

Grado en Ingeniería Electrónica Industrial y  
Automática  
2017/2018

*Trabajo Fin de Grado*  
**Calibración de los sistemas  
electromecánicos para habilitar el  
control de estabilidad del robot  
humanoide TEO**

---

Aitor González Martínez de la Casa

Tutor  
Santiago Martínez de la Casa Díaz  
Leganés, Septiembre de 2018



[Incluir en el caso de interés en su publicación en el archivo abierto]  
Esta obra se encuentra sujeta a la licencia Creative Commons **Reconocimiento - No  
Comercial - Sin Obra Derivada**



## Agradecimientos

Agradezco a mis padres todo el apoyo que me han proporcionado durante toda mi carrera, a mi hermano por estar siempre a mi lado y en general a toda mi familia. Espero que se sientan orgullosos.

A mis amigos, por brindarme tantos y tantos momentos de diversión y apoyarme en los momentos más duros.

Agradecer a mis compañeros, a los que considero también como amigos, tanto de grado como de laboratorio, el haberme hecho más amena mi trayectoria en la universidad.

A mi tutor el doctor Santiago Martínez de la Casa por haberme guiado a lo largo de todo el proyecto, al igual que al profesor Juan Miguel García Haro, que ha estado en todo momento cuando he necesitado ayuda.

Y por último, este proyecto está dedicado especialmente a mi abuelo, fallecido tristemente hace un año (se que estará orgulloso esté donde esté), y a mi abuela, por haberme cuidado durante todos estos años de universidad.



## Resumen

En el presente documento se va realizar el estudio de la estabilidad de robots humanoides utilizando los métodos de péndulo invertido y Cart-table para calcular el Punto de Momento Cero (ZMP). Todos los experimentos han sido realizados con el robot humanoide TEO (Task Environment Operator) del grupo de investigación RoboticsLab, en el departamento de Ingeniería de Sistemas y Automática de la Universidad Carlos III de Madrid.

Los datos para realizar dicho estudio se obtenían de los sensores de fuerza-par (F-T) situados en los tobillos y la Unidad de Medida Inercial (IMU), situada en el centro del robot, mediante datos de aceleraciones lineales. Para realizar los experimentos los programas utilizados fueron QtCreator, que es un IDE (entorno de desarrollo integrado) multiplataforma para programar en C++, y Python, que es un lenguaje de programación interpretado cuyo objetivo es simplificar el código del programa.

Dicho proyecto consiste en someter al robot a diferentes inclinaciones, que simulan distintas perturbaciones, para ajustar su posición de equilibrio, utilizando para ello unas mejoras que permiten acrecentar el equilibrio del robot humanoide TEO.



# Índice

<b>Agradecimientos</b>	<b>2</b>
<b>Resumen</b>	<b>4</b>
<b>Índice de figuras</b>	<b>9</b>
<b>Índice de tablas</b>	<b>12</b>
<b>1. Introducción</b>	<b>14</b>
1.1. Motivaciones y origen del proyecto . . . . .	15
1.2. Objetivos . . . . .	15
1.3. Marco regulador . . . . .	16
1.4. Planificación . . . . .	17
<b>2. Estado del arte</b>	<b>19</b>
2.1. Origen de la robótica . . . . .	19
2.1.1. Tipos de robots . . . . .	20
2.2. Evolución de los Robots Humanoides . . . . .	23
2.3. Locomoción bípeda . . . . .	28
2.4. Zero Moment Point (ZMP) . . . . .	32
<b>3. Descripción del sistema</b>	<b>36</b>
3.1. Robot humanoide TEO . . . . .	36
3.2. Sensores fuerza-par . . . . .	38
3.2.1. JR3 50M31A y 85M35A . . . . .	38
3.2.2. Adquisición de datos . . . . .	39
3.2.3. Programa de adquisición de datos . . . . .	40
3.3. Unidad de medida inercial (IMU) . . . . .	42
3.3.1. Xsens MTi-28A53G35 . . . . .	42
<b>4. Modelos simplificados usados en el estudio de la estabilidad</b>	<b>46</b>
4.1. Modelo Péndulo Invertido Lineal . . . . .	46
4.1.1. LIPM aplicado al robot humanoide TEO . . . . .	50
4.1.2. DLIPM: mejora del modelo del péndulo invertido simple . .	51
4.2. Modelo cart-table . . . . .	53
4.3. Equivalencia de modelos . . . . .	55
4.4. Estrategias de control . . . . .	56
<b>5. Estudio de la respuesta de los sensores del robot humanoide TEO y la aplicación de su información al control del equilibrio</b>	<b>59</b>
5.1. Estudio de la respuesta de los sensores F-T . . . . .	59
5.1.1. Descripción de la metodología experimental . . . . .	60
5.1.2. Respuesta del sistema LIPM . . . . .	61
5.1.3. Respuesta del modelo mejorado DLIPM . . . . .	64
5.1.4. Control de ZMP . . . . .	69
5.1.5. Validación de datos experimentales . . . . .	70

5.2. Estudio de la respuesta del sensor inercial IMU . . . . .	72
5.2.1. Descripción de la metodología experimental . . . . .	72
5.2.2. Respuesta del modelo cart-table . . . . .	72
<b>6. Conclusiones y trabajos futuros</b>	<b>76</b>
<b>7. Presupuesto</b>	<b>79</b>
<b>Referencias</b>	<b>81</b>
<b>A. Estructura del robot TEO</b>	<b>86</b>
<b>B. Información sensores JR3</b>	<b>87</b>
<b>C. Información sensor IMU</b>	<b>88</b>
<b>D. Interfaz gráfica del ZMP con sensores fuerza-par</b>	<b>90</b>



# Índice de figuras

1.1.	Robot humanoide [39] . . . . .	14
1.2.	Planificación del proyecto . . . . .	17
2.1.	Robot Industrial ABB [37] . . . . .	20
2.2.	Mars Pathfinder Rover Sojourner [29] . . . . .	21
2.3.	Robots antropomórficos . . . . .	21
2.4.	Robots zoomórficos . . . . .	22
2.5.	Robot HANDLE [35] . . . . .	22
2.6.	Armadura diseñada por Leonardo Da Vinci . . . . .	23
2.7.	Trompetista mecánico . . . . .	24
2.8.	Robot Elektro . . . . .	24
2.9.	Robots Waseda Legged [34] . . . . .	25
2.10.	Waseda's Anthropomorphic Pneumatically-activated pedipulators [34]	25
2.11.	Robots antropomórficos inteligentes WAseada roBOT [34] . . . . .	26
2.12.	De izquierda a derecha, E0 hasta ASIMO . . . . .	26
2.13.	HRP-2 . . . . .	27
2.14.	ATLAS . . . . .	27
2.15.	Centro de Masas del robot humanoide TEO (en color rojo) . . . . .	28
2.16.	Polígono de soporte . . . . .	28
2.17.	ZMP . . . . .	29
2.18.	Evolución del CoP durante el paso . . . . .	29
2.19.	Pivote del Momento Centroidal (CMP) . . . . .	30
2.20.	CoM, Polígono de soporte y ZMP . . . . .	30
2.21.	Representación gráfica del paso o caminata . . . . .	31
2.22.	Representación de fuerzas y momentos que actúan sobre el pie de apoyo . . . . .	32
2.23.	Determinación del ZMP . . . . .	34
2.24.	Relación entre ZMP, FZMP y CoP . . . . .	34
3.1.	Diagrama articulaciones TEO . . . . .	36
3.2.	Esquema comunicaciones TEO . . . . .	37
3.3.	Diagrama comunicaciones . . . . .	37
3.4.	Sensor fuerza-par JR3 85M35A . . . . .	39
3.5.	Tarjeta de adquisición de datos JR3 . . . . .	39
3.6.	Diagrama comunicaciones con PC . . . . .	41
3.7.	Interfaz del programa de adquisición de datos de los sensores JR3 .	41
3.8.	IMU . . . . .	42
3.9.	Cable RS-232 . . . . .	44
4.1.	Planos Anatómicos . . . . .	46
4.2.	Modelo Péndulo Invertido Lineal . . . . .	47
4.3.	Modelo Péndulo Invertido Lineal 3D (3DLIPM) . . . . .	48
4.4.	Modelo LIPM con un sensor F-T entre el tobillo y la suela [26] . .	51
4.5.	Modelo DLIPM . . . . .	52
4.6.	Modelo Cart-Table . . . . .	53
4.7.	Cálculo del ZMP. a) Modelo ficticio, b) Modelo real . . . . .	54
4.8.	Estrategias de balance . . . . .	56
5.1.	Diagrama del desarrollo experimental del nuevo modelo [21] . . . .	59

5.2.	Diagrama de flujo del programa . . . . .	61
5.3.	Plano de desarrollo de los experimentos del robot TEO [21] . . . . .	61
5.4.	Arquitectura de control de posición básica de ZMP para el modelo LIPM . . . . .	62
5.5.	Evolucion ZMP modelo LIPM . . . . .	62
5.6.	Arquitectura de control de ZMP mediante controlador PD . . . . .	63
5.7.	Evolución ZMP para $ZMP_{ref} = 5\text{mm}$ . . . . .	64
5.8.	Comparación experimental régimen permanente $ZMP_{F-T}$ y $ZMP_{ref}$ [21] . . . . .	66
5.9.	Comparativa de la respuesta angular entre LIPM y DLIPM [21] . . . . .	67
5.10.	Respuesta del sistema en función del factor de amortiguamiento . . . . .	68
5.11.	Respuesta angular del modelo DLIPM [21] . . . . .	69
5.12.	Arquitectura de control del modelo DLIPM . . . . .	70
5.13.	Comparación ZMP de los modelos LIPM y DLIPM . . . . .	71
5.14.	Evolucion ZMP modelo DLIPM. a) Respuesta modelo DLIPM ante diferentes simulaciones de perturbaciones; b) Comparación entre modelo LIPM y DLIPM para $ZMP_{ref} = 9\text{cm}$ [21]. . . . .	71
5.15.	Modelo cart-table en el robot TEO . . . . .	73
5.16.	Evolucion ZMP modelo cart-table . . . . .	73
5.17.	Representación ZMP modelo cart-table mejorado . . . . .	74
D.1.	Interfaz gráfica desarrollada en python . . . . .	90



## Índice de tablas

3.1. Modelos y características de los sensores F-T. [JR3 Inc.] . . . . .	38
3.2. Características sensor IMU MTi-28A [Xsens] . . . . .	43
5.1. Batería de experimentos . . . . .	63
5.2. Tabla comparación ZMP entre modelo LIPM y DLIPM . . . . .	70
7.1. Presupuesto . . . . .	79



## 1. Introducción

En los últimos años, la evolución en el campo de la industria ha sido tan grande que ha llegado al punto que se ha hecho necesario el desarrollo de robots para poder realizar las tareas de manera más eficaz, eficiente y segura. Estos robots pueden tener diferente apariencia y función según el ámbito al que se apliquen, desde robots industriales hasta robots para las tareas domésticas. En el campo industrial los robots pueden ser más simples ya que el entorno es principalmente estático, las tareas a desarrollar son repetitivas y automáticas, y apenas hay interacción humana.

Pero no todos los entornos reúnen las condiciones ideales de ser estáticos y repetitivos. Por esta razón surgió la necesidad de crear robots que fueran capaces de trabajar en entornos dinámicos siendo capaces de desarrollar tareas complejas e interactuar con los seres humanos y su entorno. De esa idea de interacción humana surgieron los robots humanoides, con forma humana e inteligencia artificial, que les permite desarrollar un comportamiento y acciones más parecidas a las de los humanos. Pero para llevar a cabo esas acciones es necesario un control ya que la posibilidad de moverse trae consigo el problema de la estabilidad.



**Figura 1.1:** Robot humanoide [39]

Evitar la pérdida de equilibrio del robot durante las fases de posición estática o caminata bípeda es una tarea compleja que está relacionada con el control.

Además, pueden aparecer perturbaciones que dificulten el equilibrio. Dichas perturbaciones pueden surgir de diversas fuentes como pueden ser un empujón o un terreno irregular. El ser humano es capaz de corregirlas inconscientemente moviendo el cuerpo o cualquier articulación. Este movimiento no lo realizan los robots por sí solos por lo que para lograrlo éstos deben imitar nuestros patrones a través de la programación que se les ha proporcionado para estabilizarse por sí solos.

## 1.1. Motivaciones y origen del proyecto

En la actualidad uno de los campos con mayor proyección en la investigación dentro del campo de la robótica es el de crear un robot capaz de desenvolverse por sí solo en el entorno de los seres humanos. De esta idea nacen los robots humanoides, robots con apariencia de ser humano que con inteligencia artificial pretenden mimetizar nuestros comportamientos para desenvolverse en entornos similares al nuestro.

La investigación ha avanzado mucho en los últimos años debido al avance tecnológico en el que nos encontramos.

Pero éstos tienen un problema, la estabilidad al realizar movimientos. Han surgido diversos estudios acerca de aproximarse lo máximo posible a la idea de estabilización del robot, pero dichos estudios poseen diferentes salidas debido a las interferencias de los sensores y el entorno.

De aquí surgió la idea del actual proyecto, en el que se pretende mejorar la respuesta de los sensores, reduciendo el error, ante estímulos externos para así poder aumentar la estabilidad del robot humanoide TEO.

## 1.2. Objetivos

El actual proyecto trata sobre el estudio de los parámetros que sirven para controlar la estabilidad del robot humanoide TEO. El parámetro principal es el Zero Moment Point (ZMP), que es un indicador de estabilidad del robot [19]. Este parámetro se obtiene a partir de la información recibida de los diferentes sensores que posee, tanto los de fuerza-par como el inercial.

Teniendo en cuenta que se van a utilizar estos sensores para medir la estabilidad, y apoyándose sobre unos modelos que están relacionados con el ZMP, los objetivos del presente trabajo son:

- Realizar un estudio del funcionamiento de los sensores, comprobar que las medidas se computan correctamente, obtener los datos y aplicarlos a los modelos.
- Verificar el modelo con los parámetros calculados a partir de la información recibida de los sensores de fuerza-par.
- Una vez que se ha verificado el modelo, desarrollar una mejora que reduzca su error.
- Aplicar dicha mejora a la IMU para que ambas medidas de ZMP coincidan.

### 1.3. Marco regulador

Debido al aumento tanto de la tecnología como de la inteligencia artificial en nuestras vidas, las grandes organizaciones gubernamentales se han visto obligadas a redactar una serie de leyes que regulen la convivencia de los robots y los humanos. Se puede considerar como precursor de las normativas a Isaac Asimov, quien estableció las tres leyes principales que todo robot debe cumplir [1]:

1. Un robot no hará daño a un ser humano, ni permitirá con su inacción que sufra daño.
2. Un robot debe obedecer las órdenes dadas por un ser humano excepto si éstas entran en conflicto con la 1<sup>a</sup> ley.
3. Un robot debe proteger su propia existencia en la medida en la que dicha protección no entre en conflicto con la 1<sup>a</sup> y la 2<sup>a</sup> ley.

Debido a la continua implantación de robots en el sector industrial y su interacción con seres humanos en un mismo espacio de trabajo, se redactaron una serie de leyes, entre las que se encuentran:

1. UNE-EN ISO 10218-1:2012 Robots y dispositivos robóticos. Requisitos de seguridad para robots industriales. Parte 1: Robots. (ISO 10218-1:2012) [27].
2. UNE-EN ISO 10218-2:2011 Robots y dispositivos robóticos. Requisitos de seguridad para robots industriales. Parte 2: Sistemas robot e integración. (ISO 10218-2:2011) [28].

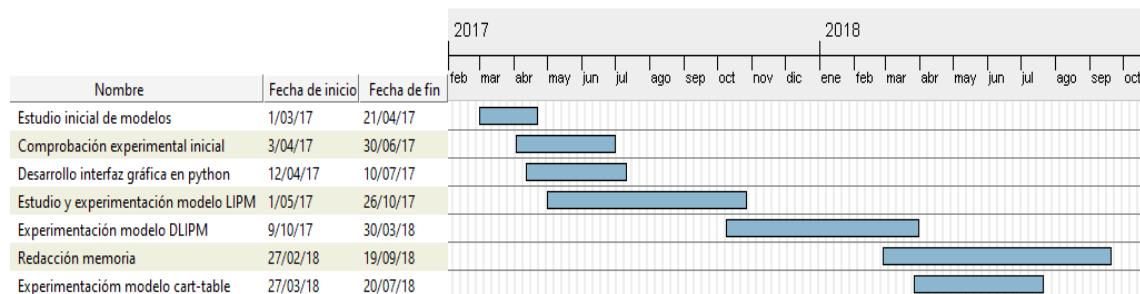
Estas leyes fueron concebidas para proporcionar unas pautas de seguridad en entornos en los que robot y humano comparten el mismo espacio de trabajo.

Fuera del ámbito industrial, la Unión Europea ha propuesto una serie de leyes para, en un futuro, poder regular la convivencia de robots y seres humanos en cualquier entorno [1]:

1. “Los robots deberán tener un interruptor de emergencia”
2. “Los robots no podrán hacer daño a los seres humanos”
3. “No podrán generarse relaciones emocionales con los robots”
4. “Los que sean más grandes deberán tener un seguro obligatorio”
5. “Derechos y obligaciones para los robots”
6. “Tendrán la obligación de pagar impuestos”

## 1.4. Planificación

Para poder llevar a cabo presente proyecto se ha pasado por diferentes fases, representadas en el siguiente diagrama de gantt 1.2:



**Figura 1.2:** Planificación del proyecto

### 1. Estudio inicial de modelos trabajados

En esta fase inicial se realizó un estudio para una toma de contacto con los modelos empleados.

### 2. Comprobación experimental inicial

A continuación se realizaron una serie de experimentos iniciales para poder comprobar el ajuste de ambos modelos al punto de equilibrio deseado.

### 3. Desarrollo de interfaz gráfica en python

Se diseñó una interfaz gráfica en python tanto de las fuerzas de los sensores de los tobillos como de las aceleraciones de la imu, detallados en el anexo D.

### 4. Estudio y experimentación modelo LIPM

Una vez que se escogió el modelo LIPM como modelo a mejorar, se realizó un estudio y una batería de experimentos para ajustar su respuesta.

### 5. Experimentación modelo DLIPM

Una vez desarrollado el nuevo modelo Dynamic LIPM, se llevaron a cabo una serie de experimentos para comprobar su validez.

### 6. Estudio y experimentación modelo cart-table

Por último, se estudió y experimentó con el modelo car-table para mejorar su respuesta ante pequeñas perturbaciones.



## 2. Estado del arte

### 2.1. Origen de la robótica

La palabra “Robótica” fue acuñada por Isaac Asimov para describir la ciencia que se encarga de estudiar la tecnología de los robots. Procede de las palabras checas robota (trabajo forzado) y robotnik (sirviente) [2].

El término “robot”, que procede de la palabra robota, se popularizó gracias a la obra literaria Rossum’s Universal Robot (R.U.R.), escrita por Karel Čapek en 1920 [3].

Según la definición de la Organización Internacional de Estándares (ISO) se considera un robot como un:

“Mecanismo accionado programable en dos o más ejes con un gradio de autonomía, moviéndose dentro de su entorno, para realizar las tareas previstas” [4].

A la hora de hablar del concepto de robótica en general, hay que distinguir entre robótica INDUSTRIAL y robótica de SERVICIO. Esta distinción es fundamental a la hora de su desarrollo, fabricación y comercialización.

Según la norma ISO 8373, la definición de robótica INDUSTRIAL es la siguiente:

“Manipulador multifuncional, controlado automáticamente, reprogramable en tres o más ejes, que puede estar fijo o móvil para uso en aplicaciones de automatización industrial”.

En cuanto a lo que respecta a la robótica de SERVICIO, el Comité ISO TC 184/SC2 lo define como:

“Todo tipo de robot que no es industrial”.

A su vez, la IFR proporciona la siguiente definición:

“Robot que opera de forma parcial o totalmente autónoma al servicio del bienestar de los seres humanos y de equipamientos, excluyendo operaciones manufactureras” [5].

### 2.1.1. Tipos de robots

Resulta complicado establecer una clasificación exacta de los robots debido a la gran cantidad de ellos que existe, por lo que se pueden clasificar de forma más general según su arquitectura [6].

#### 1. Poliarticulados

A este grupo pertenecen los robots cuya característica común es el sedentarismo, aunque pueden ser guiados de forma limitada para poder desplazarse, y cuya estructura se basa en el movimiento de sus elementos terminales para operar en entornos de trabajo fijos, con uno o más sistemas de coordenadas y con un número determinado de grados de libertad. En este grupo se encuentran principalmente los llamados Robots Industriales, como se muestra en la figura (1.1), diseñados para operar en entornos repetitivos y sin interacción humano-robot. También pertenecen los manipuladores y los Robots cartesianos, empleados para abarcar amplias zonas de trabajo, actuar sobre diferentes objetos o reducir el espacio ocupado en el suelo.



**Figura 2.1:** Robot Industrial ABB [37]

#### 2. Móviles

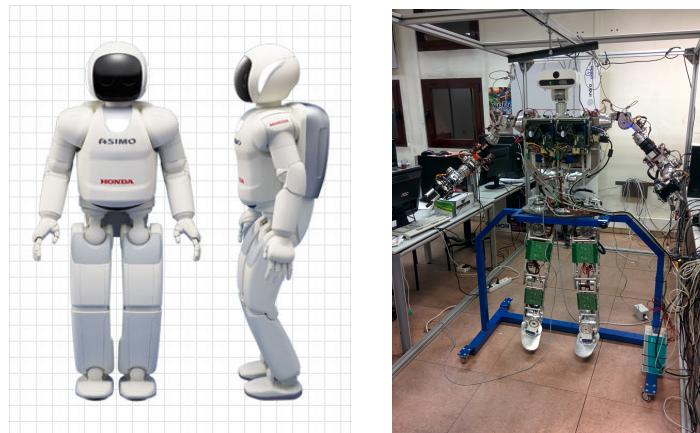
Son robots con capacidad de desplazarse, ya sea mediante ruedas o cualquier plataforma que permita dicha característica, gracias a un sistema motor de tipo rodante. Pueden ser guiados o autónomos (se guían por la información recibida del exterior mediante un sistema de sensores integrado). Estos robots se encargan normalmente de transportar piezas de un punto a otro de la cadena de producción, guiados mediante pistas materializadas a través de la radiación electromagnética de circuitos empotrados en el suelo, o a través de bandas detectadas fotoeléctricamente, otros pueden sortear obstáculos y están dotados de un nivel relativamente elevado de inteligencia.



**Figura 2.2:** Mars Pathfinder Rover Sojourner [29]

### 3. Antropomórficos

Son robots que tratan de imitar parcial o totalmente las acciones y el comportamiento de los seres humanos. Una de las acciones más complejas de éstos es la locomoción bípeda, sobre la que se centra este proyecto y la mayoría de los trabajos en la actualidad, por lo que se concentran en áreas como la investigación y el estudio. El problema que surge es el control dinámico en tiempo real y el mantenimiento simultáneo del equilibrio del robot en dicho proceso.



(a) Robot ASIMO

(b) Robot TEO

**Figura 2.3:** Robots antropomórficos

### 4. Zoomórficos

Son aquellos cuyos sistemas de locomoción imitan a los animales principalmente. Este grupo se podría clasificar en: caminadores y no caminadores.

El grupo de los caminadores es el que más desarrollado está, ya que pueden tener aplicaciones en muchos campos como puede ser el desarrollo

de vehículos todoterreno, controlados o autónomos, capaces de avanzar por terrenos con mucho desnivel y accidentados.

Por el contrario, el grupo de los no caminadores es muy escaso y está poco evolucionado. Se han realizado diversos experimentos en Japón con robots de este tipo constituidos por segmentos cilíndricos biselados acoplados axialmente entre sí, capaces de realizar movimientos relativos de rotación.



(a) Robot caminador ANYmal [36]

(b) Serpiente Robot Modular no caminador

**Figura 2.4:** Robots zoomórficos

## 5. Híbridos

A este grupo pertenecen aquellos Robots que posean en su estructura una combinación de las anteriores ya expuestas, bien sea por conjunción o por yuxtaposición.

Por ejemplo, un dispositivo con ruedas y con brazos, siendo características de los Robots móviles y de los androides al mismo tiempo.



**Figura 2.5:** Robot HANDLE [35]

## 2.2. Evolución de los Robots Humanoides

El desarrollo de los robots humanoides ha sido largo y complejo.

Hubo que esperar al año 1495 para tener los primeros registros del que posiblemente sea el primer robot humanoide, diseñado y construido por Leonardo da Vinci. Se trataba de un robot mecánico, un guerrero vestido con una armadura, que podía realizar movimientos similares a los de los seres humanos como sentarse, mover los brazos y girar la cabeza y el cuello [7].



**Figura 2.6:** Armadura diseñada por Leonardo Da Vinci

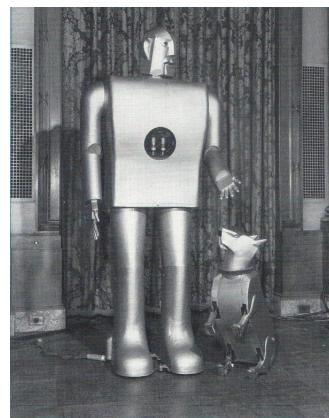
El siglo XVIII fue una época prolífica para el desarrollo de robots autónomos. En 1773, Pierre y Henry Louis inventaron la primera máquina autónoma capaz de escribir.

Pero cuando empieza realmente el desarrollo de los robots humanoides fue a partir del siglo XIX. Uno de los primeros fue el trompetista mecánico, construido por Fridrich Kaufmann en 1810, que contenía un tambor con muescas que usaba para activar algunas válvulas que ayudaban a pasar el aire a través de doce lenguas, lo que producía un tipo de sonido modulado que a su vez pasaba por una trompeta, para hacerla sonar. En 1865 John Brainerd inventó el hombre de vapor, movido por una máquina de vapor que tiraba de unos carros. En 1855 Frank Reade Junior construyó al llamado Hombre Eléctrico, una versión electrificada del hombre de vapor.



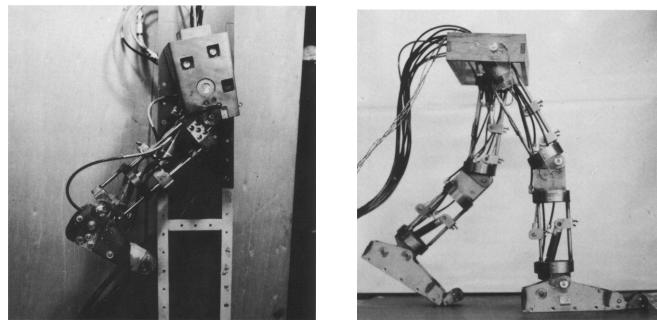
**Figura 2.7:** Trompetista mecánico

A partir del siglo XX el número de robots humanoides fue aumentando y evolucionando. Uno de los primeros fue ELEKTRO, desarrollado por la sociedad Westinghouse en 1938, un robot más cerca del concepto de humanoide que podía caminar controlado por comandos de voz, y hablaba gracias a un tocadiscos de 78 rpm. Más tarde se le unió “Sparko”, un perro robot capaz de sentarse, ladrar y suplicar.



**Figura 2.8:** Robot Elektro

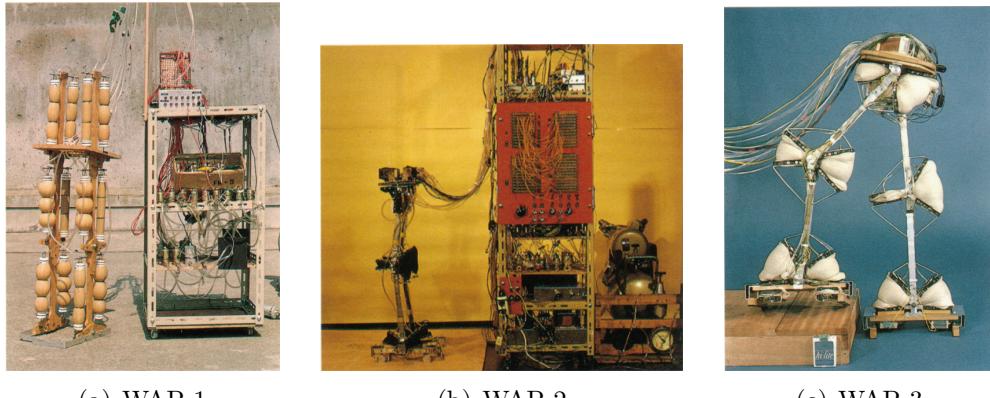
Durante las décadas de los 60's a los 90's aparecieron numerosos robots bípedos. El profesor Kato del equipo robótico de la Universidad de Waseda en Japón desarrolló varias familias de robots bípedos. La primera fue llamada Waseda Legged (WL), que se caracterizaba por ser robots compuestos de sólo dos piernas. En 1969, con la aparición del músculo artificial de goma, desarrollaron la serie WAP (Waseda's Anthropomorphic Pneumatically-activated pedipulators), activados mediante el control de dichos músculos. No fue hasta el modelo WAP-3 que consiguieron que el robot bípedo se desplazara en 3 dimensiones, ya que los dos primeros modelos sólo se movían en un plano.



(a) WL-1

(b) WL-3

**Figura 2.9:** Robots Waseda Legged [34]



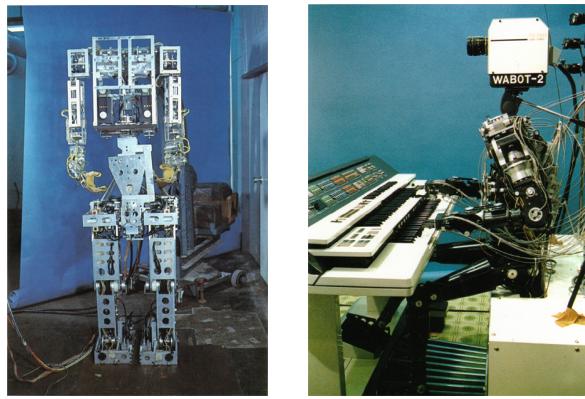
(a) WAP-1

(b) WAP-2

(c) WAP-3

**Figura 2.10:** Waseda's Anthropomorphic Pneumatically-activated pedipulators [34]

Ya en 1970, con los avances en la investigación de robots antropomórficos inteligentes, aparecieron los primeros robots con torso y piernas, pareciéndose cada vez más a los seres humanos. De estos avances se desarrollaron los robots de la familia WABOT (WAseda roBOT), basados en sistemas de control de extremidades, visión y sistemas de conversación. El WABOT-2 incluso era capaz de leer partituras y tocar un teclado.



(a) WABOT-1

(b) WABOT-2

**Figura 2.11:** Robots antropomórficos inteligentes WAseida roBOT [34]

Unos años más tarde, en 1996 se desarrolló la familia de robots WABIAN, que se caracterizaban por utilizar motores eléctricos, logrando la misma velocidad de paso que un humano [8].

En Japón la compañía HONDA llevaba desarrollando robots bípedos desde 1986. Primero fueron del modelo E0 al E6, para pasar a los P1 a P3 y terminando por el robot humanoide más inteligente llamado ASIMO, que fue desarrollado en el 2000.

**Figura 2.12:** De izquierda a derecha, E0 hasta ASIMO

Dos años más tarde, bajo el Ministerio de Económicas, Comercio e Industria de Japón, se desarrolló el Proyecto Humanoide Robot (HRP), con el HRP-2 a la cabeza de la tecnología en esos años [22].



**Figura 2.13:** HRP-2

En la actualidad, los robots humanoides están muy avanzados, pudiendo incluso correr y mantener el equilibrio. Un ejemplo de ello lo tenemos en ATLAS, desarrollado en 2013 por Boston Dynamics, capaz de caminar por terrenos irregulares sin perder el equilibrio, cargar objetos pesados, e incluso recientemente, dar saltos.

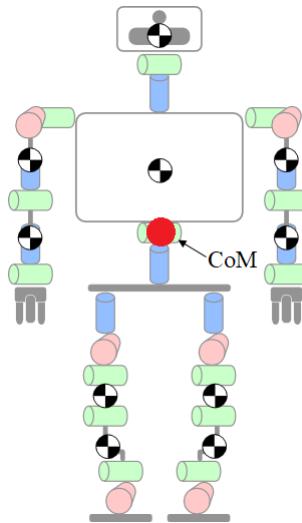


**Figura 2.14:** ATLAS

### 2.3. Locomoción bípeda

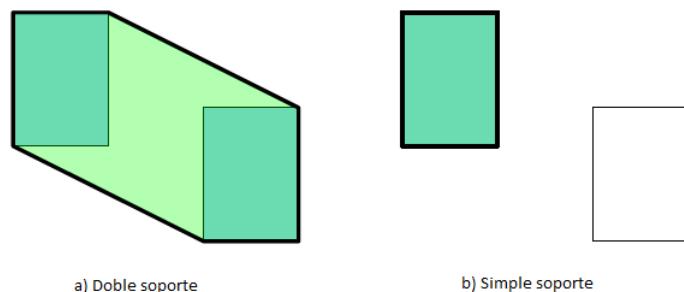
Para hablar de locomoción bípeda o caminata, primero hay que hacer una introducción de una serie de conceptos para poder entender mejor este movimiento.

- Centro de Masas (CoM): es el punto en el que se concentra toda la masa del sistema y sobre el que actúa la resultante de todas las fuerzas que afectan al sistema [9]. Según los modelos CAD del robot TEO, el CoM se encuentra a una altura de 0,893 m.



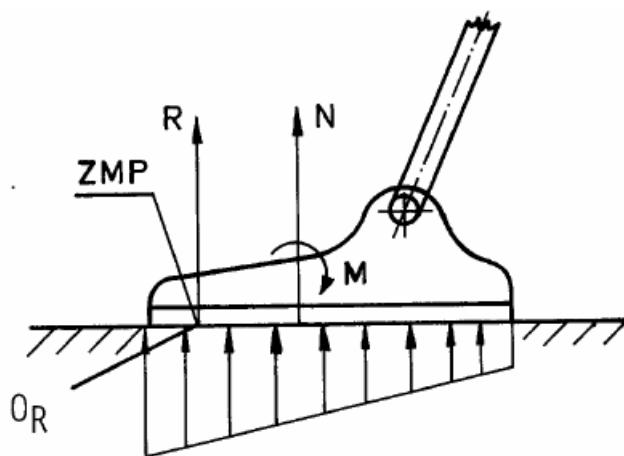
**Figura 2.15:** Centro de Masas del robot humanoide TEO (en color rojo)

- Polígono de soporte: se trata del área de contacto de los puntos de apoyo con el suelo. Puede ser simple, con un pie, o doble, con los dos pies.



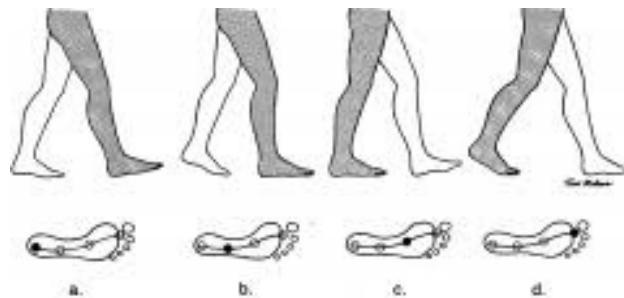
**Figura 2.16:** Polígono de soporte

- Punto de Momento Cero (ZMP): Según Vukobratović [19], es el punto  $R$  en el cual la suma de momentos es igual a 0.



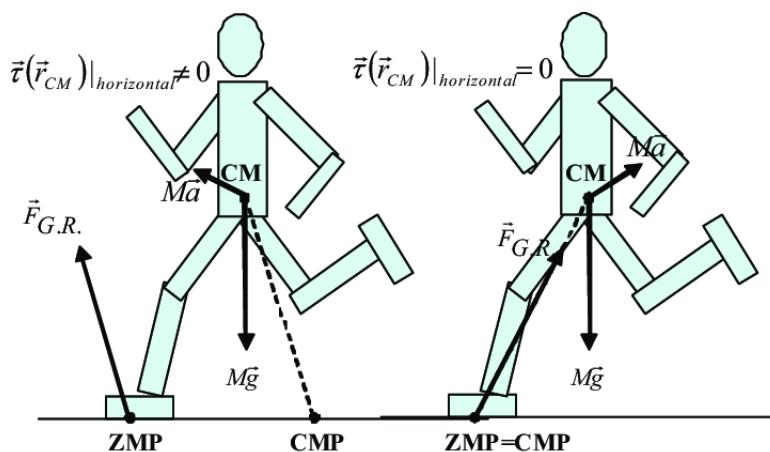
**Figura 2.17:** ZMP

- Centro de Presión (CoP): Punto en el polígono de soporte donde actúa la suma total de las fuerzas de contacto, causando una fuerza pero no un momento [24].



**Figura 2.18:** Evolución del CoP durante el paso

- Pivote del Momento Centroidal (CMP): Punto donde la fuerza de reacción del suelo actúa manteniendo constante la componente horizontal del momento angular de todo el cuerpo. El CMP es igual al ZMP en el caso en que el momento del CoM es cero. Sin embargo, cuando aparece un momento significativo sobre el CoM, aunque por definición el ZMP no puede abandonar el polígono de soporte en el suelo, el CMP puede hacerlo, como se puede observar en la figura 2.19 [25].



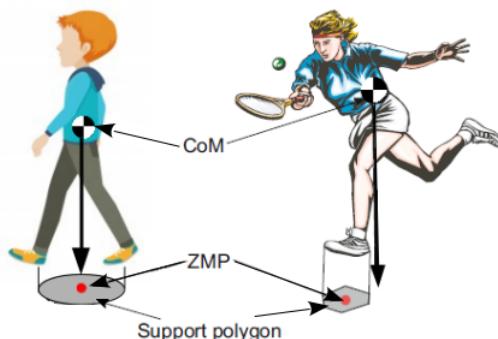
**Figura 2.19:** Pivote del Momento Centroidal (CMP)

### ¿QUÉ ES LOCOMOCIÓN BÍPEDA O CAMINATA?

La locomoción bípeda, o caminata, es la acción de desplazarse de un lugar a otro manteniendo en todo momento un pie en contacto con la superficie. En los seres humanos este movimiento complejo y periódico alterna una fase de estabilidad con una fase de inestabilidad. Esta fase de inestabilidad es provocada por la gravedad, pero gracias a ella se produce el movimiento hacia adelante. Debido a la complejidad de este movimiento, se ha utilizado el criterio de Punto de Momento Cero (ZMP) para simplificar su análisis.

Los robots humanoides han sido desarrollados para imitar los movimientos y comportamientos de los humanos. Pero la caminata resulta un comportamiento muy complejo para éstos.

Existen dos tipos de caminata, llamadas caminata estática y caminata dinámica. En la caminata estática el CoM no abandona nunca el polígono de soporte. Sin embargo, en la caminata dinámica, hay períodos en los que la proyección del CoM se sale fuera del polígono de soporte [10], como se puede observar en la figura 2.20.



**Figura 2.20:** CoM, Polígono de soporte y ZMP

Cuando los seres humanos realizan el movimiento de caminar logran mantener el equilibrio gracias a un control corporal total. Sin embargo los robots humanoides no son capaces de controlar por sí solos su cuerpo, por lo que su control de estabilidad es más complejo. En el actual proyecto se van a explicar dos modelos que desarrollan un controlador para mantener el equilibrio en los robots: el modelo de péndulo invertido y el modelo cart-table. Éstos consisten en calcular ciertos parámetros haciendo uso de los sensores de fuerza-par, en el caso del péndulo invertido, y del sensor inercial, en el caso del cart-table, para lograr mantener el equilibrio en la caminata de los robots.

## FASES DEL PASO

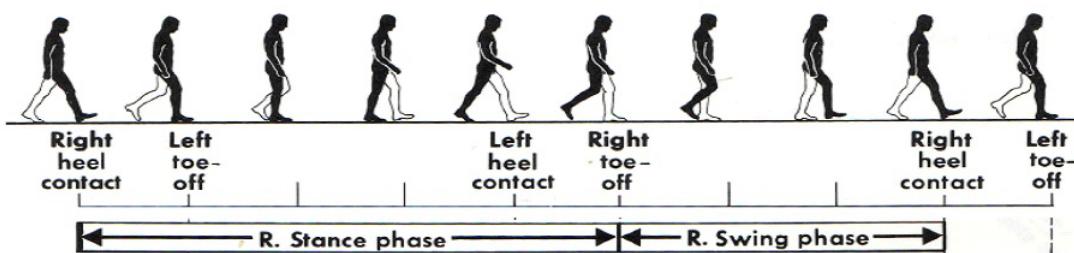
En el paso o caminata bípeda, como se puede observar en la figura 2.21, las fases se distinguen por:

- **FASE 1.**Fase de apoyo

En esta fase la pierna y el pie que están completamente apoyados en el suelo, soportan todo el peso del bípedo, mientras la pierna de balanceo está realizando el paso.

- **FASE 2.**Fase de balanceo

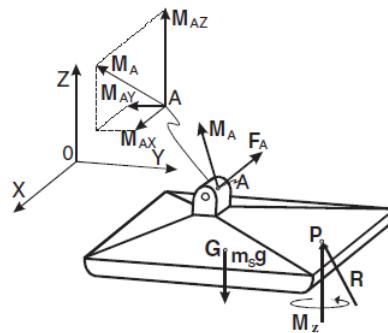
En esta fase la pierna y el pie que estaban antes apoyados ahora están realizando un paso (avanzando por el aire), hasta volver al apoyo de nuevo.



**Figura 2.21:** Representación gráfica del paso o caminata

## 2.4. Zero Moment Point (ZMP)

Vukobratović [16] fue uno de los primeros en introducir el concepto más importante y principal a la hora de mantener el equilibrio de los robots humanoides bípedos. Decía que durante el paso, aparte de la realización del movimiento relativo entre juntas del mecanismo, la tarea más importante es preservar el equilibrio dinámico. Para ello toda el área del pie (incluyendo sus bordes) debe estar en contacto con el suelo.



**Figura 2.22:** Representación de fuerzas y momentos que actúan sobre el pie de apoyo

Si el mecanismo está en equilibrio quiere decir que en el punto P de la suela, donde actúan las fuerzas de reacción del suelo,

$$\begin{aligned} M_x &= 0 \\ M_y &= 0 \end{aligned} \tag{2.1}$$

Gracias a ello el punto P pasó a denominarse Punto de Momento Cero (ZMP), pudiéndose reducir dichas fuerzas a una sola llamada  $R$  y a la componente z del momento  $M_z$  sobre dicho punto.

Para calcular dicho punto y asegurarse de que el mecanismo está en equilibrio dinámico, teniendo en cuenta la dinámica del mecanismo, el requisito indispensable es que el pie esté completamente apoyado en el suelo. Para ello hay que partir de las ecuaciones del equilibrio estático para el pie de apoyo:

$$R + F_A + m_s g = 0 \tag{2.2}$$

$$\overrightarrow{OP} \times \overrightarrow{R} + \overrightarrow{OG} \times m_s g + M_A + M_Z + \overrightarrow{OA} \times F_A = 0 \tag{2.3}$$

donde  $\overrightarrow{OP}$ ,  $\overrightarrow{OG}$  y  $\overrightarrow{OA}$  son los vectores desde el origen del sistema de coordenadas  $O_{xyz}$  al punto P donde actúan las fuerzas de reacción del suelo, el centro de masas del pie (G), la articulación del tobillo (A), y la masa del pie  $m_s$ .

Dando su proyección en el plano horizontal se obtiene:

$$(\overrightarrow{OP} \times \vec{R})^H + \overrightarrow{OG} \times m_s g + (M_A)^H + (\overrightarrow{OA} \times F_A)^H = 0 \quad (2.4)$$

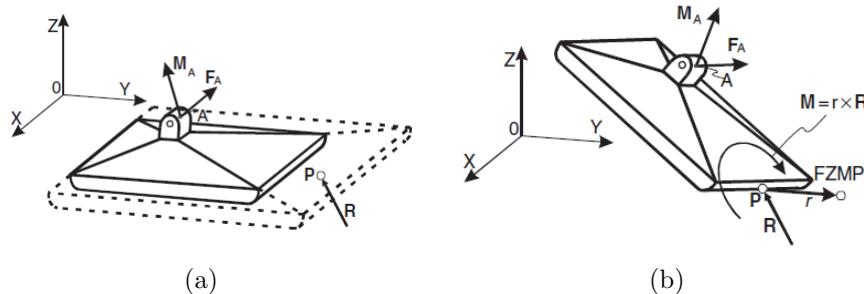
Esta ecuación permite calcular la posición del punto de actuación de la fuerza de reacción del suelo (P) para la dinámica general del mecanismo. Sin embargo esto no sirve para saber si está en equilibrio dinámico cuando el sistema está en movimiento. Para ello hay que tener en cuenta el punto P calculado y el polígono de soporte. Como el punto P calculado con ec (2.4) nunca puede existir fuera del polígono de soporte, ya que si lo hiciera la fuerza total R no actuaría sobre el sistema, sólo si éste está dentro del polígono de soporte el sistema estará en equilibrio dinámico.

Si suponemos que el punto P obtenido de  $M_x = M_y = 0$  está fuera del polígono de soporte, consideraríamos dicho punto como ZMP ficticio (**FZMP**).

Ésto se puede explicar en el caso en el que el ZMP se está acercando al borde del polígono de soporte (tanto en simple soporte como en doble soporte). Si no hubiera más momentos adicionales, el sistema estaría en equilibrio (ZMP), pero si hay momentos adicionales el mecanismo comenzaría a girar y se colapsaría, ya que el punto de actuación de la fuerza de reacción del suelo estaría al borde del pie. Dicho punto ya no sería el ZMP.

Cuando el mecanismo está en movimiento, una manera de determinar el ZMP es midiendo las fuerzas que actúan cuando el pie entre en contacto con el suelo mediante sensores ubicados en la suela del mecanismo (todos los sensores deben estar en contacto con el suelo, ya que si no se cumple el mecanismo giraría en el borde del pie y volcaría), calculando dicho punto mediante el CoP, que se ha explicado anteriormente. Otra manera sería mediante un modelo dinámico. A partir de las fuerza y el momento en la articulación del tobillo ( $F_A$  y  $M_A$ ) el procedimiento para calcular ZMP consta de dos pasos:

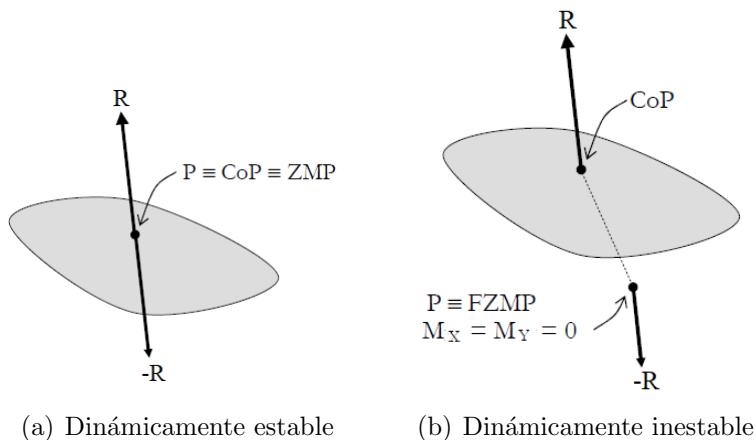
1. Calcular  $\overrightarrow{OP}$  de la ec (2.4). La posición obtenida de P sería el ZMP. Aquí no sabemos todavía si el punto P se encuentra dentro o fuera del polígono de soporte porque no se tiene en cuenta el tamaño real de dicho polígono (figura 2.23 a).
2. Una vez calculada la posición del ZMP, se compara con el tamaño real del polígono de soporte. Si está fuera significa que el punto de actuación P de las fuerzas de reacción del suelo se encuentra en el borde del polígono de soporte y se producirá una rotación del mecanismo debido al momento de desequilibrio, cuya intensidad depende de la distancia entre el borde del polígono de soporte y la posición del ZMP calculada, es decir, a la posición del FZMP. Si se encuentra dentro del polígono, el mecanismo estaría en equilibrio (figura 2.23 b).



**Figura 2.23:** Determinación del ZMP

### RELACIÓN ENTRE ZMP, FZMP Y COP

Únicamente si la fuerza resultante  $R$  está equilibrada con todas las fuerzas activas del sistema bípedo, el CoP y el ZMP coinciden, entonces podríamos decir que se encuentra en equilibrio dinámico,  $P \equiv CoP \equiv ZMP$ , como se muestra en la figura 2.24a [24].



**Figura 2.24:** Relación entre ZMP, FZMP y CoP

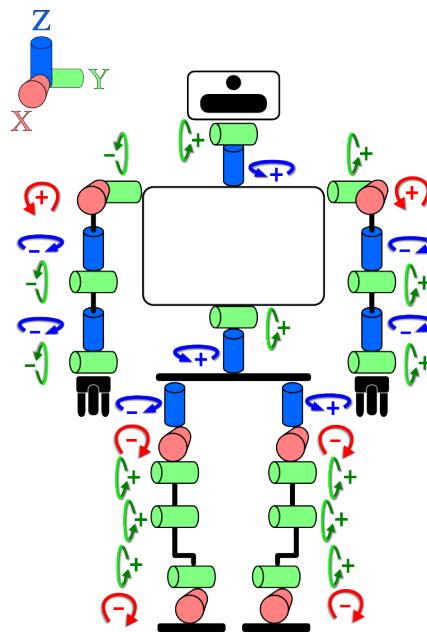
Sin embargo, si nos fijamos en la figura 2.24b vemos que el punto de ataque de las fuerzas cae fuera del soporte de apoyo, ZMP y CoP no coinciden, por lo que ahora  $P \equiv FZMP$ .



### 3. Descripción del sistema

#### 3.1. Robot humanoide TEO

El desarrollo del actual proyecto se produjo sobre el robot humanoide de tamaño completo TEO (Task Environment Operator) del grupo de investigación RoboticsLab, en el departamento de Ingeniería de Sistemas y Automática de la Universidad Carlos III de Madrid. Se trata de una versión mejorada de su predecesor Rh-1. Posee 28 grados de libertad y pesa alrededor de 63 kg.



**Figura 3.1:** Diagrama articulaciones TEO

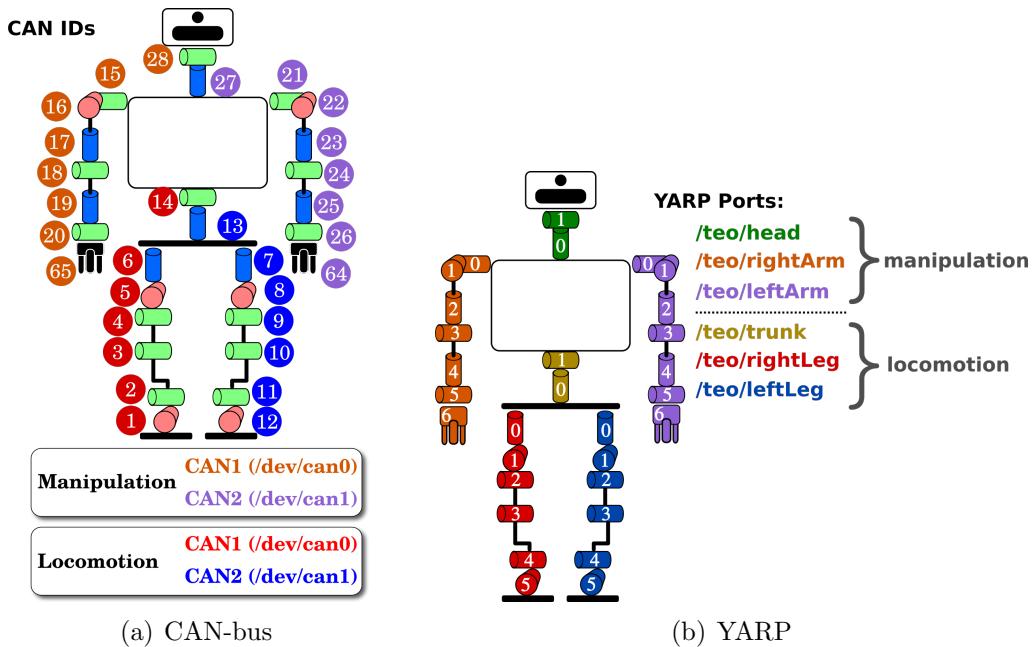
Está compuesto por varios motores paso a paso que se encargan de mover las articulaciones, 4 sensores de fuerza-par (2 en las muñecas y otros 2 en los tobillos) y un sensor inercial (IMU) que se encargan de recoger toda la información proveniente del entorno, un sistema de visión, realizado a través de una cámara ASUS, y 4 microprocesadores, que se encargan de controlar las principales funciones.

Estos microprocesadores son: visión artificial, que se encarga de computar las imágenes que recibe el robot a través de la cámara; locomotion, encargado de controlar las piernas y la posición del torso, y de obtener la información de los sensores de fuerza-par para mantener el robot en equilibrio en todo momento y mantenga una posición erguida, se encuentre en posición estática o caminando; manipulation, encargado de manejar los brazos y la cabeza; y el procesador principal, que gestiona los anteriores microprocesadores.

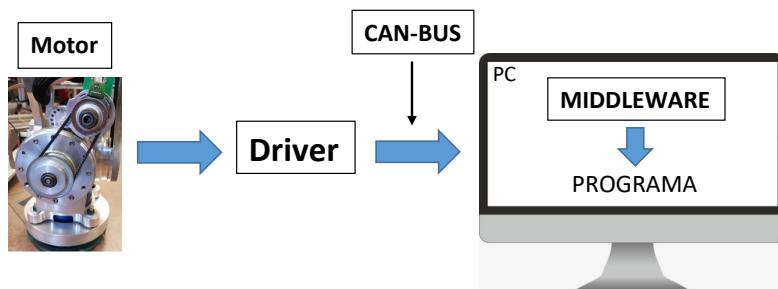
Para la adquisición de datos, como se ha comentado anteriormente, el robot posee

una IMU localizada en el tronco, capaz de medir aceleraciones, campos magnéticos y ángulos, y 4 sensores de fuerza-par situados en muñecas y tobillos, capaces de medir fuerzas y torques.

En cuanto a la comunicación se usa un protocolo CAN-bus<sup>1</sup> y el software para controlar dichas comunicaciones entre CAN-bus y el ordenador se denomina YARP<sup>2</sup>.



**Figura 3.2:** Esquema comunicaciones TEO



**Figura 3.3:** Diagrama comunicaciones

<sup>1</sup>Bus de comunicaciones industriales físicos

<sup>2</sup>YARP (Yet Another Robot Platform) es un middleware (software que ayuda a una aplicación a interactuar o comunicarse con otras aplicaciones, paquetes de programas, hardware, redes, y/o sistemas operativos) escrito en C ++ para interconectar sensores, procesadores y actuadores en robots.

### 3.2. Sensores fuerza-par

Los sensores fuerza-par son dispositivos mecánicos basados en extensómetros dispuestos de tal manera que permiten obtener medidas de fuerzas y momentos en 3 dimensiones.

#### 3.2.1. JR3 50M31A y 85M35A

Los sensores de fuerza-par montados en TEO son sensores proporcionados por la compañía JR3 Inc. y cuyos modelos son 50M31A para las muñecas y 85M35A para los tobillos. Según el fabricante los dos primeros números indican el diámetro de los sensores, seguidos del número de serie, para acabar con dos dígitos que indican su espesor.

Al tratarse de diferentes modelos, éstos poseen diferentes fondos de escala, como se aprecia en la tabla 3.1.

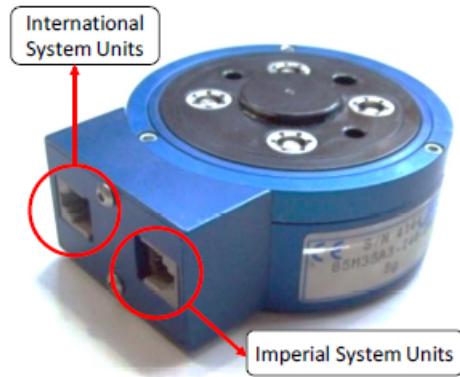
Articulación	Modelo	$F_{x,y}$	$F_z$	$M_{x,y,z}$
Muñeca	50M31A	100N	200N	5Nm
Tobillo	85M35A	250N	500N	212Nm

**Tabla 3.1:** Modelos y características de los sensores F-T. [JR3 Inc.]

Estas diferencias se deben a que los tobillos deben soportar más fuerzas y momentos no sólo producidos por cualquier carga que pueda transportar el robot sino por su propio peso también.

Los sensores de la serie M están compuestos por una serie de elementos electrónicos internos que ayudan a filtrar el ruido, proporcionan una salida digital para usar con una tarjeta PCI, para la adquisición de datos, del mismo fabricante y una salida analógica. En cuanto a su precisión los sensores de esta serie poseen una precisión nominal del 1% sobre la escala completa.

Para terminar, estos sensores permiten la adquisición de datos en unidades del Sistema Internacional o en unidades del sistema imperial, señalados en la figura 3.4.



**Figura 3.4:** Sensor fuerza-par JR3 85M35A

### 3.2.2. Adquisición de datos

Para la adquisición de datos, como se ha mencionado anteriormente, se utiliza una tarjeta PCI proporcionada por la misma compañía que los sensores JR3 Inc, cuyo modelo es PCI 1592D, compuesta por 4 puertos (nombrados en la figura 3.5). La tarjeta PCI utiliza cables de 6 u 8 pines (RJ-11 y RJ-45 respectivamente) para recibir datos de alta velocidad de los sensores que se conectan a través de estos cables, y proporcionar alimentación a dichos sensores. Esta tarjeta PCI se encuentra instalada detrás del microprocesador de manipulación en el robot.



**Figura 3.5:** Tarjeta de adquisición de datos JR3

Para acceder a los datos recibidos de los sensores, es necesario acceder a la dirección de memoria para cada dato disponible de la tarjeta.

Estos datos se pueden obtener en dos tipos de unidades: en unidades del Sistema Internacional o en unidades del Sistema Imperial, por ello hay que estar atentos de qué puerto de los sensores se están obteniendo los datos. En el SI las fuerzas están dadas en Newton [ $N$ ] y los pares en Decanewton por metro [ $dN \cdot m$ ].

### 3.2.3. Programa de adquisición de datos

El programa *jr3pci4channelYarp* (disponible en <https://github.com/roboticslab-uc3m/LoliRepo/tree/master/TFM/jr3Yarp/jr3pci4channelYarp>) lee los datos obtenidos de los 4 sensores de fuerza-par del robot TEO.

El tiempo de ciclo del programa es de aproximadamente  $20\mu s$  por cada sensor, por lo que los cuatro sensores que leen el tiempo de ciclo es de aproximadamente  $80\mu s$ .

El conjunto de datos leídos de la tarjeta de adquisición se obtiene en unidades de SI, y se manda a través de puertos YARP mediante un objeto YARP Bottle. Los datos, cuando llegan al cliente, se retrasan entre 10 y  $50\mu s$ , dependiendo del ciclo de procesamiento del lector, debido a que hasta que el cliente no finaliza el procesamiento anterior no llegan las actualizaciones de los datos, todo ello para que no se pierdan actualizaciones por parte del cliente [14].

La figura 3.6 muestra el procedimiento de adquisición de datos, y la figura 3.7 muestra una captura de pantalla del programa en ejecución.

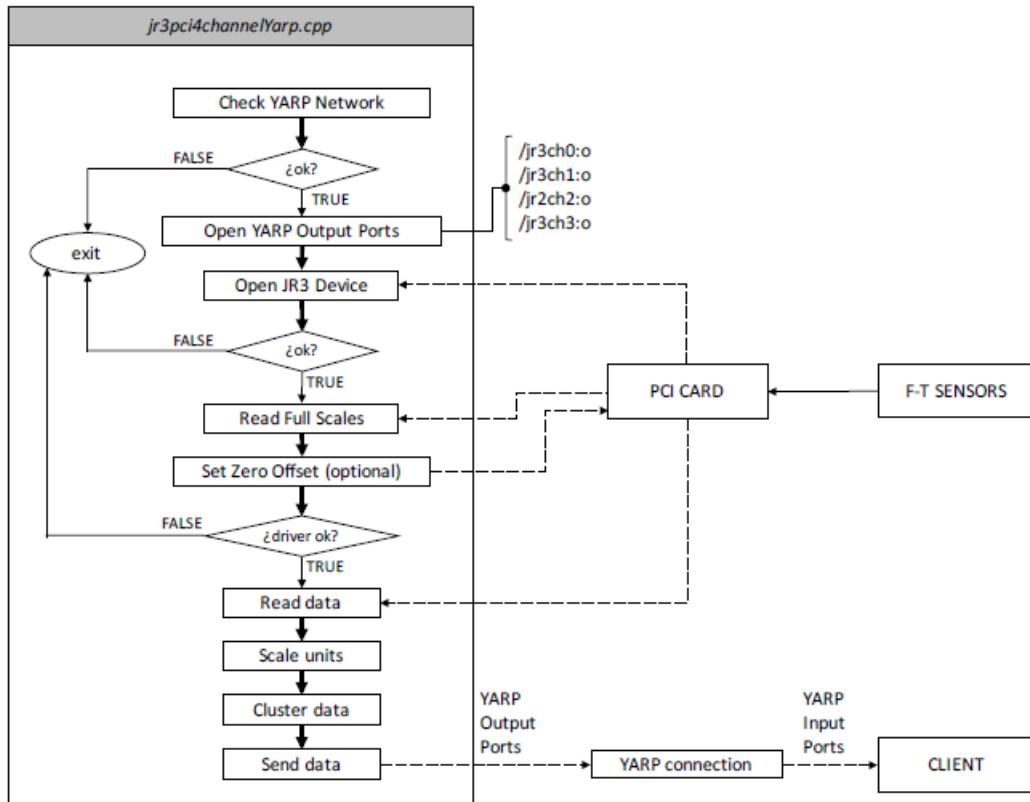


Figura 3.6: Diagrama comunicaciones con PC

```

Full scales of Sensor 0 are 317 314 845 221 219 307
Full scales of Sensor 1 are: 326 325 870 224 224 309
Full scales of Sensor 2 are: 114 106 184 55 52 60
Full scales of Sensor 3 are: 107 109 183 55 52 58

Sensor 0 data :
[1222,699,-26457,1223,809,87]
F0 = [+12.220000, +6.990000, -264.570000] N
M0 = [+12.230000, +8.090000, +0.870000] N·m
Sensor 1 data :
[-1804,1475,-33697,-454,-1006,-42]
F1 = [-18.040001, +14.750000, -336.970001] N
M1 = [-4.540000, -10.060000, -0.410000] N·m
Sensor 2 data :
[-146,203,-975,18,-109,-24]
F2 = [-1.460000, +2.030000, -9.750000] N
M2 = [+0.180000, -1.090000, -0.240000] N·m
Sensor 3 data :
[0,-77,3,-5,0,0]
F3 = [+0.000000, -0.770000, +0.030000] N
M3 = [-0.050000, +0.000000, +0.000000] N·m
t=0.000126

```

Figura 3.7: Interfaz del programa de adquisición de datos de los sensores JR3

### 3.3. Unidad de medida inercial (IMU)

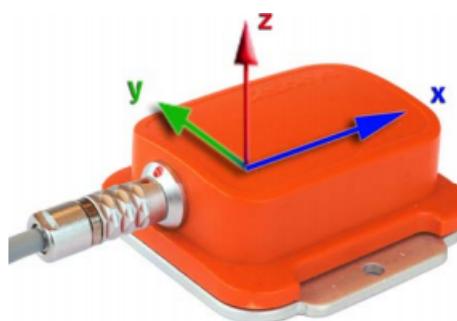
Para el sistema de percepción como se ha comentado antes se utiliza una IMU, capaz de medir aceleraciones lineales, campos magnéticos y velocidades angulares, gracias a la combinación de acelerómetros, giróscopos y magnetómetros, y en algunos casos incluso barómetros, para obtener información de la temperatura, presión y altitud.

- Acelerómetro: Dispositivo que mide el cambio de aceleración o vibración debido a los movimientos de la estructura a la que está acoplado [30].
- Giróscopo: Dispositivo mecánico que sirve para medir, mantener o cambiar la orientación en el espacio tridimensional [31].
- Magnetómetro: Dispositivo que sirve para cuantificar la fuerza o la dirección del campo magnético de un punto en el espacio [22].

#### 3.3.1. Xsens MTi-28A53G35

La unidad montada en el tronco de TEO pertenece al modelo MTi-28A53G35 proporcionado por la compañía Xsens. Se trata de un AHRS (Attitude and Heading Reference System) y a la vez de una IMU de tamaño y peso reducidos con 3 grados de libertad, compuesto por acelerómetros, giróscopos y magnetómetros 3D.

Su procesador interno de baja potencia proporciona una señal de orientación sin derivación (gracias a que utiliza la gravedad y el campo magnético de la tierra como vectores de referencia), aceleración calibrada, giros y datos de campo magnético terrestre, todo ello en 3D [13].



**Figura 3.8:** IMU

Su tamaño, peso y consumos reducidos, en adición a su robustez de diseño, lo hacen el sensor ideal para poder acoplar en cualquier dispositivo que requiera mediciones de aceleración, giros o campos magnéticos.

Entre sus características más destacadas se encuentran la capacidad de calcular en tiempo real el rumbo y datos dinámicos inerciales, orientación 360° referenciada por la gravedad y el campo magnético de la tierra, alta velocidad de actualización (120 Hz), y calibrado individual para la temperatura, la desalineación 3D y la sensibilidad cruzada del sensor. Además, el MTi incorpora una rutina de mapeado del campo magnético para corregir los efectos del hierro.

MTi-28A##G##	
Interfaz	Digital Serial (RS-232)
Voltaje de funcionamiento	4.5-15V
Consumo de Energía (modo orientación AHRS/3D)	360mW
Rango de Operación de Temperatura	0°C - 55°C
Dimensiones Exteriores	58 x 58 x 22 mm (W x L x H)
Peso	50g

**Tabla 3.2:** Características sensor IMU MTi-28A [Xsens]

Todo los datos recogidos por el sensor MTi son volcados e interpretados por la computadora gracias al MTi SDK (Software Development Kit), un paquete propietario de Xsens que sirve de interfaz a múltiples niveles: librerías binarias API(Windows, Linux), pero también proporciona código fuente implementando el protocolo de comunicación binario MTi para una fácil integración en cualquier plataforma [12].

El software para acceder al sensor inercial MTi, que está instalado en la CPU de locomoción de TEO, se ha extraído del robot iCub<sup>3</sup>, cuyo software es de código abierto siguiendo las licencias GPL/FDL y desarrollado sobre YARP, gracias a que éste tenía implementado el acceso a uno de sus sensores inerciales MTx [32].

---

<sup>3</sup>iCub se trata de un robot humanoide para la investigación en cognición corporal que fue desarrollado entre el Instituto Italiano de Tecnología y la Universidad de Génova, dentro del consorcio internacional RobotCub, en el que participan varias universidades europeas. Mide 104cm de altura, pesa 22kg y posee 53 grados de libertad. Posee sensores de cámara, micrófonos, sensores inerciales, sensores fuerza/par, sensores de posición o sensores táctiles.



**Figura 3.9:** Cable RS-232

El envío de datos se realiza a través de un cable RS-232 (CA-USB2) que incluye un convertidor a USB, como el de la figura 3.9.



## 4. Modelos simplificados usados en el estudio de la estabilidad

El control de la estabilidad en los robots bípedos resulta muy complejo. Para solucionarlo muchos investigadores han simplificado el cuerpo del robot.

En este capítulo se van a explicar las dos simplificaciones más utilizadas a la hora de estudiar la estabilidad de los robots humanoides bípedos, y más en concreto la estabilidad del robot TEO utilizado en el actual proyecto.

### 4.1. Modelo Péndulo Invertido Lineal

La primera de ellas es el modelo del péndulo invertido lineal (LIPM). Éste es el modelo más básico utilizado para simplificar la cinemática y la dinámica de los robots bípedos. Se trata de un modelo desarrollado en dos dimensiones con uno (una única unión rotacional) o dos (incluyendo una unión lineal) grados de libertad. S. Kajita [10] llegó a este modelo haciendo tres suposiciones. La primera de ellas es que la masa total del robot está concentrada en el CoM. En la segunda supone que el robot tiene piernas sin masa, cuyas puntas entran en contacto con el suelo a través de puntos de rotación individuales. Y la última considera que el robot se mueve solamente en el plano sagital (mostrado en la figura 4.1).

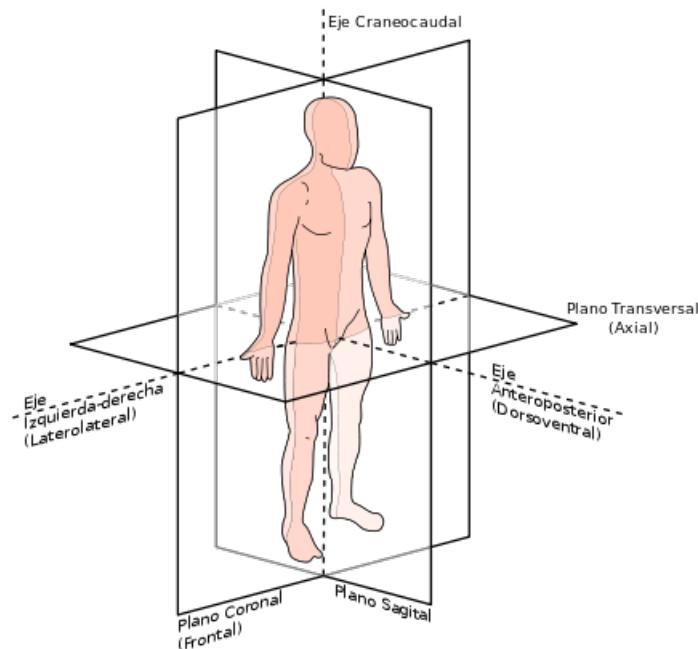
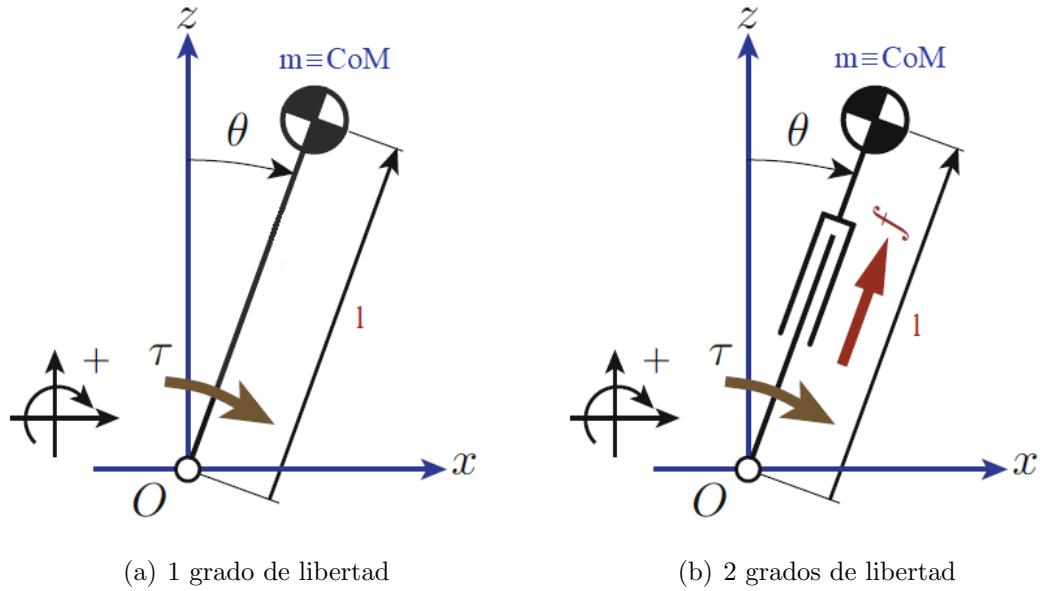


Figura 4.1: Planos Anatómicos

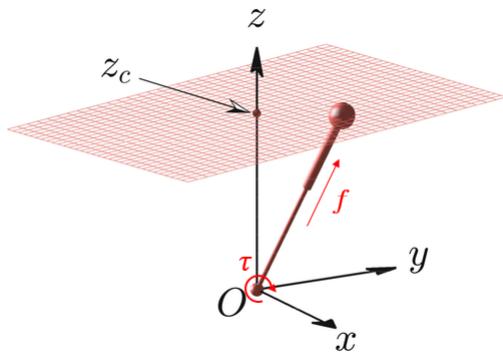


**Figura 4.2:** Modelo Péndulo Invertido Lineal

La ecuación que define el movimiento del CoM es la siguiente:

$$\tau = -ml^2\ddot{\theta} + mgl \sin \theta \quad (4.1)$$

donde  $m$  es la masa del sistema localizada en el CoM,  $l$  es la longitud del péndulo,  $\tau$  es el par en el punto de rotación y  $\theta$  es el ángulo del péndulo. Sin embargo, esta es una ecuación no lineal, por lo que resulta en un controlador muy complejo para el robot. Para resolver este problema asumimos que  $\theta$  es lo suficientemente pequeño ( $\theta < 10^\circ$ ) como para considerar que  $\sin \theta = \theta$ . Esta simplificación le permitió a Kajita desarrollar el modelo más utilizado a lo largo de los años para el estudio de la estabilidad de los robots bípedos, una evolución del modelo lineal en 2D, que le permitió trabajar en 3D, el modelo Péndulo Invertido Lineal en tres dimensiones (3DLIPM) [15].



**Figura 4.3:** Modelo Péndulo Invertido Lineal 3D (3DLIPM)

Tras la simplificación la ecuación 4.1 queda:

$$\tau = -ml^2\ddot{\theta} + mgl\theta \quad (4.2)$$

La posición del CoM  $\mathbf{p} = (x, y, z)$  está determinada por unas variables de estado  $\mathbf{q} = (\theta_r, \theta_p, r)$ , luego las ecuaciones en los 3 ejes quedan:

$$x = rS_p \quad (4.3)$$

$$y = -rS_r \quad (4.4)$$

$$z = rD \quad (4.5)$$

donde  $S_r \equiv \sin \theta_r$ ;  $S_p \equiv \sin \theta_p$ ;  $D \equiv \sqrt{1 - S_r^2 - S_p^2}$ .

Teniendo en cuenta que  $(\tau_r, \tau_p, f)$  son los pares y fuerzas asociados a las variables de estado  $(\theta_r, \theta_p, r)$ , la ecuación del movimiento del péndulo invertido 3D queda descrita matemáticamente como:

$$m \begin{pmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{z} \end{pmatrix} = (J^T)^{-1} \begin{pmatrix} \tau_r \\ \tau_p \\ f \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ -mg \end{pmatrix} \quad (4.6)$$

donde la estructura de la jacobiana es:

$$J = \frac{\partial p}{\partial q} = \begin{pmatrix} 0 & rC_p & S_p \\ -rC_r & 0 & -S_r \\ -rC_rS_r/D & -rC_pS_p/D & D \end{pmatrix} \quad (4.7)$$

$$C_r \equiv \cos \theta_r, C_p \equiv \cos \theta_p \quad (4.8)$$

$$S_r \equiv \sin \theta_r, S_p \equiv \sin \theta_p \quad (4.9)$$

$$D \equiv \sqrt{1 - S_r^2 - S_p^2} \quad (4.10)$$

Sustituyendo 4.3, 4.4 y 4.5, se obtienen las ecuaciones de la dinámica en los ejes  $x, y$  y  $z$

$$m(-z\ddot{y} - y\ddot{z}) = \frac{D}{C_r}\tau_r - mg_y \quad (4.11)$$

$$m(-z\ddot{x} - x\ddot{z}) = \frac{D}{C_p}\tau_p - mg_x \quad (4.12)$$

$$m(-x\ddot{x} + y\ddot{y} + z\ddot{z}) = rf - mg_z \quad (4.13)$$

Pero estas ecuaciones del péndulo invertido son no lineales y demasiado complejas para usarlas en la generación de tareas de caminata. Por esta razón se restringe el movimiento del CoM en la coordenada  $z$  al plano  $Z_c$ , cuyo vector normal es  $(k_x, k_y, -1)$  [38].

$$z = k_x x + k_y y + z_c \quad (4.14)$$

Si el plano de restricción es horizontal ( $k_x = k_y = 0$ ), la dinámica bajo el control de la restricción están dada por

$$\ddot{x} = \frac{g}{z_c}x - \frac{1}{mz_c}\tau_y \quad (4.15)$$

$$\ddot{y} = \frac{g}{z_c}y - \frac{1}{mz_c}\tau_x \quad (4.16)$$

$$\ddot{z} = 0 \quad (4.17)$$

En el caso de que el plano de restricción esté inclinado ( $k_x = k_y \neq 0$ ), se puede obtener la misma dinámica aplicando una restricción adicional para los pares de entrada,

$$\tau_x x + \tau_y y = 0 \quad (4.18)$$

A partir del modelo 3DLIPM junto a la restricción horizontal ( $k_x = k_y = 0$ ) podemos obtener la posición de la proyección del Punto de Momento Cero (ZMP) en el suelo [33],

$$x_{ZMP} = -\frac{\tau_y}{mg} \quad (4.19)$$

$$y_{ZMP} = -\frac{\tau_x}{mg} \quad (4.20)$$

#### 4.1.1. LIPM aplicado al robot humanoide TEO

Los sensores que se han utilizado son los F-T de los tobillos, ya que son los que están en contacto con el suelo. Gracias a su información se puede calcular la posición del ZMP para que TEO pueda conservar el equilibrio, a partir de las ecuaciones del modelo de péndulo invertido 3DLIPM.

Las ecs. 4.19 y 4.20 del ZMP para este modelo responden a la ecuación general:

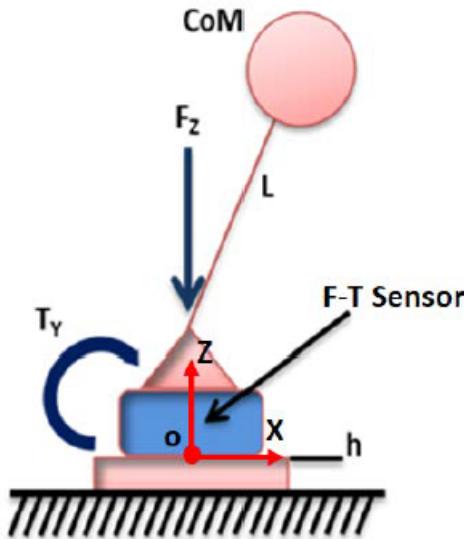
$$x_{ZMP} = -\frac{\sum x \cdot F_z}{\sum F_z} \quad (4.21)$$

Pero este modelo se ha tenido que modificar para el robot TEO utilizado en el actual proyecto, ya que se considera que cuando un robot bípedo está soportando su peso en un pie, el tobillo del robot se convierte en el punto pivotante del modelo conectado al CoM del robot a través de su pierna sin masa. Pero este punto pivotante no se encuentra totalmente en el suelo ya que la estructura de TEO está construida de tal manera que el tobillo está elevado respecto al suelo debido a la anchura de la suela del robot, como se muestra en la figura 5.1. Por lo tanto, la ecuación del ZMP adaptada sería [21]:

$$x_{ZMP} = -\frac{\tau_y + hF_x}{F_z} \quad (4.22)$$

donde  $\tau_y$  es el par en el punto pivotante alrededor del eje  $y$ ,  $F_x$  y  $F_z$  son las fuerzas

medidas en las direcciones  $x$  y  $z$  respectivamente, y  $h$  es la altura de la suela del robot hasta la localización del sensor F-T.



**Figura 4.4:** Modelo LIPM con un sensor F-T entre el tobillo y la suela [26]

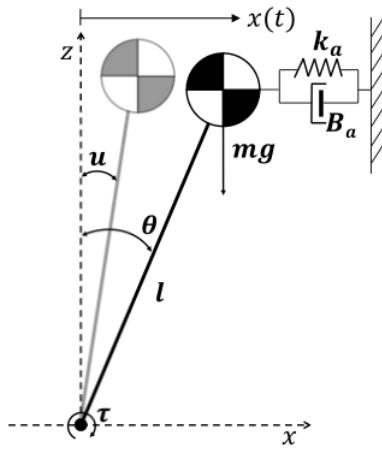
Pero esta ecuación sólo valdría cuando el robot está apoyado sobre una pierna (simple apoyo). Sin embargo, el proyecto se ha desarrollado estando apoyadas las dos piernas (doble apoyo), por lo que la ecuación para el ZMP global es:

$$x_{ZMP_{DS}} = \frac{x_{ZMP}^R F_z^R + x_{ZMP}^L F_z^L}{F_z^R + F_z^L} \quad (4.23)$$

donde R indica la pierna derecha y L la pierna izquierda. A pesar de que en doble apoyo el robot tiene dos puntos pivotantes en los tobillos, se puede aplicar el modelo de péndulo invertido simple ya que ambos tobillos tienen el mismo movimiento a lo largo del eje  $x$  (plano sagital).

#### 4.1.2. DLIPM: mejora del modelo del péndulo invertido simple

En la respuesta dinámica del modelo de péndulo invertido simple se observaba que había cierto error conforme se aumentaba el ZMP. Ésto se corrigió en [21] con una mejora de dicho modelo en la respuesta dinámica eliminando ese error, es decir, ajustando el ZMP calculado a partir de los datos obtenidos de los sensores F-T de los tobillos a un ZMP teórico al que debería llegar el robot. Este modelo se denomina DLIPM.



**Figura 4.5:** Modelo DLIPM

Esta mejora consiste en la adición de un muelle  $k_a$  y un amortiguador  $B_a$  (figura 4.5) para compensar la respuesta del estado estacionario ( $k_a$ ) y la respuesta del régimen transitorio para limitar las oscilaciones ( $B_a$ ) del sistema LIPM.

La ecuación del movimiento de dicho modelo estaría dada por:

$$\tau = -ml\ddot{x}(t) - B_a l\dot{x}(t) - k_a l x(t) + mgx(t) \quad (4.24)$$

donde  $x(t)$  indica el movimiento del CoM,  $m$  es la masa del péndulo localizada en el CoM,  $l$  la longitud del péndulo,  $k_a$  la constante del muelle y  $B_a$  la constante del amortiguador. El desplazamiento del CoM es tan pequeño que se puede asumir que  $\sin \theta = \theta$ , quedando la ecuación 4.24 como:

$$\tau = -ml\ddot{\theta}(t) - B_a l\dot{\theta}(t) - k_a l \theta(t) + mg\theta(t) \quad (4.25)$$

Por otra parte, el par también se puede obtener de la medición del ZMP como:

$$\tau = -x_{FT} \cdot mg \quad (4.26)$$

donde  $x_{FT}$  es la coordenada x del ZMP medido por los sensores de los tobillos. Combinando 4.25 y 4.26 se obtiene:

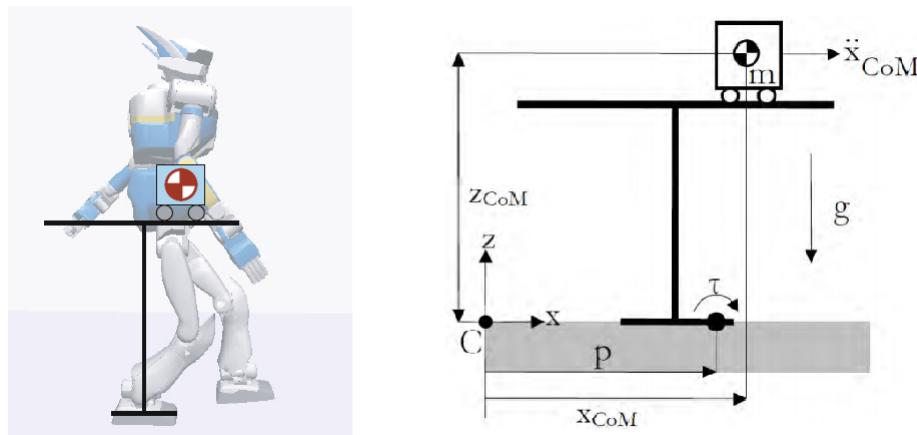
$$-x_{FT} \cdot mg = -ml\ddot{\theta}(t) - B_a l\dot{\theta}(t) - k_a l \theta(t) + mg\theta(t) \quad (4.27)$$

Siendo ésta la ecuación que explica la física del modelo DLIPM que se ha desarrollado en el actual proyecto.

## 4.2. Modelo cart-table

La siguiente simplificación más utilizada para el control del equilibrio de los robots humanoides bípedos es el modelo cart-table.

Este modelo proporciona información de las aceleraciones e inercias del cuerpo del robot, muy útiles a la hora de realizar tareas de caminata dinámica. Consiste en un carro de masa  $M$  corriendo sobre una mesa sin masa (figura 4.6). Tiene la misma dinámica de movimiento que el modelo de péndulo invertido pero añade una relación directa entre el ZMP y el movimiento.



**Figura 4.6:** Modelo Cart-Table

Las ecuaciones del modelo cart-table para controlar el ZMP se obtienen sustituyendo 4.19 y 4.20 en 4.15 y 4.16 del modelo 3DLIPM,

$$\ddot{x}_{CoM} = \frac{g}{z_{CoM}} (x_{CoM} - x_{ZMP}) \Rightarrow x_{ZMP} = x - \frac{z_{CoM}}{g} \ddot{x}_{CoM} \quad (4.28)$$

$$\ddot{y}_{CoM} = \frac{g}{z_{CoM}} (y_{CoM} - y_{ZMP}) \Rightarrow y_{ZMP} = y - \frac{z_{CoM}}{g} \ddot{y}_{CoM} \quad (4.29)$$

Estas ecuaciones se denominan **Ecuaciones del ZMP**.

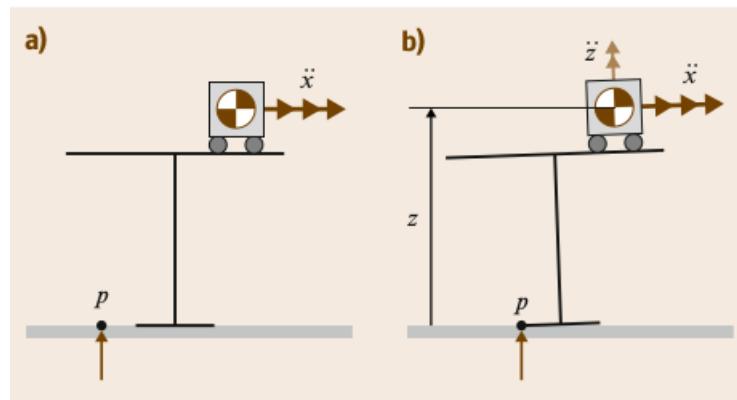
Como se puede observar el pie de la mesa es demasiado pequeño para que el carro permanezca en el borde, por lo que el carro debe llevar una determinada aceleración para que la mesa se mantenga en posición vertical durante un tiempo, en el que el ZMP exista dentro del pie de dicha mesa.

Por otro lado Shuuji Kajita y Bernard Espiau [20] añadieron que si esta aceleración es demasiado grande, el ZMP calculado se sale del polígono de soporte (4.6a). Esto es porque 4.28 y 4.29 no tienen en cuenta el polígono de soporte ni la restricción

unilateral (por lo que se asume que el pie está pegado al suelo, y esto no puede ser viable ya que el robot necesita levantar el pie para caminar). Por ello, hay que tener en cuenta dicha restricción, y como se puede observar en la 4.6b, la mesa ya no está en vertical, por lo que hay que tener en cuenta la aceleración en z a parte de la gravedad, quedando la ecuación del ZMP como:

$$\ddot{x}_{CoM} = \frac{g}{z_{CoM}} (x_{CoM} - x_{ZMP}) \Rightarrow x_{ZMP} = x - \frac{z_{CoM}}{g + \ddot{z}_{CoM}} \ddot{x}_{CoM} \quad (4.30)$$

que da el cálculo del ZMP en el borde del polígono de soporte (pie de la mesa).



**Figura 4.7:** Cálculo del ZMP. a) Modelo ficticio, b) Modelo real

Esto es importante ya que un ZMP fuera del polígono de soporte implica que el robot podría no mantener el total contacto pie-suelo y la caminata no se produjese según lo planeado. Cuando el ZMP está dentro del polígono de soporte se puede garantizar el contacto total pie-suelo.

### 4.3. Equivalencia de modelos

Para ambos modelos la variable más importante y que tienen en común es la altura del CoM, desde el punto de medida de las fuerzas y los pares para el caso del modelo LIPM (longitud del péndulo), y la longitud desde el suelo hasta el CoM para el modelo cart-table ( $Z_{CoM}$ ). Gracias a que la altura del CoM es común en los dos modelos se ha podido desarrollar una equivalencia entre ambos en el actual proyecto.

Esta equivalencia se ha desarrollado tomando como referencia el modelo LIPM (incluyendo sus mejoras), ya que era el modelo que más se acercaba al ZMP deseado en las pruebas realizadas, y a partir del mismo se despejó la altura del CoM del modelo cart-table.

Para cualquier instante  $t$  durante el paso o en posición estática, los ZMP de ambos modelos coinciden,

$$X_{ZMP_{FT_t}} = X_{ZMP_{IMU_t}} \Leftrightarrow X_{ZMP_{FT_{(t+1)}}} = X_{ZMP_{IMU_{(t+1)}}} \quad (4.31)$$

Por lo tanto:

$$ZMP_{F-T} = ZMP_{IMU} \quad (4.32)$$

$$\left. \begin{aligned} & -\frac{\tau+hF_x}{F_z} \Big|_{SS} = X_{CoM} - \frac{Z_{CoM}}{g+\ddot{z}_{CoM}} \ddot{x}_{CoM} \\ & \left. \frac{x_{ZMP}^R F_z^R + x_{ZMP}^L F_z^L}{F_z^R + F_z^L} \right|_{DS} = X_{CoM} - \frac{Z_{CoM}}{g+\ddot{z}_{CoM}} \ddot{x}_{CoM} \end{aligned} \right\} \quad (4.33)$$

donde  $x_{CoM} = 0$  ya que la referencia del sistema de coordenadas del modelo cart-table siempre es la misma, y por tanto  $z_{CoM}$  es constante en todo momento. Como se puede observar el modelo LIPM es equivalente al modelo cart-table tanto en doble como en simple apoyo.

$$x_{ZMP_{F-T}} = -\frac{Z_{CoM}}{g + \ddot{z}_{CoM}} \ddot{x}_{CoM} \quad (4.34)$$

Despejando la altura del modelo cart-table e igualando al ZMP del modelo LIPM para que ambos dieran la misma salida de ZMP, ésta quedaría:

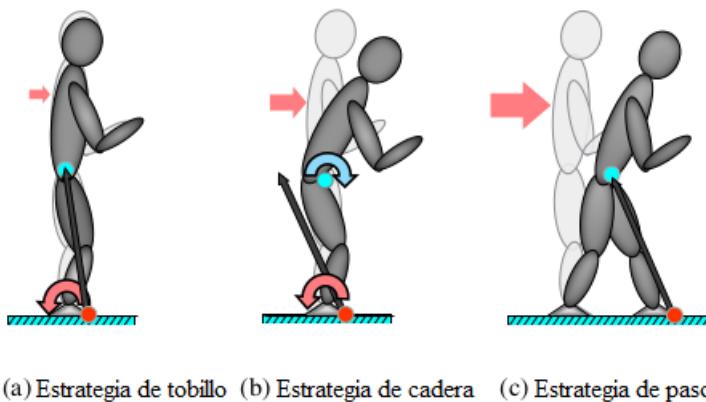
$$Z_{CoM} = -\frac{x_{ZMP_{F-T}}(g + \ddot{z}_{CoM})}{\ddot{x}_{CoM}} \quad (4.35)$$

#### 4.4. Estrategias de control

Los seres humanos son capaces de mantener el equilibrio en entornos complejos y novedosos ante diferentes perturbaciones mediante el movimiento de las diferentes partes de su cuerpo. Esta capacidad de respuesta ante perturbaciones externas ha hecho que diferentes investigadores acerquen el estudio de la estabilidad de los seres humanos a los robots humanoides.

Pueden aparecer diferentes perturbaciones que alteren el equilibrio del robot cuando éste se encuentra en una postura estable. Estas perturbaciones pueden aparecer en el plano sagital (perturbaciones anterior-posteriores) o en el plano frontal (perturbaciones mediolaterales). Se han hecho numerosos estudio del equilibrio cuando el robot se encuentra en posición estática, apareciendo diferentes estrategias de control para mantener dicho equilibrio ante la aparición de diferentes perturbaciones. Dichas estrategias se han separado en función de la magnitud de dicha perturbación y de la posición de la postura del robot en: estrategias de tobillo, cadera y paso [17] [18].

Estas estrategias están basadas en la cuantificación del cambio de dirección del CoP en las direcciones anteriormente mencionadas.



**Figura 4.8:** Estrategias de balance

La primera de ellas es la estrategia de tobillo. Ésta se aplica para pequeñas perturbaciones anterior-posteriores que tienen lugar en el plano sagital. Se usa para controlar y mantener el CoP en el polígono de soporte. Cuando ocurre una alteración del equilibrio de baja magnitud el cuerpo del robot se puede considerar como un péndulo invertido ajustándose dicho equilibrio mediante el par de la articulación del tobillo.

En la estrategia de la cadera, al igual que en la estrategia de tobillo, se actúa en el plano sagital y se basa en el balance del CMP. Ésta interviene cuando la perturbación del equilibrio es lo suficientemente intensa como para que la estrategia del tobillo no sea suficiente. Se utiliza la articulación de la cadera y se puede usar

independientemente o en combinación con la estrategia del tobillo.

La última estrategia es la del paso. Ésta se lleva a cabo cuando aparece una perturbación en la que una aplicación de par contraria en las articulaciones no es suficiente para recuperar el equilibrio. Cuando se realiza el paso el área del polígono de soporte se reajusta, creando unos nuevos límites de equilibrio.

Todas ellas dependen de diferentes factores como puede ser las condiciones del entorno (entornos rocosos o lisos, por ejemplo), la forma de la suela del robot, la altura del cuerpo, o la posición en la que se encuentre en el momento en el que se produce dicha perturbación.

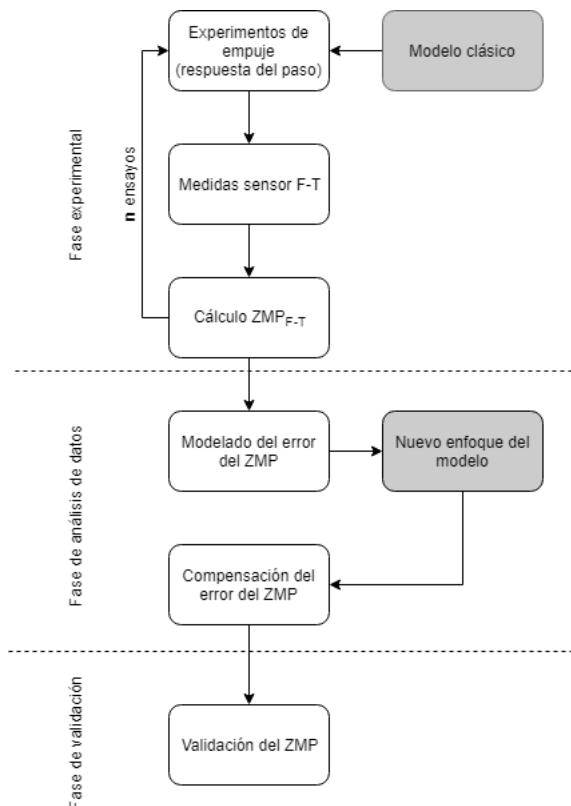


## 5. Estudio de la respuesta de los sensores del robot humanoide TEO y la aplicación de su información al control del equilibrio

En este capítulo se van a exponer las respuestas de los sistemas sensoriales del robot humanoide TEO a diferentes inclinaciones para simular perturbaciones pequeñas que cambian la posición del ZMP y mantener el equilibrio del robot haciendo uso de la estrategia de control de los tobillos. Para ello se decidió ajustar un modelo (el modelo de péndulo invertido lineal) de los dos actuales existentes y a partir de él ajustar el otro (el modelo cart-table).

### 5.1. Estudio de la respuesta de los sensores F-T

Para estudiar la respuesta de los sensores F-T ante perturbaciones y aplicar dicha información obtenida al modelo LIPM, y poder así desarrollar un nuevo modelo personalizado, se ha seguido una serie de pasos divididos en tres fases: fase experimental, fase de análisis de datos y fase de validación de resultados, que se ilustran en la figura 5.1.



**Figura 5.1:** Diagrama del desarrollo experimental del nuevo modelo [21]

### 5.1.1. Descripción de la metodología experimental

Los experimentos han seguido una serie de pasos para que las lecturas fueran correctas. Éstos se dividen en:

NOTA: Los primeros 2 pasos deben realizarse con el robot en suspensión, es decir, sin que toque el suelo con la suela de los pies. A continuación se explicará el motivo.

#### 1. Puesta del robot en posición inicial (Homeposs)

Una vez se ha encendido el robot y las CPU's se va a proceder a la puesta a cero de la posición del mismo, ya que puede haber modificaciones en su posición de anteriores experimentos o simplemente para asegurarse que los experimentos salen correctamente y siempre empiezan desde la misma posición de inicio. Esto debe realizarse con el robot en suspensión para evitar posibles colisiones de los pies con el suelo.

#### 2. Corrección offset sensores F-T

Siguiendo con el robot en suspensión y una vez realizada la posición de inicio, desde una terminal se deben iniciar los sensores F-T de los tobillos que se van a encargar de dar la información de las fuerzas y los pares al programa para así poder realizar los experimentos y eliminar los posibles offset que puedan tener los mismos. Es importante que no esté apoyado en el suelo para que cuando se ejecute esta corrección, no se elimine el valor de la fuerza ejercida por la masa del robot. Una vez que se han iniciado los sensores F-T de los tobillos ya se podría bajar el robot para que éstos puedan tener en cuenta su propio peso y las demás fuerzas y pares.

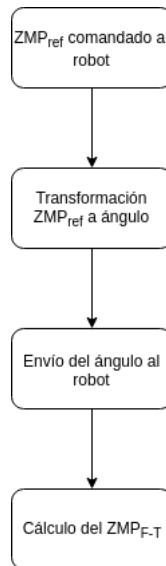
#### 3. Puesta en funcionamiento del programa

Cuando se inicia el programa, existe una primera fase en la que el robot no realiza ningún movimiento. Esto se debe a que se está configurando de tal manera que se elimine el offset que pueda haber al inicio debido a la posición de homeposs del robot, ya que ésta no es perfecta y, debido a la flexibilidad de la estructura y otros errores acumulados, puede no estar totalmente erguido y que las fuerzas en el plano axial no sean cero.

Una vez que ha pasado esta fase, y estamos seguros de que ese offset se ha eliminado en la medida de lo posible, se envía al robot un  $ZMP_{ref}$ <sup>4</sup> y TEO comanda un ángulo, que ha calculado a partir de la conversión del  $ZMP_{ref}$ , activando los motores de sus tobillos (inclinando el robot para simular el empuje) y calculando el  $ZMP_{F-T}$  del modelo de péndulo invertido lineal hasta que este  $ZMP_{F-T}$  se estabiliza, para hacer coincidir tanto la parte transitoria como la parte de régimen permanente del  $ZMP_{F-T}$  con el del  $ZMP_{ref}$ .

---

<sup>4</sup>ZMP de referencia al que el robot debería llegar atendiendo a la física y cálculos teóricos del modelo LIPM

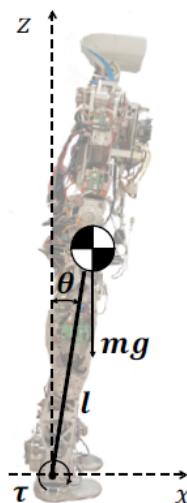


**Figura 5.2:** Diagrama de flujo del programa

### 5.1.2. Respuesta del sistema LIPM

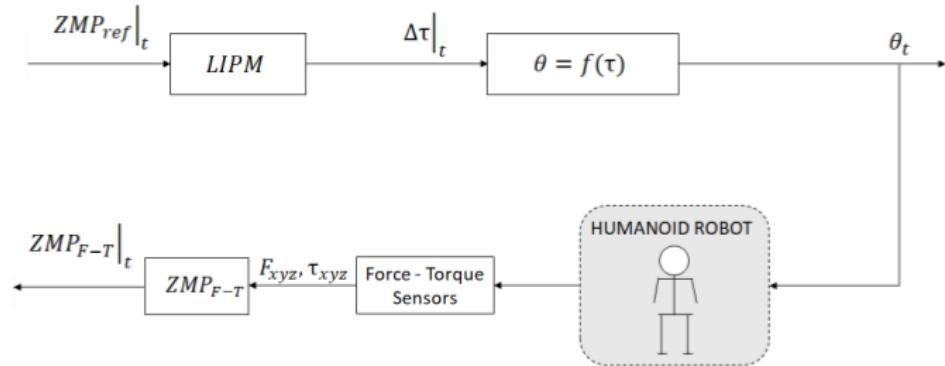
Se realizaron las pruebas sometiendo al robot a una serie de variaciones en el ángulo de los tobillos para simular perturbaciones pequeñas (empujes), con la excepción de que no volvía al ZMP inicial, para así poder modelar mejor la diferencia entre el ZMP esperado y el medido, estudiándose también su variación dinámica a lo largo del proceso.

En el presente proyecto se ha llevado a cabo una batería de 30 experimentos en el plano sagital (x-z), representado en la figura 5.3. Esta metodología también se podría aplicar para cualquier otro plano espacial ya que se obtuvieron resultados similares en el plano frontal (y-z).



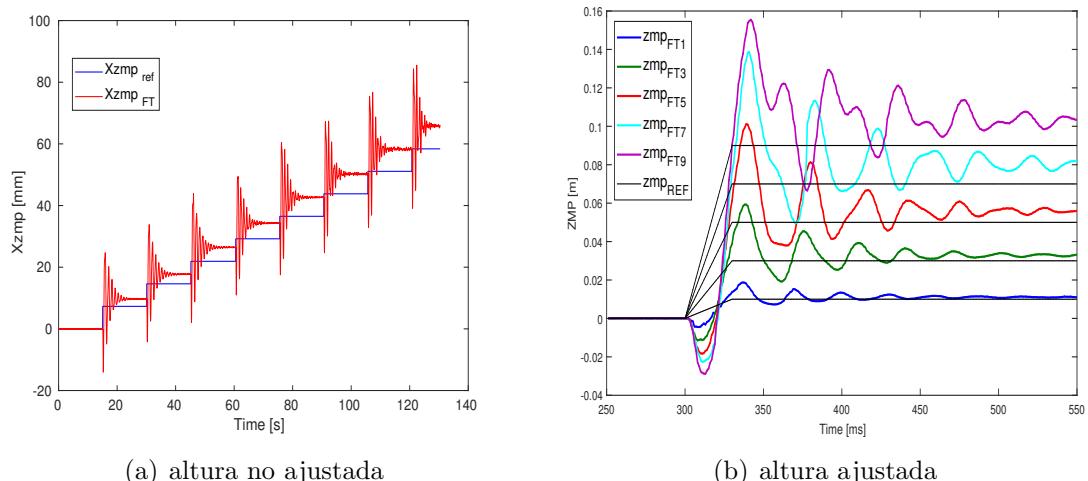
**Figura 5.3:** Plano de desarrollo de los experimentos del robot TEO [21]

La arquitectura de control utilizada en los primeros experimentos seguía el siguiente esquema:



**Figura 5.4:** Arquitectura de control de posición básica de ZMP para el modelo LIPM

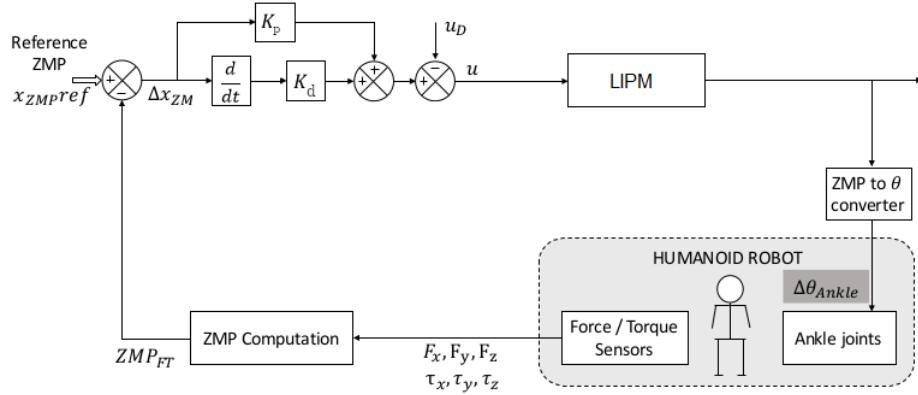
En ellos se ajustó la altura del CoM del modelo para reducir lo máximo posible la diferencia entre el ZMP calculado de los F-T y el ZMP deseado. Esta evolución se puede observar de la figura 5.5, en la que se mejoró la respuesta del sistema pero en la que todavía se mostraba un error en régimen permanente. Como se puede observar, a mayor ZMP el ángulo de inclinación es mayor y el robot es más inestable debido a que éste se encuentra más cerca del borde del polígono de soporte, y los errores tienen mayor influencia, principalmente los que tienen que ver con la flexibilidad de la estructura del robot y sus tolerancias mecánicas. Estos errores se pueden ver en el régimen permanente de la figura 5.5 b). También se observa una oscilación inicial muy grande, algo que se tiene que evitar sobre todo cuando el ZMP se encuentra en el borde del polígono de apoyo.



**Figura 5.5:** Evolucion ZMP modelo LIPM

En los siguientes experimentos, cuya arquitectura de control se muestra en la figura

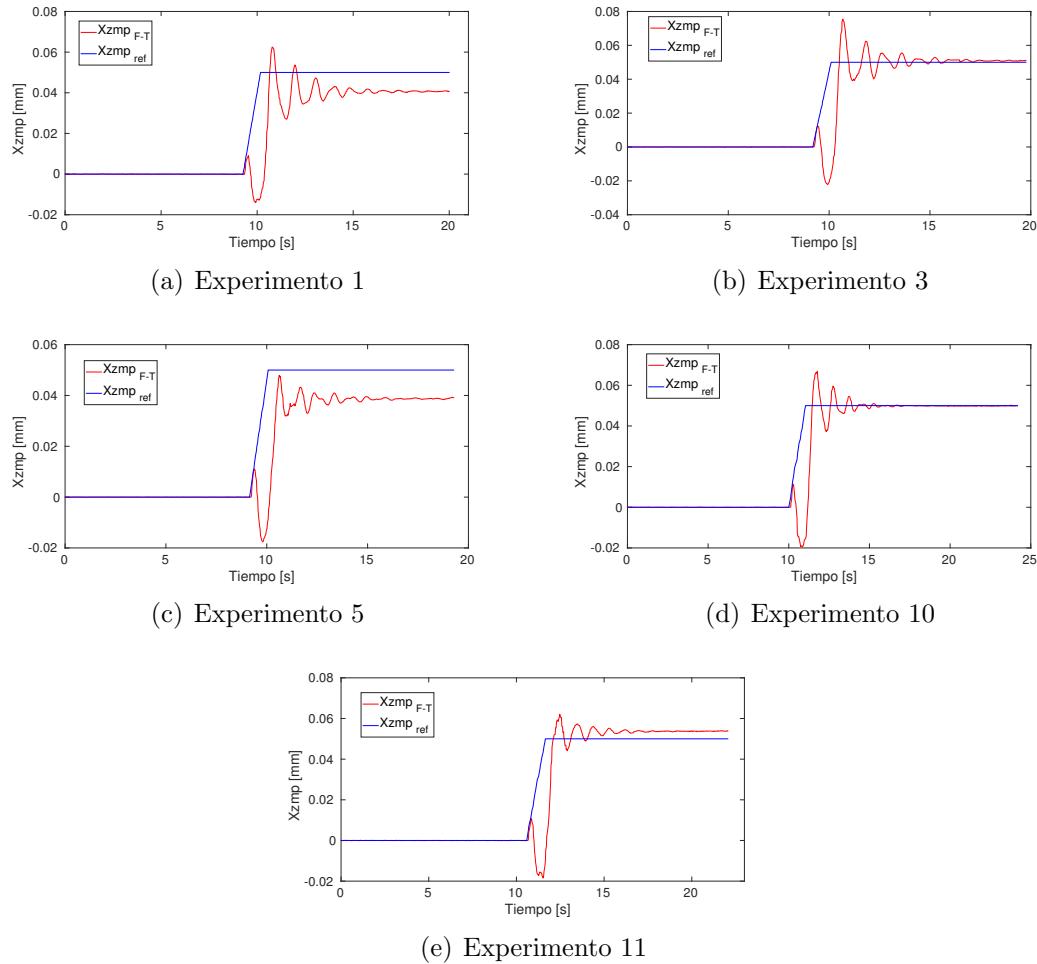
5.6, se programó un controlador en ZMP para mejorar la respuesta del sistema y reducir el error tanto en régimen permanente como en el transitorio. Se fueron ajustando diferentes variables de dicho controlador, mostrados en la tabla 5.1, para comprobar si el  $ZMP_{F-T}$  mejoraba su respuesta tanto en estado estático como en el estado transitorio.



**Figura 5.6:** Arquitectura de control de ZMP mediante controlador PD

Exp.	$k_p$	$k_d$	$k_u$
1	-0.0025	0.00005	1
2	-0.0025	0.00005	1.2
3	-0.0025	0.00005	1.4
4	-0.0025	0.00005	2
5	-0.005	0.0005	1
6	-0.005	0.0005	1.65
7	-0.01	0.0005	1
8	-0.01	0.0005	1.65
9	-0.05	0.005	1.65
10	-0.015	-0.00025	1.4
11	-0.035	-0.0001	1.4

**Tabla 5.1:** Batería de experimentos



**Figura 5.7:** Evolución ZMP para  $ZMP_{pref} = 5\text{mm}$

En las pruebas representadas en la figura 5.7, cuando se lograba ajustar la respuesta en régimen permanente, reduciendo el error entre el  $ZMP_{F-T}$  y el ZMP deseado, el régimen transitorio tenía una oscilación muy grande, y como se ha comentado antes esto se tiene que evitar sobre todo en ZMP situados en el borde del polígono de soporte. Por el contrario si se lograba hacer la oscilación inicial un poco más pequeña el tiempo de estabilización del sistema era mayor, por lo que éste se volvía más lento y no se lograba reducir el error del régimen permanente. Es por ello que se hace necesario la mejora del modelo LIPM, tanto para eliminar el error de régimen permanente como para reducir la oscilación y el sobreceso del régimen transitorio.

### 5.1.3. Respuesta del modelo mejorado DLIPM

Como se ha explicado en el apartado 4.1.2, se ha aplicado una mejora al modelo LIPM que permitía ajustar de manera dinámica la respuesta del sistema.

Aplicando Laplace a la ecuación de la física del movimiento del nuevo modelo, la función de transferencia quedaría:

$$-ml\theta(S)S^2 - B_a l \theta(S)S - k_a l \theta(S) + mg\theta(S) = -X(S) \cdot mg$$

$$\theta(S)[-ml^2S^2 - B_a l S + (-k_a l + mg)] = -X(S) \cdot mg$$

$$\frac{\theta(S)}{X(S)} = \frac{-mg}{-ml^2S^2 - SB_a l + (-k_a l + mg)}$$

$$\frac{\theta(S)}{X(S)} = \frac{\frac{g}{l^2}}{S^2 - S(\frac{B_a}{ml}) + (\frac{k_a}{ml} - \frac{g}{l})} \quad (5.1)$$

donde  $\gamma = g/l^2$ ,  $\alpha = B_a/ml$  y  $\beta = (k_a/ml) - (g/l)$ , quedando simplificada:

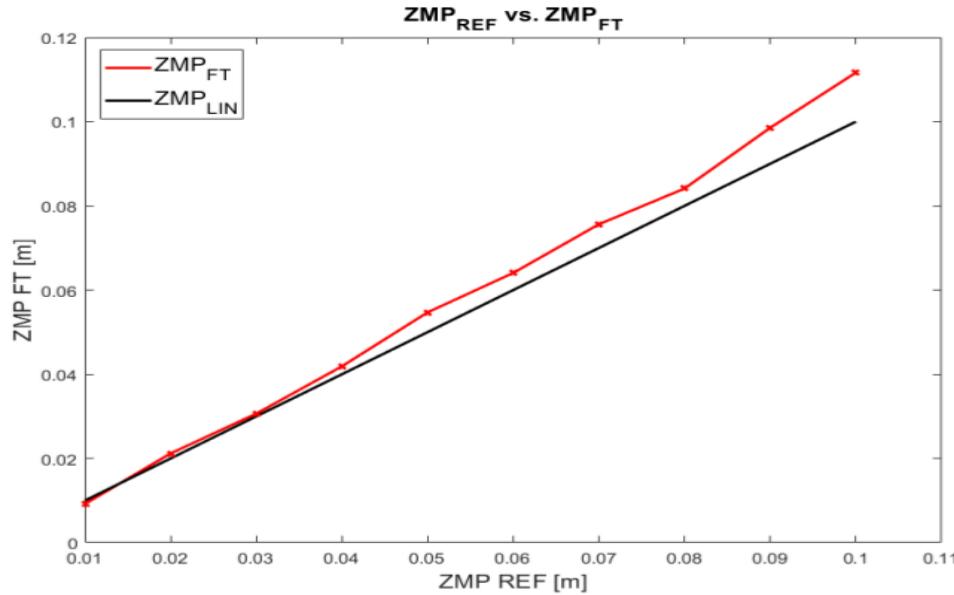
$$\frac{\theta(S)}{X(S)} = \frac{\gamma}{S^2 + \alpha S + \beta} \quad (5.2)$$

#### ■ Caracterización del error en régimen permanente

El error anteriormente comentado entre el valor deseado de ZMP y el medido, que se detecta en la figura 5.5 b), debe ser caracterizado. Es por ello que en la figura 5.8 se representa dicha desviación a partir de los datos de la figura 5.5 b) y se realiza su estudio en cada punto experimental ( $ZMP_{F-T}$ ). A partir de dicho estudio se saca una ecuación polinómica de segundo orden para modelar su desviación con respecto al  $ZMP_{ref}$ :

$$X_{ZMP_{F-T}} = a \cdot X_{ZMP_{ref}} + b \cdot X_{ZMP_{ref}} + c \quad (5.3)$$

donde  $a = 0,834$ ,  $b = 1,024$  y  $c = -0,0004$ .



**Figura 5.8:** Comparación experimental régimen permanente  $ZMP_{F-T}$  y  $ZMP_{ref}$  [21]

También sabemos que en régimen permanente, cuando el tiempo tiende a infinito, la ganancia DC del sistema de la ecuación 5.2 queda:

$$\frac{\theta(S)}{X_{ZMP_{F-T}}(S)}|_{S=\infty} = \frac{\frac{g}{l^2}}{\frac{k_a}{ml} - \frac{g}{l}} \Rightarrow K_{DC} = \frac{\gamma}{\beta} \quad (5.4)$$

$$X_{ZMP_{F-T}} = \theta_{ref} \left( \frac{-g^2 m + k_a g}{m} \right) l \quad (5.5)$$

Combinando 5.2 y 5.3, obtenemos  $k_a$ :

$$k_a = mg \left( \frac{a \cdot X_{ZMP_{ref}} + b \cdot X_{ZMP_{ref}} + c}{\theta_{ref} \cdot l} + 1 \right) \quad (5.6)$$

donde,

$$\theta_{ref} = \frac{-180}{\pi} \cdot \arcsin \left( \frac{X_{ZMP_{ref}}}{l} \right) \quad (5.7)$$

Una vez que el error estático se ha reducido gracias al parámetro  $k_a$ , se debe mejorar la respuesta del régimen transitorio para reducir tanto el tiempo de estabilización como el nivel de las oscilaciones iniciales.

- Caracterización de la respuesta transitoria del ZMP

Se ha demostrado que la respuesta del robot humanoide como péndulo invertido simple es un sistema subamortiguado. En dicho sistema se puede reducir las oscilaciones y modificar su respuesta global seleccionando adecuadamente tanto la ganancia como los parámetros dinámicos. En la figura 5.9, que muestra la diferencia entre las respuestas de las funciones de transferencia de los modelos LIPM y DLIPM ante una perturbación, en la que se puede visualizar cómo dichos parámetros dinámicos tienen la posibilidad de coger valores más altos en el modelo DLIPM, ya que posee un mayor margen de ajuste.

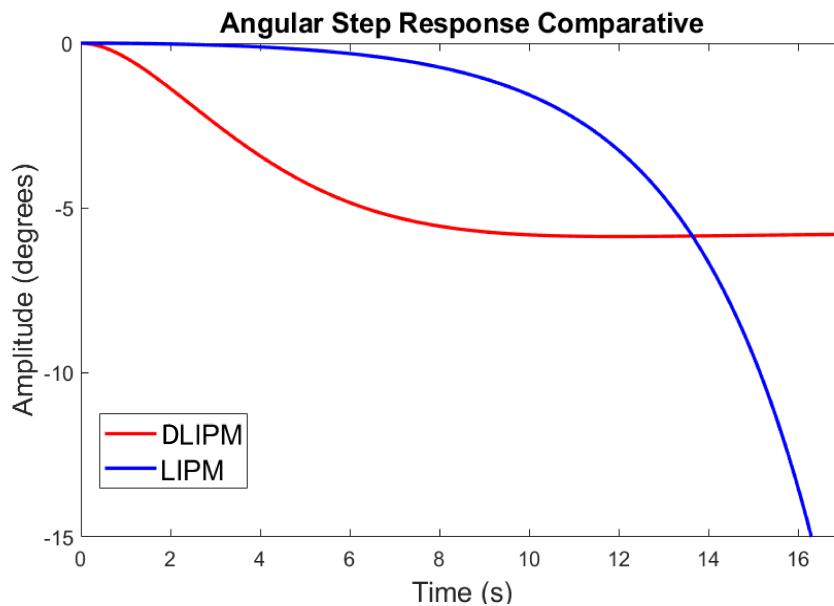


Figura 5.9: Comparativa de la respuesta angular entre LIPM y DLIPM [21]

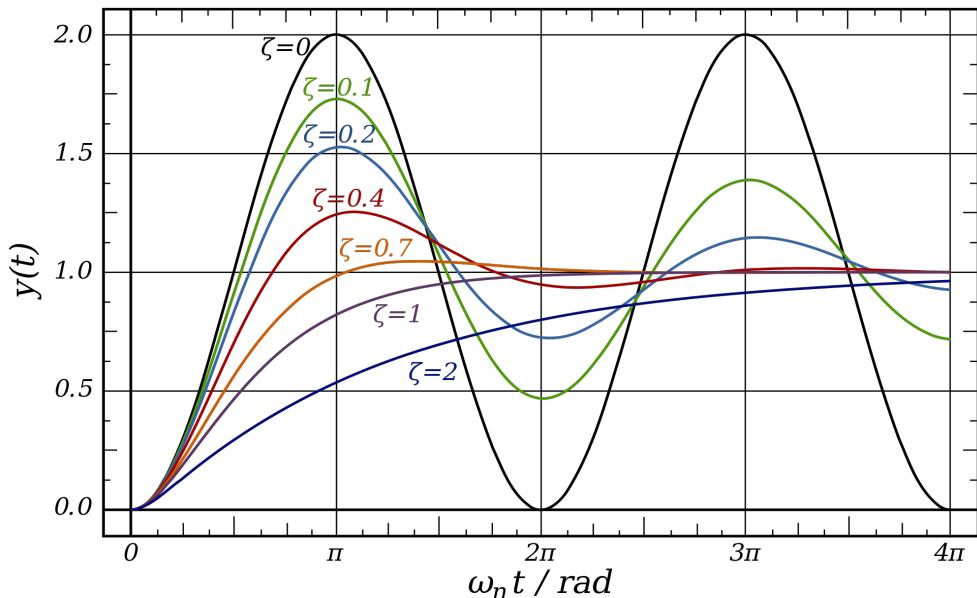
En el presente proyecto se han escogido unos valores para los parámetros dinámicos de tal manera que han permitido reducir la sobreoscilación. Estos parámetros se obtuvieron de la ecuación 5.2, ya que  $\gamma$  está relacionada con  $B_a$ , que es el parámetro del modelo DLIPM mediante el cual se puede mejorar la respuesta del sistema en régimen transitorio.

Para un sistema de segundo orden, como es el caso del modelo DLIPM, la función de transferencia en lazo cerrado se escribe como:

$$G(S) = \frac{\theta(S)}{X(S)} = \frac{Kw_n^2}{S^2 + 2\zeta w_n S + w_n^2} \quad (5.8)$$

donde  $w_n$  es la frecuencia natural no amortiguada, y  $\zeta$ , el factor de amortiguamiento relativo del sistema. Para los diferentes valores de  $\zeta$ , el

sistema actuará de diferentes maneras, como se puede observar en la figura 5.9:



**Figura 5.10:** Respuesta del sistema en función del factor de amortiguamiento

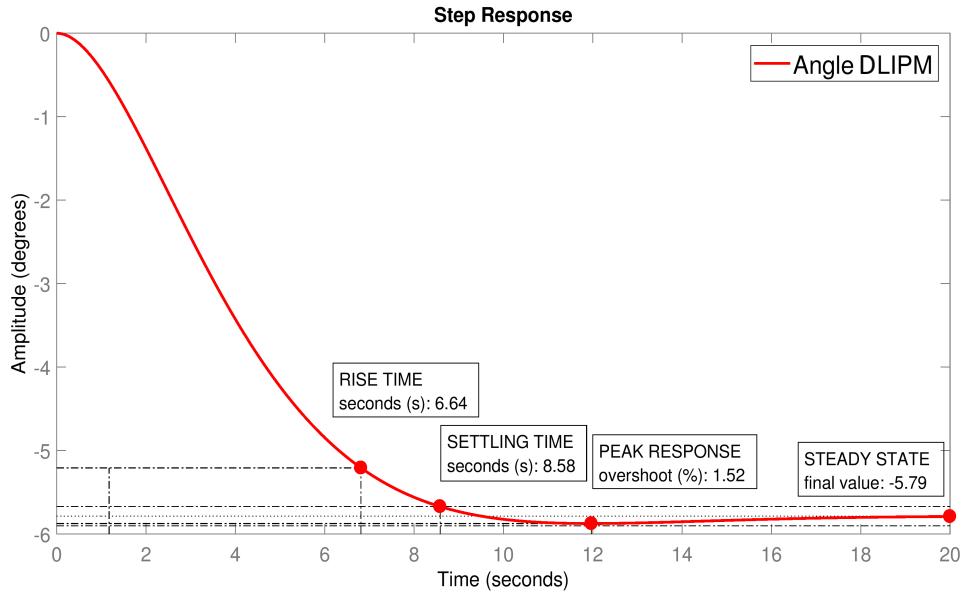
1. Críticamente amortiguado  $\zeta = 1$
2. Sobreamortiguado  $\zeta > 1$
3. Subamortiguado  $0 < \zeta < 1$

Igualando 5.1 con 5.8 se obtiene:

$$w_n = \sqrt{-\frac{g}{l} + \frac{k_a}{m}} \quad (5.9)$$

$$2\zeta w_n = \frac{B_a}{m} \quad (5.10)$$

En 5.10 como se puede observar hay 2 incógnitas para despejar, por lo que para obtener un resultado debemos fijar una de ellas y despejar la otra. Como se ha comentado antes, el sistema de péndulo invertido es un sistema subamortiguado por lo que se decidió fijar  $\zeta$ , y despejando de 5.10, obtenemos  $B_a$ . Para el actual proyecto, los parámetros dinámicos que se diseñaron fueron  $\zeta = 0,8$  y  $w_n = 0,4376$ .

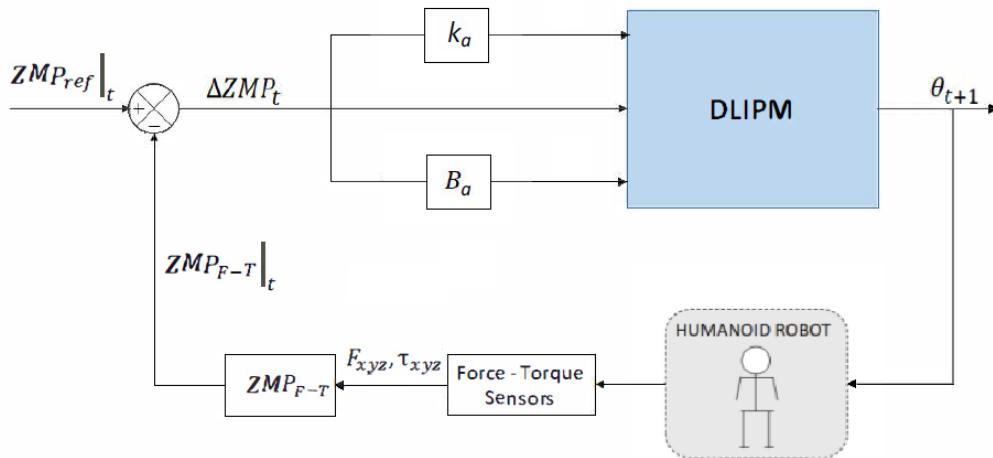


**Figura 5.11:** Respuesta angular del modelo DLIPM [21]

#### 5.1.4. Control de ZMP

Como el objetivo de este trabajo es abarcar varios puntos de trabajo, se ha propuesto un modelo de controlador no lineal, basado en una ganancia programada, que mediante la selección adecuada de los parámetros dinámicos, permite manejar múltiples puntos de trabajo. Un controlador por ganancia programable es un sistema en el que los parámetros de dicho controlador pueden variar en función de las condiciones de operación o los parámetros de la planta [22]. Este controlador utiliza tablas para especificar los valores de la ganancia en función de los parámetros programados.

Las siguientes pruebas se han llevado a cabo siguiendo el esquema de la figura 5.12, similar a la arquitectura de control utilizada en [21]. Para el actual proyecto, dependiendo de la entrada, se pueden seleccionar los valores  $k_a$  y  $B_a$  adecuados gracias al módulo que permite seleccionar estos parámetros según su planificación. Una vez seleccionados, se calculan los coeficientes de la planta del modelo DLIPM, modificando así su dinámica. Por último, una vez que se ha calculado la planta del modelo DLIPM, éste comanda un ángulo a los tobillos del robot, siendo más progresiva su implementación que en el modelo LIPM.



**Figura 5.12:** Arquitectura de control del modelo DLIPM

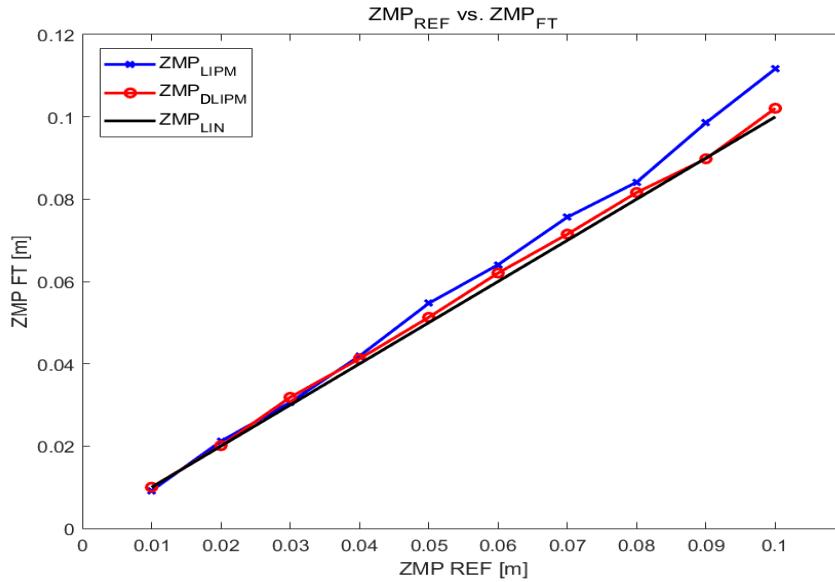
### 5.1.5. Validación de datos experimentales

Se realizaron diversos experimentos con el nuevo modelo DLIPM, que consistían en la variación del ZMP deseado para visualizar la respuesta del nuevo sistema de control. Se siguió la misma metodología que para el modelo LIPM.

Para la respuesta en régimen permanente de los experimentos se ha desarrollado la tabla 5.2, en la que se muestra una comparación de la ubicación del ZMP deseado y medido entre el modelo LIPM y el modelo DLIPM. Como se puede observar se ha reducido el error estático en todos los puntos de trabajo. Ésta mejora puede observarse de manera gráfica en la figura 5.13.

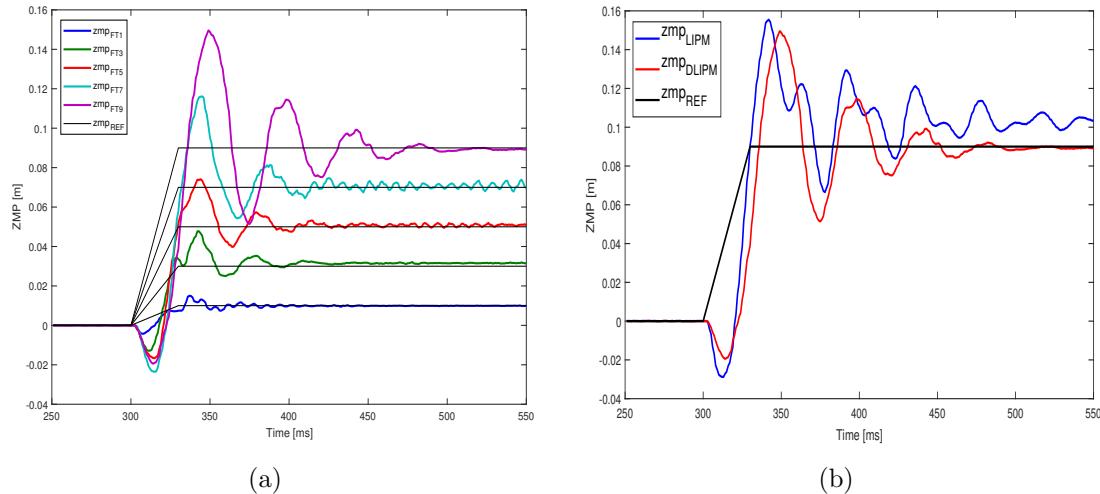
ZMP <sub>ref</sub> (m)	ZMP <sub>F-T</sub> (m)		Error (%)	
	LIPM	DLIPM	LIPM	DLIPM
0.00	5x10 <sup>-5</sup>	2x10 <sup>-7</sup>	0.0	0.0
0.01	0.0092	0.0100	8.2	0.0
0.02	0.0211	0.0201	6.0	0.5
0.03	0.0306	0.0310	2.2	3.3
0.04	0.0419	0.0412	4.8	3.0
0.05	0.0547	0.0512	9.4	2.4
0.06	0.0641	0.0620	6.8	3.3
0.07	0.0756	0.0714	8.0	2.0
0.08	0.0841	0.0816	5.2	2.0
0.09	0.0902	0.0989	9.5	0.2
0.10	0.1116	0.1020	11.6	2.0

**Tabla 5.2:** Tabla comparación ZMP entre modelo LIPM y DLIPM



**Figura 5.13:** Comparación ZMP de los modelos LIPM y DLIPM

En cuanto al régimen transitorio, la respuesta se muestra en la figura 5.14. Se puede observar, comparando ésta con la figura 5.5, que la respuesta ha mejorado tanto en régimen permanente como en transitorio, reduciendo las oscilaciones, en número pero no tanto en amplitud, y el tiempo de estabilización del robot.



**Figura 5.14:** Evolucion ZMP modelo DLIPM. a) Respuesta modelo DLIPM ante diferentes simulaciones de perturbaciones; b) Comparación entre modelo LIPM y DLIPM para  $ZMP_{ref} = 9cm$  [21].

## 5.2. Estudio de la respuesta del sensor inercial IMU

### 5.2.1. Descripción de la metodología experimental

Para los experimentos del modelo cart-table, se siguió la misma metodología de inicio del robot que para el modelo LIPM:

NOTA: Al igual que para las pruebas anteriores, es necesario que en los primeros dos pasos el robot tiene esté en suspensión.

#### 1. Puesta del robot en posición inicial (Homeposs)

#### 2. Corrección offset sensores F-T

#### 3. Puesta en marcha del sensor inercial IMU

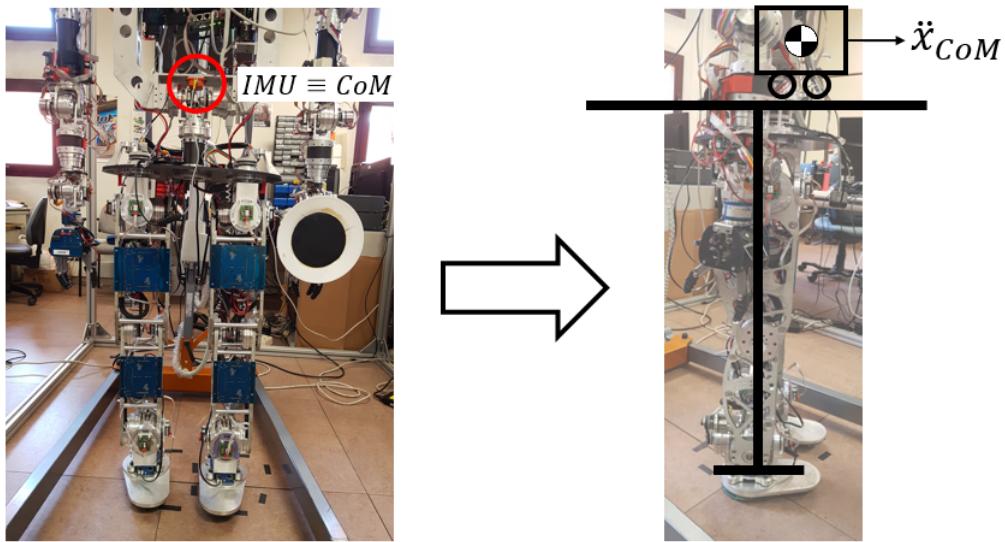
Una vez que se han realizado los dos pasos anteriores, se debe iniciar la IMU para poder obtener datos de aceleraciones del robot para calcular la altura  $z_{CoM}$  del modelo cart-table a partir del ZMP del modelo DLIPM. Por último, se baja el robot antes de iniciar el programa.

#### 4. Puesta en marcha del programa

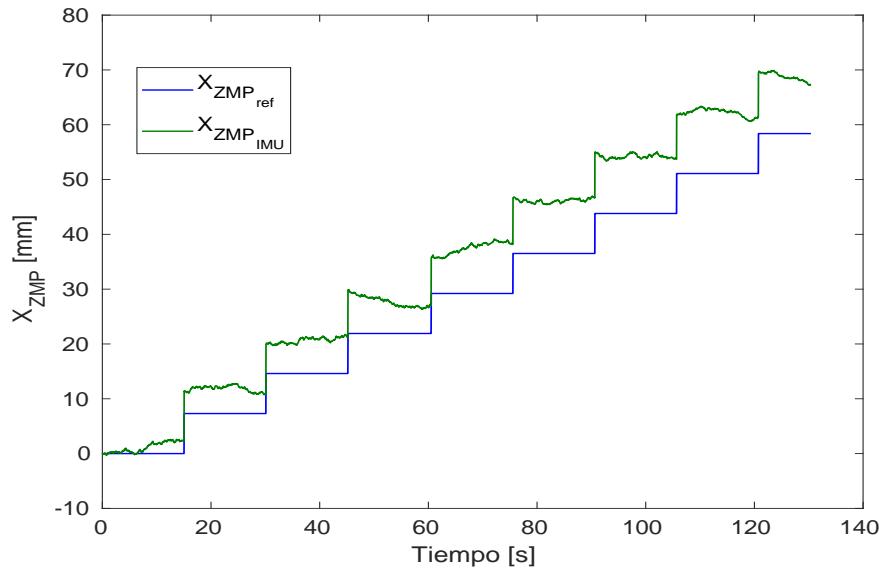
Para los experimentos de la IMU se ha desarrollado un programa que iniciase en paralelo el programa del modelo DLIPM y a la vez el del modelo cart-table, tanto para inclinar el robot como para calcular los parámetros necesarios. Como se ha comentado en la sección 4.3, ambos modelos son equivalentes, por lo que el parámetro de la altura en el modelo cart-table se calculaba a partir del modelo DLIPM, de ahí que los programas se ejecutasesen en paralelo, porque se necesitaban los datos de ambos modelos. Una vez que la altura se calculaba, se realizaba el cómputo del ZMP a partir de los datos obtenidos de la IMU.

### 5.2.2. Respuesta del modelo cart-table

Como se ha comentado en el apartado 4.3, tanto el modelo de péndulo invertido simple como el cart-table son distintas formas de computar el ZMP. En [22] el modelo cart-table presentaba desajustes en régimen permanente, como se puede observar en 5.16.

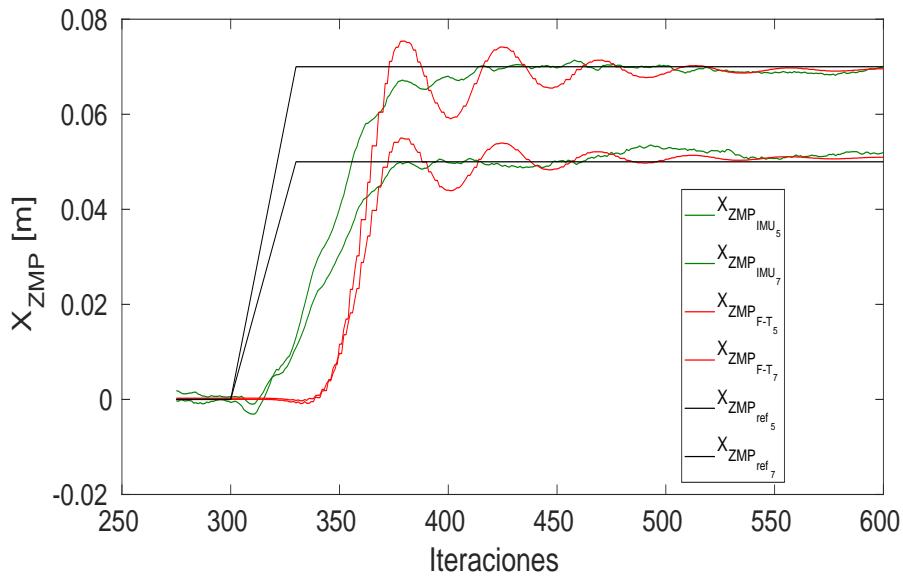


**Figura 5.15:** Modelo cart-table en el robot TEO



**Figura 5.16:** Evolucion ZMP modelo cart-table

Se hizo el estudio de ambos modelos para averiguar los parámetros más importantes de ambos a la hora de computar el ZMP. Se observó que la altura del CoM necesitaba ser modificada en el modelo cart-table para que ambos modelos dieran el mismo ZMP.



**Figura 5.17:** Representación ZMP modelo cart-table mejorado

Una vez que se computó la altura del modelo cart-table, se volvieron a hacer las pruebas, mejorando su respuesta en régimen permanente. La corrección de dicho desajuste en régimen permanente puede observarse en la figura 5.17 en la que se indica de color negro la referencia a la que debían llegar los ZMP tanto del modelo DLIPM (en rojo) como el modelo cart-table (en verde).



## 6. Conclusiones y trabajos futuros

Una vez realizadas todas las pruebas, se pueden sacar unas conclusiones que a continuación se expondrán, verificando si se han cumplido o no los objetivos propuestos en el presente proyecto, y se propondrán una serie de mejoras para trabajos futuros.

Como se ha descrito en este trabajo, éste consta de unas fases en las que se ha comprobado el correcto funcionamiento de los sensores, se han ajustado los modelos y se ha verificado su mejora mediante los diferentes experimentos.

El objetivo inicial fue realizar un estudio del correcto funcionamiento de los sensores, tanto de los de fuerza-par como de la IMU, tanto en el robot como fuera de él mediante sensores externos que tenían las mismas características que los acoplados en el propio robot, y del programa que hace que dicha información pueda ser utilizada para el control dentro del ordenador.

Una vez que ésto se verificó, se aplicaron los datos obtenidos de los sensores a los modelos estudiados a lo largo del presente trabajo. Se observó que al aplicar dicha información a los modelos, tanto LIPM como cart-table, éstos presentaban inexactitudes, debidas tanto a imperfecciones en el robot debidas a las tolerancias mecánicas como a la elasticidad de su estructura. Para intentar solucionar dichas imperfecciones e inexactitudes, se aplicó un controlador PD, pero los resultados no fueron del todo satisfactorios, por lo que se aplicó una mejora al modelo LIPM para mejorar la respuesta del controlador que permitió acercar el punto de equilibrio del robot humanoide TEO al ZMP ideal. Dicha progreso hizo que el modelo LIPM pasase a denominarse DLIPM. En él se consiguió mejorar la respuesta global de los sensores, tanto en régimen permanente, reduciendo el desajuste al mínimo con respecto al ZMP de referencia, como en régimen transitorio, reduciendo la amplitud de la oscilación inicial y el tiempo de estabilización del robot.

Una vez que se verificó que la respuesta del controlador en el nuevo modelo DLIPM se incrementaba, ésta se equiparó en el modelo cart-table de la misma forma que para el modelo LIPM, ajustando su respuesta a partir de la premisa que los ZMP de ambos modelos debían coincidir, y como se observa en la figura 5.16, dicho objetivo se logró.

Como trabajos futuros se proponen las siguientes ideas:

- Como la mejora aplicada al modelo LIPM no conseguía reducir lo suficiente la amplitud de la oscilación inicial para asegurar que el robot mantendría el equilibrio en todo momento, sobre todo cuando el ZMP se sitúa al borde del polígono de soporte. Por tanto, se propone mejorar el controlador PD utilizado al principio.

- Otro posible trabajo futuro sería la utilización de modelos más complejos que permitieran un ajuste más fino, como puede ser el péndulo doble.
- Para concluir, como las medidas de los sensores se ven afectadas por elementos físicos, se plantea mejorar las elasticidades mecánicas rigidizando la estructura del robot, una mejor situación de los sensores, la utilización de otro tipo de sensores, es decir, cambiar componentes del robot para evitar al máximo la influencia de esos elementos físicos en las medidas.



## 7. Presupuesto

En este apartado se pretende mostrar un presupuesto general de dicho proyecto. Para llevarlo a cabo se han tenido en cuenta tanto los materiales utilizados como la mano de obra por parte de los participantes.

COSTE MATERIAL				
Ítem	Descripción	Coste/ud (€/ud)	Ud	Coste (€)
1	Ordenador Acer Aspire E1-571  - Procesador Intel Core i3-2328M (2,2GHz, 3MB L3 cache) - Memoria RAM 6 DDR3 - Disco duro 500GB - Monitor, Conexión ethernet y otros	899	1	899
2	Sistema Operativo Linux (Licencia gratuita)	-	-	-
3	Programas (Licencia gratuita)  - Python - Qt Creator	-	-	-
<b>TOTAL</b>				<b>899€</b>
COSTE LABORAL				
Ítem	Descripción	Coste/h (€/h)	Horas	Coste (€)
1	Tiempo dedicado proyectante  - Desarrollo de algoritmos (pruebas, programación,...) - Memoria	15  15  15	370  250  120	5550  3750  1800
2	Tiempo tutorización	30	80	2400
<b>TOTAL</b>				<b>7950€</b>

Tabla 7.1: Presupuesto



## Referencias

- [1] Milicua Ainhoa, "Las 6 leyes de la robótica de la Unión Europea", Blogthinking, 21-06-2017. [En línea]. Disponible en: <https://blogthinkbig.com/las-6-leyes-de-la-robotica-de-la-union-europea>.
- [2] Wikipedia, Robótica", Wikipedia, 2008. [En línea]. Disponible en: <https://es.wikipedia.org/wiki/Rob%C3%BCunhbox\voidb@x\bgroup\let\unhbox\voidb@x\setbox\@tempboxa\hbox{o\global\mathchardef\accent@spacefactor\spacefactor}\accent19o\egroup\spacefactor\accent@spacefactor\futurelet\@let@token\penalty@\M\hskip\z@skiptica>.
- [3] Wikipedia, Robot", Wikipedia, 2007. [En línea]. Disponible en: <https://es.wikipedia.org/wiki/Robot>.
- [4] International Organization for Standardization, Robots and robotic devices – Vocabulary", ISO, 03-2012. [En línea]. Disponible en: <https://www.iso.org/standard/55890.html?browse=tc>.
- [5] Asociación Española de Robótica y Automatización, AER-Automation. [En línea]. Disponible en: <https://www.aer-automation.com/aer-atp/robotica-industrial-y-de-servicio/>. Acceso: junio 2018.
- [6] K. Sanchez Madriz, Clasificación de los robots según su arquitectura", El avance de la Robótica, 31-10-2011. [En línea]. Disponible en: <https://sites.google.com/site/elavancedelarobotica/clasificacion-de-los-robots/clasificacion-de-los-robots-su>.
- [7] Robots and Androids, "The History of Robots", Robots and Androids. [En línea]. Disponible en: <http://www.robots-and-androids.com/history-of-robots.html>. Acceso: junio 2018.
- [8] Md Akhtaruzzaman y A.A. Shafie, Evolution of Humanoid Robot and contribution of various countries in advancing the research and development of the platform", presentada en Control Automation and Systems (ICCAS), Korea, 27-30 oct. 2010. [En línea]. Disponible en: <https://ieeexplore.ieee.org/document/5669646/>. Acceso: julio 2018.
- [9] T. Martín y A. Serrano, "Dinámica de sistemas. Centro de masas", Universidad Politécnica de Madrid. [En línea]. Disponible en: <http://www2.montes.upm.es/dptos/digfa/cfisica/dinamsist/cdm.html>.
- [10] S. Kajita, H. Hirukawa, K. Harada y K. Yokoi, Introduction to Humanoid Robotics", Springer-Verlag Berlin Heidelberg, 2014. [En línea]. Disponible en: <https://link.springer.com/book/10.1007%2F978-3-642-54536-8>.

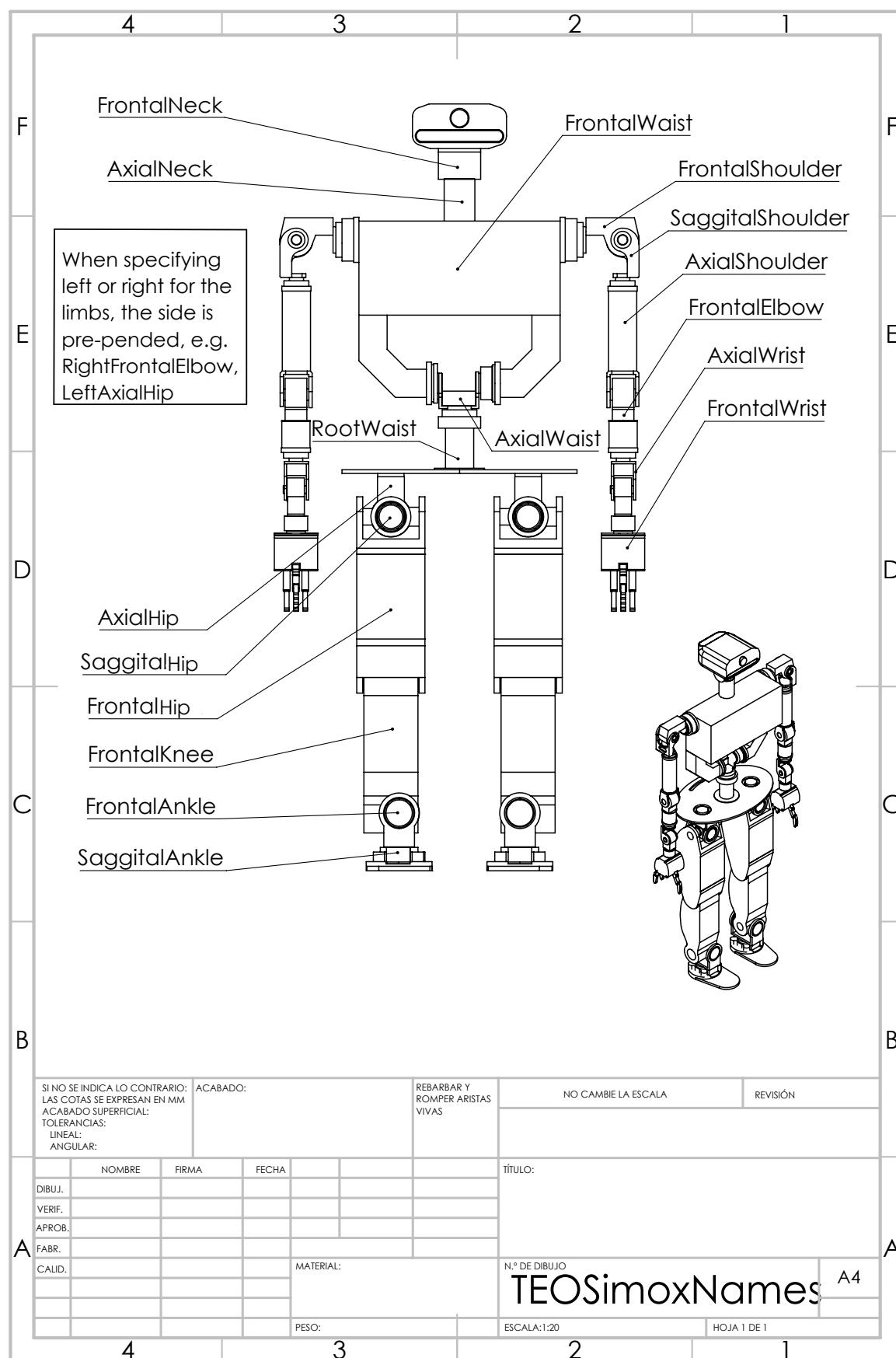
- [11] S. Fernández, "Locomoción bípeda del robot humanoide NAO", Proyecto final de carrera, Universidad Politécnica de Catalunya, Cataluña, España, 2009. [En línea]. Disponible en: <https://upcommons.upc.edu/handle/2099.1/9115>. Acceso: julio 2018.
- [12] Xsens Technologies B.V., "MTi Miniature Attitude and Heading Reference System", Xsens Technologies B.V., Holanda, 2009.
- [13] Xsens Technologies B.V., "MTi and MTx User Manual and Technical Documentation", Xsens Technologies B.V., Holanda, Informe técnico MT0100P , 2006.
- [14] G. Metta, P. Fitzpatrick y L. Natale, "YARP: Yet another robot platform", *International Journal of Advanced Robotic Systems*, vol. 3, n.º 1, pp. 43-48, marzo 2006. [En línea]. Disponible en: <http://journals.sagepub.com/doi/abs/10.5772/5761>. Acceso: julio 2018.
- [15] S. Kajita, F. Kanehiro, K. Kaneko, K. Yokoi y H. Hirukawa, "The 3D Linear Inverted Pendulum Mode: A simple modeling for a biped walking pattern generation", presentada en International Conference on Intelligent Robots and Systems, Maui, Hawaii, USA, Oct. 29 - Nov. 03, 2001. [En línea]. Disponible en: <https://pdfs.semanticscholar.org/6a31/6e0d44e35a55c41a442b3f0d0eb1f9d4d0ca.pdf>. Acceso: julio 2018.
- [16] M. Vukobratović y B. Borovac, "Zero Moment Point - Thirty five years of its life", *International Journal of Humanoid Robotics*, vol. 1, n.º 1, pp. 157-173, 2004. [En línea]. Disponible en: [https://www.researchgate.net/publication/220065796\\_Zero-Moment\\_Point\\_-\\_Thirty\\_Five\\_Years\\_of\\_its\\_Life](https://www.researchgate.net/publication/220065796_Zero-Moment_Point_-_Thirty_Five_Years_of_its_Life). Acceso: julio 2018.
- [17] D.A. Winter , F. Prince, J. S. Frank, C. Powell and K. F. Zabjek, Unified theory regarding A/P and M/L balance in quiet stance", *Journal of neurophysiology*, vol. 75, n.º 6, pp. 2334-2343, julio 1996. [En línea]. Disponible en: [https://www.researchgate.net/publication/14409014\\_Unified\\_theory\\_regarding\\_AP\\_and\\_ML\\_balance\\_in\\_quiet\\_stance](https://www.researchgate.net/publication/14409014_Unified_theory_regarding_AP_and_ML_balance_in_quiet_stance). Acceso: agosto 2018.
- [18] D.N. Nenchev y A. Nishio, .Experimental Validation of Ankle and Hip Strategies for Balance Recovery with a Biped Subjected to an Impact", presentada en International Conference on Intelligent Robots and Systems, San Diego, CA, USA, Oct. 29 - Nov. 2, 2007. [En línea]. Disponible en: [https://www.researchgate.net/publication/4296918\\_Experimental\\_validation\\_of\\_ankle\\_and\\_hip\\_strategies\\_for\\_balance\\_recovery\\_with\\_a\\_biped\\_subjected\\_to\\_an\\_impact](https://www.researchgate.net/publication/4296918_Experimental_validation_of_ankle_and_hip_strategies_for_balance_recovery_with_a_biped_subjected_to_an_impact).
- [19] M. Vukobratović, "Humanoid Robotics - Past, Present State, Future", presentada en 4<sup>th</sup> Serbian-Hungarian Joint Symposium on Intelligent Systems,

- Belgrado, 2006. [En línea]. Disponible en: [http://conf.uni-obuda.hu/sisy2006/1\\_Vuk.pdf](http://conf.uni-obuda.hu/sisy2006/1_Vuk.pdf).
- [20] S. Kajita y B. Espiau, "16. Legged Robot", en B. Siciliano y O. Khatib (Eds.) *Springer Handbook of Robotics*. Berlin, Alemania: Springer-Verlag, 2008, pp. 361-389.
- [21] S. Martínez, J.M. García-Haro, J.G. Victores, A. Jardon y C. Balaguer, "Experimental Robot Model Adjustments Based on Force-Torque Sensor Information", *Sensors*, vol. 18, n.º 3, art. nº 836, marzo 2018. [En línea]. Disponible en: <http://www.mdpi.com/1424-8220/18/3/836>.
- [22] J. Lorente, "Equilibrium control for humanoid robot TEO by inertial perception", Trabajo fin de máster, Dpto. de Ingeniería de Sistemas y Automática, Universidad Carlos III de Madrid, Madrid, España, 2016.
- [23] M. D. Pinel, "Balance control of humanoid robot TEO using force/torque sensors", Trabajo fin de máster, Dpto. de Ingeniería de Sistemas y Automática, Universidad Carlos III de Madrid, Madrid, España, 2016.
- [24] M. H. P. Dekker, "Zero-Moment Point method for stable biped walking", University of Technology, Eindhoven, Informe Técnico 2009.072, Julio 2009.
- [25] M. B. Popovic, A. Goswani y H. Herr, "Ground Reference Points in Legged Locomotion: Definitions, Biological Trajectories and Control Implications", *The International Journal of Robotics Research*, ene. 2005.
- [26] S. Martínez, "Human Inspired Humanoid Robot Control Architecture", Tesis doctoral, Dpto. de Ingeniería de Sistemas y Automática, Universidad Carlos III de Madrid, Leganés, España, 2012. [En línea]. Disponible en: <https://e-archivo.uc3m.es/handle/10016/16422>.
- [27] AENOR, UNE-EN ISO 10218-1:2012", *AENOR*, 2012. [En línea]. Disponible en: <https://www.aenor.com/normas-y-libros/buscador-de-normas/UNE?c=N0049289>.
- [28] AENOR, UNE-EN ISO 10218-2:2011", *AENOR*, 2011. [En línea]. Disponible en: <https://www.aenor.com/normas-y-libros/buscador-de-normas/UNE?c=N0048668>.
- [29] "Mars Pathfinder Rover: Sojourner", *Smithsonian: National Air and Space Museum*, 2011. [En línea]. Disponible en: <https://airandspace.si.edu/multimedia-gallery/web12070-2011640.jpg>.
- [30] Omega Engineering, "Acelerómetro", *Omega*. [En línea]. Disponible en: <https://es.omega.com/prodinfo/acelerometro.html>.

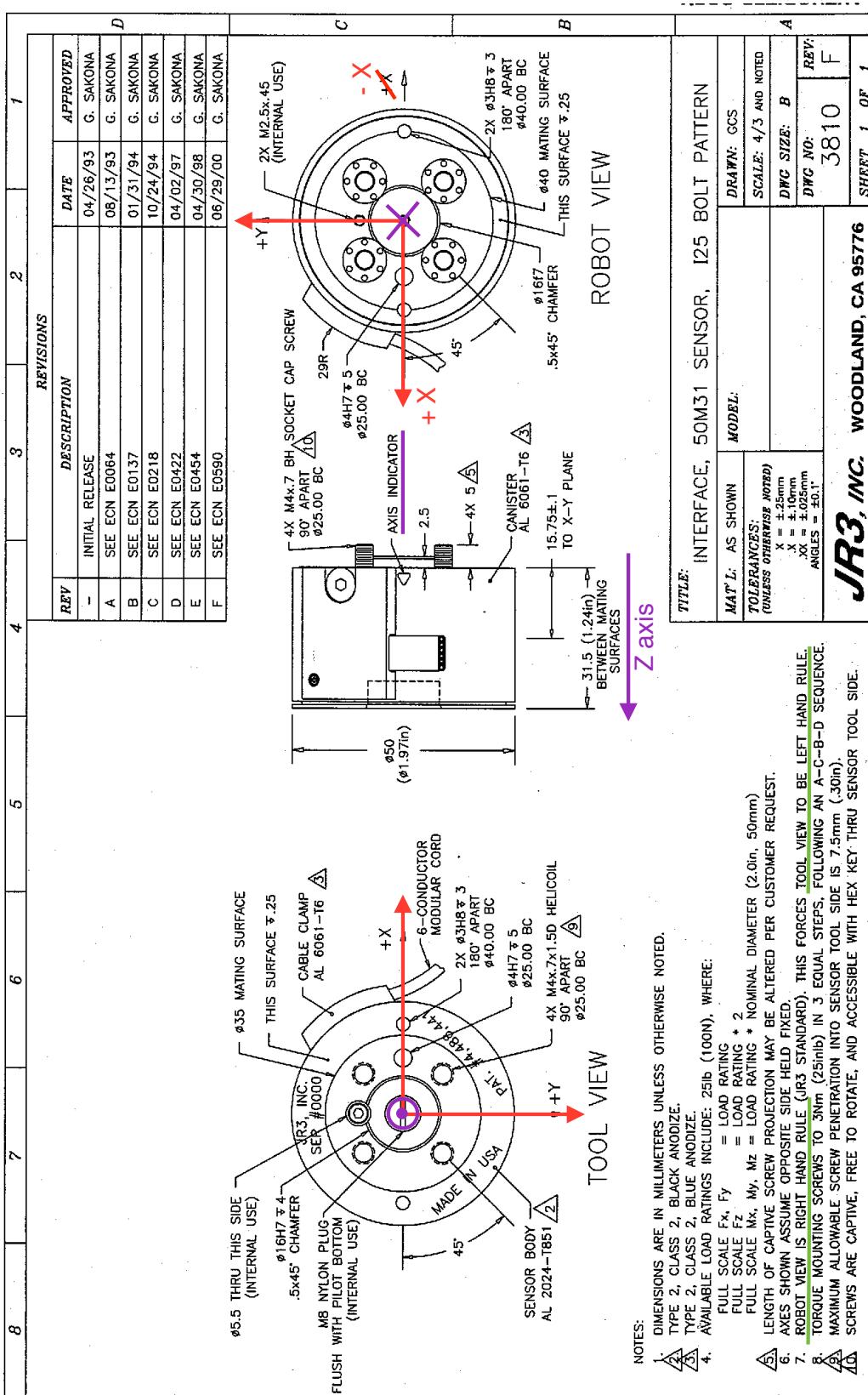
- [31] Wikipedia, "Giróscopo", Wikipedia, 06-10-2018. [En línea]. Disponible en: <https://es.wikipedia.org/wiki/Gir%C3%B3scopo>
- [32] G. Metta, G. Sandini, D. Vernon, L. Natale y F. Nori, "The iCub humanoid robot: an open platform for research in embodied cognition", presentada en Advanced Robotics and its Social Impacts, Half-Moon Bay, CA, USA, 02-04 oct., 2011. [En línea]. Disponible en: <https://ieeexplore.ieee.org/document/6301975/>.
- [33] S. Kajita et al., "Biped Walking Pattern Generation by using Preview Control of Zero-Moment Point", presentada en International Conference on Robotics and Automation, Taipei, Taiwan, 14-19 sep., 2003. [En línea]. Disponible en: <https://ieeexplore.ieee.org/document/1241826/>.
- [34] *Página web de Waseda University*. [En línea]. Disponible en: <http://www.humanoid.waseda.ac.jp/>.
- [35] *Página web de Boston Dynamics*. [En línea]. Disponible en: <https://www.bostondynamics.com/>.
- [36] *Página web de Anybotics*. [En línea]. Disponible en: <https://www.anybotics.com/>.
- [37] *Página web de ABB*. [En línea]. Disponible en: <https://new.abb.com/>.
- [38] S.J. Van Dalen, .<sup>A</sup> Linear Inverted Pendulum Walk Implemented on TULip", Trabajo fin de máster, Dpto. de Ingeniería Mecánica, University of Technology, Eindhoven, Países Bajos, 2012. [En línea]. Disponible en: [https://www.techunited.nl/media/files/humanoid/SwanVanDalen\\_GRAD2012\\_A\\_Linear\\_Inverted\\_Pendulum\\_Walk\\_Implemented\\_on\\_TULip.pdf](https://www.techunited.nl/media/files/humanoid/SwanVanDalen_GRAD2012_A_Linear_Inverted_Pendulum_Walk_Implemented_on_TULip.pdf).
- [39] "Managing Procurement in a Digital World", *Sourcing Guy*, 16-09-2015. [En línea]. Disponible en: <https://sourcingguy.wordpress.com/2015/09/16/managing-procurement-in-a-digital-world/>.



## A. Estructura del robot TEO



## B. Información sensores JR3



## C. Información sensor IMU

The MTi is a miniature size and low weight 3DOF Attitude and Heading Reference System (AHRS). The MTi contains accelerometers, gyroscopes and magnetometers in 3D, and as such is an Inertial Measurement Unit (IMU) as well. Its internal low-power signal processor provides real-time and drift-free 3D orientation as well as calibrated 3D acceleration, 3D rate of turn and 3D earth-magnetic field data.

The MTi is an excellent measurement unit for stabilization and control of cameras, robots, vehicles and other (un)manned equipment.

### Highlights

- Real-time computed attitude/heading and inertial dynamic data
- 360° orientation referenced by gravity and earth magnetic field
- Integrated 3D gyroscopes, accelerometers and magnetometers
- On board DSP with realtime sensor fusion algorithm
- Gyroscopes enable high-frequency orientation tracking
- High update rate (120 Hz), inertial data at max 512 Hz
- Individually calibrated for temperature, 3D misalignment and sensor cross-sensitivity
- Accepts and generates synchronization pulses

### Compact Design

- Compact and robust design
- Easy integration in any system or application (OEM)
- Low weight, ultra-low power consumption

### High Performance

The MTi uses inertial sensors in order to estimate the orientation. The gyroscopes are used to calculate orientation. Integration drift is inevitable whilst using gyroscopes. To compensate for drift completely, the MTi corrects its orientation every sample using the gravity and the earth magnetic field as reference vectors. The sophisticated Xsens sensor fusion algorithm can cope with temporary magnetic disturbances and short-term acceleration, resulting in a reliable orientation estimate. Additionally, the MTi incorporates a magnetic field mapping routine to correct for hard and soft iron effects.

### User scenarios

The MTi is a sensor which can be used in a wide range of applications. To accommodate application specific requirements, the fusion algorithm supports different filter settings and constraints, implemented in scenarios.

### Output

- 3D orientation (360°)
- 3D acceleration
- 3D rate of turn
- 3D magnetic field



## MTi TECHNICAL SPECIFICATIONS

### Attitude and Heading

Static accuracy (roll/pitch)	<0.5 deg
Static accuracy (heading) <sup>1</sup>	<1 deg
Dynamic accuracy <sup>2</sup>	2 deg RMS
Angular resolution <sup>3</sup>	0.05 deg
Dynamic range:	
- Pitch	± 90 deg
- Roll/Heading	± 180 deg
Maximum update rate:	
- Onboard processing	120 Hz
- External processing	512 Hz

### Interfacing

Digital interface	RS-232, RS-485, RS-422 (max 921k6 bps) and USB (ext. converter)
Operating voltage	4,5 - 30V
Power consumption	350 mW
Interface options i/o	SyncOut, AnalogIn, SyncIn (depends on digital interface)

### Maximum operational limits

Ambient temperature	-20...+60 °C
Operating range <sup>4</sup>	

## INDIVIDUAL SENSOR SPECIFICATIONS

### Sensor performance

Dimensions	3 axes
Full Scale (standard)	± 300 deg/s
Linearity	0.1% of FS
Bias stability <sup>5</sup>	1 deg/s
Scale Factor stability <sup>5</sup>	-
Noise	0.05 deg/s/√Hz
Alignment error	0.1 deg
Bandwidth	40 Hz
Max update rate	512 Hz

### Rate of turn

3 axes	± 50 m/s <sup>2</sup>
± 300 deg/s	0.2% of FS
0.1% of FS	0.02 m/s <sup>2</sup>
1 deg/s	0.03%
-	0.002 m/s <sup>2</sup> /√Hz
0.05 deg/s/√Hz	0.1 deg
0.1 deg	30 Hz
40 Hz	512 Hz

### Acceleration

3 axes	± 50 m/s <sup>2</sup>
± 300 deg/s	0.2% of FS
0.2% of FS	0.02 m/s <sup>2</sup>
1 deg/s	0.03%
-	0.002 m/s <sup>2</sup> /√Hz
0.05 deg/s/√Hz	0.1 deg
0.1 deg	30 Hz
40 Hz	512 Hz

### Magnetic field

3 axes	± 750 mGauss
± 300 deg/s	0.2% of FS
0.2% of FS	0.1 mGauss
1 deg/s	0.5%
-	0.5 mGauss
0.05 deg/s/√Hz	0.1 deg
0.1 deg	10 Hz
40 Hz	512 Hz

## HARDWARE SPECIFICATIONS

### Housing

Dimensions (WxLxH)	58x58x22 mm
Weight	50 g

### Options

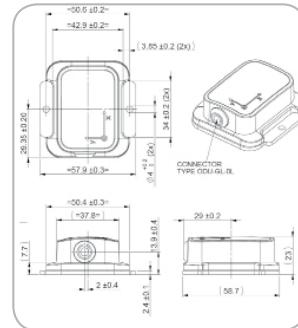
Interface:	28
RS-232	28
RS-485	48
RS-422	68

Full scale acceleration:	A33
1.7g (17 m/s <sup>2</sup> )	A33
5g (50 m/s <sup>2</sup> )	A53
18g (180 m/s <sup>2</sup> )	A83

Full scale rate of turn:	G15
150 deg/s	G15
300 deg/s	G35
1200 deg/s	G25

Product code:  
Standard version:

MTi-## A## G##  
MTi-28 A53 G35



Note: Specifications subject to change without notice

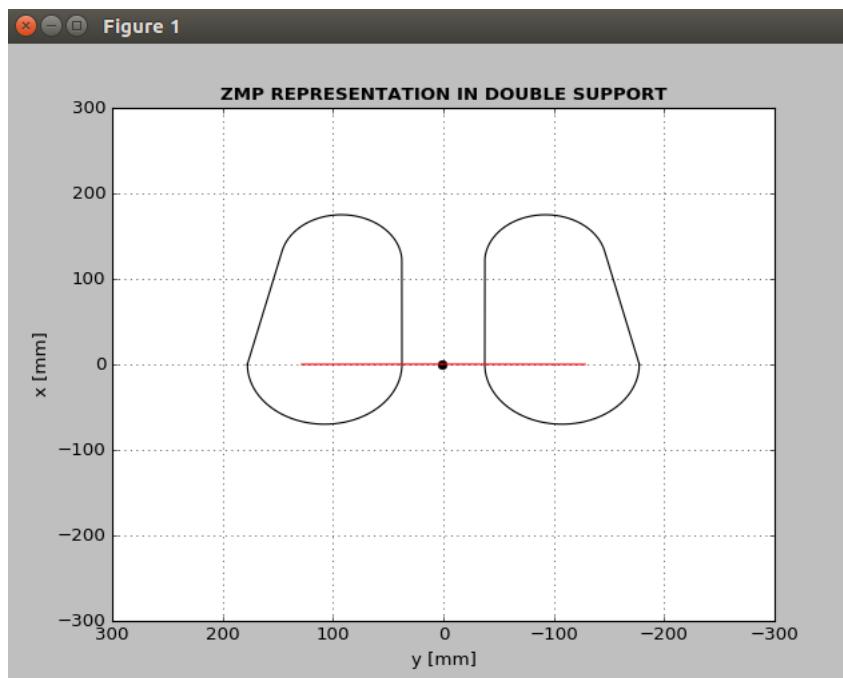
- <sup>1</sup> in homogeneous magnetic environment
- <sup>2</sup> under condition of a stabilized Xsens sensor fusion algorithm
- <sup>3</sup> 1 standard deviation of zero-mean angular random walk
- <sup>4</sup> non-condensing environment
- <sup>5</sup> deviation over operating temperature range 1σ



## D. Interfaz gráfica del ZMP con sensores fuerzapar

Se ha llevado a cabo una representación gráfica del ZMP desarrollado con el modelo de péndulo invertido simple. Ésta ha sido desarrollada en python, ya que es un lenguaje que permite la simplificación del código.

- **Interfaz gráfica**



**Figura D.1:** Interfaz gráfica desarrollada en python

- **Procedimiento para iniciar el programa**

Para iniciar el programa y poder visualizar los datos de manera más gráfica, se tienen que seguir los siguientes pasos:

**Paso 1:** Iniciamos una terminal (TERMINAL 1), y ponemos

- ssh locomotion
- launchLocomotion

**Paso 2:** En otra terminal (TERMINAL 2) tecleamos

- ssh manipulation
- launchManipulation

Y para iniciar los sensores JR3, en esta misma consola teclear

```
- yarpdev -device Jr3 -period 20 -name /jr3 -ports "(ch0:o ch1:o ch2:o  
ch3:o)-channels 24 -ch0:o 0 5 0 5 -ch1:o 6 11 0 5 -ch2:o 12 17 0 5 -ch3:o 18  
23 0 5
```

**Paso 3:** Para iniciar el programa de python, abrir una nueva terminal (TERMINAL 3), situarnos en la carpeta en la que esté el archivo con el código del programa y teclear

```
- python ZMPGraphYarpDoubleSupport.py
```

El código se puede obtener de: <https://github.com/agonzalez92/TFG>

