

ESCUELA POLITÉCNICA NACIONAL

FACULTAD DE INGENIERÍA MECÁNICA

MODELACIÓN Y ANÁLISIS DE LA CINEMÁTICA DIRECTA E INVERSA DEL MANIPULADOR STANFORD DE SEIS GRADOS DE LIBERTAD

PROYECTO PREVIO A LA OBTENCIÓN DEL TÍTULO DE INGENIERO MECÁNICO

MARÍA VICTORIA GRANJA ORAMAS

victoria.granja@hotmail.com

DIRECTOR:

ING. MARIO GERMÁN GRANJA RAMÍREZ MSc.

mario.granja@epn.edu.ec

CO-DIRECTOR:

ING. ÁLVARO GONZALO AGUINAGA BARRAGÁN PhD.

alvaro.aguinaga@epn.edu.ec

Quito, octubre 2014

DECLARACIÓN

Yo María Victoria Granja Oramas, declaro bajo juramento que el trabajo aquí descrito es de mi autoría; que no ha sido previamente presentada para ningún grado o calificación profesional; y, que he consultado las referencias bibliográficas que se incluyen en este documento.

A través de la presente declaración cedo mis derechos de propiedad intelectual correspondientes a este trabajo, a la Escuela Politécnica Nacional, según lo establecido por la Ley de Propiedad Intelectual, por su Reglamento y por la normatividad institucional vigente.

Ma. Victoria Granja O.

CERTIFICACIÓN

Certifico que el presente trabajo fue desarrollado por María Victoria Granja Oramas bajo mi supervisión.

MARIO GERMÁN GRANJA RAMIREZ
DIRECTOR DEL PROYECTO

AGRADECIMIENTO

A mis padres, por su amor y entrega incondicional en cada uno de mis pasos.

A mi padre, Ing. Mario Granja Ramírez MSc., por su brillante ejemplo e invaluable guía en mi desarrollo profesional y humano.

Al Ing. Álvaro Aguinaga PhD. por su apoyo para la consecución de este objetivo y a todos mis profesores, por su aportación en mi formación académica.

DEDICATORIA

A mis padres,
luz y guía de mi camino

ÍNDICE GENERAL

1	CAPÍTULO I	1
1.1	ROBÓTICA.....	1
1.2	ROBOT	1
1.2.1	COMPONENTES Y ESTRUCTURA DE ROBOTS	4
1.2.2	REPRESENTACIÓN SIMBÓLICA DE ROBOTS.....	5
1.2.3	GRADOS DE LIBERTAD	7
1.2.4	ESPACIO DE TRABAJO	9
1.2.5	CLASIFICACIÓN DE LOS ROBOTS	10
1.2.6	ARREGLOS CINEMÁTICOS COMUNES	12
1.3	ROBOT STANFORD.....	17
1.3.1	ESTRUCTURA DEL ROBOT STANFORD	18
2	CAPÍTULO II	19
2.1	REPRESENTACIÓN DE LA POSICIÓN (TRASLACIÓN).....	19
2.1.1	EN EL PLANO.....	19
2.1.2	EN EL ESPACIO	20
2.2	REPRESENTACIÓN DE LA ORIENTACIÓN (ROTACIÓN).....	22
2.2.1	EN EL PLANO.....	22
2.2.2	EN EL ESPACIO	25
2.3	REPRESENTACIÓN DE MOVIMIENTOS COMPUESTOS.....	28
2.3.1	POSTMULTIPLICACION MATRICIAL	28
2.3.2	PREMULTIPLICACION MATRICIAL	29
3	CAPÍTULO III	31
3.1	CINEMÁTICA DIRECTA	32
3.1.1	ASIGNACIÓN DE LOS SISTEMAS DE COORDENADAS	33
3.2	CINEMÁTICA INVERSA	39
4	CAPÍTULO IV	43
4.1	MÉTODO DE DENAVIT-HARTENBERG ESTÁNDAR, DHS.....	44
4.2	MÉTODO DE DENAVIT-HARTENBERG MODIFICADO, DHM.....	53

4.3	MÉTODO DEL MOVIMIENTO GENERAL CONTINUO	60
5	CAPÍTULO V	67
5.1	DESACOPLO CINEMÁTICO. SOLUCIÓN DE PIEPER.....	67
5.2	CINEMÁTICA INVERSA DEL ROBOT STANFORD	70
6	CAPÍTULO VI	78
6.1	ANÁLISIS DE LOS RESULTADOS DE LA CINEMÁTICA DIRECTA	78
6.1.1	MATRIZ H DEL ROBOT STANFORD UTILIZANDO EL MÉTODO DE DENAVIT-HARTENBERG ESTÁNDAR	79
6.1.2	MATRIZ H DEL ROBOT STANFORD UTILIZANDO EL MÉTODO DE DENAVIT-HARTENBERG MODIFICADO	81
6.1.3	MATRIZ H DEL ROBOT STANFORD UTILIZANDO EL MÉTODO DEL MOVIMIENTO GENERAL, SIN CONSIDERAR MOVIMIENTOS EN EL EJE Y.....	83
6.1.4	MATRIZ H DEL ROBOT STANFORD UTILIZANDO EL MÉTODO DEL MOVIMIENTO GENERAL, INCLUYENDO MOVIMIENTOS EN EL EJE Y	85
6.2	ANÁLISIS DE LOS RESULTADOS DE LA CINEMÁTICA INVERSA	87
7	CAPÍTULO VII	89
7.1	TOOLBOX ROBOTICS DE MATLAB	89
7.1.1	PASOS PARA LA CREACIÓN DE UN ROBOT UTILIZANDO EL TOOLBOX ROBOTICS DE MATLAB	90
7.2	SIMULACIÓN DEL ROBOT STANFORD	91
7.2.1	MÉTODO DE DENAVIT-HARTENBERG ESTÁNDAR	91
7.2.2	MÉTODO DE DENAVIT-HARTENBERG MODIFICADO	94
8	CAPITULO VIII	97
8.1	CONCLUSIONES	97
8.2	RECOMENDACIONES	100
9	BIBLIOGRAFIA.....	102

ÍNDICE DE FIGURAS

Figura 1.1 Robot manipulador Kuka 210.....	3
Figura 1.2 Componentes de un Robot Industrial	4
Figura 1.3 Representación simbólica de juntas de robots	5
Figura 1.4 Representación de juntas típicas de robots manipuladores	6
Figura 1.5 Estructuras mecánicas frecuentes en robots industriales	7
Figura 1.6 Comparación robot manipulador con cuerpo humano	8
Figura 1.7 Muñeca de Robot Manipulador.....	9
Figura 1.8 Espacio de trabajo de robots utilizados frecuentemente.....	10
Figura 1.9 Arreglos cinemáticos comunes de robots manipuladores.....	12
Figura 1.10 Robot cartesiano.....	13
Figura 1.11 Robot cilíndrico SEIKO RT3300	13
Figura 1.12 Robot Esférico EverRobot.....	14
Figura 1.13 Robot angular ABB 1400.....	15
Figura 1.14 Robot Scara OMRON XG	15
Figura 1.15 Robot Stanford -1969.....	17
Figura 1.16 Estructura del Robot Stanford Fuente: Propia.....	18
Figura 2.1 Traslación en el plano	19
Figura 2.2 Traslación en el espacio	21
Figura 2.3 Rotación en el plano.....	23
Figura 2.4 x_1, y_1 en función de x_0, y_0 Fuente: Propia	24
Figura 2.5 Rotación en el espacio Fuente: Propia	25
Figura 2.6 Rotación del sistema o, u, v, w alrededor de x, α , de y, ϕ , de z, θ	27
Figura 3.1 Cinemática directa e inversa	31
Figura 3.2 Estructura de una matriz homogénea.....	33
Figura 3.3 Sistema de coordenadas Denavit-Hartenberg estándar y modificado [17].	35
Figura 3.4 Asignación del sistema de coordenadas de la herramienta.	37
Figura 3.5 Problema cinemático inverso. a) Solución múltiple, b) codo abajo.....	40
Figura 4.1 Representación simbólica del Robot Stanford Fuente: Propia	43
Figura 4.2 Asignación de sistemas de coordenadas por el método Denavit-Hartenberg estándar Fuente: Propia	43
Figura 4.3 Programación en Matlab para Modelación del Robot Stanford, D-H estándar	51
Figura 4.4 Matrices de transformación A_i obtenidas con el software Matlab	51
Figura 4.5 Asignación de sistemas de coordenadas por el método Denavit-Hartenberg modificado Fuente: Propia	54
Figura 4.6 Programación en Matlab para Modelación del Robot Stanford, D-H modificado.....	58
Figura 4.7 Matrices de transformación A_i obtenidas con el software Matlab.....	59

Figura 4.8 Asignación de sistemas de coordenadas por el método del movimiento general continuo considerando el convenio de D-H (sin utilizar movimientos en el eje y) Fuente: Propia	61
Figura 4.9 Programación en Matlab para Modelación del Robot Stanford mediante MGc	63
Figura 4.10 Programación en Matlab para Modelación del Robot Stanford mediante MGc, con movimientos en cualquier eje	64
Figura 4.11 Asignación de sistemas de coordenadas por el método del movimiento general continuo (movimiento libre) Fuente: Propia	65
Figura 5.1 Robot Stanford, desacoplo cinemático Fuente: Propia	68
Figura 6.1 Comprobación del modelo obtenido con el método de D-H estándar.....	80
Figura 6.2 Comprobación del modelo obtenido con el método de D-H modificado.....	82
Figura 6.3 Comprobación del modelo obtenido con el método de MG sin movimientos en el eje "y".....	84
Figura 6.4 Comprobación del modelo obtenido con el método de MG con movimientos en el eje "y".....	86
Figura 7.1 Programación para Simulación robot Stanford estándar	92
Figura 7.2 Simulación Robot Stanford D-H estándar	93
Figura 7.3 Comando para variación de parámetros articulares del Robot Stanford D-H estándar	94
Figura 7.4 Programación para Simulación robot Stanford modificado.....	95
Figura 7.5 Simulación Robot Stanford D-H modificado.....	96
Figura 7.6 Comando para variación de parámetros articulares del Robot Stanford D-H modificado	96

ÍNDICE DE TABLAS

Tabla 1.1 Estructura cinemática y volumen de trabajo de las configuraciones típicas de robots manipuladores	16
Tabla 4.1 Parámetros D-H estándar para manipulador Stanford.	47
Tabla 4.2 Parámetros D-H modificado para manipulador Stanford.....	55
Tabla 4.3 Parámetros del Movimiento General Continuo para Robot Stanford.....	62
Tabla 4.4 Parámetros del Movimiento General Continuo para Robot Stanford.....	64
Tabla 5.1 Parámetros D-H estándar para manipulador Stanford.	71
Tabla 7.1 Parámetros D-H estándar para simulación manipulador Stanford.....	92
Tabla 7.2 Parámetros D-H modificado para simulación del manipulador Stanford.....	94

RESUMEN

El presente proyecto de titulación busca obtener la modelación de la cinemática directa e inversa del manipulador Stanford de seis grados de libertad. Además busca resolver algunas dificultades que se tiene en la modelación de los robots manipuladores, pues existe desconocimiento del concepto fundamental del método matricial utilizado por Denavit Hartenberg. Actualmente existen variantes de este método, tanto el método estándar planteado por Denavit & Hartenberg, 1955, así como el método D&H modificado, formulado por Craig, 1986; estos dos métodos frecuentemente no son presentados en un mismo libro por un mismo autor, lo cual dificulta al lector cuando se tiene que estudiar los dos métodos, originando conflictos especialmente al asignar los sistemas de coordenadas a los eslabones o a las articulaciones, así como también en la elaboración de las tablas de parámetros de Denavit Hartenberg. Además se realizará la modelación del robot Stanford utilizando el método del Movimiento General continuo, planteado por el Ing. Mario Granja, 2014.

En la modelación de robots manipuladores es indispensable delinear un modelo matemático robusto del mismo, por lo que cada uno de los modelos obtenidos serán debidamente validados.

Por otro lado se resolverá el problema de la cinemática inversa en el robot manipulador Stanford, empleando el método de Pieper, esto servirá como referencia para la resolución de este problema en otros tipos de robots manipuladores.

Nos ayudaremos del software desarrollado por Peter Corke, ToolBox Robotics de Matlab, para la simulación del Robot Stanford utilizando el método de Denavit-Hartenberg estándar y modificado.

PRESENTACIÓN

La presente investigación está estructurada en ocho capítulos que se sintetizan a continuación:

En el capítulo 1 se realiza una introducción a la robótica y a los robots manipuladores, especialmente al robot Stanford de seis grados de libertad.

En el capítulo 2, se estudia la representación matricial de la traslación y rotación de un sistema de coordenadas x_1, y_1, z_1 respecto a un sistema de coordenadas x_0, y_0, z_0 tanto en el plano como en el espacio. Además se estudia la representación matricial cuando existen movimientos compuestos.

En el capítulo 3, se presenta una introducción al problema de la cinemática directa, así como también al problema de la cinemática inversa. Se estudia además los métodos existentes para la resolución de cada uno de los mismos.

El capítulo 4 abarca la modelación del Robot manipulador Stanford, abordando el problema de la cinemática directa por tres métodos: Denavit-Hartenberg estándar, Denavit-Hartenberg modificado y Movimiento General continuo.

En el capítulo 5 se presenta la resolución del problema de la cinemática inversa del Robot manipulador Stanford utilizando el método de Pieper.

En el capítulo 6, se presenta el análisis de los resultados obtenidos en los capítulos 4 y 5. Se validan los modelos matemáticos obtenidos para el Robot Stanford y se analizan las semejanzas y diferencias que presentan los diferentes métodos de modelación.

En el capítulo 7, se simula el robot manipulador Stanford utilizando los métodos de Denavit-Hartenberg estándar y modificado por medio del Toolbox Robotics de Matlab.

En el capítulo 8, se presentan las conclusiones y recomendaciones, del presente proyecto de titulación.

1 CAPÍTULO I

INTRODUCCIÓN AL ROBOT STANFORD

1.1 ROBÓTICA

La robótica es un campo relativamente nuevo de la tecnología moderna que traspasa los límites de la ingeniería tradicional, pues para lograr comprender los robots y sus aplicaciones en toda su complejidad son necesarios conocimientos de ingeniería eléctrica, ingeniería mecánica, ingeniería industrial, ciencia computacional, economía y matemática.

Entonces, se puede definir a la robótica como el conjunto de disciplinas que convergen hacia el objetivo de cumplir las aspiraciones de suministrar al hombre un mecanismo que lo libere de actividades tediosas y/o peligrosas y que, como su nombre lo indica (*“Robotnik”*, del Checo siervo; *“Robota”*, del ruso trabajo) estén a su servicio con un buen grado de automatización e independencia.¹

La robótica como ciencia ha tenido un enorme crecimiento en los últimos años, propulsado por los rápidos avances en tecnología computacional y sensorial, así como también avances en la teoría de control y visión artificial.

1.2 ROBOT

La definición de Robot más comúnmente aceptada es la de la Asociación de Industrias de Robótica RIA (Robotics Industry Association): “Un robot, es un manipulador multifuncional reprogramable diseñado para mover material, elementos, herramientas ó dispositivos especializados, por medio de movimientos programados para la ejecución de diferentes tareas.”

¹ ASIMOV, Isaac. “Los robots”, España: Ediciones Orbis S.A.,1986. Pág. 160

Otra definición conocida es de la Asociación japonesa de robots industriales JIRA (Japan Industrial Robot Association): “Todo mecanismo permitiendo efectuar enteramente o por parte, una tarea generalmente realizada por un hombre”.

Según la Organización Internacional de Estándares (ISO) se define un robot industrial como: “Manipulador multifuncional reprogramable con varios grados de libertad, capaz de manipular materias, piezas, herramientas o dispositivos especiales según trayectorias variables programadas para realizar tareas diversas.”

Se incluye en esta definición la necesidad de que el robot tenga varios grados de libertad. Una definición más completa es la establecida por la Asociación Francesa de Normalización (AFNOR), que define primero el manipulador y, basándose en dicha definición, el robot:

Manipulador: mecanismo formado generalmente por elementos en serie, articulados entre sí, destinado al agarre y desplazamiento de objetos. Es multifuncional y puede ser gobernado directamente por un operador humano o mediante dispositivo lógico.

Robot: manipulador automático servo-controlado, reprogramable, polivalente, capaz de posicionar y orientar piezas, útiles o dispositivos especiales, siguiendo trayectoria variables reprogramables, para la ejecución de tareas variadas. Normalmente tiene la forma de uno o varios brazos terminados en una muñeca. Su unidad de control incluye un dispositivo de memoria y ocasionalmente de percepción del entorno. Normalmente su uso es el de realizar una tarea de manera cíclica, pudiéndose adaptar a otra sin cambios permanentes en su material.

Por último, la Federación Internacional de Robótica (IFR, *International Federation of Robotics*) distingue entre robot industrial de manipulación y otros robots:

"Por robot industrial de manipulación se entiende una máquina de manipulación automática, reprogramable y multifuncional con tres o más ejes que pueden posicionar y orientar materias, piezas, herramientas o dispositivos especiales para la ejecución de

trabajos diversos en las diferentes etapas de la producción industrial, ya sea en una posición fija o en movimiento"

En esta definición se debe entender que la reprogramabilidad y la multifunción se consiguen sin modificaciones físicas del robot.

Común en todas las definiciones anteriores es la aceptación del robot industrial como un brazo mecánico con capacidad de manipulación y que incorpora un control más o menos complejo. Un sistema robotizado, en cambio, es un concepto más amplio. Engloba todos aquellos dispositivos que realizan tareas de forma automática en sustitución de un ser humano y que pueden incorporar o no a uno o varios robots, siendo esto último lo más frecuente.

Actualmente existen diversos tipos de robots; sin embargo, la vasta mayoría de robots aplicados en la industria son los robots manipuladores. De hecho, se puede decir que un robot industrial es esencialmente un robot manipulador. Este tipo de robot, es básicamente un brazo mecánico que opera bajo control computacional, un ejemplo es el robot Kuka 210, ver figura 1.1.



Figura 1.1 Robot manipulador Kuka 210²

² http://www.kuka-robotics.com/en/products/industrial_robots/high/ultra/kr210_r3100_ultra/

Los robots industriales se aplican para:

- Tareas peligrosas para obreros humanos
- Tareas en lugares difícilmente accesibles, con riesgo de accidentes o con condiciones peligrosas para la salud
- Manipulación de objetos con tamaño y/o forma haciendo difícil una manipulación manual
- Tareas que requieren precisión y repetitividad

1.2.1 COMPONENTES Y ESTRUCTURA DE ROBOTS

Los robots industriales consisten de cuatro subsistemas: la unidad mecánica, el sistema de potencia, el sistema de control y las herramientas, como se puede observar en a figura 1.2

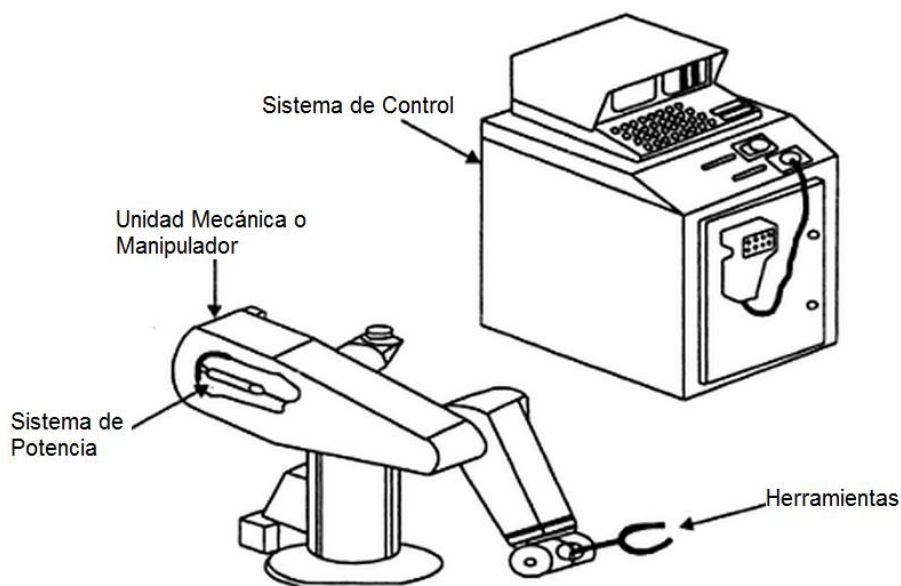


Figura 1.2 Componentes de un Robot Industrial³

Unidad Mecánica: Se refiere al brazo y base del robot manipulador. Consiste de la estructura mecánica en la que se encuentran articulaciones, guías, actuadores, válvulas de control y sensores. Las dimensiones físicas, el diseño y la capacidad de carga dependen de los requerimientos de la aplicación.

³ GUPTA A., ARORA S., Industrial Automation and Robotics, 1 Edition, University Science Press, New Delhi, 2007

Sistema de potencia: tiene como misión proveer la potencia necesaria para mover el manipulador o unidad mecánica. Este puede ser: eléctrico, neumático ó hidráulico.

Sistema de Control: Tiene tres funciones, en primer lugar dirige al sistema de potencia para que mueva al manipulador en una forma predeterminada. En segundo lugar, el sistema de control almacena uno o varios programas, así como la información recogida durante el proceso mismo del programa. En tercer lugar cuenta con diversos sistemas que permiten la comunicación, ingreso y salida de datos, en forma de teclados, pantallas, medios magnéticos.

Herramientas: Son los dispositivos que se instalan en el extremo del manipulador, los cuales interactúan con los objetos del entorno al efectuar una tarea. Las herramientas son la interface entre el manipulador y muñeca y la pieza de trabajo.

1.2.2 REPRESENTACIÓN SIMBÓLICA DE ROBOTS

Los robots manipuladores están compuestos de eslabones unidos por articulaciones formando una cadena cinemática. Las articulaciones por lo general son rotativas o prismáticas. La representación gráfica que utilizaremos para estos dos tipos de articulaciones se muestra en la figura 1.3

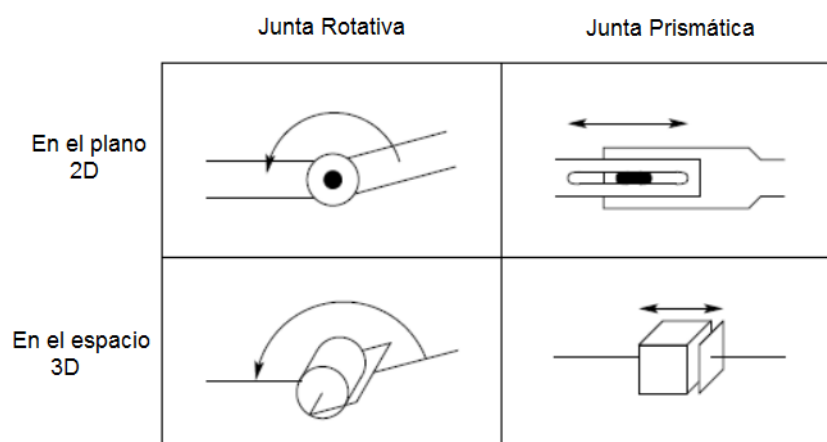


Figura 1.3 Representación simbólica de juntas de robots⁴

⁴ Spong, M. Hutchinson, S. Vidyasagar, M. (2004). Robot Dynamics and Control. USA: ED. John Wiley & Sons, Inc

Los dos tipos de articulación suministran un grado de libertad, la rotativa consiste en una rotación alrededor del eje de la articulación (rotación pura) y la prismática en una traslación a lo largo del eje de la articulación (deslizamiento puro). Se utiliza la convención (R) para representar juntas rotativas y (P) para juntas prismáticas.

En la figura 1.4 se muestran las dos juntas típicas que se utilizan en los robots, las juntas restringen los grados de libertad en los mecanismos. Como se conoce, un eslabón en el espacio tiene seis grados de libertad, pero al conectarse dos eslabones a través de una junta sea rotativa o prismática se pierden cinco de los grados de libertad del eslabón y se pasa a tener solamente un grado de libertad.

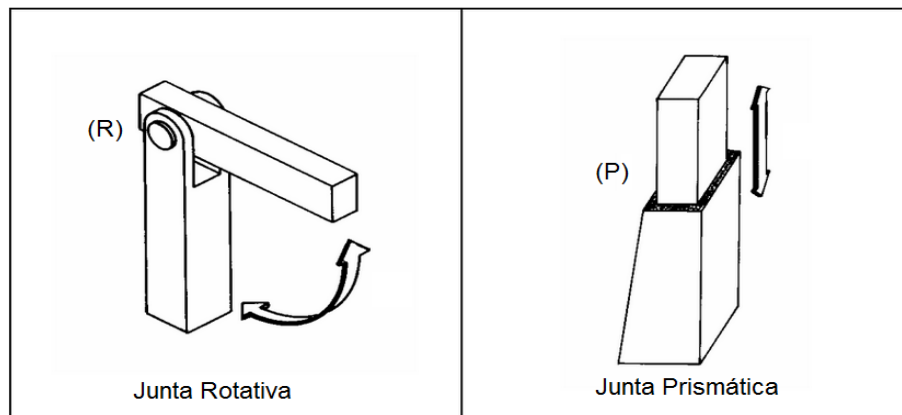


Figura 1.4 Representación de juntas típicas de robots manipuladores

Cada junta representa la interconexión entre dos eslabones, sean l_i y l_{i+1} . Denotamos el eje de rotación de una junta rotativa, o el eje a lo largo del cual la junta prismática se desliza como z_i si la junta es la interconexión de los eslabones i e $i + 1$. Las variables de las articulaciones que representan el movimiento relativo entre dos eslabones contiguos son denotadas por θ_i para una junta rotativa y por d_i para una junta prismática.

1.2.3 GRADOS DE LIBERTAD

Los grados de libertad de un robot manipulador vienen dados por la suma de los grados de libertad de las articulaciones que lo componen. Puesto que, como se ha indicado, las articulaciones empleadas básicamente son las rotativas y las prismáticas que poseen un sólo GDL, el número de GDL del robot suele coincidir con el número de articulaciones de que se compone. El empleo de diferentes combinaciones de articulaciones en el diseño y construcción de un robot, da lugar a diferentes configuraciones para diversas aplicaciones. Las combinaciones de articulaciones más frecuentes son las representadas en la figura 1.5.

Generalmente, un robot manipulador debe poseer al menos seis grados de libertad independientes: tres para posicionamiento y tres para orientación. Seis grados de libertad determinan unívocamente la posición y orientación de un cuerpo en el espacio, éstos pueden ser traslaciones a lo largo de los ejes x,y,z y rotaciones en torno a los mismos.

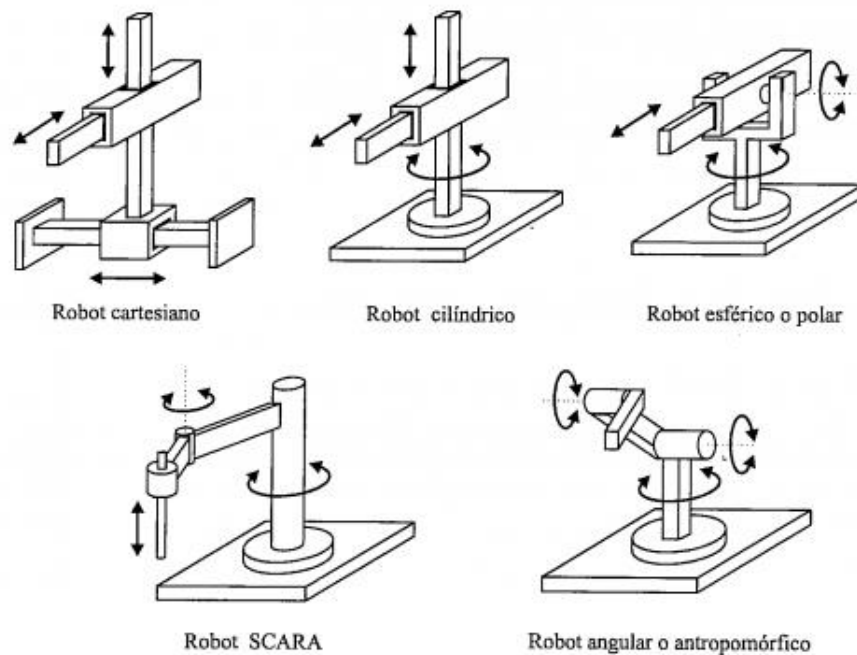


Figura 1.5 Estructuras mecánicas frecuentes en robots industriales⁵

⁵ <http://creandoelfuturo.net/es/morfologia-del-robot/estructura-mecanica-robot>

1.2.3.1 Comparación entre un brazo robot y un brazo humano

Las partes de un robot manipulador son nombradas en base a partes similares del cuerpo humano por su similitud en estructura y función. En la figura 1.6 podemos observar la comparación directa de las partes del robot manipulador con las de cuerpo humano, y podemos observar que poseen nombres en común.

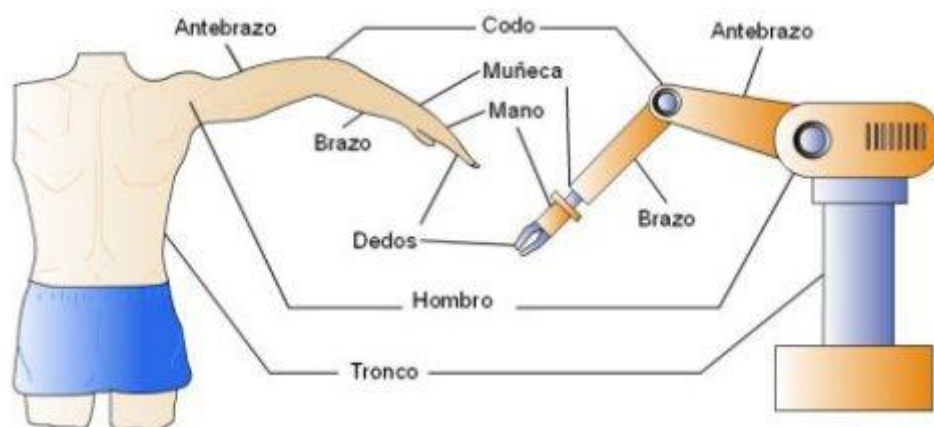


Figura 1.6 Comparación robot manipulador con cuerpo humano ⁶

1.2.3.2 Muñeca del Robot

Como se mencionó anteriormente, son necesarios 6 grados de libertad para que el robot manipulador logre alcanzar completamente la ubicación y orientación de un objeto. Generalmente los tres primeros grados de libertad son proporcionados por el brazo del robot y los otros 3 son obtenidos por la adición de una muñeca en la que se coloca la herramienta. La muñeca presenta tres movimientos que son: pitch, yaw y roll como podemos observar en la figura 1.7

⁶ <http://roboticajh.wordpress.com/>

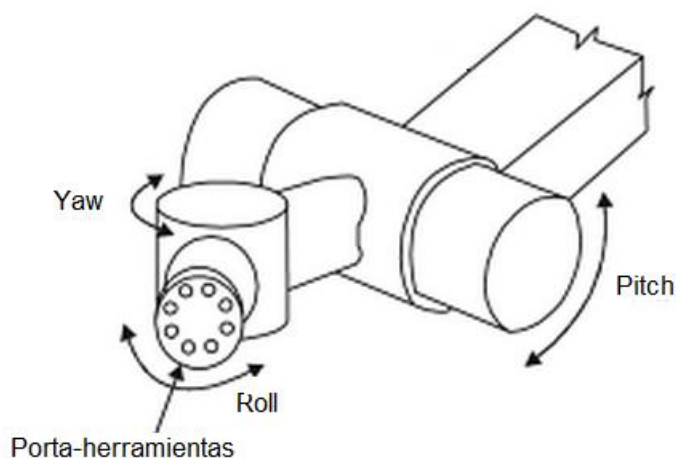


Figura 1.7 Muñeca de Robot Manipulador⁷

Donde,

Pitch: Movimiento de rotación en el plano vertical

Yaw: Movimiento de rotación en el plano horizontal

Roll: Movimiento giratorio

1.2.4 ESPACIO DE TRABAJO

El espacio de trabajo de un manipulador es el volumen total barrido por el efector final mientras el manipulador ejecuta todos los movimientos posibles. El espacio de trabajo está limitado por la geometría del manipulador, así como por las limitaciones mecánicas de las articulaciones.

Un punto del espacio se dice totalmente accesible si el efector final puede situarse en él en todas las orientaciones que permita la constitución del manipulador y se dice parcialmente accesible si es accesible por el efector final pero no en todas las orientaciones posibles. En la figura 1.8 se aprecia el volumen de trabajo de robots de distintas configuraciones.

⁷ GUPTA A., ARORA S., Industrial Automation and Robotics, Second Edition, University Science Press, New Delhi, 2011

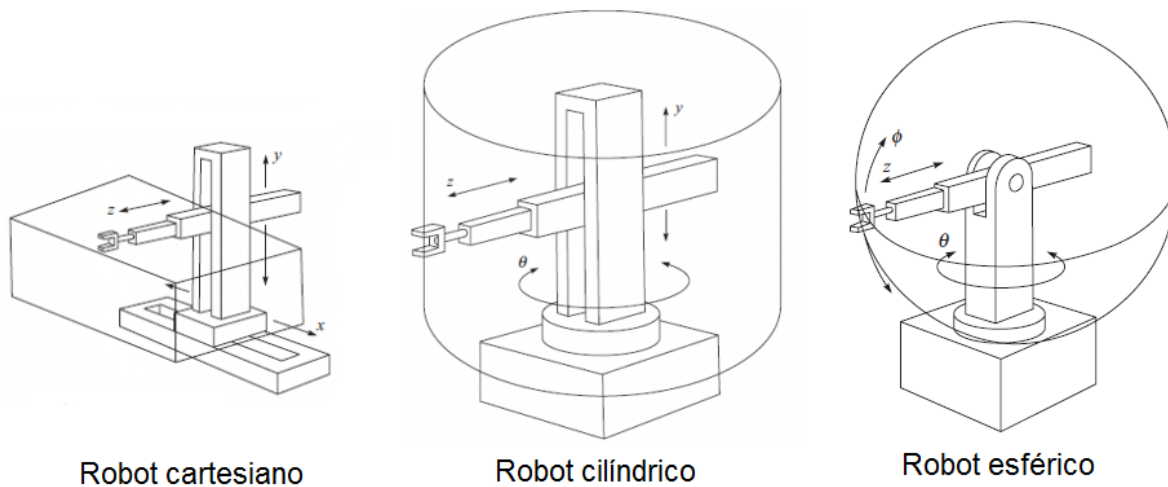


Figura 1.8 Espacio de trabajo de robots utilizados frecuentemente

1.2.5 CLASIFICACIÓN DE LOS ROBOTS

Los robots manipuladores pueden ser clasificados por diversos criterios, como su **fuentes de potencia** o la forma en la que las articulaciones son activadas, su **geometría** o estructura cinemática, su **área de aplicación** o por su **método de control**.

1.2.5.1 Por su fuente de potencia

Generalmente los robots son activados eléctrica, hidráulica o neumáticamente.

Los sistemas hidráulicos son inigualables en su velocidad de respuesta y en su capacidad de producir torque. Por lo que, los robots hidráulicos son principalmente utilizados para levantar carga pesada. La desventaja de los robots hidráulicos es que tienden a tener fugas de fluido hidráulico, requieren de más equipamiento como bombas, que también requieren mantenimiento y son ruidosas.

Los robots impulsados por servomotores DC- o AC- son cada vez más populares debido a que son más baratos, más limpios y menos ruidosos.

Los robots neumáticos son simples y económicos pero no pueden ser controlados con precisión. Como resultado, los robots neumáticos están limitados en su rango de aplicaciones y popularidad.

1.2.5.2 Por su área de aplicación

El área proyectada más grande para la aplicación de los robots a futuro, es en ensamblaje. Por lo tanto, los robots son comúnmente clasificados en robots de ensamblaje y robots de no-ensamblaje. Los robots de ensamblaje tienden a ser pequeños, potenciados eléctricamente y tanto de revolución o SCARA en su diseño. Las principales aplicaciones de no-ensamblaje hasta la fecha han sido en soldadura, pintura por spray, manejo de material, y carga y descarga de máquinas.

1.2.5.3 Por su método de Control

Los robots son clasificados por el método de control en servo y no-servo robots. Los primeros robots fueron no-servo robots. Estos robots son básicamente dispositivos de bucle abierto cuyo movimiento está limitado por paradas mecánicas predeterminadas, y son de utilidad principalmente para transferir material. De hecho, según la definición dada previamente, los robots con paradas predeterminadas difícilmente se califican como robots. Los servo robots utilizan control computacional de bucle cerrado para determinar su movimiento, por lo que son capaces de ser dispositivos reprogramables multifuncionales.

Los servo-robots son además clasificados de acuerdo al método que el controlador utiliza para guiar al efector final. El tipo de robot más simple es el robot punto a punto. A este tipo de robot se le asignan un conjunto de puntos discretos pero no existe un control de la trayectoria del efector final entre los puntos asignados. Los robots punto a punto están severamente limitados en su rango de aplicaciones. Por otra parte, en robots de trayectoria continua, la totalidad de la trayectoria del efector final puede ser controlada. Por ejemplo, el efector final puede seguir una línea recta entre dos puntos o incluso seguir un contorno como una costura de soldadura. Adicionalmente, la

velocidad y/o la aceleración del efector final puede ser controlada. Estos son los robots más avanzados y requieren del más sofisticado control computacional y desarrollo de software.

1.2.5.4 Por su geometría

La mayoría de manipuladores industriales en la actualidad tienen seis o menos grados de libertad. Estos manipuladores son usualmente clasificados cinemáticamente en base a las articulaciones del brazo, siendo de muñeca descrita por separado. La mayoría de estos manipuladores caen dentro de uno de los cinco tipos geométricos: articulares, esféricos, SCARA, cilíndricos y cartesianos.

1.2.6 ARREGLOS CINEMÁTICOS COMUNES

Se consideran, las estructuras más utilizadas como brazo de un robot manipulador. Estas estructuras tienen diferentes propiedades en cuanto a espacio de trabajo y accesibilidad a posiciones determinadas. En la figura 1.9 se muestran cuatro configuraciones básicas.

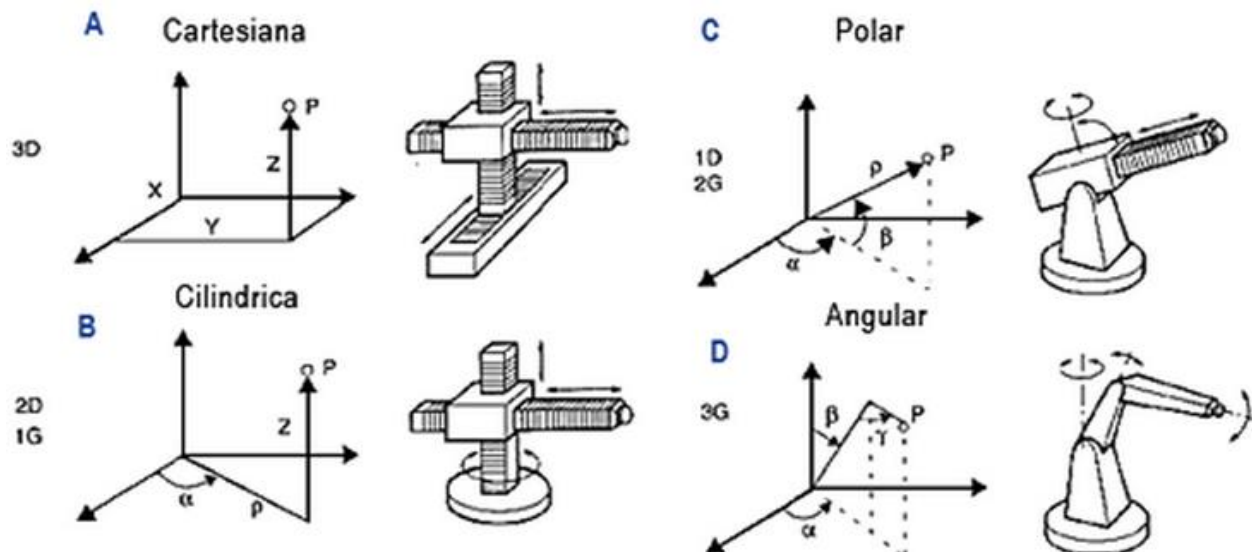


Figura 1.9 Arreglos cinemáticos comunes de robots manipuladores⁸

⁸ http://www.uvmnet.edu/investigacion/episteme/numero6-06/reportes/a_control.asp

1.2.6.1 Configuración cartesiana

Las tres primeras articulaciones son prismáticas (3D ó PPP). Esta configuración es usada frecuentemente en estructuras industriales, como pórticos, y para el transporte de cargas voluminosas. La especificación de la posición de un punto se efectúa mediante las coordenadas cartesianas (x,y,z). Presenta accesibilidad reducida, su volumen de trabajo es cúbico. Un ejemplo de robot cartesiano se muestra en la figura 1.10.

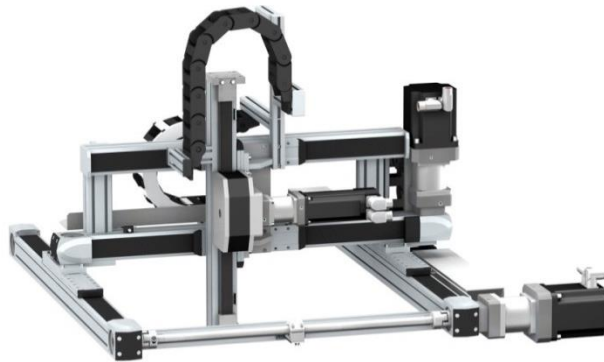


Figura 1.10 Robot cartesiano

1.2.6.2 Configuración cilíndrica

Presenta una articulación de rotación y 2 articulaciones prismáticas (2D,1G ó RPP). La posición se especifica en coordenadas cilíndricas. El volumen de trabajo suponiendo un radio de giro de 360° y un rango de desplazamiento L , es el de un toro de sección cuadrada de radio interior L y radio exterior $2L$. El volumen de trabajo total sería $3\pi L^3$. El Robot SEIKO RT3300 es un ejemplo de robot de configuración cilíndrica, figura 1.11



Figura 1.11 Robot cilíndrico SEIKO RT3300

1.2.6.3 Configuración polar o esférica

Presenta dos articulaciones de rotación y una prismática (2G, 1D o estructura RRP). Las variables articulares expresan la posición del extremo del tercer enlace en coordenadas polares. Esta configuración permite un buen volumen de trabajo, sólo inferior a la angular. El volumen de trabajo de esta estructura, suponiendo un radio de giro de 360° y un rango de desplazamiento de L , es el que existe en una esfera de radio $2L$ y una concéntrica de radio L . Por lo que, el volumen es $(28/3) \pi L^3$. El Robot de la figura 1.12 es un ejemplo de robot esférico.



Figura 1.12 Robot Esférico EverRobot

1.2.6.4 Configuración angular (universal o antropomorfa)

Presenta tres articulaciones de rotación (3G o RRR). La posición del extremo final se especifica en coordenadas angulares. La estructura tiene un mejor acceso a espacios cerrados. Es muy empleada en robots manipuladores que realicen tareas de cierta complejidad. Este tipo de configuración es la más empleada en educación, investigación y desarrollo. Como se mencionó anteriormente con este tipo de configuración, es posible barrer un gran volumen de trabajo. Si la longitud de los tres

enlaces es de L , suponiendo un radio de giro de 360° , el volumen de trabajo sería de una esfera de radio $2L$, es decir $(32/3) \pi L^3$. En la figura 1.13 se muestra el robot angular ABB 1400.



Figura 1.13 Robot angular ABB 1400

1.2.6.5 Configuración Scara

Este tipo de configuración se constituye por dos articulaciones de rotación con respecto a dos ejes paralelos, y una de desplazamiento en sentido perpendicular al plano.

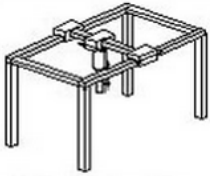
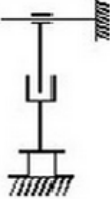

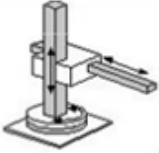
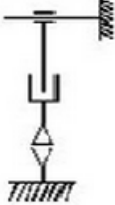

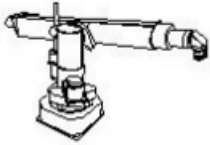
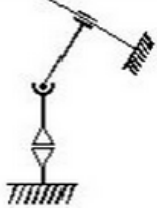


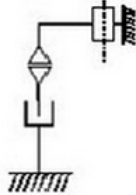




La aplicación de este tipo de estructura es esencialmente para tareas de montaje en un plano. El volumen de trabajo, considerando segmentos de longitud L , radio de giro de 360° y rango de desplazamientos de L es de $4\pi L^3$. Un ejemplo es el Robot OMRON XG que se puede observar en la figura 1.14.



Figura 1.14 Robot Scara OMRON XG

En la tabla 1.1 se presenta el resumen de las distintas configuraciones de robots manipuladores con su respectiva estructura cinemática y volumen de trabajo.

Tabla 1.1 Estructura cinemática y volumen de trabajo de las configuraciones típicas de robots manipuladores

Configuración	Esquema	Estructura cinemática	Volumen de trabajo
Cartesiana			
Cilíndrica			
Esférica			
Scara			
Angular			

1.3 ROBOT STANFORD

El robot Stanford fue diseñado en 1969 por Victor Scheinman, un estudiante de Ingeniería Mecánica que trabajaba en el laboratorio de Inteligencia Artificial de la Universidad de Stanford, ver figura 1.15. Este manipulador electro-mecánico de seis grados de libertad fue uno de los primeros robots diseñado exclusivamente para ser controlado por computador. El robot Stanford era capaz de alcanzar cualquier posición en el espacio bajo el control de una computadora, ampliando el uso de los robots a aplicaciones más complejas como el ensamblaje y la soldadura por arco.⁹

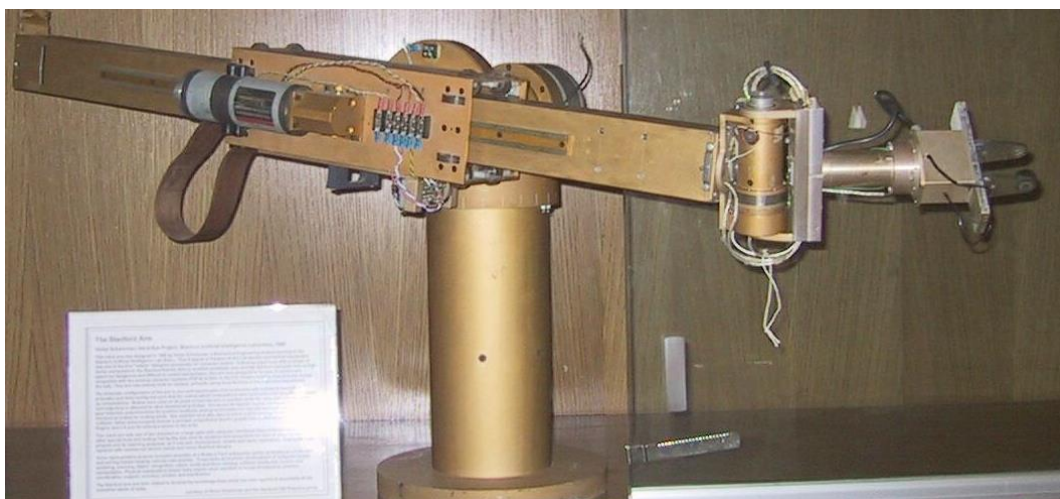


Figura 1.15 Robot Stanford -1969

Después de la experiencia con modelos previos como con el brazo Stanford-Rancho (una prótesis de brazo modificada) y el brazo hidráulico Stanford (un manipulador de alta velocidad pero difícil y peligroso de manejar), este brazo electro-mecánico fue diseñado para ser fácil de controlar y para ser compatible con los sistemas computacionales existentes (PDP-6). Este robot fue diseñado y construido en su totalidad en el campus de la Universidad de Stanford utilizando material disponible en la misma.

⁹ <http://infolab.stanford.edu/pub/voy/museum/pictures/display/1-Robot.htm>

En su aplicación inicial, fueron usados frenos en todas las articulaciones para mantener al brazo en posición mientras la computadora calculaba la siguiente trayectoria o trabajaba en otras actividades paralelas. Los actuadores son motores eléctricos DC, reductores armónicos y de engranajes rectos, potenciómetros para retroalimentación de posición, tacómetros analógicos para retroalimentación de velocidad y frenos electro-mecánicos para el bloqueo de las articulaciones. Embragues deslizantes también se utilizaron para evitar daños en la unidad en caso de una colisión. Otras mejoras incluyen un servo-motor, pinza eléctrica con contactos sensoriales táctiles en los dedos, y un sensor de fuerza / torque de 6 ejes en la muñeca .

El brazo robot Stanford ayudó a desarrollar el conocimiento base, que ha sido aplicado en la totalidad de robots industriales de hoy en día.

1.3.1 ESTRUCTURA DEL ROBOT STANFORD

El robot Stanford al igual que los demás tipos de robots manipuladores está formado por eslabones y articulaciones que integran una cadena cinemática abierta. El Robot Stanford es un robot esférico ya que presenta una estructura RRP, es decir el brazo presenta dos articulaciones de rotación y una prismática. Considerando también la muñeca del robot manipulador Stanford se observa que presenta en total 6 articulaciones, 5 de rotación y 1 prismática. Ver figura 1.16.

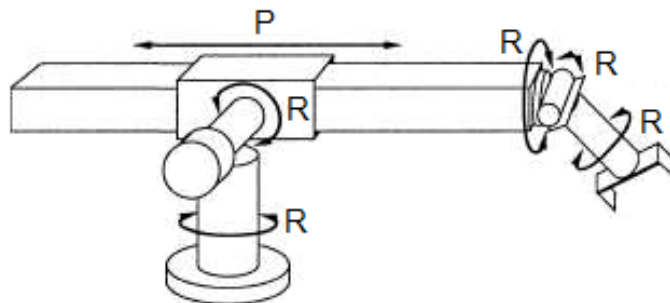


Figura 1.16 Estructura del Robot Stanford Fuente: Propia

2 CAPÍTULO II

REPRESENTACIÓN DE LA POSICIÓN Y ORIENTACIÓN [7]

Ya que la manipulación de piezas llevada a cabo por un robot implica el movimiento espacial de sus eslabones, debido a la variación de sus variables articulares, es de mucha importancia saber cómo se representa la posición y la orientación del efector final del robot para conocer con precisión en qué punto del espacio de trabajo se colocará el mismo. Para esto es necesario estudiar la representación y transformación de los movimientos típicos de un robot manipulador, que como ya se mencionó anteriormente son la rotación y la traslación.

2.1 REPRESENTACIÓN DE LA POSICIÓN (TRASLACIÓN)

2.1.1 EN EL PLANO

En un par prismático al eslabón fijo le asignamos el sistema de coordenadas $o_0x_0y_0$ y al eslabón móvil que se traslada le asignamos el sistema $o_1x_1y_1z_1$, ver figura 2.1, los orígenes de los dos sistemas coinciden inicialmente, se desea determinar las ecuaciones que relaciona las coordenadas de $P(x_1, y_1)$ con las coordenadas de $P(x_0, y_0)$ después de trasladar al sistema móvil Δx a lo largo de x_1 , y Δy a lo largo de y_1 . Un movimiento de traslación en el mundo físico puede ser representado a través de una matriz de traslación en el mundo virtual.

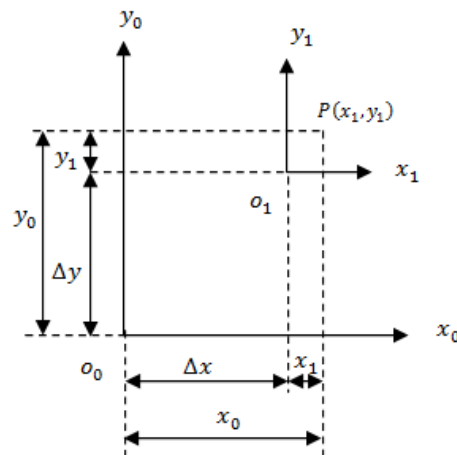


Figura 2.1 Traslación en el plano

Se asume que un punto $P(x_1, y_1)$ adherido al sistema de coordenadas móvil, se traslada conjuntamente con el punto P, un valor de Δx a lo largo de x_1 , y Δy a lo largo de y_1 , las ecuaciones que relaciona a estos sistemas de coordenadas son:

$$x_0 = x_1 + \Delta x \quad (2.1)$$

$$y_0 = y_1 + \Delta y \quad (2.2)$$

Para poder escribir las ecuaciones en notación matricial se añade una tercera ecuación $1=1$, con esta igualdad se tiene.

$$\begin{bmatrix} x_0 \\ y_0 \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & \Delta x \\ 0 & 1 & \Delta y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ y_1 \\ 1 \end{bmatrix} \quad (2.3)$$

$$\begin{bmatrix} x_0 \\ y_0 \\ 1 \end{bmatrix} = T \begin{bmatrix} x_1 \\ y_1 \\ 1 \end{bmatrix} \quad (2.4)$$

Donde, la matriz de traslación en el plano es

$$T = \begin{bmatrix} 1 & 0 & \Delta x \\ 0 & 1 & \Delta y \\ 0 & 0 & 1 \end{bmatrix} \quad (2.5)$$

2.1.2 EN EL ESPACIO

En la figura 2.2 se tiene el sistema de coordenadas móvil o, u, v, w trasladado con respecto al sistema fijo a lo largo del eje x un valor de Δx , en el eje y un valor de Δy , en el eje z un valor de Δz , se conoce las coordenadas del punto P con respecto al sistema móvil, este punto P se mueve conjuntamente con el sistema de coordenadas móvil, se desea determinar las ecuaciones que relaciona las coordenadas de estos dos sistemas después del movimiento de traslación.

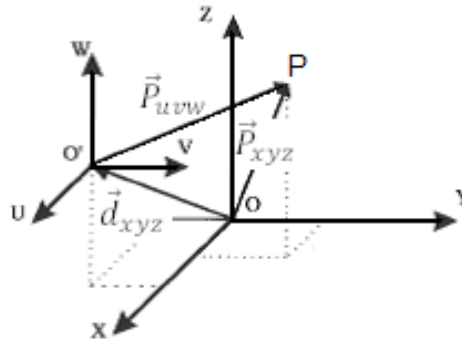


Figura 2.2 Traslación en el espacio

Las coordenadas del punto P con respecto a los dos sistemas de referencia son

$$P_{x,y,z}(p_x, p_y, p_z) \quad (2.6)$$

$$P_{u,v,w}(p_u, p_v, p_w) \quad (2.7)$$

La traslación del sistema móvil con respecto al fijo es

$$d_{x,y,z}(\Delta_x, \Delta_y, \Delta_z) \quad (2.8)$$

Utilizando algebra vectorial se plantea la ecuación:

$$\vec{P}_{xyz} = \vec{P}_{uvw} + \vec{d}_{xyz} \quad (2.9)$$

$$p_x \vec{i}_x + p_y \vec{j}_y + p_z \vec{k}_z = (p_u \vec{i}_u + p_v \vec{j}_v + p_w \vec{k}_w) + (\Delta_x \vec{i}_x + \Delta_y \vec{j}_y + \Delta_z \vec{k}_z) \quad (2.10)$$

Al trasladarse el sistema de coordenadas móvil, no cambia la orientación de los vectores unitarios en consecuencia:

$$\vec{i}_u = \vec{i}_x \quad \vec{j}_v = \vec{j}_y \quad \vec{k}_w = \vec{k}_z \quad (2.11)$$

La ecuación 2.10 se puede escribir escalarmente como sigue:

$$p_x = p_u + \Delta_x \quad (2.12)$$

$$p_y = p_v + \Delta_y$$

$$p_z = p_w + \Delta_z$$

Para poder escribir matricialmente a estas tres ecuaciones se utiliza un artificio que es utilizar una cuarta ecuación $1=1$

$$\begin{bmatrix} p_x \\ p_y \\ p_z \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & \Delta x \\ 0 & 1 & 0 & \Delta y \\ 0 & 0 & 1 & \Delta z \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} p_u \\ p_v \\ p_w \\ 1 \end{bmatrix} \quad (2.13)$$

Donde la matriz de traslación en el espacio cuando se traslada en los tres ejes viene dada por:

$$T = \begin{bmatrix} 1 & 0 & 0 & \Delta x \\ 0 & 1 & 0 & \Delta y \\ 0 & 0 & 1 & \Delta z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.14)$$

2.2 REPRESENTACIÓN DE LA ORIENTACIÓN (ROTACIÓN)

2.2.1 EN EL PLANO

Para establecer la relación entre dos sistemas de coordenadas, ambos con el mismo origen y uno de ellos rotado con respecto al otro, por ejemplo en un par de rotación, al eslabón fijo le asignamos el sistema de coordenadas $o_0x_0y_0$ y al eslabón móvil le asignamos el sistema $o_1x_1y_1$, los orígenes de los dos sistemas coinciden en el estado inicial, se desea determinar las ecuaciones que relacionan las coordenadas x_0y_0 en función de x_1y_1 .

Es conveniente indicar que un movimiento de rotación en el mundo físico puede ser representado correctamente a través de una matriz de rotación, en la modelación de mecanismos planares y espaciales, este es el concepto fundamental en el que se basa el método matricial de la robótica.

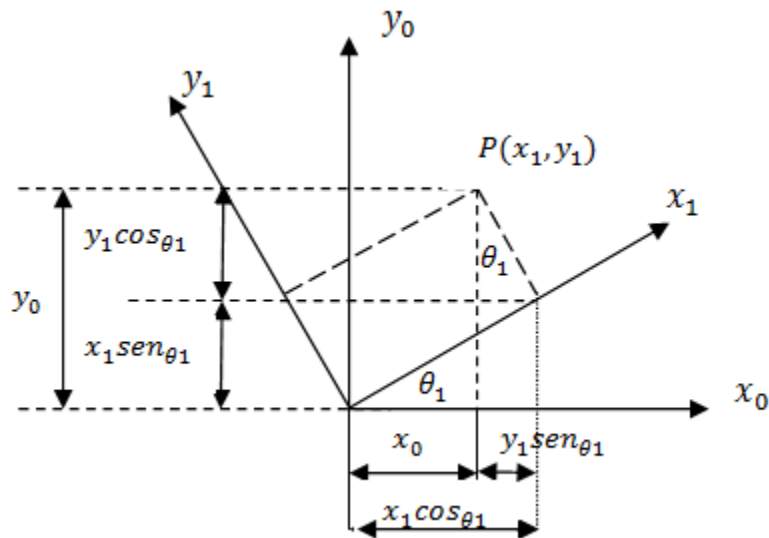


Figura 2.3 Rotación en el plano

Las ecuaciones que permiten transformar las coordenadas pueden ser escritas utilizando la figura 2.3 y el axioma el todo es igual a la suma de las partes:

$$x_0 = x_1 C_{\theta_1} - y_1 S_{\theta_1} \quad (2.15)$$

$$y_0 = x_1 S_{\theta_1} + y_1 C_{\theta_1} \quad (2.16)$$

$$1 = 1$$

A las dos primeras ecuaciones le añadimos una tercera la igualdad $1 = 1$, con la finalidad de tener una matriz 3×3 a esta matriz se le conoce como matriz de transformación homogénea, en vez de una matriz de rotación pura de 2×2 , esto nos permite operar fácilmente con las matrices de traslación que estudiaremos más adelante. Estas tres ecuaciones pueden escribirse utilizando notación matricial como se indica a continuación, note que cuando trabajamos en el plano vamos a tener matrices de orden 3, a estas matrices se les conoce como matrices ampliadas:

$$\begin{bmatrix} x_0 \\ y_0 \\ 1 \end{bmatrix} = \begin{bmatrix} C_{\theta_1} & -S_{\theta_1} & 0 \\ S_{\theta_1} & C_{\theta_1} & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ y_1 \\ 1 \end{bmatrix} \quad (2.17)$$

Donde la matriz de rotación R_{z, θ_1} representa un giro alrededor del eje z un ángulo de θ_1 , y le representaremos como sigue:

$$\begin{bmatrix} x_0 \\ y_0 \\ 1 \end{bmatrix} = R_{Z,\theta_1} \begin{bmatrix} x_1 \\ y_1 \\ 1 \end{bmatrix} \quad (2.18)$$

$$R_{Z,\theta_1} = \begin{bmatrix} C_{\theta_1} & -S_{\theta_1} & 0 \\ S_{\theta_1} & C_{\theta_1} & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (2.19)$$

En la figura 2.4 se ha dibujado el sistema de coordenadas móvil, al que se le ha rotado un ángulo θ , obsérvese que el punto P se mueve solidario con el sistema de coordenadas móvil, ahora se desea escribir las ecuaciones matriciales que relacionan a las coordenadas x_1, y_1 en función de las coordenadas x_0, y_0

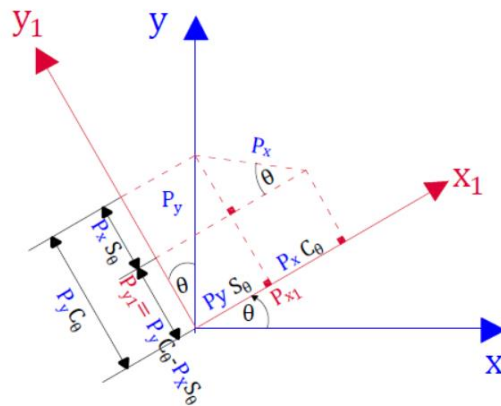


Figura 2.4 x_1, y_1 en función de x_0, y_0 Fuente:Propia

De la figura 2.4 se desprenden las ecuaciones de transformación de coordenadas

$$\begin{aligned} x_1 &= x_0 C_{\theta_1} + y_0 S_{\theta_1} \\ y_1 &= -x_0 S_{\theta_1} + y_0 C_{\theta_1} \\ 1 &= 1 \end{aligned}$$

Estas tres ecuaciones pueden ser reescritas utilizando notación matricial.

$$\begin{bmatrix} x_1 \\ y_1 \\ 1 \end{bmatrix} = \begin{bmatrix} C_{\theta_1} & S_{\theta_1} & 0 \\ -S_{\theta_1} & C_{\theta_1} & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_0 \\ y_0 \\ 1 \end{bmatrix} \quad (2.20)$$

De la ecuación 2.18 se puede despejar las coordenadas del punto P con respecto al sistema de coordenadas móvil x_1, y_1

$$\begin{bmatrix} x_1 \\ y_1 \\ 1 \end{bmatrix} = (R_{Z,\theta_1})^{-1} \begin{bmatrix} x_0 \\ y_0 \\ 1 \end{bmatrix} \quad (2.21)$$

Obligadamente la ecuación 2.20 y la 2.21 son iguales, por lo tanto se debe cumplir :

$$(R_{Z,\theta_1})^{-1} = \begin{bmatrix} C_{\theta_1} & S_{\theta_1} & 0 \\ -S_{\theta_1} & C_{\theta_1} & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (2.22)$$

Comparando la ecuación 2.22 con la ecuación 2.19 se desprende que la transpuesta de la ecuación 2.19 es exactamente igual a la matriz inversa presentada en la ecuación 2.22

$$(R_{Z,\theta_1}) = \begin{bmatrix} C_{\theta_1} & -S_{\theta_1} & 0 \\ S_{\theta_1} & C_{\theta_1} & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$$(R_{Z,\theta_1})^T = \begin{bmatrix} C_{\theta_1} & S_{\theta_1} & 0 \\ -S_{\theta_1} & C_{\theta_1} & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Se concluye que $(R_{Z,\theta_1})^T$ es igual a $(R_{Z,\theta_1})^{-1}$

2.2.2 EN EL ESPACIO

En la figura 2.5 se tiene dos sistemas de coordenadas inicialmente traslapados, el sistema de coordenadas fijo o, x, y, z y el sistemas de coordenadas móvil o, u, v, w , si se asume se tiene un punto P el cual está pegado al sistema de coordenadas móvil, esto implica si el sistema que el punto P se mueve conjuntamente con el sistema o, u, v, w .

La posición del punto P puede ser representado por el vector posición con respecto a los dos sistemas de coordenadas. Los vectores unitarios del sistema o, x, y, z serán $\vec{i}_x, \vec{j}_y, \vec{k}_z$, mientras que los de o, u, v, w serán $\vec{i}_u, \vec{j}_v, \vec{k}_w$.

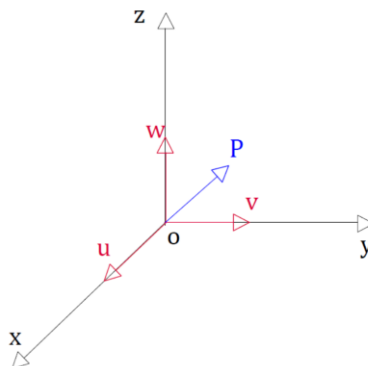


Figura 2.5 Rotación en el espacio Fuente: Propia

El vector posición del punto P referido al sistema fijo y al sistema móvil será:

$$\begin{aligned}\vec{P}_{x,y,z} &= P_x \vec{i}_x + P_y \vec{j}_y + P_z \vec{k}_z \\ \vec{P}_{u,v,w} &= P_u \vec{i}_u + P_v \vec{j}_v + P_w \vec{k}_w\end{aligned}$$

Las coordenadas del punto P con respecto al sistema fijo se pueden determinar utilizando el producto punto entre el vector $\vec{P}_{u,v,w}$ y el unitario correspondiente a la coordenada que se requiere encontrar.

$$\begin{aligned}p_x &= \vec{P}_{u,v,w} \cdot \vec{i}_x = (P_u \vec{i}_u + P_v \vec{j}_v + P_w \vec{k}_w) \cdot \vec{i}_x \\ p_y &= \vec{P}_{u,v,w} \cdot \vec{j}_y = (P_u \vec{i}_u + P_v \vec{j}_v + P_w \vec{k}_w) \cdot \vec{j}_y \\ p_z &= \vec{P}_{u,v,w} \cdot \vec{k}_z = (P_u \vec{i}_u + P_v \vec{j}_v + P_w \vec{k}_w) \cdot \vec{k}_z\end{aligned} \quad (2.23)$$

Las tres ecuaciones escritas en notación matricial se presentan a continuación:

$$\begin{bmatrix} p_x \\ p_y \\ p_z \end{bmatrix} = \begin{bmatrix} \vec{i}_u \cdot \vec{i}_x & \vec{j}_v \cdot \vec{i}_x & \vec{k}_w \cdot \vec{i}_x \\ \vec{i}_u \cdot \vec{j}_y & \vec{j}_v \cdot \vec{j}_y & \vec{k}_w \cdot \vec{j}_y \\ \vec{i}_u \cdot \vec{k}_z & \vec{j}_v \cdot \vec{k}_z & \vec{k}_w \cdot \vec{k}_z \end{bmatrix} \begin{bmatrix} p_u \\ p_v \\ p_w \end{bmatrix} \quad (2.24)$$

Donde la matriz de rotación R contiene los vectores unitarios del sistema de coordenadas móvil con respecto al sistema de coordenadas fijo, es una matriz ortonormal.

$$R = \begin{bmatrix} \vec{i}_u \cdot \vec{i}_x & \vec{j}_v \cdot \vec{i}_x & \vec{k}_w \cdot \vec{i}_x \\ \vec{i}_u \cdot \vec{j}_y & \vec{j}_v \cdot \vec{j}_y & \vec{k}_w \cdot \vec{j}_y \\ \vec{i}_u \cdot \vec{k}_z & \vec{j}_v \cdot \vec{k}_z & \vec{k}_w \cdot \vec{k}_z \end{bmatrix} \quad (2.25)$$

Utilizando la ecuación matricial 2.25 y la figura 2.6 se puede determinar las matrices de rotación pura alrededor del eje x, eje y, eje z

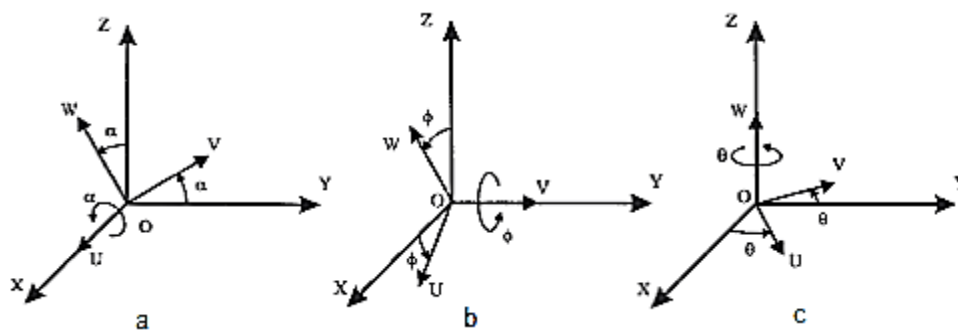


Figura 2.6 Rotación del sistema o,u,v,w alrededor de x, α , de y, ϕ , de z, θ .

Las matrices de rotación puras de orden 3x3 se presentan a continuación:

$$R_{x,\alpha} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\alpha & -\text{sen}\alpha \\ 0 & \text{sen}\alpha & \cos\alpha \end{bmatrix} \quad (2.26)$$

$$R_{y,\phi} = \begin{bmatrix} \cos\phi & 0 & \text{sen}\phi \\ 0 & 1 & 0 \\ -\text{sen}\phi & 0 & \cos\phi \end{bmatrix} \quad (2.27)$$

$$R_{z,\theta} = \begin{bmatrix} \cos\theta & -\text{sen}\theta & 0 \\ \text{sen}\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (2.28)$$

Para poder operar las matrices de rotación de orden 3x3 se necesita ampliarles y para ello se añade una columna y una fila como se indica a continuación, de tal manera que se pueda trabajar con las matrices de traslación que son de orden 4x4:

$$R_{x,\alpha} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\alpha & -\text{sen}\alpha & 0 \\ 0 & \text{sen}\alpha & \cos\alpha & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.29)$$

$$R_{y,\phi} = \begin{bmatrix} \cos\phi & 0 & \text{sen}\phi & 0 \\ 0 & 1 & 0 & 0 \\ -\text{sen}\phi & 0 & \cos\phi & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.30)$$

$$R_{z,\theta} = \begin{bmatrix} \cos\theta & -\text{sen}\theta & 0 & 0 \\ \text{sen}\theta & \cos\theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.31)$$

Si al sistema de coordenadas móvil se le realizan varios movimientos de rotación referidos al último sistema de coordenadas, en este caso se puede multiplicar las matrices de rotación en la misma secuencia de los movimientos, a esto se le conoce como postmultiplicación matricial.

Si los movimientos son realizados con referencia a un sistema de coordenadas fijo, se tiene que multiplicar las matrices en secuencia inversa a la de los movimientos, en este caso se le conoce como premultiplicación matricial

2.3 REPRESENTACIÓN DE MOVIMIENTOS COMPUESTOS

En robótica generalmente nos encontraremos con casos que involucren una serie de movimientos compuestos. A continuación se presentan los métodos para determinar las ecuaciones de transformación que relacionan al efector final (donde se encuentra el último sistema de coordenadas) con la base del robot manipulador (sistema de coordenadas del origen)

2.3.1 POSTMULTIPLICACION MATRICIAL

Al realizar los movimientos del sistema de coordenadas móvil con respecto a los ejes del último sistema de coordenadas móvil, se puede determinar las ecuaciones de transformación de coordenadas entre el último sistema de coordenadas con el sistema de coordenadas fijo, para ello se debe utilizar el producto matricial en la misma secuencia de los movimientos, a este producto de matrices se le conoce como postmultiplicación matricial y es muy común en robótica.

Todas las aplicaciones anteriores son ejemplos de postmultiplicación matricial, también cae en este caso los ángulos de Euler [11], aquí se tiene tres rotaciones consecutivas, rotación de α alrededor del eje z, rotación β alrededor del y, rotación γ alrededor del eje x, debemos aclarar que estas tres rotaciones son referidas al último sistema de coordenadas móvil.

$$R_{(\alpha,\beta,\gamma)} = R_{z1,\alpha}R_{y2,\beta}R_{x3,\gamma} \quad (2.32)$$

$$R_{(\alpha,\beta,\gamma)} = \begin{bmatrix} c\alpha & -s\alpha & 0 \\ s\alpha & c\alpha & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} c\beta & 0 & s\beta \\ 0 & 1 & 0 \\ -s\beta & 0 & c\beta \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & c\gamma & -s\gamma \\ 0 & s\gamma & c\gamma \end{bmatrix} \quad (2.33)$$

$$R_{(\alpha,\beta,\gamma)} = \begin{bmatrix} cac\beta & cas\beta s\gamma - sac\gamma & cas\beta c\gamma + sas\gamma \\ sac\beta & sas\beta s\gamma + cac\gamma & sas\beta c\gamma - cas\gamma \\ -s\beta & c\beta s\gamma & c\beta c\gamma \end{bmatrix} \quad (2.34)$$

Este principio se utiliza en los métodos de Denavit-Hartenberg estándar y modificado, la diferencia entre estos dos métodos radica en que en el estándar la secuencia de movimientos es primero en z y luego en x, mientras que en el modificado la secuencia de movimientos es primero en x y luego en z, cabe recalcar que en estos dos métodos se utilizan grupos de cuatro movimientos, estos movimientos se escriben en una fila de la tabla de parámetros de Denavit-Hartenberg, al multiplicar las matrices de estos cuatro movimientos se obtiene la matriz de transformación de coordenadas del sistema S_{i+1} con respecto al sistema S_i .

El método del Movimiento General Continuo MG, desarrollado en la Escuela Politécnica Nacional por el Ing. Mario Granja MSc. en el año 2014, utiliza este mismo principio, sin considerar grupos de cuatro movimientos si no que el producto matricial se realiza de manera continua obteniéndose directamente la matriz de transformación homogénea del brazo del robot H.

2.3.2 PREMÚLTIPLICACION MATRICIAL

Si los movimiento del sistema de coordenadas se realiza con respecto al sistema de coordenadas fijo, y se requiere la relación entre estos sistemas se debe multiplicar las matrices en secuencia inversa a la de los movimientos, a esto se le conoce como premultiplicación matricial de mucha utilidad en aeronáutica, en asuntos marítimos y en menor proporción en robótica. Una aplicación frecuente es la matriz RPY [11], que son las siglas en ingles de: roll (balanceo), pitch (inclinación), yaw (orientación).

La secuencia de movimientos es giro alrededor del eje x_0 un ángulo de γ (balanceo), rotación alrededor del eje y_0 un ángulo β (inclinación), rotación alrededor del eje z_0 un ángulo α (orientación).

$$R_{RPY(\gamma,\beta,\alpha)} = R_{z0,\alpha}R_{y0,\beta}R_{x0,\gamma} \quad (2.35)$$

$$R_{RPY(\gamma,\beta,\alpha)} = \begin{bmatrix} c\alpha & -s\alpha & 0 \\ s\alpha & c\alpha & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} c\beta & 0 & s\beta \\ 0 & 1 & 0 \\ -s\beta & 0 & c\beta \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & c\gamma & -s\gamma \\ 0 & s\gamma & c\gamma \end{bmatrix} \quad (2.36)$$

$$R_{RPY(\gamma,\beta,\alpha)} = \begin{bmatrix} c\alpha c\beta & c\alpha s\beta s\gamma - s\alpha c\gamma & c\alpha s\beta c\gamma + s\alpha s\gamma \\ s\alpha c\beta & s\alpha s\beta s\gamma + c\alpha c\gamma & s\alpha s\beta c\gamma - c\alpha s\gamma \\ -s\beta & c\beta s\gamma & c\beta c\gamma \end{bmatrix} \quad (2.37)$$

3 CAPÍTULO III

CINEMÁTICA DE UN ROBOT MANIPULADOR

En robótica, la cinemática se refiere al estudio del movimiento del extremo del robot con respecto a un sistema de coordenadas que se toma como referencia, sin considerar las fuerzas aplicadas, y busca describir analíticamente el movimiento espacial del robot, relacionando la posición del efector final con los valores articulares del robot.

La cinemática puede ser directa o inversa. La cinemática directa nos permite determinar la posición y la orientación del efector final en base a valores de las coordenadas articulares conocidos. Mientras que la cinemática inversa nos permite realizar lo contrario, determinar los valores de las coordenadas articulares conociendo la posición y la orientación del efector final, como se muestra en la figura 3.1

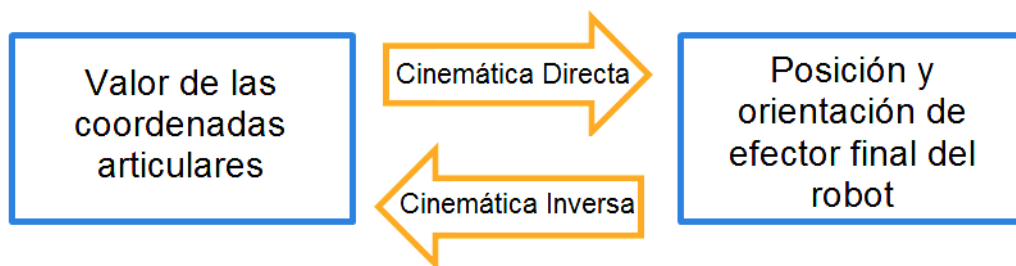


Figura 3.1 Cinemática directa e inversa

El movimiento de una cadena cinemática de un robot manipulador es modelado por las ecuaciones cinemáticas de la cadena. Estas ecuaciones definen la configuración de la cadena en términos de sus parámetros conjuntos. La cinemática directa utiliza los parámetros comunes para calcular la configuración de la cadena, y la cinemática inversa invierte este cálculo para determinar los parámetros o variables articulares que logran una configuración deseada.

3.1 CINEMÁTICA DIRECTA

En el problema cinemático directo se conocen los valores de sus coordenadas articulares y busca determinar la posición del efector final. En este caso la solución del problema es única.

En general, un robot de n grados de libertad está formado por n eslabones unidos a n articulaciones, a cada eslabón se le puede asociar un sistema de referencia solidario a él y utilizando las transformaciones homogéneas es posible representar las rotaciones y traslaciones relativas entre los distintos eslabones que componen el robot.

Se utiliza para su solución el algoritmo de Denavit-Hartenberg, mediante el cual obtendremos matrices de transformación homogénea para cada grado de libertad. Cada matriz de transformación tendrá la información de la posición, orientación, perspectiva y escala de sus ejes coordenados correspondientes, respecto a ejes anteriores o de referencia. De esta forma se puede obtener las ecuaciones cinemáticas de la cadena completa.

Según la representación D-H, escogiendo adecuadamente los sistemas de coordenadas asociados a cada eslabón, será posible pasar de uno al siguiente mediante cuatro transformaciones básicas que dependen exclusivamente de las características geométricas del eslabón.

Transformaciones básicas:

1. Rotación alrededor del eje z_{i-1} un ángulo θ_i
2. Traslación a lo largo del eje z_{i-1} una distancia d_i / vector d_i $(0,0,d_i)$
3. Traslación a lo largo del eje x_i una distancia a_i / vector a_i $(a_i,0,0)$
4. Rotación alrededor del eje x_i un ángulo α_i

Las transformaciones han de ejecutarse en el orden correcto ya que el producto de las matrices no es conmutativo.

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

De esta forma, el problema cinemático directo se reduce a encontrar una matriz homogénea de transformación H que relacione la posición y orientación del extremo del robot respecto del sistema de referencia fijo situado en la base del mismo. Esta matriz H será función de las coordenadas articulares y tendrá la estructura mostrada en la figura 3.2.

$$A_j^k = \begin{pmatrix} r_{11} & r_{12} & r_{13} & d_x \\ r_{21} & r_{22} & r_{23} & d_y \\ r_{31} & r_{32} & r_{33} & d_z \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

Figura 3.2 Estructura de una matriz homogénea

La matriz homogénea expresará la orientación (submatriz (3x3) de rotación) y posición (submatriz (3x1) de traslación) del extremo del robot en función de sus coordenadas articulares, con lo que quedará resuelto el problema cinemático directo.

3.1.1 ASIGNACIÓN DE LOS SISTEMAS DE COORDENADAS

Para un robot manipulador dado, uno puede elegir siempre los sistemas de coordenadas $0, \dots, n$ de tal manera que las dos condiciones anteriores se cumplan. En ciertas circunstancias, esto requerirá la colocación del origen de o_i del sistema de coordenadas i en un lugar que no puede ser intuitivamente satisfactorio, pero por lo general esto no será el caso. Al leer el material siguiente, es importante tener en cuenta que las opciones de los varios sistemas de coordenadas no son únicas, aun cuando

son restringidos por los requerimientos anteriores. Por lo tanto, es posible que diferentes ingenieros obtengan diferentes, pero igualmente correctas, asignaciones de los sistemas de coordenadas para los eslabones del robot. Es muy importante observar, sin embargo, que el resultado final (por ejemplo, la matriz T_n^0) será el mismo, independientemente de la asignación de los sistemas de coordenadas de los eslabones intermedios (asumiendo que el sistema de coordenadas para el eslabón n coincide). Comenzaremos derivando el procedimiento general. Luego discutiremos varios casos especiales comunes en los que es posible simplificar aún más la matriz de transformación homogénea

Para empezar, la elección de z_i es arbitraria. En particular, a partir de (3.16), vemos que eligiendo α_i y θ_i adecuadamente, podemos obtener cualquier dirección arbitraria para z_i . Por lo tanto, para nuestro primer paso, asignamos los ejes z_0, \dots, z_{n-1} de una manera agradable intuitivamente. Específicamente, asignamos z_i a ser el eje de accionamiento para la articulación $i + 1$. Por lo tanto, z_0 es el eje de accionamiento para la articulación 1, z_1 es el eje de accionamiento para la articulación 2, etc. Hay dos casos a considerar: (i) si la articulación $i + 1$ es de rotación, z_i es el eje de rotación de la articulación $i + 1$; (ii) si la articulación $i + 1$ es prismática, z_i es el eje de la traslación de la articulación $i + 1$. Al principio puede parecer un poco confuso asociar z_i con la articulación $i + 1$, pero recuerde que esta satisface la convención que hemos establecido en la Sección 3.1, a saber, que las articulaciones i se fija con respecto al sistema de coordenadas i , y que cuando la articulación i se acciona, el eslabón i y su sistema de coordenadas adjunto, $o_i x_i y_i z_i$, experimentan un movimiento resultante.

Una vez que hemos establecido los ejes z para los eslabones, se establece el sistema de coordenadas base. La elección de un sistema de coordenadas base es casi arbitraria. Podemos elegir el origen o_0 del sistema de coordenadas base en cualquier punto sobre z_0 . A continuación, elegimos x_0, y_0 de cualquier manera conveniente

siempre y cuando el sistema de coordenadas sea dextrógiro. Esto establece el sistema de coordenadas 0 .

Una vez que el sistema de coordenadas 0 se ha establecido, comenzamos un proceso iterativo en el que se define el sistema de coordenadas i utilizando el sistema de coordenadas $i - 1$, comenzando con el sistema de coordenadas 1 . La Figura 3.3 será de utilidad para la comprensión del proceso que ahora describimos.

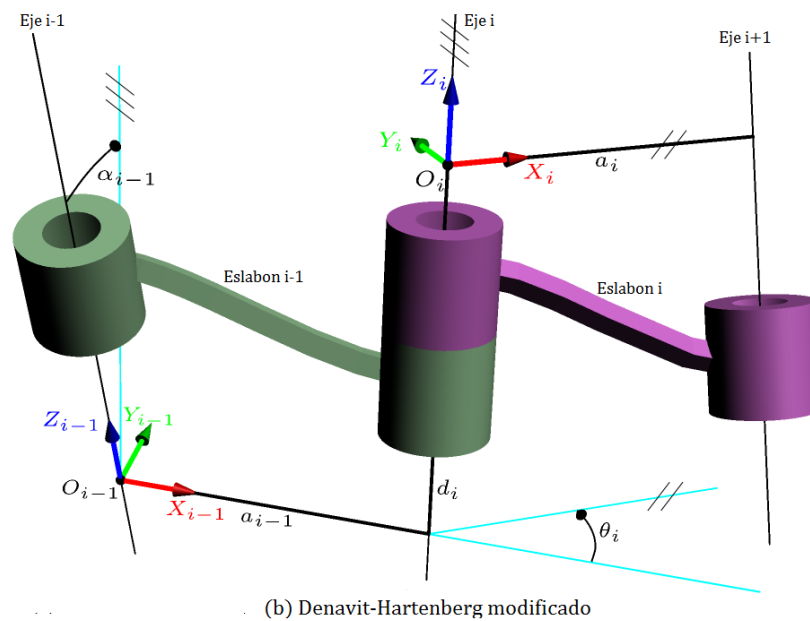
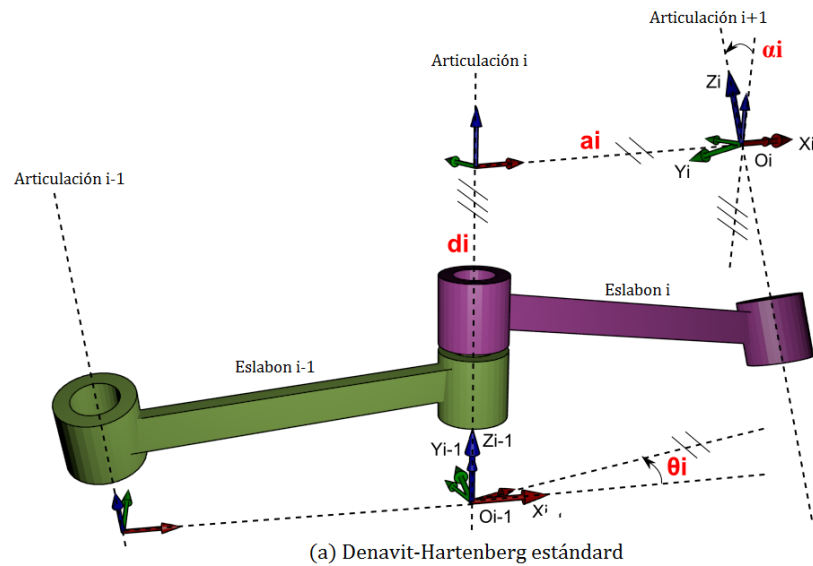


Figura 3.3 Sistema de coordenadas Denavit-Hartenberg estándar y modificado [17].

Con el fin de establecer el sistema de coordenadas i , es necesario considerar tres casos: (i) los ejes z_{i-1} , z_i no son coplanares, (ii) los ejes z_{i-1} , z_i se intersecan (iii) los ejes z_{i-1} , z_i son paralelos. Tenga en cuenta que en ambos casos (ii) y (iii) los ejes z_{i-1} , z_i son coplanares. Esta situación es de hecho bastante común, como veremos en la Sección 4.3. Consideremos ahora cada uno de estos tres casos.

(i) z_{i-1} y z_i no son coplanares: Si z_{i-1} y z_i no son coplanares, entonces existe un segmento de línea única perpendicular tanto a z_{i-1} y z_i tal que conecta las dos líneas y que tiene una longitud mínima. La línea que contiene esta normal común a z_{i-1} y z_i define x_i , y el punto donde esta línea corta z_i es el origen o_i . Por construcción, ambas condiciones (DH1) y (DH2) se cumplen y el vector de o_{i-1} a o_i es una combinación lineal de z_{i-1} y x_i . La especificación del sistema de coordenadas se completa mediante la elección del eje y_i para formar el sistema de coordenadas dextrógiro. De los supuestos (DH1) y (DH2) se satisfacen la matriz de transformación homogénea A_i y es de la forma (4.10).

(ii) z_{i-1} es paralela a z_i : Si los ejes z_{i-1} y z_i son paralelos, entonces hay un número infinito de muchas normales comunes entre ellos y la condición (DH1) no especifica completamente el x_i . En este caso tenemos la libertad de elegir el origen o_i en cualquier lugar a lo largo de z_i . A menudo se elige o_i para simplificar las ecuaciones resultantes. El eje x_i se elige entonces ya sea para ser dirigido desde o_i hacia z_{i-1} , a lo largo de la normal común, o contrario a este vector. Un común método para elegir o_i es elegir la normal que pasa a través o_{i-1} como el eje x_i ; o_i es entonces el punto en el que este se cruza normales z_i . En este caso, d_i sería igual a cero. Una vez que se fija x_i , se determina y_i , como es habitual por la regla de la mano derecha. Dado que los ejes z_{i-1} y z_i son paralelos, α_i será igual a cero en este caso.

(iii) z_{i-1} interseca al eje z_i : En este caso se elige x_i normal al plano formado por z_i y z_{i-1} . El sentido positivo de x_i es arbitraria. La opción más natural para el origen o_i en este caso es en el punto de intersección de z_i y z_{i-1} . Sin embargo, cualquier punto conveniente a lo largo del eje z_i basta. Tenga en cuenta que en este caso el parámetro a_i es igual a 0

Este procedimiento constructivo funciona para sistemas de coordenadas de $0, \dots, n - 1$ en un robot de n -eslabones. Para completar la construcción, es necesario especificar el sistema de coordenadas n . El sistema de coordenadas final $o_n x_n y_n z_n$ que comúnmente se conoce como el extremo del efector o sistema de coordenadas de la herramienta (véase la figura 3.4).

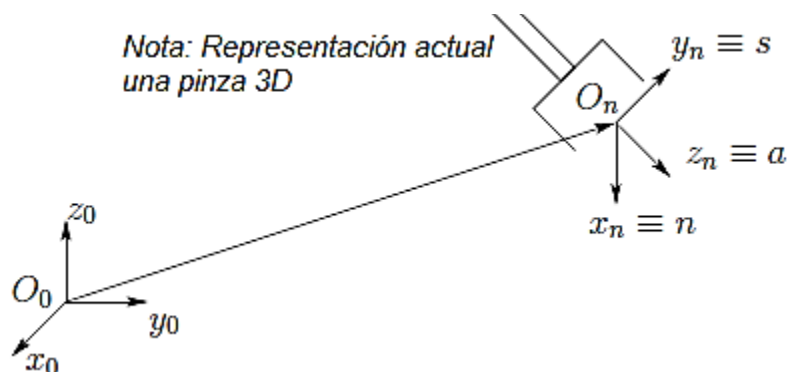


Figura 3.4 Asignación del sistema de coordenadas de la herramienta.

El origen o_n se coloca frecuentemente de manera simétrica entre los dedos de la pinza. Los vectores unitarios a lo largo de los ejes x_n, y_n , y z_n están etiquetados como n, s y a respectivamente. La terminología surge del hecho de que la dirección a es la dirección de aproximación, en el sentido de que la pinza típicamente se acerca a un objeto a lo largo de la dirección a . Del mismo modo la dirección s es la dirección de deslizamiento, la dirección a lo largo de la cual los dedos de la pinza deslizante para abrir y cerrar, y n es la dirección normal al plano formado por a y s .

En los robots contemporáneos el movimiento de la articulación final es una rotación del efector final por θ_n y los dos ejes de articulación finales, z_{n-1} y z_n , coinciden. En este caso, la transformación entre los dos últimos sistemas de coordenadas es una traslación a lo largo z_{n-1} por un distancia d_n seguida (o precedida) por una rotación de θ_n radianes alrededor de z_{n-1} . Esta es una observación importante que simplifica el cálculo de la cinemática inversa.

Finalmente, tenga en cuenta el siguiente hecho importante. En todos los casos, si la articulación en cuestión es de rotación o prismática, las cantidades a_i y α_i son siempre constante para todo i y son característica del manipulador. Si la articulación i es prismática, entonces θ_i también es una constante, mientras que d_i es la variable de la articulación i ésima. Del mismo modo, si la articulación i es de rotación, entonces d_i es constante y θ_i es la variable de articulación i ésima.

Se seguirá el algoritmo de Denavit-Hartenberg para la obtención del modelo del Robot Stanford. Los pasos del algoritmo genérico para la obtención de los parámetros D-H son los siguientes:

1. *Numerar los eslabones* comenzando con 1 (primer eslabón móvil de la cadena) y acabando con n (último eslabón móvil). Se numerará como eslabón 0 a la base fija del robot.
2. *Numerar cada articulación* comenzando por 1 (la correspondiente al primer grado de libertad y acabando en n).
3. *Localizar el eje de cada articulación*. Si esta es rotativa, el eje será su propio eje de giro. Si es prismática, será el eje a lo largo del cual se produce el desplazamiento.
4. *Ejes Z*. Situamos los Z_{i-1} en los ejes de las articulaciones i , con $i=1, \dots, n$. Es decir, Z_0 va sobre el eje de la 1ª articulación, Z_1 va sobre el eje del 2º grado de libertad, etc.
5. *Situar el origen del sistema de la base* (sistema de coordenadas cero) en cualquier punto del eje Z_0 . Los ejes X_0 e Y_0 se situarán de modo que formen un sistema dextrógiro con Z_0 .
6. *Resto de sistemas* Para $i=1$ a $n-1$, situar el sistema en la intersección del eje Z_i con la normal común a Z_{i-1} y Z_i . Si ambos ejes se cortasen se situaría en el punto de corte. Si fuesen paralelos se situaría en la articulación $i+1$.
7. *Ejes X*. Situar X_i en la dirección normal común a Z_{i-1} y Z_i .
8. *Ejes Y*. Situar Y_i de modo que forme un sistema dextrógiro con X_i y Z_i .

9. *Sistema del extremo del robot.* Situar el sistema de modo que Z_n coincida con la dirección de Z_{n-1} y X_n sea normal a Z_{n-1} y Z_n .
10. *Ángulos θ .* Cada θ_i es el ángulo desde X_{i-1} hasta X_i girando alrededor de Z_i .
11. *Distancias d .* Cada d_i es la distancia desde el sistema XYZ $i-1$ hasta la intersección de las normales común de Z_{i-1} hacia Z_i , a lo largo de Z_{i-1} .
12. *Distancias a .* Obtener a_i como la distancia medida a lo largo de X_i (que ahora coincidiría con X_{i-1}) que habría que desplazar el nuevo (S_{i-1}) para que su origen coincidiese con (S_i).
13. *Ángulos α .* Ángulo que hay que rotar Z_{i-1} para llegar a Z_i , rotando alrededor de X_i .
14. Obtener las matrices individuales de transformación A.
15. Obtener la matriz de transformación que relaciona el sistema de la base con el del extremo del robot, multiplicando todas las matrices A.

3.2 CINEMÁTICA INVERSA

En el problema cinemático inverso se conoce la posición y orientación del efector final pero se desconocen las variables articulares. Es por tanto necesario resolver un conjunto de ecuaciones no lineales, de manera que se implementan diferentes métodos numéricos para llevar a cabo este cometido.

La cinemática directa permite enfrentar el problema de manera sistemática a partir de las matrices de transformación homogéneas, e independiente de la configuración del robot. Esto no es posible en la cinemática inversa, siendo el procedimiento de obtención de las ecuaciones fuertemente dependiente de la configuración del robot.

Una robot articulado consta de un conjunto de segmentos rígidos conectados mediante articulaciones. Los múltiples ángulos que pueden adoptar estas articulaciones permiten un número indefinido de configuraciones o posiciones de la figura. La solución al problema cinemático inverso consiste en encontrar los valores que deben adoptar las coordenadas articulares del robot $q = [q_1, q_2, \dots, q_n]$ para que su extremo se

posicione y oriente según una determinada configuración deseada. En general no existe una solución única para este problema, incluso puede no existir.

La resolución de éste problema puede dar lugar a múltiples soluciones (diferentes configuraciones articulares con las que obtener la misma configuración del efector final), puede que no exista solución (por ejemplo en una posición no alcanzable), o puede dar lugar a singularidades. Todo lo cual hace más difícil la resolución de este problema que la del problema cinemático directo.

En la figura 3.5a se ilustra de manera sencilla la existencia de soluciones múltiples para el problema cinemático inverso de un manipulador de dos GDL. Las soluciones 1 y 2 son comúnmente conocidas como codo arriba y codo abajo respectivamente.

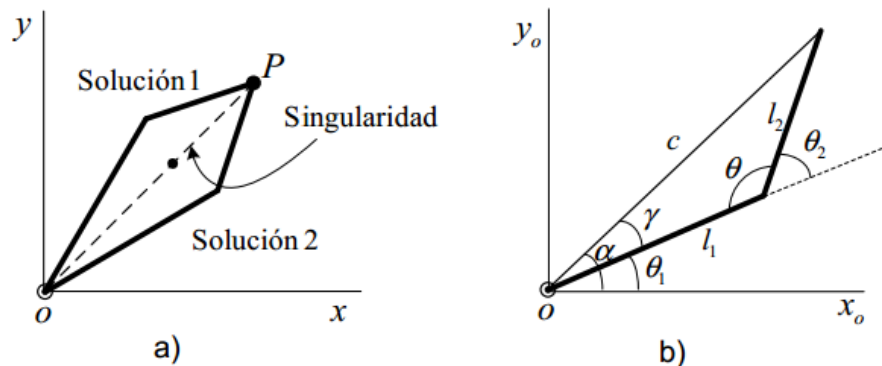


Figura 3.5 Problema cinemático inverso. a) Solución múltiple, b) codo abajo

La búsqueda de la solución suele realizarse mediante el uso de técnicas numéricas iterativas como por ejemplo Método de Newton. Esto puede resultar en cálculos lentos, por lo que habitualmente en una implementación real se acota el tiempo máximo (o iteraciones) que debe realizar el algoritmo de búsqueda.

En otros casos, para robots con pocos grados de libertad, existen soluciones analíticas mediante el uso de métodos geométricos, que consisten en la utilización de las relaciones trigonométricas y la resolución de los triángulos formados por los elementos y articulaciones del robot.

También puede ser el proceso de cálculo de la posición en el espacio del extremo de una estructura ligada, dados los ángulos de todas las articulaciones. Es fácil, y sólo hay una solución. Cinemática inversa hace lo contrario. Teniendo en cuenta el punto final de la estructura, lo que los ángulos de las articulaciones que necesitan de estar en el punto final que alcanzar. Puede ser difícil, y por lo general hay muchos o infinito de soluciones. Este proceso puede ser extremadamente útil en la robótica. Es posible que tenga un brazo robótico que tiene que agarrar un objeto. Si el software sabe dónde está el objeto en relación con el hombro, simplemente se necesita el cálculo de los ángulos de las articulaciones para llegar a él.

Los métodos usados para la solución del problema de la cinemática inversa son:

Métodos geométricos: este procedimiento es adecuado para pocos grados de libertad y se basa en encontrar suficientes relaciones geométricas en las que intervendrán las coordenadas del efector final del robot, sus coordenadas articulares y las dimensiones físicas de sus elementos.

Resolución a partir de las matrices de transformación homogénea: es posible tratar de obtener el modelo de cinemática inversa a partir del conocimiento de su modelo de cinemática directa. En la práctica esto no resulta trivial siendo en muchas ocasiones tan compleja que obliga a desecharla.

Desacoplo cinemático: consiste en la separación de orientación y posición, se utiliza en robots de 6 GDL. Los procedimientos anteriores de cinemática inversa permiten obtener los valores de las tres primeras articulaciones que posicionen su extremo en unas coordenadas determinadas y pueden ser utilizados para obtener los valores de las seis a costa de una mayor complejidad. En general, no basta con posicionar el extremo del robot en un punto en el espacio. Los robots suelen contar con tres grados de libertad adicionales, situados al final de la cadena cinemática y cuyos ejes, generalmente, se cortan en un punto que informalmente se denomina muñeca del robot.

El método del desacoplo cinemático separa ambos problemas: posición y orientación. Así, dada una posición y orientación final deseada se establece la posición del punto de corte (la muñeca del robot) calculando los valores de q_1 , q_2 y q_3 y a continuación a partir de los datos de orientación y los ya calculados se obtienen los valores del resto de las variables articulares.

Reducción polinómica: consiste en transformar las ecuaciones obtenidas algebraica o geoméricamente para que adopten una forma polinómica y se facilite su resolución.

4 CAPÍTULO IV

MODELACIÓN DEL ROBOT STANFORD CINEMÁTICA DIRECTA

En el problema de la cinemática directa se busca encontrar la ecuación que rige el movimiento del robot manipulador. Esta ecuación nos permite determinar la posición y la orientación del efector final en base a valores de las coordenadas articulares conocidos.

En la figura 4.1 se muestra la representación simbólica del Robot Stanford de seis grados de libertad. Las articulaciones de revolución son representadas con cilindros y las prismáticas con un cubo, los eslabones son representados únicamente con una línea que conecta a las articulaciones del robot manipulador. Este tipo de representación nos permitirá facilitar el análisis de los movimientos del robot, y agilizará el proceso de obtención de los parámetros de Denavit-Hartenberg tanto en el método estándar como en el modificado.

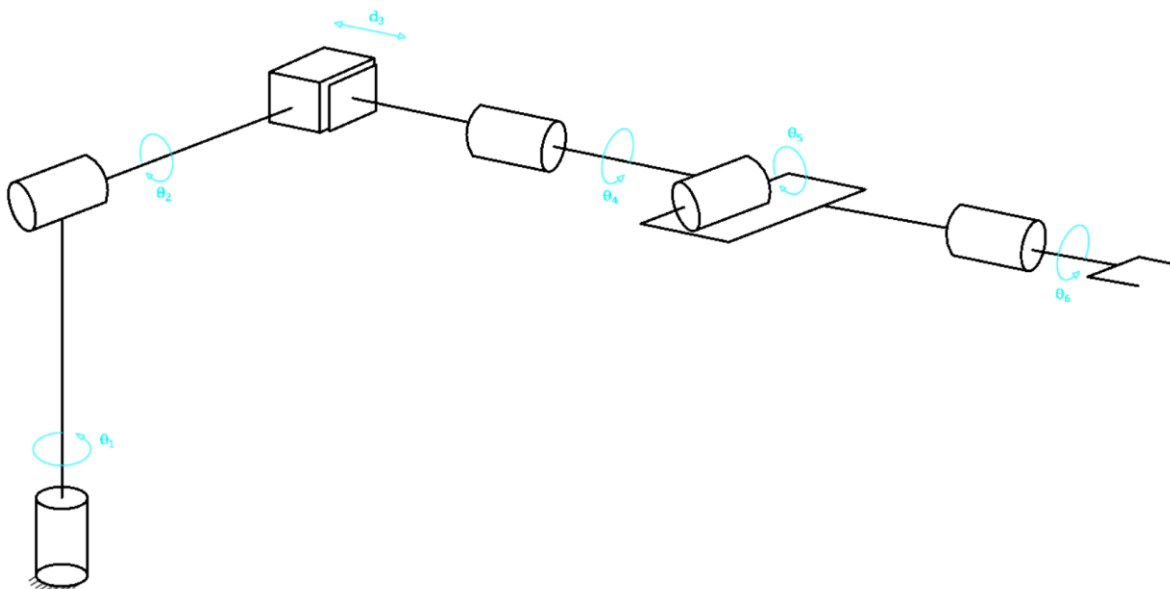


Figura 4.1 Representación simbólica del Robot Stanford Fuente:Propia

Se ha visto que en muchos libros existe cierta ambigüedad en la aplicación de los métodos de Denavit- Hartenberg estándar y modificado, por lo que se realizará la

modelación del Robot Stanford tanto con el método Denavit-Hartenberg estándar, así como también con el método Denavit-Hartenberg modificado. Por último se realizará la modelación del robot Stanford con el método del Movimiento General continuo, y así se podrá determinar y aclarar las diferencias entre los distintos métodos de modelación.

4.1 MÉTODO DE DENAVIT-HARTENBERG ESTÁNDAR, DHS.

En la figura 4.2 se muestra el manipulador Stanford, las articulaciones de revolución son representadas con un cilindro, mientras que la articulación prismática con un cubo, las líneas que unen a las articulaciones serían los ejes, los mismos que tienen un grado de libertad, así por ejemplo el Robot Stanford tiene 6 ejes y en consecuencia 6 grados de libertad, los ejes que salen de cada articulación de revolución tienen un movimiento rotatorio y la variable articular asociada con estos ejes es θ_i , el eje que sale de una articulación prismática tiene como variable articular d_i ; el número de variables de un brazo articular se puede determinar utilizando la fórmula de Kutzbach para el cálculo de los grados de libertad de un mecanismo en el espacio.

Resumiendo el procedimiento del algoritmo con base en la convención de DH para derivar la cinemática directa para cualquier manipulador, tenemos los siguientes pasos[7]:

1. Localizar y etiquetar los ejes de articulación z_0, \dots, z_{n-1} .
2. Establecer el sistema de coordenadas base. Establecer el origen en cualquier lugar en el eje z_0 . Los ejes x_0 y y_0 se eligen convenientemente para formar un sistema de coordenadas dextrógiro.

Para $i = 1, \dots, n - 1$, realice los pasos 3 a 5.

3. Localizar el origen o_i donde la normal común a z_i y z_{i-1} intersecta z_i . Si z_i intersecta z_{i-1} localice o_i en esta intersección. Si z_i y z_{i-1} son paralelos, localizar o_i en cualquier posición a lo largo de z_i .

4. Establecer x_i largo de la normal común entre z_{i-1} y z_i través o_i , o en la dirección normal al plano $z_{i-1} - z_i$ si z_{i-1} y z_i se cruzan.
5. Establecer y_i para completar un sistema de coordenadas dextrógiro.
6. Establecer el sistema de coordenadas del efector final $o_n x_n y_n z_n$. Asumiendo la articulación enésima es de rotación, establecer $n = a$ a lo largo de la dirección z_{i-1} . Establecer el origen o_n convenientemente a lo largo de z_n , preferiblemente en el centro de la pinza o en la punta de cualquier herramienta que el manipulador pueda llevar. Establecer $y_n = s$ en la dirección del cierre de la pinza y establecer $x_n = n$ como $s \times a$. Si la herramienta no es un pinza simple establezca x_n e y_n convenientemente para formar un sistema de coordenadas dextrógiro.
7. Crear una tabla de parámetros de los eslabones $a_i, d_i, \alpha_i, \theta_i$
 - a_i = distancia a lo largo de x_i desde o_i a la intersección de los ejes x_i y z_{i-1}
 - d_i = distancia a lo largo de z_{i-1} desde o_{i-1} hasta la intersección de los ejes x_i y z_{i-1} . d_i es la variable de la articulación prismática i .
 - α_i = ángulo entre z_{i-1} y z_i medido alrededor de x_i (ver Figura 3.3).
 - θ_i = ángulo entre x_{i-1} y x_i medido sobre z_{i-1} (ver Figura 3.3). θ_i es la variable de la articulación de rotación i .
8. Formar las matrices de transformación homogénea A_i sustituyendo los parámetros anteriores en (4.10).
9. Formar $T_n^0 = A_1 \dots A_n$. Esto da entonces la posición y orientación del sistema de coordenadas de la herramienta expresadas en coordenadas de la base.

En la figura 4.2 se observa la asignación de los sistemas de coordenadas sobre cada articulación utilizando el convenio de Denavit-Hartenberg estándar, según este método se tiene que utilizar 4 movimientos para pasar de un sistema de coordenadas al siguiente sistema, la secuencia de movimientos debe ser: rotación alrededor de z un ángulo θ_i , traslación a lo largo de z un valor d_i , traslación a lo largo de x un valor a_i y rotación alrededor de x α_i .

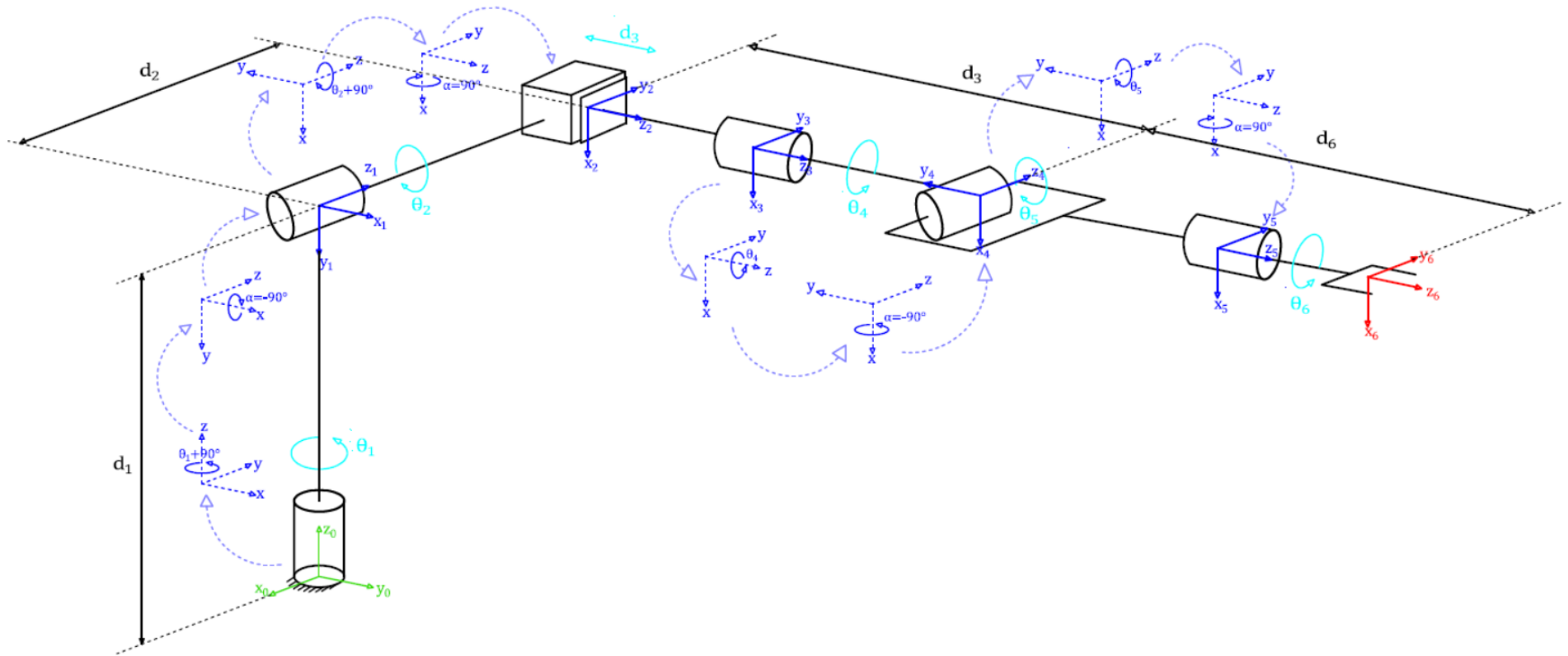


Figura 4.2 Asignación de sistemas de coordenadas por el método Denavit-Hartenberg estándar Fuente: Propia

En la figura 4.2 el sistema de coordenadas del origen está representado por color verde, mientras que el sistema de coordenadas del efector final por color rojo. Las variables articulares del Robot Stanford se encuentran representadas en la figura con color celeste y los sistema de coordenadas de las articulaciones con color azul. Además se utiliza líneas continuas para los sistemas de coordenadas que están sobre cada una de las articulaciones, los sistemas de coordenadas realizados con líneas de trazos nos indican los cuatro movimientos intermedios que nos permiten pasar de un sistema de coordenadas de una articulación al siguiente sistema de coordenadas de la articulación que está a continuación en la cadena cinemática.

En la tabla 4.1 se resume los movimientos que se realizaron. En robótica se le conoce a esto como la tabla de parámetros de D-H estándar, para ello se debe recalcar que la secuencia de movimientos primero es en z (rosca en z), luego en x (rosca en x)

Tabla 4.1 Parámetros D-H estándar para manipulador Stanford.

Eslabón	θ_i	d_i	a_i	α_i
1	$\theta_1^* + 90^\circ$	d_1	0	-90°
2	$\theta_2^* + 90^\circ$	d_2	0	90°
3	0	d_3^*	0	0°
4	θ_4^*	0	0	-90°
5	θ_5^*	0	0	90°
6	θ_6^*	d_6	0	0

*variable de la articulación

Utilizando la fórmula de la matriz de Denavit-Hartenberg estándar (4.1) se puede encontrar las matrices de transformación de coordenadas entre un sistema de coordenadas y el sistema de coordenadas contiguo.

$$A_i = R_{z,\theta_i} T_{z,d_i} T_{x,a_i} R_{x,\alpha_i}$$

$$A_i = \begin{bmatrix} c\theta_i & s\theta_i & 0 & 0 \\ s\theta_i & c\theta_i & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & a_i \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & c\alpha_i & -s\alpha_i & 0 \\ 0 & s\alpha_i & c\alpha_i & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$A_i = \begin{bmatrix} c\theta_i & -s\theta_i c\alpha_i & s\theta_i s\alpha_i & a_i c\theta_i \\ s\theta_i & c\theta_i c\alpha_i & -c\theta_i s\alpha_i & a_i s\theta_i \\ 0 & s\alpha_i & c\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4.1)$$

Las matrices de transformación de coordenadas para el robot Stanford se presentan a continuación, nótese que cada fila de la tabla de Denavit-Hartenberg nos produce una matriz de transformación de coordenadas, si la tabla tiene 6 filas esto implica 6 grados de libertad, entonces tenemos 6 matrices de transformación de coordenadas

$$A_1 = \begin{bmatrix} c_{1+90} & 0 & -s_{1+90} & 0 \\ s_{1+90} & 0 & c_{1+90} & 0 \\ 0 & -1 & 0 & d_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} -s_1 & 0 & -c_1 & 0 \\ c_1 & 0 & -s_1 & 0 \\ 0 & -1 & 0 & d_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4.2)$$

$$A_2 = \begin{bmatrix} c_{2+90} & 0 & s_{2+90} & 0 \\ s_{2+90} & 0 & -c_{2+90} & 0 \\ 0 & -1 & 0 & d_2 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} -s_2 & 0 & -c_2 & 0 \\ c_2 & 0 & -s_2 & 0 \\ 0 & 1 & 0 & d_2 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4.3)$$

$$A_3 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_3 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4.4)$$

$$A_4 = \begin{bmatrix} c_4 & 0 & -s_4 & 0 \\ s_4 & 0 & c_4 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4.5)$$

$$A_5 = \begin{bmatrix} c_5 & 0 & s_5 & 0 \\ s_5 & 0 & -c_5 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4.6)$$

$$A_6 = \begin{bmatrix} c_6 & -s_6 & 0 & 0 \\ s_6 & c_6 & 0 & 0 \\ 0 & 0 & 1 & d_6 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4.7)$$

Es costumbre en robótica para facilitar la escritura, escribir de manera más compacta a las funciones trigonométricas simplemente con subíndices numéricos que representan a las variables articulares θ de manera implícita, así por ejemplo, C_1 significa $\cos(\theta_1)$, de manera semejante el resto de términos trigonométricos.

La matriz de transformación de coordenadas total del sistema desde el origen de coordenadas hasta el sistema de coordenadas del efector final está dada por la multiplicación de todas las matrices anteriores, entonces tenemos que:

$$T_6^0 = A_1 * A_2 * A_3 * A_4 * A_5 * A_6 \quad (4.8)$$

$$T_6^0 = \begin{bmatrix} r_{11} & r_{12} & r_{13} & d_x \\ r_{21} & r_{22} & r_{23} & d_y \\ r_{31} & r_{32} & r_{33} & d_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4.9)$$

Donde,

$$r_{11} = -C_6(C_5(C_1S_4 - C_4S_1S_2) - C_2S_1S_5) - S_6(C_1C_4 + S_1S_2S_4)$$

$$r_{21} = -C_6(C_5(S_1S_4 + C_1C_4S_2) + C_1C_2S_5) - S_6(C_4S_1 - C_1S_2S_4)$$

$$r_{31} = C_6(S_2S_5 - C_2C_4C_5) + C_2S_4S_6$$

$$r_{12} = S_6(C_5(C_1S_4 - C_4S_1S_2) - C_2S_1S_5) - C_6(C_1C_4 + S_1S_2S_4)$$

$$r_{22} = S_6(C_5(S_1S_4 + C_1C_4S_2) + C_1C_2S_5) - C_6(C_4S_1 - C_1S_2S_4)$$

$$r_{32} = C_2C_6S_4 - S_6(S_2S_5 - C_2C_4C_5)$$

$$r_{13} = -S_5(C_1S_4 - C_4S_1S_2) - C_2C_5S_1$$

$$r_{23} = C_1C_2C_5 - S_5(S_1S_4 + C_1C_4S_2)$$

$$r_{33} = -C_5S_2 - C_2C_4S_5$$

$$d_x = -D_2C_1 - D_6(S_5(C_1S_4 - C_4S_1S_2) + C_2C_5S_1) - D_3C_2S_1$$

$$d_y = D_3C_1C_2 - D_6(S_5(S_1S_4 + C_1C_4S_2) - C_1C_2C_5) - D_2S_1$$

$$d_z = D_1 - D_6(C_5S_2 + C_2C_4S_5) - D_3S_2$$

Los resultados fueron obtenidos mediante el software Matlab, que como su nombre lo indica Matlab, que es la abreviatura de MATrix LABoratory, se caracteriza por ser una herramienta altamente eficiente y muy útil para aplicaciones que requieran la manipulación de matrices, como es nuestro caso.

La programación realizada en Matlab, para el desarrollo del modelo del Robot Stanford con el método de Denavit-Hartenberg estándar se presenta en la figura 4.3

Para encontrar las matrices de transformación de coordenadas entre un sistema de coordenadas y el sistema de coordenadas siguiente se utiliza la fórmula de Denavit-Hartenberg estándar: $A_i = R_{z,\theta_i} T_{z,d_i} T_{x,a_i} R_{x,\alpha_i}$.

Así mismo se encontrará la matriz de transformación homogénea del robot manipulador Stanford, es decir la matriz de transformación del último sistema de coordenadas ubicado en el extremo final del robot con respecto al sistema de coordenadas del origen ubicado en la base del robot, esto se consigue multiplicando cada una de las matrices de transformación de coordenadas A_i que se obtienen aplicando la fórmula de Denavit-Hartenberg estándar en cada fila de la tabla de parámetros; el número de matrices A_i es igual al número de filas de la tabla o lo que es lo mismo igual al número de grados de libertad del mecanismo.

Se debe remarcar que para obtener el resultado de las matrices de transformación homogénea con Matlab, es necesario realizar previamente la asignación de los sistemas de coordenadas y luego elaborar la tabla de parámetros D-H estándar, ya que estos datos serán ingresados al software, como se puede ver en la figura 4.3

```

1      % Robot stanford D-Hs
2
3      syms t1 t2 D3 t4 t5 t6          % variables articulares
4      syms a1 D1 D2 a2 a4 a5 D6      % constantes
5
6      a1=-pi/2
7      a2=pi/2
8      a4=-pi/2
9      a5=pi/2
10
11     A1=rotz(t1+pi/2)*transl(0,0,D1)*rotx(a1) % ecuacion 3.1
12
13     A2=rotz(t2+pi/2)*transl(0,0,D2)*rotx(a2)
14
15     A4=rotz(t4)*rotx(a4)
16
17     A5=rotz(t5)*rotx(a5)
18
19     A6=rotz(t6)*transl(0,0,D6)
20
21     T06=simple(A1*A2*A3*A4*A5*A6)

```

Figura 4.3 Programación en Matlab para Modelación del Robot Stanford, D-H estándar

En la figura 4.4 se muestran los resultados que nos entrega el programa, es decir las matrices de transformación A_i , las mismas que son necesarias para la obtención de la matriz de transformación homogénea del robot manipulador, la misma que se indica a través de la ecuación 4.9.

Las matrices de transformación de coordenadas A_1, A_2, A_3, A_4, A_5 y A_6 se obtuvieron utilizando la fórmula D-H estándar, a su vez multiplicando a todas estas matrices en la misma secuencia que de los movimientos nos permite determinar la matriz resultante del robot manipulador, en los libros se le simboliza con la letra T en nuestro caso sería $T_6^0 = A_1 A_2 A_3 A_4 A_5 A_6$. Algunos autores representan a la matriz de T_6^0 con una nueva letra H . Resumiendo $H = T_6^0 = A_1 A_2 A_3 A_4 A_5 A_6$

```

A1 =

[ cos(pi/2 + t1), 0, -sin(1/2*pi + t1), 0]
[ sin(pi/2 + t1), 0,  cos(pi/2 + t1), 0]
[          0, -1,          0, D1]
[          0, 0,          0, 1]

A2 =

[ cos(pi/2 + t2), 0,  sin(pi/2 + t2), 0]
[ sin(pi/2 + t2), 0, -cos(1/2*pi + t2), 0]
[          0, 1,          0, D2]
[          0, 0,          0, 1]

A3 =

[ 1, 0, 0, 0]
[ 0, 1, 0, 0]
[ 0, 0, 1, D3]
[ 0, 0, 0, 1]

A4 =

[ cos(t4), 0, -sin(t4), 0]
[ sin(t4), 0,  cos(t4), 0]
[          0, -1,          0, 0]
[          0, 0,          0, 1]

A5 =

[ cos(t5), 0,  sin(t5), 0]
[ sin(t5), 0, -cos(t5), 0]
[          0, 1,          0, 0]
[          0, 0,          0, 1]

A6 =

[ cos(t6), -sin(t6), 0, 0]
[ sin(t6),  cos(t6), 0, 0]
[          0,          0, 1, D6]
[          0,          0, 0, 1]

```

Figura 4.4 Matrices de transformación A_i obtenidas con el software Matlab

4.2 MÉTODO DE DENAVIT-HARTENBERG MODIFICADO, DHM.

En la figura 4.5 se observa la asignación de los sistemas de coordenadas sobre cada articulación utilizando el convenio de Denavit-Hartenberg modificado. Al igual que en el método de Denavit-Hartenberg estándar, en este método se tiene que utilizar grupos de cuatro movimientos para pasar de un sistema de coordenadas al siguiente sistema, con la diferencia que la secuencia de movimientos debe ser: rotación alrededor de x α_i , traslación a lo largo de x un valor a_i , rotación alrededor de z un ángulo θ_i , traslación a lo largo de z un valor d_i , en otras palabras primero rosca en x y luego rosca en z , al escribir estos movimientos en una tabla nos produce la tabla de parámetros de Denavit-Hartenberg modificado.

En la figura 4.5 se muestran los sistemas de coordenados con líneas continuas sobre las articulaciones, mientras que los sistemas de coordenada intermedios o temporales que pueden ser hasta cuatro se indican con líneas de trazos.

En el método de Denavit-Hartenberg modificado al asignar los sistemas de coordenadas primero sobre el eje x y luego sobre el eje z puede requerir de cuatro movimientos, este grupo de movimientos se escribe en una fila de la tabla de parámetros de Denavit-Hartenberg modificado, el siguiente grupo de cuatro movimientos que se pueden requerir realizar al sistema de coordenadas móvil para avanzar a la siguiente articulación se escriben en la siguiente fila de la mencionada tabla ya sí sucesivamente hasta llegar al extremo del robot manipulador o cadena cinemática abierta.

El número de filas en tabla de parámetros coincide con el número de articulaciones o lo que es lo mismo grados de libertad del mecanismo espacial.

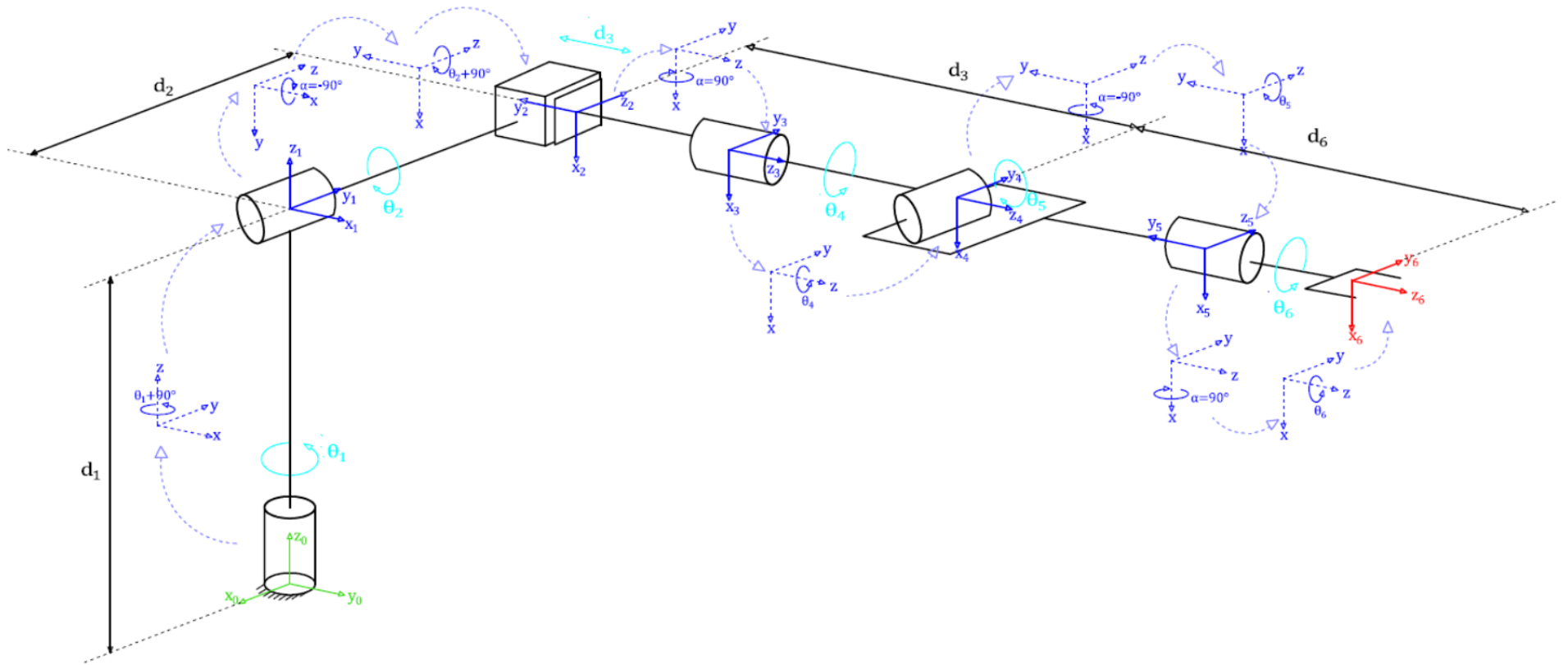


Figura 4.5 Asignación de sistemas de coordenadas por el método Denavit-Hartenberg modificado Fuente: Propia

En la tabla 4.2 se presentan los parámetros de Denavit-Hartenberg modificado, que no es más que el resumen de los movimientos que se realizaron en grupos de cuatro.

Tabla 4.2 Parámetros D-H modificado para manipulador Stanford.

Eslabón	α_i	a_i	θ_i	d_i
1	0	0	$\theta_1^* + 90^\circ$	d_1
2	-90°	0	$\theta_2^* + 90^\circ$	d_2
3	90°	0	θ	d_3^*
4	0	θ	θ_4^*	0
5	-90°	θ	θ_5^*	0
6	90°	0	θ_6^*	d_6

*variable de la articulación

Utilizando la fórmula de Denavit-Hartenberg modificada (4.10) se puede encontrar las matrices de transformación de coordenadas entre un sistema de coordenadas y el siguiente sistema de coordenadas.

$$A_i = R_{x_{i-1}, \alpha_{i-1}} \cdot T_{x_{i-1}, a_{i-1}} \cdot R_{z, \theta_i} \cdot T_{z, d_i}$$

$$A_i = \begin{bmatrix} c\theta_i & -s\theta_i & 0 & a_{i-1} \\ s\theta_i c\alpha_{i-1} & c\theta_i c\alpha_{i-1} & -s\alpha_{i-1} & s\alpha_{i-1} d_i \\ s\theta_i s\alpha_{i-1} & c\theta_i s\alpha_{i-1} & c\alpha_{i-1} & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4.10)$$

Las matrices de transformación de coordenadas A_i utilizando el convenio Denavit-Hartenberg modificado para el robot Stanford se indican a continuación:

$$A_1 = \begin{bmatrix} c_{1+90} & -s_{1+90} & 0 & 0 \\ s_{1+90} & c_{1+90} & 0 & 0 \\ 0 & 0 & 1 & d_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} -s_1 & -c_1 & 0 & 0 \\ c_1 & -s_1 & 0 & 0 \\ 0 & 0 & 1 & d_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4.11)$$

$$A_2 = \begin{bmatrix} c_{2+90} & -s_{2+90} & 0 & 0 \\ 0 & 0 & 1 & d_2 \\ -s_{2+90} & -c_{2+90} & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} -s_2 & -c_2 & 0 & 0 \\ 0 & 0 & 1 & d_2 \\ -c_2 & s_2 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4.12)$$

$$A_3 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & -1 & -d_3 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4.13)$$

$$A_4 = \begin{bmatrix} c_4 & -s_4 & 0 & 0 \\ s_4 & c_4 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4.14)$$

$$A_5 = \begin{bmatrix} c_5 & -s_5 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -s_5 & -c_5 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4.15)$$

$$A_6 = \begin{bmatrix} c_6 & -s_6 & 0 & 0 \\ 0 & 0 & -1 & -d_6 \\ s_6 & c_6 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4.16)$$

Podemos encontrar la matriz de transformación de coordenadas resultante del brazo de robot ó matriz de transformación homogénea H , multiplicando todas las matrices de transformación de coordenadas A_i

$$H = T_6^0 = A_1 * A_2 * A_3 * A_4 * A_5 * A_6 \quad (4.17)$$

$$T_6^0 = \begin{bmatrix} r_{11} & r_{12} & r_{13} & d_x \\ r_{21} & r_{22} & r_{23} & d_y \\ r_{31} & r_{32} & r_{33} & d_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4.18)$$

Donde,

$$r_{11} = -C_6(C_5(C_1S_4 - C_4S_1S_2) - C_2S_1S_5) - S_6(C_1C_4 + S_1S_2S_4)$$

$$r_{21} = -C_6(C_5(S_1S_4 + C_1C_4S_2) + C_1C_2S_5) - S_6(C_4S_1 - C_1S_2S_4)$$

$$r_{31} = C_6(S_2S_5 - C_2C_4C_5) + C_2S_4S_6$$

$$r_{12} = S_6(C_5(C_1S_4 - C_4S_1S_2) - C_2S_1S_5) - C_6(C_1C_4 + S_1S_2S_4)$$

$$r_{22} = S_6(C_5(S_1S_4 + C_1C_4S_2) + C_1C_2S_5) - C_6(C_4S_1 - C_1S_2S_4)$$

$$r_{32} = C_2C_6S_4 - S_6(S_2S_5 - C_2C_4C_5)$$

$$r_{13} = -S_5(C_1S_4 - C_4S_1S_2) - C_2C_5S_1$$

$$r_{23} = C_1C_2C_5 - S_5(S_1S_4 + C_1C_4S_2)$$

$$r_{33} = -C_5S_2 - C_2C_4S_5$$

$$d_x = -D_2C_1 - D_6(S_5(C_1S_4 - C_4S_1S_2) + C_2C_5S_1) - D_3C_2S_1$$

$$d_y = D_3C_1C_2 - D_6(S_5(S_1S_4 + C_1C_4S_2) - C_1C_2C_5) - D_2S_1$$

$$d_z = D_1 - D_6(C_5S_2 + C_2C_4S_5) - D_3S_2$$

De igual forma que con el método de D-H estándar, las matrices de transformación A_i y la matriz homogénea para el Robot Stanford con el método de D-H modificado fueron determinadas utilizando el software Matlab, posteriormente al trabajo de asignación de los sistemas de coordenadas y elaboración de la tabla de parámetros D-H modificado.

Como ya se mencionó, el orden de los movimientos cambia de un método a otro, por lo que también cambia la fórmula para encontrar las matrices de transformación de coordenadas entre un sistema de coordenadas y el siguiente sistema de coordenadas.

$$A_i = R_{x_{i-1}, \alpha_{i-1}} \cdot T_{x_{i-1}, a_{i-1}} \cdot R_{z, \theta_i} \cdot T_{z, d_i}$$

Empleando la fórmula con los parámetros de una fila de la tabla D-H modificado se calculó la matriz A_i , de manera similar se calculan todas las otras matrices A_i para cada fila de la tabla.

La matriz de transformación homogénea resultante o también conocida como la matriz del robot manipulador se determina multiplicando a todas las matrices A_i en la misma secuencia a través de la fórmula:

$$H = T_6^0 = A_1 * A_2 * A_3 * A_4 * A_5 * A_6$$

En la figura 4.6 se puede observar la programación realizada en Matlab para la obtención del modelo del Robot Stanford utilizando el método D-H modificado.

```

Editor - C:\Users\Vicky\Documents\MATLAB\MSc matlab\c3_manstanford_modificado.m*
1      % Robot stanford D-Hm
2
3      syms t1 t2 D3 t4 t5 t6          % variables articulares
4      syms a1 D1 D2 a2 a3 a4 a5 D6   % constantes
5
6      a1=0;
7      a2=-pi/2;
8      a3=pi/2;
9      a5=-pi/2;
10     a6=pi/2;
11
12     A1=rotz(t1+pi/2)*transl(0,0,D1)
13
14     A2=rotx(a2)*rotz(t2+pi/2)*transl(0,0,D2)
15
16     A3=rotx(a3)*transl(0,0,D3)
17
18     A4=rotz(t4)
19
20     A5=rotx(a5)*rotz(t5)
21
22     A6=rotx(a6)*rotz(t6)*transl(0,0,D6)
23
24     T06=simple(A1*A2*A3*A4*A5*A6)

```

Figura 4.6 Programación en Matlab para Modelación del Robot Stanford,D-H modificado

En la figura 4.7 se presentan las mencionadas matrices de transformación A_i proporcionadas por el programa, que son la base para el cálculo de la matriz de transformación homogénea del robot, ver ecuación 4.18

```

A1 =

[ cos(pi/2 + t1), -sin(1/2*pi + t1), 0, 0]
[ sin(pi/2 + t1),   cos(pi/2 + t1), 0, 0]
[           0,           0, 1, D1]
[           0,           0, 0, 1]

A2 =

[   cos(pi/2 + t2), -sin(1/2*pi + t2), 0, 0]
[           0,           0, 1, D2]
[ -sin(1/2*pi + t2), -cos(1/2*pi + t2), 0, 0]
[           0,           0, 0, 1]

A3 =

[ 1, 0, 0, 0]
[ 0, 0, -1, -D3]
[ 0, 1, 0, 0]
[ 0, 0, 0, 1]

A4 =

[ cos(t4), -sin(t4), 0, 0]
[ sin(t4),  cos(t4), 0, 0]
[   0,   0, 1, 0]
[   0,   0, 0, 1]

A5 =

[  cos(t5), -sin(t5), 0, 0]
[   0,   0, 1, 0]
[ -sin(t5), -cos(t5), 0, 0]
[   0,   0, 0, 1]

A6 =

[ cos(t6), -sin(t6), 0, 0]
[   0,   0, -1, -D6]
[ sin(t6),  cos(t6), 0, 0]
[   0,   0, 0, 1]

```

Figura 4.7 Matrices de transformación A_i obtenidas con el software Matlab

Como podemos apreciar, tanto en el método Denavit-Hartenberg estándar como en el modificado todos los eslabones y las articulaciones se numeran de forma ascendente siguiendo la cadena cinemática, en los eslabones la numeración comienza con cero en la base y n en el último eslabón; en las articulaciones la numeración inicia en 1 en la primera articulación y $n-1$ en la última. Otra semejanza es que los dos métodos agrupan los movimientos en grupos de cuatro, los mismos que se representan en una fila de la tabla de parámetros D-H.

La diferencia entre los dos métodos radica principalmente en la secuencia de movimientos que se realizan para la asignación de sistemas de coordenadas de articulación a articulación, en el método D-H estándar la secuencia de movimientos son rotación en z (ángulo θ), traslación en z (distancia d), traslación en x (distancia a) y rotación en x (ángulo α), mientras que en el método D-H modificado, la secuencia de movimientos son rotación en x y traslación en x , rotación en z y traslación en z .

Además como podemos apreciar en las figuras 4.2 y 4.5, en el método de Denavit-Hartenberg estándar el origen del sistema i está a lo largo del eje de la articulación $i+1$, mientras que en el modificado el origen del sistema i está a lo largo del eje de la articulación i , ver figura 3.3

4.3 MÉTODO DEL MOVIMIENTO GENERAL CONTINUO

El método del movimiento general continuo rompe los esquemas de los métodos tradicionales para la modelación de robots ya que se asigna los sistemas de coordenadas a los eslabones, siguiendo la cadena cinemática de forma continua sin formar grupos de cuatro movimientos como establecen los dos métodos anteriores.

En la figura 4.8 se muestra el esquema de la asignación de los sistemas de coordenadas para la modelación del robot Stanford utilizando el método del movimiento general continuo sin utilizar movimientos en el eje 'y' como establece el convenio de Denavit-Hartenberg.

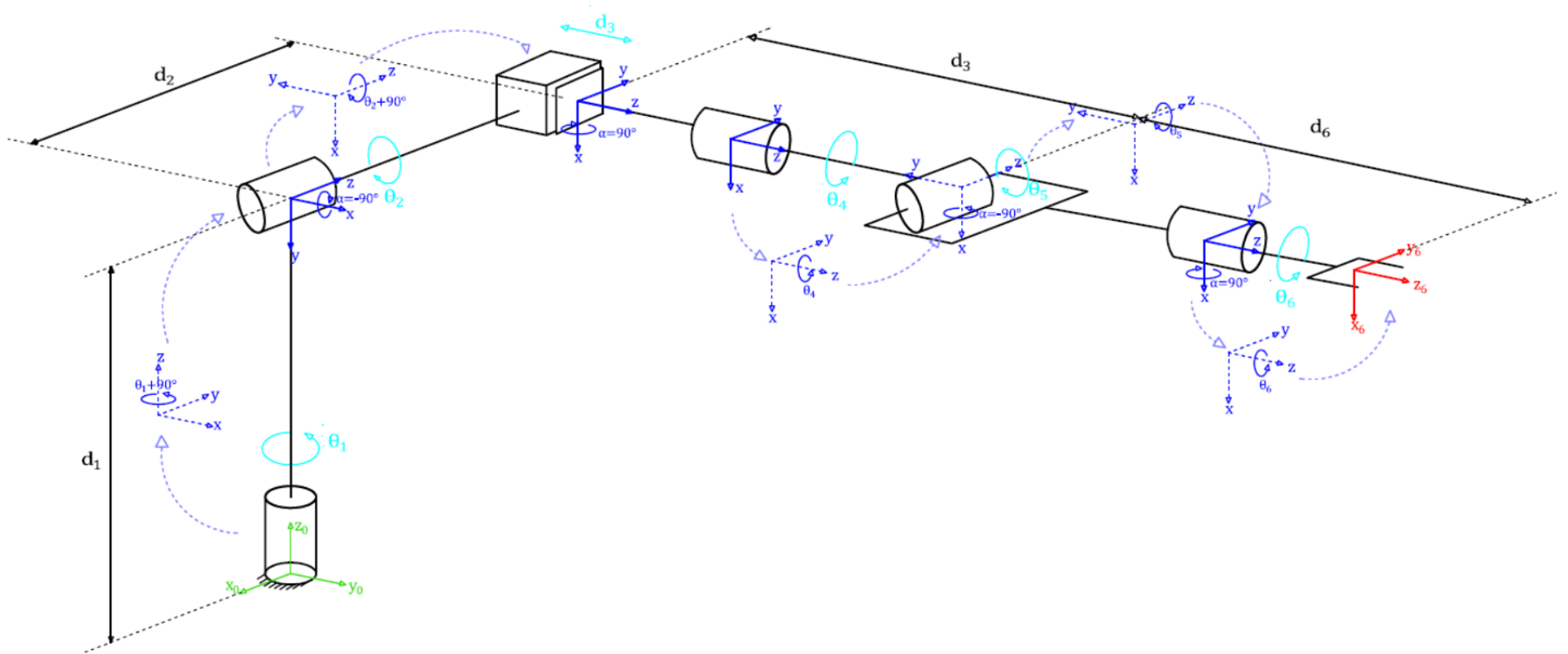


Figura 4.8 Asignación de sistemas de coordenadas por el método del movimiento general continuo sin utilizar movimientos en el eje y Fuente: Propia

En base a la figura 4.8 podemos establecer la secuencia del movimiento general continuo del Robot Stanford, como se muestra en la tabla 4.3.

Tabla 4.3 Parámetros del Movimiento General Continuo para Robot Stanford

R_{z,θ_1}	T_{z,d_1}	R_{x,a_1}	R_{z,θ_2}	T_{z,d_2}	R_{x,a_3}	T_{z,d_3}	R_{z,θ_4}	R_{x,a_5}	R_{z,θ_5}	R_{x,a_6}	R_{z,θ_6}
$\theta_1 + 90^\circ$	d_1	$\alpha = -90^\circ$	$\theta_2 + 90^\circ$	d_2	$\alpha = 90^\circ$	d_3	θ_4	$\alpha = -90^\circ$	θ_5	$\alpha = 90^\circ$	θ_6

Utilizando el principio de la post-multiplicación matricial, en la que el producto de matrices debe realizarse en el mismo orden de los movimientos, en consecuencia la matriz de transformación homogénea del robot Stanford viene dado por:

$$H = R_{z,\theta_1} T_{z,d_1} R_{x,a_1} R_{z,\theta_2} T_{z,d_2} R_{x,a_3} T_{z,d_3} R_{z,\theta_4} R_{x,a_5} R_{z,\theta_5} R_{x,a_6} R_{z,\theta_6}$$

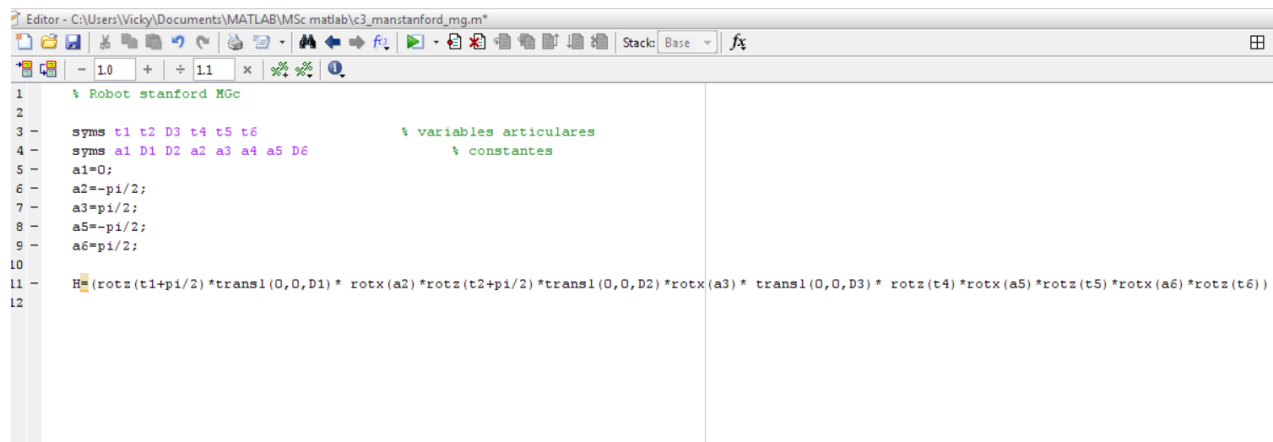
$$H = \begin{bmatrix} r_{11} & r_{12} & r_{13} & d_x \\ r_{21} & r_{22} & r_{23} & d_y \\ r_{31} & r_{32} & r_{33} & d_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4.19)$$

Donde,

$$\begin{aligned} r_{11} &= -C_6(C_5(C_1S_4 - C_4S_1S_2) - C_2S_1S_5) - S_6(C_1C_4 + S_1S_2S_4) \\ r_{21} &= -C_6(C_5(S_1S_4 + C_1C_4S_2) + C_1C_2S_5) - S_6(C_4S_1 - C_1S_2S_4) \\ r_{31} &= C_6(S_2S_5 - C_2C_4C_5) + C_2S_4S_6 \\ r_{12} &= S_6(C_5(C_1S_4 - C_4S_1S_2) - C_2S_1S_5) - C_6(C_1C_4 + S_1S_2S_4) \\ r_{22} &= S_6(C_5(S_1S_4 + C_1C_4S_2) + C_1C_2S_5) - C_6(C_4S_1 - C_1S_2S_4) \\ r_{32} &= C_2C_6S_4 - S_6(S_2S_5 - C_2C_4C_5) \\ r_{13} &= -S_5(C_1S_4 - C_4S_1S_2) - C_2C_5S_1 \\ r_{23} &= C_1C_2C_5 - S_5(S_1S_4 + C_1C_4S_2) \\ r_{33} &= -C_5S_2 - C_2C_4S_5 \\ d_x &= -D_2C_1 - D_6(S_5(C_1S_4 - C_4S_1S_2) + C_2C_5S_1) - D_3C_2S_1 \\ d_y &= D_3C_1C_2 - D_6(S_5(S_1S_4 + C_1C_4S_2) - C_1C_2C_5) - D_2S_1 \\ d_z &= D_1 - D_6(C_5S_2 + C_2C_4S_5) - D_3S_2 \end{aligned}$$

Comparando los resultados obtenidos en los tres métodos anteriores se observa que son idénticos por lo tanto son válidos.

En la figura 4.9 se presenta la programación realizada en Matlab para obtener la matriz homogénea del Robot Stanford con el método del movimiento general continuo sin utilizar movimientos en el eje y.



```

Editor - C:\Users\Vicky\Documents\MATLAB\MSc matlab\c3_manstanford_mgc.m*
1 % Robot stanford MGc
2
3 syms t1 t2 D3 t4 t5 t6           % variables articulares
4 syms a1 D1 D2 a2 a3 a4 a5 D6     % constantes
5 a1=0;
6 a2=-pi/2;
7 a3=pi/2;
8 a5=-pi/2;
9 a6=pi/2;
10
11 H=(rotz(t1+pi/2)*transl(0,0,D1)* rotx(a2)*rotz(t2+pi/2)*transl(0,0,D2)*rotx(a3)* transl(0,0,D3)* rotz(t4)*rotx(a5)*rotz(t5)*rotx(a6)*rotz(t6))
12

```

Figura 4.9 Programación en Matlab para Modelación del Robot Stanford mediante MGc

De lo anteriormente expuesto se desprende que para calcular la matriz homogénea del robot ya no tendremos que agrupar los movimientos en grupos de cuatro, y por consiguiente tampoco tendremos que calcular las matrices de transformación A_i ; sino simplemente multiplicamos las matrices en el mismo orden de los movimientos realizados, lo que simplifica en gran medida el cálculo para la obtención del modelo.

A continuación vamos a flexibilizar aún más el método, para lo cual vamos a utilizar todos los movimientos que se realicen al sistema de coordenadas móvil, siguiendo la cadena cinemática del mecanismo, incluyendo movimientos en el eje y.

En la figura 4.11 se muestra la asignación de coordenadas en las articulaciones del Robot Stanford aplicando el método del movimiento general continuo. Se aplicarán los movimientos en el orden que sean requeridos.

La simbología utilizada para traslaciones y rotaciones de los sistemas de coordenadas será la misma que la aplicada en los demás métodos, es decir θ para rotación en z, d para traslación en z, α para rotación en x y a para traslación en x, pero además se utilizará una nueva simbología para los movimientos realizados respecto al eje y, esta será Φ para rotación alrededor de eje y y b para traslación a lo largo del mismo.

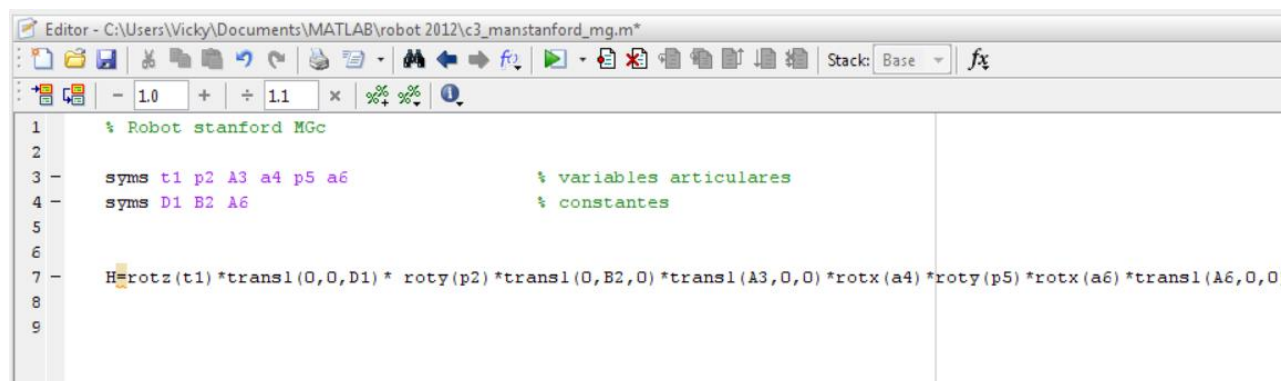
Podemos apreciar que al aplicar el método del movimiento general continuo sin ninguna restricción, la asignación de coordenadas se simplifica sobremanera, y por lo tanto también la obtención de los parámetros MGc para la modelación del Robot.

En base a la figura 4.11 establecemos la secuencia del movimiento general continuo del Robot Stanford, como se resume en la tabla 4.4.

Tabla 4.4 Parámetros del Movimiento General Continuo para Robot Stanford

R_{z,θ_1}	T_{z,d_1}	R_{y,Φ_2}	T_{y,b_2}	T_{x,a_3}	R_{x,α_4}	R_{y,Φ_5}	R_{x,α_6}	T_{x,a_6}
θ_1	d_1	Φ_2	b_2	a_3	α_4	Φ_5	α_6	a_6

Entonces empleamos nuevamente el principio de la post-multiplicación matricial, para encontrar la matriz de transformación homogénea del robot manipulador mediante el software Matlab, como se muestra en la figura 4.10.



```

1  % Robot stanford MGc
2
3  syms t1 p2 A3 a4 p5 a6           % variables articulares
4  syms D1 B2 A6                   % constantes
5
6
7  H=rotz(t1)*transl(0,0,D1)* rotz(p2)*transl(0,B2,0)*transl(A3,0,0)*rotx(a4)*roty(p5)*rotx(a6)*transl(A6,0,0)
8
9

```

Figura 4.10 Programación en Matlab para Modelación del Robot Stanford mediante MGc, con movimientos en cualquier eje

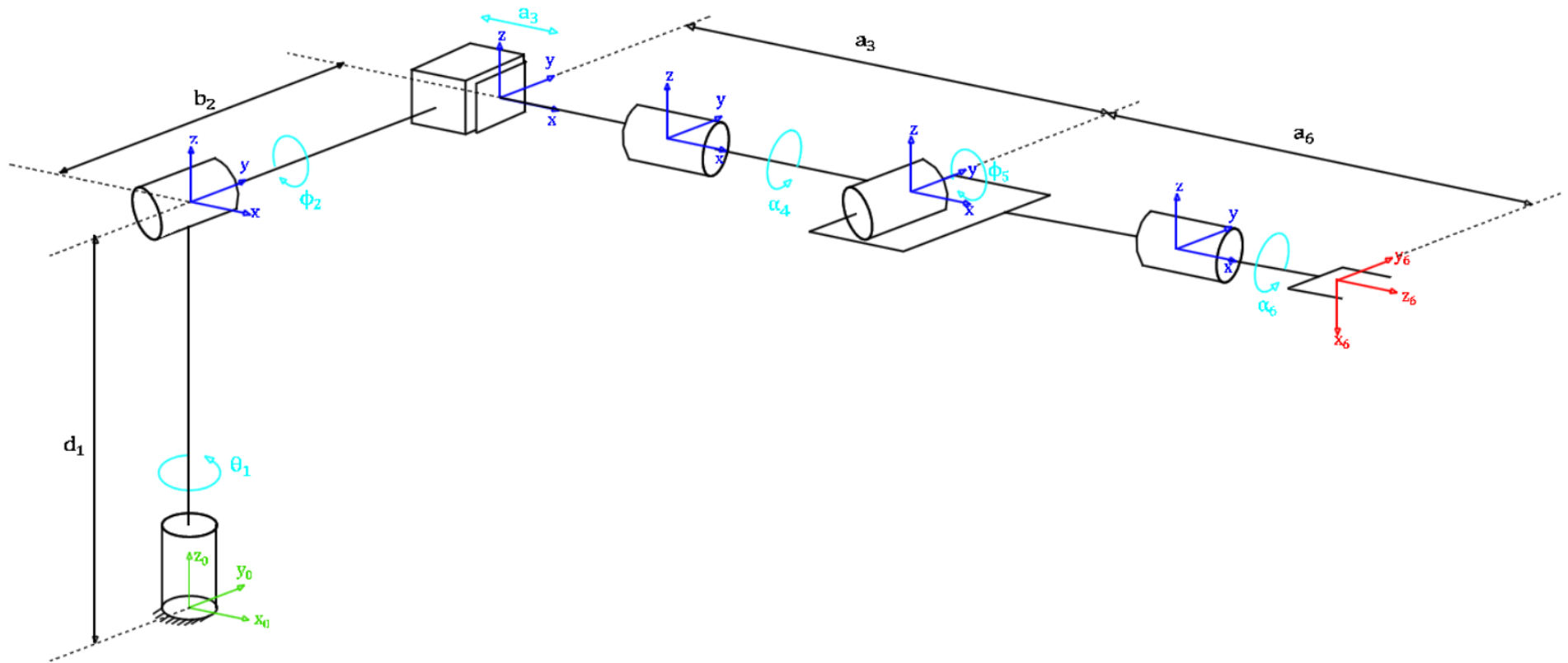


Figura 4.11 Asignación de sistemas de coordenadas por el método del movimiento general continuo (movimiento libre)
Fuente: Propia

Entonces vemos que también se simplifica la programación necesaria para la modelación del robot manipulador. El programa nos entrega directamente la matriz homogénea de transformación para el Robot Stanford. Se muestran a continuación cada uno de los elementos de la matriz resultante H.

$$H = \begin{bmatrix} r_{11} & r_{12} & r_{13} & d_x \\ r_{21} & r_{22} & r_{23} & d_y \\ r_{31} & r_{32} & r_{33} & d_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Donde,

$$r_{11} = C_{\phi_2} C_{\phi_5} C_{\theta_1} - S_{\phi_5} (S_{\alpha_4} S_{\theta_1} + C_{\alpha_4} C_{\theta_1} S_{\phi_2})$$

$$r_{21} = S_{\phi_5} S_{\alpha_4} C_{\theta_1} - C_{\alpha_4} S_{\phi_2} S_{\theta_1} + C_{\phi_2} C_{\phi_5} S_{\theta_1}$$

$$r_{31} = -C_{\phi_5} S_{\phi_2} - C_{\alpha_4} C_{\phi_2} S_{\phi_5}$$

$$r_{12} = S_{\alpha_6} (C_{\phi_5} (S_{\alpha_4} S_{\theta_1} + C_{\alpha_4} C_{\theta_1} S_{\phi_2}) + C_{\phi_2} C_{\theta_1} S_{\phi_5}) - C_{\alpha_6} (C_{\alpha_4} S_{\theta_1} - S_{\alpha_4} C_{\theta_1} S_{\phi_2})$$

$$r_{22} = -C_{\alpha_6} (C_{\phi_5} (S_{\alpha_4} C_{\theta_1} - C_{\alpha_4} S_{\phi_2} S_{\theta_1}) - C_{\phi_2} S_{\phi_5} S_{\theta_1}) - S_{\alpha_6} (C_{\alpha_4} C_{\theta_1} + S_{\alpha_4} S_{\phi_2} S_{\theta_1})$$

$$r_{32} = C_{\alpha_6} C_{\phi_2} S_{\alpha_4} - S_{\alpha_6} (S_{\phi_2} S_{\phi_5} - C_{\alpha_4} C_{\phi_2} C_{\phi_5})$$

$$r_{13} = S_{\alpha_6} (C_{\alpha_4} S_{\theta_1} - S_{\alpha_4} C_{\theta_1} S_{\phi_2}) + C_{\alpha_6} (C_{\phi_5} * (S_{\alpha_4} S_{\theta_1} + C_{\alpha_4} C_{\theta_1} S_{\phi_2}) + C_{\phi_2} C_{\theta_1} S_{\phi_5})$$

$$r_{23} = -C_{\alpha_6} * (C_{\phi_5} S_{\alpha_4} C_{\theta_1} - C_{\alpha_4} S_{\phi_2} S_{\theta_1}) - C_{\phi_2} S_{\phi_5} S_{\theta_1} - S_{\alpha_6} (C_{\alpha_4} C_{\theta_1} + S_{\alpha_4} S_{\phi_2} S_{\theta_1})$$

$$r_{33} = -C_{\alpha_6} * (S_{\phi_2} S_{\phi_5} - C_{\alpha_4} C_{\phi_2} C_{\phi_5}) - C_{\phi_2} S_{\alpha_4} S_{\alpha_6}$$

$$d_x = a_3 C_{\phi_2} C_{\theta_1} - b_2 S_{\theta_1} - a_6 (S_{\phi_5} (S_{\alpha_4} S_{\theta_1} + C_{\alpha_4} C_{\theta_1} S_{\phi_2}) - C_{\phi_2} C_{\phi_5} C_{\theta_1})$$

$$d_y = b_2 C_{\theta_1} + a_6 (S_{\phi_5} (S_{\alpha_4} C_{\theta_1} - C_{\alpha_4} S_{\phi_2} S_{\theta_1}) + C_{\phi_2} C_{\phi_5} S_{\theta_1}) + a_3 C_{\phi_2} S_{\theta_1}$$

$$d_z = d_1 - a_3 S_{\phi_2} - a_6 (C_{\phi_5} \sin(p_2) + C_{\alpha_4} C_{\phi_2} S_{\phi_5})$$

5 CAPÍTULO V

MODELACIÓN DEL ROBOT STANFORD CINEMÁTICA INVERSA

Como se estudió en el capítulo 3, en la cinemática directa se determina la posición y orientación del extremo final del robot, conocidas las variables angulares, ahora en cinemática inversa se van a determinar las variables articulares de tal manera que el extremo del brazo del robot alcance una posición y orientación deseada, en este caso se tiene como datos las coordenadas del extremo final del robot con respecto al origen de coordenadas x_6^0, y_6^0, z_6^0 , también se conoce la matriz de transformación homogénea del brazo del robot H

En cinemática inversa resulta más complejo que en cinemática directa, pues puede existir varias soluciones que resuelvan un mismo problema, existen varios métodos de solución entre ellos están: el método geométrico, método algebraico, método del desacoplo cinemático, método de la matriz inversa.

En este caso, para el robot Stanford que tiene seis grados de libertad, el método más adecuado es el de la matriz inversa, una variante del método de la matriz inversa es la del desacoplo cinemático conocido también como solución de Pieper, que se describe con detalle en el apartado 5.1.

5.1 DESACOPLO CINEMÁTICO. SOLUCIÓN DE PIEPER

El método de Pieper considera a los robots manipuladores que están conformados de un brazo y de una muñeca, el punto donde se cortan los tres ejes de rotación de la muñeca, es utilizado para el desarrollo de este método.

La posición del punto de corte de los tres ejes de la muñeca, en este punto coinciden los orígenes de los sistemas de coordenadas S_3, S_4, S_5 ; este punto puede ser localizado

a través del vector \vec{P}_5^0 , nótese que este mismo punto es el extremo del brazo del robot, mientras que el punto de la pinza del robot manipulador puede ser determinado con el vector \vec{P}_6^0 , la distancia que existe entre la muñeca y la pinza del robot viene determinado por el vector \vec{d}_6 , como puede verse en la figura 5.1

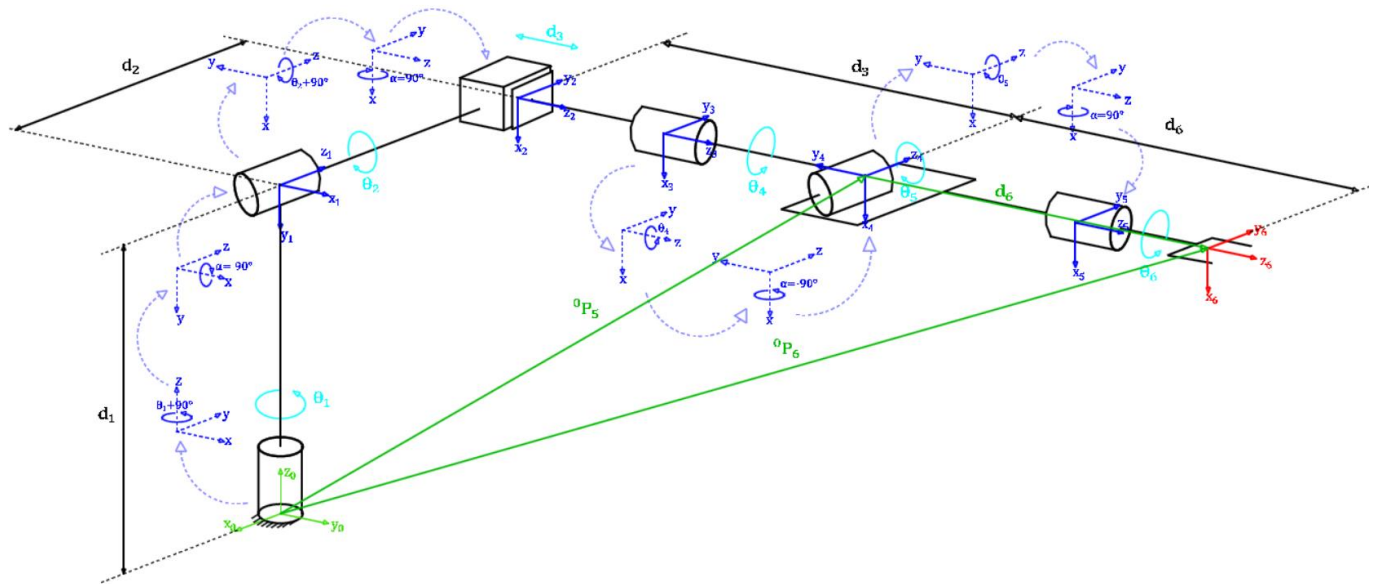


Figura 5.1 Robot Stanford, desacoplo cinemático Fuente: Propia

Utilizando la figura 5.1 se puede plantear la siguiente ecuación vectorial:

$$\vec{P}_6^0 = \vec{P}_5^0 + \vec{d}_6 \quad (5.1)$$

Despejando el vector posición de la muñeca se tiene:

$$\vec{P}_5^0 = \vec{P}_6^0 - \vec{d}_6 \quad (5.2)$$

Donde, se conoce el vector \vec{P}_6^0 se conoce, pues no es más que las coordenadas del extremo final del robot que se desean alcanza, nos faltaría encontrar el vector \vec{d}_6 .

En cinemática directa se encontró la matriz de transformación homogénea del brazo del robot Stanford (Ecuación 4.9)

$$H = T_6^0 = \begin{bmatrix} r_{11} & r_{12} & r_{13} & d_x \\ r_{21} & r_{22} & r_{23} & d_y \\ r_{31} & r_{32} & r_{33} & d_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Obsérvese que la matriz de la transformación homogénea está conformada por cuatro submatrices, la submatriz de orientación, la submatriz de posición, la submatriz de perspectiva y la submatriz del factor de escala.

La submatriz de orientación está conformada a su vez por los vectores unitarios del último sistema de coordenadas, es común en cinemática inversa cambiar la nomenclatura a la submatriz de rotación, tal como se indica a continuación:

$$R_6^0 = \begin{bmatrix} n_x & o_x & a_x \\ n_y & o_y & a_y \\ n_z & o_z & a_z \end{bmatrix} \quad (5.3)$$

Donde los vectores unitarios del último sistema de coordenadas son:

$\vec{n} = n_x\vec{i} + n_y\vec{j} + n_z\vec{k}$ vector unitario del eje x_6 $n = normal$ (*normal*)

$\vec{o} = o_x\vec{i} + o_y\vec{j} + o_z\vec{k}$ vector unitario del eje y_6 $o = open$ (*abrir*)

$\vec{a} = a_x\vec{i} + a_y\vec{j} + a_z\vec{k}$ vector unitario del eje z_6 $a = approach$ (*acercar*)

El vector \vec{d}_6 se encuentra multiplicando la magnitud o longitud d_6 por el vector unitario \vec{a} en la dirección z_6

$$\vec{d}_6 = d_6(a_x\vec{i} + a_y\vec{j} + a_z\vec{k}) \quad (5.4)$$

Reemplazando el vector posición \vec{P}_6^0 que es conocido por ser dato, y la ecuación 5.4 en la ecuación 5.2, se tiene la siguiente ecuación vectorial:

$$x_5^0\vec{i} + y_5^0\vec{j} + z_5^0\vec{k} = x_6^0\vec{i} + y_6^0\vec{j} + z_6^0\vec{k} - d_6(a_x\vec{i} + a_y\vec{j} + a_z\vec{k}) \quad (5.5)$$

Utilizando la ecuación 5.5 se puede calcular la posición de la muñeca, escrito en forma escalar se tiene:

$$\begin{aligned}x_5^0 &= x_6^0 - d_6 a_x \\y_5^0 &= y_6^0 - d_6 a_y \\z_5^0 &= z_6^0 - d_6 a_z\end{aligned}\tag{5.6}$$

Las coordenadas de la muñeca así encontradas se utilizan para resolver el problema cinemático inverso de las tres primeras articulaciones del robot q_1, q_2, q_3 utilizando el método de las matrices inversas.

Finalmente queda por determinar los valores de las tres últimas articulaciones que orientarán al extremo del robot para que el problema cinemático inverso quede totalmente resuelto.

$$\begin{aligned}R_6^0 &= R_3^0 R_6^3 \\R_6^3 &= (R_3^0)^{-1} R_6^0\end{aligned}\tag{5.7}$$

En robótica se conoce que la matriz de rotación inversa, es igual a la matriz transpuesta tal como se estudió en el capítulo 2, entonces tenemos que:

$$R_6^3 = (R_3^0)^T R_6^0\tag{5.8}$$

5.2 CINEMÁTICA INVERSA DEL ROBOT STANFORD

Utilizando la matriz de transformación homogénea entre el sistema de coordenadas de la muñeca del robot Stanford y el sistema de coordenadas de la base

$$A_3^0 = A_1 * A_2 * A_3$$

Utilizando la tabla 5.1 de parámetros de Denavit-Hartenberg estándar:

Tabla 5.1 Parámetros D-H estándar para manipulador Stanford.

Eslabón	θ_i	d_i	a_i	α_i
1	$\theta_1^* + 90^\circ$	d_1	0	-90°
2	$\theta_2^* + 90^\circ$	d_2	0	90°
3	0°	d_3^*	0	0°
4	θ_4^*	0	0	-90°
5	θ_5^*	0	0	90°
6	θ_6^*	d_6	0	0°

*variable de la articulación

Las matrices de transformación son:

$$A_1 = \begin{bmatrix} c_{1+90} & 0 & -s_{1+90} & 0 \\ s_{1+90} & 0 & c_{1+90} & 0 \\ 0 & -1 & 0 & d_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} -s_1 & 0 & -c_1 & 0 \\ c_1 & 0 & -s_1 & 0 \\ 0 & -1 & 0 & d_1 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$A_2 = \begin{bmatrix} c_{2+90} & 0 & s_{2+90} & 0 \\ s_{2+90} & 0 & -c_{2+90} & 0 \\ 0 & -1 & 0 & d_2 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} -s_2 & 0 & -c_2 & 0 \\ c_2 & 0 & -s_2 & 0 \\ 0 & 1 & 0 & d_2 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$A_3 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_3 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$A_4 = \begin{bmatrix} c_4 & 0 & -s_4 & 0 \\ s_4 & 0 & c_4 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$A_5 = \begin{bmatrix} c_5 & 0 & s_5 & 0 \\ s_5 & 0 & -c_5 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$A_6 = \begin{bmatrix} c_6 & -s_6 & 0 & 0 \\ s_6 & c_6 & 0 & 0 \\ 0 & 0 & 1 & d_6 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

La matriz de transformación homogénea entre el sistema S_6 con respecto a la base S_0 se encuentra multiplicando a las anteriores matrices y el resultado viene dado por la ecuación 4.9

$$H = T_6^0 = \begin{matrix} & \mathbf{n} & \mathbf{o} & \mathbf{a} & \\ & r_{11} & r_{12} & r_{13} & d_x \\ & r_{21} & r_{22} & r_{23} & d_y \\ & r_{31} & r_{32} & r_{33} & d_z \\ & 0 & 0 & 0 & 1 \end{matrix}$$

Donde H es la matriz de transformación homogénea del manipulador Stanford, el vector unitario en el eje z_6 , se obtiene de la submatriz de orientación más conocida como la matriz *noa*, cabe recalcar que esta submatriz es entre el sistema S_6 con respecto al sistema base S_0 y viene dado por:

$$\vec{a} = (r_{13})\vec{i} + (r_{23})\vec{j} + (r_{33})\vec{k}$$

Multiplicando la longitud del eslabón d_6 por el vector unitario \vec{a} , se tiene:

$$\vec{d}_6 = d_6 \left((r_{13})\vec{i} + (r_{23})\vec{j} + (r_{33})\vec{k} \right) \quad (5.9)$$

El vector posición del extremo final del robot Stanford viene dado por las coordenadas del punto del extremo final del robot a que queremos llegar, en consecuencia son datos

$$\vec{P}_6^0 = x_6^0\vec{i} + y_6^0\vec{j} + z_6^0\vec{k} \quad (5.10)$$

Reemplazando la ecuación 5.10 y 5.9 en la ecuación 5.6 se tiene, las coordenadas de la posición de la muñeca del Robot Stanford

$$x_5^0 = x_6^0 - d_6(r_{13})$$

$$y_5^0 = y_6^0 - d_6(r_{23})$$

$$z_5^0 = z_6^0 - d_6(r_{33})$$

Para encontrar las variables articulares q_1, q_2, q_3 que es común llamar en robótica en lugar de $\theta_1, \theta_2, \theta_3$, nosotros vamos a partir de la ecuación matricial del brazo.

$$A_3^0 = A_1 * A_2 * A_3$$

$$A_3^0 = \begin{bmatrix} -s_1 & 0 & -c_1 & 0 \\ c_1 & 0 & -s_1 & 0 \\ 0 & -1 & 0 & d_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} -s_2 & 0 & -c_2 & 0 \\ c_2 & 0 & -s_2 & 0 \\ 0 & 1 & 0 & d_2 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_3 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$A_3^0 = \begin{bmatrix} s_1 s_2 & -c_1 & -c_2 s_1 & -d_2 c_1 - d_3 c_2 s_1 \\ -c_1 s_2 & -s_1 & c_1 c_2 & -d_2 s_1 + d_3 c_1 c_2 \\ -c_2 & 0 & -s_2 & d_1 - d_3 s_2 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

La matriz A_3^0 que en este caso es la matriz del brazo Stanford puede ser escrita en forma genérica por las submatrices de orientación *noa*, esta submatriz es la matriz de rotación entre el sistema S_3 con respecto al sistema S_0 , la submatriz de posición de la muñeca que en nuestro caso viene dado por x_3^0, y_3^0, z_3^0 ; nótese que estas coordenadas son las mismas coordenadas que x_5^0, y_5^0, z_5^0 debido a que el origen de estos dos sistemas de coordenadas están traslapados. Utilizando el método de la matriz inversa se puede escribir la siguiente ecuación matricial

$$A_1^{-1} A_3^0 = A_2 * A_3$$

$$\begin{bmatrix} -s_1 & c_1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ -c_1 & -s_1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} n_x & o_x & a_x & x_3^0 \\ n_y & o_y & a_y & y_3^0 \\ n_z & o_z & a & z_3^0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} -s_2 & 0 & -c_2 & 0 \\ c_2 & 0 & -s_2 & 0 \\ 0 & 1 & 0 & d_2 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_3 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\begin{bmatrix} -s_1 & c_1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ -c_1 & -s_1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} n_x & o_x & a_x & x_5^0 \\ n_y & o_y & a_y & y_5^0 \\ n_z & o_z & a & z_5^0 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} -s_2 & 0 & c_2 & d_3 c_2 \\ c_2 & 0 & s_2 & d_3 s_2 \\ 0 & 1 & 0 & d_2 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Analizando la anterior ecuación matricial se puede igualar los 16 términos que contiene una matriz, en otras palabras se pueden plantear 16 ecuaciones, debemos escoger aquella ecuación que sea más amigable para despejar la variable articular.

Igualando los valores de los términos r_{34} del lado izquierdo y el lado derecho de la ecuación matricial se encuentra la variable θ_1

$$\begin{aligned}
 -x_5^0 \cos(\theta_1) - y_5^0 \sin(\theta_1) &= d_2 \\
 -x_5^0 \sqrt{1 - \sin^2(\theta_1)} - y_5^0 \sin(\theta_1) &= d_2 \\
 x_5^{0^2} (1 - \sin^2(\theta_1)) &= (d_2 + y_5^0 \sin(\theta_1))^2 \\
 x_5^{0^2} - x_5^{0^2} \sin^2(\theta_1) &= d_2^2 + 2d_2 y_5^0 \sin(\theta_1) + y_5^{0^2} \sin^2(\theta_1) \\
 x_5^{0^2} + d_2^2 + 2d_2 y_5^0 \sin(\theta_1) + (x_5^{0^2} + y_5^{0^2}) \sin^2(\theta_1) &= 0 \\
 \sin \theta_1 &= \frac{-2d_2 y_5^0 \pm \sqrt{(2d_2 y_5^0)^2 - 4(x_5^{0^2} + y_5^{0^2})(x_5^{0^2} + d_2^2)}}{2(x_5^{0^2} + y_5^{0^2})} \\
 \theta_1 &= \arcsen \left[\frac{-2d_2 y_5^0 \pm \sqrt{(2d_2 y_5^0)^2 - 4(x_5^{0^2} + y_5^{0^2})(x_5^{0^2} + d_2^2)}}{2(x_5^{0^2} + y_5^{0^2})} \right] \quad (5.11)
 \end{aligned}$$

Igualando los términos r_{24} se encuentra el término θ_2 , aclarando que d_3 también es incógnita y en consecuencia se tiene que generar otra ecuación para tener dos ecuaciones con dos incógnitas.

$$\begin{aligned}
 -z_5^0 &= d_3 \sin(\theta_2) \quad (5.12) \\
 \theta_2 &= \arcsen \left(\frac{-z_5^0}{d_3} \right)
 \end{aligned}$$

Considerando el término r_{14} se puede determinar la variable articular θ_2 ,

$$-x_0^5 \sin(\theta_1) * y_0^5 \cos(\theta_1) = d_3 \cos(\theta_2) \quad (5.13)$$

Para resolver más fácilmente el sistema de dos ecuaciones 5.12 y 5.13, vamos a elevar al cuadrado a las dos ecuaciones y les sumamos tal como se indica a continuación:

$$\begin{aligned}
 (-x_0^5 \sin(\theta_1) * y_0^5 \cos(\theta_1))^2 + (-z_5^0)^2 &= (d_3 \cos(\theta_2))^2 + (d_3 \sin(\theta_2))^2 \quad (5.14) \\
 d_3 &= \sqrt{(-x_0^5 \sin(\theta_1) * y_0^5 \cos(\theta_1))^2 + (-z_5^0)^2}
 \end{aligned}$$

Reemplazando d_3 en la ecuación 5.12 se obtiene la variable articular θ_2 ,

$$\theta_2 = \arcsen \left(\frac{-z_5^0}{\sqrt{(-x_0^5 \text{sen}(\theta_1) * y_0^5 \text{cos}(\theta_1))^2 + (-z_5^0)^2}} \right) \quad (5.15)$$

A continuación vamos a encontrar las variables articulares q_4, q_5, q_6 para ello utilizaremos la siguiente ecuación matricial, en la que no hace falta utilizar las matrices de transformación homogénea debido a que no se tiene traslaciones, tan solo se tiene rotaciones, por esta razón se puede utilizar sólo a las submatrices de rotación, como se indica a continuación:

$$\begin{aligned} R_6^0 &= R_3^0 R_6^3 \\ R_6^3 &= (R_3^0)^{-1} R_6^0 \\ R_6^3 &= (R_3^0)^T R_6^0 \end{aligned} \quad (5.16)$$

La matriz de rotación de 0 a 6 se puede escribir de manera genérica a través de la matriz *noa* que no es más que la matriz de rotación total que se ha realizado con el último sistema de coordenadas S_6

Utilizando la tabla de parámetros Denavit-Hartenberg estándar se puede escribir la matriz de rotación de 3 a 6

$$R_6^3 = R_4^3 * R_5^4 * R_6^5$$

$$R_4^3 = \begin{bmatrix} c_4 & 0 & -s_4 \\ s_4 & 0 & c_4 \\ 0 & -1 & 0 \end{bmatrix}$$

$$R_5^4 = \begin{bmatrix} c_5 & 0 & s_5 \\ s_5 & 0 & -c_5 \\ 0 & 1 & 0 \end{bmatrix}$$

$$R_6^5 = \begin{bmatrix} c_6 & -s_6 & 0 \\ s_6 & c_6 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Multiplicando estas matrices de rotación se tiene la matriz resultante R_6^3 ,

$$R_6^3 = \begin{bmatrix} c_4 c_5 c_6 - s_4 s_6 & -c_6 s_4 - c_4 c_5 s_6 & c_4 s_5 \\ c_4 s_6 + c_5 c_6 s_4 & c_4 c_6 - c_5 s_4 s_6 & s_4 s_5 \\ -c_6 s_5 & s_5 s_6 & c_5 \end{bmatrix} \quad (5.17)$$

La matriz de rotación de 0 a 3 se encuentra las tres primeras filas de la tabla 4.1 de parámetros de Denavit-Hartenberg

$$R_3^0 = R_1^0 * R_2^1 * R_3^2$$

$$R_1^0 = \begin{bmatrix} -s_1 & 0 & -c_1 \\ c_1 & 0 & -s_1 \\ 0 & -1 & 0 \end{bmatrix}$$

$$R_2^1 = \begin{bmatrix} -s_2 & 0 & -c_2 \\ c_2 & 0 & -s_2 \\ 0 & 1 & 0 \end{bmatrix}$$

$$R_3^2 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Multiplicando estas matrices de rotación se tiene la matriz resultante R_3^0

$$R_3^0 = \begin{bmatrix} s_1 s_2 & -c_1 & -c_2 s_1 \\ c_1 s_2 & -s_1 & c_1 c_2 \\ -c_2 & 0 & -s_1 \end{bmatrix}$$

$$(R_3^0)^{-1} = (R_3^0)^T = \begin{bmatrix} s_1 s_2 & c_1 s_2 & -c_2 \\ -c_1 & -s_1 & 0 \\ -c_2 s_1 & c_1 c_2 & -s_1 \end{bmatrix} \quad (5.18)$$

Reemplazando las ecuaciones 5.17, 5.18 y la matriz **noa** en la ecuación 5.16 se tiene:

$$R_6^3 = (R_3^0)^T R_6^0$$

$$\begin{bmatrix} c_4 c_5 c_6 - s_4 s_6 & -c_6 s_4 - c_4 c_5 s_6 & c_4 s_5 \\ c_4 s_6 + c_5 c_6 s_4 & c_4 c_6 - c_5 s_4 s_6 & s_4 s_5 \\ -c_6 s_5 & s_5 s_6 & c_5 \end{bmatrix} = \begin{bmatrix} s_1 s_2 & c_1 s_2 & -c_2 \\ -c_1 & -s_1 & 0 \\ -c_2 s_1 & c_1 c_2 & -s_1 \end{bmatrix} \begin{bmatrix} n_x & o_x & a_x \\ n_y & o_y & a_y \\ n_z & o_z & a_z \end{bmatrix}$$

De esta ecuación matricial tenemos que escoger aquellos términos que nos generen una ecuación amigable para despejar las variables articulares. Utilizando el término r_{33} podemos encontrar θ_5

$$\cos \theta_5 = -a_x \cos \theta_2 \sin \theta_1 + a_y \cos \theta_1 \cos \theta_2 - a_z \sin \theta_1$$

$$\theta_5 = \arccos(-a_x \cos \theta_2 \sin \theta_1 + a_y \cos \theta_1 \cos \theta_2 - a_z \sin \theta_1)$$

Igualando los términos r_{32} , hallamos θ_6

$$\sin \theta_5 \sin \theta_6 = -o_x \cos \theta_2 \sin \theta_1 + o_y \cos \theta_1 \cos \theta_2 - o_z \sin \theta_1$$

$$\theta_6 = \arccos\left(\frac{-o_x \cos \theta_2 \sin \theta_1 + o_y \cos \theta_1 \cos \theta_2 - o_z \sin \theta_1}{\sin \theta_5}\right)$$

Y finalmente, igualando los términos r_{23} , despejamos la variable articular θ_4

$$\sin \theta_4 \sin \theta_5 = -a_x \cos \theta_1 \sin \theta_1 + a_y \sin \theta_1$$

$$\theta_4 = \arcsen\left(\frac{-a_x \cos \theta_1 \sin \theta_1 + a_y \sin \theta_1}{\sin \theta_5}\right)$$

6 CAPÍTULO VI

ANÁLISIS DEL ROBOT STANFORD

6.1 ANÁLISIS DE LOS RESULTADOS DE LA CINEMÁTICA DIRECTA

En el análisis de la cinemática directa vamos a considerar los resultados obtenidos al utilizar los distintos métodos en la modelación, tanto el método de Denavit-Hartenberg estándar, el modificado y el método del movimiento general continuo.

Con los tres métodos se lograron llegar a los mismos resultados, las diferencias entre estos radica en la asignación de los sistemas de coordenadas y la secuencia de movimientos que se tienen que realizar en el mundo físico, dichos movimientos son reemplazados en el mundo virtual por matrices de rotación y traslación, es de mucha importancia cuidar que la secuencia del producto matricial sea en el mismo orden que de los movimientos que se realizaron al sistema de coordenadas móvil.

El procedimiento en estos tres métodos es semejante cuando se realiza el esquema , más conocido como el esqueleto del mecanismo, utilizando para ello cilindros para las articulaciones de revolución y cubos para las articulaciones prismáticas, estas articulaciones son conectadas a través de líneas rectas que son los ejes y que también representan a los eslabones de un mecanismo, luego se tiene que dibujar los ejes z a lo largo de la línea de ejes del dibujo técnico mecánico sobre cada articulación, la articulación fija a la cimentación sirve para asignar al sistema de coordenadas base, más conocido como sistema s_0 . El eje x_0 se escoge de manera que resulte amigable para los movimientos posteriores del sistema de coordenadas móvil, utilizando el sistema de coordenadas dextrógiro se asigna el otro eje de coordenadas y_0 ; el sistema de coordenadas móvil está inicialmente traslapado al sistema de coordenadas fijo y tenemos que moverle siguiendo la cadena cinemática a la siguiente articulación, según el método de Denavit-Hartenberg estándar es suficiente a lo mucho cuatro movimientos para pasar de una articulación a la siguiente, primero un movimiento de rotación y luego

uno de traslación en el eje z y otros dos movimientos de traslación y rotación en el eje x. Se resume este grupo de cuatro movimientos como rosca en z y luego rosca en x.

En el método Denavit-Hartenberg modificado también se utiliza grupos de cuatro movimientos, la diferencia está en que primero se realizan los movimientos de rotación y traslación sobre el eje x y luego sobre el eje z, esto es rosca en x y luego rosca en z.

En el método del movimiento general la diferencia con respecto a los dos métodos anteriores, está en que los movimientos que se tienen que realizar al sistema de coordenadas móviles sólo tiene que cumplir con la secuencia de los movimientos y en ese mismo orden se debe plantear el producto matricial.

A continuación se presenta la matriz de transformación homogénea del manipulador Stanford obtenida por los tres métodos descritos:

6.1.1 MATRIZ H DEL ROBOT STANFORD UTILIZANDO EL MÉTODO DE DENAVIT-HARTENBERG ESTÁNDAR

$$H = T_6^0 = \begin{bmatrix} r_{11} & r_{12} & r_{13} & d_x \\ r_{21} & r_{22} & r_{23} & d_y \\ r_{31} & r_{32} & r_{33} & d_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Donde,

$$r_{11} = -C_6(C_5(C_1S_4 - C_4S_1S_2) - C_2S_1S_5) - S_6(C_1C_4 + S_1S_2S_4)$$

$$r_{21} = -C_6(C_5(S_1S_4 + C_1C_4S_2) + C_1C_2S_5) - S_6(C_4S_1 - C_1S_2S_4)$$

$$r_{31} = C_6(S_2S_5 - C_2C_4C_5) + C_2S_4S_6$$

$$r_{12} = S_6(C_5(C_1S_4 - C_4S_1S_2) - C_2S_1S_5) - C_6(C_1C_4 + S_1S_2S_4)$$

$$r_{22} = S_6(C_5(S_1S_4 + C_1C_4S_2) + C_1C_2S_5) - C_6(C_4S_1 - C_1S_2S_4)$$

$$r_{32} = C_2C_6S_4 - S_6(S_2S_5 - C_2C_4C_5)$$

$$r_{13} = -S_5(C_1S_4 - C_4S_1S_2) - C_2C_5S_1$$

$$r_{23} = C_1C_2C_5 - S_5(S_1S_4 + C_1C_4S_2)$$

$$r_{33} = -C_5S_2 - C_2C_4S_5$$

$$d_x = -D_2C_1 - D_6(S_5(C_1S_4 - C_4S_1S_2) + C_2C_5S_1) - D_3C_2S_1$$

$$d_y = D_3C_1C_2 - D_6(S_5(S_1S_4 + C_1C_4S_2) - C_1C_2C_5) - D_2S_1$$

$$d_z = D_1 - D_6(C_5S_2 + C_2C_4S_5) - D_3S_2$$

Se puede validar rápidamente estos resultados si reemplazamos para las variables articulares cero y se observa que la función $\text{sen } 0 = 0$, mientras que $\text{cos } 0 = 1$, esto nos da que:

$$\begin{aligned}d_x &= -d_2 \\d_y &= d_3 + d_6 \\d_z &= d_1\end{aligned}$$

Lo cual es correcto al contrastar estos resultados con la figura 4.2, que corresponde al estado inicial del robot manipulador Stanford.

Estos resultados fueron validados a su vez utilizando Matlab, como se muestra en la figura 6.1

```

1  % Robot stanford D-Hs
2
3  - syms t1 t2 D3 t4 t5 t6           % variables articulares
4  - syms a1 D1 D2 a2 a4 a5 D6       % constantes
5  - a1=-pi/2
6  - a2=pi/2
7  - a4=-pi/2
8  - a5=pi/2
9
10 - t1=0;t2=0;t4=0;t5=0;t6=0;      % variables articulares= cero
11
12 - A1=rotz(t1+pi/2)*transl(0,0,D1)*rotx(a1);
13
14 - A2=rotz(t2+pi/2)*transl(0,0,D2)*rotx(a2);
15
16 - A3=transl(0,0,D3)
17
18 - A5=rotz(t5)*rotx(a5);
19
20 - A6=rotz(t6)*transl(0,0,D6)
21
22 - T06=simple(A1*A2*A3*A4*A5*A6)

```

```

Command Window

T06 =

[ 0, -1, 0, -D2]
[ 0, 0, 1, D3 + D6]
[ -1, 0, 0, D1]
[ 0, 0, 0, 1]

fx >>

```

Figura 6.1 Comprobación del modelo obtenido con el método de D-H estándar

6.1.2 MATRIZ H DEL ROBOT STANFORD UTILIZANDO EL MÉTODO DE DENAVIT-HARTENBERG MODIFICADO

$$H = T_6^0 = \begin{bmatrix} r_{11} & r_{12} & r_{13} & d_x \\ r_{21} & r_{22} & r_{23} & d_y \\ r_{31} & r_{32} & r_{33} & d_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Donde,

$$\begin{aligned} r_{11} &= -C_6(C_5(C_1S_4 - C_4S_1S_2) - C_2S_1S_5) - S_6(C_1C_4 + S_1S_2S_4) \\ r_{21} &= -C_6(C_5(S_1S_4 + C_1C_4S_2) + C_1C_2S_5) - S_6(C_4S_1 - C_1S_2S_4) \\ r_{31} &= C_6(S_2S_5 - C_2C_4C_5) + C_2S_4S_6 \\ r_{12} &= S_6(C_5(C_1S_4 - C_4S_1S_2) - C_2S_1S_5) - C_6(C_1C_4 + S_1S_2S_4) \\ r_{22} &= S_6(C_5(S_1S_4 + C_1C_4S_2) + C_1C_2S_5) - C_6(C_4S_1 - C_1S_2S_4) \\ r_{32} &= C_2C_6S_4 - S_6(S_2S_5 - C_2C_4C_5) \\ r_{13} &= -S_5(C_1S_4 - C_4S_1S_2) - C_2C_5S_1 \\ r_{23} &= C_1C_2C_5 - S_5(S_1S_4 + C_1C_4S_2) \\ r_{33} &= -C_5S_2 - C_2C_4S_5 \\ d_x &= -D_2C_1 - D_6(S_5(C_1S_4 - C_4S_1S_2) + C_2C_5S_1) - D_3C_2S_1 \\ d_y &= D_3C_1C_2 - D_6(S_5(S_1S_4 + C_1C_4S_2) - C_1C_2C_5) - D_2S_1 \\ d_z &= D_1 - D_6(C_5S_2 + C_2C_4S_5) - D_3S_2 \end{aligned}$$

Como se observa, los resultados son exactamente iguales, aún cuando aparentemente las matrices de transformación de coordenadas A_i entre un sistema y el sistema contiguo son diferentes, además la tabla de parámetros, analizando más al detalle estas dos tablas de parámetros revelan que se respeta la secuencia de movimientos o en otras palabras satisfacen el principio de la post-multiplicación matricial.

De todas maneras se procede a validar rápidamente estos resultados si reemplazamos para las variables articulares cero y nuevamente se verifica que para la función $\sin 0 = 0$, y que $\cos 0 = 1$, nos entrega los siguientes resultados:

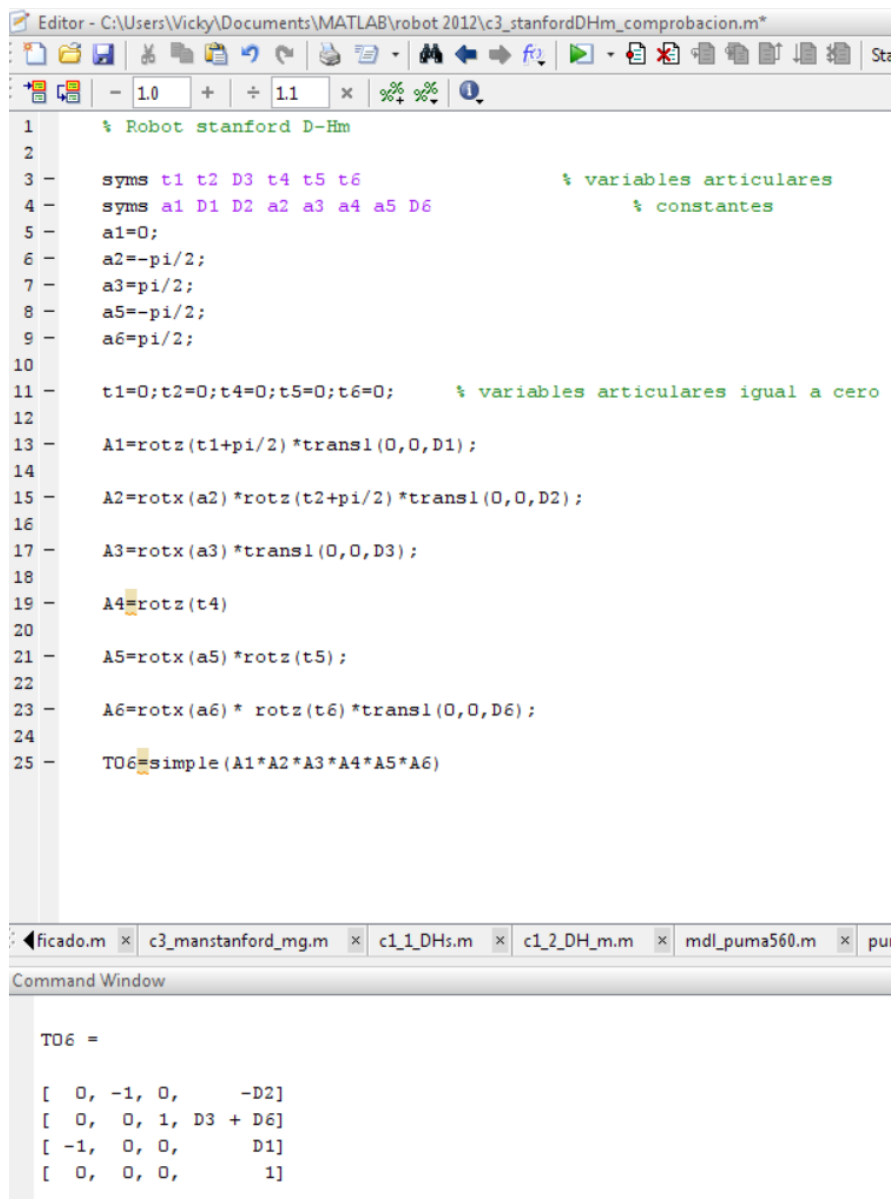
$$d_x = -d_2$$

$$d_y = d_3 + d_6$$

$$d_z = d_1$$

Lo cual es correcto al contrastar estos resultados con la figura 4.2, que corresponde al estado inicial del robot manipulador Stanford.

Estos resultados fueron validados a su vez utilizando Matlab, como se muestra en la figura 6.2



```

1 % Robot stanford D-Hm
2
3 syms t1 t2 D3 t4 t5 t6 % variables articulares
4 syms a1 D1 D2 a2 a3 a4 a5 D6 % constantes
5 a1=0;
6 a2=-pi/2;
7 a3=pi/2;
8 a5=-pi/2;
9 a6=pi/2;
10
11 t1=0;t2=0;t4=0;t5=0;t6=0; % variables articulares igual a cero
12
13 A1=rotz(t1+pi/2)*transl(0,0,D1);
14
15 A2=rotx(a2)*rotz(t2+pi/2)*transl(0,0,D2);
16
17 A3=rotx(a3)*transl(0,0,D3);
18
19 A4=rotz(t4)
20
21 A5=rotx(a5)*rotz(t5);
22
23 A6=rotx(a6)*rotz(t6)*transl(0,0,D6);
24
25 T06=simple(A1*A2*A3*A4*A5*A6)

```

Command Window

```

T06 =
[ 0, -1, 0, -D2]
[ 0, 0, 1, D3 + D6]
[ -1, 0, 0, D1]
[ 0, 0, 0, 1]

```

Figura 6.2 Comprobación del modelo obtenido con el método de D-H modificado

6.1.3 MATRIZ H DEL ROBOT STANFORD UTILIZANDO EL MÉTODO DEL MOVIMIENTO GENERAL, SIN CONSIDERAR MOVIMIENTOS EN EL EJE Y

$$H = \begin{bmatrix} r_{11} & r_{12} & r_{13} & d_x \\ r_{21} & r_{22} & r_{23} & d_y \\ r_{31} & r_{32} & r_{33} & d_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Donde,

$$\begin{aligned} r_{11} &= -C_6(C_5(C_1S_4 - C_4S_1S_2) - C_2S_1S_5) - S_6(C_1C_4 + S_1S_2S_4) \\ r_{21} &= -C_6(C_5(S_1S_4 + C_1C_4S_2) + C_1C_2S_5) - S_6(C_4S_1 - C_1S_2S_4) \\ r_{31} &= C_6(S_2S_5 - C_2C_4C_5) + C_2S_4S_6 \\ r_{12} &= S_6(C_5(C_1S_4 - C_4S_1S_2) - C_2S_1S_5) - C_6(C_1C_4 + S_1S_2S_4) \\ r_{22} &= S_6(C_5(S_1S_4 + C_1C_4S_2) + C_1C_2S_5) - C_6(C_4S_1 - C_1S_2S_4) \\ r_{32} &= C_2C_6S_4 - S_6(S_2S_5 - C_2C_4C_5) \\ r_{13} &= -S_5(C_1S_4 - C_4S_1S_2) - C_2C_5S_1 \\ r_{23} &= C_1C_2C_5 - S_5(S_1S_4 + C_1C_4S_2) \\ r_{33} &= -C_5S_2 - C_2C_4S_5 \\ d_x &= -D_2C_1 - D_6(S_5(C_1S_4 - C_4S_1S_2) + C_2C_5S_1) - D_3C_2S_1 \\ d_y &= D_3C_1C_2 - D_6(S_5(S_1S_4 + C_1C_4S_2) - C_1C_2C_5) - D_2S_1 \\ d_z &= D_1 - D_6(C_5S_2 + C_2C_4S_5) - D_3S_2 \end{aligned}$$

De estos resultados se desprende que las ecuaciones obtenidas por los tres métodos son idénticas, pese a tener mayor libertad en la asignación de sistemas de coordenadas.

En esta primera aplicación la secuencia de movimientos que se realizó al sistema de coordenadas móvil y siguiendo la cadena cinemática del mecanismo fueron realizados utilizando en parte las recomendaciones del convenio de Denavit-Hartenberg, es decir sin considerar movimientos en el eje y.

Se puede observar que es más simple en la asignación de sistemas de coordenadas, y que se tienen menor cantidad de matrices en el producto matricial.

Comprobamos que al reemplazar los valores de las variables articulares igual a cero se obtienen exactamente los mismos resultados que con los otros métodos ya analizados, esto es:

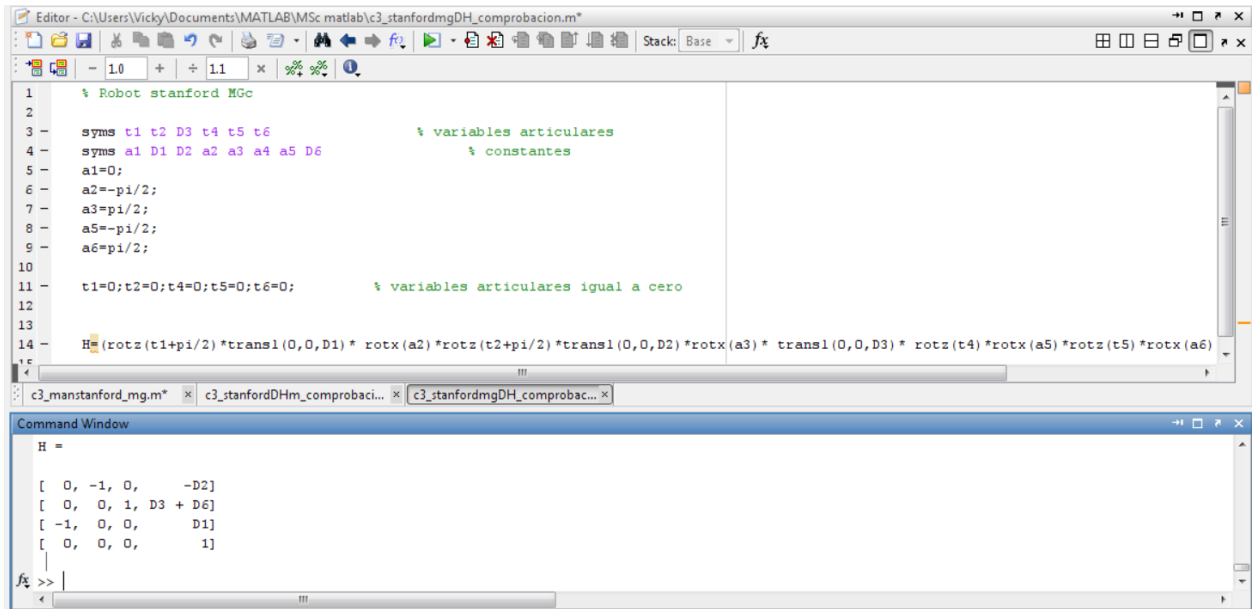
$$d_x = -d_2$$

$$d_y = d_3 + d_6$$

$$d_z = d_1$$

Lo cual es correcto al contrastar estos resultados con la figura 4.8, que corresponde al estado inicial del robot manipulador Stanford.

Estos resultados fueron validados a su vez utilizando Matlab, como se muestra en la figura 6.3



```

1 % Robot stanford MGc
2
3 - syms t1 t2 D3 t4 t5 t6           % variables articulares
4 - syms a1 D1 D2 a2 a3 a4 a5 D6    % constantes
5 - a1=0;
6 - a2=-pi/2;
7 - a3=pi/2;
8 - a5=-pi/2;
9 - a6=pi/2;
10
11 - t1=0;t2=0;t4=0;t5=0;t6=0;      % variables articulares igual a cero
12
13
14 - H=(rotz(t1+pi/2)*transl(0,0,D1)*rotx(a2)*rotz(t2+pi/2)*transl(0,0,D2)*rotx(a3)*transl(0,0,D3)*rotz(t4)*rotx(a5)*rotz(t5)*rotx(a6))
15
16

```

Command Window

```

H =
[ 0, -1, 0, -D2]
[ 0, 0, 1, D3 + D6]
[ -1, 0, 0, D1]
[ 0, 0, 0, 1]

```

Figura 6.3 Comprobación del modelo obtenido con el método de MG sin movimientos en el eje “y”

6.1.4 MATRIZ H DEL ROBOT STANFORD UTILIZANDO EL MÉTODO DEL MOVIMIENTO GENERAL, INCLUYENDO MOVIMIENTOS EN EL EJE Y

$$H = \begin{bmatrix} r_{11} & r_{12} & r_{13} & d_x \\ r_{21} & r_{22} & r_{23} & d_y \\ r_{31} & r_{32} & r_{33} & d_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Donde,

$$r_{11} = C_{\phi_2} C_{\phi_5} C_{\theta_1} - S_{\phi_5} (S_{\alpha_4} S_{\theta_1} + C_{\alpha_4} C_{\theta_1} S_{\phi_2})$$

$$r_{21} = S_{\phi_5} S_{\alpha_4} C_{\theta_1} - C_{\alpha_4} S_{\phi_2} S_{\theta_1} + C_{\phi_2} C_{\phi_5} S_{\theta_1}$$

$$r_{31} = -C_{\phi_5} S_{\phi_2} - C_{\alpha_4} C_{\phi_2} S_{\phi_5}$$

$$r_{12} = S_{\alpha_6} (C_{\phi_5} (S_{\alpha_4} S_{\theta_1} + C_{\alpha_4} C_{\theta_1} S_{\phi_2}) + C_{\phi_2} C_{\theta_1} S_{\phi_5}) - C_{\alpha_6} (C_{\alpha_4} S_{\theta_1} - S_{\alpha_4} C_{\theta_1} S_{\phi_2})$$

$$r_{22} = -C_{\alpha_6} (C_{\phi_5} (S_{\alpha_4} C_{\theta_1} - C_{\alpha_4} S_{\phi_2} S_{\theta_1}) - C_{\phi_2} S_{\phi_5} S_{\theta_1}) - S_{\alpha_6} (C_{\alpha_4} C_{\theta_1} + S_{\alpha_4} S_{\phi_2} S_{\theta_1})$$

$$r_{32} = C_{\alpha_6} C_{\phi_2} S_{\alpha_4} - S_{\alpha_6} (S_{\phi_2} S_{\phi_5} - C_{\alpha_4} C_{\phi_2} C_{\phi_5})$$

$$r_{13} = S_{\alpha_6} (C_{\alpha_4} S_{\theta_1} - S_{\alpha_4} C_{\theta_1} S_{\phi_2}) + C_{\alpha_6} (C_{\phi_5} * (S_{\alpha_4} S_{\theta_1} + C_{\alpha_4} C_{\theta_1} S_{\phi_2}) + C_{\phi_2} C_{\theta_1} S_{\phi_5})$$

$$r_{23} = -C_{\alpha_6} * (C_{\phi_5} S_{\alpha_4} C_{\theta_1} - C_{\alpha_4} S_{\phi_2} S_{\theta_1}) - C_{\phi_2} S_{\phi_5} S_{\theta_1} - S_{\alpha_6} (C_{\alpha_4} C_{\theta_1} + S_{\alpha_4} S_{\phi_2} S_{\theta_1})$$

$$r_{33} = -C_{\alpha_6} * (S_{\phi_2} S_{\phi_5} - C_{\alpha_4} C_{\phi_2} C_{\phi_5}) - C_{\phi_2} S_{\alpha_4} S_{\alpha_6}$$

$$d_x = a_3 C_{\phi_2} C_{\theta_1} - b_2 S_{\theta_1} - a_6 (S_{\phi_5} (S_{\alpha_4} S_{\theta_1} + C_{\alpha_4} C_{\theta_1} S_{\phi_2}) - C_{\phi_2} C_{\phi_5} C_{\theta_1})$$

$$d_y = b_2 C_{\theta_1} + a_6 (S_{\phi_5} (S_{\alpha_4} C_{\theta_1} - C_{\alpha_4} S_{\phi_2} S_{\theta_1}) + C_{\phi_2} C_{\phi_5} S_{\theta_1}) + a_3 C_{\phi_2} S_{\theta_1}$$

$$d_z = d_1 - a_3 S_{\phi_2} - a_6 (C_{\phi_5} \sin(p_2) + C_{\alpha_4} C_{\phi_2} S_{\phi_5})$$

Este método tienen la ventaja de mayor flexibilidad en la asignación de sistemas de coordenadas a las articulaciones en una cadena cinemática, pues se puede no solamente utilizar los movimientos en x y en z sino además ya se puede utilizar los movimientos alrededor y a lo largo del eje y, se dice entonces que en este método se utiliza el principio de la post-multiplicación matricial.

Al observar estas ecuaciones aparentemente son diferentes a las obtenidas con los otros métodos, esto se debe a que el sistema de coordenadas del origen fue colocado en una posición diferente como se puede observar en la figura 4.11 y de esta misma figura se desprende que las coordenadas de posición del efector final en su estado

inicial son las mismas que se obtienen al reemplazar los valores de las variables articulares iguales a cero en las ecuaciones d_x, d_y, d_z , esto es:

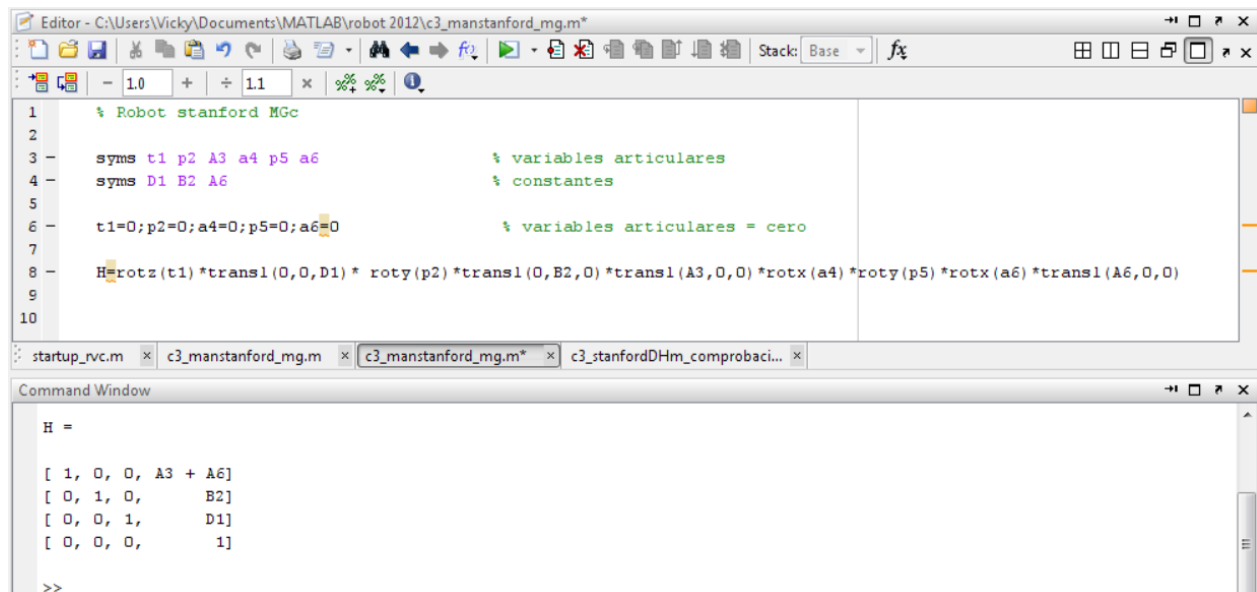
$$d_x = a_3 + a_6$$

$$d_y = b_2$$

$$d_z = d_1$$

En consecuencia se validan los resultados, téngase en cuenta que con mayor facilidad en la asignación de los sistemas de coordenadas sobre las articulaciones del robot, también se requiere menos movimientos en los sistemas de coordenadas móviles como se puede observar en la figura 4.11, la misma presenta un aspecto más diáfano y menos complejo debido a la disminución de movimientos que se tienen que realizar a los sistemas de coordenadas temporales entre las articulaciones.

Utilizando Matlab, como se muestra en la figura 6.4 se obtuvieron los mismos resultados presentados anteriormente.



```

1 % Robot stanford MGc
2
3 - syms t1 p2 A3 a4 p5 a6           % variables articulares
4 - syms D1 B2 A6                   % constantes
5
6 - t1=0;p2=0;a4=0;p5=0;a6=0       % variables articulares = cero
7
8 - H=rotz(t1)*transl(0,0,D1)*rotz(p2)*transl(0,B2,0)*transl(A3,0,0)*rotx(a4)*roty(p5)*rotx(a6)*transl(A6,0,0)
9
10

```

Command Window

```

H =
[ 1, 0, 0, A3 + A6]
[ 0, 1, 0, B2]
[ 0, 0, 1, D1]
[ 0, 0, 0, 1]
>>

```

Figura 6.4 Comprobación del modelo obtenido con el método de MG con movimientos en el eje “y”

6.2 ANÁLISIS DE LOS RESULTADOS DE LA CINEMÁTICA INVERSA

Los métodos que se disponen para resolver el problema de la cinemática inversa son cuatro: método geométrico, método algebraico, método de las matrices inversas y el método del desacoplo cinemático de Pieper. De estos el más efectivo para robots manipuladores de seis grados de libertad en los que está conformado de un brazo y en el extremo del mismo una muñeca, es el método de Pieper y es este el que vamos a analizar.

Lo importante a remarcar en el método de Pieper es que los tres ejes de las últimas articulaciones se cortan en un punto, y en este punto se considera el extremo del brazo del robot, el sistema de coordenadas S_6, S_5 y S_4 coinciden su origen sobre el mismo punto que del origen del sistema de coordenadas S_3 , por esta razón la cinemática inversa según Pieper podríamos considerarle en dos partes, primero el cálculo de la posición de las coordenadas x_5^0, y_5^0 y z_5^0 para ello utilizamos la ecuación vectorial 5.2

$$\vec{P}_5^0 = \vec{P}_6^0 - \vec{d}_6$$

El vector posición \vec{P}_6^0 es conocido pues las coordenadas se saben de antemano. También gracias a la cinemática directa se conoce la matriz *noa* del sistema S_6 con respecto al sistema S_0 . De la matriz *noa* se obtiene el vector unitario \vec{a} y en consecuencia ya se pueden calcular las coordenadas del origen del sistema S_5

Las coordenadas del sistema S_5 se utilizan en el siguiente ecuación matricial y de esta ecuación matricial se pueden determinar las variables articulares q_1, q_2, q_3 , *utilizando las matrices de transformación homogénea*

$$A_3^0 = A_1 * A_2 * A_3$$

El método de Pieper también usa las matrices inversas para generar otras ecuaciones matriciales

$$A_1^{-1} A_3^0 = A_2 * A_3$$

Del apartado 5.2 se obtienen los siguientes resultados

$$\theta_1 = \arcsen \left[\frac{-2d_2 y_5^0 \pm \sqrt{(2d_2 y_5^0)^2 - 4(x_5^{0^2} + y_5^{0^2})(x_5^{0^2} + d_2^2)}}{2(x_5^{0^2} + y_5^{0^2})} \right]$$

$$d_3 = \sqrt{(-x_0^5 \text{sen}(\theta_1) * y_0^5 \text{cos}(\theta_1))^2 + (-z_5^0)^2}$$

$$\theta_2 = \arcsen \left(\frac{-z_5^0}{\sqrt{(-x_0^5 \text{sen}(\theta_1) * y_0^5 \text{cos}(\theta_1))^2 + (-z_5^0)^2}} \right)$$

Nótese que las variables articulares están en función de las coordenadas x_5^0, y_5^0 y z_5^0 , así como también de las longitudes de los eslabones.

Para determinar las variables de las articulaciones de la muñeca q_4, q_5, q_6 , las mismas que no tienen traslaciones pues los sistemas de ejes coinciden en sus orígenes, en tal virtud para determinar q_4, q_5, q_6 , sólo se recomienda utilizar las matrices de rotación en el planteamiento de las ecuaciones matriciales, como se indica a continuación

$$R_6^0 = R_3^0 R_6^3$$

$$R_6^3 = (R_3^0)^{-1} R_6^0$$

$$R_6^3 = (R_3^0)^T R_6^0$$

Del apartado 5.2 se obtienen los siguientes resultados

$$\theta_5 = \arccos(-a_x \cos \theta_2 \text{sen} \theta_1 + a_y \cos \theta_1 \cos \theta_2 - a_z \text{sen} \theta_1)$$

$$\theta_6 = \arccos \left(\frac{-o_x \cos \theta_2 \text{sen} \theta_1 + o_y \cos \theta_1 \cos \theta_2 - o_z \text{sen} \theta_1}{\text{sen} \theta_5} \right)$$

$$\theta_4 = \arcsen \left(\frac{-a_x \cos \theta_1 \text{sen} \theta_1 + a_y \text{sen} \theta_1}{\text{sen} \theta_5} \right)$$

Tanto las matrices de transformación A_i como en las matrices de rotación R_i pueden ser determinadas utilizando cualquier método de la cinemática directa que se estudió en el capítulo 4 de la presente investigación.

7 CAPÍTULO VII

SIMULACIÓN DEL ROBOT STANFORD

Para realizar la simulación del robot Stanford emplearemos el Toolbox Robotics de Matlab, empleando los parámetros de las tablas Denavit-Hartenberg elaboradas en el capítulo 4.

Es importante mencionar que la simulación correcta del robot Stanford o de cualquier otro tipo de robot no sería posible sin un análisis gráfico previo, ya que los parámetros de las tablas Denavit-Hartenberg estándar y Denavit-Hartenberg modificada deben ser ingresados al software para la simulación del robot manipulador. En otras palabras, la correcta simulación mediante el software, solamente será posible si previamente se ha encontrado el modelo matemático del robot.

7.1 TOOLBOX ROBOTICS DE MATLAB

El Toolbox robotics de Matlab, es una herramienta desarrollada por Peter Corke, profesor de control y robótica en la universidad de tecnología de Queensland en Australia. Esta herramienta nos permite modelar robots manipuladores con un número definido de articulaciones. Permite describir la posición y orientación del extremo del robot a través de diferentes herramientas matemáticas, además permite realizar cálculos de cinemática y dinámica y generación de trayectorias de robots industriales.

Permite la representación de la posición y orientación del extremo del robot a través de vectores, matrices de rotación, transformaciones homogéneas, y cuaternios [29]

7.1.1 PASOS PARA LA CREACIÓN DE UN ROBOT UTILIZANDO EL TOOLBOX ROBOTICS DE MATLAB

Una forma de crear o definir un nuevo robot utilizando el Toolbox Robotics de Matlab, es realizando primero una descripción de cada articulación o eslabón del robot. La función que se utiliza es la siguiente:

$$L_i = \text{link}([\alpha_i \ a_i \ \theta_i \ d_i \ \sigma_i])$$

Donde,

α_i = ángulo de rotación alrededor del eje x. El signo lo da la regla de la mano derecha

a_i = distancia recorrida a lo largo del eje x. El signo lo define el sentido del eje x_i

θ_i = ángulo de rotación alrededor del eje z. El signo lo da la regla de la mano derecha

d_i = distancia recorrida a lo largo del eje z. En el caso de articulaciones prismáticas será la variable de desplazamiento.

σ_i = 0 (rotación) ó 1 (prismática)

Los primeros cuatro parámetros son los definidos en las tablas de Denavit-Hartenberg y el último parámetro define el tipo de articulación, ya sea de rotación o prismática.

Una vez definida cada articulación el siguiente paso es crear un objeto del tipo robot. Para esto se utiliza la siguiente función, cuyo parámetro es un arreglo con la descripción de cada eslabón:

$$\text{variable_robot} = \text{robot}(\{L1 \ L2 \ \dots \ L_i\})$$

Para obtener una representación gráfica simplemente se utiliza:

$$\text{plot}(\text{variable_robot}, q)$$

Donde, q es un vector con los ángulos para cada articulación.

7.2 SIMULACIÓN DEL ROBOT STANFORD

Las funciones descritas en el apartado 7.1.1 y los parámetros de la tabla Denavit-Hartenberg estándar son utilizados en la simulación del Robot Stanford en Matlab.

Como se mencionó anteriormente primero debemos definir los parámetros de Denavit-Hartenberg para cada articulación.

Una vez establecidos los valores de cada parámetro es posible escribir el código en Matlab para la definición completa del Robot Stanford.

Para crear un objeto del tipo robot $srx = robot(\{L1 L2 L3\});$

Si se desea asignar nombre al robot $srx.name = 'RobotStanford';$

Por último para mostrar la representación gráfica del robot $plot(srx, [0 0 0]);$

Se realizará la simulación del Robot Stanford, siguiendo el convenio de Denavit-Hartenberg estándar y el convenio de Denavit Hartenberg modificado. Los datos a ser ingresados en el programa son los parámetros de las tablas D-H determinados en el capítulo cuatro.

7.2.1 MÉTODO DE DENAVIT-HARTENBERG ESTÁNDAR

Para la simulación del robot Stanford de seis grados de libertad, nos basamos en la tabla de parámetros D-H estándar, ver tabla 7.1.

Tabla 7.1 Parámetros D-H estándar para simulación manipulador Stanford.

Eslabón	θ_i	d_i	a_i	α_i
1	$\theta_1^* + 90^\circ$	d_1	0	-90°
2	$\theta_2^* + 90^\circ$	d_2	0	90°
3	0	d_3^*	0	0°
4	θ_4^*	0	0	-90°
5	θ_5^*	0	0	90°
6	θ_6^*	d_6	0	0

*variable de la articulación

En la figura 7.1 se muestra la programación realizada en Matlab para la simulación del manipulador Stanford. Como podemos observar el orden de las columnas de la tabla de parámetros Denavit- Hartenberg estándar no es coincidente con el orden de las columnas de las tablas del Toolbox Robotics desarrollado por Peter Corke, por lo que se debe prestar especial cuidado al momento de ingresar estos parámetros. Además nótese que se establece un valor para los parámetros d_1, d_2, d_6 que representan la longitud de los eslabones.

```

1  % Simulación manipulador Stanford 6GDL
2  %Parámetros tabla D-H estándar
3
4  %      alfa  A  teta D  R/P
5
6  L1=link([-pi/2  0  pi/2  10  0], 'standard');
7  L2=link([pi/2  0  pi/2  20  0], 'standard');
8  L3=link([0  0  0  0  1], 'standard');
9  L4=link([-pi/2  0  0  0  0], 'standard');
10 L5=link([pi/2  0  0  0  0], 'standard');
11 L6=link([0  0  0  10  0], 'standard');
12
13 r=robot({L1 L2 L3 L4 L5 L6})
14 r.name='RobotStanford_e'
15 plot(r,[0 0 0 0 0 0])
16 drivebot(r)

```

Figura 7.1 Programación para Simulación robot Stanford estándar

En la figura 7.2 se muestra la simulación del robot Stanford, se puede apreciar la configuración del robot Stanford con sus articulaciones y eslabones, así como también los movimientos que realiza el robot en el espacio.

En la figura 7.3 se muestra el comando de las barras deslizables que nos permiten ir variando los parámetros articulares y observar qué sucede con la posición y orientación del efector final del robot manipulador

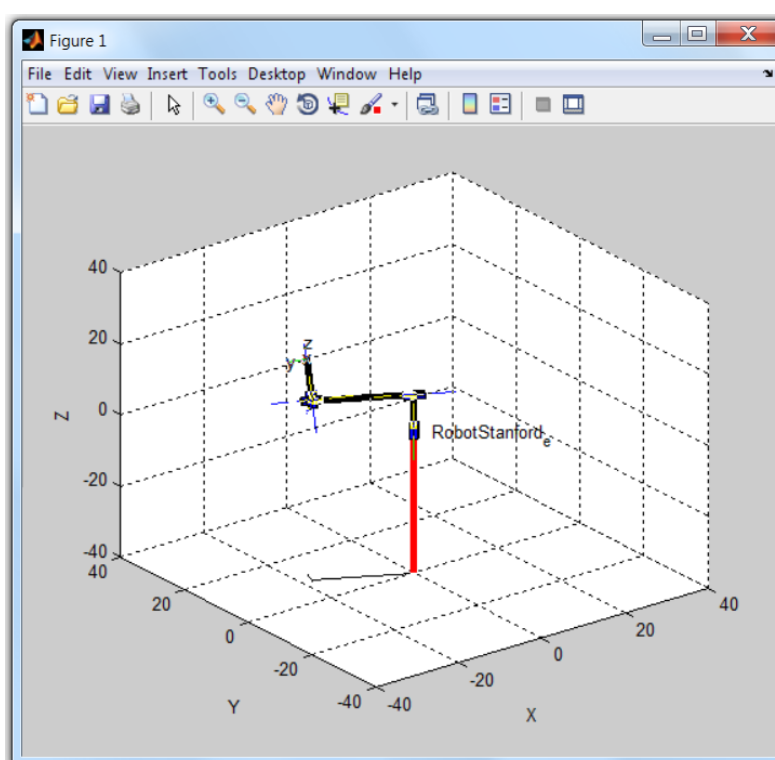


Figura 7.2 Simulación Robot Stanford D-H estándar

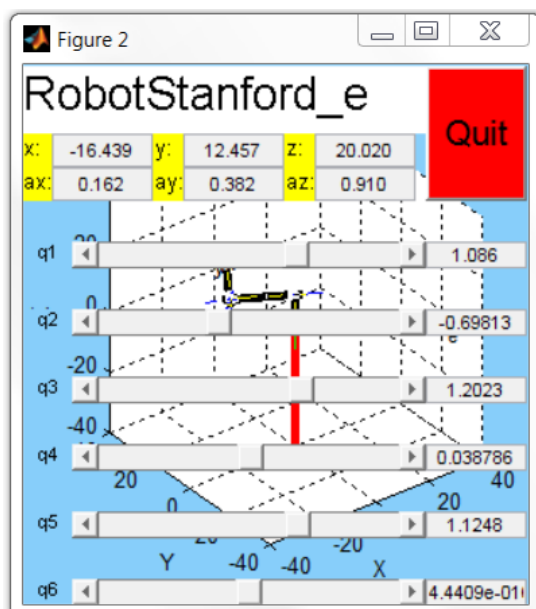


Figura 7.3 Comando para variación de parámetros articulares del Robot Stanford D-H estándar

7.2.2 MÉTODO DE DENAVIT-HARTENBERG MODIFICADO

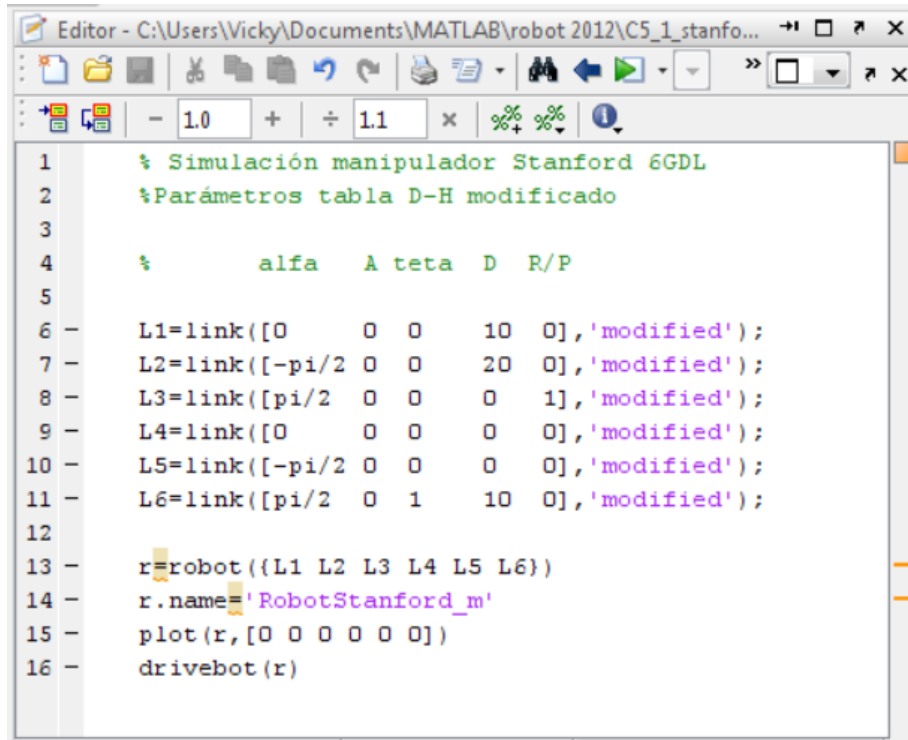
Al igual que con el método de Denavit-Hartenberg estándar, se procedió a realizar la simulación del Robot Stanford de seis grados de libertad basándonos en el convenio de Denavit-Hartenberg modificado. Como se mencionó anteriormente, es necesario ingresar al programa los parámetros establecidos en la tabla de parámetros D-H modificado. Ver tabla 7.2

Tabla 7.2 Parámetros D-H modificado para simulación del manipulador Stanford.

Eslabón	α_i	a_i	θ_i	d_i
1	0	0	$\theta_1^* + 90^\circ$	d_1
2	-90°	0	$\theta_2^* + 90^\circ$	d_2
3	90°	0	0	d_3^*
4	0	0	θ_4^*	0
5	-90°	0	θ_5^*	0
6	90°	0	θ_6^*	d_6

*variable de la articulación

Estos datos son ingresados en el software de Matlab, para obtener la simulación del Robot Stanford con el método de Denavit-Hartenberg modificado. La programación realizada en el software se muestra en la figura 7.4



```

1  % Simulación manipulador Stanford 6GDL
2  %Parámetros tabla D-H modificado
3
4  %      alfa  A teta  D  R/P
5
6  -  L1=link([0      0  0      10  0], 'modified');
7  -  L2=link([-pi/2 0  0      20  0], 'modified');
8  -  L3=link([pi/2  0  0      0   1], 'modified');
9  -  L4=link([0      0  0      0   0], 'modified');
10 -  L5=link([-pi/2 0  0      0   0], 'modified');
11 -  L6=link([pi/2  0  1      10  0], 'modified');
12
13 -  r=robot({L1 L2 L3 L4 L5 L6})
14 -  r.name='RobotStanford_m'
15 -  plot(r,[0 0 0 0 0 0])
16 -  drivebot(r)

```

Figura 7.4 Programación para Simulación robot Stanford modificado

De igual manera, hay que tener mucho cuidado al momento de ingresar los parámetros de la tabla al programa, ya que el orden de las columnas es diferente.

Podemos notar que la programación básicamente es la misma, solamente cambian los parámetros a ser ingresados por el usuario en base a las tablas D-H y la asignación de las palabra 'standard' o 'modified' según sea el caso.

En la figura 7.5 podemos observar la gráfica de la simulación del Robot Stanford de seis grados de libertad obtenida, mediante la programación, utilizando el método D-H modificado y en la figura 7.6 se presentan los comandos para la variación de las variables articulares que nos permitirán representar las diversas posiciones y orientaciones que el robot manipulador puede lograr.

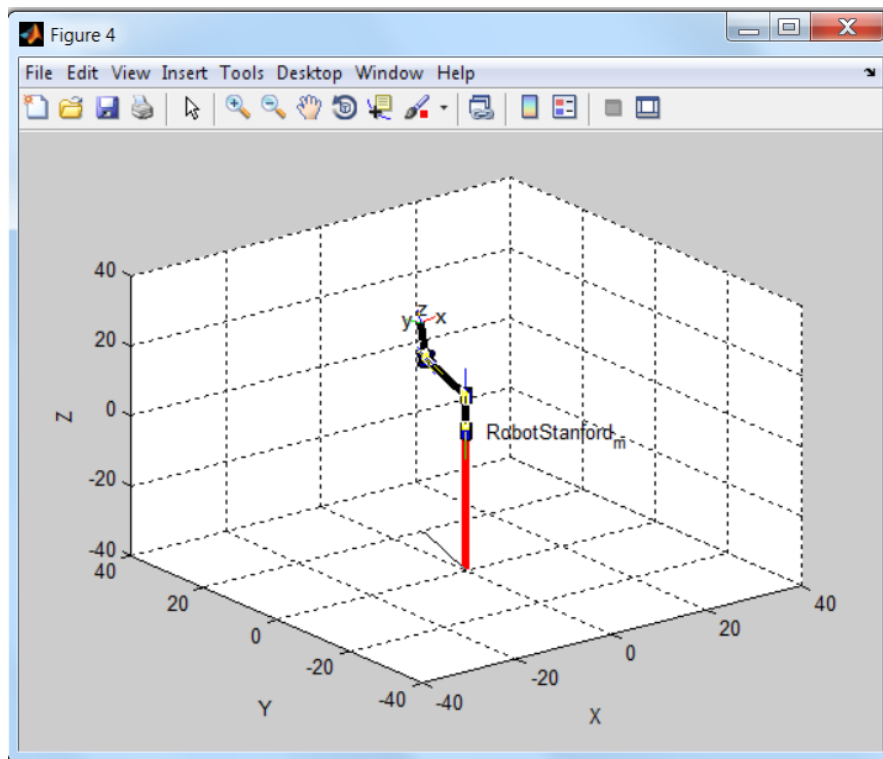


Figura 7.5 Simulación Robot Stanford D-H modificado

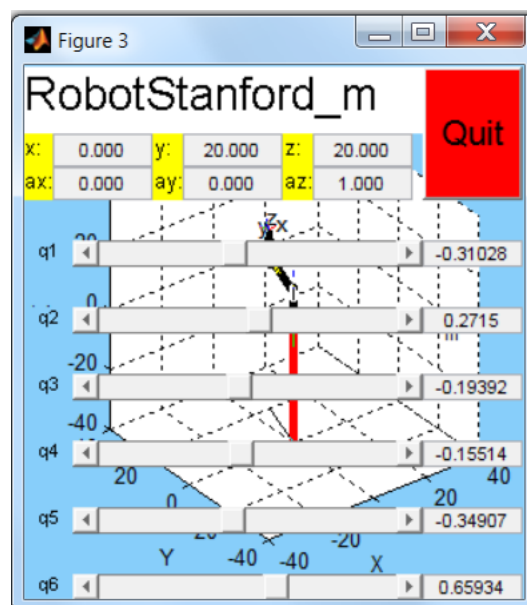


Figura 7.6 Comando para variación de parámetros articulares del Robot Stanford D-H modificado

8 CAPITULO VIII

CONCLUSIONES Y RECOMENDACIONES

8.1 CONCLUSIONES

Se cumplió el objetivo de modelar el robot manipulador Stanford. La modelación matemática fue realizada por tres métodos diferentes: convenio de Denavit-Hartenberg estándar, convenio de Denavit-Hartenberg modificado y por el método del movimiento general continuo. Cada una de las modelaciones fueron validadas en la presente tesis.

En el presente proyecto de titulación se utiliza el principio de la multiplicación matricial en la misma secuencia del movimiento de los sistemas de coordenadas, este principio es la base fundamental de la robótica y nos permite formular el modelo matemático del brazo articulado Stanford.

La matriz de transformación homogénea del brazo H sirve para vincular al sistema de coordenadas del efector final con el sistema de coordenadas de la base. A partir de esta, se pueden desprender las ecuaciones del movimiento en una cadena cinemática abierta conformada por eslabones y articulaciones que son típicamente la configuraciones de los robots manipuladores.

La representación simbólica del robot facilita la asignación de coordenadas ya que nos permite tener una mejor perspectiva del robot y de sus movimientos, la obtención de los parámetros necesarios para la modelación podrán ser realizados con mayor rapidez y con menor posibilidad de error, y por consiguiente la modelación podrá ser realizada de una manera más efectiva y eficiente.

Al producto matricial realizado en la misma secuencia de los movimientos se le conoce como el principio de la postmultiplicación, y se comprobó la validez de este principio en la modelación del robot Stanford utilizando el convenio de Denavit-Hartenberg estándar,

así como también el convenio de Denavit-Hartenberg modificado y el método del movimiento general continuo.

Un método simple y efectivo para la validación del modelo matemático de robots manipuladores, es el aplicado en el capítulo 3 de este documento, que consiste en definir a todas las variables articulares del robot con el valor de cero y verificar que las coordenadas de traslación obtenidas al reemplazar estos valores en las ecuaciones dx , dy , dz de la matriz homogénea sean iguales al valor de las coordenadas del efector final del robot en su estado inicial, ya que al igualar todas las variables articulares a cero significa que no ha existido movimiento del brazo articulado.

El presente proyecto establece el modelo matemático del Robot Stanford, en base al cual se pueden realizar proyectos de aplicación, tales como diseño ó construcción de Robot Stanford para diferentes usos.

Pese a que los softwares computacionales son de gran ayuda para el usuario en robótica, la obtención del modelo de un robot no sería posible sin un análisis y estudio previo del robot en estudio por parte del usuario, ya que los parámetros de las tablas Denavit-Hartenberg estándar y Denavit-Hartenberg modificada deben ser ingresados al software para la modelación y simulación del robot manipulador.

En el método de Denavit-Hartenberg estándar se asigna el sistema de coordenadas i sobre la línea de eje de la articulación $i+1$, en este caso se toma en cuenta los movimientos realizados al sistema de coordenadas móviles y se forma grupos de cuatro movimientos, estos movimientos se resume en cada fila de la tabla de parámetros, según este convenio es importante la secuencia , primero sobre el eje z (rosca en z) y luego sobre el eje x (rosca en x), la notación es: rotación alrededor de z , θ , traslación a lo largo de z , d ; traslación a lo largo de x , a , rotación alrededor de x , α ; con estos cuatro movimientos se forman las filas que indica de la tabla de parámetros, si uno de estos movimientos no es necesario realizar se llena con cero.

En el método de Denavit-Hartenberg modificado, se asigna al sistema de coordenadas i sobre la línea de eje de la articulación i , se toma en cuenta para formar los grupos de movimientos que se realizó al sistema de coordenadas móviles sobre el eslabón $i-1$, según este convenio la secuencia es primero sobre el eje x (rosca en x) y luego sobre el eje z (rosca en z), la notación es: giro alrededor de x , α , traslación a lo largo de x , a , giro alrededor de z , θ , traslación a lo largo de z , d .

El método de Movimiento General continuo, MGc; es el más amigable y eficiente de los métodos de modelación ya que no impone restricciones para la asignación de los sistemas de coordenadas, simplemente establece que se debe seguir la secuencia natural del movimiento para pasar de una articulación a otra hasta llegar a efector final.

Ya que la manipulación de piezas llevada a cabo por un robot implica el movimiento espacial de sus eslabones, debido a la variación de sus variables articulares, es sumamente importante saber cómo se representa la posición y la orientación del efector final del robot para conocer con precisión en qué punto del espacio de trabajo se colocará el mismo. Para esto es necesario estudiar la representación y transformación de los movimientos de los movimientos típicos de un robot manipulador,

El problema de la cinemática directa presenta una única solución, mientras que el problema de la cinemática inversa puede dar lugar a múltiples soluciones, debido a que con parámetros articulares conocidos (cinemática directa) el robot únicamente podrá tener una posición final, pero al tratar de determinar los parámetros articulares en base a una posición (cinemática inversa) pueden existir diferentes configuraciones articulares con las que obtener la misma configuración del efector final, o en el peor de los casos puede que no exista solución (por ejemplo en una posición no alcanzable). Por todo esto se concluye que la resolución del problema de la cinemática inversa es más complicado que el problema de la cinemática directa.

8.2 RECOMENDACIONES

Al emplear el método de Denavit-Hartenberg se recomienda seguir los pasos del algoritmo D-H para evitar posibles fallas en la asignación de los sistemas de coordenadas y posterior modelación del robot manipulador.

El software libre Matlab es sumamente útil al momento de trabajar con matrices, y debido a que para la modelación de robots, es necesario trabajar con gran cantidad de matrices de gran tamaño es recomendable utilizar como ayuda este tipo de herramienta.

Posteriormente a la obtención de los modelos matemáticos de cualquier tipo de robot, este debe ser validado, para esto se recomienda asignar el valor de cero a las variables articulares y reemplazarlas en las ecuaciones obtenidas en la matriz homogénea; los resultados deben ser los mismos que cuando el robot se encuentra en su estado inicial.

Se recomienda como punto de partida para la modelación, establecer cuáles son las variables articulares del sistema, ya que las mismas serán nuestra referencia para la tabla de parámetros D-H ó MGc.

Se recomienda el uso y difusión del método del movimiento general continuo para modelación de robots manipuladores, al ser el más amigable con el usuario.

Para la solución del problema de la cinemática inversa en robots manipuladores se recomienda la aplicación del método del desacoplo cinemático, el cual estudia de manera independiente el brazo y la muñeca del robot, lo que facilita la solución del problema. Así, dada una posición y orientación final deseada se establece la posición del punto de corte (la muñeca del robot) calculando los valores de q_1 , q_2 y q_3 y a continuación a partir de los datos de orientación y los ya calculados se obtienen los valores del resto de las variables articulares.

Se recomienda realizar la representación simbólica del robot a ser estudiado, en la que las articulaciones de rotación son representadas con cilindros, las articulaciones prismáticas por cubos y los eslabones por líneas que unen estas articulaciones; este tipo de representación nos permite tener una idea más clara de los componentes y de los posibles movimientos del robot estudiado, lo que facilita la asignación de sistemas de coordenadas y la determinación de los parámetros necesarios para la modelación.

Se recomienda emplear los resultados matemáticos de la modelación, así como también los resultados analíticos presentados en el presente proyecto como la base para un futuro proyecto de construcción y/o control de un robot Stanford de seis grados de libertad.

9 BIBLIOGRAFIA

- [1] Balasubramanian, R. (2011). The Denavit Hartenberg Convention. USA: Robotic Insitute Carnegie Mellon University
- [2] Barrientos A. Peñin L. Balaguer C. Aracil R. (1997). Fundamentos de Robótica. España: Ed. Mc Graw Hill
- [3] Corke, P. (1996). Robotics TOOLBOX. Australia: CSIRO
- [4] Corke, P. (2014). Robotics TOOLBOX. Australia: CSIRO
- [5] Craig, J. (2005). Introduction to robotics mechanics and control. 3rd edition. USA: PEARSON Prentice Hall.
- [6] Fu K., González R., Lee C., Robótica, Control, detección, visión e inteligencia. España: Ed. Mc Graw Hill.
- [7] Granja, M. (2014). Comparación del método de denavit-hartenberg estándar con el modificado en la modelación de la robótica. Ecuador: EPN (Tesis inédita de maestría)
- [8] Jochheim A. Gerke M. Bischoff A. (1999). Modeling and simulation of kinematic systems. Alemania: University of Hagen.
- [9] Mabie, H. (1987). Mechanism and Dynamics of Machinery. USA: John Wiley & Son.
- [10] Marghithu D. (2009) Mechanisms and Robots Analysis with MATLAB. USA: Ed. Springer.
- [11] Miranda. J. (2003). Mecanismos. México: Instituto Tecnológico y de Estudios Superiores de Monterrey.
- [12] Ollero A. (2001). Robótica, manipuladores y robots móviles. España: Ed Marcombo.
- [13] Sandler B. (1999). ROBOTICS Designing the Mechanisms for Automated Machinery. USA: Ed Ben Gurion of the Negev, Beersheva, Israel.
- [14] Santos V. (2004). Robótica Industrial. Portugal: Universidad de Aveiro.
- [15] Shigley, J. Uicker Jr. J. (). Teoría De Maquinas Y Mecanismos. Mexico: Ed. Mc Graw Hill.

- [16] Spong, M. Hutchinson, S. Vidyasagar, M. (2004). Robot Dynamics and Control. USA: ED. John Wiley & Sons, Inc.
- [17] Vinogradov, O. (2000). Fundamentals of Kinematics and Dynamics of Machines and Mechanisms. USA: CRC Press.
- [18] http://catarina.udlap.mx/u_dl_a/tales/documentos/lis/sanchez_v_ma/capitulo2.pdf
- [19] http://datateca.unad.edu.co/contenidos/299012/Protocolo_Robotica_Avanzada.pdf
- [20] <http://docslide.net/documents/curso-toolbox-robotica.html>
- [21] http://en.wikipedia.org/wiki/Denavit%E2%80%93Hartenberg_parameters
- [22] <http://isa.umh.es/asignaturas/rvc/>
- [23] http://petercorke.com/Toolbox_software.html
- [24] <http://proton.ucting.udg.mx/materias/robotica/>
- [25] <http://www.cs.cmu.edu/~rapidproto/mechanisms/chpt4.html#toc>
- [26] http://www.maplesoft.com/content/EngineeringFundamentals/14/MapleDocument_14/Forward%20Kinematics.pdf
- [27] <http://www.motiongenesis.com/MGWebSite/MGGetStarted/MGExampleStanfordArm/MGStanfordArm.html>
- [28] <http://www.petercorke.com/RVC/>
- [29] <http://www2.elo.utfsm.cl/~elo377/documentos/TMA-RoboticaJHS.pdf>