



BENEMÉRITA UNIVERSIDAD AUTÓNOMA DE PUEBLA

FACULTAD DE CIENCIAS DE LA COMPUTACIÓN
DISEÑO Y SIMULACIÓN DE UN SISTEMA DE CAMINATA PARA TREN
INFERIOR ROBÓTICO

TESIS

PARA LA OBTENCIÓN DEL GRADO DE
LICENCIADO EN INGENIERÍA EN CIENCIAS DE LA COMPUTACIÓN

PRESENTA
NILDA GABRIELA XOLO TLAPANCO

ASESOR
DR. JOSÉ LUIS HERNÁNDEZ AMECA

PUEBLA, PUE.

Noviembre 2021

Tabla de contenido

Dedicatoria	4
Agradecimientos	5
Capítulo 1. Introducción	6
1.1 Antecedentes	6
1.2 Descripción del problema	8
1.3 Objetivos generales y específicos	8
1.4 Hipótesis	8
1.5 Alcances y limitaciones	8
1.6 Justificación	9
1.7 Organización de la tesis	9
Capítulo 2. Marco teórico	11
2.1 Robótica Humanoide	11
2.1.1 Diferencias entre androide y humanoide	11
2.2 Robot caminantes	12
2.2.1 Robot bípedo	12
2.2.1.1 Partes de un robot bípedo	13
2.2.2 Eslabones	13
2.2.3 Articulaciones	14
2.3 Equilibrio estático y dinámico	17
2.4 Cinemática directa	17
2.5 Cinemática inversa	17
2.6 Simulación en Matlab	17
Capítulo 3 Diseño y simulación del Sistema de Caminata	19
3.1 Características ideales del sistema	19
3.2 Formulación de la máquina de estados	19
3.3 Piezas del sistema de caminata	22
3.4 Modelado en Software	26
3.4.1 Bloques requeridos por Simulink	27
3.4.2 Bloques requeridos para el Sistema de Caminata para Tren Inferior Robótico en Simulink	29
4. Experimentación y resultados	32
4.1 Experimentación	32

4.2 Resultados	41
5.- Conclusiones y trabajo a futuro	44
5.1. Conclusiones	44
5.2. Trabajo a futuro	45
Referencias	46

Dedicatoria

A mi familia, mi pareja, mis amigos y para todas aquellas personas con las que he tenido la maravillosa suerte de coincidir en tiempo y lugar.

Agradecimientos

A los profesores y profesoras que me enseñaron y mostraron lo maravilloso que puede ser el mundo de la computación.

A mis compañeros y compañeras con los que compartí clases, carrera y amistad.

Un agradecimiento especial al Dr. José Luis Hernández Ameca que me abrió las puertas a su espacio de trabajo y que me apoyo y guio en este trabajo de tesis.

Capítulo 1. Introducción

1.1 Antecedentes

Los robots bípedos y humanoides son ampliamente investigados para desarrollarse en entornos humanos e interacción con herramientas. Actualmente las instituciones de investigación y universidades nacionales e internacionales están desarrollando investigaciones, definiendo tareas básicas de interacción humano computadora, algoritmos de control, técnicas de caminata, principios matemáticos y generando recurso humano especializado en el área de la robótica.

La funcionalidad principal de los robots es trabajar de manera similar a un sistema humano, basándose en modelos matemáticos simples y complejos que manejen grados de libertad y precisión de movimientos.

El problema de la caminata en los robots bípedos (función indispensable en los humanos) es aún más complejo ya que las piernas soportan el peso del cuerpo al mismo tiempo que realizan movimientos y mantienen el equilibrio. De esta manera los grados de libertad entre cada articulación cambia constantemente.

Tang Zhe aborda el problema de la planificación de la caminata humanoide, definiendo la tarea como complicada debido al alto número de grados de libertad (DOF) y la estructura mecánica variable durante la caminata. En su documento, el método de planificación para movimientos tridimensionales (3-D) se desarrolló en base a un modelo de un robot humanoide típico con 12 DOF en la parte inferior del cuerpo. El proceso de planificación incluye la generación de trayectoria para las articulaciones de cadera, tobillo y rodilla en el espacio cartesiano. El equilibrio del robot se aseguró ajustando el movimiento de la cadera. Los ángulos para cada DOF se obtuvieron del cálculo cinemático tridimensional. El cálculo dio trayectorias de referencia de todos los DOF en el robot humanoide que se usaron para controlar el robot real. Los resultados de la simulación muestran que el método es efectivo [1].

Tzue-Hseng S. Li propone algoritmos en un modelo de péndulo invertido lineal mejorado (LIPM) y una planificación de la marcha. El LIPM es un concepto ampliamente utilizado para la generación de referencia de marcha, y proporciona un modelo simplificado para planificar una trayectoria de centro de masa cuando se le da un punto de momento cero adecuado. Sin embargo, uno de los supuestos de LIPM es que las piernas del robot no tienen masa, por lo que la masa de la pierna de apoyo puede descuidarse para simplificar, y entra en conflicto con las distribuciones de masa de seres humanos y en la mayoría de los robots humanoides. Por lo tanto, este documento propone un LIPM de doble enlace (DLIPM) para eliminar el conflicto sobre la distribución de masa. Además, se propone un algoritmo de planificación de la marcha para generación de referencia para caminata natural. En los resultados de la simulación, el método propuesto se implementa basado en un modelo de un robot humanoide del tamaño de un adolescente llamado David Junior. Los resultados de la simulación validan la

viabilidad y practicabilidad del método propuesto. Además, las comparaciones entre LIPM convencionales y DLIPM demuestran el rendimiento del método DLIPM propuesto [2].

Galloway y Koushil presentan un método novedoso para abordar la saturación del actuador para el sistema del híbrido no lineal incorporando directamente límites de entrada definidos por el usuario en un diseño de controlador. En particular, se considera la aplicación de la caminata bípeda y demostrar que el método [basado en una programación cuadrática (QP) más la implementación de un controlador basado en la función de control Lyapunov (CLF)] permite un rendimiento gradual de degradación mientras continúa caminando bajo límites de entrada cada vez más estrictos. Basado en trabajo previo, que ha demostrado la efectividad de los controladores basados en CLF para estabilizar la marcha periódica de caminantes bípedos [3].

Zhe Xu y Junyao Gao proponen una extensión del modelo bípedo de masa-resorte con longitud floja variable y rigidez. Dado que el modelo convencional bípedo de masa-resorte tiene problemas para pisar y bajar la velocidad al caminar, se resuelve ajustando la longitud floja de la pierna del resorte. Por otra parte, para producir golpe de talón y golpe del dedo del pie en marcha, la rigidez del resorte se cambia para conseguir correr con diferentes velocidades. Además, para imitar el comportamiento de balanceo del pie humano, se sincroniza el movimiento del centro de presión con el del centro de masa. Con base en los métodos propuestos, las simulaciones de un robot de tamaño humano logran caminar, y al correr golpear el talón y el dedo del pie. Caminar a una velocidad de 1 m / s puede generar la fuerza de reacción del suelo de doble joroba. Se logran pasos de carrera de golpe de talón y de dedo del pie a una velocidad de 4.5 m / s. Las frecuencias de dos de los pasos de marcha son consistentes con los de los datos de captura de movimiento humano. Articula posiciones angulares de caminar y la ejecución del talón son similares a los de los datos humanos. Por lo tanto, el modelo propuesto resulta ser eficaz para generar comportamientos humanos similares al caminar y correr [4].

Jin-Ling Lin y Kao-Shing Hwang presentan un método para la caminata dinámica bípeda y el control del equilibrio usando aprendizaje de refuerzo, que aprende la caminata dinámica sin un conocimiento a priori sobre el modelo dinámico. La arquitectura de aprendizaje desarrollada tiene como objetivo resolver problemas de control complejos en el control de actuación robótica mapeando el espacio de acción de un dominio discretizado a uno continuo. Emplea las acciones discretas para construir una política de acción continua. La arquitectura permite escalar la dimensionalidad del espacio de estado y la cardinalidad del conjunto de acciones que representa nuevos conocimientos o requisitos para una tarea deseada. El método de aprendizaje de equilibrio utiliza el movimiento del brazo y la pierna del robot para cambiar el momento cero en las suelas, para poder mantener al robot bípedo en un estado estable estático. Este algoritmo de equilibrio es aplicado a bípedos para caminar en una superficie plana e inclinada, logrando que los bípedos sean más estables. La simulación muestra que el método propuesto puede permitir que el robot aprenda a mejorar su comportamiento en términos de velocidad al

caminar. Finalmente, los métodos se implementan en un robot bípedo físico para demostrar la viabilidad y efectividad del esquema de aprendizaje propuesto [5].

1.2 Descripción del problema

Los robots humanoides son desarrollados con la idea de parecerse a los seres humanos, pueden tener fines funcionales en el entorno humano o interacción con herramientas, pero son mayormente usados en el área de la investigación con la finalidad de desarrollar sistemas que funcionen de manera similar a nuestro organismo, permitiéndonos un mayor entendimiento de nuestra motricidad.

En el presente trabajo se aborda el problema de diseñar y simular un sistema de caminata, para el desplazamiento de un robot bípedo, basándose en las investigaciones reportadas en bases de datos fidedignas, artículos de investigación y desarrollo científico.

1.3 Objetivos generales y específicos

- Objetivo general

Diseñar y simular un sistema de caminata para un robot bípedo, aplicando las técnicas de cinemática inversa y directa.

- Objetivos específicos
 1. Buscar información
 2. Diseñar el sistema de caminata para un robot bípedo (cinemática inversa y directa)
 3. Simular el sistema de caminata
 4. Recopilar y analizar los resultados
 5. Mostrar resultados, conclusiones y trabajo a futuro

1.4 Hipótesis

La técnica de la cinemática permite diseñar y simular la caminata de un robot bípedo.

1.5 Alcances y limitaciones

Alcances:

En el presente trabajo se diseñó y simuló un sistema de caminata para un tren inferior dinámico, basándose en herramientas de software como Matlab, Simulink,

Simscape y Solidworks, partiendo de un diseño y piezas de Onshape, plataforma de diseño asistido por computadora que permite el acceso a sus diseños a estudiantes y profesores, logrando enlazar y permitiendo que las piezas trabajen en conjunto con herramientas proporcionadas por Simulink y obteniendo el video final de la simulación.

Limitaciones:

El sistema de simulación obtenido no logró probarse de manera física debido a la incapacidad para acceder a una impresora 3D para obtener las piezas necesarias, además del material necesario para implementar las conclusiones del sistema, esto debido a las complicaciones derivadas de la pandemia del COVID-19 que inicio al poner en marcha la elaboración de la presente tesis. Además de no poder unir el sistema con los elementos necesarios para crear el robot social que se deseaba realizar.

1.6 Justificación

La presente investigación se realiza principalmente para la aplicación de los conocimientos adquiridos durante la formación universitaria, que nos permite contribuir en un trabajo colaborativo para el desarrollo de un robot social, con la finalidad de incentivar a los alumnos al desarrollo de nuevas tecnologías, que permitan el uso de la robótica social en áreas de investigación y aplicación como la salud, entretenimiento, ingeniería, psicología, educación y domótica, ya que es un tema que ha cobrado relevancia en los últimos años y que aún falta por conocer su máximo potencial.

1.7 Organización de la tesis

El presente trabajo está dividido en 5 capítulos. En el primer capítulo se presentan los antecedentes, la descripción del problema, los objetivos generales y específicos, la hipótesis, los alcances y limitaciones que se presentaron a lo largo de la elaboración y la justificación para la elaboración de la simulación de un sistema de caminata para un tren inferior robótico.

El segundo capítulo presenta el marco teórico del trabajo, es decir las definiciones y detalles que debemos de conocer para diseñar y simular el sistema de caminata, elementos que se trabajan desde el inicio y hasta el final de su desarrollo y sin los cuales no se entendería el sistema en su totalidad.

El tercer capítulo establece el diseño que se seguirá para la simulación, partiendo de las características ideales del sistema, la formulación de la máquina de estados, la descripción de las piezas del sistema, los elementos básicos de la simulación y los elementos necesarios para el Sistema de caminata para tren inferior robótico.

El cuarto capítulo se enfoca en recopilar y analizar los resultados obtenidos partiendo del sistema diseñado en el capítulo anterior.

Para terminar, contamos con el capítulo cinco, en donde se muestran las conclusiones y trabajo a futuro.

Capítulo 2. Marco teórico

2.1 Robótica Humanoide

También conocidos como androides, los cuales son máquinas antropomórficas capaces de imitar las funciones básicas del ser humano tales como caminar, hablar, ver, recolectar, limpiar y trasladar objetos. Son sistemas muy complejos no solo en su estructura mecánica, también lo son relacionado con la electrónica, los sensores, los servomotores, la programación y el modelado matemático. Los robots humanoides ocupan una posición muy importante en todo el mundo debido a su versatilidad y aplicaciones; un aspecto clave para tal aceptación en la sociedad es la similitud al ser humano que los hace ver agradables, confiables y amigables. [16]

Los robots humanoides han tenido la atención de múltiples investigaciones en los pasados 35 años. Es actualmente uno de los temas más emocionantes en el área de la robótica, y existen muchos proyectos en este campo. [17]

Los robots humanoides son máquinas, pero tienen la forma o las funciones de los seres humanos, pueden construirse con dos o múltiples piernas para añadir una mejor movilidad. La locomoción bípeda es una parte integral de ellos. Cabeza, torso, brazos, manos y piernas son algunas anatomías añadidas a dichos robots. Los robots humanoides pueden caminar y subir escaleras, son usables para el espacio, servicio doméstico, seguridad, medicina, servicio para discapacitados y manufactura.[18]

2.1.1 Diferencias entre androide y humanoide

Androides y robots humanoides están modelados específicamente a partir de la forma humana. En uso actual un androide es un robot diseñado para lucir como un humano tanto como sea posible, incluyendo orejas, cabello e incluso una boca articulada. Un robot humanoide es uno de los que comparte la arquitectura básica de un humano, pero no significa que sea una copia fisiológica.

El término de androide y humanoide son usados para describir a robots que son modelos a partir de la forma humana. Androide es un robot diseñado para parecer humano tanto como sea posible, incluso masculino, femenino o una mezcla de los dos: andrógino (androginia es de donde vienen el término androide, que significa “como hombre y mujer”).

Por el contrario -y algo confuso- un robot humanoide comparte la arquitectura básica de un humano. Tienen dos piernas, con una cabeza en la cima y dos brazos a los lados. Pero raramente duplican la apariencia de una persona, se trata más de

replicar la capacidad humana, como poder caminar por un pasillo o sentarse en una silla destinada a los humanos. [19]

2.2 Robot caminantes

Un robot caminante usa piernas para moverse. La mayoría de los caminantes tienen seis piernas, como un insecto, ya que proveen un excelente soporte y balance. Sin embargo, los que ganan en popularidad son los que caminan en dos y cuatro piernas.

Los robots caminantes requieren de una gran precisión para construirlos. Se clasifican según su número de piernas:

Bípedos: que caminan sobre dos piernas. En un verdadero robot bípedo, una pierna se levanta para dar un paso. El balance puede ser difícil cuando el robot solo tiene una pierna en el piso, por lo que este tipo de robot está entre los más difíciles de dominar.

Cuadrúpedos: significan cuatro piernas. Los robots con cuatro piernas más sofisticados demuestran una variedad de estilos al caminar, imitando algunos animales. Estos pueden ser difíciles de construir, ya que cada pierna necesita tres articulaciones separadas -conocidos también como grados de libertad o DOF- para mantener el balance y hacer giros.

Hexápodos: con seis piernas provee un excelente balance y movilidad. Entre los robots caminantes, son los más comunes y a pesar de las piernas extras, son los más fáciles de construir.

Octópodos: son los arácnidos en el mundo de los robots, ya que cuentan con ocho piernas. En muchos diseños predomina la caminata con seis piernas, usando las dos piernas adicionales como un tipo de manipulador. [19]

2.2.1 Robot bípedo

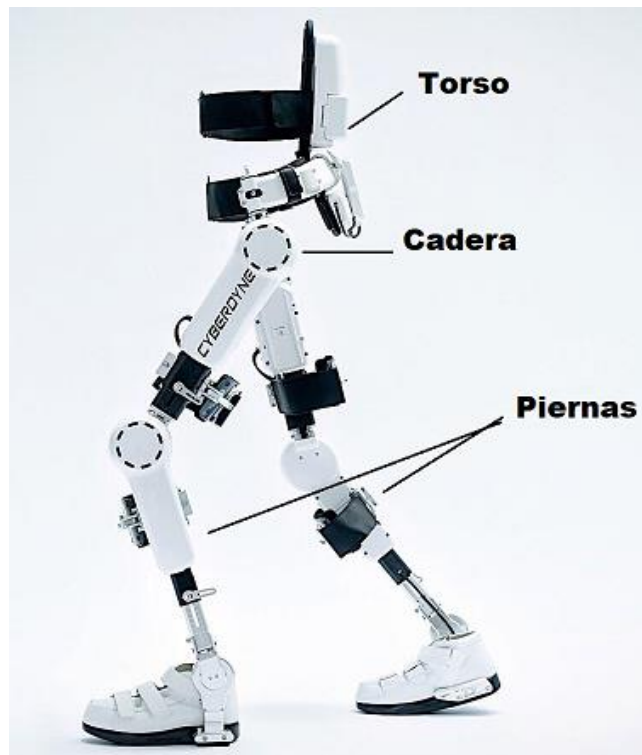
Son una clase particular de robots caminantes, que se caracterizan por su movimiento en dos piernas. La cinemática y dinámica de este tipo de robots es compleja, y la síntesis de eficiencia y la robustez de sus controles para lograr una caminata estable es una tarea desafiante. [15]

Los robots bípedos podrán operar en entornos humanos con un grado mucho mayor de eficiencia que cualquier otro tipo de robot ya ideado. Debido a esto eventualmente los robots bípedos pueden ser usados para completar tareas difíciles y peligrosas para los humanos. Esto incluye muchas aplicaciones como trabajo en entornos con condiciones extremas (como una operación de rescate en fuego), con gases tóxicos o químicos, con explosivos (como minas terrestres) o como ayudante

para los humanos en situaciones similares. También como un subproducto útil de la investigación en robots bípedos puede mejorar los dispositivos protésicos. [16]

2.2.1.1 Partes de un robot bípedo

Un robot bípedo es una cadena cinemática abierta que consiste en dos subcadenas llamadas piernas, una subcadena llamada torso, todas unidas a un mismo punto llamada cadera. Una o ambas piernas pueden estar en contacto con el piso. Cuando sólo una pierna está en contacto con el piso, esa pierna se le conoce como pierna de apoyo y a la otra se le conoce como pierna de columpiado o de movimiento. [17]



*Ilustración 1:Partes de un robot bípedo, recuperado de:
Maldonado, D. (2018). SEQ ARABIC [Ilustración]. Las piernas
robóticas de Cyberdyne ya son una realidad.
<http://www.icorp.com.mx/blog/piernas-roboticas-de-cyberdyne/>*

2.2.2 Eslabones

Eslabón: Son elementos mecánicos que constituyen la estructura del robot. Por medio de las articulaciones o uniones los eslabones pueden enlazarse entre sí para construir la cadena cinemática del robot. [18]

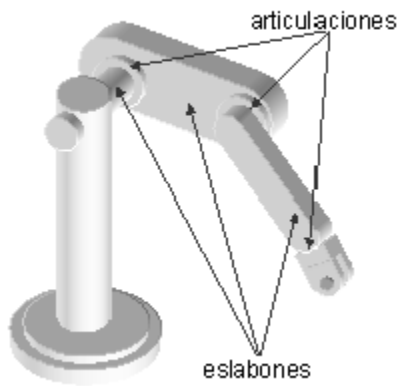


Ilustración 2: Ejemplo de eslabones en un brazo robótico

2.2.3 Articulaciones

Conexión de dos cuerpos rígidos caracterizados por el movimiento de un sólido sobre otro. [19]

Articulación: Son las uniones formadas por servomotores que permiten la conexión y movimiento relativo entre dos eslabones consecutivos del robot, dependiendo del tipo de movimiento que realicen pueden ser de tipo rotacional o lineal, las articulaciones de tipo rotacional normalmente tienen sus unidades de medición en radianes o en grados mientras que las articulaciones lineales presentan sus medidas en metros. [16]

Articulación tipo esférica: Permite la rotación libre del eslabón que se inserta, al permitir solo movimiento rotacional se restringen todos los movimientos traslacionales, por lo que dicha articulación aporta 3-DOF uno por cada eje coordinado: x, y, z.

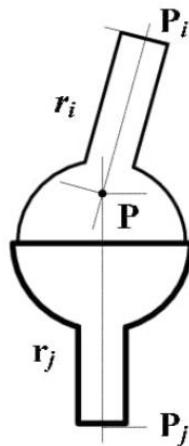


Ilustración 3: Articulación de tipo esférica, recuperado de: Pérez M., Cuevas E., Zaldívar D. (2014). Cuatro articulaciones muy comunes en sistemas mecatrónicos. [Figura] FUNDAMENTOS DE ROBÓTICA Y MECATRÓNICA CON MATLAB® Y SIMULINK. Madrid, España: RA-MA. P. 119.

Articulación cilíndrica: permite dos movimientos, un desplazamiento prismático en la dirección z_j y la rotación alrededor del eje z_i sobre el punto P_i . En esta articulación, el desplazamiento se realiza sobre el eje z_j , mientras que la rotación lo hace sobre el eje z_i , que evidentemente son colineales, su producto cruz es cero por lo que se puede deducir que este tipo de articulación solo provee 2-DOF ya que los movimientos de rotación en los ejes x y y se encuentran limitados.

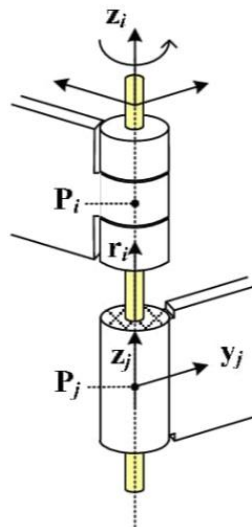


Ilustración 4: Articulación de tipo cilíndrica, recuperado de: Pérez M., Cuevas E., Zaldívar D. (2014). Cuatro articulaciones muy comunes en sistemas mecatrónicos. [Figura] FUNDAMENTOS DE ROBÓTICA Y MECATRÓNICA CON MATLAB® Y SIMULINK. Madrid, España: RA-MA. P. 119.

Articulación prismática: Permite la realización de movimientos de traslación en la dirección del eje z exclusivamente por lo que esta articulación sólo nos provee 1-DOF.

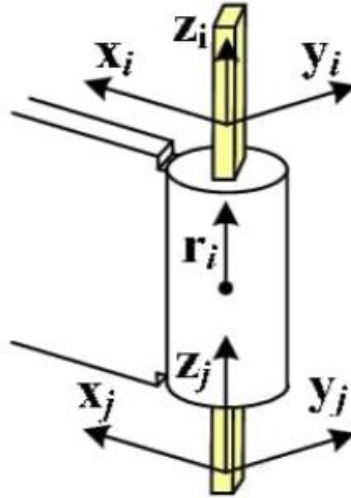


Ilustración 5: Articulación de tipo prismática, recuperado de: Pérez M., Cuevas E., Zaldívar D. (2014). Cuatro articulaciones muy comunes en sistemas mecatrónicos. [Figura] FUNDAMENTOS DE ROBÓTICA Y MECATRÓNICA CON MATLAB® Y SIMULINK. Madrid, España: RA-MA. P. 119.

Articulación rotacional: Este tipo de articulación cancela todo tipo de desplazamiento pues solo permite un giro solo se permite la rotación efectuada en el eje z por lo que nuevamente esta articulación sólo puede proveer 1-DOF. [15]

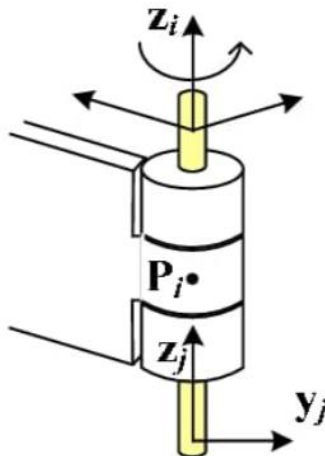


Ilustración 6: Articulación rotacional, recuperado de: Pérez M., Cuevas E., Zaldívar D. (2014). Cuatro articulaciones muy comunes en sistemas mecatrónicos. [Figura] FUNDAMENTOS DE ROBÓTICA Y MECATRÓNICA CON MATLAB® Y SIMULINK. Madrid, España: RA-MA. P. 119.

2.3 Equilibrio estático y dinámico

Existen dos modelos básicos de equilibrio (balanceo) en la locomoción bípeda: el “equilibrio estático” y el “equilibrio dinámico”. El equilibrio estático lleva a movimientos lentos y menos naturales, mientras que para ganar eficiencia y velocidad se tiene que conseguir un equilibrio dinámico. Los dos tipos de equilibrio se basan en la implementación de leyes de control que utilizan el concepto de área de soporte. [16]

2.4 Cinemática directa

Consiste en determinar cuál es la posición y orientación del extremo final del robot, con respecto a un sistema de coordenadas que se toma como referencia, conociendo los valores de las articulaciones y los parámetros geométricos de los elementos del robot.

Se utiliza fundamentalmente el álgebra vectorial y matricial para representar y describir la localización de un objeto en el espacio tridimensional con respecto a un sistema de referencia fijo.

Dado que un robot se puede considerar como una cadena cinemática formada por objetos rígidos o eslabones unidos entre sí, mediante articulaciones, se puede establecer un sistema de referencia fijo situado en la base del robot y describir la localización de cada uno de los eslabones con respecto a dicho sistema de referencia. [19]

2.5 Cinemática inversa

Consiste en determinar el sistema de coordenadas que deben de tomar las articulaciones y parámetros geométricos de los elementos del robot para llegar a una posición y orientación determinada. [19]

2.6 Simulación en Matlab

La simulación es una técnica numérica para conducir experimentos en una computadora digital. Estos experimentos comprenden ciertos tipos de relaciones matemáticas y lógicas, las cuales son necesarias para describir el comportamiento y la estructura de sistemas complejos del mundo real a través de largos periodos de tiempo. [17]

La simulación digital es una técnica que permite imitar o simular en un ordenador el comportamiento de un sistema real o hipotético según ciertas condiciones particulares de operación. Para analizar, estudiar y mejorar el comportamiento de

un sistema mediante las técnicas de simulación digital, es necesario primero describir bajo un cierto formalismo el conocimiento que se tiene sobre las dinámicas de interés (modelo conceptual), y luego codificar en un entorno de simulación para realizar experimentos y analizar resultados. [18]

Matlab es un lenguaje de programación de alto nivel orientado al cálculo técnico que integra un entorno amigable para el cálculo, la visualización de resultados y la codificación de programas. Generalmente es utilizado en:

- Cálculo y matemática
- Desarrollo de algoritmos
- Adquisición de datos
- Modelado, simulación y prototipado
- Análisis, exploración y visualización de datos
- Gráficos científicos y de ingeniería
- Desarrollo de aplicaciones de interfaces gráficas

Dentro de la familia de productos se encuentra Simulink, que es un entorno de diagramas de bloques extensibles para la simulación de sistemas y el diseño basado en modelos. Permite a los ingenieros simular y analizar una amplia gama de sistemas que incluyen controles, señales y procesamiento de imágenes, comunicaciones y sistemas físicos multidominio. [19]

Capítulo 3 Diseño y simulación del Sistema de Caminata

3.1 Características ideales del sistema

El desarrollo de la simulación del tren dinámico que se propone realizar en el presente documento requiere lograr una caminata continua manteniendo el equilibrio del sistema, sistema que pretende ser parte de un trabajo conjunto para desarrollar como primera instancia el tren inferior dinámico real y a una escala mayor un robot humanoide funcional.

3.2 Formulación de la máquina de estados

Para el diseño del sistema se necesitó de la aplicación de la cinemática directa, es decir un proceso a partir de los valores específicos (parámetros) para calcular la posición final de los elementos.

Los elementos provienen del modelo importado, en primera instancia en la tabla 1 se enlistan los parámetros iniciales del sistema:

Tabla 1: Parámetros iniciales del sistema

Piezas	Cantidades
Pierna	2
Pie	2
Cadera	2
Cintura	2
Tronco	1

Articulaciones rotacionales	6
-----------------------------	---

Una vez que se han descrito los elementos se necesita la descripción de su funcionamiento, comportamiento que empieza con un paso de la pierna derecha y utilizando la pierna izquierda como apoyo, acompañado de un movimiento del tobillo derecho; una vez que un paso ha sido dado se realiza el mismo procedimiento en forma contraria, es decir la pierna izquierda es la que da el paso mientras que la pierna derecha funge como apoyo mientras el tobillo izquierdo realiza su movimiento, siendo la caminata una sucesión finita de ambos, descrita en la tabla 2.

Tabla 2: Transición de estados

Tiempo (s)	Grados (°)					
	Rodilla		Tobillos		Cadera	
	Derecha	Izquierda	Derecho	Izquierdo	Derecha	Izquierda
0	0	0	0	0	0	0
0.5	0	0	-30	0	0	0
1	45	0	-30	0	-45	0
1.5	0	0	0	0	0	0
2	0	0	0	0	0	0
2.5	0	0	0	-30	0	0
3	0	-45	0	-30	0	45
3.5	0	0	0	0	0	0
4	0	0	0	0	0	0
4.5	0	0	-30	0	0	0
5	45	0	-30	0	-45	0
5.5	0	0	0	0	0	0
6	0	0	0	0	0	0
6.5	0	0	0	-30	0	0

7	0	-45	0	-30	0	45
7.5	0	0	0	0	0	0
8	0	0	0	0	0	0
8.5	0	0	-30	0	0	0
9	45	0	-30	0	-45	0
9.5	0	0	0	0	0	0
10	0	0	0	0	0	0

Al observar la tabla podemos omitir las repeticiones de cada paso para crear una tabla de estados, tomando cada pieza como un bit en la que la posición de 0° representa al 0 y un número diferente de grados (-30, 45 y -45) representa el 1, creando 5 estados diferentes entre sí, como puede observarse en la tabla 3.

Tabla 3: Descripción de estados

Estado	Descripción	Acción	Representación en bits
S0	Todas las piezas en 0°	Reposo	000000
S1	Tobillo derecho en -30° y el resto de las piezas en 0°	Inicio del paso con el pie derecho	001000
S2	Rodilla derecha en 45°, tobillo derecho en -30°, cadera derecha en -45° y el resto de las piezas en 0°	Término del paso con el pie derecho	101010
S3	Tobillo izquierdo en -30° y el resto de las piezas en 0°	Inicio del paso con el pie izquierdo	000100
S4	Rodilla izquierda en -45°, tobillo izquierdo en -30°, cadera izquierda en -45° y el resto de las piezas en 0°	Término del paso con el pie izquierdo	010101

Ya que se ha analizado la tabla se puede deducir el diagrama de estados con las transiciones necesarias para pasar de un estado a otro, como puede observarse en la ilustración 7.

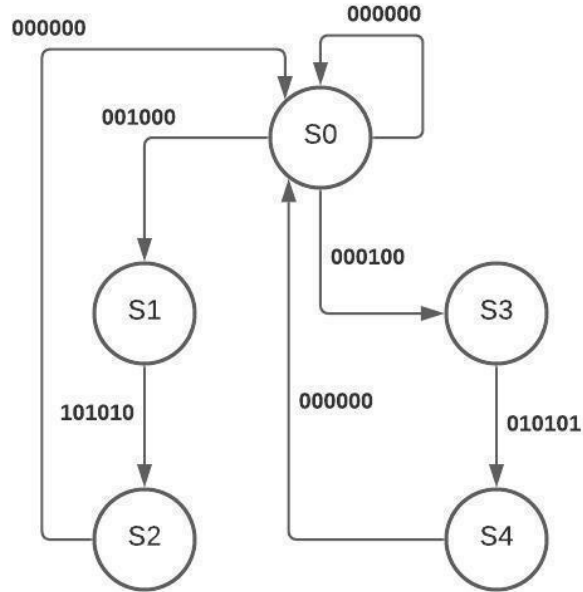


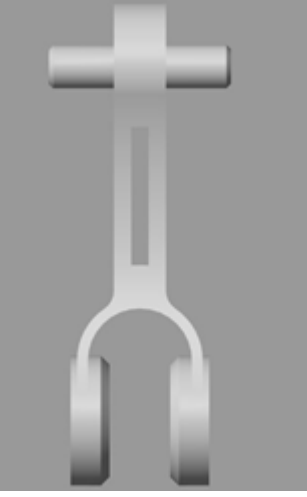


Ilustración 7: Diagrama de transición de estados

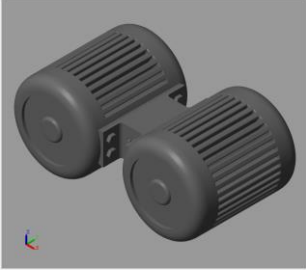

3.3 Piezas del sistema de caminata

Las piezas exportadas y sus características se enlistan en la tabla 4.

Tabla 4: Descripción de las piezas utilizadas

Pieza	Representación 3D	Ancho (mm)	Alto (mm)	Largo - Fondo (mm)	Masa (Kg)
Pie		54.42	111.63	50	0.126

Pierna inferior		53.34	143.88	38	0.161
Pierna superior		50.8	140	38	0.176
Cadera		38	62.43	32.66	0.08500 000000 000001

Cintura		94.1	39.6 5	54.22	0.311
Tronco		123.88	189	50	0.369

Para obtener parámetros como altura, anchura y profundidad se utilizó SolidWorks, como puede observarse en la ilustración 8, siendo este un software de Diseño Asistido por Computadora (CAD) para modelado mecánico en 2D y 3D, desarrollado en la actualidad por SolidWorks Corp., una filial de Dassault Systèmes, S.A. (Suresnes, Francia), para el sistema operativo Microsoft Windows.

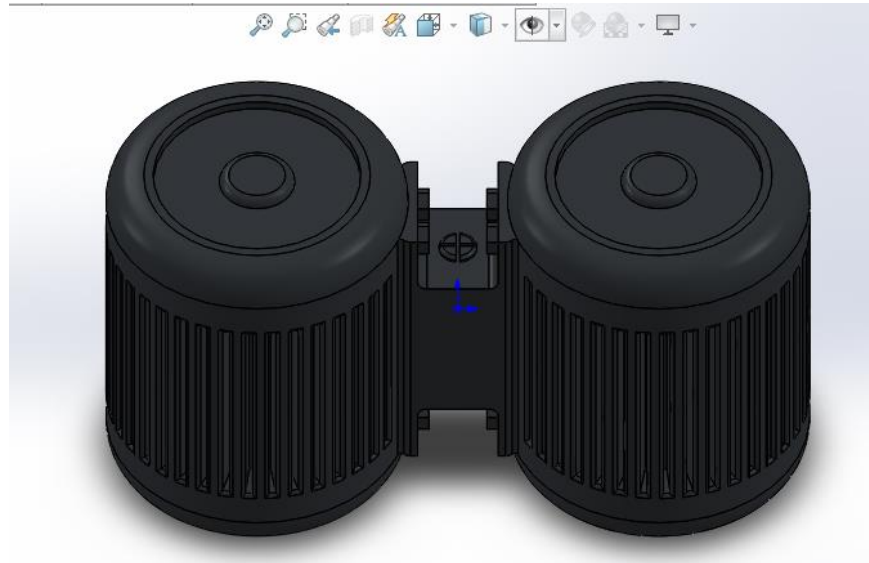


Ilustración 8: Vista isométrica de la cintura desde SolidWorks

Se muestran la vista frontal, izquierda e isométrica del sistema de caminata completo en las figuras 9 a 11.

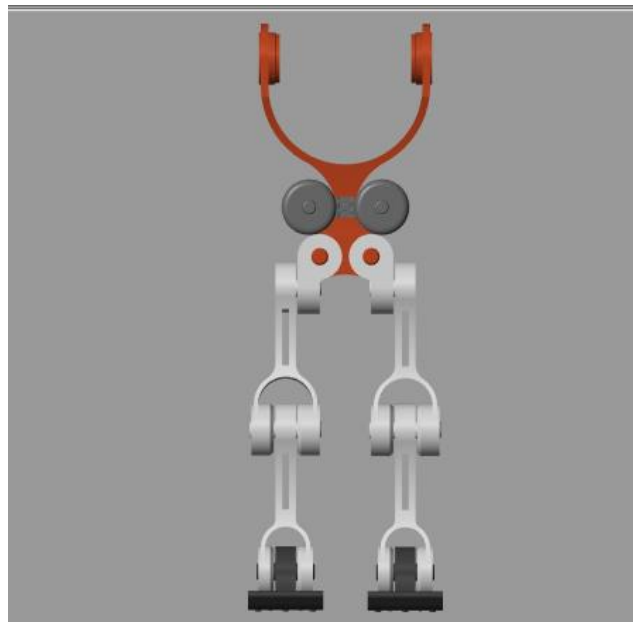


Ilustración 9: Vista frontal del modelo



Ilustración 10: Vista izquierda del modelo

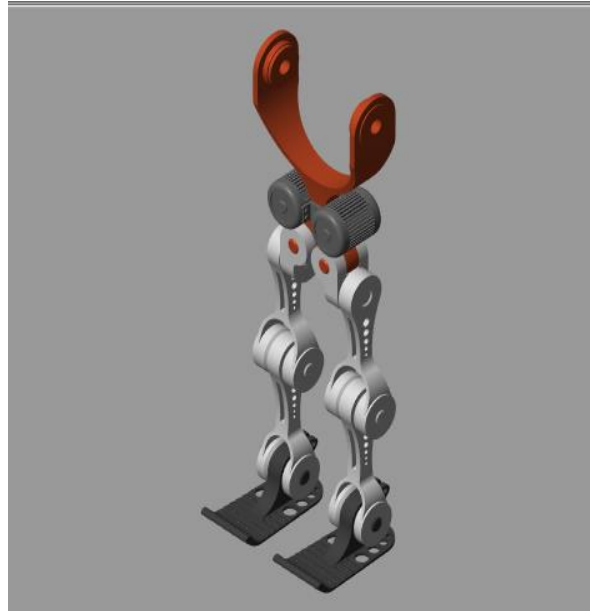


Ilustración 11: Vista isométrica del modelo

3.4 Modelado en Software

Para la realización del modelo se requirió de la instalación del software Matlab en su versión de 2020b, en una computadora personal con los requerimientos siguientes:

Lista de Requerimientos Matlab 2020b

- Sistemas operativos:
- Windows 10 (versión 1803 o superior)
- Paquete de servicio 1 de Windows 7
- Windows Server 2019
- Windows Server 2016
- Procesadores
- Procesador Intel o AMD x86-64
- Disco duro de 3.5 GB solo para MATLAB, 5-8 GB para una instalación típica (se recomienda un disco de estado sólido (SSD))
- Mínimo: 4 GB RAM preferiblemente 8 GB en RAM
- Para Polyspace, se recomiendan 4 GB por núcleo

Recomendaciones

La instalación completa de todos los productos MathWorks puede ocupar hasta 32 GB de espacio en disco. Una tarjeta gráfica acelerada por hardware compatible con OpenGL 3.3 con 1 GB de memoria GPU. Cualquier procesador Intel o AMD x86-64

con cuatro núcleos lógicos y compatibilidad con el conjunto de instrucciones AVX2. La aceleración de la GPU con Parallel Computing Toolbox requiere una GPU que tenga una capacidad informática 3.0 o superior.

Se realizó un diagrama que muestra los pasos que se siguieron para realizar el modelo, como se muestra en la ilustración 12.

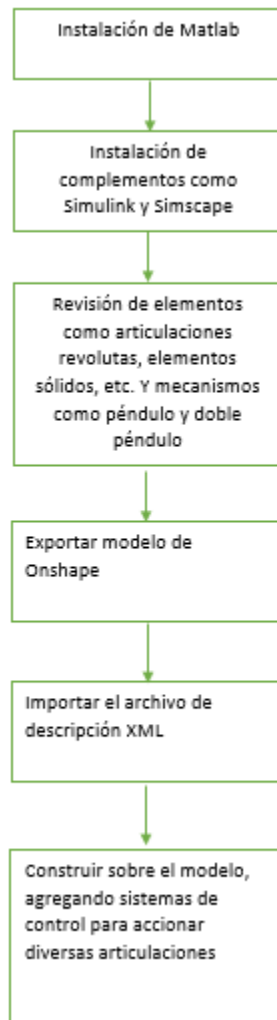


Ilustración 12: Instalación del software necesario para el modelo

3.4.1 Bloques requeridos por Simulink

Simulink es un toolbox (conjunto de funciones que provee algoritmos para solucionar diferentes tipos de problemas, además posee una interfaz gráfica amigable con usuarios menos familiarizados con la programación [20]) que ofrece un editor gráfico, bibliotecas de bloques personalizables y un conjunto de solvers,

para modelar y simular sistemas dinámicos. Está basado en un entorno de bloque multidominio bajo un diseño basado en modelos que permite el diseño y verificación continua de los sistemas embebidos.

Los diagramas de bloques es un intento de entender el comportamiento del sistema por medio de una representación gráfica, que esencialmente consiste de representaciones de los componentes individuales del sistema (bloques) junto con un flujo de señales entre estos componentes. Simulink se basa en la representación por diagramas de bloques a través de una interfaz gráfica para convertir un diagrama casi directamente en un modelo Simulink y luego simular el funcionamiento del sistema, con el fin de analizar el comportamiento del sistema con diferentes condiciones de entrada [19].

Para realizar un sistema mecánico nuevo, se requieren tres bloques iniciales, los cuales son:

World Frame: Proporciona acceso al world frame o ground frame, un marco de coordenadas ortogonal, derecho e inmóvil único predefinido en cualquier modelo mecánico. World frame es la base de todas las redes de marcos en un modelo mecánico.



Ilustración 13: Bloque World Frame

Configuración mecánica: Establece parámetros mecánicos y de simulación que se aplican a una máquina completa, la máquina de destino a la que está conectado el bloque. Puede especificar la gravedad uniforme para todo el mecanismo y también establecer el delta de linealización. El delta de linealización especifica el valor de perturbación que se utiliza para calcular las derivadas parciales numéricas para la linealización.



Ilustración 14: Bloque Configuración mecánica

Configuración del solucionador: Define la configuración del solucionador que se utilizará para la simulación.

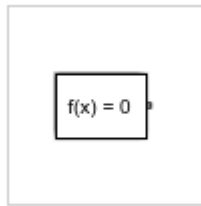


Ilustración 15: Bloque Configuración del solucionador

3.4.2 Bloques requeridos para el Sistema de Caminata para Tren Inferior Robótico en Simulink

Para simular el sistema de caminata, se requiere que las piezas sean exportadas al diagrama de bloques y que cuenten con capacidades específicas como posición en el espacio y grados de libertad. Para lo cual se requieren los siguientes bloques:

Articulación de soldadura: representa una articulación de soldadura entre dos bloques. Esta articulación tiene cero grados de libertad. El bloque base y secundario siempre coinciden.



Ilustración 16: Bloque Articulación de soldadura

Articulación rotacional: representa una articulación rotacional que actúa entre dos bloques. Esta articulación tiene un grado de libertad rotacional. La articulación restringe los orígenes de los dos bloques para que sean coincidentes y los ejes z de los bloques base y secundario coincidan, mientras que el eje x y el eje y del secundario pueden girar alrededor del eje z.

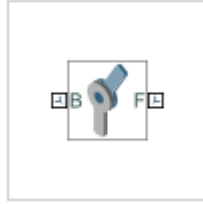


Ilustración 17: Bloque Articulación rotacional

Transformación rígida: define una transformación rígida 3-D fija entre dos bloques. Dos componentes especifican de forma independiente las partes de traslación y rotación de la transformación. Se pueden combinar libremente diferentes traslaciones y rotaciones.

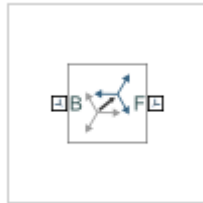


Ilustración 18: Bloque Transformación rígida

Sólido: representa un sólido con geometría, inercia y color. Es una forma prismática con centro geometría coincidente con el origen del bloque como referencia y superficies prismáticas normales a los ejes x, y y z del bloque de referencia. Este bloque puede hacer referencia a una pieza creada por diferentes softwares de modelado 3D como: CATIA, NX, SolidEdge y otros formatos. El bloque obtiene la inercia de la geometría y la densidad, de la geometría y la masa, o de un tensor de inercia que especifique.



Ilustración 19: Bloque Sólido

Inercia: agrega inercia y masa a un modelo mecánico especificando la masa total, la ubicación del centro de masa, los momentos de inercia y los productos de inercia. Si el tercero y el cuarto son cero, la inercia representa una masa puntual trasladada. Si el segundo también es cero, la inercia representa una masa puntual en el origen.

Punto de inercia de masa representa una masa concentrada en un solo punto. La inercia personalizada representa una distribución de masa general relativa a un bloque de referencia.



Ilustración 20: Bloque Inercia

Constructor de señal: Crea y genera grupos de señales intercambiables de fuentes de señales lineales por partes y usarlas en un modelo.

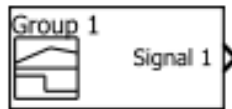


Ilustración 21: Bloque Constructor de señal

4. Experimentación y resultados

4.1 Experimentación

La experimentación consistió en cuatro pasos que se describen a continuación:

Paso 1. Ingresar las piezas en el sistema de bloques:

Simulink trabaja un entorno gráfico, por lo que al iniciarlo se muestra la interfaz mostrada en la ilustración 22 (espacio de trabajo en blanco). Para poder agregar bloques se requiere del acceso al navegador de librerías, que se muestra en la ilustración 23, las librerías disponibles están listadas del lado derecho, de esta manera podemos navegar entre ellas para encontrar los bloques requeridos, por ejemplo, uno de ellos se encuentra en la ilustración 24 que muestra bloques referentes a transformaciones (disponible en la librería Simescape/Multibody).

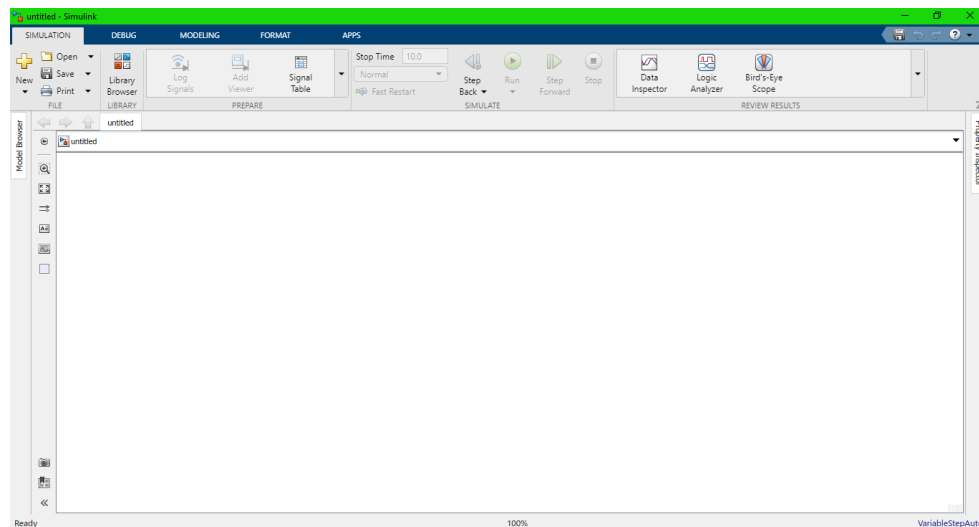


Ilustración 22: Ventana inicial de Simulink

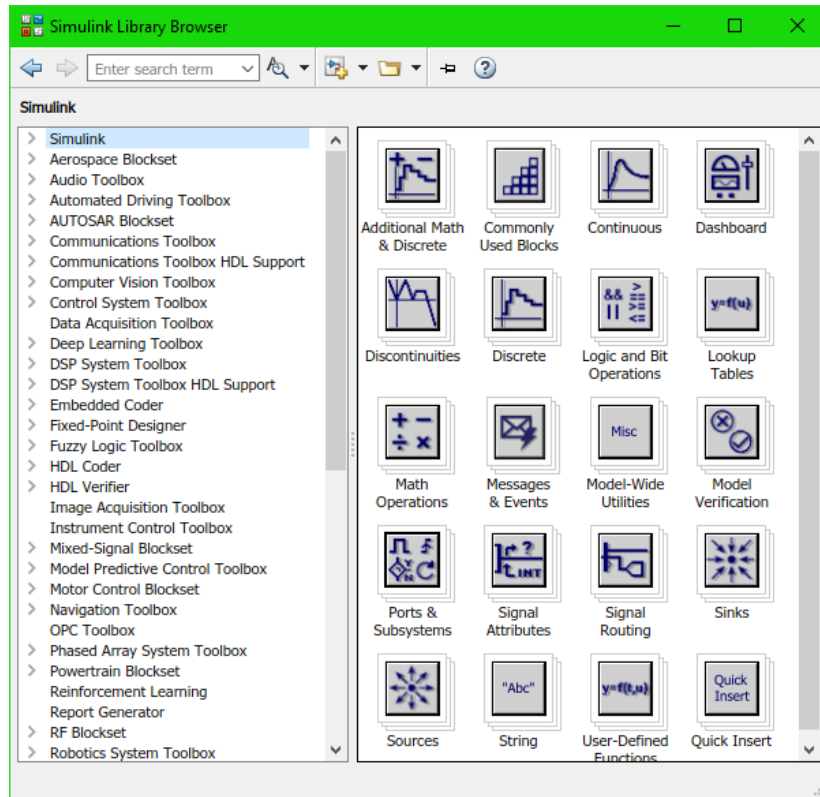


Ilustración 23: Navegador de librerías de Simulink

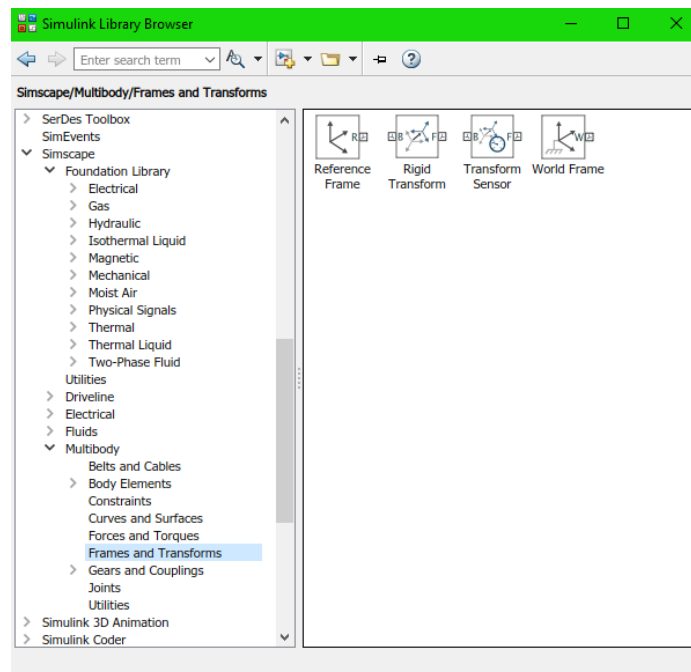


Ilustración 24: Elementos Frame and Transforms de la librería de Multibody de Simscape

Ya que se está familiarizado con las librerías y bloques necesarios, podemos empezar el diagrama del sistema con los tres bloques iniciales explicados anteriormente y observados en la ilustración 25, el diagrama funciona a través de uniones entre los bloques que permiten la conexión de sus funciones.



Ilustración 25: Bloques iniciales de un sistema mecánico en Simulink

Para exportar las piezas del modelo se requiere de un bloque sólido para cada una, como puede observarse en el caso de la cintura en la ilustración 26.

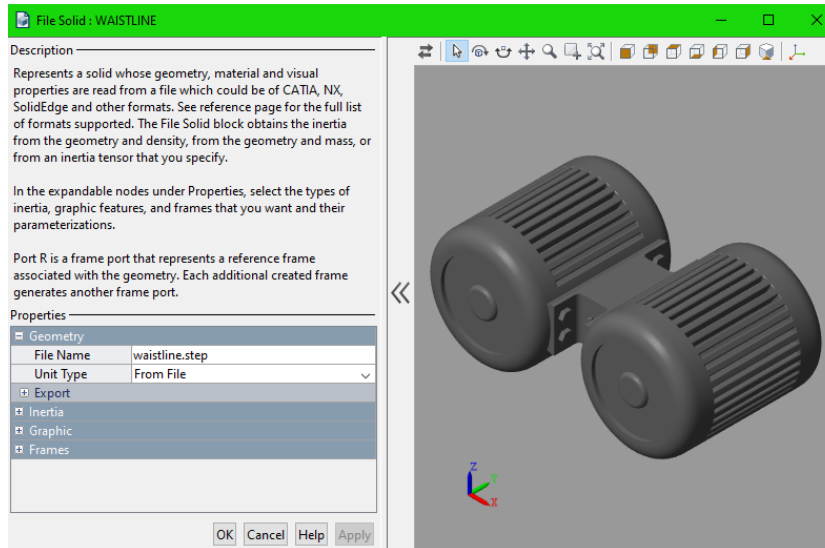


Ilustración 26: Bloque sólido de la cadera en Simulink

De esta manera agregamos las piezas (cintura, tronco, cadera derecha, cadera izquierda, pierna superior derecha, pierna superior izquierda, pierna inferior derecha, pierna inferior izquierda, pie derecho y pie izquierdo en bloques sólidos), además de agregar articulaciones de soldadura entre cada pieza para lograr unir las

en el mismo diagrama, como se observa en la ilustración 27, y su simulación correspondiente en la ilustración 28.

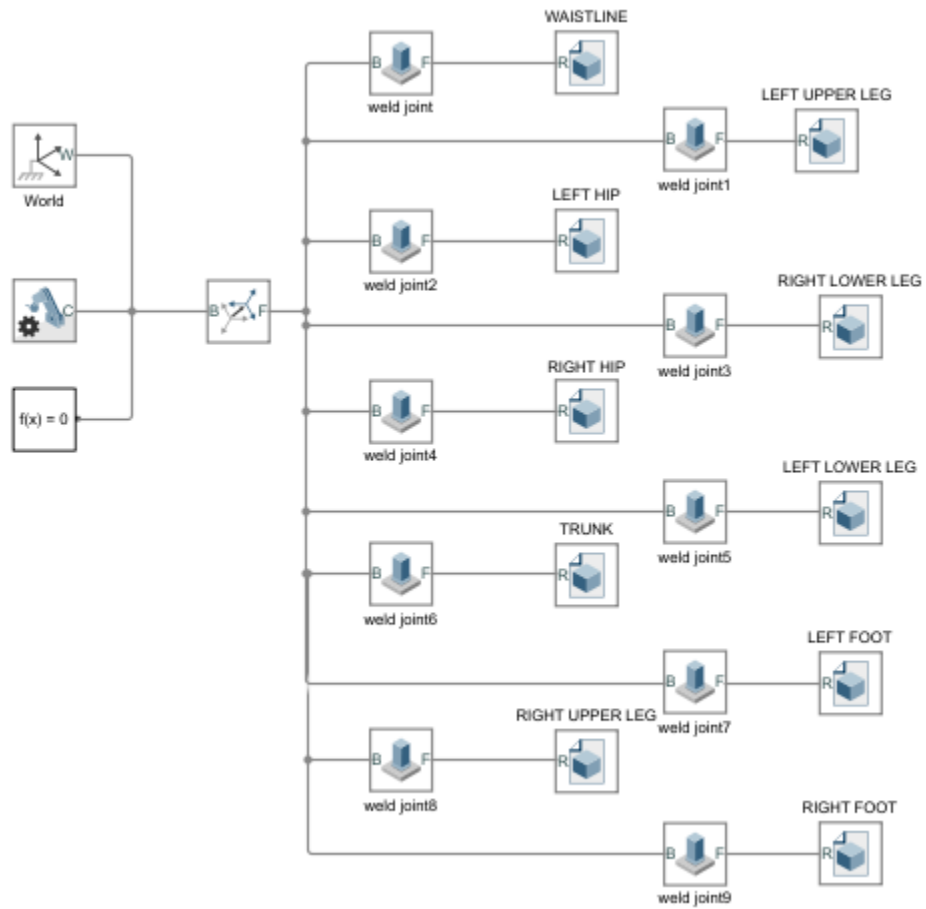


Ilustración 27: Diagrama de bloques inicial del Sistema de caminata para tren inferior robótico

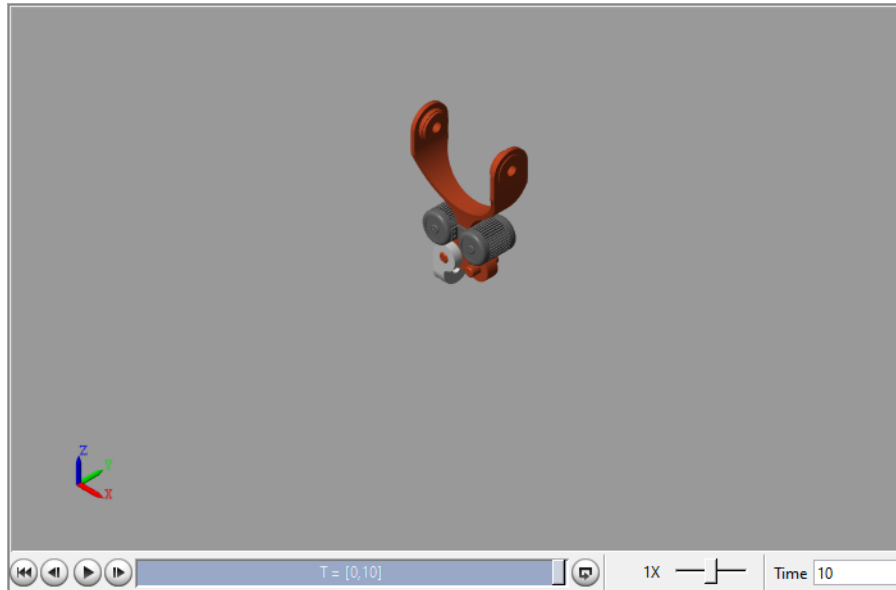


Ilustración 30: Simulación del posicionamiento del tronco, la cintura y cadera derecha del Sistema de caminata para tren inferior robótico

Como puede observarse, las transformaciones al no poder solucionarse en un solo bloque requieren una cantidad considerable de espacio, por lo que se decide encapsular las transformaciones en subsistemas (ilustraciones 32 y 33), uno por cada pieza, como puede observarse en la ilustración 31, no alterando el resultado de la simulación.

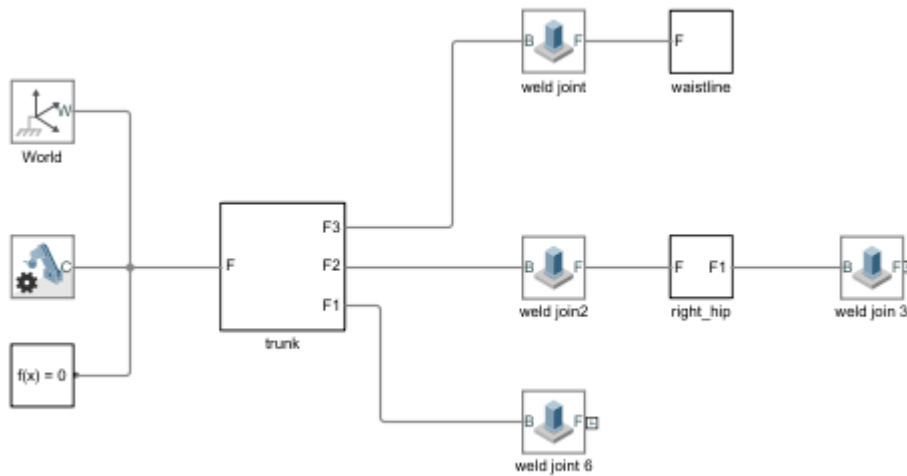


Ilustración 31: Diagrama de bloques del posicionamiento del tronco, la cintura y cadera derecha en subsistemas del Sistema de caminata para tren inferior robótico

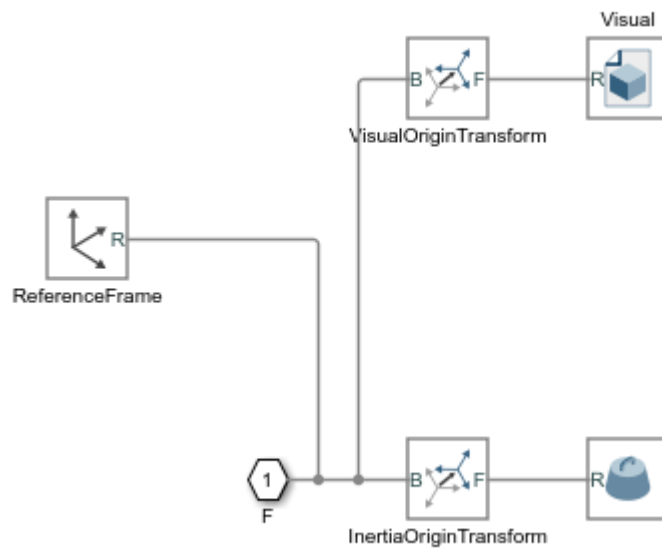


Ilustración 32: Diagrama de bloques del subsistema de la cintura

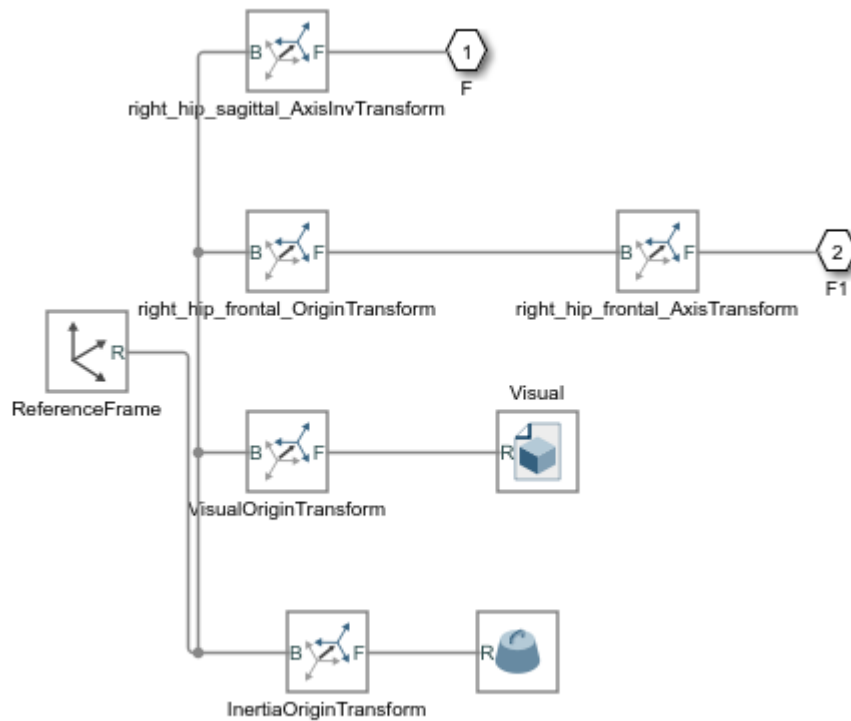


Ilustración 33: Diagrama de bloques del subsistema de la cadera derecha

Al añadir todas las piezas al diagrama en sus posiciones correspondientes se obtiene el diagrama final del segundo paso, como se observa en la ilustración 34 y su correspondiente simulación en la ilustración 35, estando las piezas en su lugar correcto.

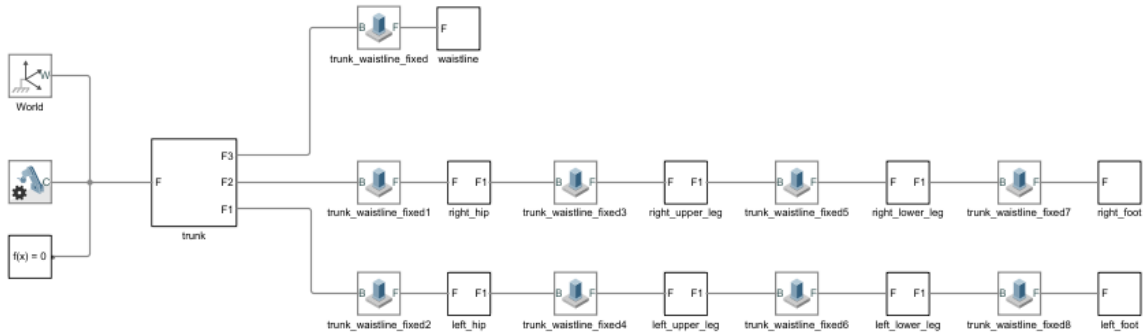


Ilustración 34: Diagrama de bloques del paso dos del Sistema de caminata para tren inferior robótico

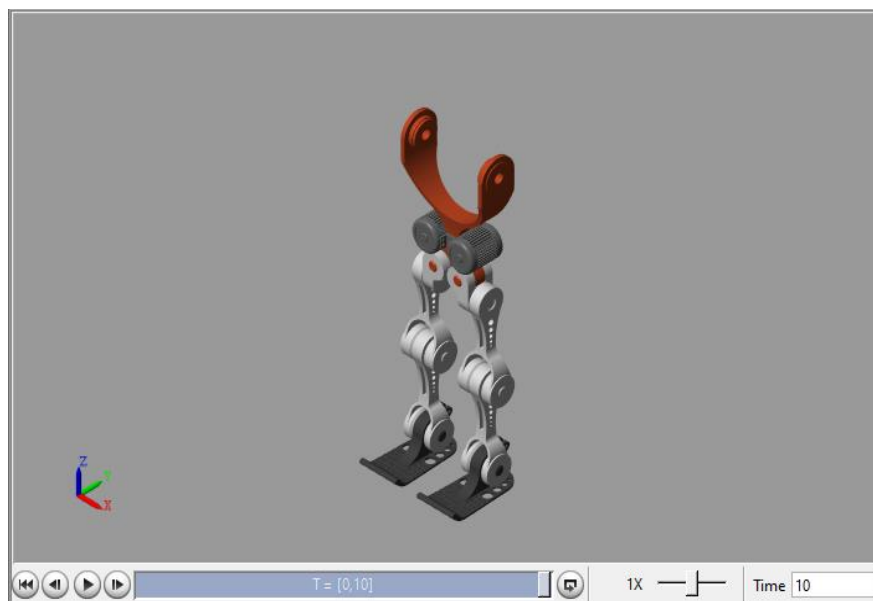


Ilustración 35: Simulación del paso dos del Sistema de caminata para tren inferior robótico

Paso 3. Articulaciones

Se agregan bloques de articulaciones rotacionales sustituyendo a los bloques de articulación de soldadura, esto únicamente se ve reflejado en el diagrama como se ve en la ilustración 36, ya que, al no tener señal de entrada, las articulaciones rotacionales permanecen estáticas.

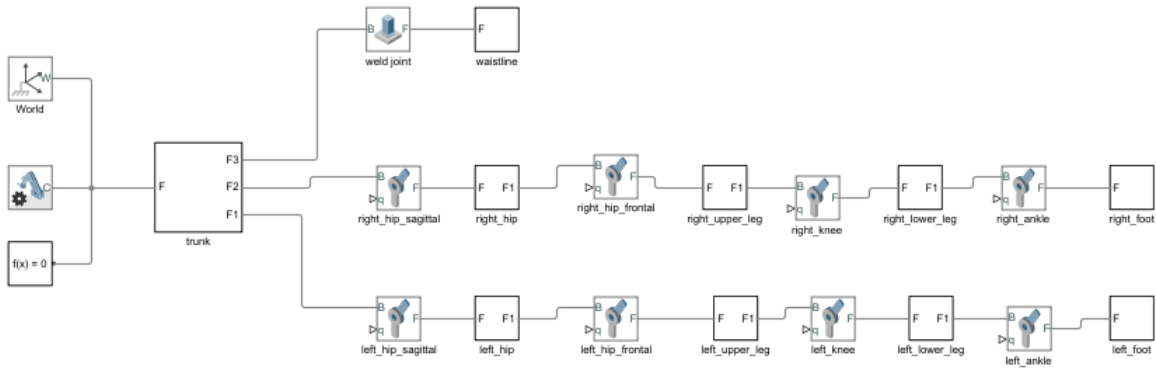


Ilustración 36: Diagrama de bloques con articulaciones rotacionales del Sistema de caminata para tren inferior robótico

Paso 4. Control de movimiento

Como último paso se elige el bloque que controlará el movimiento de las articulaciones rotacionales durante la simulación, esto recae en el bloque constructor de señal, que mediante una gráfica interactiva donde el eje x representa el tiempo en segundos y el eje y representa la cantidad de grados a controlar para el movimiento de las piezas. Los constructores de señales pasaran a llamarse como la articulación que más asemeja a su función en la caminata humana, es decir, la articulación entre la cadera y la parte superior de la pierna será cadera, la articulación entre la pierna superior y la pierna inferior será rodilla y la articulación entre la pierna inferior y el pie será tobillo. Estos constructores de señales serán programados de acuerdo con la tabla de estados, como ejemplo se muestra la ilustración 37.

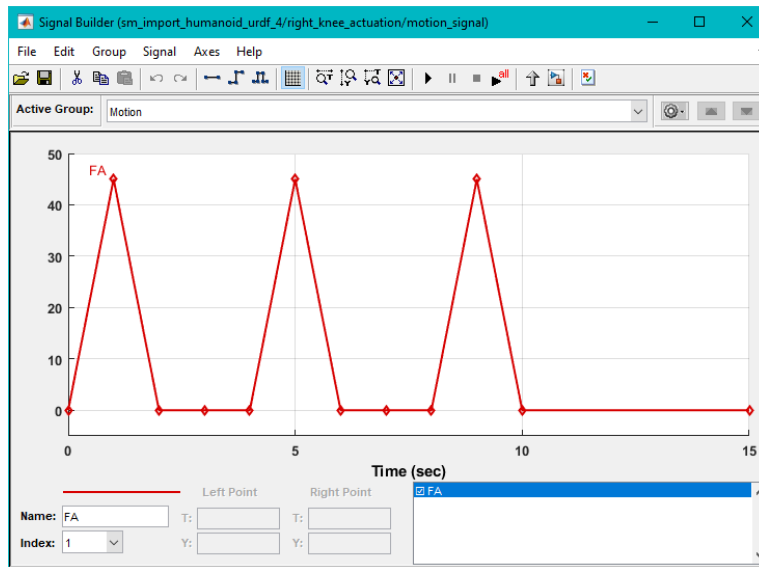


Ilustración 37: Constructor de señal de Simulink de la rodilla derecha

De esta manera se añade un constructor de señal para cada articulación rotacional (cadera derecha e izquierda, rodilla derecha e izquierda y tobillo derecho e izquierdo), como se muestra en la ilustración 38 además de un transformador de señal, para transformar la señal saliente en grados. Estos constructores de señal ocupan un espacio considerable, por lo que se plantea la idea de crear subsistemas para una presentación más limpia.

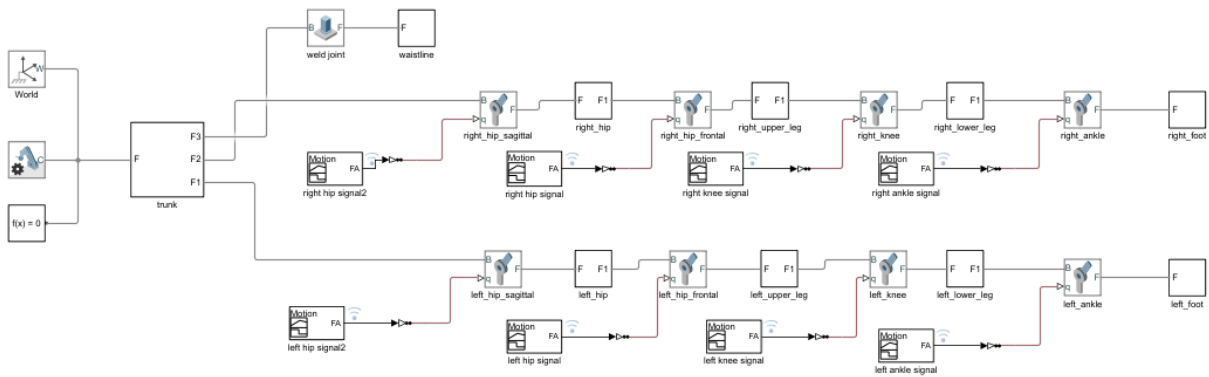


Ilustración 38: Diagrama de bloques del paso 4 del Sistema de caminata para tren inferior robótico

4.2 Resultados

Se obtuvo el diagrama de bloques del Sistema de Caminata para Tren Inferior Robótico (mostrado en la ilustración 40) y su simulación (mostrada en la ilustración 41), para poder visualizar el comportamiento final de las señales, se agrega a la simulación un inspector de datos que permite visualizar el comportamiento de todos los constructores de señales a lo largo del tiempo de simulación en una sola gráfica.

El eje x el tiempo en segundos y como eje y la cantidad en grados, en la gráfica cada señal es representada por un color diferente, siendo la cadera derecha de color azul, la cadera izquierda de color morado, la rodilla derecha de color naranja oscuro, la rodilla izquierda de color rosa, el tobillo derecho color verde y el tobillo izquierdo de color naranja claro, como puede observarse en la ilustración 14.

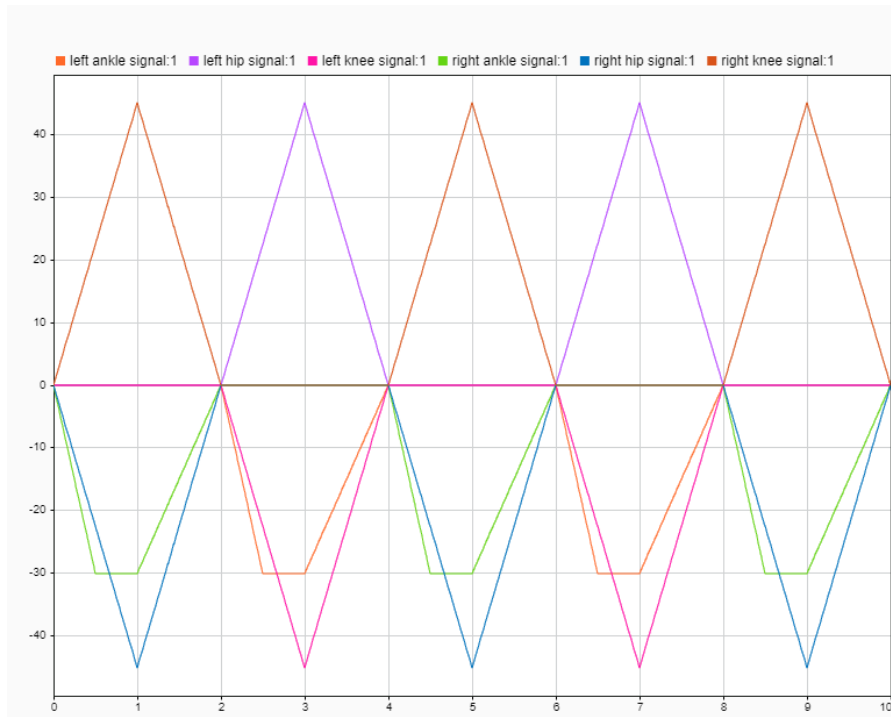


Ilustración 39: Gráfica de las señales del Sistema de caminata de tren inferior robótico

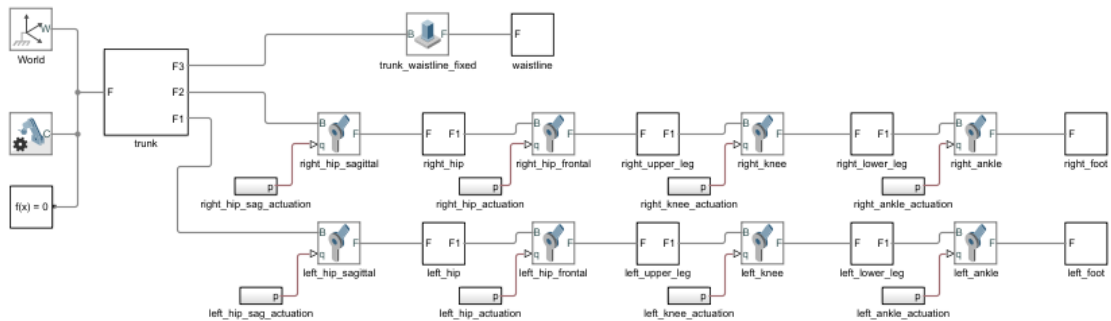


Ilustración 40: Diagrama final del Sistema de caminata de tren inferior

La simulación final muestra el resultado de los movimientos de las piezas unidas en un solo sistema a lo largo de 10 segundos; visualmente corresponde a la simulación de la caminata de tren inferior robótico. Los cuadros de derecha a izquierda corresponden a los segundos: 0, 0.61, 1.48, 2.03, 2.78, 3.64 y 4, lo cual se muestra en la ilustración 41.

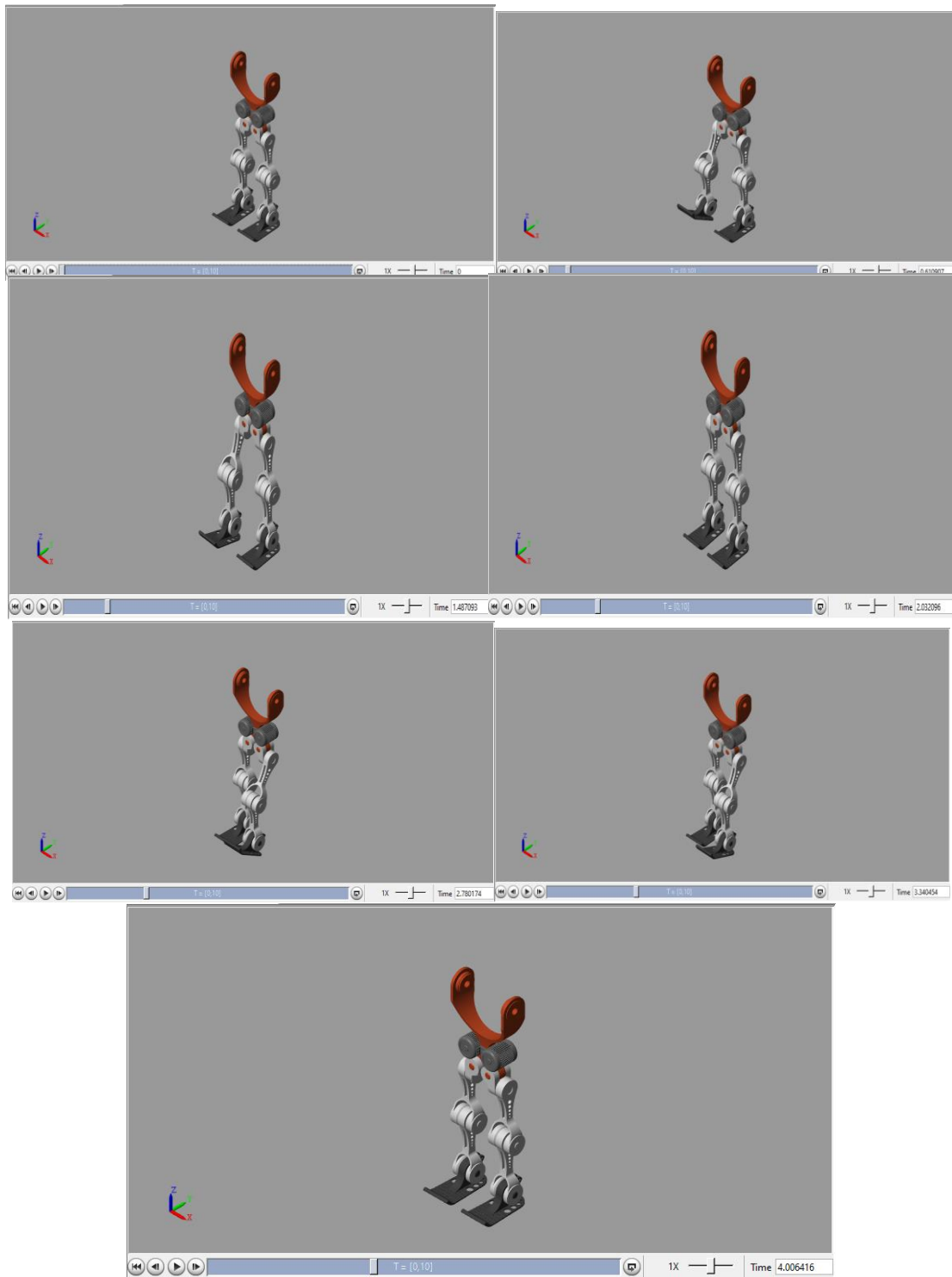


Ilustración 41: Simulación final del Sistema de caminata para tren inferior robótico

5.- Conclusiones y trabajo a futuro

5.1. Conclusiones

La robótica es un área en constante desarrollo, los robots bípedos son un reto para áreas de control, programación, electrónica, matemáticas, mecánica, etc. En particular concluyo que actualmente se trabaja con diversos sistemas de caminata para robots (modelo de péndulo invertido lineal de doble enlace, aprendizaje de refuerzo, modelo de masa-resorte), algoritmos de implementación y simulación.

Después de probar diferentes softwares como 3DS MAX, GeoGebra, ROS (Robot Operating System), AnyLogic, etc. decidí utilizar MATLAB debido a la accesibilidad y flexibilidad de sus elementos para la construcción de modelos basados en funciones matemáticas (cinética, dinámica, parámetros de gravedad, posicionamiento y grados de libertad), gracias a sus herramientas (Simulink, Simscape y Onshape), por lo que concluyo que MATLAB es óptimo para la construcción, visualización y simulación de piezas, así como de un sistema complejo.

La simulación es una herramienta que permite evitar errores, antes de que sucedan físicamente, se pueden especificar el tamaño y comportamiento de las piezas y visualizar el sistema como un conjunto que trabaja de manera coordinada, verificando que la propuesta de diseño sea viable a realizar.

Este trabajo representó un reto personal ya que tuve que investigar la información necesaria para aplicar y recordar conceptos lógicos-matemáticos.

Aprendí el uso de Matlab, software que utilicé dentro de mi formación académica para realizar procedimientos matemáticos e incluso para técnicas de inteligencia artificial, pero no llegué a trabajar con las herramientas de simulación (Simulink, Simscape y Onshape). Para lograr la simulación del tren inferior robótico, me enfrenté a problemas como el diseño y estudio profundo de cada pieza y a veces tuve que replantear lo propuesto originalmente.

La experiencia ha sido enriquecedora, ya que logré cada uno de los objetivos específicos como resultado de un trabajo que realicé a pasos pequeños, lo que me permitió asimilar conocimientos y adquirir nuevos, así como nuevas destrezas en investigación y autoestudio, en general me llevo una experiencia sobre simulación de sistemas robóticos, que actualmente son un campo con oportunidades de desarrollo.

Debido a la pandemia causada por la COVID-19 a principios del 2020, este trabajo ya no se logró implementar físicamente, ya que originalmente se planteó desarrollar

un robot humanoide completo con diferentes trabajos de tesis que se iniciaron paralelamente.

5.2. Trabajo a futuro

Como trabajo a futuro se plantea la idea de llevar el diseño simulado a la realidad, imprimiendo en 3D las piezas utilizadas en el sistema, con materiales resistentes, adaptándolas a su uso con hardware que le permita la movilidad planteada, siendo una posibilidad usar servomotores para los grados de libertad y el lenguaje de Python como medio de interpretación.

Actualmente se desarrollan las partes que complementan el cuerpo de un robot humanoide, para posteriormente ensamblar, hacer pruebas y aplicaciones para un robot social.

Referencias

- [1] Tang Z., Sun Z., Zhou C. y Hu L. (2007). Reference Trajectory Generation for 3-Dimensional Walking of a Humanoid Robot. *Tsinghua Science and Technology* ,12, pp 577-584.
- [2] Li T., Ho Y., Kuo P., Ye Y. y Wu L. (2017). Natural Walking Reference Generation Based on Double-Link LIPM Gait Planning Algorithm. *IEEE Access*,5, pp 2459-2469.
- [3] Galloway K., Sreenath K., Ames A. y Grizzle J. (2015). Torque Saturation in Bipedal Robotic Walking Through Control Lyapunov Function-Based Quadratic Programs. *Tsinghua Science and Technology* ,12, pp 577-584.
- [4] Xu Z. y Gao J. (2017). Robot Simulations Based on Bipedal Spring-Mass Model with Variable Slack Length and Stiffness. *IEEE Access*,5, pp 1156-1163.
- [5] Lin J., Hwang K., Jiang W. y Chen Y. (2016). Gait Balance and Acceleration of a Biped Robot Based on Q-Learning. *IEEE Access*,4, pp 2439-2449.
- [6] Reyes F. (2011). Robótica control de robots manipulables, Ciudad de México, México: Alfaomega Grupo Editor. pp. 43-44, 47.
- [7] Kemalettin E., Okan K. (2009) Natural ZMP Trajectories for Biped Robot Reference Generation, *IEEE Transactions on industrial electronics* vol 56, pp 835-845.
- [8] Deb S., Deb S. (2010) *Robotics Technology and Flexible Automation*, New York, EUA: McGraw-Hill Education., pp.538.
- [9] McComb G. (2018) *Robots Builder's Bonanza*, New York, EUA: McGraw-Hill Education.
- [10] Machado J., Becerra H. y Moreno M. (2015) Modeling and finite-time walking control of a biped robot with feet, *Mathematical Problems in Engineering* vol 2015, pp 1.
- [11] Nicholls E. (1998). Bipedal dynamic walking in robotics (tesis de pregrado). The University of Western Australia, Australia.
- [12] Contreras I. (2008). Análisis Cinemático y Dinámico de un Robot Bípedo (tesis de pregrado) Instituto Politécnico Nacional, Ciudad de México, México.
- [13] Miranda R. (2016). Cinemática y dinámica de robots manipuladores, Ciudad de México, México: Alfaomega Grupo Editor. p 5.
- [14] Fu, K.S.; González, R.C. y Lee, C.S.G. (1987) *Robotics: Control, Sensing, Vision, and Intelligence*. New York, EUA: McGraw-Hill Education.

- [15]Pérez M., Cuevas E., Zaldívar D. (2014). Fundamentos de robótica y mecatrónica con Matlab© y Simulink. Madrid, España: RA-MA. p. 119
- [16] Santana H., J. L. (2006). Control del equilibrio en el proceso de caminado de un robot bípedo. Guadalajara, México: Universidad de Guadalajara.
- [17]Coss R. (1994). Simulación, un enfoque práctico. Ciudad de México, México: Limusa Noriega Editores.
- [18]Guasch A. Piera M. Figueroa J. (2003). Modelado y simulación. Aplicación a procesos logísticos de fabricación y servicios. Barcelona, España: Ediciones UPC.
- [19] Ataurima Arellano M (2013). Matlab y Simulink para ingeniería, Universidad Nacional de Ingeniería. Lima, Perú: Fondo Editorial UCH, pp 7-8, 123.
- [20]Cabezas I. Páez D.J. (2010). Matlab Toolbox de optimización Aplicaciones en ciencias económicas. Colombia, Bogotá: Creative Commons, pp 4.