



**UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO**  
**FACULTAD DE INGENIERÍA**

**ALGORITMO GENÉTICO PARA LA  
OPTIMIZACIÓN DE UNA MARCHA BÍPEDA**

**TESIS**

QUE PARA OBTENER EL TÍTULO DE

**INGENIERO EN MECATRÓNICA**

**P R E S E N T A**

**JORGE ADRIÁN VIZCAYNO GARCÍA**

**DIRECTOR DE TESIS**

**M. EN I. SERAFÍN CASTAÑEDA CEDEÑO**

**CIUDAD UNIVERSITARIA, CDMX, 2016**





Universidad Nacional  
Autónoma de México

Dirección General de Bibliotecas de la UNAM

**Biblioteca Central**



**UNAM – Dirección General de Bibliotecas**  
**Tesis Digitales**  
**Restricciones de uso**

**DERECHOS RESERVADOS ©**  
**PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL**

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

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



# Agradecimientos

A mi papá y mi mamá por haberme apoyado en todo momento, a mis hermanos por su respaldo incondicional, a mis tíos, tías y primos por siempre estar ahí cuando se les necesita.

A la Universidad Nacional Autónoma de México por haberme dado ésta oportunidad de aprendizaje, a la Facultad de Ingeniería y todos mis profesores por todos los conocimientos impartidos.

Al Programa de Apoyo a Proyectos de Investigación e Innovación Tecnológica PAPIIT IT102014 “Exoesqueleto de miembro inferior para pacientes con paraplejia” por haber brindado los recursos necesarios para la realización de éste proyecto.

Y al profesor Serafín por su orientación, que hizo posible la culminación de éste trabajo.

# Índice general

Introducción .....	5
Resumen .....	5
Abstract .....	6
Objetivos .....	6
Justificación.....	3
Robótica bípeda.....	4
1.1    Historia de los robots humanoides .....	4
1.1.1    Investigación robótica en <i>Waseda University</i> .....	5
1.1.2    Desarrollo de robots en Honda .....	6
1.1.3    Otros proyectos humanoides .....	8
1.2    Caminata pasiva y activa.....	9
1.2.1    Nomenclatura y locomoción bípeda.....	11
1.3    Criterios de estabilidad.....	12
1.3.1 <i>Zero Moment Point (ZMP)</i> .....	13
Descripción del modelo.....	15
2.1    Modelo de cinemática directa.....	15
2.2    Modelo dinámico simplificado.....	19
2.3    Algoritmos genéticos.....	21
2.4    Modelo de pruebas .....	25
2.4.1 <i>Bioid GP</i> ®.....	25
Optimización de la marcha.....	29
3.1    Planificación de trayectorias de efectores finales y variables articulares.....	30
3.2    Diseño del algoritmo genético.....	32
3.2.1    Esquema general del AG.....	32
3.2.2    Generación de la población inicial con límites articulares .....	33
3.2.3    Selección, cruce y mutación.....	34

3.2.4	Función objetivo.....	36
	Implementación y Pruebas .....	38
4.1	Simulación cinemática .....	38
4.2	Experimentación en el <i>Bioloid GP</i> ®.....	40
4.3	Discusión de resultados .....	40
	Conclusiones .....	49
	Trabajo a futuro .....	49
	Referencias bibliográficas .....	51
	Referencias electrónicas .....	52
	Apéndices .....	53
	Apéndice A: Hardware y firmware.....	53
	Apéndice B.1: Funciones de generación de población inicial con restricciones .....	56
	Apéndice B.2: Funciones de selección, cruce y mutación.....	57
	Apéndice B.3: Funciones para cálculo de la función objetivo .....	58

# Introducción

## Resumen

Este trabajo presenta la optimización de las trayectorias de marcha de un robot bípedo de 10 grados de libertad (Bioloid GP®) que por medio de un algoritmo genético modifica individualmente las variables angulares de cada articulación para permitir que el avance de su parámetro de estabilidad, el *Zero Moment Point (ZMP)*, tienda a converger a la referencia definida.

Los valores de las variables articulares utilizadas fueron medidas mediante goniometría en personas realizando su ciclo de marcha y mapeadas proporcionalmente a la talla del modelo de pruebas, además para poder generar la simulación de avance, se definieron polinomios de quinto orden para la trayectoria de la cadera, considerando las derivadas temporales como las velocidades y aceleraciones del centro de masa.

También se utilizó un modelo de péndulo invertido como simplificación del modelo dinámico del robot para determinar la posición del *ZMP*, dado el centro de masa global del robot, y se realizaron simulaciones cinemáticas previas y posteriores a la optimización para analizar el desempeño del algoritmo, y finalmente verificarlo de igual manera en el modelo real.

## Abstract

*This work presents a bipedal walking patterns optimization of a 10 degrees of freedom bipedal robot (Bioloid GP<sup>®</sup>) in which a genetic algorithm is used to modify the joint variables individually in order to allow the stability parameter, the Zero Moment Point (ZMP), to converge towards the defined reference.*

*The raw values for the joint variables were measured from a goniometry test in real people while doing the gait cycle, and were mapped into the proportion of the testing model. Also, to be able to generate a moving simulation, fifth order polinomial were defined for the waist trajectory considering the derivates in time as velocities and accelerations of the center of mass.*

*Finally, an inverted pedulum model was used to simplify the dynamic model of the robot to determine the ZMP position, given the global center of mass of the robot, and for the analysis of the algorithm performance, kinematic simulations were covered before and after the optimization so they could be verified in the real model as well.*

## Objetivos

- Diseñar un algoritmo genético que optimice las trayectorias angulares del robot bípedo, para lograr tener una estabilidad dinámica tomando en cuenta el criterio del ZMP, además de encontrar soluciones de trayectorias similares a las humanas y que, al mismo tiempo, no sobrepasen los límites biomecánicos.
- Minimizar el costo computacional de las soluciones mediante el uso de un modelo dinámico simplificado de péndulo invertido, así como verificar el desempeño del algoritmo genético mediante simulaciones.
- Realizar pruebas en el robot bípedo Bioloid GP<sup>®</sup> con las trayectorias angulares generadas con el algoritmo de optimización, para su posterior revisión, análisis y comparación de los resultados teóricos con los datos reales.



## Justificación

La aparición de robots se dio con la necesidad de manipular objetos con alta precisión, también al tratar con objetos o materiales nocivos para el ser humano de manera repetitiva, así como mejorar la productividad y eficiencia en ciertos procesos.

Los robots son interesantes desde cualquier perspectiva, ya sea por su movimiento suave, su detallada mecánica de precisión o incluso la apariencia de inteligencia que proyectan, sin duda tienen características que llaman la atención de las personas. La industria ha tomado la iniciativa al respecto y ha comenzado a construir y vender robots con propósitos personales o de entretenimiento. Por esta razón, la comunidad científica se ha dado a la tarea de investigar distintos tipos de robots que les permitan moverse en el mismo entorno que los seres humanos, lo que los condujo a los robots biológicamente inspirados, y más particularmente a los robots humanoides.

Existen múltiples razones para construir robots con características humanas, partiendo de la perspectiva de un robot dedicado a servicio, tiene que ser capaz de desenvolverse en los mismos lugares por los que transitan las personas, escaleras, espacios estrechos, así como poder alcanzar los mismos lugares a donde llegamos los seres humanos. Los investigadores también han considerado el hecho que el ser humano es un diseño natural que ha pasado por millones de años de mejoramiento continuo lo cual lo hace un modelo a seguir para la resolución de distintas problemáticas, entre ellas, la movilidad. Otra razón a considerar, social, por así decirlo, es debido a que las personas se sienten incómodas ante apariencias que sean extrañas a su entorno regular, por lo mismo, el hecho de tener robots humanoides con ciertas expresiones interactivas ayuda a su integración en la vida cotidiana [12].

La robótica humanoide no se ha quedado únicamente en el desarrollo de máquinas con amplia movilidad, sino que gracias a ella se ha extendido el estudio en diversas áreas como la biomecánica y la inteligencia artificial. Un claro ejemplo de eso se aprecia en que las propiedades de los brazos y piernas robóticas puedan emplearse como prótesis artificiales, o incluso como exoesqueletos que potencien las capacidades físicas humanas, y esto se ha dado en la medida de la comprensión del funcionamiento de nuestras extremidades motrices. Partiendo de los dos ejemplos antes mencionados, éste proyecto nace de la necesidad de asistir a personas con movilidad reducida y problemas de locomoción, donde por medio de un sistema robótico proporcione el movimiento de las extremidades inferiores permitiendo al usuario caminar siguiendo los parámetros biomecánicos de una persona, sin dejar de lado la estabilidad a lo largo del proceso de caminata, y manteniendo reguladas las posiciones de sus articulaciones para evitar daño al usuario.

# Capítulo 1

## Robótica bípeda

Los robots bípedos representan uno de los movimientos más complejos que el ser humano realiza, nosotros lo desempeñamos diariamente sin considerar las complicaciones y acciones que toma en cuenta nuestro cerebro para poderlas desarrollar. Para poder hacer una caminata exitosa nuestro sistema nervioso tiene que conjuntar una serie de movimientos repetidos, regulares y suaves que desplazan nuestro centro de gravedad a lo largo de una trayectoria consumiendo el mínimo de energía.

La marcha normal es un balance entre estabilidad y movilidad, para llegar a ella necesitamos tener una buena movilidad articular y una buena acción muscular, así lograremos comenzar el movimiento, detenerlo, y hacer cambios tanto de velocidad como de dirección, entonces necesitamos cumplir con los siguientes requisitos: 1) Cada pierna debe ser capaz de soportar el peso del cuerpo sin colapsar, 2) el equilibrio debe ser mantenido tanto de forma estática como dinámica en apoyo monopodal, 3) la pierna oscilante debe ser capaz de avanzar hasta la posición en que se pueda convertir en la pierna de soporte y 4) debe poder proporcionarse la fuerza suficiente para realizar el movimiento de las extremidades y avanzar el tronco [1].

### 1.1 Historia de los robots humanoides

El desarrollo de los robots humanoides se ha dado por distintos grupos alrededor del mundo, y principalmente en Japón. Históricamente, la investigación humanoide ha estado a cargo del Profesor Ichiro Kato y su grupo en *Waseda University* desde 1970, donde ellos crearon a “WABOT” (Ver Figura 1.1). Después de ser lanzado en 1990 el “proyecto humanoide”, en colaboración de la misma *Waseda University* y otras compañías, ha habido mucha investigación en humanoide por parte del MIT, NASA, *University of Tokyo*, ERATO, ATR y más. A partir de estas investigaciones, ha nacido el interés de distintas empresas en ésta área lo que ha permitido un avance tecnológico significativo al respecto, en Japón, Honda anunció el humanoide P2 en 1996, un robot con una capacidad de caminata bípeda muy confiable. Seguido de estos resultados, el Proyecto Robot Humanoide (HRP – *Humanoid Robot Project*) fue presentado en 1998, obteniendo importantes resultados a manos de *Kawada Industries* en colaboración del AIST (*National Institute of Advanced Industrial Science and Technology*, en Japón) [2].

### 1.1.1 Investigación robótica en *Waseda University*

Como se mencionó previamente, la investigación empezó con el proyecto Wabot, dirigido por Ichiro Kato en 1970, donde terminaron de desarrollar al WABOT -1 en 1973 que fue considerado como el primer robot antropomórfico de tamaño real creado en el mundo, contaba con un sistema de control de piernas, sistema de visión y uno de conversación, era capaz de hablar con una persona en japonés y de medir distancias y direcciones a objetos usando receptores externos como ojos, oídos y boca artificiales. El WABOT-1 caminaba con sus extremidades inferiores y podía tomar cosas y transportarlas con sus manos que usaban sensores táctiles, el humanoide estaba compuesto del WAM-4 (brazos artificiales) y WL-5 (piernas artificiales). Más tarde en 1980, dentro de los mismos laboratorios comenzó el desarrollo del proyecto WABOT-2 cuya principal tarea constaba en lograr tocar el piano o teclado, del proyecto WABOT-2 cuya principal tarea constaba en lograr tocar el piano o teclado, considerando que al ser una actividad artística el robot requeriría una gran destreza e inteligencia similar a la humana.



**Figura 1.1** Robot humanoide WABOT-1 desarrollado en 1973 (*Waseda University*).

Posteriormente en 1996, los investigadores comenzaron el diseño de WABIAN (*WAseda Bipedal humANoid*), el cual tenía el tamaño de un adulto japonés mujer promedio para hacer trabajo colaborativo con humanos, tendría la capacidad de caminar a la velocidad de un humano contando con un total de 35 grados de libertad (GdL), realizando diferentes tipos de caminata, dinámica hacia adelante, hacia atrás, marcha estática, baile, etc. La versión que lo sucedió fue el WABIAN-RV, que puede realizar varios movimientos, generando sus patrones de movimiento en línea para cada paso, método que mejora adaptabilidad en su entorno y para lo cual cuenta con 43 GdL [2].

El WABIAN -2R (Ver Figura 1.2) fue diseñado para tener una altura de 1475 mm y un ancho de 64.5 kg, su rango de movimiento fue desarrollado tomando como referencia las mediciones hechas a movimientos humanos en rehabilitación. En la junta de la cadera, fue implementado un nuevo mecanismo redundante de 2 GdL que le permite caminar con las rodillas extendidas tal como lo hacen los humanos debido a la independencia de sus movimientos, además gracias a la junta pasiva que se tiene en el pie, le permite hacer el primer contacto con el talón y el despegue con los dedos del pie [3].

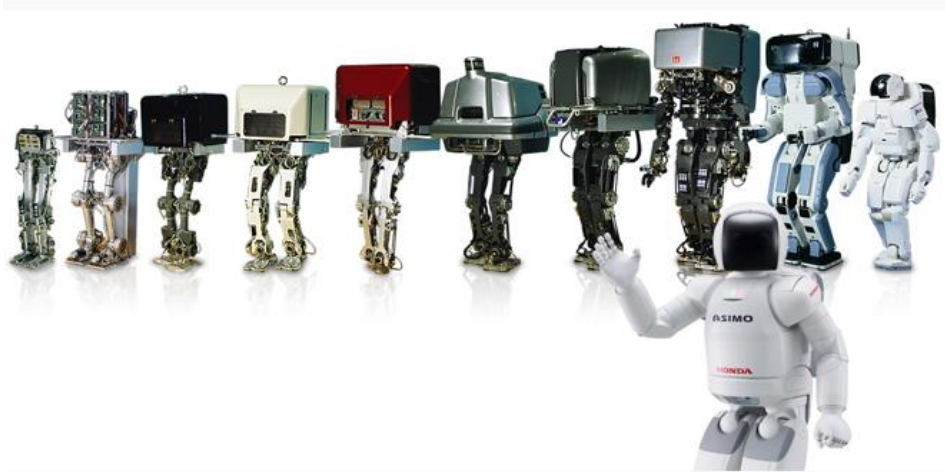


*Figura 1.2 Robot bípido WABIAN-2R (Waseda University) [3].*

### 1.1.2 Desarrollo de robots en Honda

En 1986, la compañía japonesa, Honda, empezó el desarrollo de robots modelados con base en humanos, estudiando los principios fundamentales de la locomoción bípida, Honda investigó y observó todo tipo de caminata, realizó numerosos experimentos y recolectó una inmensa cantidad de datos, con los cuales en el modelo E0, logró caminar poniendo una pierna anteponiéndose a la otra, aunque le llevaba casi 5 segundos entre cada paso y le permitía únicamente realizar caminata lenta en línea recta.

Para alcanzar una caminata rápida, Honda mantuvo la investigación y análisis de la caminata humana, considerando su movimiento y la correcta ubicación de las juntas para lograrlo. Y fue hasta el modelo E2, entre 1987 y 1991, donde se logró desarrollar un robot capaz de hacer una caminata dinámica alcanzando una velocidad de 1.2 km/h, las modificaciones que se hicieron tanto al E2 como al E3 consideraron una referencia del esqueleto humano para ubicar las juntas de la pierna, así como su rango de movimiento, dimensiones y centro de gravedad; instrumentaron a los humanoides con sensores de par en las juntas, sensores de fuerza de 6 ejes y un acelerómetro y giroscopio para determinar su posición.



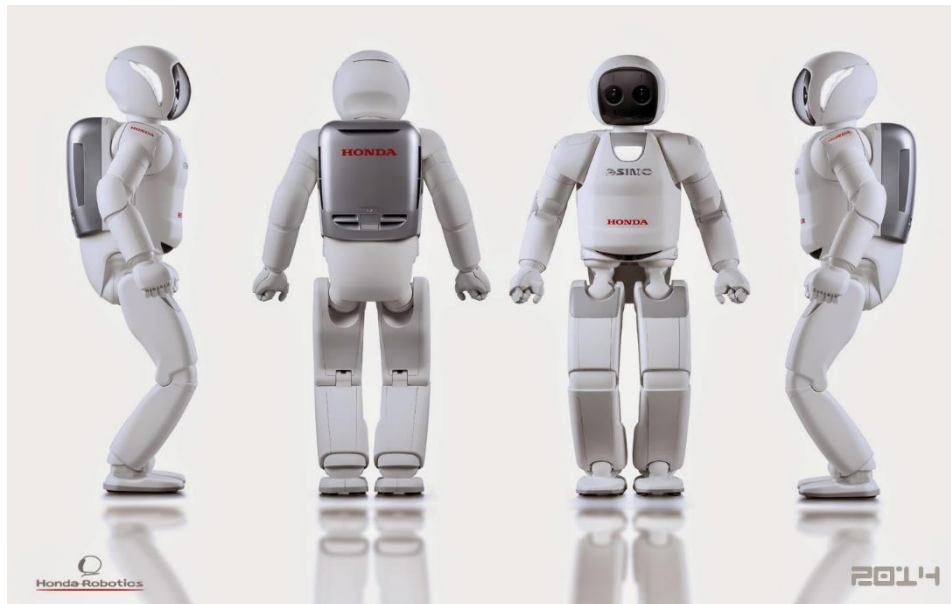
*Figura 1.3 Historia de robots bípedos en Honda (1986 - 2007) [4].*

## CAPÍTULO 1 ROBÓTICA BÍPEDA

Después, entre 1991 y 1993, Honda enfocó su investigación en la estabilidad de la caminata, en el modelo E4 aumentó la longitud de la rodilla 40 cm para simular el paso rápido humano a una velocidad de 4.7 km/h, más adelante el E5 se convirtió en el primer modelo de locomoción autónomo, y luego con el E6, alcanzó un control autónomo de balanceo subiendo y bajando escaleras o parándose sobre un obstáculo, basado en 3 funciones básicas: 1) Control de la reacción del suelo, 2) Control de la locación del pie y 3) Control del *ZMP* objetivo.

Una vez alcanzada la estabilidad en la marcha, Honda evolucionó sus robots agregando a las piernas un cuerpo superior en 1993, creando con el P1 el primer prototipo de su categoría. La razón por la que Honda intentaba realizar éstos humanoides era para poder trabajar en armonía con personas, y para facilitar la operación con el mismo, se decidió que el robot debería tener 2 brazos.

Y finalmente en noviembre del 2000, nació ASIMO (*Advanced Step in Innovative Mobility*, ver Figura 1.4), robot concebido para funcionar en un entorno rodeado de humanos en un futuro cercano, por esa razón fue diseñado con una altura de 120 cm, tamaño que le permite operar los interruptores de luz, perillas de puertas, y también utilizar las mesas o escritorios de trabajo. ASIMO cuenta con la tecnología i-WALK, la cual consiste en un control de predicción de movimiento que le permite cambiar su centro de gravedad con anticipación sabiendo su siguiente movimiento. En el 2002, dejaron de lado la movilidad y se enfocaron sobre todo en sus habilidades de comunicación, para esto, implementaron visión artificial con reconocimiento de objetos móviles, posturas y gestos, además de distinción de ciertos sonidos y reconocimiento de caras.



**Figura 1.4** Robot humanoide ASIMO (2014) [4].

Para la siguiente versión de ASIMO, en 2005, se hicieron notables cambios, empezando con las dimensiones, una altura de 130 cm, distancia hombro-hombro de 45 cm y un ancho de 37 cm. Éstas modificaciones se hicieron para poder lograr tareas que requerían capacidades físicas avanzadas, tal es el caso de saludar con la mano a una persona y poder caminar mientras lo hace, así como la función de cargar objetos mediante una bandeja o empujando un carro. Más allá, el desarrollo de una nueva postura de sistema de control mejoró dramáticamente la capacidad de correr de ASIMO, permitiéndole alcanzar una velocidad de 6 km/h y hacerlo siguiendo un patrón circular. El sistema de control consiste en prevenir el giro y deslice del pie contra el suelo al momento de aterrizar y despegar mientras está corriendo, esto es posible gracias a la teoría desarrollada para girar y flexionar el torso proactivamente, y también gracias a que ASIMO fue

## CAPÍTULO 1 ROBÓTICA BÍPEDA

equipado con un circuito de procesamiento de alta velocidad, unidad motriz de rápida respuesta y alta potencia, así como una estructura altamente rígida y liviana; además de diversos sensores para poder localizar obstáculos en su camino.

Por último, la más reciente modificación de ASIMO fue mostrada al público en el 2011, el *All-new ASIMO* (- Completamente nuevo ASIMO) cuenta con un total de 57 GdL, distribuyendo 3 en la cabeza, 7 por cada brazo, 13 en cada mano, 2 en la cadera y 6 más por pierna; lo que le permite alcanzar una velocidad de 9 km/h al correr y caminar en superficies irregulares, además de poderlo hacer hacia adelante y hacia atrás, y saltar en una pierna continuamente incluso avanzando mientras lo hace. Esta última versión pasó de ser una máquina automática a una autónoma basada en comportamiento, ASIMO genera comportamientos autónomos que le permiten hacer predicciones dada una información almacenada y de esa manera determina su siguiente comportamiento sin ser controlado por un operador. Todos esos comportamientos los hace evaluando las entradas de múltiples sensores que son equivalentes a los sentidos de visión, audición y tacto de los humanos, siendo capaz de anticipar si chocará con alguna persona en caso de que sus trayectorias se intersecten, o incluso reconocer las de muchas personas hablando al mismo tiempo simultáneamente, tarea igualmente complicada para un ser humano de realizar [5].

### 1.1.3 Otros proyectos humanoides

Además de Honda y de *Waseda University* existen otros grupos que se han dedicado al desarrollo e investigación de robots humanoides en Japón. El primero de ellos fue un proyecto que empezó después del lanzamiento del E0 de Honda, que fue promovido en Japón como el HRP (*Humanoid Robot Project*) y llevado a cabo por *Kawada Industries* y el AIST desarrollando múltiples prototipos desde 1999 donde ganaron un contrato de diseño y construcción de un robot humanoide otorgado por la Universidad de Tokio, continuando su investigación en el 2010 mostraron al público su última plataforma robótica, el HRP-4 (Ver Figura 1.5) [4].

También en la Técnica Universidad de Múnich, Alemania, desarrollaron a Johnnie, un robot bípedo autónomo; cuyo principal objetivo era realizar una marcha antropomórfica con pasos dinámicamente estables y similares a los de los humanos. El robot es capaz de caminar en terreno regular, irregular y sobre curvas; el robot es autónomo en términos de sensores, actuadores y poder computacional, pero la energía es suministrada por cable. [6] En EUA, también existen distintas agencias e instituciones enfocadas en el estudio de robots humanoides, la NASA en colaboración con *General Motors* han desarrollado en el *Johnson Space Center*, al *Robonaut 2*, diseñado para desarrollar actividades extra vehiculares (EVA) o caminatas espaciales, por lo que es capaz de manejar un amplio rango de herramientas humanas [7].



**Figura 1.5** De izquierda a derecha, HRP-2, HRP-3 y HRP-4 [10].

Otro robot ampliamente desarrollado es Atlas, creado por Boston Dynamics con un total de 28 GdL hidráulicos, una cabeza articulada con sensores que incluyen estereo cámaras, telémetro láser y alimentado externamente por medio de un arnés flexible. Éste sofisticado prototipo diseñado para trabajar en exteriores y en terrenos difíciles fue financiado por la DARPA (*Defense Advanced Research Projects Agency*) para ser distribuido en distintas instituciones permitiéndoles diseñar los controladores e inteligencia artificial de Atlas concursando en el *DARPA's Robotics Challenge*. [8] Finalmente, existen plataformas robóticas educativas desarrolladas por empresas independientes que facilitan la implementación de sistemas de control diseñados para mejorar la marcha bípeda, como es el Aldebaran Nao (ver Figura 1.6), que cuenta con 25 GdL, 2 cámaras, 1 unidad de medición inercial, sensores táctiles y micrófonos direccionados, además de incluir con un software embebido que tiene su propio kit de desarrollo y sistema operativo para facilitar la programación. [9] Robotis es otra compañía enfocada al desarrollo de humanoides, cuenta con diversas plataformas e incluso una a tamaño real, éste proyecto se realizará con el modelo educativo Bioloid GP® (ver Figura 1.6) el cual cuenta con 5 GdL por cada pierna, y 3 en cada brazo, actuados por servomotores Dynamixel AX-18A de alto desempeño, más adelante se detallará el modelo sobre el que se efectuarán las pruebas [11].



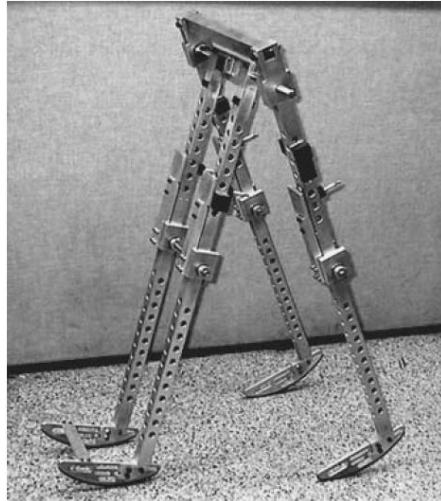
**Figura 1.6** a) Robotis Bioloid GP® [11], b) Aldebaran Nao [9].

### 1.2 Caminata pasiva y activa

La locomoción puede simplificarse como un ciclo, donde la secuencia articulada se repite siempre y cuando la velocidad sea constante. La marcha, por ejemplo, es una sucesión de fases de balanceo y apoyo. Por convención el inicio del ciclo de caminata se da cuando el talón derecho hace contacto con el suelo, seguido de dos pasos, izquierdo y luego derecho, siempre manteniendo al menos un pie aterrizado. Dividiendo éste ciclo en porcentajes de duración, el contacto del pie derecho con el suelo es considerado instantáneo. (0% de la duración de la marcha).

Ambos pies están sobre el suelo entre 0-15% del ciclo. Ésta es conocida como la fase de soporte doble que corresponde al aterrizaje de la pierna derecha, y a la propulsión de la pierna izquierda. Progresivamente, el pie izquierdo se considera levantado hasta que el dedo gordo del pie deja el piso. Entre 15-50% del ciclo, solo el pie derecho toca el piso en una fase monopodal. La pierna izquierda está en una fase de balanceo. Cuando el talón izquierdo impacta con el suelo y el pie aterriza, hay una segunda fase de soporte doble entre 50-65% del ciclo. Ésta fase se completa con el dedo gordo del pie derecho dejando el suelo. Entre 65-100% del ciclo, se tiene la fase monopodal en el pie izquierdo, con la pierna derecha balanceándose. El ciclo termina con un nuevo impacto del talón derecho con el piso [13].

El concepto de locomoción pasiva (*Passive Walking*), es decir, sin actuadores, fue introducido por primera vez en 1988 por Tad MacGeer, quien estudió y construyó un caminador pasivo. La virtud de éste mecanismo es que no necesita energía externa ni de ningún control para caminar por una pendiente, el movimiento viene propulsado por la energía potencial de una bajada. El sistema funciona como 2 péndulos acoplados, la pierna soporte es un péndulo invertido y la pierna oscilante es un péndulo normal unido a la pierna soporte. Con la distribución de masas y longitudes adecuada, se consigue un ciclo límite estable, una trayectoria nominal que se repite. Inicialmente el mecanismo no tenía rodillas, pero una mejora posterior implementó esta articulación manteniendo la estabilidad. El movimiento lateral se inhibe al contar con cuatro piernas.



**Figura 1.7** Copia del mecanismo de McGeer. Elaborado en *Cornwell University*.

McGeer hizo un estudio exhaustivo de la mecánica y dinámica del caminador. Demostró las condiciones necesarias para asegurar la estabilidad del ciclo límite. Cabe destacar una diferencia sustancial respecto al algoritmo ZMP: el caminador pasivo describe trayectorias inestables (como un péndulo invertido), manteniendo el ciclo límite estable [14].

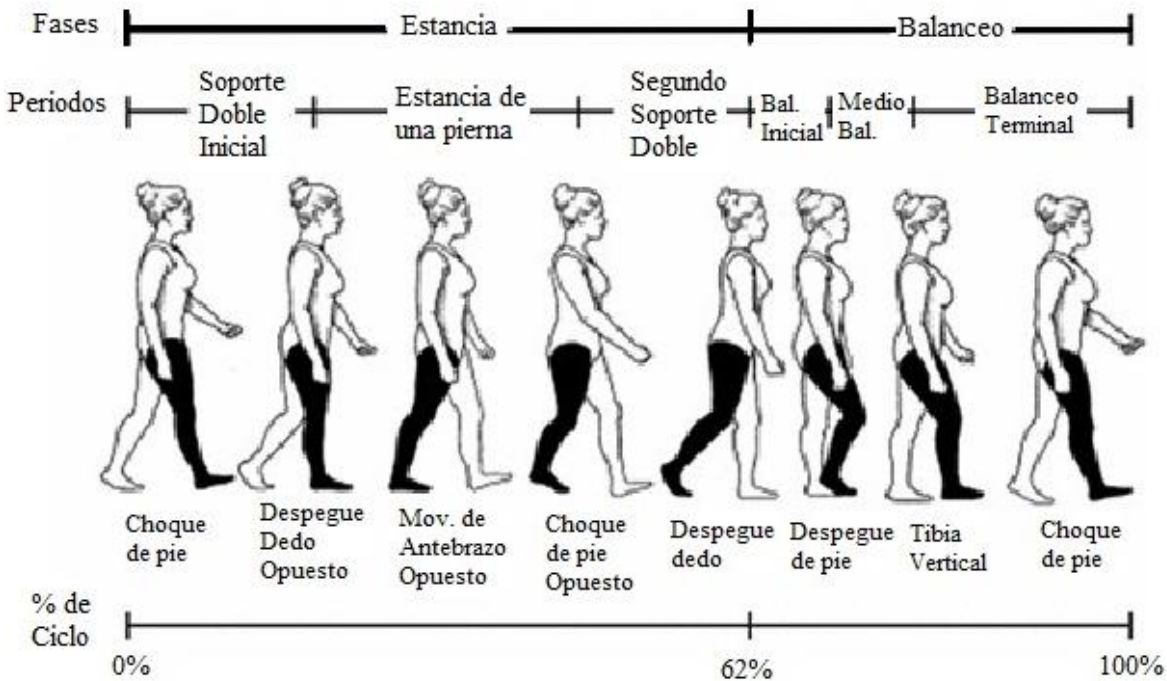
También es de gran interés el hecho que los movimientos de este simple mecanismo, que no tienen actuación ni control, tengan una gran semejanza al de los humanos. Esto sugiere la idea de que los humanos aprovechamos las dinámicas propias de nuestro cuerpo para optimizar el consumo energético durante la locomoción. Este descubrimiento abrió una nueva puerta de investigación: robots que aprovechen los mismos conceptos de la locomoción pasiva aplicándolos para conseguir locomoción sobre llanos y subidas, lógicamente añadiendo elementos actuadores y de control. Esto se le conoce como caminata dinámica (*Dynamic Walking*), [1], concepto aplicado en distintos robots como el bípedo *Cornwell*, bípedo y los robots antes mencionados como la serie HRP, ASIMO, entre otros.

En [13] también se discute la existencia de la caminata cuasi-pasiva dinámica (*Quasi-passive Dynamic Walking*), que consiste en pasos actuados por fases de soporte doble instantáneas que reinician una fase de soporte simple pasiva. Esto se logra mediante control de impulsos de par, es un concepto en el cual divide la aplicación de dichos pares en 3 eventos, el primero es para reducir las velocidades finales de la fase de soporte simple antes del impacto, el segundo evento es como tal el impacto del pie con el suelo y el tercero es la aplicación complementaria del par que determina las velocidades iniciales de la siguiente fase de soporte simple. Normalmente éste problema tiene más variables de las que las ecuaciones algebraicas podrían resolver en la condición de soporte doble, sin embargo, mediante la optimización basada en la minimización de un criterio energético se puede resolver dicha indeterminación.



1.2.1 Nomenclatura y locomoción bípeda

El interés por los patrones de movimiento humanos yace desde tiempos prehistóricos, pero no fue sino hasta 1885 donde Marey, el fisiólogo francés utilizó una “pistola” fotográfica para registrar los desplazamientos en la marcha humana y equipo crono-fotográfico para obtener un diagrama de eslabones de un corredor. El término de la descripción de dichos movimientos humanos se le conoce como *cinemática*, la cual cuenta con una determinada convención para mantener congruencia en sus variables [15].



**Figura 1.8** Fases del ciclo marcha bípeda. Inicia con una fase de soporte doble cuando impacta el talón, sigue de una fase de soporte simple cuando los dedos del pie despegan del suelo hasta que el pie opuesto impacte el talón y continúa con la fase de balanceo del pie opuesto para recomenzar el ciclo luego de impactar nuevamente el primer talón [18].

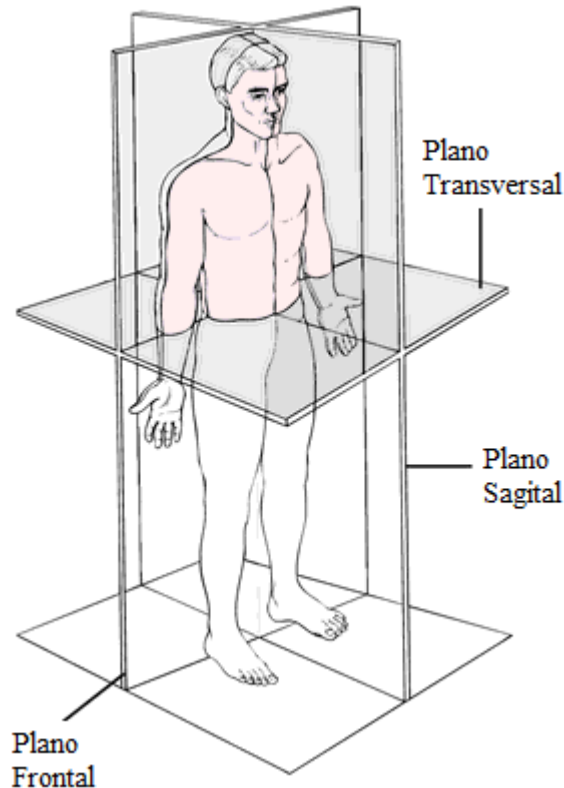
Un bípedo es una cadena cinemática abierta que consiste en 2 sub-cadenas llamadas piernas, una sub-cadena llamada torso conectadas por un punto común llamado cadera. Una o ambas piernas tienen que estar en contacto con el suelo, cuando solamente una pierna está en contacto con el suelo, a la pierna en contacto se le conoce como pierna de apoyo y la otra es llamada pierna de balanceo. El extremo de una pierna, ya sea que tenga eslabones constituyentes o no de un pie, se le denomina pie. La fase de soporte simple o fase de balanceo es definida para ser una fase de locomoción donde sólo un pie se encuentra sobre el suelo. En contraparte, el soporte doble es la fase donde ambos pies se encuentran sobre el suelo. Por lo tanto, la caminata es entonces definida como fases alternantes entre soporte simple y doble, con el requisito que el desplazamiento de componente horizontal del centro de masa (CoM) por sus siglas en inglés) del robot es estrictamente puntual, así mismo en esta descripción se asume que los pies no se deslizan cuando están en contacto con el suelo.

El plano sagital es el plano longitudinal que divide el cuerpo en secciones derecha e izquierda, el plano frontal es el plano paralelo al eje del cuerpo y éste separa el cuerpo en porciones frontal y trasera. Y el plano transversal es perpendicular al plano sagital y frontal [16]. Ver Figura 1.9.

### 1.3 Criterios de estabilidad

En un contexto mecánico, el equilibrio de un cuerpo se da cuando la proyección de su centro de gravedad se encuentra dentro de su base de sustentación o apoyo. Dicho esto, la estabilidad *estática* de un cuerpo se da siempre y cuando la condición antes mencionada se cumpla, sin embargo, la bipedestación es un caso particular, puesto que según el término físico no se encuentra en equilibrio para cualquier instante, sino que es un desequilibrio constantemente compensado [1].

Por esa razón, se utilizan otro tipo de criterios para garantizar la estabilidad en los bípedos, llamados criterios de estabilidad *dinámica*. El más popular, propuesto por el Profesor Miomir Vukobratovic, es el *Zero Moment Point (ZMP)*, es la base de casi todos los controladores de estabilidad de caminata de robots bípedos el cual se detallará en la siguiente sección. También, otros investigadores han propuesto algunos otros, como el Dr. Ambarish Goswani, que propuso el *Foot Rotation Indicator (FRI)*, el cual realiza una



**Figura 1.9** Planos de referencia en el cuerpo humano [19].

extensión al *ZMP*, en el sentido que no es necesario mantenerlo dentro del polígono convexo para obtener estabilidad dinámica de la caminata y depende más bien en el sentido del momento de inercia en el borde del soporte. Actualmente, la investigación tiene a cubrir un criterio más robusto, con puntos multi-contacto y sobre terreno irregular, para esto el *Contact Wrench Sum (CWS)* es propuesto por el Dr. Hirohisa Hirukawa y sus colaboradores del AIST, donde se busca que el *CWS* sea un elemento interno del *Contact Wrench Cone (CWC)* el cual representa el cono formado por las fuerzas de contacto. Este trabajo no tiene

como objetivo dar una descripción extensiva de dichos criterios, pero se puede consultar [2, 16, 17] para una descripción más detallada.

### 1.3.1 Zero Moment Point (ZMP)

Para definir éste criterio de estabilidad consideraremos un mecanismo en fase de soporte simple (Figura 1.10 (a)), con el pie completo sobre el piso. Para simplificar el análisis podemos despreciar la parte del mecanismo que se encuentra sobre el tobillo del pie de soporte (punto A) y reemplazar su influencia por la fuerza  $\mathbf{F}_A$  y el momento  $\mathbf{M}_A$  (Figura 1.10 (b)), donde el peso del pie mismo recae en su centro de gravedad (punto G). El pie también experimenta la reacción del piso en el punto P, la cual mantiene todo el mecanismo en equilibrio.

En general, la reacción del suelo completa consiste en tres componentes de la fuerza  $\mathbf{R}$  ( $R_x, R_y, R_z$ ) y el momento  $\mathbf{M}$  ( $M_x, M_y, M_z$ ). Debido a que la fuerza de fricción actúa en el punto de contacto con el suelo, y el pie sobre el piso esta en reposo, esas componentes de la fuerza  $\mathbf{R}$  y del momento  $\mathbf{M}$  que actúan en el plano horizontal serán balanceadas por fricción. Por lo mismo, la fuerza de reacción horizontal ( $R_x, R_y$ ) representa la fuerza de fricción que está balanceando la componente horizontal de la fuerza  $\mathbf{F}_A$ , donde el momento de reacción vertical  $M_z$  representa el momento de las fuerzas de reacción de la fricción (Figura 1.10 (c)) que balancea la componente vertical  $\mathbf{M}_A$  y el momento inducido por la fuerza  $\mathbf{F}_A$ . Así, si asumimos que el contacto entre el pie y el piso es sin deslizamiento, la fricción estática compensará las componentes de fuerza horizontales ( $R_x, R_y$ ) y el par de reacción vertical  $M_z$ . La fuerza de reacción vertical  $R_z$  representa la reacción del suelo que balancea las fuerzas verticales. Aún resta considerar el balance de la componente horizontal del momento generado por el peso del pie. De cualquier manera, debido a la naturaleza unidireccional de la conexión entre el pie y las componentes horizontales del piso de todos los momentos activos pueden ser compensados únicamente cambiando la posición de la fuerza de reacción  $\mathbf{R}$  dentro del polígono de soporte. Por consiguiente, la componente horizontal del momento  $\mathbf{M}_A$  cambiará la fuerza de reacción a la posición correspondiente, para balancear la carga adicional.

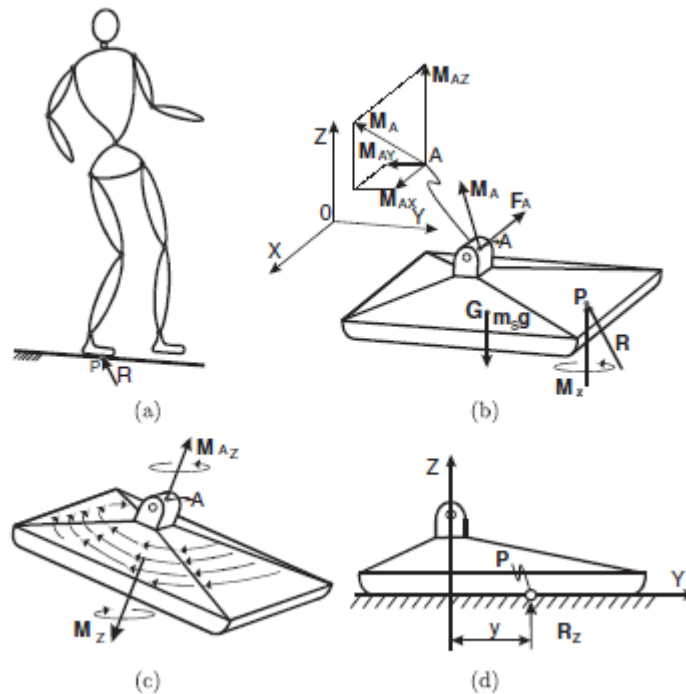


Figura 1.10 Mecanismo bípedo y fuerzas actuando sobre su suela [20].

Esto se ilustra en la Figura 1.10 (d), donde para simplificar el modelo, se presenta un caso planar en el plano  $y$ - $z$ . El momento  $M_{Ax}$  es balanceado al cambiar el punto de acción de la fuerza  $R_z$ , cuya intensidad es determinada del balance de fuerzas que actúan sobre el pie, con la correspondiente distancia  $y$ . Es necesario enfatizar que todo el tiempo la fuerza de reacción se encuentra dentro del área cubierta por el pie, el aumento del momento en el tobillo se compensará cambiando la posición de ésta fuerza, y no existirán componentes de los momentos  $M_x$  y  $M_y$ . Esta es la razón por la que en la Figura 1.10 (b) en el punto P solo existe la componente  $M_z$ .

Sin embargo, si el polígono de soporte real no es lo suficientemente grande para abarcar la posición apropiada de la fuerza  $\mathbf{R}$  para balancear la acción de momentos externos, la fuerza  $\mathbf{R}$  actuará en el borde del pie y la parte descompensada de la componente horizontal del momento de reacción causará la rotación del mecanismo sobre el borde del pie, lo que puede resultar en volcamiento del mecanismo. Por lo tanto, podemos decir que la condición suficiente y necesaria para la locomoción del mecanismo para estar en equilibrio dinámico es que para el punto P donde la reacción del suelo está actuando,

$$\begin{aligned} M_x &= 0, \\ M_y &= 0. \end{aligned}$$

Dado que las dos componentes relevantes para la realización del balance dinámico son iguales a cero, una opción natural para llamar éste punto era *Zero-Moment Point*. O, en otras palabras, todo el tiempo la reacción del suelo debida al pie cayendo en el suelo puede ser reducida a la fuerza  $\mathbf{R}$  y una componente vertical del momento  $M_z$ ; el punto P en el cual la fuerza de reacción está actuando representa el *ZMP*. Por último, en [20] se definen dos interpretaciones:

**ZMP interpretación 1:** *ZMP* es definido como el punto sobre el suelo en el cual el momento neto de fuerzas inerciales y gravitacionales no tiene componente a lo largo de los ejes horizontales.

**ZMP interpretación 2:**  $p$  es el punto que  $T_x = 0$  y  $T_y = 0$ , donde  $T_x$  y  $T_y$  representan los momentos alrededor de los ejes  $x$  y  $y$  generados por la fuerza de reacción  $F_r$  y el par de reacción  $T_r$ , respectivamente. El punto  $p$  es definido como el punto de momento cero (*ZMP*). Cuando el *ZMP* existe dentro del dominio de la superficie de soporte, el contacto entre el suelo y la pierna de soporte es estable:

$$p_{ZMP} = (x_{ZMP}, y_{ZMP}, 0) \in S,$$

Donde  $p_{ZMP}$  denota la posición del *ZMP* y  $S$  denota el dominio de la superficie de soporte. Por lo tanto, ésta condición indica que no ocurre ninguna rotación alrededor de los bordes del pie.

## Capítulo 2

### Descripción del modelo

En éste capítulo se detallará el modelo cinemático del robot bípedo Bioloid GP® para determinar la ubicación de los efectores finales, en este caso los pies y cadera, con respecto al sistema de referencia global considerando todos los desfases de componentes rígidos fuera del eje de acción del robot. También se explica como el modelado dinámico se simplificó al de un péndulo invertido móvil donde la masa concentrada se localiza en el centro de masa (por sus siglas en inglés, *CoM*) del robot para disminuir el costo computacional, dicho *CoM* es calculado por medio de la ponderación de las masas de cada eslabón tomando en cuenta sus respectivas transformaciones con respecto al sistema inercial.

De igual manera, se hace una breve descripción a los algoritmos genéticos, consideraciones y operadores como introducción al algoritmo de optimización de trayectorias desarrollado. Por último, en el cuarto subcapítulo se describen las características del modelo de pruebas, el antes mencionado Bioloid GP®, así como las respectivas modificaciones hechas en el hardware para la implementación de las pruebas.

#### 2.1 Modelo de cinemática directa

En [21], la formulación de la cinemática directa puede definirse de ésta manera, “*Dadas las variables articulares del robot, se determina la posición y orientación del efector final. Donde las variables articulares son los ángulos entre los eslabones en caso de juntas rotacionales o de revoluta, y la extensión del eslabón en caso de una junta prismática o corredera*”, donde por propósitos del análisis cinemático se considera al robot como un conjunto de objetos rígidos conectados por varias juntas o articulaciones.

El objetivo del análisis cinemático directo es el de determinar el efecto acumulado del conjunto completo de articulaciones y para esto se definirán ciertos conceptos útiles para la representación de los sistemas de referencia de cada eslabón del robot. Supóngase un robot con  $n+1$  eslabones numerados del 0 a  $n$  empezando por la base del robot, la cual se le denomina como eslabón 0 y las juntas son numeradas del 1 al  $n$ , y la  $i$ -ésima junta ( $q_i$ ) es el punto en el espacio donde los eslabones  $i-1$  e  $i$  se conectan. En caso de una junta rotacional,  $q_i$  es el ángulo de rotación, y en el caso de la junta prismática,  $q_i$  es el desplazamiento. Un sistema de coordenadas le es asignado a cada eslabón, empezando con el sistema 0 o también conocido como sistema inercial y luego se asignan sistemas del 1 hasta  $n$  definiendo dicho  $i$ -ésimo sistema con el  $i$ -ésimo eslabón. Ahora supongamos  $A_i$  es la matriz homogénea que transforma las coordenadas del punto en el sistema  $i$  al sistema  $i-1$  y varía según la configuración del robot cambia, de igual manera, se asume que todas las juntas están en función de una sola variable articular, denominada  $q_i$ .

## CAPÍTULO 2 DESCRIPCIÓN DEL MODELO

En otras palabras,

$$A_i = A_i(q_i) \quad (2.1)$$

Ahora la matriz homogénea que transforma las coordenadas de un punto del sistema  $j$  al sistema  $i$  se le conoce como matriz de transformación, y se denota  $T_i^j$ .

$$T_i^j = A_{i+1} A_{i+2} \dots A_{j-1} \quad \text{si } i < j \quad (2.2)$$

$$T_i^j = I \quad \text{si } i = j \quad (2.3)$$

De manera en la cual se tienen asignados rígidamente los sistemas de referencia a sus correspondientes eslabones, sucede que la posición de cualquier punto en el efector final, cuando está representado en el sistema  $n$ , es una constante independiente de la configuración del robot. Sin embargo, la posición y orientación del efector final con respecto al sistema inercial está dado por un vector en  $\mathbb{R}^3$ ,  $\mathbf{d}_0^n$  y una matriz de rotación de  $3 \times 3$ ,  $R_0^n$ , respectivamente definiendo la matriz homogénea  $H$  que ilustra la posición y orientación del efector final en el sistema inercial.

$$H = T_0^n = \begin{bmatrix} R_0^n & \mathbf{d}_0^n \\ 0 & 1 \end{bmatrix} = A_1(q_1) \dots A_n(q_n) \quad (2.4)$$

Así mismo, cada transformación homogénea  $A_i$ , es de la forma:

$$A_i = \begin{bmatrix} R_{i-1}^i & \mathbf{d}_{i-1}^i \\ 0 & 1 \end{bmatrix} \quad (2.5)$$

Por lo tanto,

$$T_i^j = A_{i+1} \dots A_j = \begin{bmatrix} R_i^j & \mathbf{d}_i^j \\ 0 & 1 \end{bmatrix} \quad (2.6)$$

Donde la matriz  $R_i^j$  expresa la orientación del sistema  $j$  relativo al sistema  $i$  y es dado por las partes rotacionales de las matrices  $A$  como:

$$R_i^j = R_i^{i+1} \dots R_{j-1}^j \quad (2.7)$$

Y los vectores  $\mathbf{d}_i^j$  se dan recursivamente por la fórmula:

$$\mathbf{d}_i^j = \mathbf{d}_i^{j-1} + R_i^{j-1} \mathbf{d}_{j-1}^j \quad (2.8)$$

Sabiendo que es posible hacer el análisis usando sistemas arbitrarios asignados a cada eslabón, es más sencillo llevar a cabo una selección sistemática de los marcos de referencia, por lo que se optó por usar la convención de *Denavit-Hartenberg* o *D-H*. En ésta convención cada transformación homogénea  $A_i$  se representa como un producto de cuatro transformaciones básicas:

$$A_i = Rot_{z,\theta_i} \cdot Trans_{z,d_i} \cdot Trans_{x,a_i} \cdot Rot_{x,\alpha_i} \quad (2.9)$$

$$= \begin{bmatrix} c_{\theta_i} & -s_{\theta_i} & 0 & 0 \\ s_{\theta_i} & c_{\theta_i} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 & a_i \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & c_{\alpha_i} & -s_{\alpha_i} & 0 \\ 0 & s_{\alpha_i} & c_{\alpha_i} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.10)$$

$$= \begin{bmatrix} c_{\theta_i} & -s_{\theta_i} c_{\alpha_i} & s_{\theta_i} s_{\alpha_i} & a_i c_{\theta_i} \\ s_{\theta_i} & c_{\theta_i} c_{\alpha_i} & -c_{\theta_i} s_{\alpha_i} & a_i s_{\theta_i} \\ 0 & s_{\alpha_i} & c_{\alpha_i} & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.11)$$

Donde las cuatro cantidades  $\theta_i$ ,  $a_i$ ,  $d_i$ ,  $\alpha_i$  son parámetros del  $i$ -ésimo eslabón y junta. Ahora considerando que  $A_i$  es la transformación homogénea entre dos sistemas de referencia, por ejemplo, 0 y 1, el parámetro  $a$  es la distancia entre los ejes  $z_0$  y  $z_1$ , medido a lo largo del eje  $x_1$ . El ángulo  $\alpha$  es el ángulo entre los ejes  $z_0$  y  $z_1$  medido en un plano normal a  $x_1$ . El sentido positivo de  $\alpha$  es determinado de  $z_0$  a  $z_1$  por la regla de la mano derecha. El parámetro  $d$  es la distancia entre el origen  $o_0$  y la intersección del eje  $x_1$  con  $z_0$  medida a lo largo de  $z_0$ . Finalmente,  $\theta$  es el ángulo entre  $x_0$  y  $x_1$  medido en el plano normal al eje  $z_0$ .

De igual forma como se definió la matriz homogénea de transformación entre el sistema 0 y 1, ésta convención se utiliza para  $n$  cuadros de referencia, en [21] se explica con detalle el procedimiento de la convención  $D-H$  para derivar la cinemática directa para cualquier manipulador. De esa forma se generó el modelo cinemático directo del robot Bioloid GP® que empieza con un sistema de referencia inercial fija sobre el suelo que se conecta con la cadera a través de la transformación definida en la ecuación (2.12) dando una simple translación sobre los 3 ejes de referencia para localizar el primer punto del robot en el espacio donde  $x_c$ ,  $y_c$  y  $z_c$  son el desplazamiento en sentido de los ejes coordenados de la cadera con respecto al piso y partir de ahí para derivar las ecuaciones de las dos cadenas cinemáticas abiertas que son las piernas.

$$C_{cad} = \begin{bmatrix} 1 & 0 & 0 & x_c \\ 0 & 1 & 0 & y_c \\ 0 & 0 & 1 & z_c \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.12)$$

Al estar utilizando el método de  $D-H$ , fue relativamente sencillo derivar las matrices de transformación desde la cadera hasta el efector final, en éste caso el pie, y sabiendo que las piernas eran completamente simétricas simplemente bastó con obtener aquellas de la primera pierna y después cambiar únicamente ciertos signos en los parámetros angulares dependiendo de la orientación del sistema de referencia. Se partió entonces de la consideración que el sistema cero se encuentra en la cadera en vez de lo que normalmente es el piso para facilitar la nomenclatura y es así que las transformaciones de la pierna derecha se denominaron  $D^j_i$  y aquellas de la pierna izquierda se denominaron  $I^j_i$ :

CAPÍTULO 2 DESCRIPCIÓN DEL MODELO

$$D_0^1 = T_{DH}(-90^\circ, 0, d_0, -90^\circ)$$

$$D_1^2 = T_{DH}(\theta_{d1}, 0, d_1, 0^\circ)$$

$$D_2^3 = T_{DH}(90^\circ, 0, d_2, 90^\circ)$$

$$D_3^4 = T_{DH}(\theta_{d3}, 0, d_3, 0^\circ)$$

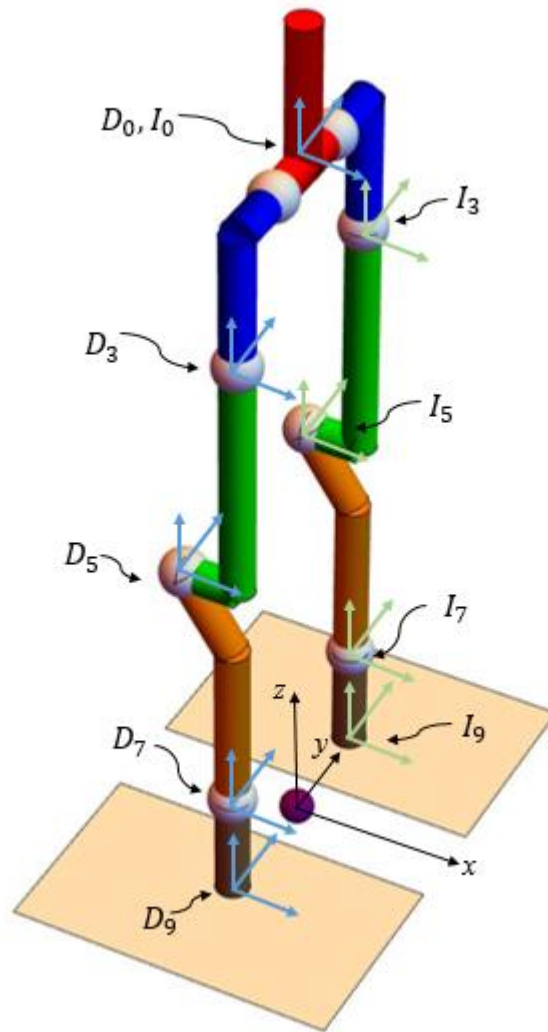
$$D_4^5 = T_{DH}(-90^\circ, 0, d_4, 0^\circ)$$

$$D_5^6 = T_{DH}(\theta_{d5} + 123.8^\circ, 0, d_5, 0^\circ)$$

$$D_6^7 = T_{DH}(-33.8^\circ, 0, d_6, 0^\circ)$$

$$D_7^8 = T_{DH}(\theta_{d7}, 0, 0, -90^\circ)$$

$$D_8^9 = T_{DH}(\theta_{d8}, 0, d_8, 0^\circ)$$



**Figura 2.1** Marcos de referencia generados a partir del modelo de cinemática del robot Bioloid GP®.



$$\begin{aligned}
 I_0^1 &= T_{DH}(90^\circ, 0, i_0, 90^\circ) \\
 I_1^2 &= T_{DH}(\theta_{i1}, 0, i_1, 0^\circ) \\
 I_2^3 &= T_{DH}(-90^\circ, 0, i_2, -90^\circ) \\
 I_3^4 &= T_{DH}(\theta_{i3}, 0, i_3, 0^\circ) \\
 I_4^5 &= T_{DH}(90^\circ, 0, i_4, 0^\circ) \\
 I_5^6 &= T_{DH}(\theta_{i5} - 123.8^\circ, 0, i_5, 0^\circ) \\
 I_6^7 &= T_{DH}(33.8^\circ, 0, i_6, 0^\circ) \\
 I_7^8 &= T_{DH}(\theta_{i7}, 0, 0, 90^\circ) \\
 I_8^9 &= T_{DH}(\theta_{i8}, 0, i_8, 0^\circ)
 \end{aligned}$$

Donde la función  $T_{DH}(\theta_i, a_i, d_i, \alpha_i)$  es la transformación homogénea definida en la ecuación (2.9) con sus respectivos parámetros para definir cada uno de los eslabones de ambas cadenas cinemáticas y de esa manera las transformaciones que relacionan las coordenadas de los efectores finales izquierdo y derecho con el sistema inercial  $c$  están definidas así:

$$D_C^9 = C_{cad} D_0^1 D_1^2 D_2^3 D_3^4 D_4^5 D_5^6 D_6^7 D_7^8 D_8^9 \quad (2.13)$$

$$I_C^9 = C_{cad} I_0^1 I_1^2 I_2^3 I_3^4 I_4^5 I_5^6 I_6^7 I_7^8 I_8^9 \quad (2.14)$$

## 2.2 Modelo dinámico simplificado

Con el objetivo de simplificar y reducir tanto como fuera posible el costo computacional para el desarrollo de la marcha bípeda en el robot de 10 GdL, se optó por utilizar un modelo simplificado de péndulo invertido en vez de desarrollar las ecuaciones dinámicas por métodos más extensivos como el de Newton-Euler o el de Euler-Lagrange. El modelo consiste en una masa concentrada en el  $CoM$  del robot conectada directamente por un segmento virtual hasta el suelo asumiendo que existe suficiente fricción entre el pie y el suelo para evitar cualquier deslizamiento, de esta manera se aproxima la dinámica del robot durante la fase de soporte simple.

Éste modelo es útil para poder determinar la posición del  $ZMP$  en todo momento dadas las trayectorias de posición, velocidad y aceleración del  $CoM$  del robot, en [22] se desarrollaron dichas ecuaciones partiendo de la ecuación del equilibrio rotacional, y al no estar considerando el número el número de grados de libertad del manipulador sino únicamente los pares creados por las fuerzas de reacción del suelo, los momentos en los tres ejes sobre el centro de masa y la fuerza de gravedad fue posible relacionar la posición del  $ZMP$  directamente con la posición y aceleración del  $CoM$  del robot determinando las siguientes expresiones:

$$x_{zmp} = \frac{z \ddot{x} - x \ddot{z}}{g} + x \quad (2.15)$$

$$y_{zmp} = \frac{y \ddot{z} - z \ddot{y}}{g} + y \quad (2.16)$$

## CAPÍTULO 2 DESCRIPCIÓN DEL MODELO

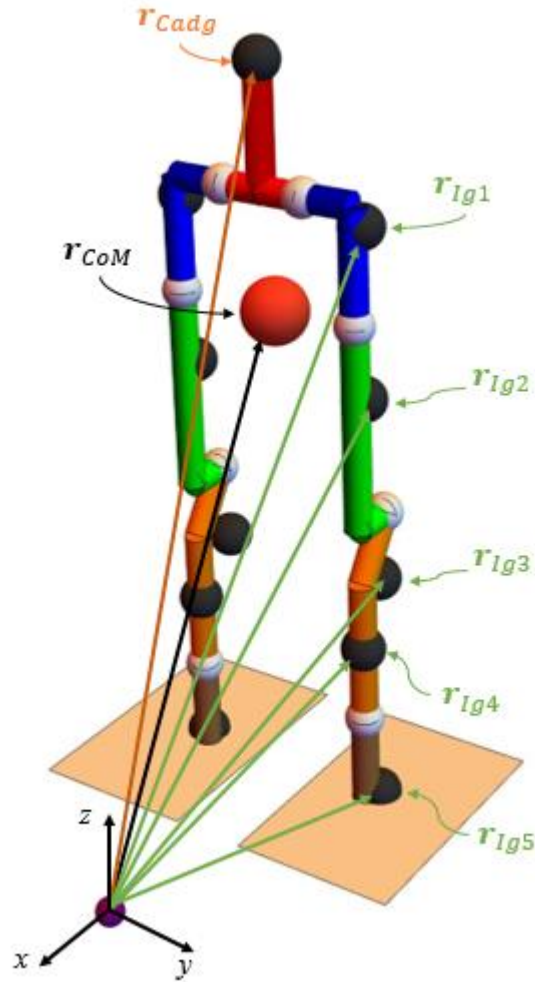
Dado que es un mecanismo en movimiento implica que la posición de dicho *CoM* variará según la configuración de las articulaciones del robot y por esa razón es necesario calcularlo en cada instante para poder determinar si el *ZMP* se encuentra o no en el polígono de soporte. Para esto se implementó una expresión encontrada en [23], la cual estima por medio de un promedio ponderado de las masas relativas de cada eslabón con respecto a la total la localización del centro de masa en coordenadas del sistema inercial.

$$\mathbf{r}_{CoM} = \sum_{i=1}^{11} \frac{m_i}{m_T} \mathbf{r}_i \quad (2.17)$$

Donde el vector  $\mathbf{r}_{CoM}$  denota la posición del centro de masa del robot,  $\mathbf{r}_i$  y  $m_i$ , la posición y masa del centro de gravedad de cada eslabón respectivamente y  $m_T$ , la masa acumulada del manipulador. El número total de eslabones enunciados es de nueve, debido a que el primero es considerado el torso completo del robot con las extremidades superiores, luego del 2 al 5 se tienen los eslabones de la pierna derecha y finalmente, los últimos cuatro hacen referencia a los eslabones de la pierna izquierda. Para el cómputo de las transformaciones de los centros de masa con respecto al sistema inercial se reutilizaron las ecuaciones resueltas en la sección anterior para posteriormente añadir los desplazamientos correspondientes que parten de determinada junta hasta el punto donde se localiza el centro de gravedad, la determinación de la ubicación de los centros de masa se hizo experimentalmente y se explicará a detalle en el subcapítulo 2.4.

$$\begin{aligned} \mathbf{r}_{Cadg} &= C_{cad} \cdot T_{xyz}(x_{cg}, y_{cg}, z_{cg}) \cdot \mathbf{n} \\ \mathbf{r}_{Dg1} &= D_C^2 \cdot T_{xyz}(x_{dg1}, y_{dg1}, -z_{dg1}) \cdot \mathbf{n} \\ \mathbf{r}_{Dg2} &= D_C^4 \cdot T_{xyz}(-x_{dg2}, -y_{dg2}, z_{dg2}) \cdot \mathbf{n} \\ \mathbf{r}_{Dg3} &= D_C^7 \cdot T_{xyz}(-x_{dg3}, y_{dg3}, z_{dg3}) \cdot \mathbf{n} \\ \mathbf{r}_{Dg4} &= D_C^8 \cdot T_{xyz}(-x_{dg4}, y_{dg4}, -z_{dg4}) \cdot \mathbf{n} \\ \mathbf{r}_{Dg5} &= D_C^9 \cdot T_{xyz}(-x_{dg5}, -y_{dg5}, -z_{dg5}) \cdot \mathbf{n} \\ \mathbf{r}_{Ig1} &= I_C^2 \cdot T_{xyz}(x_{ig1}, -y_{ig1}, -z_{ig1}) \cdot \mathbf{n} \\ \mathbf{r}_{Ig2} &= I_C^4 \cdot T_{xyz}(-x_{ig2}, y_{ig2}, z_{ig2}) \cdot \mathbf{n} \\ \mathbf{r}_{Ig3} &= I_C^7 \cdot T_{xyz}(-x_{ig3}, y_{ig3}, z_{ig3}) \cdot \mathbf{n} \\ \mathbf{r}_{Ig4} &= I_C^8 \cdot T_{xyz}(-x_{ig4}, -y_{ig4}, -z_{ig4}) \cdot \mathbf{n} \\ \mathbf{r}_{Ig5} &= I_C^9 \cdot T_{xyz}(-x_{ig5}, y_{ig5}, -z_{ig5}) \cdot \mathbf{n} \end{aligned}$$

Y así quedan definidos los vectores de posición del centro de masa en coordenadas homogéneas de cada uno de los 11 eslabones considerados en el modelo, donde el vector  $\mathbf{n} = [0 \ 0 \ 0 \ 1]^T$  y la función  $T_{xyz}(x, y, z)$  está definida como una translación simple sobre los 3 ejes coordenados, refiérase a la Figura 2.2 para la visualización de los centros de masa de cada eslabón y el *CoM* global del robot.



**Figura 2.2** Vectores de posición de los centros de masa de cada eslabón del Bioloïd GP®.

### 2.3 Algoritmos genéticos

Los algoritmos genéticos (AGs) son métodos adaptativos de búsqueda y optimización basados en el principio de selección natural de Darwin, “Supervivencia del más apto”, y la ley genética de la transferencia de factores hereditarios de padres a descendientes de Mendel. Refiriéndose a como una población dada de individuos compite entre sí en la búsqueda de comida, agua, refugio e incluso pareja, y aquellos individuos que tienen más éxito para sobrevivir tienen mayor probabilidad de generar un gran número de descendientes propagando sus genes hacia las futuras generaciones. Los principios básicos de los AGs fueron establecidos por Holland (1975), y éstos hacen una analogía directa con el comportamiento de la naturaleza en el sentido que trabajan con una población de individuos, donde cada uno de dichos individuos representa una solución factible a un problema dado. A cada individuo se le asigna un valor o calificación relacionado a su capacidad de resolver el problema. Cuanto mayor sea el valor o adaptación del individuo mayor será la probabilidad de ser seleccionado para reproducirse, cruzando material genético con otro individuo seleccionado de igual forma. Esto permite la generación de nuevos individuos con características de sus padres dando la oportunidad de que aquellos genes de individuos mejor adaptados prevalezcan en sucesivas generaciones evitando simultáneamente que el material genético de individuos menos adaptados se siga propagando.

## CAPÍTULO 2 DESCRIPCIÓN DEL MODELO

Un algoritmo genético comienza con una población aleatoria, y ésta necesita una codificación adecuada al problema junto con una función de evaluación que asigne un valor real a cada posible solución generada. En el desarrollo del algoritmo, los individuos serán seleccionados para la reproducción donde los denominados padres se cruzarán generando hijos, sobre los cuales actuará un operador de mutación posteriormente. El resultado de la combinación de las funciones mencionadas dará lugar a un nuevo conjunto de soluciones a una siguiente población [24].

Los términos definidos por varios autores como Goldberg (1975), Michalewicz (1996), Bäck (1996) y otros, para los algoritmos genéticos partieron por definir una población  $\mathcal{P}$  conformada por  $\mu$  número de elementos o individuos  $c$ .

$$\mathcal{P} = \{c_1, \dots, c_i, \dots, c_\mu\}$$

Dichos individuos o posibles soluciones pueden representarse como un conjunto de parámetros, llamados *genes*, los cuales agrupados forman una ristra de valores. Ésta ristra se le denomina  *cromosoma*, y puede ser representada con distintos alfabetos por esa razón existe una distinción entre  *fenotipo* (conjunto en el espacio solución) y  *genotipo* (cromosoma en el espacio de búsqueda), sin embargo, las representaciones más utilizadas son la binaria y la real. Otro importante parámetro para la evaluación de los individuos es la función de aptitud  $F(c_i)$ , diseñada para cada problema en específico otorgando un valor real según el nivel de adaptación o aptitud del individuo, donde  $c_i$  se constituye por un  $n$  número de variables  $c_{ij}$  que representan a sus genes. También como medida de evaluación de la población se utiliza la aptitud media  $F_m$  definida.

$$F_m = \frac{1}{\mu} \sum_{i=1}^{\mu} F(c_i) \quad (2.18)$$

Y de igual manera dependiendo del tipo de operadores genéticos deseados se utiliza la aptitud relativa  $p_i$ .

$$p_i = \frac{F(c_i)}{\sum_{i=1}^{\mu} F(c_i)} \quad (2.19)$$

Como principales operadores genéticos existen la selección, cruce y mutación, los cuales tienen sus respectivas variaciones dependiendo del problema que se desee resolver. El operador de *selección* o también conocido reproducción puede ser implementado de diferentes maneras, probablemente la más fácil es la de crear una ruleta ponderada o pesada donde cada individuo de la población tenga un lugar en la ruleta de tamaño proporcional a su aptitud, este tipo de selección se conoce como selección proporcional o de ruleta y enuncia que la probabilidad de que  $c_i$  avance a la siguiente generación es proporcional a su aptitud relativa  $p_i$  y el número de descendientes se define por  $\xi$ .

$$\xi = p_i \cdot \mu \quad (2.20)$$

Otro tipo común de selección es la de torneo. Aquí, el individuo más apto escogido de un grupo de  $z$  individuos de la población avanza a la siguiente generación. Éste proceso es repetido  $\mu$  veces, y en promedio, el ganador de un torneo más grande tiene mejor aptitud que el ganador de un torneo pequeño, y por esa razón es muy importante tomar en cuenta la presión de selección, definida como el grado en el cual los mejores individuos son favorecidos. Si la presión de selección de un AG sea muy alta puede causar convergencia rápida al mínimo local y en el caso opuesto causaría lenta convergencia. Es por eso que en la selección de torneo la presión de selección se puede modificar según el tamaño del torneo  $z$ .

## CAPÍTULO 2 DESCRIPCIÓN DEL MODELO

En el proceso de selección, solo fueron insertadas copias idénticas de los individuos, sin embargo, en el *cruce* se genera nuevo material genético intercambiándolo entre individuos de la población, por lo tanto, nuevos individuos emergen. El cruce de dos individuos se lleva a cabo por la probabilidad de cruce,  $p_c$ , la cual es definida previa al proceso de optimización. La operación de cruce como tal lleva lugar de la siguiente manera, primero, un número aleatorio entre cero y uno es generado. Si el número es más pequeño que la probabilidad de cruce, entonces los cromosomas de los individuos escogidos son divididos por un punto de cruce. El punto de cruce determina como el material genético de los nuevos individuos será compuesto. Por cada par de individuos, el punto de cruce es determinado de nuevo. En cada generación,  $\mu \cdot p_c / 2$  pares son escogidos aleatoriamente y cruzados.

Cuando los individuos de la población son representados por número binarios, los cromosomas son divididos por el punto de cruce  $i$  en dos partes y después son nuevamente reconstituidos. Dados dos viejos individuos (padres):

$$\begin{aligned} c_1 &= [c_{1,1}, c_{1,2}, \dots, c_{1,i}, c_{1,i+1}, \dots, c_{1,n}] \\ c_2 &= [c_{2,1}, c_{2,2}, \dots, c_{2,i}, c_{2,i+1}, \dots, c_{2,n}] \end{aligned}$$

Y aplicándoles la operación de cruce sobre el punto  $i$ , dos nuevos individuos (hijos o descendientes) resultan:

$$\begin{aligned} c_1^{nuevo} &= [c_{1,1}, c_{1,2}, \dots, c_{1,i}, c_{2,i+1}, \dots, c_{2,n}] \\ c_2^{nuevo} &= [c_{2,1}, c_{2,2}, \dots, c_{2,i}, c_{1,i+1}, \dots, c_{1,n}] \end{aligned}$$

Ésta descripción ilustra un solo punto de cruce, sin embargo, la extensión a dos puntos o multipuntos de cruce, así como el uso de múltiples individuos fue estudiada por Syswerda (1989,1993). Y determina que el desempeño usando cruce multipuntos o incluso con varios individuos depende del problema en cuestión.

En representación real, el individuo consiste en un vector de números reales. Por ejemplo, dados  $c_1$  y  $c_2$  como dos individuos capaces de reproducirse. Los dos descendientes  $c_1^{nuevo}$  y  $c_2^{nuevo}$  son producto de la combinación lineal de sus padres  $c_1$  y  $c_2$ :

$$c_1^{nuevo} = \lambda c_1 + (1 - \lambda)c_2 \quad (2.21)$$

$$c_2^{nuevo} = (1 - \lambda) c_1 + \lambda c_2 \quad (2.22)$$

Donde  $\lambda \in [0,1]$  es el parámetro de cruce. Como último operador, el proceso de mutación se efectúa como una variación aleatoria sobre algún gen de cierto individuo. Una mutación es ejecutada bajo el mismo criterio que el cruce, pero ésta tiene un parámetro independiente para determinar si se ejecuta o no la operación y se le denomina probabilidad de mutación  $p_m$ . El operador de mutación aplicado en representación binaria selecciona aleatoriamente un bit de un gen y lo invierte convirtiendo un valor de 0 en 1 y viceversa. Aunque el operador de mutación fue desarrollado para la representación binaria, otros métodos han sido desarrollados para permitir la modificación de genes en representación real. Estos métodos aplican una distribución de probabilidad, que se define sobre el dominio de los posibles valores para cada gen.

Dado el individuo  $c_i = [c_{i1}, \dots, c_{ij}, \dots, c_{in}]$  y  $c_{ij}$  el gen seleccionado para la mutación. El dominio de la variable  $c_{ij}$  es dado por  $c_{ij} = [c_{ij,min}, c_{ij,max}]$ , donde  $c_{ij,min}$  y  $c_{ij,max}$  denotan las fronteras superior e inferior respectivamente. Mutaciones en números reales son de dos tipos: mutación uniforme y mutación no uniforme.

En la mutación uniforme, el individuo resultante de la aplicación del operador es  $c_i^{nuevo} = [c_{i1}, \dots, \tilde{c}_{ij}, \dots, c_{in}]$  donde como su nombre lo dice  $\tilde{c}_{ij}$  es un valor aleatorio (proveniente de una distribución de probabilidad uniforme) dentro del dominio de  $c_{ij}$ . Y el operador de mutación es aplicado con una probabilidad  $p_m$ . Por el contrario, en la mutación no uniforme, resulta un individuo con la misma estructura que  $c_i^{nuevo}$  donde el  $\tilde{c}_{ij}$  es calculado por:

$$\tilde{c}_{ij} = \begin{cases} c_{ij} + \ddot{A}(g, c_{ij,max} - c_{ij}), & \text{si } h = 0 \\ c_{ij} + \ddot{A}(g, c_{ij} - c_{ij,min}), & \text{si } h = 1 \end{cases} \quad (2.23)$$

Donde  $h$  es un dígito binario escogido aleatoriamente y la función  $\ddot{A}(g, y)$  retorna un valor en el intervalo  $[0, y]$  tal que la probabilidad de  $\ddot{A}(g, y)$  inicia en cero y se incrementa con el número de generaciones  $g$  tal que:

$$\ddot{A}(g, y) = y \cdot r_a \cdot \left[1 - \frac{g}{g_{max}}\right]^b \quad (2.24)$$

El parámetro  $r_a$  es un número seleccionado aleatoriamente en el intervalo  $[0, 1]$ , y  $b$  es un parámetro de diseño que determina el grado de dependencia del número de generaciones.

---

**Algoritmo 1: algoritmo genético simple**

**Entradas:**  $F(c_i), p_c, p_m, \mu$

**Salidas:**  $c_i$

**Variables auxiliares:**  $g, g_{max}$

**Inicio**

$g = 0$

inicializa:  $P_g = \{c_1, \dots, c_\mu\}$

**mientras** ( $g \leq g_{max}$ )

    calcular aptitud:  $F(c_i)$

    selección:  $P_g \Rightarrow \tilde{P}$

    cruce:  $\tilde{P} \Rightarrow \tilde{\tilde{P}}$

    mutación:  $\tilde{\tilde{P}} \Rightarrow P_{g+1}$

**fin mientras**

**retornar**  $c_i$

---

*Algoritmo 2.1 AG simple para optimización [26].*

En la Algoritmo 2.1 se muestra un AG que resuelve una optimización no restringida. Inicialmente, se define la función de aptitud  $F(c_i)$ , y tanto las probabilidades  $p_c$  y  $p_m$  como el tamaño de la población  $\mu$  son escogidos. Los individuos de la población inicial  $P_0$ , son aleatoriamente inicializados y se continúa calculando la aptitud de dicha población inicial. Después, aplicando la selección a la población  $P_g$  resulta en una población de transición  $\tilde{P}$ . De la aplicación de cruce con la probabilidad  $p_c$ , resulta otra transición de  $\tilde{P}$  a la población  $\tilde{\tilde{P}}$ . Y al operar el proceso de mutación con la probabilidad  $p_m$  a los individuos de la población  $\tilde{\tilde{P}}$ , termina en el origen de una nueva población designada como  $P_{g+1}$ . Como parte del ciclo del algoritmo, si el máximo número de generaciones  $g_{max}$  no es alcanzado, entonces se vuelve a calcular la aptitud y a aplicar los operadores genéticos hasta que  $g = g_{max}$  la optimización culmina, teniendo al individuo más apto como solución del problema de optimización.

## 2.4 Modelo de pruebas

Para la parte de implementación real y pruebas del algoritmo de optimización, se optó por utilizar una plataforma robótica educativa comercial que contara con los grados de libertad necesarios y de esa manera desprejar los posibles errores en diseño y potenciar la funcionalidad. El Departamento de Ingeniería Mecatrónica de la U.N.A.M. proveyó el robot Bioloid GP® de la marca Robotis en el cual se efectuaron sus respectivas modificaciones para cumplir con los objetivos planteados.

### 2.4.1 Bioloid GP ®

El Bioloid GP® o Robotis GP® es un robot de alto rendimiento equipado para carreras, fútbol, peleas y otro tipo de actividades competitivas. Éste robot de primera categoría está ensamblado con marcos ultra ligeros de aluminio, cuenta con gran movilidad incluyendo cambio de dirección mientras camina, dentro de su firmware contiene movimientos precargados para jugar fútbol y batalla humanoide, además es capaz de ajustar su postura con su giroscopio. Dentro de sus componentes se tienen 18 actuadores de los cuales se tienen 8 Dynamixel® AX-12A para los miembros superiores y el resto son actuadores Dynamixel® AX-18A para completar los 10 GdL de las extremidades inferiores. El controlador principal es el CM-530 desarrollado por la misma empresa Robotis embebido con un procesador ARM Cortex M3 de 32 bits. Incluye también un módulo de comunicación inalámbrica ZIG-110A que opera sobre un protocolo de comunicación IEEE 802.15.4 (*Zig-bee*), su respectivo control remoto RC-100A, una batería LIPO de 11.1V, un sensor de distancia (*DMS*) y un giroscopio [11].

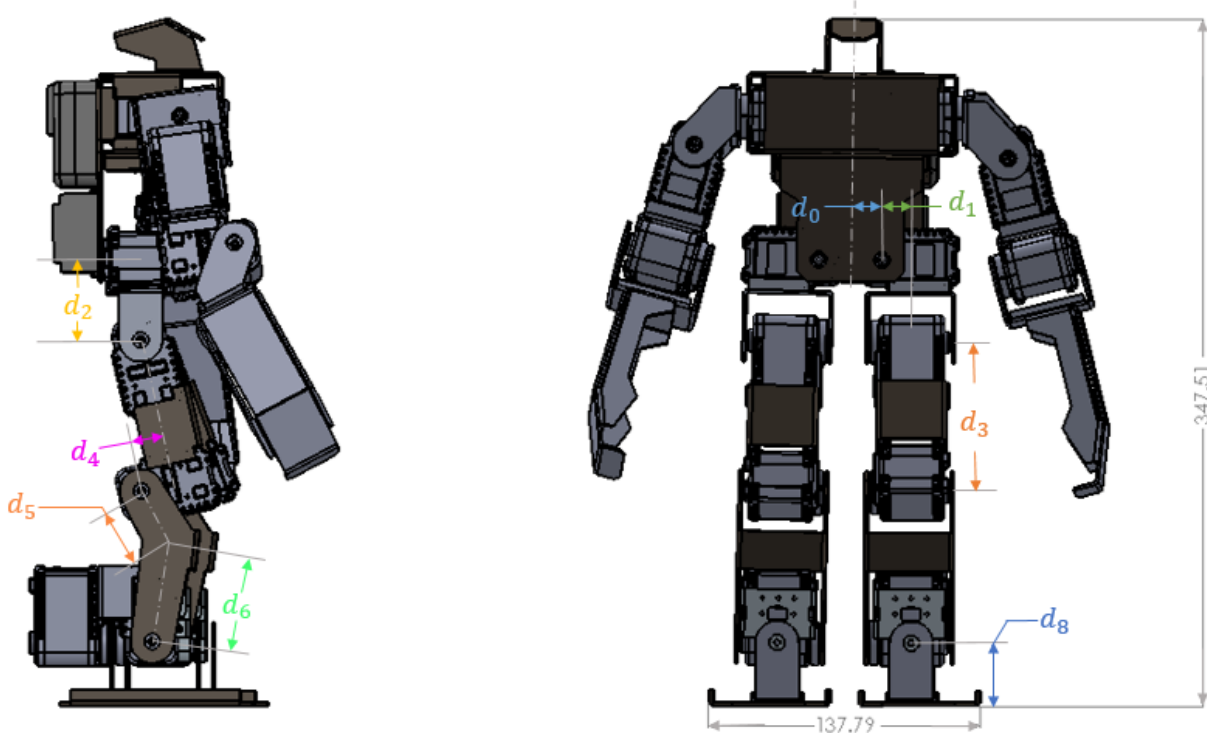


Figura 2.4 Dimensiones del Robotis Bioloid GP ®.

## CAPÍTULO 2 DESCRIPCIÓN DEL MODELO

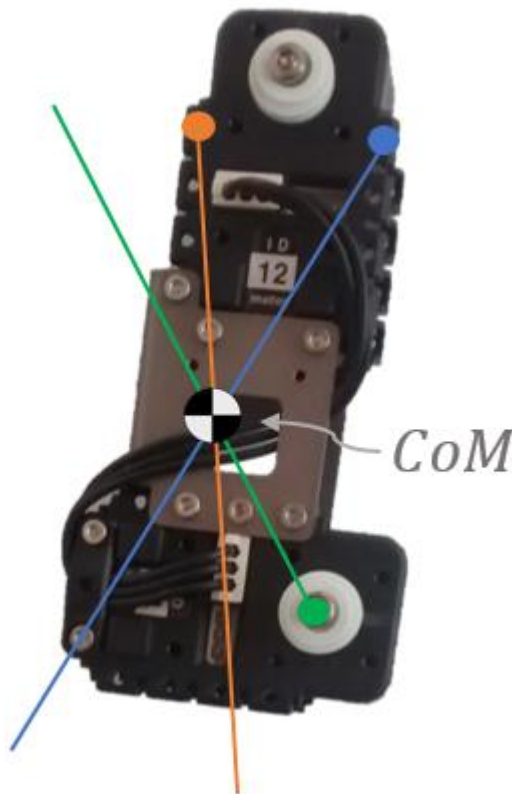
En la Figura 2.4 se muestran las dimensiones del modelo de pruebas, así como las medidas  $d_n$  que hacen referencia a las distancias de los eslabones utilizados en el modelo de cinemática directa. En la figura únicamente se ilustran las medidas  $d_n$  por cuestiones de simetría, dejando los siguientes parámetros acotados obtenidos del modelo en 3D (archivo de CAD) en milímetros:

$$\begin{aligned}d_0 &= i_0 = 16 \text{ mm} \\d_1 &= i_1 = 19.36 \text{ mm} \\d_2 &= i_2 = 41.09 \text{ mm} \\d_3 &= i_3 = 74.28 \text{ mm} \\d_4 &= i_4 = 16.12 \text{ mm} \\d_5 &= i_5 = 26.06 \text{ mm} \\d_6 &= i_6 = 53.03 \text{ mm} \\d_8 &= i_8 = 32.7 \text{ mm}\end{aligned}$$

De igual forma se determinó la altura de la cadera,

$$z_c = 220.8 \text{ mm}$$

Y finalmente se obtuvieron los valores de masa de los eslabones y sus centros de gravedad de manera experimental. Tal y como se muestra en la Figura 2.5, se desarmó el robot en cada uno de los



**Figura 2.5** Ejemplificación del método experimental para obtener los centros de masa locales de cada eslabón, los círculos de color representan los distintos puntos por los cuales es sujeta la pieza y las líneas son las marcas por donde pasa la referencia vertical.

distintos eslabones del modelo y primero, se pesó independientemente cada elemento para después determinar su respectivo centro de masa. El experimento consistió en suspender el eslabón en el aire, sujetado por un elemento rígido que conectaba dos puntos paralelos y con una referencia vertical (hilo con



## CAPÍTULO 2 DESCRIPCIÓN DEL MODELO

masa suspendida) marcar la trayectoria por la cual dividía al eslabón en dos partes. Este procedimiento se repetía 3 veces para determinar el punto de cruce entre las trayectorias y definir la proyección del *CoM* sobre ese plano. Este experimento se realizó en dos distintos planos del eslabón para poder definir completamente el *CoM* en el espacio, además cabe resaltar que no era posible determinar dicha posición de los *CoM* por medio del archivo de CAD, puesto que, dicho archivo no consideraba la masa no uniformemente distribuida del eslabón causada por la existencia de distintos materiales dentro del servo motor. También, se tomó en cuenta error inherente de la medición debido a la geometría irregular de los eslabones dentro del análisis de resultados. La localización de los *CoM* del Bioloid GP ® se ilustran en la Figura 2.6.

Como resultado de la medición experimental de la ubicación de los centros de masa se obtuvo lo siguiente:

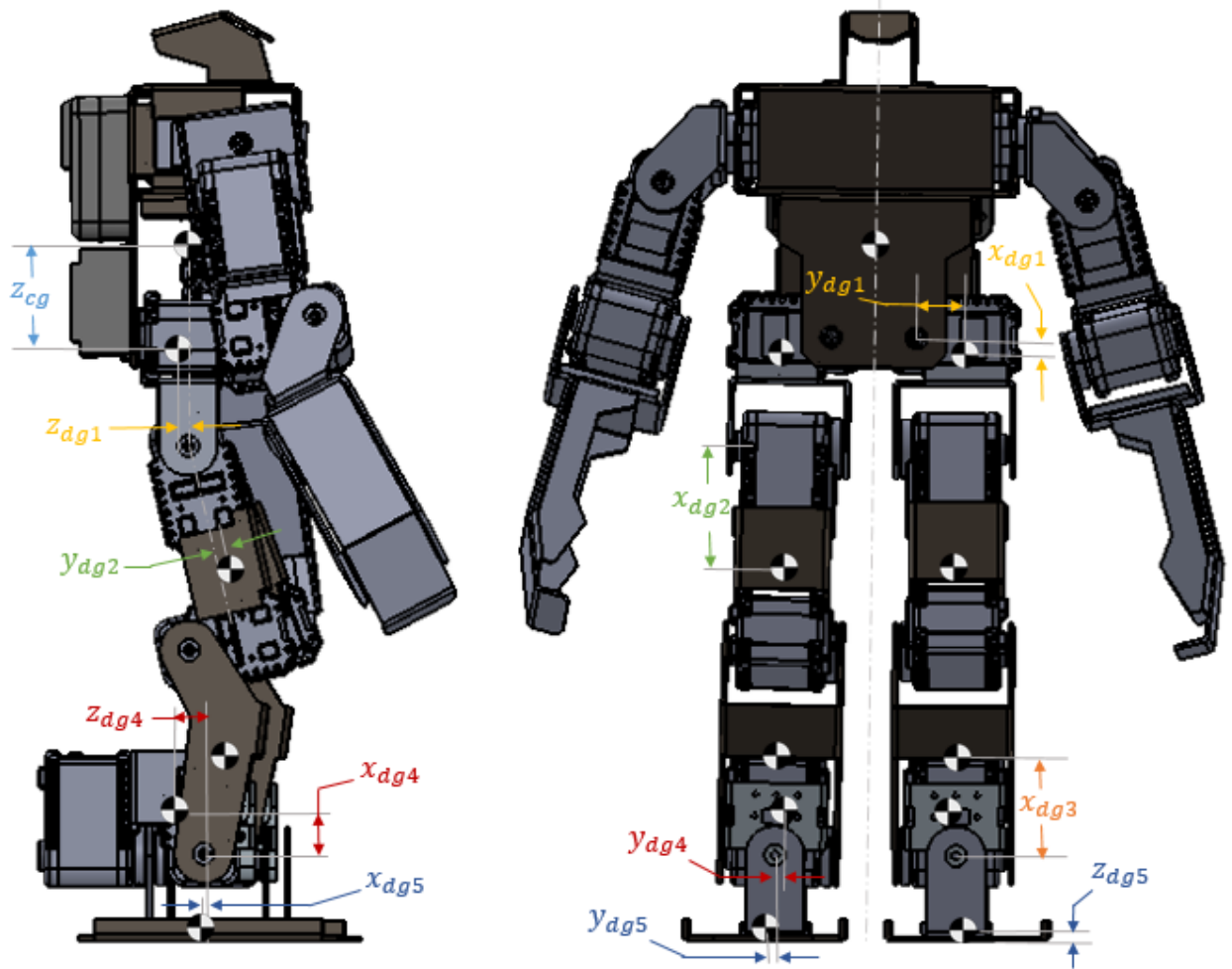
$$\begin{aligned}
 x_{cg} &= 0 \text{ mm}, & y_{cg} &= 0 \text{ mm}, & z_{cg} &= 44 \text{ mm} \\
 x_{dg1} &= 1.04 \text{ mm}, & y_{dg1} &= 7.76 \text{ mm}, & z_{dg1} &= 3.63 \text{ mm} \\
 x_{dg2} &= 46.47 \text{ mm}, & y_{dg2} &= 5.26 \text{ mm}, & z_{dg2} &= 0 \text{ mm} \\
 x_{dg3} &= 28 \text{ mm}, & y_{dg3} &= 0 \text{ mm}, & z_{dg3} &= 0 \text{ mm} \\
 x_{dg4} &= 15.1 \text{ mm}, & y_{dg4} &= 2.93 \text{ mm}, & z_{dg4} &= 15.72 \text{ mm} \\
 x_{dg5} &= 0.91 \text{ mm}, & y_{dg5} &= 3.02 \text{ mm}, & z_{dg5} &= 4.41 \text{ mm}
 \end{aligned}$$

Y nuevamente al considerar al robot completamente simétrico,

$$\begin{aligned}
 x_{dgk} &= x_{igk}, \quad \forall k = 1, \dots, 5 \\
 y_{dgk} &= y_{igk}, \quad \forall k = 1, \dots, 5 \\
 z_{dgk} &= z_{igk}, \quad \forall k = 1, \dots, 5
 \end{aligned} \tag{2.54}$$

Se puede determinar la ubicación de los centros de masa de la pierna izquierda con los parámetros de su complemento. Y como últimos datos descriptivos del modelo de pruebas, al momento de determinar el centro de masa de cada eslabón se midió la masa de cada uno de ellos obteniendo lo siguiente:

$$\begin{aligned}
 M_{cad} &= 617.5 \text{ g} \\
 M_{d1} &= M_{i1}, & M_{i1} &= 79.8 \text{ g} \\
 M_{d2} &= M_{i2}, & M_{i2} &= 135.1 \text{ g} \\
 M_{d3} &= M_{i3}, & M_{i3} &= 38 \text{ g} \\
 M_{d4} &= M_{i4}, & M_{i4} &= 157.2 \text{ g} \\
 M_{d5} &= M_{i5}, & M_{i5} &= 38 \text{ g}
 \end{aligned}$$

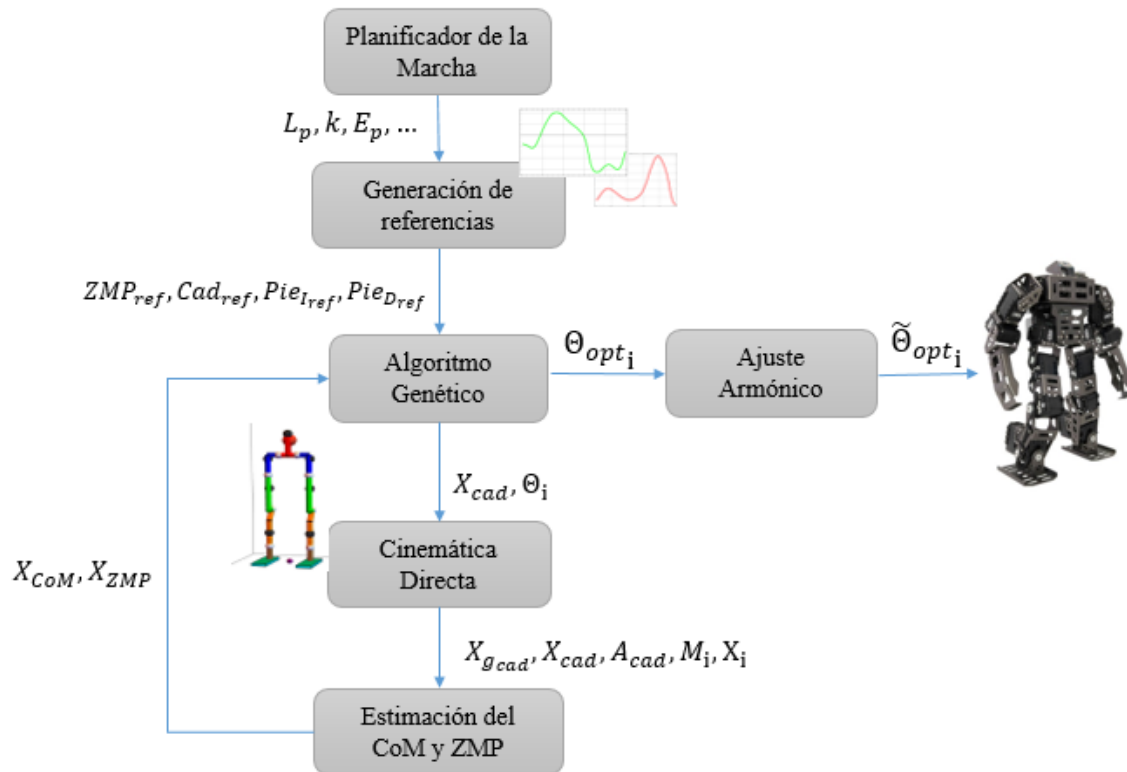


*Figura 2.6 Distancias a los centros de masa del robot Bioloid GP®, según se explicó en el modelo dinámico simplificado, en el subcapítulo 2.2.*

## Capítulo 3

### Optimización de la marcha

En éste capítulo se explicará en detalle las trayectorias base (articulares, de cadera y referencia del *ZMP*) de las cuales parte el algoritmo de optimización para su posterior modificación con el fin de lograr la estabilidad según el criterio establecido. Además, se definirán las tanto las entradas y salidas del AG como los parámetros utilizados en el proceso tales como el tipo de representación, límites en la generación de la población inicial, tipo de operadores genéticos y la definición de la función objetivo.



*Figura 3.1* Diagrama de bloques general de las funciones principales para la optimización de la marcha.

Como esquema introductorio al diseño del algoritmo, en la Figura 3.1 se muestra un diagrama de bloques que ilustra grosso modo la interacción entre cada uno de los elementos de la optimización de la marcha además de mostrar las señales que intercambia cada proceso. Empezando con planificador de la marcha, en éste caso el usuario o controlador manual, indica los parámetros con los cuales se planeará la marcha como pueden ser longitud de paso ( $L_p$ ), elevación ( $E_p$ ), número de pasos ( $k$ ), etc. Después se generan las trayectorias de referencia que se optimizan dentro del algoritmo genético por medio del modelo cinemático teórico antes descrito y se finaliza por suavizando las curvas optimizadas con el ajuste armónico antes de enviarlas al *Bioloid*®.

### 3.1 Planificación de trayectorias de efectores finales y variables articulares

Usualmente en la planificación de la marcha bípeda se ocupan 3 trayectorias espaciales, cadera y pies (2), las cuales se utilizan como variables de entrada en la solución de la cinemática inversa para así solucionar los valores de las variables articulares tal y como se realiza en [1, 22] y otros más. Por el contrario, nuestro enfoque yace en partir de trayectorias angulares medidas a partir de una persona sana y a partir de ellas modificarlas independientemente optimizándolas para converger en una trayectoria deseada del *ZMP*. Éste método se optó para disminuir el tiempo cómputo, ya que no necesita el cálculo de la cinemática inversa dado que representa un gran número de operaciones y considerando una futura planificación de marcha cuasi-*online* similar a [31] es importante tener una estimación del tiempo de cómputo.

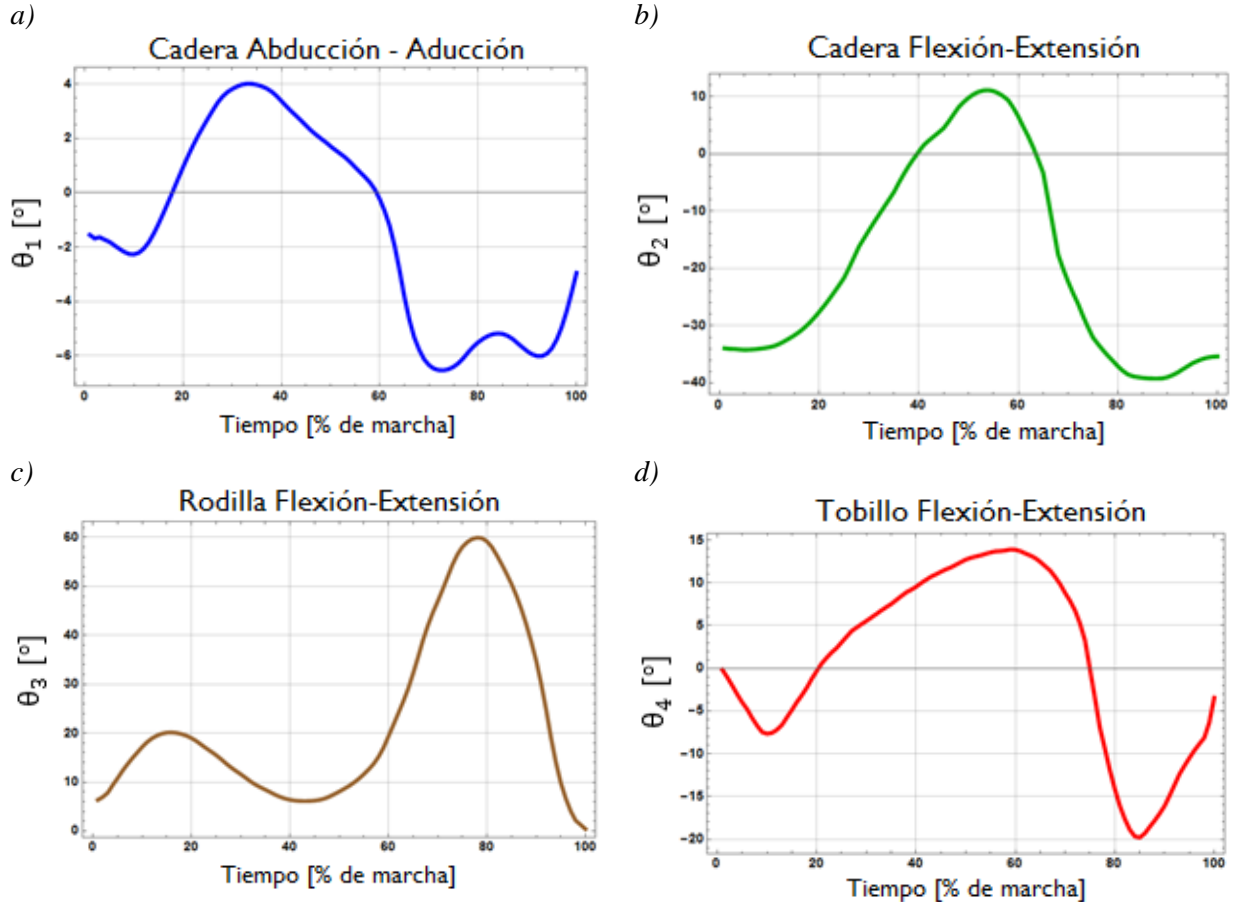
En éste método se utilizó la trayectoria espacial de la cadera para poder completar el movimiento de caminata en la simulación y las trayectorias de los pies como parámetro de optimización en el algoritmo. Dicha trayectoria de cadera del robot se utiliza como una aproximación del centro de masa para poder hacer la estimación del *ZMP* con sus valores de posición y aceleración. Sabiendo que la trayectoria de la cadera necesitaría componente de aceleración acotada se implementó por medio de un polinomio de quinto orden dando así de igual manera una componente de jerk finita, los parámetros del polinomio fueron elegidos según la nomenclatura listada en [22]. Los valores de velocidades y aceleraciones tanto finales como iniciales se fijaron en cero para obtener las siguientes ecuaciones de posición resolviendo el polinomio.

$$x_{cad}(t) = \frac{3 L_p t^5}{T^5} - \frac{15 L_p t^4}{T^4} + \frac{5 L_p t^3}{T^3} + \left( D_{cp} + k \frac{L_p}{2} \right) \quad (3.1)$$

$$y_{cad}(t) = \begin{cases} \frac{6 F_c t^5}{T^5} - \frac{15 F_c t^4}{T^4} + \frac{10 F_c t^3}{T^3}, & si \quad 0 \leq t \leq \frac{1}{4}T \\ F_c - \frac{6 F_c t^5}{T^5} + \frac{15 F_c t^4}{T^4} - \frac{10 F_c t^3}{T^3}, & si \quad \frac{1}{4}T < t \leq \frac{1}{2}T \\ -\left( \frac{6 F_c t^5}{T^5} - \frac{15 F_c t^4}{T^4} + \frac{10 F_c t^3}{T^3} \right), & si \quad \frac{1}{2}T < t \leq \frac{3}{4}T \\ -\left( F_c - \frac{6 F_c t^5}{T^5} + \frac{15 F_c t^4}{T^4} - \frac{10 F_c t^3}{T^3} \right), & si \quad \frac{3}{4}T < t \leq T \end{cases} \quad (3.2)$$

$$z_{cad}(k \cdot t) = \begin{cases} H_c + \frac{6 E_c t^5}{T^5} - \frac{15 E_c t^4}{T^4} + \frac{10 E_c t^3}{T^3}, & si \quad 0 \leq t \leq \frac{1}{2}T \\ E_c + H_c - \frac{6 E_c t^5}{T^5} + \frac{15 E_c t^4}{T^4} - \frac{10 E_c t^3}{T^3}, & si \quad \frac{1}{2}T < t \leq T \end{cases} \quad (3.3)$$

Donde  $L_p$  representa la longitud de paso,  $H_c$  es la altura de la cadera al piso,  $E_c$  la elevación de la cadera al momento de caminar,  $F_c$  el desplazamiento frontal sobre el eje  $x$ , y  $k$  es el parámetro que determina el número de pasos en la caminata. El siguiente paso fue derivar con respecto al tiempo las tres componentes para determinar las componentes de velocidad y aceleración de la cadera, con el objeto de utilizar dichos parámetros en la estimación del ZMP.



**Figura 3.2** Trayectorias articulares de la pierna izquierda a lo largo del ciclo de marcha, a) movimiento de abducción-aducción de cadera original, b) movimiento de flexión-extensión de cadera, c) flexión-extensión de rodilla y d) flexión-extensión del tobillo (\*Nota: 100% de la marcha equivale a un paso).

Al igual que la cadera, las trayectorias de los pies fueron determinadas por polinomios, en éste caso de tercer orden por el hecho de que únicamente se utilizarán como medida de referencia para garantizar hasta cierto punto que el robot está avanzando, obteniendo las siguientes ecuaciones de posición.

$$x_{pie}(t) = \begin{cases} \frac{3 L_p t^2}{T_{SS}^2} - \frac{2 L_p t^3}{T_{SS}^3} + (k L_p), & \text{si } 0 \leq t \leq T_{SS} \\ (k + 1)L_p, & \text{si } T_{SS} < t \leq T \end{cases} \quad (3.4)$$

$$y_{pie}(t) = \begin{cases} P_p + \frac{12 (F_p - P_p) t^2}{T_{ss}^2} + \frac{16(P_p - F_p) t^3}{T_{ss}^3}, & \text{si } 0 \leq t \leq \frac{1}{2}T_{ss} \\ F_p + \frac{12 (P_p - F_p) t^2}{T_{ss}^2} + \frac{16 (F_p - P_p) t^3}{T_{ss}^3}, & \text{si } \frac{1}{2}T_{ss} < t \leq T_{ss} \\ P_p, & \text{si } T_{ss} < t \leq T \end{cases} \quad (3.5)$$

$$z_{pie}(t) = \begin{cases} \frac{12 E_p t^2}{T_{ss}^2} - \frac{16E_p t^3}{T_{ss}^3}, & \text{si } 0 \leq t \leq \frac{1}{2}T_{ss} \\ E_p - \frac{12 E_p t^2}{T_{ss}^2} + \frac{16 E_p t^3}{T_{ss}^3}, & \text{si } \frac{1}{2}T_{ss} < t \leq T_{ss} \\ 0, & \text{si } T_{ss} < t \leq T \end{cases} \quad (3.6)$$

Donde  $P_p$  representa la distancia lateral sobre el plano frontal del centro del robot al pie,  $F_p$  la distancia hasta donde se desplaza el pie lateralmente a lo largo de la marcha,  $E_p$  la elevación del paso y finalmente  $L_p$  que como se mencionó anteriormente es la longitud del paso. Las trayectorias de los pies izquierdo y derecho son prácticamente iguales, pero cuentan con un desfase en el tiempo y son simétricas con respecto al plano sagital.

En el caso de las variables articulares, como se mencionó previamente, fueron tomadas experimentalmente de personas en [33] como trabajo previo a éste proyecto de investigación. Obteniendo cien muestras por ciclo de marcha de cada una de las articulaciones de cadera, rodilla y tobillo, ajustadas en magnitud a las dimensiones del robot tal y como se muestran en la Figura 3.2. Las trayectorias angulares de la pierna derecha e izquierda son las mismas, sin embargo, el signo del ángulo cambia según la configuración determinada previamente en la cinemática directa, además de que encontrarse desfasada una de la otra 40% del ciclo de marcha para lograr la conmutación entre soporte simple y doble.

### 3.2 Diseño del algoritmo genético

En la sección anterior se mostraron las trayectorias que forman parte de los cromosomas de los individuos de la población del AG, sin embargo, para el diseño o desarrollo del algoritmo de optimización hace falta determinar distintos parámetros y funciones, además de designar completamente su interacción entre ellos fijando también entradas y salidas del sistema.

#### 3.2.1 Esquema general del AG

Tal y como se puede ver en la Figura 3.3 al sistema de optimización entran las trayectorias ya mencionadas de cadera y articulaciones, además de la referencia del *ZMP* a seguir. Después se evalúa si las trayectorias generadas de referencia dadas a partir del polinomio de quinto orden para la cadera y las mediciones angulares para las juntas se encuentran dentro de la zona de estabilidad, en caso de estarlo, se pasa al siguiente punto de avance hasta terminar el ciclo de marcha según el número de pasos especificado.

Y, por el contrario, en caso de no ser estable se prosigue a ejecutar el algoritmo genético generando primero la población inicial para el tiempo  $t$ , y después revisar si alguno de sus cromosomas de la población sobrepasa los límites bio-mecánicos permitidos por un ser humano para que en todo caso restringir dichos ángulos. Una vez generada la población inicial se evalúa cada individuo para tener contabilizada su aptitud y luego ejecutar los operadores genéticos, donde el proceso se repite hasta llegar a la generación máxima y pasar al siguiente instante de tiempo donde se corre nuevamente todo hasta culminar la marcha para que

finalmente se haga un ajuste con una función polinómica de cosenos y senos con el objetivo de suavizar los cambios en la trayectoria antes de ser implementados en el robot.

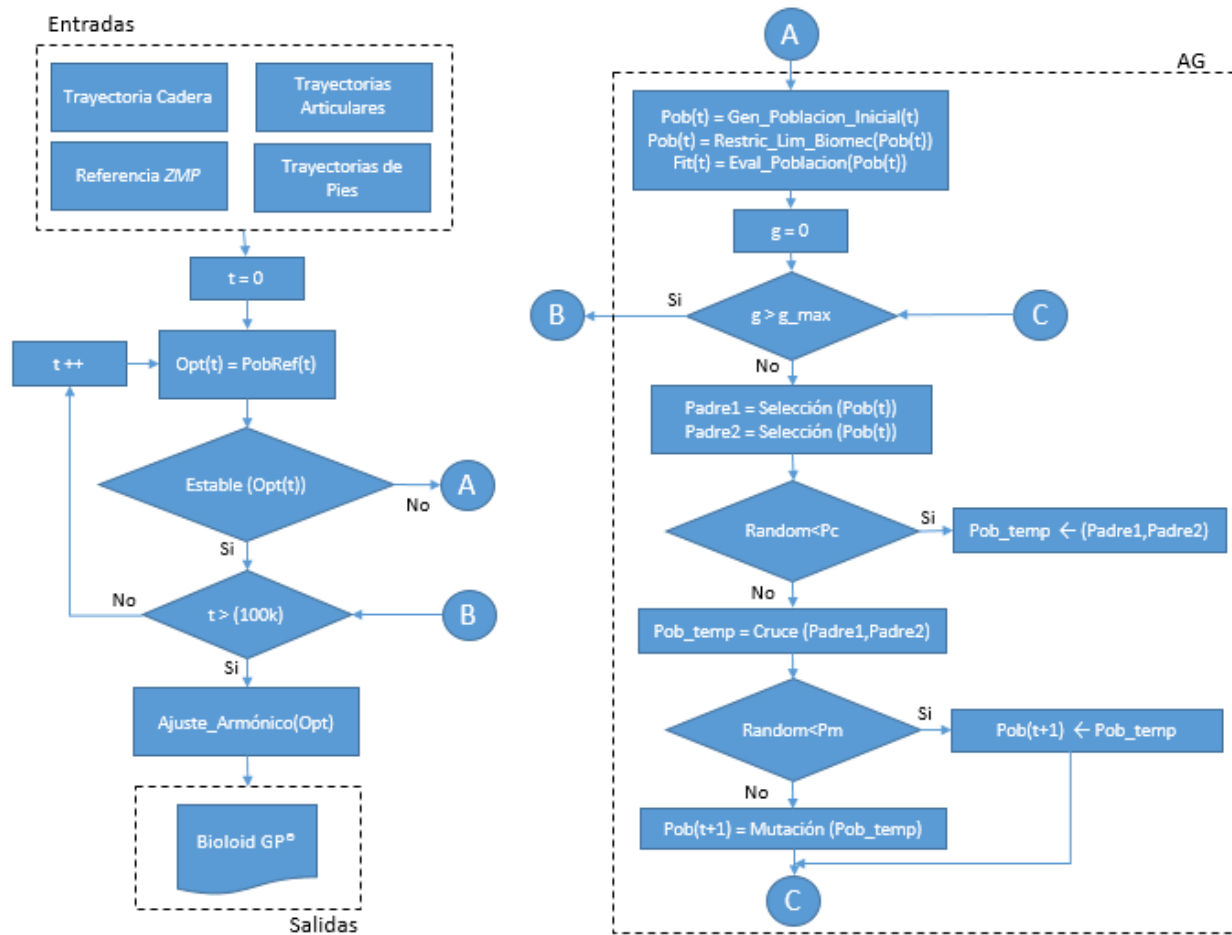


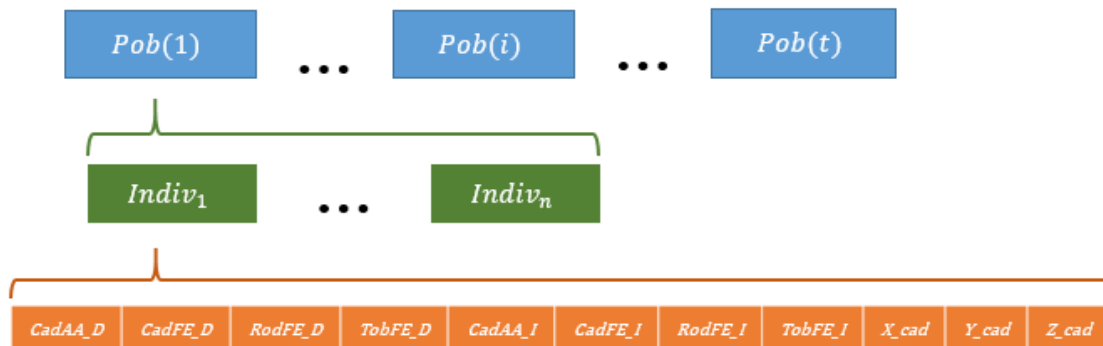
Figura 3.3 Diagrama de flujo y procesos del algoritmo genético de optimización de marcha.

### 3.2.2 Generación de la población inicial con límites articulares

Para la generación de los individuos en nuestro problema de optimización, los cromosomas de dichos individuos están constituidos por las ocho variables articulares activas en el mecanismo más las tres coordenadas de posición de la cadera, dando un total de once variables. Se seleccionó un tipo de representación real en la cual las variables articulares están comprendidas en radianes y las distancias hacia los puntos coordenados de la cadera están dadas en metros.

A diferencia de [34] que define a un cromosoma como una solución para toda la caminata y los genes definen a los valores de cada servomotor en un determinado instante de tiempo, en éste enfoque se optimiza instante por instante de la caminata, lo que permite delimitar las trayectorias en el tiempo para dar una similitud de marcha lo más parecida a la humana. La estructura de los datos del algoritmo empieza por las poblaciones, generadas primero pseudo-aleatoriamente dado que parten de una semilla definida por su número de individuo multiplicado por '1234', para garantizar consistencia en la experimentación y después generadas por la semilla aleatoria definida en *Mathematica*®, que existen en cada instante de tiempo y que se modifican con el paso de las generaciones en ese lapso, donde cada población cuenta con un número  $n$  de elementos o individuos dentro de los cuales su cromosoma consta de sus primeros ocho genes

indicando el valor de cada articulación y los últimos tres dando la posición de la cadera en sobre los tres ejes, ordenadas primero las articulaciones del miembro derecho y después su contraparte simétrica, así como se muestra en la Figura 3.4.



**Figura 3.4** De arriba para abajo, diagrama de estructuras de las poblaciones en cada instante tiempo, después individuos de la muestra poblacional y finalmente la organización de los elementos del individuo (genes del cromosoma).

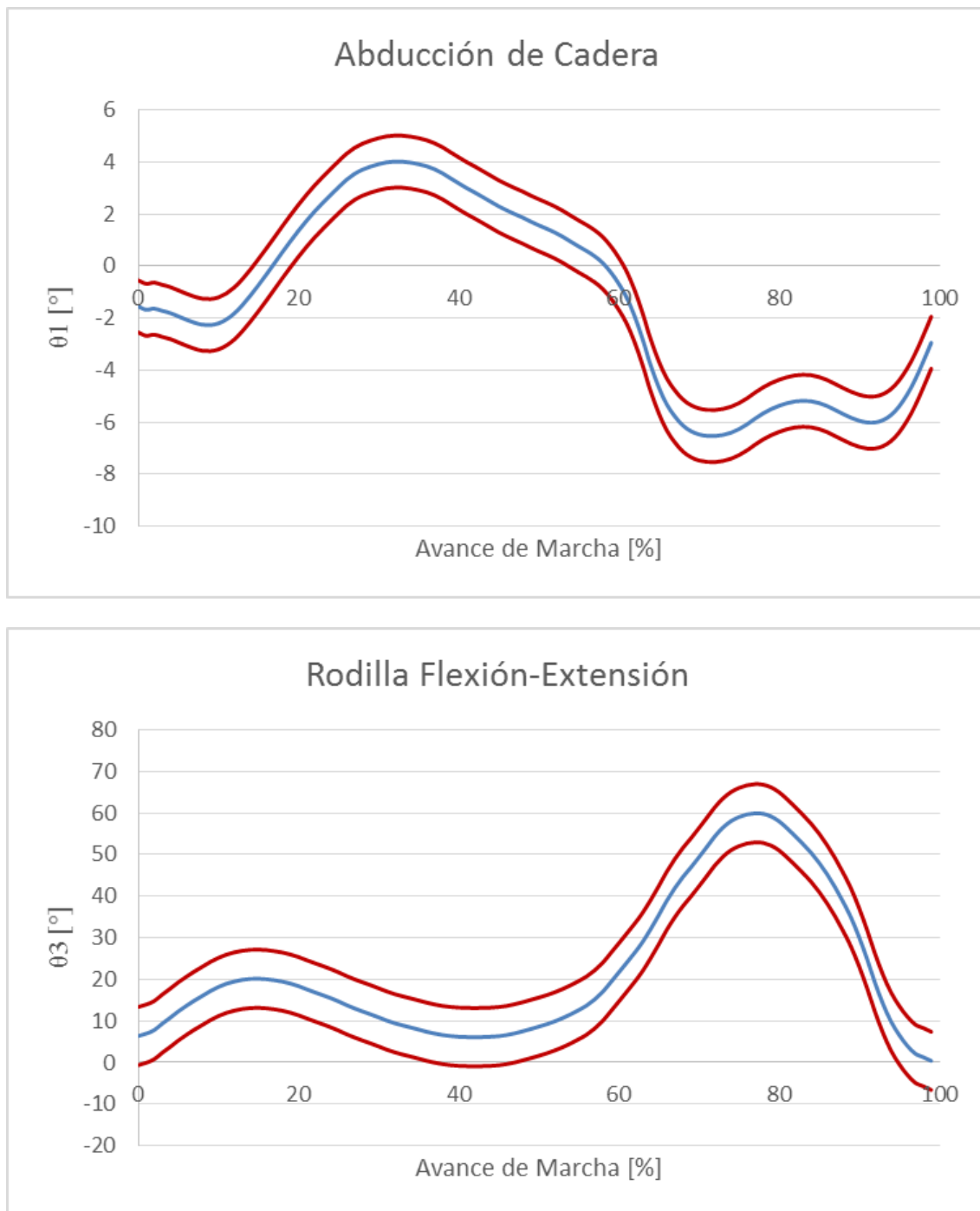
Como concepto de búsqueda en el espacio solución se define la “banda de exploración” la cual es específica para cada articulación, y está delimitada por una profundidad de búsqueda, es decir, se asigna para cada articulación que tanto se puede alejar del valor de referencia para permitir la búsqueda del elemento más apto en un espacio mayor o menor según sea el caso. En la Figura 3.5 se aprecian dos de las trayectorias articulares junto con sus respectivas bandas de exploración, donde la profundidad es modificada en las pruebas hasta lograr el desempeño óptimo.

El siguiente paso en la generación de la población inicial es revisar que cada una de las variables articulares se encuentren dentro del rango de movimiento deseado, es decir, que no superen los límites biomecánicos del ser humano además de evitar que por cuestiones geométricas las extremidades del robot colisionen entre sí en la fase de balanceo. Para esto simplemente se satura el valor de la articulación en el límite máximo o mínimo según sea el caso para restringir de manera correcta el movimiento del robot. Las funciones definidas en la implementación del algoritmo, de generación de referencia y población inicial, así como restricciones mecánicas se pueden observar en el Apéndice B.1.

### 3.2.3 Selección, cruce y mutación

En los operadores genéticos, el tipo de selección utilizado fue el de torneo binario, proceso en el cual se toman aleatoriamente a cuatro individuos, se ponen a competir entre ellos en pares y dependiendo de cuales tengan la mejor aptitud se almacenan en dos variables temporales, para proseguir con la operación de cruce. En el cruce o reproducción, se genera un número aleatorio entre 0 y 1 para compararlo con la probabilidad de cruce predefinida y en caso de excederla, las variables temporales o padres pasan a la siguiente generación directamente. En el caso contrario, se ejecuta el operador de cruce simple para representación real, el cual toma al primer gen de cada padre para aplicar una combinación lineal entre ellos y obtener dos descendientes dado el parámetro de cruce y su complemento.





**Figura 3.5** Gráficas de ejemplificación del espacio de búsqueda en las articulaciones de cadera y rodilla respectivamente.

La mutación se lleva a cabo bajo el mismo criterio que el del cruce, se genera un número aleatorio para compararlo con la probabilidad ahora de mutación y se ejecuta únicamente cuando el número aleatorio es menor a dicha probabilidad. Dado que la representación es real, la mutación no puede darse únicamente invirtiendo el valor de un bit, sino que una vez que se seleccionó la casilla o gen a modificar aleatoriamente, se cambia el valor que se encuentra dentro de la casilla por un valor completamente diferente generado también aleatoriamente, dentro del dominio de los límites definidos previamente en dicho gen.

Generalmente, la probabilidad de mutación es muy baja con respecto a la de cruce dado que el hecho de ejecutar en exceso la mutación puede provocar que la población diverja de la zona óptima. Los parámetros fijos del algoritmo se modificaron en cierta medida conforme las pruebas se efectuaron para comparar y llegar a una mejor solución, las funciones utilizadas pueden encontrarse en el Apéndice B.2.

### 3.2.4 Función objetivo

En la optimización de la marcha bípeda, se trata siempre de estar en la zona de estabilidad, en otras palabras, mantener el *ZMP* dentro del polígono de soporte determinado en la fase de soporte simple por la planta del pie de apoyo y en la fase de soporte doble por la región generada por la planta de ambos pies y la superficie intermedia. Además, es importante considerar que el tiempo que se lleva el cómputo de dicha operación es muy elevado y es necesario buscar la manera de agilizarlo por tantos medios se posible, por esa razón, nos enfocamos a delimitar estrictamente nuestra región de estabilidad para únicamente correr o ejecutar la optimización en los puntos donde la marcha sale del equilibrio dinámico.

El primer paso para determinar si el sistema se encuentra en equilibrio es el de definir las fronteras de estabilidad. En este caso, nosotros partimos de asignar un umbral de seguridad, es decir, una distancia hacia cualquier dirección a partir de la referencia deseada trazada del *ZMP* y esto genera una región de estabilidad que rodea la trayectoria del *ZMP* en el espacio. Dicha trayectoria se realiza tomando en cuenta los tiempos determinados en el ciclo marcha, así como las diferentes fases de soporte y similar a lo que explica [35], se toman en cuenta parámetros nativos del modelo de pruebas tal como la longitud de paso y distancia del centro al pie. Las ecuaciones para generar la trayectoria son las siguientes:

$$ZMP_X = \begin{cases} k \frac{L_p}{2}, & 0 \leq t < T_{ss} \\ \left(k \frac{L_p}{2}\right) + \left(\frac{0.5 L_p}{T_{sd}}\right)(t - T_{ss}), & T_{ss} \leq t < T \end{cases} \quad (3.7)$$

$$ZMP_Y = \begin{cases} \pm L_{cp}, & 0 \leq t < T_{ss} \\ \pm \left(\left(\frac{2 L_{cp}}{T_{sd}}\right)(t - T_{ss}) - L_{cp}\right), & T_{ss} \leq t < T \end{cases} \quad (3.8)$$

Donde  $L_p$  representa la longitud de paso,  $L_{cp}$  la distancia del centro del robot al pie,  $T_{ss}$  el tiempo de soporte simple,  $T_{sd}$  el tiempo de soporte doble y  $k$  representa el  $k$ -ésimo paso de la marcha. Ahora que se determinó la referencia del *ZMP* y conocemos el umbral de seguridad, sólo se necesita determinar para cada instante, si el robot se encuentra dentro del área de seguridad o no. Éste parámetro específico de una configuración dada del robot se denominó *ZMP* instantáneo, el cual se calcula con datos de la posición y aceleración del centro de masa del robot con la ecuación del modelo de péndulo invertido descrita en el capítulo anterior. Y se necesita calcular para cada instante la posición de cada uno de los eslabones por medio de la cinemática directa, y poder así estimar la posición del centro de masa con el promedio ponderado de las masas.

Ya que se obtuvieron los tres datos necesarios para el computo de la existencia de estabilidad, se prosiguió a implementar una función que revisa desde las fronteras positivas y negativas de los ejes  $x$  y  $y$  que el valor instantáneo del *ZMP* se encuentre dentro del área de seguridad, regresando el valor de 1 en caso de estar estable y 0 en caso de no estarlo. Sin embargo, la función objetivo abarca más que sólo una respuesta binaria en caso de ser estable o no, y es de hecho, un valor real que se le asigna a cada individuo de la población de búsqueda que califica que tan cerca se encuentran del valor óptimo.

### CAPÍTULO 3 OPTIMIZACIÓN DE LA MARCHA

Y es así que la función objetivo consta de un miembro que se encarga de cuantificar cuan bueno es un individuo desde el punto de vista de estabilidad y el otro miembro de la función se encarga de penalizar a aquellos individuos que ubican a los pies lejos de la referencia de marcha, dicha penalización está escalada con respecto a la aptitud del primer miembro de la ecuación con el parámetro  $W$  para evitar una búsqueda de solución fuera del área de estabilidad. Como tal, la función objetivo o de aptitud se define:

$$F(t, c_i) = dist_{refZMP \rightarrow ZMP_{inst}} + W \left( \frac{dist_{pieDer} + dist_{pieIzq}}{2} \right) \quad (3.9)$$

Donde  $dist_{pieDer}$  y  $dist_{pieIzq}$  están definidos como la distancia entre los vectores de posición de la transformación  $D_C^9$  y  $I_C^9$  respectivamente que mapean el sistema de referencia de cada pie al sistema de referencia inercial y los valores de posición de referencia definidos previamente. Dado que éste es un problema de minimización, la penalización aumenta el valor de la función otorgando una mejor aptitud a aquellos individuos que sean más estables y que además cuenten con los pies lo más cercanos a las referencias de dicho lapso de tiempo. De igual manera desarrollando el primer miembro de la ecuación,

$$dist_{refZMP \rightarrow ZMP_{inst}} = \sqrt{(ZMP_X(t) - ZMP_{Xinst}(t, c_i))^2 + (ZMP_Y(t) - ZMP_{Yinst}(t, c_i))^2} \quad (3.10)$$

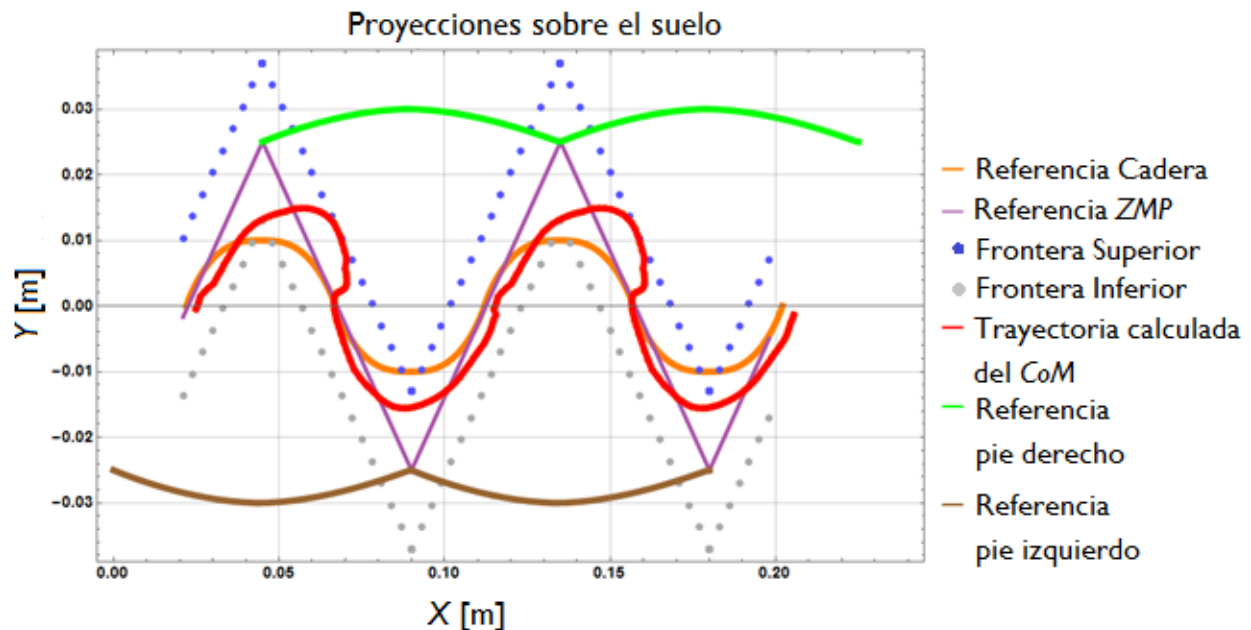
Se determina la norma o distancia euclidiana entre el punto de la referencia del  $ZMP$  y el  $ZMP$  instantáneo calculado en ese instante de tiempo. Para el cálculo de las distancias de los efectores finales se utiliza la misma norma, pero en éste caso se calcula en  $\mathbb{R}^3$  al ser vectores en el espacio. El desarrollo de las ecuaciones para el cómputo de la función objetivo completa se encuentra en el Apéndice B.3.

# Capítulo 4

## Implementación y Pruebas

### 4.1 Simulación cinemática

Para la simulación cinemática del funcionamiento del algoritmo se utilizó el paquete de cómputo de *Mathematica*® que contiene funciones de generación de gráficos primitivos a partir de vectores de posición espaciales, los cuales determinamos mediante las matrices de transformación homogéneas de cada uno de los sistemas de referencia. El procedimiento empezó por desplegar el esqueleto del robot con todas las medidas obtenidas previamente, para después ejecutar una simulación de la marcha dados el número de pasos y sus respectivas trayectorias de pies, cadera y articulaciones. Ésta primera simulación se utilizó como una muestra no optimizada y poder así comparar y evaluar el desempeño del algoritmo genético en el paso de generaciones y ver así de qué manera se aproximaba el *ZMP* al deseado.

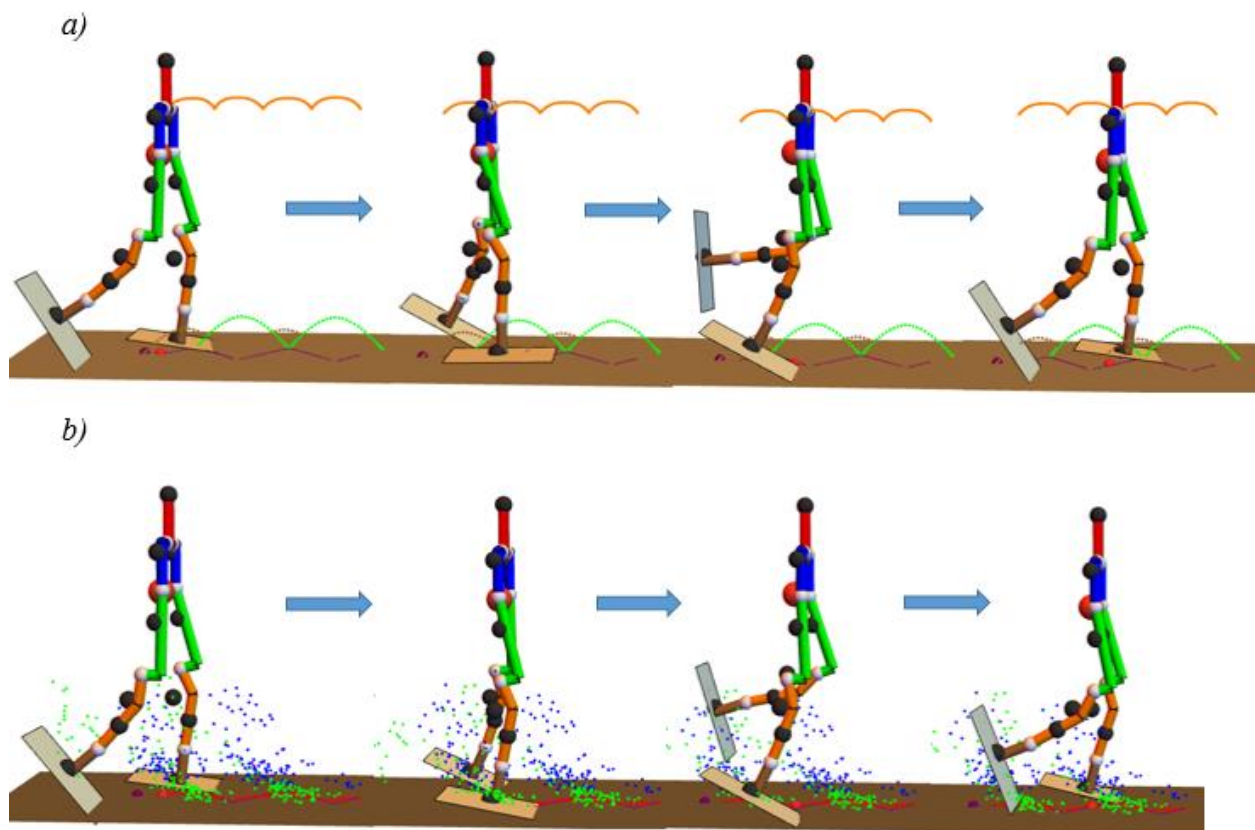


**Figura 4.1** Proyecciones sobre el piso del *CoM* calculado, de la trayectoria de referencia de la cadera, del pie derecho, del pie izquierdo y del *ZMP*, con sus respectivas fronteras de estabilidad.

En la Figura 4.1 se ilustran las trayectorias que se ocuparon tanto como entrada, cadera, pies y *ZMP* de referencia, como la salida calculada para cada instante de la marcha que es el *CoM*, dentro de la simulación fue muy importante remarcar que las trayectorias de referencia tenían que empatar y ser

coherentes no sólo en el espacio, sino que también tenían que ser coincidentes en el tiempo dado que la optimización del algoritmo se hace por intervalos discretos de tiempo definidos y el tener un desfase en dichos intervalos podía ocasionar malas interpretaciones de los resultados del algoritmo.

Ya en la parte de ejecución del algoritmo genético se optó por realizar distintas pruebas para evaluar que parámetros del algoritmo proporcionaban una mejor optimización, el proceso de experimentación se realizó de la siguiente forma, primero, se definió un peso de penalización igual a uno para poder tener una relación ecuánime entre la búsqueda de estabilidad y el éxito de caminata. Una vez fijo el factor de penalización, se corrieron distintas pruebas variando la profundidad de búsqueda y número de individuos en la población para identificar qué valor de ambos parámetros permitía la mayor exploración sin dejar de lado la optimización. En éste primer experimento se realizaron tres pruebas con una semilla constante definida por el número de individuo para poder hacer repetible el proceso pseudo-aleatorio y una prueba con la semilla definida por el paquete de software.



**Figura 4.2** a) Simulación de marcha sin optimizar con trayectorias de referencia de cadera y efectores finales, b) Simulación optimizada por el AG con puntos representando la posición de los efectores finales generados por el algoritmo.

En la segunda parte de la experimentación se realizaron pruebas variando distintos parámetros, tanto el número de individuos de la población, como el peso de penalización y se registraron los resultados obtenidos después de un cierto número de generaciones. Para poderlos comparar se normalizaron los valores de aptitud, de manera que tuviéramos valores adimensionales entre 0 y 1, y así poder revisar que configuración se acoplaba de mejor manera a nuestro problema. La tabla de promedios de aptitud a lo largo de la marcha de los dos experimentos con todas las pruebas realizadas, se muestra en la Tabla 4.1 para su realizar posterior análisis en la tercera sección de éste capítulo.

**Tabla 4.1** Resultados de las pruebas de simulación realizadas en *Mathematica*® del algoritmo de optimización de marcha.

	# Ind.	Peso pen.	Prof.	Muestra 1		Muestra 2		Muestra 3		Muestra 4	
				No. Gen.	Aptitud promedio	No. Gen.	Aptitud promedio	No. Gen.	Aptitud promedio	No. Gen.	Aptitud promedio
1	8	1.2	3.0	2	0.141677	5	0.141682	8	0.141682	15	0.141682
2	6	1.0	3.0	1	0.119030	4	0.119120	7	0.119120	13	0.119120
3	10	1.0	3.0	1	0.119718	5	0.120001	10	0.120001		
5	30	1.0	4.0	1	0.119718	2	0.118131	3	0.118131	4	0.118132
6	15	3.0	5.0	1	0.119718	2	0.316370	3	0.316370		
7	8	5.0	6.0	1	0.119718	2	0.510368	3	0.510368		
8*	6	50.0	6.0	1	0.119718	2	4.851495	4	4.918907		
9*	6	0.5	6.0	1	0.119030	2	0.067455	3	0.068435	4	0.068168
10*	12	1.0	6.0	1	0.119030	20	0.115480				

(\* Nota: Pruebas realizadas con semilla aleatoria).

#### 4.2 Experimentación en el *Bioloid GP*®

La parte de la implementación sobre el modelo de pruebas antes descrito, se llevó a cabo sobre el mismo software de *Mathematica*® con el paquete incluido de lenguaje simbólico que permite la comunicación serial con dispositivos externos. Para llevar a cabo dicha comunicación se reemplazó el controlador que contenía el *Bioloid GP*® y permitir así una comunicación directa. Los cambios tanto en hardware como en firmware y el protocolo de comunicación implementado están completamente descritos en el Apéndice A.

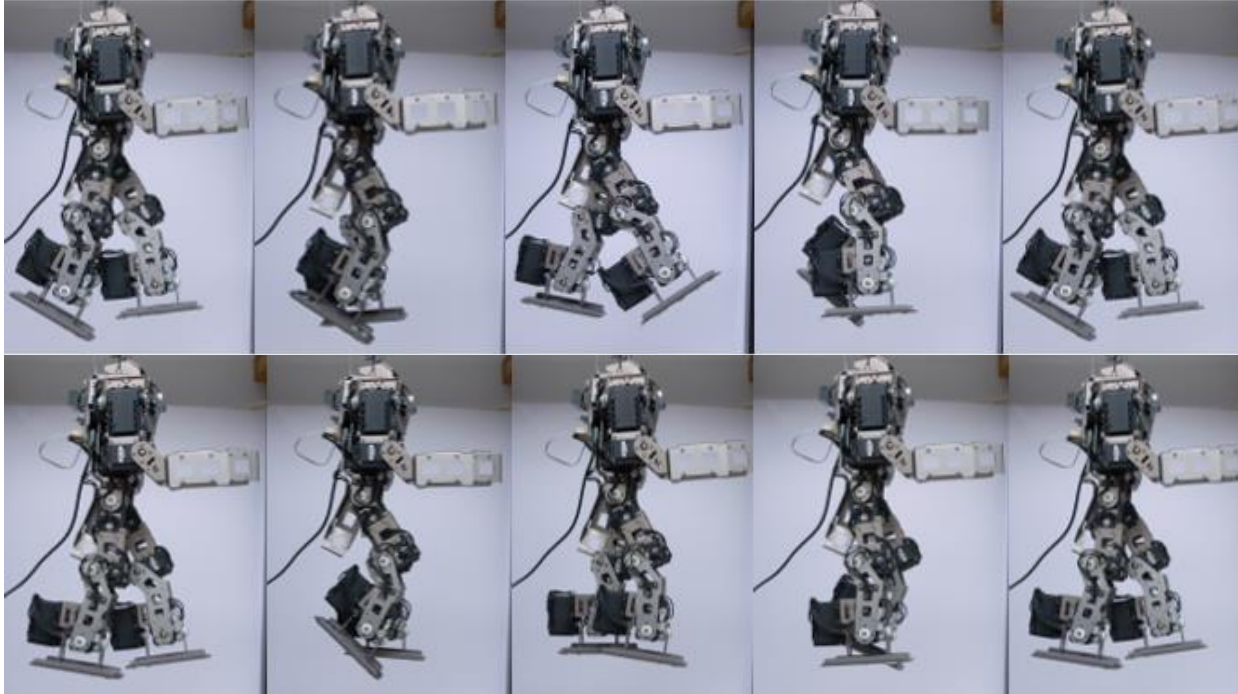
El principal objetivo de realizar pruebas sobre el modelo físico es de revisar que las trayectorias cumplieran con los parámetros requeridos desde un inicio de no sobrepasar los límites biomecánicos permisibles en la cinemática humana y que además dichas trayectorias angulares obtenidas fueran lo más similares posibles a las de una persona sana. Así en la Figura 4.4 se comparan las trayectorias directamente medidas de la persona con aquellas que son optimizadas y notar de esa manera cuanto se modifican las trayectorias articulares, donde el principal cambio en el desarrollo de la trayectoria se nota en la fase de soporte doble que muestra como la orientación de la planta del pie con respecto al piso es casi paralela en la secuencia optimizada comparada con la secuencia natural donde la planta del pie muestra un ángulo cercano a los 30° con respecto a la horizontal, además de verse como el paso se acorta de manera que se acerque a la referencia de paso designada. Otro objetivo secundario fue el de realizar pruebas para verificar el desempeño del robot con las trayectorias enviadas y analizar como difieren los resultados de caminata del modelo teórico con la marcha real del robot donde no se obtuvo una caminata satisfactoria en lazo abierto y más adelante se discutirán dichos resultados.

#### 4.3 Discusión de resultados

En el análisis de resultados se partió en un inicio por determinar qué parámetros eran los que más se ajustaban al problema de optimización de marcha, y es por eso que se ejecutaron distintas simulaciones para hacer las comparaciones pertinentes. Y dado que se tenían distintas variables a modificar, fue muy importante realizar pruebas que permitieran visualizar bajo qué tendencia se obtenían mejores resultados.

## CAPÍTULO 4 IMPLEMENTACIÓN Y PRUEBAS

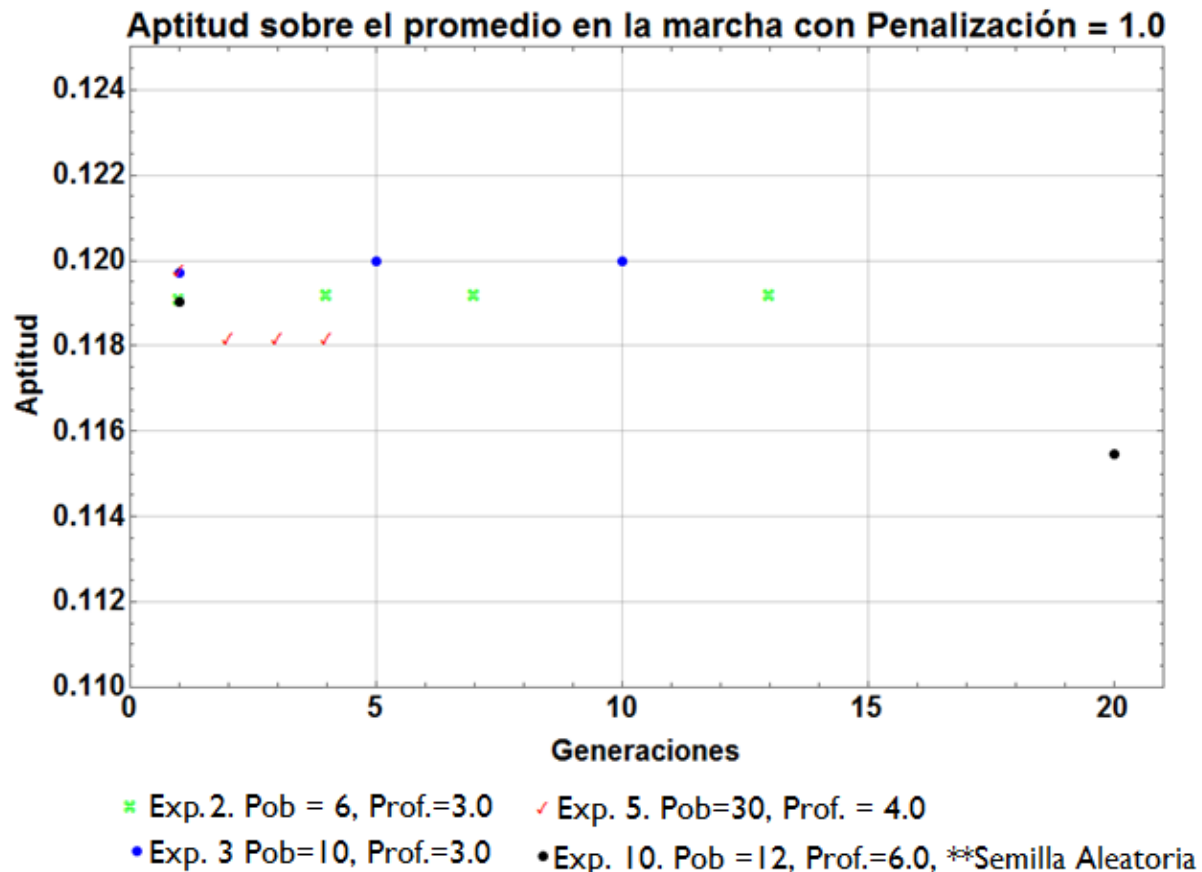
Partiendo de esa idea, se graficaron las cuatro pruebas o simulaciones con peso de penalización igual a 1, y tal y como se aprecia en la Figura 4.5, la aptitud va mejorando según se aumentaba el espacio de exploración con la profundidad de búsqueda, además que es notable en el experimento 10 como el número de generaciones también jugó un papel importante en la disminución del valor de la función objetivo. Y por el otro lado, el tamaño de la población no aportaba tanto como las generaciones en comparación del tiempo de cómputo que agrega el hecho de generar el doble de individuos por cada instante de la marcha.



**Figura 4.4** Revisión del progreso en la caminata, en la secuencia superior se muestran las trayectorias naturales y en la secuencia inferior las ya optimizadas. Los lapsos de tiempo tomados para la secuencia corresponden al 0, 25, 50, 75 y 99% del ciclo de marcha.

Una vez analizada la repercusión que tiene tanto el tamaño de la población como la profundidad de búsqueda, era necesario verificar qué proporción de penalización otorga mejor desempeño. Sin embargo, debido a que el hecho de aumentar o disminuir el peso de penalización ocasiona una variación en el valor de la aptitud, se necesitaba buscar una manera de evaluar el mejoramiento al paso de las generaciones, y por esa razón, dicha comparación se realizó de una manera adimensional o normalizada con respecto a los valores obtenidos con los mismos parámetros a lo largo del aumento de las generaciones. Así que simplemente se obtuvo la norma de las  $n$  muestras tomadas y se dividió cada valor medido entre la norma, proporcionando un valor entre 0 y 1, a la cual denominó aptitud relativa.

Gracias a la normalización de las aptitudes, es más sencillo visualizar el desempeño de los distintos experimentos, remarcando sobre todo que las pruebas con la semilla aleatoria y con mayor profundidad muestran nuevamente mejores resultados. Al estar buscando que la aptitud se minimice tanto como sea posible, se puede determinar que únicamente los últimos dos experimentos están cumpliendo con ese comportamiento y, por lo tanto, el problema de encontrar el valor óptimo del peso de penalización se reduce a revisar la proporción que disminuye la aptitud al paso de las generaciones, dejando como claro ganador al valor de 0.5, disminuyendo en un 43.33% el valor de aptitud relativa de la primera a la segunda generación, comparado con el 2.98% de disminución que muestra el valor de 1.0.



*Figura 4.5 Evolución de la aptitud con penalización unitaria y con variación de parámetros de profundidad de búsqueda y tamaño de población.*

Así es como el último parámetro para obtener el mejor desempeño posible del algoritmo es el número de generaciones, el cual se determina por la rapidez de convergencia del mismo. En este caso, se verificó que el algoritmo ha convergido cuando los valores de aptitud de cada uno de los individuos de la población en determinado lapso de tiempo tienen una variación mínima con respecto al promedio, es decir, que la desviación estándar es menor al 5% del valor del promedio de las aptitudes obtenidas. Bajo ese criterio, se determinó utilizar un total de 12 generaciones para realizar las pruebas sobre el modelo físico, simulación en la cual se muestra el nivel de convergencia en la Figura 4.7.

Después de haber analizado la cantidad de posibles configuraciones en las que puede trabajar el algoritmo desarrollado con sus distintos parámetros, y haber encontrado de manera analítica una de las tantas configuraciones óptimas, se demostró que el algoritmo desde el punto de vista teórico está cumpliendo correctamente con su función de optimización; sin embargo, al ser un problema con tal cantidad de grados de libertad, es difícil asegurar que el algoritmo encontró el mínimo global para dicha función escogida. Como sea, luego de haber realizado las pruebas sobre el modelo físico, se hizo evidente que, aunque el robot estaba siguiendo correctamente las trayectorias cuando se encontraba suspendido, al momento en que este se encontraba con alguna fuerza externa que se opusiera a su movimiento empezaba a mostrar errores en el seguimiento. Esto se debe, sin dudas, a que el esquema de control con el que se está probando el robot es de lazo abierto, y al no tener realimentación, ni de posición, ni de localización del *ZMP* no le es posible compensar el sistema para poder lograr la caminata exitosa. Por esa razón, uno de los principales factores a trabajar en el futuro es el desarrollo de las pruebas de seguimiento de trayectorias con un esquema de lazo cerrado.



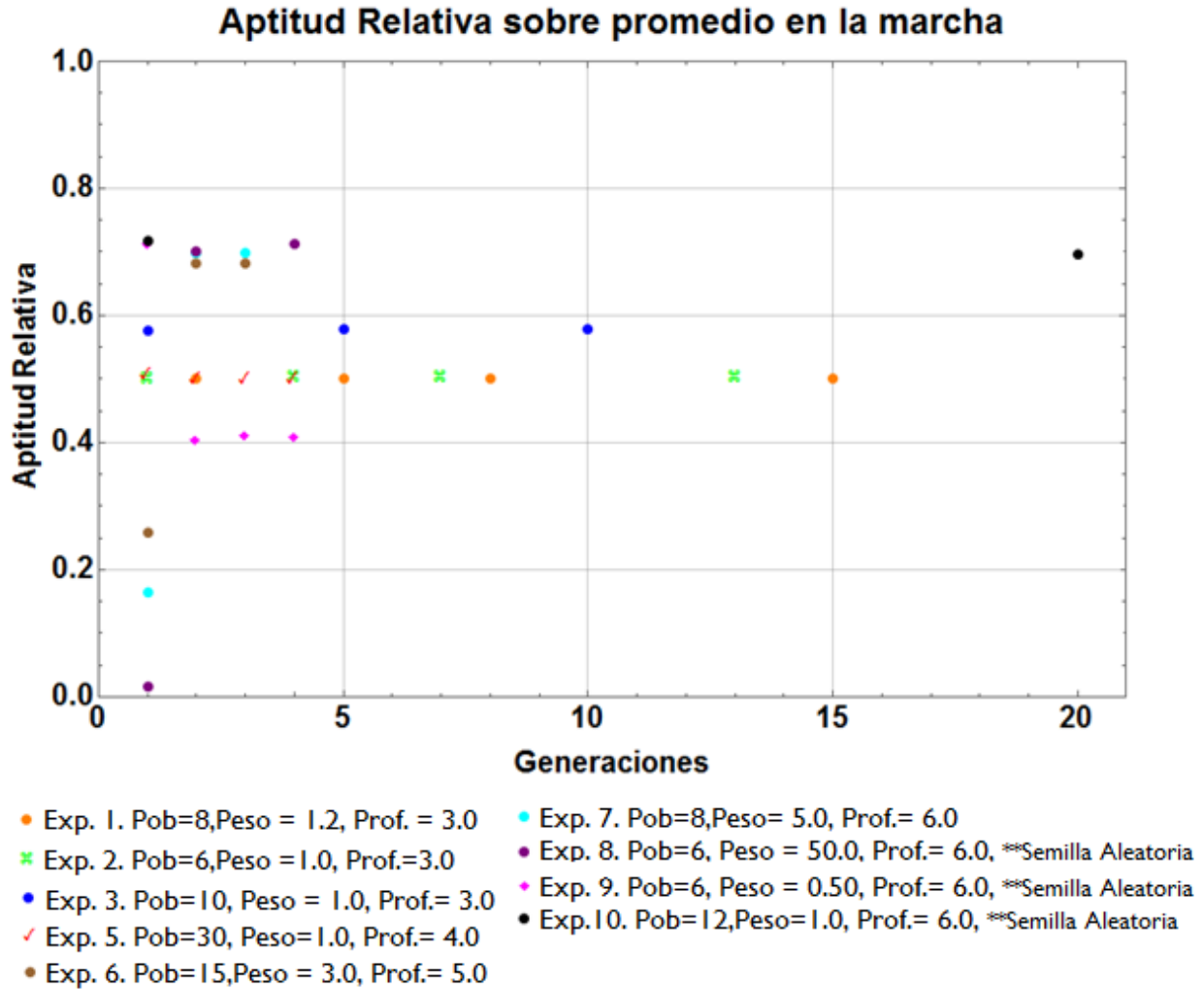


Figura 4.6 Aptitudes normalizadas resultado de la simulación al paso de las generaciones.

En las Figura 4.8 y 4.9 se muestran las trayectorias angulares obtenidas después de la optimización del algoritmo, comparadas con las trayectorias de entrada del AG, donde se observa cómo la evaluación por puntos discretos genera soluciones que se alejan mucho las unas de las otras con el paso del tiempo, y es por eso que fue necesario hacer un ajuste para evitar vibraciones al momento de implementarlo en el robot. En las Figuras 4.10 y 4.11 se puede observar el resultado de las trayectorias suavizadas por el ajuste armónico con diez términos de seno y coseno que se muestra a continuación.

$$\tilde{\theta}_i(t) = \frac{a_0}{2} + \sum_{n=1}^{20} a_n \cos\left(\frac{2\pi n}{100} t\right) + b_n \sin\left(\frac{2\pi n}{100} t\right) \quad (4.1)$$

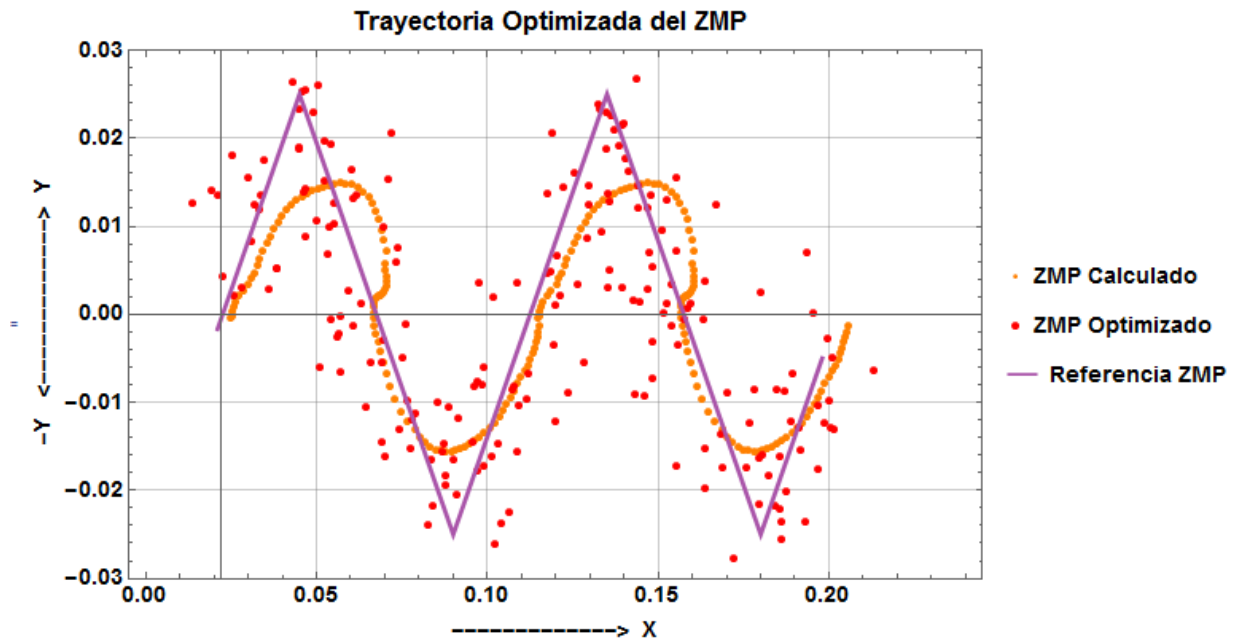


Figura 4.7a ZMP evaluado para cada instante de marcha.

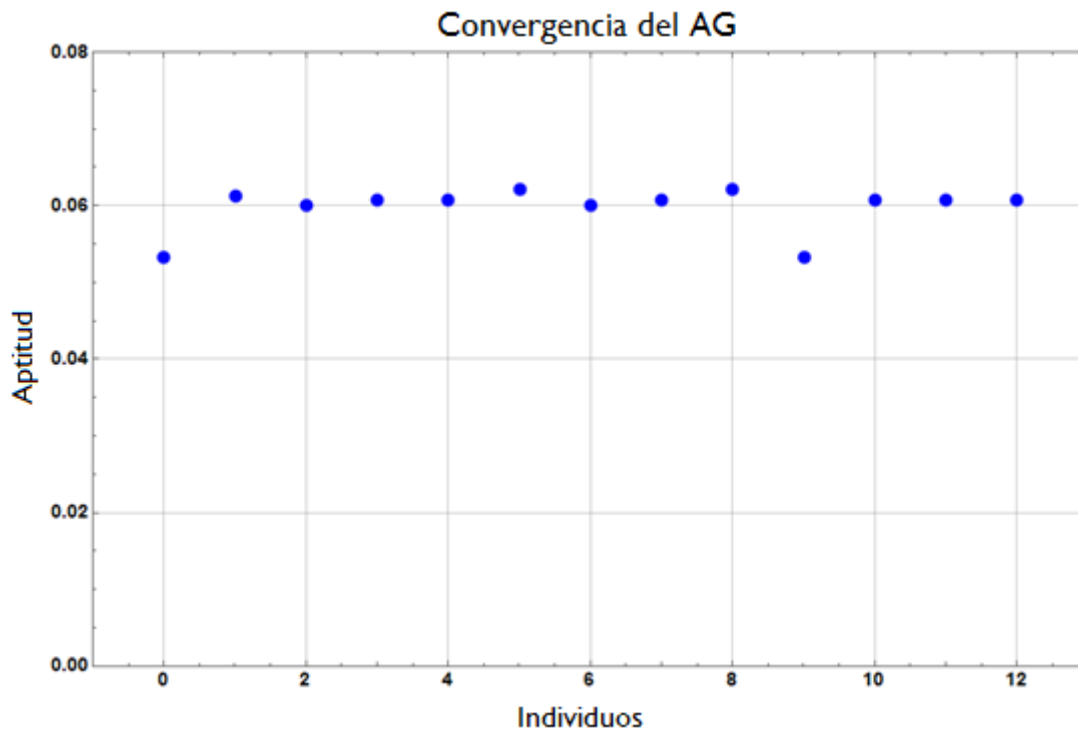
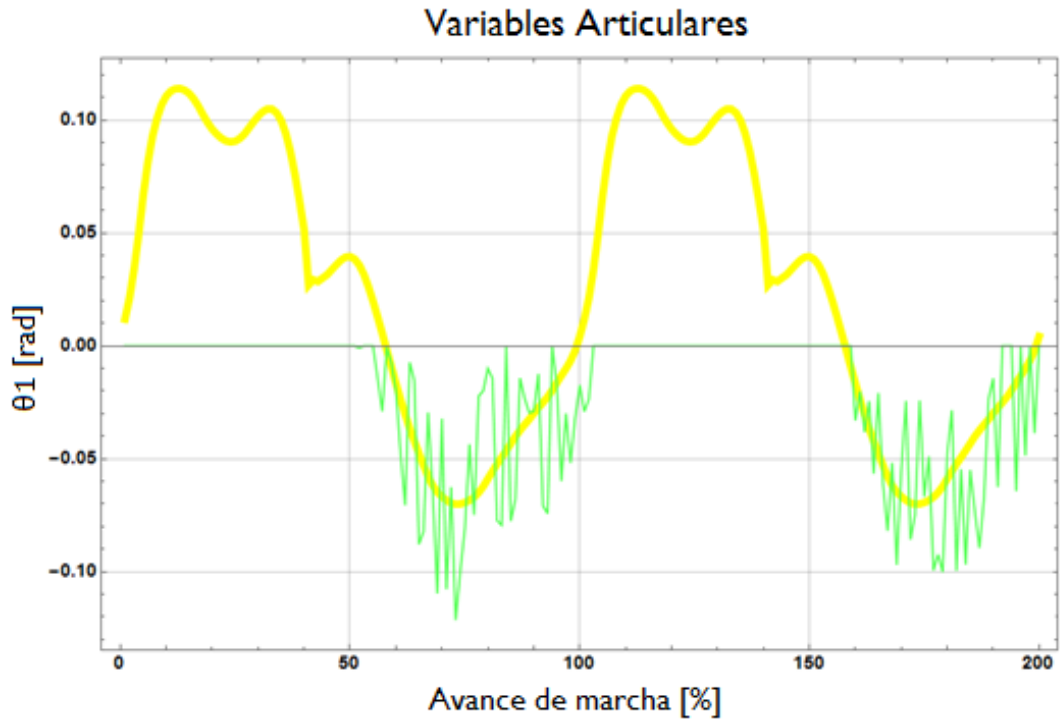
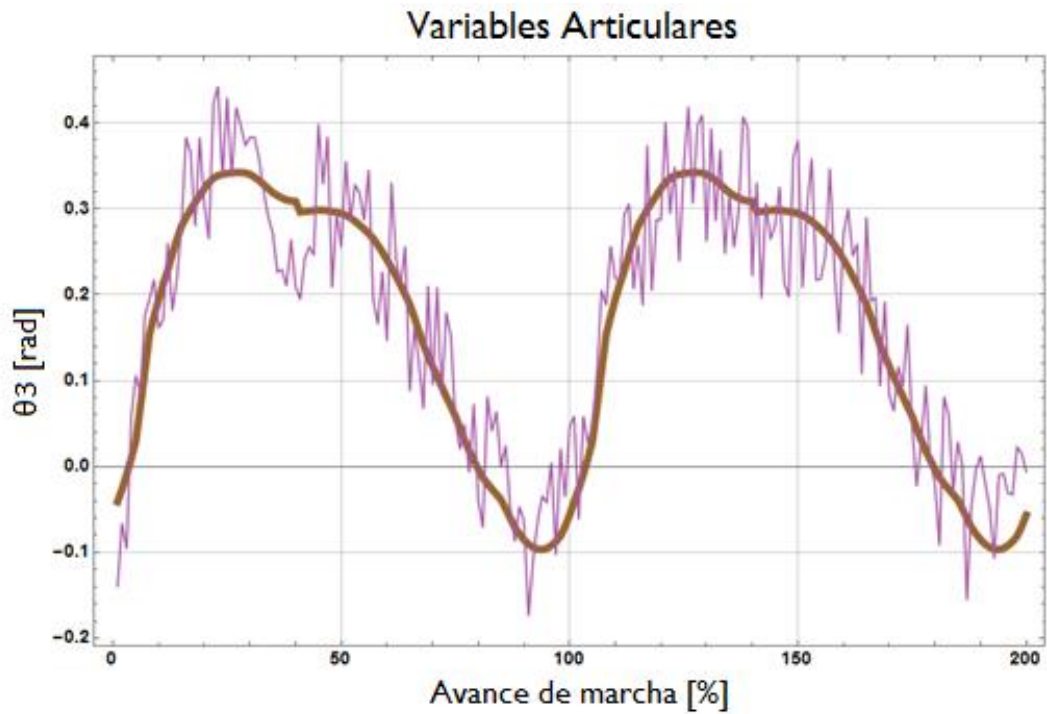


Figura 4.7b Valores de aptitud de la población en el tiempo 100, con una desviación estándar igual a 4.93% del promedio de las aptitudes.

a)

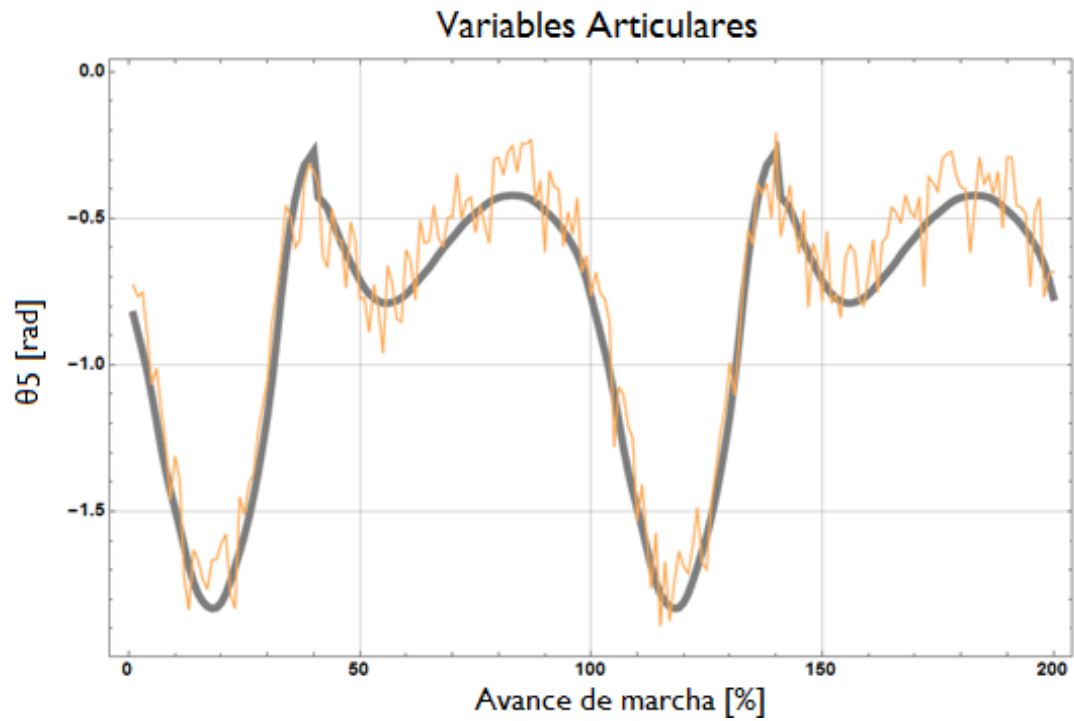


b)

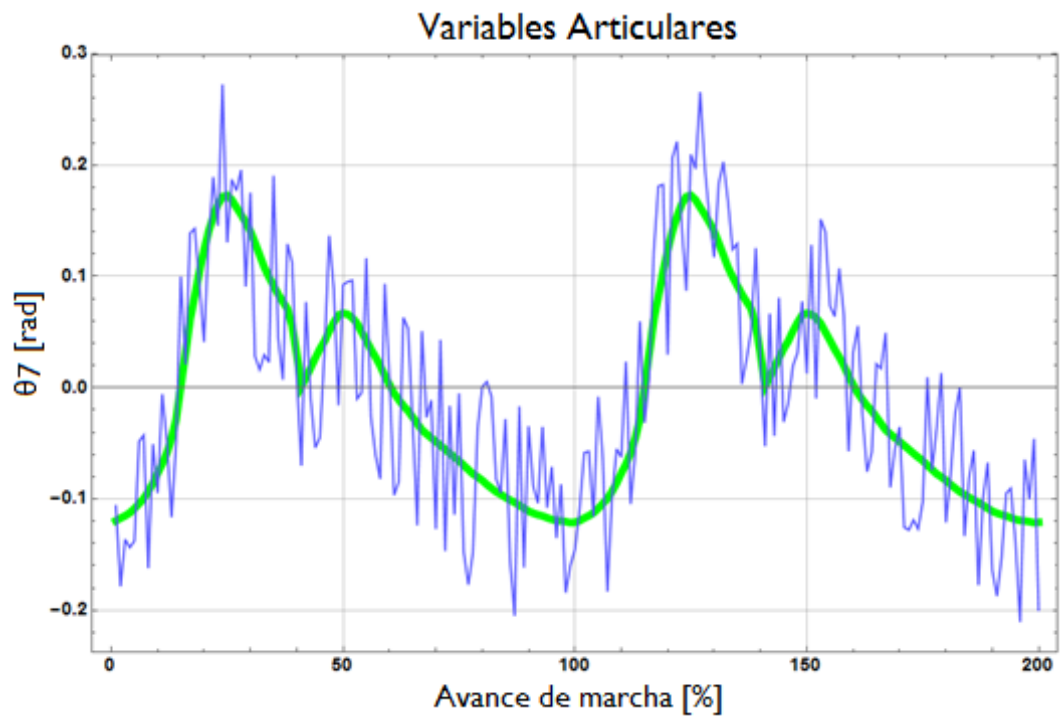


**Figura 4.8** Trayectorias angulares de la pierna derecha, a) trayectoria de abducción-aducción de cadera medida (amarillo) y optimizada (verde), b) trayectoria de flexión-extensión de cadera medida (marrón) y optimizada (morada).

a)

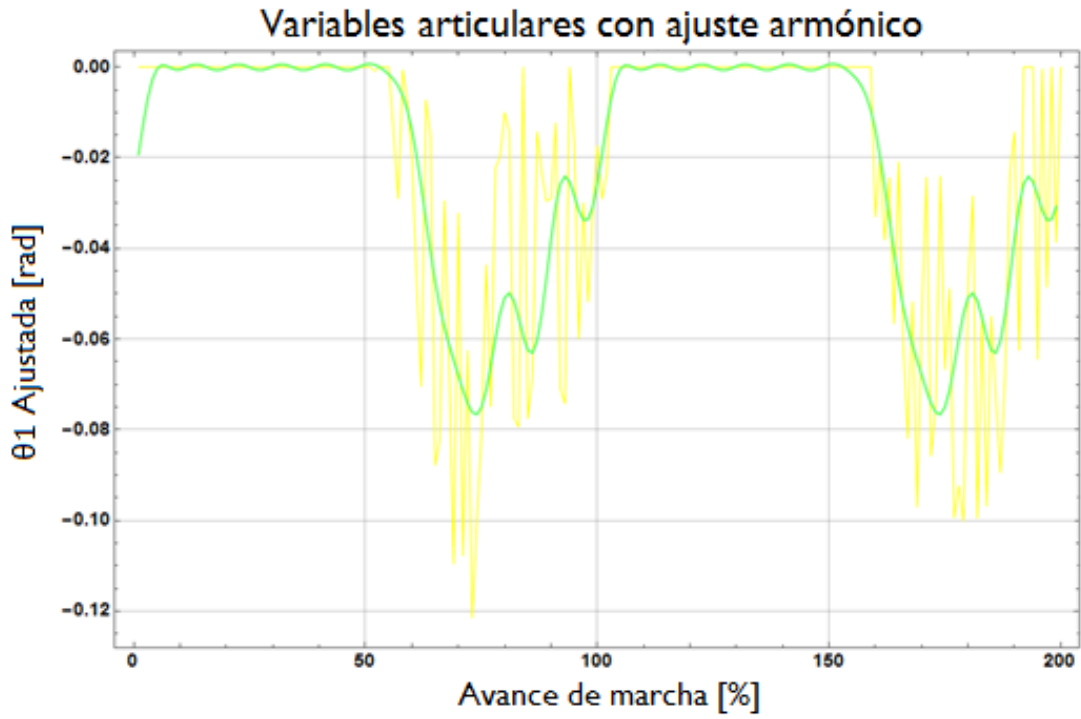


b)

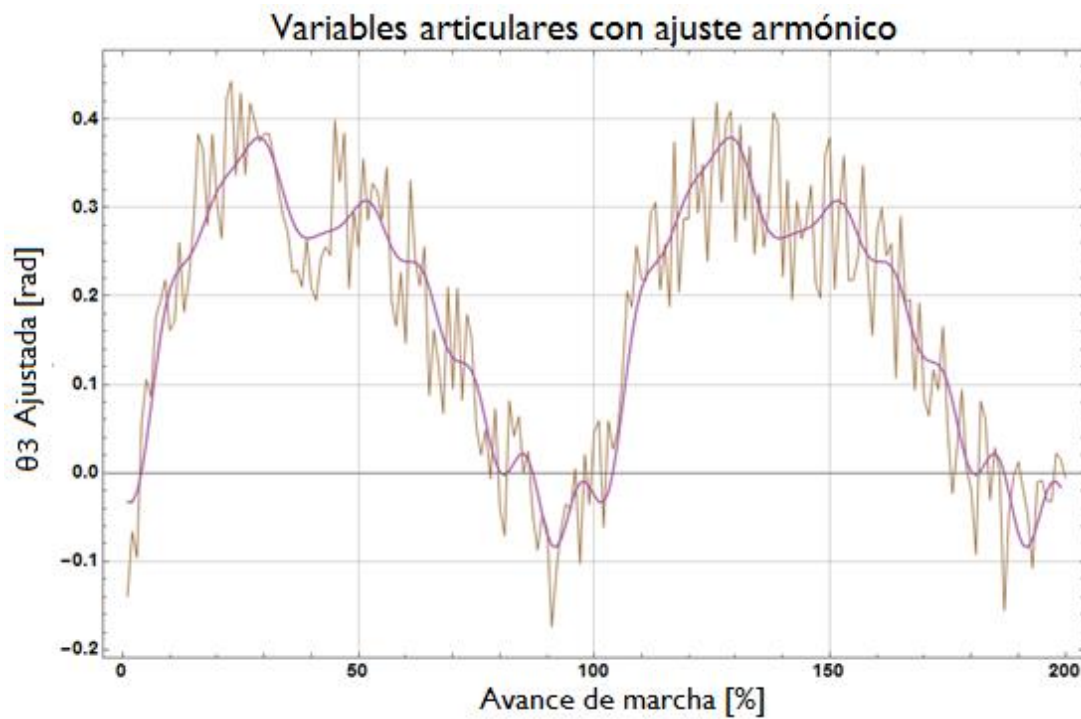


**Figura 4.9** Trayectorias angulares de la pierna derecha, a) trayectoria de flexión-extensión de rodilla medida (gris) y optimizada (anaranjado), b) trayectoria de flexión-extensión de tobillo medida (verde) y optimizada (azul).

a)

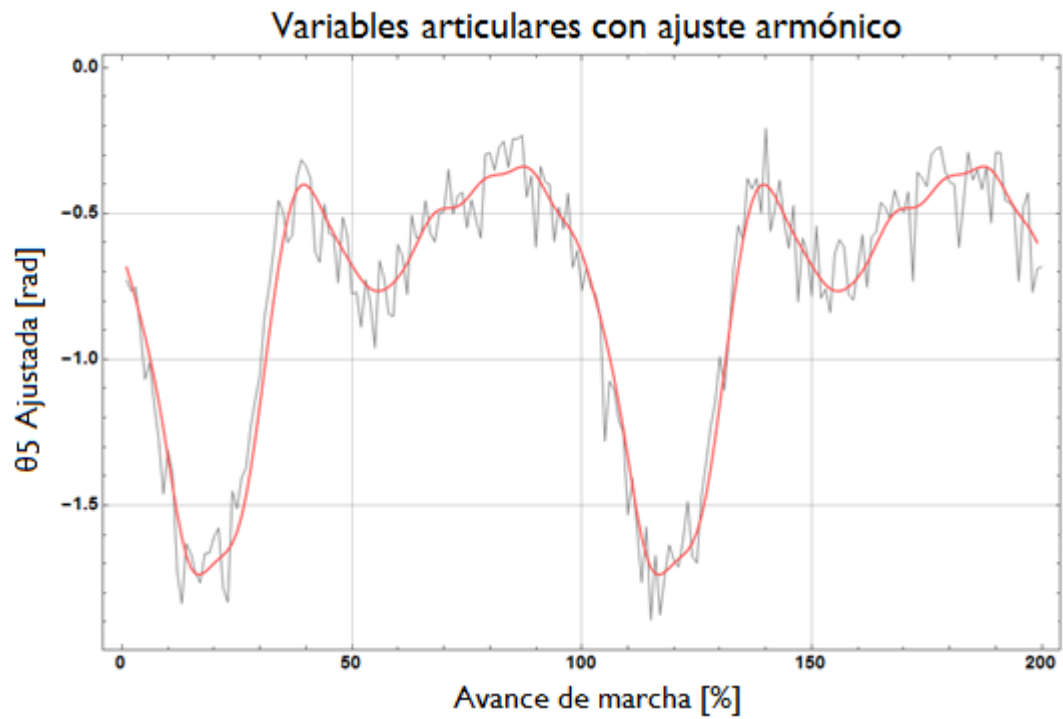


b)

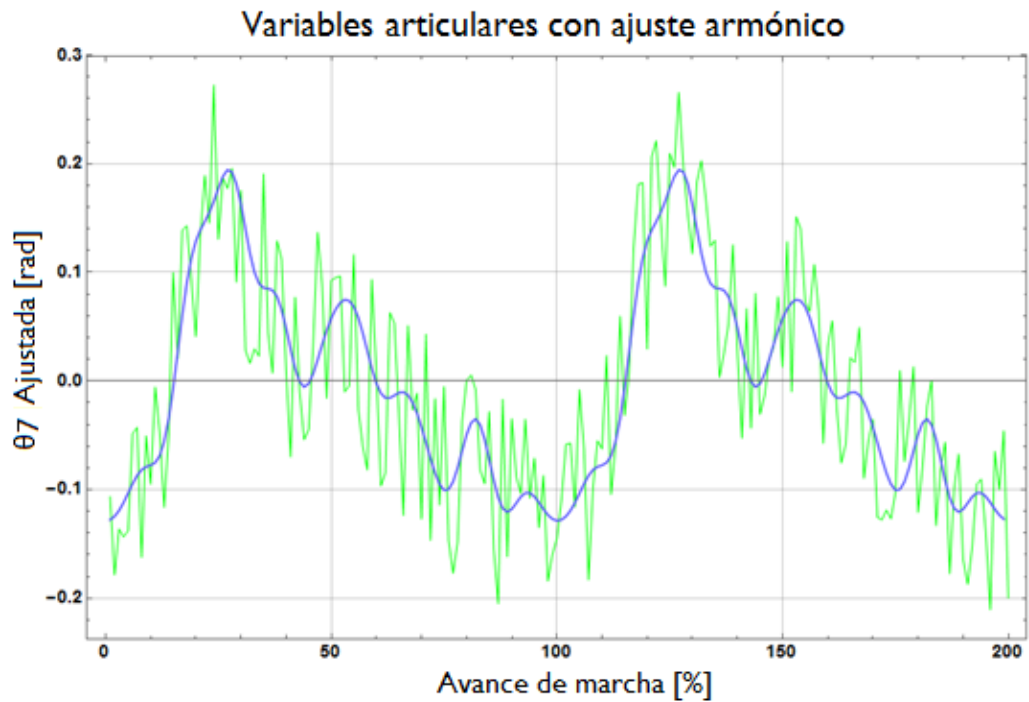


**Figura 4.10** Trayectorias angulares ajustadas de la pierna derecha, a) trayectoria de abducción-aducción de cadera optimizada (amarillo) y ajustada (verde), b) trayectoria de flexión-extensión de cadera optimizada (marrón) y ajustada (morada).

a)



b)



**Figura 4.9** Trayectorias angulares ajustadas de la pierna derecha, a) trayectoria de flexión-extensión de rodilla medida (gris) y optimizada (anaranjado), b) trayectoria de flexión-extensión de tobillo medida (verde) y optimizada (azul).

## Conclusiones

En éste trabajo se logró con éxito el diseño de un algoritmo genético de optimación de las trayectorias angulares de un robot bípedo partiendo de trayectorias medidas de un ser humano, aproximando de manera correcta el *ZMP* calculado con el de referencia y limitando dichos valores de articulaciones según los límites biomecánicos establecidos.

Además, se utilizó de manera adecuada el modelo dinámico simplificado estimando la aceleración del centro de masa por medio de las trayectorias establecidas de la cadera para evitar el cálculo de la dinámica del sistema completo y disminuir así el costo computacional. Aunque el cómputo de la cinemática directa para cada instante alentaba el proceso de evaluación, fue sin duda un ahorro de tiempo comparado con la solución del método completo. También se identificaron de manera analítica los parámetros óptimos para el mejor desempeño del algoritmo, tanto de valores de la función objetivo como del número necesario de generaciones para lograr la convergencia.

Por último, es importante resaltar que el algoritmo cumplió con su función de optimizar la posición del *ZMP* para localizarla dentro de la zona de estabilidad, sin embargo, el desempeño del robot en pruebas reales no consiguió de manera exitosa la caminata, principalmente por la falta de un controlador de lazo cerrado para compensar las perturbaciones existentes en el sistema, tales como las fuerzas externas, inercias, etc. Además, al momento de la ejecución de las pruebas fue notable que la orientación tanto de cadera como de pies es fundamental para poder lograr una caminata exitosa, y al no estar ésta incluida dentro de las variables de optimización o como parámetro dentro de la función objetivo, la solución cinemática dejó dichos valores de efectores finales abiertos a cualquier configuración.

## Trabajo a futuro

Una vez descritas las limitaciones actuales del sistema, es importante mencionar las modificaciones y adiciones que se harán al sistema para poder lograr una marcha exitosa con el robot *Bioid*®. Primero, a la parte de planificación de la marcha, como lo es el algoritmo genético diseñado en este trabajo, se le agregarán los cromosomas respectivos a la orientación de pies y cadera, de manera que permitan optimizar la marcha considerando esos parámetros, sobre todo por el hecho de que una de las principales limitantes para poder hacer caminar de manera correcta al robot era que el contacto del pie de balanceo con el suelo se hacía con alguna esquina de la planta del pie, ocasionando un volcamiento inminente del robot. Se considerará la opción de un algoritmo con búsqueda multi-objetivo en lugar de una sola función de evaluación con penalizaciones, para poder identificar el grupo de individuos que cumpla mejor, ya sea con la parte de estabilidad dinámica o aquellos que desarrollen una caminata más similar a la humana.

## CONCLUSIONES Y TRABAJO A FUTURO

Para analizar el desempeño en el tiempo de cómputo del algoritmo, se propone hacer pruebas en otros lenguajes de programación como *Matlab* o la biblioteca *SciPy* de *Python* para revisar cuál de estos disminuye en mayor medida el tiempo de ejecución del algoritmo genético.

En cuanto a la parte del modelo de pruebas, se espera implementar al robot con dos tipos de sensores, un arreglo de sensores de fuerza resistivos (*FSR*) para cada planta del pie del humanoide y una unidad de medición inercial (*IMU*). Utilizando el arreglo de *FSR* para poder conocer en tiempo real la localización del *ZMP* para ambas fases de soporte, mapeando la fuerza resultante de reacción que se genera contra el suelo; sin embargo, al estar utilizando trayectorias con movimiento de tobillo, es decir, que la planta del pie no se mantiene horizontal, es necesario modificar dicha planta del pie agregando una junta pasiva que emule el movimiento de los dedos del pie, para poder medir así la posición del *ZMP* en todo momento. Y la *IMU* se localizará próximo al centro de masa del robot para reemplazar el valor estimado de aceleración según la trayectoria definida por un valor de aceleración medido directamente en el prototipo real.

Finalmente, tal y como se comentó anteriormente, la implementación de un controlador de lazo cerrado para compensar las perturbaciones del sistema es fundamental para lograr un seguimiento en las trayectorias de referencia, además de que dicho controlador permitirá mantener la estabilidad del robot a lo largo de la marcha bípeda. Según se llegue a optimizar el tiempo de cómputo en el cómputo de trayectorias o no, se pretende recalcular las trayectorias sobre la marcha para seguir permitiendo la similitud en la caminata a la de un humano.

Ciertamente, según C. Chevallereau et al. [16], el ser humano no cumple en todo momento con el criterio del *ZMP* al momento de caminar, y por esa razón es importante revisar si es factible o no el uso de un criterio más robusto de estabilidad para la corrección en línea de la marcha bípeda. Desafortunadamente, criterios como el *CWS* necesitan forzosamente el cálculo completo de la dinámica del sistema, para encontrar las fuerzas de contacto entre los efectores finales y el suelo, ocasionando mayores complicaciones computacionales pero que podrían, sin lugar a dudas, hacer más robusta y versátil la marcha del *Bioloid GP*®.



# Referencias bibliográficas

- [1] F. Sáenz Navarro, “*Generación de trayectorias para el tren inferior del robot humanoide TEO subiendo escaleras*”, Tesis Ing., Universidad Carlos III, Madrid, España, junio 2012.
- [2] M. R. Arbulú Saavedra, “*Stable Locomotion of humanoid robots based on mass concentrated model*”, Ph. D. Thesis., Universidad Carlos III, Madrid, España, octubre 2008.
- [12] V. Duindam, S. Stramigioli, “*Modeling and Control for Efficient Bipedal Walking Robots*”, Ed. Springer, 2008.
- [13] C. Chevallereau, G. Bessonnet, G. Abba, Y. Aoustin, “*Bipedal Robots: Modeling, Design and Walking Synthesis*”, Ed. Wiley, Francia, 2007.
- [14] S. H. Collins, M. Wisse, A. Ruina, “*A Three-Dimensional Passive-Dynamic Walking Robot with two Legs and Knees*”, The International Journal of Robotics Research, Vol. 20, pp. 607-615, 2001.
- [15] D. A. Winter, “*Biomechanics and Motor Control of Human Movement*”, Ed. Wiley, 4th ed., 2009.
- [16] E. R. Westervelt, J. W. Grizzle, C. Chevallereau, J. H. Choi, B. Morris, “*Feedback Control of Dynamic Bipedal Robot Locomotion*”, CRC Press, 1st. ed., 2007.
- [17] H. Hirukawa, S. Hattori, S. Kajita, K. Harada, K. Kaneko, F. Kanehiro, M. Morisawa, S. Nakaoka, “*A Pattern Generator of Humanoid Robots Walking on a Rough Terrain*”, IEEE International Conference of Robotics and Automation, Italy, 2007.
- [18] P. Alves et al., “*Synthesis of a Mechanism for Human Gait Rehabilitation: An Introductory Approach*”, Springer International Publishing Switzerland 2015.
- [20] M. Vukobratíć, B. Borovac, “*Zero-Moment Point – Thirty five years of its life*”, International Journal of Humanoid Robotics, Vol. 1, No. 1, pp. 157-173, 2004.
- [21] M. W. Spong, M. Vidyasagar, “*Robot Dynamics and Control*”, Ed. Wiley, Singapur, 1989.
- [22] R. López-García, E. Rocha-Cózatl, O. Narváez-Aroche, “*Planificación y optimización de ciclos de marcha para un robot bípedo*”, Congreso Nacional de Control Automático, México, 2012.
- [23] R. López-García, “*Generación de patrones de marcha dinámicamente estables para un robot bípedo antropomórfico*”, Tesis Ing., Universidad Nacional Autónoma de México, México, 2012.
- [25] D. E. Goldberg, “*Genetic Algorithms in Search, Optimization, and Machine Learning*”, Ed. Addison-Wesley, U.S.A., 1989.
- [26] M. Jashidi, L. dos Santos Coelho, R. A. Krohling, P.J. Fleming, “*Robust Control Systems with Genetic Algorithms*”, CRC Press, U.S.A., 2002.
- [32] K. Harada, S. Kajita, K. Kaneko, H. Hirukawa, “*An analytical method for real-time gait planning for humanoid robots*”, International Journal of Humanoid Robotics, Vol. 3, No. 1, pp. 1-19, 2006.
- [33] R. Pinto Palmero, “*Análisis cinemático para mejorar la ergonomía humano máquina*”, Tesis de Maestría, Universidad Nacional Autónoma de México, 2016.
- [34] F. Barroso, J. Gómez, M. Rodríguez, A. Peregrín, “*Optimización Evolutiva de la Locomoción de un Robot Bípedo*”, Universidad de Huelva.
- [35] K. Erbaturo, O. Kurt, “*Natural ZMP Trajectories for Biped Robot Reference Generation*”, IEEE Transactions on Industrial Electronics, Vol. 56, No. 3, 2009.

# Referencias electrónicas

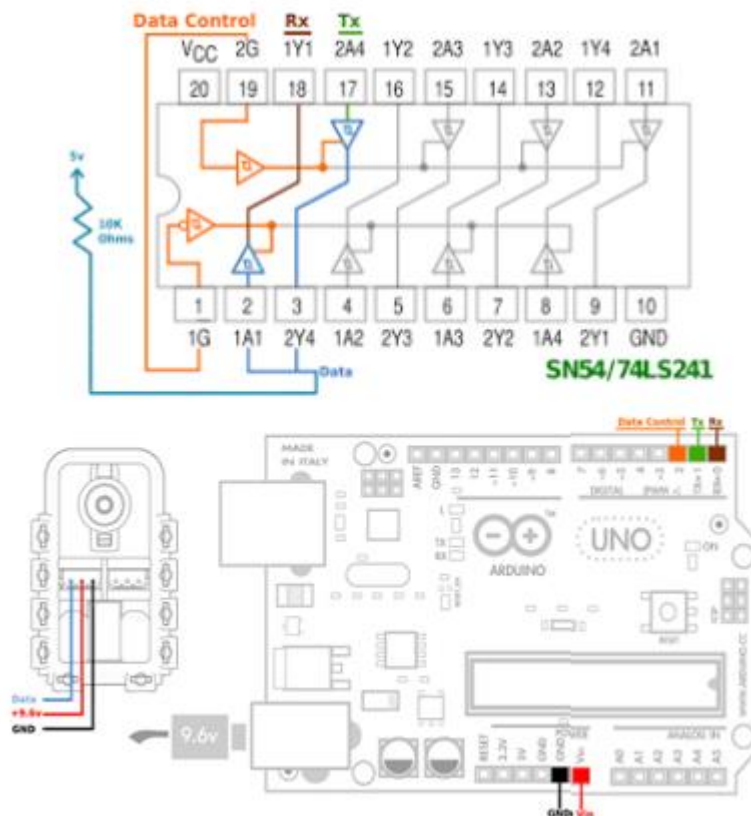
- [3] Takanishi Lab. Website (<http://www.takanishi.mech.waseda.ac.jp>), Waseda University. Consultada en septiembre 2015.
- [4] Honda ASIMO Website (<http://asimo.honda.com/>). Consultada en septiembre 2015.
- [5] Worldwide Honda Website (<http://world.honda.com/ASIMO/technology/index.html>). Consultada en septiembre 2015.
- [6] Institute of Applied Mechanics, TUM Website (<http://www.amm.mw.tum.de/en/research/current-projects/humanoid-robots/johnnie/>). Consultada en septiembre 2015.
- [7] NASA's Robonaut Website (<http://robonaut.jsc.nasa.gov/default.asp>). Consultada en septiembre 2015.
- [8] Boston Dynamics Website ([http://www.bostondynamics.com/robot\\_Atlas.html](http://www.bostondynamics.com/robot_Atlas.html)). Consultada en septiembre 2015.
- [9] Aldebaran Website (<https://www.aldebaran.com/en/humanoid-robot/nao-robot>). Consultada en septiembre 2015.
- [10] 90th Anniversary Special Kawada Industries Website (<http://anniversary.kawada.jp/>). Consultada en septiembre 2015.
- [11] Robotis Bioloid GP Website ([http://en.robotis.com/index/product.php?cate\\_code=121511](http://en.robotis.com/index/product.php?cate_code=121511)). Consultada en junio 2016.
- [19] Medical dictionary Website, transverse plane definition (<http://medical-dictionary.thefreedictionary.com/transverse+plane>). Consultada en junio 2016.
- [24] Temas Genéticos. Archivo, Intelligent Systems Group, Universidad del País Vasco Website (<http://www.sc.ehu.es/ccwbayes/docencia/mmcc/docs/temageneticos.pdf>). Consultada en junio 2016.
- [27] Robotis OpenCM Website ([http://en.robotis.com/index/product.php?cate\\_code=131010](http://en.robotis.com/index/product.php?cate_code=131010)). Consultada en junio 2016.
- [28] Robotis e-Manual, Dynamixel AX-12A Datasheet ([http://support.robotis.com/en/product/dynamixel/ax\\_series/dxl\\_ax\\_actuator.htm](http://support.robotis.com/en/product/dynamixel/ax_series/dxl_ax_actuator.htm)). Consultada en junio 2016.
- [29] Referencia de biblioteca Dynamixel para Arduino (<http://savageelectronics.blogspot.mx/2011/01/arduino-y-dynamixel-ax-12.html>). Consultada en junio 2016.
- [30] Teensy Specifications, PJRC Website (<https://www.pjrc.com/teensy/index.html>). Consultada en junio 2016.
- [31] Wolfram Mathematica® Website (<http://www.wolfram.com/mathematica/?source=nav>). Consultada en junio 2016.

# Apéndices

## Apéndice A: Hardware y firmware

El *Bioloid GP*® cuenta con su propio controlador, protocolo de comunicación y software, lo cual complica el hecho de hacer modificaciones o de adicionar componentes auxiliares. Por esa razón se optó por implementar una tarjeta de desarrollo con los componentes necesarios para cumplir con los objetivos, además de permitirnos reprogramar y agregar sensores que nos permitan un mejor desempeño del robot.

Como parte del trabajo previo a este proyecto de investigación, alumnos del Departamento de Ingeniería Mecatrónica, colaboraron en el diseño y desarrollo de un circuito impreso que implementa un microcontrolador ( $\mu C$ ), un módulo *Bluetooth*®, un sistema de regulación de voltaje, y el circuito complementario para lograr la comunicación con los servomotores Dynamixel®. Éste último es de gran importancia ya que los servos utilizados cuentan con su propio microcontrolador ATmega8 el cual se comunica a través de un protocolo Half-Duplex asíncrono de comunicación con interfaz TTL [28] y se necesita, por lo tanto, un búfer de tres estados o un circuito integrado MAX485 para lograr la comunicación.

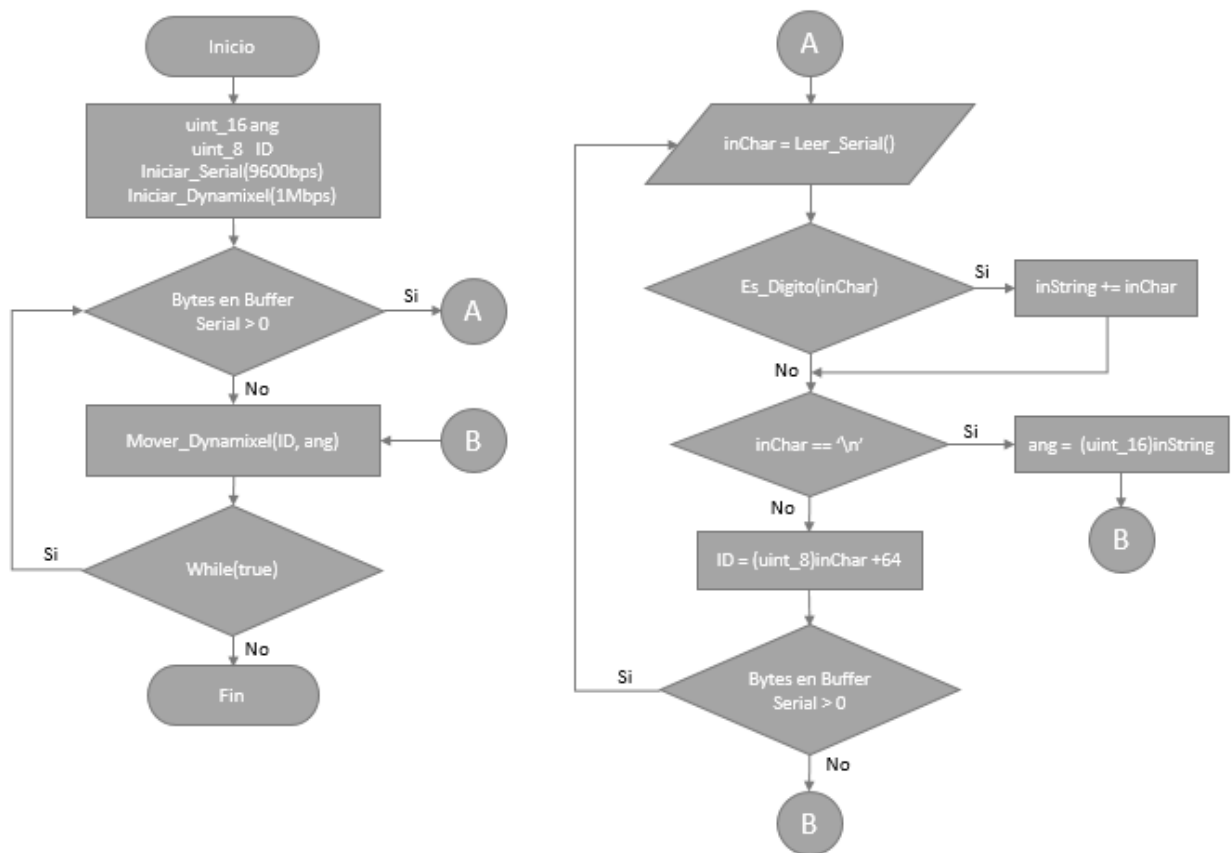


**Figura A.1** Diagrama esquemático de la interfaz de comunicación RS-485. El  $\mu C$  utiliza la señal anaranjada para controlar el flujo entrante o saliente de datos, para escribirlos o leerlos a partir de la UART (Tx y Rx). Así, el servomotor recibe únicamente la señal de alimentación (roja y negra) otorgando 9.6V y la señal de datos (azul) proveniente del búfer.

Para acelerar el proceso de pruebas se utilizó una biblioteca implementada para Arduino la cual implementa el protocolo de comunicación con el servomotor vía serial. Dicha biblioteca fue provista por

[29], donde también da un ejemplo muestra de los circuitos integrados necesarios para su funcionamiento junto con el diagrama esquemático mostrado en la Figura A.1. Tomando eso en cuenta, el microcontrolador seleccionado fue el Teensy 3.1, que cuenta con un procesador ARM Cortex-M4 de 32 bits operando a una frecuencia de 72 MHz, con 256KB de memoria Flash, 64KB de RAM, y una memoria EEPROM de 2KB, 34 pines e entrada/salida, 3 unidades de comunicación UART, dos I2C y una SPI, además de ser compatible con las bibliotecas del software de Arduino [30].

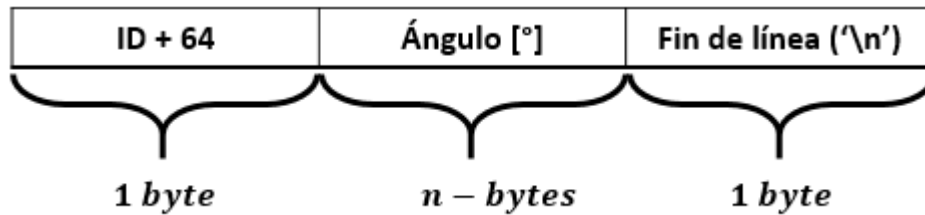
Como firmware del  $\mu C$ , se programó una secuencia de instrucciones que ejecutara como esclavo leyendo de un maestro el identificador del servomotor, el ángulo deseado y la frecuencia de actualización. De ésta manera, lo primero que se realiza en el  $\mu C$  es la inicialización de la comunicación serial con el maestro y con los servomotores Dynamixel®, determinando la velocidad de transmisión (*baudrate*) para cada una de ellas. Una vez hecho esto, en el ciclo principal se realiza una revisión para saber si el maestro envió alguna instrucción, en caso de no hacerlo, el  $\mu C$  se comunica con el último motor con el que mandó una instrucción, reescribiendo el valor del ángulo almacenado en la variable global.



**Figura A.2** Diagrama de flujo del firmware cargado en el microcontrolador.

Pero cuando el  $\mu C$  detecta información en su búfer de entrada, entonces la procesa byte por byte discriminando entre identificador, ángulo deseado y fin de instrucción. Para reconocer al identificador del motor, el maestro envía el número de servo agregándole sesenta y cuatro unidades ( $ID + 64$ ), así el esclavo al momento de recibirlo reconocerá como letra el identificador y podrá hacer la operación inversa para asignar que motor se moverá. Luego de definir el identificador, se prosigue a almacenar todos los bytes entrantes que sean dígitos en una cadena para posteriormente hacer la conversión a un entero de 16 bits no signado (tipo de dato requerido por el protocolo), y terminar el proceso de lectura cuando se detecte el carácter de fin de línea.

En nuestro caso, el programa principal o maestro se ejecuta en el software de *Mathematica*®, que utilizando el lenguaje simbólico incorporado de *Wolfram* le permite la comunicación serial con dispositivo externos, y así enviar las trayectorias directamente después de realizar la optimización. Una vez ejecutado el algoritmo de optimización, se obtienen las trayectorias angulares de cada uno de los motores del robot listas para ser enviadas desde el maestro, interpretadas por el esclavo para después ser ejecutadas por el servo. Ya que el  $\mu C$  está trabajando únicamente como un intérprete entre el alto y bajo nivel de programación, el maestro es aquel que define la frecuencia de actualización de los datos de envío dejando haciendo una espera de  $t$  tiempo antes de reenviar las 10 coordenadas angulares de cada articulación, las cuales son transmitidas de la siguiente forma:



**Figura A.3** Distribución del arreglo de bytes de transmisión. El identificador del servo (ID) toma valores del 9 al 18, el ángulo admite valores entre 0 y 180°, y el byte de término corresponde al número 10 en código ASCII.

El número de ciclos o repeticiones que se necesitan las dicta directamente el maestro en su algoritmo de planificación de marcha y todo esto sirve para maximizar la tasa de transmisión de datos entre el esclavo y maestro dado que el esclavo al tener una frecuencia de reloj menor de 100 veces la del maestro necesita reducir el número de instrucciones ejecutadas para optimizar su desempeño.

Apéndice B.1: Funciones de generación de población inicial con restricciones

```
(***** Población Inicial *****)
GeneraPobReferencia[t_] := {θ1dT[t], θ2dT[t], θ3dT[t], θ4dT[t],
                            θ1iT[t], θ2iT[t], θ3iT[t], θ4iT[t],
                            xCad[t], yCad[t], zCad[t]};

GeneraCromosomas[t_, Ind_] := {
    SeedRandom[Ind*1234]; RandomReal[{θ1dT[t]-bandCadAA, θ1dT[t] + bandCadAA}],
    SeedRandom[Ind*1234]; RandomReal[{θ2dT[t]-bandCadFE, θ2dT[t] + bandCadAA}],
    SeedRandom[Ind*1234]; RandomReal[{θ3dT[t]-bandRod, θ3dT[t] + bandRod}],
    SeedRandom[Ind*1234]; RandomReal[{θ4dT[t]-bandTob, θ4dT[t] + bandTob}],
    SeedRandom[Ind*1234]; RandomReal[{θ1iT[t]-bandCadAA, θ1iT[t] + bandCadAA}],
    SeedRandom[Ind*1234]; RandomReal[{θ2iT[t]-bandCadFE, θ2iT[t] + bandCadAA}],
    SeedRandom[Ind*1234]; RandomReal[{θ3iT[t]-bandRod, θ3iT[t] + bandRod}],
    SeedRandom[Ind*1234]; RandomReal[{θ4iT[t]-bandTob, θ4iT[t] + bandTob}],
    SeedRandom[Ind*1234]; RandomReal[{xCad[t]-bandXcad, xCad[t] + bandXcad}],
    SeedRandom[Ind*1234]; RandomReal[{yCad[t]-bandYcad, yCad[t] + bandYcad}],
    SeedRandom[Ind*1234]; RandomReal[{zCad[t]-bandZcad, zCad[t] + bandZcad}];

(***** Restricción articular de Rodillas conforme a límites humanos *****)
RestriccionRodDer[Pob_] := {
    Pob[[1]], Pob[[2]], 0, Pob[[4]], Pob[[5]], Pob[[6]], Pob[[7]], Pob[[8]], Pob[[9]], Pob[[10]], Pob[[11]]};
RestriccionRodIz[Pob_] := {
    Pob[[1]], Pob[[2]], Pob[[3]], Pob[[4]], Pob[[5]], Pob[[6]], 0, Pob[[8]], Pob[[9]], Pob[[10]], Pob[[11]]};
RestriccionRod[Pob_] := {
    Pob[[1]], Pob[[2]], 0, Pob[[4]], Pob[[5]], Pob[[6]], 0, Pob[[8]], Pob[[9]], Pob[[10]], Pob[[11]]};

(***** Restricción articular de Cadera conforme a límites humanos *****)
RestriccionCadDer[Pob_] := {
    0, Pob[[2]], Pob[[3]], Pob[[4]], Pob[[5]], Pob[[6]], Pob[[7]], Pob[[8]], Pob[[9]], Pob[[10]], Pob[[11]]};
RestriccionCadIz[Pob_] := {
    Pob[[1]], Pob[[2]], Pob[[3]], Pob[[4]], 0, Pob[[6]], Pob[[7]], Pob[[8]], Pob[[9]], Pob[[10]], Pob[[11]]};
RestriccionCad[Pob_] := {
    0, Pob[[2]], Pob[[3]], Pob[[4]], 0, Pob[[6]], Pob[[7]], Pob[[8]], Pob[[9]], Pob[[10]], Pob[[11]]};
```

Apéndice B.2: Funciones de selección, cruce y mutación

```

(***** Selección por torneo binario & cruce con representación real *****)
For[ ind= 0, ind≤ Pob, ind+= 2,
  Conc1= RandomInteger[{0,Pob}];
  Conc2= RandomInteger[{0,Pob}];
  If[(FitActual[Conc1] ≤ FitActual[Conc2],
    Padre1 = PobActual[Conc1],
    Padre1 = PobActual[Conc2]];
  Conc3= RandomInteger[{0,Pob}];
  Conc4= RandomInteger[{0,Pob}];
  If[(FitActual[Conc3] ≤ FitActual[Conc4],
    Padre2 = PobActual[Conc3],
    Padre2 = PobActual[Conc4]];
  Cruce = RandomReal[{0,1}];
  If[(Cruce < Pc),
    For[gen = 1, gen≤ 11, gen++,
      PobSiguiete[ind,gen] = μ*Padre1[[gen]] + (1-μ)*Padre2[[gen]];
      PobSiguiete[ind+1,gen] = (1-μ)*Padre1[[gen]] + μ*Padre2[[gen]];
    ],
    PobSiguiete[ind] = Padre1;
    PobSiguiete[ind+1] = Padre2 ];
];

(***** Mutación con número aleatorio *****)
For[ind = 0, ind ≤ Pob, ind++,
  Mutacion = randomReal[{0,1}];
  If[(Mutacion < Pm),
    CromosomaMuta= RandomInteger[{0,11}];
    Pob Siguiete[ind,CromosomaMuta] = MutacionRepReal[CromosomaMuta,av]];
];

MutacionRepReal[Cromosoma_,t_] := Switch[ (Cromosoma),
  1, SeedRandom[Cromosoma*1234]; RandomReal[{01dT[t]-bandCadAA, 01dT[t] + bandCadAA}],
  2, SeedRandom[Cromosoma*1234]; RandomReal[{02dT[t]-bandCadFE, 02dT[t] + bandCadAA}],
  3, SeedRandom[Cromosoma*1234]; RandomReal[{03dT[t]-bandRod, 03dT[t] + bandRod}],
  4, SeedRandom[Cromosoma*1234]; RandomReal[{04dT[t]-bandTob, 04dT[t] + bandTob}],
  5, SeedRandom[Cromosoma*1234]; RandomReal[{01iT[t]-bandCadAA, 01iT[t] + bandCadAA}],
  6, SeedRandom[Cromosoma*1234]; RandomReal[{02iT[t]-bandCadFE, 02iT[t] + bandCadAA}],
  7, SeedRandom[Cromosoma*1234]; RandomReal[{03iT[t]-bandRod, 03iT[t] + bandRod}],
  8, SeedRandom[Cromosoma*1234]; RandomReal[{04iT[t]-bandTob, 04iT[t] + bandTob}],
  9, SeedRandom[Cromosoma*1234]; RandomReal[{xCad[t]-bandXcad, xCad[t] + bandXcad}],
  10, SeedRandom[Cromosoma*1234]; RandomReal[{yCad[t]-bandYcad, yCad[t] + bandYcad}],
  11, SeedRandom[Cromosoma*1234]; RandomReal[{zCad[t]-bandZcad, zCad[t] + bandZcad}];

```

Apéndice B.3: Funciones para cálculo de la función objetivo

```

(***** Función de evaluación: Error del ZMP y Penalización de Pie ***)
EvaluacionIndividuos[t_, pob_] :=
    (√((ZMPx[t]-ZMPinst[t,pob][[1]])^2 + (ZMPy[t]-ZMPinst[t,pob][[2]])^2))
    - W*((DistDerechoRef[t,pob]+DistIzquierdoRef[t,pob])/2));

(***** Cálculo del ZMP instantáneo *****)
ZMPinst[t_, pob_] := {
    (1/g ((CoMinst[t,pob][[3]]*axCad[t])-(CoMinst[t,pob][[1]]*azCad[t]))+CoMinst[t,pob][[1]]),
    (1/g ((CoMinst[t,pob][[2]]*azCad[t])-(CoMinst[t,pob][[3]]*ayCad[t]))+CoMinst[t,pob][[2]])};

(***** Cálculo del CoM instantáneo *****)
CoMinst[t_, pob_] := ((M[1,1]/Mtotal)*(T3D[CCad.Inst[t,pob].Txyz[xcg,ycg,zcg]).n)+
    (M[2,1]/Mtotal)*(T3D[CD02Inst[t,pob].Txyz[xdg1,ydg1,-zdg1]).n)+
    (M[3,1]/Mtotal)*(T3D[CD04Inst[t,pob].Txyz[-xdg2,-ydg2,zdg2]).n)+
    (M[4,1]/Mtotal)*(T3D[CD07Inst[t,pob].Txyz[xdg3,ydg3,zdg3]).n)+
    (M[5,1]/Mtotal)*(T3D[CD08Inst[t,pob].Txyz[-xdg4,ydg4,-zdg4]).n)+
    (M[6,1]/Mtotal)*(T3D[CD09Inst[t,pob].Txyz[-xdg5,-ydg5,-zdg5]).n)+
    (M[7,1]/Mtotal)*(T3D[CI02Inst[t,pob].Txyz[xig1,-yig1,-zig1]).n)+
    (M[8,1]/Mtotal)*(T3D[CI04Inst[t,pob].Txyz[-xig2,yig2,-zig2]).n)+
    (M[9,1]/Mtotal)*(T3D[CI07Inst[t,pob].Txyz[-xig3,yig3,zig2]).n)+
    (M[10,1]/Mtotal)*(T3D[CI08Inst[t,pob].Txyz[-xig4,-yig4,-zig4]).n)+
    (M[11,1]/Mtotal)*(T3D[CI09Inst[t,pob].Txyz[-xig5,yig5,-zig5]).n);

(***** Funciones de Cinemática Directa *****)
CCadInst[t_, pob_] := Txyz[pob[[9]], pob[[10]],pob[[11]]];

CD02Inst[t_, pob_] := CCadInst[t,pob].(TDH[-90*Degree,0,d0,-90*Degree).(TDH[pob[[1]],0,d1,0]);
CD04Inst[t_, pob_] := CD02Inst[t,pob].(TDH[90*Degree,0,d2,90*Degree).(TDH[pob[[2]],0,d3,0]);
CD07Inst[t_, pob_] := CD04Inst[t,pob].(TDH[-90*Degree,0,d4,0).
    (TDH[pob[[3]]+123.8*Degree, 0 ,d5,0]).(TDH[-33.8*Degree,0,d6,0]);
CD08Inst[t_, pob_] := CD07Inst[t,pob].(TDH[pob[[4]],0,0,-90*Degree);
CD09Inst[t_, pob_] := CD08Inst[t,pob].(TDH[0,0,d8,0]);

CI02Inst[t_, pob_] := CCadInst[t,pob].(TDH[90*Degree,0,i0,90*Degree).(TDH[pob[[5]],0,i1,0]);
CI04Inst[t_, pob_] := CI02Inst[t,pob].(TDH[-90*Degree,0,i2,-90*Degree).(TDH[pob[[6]],0,i3,0]);
CI07Inst[t_, pob_] := CI04Inst[t,pob].(TDH[90*Degree,0,i4,0).
    (TDH[pob[[7]]-123.8*Degree, 0 ,i5,0]).(TDH[33.8*Degree,0,i6,0]);
CI08Inst[t_, pob_] := CI07Inst[t,pob].(TDH[pob[[8]],0,0,90*Degree);
CI09Inst[t_, pob_] := CI08Inst[t,pob].(TDH[0,0,i8,0);

```



## APÉNDICES

(\*\*\*\*\* Localización del talón \*\*\*\*\*)

```
PieInst[t_, pob_] := T3D[CD09Inst[t,pob].n];  
PieIzInst[t_, pob_] := T3D[CI09Inst[t,pob].n];
```

(\*\*\*\*\* Distancias de los pies a las trayectorias de referencia \*\*\*\*\*)

```
DistDerechoRef[t_, pob_] :=
```

```
 $\sqrt{((\text{PieInst}[t,\text{pob}][[1]] - \text{xSwi}[t])^2 + (\text{PieInst}[t,\text{pob}][[2]] - \text{ySwi}[t])^2 + (\text{PieInst}[t,\text{pob}][[3]] - \text{zSwi}[t])^2)}$ ;
```

```
DistIzquierdoRef[t_, pob_] :=
```

```
 $\sqrt{((\text{PieIzInst}[t,\text{pob}][[1]] - \text{xSwiI}[t])^2 + (\text{PieIzInst}[t,\text{pob}][[2]] - \text{ySwiI}[t])^2 + (\text{PieIzInst}[t,\text{pob}][[3]] - \text{zSwiI}[t])^2)}$ ;
```