



Trabajo Fin de Máster

Diseño del control de par adaptativo
de un exoesqueleto robotizado
para rehabilitación motora

Autor

Violeta Rioseras Aurensanz

Director

D. Luis Montano Gella

Máster Universitario en Ingeniería Biomédica
Escuela de Ingeniería y Arquitectura
Universidad de Zaragoza

Noviembre 2018

*Agradecimientos a mi familia, amigos, tutor,
compañeros de laboratorio y de máster por hacerlo posible.*



DECLARACIÓN DE AUTORÍA Y ORIGINALIDAD

(Este documento debe acompañar al Trabajo Fin de Grado (TFG)/Trabajo Fin de Máster (TFM) cuando sea depositado para su evaluación).

D./Dª. Violeta Rioseras Aurensanz

con nº de DNI 17.762.751N en aplicación de lo dispuesto en el art.

14 (Derechos de autor) del Acuerdo de 11 de septiembre de 2014, del Consejo de Gobierno, por el que se aprueba el Reglamento de los TFG y TFM de la Universidad de Zaragoza,

Declaro que el presente Trabajo de Fin de (Grado/Máster) Máster, (Título del Trabajo)

Diseño del control de par adaptativo de un exoesqueleto robotizado para rehabilitación motora

es de mi autoría y es original, no habiéndose utilizado fuente sin ser citada debidamente.

Zaragoza, a 23 de Noviembre de 2018

Fdo: Violeta Rioseras Aurensanz

Diseño del control de par adaptativo de un exoesqueleto robotizado para rehabilitación motora

Resumen

El presente Trabajo Fin de Máster, se enmarca en el proyecto HYPER-Consolider, en el Grupo de Robótica de la Universidad de Zaragoza. La finalidad de este proyecto es diseñar un control de par adaptativo a pares externos y su posterior implementación en el sistema físico de un exoesqueleto para ayudar en la rehabilitación de miembros inferiores de pacientes que han sufrido un accidente cerebrovascular. El clásico control de posición de los exoesqueletos, un control rígido, es sustituido por un control de par adaptativo flexible, que permite adaptar el movimiento del exoesqueleto a fuerzas o pares externos. Estos pares externos pueden ser proporcionados por la persona que usa el exoesqueleto y al que debe adaptarse o pueden provenir del par ejercido por un rehabilitador si se utiliza para ejercicios de rehabilitación motora. Se diseñan y evalúan diversas situaciones de asistencia del robot y de contribuciones de un sujeto de 60 Kg. de acuerdo a su capacidad residual estimada, desde una contribución parcial del paciente a una contribución casi total.

Índice

1	Introducción	1
1.1	Motivación	1
1.2	Objetivos del proyecto	3
1.3	Organización de la memoria descriptiva	3
2	Exoesqueleto H1 para miembros inferiores	5
2.1	Prototipo H1	5
2.1.1	Características <i>hardware</i> del prototipo H1	6
2.1.2	Características <i>software</i> del prototipo H1	8
2.2	Condiciones actuales del prototipo H1	8
2.3	Conclusiones	9
3	Biomecánica de la marcha humana normal. Controladores de posición y par motor para la articulación de la cadera	11
3.1	Introducción a la biomecánica de la marcha humana normal	11
3.1.1	Análisis cinemático de la marcha normal en el plano sagital para la articulación de la cadera	12
3.1.2	Obtención de la referencia del exoesqueleto en base al análisis cinemático de la marcha normal en el plano sagital	14
3.1.3	Puesta en marcha y calibración del sensor de posición en base al análisis cinemático	14
3.2	Controladores de posición y par motor para la articulación de la cadera	15
3.2.1	Control de posición	16
3.2.2	Control de par motor	18
3.3	Conclusiones	21
4	Control de par adaptativo para la articulación de la cadera	23
4.1	Concepto y esquema general del control de par adaptativo	23
4.2	Diseño del control de par adaptativo	25
4.2.1	Obtención del τ_{PESO}	27

4.2.2 Análisis cinético de la marcha normal en el plano sagital para la articulación de la cadera. Obtención del $\tau_{ARTICULAR}$ mediante un modelo biomecánico computacional	29
4.3 Simulación del control de par adaptativo	30
4.4 Conclusiones	32
5 Identificación para métrica y experimentación del control de par adaptativo en el exoesqueleto H1	34
5.1 Introducción a la identificación paramétrica	34
5.1.1 Identificación paramétrica del exoesqueleto H1	35
5.2 Concepto y esquema general del control de par adaptativo con identificación	37
5.3 Conclusiones	37
6 Conclusiones y líneas futuras	39
A Entorno de control de tiempo real	42
A.1 Aplicación xPC Target (versión 5.5)	42
A.2 Proceso para la obtención rápida del prototipo del exoesqueleto H1	43
A.2.1 Proceso para la obtención del modelo según Host PC de 32 bits	44
A.2.1.1 Instalación, configuración y conexión de los equipos (32 bits)	44
A.2.1.2 Protocolo para poner en marcha el exoesqueleto (32 bits)	47
A.2.2 Proceso para la obtención del modelo según Host PC de 64 bits	49
A.2.2.1 Instalación, configuración y conexión de los equipos (64 bits)	49
A.2.2.2 Protocolo para poner en marcha el exoesqueleto (64 bits)	50
B Protocolo de reinicio del sistema embebido PC104	53
B.1 Descripción y solución del problema	53
B.1.1 Protocolo de reinicio del PC104 ante un error en su disco duro	53
C Obtención de la consigna de par motor	59
C.1 Código para la obtención de la consigna de par motor	59
D Simulación dinámica de la marcha mediante <i>OpenSim</i>. Dinámica inversa	62
D.1 Introducción al entorno de simulación de <i>OpenSim</i> 3.3	62
D.2 Simulación de la dinámica inversa para la articulación de la cadera izquierda durante la marcha normal	62
D.2.1 Protocolo para calcular el momento neto en la articulación de la cadera izquierda	63

D.2.2 Resultados de la cinemática y de la cinética de la marcha normal humana en la articulación de la cadera izquierda	64
E Simulación del control de par adaptativo	67
E.1 Caso ideal	67
F Proceso de identificación paramétrica	76
F.1 Obtención del proceso de identificación paramétrica	76
F.1.1 Adquisición de los datos de entrada – salida del sistema	76
F.1.2 Tratamiento previo de los datos registrados	77
F.1.3 Estimación del modelo y obtención de sus parámetros	78
F.1.4 Evaluación del modelo	78
F.2 Análisis del proceso de identificación paramétrica	78
F.2.1 Identificación paramétrica del exoesqueleto H1 mediante dos señales de entrada	78
F.2.2 Identificación paramétrica simplificada del exoesqueleto H1 mediante dos señales de entrada	79
Bibliografía	81

CAPÍTULO 1

Introducción

1.1 Motivación

La terapia robótica para la rehabilitación de la marcha, en patologías neurológicas, se puede clasificar según su estado de desarrollo y comercialización, o según el tipo de estructura: tipo exoesqueleto, estáticos o portátiles y con sistemas de efector final o sistemas híbridos. Aunque son múltiples los exoesqueletos de rehabilitación de miembros inferiores desarrollados con el mismo propósito de lograr la recuperación de las funciones motoras del paciente con o sin la ayuda de personal médico cualificado, presentan diferentes técnicas de control para este propósito. Por ejemplo, el exoesqueleto HAL [1] en su versión de miembros inferiores posee un control de impedancia y su mecanismo de actuación está basado en señales electromiográficas (EMGs). Por otro lado, el exoesqueleto ALLOR [2], propone una rehabilitación estática de la marcha usando controladores de posición y admitancia basados en la interacción física y no en la interacción cognitiva. En [3] el exoesqueleto ALEX plantea un controlador de campo de fuerza para proporcionar asistencia al paciente, en cambio el exoesqueleto MINA [4] usa un control posición-derivativo de par. En la figura 1.1 se muestra un resumen de diferentes ejemplos exoesqueletos y sus características más relevantes: grados de libertad, peso y altura que son capaces de soportar... etc

Viendo la variabilidad de los casos nombrados anteriormente, el presente Trabajo Fin de Máster (TFM) presenta el diseño de un control de par motor adaptativo (al aportar una mayor flexibilidad a fuerzas externas) para asistir ejercicios de rehabilitación en pacientes que presentan descompensación en la coordinación de la marcha tras sufrir un accidente cerebrovascular. El equipo utilizado para este fin, es una plataforma robótica desarrollada dentro del proyecto HYPER - Consolider Ingenio 2010 [5] destinado a la investigación y desarrollo de dispositivos neurorrobóticos (NR) y neuroprotésicos (NP). El objetivo de HYPER, es lograr la restauración de las funciones motoras en pacientes afectados a través de la compensación funcional y de la rehabilitación, mediante la validación funcional y clínica de sistemas híbridos

robóticos (NR-NP), fomentando el reaprendizaje de dichos trastornos motores, bajo el paradigma de asistencia bajo demanda según necesidades del paciente.

	Bionic Leg	Ekso	HAL	Indego
Exoskeleton				
Degrees of freedom	K	HKA	HKA	HK
Weight-bearing devices	W	C	W/C/S	W/C
Sensor measurements	JA, JT, FF	JA, ACF, FF, Acc/Ori (arm)	EMG, JA, FF, Acc	JA, Acc, Ori
Device weight (kg)	3.6	20	15	12
User height (cm) limit	153–182	158–188	145–185	155–191
User weight (kg) limit	136	100	80	113
Gait initiation mode	Foot sensors and knee extension	1. Body tilt 2. Button push	Knee EMG activation	Body tilt
Unique features	Unilateral	—	—	—

	Kinesis	ReWalk	WalkTrainer	WPAL
Exoskeleton				
Degrees of freedom	KA	HKA	HaHKA	HKA
Weight-bearing devices	W/C	C	S	W/B
Sensor measurements	JA, FF, IT, Ori	JA, FF, Ori	IT, JA	JA, JT
Device weight (kg)	9.2	23	?	13
User height (cm) limit	<1.85	160–190	?	145–180
User weight (kg) limit	90	100	?	80
Gait initiation mode	Button push	Body tilt	?	Button push
Unique features	Hybrid (FES)	—	Hybrid (FES), active bodyweight support suspension harness moves with exoskeleton	Frame fits between legs, easy to don within wheelchair

Figura 1. 1. Ejemplos de exoesqueletos de miembros inferiores y de sus características más relevantes.

1.2 Objetivos del proyecto

La finalidad de este TFM, será el diseño un control de par motor adaptativo a pares externos y su posterior implementación en el sistema físico de un exoesqueleto desarrollado para ayudar en la rehabilitación de miembros inferiores. El principal objetivo, será evaluar y conseguir que el par de asistencia que ejecute la plataforma robótica, se pueda regular en base al par externo que ofrezcan las extremidades de un paciente durante la acción de caminar, que compensará la falta de fuerza necesaria para realizar el ejercicio.

Con este propósito, el clásico control de posición de los exoesqueletos, un control rígido, se sustituirá por un control flexible de par que permitirá adaptar el movimiento del exoesqueleto a fuerzas o pares externos. Éstos, pueden provenir del propio momento articular ejercido por el usuario o del par ejercido por un rehabilitador si se utiliza para ejercicios de rehabilitación motora. Se utilizará un modelo biomecánico para estimar la activación muscular de pacientes con capacidad motora reducida y será aplicado para estimar el par motor que debe ejercer el exoesqueleto para completar el movimiento normal de la marcha.

A partir de los controles de posición y fuerza que incorpora el exoesqueleto se desarrollará un nuevo esquema de control de par motor adaptativo. Éste se implementará en el exoesqueleto y se ajustarán los parámetros de control a los ejercicios a realizar para obtener un comportamiento estable y suave de la marcha humana normal. Se diseñarán y evaluarán diversas situaciones de asistencia del robot, para un sujeto de 60 Kg. de acuerdo a su capacidad residual estimada a partir del modelo biomecánico.

Como principal herramienta de trabajo se utilizará un exoesqueleto de seis grados de libertad que trabaja en el plano sagital y fue desarrollado para realizar ejercicios de rehabilitación de las extremidades inferiores. En concreto, se empleará la versión H1 [6,7] de un exoesqueleto desarrollado con este propósito y enmarcado en el proyecto HYPER, dentro del programa Consolider – Ingenio 2010 (ver apartado 1.1). El modelado de los sistemas de control se desarrollará y analizará utilizando el entorno de desarrollo y simulación *Matlab – Simulink* [8], ambos del grupo *Mathworks*. El modelo biomecánico se obtendrá con *OpenSim* [9] v.3.2, un software de análisis de movimiento humano para generar datos y simulaciones dinámicas músculo - esqueléticas.

El desarrollo ha sido realizado dentro del grupo de investigación de Robótica, Percepción y Tiempo Real (RoPeRT) del Instituto de Investigación de Ingeniería en Aragón (i3A), en el Laboratorio del Instituto.

1.3 Organización de la memoria descriptiva

Este documento está estructurado en seis capítulos que resumen los planteamientos teóricos y los resultados prácticos obtenidos durante el desarrollo del proyecto, además incluye los anexos correspondientes a los problemas surgidos, su resolución y los scripts de *Matlab-Simulink* empleados. El Capítulo 1, presenta la introducción y los objetivos del proyecto, así como un breve estado del arte con algunos ejemplos de exoesqueletos de rehabilitación de miembros inferiores. En el Capítulo 2, se describen los elementos *hardware* y detalles *software* de interés

del exoesqueleto H1 y las condiciones actuales del equipo. En el Capítulo 3, se introduce la biomecánica de la marcha normal humana y se explica la sintonización de los controladores de posición y par motor realizada en este trabajo, para poder desempeñar la acción de caminar, generando movimientos suaves con los controladores de posición y par. En el Capítulo 4, se desarrolla un modelo de control de par adaptativo a partir de la estimación de pares externos, orientado bajo el paradigma de asistencia bajo demanda (*Assist-As-Needed*, AAN), para un sujeto de 60 Kg. Posteriormente en el Capítulo 5, se realiza la identificación paramétrica posición-par del equipo, que se integra el control de par adaptativo del exoesqueleto. De esta manera resulta más sencillo obtener las consignas de par para el control adaptativo de par, a partir únicamente de la definición de las trayectorias, el patrón de marcha o rehabilitación deseado. Finalmente, el Capítulo 6, presenta las conclusiones del proyecto y algunas de las posibles líneas de futuras para poder trabajar con el exoesqueleto en el campo de la rehabilitación motora, o para la marcha de una persona en la vida diaria.

CAPÍTULO 2

Exoesqueleto H1 para miembros inferiores

Este capítulo incluye una descripción general de la principal herramienta de trabajo del presente TFM, el exoesqueleto H1. Se describen los elementos hardware de los que consta el prototipo, así como las conexiones requeridas para el correcto funcionamiento del sistema y otros detalles software de interés. Seguidamente, se exponen las condiciones actuales del mismo para tener una mejor comprensión del posterior trabajo realizado.

2.1 Prototipo H1

La plataforma robótica H1 [6,7] es un exoesqueleto diseñado para rehabilitar la marcha normal humana en pacientes adultos. En concreto, es ideado para pacientes que presentan un déficit en la coordinación de la marcha ambulatoria tras sufrir un accidente cerebrovascular, pero que pueden estabilizar el tronco con cierto nivel de equilibrio.

La terapia de rehabilitación de la marcha, debe realizarse mediante ejercicios de asistencia dentro de un entorno clínico y bajo la constante supervisión de personal médico capacitado. Este dispositivo (figura 2.1) posee seis grados de libertad, lo que permite que las articulaciones de cadera, rodilla y tobillo de ambas piernas, puedan realizar ejercicios pasivos y activos de flexión y extensión en el plano sagital de la marcha (figura 3.2.b).

Diferentes estrategias de control [6,7] son desarrolladas con idea de que el entrenamiento de la marcha asistido por el exoesqueleto H1 utilice el principio de terapia de asistencia bajo demanda, más conocido como paradigma *Assist-As-Needed* (AAN). Este método hace que el robot no asista de manera continuada todo el movimiento de rehabilitación, si no que aporte la asistencia necesaria demandada por el miembro inferior afectado. De esta manera, y a diferencia de la terapia de rehabilitación robótica de asistencia continuada, se fomenta la participación activa del paciente tanto en el aprendizaje del entrenamiento físico como en la restauración de las funciones locomotoras ambulatorias [10].

2.1.1 Características hardware del prototipo H1

El diseño de la estructura mecánica del exoesqueleto (figura 2.1.a) permite que se adapte cómodamente a la anatomía de cada paciente sin necesidad de tener que quitarse el dispositivo. Cada lado de la estructura bilateral y su reposapiés correspondiente, poseen varios eslabones ajustables de aluminio y acero inoxidable que permiten adaptar la longitud y posición del robot a cada paciente. Además, cuatro estructuras redondeadas con almohadillas de espuma y correas de velcro garantizan una mejor sujeción y acople del exoesqueleto a cada miembro inferior. Toda la arquitectura *hardware* de control junto con el sistema de alimentación del dispositivo, están ubicados en una mochila que portea el paciente a su espalda.

La tecnología de actuación se elige en base a las características de potencia y par que cada articulación presenta durante el ciclo habitual de la marcha a una velocidad normal [6]. Está formada por seis actuadores, seis reductores de velocidad y seis controladores electrónicos, uno por cada articulación y grado de libertad. También se establecen límites mecánicos en los actuadores (figura 2.1.c) para no rebasar el máximo rango articular de los movimientos normales de flexión y extensión (tabla 2.1). Todo ello, con objeto de evitar que el paciente realice movimientos perjudiciales que le puedan provocar posteriores lesiones.

Se escogen actuadores de corriente continua planos y sin escobillas ya que reúnen las características necesarias de potencia y portabilidad demandadas por los exoesqueletos. Como este tipo de motor por sí mismo no proporciona las características de par y velocidad que demandan las articulaciones, se acopla a su eje de salida un sistema reductor de velocidad *Harmonic Drive* [11]. De esta manera, se incrementa el par motor del actuador y se reduce su velocidad a la ideal desarrollada durante el ciclo normal de la marcha. El sistema de potencia que gobierna cada actuador es un *servo drive* de tipo PWM, el cual, además, usa la intensidad de corriente del actuador para calcular el par motor de salida del mismo [6,7].

Cada articulación posee una sensórica (figura 2.1.c) que consta: de un potenciómetro usado como sensor de posición angular y de una galga extensiométrica usada como sensor de fuerza para medir el par de interacción producido entre el sujeto y el exoesqueleto. Como excepción, la articulación del tobillo también contiene sensores de fuerza en la plantilla de apoyo del pie para detectar las diferentes fases de apoyo y balanceo de la marcha.

Los datos captados por los sensores son digitalizados mediante el módulo de adquisición *JointCan* de cada articulación. Posteriormente, ésta información es adquirida por dos módulos de adquisición (uno para cada lado del dispositivo) para después ser enviada al procesador (ver apartado 2.1.2) del exoesqueleto mediante protocolo *busCAN*. A su vez, estas dos placas también están conectadas a los controladores electrónicos de los motores, ya que son las encargadas de generar las señales analógicas de salida usadas por los controladores como referencia.

La forma que tiene para comunicarse el terapeuta con el exoesqueleto, es mediante un *SmartPhone* vía *Bluetooth* a través de un módulo de adquisición del robot que contiene una IMU y un adaptador *Bluetooth/CAN*. El dispositivo móvil ejecuta una aplicación que permite al profesional médico: parar y arrancar el dispositivo, variar la velocidad de marcha y cambiar los diferentes modos de control del prototipo [6]. En este TFM, no se ha tenido en cuenta esta parte, debido a que en ningún momento se ha tenido ni acceso, ni disponibilidad de la misma.

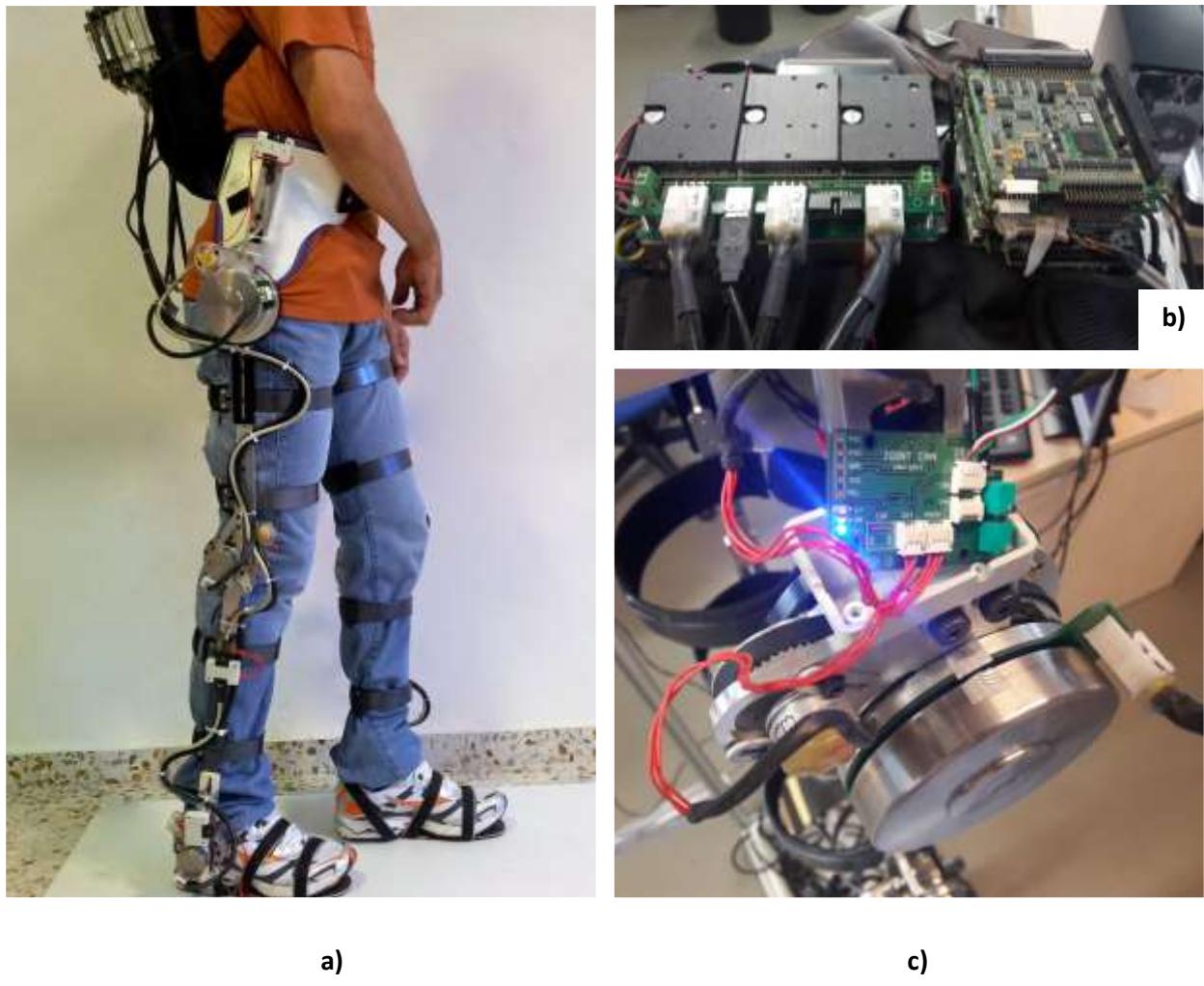


Figura 2. 1 Exoesqueleto **a)** Estructura del prototipo **b)** Mochila que portea la arquitectura de control: controladores, placas de adquisición y procesador, junto con el sistema de alimentación **c)** Actuador, sensores, placa JointCAN y límites mecánicos de la articulación de la cadera izquierda.

Articulación	Máximo rango articular		Actuadores			
	Flexión	Extensión	Potencia	Par Nominal	Reducción	Par máximo
Cadera	100 °	-20 °	90 W	390 mNm	100:1	39 Nm
Rodilla	-5 °	100 °	70 W	130 mNm	160:1	20.8 Nm
Tobillo	-15 °	20 °	70 W	130 mNm	160:1	20.8 Nm

Tabla 2. 1 Datos más relevantes del hardware del prototipo H1 para este TFM.

2.1.2 Características software del prototipo H1

La estrategia de control de par adaptativo realizada en este TFM (ver Capítulo 4) para el exoesqueleto H1 y las implementadas en el mismo de manera inicial, se han desarrollado mediante el programa *Matlab* y sus diferentes *toolboxes*. En ambos casos, el código necesario para el funcionamiento del prototipo se realiza mediante el programa *Simulink* y el uso de la *toolbox xPC Target* (versión 5.5). En el Anexo A, se explica de manera extensa el manejo de esta aplicación junto con la puesta en marcha del prototipo.

Como procesador del exoesqueleto se usa un sistema embebido, en concreto, un ordenador industrial PC104. Éste, ejecuta en tiempo real mediante la *toolbox xPC Target* de *Matlab* los algoritmos de control diseñados previamente en *Simulink*. Para poder realizar un análisis *offline* u *online* de los datos generados en el prototipo H1, el PC104 se conecta mediante protocolo *Ethernet* a un ordenador convencional.

Todo el proceso del *software* de control del dispositivo [6,7] se divide en dos partes principales: un control de alto nivel (*High Level Control*, HLC) y un control de bajo nivel (*Hardware Abstraction Layer*, HAL). Mediante el HLC, el dispositivo se encarga de implementar diferentes estrategias de control. Por otro lado, el HAL es responsable de gobernar los actuadores del dispositivo en base a la estrategia de control seleccionada.

Aunque el actuador de cada articulación puede implementar tres tipos de controladores según el modo seleccionado: posición, admitancia o par motor, son dos las estrategias de control desarrolladas por este exoesqueleto y validadas en sujetos sanos llevando a cabo diferentes experimentos [6]. En concreto, el modo de control de posición implementa la clásica estrategia de control de trayectoria que aporta asistencia de manera continuada al paciente sin considerar el esfuerzo voluntario del mismo. Este tipo de control consiste en que, mediante una trayectoria predeterminada basada en la cinemática de la marcha de sujetos sanos (ver Capítulo 3), un controlador Proporcional Integral Derivativo (PID), en el caso que nos aplica, guía las piernas del paciente por dicha trayectoria prefijada. Por otro lado, el modo de control de admitancia implementa la estrategia de control de trayectoria adaptativa, 'la idea es guiar la extremidad del paciente hacia una trayectoria de referencia permitiendo una desviación variable basada en la amplitud del par de interacción entre sujeto y exoesqueleto; cuando se incrementa el par, el robot proporcionalmente aplica una corrección a la trayectoria del paciente en un intento de adaptar el patrón de referencia de la marcha normal, a la propia marcha del sujeto' [7]. En esta memoria descriptiva se ha diseñado una estrategia de control de par adaptativa, basada en el modo de control de par motor. Esta estrategia se desarrolla ampliamente en el Capítulo 4 con idea de poder rehabilitar las discapacidades motoras de los pacientes.

2.2 Condiciones actuales del prototipo H1

De manera inicial los seis actuadores que posee el prototipo funcionaban correctamente, pero actualmente sólo funcionan los motores de la articulación de la cadera y de ambos tobillos. A excepción del actuador de la cadera izquierda, los actuadores funcionales tienen un comportamiento no lineal y presentan bastante rozamiento debido a problemáticas acontecidas en sus sistemas reductores. Todo ello sumado a la antigüedad del equipo y al poco mantenimiento del mismo, ha dado lugar a un desgaste mecánico grave y a la consecuente aparición de problemas mecánicos durante todo el desarrollo de este trabajo, dejando sólo un motor funcional.

Respecto a los sensores, destacar que la posición determinada por los potenciómetros y la medición de corriente por parte de los controladores electrónicos es correcta. Las galgas extensiométricas no interfieren en el desarrollo de este proyecto al no ser utilizadas, pero citar que no miden correctamente el par de iteración entre el exoesqueleto y el usuario.

Por otro lado, el funcionamiento del procesador del prototipo no es completamente óptimo. Durante una sesión de trabajo el sistema embebido no cargó el ejecutable del modelo diseñado en *Simulink*. Se decidió conectar un monitor y un teclado al sistema embebido (PC104), y se pudo averiguar que había un fallo en su disco de arranque. En el Anexo B, se muestran los diferentes pasos de cómo se solventó este problema permitiendo reiniciar el sistema de manera satisfactoria. Desde entonces, este incidente ocurre con bastante frecuencia, mayoritariamente cada vez que se enciende el equipo.

2.3. Conclusiones

El exoesqueleto H1 no se encuentra en condiciones óptimas de funcionamiento. Los problemas mecánicos ocurridos en los motores, a parte de encontrarse fuera del alcance de este TFM, no se han podido resolver completamente en el período que se ha desarrollado este trabajo. Por todo esto, las pruebas y el desarrollo de la estrategia de control descrita en esta memoria, se han realizado en el motor de la articulación de la cadera izquierda.

CAPÍTULO 3

Biomecánica de la marcha humana normal. Controladores de posición y par motor para la articulación de la cadera

Para estudiar e interpretar la marcha humana, se llevan a cabo dos tipos de análisis el cinético y el cinemático, ambos comparten el uso de sistemas de análisis de movimiento en tres dimensiones. Los datos facilitados por estos análisis son utilizados por distintos profesionales para justificar o evaluar diferentes tratamientos médicos y estudios. En este capítulo se trata únicamente el análisis cinemático, posteriormente, el análisis cinético se verá en el Capítulo 4 de este mismo documento.

Como ya se ha explicado en el Capítulo 2, el desarrollo de este TFM se enfoca en la articulación de la cadera izquierda. Por este motivo, en el presente capítulo se hace énfasis en el análisis cinemático de la marcha normal de dicha articulación visto desde el plano sagital, el cual, comprende los movimientos de flexión y extensión. Seguidamente se explica la calibración del correspondiente actuador y la sintonización de sus controles de posición y par motor.

3.1 Introducción a la biomecánica de la marcha humana normal

El ciclo de la marcha humana normal o zancada se define como 'el proceso de locomoción en el cual el cuerpo humano, en posición erguida, se mueve hacia delante, siendo su peso soportado, alternativamente, por ambas piernas' [12]. Como origen del ciclo se suele tomar el momento en el que uno de los dos pies entra en contacto con el suelo. Es decir, el ciclo empezaría cuando, por ejemplo, el pie derecho tuviera contacto con el suelo y finalizaría en el siguiente apoyo del mismo. A su vez, el pie izquierdo experimentaría exactamente los mismos desplazamientos que el pie derecho, pero trasladados en el tiempo medio ciclo. A diferencia de la carrera o del salto donde el cuerpo experimenta suspensión durante un instante (fase aérea), en la marcha nunca deja de haber apoyo en el suelo. Durante un ciclo completo de la marcha se da una sucesión de apoyo bipodal (ambos pies están en contacto con el suelo) y de apoyo monopodal (un solo pie está en contacto con el suelo). Las principales fases que describen la locomoción humana son la fase de apoyo y la fase de balanceo u oscilación. La fase de apoyo se produce

cuando el pie está en contacto con el suelo, comienza con el contacto inicial del retropié o talón de referencia y finaliza con el despegue del antepié. Por el contrario, la fase de balanceo es cuando el pie no está en contacto con el suelo, se inicia en el levantamiento del antepié hasta el siguiente apoyo. La figura 3.1. presenta un ciclo de la marcha completo, durante el cual, ambas fases son realizadas por cada pierna de manera alternada. Normalmente, la fase de apoyo abarca el 60% de todo el ciclo y la fase de balanceo constituye el 40% restante [12, 13].

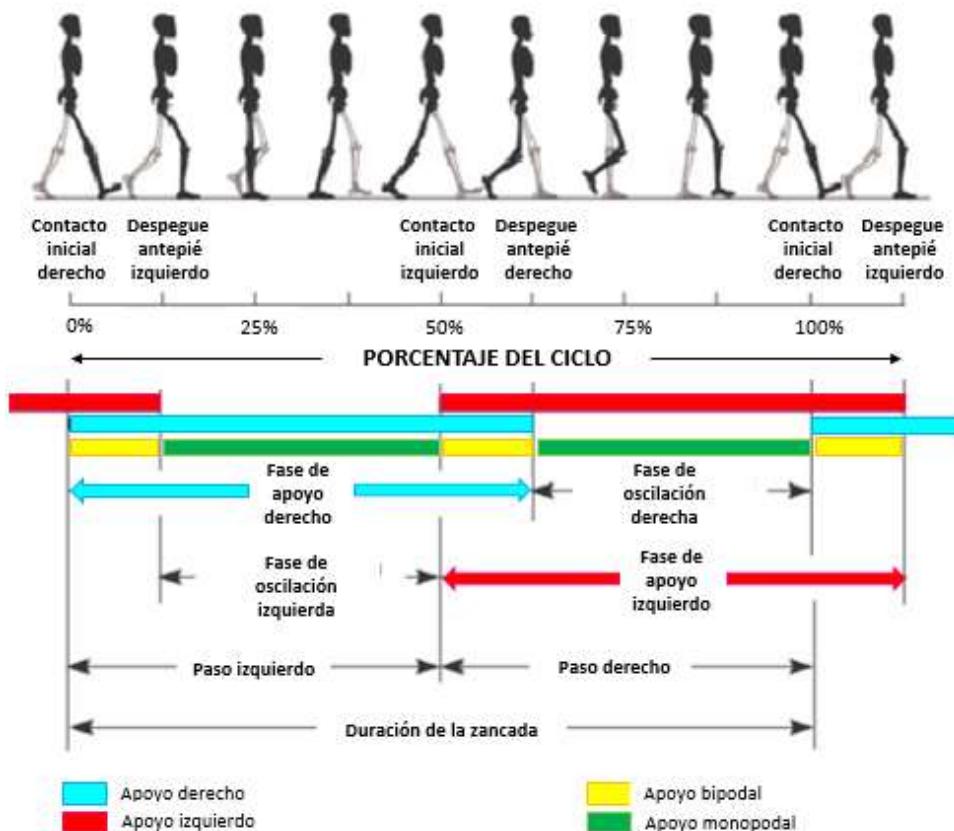


Figura 3. 1. Estructura de un ciclo completo de la marcha humana normal.

3.1.1 Análisis cinemático de la marcha normal en el plano sagital para la articulación de la cadera

En la práctica clínica, se considera como posición neutral o posición de equilibrio de la cadera, la correspondiente al fémur vertical, es decir, 0° en el plano sagital [12]. El rango total de movimiento en la articulación de la cadera durante la marcha normal es de unos 40° , resultando un rango articular de entre 10° de extensión y 30° de flexión (figura 3.2.b). La definición de los límites de la flexo-extensión del exoesqueleto H1 comprenden el rango articular límite de la cadera, el cual, abarca desde 20° de extensión hasta 100° de flexión (tabla 2.1).

Tanto la fase de apoyo como la fase de balanceo influyen en los movimientos de las distintas partes del cuerpo. En concreto, durante una zancada normal en el plano sagital, la articulación de la cadera realiza

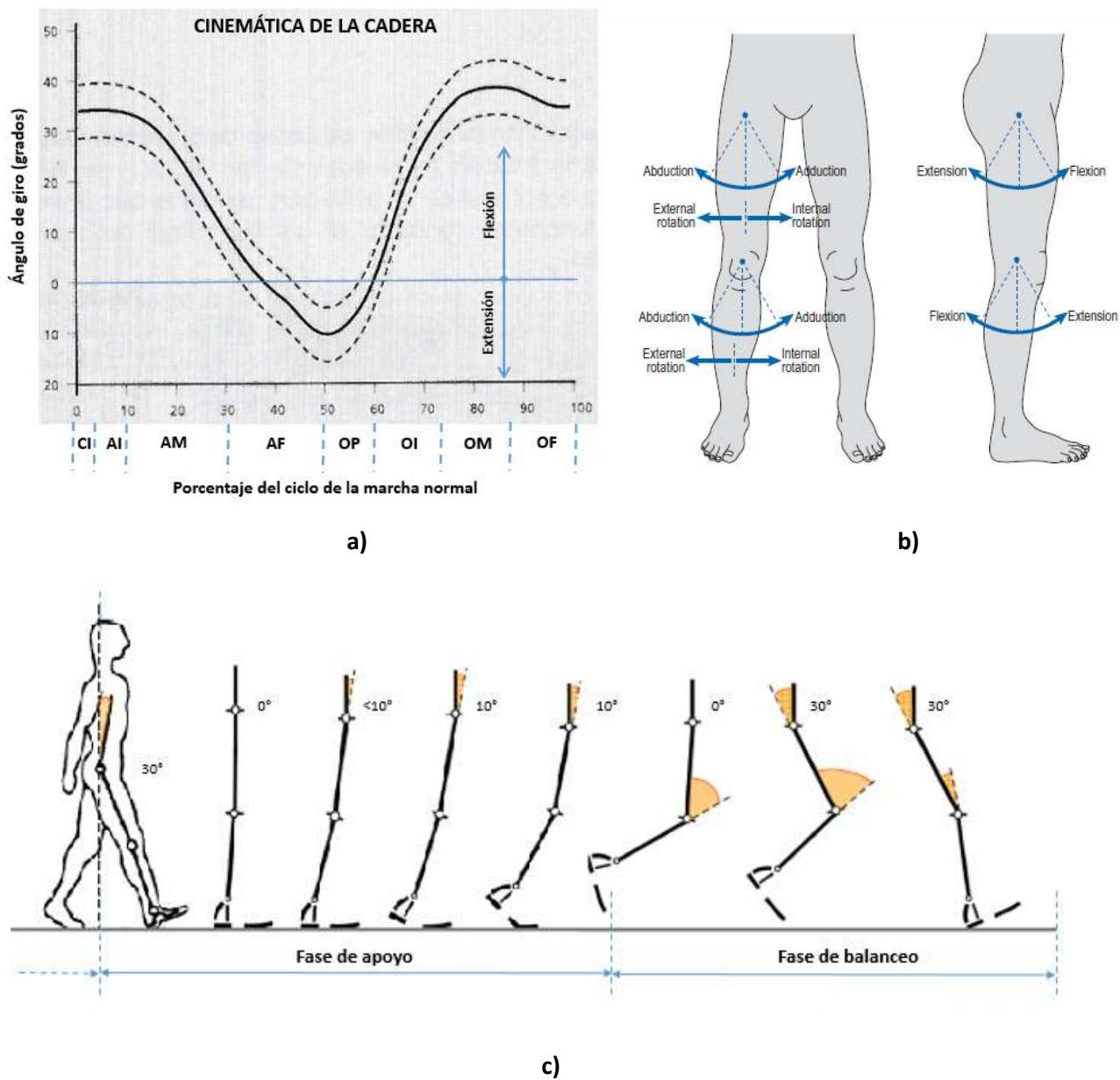


Figura 3.2. Cinemática de la articulación de la cadera **a)** Ángulos articulares en el plano sagital durante un ciclo de la marcha completo. Subfases de la fase de apoyo: CI= contacto inicial; AI= inicio apoyo; AM=apoyo medio; AF=final apoyo; OP= oscilación previa. Subfases de la fase de oscilación: OI=inicio oscilación; OM=oscilación media; OF=final oscilación. Los trazos discontinuos representan las bandas de dispersión (una desviación típica) **b)** Movimientos referentes a los tres grados de libertad de la articulación de la cadera (parte superior) y de la rodilla (parte inferior). Para ambas articulaciones, se observan los movimientos de flexión y extensión en el plano sagital, de abducción y aducción en el plano frontal y de rotación interna y externa en el plano transversal. **c)** Análisis funcional de la cadera en cada una de las fases que componen el ciclo de la marcha en el plano sagital.

un movimiento de extensión en la fase de apoyo y un movimiento de flexión en la fase de oscilación, produciéndose la inversión de forma gradual. La fase de apoyo se puede subdividir en cinco intervalos y la fase de balanceo en otras tres subfases o intervalos (figura 3.2.a y 3.2.c). En el instante de contacto inicial (CI) del talón con el suelo, la cadera presenta una flexión de 30° con respecto a la vertical, la cual, se mantiene durante la subfase inicial de apoyo (AI). Despues, se produce una extensión progresiva de la cadera en la subfase media de apoyo (AM) hasta alcanzar los 0° en la subfase final de apoyo (AF). En el 50% del ciclo de la marcha, cuando el otro pie contacta con el suelo, la cadera presenta un movimiento de hiperextensión de 10°. Posteriormente, la dirección del movimiento se invierte con cierta rapidez, produciéndose un movimiento de flexión hasta llegar a su posición neutra (subfase previa a la oscilación, OP). Finalmente, en la subfase inicial y media de la oscilación (OI y OM respectivamente) la cadera parte desde su posición neutra hasta una flexión de aproximadamente 30° en la cual se mantiene. La subfase final de oscilación (OF) es la transición entre la oscilación y el apoyo, donde la flexión de la cadera cesa para empezar otra nueva zancada [12].

3.1.2 Obtención de la referencia del exoesqueleto en base al análisis cinemático de la marcha normal en el plano sagital

El estudio cinemático del prototipo H1, se realiza mediante un sistema de captura de movimiento basado en cámaras de infrarrojos de alta velocidad y marcadores especiales adheridos a las extremidades inferiores de un paciente que presenta un ciclo normal de la marcha [6]. Las filmaciones simultáneas de las cámaras permiten reconstruir la localización tridimensional de los marcadores que reflejan la luz infrarroja. Esta información es procesada para conseguir la trayectoria tridimensional desarrollada por el sujeto y obtener los ángulos de las articulaciones a lo largo de la marcha. Dichos ángulos son usados como trayectoria de referencia en la programación del control de posición del exoesqueleto, los cuales, permiten generar un ciclo de la marcha similar al de la pierna real. Como la marcha humana es un proceso cíclico, sólo es necesario programar un patrón de referencia de los ángulos correspondientes a un solo paso.

3.1.3 Puesta en marcha y calibración del sensor de posición en base al análisis cinemático

Antes de poner en marcha el prototipo H1 por primera vez (ver Anexo A) y evitar un comportamiento no deseado del motor, es necesario realizar la calibración del potenciómetro del actuador de la cadera izquierda (figura 2.1.c). La calibración se hace en base a los ángulos límite de flexión y extensión que sopresa la articulación, los cuales, aparecen indicados en la tabla 2.1. De esta manera, se obtienen las medidas de posición dentro de los límites mecánicos del motor y a su vez, se respeta la cinemática de la cadera para evitar posibles fatalidades articulares al paciente

Para calibrar el potenciómetro del actuador de la cadera izquierda, inicialmente, se carga en el sistema embebido el *software* del prototipo del modelo de *Simulink* con modo control de posición y sin alimentar los actuadores. Luego, se coloca el exoesqueleto en la posición de equilibrio y se calibra manualmente el potenciómetro hasta obtener el ángulo de giro deseado. Seguidamente, y del mismo modo, se realiza la calibración de las posiciones extremas de flexión y extensión indicadas por los límites mecánicos (figura 3.3). Citar que, para la pierna derecha e izquierda del exoesqueleto, el movimiento del ángulo de giro positivo y negativo del potenciómetro son contrarios. Es decir, para la pierna izquierda el movimiento

positivo del potenciómetro se realiza en la dirección de las agujas del reloj, mientras que el movimiento negativo del mismo se efectúa en sentido contrario.

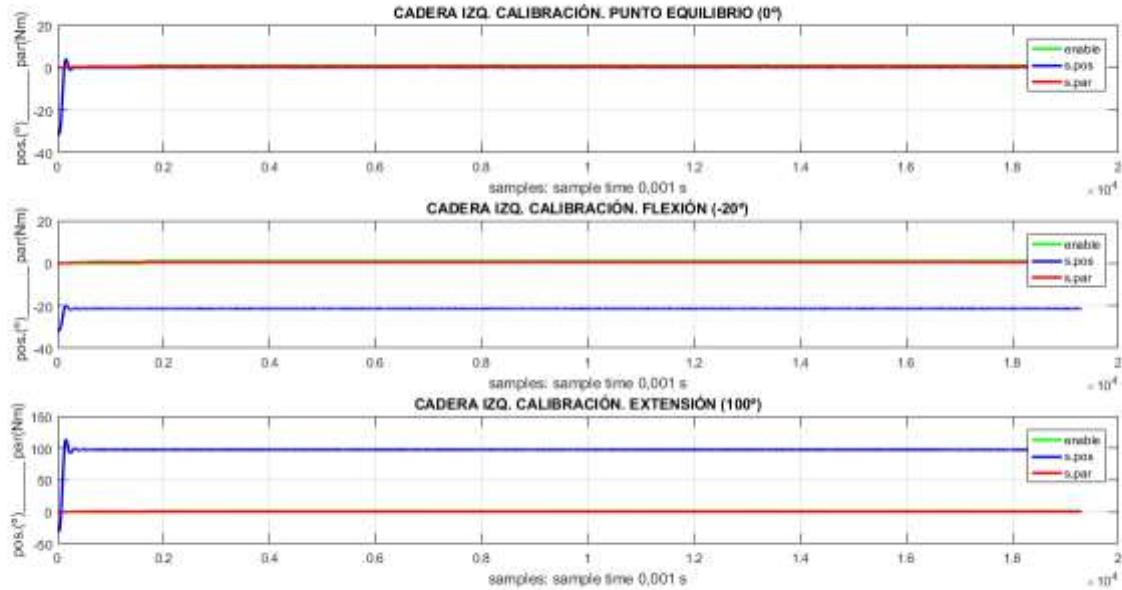


Figura 3. 3. Calibración de la posición del prototipo H1 (en azul). En la primera gráfica se muestra la posición neutra de la cadera (0°). En la segunda y tercera gráfica se observan los ángulos límite de los movimientos de flexión (100°) y extensión (-20°) respectivamente, según datos de la tabla 2.1.

3.2 Controladores de posición y par motor para la articulación de la cadera

El exoesqueleto H1, según el modo de control elegido, permite implementar un control de posición, admitancia o par en cada una de las articulaciones (ver Capítulo 2). El propósito final de este TFM, implica trabajar tanto con el control de posición como con el control de par motor del actuador de la cadera izquierda, siendo ambos implementados mediante un controlador digital PID.

Los controladores PID ofrecen unas prestaciones muy aptas para la mayoría de los sistemas de control, por ello son los más utilizados en toda clase de industrias. Si se conoce el modelo matemático del proceso o planta es posible enfocar el uso del algoritmo PID analíticamente, en caso contrario, la sintonía experimental de estos controladores mediante unas pocas reglas empíricas, resulta de mucha utilidad.

Este sistema dinámico de tiempo continuo (ecuación 3.1) tiene como propósito que el error ($e(t)$) en estado estacionario entre el valor de referencia o consigna ($y_{ref}(t)$) y la señal de salida del proceso ($y(t)$), sea cero de manera asintótica en el tiempo. La señal a controlar ($u(t)$) es la suma de tres términos, una parte proporcional que actúa en el valor presente del error minimizándolo, una parte integral que reduce el error a cero en régimen permanente representando el promedio de los errores pasados y una parte derivativa que puede ser interpretada como una acción anticipativa, la cual, reduce la sobreoscilación e incrementa la estabilidad de la planta [14]. Para sintonizar este tipo de controladores (figura 3.4), analítica o empíricamente, según el conocimiento o no de la dinámica del proceso, se deben definir los parámetros de las ganancias proporcional (K_P), integral (K_I) y derivativa (K_D) y así lograr un

comportamiento del sistema aceptable y robusto. Si este tipo de controlador es implementado usando un computador, microprocesadores, conversores ADC y DAC.. etc, el sistema de tiempo continuo debe ser aproximado por un sistema de tiempo discreto con una frecuencia elevada de muestreo (muestreo rápido; T pequeño), obteniendo de esta manera un controlador digital PID. Para ello, el término integral analógico se puede aproximar a través de una sumatoria y el término derivativo analógico se puede aproximar mediante una función de primer orden, dónde n indica el tiempo de muestreo actual y T es el período de muestreo del sistema (ecuación 3.2).

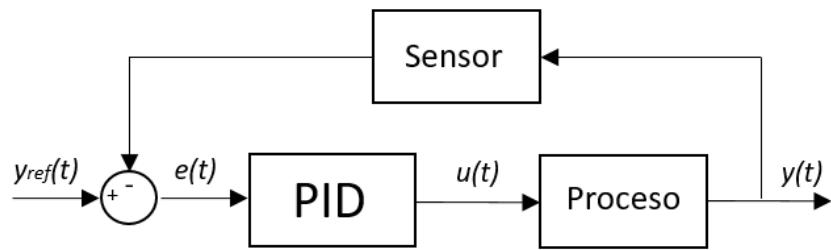


Figura 3. 4. Esquema general de un controlador PID en tiempo continuo.

$$u(t) = K_P e(t) + K_I \int_0^t e(\tau) d\tau + K_D \frac{d}{dt} e(t) \quad (3.1)$$

$$u(n) = K_P e(n) + K_I \sum_{i=1}^n e(i)T + K_D \frac{e(n)-e(n-1)}{T} \quad (3.2)$$

En la experimentación preliminar con los controladores de posición y par motor del exoesqueleto H1, según los parámetros de los PID digitales planteados en [6], y junto a la situación actual del equipo (ver Capítulo 2), no se obtiene el funcionamiento deseado (rapidez de respuesta, sin o con pequeña oscilación) de ambos controladores. Por esta razón, en los siguientes apartados, se procede a realizar el reajuste empírico de los parámetros proporcional, integral y derivativo del sistema dinámico de tiempo discreto.

3.2.1 Control de posición

El controlador de posición, se reajusta empíricamente mediante uno de los dos métodos desarrollados por Ziegler y Nichols (ZN) en los 1940s [15] y que todavía se usan de manera muy frecuente sin necesidad de conocer las ecuaciones de la planta. En concreto, la sintonización del controlador se realiza mediante el método de respuesta en frecuencia o método de oscilación. El cual, a través de datos experimentales extrae una aproximación de las características del proceso dinámico, para posteriormente determinar a partir de dichas propiedades, los parámetros K_P , K_I y K_D del controlador.

Aplicando ZN, el proceso de los datos es obtenido conectando inicialmente un lazo realimentado con un control proporcional ($K_I = 0$ y $K_D = 0$). La ganancia del controlador proporcional (K_P) es aumentada hasta un valor crítico (K_C) dónde el sistema alcanza una oscilación sostenida haciéndolo críticamente

estable. Se toma el valor de la ganancia crítica y del período crítico (T_C), para calcular los parámetros aproximados de control mediante las fórmulas de sintonía de la tabla 3.1.

	K_P	K_I	K_D
P	$0.50 * K_C$	-	-
PI	$0.45 * K_C$	$0.54 * K_C / T_C$	-
PID	$0.59 * K_C$	$1.18 * K_C / T_C$	$0.074 * K_C / T_C$

Tabla 3. 1. Fórmulas de sintonía del método de respuesta en frecuencia de ZN.

Para obtener un ajuste fino de los parámetros y conseguir un comportamiento dinámico deseado, el procedimiento de ZN se complementa con el procedimiento experimental de ajuste de prueba y error. Es decir, partiendo de los parámetros obtenidos empíricamente mediante ZN, se realizan manualmente varias modificaciones sucesivas de los parámetros de control hasta conseguir los adecuados (figura 3.5). De esta manera, se obtiene la respuesta que mejor se ajusta a la referencia indicada del control de posición (figura 3.6).

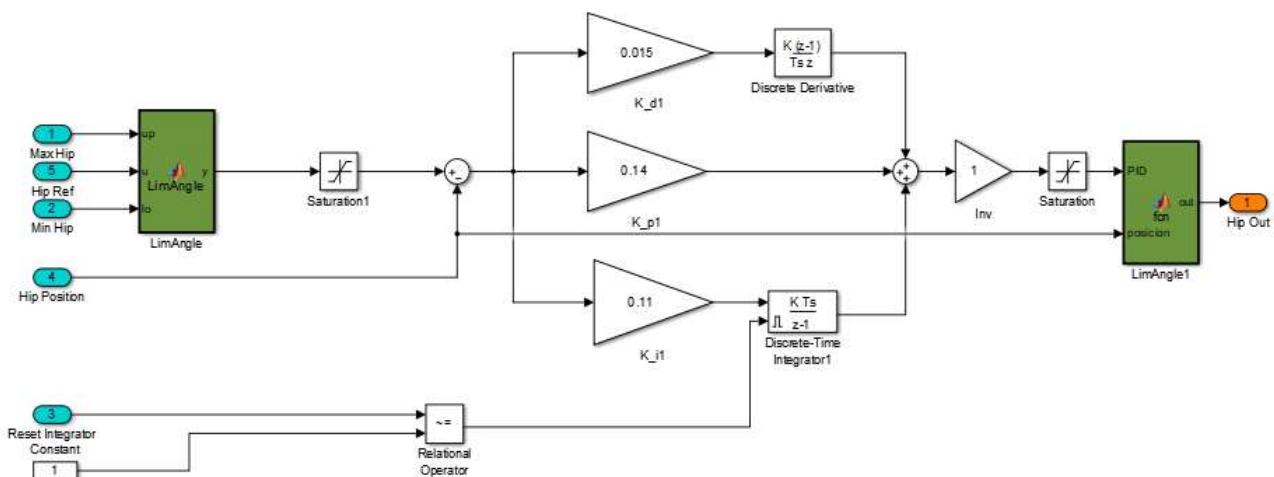


Figura 3. 5. Esquema del controlador de posición Simulink para la articulación de la cadera izquierda. Parámetros sintonizados: $K_P = 0.14$, $K_I = 0.11$ y $K_D = 0.015$, aplicando el procedimiento de ajuste de ZN, seguido de una corrección manual para evitar sobreoscilación y error estacionario.

La figura 3.6, presenta la respuesta (azul grueso) del prototipo H1 (tras la sintonización del controlador de posición) ante la trayectoria de referencia (azul fino) de la cinemática que desempeña la cadera izquierda durante la marcha humana normal. Como origen del ciclo de la marcha se toma el pie derecho, por lo que la referencia de la cadera izquierda va retrasada medio ciclo en el tiempo. Por esto, el comienzo del paso izquierdo empieza en el 50%-60% del ciclo de la marcha, es decir, la cadera está en su posición neutra, los 0° (figura 3.2.a). En este comienzo, el actuador desarrolla un par (rojo) positivo hasta que el exoesqueleto simula un movimiento de flexión de aproximadamente 30° (fase de balanceo). En la transición entre la fase de oscilación y de apoyo, cuando empieza una nueva zancada, el motor presenta un par negativo (instante en el que el talón contactaría con el suelo) y el dispositivo una flexión de 30° . Desde este momento hasta alcanzar la posición correspondiente a una extensión de 5° , el exoesqueleto

simula una extensión progresiva acompañada por una disminución gradual del par negativo del actuador junto a la asistencia de la gravedad.

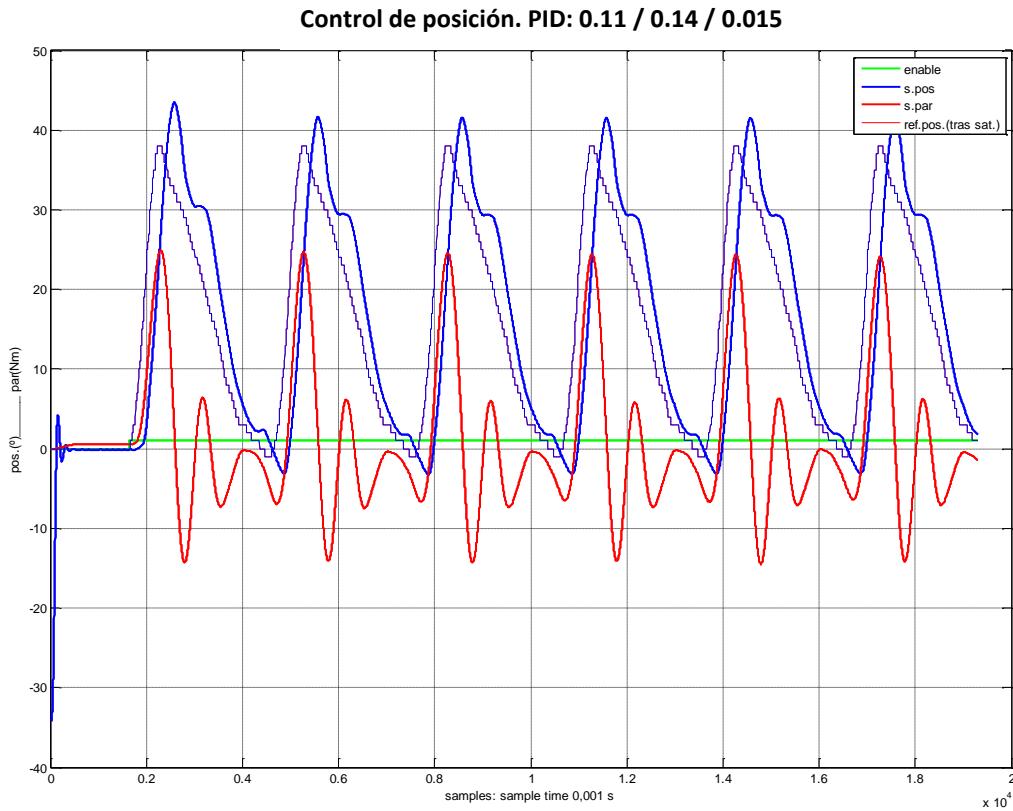


Figura 3. 6. Respuesta del exoesqueleto H1 durante los seis primeros pasos para el controlador de posición sintonizado.

3.2.2 Control de par motor

Esta estrategia de control, a diferencia de la planteada en [6,7] no usa la trayectoria de referencia de la cinemática de la cadera, sino que toma como referencia el par generado por el motor durante el control de posición. La idea, es conseguir que el exoesqueleto realice el mismo movimiento de la marcha controlando esta vez el par motor del actuador. Posteriormente se justificará por qué este tipo de control de par se adapta mejor que el control de posición para una asistencia mediante exoesqueleto en un procedimiento de rehabilitación de discapacidades motoras (ver Capítulo 4).

Se desarrolla un *script* en *Matlab* (ver Anexo C) para obtener la consigna de par partiendo del control de posición (independientemente del número de períodos o pasos de la señal). Se modifica la frecuencia de muestreo de cada uno de los pasos (3 seg.) de las señales originales de posición y par motor obtenidas en el control de posición (figura 7.a), para conseguir los 200 valores que necesita la matriz de referencia almacenada en la memoria del PC104. El *software* acumula uno de estos valores para la articulación cada 15 mseg. (200 x 15 mseg. = 3 seg.). Tras diezmar las señales, se aplica un bucle que calcula el error cuadrático medio (ECM) entre la consigna de posición (Y_i) y la posición medida por el potenciómetro (\hat{Y}_i) para las 200 iteraciones de cada paso diezmado (ecuación 3.3). Posteriormente, se trasladan circularmente las señales diezmadas (de posición y de par motor) tantas iteraciones como indica el

mínimo EMC para cada período, con el objetivo de evitar cualquier tipo de desfase. De esta manera, se obtiene una gran similitud entre la trayectoria de posición de referencia y la señal de posición trasladada, además de conseguir ausencia de desfase entre las señales trasladadas y las señales originales de posición y par motor (figura 7.b).

$$ECM = \frac{1}{n} \sum_{i=1}^n (\hat{Y}_i - Y_i)^2 \quad (3.3)$$

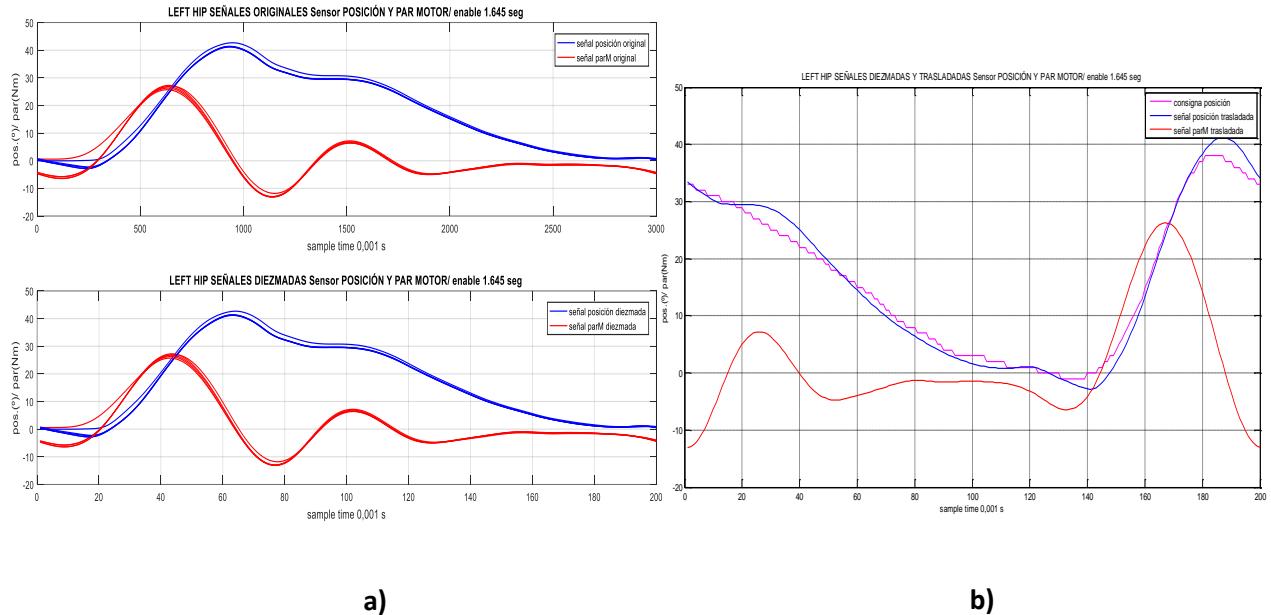


Figura 3. 7. Obtención de la consigna de control de par **a)** La gráfica superior muestra el período de cada uno de los pasos de las señales originales de posición (azul) y par motor (rojo) para 6 pasos. La gráfica inferior muestra el período de las mismas señales, pero diezmadas a 200 valores. **b)** Referencia de la trayectoria de la cadera izquierda (magenta), señal de posición (azul) y par motor (rojo) trasladadas el valor del mínimo ECM. La señal de par obtenida (rojo), es la referencia que se usará en el control de par motor de la cadera izquierda.

La sintonización del controlador del par motor se realiza siguiendo los mismos procedimientos indicados en el control de posición. Para el cálculo de las ganancias K_P , K_I y K_D (figura 3.8), previamente se aplica el método de ZN y posteriormente se aplica una corrección manual de los parámetros. Los bloques *switch* de esta figura, tienen como función limitar el par motor. En este proyecto no se tienen en cuenta ya que se ha trabajado con un control de par motor puro.

La figura 3.9, presenta la respuesta (azul grueso) del prototipo H1 (tras la sintonización del controlador de par) ante la trayectoria de referencia (azul fino) del par que desempeña el exoesqueleto durante la cinemática de la cadera izquierda en el proceso de la marcha humana normal. De la misma manera que ocurría en el control de posición, para una simulación de la flexión de la marcha humana, el motor aplicar par positivo, y para una extensión ocurre lo contrario, el par se vuelve negativo. El exoesqueleto reproduce de manera satisfactoria el movimiento de la trayectoria de posición.

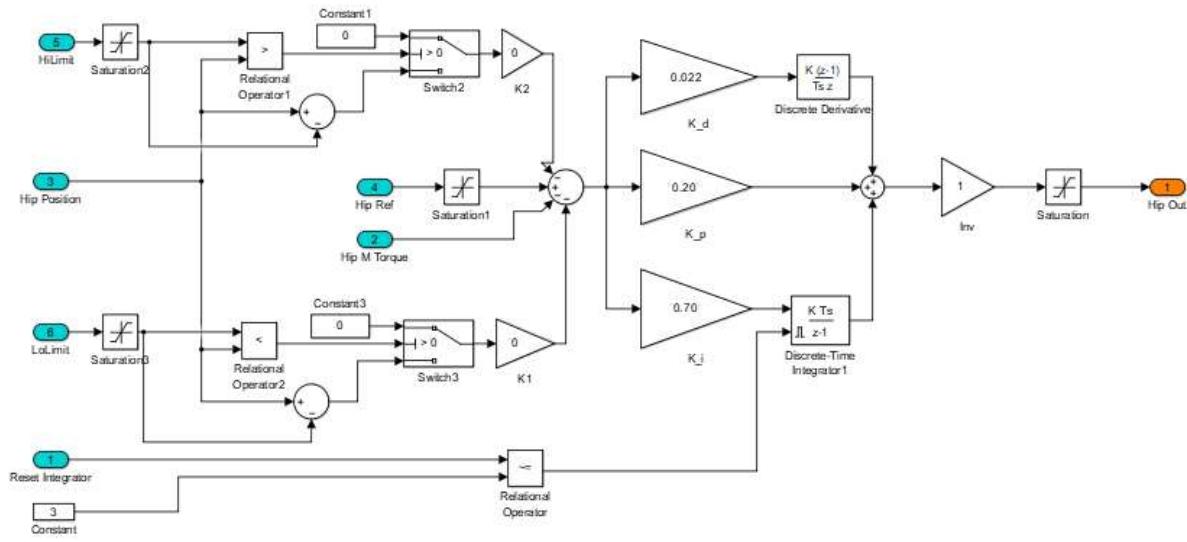


Figura 3. 8. Esquema del controlador de par Simulink para la articulación de la cadera izquierda. Parámetros sintonizados: $K_p = 0.20$, $K_i = 0.70$ y $K_d = 0.022$, aplicando el procedimiento de ajuste de ZN, seguido de una corrección manual para evitar sobreoscilación y error estacionario.

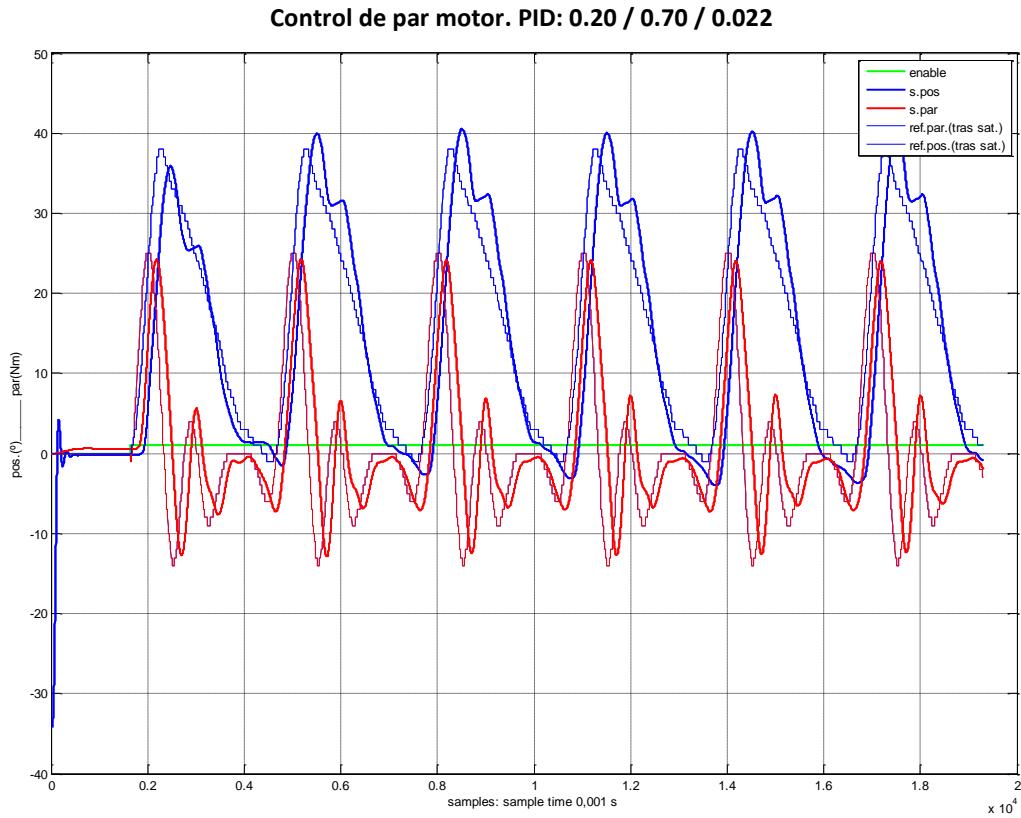


Figura 3. 9. Respuesta del exoesqueleto H1 durante los seis primeros pasos para el controlador de par motor sintonizado.

3.3 Conclusiones

Tras sintonizar los controles de posición y para motor para poder realizar de manera adecuada la función de la marcha normal humana, se procede a diseñar en el siguiente capítulo, el control de par adaptativo a pares externos para un paciente de 60 Kg.

CAPÍTULO 4

Control de par adaptativo para la articulación de la cadera

En este capítulo, se indican los pasos del desarrollo del control de par adaptativo a pares externos. Partiendo del esquema general del mismo y para poder construir una idea sobre su alcance, se explican los distintos bloques que lo componen. Después, se muestran varias simulaciones de asistencia del robot para distintas capacidades residuales de par articular del miembro inferior de un paciente de 60 Kg.

4.1 Concepto y esquema general del control de par adaptativo

Para diseñar este control en la articulación de la cadera izquierda, se parte de la idea de que un simple control de posición no permite adaptarse a pares externos, ventaja que sí presenta un control de par motor. Por ello, este tipo de control se adapta mejor para poder realizar terapias de rehabilitación de la marcha en pacientes. De tal forma que el par de asistencia que ejecute el actuador del exoesqueleto, se pueda regular en base al par externo que ofrezca la cadera del paciente durante la acción de caminar.

El objetivo de este control, es el de conseguir una recuperación gradual del movimiento de la cadera de un paciente durante la marcha normal humana, a través de la asistencia del exoesqueleto H1 y orientado bajo el paradigma AAN. Es decir, el diseño de este sistema de control permite aplicar al prototipo diferentes niveles de asistencia, en mayor o menor grado, según sea la capacidad latente contributiva del miembro inferior del paciente. Con esto, se pretende lograr una recuperación y un fortalecimiento de las discapacidades motoras del mismo. En este trabajo se ha adaptado a una persona de 60 Kg.

En la figura 4.1 se muestra el esquema general de los distintos bloques que forman parte del control de par adaptativo diseñado. Posteriormente, se explica el desarrollo, funcionamiento e implicación de cada una de las partes.

- Bloque de control de par adaptativo (ver apartado 4.2), es el encargado de generar el par de asistencia a desarrollar por el exoesqueleto según la terapia de rehabilitación demandada por la capacidad residual del sujeto.

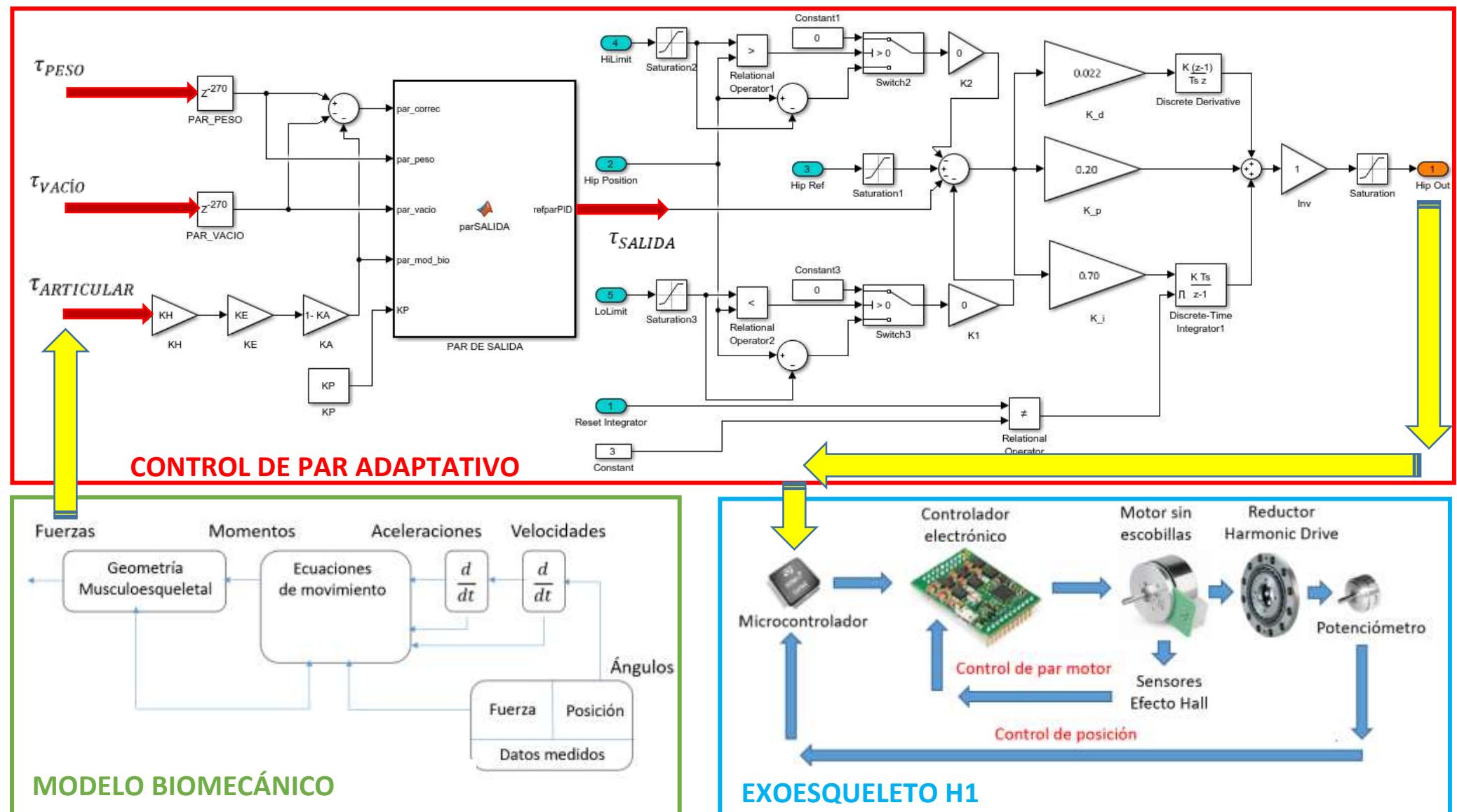


Figura 4. 1. Esquema general del control adaptativo de par motor

- Bloque del modelo biomecánico (ver apartado 4.2.2 y anexo E), genera el par articular desarrollado por la cadera del supuesto paciente. El objetivo es conseguir unos resultados en simulación lo más coherentes con la realidad.
- Bloque del exoesqueleto H1, el cual representa el sistema físico que va a ayudar al paciente en la rehabilitación de la marcha humana normal. Este bloque se detalla en profundidad en el Capítulo 2 de este documento.

El control de par permite controlar el movimiento del exoesqueleto en base a diferentes pares (figura 4.1):

- 1) par necesario para mover el exoesqueleto en vacío ($\tau_{VACÍO}$), es decir el controlador ejerce el par en las articulaciones necesario para que la persona no tenga que ejercer fuerza alguna adicional a la correspondiente a su propio peso y dinámica (robot ingravido).
- 2) par ejercido por la propia persona ($\tau_{ARTICULAR}$), a partir de su capacidad latente. Dado que se orienta a ejercicios de rehabilitación, la persona debe realizar esfuerzo de movimiento, no dejar que el robot realice todo el trabajo. Este par puede calcularse de diversas maneras: i) a partir de medidas de activación muscular (EMG), con las que calcular los pares articulares reales; ii) a partir de sensores de par que midan la fuerza de contacto ejercida por la persona sobre el exoesqueleto; iii) a partir de un modelo biomecánico que calcule los pares articulares ejercidos para un movimiento determinado. En este trabajo se ha optado por este último método, por dos motivos fundamentales:
 - los otros dos suponen un procesamiento adicional de las señales, fuera de los objetivos de este trabajo.
 - el uso de un modelo biomecánico permite evaluar el método de control, objetivo del trabajo, de una manera más sencilla para diferentes condiciones y capacidades latentes, desde una contribución nula del paciente a una contribución total.
- 3) par de control para complementar el par ejercido por el paciente en la ejecución del movimiento ($\tau_{SALIDA} = \tau_{ASISTENCIA}$; ver apartado 4.2). Si el paciente es capaz de realizar toda la fuerza necesaria, la única asistencia del controlador es para ejercer el par necesario explicado en el punto 1); si no es capaz de ejercer toda la fuerza, el controlador aplicará el par adicional necesario para la ejecución del ejercicio.

4.2 Diseño del control de par adaptativo

Con este controlador se establece un sistema que controla la trayectoria de posición de la marcha mediante un control de par motor, el cual, puede adaptarse a pares externos como se ha mencionado previamente. El diseño realizado permite realizar dos tipos de ajuste del sistema:

- la contribución del sujeto, permitiendo simular distintos grados de contribución (0% a 100%)
- la asistencia del exoesqueleto, permitiendo simular distintos grados de contribución de cara a forzar en un ejercicio de rehabilitación una mayor participación del paciente en su rehabilitación e incluso una asistencia negativa ((esto no se ha contemplado en este trabajo) del exoesqueleto con objeto de forzar más la participación del paciente, como se hace en ejercicios de rehabilitación habitualmente.

En el bloque de control de par adaptativo de la figura 4.1, destaca una parte inicial que permite seleccionar cual va a ser el par deseado a ejecutar por el motor del exoesqueleto. Posteriormente, este par seleccionado según la terapia de estudio, sirve como consigna de referencia al controlador de par motor del actuador (ver Capítulo 3), sintonizado para simular la cinemática de la flexo-extensión de la articulación de la cadera durante la acción de caminar en el prototipo H1 (ver Capítulo 2).

En el sistema de control de par adaptativo intervienen tres tipos de pares de entrada (figura 4.1):

- El par robot en vacío ($\tau_{VACÍO}$) o par ejercido por el motor del exoesqueleto para moverse a sí mismo sin miembro inferior del paciente (ver Capítulo 3).
- El par con peso (τ_{PESO}) o par ejercido por el motor del exoesqueleto para moverse a sí mismo y al peso de un miembro inferior de un paciente de 60 Kg (ver apartado 4.2.1).
- El par o momento articular ($\tau_{ARTICULAR}$) que realiza la articulación de la cadera de un paciente sano de 60 Kg., durante la trayectoria de la marcha sin tener en cuenta fuerzas externas. Es decir, debido a que el prototipo H1 no está en contacto con el suelo sino colgado de una estructura metálica, no se tiene en cuenta el momento de la fuerza de reacción en la articulación de la cadera, en el plano sagital, durante la fase de apoyo de la marcha (ver apartado 4.2.2). La situación simula por tanto un ejercicio de flexión-extensión de cadera sin contacto con el suelo.

Mediante las tres siguientes variables, el sistema de control de par permite al exoesqueleto realizar múltiples simulaciones de casos de asistencia para ayudar al paciente de 60 Kg. a completar el ciclo de la marcha, según se indique el nivel de contribución del mismo, el cual, dependerá de su capacidad residual (figura 4.1):

- K_H , simula la capacidad residual que puede tener el paciente de 60 Kg según el grado de lesión cerebrovascular sufrida. Este nivel de capacidad variará entre $0 < K_H \leq 1$; si $K_H = 1$ la capacidad del paciente será la de un paciente sano.
- K_E , simula la contribución del paciente dentro de su rango de capacidad residual. Este nivel de contribución variará entre $0 < K_E \leq 1$; si $K_E = 1$ el paciente contribuye con el 100% de su par residual para llevar a cabo el movimiento del ejercicio y el robot asistirá en consecuencia.
- K_A , permite variar la asistencia del exoesqueleto. El nivel de asistencia se establece entre $0 < K_A \leq 1$, entonces el exoesqueleto suministrará parte del par para hacer el movimiento y el paciente deberá ejercer el resto de la fuerza para completar el ejercicio; si $K_A = 1$ y $K_H = 1$ será el robot el que haga toda la asistencia del movimiento y la contribución del paciente será nula.

El par de salida del sistema (τ_{SALIDA}) es controlado por el parámetro selector K_P , es decir, según el valor definido entre $K_P \in \{0, 1, 2, 3, 4\}$ se pueden obtener diferentes pares de salida según sea el objetivo de estudio a desarrollar con el prototipo H1:

- Para $K_P = 0$, $\tau_{SALIDA} = \tau_{ASISTENCIA}$; ver ecuación 4.1.
- Para $K_P = 1$, $\tau_{SALIDA} = \tau_{PESO}$
- Para $K_P = 2$, $\tau_{SALIDA} = \tau_{VACÍO}$
- Para $K_P = 3$, $\tau_{SALIDA} = \tau_{ASISTENCIA} + \tau_{VACÍO}$
- Para $K_P = 4$, $\tau_{SALIDA} = \tau_{ARTICULAR}$

$$\tau_{ASISTENCIA} = \tau_{PESO} - \tau_{VACÍO} - (1 - K_A) * (K_H * (K_E * \tau_{ARTICULAR})) \quad (4.1)$$

Para que el conjunto del paciente y del prototipo se muevan correctamente, es necesario sumar al par establecido por el nivel de asistencia, el par que ejerce el exoesqueleto en vacío para que este no caiga por su propio peso y pueda dañar al paciente.

4.2.1 Obtención del τ_{PESO}

En la figura 4.2.b se muestra la distribución de diferentes pesos a lo largo del exoesqueleto para simular la masa corporal del miembro inferior izquierdo de un paciente de 60 Kg. La idea es poder identificar el comportamiento del par motor cuando el exoesqueleto es usado por un paciente (τ_{PESO}).

La distribución del peso se basa en los diferentes porcentajes de peso corporal correspondientes a las distintas partes del cuerpo humano de la figura 4.2.a (bibliografía). Para el caso que nos atañe en este TFM, el peso corporal total que representa el miembro inferior entero de un usuario de 60kg y el peso corporal de cada una de las partes que lo componen es indicado en la tabla 4.1.

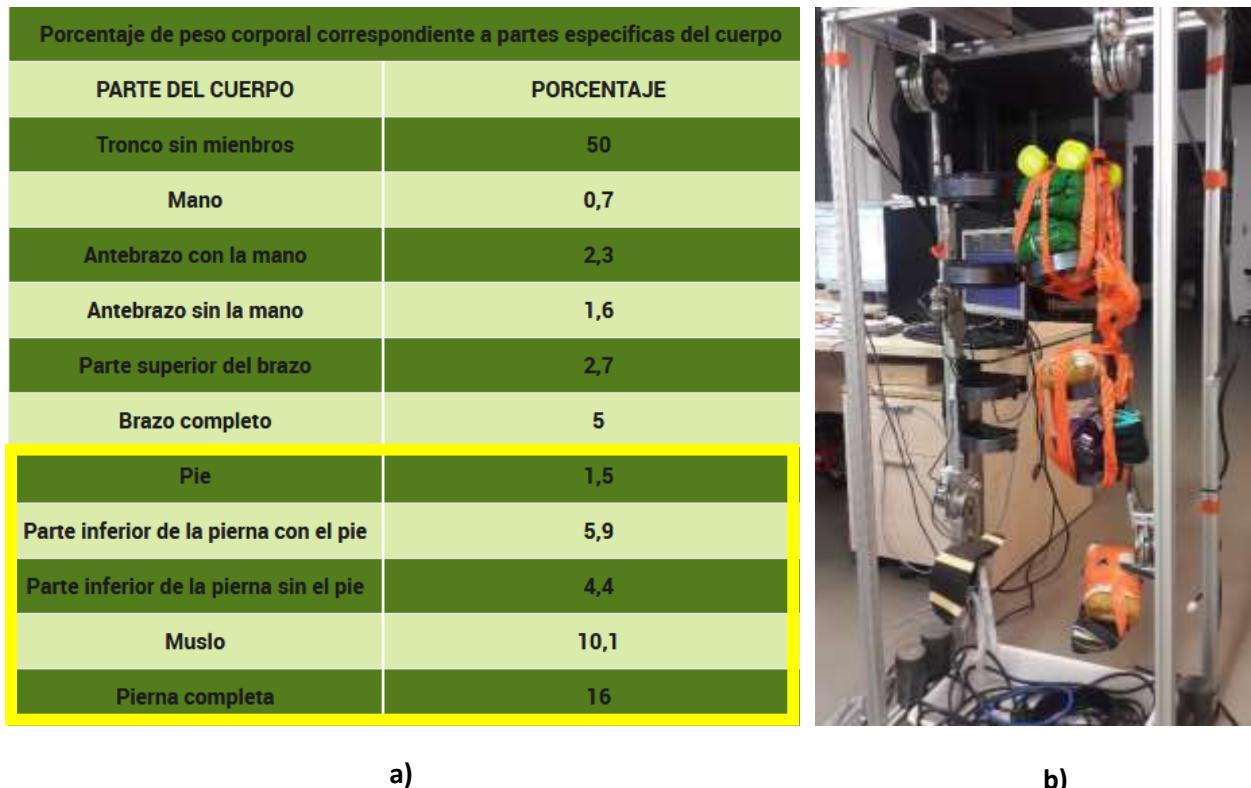


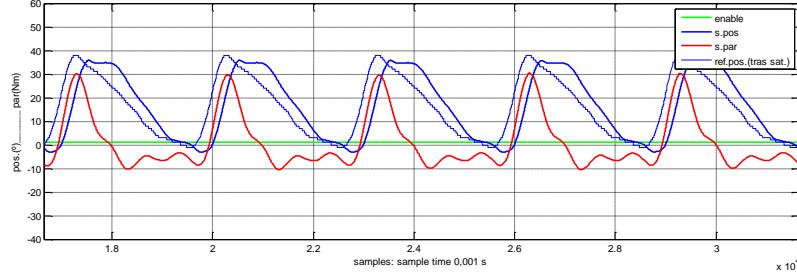
Figura 4. 2 a) Porcentajes de peso corporal correspondientes a diferentes partes del cuerpo humano
b) Simulación del peso corporal de un miembro inferior izquierdo mediante pesos en el prototipo H1

PESO SUJETO	PIERNA COMPLETA (16 %)	MUSLO (10,1 %)	PIERNA INFERIOR SIN PIE (5,9 %)	PIERNA INFERIOR CON PIE (4,4 %)
60 Kg	9,6 Kg	6,06 Kg	2,64 Kg	3,54 Kg

Tabla 4. 1 Distribución de los pesos de las diferentes partes del miembro inferior de un paciente de 60 kg

La respuesta del motor al control de posición sin peso (gráfica superior) y con peso (gráfica inferior) se aprecia en la figura 4.3. Se observa que la cinemática de la flexo-extensión de la cadera (en azul grueso) es alcanzada favorablemente por el controlador sintonizado en el Capítulo 3. Por otro lado, el par ejercido por el motor (en rojo) es mucho mayor que el realizado en el control de posición con el exoesqueleto en vacío debido, en este caso, a que el motor debe cargar con el peso del conjunto paciente - exoesqueleto.

Control de posición sin peso. PID: 0.11 / 0.14 / 0.015



Control de posición con peso. PID: 0.11 / 0.14 / 0.015

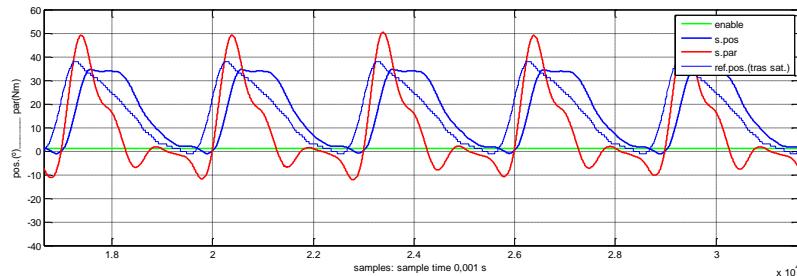
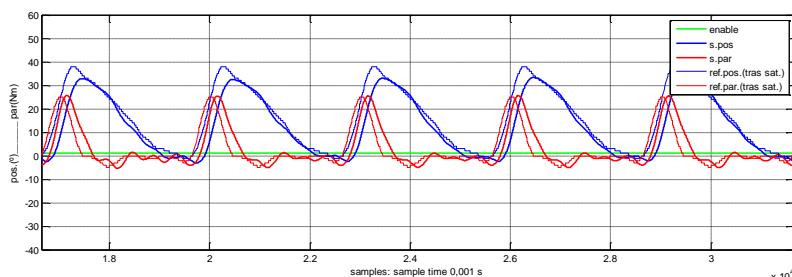


Figura 4. 3 Respuesta del control de posición (azul grueso) al peso del propio exoesqueleto (gráfica superior) y al conjunto de paciente – exoesqueleto (gráfica inferior).

Control de par motor sin peso. PID: 0.20 / 0.70 / 0.022



Control de par motor con peso. PID: 0.20 / 0.70 / 0.022

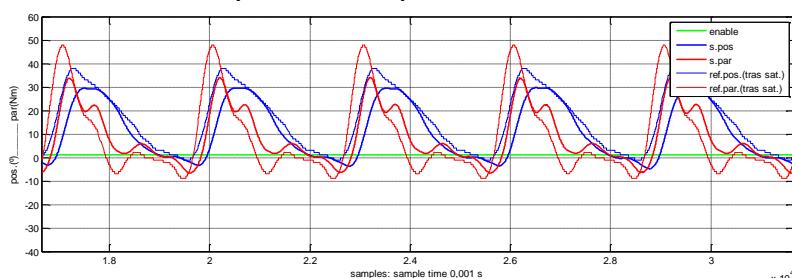


Figura 4. 4 Respuesta del control de par motor (en rojo grueso) al peso del propio exoesqueleto (gráfica superior) y al conjunto de paciente – exoesqueleto (gráfica inferior).

Por otro lado, la respuesta del actuador al control de par motor sin peso (gráfica superior) y con peso (gráfica inferior) se aprecia en la figura 4.4. De la misma manera que ocurre en el caso anterior, la flexo-extensión es realizada adecuadamente por el controlador de par motor sintonizado en el Capítulo 3. En el caso del conjunto paciente-exoesqueleto, debido al peso adicional, es más notoria la variabilidad del par, debido al deterioro mecánico sufrido por el conjunto motor-engranaje.

Hay que comentar que las gráficas de las figuras 4.3 y 4.4, son anteriores a las gráficas de las figuras 3.6 y 3.9 vistas hasta ahora. Es decir, las respuestas del control de posición y par motor con y sin peso, de las figuras 4.3 y 4.4 son obtenidas cuando varios rodamientos fuera de lugar provocaban un roce, el cual afectaba a la respuesta del par motor. El comportamiento actual del actuador de la cadera izquierda, tras repararse, se muestra en las figuras 3.6 y 3.9 del Capítulo 3, además de en la figura 4.5. Se puede observar en la gráfica superior el control de posición para el exoesqueleto ingravido (misma gráfica que la figura 3.6) y en la gráfica inferior se muestra el control de posición para el conjunto exoesqueleto-paciente, el cual, muestra un par más oscilante que en la figura 4.3 y 4.4, debido a que el comportamiento dinámico del motor deteriorado ha cambiado respecto al sistema anterior.

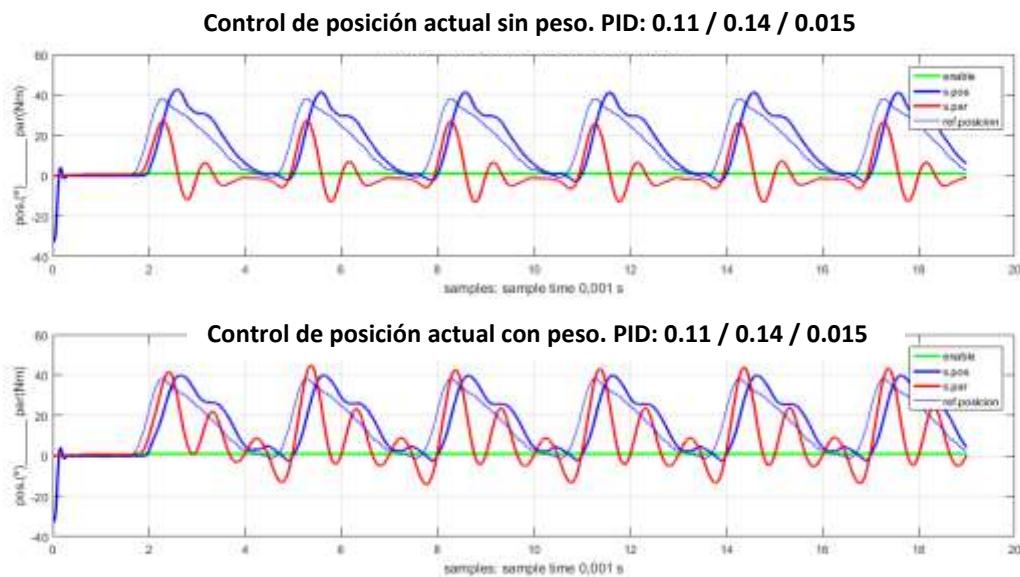


Figura 4. 5 Respuesta actual del control de posición (en azul grueso) al peso del propio exoesqueleto (gráfica superior) y al conjunto de paciente – exoesqueleto (gráfica inferior).

4.2.2 Análisis cinético de la marcha normal en el plano sagital para la articulación de la cadera. Obtención del $\tau_{ARTICULAR}$ mediante un modelo biomecánico computacional

En el Capítulo 3, se comentó que, para estudiar y interpretar la marcha humana mediante un modelo biomecánico, es necesaria la combinación de los datos obtenidos en el análisis cinemático y en el análisis cinético o dinámico de la misma. En el apartado 3.1.2 se explica cómo se realizó el análisis cinemático de la marcha humana, mediante la utilización de un sistema de captura de movimiento 3D, marcadores y cámaras de infrarrojos. Lo ideal para realizar el análisis cinético completo para obtener un modelo biomecánico, sería obtener los datos de fuerza a través de una plataforma dinamométrica

conectada a un sistema de captura de movimiento, y así, poder medir las fuerzas que los pies ejercen sobre el plano de apoyo. Estos datos junto con los obtenidos en cinemática permiten reconstruir las fuerzas, momentos o pares y la potencia que actúa sobre cada articulación durante la acción de caminar.

En este TFM no se dispone del instrumental necesario para realizar un análisis cinético ideal, por ello, la alternativa es realizar una simulación dinámica de la marcha humana mediante un modelo biomecánico computacional. Hay fundamentalmente dos enfoques para el estudio de la biomecánica del cuerpo humano:

- Cinética o dinámica directa: conociendo las fuerzas aplicadas a un sistema dinámico, el objetivo es determinar el movimiento resultante debido a esas fuerzas aplicadas.
- Cinética o dinámica inversa: conociendo la trayectoria de posición del sistema dinámico, el objetivo es hallar las fuerzas de dicho movimiento.

En el bloque del modelo biomecánico de la figura 4.1, se estiman los pares generados en la articulación de la cadera ($\tau_{ARTICULAR}$) durante la marcha humana, mediante la dinámica inversa que necesita como dato la cinemática del movimiento. A través del software *OpenSim* versión 3.3 (un simulador que permite generar simulaciones dinámicas de un modelo musculo esquelético) y de los datos reales de la cinemática de la cadera izquierda que se disponen, se resuelve el problema de la dinámica inversa. En el Anexo D se muestran los pasos a seguir para obtener los pares estimados de un paciente de 60 Kg mediante el software *OpenSim* de libre acceso desarrollado por la Universidad de Stanford.

4.3 Simulación del control de par adaptativo

La simulación del control de par adaptativo, se realiza mediante *Simulink* para los dos siguientes casos (los casos 1), 2) y 3) son explicados en el apartado 4.1):

- 4) Caso ideal: $\tau_{ARTICULAR1} = \tau_{PESO} - \tau_{VACÍO}$. Es el par que ejercería una persona sin exoesqueleto para el movimiento, obtenida a partir de señales reales.
- 5) Caso según modelo biomecánico: $\tau_{ARTICULAR2}$; par articular de la persona obtenido a través del software *OpenSim*.

La figura 4.6 muestra el par obtenido en 5), en comparativa con las señales de par vacío y par peso. Se aprecia que la señal del modelo biomecánico (ver Anexo D), sí proporciona una amplitud con los valores demandados por la articulación de la cadera durante la marcha humana normal (ver tabla 2.1), pero no posee una similitud con la alcanzada en el par peso, lo que implica obtener datos en simulación erróneos. Debido a esto y a su descompensación, se opta por trabajar con el caso 4) para evaluar el control de par adaptativo diseñado (ver Capítulo 5).

El Anexo E contiene todas las simulaciones realizadas para el caso 4). Se aplican distintos porcentajes de asistencia para poder simular y comparar el par de ayuda necesario que un paciente de 60 Kg. necesita para completar el ciclo de la marcha humana normal. En la figura 4.7, se muestran las simulaciones de los dos casos de asistencia más extremos según el par articular de 4) (ver Anexo E). Según este caso ideal y para cuando el paciente posee el 80% de la capacidad residual de un paciente sano ($K_H = 0.8$), en la simulación 4, se observa que la contribución del paciente es la máxima ($K_E = 1$), con lo cual el

exoesqueleto sólo asiste al paciente mediante un 10% de ayuda ($K_A = 0.1$). Por otro lado, en la simulación 10, la contribución del paciente es del 20% sobre la capacidad residual que el sujeto posee (K_E), por lo que en este caso el exoesqueleto ayuda al paciente en un 80% (K_A) para alcanzar el ciclo de la marcha humana normal.

Control de par adaptativo con el modelo biomecánico

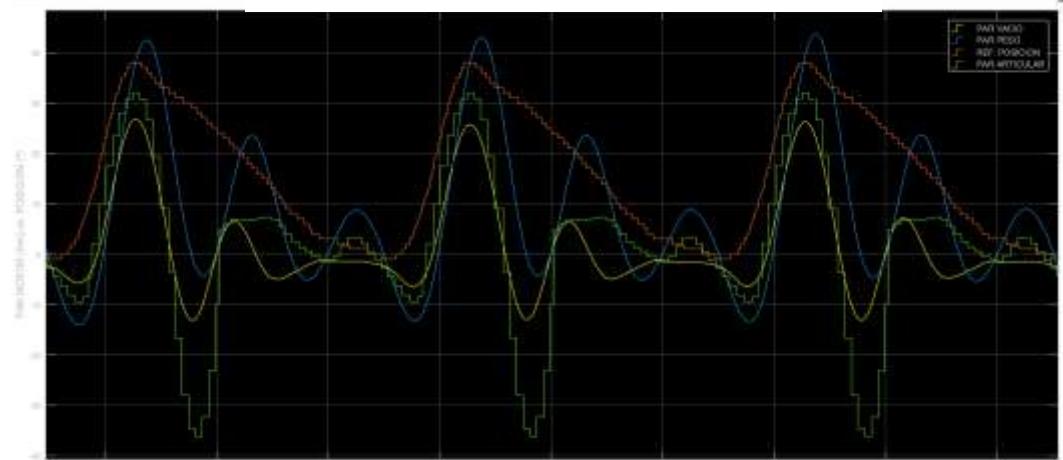
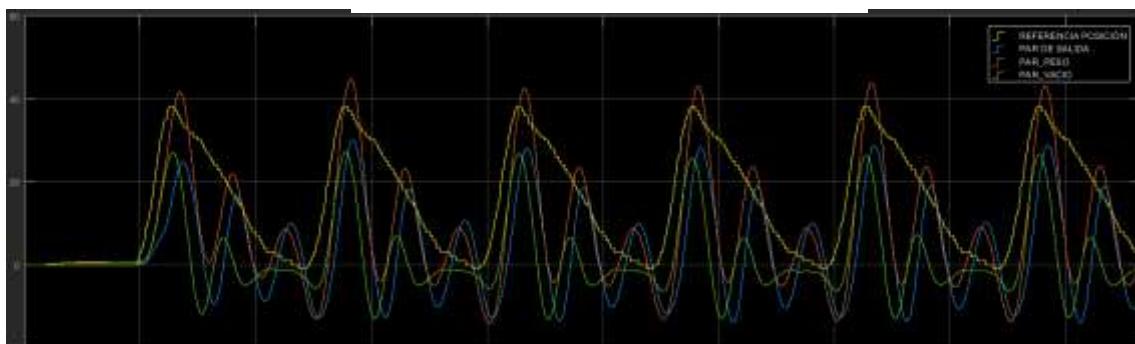


Figura 4. 6 Simulación que muestra la comparativa del par biomecánico obtenido mediante el software OpenSim (en verde), el par necesario para mover el robot ingravido (en amarillo) y el par ejercido por el motor para mover el conjunto paciente - exoesqueleto (en azul); la referencia de posición se muestra en color rojo

Simulación 4: KH=0.8, KE=1, KA=0.1, KP=3



Simulación 10: KH=0.8, KE=0.2, KA=0.8, KP=3

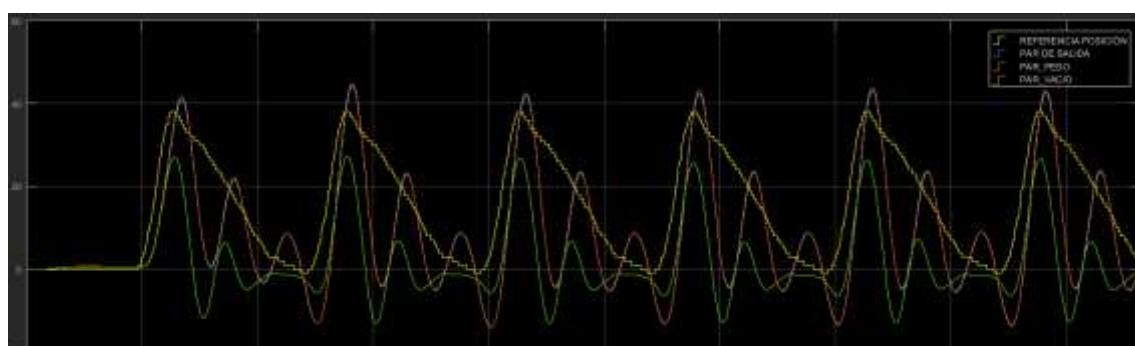


Figura 4. 7 Simulación del par de asistencia (casos extremos) para el caso en el que el paciente posee el 80% de la capacidad residual de un paciente sano en 4).

Estas simulaciones permiten verificar que el funcionamiento del esquema de control es el deseado, para distintas asistencias del robot y contribuciones del paciente.

4.4 Conclusiones

El modelo de control de par adaptativo no se ha podido evaluar mediante los pares estimados de un paciente de 60 Kg., a través del *software OpenSim*. En sustitución del mismo, se ha obtenido el par articular humano mediante la diferencia de señales reales del exoesqueleto con peso y en vacío, mostrando un resultado positivo del funcionamiento del control de par adaptativo.

CAPÍTULO 5

Identificación paramétrica y experimentación del control de par adaptativo en el exoesqueleto H1

Para conseguir que el control de par adaptativo desarrollado en el Capítulo 4, sea general y no haya que obtener los pares de consigna a partir del control de posición para diferentes trayectorias (amplitud del movimiento, frecuencia del paso) cada vez que se programa una trayectoria, se realiza una identificación del sistema, es decir, se obtiene su función de transferencia. De esta manera, una vez definida una trayectoria, se puede obtener a partir de dicha función las consignas de pares a aplicar al control adaptativo de par motor, sin necesidad de una experimentación previa. En este capítulo se desarrolla la identificación paramétrica posición-par para el sistema físico del exoesqueleto sin usuario (robot ingravido) y para cuando el sistema es utilizado por un usuario (robot con el peso corporal de un miembro inferior de un sujeto de 60Kg.). Posteriormente, los modelos matemáticos obtenidos de ambos sistemas, se integran en el esquema de control de par adaptativo para poder realizar en tiempo real las distintas terapias de rehabilitación que puedan plantearse.

5.1 Introducción a la identificación paramétrica

En ingeniería y en particular en el área de control, es indispensable tener un modelo matemático analítico que represente el sistema físico con el que se está trabajando. De esta manera, se puede analizar, predecir, simular, diseñar y controlar dicho sistema bajo las condiciones deseadas o de interés. En ocasiones, las leyes físicas que rigen cada proceso hacen que las ecuaciones diferenciales del mismo no puedan ser obtenidas. En este caso, para conseguir la representación de los modelos matemáticos dinámicos de sistemas desconocidos, se emplea el método experimental de identificación de sistemas.

La identificación paramétrica de sistemas dinámicos consiste en la deducción experimental, tanto de la estructura como de los parámetros de los modelos matemáticos, a partir de señales de entrada y de salida observadas del propio sistema. Dependiendo del proceso o aplicación, se pueden llevar a cabo métodos de identificación *off-line* ‘si se tiene la seguridad de que no habrá variaciones en la estructura del proceso ni en sus características dinámicas’, o métodos de identificación *on-line* ‘cuando el proceso puede variar su comportamiento dinámico con el tiempo’ [18]. Para el caso de sistemas de control

adaptativo, aplicado a sistemas en los que los parámetros pueden variar frecuentemente con el tiempo, se realiza la identificación paramétrica *on-line*. Básicamente el proceso de identificación paramétrica *off-line que se realiza aquí*, estima un modelo definitivo del sistema, el cual, reproduce con bastante aproximación sus características dinámicas, sin tener ningún conocimiento físico previo del mismo. Es decir, se calculan los coeficientes y las órdenes de la función de transferencia del sistema objetivo.

5.1.1 Identificación paramétrica del exoesqueleto H1

Con motivo de realizar la simulación en tiempo real del diseño del control de par adaptativo del exoesqueleto (ver Capítulo 4), se plantea la necesidad de realizar el modelado de dos sistemas físicos reales:

- Modelo matemático que represente el sistema físico del exoesqueleto en vacío (SFEV), es decir, cuando no está siendo utilizado por ningún usuario.
- Modelo matemático que represente el sistema físico del exoesqueleto con peso (SFEP), simulando que sí hay miembro inferior de un sujeto de 60 Kg., en este caso.

Para la estimación y obtención de las funciones de transferencia discretas de los sistemas físicos del exoesqueleto con y sin sujeto, se desarrolla un *script Matlab-Simulink* específico para realizar el proceso de identificación y obtener ambos modelos matemáticos.

En el Anexo F, se desarrolla la identificación posición-par del exoesqueleto únicamente para el SFEV, además de un breve análisis del comportamiento del sistema mediante la posición de sus polos y ceros en el plano z y de su respuesta en frecuencia. Se hubiera seguido el mismo proceso para el SFEP, sin embargo, esto se realizó cuando el exoesqueleto se había deteriorado, por lo que no pude obtenerse un modelo identificado para este caso.

Inicialmente se realiza la identificación del sistema SFEV introduciendo como entrada la señal generada por el potenciómetro en el modo control de posición, obteniendo un bajo porcentaje de similitud inferior al 50%. Para lograr una buena estimación se requiere al menos entre un 70% y un 80% de semejanza entre el sistema real y el modelo estimado. Por ello, se añade como entrada adicional la referencia de posición del propio control, logrando el mejor resultado con un 72,33% de semejanza para el sistema del exoesqueleto en vacío.

A partir de diagrama de Bode obtenido en el proceso de identificación, se deduce que el modelo se puede aproximar a un primer orden. Por lo que se ajusta a una función de primer orden. Las funciones de transferencia del modelo no simplificado y simplificado para SFEV, se muestran en las ecuaciones 5.1 y 5.2 respectivamente:

$$\frac{0.04237z^8 - 0.1251z^7 + 0.07198z^6 + 0.06126z^5 + 0.007682z^4 - 0.116z^3 + 0.05775z^2}{z^8 - 0.9815z^7 - 0.5893z^6 - 0.1675z^5 + 0.2225z^4 + 0.4628z^3 + 0.4431z^2 + 0.07962z - 0.4697} \quad (5.1)$$

$$\frac{0.0007375}{z - 0.999} \quad (5.2)$$

En la figura 5.1 Se puede comprobar que las salidas del modelo completo (azul) y del simplificado (rojo) presentan una gran similitud, y sus diagramas de Bode también (Anexo F), por lo que se valida el modelo simplificado.

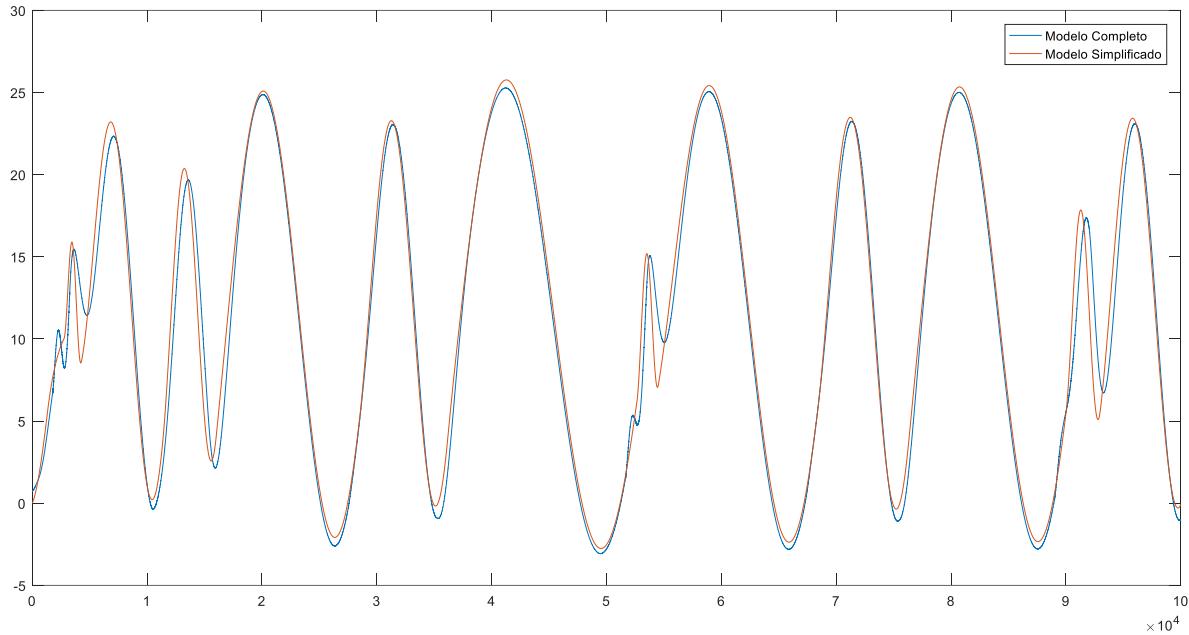


Figura 5. 1. Validación del modelo completo y del modelo simplificado.

La figura 5.2 se muestra la simulación real del par motor obtenido del modelo simplificado de posición – par identificado.

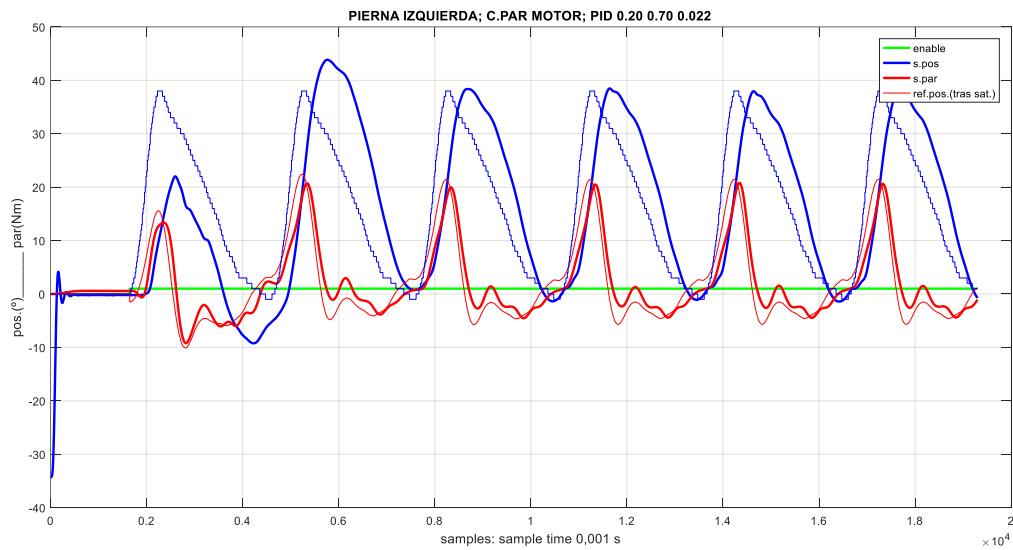


Figura 5. 2. Simulación real del modelo simplificado del sistema del exoesqueleto en vacío

5.2 Concepto del control de par adaptativo con identificación

Para poder realizar la simulación real del control de par adaptativo a pares externos en el hardware del exoesqueleto, es necesario añadir a éste el bloque de identificación posición-par desarrollado para SFEV y sustituir el bloque del modelo biomecánico de la figura 4.1, es decir, $\tau_{ARTICULAR1}$ por $\tau_{ARTICULAR2}$ (ver apartado 4.3). De esta manera, se puede verificar la posición y par motor del exoesqueleto correspondientes a las simulaciones realizadas para diferentes porcentajes de par de asistencia en un paciente de 60 Kg (ver Capítulo 4).

La experimentación con el exoesqueleto, es decir, la ejecución en tiempo real de las simulaciones realizadas mediante *Simulink* en el apartado 4.3, no han podido realizarse al no estar disponible el *hardware* del exoesqueleto por problemas técnicos en el PC104.

5.3 Conclusiones

Se ha desarrollado y simplificado la función de transferencia para el sistema del exoesqueleto en vacío, obteniendo un modelo simplificado de primer orden, mediante la identificación posición-par. De esta manera, el siguiente paso a realizar sería plantear el esquema completo con el sistema identificado con peso y así poder ejecutar en tiempo real el sistema de control de par adaptativo, cuando el equipo esté arreglado.

CAPÍTULO 6

Conclusiones y líneas futuras

El objetivo de este TFM era el de diseñar, implementar y evaluar, un control adaptativo de par motor en un exoesqueleto robotizado de miembros inferiores destinado a la rehabilitación motora bajo un paradigma de asistencia bajo demanda. Para ello había que sustituir el clásico control de posición rígido, por un control de par que permitiera adaptar el movimiento del exoesqueleto a fuerzas o pares externos. Estos, pueden provenir del propio par residual ejercido por el usuario o del ejercido por un rehabilitador si se utiliza el exoesqueleto para ejercicios de rehabilitación motora, dentro de un entorno clínico y bajo la constante supervisión de personal médico capacitado.

En la primera parte del TFM, se ha realizado un exhaustivo trabajo de ajuste de los controladores de posición y de par del exoesqueleto. Ello fue necesario debido a que los valores de los parámetros que inicialmente tenían los controladores PID, no proporcionaban el comportamiento suave y rápido deseado, necesario para su posterior utilización en el controlador adaptativo de par, objetivo fundamental del TFM. El ajuste se realizó para el exoesqueleto en vacío y con un peso equivalente a una persona de 60 Kg. Ambos son necesarios para el control adaptativo de par propuesto.

El control adaptativo de par diseñado, además de los controladores de par correctamente ajustados, necesita la estimación de par ejercido por la propia persona usuaria del exoesqueleto, a partir de su capacidad latente. Ésta se puede calcular de diversas formas: 1) a partir de medidas de activación muscular (EMG), de las que se pueden calcular los pares mediante modelos que relacionan los pares articulares con las señales EMG; 2) utilizando sensores de par que midan las fuerzas de interacción de la persona sobre el exoesqueleto; y 3) a partir de un modelo biomecánico que calcule los pares articulares ejercidos durante la cinemática de la marcha humana normal. En este trabajo, se ha optado por esta última opción para evaluar el método de control de una manera sencilla, para condiciones y capacidades latentes, desde una contribución parcial del paciente a una contribución casi total. Desafortunadamente, los pares estimados de un paciente de 60 Kg. mediante el *software OpenSim* de libre acceso desarrollado por la Universidad de *Stanford*, no han dado un resultado satisfactorio. En sustitución del mismo, se ha obtenido el par articular humano mediante un cálculo a partir de diferencia de señales reales de exoesqueleto con peso y en vacío. Se ha obtenido un resultado positivo del funcionamiento del control realizado para este caso ideal. Sin embargo, estos resultados no se han podido implementar finalmente en el exoesqueleto debido a un fallo en el sistema informático embarcado en dicha plataforma robótica.

Una línea de desarrollo futuro evidente sería actualizar tanto el *software* como el *hardware* del prototipo, para poder realizar el diseño de los seis controladores de ambos miembros inferiores a partir de lo conseguido en este TFM. Otra posible vía de desarrollo, sería aplicar un rediseño total del prototipo siguiendo el modelo del exoesqueleto H2 [18], aplicando un controlador híbrido que se adapta a la función residual del paciente durante la marcha. La experimentación del exoesqueleto con personas sanas y con pacientes, es una tarea fundamental para una validación completa del exoesqueleto y sus prestaciones, implicando en la utilización y ajuste posterior al personal especialista de rehabilitación.

ANEXO A

Entorno de control de tiempo real

Tanto en tiempo real como en tiempo no real, Simulink permite diseñar y simular modelos de sistemas de control sin necesidad de escribir un código fuente (en C, C++ o HDL). Simulink genera dicho código basándose en modelos gráficos de sistemas de control. De esta manera, se puede proceder a la depuración de posibles fallos antes de convertir estos sistemas a hardware.

Los modelos ejecutados por Simulink en un ordenador estándar, no suelen ser compatibles para simulaciones en tiempo real y tampoco suelen poseer entradas/salidas de hardware para adquisición de señales y comunicaciones en tiempo real. Como solución a este problema y a su vez, poder trabajar con la plataforma robótica H1, se usa la herramienta de software xPC Target (versión 5.5) de Mathworks y una plataforma hardware que posea módulos de entradas/salidas adecuados para los requerimientos del funcionamiento en tiempo real.

A.1 Aplicación xPC Target (versión 5.5)

La toolbox xPC Target (versión 5.5) tiene como finalidad que modelos desarrollados mediante *Simulink* en un ordenador convencional, se ejecuten en tiempo real en un sistema embebido independiente conectado a un sistema físico a evaluar. Esto, es posible gracias a que esta aplicación cuenta con una gran cantidad de dispositivos con entradas/salidas de diferentes fabricantes. Dichos módulos tienen forma de bloques dentro del modelo de *Simulink* y son los que permiten la comunicación con el sistema físico real.

Esta aplicación ofrece la posibilidad de implementar, visualizar y analizar sistemas en tiempo real usando el *hardware* de un ordenador estándar. Además, puede llevar a cabo procesos de obtención rápida de un prototipo o *Rapid Prototyping* (RP) y de simulación con *hardware* en el lazo o *Hardware-in-the-loop* (HIL)

El entorno para construir aplicaciones en tiempo real con xPC Target (versión 5.5) consiste en un *Host PC* u ordenador convencional, un *Target PC* o equipo embebido independiente y un *hardware* real a evaluar (figura A.1). Aunque RP y HIL son procesos con propósitos diferentes, comparten el mismo procedimiento para obtener el código fuente correspondiente al modelo de un controlador o al modelo de una planta respectivamente. En ambos procesos, primero se diseña el modelo pertinente en *Simulink* instalado en el *Host PC*. Posteriormente, el *Target PC* ejecuta en tiempo real el modelo diseñado según el escenario

de evaluación demandado: RP o HIL. Finalmente se realiza una simulación real del modelo diseñado en el *hardware* a evaluar, mostrando este último, las mismas características de frecuencia, velocidad y tiempo que el controlador o planta modelado.

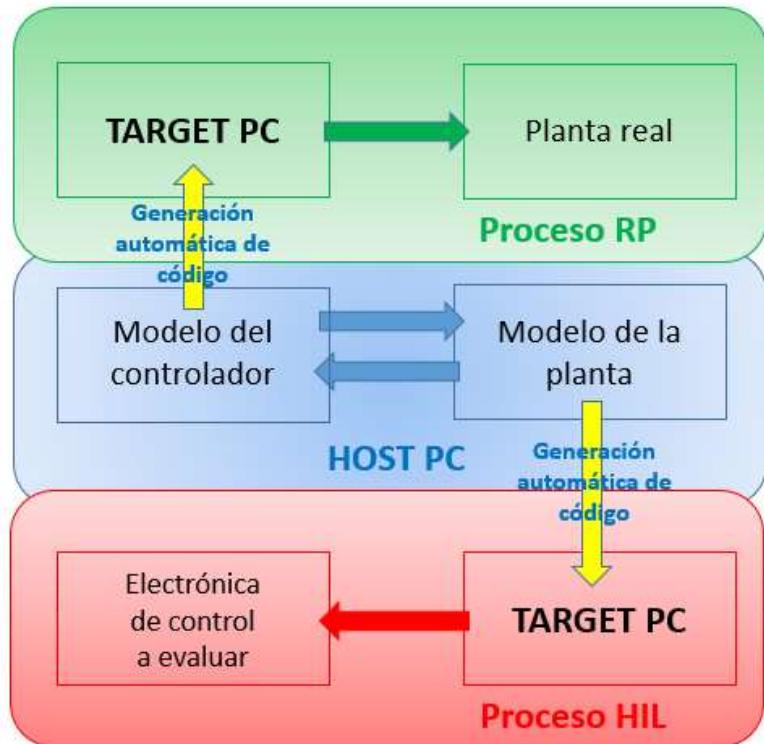


Figura A. 1 Entorno para construir aplicaciones en tiempo real para los procesos de obtención rápida de un prototipo (RP) y de simulación con hardware en el lazo Hardware-in-the-loop (HIL).

A.2 Proceso para la obtención rápida del prototipo del exoesqueleto H1

Antes de realizar el proceso de construcción del modelo diseñado para poder trabajar en tiempo real, es importante saber cómo proceder dependiendo del sistema operativo que posea el *Host PC*.

La versión 5.5 de la aplicación *xPC Target* está diseñada para *Host PC*'s que tienen sistemas operativos de *32 bits*. Permitiendo diferentes maneras de interacción entre el usuario y los equipos durante la ejecución en tiempo real del modelo diseñado. También es posible trabajar con esta versión en un sistema de *64 bits*, aunque el programa sólo admite interactuar con la aplicación en tiempo real mediante comandos. Otra opción muy interesante de trabajo y que garantiza mayor comodidad, es la creación de una máquina virtual con un sistema operativo de *32 bits* en un disco duro externo o en el propio *Host PC* de *64 bits*.

En los siguientes apartados, se explica cómo obtener una aplicación en tiempo real del modelo diseñado para cuando el sistema operativo del *Host PC* es de *32 bits*, o para el caso en el que el *Host PC* posea *64 bits*.

A.2.1 Proceso para la obtención del modelo según *Host PC* de 32 bits

A.2.1.1 Instalación, configuración y conexión de los equipos (32 bits)

El paquete *xPC Target* versión 5.5 demanda que el *Host PC* posea lo siguientes requerimientos software:

- *Matlab R2011a*
- *Toolbox Simulink*
- *Toolbox Simulink Coder*
- *Toolbox xPC Target*
- Compilador de código C (*Visual Studio 2010*)

Para construir la aplicación en tiempo real, se debe realizar la configuración de los equipos. Basta con ejecutar el comando *xpcexplr* en la ventana de comandos de *Matlab*, dando acceso a la interfaz gráfica del explorador de la aplicación. Otra opción de configuración es mediante la introducción de comandos. Las figuras A.2 y A.3, muestran cómo configurar el *Host PC* y el *Target PC* de forma rápida y sencilla a través de la interfaz gráfica. La ventana de la interfaz se divide en dos mitades: en la mitad izquierda aparece un esquema indicando el *Host PC* y el *Target PC* por defecto (un solo equipo puede trabajar con varios *Targets* a la vez) y en la mitad derecha se observan los parámetros y opciones del equipo configurado.

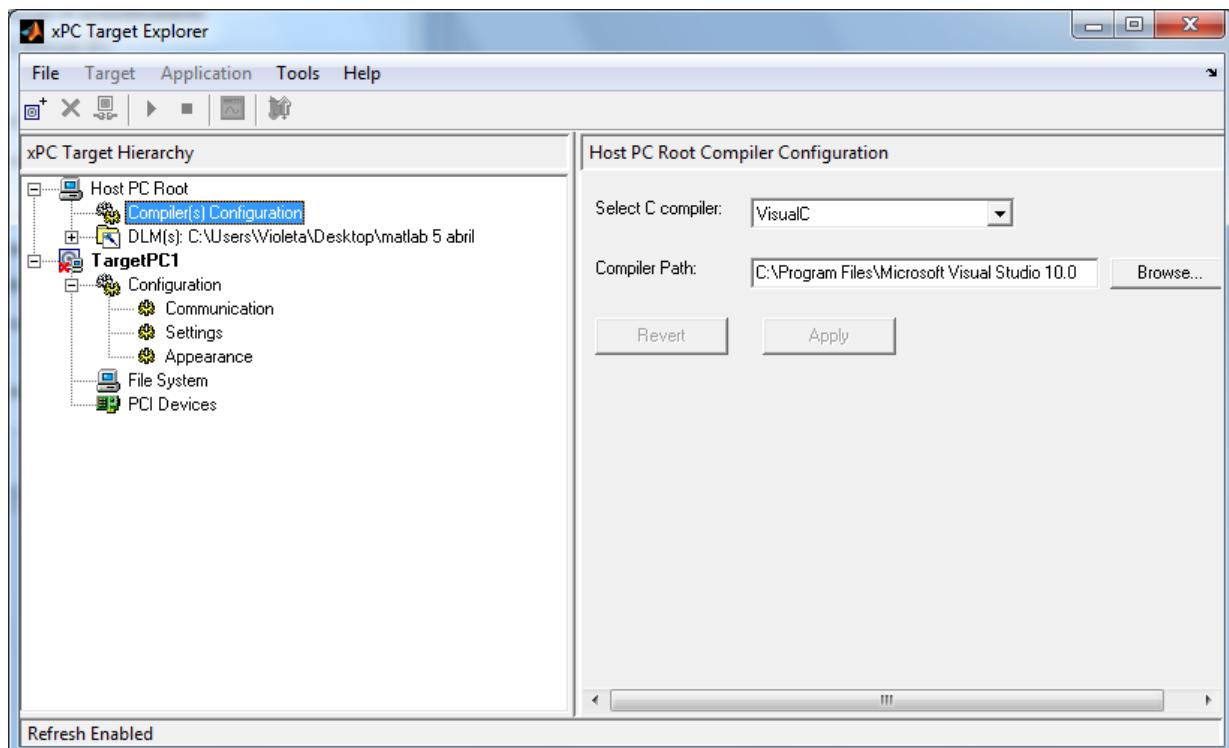


Figura A. 2 Configuración del *Host PC* a través del explorador de la aplicación.

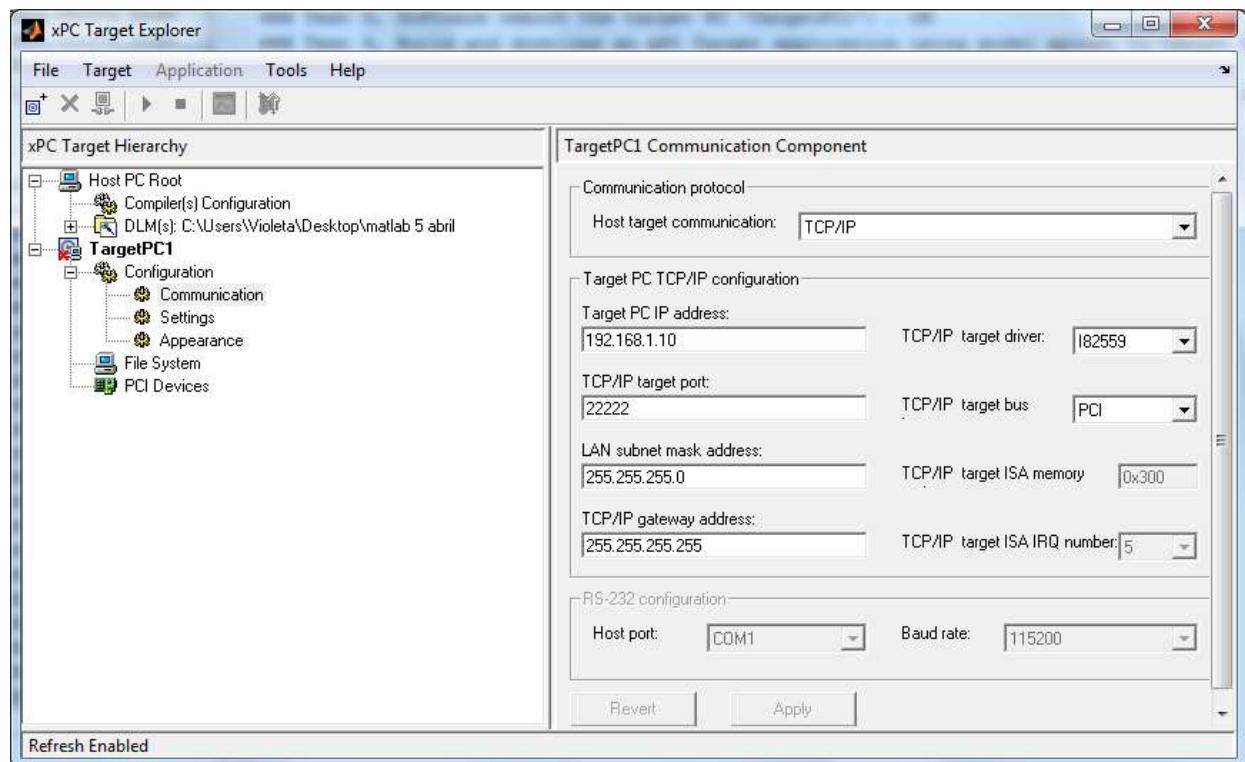


Figura A. 3. Configuración del Target PC a través del explorador de la aplicación.

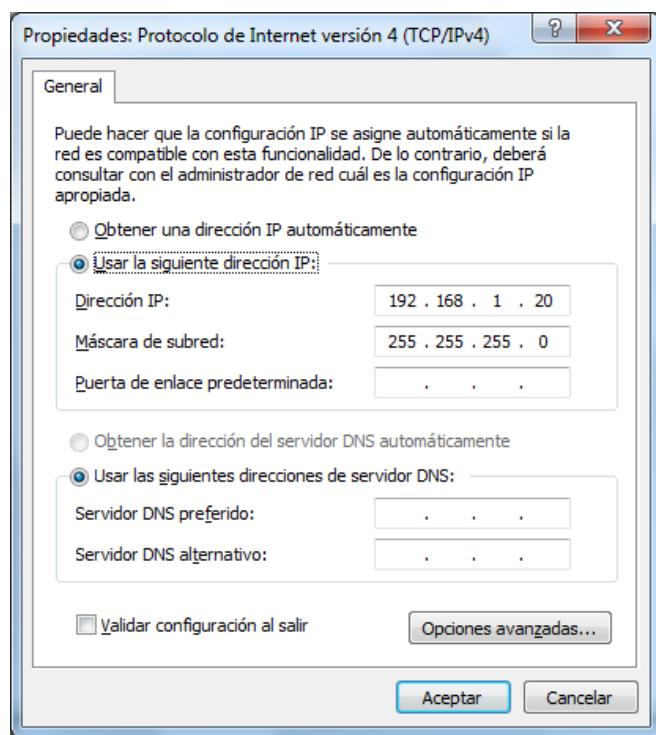


Figura A. 4 Conexión entre los equipos Host PC y Target PC.

La conexión entre el *Host PC* y el *Target PC* puede ser de dos tipos: conexión *Ethernet* o conexión puerto serie *RS-232*. En el caso que atañe este TFM, los equipos están conectados mediante un cable *Ethernet* usando el protocolo de comunicación *TCP/IP*. Para que la comunicación sea correcta, se deben configurar la dirección IP (fija asignada al *Target PC*) y la máscara de subred en las propiedades *TCP/IP* de la red de Windows del *Host PC* (figura A.4).

La interacción entre el usuario y el *Target PC* durante la ejecución de la aplicación en tiempo real se puede realizar principalmente:

- Mediante la interfaz gráfica del propio explorador de la *toolbox xPC Target*.
- A través de la ventana de comandos de *Matlab* y mediante las instrucciones adecuadas.
- Usando la *Guide User Interface (GUI)* de *Matlab* ...etc.

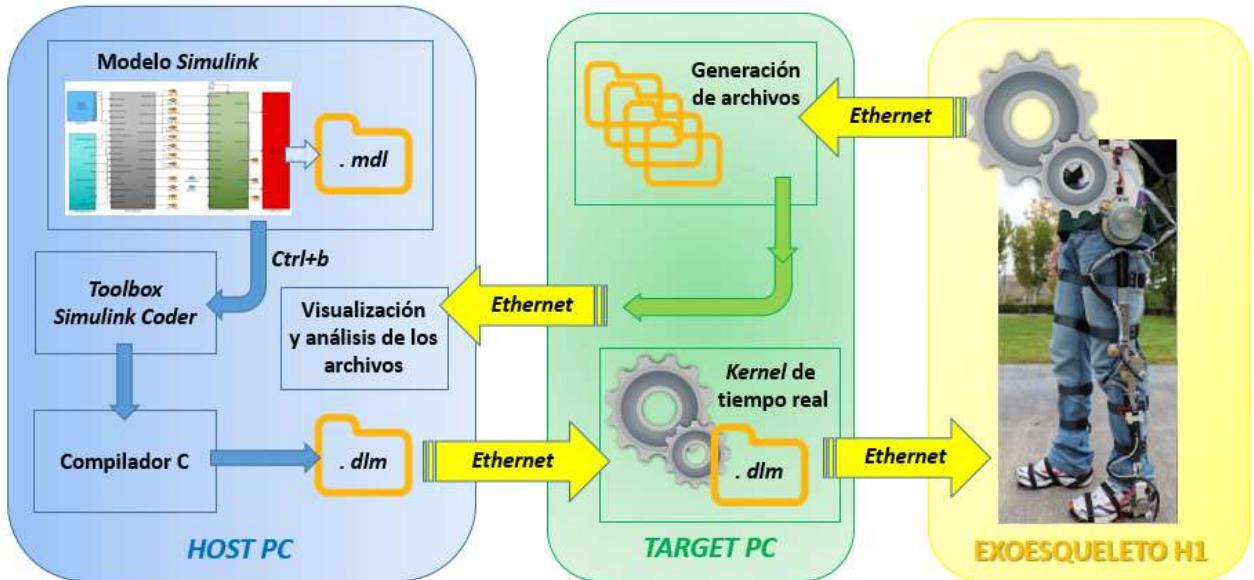


Figura A. 5 Proceso de obtención rápida del prototipo

Tras instalar, configurar y conectar los equipos, se debe comprobar que las conexiones entre el *Target PC* y el sistema real son las adecuadas. Si todo está correcto, se procede a llevar a cabo el proceso de obtención rápida del prototipo (figura A.5). Previamente se crea el modelo en *Simulink* con la extensión *mdl*. Luego, tecleando *Ctrl+b* se construye y compila el modelo diseñado. Este proceso se inicia de manera automática con la generación del código C del modelo mediante la *toolbox Simulink Coder*. Posteriormente un compilador de código C (*Visual Studio 2010*) crea un archivo ejecutable del modelo con la extensión *dlm*. A este archivo se le llama aplicación en tiempo real y se descarga en el *Target PC* mediante conexión *Ethernet*. En este caso, un ordenador industrial PC104 se inicializa a través de la imagen del modelo, la cual, se encarga de encender su *kernel de tiempo real*. Entonces, es cuando la aplicación del modelo diseñado se ejecuta en el *hardware* físico real (plataforma robótica H1) durante el tiempo de ejecución establecido.

Se pueden adquirir señales y sintonizar parámetros antes, durante y después de la ejecución de la aplicación en tiempo real. Esto se consigue mediante la creación de osciloscopios en la ventana del explorador; según el tipo de osciloscopio elegido (tipo *host*, tipo *target* o tipo archivo) se obtiene una información u otra. En este trabajo, se han utilizado osciloscopios de tipo archivo que permiten recoger información durante el tiempo de ejecución en archivos ubicados en la memoria del *Target PC*. Posteriormente son descargados en forma de variable en el entorno de *Matlab*, con la finalidad de poder analizar *offline* las señales proporcionadas por el sistema real.

A.2.1.2 Protocolo para poner en marcha el exoesqueleto (32 bits)

Tras la instalación, configuración y conexión de los equipos (ver apartado A.2.1.1 de este anexo) se procede a la puesta en marcha del exoesqueleto mediante los siguientes pasos:

1. Dentro del entorno de *Matlab* seleccionar el path de trabajo donde se crearán los archivos correspondientes de la aplicación en tiempo real.
2. Encender el *Target PC* activando el interruptor negro (tipo *rocker*) de alimentación del PC104 situado en el lateral de la caja de conexiones y mantener el interruptor de los actuadores desactivado situado en la parte superior de la caja de conexiones.

Si el monitor que está conectado al *Target PC* muestra la *interface* indicada en la figura A.6, el arranque es correcto. De lo contrario, actuar según lo indicado en el Anexo B.

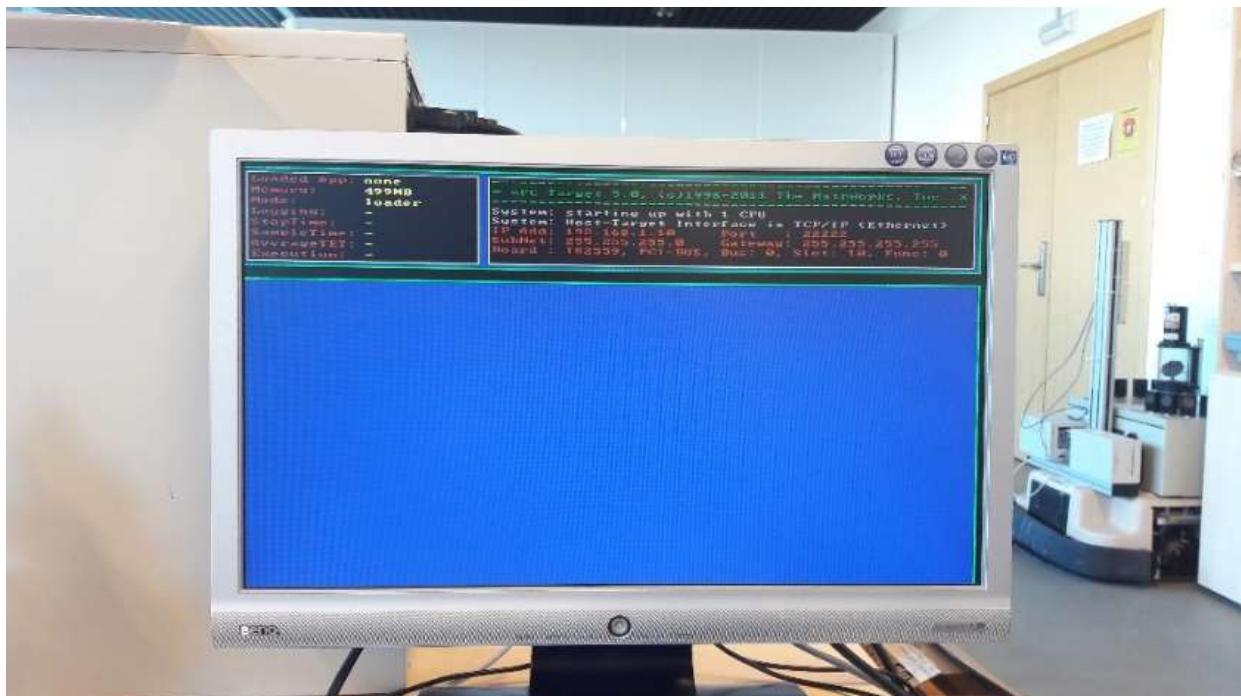
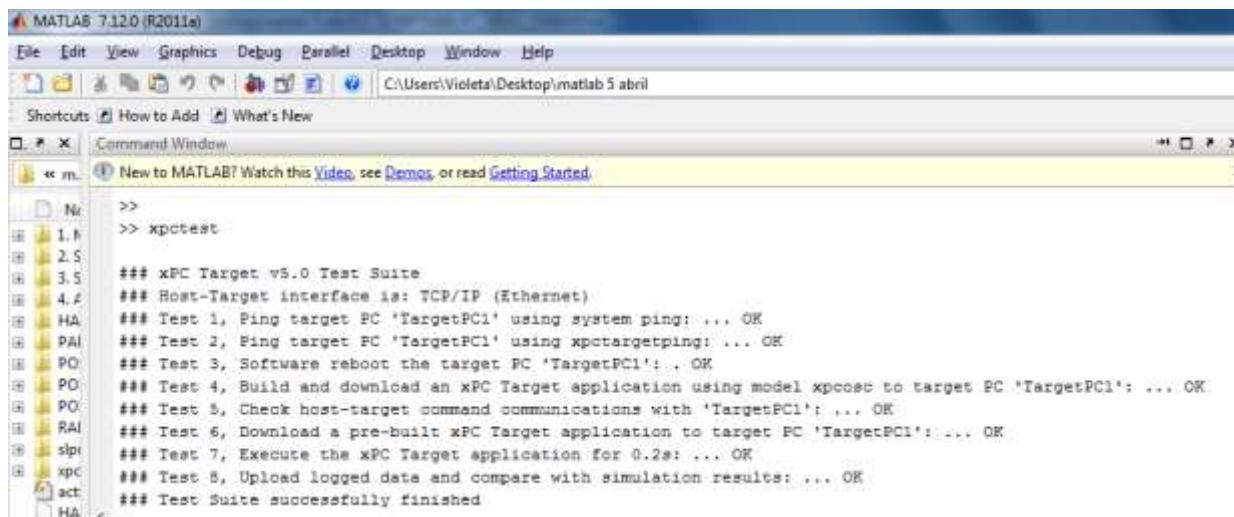


Figura A. 6. Interface de arranque del PC104

3. Ejecutar el comando xpctest en la ventana de comandos de *Matlab* para comprobar que la instalación, comunicaciones y el proceso de compilación de prueba funcionan correctamente (figura A. 7). En caso de cualquier fallo consultar.



```

>> xpctest
## xPC Target v5.0 Test Suite
## Host-Target interface is: TCP/IP (Ethernet)
## Test 1, Ping target PC 'TargetPC1' using system ping: ... OK
## Test 2, Ping target PC 'TargetPC1' using xpctargetping: ... OK
## Test 3, Software reboot the target PC 'TargetPC1': ... OK
## Test 4, Build and download an xPC Target application using model xpcosc to target PC 'TargetPC1': ... OK
## Test 5, Check host-target command communications with 'TargetPC1': ... OK
## Test 6, Download a pre-built xPC Target application to target PC 'TargetPC1': ... OK
## Test 7, Execute the xPC Target application for 0.2s: ... OK
## Test 8, Upload logged data and compare with simulation results: ... OK
## Test Suite successfully finished

```

Figura A. 7 Proceso para corroborar desde la ventana de comandos de *Matlab* que las conexiones entre ambos equipos son correctas.

4. Ejecutar el comando xpcexplr para acceder el explorador de la toolbox *xPC Target* y trabajar en su entorno (figura A.8).

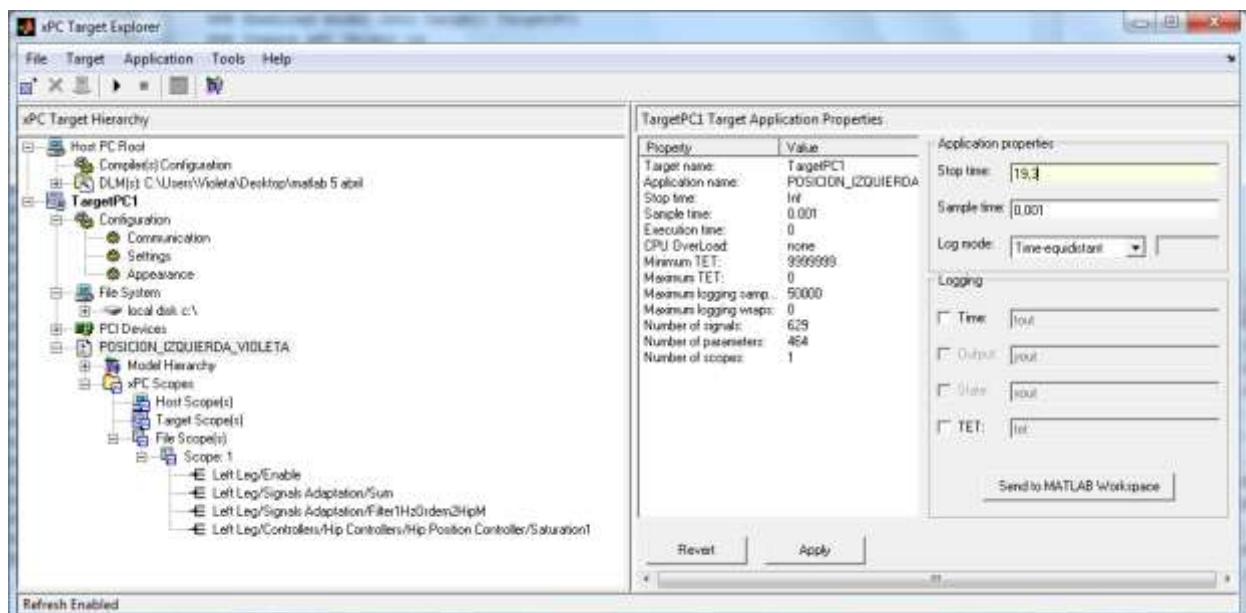


Figura A. 8. Interface que muestra la aplicación en tiempo real construida y compilada, el tiempo de ejecución deseado, el tiempo de muestreo y los oscilloscopios tipo archivo con las señales requeridas para visualizar y analizar el control de posición de la cadera izquierda.

5. Conectar el *Target PC*, éste siempre debe estar conectado antes de cargar cualquier aplicación con extensión *dlm*. Para conectar el *Target PC* hay que clicar con el botón derecho sobre *TargetPC1* en el panel izquierdo del explorador.
6. Abrir el modelo del sistema de control del exoesqueleto y colocar el ratón sobre él para seleccionarlo. Hecho esto, pulsar en el teclado *Ctrl+b* para construir y compilar la aplicación en tiempo real del prototipo (este proceso lleva un par de minutos). Esta aplicación aparecerá por defecto cargada en el *Target PC* (figura A.8).
7. En el explorador y dentro de las propiedades de la aplicación, indicar en segundos el tiempo de ejecución deseado y el tiempo de muestreo del modelo diseñado; 0.001 seg. es el tiempo de muestreo del exoesqueleto (figura A.8).
8. Añadir los osciloscopios necesarios de tipo archivo y añadir dentro de ellos las señales que se quieren visualizar y analizar. Después conectar los osciloscopios. (figura A.8).
9. Asegurarse de que el exoesqueleto está en posición de equilibrio (ver Capítulo 3 apartado 3.1.3). Posteriormente alimentar los actuadores mediante el interruptor de tipo palanca (*toggle*) situado en la parte superior de la caja de conexiones.
10. Ejecutar la aplicación en tiempo real clicando con el botón izquierdo sobre el icono de *play*. Entonces el exoesqueleto entra en funcionamiento durante el tiempo de ejecución previamente establecido. Si se desea parar antes de que finalice el tiempo de ejecución, clicar con el botón izquierdo sobre el icono de *stop*.
11. Terminada la ejecución, desactivar el interruptor que alimenta los actuadores.
12. A continuación, se exportan al espacio de trabajo de *Matlab* en forma de estructura, los archivos de datos generados durante la ejecución. Esto se realiza desde la opción *export to workspace* de los osciloscopios.
13. Finalmente, se representan gráficamente las señales seleccionadas según propósito, con objeto de tener una mejor comprensión del comportamiento del sistema físico. De esta manera se puede realizar un completo análisis visual y paramétrico del exoesqueleto.

A.2.2 Proceso para la obtención del modelo según *Host PC de 64 bits*

A.2.2.1 Instalación, configuración y conexión de los equipos (64 bits)

Cuando el entorno de trabajo posee un *Host PC* de 64 bits, la instalación y conexión de los equipos se realiza de la misma manera que la indicada en el apartado A.2.1.1 de este mismo anexo. Por el contrario, la configuración de los equipos se realiza mediante la ejecución de los siguientes comandos. (figura A.9).

```
% Configuración de la dirección del compilador, VisualC 2010
% Configuración de los equipos Host PC y Target PC
xpcsetCC('VISUALC', 'C:\Program Files (x86)\Microsoft Visual Studio 10.0')
.
setxpcenv('HostTargetComm','TcpIp','MulticoreSupport','off',...
  'TcpIpTargetAddress','255.255.255.0','TargetBoot','CDBoot',...
  'TargetMACAddress','F8:32:E4:FA:9E:17','TargetRAMSizeMB','Auto','TcpIpGateway','255.255.255.255',...
  'TcpIpSubNetMask','255.255.255.0','TcpIpTargetAddress','192.168.1.10',...
  'TcpIpTargetBusType','PCI','TcpIpTargetDriver','I82559',...
  'TcpIpTargetPort','22222')
%
% Test
xpctest
```

Figura A. 9 Comandos para la configuración del equipos desde un *Host PC de 64 bits*.

A.2.2.2 Protocolo para poner en marcha el exoesqueleto (64 bits)

1. El proceso es muy similar al indicado en el apartado A.2.1.2. Sólo hay una diferencia, que la interacción durante el tiempo de ejecución entre el usuario y la aplicación, únicamente se puede realizar mediante comandos. Para ello, es aconsejable guardar todas las líneas de comando en un único archivo *script* propio del entorno de *Matlab*.
2. La figura A.10 muestra todos los comandos y los comentarios del código desarrollado para: construir y compilar el modelo, configurar el tiempo de ejecución y de muestreo, inicializar los osciloscopios y ejecutar la aplicación. Se usa como ejemplo el control de posición de la cadera izquierda.

```
% Como crear el archivo .dlm del modelo diseñado en Simulink
app='POSICIONIZQUIERDA_VIOLETA'; % Nombre de la aplicación
rtwbuild(app); % Construye y compila

tg = xpc; % Creación del objeto xPC Target
tg.SampleTime = 0.001; % Tiempo de muestreo en msegs
tg.StopTime = 19,3; % Tiempo de ejecución en segs

sc = getscope(tg); % Definición osciloscopios
start(tg) % start objeto
start(sc) % start osciloscopios
```

Figura A. 10. Comandos para construir y compilar el modelo diseñado en Simulink, definir osciloscopios y ejecutar la aplicación.

3. Cuando el tiempo de ejecución se termina, para extraer los datos de la simulación se deben ejecutar las siguientes líneas de código por cada osciloscopio (figura A.11). De esta manera, se obtiene en el espacio de trabajo del entorno de *Matlab* dos variables por cada osciloscopio: una contiene el tiempo de ejecución y la otra posee los datos de la señal a evaluar (figura A.12).

```
% Extracción de los ficheros
% Adjuntar al Host PC los archivos del Target PC
f=xpctarget.fs;
% Abrir el fichero, leer los datos y cerrar el fichero (4 señales)
h1=fopen(f,'data1.dat'); % Señal enable
data1=fread(f,h1);
fclose(f,h1);
h2=fopen(f,'data2.dat'); % Señal posición sensor
data2=fread(f,h2);
fclose(f,h2);
h3=fopen(f,'data3.dat'); % Señal par motor
data3=fread(f,h3);
fclose(f,h3);
h5=fopen(f,'data5.dat'); % Señal posición de referencia
data5=fread(f,h5);
fclose(f,h5);
% Descomprimir los datos
x1=readxpcfile(data1);
x2=readxpcfile(data2);
x3=readxpcfile(data3);
x5=readxpcfile(data5);
```

Figura A. 11 Comandos para extraer en el Host PC los datos de los ficheros creados en el Target PC durante la simulación del control de posición del exoesqueleto H1.

```
% Asignación de variables
X10_1=x1.data(:,1); % enable
X10_2=x2.data(:,1); % sensor posición
X10_3=x3.data(:,1); % sensor par motor
X10_5=x5.data(:,1); % consigna del tipo de control 'setpoint'
% Asignación tiempos
tX10_1=x1.data(:,2); % tiempo enable
tX10_2=x2.data(:,2); % tiempo sensor posición
tX10_3=x3.data(:,2); % tiempo sensor par M
tX10_5=x5.data(:,2); % tiempo consigna posición
```

Figura A. 12 Comandos para obtener las variables y los tiempos de las mismas, en el espacio de trabajo de Matlab, para su posterior representación.

ANEXO B

Protocolo de reinicio del sistema embebido PC104

Este anexo muestra los diferentes pasos de cómo proceder para reiniciar el sistema embebido del exoesqueleto H1 cuando se presenta un error en su disco de arranque tras iniciar el sistema.

B.1 Descripción y solución del problema

A lo largo de la ejecución de este TFM, el sistema embebido *PC104* falló repentinamente y dejó de funcionar de manera correcta. El problema no estaba claro debido a que sí se encendía el sistema, pero no cargaba los ejecutables con extensión *d/m* de los modelos diseñados en *Simulink*.

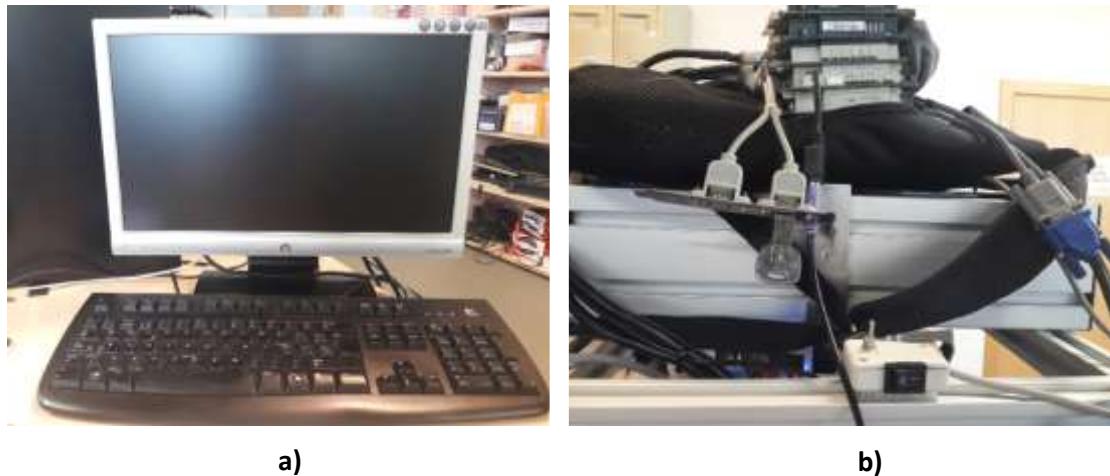
Con motivo de solventar el incidente y averiguar el fallo desconocido del embebido, se decide conectar al mismo un monitor y un teclado. De esta manera, se pudo observar que una de las particiones de su disco de arranque estaba corrompida. La solución fue copiar sobre el disco duro estropeado, una imagen del disco duro de un *PC104* no deteriorado a través de un sistema operativo externo.

A continuación, se muestran los pasos a seguir para realizar correctamente el proceso de reinicio del sistema embebido y poder trabajar posteriormente con el exoesqueleto *H1*. Para ello se necesita:

- Un USB que contenga la imagen del disco duro de un *PC104* y un disco de arranque con una imagen ISO del sistema operativo *Ubuntu* versión 11.10.
- Adaptadores especiales que permitan conectar al embebido: un monitor, un teclado y un pendrive.

B.1.1 Protocolo de reinicio del *PC104* ante un error en su disco duro.

1. Conectar al sistema embebido un monitor y un teclado (figura B.1).

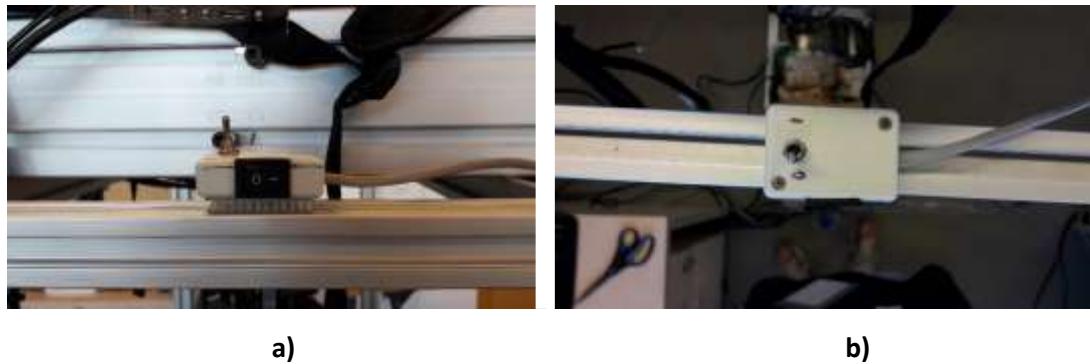


a)

b)

Figura B. 1 Equipo y conexiones necesarias para la visualización del estado del PC104 **a)** Hardware: monitor y teclado **b)** Conexiones y adaptadores del hardware utilizado, junto al USB que contiene la imagen del disco duro del PC104 y el sistema Ubuntu 11.10.

2. Activar el interruptor de alimentación del PC104 (figura B.2.a)



a)

b)

Figura B. 2 Caja de conexiones que empalma los interruptores de alimentación del prototipo **a)** Interruptor negro tipo rocker situado en el lateral encargado de alimentar el sistema embebido **b)** Interruptor de tipo palanca (toggle) localizado en la parte superior que acciona ('1') y/o para ('0') los actuadores de corriente continua.

3. Tras encender el equipo pueden observarse dos posibilidades. Por un lado, la interface del sistema embebido puede indicar un mensaje de error en el disco de arranque (figura B.3), cuya solución se muestra en las siguientes instrucciones de este anexo. Por otro lado, al encender el equipo puede aparecer la interface indicativa de un correcto funcionamiento del sistema embebido PC104 (figura B.8). En este caso, seguir con la puesta en marcha del exoesqueleto desarrollada en el Anexo A según los bits que posea el sistema operativo del Host PC (apartado A.2.1.2 para un sistema de 32 bits y/o apartado A.2.2.2 para un sistema 64 bits).



Figura B. 3 Interface del sistema embebido que muestra el error en el disco de arranque tras activar el interruptor de alimentación del mismo.

4. Introducir el USB en el adaptador (figura B.1.b)
5. Pulsar *Ctrl+Alt+Supr* para inicializar el sistema operativo *Ubuntu* versión 11.10

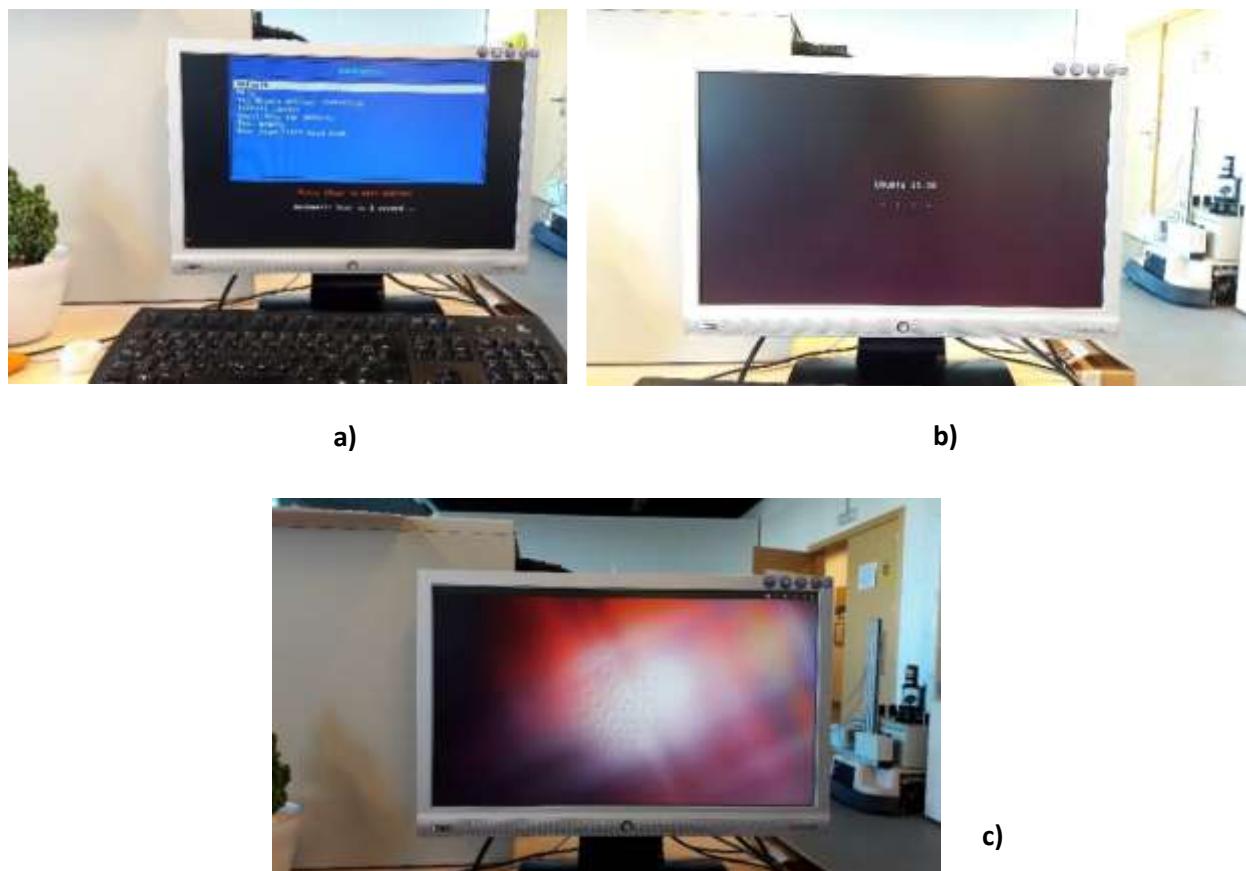


Figura B. 4. Interfaces que muestran el proceso de inicialización del sistema operativo *Ubuntu* **a)** Interface de tiempo de espera de carga **b)** Interface de carga del sistema **c)** Interface de inicio del sistema operativo *Ubuntu* versión 11.10.

6. Pulsar *Ctrl+Alt+F1* para acceder a la consola del sistema operativo *Ubuntu* (figura B.5).



Figura B. 5 Interface de la consola del sistema operativo Ubuntu, mediante la que se introducen las instrucciones pertinentes según el cometido a realizar.

7. Introducir las siguientes instrucciones para realizar una copia bit a bit de la imagen del disco de arranque.

```
sudo mount /dev/sda1 /mnt
ls /mnt
cd /cdrom
ls
sudo if=/cdrom/imageDiscoDuro.img of=/dev/sda
```

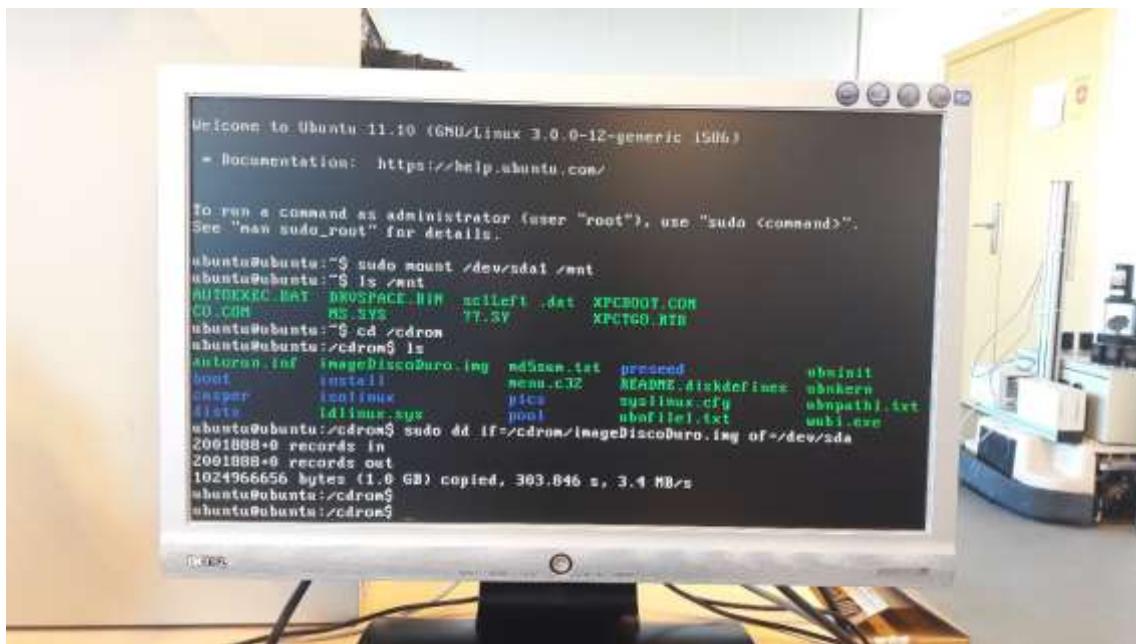


Figura B. 6 Interface que muestra las instrucciones necesarias para realizar bit a bit la copia del disco de arranque no deteriorado y el tiempo que conlleva dicho proceso.

8. Acabado el proceso, reiniciar el sistema mediante la instrucción ‘*sudo reboot*’.
9. Quitar el USB y pulsar *enter*.



Figura B. 7 Interface que indica el cierre del sistema Ubuntu, por lo que ya se puede retirar el pendrive que lo contiene. Tras ello, se pulsa la tecla enter para finalizar el proceso.

10. Sistema embebido reiniciado.

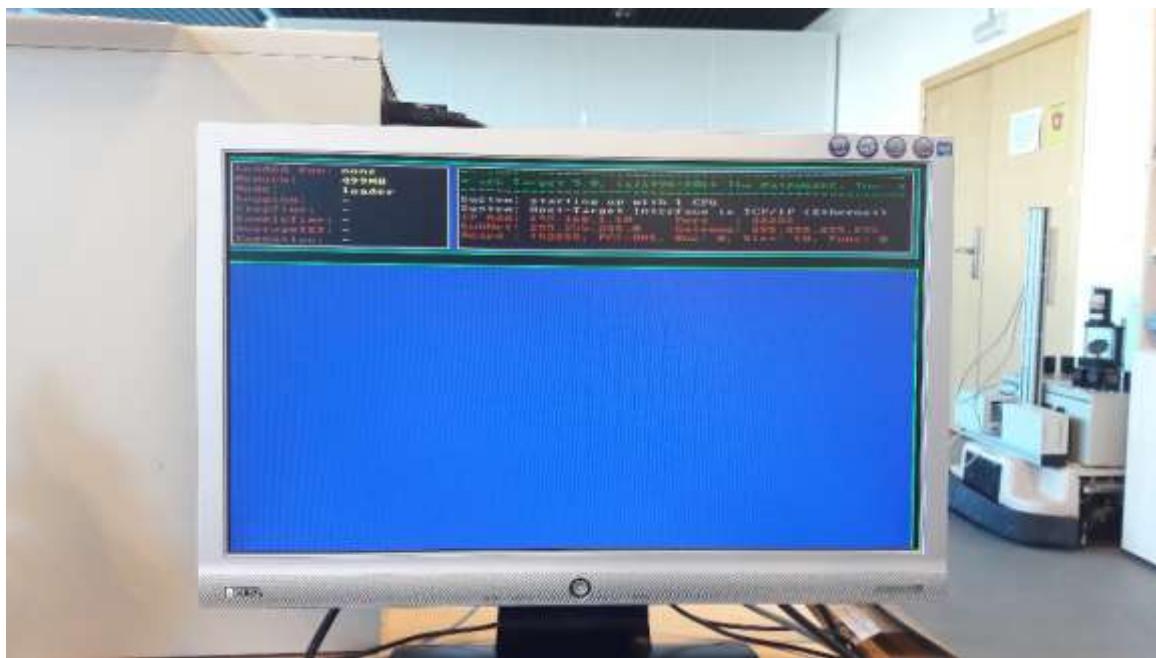


Figura B. 8 Interface indicativa del correcto funcionamiento del sistema embebido PC104.

ANEXO C

Obtención de la consigna de par motor

En documento contiene el script de Matlab creado para obtener la consigna de referencia del control de par motor del actuador de la cadera izquierda.

C.1 Código para la obtención de la consigna de par motor

A continuación, en las figuras C.1 y C.2, se muestran los comandos desarrollados para obtener la consigna de referencia de par motor a través del control de posición del exoesqueleto H1 ingravido.

```
%> OBTENCIÓN REBAL PAR SIN PESO. PARTIENDO DEL CONTROL DE POSICIÓN |

% PAR VACÍO. EXOESQUELETO INGRÁVIDO
% CAPÍTULO 3: BIOMEÁNICA DE LA MARCHA HUMANA. CONTROLADORES DE POSICIÓN Y PAR MOTOR PARA LA ARTICULACIÓN DE LA CADERA
% ALGORITMO PARA OBTENER LA CONSIGNA DEL CONTROL DE PAR MOTOR

rme_período=zeros(5,1); % vector columna (1º períodos/ iteraciones)
for p=1:5
    ini=1645+(p-1)*3000;
    fin=ini+2999;
    % variable medida sensor posición
    ref_pos_18 = V25_cpos15.data(ini:fin,2); % 3000 muestras señal sensor posición (una xanada) correspondiente a la iteración en cuestión
    ref_pos_diez_18= ref_pos_18(1:15:end);% señal ref_pos decimada.

    %variable medida sensor par motor
    ref_par_18 = V25_cpos15.data(ini:fin,3); % 3000 muestras señal sensor posición (una xanada) correspondiente a la iteración en cuestión
    ref_par_diez_18= ref_par_18(1:15:end);% señal ref_par decimada.

    %Grafica que muestra el periodo de todas las iteraciones
    % la primera gráfica muestra la señal del sensor de posición y la señal del sensor de par motor.
    % la segunda gráfica muestra las señales anteriores pero decimadas.
    figure (2);
    % señales originales
    subplot (2,1,1), plot (ref_pos_18,'b'); grid on; hold on; plot (ref_par_18,'r');
    title('LEFT HIP SEÑALES ORIGINALES Sensor POSICIÓN Y PAR MOTOR/ emble 1.645 seg');
    xlabel('sample time 0,001 s'); ylabel('pos.18/ par.18');
    legend('señal posición original','señal parM original');
    % señales decimadas
    subplot (2,1,2), plot (ref_pos_diez_18,'b'); grid on; hold on; plot (ref_par_diez_18,'r');
    title('LEFT HIP SEÑALES DISEÑADAS Sensor POSICIÓN Y PAR MOTOR/ emble 1.645 seg');
    xlabel('sample time 0,001 s'); ylabel('pos.18/ par.18');
    legend('señal posición decimada','señal parM decimada');
```

Figura C. 1 Comandos para obtener la referencia de consigna del control de par del actuador de la cadera izquierda.

```

% AHORA SÓLO SE TRABAJA CON LAS SEÑALES DISEMADAS
% Asignando nombres más cortos a las siguientes variables
a= ref_pos_dies_15; % sensor posición señal disemada
b= LeftLeg(1:200,1); % consigna posición
c= ref_par_dies_15; % sensor par señal disemada

lom=length(b);
rms=zeros(lom,1);
for i=1:200
    error= b - circshift(a,1-1);
    rms(i)=(i/length(error))* (error'*error);
end

[Mi,Pos] = min (rms); % valor del mínimo y posición del mínimo

rms_período(p)=Mi;
tras_pos=circshift(a,Pos); % 106-156=50 veces trasladada

tras_par=circshift(c,Pos); % 106-156=50 veces trasladada

figure (3)
plot (a, 'm'); grid on; hold on; plot (a,'b'); plot (tras_pos,'g'); plot (c,'r'); plot (tras_par,'y');
title('LEFT HIP SEÑALES DISEMADAS Y TRASLADADAS Sensor POSICIÓN Y PAR MOTOR/ ensamble 1.645 seg');
xlabel('sample time 0,001 s'); ylabel('pos.(")/ par(Nm)');
legend('consigna posición','señal posición disemada','señal posición trasladada','señal parM disemada','señal parM trasladada');

end

%plot(rms_período);
%y,ind=min(rms_período);

ini=1645+(ind-1)*3000;
fin=ini+2999;
ref_pos_15 = V25_pos15.data(ini:fin,2); % 3000 últimas muestras señal sensor posición (una zancada)
ref_pos_dies_15= ref_pos_15(1:15:end);% señal ref_pos disemada.
ref_par_15 = V25_pos15.data(ini:fin,3); % 3000 últimas muestras señal sensor posición (una zancada)
ref_par_dies_15= ref_par_15(1:15:end);% señal ref_pos disemada.

lom=length(b);
rms=zeros(lom,1);
for i=1:200
    error= b - circshift(a,i-1);
    rms(i)=(i/length(error))* (error'*error);
end

[Mi,Pos] = min (rms); % valor del mínimo y posición del mínimo

rms_período(p)=Mi;
tras_pos=circshift(a,Pos); % 106-156=50 veces trasladada
tras_par=circshift(c,Pos); % 106-156=50 veces trasladada

figure (4)
plot (b, 'm'); grid on; hold on; plot (a,'b', 'LineWidth',3); plot (tras_pos,'b'); plot (c,'r', 'LineWidth',3); plot (tras_par,'y');
title('LEFT HIP SEÑALES DISEMADAS Y TRASLADADAS Sensor POSICIÓN Y PAR MOTOR/ ensamble 1.645 seg');
xlabel('sample time 0,001 s'); ylabel('pos.(")/ par(Nm)');
legend('consigna posición','señal posición disemada','señal posición trasladada','señal parM disemada','señal parM trasladada');

%%%%%%%%%%%%%
%%%%%%%%%%%%%

```

Figura C. 2 Continuación de los comandos desarrollados para obtener la referencia de consigna del control de par del actuador de la cadera izquierda

ANEXO D

Simulación dinámica de la marcha mediante *OpenSim*. Dinámica inversa

En este documento, se muestra el protocolo a seguir para resolver el problema de la dinámica inversa usando datos experimentales mediante el software de simulación dinámica OpenSim (versión 3.3). En este caso, se parte de la posición en el tiempo de los ángulos articulares de la cadera izquierda para obtener los momentos o pares de dicha articulación durante la marcha normal humana.

D.1 Introducción al entorno de simulación de *OpenSim* 3.3

La herramienta *software OpenSim* permite modelar modelos músculo esqueléticos del cuerpo humano y realizar simulaciones dinámicas incorporando movimientos a estos modelos. También puede realizar análisis cinéticos y cinemáticos, comparativas de distintos patrones de la marcha normal y patológica, y graficar su comportamiento.

Entre otras opciones, el entorno gráfico posibilita poder escoger dentro del modelo músculo esquelético, las coordenadas iniciales de las articulaciones y su orientación, especificar la geometría músculo esquelética y dentro del archivo genérico o modificado del movimiento a estudio, permite predecir las fuerzas que desarrollarían músculos y tendones junto con los momentos articulares para poder realizar su posterior análisis.

D.2 Simulación de la dinámica inversa para la articulación de la cadera izquierda durante la marcha normal

Para determinar la cinética de las articulaciones, se utiliza la dinámica inversa con objeto de estimar los momentos de estas articulaciones durante el movimiento. Es decir, la dinámica es el estudio de las fuerzas y momentos que produce un movimiento.

En este apartado se muestra el protocolo a seguir para resolver el problema de la dinámica inversa en la cadera izquierda durante el movimiento de la marcha normal humana a través del software *OpenSim* versión 3.3.

D.2.1 Protocolo para calcular el momento neto en la articulación de la cadera izquierda

1. Cargar el modelo genérico músculo esquelético con extensión *.osim*

Se carga el modelo genérico *gait2392_simbody.osim* de las extremidades inferiores. Este modelo biomecánico representa a un adulto de 1,80m y unos 75Kg de peso y está formado por unos 92 músculos (46 por cada pierna y 6 en el torso) y 13 segmentos rígidos representados por una línea roja que conecta un punto inicial y otro final (figura D.1). El color rojo indica que no hay activación muscular, por el contrario, cuando hay activaciones musculares estos segmentos se vuelven de color azul.

2. Escalar el modelo biomecánico genérico a la antropometría del sujeto.

Mediante la escalación se altera la antropometría del modelo a la del sujeto. En este TFM, el control adaptativo de par se adapta para un paciente de 60 Kg., por lo que se ajustan las características de este modelo a las deseadas mediante unos parámetros de masa.

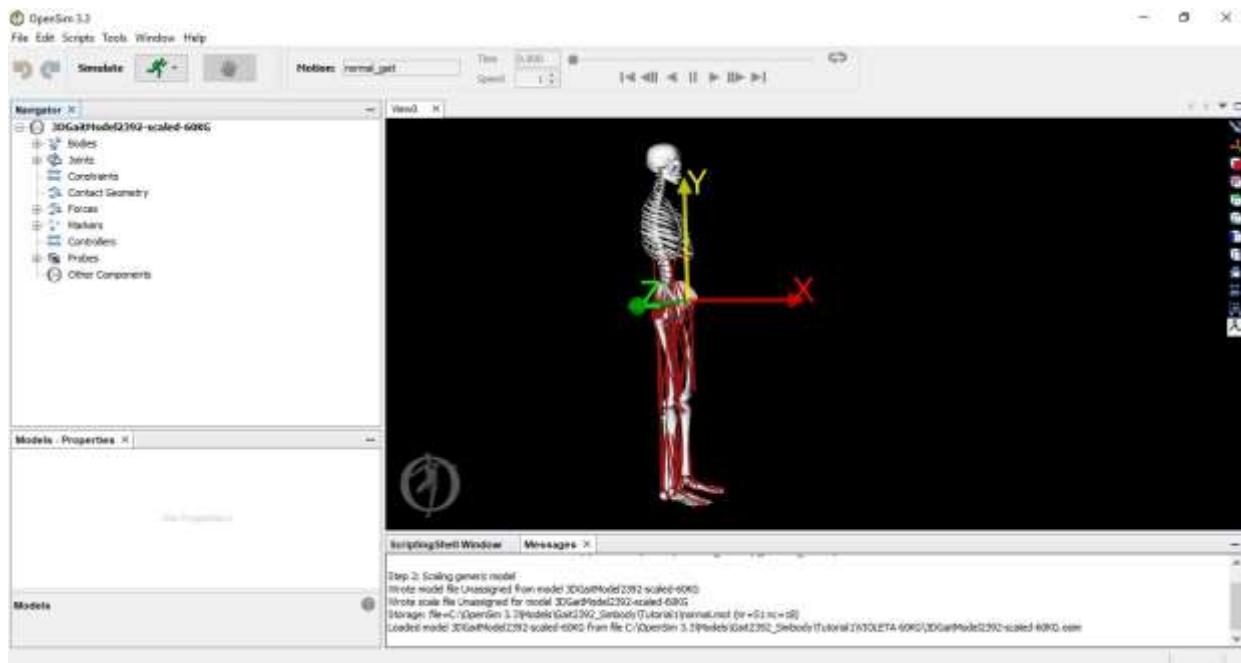


Figura D. 1 Modelo escalado a la antropometría de un sujeto de 60 Kg., partiendo del modelo genérico *gait2392_simbody.osim*

3. Cargar el movimiento con extensión *.mot* en el modelo escalado.

La dinámica inversa, necesita como dato la cinemática del movimiento. Por ello, se modifica el archivo genérico *normal.mot* que contiene los datos reales del movimiento de la marcha humana normal, con los datos que se disponen del movimiento de la flexo-extensión de la articulación de la cadera izquierda.

Posteriormente, se simula en el modelo biomecánico la trayectoria del movimiento de la marcha en el plano sagital.

4. Obtención de la dinámica inversa del movimiento de la marcha norma.

Para obtener los momentos articulares generados por los movimientos de flexión y extensión en el plano sagital durante la acción de caminar, se genera el archivo *inverse_dynamics.sto*. Tras ejecutar dicho archivo, se observa la activación muscular del modelo biomecánico en color azul (Figura D. 2)

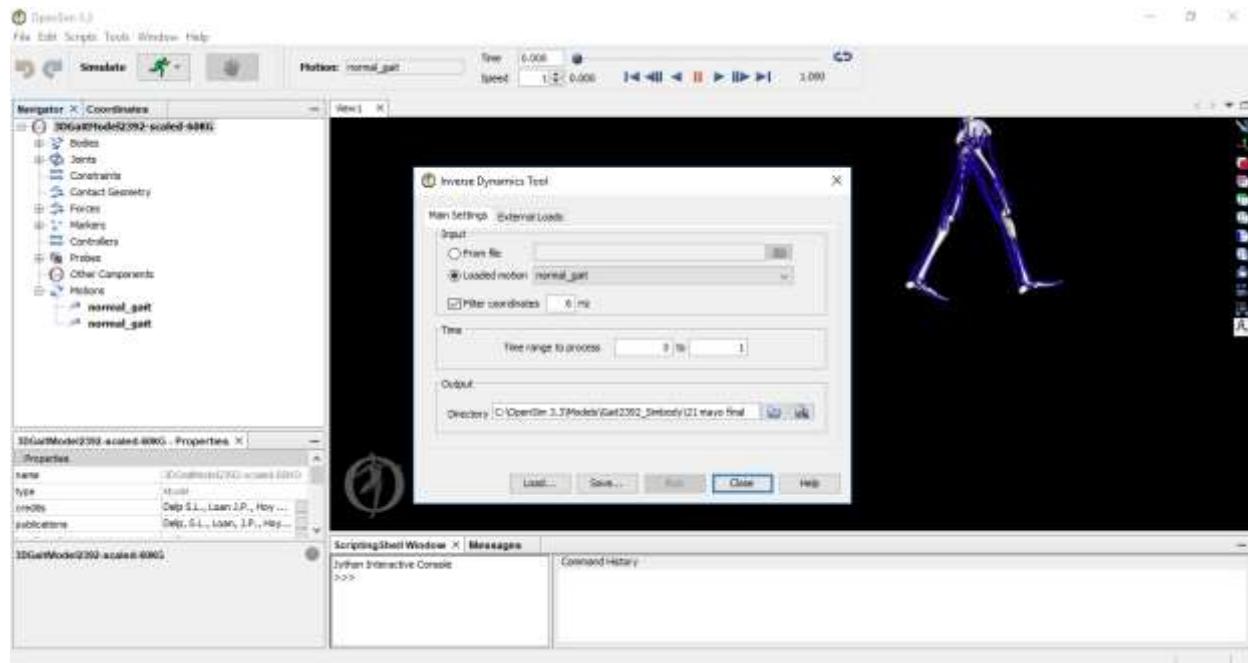


Figura D. 2 Obtención del archivo *inverse_dynamics.sto* para obtener los momentos articulares y la activación muscular del modelo escalado indicada en azul

D.2.2 Resultados de la cinemática y de la cinética de la marcha normal humana en la articulación de la cadera izquierda

A continuación, se muestran los resultados de la trayectoria articular de la cadera izquierda y derecha (figura D.3) y sus respectivos momentos articulares (figura D.4). Posteriormente estos datos son exportados a un fichero de texto para posteriormente poder ser empleados en *Matlab*.

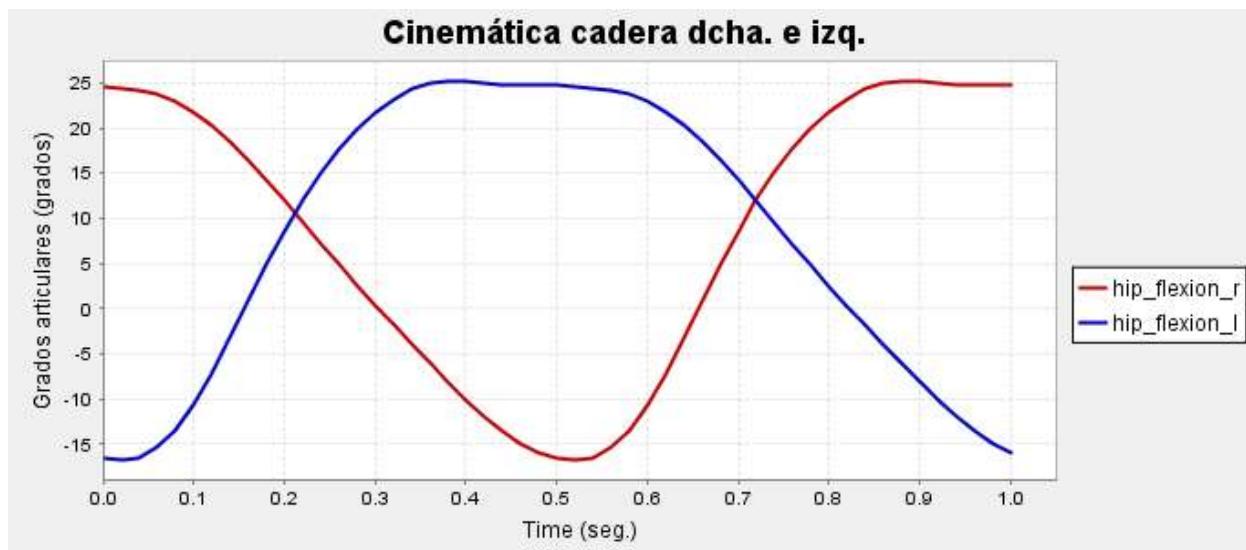


Figura D. 3 Trayectoria articular de la cadera izquierda y derecha durante un paso de la marcha normal humana

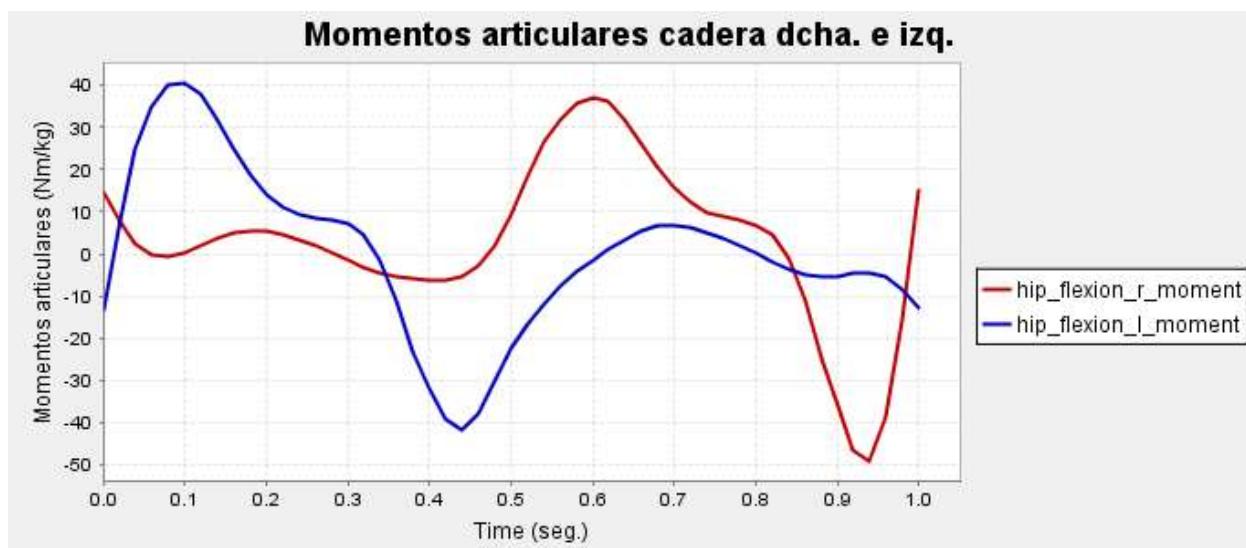


Figura D. 4 Momentos articulares de la cadera izquierda durante un paso de la marcha normal humana

ANEXO E

Simulación del control de par adaptativo

En este anexo se muestran en detalle todas las simulaciones realizadas para diferentes porcentajes de par de asistencia a desarrollar por el exoesqueleto en un paciente de 60kg para el caso de simulación ideal,

E.1 Caso ideal

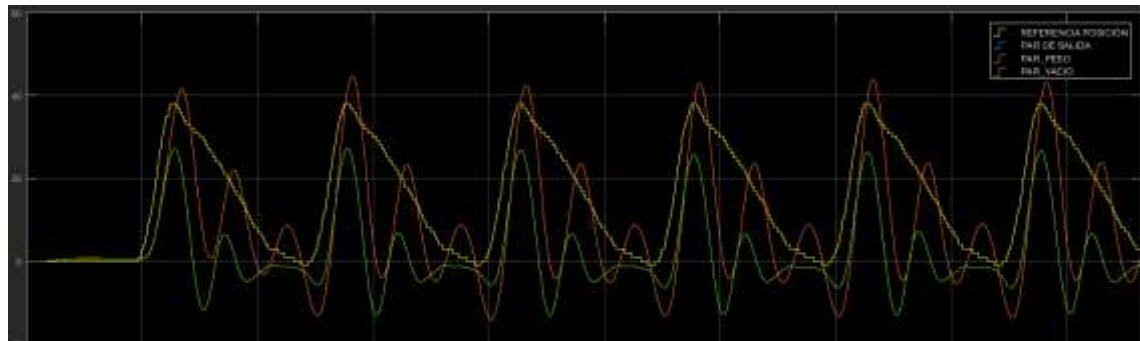
Para el caso ideal: $\tau_{ARTICULAR1} = \tau_{PESO} - \tau_{VACÍO}$, se aplican los siguientes porcentajes de asistencia, dónde:

- K_H , simula la capacidad residual que puede tener el paciente de 60 Kg según el grado de lesión cerebrovascular sufrida. Este nivel de capacidad variará entre $0 < K_H \leq 1$; si $K_H = 1$ la capacidad del paciente será la de un paciente sano.
- K_E , simula la contribución del paciente dentro de su rango de capacidad residual. Este nivel de contribución variará entre $0 < K_E \leq 1$; si $K_E = 1$ el paciente contribuye con el 100% de su par residual para llevar a cabo el movimiento del ejercicio y el robot asistirá en consecuencia.
- K_A , permite variar la asistencia del exoesqueleto. El nivel de asistencia se establece entre $0 < K_A \leq 1$, entonces el exoesqueleto suministrará parte del par para hacer el movimiento y el paciente deberá ejercer el resto de la fuerza para completar el ejercicio; si $K_A = 1$ y $K_H = 1$ será el robot el que haga toda la asistencia del movimiento y la contribución del paciente será nula.

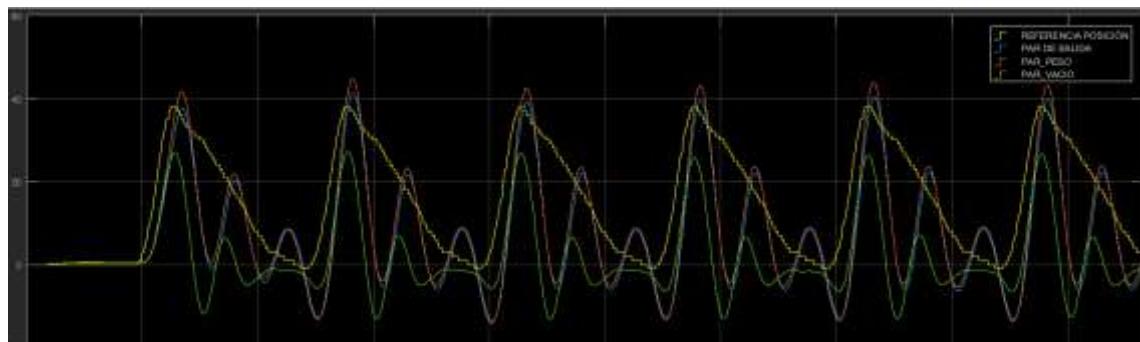
CASO IDEAL									
	KH	KE	KA	KP		KH	KE	KA	KP
Simulación 1	0.8	1	1	3	Simulación 13	0.2	1	1	3
Simulación 2	0.8	1	0.8	3	Simulación 14	0.2	1	0.8	3
Simulación 3	0.8	1	0.5	3	Simulación 15	0.2	1	0.5	3
Simulación 4	0.8	1	0.1	3	Simulación 16	0.2	1	0.1	3
Simulación 5	0.8	0.8	1	3	Simulación 17	0.2	0.8	1	3
Simulación 6	0.8	0.8	0.8	3	Simulación 18	0.2	0.8	0.8	3

Simulación 7	0.8	0.8	0.5	3	Simulación 19	0.2	0.8	0.5	3
Simulación 8	0.8	0.8	0.1	3	Simulación 20	0.2	0.8	0.1	3
Simulación 9	0.8	0.2	1	3	Simulación 21	0.2	0.2	1	3
Simulación 10	0.8	0.2	0.8	3	Simulación 22	0.2	0.2	0.8	3
Simulación 11	0.8	0.2	0.5	3	Simulación 23	0.2	0.2	0.5	3
Simulación 12	0.8	0.2	0.1	3	Simulación 24	0.2	0.2	0.1	3

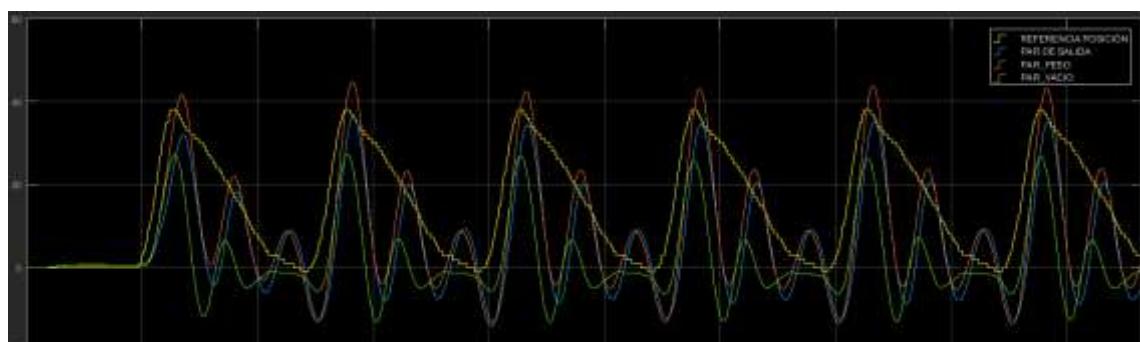
Simulación 1: KH=0.8, KE=1, KA=1, KP=3



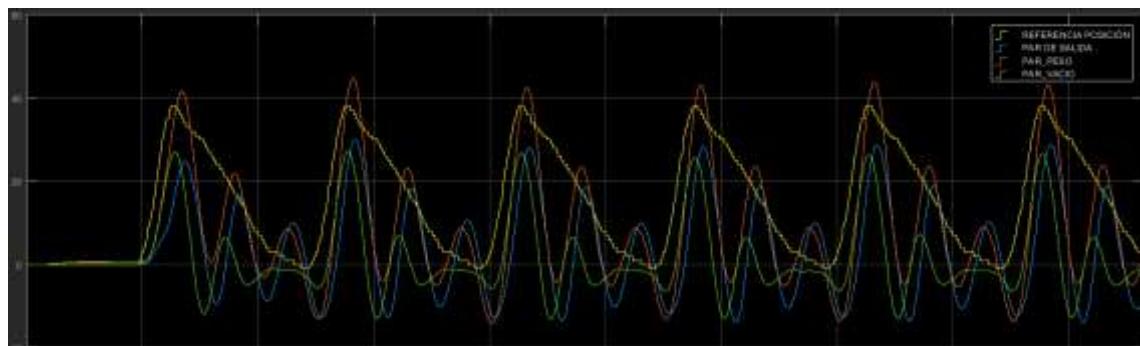
Simulación 2: KH=0.8, KE=1, KA=0.8, KP=3



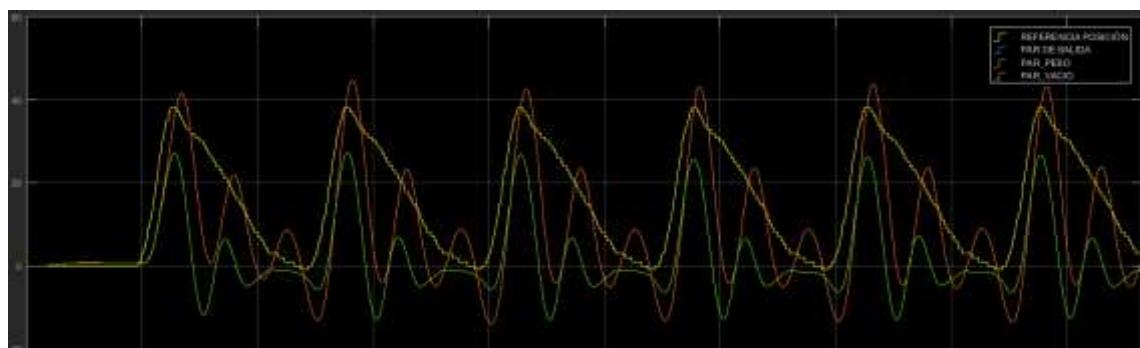
Simulación 3: KH=0.8, KE=1, KA=0.5, KP=3



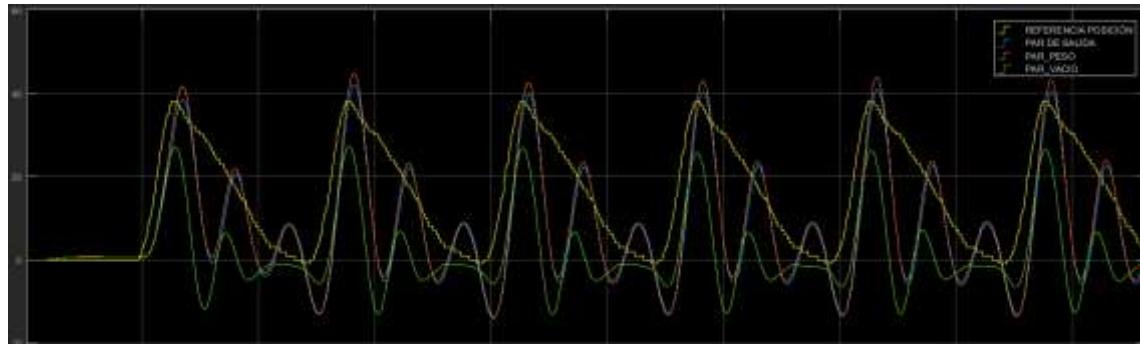
Simulación 4: KH=0.8, KE=1, KA=0.1, KP=3



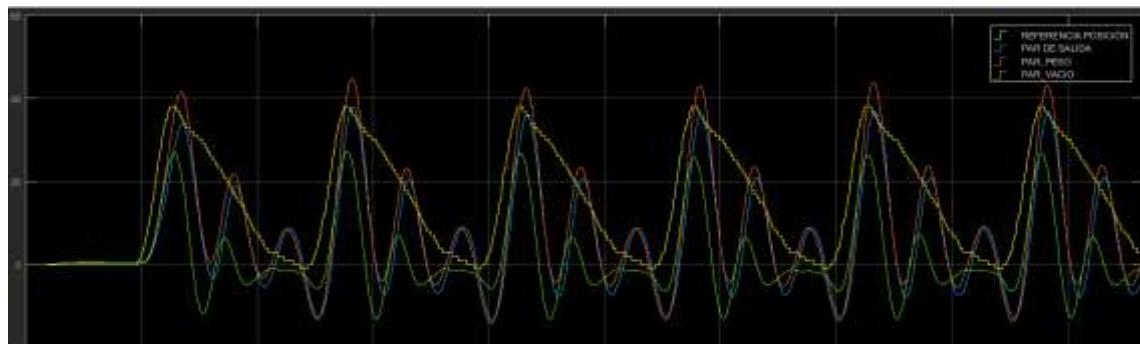
Simulación 5: KH=0.8, KE=0.8, KA=1, KP=3



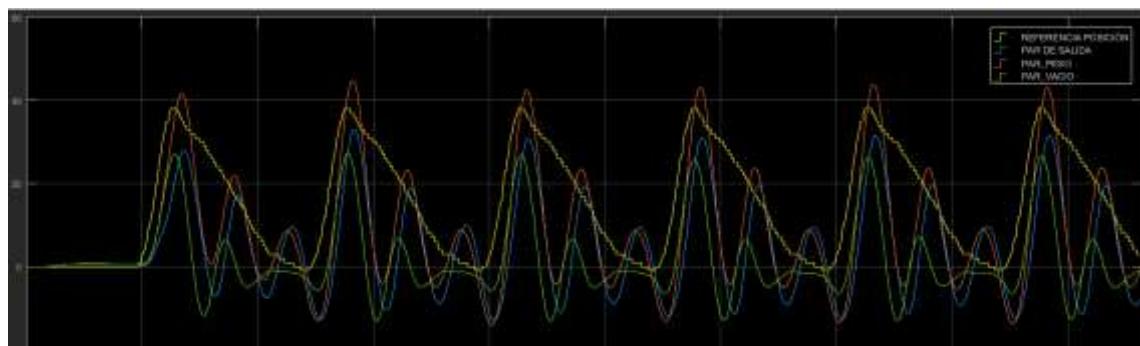
Simulación 6: KH=0.8, KE=0.8, KA=0.8, KP=3



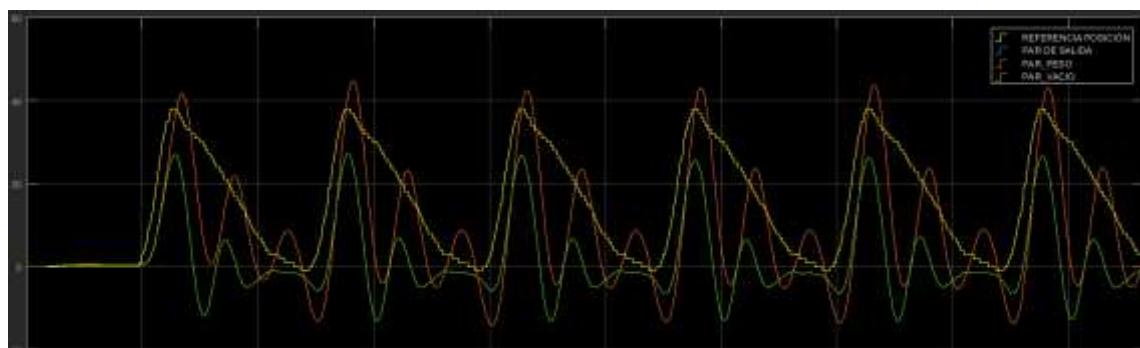
Simulación 7: KH=0.8, KE=0.8, KA=0.5, KP=3



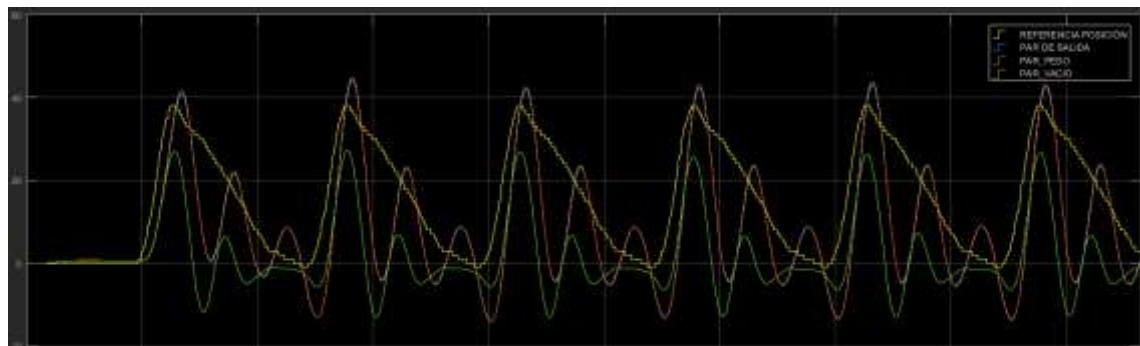
Simulación 8: KH=0.8, KE=0.8 KA=0.1, KP=3



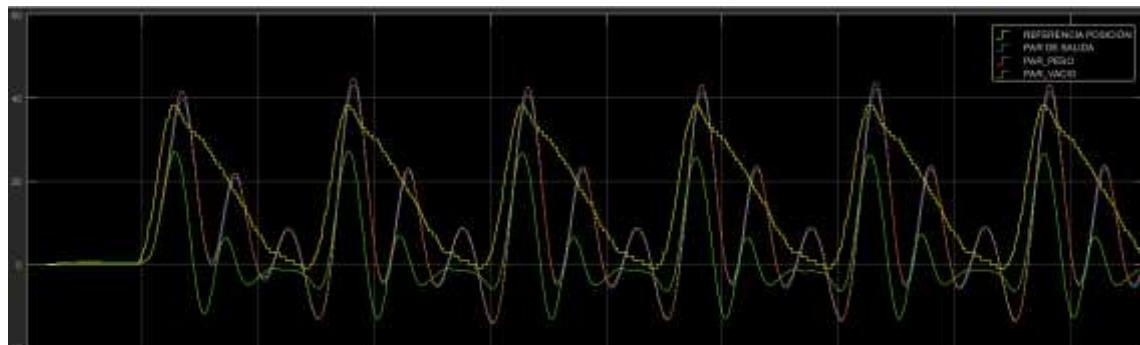
Simulación 9: KH=0.8, KE=0.2, KA=1, KP=3



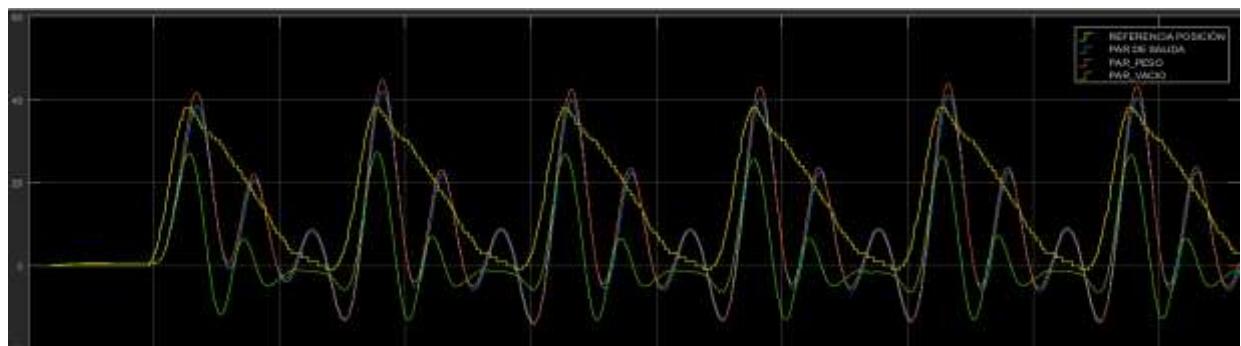
Simulación 10: KH=0.8, KE=0.2, KA=0.8, KP=3



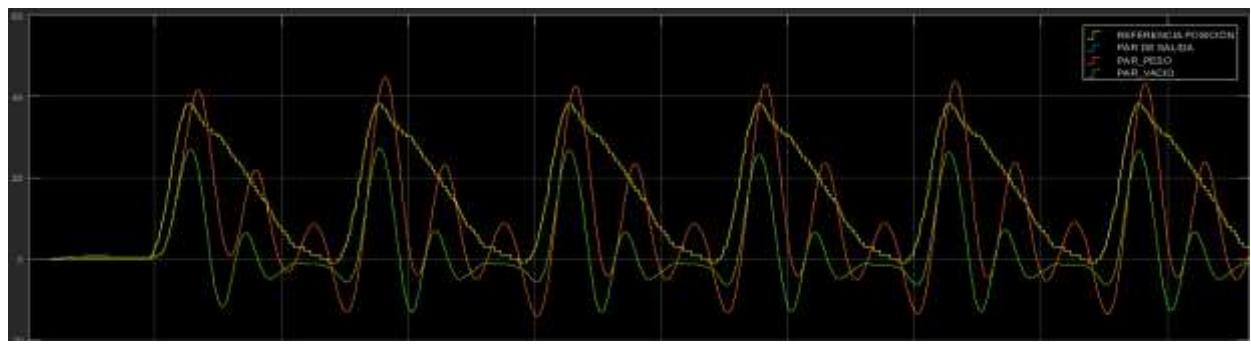
Simulación 11: KH=0.8, KE=0.2, KA=0.5, KP=3



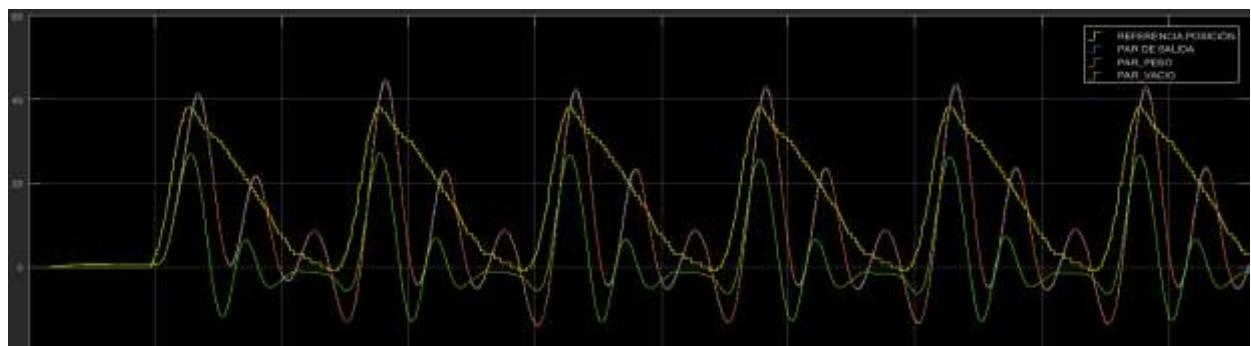
Simulación 12: KH=0.8, KE=0.2 KA=0.1, KP=3



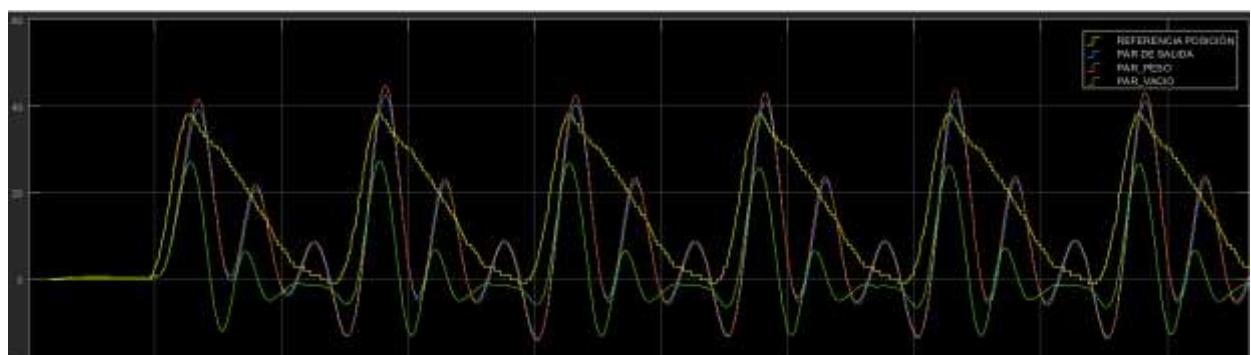
Simulación 13: KH=0.2, KE=1, KA=1, KP=3



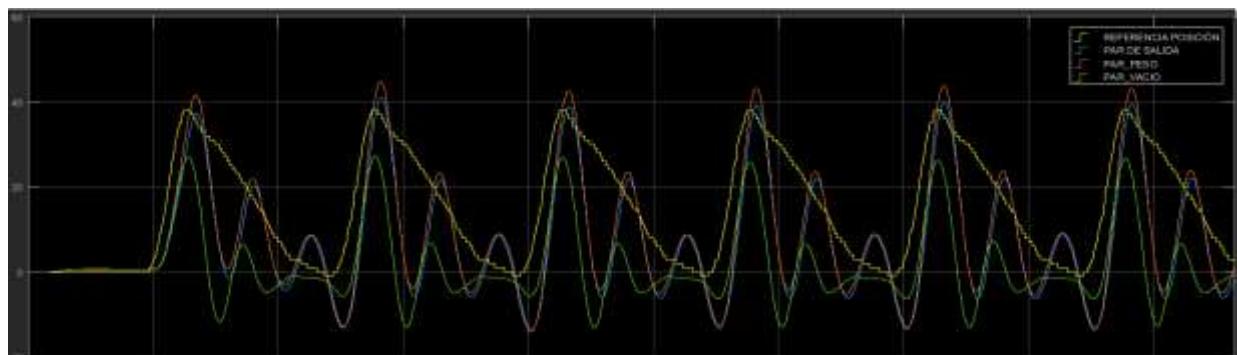
Simulación 14: KH=0.2, KE=1, KA=0.8, KP=3



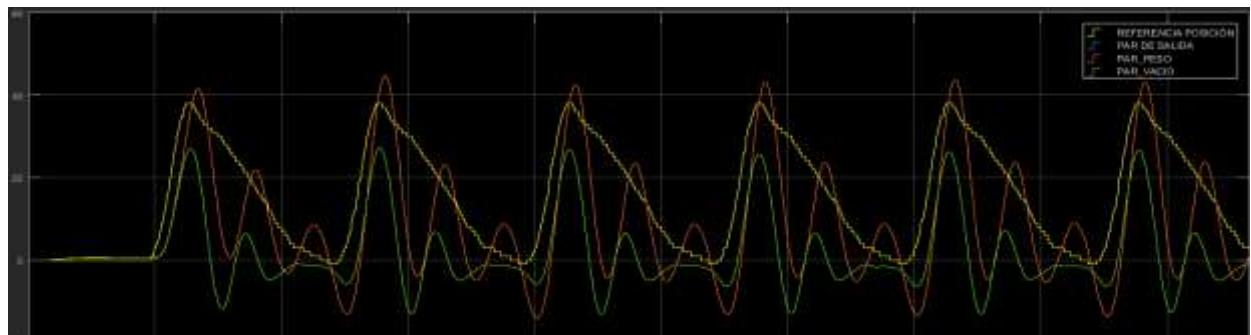
Simulación 15: KH=0.2, KE=1, KA=0.5, KP=3



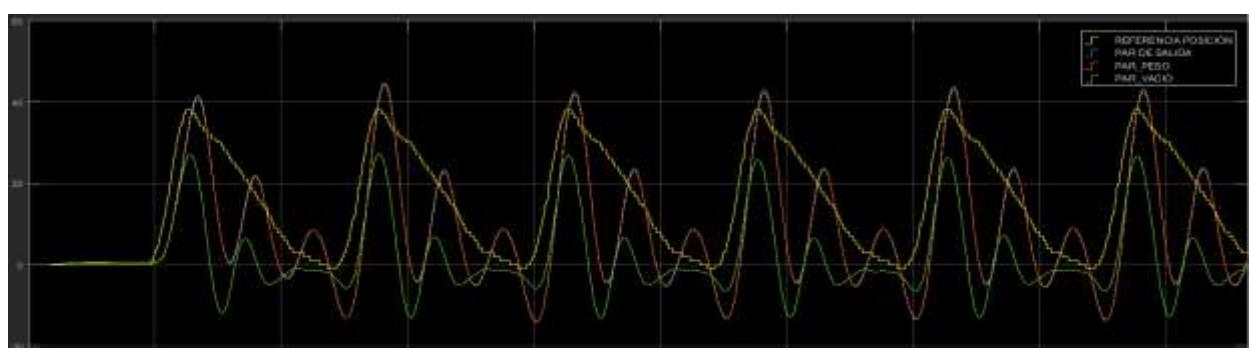
Simulación 16: KH=0.2, KE=1, KA=0.1, KP=3



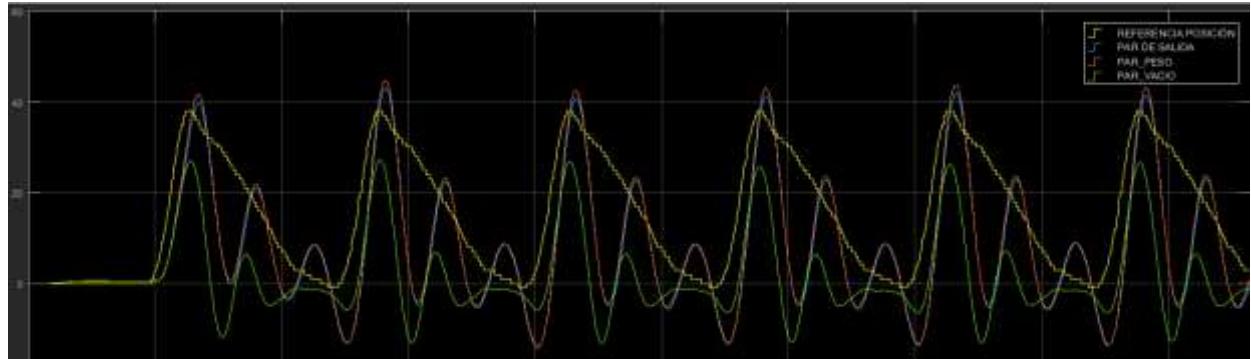
Simulación 17: KH=0.2, KE=0.8, KA=1, KP=3



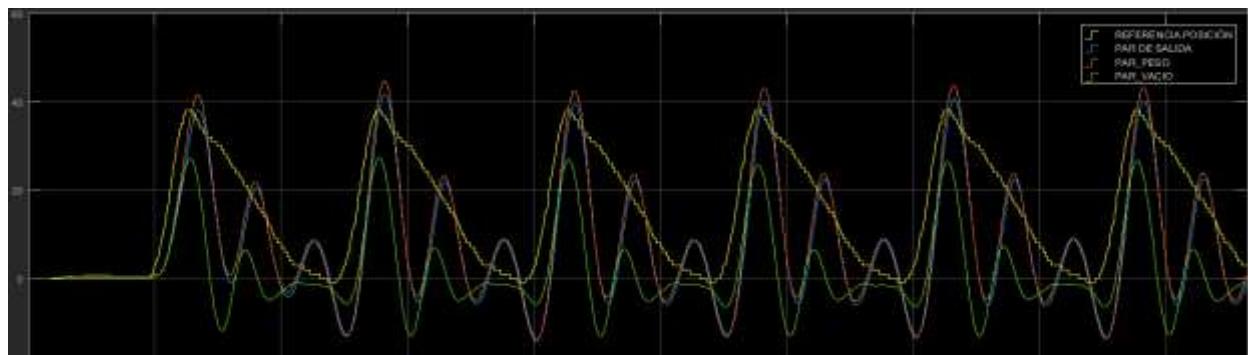
Simulación 18: KH=0.2, KE=0.8, KA=0.8, KP=3



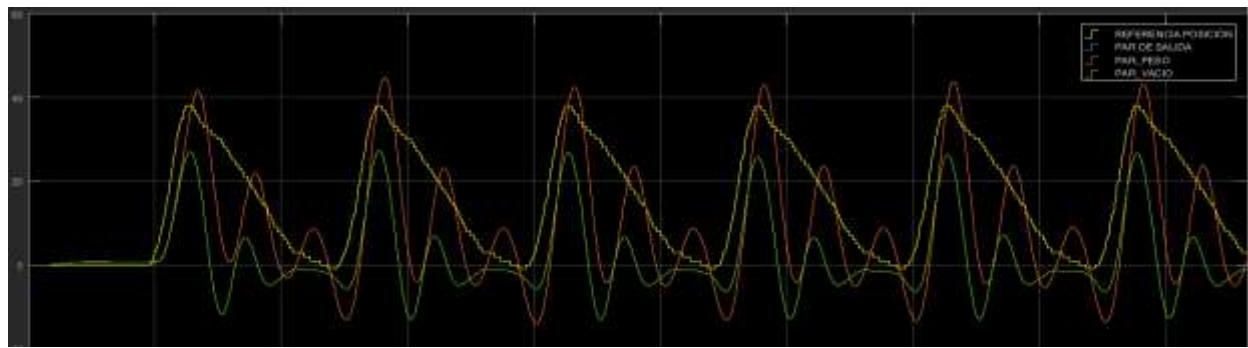
Simulación 19: KH=0.2, KE=0.8, KA=0.5, KP=3



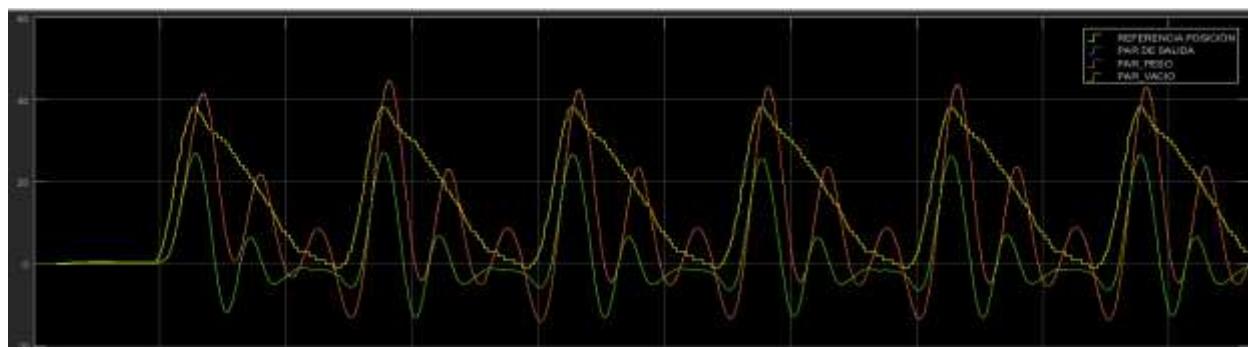
Simulación 20: KH=0.2, KE=0.8 KA=0.1, KP=3



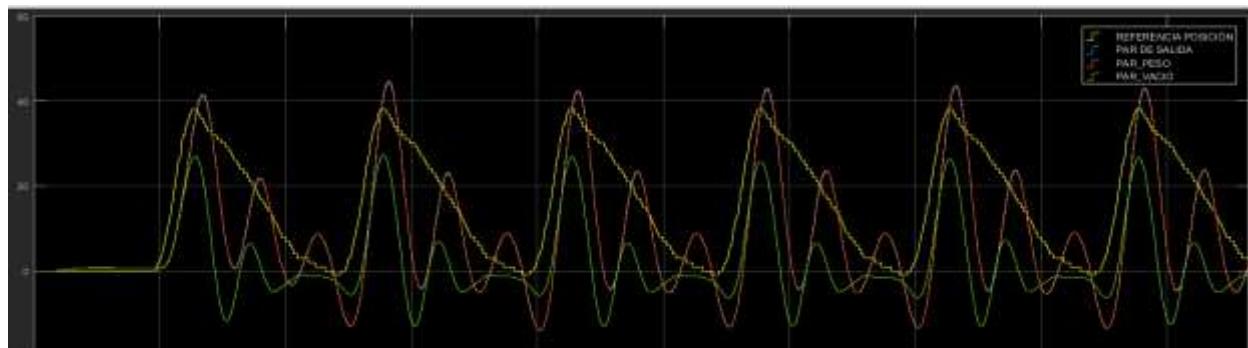
Simulación 21: KH=0.2, KE=0.2, KA=1, KP=3



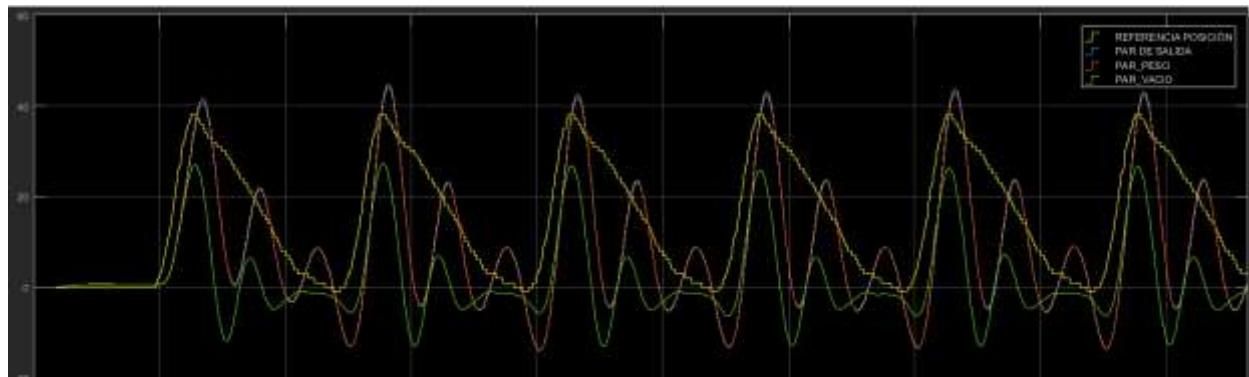
Simulación 22: KH=0.2, KE=0.2, KA=0.8, KP=3



Simulación 23: KH=0.2, KE=0.2, KA=0.5, KP=3



Simulación 24: KH=0.2, KE=0.2 KA=0.1, KP=3



ANEXO F

Proceso de identificación paramétrica

En este anexo, se muestra todo el proceso de obtención de la identificación paramétrica para el sistema del exoesqueleto en vacío (robot ingravido).

F.1 Obtención del proceso de identificación paramétrica

Para realizar el análisis del sistema físico del exoesqueleto sin sujeto (SFEV), se requiere de un modelo matemático que lo represente. A continuación, se muestran los pasos desarrollados en un *script* de *Matlab-Simulink* para realizar el proceso de identificación.

- Adquisición de los datos de entrada-salida del sistema
- Tratamiento previo de los datos registrados
- Estimación del modelo y obtención de sus parámetros
- Evaluación del modelo

F.1.1 Adquisición de los datos de entrada-salida del sistema

Para la obtención de los datos de entrada y salida en un proceso de identificación, lo ideal es excitar al sistema con una señal de entrada cuadrada aperiódica de múltiples frecuencias aleatorias. Con motivo de que el exoesqueleto ejecute dicha trayectoria, lo ideal sería programar en la parte de control de alto nivel mediante *Simulink*, un *script* que permitiera realizar al sistema dicha señal de excitación persistente durante un intervalo de tiempo adecuado y respetando el rango de posiciones de trabajo que soporta la cadera durante la marcha humana. Pero para evitar movimientos bruscos, con los correspondientes pares elevados que se producirían con una señal cuadrada, y contribuiría al deterioro mecánico del equipo, se opta por utilizar como señal de entrada del sistema una señal sinusoidal aperiódica de múltiples frecuencias aleatorias. De igual manera que se hubiera realizado para la señal anterior, su generación se realiza programando una nueva trayectoria en el control de posición del exoesqueleto. Con esta nueva señal de entrada el sistema físico no sufre cambios bruscos de posición ni de par, por lo que la identificación paramétrica del sistema del exoesqueleto sin y con usuario, se realizará con esta señal de entrada.

F.1.2 Tratamiento previo de los datos registrados.

En este punto se procesan y corrigen los datos de las señales de entrada y salida, con motivo de facilitar y mejorar el proceso de identificación. Es decir, se eliminan las perturbaciones de alta frecuencia, los datos erróneos y se tratan los niveles de continua de los datos registrados.

F.1.3 Estimación del modelo y obtención de sus parámetros

Con los datos de entrada y salida preparados, se procede a estimar el modelo con el que se va a trabajar. Se estiman diferentes modelos usando modelos polinomiales de estructura ARX. Posteriormente se escoge el modelo de estructura ARX que mejor porcentaje de semejanza presenta con la salida real, en torno a una estimación de entre un 70% a un 80%

F.1.4 Evaluación del modelo

La evaluación del modelo se lleva a cabo mediante simulación a través de *Matlab-Simulink*, realizando la comparación entre la salida real del par electromagnético del actuador de la cadera y de los modelos ajustados, ingresando las funciones de transferencia discretas obtenidas.

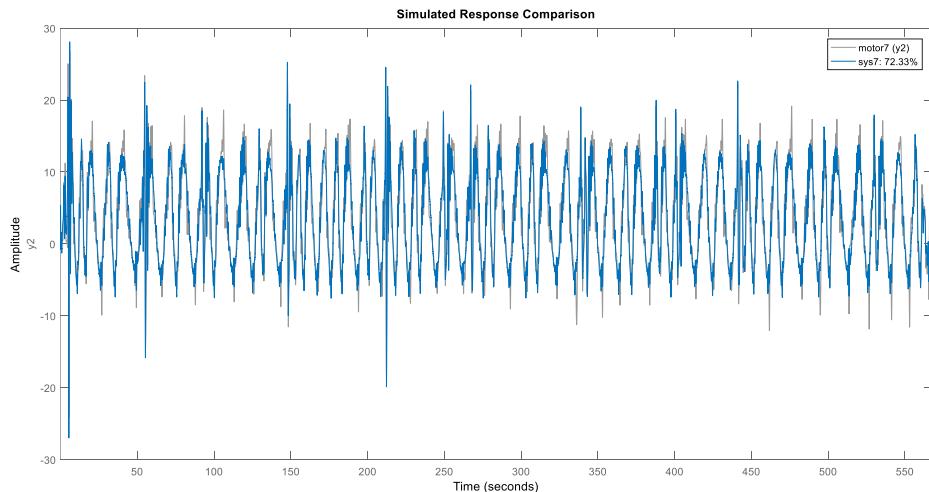
Otra forma realizar la evaluación del modelo es mediante la visualización de polos y ceros que permiten la simplificación del modelo. El análisis de la respuesta en frecuencia del modelo y de la señal real proporciona una información muy rica para la obtención del mejor modelo.

F.2 Análisis del proceso de identificación paramétrica.

En este apartado se desarrollan los apartados anteriores para obtener los modelos identificados de los sistemas SFEP y SFEV.

F2.1 Identificación paramétrica del exoesqueleto H1 mediante dos señales de entrada.

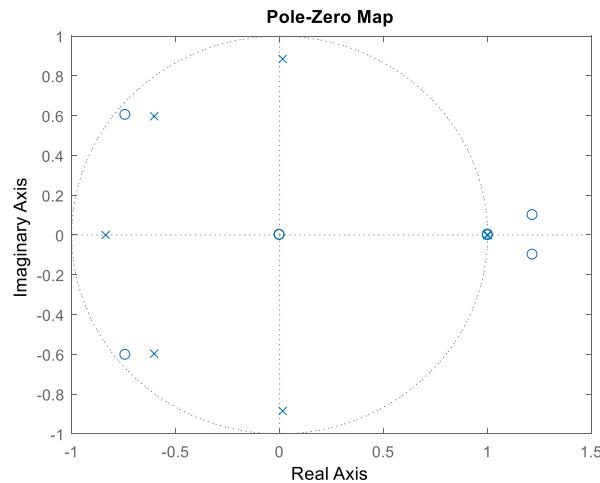
- Similitud del sistema (72,33%)



- Parámetros del sistema completo:

$$\frac{0.04237z^8 - 0.1251z^7 + 0.07198z^6 + 0.06126z^5 + 0.007682z^4 - 0.116z^3 + 0.05775z^2}{z^8 - 0.9815z^7 - 0.5893z^6 - 0.1675z^5 + 0.2225z^4 + 0.4628z^3 + 0.4431z^2 + 0.07962z - 0.4697}$$

- Evaluación del sistema: Mapa de polos y ceros

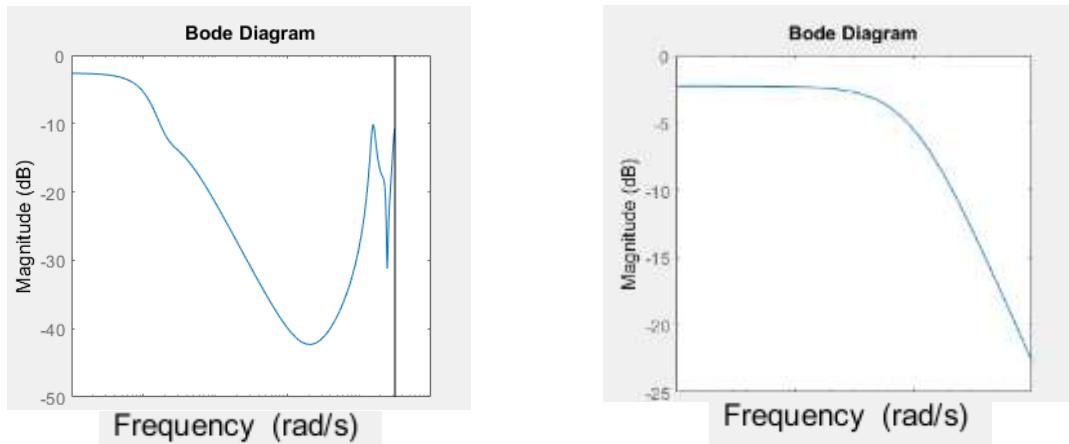


F2.2 Identificación paramétrica simplificada del exoesqueleto H1 mediante dos señales de entrada

- Parámetros del sistema simplificado:

$$\frac{0.0007375}{z - 0.999}$$

- Evaluación del sistema completo (izquierda) y simplificado (derecha) mediante Diagrama de bode:



Bibliografía

- [1] A. Tsukahara, Y. Hasegawa, y Y. Sankai, «Gait support for complete spinal cord injury patient by synchronized leg-swing with HAL», en 2011 IEEE/RSJ International Conference on Intelligent Robots and Systems, 2011, pp. 1737-1742.
- [2] www.reasiste.edu.umh.es/portfolio/material-exoesqueletos-roboticos/
- [3] S. K. Banala, S. K. Agrawal, y J. P. Scholz, «Active Leg Exoskeleton (ALEX) for Gait Rehabilitation of Motor-Impaired Patients», en 2007 IEEE 10th International Conference on Rehabilitation Robotics, 2007, pp. 401-407.
- [4] P. D. Neuhaus, J. H. Noorden, T. J. Craig, T. Torres, J. Kirschbaum, y J. E. Pratt, «Design and evaluation of Mina: A robotic orthosis for paraplegics», en 2011 IEEE International Conference on Rehabilitation Robotics, 2011, pp. 1-8.
- [5] www.neuralrehabilitation.org/projects/HYPER/
- [6] M. Bortole, «Design and control of a robotic exoskeleton for gait rehabilitation», Universidad carlos III de Madrid, Madrid, 2013.
- [7] M. Bartole, A. del Ama, E. Rocon, J. C. Moreno, F. Brunetti y J. L. Pons, «A robotic exoskeleton for overground gait rehabilitation» en 2013 IEEE International Conference on Robotics and Automation (ICRA)
- [8] www.es.mathworks.com/products/simulink.html
- [9] www.opensim.stanford.edu/
- [11] www.harmonicdrive.net/
- [12] «Biomecánica de la marcha humana normal y patológica». coordinador: Jaime Prat ; Autores: J. Javier Sánchez Lacuesta ... [et al.] Valencia : Instituto de Biomecánica, 1999.
- [13] Whittle, M.W. Gait Analysis: An introduction; Butterworth-Heinemann: Oxford, UK. 2014
- [14] G.F. Franklin, J. DAvid Powell, A. Emami-Naeini, "Feedback Control of Dynamic systems", Pearson Education, 2010.
- [18] Ama Espinosa, Antonio José del, «Hybrid walking therapy with fatigue management for spinal cord injured individuals», Universidad carlos III de Madrid, Madrid, 2013

