworks.com/gimbal.htm>
28. Quiroz, B. B. S., Christelle, G., & Lesecq, S. (s/f). Fusión de Datos de Multicaptore para la Captura de Movimiento. Hal.science. Recuperado el 29 de junio de 2024, de <https://hal.science/hal-00160745/document>
29. Wilkins, E. by D. (s/f). *William Rowan Hamilton*. Tcd.ie., <https://www.maths.tcd.ie/pub/HistMath/People/Hamilton/Quatern2/Quatern2.pdf>
30. Jia, Y. (n.d.). Quaternions and rotation sequences: A primer with applications to orbits, aerospace, and virtual reality. Recuperado el 23 de junio de 2024, de <https://faculty.sites.iastate.edu/jia/files/inline-files/quaternion.pdf>
31. Kalman, R. E. (1960). A new approach to linear filtering and prediction problems. *Journal of Basic Engineering*, 82(1), 35–45. <https://doi.org/10.1115/1.3662552>
32. Callejo, D., López, F., Pérez, J., & Vidal, J. M. V. (s/f). e-REDING. Biblioteca de la Escuela Superior de Ingenieros de Sevilla. Biblus.us.es. Recuperado el 29 de junio de 2024, de <https://biblus.us.es/bibing/proyectos/abreproy/11611/fichero/Memoria%252FApéndice+B+-+El+Filtro+de+Kalman.pdf+>
33. Wikipedia contributors. (2024, junio 21). Kalman filter. Wikipedia, The Free Encyclopedia. https://en.wikipedia.org/w/index.php?title=Kalman_filter&oldid=1230233207
34. Keisuke Fujii The ACFA-Sim-J Group. (s/f). *Extended Kalman Filter*. Kek.jp. Recuperado el 1 de julio de 2024, de <https://www-jlc.kek.jp/2004sep/subg/offl/kaltest/doc/ReferenceManual.pdf>
35. Sabatini, A. M. (2006). Quaternion-based extended Kalman filter for determining orientation by inertial and magnetic sensing. *IEEE Transactions on Bio-Medical Engineering*, 53(7), 1346–1356. <https://doi.org/10.1109/tbme.2006.875664>
36. Barfoot, T., Forbes, J. R., & Furgale, P. T. (2011). Pose estimation using linearized rotations and quaternion algebra. *Acta Astronautica*, 68(1–2), 101–112. <https://doi.org/10.1016/j.actaastro.2010.06.049>

37. Potter, M. V., Cain, S. M., Ojeda, L. V., Gurchiek, R. D., McGinnis, R. S., & Perkins, N. C. (2022). Evaluation of error-state Kalman filter method for estimating human lower-limb kinematics during various walking gaits. *Sensors (Basel, Switzerland)*, 22(21), 8398. <https://doi.org/10.3390/s22218398>
38. Potter, M. V., Cain, S. M., Ojeda, L. V., Gurchiek, R. D., McGinnis, R. S., & Perkins, N. C. (2021). Error-state Kalman filter for lower-limb kinematic estimation: Evaluation on a 3-body model. *PLoS One*, 16(4), e0249577. <https://doi.org/10.1371/journal.pone.0249577>
39. Ghorbani, E., Dollon, Q., & Gosselin, F. P. (2024). Physics-aware tuning of the unscented Kalman filter: statistical framework for solving inverse problems involving nonlinear dynamical systems and missing data. *Nonlinear Dynamics*. <https://doi.org/10.1007/s11071-024-09760-z>
40. Scardua, L. A., & da Cruz, J. J. (2016). Particle-based tuning of the unscented Kalman filter. *Journal of Control Automation and Electrical Systems*, 27(1), 10–18. <https://doi.org/10.1007/s40313-015-0223-1>
41. Piray, P., & Daw, N. D. (2020). A simple model for learning in volatile environments. *PLoS Computational Biology*, 16(7), e1007963. <https://doi.org/10.1371/journal.pcbi.1007963>
42. Li, X., Wang, Y., & Khoshelham, K. (2018). A robust and adaptive Complementary Kalman Filter based on Mahalanobis distance for ultra wideband/Inertial Measurement Unit fusion positioning. *Sensors (Basel, Switzerland)*, 18(10), 3435. <https://doi.org/10.3390/s18103435>
43. Madgwick, S. O. H. (s/f). *An efficient orientation filter for inertial and inertial/magnetic sensor arrays*. X-io.co.uk. https://x-io.co.uk/downloads/madgwick_internal_report.pdf
44. Madgwick orientation filter — AHRS 0.4.0 documentation. Readthedocs.Io. <https://ahrs.readthedocs.io/en/latest/filters/madgwick.html>
45. Cirillo, A., Cirillo, P., De Maria, G., Natale, C., Pirozzi, S., Fourati, H., & Belkhiat, D. (2016). A comparison of multisensor attitude estimation algorithms. *Multisensor Attitude Estimation: Fundamental Concepts and Applications*, 1, 529-540.

46. Böer, J. Madgwickahrs.Py at master morgil/madgwick_py. https://github.com/morgil/madgwick_py/blob/master/madgwickahrs.py.
47. Segarra, D., Caballeros, J., Aguilar, W. G., Samà, A., & Rodríguez-Martín, D. (2019). Orientation estimation using filter-based inertial data fusion for posture recognition. En Algorithms for Sensor Systems (pp. 220–233). Springer International Publishing.
48. Huang, Y., Kaufmann, M., Aksan, E., Black, M. J., Hilliges, O., & Pons-Moll, G. (2018). Deep inertial poser: Learning to reconstruct human pose from sparse inertial measurements in real time. <https://doi.org/10.48550/ARXIV.1810.04703>
49. Huang, Y., Kaufmann, M., Aksan, E., Black, M. J., Hilliges, O., & Pons-Moll, G. (2018). Deep Inertial Poser: Learning to Reconstruct Human Pose from Sparse Inertial Measurements in Real Time. <https://github.com/eth-ait/dip18>
50. Velocities in robotics: Angular velocities & twists. Mecharithm. <https://mecharithm.com/learning/lesson/velocities-in-robotics-angular-velocities-twists-10>
51. Luo, Y., Coppola, S. M., Dixon, P. C., Li, S., Dennerlein, J. T., & Hu, B. (2020). A database of human gait performance on irregular and uneven surfaces collected by wearable sensors. Scientific Data, 7(1). <https://doi.org/10.1038/s41597-020-0563-y>
52. Orlando, J. (2017). Manual de usuario Tech-MCS V.3.0. Technaid.
53. Software para Captura de Movimiento. (2016, mayo 2). Technaid. <https://www.technaid.com/es/productos/motion-capture-software/>
54. Numpy.Unwrap — NumPy v2.0 manual. Numpy.org. <https://numpy.org/doc/stable/reference/generated/numpy.unwrap.html>