

UNIVERSIDAD POLITÉCNICA DE PUEBLA



TESIS DE MAESTRÍA



Estimación Vía Cuaterniones De La Orientación De Un Robot De 6GDL Usado En La Industria Automotriz

Autor:

Luis Castillo Bermúdez

Director de Tesis:

Dr. Bernardino Benito

Salmeron Quiroz

*Esta tesis se presenta como un requisito
para obtener el grado de Maestría en Ingeniería*

en

Automatización de Procesos Industriales
Departamento de Posgrado

20 de agosto de 2014



“El hecho fundamental nunca falla; su prueba siempre es verdadera.”

Michael Faraday

UNIVERSIDAD POLITÉCNICA DE PUEBLA

Resumen

Automatización de Procesos Industriales

Departamento de Posgrado

Maestría en Ingeniería

**Estimación Vía Cuaterniones De La Orientación De Un Robot De 6GDL
Usado En La Industria Automotriz**

por Luis Castillo Bermúdez

En el desarrollo de este trabajo se presenta el desarrollo del modelo rotacional de un robot de 6 grados de libertad mediante cuaterniones, mediante este modelo se puede comprobar la cinemática directa e inversa a través de una interfaz gráfica de usuario desarrollada en comprobando los resultados de este desarrollo mediante la adquisición de datos generados por un robot IRB1600 de la empresa ABB.

Agradecimientos

Al Dr. Bernardino Benito Salmeron Quiroz por su asesoría y apoyo durante la realización de esta tesis"

Al CONACYT por el apoyo que me ha otorgado

A la Universidad Politécnica de Puebla por la formación que me ha brindado

A mis compañeros, amigos y profesores del posgrado

Finalmente, mi familia - mi esposa e hijos - han estado esperando mucho tiempo por más atención de mi parte, para pasar tiempo de familia conmigo. Les agradezco su paciencia, comprensión, apoyo y cooperación.

Índice

Resumen	III
Agradecimientos	IV
Lista de figuras	VII
Lista de tablas	IX
Abreviaciones	XI
1. Introducción	1
2. Estado del arte	5
2.1. Los cuaterniones de Hamilton	5
2.2. Robótica industrial	7
2.2.1. Definición de robótica industrial	8
2.2.2. Tipo de movimiento en las articulaciones de los robots	10
2.3. Cinemática de los robots	12
2.3.1. Cinemática directa	14
2.3.2. Cinemática inversa	37
2.4. Los cuaterniones en la robótica	43
3. Fundamentos matemáticos de los cuaterniones	45
3.1. Historia del álgebra de cuaterniones	45
3.1.1. Definición de cuaternión	47
3.1.2. Igualdad y adición	48
3.1.3. Producto	49
3.1.4. Complejo conjugado	53
3.1.5. Norma del cuaternión	53
3.1.6. Interpretación geométrica del cuaternión	54
4. Modelado vía cuaterniones	67

4.1. Modelado rotacional mediante cuaterniones	67
4.1.1. Relación del espacio vectorial de cuaterniones	68
4.1.2. Representación paramétrica de las rotaciones de un cuerpo rígido	70
4.1.3. Rotaciones sucesivas de un sistema cartesiano de referencia	71
4.2. Robot ABB-IRB1600	74
4.2.1. Estructura paramétrica del IRB1600	75
4.2.2. Modelo matemático del IRB1600	76
4.2.3. Solución de la cinemática directa	83
4.2.4. Solución cinemática inversa	85
5. Simulación VRLM e interfaz gráfica.	91
5.1. Modelado en realidad virtual	91
5.1.1. Desarrollo del modelo en realidad virtual	93
5.1.2. Animación del modelo VRML mediante MatLab®	95
5.2. Creación de la GUI en MatLab®	96
5.2.1. GUI Cinemática directa	96
5.2.2. GUI Cinemática inversa	100
6. Validación física del modelo	105
6.1. Medición de parámetros en el robot	105
6.1.1. Girómetros	106
6.1.2. Acelerómetros	107
6.1.3. Magnetómetros	108
6.1.4. Brazo articulado	109
6.2. Validación física del modelo rotacional	110
7. Conclusiones	115
7.1. Trabajo futuro	116
A. Robot IRB1600	119
B. Algoritmo del modelo rotacional del robot IRB1600	123
C. Programación de la GUI	125
D. Unidad inercial	129
E. Recursos computacionales usados en el modelado por cuaterniones y el modelo convencional DH	133
Referencias	137

Índice de figuras

2.1. Robot manipulador industrial	9
2.2. Robot con movimiento según coordenadas cartesianas.	11
2.3. Robot con movimiento según coordenadas cilíndricas.	11
2.4. Robot con movimiento según coordenadas polares en una estructura esférica.	12
2.5. Robot con movimiento según coordenadas polares en una estructura articulada.	12
2.6. Relación entre la cinemática directa e inversa.	13
2.7. Representación de ambos sistemas de coordenadas ligados a un cuerpo.	15
2.8. Representación gráfica de las matrices de rotación.	19
2.9. Primer sistema de ángulos <i>eulerianos</i>	22
2.10. Segundo sistema de ángulos	23
2.11. Tercer sistema de ángulos de Euler	24
2.12. Ángulos de Euler (Gimbal Lock).	25
2.13. Articulaciones y elementos de un Robot PUMA	30
2.14. Articulaciones y elementos de un robot	31
2.15. Sistema de coordenadas de eslabones y sus parámetros	31
2.16. Robot IRB6400C	36
2.17. Soluciones Admisibles de configuración	38
2.18. Desacoplo Cinemático	41
2.19. Brazo planar de 3 articulaciones	42
3.1. Correspondencia: Vectores \leftrightarrow Cuaterniones	56
3.2. Acción del operador rotacional en termino de cuaterniones	59
3.3. Construcción de dos rotaciones	62
3.4. Secuencia de rotaciones	63
4.1. Sistema de rotaciones de un sistema de referencia	73
4.2. Robot IRB1600 de ABB	75
4.3. Sistemas de referencia de los robots.	76
4.4. Sistemas de referencia del robot.	77
4.5. Relaciones de los sistemas de referencia.	78
4.6. Relación de ejes rotacionales y ángulos de giro.	79

4.7. Definición de los sistemas de coordenadas en cada articulación.	84
4.8. Relación de la orientación de la muñeca del robot.	87
4.9. Relación de la posición de la base del robot.	88
5.1. Ejemplo de realidad virtual.	92
5.2. Base del robot creada mediante VRML.	94
5.3. Union de los cuerpos del robot.	94
5.4. GUI para la animación VRML del robot.	97
5.5. Selección del problema cinemático.	97
5.6. Selección codo arriba/codo abajo.	97
5.7. Selección de los desplazamientos angulares.	98
5.8. Botones de animación.	98
5.9. Botones de animación.	98
5.10. Cuaterniones de rotación.	99
5.11. Representación de la animación MatLab [®] /VRML.	100
5.12. Esquema de funcionamiento de la GUI.	101
5.13. Campos θ_1 , θ_2 y θ_3 desactivados.	101
5.14. Selección de la orientación deseada, para la cinemática inversa.	102
5.15. Selección de la posición deseada, para la cinemática inversa.	102
5.16. Calculo de las variables θ_1 , θ_2 y θ_3	103
5.17. Calculo de las variables θ_1 , θ_2 y θ_3 , selección de la configuración <i>Codo Arriba y Codo Abajo</i>	103
5.18. Animación mediante la configuración codo arriba.	104
5.19. Animación mediante la configuración codo abajo.	104
6.1. Principio de funcionamiento de un giroscopio	107
6.2. Gráfica de datos obtenidos mediante la IMU Pololu MinIMU-9 v2	110
6.3. Animación obtenida mediante los valores tomados de la IMU	111
6.4. Cuaterniones vs Matrices de rotación para la primer articulación	111
6.5. Cuaterniones vs Matrices de rotación para la segunda articulación	112
6.6. Cuaterniones vs Matrices de rotación para la tercera articulación	112
6.7. Cuaterniones vs Matrices de rotación para la cuarta articulación	113
6.8. Cuaterniones vs Matrices de rotación para la quinta articulación	113
6.9. Cuaterniones vs Matrices de rotación para la sexta articulación	114

Índice de tablas

2.1. Tres tipos de representación de ángulos de Euler	21
2.2. Representación D-H del robot IRB6400C	36
2.3. Complejidad computacional: cuaterniones vs ángulos de Euler	43
4.1. Representación Denavit-Hartenberg del robot IRB1600	85
4.2. Representación del modelo rotacional basado en cuaterniones del robot IRB1600.	85

Abreviaciones

MATLAB	MATrix LABoratory
VRML	Virtual Reality Modeling Language
LabVIEW	Laboratoy Virtual Instrumentation Engineering Workbench
GDL	Grados De Libertad
3D	3 Dimensiones
RIA	Robotics Industries Association
ISO	International Organization for Standardization
AFNOR	Asociación Francesa de Normalización
GUI	Graphic User Interface

Dedicatoria

Con todo mi cariño y mi amor para las personas que hicieron todo en la vida para que yo pudiera lograr mis sueños, por motivarme y darme la mano cuando sentía que el camino se terminaba, a ustedes por siempre mi corazón y mi agradecimiento.

Papá y mamá

Capítulo 1

Introducción

En el estudio de las aplicaciones en Mecatrónica ha sido necesario desarrollar modelos de análisis y simulación de sistemas mecánicos, tal es el caso de los mecanismos robóticos. Para la moderación y análisis es necesario hacer uso de herramientas matemáticas, que sean eficientes con las cuales se puedan construir modelos cinemáticos o dinámicos de los robots manipuladores. Las herramientas matemáticas utilizadas requieren de considerables conocimientos en el área para el desarrollo de los modelos cinemáticos.

La cinemática directa cartesiana de los robots manipuladores permite una descripción entre las coordenadas cartesianas del extremo final del robot y las coordenadas de las articulaciones. La convención Denavit - Hartenberg es una herramienta de suma utilidad para la obtención del modelo cinemático directo de los robots manipuladores, existen algunos otros métodos para llevar a cabo la solución de la cinemática directa, como lo son: los ángulos de Euler, matrices de transformación homogénea y el uso de cuaterniones.

La cinemática inversa consiste en encontrar los valores que deben adoptar las articulaciones del robot para que su extremo (efector final) se posicione y oriente según una determinada localización espacial. Para la solución de la cinemática inversa no existe una determinación sistemática como en el caso de la cinemática directa, por lo tanto, el procedimiento para la obtención de las ecuaciones depende fuertemente de la configuración del robot. Algunos de estos métodos que se han desarrollado son: las matrices Jacobianas, desacoplo cinemático y métodos geométricos, estos si bien resuelven los

problemas de dichos modelos hasta cierto número de grados de libertad se convierte en un problema complejo cuando el número de dichos grados de libertad del robot crece, ya que el inconveniente de estos procedimientos es que se trata de métodos numéricos iterativos, cuya velocidad de convergencia e incluso su convergencia en sí no está siempre garantizada [1].

En 1843 el matemático Irlandés Sir William Rowan Hamilton^a descubrió los cuaterniones con el fin de extender el álgebra vectorial tridimensional para la inclusión de la multiplicación y la división. Sin embargo, se ha encontrado que el álgebra de vectores ordinarios proporciona un mejor medio matemático para la investigación de problemas físicos. No obstante, es verdad que el álgebra del cuaternión provee una representación simple y elegante para describir rotaciones finitas en el espacio [2].

Los cuaterniones son un conjunto de cuatro parámetros, los cuales requieren satisfacer la condición de normalidad (su norma es la unidad). La representación de los cuaterniones para describir la orientación de un sistema de coordenadas ha tenido varias ventajas sobre las representaciones convencionales (ángulos de Euler y cosenos directores). Primero, un sistema mecánico que involucra rotación entre varios sistemas de coordenadas no se degenera para ninguna orientación angular. Segundo, los requerimientos computacionales usando cuaterniones son mucho menor que el uso de cosenos directores o ángulos de Euler aún cuando se aplica multiplicación entre cuaterniones. Además, las cantidades físicas pertenecientes al movimiento de la rotación tales como desplazamiento angular, velocidad, aceleración y momento se derivan en términos de cuaterniones con mayor facilidad. La manipulación de ecuaciones es más fácil cuando se usa álgebra de cuaterniones.

Por otra parte cuando se hace referencia a la “*animación*” significa “*traer a la vida*”[3], la animación es una representación visual de del cambio. Durante algunos años se a usado en el mundo del entretenimiento, los video - juegos. La simulación computacional es una herramienta poderosa utilizada por las industrias para el diseño y la modificación de los sistemas productivos.

^aSir William Rowan Hamilton, 4 de Agosto 1805 - 2 de Septiembre 1865(Historia de las matemáticas
Página web de la universidad de St. Andrews, Scotland)

Como ya se menciona simular es sinónimo de imitar el comportamiento de un sistema con algún propósito específico, en el caso de la simulación computacional es necesario programar los modelos cinemáticos en algún tipo de software el cual permite visualizar el comportamiento de dicho modelo, existen modelos en los cuales para realizar la simulación es necesario el uso de más de un software, cuando el modelo requiere hacer uso de más de un software se utiliza uno para el procesamiento matemático (software's como MatLab, Simulink, Octave) y algún otro para la realización de la parte visual donde se pueda observar el movimiento o la representación del cambio que se ha generado, para lo cual se utilizan software's como OpenGL, VRLM, por citar algunos. Teniendo en consideración la compatibilidad de los sistemas a ocupar para el desarrollo y validación de dichas simulaciones.

Se han llevado a cabo investigaciones las cuales se centran en desarrollar modelos que puedan ser de fácil comprensión y que su solución matemática sea cada vez más sencilla, una de ellas es el uso de los cuaterniones para el desarrollo de los modelos, en diferentes ramas de la ciencias como el mencionado en el libro "Visualizing Quaternions [4], The Apollo 10 Gimbal - Lock Incident", otro caso es la obtención de los modelos cinemáticos directo e inverso de robots manipuladores mediante el uso de cuaterniones, los cuales ya que han probado ser modelos eficientes (proporcionando soluciones cerradas) para robots de hasta 3 GDL.

En este proyecto de tesis se propone el uso de cuaterniones para determinar la orientación y posición de un robot de 6 grados de libertad (GDL), así obtener la cinemática directa e inversa del robot IRB1600 de la empresa ABB.

Para la estimación de la orientación y posición, se utilizarán cuaterniones en lugar de ángulos de Euler para eliminar el problema de las singularidades asociadas a la estimación de la orientación, la simulación del modelo sera realizado mediante VRLM, ya que es un lenguaje de programación sencillo que puede interactuar con Software de procesamiento matemático, como Matlab, además de ser un metalenguaje que no requiere grandes conocimientos de animación o capacidades de procesamiento para realizar dichas simulaciones.

Capítulo 2

Estado del arte

Una vez que Hamilton publicó los resultados de su investigación, no paso mucho tiempo para que matemáticos como Josiah Willard Gibbs, entre otros, realizaran trabajos detallados de lo que ahora se conoce como el algebra del espacio vectorial. A consecuencia de esto, el trabajo que había desarrollado Hamilton se vio opacado teniendo como resultado el desinterés de su investigación [5].

Sin embargo, el desarrollo tecnológico en áreas como la aeronáutica, animación y la robótica han despertado nuevamente el interés en el estudio de los cuaterniones del tal manera que se han estudiado nuevas formas con las cuales el álgebra de cuaterniones muestre un mayor grado de efectividad comparado con el uso del álgebra vectorial ordinaria.

A lo largo de este capítulo se presentará cómo las formas convencionales para la obtención de la cinemática de los robots manipuladores han contribuido a la ciencia, así como también se indagara en el uso de los cuaterniones para estudio de transformaciones en el espacio ordinario de tres dimensiones, lo cual es el objetivo de este trabajo.

2.1. Los cuaterniones de Hamilton

En la historia de las matemáticas existe una concordancia en que los tres principales descubrimientos realizados durante el siglo XIX fueron:

1. El desarrollo de la geometría no Euclidiana.
2. El álgebra no conmutativa.
3. La teoría del cálculo.

Durante mucho tiempo se pensó que una restricción para la geometría era lo que se conocía como "*Geometría Euclidiana*". Fue sólo hasta que investigaciones desarrolladas de forma simultánea a la de la geometría Euclidiana, que se descubrieron formas diferentes para el estudio de la geometría.

El álgebra se utilizó como una generalización matemática, como una representación de objetos sometidos a estudio, utilizando las literales algebraicas como representación de dichos objetos, así mismo todas las reglas ordinarias de operaciones aritméticas eran válidas para el desarrollo del álgebra. Mediante el transcurso del tiempo los estudios se enfocaron en la búsqueda para el fundamento teórico de las propiedades algebraicas (asociativa, conmutativa y propiedades de agrupamiento) para el desarrollo de la matemática.

Con base en este contexto, fue entonces que Sir William Rowan Hamilton realizó la introducción del álgebra de cuaterniones. La comunidad matemática vio con sorpresa que el álgebra de cuaterniones violaba la ley de conmutatividad para la multiplicación, en ese momento la introducción de los cuaterniones de Hamilton les pareció imposible, el hecho de que hubiera otra forma consistente que no sostuviera las propiedades fundamentales del álgebra de números reales.

Números hiper-complejos

El objetivo que motivó a Hamilton era la generalización de forma análoga del espacio tridimensional, mediante el uso de números reales triples. Sin embargo, durante mucho tiempo luchó por obtener un sistema algebraico que pudiera dar una solución óptima para su investigación.

La historia dice que una mañana, mientras caminaba con su esposa a lo largo del Canal Royal, tuvo una chispa de iluminación. En un instante descubrió que un número triple no era suficiente y que en su lugar era necesario un número cuádruple. Él visualizó que

no sólo requería el componente complejo \mathbf{i} , sino que además, los componentes \mathbf{ij} y \mathbf{k} que satisficieran la siguiente relación:

$$i^2 = j^2 = k^2 = ijk = -1$$

Este descubrimiento causó tal impacto en Hamilton que supuestamente no pudo contener el impulso y se detuvo a grabar su descubrimiento en una piedra del puente Broome del canal de Dublin, de esa forma fue que el cuaternión llegó a la vida.

Hamilton tubo muchas dificultades para introducir el desarrollo que había realizado ante la comunidad matemática. Frobenius se dio cuenta que Hamilton tenía razón cuando trato de rehacer la teoría propuesta por Hamilton utilizando triadas de números lo cual no fue posible. Motivo por el que tuvo que regresar a usar los cuaterniones, algunos matemáticos llamaron al cuaternión como número complejo de rango cuatro. Posteriormente dentro de este trabajo se vera que los cuaterniones, de forma particular, satisfacen la función de operadores rotacionales en el espacio tridimensional [5].

El conjunto de cuaterniones, aunado a las características determinadas para la adición y multiplicación, matemáticamente establecen un sistema que de forma particular enfatizan en el hecho que, mediante la multiplicación entre cuaterniones no es conmutativa, también que existe un inverso multiplicativo para cada elemento diferente de cero en el conjunto.

A manera de resumen, el conjunto de los cuaterniones bajo la operación de adición y multiplicación satisfacen todas las propiedades del campo, lo cual se discutirá posteriormente, excepto por la ley de conmutatividad para la multiplicación, tomando estas propiedades para su uso en la robótica de manipuladores.

2.2. Robótica industrial

La finalidad de incluir esta sección es poner en contexto la robótica industrial, mediante una breve y concreta definición de la misma, antes de poder continuar con la explicación detallada a que se refiere la parametrización mediante cuaterniones en el cálculo de la cinemática de un robot industrial.

Así mismo, el interés de mostrar como un robot tiene movimientos específicos, para lo cual es necesario mostrar los tipos de articulaciones, sus posibles combinaciones con las cuales se pueden realizar movimientos con un brazo robótico.

2.2.1. Definición de robótica industrial

Para la robótica industrial existe una gran cantidad de definiciones, lo cual, hace difícil de esta tarea a la hora de establecer una definición formal de lo que es un robot industrial. La primera de estas nace de la diferencia sobre el concepto de lo que es un robot y lo que es un manipulador para el mercado japonés y el euro-americano. En tanto que, los japoneses definen un robot industrial como cualquier dispositivo mecánico dotado de articulaciones móviles destinado a la manipulación, el mercado occidental es más restrictivo, exigiendo una mayor complejidad, sobre todo en lo relativo al control. En segundo lugar, y centrándose ya en el concepto occidental, aunque existe una idea común acerca de lo qué es un robot industrial, no es fácil ponerse de acuerdo a la hora de establecer una definición formal. Además, la evolución de la robótica ha ido obligando, a quienes se dedican al estudio de la robótica, a diferentes actualizaciones de su definición.

La definición más comúnmente aceptada posiblemente sea la de la Asociación de Industrias Robóticas (RIA)[6], según la cual:

Definición 2.1. Un robot industrial (Figura 2.1) es un manipulador multifuncional reprogramable, capaz de mover materias, piezas, herramientas, o dispositivos especiales, según trayectorias variables, programadas para realizar tareas diversas.

Esta definición, ligeramente modificada, es la que ha sido adoptada por la Organización Internacional de Estándares (ISO) que define al robot industrial como:

Definición 2.2. 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. Otra definición es la establecida por la Asociación Francesa de Normalización (AFNOR) [7], que define primero el manipulador y, basándose en dicha definición, el robot:



FIGURA 2.1: Robot manipulador industrial

Definición 2.3. 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 un dispositivo lógico.

Definición 2.4. 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 ultimo, la Federación Internacional de Robótica (IFR) [8] distingue entre robot industrial de manipulación y otros robots:

Definición 2.5. 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.

De la Definición 2.5 debemos hacer notar que cuando nos referimos a reprogramabilidad y multifunción se consiguen sin modificaciones físicas del robot.

De las definiciones 2.1, 2.2, 2.3, 2.4, 2.5 podemos establecer que concuerdan con la aceptación de un robot industrial como un brazo mecánico con capacidad de manipulación y que incorpora un control de mayor o menor complejidad. Sin embargo, un sistema robotizado, es un concepto más amplio. Éste engloba todos aquellos dispositivos que realizan tareas de forma automatizada, lo cual conduce a una sustitución importante del ser humano y puede incorporar uno o varios tipos de robots, siendo esto último lo más frecuente.

2.2.2. Tipo de movimiento en las articulaciones de los robots

Una de las principales características de un robot industrial manipulador es la de contar con un brazo mecánico con la capacidad de llevar a cabo distintas tareas. En el extremo final del robot está situado el elemento manipulador (comúnmente llamado efector final), el cual puede estar constituido de diferentes formas acorde a las tareas que deba llevar a cabo. El manipulador tipo pinza es el que se utiliza con mayor frecuencia debido a que con él se pueden realizar tareas de sujeción de piezas las cuales pueden ser depositadas en lugares y posiciones específicas. Otros robots industriales incorporan herramientas distintas, por ejemplo; una punta de soldador o brazos compactos si han de transmitir un golpe, etc.

Los brazos de estos robots cuentan con varios eslabones que los unen entre sí mediante articulaciones que pueden ser de dos tipos:

- ⇒ **Articulación prismática (P)**: permite realizar sólo traslaciones lineales, de avance o retroceso.
- ⇒ **Articulación de revolución o giratoria (R)**: permite realizar sólo un movimiento de giro.

El brazo de un robot industrial puede incorporar en sus eslabones varios tipos de estas articulaciones, de tal manera que sus movimientos sean una combinación con mayor o menor complejidad entre traslación y giro.

Un robot industrial puede tener las siguientes estructuras fundamentales:

- ▷ **Estructura cartesiana:** de configuración PPP (ver Figura 2.2), está conformado por tres articulaciones prismáticas con ejes perpendiculares y de movimiento lineal que se realiza a lo largo de dichos ejes. Este tipo como su nombre lo indica utiliza coordenadas cartesianas (altura, anchura y profundidad) para situar un objeto en el espacio.

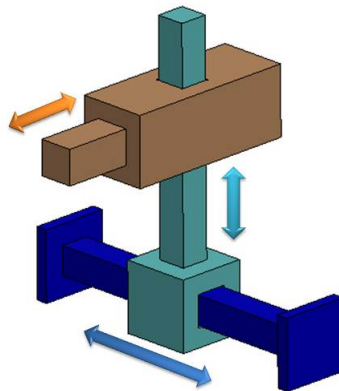


FIGURA 2.2: Robot con movimiento según coordenadas cartesianas.

- ▷ **Estructura cilíndrica:** de configuración RPP (ver Figura 2.3), consiste en dos ejes lineales que se deslizan ortogonalmente entre sí los cuales están montados sobre una base giratoria. Este tipo de robots utiliza coordenadas cilíndricas para situar un objeto en el espacio.

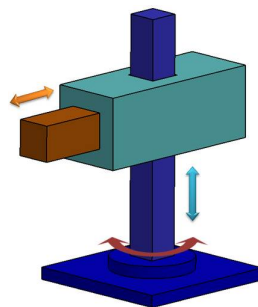


FIGURA 2.3: Robot con movimiento según coordenadas cilíndricas.

- ▷ **Estructura esférica:** de configuraciones RRP (ver Figura 2.4), consta de dos articulaciones giratorias y una prismática. El alcance de un punto en el espacio se logra mediante un brazo telescópico que se extiende y se retrae a lo largo de su eje, montado sobre una base giratoria con dos ejes rotacionales independientes. Este tipo de robots utiliza coordenadas polares para situar un objeto en el espacio.

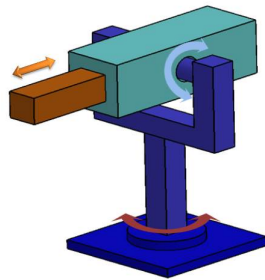


FIGURA 2.4: Robot con movimiento según coordenadas polares en una estructura esférica.

- ▷ **Estructura articulada:** de configuración RRR (ver Figura 2.5), este tipo de estructura también denominada antropomórfica es la que más se le asemeja al brazo humano. Utiliza coordenadas polares para situar un objeto en el espacio y realiza movimientos similares a los de un robot con estructura esférica mediante la combinación de tres giros.

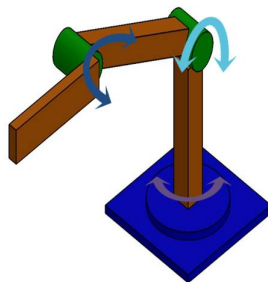


FIGURA 2.5: Robot con movimiento según coordenadas polares en una estructura articulada.

2.3. Cinemática de los robots

Un manipulador mecánico (Definición 2.2, 2.3, 2.4 y 2.5), puede tomar la forma de un cuerpo rígido conectado mediante una serie de articulaciones de revolución o prismáticas (ver Sección 2.2.2) que es movida mediante actuadores, el cual, puede ser modelado como una cadena de articulaciones que se encuentran en lazo abierto, es decir, el inicio de la cadena articulada se encuentra unida a la base mientras que el final de la cadena (el effector final) se encuentra libre, esto le permite al robot llevar a cabo manipulación de objetos o alguna otra tarea de montaje. La orientación y posición del brazo articulado

es consecuencia de los movimientos relativos de cada articulación. Gran parte de las aplicaciones de la robótica, se interesan en el estudio de la descripción espacial del efector final del manipulador con respecto a un sistema de coordenadas de referencia fija.

Cuando se habla de cinemática del brazo de un robot, se refiere al estudio analítico de la geometría del movimiento en las articulaciones de dicho robot con referencia a un sistema de coordenadas fijo como una función del tiempo, sin tomar en cuenta las fuerzas/torques que originan el movimiento. De tal manera que se enfoca en la descripción del desplazamiento espacial que efectúa el robot al transcurrir un determinado tiempo, en particular las relaciones entre las variables espaciales del tipo articulación, la posición y orientación del efector final del robot.

El problema de la cinemática origina dos situaciones esenciales: la primera conocida como "*Problema cinemático directo*", mientras la segunda se le conoce "*Problema cinemático inverso*". El problema cinemático inverso se utiliza con más frecuencia, que el problema cinemático directo, ya que con él las variables independientes en un robot son las variables de articulación y una tarea se suele dar en términos de las coordenadas de referencia. La Figura 2.6 muestra un diagrama a bloques que indica las relaciones entre estos dos problemas [9].

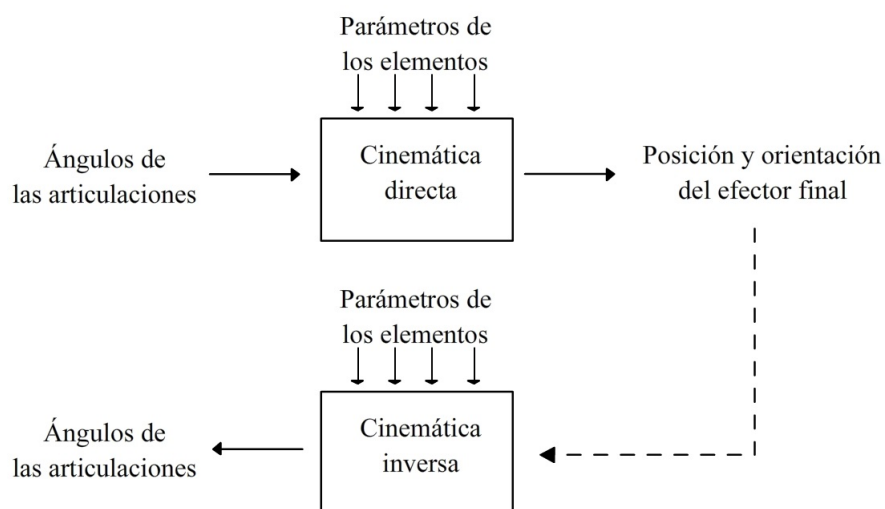


FIGURA 2.6: Relación entre la cinemática directa e inversa.

Los tipos de movimiento que puede realizar una articulación (rotación y traslación) con respecto a un sistema de coordenadas de referencia, la posición del efector final

es consecuencia del desplazamiento total de las traslaciones y las rotaciones angulares de todos los elementos. Denavit y Hartenberg [10] propusieron un método sistemático y generalizado de utilizar álgebra matricial para describir y representar la geometría espacial de los elementos de un brazo con respecto a un sistema de referencia fijo. Este método utiliza una matriz de transformación homogénea 4×4 para describir la relación espacial entre dos elementos mecánicos rígidos adyacentes y reduce el problema cinemático a encontrar una matriz de transformación homogénea 4×4 equivalente que relaciona el desplazamiento espacial del “*sistema de coordenadas*” del efector final al sistema de coordenadas de referencia. Estas matrices de transformación homogénea son también útiles para derivar las ecuaciones de movimiento dinámico de un brazo.

De forma general el problema cinemático inverso puede ser resuelto mediante diferentes técnicas, las que se utilizan con mayor frecuencia son; la matriz algebraica, métodos iterativos o geométricos.

A continuación dará de forma breve un planteamiento más general utilizando matrices homogéneas 4×4 para obtener una solución de la articulación para manipuladores simples.

2.3.1. Cinemática directa

Para desarrollar un método de forma general y sistemática, se ha utilizado, el álgebra de vectores y desarrollo matricial para describir y representar la localización de las articulaciones del brazo de un robot con respecto a un sistema de coordenadas fijo. Debido a que los elementos del brazo de un robot pueden girar y/o trasladarse respecto a un sistema de coordenadas de referencia, se establece un sistema de coordenadas ligado al cuerpo a lo largo del eje de la articulación para cada eslabón.

El problema cinemático directo se reduce a encontrar una matriz de transformación que describa una relación de un cuerpo ligado al sistema de coordenadas de referencia. Esto se logra mediante una matriz de rotación de 3×3 , la cual describe las operaciones rotacionales del sistema ligado al cuerpo con respecto al sistema de referencia. De esta manera, es necesario utilizar coordenadas homogéneas las cuales representen vectores de posición en un espacio tridimensional así las matrices de rotación se amplían a matrices

de transformación homogénea de 4×4 para incluir operaciones de traslación del sistema de coordenadas ligado al cuerpo. Esta representación matricial de un elemento mecánico rígido para describir la geometría espacial de un brazo fue utilizada por primera vez por Denavit y Hartenberg [9, 10].

a. Matrices de rotación

Las matrices de rotación (una matriz de 3×3), puede ser definida como una matriz de transformación que opera sobre un vector de posición en un espacio Euclidiano, de este modo, transforma sus coordenadas y las expresa en un sistema de coordenadas rotado como OUVW de un sistema de coordenadas de referencia OXYZ. La Figura 2.7 muestra ambos sistemas de coordenadas rectangulares, el primero muestra el sistema de coordenadas de referencia OXYZ (con OX, OY y OZ como los ejes de coordenadas), el segundo, el sistema de coordenadas OUVW (con OU, OV y OW como sus ejes de coordenadas), para ambos sistemas el punto de origen se encuentra en el punto O.

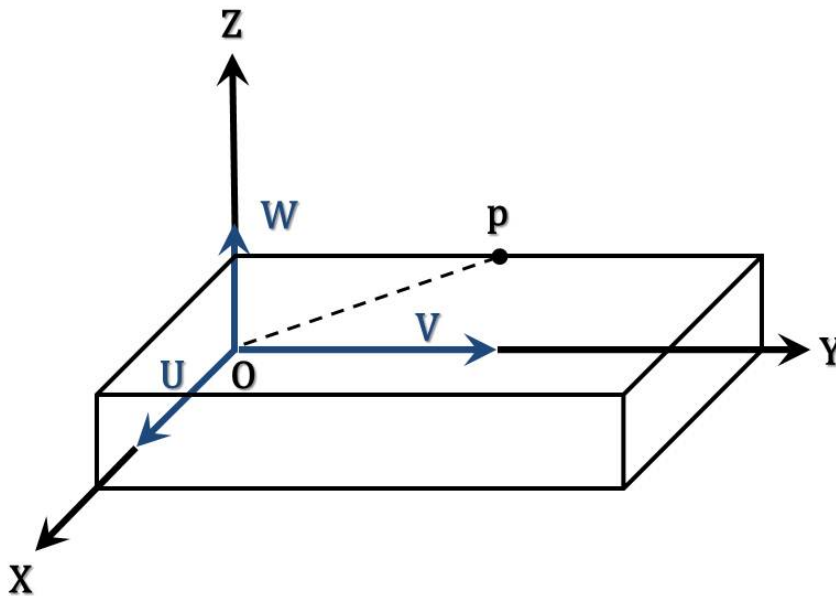


FIGURA 2.7: Representación de ambos sistemas de coordenadas ligados a un cuerpo.

Se considera como sistema de coordenadas fijo o sistema de referencia al sistema compuesto por OXYZ, el cual representa un espacio tridimensional. El sistema OXYZ sirve como referencia para que el sistema compuesto por OUVW gire en torno a él. El sistema OUVW se puede considerar físicamente como el sistema de

coordenadas ligado al cuerpo rígido, lo cual significa que se encuentra unido al cuerpo esto es conveniente ya que por ejemplo, en un robot, así como, en alguna nave espacial este sistema se mueve junto con él, siendo (i_x, j_y, k_z) los vectores unitarios a lo largo del sistema de coordenadas OXYZ y (i_u, j_v, k_w) para OUVW. Un punto p en el espacio puede ser representado en ambos sistemas de coordenadas. Para facilitar el análisis suponga que p está en reposo y fijo con respecto al sistema de coordenadas OUVW. Entonces el punto p se puede representar por sus coordenadas con respecto al sistema OUVW y OXYZ, respectivamente, como:

$$p_{uvw} = (p_u, p_v, p_w)^T \quad \text{y} \quad p_{xyz} = (p_x, p_y, p_z)^T \quad (2.1)$$

donde p_{xyz} y p_{uvw} son representaciones de un mismo punto pero con respecto a distintos sistemas de coordenadas, el superíndice T en una matriz así como en un vector hacen notar que se trata de una operación de transposición. Las coordenadas del punto P_{uvw} expresadas con respecto al sistema de coordenadas OXYZ, pueden ser transformadas mediante una matriz R de transformación 3×3 , después de que el sistema de coordenadas ha sido girado se tiene:

$$p_{xyz} = R p_{uvw} \quad (2.2)$$

físicamente, el punto P_{uvw} ha sido girado junto con el sistema de coordenadas OUVW. Recordando la definición de las componentes de un vector se tiene

$$p_{uvw} = p_u i_u + p_v j_v + p_w k_w \quad (2.3)$$

donde p_x ; p_y ; p_z son las representaciones de las componentes del vector p a lo largo de los ejes OX, OY y OZ, o lo que es lo mismo, la proyección del vector p sobre sus respectivos ejes. de este modo utilizando la definición del producto escalar (producto punto) y de acuerdo con la Ecuación 2.3, se obtiene la expresión en forma matricial,

$$\begin{bmatrix} p_x \\ p_y \\ p_z \end{bmatrix} = \begin{bmatrix} i_x \cdot i_u & i_x \cdot j_v & i_x \cdot k_w \\ j_y \cdot i_u & j_y \cdot j_v & j_y \cdot k_w \\ k_z \cdot i_u & k_z \cdot j_v & k_z \cdot k_w \end{bmatrix} \begin{bmatrix} p_u \\ p_v \\ p_w \end{bmatrix} \quad (2.4)$$

por lo tanto, al utilizar esta notación, la matriz \mathbf{R} en la Ecuación 2.2 está dada por

$$\mathbf{R} = \begin{bmatrix} \mathbf{i}_x \cdot \mathbf{i}_u & \mathbf{i}_x \cdot \mathbf{j}_v & \mathbf{i}_x \cdot \mathbf{k}_w \\ \mathbf{j}_y \cdot \mathbf{i}_u & \mathbf{j}_y \cdot \mathbf{j}_v & \mathbf{j}_y \cdot \mathbf{k}_w \\ \mathbf{k}_z \cdot \mathbf{i}_u & \mathbf{k}_z \cdot \mathbf{j}_v & \mathbf{k}_z \cdot \mathbf{k}_w \end{bmatrix} \quad (2.5)$$

por lo que de forma análoga se pueden obtener las coordenadas del punto p_{uvw} con las coordenadas de p_{xyz}

$$p_{uvw} = \mathbf{Q}p_{xyz} \quad (2.6)$$

esto corresponde a

$$\begin{bmatrix} p_u \\ p_v \\ p_w \end{bmatrix} = \begin{bmatrix} \mathbf{i}_u \cdot \mathbf{i}_x & \mathbf{i}_u \cdot \mathbf{j}_y & \mathbf{i}_u \cdot \mathbf{k}_z \\ \mathbf{j}_v \cdot \mathbf{i}_x & \mathbf{j}_v \cdot \mathbf{j}_y & \mathbf{j}_v \cdot \mathbf{k}_z \\ \mathbf{k}_w \cdot \mathbf{i}_x & \mathbf{k}_w \cdot \mathbf{j}_y & \mathbf{k}_w \cdot \mathbf{k}_z \end{bmatrix} \begin{bmatrix} p_x \\ p_y \\ p_z \end{bmatrix} \quad (2.7)$$

debido a que los productos escalares son conmutativos, se puede decir que la ecuación 2.5

$$\mathbf{Q} = \mathbf{R}^{-1} = \mathbf{R}^T \quad (2.8)$$

y la ecuación 2.7

$$\mathbf{Q}\mathbf{R} = \mathbf{R}^T\mathbf{R} = \mathbf{R}^{-1}\mathbf{R} = \mathbf{I}_3 \quad (2.9)$$

donde \mathbf{I}_3 es la matriz identidad 3×3 .

Las transformaciones en las ecuaciones 2.2 y 2.6, se le llama transformación ortogonal y debido a que en los productos escalares son todos vectores unitarios, es también ortonormal.

El primer interés en el desarrollo de la anterior matriz de transformación (Ecuación 2.11) consiste en encontrar las matrices de rotación las cuales representan rotaciones del sistema de coordenadas $OUVW$ respecto con cada uno de los diferentes ejes del sistema fijo (sistema de referencia) $OXYZ$. Si el sistema de coordenadas $OUVW$ tiene una rotación con un ángulo α sobre el eje OX , llegará a una nueva posición, por lo tanto, si se considera un punto p_{uvw} que tiene coordenadas $(p_u, p_v, p_w)^T$ con respecto al sistema de coordenadas $OUVW$, dicho punto tendrá diferentes coordenadas $(p_x, p_y, p_z)^T$ con respecto al sistema $OXYZ$. La matriz de transformación necesaria para realizar $R_{x,\alpha}$, se le conoce como matriz de rotación

respecto al eje OX con un ángulo α . Se puede deducir $R_{x,\alpha}$ del anterior concepto de matriz de transformación, es decir,

$$p_{xyz} = R_{x,\alpha} p_{uvw} \quad (2.10)$$

tomando como $i_x \equiv i_u$

$$R_{x,\alpha} = \begin{bmatrix} i_x \cdot i_u & i_x \cdot j_v & i_x \cdot k_w \\ j_y \cdot i_u & j_y \cdot j_v & j_y \cdot k_w \\ k_z \cdot i_u & k_z \cdot j_v & k_z \cdot k_w \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\alpha & -\text{sen}\alpha \\ 0 & \text{sen}\alpha & \cos\alpha \end{bmatrix} \quad (2.11)$$

“de manera análoga” se pueden obtener las matrices de transformación 3×3 con respecto a los ejes de rotación OY con ángulo ϕ y con respecto al eje de rotación OZ con ángulo de rotación θ , las cuales son:

$$R_{y,\phi} = \begin{bmatrix} \cos\phi & 0 & \text{sen}\phi \\ 0 & 1 & 0 \\ -\text{sen}\phi & 0 & \cos\phi \end{bmatrix} \quad R_{z,\theta} = \begin{bmatrix} \cos\alpha & -\text{sen}\alpha & 0 \\ \text{sen}\alpha & \cos\alpha & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (2.12)$$

a las matrices de rotación $R_{x,\alpha}$, $R_{y,\phi}$, $R_{z,\theta}$ se les llama “*matrices de rotación básicas*”. El uso de estas matrices se representa de forma gráfica en la Figura 2.8.

b. Matrices de rotación compuesta

Para representar una secuencia de rotaciones finitas respecto al sistema de coordenadas de referencia OXYZ, las matrices de rotación básicas se pueden multiplicar entre sí. Debido a que las multiplicaciones de matrices no son conmutativas, es necesario conservar el orden en el cual se realizan dichas rotaciones. Es decir si, se desea realizar una rotación respecto al eje OX con un ángulo α , seguida de una rotación con ángulo θ , con respecto al eje OZ, por ultimo una rotación con respecto al eje OY con un ángulo ϕ . La matriz que resulta de representar dicha

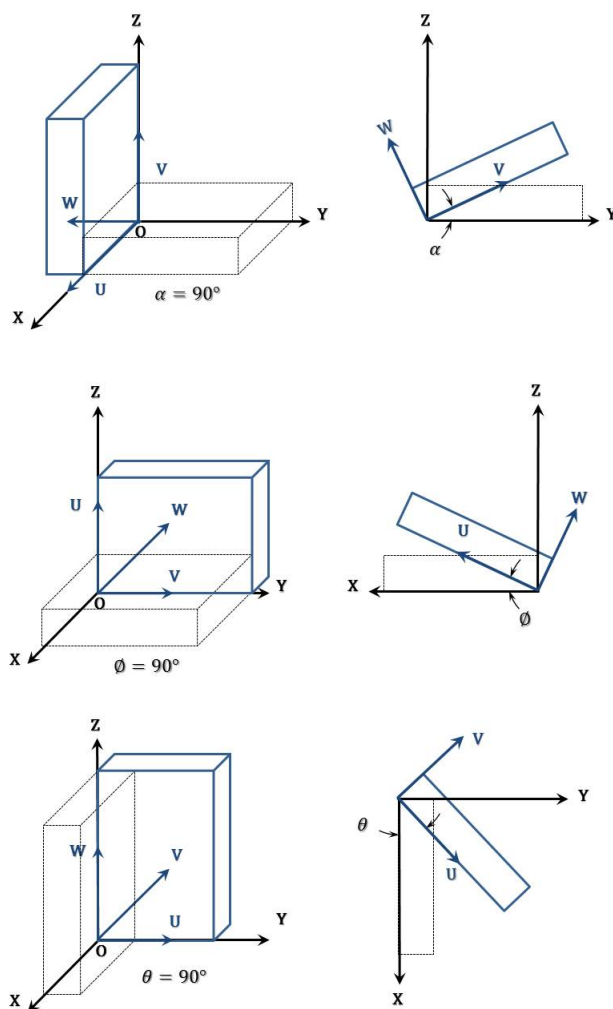


FIGURA 2.8: Representación gráfica de las matrices de rotación.

secuencia de rotaciones es la siguiente:

$$\begin{aligned}
 R &= R_{y,\phi} R_{z,\theta} R_{x,\alpha} = \begin{bmatrix} C\phi & 0 & S\phi \\ 0 & 1 & 0 \\ -S\phi & 0 & C\phi \end{bmatrix} \begin{bmatrix} C\theta & -S\theta & 0 \\ S\theta & C\theta & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & C\alpha & -S\alpha \\ 0 & S\alpha & C\alpha \end{bmatrix} \\
 &= \begin{bmatrix} C\phi C\theta & S\phi S\alpha - C\phi S\theta C\alpha & C\phi S\theta S\alpha + S\phi C\alpha \\ S\theta & C\theta C\alpha & -C\theta S\alpha \\ -S\phi C\theta & S\phi S\theta C\alpha + C\phi S\alpha & C\phi C\alpha - S\phi S\theta S\alpha \end{bmatrix}
 \end{aligned}$$

(2.13)

donde

$$\begin{aligned} C\phi &\equiv \cos\phi, & S\phi &\equiv \operatorname{sen}\phi, \\ C\theta &\equiv \cos\theta, & S\theta &\equiv \operatorname{sen}\theta, \\ C\alpha &\equiv \cos\alpha, & S\alpha &\equiv \operatorname{sen}\alpha, \end{aligned}$$

Además de rotar respecto de los ejes principales del sistema de referencia OXYZ, el sistema de coordenadas que esta rotando OUVW puede también rotar respecto de su propio eje principal. En este caso, la matriz de rotación resultante o compuesta se puede obtener de las siguientes reglas [9].

- 1.- Inicialmente ambos sistemas de coordenadas son coincidentes, de aquí que la matriz de rotación es una matriz identidad I_3 .
- 2.- Si el sistema de coordenadas giratorio OUVW está girando respecto de uno de los ejes principales del sistema OXYZ, entonces premultiplicar la matriz de rotación previa (resultante) por una matriz de rotación básica apropiada.
- 3.- Si el sistema de coordenadas rotante OUVW está girado respecto de su propio eje principal, entonces postmultiplicar la matriz de rotación previa (resultante) por una matriz de rotación básica apropiada.

c. Matriz de rotación con representación de ángulos de Euler

La forma matricial para la representación del movimiento de un cuerpo rígido simplifica muchas operaciones, pero son necesarios nueve elementos para describir completamente esta rotación. Esto conduce directamente a un conjunto completo de coordenadas generalizadas. Este conjunto de coordenadas puede describir la orientación de un cuerpo rígido en rotación a través de un sistema de coordenadas de referencia y puede ser representado mediante ϕ , θ y ψ que son los llamados ángulos de Euler. Aunque los ángulos de Euler describen la orientación de un cuerpo rígido con respecto a un sistema de coordenadas de referencia fijo, hay muchos tipos diferentes de ángulos de Euler. Las tres representaciones más utilizadas de los ángulos de Euler se muestran en la Tabla 2.1:

TABLA 2.1: Tres tipos de representación de ángulos de Euler [9]

	Primer sistema de ángulos de Euler	Segundo sistema de ángulos de Euler	Tercer sistema, elevación, desviación y giro
Secuencia de rotaciones	ϕ respecto del eje OZ θ respecto del eje OU ψ respecto del eje OW	ϕ respecto del eje OZ θ respecto del eje OV ψ respecto del eje OW	ψ respecto del eje OX θ respecto del eje OY ϕ respecto del eje OZ

Un movimiento giroscópico puede ser asociado respecto al primer sistema de los ángulos de Euler representados en la Tabla 2.1. La representación de dicho movimiento es comúnmente nombrado como ángulos *eulerianos*. La Figura 2.9 muestra la secuencia de rotaciones correspondientes:

- 1.- Rotación con respecto al eje OZ, con un ángulo ϕ , $R_{z,\phi}$
- 2.- Rotación con respecto al eje OU, con un ángulo θ , $R_{u,\theta}$
- 3.- Rotación con respecto al eje OW, con un ángulo ψ , $R_{w,\psi}$

Las matrices de rotación para los ángulos Eulerianos son:

$$\begin{aligned}
 R = R_{z,\phi} R_{u,\theta} R_{w,\psi} &= \begin{bmatrix} C\phi & -S\phi & 0 \\ S\phi & C\phi & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & C\theta & -S\theta \\ 0 & S\theta & C\theta \end{bmatrix} \begin{bmatrix} C\psi & -S\psi & 0 \\ S\psi & C\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \\
 &= \begin{bmatrix} C\phi C\psi - S\phi C\theta S\psi & -C\phi S\psi - S\phi C\theta C\psi & S\phi S\theta \\ S\phi C\psi + C\phi C\theta S\psi & -S\phi S\psi + C\phi C\theta C\psi & -C\phi S\theta \\ S\theta S\psi & S\theta C\psi & C\theta \end{bmatrix}
 \end{aligned} \tag{2.14}$$

Otra forma de representación de ángulos eulerianos de la matriz de rotación $R_{\phi,\theta,\psi}$, mostrada en la Ecuación 2.14, es con respecto al sistema de referencia el cual representa una rotación en los ejes principales de la siguiente forma: rotación con respecto al eje OZ con un ángulo ψ seguida de una rotación de un ángulo θ a través del eje OX y finalmente una rotación sobre el eje OZ con un ángulo ϕ .

La Figura 2.10, representa el segundo sistema de representaciones de ángulos de Euler (ϕ, θ, ψ) , el cual corresponde a las siguientes rotaciones:

- 1.- Rotación con respecto al eje OZ, con un ángulo ϕ , $R_{z,\phi}$

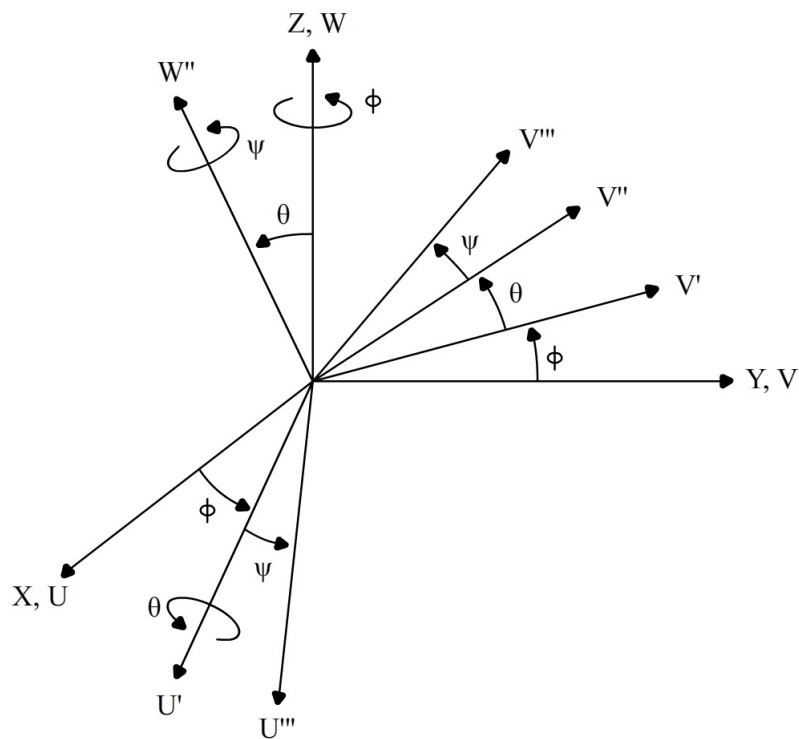


FIGURA 2.9: Primer sistema de ángulos *eulerianos* [9].

2.- Rotación con respecto al eje OV, con un ángulo θ , $R_{V,\theta}$

3.- Rotación con respecto al eje OW, con un ángulo ψ , $R_{W,\psi}$

La representación en forma matricial de estas rotaciones es:

$$\begin{aligned}
 R &= R_{z,\phi} R_{v,\theta} R_{w,\psi} = \begin{bmatrix} C\phi & -S\phi & 0 \\ S\phi & C\phi & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} C\theta & 0 & S\theta \\ 0 & 1 & 0 \\ -S\theta & 0 & C\theta \end{bmatrix} \begin{bmatrix} C\psi & -S\psi & 0 \\ S\psi & C\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \\
 &= \begin{bmatrix} C\phi C\psi - S\phi C\theta S\psi & -C\phi S\psi - S\phi C\theta C\psi & S\phi S\theta \\ S\phi C\psi + C\phi C\theta S\psi & -S\phi S\psi + C\phi C\theta C\psi & -C\phi S\theta \\ S\theta S\psi & S\theta C\psi & C\theta \end{bmatrix} \tag{2.15}
 \end{aligned}$$

La matriz de rotación (Ecuación 2.15) de ángulos de Euler ($R_{\phi,\theta,\psi}$), también se puede especificar con respecto a los ejes del sistema de coordenadas de referencia: una rotación con respecto al eje OZ con un ángulo ψ seguido de una rotación con ángulo θ sobre el eje OY y finalmente una rotación sobre el eje OZ con un ángulo

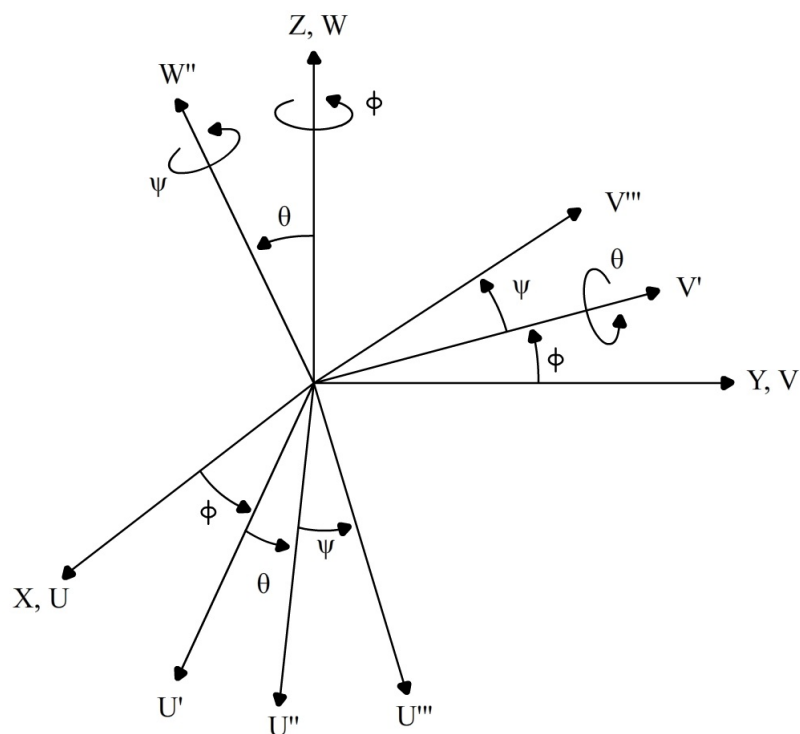


FIGURA 2.10: Segundo sistema de ángulos eulerianos [9].

ϕ .

El tercer conjunto de representaciones de ángulos de Euler mostrados en la Tabla 2.1, pertenecen a las rotaciones que llevan como nombre giro, elevación y desviación, conocidas por su nomenclatura en inglés como *Roll, Pitch, Yaw (RPY)*. Este conjunto es utilizado principalmente en la ingeniería aeronáutica para el análisis de vehículos espaciales y corresponde a la secuencia de las siguientes rotaciones:

- 1.- Rotación con respecto al eje OX , con un ángulo ψ , $R_{x,\psi}$ corresponde a la desviación
- 2.- Rotación con respecto al eje OY , con un ángulo θ , $R_{y,\theta}$ corresponde a la elevación
- 3.- Rotación con respecto al eje OZ , con un ángulo ϕ , $R_{z,\phi}$ corresponde al giro

La matriz resultante de la secuencia de rotaciones es:

$$\begin{aligned}
 R = R_{z,\phi} R_{y,\theta} R_{x,\psi} &= \begin{bmatrix} 1 & 0 & 0 \\ 0 & C\phi & -S\phi \\ 0 & S\phi & C\phi \end{bmatrix} \begin{bmatrix} C\theta & 0 & S\theta \\ 0 & 1 & 0 \\ -S\theta & 0 & C\theta \end{bmatrix} \begin{bmatrix} 1 & 0 & 1 \\ 0 & C\psi & -S\psi \\ 0 & S\psi & C\psi \end{bmatrix} \\
 &= \begin{bmatrix} C\phi C\theta & C\phi S\theta S\psi - S\phi C\psi & C\phi S\theta C\psi + S\phi S\psi \\ S\phi C\theta & S\phi S\theta S\psi + C\phi C\psi & S\phi S\theta C\psi - C\phi S\psi \\ -S\theta & C\theta S\psi & C\theta C\psi \end{bmatrix}
 \end{aligned} \tag{2.16}$$

La matriz de rotación $R_{\phi,\theta,\psi}$ para el giro, elevación y desviación se puede especificar en términos de la rotación respecto de los ejes principales del sistema de coordenadas de referencia y del sistema de coordenadas giratorio: una rotación de ángulo ϕ respecto del eje OZ seguida por una rotación del ángulo θ respecto del eje girado OV y finalmente una rotación del ángulo respecto del eje OU, estas rotaciones se representan en la Figura 2.11.

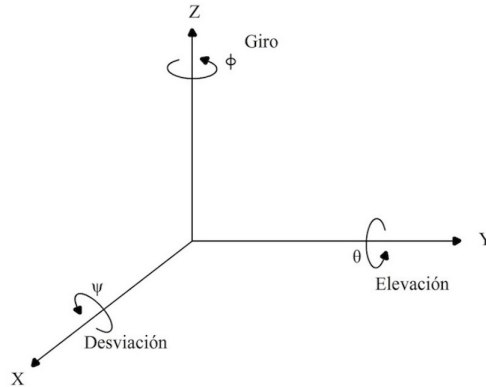


FIGURA 2.11: Tercer sistema de ángulos de Euler, giro, elevación y desviación [9].

Singularidad Gimbal Lock

Se conoce como “*Gimbal Lock*” a la pérdida de un grado de libertad. Esta se produce debido a que la matriz de rotación final depende del orden de las multiplicaciones, ya que eso lleva a la posibilidad que en ocasiones una de las rotaciones

lleve un eje a confundirse con el otro. Lo cual torna prácticamente imposible determinar el ángulo en el que un objeto fue girado sobre ese eje, esto conduce al fenómeno llamado “*Gimbal Lock*”.

Por ejemplo; Un objeto se hace girar en siguiente orden Z, Y y X, la rotación sobre el eje Y es de 90° . En este caso la rotación sobre el eje Z se hace correctamente puesto que es la primera. La rotación sobre Y se efectúa también correctamente. Sin embargo, después de esta rotación, el eje X girado se encuentra superpuesto al eje Z inicial. Así (ver Figura 2.12), cualquier rotación alrededor de este nuevo eje X gira el objeto sobre el eje Z inicial y se vuelve imposible determinar el ángulo de rotación del objeto alrededor del eje X. Tomando como marco de referencia el

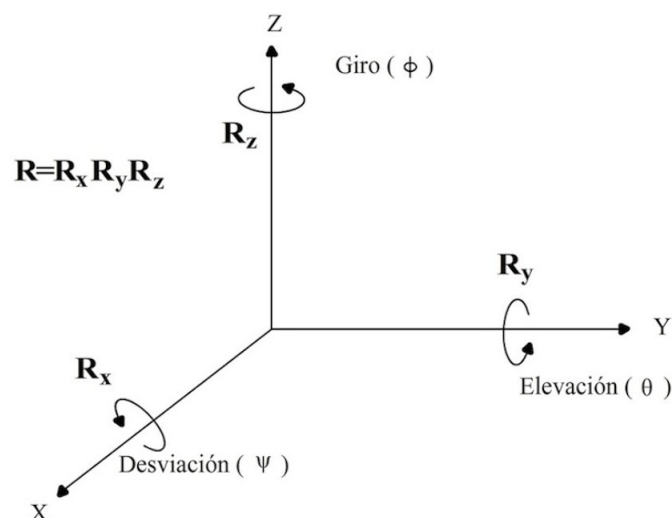


FIGURA 2.12: Ángulos de Euler (Gimbal Lock).

de la Figura 2.12, el problema de “Gimbal Lock” aparece cuando:

⇒ Si $R = R_x R_y R_z$ y si la rotación en $R_y \approx \frac{\pi}{2}$.

⇒ Si $R = R_x R_y R_z$ y $R_y \approx 0$ ó $R_y \approx \pi$ lo que lleva al eje X sobre el eje Z.

d. Coordenadas homogéneas y matriz de transformación

Como las matrices de 3×3 solo dan la posibilidad de representar la rotación de un cuerpo, para representar la traslación y el escalamiento de ese cuerpo, es necesario

introducir una cuarta coordenada o componente del vector, ésta representa la posición $p = (p_x; p_y; p_z)^T$, en un espacio tridimensional y lo transforma en $\hat{p} = (wp_x; wp_y; wp_z)^T$. Se dice que el vector de posición \hat{p} se encuentra expresado en *coordenadas homogéneas*.

El concepto de una representación en coordenadas homogéneas en un espacio Euclidiano tridimensional es útil para desarrollar transformaciones matriciales que incluyan rotación, traslación, escalado y transformación de perspectiva. En general, la representación de un vector de posición de N componentes por un vector de $(N + 1)$ componentes se llama representación en coordenadas homogéneas. La representación de un vector N -dimensional se efectúa en el espacio $(N + 1)$ -dimensional y el vector físico N -dimensional se obtiene dividiendo las coordenadas homogéneas por la coordenada $N + 1$ que es w . Así, en un espacio tridimensional, un vector de posición $p = (p_x; p_y; p_z)^T$ se representa por un vector ampliado $(wp_x; wp_y; wp_z)^T$. Las coordenadas físicas se relacionan a las coordenadas homogéneas como sigue:

$$p_x = \frac{wp_x}{w} \quad p_y = \frac{wp_y}{w} \quad p_z = \frac{wp_z}{w}$$

La representación de vector en coordenadas homogéneas para un espacio tridimensional no es único, ya que un vector de posición $p = (p_x, p_y, p_z)^T$, el cual se encuentra en coordenadas homogéneas, puede ser representado en como: $\hat{p} = (w_1 p_x, w_1 p_y, w_1 p_z, w_1)^T$ ó $\hat{p} = (w_2 p_x, w_2 p_y, w_2 p_z, w_2)^T$. De esta forma se puede ver que se ha introducido la cuarta componente de las coordenadas homogéneas w , esta se refiere a un factor de escala. Por lo tanto, si esta coordenada w es la unidad ($w = 1$), entonces las coordenadas físicas del vector son las mismas que las coordenadas homogéneas transformadas. Este factor de escala es muy útil para aplicaciones de informática gráfica y se le conoce como “*factor de escala universal*” y puede tomar cualquier valor positivo. En aplicaciones de robótica, $w = 1$, debido a que los eslabones no sufren ningún escalamiento.

Un vector de posición que se encuentra expresado en coordenadas homogéneas puede ser transformado de un sistema de coordenadas a otro, mediante la matriz de transformación homogénea, la cual es una matriz de 4×4 . Una matriz de transformación homogénea puede ser considerada como cuatro diferentes submatrices:

$$T = \left[\begin{array}{c|c} \mathbf{R}_{3 \times 3} & \mathbf{p}_{3 \times 1} \\ \hline - & - \\ \hline \mathbf{f}_{1 \times 3} & 1 \times 1 \end{array} \right] = \left[\begin{array}{c|c} \text{Matriz de} & \text{Vector de} \\ \text{rotación} & \text{posición} \\ \hline - & - \\ \hline \text{Transformación} & \text{Escala} \\ \text{de perspectiva} & \end{array} \right] \quad (2.17)$$

Cada una de las submatrices tiene una función específica. La submatriz superior izquierda es de 3×3 la cual representa una matriz de rotación, la submatriz superior derecha es de 3×1 representa el vector de posición con respecto al origen del sistema de coordenadas rotado con respecto al sistema de coordenadas de referencia, la submatriz que representa la perspectiva se encuentra en la parte inferior izquierda es de dimension 1×3 , el factor de escala se encuentra en la parte inferior derecha y es un escalar. En robótica el uso de la matriz de transformación homogénea es representar la relación geométrica que existe entre el sistema ligado al cuerpo OUVW y el sistema de coordenadas de referencia OXYZ.

Si se tiene un vector de posición p en un espacio tridimensional, el cual está expresado en coordenadas homogéneas (es decir, $\hat{p} = (p_x, p_y, p_z, 1)^T$), por lo tanto, utilizando el concepto de matriz de transformación, una matriz de rotación de 3×3 se puede ampliar a una matriz de rotación homogénea de 4×4 T_{rot} , para operaciones de rotación pura. de este modo las ecuaciones 2.11 y 2.12, se expresan como matrices de rotación homogénea de la siguiente forma:

$$T_{x,\alpha} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\alpha & -\text{sen}\alpha & 0 \\ 0 & \text{sen}\alpha & \cos\alpha & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad T_{y,\phi} = \begin{bmatrix} \cos\phi & 0 & \text{sen}\phi & 0 \\ 0 & 1 & 0 & 0 \\ -\text{sen}\phi & 0 & \cos\phi & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.18)$$

$$T_{z,\theta} = \begin{bmatrix} \cos\theta & -\text{sen}\theta & 0 & 0 \\ \text{sen}\theta & \cos\theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Las matrices de rotación de 4×4 expresadas en la Ecuación 2.18, se llaman *matrices de rotación homogéneas básicas*.

Como se menciono en la Ecuación 2.17 la submatriz que se encuentra en la parte

superior derecha 3×1 que se encuentra dentro de la matriz de transformación homogénea, representa el efecto de trasladar el sistema de coordenadas OUVW que tiene ejes paralelos al sistema de coordenadas de referencia OXYZ, con la diferencia que en el origen está en (dx, dy, dz) con respecto al sistema de coordenadas de referencia.

$$T_{\text{tras}} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.19)$$

Esta matriz de transformación 4×4 se le llama matriz de *traslación homogénea básica*.

La submatriz que se encuentra en la parte inferior izquierda de la matriz de transformación homogénea representa la transformación de perspectiva, la cual es útil para aplicaciones de visión por computadora y calibración de modelos de cámara.

Los elementos que se encuentran en la diagonal principal de la matriz de transformación homogénea producen escalado global y local. Los primeros tres elementos diagonales de esta matriz producen un alargamiento o escalado local, es decir:

$$\begin{bmatrix} a & 0 & 0 & 0 \\ 0 & b & 0 & 0 \\ 0 & 0 & c & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \begin{bmatrix} ax \\ by \\ cz \\ 1 \end{bmatrix} \quad (2.20)$$

los valores de la coordenadas se alargan mediante los escalares a , b y c , respectivamente.

Se puede observar que las matrices de rotación básica, T_{rot} , no producen ningún efecto de escalado global. El cuarto elemento de la diagonal tiene como propósito realizar un escalado global del sistema, de la siguiente forma:

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & s \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \begin{bmatrix} x \\ y \\ z \\ s \end{bmatrix} \quad (2.21)$$

donde $s > 0$. Las coordenadas cartesianas físicas del vector son:

$$p_x = \frac{x}{s}, \quad p_y = \frac{y}{s}, \quad p_z = \frac{z}{s}, \quad w = \frac{s}{s} \quad (2.22)$$

el cuarto elemento de la digonal de la matriz de transformación homogénea tiene el efecto de reducir globalmente las coordenadas si $s > 1$ y de alargar las coordenadas si $0 < s < 1$.

Resumiendo, una matriz de transformación homogénea 4×4 transforma un vector expresado en coordenadas de transformación homogénea con respecto al sistema OUVW en el sistema de coordenadas de referencia OXYZ, esto con $w = 1$.

$$\widehat{p}_{xyz} = T \widehat{p}_{xyz} \quad (2.23)$$

$$H = \begin{bmatrix} n_x & s_x & a_x & P_x \\ n_y & s_y & a_y & P_y \\ n_z & s_z & a_z & P_z \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} n & s & a & P \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.24)$$

Para obtener una matriz de transformación homogénea, las matrices de rotación y traslación pueden ser multiplicadas de forma simultanea obteniendo de este modo una matriz T .

$$T_n^0(p) = A_1^0(p_1)A_2^1(p_2) \dots A_n^{n-1}(p_n) \quad (2.25)$$

Debido a que la multiplicación de matrices no es conmutativa, es necesario tener un bien definido el orden en el que serán realizadas estas multiplicaciones. A continuación se presentan algunas reglas útiles para determinar la matriz de transformación homogénea compuesta [11]:

- a) Inicialmente ambos sistemas de coordenadas son coincidentes, ya que la matriz de transformación homogénea es una matriz identidad 4×4 , I_4 .
- b) Si el sistema de coordenadas rotante OUVW está rotando/trasladándose respecto de los ejes principales del sistema OXYZ, entonces *premultiplicar* la matriz de transformación homogénea previa (resultante) por una matriz de traslación/rotación básica apropiada.

- c) Si el sistema de coordenadas rotante $OUVW$ está rotando/trasladándose respecto de su propio eje principal, entonces *postmultiplicar* la matriz de transformación homogénea (resultante) por una matriz de rotación/traslación básica apropiada.

e. Eslabones, articulaciones y sus parámetros

De acuerdo con las definiciones dadas en la Sección 2.2.1 y la Sección 2.2.2, un manipulador mecánico consiste en una secuencia de cuerpos rígidos, llamados eslabones (elementos), conectados mediante articulaciones que pueden ser del tipo prismáticas o de revolución (ver Figura 2.13). Cada conjunto de articulación con eslabón constituyen un grado de libertad. Por lo tanto se establece que para un

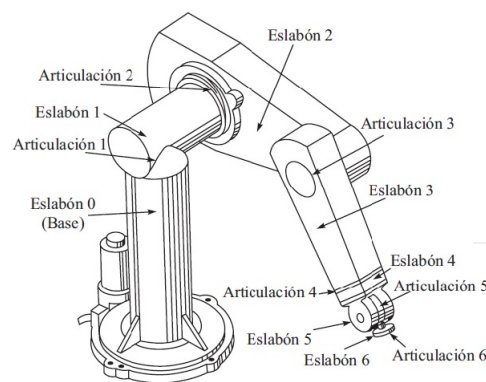


FIGURA 2.13: Articulaciones y elementos de un Robot PUMA [12].

manipulador con N grados de libertad existen N de articulación-eslabón con un enlace 0 (no considerado parte del robot), que se encuentra ligado a un soporte base en el que se establece un sistema de coordenadas inercial para el sistema dinámico, en el que el último elemento es el efector final que se encuentra unido a la herramienta (ver Figura 2.14). La forma en la que se encuentran numerados los elementos desde dentro hacia afuera comenzando en la base, teniendo como resultado, la articulación 1 como el punto de la conexión entre la base y el eslabón 1, cada uno de los eslabones se conecta máximo con otros dos, así no se forman lazos cerrados.

En general, dos eslabones se conectan mediante un tipo de articulación que tiene dos superficies deslizantes, una sobre la otra, mientras permanecen en contacto.

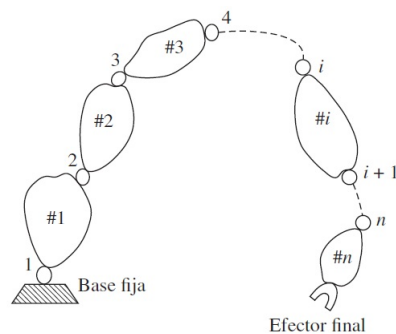


FIGURA 2.14: Articulaciones y elementos de un robot [12].

Para establecer el eje de una articulación se acoplan dos eslabones (para la articulación i), este eje tendrá dos normales conectadas a él las cuales corresponden a cada uno de los eslabones. La posición relativa de tales eslabones conectados (elemento $i-1$ y elemento i) viene dada por b_i , que es la distancia medida a lo largo del eje de la articulación entre las normales. El ángulo de articulación θ_i entre las normales se mide en un plano normal al eje de la articulación. De aquí que b_i y θ_i se puede llamar la distancia y el ángulo entre los eslabones adyacentes, respectivamente (ver Figura 2.15) y determinan la posición relativa de los eslabones vecinos.

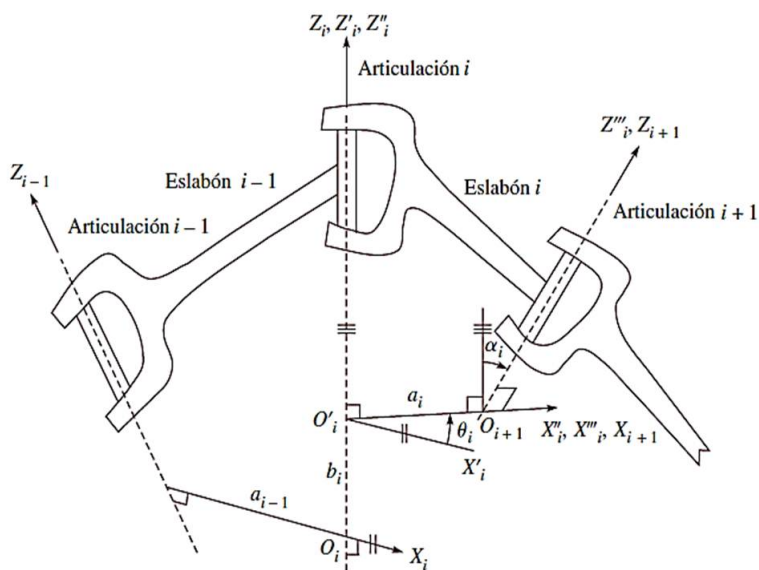


FIGURA 2.15: Sistema de coordenadas de eslabones y sus parámetros [9].

Un eslabón i ($i = 1, \dots, n$) se conecta con a otros dos eslabones $i-1$ e $i+1$; de este modo se establecen dos ejes de articulación en ambos extremos de la conexión. Un eslabón desde la perspectiva cinemática, es mantener una configuración fija entre sus articulaciones, que es caracterizada por dos parámetros: a_i y α_i . El parámetro a_i es la distancia más corta medida a lo largo de la normal común entre los ejes de la articulación, y α_i es el ángulo medido entre los ejes de articulación medidos en un plano perpendicular a α_i . Así a_i y α_i se pueden llamar *longitud* y *ángulo de torsión* del eslabón i respectivamente y determinan la estructura del eslabón i .

En general, se asocian cuatro parámetros a_i , α_i , b_i y θ_i a cada eslabón de un manipulador.

f. La representación Denavit-Hartenberg

Para describir la relación traslación y rotación entre eslabones adyacentes, Denavit y Hartenberg [10] propusieron un método matricial para establecer de forma sistemática un sistema de coordenadas (sistema ligado al cuerpo) para cada eslabón de una cadena articulada. La representación de Denavit-Hartenberg (**D-H**) resulta en una matriz de transformación homogénea 4×4 que representa cada uno de los sistemas de coordenadas de los eslabones en la articulación con respecto al sistema de coordenadas del eslabón previo. Así, mediante transformaciones secuenciales, las *coordenadas del eslabón final* se puede transformar y expresar en *coordenadas de la base* que constituyen el sistema inercial de este sistema dinámico.

Se puede establecer en los ejes de articulación de cada eslabón un sistema de coordenadas cartesiano ortonormal (x_i, y_i, z_i) , donde $i = 1, 2, \dots, n$ ($n =$ número de grados de libertad), más el sistema de coordenadas de la base. Como una articulación giratoria tiene solamente un grado de libertad, cada sistema de coordenadas (x_i, y_i, z_i) del brazo de un robot corresponde a la articulación $i+1$ y está fija en el eslabón i . Cuando el actuador de la articulación activa la articulación i , el eslabón i se moverá con respecto al eslabón $i-1$. Como el sistema de coordenadas i -ésimo está fijo en el eslabón i , se mueve junto con el eslabón i . Así pues, el sistema de coordenadas n -ésimo se mueve con el efector final (eslabón n). Las coordenadas de la base se definen como el sistema de coordenadas número 0

(x_0, y_0, z_0) , que también es el sistema de coordenadas inercial del brazo. Así, para un brazo como el PUMA de seis ejes (véase Figura 2.13), se tiene siete sistemas de coordenadas, representadas con $(x_0, y_0, z_0), (x_1, y_1, z_1), \dots, (x_6, y_6, z_6)$.

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

- 1.- El eje z_{i-1} yace a lo largo del eje de la articulación.
- 2.- El eje x_i es normal al eje z_{i-1} y apunta afuera de él.
- 3.- El eje y_i completa el sistema de coordenadas dextrógiro según se requiera.

Mediante estas reglas, uno es libre de escoger la localización del sistema de coordenadas 0 en cualquier parte de la base soporte, mientras que el eje z_0 esté a lo largo del eje de movimiento de la primera articulación. El último sistema de coordenadas (el n -ésimo) se puede colocar en cualquier parte de la mano, mientras que el eje x_n sea normal al eje z_{n-1} .

La representación D-H de un eslabón rígido depende de cuatro parámetros geométricos asociados a cada eslabón, que describen completamente cualquier articulación prismática o de revolución. Con referencia en la Figura 2.14, estos cuatro parámetros se definen como [1]:

θ_i es el ángulo de la articulación del eje x_{i-1} al eje x_i respecto del eje z_{i-1} (utilizando la regla de la mano derecha).

b_i es la distancia desde el origen del sistema de coordenadas $(i-1)$ -ésimo hasta la intersección del eje z_{i-1} con el eje x_i a lo largo del eje z_{i-1} .

a_i es la distancia de separación desde la intersección del eje z_{i-1} con el eje x_i hasta el origen del sistema i -ésimo a lo largo del eje x_i (o la distancia más corta entre los ejes z_{i-1} y z_i).

α_i es el ángulo de separación del eje z_{i-1} al eje z_i respecto del eje x_i (utilizando la regla de la mano derecha).

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

y permanecen constantes para un robot, mientras que b_i es la variable de la articulación.

Según la representación de D-H, escogiendo adecuadamente los sistemas de coordenadas asociados a cada eslabón, será posible pasar de uno al siguiente mediante 4 transformaciones básicas que dependen exclusivamente de las características geométricas del eslabón. Estas transformaciones básicas consisten en una sucesión de rotaciones y traslaciones que permiten relacionar el sistema de referencia del elemento i con el sistema del elemento $i-1$. Las transformaciones en cuestión son las siguientes:

- a) Rotación alrededor del eje z_{i-1} un ángulo θ
- b) Traslación a lo largo de z_{i-1} una distancia b_i a través del vector b_i $(0,0,b_i)$
- c) Traslación a lo largo de x_i una distancia a_i a través del vector a_i $(0,0,a_i)$
- d) Rotación alrededor del eje x_i un ángulo α_i

La transformación del sistema de coordenadas resultante es resultado de la multiplicación de las matrices correspondientes. De tal modo se obtiene la matriz de transformación A_i^{i-1} para cada eslabón, de la siguiente forma:

$$\begin{aligned}
 A_i^{i-1} &= \begin{bmatrix} C\theta & -S\theta & 0 & 0 \\ S\theta & C\theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & b_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & a_i \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & C\alpha & -S\alpha & 0 \\ 0 & S\alpha & C\alpha & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\
 &= \begin{bmatrix} C\theta_i & -C\alpha_i S\theta_i & S\alpha_i S\theta_i & a_i C\theta_i \\ S\theta_i & C\alpha_i C\theta_i & -S\alpha_i C\theta_i & a_i S\theta_i \\ 0 & S\alpha & C\theta & b_i \\ 0 & 0 & 0 & 1 \end{bmatrix}
 \end{aligned} \tag{2.26}$$

donde θ_i , a_i , b_i , α_i son los parámetros D-H del eslabón i . Basta con identificar dichos parámetros para obtener las matrices A y relacionar así todos y cada uno de los eslabones del robot.

Algoritmo Denavit-Hartenberg para la obtención de un modelo cinemático directo

Según [1] el algoritmo establecido por Denavit-Hartenberg, tiene el siguiente orden:

- D-H 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.
- D-H 2. Numerar cada articulación comenzando por 1 (la correspondiente al primer grado de libertad) y acabando en n .
- D-H 3. Localizar el eje de cada articulación, si ésta 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.
- D-H 4. Para i de 0 a $n - 1$ situar el eje z_i sobre el eje de la articulación $i + 1$.
- D-H 5. Situar el origen del sistema de la base $\{S_0\}$ 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 .
- D-H 6. Para i de 1 a $n - 1$, situar el sistema $\{S_i\}$ (solidario al eslabón i) en la intersección del eje z_i con la línea normal común a $z_{(i-1)}$ y z_i . Si ambos ejes se cortasen se situaría $\{S_i\}$ en el punto de corte. Si fuesen paralelos $\{S_i\}$ se situaría en la articulación $i + 1$.
- D-H 7. Situar x_i en la línea normal común a $z_{(i-1)}$ y z_i .
- D-H 8. Situar y_i de modo que forme un sistema dextrógiro con x_i y z_i .
- D-H 9. Situar el sistema $\{S_n\}$ en el extremo del robot 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 .
- D-H 10. Obtener θ_i como el ángulo que hay que girar en torno a $z_{(n-1)}$ para que $x_{(i-1)}$ y x_i queden paralelos.
- D-H 11. Obtener d_i como la distancia, medida a lo largo de $z_{(n-1)}$, que habría que desplazar $\{S_{(i-1)}\}$ para que x_i y $x_{(i-1)}$ quedasen alineados.
- D-H 12. 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 $\{S_i\}$.
- D-H 13. Obtener α_i como el ángulo que habría que girar en torno a x_i (que ahora coincidiría con $x_{(i-1)}$), para que el nuevo $\{S_{(i-1)}\}$ coincidiese totalmente con $\{S_i\}$.

- D-H 14. Obtener las matrices de transformación A_i^{i-1} definidas en la Ecuación ??.
- D-H 15. Obtener la matriz de transformación que relaciona el sistema de la base con el sistema del extremo del robot $T = A_1^0, A_2^1, \dots, A_n^{n-1}$.
- D-H 16. La matriz T define la orientación (submatriz de rotación) y posición (submatriz de traslación) del extremo referido a la base en función de las n coordenadas articulares.

este es un ejemplo del uso del algoritmo D-H para un robot de 6 GDL. la Tabla 2.2

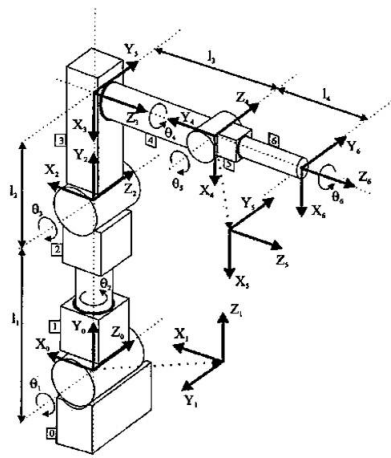


FIGURA 2.16: Robot IRB6400C [1].

representa los parámetros Denavit-Hartenberg para un robot IRB6400C de la empresa ABB.

TABLA 2.2: Representación D-H del robot IRB6400C [1]

Articulación	θ_i	b_i	a_i	α_i
1	θ_1	0	0	-90
2	θ_2	l_1	0	90
3	$\theta_3 - 90$	0	l_2	90
4	θ_4	$-l_3$	0	-90
5	θ_5	0	0	90
6	θ_6	l_4	0	0

2.3.2. Cinemática inversa

El problema cinemático inverso es usado con mayor frecuencia [12], sin embargo, así como para la cinemática directa (Sección 2.3.1) existe un método sistemático para la obtención de las ecuaciones que representan la configuración de las articulaciones del robot, en el caso de la cinemática inversa no existe un método con el cual se garantice la obtención de esas ecuaciones. Existen algunos procedimientos con los cuales se puede dar un acercamiento para la obtención de las ecuaciones, aunque estos dependen fuertemente de la configuración del robot.

Para el caso de la cinemática inversa existen algunas razones por las cuales este problema es más complejo, algunas de estas razones son [11, 12]:

- Generalmente las ecuaciones a resolver son no lineales y no siempre es posible encontrar una solución cerrada
- Existen múltiples soluciones
- Existen infinitas soluciones, por ejemplo en la redundancia cinemática del manipulador
- La solución podría no ser admisible debido a la estructura del manipulador

Se garantiza una solución solo si la posición y orientación del efector final dados pertenecen al espacio de trabajo derecho. Por otra parte el problema de múltiples soluciones no solo depende del número de grados de libertad, sino también en el número de parámetros DH no nulos, en general, entre mayor sea el número de parámetros no nulos, mayor será el número de soluciones admisibles.

Por ejemplo, para un manipulador de seis grados de libertad sin restricciones mecánicas en los límites articulación, en general, existen más de 16 soluciones admisibles, lo cual demanda un mayor criterio al momento de elegir entre las soluciones admisibles (por ejemplo, configuración de codo elevado y configuración de codo abajo ver Figura 2.16). Cuando existen restricciones mecánicas de límite en las articulaciones, el número de soluciones múltiples admisibles se reduce debido a la estructura real.

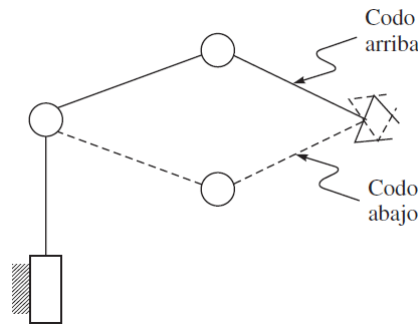


FIGURA 2.17: Soluciones Admisibles de configuración [12].

El cálculo de soluciones cerradas, requieren de la intuición algebraica para encontrar el significado de las ecuaciones que contienen las incógnitas o de la intuición geométrica para encontrar el significado de los puntos en la estructura para decidir con base en cual método es conveniente expresar la función de la orientación y posición en un numero reducido de incógnitas. Por otra parte cuando no existe una solución cerrada para el problema cinemático inverso, podría ser conveniente recurrir a técnicas de solución numéricas, ya que estas técnicas tienen la ventaja de ser aplicables a cualquier estructura cinemática, aunque en general estas no permiten el cálculo de todas las soluciones admisibles. Debido a que es difícil encontrar una solución general para un manipulador con estructura arbitraria, las soluciones de cinemática inversa para la posición se presentan respecto a estructuras de robots específicas, como se explicará a continuación.

Generalidades de la cinemática inversa

Para dar inicio a la solución de la cinemática inversa se puede establecer lo siguiente; dada una matriz de transformación homogénea 4×4

$$H = \begin{bmatrix} \mathbf{R} & \mathbf{O} \\ 0 & 1 \end{bmatrix} \quad (2.27)$$

para encontrar todas la soluciones de la ecuación,

$$T_n^0(p_1, \dots, p_n) = H \quad (2.28)$$

donde

$$T_n^0(p_1, \dots, p_n) = A_1 p_1 \dots A_n q_n \quad (2.29)$$

donde, H representa la posición y orientación deseada del efector final y la pregunta es, para encontrar las ecuaciones de las articulaciones p_1, \dots, p_n , tal que $T_n^0(p_1, \dots, p_n) = H$.

De tal manera la Ecuación 2.28 resulta en doce ecuaciones no lineales, con n incógnitas, las cuales pueden ser escritas de la siguiente forma;

$$T_{ij}(p_1, \dots, p_n) = h_{ij}, \quad i = 1, 2, 3, \quad j = 1, \dots, 4 \quad (2.30)$$

donde T_{ij} , h_{ij} representan las doce ecuaciones no lineales, esto debido a que la última fila de T_n^0 y H es $(0, 0, 0, 1)$, es decir, cuatro de las ecuaciones que representa Ecuación 2.28 son triviales.

Desacoplo cinemático

Aunque el problema cinemático inverso es bastante difícil, resulta que para manipuladores que cuentan con seis articulaciones, con las últimas tres en un punto de intersección, es posible realizar un desacoplo de la cinemática inversa, lo cual convierte el problema, en dos simples problemas conocidos como; *la cinemática inversa de posición y la cinemática inversa de orientación*.

De este modo, para un manipulador con seis grados de libertad con efector final (muñeca) con articulación esférica, la solución al problema cinemático inverso, puede ser desacoplado (separado), en dos problemas más simples, los cuales, en primera instancia busca encontrar la posición de la intersección con el eje de la muñeca, llamado centro de la muñeca, y posteriormente encontrar la orientación de la muñeca.

De forma concreta se permute suponer que existen exactamente seis grados de libertad y que los últimos tres ejes se unen en la intersección de la muñeca. Por lo tanto la Ecuación 2.28 representa dos conjuntos de ecuaciones, las de rotación y las de orientación.

$$R_6^0(p_1, \dots, p_6) = R \quad (2.31)$$

$$O_6^0(p_1, \dots, p_6) = O \quad (2.32)$$

donde O y R son la posición y la orientación deseada del efector final, expresados con respecto al sistema de coordenadas de referencia. Por lo tanto, dados O y R , y el problema de la cinemática inversa es resolver las ecuaciones para q_1, \dots, q_6 . El asumir

que la articulación de la muñeca es esférica, significa que los ejes z_3 , z_4 y z_5 tienen intersección en O_c , y por lo tanto, el origen para O_4 y O_5 asignados por la convención DH siempre podrán estar en el centro de la muñeca O_c . En algunas ocasiones el centro de O_3 puede también estar en O_c , aunque este no necesariamente tenga un desarrollo posterior. El punto importante de esta suposición para la cinemática inversa es que el movimiento de los tres últimos eslabones sobre estos ejes no va a cambiar la posición de O_c , y por lo tanto, la posición del centro de la muñeca esta en función de sólo los tres primeros eslabones.

El origen del efector final (cuyas coordenadas deseadas están dadas por O) se puede obtener mediante la translación de una distancia b_6 a lo largo de z_5 a partir de O_c . Los manipuladores que tienen el eje en el centro de la muñeca generalmente tienen coincidencia entre el eje z_5 y z_6 , por lo tanto, la tercera columna de \mathbf{R} representa la dirección de z_6 con respecto al sistema de referencia de la base. Entonces, tenemos

$$O = O_c^0 + b_6 \mathbf{R} \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \quad (2.33)$$

De tal manera, la intención de tener el efector final del robot en el punto con coordenadas dadas por O y con la orientación del efector final dada por $\mathbf{R} = (r_{ij})$, es necesario y suficiente que el centro de la muñeca O_c tenga coordenadas dadas por

$$O_c^0 = O - b_6 \mathbf{R} \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \quad (2.34)$$

y la orientación de la estructura correspondiente a $O_6 x_6 y_6 z_6$ con respecto a la base esta dada por \mathbf{R} . Si los componentes de posición del efector están denotados por O_x O_y O_z , y los componentes del centro de la muñeca O_c se encuentran denotados por x_c y_c z_c , por lo tanto las relaciones están dadas por

$$\begin{bmatrix} x_c \\ y_c \\ z_c \end{bmatrix} = \begin{bmatrix} O_x - b_6 r_{13} \\ O_y - b_6 r_{23} \\ O_z - b_6 r_{33} \end{bmatrix} \quad (2.35)$$

Usando la Ecuación 2.35 se encuentran las variables de las primeras tres articulaciones. Esto determina la transformación de la orientación \mathbf{R}_3^0 ya que esta solo depende de las tres primeras articulaciones. Con esto la orientación del efector final se determina relativamente de la estructura $O_3 \ x_3 \ y_3 \ z_3$ de la expresión

$$\mathbf{R} = \mathbf{R}_3^0 \mathbf{R}_6^3 \quad (2.36)$$

como

$$\mathbf{R}_6^3 = (\mathbf{R}_3^0)^{-1} \mathbf{R} = (\mathbf{R}_3^0)^T \mathbf{R} \quad (2.37)$$

Por lo tanto, los tres ángulos de las articulaciones finales entonces se puede encontrar como un conjunto de ángulos de Euler correspondientes a \mathbf{R}_6^3 , Teniendo en cuenta que el lado derecho de la Ecuación 2.37 se conocen por completo, dado que \mathbf{R} es dada y \mathbf{R}_3^0 se puede calcular una vez que se conocen las tres primeras variables conjuntas. La idea del desacoplamiento cinemático se ilustra en la Figura 2.18.

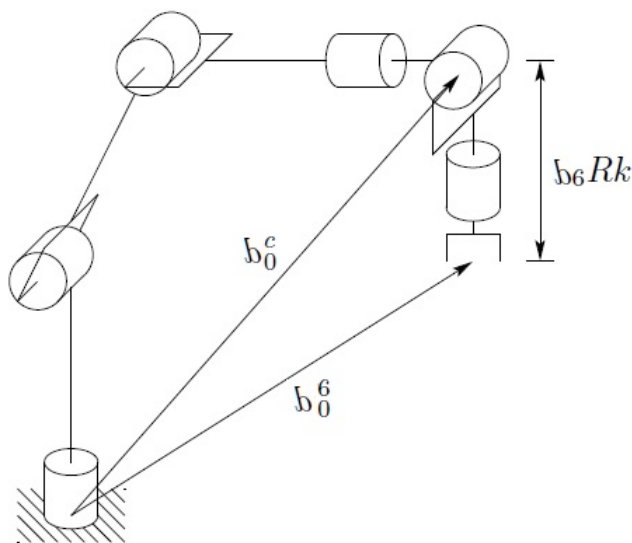


FIGURA 2.18: Desacoplo Cinemático [13].

Solución Algebraica

La determinación de la cinemática inversa utilizando el método de solución algebraica, consiste en generar ecuaciones como el nombre del método lo dice que puedan ser solucionadas de forma algebraica, generando sistemas de ecuaciones que puedan determinar las posiciones y orientaciones del manipulador.

Para el caso de la solución algebraica, se considera el brazo de la Figura 2.19, en el cual lo que se busca es encontrar los ángulos de las articulaciones (θ_1 , θ_2 , y θ_3) las cuales corresponden a la orientación y posición determinadas para el efector final.

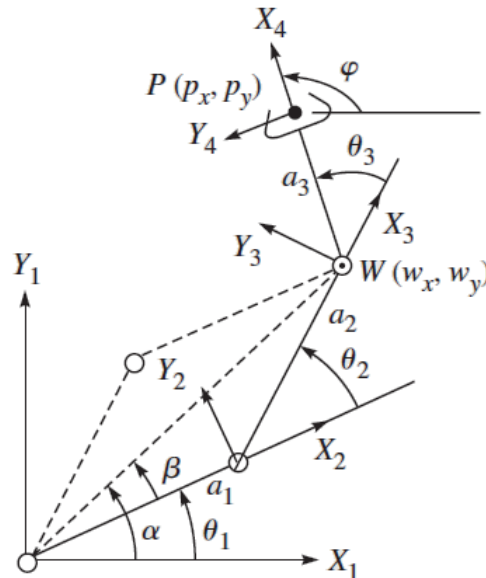


FIGURA 2.19: Brazo planar de 3 articulaciones [12].

como se observar la Figura 2.19 se pueden obtener estas ecuaciones que definen los puntos de las articulaciones

$$\phi = \theta_1 + \theta_2 + \theta_3 \quad (2.38)$$

$$p_x = a_1 c_1 + a_2 c_{12} + a_3 c_{123} \quad (2.39)$$

$$p_y = a_1 s_1 + a_2 s_{12} + a_3 s_{123} \quad (2.40)$$

como se puede ver en las ecuaciones anteriores, son de tipo no lineal en los ángulos de las articulaciones θ_1 , θ_2 y θ_3 . Además, el problema de la cinemática inversa se convierte en problema menos complejo a través de la subdivisión de tarea, es decir, el efector final se orienta una vez que se ha posicional la muñeca W mostrado en la Figura 2.19. Las coordenadas que se tiene en W son w_x y w_y . por lo tanto las ecuaciones 2.38, 2.39 y

2.40 se pueden expresar de la siguiente forma

$$\phi = \theta_1 + \theta_2 + \theta_3 \quad (2.41)$$

$$p_x = a_1 c_1 + a_2 c_{12} + a_3 c_{123} \quad (2.42)$$

$$p_y = a_1 s_1 + a_2 s_{12} + a_3 s_{123} \quad (2.43)$$

2.4. Los cuaterniones en la robótica

Esta sección tiene el objetivo de mostrar de forma breve el uso del álgebra de cuaterniones y el impacto que su uso ha tenido para el modelado de aplicaciones en la robótica. El álgebra de cuaterniones, como se había mencionado en el Capítulo ??, fue desarrollada por Sir William Hamilton en el año de 1843, sin embargo no fue sino hasta la década de los 60's del siglo XX cuando se empezaron a aplicar en la dinámica de cuerpos rígidos. Es importante mencionar que tal aplicación no se realizaba de una manera sistemática, puesto que el álgebra de cuaterniones utilizada era una extensión formal no rigurosa del álgebra de los números complejos [14].

Actualmente el estudio del álgebra de los cuaterniones ha trascendido, sin embargo, el modelado tridimensional de rotaciones y traslaciones de la cinemática de robot es comúnmente realizado usando matrices de transformación homogénea o ángulos de Euler. Por otra parte, el uso de cuaterniones como operadores espaciales, comparado con la matriz de transformación homogénea, muestra más eficacia en el término computacional y la economía de almacenaje [15].

La parametrización de la orientación usando ángulos de Euler requiere de más operaciones aritméticas, comparada con el uso de cuaterniones (Tabla 2.3).

TABLA 2.3: Complejidad computacional: Cuaterniones vs Ángulos de Euler [16]

	Cuaterniones	Ángulos de Euler
Representación de la rotación	4 Elementos	9 elementos (16 elementos si las coordenadas homogéneas son usadas)
Composición de la rotación	16 multiplicaciones y dos adiciones	27 multiplicaciones y 18 adiciones

Además de la orientación de cuerpos rígidos, muchas identidades útiles que pertenecen a multiplicaciones de cuaterniones, son generalizadas. El espacio del cuaternión incluye el espacio vectorial de 3D, las cantidades físicas relacionadas con rotaciones, así como el desplazamiento angular, la velocidad, aceleración y el momento. Sus expresiones son derivadas en el espacio del cuaternión [17].

El álgebra de cuaterniones se basa en el uso del cuaternión unitario, el cual, es una representación de cuatro parámetros con una constante. Por lo tanto, es la dimensionalidad más baja posible para una representación a escala mundial no singular de la orientación [17, 18].

Se puede sistematizar el uso de los cuaterniones en la parametrización de las ecuaciones de equilibrio cinemático y dinámico de los movimientos de los cuerpos rígidos a través de funciones llamadas hipercomplejas [19]. Las propiedades del cuaternión se desarrollarán de forma más completa en los siguientes capítulos.

La conclusión dibujada y a la que se pretende llegar en este trabajo es demostrar que la representación de la cinemática con cuaterniones es más eficiente, más compacta y más elegante que la matriz de transformación homogénea. En el Capítulo 2 se dará una amplia explicación de las características más importantes para el uso de cuaterniones y en el Capítulo 3 se desarrollará el modelo matemático de las rotacionales que se puedan emplear para modelar la cinemática de robots.

Capítulo 3

Fundamentos matemáticos de los cuaterniones

En este capítulo se desarrollara el algebra de cuaterniones con el fin de establecer las bases que se requieren para la obtención de modelo matemático que describa la cinemática del robot IRB 1600.

3.1. Historia del álgebra de cuaterniones

El objetivo de Hamilton era generalizar los números complejos de algún modo que pudieran ser aplicados al espacio tridimensional. para lo cual extendió esta notación a los cuaterniones y escribió los resultados en los libros “ Elements of Quaternions” [20] y “Lectures on quaternions” [21]. En contexto con los cuaterniones es posible pensar en los números reales como una parte esencial de los números hiper-complejos de rango 1 tomando en cuenta que los números reales satisfacen las propiedades de campo bajo la suma o producto ordinario, aunado a esto, también se puede decir que los número complejos ordinarios forman parte de un número hiper-complejo de rango 2, tomando el número real como un subconjuto del numero complejo del cual la parte imaginaria es cero, el número complejo también satisface las propiedades de campo.

Cualquier conjunto de números hiper-complejos con un rango mayor de 2 no satisface las propiedades de campo. Fue este hecho el que preocupó y constituyó un obstáculo

a aquellos matemáticos que buscaban rangos más altos de extensiones sugeridas por la aceptación gradual de los números complejos en \mathbb{R}^2 .

Debido a que los números complejos constan de dos partes, las cuales una es simplemente un número real ordinario y otra es la parte imaginaria. Hamilton concluyó que necesitaba un complemento para la parte imaginaria, así llegó al desarrollo de un sistema que consta de “tres partes imaginarias” para posterior meta llamarla “álgebra de cuaterniones”.

Hamilton introdujo el concepto de número complejo como un álgebra formal de pares de números reales con operaciones de adición y producto definidos como en la actualidad. de forma siguiente:

$$\begin{aligned}(x_1, y_1) + (x_2, y_2) &= (x_1 + x_2, y_1 + y_2) \\ \lambda(x_1, y_1) &= (\lambda x_1, \lambda y_1) \\ (x_1, y_1)(x_2, y_2) &= (x_1 x_2 - y_1 y_2, x_1 y_2 + y_1 x_2)\end{aligned}$$

Por analogía con los números complejos, Hamilton se preguntó si sería posible hacer lo mismo para *tripletas* de números lo que había hecho para *dobletes* o pares de números reales [22], así de esta forma, concibió que para describir una rotación seguida de un escalamiento es necesario de cuatro parámetros. Un número describe el escalamiento, otro el número de grados a ser rotado, mientras que los últimos dos el plano en el cual el vector deberá girar. Hamilton descubrió un producto cerrado de números hiper-complejos de rango 4 de la forma:

$$ix + jy + kz, \quad \text{donde,} \quad i^2 = j^2 = k^2 = ijk = -1$$

a este número complejo de \mathbb{R}^4 lo nombro *cuaternión*.

Finalmente, aún cuando los números hiper-complejos de rango cuatro pueden ser definidos a través de n , pocos usos han sido encontrados para los números hiper-complejos de rango $n > 4$. Ya que nuestro interés es el área de la robótica y para una variedad de aplicaciones dinámicas que implica rotaciones, restringiremos la atención en los números hiper-complejos de rango 4, los cuales como se ha dicho tienen el nombre de cuaterniones y serán empleados como operadores de rotación.

En el conjunto de cuaterniones, con las operaciones de adición y producto, se forma un sistema matemático llamado anillo, o de forma más precisa, un anillo de división no conmutativa. Esto último simplemente acentúa el hecho de que el producto entre cuaterniones en general es no conmutativo y que existe el inverso multiplicativo, como es usual para cada elemento diferente de cero en el conjunto.

3.1.1. Definición de cuaternión

Un cuaternión representa una forma alternativa de representar rotaciones a través de cualquier eje, para este caso y para facilitar la identificación de los cuaterniones, éstos serán denotados por las letras p , q o r . En particular serán usadas las letras \mathbf{i} , \mathbf{j} y \mathbf{k} para hacer referencia a las bases ortonormales en \mathbb{R}^3 . Los vectores en el espacio generalmente son escritos como la triada de números reales o escalares, las bases ortonormales pueden ser escritas de la siguiente forma:

$$\begin{aligned}\mathbf{i} &= (1, 0, 0) \\ \mathbf{j} &= (0, 1, 0) \\ \mathbf{k} &= (0, 0, 1)\end{aligned}\tag{3.1}$$

ahora como su nombre lo indica, un cuaternión puede ser considerado como un cuádruple de números reales, es decir como un elemento de \mathbb{R}^4 y puede ser descrito como:

$$q = (q_0, q_1, q_2, q_3)\tag{3.2}$$

donde q_0, q_1, q_2 y q_3 son números reales o escalares.

También puede ser adoptada una forma alternativa para la representación de un cuaternión, primero se define la parte escalar como q_0 , posteriormente se define la parte vectorial como \mathbf{q} , el cual es un vector ordinario en \mathbb{R}^3

$$q = (\mathbf{i}q_1, \mathbf{j}q_2, \mathbf{k}q_3)\tag{3.3}$$

donde las bases ortonormales (\mathbf{i} , \mathbf{j} y \mathbf{k}) en \mathbb{R}^3 se conservan, así el *cuaternión* se puede definir como:

$$q = q_0 + \mathbf{q} = q_0 + \mathbf{i}q_1 + \mathbf{j}q_2 + \mathbf{k}q_3\tag{3.4}$$

donde q_0, q_1, q_2 y q_3 son números reales llamados *componentes del cuaternión*, donde la base canónica para \mathbb{H} es $(1, i, j, k)$. Para extensión de los números complejos de \mathbb{C} hay que tomar en cuenta que $\mathbf{i}^2 = \mathbf{j}^2 = \mathbf{k}^2 = -1$.

Definido de esta forma un cuaternión es un objeto matemáticamente extraño, la suma de un escalar y un vector, algo no definido en el álgebra lineal ordinaria. Por lo tanto es necesario dar más significado a esta definición mostrando cómo los cuaterniones deben ser sumados y multiplicados.

3.1.2. Igualdad y adición

Se dice que los cuaterniones \mathbf{p} y \mathbf{q} son *iguales* si y sólo si, los componentes de ambos son exactamente iguales, es decir:

$$\mathbf{p} = p_0 + \mathbf{i}p_1 + \mathbf{j}p_2 + \mathbf{k}p_3 \quad \text{y} \quad \mathbf{q} = q_0 + \mathbf{i}q_1 + \mathbf{j}q_2 + \mathbf{k}q_3$$

entonces el cuaternión \mathbf{p} es igual al cuaternión \mathbf{q} , si y sólo si:

$$p_0 = q_0$$

$$p_1 = q_1$$

$$p_2 = q_2$$

$$p_3 = q_3$$

la adición de los cuaterniones se define como: la suma de sus componentes correspondientes, es decir:

$$\mathbf{p} + \mathbf{q} = (p_0 + q_0) + \mathbf{i}(p_1 + q_1) + \mathbf{j}(p_2 + q_2) + \mathbf{k}(p_3 + q_3) \quad (3.5)$$

la adición de cuaterniones, es exactamente igual que la adición de vectores cuádruples de números reales y cumple con las propiedades de campo aplicadas a estos. Se puede observar que la adición de cuaterniones tiene como resultado otro cuaternión, esto significa que, el conjunto de los cuaterniones es cerrado bajo la adición. También se puede establecer la existencia de un cuaternión *cerro*, el cual tiene como valor 0 en cada una de sus componentes.

De igual forma, cada cuaternión q tiene un negativo o un inverso aditivo (denotado por $-q$) en el cual cada componente corresponde al negativo de cada uno de los elementos de q . Incluso se puede verificar que la adición de cuaterniones es asociativa y conmutativa, debido a que la adición de números reales tiene estas propiedades.

3.1.3. Producto

En el desarrollo de esta sección se definirán las formas de producto entre un cuaternión y un escalar así como el producto entre cuaterniones.

a. Producto de un cuaternión por un escalar

Este tipo de producto obedece a las reglas de los vectores en \mathbb{R}^3 , por lo que se define de forma directa, es decir, si c es un escalar y q es un cuaternión definido por:

$$p = p_0 + ip_1 + jp_2 + kp_3$$

entonces la resultante del producto del cuaternión q por el escalar c está dado por:

$$cq = cq_0 + icq_1 + jcq_2 + kcq_3 \quad (3.6)$$

Así, al multiplicar el cuaternión por un escalar, solo se multiplica cada componente del cuaternión por el escalar. Nótese que el resultado es nuevamente un cuaternión, esto es, el conjunto de los cuaterniones es cerrado bajo el producto por escalares.

b. Producto entre cuaterniones El producto de cuaterniones es definida de tal manera que el producto fundamental especial satisfaga:

$$\begin{aligned} i^2 &= j^2 = k^2 = ijk = -1 \\ ij &= k = -ji \\ jk &= i = -kj \\ ki &= -j = -ik \end{aligned} \quad (3.7)$$

Teniendo en cuenta que los productos mostrados en Ecuación 3.7 son los que Hamilton vio como necesarios para alcanzar su objetivo. También se puede notar

que el producto de estos números complejos no es conmutativo, por lo tanto, se puede establecer esta misma propiedad para el producto entre cuaterniones.

Utilizando las reglas ordinarias para el producto algebraico y los productos fundamentales mostrados en la Ecuación 3.7, es posible verificar que el producto de cuaterniones debe seguir la siguiente forma:

Si p y q son los cuaterniones;

$$p = p_0 + \mathbf{i}p_1 + \mathbf{j}p_2 + \mathbf{k}p_3 \quad \text{y} \quad q = q_0 + \mathbf{i}q_1 + \mathbf{j}q_2 + \mathbf{k}q_3$$

el producto derivado de éstos es:

$$\begin{aligned} pq &= (p_0 + \mathbf{i}p_1 + \mathbf{j}p_2 + \mathbf{k}p_3)(q_0 + \mathbf{i}q_1 + \mathbf{j}q_2 + \mathbf{k}q_3) \\ &= p_0q_0 + \mathbf{i}p_1q_0 + \mathbf{j}p_2q_0 + \mathbf{k}p_3q_0 \\ &\quad + \mathbf{i}p_0q_1 + \mathbf{i}^2p_1q_1 + \mathbf{i}p_2q_1 + \mathbf{i}p_3q_1 \\ &\quad + \mathbf{j}p_0q_2 + \mathbf{j}p_1q_2 + \mathbf{j}^2p_2q_2 + \mathbf{j}p_3q_2 \\ &\quad + \mathbf{k}p_0q_3 + \mathbf{k}p_1q_3 + \mathbf{k}p_2q_3 + \mathbf{k}^2p_3q_3 \end{aligned} \tag{3.8}$$

Se puede observar que el producto se encuentra escrito de manera uniforme, la parte escalar se encuentra a la derecha del vector y se mantiene el orden correcto en el orden de los vectores, esto con el fin de hacer notoria la aplicación de las reglas mencionadas en la Ecuación 3.7, reordenando los términos se tiene:

$$\begin{aligned} pq &= p_0q_0 + \mathbf{i}p_1q_0 + \mathbf{j}p_2q_0 + \mathbf{k}p_3q_0 \\ &\quad + \mathbf{i}p_0q_1 - p_1q_1 + \mathbf{k}p_2q_1 - \mathbf{j}p_3q_1 \\ &\quad + \mathbf{j}p_0q_2 - \mathbf{k}p_1q_2 - p_2q_2 + \mathbf{i}p_3q_2 \\ &\quad + \mathbf{k}p_0q_3 + \mathbf{j}p_1q_3 - \mathbf{i}p_2q_3 - p_3q_3 \end{aligned} \tag{3.9}$$

realizando arreglos algebraicos y agrupación de términos, la expresión se puede reescribir como:

$$\begin{aligned} pq &= p_0q_0 - (p_1q_1 + p_2q_2 + p_3q_3) \\ &\quad + p_0(\mathbf{i}q_1 + \mathbf{j}q_2 + \mathbf{k}q_3) + q_0(\mathbf{i}p_1 + \mathbf{j}p_2 + \mathbf{k}p_3) \\ &\quad + \mathbf{i}(p_2q_3 - p_3q_2) + \mathbf{j}(p_3q_1 - p_1q_3) + \mathbf{k}(p_1q_2 - p_2q_1) \end{aligned} \tag{3.10}$$

después de escribir la expresión anterior, se puede observar que la dichos productos son iguales al producto punto y al producto cruz de algebra de vectores en \mathbb{R}^3 .

Para poder reescribir la Ecuación 3.10 de forma más practica:

$$\mathbf{a} = (a_1, a_2, a_3) \quad \mathbf{b} = (b_1, b_2, b_3)$$

El producto punto esta dado por:

$$\mathbf{a} \cdot \mathbf{b} = a_1 b_1 + a_2 b_2 + a_3 b_3$$

Y el producto cruz es:

$$\begin{aligned} \mathbf{a} \times \mathbf{b} &= \begin{bmatrix} \mathbf{i} & \mathbf{j} & \mathbf{k} \\ a_1 & a_2 & a_3 \\ b_1 & b_2 & b_3 \end{bmatrix} \\ &= \mathbf{i}(a_2 b_3 - a_3 b_2) + \mathbf{j}(a_3 b_1 - a_1 b_3) + \mathbf{k}(a_1 b_2 - a_2 b_1) \end{aligned}$$

tomando los las definiciones del producto punto y producto cruz, se puede definir de forma más concisa al producto entre dos cuaterniones como:

$$pq = p_0 q_0 - \mathbf{p} \cdot \mathbf{q} + p_0 \mathbf{q} + q_0 \mathbf{p} + \mathbf{p} \times \mathbf{q} \quad (3.11)$$

La Ecuación 3.11 es la definición que se usara a lo largo de este trabajo cuando se haga referencia al producto de dos cuaterniones.

El producto de dos cuaterniones (definido en la Ecuación 3.11) también puede ser definido como un cuaternión resultante de la siguiente forma:

$$pq = r = r_0 + \mathbf{r} = r_0 + \mathbf{i}r_1 + \mathbf{j}r_2 + \mathbf{k}r_3$$

entonces si tomamos los términos de la Ecuación 3.10, tenemos:

$$\begin{aligned} r_0 &= p_0 q_0 - p_1 q_1 - p_2 q_2 - p_3 q_3 \\ r_1 &= p_0 q_1 + p_1 q_0 + p_2 q_3 - p_3 q_2 \\ r_2 &= p_0 q_2 - p_1 q_3 - p_2 q_1 + p_3 q_1 \\ r_3 &= p_0 q_3 + p_1 q_2 - p_2 q_1 + p_3 q_0 \end{aligned} \quad (3.12)$$

para simplificar esta ecuación se puede escribir de forma matricial como:

$$\begin{bmatrix} r_0 \\ r_1 \\ r_2 \\ r_3 \end{bmatrix} = \begin{bmatrix} p_0 & -p_1 & -p_2 & -p_3 \\ p_1 & p_0 & -p_3 & p_2 \\ p_2 & p_3 & p_0 & -p_1 \\ p_3 & -p_2 & p_1 & p_0 \end{bmatrix} \begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \end{bmatrix} \quad (3.13)$$

Lo que concierne a la adición de cuaterniones satisface las propiedades de campo. También se menciona que el conjunto de cuaterniones es un anillo dividido no conmutativo, es decir, satisface las propiedades de campo en el producto, con excepción de la conmutativa. Las siguientes notas son solo para enfatizar en el producto de cuaterniones:

Nota 1: El producto de dos cuaterniones tiene como resultado otro cuaternión con una parte escalar;

$$p_0q_0 - \mathbf{p} \cdot \mathbf{q}$$

y una parte vectorial;

$$p_0\mathbf{q} + q_0\mathbf{p} + \mathbf{p} \times \mathbf{q}$$

de tal manera que el conjunto de los cuaterniones es cerrado para la adición como para el producto.

Nota 2: El producto entre cuaterniones es asociativo.

Nota 3: Debido a que el producto entre $\mathbf{p} \times \mathbf{q}$ no es conmutativo, el producto entre cuaterniones tampoco lo es.

Por otro lado, al igual que en la adición, el conjunto de los cuaterniones tiene una identidad multiplicativa, este cuaternión consta de una parte escalar positiva y la parte vectorial igual a cero, es decir:

$$q = 1 + 0$$

Finalmente hay que remarcar que todo cuaternión diferente de cero tiene un inverso multiplicativo acentuando el hecho de que el conjunto de los cuaterniones es un anillo dividido no conmutativo.

3.1.4. Complejo conjugado

El complejo conjugado es un concepto algebraico importante relacionado con el cuaternión, el cual se define de la siguiente forma:

Sea p un cuaternión de la forma $p = p_0 + \mathbf{p}$, el complejo conjugado p^* es un cuaternión $p = p_0 - \mathbf{p}$ es decir:

$$p = p_0 + \mathbf{p} = p_0 + p_1 + p_2 + p_3$$

el complejo conjugado es:

$$p^* = p_0 - \mathbf{p} = p_0 - p_1 - p_2 - p_3 \quad (3.14)$$

Como ejemplo del uso del complejo conjugado de un cuaternión, se puede demostrar que para dos cuaterniones cualquiera el complejo conjugado de su producto es igual al producto en orden invertido del complejo conjugado individual de ambos cuaterniones. Es decir, dados los cuaterniones p y q tenemos:

$$(pq)^* = q^*p^*$$

se puede notar que para cualquier cuaternión p , la adición con su complejo conjugado p^* , tiene como resultado un escalar de la siguiente forma:

$$p + p^* = (p_0 + \mathbf{p}) + (p_0 - \mathbf{p}) = 2p_0$$

3.1.5. Norma del cuaternión

Para el desarrollo de algunas operaciones entre cuaterniones así como en la aplicaciones en la robótica es necesario tener en cuenta el concepto algebraico de *norma* del cuaternión, el cual se define como:

$$N_{(q)} = \sqrt{q^*q} \quad (3.15)$$

el resultado de la norma del cuaternión es un escalar, definido por N , lo anterior se puede comprobar mediante:

$$\begin{aligned}
 N_{(q)}^2 &= (q_0 - \mathbf{q})(q_0 + \mathbf{q}) \\
 &= q_0 q_0 - (-\mathbf{q}) \cdot \mathbf{q} + q_0 \mathbf{q} + (-\mathbf{q}) q_0 + (-\mathbf{q}) \times \mathbf{q} \\
 &= q_0^2 + \mathbf{q} \cdot \mathbf{q} \\
 &= |\mathbf{q}|^2
 \end{aligned}$$

Esta definición es la misma para un cuaternión en \mathbb{R}^4 , que para vectores en \mathbb{R}^3 , por lo que tiene el mismo significado que la norma Euclidiana. Es importante considerar que para el desarrollo de este trabajo es necesario utilizar los cuaterniones de norma 1 (cuaternión unitario). Como característica de los cuaterniones unitarios se observa que cada uno de sus componentes debe tener un valor absoluto es menor o igual a 1.

3.1.6. Interpretación geométrica del cuaternión

En la Sección 2.3.1, se hizo referencia, de como las rotaciones en \mathbb{R}^3 , son generalmente representadas por matrices de 3×3 , como se mostró, estas matrices son ortogonales con determinante $+1$, de manera alternativa, la representación geométrica de esta matrices puede ser la de un operador rotacional en \mathbb{R}^3 . Esto se puede mostrar cuando se pretende encontrar la imagen \mathbf{w} de un vector \mathbf{v} , el cual, ha tenido una rotación, esto se realiza mediante la representación en forma matricial de dicho vector en forma de matriz columna, cuyas entradas son las componentes vectoriales x , y y z de \mathbf{v} , realizando la multiplicación por la izquierda de este ultimo por la matriz de rotación, así obtenemos de forma matricial, el vector imagen \mathbf{w} .

$$\mathbf{w} = A\mathbf{v} \tag{3.16}$$

Es conocido que los cuaterniones son una forma alternativa de un operador rotacional, para poder llegar a interpretación geométrica de como los cuaterniones juegan este papel, un poco diferente de los operadores rotacionales. Los cuaterniones pueden ser

eficientes al momento de realizar rotaciones sobre \mathbb{R}^3 , ya que como se ha mencionado un cuaternión es un vector formado por un vector en \mathbb{R}^4 , mientras que las matrices de rotación tienen nueve elementos. Para mostrar de forma clara cómo un cuaternión funge como operador rotacional, lo cual forma parte del objetivo de este trabajo, es necesario explorar, entender y despejar esta interrogativa. En el desarrollo de esta sección se pretende realizar el desglose de algunas consideraciones necesarias para poder definir el operador rotacional con cuaterniones.

3.1.6.1. Desarrollo algebraico del operador rotacional

El interés general en el desarrollo de este trabajo es desarrollar un modelo matemático que determine la orientación de un objeto en el espacio 3D. Un cuerpo que se encuentra definido en el espacio \mathbb{R}^3 , se puede considerar que está definido por una serie de puntos dentro de dicho espacio, por lo tanto, se puede identificar a estos puntos como vectores en \mathbb{R}^3 , así, la orientación de dicho objeto puede ser estudiada mediante las operaciones que describen a los cuerpos definidos en \mathbb{R}^3 . Ahora bien, para lograr este objetivo es necesario definir un operador en términos de cuaterniones que describa estas operaciones, para llegar a una concepción más clara de cómo lograr esto, se puede comenzar por realizar un cuestionamiento importante:

¿Cómo puede un cuaternión, que pertenece a \mathbb{R}^4 , operar sobre vectores, que pertenecen a \mathbb{R}^3 ?

La respuesta a esta pregunta, para algunos obvia, es posible mediante la siguiente correspondencia:

Un vector $\mathbf{v} \in \mathbb{R}^3$, puede ser tratado como un cuaternión $\mathbf{q} \in \mathbb{R}^4$ cuya parte real es cero.

mediante esta correspondencia los vectores pueden ser tomados como cuaterniones, a los cuales se les conoce como: *cuaterniones puros* [5]. Si se considera la posibilidad que Q_0 sea el conjunto de los cuaterniones puros, los cuales son un subconjunto de Q (conjunto de los cuaterniones). De acuerdo con esto se puede definir una correspondencia uno-a-uno entre los elementos de que se encuentran en estos dos conjuntos, en dicha correspondencia un vector $\mathbf{v} \in \mathbb{R}^3$ equivale a un cuaternión puro de al siguiente

forma $v = 0 + \mathbf{v} \in Q_0$, esto es:

$$\mathbf{v} \in \mathbb{R}^3 \leftrightarrow v = 0 + \mathbf{v} \in Q_0 \subset Q \tag{3.17}$$

En la Figura 3.1, es posible observar que para la adición y multiplicación por un escalar esta correspondencia es clara, es decir, la adición de dos vectores cualquiera ($\mathbf{v} \in \mathbb{R}^3$) corresponde a la suma de cada uno de sus cuaterniones puros correspondientes en Q_0 . De acuerdo con la Figura 3.1, se puede suponer que, un operador rotacional definido

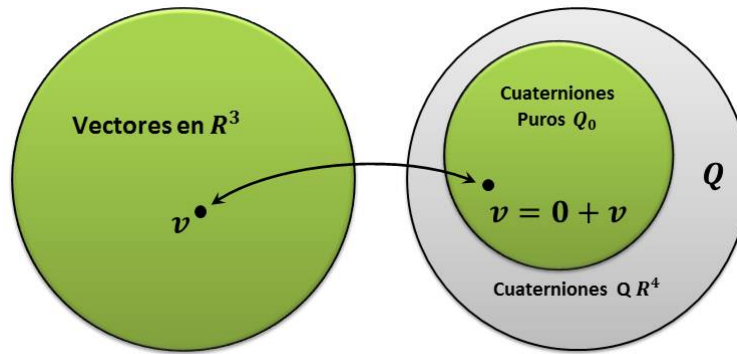


FIGURA 3.1: Correspondencia: Vectores \leftrightarrow Cuaterniones [5].

mediante cuaterniones, opera de la misma forma que los operadores rotacionales matriciales. Es decir que de alguna manera $q \in Q$ representa una rotación y la imagen de rotación $\mathbf{W} \in \mathbb{R}^3$ un vector $\mathbf{v} \in \mathbb{R}^3$ puede ser encontrada al aplicar la regla de los operadores matriciales:

$$\mathbf{w} = q\mathbf{v}$$

Dicha regla, hace notar que, el simple producto de un cuaternión q por un vector \mathbf{v} puede no ser definido, pero el resultado siempre será un vector. Para que el producto se lleva a cabo es necesario hacer uso de la correspondencia uno-a-uno entre \mathbb{R}^3 y Q_0 descrita en la Ecuación 3.17. Para el producto entre cuaterniones descrito en la Ecuación 3.11, sustituimos el vector \mathbf{v} por el cuaternión puro correspondiente $v = 0 + \mathbf{v}$. De este modo, teniendo cualquier cuaternión $q = q_0 + \mathbf{q} \in Q$ y cualquier vector $\mathbf{v} \in \mathbb{R}^3$, el producto se

realiza de forma directa:

$$\begin{aligned}
 \mathbf{q}\mathbf{v} &= (q_0 + \mathbf{q})(0 + \mathbf{v}) \\
 &= q_0 \cdot 0 - \mathbf{q} \cdot \mathbf{v} + 0 \cdot \mathbf{q} + q_0 \mathbf{v} + \mathbf{q} \times \mathbf{v} \\
 &= -\mathbf{q} \cdot \mathbf{v} + q_0 \mathbf{v} + \mathbf{q} \times \mathbf{v}
 \end{aligned}
 \tag{3.18}$$

Cómo se puede observar, en la Ecuación 3.18, el resultado de la interpretación del operador rotacional definido mediante cuaterniones opera de la misma forma que los operadores matriciales, no es un vector $\mathbf{v} \in \mathbb{R}^3$ como se espera excepto en el caso especial que $\mathbf{q} \cdot \mathbf{v} = 0$, lo cual, hace notar que \mathbf{q} y \mathbf{v} son ortogonales. Por lo tanto, no es posible esperar que el operador rotacional en termino de cuaterniones consista simplemente en un solo cuaternión. Ahora bien si se considera que la conmutatividad de estos dos cuaterniones no cambia la parte real del cuaternión resultante se puede desechar esta opción. Sin embargo, si consideramos que q y s son dos cuaterniones del conjunto Q y v es un tercer cuaternión que pertenece al subconjunto Q_0 , es decir un cuaternión puro, que representa al vector $\mathbf{v} \in \mathbb{R}^3$, existen seis posibles combinaciones para que el producto:

$$\begin{array}{ccc}
 vqs & qsv & svq \\
 vsq & sqv & qvs
 \end{array}$$

Ahora, una de las propiedades algebraicas del cuaternión es que el conjunto Q es cerrado bajo multiplicación (no así el conjunto Q_0). Por lo tanto, los productos qs y sq de la primera y segunda columna son simplemente cuaterniones, lo cual significa que el producto entre estos es esencialmente un producto doble (véase Ecuación 3.11). Puesto que este producto no es adecuado para definir a un operador rotacional entonces se descartan estas posibilidades. Sí v es un cuaternión puro, que representa un vector, y no hacemos ninguna distinción entre el cuaternión q y el cuaternión r , los dos productos restantes de la anterior lista no tienen ninguna distinción, de ahí que podríamos tomar cualquier producto. En nuestro caso tomaremos el producto triple qvs .

Al tener en cuenta que cada uno de los cuaterniones corresponde a: $q = q_0 + \mathbf{q}$, $v = 0 + \mathbf{v}$ y $s = s_0 + \mathbf{s}$, se puede verificar mediante la Ecuación 3.11 que el resultado es:

$$-s_0(\mathbf{q} \cdot \mathbf{v}) - q_0(\mathbf{v} \cdot \mathbf{s}) - (\mathbf{q} \times \mathbf{v}) \cdot \mathbf{s}$$

utilizando las propiedades del álgebra vectorial se puede reescribir la el resultado de la siguiente forma:

$$-s_0(\mathbf{q} \cdot \mathbf{v}) - q_0(\mathbf{s} \cdot \mathbf{v}) - (\mathbf{q} \times \mathbf{s}) \cdot \mathbf{v}$$

Si lo que se busca es que el resultado de este operador rotacional sea un cuaternión puro (que represente un vector), por lo tanto se espera que, la parte real de este cuaternión resultante sea cero.

¿Cómo lograr que la parte real de este cuaternión sea cero?

Se puede plantear la siguiente suposición, si $s_0 = -q_0$, tenemos que la parte real del cuaternión resultante se reescribe de la siguiente forma:

$$-q_0(\mathbf{q} + \mathbf{s}) \cdot \mathbf{v} + (\mathbf{q} \times \mathbf{s}) \cdot \mathbf{v}$$

como se puede ver de forma clara la parte real sera cero sí, $\mathbf{s} = -\mathbf{q}$, para lograr esto, recurrimos a la siguiente igualdad:

$$s = s_0 + \mathbf{s} = q_0 - \mathbf{q} = \mathbf{q}^* \implies \mathbf{q} = \mathbf{s}^*$$

De esta forma el se hace evidente que el triple producto de estos cuaterniones tiene como resultado un cuaternión puro o vector $\mathbf{v} \in \mathbb{R}^3$, siempre que \mathbf{v} sea un cuaternión puro.

$$qvq^* \quad \text{y} \quad q^*vq$$

Ahora bien, al aplicar este producto al vector de entrada p , considerando que p es un cuaternión puro $p = 0 + \mathbf{p}$ y considerando los dos posibles operadores tenemos que:

$$\begin{aligned} \mathbf{w}_1 &= qpq^* \\ \mathbf{w}_2 &= q^*pq \end{aligned} \tag{3.19}$$

Mediante la Figura 3.2 se puede tener una interpretación gráfica de la acción algebraica de la Ecuación 3.19. De acuerdo con estos dos operadores se ha encontrado una forma consistente de manejar vectores $v \in \mathbb{R}^3$ en el espacio de los cuaterniones $Q \in \mathbb{R}^4$, y se ha demostrado cómo, mediante estos dos operadores, se tornan vectores en vectores. después del desarrollo algebraico al cual se ha llegado surge otra pregunta que es de

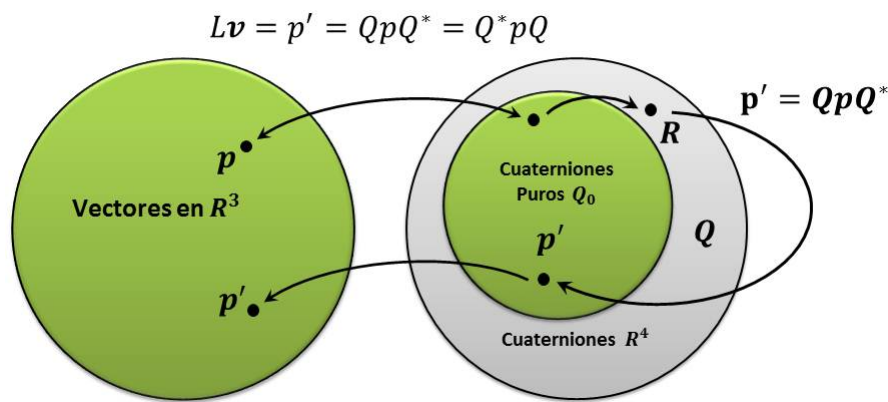


FIGURA 3.2: Acción del operador rotacional en termino de cuaterniones [5].

gran importancia para el desarrollo de este trabajo, y es:

¿Cuál es la interpretación, geoméricamente hablando, que se puede dar a estos dos operadores?

3.1.6.2. Interpretación geométrica del operador

El desarrollo de la Ecuación 3.19, conduce a la búsqueda de una asociación geométrica que represente la rotación de puntos o vectores $v \in \mathbb{R}^3$, como se concibe de forma general, lo cual lleva a la siguiente interrogante:

¿Existe alguna forma de asociar un ángulo a un cuaternión, análogo a la forma que se realiza con las matrices de rotación conocidas?

De ser esto posible, esta asociación, se podrá llevar a cabo para los operadores del cuaternión propuestos. De acuerdo con [4] Hamilton hallo la respuesta a esta pregunta y en nuestra consideración se ha descubierto un hecho interesante y provechoso.

Basado en esto el cuaternión q , que describe al operador, deberá ser de norma 1, es decir, se deberá trabajar con cuaterniones normalizados, esto debido a las consideraciones que a continuación se harán evidentes.

Como se ha mencionado, el cuaternión se define como:

$$q = q_0 + \mathbf{q}$$

por lo que el cuaternión de norma 1, se puede decir que:

$$q_0^2 + |\mathbf{q}|^2 = 1$$

También es fácil verificar que, de las identidades trigonométricas, para cualquier ángulo dado se tiene:

$$\cos^2(\theta) + \operatorname{sen}^2(\theta) = 1$$

entonces debe existir un ángulo θ tal que:

$$\cos^2(\theta) = q_0^2 \quad \text{y} \quad \operatorname{sen}^2(\theta) = |\mathbf{q}|^2$$

Ahora bien, el ángulo θ se define dentro de la restricción apropiada de su dominio. En general, θ satisface:

$$-\pi < \theta \leq \pi$$

Para definir la asociación del ángulo a θ , al cuaternión unitario $q = q_0 + \mathbf{q}$, es conveniente definir el vector unitario \mathbf{u} , este último representa la dirección de q , de tal manera que:

$$\mathbf{u} = \frac{\mathbf{q}}{|\mathbf{q}|} = \frac{\mathbf{q}}{\operatorname{sen}(\theta)}$$

de esta forma se puede escribir el cuaternión unitario en términos del ángulo θ y el vector unitario \mathbf{u} , de la siguiente forma:

$$q = q_0 + \mathbf{q} = \cos(\theta) + \mathbf{u}\operatorname{sen}(\theta)$$

es posible notar que de esta forma puede ser sustituido el θ por el $-\theta$, para conseguir el complejo conjugado de q :

$$\begin{aligned}\cos(-\theta) + \mathbf{u}\text{sen}(-\theta) &= \cos(\theta) + \mathbf{u}\text{sen}(-\theta) \\ &= \cos(\theta) - \mathbf{u}\text{sen}(\theta) \\ &= q^*\end{aligned}$$

3.1.6.3. Análisis geométrico del cuaternión

Para explorar mas a detalle el significado geométrico relacionado a los cuaterniones, y el operador rotacional propuesto, se puede analizar la secuencia de dos rotaciones desde el punto de vista geométrico.

Considerando una esfera, o el conjunto de de puntos que constituyen la superficie de esa esfera. Sí se toma el centro de esta esfera como el origen de un sistema fijo de coordenadas \mathbb{R}^3 . Esto supone que cualquier rotación de la esfera sobre un vector sobre este sistema de referencia, moverá los puntos de la superficie de esa esfera a en la superficie. Como resultado de esto se tiene una trayectoria de los puntos de la misma superficie, conservando la distancia entre ellos.

Esto también conserva la distancia y la dirección de el ángulo entre dos arcos cualquiera. Cada punto tiene dos puntos invariantes, comúnmente llamados, el eje de rotación y la superficie de la esfera.

Ahora, sí se suponen dos puntos cualquiera A y B dentro de la superficie de la esfera, esos dos puntos definen a los vectores \vec{a} y \vec{b} fijos en el sistema de coordenadas y se toma esos vectores como ejes de rotación dentro de la esfera.

Al considerar realizar dos rotaciones sucesivas: la primera sobre el eje \vec{a} , seguida por la otra rotación sobre el eje \vec{b} , se podría plantear la siguiente pregunta

¿Puede existir un eje de rotación en la que se pueda describir la secuencia de rotaciones realizadas?

Tal como ya se ha mencionado a lo largo de este trabajo la respuesta a esta pregunta es afirmativa. Aunque, en cual de los sistemas antes mencionados se puede confirmar este hecho geoméricamente. Se puede comenzar por representar una esfera unitaria la cual define dos puntos que permanecen invariantes bajo cualquier secuencia de rotaciones dada. Esta construcción relaciona los dos ejes dados y sus respectivas rotaciones con

un eje de rotación compuesto equivalente definido por los dos puntos invariantes. Si se consideran dos puntos arbitrarios distintos, tales como A y B en la Figura 3.3, en la superficie de la esfera unitaria con centro en el origen de un punto fijo de referencia en \mathbb{R}^3 . Asociados con esos dos puntos son relacionados a sus vectores unitarios, designados como \mathbf{a} y \mathbf{b} , respectivamente. Las rotaciones de la esfera dentro de si misma son tomadas, sobre esos dos vectores como ejes. Dado que la secuencia de dos rotaciones consiste primero en una rotación sobre el eje definido por el vector unitario \mathbf{a} , por medio de un ángulo α , seguido por una segunda rotación sobre el eje definido por el vector unitario \mathbf{b} , a través de un ángulo β .

La siguiente construcción nos permite encontrar la dirección del, definida por el vector unitario \mathbf{c} , por el eje de la rotación compuesta por $R[\mathbf{c}, \gamma]$, considerado los puntos C y C' , en la superficie de la esfera, localizando de esta forma $\angle CAB = \angle C'AB = \frac{\alpha}{2}$ y $\angle CBA = \angle C'BA = \frac{\alpha}{2}$, ahora si tenemos que la rotación es con sentido al derecha en el punto A, con un ángulo α , se puede observar que el arco formado por el segmento AC y AC' tienen la misma longitud que si tomamos el punto C a C' , es decir,

$$R[\mathbf{a}, \alpha]C = C'$$

En la siguiente rotación ahora con respecto al eje formado por el vector \mathbf{b} con un ángulo β , es posible notar que existe una relación con la rotación anterior, por esta razón es posible, verificar que, geométicamente el punto C' regresa al punto C, mediante esta rotación, es decir,

$$R[\mathbf{b}, \beta]C' = C$$

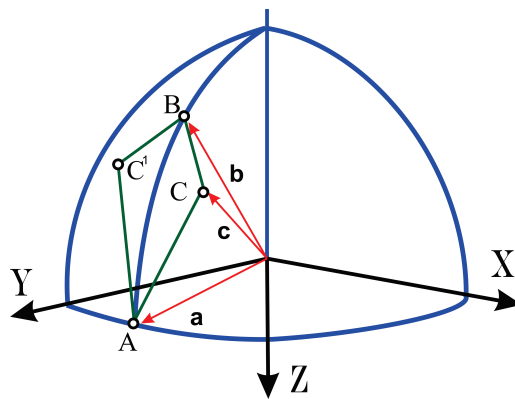


FIGURA 3.3: Construcción de dos rotaciones [5].

por lo tanto se puede hacer notar que el punto C es un punto fijo bajo cualquier secuencia de estas dos rotaciones por lo que se puede establecer la siguiente relación:

$$R[\mathbf{b}, \beta]R[\mathbf{a}, \alpha]C = R[\mathbf{b}, \beta]C' = C$$

La representación de rotaciones mediante el operador rotacional de cuaterniones es algebraicamente conveniente y consistente dado que como se la construcción geométrica de una rotación del punto C (la cual define al vector unitario \mathbf{b} y \mathbf{a} de como ejes de rotación) se puede establecer que:

$$A' = R[\mathbf{c}, \gamma]A$$

Usando la Ecuación 3.19 esta rotación $R[\mathbf{c}, \gamma]A$ se puede expresar en la forma del operador rotacional de cuaterniones como:

$$\mathbf{w} = L_q(\mathbf{v}) = q_{\mathbf{c}, \gamma} \mathbf{p} q_{\mathbf{c}, \gamma}^*$$

para este cuaternión de rotación el operador para $q_{\mathbf{c}, \gamma}$ y $q_{\mathbf{c}, \gamma}^*$ esta dado por: $q_{\mathbf{c}, \gamma} = \cos\left(\frac{\gamma}{2}\right) + \mathbf{c} \sin\left(\frac{\gamma}{2}\right)$ y $q_{\mathbf{c}, \gamma}^* = \cos\left(\frac{\gamma}{2}\right) - \mathbf{c} \sin\left(\frac{\gamma}{2}\right)$

Tanto en la dirección compuesta por el eje \mathbf{c} y el ángulo γ de rotación sobre este eje, se pueden expresar como parámetros de que constituyen a cada uno de los dos

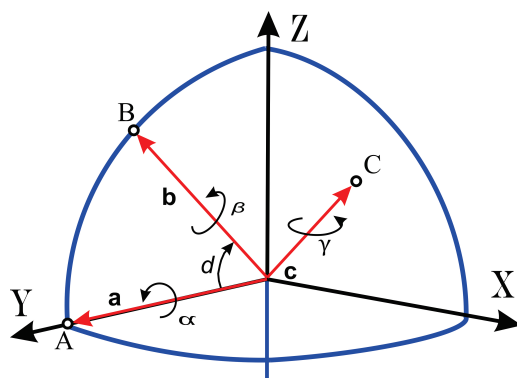


FIGURA 3.4: Secuencia de rotaciones [5].

cuaterniones constituidos por $q_{a,\alpha}$ y $q_{b,\beta}$, de esta forma localizamos los dos ejes de rotación constituidos por, \mathbf{a} y \mathbf{b} Figura 3.4, donde se muestran los puntos A y B que están localizados en en la esfera unitaria. El eje de rotación \mathbf{a} esta localizado justo a lo largo del eje de referencia X, y el eje de rotación \mathbf{b} se encuentra en el plano de referencia ZX. Los puntos A y B se encuentran separados por una distancia radial d, la cual es el ángulo entre los dos ejes de rotación por lo tanto:

$$\mathbf{b} = \mathbf{i}\cos(d) + \mathbf{k}\sin(d)$$

al recordar que el cuaternión $q_{c,\gamma}$ es igual al producto de los cuaterniones constituidos por $q_{a,\alpha}$ y $q_{b,\beta}$ que es:

$$q_{c,\gamma} = \cos\left(\frac{\gamma}{2}\right) + \mathbf{c}\sin\left(\frac{\gamma}{2}\right) = q_{b,\beta}q_{a,\alpha}$$

al utilizar los cuaterniones con su forma trigonométrica, el vector \mathbf{b} se calcula de la siguiente forma:

$$\begin{aligned} q_{c,\gamma} &= \cos\left(\frac{\gamma}{2}\right) + \mathbf{c}\left(\frac{\gamma}{2}\right) & (3.20) \\ &= q_{b,\beta}q_{a,\alpha} \\ &= \left(\cos\left(\frac{\beta}{2}\right) + \mathbf{b}\sin\left(\frac{\beta}{2}\right)\right) \left(\cos\left(\frac{\alpha}{2}\right) + \mathbf{i}\sin\left(\frac{\alpha}{2}\right)\right) \\ &= \left(\cos\left(\frac{\alpha}{2}\right)\cos\left(\frac{\beta}{2}\right) - \sin\left(\frac{\alpha}{2}\right)\cos(d)\sin\left(\frac{\beta}{2}\right)\right) \\ &\quad + \mathbf{i}\left(\sin\left(\frac{\alpha}{2}\right)\cos\left(\frac{\beta}{2}\right) + \cos\left(\frac{\alpha}{2}\right)\cos(d)\sin\left(\frac{\beta}{2}\right)\right) \\ &\quad + \mathbf{j}\left(\sin\left(\frac{\alpha}{2}\right)\sin(d)\sin\left(\frac{\beta}{2}\right)\right) \\ &\quad + \mathbf{k}\left(\cos\left(\frac{\alpha}{2}\right)\sin(d)\sin\left(\frac{\beta}{2}\right)\right) & (3.21) \end{aligned}$$

mediante este análisis se puede observar que la expresión corresponde a la parte *real* y la parte *vectorial* del cuaternión $q_{c,\gamma}$ mostrados en las Ecuaciones 3.20 y 3.21, al igualar la parte real del cuaternión tenemos:

$$\cos\left(\frac{\gamma}{2}\right) = \cos\left(\frac{\alpha}{2}\right)\cos\left(\frac{\beta}{2}\right) - \sin\left(\frac{\alpha}{2}\right)\cos(d)\sin\left(\frac{\beta}{2}\right)$$

de la parte vectorial del cuaternión se puede obtener el eje de rotación el cual se encuentra en el punto C, (ver Figura 3.4), por lo que al igualar la parte vectorial de las ecuaciones 3.20 y 3.21, es posible observar este eje de rotación:

$$\begin{aligned} \mathbf{c} \sin\left(\frac{\gamma}{2}\right) &= \mathbf{i}u_x, \mathbf{j}u_y, \mathbf{k}u_z \\ \text{donde } u_x &= \sin\left(\frac{\alpha}{2}\right) \cos\left(\frac{\beta}{2}\right) + \cos\left(\frac{\alpha}{2}\right) \cos(d) \sin\left(\frac{\beta}{2}\right) \\ u_y &= \sin\left(\frac{\alpha}{2}\right) \sin(d) \sin\left(\frac{\beta}{2}\right) \\ u_z &= \cos\left(\frac{\alpha}{2}\right) \sin(d) \sin\left(\frac{\beta}{2}\right) \end{aligned}$$

de esta forma la composición del eje de rotación se encuentra expresado en forma de un vector unitario, el cuál puede ser calculado de la siguiente forma:

$$\mathbf{c} = \frac{\mathbf{i}u_x + \mathbf{j}u_y + \mathbf{k}u_z}{\sin\left(\frac{\gamma}{2}\right)} \quad (3.22)$$

así se define el eje de rotación resultante, de una composición de rotaciones, como un vector en un sistema de referencia, por lo tanto se tiene:

$$\sin\left(\frac{\gamma}{2}\right) = \sqrt{1 - \cos^2\left(\frac{\gamma}{2}\right)} \quad (3.23)$$

3.1.6.4. Algoritmo del operado rotacional

Es necesario para futuras referencias, tener una forma general para la salida de un vector cuando es aplicada una rotación mediante el uso de la Ecuación 3.19, a un vector de entrada $\mathbf{v} \in \mathbb{R}^3$. Si tenemos el cuaternión $q = q_0 + \mathbf{q}$ y el vector \mathbf{v} , el cual corresponde al cuaternión puro $p = 0 + \mathbf{v}$, tenemos que es posible verificar el siguiente resultado:

$$\begin{aligned} w = qpq^* &= (q_0 + \mathbf{q})(0 + \mathbf{v})(q_0 - \mathbf{q}) \\ &= (2q_0^2 - 1)\mathbf{v} + 2(\mathbf{q} \cdot \mathbf{v})\mathbf{q} - 2q_0(\mathbf{q} \times \mathbf{v}) \\ \text{ó } w = qpq^* &= (2q_0^2 - |\mathbf{q}|^2)\mathbf{v} + 2(\mathbf{q} \cdot \mathbf{v})\mathbf{q} - 2q_0(\mathbf{q} \times \mathbf{v}) \end{aligned} \quad (3.24)$$

Capítulo 4

Modelado vía cuaterniones

En el desarrollo de este capítulo se presenta el estudio del álgebra de cuaterniones el cual permite generar las ecuaciones que puedan predecir el comportamiento cinemático del robot IRB1600. El objetivo es demostrar la validez que tiene el álgebra de cuaterniones aplicado al modelar la cinemática directa, resolver el problema de la cinemática inversa y simular los resultados obtenidos mediante el calculo computacional de una tarea desarrollada por el robot.

Como primera parte se desarrollara el modelo vía cuaterniones como base del planteamiento teórico, esto con el fin de modelar la cinemática directa del robot. En la segunda parte se resuelve la cinemática inversa del robot mediante el de desacoplo cinemático y por ultimo la simulación de este modelo mediante realidad virtual del robot IRB1600.

4.1. Modelado rotacional mediante cuaterniones

Al modelar matemáticamente n cuerpos rígidos que se encuentran acoplados como una cadena cinemática(robot), es necesario que las representaciones sean aplicables a algoritmos que puedan ser implementados de una forma sencilla y que a su vez sean aplicados mediante desarrollos computacionales. El modelado matemático busca hacer un planteamiento analítico con soluciones discretas, esto para algunas cadenas cinemáticas de robots es sencillo, pero generalmente, los modelos resultantes son sistemas matemáticos no lineales que necesitan de soluciones numéricas. El álgebra de cuaterniones provee una forma de modelar algunos tipos de cadenas cinemáticas, manejando

un espacio en \mathbb{R}^4 . Con el uso de este espacio es posible definir transformaciones que representan las rotaciones finitas de un cuerpo rígido. Por otra parte, las representaciones con cuaterniones resultan ser relaciones simbólicas de manejo menos complejo que los sistemas convencionales, por lo que, es posible establecer las ecuaciones que describen la cinemática de algunos robots[23].

Hasta este punto ya se ha revisado el algebra de cuaterniones y sus propiedades fundamentales del espacio vectorial \mathbb{R}^4 , la representación de un espacio vectorial normado por un producto interno, puede ser representado como \mathbb{Q} , en esta sección se realizara el desarrollo de un algoritmo capaz de ser programado para analizar la posición de un cuerpo en el espacio dada una rotación realizada por el operador rotacional cuaterniónico y que pueda ser capaz de ser aplicado a n cuerpos rígidos.

4.1.1. Relación del espacio vectorial de cuaterniones

Al recordar el algebra de cuaterniones (ver Sección 3.1) se tiene que existen dos operaciones con cuaterniones fundamentales en el modelado rotacional, representadas mediante una estructura algebraica en \mathbb{R}^4 , dichas operaciones son:

- ▷ La adición representada en Ecuación 3.5 la cual sera representada por el operador $\oplus : \mathbb{R}^4 \oplus \mathbb{R}^4 \rightarrow \mathbb{R}^4$, esta operación esta definida como:

$$p \oplus q = (p_0 + q_0) + \mathbf{i}(p_1 + q_1) + \mathbf{j}(p_2 + q_2) + \mathbf{k}(p_3 + q_3) \quad (4.1)$$

- ▷ El producto entre cuaterniones representado en Ecuación 3.11 la cual sera representada por el operador $\otimes : \mathbb{R}^4 \otimes \mathbb{R}^4 \rightarrow \mathbb{R}^4$, esta operación esta definida como:

$$p \otimes q = p_0 q_0 - p \cdot q + p_0 q + q_0 p + q \times p \quad (4.2)$$

derivado de las parejas \mathbb{R}^4, \oplus y \mathbb{R}^4, \otimes se define un grupo aditivo conmutativo y un grupo multiplicativo no conmutativo, respectivamente.

Las propiedades algebraicas de este conjunto son definidas de acuerdo con [23, 24] son resumidas en el siguiente teorema:

Teorema 4.1. *la terna $(\mathbb{R}^4, \oplus, \otimes)$ es un campo no conmutativo.*

i. *La operación $\otimes : \mathbb{R}^4 \otimes \mathbb{R}^4 \rightarrow \mathbb{R}^4$ es asociativa debido a:*

$$p \otimes (q \otimes s) = (p \otimes q) \otimes s \quad (4.3)$$

$$\forall p, q, s \in \mathbb{R}^4$$

ii. *El elemento $1 = (1, 0, 0, 0) \in \mathbb{R}^4$ es tal que: $1 \otimes p = p \otimes 1, \forall p \in \mathbb{R}^4$. Éste es el llamado elemento neutro de la multiplicación en \mathbb{R}^4 .*

iii. *$\forall p \in \mathbb{R}^4, p \neq (0, 0, 0, 0)$ existe $p^{-1} \in \mathbb{R}^4$ tal que $p \otimes p^{-1} = 1$. El elemento p^{-1} es llamado la inversa del cuaternión "p" para la multiplicación. La inversa de un cuaternión unitario o normalizado es el complejo conjugado del cuaternión ($p^{-1} = p^*$)*

iv. *La operación $\otimes : \mathbb{R}^4 \otimes \mathbb{R}^4 \rightarrow \mathbb{R}^4$ no es conmutativa ya que:*

$$p \otimes q \neq q \otimes p$$

v. *Las propiedades de distributividad se satisfacen:*

$$(p \oplus q) \otimes s = p \otimes s \oplus q \otimes s$$

$$p \otimes (q \oplus s) = p \otimes q \oplus p \otimes s \quad (4.4)$$

$$\forall p, q, s \in \mathbb{R}^4$$

Para la operación representa por $\bullet : \mathbb{R} \bullet \mathbb{R}^4 \rightarrow \mathbb{R}^4$ definida por:

$$\alpha \bullet (a, b, c, d) = (\alpha a, \alpha b, \alpha c, \alpha d) \quad (4.5)$$

$$\forall (a, b, c, d) \in \mathbb{R}^4, \alpha \in \mathbb{R}$$

lo cual se conoce como multiplicación escalar. Se observa con esto que la relación de la terna $(\mathbb{R}^4, \oplus, \bullet)$ es un espacio vectorial real.

Además se tiene que la transformación $\langle \bullet, \bullet \rangle: \mathbb{R}^4 \bullet \mathbb{R}^4 \rightarrow \mathbb{R}$ definida por:

$$\langle q, p \rangle = \sum_{i=0}^3 p_i q_i \in \mathbb{R}, \quad (4.6)$$

es un producto interno en \mathbb{R}^4 , y por tanto, $Q = (\mathbb{R}^4, \oplus, \otimes, \bullet, \langle \bullet, \bullet \rangle)$ es un espacio vectorial real con producto interno. La norma asociada a este producto interno es la siguiente:

$$|q| = \langle q, q \rangle^{\frac{1}{2}} = \left(q_0^2, q_1^2, q_2^2, q_3^2 \right)^{\frac{1}{2}} \quad (4.7)$$

por lo cual $Q = (\mathbb{R}^4, \oplus, \otimes, \bullet, |\bullet|)$ es un espacio normado al cual se le llamará *espacio vectorial de cuaterniones* y sus elementos serán llamados *cuaterniones*.

4.1.2. Representación paramétrica de las rotaciones de un cuerpo rígido

El objetivo de esta sección es unificar las diferentes formas utilizadas en la mecánica y la robótica para representar paramétricamente las rotaciones finitas de los cuerpos rígidos mediante cuaterniones.

Como se definió en la Sección 3.1.6.1 el operador rotacional $L\mathbf{v} = q \otimes \mathbf{v} \otimes q_*$, corresponde a una rotación en sentido positivo del vector \mathbf{v} con respecto al sistema coordenado de referencia por un ángulo de 2θ el rededor del eje del cuaternión q . Si aplicamos esta rotación al cuaternión puro $v = 0 + \mathbf{v}$ y tomando en cuenta que la rotación se tiene que definir en un ángulo de $\frac{\theta}{2}$ para que con esto pueda representarse una rotación positiva al rededor del vector \mathbf{v} con respecto al sistema coordenado de referencia por un ángulo θ al rededor del eje del cuaternión q , mediante estas consideraciones se puede realizar el siguiente arreglo.

$$L\mathbf{v} = q \otimes \mathbf{v} \otimes q_{-1} = \frac{1}{|q|^2} \bullet (q \otimes \mathbf{v} \otimes q_*) \quad (4.8)$$

$$\forall \mathbf{v} \in Q$$

En general, la norma de q puede ser arbitraria. Sin embargo, en este trabajo se usan cuaterniones unitarios, esto es, $|q| = 1$ y en este caso:

$$L\mathbf{v} = \mathbf{q} \otimes \mathbf{v} \otimes \mathbf{q}_*$$

Ahora bien al utilizar las consideraciones hechas en la Sección 3.1.6.3, un cuaternión \mathbf{q} puede ser escrito como la rotación del ángulo $\theta \in [0, \pi]$ sobre el eje $\mathbf{v} \in Q_v$ de la siguiente forma:

$$\begin{aligned} q_0 &= \pm |\mathbf{q}| \cos \frac{\theta}{2} \\ q_v &= \pm |\mathbf{q}| \operatorname{sen} \frac{\theta}{2} \mathbf{v} \end{aligned}$$

por lo tanto como se había mencionado, para este trabajo los se utilizan cuaterniones unitarios $|\mathbf{q}| = 1$, por lo tanto se tiene:

$$q_0 = \pm \cos \frac{\theta}{2} \tag{4.9}$$

$$q_v = \pm \operatorname{sen} \frac{\theta}{2} \mathbf{v} \tag{4.10}$$

los componentes del cuaternión $\mathbf{q} \in \mathbb{R}^4$ se expresan con los parámetros de Euler por su utilidad en el modelo mediante cuaterniones.

4.1.3. Rotaciones sucesivas de un sistema cartesiano de referencia

En esta sección se presenta la forma de aplicar la rotación de un cuerpo rígido aplicando el algebra de cuaterniones, con lo que posteriormente se podrá aplicar estas rotaciones a los vectores \mathbb{R}^3 .

Para representar esta transformación en la modelación se utilizará $L\mathbf{v} = \mathbf{q} \otimes \mathbf{v} \otimes \mathbf{q}_*$ la cual es una rotación en el espacio de los cuaterniones. Los componentes de $\mathbf{q} \in Q$ están relacionados con los parámetros físicos descritos en las Ecuaciones 4.9 y 4.10.

Si $\mathbf{p} \in Q$ es un cuaternión de rotación y $\mathbf{v} \in Q$ es un cuaternión puro al que se le aplicara una rotación, $L\mathbf{v}$ se simplifica de la siguiente forma:

$$L\mathbf{v} = (\mathbf{q}, \mathbf{v}) \tag{4.11}$$

$$\forall \mathbf{q} \in Q$$

Esta acción del operación de rotación esta simplificada mediante el algebra de cuaterniones, sabiendo que la representación de un cuerpo rígido se tiene hacer mediante el espacio vectorial en \mathbb{R}^3 lo que nos lleva a la simplificación del operador rotacional visto en la Ecuación 3.24, donde al expandir cada uno de los términos se puede notar que la rotación puede ser aplicada al espacio vectorial \mathbb{R}^3 de forma directa, partiendo de la multiplicación del triple producto del operador rotacional $L\mathbf{v} = \mathbf{q} \otimes \mathbf{v} \otimes \mathbf{q}_*$, teniendo el siguiente resultado:

$$\begin{aligned} w = \mathbf{q} \mathbf{p} \mathbf{q}^* &= (\mathbf{q}_0 + \mathbf{q})(0 + \mathbf{v})(\mathbf{q}_0 - \mathbf{q}) \\ &= (2q_0^2 - 1)\mathbf{v} + 2(\mathbf{q} \cdot \mathbf{v})\mathbf{q} - 2q_0(\mathbf{q} \times \mathbf{v}) \\ w = \mathbf{q} \mathbf{p} \mathbf{q}^* &= (2q_0^2 - |\mathbf{q}|^2)\mathbf{v} + 2(\mathbf{q} \cdot \mathbf{v})\mathbf{q} - 2q_0(\mathbf{q} \times \mathbf{v}) \end{aligned}$$

donde expandiendo cada uno de los términos tenemos:

$$\begin{aligned} (2q_0^2 - 1)\mathbf{v} &= \begin{bmatrix} (2q_0^2 - 1) & 0 & 0 \\ 0 & (2q_0^2 - 1) & 0 \\ 0 & 0 & (2q_0^2 - 1) \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix} \\ 2(\mathbf{q} \cdot \mathbf{v})\mathbf{q} &= \begin{bmatrix} 2q_1^2 & 2q_1q_2 & 2q_1q_3 \\ 2q_1q_2 & 2q_2^2 & 2q_2q_3 \\ 2q_1q_3 & 2q_2q_3 & 2q_3^2 \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix} \\ 2q_0(\mathbf{q} \times \mathbf{v}) &= \begin{bmatrix} 0 & 2q_0q_3 & -2q_0q_2 \\ -2q_0q_3 & 0 & 2q_0q_1 \\ 2q_0q_2 & -2q_0q_1 & 0 \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix} \end{aligned}$$

Por lo que el operador rotacional $L\mathbf{v} = \mathbf{q} \otimes \mathbf{v} \otimes \mathbf{q}_*$ queda simplificado de la siguiente forma:

$$\mathbf{v}' = \mathbf{q} \mathbf{v} \tag{4.12}$$

$$\begin{bmatrix} v'_1 \\ v'_2 \\ v'_3 \end{bmatrix} = \begin{bmatrix} 2q_0^2 - 1 + 2q_1^2 & 2q_1q_2 - 2q_0q_3 & 2q_1q_3 + 2q_0q_2 \\ 2q_1q_2 + 2q_0q_3 & 2q_0^2 - 1 + 2q_2^2 & 2q_2q_3 - 2q_0q_1 \\ 2q_1q_3 - 2q_0q_2 & 2q_2q_3 + 2q_0q_1 & 2q_0^2 - 1 + 2q_3^2 \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix}$$

Esta operación es muy útil en la modelación de la cinemática cuando se relaciona un sistema de coordenadas de referencia a partir de otro, relacionando mediante rotaciones la base que forma cada uno de estos sistemas.

Para analizar como rota un sistema de coordenadas a partir de un sistema de referencia

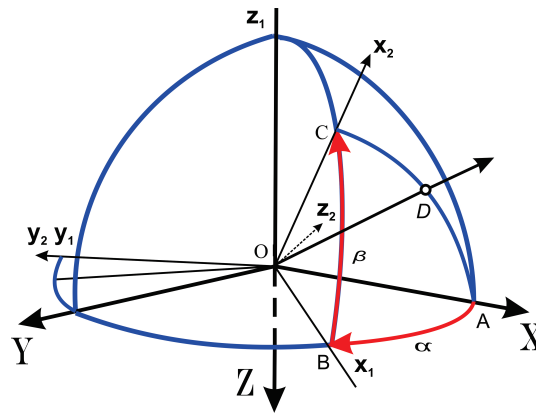


FIGURA 4.1: Sistema de rotaciones de un sistema de referencia [5].

(ver Figura 4.1), se emplean dos rotaciones sucesivas sobre ejes, los cuales entre ellos son mutuamente ortogonales. Recalcando que estas dos rotaciones son una secuencia ordenada tomando el primer eje de rotación sobre el eje Z a través de un ángulo α seguida de una rotación tomando como eje de rotación la nueva posición de $Y = Y_1$ a través de un ángulo β así los cuaterniones que definen estas rotaciones, de acuerdo con la Ecuación 3.3, son los siguientes:

$$q = \cos \frac{\alpha}{2} + \mathbf{k} \operatorname{sen} \frac{\alpha}{2}$$

$$p = \cos \frac{\beta}{2} + \mathbf{j} \operatorname{sen} \frac{\beta}{2}$$

lo que de manera simplificada queda como el producto de estos dos cuaterniones quedando de la siguiente forma:

$$\begin{aligned}
 \mathbf{r} &= \mathbf{q} \otimes \mathbf{p} \\
 &= \left(\cos \frac{\alpha}{2} + \mathbf{k} \sin \frac{\alpha}{2} \right) \left(\cos \frac{\beta}{2} + \mathbf{j} \sin \frac{\beta}{2} \right) \\
 &= \cos \frac{\alpha}{2} \cos \frac{\beta}{2} + \mathbf{j} \cos \frac{\alpha}{2} \sin \frac{\beta}{2} \\
 &\quad + \mathbf{k} \sin \frac{\alpha}{2} \cos \frac{\beta}{2} + (\mathbf{k} \times \mathbf{j}) \sin \frac{\alpha}{2} \sin \frac{\beta}{2} \\
 &= \cos \frac{\alpha}{2} \cos \frac{\beta}{2} - \mathbf{i} \sin \frac{\alpha}{2} \sin \frac{\beta}{2} \\
 &\quad \mathbf{j} \cos \frac{\alpha}{2} \sin \frac{\beta}{2} + \mathbf{k} \sin \frac{\alpha}{2} \cos \frac{\beta}{2}
 \end{aligned}$$

Por lo que el cuaternión de rotación resultante esta compuesto por una parte vectorial:

$$\begin{aligned}
 \mathbf{r}_v &= -\mathbf{i} \sin \frac{\alpha}{2} \sin \frac{\beta}{2} \\
 &\quad \mathbf{j} \cos \frac{\alpha}{2} \sin \frac{\beta}{2} \\
 &\quad + \mathbf{k} \sin \frac{\alpha}{2} \cos \frac{\beta}{2}
 \end{aligned}$$

y un ángulo de rotación:

$$\cos \frac{\phi}{2} = \cos \frac{\alpha}{2} \cos \frac{\beta}{2}$$

De esta forma utilizamos el álgebra de cuaterniones para realizar rotaciones sucesivas y el operador rotacional para representar rotaciones de un cuerpo rígido representado por un vector en \mathbb{R}^3 , ambos empleados en el modelado algunos sistemas cinemáticos de robots.

4.2. Robot ABB-IRB1600

Como se definió en el Capítulo 2 Sección 2.2, un robot, es diseñado, principalmente para dar solución al menos a un problema o cubrir una necesidad en específico. De manera específica, el IRB1600, es empleado de forma industrial en diversas operaciones distintas: montaje, asistencia para el mecanizado, empaquetado, manipulación de materiales,

soldadura, manipulación de materiales, moldeo por inyección [25], Estas aplicaciones determinan en gran parte la estructura y el diseño del robot, lo cual, incluye su estructura cinemática [26]

4.2.1. Estructura paramétrica del IRB1600

El robot industrial ABB-IRB1600 es considerado un robot de alta tecnología ya que cuenta con 6 GDL, a los que generalmente se les clasifica cinemáticamente sobre la base del primer brazo, es decir las tres primeras juntas, describiendo por separado el efector final. El IRB1600 se clasifica, por su geometría, como articulado ó antropomórfico (ver Figura 4.2). Con esta configuración proporciona, relativamente, una amplia libertad de movimiento en un espacio altamente reducido.

Las principales características del robot IRB1600 se pueden encontrar en sus hojas de datos. Este robot puede modelarse como una cadena cinemática abierta con elementos rígidos unidos en serie por articulaciones de revolución. Un extremo de la cadena es fijo a una base global de referencia (inercial) y el otro extremo está libre y unido a una herramienta para ejecutar aplicaciones específicas. La cinemática del robot (ver Sec-

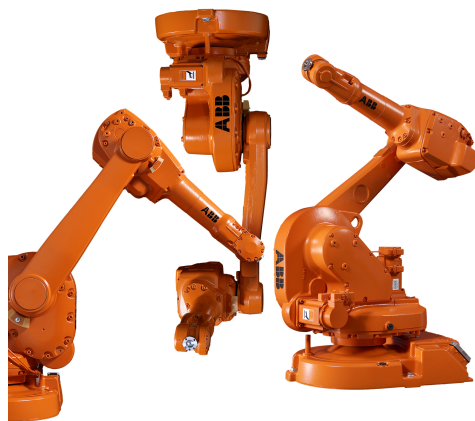


FIGURA 4.2: Robot IRB1600 de ABB [27].

ción 2.3), trata con el estudio analítico de su movimiento con respecto a una referencia global como una función del tiempo. EL robot IRB1600 constituye un buen ejemplo de robot articulado de seis grados de libertad que permite demostrar la validez de la metodología del modelado vía cuaterniones.

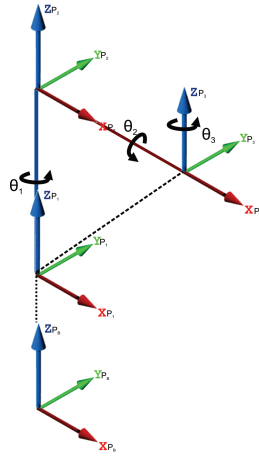


FIGURA 4.3: Sistemas de referencia de los robots.

4.2.2. Modelo matemático del IRB1600

En el desarrollo de esta sección se mostrará cómo determinar las ecuaciones del modelo rotacional que determinan la cinemática del robot IRB1600, mediante la aplicación del operador cuaterniónico de rotación L_v . Este método es la base para obtener la cinemática directa de cualquier robot rotacional de n grados de libertad.

Antes de determinar las ecuaciones que describen el modelo rotacional hay que tomar en cuenta las siguientes consideraciones: a cada junta se fija una referencia móvil. Esto quiere decir que para cualquier movimiento que realice la articulación del robot los vectores coordenados de cada punto sobre el eslabón $(1, 2, \dots, n)$ estarán constantemente expresados en referencia del móvil $\{V_1\}, \{V_2\}, \dots, \{V_n\}$, respectivamente. Para describir las iteraciones de los eslabones adyacentes se hace coincidir un eje de referencia móvil a cada eslabón del robot, como se muestra en la Figura 4.3, el inicio de cada eslabón se asocia de forma sistemática a la referencia global (X_{pb}, Y_{pb}, Z_{pb}) , que es la base del robot, hasta llegar al efector final.

Para comenzar con la obtención de las ecuaciones del modelo rotacional se asigna un sistema global de referencia, y “ n ” sistemas de referencia móviles para cada rotación que efectúe el robot, como se muestra en la Figura 4.4, de esta forma se construyen vectores en R^3 , los cuales son de forma representativa cada uno de los eslabones del robot. Con el arreglo de referencias mostrado en la Figura 4.4, el vector (V_1) formado

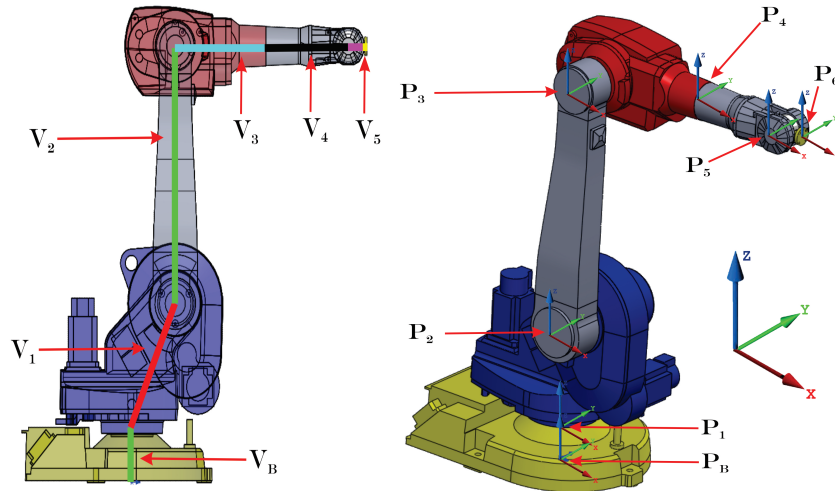


FIGURA 4.4: Sistemas de referencia del robot.

por los puntos P_1 y P_2 , es colineal al vector formado por los puntos P_2 y P_3 , es decir V_2 los cuales son perpendiculares a los siguientes, con esto se establecen las restricciones para proceder de la siguiente forma:

Siendo los seis cuaterniones de rotación q , r , s , t , u y w unidos a cada junta rotacional, que definen la orientación de cada referencia móvil,

$$P_1 = P_{1x}, P_{1y}, P_{1z}$$

$$P_2 = P_{2x}, P_{2y}, P_{2z}$$

$$P_3 = P_{3x}, P_{3y}, P_{3z}$$

$$P_4 = P_{4x}, P_{4y}, P_{4z}$$

$$P_5 = P_{5x}, P_{5y}, P_{5z}$$

$$P_6 = P_{6x}, P_{6y}, P_{6z}$$

y seis vectores $V_b(t)$, $V_1(t)$, $V_2(t)$, $V_3(t)$, $V_4(t)$ y $V_5(t)$ que definen la posición del robot, con respecto a la referencia global X, Y, Z , definida como P_{Bx}, P_{By}, P_{Bz} , como se muestra en Figura 4.5 De esta relación se encuentra un vector resultante $V_r(t)$, el cual, define la relación entre el sistema de referencia global y el sistema de referencia del ultimo eslabón del robot, que es la relación de posición del efector final con respecto al tiempo (ver Figura 4.5). De esta forma la relación que existe entre la base y el efector final se

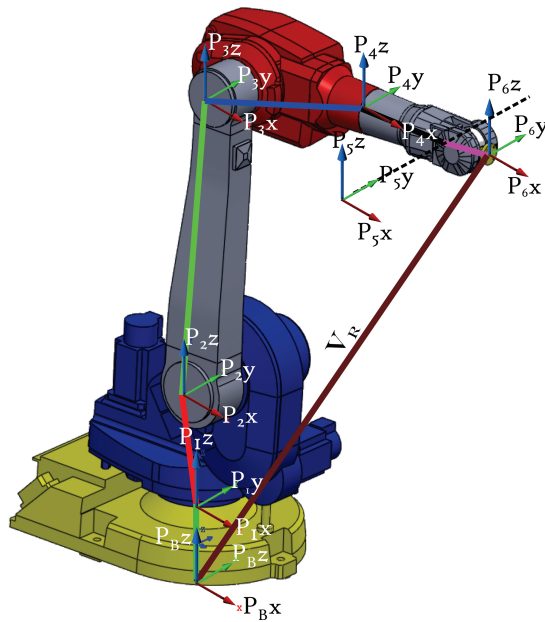


FIGURA 4.5: Relaciones de los sistemas de referencia.

expresa de la siguiente forma:

$$\mathbf{V}_r(t) = \mathbf{V}_1(t) + \mathbf{V}_1(t) + \dots + \mathbf{V}_n(t) \quad (4.13)$$

Por lo que para el IRB1600, la posición del efector final se expresa como:

$$\mathbf{V}_r(t) = \mathbf{V}_B(t) + \mathbf{V}_1(t) + \mathbf{V}_2(t) + \mathbf{V}_3(t) + \mathbf{V}_4(t) + \mathbf{V}_5(t) \quad (4.14)$$

La magnitud del vector \mathbf{V}_b es relacionada a la distancia que existe entre el punto de referencia P_b y P_1 , la orientación se encuentra fija debido a que este eslabón no es coincidente con ninguna base móvil del robot.

Para obtener la primer rotación, se tiene que el punto P_1 se convierte en el sistema de referencia móvil del segundo vector \mathbf{V}_1 , así P_1 es ahora el nuevo origen del vector, el cual, se encuentra formado por el desplazamiento que existe entre el punto P_1 y P_2 , para obtener la primera rotación de la base móvil, se obtiene de la siguiente forma:

$$\mathbf{V}_1(t) = L v_1 = Q \mathbf{V}_1 \quad (4.15)$$

siendo Ecuación 4.15 la rotación de referencia de P_2 , donde el cuaternión de rotación q se define como:

$$Q = q = \cos \frac{\theta_1}{2} + \mathbf{k} \text{sen} \frac{\theta_1}{2} \quad (4.16)$$

ya que la rotación del primer eslabón se realiza sobre el eje rotacional Z y θ_1 el ángulo de rotación, como se muestra en la Figura 4.6 y el vector \mathbf{V}_1 como ya se menciona es la longitud que existe entre P_1 y P_2 quedando de la siguiente forma:

$$\mathbf{V}_1 = P_2 - P_1$$

y la posición del primer eslabón se asocia de la siguiente manera:

$$\mathbf{V}_{r_1} = \mathbf{V}_B + \mathbf{V}_1(t)$$

Para la posición del eslabón dos asociamos los valores del vector \mathbf{V}_2 , el cual define la

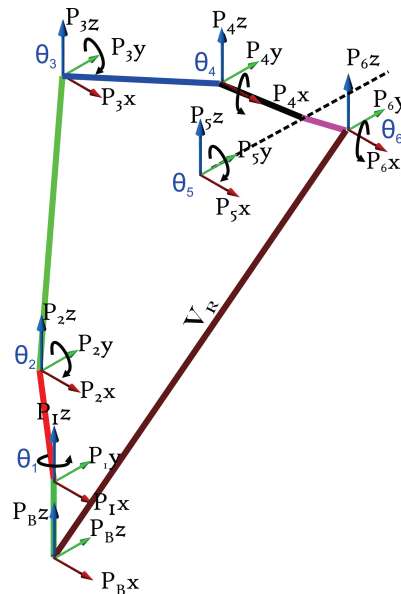


FIGURA 4.6: Relación de ejes rotacionales y ángulos de giro.

posición y la orientación del eje de la base móvil, que esta definida por el cuaternión de rotación R . con respecto al sistema de referencia global del robot de la siguiente forma: De acuerdo con Sección 4.1.3 para aplicar el operador rotacional primero tenemos que aplicar un producto entre cuaterniones, con el fin de asociar una cadena de rotaciones precedentes, esto forma una secuencia de rotaciones con las cuales se obtienen las ecuaciones de la cinemática del robot, así, el cuaternión de rotación R está formado por la

multiplicación del cuaternión Q con el cuaternión r .

$$\begin{aligned}
 R &= Q \otimes r \\
 &= q \otimes r \\
 &= \left(\cos \frac{\theta_1}{2} + \mathbf{k} \operatorname{sen} \frac{\theta_1}{2} \right) \otimes \left(\cos \frac{\theta_2}{2} + \mathbf{j} \operatorname{sen} \frac{\theta_2}{2} \right)
 \end{aligned} \tag{4.17}$$

como se observa $r = \cos \frac{\theta_2}{2} + \mathbf{j} \operatorname{sen} \frac{\theta_2}{2}$ ya que la rotación del segundo eslabón se realiza sobre el eje rotacional Y y un ángulo de rotación θ_2 , por lo que así se define la orientación del segundo eslabón.

Ahora el operador rotacional se aplica para encontrar la posición de este eslabón, así tenemos lo siguiente:

$$\mathbf{V}_2(t) = L\mathbf{v}_2 = R\mathbf{V}_2 \tag{4.18}$$

Donde, \mathbf{V}_2 es la diferencia que existe entre el punto P_2 y P_3 . Para conocer la posición del segundo eslabón se suma la posición de los vectores anteriores rotados así como el de la base, es decir:

$$\mathbf{V}_{r_2} = \mathbf{V}_{r_1} + \mathbf{V}_2(t)$$

como se muestra en la Ecuación 4.14, de este modo se asocia la posición del segundo eslabón con respecto a la referencia global del robot.

Ahora para obtener la orientación del tercer eslabón, se asocia el quaternion de rotación s y se procede de forma sistemática para obtener al cuaternión que define la orientación S , como en el caso anterior, es decir:

$$\begin{aligned}
 S &= R \otimes s \\
 &= q \otimes r \otimes s \\
 &= \left(\cos \frac{\theta_1}{2} + \mathbf{k} \operatorname{sen} \frac{\theta_1}{2} \right) \otimes \left(\cos \frac{\theta_2}{2} + \mathbf{j} \operatorname{sen} \frac{\theta_2}{2} \right) \otimes \left(\cos \frac{\theta_3}{2} + \mathbf{j} \operatorname{sen} \frac{\theta_3}{2} \right)
 \end{aligned} \tag{4.19}$$

donde el cuaternión asociado a la tercer articulación

$$s = \cos \frac{\theta_3}{2} + \mathbf{j} \operatorname{sen} \frac{\theta_3}{2}$$

como se puede observar en la Ecuación 4.19, la rotación se efectúa sobre el eje rotacional Y y ángulo de rotación θ_3 . La posición asociada al tercer eslabón es definida como:

$$\mathbf{V}_3(t) = L_{v_3} = S\mathbf{V}_3 \quad (4.20)$$

así la posición con respecto a la referencia global, es:

$$\mathbf{V}_{r_3} = \mathbf{V}_{r_2} + \mathbf{V}_3(t)$$

La posición y orientación del cuarto eslabón con respecto a la referencia global se tiene:

La rotación de la base móvil es:

$$\begin{aligned} T &= S \otimes t \\ &= \mathbf{q} \otimes \mathbf{r} \otimes \mathbf{s} \otimes t \\ &= \left(\cos \frac{\theta_1}{2} + \mathbf{k} \operatorname{sen} \frac{\theta_1}{2} \right) \otimes \left(\cos \frac{\theta_2}{2} + \mathbf{j} \operatorname{sen} \frac{\theta_2}{2} \right) \otimes \dots \\ &\quad \dots \otimes \left(\cos \frac{\theta_3}{2} + \mathbf{j} \operatorname{sen} \frac{\theta_3}{2} \right) \otimes \left(\cos \frac{\theta_4}{2} + \mathbf{i} \operatorname{sen} \frac{\theta_4}{2} \right) \end{aligned} \quad (4.21)$$

Los elementos del cuaternión asociados a esta rotación son:

$$t = \cos \frac{\theta_4}{2} + \mathbf{i} \operatorname{sen} \frac{\theta_4}{2}$$

de acuerdo con la Ecuación 4.19 es posible observar que la rotación se realiza sobre el eje rotacional X y un ángulo θ_4 , por lo que la posición asociada al cuarto eslabón es:

$$\mathbf{V}_4(t) = L_{v_4} = T\mathbf{V}_4 \quad (4.22)$$

por lo tanto al aplicar de manera sistemática lo anterior se tiene que la posición con respecto a la referencia global, es:

$$\mathbf{V}_{r_4} = \mathbf{V}_{r_3} + \mathbf{V}_4(t)$$

Para determinar la orientación y posición del quinto eslabón se procede, como ya se menciono, de forma sistemática a obtener las relaciones siguientes: Para la rotación de

la base móvil:

$$\begin{aligned}
\mathbf{U} &= \mathbf{S} \otimes \mathbf{u} \\
&= \mathbf{q} \otimes \mathbf{r} \otimes \mathbf{s} \otimes \mathbf{t} \otimes \mathbf{u} \\
&= \left(\cos \frac{\theta_1}{2} + \mathbf{k} \operatorname{sen} \frac{\theta_1}{2} \right) \otimes \left(\cos \frac{\theta_2}{2} + \mathbf{j} \operatorname{sen} \frac{\theta_2}{2} \right) \otimes \dots \\
&\quad \dots \otimes \left(\cos \frac{\theta_3}{2} + \mathbf{j} \operatorname{sen} \frac{\theta_3}{2} \right) \otimes \left(\cos \frac{\theta_4}{2} + \mathbf{i} \operatorname{sen} \frac{\theta_4}{2} \right) \otimes \dots \\
&\quad \dots \otimes \left(\cos \frac{\theta_5}{2} + \mathbf{j} \operatorname{sen} \frac{\theta_5}{2} \right)
\end{aligned} \tag{4.23}$$

Los elementos asociados al cuaternión de rotación \mathbf{u} son:

$$\mathbf{u} = \cos \frac{\theta_5}{2} + \mathbf{j} \operatorname{sen} \frac{\theta_5}{2}$$

y es posible verificar de forma que el eje de rotación para el cuaternión \mathbf{u} es \mathbf{Y} y un ángulo θ_5 . La posición es asociada a:

$$\mathbf{V}_5(t) = \mathbf{L}_{\mathbf{V}_5} = \mathbf{T}\mathbf{V}_5 \tag{4.24}$$

por lo tanto la posición con respecto a la referencia global, es:

$$\mathbf{V}_{r_5} = \mathbf{V}_{r_4} + \mathbf{V}_5(t)$$

Para el sexto eslabón realizamos el mismo procedimiento por lo que se asocia el ultimo cuaternión de giro \mathbf{w} y se procede a determinar su orientación con respecto al sistema global de referencia, por lo que para la rotación de la base móvil se tiene:

$$\begin{aligned}
\mathbf{W} &= \mathbf{U} \otimes \mathbf{w} \\
&= \mathbf{q} \otimes \mathbf{r} \otimes \mathbf{s} \otimes \mathbf{t} \otimes \mathbf{u} \otimes \mathbf{w} \\
&= \left(\cos \frac{\theta_1}{2} + \mathbf{k} \operatorname{sen} \frac{\theta_1}{2} \right) \otimes \left(\cos \frac{\theta_2}{2} + \mathbf{j} \operatorname{sen} \frac{\theta_2}{2} \right) \otimes \dots \\
&\quad \dots \otimes \left(\cos \frac{\theta_3}{2} + \mathbf{j} \operatorname{sen} \frac{\theta_3}{2} \right) \otimes \left(\cos \frac{\theta_4}{2} + \mathbf{i} \operatorname{sen} \frac{\theta_4}{2} \right) \otimes \dots \\
&\quad \dots \otimes \left(\cos \frac{\theta_5}{2} + \mathbf{j} \operatorname{sen} \frac{\theta_5}{2} \right) \otimes \left(\cos \frac{\theta_6}{2} + \mathbf{i} \operatorname{sen} \frac{\theta_6}{2} \right)
\end{aligned} \tag{4.25}$$

Los elementos asociados al cuaternión de rotación w son:

$$w = \cos \frac{\theta_6}{2} + \mathbf{i} \operatorname{sen} \frac{\theta_6}{2}$$

es posible verificar que el eje de rotación para el cuaternión w es X y un ángulo θ_6 , el cual representa la orientación del efector final del robot.

Ahora si sustituimos los valores obtenidos de las posiciones parciales en la Ecuación 4.14,

$$\mathbf{V}_r(t) = \mathbf{V}_B(t) + \mathbf{V}_1(t) + \mathbf{V}_2(t) + \mathbf{V}_3(t) + \mathbf{V}_4(t) + \mathbf{V}_5(t)$$

se obtiene la posición con respecto a la referencia global fija.

4.2.3. Solución de la cinemática directa

Para dar solución al problema cinemático directo se emplean las Ecuaciones 4.14 y 4.25, las cuales definen la orientación y posición del efector final y es posible expresarlas de forma simbólica como una función de los desplazamientos de angulares de las juntas rotacionales, para realizar esto utilizamos la ayuda del MatLab[®].

Las ecuaciones que representan la cinemática directa se pueden iniciar expresando las ecuaciones de posición y de orientación en función de los parámetros de los cuaterniones q , r , s , t , u y w , asociados a las juntas rotacionales para lo que se utiliza el algoritmo programado en MatLab[®], que se puede observar en el Apéndice B.

Se puede observar que, se expresan las distancias entre cada junta del robot, y los cuaterniones asociados a cada base móvil. Para este análisis se toman los parámetro que provee la hoja de datos del robot IRB1600, las cuales pueden ser comprobadas en el modelo 3D también provee la empresa ABB [28].

En el código del programa se puede observar, que el producto entre cuaterniones que representan las rotaciones de las bases definido en la Ecuación 4.2 se expresa como

Qpro y las rotaciones de cada punto expresadas como el operador rotacional definido en la Ecuación 4.12 se expresan como Q_{rot} y de una forma compacta podemos observar que cada una de las rotaciones corresponden a las Ecuaciones 4.15 ... 4.25, llegando a la determinación del punto final de la cadena cinemática, teniendo su orientación y posición del efector final.

4.2.3.1. Comparación del modelo basado en cuaterniones y el modelo basado en la convención Denavit-Hartenberg

De acuerdo con [29] el modelo para la cinemática directa basado en la convención Denavit-Hartenberg (ver Sección 2.3.1) define cada uno de los sistemas de coordenadas en cada articulación como se muestra en la Figura 4.7,

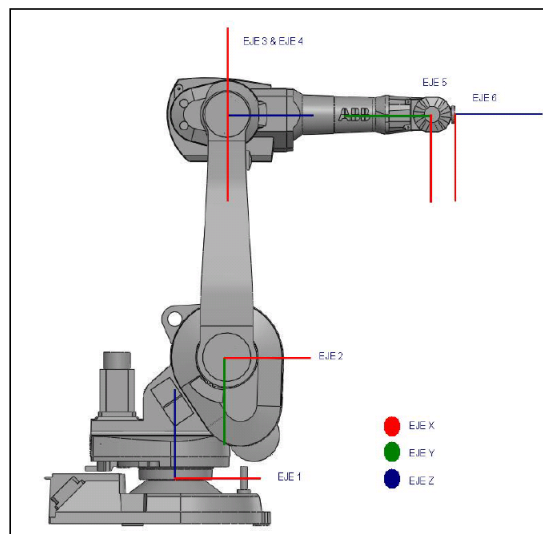


FIGURA 4.7: Definición de los sistemas de coordenadas en cada articulación [29].

por lo que al relacionar los parámetros de cada una de las articulaciones se obtienen la Tabla 4.1.

Este modelo puede ser comparado con el modelo basado en cuaterniones, el cual se simplifica en la Tabla 4.2, con lo que es posible observar que el modelo basado en cuaterniones es significativamente más compacto, lo cual reduce el número de operaciones

TABLA 4.1: Representación Denavit-Hartenberg del robot IRB1600 [29]

Articulación	θ_i	d_i	a_i	α_i
1	θ_1	d_1	a_1	$-\frac{\pi}{2}$
2	θ_2	0	a_2	0
3	θ_3	0	0	$\frac{\pi}{2}$
4	θ_4	d_4	0	$-\frac{\pi}{2}$
5	θ_5	0	0	$\frac{\pi}{2}$
6	θ_6	d_6	0	0

matemáticas de forma significativa (ver Tabla 2.3). Al programar ambos modelos en

TABLA 4.2: Representación del modelo rotacional basado en cuaterniones del robot IRB1600.

Cuaternión de rotación	Vector a ser rotado
$Q = \begin{bmatrix} \cos \frac{\theta_1}{2} & 0 & 0 & \text{sen} \frac{\theta_1}{2} \end{bmatrix}$	V_1
$R = \begin{bmatrix} \cos \frac{\theta_2}{2} & \text{sen} \frac{\theta_2}{2} & 0 & 0 \end{bmatrix}$	V_2
$S = \begin{bmatrix} \cos \frac{\theta_3}{2} & \text{sen} \frac{\theta_3}{2} & 0 & 0 \end{bmatrix}$	V_3
$T = \begin{bmatrix} \cos \frac{\theta_4}{2} & 0 & \text{sen} \frac{\theta_4}{2} & 0 \end{bmatrix}$	V_4
$U = \begin{bmatrix} \cos \frac{\theta_5}{2} & \text{sen} \frac{\theta_5}{2} & 0 & 0 \end{bmatrix}$	V_5
$R = \begin{bmatrix} \cos \frac{\theta_6}{2} & 0 & \text{sen} \frac{\theta_6}{2} & 0 \end{bmatrix}$	V_6

MatLab[®] es posible observar con mayor detalle la reducción de recursos computacionales usados cuando se utiliza el modelo rotacional basado en cuaterniones comparado con el modelo convencional basado en la convención Denavit-Hartenberg. Mediante MatLab[®] se puede obtener un informe de los recursos computacionales usados para cada uno de los modelos el cual es posible observar en el Apéndice E.

4.2.4. Solución cinemática inversa

De las ecuaciones obtenidas del modelo rotacional de cada una de las juntas móviles y el vector de posición $\mathbf{V}_r(t)$, se obtiene una relación entre el movimiento y los desplazamientos angulares de las uniones o juntas, las cuales proporcionan, la orientación y posición resultante del efector final. Al sustituir los valores de los desplazamientos angulares en las ecuaciones ya mencionadas, se puede determinar de forma directa la posición y orientación del efector final, es decir, se soluciona el problema cinemático

directo.

El problema de la cinemática inversa resulta de situar el elemento terminal en una posición y orientación predeterminada. Las ecuaciones que se desprenden de la posición ($\mathbf{V}_r(t) = [V_{rx}, V_{ry}, V_{rz}]$) y las ecuaciones que se obtienen de la orientación (orientación = $W = q \otimes r \otimes s \otimes t \otimes u \otimes w$) deben resolverse para obtener los cuaterniones que determinan el desplazamiento angular de cada una de las juntas, dada la posición y orientación del efector final, una vez resuelta la ecuación cinemática, se podrá realizar el movimiento previsto de la herramienta sustituyendo cada uno de los valores obtenidos mediante el método de solución (numérico, geométrico, simbólico, etc.).

En el problema cinemático directo se obtiene una solución única para un determinado conjunto de cuaterniones en las juntas. El problema de la cinemática inversa es más complejo en el sentido de que pueden existir un conjunto de soluciones para una misma posición de la herramienta. Igualmente, para una determinada estructura de brazos y sobre un rango de posiciones y orientaciones de la herramienta, puede no existir solución para el problema cinemático inverso[26].

Para resolver el problema cinemático del IRB1600 y comprobar el funcionamiento de modelo rotacional descrito en la Sección 4.2.2 se hará uso del métodos geométricos y el desacoplo cinemático se hará el análisis de este problema. El procedimiento en sí se basa en encontrar el suficiente numero de relaciones geométricas en las que intervendrán las coordenadas del extremo del robot, sus coordenadas articulares y las dimensiones físicas de sus elementos.

Mediante el siguiente procedimiento se hallarán las ecuaciones correspondientes para el cálculo de las variables que determinan la posición de las tres primeras articulaciones del robot, teniendo en cuenta el punto que deberá alcanzar el extremo final del robot (V_{rx}, V_{ry}, V_{rz}), teniendo en cuenta que como ya se menciona la orientación en la orientación de la muñeca debe ser conocida ya que existen mas de una configuración posible en las que se puede posicionar el efector final y es una de las condicionantes para el

desarrollo de este método.

Para poder encontrar las relaciones geométricas que describan los movimientos articulares de las tres primeras articulaciones, como ya se mencionó, debemos conocer la orientación deseada que tendrá la herramienta, al conocer estos parámetros podemos hacer algunas suposiciones con lo que daremos solución a este problema.

Primero obtenemos los valores de la orientación deseada y realizamos los movimientos que refieren a la muñeca es decir el cuarto y quinto eslabón de la cadena cinemática, como se puede observar en la Figura 4.8, de la cual se desprenden las ecuaciones:

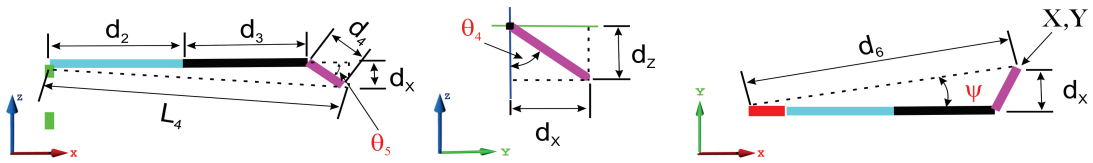


FIGURA 4.8: Relación de la orientación de la muñeca del robot.

$$d_4 = d_7 \cos \theta_5 \quad (4.26)$$

$$d_z = d_7 \sin \theta_5 \sin(270 + \theta_4) \quad (4.27)$$

$$d_x = d_7 \sin \theta_5 \cos(270 + \theta_4) \quad (4.28)$$

$$d_6 = \sqrt{x^2 + y^2} \quad (4.29)$$

$$\phi = \operatorname{atan} \left(\frac{d_z}{d_6} \right) \quad (4.30)$$

$$\psi = \operatorname{atan} \left(\frac{d_x}{d_5} \right) \quad (4.31)$$

$$\vartheta = \operatorname{atan} \left(\frac{y}{x} \right) \quad (4.32)$$

$$d_5 = d_2 + d_3 + d_4 \quad (4.33)$$

$$L_4 = \frac{d_5}{\cos \phi} \quad (4.34)$$

donde y, x son los puntos que describen la posición del efector final del robot, estas variables son calculadas en la posición inicial del robot.

Una vez que se han determinado las variables que describen los desplazamientos que debe adoptar el robot para tomar una orientación específica es posible calcular los desplazamientos angulares que necesita llevar a cabo para posicionarse en las coordenadas

en un punto específico.

Para la primer articulación tenemos que de acuerdo a la Figura 4.9, el robot posee una estructura plana, quedando este ángulo definido por la primera variable articular θ_1 .

Para obtener el valor de θ_1 se realiza la siguiente relación:

$$\theta_1 = \vartheta - \psi \quad (4.35)$$

Cabe mencionar que respecto a las limitaciones físicas del robot la base no puede tener un desplazamiento en un ángulo mayor a los $\pm 180^\circ$, por lo que el espacio de trabajo se limita a los puntos alcanzable para esos desplazamientos. Para las segunda y tercer

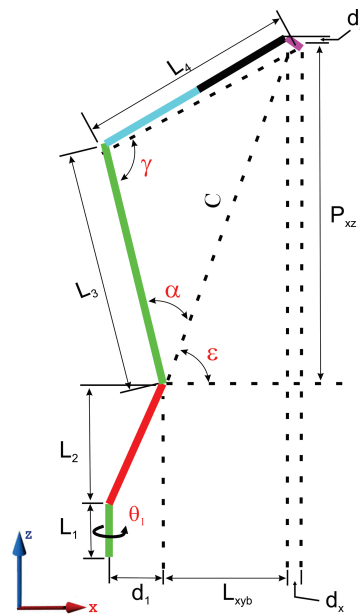


FIGURA 4.9: Relación de la posición de la base del robot.

articulaciones tenemos se realiza un estudio más detallado de la geometría del robot.

Para determinar el ángulo ϵ y la distancia L_{xyb} se tienen las siguientes relaciones:

$$L_{xyb} = \sqrt{x^2 + y^2 - d_x^2} - d_1 \quad (4.36)$$

$$\epsilon = \text{atan} \left(\frac{z - (L_1 + L_2)}{L_{xyb}} \right) \quad (4.37)$$

como se puede observar en la Figura 4.9 se forma un triángulo mediante los segmentos L_3, L_4 y C del cual podemos determinar los ángulos α y γ mediante la relación que existe en sus longitudes utilizando la "fórmula de Herón" [30], las siguientes relaciones

son dadas:

$$C = \sqrt{(z - (L_1 + L_2))^2 + L_{xyb}^2} \quad (4.38)$$

$$s = \frac{L_3 + L_4 + C}{2} \quad (4.39)$$

$$r = \sqrt{\frac{(s - L_3)(s - L_4)(s - C)}{s}} \quad (4.40)$$

Las distancias L_1, L_2 y L_3 son dadas por los parámetros del robot ver ANEXO DE DATOS DEL ROBOT, al calcular las variables que representan el triángulo es posible determinar los ángulos internos de éste, teniendo con esto las relaciones angulares para los desplazamiento de la segunda y tercera articulaciones mediante:

$$\alpha = 2\text{atan}\left(\frac{r}{(s - L_4)}\right) \quad (4.41)$$

$$\gamma = 2\text{atan}\left(\frac{r}{(s - C)}\right) \quad (4.42)$$

de esta forma quedan definidos los ángulos de desplazamiento θ_2 y θ_3 como:

$$\theta_2 = 90 - (\alpha + \varepsilon) \quad (4.43)$$

$$\theta_3 = 90 - \gamma + \phi \quad (4.44)$$

ahora bien, si se toma en cuenta que existen diferentes configuraciones que puede tomar el robot y que mediante este método se toma en cuenta la geometría del manipulador existe otra solución dada por la naturaleza propia de la geometría, a estas dos configuraciones se les conoce como “*codo arriba*” y “*codo abajo*” descritos en la Sección 2.3.1, para la cual se tiene el siguiente conjunto de ecuaciones para la configuración codo abajo:

$$\theta_2 = 90 - (\alpha - \varepsilon) \quad (4.45)$$

$$\theta_3 = -270 + \gamma + \phi \quad (4.46)$$

De este modo al calcular los desplazamientos angulares de la base que representa la posición y dada una orientación es posible calcular los cuaterniones rotacionales y aplicarlo al modelo rotacional desarrollado en la Sección 4.2.2.

Capítulo 5

Simulación VRLM e interfaz gráfica.

En este capítulo se llevara acabo la simulación de las ecuaciones obtenidas en la sección Sección 4.2.2. Para la simulación del movimiento del manipulador, gobernado por las ecuaciones del modelado vía cuaterniones, se desarrolló una interface visual que aprovecha las características del modelo en CAD realizado en el lenguaje VRML (Virtual Reality Modeling Language)[28], así como la solución de las ecuaciones por medio de MatLab[®].

5.1. Modelado en realidad virtual

El Lenguaje para modelado de realidad virtual VRML (por sus siglas en inglés), es un formato de de archivo normalizado, el cuál, tiene por objetivo la representación de escenas u objetos representados de forma tridimensional diseñados principalmente para su empleo en la WEB[31].

El termino VRML aprecio por primera ves en 1944 por Tim Berners-Lee en una conferencia web, donde habló de la necesidad de estandarizar los modelos virtuales 3D en la web. De esta conferencia nació la necesidad de desarrollar un lenguaje común para la descripción de los mundos 3D.

De este modo, en la Primera Conferencia Mundial de la *www* en Ginebra se aprobó el desarrollo de un nuevo lenguaje que permitiese crear mundos en tres dimensiones a los que se pudiera acceder por la *www*.

Con el tiempo se desarrolló el lenguaje dentro de varios requisitos: que fuese adaptable a la red, que no requiriese una línea de alta velocidad (anchos de banda elevados), que fuese multiplataforma, etc[31].

Mediante el uso del lenguaje VRML es posible la descripción de escenas compuestas por objetos 3D, que pueden estar basados en formas geométricas básicas o de estructuras complejas compuestas por vértices y aristas que forman un polígono tridimensional con un color en la superficie. VRML permite también definir objetos 3D multimedia a los cuales se puede asociar un enlace de manera que el usuario pueda acceder a una página web, imágenes, videos u otro archivo VRML de internet cada vez que haga click en el componente gráfico en cuestión.

Este modo de visitar sitios en internet es mucho más avanzado y posee grandes ven-



FIGURA 5.1: Ejemplo de realidad virtual.

tajas. Para empezar la navegación se desarrolla de una manera mucho más intuitiva, dado que la forma de actuar dentro del mundo virtual es similar a la de la vida real. Podemos movernos en todas las direcciones, no sólo izquierda y derecha sino también adelante, atrás, arriba y abajo. Podemos tratar con los objetos como en la vida misma,

tocarlos, arrastrarlos, etc. También los escenarios son mucho más reales, pensemos por ejemplo en una biblioteca virtual, en ella podríamos andar por cada una de las salas, tomar determinados libros y leerlos.

Cada vez los accesos a internet se han convertido en experiencias más reales. Sin embargo VRML presenta grandes limitaciones en sus potencialidad de desarrollo, debido a la necesidad de procesar gran cantidad de datos, esto se ira cubriendo con la llegada de nuevos periféricos, como cascos o guantes, que ayuden en la manipulación de estos entornos.

5.1.1. Desarrollo del modelo en realidad virtual

Como se vio en la sección anterior “Virtual Reality Modeling Language” (VRML), ayuda en la creación de modelos virtuales en 3D, de cualquier objeto, mundo, entorno, o elemento que se requiera.

Las ventajas que se tiene al desarrollar el modelo virtual del robot IRB1600 son:

- ▷ Software de acceso libre
- ▷ Comandos de desarrollo partir de comando de texto
- ▷ Interfaz numérico
- ▷ Compatibilidad con MatLab[®] y Simulink
- ▷ Creación de cualquier geometría en 2D y 3D

La programación de objetos en VRML se logra por medio de nodos y campos en los cuales se especifican las características visuales y dimensionales de dicho objeto, siendo estas ultimas programadas en coordenadas cartesianas que unen los puntos de cada nodo, para la obtención del modelo virtual del robot IRB1600 se tomaron los parámetros proporcionados por ABB[28].

Cada una de las piezas es creada por separado, comenzando con la base (ver Figura 5.2), desarrollando cada uno de los eslabones hasta llegar a obtener el cuerpo del robot. Posteriormente cada una de estas piezas son anidadas una tras otra hasta formar

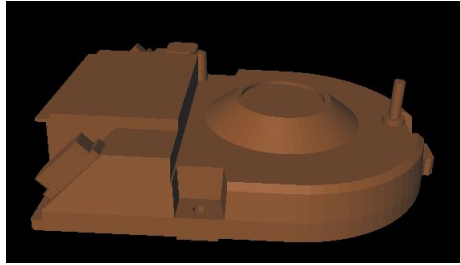


FIGURA 5.2: Base del robot creada mediante VRML.

una cadena cinemática como la descrita en la Figura 2.13, para realizar este ensamble es necesario utilizar el complemento de MatLab[®], “*V-Realm Builder*”, el cual nos ayuda a anudar de forma eficiente cada uno de los cuerpos que conforman el robot, dando las características de requeridas a dichos cuerpos, como se muestra en la Figura 5.3 al rea-

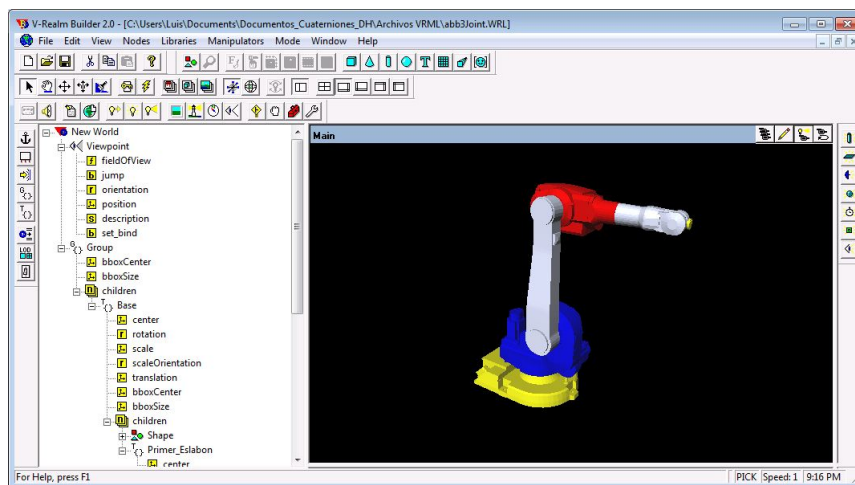


FIGURA 5.3: Union de los cuerpos del robot.

lizar esta union es posible manipular el modelo virtual del robot, colocando los centros rotacionales adecuados para realizar la simulación del movimiento de este robot.

5.1.2. Animación del modelo VRML mediante MatLab®

Mediante el uso de MatLab® es posible manipular cada una de las articulaciones del modelo utilizando un visor de navegación. Mediante este navegador es posible manipular los objetos creados y visualizar su entorno gráfico.

MatLab® genera un enlace entre el mundo virtual y el cálculo obtenido de las ecuaciones programadas a través del comando:

- ‡ vrworld: Crea un mundo virtual asociado con el archivo de nombre de archivo VRML y devuelve su identificador.
- ‡ vr.canvas: Crea una figura de MatLab® que contiene un lienzo de realidad virtual que muestra al mundo la realidad virtual especificado.

Una vez que el enlace se ha realizado se puede acceder y modificar cualquier parámetros correspondiente a este mundo virtual.

Para la generación del movimiento del robot IRB1600, gobernado por las ecuaciones desarrolladas en el modelo via cuaterniones es necesario crear una interfaz, que utilice de forma eficiente las características del modelo VRML, el cuál proviene del modelo CAD del robot, así como también debe contar con la capacidad de solucionar las ecuaciones del modelo rotacional mediante MatLab®, las cuales ya han sido programadas, dicha union se logra mediante una serie de comandos que funcionan en MatLab®, y que modifican los valores asociados al mundo virtual relacionado al modelo VRML. Entre los comandos más útiles y comunes para los objetos se tiene:

- ‡ Center (centro): Representa la posición (x,y,z) del punto de un objeto, sobre el cual se establece la traslación y rotación del mismo (se debe de tener en cuenta esto al momento de enlazar con MatLab®)
- ‡ Rotation (rotación): Establece la rotación absoluta del cuerpo con respecto al sistema de coordenadas globales.
- ‡ Scale (escalado): Constituye al factor de escalado del objeto en cada uno de los ejes.

- ‡ Traslación (traslación): Establece la traslación del cuerpo con respecto al sistema de coordenadas globales.

Para dar movimiento al robot, a cada pieza que se mueve se le asigna una rotación o traslación, este movimiento se toma con respecto al centro de gravedad de las piezas a mover, por lo que es necesario agregar el centro de rotación antes de realizar el movimiento.

5.2. Creación del la GUI en MatLab[®]

Las herramientas de realidad virtual proveen una flexible interfaz con MatLab[®] y los mundos de realidad virtual. Una vez que los objetos en un ambiente 3D son creados, asociamos estos objetos con MatLab[®], obteniendo su control mediante el uso de funciones y métodos matemáticos.

Desde MatLab[®] pueden ser creadas Interfaces Gráficas de Usuario (GUI's por sus siglas en ingles), mediante las cuales es posible realizar mapas virtuales del funcionamiento de dichos objetos, creando llamadas, funciones que provean de movimiento, propiedades de los objetos, determinar la estructura de los visores o asignar cualquier nuevo valor a todas las variables de los nodos y sus campos.

5.2.1. GUI Cinemática directa

Para tener un control de forma gráfica se construyo una GUI en la cual se pudiera manipular las variables relacionadas con cada una de las articulaciones del robot IRB1600 y se pudiera compara los resultados obtenidos de forma numérica, a través de una representación gráfica que muestre los valores calculados mediante el modelo rotacional además de una representación gráfica mediante el control de movimiento del modelo VRML del robot.

La Figura 5.4 muestra la interfaz gráfica de usuario en la cual se lleva a cabo la animación del modelo construido en VRML del robot, esta animación se realiza mediante

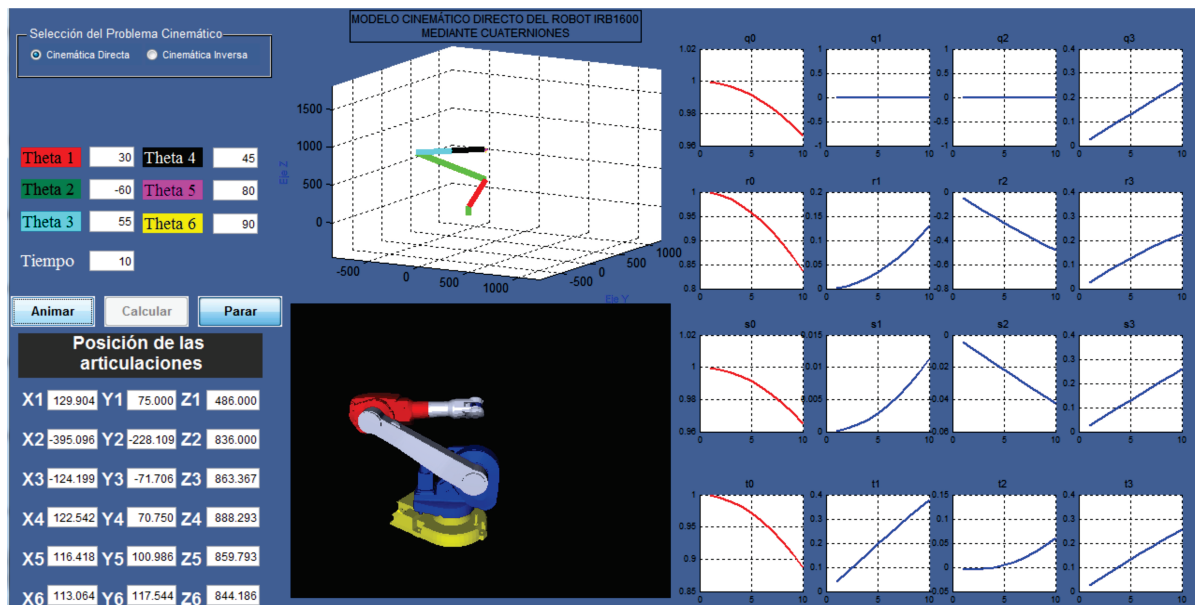


FIGURA 5.4: GUI para la animación VRML del robot.

la introducción de datos, los cuales, son calculados por los algoritmos del modelo rotacional, el funcionamiento de esta GUI se describe a continuación:

En la GUI se cuenta con una sección para la selección del problema cinemático, en

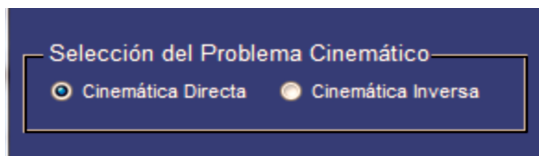


FIGURA 5.5: Selección del problema cinemático.

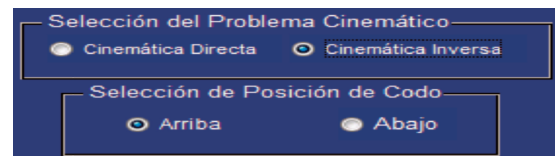


FIGURA 5.6: Selección codo arriba/codo abajo.

el cuál, se puede elegir entre cinemática directa y cinemática inversa como se muestra en la Figura 5.4, si es seleccionada la opción de la cinemática inversa aparece una subopción para elegir entre la configuración codo arriba y codo abajo, como se muestra en la Figura 5.5, ya que cada una de los botones de opción están enlazados a una parte del algoritmo creado para la solución de la cinemática inversa descrito en la Sección 4.2.4. Para el caso de “Cinemática Directa” se activan los parámetros que corresponden a los desplazamientos angulares correspondientes a los cuaterniones relativos de cada articulación (ver Figura 5.6), hay que tomar en cuenta que estos valores están restringidos a las limitaciones físicas del robot (ver ANEXO DEL DATA SHEET DEL ROBOT), también se selecciona el tiempo en el cuál se realizara la animación. una ves selecciona-

Theta 1	30	Theta 4	45
Theta 2	-60	Theta 5	80
Theta 3	55	Theta 6	90
Tiempo	10		

FIGURA 5.7: Selección de los desplazamientos angulares.

dos los desplazamientos angulares y el tiempo en el que se realizara la animación, para visualizar el movimiento y calcular los desplazamientos que tendrá el robot en cada una de sus articulaciones es necesario pulsar el botón de “Animar”(ver Figura 5.8), dicho botón activa el comienzo del cálculo con el cual se realiza el movimiento de los eslabones del robot .

Animar	Calcular	Parar
--------	----------	-------

FIGURA 5.8: Botones de animación.

Los datos obtenidos del cálculo se muestran en la parte “Posición de las articulaciones” (ver Figura 5.9) ,también se pueden observar las gráficas que se derivan de los

Posición de las articulaciones					
X1	129.904	Y1	75.000	Z1	486.000
X2	-395.096	Y2	-228.109	Z2	836.000
X3	-124.199	Y3	-71.706	Z3	863.367
X4	122.542	Y4	70.750	Z4	888.293
X5	116.418	Y5	100.986	Z5	859.793
X6	113.064	Y6	117.544	Z6	844.186

FIGURA 5.9: Botones de animación.

cuaterniones obtenidos del modelo rotacional.

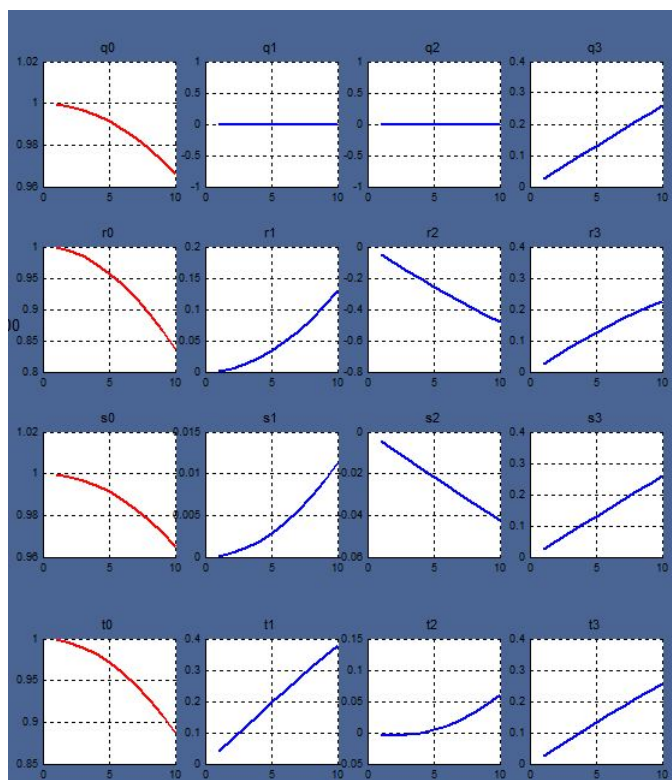


FIGURA 5.10: Cuaterniones de rotación.

Los datos numéricos de dichos cuaterniones son los que dan el movimiento al modelo VRML, a través de la modificación del parámetro “*Rotation*”.

Es posible observar en la programación que describe el funcionamiento de la GUI como se realiza la modificación de los valores que realizan la animación del modelo VRML (ver Apéndice C). De esta forma se puede verificar que el modelo rotacional desarrollado con cuaterniones es una representación confiable de las rotaciones realizadas por el robot para la cinemática directa, como se muestra en la Figura 5.11.

NOTA: Los valores que realizan las rotaciones para el mundo virtual (VRML) se ejecutan mediante las rotaciones descritas por un eje-ángulo por lo que los cuaterniones obtenidos del modelo rotacional se convierten a este formato mediante la siguientes

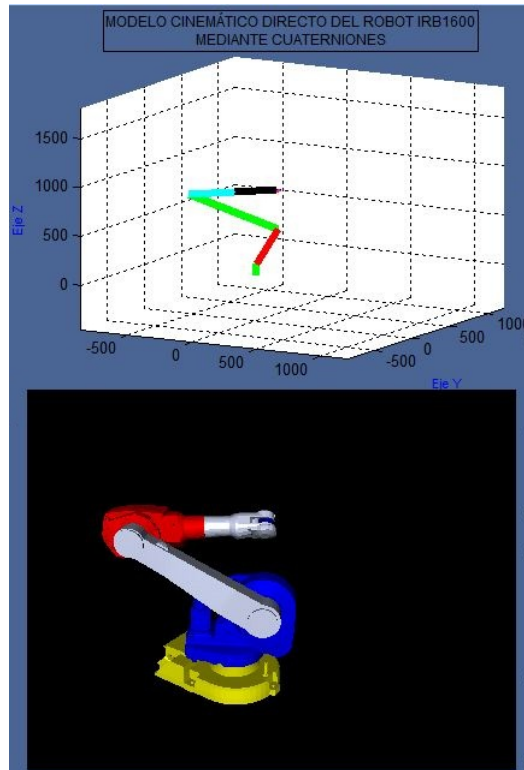


FIGURA 5.11: Representación de la animación MatLab®/VRML.

ecuaciones:

$$\alpha_q(q) = 2\arccos(q_0) \quad (5.1)$$

$$\mathbf{n}_q(q) = \frac{\mathbf{q}_{1:3}}{|\mathbf{q}_{1:3}|} = \frac{\mathbf{q}_{1:3}}{\sqrt{1 - q_0^2}} \quad (5.2)$$

donde $\mathbf{n}_q(q)$ corresponde al eje de rotación y $\alpha_q(q)$ es el ángulo de rotación[32].

El diagrama de la Figura 5.12 es una representación esquemática del funcionamiento de la GUI para la cinemática directa.

5.2.2. GUI Cinemática inversa

La GUI cuenta con la opción de representar de forma gráfica el comportamiento del modelo rotacional, desarrollado con cuaterniones, mediante la comprobación para la cinemática inversa. En la Sección 4.2.4 se muestra el desarrollo de la solución para el problema cinemático inverso para el robot IRB1600.

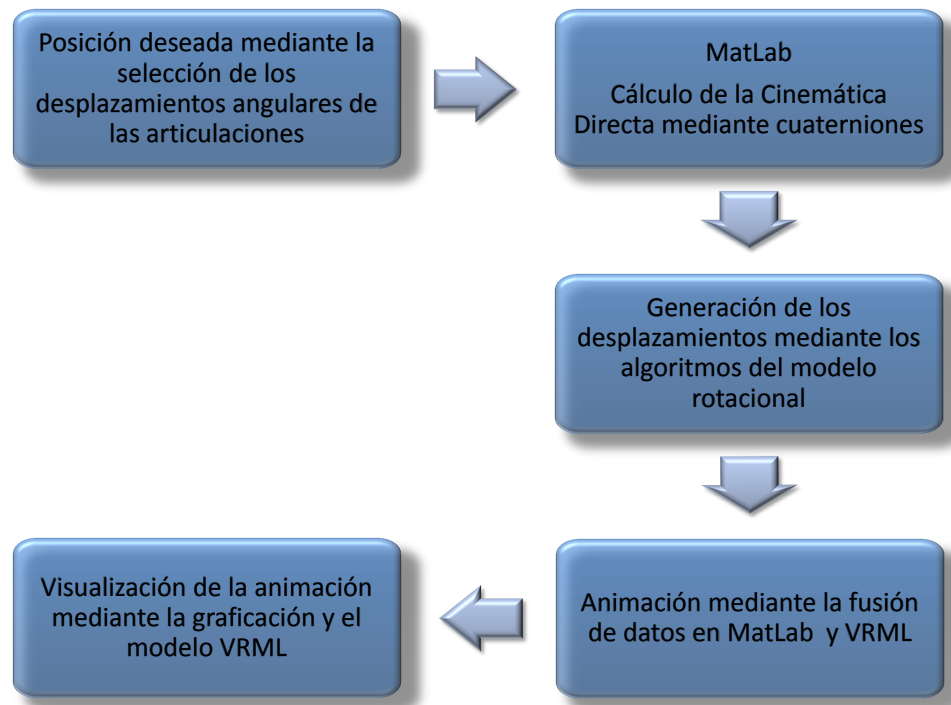
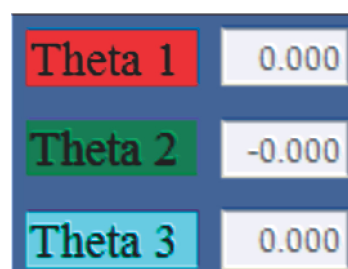


FIGURA 5.12: Esquema de funcionamiento de la GUI.

Para poder calcular los cuaterniones necesarios para adquirir una configuración específica para el robot a través del problema cinemático inverso, es necesario seleccionar en la opción “*Cinemática Inversa*” en la sección “*Selección del Problema Cinemático*” (ver Figura 5.5) como se menciona en la Sección 4.2.4, el problema cinemático inverso cuenta con dos soluciones posibles conocidas como “*Codo Arriba y Codo Abajo*” por lo que es necesario, especificar la configuración deseada, en la sección “*Selección de Posición de Codo*” de la GUI como se muestra en la Figura 5.6.

Al realizar esta selección se desactivan los campos de las variables a calcular (θ_1 , θ_2 y

FIGURA 5.13: Campos θ_1 , θ_2 y θ_3 desactivados.

θ_3), como se muestra en la Figura 5.13, estos son los valores que determinan la posición del robot, como se describe en la Sección 4.2.4

Los campos correspondientes a la orientación (θ_4 , θ_5 y θ_6), permanecen activos para establecer la orientación que deberá adquirir el robot al alcanzar el valor deseado, siempre y cuando se encuentre en su espacio de trabajo (ver Apendice A.), como se muestra en la Los campos correspondientes a la posición, son los valores relacionados con la

Theta 4	45
Theta 5	80
Theta 6	90

FIGURA 5.14: Selección de la orientación deseada, para la cinemática inversa.

posición del efector final, es decir, X_6 , Y_2 y Z_6 , los cuales se activan al seleccionar la opción “*Cinemática Inversa*” (ver Figura 5.15).

X6	1000	Y6	400	Z6	1100
----	------	----	-----	----	------

FIGURA 5.15: Selección de la posición deseada, para la cinemática inversa.

Una vez seleccionada la posición y orientación deseada se pulsa el botón “*Calcular*” (este se encuentra activo una vez que se selecciona la opción *Cinemática Inversa*), el cuál, realiza el calculo de las variables θ_1 , θ_2 y θ_3 , colocando el valor calculado en la interfaz (ver Figura 5.16).

en la opción “*Cinemática Inversa*—→*Selección del Problema Cinemático*” se puede alternar entre la configuración “*Codo Arriba y Codo Abajo*”, con lo que al pulsar nuevamente el botón calcular, se observa como cambian los valores de θ_1 , θ_2 y θ_3 , para cada una de las configuraciones (ver Figura 5.17) Cuando la selección esta hecha se puede observar la animación del recorrido que tendrá el robot pulsando el botón “*Animar*”, el cuál, toma los parámetros que se encuentran en cada uno de los desplazamientos angulares y realiza el movimiento mediante la utilización del modelo rotacional basado

Theta 1	19.395	Theta 4	45
Theta 2	26.582	Theta 5	80
Theta 3	-29.695	Theta 6	90
Tiempo	10		

Animar Calcular Parar

FIGURA 5.16: Cálculo de las variables θ_1 , θ_2 y θ_3 .

Selección de Posición de Codo			
<input checked="" type="radio"/> Arriba <input type="radio"/> Abajo			
Theta 1	19.395	Theta 4	45
Theta 2	26.582	Theta 5	80
Theta 3	-29.695	Theta 6	90
Tiempo	10		

Animar Calcular Parar

Selección de Posición de Codo			
<input type="radio"/> Arriba <input checked="" type="radio"/> Abajo			
Theta 1	19.395	Theta 4	45
Theta 2	86.328	Theta 5	80
Theta 3	-158.774	Theta 6	90
Tiempo	10		

Animar Calcular Parar

FIGURA 5.17: Cálculo de las variables θ_1 , θ_2 y θ_3 , selección de la configuración *Codo Arriba* y *Codo Abajo*.

en cuaterniones. En las Figuras 5.18 y 5.19) se puede observar la animación realizada mediante el modelo VRML, las posiciones que tiene cada una de las articulaciones, así como, las gráficas relacionadas a sus cuaterniones de rotación, mostrando que se tienen una rotación continua evitando puntos singulares.

Mediante el uso de esta GUI es posible observar que el modelo VRML del robot IRB1600 representa de forma confiable las rotaciones mediante cuaterniones de las articulaciones de este robot, teniendo una representación gráfica del movimiento realizado sin necesidad de realizarlo físicamente, teniendo con esto una gran ventaja a la hora de verificar los movimientos realizados por el robot.

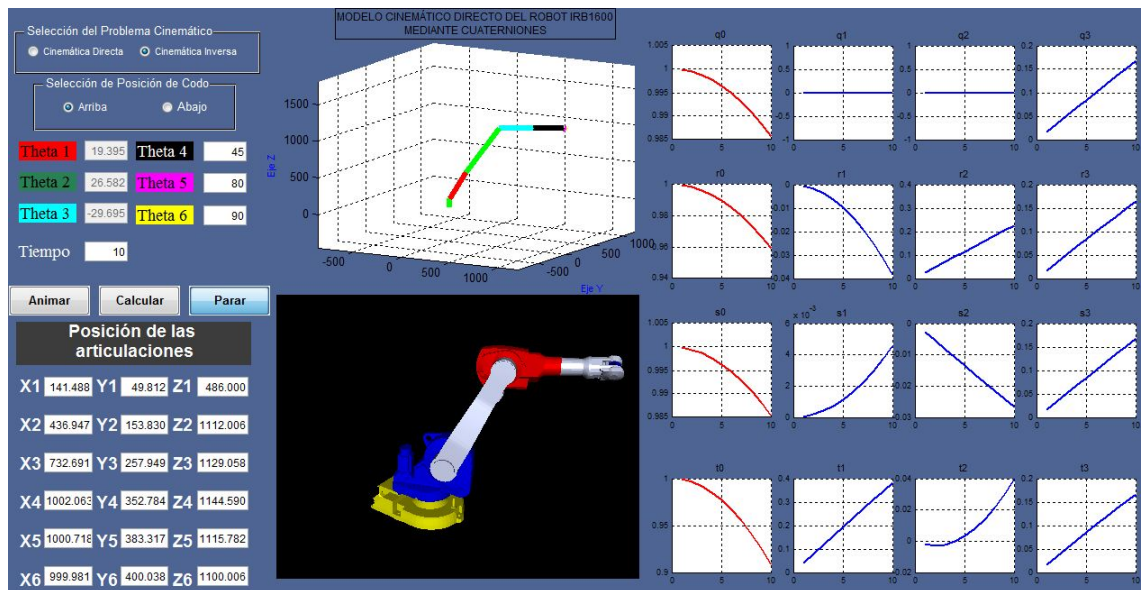


FIGURA 5.18: Animación mediante la configuración codo arriba.

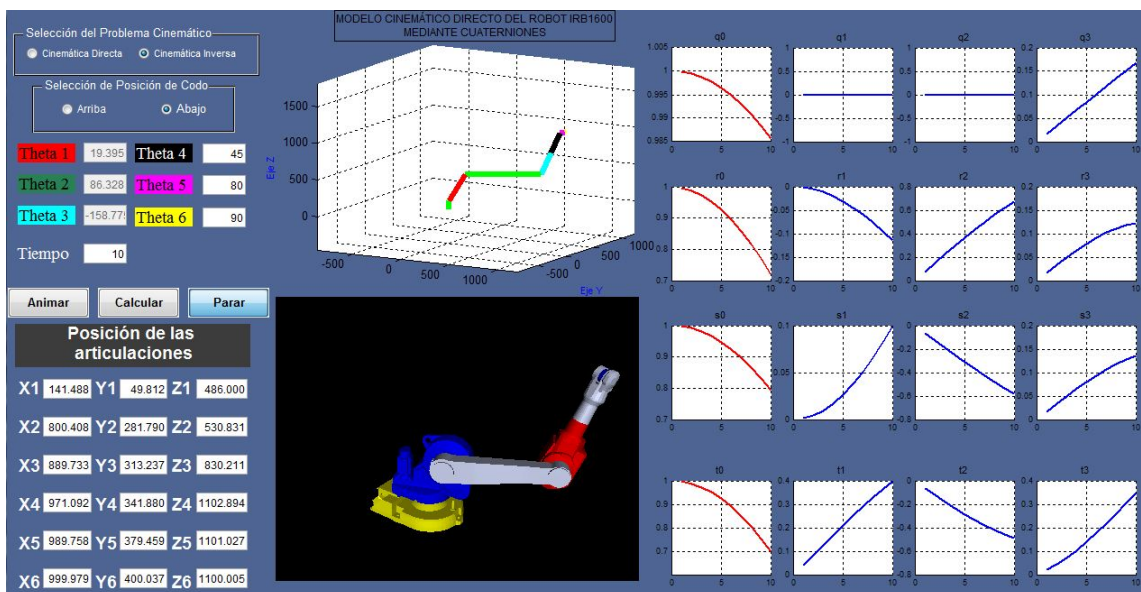


FIGURA 5.19: Animación mediante la configuración codo abajo.

Capítulo 6

Validación física del modelo

A lo largo de este capítulo se llevará a cabo la validación del modelo rotacional desarrollado vía cuaterniones realizado en el Capítulo 4 con el modelo virtual en VRML del Capítulo 5. La finalidad de esta validación es poder utilizar el modelo virtual como una herramienta útil y actual en el modelado de la cinemática directa e inversa.

Por otra parte se realizó la validación física del modelo matemático vía cuaterniones y el modelo virtual, utilizando para este fin la central inercial Pololu MinIMU-9 v2 (ver Apéndice D).

6.1. Medición de parámetros en el robot

Para llevar a cabo la validación física del modelo rotacional, se utilizó la central inercial Pololu MinIMU-9 v2 (ver Apéndice D), la cual combina un girómetro con tres acelerómetros ortogonales DC y tres magnetómetros ortogonales, los cuales, actúan como dispositivos esclavos en un mismo bus I²C, así es posible adaptar esta central inercial, a un microcontrolador para la salida de datos en ambientes dinámicos y estáticos.

Las centrales inerciales en general son aptas para aplicaciones como; vehículos aéreos, submarinos, navegación robótica, ciencias computacionales, animación, manufactura, etc.

La siguiente información es importante tomarla en cuenta para el entendimiento del funcionamiento de la central inercial.

6.1.1. Girómetros

Un girómetro es un aparato que mide velocidades de rotación y gracias a la tecnología actual tiene alta precisión. Se pueden utilizar en aplicaciones que involucren navegación, estudios geofísicos o pruebas que incluyan relatividad en general[33].

Un girómetro puede utilizar diferentes tipos de giroscopios uno de los más importantes es el giroscopio vibrador que funciona gracias al efecto piezoeléctrico el cual se genera cuando una fuerza o presión son ejercidas sobre el material piezoeléctrico, este produce un voltaje. Recíprocamente, el estado opuesto puede ser inducido. Si una corriente alterna se aplica al material, este resuena o vibra en una cierta frecuencia. A esto se le llama el efecto piezoeléctrico inverso.

Cuando una corriente alterna es aplicada al giroscopio, esto causa que los dos brazos tipo "T" que son externos del elemento oscilen y se muevan en dirección opuesta el uno del otro a una frecuencia alta. Cuando se es aplicada una rotación, los elementos del giroscopio son minuciosamente girados y una segunda fuerza llamada fuerza de Coriolis toma un papel importante en el funcionamiento de este sistema de sensado.

La fuerza de Coriolis actúa perpendicularmente sobre el elemento oscilante del giroscopio de arriba a abajo, causando que los elementos de sensado que se encuentran en medio del sensor vibren y por lo tanto produzcan un corriente proporcional a la razón de la velocidad angular o rotación aplicada. La corriente es amplificada dentro del giroscopio, y un voltaje correspondiente a la velocidad angular es generado. Esta señal se calcula electrónicamente y se envía la información obtenida por el sensor a la interface o al actuador correspondiente (ver) [34].

La fuerza de Coriolis que actúan sobre la masa de perturbación es más fuerte con una masa de perturbación más pesada, porque la fuerza de Coriolis es proporcional a la velocidad de vibración, la velocidad angular y el peso de perturbación [35].

Sin embargo los giroscopios tienen índices de flujo relativamente altos, que causan el crecimiento ilimitado en errores en orientación. A causa de su potencial de el crecimiento ilimitado de errores, los giroscopios sólo puede ser usados en conjunción con periódicas actualizaciones de posición absoluta[36], también presentan un offset que

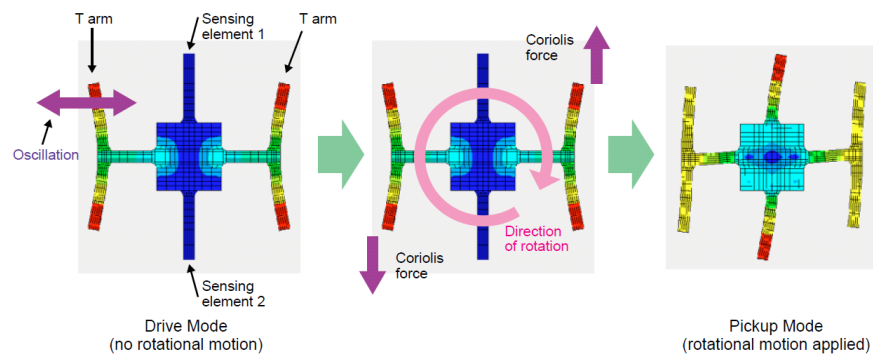


FIGURA 6.1: Generalmente, en el Modo de reposo (sin rotación), los elementos de sensado se encuentran estables y solo los brazos tipo "T" oscilan. Cuando un movimiento rotatorio es aplicado, los elementos de sensado vibran, y el sensor detecta una señal diferencial de la rotación [34]

varía con la temperatura.

Para medir posiciones angulares en los tres ejes (pitch, roll y yaw), son necesarios tres giroscopios y de esta forma si los tres sensores giroscopios se encuentran integrados en un solo dispositivo, las velocidades angulares generadas por las perturbaciones en la masa o por la fuerza de Coriolis pueden ser detectadas fácilmente.

6.1.2. Acelerómetros

Se denomina acelerómetro a cualquier instrumento destinado a medir aceleraciones. El acelerómetro es uno de los transductores más versátiles, siendo el más común el piezoeléctrico por compresión. Este se basa en que, cuando se comprime un retículo cristalino piezoeléctrico, se produce una carga eléctrica proporcional a la fuerza aplicada.

Los elementos piezoeléctricos están hechos normalmente de circonato de plomo. Los elementos piezoeléctricos se encuentran comprimidos por una masa, sujeta al otro lado por un muelle y todo el conjunto dentro de una caja metálica. Cuando el conjunto es sometido a vibración, el disco piezoeléctrico se ve sometido a una fuerza variable, proporcional a la aceleración de la masa.

Este dispositivo junto con los circuitos eléctricos asociados se puede usar para la medida de velocidad y desplazamiento además de la determinación de formas de onda y frecuencia. Una de las ventajas principales de este tipo de transductor es que se puede hacer tan pequeño que su influencia sea despreciable sobre el dispositivo vibrador. El intervalo de frecuencia típica es de 2 Hz a 10 KHz.

Los acelerómetros electrónicos permiten medir la aceleración en una, dos o tres dimensiones, esto es, en tres direcciones del espacio ortonormales. Esta característica permite medir la inclinación de un cuerpo, puesto que es posible determinar con el acelerómetro la componente de la aceleración provocada por la gravedad que actúa sobre el cuerpo. Un acelerómetro también es usado para determinar la posición de un cuerpo, pues al conocerse su aceleración en todo momento, es posible calcular los desplazamientos que tuvo. Considerando que se conocen la posición y velocidad original del cuerpo bajo análisis, y sumando los desplazamientos medidos se estima la posición [37]. Sin embargo, es necesario separar la medida de aceleración debida al movimiento y la aceleración gravitacional para la estimación de la orientación [38].

6.1.3. Magnetómetros

Se llama magnetómetro a los dispositivos que sirven para cuantificar en fuerza y/o dirección de la señal magnética de una muestra. Hay algunos que miden el cambio en peso aparente que se produce en una muestra al aplicar un campo magnético, y también los hay muy sofisticado, como los dotados de Dispositivos superconductores de interferencia cuántica, que son los más sensibles actualmente.

Magnetómetros de tres ejes (Three axis magnetometers - TAMs) son usados extensamente para operaciones de navegación espacial. El concepto básico detrás de estos dispositivos es uno muy simple, implicando un sensor magnético acoplado con una unidad de electrónica para proporcionar datos en un formato digital. Estos sensores son útiles ya que proporcionan tanto dirección como magnitud del campo magnético; Estos son de peso ligero, confiables, y tienen bajo exigencias de potencia. Es conocido que un TAM puede ser usado determinar una posición en tres ejes cuando se acopla a un giroscopios o un modelo dinámico [39].

Una problema en el uso de magnetómetros es exactitud de la posición obtenida debida a la precisión de la calibración. La exactitud obtenida usando un TAM depende de ciertos factores, incluyendo: perturbación ferromagnética, factores de escala y correcciones de no ortogonalidad. Los factores de escala y correcciones de no ortogonalidad ocurren porque los ejes del magnetómetro individuales no son ortogonales, típicamente debido a gradientes térmicos dentro del magnetómetro o a la tensión mecánica. Aún perfectamente calibrado y sin ruido, existen inexactitudes en los magnetómetros, debido a la

incertidumbre en el modelo del campo geomagnético de la Tierra. Estos errores lamentablemente son típicamente no lineales y no gaussianos, que depende de muchos factores (véase [40] para más detalles). Estos factores limitan la estimación de la posición. La calibración de magnetómetro a menudo es lograda usando métodos donde una colección entera de datos debe ser almacenada para determinar los parámetros desconocidos. Este proceso a menudo es repetido muchas veces durante el uso en un sistema espacial para asegurar la mejor precisión posible obtenida de las medidas de un magnetómetro.

Ahora el funcionamiento de un magnetómetro es básicamente esta: la señal del magnetómetro (y_M) mide el campo magnético terrestre (y_{ME}) y es sensible a cualquier perturbación electromagnética del entorno (y_d):

$$y_M = y_{ME} + y_d$$

En ausencia de perturbaciones ferromagnéticas, la medida normalizada de los magnetómetros nos ofrece la orientación en el plano horizontal (acimut) del sistema local o inercial ($[X_L Y_L]$) respecto al de referencia global ($[X_G Y_G]$). Por tanto, fusionando ambas estimaciones podemos conocer la orientación local ($[X_L Y_L Z_L]$) [41].

6.1.4. Brazo articulado

Con respecto a la modelización, esta se baso en un brazo articulado, en el marco de la captura de movimiento, para ello, nos centramos en los robots denominados “ligeros”, robot en los cuales todos los actuadores son solidarios con la armadura a fin de limitar el movimiento de las masas de los diferentes elementos.

Aunque las herramientas son válidas para todos los tipos de robots, nos basamos exclusivamente en los robots de tipo 3R [42], (**R** para la **R**otación) que ofrece un interés especial para nuestra aplicación, ya que nuestro estudio se basa en articulaciones de tipo rotacionales.

6.2. Validación física del modelo rotacional

Los datos obtenidos de la central inercial de acuerdo con lo establecido en la Sección 6.1.1, Sección 6.1.2 y Sección 6.1.3 son acoplados mediante su conexión a través de la plataforma de desarrollo abierta “Arduino” [43], el cual, toma los valores de la central inercial Pololu MinIMU-9 v2 mediante el algoritmo de programación basada en [44], el cual, realiza las mediciones para cada uno de los sensores acoplados a esta unidad inercial, estos valores son entregados en ángulos de Euler y guardados para posteriormente ser procesados.

La central inercial es montada en cada una de las articulaciones móviles del robot, con esto se guardan los valores de desplazamiento rotacional obtenidos a través de la central inercial, una vez tomados los valores de los desplazamientos rotacionales, de cada una de las articulaciones, los datos son recuperados por MatLab[®] y procesados para reproducir el movimiento realizado por el robot a través del modelo virtual desarrollado en la Sección 5.1.

De los valores obtenidos es posible obtener los valores mostrados en la Figura 6.2 estos datos corresponden a una tarea de trabajo utilizada en una celda de manufactura industrial. con los valores obtenidos de la central inercial se realiza la validación del

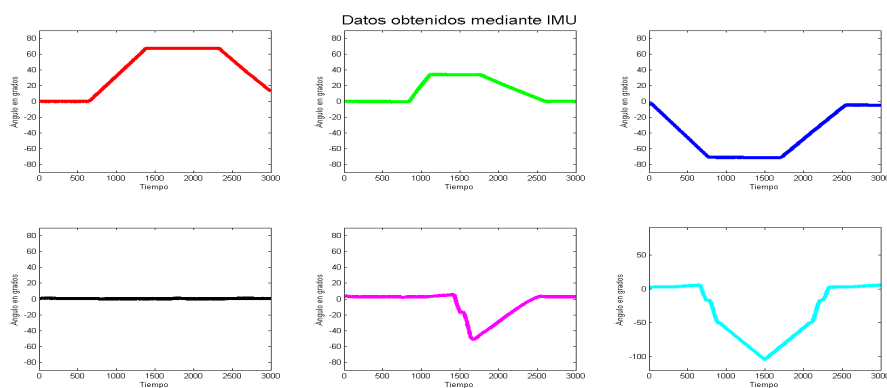


FIGURA 6.2: Gráfica de datos obtenidos mediante la IMU Pololu MinIMU-9 v2

modelo rotacional (ver Capítulo 4) a través del modelo virtual Capítulo 5. Mediante la

animación producida por el modelo virtual es posible observar que se representa de una forma confiable el movimiento realizado físicamente por el robot a través del modelo virtual como se puede observar en la Figura 6.3, como se menciona en el Capítulo 5 este modelo virtual obedece a las ecuaciones obtenidas del modelado rotacional basado en cuaterniones para el robot IRB1600.

Del modelo rotacional basado en cuaterniones también se pueden obtener las gráficas

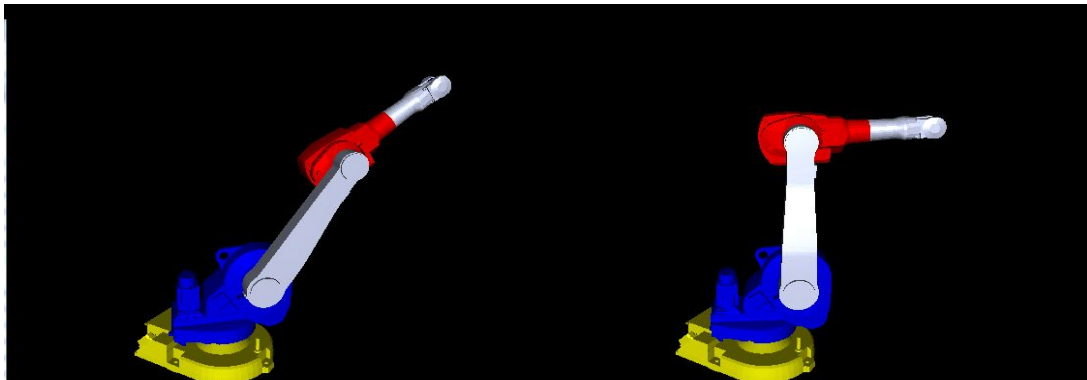


FIGURA 6.3: Animación obtenida mediante los valores tomados de la IMU

comparativas de los valores obtenidos para cada una de las juntas rotacionales tomando los valores que representan los cuaterniones para cada una de esas juntas en ángulos de Euler así como los valores obtenidos mediante las matrices de rotación en las que se basa el “*algoritmo Denavit-Hartenberg*” para la obtención del modelo cinemático directo (ver Sección 2.3.1),

la Figura 6.4 representa el movimiento generado por la primer articulación, el cual,

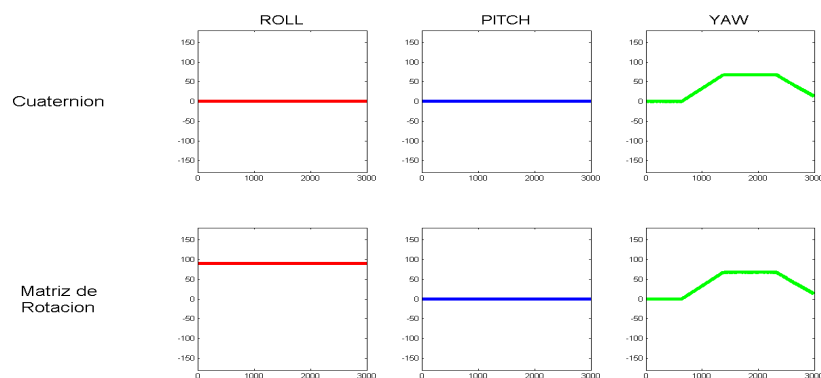


FIGURA 6.4: Valores comparativos de los Cuaterniones vs Matrices de rotación para la primer articulación

corresponde al movimiento de la base, teniendo como resultado un desplazamiento angular en “*Yaw*”.

La Figura 6.5 representa el movimiento de la segunda articulación sobre la cadena

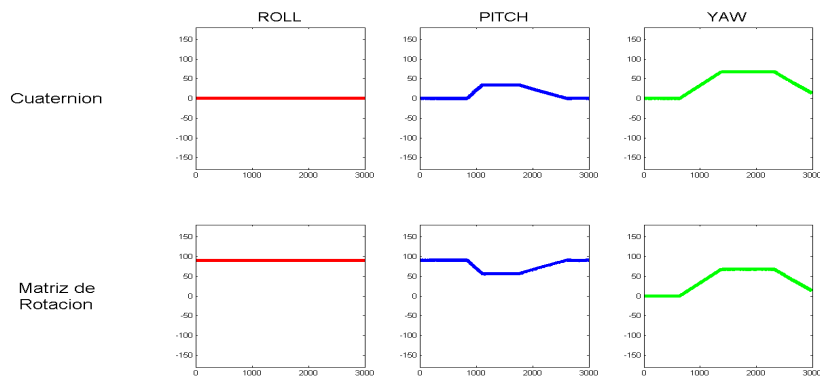


FIGURA 6.5: Valores comparativos de los Cuaterniones vs Matrices de rotación para la segunda articulación

cinemática, por lo que el movimiento generado tiene como resultado, el desplazamiento angular en “*Pitch*” más el movimiento anterior en “*Yaw*”, debido a sus respectivos sentidos de rotación de acuerdo a al sistema de coordenadas de referencia.

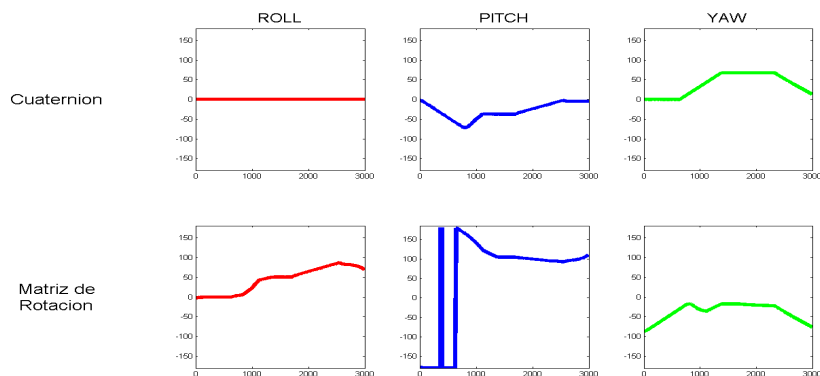


FIGURA 6.6: Valores comparativos de los Cuaterniones vs Matrices de rotación para la tercera articulación

En la Figura 6.6 es posible observar que los valores obtenidos mediante las matrices de rotación sufren indeterminaciones causadas por el efecto del “*Gimbal-Lock*” (ver Sección 2.3.1) en las rotaciones sucesivas de la cadena cinemática, debido al efecto de que

tienen las matrices de rotación en la convención Denavit-Hartemberg el giro de la tercer articulación es reflejado en el eje "Roll" ocasionando el ya mencionado Gimbal-Lock.

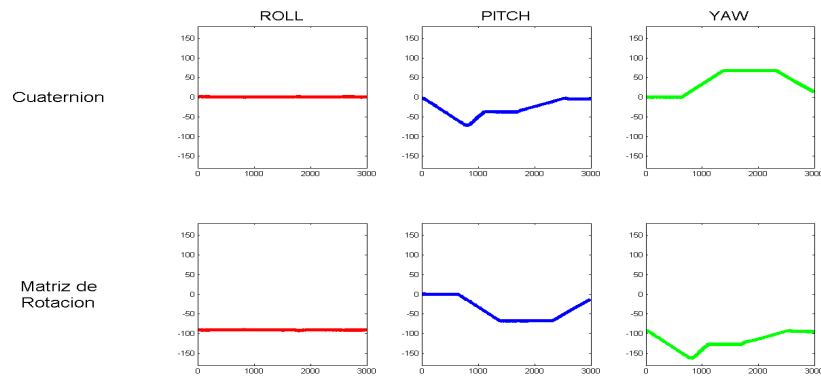


FIGURA 6.7: Valores comparativos de los Cuaterniones vs Matrices de rotación para la cuarta articulación

La Figura 6.7 muestra los movimientos relacionados con la cadena cinemática llegando a la cuarta articulación, de la cual es posible observar como la indeterminación que sufrió en la cuarta articulación genera cambios en ésta aumentando de manera significativa.

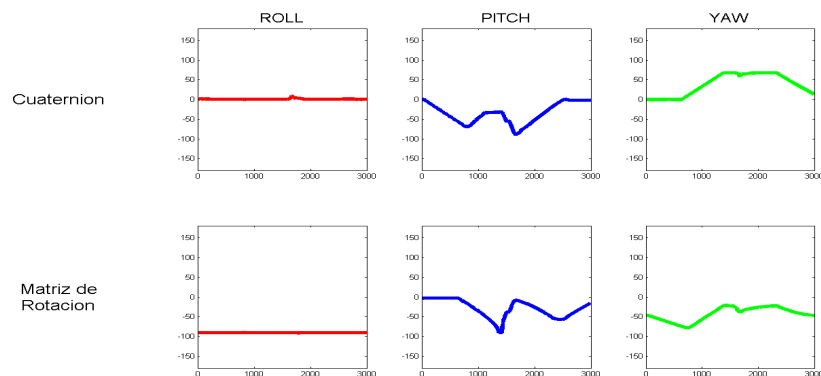


FIGURA 6.8: Valores comparativos de los Cuaterniones vs Matrices de rotación para la quinta articulación

Las figuras anteriores muestran como el uso de los cuaterniones para el desarrollo del modelo cinemático del robot, disminuye las indeterminaciones que se generan con el uso de las matrices de rotación.

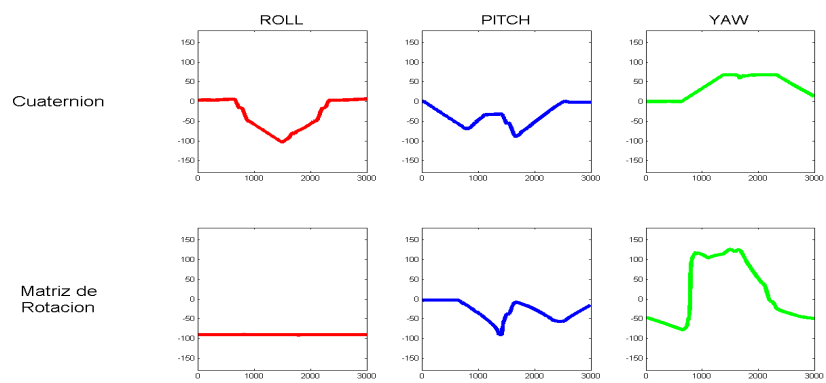


FIGURA 6.9: Valores comparativos de los Cuaterniones vs Matrices de rotación para la sexta articulación

Capítulo 7

Conclusiones

En el desarrollo de este trabajo se han estudiado las herramientas con las cuáles es posible desarrollar el modelo rotacional de un robot de 6 GDL con la estructura cinemática del IRB1600, de las que podemos concluir lo siguiente:

- ✓ Con el cuaternión se comprobó que es posible obtener un operador lineal y que este operador qvq^* es una rotación en \mathbb{R}^3 de un ángulo de 2θ al rededor del eje \mathbf{q} , mostrando que la representación de la rotación de un objeto en el espacio 3D es más sencilla y práctica.
- ✓ Con el operador qvq^* y un ángulo de $\frac{\theta}{2}$ se puede definir una transformación lineal $L\mathbf{v}$ la cual preserva el producto interno, la norma y el ángulo y además representa una rotación positiva del vector \mathbf{v} con respecto al marco de coordenadas por un ángulo de θ al rededor del eje del cuaternión \mathbf{q} .
- ✓ Con el uso operador lineal $L\mathbf{v}$ se llegó a desarrollar un operador rotacional mediante el cual se describen rotaciones sucesivas del cuerpo rígido, haciendo más sencillo el uso de los cuaterniones en el modelado de las rotaciones del vector \mathbf{v} .
- ✓ Mediante el uso del operador se demuestra que el modelo rotacional desarrollado en la Sección 4.2.2, es una representación fiel y confiable de la cinemática directa.

- ✓ Mediante el uso del método geométrico para la solución del problema cinemático inverso, se puede verificar que el uso de los cuaterniones para el desarrollo del modelo rotacional es una forma simple de obtener los cálculos para el desplazamiento angular de cada uno de los eslabones del robot.
- ✓ A través del desarrollo del modelo virtual se tiene es posible constatar que el modelo rotacional funciona de forma adecuada por lo que se puede confiar en las animaciones resultantes.
- ✓ Con el desarrollo de la GUI y la unión de esta con el modelo VRML y el modelo rotacional, se puede evaluar el correcto desplazamiento del robot para alcanzar un punto, esta aplicación podrá tener un futuro importante en el área académica e industrial.
- ✓ Con el uso de un software comercial (MatLab[®]) es posible realizar una programación de forma sencilla, esto facilita la exploración de la información, lo cual, permite generar ambientes amigables y complejos a costos relativamente bajos.
- ✓ La estimación de la orientación mediante la incorporación de los valores adquiridos de la central inercial Pololu MinIMU-9 v2, con el fin de resolver el caso cinemático permite validar el modelo con información real en tiempo diferido.

7.1. Trabajo futuro

Derivado de este trabajo se puede ampliar a varios trabajos a futuro, con la finalidad de poder implementar la programación de basada en cuaterniones y así conocer el comportamiento rotacional de un cuerpo rígido rotando alrededor de un vector unitario de forma tridimensional, obteniendo la posición del efector final en tiempo real.

- ✓ Realizar mejoras en la parte de la GUI donde se puedan implementar rutinas que pudieran ser visualizadas en el modelo virtual, con el fin de incrementar la funcionalidad de esta.
- ✓ Llevar a cabo una generalización de este trabajo a fin de obtener la cinemática de cualquier tipo de robot.

-
- ✓ Estimación en tiempo real de la desviación que se genera en los girómetros, lo cual es un problema de estimación en tiempo real ó en linea.
 - ✓ Debido a que en este trabajo se presenta un método de fácil implementación a futuro se desea reconocer la posición y la orientación para cualquier tipo de robot, lo cual, implica trabajar con cuaterniones duales los cuales ofrecen una solución a este problema ya que el modelo basado en cuaterniones duales trabaja con información como las orientaciones y las distancias.
 - ✓ Desarrollar el modelo dinámico para este tipo de robots basado en cuaterniones duales.
 - ✓ Implementar el modelo rotacional obtenido en una base que no se encuentre fija.

Apéndice A

Robot IRB1600

IRB 1600

The highest performance 10 kg robot.

Performance is often a trade off, optimizing for speed or accuracy. With ABB's IRB 1600, you don't have to choose. The robot's cycle times are shorter, sometimes half that of other robots, allowing you to increase throughput. Meanwhile, you will enjoy the work piece quality that only an ABB robot can offer. Extra everything.



Double your throughput

The IRB 1600 has up to 50 percent shorter cycle times than competing robots in material handling, machine tending and process applications. It speeds up and slows down faster than other robots, saving time while moving between tasks. This is possible due to ABB's patented second generation QuickMove motion control, combined with the robot's strong motors and low friction losses in the spur gears.

No more cutting corners

At high speed, most robots will cut corners. With the IRB 1600, the path will be the same regardless of speed, thanks to the robot's unique combination of brains and brawn. Intelligent second generation TrueMove motion control means that "what-you-program-is-what-you-get". Add muscle – a heavy and stiff design – low vibrations and low friction – and you have a robot that will deliver consistently high work piece quality, high yield and few rejects.

Outstanding reliability

The IRB 1600 offers outstanding reliability, even in the toughest environments and the most demanding 24/7 duty cycles. The entire manipulator is IP 54 classed and sensitive parts are IP 67 classed as standard. The optional protection Foundry Plus offers IP 67, special paint, rust protection and is tailor made for tough foundry environments. The rigid and heavy design combined with spur gears, make the robot extremely robust. Smart collision detection software further adds to the robot's outstanding reliability.

Easy to integrate

Mounting is fully flexible: on a shelf, on the wall, tilted or inverted. By choosing the compact short-arm version with the 1.2 m reach, you can even fit the IRB 1600 inside a machine, while ensuring sufficient payload as the maximum total load is as high as 36 kg.

Sustainable and healthy

Low friction spur gears, and no unnecessary moves due to QuickMove and TrueMove, reduces power consumption down to 0.58 kW at max speed, and even less at low speeds. The airborne noise level of just <70 dB (A) secures a healthy sound environment.

IRB 1600

Main applications

Machine tending, material handling, arc welding, cutting, dispensing, assembly, palletizing and packaging, measuring, die casting, injection moulding

Specification

Variants	Reach(m)	Payload(kg)	Armload(kg)
IRB 1600-6/1.2	1.2	6	30,5
IRB 1600-6/1.45	1.45	6	30,5
IRB 1600-10/1.2	1.2	10	20,5
IRB 1600-10/1.45	1.45	10	20,5
Number of axes	6+3 external (up to 36 with MultiMove)		
Protection	Standard IP54; opt. FoundryPlus 2 (IP 67)		
Mounting	Floor, wall, shelf, tilted, inverted		
IRC5 Controller variants	Single cabinet, Dual cabinet, Compact		

Physical

Dimensions robot base:	484 x 648
Robot height: IRB 1600-6/1.2 and IRB 1600-10/1.2	1069 mm
Robot height: IRB 1600-6/1.45 and IRB 1600-10/1.45	1294 mm
Robot weight:	250 kg

Performance (according to ISO 9283)

	6/1.2	6/1.45	10/1.2	10/1.45
Pos. repeatability (RP)	0.02 mm	0.02 mm	0.02 mm	0.05 mm
Path repeatability (RT)	0.13 mm	0.19 mm	0.06 mm	0.13 mm

Movement

Working range	1.2	1.45
Axis 1	+180° to -180°	+180° to -180°
Axis 2	+136° to -63°	+150° to -90°
Axis 3	+55° to -235°	+65° to -245°
Axis 4	+200° to -200° def. +/-190° revolution	+200° to -200° def. +/-190° revolution
Axis 5	+115° to -115°	+115° to -115°
Axis 6	+400° to -400° def. +/-288 revolution	+400° to -400° def. +/-288 revolution
Maximum Speed	6 kg	10 kg
Axis 1	150°/s	180°/s
Axis 2	160°/s	180°/s
Axis 3	170°/s	185°/s
Axis 4	320°/s	385°/s
Axis 5	400°/s	400°/s
Axis 6	460°/s	460°/s

Electrical connections

Supply voltage	200-600 V, 50-60 Hz
Power consumption	ISO-Cube at max speed 0.58 kW

Environment

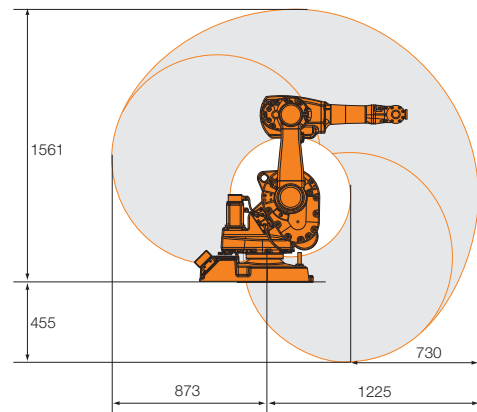
Ambient temperature for mechanical unit:

During operation	+ 5°C (41°F) to + 45°C (113°F)
During transportation and storage	- 25°C (- 13°F) to + 55°C (131°F)
For short periods (max 24h)	up to + 70°C (158°F)
Relative humidity	Max. 95% at constant temperature
Safety	Double circuits with supervisions, emergency stops and safety functions, 3-position enable device
Emission	EMC/EMI shielded

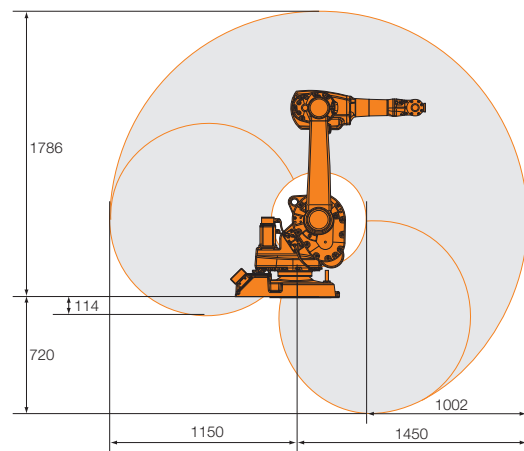
Data and dimensions may be changed without notice

Working range

IRB 1600-x/1.2



IRB 1600-x/1.45



Apéndice B

Algoritmo del modelo rotacional del robot IRB1600

```
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
function [R]=Qrot(q,p)
%Esta funcion rota un punto mediante el operador rotacional Lv
% donde el cuaternion de rotacion es q y el punto a rotar es p
% teniendo como retorno un punto R
p1=[      2*q(1)^2-1+2*q(2)^2,      2*q(2)*q(3)-2*q(1)*q(4),      2*q(2)*q(4)+2*q(1)*q(3);
      2*q(2)*q(3)+2*q(1)*q(4),      2*q(1)^2-1+2*q(3)^2,      2*q(3)*q(4)-2*q(1)*q(2);
      2*q(2)*q(4)-2*q(1)*q(3),      2*q(3)*q(4)+2*q(1)*q(2),      2*q(1)^2-1+2*q(4)
      ^2]*[p(1);p(2);p(3)];
R=[p1'];
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

function [p01, p0, p1, p2, p3, p4, p5, p6,q,r,s,t,u,w,R,S,T,U,W]=QRQ(theta1, theta2,
      theta3,theta4,theta5,theta6)
% Esta funcion requiere como entrada los desplazamientos angulares de cada articulacion
% Cuaterniones de Rotacion de cada articulacion
q = [cos(theta1/2),      0,      0,      sin(theta1/2)];
r = [cos(theta2/2),      0,      sin(theta2/2),      0];
s = [cos(theta3/2),      0,      sin(theta3/2),      0];
t = [cos(theta4/2)      sin(theta4/2),      0,      0];
u = [cos(theta5/2),      0,      sin(theta5/2),      0];
w = [cos(theta6/2),      sin(theta6/2),      0,      0];
% Puntos que describen cada una de las articulaciones del robot
pb = [ 0, 0, 124];
p1 = [ 150, 0, 362];
p2 = [ 0, 0, 700];
p3 = [ 314, 0, 0];
p4 = [ 286, 0, 0];
p5 = [ 0, 0, 0];
p6 = [ 65, 0, 0];
% Producto de los cuaterniones de cada articulacion
R=Qpro(q,r);
```

```
S=Qpro(Qpro(q,r),s);
T=Qpro(Qpro(Qpro(q,r),s),t);
U=Qpro(Qpro(Qpro(Qpro(q,r),s),t),u);
W=Qpro(Qpro(Qpro(Qpro(Qpro(q,r),s),t),u),w));

% Determinacion de la posicion de cada una de las articulaciones
% Articulacion 1
V1_t=Qrot(q,p1);
Vr_1=pb+V1_t;
% Articulacion 2
V2_t=Qrot(R,p2);
Vr_2=Vr_1+V2_t;
% Articulacion 3
V3_t=Qrot(S,p3);
Vr_3=Vr_2+V3_t;
% Articulacion 4
V4_t=Qrot(T,p4);
Vr_4=Vr_3+V4_t;
% Articulacion 5
V5_t=Qrot(U,p5);
Vr_5=Vr_4+V5_t;
% Articulacion 6 (Efector final)
V6_t=Qrot(W,p6);
Vr_6=Vr_5+V6_t;

end
```

Apéndice C

Programación de la GUI

```
% FUNCION DEL BOTON ANIMAR
function pushbutton1_Callback(hObject , eventdata , handles)
set(handles.pushbutton2,'Value',1);
axes(handles.VR)
w=vrworld('D:\MATLAB\cuaterniones\abb3Joint');
art1=vrnode(w,'Primer_Eslabon');
art2=vrnode(w,'Segundo_Eslabon');
art3=vrnode(w,'Tercer_Eslabon');
art4=vrnode(w,'Quarto_Eslabon');
art5=vrnode(w,'Quinto_Eslabon');
art6=vrnode(w,'Sexto_Eslabon');
% VARIABLES DE PLOTEO DE LA FIGURA Y POSICION INICIAL(HOME DEL ROBOT)
axes(handles.axes1)
cla;
set(handles.axes1,'XLim',[-873 1225])
set(handles.axes1,'YLim',[-873 1225])
set(handles.axes1,'ZLim',[-455 1800])
title({'MODELO CINEMATICO DIRECTO DEL ROBOT IRB1600';'MEDIANTE CUATERNIONES'},'FontSize'
,9,'Color','k','EdgeColor','k')
xlabel('Eje X','FontSize',8,'Color','B');
ylabel('Eje Y','FontSize',8,'Color','B');
zlabel('Eje Z','FontSize',8,'Color','B');
grid on;
view([30,15]);

% Puntos de los centros rotacionales para el VRML
pb = [ 0, 0, 124.5];
p1 = [ 150, 0, 362];
p2 = [ 0, 0, 700];
p3 = [ 314, 0, 0];
p4 = [ 286, 0, 0];
p5 = [ 42, 0, 0];
p6 = [ 23, 0, 0];
% POSICION DE LOS CENTROS ROTACIONALES DE LAS ARTICULACIONES
cen1=pb/1000;
```

```

cen2=(pb+p1)/1000;
cen3=(pb+p1+p2)/1000;
cen4=(pb+p1+p2+p3)/1000;
cen5=(pb+p1+p2+p3+p4)/1000;
cen6=(pb+p1+p2+p3+p4+p5)/1000;

art1.center=cen1;
art2.center=cen2;
art3.center=cen3;
art4.center=cen4;
art5.center=cen5;
art6.center=cen6;

% CALCULO DE LAS POSICIONES DEL ROBOT PARA CADA UNO DE LOS DESPLAZAMIENTOS
for i = 1:1:tm
    theta1 = th1(i+1);
    theta2 = th2(i+1);
    theta3 = th3(i+1);
    theta4 = th4(i+1);
    theta5 = th5(i+1);
    theta6 = th6(i+1);

% LLAMADA DE LA FUNCION DEL MODELO ROTACIONAL BASADO EN CUATERNIONES
    [p01,p0, p1, p2, p3,p4,p5,p6,q1,q2,q3,q4,q5,q6,q1_2,q2_3,q3_4,q4_5,q5_6]=QRQ(theta1,
    theta2,theta3,theta4,theta5,theta6);
    set(10,'xdata',[p01.x, p0.x],'ydata', [p01.y, p0.y], 'zdata', [p01.z, p0.z]);
    set(11,'xdata',[p0.x, p1.x],'ydata', [p0.y, p1.y], 'zdata', [p0.z, p1.z]);
    set(12,'xdata',[p1.x, p2.x],'ydata', [p1.y, p2.y], 'zdata', [p1.z, p2.z]);
    set(13,'xdata',[p2.x, p3.x],'ydata', [p2.y, p3.y], 'zdata', [p2.z, p3.z]);
    set(14,'xdata',[p3.x, p4.x],'ydata', [p3.y, p4.y], 'zdata', [p3.z, p4.z]);
    set(15,'xdata',[p4.x, p5.x],'ydata', [p4.y, p5.y], 'zdata', [p4.z, p5.z]);
    set(16,'xdata',[p5.x, p6.x],'ydata', [p5.y, p6.y], 'zdata', [p5.z, p6.z]);
    pause(.01)

% VALORES DE LOS ANGULOS PARA LAS ARTICULACIONES DEL VRML

v1= [([q1(2) q1(3) q1(4)]/(1-q1(1)^2)) (2*acos(q1(1)))];
v2= [([q2(2) q2(3) q2(4)]/(1-q2(1)^2)) (2*acos(q2(1)))];
v3= [([q3(2) q3(3) q3(4)]/(1-q3(1)^2)) (2*acos(q3(1)))];
v4= [([q4(2) q4(3) q4(4)]/(1-q4(1)^2)) (2*acos(q4(1)))];
v5= [([q5(2) q5(3) q5(4)]/(1-q5(1)^2)) (2*acos(q5(1)))];
v6= [([q6(2) q6(3) q6(4)]/(1-q6(1)^2)) (2*acos(q6(1)))];

art1.rotation=v1;
art2.rotation=v2;
art3.rotation=v3;
art4.rotation=v4;
art5.rotation=v5;
art6.rotation=v6;
vrdrawnow;

% FUNCION PARO DEL ROBOT
if (get(handles.pushbutton2,'Value')==0)
    i=1;
    break;

```

```

        end
    q1t(i,1:4)=q1';
    r1t(i,1:4)=q1_2';
    s1t(i,1:4)=q2_3';
    t1t(i,1:4)=q3_4';
    Tm(i,1)=i;
end

axes(handles.T_3)
cla;
title('t3','FontSize',8,'Color','k');
hold on; grid on;
plot(Tm,t1t(:,4),'b','LineWidth',2);
% LLAMADA DEL MUNDO VIRTUAL VRML
function VR_CreateFcn(hObject, eventdata, handles)
global w c1
% SELECCION DEL VISOR Y PARAMETROS GLOBALES
vrsetpref('DefaultViewer','internalv5');
axis off
w=vrworld('D:\MATLAB\cuaterniones\abb3Joint');
open(w);
c1=vr.canvas(w,gcf,[299 21 402 312]);
set(c1,'Units','normalized');
end

guidata(hObject,handles);

% CALCULO DE LA CINEMATICA INVERSA.
function pushbutton3_Callback(hObject, eventdata, handles)
x = str2double(get(handles.X6,'string'));
y = str2double(get(handles.Y6,'string'));
z = str2double(get(handles.Z6,'string'));

theta4=str2double(get(handles.T4,'string'));
theta5=str2double(get(handles.T5,'string'));
% CONSTANTES
d1=150;
d2=314;
d3=286;
d4=(cosd(theta5)*65);
L1 = 124;
L2 = 362;
L3 = 700;
d5 = d2+d3+d4;
% VARIABLES
d6=sqrt(x^2+y^2);
dz = 65*sind(theta5)*sind(270+theta4);
dx = 65*sind(theta5)*cosd(270+theta4);
psi = rad2deg(atan2(dx,d6));
phi = rad2deg(atan2(dz,d5));
theta11 = rad2deg(atan2(y,x));

```

```
L4 = d5/cosd(phi);
Lxyb = sqrt(x^2+y^2-dx^2)-d1;
epsilon = atan2((z-(L1+L2)),Lxyb);
C = sqrt((z-(L1+L2))^2 + Lxyb^2);
s = (L3+L4+C)/2;
r = sqrt((s-L3)*(s-L4)*(s-C)/s);
alpha = 2*atan2(r,(s-L4));
gama = 2*atan2(r,(s-C));

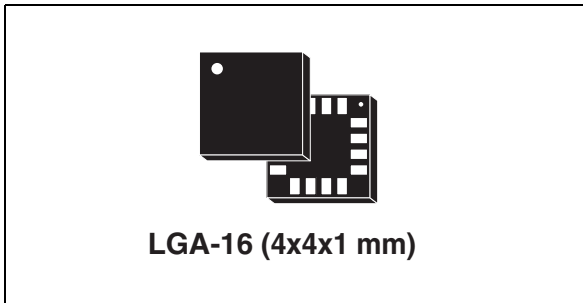
% CALCULO DE LOS ANGULOS
if handles.codo==1
theta1 = theta11 - psi;
theta2 = 90-rad2deg(alpha+epsilon);
theta3 = 90 - rad2deg(gama)+ phi;
```

Apéndice D

Unidad inercial

MEMS motion sensor: three-axis digital output gyroscope

Datasheet - production data



Applications

- Gaming and virtual reality input devices
- Motion control with MMI (man-machine interface)
- GPS navigation systems
- Appliances and robotics

Description

The L3GD20 is a low-power three-axis angular rate sensor.

It includes a sensing element and an IC interface capable of providing the measured angular rate to the external world through a digital interface (I²C/SPI).

The sensing element is manufactured using a dedicated micro-machining process developed by STMicroelectronics to produce inertial sensors and actuators on silicon wafers.

The IC interface is manufactured using a CMOS process that allows a high level of integration to design a dedicated circuit which is trimmed to better match the sensing element characteristics. The L3GD20 has a full scale of $\pm 250/\pm 500/\pm 2000$ dps and is capable of measuring rates with a user-selectable bandwidth.

The L3GD20 is available in a plastic land grid array (LGA) package and can operate within a temperature range of -40 °C to +85 °C.

Features

- Three selectable full scales (250/500/2000 dps)
- I²C/SPI digital output interface
- 16 bit-rate value data output
- 8-bit temperature data output
- Two digital output lines (interrupt and data ready)
- Integrated low- and high-pass filters with user-selectable bandwidth
- Wide supply voltage: 2.4 V to 3.6 V
- Low voltage-compatible IOs (1.8 V)
- Embedded power-down and sleep mode
- Embedded temperature sensor
- Embedded FIFO
- High shock survivability
- Extended operating temperature range (-40 °C to +85 °C)
- ECOPACK[®] RoHS and “Green” compliant

Table 1. Device summary

Order code	Temperature range (°C)	Package	Packing
L3GD20	-40 to +85	LGA-16 (4x4x1 mm)	Tray
L3GD20TR	-40 to +85	LGA-16 (4x4x1 mm)	Tape and reel



LSM303DLHC

Ultra compact high performance e-compass 3D accelerometer and 3D magnetometer module

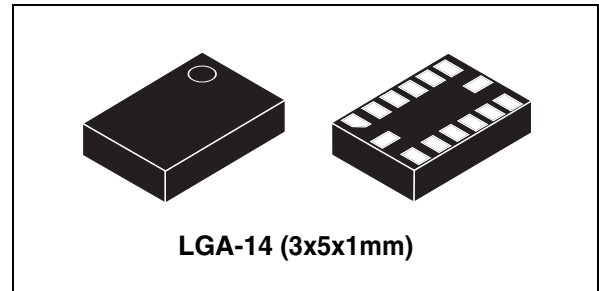
Preliminary data

Features

- 3 magnetic field channels and 3 acceleration channels
- From ± 1.3 to ± 8.1 gauss magnetic field full-scale
- $\pm 2g/\pm 4g/\pm 8g/\pm 16g$ selectable full-scale
- 16 bit data output
- I²C serial interface
- Analog supply voltage 2.16 V to 3.6 V
- Power-down mode/ low-power mode
- 2 independent programmable interrupt generators for free-fall and motion detection
- Embedded temperature sensor
- Embedded FIFO
- 6D/4D orientation detection
- ECOPACK[®] RoHS and “Green” compliant

Applications

- Compensated compass
- Map rotation
- Position detection
- Motion-activated functions
- Free-fall detection
- Click/double click recognition
- Pedometer
- Intelligent power-saving for handheld devices
- Display orientation
- Gaming and virtual reality input devices
- Impact recognition and logging
- Vibration monitoring and compensation



Description

The LSM303DLHC is a system-in-package featuring a 3D digital linear acceleration sensor and a 3D digital magnetic sensor.

LSM303DLHC has linear acceleration full-scales of $\pm 2g / \pm 4g / \pm 8g / \pm 16g$ and a magnetic field full-scale of $\pm 1.3 / \pm 1.9 / \pm 2.5 / \pm 4.0 / \pm 4.7 / \pm 5.6 / \pm 8.1$ gauss. All full-scales available are fully selectable by the user.

LSM303DLHC includes an I²C serial bus interface that supports standard and fast mode 100 kHz and 400kHz. The system can be configured to generate interrupt signals by inertial wake-up/free-fall events as well as by the position of the device itself. Thresholds and timing of interrupt generators are programmable by the end user on the fly. Magnetic and accelerometer parts can be enabled or put into power-down mode separately.

The LSM303DLHC is available in a plastic land grid array package (LGA) and is guaranteed to operate over an extended temperature range from $-40\text{ }^{\circ}\text{C}$ to $+85\text{ }^{\circ}\text{C}$.

Table 1. Device summary

Part number	Temperature range [°C]	Package	Packing
LSM303DLHC	-40 to +85	LGA-14	Tray
LSM303DLHCTR	-40 to +85	LGA-14	Tape and reel

Apéndice E

Recursos computacionales usados en el
modelado por cuaterniones y el modelo
convencional DH

Profile Summary

Generated 28-Feb-2014 12:19:25 using *cpu time*.

Function Name	Calls	Total Time	Self Time*	Total Time Plot (dark band = self time)
Prueba_tiempo	1	0.835 s	0.475 s	
QRQ	1001	0.294 s	0.182 s	
Qpro	5005	0.058 s	0.058 s	
Qrot	6006	0.054 s	0.054 s	
axis	1	0.028 s	0.025 s	
xlabel	2	0.011 s	0.008 s	
view	1	0.008 s	0.002 s	
zlabel	2	0.008 s	0.008 s	
grid	1	0.005 s	0.005 s	
view>ViewCore	1	0.005 s	0.005 s	
ylabel	2	0.005 s	0.005 s	
axescheck	6	0.003 s	0.003 s	
axis>LocSetLimits	1	0.002 s	0.002 s	
axis>allAxes	1	0.001 s	0.001 s	
view>isAxesHandle	1	0.001 s	0.001 s	
deg2rad	2	0.001 s	0.001 s	

Self time is the time spent in a function excluding the time spent in its child functions. Self time also includes overhead resulting from the process of profiling.

Profile Summary

Generated 28-Feb-2014 12:23:03 using cpu time.

Function Name	Calls	Total Time	Self Time*	Total Time Plot (dark band = self time)
Robot	1	1.170 s	0.307 s	
DH	1001	0.801 s	0.242 s	
Tte	7007	0.245 s	0.245 s	
Tal	7007	0.244 s	0.244 s	
Tze	7007	0.041 s	0.041 s	
clf	1	0.036 s	0.003 s	
graphics\private\clo	1	0.033 s	0.004 s	
setdiff	1	0.029 s	0.002 s	
Txe	7007	0.029 s	0.029 s	
setdiff>setdifflegacy	1	0.027 s	0.011 s	
axis	1	0.020 s	0.012 s	
unique	1	0.016 s	0.003 s	
unique>uniquelegacy	1	0.013 s	0.013 s	
axis>LocSetLimits	1	0.008 s	0.007 s	
view	1	0.004 s	0.002 s	
grid	1	0.002 s	0.002 s	
ishold	1	0.001 s	0.001 s	
view>isAxesHandle	1	0.001 s	0.001 s	
view>ViewCore	1	0.001 s	0.001 s	
graphics\private\clo>find_kids	1	0 s	0.000 s	
opaque.double	1	0 s	0.000 s	
refresh	1	0 s	0.000 s	
axis>allAxes	1	0 s	0.000 s	

Self time is the time spent in a function excluding the time spent in its child functions. Self

Referencias

- [1] Barrientos A. Peñín L.F. Baluguer C. Aracil. *Fundamentos de Robótica*. McGraw-Hill, Edificio Valrealty Basaury, 17,1ra planta 28023 Aravaca(Madrid), 1997. ISBN 84-481-0815-9.
- [2] Jack CK Chou. Quaternion kinematic and dynamic differential equations. *Robotics and Automation, IEEE Transactions on*, 8(1):53–64, 1992.
- [3] Villa R Hurtado J Jiménez E. *Modelación y Simulación de un Robot de 3 GDL Usando Cuaterniones y Matlab*. Congreso Nacional de Mecatrónica, Instituto Tecnológico Superior de Cananea, Noviembre, 3 2011.
- [4] Andrew J Hanson. Visualizing quaternions. In *ACM SIGGRAPH 2005 Courses*, page 1. ACM, 2005.
- [5] Jack B Kuipers. *Quaternions and rotation sequences*, volume 66. Princeton university press Princeton, 1999.
- [6] American National Standards Institute. Robot industries association (ansi/ria) standards. <http://www.robotics.org/>. Accesado 19-Agosto-2013.
- [7] Association française de normalisation (afnor). <http://www.afnor.org/>, . Accesado 10-October-2013.
- [8] International federation of robotics (ifr). <http://www.ifr.org/>, . Accesado 10-October-2013.
- [9] Fu k. S. González R. C. Lee C. S. G. *Robótica: Control, Detección, Visión e Inteligencia*. Mc. Graw Hill, 1990.
- [10] Denavit J. Hartenberg R. S. Paul R.P. A kinematic notation for lower-pair mechanisms based on matrices. *J.App. Mech.*

- [11] Lorenzo Sciavicco and Luigi Villani. *Robotics: modelling, planning and control*. Springer, 2009.
- [12] Kumar S. S. *Introducción a la robótica*. McGraw-Hill, 2010.
- [13] Mark W Spong, Seth Hutchinson, and Mathukumalli Vidyasagar. *Robot modeling and control*, volume 3. Wiley New York, 2006.
- [14] Méndez C. M. *Dinámica de cuerpos rígidos con cuaterniones: aplicación a los mecanismos*. Reporte interno No. EGI-001/TDI/MM-01-0498. Escuela de graduados e investigación. Universidad Anáhuac del Sur, S.C. México, 1998 2011. Primera edición. p.ii-iii.
- [15] Janez Funda, Russell H Taylor, and Richard P Paul. On homogeneous transforms, quaternions, and computational efficiency. *Robotics and Automation, IEEE Transactions on*, 6(3):382-388, 1990.
- [16] R Mukundan. Quaternions: From classical mechanics to computer graphics, and beyond. In *Proceedings of the 7th Asian Technology conference in Mathematics*, pages 97-105, 2002.
- [17] Jack CK Chou. Quaternion kinematic and dynamic differential equations. *Robotics and Automation, IEEE Transactions on*, 8(1):53-64, 1992.
- [18] Salmerón Q.B.B. Guerrero C.J.F. *Estimation of robot's attitude: Prediction of the movement via data fusion*. SEPI ESIME UA-SIP 20082294 -ICyTDF-IPN 069, SEPI ESIME IPN, September 2008.
- [19] Klaus Gürlebeck, Uwe Kähler, John Ryan, and Wolfgang Sproßig. *Clifford analysis over unbounded domains*, volume 19. Elsevier, 1997.
- [20] Chou J.C.K. Kamel M. In *Quaternion Approach the solve the Kinematic Equation of the rotation $AaAx = AxAb$ of the sensor-mounted robotic manipulator.*, volume 2 Philadelphia, PA, pages 2656-2662. IEEE International Conference on Robotics and Automation, Apr.24-29 1988.
- [21] Márquez M. M. Modelado cinemático de sistemas mecánicos de cadena abierta. Tesis de grado de maestría en ingeniería., UNAM., Agosto. 1995.

- [22] González J.F. El producto vectorial. 2001. <http://thespectrumofriemannium.wordpress.com/>. Accesado 19-Agosto-2013.
- [23] A.L Reyes. Quaternions: Une representation parametrique systematique des rotations finies. Rapport de Recherche Inria-Rocquencourt Partie 2: Quelques Applications, France, 1990.
- [24] C.M. Mendez. Dinámica de cuerpos rígidos con cuaterniones: aplicación a los mecanismos. Reporte interno No. EGI-001/TDI/MM-01-0498 Primera edición, Universidad Anáhuac del Sur, Escuela de graduados e investigación, S.C. México, 1998.
- [25] ABB. Guía de producto-aplicaciones. <http://www.abb.com/product/seitp327/a34b6be89dd1c941c12572480041c57f.aspx?productLanguage=es&country=MX&tabKey=6>, .
- [26] M.M Márquez. Solución de la cinemática inversa del robot puma 560 mediante cuaterniones. Reporte interno No. EGI-012/TDI/AR-07-0499., Escuela de Graduados e Investigación., Universidad Anáhuac del Sur, S.C., México, 1999.
- [27] ABB. Guía de producto. <http://www.abb.com/cawp/seitp202/3ea389a38ad28553c12575b5003949eb.aspx>, . Accesado 19 de junio 2014.
- [28] ABB. Modelos cad. <http://www.abb.com/product/seitp327/a34b6be89dd1c941c12572480041c57f.aspx?productLanguage=es&country=MX&tabKey=7>, . Accesado 19 de junio 2014.
- [29] ALBERTO MORALES VARELA. *INTERCEPCION EN MOVIMIENTO DE PIEZAS METALICAS QUE NO CUMPLEN CON REQUERIMIENTOS DE CALIDAD*. PhD thesis, 2012.
- [30] W. Kahan. Miscalculating area and angles of a needle-like triangle. <http://www.cs.berkeley.edu/~wkahan/Triangle.pdf>, Marzo 2000. Accesado 30 de junio 2014.
- [31] MathWorks. Virtual reality modeling language (vrml). <http://www.mathworks.es/es/help/sl3d/vrml.html#f2-30062>. Accesado 02 de julio 2014.

- [32] James Diebel. Representing attitude: Euler angles, unit quaternions, and rotation vectors. *Matrix*, 58:15–16, 2006.
- [33] TL Gustavson, P Bouyer, and MA Kasevich. Precision rotation measurements with an atom interferometer gyroscope. *Physical Review Letters*, 78(11):2046, 1997.
- [34] J. Boyd. Good vibrations from eponos's new gyro-sensor. http://www.digikey.com/Web%20Export/Supplier%20Content/EpsonToyocom_114/PDF/epson-good-vibrations.pdf?redirected=1. Acceso 02 de julio 2014.
- [35] Minoru Kurosawa, Yoshimitsu Fukuda, Masaya Takasaki, and Toshiro Higuchi. A surface-acoustic-wave gyro sensor. *Sensors and Actuators A: Physical*, 66(1):33–39, 1998.
- [36] Johann Borenstein and Liqiang Feng. Gyrodometry: A new method for combining data from gyros and odometry in mobile robots. In *Robotics and Automation, 1996. Proceedings., 1996 IEEE International Conference on*, volume 1, pages 423–428. IEEE, 1996.
- [37] Raymond Serway and John Jewett. *Physics for scientists and engineers*. Cengage Learning, 2013.
- [38] Henk J Luinge and Peter H Veltink. Measuring orientation of human body segments using miniature gyroscopes and accelerometers. *Medical and Biological Engineering and computing*, 43(2):273–282, 2005.
- [39] John L Crassidis, Kok-Lam Lai, and Richard R Harman. Real-time attitude-independent three-axis magnetometer calibration. *Journal of Guidance, Control, and Dynamics*, 28(1):115–120, 2005.
- [40] R. A. Langel. The main field. *Geomagnetism*, edited by J. A. Jacobs, (1):249–512, 1987.
- [41] R Raya, R Ceres, E Rocon, A Frizera, and JL Pons. Comunicador inercial para potenciar la autonomía de niños con parálisis cerebral en el uso del computador.

-
- [42] Olivier Company, Sebastien Krut, and François Pierrot. Internal singularity analysis of a class of lower mobility parallel manipulators with articulated traveling plate. *Robotics, IEEE Transactions on*, 22(1):1–11, 2006.
- [43] Arduino. Plataforma arduino. <http://www.arduino.cc/es/>. Accesado 01 de julio 2014.
- [44] Pololu. Pololu minimu-9 + arduino ahrs (attitude and heading reference system). <https://github.com/pololu/minimu-9-ahrs-arduino>. Accesado 01 de julio 2014.