

**SEP**

SECRETARÍA DE  
EDUCACIÓN PÚBLICA



TECNOLÓGICO NACIONAL DE MEXICO  
Instituto Tecnológico de la Laguna

**DIVISIÓN DE ESTUDIOS DE POSGRADO E INVESTIGACIÓN**

**“Control de marcha del robot humanoide  
NAO”**

POR

**Ing. Gerardo Manuel Ruiz López**

**TESIS**

**PRESENTADA COMO REQUISITO PARCIAL PARA OBTENER EL  
GRADO DE MAESTRO EN CIENCIAS EN INGENIERÍA ELÉCTRICA**

**DIRECTOR DE TESIS**

**Dr. Víctor Adrián Santibáñez Dávila**

**CODIRECTOR DE TESIS**

**M. C. Edmundo Javier Ollervides Vázquez**

**ISSN: 0188-9060**



**RIITEC: (18)-TMCIE-2015**

Torreón, Coahuila. México

Diciembre 2015

"2015, Año del Generalísimo José María Morelos y Pavón"

Torreón, Coah., **11/Diciembre/2015**  
Dependencia: DEPI/CPCIE  
Oficio: DEPI/CPCIE/236/2015  
Asunto: Autorización de impresión  
de tesis.

**C. Gerardo Manuel Ruiz López**  
**CANDIDATO AL GRADO DE MAESTRO EN CIENCIAS EN INGENIERÍA ELÉCTRICA.**  
**PRESENTE**

Después de haber sometido a revisión su trabajo de tesis titulado:

**"Control de marcha del robot humanoide NAO"**

Habiendo cumplido con todas las indicaciones que el jurado revisor de tesis hizo, se le comunica que se le concede la autorización con número de registro **RIITEC: (18)-TMCIE-2015**, para que proceda a la impresión del mismo.

**ATENTAMENTE**  
EDUCACIÓN TECNOLÓGICA. FUENTE DE INNOVACIÓN

**DR. JOSÉ LUIS MEZA MEDINA**  
Jefe de la División de Estudios de Posgrado e Investigación  
del Instituto Tecnológico de la Laguna

JLMM/RHSR



# Dedicatoria

*A mi esposa Isamar Martínez Pacheco. A mis padres Gerardo Ruiz y Martha López. A mi familia y amigos quienes de alguna u otra forma me han apoyado y han estado siempre a mi lado.*

# Agradecimientos

Agradezco al Instituto Tecnológico de La Laguna por hacer posible esta clase de estudios de posgrado. A la DGEST (ahora Tecnológico Nacional de México) por apoyar los proyectos de investigación de los cuales esta tesis forma parte. Al CONACYT por la financiación del proyecto 134534 y además de apoyarme durante este tiempo al otorgarme una beca de estudios.

Agradezco a mi director de tesis Dr. Víctor Adrián Santibáñez Dávila, por su entera confianza y gran apoyo, además de sus valiosos consejos y enseñanzas durante esta etapa. Al Dr. Víctor Samuel de León Gómez por su gran apoyo y amistad durante la realización de este trabajo. Al Dr. José Alfonso Pámanes García, quien compartió gran parte de sus conocimientos y experiencias para llevar a cabo este proyecto. A Mi coasesor el MC. Edmundo Javier Olivides Vázquez por sus comentarios y observaciones que han permitido enriquecer este trabajo.

Quisiera agradecer a todo el personal docente del área de posgrado del Instituto Tecnológico de la Laguna y a todos aquellos investigadores de quienes he aprendido lo bueno y lo malo de esta maravillosa profesión. A todos ellos mi cariño, admiración y respeto.

Agradezco a mis compañeros de posgrado por su amistad, compañerismo con los cuales compartí muchas alegrías, experiencias y consejos.

Agradezco a mi familia quienes han sido pieza fundamental en el transcurso de mi vida. A mis hermanas Itzel Ruiz, Arleth Ruiz, y Nallely Ruiz y muy especialmente a mis padres Gerardo Ruiz y Martha López por todo su cariño y enseñanzas que me han dado en toda mi vida.

Agradezco a mi amiga y esposa Isamar Martínez Pacheco por todo su apoyo, paciencia y comprensión que me ha dado durante este proyecto de vida, pero sobre todo por su amor incondicional y compañerismo.

Finalmente agradezco a DIOS por todas las bendiciones que he recibido durante mi vida.



# Resumen

Este trabajo está orientado al estudio de patrones de marcha, modelado y control para la realización del caminado del robot humanoide NAO. Primeramente se realiza el estudio del estado del arte de robots humanoides, el cual sirve de base para el desarrollo de nuevas propuestas de caminado y control. Se presentan los modelados de la cinemática directa e inversa, así como también el modelado dinámico del robot mediante las ecuaciones de Euler Lagrange. Se propone un patrón de marcha, el cual está basado en movimientos traslacionales de la pelvis y del pie libre, que especifica un caminado factible del robot NAO. Para la realización del estudio aquí presentado, se desarrolla un diagrama alambre en Matlab. Además, se realiza un control de marcha mediante seguimiento de trayectorias usando un controlador PID, y finalmente se hacen pruebas experimentales con el robot NAO.

# Abstract

This work focuses into the survey of gait patterns, modeling and control of walking gaits of the humanoid robot NAO. First, the state of art of humanoid robots is reviewed, which it is useful for the development of new walking and control proposes. The forward and inverse kinematics modeling as well as dynamic modeling by Euler Lagrange equations of the robot are presented. A gait pattern which is based on translational movement of the pelvis and the free foot, which specifies a feasible walked of the NAO is proposed. To carry out the study presented here is developed a wire diagram in Matlab. Moreover, a gait control is performed by path tracking using a PID controller and finally tests are done with the NAO robot.

# Índice general

<b>1. Introducción</b>	<b>1</b>
1.1. Antecedentes	2
1.2. Desarrollo de robots humanoides	2
1.3. Locomoción dinámica pasiva	5
1.4. Locomoción dinámica activa	6
1.5. El método de dinámica cero híbrida	7
1.6. Objetivo de la Tesis	7
1.6.1. Objetivos específicos	8
1.7. Contenido de la tesis	9
<b>2. Descripción del robot humanoide NAO</b>	<b>10</b>
2.1. Marcha del robot humanoide NAO	10
2.2. Movimiento	11
2.3. Visión	12
2.4. Audio	13
2.5. Sonares	14
2.6. Conectividad	14
2.7. Parámetros geométricos del robot NAO	14
<b>3. Modelado del robot NAO</b>	<b>16</b>
3.1. Identificación de la cadena cinemática	17
3.2. Modelo cinemático directo	17
3.3. Modelado cinemático inverso de posición	24
3.3.1. Modelo cinemático inverso de posición de los brazos izquierdo y derecho	24
3.3.2. Modelado cinemático inverso de posición de las piernas	28
3.4. Modelo de velocidad y aceleración de las piernas de robot NAO	31
3.5. Modelado dinámico mediante las ecuaciones de Euler-Lagrange	35
<b>4. Patrón de Marcha del Robot NAO</b>	<b>42</b>
<b>5. Control de marcha mediante seguimiento de trayectorias</b>	<b>45</b>
5.1. Ley de control PID	45

## ÍNDICE GENERAL

---

6. Simulaciones	48
7. Evaluación experimental	66
8. Conclusión	75
Bibliografía	78
9. Anexos	83



# Índice de figuras

1.1. Robots bípedos	3
1.2. Robots Humanoides	4
1.3. Mecanismo inventado por McGeer	5
1.4. Robots bípedos	7
1.5. Robot RABBIT	8
2.1. Robot NAO	11
2.2. Descripción del Robot NAO	12
2.3. Cámaras del robot NAO	13
2.4. Micrófonos del robot NAO	13
2.5. Sensores del robot NAO	14
2.6. Parámetros geométricos de los brazos del robot NAO	15
2.7. Parámetros geométricos del robot NAO	15
3.1. Marcos asignados a los eslabones del robot NAO	18
3.2. Diagrama alambre del robot NAO	24
3.3. Posiciones de los centros de masa del robot NAO	36
3.4. Modelo geométrico del robot NAO	37
4.1. Marcos de las piernas en cada apoyo	42
5.1. Diagrama a bloques del control de marcha mediante seguimiento de trayectorias del robot NAO	46
5.2. Control de seguimiento de trayectorias	47
6.1. Vista frontal del caminado del robot NAO	50
6.2. Vista lateral del caminado del robot NAO	50
6.3. Posiciones articulares de $\theta_1$ a $\theta_6$ durante el caminado del robot NAO	51
6.4. Posiciones articulares de $\theta_7$ a $\theta_{12}$ durante el caminado del robot NAO	51
6.5. Velocidades articulares de $\dot{\theta}_1$ a $\dot{\theta}_6$ durante el caminado del robot NAO	52
6.6. Velocidades articulares de $\dot{\theta}_7$ a $\dot{\theta}_{12}$ durante el caminado del robot NAO	52
6.7. Aceleraciones articulares de $\ddot{\theta}_1$ a $\ddot{\theta}_6$ durante el caminado del robot NAO	53
6.8. Aceleraciones articulares de $\ddot{\theta}_7$ a $\ddot{\theta}_{12}$ durante el caminado del robot NAO	53

6.9. Posición articular $\theta_1$ contra la posición deseada $\theta_{1d}$ de la pierna durante la tarea de seguimiento.	54
6.10. Posición articular $\theta_2$ contra la posición deseada $\theta_{2d}$ de la pierna durante la tarea de seguimiento.	54
6.11. Posición articular $\theta_3$ contra la posición deseada $\theta_{3d}$ de la pierna durante la tarea de seguimiento.	54
6.12. Posición articular $\theta_4$ contra la posición deseada $\theta_{4d}$ de la pierna durante la tarea de seguimiento.	55
6.13. Posición articular $\theta_5$ contra la posición deseada $\theta_{5d}$ de la pierna durante la tarea de seguimiento.	55
6.14. Posición articular $\theta_6$ contra la posición deseada $\theta_{6d}$ de la pierna durante la tarea de seguimiento.	55
6.15. Posición articular $\theta_7$ contra la posición deseada $\theta_{7d}$ de la pierna durante la tarea de seguimiento.	56
6.16. Posición articular $\theta_8$ contra la posición deseada $\theta_{8d}$ de la pierna durante la tarea de seguimiento.	56
6.17. Posición articular $\theta_9$ contra la posición deseada $\theta_{9d}$ de la pierna durante la tarea de seguimiento.	56
6.18. Posición articular $\theta_{10}$ contra la posición deseada $\theta_{10d}$ de la pierna durante la tarea de seguimiento.	57
6.19. Posición articular $\theta_{11}$ contra la posición deseada $\theta_{11d}$ de la pierna durante la tarea de seguimiento.	57
6.20. Posición articular $\theta_{12}$ contra la posición deseada $\theta_{12d}$ de la pierna durante la tarea de seguimiento.	57
6.21. Error de posición articular durante la realización de la tarea de seguimiento $\tilde{\theta}_1$ .	58
6.22. Error de posición articular durante la realización de la tarea de seguimiento $\tilde{\theta}_2$ .	58
6.23. Error de posición articular durante la realización de la tarea de seguimiento $\tilde{\theta}_3$ .	58
6.24. Error de posición articular durante la realización de la tarea de seguimiento $\tilde{\theta}_4$ .	59
6.25. Error de posición articular durante la realización de la tarea de seguimiento $\tilde{\theta}_5$ .	59
6.26. Error de posición articular durante la realización de la tarea de seguimiento $\tilde{\theta}_6$ .	59
6.27. Error de posición articular durante la realización de la tarea de seguimiento $\tilde{\theta}_7$ .	60
6.28. Error de posición articular durante la realización de la tarea de seguimiento $\tilde{\theta}_8$ .	60
6.29. Error de posición articular durante la realización de la tarea de seguimiento $\tilde{\theta}_9$ .	60

6.30. Error de posición articular durante la realización de la tarea de seguimiento $\tilde{\theta}_{10}$ . . . . .	61
6.31. Error de posición articular durante la realización de la tarea de seguimiento $\tilde{\theta}_{11}$ . . . . .	61
6.32. Error de posición articular durante la realización de la tarea de seguimiento $\tilde{\theta}_{12}$ . . . . .	61
6.33. Par articular $\tau_1$ . . . . .	62
6.34. Par articular $\tau_2$ . . . . .	62
6.35. Par articular $\tau_3$ . . . . .	62
6.35. Par articular $\tau_4$ . . . . .	63
6.37. Par articular $\tau_5$ . . . . .	63
6.38. Par articular $\tau_6$ . . . . .	63
6.39. Par articular $\tau_7$ . . . . .	64
6.40. Par articular $\tau_8$ . . . . .	64
6.41. Par articular $\tau_9$ . . . . .	64
6.42. Par articular $\tau_{10}$ . . . . .	65
6.43. Par articular $\tau_{11}$ . . . . .	65
6.44. Par articular $\tau_{12}$ . . . . .	65
7.1. Posiciones iniciales del caminador . . . . .	67
7.2. Patrón de marcha para el caminado del robot. . . . .	68
7.3. Código en Python 1 . . . . .	69
7.4. Código en Python 2 . . . . .	69
7.5. Código en Python 3 . . . . .	69
7.6. Posiciones articulares de $\theta_1$ . . . . .	70
7.7. Posiciones articulares de $\theta_2$ . . . . .	70
7.8. Posiciones articulares de $\theta_3$ . . . . .	71
7.9. Posiciones articulares de $\theta_4$ . . . . .	71
7.10. Posiciones articulares de $\theta_5$ . . . . .	71
7.11. Posiciones articulares de $\theta_6$ . . . . .	72
7.12. Posiciones articulares de $\theta_7$ . . . . .	72
7.13. Posiciones articulares de $\theta_8$ . . . . .	72
7.14. Posiciones articulares de $\theta_9$ . . . . .	73
7.15. Posiciones articulares de $\theta_{10}$ . . . . .	73
7.16. Posiciones articulares de $\theta_{11}$ . . . . .	73
7.17. Posiciones articulares de $\theta_{12}$ . . . . .	74
7.18. Secuencia de configuraciones del robot NAO durante la ejecución de dos pasos de la marcha con el patrón propuesto . . . . .	74



# Índice de tablas

3.1. Parámetros Denavit-Hartenberg del las piernas del robot NAO	19
3.2. Parámetros Denavit-Hartenberg del brazo derecho del robot NAO	19
3.3. Parámetros Denavit-Hartenberg del brazo Izquierdo del robot NAO	20
3.4. Parámetros Denavit-Hartenberg de la cabeza del robot NAO	20
3.5. Valores de los parámetros geométricos de las piernas	20
3.6. Valores de los parámetros geométricos de los brazos	20
3.7. Valores de los parámetros geométricos de la cabeza	20
3.8. Posiciones de los centros de masa de cada eslabon respecto al marco $\sum_{cabe}$ unido a la articulación anterior más cercana	38
3.9. Valores de las masas de los eslabones del robot NAO	38
3.10. Valores de los tensores de inercias respecto al marco unido al centro de masa de su eslabón	38
6.1. Parámetros Del Patrón de marcha del robot NAO	49
7.1. Parámetros Del Patrón de marcha del robot NAO	67

# Capítulo 1

## Introducción

Durante varios años, la necesidad de construir máquinas capaces de realizar actividades monótonas, de gran precisión o de alto riesgo para los obreros, motivó al desarrollo de autómatas con objetivos particulares y características muy distintas a las de los seres humanos, tales como los manipuladores robóticos. Si bien su estudio aún persiste, el propio progreso en la robótica clásica, así como la inquietud por construir mecanismos semejantes a los organismos biológicos y los beneficios que estos implican, han motivado la investigación en robots biológicamente inspirados. Los investigadores, persuadidos por la idea de que en un futuro los robots convivirán con humanos y, por lo tanto, deberán ser capaces de desenvolverse en los mismos entornos, han mantenido gran interés por el desarrollo de robots humanoides. Gracias al impacto de humanoides como ASIMO y HRP-2, a la necesidad de contar con robots capaces de encargarse de tareas donde nuestra sociedad tiene deficiencias y a las aportaciones en el estudio y análisis de la locomoción bípeda humana, el desarrollo de los robots humanoides no ha cesado y ha motivado a la definición de nuevos conceptos y a la generación de nuevos métodos para solucionar problemas de diseño, estabilidad y control de robots humanoides, donde han emergido nuevos desafíos: algunos, incluso, imposibles de resolver usando métodos tradicionales, impulsando la adopción de enfoques multidisciplinarios, que abarcan tanto campos de la ingeniería que van desde la robótica, el control y la electrónica, hasta la inteligencia artificial y la computación gráfica como campos de otras áreas, entre ellas la biología, la psicología y las neurociencias [1]. Mientras las primeras investigaciones en la robótica bípeda y humanoide se enfocaban en la generación de ciclos de marcha que guardaran una estabilidad estática, en la actualidad los temas se han ampliado con el propósito de emplear a los humanoides como robots de servicio, que, en un futuro, transformarán la vida de las personas, realizarán una gran cantidad de tareas en beneficio de la humanidad y ayudarán a resolver problemas sociales y de salud, cuidarán de personas de edad avanzada, ayudarán y asistirán a discapacitados, vigilarán y patrullarán zonas públicas y, además, colaborarán en actividades de la vida cotidiana [1] y [2].

Se espera que los desafíos que implica el desarrollo de la robótica Humanoide estimulen el progreso tecnológico; la necesidad de los robots humanoides por contar con sistemas



## 1. Introducción

---

mecánicos lo suficientemente sofisticados como para desenvolverse en entornos humanos, incluso interactuar con ellos y emplear sus mismas herramientas, con algoritmos avanzados para la generación de movimientos versátiles y sincronizados, con sistemas inteligentes y esquemas de control robustos que actúen frente a perturbaciones y situaciones imprevistas, y con sistemas artificiales que simulen el comportamiento del cerebro humano. Esto producirá un importante avance científico y tecnológico, además de ampliar el conocimiento que tenemos de nosotros mismos [1] y [2].

La robótica humanoide, al igual que la industria automotriz y de las computadoras, vista como tal, tendrá en el futuro un efecto multiplicador, generando industrias proveedoras y consumidoras que impulsarán el desarrollo de los países que inviertan y se coloquen a la vanguardia de ellas [2]. El desarrollo de robots bípedos y humanoides tiene grandes beneficios y consecuencias: estimula la investigación científica e impulsa el progreso tecnológico, motiva la sinergia entre diversos campos de la ingeniería y de otras ciencias, propone alternativas a problemas sociales y de salud. Los robots humanoides, por la capacidad de adaptarse a entornos humanos así como de emplear sus mismas herramientas, por la capacidad de interacción y convivencia con otros robots y con los seres humanos, además de sus habilidades y destrezas motoras, no solo representan una área de investigación, sino una nueva industria que invierte en su desarrollo, promoviendo su comercialización, tanto de robots de servicio como de entretenimiento.

### 1.1. Antecedentes

En los últimos 30 años el estudio de la marcha de los robots bípedos ha sido de gran interés para numerosos investigadores, ya que representa un problema complejo. El objetivo principal de este tipo de robots es realizar caminatas estables de manera similar a la marcha humana. En el caso de ambientes con soportes discontinuos, tales como pendientes rocosas, peldaños y escalones en escaleras, es indiscutible pensar que la locomoción más apropiada y versátil es utilizando las piernas. Las piernas permiten evadir discontinuidades del suelo al pasar sobre ellas, además son una opción obvia para la locomoción en ambientes diseñados para los humanos. La motivación de estudiar los robots humanoides en particular surge de diversos intereses sociológicos y comerciales, que van desde el deseo de reemplazar humanos en ocupaciones peligrosas (remoción de minas, inspección en plantas de energía nuclear, intervenciones militares, etc.), hasta la restauración de movimiento en personas discapacitadas (prótesis inferiores controladas dinámicamente, rehabilitación robótica) [3].

### 1.2. Desarrollo de robots humanoides

En la actualidad existen múltiples prototipos de robots humanoides, con atributos diversos, que han sido presentados a la comunidad científica en foros especializados. El robot

## 1. Introducción

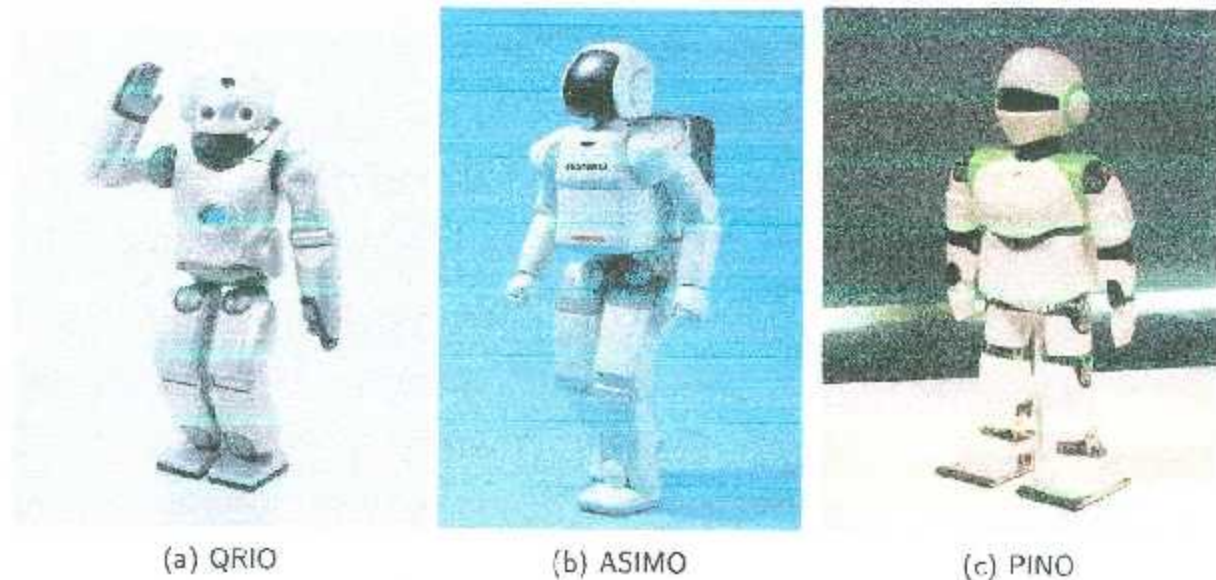


Figura 1.1: Robots bípedos

QRIO de Sony [4] fue diseñado para interactuar con gente en actividades de entretenimiento (ver Figura 1.1a); posee 48 grados de libertad (gdl), de los cuales 4 corresponden al cuello; mide 58 centímetros de altura y pesa aproximadamente 6.5 kg. El robot ASIMO, de la compañía Honda [5], es el robot con el más avanzado desempeño mecánico actualmente; es el primer robot capaz de subir escaleras y correr; tiene 26 gdl (ver Figura 1.1b), mide 120 cm de altura, y pesa aproximadamente 52 kg. El robot PINO [6] es un humanoide para investigación sobre aspectos de percepción y comportamiento, usando múltiples canales de percepción; su altura es equivalente a la de un niño de 1.5 años de edad (70 cm); posee 26 gdl y 4.5 kg (ver Figura 1.1c). El robot HRP-3 [7] es de 30 gdl, 1.54 m de estatura y pesa 58 Kg; es uno de los humanoides más utilizados actualmente en laboratorios de investigación (ver Figura 1.2a).

El robot iCub es un humanoide que se desarrolla en la Unión Europea con la participación de diversas universidades y centros de investigación [8]. Este robot pesa 23 kg, tiene una altura de aproximadamente 90 cm (características equivalentes a las de un niño de 2 años de edad), y cuenta con 53 grados de libertad en total (ver Figura 1.2b). El cuello tiene 3 gdl y los ojos también 3 gdl. Es el humanoide con más movilidad de sus partes que se ha diseñado hasta la fecha. Sus especificaciones se basan en datos anatómicos del ser humano [9], y sus atributos sobre percepción del entorno del robot son probablemente los más avanzados [10]. Un estudio sobre la cinemática de la cabeza y los ojos del robot iCub, fue publicado recientemente [11].

Los progresos que se han alcanzado en el desarrollo de humanoides, como los citados precedentemente, son considerables. En particular, la locomoción de humanoides bípedos ha sido estudiada ampliamente en la literatura científica. Los principales enfoques que han



## 1. Introducción

---

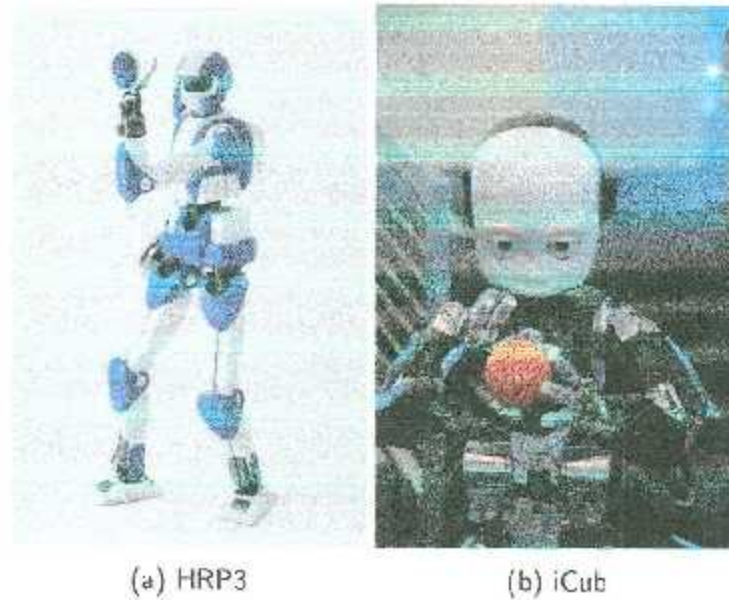


Figura 1.2: Robots Humanoides

siendo propuestos se pueden clasificar en tres grupos, dependiendo del control ejercido sobre las articulaciones del robot: locomoción activa, pasiva e híbrida (en trabajos previos se pueden encontrar aplicaciones típicas de estos enfoques [12], [13], [14] y [15]).

En múltiples trabajos sobre la marcha bípeda destaca la aplicación del criterio de estabilidad basado en el punto de momento cero, término acuñado en lengua inglesa como zero moment point ZMP [16]; se trata de un punto de la planta del pie de apoyo en el cual la suma de los momentos producidos por las fuerzas de inercia y la acción de la gravedad sobre los elementos físicos del cuerpo debe anularse con el momento producido por la fuerza reactiva del suelo sobre el pie durante la fase de simple apoyo. Para que un patrón de marcha de un humanoide sea factible, la ruta del ZMP debe permanecer dentro del polígono de sustentación del robot con uno o dos pies; sin embargo, el margen disponible para satisfacer dicha condición es relativamente pequeño desde el punto de vista geométrico [17], toda vez que las plantas de los pies deben ser de un tamaño moderado de tal manera que no se penalice su movilidad. Teniendo en cuenta lo anterior, las características del patrón de movimientos que se le especifique al robot, en términos del desplazamiento del torso y del pie libre, deben contribuir al logro de un nivel adecuado de estabilidad (considerada ésta como la capacidad del bípedo para soportar perturbaciones sin perder el equilibrio) durante la marcha [18], [19].

Algunos autores han tratado el tema de la generación de patrones de marcha de bípedos con base en las características del caminado humano [20] y [21]; otros han sintetizado trayectorias a partir de criterios tales como la estabilidad de la marcha [22], [23], [24], [25] y de la optimización del desempeño del robot, pero con limitaciones diversas. Ch. L. Shih

## 1. Introducción

---

y colegas [26] sintetizan las trayectorias articulares óptimas de un robot, minimizando el desplazamiento del ZMP, lo cual privilegia su estabilidad; sin embargo, las trayectorias generadas no garantizan una marcha sin impactos, lo cual, desafortunadamente, penaliza la estabilidad y el desempeño mecánico del robot. En otro estudio [27], se proponen funciones polinomiales de 6<sup>o</sup> orden para caracterizar los movimientos 3D de la cadera y del pie libre; en este caso sí se consideran restricciones para la prevención de impactos; no obstante, el número de parámetros requeridos por los polinomios es grande, lo cual complica la identificación de sus valores óptimos. Pámanes y Núñez [28] propusieron patrones de marcha de tipo cicloidal para un robot plano de 6 gdl, los cuales se aplicaron en otro trabajo [29] para el caminado en el plano sagital de un prototipo experimental, asegurando la prevención de impactos. En esas referencias no se consideraron movimientos espaciales de la pelvis y del pie libre del robot.

### 1.3. Locomoción dinámica pasiva

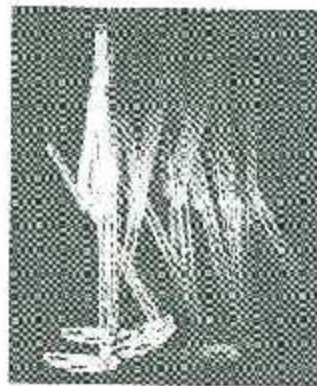


Figura 1.3: Mecanismo inventado por McGeer

El concepto de locomoción pasiva (“Passive Walking”), es decir, sin actuadores, fue introducido por primera vez en 1988 por Tad McGeer [30], [31], [32] y [33], quien estudió y construyó un caminador pasivo.

La virtud de este mecanismo es que no necesita energía externa ni ningún tipo de control para caminar por una pendiente, el movimiento viene propulsado por la energía gravitacional de la bajada. El sistema actúa como dos 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.



## 1. Introducción

---

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 (por definición, se comporta como un péndulo invertido inestable), manteniendo el ciclo límite estable.

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 de 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 conoce como caminata dinámica (*dynamie walking*).

### 1.4. Locomoción dinámica activa

Steve Collins y sus compañeros hicieron interesantes estudios de robots humanoides que usan los conceptos de la locomoción pasiva [34]. Desarrollaron tres robots diferentes, sustituyendo la propulsión gravitacional por actuadores simples.

El bípedo "CORNELL" (ver Figura 1.4a) está basado en el mecanismo pasivo, y propulsado por motores eléctricos y muelles que ayudan al movimiento de los tobillos cuando se levantan los pies. Tiene cinco grados de libertad (dos tobillos, dos rodillas y la cadera), los brazos están unidos mecánicamente a la pierna opuesta. El bípedo "DELFT" (ver Figura 1.4b) tiene una morfología similar, pero con actuación neumática. El último (ver Figura 1.4c) fue desarrollado en el MIT (Massachusetts Institute of Technology). Tiene la peculiaridad de estar controlado mediante un aprendizaje con refuerzo (*reinforcement learning*), que converge automáticamente a la estrategia de control óptima.

El bípedo "CORNELL" está específicamente diseñado para minimizar la energía de propulsión. Mediante una política adecuada en los actuadores se consigue que siempre aporten trabajo positivo y no se usen como disipadores. Este hecho está en contra posición con el problema que presentan muchos otros robots. En el momento en el que el pie impacta en el suelo se disipa energía debido a que el robot está frenando, afectando así al rendimiento, ya que luego hay que recuperarla. Aunque los robots "MIT" y "DELFT" no fueron diseñados específicamente para usar poca energía, ambos tienen las ventajas inherentes de los caminadores pasivos en lo que respecta a bajo consumo.



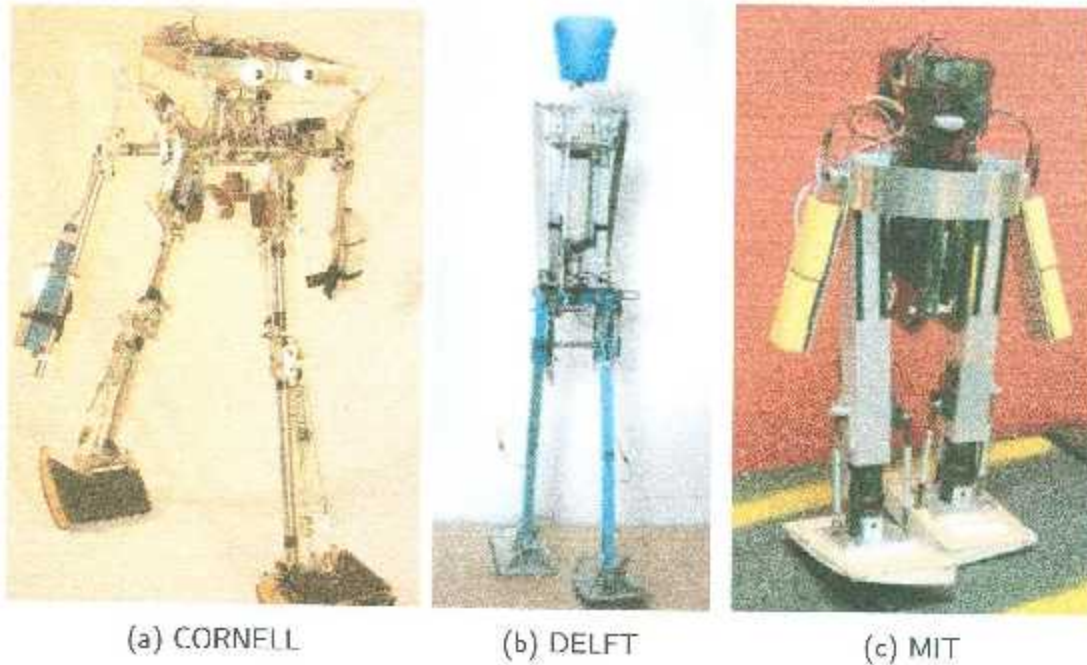


Figura 1.4: Robots bípedos

### 1.5. El método de dinámica cero híbrida

Una estrategia alternativa para controlar la locomoción bípeda concebida por Grizzle, Westervelt, Chevallereau, y otros en los 90's, es el método de dinámica cero híbrida [35], [36]. Esta técnica consiste en seleccionar un conjunto de funciones de las variables articulares. Este conjunto de configuraciones correspondientes a las funciones que son idénticamente cero definen una variedad, y las trayectorias que forman órbitas cerradas en la variedad cero corresponden a los ciclos de caminado para el bípedo. El control es entonces aplicado para hacer la variedad cero invariante y exponencialmente estabilizar las orbitas cerradas en la variedad. La propuesta de dinámica cero híbrida ha sido probada extensamente en el bípedo RABBIT [37] (ver Figura 1.5), y ha llamado mucho el interés de investigadores en la comunidad de control debido a las elegantes pruebas de estabilidad. Desafortunadamente, al igual que las propuestas de planificación y seguimiento de trayectorias, las estrategias de dinámica cero híbridas sufren de ineficiencia de energía.

### 1.6. Objetivo de la Tesis

En el presente tema de tesis se pretende abordar los puntos necesarios para la puesta en marcha del robot humanoide NAO, desde la explicación del funcionamiento del hardware y software proporcionado por el fabricante, el desarrollo del modelado cinemático y dinámico del robot, hasta la aplicación de algunos controladores propuestos en la literatura de

## 1. Introducción

---

robots bípedos.

Se pretende realizar simulaciones y pruebas experimentales que avalen la teoría previamente estudiada.

### 1.6.1. Objetivos específicos

Como objetivos específicos tenemos:

- Modelado cinemático directo de posición.
- Modelado cinemático inverso de posición.
- Modelado dinámico.
- Estudio sobre el control de caminado.



Figura 1.5: Robot RABBIT

### 1.7. Contenido de la tesis

En capítulo 2 se presenta la descripción del robot NAO, en este capítulo se explica de manera individual los componentes principales del robot NAO como lo son la visión, el audio, los sonares, el tipo de comunicación que utiliza para poder comunicarse con el usuario, como también el patrón de marcha que utiliza para poder caminar y finalmente se describen los parámetros geométricos del robot. En el capítulo 3 se obtienen los parámetros de Denavit-Hartenberg, para poder obtener los modelados cinemáticos directo e inverso de posición del robot NAO. El modelo cinemático inverso de posición, de los brazos y cabeza se obtienen aplicando el método de PAUL, mientras que el modelado inverso de posición de las piernas se obtiene aplicando una metodología propuesta en [43], finalmente se obtiene el modelo dinámico mediante las ecuaciones de Euler-Lagrange. En el capítulo 4 se propone un patrón de marcha basado en leyes de movimiento de tipo cicloidal, estos patrones están basados en la propuesta [47] y [48], también se definen los parámetros del patrón de marcha para realizar un caminado con el robot NAO. En el capítulo 5 se realiza un control de marcha mediante seguimiento de trayectorias utilizando los patrones de marcha, el modelado dinámico y las ecuaciones cinemáticas descritos en el capítulo 3 y 4, se hacen simulaciones en Simulink/MATLAB que permitan evaluar los pares aplicados a los motores durante un caminado del robot NAO. En el capítulo 6 se presentan las simulaciones del caminado del robot NAO realizadas en MATLAB, también se presentan las gráficas de las posiciones, velocidades y aceleraciones articulares durante todo el caminado del robot NAO, y finalmente se presentan las gráficas de los resultados obtenidos del control de seguimiento de trayectorias. En el capítulo 7 se hacen pruebas experimentales físicamente con el robot NAO, se obtienen las gráficas de las posiciones deseadas obtenidas con los parámetros del patrón de marcha, las cuales fueron graficadas en MATLAB para poder ser comparadas con las gráficas de las posiciones reales leídas por los encoders del robot con la finalidad de comprobar que el robot siguiera las posiciones articulares deseadas. En el capítulo 8 se mencionan las conclusiones finales de la tesis, y finalmente se presentan las bibliografías utilizadas en esta tesis y en la parte de anexos se proporciona el código realizado en MATLAB para hacer las simulaciones del patrón de marcha.



## Capítulo 2

# Descripción del robot humanoide NAO

### 2.1. Marcha del robot humanoide NAO

Un primer estudio de simulación de la marcha del robot humanoide NAO a partir de los patrones de marcha de tipo cicloidal fue presentado recientemente [52]. En las simulaciones efectuadas en esta referencia, a fin de simplificar el modelado de la cinemática inversa y teniendo en cuenta la arquitectura del NAO, se asumió que el movimiento de la pelvis es traslacional y que la recta normal a la superficie de la suela del pie oscilante, al momento de aterrizar en cada paso, se mantiene perpendicular a la superficie del piso. Los resultados mostraron que las consideraciones son correctas toda vez que se verifica la condición impuesta a las articulaciones de la pelvis por el hecho de que ambas son movidas por un solo motor. En [53] se desarrolló un algoritmo de control para conseguir que el robot humanoide NAO realizara un caminado, esto se llevo a cabo mediante un algoritmo que aprovecha las oscilaciones propias del sistema para generar la locomoción y así tener una dinámica de locomoción más eficiente el cual conlleva hacer estudios de frecuencia de oscilaciones del robot y atractores en el espacio. En la referencia [54] se ocupa de la generación de trayectorias para los centros de masa del robot NAO, mientras el pie sigue un patrón de marcha omnidireccional, las trayectorias son generadas siguiendo el criterio de estabilidad dinámica, asegurando analíticamente que el ZMP (Zero Moment Point) del robot se encuentre dentro del polígono de soporte. En [49] se centran en el modelado dinámico mediante las ecuaciones de Euler-Lagrange del robot humanoide NAO, además se realiza un análisis de la cinemática directa e inversa del robot, luego se proponen trayectorias para que el robot NAO realice un caminado y finalmente se analizan los pares aplicados a los motores mediante el modelado dinámico obtenido.

NAO es un robot humanoide programable y autónomo, desarrollado por Aldebaran Robotics, una compañía de robótica francesa con sede en París. El desarrollo comenzó con el lanzamiento del proyecto NAO en 2004. NAO sustituye al perro robot Aibo de Sony como plataforma estándar para la RoboCup ("Robot Soccer World Cup"). La Edición Académica fue desarrollada para las universidades y laboratorios con fines de investigación y educación. Dicha edición fue lanzada a las instituciones en 2008, y se puso a disposición

## 2. Descripción del robot humanoide NAO

---

antes del 2011. El modelo de robot, perteneciente al Laboratorio de Mecatronica y Control del Instituto Tecnológico de la Laguna, es el NAO H-25; cuenta con 25 grados de libertad (gdl): 5 en cada brazo, 1 en cada mano, 2 en el cuello, más 11 en las dos piernas, su altura es de 57 centímetros y tiene un peso de 5.2 kg, véase Figura 2.1.



Figura 2.1: Robot NAO

El robot NAO cuenta con una red de sensores que incluyen dos cámaras, cuatro micrófonos, sonares, dos emisores y receptores infrarrojos, una tarjeta de medición inercial, nueve sensores táctiles y 8 sensores de presión. Tiene distintos dispositivos de comunicación que incluye sintetizador de voz, luces led, y dos altavoces de alta fidelidad. La unidad central de procesamiento es una intel ATOM 1.6 GHz, localizado en la cabeza. Este procesador utiliza Linux Kernel y soporta el middleware propiedad de Aldebaran, NAOqi. NAO cuenta además con un segundo CPU localizado en el torso. La batería es de 27.6 watts-hora que provee a NAO con 1.5 horas de autonomía dependiendo del uso, véase Figura 2.2.

### 2.2. Movimiento

El caminado de NAO usa un modelo dinámico simple (péndulo lineal invertido) y programación cuadrática. Es estabilizado por la retroalimentación de los sensores de las articulaciones. Esto hace que el caminado sea robusto y resistente a pequeños disturbios, además de que las oscilaciones del torso en el plano frontal y lateral sean absorbidas. NAO puede caminar en diferentes superficies como alfombra, mosaico y madera. Además, puede resistir la transición entre estas superficies mientras camina. El módulo de movimiento de





## 2. Descripción del robot humanoide NAO

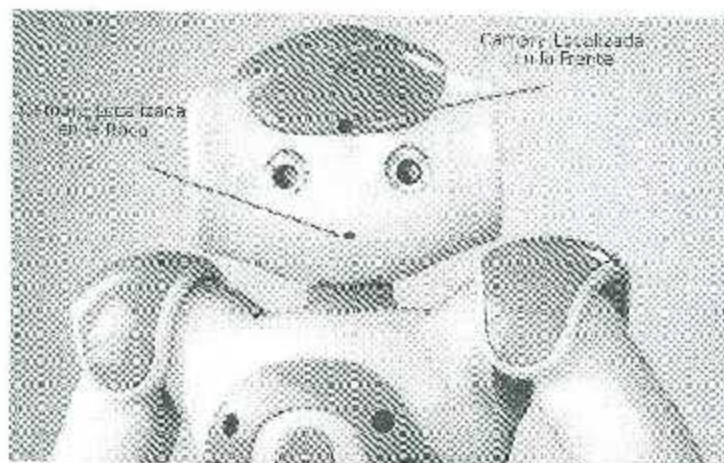


Figura 2.3: Cámaras del robot NAO

### 2.4. Audio

NAO usa cuatro micrófonos para rastrear sonidos y sus habilidades de reconocimiento de voz y conversación de texto a voz le permita comunicarse en 8 idiomas (ver Figura 2.4).

NAO realiza la localización de la fuente del sonido basado en un enfoque robusto conocido como "Diferencia de tiempo de llegada". Cuando una fuente cercana emite un sonido, cada uno de los cuatro micrófonos de NAO recibe la onda de sonido en diferentes tiempos. Estas diferencias, llamadas diferencia de tiempo inter-aural (ITD, por sus siglas en inglés), pueden ser matemáticamente procesadas para determinar la localización de la fuente emisora.

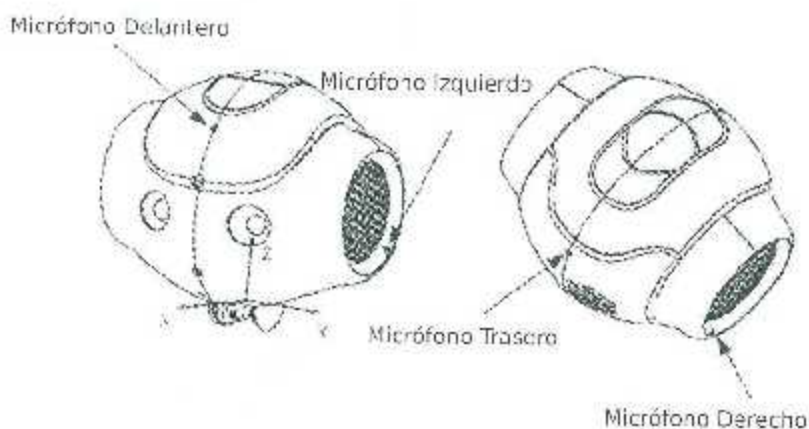


Figura 2.4: Micrófonos del robot NAO

### 2.5. Sonares

El NAO está equipado con dos canales de sonares, dos transmisores y dos receptores que permiten estimar distancias a obstáculos en el entorno. La distancia de detección es de 0-70 cm. A una distancia menor de 15 cm no hay información en la detección (ver Figura 2.5).

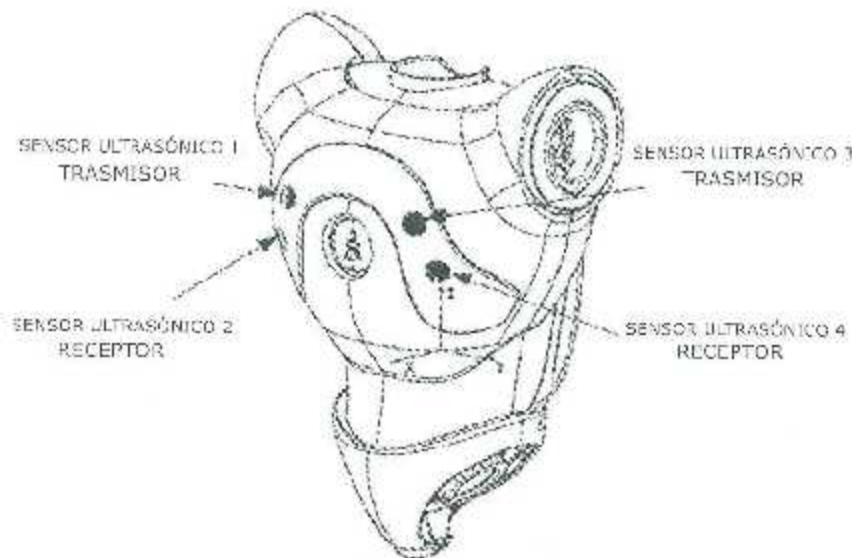


Figura 2.5: Sonares del robot NAO

### 2.6. Conectividad

Actualmente el NAO soporta Wi-Fi (a, b, g) y Ethernet. Además, cuenta con transceptores infrarrojos en los ojos que le permiten conectarse a otros objetos en el entorno. NAO es compatible con el estándar Wi-Fi IEEE 802.11g y puede ser usado con redes WPA y WEP, haciendo posible que se pueda conectar a la mayoría de las redes de casa y oficina. El sistema operativo de NAO soporta conexiones Ethernet y Wi-Fi, requiriendo para esta última sólo la contraseña de conexión. Usando infrarrojo, NAO puede comunicarse con otros NAO y dispositivos que soportan infrarrojos. Se puede configurar a NAO para que utilice infrarrojo para controlar otros dispositivos. Además, puede recibir instrucciones de emisores infrarrojos. NAO también puede detectar si la señal infrarroja proviene del lado derecho o izquierdo.

### 2.7. Parámetros geométricos del robot NAO

En las Figuras 2.6 y 2.7 se presentan los parámetros geométricos del robot NAO. Dichos parámetros se utilizarán para hacer los cálculos del modelado cinemático y dinámico



## 2. Descripción del robot humanoide NAO

del robot NAO así como también las simulaciones hechas en MATLAB.

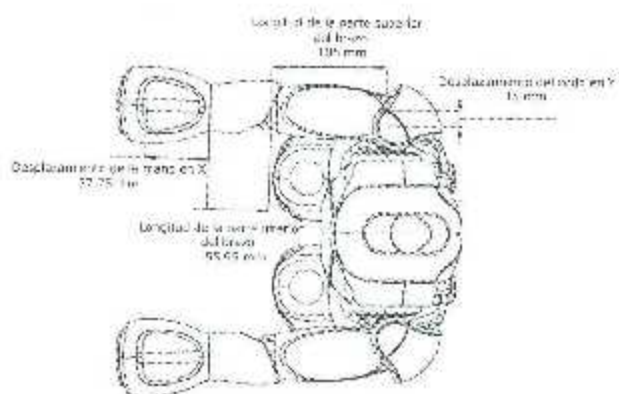


Figura 2.6: Parámetros geométricos de los brazos del robot NAO

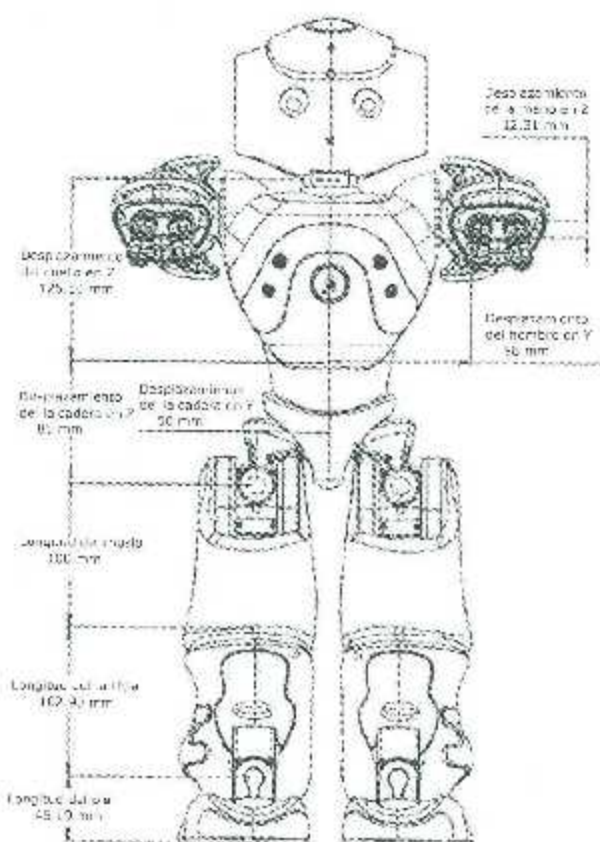


Figura 2.7: Parámetros geométricos del robot NAO

## Capítulo 3

# Modelado del robot NAO

Las actividades de investigación científica relacionadas con el desarrollo de robots humanoides han sido aceleradas a lo largo de las últimas décadas. Específicamente, la cinemática inversa de robots humanoides, que poseen generalmente 12 grados de libertad en piernas [38] y [39] se ha resuelto bajo diversos enfoques. En robots con piernas de menos grados de libertad, en los que la movilidad de la pelvis y del pie libre están limitadas, es posible resolver el problema inverso considerando las restricciones de manera explícita [40], [41]. En contraste con los casos de los robots considerados en las referencias precedentes, la cinemática inversa del robot NAO es particularmente compleja debido a su arquitectura. En efecto, el robot humanoide NAO, durante el caminado, no puede mover arbitrariamente la pelvis y el pie en desplazamiento, debido a que sólo posee 11 grados de libertad en sus piernas. El diseño mecánico del robot es tal que la transmisión de potencia hacia las dos articulaciones oblicuas de la pelvis (articulaciones 6 y 7 de la cadena cinemática del robot (ver Figura 3.1)) parte de un solo motor y, en consecuencia, obliga a éstas a mantener una relación constante entre sus correspondientes desplazamientos articulares. Esta restricción limita la movilidad de la pelvis y del pie libre, de tal manera que si se especifican libremente sus posiciones en un proceso de marcha, sus orientaciones estarán constreñidas y, eventualmente, se podría comprometer la factibilidad de la marcha bajo un criterio cinemático. En este capítulo se efectúa un estudio preliminar sobre la cinemática del robot NAO que prueba implícitamente que, si los desplazamientos de la pelvis y del pie libre permanecen con movimiento de traslación pura (es decir, con orientación constante), en una marcha sobre un plano horizontal, es posible generar una marcha factible especificando solamente los desplazamientos lineales de esas partes del robot. El procedimiento consiste, por una parte, en resolver la cinemática inversa del robot, considerándolo como si fuera de 12 grados de libertad y, por otra parte, en verificar que los movimientos articulares obtenidos cumplen la restricción impuesta a las articulaciones 6 y 7 por la transmisión de potencia. Los resultados obtenidos permiten planificar una marcha factible del robot a partir de un cierto número de parámetros del movimiento de la pelvis y del pie libre.



#### 3.1. Identificación de la cadena cinemática

Para la descripción de la cadena cinemática de las piernas, brazos y cabeza del robot NAO, a los eslabones de estos se les asignan marcos de referencia ortonormales aplicando la convención modificada de Denavit-Hartenberg (Khalil y Kleifinger, 1986) [42]. Bajo esta convención, dichos marcos se numeran de manera consecutiva. Para el caso de la cadena cinemática de las piernas del robot se numeran desde el eslabón fijo (pie de apoyo), hasta el 12, asignado al eslabón que corresponde al pie en desplazamiento (pie libre). En el caso de los brazos izquierdo y derecho del robot, el eslabón fijo sería el (hombro), hasta el 5, asignado al eslabón que corresponde a la mano; y por último para el caso de la cabeza del robot el eslabón fijo sería el (cuello) hasta el eslabón que corresponde a la cabeza véase Figura. 3.1.

#### 3.2. Modelo cinemático directo

El modelado directo consiste en el establecimiento de las funciones que hacen posible determinar las coordenadas operacionales de un robot a partir de sus coordenadas articulares. Puesto que las relaciones existentes entre las coordenadas operacionales y las variables articulares de un robot dependen de la estructura geométrica de éste, resulta de interés establecer un procedimiento normalizado para la descripción de dicha estructura, y de esa manera simplificar el tratamiento de los problemas del modelado directo e inverso. En esta sección nos referimos a la Figura 3.1 para presentar un procedimiento para tal propósito basado en los parámetros de Denavit-Hartenberg modificados. Los ejes del  $j$ -ésimo marco de referencia se asignan conforme a la siguiente convención:

- El eje  $z_j$  se define a lo largo del eje de la articulación, tipo R (rotacional), o el de deslizamiento de la corredera en el caso de una articulación tipo P (prismática). En este último caso la posición de  $z_j$  se fijará arbitrariamente.
- El eje  $x_j$  se define a lo largo de la perpendicular común a  $z_j$  y a  $z_{j-1}$ . Si estos dos ejes son paralelos,  $x_j$  no se puede definir de manera única; en tal caso se recomienda seguir un criterio de simetría o de simplicidad para definir la ubicación de este eje. Por supuesto, el punto de intersección de  $x_j$  y  $z_j$  define al origen  $o_j$  del marco.
- El eje  $y_j$  se define a partir de los ejes  $x_j$  y  $z_j$ , de tal manera que se complete un marco de mano derecha.

Una vez asignado un marco a cada eslabón, se definirán los parámetros geométricos que especifican la posición y orientación de cada marco respecto al precedente:

- $\alpha_j$  es el ángulo de  $z_{j-1}$  a  $z_j$ , medido respecto a  $x_{j-1}$  conforme a la regla de la mano derecha.
- $d_j$  es la distancia entre  $z_{j-1}$  y  $z_j$ , a lo largo de  $x_{j-1}$ .

### 3. Modelado del robot NAO

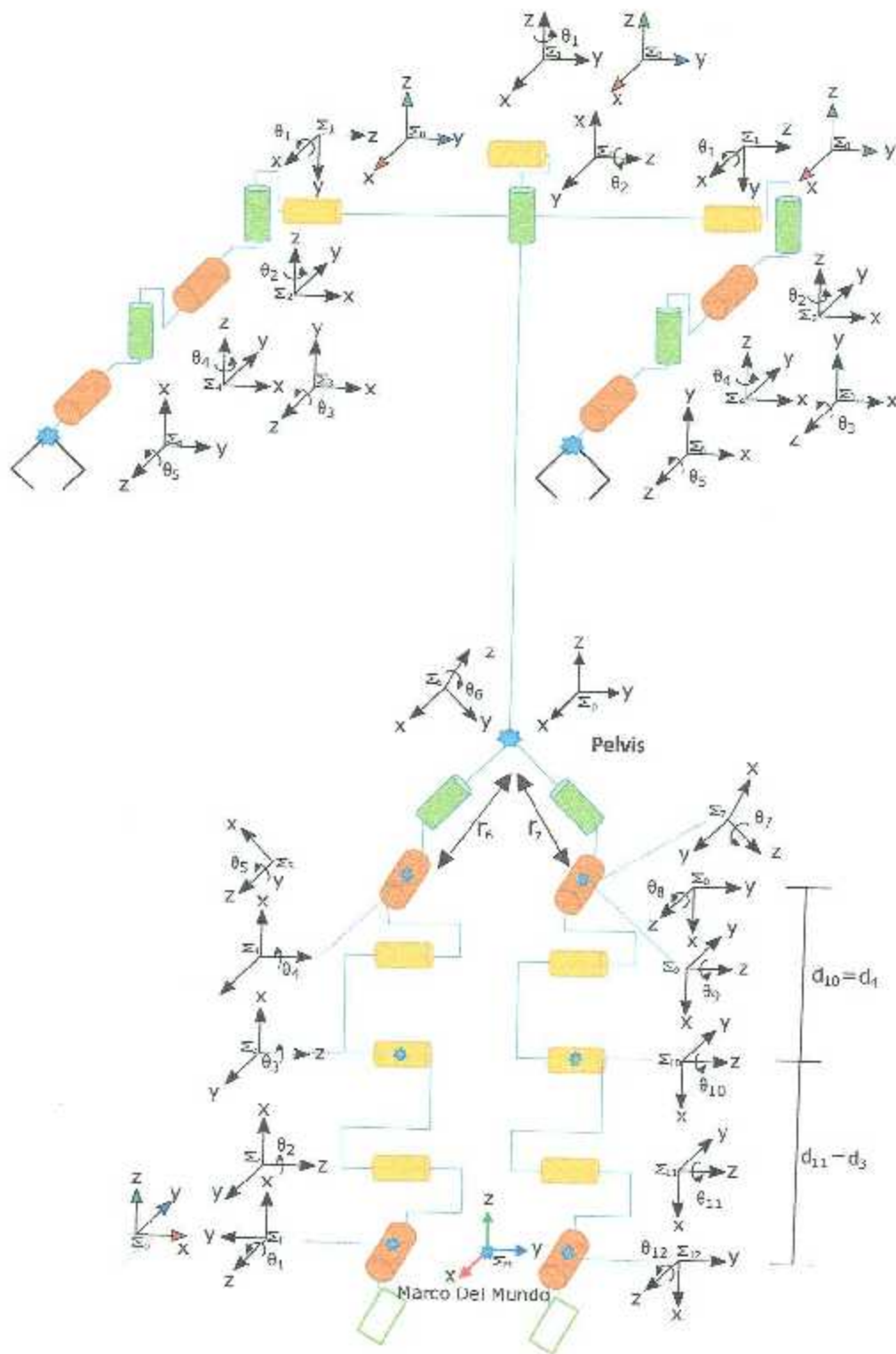


Figura 3.1: Marcos asignados a los eslabones del robot NAO

### 3. Modelado del robot NAO

- $\theta_j$  es el ángulo de  $x_{j-1}$  a  $x_j$ , medido respecto a  $x_j$  conforme a la regla de la mano derecha.
- $r_j$  es la distancia entre  $x_{j-1}$  y  $x_j$  a lo largo de  $x_j$ .

El modelo directo de posición del robot NAO permite determinar la posición y orientación de los brazos, piernas y cabeza del robot con respecto al marco del mundo, el cual se asume que se encuentra en medio de las dos piernas del robot NAO durante su primer paso. A continuación se obtienen los parámetros de Denavit-Hartenberg para la cadena cinemática de los brazos, piernas y cabeza del robot, de igual forma se especifican los valores de los parámetros geométricos del robot NAO.

Las tablas 3.1 a 3.4 muestran los parámetros de Denavit-Hartenberg de las piernas, brazos y cabeza del robot NAO. Las tablas 3.5 a 3.7 muestran sus valores geométricos.

Tabla 3.1: Parámetros Denavit-Hartenberg de las piernas del robot NAO

$i$	$\alpha_i$	$d_i$	$\theta_i$	$r_i$
1	90	0	$\theta_1$	0
2	90	0	$\theta_2$	0
3	0	d3	$\theta_3$	0
4	0	d4	$\theta_4$	0
5	-90	0	$\theta_5$	0
6	90	0	$\theta_6$	$r_6$
7	-90	0	$\theta_7$	$r_7$
8	90	0	$\theta_8$	0
9	-90	0	$\theta_9$	0
10	0	d10	$\theta_{10}$	0
11	0	d11	$\theta_{11}$	0
12	90	0	$\theta_{12}$	0

Tabla 3.2: Parámetros Denavit-Hartenberg del brazo derecho del robot NAO

$i$	$\alpha_i$	$d_i$	$\theta_i$	$r_i$
1	90	0	$\theta_1$	0
2	90	0	$\theta_2$	0
3	90	-d3l	$\theta_3$	r3l
4	-90	0	$\theta_4$	0
5	90	0	$\theta_5$	0

### 3. Modelado del robot NAO

Tabla 3.3: Parámetros Denavit-Hartenberg del brazo izquierdo del robot NAO

$i$	$\alpha_i$	$d_i$	$\theta_i$	$r_i$
1	-90	0	$\theta_1$	0
2	90	0	$\theta_2$	0
3	90	d3I	$\theta_3$	r3I
4	-90	0	$\theta_4$	0
5	90	0	$\theta_5$	0

Tabla 3.4: Parámetros Denavit-Hartenberg de la cabeza del robot NAO

$i$	$\alpha_i$	$d_i$	$\theta_i$	$r_i$
1	0	0	$\theta_1$	0
2	-90	0	$\theta_2$	0

Tabla 3.5: Valores de los parámetros geométricos de las piernas

Dimensiones	d3	d4	d10	d11	r6	r7
mm	102.90	100	102.90	100	70.7106	70.7106

Tabla 3.6: Valores de los parámetros geométricos de los brazos

Dimensiones	d3I	r3I
mm	15	105

Tabla 3.7: Valores de los parámetros geométricos de la cabeza

Dimensiones	L1	L2	L3
mm	53.9	67.9	161.5

Habiendo definido los parámetros de Denavit-Hartenberg, se pueden conformar las matrices de transformación elementales del robot, las cuales describen la posición y orientación del marco  $\sum_j$  con respecto al  $\sum_{j-1}$ . Dicha matriz está dada por:

$$T_j^{j-1} = \begin{bmatrix} c_{\theta_j} & -s_{\theta_j} & 0 & d_j \\ c_{\alpha_j} s_{\theta_j} & c_{\alpha_j} c_{\theta_j} & -s_{\alpha_j} & r_j s_{\alpha_j} \\ s_{\alpha_j} s_{\theta_j} & s_{\alpha_j} c_{\theta_j} & c_{\alpha_j} & r_j c_{\alpha_j} \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad (3.1)$$

donde  $c$  representa la función coseno y  $s$  la función seno, es decir  $c_{\theta_j} = \cos(\theta_j)$ ,  $s_{\theta_j} = \sin(\theta_j)$ ,  $c_{\alpha_j} = \cos(\alpha_j)$ ,  $s_{\alpha_j} = \sin(\alpha_j)$ .

A continuación se muestran las matrices de transformación elementales que describen la posición y orientación de la cabeza con respecto al marco fijo (cuello):

$$T_1^0 = \begin{bmatrix} c_{\theta_1} & -s_{\theta_1} & 0 & 0 \\ s_{\theta_1} & c_{\theta_1} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$



### 3. Modelado del robot NAO

---

$$T_2^1 = \begin{bmatrix} c\theta_2 & -s\theta_2 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ s\theta_2 & c\theta_2 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Las matrices de transformación elementales que describen la posición y orientación del brazo derecho con respecto al marco fijo (hombro derecho) son:

$$T_1^0 = \begin{bmatrix} c\theta_1 & -s\theta_1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -s\theta_1 & -c\theta_1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$T_2^1 = \begin{bmatrix} -s\theta_2 & -c\theta_2 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ c\theta_2 & -s\theta_2 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$T_3^2 = \begin{bmatrix} c\theta_3 & -s\theta_3 & 0 & -d_3I \\ 0 & 0 & -1 & -r_3I \\ s\theta_3 & c\theta_3 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$T_4^3 = \begin{bmatrix} c\theta_4 & -s\theta_4 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -s\theta_4 & -c\theta_4 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$T_5^4 = \begin{bmatrix} c\theta_5 & -s\theta_5 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ s\theta_5 & c\theta_5 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Las matrices de transformación elementales que describen la posición y orientación del brazo izquierdo con respecto al marco fijo (hombro izquierdo) son:

$$T_1^0 = \begin{bmatrix} c\theta_1 & -s\theta_1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -s\theta_1 & -c\theta_1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$T_2^1 = \begin{bmatrix} -s\theta_2 & -c\theta_2 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ c\theta_2 & -s\theta_2 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$T_3^2 = \begin{bmatrix} c\theta_3 & -s\theta_3 & 0 & d_3I \\ 0 & 0 & -1 & -r_3I \\ s\theta_3 & c\theta_3 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$T_4^3 = \begin{bmatrix} c\theta_4 & -s\theta_4 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -s\theta_4 & -c\theta_4 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

### 3. Modelado del robot NAO

$$T_5^4 = \begin{bmatrix} c\theta_5 & -s\theta_5 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ s\theta_5 & c\theta_5 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Para el caso de la cadena cinemática de las piernas del robot, es necesario determinar la posición y orientación de la pelvis y del pie libre con respecto al marco del mundo, el cual se asume que se encuentra en medio de las dos piernas.

$$T_1^0 = \begin{bmatrix} c\theta_1 & -s\theta_1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ s\theta_1 & c\theta_1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$T_2^1 = \begin{bmatrix} c\theta_2 & -s\theta_2 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ s\theta_2 & c\theta_2 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$T_3^2 = \begin{bmatrix} c\theta_3 & -s\theta_3 & 0 & d3 \\ s\theta_3 & c\theta_3 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$T_4^3 = \begin{bmatrix} c\theta_4 & -s\theta_4 & 0 & d4 \\ s\theta_4 & c\theta_4 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$T_5^4 = \begin{bmatrix} c\theta_5 & -s\theta_5 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -s\theta_5 & -c\theta_5 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$T_6^5 = \begin{bmatrix} c\theta_6 & -s\theta_6 & 0 & 0 \\ 0 & 0 & -1 & r_6 \\ s\theta_6 & c\theta_6 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$T_7^6 = \begin{bmatrix} c\theta_7 & -s\theta_7 & 0 & 0 \\ 0 & 0 & 1 & r_7 \\ -s\theta_7 & -c\theta_7 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$T_8^7 = \begin{bmatrix} c\theta_8 & -s\theta_8 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ s\theta_8 & c\theta_8 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$T_9^8 = \begin{bmatrix} c\theta_9 & -s\theta_9 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -s\theta_9 & -c\theta_9 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$T_{10}^9 = \begin{bmatrix} c\theta_{10} & -s\theta_{10} & 0 & d10 \\ s\theta_{10} & c\theta_{10} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

### 3. Modelado del robot NAO

$$T_{11}^{10} = \begin{bmatrix} c\theta_{11} & -s\theta_{11} & 0 & d_{11} \\ s\theta_{11} & c\theta_{11} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$T_{12}^{11} = \begin{bmatrix} c\theta_{12} & -s\theta_{12} & 0 & 0 \\ 0 & 0 & -1 & 0 \\ s\theta_{12} & c\theta_{12} & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Ya obtenidas las matrices elementales de cada miembro del robot NAO, se procede a obtener los elementos de la matriz homogénea que se utiliza para especificar la posición y orientación deseada de cada miembro del robot, con respecto al marco del mundo  $\sum_{12}$ , el cual se encuentra colocado en medio de las dos piernas.

En el caso de la cabeza del robot NAO, los elementos de la matriz homogénea  $T_2^0$  describen la posición y orientación de la cabeza (marco de referencia  $\sum_2$ ), con respecto al cuello (marco de referencia  $\sum_0$ ).

$$T_2^0 = T_1^0 T_2^1 \quad (3.2)$$

En el caso del brazo derecho e izquierdo del robot NAO, los elementos de la matriz homogénea  $T_5^0$  describen la posición y orientación de la mano (marco de referencia  $\sum_5$ ) con respecto al hombro (marco de referencia  $\sum_0$ ).

$$T_5^0 = T_1^0 T_2^1 T_3^2 T_4^3 T_5^4 \quad (3.3)$$

En el caso de la cadena cinemática de las piernas del robot NAO, hay que determinar la posición y orientación de la pelvis y del pie libre con respecto al marco del mundo, el cual se asume que se encuentra en medio de las dos piernas.

Los elementos de la matriz  $T_6^0$  describe la posición y orientación de la cadera (marco de referencia  $\sum_P$ ) con respecto al pie fijo (marco de referencia  $\sum_0$ ).

$$T_6^0 = T_1^0 T_2^1 T_3^2 T_4^3 T_5^4 T_6^5 T_P^6 \quad (3.4)$$

Por último, los elementos de la matriz homogénea  $T_{12}^0$ , que definen la posición y orientación del pie móvil (marco de referencia  $\sum_{12}$ ), con respecto al pie fijo, (marco de referencia  $\sum_0$ ).

$$T_{12}^0 = T_1^0 T_2^1 T_3^2 T_4^3 T_5^4 T_6^5 T_7^6 T_8^7 T_9^8 T_{10}^9 T_{11}^{10} T_{12}^{11} \quad (3.5)$$

Una vez obtenidas las matrices elementales del modelo directo, se procede a realizar el diagrama alambre en MATLAB (ver Anexo A), dicho diagrama nos ayudará para poder hacer simulaciones tanto para el modelado cinemático directo e inverso del robot así como también su modelado dinámico. En La Figura 3.2 se puede observar el diagrama alambre del robot NAO.



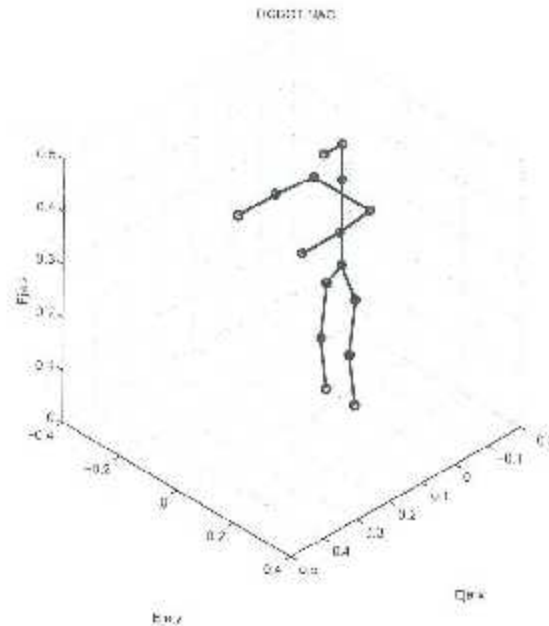


Figura 3.2: Diagrama alambre del robot NAO

### 3.3. Modelado cinemático inverso de posición

En el modelado inverso, se deducen las expresiones para calcular las coordenadas articulares que configuran al robot, de tal manera que se coloque a su órgano terminal en una situación especificada previamente por el usuario.

#### 3.3.1. Modelo cinemático inverso de posición de los brazos izquierdo y derecho

La obtención del modelado inverso de posición de los brazos izquierdo y derecho se realizó mediante el método de Paul, basado en el uso de las matrices elementales, las cuales permiten obtener sucesivamente las variables articulares a partir de la siguiente ecuación.

$$U_0 = T_n^0, \quad (3.6)$$

Recordando que la matriz  $T_n^0$  resulta del producto de las matrices elementales, se puede escribir como:

$$U_0 = T_1^0 T_2^1 \dots T_{n-1}^{n-2} T_n^{n-1}, \quad (3.7)$$

donde los elementos de la matriz conocida  $U_0$  se definirán de la siguiente manera:

$$U_0 = \begin{bmatrix} s_x & n_x & a_x & p_x \\ s_y & n_y & a_y & p_y \\ s_z & n_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.8)$$

Se observa que de la ecuación (3.7) resultan 12 ecuaciones escalares no triviales, conteniendo como incógnitas las variables articulares del manipulador. Estas ecuaciones son altamente no lineales, lo cual complica significativamente su resolución. El enfoque de Paul permite simplificar el aislamiento de cada variable articular mediante el principio que se ilustra para  $\theta_1$  como sigue:

Pre multiplique ambos miembros de la ecuación (3.7) por  $T_0^1$ . Así se obtiene

$$T_0^1 U_0 = T_2^1 \dots T_{n-1}^{n-2} T_n^{n-1}. \quad (3.9)$$

En esta ecuación se observa que el producto del lado derecho queda en función de las incógnitas  $\theta_1, \theta_2, \dots, \theta_5$  mientras que el lado izquierdo contiene solamente a  $\theta_1$ , quedando así aislada esta incógnita. Entonces, de las 12 ecuaciones escalares que resultan, se busca una que facilite la determinación de  $\theta_1$ . Aplicando de manera reiterada el procedimiento anterior, es posible obtener todas las variables articulares.

Para el brazo derecho:

Obtención de  $\theta_1$

$$\theta_1 = \text{atan2}(-E_1 p_x, E_1 p_x)$$

donde

$$E_1 = 1$$

Obtención de  $\theta_2$

$$\theta_2 = \text{atan2}(A_2, B_2)$$

donde

$$\begin{aligned}
 E_2 &= 1 \\
 a_2 &= r_3 \\
 b_2 &= -d_3 \\
 c_2 &= \tilde{n}_3 c_{\theta_1} - \tilde{p}_3 s_{\theta_1} \\
 A_2 &= b_2 c_2 + E_2 a_2 \sqrt{a_2^2 - b_2^2 - c_2^2} \\
 B_2 &= a_2 c_2 - E_2 b_2 \sqrt{a_2^2 + b_2^2 - c_2^2}
 \end{aligned}$$

Obtención de  $\theta_3$

$$\theta_3 = \text{atan2}(A_2, B_2)$$

donde

$$\begin{aligned}
 A_3 &= a_2 c_{\theta_1} - a_2 s_{\theta_1} \\
 B_3 &= a_2 c_{\theta_2} - \tilde{a}_x c_{\theta_1} s_{\theta_2} + \tilde{a}_z s_{\theta_1} s_{\theta_2}
 \end{aligned}$$

Obtención de  $\theta_4$

$$\theta_4 = \text{atan2}(A_4, B_4)$$

donde

$$\begin{aligned}
 A_4 &= a_y c_{\theta_2} c_{\theta_3} + a_z (c_{\theta_3} s_{\theta_1} s_{\theta_2} + c_{\theta_1} s_{\theta_3}) + a_x (-c_{\theta_1} c_{\theta_3} s_{\theta_2}) + s_{\theta_1} s_{\theta_3} \\
 B_4 &= a_x c_{\theta_1} c_{\theta_2} - a_z c_{\theta_2} s_{\theta_1} + a_y s_{\theta_2}
 \end{aligned}$$

Obtención de  $\theta_5$

$$\theta_5 = \text{atan2}(A_5, B_5)$$

donde

$$\begin{aligned}
 A_5 &= -(s_y c_{\theta_2} s_{\theta_3}) + s_x (c_{\theta_1} s_{\theta_2} + c_{\theta_1} s_{\theta_2} s_{\theta_3}) + s_z (c_{\theta_1} c_{\theta_3} - s_{\theta_1} s_{\theta_2} s_{\theta_3}) \\
 B_5 &= -(n_y c_{\theta_2} s_{\theta_3}) + n_x (c_{\theta_1} s_{\theta_2} + c_{\theta_1} s_{\theta_2} s_{\theta_3}) + n_z (c_{\theta_1} c_{\theta_3} - s_{\theta_1} s_{\theta_2} s_{\theta_3})
 \end{aligned}$$



### 3. Modelado del robot NAO

---

En el caso del brazo izquierdo, se obtuvieron los siguientes resultados:

Obtención de  $\theta_1$

$$\theta_1 = \text{atan2}(-E_1 p_x, F_1 p_x)$$

donde

$$E_1 = 1$$

Obtención de  $\theta_2$

$$\theta_2 = \text{atan2}(A_2, B_2)$$

donde

$$E_2 = 1$$

$$a_2 = r_3$$

$$b_2 = -d_3$$

$$c_2 = p_x c \theta_1 - p_y s \theta_1$$

$$A_2 = b_2 c_2 + E_2 a_2 \sqrt{a_2^2 - b_2^2 - c_2^2}$$

$$B_2 = a_2 c_2 - E_2 b_2 \sqrt{a_2^2 - b_2^2 - c_2^2}$$

Obtención de  $\theta_3$

$$\theta_3 = \text{atan2}(A_3, B_3)$$

dónde

$$A_3 = a_2 c \theta_1 - a_x s \theta_1$$

$$B_3 = c_y c \theta_2 - a_x c_2 s \theta_2 + a_y s \theta_1 s \theta_2$$

Obtención de  $\theta_4$

$$\theta_4 = \text{atan2}(A_4, B_4)$$

donde

$$\begin{aligned} A_4 &= a_y c_{\theta_2} c_{\theta_1} - a_z (c_{\theta_1} s_{\theta_2} s_{\theta_3} + c_{\theta_1} s_{\theta_3}) + a_x (-c_{\theta_1} c_{\theta_2} s_{\theta_3} - s_{\theta_1} s_{\theta_3}) \\ B_4 &= a_x c_{\theta_1} c_{\theta_2} - a_z c_{\theta_2} s_{\theta_1} - a_y s_{\theta_2} \end{aligned}$$

Obtención de  $\theta_5$ .

$$\theta_5 = \text{atan2}(A_5, B_5)$$

donde

$$\begin{aligned} A_5 &= -(s_y c_{\theta_2} s_{\theta_3}) + s_x (c_{\theta_2} s_{\theta_1} + c_{\theta_2} s_{\theta_2} s_{\theta_3}) + s_z (c_{\theta_1} c_{\theta_3} - s_{\theta_1} s_{\theta_2} s_{\theta_3}) \\ B_5 &= -(n_y c_{\theta_2} s_{\theta_3}) + n_x (c_{\theta_2} s_{\theta_1} + c_{\theta_2} s_{\theta_2} s_{\theta_3}) - n_z (c_{\theta_1} c_{\theta_3} - s_{\theta_1} s_{\theta_2} s_{\theta_3}) \end{aligned}$$

#### 3.3.2. Modelado cinemático inverso de posición de las piernas

A partir de las matrices elementales de las piernas del robot, se resuelve el modelo inverso de posición de éstas, con el cual se calculan los valores de las variables articulares que permiten obtener la posición y la orientación deseada tanto de la pelvis, como del pie oscilante del robot. La derivación de las ecuaciones del mencionado modelo se basó en el enfoque propuesto en [13], que a su vez, se inspira en el principio del que parte el método de Paul [14].

La matriz de transformación homogénea que se utiliza para especificar la posición y orientación deseada de la pelvis del robot, con respecto al marco del mundo  $\sum_{M}$ , está dada por:

$$T_P^M = \begin{bmatrix} t_{11} & t_{12} & t_{13} & x_p \\ t_{21} & t_{22} & t_{23} & y_p \\ t_{31} & t_{32} & t_{33} & z_p \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

A partir de esta matriz, y de las matrices elementales de las piernas del robot, se obtienen las siguientes ecuaciones del modelo inverso de posición de la pierna de apoyo del robot:

### 3. Modelado del robot NAO

$$\begin{aligned}
 a_6 &= c_{\theta_1} l_{21} - s_{\theta_1} t_{31} \\
 b_6 &= -c_p c_{\theta_1} t_{32} - c_p s_{\theta_1} l_{32} - s_p c_{\theta_1} t_{23} - s_p s_{\theta_1} t_{33} \\
 E_6 &= 1 \\
 \theta_6 &= \text{atan2}(E_6 a_6, -E_6 b_6) \\
 a_1 &= c_p r_6 t_{23} - y_p - d_7 - r_8 s_p t_{22} \\
 b_1 &= c_p r_6 t_{33} - z_p - r_8 s_p t_{32} \\
 E_1 &= 1 \\
 \theta_1 &= \text{atan2}(E_1 a_1, -E_1 b_1) \\
 s_5 &= s_6 (c_{\theta_1} l_{21} + s_{\theta_1} t_{31}) + c_{\theta_6} c_p (c_{\theta_1} t_{22} + s_{\theta_1} t_{32}) \\
 &\quad - c_{\theta_6} s_p (c_{\theta_1} t_{23} + s_{\theta_1} t_{33}) \\
 c_5 &= c_p (c_{\theta_1} t_{23} + s_{\theta_1} t_{33}) - s_p (c_{\theta_1} t_{22} + s_{\theta_1} t_{32}) \\
 \theta_5 &= \text{atan2}(s_5, c_5) - \frac{\pi}{4} \\
 x_{14} &= x_{ped} c_{\theta_1} - d_7 s_{\theta_1} - d_1 c_{\theta_1} - y_{ped} s_{\theta_1} - d_{77} (t_{33} c_{\theta_1} - t_{23} s_{\theta_1}) \\
 z_{14} &= x_{ped} - t_{13} d_{77} \\
 r_{14} &= \sqrt{x_{14}^2 + z_{14}^2} \\
 \tilde{x}_{14}/u &= \frac{z_{14}}{r_{14}} \\
 x_A/u &= \frac{z_{14}}{r_{14}} \\
 r_{st} &= d_3 \\
 r_{sf} &= d_4 \\
 g &= \frac{r_{st}^2 - r_{sf}^2 - r_{14}^2}{2r_{14}} \\
 h &= \sqrt{r_{st}^2 - g^2} \\
 E_2 &= 1 \\
 r_{str} &= E_2 h x_{14}/u + g z_{14}/u \\
 r_{st\theta} &= -E_2 h z_{14}/u - g x_{14}/u \\
 r_{sfr} &= z_{14} - r_{str} \\
 r_{sf\theta} &= x_{14} - r_{st\theta} \\
 \theta_{st} &= \text{atan2}(r_{st\theta}, r_{str}) \\
 \theta_{sf} &= \text{atan2}(r_{sf\theta}, r_{sfr}) \\
 \theta_2 &= \theta_{st} - \frac{\pi}{2} \\
 \theta_3 &= \theta_{sf} - \theta_{st} \\
 s_{234} &= -t_{13} (c_p s_{\theta_1} - c_{\theta_6} c_{\theta_1} s_p) + t_{12} (s_{\theta_1} s_p + c_{\theta_6} c_{\theta_1} c_p) + c_{\theta_6} s_{\theta_1} t_{11} \\
 c_{234} &= -c_p s_{\theta_1} t_{12} + c_{\theta_6} t_{11} - s_{\theta_6} s_p t_{13} \\
 \theta_{234} &= \text{atan2}(s_{234}, c_{234}) \\
 \theta_4 &= \theta_{234} - \theta_2 - \theta_3
 \end{aligned}$$



### 3. Modelado del robot NAO

La matriz de transformación homogénea que se utiliza para especificar la posición y orientación deseadas del pie oscilante del robot, con respecto al marco de la pelvis  $\sum_P$ , se calcula mediante:

$$T_{12}^P = T_{M}^P T_{12}^M, \quad (3.10)$$

donde  $T_{M}^P$  es la matriz conocida que determina la posición y la orientación deseada del marco del mundo con respecto a la pelvis, mientras que  $T_{12}^M$  especifica la posición y orientación deseada del pie oscilante con respecto al marco del mundo  $\sum_M$ . Los elementos de la matriz  $T_{12}^P$ , calculada mediante la ecuación (3.10), se designan de la siguiente manera:

$$T_P^M = \begin{bmatrix} s_{fpy} & n_{fpy} & a_{fpr} & p_{fpx} \\ s_{fpy} & n_{fpy} & a_{fpy} & p_{fpy} \\ s_{fpz} & n_{fpz} & a_{fpz} & p_{fpz} \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Siguiendo la misma metodología utilizada para la pierna de apoyo, se obtienen las siguientes ecuaciones del modelo inverso de posición de la pierna oscilante:

$$\begin{aligned} a_7 &= a_{fpy} p_{fpx} s_p^2 - c_{fpx} p_{fpy} s_p^2 + a_{fpx} c_p r_7 - a_{fpy} c_p r_7 s_p \\ &\quad - a_{fpy} c_p^2 p_{fpx} - a_{fpx} c_p^2 p_{fpy} \\ b_7 &= a_{fpx} r_7 - a_{fpx} c_p p_{fpy} + a_{fpy} c_p p_{fpx} - a_{fpx} p_{fpy} s_p \\ &\quad + a_{fpy} p_{fpx} s_p \\ E_7 &= 1 \\ \theta_7 &= \text{atan2}(E_7 a_7, -E_7 b_7) \\ a_8 &= r_7 - p_{fpy} c_p - p_{fpx} s_p \\ b_8 &= -p_{fpy} c_p r_7 s_p + p_{fpx} c_p c_p + p_{fpx} s_p r_7 \\ E_8 &= 1 \\ \theta_8 &= \text{atan2}(E_8 a_8, -E_8 b_8) - \frac{3\pi}{4} \\ r_{812_x} &= p_{fpy} s_{\theta_8} - d_{77} s_{\theta_8} + p_{fpx} c_{\theta_7} c_{\theta_8} + p_{fpy} s_{\theta_7} c_{\theta_8} \\ r_{812_z} &= p_{fpx} c_{\theta_7} - p_{fpy} s_{\theta_7} \\ r_{812} &= \sqrt{r_{812_x}^2 + r_{812_z}^2} \\ r_{812/w_7} &= \frac{r_{812_x}}{r_{812}} \\ r_{812/w_8} &= \frac{r_{812_z}}{r_{812}} \\ r_{wf} &= d_{10} \\ r_{pt} &= d_{11} \end{aligned}$$

$$\begin{aligned}
 g_w &= \frac{r_{wf}^2 - r_{wt}^2 - r_{s12}^2}{2r_{s12}} \\
 h_w &= \sqrt{r_{wf}^2 - g_w^2} \\
 E_g &= 1 \\
 r_{wfx} &= E_g h_w r_{s12}/u_x + g_w r_{s12}/u_x \\
 r_{wfy} &= -E_g h_w r_{s12}/u_y - g_w r_{s12}/u_y \\
 r_{wtz} &= r_{s12} - r_{wfx} \\
 r_{wty} &= r_{s12} - r_{wfy} \\
 \theta_{wf} &= \text{atan2}(r_{wfy}, r_{wfx}) \\
 \theta_{wt} &= \text{atan2}(r_{wty}, r_{wtz}) \\
 \theta_2 &= \theta_{wf} - \frac{\pi}{2} \\
 \theta_{10} &= \theta_{wt} - \theta_{wf} \\
 s_{12} &= s_{\theta_7} s_{\theta_8} s_{jyz} - s_{jyz} (c_{\theta_8} s_{\theta_7} - c_{\theta_7} c_{\theta_8} s_{\theta_9}) \\
 &\quad - s_{jyz} (c_{\theta_8} c_{\theta_7} + c_{\theta_7} s_{\theta_8} s_{\theta_9}) \\
 c_{12} &= n_{jyz} s_{\theta_7} s_{\theta_8} - n_{jyz} (c_{\theta_8} s_{\theta_7} - c_{\theta_7} c_{\theta_8} s_{\theta_9}) \\
 \theta_{12} &= \text{atan2}(s_{12}, c_{12}) \\
 a_{11} &= c_{\theta_{10}} \\
 b_{11} &= -s_{\theta_{10}} \\
 f_{11} &= a_{jyz} c_{\theta_7} - a_{jyz} c_{\theta_7} s_{\theta_9} + a_{jyz} s_{\theta_7} s_{\theta_9} \\
 E_{11} &= 1 \\
 s_{11} &= \frac{b_{11} f_{11} + E_{11} a_{11} \sqrt{a_{11}^2 - b_{11}^2 - f_{11}^2}}{a_{11}^2 + b_{11}^2} \\
 c_{11} &= \frac{a_{11} f_{11} - E_{11} b_{11} \sqrt{a_{11}^2 + b_{11}^2 - c_{11}^2}}{a_{11}^2 + b_{11}^2} \\
 &\quad - n_{jyz} (c_{\theta_8} c_{\theta_7} + c_{\theta_7} s_{\theta_8} s_{\theta_9}) \\
 \theta_{11} &= \text{atan2}(s_{11}, c_{11})
 \end{aligned}$$

### 3.4. Modelo de velocidad y aceleración de las piernas de robot NAO

El modelo de velocidad y aceleración del robot se expresa mediante la ecuación general de la cinemática de manipuladores seriales. Sin embargo, es necesario tener en cuenta que la velocidad y aceleración del robot se especifica mediante dos vectores de velocidades

### 3. Modelado del robot NAO

operacionales: el primero correspondiente al movimiento de la pelvis, y el segundo asociado al movimiento del pie libre. Así, se tienen las dos ecuaciones siguientes del modelo de velocidad del robot.

Para el estado de la velocidad de la pelvis:

$$\dot{\mathbf{x}}_{pel} = J_{pa}(\theta_{pa})\dot{\theta}_{pa}. \quad (3.11)$$

Para el estado de velocidad del pie libre :

$$\dot{\mathbf{x}}_{pie/pel} = J_{pl}(\theta_{pl})\dot{\theta}_{pl}. \quad (3.12)$$

donde  $J(\theta) \in \mathbb{R}^{n \times n}$  es la matriz jacobiana y  $\theta \in \mathbb{R}^n$  es el vector de velocidades articulares de la cadena cinemática de la pierna que corresponda del robot (n-6), mientras que  $\dot{\mathbf{x}} \in \mathbb{R}^n$  denota el estado de velocidad (velocidad lineal y velocidad angular) del sólido correspondiente. Los subíndices **pel** y **pie** se utilizan para referirse a la pelvis o al pie libre, respectivamente; **pl** y **pa** indica pierna libre y pierna de apoyo, respectivamente,  $\dot{\mathbf{x}}_{pie/pel}$  se refiere a la velocidad aparente del pie libre con respecto a la pelvis.

Para obtener el modelo de aceleración, se derivan las ecuaciones (3.11) y (3.12) con respecto al tiempo, así se obtiene el modelado directo de aceleración; el primero corresponde al movimiento de la pelvis, y el segundo asociado al movimiento del pie libre. Así, se tienen las dos ecuaciones siguientes del modelo de aceleración del robot.

Para el estado de la aceleración de la pelvis:

$$\ddot{\mathbf{x}}_{pel} = J_{pa}(\theta_{pa})\ddot{\theta}_{pa} + \dot{J}_{pa}\dot{\theta}_{pa}. \quad (3.13)$$

Para el estado de aceleración del pie libre :

$$\ddot{\mathbf{x}}_{pie/pel} = J_{pl}(\theta_{pl})\ddot{\theta}_{pl} + \dot{J}_{pl}\dot{\theta}_{pl}. \quad (3.14)$$

donde  $\ddot{\mathbf{x}} \in \mathbb{R}^n$  es el vector aceleración del órgano terminal correspondiente del robot.  $\ddot{\theta} \in \mathbb{R}^n$  es el vector de aceleraciones de las articulaciones.

Como se observa en las ecuaciones (3.10) - (3.13) ambos modelos requieren solamente de la determinación de la matriz jacobiana de la pierna de interés. Estas matrices se obtienen aplicando el paquete SYMORO a partir de los parámetros geométricos de las piernas que se establecieron en la Tabla 3.1. A continuación se proporcionan los resultados obtenidos.

Matriz Jacobiana del pie de apoyo a la pelvis.



$$\begin{aligned}
 J_{11} &= -(d_3 c_{\theta_2} c_{\theta_3}) - d_4 c_{\theta_1} c_{(\theta_2 + \theta_3)} \\
 J_{21} &= -(d_3 c_{\theta_2} s_{\theta_3}) - d_4 s_{\theta_1} c_{(\theta_2 + \theta_3)} \\
 J_{31} &= 0 \\
 J_{41} &= 0 \\
 J_{51} &= 0 \\
 J_{61} &= 1 \\
 J_{21} &= d_3 s_{\theta_1} s_{\theta_2} + d_4 s_{\theta_1} s_{(\theta_2 - \theta_3)} \\
 J_{22} &= -(d_3 c_{\theta_2} s_{\theta_3}) - d_4 c_{\theta_1} s_{(\theta_2 - \theta_3)} \\
 J_{32} &= -(d_3 c_{\theta_2}) - d_4 c_{(\theta_2 + \theta_3)} \\
 J_{42} &= -c_{\theta_1} \\
 J_{52} &= -s_{\theta_1} \\
 J_{62} &= 0 \\
 J_{13} &= d_4 s_{\theta_1} s_{(\theta_2 + \theta_3)} \\
 J_{23} &= -(d_3 c_{\theta_1} s_{(\theta_2 + \theta_3)}) \\
 J_{33} &= -(d_1 c_{(\theta_2 + \theta_3)}) \\
 J_{43} &= -c_{\theta_1} \\
 J_{53} &= -s_{\theta_1} \\
 J_{63} &= 0 \\
 J_{14} &= 0 \\
 J_{24} &= 0 \\
 J_{34} &= 0 \\
 J_{44} &= -c_{\theta_1} \\
 J_{54} &= -s_{\theta_1} \\
 J_{64} &= 0 \\
 J_{15} &= 0 \\
 J_{25} &= 0 \\
 J_{35} &= 0 \\
 J_{45} &= s_{\theta_1} s_{(\theta_2 - \theta_3 + \theta_4)} \\
 J_{55} &= -(c_{\theta_1} s_{(\theta_2 + \theta_3 + \theta_4)}) \\
 J_{65} &= -c_{(\theta_2 + \theta_3 + \theta_4)} \\
 J_{16} &= 0 \\
 J_{26} &= 0 \\
 J_{36} &= 0 \\
 J_{46} &= -(c_{\theta_1} c_{(\frac{3\pi}{4} + \theta_5)}) - c_{(\theta_2 + \theta_3 + \theta_4)} s_{\theta_1} s_{(\frac{3\pi}{4} - \theta_5)} \\
 J_{56} &= -(s_{\theta_1} c_{(\frac{3\pi}{4} - \theta_5)}) + c_{(\theta_2 + \theta_3 + \theta_4)} c_{\theta_1} s_{(\frac{3\pi}{4} + \theta_5)} \\
 J_{66} &= -(s_{(\theta_1 + \theta_2 + \theta_4)} s_{(\frac{3\pi}{4} - \theta_5)})
 \end{aligned}$$

### 3. Modelado del robot NAO

Matriz jacobiana de la pelvis al pie libre

$$\begin{aligned}
 J_{11} &= d_{10}c_{\theta_7}c_{(\frac{3\pi}{4}-\theta_8)}c_{\theta_9} + d_{11}c_{\theta_7}c_{(\frac{3\pi}{4}-\theta_8)}c_{(\theta_{10}+\theta_9)} + d_{10}s_{\theta_7}s_{\theta_9} + d_{11}s_{\theta_7}s_{(\theta_{10}+\theta_9)} \\
 J_{21} &= 0 \\
 J_{31} &= -(d_{10}c_{\theta_7}c_{(\frac{3\pi}{4}-\theta_8)}c_{\theta_9}) - d_{11}s_{\theta_7}c_{(\frac{3\pi}{4}-\theta_8)}c_{(\theta_{10}+\theta_9)} + d_{10}c_{\theta_7}s_{\theta_9} - d_{11}c_{\theta_7}s_{(\theta_{10}+\theta_9)} \\
 J_{41} &= 0 \\
 J_{51} &= 0 \\
 J_{61} &= 0 \\
 J_{12} &= d_{10}c_{\theta_7}s_{\theta_7}s_{(\frac{3\pi}{4}-\theta_8)} + d_{11}c_{(\theta_{10}+\theta_9)}s_{\theta_7}s_{(\frac{3\pi}{4}-\theta_8)} \\
 J_{22} &= -(d_{10}c_{(\frac{3\pi}{4}-\theta_8)}c_{\theta_9}) - d_{11}c_{(\frac{3\pi}{4}-\theta_8)}c_{(\theta_{10}+\theta_9)} \\
 J_{32} &= d_{10}c_{\theta_7}c_{\theta_9}s_{(\frac{3\pi}{4}-\theta_8)} + d_{11}c_{(\theta_{10}+\theta_9)}c_{\theta_7}s_{(\frac{3\pi}{4}-\theta_8)} \\
 J_{42} &= c_{\theta_7} \\
 J_{52} &= 0 \\
 J_{62} &= -s_{\theta_7} \\
 J_{13} &= -(d_{10}c_{\theta_7}c_{\theta_9}) - d_{11}c_{\theta_7}c_{(\theta_{10}+\theta_9)} - d_{10}c_{(\frac{3\pi}{4}-\theta_8)}s_{\theta_7}s_{\theta_9} - d_{11}c_{(\frac{3\pi}{4}-\theta_8)}s_{\theta_7}s_{(\theta_{10}+\theta_9)} \\
 J_{23} &= -(d_{10}s_{(\frac{3\pi}{4}-\theta_8)}s_{\theta_9}) - d_{11}s_{(\frac{3\pi}{4}-\theta_8)}s_{(\theta_{10}+\theta_9)} \\
 J_{33} &= d_{10}s_{\theta_7}c_{\theta_9} + d_{11}s_{\theta_7}c_{(\theta_{10}+\theta_9)} - d_{10}c_{(\frac{3\pi}{4}-\theta_8)}c_{\theta_7}s_{\theta_9} - d_{11}c_{(\frac{3\pi}{4}-\theta_8)}c_{\theta_7}s_{(\theta_{10}+\theta_9)} \\
 J_{43} &= s_{\theta_7}s_{(\frac{3\pi}{4}-\theta_8)} \\
 J_{53} &= -c_{(\frac{3\pi}{4}-\theta_8)} \\
 J_{63} &= c_{\theta_7}s_{(\frac{3\pi}{4}-\theta_8)} \\
 J_{14} &= -d_{11}c_{\theta_7}c_{(\theta_{10}+\theta_9)} - d_{11}c_{(\frac{3\pi}{4}-\theta_8)}s_{(\theta_{10}+\theta_9)}s_{\theta_7} \\
 J_{24} &= -d_{11}s_{(\frac{3\pi}{4}-\theta_8)}s_{(\theta_{10}+\theta_9)} \\
 J_{34} &= d_{11}s_{\theta_7}c_{(\theta_{10}+\theta_9)} - d_{11}c_{(\frac{3\pi}{4}-\theta_8)}s_{(\theta_{10}+\theta_9)}c_{\theta_7} \\
 J_{44} &= s_{\theta_7}s_{(\frac{3\pi}{4}-\theta_8)} \\
 J_{54} &= c_{(\frac{3\pi}{4}-\theta_8)} \\
 J_{64} &= c_{\theta_7}s_{(\frac{3\pi}{4}-\theta_8)} \\
 J_{15} &= 0 \\
 J_{25} &= 0 \\
 J_{35} &= 0 \\
 J_{45} &= s_{\theta_7}s_{(\frac{3\pi}{4}-\theta_8)} \\
 J_{55} &= -c_{(\frac{3\pi}{4}-\theta_8)} \\
 J_{65} &= c_{\theta_7}s_{(\frac{3\pi}{4}-\theta_8)} \\
 J_{16} &= 0 \\
 J_{26} &= 0 \\
 J_{36} &= 0 \\
 J_{46} &= c_{\theta_7}c_{(\theta_{10}+\theta_{11}+\theta_9)} + c_{(\frac{3\pi}{4}-\theta_8)}s_{\theta_7}s_{(\theta_{10}+\theta_{11}+\theta_9)} \\
 J_{56} &= s_{(\frac{3\pi}{4}-\theta_8)}s_{(\theta_{10}+\theta_{11}+\theta_9)} \\
 J_{66} &= -s_{\theta_7}c_{(\theta_{10}+\theta_{11}+\theta_9)} - c_{(\frac{3\pi}{4}-\theta_8)}c_{\theta_7}s_{(\theta_{10}+\theta_{11}+\theta_9)}
 \end{aligned}$$

### 3.5. Modelado dinámico mediante las ecuaciones de Euler-Lagrange

El modelo dinámico de un robot consiste en una ecuación diferencial (ordinaria) vectorial en términos de las posiciones, velocidades y aceleraciones, ya sean articulares o cartesianas, generalmente de segundo orden [45].

Los parámetros dinámicos del robot NAO (longitudes, masas, e inercias) se encuentran descritos en las Tablas 3.8, 3.9 y 3.10. Es importante señalar que las posiciones locales de los centros de masa encontrados en la literatura, [49] (y mostrados en la Tabla 3.8) son vectores que parten de la articulación anterior más cercana a la masa  $m_c$  con  $c = 1, 2, \dots, 6$  (ver Figura 3.4a), pero tomando en cuenta un marco  $\sum_{b_c}$  con una orientación idéntica a la del marco del mundo  $\sum_M$  (es decir,  $x$  hacia el frente,  $y$  hacia la derecha y  $z$  hacia arriba) cuando el robot NAO se encuentra en posición cero, como se muestra en la Figura 3.3 a; sin embargo para aplicar el método de forma correcta, es necesario encontrar el vector de posición local del centro de masa referido al marco  $\sum_n$  más próximo unido a la articulación anterior más cercana (ver Figura 3.3b), para esto es necesario realizar la siguiente rotación constante.

$${}^n \mathbf{l}_{m_c} = {}^M R_{(0)}^{b_c} \mathbf{l}_{m_c}, \quad (3.15)$$

donde  ${}^n \mathbf{l}_{m_c} \in \mathbb{R}^3$  es el vector de posición de la masa  $m_c$  respecto al marco  $\sum_n$  más próximo unido a la articulación anterior más cercana a la masa  $m_c$ ,  ${}^M R_{(0)}^{b_c}$  es la matriz de rotación constante del marco  $\sum_M$  al marco  $\sum_n$ , con valores  $\theta_i = 0$ , para  $i = 1, 2, \dots, 12$ ;  ${}^b_c \mathbf{l}_{m_c}$  es el vector de posición de la masa  $m_c$  respecto al marco  $\sum_{b_c}$ .

Una vez hecho lo anterior, el siguiente paso para obtener el modelo dinámico consiste en calcular las posiciones globales de los centros de masa de cada eslabón con respecto a un marco fijo, en el caso del modelado en simple soporte del robot NAO, se escoge como marco fijo el situado en el pie de apoyo, denominado marco  $\sum_j$ , como se muestra en la Figura 3.4.

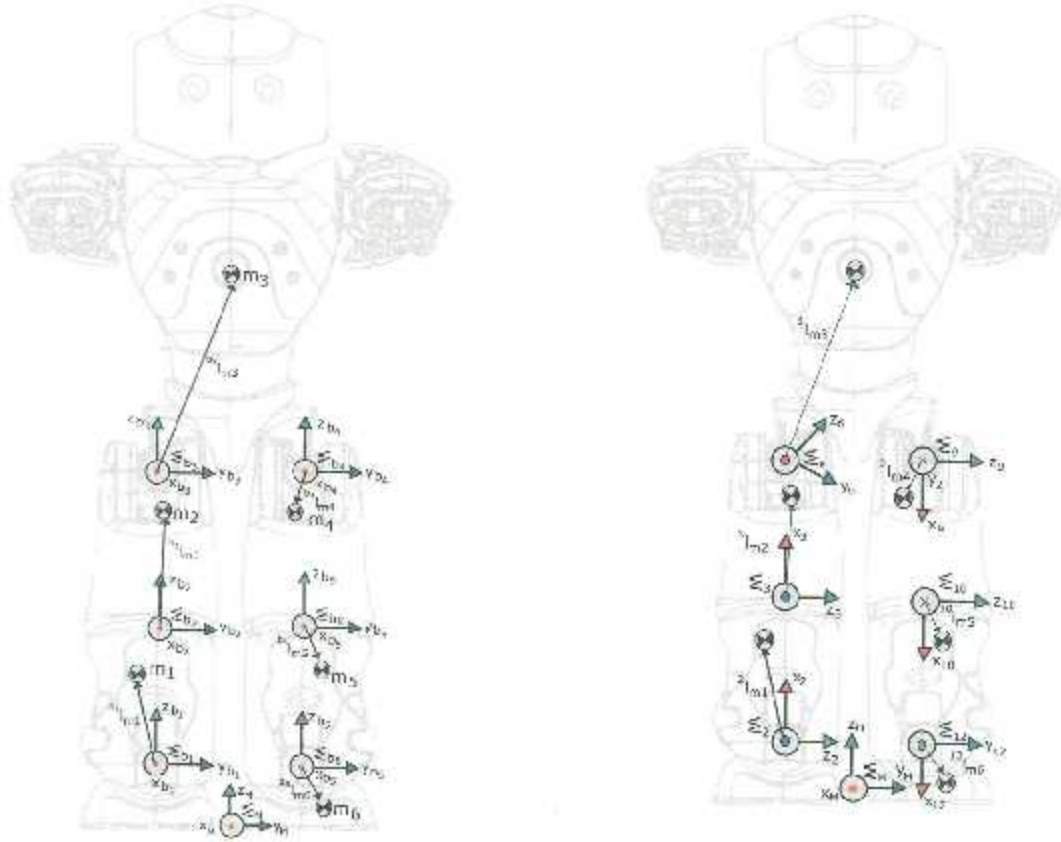
Para el cálculo de las posiciones globales de cada centro de masa del robot, se hará uso de la siguiente ecuación:

$$\mathbf{p}_{m_c} = \left( \prod_{j=1}^n T_j^{j-1} \right) {}^n \mathbf{l}_{m_c}, \quad (3.16)$$

donde  $\mathbf{p}_{m_c} = [p_{m_{cx}}, p_{m_{cy}}, p_{m_{cz}}]^T$  es el vector de posición global de la masa  $m_c$ , respecto al marco fijo  $\sum_j$ ,  $T_j^{j-1} \in \mathbb{R}^{4 \times 4}$ , y representa la matriz de transformación homogénea que

<sup>1</sup>Se utiliza  $\sum_M$  en lugar de  $\sum_b$ , ya que tienen la misma orientación.

### 3. Modelado del robot NAO



(a) Centros de masas con respecto al marco  $\Sigma_{b_i}$ . (b) Centros de masas con respecto al marco  $\Sigma_{n_i}$

Figura 3.3: Posiciones de los centros de masa del robot NAO

indica la posición y orientación del marco  $\Sigma_j$  respecto al marco  $\Sigma_{j-1}$ , los cuales fueron definidos en el Capítulo 3, y finalmente  $n$  representa el marco de referencia anterior más cercano a la masa  $m_i$ , por lo tanto las posiciones de los 6 centros de masa con respecto al pie de apoyo se calculan de la siguiente manera:

$$p_{m1} = (T_1^0 T_2^1)^2 l_{m1} \quad (3.17)$$

$$p_{m2} = (T_1^0 T_2^1 T_3^2)^3 l_{m2} \quad (3.18)$$

$$p_{m3} = (T_1^0 T_2^1 T_3^2 T_4^3 T_5^4 T_6^5)^6 l_{m3} \quad (3.19)$$

$$p_{m4} = (T_1^0 T_2^1 T_3^2 T_4^3 T_5^4 T_6^5 T_7^6 T_8^7 T_9^8)^9 l_{m4} \quad (3.20)$$

$$p_{m5} = (T_1^0 T_2^1 T_3^2 T_4^3 T_5^4 T_6^5 T_7^6 T_8^7 T_9^8 T_{10}^9)^{10} l_{m5} \quad (3.21)$$

$$p_{m6} = (T_1^0 T_2^1 T_3^2 T_4^3 T_5^4 T_6^5 T_7^6 T_8^7 T_9^8 T_{10}^9 T_{11}^{10} T_{12}^{11})^{12} l_{m6} \quad (3.22)$$

Ahora, obteniendo la derivada temporal de las ecuaciones (3.17) - (3.22), se obtienen las velocidades de cada centro de masa las cuales están dadas por



### 3. Modelado del robot NAO

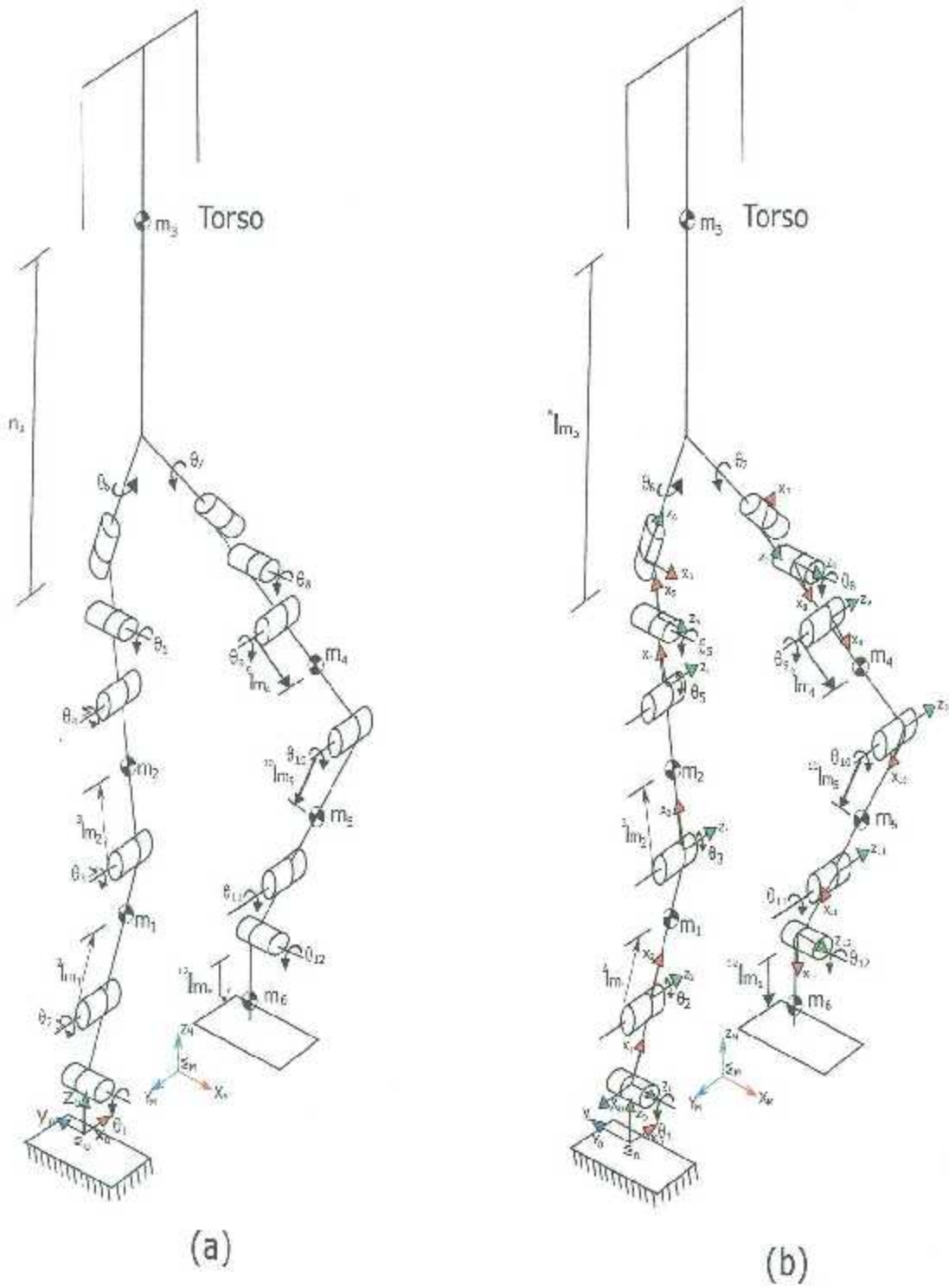


Figura 3.4: Modelo geométrico del robot NAO

### 3. Modelado del robot NAO

Tabla 3.8: Posiciones de los centros de masa de cada eslabon respecto al marco  $\sum_{i=bc}$  unido a la articulación anterior más cercana

Vector de centros de masas (mm)	
${}^{b_1}l_{m1}$	$= [3.66, -1.56, 38.72]^T$
${}^{b_2}l_{m2}$	$= [-3.68, 0.03, 66.79]^T$
${}^{b_3}l_{m3}$	$= [-4.80, 50.06, 127.27]^T$
${}^{b_4}l_{m4}$	$= [-3.38, -0.03, -33.21]^T$
${}^{b_5}l_{m5}$	$= [3.66, 1.52, -64.03]^T$
${}^{b_6}l_{m6}$	$= [24.89, 3.3, -32.08]^T$

Tabla 3.9: Valores de las masas de los eslabones del robot NAO

Masas (gramos)
$m_1 = 435.98$ (Tibia derecha)
$m_2 = 605.72$ (Fémur derecho)
$m_3 = 1026.28$ (Torso)
$m_4 = 605.72$ (Fémur Izquierdo)
$m_5 = 435.98$ (Tibia derecha)
$m_6 = 216.98$ (Pie libre)

Tabla 3.10: Valores de los tensores de inercias respecto al marco unido al centro de masa de su eslabón

Tensor de inercia	
$I_1 = \text{diag}\{I_{xx1}, I_{yy1}, I_{zz1}\}$	$= \text{diag}\{0.0012, 0.0012, 0.0006\}$
$I_2 = \text{diag}\{I_{xx2}, I_{yy2}, I_{zz2}\}$	$= \text{diag}\{0.0018, 0.0018, 0.0010\}$
$I_3 = \text{diag}\{I_{xx3}, I_{yy3}, I_{zz3}\}$	$= \text{diag}\{0.0049, 0.0047, 0.0016\}$
$I_4 = \text{diag}\{I_{xx4}, I_{yy4}, I_{zz4}\}$	$= \text{diag}\{0.0018, 0.0018, 0.0010\}$
$I_5 = \text{diag}\{I_{xx5}, I_{yy5}, I_{zz5}\}$	$= \text{diag}\{0.0012, 0.0012, 0.0006\}$
$I_6 = \text{diag}\{I_{xx6}, I_{yy6}, I_{zz6}\}$	$= \text{diag}\{0.0012, 0.0012, 0.0006\}$

$$v_1 = \dot{p}_{m1} = [\dot{p}_{m1x}, \dot{p}_{m1y}, \dot{p}_{m1z}]^T \quad (3.23)$$

$$v_2 = \dot{p}_{m2} = [\dot{p}_{m2x}, \dot{p}_{m2y}, \dot{p}_{m2z}]^T \quad (3.24)$$

$$v_3 = \dot{p}_{m3} = [\dot{p}_{m3x}, \dot{p}_{m3y}, \dot{p}_{m3z}]^T \quad (3.25)$$

$$v_4 = \dot{p}_{m4} = [\dot{p}_{m4x}, \dot{p}_{m4y}, \dot{p}_{m4z}]^T \quad (3.26)$$

$$v_5 = \dot{p}_{m5} = [\dot{p}_{m5x}, \dot{p}_{m5y}, \dot{p}_{m5z}]^T \quad (3.27)$$

$$v_6 = \dot{p}_{m6} = [\dot{p}_{m6x}, \dot{p}_{m6y}, \dot{p}_{m6z}]^T \quad (3.28)$$

Por otro lado, es necesario también calcular la velocidad angular de cada centro de masa. Para esto, se tiene que:

$$\omega_j = \omega_{j-1} + {}^0R\dot{\theta}_j^j \hat{z}_j \quad (3.29)$$

### 3. Modelado del robot NAO

donde  $j = 1, 2, \dots, 12$ ,  ${}^0R^j$  representa la matriz de rotación desde el marco  $\sum_0$  hasta el marco  $\sum_j$ , y finalmente el vector  ${}^j\hat{z} \in \mathbb{R}^3$  representa el eje sobre el cual tiene lugar la rotación. Las ecuaciones que representan las velocidades angulares de cada una de las articulaciones, con respecto al marco fijo  $\sum_0$ , se muestran a continuación.

$$\omega_1 = \omega_0 + {}^0R\dot{\theta}_1^1 \hat{z} \quad (3.30)$$

$$\omega_2 = \omega_1 + {}^1R\dot{\theta}_2^2 \hat{z} \quad (3.31)$$

$$\omega_3 = \omega_2 + {}^2R\dot{\theta}_3^3 \hat{z} \quad (3.32)$$

$$\omega_4 = \omega_3 + {}^3R\dot{\theta}_4^4 \hat{z} \quad (3.33)$$

$$\omega_5 = \omega_4 + {}^4R\dot{\theta}_5^5 \hat{z} \quad (3.34)$$

$$\omega_6 = \omega_5 + {}^5R\dot{\theta}_6^6 \hat{z} \quad (3.35)$$

$$\omega_7 = \omega_6 + {}^6R\dot{\theta}_7^7 \hat{z} \quad (3.36)$$

$$\omega_8 = \omega_7 + {}^7R\dot{\theta}_8^8 \hat{z} \quad (3.37)$$

$$\omega_9 = \omega_8 + {}^8R\dot{\theta}_9^9 \hat{z} \quad (3.38)$$

$$\omega_{10} = \omega_9 + {}^9R\dot{\theta}_{10}^{10} \hat{z} \quad (3.39)$$

$$\omega_{11} = \omega_{10} + {}^{10}R\dot{\theta}_{11}^{11} \hat{z} \quad (3.40)$$

$$\omega_{12} = \omega_{11} + {}^{11}R\dot{\theta}_{12}^{12} \hat{z} \quad (3.41)$$

donde  $\omega_0 = [0, 0, 0]^T$  es la velocidad angular del marco  $\sum_0$

Una vez que las velocidades lineales y angulares del centro de masa son obtenidas, se procede a calcular la energía cinética del sistema dada por la siguiente ecuación:

$$\begin{aligned} K = & \frac{1}{2}m_1 \|v_1\|^2 + \frac{1}{2}\omega_2^{T0} I_1 \omega_2 + \frac{1}{2}m_2 \|v_2\|^2 + \frac{1}{2}\omega_3^{T0} I_2 \omega_3 + \\ & \frac{1}{2}m_3 \|v_3\|^2 + \frac{1}{2}\omega_6^{T0} I_3 \omega_6 + \frac{1}{2}m_4 \|v_4\|^2 + \frac{1}{2}\omega_5^{T0} I_4 \omega_5 + \\ & \frac{1}{2}m_5 \|v_5\|^2 + \frac{1}{2}\omega_{10}^{T0} I_5 \omega_{10} + \frac{1}{2}m_6 \|v_6\|^2 + \frac{1}{2}\omega_{12}^{T0} I_6 \omega_{12} \end{aligned} \quad (3.42)$$

donde  $m_1$  a  $m_6$  representan las masas del robot NAO, cuyos valores se describen en la Tabla 3.9,  ${}^0I_c$  representa el tensor de inercia del eslabón  $c$  con respecto al marco  $\sum_0$ , los cuales son obtenidos mediante la siguiente relación [50].

$${}^0I_c = R_c I_c R_c^T, \quad (3.43)$$

donde

$$R_c = {}^0R \begin{pmatrix} M \\ n \end{pmatrix} R(0)^{-1}, \quad (3.44)$$

### 3. Modelado del robot NAO

con  ${}^0R$  siendo la matriz de relación del marco  $\sum_0$  al marco  $\sum_n$ , y  ${}^M R(0)$  la matriz de rotación constante ( $\theta_i = 0$  para  $i = 1, 2, \dots, 12$ ) del marco  $\sum_M$  al marco  $\sum_n$ , siendo  $n$  el marco anterior al eslabón  $e$ . Los valores de los tensores de inercia  $I_{e_i}$  de cada eslabón con respecto a su centro de masa, se describen en la Tabla 3.10.

Para el cálculo de la energía potencial  $V$ , es necesario calcular la altura de cada centro de masa medido con respecto al marco fijo  $\sum_0$ , la cual se puede expresar como:

$$h_1 = p_{m_{1z}} \quad (3.45)$$

$$h_2 = p_{m_{2z}} \quad (3.46)$$

$$h_3 = p_{m_{3z}} \quad (3.47)$$

$$h_4 = p_{m_{4z}} \quad (3.48)$$

$$h_5 = p_{m_{5z}} \quad (3.49)$$

$$h_6 = p_{m_{6z}} \quad (3.50)$$

Se utiliza la componente en  $z$  de  $p_{mc}$  ya que como está referido al marco  $\sum_0$ ,  $z_0$  indica la altura.

Luego se sustituyen las ecuaciones de (3.45)-(3.50) en la siguiente ecuación

$$V = m_1gh_1 + m_2gh_2 + m_3gh_3 + m_4gh_4 + m_5gh_5 + m_6gh_6 \quad (3.51)$$

donde  $g$  representa la magnitud de la fuerza de gravedad.

El siguiente paso es calcular el Lagrangiano del sistema, donde se hará uso de las ecuaciones (3.42) y (3.51), el cual se define como:

$$L = K - V \quad (3.52)$$

Ahora es posible aplicar la ecuación de movimiento de Euler-Lagrange dada por

$$\tau_i = \frac{d}{dt} \left[ \frac{\partial L}{\partial \dot{\theta}_i} \right] - \frac{\partial L}{\partial \theta_i} \quad (3.53)$$

donde  $i = 1, 2, 3, \dots, 12$  y  $\tau_i$  es el par aplicado en el eje de giro de la coordenada generalizada, esto es, el par aplicado en cada uno de los ejes de giro en el tobillo, la rodilla y la cadera del pie de soporte y en el tobillo, la rodilla y la cadera del pie libre.



### 3. Modelado del robot NAO

---

La ecuación generalizada de movimiento de cada actuador considerando  $M(\boldsymbol{\theta})$  para el efecto inercial,  $C(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}})$  para efecto centrífugo y de coriolis, y  $G(\boldsymbol{\theta})$  para el efecto gravitatorio se describe en la ecuación (3.54).

$$\tau = M(\boldsymbol{\theta})\ddot{\boldsymbol{\theta}} + C(\boldsymbol{\theta}, \dot{\boldsymbol{\theta}}) + G(\boldsymbol{\theta}) \quad (3.54)$$

El modelo dinámico obtenido es usado en las simulaciones del Capítulo 6, dado más adelante.

# Capítulo 4

## Patrón de Marcha del Robot NAO

Para la descripción de la cadena cinemática de las piernas del robot, a los eslabones de éstas se les asignan marcos de referencia ortonormales aplicando la convención modificada de Denavit-Hartenberg (Khalil y Kleifinger, 1986). Bajo esta convención, dichos marcos se numeran de manera consecutiva desde el cero, asignado al eslabón fijo (pie de apoyo), hasta el 12, asignado al eslabón que corresponde al pie en desplazamiento (pie libre) durante la marcha. En consecuencia, y tomando en cuenta que durante el proceso de marcha la función del pie de apoyo y la del pie libre la realizan de manera alternativa los pies izquierdo y derecho, se establecerán dos conjuntos de marcos de referencia a la misma cadena cinemática: el primero se aplicará cuando el pie apoyado sea el derecho (pda), y el segundo sea el pie izquierdo el de soporte (pia).

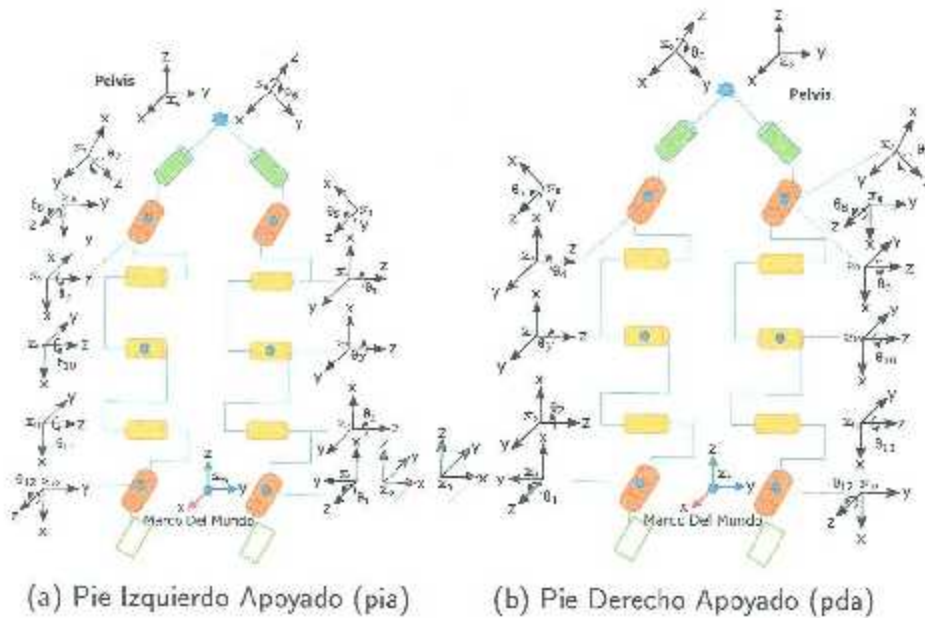


Figura 4.1: Marcos de las piernas en cada apoyo

#### 4. Patrón de Marcha del Robot NAO

El patrón de marcha se basa en leyes de movimiento de tipo cicloidal. Estos patrones están basados en la propuesta [47] y [48], sin embargo los parámetros que definen estos patrones de marcha, en el caso del robot NAO, han sido elegidos de tal forma que el robot pueda realizar el caminado. Las trayectorias propuestas permiten que el robot inicie en una posición casi vertical y que desplace su cadera hacia un lado al momento de dar un paso, ayudando de esta forma a posicionar el centro de masa con el polígono de soporte con la finalidad de que la marcha sea estable. Para lograr posicionar el centro de masa con el polígono de soporte, se utilizó el programa "webots for nao". Para el robot NAO estas trayectorias se definen mediante las siguientes ecuaciones:

Para la cadera

$$x_{pel} = \begin{cases} x_{ca} + v_c t, & \text{si } 0 < t \leq T_1 \\ x_{ca} + \frac{s}{2}, & \text{si } T_1 < t \leq T \end{cases}$$

$$z_{pel} = \begin{cases} z_{ca} + h_c \left[ \frac{2t}{T_1} - \frac{1}{2\pi} \sin\left(\frac{4t\pi}{T_1}\right) \right], & \text{si } 0 < t \leq \frac{T_1}{2} \\ z_{ca} + h_c \left[ 1 - \left( \frac{2t - T_1}{T_1} - \frac{1}{2\pi} \sin\left(\frac{4t\pi}{T_1}\right) \right) \right], & \text{si } \frac{T_1}{2} < t \leq T_1 \\ z_{ca}, & \text{si } T_1 < t \leq T \end{cases}$$

$$y_{pel} = \begin{cases} y_{ca} + f_c \left[ \frac{2t}{T_1} - \frac{1}{2\pi} \sin\left(\frac{4t\pi}{T_1}\right) \right], & \text{si } 0 < t \leq \frac{T_1}{2} \\ y_{ca} + f_c \left[ 1 - \left( \frac{2t - T_1}{T_1} - \frac{1}{2\pi} \sin\left(\frac{4t\pi}{T_1}\right) \right) \right], & \text{si } \frac{T_1}{2} < t \leq T_1 \\ y_{ca}, & \text{si } T_1 < t \leq T \end{cases}$$

Para el pie libre

$$x_{pvc} = \begin{cases} x_{pvc} + s \left[ \frac{t}{T_1} - \frac{1}{2\pi} \sin \left( \frac{2\pi t}{T_1} \right) \right], \\ \text{si } 0 < t \leq T_1 \\ x_{pvc} + s, \\ \text{si } T_1 < t \leq T \end{cases}$$

$$z_{pvc} = \begin{cases} h_p \left[ \frac{2t}{T_1} - \frac{1}{2\pi} \sin \left( \frac{4t\pi}{T_1} \right) \right], \\ \text{si } 0 < t \leq \frac{T_1}{2} \\ h_p - h_v \left[ 1 - \left( \frac{2t - T_1}{T_1} - \frac{1}{2\pi} \sin \left( \frac{4t\pi}{T_1} \right) \right) \right], \\ \text{si } \frac{T_1}{2} < t \leq T_1 \end{cases}$$

donde:

- $X_{pet}$  = Posición deseada de la pelvis en  $X_M$ .
- $x_{ci}$  = Posición inicial de la pelvis en  $X_M$ .
- $v_c$  = Velocidad de desplazamiento en  $X_M$ .
- $Z_{pet}$  = Posición deseada de la pelvis en  $Z_M$ .
- $z_{ci}$  = Posición inicial de la pelvis en  $Z_M$ .
- $H_c$  = Desplazamiento máximo de la pelvis en  $Z_M$ .
- $Y_{pet}$  = Posición deseada de la pelvis en  $Y_M$ .
- $y_{ci}$  = Posición inicial de la pelvis en  $Y_M$ .
- $f_c$  = Desplazamiento máximo de la pelvis en  $Y_M$ .

- $X_{pvl}$  = Posición deseada del pie libre en  $X_M$ .
- $X_{pvl}$  = Posición inicial del pie libre en  $X_M$ .
- $S$  = Longitud del paso del robot.
- $Z_{pvl}$  = Posición deseada del pie libre en  $Z_M$ .
- $H_p$  = Altura del pie máxima.



## Capítulo 5

# Control de marcha mediante seguimiento de trayectorias

Utilizando los patrones de marcha, el modelado dinámico y las ecuaciones cinemáticas anteriormente descritas, se han realizado simulaciones en Simulink/MATLAB que permiten evaluar los pares aplicados por los motores durante el caminado del robot humanoide NAO. Se probó el funcionamiento del controlador PID simple. El desempeño de este controlador fue probado con una trayectoria de caminado que considera la fase de doble soporte instantánea, además de una posición completamente vertical del robot humanoide NAO.

### 5.1. Ley de control PID

La ley de control PID está definida por la siguiente ecuación:

$$\tau = K_p \bar{\theta} + K_v \dot{\bar{\theta}} - K_i \int_0^t \bar{\theta}(s) ds \quad (5.1)$$

donde  $\bar{\theta} = \theta_d - \theta$  y  $\dot{\bar{\theta}} = \dot{\theta}_d - \dot{\theta}$  denotan los vectores de error de posición y velocidad articular respectivamente, con  $K_p = \text{diag} \{3000, 1700, 750, 300, 600, 500, 200, 40, 200, 30, 100, 50\}$ ,  $K_v = \text{diag} \{3000, 1700, 750, 300, 600, 500, 200, 40, 200, 30, 50, 20\}$ , y  $K_i = \text{diag} \{3000, 1700, 750, 300, 600, 500, 200, 40, 200, 30, 30, 10\}$  matrices definidas positivas; estas ganancias fueron ajustadas manualmente de tal forma que el par aplicado por el controlador no fuera demasiado demandante.

En la Figura 5.1 se observa el diagrama a bloques del control del robot NAO; en él se observa el bloque “planación de trayectorias”. Este bloque contiene las ecuaciones que describen las trayectorias deseadas  $x_d$  y velocidades deseadas  $\dot{x}_d$  de la pelvis y del pie libre.

## 5. Control de marcha mediante seguimiento de trayectorias

Estas trayectorias corresponden al patrón de marcha visto en el Capítulo 4. Dichas trayectorias deseadas entran al bloque de la "Cinemática Inversa", el cual contiene las ecuaciones del modelado cinemático inverso de posición de las piernas, así como también las ecuaciones del modelado de velocidad, por lo tanto en dicho bloque se calculan las posiciones articulares deseadas  $q_d$  y las velocidades articulares deseadas  $\dot{q}_d$ . Estas variables entran al bloque sumador para poder hacer la diferencia entre la posición y velocidad deseada contra la posición y velocidad actual del robot, para después entrar al bloque de "control", el cual contiene la ley de control PID. Las salidas del controlador son los pares ejercidos por los motores, estos pares entran al bloque "Modelado Dinámico" para poder calcular las aceleraciones de los motores, después se integran para poder obtener las velocidades y finalmente se vuelven a integrar para poder obtener las posiciones.

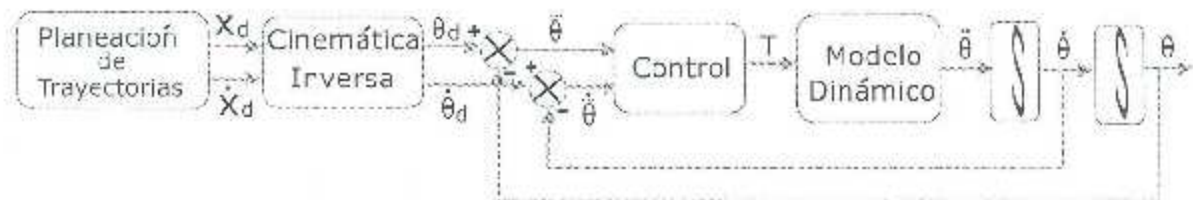


Figura 5.1: Diagrama a bloques del control de marcha mediante seguimiento de trayectorias del robot NAO

En la Figura 5.2 se muestra el controlador PID. Este controlador se realizó en Simulink de MATLAB, siguiendo el diagrama a bloques del controlador NAO. Los resultados obtenidos por este controlador se muestran en el capítulo siguiente.





## Capítulo 6

### Simulaciones

Una vez obtenidos los patrones de marcha, las ecuaciones que representan la cinemática directa e inversa, como también las ecuaciones que representan al modelo dinámico descritos anteriormente, se han realizado simulaciones en MATLAB y Simulink que permiten evaluar las posiciones articulares de las piernas del robot NAO durante toda su marcha, como también las velocidades y aceleraciones articulares, y finalmente los pares aplicados por los motores.

Las ecuaciones que especifican el patrón de marcha que se asigna al robot NAO son las que se proporcionan en el Capítulo 4. Los valores que se utilizan para los parámetros del patrón de marcha se proporcionan en la Tabla 6.1.

A continuación, en la Figura 6.1 se muestra la simulación del caminado del robot NAO, realizada en MATLAB; como se puede observar en la figura, el robot dará 4 pasos antes de detener su marcha, también se observa que el robot inclina su cadera hacia la izquierda y hacia la derecha, esto lo hace con la finalidad de posicionar su centro de masa con el polígono de soporte al momento de dar un paso, de lo contrario el robot no podría caminar.

En la Figura 6.2 se muestra que la pelvis siempre permanece derecha en cada instante del caminado, esto es gracias al patrón de marcha especificado a fin de generar un caminado cinemáticamente factible sobre un plano horizontal. En la simulación efectuada, a fin de simplificar el modelado de la cinemática inversa y teniendo en cuenta la arquitectura del NAO, se asumió que el movimiento de la pelvis es traslacional y que la normal a la superficie de la suela del pie oscilante, al momento de aterrizar en cada paso, se mantiene perpendicular a la superficie del piso.

En las Figuras 6.3 y 6.4 se muestran las 12 posiciones articulares de las piernas del robot NAO durante todo su caminado. Como se observa en las articulaciones 6 y 7,  $\theta_6$  y  $\theta_7$  son iguales entre sí en todo instante durante la marcha, condición que es compatible con las restricciones impuestas por la transmisión de potencia hacia las articulaciones 6 y 7 y, por lo tanto, la marcha del robot es factible para el patrón de marcha especificado.



Tabla 6.1: Parámetros Del Patrón de marcha del robot NAO

Parámetros	Unidades	Magnitud
$n_A$	Pasos	4
$x_{cs}$	mm	15
$v_c$	mm/seg	6
$z_{cs}$	mm	245
$h_c$	mm	0
$h_{ci}$	mm	0
$J_c$	mm	1.40
$S$	mm	60
$h_p$	mm	15

Una vez obtenidas las gráficas de posiciones articulares de las 12 articulaciones de las piernas del robot, se procede a obtener las gráficas de velocidad durante cada paso que realiza el robot NAO, estas gráficas se pueden observar en las Figuras 6.5 y 6.6.

Asimismo, ya que se obtuvieron las gráficas de posición y de velocidad, se procedió a obtener las gráficas de aceleraciones articulares, véase las Figuras 6.7 y 6.8, donde se muestran las aceleraciones articulares de cada articulación durante cada paso que realiza el robot NAO durante su marcha.

Los resultados obtenidos del control de marcha mediante seguimiento de trayectorias se presentan a continuación. A partir de las Figura 6.9 a la 6.20 se muestran las posiciones articulares de las piernas durante la tarea de seguimiento, donde la señal de color azul es la señal de referencia, y la señal de color rosa representa la señal de seguimiento del controlador PID. Las Figuras 6.21 - 6.32 muestran los errores de posición articular durante la realización de la tarea de seguimiento. Finalmente las Figuras 6.33 - 6.44 muestran los pares articulares.

## 6. Simulaciones

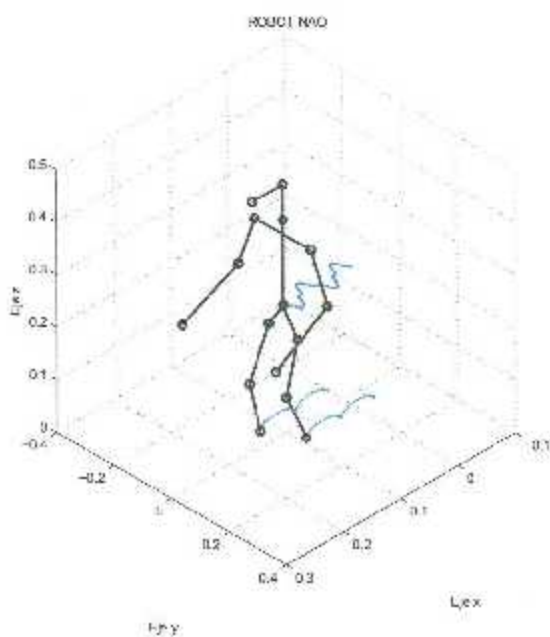


Figura 6.1: Vista frontal del caminado del robot NAO

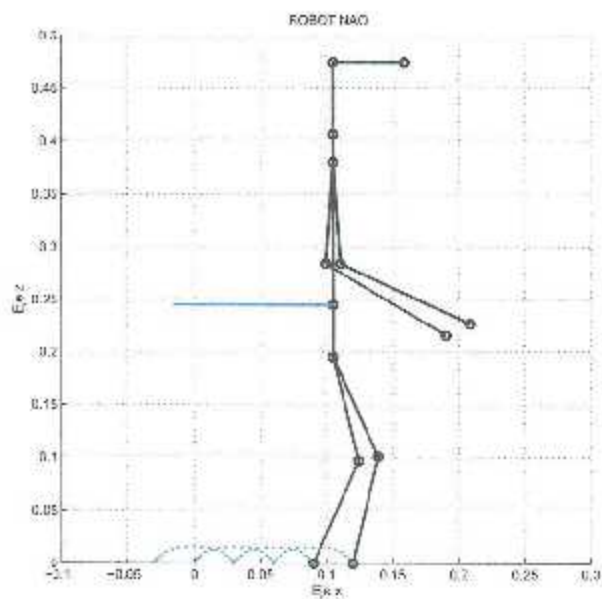


Figura 6.2: Vista lateral del caminado del robot NAO

## 6. Simulaciones

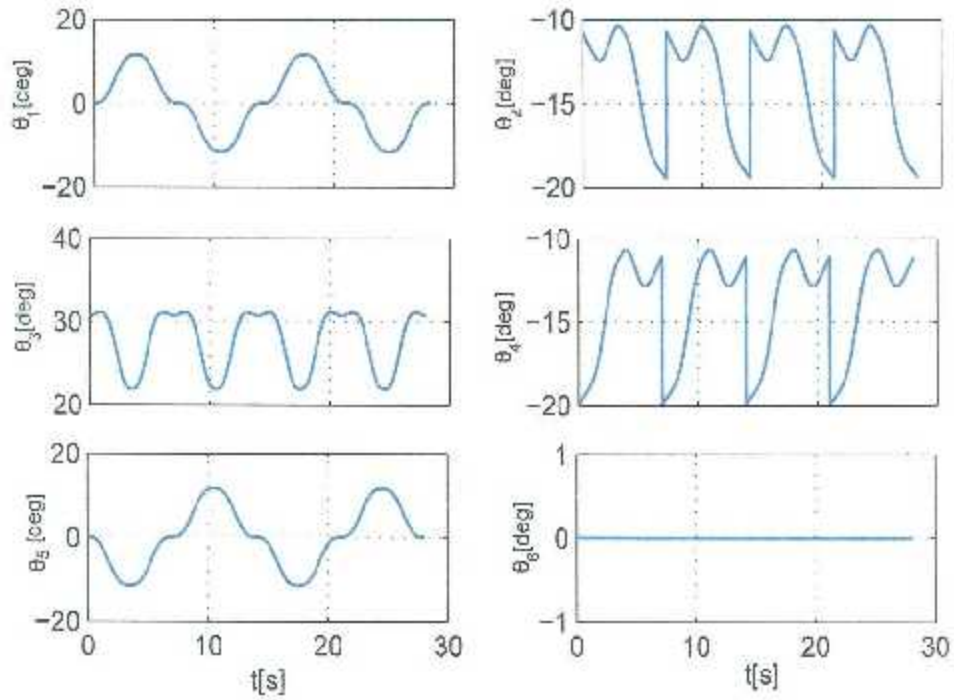


Figura 6.3: Posiciones articulares de  $\theta_1$  a  $\theta_6$  durante el caminado del robot NAO

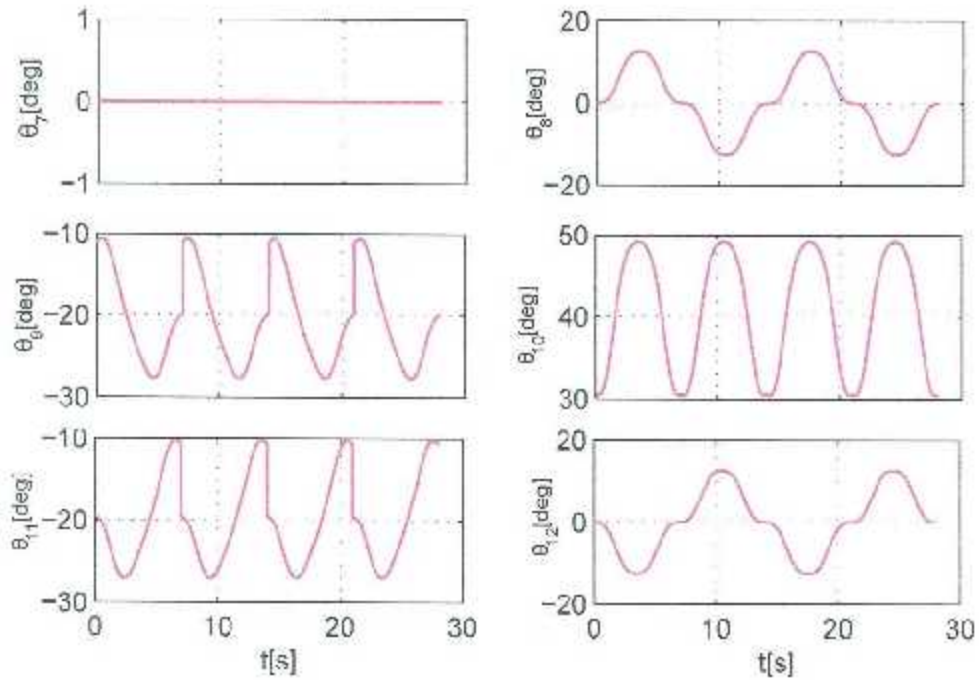


Figura 6.4: Posiciones articulares de  $\theta_7$  a  $\theta_{12}$  durante el caminado del robot NAO

## 6. Simulaciones

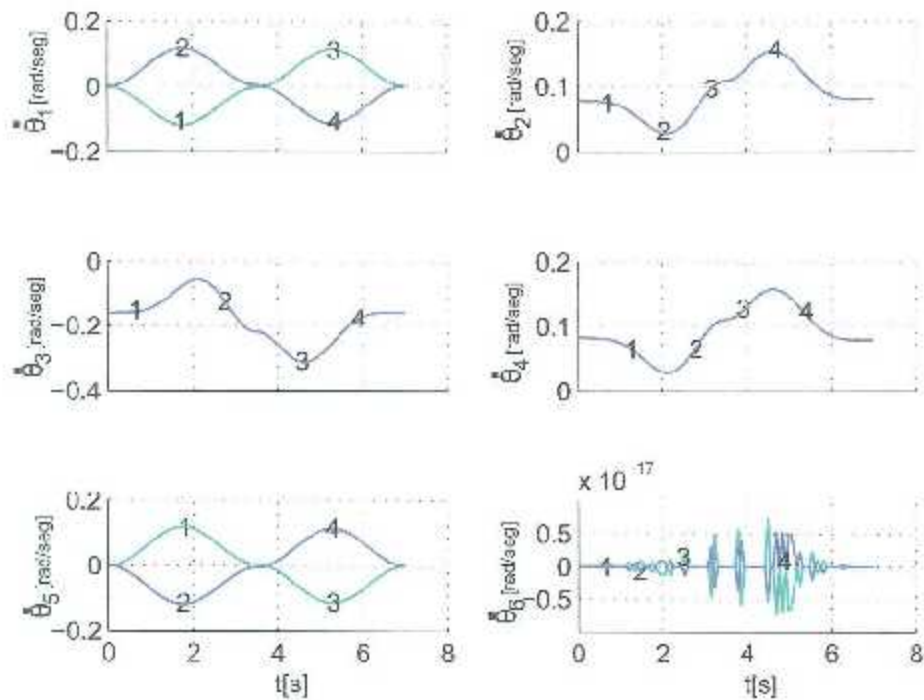


Figura 6.5: Velocidades articulares de  $\dot{\theta}_1$  a  $\dot{\theta}_6$  durante el caminado del robot NAO

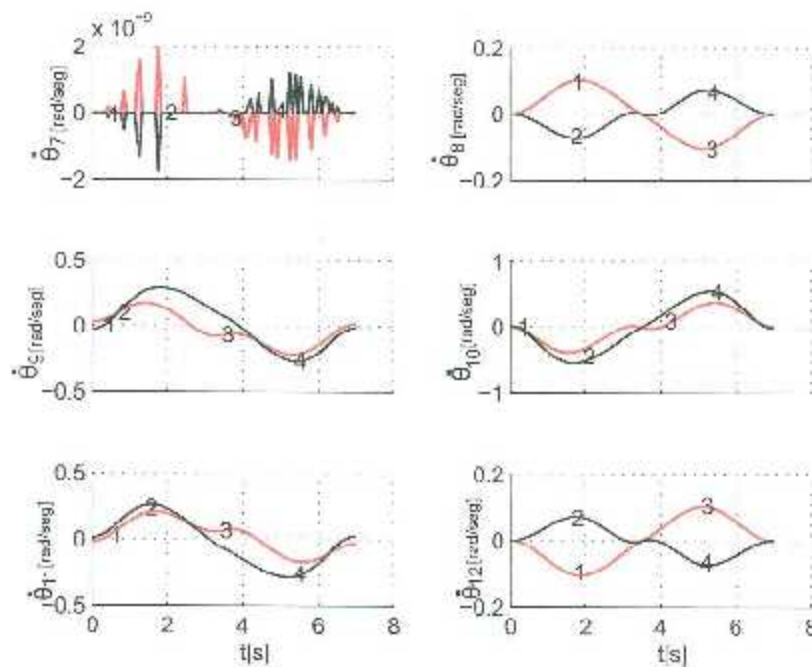


Figura 6.6: Velocidades articulares de  $\dot{\theta}_7$  a  $\dot{\theta}_{12}$  durante el caminado del robot NAO



## 6. Simulaciones

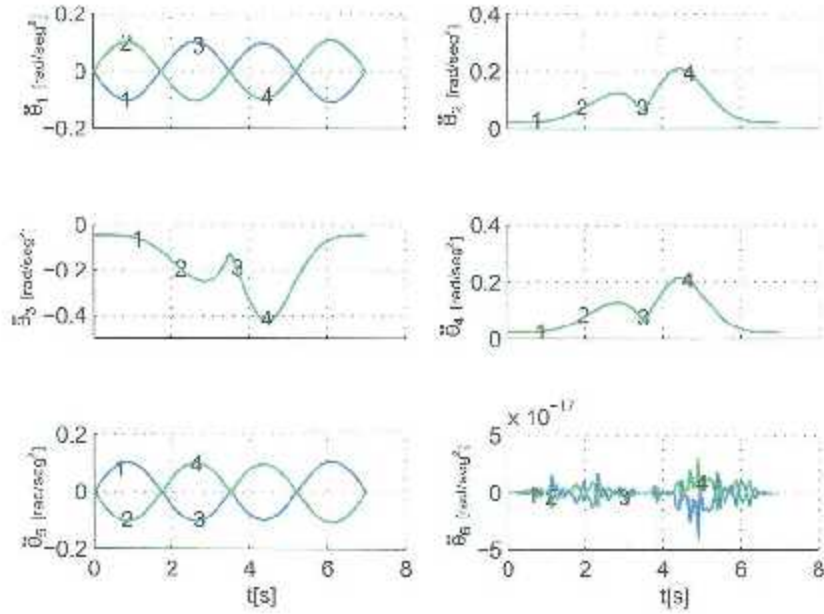


Figura 6.7: Aceleraciones articulares de  $\ddot{\theta}_1$  a  $\ddot{\theta}_6$  durante el caminado del robot NAO

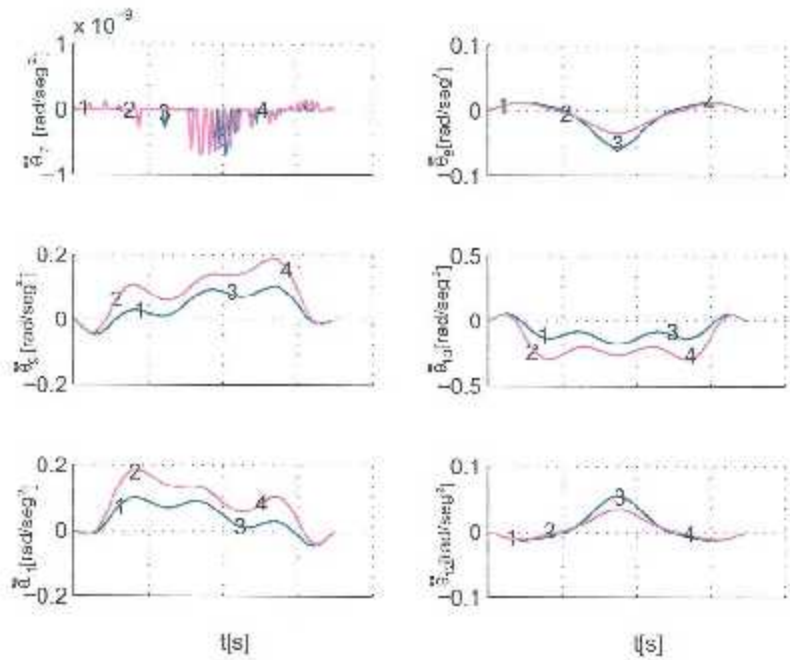


Figura 6.8: Aceleraciones articulares de  $\ddot{\theta}_7$  a  $\ddot{\theta}_{12}$  durante el caminado del robot NAO

## 6. Simulaciones

---

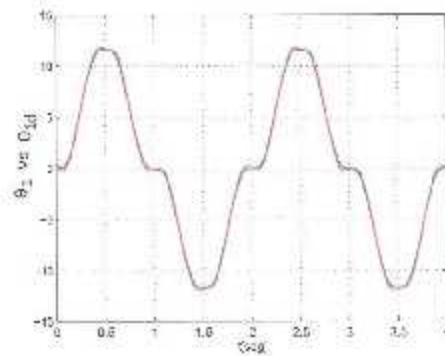


Figura 6.9: Posición articular  $\theta_1$  contra la posición deseada  $\theta_{1d}$  de la pierna durante la tarea de seguimiento.

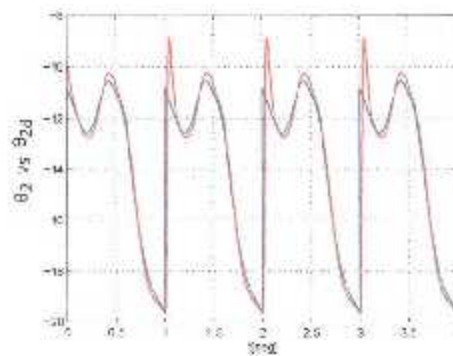


Figura 6.10: Posición articular  $\theta_2$  contra la posición deseada  $\theta_{2d}$  de la pierna durante la tarea de seguimiento.

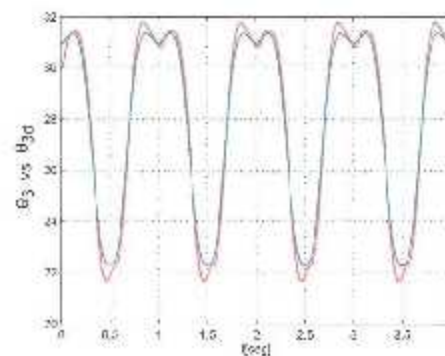


Figura 6.11: Posición articular  $\theta_3$  contra la posición deseada  $\theta_{3d}$  de la pierna durante la tarea de seguimiento.

## 6. Simulaciones

---

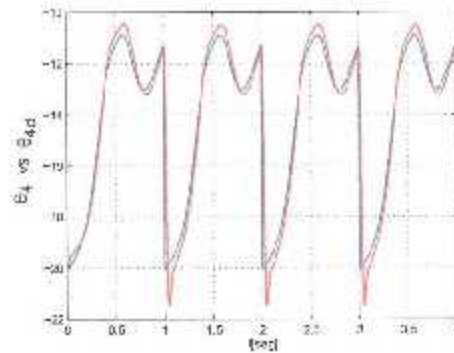


Figura 6.12: Posición articular  $\theta_4$  contra la posición deseada  $\theta_{4d}$  de la pierna durante la tarea de seguimiento.

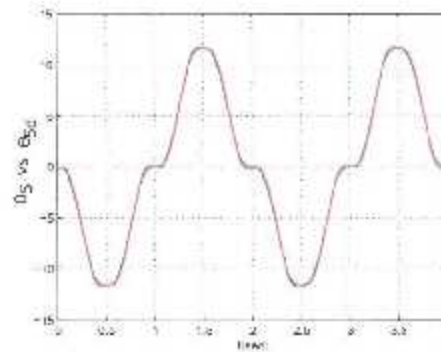


Figura 6.13: Posición articular  $\theta_5$  contra la posición deseada  $\theta_{5d}$  de la pierna durante la tarea de seguimiento.

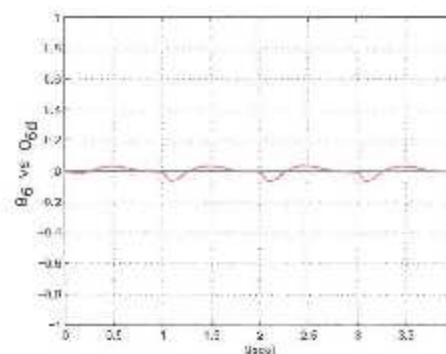


Figura 6.14: Posición articular  $\theta_6$  contra la posición deseada  $\theta_{6d}$  de la pierna durante la tarea de seguimiento.

## 6. Simulaciones

---

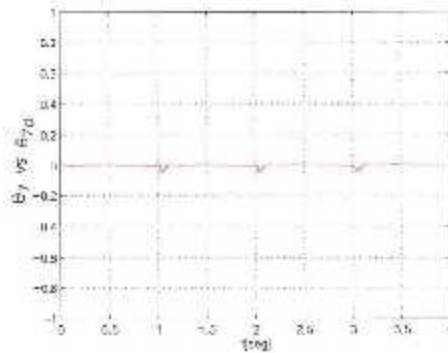


Figura 6.15: Posición articular  $\theta_7$  contra la posición deseada  $\theta_{7d}$  de la pierna durante la tarea de seguimiento

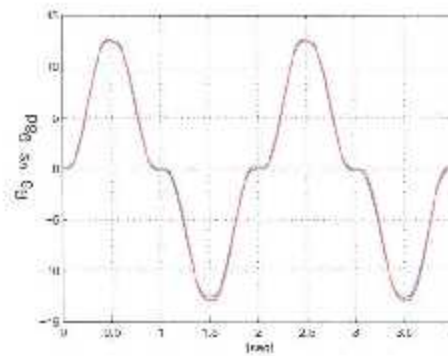


Figura 6.16: Posición articular  $\theta_8$  contra la posición deseada  $\theta_{8d}$  de la pierna durante la tarea de seguimiento

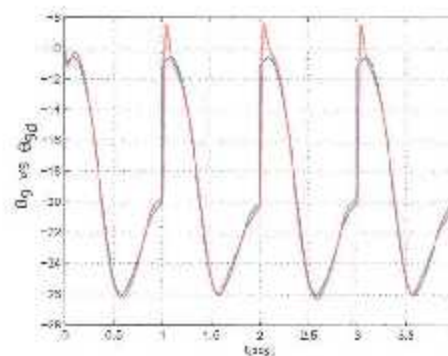


Figura 6.17: Posición articular  $\theta_9$  contra la posición deseada  $\theta_{9d}$  de la pierna durante la tarea de seguimiento



## 6. Simulaciones

---

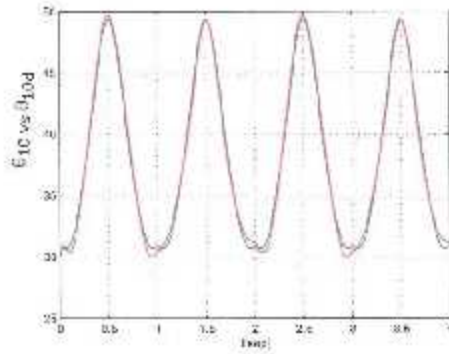


Figura 6.18: Posición articular  $\theta_{10}$  contra la posición deseada  $\theta_{10d}$  de la pierna durante la tarea de seguimiento

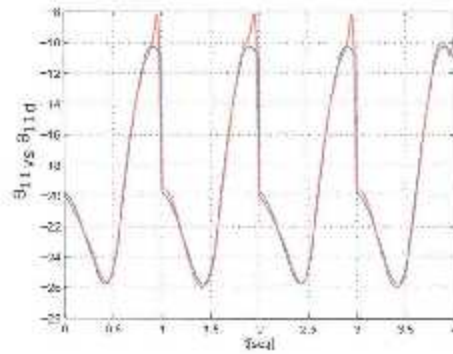


Figura 6.19: Posición articular  $\theta_{11}$  contra la posición deseada  $\theta_{11d}$  de la pierna durante la tarea de seguimiento

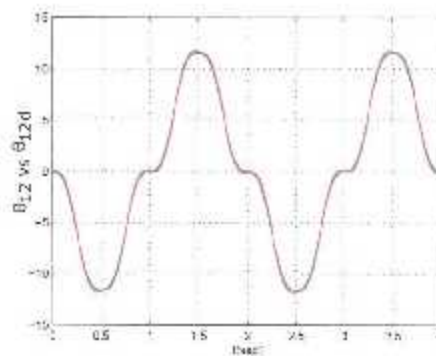


Figura 6.20: Posición articular  $\theta_{12}$  contra la posición deseada  $\theta_{12d}$  de la pierna durante la tarea de seguimiento

## 6. Simulaciones

---

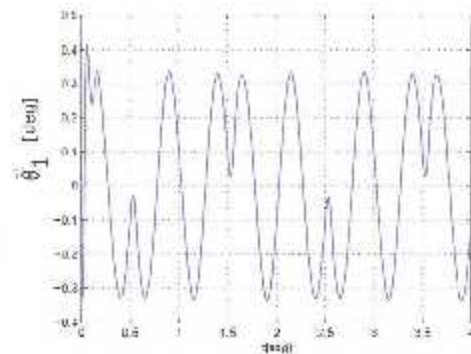


Figura 6.21: Error de posición articular durante la realización de la tarea de seguimiento  $\tilde{\theta}_1$

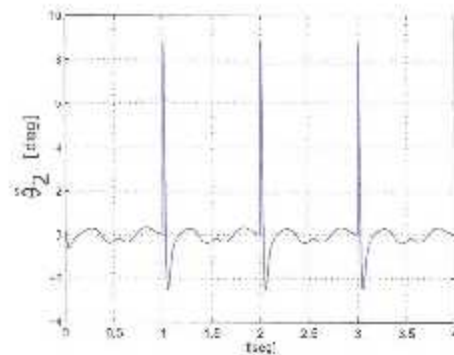


Figura 6.22: Error de posición articular durante la realización de la tarea de seguimiento  $\tilde{\theta}_2$

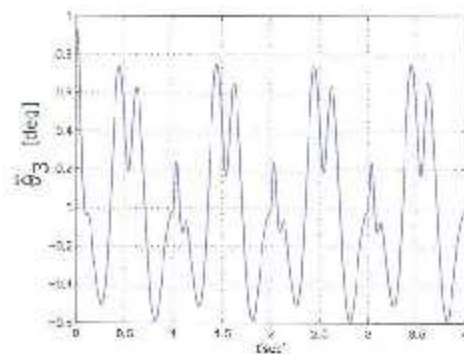


Figura 6.23: Error de posición articular durante la realización de la tarea de seguimiento  $\tilde{\theta}_3$

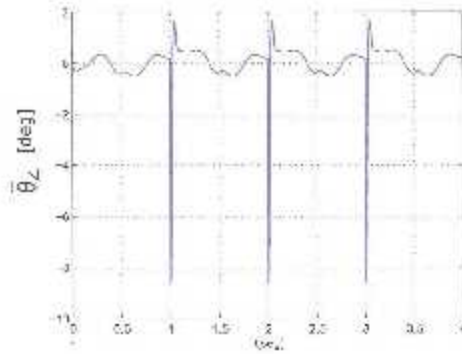


Figura 6.24: Error de posición articular durante la realización de la tarea de seguimiento  $\bar{\theta}_4$

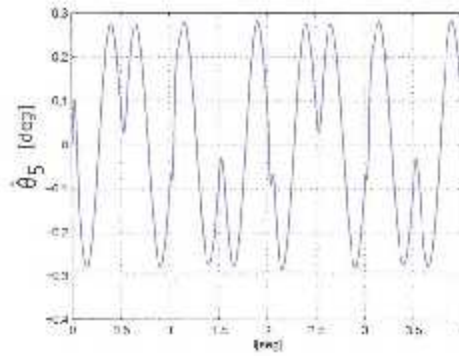


Figura 6.25: Error de posición articular durante la realización de la tarea de seguimiento  $\bar{\theta}_5$

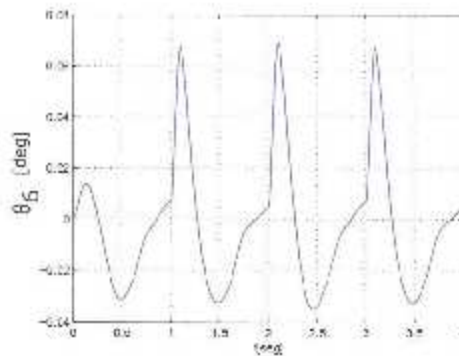


Figura 6.26: Error de posición articular durante la realización de la tarea de seguimiento  $\bar{\theta}_6$

## 6. Simulaciones

---

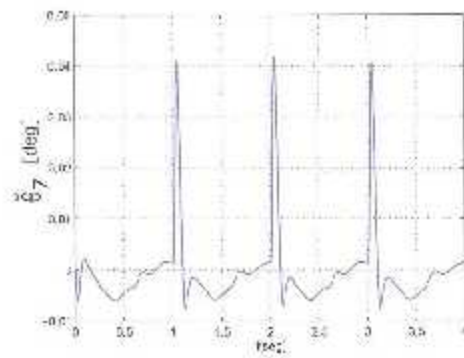


Figura 6.27: Error de posición articular durante la realización de la tarea de seguimiento  $\tilde{\theta}_7$

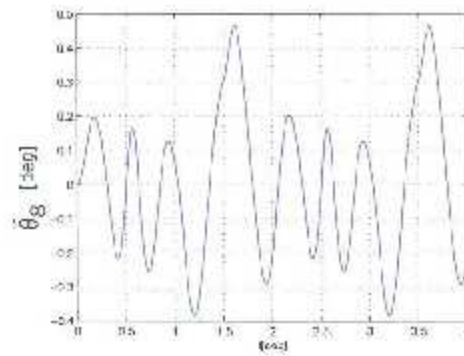


Figura 6.28: Error de posición articular durante la realización de la tarea de seguimiento  $\tilde{\theta}_8$

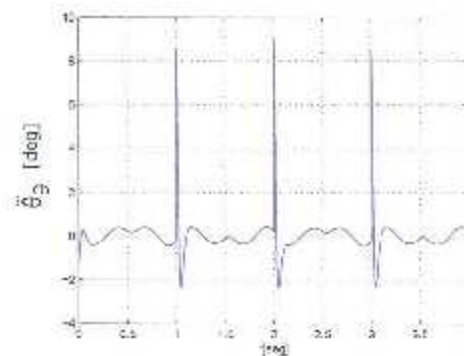


Figura 6.29: Error de posición articular durante la realización de la tarea de seguimiento  $\tilde{\theta}_9$



## 6. Simulaciones

---

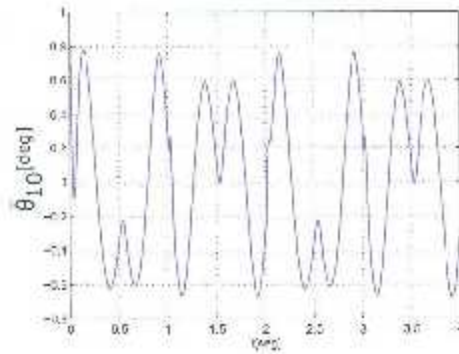


Figura 6.30: Error de posición articular durante la realización de la tarea de seguimiento  $\tilde{\theta}_{10}$

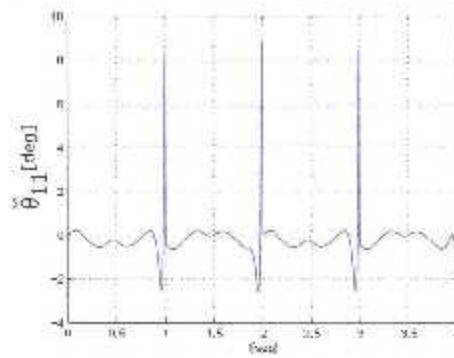


Figura 6.31: Error de posición articular durante la realización de la tarea de seguimiento  $\tilde{\theta}_{11}$

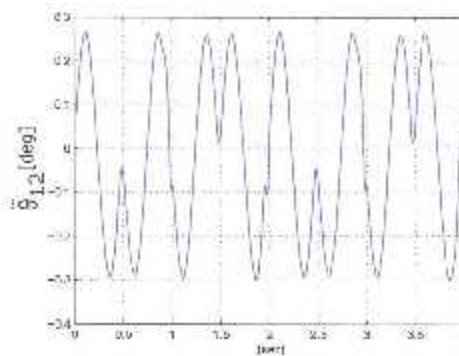


Figura 6.32: Error de posición articular durante la realización de la tarea de seguimiento  $\tilde{\theta}_{12}$

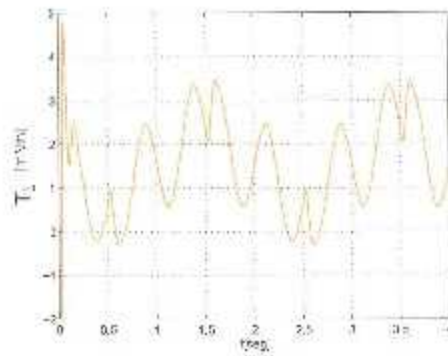


Figura 6.33: Par articular  $\tau_1$

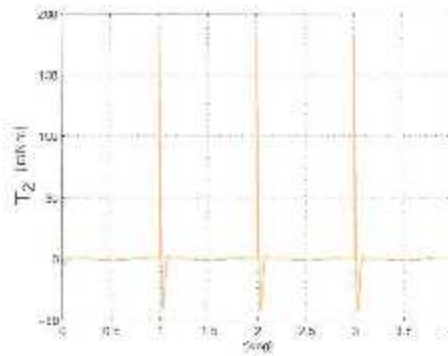


Figura 6.34: Par articular  $\tau_2$

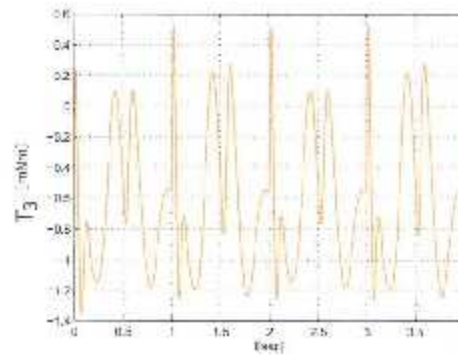


Figura 6.35: Par articular  $\tau_3$

6. Simulaciones

---

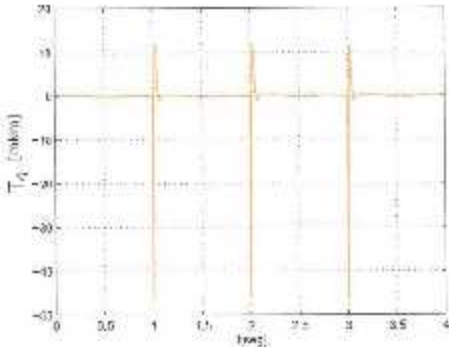


Figura 6.36: Par articular  $\tau_4$

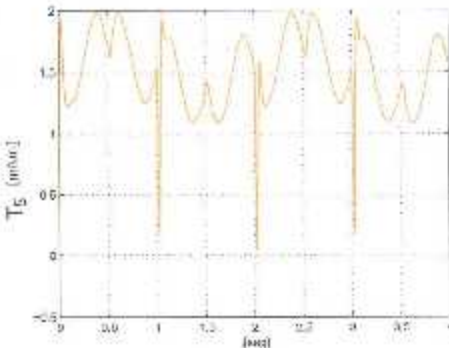


Figura 6.37: Par articular  $\tau_5$

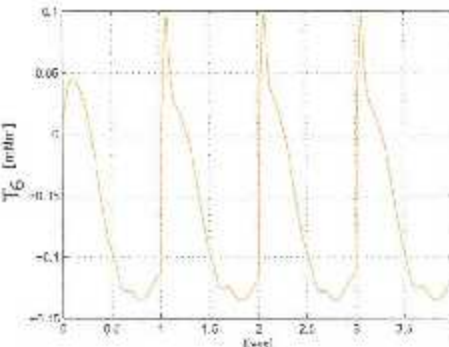


Figura 6.38: Par articular  $\tau_6$

## 6. Simulaciones

---

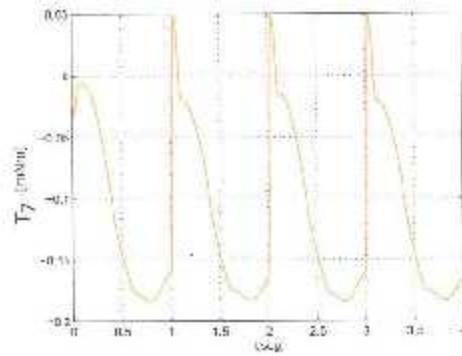


Figura 6.39: Par articular  $T_7$

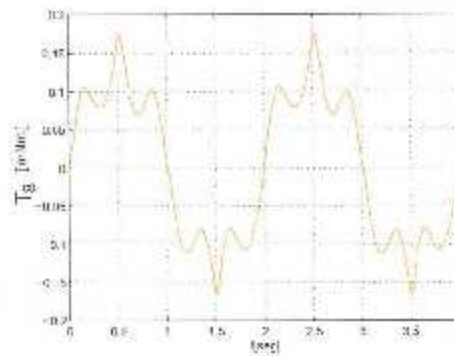


Figura 6.40: Par articular  $T_8$

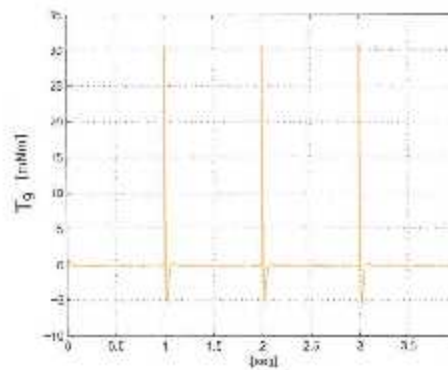


Figura 6.41: Par articular  $T_9$



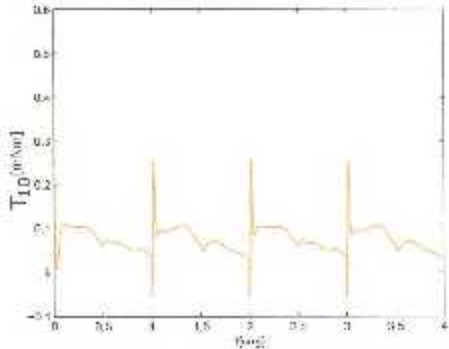


Figura 6.42: Par articular  $\tau_{10}$

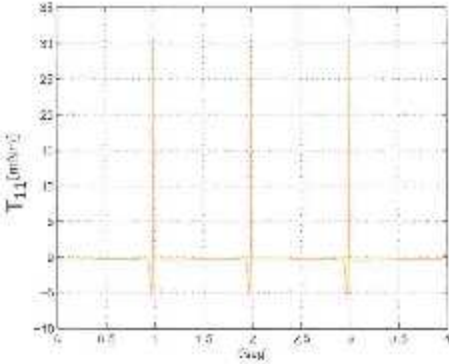


Figura 6.43: Par articular  $\tau_{11}$

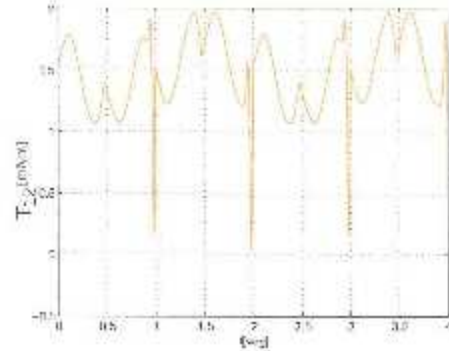


Figura 6.44: Par articular  $\tau_{12}$

## Capítulo 7

# Evaluación experimental

Mediante el programa Webots de NAO, se realizó una serie de experimentos del robot a fin de efectuar un análisis preliminar de la marcha basada en leyes de tipo cicloidal, considerando solamente la cinemática del humanoide. Después de realizar los experimentos en Webots, y con la certeza de que el patrón de marcha es factible, se procedió a realizar el experimento físicamente con el robot NAO. En este capítulo se describen los resultados obtenidos en tales experimentos. Para simplificar el protocolo de comunicación entre la computadora y el robot, se utilizó un programa Python que sirve como módulo de control para transmitir la información por el puerto ethernet del ordenador al robot NAO.

Los pasos que hay que seguir para conectar al robot NAO con nuestro ordenador se presentan a continuación.

1. Remover la cubierta ubicada detrás de la cabeza para tener acceso al puerto Ethernet.
2. Conectar a NAO el router por medio del cable Ethernet.
3. Encender a NAO.
4. Obtener la dirección IP del NAO presionando el botón del torso. Haciendo esto NAO dice su dirección IP.
5. Abrir un explorador de internet.
6. Escribir la dirección IP del NAO en la barra de dirección y presionar enter.
7. Se abrirá la página de NAO. Escribir el usuario y contraseña por default, es decir "nao" para ambos.
8. Listo, el robot NAO está conectado a la red por el puerto ethernet y ya podemos trabajar con dicho robot.

## 7. Evaluación experimental

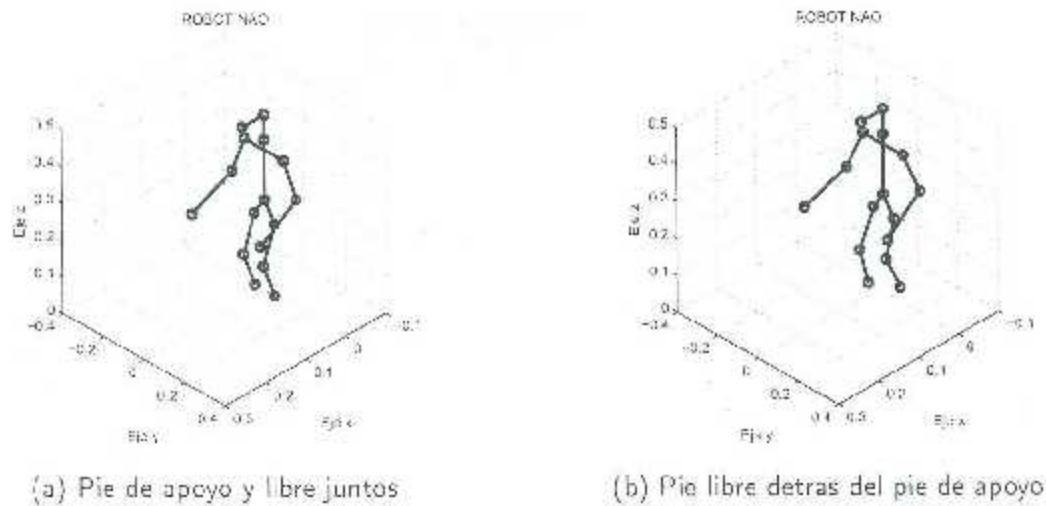


Figura 7.1: Posiciones iniciales del caminado

Para realizar el caminado del robot NAO se propone un patrón de marcha diferente al del capítulo 6, en este caso se propone que el robot inicie su caminado con los dos pies juntos es decir, pie de apoyo y pie libre juntos véase Figura 7.1a, en comparación con el capítulo 6 donde el robot iniciaba su caminado con el pie libre detrás del pie de apoyo como se muestra en la Figura 7.1b. Esto se hace con la finalidad de hacer un caminado más sencillo, los valores de los parámetros utilizados en el patrón de marcha se especifican en la Tabla 7.1.

Realizadas las simulaciones del patrón de marcha hecha en MATLAB véase Figura 7.2, y gracias a la cinemática inversa de posición, se obtienen las consignas de posición de las variables articulares de las piernas del robot NAO durante todo su caminado. Las consignas de posición se guardan en un archivo.txt para después ser exportadas al programa python. Dentro de python se realiza el programa que enviará las consignas de posición a las articulaciones de las piernas del robot NAO. A continuación se muestra paso a paso el programa realizado en python para poder realizar la marcha del robot NAO.

Tabla 7.1: Parámetros Del Patrón de marcha del robot NAO

Parámetros	Unidades	Magnitud
$n_s$	Pasos	4
$x_{cs}$	mm	0
$v_c$	mm/seg	1.5
$z_{cs}$	mm	245
$H_c$	mm	0
$y_{ca}$	mm	0
$J_c$	mm	= 50
$S$	mm	60
$H_p$	mm	20

## 7. Evaluación experimental

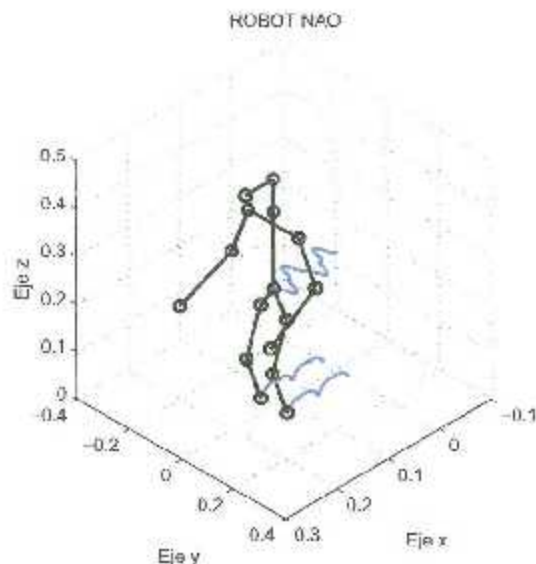


Figura 7.2: Patrón de marcha para el caminado del robot

Como se muestra en la Figura 7.3, lo primero que se hizo fue agregar las librerías necesarias para realizar la prueba, las cuales se muestran en la parte superior de la figura. Enseguida se define el puerto y el proxy, en este caso el proxy a utilizar es *ALMotion* el cual se utiliza para el control de las articulaciones.

Enseguida se define el primer argumento, en este caso *Nombre1*, el cual nos indica el nombre de las articulaciones que se desean mover, en este caso para realizar el primer paso hay que especificar las 12 articulaciones de las piernas, las cuales son, *LAnkleRoll*, *LAnklePitch*, *LKneePitch*, *LHipPitch*, *LHipRoll*, *LHipYawPitch*, *RHipYawPitch*, *RHipRoll*, *RHipPitch*, *RKneePitch*, *RAnklePitch* y *RAnkleRoll*, después se declara *ListaAngulos1*, en el cual van los valores de las consignas de posición en radianes de cada articulación durante todo el caminado. Debido a que el programa es muy extenso sólo se muestra el primer valor de cada articulación. Luego se declara *ListaTiempos1*, es decir la lista de tiempos en segundos para el traslado de cada articulación, en este caso para realizar un paso al robot le toma 20 segundos, el número de consignas en *ListaTiempo1* debe de ser de la misma dimensión que la de *ListaAngulos1*, y por último se declara el parámetro *Absolute1* el cual nos indica el valor booleano *true* para absoluto y *false* para relativo véase Figura 7.4.

Finalmente se llama a la función *motion.angleInterpolation*, la cual recibe los 4 argumentos anteriores y los ejecuta y se define la variable *name* la cual es una palabra reservada del modulo *sys*, el cual es propio de python para definir la función principal, en este caso se define la dirección IP del robot NAO véase Figura 7.5.

En las Figuras 7.6a - 7.17a se muestran las gráficas de las 12 posiciones articulares



## 7. Evaluación experimental

```
import sys
import time
import smath
from naoqi import ALProxy

def main(robotIP):
    PORT=9559
    naoqi =
        ALProxy("ALMotion", robotIP, PORT)
    except Exception, e:
        print "No se pudo conectar al robot"
```

Figura 7.3: Código en Python1

```
Nombre1=["CodoIzquierdo", "CodoDerecho", "CodoEsquina", "CodoEsquina", "CodoEsquina", "CodoEsquina"]
ListaAngulos1=[[-0.0701e+12,...], [-0.2730,...], [0.0090,...], [-0.2020,...], [-0.8761e+12,...], [0,...]]
ListaTiempos1=[0.25,...], [0.25,...], [0.25,...], [0.25,...], [0.25,...], [0.25,...]]
Absoluto1=True

Nombre2=["CodoIzquierdo", "CodoDerecho", "CodoEsquina", "CodoEsquina", "CodoEsquina", "CodoEsquina"]
ListaAngulos2=[[-0,...], [-0.8779e+12,...], [-0.2812,...], [0.0540,...], [-0.2020,...], [-0.8779e+12,...]]
ListaTiempos2=[0.25,...], [0.25,...], [0.25,...], [0.25,...], [0.25,...], [0.25,...]]
Absoluto2=True
```

Figura 7.4: Código en Python 2

deseadas de las piernas del robot NAO durante todo su caminado, de igual forma en las figuras 7.6b - 7.17b se muestran las 12 posiciones articulares reales durante todo el caminado del robot NAO; hay que señalar que dichas posiciones articulares reales se obtienen mediante un programa proporcionado por el fabricante del robot NAO llamado MONITOR, el cual nos permite monitoriar las posiciones articulares en tiempo real del robot, debido a que no es posible obtener los valores articulares del programa para poderlos exportar a MATLAB y poder realizar la diferencia entre la posición deseada y la posición real con el objetivo de obtener el error, sólo se muestran las gráficas de la posición deseadas del lado izquierdo y del lado derecho las posiciones reales durante el caminado del robot NAO.

Finalmente en la Figura 7.18 se muestra los resultados del caminado del robot NAO. Como se puede observar el robot inicia con los dos pies juntos, después con los valores de los parámetros especificados se logró hacer que el robot incline su cadera primero hacia la izquierda y luego hacia la derecha; esto se hizo con la finalidad de desplazar su centro de masa al polígono de soporte el cual se encuentra en el pie fijo (pie izquierdo para el primer

```
naoqi.engine.Interpolation(NombreL, ListaAngulosL, ListaTiemposL, AbsolutoL)

if __name__ == '__main__':
    robotIP="192.168.0.100"
    main(robotIP)
```

Figura 7.5: Código en Python 3

## 7. Evaluación experimental

paso, pie derecho para el segundo paso). Hecho esto el robot despega del piso el pie libre para poder dar el primer paso, después el robot posiciona de nuevo su cadera a la posición inicial para volver a inclinar su cadera pero ahora hacia la derecha con la finalidad de desplazar su centro de masa al pie fijo y así poder realizar el segundo paso; finalmente el robot regresa su cadera a su posición inicial y se vuelve a repetir la secuencia para  $n$  pasos.

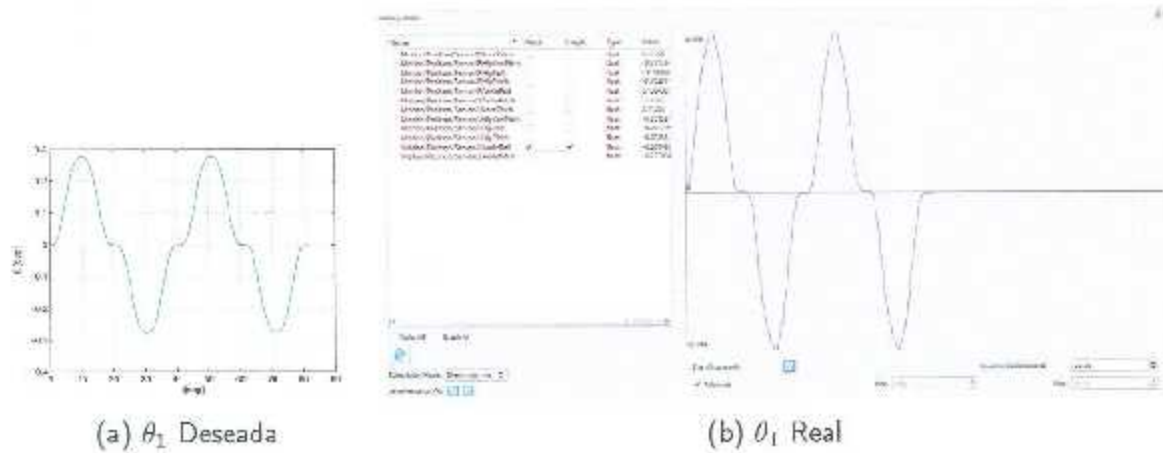


Figura 7.6: Posiciones articulares de  $\theta_1$



Figura 7.7: Posiciones articulares de  $\theta_2$

## 7. Evaluación experimental



Figura 7.8: Posiciones articulares de  $\theta_3$

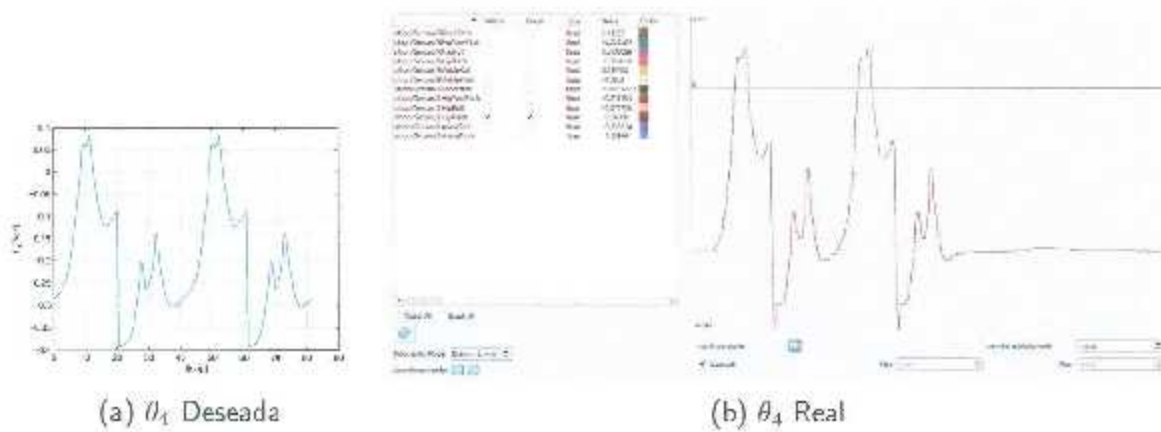


Figura 7.9: Posiciones articulares de  $\theta_4$



Figura 7.10: Posiciones articulares de  $\theta_5$





## 7. Evaluación experimental

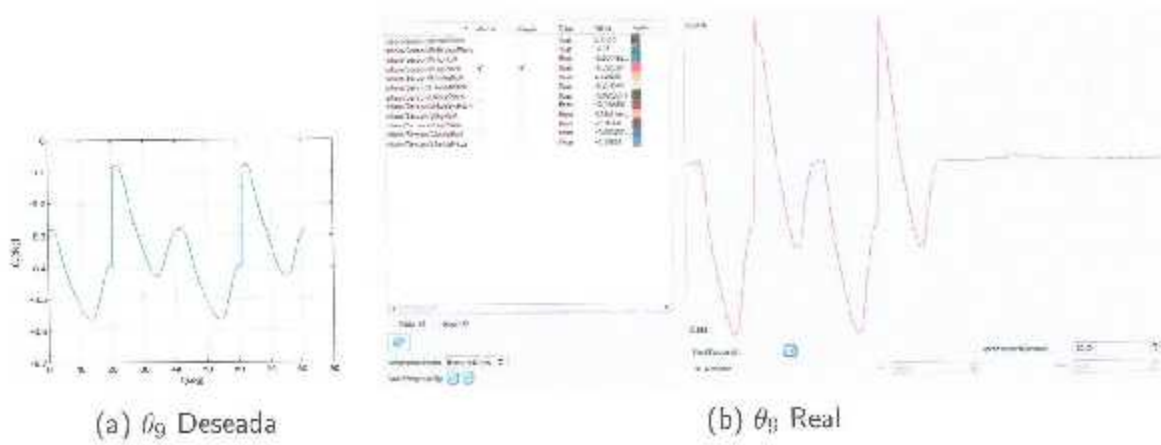


Figura 7.14: Posiciones articulares de  $\theta_9$

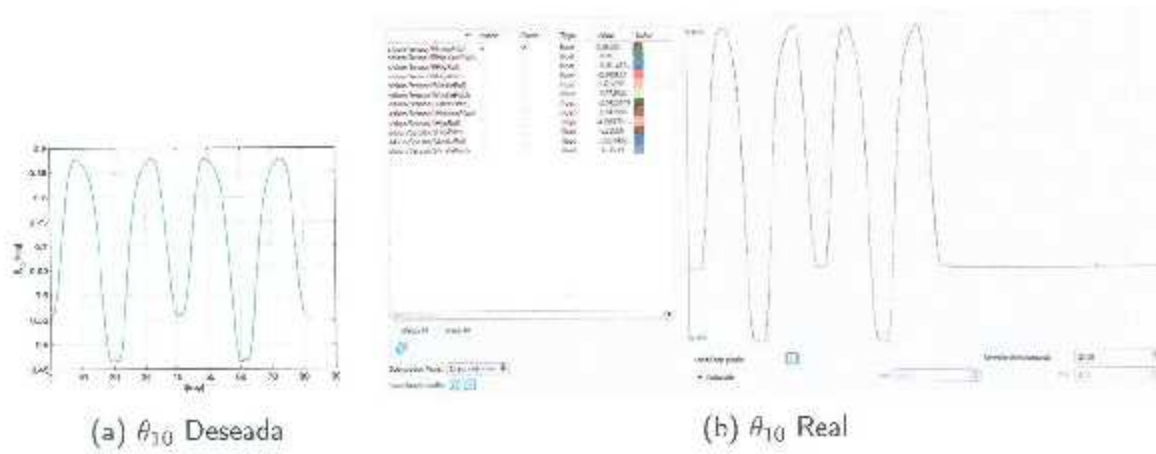


Figura 7.15: Posiciones articulares de  $\theta_{10}$

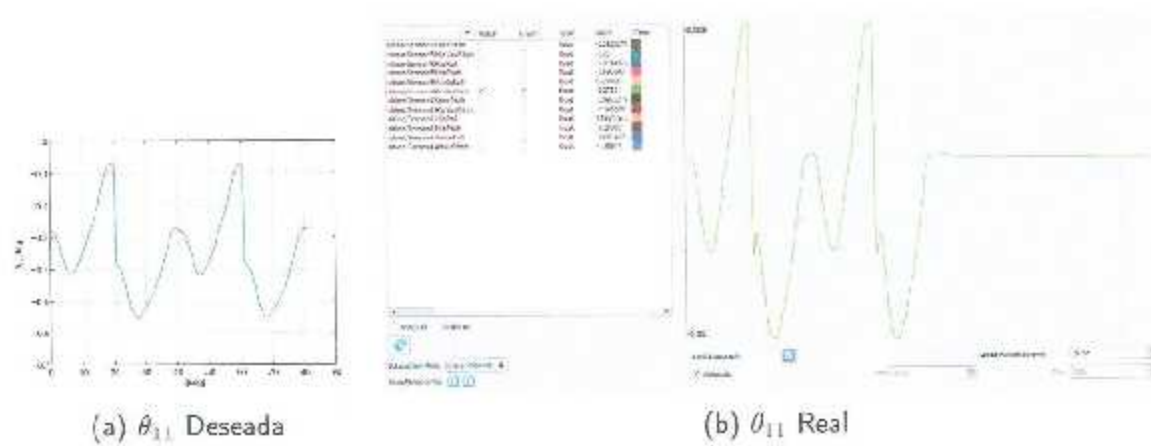


Figura 7.16: Posiciones articulares de  $\theta_{11}$

## 7. Evaluación experimental

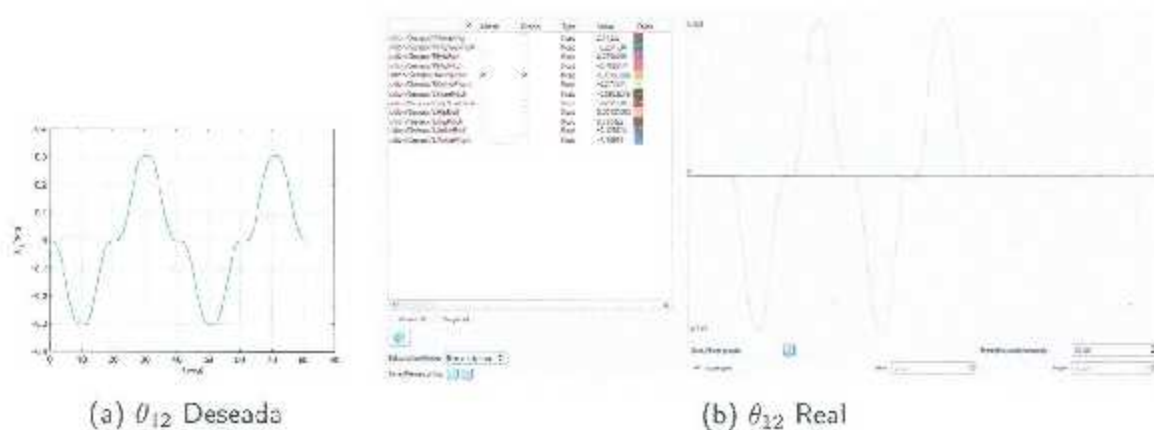


Figura 7.17: Posiciones articulares de  $\theta_{12}$ .

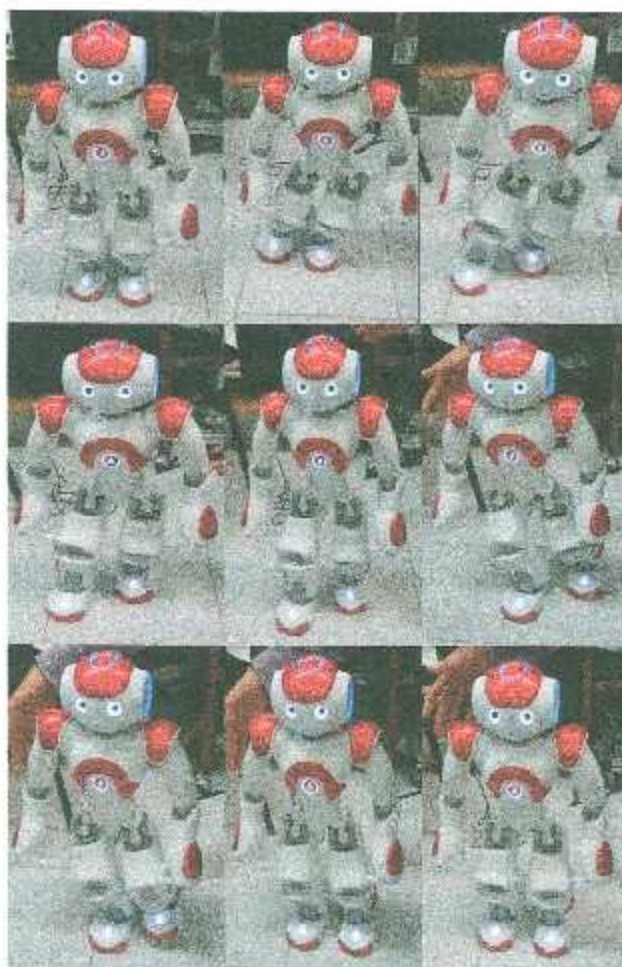


Figura 7.18: Secuencia de configuraciones del robot NAO durante la ejecución de dos pasos de la marcha con el patrón propuesto

## Capítulo 8

### Conclusión

Los robots humanoides han sido un punto de partida para el desarrollo tecnológico, los avances en su investigación son esenciales para incrementar las funciones de los robots de servicio, y además, su estudio es útil para comprender el sistema de la locomoción humana. Algunos problemas planteados dentro del caminado de robots humanoides son la estabilidad, la generación de ciclos de marcha y el diseño de controladores para generar movimientos en lazo cerrado. Conocer la historia de la robótica en humanoides permite entender su importancia, ayuda a asimilar sus alcances y a esclarecer la viabilidad de sus metas; además, es necesario para conocer los problemas que plantea y las maneras en las que se han abordado. El Capítulo 1 resultó vital para este trabajo, proporcionó algunos de los antecedentes históricos en robots humanoides, brindó una idea de los aportes y contribuciones de esta investigación a la robótica en humanoides y, además ayudó a plantear objetivos futuros. Al comparar la locomoción de un robot humanoide con la del ser humano, resulta increíble la naturalidad, coordinación, flexibilidad y versatilidad que el ser humano ha llegado a desarrollar, gracias a un proceso de evolución y adaptación que ha llevado millones de años y a un proceso de aprendizaje y experimentación que se lleva a cabo durante la infancia. En cambio, los robots humanoides surgieron hace algunas décadas y, por lo general, los algoritmos que utilizan para la generación de movimientos consideran a la marcha como un reflejo innato. Algunos métodos para la generación de patrones de marcha emplean demostraciones de ciclos de marcha realizados por humanos, y algunos controladores para la estabilidad del robot están inspirados en el funcionamiento del sistema locomotor humano. Las investigaciones en la marcha humana han facilitado el avance de la robótica en humanoides. Tener una idea de la complejidad de la marcha humana y de las actividades que realiza el sistema locomotor durante la marcha, ayuda a entender y a conocer las razones de las diferencias entre el desempeño de un robot humanoide y un humano, pero también brindar posibles respuestas para reducirlas.

En el capítulo 3 se obtuvieron las ecuaciones que representan el modelado cinemático directo e inverso de posición del robot NAO así como también el modelado de velocidad y aceleración. Para obtener el modelado cinemático inverso de posición de las piernas del robot se basó en un enfoque propuesto en [43] que a su vez, se inspira en el principio



## 8. Conclusión

---

del que parte el método de Paul. Para obtener el modelado cinemático inverso de posición de los brazos y la cabeza del robot se utilizó el método de Paul, y se comprobó con el programa SYMORO. Ambos modelos cinemáticos fueron fundamentales en el estudio realizado; el modelo inverso para la generación de las consignas articulares del robot, y el modelo directo para la visualización preliminar de los movimientos del robot mediante un modelo alambre realizado en el paquete MATLAB. En la referencia [53] se realiza un estudio completo del modelado cinemático directo e inverso de posición del robot NAO; sin embargo en el modelado inverso las consignas articulares es decir  $\theta_1, \theta_2, \dots, \theta_{12}$ , están en función de  $asen$  y  $acos$  por lo que al momento de hacer las simulaciones en MATLAB para poder realizar una trayectoria cicloidal dichas trayectorias no son seguidas adecuadamente. Además se resolvió el modelado dinámico de las piernas del robot mediante las ecuaciones de Euler-Lagrange, el cual se basó en un enfoque propuesto en [49] sin embargo este modelo obtiene el vector de posición de los centros de masa con respecto a un marco fijo que tiene una orientación idéntica a la del marco del mundo, esto es factible si el robot se encuentra en posición cero pero cuando se realiza un movimiento articular las posiciones de los centros de masa ya no son factibles, sin embargo para aplicar el método de forma correcta, es necesario encontrar el vector de posición local del centro de masa referido al marco  $\sum_{2i}$  más próximo unido a la articulación anterior más cercana, para esto es necesario realizar una rotación constante proporcionada por la ecuación (3.15), esto se puede ver con más detalle en la sección 3.5 del capítulo 3.

En el Capítulo 4 se propuso un patrón de marcha, el cual está basado en la propuesta [47] y [48]; sin embargo los parámetros que definen estos patrones de marcha, en el caso del robot NAO, han sido elegidos de tal forma que el robot pueda realizar el caminado. Se probó que si se especifica una marcha del robot NAO basada en movimientos traslacionales de la pelvis y del pie libre, el caminado del robot es factible bajo un criterio cinemático. Esta condición es necesaria para la realización de una caminata del NAO, mas no es suficiente. Es indispensable la realización de un segundo estudio que permita obtener los parámetros del movimiento del robot que verifiquen una marcha dinámicamente balanceada. En el Capítulo 5 se realizó un control de marcha en simulación mediante seguimiento de trayectorias, este controlador se llevó a cabo gracias al patrón de marcha, el modelado dinámico y las ecuaciones de la cinemática anteriormente descritos.

En el Capítulo 7 se hicieron pruebas experimentales con el robot NAO, en físico en las cuales se le dieron las consignas de posición obtenidas con el modelo cinemático inverso. Se obtuvieron las gráficas en MATLAB de las 12 posiciones articulares deseadas de las piernas del robot NAO así como también las 12 posiciones articulares reales obtenidas de los encoders del robot NAO; esto se hizo durante todo su caminado. La lectura de los encoders se logró gracias a que el fabricante del robot NAO proporciona un programa llamado MONITOR el cual nos permite monitoria la posición de cualquier articulación del robot NAO en tiempo real, la única desventaja de este programa es que no es posible extraer los valores articulares. Finalmente se muestra una imagen del caminado del robot NAO; cabe mencionar que durante el caminado el robot nunca sufrió ninguna caída, esto es debido



## 8. Conclusión

---

a que primero se hicieron pruebas en WEBOST y se fueron modificando los valores de los parámetros del patrón de marcha para lograr un caminado estable. Además como se sabe los patrones de marcha basados en movimientos cicloidalés evitan el impacto del pie oscilante durante el aterrizaje en cada paso, con lo que favorece el equilibrio dinámico del robot y se reduce el desgaste de los elementos mecánicos que componen sus piernas.

Como resultado de esta tesis, en colaboración con el Dr. Alfonso Pámanes, se generó un artículo [51], el cual fue presentado en el Congreso Mexicano de Robótica COMRob 2015.

- “Condiciones para una marcha elemental del robot NAO” Jesús E. Fierro, J. Alfonso Pámanes, Víctor A. Sanlibáñez, Gerardo Ruiz, Javier Ollervides. Memorias del Congreso Mexicano de Robótica (COMRob), 2015, Los Cabos, Baja California Sur, México.

# Bibliografía

- [1] S. SCHIALL *the new robotics.*, towards human - centered machines; pp. 115-126, 2007.
- [2] MORALES EDUARDO Y SUCAR SUCCAR ENRIQUE., *Los robots del futuro y su importancia para México.*, Komputer Sapiens; pp. 7-12, 2009
- [3] KATO T, TAKANISHI A, ISHIKAWA H, KATO I *The realization of dynamic walking by the biped walking robot WL-10RD*, Proc. of the 1985 Int. Conf. on Advanced Robotics ; pp. 459-466, Tokyo, Sep 9-10, 1985.
- [4] GEPPERT L; *Qrio, the robot that could*, IEEE Spectrum, 41(5), pp. 34-37, May 2004.
- [5] HIRAI K, HIROSE M, HAIKAWA Y, TAKENAKA T; *The Development of Honda Humanoid Robot.*, Proc. of the IEEE International Conference on Robotics and Automation, pp.1321-1326, 1998.
- [6] YAMASAKI F, MIYASHITA T, MATSUI T, GIJANO H; *PINO the humanoid*; Proc. of the Fourth International Workshop on RoboCup, Melbourne, Australia, August 2000.
- [7] ARACHI K, KANEKO K, KANEHIRA N, OTA S, MIYAMORI M, HIRATA M, KAJITA S, KANEHIRO K. *Development of humanoid robot HRP-3P*; Proc. of the 2004 IEEE International Conference on Humanoid Robots, Tsukuba, Japan.
- [8] SANDINI G., METTA G., VERON D. *Roboteub: An open framework for research in embodied cognition*; International Journal of Humanoid Robotics, 8 (2), November 2004.
- [9] BEIRA R., LOPES M., PRAÇA M., SANTOS - VICTOR J., BERNARDINO A., METTA G., BECCHI F., SALTARÉN R. *Design of the Robot-Cub (iCub) Head*; Proc. of the IEEE International Conference on Robotics and Automation, Orlando, May 2006.
- [10] RUESCH J, LOPES M, BERNARDINO A, HÖRNSTEIN J, SANTOS-VICTOR, PFEIFER R; *Multimodal Saliency- Based Bottom-Up Attention: A Framework for the Humanoid Robot iCub*; Proc. of IEEE International Conference on Robotics and Automation; Pasadena, CA, USA, May, 2008.
- [11] PAMANES JA, MURILLO OF, SANTOS-VICTOR J, ZEGHLOUL S; *Análisis de los Movimientos del Cuello y los Ojos del Robot iCub Basados en Velocidades Articulares de Mínima Norma*; 10 Congreso Mexicano de Robótica Septiembre 2008, México D.F.

- [12] FUKUDA T., KOMATA Y., ARAKAWA T.; *Stabilization Control of Biped Locomotion Robot based Learning with Gas having Self-adaptive Mutation and Recurrent Neural Networks*; Proc. of the IEEE International Conference on Robotics and Automation, pp. 217-221, 1997.
- [13] MCGEER T.; *Passive dynamic walking*; International Journal of Robotic Research, Vol. 9, No.6, pp-62-82, 1990.
- [14] ROSTAMI M., BESSONET G.; *Impactless Sagittal Gait of a Biped Robot During the Simple Support Phase*; Proc. of the IEEE International Conference on Robotics and Automation, pp. 1385-1391, 1998.
- [15] CHAVALLEREAU C., SARDIN P.; *Design and Actuation Optimization of a 4 axes Biped Robot for Walking and Running*; Proc. of the IEEE International Conference on Robotics and Automation, pp. 3365-3370, 2000.
- [16] VUKOBRATOVIC, M. AND JURICIC, D; *Contribution to the synthesis of biped gait*; In IEEE Trans. Bio-Medical Eng., Vol. 16, pp. 1-6, 1969.
- [17] NISHIWAKI K, KAGAMI S, KUNIYOSHI Y, INABA M AND INOUE H. *Online Generation of Humanoid Walking Motion based on a Fast Generation Method of Motion Pattern that Follows Desired ZMP*; Proc. of the 2002 IEEE/RSJ Int. Conference on Intelligent Robots and Systems, pp. 2684-2689, 2002.
- [18] PARK JH AND RHEE YK; *ZMP Trajectory Generation for Reduced Trunk Motions of Biped Robots*; Proc. of 1998 IEEE/RSJ International Conference on Intelligent Robots and Systems, Victoria, B.C., Canada, pp. 90-95, October 1998..
- [19] YAMAGUCHI J, SOGA E, INOUE S AND TAKANASHI A. *Development of a Bipedal Humanoid Robot - Control Method of Whole Body Cooperative Dynamic Biped Walking*; Proceedings of the 1999 IEEE International Conference of Robotics and Automation, Detroit Michigan, pp. 368-374, May 1999.
- [20] BRUNEAU O., OUESDOU F.B., WIEBER P.B.; *Dynamic transition simulation of a walking anthropomorphic robot*; Proc. of the IEEE International Conference on Robotics and Automation, 1998.
- [21] SILVA F., TENBEIRO J.A; *Controllability analysis of biped walking robots*; Proc. of the 2000 International Workshop on Advance Motion Control, pp. 595-600, 2000.
- [22] PARK J.H.; *Impedance control for biped robot locomotion*; IEEE Transactions on Robotics and Automation, Vol. 17, No. 6, pp.870-882, 2001.
- [23] MARCHESI S., MUSCATO G., VIRK G.S.; *Dynamically stable trajectory synthesis for a biped robot during the single support phase* ;Proc. of the IEEE/ASME International Conference on Advanced Intelligent Mechatronics, pp 953-958, 2001.



- [24] HUANG Q, KANEKO K, YOKOI K, KAJITA S, KOTOKU T, KOYACHI N, ARAI H, IMAMURA N, KOMORIYA K AND TANIE K.; *Balance Control of a Biped Robot Combining Off-line Pattern with Real time Modification*; Proc. of the 2000 IEEE International Conference on Robotics and Automation, pp. 3346-3352, 2000.
- [25] HUANG Q, YOKOI K, S. KAJITA, KANEKO K, ARAI H, KOYACHI N AND TANIE K.; *Planning Walking Patterns for a Biped Robot*; IEEE Transactions on Robotics and Automation, Vol. 10, No. 3, pp. 280-289, June 2001.
- [26] SHIH CHL, ZHU Y, GRUVER WA; *Optimization of the Biped Robot Trajectory.*; Proc. of the 1991 IEEE International Conference on Robotics and Automation (ICRA 1991), pp. 899-903, 1991.
- [27] LIM H, TAKANISHI A.; *Realization of Continuous Biped Walking*; Paper IEEE 0-7803-7087-2/0, pp 1630-1635, 2001.
- [28] PAMANES GJA, NUÑEZ NJV; *Análisis del Caminado de un Robot Bípedo Para un Patrón de Marcha Basado en Movimientos Cicloidales.*; Memorias del III Congreso Mexicano de Robótica de la AMRob, pp. 62-67, Querétaro, México, 2001.
- [29] ÁLVAREZ FE.; *Modelado Dinámico de un Robot Bípedo: Aplicación al Diseño de sus Mecanismos y al Análisis de la Marcha.*; Tesis para la obtención del grado de Maestro en Ciencias en Mecatrónica y Control, por el Instituto Tecnológico de la Laguna, Torreón, México, 2006.
- [30] MCGEER, T; *Passive dynamic walking*; tech. report, Simon Fraser University, Burnaby, British Columbia, Canada, 1988.
- [31] MCGEER, T; *Stability and control of two-dimensional bipedal walking*; tech. report, Simon Fraser University, Burnaby, British Columbia, Canada, 1988.
- [32] MCGEER, T; *Passive bipedal running*; tech. report, Simon Fraser University, Burnaby, British Columbia, Canada, 1989.
- [33] MCGEER, T; *Passive walking with knees*; Proc. 1990 IEEE Robotics Automation Conference, Cincinnati, OH, (1990), pp. 1640-1645.
- [34] COLLINS AL. *Efficient Bipedal Robots Based on Passive-Dynamic Walkers*; Science, 307 (2005), pp. 1082-1085.
- [35] J. GRIZZIE, G. ABBA, AND F. PLESTAN, *Asymptotically stable walking for biped robots: Analysis via systems with impulse effects*, IEEE Trans. Automatic Control, vol. 46, no. 1, pp. 51-64, January 2001.
- [36] E. WESTERVELT, J. GRIZZIE, AND D. KODITSCHER, *Hybrid zero dynamics of planar biped walkers*, IEEE Trans. Automatic Control, vol. 48, no. 1, pp. 42-56, 2003.



- [37] C. CHEVALLEREAU, G. ABBA, Y. Aoustin, F. R. W. F. PLESTAK, C. CANUDAS, AND J. GRIZZLE, *Rabbit: a testbed for advanced control theory*, IEEE Control Systems Magazine, vol. 23(5), pp. 57-79, October 2003.
- [38] HIRAI K, HIROSE M, HAIKAWA Y, TAKENAKA T. *The Development of Honda Humanoid Robot*, Proc. of the IEEE International Conference on Robotics and Automation, pp.1321-1326, 1998.
- [39] ARMADA M, CABALLERO R, AKINFIEV T, MONTES H, MANZANO C, PEDRAZA L, RÍOS S, GONZÁLEZ P. *Design of SILO2 humanoid robot*, Proc. of the International Workshop on Humanoid and Friendly Robots, 2002.
- [40] MOQUET S, WENGER F, PÁMANES GJA, MORENO H, FRANCO G. *Aspectos del Diseño Conceptual de un Robot Ambulante Bípedo de 4 Grados de Libertad*, Memorias del VI Congreso Mexicano de Robótica COMRob 2004, pp. 35-44, Torreón, México, 2004.
- [41] CISNEROS LR. *Estrategias de Modelado Cinemático y Simulación en Robots Humanoides*, Tesis para la obtención del grado de Maestro en Ciencias en Control Automático, por el Centro de Investigación y Estudios Avanzados del IPN, México DF, 2009.
- [42] KHALIL W, KLEIFINGER M *A new geometric notation for open and closed-loop robots*, Proc. of the IEEE International Conference on Robotics and Automation, pp. 1174-1180, San Francisco, 1986.
- [43] ARIAS, L., OIVERA L., PÁMANES, J., Y NUÑEZ, J. *Patrón de Marcha 3D de Tipo Cíclico para Humanoides y su Aplicación al robot Bioloid*.
- [44] PAUL, R. (1981) *Robot Manipulators: Mathematics, Programming, and Control*. Massachusetts Institute of Technology, 1981.
- [45] R. KELLY AND V. SANTIBANEZ, *Control de Movimiento de Robots Manipuladores*. Madrid: Prentice Hall, 2003.
- [46] JOHN, J.; CRAIG, *Introduction to Robotics: Mechanics and Control*, Pearson Prentice Hall, 3rd edn (2005). ISBN: 0-13-123629-6
- [47] PAMANES J A, NUÑEZ J V *Análisis del Caminado de un Robot Bípedo Para un Patrón de Marcha Basado en Movimientos Cíclicos*, Memorias del III Congreso Mexicano de Robotica de la AMRob; pp. 62-67, Querétaro, México, 2001.
- [48] ARIAS L E, PAMANES J A *Especificación de Movimientos Generales de la Pelvis y el Pie Libre de Robots Bípedos para Marcha sin Impacto*, Memorias del IX Congreso Iberoamericano de Ingeniería Mecánica (CIBIM9), pp. 10-141 a 10-148, ISBN 978-84-692-8516-9, Las Palmas de Gran Canaria, España: Noviembre de 2009.

- [49] E. HASHEMI AND M. GHAFFARI JADIDI *Dynamic Modeling and Control Study of the NAO Biped Robot with Improved Trajectory Planning*, Materials with Complex Behaviour II, Berlin Heidelberg 2012
- [50] MARK W. SPONG, SETH HUTCHINSON, AND M. VIDYASAGAR *Robot Modeling and Control*, First Edition John Wiley and Sons, INC.
- [51] JESÚS E. FIERRO, J. ALFONSO PÁMANES, VÍCTOR A. SANTIBÁÑEZ, GERARDO RUIZ, JAVIER OLLERVIDES *Condiciones para una marcha elemental del robot NAO*, Memorias del Congreso Mexicano de Robótica, Los Cabos, Baja California Sur, México 2015.
- [52] FIERRO J.E, J. PÁMANES J.A., ARIAS L.E; *Marcha del robot humanoide NAO basada en movimientos cicloidales*, Artículo A4-219, aceptado para su publicación en las Memorias del XXI Congreso Internacional anual de la SOMIV, Coatzacoalcos, México, Septiembre 23-25, 2015.
- [53] SAMUEL FERNANDEZ IGLESIAS, JOSEP M. FUERTES I ARMENGOL; *Locomoción bípeda del robot humanoide NAO*, Proyecto final de la carrera, Universitat Politècnica de Catalunya, 2009.
- [54] J.J. ALCARAZ JIMENEZ, D. HERRERO PEREZ II, H. MARTINEZ BARBERA; *Planning CoM Trajectories for an Omnidirectional Dynamic Gait in Humanoid Soccer Robots*, Faculta Informatica, Universidad de Murcia.

# Capítulo 9

## Anexos

A continuación se presenta el código utilizado en MATLAB para poder hacer las simulaciones del caminado del robot NAO. Este código está desarrollado en un script de MATLAB y genera las consignas articulares para que el humanoide pueda caminar.

Breve descripción del programa.

1. Declaración de variables. Se declaran dimensiones físicas y tiempos a utilizar.
2. Movimientos y coordenadas del robot. Ubicaciones espaciales del robot en sus fases.
3. Modelo directo e inverso. A partir de una posición deseada del órgano terminal se obtienen las consignas articulares.
4. Visualización del caminado del robot NAO.
5. Visualización de las 12 posiciones articulares de las piernas del robot durante todo su caminado.
6. Visualización de las velocidades y aceleraciones articulares durante cada paso que realiza el robot.

```

1  * PROGRAMA PARA LA GENERACION DEL CAMINATO DEL ROBOT NAO
2  * CREADO POR EL ING. GERARDO MANUEL ROJE LOPEZ:
3
4  clear all
5  clc
6
7  np=20; %numero de puntos
8  P=6; %fin del movimiento
9
10 %GENERACION DE PUNTOS DEL ROBOT NAO
11
12
13 d3 = 0.1029; % Longitud de la tibia.
14 d4 = 0.1; % Longitud del muslo.
15 d7 = 0.05; % Desplazamiento de la pierna en y.
16 d10 = d4; % Longitud del muslo.
17 d11 = d3; % Longitud de la tibia.
18 d77 = 0.0707106781; %Longitud de la articulación B en Z en la pierna.
19 d1 = 0;
20
21 L4 = 0.090; % Desplazamiento del hombro en x.
22 L5 = 0.135; % Desplazamiento del hombro en z.
23 L6 = 0.1137; % Desplazamiento de la mano en x : Longitud del brazo inferior.
24 d31 = 0.015; % Desplazamiento del codo en y.
25 r31 = 0.105; % Longitud del brazo superior.
26
27 L1 = 0.0539; % Longitud de la cadera en x.
28 L2 = 0.0679; % Longitud de la cadera en z.
29 L3 = 3.1615; % Longitud de la cadera en y.
30
31
32 %*****
33 % *****
34
35 % Para definir los angulos theta en grados del modelo directo de posición:
36
37 Q10=deg2rad(0);
38 Q20=deg2rad(0);
39
40 %*****
41
42 % *****
43
44 %Para definir los angulos theta en grados en el modelo directo de posición:
45
46 Q1D = deg2rad(30); %cadera (pi0).
47 Q2D = deg2rad(-17.2); %hombro (rho1).
48 Q3D = deg2rad(75.0); %codo (var1).
49 Q4D = deg2rad(37.9); %codo (rho2).
50 Q5D = deg2rad(0); %codo (yaw).
51
52
53 %*****

```



54  
55  
56  
57  
58  
59  
60  
61  
62  
63  
64  
65  
66  
67  
68  
69  
70  
71  
72  
73  
74  
75  
76  
77  
78  
79  
80  
81  
82  
83  
84  
85  
86  
87  
88  
89  
90  
91  
92  
93  
94  
95  
96  
97  
98  
99  
100  
101  
102  
103  
104  
105  
106

\* MANO IZQUIERDA.

? Para definir los ángulos theta en grados del modelo directo de posición.

Q1I = deg2rad(50); % ángulo (pitch).

Q2I = deg2rad(17.21); % ángulo (roll).

Q3I = deg2rad(-79.6); % ángulo (yaw).

Q4I = deg2rad(-37.9); % ángulo (roll).

Q5I = deg2rad(0); % ángulo (yaw).

\*\*\*\*\*

? Para graficar las posiciones, velocidades y aceleraciones si es true:

qDesPie = true; % Grafica las trayectorias de posición deseadas en X y Y del pie.

Acel Articulares = true; % Grafica las aceleraciones articulares (en cada paso).

Vel Articulares = true; % Grafica las velocidades articulares (en cada paso).

qPosContinuas = true; % Grafica las posiciones de cada eslabón durante todo el recorrido.

qPosContinuas2 = true; % Grafica las posiciones de cada eslabón durante todo el recorrido.

qDesCed = true; % Grafica las trayectorias de velocidad deseadas en X y Y de la cadera (para un ciclo de caminata) si es true.

qDesCed1 = true; % Grafica las trayectorias de aceleración deseadas en X y Y de la cadera (para un ciclo de caminata).

\*\*\*\*\*  
% Parametros para el brazo al momento del cambio del robot NAO

? Para el brazo derecho.

Ejini = deg2rad(-50); % Es el ángulo inicial de la articulación con respecto al robot.

Sbj = 0.25; % Es la amplitud de la oscilación de la articulación con respecto al robot.

Lb=-1; % Es 1 si la oscilación es hacia adelante y -1 si la oscilación es hacia atrás.

K=1; % Se hace igual a uno en el primer paso y en el último paso.

Ta = 2\*pi; % Es el período de oscilación del desplazamiento angular.

```

107                                     % Para el brazo izquierdo.
108 Bjin12 = deg2rad(-100); % Es el ángulo de inicial de la articulación hombro
109 (pi/2).
110 Sb12 = 0.25; % Es la amplitud de la variación de la articulación
111 hombro (total) a partir de Bjin1.
112 Eb12=1; % Es 1 si la cadera se hacia adelante y -1 si la
113 cadera se hacia atrás.
114 K2=1; % Se hace igual a uno en el primer paso y en el
115 último paso.
116
117 %-----
118
119 %Parámetros para el semicírculo del robot NAO
120
121
122 Xfijo = 0;
123 Trayectoria = 12;
124 nPasos = 4; % Numero de pasos que desea que haga el robot nao
125 permitirPausa = false;
126 switch Trayectoria
127     case 12
128         s = 0.06; % Longitud del paso en milímetros.
129         hp = 0.015; % Altura del pie máximo.
130         Zel = 0.245; % Posición lateral de la cadera en milímetros.
131         hc = 0; % Desplazamiento vertical máximo de la cadera.
132         Tl = 1;
133         Cadera = Tl;
134     end
135
136
137 Xc1 = -s/2; % Posición inicial del pie (lora) la cara del pie de apoyo
138 Xc2 = Xp + s/2; % Posición inicial de la cadera, a la mitad del robot
139 (tanto las 2 piernas)
140
141
142 Vc = s/(2*(Cadera)); % Velocidad de la cadera (2s entre 2 porque la cadera
143 recorre la mitad de la longitud del paso s)
144 t=1;
145
146 for j=1:nPasos
147     for i=0:np
148         t=i/np;
149
150         if mod(i,2)==1
151             Yc1 = 0; % Posición inicial de la cadera en y.
152             Yc2 = -0.04; % Desplazamiento horizontal de la cadera.
153         else
154             Yc1 = 0;
155             Yc2 = 0.04;
156         end
157
158
159 %-----
160

```

```

161
162 % Funciones auxiliares para el tiempo del vuelo al momento de terminar.
163
164 if t<T1
165     Bj = Bjini+2b*K*Sbj*((t/Ts)-(1/2*pi)*sin((2*t*pi)/Ts));
166 end
167     Bj = rad2deg(Bj);
168
169 if t<T1
170     Bj2 = Bjini2+Rb2*K2*Scj2*((t/Ts)-(1/2*pi)*sin((2*t*pi)/Ts));
171 end
172
173
174
175         %==== Movimiento de la cadera ====
176
177 switch trayectoria
178     case {1,2}
179         if t<T1
180             Xc = Xci + Vc*t;
181             if t< T1/2
182                 Zc = Zci - Hc*(2*t/T1 - (1/(2*pi))*sin(4*t*pi/T1));
183                 Yc = Yci - Ycf*(2*t/T1 - (1/(2*pi))*sin(4*t*pi/T1));
184             else
185                 Zc = Zci + Hc*(1-((2*t-T1)/T1 - (1/(2*pi))*sin(4*t*pi/T1)));
186                 Yc = Yci + Ycf*(1-((2*t-T1)/T1 - (1/(2*pi))*sin(4*t*pi/T1)));
187             end
188         else
189             Xc = Xci - S/2;
190             Zc = Zci;
191         end
192     otherwise
193         Xc = Xci+ Vc*t;
194         if t<=T/2
195             Zc = Zci + Hc*(2*t/T - (1/(2*pi))*sin(4*t*pi/T));
196             Yc = Yci + Ycf*(2*t/T - (1/(2*pi))*sin(4*t*pi/T));
197         else
198             Zc = Zci + Hc*(1-((2*t-T)/T - (1/(2*pi))*sin(4*t*pi/T)));
199             Yc = Yci + Ycf*(1-((2*t-T)/T - (1/(2*pi))*sin(4*t*pi/T)));
200         end
201     end
202
203
204
205         %==== Movimiento del pie libre ====
206
207
208     if t<=T1
209         Xpie = Xpiei + S*(t/T1-(1/(2*pi))*sin(2*pi*t/T1));
210     end
211
212     if t< T1/2
213         Zpie = Hp*(2*t/T1 - (1/(2*pi))*sin(4*t*pi/T1));
214     elseif t<=T1

```

```

215     Xpfe = Hp - Hp*((2*t-T1)/T1 - (1/(2*pi))*sin(4*t*pi/T1));
216 end
217
218
219 % Velocidades deseadas (definidas a la trayectoria definida en la ley de
220 movimiento)
221 % -----
222 % -----
223
224 switch Traectoria
225     case {1,2}
226         if t<=T1
227             Xcp = Vc;
228             if t<=T1/2
229                 Zcp = (2*Hc/T1)*(1-cos((4*pi*t)/T1));
230             else
231                 Zcp = -(2*Hc/T1)*(1-cos((4*pi*t)/T1));
232             end
233         else
234             Xcp = 0;
235             Zcp = 0;
236         end
237     otherwise
238         Xcp = Vc;
239         if t<=T/2
240             Zcp = (2*tc/T)*(1-cos((4*pi*t)/T));
241         else
242             Zcp = -(2*Hc/T)*(1-cos((4*pi*t)/T));
243         end
244     end
245
246
247
248 % ----- Pie libre -----
249 if t<=T1
250     Xpiep = (S/T1)*(1-cos(2*pi*t/T1));
251 end
252
253 switch Traectoria
254     case {1,4}
255         Xpiep = 0;
256     otherwise
257         Xpiep = ((Hp*pi)/S)*cos((Xpie/S+L/2)*pi)*Xpiep;
258     end
259
260
261
262
263 % Aceleraciones deseadas (definidas a la trayectoria definida en la ley de
264 movimiento)
265 % -----
266 % -----
267
268

```



```

269 switch Trayectoria
270     case {1,2}
271         if t<=T1
272             Xcopp = 0;
273             if t<=T1/2
274                 Zcopp = (8*pi*Hc/(T1^2))*sin((4*pi*t)/T1);
275             else
276                 Zcopp = -(8*pi*Hc/(T1^2))*sin((4*pi*t)/T1);
277             end
278         else
279             Xcopp = 0;
280             Zcopp = 0;
281         end
282     otherwise
283         Xcopp = 0;
284         if t<=T/2
285             Zcopp = (8*pi*Hc/(T^2))*sin((4*pi*t)/T);
286         else
287             Zcopp = -(8*pi*Hc/(T^2))*sin((4*pi*t)/T);
288         end
289     end
290
291     % ----- Die 3. und 4.
292     if t<=T
293         Xpiepp = (2*pi*S/(T^2))*sin(2*pi*t/T);
294     end
295
296     switch Trayectoria
297         case {1,4}
298             Zpiepp = 0;
299         otherwise
300             Zpiepp = (Hp*pi)/(S)*cos((Xpie/S+1/2)*pi)*Xpiepp =
301 Xpiepp*((Hp*pi*pi*Xpiep)/(S^2))*sin((Xpie/S+1/2)*pi);
302         end
303
304
305 %-----
306 % Modelado Lineal de posición de las barras, respecto de las 12 juntas de
307 % la pilaera (SOLUCO DE LAS 12 JUNTAS DE LA PILAERA Y DE LAS BARRAS LA
308 % EQUACION)
309 %-----
310
311 %Cálculo de theta 6 a theta9:
312
313 if rem(j,2)==1
314
315 %Posición: Desplaz de las Pilaeras 6 Anterior/Segu.
316
317 t11 = 1;
318 t21 = 0;
319 t31 = 0;
320
321 t12 = 0;
322 t22 = sqrt(.5);
323 t32 = -sqrt(.5);

```

```

324
325 t13 = 0;
326 t23 = sqrt(1.5);
327 t33 = sqrt(1.5);
328
329
330 xp = Xc;
331 yp = Yc;
332 zp = Zc;
333
334
335 %----- 1
336
337 a1 = d77*t23-yp-d7;
338 b1 = d77*t33-xp;
339 r1 = 1;
340
341
342 Q1 = atan2(r1*a1 , -(r1*b1));
343
344 theta1 = rad2deg(Q1);
345
346 %----- 2
347
348 a5 = t21*cos(Q1)+t31*sin(Q1);
349 b5 = -t22*cos(Q1)-t32*sin(Q1);
350 r5 = 1;
351
352 Q5 = atan2(r5*a5 , -(r5*b5));
353
354 theta5 = rad2deg(Q5);
355
356 %----- 3
357
358 S5 = (t22*cos(Q1)+t32*sin(Q1))*cos(Q5) + (t21*cos(Q1)+t31*sin(Q1))*sin(Q5);
359 C5 = -t23*cos(Q1)-t33*sin(Q1);
360
361 Q5 = atan2(S5 , C5)-(3*pi/4);
362
363 theta5 = rad2deg(Q5);
364
365 %----- 2 y 3
366
367 z14 = zp*cos(Q1) + d7*sin(Q1) - yp*sin(Q1) - d77*(t33*cos(Q1)-t23*sin(Q1));
368 z14 = zp-t13*d7;
369
370 %-----
371
372 r14 = sqrt((z14)^2+((x14)^2));
373 z14a = z14/r14;
374 x14a = x14/r14;
375

```

```

376 rfa = 63;
377 rfb = 44;
378
379 g = ((rta^2)-(rfa^2)+(r14^2))/(2*r14);
380 a = ((rta^2)-(rfa^2)+(r14^2))/(2*r14);
381 h = sqrt((rta^2)-(g^2));
382
383 E11 = 1;
384 rtax = abs(E11*h*x14a+g*x14a);
385 rtay = abs(g*x14a-E11*h*x14a);
386
387 rfax = r14-rtax;
388 rfay = r14-rtay;
389
390 Qtach = atan2(rtay, rtax);
391 Qfach = atan2(rfay, rfax);
392
393 Q2 = Qtach-pi/2;
394
395 theta2 = rad2deg(Q2);
396
397 Q3 = Qfach-Qtach;
398
399 theta3 = rad2deg(Q3);
400
401 %PART 4
402
403 S234 = (-t12*cos(Q6)-t11*sin(Q6))*cos((3*pi)/4)+Q5-t13*sin((3*pi)/4)+Q6;
404 C234 = t11*cos(Q6)-t12*sin(Q6);
405
406 Q234 = atan2(S234, C234);
407
408 Q4 = Q234-Q2-Q3;
409
410 theta4 = rad2deg(Q4);
411
412
413 % Positions Desees de la 7 e la 12 articulation
414
415
416 sfpz = 0;
417 sfpz = sqrt(.5);
418 sfpz = -sqrt(.5);
419
420 afpz = 0;
421 afpz = sqrt(.5);
422 afpz = sqrt(.5);
423
424 afpx = 1;
425 afpy = 0;
426 afpz = 0;
427
428 pfpx = Xp1e-Xc1;

```

```

429 p1py = sqrt(.5)*Zc-sqrt(.5)*Yc-sqrt(.5)*Zpie+(25/1000)*2^(1/2);
430 p1pz = sqrt(.5)*Zpie-sqrt(.5)*Zc-sqrt(.5)*Yc+(25/1000)*2^(1/2);
431
432
433
434 %TETA 3
435
436 a7 = d77*atpz+afcy*p1pz-p1py*afcz;
437 b7 = d77*afpx+afcy*r1px-p1py*afpx;
438 E7 = 1;
439
440 Q7 = atan2(E7*a7 , -(E7*b7));
441
442 teta7 = rad2deg(Q7);
443 %TETA 3
444
445 a8 = d77-p1py;
446 b8 = p1pz*cos(Q7)-p1px*sin(Q7);
447 E8 = -1;
448
449 Q8 = -atan2(E8*a8 , -(E8*b8))+((3*pi)/4);
450
451 cheta8 = rad2deg(Q8);
452
453 % TETA 3 * 30 DE; Q8123
454
455 r812x = p1py*sin((3*pi)/4)-Q8)-177*sin((3*pi)/4)-Q8)-p1pz*cos(Q7)*cos((3*pi)/4)-
456 Q8)+p1px*sin(Q7)*cos((3*pi)/4)-Q8);
457 r812z = p1px*cos(Q7)-p1pz*sin(Q7);
458
459 r812 = sqrt((r812x^2)+(r812z^2));
460
461 r812ax = r812x/r812;
462
463 r812az = r812z/r812;
464
465 r11 = d10;
466 r11 = d11;
467
468 gg = ((r11^2)-(r11^2)+(r812^2))/(2*r812);
469 mg = ((r11^2)-(r11^2)+(r812^2))/(2*r812);
470
471 E9 = sqrt((d10^2)-(gg^2));
472
473 E9 = 1;
474
475 r11x = cos(E9*E9*r812ax-cd*r812az);
476 r11y = cos(gg*r812ax-E9*E9*r812az);
477 r11x = r812x-r11x;
478 r11y = r812x-r11y;
479
480 Q11ch = atan2(r11y , r11x);
481 Q11ch = atan2(r11y , r11x);

```



```

482
483 Q9 = Qf1ch-pi/2;
484 Q10 = Qf1ch-Qf1c^2;
485
486 theca9 = rad2deg(Q9);
487 theca10 = rad2deg(Q10);
488
489 ***** 12
490
491 S12 = a1px*cos(Q7)*sin((3*pi/4)-Q9)-a1py*cos((3*pi/4)-
492 Q9)+a1px*sin(Q7)*sin((3*pi/4)-Q9);
493 C12 = n1pz*cos(Q7)*sin((3*pi/4)-Q9)-n1py*cos((3*pi/4)-
494 Q9)+n1px*sin(Q7)*sin((3*pi/4)-Q9);
495
496 Q12 = atan2(S12 , C12);
497 thets12 = rad2deg(Q12);
498 ***** 11
499
500 a11 = cos(Q9-Q10);
501 b11 = -sin(Q9-Q10);
502 c11 = a1px*cos(Q7)-a1py*sin(Q7);
503
504 E11 = 1;
505
506 S11 = b11*a11-E11*c11*sqrt(abs((a11^2)+(b11^2)-(c11^2)));
507
508 C11 = a11*c11-E11*b11*sqrt(abs((a11^2)+(b11^2)-(c11^2)));
509
510
511 Q11 = atan2(S11 , C11);
512
513 thes11 = rad2deg(Q11);
514
515 else
516
517 ~~~~~~
518 * Montaña inverso de posición de las personas, cálculo de las 12 direcciones de
519 * la iglesia CUANDO EL NIE SE MOVO EN LA IZQUIERDA Y EL PIE LIBRE LA
520 * DERECHA)
521 ~~~~~~
522
523 * Posición pasada de las Primeras 6 Asociaciones
524
525 t11 = 1;
526 t21 = 0;
527 t31 = 0;
528
529 t12 = 0;
530 t22 = sqrt(.5);
531 t32 = -sqrt(.5);
532
533 t13 = 0;
534 t23 = sqrt(.5);

```

```

535 t33 = sqrt(.3);
536
537
538 xp = Xo;
539 yp = Yo;
540 zp = Zo;
541
542 % xf = 50;
543 % yf = -50;
544 % zf = 0;
545
546 xtt = 60;
547 ytt = -50;
548 ztt = 0;
549
550 %----- 1
551
552 a1 = d/yp-d/1*t32;
553 b1 = -d/1*t32-zp;
554 E1 = 1;
555
556
557 Q1 = atan2(E1*a1 , -(E1*b1));
558
559 theta1 = rad2deg(Q1);
560
561 %----- 2
562
563 a6 = t21*cos(Q1)+t31*sin(Q1);
564 b6 = -t22*cos(Q1)-t32*sin(Q1);
565 E6 = 1;
566
567 Q6 = atan2(E6*a6 , -(E6*b6));
568
569 theta6 = rad2deg(Q6);
570
571 %----- 3
572
573 S5 = (t23*cos(Q1)-t32*sin(Q1))*cos(Q6)+(t21*cos(Q1)-t31*sin(Q1))*sin(Q6);
574 C5 = -t23*cos(Q1)-t33*sin(Q1);
575
576 Q5 = atan2(S5 , C5)-(3*pi/4);
577
578 theta5 = rad2deg(Q5);
579
580 %----- 2 X 3
581
582 x14 = d/1*sin(Q1)+zp*cos(Q1)-yp*sin(Q1)+d/1*(t32*cos(Q1)-t22*sin(Q1));
583 z14 = xp+(t12*d/1);
584
585 %----- 2 X 3
586
587 r14 = sqrt((z14^2)-(x14^2));

```

```

588 x14a = z14/r14;
589 x14b = x14/z14;
590
591 rta = d3;
592 rfa = d4;
593
594 g = ((rta^2)-(rfa^2)-(-14^2))/(2*z14);
595 m = ((rfa^2)-(z14^2)+(z14^2))/(2*z14);
596 h = sqrt((rta^2)-(g^2));
597
598 E11 = 1;
599 rtax = abs(E11*h*x14a+g*z14a);
600 rtay = abs(g*x14a-E11*m*z14a);
601
602 r1ax = x14-rtax;
603 r1ay = x14-rtay;
604
605 Q1ach = atan2(r1ay, r1ax);
606 Q1ach = atan2(r1ay, r1ax);
607
608 Q2 = Q1ach-pi/2;
609
610 theta2 = rad2deg(Q2);
611
612 Q3 = Q1ach-Q1ach;
613
614 theta3 = rad2deg(Q3);
615
616 theta3;
617
618 S234 = (-t12*cos(Q6)-t11*sin(Q6))*cos((3*pi/7)+Q5)-t13*sin((3*pi/4)+Q5);
619 C234 = t11*cos(Q6)-t12*sin(Q6);
620
621 Q234 = atan2(S234, C234);
622
623 Q4 = Q234-Q2-Q3;
624
625 theta4 = rad2deg(Q4);
626
627
628
629 % Variables Desconocidas De La 1 a La 13 Articulacion
630
631 a1px = 0;
632 a1py = sqrt(.5);
633 a1pz = -sqrt(.5);
634
635 a1fx = 0;
636 a1fy = sqrt(.5);
637 a1fz = sqrt(.5);
638
639 a1px = 1;
640 a1py = 0;

```

```

641 a1pz = 0;
642
643 p1pz = X01e-X0;
644 p1py = sqrt(.5)*Z0-sqrt(.5)*Y0-sqrt(.5)*Zpie-(25/1000)*2^(1/2);
645 p1pz = sqrt(.5)*3pie-sqrt(.5)*Z0-sqrt(.5)*Y0-(25/1000)*2^(1/2);
646
647
648 *****
649
650 a7 = -p1py*a1pz+a1py*d77+p1pz*a1py;
651 b7 = -p1py*a1px+a1py*p1px;
652 E7 = 1;
653
654 Q7 = atan2(E7*a7 , -(E7*b7));
655
656 theta7 = rad2deg(Q7);
657 *****
658
659 a8 = -p1py;
660 b8 = d77*cos(Q7)+p1pz*cos(Q7)+p1px*sin(Q7);
661 E8 = -1;
662
663 Q8 = -atan2(E8*a8 , -(E8*b8))-(3*pi/4);
664
665 theta8 = rad2deg(Q8);
666
667 *****
668
669 r812x = p1py*sin((3*pi/4)-Q8)+p1pz*cos(Q7)*cos((3*pi/4)-
670 Q8)+p1px*sin(Q7)*cos((3*pi/4)-Q8)+d77*cos((3*pi/4)-Q8)*cos(Q7);
671 r812z = p1px*cos(Q7)-p1pz*sin(Q7)-d77*sin(Q7);
672
673 r812 = sqrt((r812x^2)-(r812z^2));
674
675 r812ax = r812x/r812;
676
677 r812az = r812z/r812;
678
679 rE1 = d10;
680 rL1 = d11;
681
682 gg = ((rE1^2)-(rL1^2)+(r812^2))/(2*r812);
683 mg = ((rE1^2)-(rE1^2)+(r812^2))/(2*r812);
684
685 hq = sqrt((d10^2)-(gg^2));
686
687 Eg = 1;
688
689 r11x = abs(Eg*hq*r812ax-gg*r812ax);
690 r11y = abs(gg*r812ax-Eg*hq*r812ax);
691 r11x = r812x-r11x;
692 r11y = r812x-r11y;
693

```



```

694 Qf1e8 = atan2(rf1y , rf1x);
695 Ql1e8 = atan2(rt1y , rt1x);
696
697 Q9 = Qf1e8-pi/2;
698 Q10 = Ql1e8-Qf1e8;
699
700 theta9 = rad2deg(Q9);
701 theta10 = rad2deg(Q10);
702
703 ***** 12
704
705 S12 = s1px*cos(Q7)*sin((3*pi/4)-Q8)-s1py*cos((3*pi/4)-
706 Q8)-s1px*sin(Q7)*sin((3*pi/4)-Q8);
707 C12 = n1px*cos(Q7)*sin((3*pi/4)-Q8)-n1py*cos((3*pi/4)-
708 Q8)+n1px*sin(Q7)*sin((3*pi/4)-Q8);
709
710 Q12 = atan2(S12 , C12);
711 theta12 = rad2deg(Q12);
712 ***** 13
713
714 a11 = cos(Q9+Q10);
715 b11 = -sin(Q9+Q10);
716 c11 = a1px*cos(Q7)-a1py*sin(Q7);
717
718 b11 = 1;
719
720 S11 = b11*c11-b11*a11*sqrt(abs((a11^2)+(b11^2)-(c11^2)));
721
722 C11 = a11*c11-b11*b11*sqrt(abs((a11^2)+(b11^2)-(c11^2)));
723
724
725 Q11 = atan2(S11 , C11);
726
727 theta11 = rad2deg(Q11);
728
729 end
730
731
732
733 .....
734 .....
735
736 *****
737
738 J11 = -(d3*cos(Q1)*cos(Q2) - d4*cos(Q1)*cos(Q2 + Q3);
739
740 J21 = -(d3*cos(Q2)*sin(Q1) - d4*cos(Q2 + Q3)*sin(Q1);
741
742 J31 = 0;
743
744 J41 = 0;
745
746 J51 = 0;

```

```

747
748 J61 = 1;
749
750 J12 = d3*sin(Q1)*sin(Q2) - d4*cos(Q1)*sin(Q2 + Q3);
751
752 J22 = -(d3*cos(Q1)*sin(Q2)) - d4*cos(Q1)*sin(Q2 + Q3);
753
754 J32 = -(d3*cos(Q2)) - d4*cos(Q2 + Q3);
755
756 J42 = -cos(Q1);
757
758 J52 = -sin(Q1);
759
760 J62 = 0;
761
762 J13 = d4*sin(Q1)*sin(Q2 + Q3);
763
764 J23 = -(d4*cos(Q1)*sin(Q2 + Q3));
765
766 J33 = -(d4*cos(Q2 + Q3));
767
768 J43 = -cos(Q1);
769
770 J53 = -sin(Q1);
771
772 J63 = 0;
773
774 J14 = 0;
775
776 J24 = 0;
777
778 J34 = 0;
779
780 J44 = -cos(Q1);
781
782 J54 = -sin(Q1);
783
784 J64 = 0;
785
786 J15 = 0;
787
788 J25 = 0;
789
790 J35 = 0;
791
792 J45 = sin(Q1)*sin(Q2 + Q3 + Q4);
793
794 J55 = -(cos(Q1)*sin(Q2 + Q3 + Q4));
795
796 J65 = -cos(Q2 + Q3 + Q4);
797
798 J16 = 0;

```

```

799
800 J26 = 0;
801
802 J36 = 0;
803
804 J46 = -(cos(Q1)*cos(((3*pi)/4) + Q5)) - cos(Q2 + Q3 -
805 Q4)*sin(Q1)*sin(((3*pi)/4) - Q5);
806
807 J56 = -(cos(((3*pi)/4) + Q5)*sin(Q1)) + cos(Q1)*cos(Q2 + Q3 +
808 Q4)*sin(((3*pi)/4) + Q5);
809
810 J66 = -(sin(Q2 + Q3 + Q4)*sin(((3*pi)/4) + Q5));
811
812
813 V06 = [Xop; 0; Zop; 0; 0; 0];
814
815
816 J = [ J11 J12 J13 J14 J15 J16;
817       J21 J22 J23 J24 J25 J26;
818       J31 J32 J33 J34 J35 J36;
819       J41 J42 J43 J44 J45 J46;
820       J51 J52 J53 J54 J55 J56;
821       J61 J62 J63 J64 J65 J66];
822
823
824 qp = inv(J)*V06;
825 Q1P=qp(1);
826 Q2P=qp(2);
827 Q3P=qp(3);
828 Q4P=qp(4);
829 Q5P=qp(5);
830 Q6P=qp(6);
831
832 % -----
833 % Modelo de Aceleracion del Pie de Tropa a la Pivote del robot IAD
834 % -----
835
836 J11P = d3*cos(Q2)*sin(Q1)*(Q1P)+d3*cos(Q1)*sin(Q2)*(Q2P)-d4*cos(Q2 +
837 Q3)*sin(Q1)*(Q1P)-d4*sin(Q2 + Q3)*cos(Q1)*(Q2P-Q3P);
838 J21P = -d3*cos(Q1)*cos(Q2)*(Q1P)-d3*sin(Q1)*sin(Q2)*(Q2P)-d4*cos(Q2 +
839 Q3)*cos(Q1)*(Q1P)-d4*sin(Q2 + Q3)*sin(Q1)*(Q2P-Q3P);
840 J31P = 0;
841 J41P = 0;
842 J51P = 0;
843 J61P = 0;
844 J12P = d3*cos(Q1)*sin(Q2)*(Q1P)+d3*cos(Q2)*sin(Q1)*(Q2P)+d4*sin(Q2 +
845 Q3)*cos(Q1)*(Q1P)+d4*cos(Q2 - Q3)*sin(Q1)*(Q2P+Q3P);
846 J22P = d3*sin(Q1)*sin(Q2)*(Q1P)-d3*cos(Q1)*cos(Q2)*(Q2P)+d4*sin(Q2 +
847 Q3)*sin(Q1)*(Q1P)-d4*cos(Q2 + Q3)*cos(Q1)*(Q2P+Q3P);
848 J32P = d3*sin(Q2)*(Q2P)-d4*cos(Q2 - Q3)*(Q2P-Q3P);
849 J42P = sin(Q1)*(Q1P);
850 J52P = -cos(Q1)*(Q1P);
851 J62P = 0;
852 J13P = d4*sin(Q2 + Q3)*cos(Q1)*(Q1P)+d4*cos(Q2 + Q3)*sin(Q1)*(Q2P+Q3P);

```

```

853 J23P = s4*sin(Q2 + Q3)*sin(Q1)*(Q1P)-s4*cos(Q2 + Q3)*cos(Q1)*(Q2P+Q3P);
854 J33P = s4*sin(Q2 - Q3)*(Q2P-Q3P);
855 J43P = sin(Q1)*(Q1P);
856 J53P = -cos(Q1)*(Q1P);
857 J63P = 0;
858 J14P = 0;
859 J24P = 0;
860 J34P = 0;
861 J44P = sin(Q1)*(Q1P);
862 J54P = -cos(Q1)*(Q1P);
863 J64P = 0;
864 J15P = 0;
865 J25P = 0;
866 J35P = 0;
867 J45P = cos(Q1)*sin(Q2 - Q3 + Q4)*(Q1P)-cos(Q2 + Q3 +
868 Q4)*sin(Q1)*(Q2P-Q3P+Q4P);
869 J55P = sin(Q1)*sin(Q2 + Q3 + Q4)*(Q1P)-cos(Q1)*cos(Q2 + Q3 +
870 Q4)*(Q2P+Q3P+Q4P);
871 J65P = sin(Q2 + Q3 + Q4)*(Q2P-Q3P+Q4P);
872 J16P = 0;
873 J26P = 0;
874 J36P = 0;
875 J46P = cos((3*pi)/4 - Q5)*sin(Q1)*(Q1P)-cos(Q1)*sin((3*pi)/4 + Q5)*(Q5P)-
876 cos(Q1)*cos(Q2 + Q3 + Q4)*sin((3*pi)/4 + Q5)*(Q1P)-cos(Q2 + Q3 +
877 Q4)*cos((3*pi)/4 + Q5)*sin(Q1)*(Q5P)+sin(Q1)*sin(Q2 + Q3 + Q4)*sin((3*pi)/4 +
878 Q5)*(Q2P-Q3P+Q4P);
879 J56P = -cos(Q1)*cos((3*pi)/4 + Q5)*(Q1P)+sin(Q1)*sin((3*pi)/4 - Q5)*(Q5P)-
880 cos(Q2 + Q3 - Q4)*sin(Q1)*sin((3*pi)/4 + Q5)*(Q1P)-cos(Q1)*cos(Q2 - Q3 -
881 Q4)*cos((3*pi)/4 - Q5)*(Q5P)-cos(Q1)*sin(Q2 - Q3 + Q4)*sin((3*pi)/4 -
882 Q5)*(Q2P+Q3P+Q4P);
883 J66P = -cos((3*pi)/4 + Q5)*sin(Q2 + Q3 + Q4)*(Q5P)-cos(Q2 + Q3 +
884 Q4)*sin((3*pi)/4 + Q5)*(Q2P+Q3P+Q4P);
885
886
887 Jp = [J11P J12P J13P J14P J15P J16P;
888       J21P J22P J23P J24P J25P J26P;
889       J31P J32P J33P J34P J35P J36P;
890       J41P J42P J43P J44P J45P J46P;
891       J51P J52P J53P J54P J55P J56P;
892       J61P J62P J63P J64P J65P J66P];
893
894 A06 = [Xopp; 0; Zopp; 0; 0; 0];
895 cop = inv(C)*(A06-(Jp*qp));
896
897 Q1PP = qpp(1);
898 Q2PP = qpp(2);
899 Q3PP = qpp(3);
900 Q4PP = qpp(4);
901 Q5PP = qpp(5);
902 Q6PP = qpp(6);
903
904
905
906
907

```

© Modelo de Velocidad del la Pelvis al Pie (Lima del Robot FAO)



```

908
909
910
911      Matrice Jacobienne de la poutre au pic libre
912
913      J111 = d10*cos(Q7)*cos(((3*pi)/4) - Q8)*cos(Q9) + d11*cos(Q7)*cos(((3*pi)/4)
914      - Q8)*cos(Q10 + Q9) + d10*sin(Q7)*sin(Q9) - d11*sin(Q7)*sin(Q10 + Q9);
915
916      J211 = 0;
917
918      J311 = -(d10*cos(((3*pi)/4) - Q8)*cos(Q9)*sin(Q7)) - d11*cos(((3*pi)/4) -
919      Q8)*cos(Q10 - Q9)*sin(Q7) + d10*cos(Q7)*sin(Q9) + d11*cos(Q7)*sin(Q10 + Q9);
920
921      J411 = 0;
922
923      J511 = 1;
924
925      J611 = 0;
926
927      J122 = d10*cos(Q9)*sin(Q7)*sin(((3*pi)/4) - Q8) + d11*cos(Q10 +
928      Q9)*sin(Q7)*sin(((3*pi)/4) - Q8);
929
930      J222 = -(d10*cos(((3*pi)/4) - Q8)*cos(Q9)) - d11*cos(((3*pi)/4) - Q8)*cos(Q10
931      - Q9);
932
933      J322 = d10*cos(Q7)*cos(Q9)*sin(((3*pi)/4) - Q8) + d11*cos(Q7)*cos(Q10 +
934      Q9)*sin(((3*pi)/4) - Q8);
935
936      J422 = cos(Q7);
937
938      J522 = 0;
939
940      J622 = -sin(Q7);
941
942      J133 = -(d10*cos(Q7)*cos(Q9)) - d11*cos(Q7)*cos(Q10 - Q9) -
943      d10*cos(((3*pi)/4) - Q8)*sin(Q7)*sin(Q9) - d11*cos(((3*pi)/4) -
944      Q8)*sin(Q7)*sin(Q10 - Q9);
945
946      J233 = -(d10*sin(((3*pi)/4) - Q8)*sin(Q9)) - d11*sin(((3*pi)/4) - Q8)*sin(Q10
947      + Q9);
948
949      J333 = d10*cos(Q9)*sin(Q7) + d11*cos(Q10 - Q9)*sin(Q7) -
950      d10*cos(Q7)*cos(((3*pi)/4) - Q8)*sin(Q9) - d11*cos(Q7)*cos(((3*pi)/4) -
951      Q8)*sin(Q10 + Q9);
952
953      J433 = sin(Q7)*sin(((3*pi)/4) - Q8);
954
955      J533 = -cos(((3*pi)/4) - Q8);
956
957      J633 = cos(Q7)*sin(((3*pi)/4) - Q8);
958
959      J144 = -(d11*cos(Q7)*cos(Q10 + Q9)) - d11*cos(((3*pi)/4) -
960      Q8)*sin(Q7)*sin(Q10 + Q9);

```

```

961
962 J244 = -(d11*sin(((3*pi)/4) - Q8)*sin(Q10 + Q9));
963
964 J344 = d11*cos(Q10 - Q9)*sin(Q7) - d11*cos(Q7)*cos(((3*pi)/4) - Q8)*sin(Q10 +
965 Q9);
966
967 J444 = sin(Q7)*sin(((3*pi)/4) - Q8);
968
969 J544 = -cos(((3*pi)/4) - Q8);
970
971 J644 = cos(Q7)*sin(((3*pi)/4) - Q8);
972
973 J155 = 0;
974
975 J255 = 0;
976
977 J355 = 0;
978
979 J455 = sin(Q7)*sin(((3*pi)/4) - Q8);
980
981 J555 = -cos(((3*pi)/4) - Q8);
982
983 J655 = cos(Q7)*sin(((3*pi)/4) - Q8);
984
985 J166 = 0;
986
987 J266 = 0;
988
989 J366 = 0;
990
991 J466 = cos(Q7)*cos(Q10 + Q11 + Q9) + cos(((3*pi)/4) - Q8)*sin(Q7)*sin(Q10 +
992 Q11 + Q9);
993
994 J566 = sin(((3*pi)/4) - Q8)*sin(Q10 - Q11 + Q9);
995
996 J666 = -(cos(Q10 + Q11 + Q9)*sin(Q7)) + cos(Q7)*cos(((3*pi)/4) - Q8)*sin(Q10
997 + Q11 - Q9);
998
999
1000 V612 = [-Xsp+Xsp1ep; 0; -2sp+2pi1ep; 0; 0; 0];
1001
1002
1003 JJ = [ J111 J122 J133 J144 J155 J166;
1004        J211 J222 J233 J244 J255 J266;
1005        J311 J322 J333 J344 J355 J366;
1006        J411 J422 J433 J444 J455 J466;
1007        J511 J522 J533 J544 J555 J566;
1008        J611 J622 J633 J644 J655 J666];
1009
1010
1011
1012 qpc = inv(JJ)*V612;

```

```

1013 Q7P=qpd(1);
1014 Q8P=qpd(2);
1015 Q9P=qpd(3);
1016 Q10P=qpd(4);
1017 Q11P=qpd(5);
1018 Q12P=qpd(6);
1019
1020
1021 .....
1022 .....
1023
1024 J111P = -d10*cos(Q9)*cos(Q8 - (3*pi)/4)*sin(Q7)*(Q7P)-
1025 d10*cos(Q7)*cos(Q9)*sin(Q8 - (3*pi)/4)*(Q8P)-d10*cos(Q7)*cos(Q8 -
1026 (3*pi)/4)*sin(Q9)*(Q9P)-d11*cos(Q9 - Q10)*cos(Q8 - (3*pi)/4)*sin(Q7)*(Q7P)-
1027 d11*cos(Q9 - Q10)*cos(Q7)*sin(Q8 - (3*pi)/4)*(Q8P)-d11*sin(Q9 +
1028 Q10)*cos(Q7)*cos(Q8 -
1029 (3*pi)/4)*(Q9P+Q10P)+d10*cos(Q7)*sin(Q9)*(Q7P)+d10*cos(Q8)*sin(Q7)*(Q9P)+d11*
1030 sin(Q8 + Q10)*cos(Q7)*(Q7P)+d11*cos(Q9 - Q10)*sin(Q7)*(Q9P+Q10P);
1031 J211P = 0;
1032 J311P = -d10*cos(Q7)*cos(Q8)*cos(Q8 -
1033 (3*pi)/4)*(Q7P)+d10*cos(Q9)*sin(Q7)*sin(Q8 - (3*pi)/4)*(Q8P)+d10*cos(Q8 -
1034 (3*pi)/4)*sin(Q7)*sin(Q9)*(Q9P)-d11*cos(Q9 - Q10)*cos(Q7)*cos(Q8 -
1035 (3*pi)/4)*(Q7P)+d11*cos(Q9 - Q10)*sin(Q7)*sin(Q8 - (3*pi)/4)*(Q8P)+d11*sin(Q9
1036 - Q10)*cos(Q8 - (3*pi)/4)*sin(Q7)*(Q9P+Q10P)-
1037 d10*sin(Q7)*sin(Q9)*(Q7P)-d10*cos(Q7)*cos(Q9)*(Q9P)-d11*sin(Q9 -
1038 Q10)*sin(Q7)*(Q7P)-d11*cos(Q9 - Q10)*cos(Q7)*(Q9P+Q10P);
1039 J411P = 0;
1040 J511P = 0;
1041 J611P = 0;
1042 J122P = -d10*cos(Q7)*cos(Q9)*sin(Q8 - (3*pi)/4)*(Q7P)-d10*cos(Q9)*cos(Q8 -
1043 (3*pi)/4)*sin(Q7)*(Q8P)+d10*sin(Q7)*sin(Q9)*sin(Q8 - (3*pi)/4)*(Q9P);
1044 J222P = d10*cos(Q9)*sin(Q8 - (3*pi)/4)*(Q8P)+d10*cos(Q8 -
1045 (3*pi)/4)*sin(Q9)*(Q9P)-d11*cos(Q9 - Q10)*sin(Q8 - (3*pi)/4)*(Q8P)+d11*sin(Q9
1046 + Q10)*cos(Q8 - (3*pi)/4)*(Q9P+Q10P);
1047 J322P = d10*cos(Q9)*sin(Q7)*sin(Q8 - (3*pi)/4)*(Q7P)-
1048 d10*cos(Q7)*cos(Q9)*cos(Q8 - (3*pi)/4)*(Q8P)+d10*cos(Q7)*sin(Q9)*sin(Q8 -
1049 (3*pi)/4)*(Q9P)+d11*cos(Q9 - Q10)*sin(Q7)*sin(Q8 - (3*pi)/4)*(Q7P)-d11*cos(Q9
1050 - Q10)*cos(Q7)*cos(Q8 - (3*pi)/4)*(Q8P)+d11*sin(Q9 + Q10)*cos(Q7)*sin(Q8 -
1051 (3*pi)/4)*(Q9P+Q10P);
1052 J422P = -sin(Q7)*(Q7P);
1053 J522P = 0;
1054 J622P = -cos(Q7)*(Q7P);
1055 J133P = d10*cos(Q9)*sin(Q7)*(Q7P)+d10*cos(Q7)*sin(Q9)*(Q9P)+d11*cos(Q9 -
1056 Q10)*sin(Q7)*(Q7P)+d11*sin(Q9 + Q10)*cos(Q7)*(Q9P+Q10P)-d10*cos(Q7)*cos(Q8 -
1057 (3*pi)/4)*sin(Q9)*(Q7P)-d10*sin(Q7)*sin(Q9)*sin(Q8 - (3*pi)/4)*(Q8P)-
1058 d10*cos(Q9)*cos(Q8 - (3*pi)/4)*sin(Q7)*(Q9P)-d11*sin(Q9 + Q10)*cos(Q7)*cos(Q8
1059 - (3*pi)/4)*(Q7P)+d11*sin(Q9 - Q10)*sin(Q7)*sin(Q8 - (3*pi)/4)*(Q8P)-
1060 d11*cos(Q9 - Q10)*cos(Q8 - (3*pi)/4)*sin(Q7)*(Q9P+Q10P);
1061 J233P = d10*cos(Q8 - (3*pi)/4)*sin(Q7)*(Q8P)+d10*cos(Q9)*sin(Q8 -
1062 (3*pi)/4)*(Q9P)+d11*sin(Q9 + Q10)*cos(Q8 - (3*pi)/4)*(Q8P)+d11*cos(Q9 -
1063 Q10)*sin(Q8 - (3*pi)/4)*(Q9P+Q10P);
1064 J333P = d10*cos(Q7)*cos(Q9)*(Q7P)-d10*sin(Q7)*sin(Q9)*(Q9P)-d11*cos(Q9 +
1065 Q10)*cos(Q7)*(Q7P)-d11*sin(Q9 - Q10)*sin(Q7)*(Q9P+Q10P)+d10*cos(Q8 -
1066 (3*pi)/4)*sin(Q7)*sin(Q9)*(Q7P)+d10*cos(Q7)*sin(Q9)*sin(Q8 - (3*pi)/4)*(Q8P)-
1067 d10*cos(Q7)*cos(Q9)*cos(Q8 - (3*pi)/4)*(Q9P)+d11*sin(Q9 - Q10)*cos(Q8 -

```



```

1068 (3*pi)/4)*s_n(Q7)*(Q7P+d11*sin(Q9 + Q10)*cos(Q7)*sin(Q8 - (3*pi)/4)*(Q8P)-
1069 d11*cos(Q9 + Q10)*cos(Q7)*cos(Q8 - (3*pi)/4)*(Q9P+Q10P));
1070 J433P = cos(Q7)*sin(Q8 - (3*pi)/4)*(Q7P)-cos(Q8 - (3*pi)/4)*sin(Q7)*(Q8P);
1071 J533P = sin(Q8 - (3*pi)/4)*(Q8P);
1072 J633P = s_n(Q7)*sin(Q8 - (3*pi)/4)*(Q7P)-cos(Q7)*cos(Q8 - (3*pi)/4)*(Q8P);
1073 J144P = d11*cos(Q9 + Q10)*sin(Q7)*(Q7P)+d11*sin(Q9 + Q10)*cos(Q7)*(Q9P-Q10P);
1074 J244P = d11*sin(Q9 + Q10)*cos(Q8 - (3*pi)/4)*(Q8P) d11*cos(Q9 + Q10)*sin(Q8 -
1075 (3*pi)/4)*(Q9P+Q10P);
1076 J344P = d11*cos(Q9 + Q10)*cos(Q7)*(Q7P)-d11*sin(Q9 +
1077 Q10)*sin(Q7)*(Q9P-Q10P)-d11*sin(Q9 + Q10)*cos(Q8 -
1078 (3*pi)/4)*sin(Q7)*(Q7P)-d11*sin(Q9 + Q10)*cos(Q7)*sin(Q8 - (3*pi)/4)*(Q8P)-
1079 d11*cos(Q9 + Q10)*cos(Q7)*cos(Q8 - (3*pi)/4)*(Q9P-Q10P);
1080 J444P = -cos(Q7)*sin(Q8 - (3*pi)/4)*(Q7P)-cos(Q8 - (3*pi)/4)*sin(Q7)*(Q8P);
1081 J544P = s_n(Q8 - (3*pi)/4)*(Q8P);
1082 J644P = d11*(Q7)*sin(Q8 - (3*pi)/4)*(Q7P)-cos(Q7)*cos(Q8 - (3*pi)/4)*(Q8P);
1083 J155P = 0;
1084 J255P = 0;
1085 J355P = 0;
1086 J455P = -cos(Q7)*sin(Q8 - (3*pi)/4)*(Q7P)-cos(Q8 - (3*pi)/4)*sin(Q7)*(Q8P);
1087 J555P = sin(Q8 - (3*pi)/4)*(Q8P);
1088 J655P = sin(Q7)*sin(Q8 - (3*pi)/4)*(Q7P)-cos(Q7)*cos(Q8 - (3*pi)/4)*(Q8P);
1089 J166P = 0;
1090 J266P = 0;
1091 J366P = 0;
1092 J466P = -cos(Q8 + Q10 + Q11)*s_n(Q7)*(Q7P)-cos(Q7)*sin(Q9 + Q10 +
1093 Q11)*(Q9P+Q10P-Q11P)+cos(Q7)*cos(Q8 - (3*pi)/4)*sin(Q9 + Q10 + Q11)*(Q7P)-
1094 sin(Q7)*sin(Q9 + Q10 + Q11)*sin(Q8 - (3*pi)/4)*(Q8P)+cos(Q8 + Q10 +
1095 Q11)*cos(Q8 - (3*pi)/4)*sin(Q7)*(Q9P-Q10P-Q11P);
1096 J566P = -cos(Q8 - (3*pi)/4)*sin(Q9 + Q10 + Q11)*(Q8P)-cos(Q9 + Q10 +
1097 Q11)*sin(Q8 - (3*pi)/4)*(Q9P-Q10P-Q11P);
1098 J666P = -cos(Q7)*sin(Q8 + Q10 + Q11)*(Q7P)-sin(Q7)*sin(Q9 + Q10 +
1099 Q11)*(Q9P-Q10P+Q11P)-cos(Q8 - (3*pi)/4)*sin(Q7)*sin(Q9 + Q10 + Q11)*(Q7P)-
1100 cos(Q7)*sin(Q9 + Q10 + Q11)*sin(Q8 - (3*pi)/4)*(Q8P)+cos(Q7)*cos(Q9 + Q10 +
1101 Q11)*cos(Q8 - (3*pi)/4)*(Q9P+Q10P+Q11P);
1102
1103
1104 Jp3 = [J111P J122P J133P J144P J155P J166P;
1105 J211P J222P J233P J244P J255P J266P;
1106 J311P J322P J333P J344P J355P J366P;
1107 J411P J422P J433P J444P J455P J466P;
1108 J511P J522P J533P J544P J555P J566P;
1109 J611P J622P J633P J644P J655P J666P];
1110
1111 A612 = [Xpicpp; 0; Jp3*cpp; 0; 0; 0];
1112
1113
1114 qpp1 = inv(JJ)*(A612-(Jp1*ppd));
1115
1116
1117
1118 Q7PP = qpp1(1);
1119 Q8PP = qpp1(2);
1120 Q9PP = qpp1(3);
1121 Q10PP = qpp1(4);
1122 Q11PP = qpp1(5);

```



```

1123 Q12PP = que(6);
1124
1125
1126 *Definición de los ángulos  $Q1, Q2, Q3, Q4, \dots$  para las piernas del
1127 robot
1128
1129
1130 s1=sin(Q1);
1131 c1=cos(Q1);
1132 s2=sin(Q2);
1133 c2=cos(Q2);
1134 s3=sin(Q3);
1135 c3=cos(Q3);
1136 s4=sin(Q4);
1137 c4=cos(Q4);
1138 s5=sin((3*pi/4)+Q5);
1139 c5=cos((3*pi/4)+Q5);
1140 s6=sin(Q6);
1141 c6=cos(Q6);
1142 s7=sin(Q7);
1143 c7=cos(Q7);
1144 s8=sin((3*pi/4)+Q8);
1145 c8=cos((3*pi/4)+Q8);
1146 s9=sin(Q9);
1147 c9=cos(Q9);
1148 s10=sin(Q10);
1149 c10=cos(Q10);
1150 s11=sin(Q11);
1151 c11=cos(Q11);
1152 s12=sin(Q12);
1153 c12=cos(Q12);
1154
1155
1156
1157 *Matriz de transición del marco del mundo a la base de la primera
1158 articulación de la
1159 *pierna.
1160
1161 T0 = rotz(2) - 1
1162
1163     TMC = [0 0 1 Xfijo; % Cuando la pierna derecha es el pie de apoyo
1164           1 0 0 -d7;
1165           0 1 0 0;
1166           0 0 0 1];
1167
1168 else
1169
1170     TMC = [0 0 1 Xfijo; % Cuando la pierna izquierda es el pie de apoyo
1171           1 0 0 d3;
1172           0 1 0 0;
1173           0 0 0 1];
1174 end
1175
1176

```

```

1177  @Matrices elementales de la pila del modo KAO.
1178
1179
1180  TP01 = [-s1 -c1 0 0;
1181         c1 -s1 0 0;
1182         0 0 1 0;
1183         0 0 0 1];
1184
1185
1186  TP02 = [s2 -s2 0 0;
1187         0 0 1 0;
1188         -s2 -c2 0 0;
1189         0 0 0 1];
1190
1191
1192  TP03 = [c3 -s3 0 d3;
1193         s3 c3 0 0;
1194         0 0 1 0;
1195         0 0 0 1];
1196
1197
1198  TP04 = [c4 -s4 0 d4;
1199         s4 c4 0 0;
1200         0 0 1 0;
1201         0 0 0 1];
1202
1203
1204  TP05 = [s5 -s5 0 0;
1205         0 0 1 0;
1206         -s5 -c5 0 0;
1207         0 0 0 1];
1208
1209
1210
1211  TP06 = [s6 c6 0 0;
1212         0 0 -1 0;
1213         -c6 s6 0 0;
1214         0 0 0 1];
1215
1216
1217
1218  if rem(i,2)==1
1219
1220      P6B = [1 0 0 0; % Cuando la pila derecha es el pie de apoyo
1221            0 1 0 0;
1222            0 0 1 d7/z;
1223            0 0 0 1];
1224
1225      CP7 = [s7 c7 0 0;
1226            0 0 1 d7/z;
1227            c7 -s7 0 0;
1228            0 0 0 1];
1229

```

```

1230 else
1231
1232     TP67 = [1 0 0 0; % cuando la pierna izquierda es el pie de apoyo
1233            0 1 0 -d77;
1234            0 0 1 0;
1235            0 0 0 1];
1236
1237     TP7 = [s7 c7 0 0;
1238           0 0 1 0;
1239           c7 -s7 0 -d77;
1240           0 0 0 1];
1241
1242 end
1243
1244
1245 TP78 = [c8 s8 0 0;
1246        0 0 1 0;
1247        s8 -c8 0 0;
1248        0 0 0 1];
1249
1250
1251 TP89 = [c9 -s9 0 0;
1252        0 0 1 0;
1253        -s9 -c9 0 0;
1254        0 0 0 1];
1255
1256
1257 TP910 = [c10 -s10 0 d10;
1258         s10 s10 0 0;
1259         0 0 1 0;
1260         0 0 0 1];
1261
1262
1263 TP1011 = [c11 -s11 0 d11;
1264          s11 c11 0 0;
1265          0 0 1 0;
1266          0 0 0 1];
1267
1268
1269
1270 TP1112 = [c12 -s12 0 0;
1271          0 0 1 0;
1272          s12 s12 0 0;
1273          0 0 0 1];
1274
1275
1276
1277 *Recibir de C a cada punto del robot.
1278
1279 m03P = TMC;
1280 m01P = TMC*TP01;
1281 m02P = TMC*TP01*TP12;
1282 m03P = TMC*TP01*TP12*TP23;

```

```

1283 m04P = TMO*TP01*TP12*TP23*TP34;
1284 m05P = TMO*TP01*TP13*TP23*TP34*TP45;
1285 m06P = TMO*TP01*TP12*TP23*TP34*TP45*TP56;
1286 m60P = TMO*TP01*TP12*TP23*TP34*TP45*TP56*TP6P;
1287
1288 m07P = TMO*TP01*TP12*TP23*TP34*TP45*TP56*TP6P*TP7;
1289 m08P = TMO*TP01*TP13*TP23*TP34*TP45*TP56*TP6P*TP7*TP78;
1290 m09P = TMO*TP01*TP12*TP23*TP34*TP45*TP56*TP6P*TP7*TP78*TP89;
1291 m10P = TMO*TP01*TP12*TP23*TP34*TP45*TP56*TP6P*TP7*TP78*TP89*TP910;
1292 m11P = TMO*TP01*TP12*TP23*TP34*TP45*TP56*TP6P*TP7*TP78*TP89*TP910*TP11;
1293 m12P =
1294 TMO*TP01*TP12*TP23*TP34*TP45*TP56*TP6P*TP7*TP78*TP89*TP910*TP1011*TP1112;
1295
1296
1297
1298 p00P = [m00P(1,4);m00P(2,4);m00P(3,4)];
1299 p11P = [m01P(1,4);m01P(2,4);m01P(3,4)];
1300 p22P = [m02P(1,4);m02P(2,4);m02P(3,4)];
1301 p33P = [m03P(1,4);m03P(2,4);m03P(3,4)];
1302 p44P = [m04P(1,4);m04P(2,4);m04P(3,4)];
1303 p55P = [m05P(1,4);m05P(2,4);m05P(3,4)];
1304 p66P = [m06P(1,4);m06P(2,4);m06P(3,4)];
1305 p66P = [m60P(1,4);m60P(2,4);m60P(3,4)];
1306
1307 p77P = [m07P(1,4);m07P(2,4);m07P(3,4)];
1308 p88P = [m08P(1,4);m08P(2,4);m08P(3,4)];
1309 p99P = [m09P(1,4);m09P(2,4);m09P(3,4)];
1310 p010P = [m10P(1,4);m10P(2,4);m10P(3,4)];
1311 p011P = [m11P(1,4);m11P(2,4);m11P(3,4)];
1312 p012P = [m12P(1,4);m12P(2,4);m12P(3,4)];
1313
1314
1315 4 Bars: 3-bujas los eslabones de las pircas
1316
1317 10xPD=[p00P(1) p11P(1)];
1318 10yPD=[p00P(2) p11P(2)];
1319 10zPD=[p00P(3) p11P(3)];
1320 11xPD=[p11P(1) p22P(1)];
1321 11yPD=[p11P(2) p22P(2)];
1322 11zPD=[p11P(3) p22P(3)];
1323 12xPD=[p22P(1) p33P(1)];
1324 12yPD=[p22P(2) p33P(2)];
1325 12zPD=[p22P(3) p33P(3)];
1326 13xPD=[p33P(1) p44P(1)];
1327 13yPD=[p33P(2) p44P(2)];
1328 13zPD=[p33P(3) p44P(3)];
1329 14xPD=[p44P(1) p55P(1)];
1330 14yPD=[p44P(2) p55P(2)];
1331 14zPD=[p44P(3) p55P(3)];
1332 15xPD=[p55P(1) p66P(1)];
1333 15yPD=[p55P(2) p66P(2)];
1334 15zPD=[p55P(3) p66P(3)];
1335 16xPD=[p66P(1) p66P(1)];
1336 16yPD=[p66P(2) p66P(2)];
1337 16zPD=[p66P(3) p66P(3)];

```



```

1338 17xPD=[p66P(1) p77P(1)];
1339 17yPD=[p66P(2) p77P(2)];
1340 17zPD=[p66P(3) p77P(3)];
1341 18xPD=[p77P(1) p88P(1)];
1342 18yPD=[p77P(2) p88P(2)];
1343 18zPD=[p77P(3) p88P(3)];
1344 19xPD=[p88P(1) p99P(1)];
1345 19yPD=[p88P(2) p99P(2)];
1346 19zPD=[p88P(3) p99P(3)];
1347 110xPD=[p99P(1) p010P(1)];
1348 110yPD=[p99P(2) p010P(2)];
1349 110zPD=[p99P(3) p010P(3)];
1350 111xPD=[p010P(1) p011P(1)];
1351 111yPD=[p010P(2) p011P(2)];
1352 111zPD=[p010P(3) p011P(3)];
1353 112xPD=[p011P(1) p012P(1)];
1354 112yPD=[p011P(2) p012P(2)];
1355 112zPD=[p011P(3) p012P(3)];
1356
1357
1358 %-----
1359
1360 %-----
1361
1362 %Definición de las constantes a1, a2, a3, a4, a5, a6 para el modo normal.
1363
1364 a1D=sin(-B);
1365 a1D=cos(-B);
1366 a2D=sin(Q2D);
1367 a2D=cos(Q2D);
1368 a3D=sin(Q3D);
1369 a3D=cos(Q3D);
1370 a4D=sin(Q4D);
1371 a4D=cos(Q4D);
1372 a5D=sin(Q5D);
1373 a5D=cos(Q5D);
1374
1375
1376 % Matriz de desplazamiento
1377
1378 Map=[1 0 0 0;
1379      0 1 0 0;
1380      0 0 1 0;
1381      0 0 0 1];
1382
1383 pel=[t11 t12 t13 %fijojop;
1384      t21 t22 t23 %o;
1385      t31 t32 t33 %p=L5;
1386      0 0 0 1];
1387
1388 A1BD = pel*[1 0 0 0;
1389           0 sqrt(.5) -sqrt(.5) 0;
1390           0 sqrt(.5) sqrt(.5) 0;
1391           0 0 0 1];
1392

```

```

1393 ATBDD = [1 0 0 0;
1394          0 1 0 -L4;
1395          0 0 1 0;
1396          0 0 0 1];
1397
1398
1399 Matriz de transformacion del Tm articulacion 4 del Brazo izquierdo a la coordenada
1400 de la mano izquierda.
1401
1402
1403 A4BD = [1 0 0 0;
1404         0 1 0 0;
1405         0 0 1 L6;
1406         0 0 0 1];
1407
1408
1409 Matrices elementales de la Mano Derecha del Robot 2d.
1410
1411 TBD00 = [c1D -s1D 0 0;
1412         0 0 1 0;
1413         -s1D -c1D 0 0;
1414         0 0 0 1];
1415
1416 TBD12 = [-s2D -c2D 0 0;
1417         0 0 -1 0;
1418         c2D -s2D 0 0;
1419         0 0 0 1];
1420
1421 TBD23 = [c3D -s3D 0 -d3T;
1422         0 0 -1 -r3T;
1423         s3D c3D 0 0;
1424         0 0 0 1];
1425
1426 TBD34 = [c4D -s4D 0 0;
1427         0 0 1 0;
1428         -s4D -c4D 0 0;
1429         0 0 0 1];
1430
1431 TBD45 = [c5D -s5D 0 0;
1432         0 0 -1 0;
1433         s5D c5D 0 0;
1434         0 0 0 1];
1435
1436 Matriz de Estado Unica del Robot
1437
1438 mtrD = ATBD;
1439 mncD = ATBD*ATBDD;
1440 mc1D = ATBD*ATBDD*TBD01;
1441 mc2D = ATBD*ATBDD*TBD01*TBD12;
1442 mc3D = ATBD*ATBDD*TBD01*TBD12*TBD23;
1443 mc4D = ATBD*ATBDD*TBD01*TBD12*TBD23*TBD34;
1444 mc5D = ATBD*ATBDD*TBD01*TBD12*TBD23*TBD34*TBD45;
1445 mcpD = ATBD*ATBDD*PA0T*TBD12*TBD23*TBD34*TR045*A4BD;
1446

```

```

1447
1448 *Definición de los puntos
1449
1450 p1xD = [m1xD(1,4);m1xD(2,4);m1xD(3,4)]';
1451 p1yD = [m1yD(1,4);m1yD(2,4);m1yD(3,4)]';
1452 p11D = [m01D(1,4);m01D(2,4);m01D(3,4)]';
1453 p22D = [m02D(1,4);m02D(2,4);m02D(3,4)]';
1454 p33D = [m03D(1,4);m03D(2,4);m03D(3,4)]';
1455 p44D = [m04D(1,4);m04D(2,4);m04D(3,4)]';
1456 p55D = [m05D(1,4);m05D(2,4);m05D(3,4)]';
1457 p5pD = [m0pD(1,4);m0pD(2,4);m0pD(3,4)]';
1458
1459 % Para dibujar los eslabones de la mano derecha del robot nro
1460
1461 p1xD=[p1xD(1) p1xD(1)];
1462 p1yD=[p1yD(1) p1yD(1)];
1463 p1zD=[p1zD(1) p1zD(1)];
1464 a2xD=[p1xD(1) p11D(1)];
1465 a2yD=[p1yD(1) p11D(1)];
1466 a2zD=[p1zD(1) p11D(1)];
1467 l1xD=[p11D(1) p22D(1)];
1468 l1yD=[p11D(2) p22D(2)];
1469 l1zD=[p11D(3) p22D(3)];
1470 l2xD=[p22D(1) p33D(1)];
1471 l2yD=[p22D(2) p33D(2)];
1472 l2zD=[p22D(3) p33D(3)];
1473 l3xD=[p33D(1) p44D(1)];
1474 l3yD=[p33D(2) p44D(2)];
1475 l3zD=[p33D(3) p44D(3)];
1476 l4xD=[p44D(1) p55D(1)];
1477 l4yD=[p44D(2) p55D(2)];
1478 l4zD=[p44D(3) p55D(3)];
1479 lpxD=[p55D(1) p5pD(1)];
1480 lpyD=[p55D(2) p5pD(2)];
1481 lpzD=[p55D(3) p5pD(3)];
1482
1483
1484 %%%*****%%
1485
1486 *MANG IZQUIERDA
1487
1488
1489 *Definición de las constantes s1,s1',s2',s2',s3',s3',... para la mano izquierda
1490
1491 s1I=sin(-B12);
1492 c1I=cos(-B12);
1493 s2I=sin(Q21);
1494 c2I=cos(Q21);
1495 s3I=sin(Q31);
1496 c3I=cos(Q31);
1497 s4I=sin(Q41);
1498 c4I=cos(Q41);
1499 s5I=sin(Q51);
1500 c5I=cos(Q51);
1501

```

```

1502
1503 *Matriz de transformacion del torso a la base de la primera articulacion de la
1504 *mano izquierda.
1505
1506
1507 ATB1 = pel*[1 0 0 0;
1508         0 sqrt(.5) -sqrt(.5) 0;
1509         0 sqrt(.5) sqrt(.5) 0;
1510         0 0 0 1];
1511
1512
1513 APB1 = 1 0 0 0;
1514       0 1 0 0;
1515       0 0 1 0;
1516       0 0 0 1];
1517
1518 *Matriz de transformacion del la articulacion 4 del brazo izquierdo a la pinna
1519 *de la mano izquierda.
1520
1521
1522 A4B1 = [1 0 0 0;
1523         0 1 0 0;
1524         0 0 1 0;
1525         0 0 0 1];
1526
1527
1528
1529 *Matrices Elementales de la Mano Izquierda del Robot.
1530
1531
1532 TB101 = [c11 -s11 0 0;
1533          0 0 1 0;
1534          -s11 -c11 0 0;
1535          0 0 0 1];
1536
1537 TB112 = [-s21 -c21 0 0;
1538          0 0 -1 0;
1539          c21 -s21 0 0;
1540          0 0 0 1];
1541
1542 TB123 = [c31 -s31 0 d31;
1543          0 0 -1 -s31;
1544          s31 c31 0 0;
1545          0 0 0 1];
1546
1547 TB134 = [c41 -s41 0 0;
1548          0 0 1 0;
1549          -s41 -c41 0 0;
1550          0 0 0 1];
1551
1552 TB145 = [c51 -s51 0 0;
1553          0 0 -1 0;
1554          s51 c51 0 0;

```



```

1555      0 0 0 1];
1556
1557
1558  *Matriz de 5 a Carta Falso del Rombo.
1559
1560  m1z = A1B1;
1561  m1d = A1B1*ATB11;
1562  m01 = A1B1*ATB11*ATB101;
1563  m02 = A1B1*ATB11*ATB101*ATB112;
1564  m03 = A1B1*ATB11*ATB101*ATB112*ATB123;
1565  m04 = A1B1*ATB11*ATB101*ATB112*ATB123*ATB134;
1566  m05 = A1B1*ATB11*ATB101*ATB112*ATB123*ATB134*ATB145;
1567  m0p = A1B1*ATB11*ATB101*ATB112*ATB123*ATB134*ATB145*A4B1;
1558
1559  *Determinación de puntos.
1570
1571  p1z = [m1z(1,4);m1z(2,4);m1z(3,4)];
1572  p1d = [m1d(1,4);m1d(2,4);m1d(3,4)];
1573  p01 = [m01(1,4);m01(2,4);m01(3,4)];
1574  p02 = [m02(1,4);m02(2,4);m02(3,4)];
1575  p03 = [m03(1,4);m03(2,4);m03(3,4)];
1576  p04 = [m04(1,4);m04(2,4);m04(3,4)];
1577  p05 = [m05(1,4);m05(2,4);m05(3,4)];
1578  p0p = [m0p(1,4);m0p(2,4);m0p(3,4)];
1579
1580  * Para dibujar los arlabones.
1581
1582  p1x=[p1z(1) p1d(1)];
1583  p1y=[p1z(2) p1d(2)];
1584  p1z=[p1z(3) p1d(3)];
1585  d2x=[p1d(1) p01(1)];
1586  d2y=[p1d(2) p01(2)];
1587  d2z=[p1d(3) p01(3)];
1588  l1x=[p01(1) p02(1)];
1589  l1y=[p01(2) p02(2)];
1590  l1z=[p01(3) p02(3)];
1591  l2x=[p02(1) p03(1)];
1592  l2y=[p02(2) p03(2)];
1593  l2z=[p02(3) p03(3)];
1594  l3x=[p03(1) p04(1)];
1595  l3y=[p03(2) p04(2)];
1596  l3z=[p03(3) p04(3)];
1597  l4x=[p04(1) p05(1)];
1598  l4y=[p04(2) p05(2)];
1599  l4z=[p04(3) p05(3)];
1600  lpx=[p05(1) p0p(1)];
1601  lpy=[p05(2) p0p(2)];
1602  lpz=[p05(3) p0p(3)];
1603
1604
1605  *-----*
1606
1607      *CADUCA
1608

```

```

1609
1610  SdaPinta:en Se las constantes x1,x1,x2,x2,x3,x3,... del 1a Capitulo
1611
1612  s1c=sin(Q1C);
1613  c1c=cos(Q1C);
1614  s2c=sin(Q2C);
1615  c2c=cos(Q2C);
1616
1617
1618  pel=[t11 t12 t13 Xff]e-exp;
1619      t21 t22 t23 Ye;
1620      .31 t32 t33 zc;
1621      0 0 0 1];
1622
1623  A1C = pel*[1 0 0 0;
1624          0 sqrt(.5) -sqrt(.5) 0;
1625          0 sqrt(.5) sqrt(.5) 0;
1626          0 0 0 1];
1627
1628
1629  A2C = [1 0 0 0;
1630        0 1 0 0;
1631        0 0 1 t3;
1632        0 0 0 1];
1633
1634  A3C = [1 0 0 0;
1635        0 1 0 0;
1636        0 0 1 t2;
1637        0 0 0 1];
1638
1639  A22C = [1 0 0 t1;
1640         0 1 0 0;
1641         0 0 1 0;
1642         0 0 0 1];
1643  %Matrices de Rotacion
1644
1645  RxC = [1 0 0 0;
1646        0 0 -1 0;
1647        0 1 0 0;
1648        0 0 0 1];
1649
1650  RyC = [0 0 1 0;
1651        0 1 0 0;
1652        -1 0 0 0;
1653        0 0 0 1];
1654
1655
1656  MATRICES TRANSFORMADAS DEL ROBO WAO
1657
1658  TCC1 = [e1c -e1s 0 0;
1659         e1c e1c 0 0;
1660         0 0 1 0;
1661         0 0 0 1];
1662

```

```

1663 TC12 = 1s2c+e2c 0 0;
1664      0 0 1 0;
1665      s2c -e2c 0 0;
1666      3 0 0 1;
1667
1668
1669 %Matriz de C a Cada Punto del Robot
1670 mepc = Mep*A10;
1671 m00c = Yep*A10*A10;
1672 m01c = A10*A10*TC01;
1673 m02c = A10*A10*TC01*TC12;
1674 m1c = A10*A10*TC01*TC12*x0c+3yc*A20;
1675 m22c = A10*A10*TC01*TC12*x0c+3yc*A20*A22c;
1676
1677 %Definición de los vectores
1678
1679 p0pc = [mepc(1,4); mepc(2,4); mepc(3,4)];
1680 p00c = [m00c(1,4); m00c(2,4); m00c(3,4)];
1681 p11c = [m01c(1,4); m01c(2,4); m01c(3,4)];
1682 p22c = [m02c(1,4); m02c(2,4); m02c(3,4)];
1683 p2c = [m2c(1,4); m2c(2,4); m2c(3,4)];
1684 p22c = [m22c(1,4); m22c(2,4); m22c(3,4)];
1685 % Para obtener los valores
1686
1687 1pxC=[p0pc(1) p00c(1)];
1688 1pyC=[p0pc(2) p00c(2)];
1689 1pzC=[p0pc(3) p00c(3)];
1690 1x0C=[p00c(1) e11c(1)];
1691 1y0C=[p00c(2) e11c(2)];
1692 1z0C=[p00c(3) e11c(3)];
1693 1x1C=[p11c(1) p22c(1)];
1694 1y1C=[p11c(2) p22c(2)];
1695 1z1C=[p11c(3) p22c(3)];
1696 1x2C=[p22c(1) p2c(1)];
1697 1y2C=[p22c(2) p2c(2)];
1698 1x3C=[p22c(3) p2c(3)];
1699 1x2xC=[p2c(1) p2cc(1)];
1700 1y2xC=[p2c(2) p2cc(2)];
1701 1z2xC=[p2c(3) p2cc(3)];
1702
1703
1704 %-----
1705 * Para poder graficar las posiciones, velocidades y aceleraciones deseadas.
1706 %-----
1707
1708 time(i+1) = 1/i/rp;
1709
1710 PosDeseadas(i)(i+1,:) = [Xc Zc Xpie Zpie];
1711 VelDeseadas(i)(i+1,:) = [xcp zcp xpiep zpiep];
1712 AccDeseadas(i)(i+1,:) = [xcpp zcpp xpiepp zpiepp];
1713
1714 VelArticulares(i)(i+1,:) = [Q1F Q2F Q3F Q4F Q5F Q6F];
1715 AccArticulares(i)(i+1,:) = [Q7F Q8F Q9F Q10F Q11F Q12F];
1716
1717 AccArticulares2(i)(i+1,:) = [Q1FF Q2FF Q3FF Q4FF Q5FF Q6FF];
1718 AccArticulares2(i)(i+1,:) = [Q7FF Q8FF Q9FF Q10FF Q11FF Q12FF];

```

```

1718
1719
1720 %-----
1721 % Para poder graficar las posiciones, velocidades y aceleraciones articulares
1722 % de cada articulación.
1723 %-----
1724
1725
1726 if rem(i,2)==1 % Para el paso 1,3,5...
1727     PosGlobales(k,:) = rad2deg([Q1 Q2 Q3 Q4 Q5 Q6]);
1728     PosGlobales2(k,:) = rad2deg([Q7 Q8 Q9 Q10 Q11 Q12]);
1729 else % Para el paso 2,4,6...
1730     PosGlobales(k,:) = rad2deg([Q1 Q2 Q3 Q4 Q5 Q6]);
1731     PosGlobales2(k,:) = rad2deg([Q7 Q8 Q9 Q10 Q11 Q12]);
1732 end
1733
1734 if rem(j,2)==1 % Para el paso 1,3,5...
1735     AcciArticulares1(k,:) = rad2deg([Q1PP Q2PP Q3PP Q4PP Q5PP Q6PP]);
1736     AcciArticulares2(k,:) = rad2deg([Q7PP Q8PP Q9PP Q10PP Q11PP Q12PP]);
1737 else
1738     AccArticulares1(k,:) = rad3deg([Q1PP Q2PP Q3PP Q4PP Q5PP Q6PP]);
1739     AccArticulares2(k,:) = rad2deg([Q7PP Q8PP Q9PP Q10PP Q11PP Q12PP]);
1740 end
1741
1742 timeTotal(k) = T*(k-1)/np;
1743
1744 k=s+1;
1745
1746
1747 %-----
1748 %-----
1749
1750 figure(1)
1751 clf
1752 hold on
1753
1754 %----- MANO DERECHA -----
1755 plot3(plxD,plyD,plzD,'k',d2xD,d2yD,d2zD,'k',11xD,11yD,11zD,'b-
1756 s',12xD,12yD,12zD,'k-o',13xD,13yD,13zD,'k-o',14xD,14yD,14zD,'k-
1757 s',1pxD,1pyD,1pzD,'k-o','linewidth',2)
1758
1759
1760
1761 %----- MANO IZQUIERDA -----
1762
1763 plot3(plx,ply,plz,'k',d2x,d2y,d2z,'k',11x,11y,11z,'b-o',12x,12y,12z,'k-
1764 s',13x,13y,13z,'k-o',14x,14y,14z,'k-o',1px,1py,1pz,'k-o','linewidth',2)
1765
1766
1767 %----- CINTURA -----
1768
1769

```



```

1770 plot3(1pxC,1pyC,1pzC,'k',10xC,10yC,10zC,'g',11xC,11yC,11zC,'b-
1771 c',12xC,12yC,12zC,'k-o',122xC,122yC,122zC,'k-o','lineaddb',2)
1772
1773
1774
1775 plot3(10xPD,10yPD,10zPD,'k-o',11xPD,11yPD,11zPD,'k-o',12xPD,12yPD,12zPD,'k-
1776 o',13xPD,13yPD,13zPD,'k-o',14xPD,14yPD,14zPD,'k-o',15xPD,15yPD,15zPD,'k-
1777 o',16xPD,16yPD,16zPD,'k-o',17xPD,17yPD,17zPD,'k-o',18xPD,18yPD,18zPD,'k-
1778 o',19xPD,19yPD,19zPD,'k-o',110xPD,110yPD,110zPD,'k-
1779 o',111xPD,111yPD,111zPD,'k-o',112xPD,112yPD,112zPD,'k-o','lineadda',2)
1780
1781
1782
1783
1784
1785     * Configuración de la ventana de visualización de la simulación.
1786
1787     title('ROBOT KUKA');
1788     xlabel('Eje x');
1789     ylabel('Eje y');
1790     zlabel('Eje z');
1791     axis([-200,200,-400,400,-0,500]);
1792     axis('square');
1793     view([0,0]);
1794     view([1,1,1]);
1795     grid on;
1796
1797
1798     end
1799
1800
1801     if rem(j,2)~=1
1802
1803         theta1 = deg2rad(-90-10); %Es el ángulo de inclinación de la articulación j
1804         theta2 = 0.25; %Es la amplitud de la oscilación de la articulación j a partir de
1805         theta;
1806         Eb=1; % Es 1 si la oscilación es hacia adelante y -1 si la oscilación es hacia
1807         atrás;
1808         K=1; % Se hace igual a uno en el primer paso y en el último paso;
1809         Tw = 2*pi; %Es el periodo de oscilación del desplazamiento angular
1810
1811
1812         theta12 = deg2rad(-100+10); %Es el ángulo de inclinación de la articulación j
1813         theta22 = 0.25; %Es la amplitud de la oscilación de la articulación j a partir de
1814         theta1;
1815         Eb2=-1; % Es 1 si la oscilación es hacia adelante y -1 si la oscilación es hacia
1816         atrás;
1817         K2=1; % Se hace igual a uno en el primer paso y en el último paso;
1818
1819     else
1820
1821         theta1 = deg2rad(-90); %Es el ángulo de inclinación de la articulación j
1822         theta2 = 0.25; %Es la amplitud de la oscilación de la articulación j a partir
1823         de theta1;

```

```

1824     Eb=-1; % Es 1 si la oscilación es hacia adelante y -1 si la oscilación es
1825     hacia atrás
1826     K=1; % Se hace igual a uno en el primer paso y en el último paso
1827     Ts = 2*T; %Es el periodo de ejecución del desplazamiento angular
1828
1829
1830     Bjin2 = deg2rad(-120); %Es el ángulo  $\alpha$  inicial de la articulación j
1831     Eb2 = 0.25; %Es la amplitud de la oscilación en la articulación j a
1832     partir de Bjin2
1833     Eb2=1; % Es 1 si la oscilación es hacia adelante y -1 si la oscilación es
1834     hacia atrás
1835     K2=1; % Se hace igual a uno en el primer paso y en el último paso
1836     Ts = 2*T; %Es el periodo de ejecución del desplazamiento angular
1837
1838 end
1839
1840
1841 % Desplazamiento del brazo fijo
1842
1843 if rem(j,2)==1
1844     Xfijo = (j*S)/2;
1845     Yfijo = 0;
1846     Zfijo = 0;
1847 else
1848     Xfijo = (j*S)/2;
1849     Yfijo = 0;
1850     Zfijo = 0;
1851 end
1852
1853
1854
1855 end
1856
1857
1858 % Graficar las trayectorias deseadas en X y Y de la cadera (para un
1859 % variación sinusoidal de posición, velocidad y aceleración de cada
1860 % articulación durante todo su recorrido.
1861
1862 if g'DesCad
1863     figure ('name','Trayectorias deseadas en X y Y de la cadera (para un
1864     ciclo de caminado)'); % Ahora figura 3
1865     subplot(3,2,1)
1866     plot(time, PosDeseadas(1) (:,1));
1867     xlabel('t');
1868     ylabel('Xc');
1869     subplot(3,2,3)
1870     plot(time, VelDeseadas(1) (:,1));
1871     xlabel('t');
1872     ylabel('Xvc');
1873     subplot(3,2,5)
1874     plot(time, AccDeseadas(1) (:,1));
1875     xlabel('t');
1876     ylabel('Xacc');
1877     subplot(3,2,2)

```

```

1878     plot (time, PosDeseadas{1}(:,2));
1879     xlabel('x');
1880     ylabel('Ee');
1881     subplot(3,2,4)
1882     plot (time, VelDeseadas{1}(:,2));
1883     xlabel('t');
1884     ylabel('Zee');
1885     subplot(3,2,5)
1886     plot (time, AccDeseadas{1}(:,2));
1887     xlabel('');
1888     ylabel('Zapp');
1889 end
1890
1891
1892 if qDesPle
1893     figure ('name', 'Traectorias deseadas en X y Y del pie (para un 100% de
1894     controlado)');
1895     subplot(3,2,1)
1896     plot (time, PosDeseadas{1}(:,3));
1897     xlabel('t');
1898     ylabel('Xdes');
1899     subplot(3,2,3)
1900     plot (time, VelDeseadas{1}(:,3));
1901     xlabel('t');
1902     ylabel('Zdesp');
1903     subplot(3,2,5)
1904     plot (time, AccDeseadas{1}(:,3));
1905     xlabel('');
1906     ylabel('Xdespp');
1907     subplot(3,2,2)
1908     plot (time, PosDeseadas{1}(:,4));
1909     xlabel('t');
1910     ylabel('Ydes');
1911     subplot(3,2,4)
1912     plot (time, VelDeseadas{1}(:,4));
1913     xlabel('t');
1914     ylabel('Zdesp');
1915     subplot(3,2,6)
1916     plot (time, AccDeseadas{1}(:,4));
1917     xlabel('t');
1918     ylabel('Zdespp');
1919 end
1920
1921
1922 if Vel_Articulares
1923     figure ('name', 'velocidades articulares (en cada paso)');
1924     for j=1:nPasos
1925         rand1 = rand;
1926         rand2 = rand;
1927         rand3 = rand;
1928         for i=1:6
1929             subplot(3,6,i)
1930             hold on
1931             plot (time, VelArticulares{j}(:,i), 'Color', [rand1 rand2 rand3]);
1932             xlabel('t');
1933             ylabel(['Vj' int2str(i)]);

```

```

1934         text(time:(2+(j*3+3)),VelArticulares(j):(2-(j*3-3),1),int2str(j));
1935     subplot(3,6,6-i)
1936     hold on
1937     plot(time,VelArticulares2(j):(1,6),'Color',[rand1 rand2 rand3]);
1938     xlabel('t');
1939     ylabel(['Velocidad pp_' int2str(6+i)]);
1940     text(time:(2+(j*3+3)),VelArticulares2(j):(2+(j*3+3),1),int2str(j));
1941     hold on
1942     end
1943     end
1944 end
1945
1946
1947
1948 if AcelArticulares
1949     figure('name','Aceleraciones articulares (en cada paso)');
1950     for j=1:nPasos
1951         rand4 = rand;
1952         rand5 = rand;
1953         rand6 = rand;
1954         for i=1:6
1955             subplot(4,6,i)
1956             hold on
1957             plot(time,AcelArticulares1(j):(1,6),'Color',[rand4 rand5 rand6]);
1958             xlabel('t');
1959             ylabel(['Aceleración pp_' int2str(i)]);
1960             text(time:(2+(j*3+3)),AcelArticulares1(j):(2-(j*3+3),1),int2str(j));
1961             subplot(4,6,6+i)
1962             hold on
1963
1964             plot(time,AcelArticulares2(j):(1,6),'Color',[rand4 rand5 rand6]);
1965             xlabel('t');
1966             ylabel(['Aceleración pp_' int2str(6+i)]);
1967             text(time:(2+(j*3+3)),AcelArticulares2(j):(2-(j*3-3),1),int2str(j));
1968             hold on
1969         end
1970     end
1971     end
1972 end
1973 end
1974
1975
1976 if yPosContinuas
1977     figure('name','Posiciones de cada eslabón durante TMO al recorrerlo');
1978     rand1 = rand;
1979     rand2 = rand;
1980     rand3 = rand;
1981     for i=1:6
1982         names = '03456';
1983         subplot(3,2,i)
1984         plot(time:delta,PosEslabones(i,1),'Color',[rand1 rand2 rand3]);
1985         xlabel('t');
1986         ylabel(['X.Robots_' names(i)]);
1987     end
1988 end

```



```

1989
1990
1991 if g'PosContinuas2
1992     figure ('name','Posiciones de cada celabón durante 2000 al recorrido');
1993     rand1 = rand;
1994     rand2 = rand;
1995     rand3 = rand;
1996     for i=1:6
1997         names = 'V88201112';
1998         subplot(3,2,i)
1999         plot(timeTotal,PosGlobales2(:,i),'b|o|', [rand1 rand2 rand3]);
2000         xlabel('t');
2001         ylabel(['Celabón_' names(i)]);
2002     end
2003 end
2004
2005

```