



Universidad Autónoma de San Luis Potosí

Facultad de Ingeniería

Centro de Investigación y Estudios de Posgrado

“Herramientas de análisis de arquitecturas para imitar el caminado humano”

T E S I S

Que para obtener el grado de:

Maestro en Ingeniería Mecánica.

Presenta:

Rodolfo Auristel López Palau.

Asesor:

Dr. Mauro Eduardo Maya Méndez

San Luis Potosí, S. L. P.

Diciembre de 2014



A mi familia

Con cariño dedico esta tesis a mi familia porque gracias a ellos yo soy lo que soy. A mi amada esposa Lía Natalli quien ha sido el impulso durante toda mi carrera y el pilar principal para la culminación de la misma, que con su apoyo constante y amor incondicional ha sido amiga y compañera inseparable, fuente de sabiduría, calma y consejo en todo momento.

Les agradezco a mis hijos Lía Dánae, Daniel Alejandro y Leonardo quienes son el motivo y la razón que me ha llevado a seguir superándome día a día, para alcanzar mis más apreciados ideales de superación. Ellos fueron quienes en los momentos más difíciles me dieron su amor y comprensión. Quiero también dejar a cada uno de ellos una enseñanza que cuando se quiere alcanzar algo en la vida, no hay tiempo ni obstáculo que lo impida para poderlo LOGRAR.

Agradecimientos

A lo largo de este proceso recibí el apoyo incondicional de innumerables personas e instituciones que me permitieron concluir satisfactoriamente este proyecto de investigación. En general quisiera agradecer a todos los que me brindaron un poco de sí, ya que sin su apoyo y la suma total de sus contribuciones no habría podido llevar a buen término esta etapa de mi vida. De corazón les estoy infinitamente agradecido.

Particularmente quisiera agradecer a CONACyT por el apoyo financiero otorgado a través de una beca de estudios, al posgrado en Ingeniería Mecánica de la Facultad de Ingeniería de la Universidad Autónoma de San Luis Potosí quien me abrió las puertas para llevar a cabo mi formación de maestría. Al Dr. Mauro Eduardo Maya Méndez le hago extensivo mi agradecimiento por la confianza y la atención que mostró en mi persona para dirigir esta investigación. A mis compañeros del posgrado les agradezco su tiempo y sus valiosos consejos, mi enorme gratitud a ustedes mis colegas por el permitirme desenvolverme dentro de este agradable ambiente de trabajo.

Finalmente, quisiera agradecer a mi familia toda la paciencia y comprensión que me han mostrado a lo largo de estos años. A mis padres Rodolfo y Elizabeth y a mis hermanas Nelly, Diana y Denisse; a mis suegros Juan Francisco, Maricela y mi cuñada Miriam les agradezco su tiempo y su apoyo incondicional durante todo este proceso. A mi esposa Lía y a mis hijos Danae, Daniel y Leonardo, gracias por darme el motivo más importante para superarme y ser mejor cada día, de ustedes son los frutos de mi esfuerzo y para ustedes mi eterna y más sincera gratitud.

Contenido

Contenido	1
Lista de tablas	4
Lista de figuras.....	5
Resumen.....	10
Introducción.....	11
Capítulo I	15
Antecedentes en locomoción bípeda	15
1.1. Análisis de la biomecánica del caminado humano.....	15
1.1.1. Análisis de la zancada	16
1.1.2. Planos anatómicos.....	19
1.1.3. Balanceo y apoyo	19
1.1.4. Cinemática de las articulaciones	20
1.1.5. Dimensiones del cuerpo humano.....	26
1.2. Arquitecturas robóticas	27
1.2.1. Historia de los robots humanoides.....	27
1.2.2. Robots humanoides	29
1.2.3. Robots bípedos comerciales.....	31
1.2.4. Investigaciones de robots bípedos de bajo costo	33
1.3. Plataformas de simulación robóticas	35
1.3.1. Ventajas y desventajas de la simulación	36
Capítulo II.....	40
Modelado del caminado humano	40
2.1. Solidworks.....	41
2.2. Propuesta para las extremidades inferiores del cuerpo humano.....	41
2.3. Cinemática de una arquitectura de 21 grados de libertad	45
2.3.1. Matrices de transformación homogénea	46
2.3.2. Representación de Denavit- Hartenberg.....	46
2.3.3. Análisis cinemático de una arquitectura de 21 grados de libertad.....	48
2.3.4. Parámetros de D-H de arquitectura propuesta.....	49
2.3.5. Comparación del estudio cinemático con modelo CAD.....	53

2.4.	Modelo dinámico de un robot serial	53
2.4.1.	Velocidades de las articulaciones.....	54
2.4.2.	Energía cinética	56
2.4.3.	Energía potencial	57
2.4.4.	Ecuaciones de movimiento	58
2.5.	Ejemplo 1 de aplicación de un manipulador (RR)	58
2.6.	Ejemplo 2 de aplicación un manipulador (RP).....	62
2.7.	Manipulador serial de n grados de libertad	65
2.8.	Desarrollo de un programa para obtener las ecuaciones dinámicas de un manipulador serial	66
Capítulo III.....		72
	Desarrollo de una plataforma de simulación de arquitecturas bípedas	72
3.1.	Herramientas para el desarrollo de la plataforma de simulación	73
3.1.1.	Matlab.....	75
3.1.2.	Simulink-SimMechanics.....	75
3.2.	Desarrollo de la plataforma	78
3.2.1.	Modelo en Simulink-SimMechanics de la arquitectura propuesta.....	78
3.2.2.	Movimiento de las articulaciones	80
3.2.3.	Datos para el movimiento de las articulaciones	83
3.2.4.	Exportar datos a Simulink.....	86
3.2.5.	Piso virtual.....	89
3.3.	Validación de plataforma de simulación de robots seriales.....	95
3.3.1.	Simulaciones de la plataforma con la arquitectura propuesta	100
3.3.2.	Gráficas de movimiento de arquitectura propuesta.....	101
Capítulo IV.....		107
	Validación experimental en un prototipo.....	107
4.1.	Robot bípedo de 6 grados de libertad.....	107
4.2.	Vitalización del prototipo.....	108
4.3.	Programa a bloques de robot de CICATA	110
4.4.	Propuesta de trayectorias para robot virtual de 6 grados de libertad.....	112
4.5.	Validación experimental	116
4.6.	ARDUINO	118

4.6.1. ARDUINO Mega 2560	118
4.7. Movimientos de robot de 6 grados de libertad	118
Conclusiones	121
Anexos	123
A-1. Función para calcular matriz D-H.....	123
A-2. Programa para calcular los valores de posición de cada eslabón de arquitectura de 21 grados de libertad.	124
A-3. Programa para importar datos de simulación de robot de CICATA y gráfica centro de masa.....	125
A-4. Programa para mover servomotores en Arduino.....	126
Referencias.....	127

Lista de tablas

Tabla 1-1	Ciclo del caminado.....	18
Tabla 1-2	Velocidades del caminado humano en diferentes investigaciones.....	17
Tabla 1-4	Robots construidos a lo largo de los años por KAIST.	31
Tabla 2-1	Eslabones utilizados para representar las extremidades inferiores del cuerpo humano.	43
Tabla 2-2	Parámetros D-H origen - pelvis.	50
Tabla 2-3	Parámetros D-H pelvis - cadera.....	50
Tabla 2-4	Parámetros D-H cadera - rodilla.....	51
Tabla 2-5	Parámetros D-H rodilla - tobillo.	51
Tabla 2-6	Parámetros D-H tobillo - (pie, talón).	52
Tabla 2-7	Parámetros D-H segunda cadena.....	52
Tabla 3-1	Prueba 1, Izquierda valores obtenidos en simulación, Derecha valores obtenidos utilizando las Ec.[2.10] y Ec.[2.11].	99
Tabla 3-2	Prueba 2, Izquierda valores obtenidos en simulación, Derecha valores obtenidos utilizando las Ec.[2.10] y Ec.[2.11].	99
Tabla 3-3	Prueba 3, Izquierda valores obtenidos en simulación, Derecha valores obtenidos utilizando las Ec.[2.10] y Ec.[2.11].	99

Lista de figuras

Fig. 1-1	Fases del caminado.....	16
Fig. 1-2	Longitud de paso.....	17
Fig. 1-3	Planos del cuerpo: sagital, coronal (frontal), horizontal.....	19
Fig. 1-4	Movimiento de pelvis en plano sagital durante un ciclo del caminado.....	21
Fig. 1-5	Movimiento de cadera en plano sagital durante un ciclo del caminado.....	21
Fig. 1-6	Movimiento de la rodilla en plano sagital durante un ciclo del caminado..	22
Fig. 1-7	Movimiento del tobillo en plano sagital durante un ciclo del caminado.....	22
Fig. 1-8	Movimiento de pelvis en plano frontal durante un ciclo del caminado.....	23
Fig. 1-9	Movimiento de la cadera en plano frontal durante un ciclo del caminado.	23
Fig. 1-10	Movimiento de la rodilla en plano frontal durante un ciclo del caminado..	24
Fig. 1-11	Movimiento del tobillo en plano frontal durante un ciclo del caminado	24
Fig. 1-12	Movimientos de la pelvis, fémur y tibia en el plano horizontal durante un ciclo del caminado.	25
Fig. 1-13	Medidas corporales.....	26
Fig. 1-14	Robot bípedo desarrollado por Shuuji Kajita en AIST.	28
Fig. 1-15	Simulación de robots bípedos que utilizaron Kajita y Tani durante sus estudios del caminado.....	29
Fig. 1-16	Robot bípedo Wabian-2 desarrollado en la universidad Japonesa de Waseda.....	29
Fig. 1-17	Robot ASIMO desarrollado por Honda.....	30
Fig. 1-18	Robot HUBO Y HUBO ALBERT una de las series de prototipos bípedos desarrollados por KAIST.	30
Fig. 1-19	Robot comercial Nao.....	31
Fig. 1-20	Robot comercial Kondo.	32
Fig. 1-21	Kit de robótica Bioloid.....	32
Fig. 1-22	Robots bípedos desarrollados por la universidad de Bonn.....	33
Fig. 1-23	Johnny Walker y Jack Daniels.	34
Fig. 1-24	Robot bípedo Andy Droid 2.....	34
Fig. 1-25	Robot bípedo Rock Steady.	35
Fig. 1-26	Entorno virtual para simulación de robots.....	36

Fig. 1-27	Plataforma de simulación Webots.....	37
Fig. 1-28	Plataforma de simulación V-rep.	38
Fig. 1-29	Ventana de simulación de RoboWorks.	38
Fig. 1-30	Ventana de simulación de Robostudio.....	39
Fig. 2-1	Ensamble realizado en Solidworks.....	41
Fig. 2-2	Representación de articulación esférica utilizando 3 actuadores cilíndricos.....	42
Fig. 2-3	Ensamble de la representación de las extremidades inferiores del cuerpo humano utilizando 21 grados de libertad.....	44
Fig. 2-4	Círculos utilizados para tener una referencia durante el ensamble.....	44
Fig. 2-5	Dimensiones (mm) de arquitectura propuesta.....	45
Fig. 2-6	Cadenas cinemáticas para arquitectura simplificada.	48
Fig. 2-7	Eje coordenado para cada articulación.....	49
Fig. 2-8	Sistema coordenado para cada eslabón.....	49
Fig. 2-9	Sistema coordenado origen - pelvis.	50
Fig. 2-10	Sistema coordenado pelvis-cadera.....	50
Fig. 2-11	Sistema coordenado cadera-rodilla.....	51
Fig. 2-12	Sistema coordenado rodilla -tobillo.	51
Fig. 2-13	Sistema coordenado tobillo – (pie, talón).....	52
Fig. 2-14	Carpeta que contiene los sensores dentro del árbol de operaciones de Solidworks.....	53
Fig. 2-15	Arquitectura serial de 2 grados de libertad (RR).....	59
Fig. 2-16	Arquitectura serial de 2 grados de libertad (RP).	62
Fig. 2-17	Parámetros necesarios en el programa para calcular la ecuación dinámica.....	66
Fig. 2-18	Calculo de parámetro Uij.....	67
Fig. 2-19	Calculo de parámetro Uijk.....	68
Fig. 2-20	Calculo de parámetro Dik.....	68
Fig. 2-21	Calculo de parámetro hikm.....	69
Fig. 2-22	Calculo de parámetro hi.	69
Fig. 2-23	Calculo de parámetro ci.....	70
Fig. 2-24	Manipulador de 2 grados de libertad usado en el segundo ejemplo para validar plataforma en análisis dinámico.....	71

Fig. 3-1	Diagrama a bloques de plataforma de simulación.	74
Fig. 3-2	Biblioteca de Simulink.	75
Fig. 3-3	Biblioteca de restricciones de Simmechanics.	76
Fig. 3-4	Biblioteca de articulaciones de Simmechanics.	77
Fig. 3-5	Biblioteca de sensores de Simmechanics.	77
Fig. 3-6	Interfaz Solidworks y Matlab.	78
Fig. 3-7	Programa a bloques en Simulink de la arquitectura propuesta.	79
Fig. 3-8	Derecha, representación virtual del ensamble realizado en Solidworks dentro del entorno de Simulink. Izquierda, ensamble en Solidworks que representa las piernas de un ser humano.	79
Fig. 3-9	Programa a bloques del tobillo izquierdo de la arquitectura propuesta. ...	80
Fig. 3-10	Bloque sensor de posición, velocidad y aceleración utilizado en el programa a bloques de la rodilla derecha.	81
Fig. 3-11	Derecha, sensores de posición de 4 puntos en el área que representa el pie derecho.	82
Fig. 3-12	Gráfica de posición del pie durante dos ciclos de caminado.	83
Fig. 3-13	Imagen importada en Solidworks que fue utilizada para obtener una referencia.	84
Fig. 3-14	Imagen importada en Solidworks, de movimiento de cadera derecha.	84
Fig. 3-15	Función de cadera derecha para obtener gráfica de movimiento de la articulación.	85
Fig. 3-16	Izquierda, imagen del movimiento de la articulación de la cadera derecha en el plano sagital, obtenida durante el estudio del caminado. Derecha, imagen digitalizada en Matlab del movimiento de la cadera derecha.	86
Fig. 3-17	Bloque <i>Signal Builder</i> utilizado para obtener movimientos continuos en la simulación en Matlab.	87
Fig. 3-18	Señal importada desde Matlab dentro de Simulink.	87
Fig. 3-19	Programa que llama las funciones que contienen los datos de posición de las articulaciones y los exporta al programa a bloques en Matlab.	88
Fig. 3-20	Sistema masa-resorte-amortiguador.	90
Fig. 3-21	Simulación de robot virtual con referencia de un piso.	91
Fig. 3-22	Sensor (body sensor) recopila información de simulación y guarda los datos en un archivo .mat.	91
Fig. 3-23	Sub bloque de control de posición en el eje Z.	92
Fig. 3-24	Actuador para bloque que aplica corrección de posición.	92

Fig. 3-25	Bloque que obtiene valor de la posición del pie y aplica una corrección cuando el pie pasa un límite de posición con respecto al piso.....	93
Fig. 3-26	Simulación de arquitectura de 21 grados de libertad.....	94
Fig. 3-27	Bloques agregados para el control de posición de piso y referencia de posición de x.....	94
Fig. 3-28	Modelo CAD de robot serial (RP).....	95
Fig. 3-29	Programa a bloques del robot serial (RP).....	96
Fig. 3-30	Movimiento de articulaciones prismática a la izquierda y de revolución a la derecha.....	97
Fig. 3-31	Simulación de manipulador serial (RP).	98
Fig. 3-32	Valores de la articulación prismática exportados desde el programa a bloques de Simulink hacia Workspace de Matlab.	98
Fig. 3-33	Gráfica de posición de los pies izquierdo y derecho.	100
Fig. 3-34	Simulación de movimiento de las articulaciones de la arquitectura simplificada.	101
Fig. 3-35	Gráfica de posición de los pies izquierdo y derecho.	102
Fig. 3-36	Gráficas de trayectorias de los pies izquierdo y derecho en el eje X.	103
Fig. 3-37	Gráficas de trayectorias de los pies izquierdo y derecho en el eje Y.	103
Fig. 3-38	Gráficas de trayectorias tridimensionales de los pies izquierdo y derecho.....	104
Fig. 3-39	Gráficas de trayectorias del centro de masa sobre cada plano.	105
Fig. 3-40	Gráfica de trayectoria tridimensional del centro de masa.	105
Fig. 3-41	Gráfica del área de los pies izquierdo, derecho y el centro de masa durante la simulación.	106
Fig. 4-1	Robot de 6 grados de libertad construido en CICATA-IPN.....	108
Fig. 4-2	Izquierda, Robot de 6 grados de libertad construido en CICATA-IPN. Derecha ensamble en Solidworks de robot.	109
Fig. 4-3	Izquierda, Robot de 6 grados de libertad construido en CICATA-IPN. Derecha ensamble en vista explosión en Solidworks.	109
Fig. 4-4	Programa a bloques en Simulink, de robot de 6 grados de libertad.....	110
Fig. 4-5	Sensores agregados para obtener datos de centro de masa y posición de pie derecho.....	111
Fig. 4-6	Derecha programa a bloques de robot de 6 grados de libertad. Izquierda ventana de simulación de robot 6 grados de libertad.....	111
Fig. 4-7	Área de soporte para el robot.	113

Fig. 4-8	Gráfica donde se representa la posición de los pies durante la simulación, el centro de masa se desplaza sobre el pie derecho.....	113
Fig. 4-9	Código de simulación de movimiento del robot.	114
Fig. 4-10	Robot moviendo el centro de masa hacia delante.	115
Fig. 4-11	Robot empieza a desplazar el centro de masa hacia el área de pie izquierdo.....	115
Fig. 4-12	Robot pasa el pie derecho hacia delante.	115
Fig. 4-13	El robot apoya los dos pies en el piso ($Z=0$) y el centro de masa regresa a la mitad del polígono de apoyo.....	116
Fig. 4-14	Datos utilizados para configurar rodilla, pie y cadera de ambas piernas del robot.....	117
Fig. 4-15	Gráfica obtenida de los datos utilizando Simulink, representa movimiento de pie derecho.....	117
Fig. 4-16	Plataforma Arduino Mega 2560.	118
Fig. 4-17	Conexión de los servomotores con Arduino.	119
Fig. 4-18	Conexión física del robot con Arduino.	119

Resumen

En este trabajo se realiza un estudio de la mecánica del caminado humano. Durante este análisis se investiga los grados de libertad implicados en la bipedestación. Cada movimiento de las articulaciones que contribuyen de manera más significativa al caminado humano es estudiado de manera detallada. A partir de este conocimiento se analizan y evalúan varias arquitecturas bípedas simplificadas que logren la tarea de caminado. Una de las arquitecturas bípedas estudiadas corresponde a la simplificación de las extremidades inferiores del cuerpo humano. Para ello se desarrolla una plataforma de simulación que permite la evaluación de diferentes arquitecturas bípedas en la tarea de caminado. Para la creación de la plataforma se utilizan dos programas comerciales conocidos ampliamente Matlab y Solidworks. La plataforma se desarrolla en un entorno de Simulink.

Adicionalmente se realiza una validación experimental preliminar en un prototipo bípedo simplificado de 6 grados de libertad. Este prototipo fue construido en el CICATA-IPN Querétaro.

Introducción

Antecedentes sobre robótica se hacen presentes desde hace varios siglos con la aparición de los primeros mecanismos articulados desarrollados por da Vinci y el desarrollo de máquinas programables utilizadas en la industria textil del siglo XVIII. Desde entonces y a la fecha, numerosos robots han sido realizados con distintas finalidades y diferentes configuraciones. Este es el caso de robots seriales que pueden soldar o pintar, otro caso son mecanismos capaces de aumentar la fuerza humana. También se han construido brazos y piernas robóticas para personas que han perdido una extremidad, en general, buscando imitar partes del cuerpo humano o actividades realizadas por el hombre. Estos esfuerzos, en las últimas décadas, han dado lugar de la creación de un gran número de robots para desarrollar distintas tareas y con variados niveles de complejidad [(Phil, y otros, 2007), (Pratt, y otros, 1998), (American Honda Motor Co. Inc., 2014), (Kawada Industries, INC., 2014) (Technische Universität München, 2009) (Nimbro, 2014)]. Una de las actividades que más ha llamado la atención de la comunidad científica ha sido el desarrollo de robots bípedos capaces de imitar la marcha o caminado humano. Especialmente, el área de biomecánica ha mostrado un interés en la comprensión del caminado humano por ejemplo para el desarrollo de prótesis humanas que permitan la rehabilitación del andar de los pacientes.

A pesar de que los robots bípedos han sido estudiados a lo largo de mucho tiempo, sólo durante los últimos 20 años, gracias al desarrollo de computadoras más veloces, los robots pueden realizar movimientos complejos. Sin embargo, la locomoción de estos robots aún está lejos de alcanzar los movimientos gráciles y la destreza observada en el caminado humano. La complejidad del caminado humano, no ha sido posible ser imitado por los robots bípedos. En la actualidad los científicos han desarrollado muy variados modelos matemáticos (Kajita, y otros, 2003), (Chenbo, y otros, 2014), (Kajita, y otros, 2010) de robots bípedos. En donde se modela la dinámica del robot o se proponen métodos alterno como *Zero Moment Point* (ZMP) (Kajita, y otros, 2003), (Kajita, 2010). Sin embargo, a pesar de sus considerables logros, estos métodos muestran varios inconvenientes como la fuerte dependencia del conocimiento completo del modelo dinámico, los altos costos de cómputo, y la vulnerabilidad a las perturbaciones desconocidas, entre otras. La complejidad del modelo dinámico también aumenta con el número de grados de libertad de los robots.

Una de las primeras investigaciones relevantes en robots bípedos fue la contribución de Ichiro Kato en la Universidad de Waseda con la serie Wabot (Humanoid Robotics Institute Copyright , 2014). En 1973, en la Universidad de Waseda en Japón, se desarrolló Wabot-1 que se convirtió en la primera serie de robots humanoides capaces de caminar. El Wabot-1 fue el primer robot desarrollado en el mundo antropomórfico de tamaño proporcional a un humano (Rauterberg, 2014).

Uno de los más famosos robots humanoides de la actualidad que puede caminar es ASIMO de Honda, el cual realizó su primera presentación en el año 2000. La capacidad de ASIMO ha sido demostrada a través de las habilidades de caminar y subir escaleras (American Honda Motor Co. Inc., 2014). En el 2003, la Universidad Técnica de Múnich creó a JOHNNIE un robot bípedo autónomo (Bräunl, 2014). Su objetivo principal fue construir unas máquinas que camina como el ser humano, con una caminata dinámica estable. También desarrolló un humanoide de 25 grados de libertad en total llamado LOLA, en 2009. El objetivo de este proyecto fue desarrollar un robot más rápido que fuera capaz de caminar como el ser humano (Technische Universität München, 2009).

Desde que la universidad de Waseda introdujo las series Wabot, muchos grupos de investigadores han tenido éxito en el desarrollo de robots bípedos humanoides. El grupo de investigadores de Honda desarrolló las plataformas humanoides P2, P3, y ASIMO (American Honda Motor Co. Inc., 2014). El Instituto Japonés de Ciencia Industrial Avanzada y Tecnológica (National Institute of Advanced Industrial Science and Technology) y las industrias Kawada han desarrollado el HRP-2P (Kawada Industries, INC., 2014). La Universidad de Tokio ha construido el H6 y H7 (kagami, 2001), El Instituto Superior Coreano de Ciencia y Tecnología (Korea Advanced Institute of Science and Technology) ha desarrollado el 41-DOF KHR-2 y HUBO (Hackel, 2007).

La tecnología de robots humanoides ha hecho un progreso importante en las últimas décadas, sin embargo nuestras expectativas siempre van más allá de la tecnología actual. Los seres humanos desean que los robots se hagan cargo de la totalidad de sus tareas diarias complicadas, sin embargo nuestra más avanzada tecnología de los robots humanoides solo es capaz de algunas tareas sencillas en entornos de laboratorios. Del mismo modo esperamos que los robots sean capaces de caminar y correr con la misma habilidad que los seres humanos, pero solo una pequeña parte del grupo de investigadores de robots bípedos, tiene la capacidad de crear robots humanoides de tamaño completo que pueden caminar y correr.

No obstante, algunos otros investigadores de robots bípedos, tienen una filosofía de diseño de robots humanoides de bajo costo. Esta clase de robots humanoides son similares a sus contrapartes más costosas en el sentido de que pueden ofrecer las capacidades para estudiar y mejorar nuevos algoritmos de caminado en humanoides, pero éstos son más asequibles. Sin embargo reducir el número de grados de libertad para que un robot sea barato necesita de un estudio exhaustivo. También el reducir los grados de libertad implica que el caminado que pueda realizar el robot no se parece al del ser humano.

Pese al esfuerzo de las últimas décadas de investigación, el problema de la locomoción bípeda mecánica aún está lejos de ser resuelto. A pesar de que en la actualidad existen robots humanoides con un número elevado de grados de libertad, existen a la fecha una gran variedad de configuraciones que no se ha estudiado, debido a que los grados de libertad que tiene el ser humano son mucho mayores que los robots construidos en la actualidad. Algunas preguntas que permanecen abiertas son: ¿qué pasa con los demás grados de libertad que tiene el ser humano y que no han sido utilizados en la construcción de los robots más complejos?, ¿en qué puede ayudar estudiar estas diferentes configuraciones?, ¿qué pueden hacer las demás configuraciones? Una forma de contestar estas preguntas es a través del análisis del caminado bípedo de robots con distintas configuraciones. Este tipo de análisis puede ser una herramienta de estudio de diferentes arquitecturas, puede ayudar en la concepción de diseños simplificados de arquitecturas bípedas. También nos podría ayudar a comprender los principios fundamentales tras la locomoción humana. Sin embargo, para realizar varios análisis de todas las arquitecturas posibles necesitaría un estudio cinemático directo e inverso y también un estudio dinámico de un robot para cada caso. Realizar este tipo de análisis es complicado y tardado (Muhammad, y otros, 2010), (K., 2012), (De Lope, y otros, 2003), algunos investigadores de robots bípedos realizan aproximaciones de un péndulo invertido (Edgar, y otros, 2010) para facilitar la aproximación de la dinámica del robot, inclusive también se manejan redes neuronales para encontrar la cinemática inversa (Gómez, 2012). Sería deseable una herramienta que permitiera simplificar los estudios anteriores. Una posibilidad prometedora es una herramienta de simulación que sea capaz de reproducir lo más cercano posible a la realidad los fenómenos deseados y tener accesible información acerca de ellos (tales como parámetros y variables).

Según la necesidad detectada previamente en este trabajo de tesis se busca desarrollar herramientas de análisis de arquitecturas de caminado humano. Con base en una revisión detallada de la biomecánica del caminado humano, topologías con distintos grados de libertad son analizadas y simuladas. En este sentido los objetivos tanto generales como particulares de esta tesis quedan definidos de la siguiente manera:

Objetivo General: Desarrollar herramientas de análisis del caminado humano y evaluar algunas arquitecturas bípedas simples para realizar tareas de caminado.

Objetivos Particulares:

- Analizar el caminado humano e identificar los grados de libertad asociados.
- Desarrollo de una plataforma de simulación.
- Análisis y simulación de arquitecturas simples de robots bípedos que imitan el caminado humano.
- Comprobación del análisis del caminado con un prototipo de laboratorio.

La organización de la tesis queda de la siguiente manera, en el Capítulo I se realizará una breve introducción al caminado humano y una revisión a los diferentes prototipos de robots bípedos existentes, en el Capítulo II se describe el modelado de la parte inferior del cuerpo humano y un análisis cinemático de la arquitectura simplificada así como algunos estudios dinámicos de robots seriales, en el Capítulo III se muestra el desarrollo de la plataforma de simulación, en el Capítulo IV se realiza una validación experimental preliminar utilizando un robot físico de 6 grados de libertad construido en el CICATA-IPN, finalmente se muestran las conclusiones obtenidas a lo largo del desarrollo de la plataforma y pruebas preliminares.

Capítulo I

Antecedentes en locomoción bípeda

Durante el capítulo estudiaremos la cinemática de las articulaciones de las extremidades inferiores del cuerpo humano. Se analizarán los movimientos todas las articulaciones que contribuyen al caminado. Los movimientos de las articulaciones se analizarán en sus diferentes planos corporales.

Enseguida se mencionarán algunos de los trabajos más importantes de robots bípedos. Estudiando las más sobresalientes investigaciones en arquitecturas bípedas. También se analizarán los grados de libertad de cada uno de los robots bípedos.

Con los datos recolectados de la literatura sobre el caminado humano, se pretende imitar el movimiento de las piernas con alguna arquitectura robótica virtual. Para lo cual se necesita estudiar las arquitecturas robóticas que existen en la actualidad.

Realizar pruebas en prototipos de robots bípedos es tardado, complicado y costoso. Por lo tanto se estudiarán algunas de las plataformas de simulación de arquitecturas bípedas más sobresalientes en la actualidad.

1.1. Análisis de la biomecánica del caminado humano

El caminar humano es una necesidad básica que nos sirve para trasladarnos. Esta actividad es una de las que más comúnmente hacemos los seres humanos.

Años de práctica nos proveen de la destreza de poder realizar la acción de caminar mientras hacemos diferentes actividades, como mirar hacia un lado, conversar con otra persona o incluso evadir obstáculo que se nos presentan. A temprana edad empezamos a aprender cómo pararnos y caminar. De hecho alrededor de los 7 años es cuando maduramos la actividad de caminar pareciéndose a la de los adultos.

Los hermanos Weber Brothers publicaron unos de los primeros trabajos científicos del caminado en 1836 (Rainone, y otros, 2008). Ellos usaron instrumentos como un cronómetro y un telescopio con una escala para medir. Durante su trabajo definieron los elementos básicos del ciclo del caminado como fase de balanceo, fase de apoyo, fase de doble soporte, muchos de estos términos se siguen utilizando en la actualidad.

En el siglo XIX existieron muchos investigadores interesados en el caminado humano. Unos de los investigadores más reconocidos fueron Marey y Vierordt (Huei-Ming, 2007) quienes usaron ingeniosa tecnología para expandir nuestro conocimiento acerca del caminado. Por ejemplo usaron una recámara de aire para grabar los movimientos del pie

durante el caminado. Los estudios hechos fueron los primeros que ayudaron a evaluar, planificar y tratar a las personas con condiciones que afectan su capacidad para caminar.

Uno de los más grandes avances en el campo fue cinematografía. Con esta tecnología se estudiaron los patrones de la cinemática del caminado humano y animal. Muybridge (Stanford, 2014) puede ser el más conocido debido secuencias de movimientos fotografiados en 1872 de un caballo corriendo. En este trabajo se observó que durante un periodo muy corto de tiempo todas las extremidades se encontraban en el aire.

Inicialmente el análisis del caminado era en una dimensión en el plano sagital. Braúne en 1895 experimentó con el caminado en tres dimensiones usando 4 cámaras un par a cada lado del cuerpo. Al cuerpo estaban pegados tubos de luz para poder ver los movimientos producidos.

En la actualidad los avances tecnológicos nos han permitido tener un desarrollo mayor en cuanto a la calidad del análisis del caminado.

1.1.1. Análisis de la zancada

El ciclo del caminado comprende el periodo de tiempo de un paso desde el talón de ataque (cualquier talón que toca primero el suelo) hasta que el mismo talón vuelve a tocar el mismo psio como se muestra en la Fig. 1-1 en donde toca primero el pie derecho el suelo (pie de color negro). El 0% del ciclo comienza con el contacto del talón en el suelo de cualquiera de los dos pies y el 100% del ciclo se completa en cuanto el talón del mismo pie vuelve a tocar el suelo.

Una zancada es una secuencia de eventos (Fig. 1-1) tomando como referencia el contacto del talón. La zancada se completa cuando el talón que tocó primero el piso ya sea izquierdo o derecho completa el ciclo volviendo a tocar el suelo. Un paso ocurre entre el contacto del talón del pie contrario. Por ejemplo entre el contacto del talón del pie izquierdo y el paso finalizaría con el contacto del talón del pie derecho.

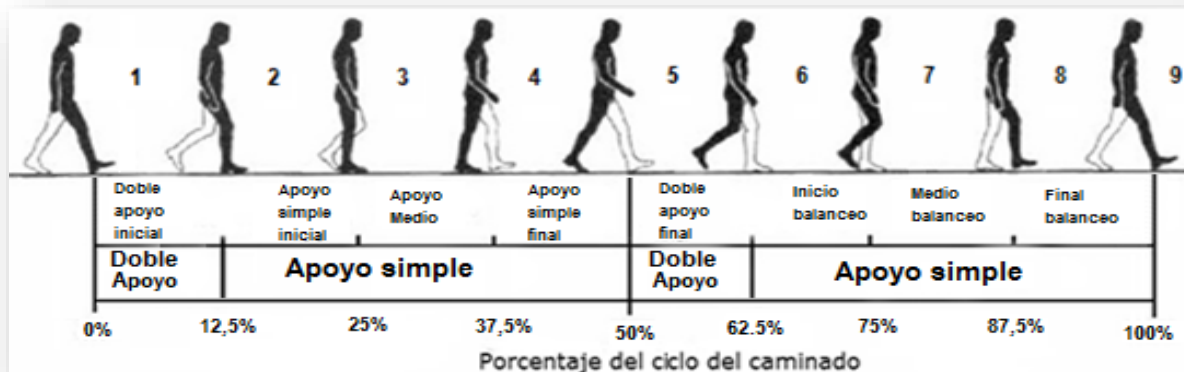


Fig. 1-1 Fases del caminado.

La descripción más básica del ciclo del caminado incluye el ancho de la zancada y la longitud del paso como se muestra en la Fig. 1-2.



Fig. 1-2 Longitud de paso.

En una persona sana la longitud del paso es de aproximadamente 72 cm como lo muestra la Fig. 1-2 anterior. Esta medida tiene como referencia la distancia entre talón del pie izquierdo y el talón del pie derecho cuando las piernas están abiertas a máxima amplitud mientras se camina.

Una de las medidas básicas para la cadencia del caminado son los pasos por minuto, que también es llamado velocidad de paso (step rate). Otra de las medidas importantes es el tiempo del ciclo completo del caminado (stride time). Esta última medida se puede derivar de los pasos por minuto.

Velocidad de caminado es la distancia recorrida en un tiempo unitario. Esta medida por lo general se mide en metros por segundo (m/s). La velocidad de caminado puede variar de una persona a otra dependiendo de su edad, estatura, género, entre otras características.

La velocidad promedio del caminado de una persona promedio de 1.80 mts toma alrededor de 1.37 m/s como se muestra en la Tabla 1-1.

	Drillis (1961) (Neumann, 2008)	Molen (1973) (Neumann, 2008)	Finley and Cody (1970) (Neumann, 2008)
Velocidad de caminado (m/s)	1.46	1.39 (hombres) 1.37 (mujeres)	1.37(hombres) 1.24(mujeres)
Velocidad de paso (Pasos/s)	1.9	1.99(hombres) 1.88(mujeres)	1.84(hombres) 1.94(mujeres)
Longitud de paso (m)	0.76	0.77(hombres) 0.67(mujeres)	0.74(hombres) 0.63(mujeres)

Tabla 1-1 Velocidades del caminado humano en diferentes investigaciones.

Ciclo del caminado.

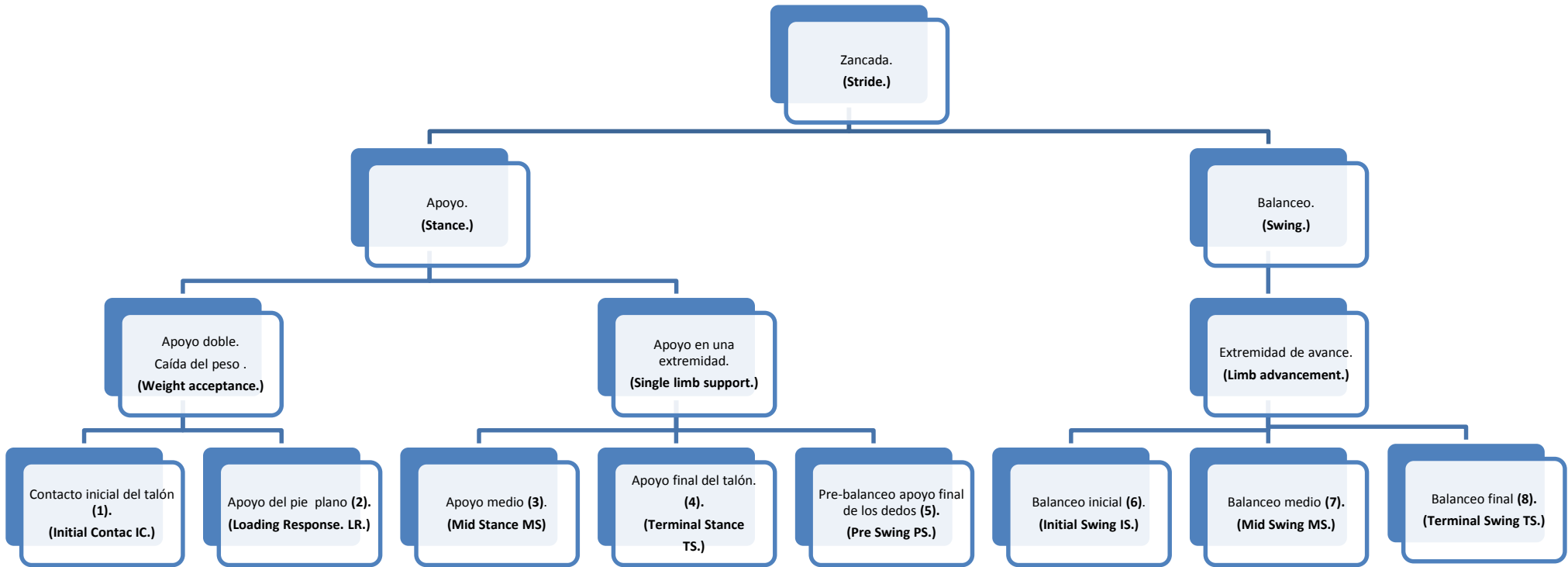


Tabla 1-2 Ciclo del caminado

Hay varias estrategias para aumentar la velocidad de caminado incrementando la zancada o incrementando la cadencia. Sin embargo la más utilizada es aumentar la cadencia.

1.1.2. Planos anatómicos

Para describir al cuerpo humano se debe colocar éste en la llamada posición anatómica (ver Fig. 1-3) el cuerpo en posición erecta, los pies dirigidos al frente y planos sobre el piso, la vista dirigida hacia el horizonte, los brazos a los lados del tronco, con las palmas de las manos dirigidas hacia adelante. Los planos anatómicos son cortes imaginarios que se le practican al cuerpo o a una parte de él; éstos son perpendiculares entre sí y permiten describir la localización de los distintos componentes corporales. La terminología empleada es la Nomenclatura Internacional (anatolandia, 2014).

La posición anatómica mostrada Fig. 1-4 también nos sirve como referencia para determinar los grados que tiene una articulación con respecto a otra, tomando como 0° los grados entre las articulaciones.

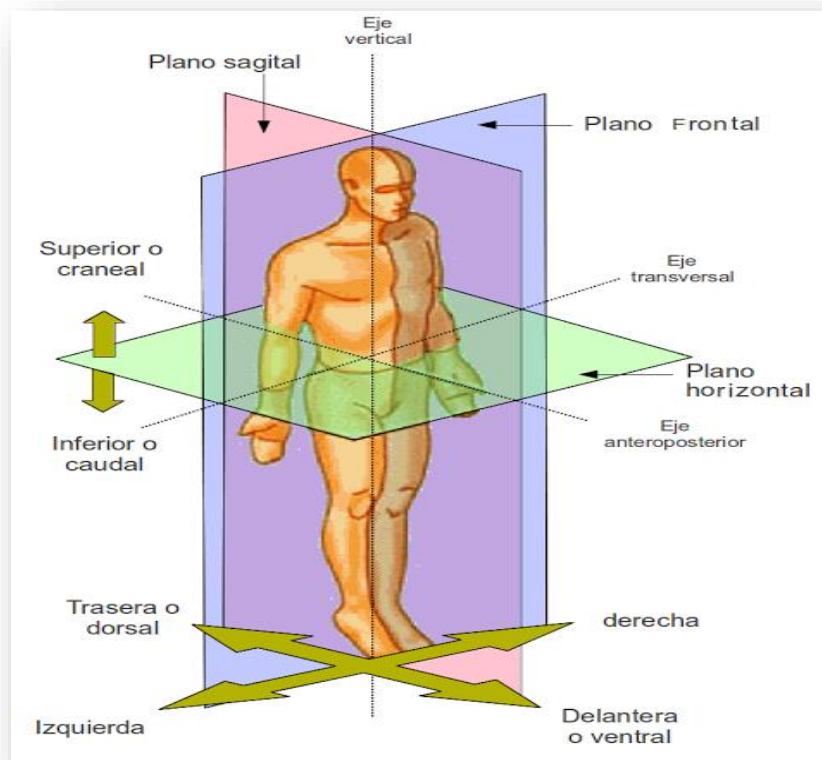


Fig. 1-3 Planos del cuerpo: sagital, coronal (frontal), horizontal.

1.1.3. Balanceo y apoyo

El ciclo de caminado se pueden dividir en 2 fases principales como se muestra en la Fig. 1-1 fase de apoyo y fase de balanceo. La fase de apoyo ocurre cuando el talón del pie derecho toca el suelo y acaba cuando los dedos de ese mismo pie dejan el suelo. Durante esta fase el peso del cuerpo se apoya sobre la extremidad. La fase de balanceo ocurre cuando los dedos del pie derecho dejan el suelo, hasta un

momento antes de que el talón del mismo pie vuelva a contactar el suelo. Durante esta fase el pie se encuentra en el aire y avanzando hacia adelante. La fase de apoyo ocupa el 62% del ciclo mientras que la fase de balanceo ocupa el 38% del ciclo como se muestra en la Fig. 1-1.

Entre cada ciclo del caminado se experimentan dos periodos de doble soporte (cuando ambos pies están sobre el suelo) y dos periodos de soporte simple (cuando un solo pie se encuentra en el suelo) como se muestra en la Fig. 1-1.

En el primer periodo, que ocurre del 0 al 50% del ciclo del caminado, el tobillo de la pierna derecha toca el suelo. Durante este tiempo el peso del cuerpo se empieza a trasladar de un pie al otro. Después la extremidad izquierda comienza en el periodo de soporte simple entre el 50 y 60% del ciclo. Finalmente entre el 60 y el 100% del ciclo se encuentra en soporte simple pero en el pie izquierdo.

Desplazamiento y control del centro de masa del cuerpo.

Caminar puede ser definido como una serie pérdida y recuperación del balance. Para prevenir la falla en la recuperación del balance se mueve cualquiera de las extremidades a una nueva locación. Una vez que empieza la caminata el centro de masa es llevado a una nueva locación por cualquiera de los pies. La caminata hacia adelante es lograda por la sucesión de recolocación de las extremidades.

En el movimiento del centro de masa se encuentran dos ondas sinusoidales una corresponde al plano sagital y el otro corresponde al plano coronal. La mínima altura del centro de masa en el plano sagital ocurre en ambos periodos de doble apoyo alrededor del 5 y el 55% del ciclo como se muestra en la Fig. 1-1. La máxima altura del centro de masa ocurre en el periodo de apoyo medio en 30 y el 80% del ciclo. El desplazamiento máximo es de alrededor de 5cm en el plano sagital. El movimiento del centro de masa también ocurre de un lado a otro en el plano coronal. Este movimiento también describe una onda sinusoidal con una amplitud de 4 cm máximo. Durante este movimiento el centro de masa es cambiado de una extremidad a la otra. La máxima posición del centro de masa en el lado derecho ocurre en apoyo medio de la extremidad derecha en el 30% del ciclo y la máxima posición alcanzada por el centro de masa en la posición derecha ocurre en el 80% del ciclo.

1.1.4. Cinemática de las articulaciones

Durante el ciclo del caminado el centro de masa es movido por la acción de la rotación de las articulaciones. Primero ocurre un movimiento importante de las articulaciones en el plano sagital. Un instante después ocurre un segundo movimiento en el plano coronal (Neumann, 2008).

A continuación se describe el movimiento cinemático de las articulaciones en los planos corporales. Primero se analiza el movimiento de todas las articulaciones en el plano sagital, a continuación en el plano frontal y por último el movimiento en el plano horizontal.

Cinemática del plano sagital.

Pelvis. El movimiento de la pelvis es pequeño y es descrito como el movimiento de toda la estructura ósea que sostiene a los órganos. Este movimiento es descrito como la inclinación anterior y posterior del eje medio- lateral. La posición neutral Fig. 1-3 de la pelvis en 0° definida como la posición de la pelvis relajada.

Durante el caminado el movimiento de la pelvis es pequeño alrededor de 2 a 4° . Para el caminado la pelvis se toma como un solo eslabón. La trayectoria del movimiento de la pelvis durante el ciclo completo del caminado se muestra en la Fig. 1-4. En la Fig. 1-4 se muestra la gráfica el movimiento de la pelvis durante un ciclo del caminado humano. En donde 0% - 100% representa una zancada.

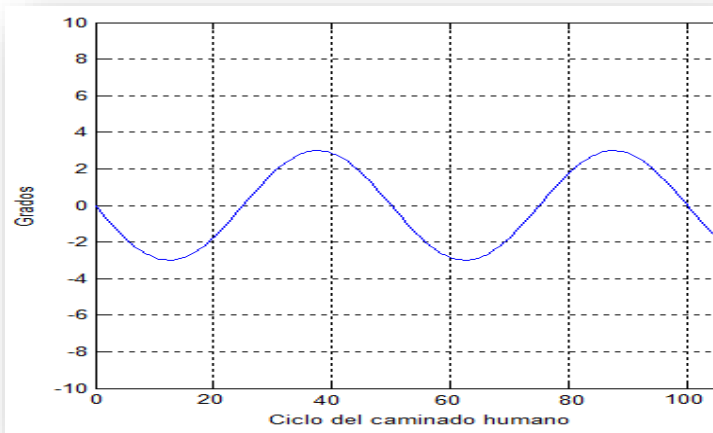


Fig. 1-4 Movimiento de pelvis en plano sagital durante un ciclo del caminado.

Cadera. Cuando el talón toca el suelo por primera vez la cadera tiene alrededor de 30° . Cuando el cuerpo se mueve sobre el pie de apoyo la cadera tiene su máxima extensión alrededor de 10° . La cadera se encuentra cerca de 0° alrededor del 60% del ciclo del caminado humano. Durante la fase de balanceo la cadera se flexiona para brindar a la extremidad inferior el siguiente movimiento de pie necesario para el caminar. En la Fig. 1-5 se muestra la trayectoria de la cadera durante el ciclo del caminado.

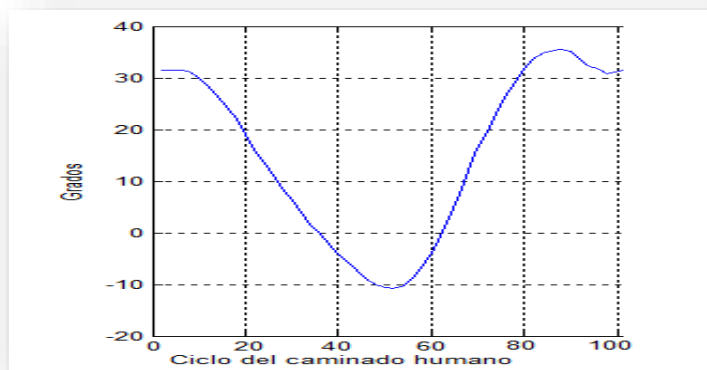


Fig. 1-5 Movimiento de cadera en plano sagital durante un ciclo del caminado.

Rodilla. Cuando el talón toca el suelo la rodilla está flexionada 5° y se sigue flexionando entre 10° y 15° durante el 15% del ciclo del caminado como se muestra en la Fig. 1-6. La rodilla alcanza casi su extensión máxima cuando el talón deja el suelo alrededor del 40% del ciclo. En este punto la rodilla comienza a flexionarse 35° para el momento en el que los dedos dejan el suelo alrededor de 60% del ciclo. Máxima extensión de la rodilla es de 60° está extensión se presenta durante la fase de balanceo medio alrededor del 73% del ciclo.

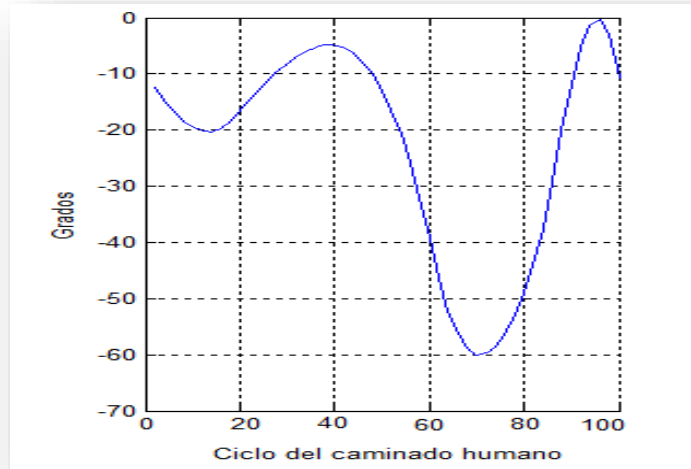


Fig. 1-6 Movimiento de la rodilla en plano sagital durante un ciclo del caminado.

Tobillo. Cuando el talón toca el suelo el tobillo empieza con una flexión plantar entre 0 y 5° en la Fig. 1-7 se muestra las trayectorias del tobillo durante el ciclo del caminado. Después del contacto del talón entre el 8% del ciclo del caminado el pie se encuentra completamente plano en el suelo. Poco después de que el talón deja el suelo en el 40% del ciclo el tobillo alcanza un máximo de 15° a 20° . Durante la fase de balanceo el tobillo se encuentra en una posición neutral permitiendo dejar los dedos el piso limpiamente.

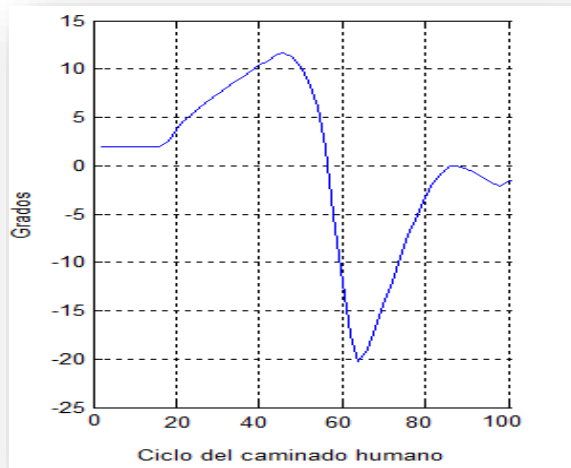


Fig. 1-7 Movimiento del tobillo en plano sagital durante un ciclo del caminado.

Cinemática en el plano frontal.

Pelvis. En el plano frontal la pelvis rota alrededor de 10 a 15°, durante la caída del peso en el pie derecho entre el 15% y el 20% del ciclo. En este tiempo la pelvis cae mientras el pie izquierdo se balancea. En la Fig. 1-8 se muestra el movimiento de la pelvis a lo largo de un ciclo del caminado.

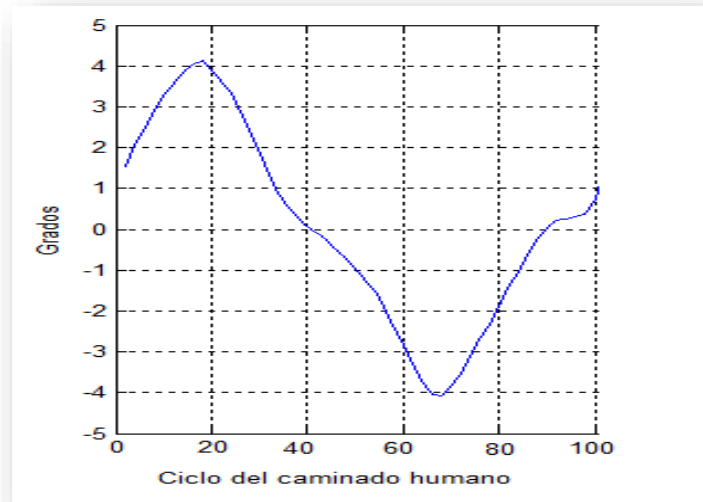


Fig. 1-8 Movimiento de pelvis en plano frontal durante un ciclo del caminado.

Cadera. Durante el caminado humano el movimiento de la cadera tiene un movimiento máximo de 6° alrededor del 20% del ciclo así como un movimiento de -6° alrededor del 70% del ciclo en el plano frontal. En la Fig. 1-9 se muestra la trayectoria que sigue la cadera en el plano frontal durante un ciclo de caminado.

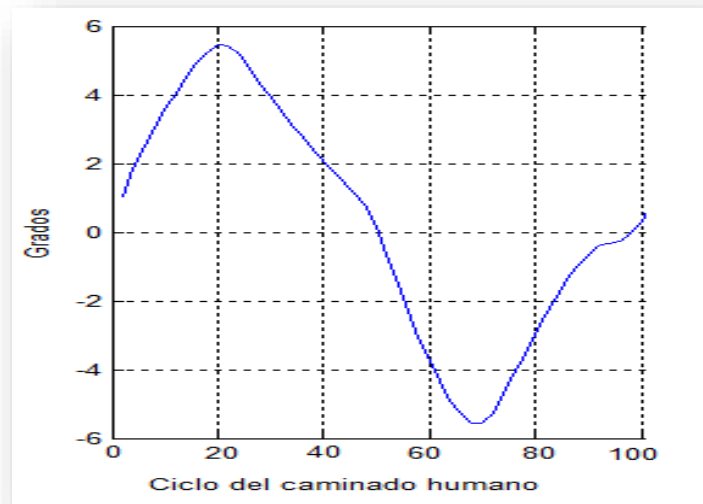


Fig. 1-9 Movimiento de la cadera en plano frontal durante un ciclo del caminado.

Rodilla. En el plano frontal la rodilla es muy estable sin embargo esta articulación presenta un pequeño movimiento con un máximo de -6° alrededor del 80% del ciclo del caminado como se muestra en la Fig. 1-10.

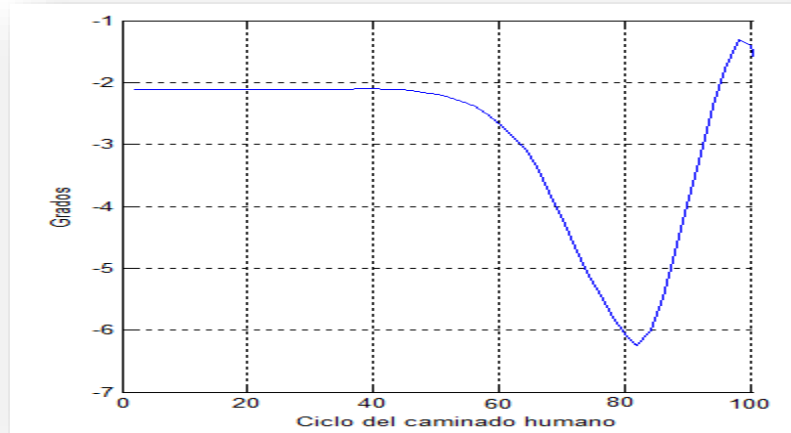


Fig. 1-10 Movimiento de la rodilla en plano frontal durante un ciclo del caminado.

Tobillo. El movimiento de la articulación en el plano frontal es muy pequeño como se muestra en la Fig. 1-11 pero se toma en cuenta para el estudio.

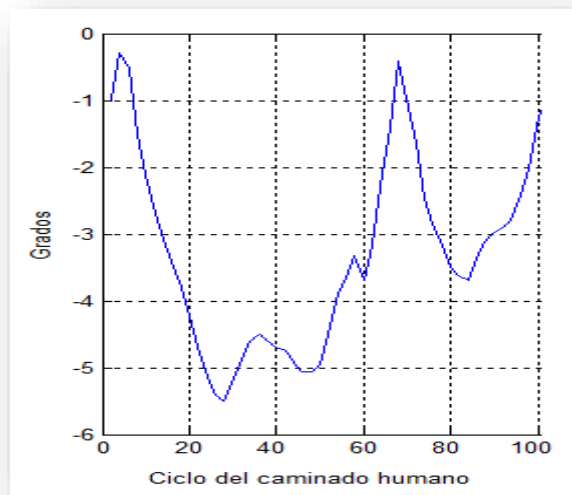


Fig. 1-11 Movimiento del tobillo en plano frontal durante un ciclo del caminado

Cinemática del plano horizontal

Hay muy pocos estudios del movimiento de las articulaciones en el plano horizontal. Sin embargo hay algunas articulaciones que tiene un movimiento significativo durante el caminado.

Pelvis. Durante el caminado la pelvis rota en el plano horizontal a través de un eje vertical que pasa por la articulación de la cadera del pie que se encuentra en la fase de apoyo. Cuando el talón derecho toca el suelo, en el 15 a 20% del ciclo la pelvis gira en sentido anti horario como se muestra en la Fig. 1-12.

Fémur. Después del contacto del talón, el fémur rota internamente en los primeros 15 a 20% del ciclo revierte su rotación y rota externamente durante el corto tiempo en que se despegan los dedos del suelo. El fémur rota aproximadamente 6 a 7° en cada dirección durante el caminado.

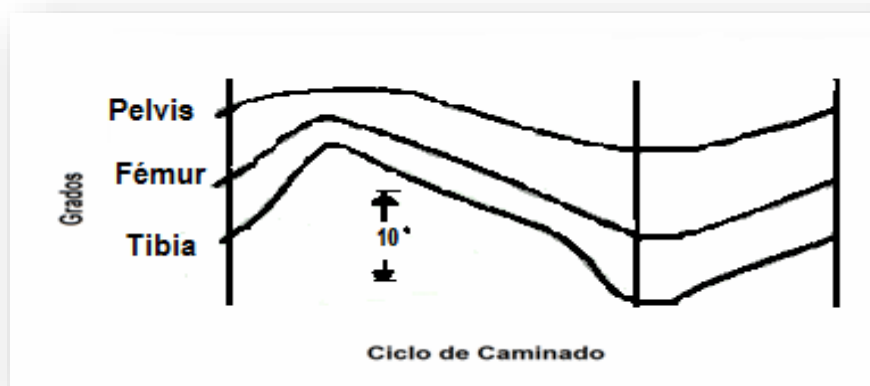


Fig. 1-12 Movimientos de la pelvis, fémur y tibia en el plano horizontal durante un ciclo del caminado.

Tibia. La trayectoria que describe el movimiento de la tibia es muy similar al movimiento descrito por el fémur. En la Fig. 1-12 se muestra esta trayectoria. La magnitud del movimiento en cada dirección es de 8 a 9°.

Cadera. Una rotación interna neta del movimiento de la cadera derecha ocurre durante la mayor parte de la fase de apoyo de la extremidad inferior derecha. La máxima rotación interna es alcanzada durante en el 50% del ciclo. La rotación externa de la cadera derecha ocurre del 50% del ciclo hasta la fase de balanceo medio.

Rodilla. En el tiempo en el que el talón toca el piso, la rodilla se encuentra a 2 a 3° de rotación externa.

Tobillo y pie. Los movimientos que realizan el tobillo y el pie son muy pequeños de manera que no se tomarán en cuenta para el estudio del caminado.

Caminado humano normal

La locomoción humana normal se ha descrito como una serie de movimientos alternantes, rítmicos, de las extremidades y del tronco que determinan un desplazamiento hacia delante del centro de gravedad. Más específicamente, la locomoción humana normal puede describirse enumerando algunas de sus características (Perry, 1992).

Requisitos del caminado humano normal.

- Paso libre de un pie durante el caminado.
- Paso libre del pie durante el balanceo.
- Posición adecuada del pie.
- Longitud adecuada del paso.

1.1.5. Dimensiones del cuerpo humano

Las medidas del cuerpo humano, ya sea en reposo o en movimiento están determinadas por el largo de los huesos, las capas musculares y la mecánica de las articulaciones. Para poder proponer un buen modelo es necesario el conocimiento de las medidas más importantes del cuerpo humano y las extensiones de los movimientos de las manos, brazos, piernas y pies. Las dimensiones y proporciones del cuerpo humano son muy diferentes de una persona a otra varían de acuerdo a qué población se estudia, masculina o femenina.

Las posturas naturales, es decir, las posiciones del tronco, de los brazos y de las piernas que no generen esfuerzos estáticos, y los movimientos naturales son tomadas en cuenta para construir un puesto de trabajo (escritorio, pupitre, mesa de trabajo, etc.). Es por lo tanto imprescindible que el puesto de trabajo se adapte a las dimensiones del trabajador. “Lo antedicho genera una serie de problemas importantes, la gran variedad de estaturas de cada sexo y las diferencias corporales entre ambos sexos, además hay que agregar también las diferencias existentes entre las distintas razas, habitantes de zonas diferentes, (llanura, montaña)” (Melo, 2014).

Las dimensiones y medidas del cuerpo son muy diferentes de una persona a otra. Por esta razón se optó por seguir las dimensiones estándar mostradas en la Tabla 1-3, en donde se presentan las principales medidas del cuerpo humano, según la Norma DIN 33.402 segunda parte. Se dan en ella las medidas media aritmética como también los valores límites, (superior e inferior). Dichos valores límite han sido establecidos de manera tal que solo el 5% de las personas sometidas a estudio evidencian valores mayores al límite superior y otro 5% acusan valores menores al límite inferior Fig. 1-13.

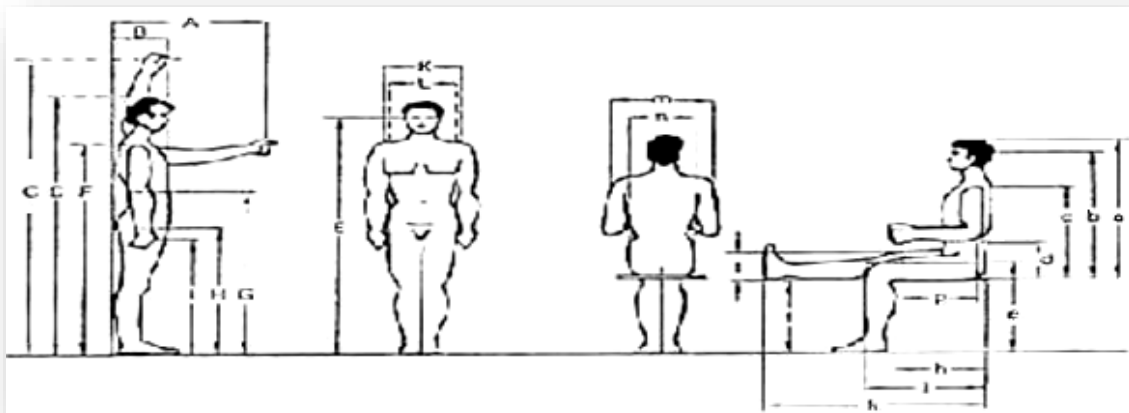


Fig. 1-13 Medidas corporales.

En la Tabla 1-3 se muestran las medidas entre articulaciones.

Designación.	Hombres.			Mujeres.		
Posición erguida.						
A-Alcance hacia delante.	662	722	787	616	690	762
B-Profundidad del cuerpo parado.	233	276	318	238	285	357
C-Alcance hacia arriba.	1910	2051	2210	1748	1870	2000
D-Estatura.	1629	1733	1841	1510	1619	1725
E-Altura de los ojos parado.	1509	1613	1721	1402	1502	1596
F-Altura de los hombros.	1349	1445	1542	1234	1339	1436
G-Altura de los codos desde el piso.	1021	1096	1179	957	1030	1100
H-Altura entre piernas.	752	816	908	-	-	-
I-Altura de las manos.	728	767	828	664	738	803
K-Ancho de los hombros.	367	398	428	323	355	388
L-Ancho de cadera.	310	344	368	314	358	405
Posición sentado.						
a-Altura del cuerpo desde el asiento.	849	907	962	805	857	914
b-Altura de los ojos desde el asiento.	739	790	844	680	735	785
c-Altura de los hombros.	561	610	655	538	585	631
d-Altura de los ojos desde el asiento.	193	230	280	191	233	278
e-Altura de las rodillas.	493	535	574	462	500	542
f-Largo de pantorrilla a pie.	399	442	480	351	395	434
g-Distancia de codo a pie de agarre.	327	362	389	292	322	364
h-Profundidad del cuerpo sentado.	452	500	552	426	484	532
i-Distancia nalga rodilla.	554	559	645	530	587	631
k-Distancia nalga pie.	964	1035	1125	955	1044	1126
l-Espesor del muslo.	117	136	157	118	144	173
m-Ancho sobre los codos.	399	451	512	370	456	544
n-Ancho de asiento.	325	362	391	340	387	451

Tabla 1-3 Tabla de medidas corporales.

En esta sección se estudió el movimiento que realizan las articulaciones durante el caminado humano. Se analizaron los movimientos de las piernas en el plano sagital, frontal y horizontal. También se observaron las medidas anatómicas del cuerpo humano.

1.2. Arquitecturas robóticas

1.2.1. Historia de los robots humanoides

La locomoción en el ser humano es el resultado de una evolución de varios millones de años. Cuando nos movemos de un lugar a otro realizamos movimientos de brazos y piernas sin siquiera pensarlo, a menudo necesitamos escoger solamente el destino del lugar a donde queremos llegar y nuestro sistema locomotor nos lleva a ese lugar. De manera inconsciente podemos caminar y a la par realizar otros movimientos. Los seres humanos tomamos este proceso tan natural que no se había prestado especial atención a este tema hasta principios del siglo XIX como se mencionó en el tema anterior. Con la ayuda de estos estudios las personas tenían mejores herramientas para imitar de manera fiel el movimiento de alguna extremidad. Por ejemplo para remplazar una pierna perdida. Sin embargo la construcción de máquinas o dispositivos que imitan el movimiento humano tiene una larga historia.

La construcción de robots tiene una larga historia. Uno de los primeros robots documentados fueron los de Leonardo da Vinci. En el siglo XVI Leonardo da Vinci diseñó una máquina parecida a un humano desafortunadamente solamente se mantuvo en diseño ya que nunca fue construida (Allonrobots, 2013). El francés Jacques de Vaucanson (Dalakov, 2014) entre los siglos XVII y XVIII construyó tres mecanismos humanoides, uno tocaba el mandolín y cantaba otro tocaba el piano y simulaba la respiración y el tercero era flautista estos mecanismos no podían interactuar con el ambiente. La mayoría de los robots construidos anteriores a la computadora fueron mecanismos que no podían interactuar con las personas eran pre-diseñados para realizar algunos movimientos o eran manejados por personas. También existen trabajos que se enfocan en el análisis del caminado pasivo (F., 1888) en donde no hay actuadores ni computadoras. Otro trabajo es el realizado por Steve Collins y sus colegas (Collins, y otros, 2005) en el que crearon una serie de robots articulados que se deslizan por una rampa. La única energía utilizada para realizar el movimiento es la gravedad.

Shuuji Kajita (Kajita, y otros, 1992) investigador en el *National Institute of Advanced Industrial Science and Technology* (AIST), fue uno de los pioneros en el desarrollo de los primeros robots bípedos, en donde proponía que su centro de masa se movía de manera horizontal. En la Fig. 1-14 se muestra uno de los primeros proyectos de robots bípedos de Shuuji Kajita.

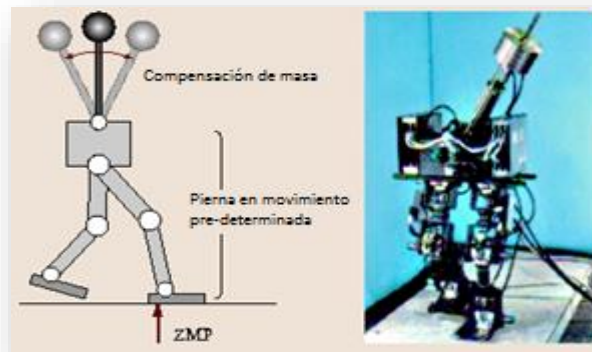


Fig. 1-14 Robot bípedo desarrollado por Shuuji Kajita en AIST.

Otro investigador que propuso incluir sensores en los pies del robot para permitirle subir terrenos inclinados, fue Zhen (J.Shen, 1990). Utilizando múltiples sensores en los pies podía detectar la inclinación del robot y de esta manera poder medir el centro de masa. Esta técnica es muy utilizada en la actualidad.

Utilizar el péndulo invertido para lograr el caminado es una técnica que fue utilizada por primera vez por Kajita y Tani (Kajita, y otros, 2003). Ellos llevaron a cabo varios estudios, por ejemplo la fase de soporte en una sola pierna y el cambio de soporte en una pierna vertical, así como mantener por unos instantes la fase de doble soporte. El método del péndulo invertido es uno de los más frecuentes métodos utilizados para generar un caminado estable utilizando el concepto de *Zero Moment Point* (ZMP) para obtener el movimiento del robot.

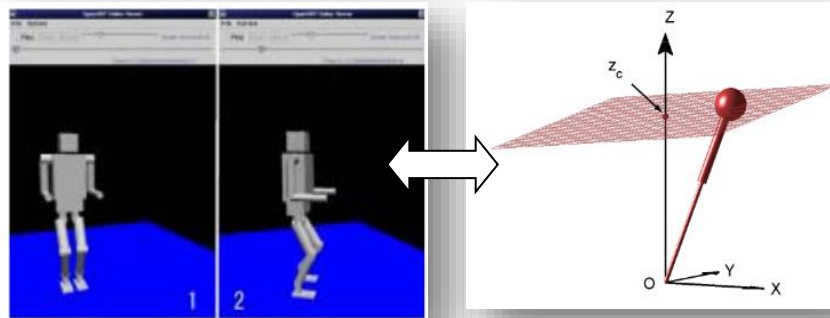


Fig. 1-15 Simulación de robots bípedos que utilizaron Kajita y Tani durante sus estudios del caminado.

1.2.2. Robots humanoides

Conforme la tecnología fue evolucionando los robots humanoides también lo fueron haciendo como Wabian-2 un robot construido únicamente para la investigación en la universidad japonesa de Waseda (Pratt, y otros, 1998). El robot bípedo fue diseñado con una altura de 1.50 m y un peso de 60 kg. Wabian-2 tiene 41 grados de libertad en total como se muestra en la Fig. 1-16. En las piernas 12 grados de libertad, 2 grados de libertad pasivos en los pies, cintura 2, tronco 2, brazos 14, manos 6, cuello 3.

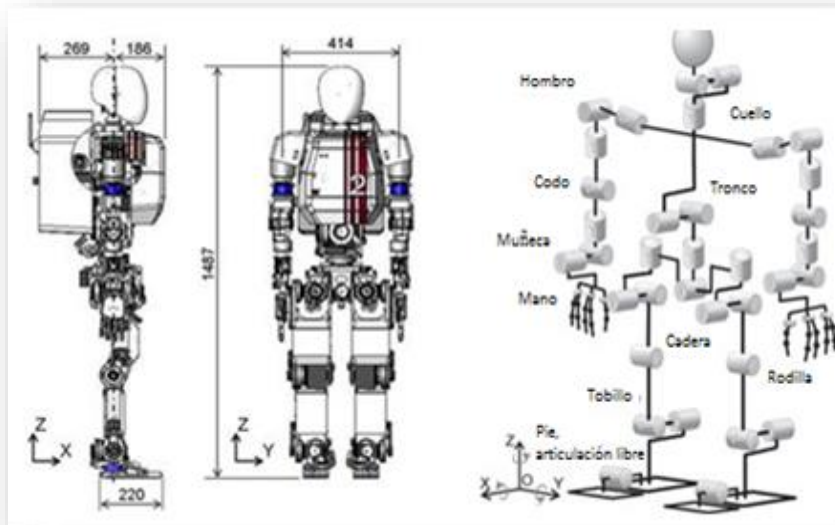


Fig. 1-16 Robot bípedo Wabian-2 desarrollado en la universidad japonesa de Waseda.

Uno de los robots más sofisticados es ASIMO (Fig. 1-17) desarrollado por la compañía Honda Motors (American Honda Motor Co. Inc., 2014). Tiene 1.2 metros de alto y pesa 52 kilogramos. ASIMO tiene 24 grados de libertad, 5 en cada brazo, 1 en cada mano, 6 por pierna. Los 6 grados de libertad de cada pierna están distribuidos de la siguiente manera: tobillo 2 grados de libertad, rodilla 1 grado de libertad, cadera 3 grados de libertad. ASIMO es controlado por una unidad que lleva a bordo y en la espalda lleva una fuente de poder para su autonomía.

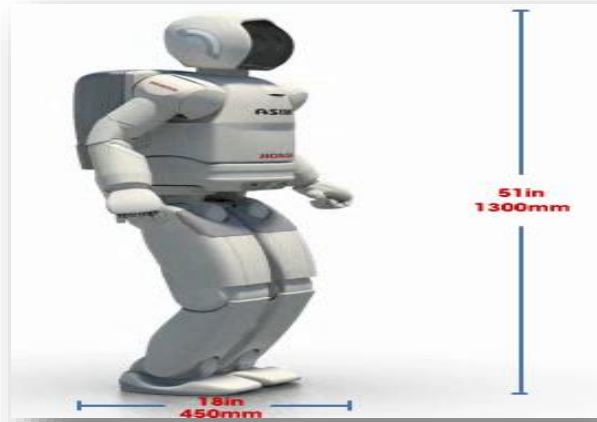


Fig. 1-17 Robot ASIMO desarrollado por Honda.

El robot humanoide HUBO de KAIST Fig. 1-18 (citizendum, 2013) *Korea Advanced Institute of Science & Technology* (KAIST), ha tenido la atención de la comunidad robótica a nivel mundial, con un costo de un millón de dólares HUBO sigue los pasos de ASIMO pero usando lo último en tecnología. El equipo KAIST ha realizado un avance rápido comparado con ASIMO que tomó 15 años y 300 millones de dólares. HUBO tiene 125 cm de alto con un peso de 55 kg, tiene 12 grados de libertad en las piernas (3 cadera, 1 rodilla, 2 tobillo) y 8 grados de libertad en los brazos (3 hombro, 1 codo). Además, puede mover de forma independiente sus dedos y los globos oculares porque tiene 2 grados de libertad para cada ojo (uno para giro de la cámara y otro para la inclinación) y en cada mano tiene 7 grados de libertad también tiene 2 en el cuello y 1 en el torso.

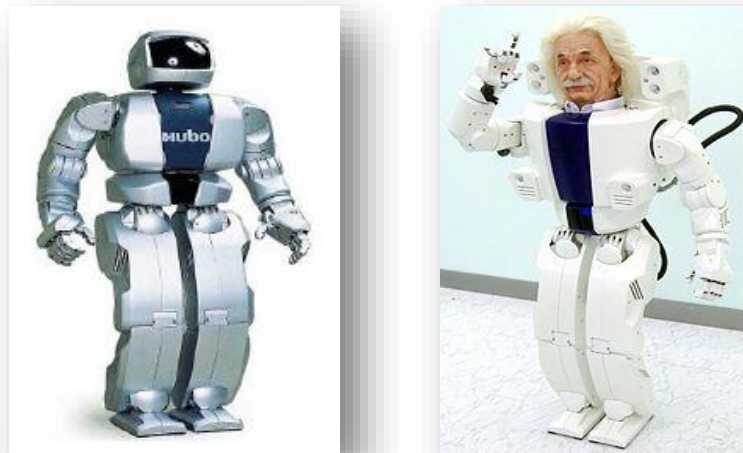


Fig. 1-18 Robot HUBO Y HUBO ALBERT una de las series de prototipos bípedos desarrollados por KAIST.

Sin embargo HUBO no es el único robot diseñado por KAIST también esta HUBO ALBERT, HUBO 2, HUBO 2 PLUS entre otros, en la Tabla 1-4 se muestran los diferentes robots que ha desarrollado KAIST.

	KHR-0 (2001)	KHR-1 (2002)	KHR-2 (2004)	HUBO (KHR-3) (2005)	Albert HUBO (2005)	HUBO 2 (KHR-4) (2008)	HUBO 2 Plus (2011)
Weight	29 kg	48 kg	56 kg	56 kg	57 kg	45 kg ^[3]	43 kg
Height	110 cm	120 cm	120 cm	125 cm	137 cm	125 cm	130 cm
Walking speed	-	1.0 km/h	1.2 km/h	1.25 km/h	1.25 km/h	1.5 km/h	1.5 km/h
Continuous operating time	-	-	-	60 minutes	60 minutes	120 minutes	120 minutes
Degrees of Freedom	12	21	41	41	66	40	38

Tabla 1-4 Robots construidos a lo largo de los años por KAIST.

1.2.3. Robots bípedos comerciales

Debido al alto costo de los robots antes mencionados han surgido compañías dedicadas a fabricar robots de bajo costo y que cumplan con las necesidades básicas para realizar investigaciones sobre el caminado en robots bípedos y otras configuraciones. Estos robots poseen limitaciones en cuanto a los motores utilizados o sensores pero ofrecen una plataforma efectiva para desarrollar estrategias de control de caminado entre otros algoritmos.

A continuación se muestran algunos de los robots comerciales más sobresalientes que existen en la actualidad. Los robots mencionados no son una lista exhaustiva hay muchos otros que por el momento no se estudiarán.

Nao Fig. 1-19 es un robot humanoide programable y autónomo, desarrollado por Aldebaran Robotics (Aldebaran Robotics, 2014). Nao ocupó el lugar del perro robot Aibo de Sony en el concurso Robo Cup. Tiene 54 cm de altura y existen tres versiones de 14, 21 o 25 grados de libertad. La versión utilizada en la Robo Cup es la de 21 grados de libertad. La versión con más grados de libertad es la de 25 que se dividen de la siguiente manera: cabeza 2 grados de libertad, 5 grados de libertad en cada brazo, 1 grado de libertad pelvis, 5 grados de libertad en cada pierna y 1 grado de libertad en cada mano. El costo de la versión de 25 grados de libertad está alrededor de los \$70,000.00 MXN.



Fig. 1-19 Robot comercial Nao.

Otro ejemplo de estos robots son los modelos de Kondo (Kondo-kagaku CO. LTD, 2013), se muestra en la Fig. 1-20, los cuales son ampliamente usados en laboratorios de investigación y competencias de robótica. Por lo general estos modelos se venden como kits con los cuales es posible construir varias estructuras incluyendo una de tipo humanoide. La estructura humanoide consta de 17 servo motores, lo que le proporciona 17 grados de libertad (5 en cada pierna, 3 en cada brazo y uno en el cuello).



Fig. 1-20 Robot comercial Kondo.

Otra plataforma robótica para construir humanoides viene de la compañía coreana Robotis, la cual desarrolla el kit de robótica Bioloid ((ROBOTIS, 2014), ver Fig. 1-21). Con esta plataforma se pueden armar diferentes tipos de estructuras robóticas. La plataforma Bioloid tiene la ventaja sobre las demás en el tipo de servo motor que usa, ya que cada servo motor es de tipo digital. El kit cuenta con 18 servomotores los cuales pueden ser usados para construir un robot humanoide. La disposición de los grados de libertad para construir el robot humanoide depende de cada investigador. Se pueden armar robots humanoides con 2 grados de libertad en las piernas hasta el más complejo utilizando todos los servomotores. La configuración del robot de 18 grados de libertad sería 4 grados de libertad por pierna, 3 grados de libertad por brazos, 2 grados de libertad tronco, 2 grados de libertad cuello y cabeza.



Fig. 1-21 Kit de robótica Bioloid.

1.2.4. Investigaciones de robots bípedos de bajo costo

Algunos investigadores de robots bípedos han decidido no limitar sus ideas por los altos costos de las estructuras de algunos robots bípedos comerciales por ejemplo el Robot Nao, que a pesar de que no es tan costoso como el robot ASIMO el valor comercial sigue siendo alto. Por lo tanto algunos investigadores han optado por construir sus propias estructuras de robots bípedos, para probar nuevos algoritmos de caminado. Ellos suelen construir pequeñas estructuras con solo los grados de libertad necesarios para implementar un caminado básico, pero suficiente en el robot bípedo. La mayoría de ellos usan actuadores de bajo costo como los servomotores de aeromodelismo.

En la universidad de Bonn (Alemania), el grupo de investigación NimbRo-Learning Humanoid Robots (NimbRo, 2014) ha desarrollado algunos robots bípedos de bajo costo. Todos estos robots humanoides basados en servos son totalmente autónomos. Ellos tienen un amplio campo visual de cámaras, amplio poder de cómputo, y comunicación inalámbrica tales robots son controlados por un Pocket PC, también poseen una habilidad básica para jugar fútbol, caminar dinámicamente en todas direcciones (hasta 20cm/s en dirección frontal) y son capaces de girar en el sitio. Detectan el balón, las porterías, y a ellos mismos en el campo de juego. También se pueden aproximar al balón y regatear con él. Éstos son usados principalmente en la competencia Robocup y se encuentran en continuo mejoramiento. Dentro de este grupo de robots se encuentran robots de (19 a 21) grados de libertad, las alturas de 60 a 100 cm. y pesos de 2.4 a 5 Kg. En la Fig. 1-22 se muestran algunos prototipos de este robot.

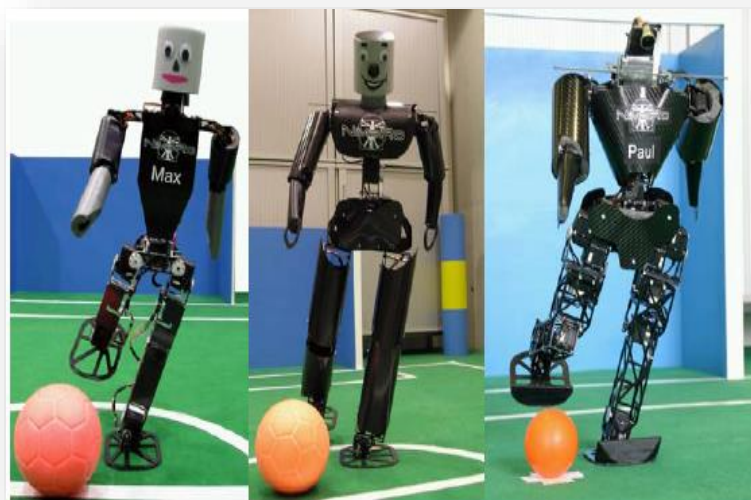


Fig. 1-22 Robots bípedos desarrollados por la universidad de Bonn.

Algunos robots bípedos de bajo costo han sido desarrollados en la universidad de Oeste de Australia. Por ejemplo, “Johnny Walker” (Bräunl, 2014) que se muestra en la Fig. 1-23, el cual tiene 9 grados de libertad: 2 grados de libertad en los tobillos, 2 grados de libertad en la rodilla, 4 grados de libertad en la cadera y 1 grado de libertad en el torso. Este usaba un controlador de lazo abierto para el caminado que no obtuvo el éxito deseado. El siguiente modelo tenía brazos agregando 2 grados de libertad. Este nuevo robot fue llamado “Jack Daniels” y fue muy similar al robot “Johnny Walker”.

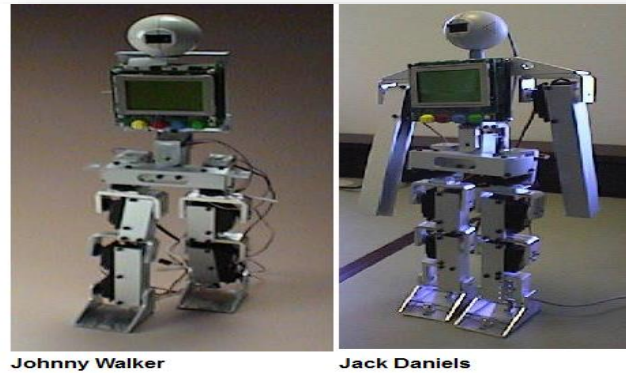


Fig. 1-23 Johnny Walker y Jack Daniels.

En el siguiente modelo se le quitó el servo del torso y los del giro en las caderas. Los tobillos obtuvieron más grados de libertad, permitiendo al robot doblarse hacia los lados. En total este robot, “Andy Droid 2” que se muestra en la Fig. 1-24, tenía 17 grados de libertad, 5 grados de libertad por pierna, 3 por brazo y 1 grado de libertad en la cabeza.

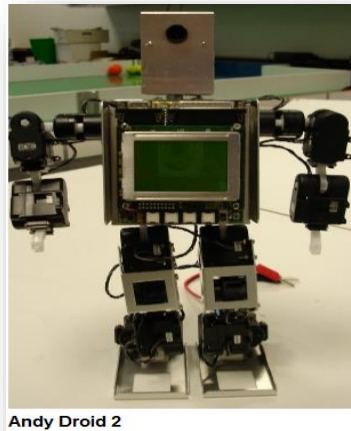


Fig. 1-24 Robot bípedo Andy Droid 2.

Todos los robots mencionados hasta este momento eran impulsados por servo motores. El robot “Rock Steady” es impulsado por motores de DC. Cuenta con un mecanismo que le permite realizar un paso. Un tercer motor está montado detrás de la pantalla del controlador para cargar el peso de derecha a izquierda, y por ende reacomodar el centro de masa. “Rock Steady” tiene solo tres grados de libertad y la estructura está hecha de plástico.

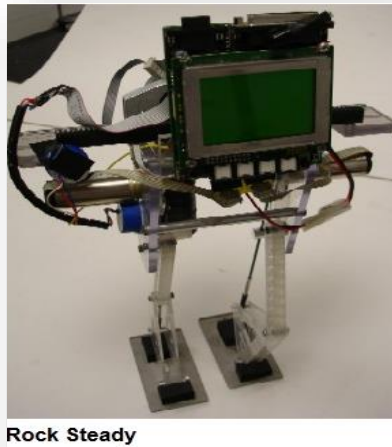


Fig. 1-25 Robot bípedo Rock Steady.

La investigación con robots humanoides necesita de mucho tiempo de estudio y la inversión de mucho dinero, por ejemplo el desarrollo de los robots ASIMO y HUBO. En donde se gastaron millones de dólares para su desarrollo.

Cuando no se cuenta con una inversión de millones de dólares se cuenta con robots comerciales que son más asequibles. Los robots comerciales son una opción en donde el costo no es tan elevado alrededor de 2,000 dls a 7,000 dls dependiendo del modelo. Hay robots configurables que permiten probar con diferentes grados de libertad como Bioloid.

Con los robots configurables estudiar el caminado humano utilizando diferentes configuraciones sería tardado se tendría que armar las configuraciones a estudiar y programar el robot. También habría limitante en cuanto a los grados de libertad. Como se vio en el capítulo anterior el ser humano cuenta con más de 30 grados de libertad en total en las piernas. Poder simular y modelar un robot bípedo en un entorno virtual ayudaría con el estudio al no tener que comprar el robot o tener que reconfigurarlo para cada estudio.

1.3. Plataformas de simulación robóticas

Como se mencionó en la sección anterior simular un robot bípedo es de gran ayuda para estudiar los movimientos y la interacción física del robot mientras camina, sin tener que construir el robot o comprarlo. Con la ayuda de la plataforma virtual se realizarían primero las pruebas del movimiento del robot de manera virtual.

Para que la plataforma de simulación sea una herramienta para el desarrollo de robots bípedos necesita tener varias características. Unas de estas características sería simular entornos dinámicos por ejemplo simular la gravedad, el torque de los motores, el peso de los eslabones, la inercia de los eslabones, simular la posición, velocidad y aceleración de las articulaciones. También se necesita un modelo virtual del robot bípedo que quisiéramos estudiar. Dentro de la plataforma de simulación sería de gran ayuda tener herramientas de control para realizar la pruebas de caminado del robot bípedo. Por último tener un entorno virtual en donde se pudiera observar el robot realizando las actividades que se le programaron es indispensable para el estudio.

La ventaja de las plataformas de simulación es que utiliza el modelado 3D y renderizado en un entorno virtual que imitan el entorno real del robot. Por ejemplo, podemos simular un aspirador robótico en una casa. Primero tendríamos que construir un modelo virtual del robot y después construir el modelo correspondiente de la casa.

Este ejemplo de simulación incluye al menos dos principales modelos virtuales en 3D: un modelo virtual en 3D para el robot y un modelo para el entorno donde va a trabajar el robot, una casa, calles, etc. Fig. 1-26. Otros objetos, como una silla, una cama, un sofá, piso, etc. deben ser también simulados ya que el robot debe evitar la colisión con estos objetos. Un proyecto de simulación completo incluye el código de programación que se instalará en el modelo físico del robot.



Fig. 1-26 Entorno virtual para simulación de robots.

Para comenzar con el estudio del robot bípedo se necesita realizar una simulación en su entorno virtual. Realizar estas pruebas permite al investigador comprobar las características mecánicas del robot. Como el robot no está construido físicamente realizar modificaciones mecánicas no tendría ningún costo adicional.

1.3.1. Ventajas y desventajas de la simulación

Cualquier tecnología tiene sus ventajas y desventajas. Utilizando herramientas de simulación para el diseño de robots es un método sencillo y menos costoso para construir robots complejos o probar diferentes arquitecturas.

Ventajas

- Un costo menor para desarrollar un robot desde cero;
- Si el diseño no cumple con las especificaciones realiza una modificación tendría un menor costo;
- El diseño mecánico del robot puede ser probado antes de su construcción;
- Para desarrollar proyectos grandes las arquitecturas robóticas pueden ser probadas por módulos sin tener que realizar el ensamble completo;
- Realizando simulaciones se puede determinar si el robot cumple con las especificaciones del proyecto;
- El tiempo transcurrido entre el inicio del proyecto y su realización puede ser acortado;

Desventajas

- El mundo real puede someter al robot en muchos más escenarios que un mundo virtual;
- Todos los programas de simulación simulan lo que están programados para simular;
- Los costos del software pueden ser muy altos.
- La información para empezar a programar puede no estar disponible.

En la actualidad existen muchas plataformas de simulación que pueden ayudarnos con el desarrollo de arquitecturas robóticas. A continuación se muestra una lista de algunos de los programas de simulación más importantes en la actualidad.

Webots es plataforma de simulación en 3D desarrollado por Cyberbotics (Cyberbotics Ltd, Copyright © 2014, 2014). Webots es un entorno de desarrollo para programar y simular robots móviles. El software cuenta con módulos dinámicos en donde se puede simular la masa, fricción, textura, etc. También cuenta con sensores y actuadores para el robot. Otra característica es que cuenta con su propio entorno de desarrollo para programar micro controladores. Su costo es de alrededor de 2,700 dls solo a instituciones académicas.

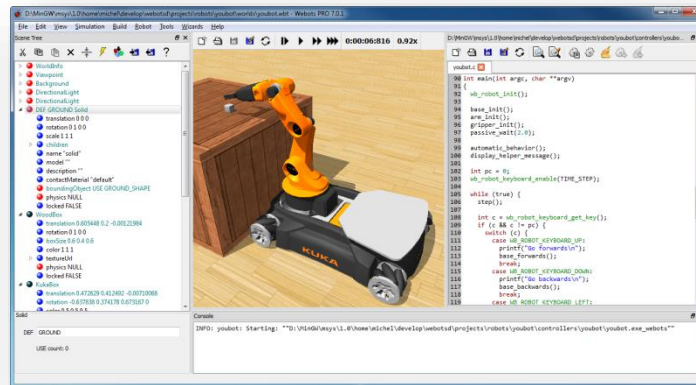


Fig. 1-27 Plataforma de simulación Webots.

Diseñado por *Coppelia Robotics* (Coppelia Robotics, 2014), el simulador V-REP, con entorno de desarrollo integrado, se basa en una arquitectura de control distribuido: cada objeto / modelo puede ser controlado individualmente a través de una secuencia de comandos incrustada. Esto hace a V-REP muy versátil e ideal para aplicaciones multi-robot. Los controladores pueden ser escritos en C / C ++, Python, Java, Lua, Matlab, Octave o Urbi. V-REP se utiliza para el desarrollo de algoritmos rápidos, simulaciones de automatización de fábricas, prototipado rápido, investigación de robots, monitoreo remoto, etc. También cuenta con propiedades de simulación dinámica, como fricción, gravedad, velocidades y aceleración y con la simulación de un piso virtual. En el año 2014 V-REP liberó una versión estudiantil en donde no limita o restringe el software.

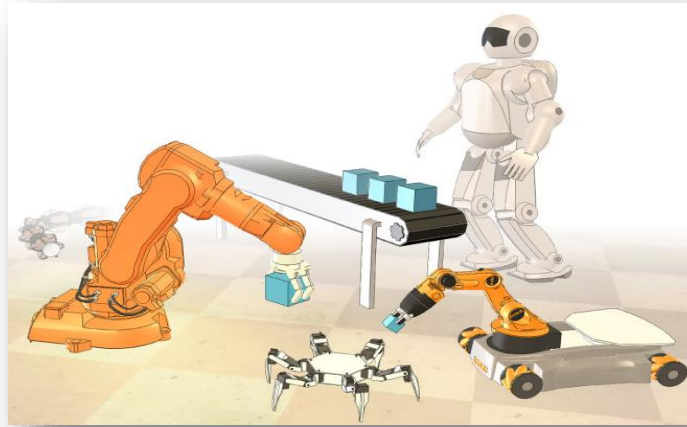


Fig. 1-28 Plataforma de simulación V-rep.

RoboWorks es una herramienta de simulación en 3D desarrollada por la empresa Newtonium (Newtonium Copyright 2000, 2008). El software se puede utilizar para simular en un mundo virtual en 3D el comportamiento de los robots industriales y de servicios. Puede comunicarse con plataformas como Matlab Y Labview. Con el software RoboWorks se puede importar programas de Matlab y simularlos en el entorno RoboWorks. Esto nos da la ventaja de utilizar todas las bibliotecas de Matlab. Sin embargo los gráficos que proporciona RoboWorks están muy limitados y la construcción de geometrías complejas dentro del entorno RoboWorks solo permite construcción de geometrías sencillas.

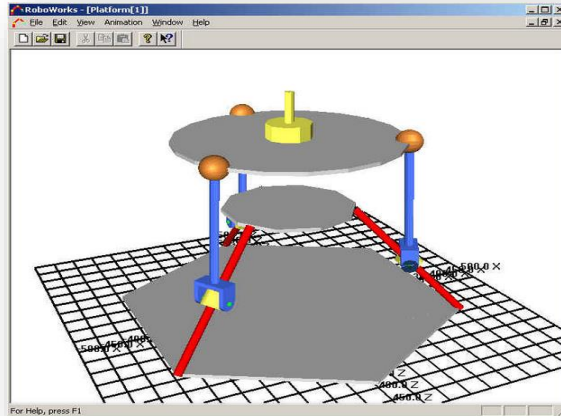


Fig. 1-29 Ventana de simulación de RoboWorks.

Robostudio es una herramienta de simulación industrial desarrollada por ABB (ABB Copyright 2014, 2014). El software le permite desarrollar escenarios de simulación realistas de robots industriales de ABB. Cuenta con simulaciones muy realistas de una línea de proceso Fig. 1-30. Tiene simulaciones dinámicas como fricción, masa, detección de colisión con objetos etc. Una característica importante es que puede trabajar fuera de línea. Esto quiere decir que puedes simular una línea de ensamble en una empresa utilizando modelos virtuales de los robots reales. Cuando acabas con las simulaciones puedes conectarte a los robots físicos y probar los cálculos realizados para la simulación.

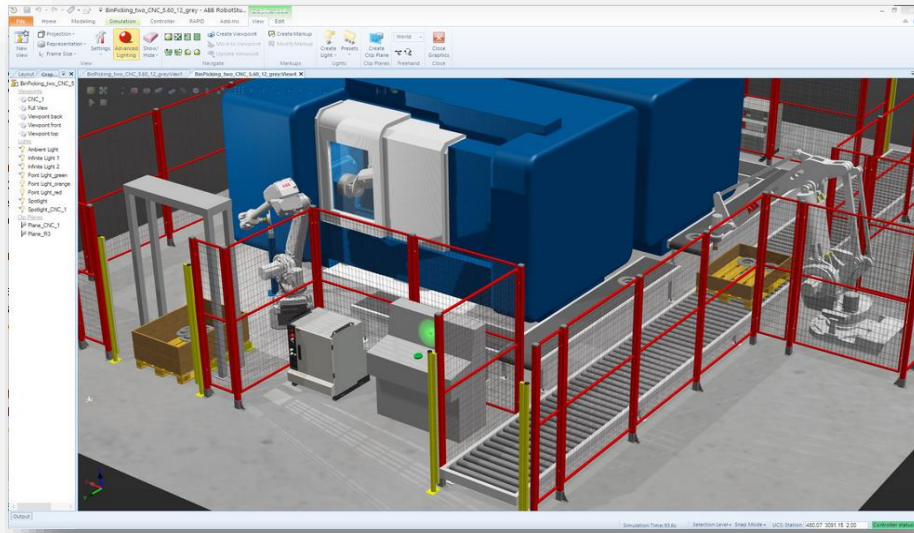


Fig. 1-30 Ventana de simulación de Robostudio.

Roboguide es un software desarrollado por Fanuc y es totalmente compatible con los robots Fanuc (FANUC America Corporation Copyright, 2014). La plataforma se ha diseñado para ser utilizado para simular en entornos virtuales del comportamiento de los robots industriales Fanuc. Roboguide cuenta con una serie de programas para realizar diferentes simulaciones, por ejemplo HandlingPRO de la familia de Roboguide permite simular robots seriales, PaintPRO permite simular el trayecto de un robot serial para un proceso de pintura, Weldpro permite simular procesos de soldadura con robots Fanuc estos son los más importantes de la familia Roboguide.

A pesar de que existe una amplia variedad de programas de simulación robótica la mayoría de estos programas son caros y difíciles de conseguir. Además algunos de estos programas tienen un mercado muy especializado como los programas de Fanuc y ABB, que solo sirven para sus robots.

Algunos de los programas de simulación mencionados trabajan con robots seriales e inclusive se especializan en robots manipuladores de su propia marca por ejemplo Fanuc y ABB. Si quisiéramos simular robots bípedos en algunas de estas plataformas sería imposible debido a que los programas de simulación fueron diseñados con otro objetivo.

Si quisiéramos simular robots que no fueran seriales por ejemplo robots móviles, hay dos programas de simulación que ofrecen un buen paquete de herramientas para poder realizar este tipo de estudio Webots y V-REP El primer programa es uno de los más utilizados en la investigación con robots bípedos. Inclusive este programa ofrece una plataforma de simulación para el robot NAO. Sin embargo su precio es alto y no existe una versión para estudiantes. El segundo programa de simulación V-REP es una muy buena opción si quisiéramos trabajar con robots bípedos. Dos años atrás este programa tampoco estaba accesible para estudiantes. Las últimas versiones de este programa ofrecen una descarga gratuita del programa completo solo para estudiantes.

Durante el estudio de esta tesis no se trabajó con V-REP debido a que su accesibilidad es relativamente nueva. No se descarta como una opción a futuro para evaluar las capacidades de simulación en cuanto a robots bípedos.

Capítulo II

Modelado del caminado humano

Si quisiéramos imitar el caminado humano ¿sería necesario crear una arquitectura robótica con todos los grados de libertad que tiene el cuerpo humano? Probablemente la respuesta sería no. En la actualidad hay muchas arquitecturas robóticas que pueden imitar este movimiento, con menos grados de libertad que el cuerpo humano.

Sin embargo ¿qué ventaja representan estos grados de libertad adicionales? o ¿qué configuraciones diferentes se podrían proponer con los mismo grados de libertad que tienen las arquitecturas robóticas en la actualidad? y si pudiéramos cambiar solo las configuraciones de las articulaciones ¿cuáles configuraciones pueden caminar? o ¿cuáles no pueden caminar? Se vuelve interesante poder estudiar todas las combinaciones que se puedan proponer. Para estudiar todas estas configuraciones habría que proponer una arquitectura que tuviera los mismos grados de libertad que el cuerpo humano.

Durante este capítulo se propuso una arquitectura lo más similar posible a las dimensiones de una persona adulta. Las dimensiones se basaron en el estudio ergonómico el cual se vio en el capítulo anterior. Utilizando las tablas de medidas corporales se dibujaron eslabones simples que comprenden una parte del tronco y las extremidades inferiores del cuerpo humano como eslabones para la cadera, para el fémur, las rodillas, los pies.

Basados en el estudio del caminado humano se tomaron en cuenta los grados de libertad de las extremidades inferiores con mayor aportación para lograr la acción de caminar. Éstos fueron 21 grados de libertad, 3 grados de libertad para la pelvis, 6 grados de libertad para la cadera (lado izquierdo y derecho), 6 grados de libertad para rodilla (izquierda y derecha), 6 grados de libertad para el tobillo (izquierdo y derecho). Para la arquitectura propuesta se tomaron las medidas de una persona promedio de 1.80 mts.

Para representar las medidas y eslabones del cuerpo humano se necesitó de una herramienta CAD que nos permitiera realizar un ensamble de las extremidades inferiores del cuerpo humano. El programa CAD utilizado durante el estudio fue Solidworks. *Solidworks* es un programa que tiene muchas herramientas para ensamblar arquitecturas mecánicas.

En este capítulo también se desarrollará un análisis cinemático de la arquitectura propuesta de 21 grados de libertad. Después se mostrará el análisis dinámico de unas arquitecturas de 2 grados de libertad cada una. El fin de realizar el análisis dinámico y cinemático es demostrar la dificultad que tienen este tipo de análisis.

2.1. Solidworks

Solidworks es un programa de diseño asistido por computadora para modelado mecánico desarrollado en la actualidad por Solidworks Corp., para el sistema operativo Microsoft Windows.

El software de automatización de diseño mecánico de Solidworks es una herramienta de diseño y de modelado sólido paramétrica basada en operaciones que aprovecha la facilidad de aprendizaje de la interfaz gráfica de usuario de Windows™.

Del mismo modo que un ensamble está compuesto por una serie de piezas individuales, un modelo de Solidworks también está compuesto por elementos individuales. Cuando se crea un modelo mediante el software de Solidworks trabaja con operaciones geométricas fáciles de entender, como salientes, cortes, taladros, nervios, redondeos, chaflanes y ángulos de salida.

Se pueden construir ensambles complejos como el que se muestra en la Fig. 2-1. Los ensambles consisten en numerosos componentes que pueden ser piezas de otros ensambles llamados sub ensambles.



Fig. 2-1 Ensamble realizado en Solidworks.

2.2. Propuesta para las extremidades inferiores del cuerpo humano

En el capítulo anterior se estudió la mecánica de las articulaciones del cuerpo humano. Basados en este estudio se decidió desarrollar un modelo simplificado de las piernas del ser humano, utilizando 21 grados de libertad para la arquitectura simplificada.

Para proponer una arquitectura con la mayoría de los grados de libertad se utilizaron geometrías sencillas. Para los eslabones que representan el fémur, la tibia, una parte del tronco y la cadera, se utilizó una forma de cilindro. Como se muestra en la Fig. 2-2 al final de cada eslabón se encuentra la mitad de un cilindro en el cual se encuentra la unión de las articulaciones.

Como se muestra en la Tabla 2-1 cada eslabón se diseñó de manera que pudiera conectarse con una articulación. Para las uniones de cada eslabón se utilizaron 3 articulaciones, cada articulación tiene un grado de libertad en diferente plano (Sagital, Frontal, Transversal) como se muestra en la Fig. 2-2. Todas las articulaciones están unidas en el mismo punto. Esto nos ayuda a formar una articulación esférica.

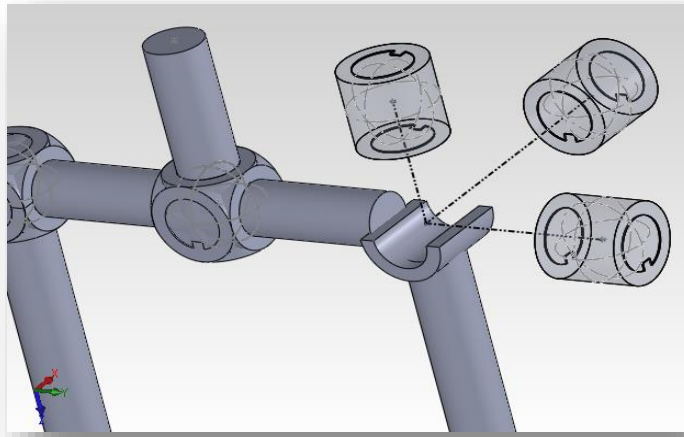
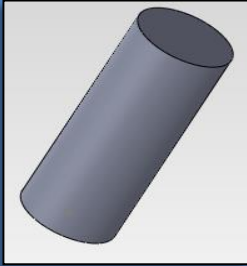
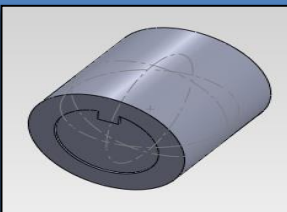


Fig. 2-2 Representación de articulación esférica utilizando 3 actuadores cilíndricos.

Dibujo de eslabón.	Parte del cuerpo que representa.	Medidas (mm).	Cantidad.
	Parte del tronco. Comprende desde la pelvis hasta el ombligo.	Largo de 100, diámetro de 60.	1
	Articulación.	Largo de 60, diámetro 60.	21

	Unión cadera tronco.	Largo de 100, diámetro 40.	2
	Fémur.	Largo de 360, diámetro 40.	2
	Tibia.	Largo 370, diámetro 40	2
	Pie.	Altura de 100, largo 210, ancho 94.	2

Tabla 2-1 Eslabones utilizados para representar las extremidades inferiores del cuerpo humano.

Cada eslabón de la arquitectura fue creada en un archivo individual. Una vez que se tienen todos los archivos de las articulaciones y los eslabones, se ensamblaron utilizando la herramienta de Solidworks para ensambles de piezas. Cada pieza es exportada desde su dibujo de origen e insertada dentro del ensamble. En la Fig. 2-3 se muestra la ruta de ensamble para cada pieza.

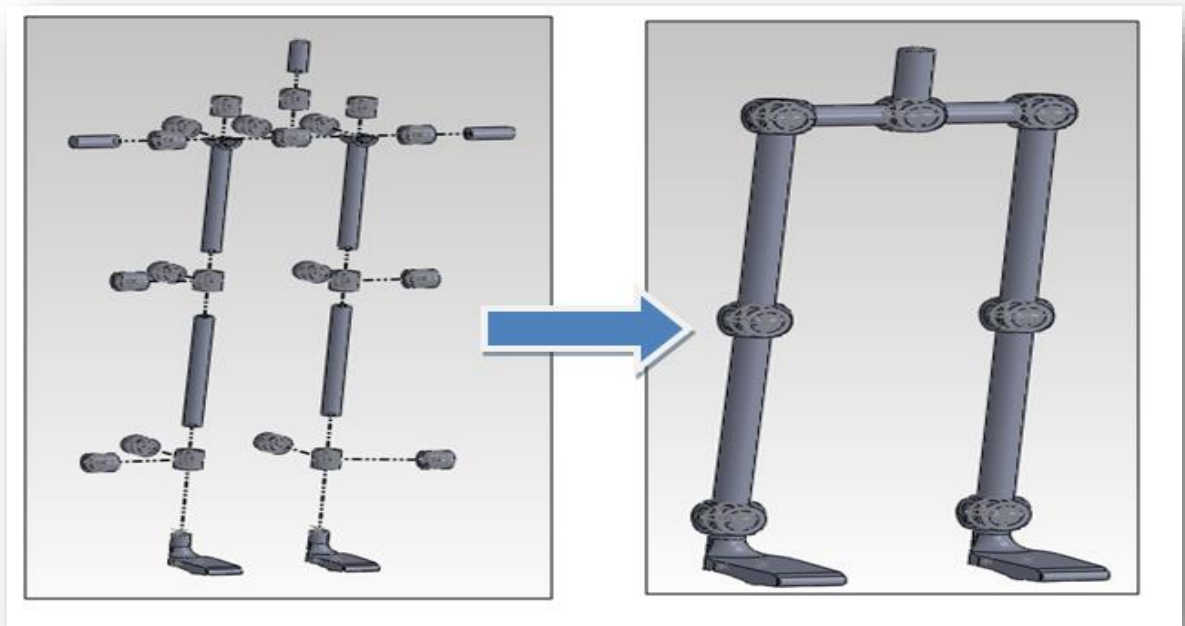


Fig. 2-3 Ensamble de la representación de las extremidades inferiores del cuerpo humano utilizando 21 grados de libertad.

En el ensamble se restringió los grados de libertad de cada articulación a rotación solamente. Para el estudio no se tomaron en cuenta los grados de libertad de los dedos de los pies debido a que estos grados de libertad contribuyen poco durante el caminado humano.

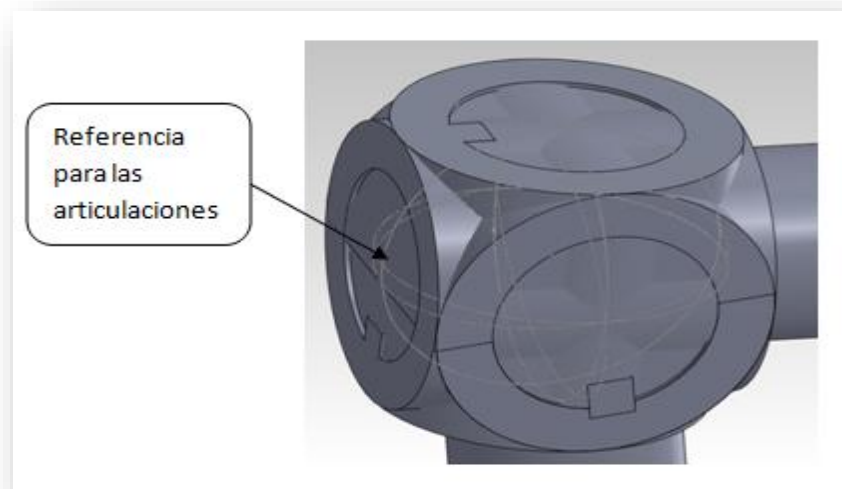


Fig. 2-4 Círculos utilizados para tener una referencia durante el ensamble.

Poder diseñar el ensamble con solo grados de restricción rotacionales en las articulaciones, genera algunas dificultades en el ensamble debido a que todas las articulaciones tienen como referencia un solo punto. Por ejemplo si se intenta girar una articulación las demás articulaciones también rotan alrededor del mismo punto. Por lo tanto se pierde la referencia del ángulo inicial y causa conflictos en el ensamble. Para ensamblar de manera exitosa la arquitectura, se agregaron aros virtuales que permiten a Solidworks

tomarlos como referencia. Las referencias nos ayudan para girar cada articulación respecto del aro virtual de una articulación diferente. En la Fig. 2-4 se muestran los aros virtuales utilizados como referencias.

Una vez que se ensamblaron todos los eslabones y articulaciones, se tomaron medidas para poder observar la relación del ensamble con las medidas de una persona adulta. A pesar de que se tomaron en cuenta las medidas corporales para crear los eslabones se tuvo que ir ajustando los eslabones de manera que la figura para pudiera representar una persona adulta. Las medidas finales quedan como se muestra en la Fig. 2-5.

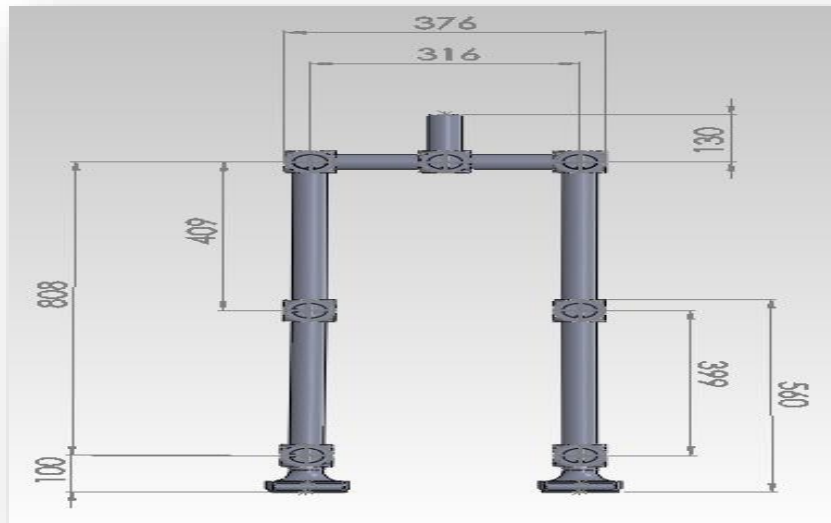


Fig. 2-5 Dimensiones (mm) de arquitectura propuesta.

2.3. Cinemática de una arquitectura de 21 grados de libertad

Un manipulador mecánico se puede modelar como una cadena articulada en lazo abierto con algunos cuerpos rígidos conectados por una articulación prismática o de revolución movida por actuadores. Un extremo de la cadena se une a una base de soporte mientras que el otro extremo está libre y unido con una herramienta (efector final) para manipular objetos o realizar tareas de montaje.

A partir de valores conocidos de articulaciones y parámetros geométricos de los eslabones de algún robot, se calcula la posición y la orientación del efector final con respecto a un sistema de referencia. Se controla el robot serial de manera indirecta, a partir del movimiento de las articulaciones.

Para resolver el problema de la cinemática directa se utiliza álgebra vectorial y matricial para desarrollar un método generalizado y sistemático para describir y representar la localización de los elementos de un brazo con respecto a un sistema de referencia fijo. Uno de los métodos más utilizados es el que utiliza matrices de transformación homogénea por presentar varias ventajas al momento de analizar un sistema complejo.

2.3.1. Matrices de transformación homogénea

Una matriz de transformación homogénea de 4X4 relaciona dos sistemas de coordenadas, de forma que:

$$T = \begin{bmatrix} R_{3 \times 3} & P_{3 \times 1} \\ 0 & 1 \end{bmatrix}$$

donde la matriz T representa la posición y la orientación de un sistema de coordenadas con respecto a un sistema de referencia, $R_{3 \times 3}$ representa una matriz de rotación entre dos sistemas coordenados, $P_{3 \times 1}$ representa el vector de translación entre el origen de los mismos sistemas y el ultimo renglón de ceros y uno de la matriz T sirve para el manejo consistente de coordenadas homogéneas. Mediante una matriz de transformación homogénea se asocia a cada eslabón un sistema de referencia, siendo posible representar traslaciones y rotaciones relativas entre los eslabones consecutivos que se puede representar de la siguiente manera.

$$H_i^{i-1}$$

donde H representa una matriz homogénea de transformación entre el sistemas de coordenadas asociado a la (i-1)-ésima articulación y el sistema de coordenadas asociado a la i-ésima articulación.

Mediante estas matrices se puede representar ya sea el totalmente o parcialmente la cinemática de un robot por ejemplo.

$$T = H_3^0 = H_1^0 H_2^1 H_3^2$$

2.3.2. Representación de Denavit- Hartenberg

Para describir la relación rotacional y traslacional entre los elementos adyacentes, Denavit y Hartenberg (1955) propusieron un método matricial de establecer de forma sistemática un sistema de coordenadas (sistema ligado al cuerpo) para cada elemento de una cadena articulada.

La representación de Denavit-Hartenberg (D-H) resulta en una matriz de transformación homogénea que representa cada uno de los sistemas de coordenadas de los elementos en la articulación con respecto al sistema de coordenadas del elemento previo.

Cada sistema de coordenadas se determina y se establece sobre la base de tres reglas:

Localice y asigne los ejes de las juntas z_0, \dots, z_{n-1} . El eje z_i es el eje de revolución de la junta $i + 1$ si ésta es de revoluta, o el eje de translación de la junta $i + 1$ si ésta es prismática.

Establezca el marco de referencia del origen. Coloque el origen en cualquier punto a lo largo del eje z_0 , los ejes x_0 y y_0 se seleccionan de tal manera que formen un sistema coordenado de mano derecha.

Localice el origen del sistema coordenado $x_i y_i z_i$ en el punto donde la normal común a z_i y z_{i-1} intercepte a z_i .

Establezca x_i a lo largo de la normal común entre los ejes z_i y z_{i-1} , a través del origen del sistema $x_i y_i z_i$ localizado en el paso anterior. En caso de que los ejes z_i y z_{i-1} se intercepte x_i en la dirección normal al plano formado por los ejes z_i y z_{i-1} .

Establezca y_i para completar un sistema coordenado de mano derecha.

Establezca un sistema de coordenadas en el extremo del manipulador o de la herramienta colocada en esta posición.

La representación de D-H de un elemento rígido depende de cuatro parámetros geométricos asociados con cada elemento. Estos cuatro parámetros describen cualquier articulación prismática o de revoluta.

d_i : Es la distancia media a lo largo del eje z_{i-1} desde el origen del sistema coordenado $x_{i-1} y_{i-1} z_{i-1}$, hasta la intersección de los ejes x_i y z_{i-1} . Este parámetro es variable si la articulación es prismática.

a_i : Es la distancia medida a lo largo del eje x_i desde la intersección de los ejes x_i y z_{i-1} hasta el origen del sistema coordenado $x_i y_i z_i$.

α_i : Es el ángulo entre los ejes z_i y z_{i-1} medido alrededor del eje x_i .

θ_i : Es el ángulo entre x_i y x_{i-1} medido alrededor del eje z_{i-1} . Este parámetro es variable si la junta i es de revoluta.

Para una articulación de revoluta, d_i , a_i y α_i son los parámetros de articulación y permanecen constantes para un robot, mientras que θ_i es la variable de articulación que cambia cuando el elemento i gira con respecto al elemento $i - 1$. Para una articulación prismática θ_i , a_i y α_i son los parámetros de la articulación y permanecen constantes para un robot, mientras que d_i es la variable de la articulación.

En esta convención cada transformación homogénea H_i es representada como un producto de 4 transformaciones básicas:

$$\begin{aligned}
 H_i &= R(z, \theta_i)_i^{i-1} H(z, d_i)_i^{i-1} H(x, a_i)_i^{i-1} R(x, \alpha_i)_i^{i-1} \\
 &= \begin{bmatrix} c\theta_i & -s\theta_i & 0 & 0 \\ s\theta_i & c\theta_i & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & a_i \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & c\alpha_i & -s\alpha_i & 0 \\ 0 & s\alpha_i & c\alpha_i & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\
 &= \begin{bmatrix} c\theta_i & -s\theta_i c\alpha_i & s\theta_i s\alpha_i & a_i c\theta_i \\ s\theta_i & c\theta_i c\alpha_i & -c\theta_i s\alpha_i & a_i s\theta_i \\ 0 & s\alpha_i & c\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \text{Ec. [2.1]}
 \end{aligned}$$

donde $R(\cdot,*)$ es la matriz de $Rot_{3 \times 3}$ que representa un giro de un ángulo $*$ alrededor del eje \cdot y $H(\%, \&)$ es la matriz homogénea que representa un desplazamiento $\&$ con respecto al origen $\%$.

2.3.3. Análisis cinemático de una arquitectura de 21 grados de libertad

Con las tres reglas básicas anteriores para establecer un sistema de coordenadas ortonormal en cada eslabón y la interpretación geométrica de los parámetros, se desarrollan los parámetros D-H para nuestra arquitectura simplificada.

El análisis por D-H es comúnmente aplicado para robots seriales, de tal forma que para poderlo aplicar a nuestra arquitectura había que formar dos cadenas cinemáticas. La primera cadena se tomó desde el centro de masa hasta la pierna izquierda. La segunda cadena se tomó desde el centro de masa hacia la pierna derecha como se muestra en la Fig. 2-6.

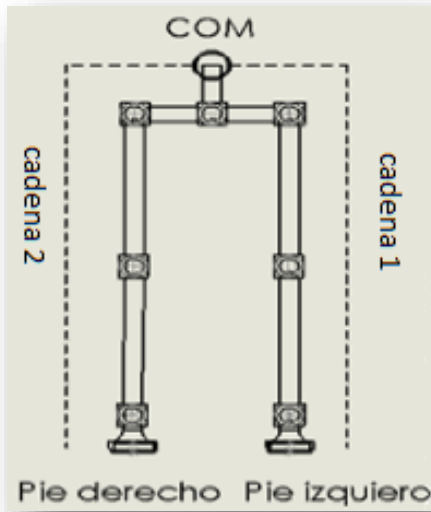


Fig. 2-6 Cadenas cinemáticas para arquitectura simplificada.

Se utilizaron matrices de transformación homogénea para determinar la posición del efector final del el pie. Por ejemplo una transformación se determina desde el centro de masa hasta cualquiera de los dos pies. Estas transformaciones estarían dadas por:

$$H_{13}^0 = H_1^0 H_2^1 H_3^2 H_4^3 \dots H_{13}^{12} \text{ para el pie der.}$$

$$H_{25}^0 = H_1^0 H_2^1 H_3^2 H_{15}^3 \dots H_{25}^{24} \text{ para el pie izq.}$$

A cada articulación se le agregó un eje coordenado que nos ayudara para poder realizar este análisis, como se muestra en la Fig. 2-7 las líneas de color rojo, verde y azul representan un eje coordenado. Este eje se sitúa en el centro de masa de cada articulación. El centro de masa fue determinado por medio de los parámetros que calcula Solidworks.

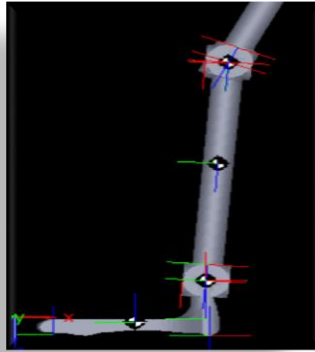


Fig. 2-7 Eje coordenado para cada articulación.

2.3.4. Parámetros de D-H de arquitectura propuesta

Para cada uno de los sistemas coordenados situados en el centro de las articulaciones de las cadenas cinemáticas (pie izquierdo, pie derecho), se obtuvieron los parámetros de D-H.

Para la primera cadena cinemática que se muestra en la Fig. 2-8 se realizó el análisis, proponiendo los marcos de referencia de acuerdo al eje de actuación de cada articulación. La cadena comprende desde el origen, hasta el efector final derecho $H_0 - H_{14}$ como se muestra en la Fig. 2-6. La segunda cadena va desde el origen hasta el efector final izquierdo $H_0 - H_{25}$.

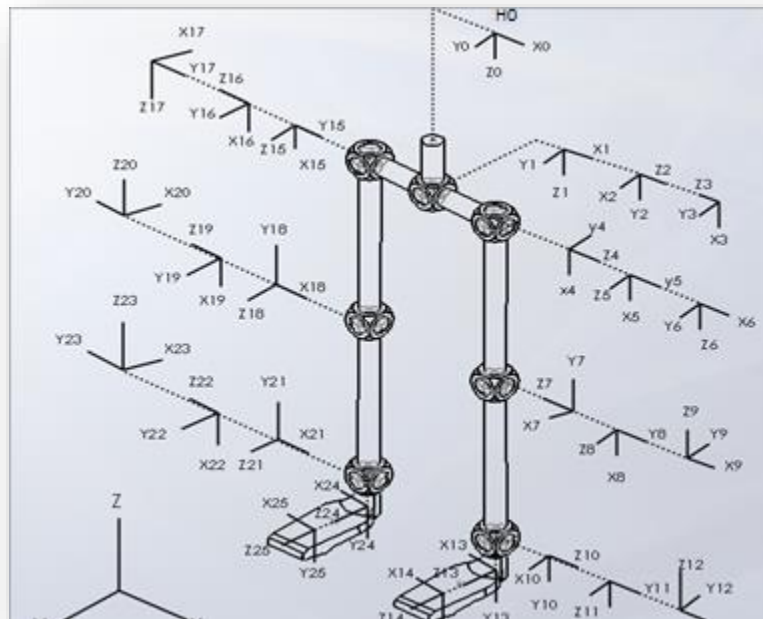


Fig. 2-8 Sistema coordenado para cada eslabón.

Utilizando el análisis D-H se calculó la cinemática directa de las cadenas. Las cadenas cinemáticas se dividieron en cuatro puntos. En cada división se tiene una junta esférica representada por 3 actuadores. Cada actuador trabaja en un plano diferente (frontal, sagital y horizontal).

Los cuatro puntos de análisis son: la primera cadena cinemática que comprende desde el origen hasta la pelvis Fig. 2-9, el segundo punto va desde la pelvis hasta la cadera derecha Fig. 2-10, la tercera sería cadera derecha hasta la rodilla derecha Fig. 2-11 y por último rodilla derecha a tobillo derecho Fig. 2-12.

Utilizando las reglas D-H se calculó la cinemática directa de las dos cadenas como se muestra en las Tabla 2-2 a la Tabla 2-6.

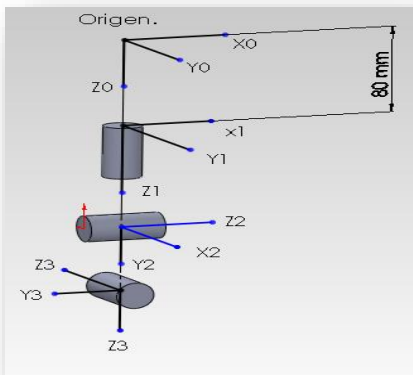


Fig. 2-9 Sistema coordenado origen - pelvis.

Tabla 2-2 Parámetros D-H origen - pelvis.

	a_1 (mm)	d_i (mm)	α_i	θ_i
H_1^0	0	80	0	0
H_2^1	0	0	$\pi/2$	$\pi/2$
H_3^2	0	0	$-\pi/2$	$\pi/2$

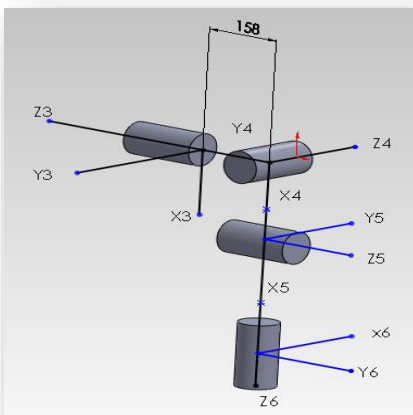


Fig. 2-10 Sistema coordenado pelvis-cadera.

Tabla 2-3 Parámetros D-H pelvis - cadera.

	a_1 (mm)	d_i (mm)	α_i	θ_i
H_4^3	0	-158	$\pi/2$	0
H_5^4	0	0	$\pi/2$	0
H_6^5	0	0	$\pi/2$	$\pi/2$

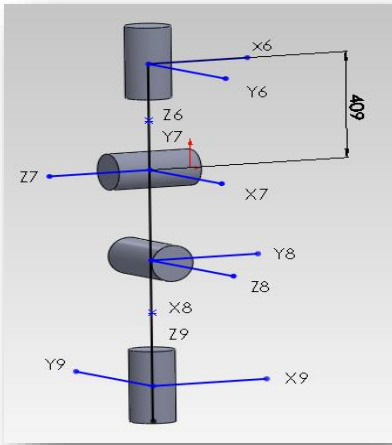


Fig. 2-11 Sistema coordinado cadera-rodilla.

Tabla 2-4 Parámetros D-H cadera - rodilla.

	a_1 (mm)	d_i (mm)	α_i	θ_i
H_7^6	0	409	$-\pi/2$	$\pi/2$
H_8^7	0	0	$-\pi/2$	$-\pi/2$
H_9^8	0	0	$-\pi/2$	$\pi/2$

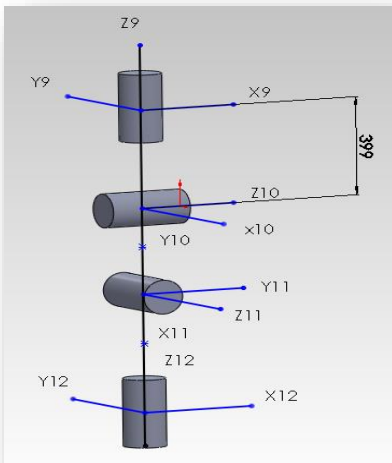


Fig. 2-12 Sistema coordinado rodilla -tobillo.

Tabla 2-5 Parámetros D-H rodilla - tobillo.

	a_1 (mm)	d_i (mm)	α_i	θ_i
H_{10}^9	0	-399	$-\pi/2$	$-\pi/2$
H_{11}^{10}	0	0	$\pi/2$	$\pi/2$
H_{12}^{11}	0	0	$-\pi/2$	$\pi/2$

Debido a que el área del efector final (pie) no solo es un punto, hay un particular interés por conocer más de una coordenada. Con el fin de obtener más de una información se colocaron dos marcos de referencia, uno para el talón y otro para la punta, como se muestra en la Fig. 2-13, serían los marcos de referencia X13, Y13, Z13 y X14, Y14, Z14.

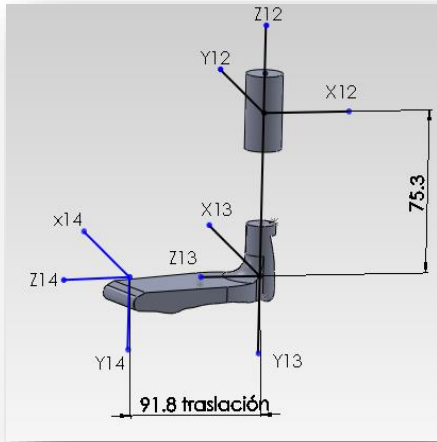


Fig. 2-13 Sistema coordinado tobillo – (pie, talón).

Tabla 2-6 Parámetros D-H tobillo – (pie, talón).

	a_1 (mm)	d_i (mm)	α_i	θ_i
H_{13}^{12}	0	75.3	$-\pi/2$	$\pi/2$
H_{14}^{13}	0	91,8	0	0

De la misma manera utilizando las reglas D-H y tomando en cuenta que los eslabones de las dos cadenas son idénticos se obtuvieron los parámetros D-H para la segunda cadena. Separando está en 4 divisiones como se hizo en la primera cadena. Los parámetros para la segunda cadena quedan como se muestra en la Tabla 2-7.

Tabla 2-7 Parámetros D-H segunda cadena.

	a_1 (mm)	d_i (mm)	α_i	θ_i
H_{15}^3	0	158	$-\pi/2$	0
H_{16}^{15}	0	0	$\pi/2$	0
H_{17}^{16}	0	0	$-\pi/2$	$-\pi/2$
H_{18}^{17}	0	409	$-\pi/2$	$\pi/2$
H_{19}^{18}	0	0	$\pi/2$	$-\pi/2$
H_{20}^{19}	0	0	$\pi/2$	$-\pi/2$
H_{21}^{20}	0	399	$\pi/2$	$-\pi/2$
H_{22}^{21}	0	0	$\pi/2$	$-\pi/2$
H_{23}^{22}	0	0	$\pi/2$	$-\pi/2$
H_{24}^{23}	0	-75.3	$-\pi/2$	$\pi/2$
H_{25}^{24}	0	91,8	0	0

2.3.5. Comparación del estudio cinemático con modelo CAD

Para validar los valores obtenidos de D-H se decidió desarrollar un programa en Matlab que compare los datos D-H, con la arquitectura de 21 grados de libertad desarrollada en el programa CAD. Se crearon dos programas en donde se introducen los valores de los parámetros D-H (ver anexos A-1.A-1 y A-1.A-2). Los programas y funciones utilizados a lo largo del capítulo se encuentran en un CD de apoyo.

Los datos arrojados por los programas de los anexos A-1.A-1 y A-1.A-2 se compararon con la estructura propuesta. Se utilizó sensores proporcionados por Solidworks como se muestra en la Fig. 2-14. En el programa se pueden obtener los valores de cada punto de análisis.

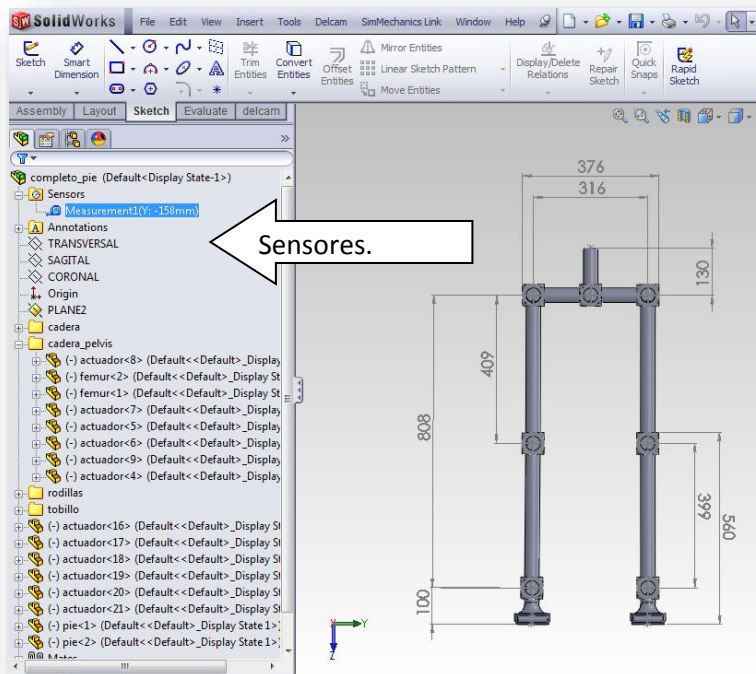


Fig. 2-14 Carpeta que contiene los sensores dentro del árbol de operaciones de Solidworks.

A pesar de la ayuda del software CAD, obtener los valores adecuados de D-H fue complicado debido a la gran cantidad de parámetros. Con la ayuda de los valores obtenidos del software CAD se pudo ir comprobando los valores D-H.

2.4. Modelo dinámico de un robot serial

El caminar humano puede ser descrito como una pérdida y recuperación del balance. Para evitar caer tenemos que posicionar el pie a una nueva locación, y así sucesivamente durante todo el ciclo.

Durante el caminado existen fuerzas generadas por el movimiento y la por gravedad. Entender y aplicar la dinámica que afecta el caminado en una estructura robótica es de suma importancia, ya que solamente con un estudio cinemático no se puede tener la certeza de que esa estructura pueda caminar.

Los robots manipuladores pueden ser descritos matemáticamente en diferentes formas. El problema de la cinemática es describir el movimiento de un manipulador dejando a un lado las fuerzas y torques que causan el movimiento. La posición del efector final es determinada por las variables de las articulaciones (cinemática directa).

El problema del modelo dinámico es encontrar una relación entre las fuerzas aplicadas y el movimiento del robot.

Existen dos diferentes métodos del cálculo del modelo dinámico: Euler-Lagrange y Newton-Euler. Para el desarrollo de la tesis se decidió utilizar el método de Euler-Lagrange. Esto debido a que el número de parámetros para describir el modelo es reducido. Muchas veces esta característica es deseable para obtener un modelo dinámico. Además es un método del cual se tiene dominio.

Con base en (Gonzalez-Galvan, 2012) se presentan las fórmulas siguientes para obtener la dinámica de un robot:

Energía cinética $T(\dot{q}, q)$.

Energía potencial $V(q)$.

con q el vector de variables articulares.

Por lo tanto el Lagrangiano L , $L = T - V$.

Por lo tanto:

$$L(q, \dot{q}) = T(q, \dot{q}) - V(q)$$

$$\frac{d}{dt} \left(\frac{\partial L}{\partial \dot{q}_i} \right) - \frac{\partial L}{\partial q_i} = \frac{d}{dt} \left(\frac{\partial T}{\partial \dot{q}_i} \right) - \left(\frac{\partial T}{\partial q_i} - \frac{\partial V}{\partial q_i} \right) = \tau_i \quad \text{Ec.[2.2]}$$

A la ecuación Ec.[2.2] se le conoce como Euler-Lagrange. Donde τ_i representa el torque para cada articulación del mecanismo \dot{q} , q son la velocidad y posición de las articulaciones.

2.4.1. Velocidades de las articulaciones

Para determinar la energía cinética primero debemos calcular la velocidad del eslabón. Debido a que la posición de cada eslabón está dada por los parámetros de Denavit- Hartenberg (D-H) encontrar la velocidad articular de cada eslabón lleva un proceso.

Considere un punto (X_i, Y_i, Z_i) definido respecto a un sistema x_i, y_i, z_i . Este punto es fijo con respecto a este sistema. La posición de este punto con respecto a un sistema inercial se define como:

$$\begin{bmatrix} X_0 \\ Y_0 \\ X_0 \\ 1 \end{bmatrix} = A_1 A_2 \dots A_i \begin{bmatrix} X_i \\ Y_i \\ Z_i \\ 1 \end{bmatrix}$$

donde A_i son matrices de transformación homogéneas.

La velocidad de este punto con respecto al sistema inercial es:

$$\mathbf{V}_i := \begin{bmatrix} \dot{X}_0 \\ \dot{Y}_0 \\ \dot{Z}_0 \\ 0 \end{bmatrix}$$

Tomando en cuenta que el último sistema de coordenado es fijo y por lo tanto es constante se tiene que la velocidad para este punto es:

$$\begin{bmatrix} \dot{X}_i \\ \dot{Y}_i \\ \dot{Z}_i \\ 0 \end{bmatrix} = 0$$

Para encontrar la velocidad V_i utilizando la regla del producto se tiene que:

$$\begin{aligned} V_i &= \dot{A}_1 A_2 \dots A_i \begin{bmatrix} X_i \\ Y_i \\ Z_i \\ 1 \end{bmatrix} + A_1 \dot{A}_2 \dots A_i \begin{bmatrix} X_i \\ Y_i \\ Z_i \\ 1 \end{bmatrix} + A_1 A_2 \dots \dot{A}_i \begin{bmatrix} X_i \\ Y_i \\ Z_i \\ 1 \end{bmatrix} \\ &= (\dot{A}_1 A_2 \dots A_i + A_1 \dot{A}_2 \dots A_i + A_1 A_2 \dots \dot{A}_i) \begin{bmatrix} X_i \\ Y_i \\ Z_i \\ 1 \end{bmatrix} \\ &= \left(\frac{\partial A_1}{\partial q_1} \dot{q}_1 A_2 \dots A_i + A_1 \frac{\partial A_2}{\partial q_2} \dot{q}_2 \dots A_i + A_1 A_2 \dots \frac{\partial A_i}{\partial q_i} \dot{q}_i \right) \begin{bmatrix} X_i \\ Y_i \\ Z_i \\ 1 \end{bmatrix} \\ &= \sum_{j=1}^i \frac{\partial}{\partial q_j} (A_1 A_2 \dots A_i) \dot{q}_j \begin{bmatrix} X_i \\ Y_i \\ Z_i \\ 1 \end{bmatrix} \end{aligned}$$

Considerando la convención de D-H, cada matriz A_i puede definirse como:

$$A_i = \begin{bmatrix} \cos\theta_i & -\text{sen}\theta_i \cos\alpha_i & \text{sen}\theta_i \text{sen}\alpha_i & \alpha_i \cos\theta_i \\ \text{sen}\theta_i & \cos\theta_i \cos\alpha_i & -\cos\theta_i \text{sen}\alpha_i & \alpha_i \text{sen}\theta_i \\ 0 & \text{sen}\alpha_i & \cos\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

De esta forma para obtener la velocidad de la articulación

$$\frac{\partial A_i}{\partial q_i} = \frac{\partial A_i}{\partial \theta_i} = \begin{bmatrix} -\text{sen}\theta_i & -\cos\theta_i \cos\alpha_i & \cos\theta_i \text{sen}\alpha_i & -\alpha_i \text{sen}\theta_i \\ \cos\theta_i & -\text{sen}\theta_i \cos\alpha_i & \text{sen}\theta_i \text{sen}\alpha_i & \alpha_i \cos\theta_i \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad \text{Ec.[2.3]}$$

Podemos definir la siguiente función

$$U_{ij} := \begin{cases} \frac{\partial}{\partial q_j} (A_1 A_2 \dots A_i) & \text{para } j \leq i \\ 0 & \text{para } j > i \end{cases}$$

Así,

$$v_i = \sum_{j=1}^i U_{ij} q_j \begin{bmatrix} X_i \\ Y_i \\ Z_i \\ 1 \end{bmatrix}$$

Por conveniencia se define también

$$U_{ijk} := \frac{\partial^2}{\partial q_j \partial q_k} (A_1 A_2 \dots A_i) = \begin{cases} \left(A_1 A_2 \frac{\partial A_j}{\partial q_j} \dots \frac{\partial A_k}{\partial q_k} \dots A_i \right) & \text{para } j \leq k \leq i \\ \left(A_1 A_2 \frac{\partial A_k}{\partial q_k} \dots \frac{\partial A_j}{\partial q_j} \dots A_i \right) & \text{para } k \leq j \leq i \\ 0 & \text{para } j > i \text{ ó } k > i \end{cases} \quad \text{Ec.[2.4]}$$

2.4.2. Energía cinética

De la misma forma (Gonzalez-Galvan, 2012) se puede calcular la energía cinética para cada eslabón de la siguiente forma.

Sea T_i la energía cinética del eslabón i y sea dT_i la energía cinética de un elemento diferencial en el mismo eslabón, con una masa dm .

Así,

$$dT_i = \frac{1}{2} (\dot{x}_i + \dot{y}_i + \dot{z}_i) dm = \frac{1}{2} \text{tr}(\mathbf{V}_i \mathbf{V}_i^t) dm$$

Donde $\text{tr}(\mathbf{O})$ representa la traza de una matriz. A partir de la ecuación anterior se encuentra que

$$\begin{aligned} dT_i &= \frac{1}{2} \text{tr} \left\{ \sum_{p=1}^i U_{ip} \dot{q}_p \begin{bmatrix} X_i \\ Y_i \\ Z_i \\ 1 \end{bmatrix} \sum_{r=1}^i \begin{bmatrix} X_i \\ Y_i \\ Z_i \\ 1 \end{bmatrix} U_{ir}^t \dot{q}_r \right\} dm \\ &= \frac{1}{2} \text{tr} \left\{ \sum_{p=1}^i \sum_{r=1}^i U_{ip} \begin{bmatrix} X_i \\ Y_i \\ Z_i \\ 1 \end{bmatrix} \begin{bmatrix} X_i \\ Y_i \\ Z_i \\ 1 \end{bmatrix}^t U_{ir}^t \dot{q}_p \dot{q}_r \right\} dm \\ &= \frac{1}{2} \text{tr} \left\{ \sum_{p=1}^i \sum_{r=1}^i U_{ip} \begin{bmatrix} X_i^2 dm & X_i Y_i dm & X_i Z_i dm & X_i dm \\ Y_i X_i dm & Y_i^2 dm & Y_i Z_i dm & Y_i dm \\ Z_i X_i dm & Z_i Y_i dm & Z_i^2 dm & Z_i dm \\ X_i dm & Y_i dm & Z_i dm & dm \end{bmatrix} U_{ir}^t \dot{q}_p \dot{q}_r \right\} dm \end{aligned}$$

Integrando sobre el volumen del eslabón i , se obtiene

$$T_i = \int_v dT_i = \frac{1}{2} \text{tr} \left[\sum_{p=1}^i \sum_{r=1}^i U_{ip} J_i U_{ir}^t \dot{q}_p \dot{q}_r \right]$$

Donde J_i representa la matriz de inercia. Entonces la energía cinética total T de un brazo robot es:

$$\begin{aligned} T &= \sum_{i=1}^n T_i = \frac{1}{2} \sum_{i=1}^n \text{tr} \left[\sum_{p=1}^i \sum_{r=1}^i U_{ip} J_i U_{ir}^t \dot{q}_p \dot{q}_r \right] \\ T &= \frac{1}{2} \sum_{i=1}^n \sum_{p=1}^i \sum_{r=1}^i [\text{tr}(U_{ip} J_i U_{ir}^t) \dot{q}_p \dot{q}_r] \end{aligned} \quad \text{Ec.[2.5]}$$

2.4.3. Energía potencial

Sea V la energía potencial total de un robot y V_i la energía potencial del eslabón i , entonces

$$V_i = -m_i \begin{bmatrix} gx \\ gy \\ gz \\ 0 \end{bmatrix} A_1 A_2 \dots A_i \begin{bmatrix} \bar{X}_i \\ \bar{Y}_i \\ \bar{Z}_i \\ 1 \end{bmatrix}$$

Donde g_x, g_y, g_z representan los componentes de gravedad con respecto al sistema coordenado inercial (en la base del robot), mientras que $\bar{x}_i, \bar{y}_i, \bar{z}_i$ son coordenadas del centro de masa del eslabón con respecto al sistema $X_i Y_i Z_i$.

La energía potencial total del brazo es:

$$V = \sum_{i=1}^n V_i = \sum_{i=1}^n -m_i \begin{bmatrix} g_x \\ g_y \\ g_z \\ 0 \end{bmatrix}^T A_1 A_2 \dots A_i \begin{bmatrix} \bar{X}_i \\ \bar{Y}_i \\ \bar{Z}_i \\ 1 \end{bmatrix} \quad \text{Ec.[2.6]}$$

con m_i la masa del i -ésimo eslabón.

2.4.4. Ecuaciones de movimiento

Una vez definida la energía cinética y potencial se define el Lagrangiano:

$$L = T - V = \frac{1}{2} \sum_{i=1}^n \sum_{p=1}^i \sum_{r=1}^i [\text{tr}(U_{ip} J_i U_{ir}^t) \dot{q}_p \dot{q}_r] - \sum_{i=1}^n -m_i \begin{bmatrix} g_x \\ g_y \\ g_z \\ 0 \end{bmatrix}^T A_1 A_2 \dots A_i \begin{bmatrix} \bar{X}_i \\ \bar{Y}_i \\ \bar{Z}_i \\ 1 \end{bmatrix}$$

Utilizando las fórmulas de Euler-Lagrange aplicando la simplificación utilizada en [1] se tiene que;

$$\frac{d}{dt} \left(\frac{\partial L}{\partial \dot{q}_i} \right) - \frac{\partial L}{\partial q_i} = \tau$$

$$= \sum_{j=i}^n \sum_{k=1}^j (U_{jk} J_j U_{ji}^T) \ddot{q}_k + \sum_{j=i}^n \sum_{k=1}^j \sum_{m=1}^j \text{tr}(U_{jkm} J_j U_{ji}^T) \dot{q}_k \dot{q}_m - \sum_{j=1}^n m_j \begin{bmatrix} g_x \\ g_y \\ g_z \\ 0 \end{bmatrix}^T U_{ji} \begin{bmatrix} \bar{X}_i \\ \bar{Y}_i \\ \bar{Z}_i \\ 1 \end{bmatrix}$$

Esta última expresión puede escribirse de manera más sencilla como sigue:

$$\tau_i = \sum_{k=1}^n D_{ik} \ddot{q}_k + \sum_{k=1}^n \sum_{m=1}^n h_{ikm} \dot{q}_k \dot{q}_m + c_i \quad i = 1, 2, \dots, n \quad \text{Ec.[2.7]}$$

con $D_{ik} = \sum_{j=\max(i,k)}^n \text{tr}(U_{jk} J_j U_{ji}^T)$; $i, k = 1, 2, \dots, n$

2.5. Ejemplo 1 de aplicación: manipulador (RR)

Para entender mejor el concepto de las ecuaciones de movimiento de un manipulador se realizó un ejemplo con un mecanismo de dos grados de libertad. Como el que se muestra en la Fig. 2-15.

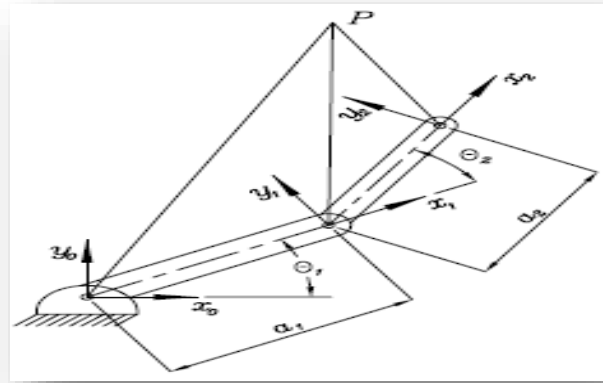


Fig. 2-15 Arquitectura serial de 2 grados de libertad (RR).

Primero se determinan los parámetros de Denavit-Hartenberg (D-H) utilizando la Ec.[2.1], de esta forma podemos determinar las matrices de transformación homogénea para cada eslabón. Las matrices quedan de la siguiente manera

$$A_1 = \begin{bmatrix} \cos(\theta_1) & -\sin(\theta_1) & 0 & a1 * \cos(\theta_1) \\ \sin(\theta_1) & \cos(\theta_1) & 0 & a1 * \sin(\theta_1) \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$A_2 = \begin{bmatrix} \cos(\theta_2) & -\sin(\theta_2) & 0 & a2 * \cos(\theta_2) \\ \sin(\theta_2) & \cos(\theta_2) & 0 & a2 * \sin(\theta_2) \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Una vez que se tienen las matrices de transformación homogénea se proceden a calcular cada uno de los parámetros para encontrar la torsión de cada articulación.

Primero se calculan los parámetros U_{ij} para poder encontrar este parámetro se tiene que derivar con respecto a cada variable (θ_i para $i = 1, 2 \dots n$ y $j = 1, 2, \dots, n$), donde n representa el número de variables. Apoyándose en la Ec.[2.3] se pueden obtener las velocidades. De esta manera los parámetros U_{ij} quedan de la siguiente manera

$$U_{11} = \begin{bmatrix} -\sin(\theta_1) & -\cos(\theta_1) & 0 & -a1 * \sin(\theta_1) \\ \cos(\theta_1) & -\sin(\theta_1) & 0 & a1 * \cos(\theta_1) \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

$$U_{12} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

$$U_{21} = \begin{bmatrix} -\sin(\theta_1 + \theta_2) & -\cos(\theta_1 + \theta_2) & 0 & -a2 * \sin(\theta_1 + \theta_2) - a1 * \sin(\theta_1) \\ \cos(\theta_1 + \theta_2) & -\sin(\theta_1 + \theta_2) & 0 & a2 * \cos(\theta_1 + \theta_2) + a1 * \cos(\theta_1) \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

$$U_{22} = \begin{bmatrix} -\sin(\theta_1 + \theta_2) & -\cos(\theta_1 + \theta_2) & 0 & -a2 * \sin(\theta_1 + \theta_2) \\ \cos(\theta_1 + \theta_2) & -\sin(\theta_1 + \theta_2) & 0 & a2 * \cos(\theta_1 + \theta_2) \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

En relación a la matriz de inercia, suponemos que cada eslabón i consiste de una barra delgada de masa m_i , cuyo centro de masa está ubicado a la mitad del eslabón. Considerando la ubicación de los sistemas coordenados en la figura anterior.

$$J_1 = \begin{bmatrix} I1 & 0 & 0 & -(a1 * m1)/2 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ -(a1 * m1)/2 & 0 & 0 & m1 \end{bmatrix}$$

$$J_2 = \begin{bmatrix} I2 & 0 & 0 & -(a2 * m2)/2 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ -(a2 * m2)/2 & 0 & 0 & m2 \end{bmatrix}$$

Una vez que se obtuvo la matriz de inercia se puede calcular la matriz cinética de inercia (D_{ik})

$$D_{11} = \sum_{j=1}^2 \text{tr}(U_{j1} J_j U_{j1}^T) = m2 * a1^2 + a2 * m2 * \cos(\theta_2) * a1 + I1 + I2$$

$$D_{12} = \sum_{j=2}^2 \text{tr}(U_{j2} J_j U_{j2}^T) = I2 + (a1 * a2 * m2 * \cos(\theta_2))/2$$

$$D_{21} = \sum_{j=2}^2 \text{tr}(U_{j1} J_j U_{j2}^T) = I2 + (a1 * a2 * m2 * \cos(\theta_2))/2$$

$$D_{22} = \sum_{j=2}^2 \text{tr}(U_{j2} J_j U_{j2}^T) = I2$$

El vector de fuerza de Coriolis y de fuerza centrífuga se forma con los siguientes elementos

$$h_{i(q,\dot{q})} = \sum_{k=1}^n \sum_{m=1}^n h_{ikm} \dot{q}_k \dot{q}_m ; \quad i = 1, 2, \dots, n$$

$$h_{ikm} = \sum_{j=\max(i,k,m)}^n \text{tr}(U_{jkm} J_j U_{ji}^T); \quad i, k, m = 1, 2, \dots, n$$

Primero hay que encontrar los valores U_{jkm} para los parámetros h_{ikm} las matrices quedan de la siguiente manera

$$U_{111} = \begin{bmatrix} -\cos(\theta_1) & \sin(\theta_1) & 0 & -a1 * \cos(\theta_1) \\ -\sin(\theta_1) & -\cos(\theta_1) & 0 & -a1 * \sin(\theta_1) \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

$$U_{112} = U_{121} = U_{122} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

$$U_{211} = \begin{bmatrix} -\cos(\theta_1 + \theta_2) & \sin(\theta_1 + \theta_2) & 0 & -a2 * \cos(\theta_1 + \theta_2) - a1 * \cos(\theta_1) \\ -\sin(\theta_1 + \theta_2) & -\cos(\theta_1 + \theta_2) & 0 & -a2 * \sin(\theta_1 + \theta_2) - a1 * \sin(\theta_1) \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

$$U_{212} = U_{221} = U_{222} = \begin{bmatrix} -\cos(\theta_1 + \theta_2) & \sin(\theta_1 + \theta_2) & 0 & -a2 * \cos(\theta_1 + \theta_2) \\ -\sin(\theta_1 + \theta_2) & -\cos(\theta_1 + \theta_2) & 0 & -a2 * \sin(\theta_1 + \theta_2) \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

De esta forma h_{ikm} queda de la siguiente manera

$$\begin{aligned} h_{111} &= h_{221} = h_{212} = h_{222} = 0 \\ h_{121} &= -(a1 * a2 * m2 * \sin(\theta_2))/2 \\ h_{211} &= (a1 * a2 * m2 * \sin(\theta_2))/2 \\ h_{112} &= -(a1 * a2 * m2 * \sin(\theta_2))/2 \\ h_{122} &= -(a1 * a2 * m2 * \sin(\theta_2))/2 \end{aligned}$$

Una vez que se calculó el parámetro h_{ikm} se procede a calcular

$$\begin{aligned} h_1 &= -(a1 * a2 * m2 * \dot{\theta}_2 * \sin(\theta_2) * (2 * \dot{\theta}_1 + \dot{\theta}_2))/2 \\ h_2 &= -(a1 * a2 * m2 * \dot{\theta}_2 * \sin(\theta_2) * (2 * \dot{\theta}_1 + \dot{\theta}_2))/2 \end{aligned}$$

Por último queda evaluar el vector de carga gravitatoria. El vector de gravedad tiene componente en la dirección Y0

$$\begin{bmatrix} g_x \\ g_y \\ g_z \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ -9.8 = g \\ 0 \\ 0 \end{bmatrix}$$

Los valores de c_i quedan de la siguiente manera

$$c_1 = (g * (a1 * m1 * \cos(\theta_1) + 2 * a1 * m2 * \cos(\theta_1) + a2 * m2 * \cos(\theta_1 + \theta_2)))/2$$

$$c_2 = (a2 * g * m2 * \cos(\theta_1 + \theta_2))/2$$

Sustituyendo en la Ec.[2.7] se tiene que los torques τ_1, τ_2 son:

$$\tau_1 = (g * (a1 * m1 * \cos(\theta_1) + 2 * a1 * m2 * \cos(\theta_1) + a2 * m2 * \cos(\theta_1 + \theta_2)))/2 + \dot{\theta}_1 * (m2 * a1^2 + a2 * m2 * \cos(\theta_2) * a1 + I1 + I2) + \ddot{\theta}_2 * (I2 + (a1 * a2 * m2 * \cos(\theta_2))/2) - (a1 * a2 * m2 * \dot{\theta}_2 * \sin(\theta_2) * (2 * \dot{\theta}_1 + \dot{\theta}_2))/2$$

Ec.[2.8]

$$\tau_2 = (a1 * a2 * m2 * \sin(\theta_2) * \dot{\theta}_1^2)/2 + I2 * \ddot{\theta}_2 + \dot{\theta}_1 * (I2 + (a1 * a2 * m2 * \cos(\theta_2))/2) + (a2 * g * m2 * \cos(\theta_1 + \theta_2))/2$$

Ec.[2.9]

2.6. Ejemplo 2 de aplicación: manipulador (RP)

En el siguiente ejemplo se maneja un robot de 2 grados de libertad, el cual tiene una articulación de revoluta y una prismática (RP). El robot tomado para este ejemplo se muestra en la Fig. 2-16.

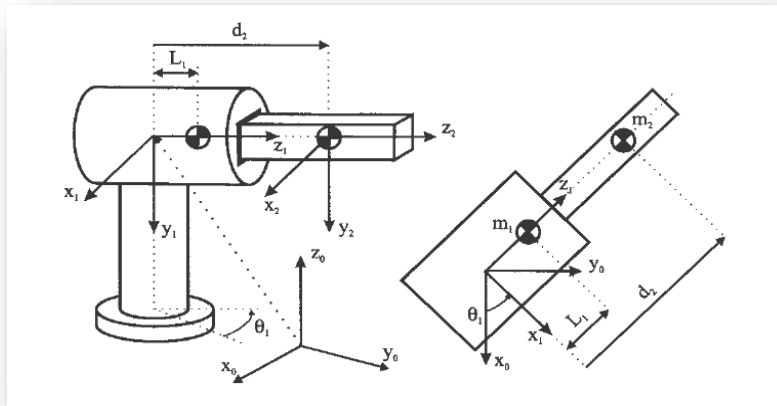


Fig. 2-16 Arquitectura serial de 2 grados de libertad (RP).

Los datos para poder establecer el análisis Euler-Lagrange se muestran en la parte inferior. Para las matrices homogéneas A_i se utilizaron los parámetros de Denavit-Hartenberg (D-H). Las matrices J_i representan las matrices de inercia de cada eslabón.

$$A_1 = \begin{bmatrix} \cos(\theta_1) & 0 & -\sin(\theta_1) & 0 \\ \sin(\theta_1) & 0 & \cos(\theta_1) & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad J_1 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & L_1^2 * m_1 & L_1 * m_1 \\ 0 & 0 & L_1 * m_1 & m_1 \end{bmatrix}$$

$$A_2 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d2 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad J_2 = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & m_2 \end{bmatrix}$$

Puesto que se considera que la masa concentrada en el centro y el origen del sistema de coordenadas del eslabón 2 (prismático) se toma en el mismo centro de masas la matriz J_2 queda con un solo elemento como se muestra en la fórmula anterior.

Utilizando el mismo método para encontrar la ecuación dinámica que se utilizó en el ejemplo 1 se muestran los resultados para el ejemplo 2.

Matrices U_{ij}

$$U_{11} = \begin{bmatrix} -\sin(\theta_1) & 0 & -\cos(\theta_1) & 0 \\ \cos(\theta_1) & 0 & -\sin(\theta_1) & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

$$U_{12} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

$$U_{21} = \begin{bmatrix} -\sin(\theta_1) & 0 & -\cos(\theta_1) & -d_2 * \cos(\theta_1) \\ \cos(\theta_1) & 0 & -\sin(\theta_1) & -d_2 * \sin(\theta_1) \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

$$U_{22} = \begin{bmatrix} 0 & 0 & 0 & -\sin(\theta_1) \\ 0 & 0 & 0 & \cos(\theta_1) \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

Matrices U_{ijk}

$$U_{111} = \begin{bmatrix} -\cos(\theta_1) & 0 & \sin(\theta_1) & 0 \\ -\sin(\theta_1) & 0 & -\cos(\theta_1) & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

$$U_{112} = U_{121} = U_{122} = U_{222} = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

$$U_{211} = \begin{bmatrix} -\cos(\theta_1) & 0 & \sin(\theta_1) & d_2 * \sin(\theta_1) \\ -\sin(\theta_1) & 0 & -\cos(\theta_1) & -d_2 * \cos(\theta_1) \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

$$U_{212} = U_{221} = \begin{bmatrix} 0 & 0 & 0 & -\cos(\theta_1) \\ 0 & 0 & 0 & -\sin(\theta_1) \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

Matriz de inercia D_{ik}

$$D_{ik} = \begin{bmatrix} L_1^2 * m_1 + m_2 * d_2^2 & 0 \\ 0 & m_2 \end{bmatrix}$$

Términos h_{ikm}

$$h_{111} = h_{221} = h_{212} = h_{222} = h_{122} = 0$$

$$h_{121} = d_2 * m_2$$

$$h_{211} = -d_2 * m_2$$

$$h_{112} = d_2 * m_2$$

Términos $h(q, \dot{q})$ vector de fuerza de Coriolis y centrífuga

$$h_1 = 2 * d_2 * \dot{d}_2 * m_2 * \dot{\theta}_1$$

$$h_2 = -d_2 * m_2 * \dot{\theta}_1^2$$

Para este segundo ejemplo el vector gravedad está en la dirección z

$$\begin{bmatrix} g_x \\ g_y \\ g_z \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ -9.8 = g \\ 0 \end{bmatrix}$$

Los vectores de coordenadas homogéneas de posición del centro de masas para cada eslabón

$$r_1 = \begin{bmatrix} 0 \\ 0 \\ L_1 \\ 1 \end{bmatrix} \quad r_2 = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}$$

Los valores de c_i quedan de la siguiente manera

$$c_1 = c_2 = 0.$$

Por lo tanto la ecuación dinámica del robot será:

$$\tau_i = \sum_{k=1}^n D_{ik} \ddot{q}_k + h_i + c_i$$

$$\tau_1 = \ddot{\theta}_1 * (L_1^2 * m_1 + m_2 * d_2^2) + 2 * d_2 * \dot{d}_2 * m_2 * \dot{\theta}_1 \quad \text{Ec.[2.10]}$$

$$f_1 = m_2 * (\ddot{d}_2 - d_2 * \dot{\theta}_1^2) \quad \text{Ec.[2.11]}$$

2.7. Manipulador serial de n grados de libertad

A pesar de que en los ejemplos anteriores los robots seriales son de dos grados de libertad, los cálculos con el método de Euler-Lagrange requieren de bastantes operaciones con matrices.

Si tomamos en cuenta que para calcular los parámetros U_{ij} se necesitan n^2 matrices donde n es el número de grados de libertad que tiene el robot serial. De igual manera para calcular los parámetros U_{ijk} se necesitan n^3 matrices, para los parámetros D_{ik} una matriz de $n \times n$, para h_{ikm} se necesitan n^3 matrices, para h_i se necesitan n^1 términos y finalmente para los parámetros c_i se necesitan n^1 términos. El cálculo para un robot serial, por ejemplo de 4 grados de libertad necesitaríamos para los parámetros $U_{ijk} = 4^3 = 64$ matrices, lo mismo para los parámetros $h_{ikm} = 4^3 = 64$ matrices, simplemente para el manejo de estos dos términos se necesitan manejar 128 matrices de 4x4 términos.

Debido a que en el estudio de arquitecturas simplificadas del cuerpo humano requieren analizar diferentes arquitecturas. La implementación de esta metodología para calcular las ecuaciones dinámicas de manera más eficiente sería de suma importancia. De lo contrario para calcular la dinámica de un robot se tendría que manejar todas las matrices de manera manual. Lo cual lo volvería muy complicado o prácticamente imposible para robots de muchos grados de libertad.

Para obtener las ecuaciones dinámicas de un robot se desarrolló un programa con la ayuda del software MATLAB®. Con un programa que calculara la dinámica de un robot serial se podrían obtener ecuaciones en un tiempo razonable y se podrían manejar robots seriales con un número mayor de grados de libertad. De esta manera la carga de realizar todas las operaciones sería del programa.

2.8. Desarrollo de un programa para obtener las ecuaciones dinámicas de un manipulador serial

Como se vio en los ejemplos anteriores obtener las ecuaciones dinámicas de un robot serial por medio del método de Euler-Lagrange es relativamente sencillo para robots seriales de 2 grados de libertad, cuando se tiene dominado el tema. Sin embargo para poder continuar con nuestro estudio se requiere de un análisis más complejo.

El método utilizado anteriormente se puede programar utilizando la versatilidad que tiene MATLAB® para el manejo de variables simbólicas.

Symbolic Math Toolbox proporciona un completo conjunto de herramientas para el cálculo simbólico. Esta Toolbox ofrece una amplia funcionalidad simbólica a la que se puede acceder directamente desde la línea de comandos de MATLAB® o mediante la aplicación MuPAD Notebook.

Utilizando la herramienta se programó el método de Euler-Lagrange de forma simbólica. Para que el programa funcione solo se necesitan 3 datos para calcular la ecuación dinámica del robot serial. Se necesitan los parámetros D-H los cuales nos dan la posición de todos los eslabones con respecto a un marco de referencia, la matriz de inercia de cada eslabón y el vector posición del centro de masa del eslabón con respecto a su sistema de coordenadas.

Para seguir con la descripción del programa abordaremos el último ejemplo de robot serial el robot (RP) de dos grados de libertad, en donde se muestran bloques del programa desarrollado. El programa completo se encuentra en el CD de apoyo de la tesis.

El primer paso es obtener los parámetros D-H para calcular las matrices homogéneas en este caso se programó una función (ver anexo A-1.A-1) que calculara la matriz homogénea de acuerdo a los datos proporcionados para cada sistema de referencia. Se necesitan 4 parámetros para la función θ_i, ϕ_i, d_i y a_i .

De esta manera al introducir los datos en la función D-H se tiene la matriz homogénea de manera simbólica. Los 2 datos restantes se introducen de manera manual como se muestra en la Fig. 2-17.

Matrices de inercia y vectores de centro de masa de cada eslabón.

```
J(:, :, 1)=[0 0 0 0;0 0 0 0;0 0 m1*L1^2 m1*L1;0 0 m1*L1 m1];
J(:, :, 2)=[0 0 0 0;0 0 0 0;0 0 0 0;0 0 0 m2];

com1=[0;0;L1;1];
com2=[0;0;0;1];
```

Fig. 2-17 Parámetros necesarios en el programa para calcular la ecuación dinámica.

Una vez que se tienen todos los datos para calcular la ecuación dinámica se empieza calculando los parámetros U_{ij} . Estos parámetros corresponden a la primera derivada. Como se muestra en la Fig. 2-18 se utiliza la función *diff* de Matlab para calcular la primera derivada.

```
function uu=Uij(i,j,A,q)
% comienza condición
if j<=i%-----si j<= continua con cálculos
for s=1:i
if s==1 AA=A(:, :, 1);% para la primera iteración condición
else
AA=AA*A(:, :, i);% multiplicación de las matrices
end
end
uu=diff(AA,q(j));% derivada con respecto a la variable qi
else%-----si j>i uu=0
uu=zeros(4);
end
```

Fig. 2-18 Cálculo de parámetro Uij.

Siguiendo el procedimiento de los ejemplos de mecanismos seriales de dos grados de libertad donde se calculó la ecuación dinámica, se realizó el programa en MATLAB® utilizando las ecuaciones descritas anteriormente. En los siguientes bloques se muestra una descripción rápida para cada ciclo.

Para obtener los parámetros U_{ijk} (aceleración) se realizó un ciclo para n elementos. En el caso de los ejemplos se tienen 2 matrices D-H esto debido a los dos sistemas coordinados. Como se muestra en el Fig. 2-19 se calcula la segunda derivada para las matrices A_i .

$$U_{ijk} = \frac{\partial^2}{\partial q_j \partial q_k} (A_1 A_2 \dots A_i) = \begin{cases} \left(A_1 A_2 \frac{\partial A_j}{\partial q_j} \dots \frac{\partial A_k}{\partial q_k} \dots A_i \right) & \text{para } j \leq k \leq i \\ \left(A_1 A_2 \frac{\partial A_k}{\partial q_k} \dots \frac{\partial A_j}{\partial q_j} \dots A_i \right) & \text{para } k \leq j \leq i \\ 0 & \text{para } j > i \text{ ó } k > i \end{cases}$$

```

% Uijk
% comienza ciclo de iteraciones
cont=1;
for i=1:n
    for j=1:n
        for k=1:n
            %condición si cumple no existe la segunda derivada
            if or(j>i,k>i)
                Uijk(:,:,cont)=zeros(4);
            else
                Uijk(:,:,cont)=diff(U(:,:,ij(i,j,n)),q(k));
                %realiza la derivada de la matriz
            end
            IJJ=ij(i,j,n);
            cont=cont+1;
        end
    end
end

```

Fig. 2-19 Cálculo de parámetro Uijk.

Los parámetros D_{ik} se guardan en una sola matriz de $n \times n$ elementos. En el caso de los ejemplos anteriores tendríamos una matriz de 2×2 elementos. En este ciclo se utiliza la función *trace* que calcula la traza de la multiplicación de las matrices como se muestra en la Fig. 2-20.

$$D_{ik} = \sum_{j=\max(i,k)}^n \text{tr}(U_{jk} J_j U_{ji}^T); \quad i, k = 1, 2, \dots, n$$

```

%n=2 Dik      j=max(i,k)
suma=0;
%ciclo que varia i,k=1,2,3.....n
for i=1:n
    for k=1:n
        %condiciones para saber j=max(i,k)
        j=max([i k]);
        % fin de condición
        for s=j:n
            %suma ( trace ( Ujk * Jj * Uji.' ) ) donde j=s
            suma=trace(U(:,:,ij(s,k,n))*J(:,:,s)*U(:,:,ij(s,i,n)).')+suma;
        end
        D(i,k)=simplify(suma);
        suma=0;
    end
end
end

```

Fig. 2-20 Cálculo de parámetro Dik.

Los términos h_{ikm} son guardados en n matrices de $n \times n$. Cada matriz representa los datos para el n eslabón. En la Fig. 2-21 se muestra una rápida descripción del ciclo que obtiene estos términos.

```

% Hikm
suma=0;
for i=1:n
    for k=1:n
        for m=1:n
            j_max=max([i k m]); % el máximo de (i,k,m,)
            for j=j_max:n
                %suma se guarda el valor de sumatoria de la traza de la multiplicación de
                % la multiplicación de matrices
                suma=(trace(Uijk(:, :, ijk(j,k,m,n)) * J(:, :, j) * U(:, :, ij(j,i,n)).') + suma);
            end
            H(i,k,m)=simplify(suma); % guardar el resultado
            suma=0;
        end
    end
end

```

Fig. 2-21 Cálculo de parámetro hikm.

Los términos de h_i se calculan de acuerdo a la Ec.[2.7]. Estos términos se guardan en una matriz de $1 \times n$.

$$h_i = \sum_{k=1}^n \sum_{m=1}^n h_{ikm} \dot{q}_k \dot{q}_m ; \quad i = 1, 2, \dots, n$$

```

% Hi
qp=[t1p d2p]; % vector que contiene velocidades de variables
suma=0;
for i=1:n
    for k=1:n
        for m=1:n
            suma=H(i,k,m)*qp(k)*qp(m)+suma; % guardar valores de
            % multiplicación en suma
        end
    end
    h(i)=simplify(suma); % matriz de parámetros hi
    suma=0;
end

```

Fig. 2-22 Cálculo de parámetro hi.

Para calcular el último término c_i se utilizan las coordenadas de los centros de masa de cada eslabón con respecto a su sistema coordenado.

$$C_i = \sum_{j=1}^n m_j \begin{bmatrix} g_x \\ g_y \\ g_z \\ 0 \end{bmatrix}^T U_{ji} \begin{bmatrix} x_i \\ y_i \\ z_i \\ 1 \end{bmatrix}$$

```

% vector de cargas por gravedad energía potencial
G=[0;0;-g;0]; % dirección de gravedad con respecto a sistema XoYoZo
com1=[0;0;L1;1]; % vector posición de centro de más eslabón 1
com2=[0;0;0;1]; % vector posición de centro de más eslabón 2
com=[com1 com2]; % matriz de posiciones de centros de masa
m=[m1 m2];
suma=0;
for i=1:n
    for j=1:n
% suma contiene le valor de la sumatoria
        suma=(-m(j)*G.'*U(:, :, ij(j,i,n))*com(:,j)) + suma;
    end
    c(i)=simplify(suma); % guardar términos ci
    suma=0;
end

```

Fig. 2-23 Cálculo de parámetro c_i .

Gracias a la herramienta de Matlab *SYM* el programa desarrollado puede manejar ecuaciones simbólicas. El programa tiene las características de poder calcular la ecuación dinámica para n grados de libertad y configuraciones diferentes de mecanismo seriales. Aunque si quisiéramos analizar un mecanismo serial de un número considerable de grados de libertad, el costo computacional sería alto. El resultado de este programa sería ecuaciones muy difíciles de manejar en cuanto a un número mayor de términos.

Para la primera prueba que fue la de la Fig. 2-15, las ecuaciones dinámicas son conocidas por lo tanto se comprobó el resultado del programa con respecto a las ecuaciones obtenidas del análisis dinámico. Ec.[2.12] y Ec.[2.13] se muestran los resultados obtenidos mediante el software desarrollado.

$$\tau_1 = (g * (a1 * m1 * \cos(\theta_1) + 2 * a1 * m2 * \cos(\theta_1) + a2 * m2 * \cos(\theta_1 + \theta_2)))/2 + \ddot{\theta}_1 * (m2 * a1^2 + a2 * m2 * \cos(\theta_2) * a1 + I1 + I2) + \ddot{\theta}_2 * (I2 + (a1 * a2 * m2 * \cos(\theta_2))/2) - (a1 * a2 * m2 * \dot{\theta}_2 * \sin(\theta_2) * (2 * \dot{\theta}_1 + \dot{\theta}_2))/2 \quad \text{Ec.[2.12]}$$

$$\tau_2 = (a1 * a2 * m2 * \sin(\theta_2) * \dot{\theta}_1^2)/2 + I2 * \ddot{\theta}_2 + \dot{\theta}_1 * (I2 + (a1 * a2 * m2 * \cos(\theta_2))/2) + (a2 * g * m2 * \cos(\theta_1 + \theta_2))/2 \quad \text{Ec.[2.13]}$$

Las ecuaciones dinámicas obtenidas mediante el programa y las calculadas para el mecanismo serial son las mismas. Incluso en cuanto a la reducción de términos de la Ec.[2.8] y Ec.[2.9].

Para el segundo caso se realizó una validación del programa utilizando otro mecanismo serial en el cual se tuvieran sus ecuaciones dinámicas. En la Fig. 2-24 se muestra la segunda configuración utilizada para probar el programa.

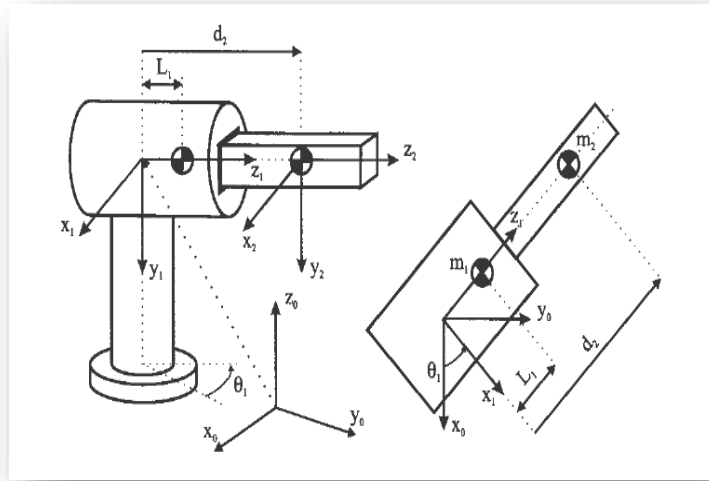


Fig. 2-24 Manipulador de 2 grados de libertad usado en el segundo ejemplo para validar plataforma en análisis dinámico.

Para este mecanismo serial se utilizó el mismo procedimiento. Los datos fueron introducidos al programa, por ejemplo sus matrices de D-H, posición de centro de masa, longitudes, etc. Se comprobó que el resultado fuera el mismo con respecto al análisis realizado previamente de manera analítica. En las Ec.[2.14] y Ec.[2.15] que se muestran, se tiene el resultado de la ejecución de este programa para el segundo mecanismo.

$$\tau_1 = \ddot{\theta}_1 * (L_1^2 * m_1 + m_2 * d_2^2) + 2 * d_2 * \dot{d}_2 * m_2 * \dot{\theta}_1 \quad \text{Ec.[2.14]}$$

$$f_1 = m_2 * (\ddot{d}_2 - d_2 * \dot{\theta}_1^2) \quad \text{Ec.[2.15]}$$

De la misma manera que en el ejemplo anterior, el resultado fue el mismo. Las ecuaciones dinámicas para el mecanismo calculadas de forma tradicional Ec.[2.10] y Ec.[2.11] son las mismas que las calculadas por el programa.

Sin embargo se notó que cuando se trata de analizar una arquitectura con un número mayor de grados de libertad las ecuaciones que resultan son muy largas. A pesar de que en el programa desarrollado se agregaron funciones para reducir términos como *simplify()* (ver, CD) las ecuaciones resultan largas y por tanto difíciles de manejar.

Para análisis mayores a 4 grados de libertad el costo computacional es alto, ya que la solución puede tardar varias horas.

Capítulo III

Desarrollo de una plataforma de simulación de arquitecturas bípedas

Como se vio en el capítulo anterior el estudio dinámico de los robots seriales es complejo y consume mucho tiempo. Debido a que un robot bípedo se modela como un robot serial con dos cadenas cinemáticas se vuelve aún más complejo ya que no solo se tiene que calcular la dinámica de un robot serial si no que se tiene que hacer dos veces, una para cada pierna. Calcular la dinámica de un robot bípedo implica el cálculo de muchas matrices como se mostró en el capítulo anterior. Por ejemplo si quisiéramos estudiar un robot con más de 2 grados de libertad como el caso de la arquitectura propuesta de 21 grados de libertad, se vuelve prácticamente imposible calcular la dinámica de un robot serial.

Por lo anterior para este capítulo se propuso crear una plataforma de simulación de robots bípedos. La plataforma de simulación nos permitirá realizar análisis de diferentes arquitecturas robóticas. Sin tener que desarrollar manualmente los cálculos dinámicos para cada arquitectura robótica que se quiera analizar.

Las características necesarias para desarrollar una plataforma, en donde se pueda simular a nivel dinámico diferentes arquitecturas robóticas, serían por ejemplo simular fuerzas, torques, masas, inercia, aceleraciones, velocidades, gravedad, etc. También se necesita la capacidad para dibujar geometrías complicadas, tener restricciones de grados de libertad, controlar la posición de cada articulación, saber la posición de cada objeto en todo momento, tener la posición del centro de masa, entre otras.

Otra característica importante es la capacidad para intercambiar información entre la plataforma y otros programas. En la plataforma se requiere facilidad a la hora de incorporar diferentes arquitecturas robóticas. Para simular un robot bípedo caminando en un entorno virtual se necesita controlar el robot por algoritmos avanzados.

Como se vio en el capítulo II, al momento de este estudio no existen plataformas con todas las características requeridas para simular la arquitectura propuesta. Por lo anterior, se optó por desarrollar una plataforma de simulación para robots bípedos basada en dos programas ampliamente conocidos como Solidworks y Matlab.

La unión de estos dos tipos de programas se les llama CADLab. CADLab es una plataforma donde se combinan dos programas con propiedades diferentes como análisis de mecánico (Solidworks) y análisis numérico (Matlab). CADLab nos ayuda a incluir las mejores propiedades y características de cada programa.

Al unir Matlab con Solidworks se pueden aprovechar todas las bibliotecas que tienen Matlab y la facilidad para crear elementos mecánicos que tiene Solidworks. Aparte Matlab cuenta con una extensa biblioteca para el desarrollo de sistemas de control y cuenta con amplias herramientas y funciones matemáticas. También cuenta con el entorno Simulink el cual permite simular elementos mecánicos en un entorno virtual.

Solidworks también cuenta con un entorno para el desarrollo de proyectos mecánicos muy potente. Permite el ensamble de máquinas complejas o con un número elevado de piezas. Sin embargo tanto Solidworks como Matlab son programas que cuentan con una gran cantidad de herramientas y para tener el dominio necesario para desarrollar la plataforma se requiere de un importante tiempo de estudio.

3.1. Herramientas para el desarrollo de la plataforma de simulación

Para llevar a cabo el desarrollo de la plataforma se requiere diseñar varios módulos (ver Fig. 3-1). El primer módulo como se muestra en la Fig. 3-1 consta de un diseño de un mecanismo robótico. Para el siguiente módulo se necesita la ayuda de un software comercial (Solidworks) del cual se obtendrá un modelo CAD, a partir del diseño del robot.

Un vez que se tiene el modelo CAD y con la ayuda de un segundo programa que permite modelar la dinámica de los mecanismo (Matlab-Simulink), se obtiene un programa a bloques que representa todo el ensamble de la arquitectura robótica (CadLab).

El programa a bloques que representa la arquitectura robótica tiene la capacidad de simular movimientos y la dinámica del mecanismo. Primero para lograr los movimientos de las articulaciones se recopilan los datos obtenidos del estudio del ciclo del caminado humano. Una vez que se tiene los datos de movimiento para las articulaciones se agregan actuadores a cada una.

Para obtener datos de posiciones, velocidades y aceleraciones de la simulación se agregan sensores en las articulaciones. Los sensores pueden obtener datos de la simulación para posteriormente trabajar con ellos.

Con la ayuda de los sensores se crea un sistema para simular un piso virtual el cual el robot puede utilizar como referencia para “apoyarse”. Durante la simulación se recopilan datos de posiciones de los pies y centro de masa. Por último se tiene un entorno visual donde se puede interactuar con la arquitectura robótica.

En la Fig. 3-1 se muestran los principales bloques que conforman la plataforma de simulación. En la parte superior se muestran los bloques (Solidworks y Matlab) los cuales son la base de la plataforma de simulación. A la izquierda del recuadro se encuentra una entrada (datos para movimiento del robot) y la salida de posiciones de los cuerpos en el espacio virtual.

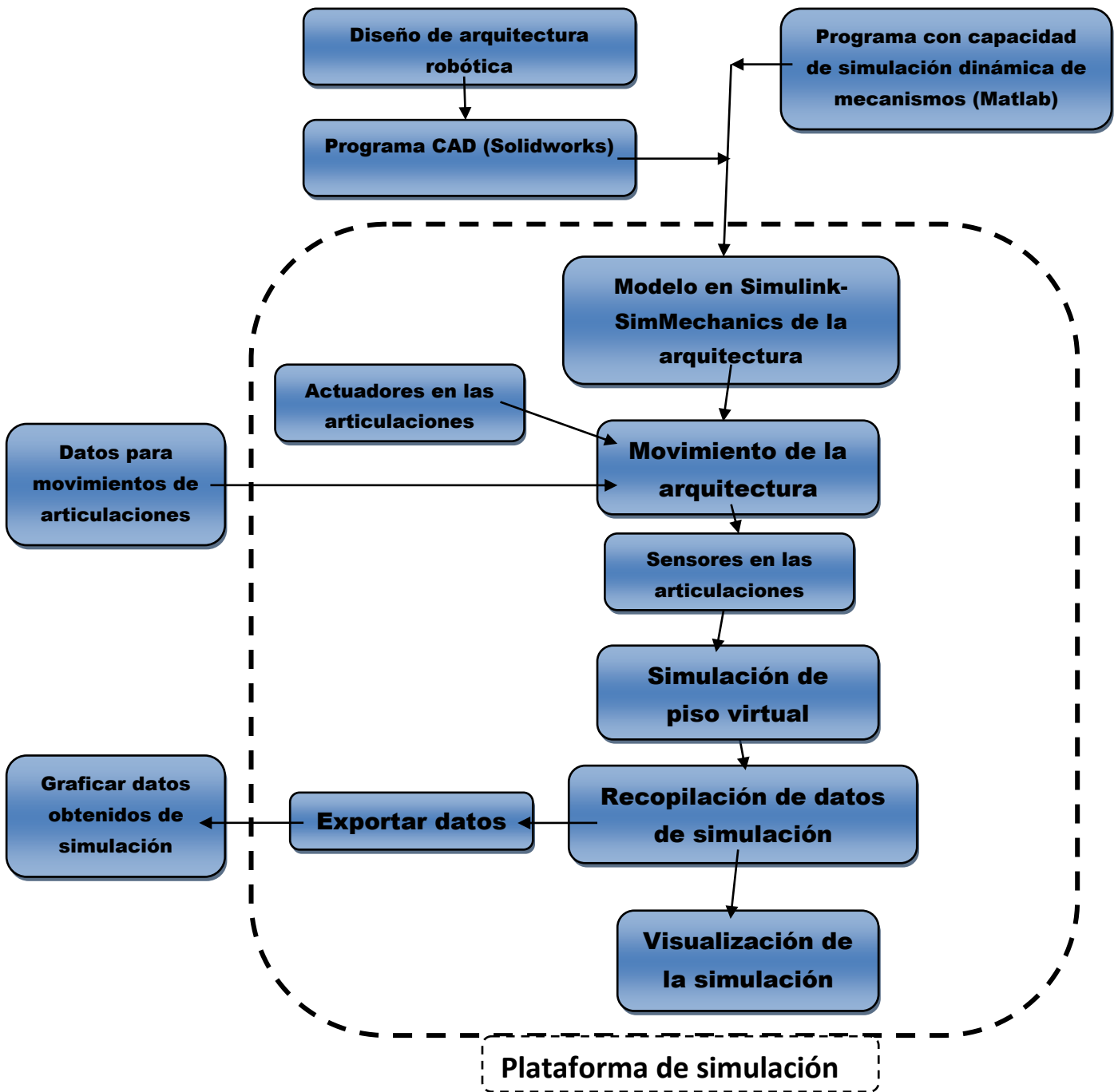


Fig. 3-1 Diagrama a bloques de plataforma de simulación.

3.1.1. Matlab

Uno de los programas que fueron la base para desarrollar la plataforma de simulación fue Matlab. Éste es un programa de lenguaje de alto nivel y un entorno interactivo para el cálculo numérico, la visualización y la programación. Mediante Matlab, es posible analizar datos, desarrollar algoritmos y crear modelos mecánicos o aplicaciones. El lenguaje, las herramientas y las funciones matemáticas incorporadas permiten explorar diversos enfoques y llegar a una solución antes que con hojas de cálculo o lenguajes de programación tradicionales, como pueden ser C/C++ o Java™.

Otra característica importante de Matlab es que cuenta con una serie de herramientas adicionales, por ejemplo Guide y Simulink. Guide es un complemento en el cual se puede programar de manera gráfica tal como Visual C, pero con la capacidad de utilizar todas las bibliotecas de Matlab y Simulink.

3.1.2. Simulink-SimMechanics

Simulink nos permite realizar programas de forma visual utilizando bloques. También nos permite simular y modelar sistemas. Una de las principales ventajas es que Simulink tiene la capacidad de simular sistemas no lineales. Por lo anterior Simulink es uno de los principales bloques que conforman la plataforma de simulación.

Simulink cuenta extensas bibliotecas que nos permiten simular diferentes sistemas. Una de estas bibliotecas es SimMechanics. SimMechanics es una biblioteca que nos permite simular sistemas mecánicos. Con la ayuda de SimMechanics no se tiene que calcular el modelo dinámico de un mecanismo de forma analítica.

Cuando se habla de sistemas mecánicos se hace referencia a cuerpos y articulaciones en movimiento. SimMechanics puede simular la masa, la acción de la gravedad, posición, velocidad, aceleración de los elementos mecánicos también puede simular fuerzas y torques. En la Fig. 3-2 se muestran las diferentes bibliotecas de SimMechanics.

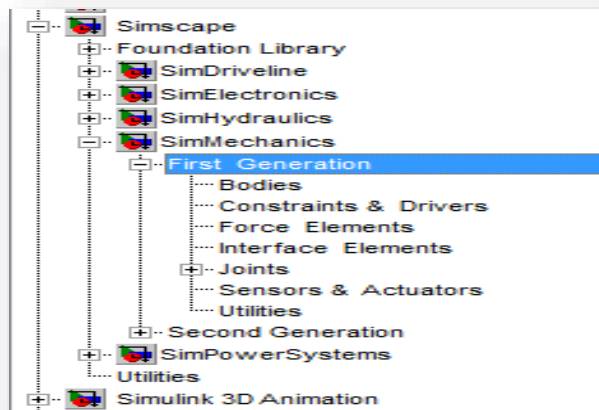


Fig. 3-2 Biblioteca de Simulink.

Los bloques en SimMechanics representan objetos, por ejemplo cuerpos de un sistema mecánico, actuadores, sensores, etc. A partir de los bloques se puede crear un programa que represente un sistema mecánico. Para entender mejor esta herramienta revisaremos los principales bloques de SimMechanics.

Los cuerpos (Bodies) son componentes elementales de un sistema mecánico dentro de SimMechanics. Los mismos que son definidos en SimMechanics como objetos en el espacio que tienen masa.

Existen dos tipos de cuerpos: uno es "Ground", el cual representa una referencia fija para las articulaciones y cuerpos en movimiento. El segundo es el elemento "Body" que representa a un cuerpo rígido (masa, momentos de inercia en cada uno de sus ejes, etc.) y su configuración en el espacio (posición y orientación tanto del centro de gravedad, denominado CG, como las partes de conexión mecánica del cuerpo, llamados CS en el programa).

Dentro de la biblioteca de SimMechanics existe un bloque denominado "Machine Environment" que se debe poner en todo sistema de cuerpos rígidos y que es el encargado de permitir la simulación mecánica del sistema.

Las restricciones (ver Fig. 3-3) son las encargadas de limitar los grados de libertad entre dos cuerpos. Mediante los bloques limitadores por ejemplo que un cuerpo solo se mueva paralelo a otro cuerpo, que algún cuerpo solo se pueda mover en un eje, restricciones de engranes, etc.

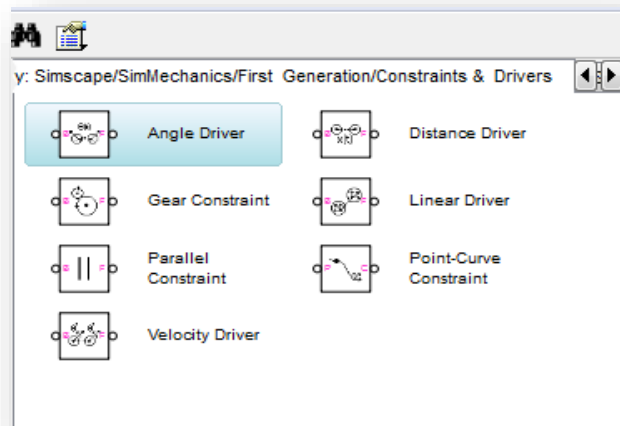


Fig. 3-3 Biblioteca de restricciones de Simmechanics.

Los bloques (Joints) representan los grados de libertad que un cuerpo puede tener. Algunos de estos bloques son restricción de revolución, prismática, cilíndrica, planar, etc. En la Fig. 3-4 se muestran todos los bloques "Joint" disponibles en SimMechanics.

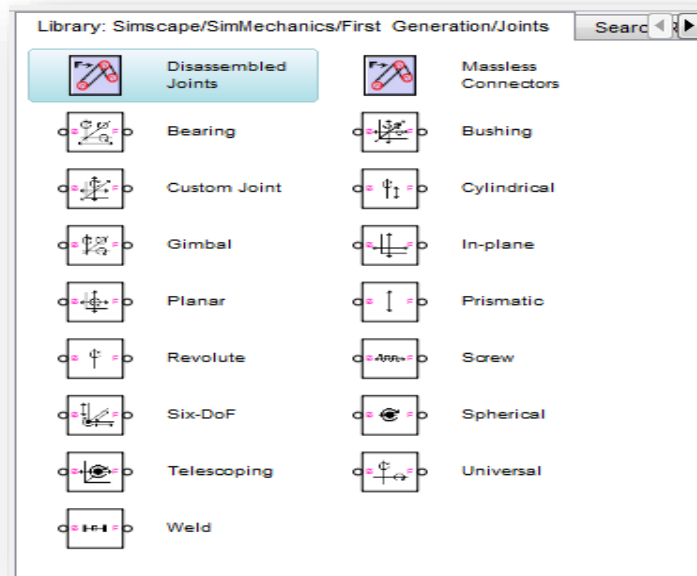


Fig. 3-4 Biblioteca de articulaciones de Simmechanics.

Los bloques de sensores (Fig. 3-5) son bloques que permiten monitorear el movimiento y la fuerza o par sobre un cuerpo o articulación. Los actuadores permiten suministrar movimientos a los cuerpos y articulaciones.

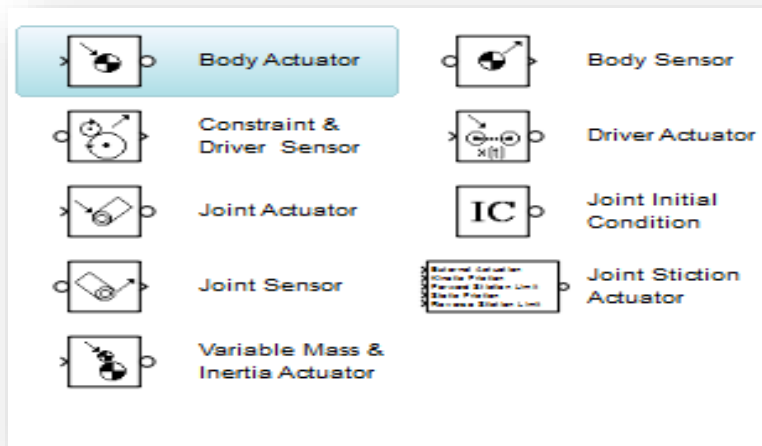


Fig. 3-5 Biblioteca de sensores de Simmechanics.

Con los bloques de Simmechanics, se pueden formar la mayoría de los sistemas mecánicos. Sin embargo una desventaja de SimMechanics es su limitada capacidad para generar diseños mecánicos complejos. Para resolver el problema se necesita un programa adicional CAD. Que nos permita realizar diseños mecánicos complejos por ejemplo Solidworks.

3.2. Desarrollo de la plataforma

3.2.1. Modelo en Simulink-SimMechanics de la arquitectura propuesta

Retomando lo que se vio en el capítulo II, con la ayuda de Solidworks se propuso una arquitectura con 21 grados de libertad utilizando geometrías sencillas (ver, Fig. 3-6). Para los eslabones que representan el fémur, la tibia, una parte del tronco y la cadera, se utilizó una forma de cilindro. Se diseñó de manera que pudiera conectarse con una articulación. Para las uniones de cada eslabón se utilizaron 3 articulaciones, cada articulación tiene un grado de libertad en diferente plano (Sagital, Frontal, Transversal).

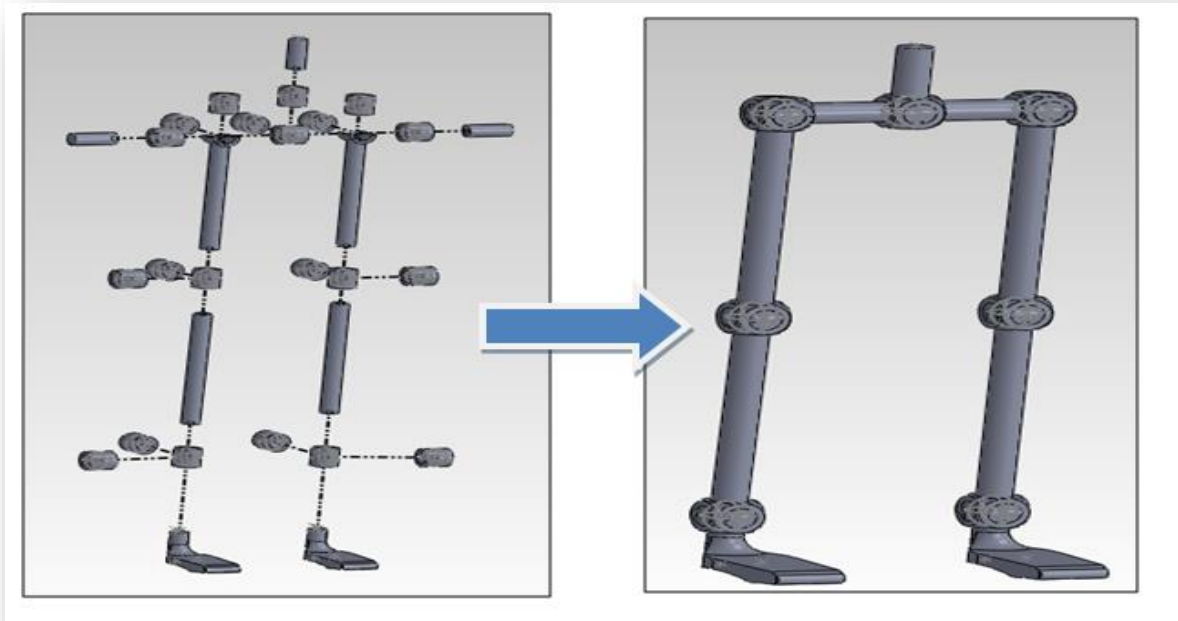


Fig. 3-6 Interfaz Solidworks y Matlab.

Utilizando el modelo de las piernas del ser humano se decide crear un programa a bloques utilizando SimMechanics. Sin embargo Solidworks no es el único programa CAD con el que SimMechanics puede trabajar, también puede trabajar con Pro/ENGINEER y Autodesk Inventor. Los programas antes mencionados son los únicos que pueden generar un archivo que Matlab pueda leer.

Tomando el ensamblaje de Solidworks de las extremidades inferiores del cuerpo humano se preparó para ser exportado a Matlab. Esto se hace instalando el módulo SimMechanics link en Solidworks, que permite guardar el ensamblaje, de tal manera que pueda ser leído por Matlab. Este archivo guarda la forma geométrica de las piezas y las restricciones del ensamblaje.

El archivo generado por Solidworks (.xml) ahora se puede leer desde Matlab. El archivo tiene que ser abierto por medio de un comando especial (`mech_import('nombre_archivo.xml')`).

El archivo que puede leer Matlab queda como se muestra en la Fig. 3-7 donde se generan bloques, para las restricciones y las dimensiones de los eslabones.

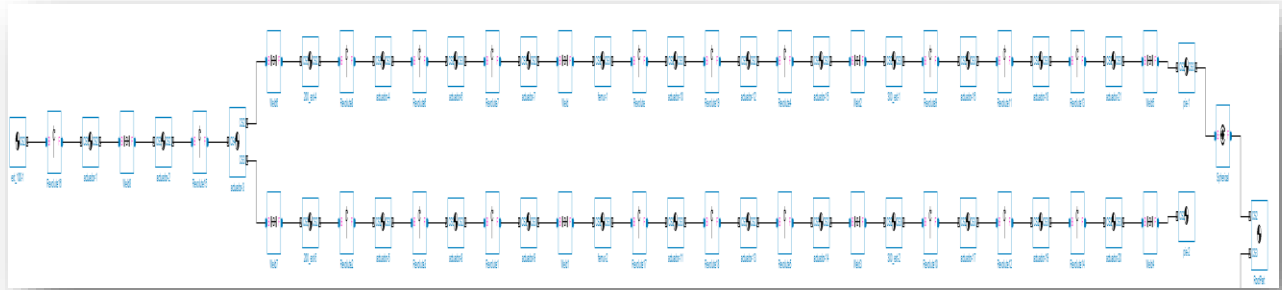


Fig. 3-7 Programa a bloques en Simulink de la arquitectura propuesta.

Una vez que el programa es abierto en Simulink solamente queda guardarlo con un formato que Simulink pueda leer y modificar (.mdl). Una vez que se guardó el programa con el formato (.mdl) la arquitectura queda lista para iniciar algunas simulaciones en el entorno virtual de Matlab Fig. 3-8.

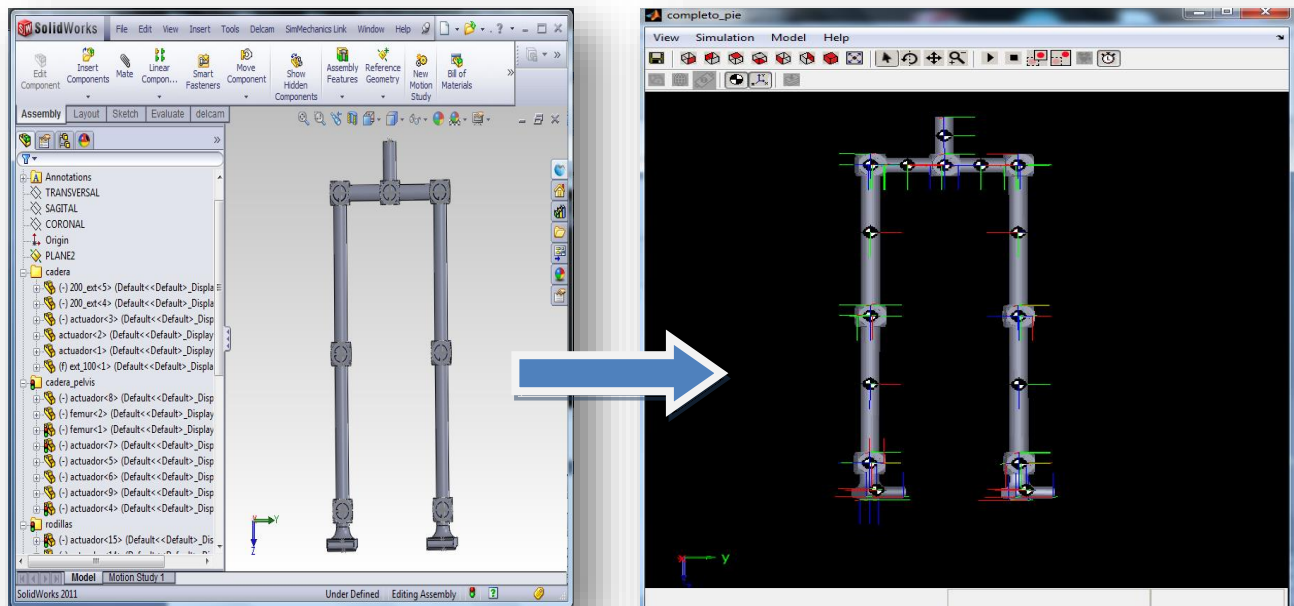


Fig. 3-8 Derecha, representación virtual del ensamblaje realizado en Solidworks dentro del entorno de Simulink. Izquierda, ensamblaje en Solidworks que representa las piernas de un ser humano.

El programa a bloques generado a partir de Solidworks y SimMechanics, es muy grande y con bloques sin organización. Esto quiere decir que los bloques que se encargan de restringir y representar las figuras geométricas de por ejemplo el pie, podrían estar revueltos con los bloques que pueden generar la configuración de la pelvis. Por lo tanto el programa a bloques se tiene que organizar de tal forma que sea fácil encontrar cada uno de los bloques.

El programa que se muestra en la Fig. 3-7 se modificó para organizarlo de una manera más comprensible. Se crearon subsistemas en los cuales se engloban varios bloques para identificar rápidamente un área en particular. Como se muestra en la Fig. 3-9 el programa principal se dividió en 7 sub sistemas, tronco, pelvis, cadera derecha e izquierda, rodilla derecha e izquierda, pie derecho e izquierdo.

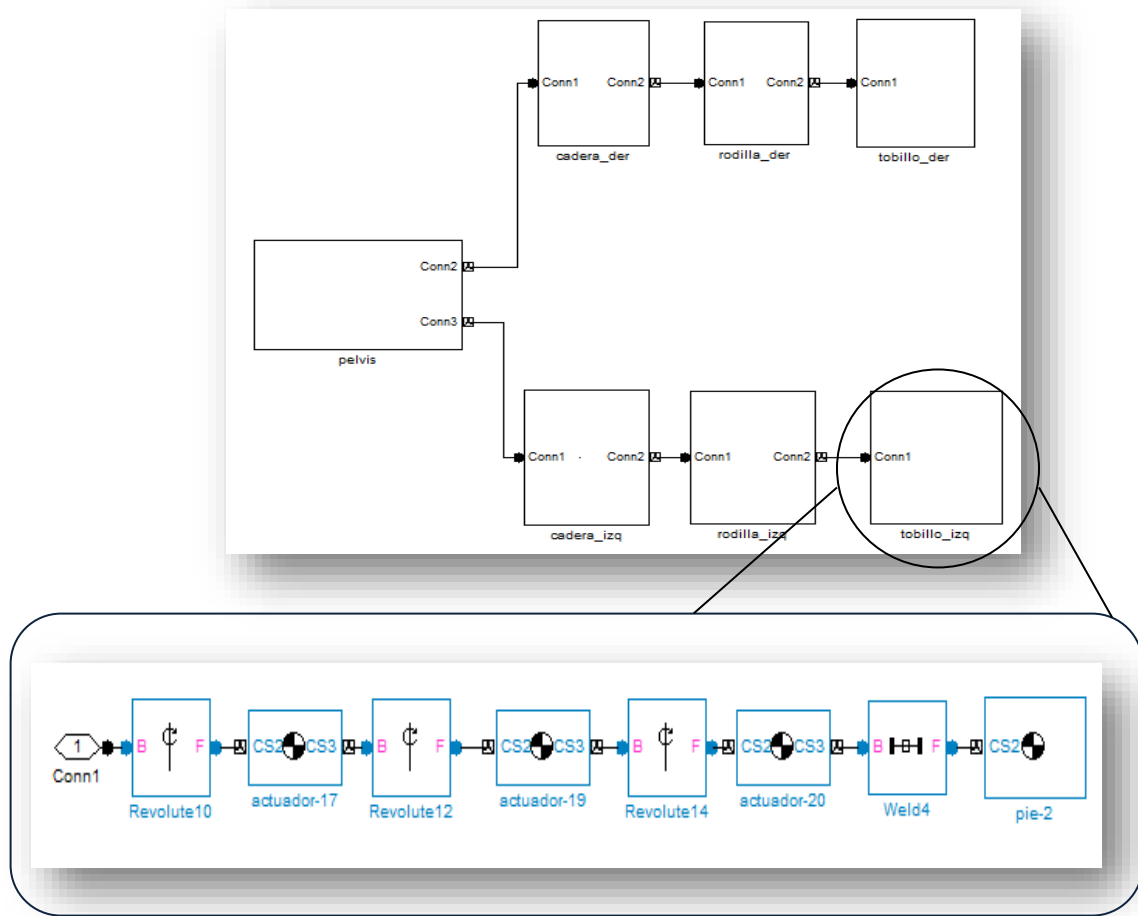


Fig. 3-9 Programa a bloques del tobillo izquierdo de la arquitectura propuesta.

3.2.2. Movimiento de las articulaciones

Cuando se exporta un ensamble de Solidworks a Simulink las articulaciones tienen libre movimiento. Esto quiere decir que las articulaciones se mueven libremente en el entorno virtual. Si

quisiéramos realizar una simulación con la arquitectura propuesta las articulaciones que tienen movimiento rotatorio se moverían libremente debido a la acción que ejerce la gravedad y el mecanismo caería.

Para agregar movimiento a la arquitectura propuesta y evitar que las articulaciones se muevan libremente se debe agregar un actuador a cada grado de libertad. Para realizar esta acción Matlab tiene un bloque que nos permite aplicar un movimiento a cada articulación. El movimiento de las articulaciones se puede realizar mediante el bloque (Joint Actuator) ver Fig. 3-10. El cual nos permite mover la articulación aplicando una posición, velocidad y aceleración o aplicando un torque directamente a la unión que queremos mover.

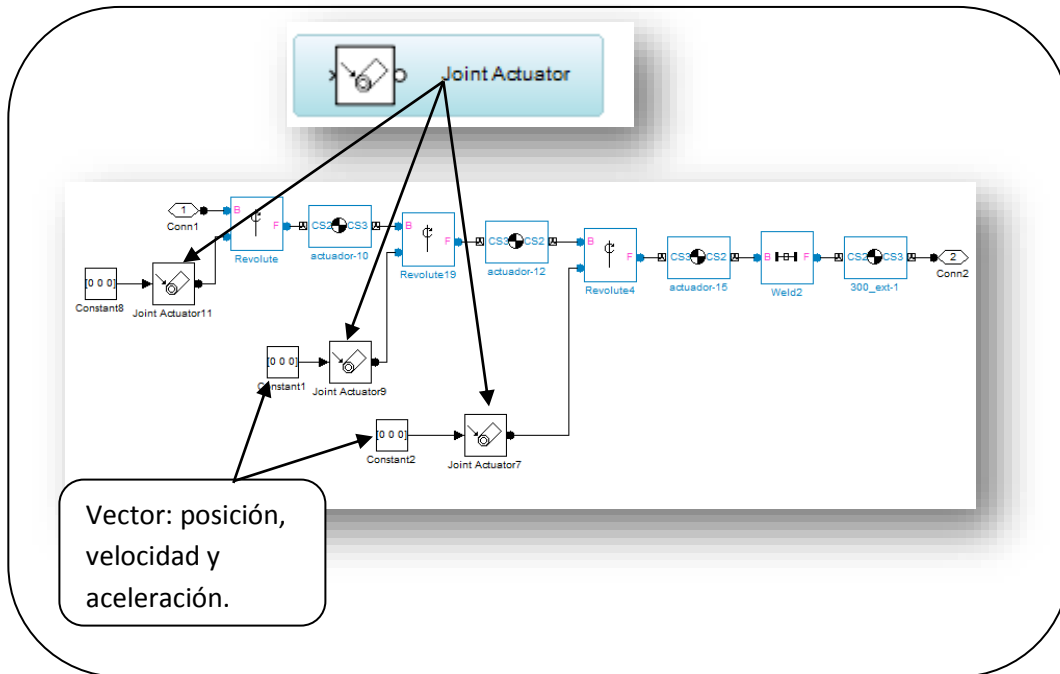


Fig. 3-10 Bloque sensor de posición, velocidad y aceleración utilizado en el programa a bloques de la rodilla derecha.

De esta misma forma se agregaron bloques de movimiento para los 21 grados de libertad de la arquitectura propuesta. Los datos para mover la arquitectura se obtuvieron del estudio del caminado humano en los tres planos corporales que se vio en el capítulo I.

Simulink tiene una serie de herramientas, que nos permite determinar la posición, velocidad y aceleración de cada eslabón durante la simulación. También se puede obtener torque y fuerza en el caso de utilizar ensambles. Una ventaja de esta herramienta es que con un solo bloque se pueden obtener hasta 5 datos a la vez (velocidad, posición, aceleración, torque y fuerza).

Estas herramientas pueden ser de gran utilidad ya que no tenemos que realizar un análisis cinemático o dinámico de un mecanismo para determinar dónde se encuentra cada eslabón o la fuerza que está ejerciendo en cada momento.

Los bloques de “Body Sensor” pueden ayuda a determinar la posición de los eslabones durante la simulación. Se agregaron 4 bloques sensor a cada cuerpo que representa los pies de la arquitectura propuesta (ver, Fig. 3-11). El bloque “Body Sensor” tiene como salida una matriz de valores (X, Y, Z, en el caso de obtener el valor de posición) por cada muestra de tiempo de simulación.

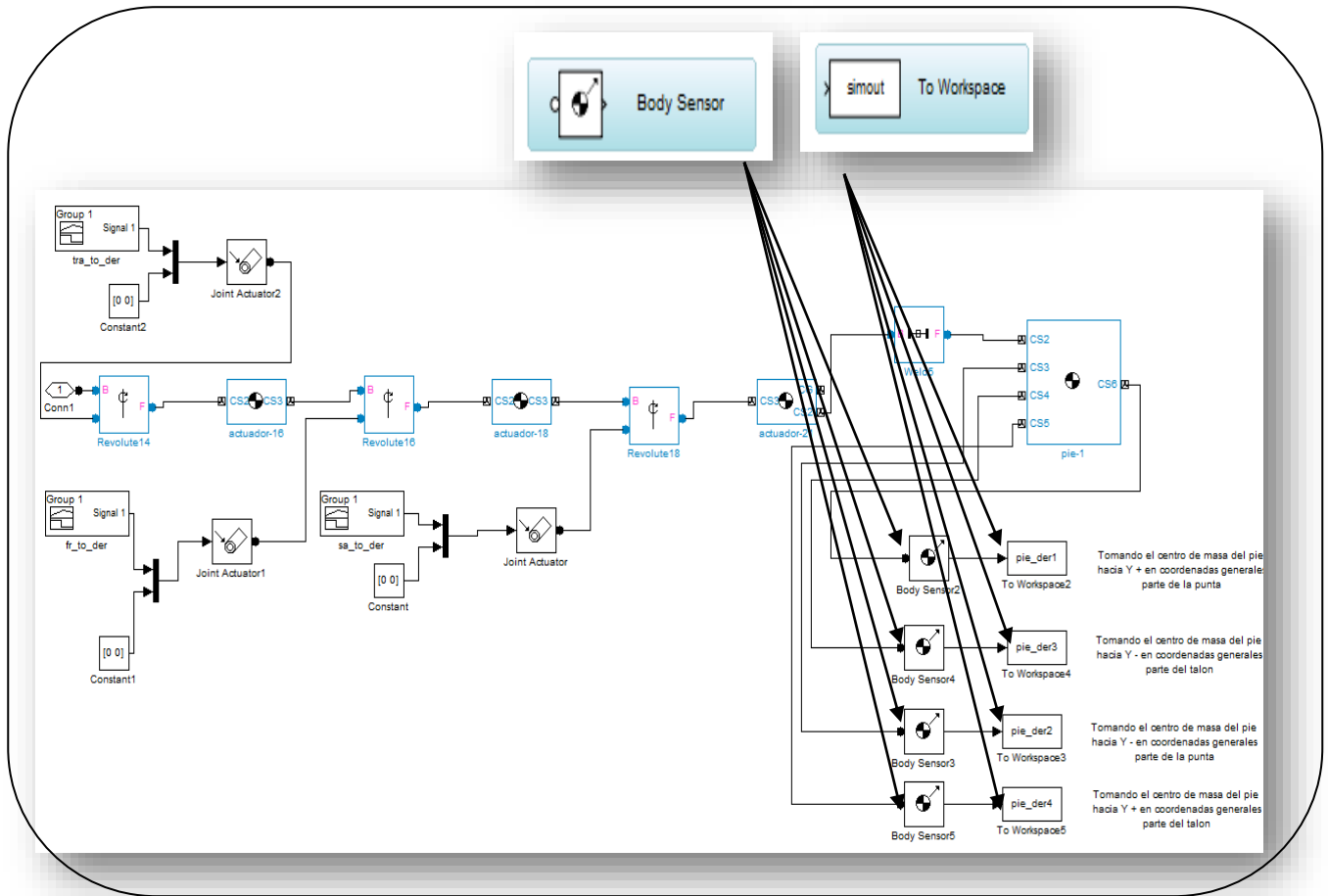


Fig. 3-11 Derecha, sensores de posición de 4 puntos en el área que representa el pie derecho.

Se agregaron bloques sensores en las plantas de los pies de la arquitectura propuesta. Como se muestra en la Fig. 3-11 se colocaron 4 sensores en los 4 puntos más alejados que conforman la geometría del pie.

Los datos sensados son enviados a Workspace en Matlab con la ayuda del bloque “To Workspace”. Como se muestra en la Fig. 3-11 se agregó un bloque “To Workspace” a cada bloque sensor.

Los datos obtenidos de cada sensor se guardan en un archivo (.mat) con la ayuda del bloque “To File” los cuales fueron agregados a cada sensor. Estos datos tienen que ser procesados por un programa externo que se encarga de obtener la matriz de datos para posteriormente graficarlos. Como se muestra en la Fig. 3-12 se pueden observar la posición del efector final durante dos ciclos del caminado humano. El eje vertical corresponde a la distancia que se mueve en el eje Z el pie izquierdo. El eje horizontal corresponde al porcentaje de simulación donde 100% corresponde a 2 ciclos del caminado humano

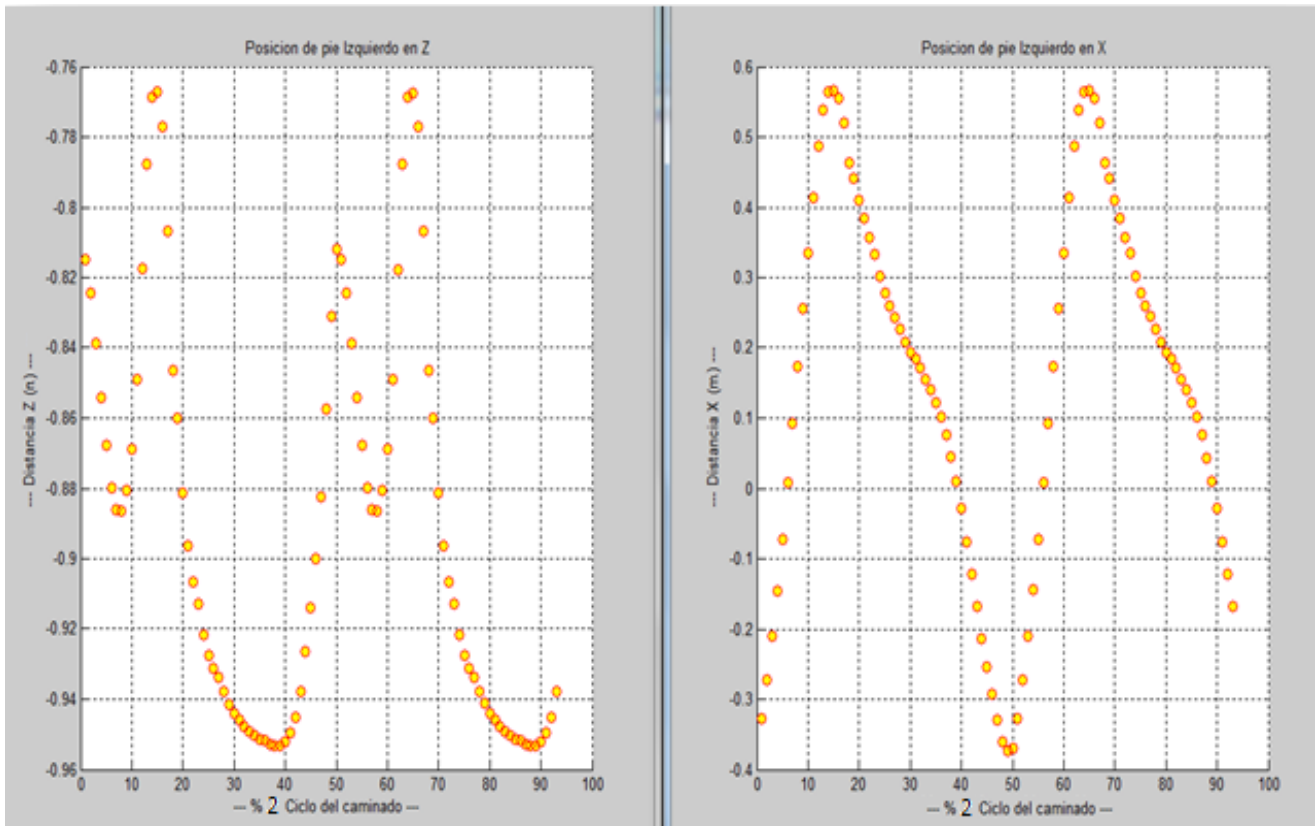


Fig. 3-12 Gráfica de posición del pie durante dos ciclos de caminado.

3.2.3. Datos para el movimiento de las articulaciones

Los datos que se obtuvieron del movimiento de las articulaciones en el capítulo I fueron imágenes recopiladas en varios trabajos. A pesar de que se realizó una búsqueda exhaustiva de alguna base de datos que contuviera el movimiento de las articulaciones, no fue posible encontrarla. Por lo tanto con la ayuda de las imágenes recopiladas en el estudio del caminado humano se realizó una base de datos para el movimiento de cada una de las articulaciones de la arquitectura propuesta.

Como se muestra en la Fig. 3-13 se obtuvieron las imágenes de las trayectorias que sigue cada articulación durante un ciclo del caminado humano. Con la ayuda de Solidworks se importaron las imágenes y después se trazaron puntos sobre la imagen para obtener una referencia de la gráfica.

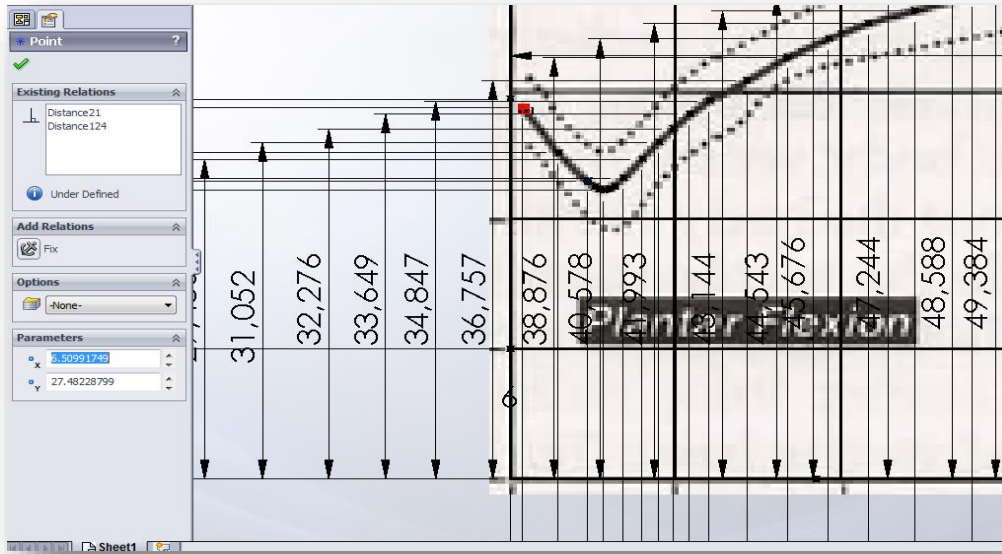


Fig. 3-13 Imagen importada en Solidworks que fue utilizada para obtener una referencia.

En la Fig. 3-14 a la izquierda, se muestra una imagen de los grados que se mueve la cadera durante un ciclo del caminado. A la derecha se muestra la misma imagen pero importada dentro del entorno de Solidworks. Cada punto puesto sobre la imagen contiene una coordenada en x, y con respecto a un marco de referencia. Estas coordenadas son incorporadas una a una en una base de datos que fue creada para guardar la información. Este mismo procedimiento se realizó para todas las posiciones de las articulaciones.

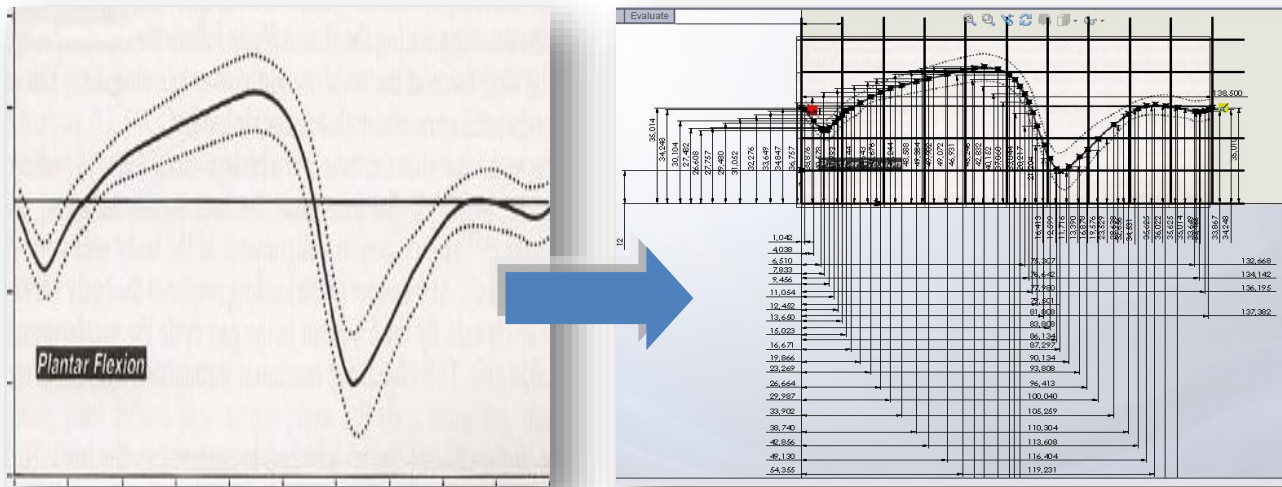


Fig. 3-14 Imagen importada en Solidworks, de movimiento de cadera derecha.

Los datos que se obtuvieron de la imagen fueron alrededor de 50 puntos. Los puntos obtenidos se utilizaron para desarrollar un programa en Matlab el cual permitiera reproducir la imagen.

En el programa se creó una base de datos para una articulación donde se utilizan los puntos con coordenadas, para realizar una aproximación con un spline cúbico. La función en Matlab para el spline cúbico es (*spline(porc,gra,i)*) y viene incorporada dentro de la biblioteca de Matlab. Utiliza tres parámetros los cuales son datos en y, datos en x y el número de puntos que se quiera obtener. Como se muestra en la Fig. 3-15 se realizó un programa (el programa se encuentra en el CD de apoyo) con una base de datos diferente para todos los movimientos de las articulaciones.

Como se muestra en la Fig. 3-15 primero se genera una base de datos para cada función. Después se agrega una corrección a los puntos generados, debido a que la referencia de las coordenadas de los puntos es la esquina de la imagen. Por lo tanto se tienen que modificar cada una de las coordenadas de cada punto. Con la ayuda de la función *spline* se generan 100 datos que corresponden a un ciclo del caminado humano (0%-100%). Por último se duplican los datos para que la arquitectura pueda reproducir dos ciclos del caminado.

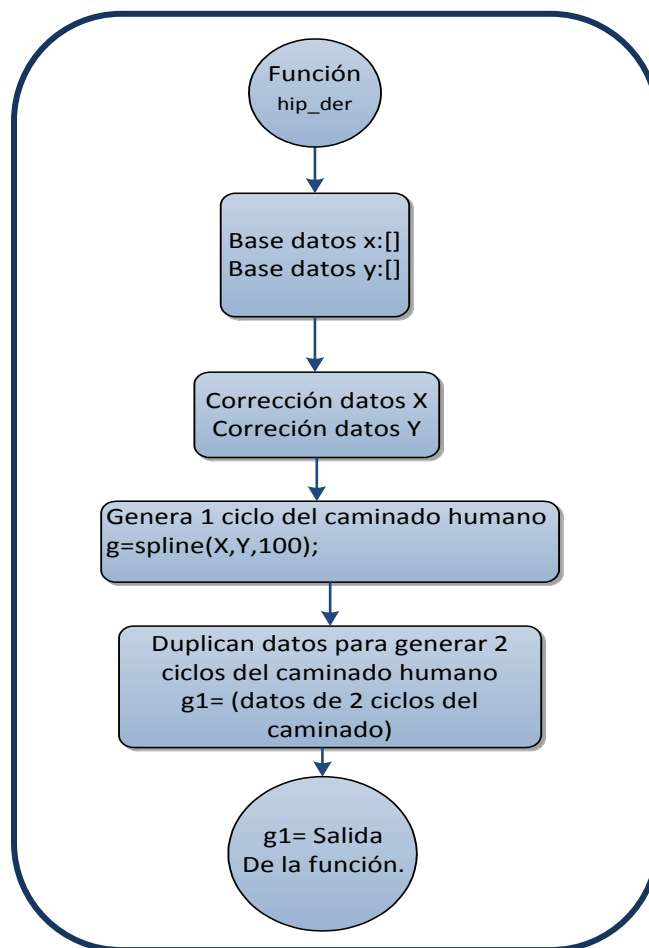


Fig. 3-15 Función de cadera derecha para obtener gráfica de movimiento de la articulación.

Se desarrolló un programa similar para cada articulación. Para las articulaciones del pie opuesto se utilizó un programa similar al de la Fig. 3-15. La diferencia fue que el programa empieza desfasado la mitad del ciclo con respecto al de la Fig. 3-15. Esto fue debido a que a pesar de que las piernas realizan el mismo movimiento al caminar siempre va a haber una pierna que este más adelantada que la otra. Por ejemplo el movimiento del pie derecho es el mismo movimiento que el del pie izquierdo pero desfasados uno de otro medio ciclo del caminado humano.

Una vez que los programas están completos se pueden reproducir los datos obtenidos. Como se puede observar en la Fig. 3-16 se reproducen el movimiento para la cadera derecha en el plano sagital.

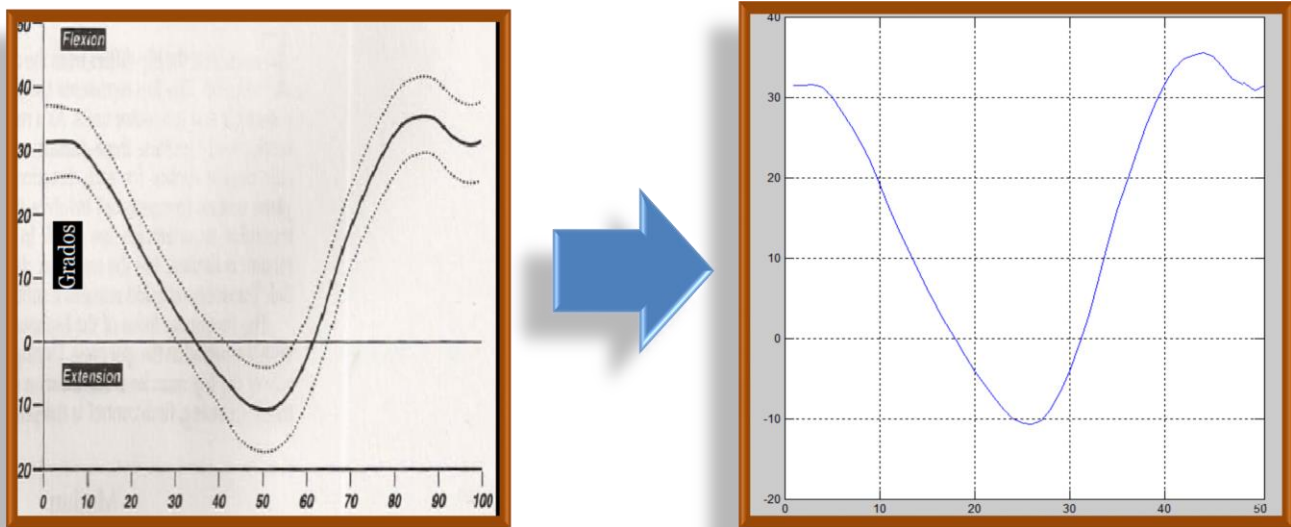


Fig. 3-16 Izquierda, imagen del movimiento de la articulación de la cadera derecha en el plano sagital, obtenida durante el estudio del caminado. Derecha, imagen digitalizada en Matlab del movimiento de la cadera derecha.

Los programas que reproducen los movimientos de todas las articulaciones se encuentran dentro del área de trabajo (Workspace) en Matlab. Como se había mencionado antes la plataforma de simulación fue desarrollada con la ayuda de la herramienta de Simulink. Aunque es una herramienta de Matlab, importar datos o exportar datos desde Workspace hacia Simulink o viceversa, requiere de programación extra.

3.2.4. Exportar datos a Simulink

Para iniciar con los movimientos de las articulaciones se necesita guardar datos dentro del programa a bloques desarrollado en Simulink. Si quisiéramos intentar correr los datos en línea o tratar de insertar datos cuando se está ejecutando un programa a bloques no es posible debido a que Simulink no está diseñado para trabajar de esa manera. Por esta razón los datos tienen que ser guardados de alguna forma antes de iniciar con alguna simulación.

Un bloque que sirvió para guardar los datos de las articulaciones provenientes de "Workspace" fue (*signal builer block*, ver Fig. 3-17). El bloque (*Signar Builer*) muestra y permite crear o editar grupos

intercambiables de fuentes de señal y cambiar rápidamente los grupos dentro y fuera de un modelo. Grupos de señal se pueden facilitar en gran medida la prueba de un modelo, sobre todo cuando se utilizan con conjunción con Simulink.

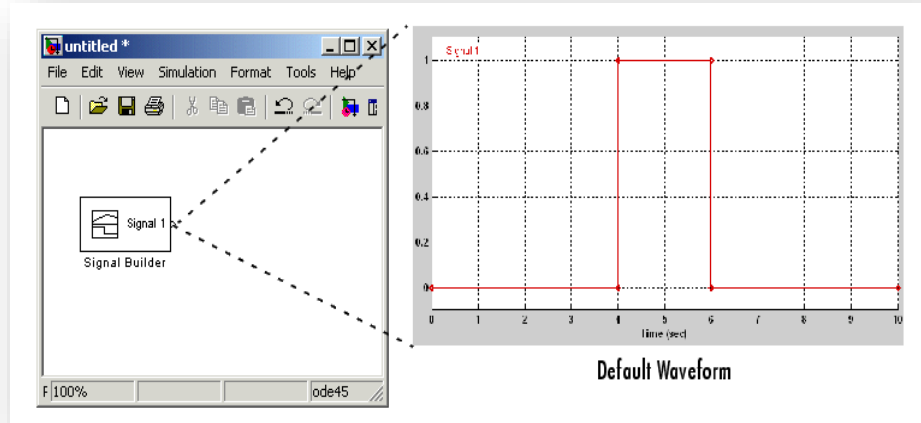


Fig. 3-17 Bloque *Signal Builder* utilizado para obtener movimientos continuos en la simulación en Matlab.

El bloque *Signal Builder* es una forma de manejar los datos conforme la simulación avanza, como se muestra en la Fig. 3-18. La señal de la izquierda representan los datos obtenidos para dos ciclos de caminado humano para la articulación del tobillo derecho en el plano frontal. Los datos son utilizados para ir modificando la posición de la articulación del tobillo.

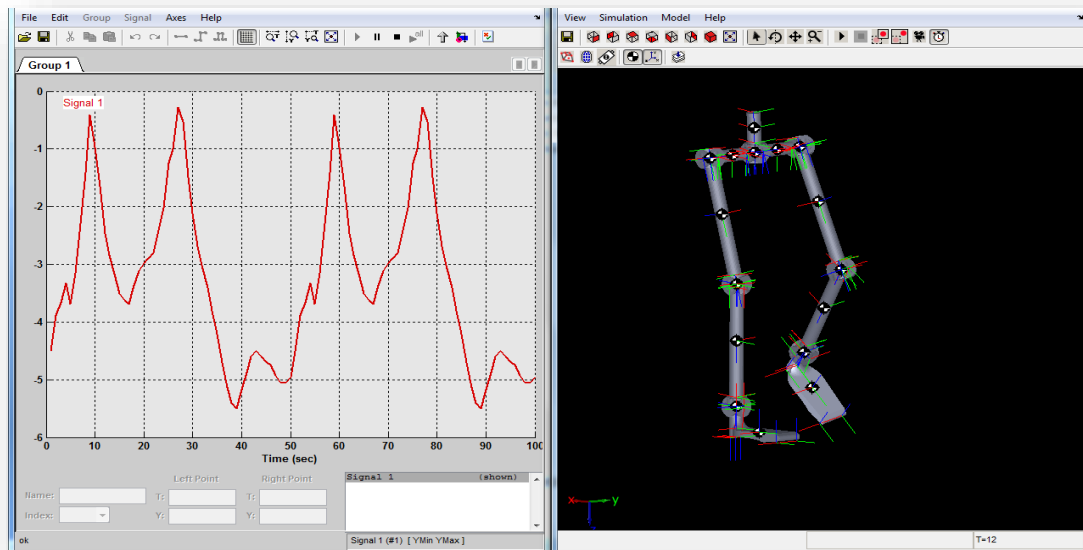


Fig. 3-18 Señal importada desde Matlab dentro de Simulink.

La manera en la que los datos de las gráficas que se muestra en la Fig. 3-18, se insertan en Simulink es manual. Cada uno de los puntos tiene que ser introducido al bloque de Simulink por medio de dos vectores los cuales contiene las variables (x, y) que corresponde a los puntos de las gráficas.

Pero si se quisiera realizar estudios de arquitecturas diferentes o configuraciones diferentes para el mismo robot se tendría que introducir los datos uno por uno para cada simulación. Por lo tanto se optó por crear un nuevo programa para cada articulación que permita escribir la información sobre el bloque de *signal builder* sin tener que escribir los datos uno a uno de manera manual.

Matlab tiene una extensa *Application Program Interface* (API) para la construcción y modificación de los modelos de Simulink de código Matlab. Esto puede ser desde la línea de comandos de Matlab (Workspace), desde dentro de una función o un script, o desde cualquier lugar donde un código (.m) puede ser ejecutado.

Las funciones que nos permiten esto son `get_param` y `set_param`. Permiten a todas las propiedades de un modelo o bloque que ser inspeccionados y cambiados respectivamente a partir de código Matlab. En la Fig. 3-19 se muestra una diagrama de flujo del programa que modifica el bloque *signal builder* en Simulink. El programa completo se muestra en el CD de apoyo.

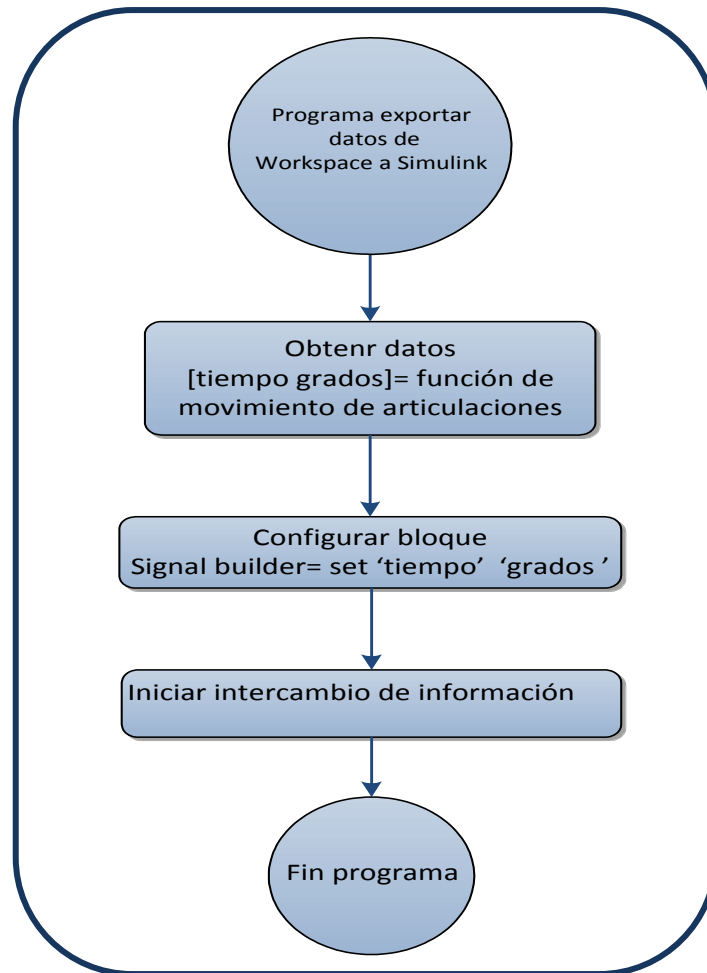


Fig. 3-19 Programa que llama las funciones que contienen los datos de posición de las articulaciones y los exporta al programa a bloques en Matlab.

3.2.5. Piso virtual

Durante el caminado hay una serie de pérdida y recuperación del balance. Para prevenir la falla en la recuperación del balance se mueve cualquiera de las extremidades a una nueva locación. Lo cual previene que perdamos el balance.

Saber la posición del centro de masa durante el caminado humano es importante para el estudio de la arquitectura propuesta. Con la posición del centro de masa podríamos determinar si el robot es capaz de caminar. También con las trayectorias del centro de masa se pueden comparar diferentes tipos de caminados.

Para obtener las trayectorias del centro de masa desde la plataforma de simulación el centro de masa tiene que estar en movimiento. Como se mencionó antes, en una primera instancia al importar el modelo CAD hacia Simulink, la estructura está fija en un punto en el espacio virtual. El punto fijo para la arquitectura propuesta es el centro de masa. Por lo tanto para obtener trayectorias del centro de masa se necesitó separar el robot de su punto fijo.

Al quitar el punto de apoyo fijo, el robot caía sin fin en el espacio virtual debido a que la plataforma cuenta con la simulación de gravedad.

Una solución para evitar que el robot cayera por el espacio virtual fue simular una superficie de apoyo justo bajo el robot. Una propuesta para simular este piso fue restringir algunos grados de libertad al efector final (pie) de tal forma que éste pudiera apoyarse en un punto. Una opción puede ser quitar el movimiento en el eje Z en donde se tiene el efecto de la gravedad. También se podrían eliminar varios grados de libertad al mismo tiempo en diferentes ejes (X, Y, Z) fijando el pie. Sin embargo si quitáramos estos grados de libertad el pie quedaría estático, sería lo mismo que dejar al pie con referencia fija.

Otra alternativa para resolver el problema sin tener que fijar algún eslabón fue agregar una simulación de un piso virtual en la plataforma desarrollada. A pesar de que la plataforma de simulación cuenta con un entorno dinámico, la simulación de un piso no está prevista en ésta.

Como la simulación de un piso virtual no es usual en Simulink, hay unos cuantos trabajos en la literatura donde se contempla un piso virtual en Simulink. Como se muestra en el trabajo realizado por Jorge Aparicio (Aparicio, 2013) se propone una solución a este problema realizando una aproximación mecánica donde utilizan un sistema masa resorte amortiguador como el que se muestra en la Fig. 3-20.

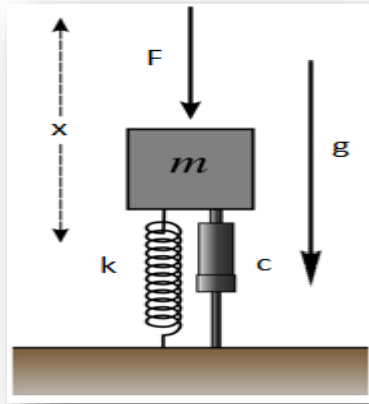


Fig. 3-20 Sistema masa-resorte-amortiguador.

$$c\dot{x} + kx = m\ddot{x} + F \quad \text{Ec.[3.1]}$$

En la ecuación 16 se expresa la suma de las fuerzas en el sistema de la Fig. 3-20. En esta ecuación x es el desplazamiento de la masa, mientras que \dot{x} y \ddot{x} (primera y segunda derivada de x con respecto al tiempo) son la velocidad y la aceleración respectivamente. F es la fuerza externa aplicada al sistema mientras que (c , k y m) son los parámetros de viscosidad del amortiguador, rigidez del resorte y masa respectivamente. Simular la ecuación (16) en SimMechanics implica realizar una primera derivada numérica con el inconveniente de volver muy lenta la simulación cuando se trabaja con SimMechanics.

Para resolver el problema sin utilizar bloques de derivada, se propone utilizar una corrección únicamente proporcional. Lo que quiere decir que solamente se utiliza la parte que representa el resorte.

La propuesta para resolver el problema es aplicar una fuerza en la planta del pie cuando este pase el límite de una referencia (piso virtual). Si el pie pasa la referencia del piso virtual por ejemplo un piso virtual, ver Fig. 3-21 ($Z=0$, en el eje Z se representa la altura del robot), se aplicaría una fuerza para que el robot pueda estar sobre el piso. Si la planta del pie pasa la referencia del piso virtual con un desplazamiento corto se aplicaría una fuerza pequeña. Si el desplazamiento con respecto a la referencia es grande se aplica una fuerza mayor. Para evitar tener el efecto de resorte una vez que el pie está sobre el piso virtual, un bloque impide que la fuerza se siga aplicando en la planta del pie.

Para simular el piso virtual se toma como referencia un punto, que en el caso de la simulación de la arquitectura sería $Z=0$. Para evitar que el robot caiga indefinidamente por el espacio virtual al pie no se le permite pasar de esta referencia.

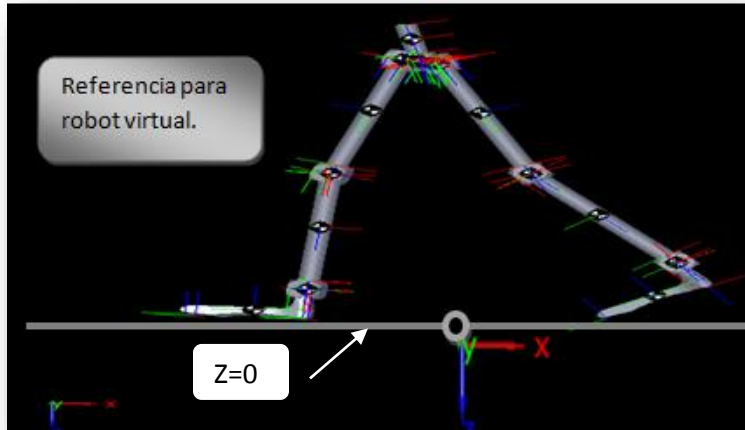


Fig. 3-21 Simulación de robot virtual con referencia de un piso.

Sin embargo para determinar si el pie pasó un límite se necesita saber la posición del talón y la punta del pie. Por lo anterior se utilizaron los sensores que se habían agregado a los pies del robot para obtener las gráficas de posición (ver, Fig. 3-22). Los sensores proporcionan la posición (X, Y, Z) de cualquier objeto dentro del espacio virtual.

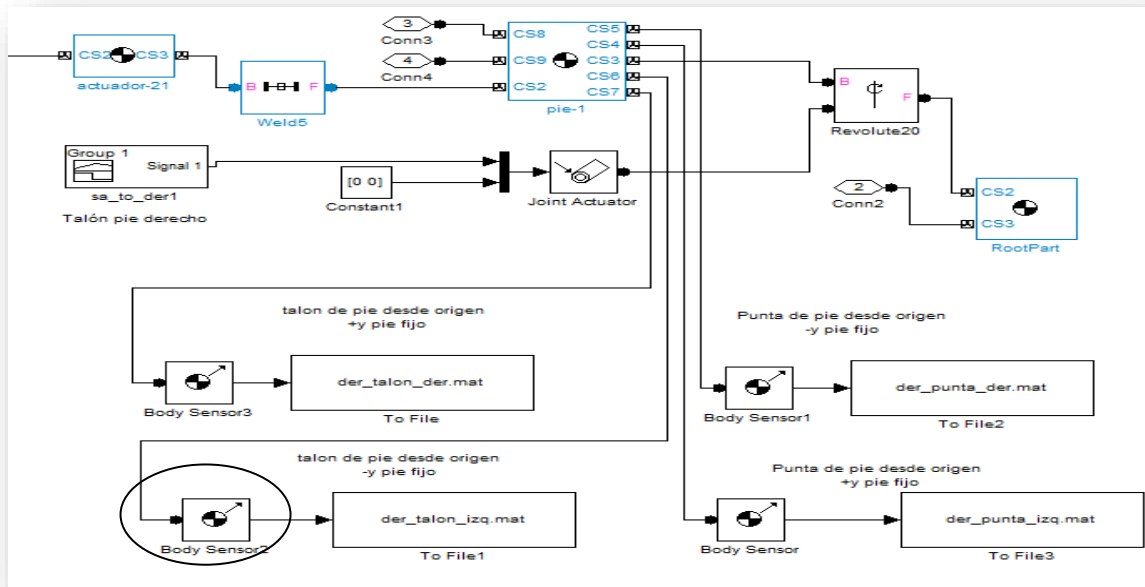


Fig. 3-22 Sensor (body sensor) recopila información de simulación y guarda los datos en un archivo .mat.

Una vez que se tiene la posición del efector final, esta posición es comparada con una referencia, en este caso es Z igual a cero. Como se muestra en la Fig. 3-23 se configura la referencia (piso) en cero. Esta comparación es retroalimentada a unos bloques actuadores (Fig. 3-24) que se agregaron a la simulación en la planta del pie. Los actuadores se encargan de proporcionar una fuerza en la planta del pie.

En la Fig. 3-23 se muestra el sub bloque de control de posición en el eje z.

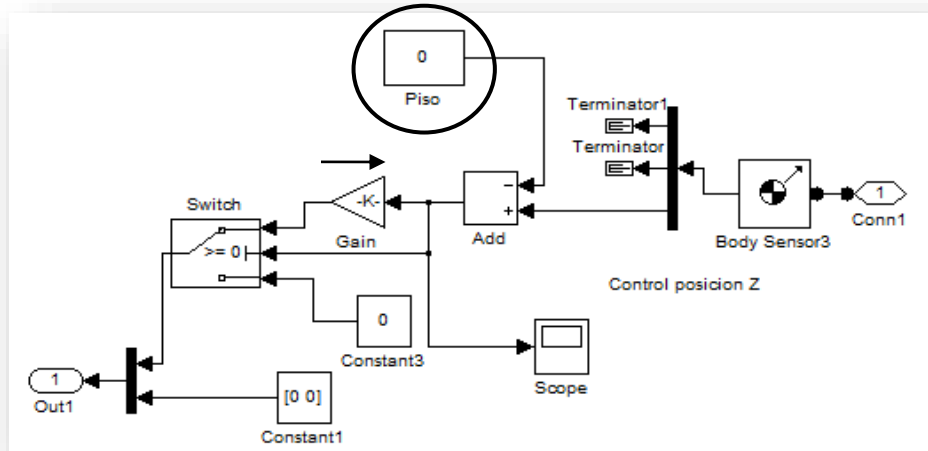


Fig. 3-23 Sub bloque de control de posición en el eje Z.

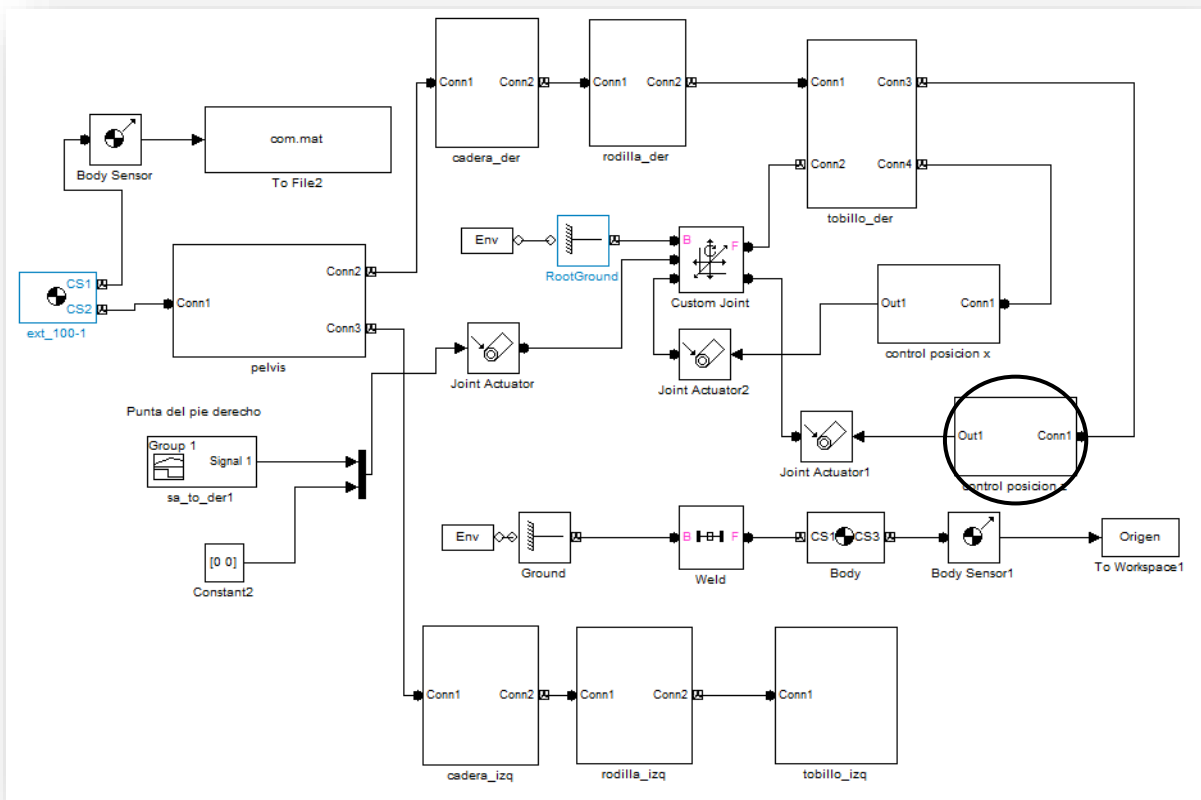


Fig. 3-24 Actuator para bloque que aplica corrección de posición.

Cuando el actuador aplica una fuerza mueve el pie en sentido positivo o negativo, elevando o bajando el robot virtual si tomáramos como referencia el piso ($Z=0$). Si la fuerza aplicada es menor que la fuerza que ejerza el robot debido a la gravedad puede “caer” en el espacio virtual.

En el programa a bloques que se muestra en la Fig. 3-23 se utiliza un bloque *switch*. Este bloque detecta el cruce por el valor de cero ($Z=0$, piso) lo cual nos permite detectar si la arquitectura se encuentra sobre el piso virtual. La adición del bloque nos permite escoger, entre aplicar una corrección (aplicar fuerza) o dejar de hacerlo.

Cuando el bloque detecta que el pie está por encima del piso, la corrección se desactiva y se aplica un valor de cero (Fuerza=0) en el actuador. Si no existe una fuerza aplicada en el actuador el robot empieza a deslizarse en sentido positivo en el eje Z o empieza a “caer” dentro del espacio virtual. Cuando el robot virtual pasa la referencia del piso se vuelve aplicar una corrección y el robot “sube”. De esta manera se activa o desactiva una fuerza manteniendo el robot en una referencia.

En el espacio virtual no solo se mueve en el eje Z sino que también se puede mover en los demás ejes. Para evitar que el robot virtual se deslice en el eje X también se tuvo que agregar una corrección. Como se muestra en la Fig. 3-25 se agregó un sub bloque para la corrección en el eje X.

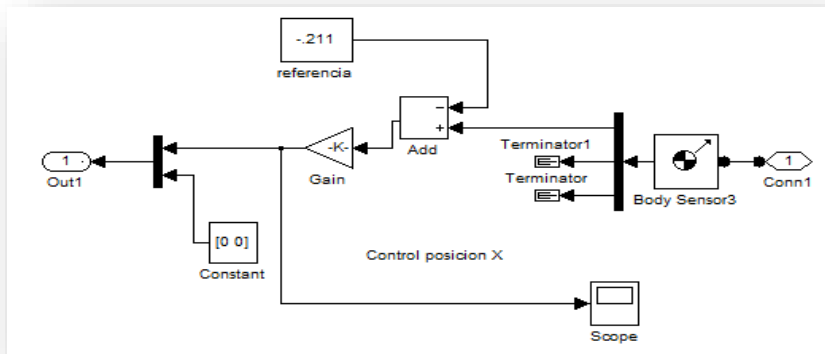


Fig. 3-25 Bloque que obtiene valor de la posición del pie y aplica una corrección cuando el pie pasa un límite de posición con respecto al piso.

Utilizando el mismo método para el control del eje Z se aplicó un control para el eje X. tomando como referencia $X=-0.211$ como se muestra en la Fig. 3-25. Este valor fue utilizado debido a que la simulación empieza con el efector final derecho del robot virtual dando el primer paso como se muestra en la Fig. 3-26.

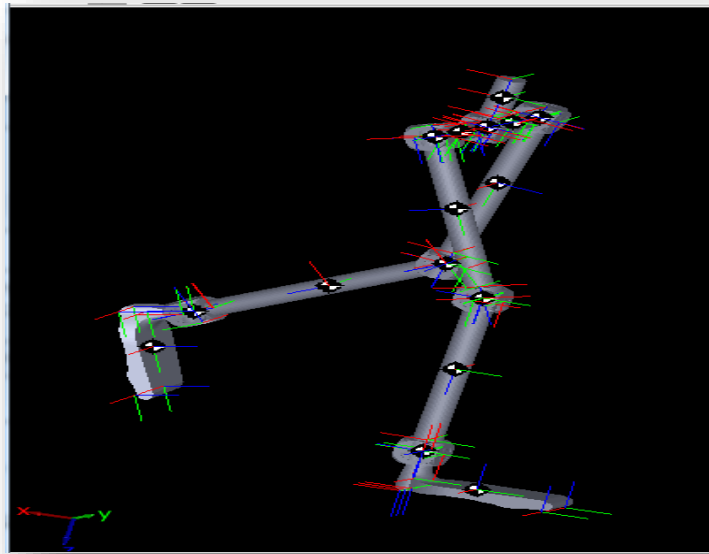


Fig. 3-26 Simulación de arquitectura de 21 grados de libertad.

Se utilizó un actuador para el eje X el cual tiene el nombre *joint Actuador 2* como se muestra en la Fig. 3-27. Este actuador se encarga de corregir cualquier deslizamiento en el eje X.

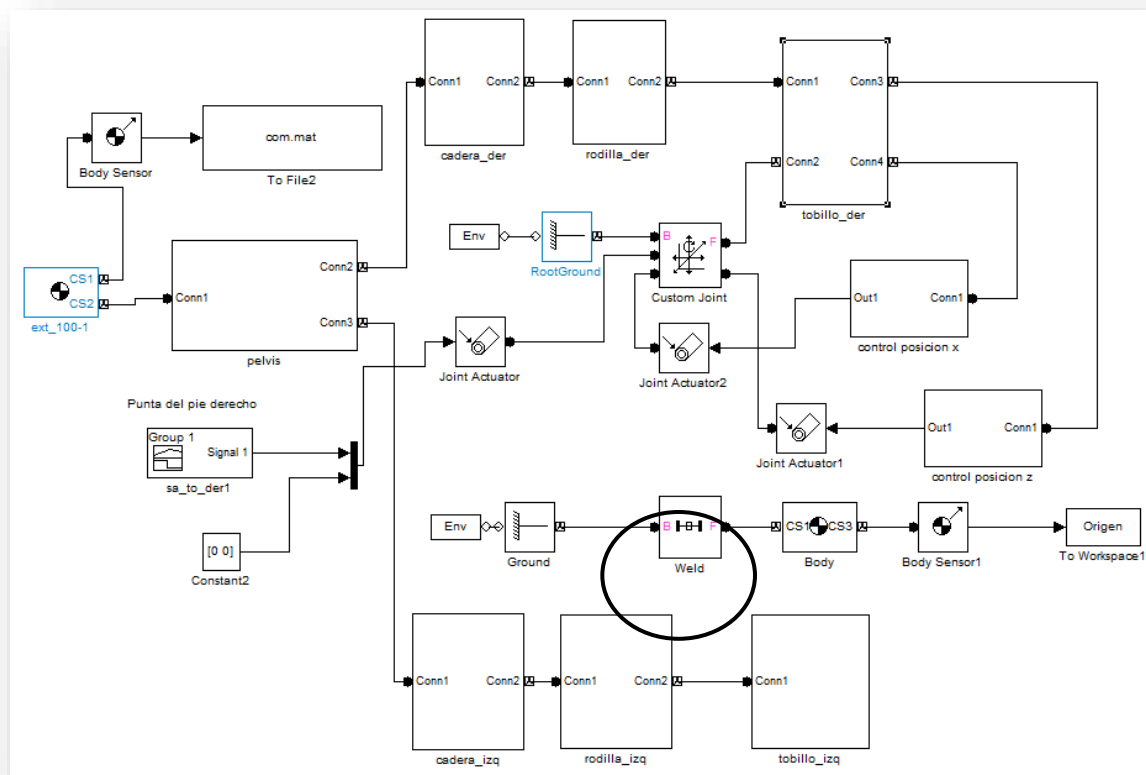


Fig. 3-27 Bloques agregados para el control de posición de piso y referencia de posición de x.

Con la adición de un control de posición para los efectores finales se pudo simular el piso virtual. Se podría mejorar si se agregara también un control en el eje Y para evitar el deslizamiento. Por el momento solamente se restringió al robot virtual para que no tuviera movimiento en este eje. Al evitar utilizar derivadas en Simulink se pudo acortar el tiempo de simulación.

3.3. Validación de plataforma de simulación de robots seriales

Una vez completada la plataforma de simulación es necesario validarla. Para ello se acude a la simulación dinámica de mecanismos conocidos. En particular se utilizó el estudio hecho en el capítulo II en donde se analizaron dos ejemplos de robots seriales de dos grados de libertad cada uno.

Se tomó como referencia el ejemplo 2 del capítulo II en donde se estudió un robot serial de 2 grados de libertad (R-P) como el que se muestra en la Fig. 3-28, en donde se encontraron las ecuaciones dinámicas (Ec.[2.10]) y (Ec.[2.11]) que rigen el mecanismo. Para la construcción del modelo CAD se agregaron dimensión a los eslabones y también se agregó masa.

$$\tau_1 = \ddot{\theta}_1 * (L_1^2 * m_1 + m_2 * d_2^2) + 2 * d_2 * \dot{d}_2 * m_2 * \dot{\theta}_1 \quad \text{Ec.[2.10]}$$

$$f_1 = m_2 * (\ddot{d}_2 - d_2 * \dot{\theta}_1^2) \quad \text{Ec.[2.11]}$$

Se seleccionó el centro de masa en una posición conocida para facilitar la simulación. Todo el modelo CAD del mecanismo serial se realizó en Solidworks. Los valores usados para el ensamble en Solidworks fueron para la masa₁ 2.0 kg, el valor de la masa₂ 0.5 kg, longitud L₁ 0.020 m y finalmente la distancia d₂ 0.045m.

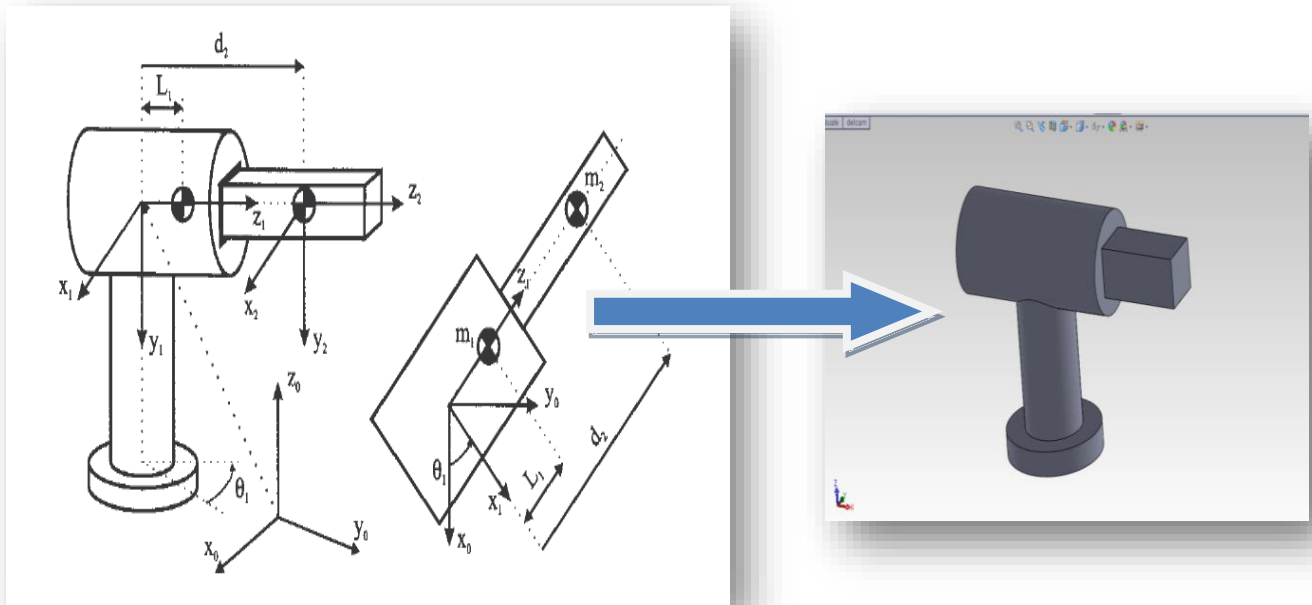


Fig. 3-28 Modelo CAD de robot serial (RP).

En el ensamble del modelo CAD solo se agregaron dos articulaciones, una de revolución para la base y una relación de posición prismática para el eslabón en forma de barra. Una vez que se completó el ensamble en Solidworks se exportó a Matlab como programa a bloques como se muestra en la Fig. 3-29.

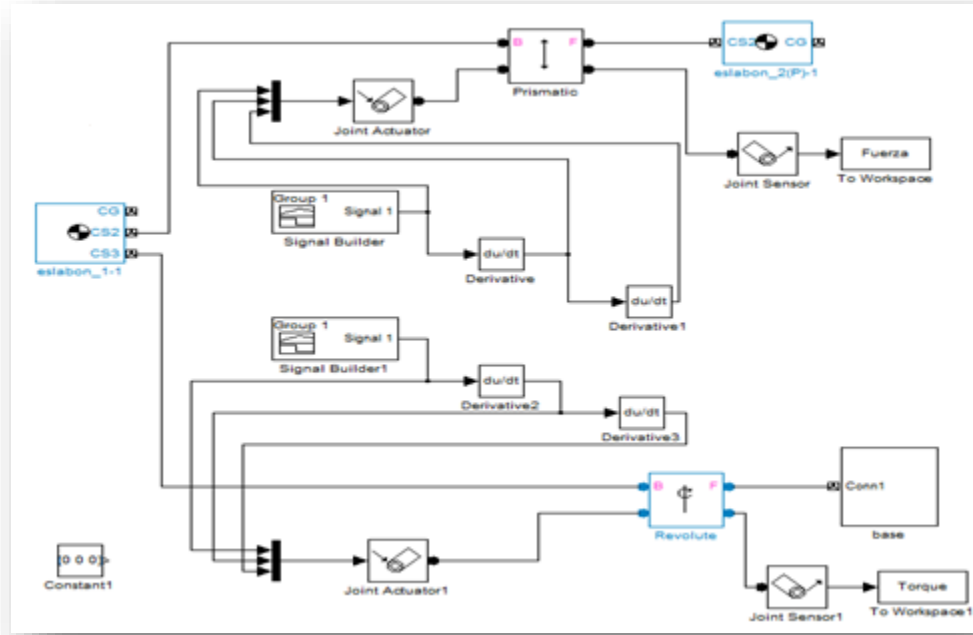


Fig. 3-29 Programa a bloques del robot serial (RP).

Una vez que el programa a bloques es completado, se agregan bloques para el movimiento del mecanismo. Una ventaja de utilizar SimMechanics es que podemos agregar movimiento a los mecanismos de dos maneras. La primera es agregando torsión o fuerza directamente a las articulaciones lo cual nos permite saber la posición, velocidad y aceleración de la articulación que se mueve. La segunda opción es que podemos mover la articulación agregando posición, velocidad y aceleración. Esta opción nos permite saber cuánta fuerza o torsión necesita la articulación para poder moverse.

Muchas veces cuando se quiere mover un mecanismo o un robot se tienen posiciones de las articulaciones para que este realice su tarea. Lo mismo pasa para el estudio que se desarrolló de la arquitectura propuesta, en donde se tiene las posiciones de las articulaciones para reproducir el caminado humano.

Por lo anterior se realizaron pruebas moviendo las articulaciones de un mecanismo, proporcionando como entrada posiciones que debe cumplir en un determinado tiempo.

Para lograr el movimiento de las articulaciones en la simulación se agregaron bloques *Signal Builder*. Este bloque de Matlab nos permite agregar valores conforme la simulación avanza. En la Fig. 3-30 se muestra la interfaz de *Signal Builder Block* donde se muestran los valores utilizados para la simulación.

Para la simulación se aplicó movimiento en las dos articulaciones. Para la articulación de revolución se agregó un movimiento $\Theta_1 = 0^\circ - 45^\circ$. Para la articulación prismática se aplicó un movimiento de $d_2 = 0$ a 150 mm en un tiempo de simulación de 100 segundos para cada una.

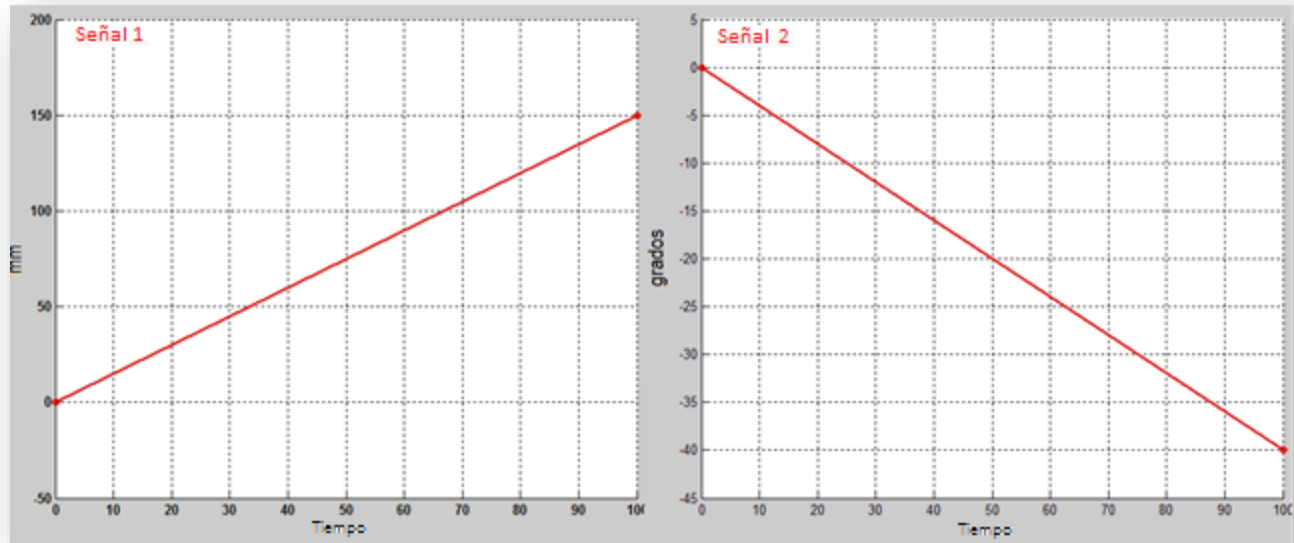


Fig. 3-30 Movimiento de articulaciones prismática a la izquierda y de revolución a la derecha.

También se agrega velocidad y aceleración a las articulaciones dentro del programa a bloques, ya que estos valores son necesarios para calcular la torsión y la fuerza (ver, Ec.[2.10] y Ec.[2.11]) que necesita el mecanismo para poder desplazarse a la posición deseada. En la Fig. 3-29 se muestran 4 bloques con el nombre *Derivative*. Estos bloques nos permiten obtener la derivada con respecto al tiempo. De esta manera si derivamos la señal de entrada de por ejemplo la articulación prismática obtenemos la velocidad y su vez si volvemos a derivar la señal de salida del bloque *Derivative 1* obtenemos la aceleración.

Por último se agregaron sensores de fuerza y torque en las articulaciones prismática y de revolución. Los datos obtenidos durante la simulación son exportados a Workspace de Matlab.

El programa es ejecutado como se muestra en la Fig. 3-31, la articulación de revolución gira en sentido horario y para la articulación prismática se aplica fuerza hacia la pieza. Los datos obtenidos se comparan utilizando las Ec.[2.10] y Ec.[2.11] que se obtuvieron en el capítulo II.

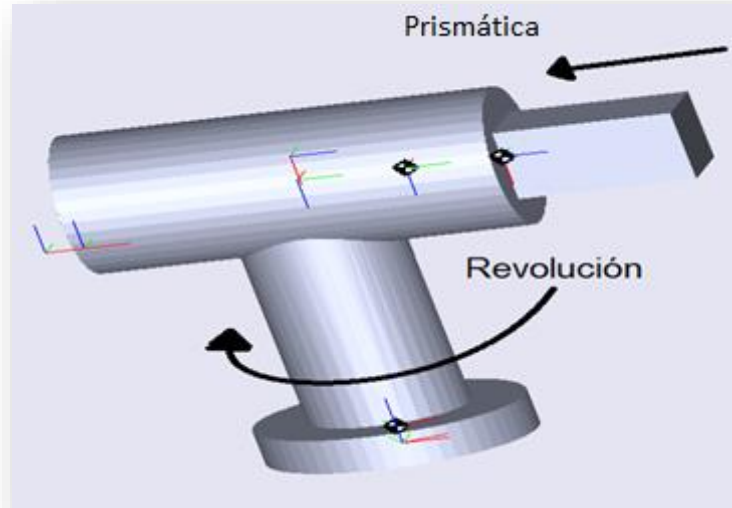


Fig. 3-31 Simulación de manipulador serial (RP).

De acuerdo con las Ec.[2.10] y Ec.[2.11] se necesitan tener los valores m_1 y m_2 también se necesita el valor de L_1 . Para los valores de posición, velocidad y aceleración se utilizaron los valores obtenidos en la simulación por medio de los bloques *Derivative*. En la Fig. 3-32 se muestran los primeros 7 valores que corresponden al tiempo de simulación de 0 -7 seg. Para cada simulación se obtienen 100 datos se genera 1 dato por cada segundo de simulación.

Con los movimientos que se representan en la Fig. 3-30 se realizaron tres pruebas. Primero un movimiento en la articulación de revolución dejando la articulación prismática libre esto quiere decir que no se aplica ninguna fuerza. En segundo lugar se aplicó un movimiento en la articulación prismática sin aplicar movimiento en la articulación de revolución y por último se aplicó el movimiento a las dos articulaciones.

```
>> Fuerza.signals.values
ans = 1.0e-03 *
    Posición  Velocidad  Aceleración  Fuerza
         0         0         0         0
    -0.8000   -0.4000   -0.2000   -0.1151
    -1.6000   -0.4000         0   -0.0149
    -2.4000   -0.4000         0   -0.0146
    -3.2000   -0.4000    0.0000   -0.0143
    -4.0000   -0.4000   -0.0000   -0.0141
    -4.8000   -0.4000         0   -0.0138
```

Fig. 3-32 Valores de la articulación prismática exportados desde el programa a bloques de Simulink hacia Workspace de Matlab.

Prueba 1 aplicando movimiento de revolución $0^\circ - 40^\circ$, y sin movimiento prismático. Para obtener los datos de posición, velocidad y aceleración, se utilizaron los datos exportados desde la simulación.

$$\theta_1=0.0524; \quad \dot{\theta}_1=0.0262; \quad \ddot{\theta}_1= 0.0131;$$

$$d_2=0 + 0.045; \quad \dot{d}_2=0; \quad \ddot{d}_2=0;$$

Valores obtenidos en simulación

Valores calculados

Fuerza= -0.1542-4 N.
Torque= 2.3726e-005 N*m.

Fuerza= -1.5445e-05 N.
Torque= 2.3744e-05 N*m.

Tabla 3-1 Prueba 1, Izquierda valores obtenidos en simulación, Derecha valores obtenidos utilizando las Ec.[2.10] y Ec.[2.11].

Prueba 2 aplicando movimiento prismático 0mm – 150mm, sin movimiento de revolución 0. Para obtener los datos de posición, velocidad y aceleración, se utilizaron los datos exportados desde la simulación.

$$\theta_1=0; \quad \dot{\theta}_1=0; \quad \ddot{\theta}_1= 0;$$

$$d_2= 0.600e-3 + 0.045; \quad \dot{d}_2= 0.3e-3; \quad \ddot{d}_2= 0.15e-3;$$

Valores obtenidos en simulación

Valores calculados

Fuerza= 0.0750e-3 N.
Torque= 0 N*m.

Fuerza= 7.50e-05 N.
Torque= 0 N*m.

Tabla 3-2 Prueba 2, Izquierda valores obtenidos en simulación, Derecha valores obtenidos utilizando las Ec.[2.10] y Ec.[2.11].

Prueba 3 aplicando movimiento prismático 0mm – (-150mm), movimiento de revolución 0° - 40°. Para obtener los datos de posición, velocidad y aceleración, se utilizaron los datos exportados desde la simulación.

$$\theta_1=0.0524; \quad \dot{\theta}_1=0.0262; \quad \ddot{\theta}_1= 0.0131;$$

$$d_2= -0.8000e-3 + 0.045; \quad \dot{d}_2= -0.4000e-3; \quad \ddot{d}_2= -0.2000e-3;$$

Valores obtenidos en simulación

Valores calculados

Fuerza= -1.1515e-4 N.
Torque= 2.2796e-005 N*m.

Fuerza= -1.1517e-04 N.
Torque= 2.2813e-05 N*m.

Tabla 3-3 Prueba 3, Izquierda valores obtenidos en simulación, Derecha valores obtenidos utilizando las Ec.[2.10] y Ec.[2.11].

Con los datos anteriores se puede observar que los resultados obtenidos mediante la simulación y los cálculos obtenidos en el análisis dinámico, varían alrededor de un 0.015%.

3.3.1. Simulaciones de la plataforma con la arquitectura propuesta

Una vez que se solucionó el problema del piso virtual y se realizaron pruebas de simulación dinámica. Se comenzó con las primeras simulaciones de la arquitectura en el entorno virtual de Simulink.

La entrada para los movimientos de las articulaciones en la simulación, fueron los programas que reproducen las gráficas obtenidas en el estudio del caminado humano. Los movimientos recopilados durante el estudio del caminado humano corresponden a una persona adulta caminando en línea recta.

Durante la simulación la arquitectura propuesta realiza dos ciclos del caminado humano. Los dos ciclos del caminado humano se representan en la simulación en una escala de 0 – 100 (ver, Fig. 3-33) en donde 0 -50 representa un ciclo del caminado humano y del 50 -100 representa el segundo ciclo del caminado humano.

Para la simulación se configura el *solver* de Matlab para que la simulación fuera de paso fijo de 1 seg. Esto nos ayuda a que el tiempo que tarda la computadora para el cálculo de la solución de la simulación es menor que al utilizar otro *solver* por ejemplo ODE.

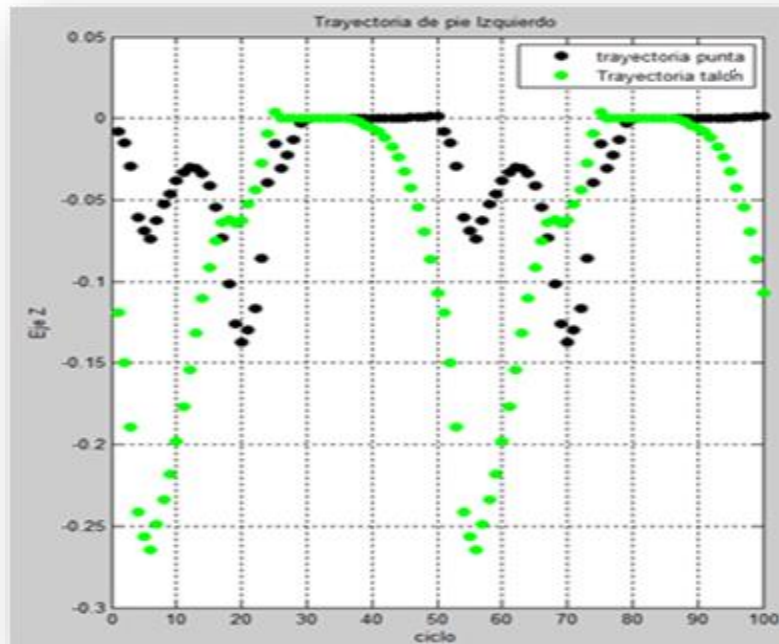


Fig. 3-33 Gráfica de posición de los pies izquierdo y derecho.

Como se muestra en la Fig. 3-34 se tomaron varios cuadros de este movimiento visto en diferentes planos. Como se muestra en la Fig. 3-34 plano sagital para la figura de la izquierda y plano coronal para la figura de la derecha.

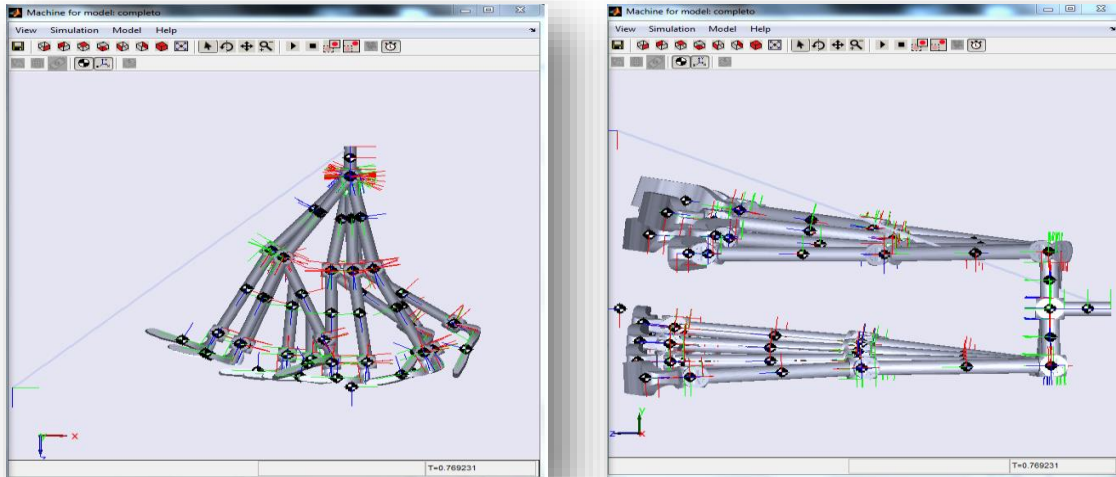


Fig. 3-34 Simulación de movimiento de las articulaciones de la arquitectura simplificada.

Durante la simulación todos los sensores que están conectados a las articulaciones van guardando los datos en archivos especiales de Matlab (.mat). Cuando la simulación termina se cierra el archivo y los datos guardados pueden ser llamados para procesarlos.

En los archivos .mat se guarda el tiempo de simulación y los valores de posición de los pies en los tres ejes (x,y,z).

3.3.2. Gráficas de movimiento de arquitectura propuesta

Para procesar los datos guardados de la simulación se creó un programa que se encarga de leer todos los archivos y guardarlos en variables dentro de Workspace en Matlab. En la Fig. 3-35 se muestran las gráficas de los sensores de los pies izquierdo y derecho en el eje Z. Estas gráficas nos ayudaron a determinar el deslizamiento en el eje Z durante la simulación de la arquitectura propuesta.

Como se muestra en la Fig. 3-35, el piso virtual se sitúa sobre la referencia en cero. Para las gráficas de los pies se utilizaron 4 sensores en cada pie, dos en las puntas y dos en el talón. En la Fig. 3-35 no se muestran las 4 gráficas debido a que los puntos están alineados por ejemplo los dos puntos de la punta del pie; los sensores adicionales podrían servir para estudios posteriores.

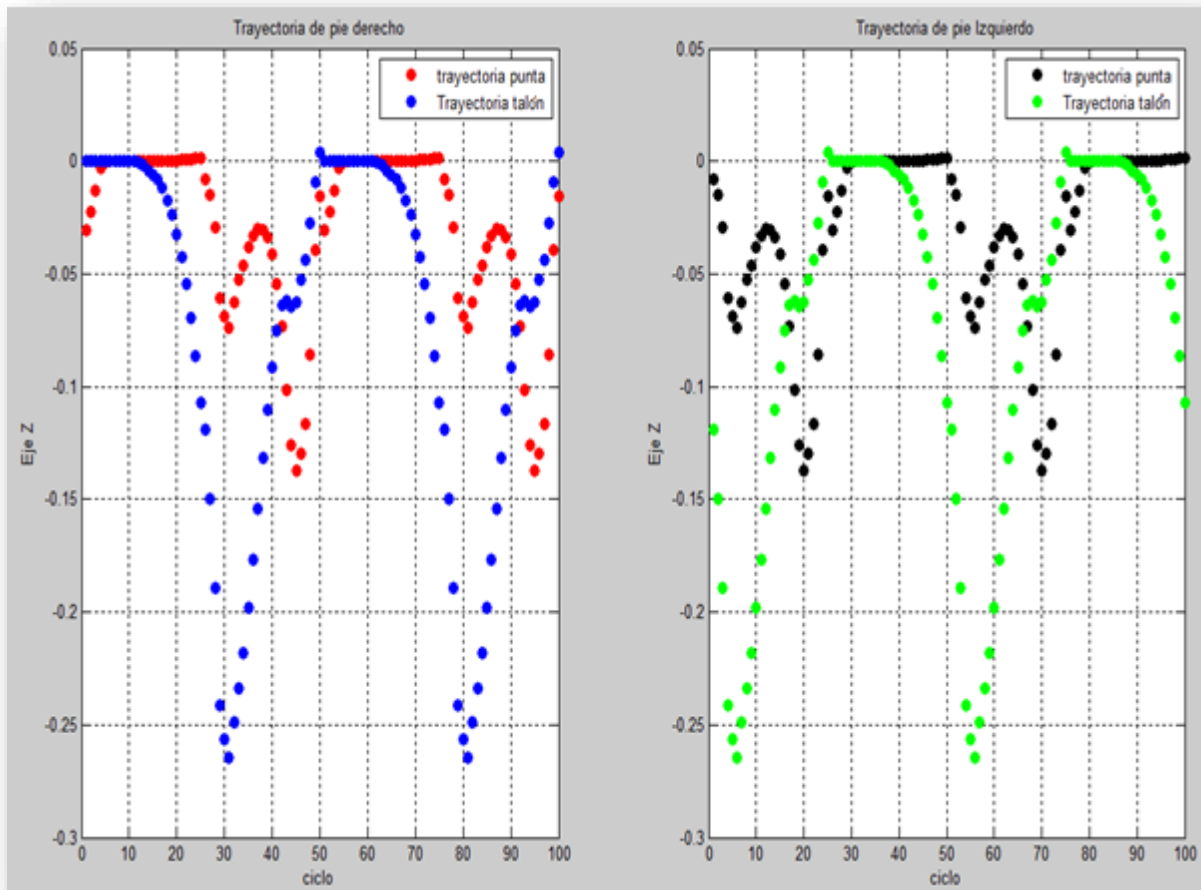


Fig. 3-35 Gráfica de posición de los pies izquierdo y derecho.

Con la ayuda del programa desarrollado también se obtuvieron los datos de posición de los pies durante la simulación del caminado de la arquitectura. Como se muestra en la Fig. 3-36 se grafica la trayectoria del pie izquierdo y derecho en el eje X; las gráficas representan dos ciclos del caminado. En la Fig. 3-37 se grafican las trayectorias de los pies izquierdo y derecho en el eje Y.

Con todos los datos obtenidos se grafican tridimensionalmente las trayectorias del pie a lo largo de la simulación como se muestra en la Fig. 3-38.

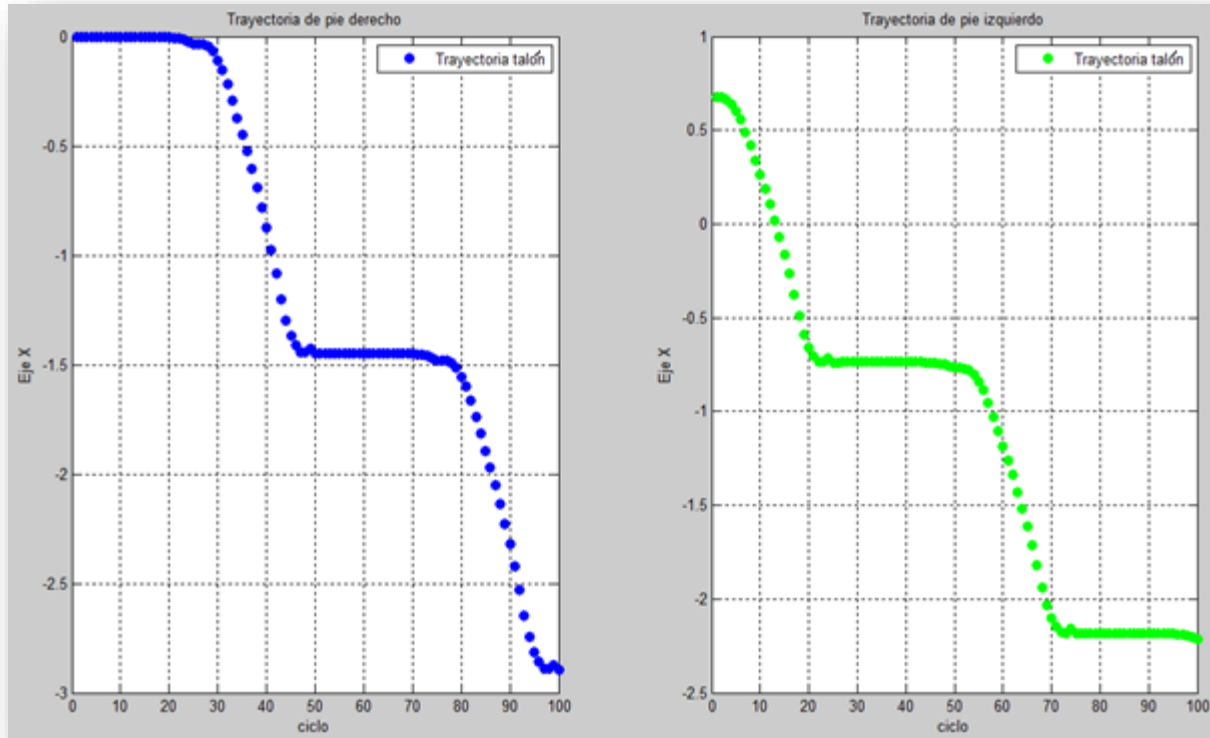


Fig. 3-36 Gráficas de trayectorias de los pies izquierdo y derecho en el eje X.

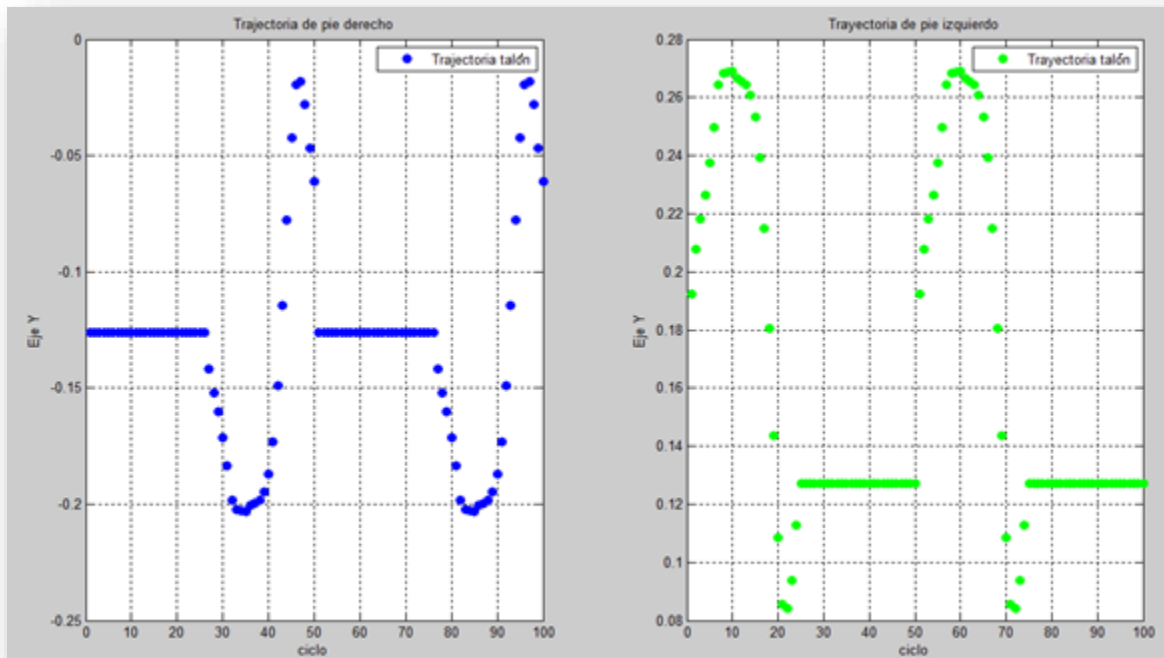


Fig. 3-37 Gráficas de trayectorias de los pies izquierdo y derecho en el eje Y.

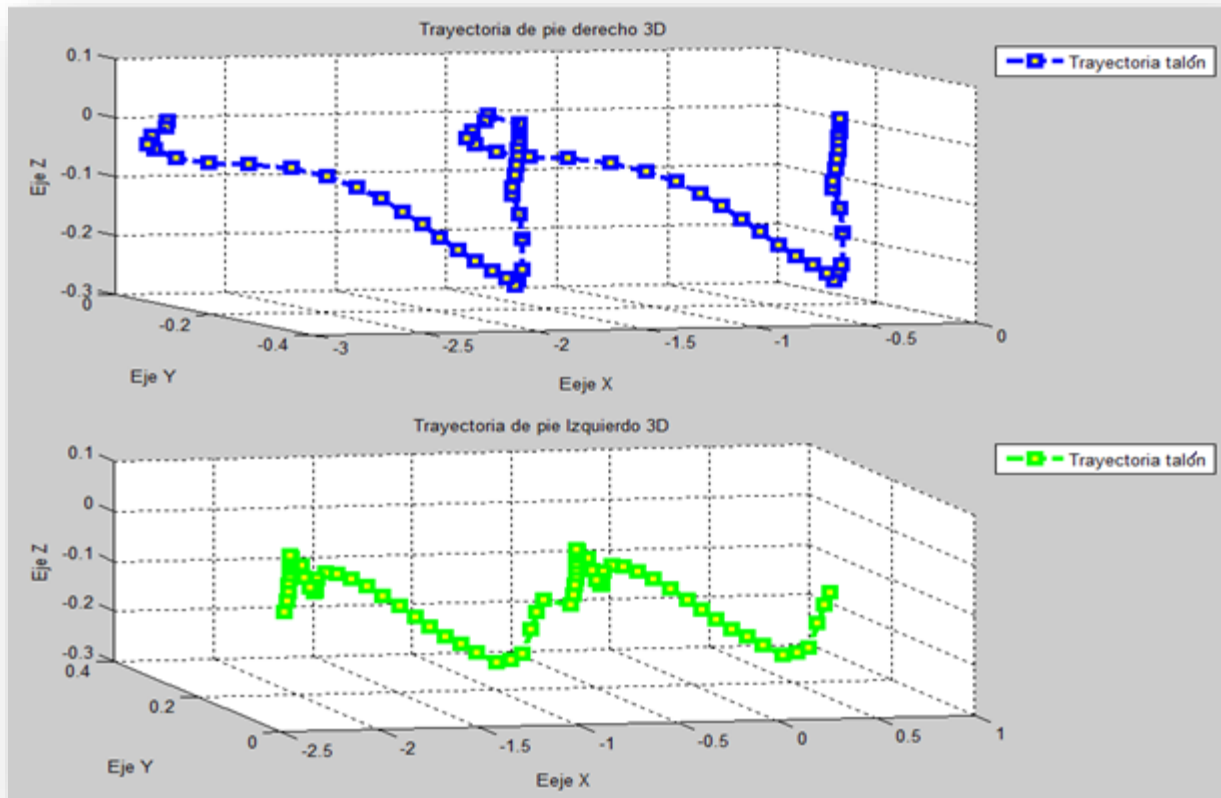


Fig. 3-38 Gráficas de trayectorias tridimensionales de los pies izquierdo y derecho.

Como el centro de masa es importante para determinar la estabilidad del caminado. Se obtuvieron gráficas de la trayectoria del centro de masa durante la simulación como se muestra en la Fig. 3-39. También se obtuvo la gráfica tridimensional del centro de masa como se muestra en la Fig. 3-40.

Por último se obtuvo la gráfica del centro de masa con respecto al área que forman los cuatro sensores puestos en las esquinas de los pies. En la Fig. 3-41 de color rojo se muestra el área del pie izquierdo, de color azul se muestra el área del pie derecho y por último de color negro translúcido se muestra un círculo, en el que se representa el centro de masa.

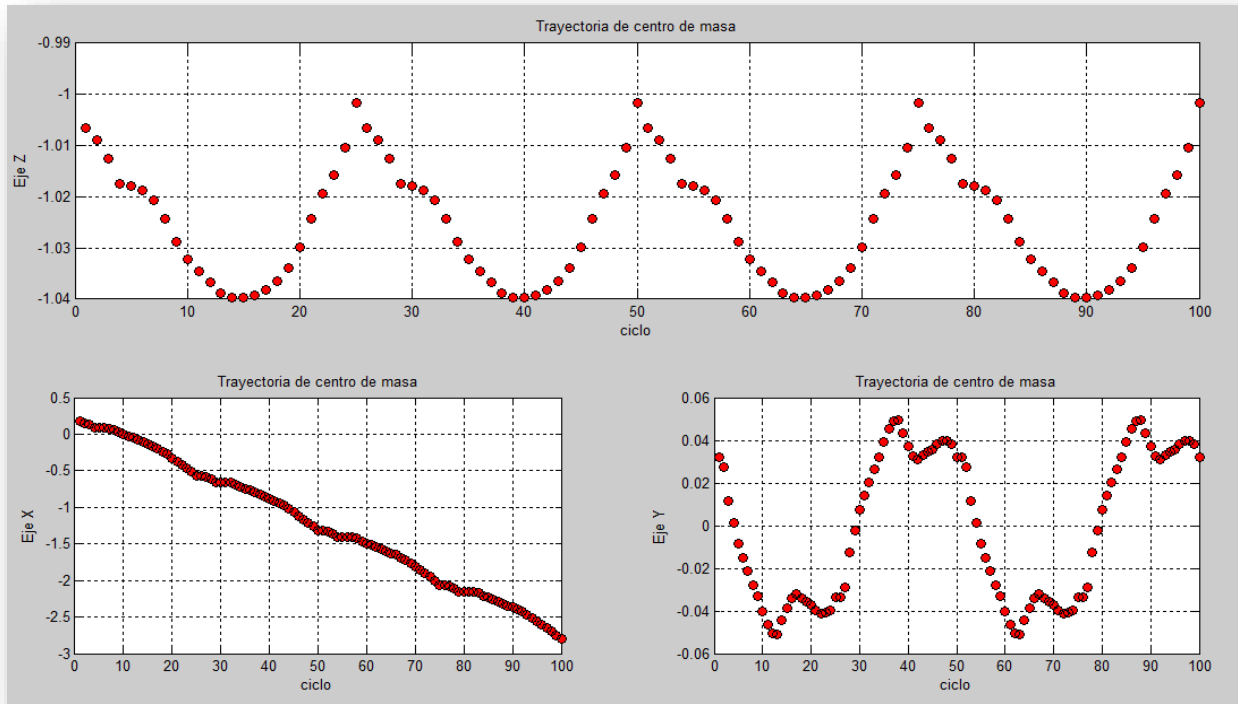


Fig. 3-39 Gráficas de trayectorias del centro de masa sobre cada plano.

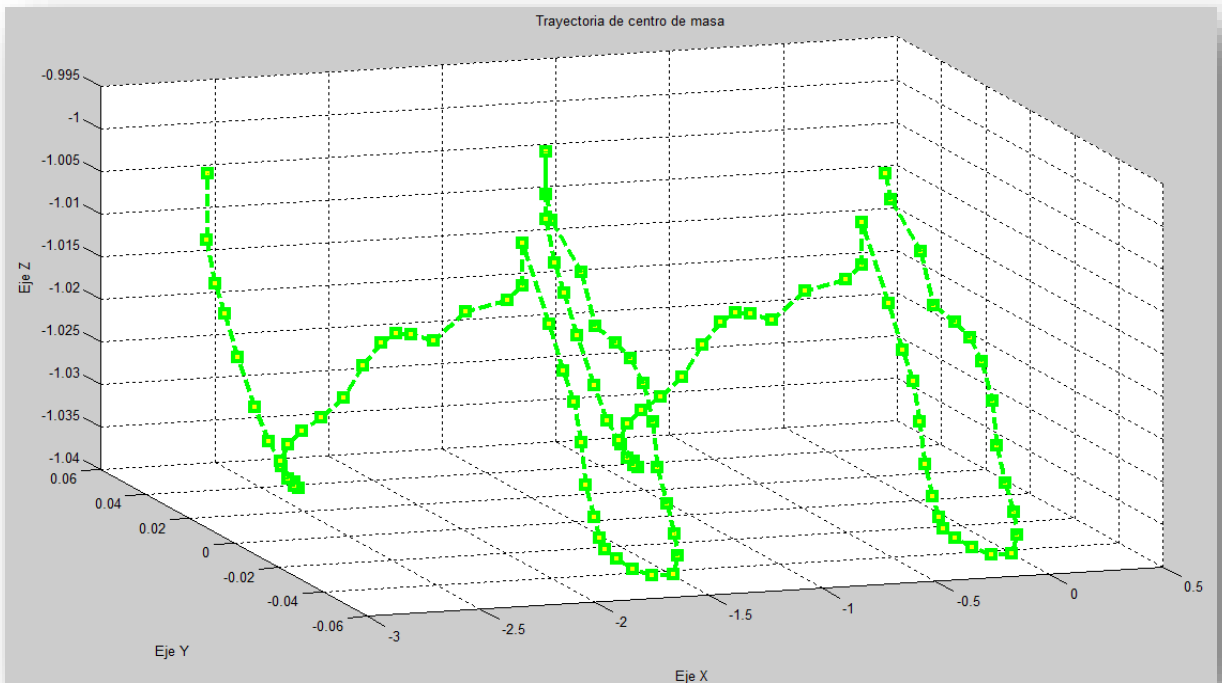


Fig. 3-40 Gráfica de trayectoria tridimensional del centro de masa.

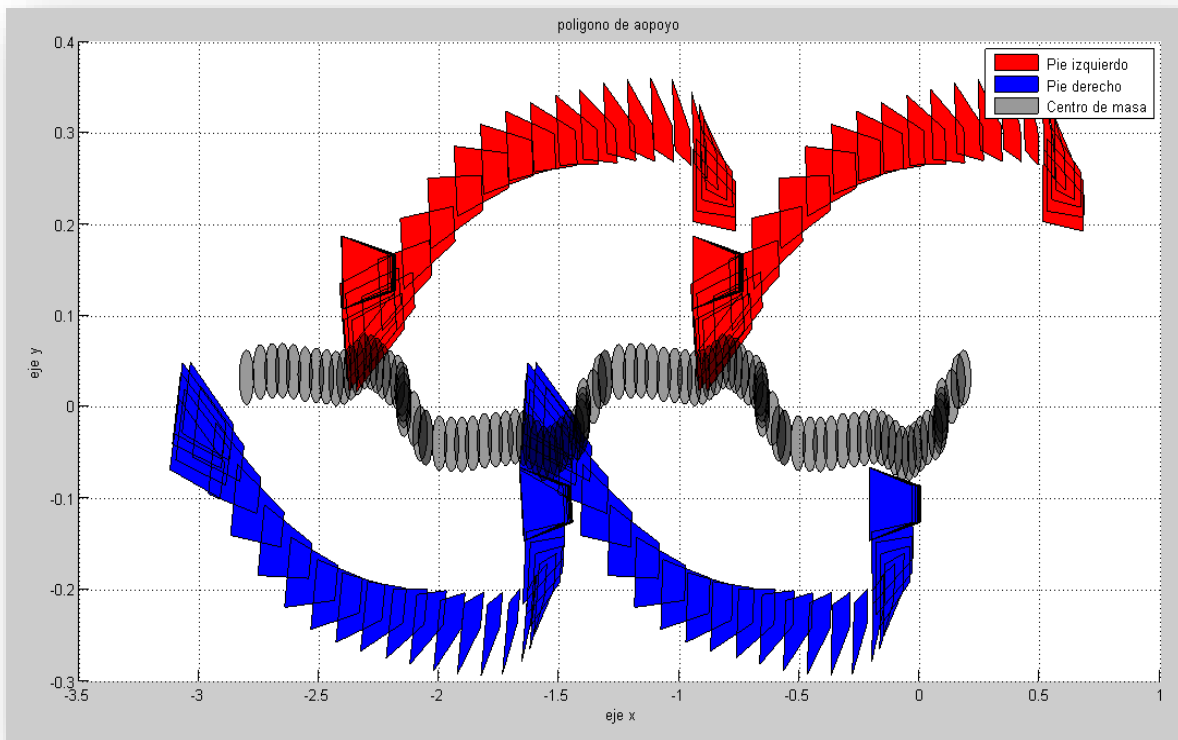


Fig. 3-41 Gráfica del área de los pies izquierdo, derecho y el centro de masa durante la simulación.

Durante este capítulo se desarrolló una plataforma de simulación que nos permita trabajar con robots bípedos utilizando dos herramientas muy conocidas que son Solidworks y Matlab. Al utilizar estas herramientas se pudo utilizar las características más sobresalientes de cada programa.

Con la ayuda de la plataforma de simulación se puede trabajar con arquitecturas complejas y simular su comportamiento cosa que es muy complicada casi imposible de hacer de forma analítica. También la plataforma nos ayuda a simular diferentes arquitecturas sin tener que cambiar muchas cosas.

A pesar de que Simulink no está diseñado para trabajar con robots móviles. Se pueden acomodar los programas, como se vio durante el desarrollo de este capítulo, para poder estudiar robots de este tipo. Las versiones más nuevas de Matlab contienen bibliotecas que permiten implementar modelos mecánicos más complejos.

Capítulo IV

Validación experimental en un prototipo

Una vez que la plataforma de simulación quedó completa se decidió probar los resultados en una arquitectura física construida en el CICATA-IPN. El robot está constituido por 6 grados de libertad (2 tobillos, 2 rodillas y 2 caderas). La idea es analizar el prototipo utilizando la plataforma de simulación e identificar las trayectorias que permitan el caminado del prototipo físico. Posteriormente se procede a programar y ejecutar las trayectorias en el prototipo.

Para el desarrollo de la validación experimental primero se modela el robot físico dentro del software CAD (Solidworks). Una vez que se obtiene el modelo CAD se creó un programa a bloques en Simulink-SimMechanics. Con este programa se creó una plataforma de simulación para el robot virtual de 6 grados de libertad. Cuando la plataforma de simulación quedó completa, se propusieron trayectorias para que el robot virtual realice un paso. Cuando se logró obtener una trayectoria en la que el robot virtual pueda caminar, se probaron los movimientos obtenidos en la plataforma de simulación, en el robot físico.

El robot físico cuenta con 6 servomotores que corresponde a los grados de libertad que tiene. Para lograr reproducir los movimientos en el robot físico se utilizó un microcontrolador que puede manejar servomotores. En la programación del microcontrolador se utilizó un entorno de desarrollo creado por el fabricante. Finalmente se desarrolló un programa para el micro controlador, el cual tendrá como objetivo mover todos los servomotores del robot físico.

4.1. Robot bípedo de 6 grados de libertad

Durante el desarrollo de la validación experimental se utilizó un robot físico construido en CICATA IPN-Querétaro con una configuración de 6 grados de libertad como se muestra en la Fig. 4-1

El particular interés en el estudio de este robot físico es debido a que durante la construcción del robot no se realizó un análisis profundo del caminado. Por lo tanto cuando se terminó con la construcción del robot y se realizaron pruebas del caminado se observó que el robot tenía problemas para lograr un paso. Las pruebas que se desarrollaron fueron a prueba y error

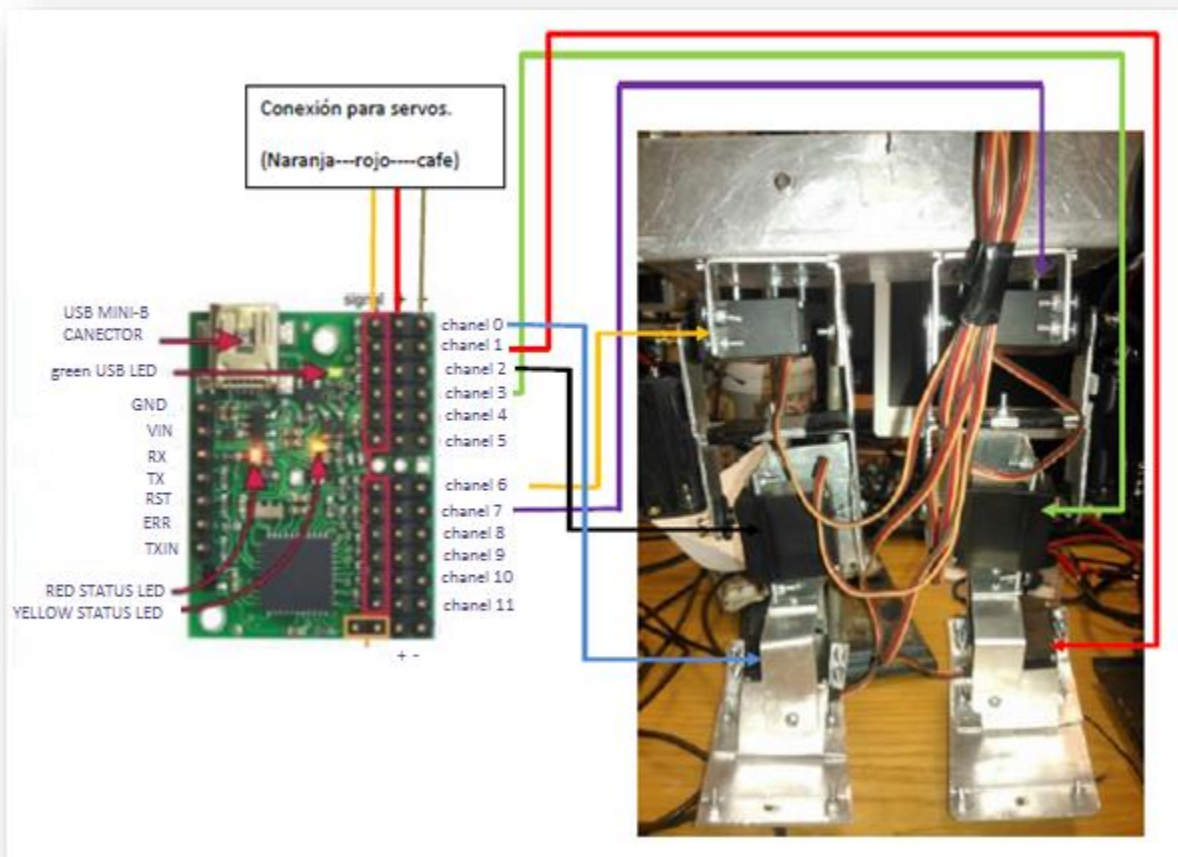


Fig. 4-1 Robot de 6 grados de libertad construido en CICATA-IPN.

La forma que se encontró para hacer que se desplace el robot, fue deslizando los pies del robot sin despegarlos del suelo. Teniendo como antecedentes estas pruebas se decidió estudiar el robot para determinar si efectivamente no hay ninguna forma de que pueda caminar sin tener que deslizar los pies.

4.2. Vitalización del prototipo

La primera tarea para analizar el robot fue construir un modelo CAD de la estructura. Se tomaron en cuenta todas las medidas y las configuraciones de los servo motores utilizados. Como se muestra en la Fig. 4-2 para la reproducción de las partes del robot se tuvo que dimensionar cada uno de los eslabones desarmando algunas partes para poder medirlas de manera adecuada como se muestra en Fig. 4-3.

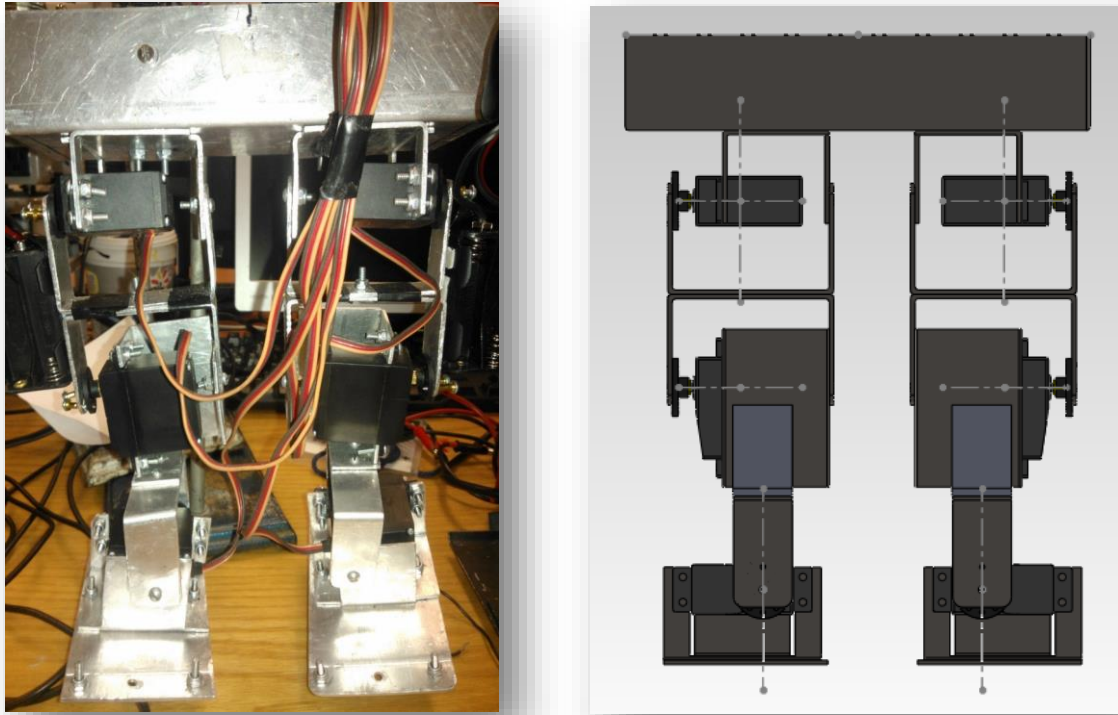


Fig. 4-2 Izquierda, Robot de 6 grados de libertad construido en CICATA-IPN. Derecha ensamble en Solidworks de robot.

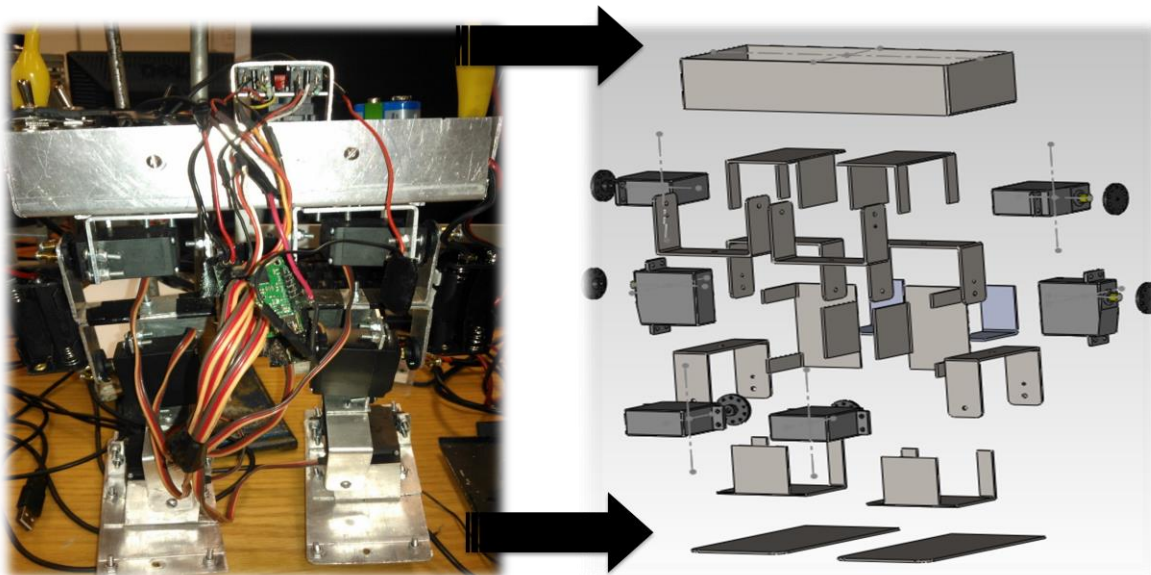


Fig. 4-3 Izquierda, Robot de 6 grados de libertad construido en CICATA-IPN. Derecha ensamble en vista explosión en Solidworks.

La manera de ensamblar las piezas dentro del modelo CAD fueron las mismas que se utilizaron para el estudio la arquitectura propuesta. Aplicando solamente restricciones rotacionales a las articulaciones que en este caso es el eje del servo motor.

4.3. Programa a bloques de robot de CICATA

Utilizando la herramienta para exportar ensambles de Solidworks a Matlab se obtuvo un programa a bloques que representa la arquitectura robótica dentro de la plataforma como se muestra en Fig. 4-4. Previamente se seleccionaron los materiales en el entorno CAD con los que fue construido el robot.

Los bloques que representan las restricciones y cuerpos fueron ordenados para poder manejar más fácil el programa. El programa final quedo como se muestra en la Fig. 4-4 en donde se puede observar que los bloques fueron acomodados en las diferentes partes que conforman el robot cadera, rodilla, tobillo y cuerpo.

Se agregaron actuadores y sensores en cada una de las 6 articulaciones del robot (ver, Fig. 4-5). También a cada sensor se le agregó un bloque para exportar los datos a archivos.

Utilizando el programa a bloques se obtuvieron datos como centro de masa de cada cuerpo, posición de los cuerpos, ángulo. Todos los datos son almacenados en un archivo de Matlab (.m) en donde se guardaron los parámetros del robot para posteriormente trabajar con ellos.

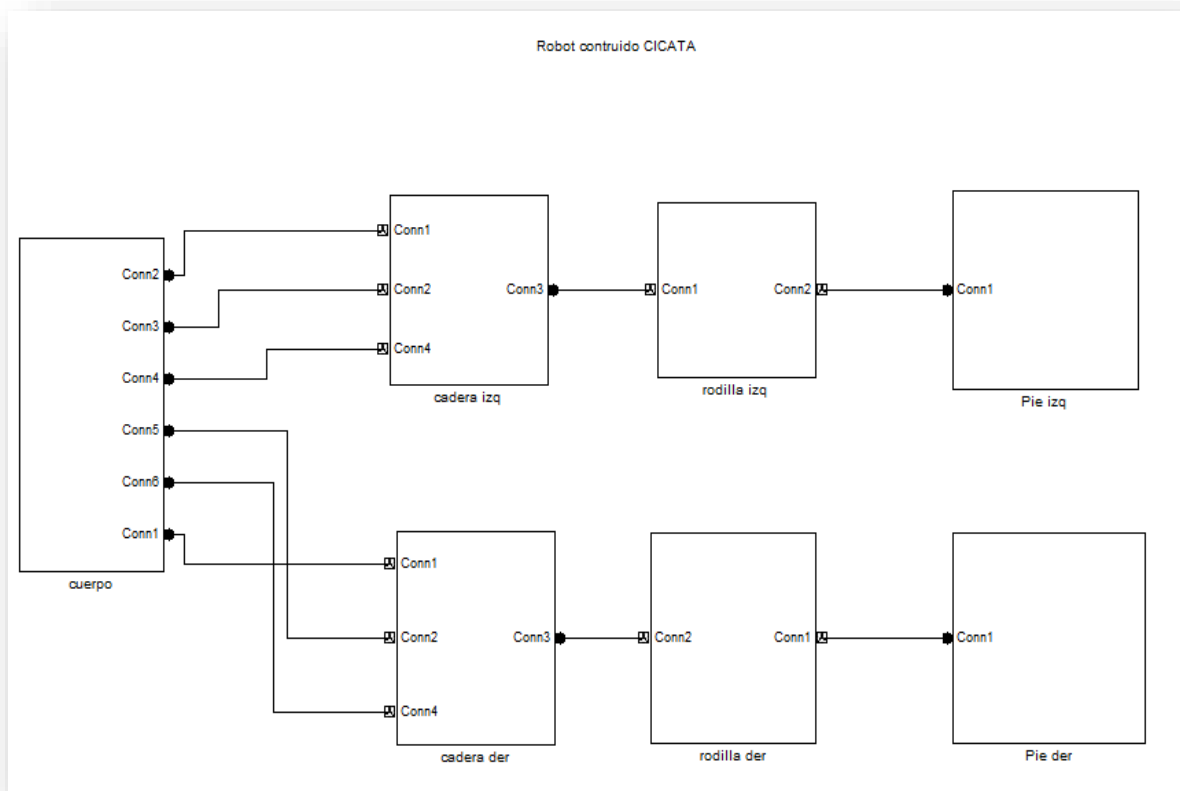


Fig. 4-4 Programa a bloques en Simulink, de robot de 6 grados de libertad.

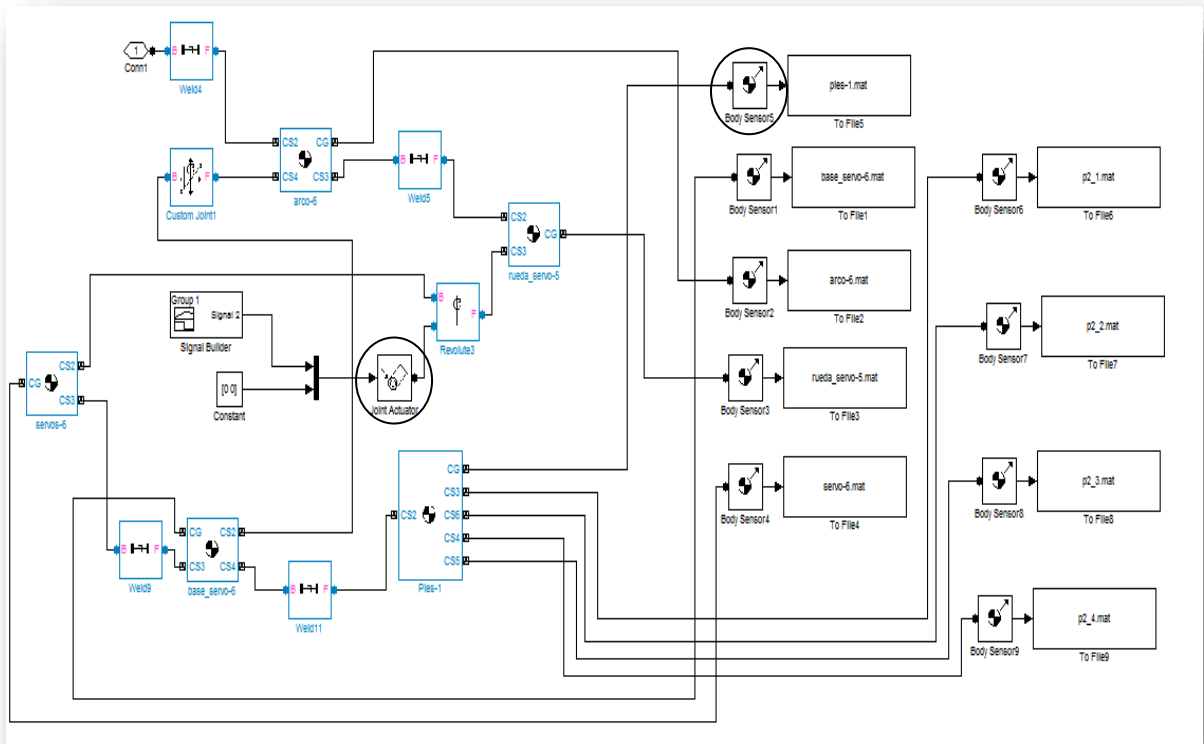


Fig. 4-5 Sensores agregados para obtener datos de centro de masa y posición de pie derecho.

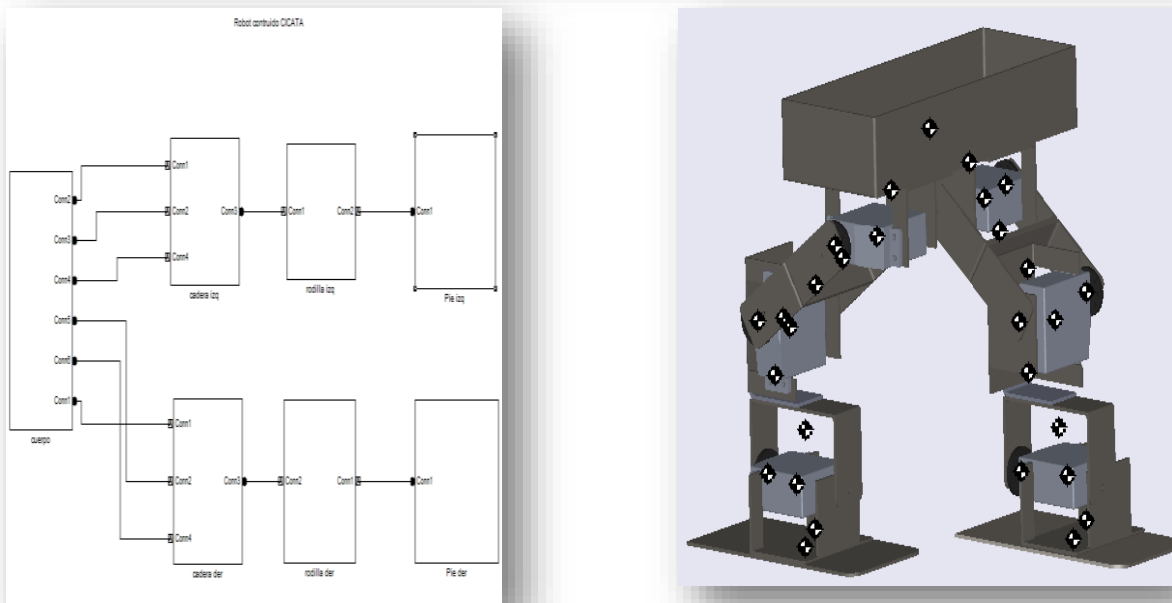


Fig. 4-6 Derecha programa a bloques de robot de 6 grados de libertad. Izquierda ventana de simulación de robot 6 grados de libertad.

Una vez que se tiene el modelo en Matlab se puede continuar con la simulación. El modelo virtual se puede utilizar para simular sin tener que estar experimentando físicamente con el robot. Esto evita

daños a los servomotores o estar programando un microcontrolador que mueva los servomotores a las posiciones deseadas para cada prueba.

4.4. Propuesta de trayectorias para robot virtual de 6 grados de libertad

La plataforma de simulación fue acondicionada para obtener todos los datos del robot durante la simulación. Sin embargo el robot no cuenta con ningún movimiento. Para mover el robot virtualmente hay que proponer algún movimiento en las articulaciones. Sin embargo encontrar trayectorias que permitan al robot desplazarse es una tarea complicada lo suficientemente retardadora como para dedicar un estudio completo a encontrar los movimientos que le permitan al robot caminar.

Para determinar el movimiento de las articulaciones, se decidió utilizar un método simplificado utilizando el centro de masa que se obtiene mediante la plataforma de simulación. La idea es utilizar el centro de masa proyectado en el piso y tratar de que el centro de masa no abandone el polígono de apoyo del robot. Debido a que el prototipo no cuenta con control de torque de los motores. Los movimientos realizados por el robot tienen que ser muy lentos para que la dinámica del robot no afecte mucho el resultado final.

El centro de masa va cambiando conforme el robot se va movimiento. Para encontrar el centro de masa en todo momento se desarrolló un segundo programa en Matlab. Donde se recopilan los datos obtenidos en cada simulación y se grafican para saber en dónde se encuentra el centro de masa durante cada paso de la simulación. Un diagrama de flujo se encuentra en el anexo A-1.A-3, para ver el programa completo consultar el CD de apoyo.

Para proponer una trayectoria en donde se pueda desplazar el robot, primero se trató de pasar el centro de masa a un solo polígono de apoyo como se muestra en la Fig. 4-8. El centro de masa de color negro se movió hacia el polígono de apoyo (color azul) que representa el pie derecho del robot.

En la Fig. 4-7 aparecen 3 recuadros, uno grande de color verde translúcido que encierra a dos más pequeños en su interior uno rojo muy claro (a la izquierda en la Fig. 4-7, representa pie izquierdo) y uno azul que se ve más oscuro (a la derecha en la Fig. 4-7, representa el pie derecho). El recuadro más grande representa el área total de apoyo del robot. Esto quiere decir que si el robot se apoya en sus dos pies el área o polígono de apoyo estará representado por el recuadro verde translúcido que aparece sobre los dos recuadros más pequeños. Como se muestran en las Fig. 4-10 a la Fig. 4-13 el área parece de color rojo y azul cuando el polígono de apoyo está formado por un solo pie y aparece un recuadro verde translúcido sobre los recuadros rojo y azul cuando los dos pies están en el suelo.

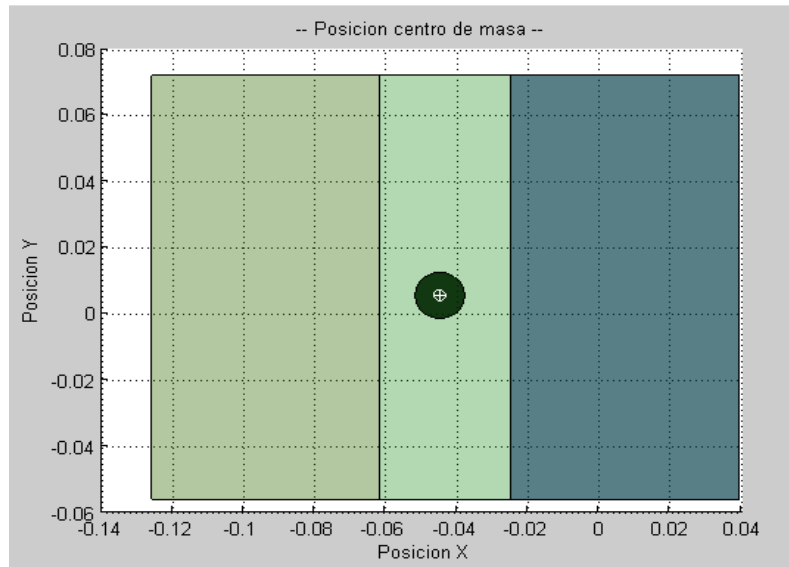


Fig. 4-7 Área de soporte para el robot.

Cuando solo un pie está apoyado sobre el piso desaparece el área translúcida de color verde y queda solamente el pie que está apoyado sobre el suelo como se muestra en la Fig. 4-8. Marcando el área de apoyo de un color más fuerte.

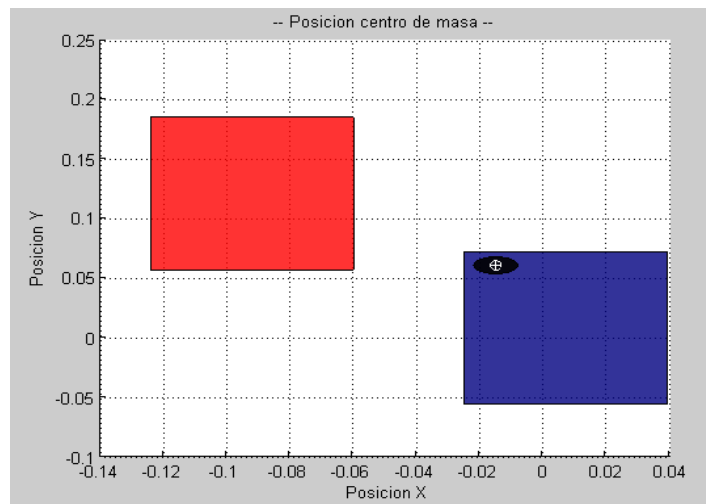


Fig. 4-8 Gráfica donde se representa la posición de los pies durante la simulación, el centro de masa se desplaza sobre el pie derecho.

Utilizando las gráficas se trató de posicionar siempre el centro de masa en el polígono de apoyo moviendo lentamente los servomotores, para despreciar las fuerzas ejercidas por la dinámica del mecanismo.

Se desarrolló un tercer programa en Matlab para proponer movimientos en las articulaciones del robot sin tener que estar abriendo el programa a bloques (ver, Fig. 4-9). De esta manera permite cambiar

valores fácilmente y correr la simulación de manera rápida. El vector (time) representa el tiempo de simulación en segundos y el vector (data) representa la posición en grados de los servomotores en el tiempo correspondiente.

Programa para exportar datos a Simulink.

```

clc;
clear all;
%-----pie de apoyo
% pie der
data=[0 0 20 20 20 7 7 0];
time=[0 5 7 9 10 15 16 20];
signalbuilder('Robot_3_DOF/Pie der/S1', 'set', 1, 1, time, data);%
% rodilla der
data=[0 55 55 55];
time=[0 5 10 20];
signalbuilder('Robot_3_DOF/rodilla der/Signal Builder', 'set','Signal 1', 'Group
1',time,data);
% cadera der
data=[0 55 55 55];
time=[0 5 10 20];
signalbuilder('Robot_3_DOF/cadera der/Signal Builder', 'set','Signal 1', 'Group
1',time,data);
%-----pie balanceando
% pie izq
data=[0 0 5.5 20 20 20 7 7 0];
time=[0 5 5.5 7 10 12 15 16 20];
signalbuilder('Robot_3_DOF/Pie izq/Signal Builder', 'set', 'Signal 2', 1, time,
data);%
% rodilla izq
data=[0 -55 -40 0 45 45 45 55];
time=[0 5 6 7 9 10 16 20];
signalbuilder('Robot_3_DOF/rodilla izq/Signal Builder', 'set','Signal 1', 'Group
1',time,data);
% cadera izq
data=[0 -55 -40 0 45 45 45 55];
time=[0 5 6 7 9 15 16 20];
signalbuilder('Robot_3_DOF/cadera izq/Signal Builder', 'set','Signal 1', 'Group
1',time,data);

```

Fig. 4-9 Código de simulación de movimiento del robot.

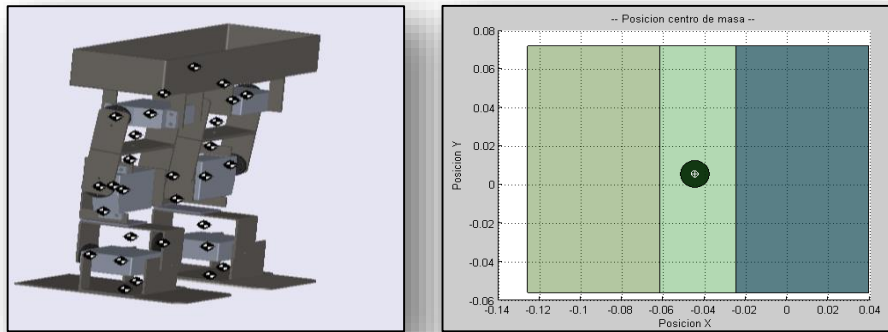


Fig. 4-10 Robot moviendo el centro de masa hacia delante.

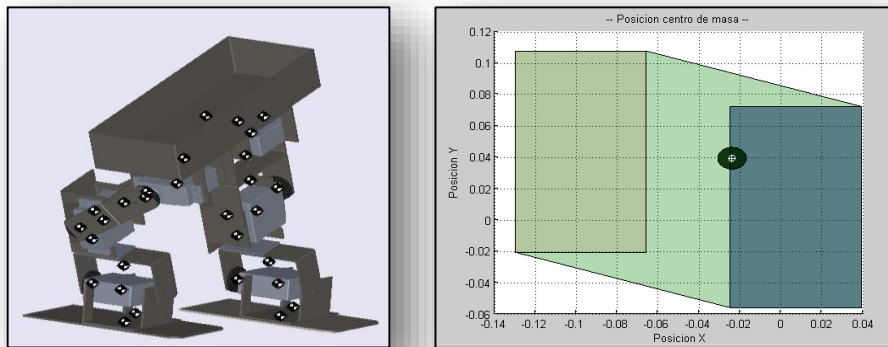


Fig. 4-11 Robot empieza a desplazar el centro de masa hacia el área de pie izquierdo.

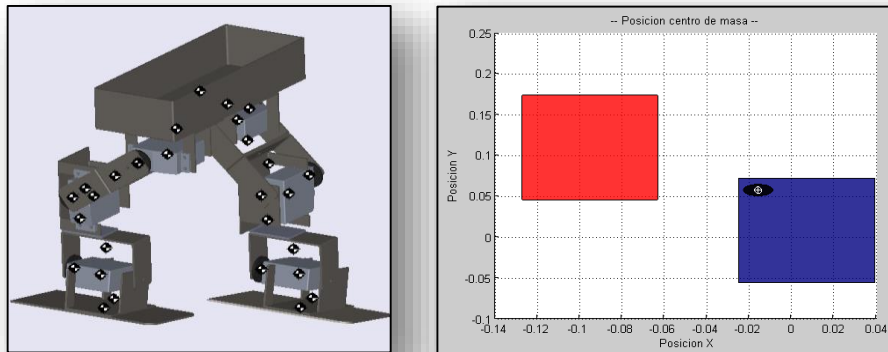


Fig. 4-12 Robot pasa el pie derecho hacia delante.

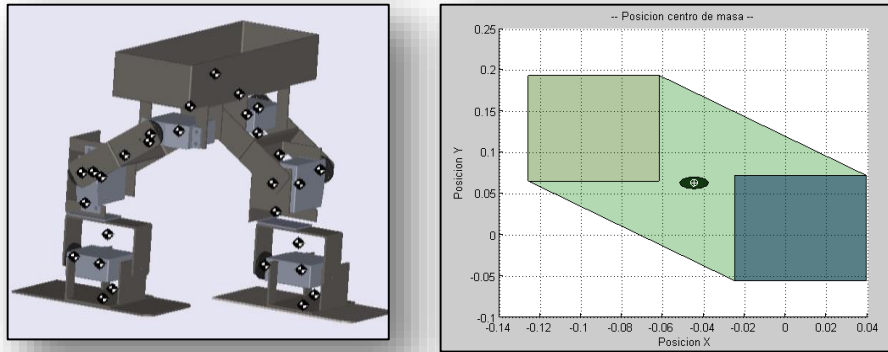


Fig. 4-13 El robot apoya los dos pies en el piso ($Z=0$) y el centro de masa regresa a la mitad del polígono de apoyo.

En las Fig. 4-10 a Fig. 4-13 se muestra una parte de la simulación realizada en los minutos 2, 8, 15,10 para describir el movimiento caminado propuesto para el robot.

Para los movimientos que se encontraron en el robot virtual, el centro de masa se encontraba muy cerca del límite del polígono de apoyo. A pesar de este inconveniente el robot virtual puede lograr un paso levantando las extremidades en lugar de deslizarlas.

4.5. Validación experimental

Con la ayuda de la plataforma de simulación se requirió de un menor tiempo de prueba para las posiciones de las articulaciones. Gracias a que en la plataforma de simulación se pueden probar diferentes posiciones rápidamente sin tener que estar manipulando el robot físico.

Las posiciones para mover el robot que se utilizaron durante la primera parte de la simulación, en donde el robot se inclina para dar un paso (Fig. 4-10-Fig. 4-13), quedaron como se muestra en la Fig. 4-14. Todas las posiciones que se utilizaron durante la simulación del robot quedan dentro del programa que se muestra en la Fig. 4-9 el cual se anexa en el CD de apoyo. El vector data representa las posiciones de los servomotores (grados) y el vector data representa el tiempo (seg.) en el que se ejecutaron las posiciones en la simulación.

```

%-----pie de apoyo
% pie der
data=[0 0 20 20 20 7 7 0];
time=[0 5 7 9 10 15 16 20];
% rodilla der
data=[0 55 55 55];
time=[0 5 10 20];
% cadera der
data=[0 55 55 55];
time=[0 5 10 20];
%-----pie balanceando
% pie izq
data=[0 0 5.5 20 20 20 7 7 0];
time=[0 5 5.5 7 10 12 15 16 20];
% rodilla izq
data=[0 -55 -40 0 45 45 45 55];
time=[0 5 6 7 9 10 16 20];
% cadera izq
data=[0 -55 -40 0 45 45 45 55];
time=[0 5 6 7 9 15 16 20];
    
```

Fig. 4-14 Datos utilizados para configurar rodilla, pie y cadera de ambas piernas del robot.

Los valores que se propusieron para la simulación (Fig. 4-14) son exportados al programa a bloques del robot virtual para iniciar con la simulación. Con los datos exportados se crean una curva uniendo los puntos como se muestra en la Fig. 4-15. Estos valores son usados para la posición de los servomotores. Los valores completos para las posiciones de los servomotores se encuentran en los programas en el CD de apoyo.

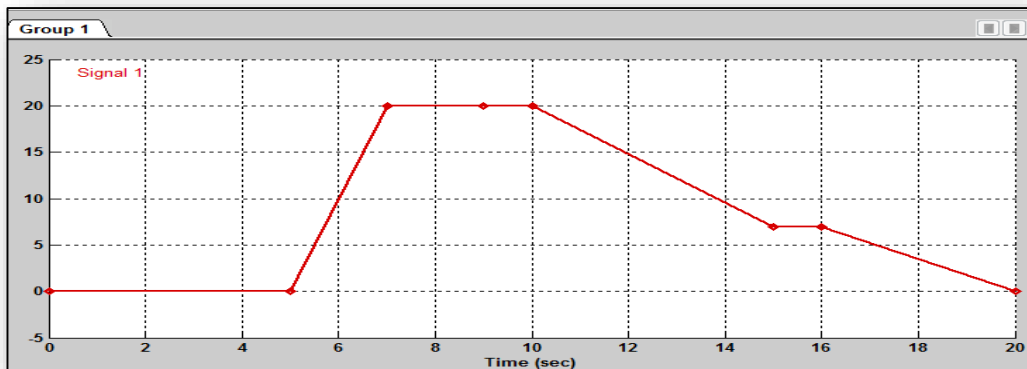


Fig. 4-15 Gráfica obtenida de los datos utilizando Simulink, representa movimiento de pie derecho.

Una vez que se lograron determinar los movimientos del robot, se decidió aplicar estos movimientos al robot físico. Para lograr que el robot se mueva se necesitó aplicar programación con micro controladores para mover los servomotores. En el caso de la prueba se utilizó la plataforma ARDUINO.

4.6. ARDUINO

Arduino es una plataforma electrónica de software y hardware abierta. Arduino puede tomar información del entorno a través de sus pines de entrada. Las entradas pueden ser analógicas o digitales. El microcontrolador en la placa de Arduino se programa mediante el lenguaje de programación Arduino y el entorno de desarrollo Arduino. Los proyectos hechos con Arduino pueden ejecutarse sin necesidad de conectar a una computadora.

Las placas pueden ser hechas a mano o compradas montadas de fábrica; el software puede ser descargado de forma gratuita. Los archivos de diseño de referencia (CAD) están disponibles bajo una licencia abierta

4.6.1. ARDUINO Mega 2560

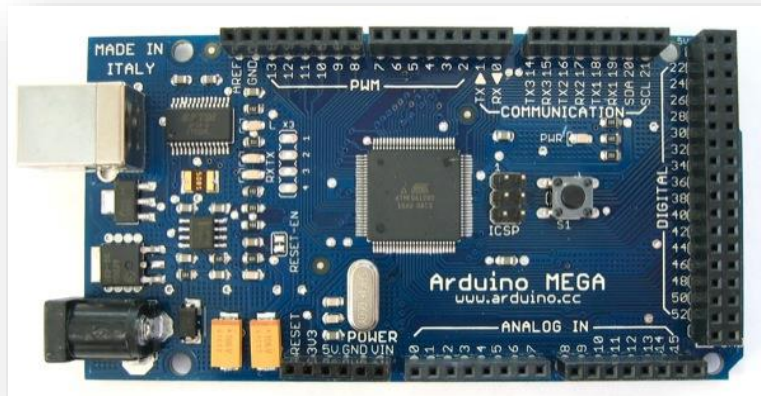


Fig. 4-16 Plataforma Arduino Mega 2560.

El Arduino Mega (Fig. 4-16) es una placa microcontrolador basada ATmeg1280. Tiene 54 entradas/salidas digitales (de las cuales 14 proporcionan salida PWM, utilizadas para control de servomotores), 16 entradas digitales, 4 UARTS (puertos serie por hardware), un cristal oscilador de 16MHz, conexión USB, entrada de corriente, conector ICSP y botón de reset.

Contiene todo lo necesario para hacer funcionar el micro controlador, simplemente conectándolo a la PC con el cable USB o aliméntalo con un transformador o batería. El Arduino Mega es compatible con la mayoría de las placas diseñadas para el Arduino Duemilanove o Diecimila.

4.7. Movimientos de robot de 6 grados de libertad

Los datos obtenidos en la plataforma son exportados a un programa desarrollado en Arduino (ver anexo A-1.A-4). El programa desarrollado nos permite mover todos los servomotores al mismo tiempo. Las posiciones de los servomotores son reproducidas por los motores una a una con una pausa de 1 seg entre cada dato. Esto nos ayuda a reducir un poco la velocidad con el que se reproducen las posiciones en los servomotores.

Las conexiones quedan como se muestra en la Fig. 4-17. La señal para las posiciones de los servomotores comprenden las salidas PWM (salida 0 - salida 5). Se utilizó una fuente de voltaje externa de 5v para el movimiento de los servomotores.

A pesar de que se disminuyó la velocidad de los servomotores pausando el movimiento entre cada posición, el movimiento resultante no fue con las características de velocidad esperadas. El cambio de una posición a otra los servomotores lo hacen a la máxima velocidad. Haciendo que el robot tenga cambios de posición muy bruscos y esto desestabiliza la estructura del robot.

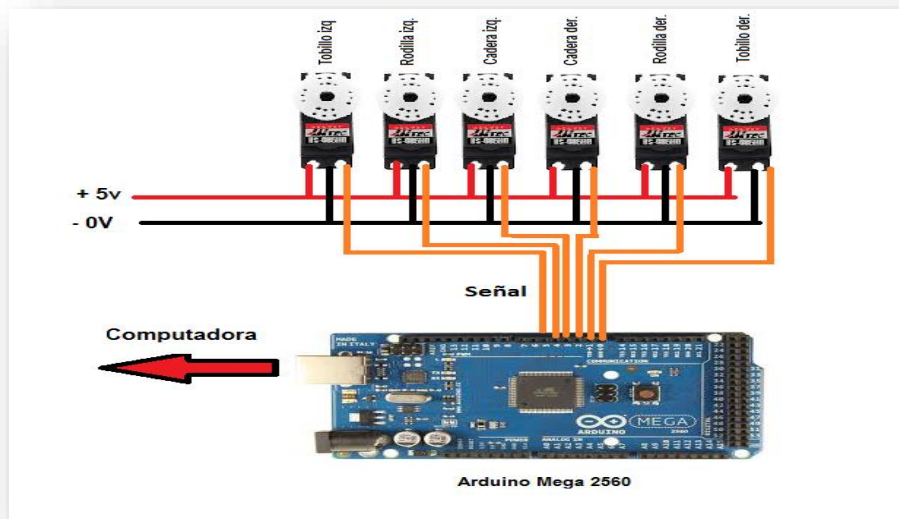


Fig. 4-17 Conexión de los servomotores con Arduino.

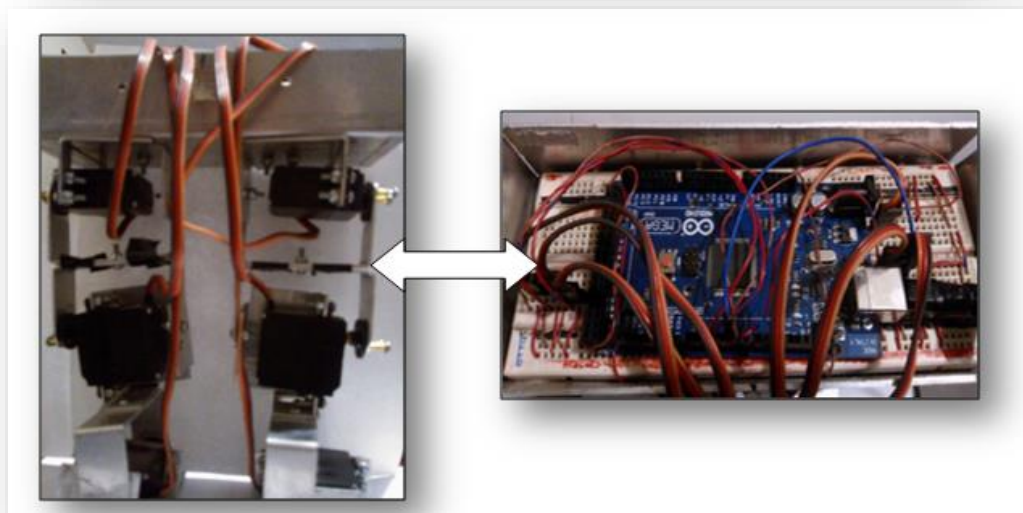


Fig. 4-18 Conexión física del robot con Arduino.

Cuando el robot físico reprodujo los movimientos obtenidos por medio de la plataforma de simulación, no se obtuvo el resultado esperado. En particular al reproducir los movimientos de caminado en el prototipo, éste pierde el equilibrio. Lo anterior es debido a que los eslabones con los que está construido el robot no están perfectamente alineados. Lo que produce que la posición obtenida en la plataforma de simulación no corresponda con la posición de los eslabones del robot físico. Cuando el

centro de gravedad del robot sale del polígono de apoyo se causa la pérdida de equilibrio. Además los movimientos que se obtienen con los servomotores empleados agravan la pérdida de equilibrio.

A pesar de lo anterior, el prototipo físico es capaz de reproducir aproximadamente los movimientos planeados lo que parcialmente corrobora el estudio realizado.

Al no contar con una plataforma física apta, la validación experimental no permite obtener el resultado esperado completo. Aún así, los resultados obtenidos permiten prever que al contar con un prototipo de mejores características se podrán complementar la validación en simulación con una validación experimental completa.

Conclusiones

Se desarrolló una plataforma de simulación para arquitecturas bípedas con un número y configuración arbitrarios de grados de libertad. La plataforma tiene las siguientes características: se pueden simular a nivel dinámico diferentes arquitecturas robóticas, capacidad para simular diferentes robots sin realizar cambios importantes en la plataforma, simulación cinemática, un entorno visual, capacidad de intercambiar información entre la plataforma y otros con programas. Se validó la plataforma con robots de dos grados de libertad simulando la dinámica de los mecanismos.

La plataforma desarrollada proporciona una herramienta de estudio de la habilidad para caminar de cualquier arquitectura bípeda. Se pueden estudiar y simular diferentes trayectorias y controles para caminado así como evaluar cuantitativamente los resultados.

Se validaron los resultados obtenidos en simulación en un prototipo real. En esta validación se pudo comprobar que incorporar un nuevo prototipo para la simulación requiere de poco esfuerzo. Realizar pruebas con el robot virtual es rápido sin embargo se tuvieron dificultades para modelar y simular un piso virtual.

Durante el capítulo I se estudió el caminado humano definiendo sus características, también se estudiaron algunos de los más relevantes prototipos bípedos que existen en la actualidad y por último se realizó una revisión de algunos programas de simulación para robots. Entre estos programas se encontraron algunos con características similares a la plataforma de simulación como V-REP Y WEBOT. Sin embargo una de la plataforma de simulación WEBOT es cara. La plataforma V-REP recientemente se volvió de acceso gratuito para estudiantes y habría que evaluarla para determinar su aptitud para este tipo de simulaciones.

Para modelar un robot bípedo se necesita encontrar ecuaciones que representen la física del robot por ejemplo un modelo cinemático y dinámico. Muchas veces encontrar estas ecuaciones de manera analítica resulta complicado y tardado. Una propuesta para lidiar con el problema fue desarrollar un programa que calcule la dinámica y cinemática de un mecanismo serial. Sin embargo a pesar del desarrollo del programa encontrar las ecuaciones dinámicas sigue siendo complicado debido a que los términos que conforman las ecuaciones son muy largos, lo que no permite manejar las ecuaciones de manera tradicional. Tener la capacidad de simular movimientos de un robot en un software de simulación dinámica, permite acortar el tiempo de análisis del mecanismo.

El robot físico que fue construido en el CICATA requirió de una inversión de tiempo muy corto para obtener un modelo virtual con el que se pudieran realizar pruebas. Con el modelo virtual del robot del CICATA se realizaron pruebas para determinar si el robot tenía la capacidad de caminar. En la simulación se pudo obtener una trayectoria de caminado para el robot.

Por otro lado cuando el robot físico reprodujo los movimientos obtenidos por medio de la plataforma de simulación, no se obtuvo el resultado esperado. En particular al reproducir los movimientos de caminado en el prototipo, éste pierde el equilibrio. Lo anterior es debido a que los eslabones con los que está construido el robot no están perfectamente alineados. Lo que produce que la posición obtenida en la plataforma de simulación no corresponda con la posición de los eslabones del robot físico. Cuando el centro de gravedad del robot sale del polígono de apoyo se causa la pérdida de equilibrio. Además los movimientos que se obtienen con los servomotores empleados agravan la pérdida de equilibrio. A pesar de lo anterior, el prototipo físico es capaz de reproducir aproximadamente los movimientos planeados lo que parcialmente corrobora el estudio realizado.

Al no contar con una plataforma física apta, la validación experimental no permite obtener el resultado esperado completo. Aún así, los resultados obtenidos permiten prever que al contar con un prototipo de mejores características se podrán complementar la validación en simulación con una validación experimental completa.

Un contratiempo para el desarrollo de la tesis fue a construcción del prototipo físico. Debido a su construcción el robot tenía poca estabilidad cuando realizaba movimientos. Algunos eslabones estaban desacomodados lo que producía que las posiciones de los servomotores no fueran respetados lo que producía que la planta del pie del robot no llegara a la posición deseada.

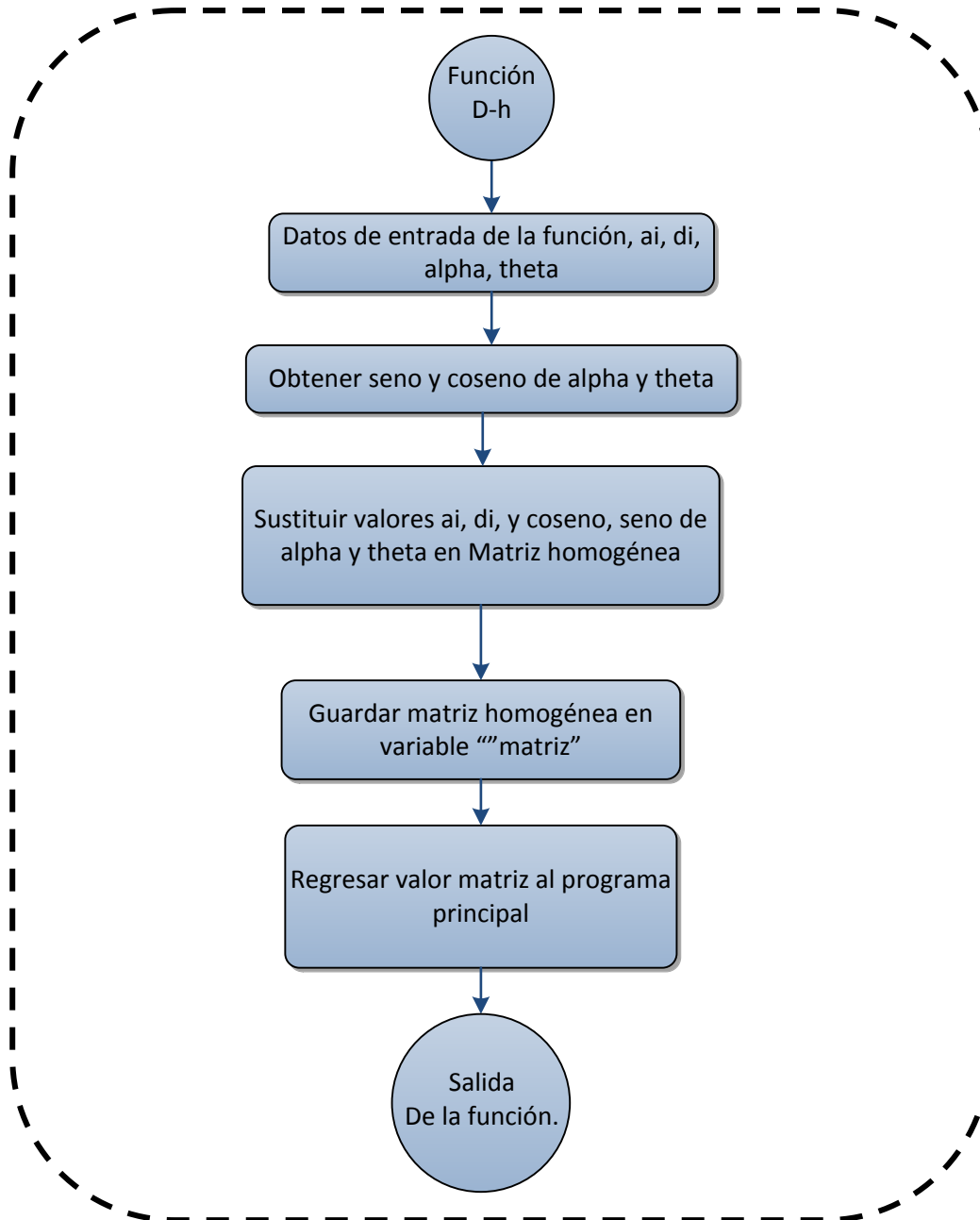
Trabajo a futuro

Dentro del trabajo que complementaría el resultado de esta tesis se encuentra:

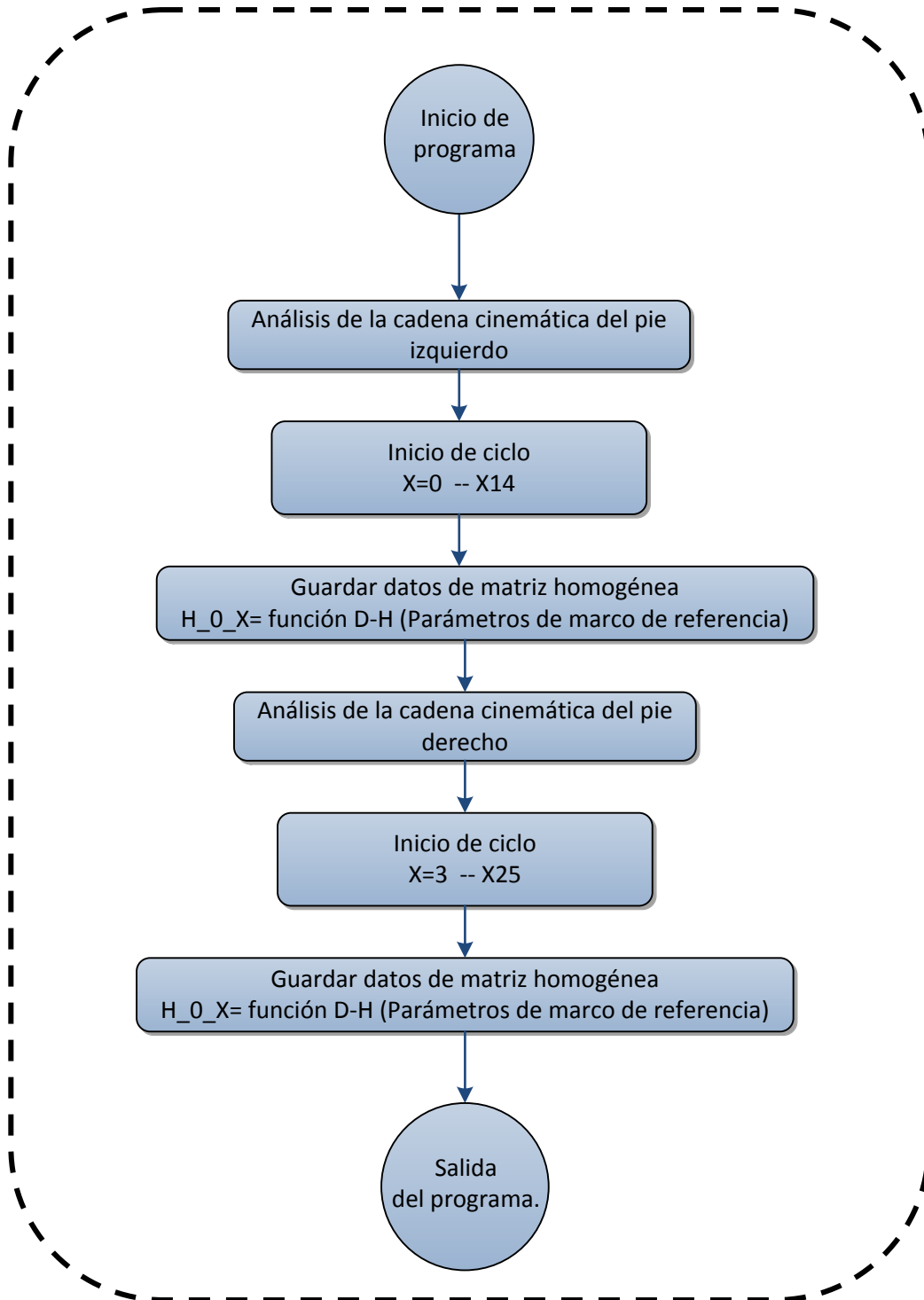
- Validación experimental utilizando un prototipo bípedo diferente, con una mejor construcción.
- Estudio de diferentes arquitecturas. Tal estudio ayudaría en la identificación de arquitecturas simplificadas que puedan caminar.
- Diseño de algoritmos de control para la tarea de caminado de diferentes arquitecturas.

Anexos

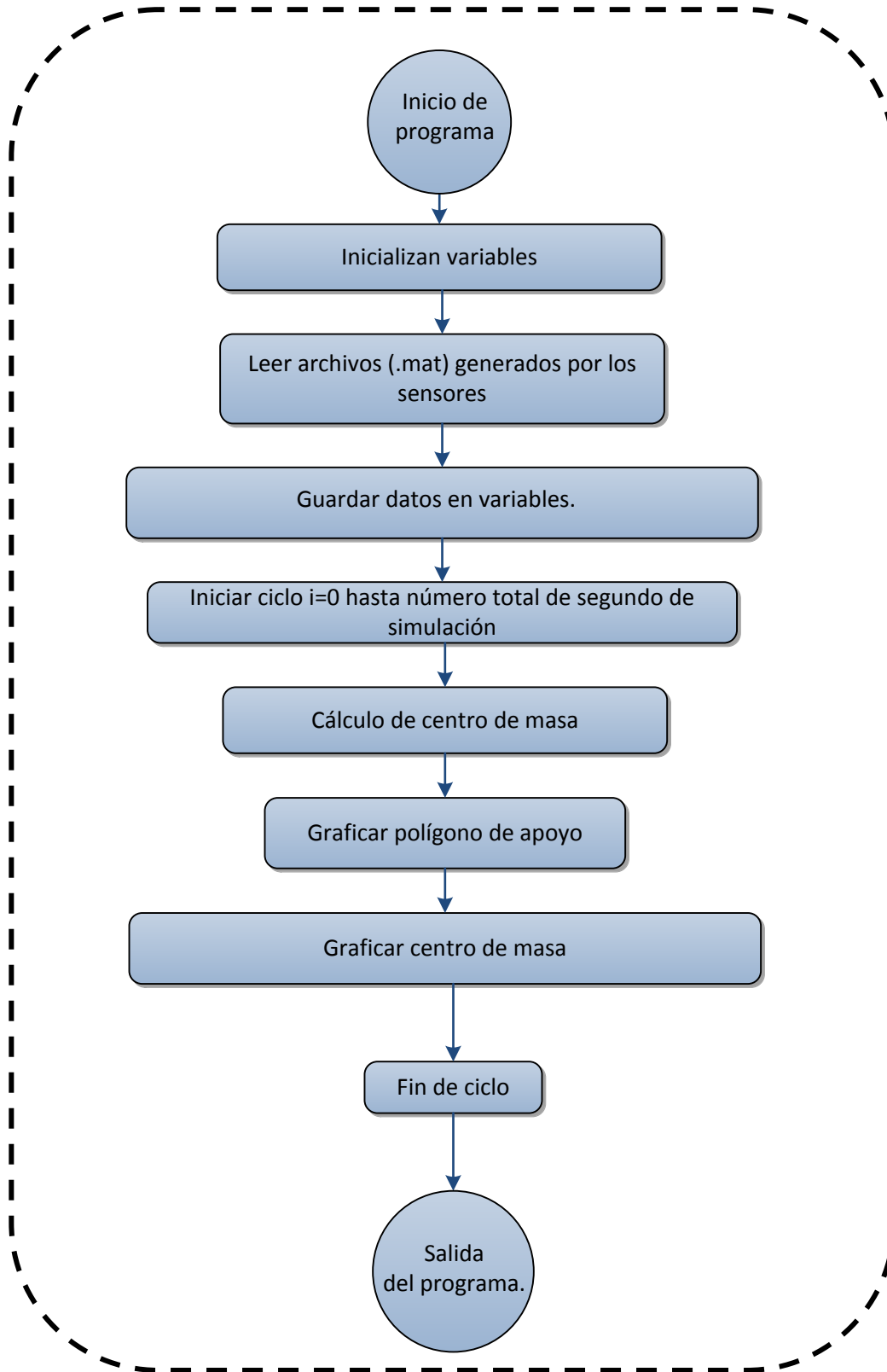
A-1. Función para calcular matriz D-H



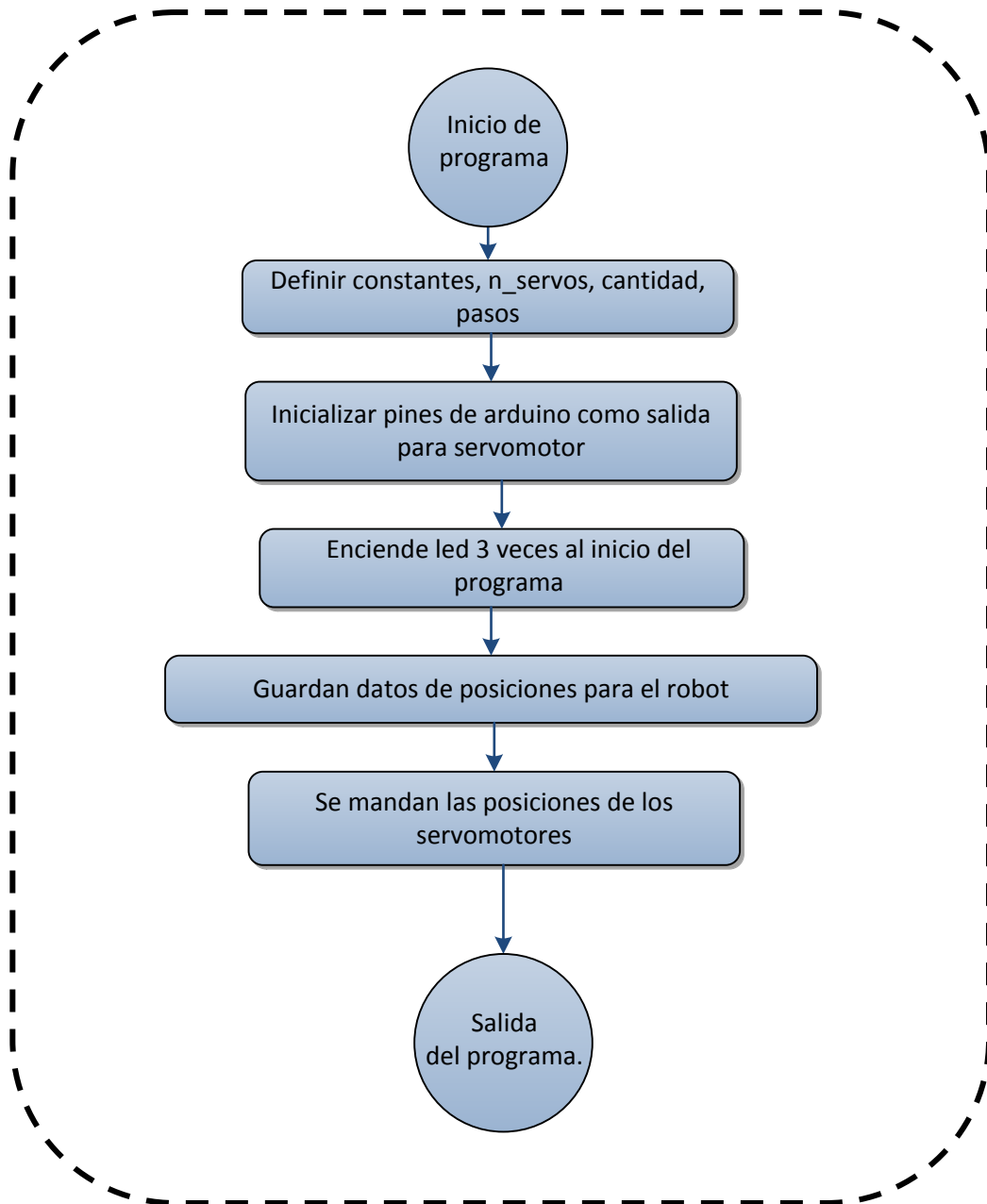
A-2. Programa para calcular los valores de posición de cada eslabón de arquitectura de 21 grados de libertad.



A-3. Programa para importar datos de simulación de robot de CICATA y gráfica centro de masa



A-4. Programa para mover servomotores en Arduino



Referencias

- ABB Copyright 2014** RobotStudio [En línea] // It's just like having the real robot on your PC!. - 2014. - junio de 2104. - <http://new.abb.com/products/robotics/robotstudio>.
- Cyberbotics Ltd, Copyright © 2014** Webots 7 [En línea] // cyberbotics. - 5 de sep de 2014. - 5 de sep de 2014. - <http://www.cyberbotics.com/overview>.
- Cyberbotics Ltd, Copyright © 2014** Webots: the mobile robotics simulation software [En línea] // Webots 7. - 2014. - junio de 2104. - <http://www.cyberbotics.com/>.
- MIT Leg Laboratoryb © Copyright** Spring Flamingo [En línea] // MIT.edu. - junio de 2014. - http://www.ai.mit.edu/projects/leglab/robots/Spring_Flamingo/Spring_Flamingo.html.
- Newtonium Copyright 2000** RoboWorks software [En línea] // HOME. - 05 de 10 de 2008. - junio de 2104. - <http://www.newtonium.com/>.
- Technische Universität München** Institute of Applied Mechanics [En línea] // Humanoid Robot LOLA. - 2009. - junio de 2014. - <http://www.amm.mw.tum.de/en/research/current-projects/humanoid-robots/lola/>.
- Acosta P.** Fisiointegracion's Blogs [En línea] // Notas de clase. - 24 de Marzo de 2012. - <http://fisiointegracion.wordpress.com/>.
- Aldebaran Robotics** A-Robots Nao [En línea]. - Aldebaran, 2014. - agosto de 2014. - <http://www.aldebaran.com/en/humanoid-robot/nao-robot>.
- Allonrobots** Leonardo da Vinci's robots [En línea] // All on robots. - 2013. - junio de 2014. - <http://www.allonrobots.com/leonardo-da-vinci.html>.
- American Honda Motor Co. Inc.** ASIMO The mosst advanced Humanoid Robot [En línea] // History of ASIMO. - 2014. - junio de 2014. - <http://asimo.honda.com/asimo-history/>.
- anatolandia** Planos anatómicos [En línea] // Temas de estudio para la anatomía humana general. - 03 de August de 2014. - Junio de 2014. - <http://anatolandia.blogspot.mx/2013/10/planos-anatomicos.html#.U96LAaOvcvk>.
- Aparicio J.** Simmechanics: Simulating floor interaction/collision [En línea] // Embedded Programmer. - enero de 2013. - junio de 2014. - <http://embeddedprogrammer.blogspot.mx/2013/01/simmechanics-simulating-floor.html>.
- ARDUINO Co.** Arduino home [En línea]. - 2014. - junio de 2014. - <http://www.arduino.cc/en/pmwiki.php?n=>.
- Arduino CO.** Arduino Mega [En línea]. - 2014. - junio de 2014. - <http://arduino.cc/en/pmwiki.php?n=Main/ArduinoBoardMega>.
- Barrientos A. [y otros]** Fundamentos de Robótica [Libro]. - Madrid, Espana : McGraw-Hill, 2007.

Bräunl T. Androids (JJohnny Walker) [En línea] // Robotics & Automation Lab. - The University of Western Australia, 2014. - junio de 2014. - <http://robotics.ee.uwa.edu.au/robotics.html>.

Carmen M. S. Terapia ocupacional Depto. de anatomía e historia, Universidad de Zaragoza [En línea] // CINESIOLOGÍA DE LA MARCHA HUMANA NORMAL. - 18 de julio de 2014. - <http://wzar.unizar.es/acad/cinesio/>.

Chenbo Y., Jie Z. y Haihan X. Walking Gait Planning And Stability Control [En línea] // intechopen. - Nanjing University of Technology,, 2014. - <http://cdn.intechopen.com/pdfs-wm/6248.pdf>.

citizendium Hubo History [En línea]. - citizendium, 22 de octubre de 2013. - junio de 2014. - <http://en.citizendium.org/wiki/HUBO>.

Collins S. [y otros] Efficient Bipedal Robots Based on Passive Dynamic Walkers [Publicación periódica]. - [s.l.] : Science Magazine, 2005. - Vol. 307.

Coppelia Robotics V-REP virtual robot experimentation plataform [En línea] // Home. - septiembre de 2014. - junio de 2104. - <http://www.coppeliarobotics.com/index.html>.

Corporation Solidworks Solidworks office Premium. Conceptos Básicos de Solidworks: Piezas y ensamblajes [Libro]. - 300 baker Avenue Mássachusetts : [s.n.]. - PMT0010-ESP.

Dalakov G. Jacques de Vaucanson [En línea] // history-compute. - agosto de 2014. - junio de 2014. - <http://history-computer.com/Dreamers/Vaucanson.html>.

De Lope J. [y otros] Solving the Inverse Kinematics in Humanoid Robots: A Neural Approach [Publicación periódica]. - [s.l.] : 3, 2003. - 978-3-540-44869-3 : Vol. 6.

Dragos Calin Robotics Simulation Softwares With 3D Modeling and Programming Support [En línea] // .intorobotics. - 2014. - junio de 2014. - <http://www.intorobotics.com/robotics-simulation-softwares-with-3d-modeling-and-programming-support/>.

Edgar E. y Canal C. Algoritmos genéticos modificados para el balance de robots [Publicación periódica]. - México, Guadalajara : [s.n.], 2010.

F. George T. Walking Toy. - St. Joseph, Missouri : United States Patent Office, 1888.

FANUC America Corporation Copyright ROBOGUIDE - FANUC Simulation Software [En línea] // Fanuc. - 2014. - junio de 2014. - <http://robot.fanucamerica.com/products/vision-software/roboguide-simulation-software.aspx>.

Goddard Consulting. Goddard Consulting [En línea] // Building Simulink Models using MATLAB Code. - 2014. - junio de 2014. - <http://www.goddardconsulting.ca/simulink-creating-using-matlab-code.html>.

Gómez E.B. Optimización y modelado del movimiento [Publicación periódica]. - Castilla, Espana : [s.n.], 2012.

Gonzalez-Galvan E. J. Notas sobre teoría de estimacion aplicada a la robotica. - San Luis Potosí : Centro de investigación y estudios de posgrado, Facultad de Ingeniería, 2012.

Hackel M. Humanoid Robots: Human-like Machines [Libro]. - Vienna, Austria : I-tech, 2007. - 978-3-902613-07-3.

Hernández S. Diseño y construcción de prototipo neumático de prótesis de pierna humana [Informe]. - Puebla : Universidad de las Américas Puebla, 2008.

Huang Q. [y otros] Planning walking patterns for a biped robot, [Publicación periódica]. - [s.l.] : IEEE Trans. Rob. Aut., 1998. - 6 : Vol. 17.

Huei-Ming C. Contributors to gait analysis [En línea] // Phases of Gait Cycle. - 14 de enero de 2007. - agosto de 2014. - <http://www.pt.ntu.edu.tw/hmchai/Kinesiology/KINapplication/GaitPhases.htm>.

Humanoid Robotics Institute Copyright History of Humanoid Robot in Waseda University [En línea] // Waseda University Humanoid. - 2014. - junio de 2014. - <http://www.humanoid.waseda.ac.jp/history.html>.

J.Shen Y. Zheng & Gait book for the SD-2 biped robot to climb sloped surface [Publicación periódica]. - [s.l.] : IEEE, 1990. - 1 : Vol. 6.

K. Nikolaos Forward and Inverse Kinematics. - Grecia : Technical University of Crete, Greece, 2012.

K.S. FU R.C. y LEE C.S.G Robótica: control, detección, visión e inteligencia. - Madrid : Mc Graw-Hill, 1998.

kagami Perception-Action Integrated Humanoid Robot : H6 & H7 [En línea]. - julio, de 2001. - junio de 2014. - <http://www.jsk.t.u-tokyo.ac.jp/research/h6/>.

Kajita S. [y otros] Biped Walking Pattern Generation by using Preview Control of Zero-Moment Point [Conferencia]. - Taipei, Taiwan, : IEEE, 2003. - Vol. 14. - 0-7803-7736-2.

Kajita S. [y otros] Biped Walking Stabilization Based [Publicación periódica]. - Taipei, Taiwan : The 2010 IEEE/RSJ International Conference on, 2010. - 22 : Vol. 18. - 978-1-4244-6676-4.

Kajita S. on Linear Inverted Pendulum Tracking Biped Walking Stabilization Based [Conferencia]. - Taipei, Taiwan : IEEE, 2010. - 978-1-4244-6676-4.

Kajita S., Ibaraki y Tani K. Experimental study of biped dynamic walking [Publicación periódica]. - [s.l.] : IEEE , Control Systems,, 1996. - 1 : Vol. 16.

Kajita S., Yamura T. y KOBAYASHI A. Dynamic Walking control of biped robot along a potencial energy conserving orbit [Publicación]. - [s.l.] : IEEE Trans. Robot Automat, 1992. - Vol. 8.

Kawada Industries, INC. Humanoid Robot HRP-2 "Promet" [En línea] // Robotics. - 2014. - junio de 2014. - <http://global.kawada.jp/mechatronics/hrp2.html>.

Kondo-kagaku CO. LTD Kondo [En línea] // Promera vez Kondo. - Copyright © 2013 Kondo Kagaku co.,, 2013. - junio de 2014. - <http://kondo-robot.com/beginners>.

Kuo A. A mechanical analysis of force distribution between redundant, multiple degree of freedom actuators in the human. - Michigan : Mechanical Engineering, University of Michigan.

Mathworks Exportar Simmechanics [En línea]. - 2014. - junio de 2014. - <http://www.mathworks.es/products/simmechanics/features.html>.

Mathworks Jointsensor [En línea] // Matlab. - 2014. - junio de 2014. - <http://www.mathworks.es/es/help/physmod/sm/mech/ref/jointsensor.html>.

Mathworks Signalbuilder [En línea] // Matlab. - 2014. - junio de 2014. - http://www.mathworks.es/es/help/simulink/slref/signalbuilder_cmd.html.

Mathworks Simulink [En línea] // Matlab. - 2014. - junio de 2014. - <http://www.mathworks.es/products/simulink/features.html>.

Mathworks Simulink-working whit signals groups [En línea] // Matlab. - 2014. - junio de 2014. - <http://www.mathworks.es/es/help/simulink/ug/working-with-signal-groups.html>.

Matworks Matlab el lenguaje de calculo técnico [En línea] // Matlab. - 2014. - junio de 2014. - <http://www.mathworks.es/products/matlab/>.

Melo J. L. ERGONOMIA-Y-CIBERNETICA [En línea] // Ergonomía desde el punto de vista de la Higiene y seguridad Industrial. - 2014. - 15 de mayo de 2014. - <http://ergonomia-y-cibernetica.wikispaces.com/file/view/Antropometria%20y%20Ergonomia.pdf/80897043/Antropometria%20y%20Ergonomia.pdf>.

Muhammad A., Andy Park I. y George Lee C. S. Closed-Form Inverse Kinematic Joint Solution for Humanoid Robots [Conferencia]. - Taipei : IEEE, 2010. - 978-1-4244-6674-0 .

Neumann D. A. Kinesiology of the musculoskeletal system: foundations for physical rehabilitation [Libro]. - the University of Michigan : illustrated, 2008.

Nimbro Nimbro-Learning Humanoid Robots [En línea] // Robots. - University of Bonn, 2014. - junio de 2104. - <http://www.nimbro.net/robots.html>.

ohmy news International Interviews [En línea] // Korean Robotics Steps Into the Future. - 2014. - junio de 2014. - http://english.ohmynews.com/articleview/article_view.asp?no=206848&rel_no=1.

Perry J. Gait Analysis normal and Pathological Function [Libro]. - United States of America : SLACK Incorporated, 1992.

Phil A. y Trans. R. Soc. The TUM walking mahines [Informe]. - [s.l.] : doi, 2007. - 10.1098.

Plas F., Éric V. y Blanc Y. La marcha humana: cinesiología dinámica, biomecánica y patomecánica [Libro]. - España : Masson, 1984.

Pratt J. [y otros] Humanoid Robots in Waseda University - Hadalay-2 and WABIAN [Publicación periódica]. - Tsukuba, Japan : Fisrt International Workshop on humanoid and human Friandly, 1998. - 2 : Vol. 20. - 129-143.

Rainone R., Gardner B. y Frost F. History of Gait Analysis [Sección del libro] // Plate, Gait Efficiency Analysis Using Three Axis Force / aut. libro Ryan Rainone Benjamin Gardner, Jonathan Frost. - [s.l.] : Brian J. Savilonis, 2008.

Rauterberg M. History of HCI [En línea] // Key systems, people and ideas. - Technical University Eindhoven (TU/e), 2014. - junio de 2014. - <http://www.idemployee.id.tue.nl/g.w.m.rauterberg/presentations/hci-history/tsld001.htm>.

ROBOTIS BIOLOID [En línea] // BIOLOID KIT. - Korean robot, 2014. - junio de 2014. - http://www.robotis.com/xen/bioloid_en.

Rubio A.A. y Raffo G.V. Manual de Simulink para la asignatura de teoría de sistemas. Escuela superior de Ingenieros [Libro]. - Sevilla : Universidad de Sevilla, 2009.

Shih C.L. [y otros] Trajectory book and physical admissibility for a biped robot during the single support phase [Publicación periódica]. - [s.l.] : IEE Int Conf. Rob and Autom., 1990. - 1642-1652.

Sias F.R.Jr. y Zheng Y.F. How Many Degrees of freedom does a biped need? [Conferencia] // Intelligent robots and systems, IEEE international Workshop on. - Clemson, South California : IEEE, 1990. - Vol. 1.

Stanford Universidad de The Man Who Stopped Time [En línea] // Stanford alumni. - 2014. - agosto de 2014. - https://alumni.stanford.edu/get/page/magazine/article/?article_id=39117.

Tilden M. Robosapien™ [En línea] // WOWWee. - WowWee Group Limited., 2014. - junio de 2014. - <http://www.wowwee.com/en/products/toys/robots/robotics/robosapiens/robosapien>.

Vivero I. M. Estudio del comportamiento dinámico de un vehículo utilizando la herramienta de SimMechanics de Matlab // Tesis de Ingeniería. - Madrid : Universidad Carlos III de Madrid, 2009.