

**DESARROLLO DE UN MODELO MATEMÁTICO
PARA SIMULAR LA DINÁMICA Y LA
GENERACIÓN DE TRAYECTORIAS DE UN
MANIPULADOR CON CINCO GRADOS DE
LIBERTAD**

**JAVIER FERNANDO COVELLI CADENA
FREDY ALONSO DUARTE SANGUINO**

**UNIVERSIDAD INDUSTRIAL DE SANTANDER
FACULTAD DE INGENIERÍAS FÍSICO - MECÁNICAS
ESCUELA DE INGENIERÍA MECÁNICA
BUCARAMANGA**

2.005

**DESARROLLO DE UN MODELO MATEMÁTICO PARA
SIMULAR LA DINÁMICA Y LA GENERACIÓN DE
TRAYECTORIAS DE UN MANIPULADOR CON CINCO
GRADOS DE LIBERTAD**

**JAVIER FERNANDO COVELLI CADENA
FREDY ALONSO DUARTE SANGUINO**

**Trabajo de Grado para optar al título de
Ingeniero Mecánico**

Director

**JABID EDUARDO QUIROGA MÉNDEZ
Ingeniero Mecánico**

**UNIVERSIDAD INDUSTRIAL DE SANTANDER
FACULTAD DE INGENIERÍAS FÍSICO - MECÁNICAS
ESCUELA DE INGENIERÍA MECÁNICA
BUCARAMANGA**

2.005

DEDICATORIA

A mi Madre, razón de mis esfuerzos.

Guido Javier Fernando Covelli Cadena

DEDICATORIA

A quien con su invaluable esfuerzo hace posible la realización de mis metas,
mi Madre.

Fredy Alonso Duarte Sanguino

AGRADECIMIENTOS

A Dios Todopoderoso por permitirme alcanzar un nuevo logro.

A Jabid Eduardo Quiroga, ingeniero mecánico, director del proyecto y amigo, por su respaldo, confianza y colaboración oportuna.

Fredy Alonso Duarte Sanguino

CONTENIDO

	pág.
INTRODUCCIÓN	1
1. FUNDAMENTOS DE ROBÓTICA	4
1.1. DESCRIPCIÓN DE LA POSICIÓN Y ORIENTACIÓN	4
1.1.2. Descripción De Una Posición	6
1.1.3 Descripción De Una Orientación	8
1.1.4 Descripción De Un Marco	10
1.1.5 Relaciones Entre Diferentes Marcos	10
1.1.6 Ecuaciones De Transformación	12
1.2. DESCRIPCIÓN DEL MANIPULADOR ROBOT ARM MR 999	13
1.2.1. Mecanismo	13
2. MODELO CINEMÁTICO	15
2.1. CINEMÁTICA DIRECTA	15
2.1.1. Descripción De Las Conexiones Entre Enlaces	15
2.1.2. Derivación De Las Transformaciones	18
2.1.3. Aplicación Al Robot Arm Mr 999 E	21
2.2. CINEMÁTICA INVERSA	23
2.2.1. Solucionabilidad	24
2.2.2. Método De Solución	25
2.2.3. Solución Algebraica Del Manipulador Robot Arm	26
2.3. JACOBIANO DEL MANIPULADOR	33

3. CINÉTICA DEL MANIPULADOR	37
3.1. CONSIDERACIONES GENERALES SOBRE LA CINÉTICA	38
3.2. GENERALIDADES DE LOS MÉTODOS CONVENCIONALES	41
3.3. ARTICULACIÓN SIMPLE DE ROTACIÓN	43
3.3.1. Generalizando A n Articulaciones	44
3.4. OBTENCIÓN DEL MODELO DINÁMICO MEDIANTE FORMULACIÓN ITERATIVA DE NEWTON-EULER	44
3.4.1. Ecuaciones Básicas	44
3.4.2. El Algoritmo Iterativo De Newton-Euler	49
3.5. PARÁMETROS FÍSICOS Y GEOMÉTRICOS DEL ROBOTARM	50
3.6. MODELO DINÁMICO DEL ROBOTARM MR 999	52
3.6.1. Dinámica Directa	52
3.6.2. Dinámica Inversa	56
4. GENERACIÓN DE TRAYECTORIAS	64
4.1. CONSIDERACIONES GENERALES	64
4.2. ESTRATEGIAS PARA LA GENERACIÓN DE TRAYECTORIAS	65
4.3. GENERACIÓN DE TRAYECTORIAS PARA MANIPULADORES	67
4.3.1. Planeado En El Espacio Cartesiano	67
4.3.2. Planeado En El Espacio Articular	72
4.4. GENERACIÓN DE TRAYECTORIAS EMPLEANDO POLINOMIOS DE QUINTO GRADO	74
CONCLUSIONES	78
RECOMENDACIONES	79
BIBLIOGRAFÍA	80
ANEXOS	82

LISTA DE FIGURAS

	pág.
Figura 1. 1. Sistemas de coordenadas atados al manipulador	5
Figura 1. 2. Vector de posición relativo a un marco de referencia {A}	7
Figura 1. 3. Ubicación de un objeto mediante su posición y orientación	8
Figura 1. 4. Transformación general de un vector	11
Figura 1. 5. Conjunto de transformaciones formando un bucle	12
Figura 1. 6. Vista real del manipulador ROBOT ARM MR 999	14
Figura 1. 7. Rango de movimiento del ROBOT ARM	14
Figura 2. 1. Representación de los sistemas de coordenadas	18
Figura 2. 2. Ubicación de los marcos intermedios {P}, {Q}, y {R}.	19
Figura 2. 3. Representación de una orientación	32
Figura 3.1. Articulación de rotación	41
Figura 3.2. Manipulador con una articulación	42
Figura 3.3. Balance de fuerzas en dos articulaciones consecutivas	46
Figura 3.4. Vista de la herramienta del CAD	51
Figura 3.5. Diagrama de bloques en Simulink para torques constantes	54
Figura 3.6. Subsistema <i>Colisiones</i> .	55
Figura 3.7. Diagrama de bloques para entrada de torques variables	56
Figura 3.8. Subsistema <i>Trayectorias articulares</i> .	56
Figura 4.1. Trayectoria con puntos intermedios inalcanzables	71
Figura 4.2. Velocidades altas cerca de las singularidades	72
Figura 4.3. Puntos alcanzables con algunas orientaciones	73

LISTA DE TABLAS

	pág.
Tabla 1. Rango de movimiento del RobotArm999	14
Tabla 2. Parámetros de Denavit-Hartenberg del RobotArm999	18
Tabla 3. Parámetros físicos y geométricos del RobotArm999	51

LISTA DE ANEXOS

	pág.
Anexo A. COMO UTILIZAR LA INTERFASE PARA SIMULAR EL ROBOT ARM MR999	83
Anexo B. ANIMACIÓN EN ROBOWORKS	93

NOMENCLATURA O GLOSARIO

NOTACIÓN

${}^A P$: Posición de un punto respecto al sistema de coordenadas {A}.

\hat{X} : Vector unitario en la dirección x de un sistema de coordenadas.

\hat{Y} : Vector unitario en la dirección y de un sistema de coordenadas.

\hat{Z} : Vector unitario en la dirección z de un sistema de coordenadas.

${}^A R_B$: Matriz de rotación de un marco {B} respecto al marco {A}.

${}^A P_{BORG}$: Posición del origen del marco {B} respecto al origen del marco {A}.

${}^A T_B$: Matriz de transformación homogénea de {B} respecto a {A}.

A1: Longitud del primer enlace del manipulador ROBOT ARM MR.

A2: Longitud del segundo enlace del manipulador ROBOT ARM MR.

A3: Longitud del tercer enlace del manipulador ROBOT ARM MR.

a_i : Parámetro de Denavit-Hartenberg para manipuladores.

d_i : Parámetro de Denavit-Hartenberg para manipuladores.

s: contracción para denominar la función seno.

c: contracción para denominar la función coseno.

$[A]^{-1}$: Inversas de la matriz $[A]$.

${}^A R_{XYZ}$: Matriz de rotación de {B} respecto a {A} según los ángulos fijos.

${}^A J$: Jacobiano del manipulador referido al marco {A}.

${}^A v$: Velocidad lineal de un cuerpo relativa al marco {A}.

${}^A \omega$: Velocidad angular de un cuerpo relativa al marco {A}.

LETRAS GRIEGAS

γ : Primer ángulo de rotación en la representación de ángulos fijos.

β : Segundo ángulo de rotación en la representación de ángulos fijos.

α : Tercer ángulo de rotación en la representación de ángulos fijos.

α_i : Parámetro de Denavit-Hartenberg para manipuladores.

θ_i : Parámetro de Denavit-Hartenberg para manipuladores.

θ'_i : Primera derivada de la variable articular de posición θ_i

θ''_i : Segunda derivada de la variable articular de posición θ_i .

SUBINDICES

x : referido al eje x de un sistema de coordenadas.

y : referido al eje y de un sistema de coordenadas.

z : referido al eje z de un sistema de coordenadas.

A: referido al sistema de coordenadas {A}.

B: referido al sistema de coordenadas {B}.

i : referido al elemento (enlace o articulación) de un manipulador.

RESUMEN

TÍTULO:

DESARROLLO DE UN MODELO MATEMÁTICO PARA SIMULAR LA DINÁMICA Y LA GENERACIÓN DE TRAYECTORIAS DE UN MANIPULADOR CON CINCO GRADOS DE LIBERTAD*

AUTORES:

Javier Fernando Covelli Cadena
Fredy Alonso Duarte Sanguino**

PALABRAS CLAVES:

Robótica, manipulador, MATLAB, simulink, Roboworks, simulación, dinámica, generación de trayectorias.

DESCRIPCIÓN:

Cuando se inicia con el estudio de manipuladores robóticos, surgen varios interrogantes fundamentales: ¿Cuál es la posición de su extremo con respecto a un sistema de referencia conocido? ¿Cuál es la velocidad de dicho extremo? ¿Cuál es su aceleración? ¿Cuál es la trayectoria que debe seguir para llegar a un objetivo de ubicación conocida?

Este proyecto proporciona las respuestas a estos interrogantes para el caso del manipulador RobotArm – MR 999.

Se obtuvieron los modelos matemáticos que permiten simular la dinámica (cinemática directa, inversa y cinética) y la generación de trayectorias del manipulador mencionado. Se introdujeron estos modelos en el software MATLAB-SimuLink y se construyeron algoritmos para encontrar soluciones numéricas al modelo desarrollado. Elaborando, además, una interfaz gráfica amigable desde la cual un usuario puede ingresar datos para realizar la simulación con cualquiera de los modelos desarrollados y a su vez observar los resultados obtenidos en forma numérica, gráfica y de manera continua. Se realizó también el modelamiento gráfico del manipulador en los CAD, Solid Edge V12 y RoboWorks que permite una interacción amigable con el usuario y con otros programas, para nuestro caso, MATLAB.

* Trabajo de grado

** Facultad de Ingenierías Físico-Mecánicas, Escuela de Ingeniería Mecánica, Ing. Jabid Quiroga

SUMMARY

TITLE:

DEVELOPMENT OF A MATHEMATICAL MODEL TO SIMULATE THE DYNAMICS AND THE GENERATION OF TRAJECTORIES OF A MANIPULATOR WITH FIVE DEGREES OF FREEDOM*

AUTHORS:

Javier Fernando Covelli Cadena
Freyd Alonso Duarte Sanguino**

KEY WORDS:

Robotics, manipulator, MATLAB, simulink, Roboworks, simulation, dynamics, trajectory generation.

DESCRIPTION:

When one begins with the study of robotics manipulators, several fundamental queries arise: Which is the position of its end respect to a known reference system? Which is the speed of this end? Which is its acceleration? Which is the trajectory that should continue to achieve a well-known location target?

This project provides the answers to these queries for the case of the manipulator RobotArm. MR 999.

All Objectives were archived, between them the mathematical models that allow simulating the dynamics (direct, inverse kinematics and kinetics) and the trajectories' generation of the aforementioned manipulator. These models were introduced in the software MATLAB-SimuLink and the algorithms were built to find numeric solutions to the developed model. Elaborating, also, a friendly interface from which an user can enter data to carry out the simulation with any of the developed models and in return to observe the results obtained in a numeric, graphic and continuously way. Also was made the graphic model of the manipulator in the CAD, Solid Edge V12 and RoboWorks that it allows a friendly interaction with the user and with other programs, in our case, MATLAB.

* Degree Work.

** Physical-Mechanical Engineering Faculty, Mechanical Engineering, Eng. Jabid Quiroga

INTRODUCCIÓN

Son las universidades, los entes llamados a generar desarrollos tecnológico-científicos que jalonan la industria hacia las sendas del progreso productivo, y estando plenamente convencidos de la vital importancia de la mecatrónica en el desarrollo de toda industria, además de la sorprendente relevancia de los manipuladores en ella, podemos decir que su estudio es por demás justificado y necesario; teniendo en cuenta los más recientes cambios en el plan de estudios de la carrera de Ingeniería Mecánica y previendo la inminente inclusión de asignaturas relacionadas con la mecatrónica. El desarrollo de este proyecto puede ser utilizado como referencia tangible para tales asignaturas y su resultado también podrá servir como punto de partida para la implementación del sistema de control y mando del manipulador estudiado.

Conocer la posición, velocidad y aceleración del extremo del manipulador engloba con suficiencia las características dinámicas de éste, necesarias para su comprensión y óptimo aprovechamiento del mismo. Planear las trayectorias nos permitirá determinar y delimitar las posibilidades reales en cuanto a sus aplicaciones, además de asegurar que dichas trayectorias se ejecuten en forma adecuada.

En este orden de ideas, el **Capítulo 1**, presenta unas nociones básicas sobre robótica, en lo que respecta a la forma de representar posiciones y orientaciones de un sistema de coordenadas al que denominaremos “Marco” en relación a un sistema de coordenadas de referencia; además de introducir la notación que se seguirá a lo largo del trabajo.

El **Capítulo 2**, aborda el tema de la cinemática, así pues, trata con la descripción analítica del desplazamiento espacial del robot como función del tiempo, en particular las relaciones entre las variables espaciales de tipo articulación y la posición y orientación del efector final del robot. En este capítulo se plantean dos cuestiones fundamentales, ambas de interés teórico y práctico en la cinemática del robot; la cinemática directa, que trata el problema de conocer la posición del efector final, teniendo como entrada las posiciones articulares, y la cinemática inversa que obtiene las posiciones articulares partiendo de una posición y orientación del efector final conocida.

El **Capítulo 3** trata el problema de la **cinética** de manipulador, es decir, incluye un conjunto de ecuaciones matemáticas que describen su conducta dinámica. Cabe indicar que las ecuaciones que representan el movimiento del brazo articulado son ecuaciones diferenciales no lineales acopladas, para las que, en un caso general, resulta difícil obtener soluciones analíticas. Físicamente, los términos de acoplamiento representan: pares gravitacionales que dependen de la posición de las articulaciones, pares de reacción debidos a las aceleraciones en otras articulaciones, y pares debidos a la aceleración de Coriolis y fuerzas centrífugas.

El **Capítulo 4** abarca el tema de la **generación de trayectorias** para el manipulador, empleando específicamente polinomios de grado 5 (quínticos) que garantizan funciones de ejecución suaves para cada articulación del manipulador.

Cada capítulo describe el procedimiento solución del problema planteado y explica brevemente como pueden emplearse los útiles computacionales anexados a este trabajo que funcionan bajo MATLAB 7, Solid Edge V12 y

RoboWorks², que pueden ser utilizados por el interesado permitiendo ingresar datos para realizar la simulación con cualquiera de los modelos desarrollados y a su vez observar los resultados obtenidos en forma numérica, gráfica³ y de manera continua. El anexo incluye una ayuda para el empleo de estos útiles computacionales, así como la sintaxis de las funciones utilizadas.

² El instalador de RoboWorks Demo Version 2.0 está incluido como anexo.

³ Apoyada en el modelo 3D desarrollado del RobotArm MR999 y en las funciones de graficación de MATLAB

1. FUNDAMENTOS DE ROBÓTICA

El presente capítulo introduce alguna terminología y muestra brevemente algunos tópicos que es necesario conocer para una mejor comprensión del trabajo presentado en esta tesis. La gran mayoría de los robots industriales en la actualidad son brazos articulados, por lo cual la totalidad de definiciones que se harán de aquí en adelante se referirán a este tipo de robots y en algunos casos mas exactamente se referirán al modelo particular del manipulador robótico ROBOT ARM MR 999, objeto del presente estudio.

1.1. DESCRIPCIÓN DE LA POSICIÓN Y ORIENTACIÓN

En el estudio de la robótica estamos frecuentemente interesados con la ubicación de objetos en el espacio tridimensional. Estos objetos son los enlaces del manipulador, las piezas y herramientas con las cuales este debe interactuar, y otros objetos encontrados en el ambiente del manipulador. A un bajo pero importante nivel estos objetos pueden ser descritos por solo dos atributos, su posición y su orientación. Naturalmente, un tema de nuestro interés, es la manera en la cual nosotros representamos estas cantidades y cómo las podemos manipular matemáticamente.

Tendiendo a describir la posición y orientación de un cuerpo en el espacio, siempre consideraremos un sistema de coordenadas, o marco, rígidamente atado al objeto de interés, y luego procedemos a describir la posición y orientación de este marco respecto a otro sistema de coordenadas tomado como referencia, este marco de referencia por lo general es uno atado a la base fija del manipulador (ver figura 1.1), como será nuestro caso. Debido a que cualquier marco puede servir como sistema de referencia para expresar la

posición y orientación de un cuerpo, podemos pensar en alguna manera de transformar o cambiar la descripción de la posición y orientación de un cuerpo desde un marco a otro.

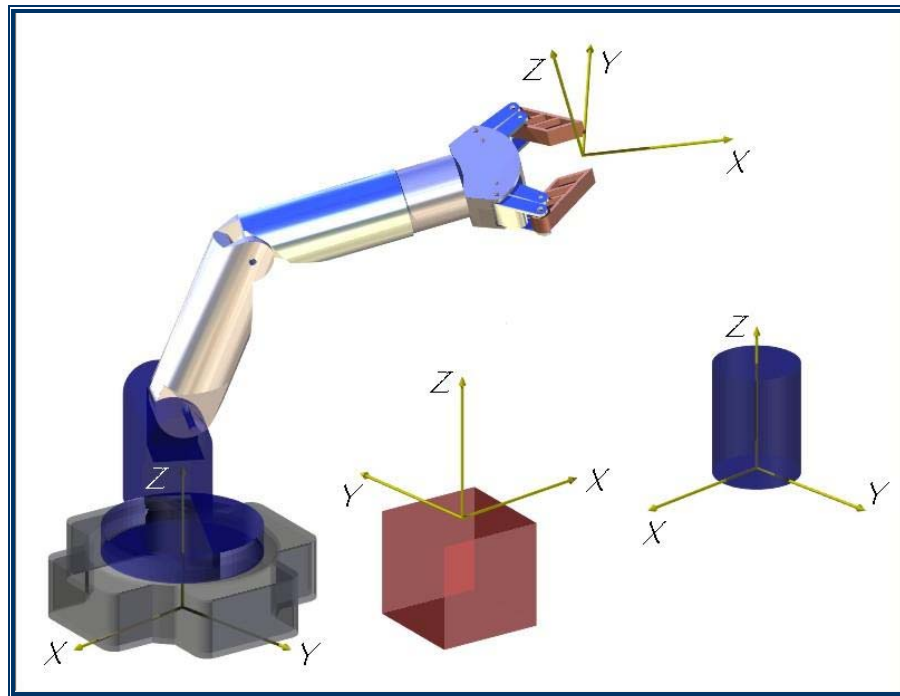


Figura 1. 1. Sistemas de coordenadas o “marcos” atados al manipulador y los objetos en su ambiente

Un manipulador consiste de elementos rígidos consecutivos llamados enlaces conectados por medio de juntas que permiten el movimiento relativo de enlaces consecutivos. Estas juntas son usualmente instrumentadas con sensores de posición que permiten medir la posición relativa entre enlaces consecutivos.

En cuanto al número de grados de libertad de un manipulador, se dice que este es el número de variables de posición independientes que deberán ser especificadas si se quiere determinar la ubicación de todas las partes del manipulador. Este es un término general usado para cualquier mecanismo.

En el caso típico de robots industriales, debido a que un manipulador es una cadena cinemática abierta, y debido también a que la posición de cada junta es usualmente definida con una sola variable, el número de juntas es igual al número de grados de libertad.

Por otra parte, en el extremo de la cadena de enlaces que representa al manipulador se encuentra el elemento denominado efector final. Dependiendo de la aplicación deseada para el manipulador, el efector final puede ser un gripper o pinza, como en el caso del Robot Arm MR 999, o en otros casos puede tratarse de elementos soldadores, de pintura, etc. Generalmente la descripción de la posición del manipulador se hace describiendo la posición y orientación del sistema de coordenadas o marco atado al efector final, relativo al marco atado a la base fija del manipulador, como se hace en el presente trabajo.

1.1.2. DESCRIPCIÓN DE LA POSICIÓN

Una vez que un sistema de coordenadas es establecido, podemos localizar cualquier punto en el espacio con un vector 3×1 llamado vector de posición. debido a que frecuentemente se definen muchos sistemas de coordenadas dentro del mismo espacio los vectores de posición deben ser dotados con la información concerniente de a que sistema de coordenadas está referido el vector. Aquí se adoptara la convención de suministrar esta información con un superíndice que antecede al nombre del vector. Por ejemplo, ${}^A P$ mostrado en la figura 1.2. Esto significa que las componentes de ${}^A P$ son valores numéricos que indican distancias a lo largo de los ejes de $\{A\}$. Cada una de

estas distancias a lo largo de un eje es el resultado de proyectar el vector sobre los ejes correspondientes.

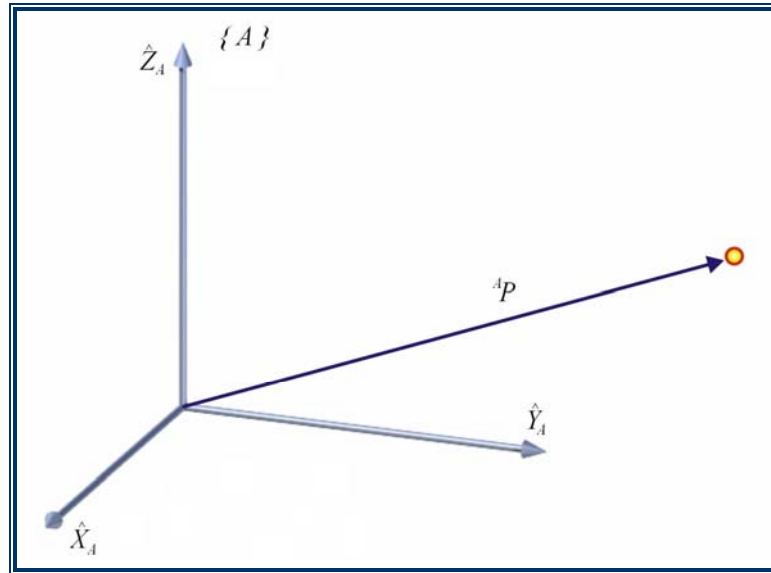


Figura 1. 2. Vector de posición relativo a un marco de referencia {A}

La Figura 1.2. representa imaginariamente un sistema de coordenadas, {A}, con tres ejes mutuamente perpendiculares representados por sus vectores unitarios. Un punto ${}^A P$ es representado con un vector que puede ser visto como una posición en el espacio o simplemente como un conjunto ordenado de tres números. Los elementos individuales de un vector son dados con subíndices, así:

$${}^A P = \begin{bmatrix} P_x \\ P_y \\ P_z \end{bmatrix} \quad (1.1)$$

En resumen, nosotros describiremos la posición de un punto en el espacio con un vector posición, cuyas componentes representan las distancias que

debemos desplazarnos a lo largo de los ejes principales del marco de referencia para llegar al punto.

1.1.3 DESCRIPCIÓN DE LA ORIENTACIÓN

Será necesario no solo representar un punto en el espacio, sino también describir la orientación de un cuerpo en el espacio. Por ejemplo, si el vector ${}^A P$ en la figura 1.3 ubica el punto directamente entre los dedos de el gripper del manipulador, la ubicación completa de este gripper no estará especificada hasta que se de la orientación. Asumiendo que un manipulador tiene un número suficiente de juntas, el gripper puede ser orientado arbitrariamente mientras mantiene sus dedos en la misma posición en el espacio. En orden a describir la orientación de un cuerpo nosotros ataremos un sistema de coordenadas a dicho cuerpo y entonces daremos la descripción de este sistema de coordenadas respecto a un sistema de referencia. En la figura 1.3, se muestra el sistema de coordenadas $\{B\}$ atado al cuerpo en una forma conocida. Ahora, una descripción de $\{B\}$ relativa a $\{A\}$, será suficiente para dar la orientación del cuerpo.

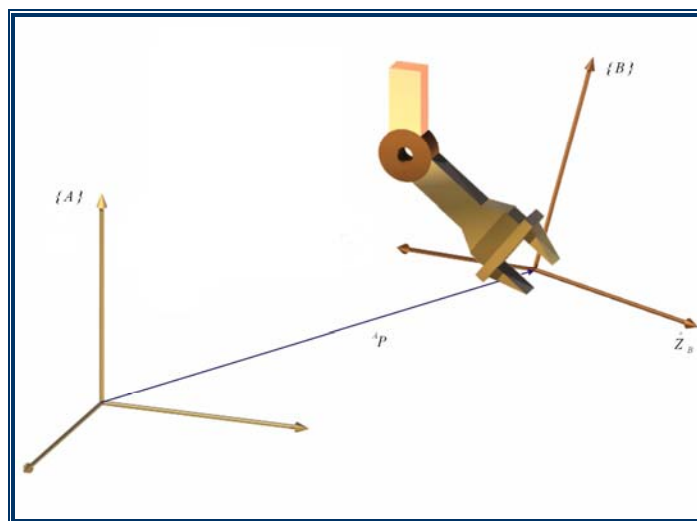


Figura 1. 3. Ubicación de un objeto mediante su posición y orientación

Así, posiciones de puntos serán descritas con vectores, y orientaciones de cuerpos serán descritas con sistemas de coordenadas. Una forma de describir el sistema de coordenadas {B} atado al cuerpo, es escribir los vectores unitarios de sus tres ejes principales en términos del sistema de coordenadas {A}. Se denotarán los vectores unitarios dando las direcciones principales del sistema de coordenadas {B} como $\hat{X}_B, \hat{Y}_B, \text{y } \hat{Z}_B$. Cuando se escriban en términos del sistema de coordenadas {A}, serán llamados ${}^A\hat{X}_B, {}^A\hat{Y}_B, \text{y } {}^A\hat{Z}_B$.

Será conveniente si agrupamos los tres vectores unitarios anteriores como las columnas de una matriz 3x3 en su respectivo orden. Llamaremos esta matriz, una matriz de rotación, y debido a que esta matriz de rotación en particular describe al marco {B} relativa al {A} será nombrada con la notación ${}^A_B R$. La selección de los subíndices y superíndices en la definición de las matrices de rotación se hará mas clara en las siguientes secciones. De acuerdo a lo anterior nosotros escribiremos:

$${}^A_B R = \left[{}^A\hat{X}_B, {}^A\hat{Y}_B, {}^A\hat{Z}_B \right] = \begin{bmatrix} r_{11}, r_{12}, r_{13} \\ r_{21}, r_{22}, r_{23} \\ r_{31}, r_{32}, r_{33} \end{bmatrix} \quad (1.2)$$

En resumen, un conjunto de tres vectores puede ser usado para especificar una orientación. Por conveniencia, construiremos la anteriormente mencionada matriz 3x3 de rotación la cual tiene estos tres vectores como sus columnas. De acuerdo a todo lo anterior, la posición de un punto será representada con un vector, y la orientación de un cuerpo con una matriz. Debido a que el producto punto de dos vectores unitarios proporciona el coseno del ángulo entre estos, es claro el por qué los componentes de una matriz de rotación son frecuentemente llamados como cosenos directores.

1.1.4 DESCRIPCIÓN DE UN MARCO

La información necesaria para especificar completamente la ubicación del efector final de un manipulador es una posición y una orientación. El punto sobre el gripper cuya posición describirá la del efector final del manipulador podrá ser seleccionado arbitrariamente, sin embargo: Por conveniencia, el punto cuya posición se describirá será seleccionado como el origen del marco atado al efector final del manipulador. Debido a que la pareja posición – orientación es muy frecuente en robótica, se definirá un **marco** como un conjunto de cuatro vectores dando información sobre posición y orientación de un objeto. Nótese que un marco es un sistema de coordenadas, donde adicionalmente a la orientación, damos un vector posición el cual localiza su origen en relación al origen de otro marco.

1.1.5 RELACIONES ENTRE DIFERENTES MARCOS

Muy a menudo, nosotros conocemos la descripción de un vector con respecto a algún marco {B}, y deseamos conocer su descripción con respecto a otro marco {A}. Generalmente el origen del marco {B} no es coincidente con el origen de {A}, pero un vector posición relativa de {B} a {A} es dado. El vector que localiza al origen de {B} es llamado ${}^A P_{BORG}$. Además {B} está rotado con respecto a {A} en una forma descrita por la matriz de rotación ${}^A_B R$. De acuerdo a lo anterior, dado ${}^B P$, nosotros podemos calcular ${}^A P$, como es visto en la figura 1.4. Para lo anterior, primero que todo podemos cambiar ${}^B P$ a su descripción relativa a un marco intermedio que tenga la misma orientación que el marco {A} pero cuyo origen sea coincidente con el del marco {B}. Esto se hace premultiplicando ${}^B P$ por ${}^A_B R$. Para completar el

proceso, podemos describir la traslación entre los orígenes de {A} y el marco intermedio simplemente sumando vectores, obteniendo en definitiva:

$${}^A P = {}^A R * {}^B P + {}^A P_{BORG} \quad (1.3)$$

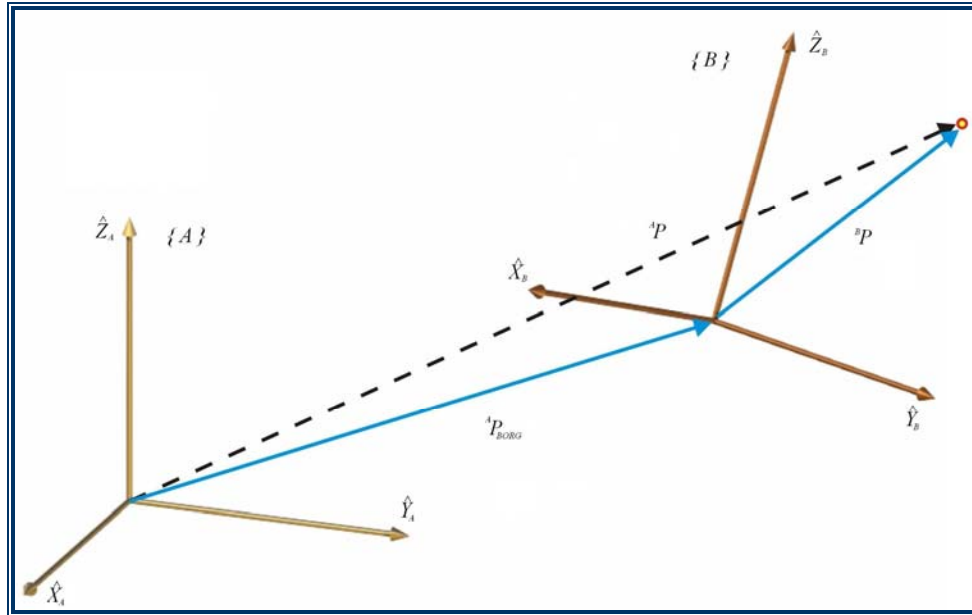


Figura 1. 4. Transformación general de un vector

La forma de la ecuación (1.3) no es tan agradable como la forma conceptual

$${}^A P = {}^A T * {}^B P \quad (1.4)$$

donde podemos pensar en la transformación de un marco a otro como un operador en forma de matriz. Usando ahora vectores de posición 4x1 podemos definir este operador, al que llamaremos **matriz de transformación homogénea**, de la siguiente manera:

$$\begin{bmatrix} {}^A P \\ \dots \\ 1 \end{bmatrix} = \begin{bmatrix} {}^A R & | & {}^A P_{BORG} \\ \dots & \dots & \dots \\ 0 & 0 & 0 & | & 1 \end{bmatrix} \begin{bmatrix} {}^B P \\ \dots \\ 1 \end{bmatrix} \quad (1.5)$$

Esto es:

- ✓ Un "1" es adicionado como el ultimo elemento de los vectores de posición 4x1.
- ✓ Una fila "[0 0 0 1]" es adicionada como la ultima fila de la matriz 4x4.

1.1.6 ECUACIONES DE TRANSFORMACIÓN

La figura 1.5 muestra una situación en la cual un marco {D} puede ser expresado como productos de transformaciones en dos formas diferentes.

Primero,

$${}^U_D T = {}^U_A T * {}^A_D T \quad (1.6)$$

o, también puede expresarse como:

$${}^U_D T = {}^U_B T * {}^B_C T * {}^C_D T \quad (1.7)$$

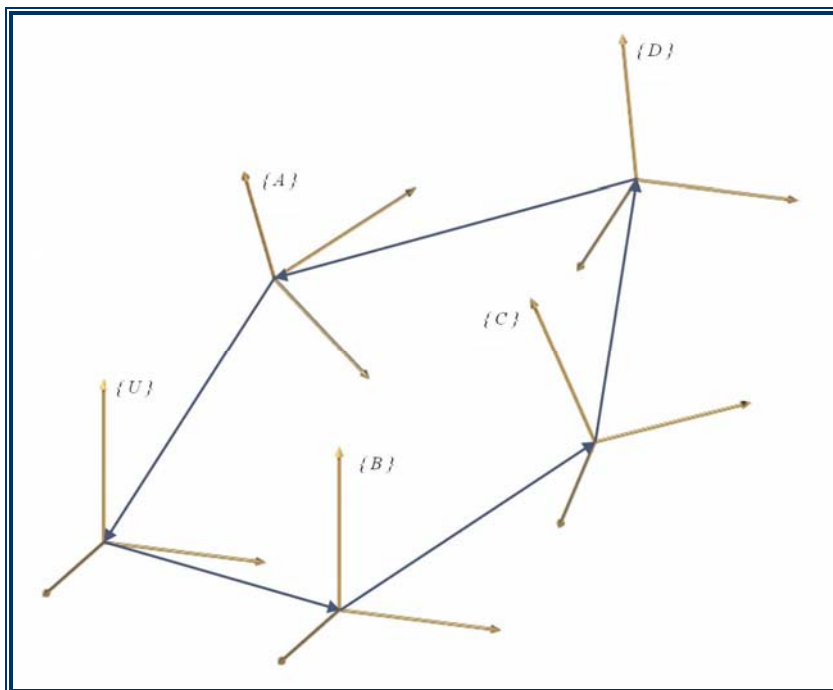


Figura 1. 5. Conjunto de transformaciones formando un bucle

Podemos unir las dos anteriores descripciones de ${}^U_D T$, igualarlas y formar una **ecuación de transformación**:

$${}^U_A T * {}^A_D T = {}^U_B T * {}^B_C T * {}^C_D T \quad (1.8)$$

Las ecuaciones de transformación pueden ser usadas para encontrar soluciones en el caso de tener n transformaciones desconocidas y n ecuaciones de transformación.

1.2. DESCRIPCIÓN DEL MANIPULADOR ROBOT ARM MR 999

El objeto de estudio en este trabajo es el manipulador Robot Arm Mr 999 el cual se describirá brevemente a continuación, nombrando solo los parámetros relevantes a la hora de tomarlo como aplicación en los diferentes tópicos concernientes a su simulación.

1.2.1. MECANISMO

ROBOT ARM tiene cinco movimientos separados: agarrar / liberar, levantar / bajar (2), rotar la muñeca y el pivote (de lado), que son controlados por cinco interruptores ubicados en su controlador como puede observarse en la figura 1.6. El ROBOT ARM es ideal para introducirnos en los fundamentos de la robótica con miras a la simulación. Para efectos de su simulación se considerará al ROBOT ARM como un sistema mecánico formado por cuatro articulaciones, todas de rotación enlazadas en cadena abierta por medio de tres elementos rígidos que forman el brazo del manipulador y finalizándose con un gripper ubicado en el extremo del enlace que actúa como efector final.



Figura 1. 6. Vista real del manipulador ROBOT ARM MR 999

En cuanto al rango de movimiento del ROBOT ARM MR 999 respecto a sus cinco ejes (uno para cada elemento), se tiene (ver Tabla 1 y Figura 1.7):

	Base	Hombro	Codo	Prona	Gripper
Rango	0→350°	-90°→30°	-90°→45°	0→340°	0→50mm

Tabla 1. Rango de movimiento del RobotArm999

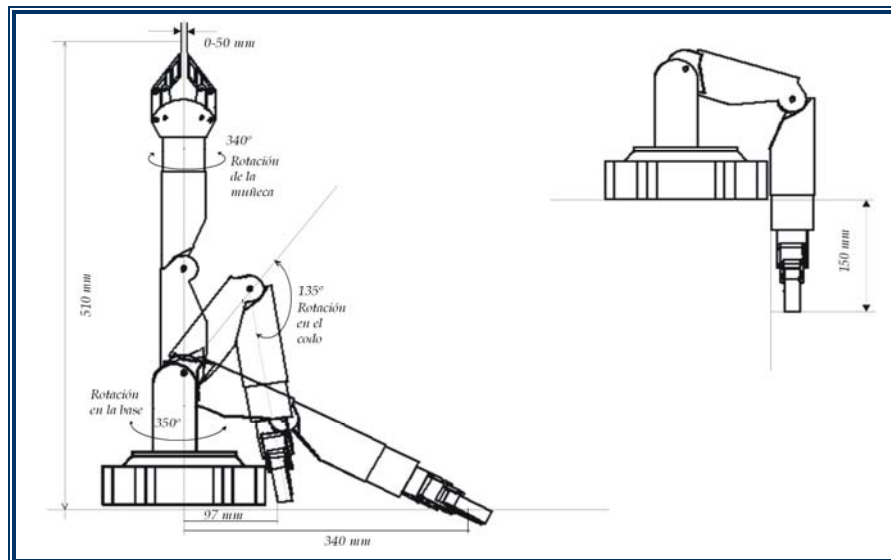


Figura 1. 7. Rango de movimiento del ROBOT ARM

2. MODELO CINEMÁTICO

En este capítulo se desarrollan los modelos matemáticos referentes a la cinemática directa e inversa del manipulador Robot Arm Mr 999. Estos modelos se emplean tanto para la simulación elaborada en este trabajo, así como para el control, campo que se espera sea estudiado y aplicado en un futuro cercano ya que con el presente trabajo están dadas las condiciones para su implementación.

2.1. CINEMÁTICA DIRECTA

Cinemática es la ciencia que trata el movimiento sin involucrar las fuerzas que lo producen, por lo cual el tema central de esta sección es un método para calcular la posición y orientación del efector final del manipulador relativo a la base del mismo como una función de las variables articulares, es decir, de los ángulos alcanzados por cada una de sus articulaciones.

2.1.1 DESCRIPCIÓN DE LAS CONEXIONES ENTRE ENLACES

Cualquier manipulador puede ser descrito cinemáticamente dando los valores de cuatro cantidades para cada enlace que conforma su cadena, dos de estos valores describen el enlace en sí mismo y los otros dos describen la conexión del enlace con su homologo consecutivo.

La definición de mecanismos por medio de estas cantidades es una convención usualmente llamada la **notación de Denavit-Hartenberg** explicada en forma detallada en [1]⁴.

⁴ CRAIG, Jhon J. Introduction to robotics: Mechanics and control. Addison-Wesley. 1989

Un resumen del procedimiento para ubicar los sistemas de coordenadas necesarios para la descripción del manipulador, así como de los cuatro parámetros de enlace, es presentado a continuación:

1. Identificar los ejes de cada junta (eje de rotación o de deslizamiento según el caso) e imaginar o dibujar líneas infinitas a lo largo de estos. Para los pasos de 2 a 5 siguientes, considere dos de estas líneas consecutivas (ejes i e $i+1$).
2. Identificar la perpendicular común entre estos, o el punto de intersección. En el punto de intersección o el punto donde la perpendicular común se encuentra con el eje i , asignar el origen del marco que va a estar solidario al enlace i .
3. Asignar el eje \hat{Z}_i apuntando a lo largo del eje de la junta i .
4. Asignar el eje \hat{X}_i apuntando a lo largo de la perpendicular común, o si los ejes se intersecan, asignar \hat{X}_i de tal forma que sea normal al plano que contiene los dos ejes.
5. Asignar el eje \hat{Y}_i de tal forma que se cumpla la regla de la mano derecha en el sistema de coordenadas.
6. Hacer coincidir el sistema de coordenadas $\{0\}$ con el $\{1\}$ cuando la primer variable articular tiene un valor de cero. Para el sistema $\{N\}$ seleccionar la ubicación del origen y la dirección de \hat{X}_n libremente, pero generalmente de tal forma que ocasione que la mayoría de parámetros como sea posible tengan un valor de cero.

Sí los sistemas de coordenadas han sido atados a los enlaces de acuerdo a la convención anterior, las siguientes definiciones de los parámetros de enlace son validas:

- ✓ a_i = la distancia desde \hat{Z}_i a \hat{Z}_{i+1} medida a lo largo de \hat{X}_i ;
- ✓ α_i = el ángulo entre \hat{Z}_i y \hat{Z}_{i+1} medido alrededor de \hat{X}_i ;
- ✓ d_i = la distancia desde \hat{X}_{i-1} a \hat{X}_i medida a lo largo de \hat{Z}_i ; y
- ✓ θ_i = el ángulo entre \hat{X}_{i-1} y \hat{X}_i medido alrededor de \hat{Z}_i .

Usualmente se selecciona $a_i > 0$ debido a que corresponde a una distancia, sin embargo, $\alpha_i, d_i, y \theta_i$ son cantidades con signo.

Aplicando el procedimiento anterior y siguiendo las convenciones mencionadas al manipulador Robot Arm Mr 999 se obtiene los parámetros mostrados en la tabla 2 de acuerdo a la asignación de los sistemas de coordenadas mostrados en la figura 2.1 donde cabe anotar que para la posición mostrada del manipulador todas sus variables articulares tienen un valor de cero, y donde además A1, A2 y A3 son las longitudes de los enlaces 1, 2, y 3 con valores de 150, 118 y 230 mm respectivamente.

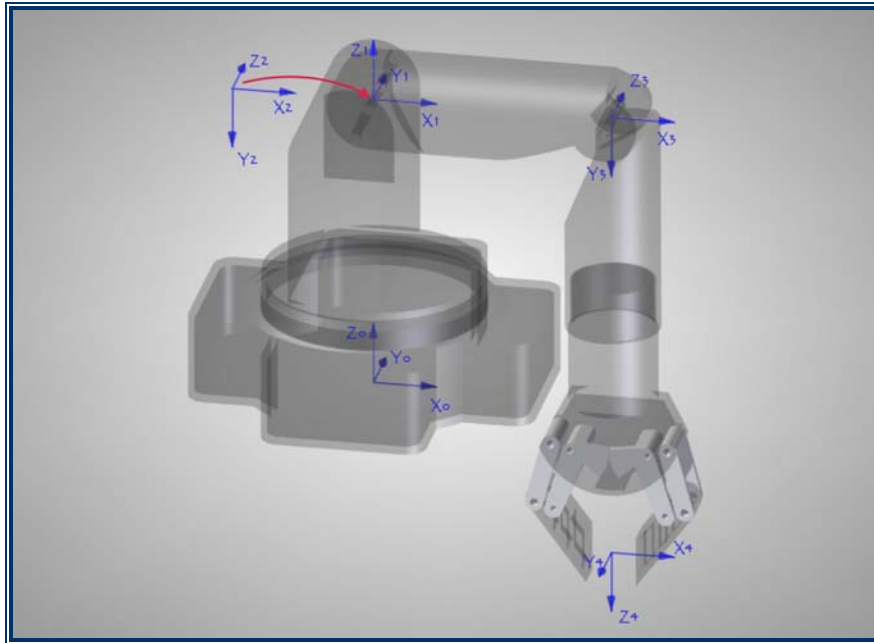


Figura 2. 1. Representación de los sistemas de coordenadas atados al manipulador ROBOT ARM MR 999

	α_{i-1}	a_{i-1}	θ_i	d_i
1	0	0	θ_1	A1
2	$-\pi/2$	0	θ_2	0
3	0	A2	θ_3	0
4	$-\pi/2$	0	θ_4	A3

Tabla 2 Parámetros de Denavit-Hartenberg del manipulador ROBOT ARM

2.1.2 DERIVACIÓN DE LAS TRANSFORMACIONES ENTRE ENLACES

Lo que deseamos en este caso es determinar la transformación que describa el sistema de coordenadas $\{i\}$ relativo al sistema de coordenadas $\{i-1\}$. En

general esta transformación es una función de los cuatro parámetros de enlace. Para cualquier robot dado esta transformación será función de solo una variable, ya que los otros tres parámetros son fijos debido al diseño mecánico. Como hemos definido un sistema de coordenadas para cada uno de los cuatro enlaces, hemos dividido el problema de la cinemática directa en cuatro subproblemas, a saber, cada subproblema es la transformación de un enlace respecto a su precedente, denominada ${}^{i-1}T$.

Para resolver estos subproblemas se empieza por definir tres sistemas de coordenadas intermedios para cada enlace, los cuales se denominarán: {P}, {Q}, y {R}.

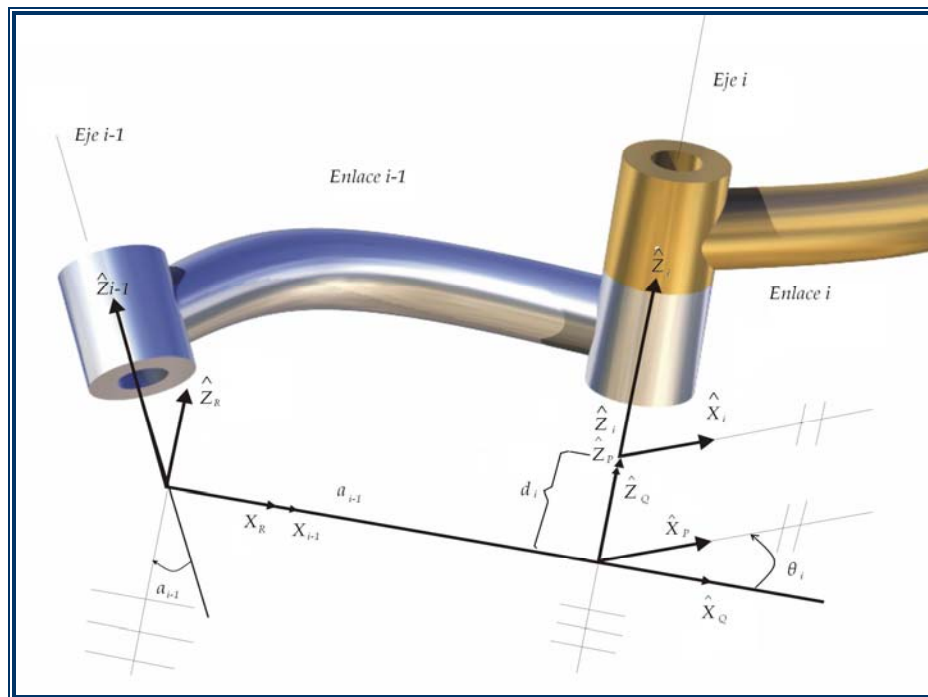


Figura 2. 2.Ubicación de los marcos intermedios {P}, {Q}, y {R}.

La figura 2.2 muestra un par de juntas con los sistemas {P}, {Q}, y {R} definidos así: el marco {R} difiere del marco {i-1} solo por una rotación de α_{i-1} . El marco {Q} difiere del {R} por una traslación de a_{i-1} . El marco {P}

difiere del {Q} por una rotación de θ_i , y el marco {i} difiere del {P} por una traslación d_i . Si queremos obtener la transformación la cual convierte vectores definidos en {i} a su descripción en {i-1}, podemos escribir:

$${}^{i-1}P = {}^{i-1}T_R {}^R T_Q {}^Q T_P {}^P T_i {}^i T^* P \quad (2.1)$$

o

$${}^{i-1}P = {}^{i-1}T_i {}^i T^* P \quad (2.2)$$

donde

$${}^{i-1}T = {}^{i-1}T_R {}^R T_Q {}^Q T_P {}^P T_i \quad (2.3)$$

Considerando cada una de estas transformaciones, se puede ver que (3) puede ser escrita así:

$${}^{i-1}T = Rx(\alpha_{i-1})Dx(a_{i-1})R_Z(\theta_i)D_Z(d_i) \quad (2.4)$$

o

$${}^{i-1}T = Screw_x(a_{i-1}, \alpha_{i-1})Screw_Z(d_i, \theta_i) \quad (2.5)$$

donde la notación $Screw_Q(r, \Phi)$ se utiliza para una traslación a lo largo del eje Q por una distancia r, y una rotación alrededor del mismo eje por un ángulo Φ .

Realizando las multiplicaciones descritas en (2.3) se obtiene la forma general de ${}^{i-1}T$, así:

$${}^{i-1}T = \begin{bmatrix} c\theta_i & -s\theta_i & 0 & a_{i-1} \\ s\theta_i \cdot c\alpha_{i-1} & c\theta_i \cdot c\alpha_{i-1} & -s\alpha_{i-1} & -s\alpha_{i-1} \cdot d_i \\ s\theta_i \cdot s\alpha_{i-1} & c\theta_i \cdot s\alpha_{i-1} & c\alpha_{i-1} & c\alpha_{i-1} \cdot d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.6)$$

donde las letras *s* y *c* representan las funciones seno y coseno respectivamente.

2.1.3 APLICACIÓN AL ROBOT ARM MR 999 E

Ahora, para el manipulador Robot Arm Mr 999, usando la expresión (2.6) podemos calcular la matriz de transformación para cada enlace, obteniendo:

$$\begin{aligned}
 {}^0T_1 &= \begin{bmatrix} \cos(\theta_1) & -\text{sen}(\theta_1) & 0 & 0 \\ \text{sen}(\theta_1) & \cos(\theta_1) & 0 & 0 \\ 0 & 0 & 1 & A1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\
 {}^1T_2 &= \begin{bmatrix} \cos(\theta_2) & -\text{sen}(\theta_2) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -\text{sen}(\theta_2) & -\cos(\theta_2) & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\
 {}^2T_3 &= \begin{bmatrix} \cos(\theta_3) & -\text{sen}(\theta_3) & 0 & A2 \\ -\text{sen}(\theta_3) & \cos(\theta_3) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\
 {}^3T_4 &= \begin{bmatrix} \cos(\theta_4) & -\text{sen}(\theta_4) & 0 & 0 \\ 0 & 0 & 1 & A3 \\ -\text{sen}(\theta_4) & -\cos(\theta_4) & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}
 \end{aligned} \tag{2.7}$$

Ahora podemos obtener la matriz de transformación total, es decir del efector final a la base del manipulador, si multiplicamos todas las matrices de transformación individuales, de la siguiente forma:

$${}^0T_4 = {}^0T_1 {}^1T_2 {}^2T_3 {}^3T_4;$$

al realizar las operaciones matemáticas correspondientes en la expresión anterior, se obtiene:

$${}^0_4T = \begin{bmatrix} r_{11} & r_{12} & r_{13} & P_x \\ r_{21} & r_{22} & r_{23} & P_y \\ r_{31} & r_{32} & r_{33} & P_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.8)$$

donde:

$$\begin{aligned} r_{11} &= \cos(\theta_4)\cos(\theta_1)\cos(\theta_2)\cos(\theta_3) - \cos(\theta_4)\cos(\theta_1)\sin(\theta_2)\sin(\theta_3) + \sin(\theta_1)\sin(\theta_4), \\ r_{21} &= \cos(\theta_4)\sin(\theta_1)\cos(\theta_2)\cos(\theta_3) - \cos(\theta_4)\sin(\theta_1)\sin(\theta_2)\sin(\theta_3) - \cos(\theta_1)\sin(\theta_4), \\ r_{31} &= -(\sin(\theta_2)\cos(\theta_3) + \cos(\theta_2)\sin(\theta_3))\cos(\theta_4), \\ r_{12} &= -\sin(\theta_4)\cos(\theta_1)\cos(\theta_2)\cos(\theta_3) + \sin(\theta_4)\cos(\theta_1)\sin(\theta_2)\sin(\theta_3) + \sin(\theta_1)\cos(\theta_4), \\ r_{22} &= -\sin(\theta_4)\sin(\theta_1)\cos(\theta_2)\cos(\theta_3) + \sin(\theta_4)\sin(\theta_1)\sin(\theta_2)\sin(\theta_3) - \cos(\theta_1)\cos(\theta_4), \\ r_{32} &= (\sin(\theta_2)\cos(\theta_3) + \cos(\theta_2)\sin(\theta_3))\sin(\theta_4), \\ r_{13} &= -\cos(\theta_1)(\sin(\theta_2)\cos(\theta_3) + \cos(\theta_2)\sin(\theta_3)), & ; & (2.9) \\ r_{23} &= -\sin(\theta_1)(\sin(\theta_2)\cos(\theta_3) + \cos(\theta_2)\sin(\theta_3)), \\ r_{33} &= \sin(\theta_2)\sin(\theta_3) - \cos(\theta_2)\cos(\theta_3), \\ p_x &= \cos(\theta_1)(-A_3\cos(\theta_2)\sin(\theta_3) - A_3\sin(\theta_2)\cos(\theta_3) + \cos(\theta_2)A_2), \\ p_y &= \sin(\theta_1)(-A_3\cos(\theta_2)\sin(\theta_3) - A_3\sin(\theta_2)\cos(\theta_3) + \cos(\theta_2)A_2), \\ p_z &= A_3\sin(\theta_2)\sin(\theta_3) - A_3\cos(\theta_2)\cos(\theta_3) - \sin(\theta_2)A_2 + A_1. \end{aligned}$$

Las ecuaciones (2.9) constituyen la cinemática directa del manipulador ROBOT ARM MR 999. Ellas especifican como calcular la posición y orientación del marco {4} atado al efector final del manipulador, relativo al marco {0} atado a la base del manipulador.

Para efectos de agilizar los cálculos, al realizar la simulación de la cinemática directa del manipulador en la interfase elaborada para tal fin, y para cualquier conjunto de posiciones articulares, se utiliza la función **fkine**

encontrada en la herramienta **HEMERO**⁵ que se describe detalladamente en [7]. La función **fkine** se limita a resolver el conjunto de ecuaciones (2.9), es decir, hallar el valor numérico de la matriz de transformación 0_4T , de manera rápida y eficaz, como es necesario en una simulación.

Para el uso de esta función, luego de haber indexado la herramienta en el path de Matlab basta con escribir la siguiente sentencia en la ventana de comandos del programa:

T=fkine (dhARM,q);

donde $T={}_4^0T$ (descrita anteriormente) , **dhARM** es la matriz de parámetros de Denavit-Hartenberg del manipulador mostrada en la tabla dos, adicionándole una quinta columna con cuatro ceros que indican cada uno el tipo de articulación a que se refiere, en este caso todas de rotación (se debería escribir unos si fueran de deslizamiento) y **q** es un vector de cuatro columnas donde cada una indica el valor de la respectiva variable articular.

2.2. CINEMÁTICA INVERSA

En esta sección se estudiará un problema con un grado de dificultad mayor al estudiado en la sección anterior, a saber, dada la posición y orientación del marco atado al efector final del manipulador, calcular el conjunto de ángulos (posiciones articulares) que logran ubicar el efector final en la posición deseada.

⁵ Herramienta Matlab-simulink para la Enseñanza de la RObótica

2.2.1 SOLUCIONABILIDAD

El problema de resolver las ecuaciones cinemáticas del manipulador es un problema no lineal. Este se trata de que, dado el valor numérico de 0_4T nosotros queremos encontrar los valores de θ_1 , θ_2 , θ_3 , y θ_4 .

Considerando las ecuaciones (2.9), el estado real de nuestro problema es: Dada 0_4T como un conjunto de dieciséis valores numéricos (cuatro de los cuales son triviales), debemos resolver las ecuaciones (2.9) para los cuatro ángulos de junta θ_1 , θ_2 , θ_3 , y θ_4 . Las ecuaciones resultantes al intentar resolver el problema mencionado son no lineales y difíciles de resolver.

En este punto es bueno mencionar un ítem importante, el espacio de trabajo del manipulador. Toscamente hablando, el espacio de trabajo del manipulador, es el volumen de espacio el cual el efector final del manipulador puede alcanzar. Para que una solución al problema planteado exista, el punto objetivo debe estar dentro del espacio de trabajo del manipulador. Algunas veces es útil considerar dos definiciones de *espacio de trabajo*: Espacio de trabajo diestro, que es el volumen de espacio el cual el manipulador puede alcanzar con cualquier orientación, esto quiere decir que cada punto en el espacio de trabajo diestro, el efector final puede ser orientado arbitrariamente. El espacio de trabajo alcanzable, que es el volumen de espacio el cual el manipulador puede alcanzar en por lo menos una orientación. Claramente, el espacio de trabajo diestro es un subconjunto del espacio de trabajo alcanzable.

En resumen, si la posición y orientación deseada del marco atado al efector final está en el espacio de trabajo, entonces por lo menos una solución existe.

2.2.2 MÉTODO DE SOLUCIÓN

Diferente al caso de las ecuaciones lineales, no hay algoritmos generales los cuales puedan ser empleados para resolver un conjunto de ecuaciones no lineales. Hay dos métodos generales para resolver la cinemática inversa de un manipulador: las soluciones en forma cerrada y las soluciones numéricas. Debido a su naturaleza iterativa, las soluciones numéricas generalmente son mucho más lentas que la correspondiente solución en forma cerrada, de hecho, las soluciones numéricas son de poco interés a la hora de resolver la cinemática inversa de un manipulador.

Dentro de las soluciones en forma cerrada están las soluciones geométricas y las soluciones algebraicas. Debido a que cualquier relación geométrica debe ser resuelta por métodos algebraicos, ambos métodos son similares. Estos métodos difieren quizás, únicamente en su enfoque.

En MATLAB bastaría con invocar una función (**ikine**), para obtener una solución a la cinemática inversa del manipulador, esta solución sería obtenida por métodos numéricos y ya se mencionó los inconvenientes de este tipo de solución, además de que el autor de la herramienta nos advierte en [7] de los problemas de convergencia al utilizar dicha función.

Por lo anterior, la solución a la cinemática inversa del manipulador ROBOT ARM MR 999 E se hizo en forma algebraica de la manera descrita a continuación.

2.2.3 SOLUCIÓN ALGEBRAICA DEL MANIPULADOR ROBOT ARM

Queremos resolver:

$${}^0_4T = \begin{bmatrix} r_{11} & r_{12} & r_{13} & P_x \\ r_{21} & r_{22} & r_{23} & P_y \\ r_{31} & r_{32} & r_{33} & P_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.10)$$

$$= {}^0_1T(\theta_1) * {}^1_2T(\theta_2) * {}^2_3T(\theta_3) * {}^3_4T(\theta_4)$$

para θ_i cuando 0_4T es dada como valores numéricos.

Un rearrreglo de (2.10) el cual pone la dependencia de θ_1 del lado izquierdo de la ecuación es:

$$\left[{}^0_1T(\theta_1) \right]^{-1} * {}^0_4T = {}^1_2T(\theta_2) * {}^2_3T(\theta_3) * {}^3_4T(\theta_4) \quad (2.11)$$

Invirtiendo 0_1T podemos escribir la ecuación anterior como:

$$\begin{bmatrix} \cos(\theta_1) & \text{sen}(\theta_1) & 0 & 0 \\ -\text{sen}(\theta_1) & \cos(\theta_1) & 0 & 0 \\ 0 & 0 & 1 & -A1 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} r_{11} & r_{12} & r_{13} & P_x \\ r_{21} & r_{22} & r_{23} & P_y \\ r_{31} & r_{32} & r_{33} & P_z \\ 0 & 0 & 0 & 1 \end{bmatrix} = {}^1_4T \quad (2.12)$$

de donde conocemos 1_4T , que es:

$${}^1_4T = \begin{bmatrix} \frac{1}{2} \cdot c(\lambda) + \frac{1}{2} \cdot c(\eta) & -\frac{1}{2} \cdot s(\eta) + \frac{1}{2} \cdot s(\lambda) & -s(\mu) & -s(\mu) \cdot A3 + c(\theta_2) \cdot A2 \\ -s(\theta_4) & -c(\theta_4) & 0 & 0 \\ -\frac{1}{2} \cdot s(\eta) - \frac{1}{2} \cdot s(\lambda) & \frac{1}{2} \cdot c(\lambda) - \frac{1}{2} \cdot c(\eta) & -c(\mu) & -c(\mu) \cdot A3 - s(\theta_2) \cdot A2 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

con:

$$\lambda = \theta_2 + \theta_3 - \theta_4 ;$$

$$\eta = \theta_2 + \theta_3 + \theta_4 ;$$

$$\mu = \theta_2 + \theta_3 ;$$

Esta técnica simple de multiplicar cada lado de la ecuación por una inversa es frecuentemente usada para la separación de variables en busca de conseguir una ecuación soluble.

Igualando los elementos (2,4) a ambos lados de la ecuación (2.12) tenemos:

$$-\text{sen}(\theta_1) \cdot p_x + \text{cos}(\theta_1) \cdot p_y = 0 \quad (2.13)$$

Para resolver la ecuación de esta forma, se debe hacer la siguiente sustitución trigonométrica:

$$\begin{aligned} P_x &= \rho \cdot \text{cos}(\Phi), \\ P_y &= \rho \cdot \text{sen}(\Phi), \end{aligned} \quad (2.14)$$

donde

$$\begin{aligned} \rho &= \sqrt{P_x^2 + P_y^2}, \\ \Phi &= \text{Atan2}(P_y, P_x). \end{aligned} \quad (2.15)$$

sustituyendo (2.14) en (2.13), obtenemos:

$$\text{cos}(\theta_1) \cdot \text{sen}(\Phi) - \text{sen}(\theta_1) \cdot \text{cos}(\Phi) = 0 \quad (2.16)$$

usando la formula de la diferencia de ángulos:

$$\text{sen} (\Phi - \theta_1) = 0 \quad (2.17)$$

de donde se observa que $(\Phi - \theta_1) = n*\pi$ con $n = 0, 1, 2, 3, \dots$

por lo cual obtenemos $\theta_1 = \Phi - n*\pi$;

entonces

$$\theta_1 = \text{Atan2} (P_y, P_x) - n*\pi; \quad (2.18)$$

Nótese que se han encontrado dos posibles soluciones para θ_1 , que corresponden a ángulos girados 180° uno de otro.

Ahora que θ_1 es conocido, el lado izquierdo de la ecuación (2.12) es conocido.

Si igualamos ahora los elementos (1,4) y (3,4) a ambos lados de la ecuación (2.12) se obtiene:

$$\begin{aligned} \cos (\theta_1)* p_x + \text{sen} (\theta_1)*p_y &= - \text{sen} (\theta_2+\theta_3) *A_3 + \cos (\theta_2)*A_2 \quad , \\ p_z - A_1 &= - \cos (\theta_2+\theta_3)*A_3 - \text{sen} (\theta_2)*A_2 \quad , \end{aligned} \quad (2.19)$$

si ahora elevamos al cuadrado las ecuaciones (2.19) y la (2.13) y se suman las ecuaciones resultantes se obtiene:

$$\text{sen} (\theta_3) = k \quad , \quad (2.20)$$

donde

$$k = \frac{(a_3^2 + a_2^2 - P_x^2 - P_y^2 - P_z^2 - a_1^2 + 2*a_1*P_z)}{2*a_3*a_2} \quad (2.21)$$

por lo cual se obtiene:

$$\theta_3 = \text{Asen} (k) \quad (2.22)$$

Si consideramos nuevamente la ecuación (2.11) podemos reescribirla de tal manera que todo el lado izquierdo es una función únicamente de variables conocidas y de θ_2 :

$$\left[{}^0_3T(\theta_2) \right]^{-1} * {}^0_4T = {}^3_4T \quad (2.23)$$

de donde sabemos como obtener el lado izquierdo de la ecuación (2.23) simbólicamente (ver las expresiones (2.7)) y conocemos el lado derecho de la ecuación, por lo cual la anterior expresión se convierte en:

$$\begin{bmatrix} \cos(\theta_1)\cos(\mu) & \text{sen}(\theta_1)\cos(\mu) & -\text{sen}(\mu) & A1 \cdot \text{sen}(\mu) - A2 \cos(\theta_3) \\ -\cos(\theta_1)\text{sen}(\mu) & -\text{sen}(\theta_1)\text{sen}(\mu) & -\cos(\mu) & A2 \cdot \text{sen}(\theta_3) + A1 \cos(\mu) \\ -\text{sen}(\theta_1) & \cos(\theta_1) & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

*

$$\begin{bmatrix} r_{11} & r_{12} & r_{13} & P_x \\ r_{21} & r_{22} & r_{23} & P_y \\ r_{31} & r_{32} & r_{33} & P_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.24)$$

=

$$\begin{bmatrix} \cos(\theta_4) & -\text{sen}(\theta_4) & 0 & 0 \\ 0 & 0 & 1 & A3 \\ -\text{sen}(\theta_4) & -\cos(\theta_4) & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Si igualamos los elementos (1,4) y (2,4) a ambos lados de la ecuación anterior obtenemos:

$$\begin{aligned} \cos(\theta_1) \cdot \cos(\theta_2 + \theta_3) \cdot p_x + \sin(\theta_1) \cdot \cos(\theta_2 + \theta_3) \cdot p_y - \sin(\theta_2 + \theta_3) \cdot p_z - A_2 \cdot \cos(\theta_3) + A_1 \cdot \sin(\theta_2 + \theta_3) &= 0, \\ -\cos(\theta_1) \cdot \sin(\theta_2 + \theta_3) \cdot p_x - \sin(\theta_1) \cdot \sin(\theta_2 + \theta_3) \cdot p_y - \cos(\theta_2 + \theta_3) \cdot p_z + A_2 \cdot \sin(\theta_3) + A_1 \cdot \cos(\theta_2 + \theta_3) &= A_3, \end{aligned}$$

$$(2.25)$$

resolviendo simultáneamente para $\sin(\theta_2 + \theta_3)$ y $\cos(\theta_2 + \theta_3)$ se obtiene:

$$\sin(\theta_2 + \theta_3) = \frac{(a_1 - P_z) \cdot a_2 \cdot \cos(\theta_3) - (a_3 - a_2 \cdot \sin(\theta_3)) \cdot (\cos(\theta_1) \cdot P_x + \sin(\theta_1) \cdot P_y)}{(a_1 - P_z)^2 + (\cos(\theta_1) \cdot P_x + \sin(\theta_1) \cdot P_y)^2},$$

y

$$\cos(\theta_2 + \theta_3) = \frac{(a_1 - P_z) \cdot (a_3 - a_2 \cdot \sin(\theta_3)) + a_2 \cdot \cos(\theta_3) \cdot (\cos(\theta_1) \cdot P_x + \sin(\theta_1) \cdot P_y)}{(a_1 - P_z)^2 + (\cos(\theta_1) \cdot P_x + \sin(\theta_1) \cdot P_y)^2},$$

$$(2.26)$$

como los denominadores son iguales y positivos, podemos resolver para $(\theta_2 + \theta_3)$ de la siguiente manera:

$$\begin{aligned} (\theta_2 + \theta_3) &= \\ \text{Atan2} &\left[\frac{(a_1 - P_z) \cdot a_2 \cdot \cos(\theta_3) - (a_3 - a_2 \cdot \sin(\theta_3)) \cdot (\cos(\theta_1) \cdot P_x + \sin(\theta_1) \cdot P_y)}{(a_1 - P_z) \cdot (a_3 - a_2 \cdot \sin(\theta_3)) + a_2 \cdot \cos(\theta_3) \cdot (\cos(\theta_1) \cdot P_x + \sin(\theta_1) \cdot P_y)}, \right] \end{aligned}$$

$$(2.27)$$

de cuya solución se puede extraer θ_2 así:

$$\theta_2 = (\theta_2 + \theta_3) - \theta_3 \quad (2.28)$$

En este momento el lado izquierdo completo de la ecuación (2.23) es conocido y podemos igualar los elementos (1,2) y (3,2) a ambos lados de la ecuación para obtener:

$$\begin{aligned} r_{12} \cos(\theta_1) \cos(\theta_2 + \theta_3) + r_{22} \sin(\theta_1) \cos(\theta_2 + \theta_3) - r_{32} \sin(\theta_2 + \theta_3) &= -\sin(\theta_4), \\ -r_{12} \sin(\theta_1) + r_{22} \cos(\theta_1) &= -\cos(\theta_4), \end{aligned} \quad (2.29)$$

luego:

$$\theta_4 = \text{Atan2} [r_{32} \sin(\theta_2 + \theta_3) - r_{12} \cos(\theta_1) \cos(\theta_2 + \theta_3) - r_{22} \sin(\theta_1) \cos(\theta_2 + \theta_3), r_{12} \sin(\theta_1) - r_{22} \cos(\theta_1)] \quad (2.30)$$

donde los valores de r_{12} , r_{22} , y r_{32} son conocidos, ya que se supuso el previo conocimiento de la matriz de transformación 0_4T .

Para efectos de simular la cinemática inversa del manipulador Robot Arm Mr 999, las expresiones (2.18), (2.22), (2.27), (2.28) y (2.30) fueron introducidas en una función de MATLAB llamada **InversaRobotArm.m** que se ha anexado. Para su instalación y correcto funcionamiento en la interfase de aplicación desarrollada para tal fin, cabe advertir que la citada función no invoca a ninguno de los toolbox empleados.

La función **InversaRobotArm.m** requiere como parámetros de entrada la posición de un punto objetivo en coordenadas cartesianas que viene a ser la ubicación del sistema de coordenadas {4}, así como la orientación de dicho marco. Aquí hay que aclarar que la orientación del marco que debe pasársele a la función está en concordancia con la convención para la representación de la orientación de sistemas de coordenadas llamado de **ángulos fijos X-Y-Z** descrito de forma detallada en [1], es decir, se empieza con el marco {4} coincidente con el marco {0}, y, primero se rota el marco {4} alrededor del eje

\hat{X}_0 un ángulo φ , luego se rota alrededor del eje \hat{Y}_0 un ángulo β , y por ultimo se rota alrededor del eje \hat{Z}_0 un ángulo α . Cada una de las tres rotaciones toma lugar alrededor de un eje del marco de referencia fijo $\{0\}$ atado a la base del manipulador. Un esquema de las rotaciones causadas por esta convención es mostrada en la figura 2.3.

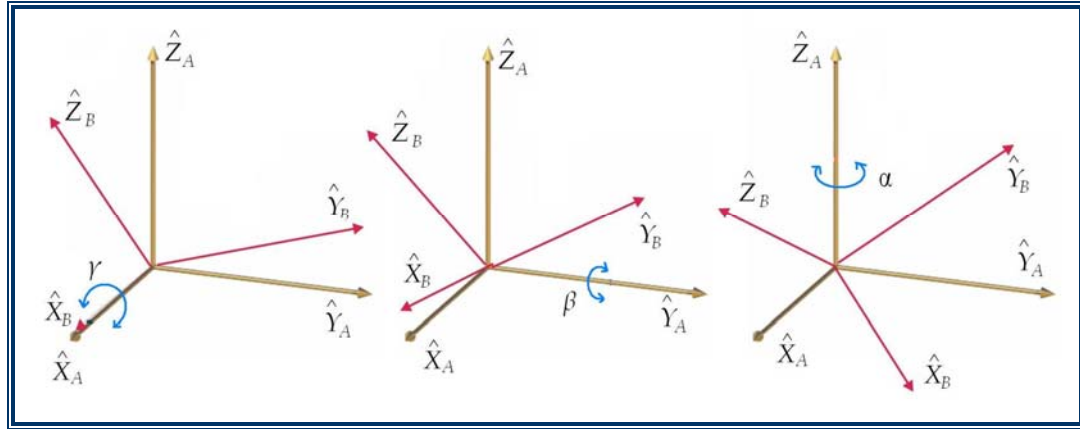


Figura 2. 3. Esquema de la representación de una orientación por medio de la notación de ángulos fijos X - Y - Z.

Es extremadamente importante comprender el orden de las rotaciones mostradas en la figura 2.3. Pensando en términos de rotaciones como operadores, se puede aplicar las rotaciones para obtener la matriz de rotación:

$${}^0R_{XYZ}(\gamma, \beta, \alpha) = \begin{bmatrix} c(\alpha)c(\beta), c(\alpha)s(\beta)s[\gamma] - s[\alpha]c(\gamma), c(\alpha)s[\beta]c(\gamma) + s(\alpha)s[\gamma] \\ s(\alpha)c(\beta), s(\alpha)s(\beta)s[\gamma] + c[\alpha]c(\gamma), s(\alpha)s[\beta]c(\gamma) - c(\alpha)s[\gamma] \\ -s(\beta), c(\beta)s(\gamma), c(\beta)c(\gamma) \end{bmatrix} \quad (2.31)$$

Debemos mantener en mente la definición dada que especifica el orden de las rotaciones. La ecuación (2.31) es correcta solo para rotaciones realizadas en el siguiente orden: alrededor del eje \hat{X}_0 un ángulo φ , luego alrededor del eje \hat{Y}_0 un ángulo β , y por ultimo alrededor del eje \hat{Z}_0 un ángulo α .

De acuerdo a la ecuación (2.31), si ingresamos los valores de φ , β , y α en la función *InversaRobotArm.m* podemos calcular los valores r_{12} , r_{22} , y r_{32} requeridos para el calculo de θ_4 en la ecuación (2.30) y de esta manera podemos simular en forma completa la cinemática inversa del manipulador Robot Arm Mr 999.

2.3. JACOBIANO DEL MANIPULADOR

El jacobiano de un manipulador es una matriz que relaciona las velocidades articulares con las velocidades cartesianas del extremo en la siguiente forma:

$${}^0v = {}^0J(q) * q' \quad (2.32)$$

expresión en la cual se ha tomado como referencia el sistema {0} atado a la base del manipulador. Debe observarse como el jacobiano se expresa respecto a un determinado sistema de referencia, igualmente, debe notarse como, para unos determinados valores de las variables articulares, las relaciones entre las velocidades articulares y las velocidades en el espacio cartesiano son lineales. El vector de velocidades en el espacio cartesiano puede escribirse como:

$${}^0v = \begin{bmatrix} {}^0v \\ {}^0\omega \end{bmatrix} \quad (2.33)$$

siendo 0v un vector 3 x 1 de velocidades lineales, y ${}^0\omega$ un vector 3 x 1 de velocidades rotacionales.

En general, el número de columnas de J es el número de articulaciones del manipulador y el número de filas el de grados de libertad.

Existen diferentes métodos para obtener el jacobiano de un determinado manipulador. El mas evidente es el de la obtención analítica de expresiones de las velocidades cartesianas en función de las velocidades articulares. Para ello, puede aplicarse la técnica de propagación de las velocidades expuesta con gran detalle en [1].

La obtención de las velocidades articulares en función de las velocidades cartesianas en el extremo involucra la consideración de la inversa del jacobiano. Si la matriz es no singular puede escribirse:

$$q' = J^{-1}(q) * v \quad (2.34)$$

Sin embargo, es frecuente que existan valores de q para los que el jacobiano es singular. Estas singularidades se presentan en los límites del espacio de trabajo, o en su interior cuando dos o más ejes de articulaciones están alineados.

La ya mencionada herramienta HEMERO está dotada de dos funciones (**jacobn** y **velprop**) que calculan en forma simbólica el jacobiano de un manipulador al dársele como parámetros de entrada un vector cuyos elementos corresponden uno a uno con las variables articulares.

En nuestro caso particular se hizo uso de la función **jacob0 (dhARM, q)** que corresponde a la aplicación de un método alternativo⁶ para la obtención del jacobiano del manipulador Robot Arm Mr 999 respecto al marco {0}.

Para aplicar esta función en Matlab se emplean las instrucciones:

⁶ **R. P. Paul**, *Robot Manipulators: Mathematics, Programming, and Control*. Cambridge, Massachusetts: MIT Press, 1981.

```

syms theta1 theta2 theta3 theta4 real;           %variables articulares theta1, theta2, theta3 y theta4
syms A1 A2 A3 real;                             %parámetros A1, A2 y A3 del manipulador
dhARM=[0      0  theta1  A1  0;
      -pi/2   0  theta2  0  0;
      0      A2  theta3  0  0;
      -pi/2   0  theta4  A3  0];
q = [theta1 theta2 theta3 theta4];
J0 = simple (jacob0(dhARM,q))

```

cuya ejecución proporciona como resultado la siguiente matriz:

$${}^3_4T = \begin{bmatrix} 1.1 & 1.2 & 1.3 & 1.4 \\ 2.1 & 2.2 & 2.3 & 2.4 \\ 3.1 & 3.2 & 3.3 & 3.4 \\ 4.1 & 4.2 & 4.3 & 4.4 \\ 5.1 & 5.2 & 5.3 & 5.4 \\ 6.1 & 6.2 & 6.3 & 6.4 \end{bmatrix} \quad (2.35)$$

donde:

```

1.1 = sin(theta1)*(cos(theta2)*sin(theta3)*A3-cos(theta2)*A2+sin(theta2)*cos(theta3)*A3);
1.2 = cos(theta1)*(-sin(theta2)*A2-cos(theta2)*cos(theta3)*A3+sin(theta2)*sin(theta3)*A3);
1.3 = A3*cos(theta1)*(sin(theta2)*sin(theta3)-cos(theta2)*cos(theta3));
1.4 = 0;
2.1 = -cos(theta1)*(cos(theta2)*sin(theta3)*A3-cos(theta2)*A2+sin(theta2)*cos(theta3)*A3);
2.2 = sin(theta1)*(-sin(theta2)*A2-cos(theta2)*cos(theta3)*A3+sin(theta2)*sin(theta3)*A3);
2.3 = A3*sin(theta1)*(sin(theta2)*sin(theta3)-cos(theta2)*cos(theta3));
2.4 = 0; 3 = A3*(cos(theta2)*sin(theta3)+sin(theta2)*cos(theta3));
3.1 = 0;
3.2 = cos(theta2)*sin(theta3)*A3-cos(theta2)*A2+sin(theta2)*cos(theta3)*A3;
3.3 = 0;

```

```

3.4 = 0;
4.1 = 0;
4.2 = -sin(theta1);
4.3 = -sin(theta1);
4.4 = -cos(theta1)*(cos(theta2)*sin(theta3)+sin(theta2)*cos(theta3));
5.1 = 0;
5.2 = cos(theta1);
5.3 = cos(theta1);
5.4 = -sin(theta1)*(cos(theta2)*sin(theta3)+sin(theta2)*cos(theta3));
6.1 = 1;
6.2 = 0;
6.3 = 0;
6.4 = sin(theta2)*sin(theta3)-cos(theta2)*cos(theta3);

```

Ahora, para efectos de simulación y teniendo en cuenta la ecuación (2.32), se elaboró un archivo en Matlab llamado **velocidades.m** el cual toma los parámetros de entrada que deben ser las cuatro posiciones articulares y sus correspondientes velocidades rotacionales. Con las posiciones articulares evalúa el jacobiano mostrado en la expresión (2.35) para dicho instante, y con las velocidades rotacionales forma el vector q' para luego realizar las respectivas operaciones matemáticas y entregar como resultado un vector columna llamado v_0 cuyos elementos correspondientes a las tres primeras filas son las componentes v_x , v_y , y v_z respectivamente de la velocidad lineal del efector final relativas al marco $\{0\}$, y los elementos correspondientes a las tres últimas filas son las componentes ω_x , ω_y , y ω_z respectivamente de la velocidad angular del efector final relativas al marco $\{0\}$.

3. CINÉTICA DEL MANIPULADOR

3.1. CONSIDERACIONES GENERALES SOBRE LA CINÉTICA

La cinética del robot trata con la formulación matemática de las ecuaciones de movimiento del brazo. Las ecuaciones de movimiento de un manipulador son un conjunto de ecuaciones matemáticas que describen su conducta dinámica. Tales ecuaciones son útiles para la simulación en computador del movimiento del robot, el diseño de ecuaciones de control apropiadas para el robot y la evaluación del diseño y estructura del brazo.

Nos centramos, en la formulación, características y propiedades de las ecuaciones dinámicas de movimiento que son adecuadas con fines de control. El objetivo del control de un manipulador basado en computador es mantener la respuesta dinámica del mismo de acuerdo con algún rendimiento del sistema preespecificado y objetivos deseados. En general, el rendimiento dinámico de un manipulador depende directamente de la eficacia de los algoritmos de control y de su modelo dinámico.

El problema de control consiste en obtener modelos dinámicos del brazo del robot físico y a continuación especificar leyes o estrategias de control correspondientes para conseguir la respuesta y rendimiento del sistema deseado. Este capítulo trata solamente con la primera parte el problema de control del manipulador; esto es, la modelización y evaluación de las propiedades y conducta dinámica de robot controlados por computador.

3.2. GENERALIDADES DE LOS MÉTODOS CONVENCIONALES⁷

El modelo dinámico de un robot se puede obtener a partir de leyes físicas conocidas tales como newtoniana y lagrangiana. Esto conduce al desarrollo de las ecuaciones de movimiento dinámico para las diversas articulaciones del manipulador en términos de los parámetros geométricos e inerciales de los elementos.

Métodos convencionales como las formulaciones de (Lagrange-Euler L-E) y Newton-Euler (N-E) se pueden aplicar entonces sistemáticamente para desarrollar las ecuaciones de movimiento del robot. De estas dos formulaciones se pueden obtener diferentes formas de describir la dinámica del brazo del robot, tales como las ecuaciones de Lagrange-Euler de Uicker (Uicker [1965], Bejczy [1974]) las ecuaciones recursivas de Lagrange de Hollerbach (Hollerbach[1980]), las ecuaciones de Newton-Euler de Luh (Luh y col [1980]) y las ecuaciones generalizadas d'Alembert y Lee (Lee y col. [1983]).

Estas ecuaciones de movimiento son "equivalentes" unas a otras en el sentido que describen la conducta dinámica del mismo robot físico. Sin embargo sus estructuras pueden diferir porque se obtienen por diversas razones y objetivos. Algunas se obtienen para lograr tiempos de cálculo rápido en la evaluación de los pares de las articulaciones nominales para controlar el manipulador, otras se obtienen para facilitar el análisis y la síntesis de control, y todavía otras se obtienen para mejorar la simulación en un computador del movimiento del robot.

La obtención del modelo dinámico de un manipulador basado en la formulación de L-E es simple y sistemática. Suponiendo el movimiento del

⁷ Para profundizar en diferentes métodos [2], [5].

cuerpo rígido, las ecuaciones de movimiento resultante, excluyendo la dinámica de los dispositivos de control electrónico, huelgo y el rozamiento de los engranajes, son un conjunto de ecuaciones diferenciales no lineales acopladas de segundo orden. Bejczy [1974], utilizando la representación de la matriz de transformación homogénea 4x4 de la cadena cinemática y la formulación lagrangiana, ha demostrado que las ecuaciones de movimiento dinámico para un robot tipo Stanford de seis articulaciones son fuertemente no lineales y constan de carga inercial, fuerzas de reacción de acople entre las articulaciones (Coriolis y centrífuga) y efectos de carga de la gravedad. Más aún, estos pares/fuerzas dependen de los parámetros físicos del manipulador, de la configuración instantánea de las articulaciones, de la velocidad y aceleración de las articulaciones y de la carga que está soportando el robot.

Lagrange-Euler. Las ecuaciones de movimiento L-E proporcionan ecuaciones de estado explícitas para la dinámica del robot y se pueden utilizar para analizar y diseñar estrategias de control avanzadas en el espacio de las variables de articulación. En una menor medida se están utilizando para resolver el problema *dinámico directo*, esto es, dadas las fuerzas/pares deseadas, se utilizan las ecuaciones dinámicas para resolver las aceleraciones de las articulaciones, que se integran a continuación para obtener las coordenadas y velocidades generalizadas; o para el problema *dinámico inverso*, esto es, dadas las coordenadas generalizadas deseadas y sus primeras dos derivadas respecto del tiempo, se calculan las fuerzas/pares generalizados. En ambos casos se puede necesitar los coeficientes dinámicos, desgraciadamente el cálculo de estos coeficientes requiere una relativa cantidad de operaciones aritméticas. Así las ecuaciones de L-E son muy

difíciles de utilizar con fines de control en tiempo real a menos que se simplifiquen⁸.

Newton-Euler. Como una alternativa para derivar ecuaciones de movimiento más eficientes, se dirigió la atención a desarrollar algoritmos para calcular las fuerzas/pares generalizados basados en las ecuaciones de movimiento de N-E (Armstrong [1979], Orin y col. [1979], Luh y col. [1980]). La obtención es simple pero engorrosa, e implica términos de producto vectorial. Las ecuaciones dinámicas resultantes, excluyendo la dinámica del dispositivo de control huelgo y rozamiento de los engranajes, son un conjunto de ecuaciones recursivas hacia delante y hacia atrás. Este conjunto se puede aplicar secuencialmente a los elementos del robot. La recursión hacia delante propaga la información cinemática –tal como velocidades lineales, velocidades angulares, aceleraciones angulares y aceleraciones lineales del centro de masas de cada elemento- desde el sistema de coordenadas inercial hasta el sistema de coordenadas de la mano. La recursión hacia atrás propaga las fuerzas y momentos ejercidos sobre cada elemento desde el efector final del manipulador hasta el sistema de referencia de la base.

Para el manipulador RobotArm MR 999, se aplica en método de Newton-Euler (N-E) en su forma iterativa, por las ventajas antes expuestas en cuanto a la versatilidad y a la economía computacional. En el apartado siguiente y los venideros se estudiarán los parámetros implicados en este método, y como se aplican a nuestro manipulador específico.

3.3. ARTICULACIÓN SIMPLE DE ROTACIÓN

⁸ FU, King Sun. Robótica: Control, detección, Visión e inteligencia. McGraw-Hill. 1988

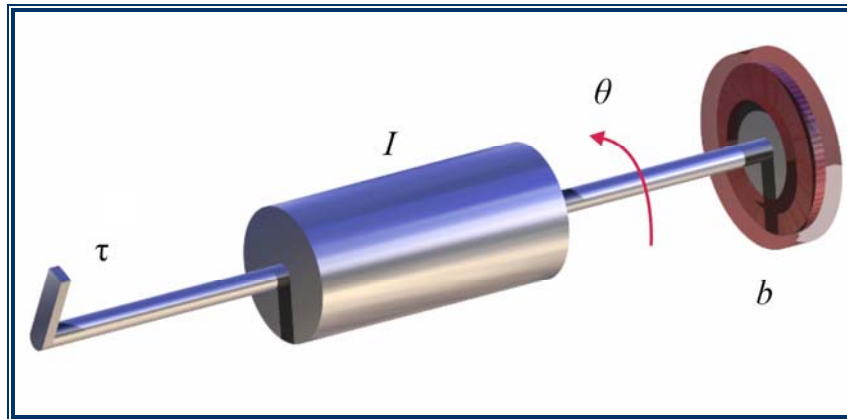


Figura 3.1. Articulación de rotación

Considere una junta de rotación como la de la Figura 3.1, donde se aplica un par τ , a un eje con un momento de inercia I y con un cojinete con coeficiente de fricción viscosa b . La relación entre τ , y el ángulo rotado θ viene dada por:

$$\tau = I\ddot{\theta} + b\dot{\theta} \quad (3.1)$$

Donde el primer término del lado derecho representa el par de inercia, y el segundo el de rozamiento viscoso.

En este caso, el modelo es lineal. Tomando transformadas de Laplace para condiciones iniciales nulas, se tiene que la función de transferencia viene dada por:

$$\frac{\theta(s)}{\tau(s)} = \frac{1}{Is^2 + bs} \quad (3.2)$$

Siendo $\tau(s)$ la transformada de Laplace del par de entrada, y $\theta(s)$ la transformada de Laplace del ángulo rotado.

Este modelo simple puede complicarse si consideramos, por ejemplo, un modelo mas completo de fricciones, incluyendo tanto la fricción viscosa como la de Coulomb. En cuyo caso, se obtendría un modelo no lineal⁹:

$$\tau = I\ddot{\theta} + b\dot{\theta} + c \cdot \text{sgn}(\dot{\theta}) \quad (3.3)$$

Donde $\text{sgn}(\dot{\theta})$ es la función signo definida como:

$$\begin{aligned} \text{sgn}(\dot{\theta}) &= 1 \cdots \text{para } \dot{\theta} > 0 \\ \text{sgn}(\dot{\theta}) &= -1 \cdots \text{para } \dot{\theta} < 0 \end{aligned} \quad (3.4)$$

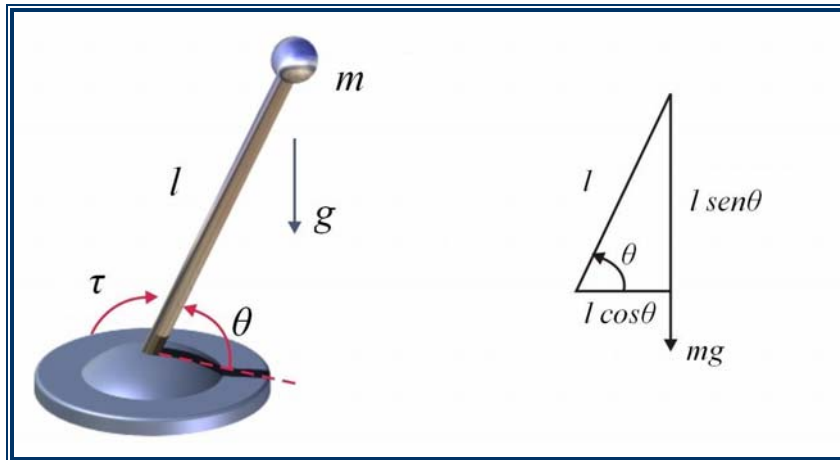


Figura 3.2. Manipulador con una articulación

Consideremos ahora, un péndulo invertido, que representa un manipulador de una articulación de rotación, ilustrado en la Figura 3.2. Asumiendo la masa concentrada en el extremo, se tiene un momento de inercia $I = mL^2$. La gravedad genera un momento sobre el eje de la base $mgl \cos \theta$. Retomemos el

⁹ OLLERO, Aníbal. Robótica, manipuladores y robots móviles. Alfaomega. 2003

modelo de fricciones en el que se tuvieron en cuenta las fricciones viscosas y las de Coulomb (3.3). Así tendremos el siguiente modelo:

$$\tau = ml^2\ddot{\theta} + b\dot{\theta} + mgl \cos \theta + c \cdot \text{sgn}(\dot{\theta}) \quad (3.5)$$

La ecuación anterior puede escribirse como:

$$\tau = I\ddot{\theta} + F(\dot{\theta}) + G(\theta) \quad (3.6)$$

en la cual

$$F(\dot{\theta}) = b\dot{\theta} + c \cdot \text{sgn}(\dot{\theta}) \quad (3.7)$$

son los momentos generados por las fricciones y

$$G(\theta) = mlg \cdot \cos \theta \quad (3.8)$$

representa los efectos gravitatorios.

3.3.1. GENERALIZANDO A UN MANIPULADOR DE n ARTICULACIONES¹⁰

Continuando con la hipótesis del apartado anterior, de las masas concentradas, consideraremos un manipulador con n articulaciones.

En cuyo caso, al analizar una articulación es necesario contabilizar los momentos generados sobre esta por las demás articulaciones.

¹⁰ FU, King Sun. Robótica: Control, detección, Visión e inteligencia. McGraw-Hill. 1988

Entonces tendremos una expresión de (3.6) generalizada para n articulaciones de la siguiente forma:

$$\tau = M(\theta)\ddot{\theta} + F(\dot{\theta}) + V(\theta, \dot{\theta}) + G(\theta) \quad (3.9)$$

donde τ , θ , F , V , G son ahora vectores de n elementos, y $M(\theta)$ es una matriz de $n \times n$ a la que se conoce como matriz de masas del manipulador. Los elementos de M dependen de las distintas variables articulares y tienen como coeficientes expresiones en las que intervienen los valores de las masas concentradas y las longitudes entre ellas.

Los elementos del vector de términos gravitatorios $G(\theta)$ también dependen de las diferentes variables articulares. En lo que concierne al vector de fricciones $F(\dot{\theta})$, sus elementos dependen solamente de la variable articular correspondiente; es decir, pueden desacoplarse. Adviértase, que se ha añadido un vector $V(\theta, \dot{\theta})$ que permite considerar los momentos generados por las fuerzas centrífugas y de Coriolis.

3.4. OBTENCIÓN DEL MODELO DINÁMICO MEDIANTE FORMULACIÓN ITERATIVA DE NEWTON-EULER

3.4.1. ECUACIONES BÁSICAS¹¹

Considérese un cuerpo rígido de masa total m cuyo centro de masas tiene una aceleración \dot{v}_c . De acuerdo con la ecuación de Newton, la fuerza que actúa en el centro de masas viene dado por:

$$F = m\dot{v}_c \quad (3.10)$$

¹¹ En esta sección utilizaremos la notación y desarrollos de **CRAIG, Jhon J.** Introduction to robotics: Mechanics and control. Addison-Wesley. 1989

Por otra parte, si un cuerpo rígido rota con velocidad angular ω y aceleración angular $\dot{\omega}$, el momento angular que actúa en el cuerpo viene dado por la ecuación de Euler:

$$N = {}^c I \dot{\omega} + \omega \times {}^c I \omega \quad (3.11)$$

Siendo ${}^c I$ el tensor de inercias del cuerpo en un sistema {C} cuyo origen esta en el centro de masas. Se trata ahora de aplicar estas ecuaciones para calcular los momentos necesarios en las articulaciones de un robot para que la posición θ , velocidad $\omega = \dot{\theta}$ y aceleración $\dot{\omega} = \ddot{\theta}$ de las articulaciones sean las deseadas. Se supondrá que la distribución de masas es conocida.

El problema puede resolverse en forma iterativa realizando, en primer lugar, iteraciones hacia fuera, desde la base a la última articulación, con objeto de calcular velocidades y aceleraciones y, a continuación, se calculan fuerzas y momentos actuando en cada articulación. En la iteraciones hacia fuera se calculan la velocidad angular y las aceleraciones lineales y angulares del centro de masas, desde la articulación 1 a la n . para ello es necesario propagar las velocidades angulares de la misma forma que se trató en el modelo cinemático. Considérese el enlace, o eslabón de la cadena cinemática, i entre las articulaciones i e $i+1$ tal como se aprecia en la Figura 3.3 la relación entre las velocidades angulares un los cuadros de referencia {i} e {i+1} asociados a las articulaciones viene dada por:

$${}^{i+1}\omega_{i+1} = {}^{i+1}R^i \omega_i + \dot{\theta}_{i+1} {}^{i+1}\hat{Z}_{i+1} \quad (3.12)$$

Siendo ${}^{i+1}\hat{Z}_{i+1}$ el vector unitario según el eje de la articulación {i+1}

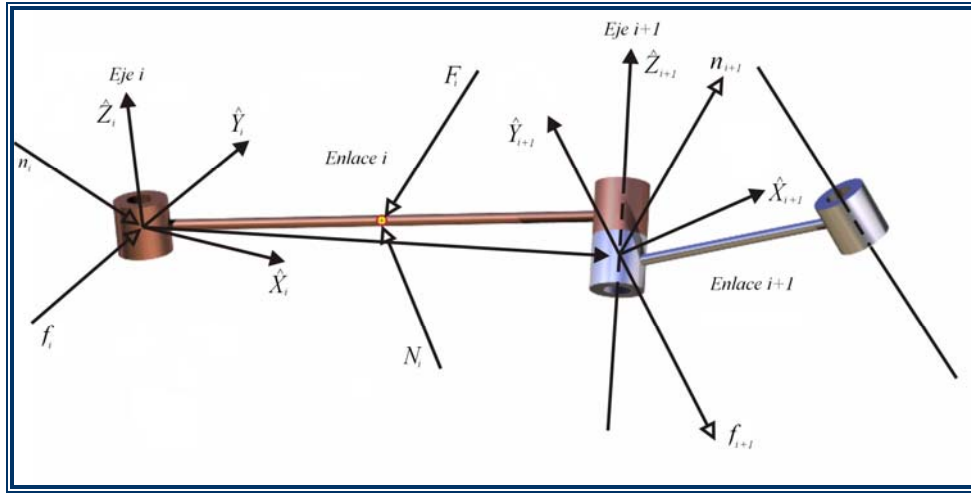


Figura 3.3. Balance de fuerzas en dos articulaciones consecutivas

Igualmente, es necesario propagar las aceleraciones. Teniendo en cuenta que la relación entre las aceleraciones de dos articulaciones consecutivas viene dada por:

$${}^{i+1}\dot{\omega}_{i+1} = {}^{i+1}R^i \dot{\omega}_i + {}^{i+1}R^i \omega_i \times \dot{\theta}_{i+1} {}^{i+1}\hat{Z}_{i+1} + \ddot{\theta}_{i+1} {}^{i+1}\hat{Z}_{i+1} \quad (3.13)$$

Cuando la articulación i+1 es prismática, esto se simplifica a

$${}^{i+1}\dot{\omega}_{i+1} = {}^{i+1}R^i \dot{\omega}_i \quad (3.14)$$

La aceleración lineal del origen de {i+1} esta dada por:

$${}^{i+1}\dot{v}_{i+1} = {}^{i+1}R^i \left[{}^i\dot{\omega}_i \times {}^iP_{i+1} + {}^i\omega_i \times ({}^i\omega_i \times {}^iP_{i+1}) + {}^i\dot{v}_i \right] \quad (3.15)$$

siendo ${}^iP_{i+1}$ el vector que expresa la posición del origen del sistema {i+1} en el sistema {i} tal como se muestra en la Figura 3.3.

Si la articulación es prismática, hace falta añadir los términos correspondientes al desplazamiento d_i según la variable articular. Utilizando la ecuación general de la aceleración lineal es posible escribir:

$${}^{i+1}\dot{\hat{v}}_{i+1} = {}^{i+1}R({}^i\dot{\omega}_i \times {}^iP_{i+1} + {}^i\omega_i \times ({}^i\omega_i \times {}^iP_{i+1}) + {}^i\dot{v}_i) + 2{}^{i+1}\omega_{i+1} \times \dot{d}_{i+1} {}^{i+1}\hat{Z}_{i+1} + \ddot{d}_{i+1} {}^{i+1}\hat{Z}_{i+1} \quad (3.16)$$

Por ultimo, suponiendo que existe un sistema $\{C_i\}$ solidario a cada segmento, con su origen situado en el centro de masas y con la misma orientación que la $\{i\}$, la aceleración de este centro viene dada por la misma expresión que la del origen, pero en este caso expresada en el sistema $\{i\}$:

$${}^i\dot{v}_{C_i} = {}^i\dot{\omega}_i \times {}^iP_{C_i} + {}^i\omega_i \times ({}^i\omega_i \times {}^iP_{C_i}) + {}^i\dot{v}_i \quad (3.17)$$

En el segmento 1 las expresiones se simplifican ya que ${}^0\omega_0 = {}^0\dot{\omega}_0 = 0$

Habiendo calculado la aceleración del centro de masas, pueden aplicarse las ecuaciones de Newton-Euler (3.10 y 3.11) para determinar la fuerza y el par que actuaran en el centro de masas de cada enlace:

$$F_i = m\dot{v}_{C_i} \quad (3.18)$$

$$N_i = {}^{c_i}I\dot{\omega}_i + \omega_i \times {}^{c_i}I\omega_i \quad (3.19)$$

A continuación se procede a realizar las iteraciones hacia dentro; es decir, desde la última articulación a la base. Sean f_i y n_i la fuerza y el momento ejercido en i por $i-1$. El equilibrio de fuerzas en el eslabón i (Figura 3.3) es:

$${}^iF_i = {}^if_i - {}^{i+1}R{}^{i+1}f_{i+1} \quad (3.20)$$

El equilibrio de momentos en el centro de masas es:

$${}^iN_i = {}^i n_i - {}^i n_{i+1} + (-{}^iP_{C_i}) \times {}^i f_i - ({}^iP_{i+1} - {}^iP_{C_i}) \times {}^i f_{i+1} \quad (3.21)$$

Obteniéndose:

$${}^iN_i = {}^i n_i - {}^iR^{i+1} n_{i+1} - {}^iP_{C_i} \times {}^iF_i - {}^iP_{i+1} \times {}^iR^{i+1} f_{i+1} \quad (3.22)$$

Estas ecuaciones se aplican desde la última hasta la primera articulación. Si el extremo del manipulador esta libre, se tendrá: ${}^{N+1}f_{N+1} = 0$, ${}^{N+1}n_{N+1} = 0$; en caso contrario, se comenzaría con los valores de las fuerzas y momentos externos que actúan en el extremo.

Las expresiones en forma iterativa son:

$${}^i f_i = {}^iF_i + {}^iR^{i+1} f_{i+1} \quad (3.23)$$

$${}^i n_i = {}^iN_i + {}^iR^{i+1} n_{i+1} + {}^iP_{C_i} \times {}^iF_i + {}^iP_{i+1} \times {}^iR^{i+1} f_{i+1} \quad (3.24)$$

El momento en una articulación rotacional es la componente \hat{Z}_i del momento aplicado por un enlace siguiente:

$$\tau_i = {}^i n_i^T \hat{Z}_i \quad (3.25)$$

De la misma forma, la fuerza en una articulación prismática es:

$$\tau_i = {}^i f_i^T \hat{Z}_i \quad (3.26)$$

donde hemos utilizado el símbolo τ para una fuerza de actuador lineal.

3.4.2. EL ALGORITMO ITERATIVO DE NEWTON-EULER

El algoritmo completo para computar los torques en las articulaciones esta compuesto de dos partes. Primero, se calculan las velocidades y aceleraciones de los enlaces directamente desde el enlace 1 al n , aplicando las ecuaciones de Newton-Euler en cada enlace. En la segunda, se calculan las fuerzas y momentos de iteración y finalmente los momentos que hay que aplicar en los actuadores de las articulaciones, todo ello comenzando en la articulación n y terminando en la 1. Las ecuaciones que están resumidas abajo, son para el caso en el que todas las articulaciones son rotacionales [1].

Iteraciones hacia fuera: $i = 0 \rightarrow 3$

$${}^{i+1}\omega_{i+1} = {}^i R^{i+1} \omega_i + \dot{\theta}_{i+1} {}^{i+1} \hat{Z}_{i+1} \quad (3.27)$$

$${}^{i+1}\dot{\omega}_{i+1} = {}^{i+1} R^i \dot{\omega}_i + {}^{i+1} R^i \omega_i \times \dot{\theta}_{i+1} {}^{i+1} \hat{Z}_{i+1} + \ddot{\theta}_{i+1} {}^{i+1} \hat{Z}_{i+1} \quad (3.28)$$

$${}^{i+1}\dot{v}_{i+1} = {}^{i+1} R \left({}^i \dot{\omega}_i \times {}^i P_{i+1} + {}^i \omega_i \times ({}^i \omega_i \times {}^i P_{i+1}) + {}^i \dot{v}_i \right) \quad (3.29)$$

$${}^{i+1}\dot{v}_{C_{i+1}} = {}^{i+1} \dot{\omega}_{i+1} \times {}^{i+1} P_{C_{i+1}} + {}^{i+1} \omega_{i+1} \times ({}^{i+1} \omega_{i+1} \times {}^{i+1} P_{C_{i+1}}) + {}^{i+1}\dot{v}_{i+1} \quad (3.30)$$

$${}^{i+1} F_{i+1} = m_{i+1} {}^{i+1}\dot{v}_{C_{i+1}} \quad (3.31)$$

$${}^{i+1} N_{i+1} = {}^{C_{i+1}} I_{i+1} {}^{i+1} \dot{\omega}_{i+1} + {}^{i+1} \omega_{i+1} \times {}^{C_{i+1}} I_{i+1} {}^{i+1} \omega_{i+1} \quad (3.32)$$

Iteraciones hacia dentro: $i = 4 \rightarrow 1$

$${}^i f_i = {}^i F_i + {}^i R^{i+1} f_{i+1} \quad (3.33)$$

$${}^i n_i = {}^i N_i + {}^i R^{i+1} n_{i+1} + {}^i P_{C_i} \times {}^i F_i + {}^i P_{i+1} \times {}^i R^{i+1} f_{i+1} \quad (3.34)$$

$$\tau_i = {}^i n_i^T {}^i \hat{Z}_i \quad (3.35)$$

El efecto de la carga gravitacional sobre los eslabones puede ser incluida haciendo ${}^0\dot{v}_o = G$, donde G es el vector gravedad. Esto es equivalente a decir que la base del manipulador esta acelerada hacia arriba con la aceleración de 1 G. Esta aceleración ficticia causa el mismo efecto sobre los eslabones que el de la gravedad, sin generar un cálculo extra.

Las ecuaciones presentadas pueden utilizarse como método numérico o de forma analítica para desarrollar ecuaciones simbólicas. El empleo como método numérico puede aplicarse de forma general para cualquier robot manipulador. Una vez que se han especificado tensores de inercia, masas de enlaces, vectores P_C y matrices ${}^{i+1}R_i$ de un determinado manipulador, las ecuaciones pueden aplicarse directamente para calcular los momentos de las articulaciones en cualquier movimiento.

3.5. PARÁMETROS FÍSICOS Y GEOMÉTRICOS DEL ROBOTARM MR999

Para poder utilizar los modelos dinámicos descritos anteriormente es necesario conocer algunos parámetros físicos y geométricos del manipulador estudiado como: matriz de parámetros de Denavit-Hartenberg¹², la masa de cada eslabón, ubicación del centro de masas, los momentos de inercia de cada eslabón respecto a ejes convenientes.

Con esto en mente y teniendo en cuenta la complejidad geométrica del RobotArm MR 999, lo que se traduce en una compleja forma de determinar estos parámetros, se optó por emplear el CAD SolidEdge V12, para la

¹² Definida en el Capítulo 2

obtención de estos datos, basados en el modelo 3D geométrico detallado (ver Figura 3.4) elaborado con esta herramienta CAD¹³.

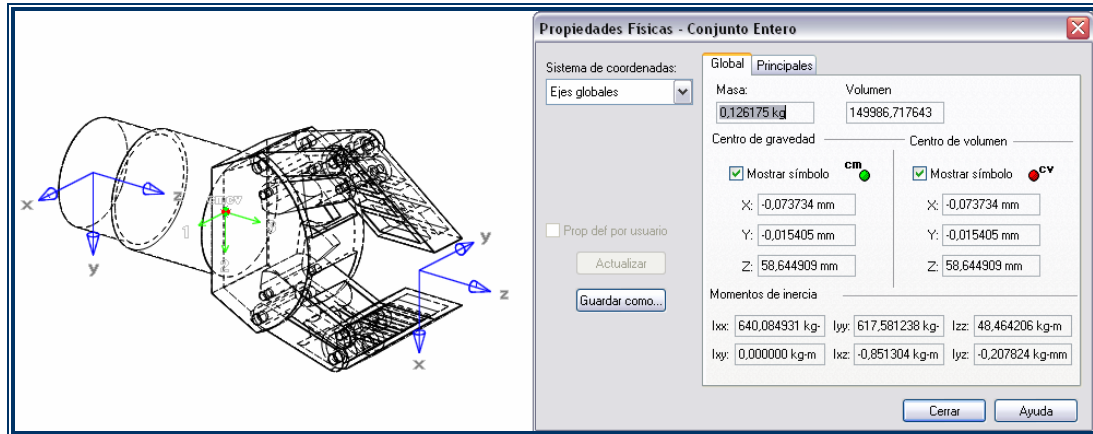


Figura 3.4. Vista de la herramienta del CAD para el cálculo de propiedades físicas y geométricas.

La tabla 3, resume los datos obtenidos empleando el software mencionado y una balanza de precisión, para abordar detalles referirse a los anexos citados.

Parámetro	Base	Eslabón 1	Eslabón 2	Gripper	
Masa [Kg]	0.487	0.184	0.111	0.126	
Coordenadas del centro de masa. [m]	r_x	-0.0071	0.05289	0.00206	0
	r_y	0	-0.00173	0.05118	0
	r_z	-0.06351	0	0	0
Momentos de Inercia [Kg-m ²]	I_{xx}	0.0098	0.0005	0.0009	0.0063
	I_{yy}	0.0075	0.0028	0.0003	0.0126
	I_{zz}	0.0061	0.0027	0.0009	0.0126
	I_{xy}	0	-0.0001	-0.0001	0
	I_{yz}	0	0	0	0
	I_{xz}	-0.0007	0	0	0

Tabla 3. Parámetros físicos y geométricos utilizados en el modelo dinámico

¹³ Planos, piezas y ensamble adjuntos en Anexo.

3.6. MODELO DINÁMICO DEL ROBOTARM MR 999

3.6.1. DINÁMICA DIRECTA

El problema de este apartado consiste en encontrar la posición, velocidad y aceleración teniendo como variables de entrada los torques aplicados en cada una de las articulaciones. Es decir, obtener el histórico de variables articulares como funciones del momento aplicado en cada una.

Si esto es lo que se quiere, se puede acudir a la ecuación generalizada (3.9), $\tau = M(\theta)\ddot{\theta} + F(\dot{\theta}) + V(\theta, \dot{\theta}) + G(\theta)$, de la cual se debe advertir varias cosas:

Aquí, $\tau = [\tau_1, \dots, \tau_n]^T$ representa los momentos ejercidos por los actuadores en las n articulaciones, $\theta = [\theta_1, \dots, \theta_n]^T$ es el vector de variables articulares, $M(\theta)$ la matriz $n \times n$ de masas del manipulador, $V(\theta, \dot{\theta})$ los momentos ejercidos por las fuerzas centrífugas y de Coriolis, $F(\dot{\theta})$ los momentos debidos a las fricciones y $G(\theta)$ los debidos a la gravedad.

Aprovechando las particularidades de la matriz $M(\theta)$, es decir que es simétrica y siempre invertible, podemos reescribir la ecuación (3.9) de la siguiente manera:

$$\ddot{\theta} = M^{-1}(\theta) [\tau - F(\dot{\theta}) - V(\theta, \dot{\theta}) - G(\theta)] \quad (3.36)$$

que puede resolverse mediante integración numérica suponiendo conocidas las condiciones iniciales: $\theta(0) = \theta_o$; $\dot{\theta}(0) = 0$; $\ddot{\theta}(0) = 0$.

Teniendo esto en mente, se empleó el software Simulink para simular el comportamiento dinámico del manipulador RobotArm MR 999, para ello es necesario implementar la ecuación (3.36) como un diagrama de bloques. En la Figura 3.5 se observa el diagrama que se dispuso para el caso en el que los torques de entrada son constantes¹⁴.

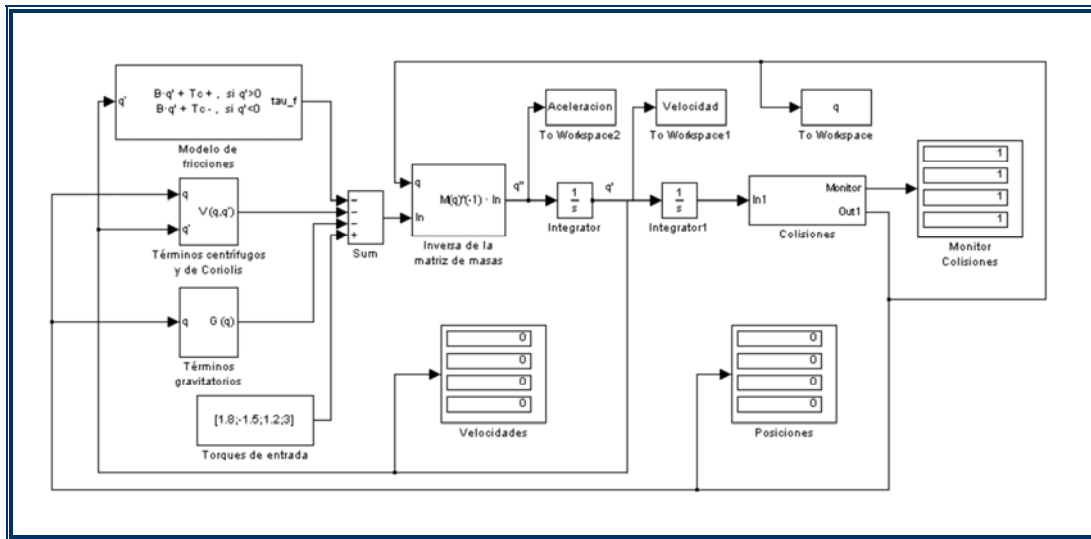


Figura 3.5. Diagrama de bloques en Simulink que implementa la ecuación (3.36)

Como se puede ver en la figura, los bloques se han ensamblado conforme a la ecuación (3.36); veamos en detalle algunos aspectos importantes de este diagrama.

En el bloque *Torques de entrada*, se introducen, haciendo doble click sobre el mismo, los valores de torque para cada una de las variables articulares.

El bloque *Modelo de fricciones* implementa la ecuación (3.7) del apartado 3.3 del presente capítulo, asimismo los bloques *Términos centrífugos y de Coriolis* y *Términos gravitatorios* se rigen por sus respectivas funciones¹⁵.

¹⁴ Este diagrama de bloques se incluye en el anexo como *TorquesConstantes.mdl*

¹⁵ Estos tres bloques junto con *Inversa de la matriz de masas* emplean funciones-s respectivas del toolbox HEMERO.

Estos cuatro bloques sumados, con su correspondiente signo (ver bloque *Sum* en la Figura 3.5), conforman el paréntesis del miembro derecho en la ecuación (3.36), ahora este valor es operado por el bloque *Inversa de la matriz de masas*, de esta forma se obtiene $\ddot{\theta}^{16}$, que seguidamente es integrado por el bloque *Integrator*, lográndose, almacenándose y visualizándose $\dot{\theta}$, que inmediatamente sirve de valor de retroalimentación para los bloques que lo solicitan, y además es nuevamente integrado por un segundo bloque *Integrator1* que entrega el valor de θ que es evaluado por el subsistema *Colisiones*, quien se encarga de confrontar este valor con los límites alcanzables por el manipulador (Ver Figura 3.6) al salir de este subsistema obviamente es almacenado, visualizado y retroalimentado a los bloques solicitantes.

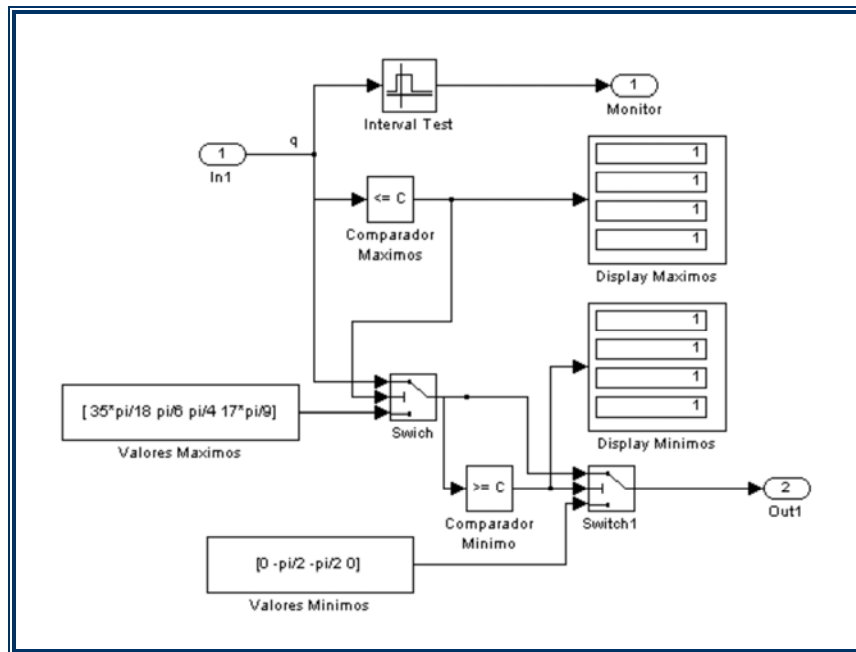


Figura 3.6. Subsistema Colisiones. Consiste en un comparador de máximos y mínimos puesto en serie, que evalúa el valor de cada una de las variables articulares y discrimina cuando alguna supera un límite; visualizándolo en los displays y enviando una señal al *Monitor de colisiones*.

¹⁶ En Simulink como en MATLAB, θ , $\dot{\theta}$ y $\ddot{\theta}$ son llamados q , q' y q'' respectivamente

Del mismo modo se implementó el diagrama de bloques en Simulink *TorquesVariables.mdl* en la Figura 3.7 que se rige por la misma ecuación, pero en el que se han dispuesto como entradas torques variables en el tiempo $\tau = \tau(t)$. El bloque *Tiempo* genera el parámetro para evaluar las funciones de entrada, estos valores son llevados al subsistema *Trayectorias articulares* cuyo principio es el mismo que el anterior y su contenido se ve en la Figura 3.8.

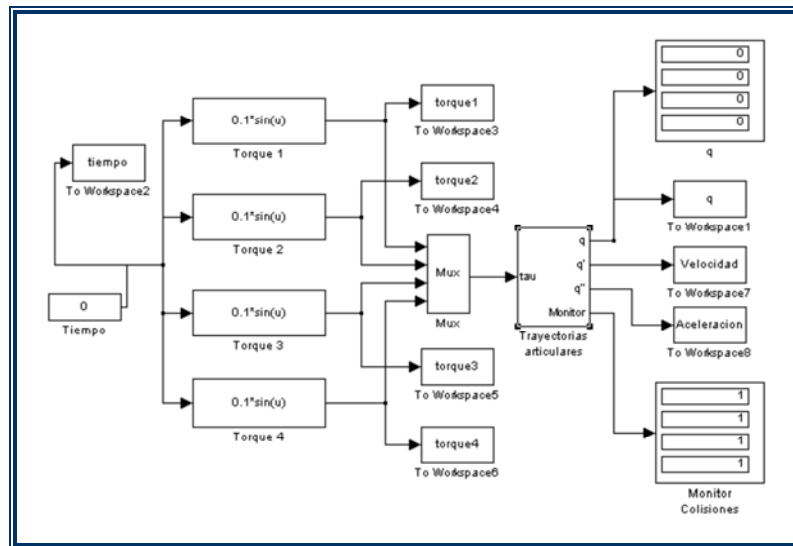


Figura 3.7. Diagrama de bloques para entrada de torques variables en el tiempo

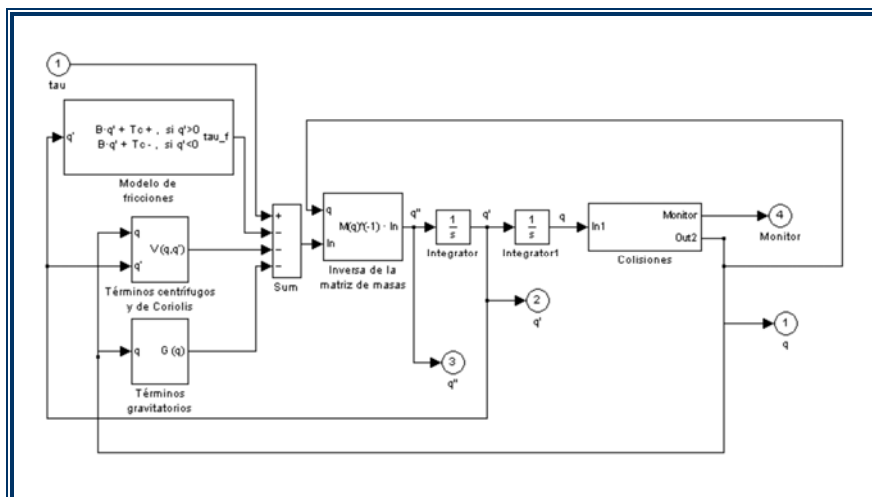


Figura 3.8. Subsistema *Trayectorias articulares*.

3.6.2. DINÁMICA INVERSA

Ahora podemos aplicar lo expuesto en los apartados inmediatamente anteriores, al manipulador ROBOT ARM MR 999 para obtener las expresiones simbólicas de los torques en cada articulación como una función de sus variables articulares, a saber, posición, velocidad y aceleración. Como se mencionó anteriormente el método de Newton – Euler es ventajoso por su naturaleza iterativa, pero es muy engorroso y susceptible a demasiadas equivocaciones si se tratara de desarrollar manualmente. Para solucionar lo anterior se utilizó la función **rne** del “Robotics Toolbox¹⁷”, donde para su aplicación se introdujeron las siguientes líneas en la ventana de comandos de Matlab:

```
syms  $\theta_1 \theta_2 \theta_3 \theta_4$  real;      % variables articulares de posición (simbólicas)
syms  $\theta'_1 \theta'_2 \theta'_3 \theta'_4$  real;  % variables articulares de velocidad
syms  $\theta''_1 \theta''_2 \theta''_3 \theta''_4$  real; % variables articulares de aceleración
syms A1 A2 A3 real;             % longitudes de los enlaces
syms m1 m2 m3 m4;              % masas de los enlaces
syms g real;                    % aceleración de la gravedad
```

dyn = [... debe introducirse la matriz de parámetros dinámicos mostrada
en la expresión (3.36)]

```
q=[ $\theta_1 \theta_2 \theta_3 \theta_4$ ];      % vector de variables articulares de posición
qd=[ $\theta'_1 \theta'_2 \theta'_3 \theta'_4$ ]; % vector de variables articulares de velocidad
qdd=[ $\theta''_1 \theta''_2 \theta''_3 \theta''_4$ ]; % vector de variables articulares de aceleración
grav=[0 0 -g];                  % vector aceleración de la gravedad
```

¹⁷ CORKE, Peter I. Manufacturing Science and Technology, Pinjarra Hills, AUSTRALIA.2001

Torques=rne(dynARM,q,qd,qdd,grav);
simple(Torques)

De esta forma se obtiene una expresión simbólica de los torques presentes en cada articulación como función de las variables articulares.

$$\text{dyn} = \begin{bmatrix} 0 & 0 & 0 & A1 & 0 & m1 & rx1 & 0 & rz1 & Ixx1 & Iyy1 & Izz1 & 0 & 0 & Iyz1 & 0 & 1 & 0 & 0 & 0 \\ -\pi/2 & 0 & 0 & 0 & 0 & m2 & rx2 & ry2 & 0 & Ixx2 & Iyy2 & Izz2 & Ixy2 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & A2 & 0 & 0 & 0 & m3 & rx3 & ry3 & 0 & Ixx3 & Iyy3 & Izz3 & Ixy3 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ -\pi/2 & 0 & 0 & A3 & 0 & m4 & 0 & 0 & 0 & Ixx4 & Iyy4 & Izz4 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}$$

(3.36)

El resultado de aplicar la función **rne** es un vector fila de cuatro columnas donde cada una de ellas representa el torque en la respectiva articulación. A continuación se presentan las cuatro expresiones simbólicas obtenidas:

Torque 1 = $1/4 * Iyy4 * \theta'^2 * \sin(-2 * \theta4 + \theta3 + \theta2) -$
 $1/4 * Iyy4 * \theta'^2 * \sin(2 * \theta4 + \theta3 + \theta2) + 1/2 * Izz4 * \theta'' * \cos(2 * \theta3 + 2 * \theta2) + \theta' * \theta'^2 * Ixx3 * \sin(2 * \theta3 + 2 * \theta2) + 1/2 * rx2^2 * m2 * \theta'' + 1/2 * rx2^2 * m2 * \theta'' * \cos(2 * \theta2) + 1/2 * m3 * A2^2 * \theta'' + 1/2 * m3 * A2^2 * \theta'' * \cos(2 * \theta2) + 1/4 * Iyy4 * \theta'^3 * \sin(-2 * \theta4 + \theta3 + \theta2) -$
 $1/4 * Iyy4 * \theta'^3 * \sin(2 * \theta4 + \theta3 + \theta2) - \cos(\theta2) * Ixz2 * \theta'^2 -$
 $\sin(\theta2) * Ixz2 * \theta'' + 1/4 * Ixx4 * \theta' * \theta'^2 * \sin(2 * \theta4 + 2 * \theta3 + 2 * \theta2) -$
 $Iyy3 * \theta'^2 * \theta' * \sin(2 * \theta3 + 2 * \theta2) - 2 * ry2 * m2 * \theta' * \theta'^2 * rx2 * \cos(2 * \theta2) -$
 $1/4 * Ixx4 * \theta'' * \cos(2 * \theta4 + \theta3 + \theta2) + 1/4 * Ixx4 * \theta'' * \cos(-$
 $2 * \theta4 + \theta3 + \theta2) + 1/2 * Ixx4 * \theta'^4 * \theta'^2 * \sin(2 * \theta4 + \theta3 + \theta2) + 1/2 * Ixx4 * \theta'^4 * \theta'^3 * \sin(-$
 $2 * \theta4 + \theta3 + \theta2) + 1/2 * Ixx4 * \theta'^4 * \theta'^3 * \sin(2 * \theta4 + \theta3 + \theta2) - rx2^2 * m2 * \theta' * \theta'^2 * \sin(2 * \theta2) -$

$$\begin{aligned}
& m^3 \theta' \theta'^2 A^2 \sin(2\theta_2) - 1/2 A^3 \theta'^4 \cos(2\theta_3 + 2\theta_2) - \\
& 1/4 I_{yy} \theta'^3 \cos(-2\theta_4 + \theta_3 + \theta_2) - \\
& 1/4 I_{yy} \theta'^4 \cos(2\theta_3 + 2\theta_2) + 1/8 I_{yy} \theta'^4 \cos(- \\
& 2\theta_4 + 2\theta_3 + 2\theta_2) + 1/2 I_{xx} \theta'^3 + 1/4 I_{xx} \theta'^4 + 1/4 I_{yy} \theta'^4 - \\
& A^2 m^3 r_y \theta' \theta'^3 \cos(\theta_3) - A^2 m^3 r_y \theta' \theta'^3 \cos(\theta_3 + 2\theta_2) - \\
& \theta'^2 I_{yy} \theta' \sin(2\theta_2) + I_{zz} \theta'^4 \theta'^2 \sin(\theta_3 + \theta_2) + 1/2 I_{xx} \theta' \theta'^2 \sin(2\theta_3 + 2\theta_2) \\
& - 1/4 I_{yy} \theta'^4 \theta' \sin(2\theta_4 + 2\theta_3 + 2\theta_2) - \\
& I_{zz} \theta'^4 \cos(\theta_3 + \theta_2) + I_{xy} \theta'^2 \sin(2\theta_2) + r_y^2 \theta'^2 m^2 \theta' \theta'^2 \sin(2\theta_2) - \\
& r_x^3 m^3 r_y \theta'^3 \sin(2\theta_3 + 2\theta_2) - \\
& 1/4 I_{yy} \theta'^4 \theta'^2 \sin(2\theta_4 + 2\theta_3 + 2\theta_2) + 1/2 r_y^3 \theta'^2 m^3 \theta'^4 - \\
& 1/2 r_y^3 \theta'^2 m^3 \theta'^4 \cos(2\theta_3 + 2\theta_2) - \\
& r_y^2 m^2 r_x \theta'^2 \sin(2\theta_2) + r_x^3 m^3 A^2 \theta'^4 \cos(\theta_3 + 2\theta_2) - \\
& 1/2 I_{yy} \theta'^4 \theta'^2 \sin(-2\theta_4 + \theta_3 + \theta_2) - \\
& 1/2 I_{yy} \theta'^4 \theta'^2 \sin(2\theta_4 + \theta_3 + \theta_2) + 1/2 I_{xx} \theta'^4 \theta'^2 \sin(- \\
& 2\theta_4 + \theta_3 + \theta_2) + 1/2 I_{yy} \theta'^4 + 1/2 I_{yy} \theta'^4 \cos(2\theta_2) + 1/2 A^3 \theta'^4 + \theta'^4 \theta'^3 \\
& I_{xx} \sin(2\theta_3 + 2\theta_2) - \\
& 1/4 I_{yy} \theta'^4 \theta'^3 \sin(2\theta_4 + 2\theta_3 + 2\theta_2) + 1/2 I_{yy} \theta'^4 \theta'^4 \sin(2\theta_4) + 1/4 I_{yy} \theta'^4 \theta'^4 \\
& \theta' \sin(- \\
& 2\theta_4 + 2\theta_3 + 2\theta_2) + 1/2 I_{yy} \theta'^4 \theta'^2 \sin(2\theta_3 + 2\theta_2) + r_y^3 \theta'^2 m^3 \theta' \theta'^3 \sin(2\theta_3 \\
& + 2\theta_2) + 1/4 I_{xx} \theta' \theta'^3 \sin(2\theta_4 + 2\theta_3 + 2\theta_2) + 1/2 r_x^3 \theta'^2 m^3 \theta'^4 \cos(2\theta_3 + 2\theta_2) \\
& + I_{zz} \theta'^4 \theta'^3 \sin(\theta_3 + \theta_2) - 2 r_x^3 m^3 A^2 \theta' \theta'^2 \sin(\theta_3 + 2\theta_2) - \\
& 1/8 I_{xx} \theta'^4 \cos(2\theta_4 + 2\theta_3 + 2\theta_2) - \\
& 1/4 I_{xx} \theta'^3 \cos(2\theta_4 + \theta_3 + \theta_2) + 1/4 I_{xx} \theta'^3 \cos(-2\theta_4 + \theta_3 + \theta_2) - \\
& 2 r_x^3 m^3 r_y \theta' \theta'^2 \cos(2\theta_3 + 2\theta_2) - r_x^3 \theta'^2 m^3 \theta'^2 \theta' \sin(2\theta_3 + 2\theta_2) - \\
& 1/2 I_{xx} \theta'^2 \theta'^3 \sin(-2\theta_4 + \theta_3 + \theta_2) - I_{zz} \theta' \theta'^3 \sin(2\theta_3 + 2\theta_2) - \\
& 2 A^3 m^4 \theta' \theta'^2 A^2 \cos(\theta_3 + 2\theta_2) - \\
& 1/4 I_{yy} \theta'^4 \cos(2\theta_4) + 1/4 I_{yy} \theta'^4 \cos(2\theta_4 + \theta_3 + \theta_2) - 1/4 I_{yy} \theta'^4 \cos(- \\
& 2\theta_4 + \theta_3 + \theta_2) + 1/4 I_{yy} \theta'^3 \cos(2\theta_4 + \theta_3 + \theta_2) - A^2 m^4 A^3 \theta' \theta'^3 \cos(\theta_3) -
\end{aligned}$$

$$\begin{aligned}
& A2*m4*A3*\theta'1*\theta'3*\cos(\theta3+2*\theta2)-1/2*Ixx2*\theta''1*\cos(2*\theta2)- \\
& A2*m3*rx3*\theta'1*\theta'3*\sin(\theta3+2*\theta2)-A3*m4*A2*\theta''1*\sin(\theta3)- \\
& Izz4*\theta'2*\theta'1*\sin(2*\theta3+2*\theta2)-1/2*Iyy4*\theta'4*\theta'3*\sin(-2*\theta4+\theta3+\theta2)- \\
& 1/2*Iyy4*\theta'4*\theta'3*\sin(2*\theta4+\theta3+\theta2)-1/4*Iyy4*\theta'1*\theta'2*\sin(- \\
& 2*\theta4+2*\theta3+2*\theta2)+1/4*Ixx4*\theta'1*\theta'2*\sin(-2*\theta4+2*\theta3+2*\theta2)- \\
& 1/4*Iyy4*\theta'1*\theta'3*\sin(-2*\theta4+2*\theta3+2*\theta2)+1/4*Ixx4*\theta'1*\theta'3*\sin(- \\
& 2*\theta4+2*\theta3+2*\theta2)+Izz1*\theta''1- \\
& 1/2*Ixx3*\theta''1*\cos(2*\theta3+2*\theta2)+1/8*Iyy4*\theta''1*\cos(2*\theta4+2*\theta3+2*\theta2)+Ixy3*\theta''1*si \\
& n(2*\theta3+2*\theta2)+A3^2*m4*\theta'1*\theta'3*\sin(2*\theta3+2*\theta2)- \\
& 2*rx3*m3*ry3*\theta'3*\theta'1*\cos(2*\theta3+2*\theta2)+A3^2*m4*\theta'2*\theta'1*\sin(2*\theta3+2*\theta2)- \\
& ry3*m3*A2*\theta''1*\sin(\theta3)+1/4*Ixx4*\theta''1*\cos(2*\theta4)-1/4*Ixx4*\theta''1*\cos(2*\theta3+2*\theta2)- \\
& 1/8*Ixx4*\theta''1*\cos(-2*\theta4+2*\theta3+2*\theta2)-1/4*Ixx4*\theta'4*\theta'1*\sin(- \\
& 2*\theta4+2*\theta3+2*\theta2)+rx1^2*m1*\theta''1+rx3*m3*A2*\theta''1*\cos(\theta3)+1/2*Ixx2*\theta''1- \\
& Iyy3*\theta'1*\theta'3*\sin(2*\theta3+2*\theta2)+1/2*Ixx4*\theta'2*\theta'3*\sin(2*\theta4+\theta3+\theta2)+1/2*Iyy4*\theta'2* \\
& \theta'3*\sin(-2*\theta4+\theta3+\theta2)- \\
& 1/2*Iyy4*\theta'2*\theta'3*\sin(2*\theta4+\theta3+\theta2)+1/4*Ixx4*\theta'4*\theta'1*\sin(2*\theta4+2*\theta3+2*\theta2)+1/2* \\
& Ixx4*\theta'1*\theta'3*\sin(2*\theta3+2*\theta2)+1/2*m4*A2^2*\theta''1+1/2*m4*A2^2*\theta''1*\cos(2*\theta2)+ \\
& ry3^2*m3*\theta'2*\theta'1*\sin(2*\theta3+2*\theta2)+1/4*Ixx4*\theta'3^2*\sin(2*\theta4+\theta3+\theta2)+1/2*Iyy4* \\
& \theta'1*\theta'3*\sin(2*\theta3+2*\theta2)-1/2*Ixx4*\theta'4*\theta'1*\sin(2*\theta4)+Ixx2*\theta'1*\theta'2*\sin(2*\theta2)- \\
& 1/2*ry2^2*m2*\theta''1*\cos(2*\theta2)+2*Ixy2*\theta'1*\theta'2*\cos(2*\theta2)+1/2*Iyy3*\theta''1+1/2*Izz \\
& 4*\theta''1+1/2*ry2^2*m2*\theta''1- \\
& rx3^2*m3*\theta'1*\theta'3*\sin(2*\theta3+2*\theta2)+1/2*rx3^2*m3*\theta''1- \\
& m4*\theta'1*\theta'2*A2^2*\sin(2*\theta2)-1/4*Ixx4*\theta'2^2*\sin(- \\
& 2*\theta4+\theta3+\theta2)+1/4*Ixx4*\theta'2^2*\sin(2*\theta4+\theta3+\theta2)-A3*m4*A2*\theta''1*\sin(\theta3+2*\theta2)- \\
& ry3*m3*A2*\theta''1*\sin(\theta3+2*\theta2)+1/2*Iyy3*\theta''1*\cos(2*\theta3+2*\theta2)- \\
& 1/4*Ixx4*\theta'3^2*\sin(-2*\theta4+\theta3+\theta2)- \\
& A2*m3*rx3*\theta'1*\theta'3*\sin(\theta3)+2*Ixy3*\theta'1*\theta'2*\cos(2*\theta3+2*\theta2)+2*Ixy3*\theta'3*\theta'1*\cos(\\
& 2*\theta3+2*\theta2)-2*ry3*m3*\theta'1*\theta'2*A2*\cos(\theta3+2*\theta2).
\end{aligned}$$

$$\begin{aligned}
\text{Torque 2} = & 1/2*\theta'1^2*Iyy3*\sin(2*\theta3+2*\theta2)- \\
& Iyy4*\theta'4*\theta'2*\sin(2*\theta4)+Ixx4*\theta'4*\theta'2*\sin(2*\theta4)+1/2*\theta'1^2*Iyy2*\sin(2*\theta2)- \\
& \theta'1^2*Ixy2*\cos(2*\theta2)-1/2*\theta'1^2*Ixx2*\sin(2*\theta2)- \\
& \theta'1^2*Ixy3*\cos(2*\theta3+2*\theta2)+1/2*rx3^2*m3*\theta'1^2*\sin(2*\theta3+2*\theta2)- \\
& rx3*m3*g*\cos(\theta3+\theta2)+1/2*Ixx4*\theta''2- \\
& 1/2*\theta'1^2*Ixx3*\sin(2*\theta3+2*\theta2)+m3*\theta''2*A2^2+m4*A3^2*\theta''3- \\
& 2*A3*m4*\sin(\theta3)*\theta''2*A2+m4*\theta''2*A2^2+m4*A3^2*\theta''2+1/4*Ixx4*\theta''1*\cos(- \\
& 2*\theta4+\theta3+\theta2)-2*A2*\sin(\theta3)*m3*rx3*\theta'2*\theta'3- \\
& 1/2*ry2^2*m2*\theta'1^2*\sin(2*\theta2)+1/2*Ixx4*\theta'4*\theta'1*\sin(- \\
& 2*\theta4+\theta3+\theta2)+1/2*Ixx4*\theta'4*\theta'1*\sin(2*\theta4+\theta3+\theta2)+1/4*Iyy4*\theta''1*\cos(2*\theta4+\theta3+\theta \\
& 2)-A2*\sin(\theta3)*m4*A3*\theta''3-A2*m3*\cos(\theta2)*g-A2*m4*\cos(\theta2)*g- \\
& 1/2*Ixx4*\theta''3*\cos(2*\theta4)-1/2*Ixx4*\theta''2*\cos(2*\theta4)- \\
& 2*A2*\cos(\theta3)*m3*ry3*\theta'2*\theta'3+A3*m4*\theta'1^2*A2*\cos(\theta3+2*\theta2)- \\
& 1/2*A3^2*m4*\theta'1^2*\sin(2*\theta3+2*\theta2)+A3*m4*g*\sin(\theta3+\theta2)+Ixx4*\theta'4*\theta'3*\sin(2* \\
& \theta4)-Iyy4*\theta'4*\theta'3*\sin(2*\theta4)-1/8*Ixx4*\theta'1^2*\sin(-2*\theta4+2*\theta3+2*\theta2)- \\
& 1/4*Iyy4*\theta''1*\cos(-2*\theta4+\theta3+\theta2)+1/2*rx2^2*m2*\theta'1^2*\sin(2*\theta2)- \\
& 1/4*Ixx4*\theta''1*\cos(2*\theta4+\theta3+\theta2)+1/2*Izz4*\theta'1^2*\sin(2*\theta3+2*\theta2)+1/2*Ixx4*\theta''3+ \\
& 1/2*Iyy4*\theta''3*\cos(2*\theta4)+rx2*m2*\theta'1^2*ry2*\cos(2*\theta2)- \\
& 1/2*Iyy4*\theta'4*\theta'1*\sin(2*\theta4+\theta3+\theta2)- \\
& 1/2*ry3^2*m3*\theta'1^2*\sin(2*\theta3+2*\theta2)+ry3*m3*g*\sin(\theta3+\theta2)- \\
& 1/8*Ixx4*\theta'1^2*\sin(2*\theta4+2*\theta3+2*\theta2)- \\
& 1/4*Ixx4*\theta'1^2*\sin(2*\theta3+2*\theta2)+1/2*Iyy4*\theta''2+1/2*Iyy4*\theta''2*\cos(2*\theta4)+1/2*Iy \\
& y4*\theta''3-1/2*Iyy4*\theta'4*\theta'1*\sin(- \\
& 2*\theta4+\theta3+\theta2)+rx3^2*m3*\theta''2+rx3^2*m3*\theta''3+ry3^2*m3*\theta''2+ry3^2*m3*\theta''3+rx \\
& 2^2*m2*\theta''2+ry2^2*m2*\theta''2-Ixz2*\sin(\theta2)*\theta''1+Izz3*\theta''2+Izz3*\theta''3- \\
& rx2*m2*\cos(\theta2)*g+ry2*m2*\sin(\theta2)*g+1/2*m3*\theta'1^2*A2^2*\sin(2*\theta2)+1/2*m4*\theta \\
& '1^2*A2^2*\sin(2*\theta2)-2*A2*\cos(\theta3)*m4*A3*\theta'2*\theta'3-A2*\sin(\theta3)*m3*ry3*\theta''3-
\end{aligned}$$

$$\begin{aligned}
& A^2 \sin(\theta_3) m^3 r_x^3 \theta'^3 + A^2 \cos(\theta_3) m^3 r_x^3 \theta''^3 - A^2 \cos(\theta_3) m^3 r_y^3 \theta'^3 - \\
& A^2 \cos(\theta_3) m^4 A^3 \theta'^3 + 1/8 I_{yy}^4 \theta'^1 \wedge^2 \sin(2\theta_4 + 2\theta_3 + 2\theta_2) - \\
& 1/4 I_{yy}^4 \theta'^1 \wedge^2 \sin(2\theta_3 + 2\theta_2) + 1/8 I_{yy}^4 \theta'^1 \wedge^2 \sin(- \\
& 2\theta_4 + 2\theta_3 + 2\theta_2) + I_{zz}^2 \theta''^2 + r_x^3 m^3 \theta'^1 \wedge^2 r_y^3 \cos(2\theta_3 + 2\theta_2) + r_x^3 m^3 \theta'^1 \wedge^2 \\
& A^2 \sin(\theta_3 + 2\theta_2) + 2 r_x^3 m^3 \cos(\theta_3) \theta''^2 A^2 - \\
& 2 r_y^3 m^3 \sin(\theta_3) \theta''^2 A^2 + r_y^3 m^3 \theta'^1 \wedge^2 A^2 \cos(\theta_3 + 2\theta_2) - \\
& I_{zz}^4 \theta'^4 \theta'^1 \sin(\theta_3 + \theta_2).
\end{aligned}$$

$$\begin{aligned}
\mathbf{Torque\ 3} &= 1/2 \theta'^1 \wedge^2 I_{yy}^3 \sin(2\theta_3 + 2\theta_2) - \\
& I_{yy}^4 \theta'^4 \theta'^2 \sin(2\theta_4) + I_{xx}^4 \theta'^4 \theta'^2 \sin(2\theta_4) - \\
& \theta'^1 \wedge^2 I_{xy}^3 \cos(2\theta_3 + 2\theta_2) + 1/2 r_x^3 \wedge^2 m^3 \theta'^1 \wedge^2 \sin(2\theta_3 + 2\theta_2) - \\
& r_x^3 m^3 g \cos(\theta_3 + \theta_2) + 1/2 I_{xx}^4 \theta''^2 - \\
& 1/2 \theta'^1 \wedge^2 I_{xx}^3 \sin(2\theta_3 + 2\theta_2) + m^4 A^3 \wedge^2 \theta''^3 - \\
& A^3 m^4 \sin(\theta_3) \theta''^2 A^2 + 1/2 A^2 m^3 \theta'^1 \wedge^2 r_y^3 \cos(\theta_3) + 1/2 A^2 m^3 \sin(\theta_3) \theta' \\
& \wedge^2 r_x^3 + 1/2 A^2 m^4 A^3 \theta'^1 \wedge^2 \cos(\theta_3) + m^4 A^3 \wedge^2 \theta''^2 + 1/4 I_{xx}^4 \theta''^1 \cos(- \\
& 2\theta_4 + \theta_3 + \theta_2) + 1/2 I_{xx}^4 \theta'^4 \theta'^1 \sin(- \\
& 2\theta_4 + \theta_3 + \theta_2) + 1/2 I_{xx}^4 \theta'^4 \theta'^1 \sin(2\theta_4 + \theta_3 + \theta_2) + 1/4 I_{yy}^4 \theta''^1 \cos(2\theta_4 + \theta_3 + \theta_2) - \\
& 1/2 I_{xx}^4 \theta''^3 \cos(2\theta_4) - \\
& 1/2 I_{xx}^4 \theta''^2 \cos(2\theta_4) + 1/2 A^3 m^4 \theta'^1 \wedge^2 A^2 \cos(\theta_3 + 2\theta_2) - \\
& 1/2 A^3 \wedge^2 m^4 \theta'^1 \wedge^2 \sin(2\theta_3 + 2\theta_2) + A^3 m^4 g \sin(\theta_3 + \theta_2) + I_{xx}^4 \theta'^4 \theta'^3 \sin(2\theta_4) - \\
& I_{yy}^4 \theta'^4 \theta'^3 \sin(2\theta_4) - 1/8 I_{xx}^4 \theta'^1 \wedge^2 \sin(-2\theta_4 + 2\theta_3 + 2\theta_2) - \\
& 1/4 I_{yy}^4 \theta''^1 \cos(-2\theta_4 + \theta_3 + \theta_2) + r_x^3 m^3 \sin(\theta_3) \theta'^2 \wedge^2 A^2 - \\
& 1/4 I_{xx}^4 \theta''^1 \cos(2\theta_4 + \theta_3 + \theta_2) + r_y^3 m^3 \cos(\theta_3) \theta'^2 \wedge^2 A^2 + 1/2 I_{zz}^4 \theta'^1 \wedge^2 \sin(\\
& 2\theta_3 + 2\theta_2) + 1/2 I_{xx}^4 \theta''^3 + 1/2 I_{yy}^4 \theta''^3 \cos(2\theta_4) - \\
& 1/2 I_{yy}^4 \theta'^4 \theta'^1 \sin(2\theta_4 + \theta_3 + \theta_2) - \\
& 1/2 r_y^3 \wedge^2 m^3 \theta'^1 \wedge^2 \sin(2\theta_3 + 2\theta_2) + r_y^3 m^3 g \sin(\theta_3 + \theta_2) - \\
& 1/8 I_{xx}^4 \theta'^1 \wedge^2 \sin(2\theta_4 + 2\theta_3 + 2\theta_2) - \\
& 1/4 I_{xx}^4 \theta'^1 \wedge^2 \sin(2\theta_3 + 2\theta_2) + 1/2 I_{yy}^4 \theta''^2 + 1/2 I_{yy}^4 \theta''^2 \cos(2\theta_4) + 1/2 I_y
\end{aligned}$$

$$\begin{aligned}
& y4*\theta''^3-1/2*Iyy4*\theta^4*\theta'1*\sin(- \\
& 2*\theta4+\theta3+\theta2)+rx3^2*m3*\theta''^2+rx3^2*m3*\theta''^3+ry3^2*m3*\theta''^2+ry3^2*m3*\theta''^3+Iz \\
& z3*\theta''^2+Izz3*\theta''^3+A2*\cos(\theta3)*m4*A3*\theta^2^2+1/8*Iyy4*\theta'1^2*\sin(2*\theta4+2*\theta3+2* \\
& \theta2)-1/4*Iyy4*\theta'1^2*\sin(2*\theta3+2*\theta2)+1/8*Iyy4*\theta'1^2*\sin(- \\
& 2*\theta4+2*\theta3+2*\theta2)+rx3*m3*\theta'1^2*ry3*\cos(2*\theta3+2*\theta2)+1/2*rx3*m3*\theta'1^2*A2*\sin \\
& (\theta3+2*\theta2)+rx3*m3*\cos(\theta3)*\theta''^2*A2- \\
& ry3*m3*\sin(\theta3)*\theta''^2*A2+1/2*ry3*m3*\theta'1^2*A2*\cos(\theta3+2*\theta2)- \\
& Izz4*\theta^4*\theta'1*\sin(\theta3+\theta2).
\end{aligned}$$

Torque 4 = -

$$\begin{aligned}
& Izz4*\theta'1*\cos(\theta3+\theta2)+Izz4*\theta'1*\theta^2*\sin(\theta3+\theta2)+Izz4*\theta'1*\theta^3*\sin(\theta3+\theta2)+Izz4*\theta'' \\
& 4+1/2*Iyy4*\theta'1*\theta^2*\sin(2*\theta4+\theta3+\theta2)+1/2*Iyy4*\theta'1*\theta^3*\sin(- \\
& 2*\theta4+\theta3+\theta2)+1/2*Iyy4*\theta'1*\theta^2*\sin(- \\
& 2*\theta4+\theta3+\theta2)+1/2*Iyy4*\theta'1*\theta^3*\sin(2*\theta4+\theta3+\theta2)+Iyy4*\theta^2*\theta^3*\sin(2*\theta4)+1/8*I \\
& yy4*\theta'1^2*\sin(2*\theta4+2*\theta3+2*\theta2)-1/8*Iyy4*\theta'1^2*\sin(-2*\theta4+2*\theta3+2*\theta2)- \\
& 1/4*Iyy4*\theta'1^2*\sin(2*\theta4)+1/2*Iyy4*\theta^2^2*\sin(2*\theta4)+1/2*Iyy4*\theta^3^2*\sin(2*\theta4) \\
& +1/8*Ixx4*\theta'1^2*\sin(-2*\theta4+2*\theta3+2*\theta2)-1/2*Ixx4*\theta'1*\theta^2*\sin(- \\
& 2*\theta4+\theta3+\theta2)+1/4*Ixx4*\theta'1^2*\sin(2*\theta4)-1/2*Ixx4*\theta^2^2*\sin(2*\theta4)- \\
& 1/2*Ixx4*\theta'1*\theta^3*\sin(2*\theta4+\theta3+\theta2)-1/2*Ixx4*\theta'1*\theta^2*\sin(2*\theta4+\theta3+\theta2)- \\
& 1/8*Ixx4*\theta'1^2*\sin(2*\theta4+2*\theta3+2*\theta2)-1/2*Ixx4*\theta'1*\theta^3*\sin(-2*\theta4+\theta3+\theta2)- \\
& Ixx4*\theta^2*\theta^3*\sin(2*\theta4)-1/2*Ixx4*\theta^3^2*\sin(2*\theta4).
\end{aligned}$$

Como puede observarse, las expresiones resultantes para los torques no son nada simples, por esto debe notarse la importancia de la gran ayuda que presta una herramienta como Robotics Toolbox.

Los valores de las propiedades físicas y geométricas incluidas en las expresiones anteriores pueden encontrarse en forma resumida en la Tabla 2

Con miras a obtener la simulación dinámica del manipulador ROBOT ARM se implementó la función **dinámica.m** involucrada dentro de la aplicación realizada para tal fin, la cual, haciendo uso de los resultados de la función **rne** y teniendo como parámetros de entrada los valores numéricos de las doce variables articulares para un instante, entrega como salida los cuatro valores numéricos de los torques ejercidos sobre cada articulación en el instante estudiado.

4. GENERACIÓN DE TRAYECTORIAS

4.1. CONSIDERACIONES GENERALES

En este capítulo, nos concentraremos en calcular trayectorias en el espacio multidimensional que son descritas en el movimiento propuesto de un manipulador. Aquí, *trayectoria* se refiere al histórico temporal de posición, velocidad y aceleración para cada grado de libertad¹⁸.

Es importante tratar el problema de cómo especificar debidamente la trayectoria deseada, podría por ejemplo, solamente indicarse el punto destino y dejar que el programa, automáticamente genere la trayectoria como resultado de la ejecución del programa de planificación.

Una forma común de provocar que el manipulador se mueva de un punto a otro en una manera suave y controlada es causar que cada junta se mueva, como se ha especificado, por una función suave del tiempo. Comúnmente cada junta empieza y termina su movimiento al mismo tiempo tal que el movimiento del manipulador parece coordinado. Exactamente, cómo calcular esas funciones de movimiento es el problema de la generación de trayectorias.

Frecuentemente un camino es descrito no solo por un destino deseado sino también por ubicaciones intermedias, o puntos vía, a través de los cuales el manipulador debe pasar en la ruta a su destino. Restringir el tiempo de duración es también otra forma de definir una trayectoria, tal como:

¹⁸ CRAIG, Jhon J. Introduction to robotics: Mechanics and control. Addison-Wesley. 1989

Ir al punto B pasando por C en 6 segundos

Es un caso en el que se limita un tiempo de duración de trayectoria y además se exige el paso por un punto intermedio (C). Puede restringirse igualmente la duración de cada tramo de trayectoria, por ejemplo:

Ir al punto B vía C, D, E con duraciones de 2, 6, 4 segundos respectivamente.

Con lo cual se aumenta la complejidad del modelo de generación de trayectorias, y podría producir discontinuidades o pasos bruscos de un tramo a otro, por los previsibles condicionamientos sobre la velocidad y la aceleración

4.2. ESTRATEGIAS PARA LA GENERACIÓN DE TRAYECTORIAS

Existen dos formas básicas para especificar el movimiento que debe realizar un manipulador para alcanzar un objetivo deseado, a saber:

1. El denominado *control punto a punto* que se realiza suministrando puntos consecutivos e ignorando la trayectoria espacial que debe seguir describir el robot entre cada dos puntos. Ésta solo es de interés cuando los puntos están suficientemente separados ya que en caso contrario, la especificación requerirá de un gran trabajo¹⁹. Por otra parte, los puntos no pueden estar demasiado distantes ya que el riesgo de generar movimientos descontrolados o imprevisibles es amplio.

¹⁹ OLLERO, Aníbal. Robótica, manipuladores y robots móviles. Alfaomega. 2003

2. La estrategia denominada como *control de trayectoria continua* en la cual se especifica el camino que debe unir los puntos mediante una determinada trayectoria, tal como una línea recta o un círculo, que debe describir el robot en el espacio de trabajo.

Suponiendo que la trayectoria que debe seguir el robot se especifica en el espacio cartesiano, existen dos alternativas:

1. Definir los lazos de control directamente en el espacio cartesiano y controlar al robot para que se anule el error de seguimiento de la trayectoria en este espacio.
2. Transformar la trayectoria del espacio cartesiano al espacio de las variables articulares y definir los lazos de control en este espacio.

La primera alternativa es usual en robots móviles y en manipuladores de configuración cartesiana, pero no para las demás configuraciones, el manipulador RobotArm MR999, se ajusta mejor a la segunda alternativa, es decir, recurriendo al modelo cinemático inverso para obtener las variables articulares que son las que definen los lazos de control. De esta manera, el problema de generación de trayectorias se plantea comúnmente en el espacio articular, en cuyo caso se trata de especificar la posición, velocidad y aceleración para cada una de las articulaciones.

Para la elección de la estrategia a utilizar debe tenerse en cuenta que las trayectorias generadas deben tener buenas propiedades en cuanto se refiere a su ejecución por el robot. Por consiguiente, es necesario tener en cuenta, directa o indirectamente, las restricciones cinemáticas involucradas. En general, las trayectorias deben ser suaves, lo que implica restricciones sobre

las derivadas a obtener con el método utilizado. Normalmente se exige que al menos la primera derivada sea continua, pudiendo exigirse también la continuidad en derivadas de orden superior²⁰.

Hay que recordar también que el problema de la generación de trayectorias debe resolverse en tiempo real, por tanto, se trata también de que la generación de trayectorias sea computacionalmente eficiente. En robots manipuladores, la generación de trayectorias articulares suele realizarse en tiempos del orden de los milisegundos o decenas de milisegundos.

4.3. GENERACIÓN DE TRAYECTORIAS PARA MANIPULADORES

Se expondrán aquí brevemente los dos enfoques mas empleados en la generación de trayectorias para manipuladores. Argumentando la opción elegida para nuestro caso de estudio.

Como ya se ha anotado el problema del planeado de trayectorias, puede trazarse en el espacio cartesiano como en el espacio de las variables articulares.

4.3.1. PLANEADO EN EL ESPACIO CARTESIANO

Se realiza especificando posiciones y orientaciones de la herramienta {T} con respecto a un marco de referencia {S} sin antes resolver la cinemática inversa, en otras palabras se intenta conocer el cambio de las variables cartesianas en el tiempo.

²⁰ CRAIG, Jhon J. Introduction to robotics: Mechanics and control. Addison-Wesley. 1989

Los métodos de generación de trayectorias en el espacio cartesiano son computacionalmente mas complejos, ya que es necesario resolver en tiempo real el modelo cinemático inverso y actualizarlo a la ruta.

Si combinamos la representación de orientación por tres números con la representación para la posición 3×1 , tenemos una representación 6×1 para posición y orientación.

Si se especifica la orientación mediante matrices en los puntos de paso, no es posible interpolar linealmente sus elementos ya que se generarían matrices no validas. Las matrices de rotación deben tener columnas ortonormales, condición que no se garantiza si se interpolan linealmente elementos de dos matrices²¹. Representando un punto de paso como:

$${}^S T_A = \begin{bmatrix} {}^S R_A & | & {}^S P_{AORG} \\ \hline 000 & | & 1 \end{bmatrix} \quad (4.1)$$

Transformación para la que ${}^S P_{AORG}$ da la posición del origen del sistema {A} con respecto a {S} y ${}^S R_A$ es una matriz de rotación que define la orientación del sistema {A} con respecto a {S}.

Usamos el símbolo \mathfrak{N} para representar el correspondiente vector 6×1

$${}^S \mathfrak{N}_A = \begin{bmatrix} {}^S P_{AORG} \\ {}^S K_A \end{bmatrix} \quad (4.2)$$

Donde ${}^S K_A$ se formó por escalar el vector unitario ${}^S \hat{K}_A$ por la magnitud de rotación, θ_{SA} . El problema del uso de estos métodos es la aparición de singularidades en el espacio cartesiano. El inconveniente es ahora utilizar funciones de interpolación suaves de movimientos entre puntos en el espacio de seis dimensiones definidas por ${}^S \mathfrak{N}_A$.

²¹ ANGELES, Jorge. Fundamentals of Robotic Mechanical Systems. Springer. 2003

Una leve complicación surge del hecho de que la representación de orientación ángulo-ejes no es única²²:

$$\left({}^S K_A, \theta_{SA}\right) = \left({}^S \hat{K}_A, \theta_{SA} + n360^\circ\right) \quad (4.3)$$

Donde n es cualquier número entero positivo o negativo. Comúnmente se utiliza la que haga que la rotación total sea menor. Si se quiere ir del punto {A} al punto {B}, suponiendo que la orientación en A viene dada por ${}^S K_A$, se busca ${}^S K_B$ tal que $\left|{}^S K_B - {}^S K_A\right|$ sea mínimo.

Problemas Geométricos Con Trayectorias Cartesianas. Consideremos algunos problemas geométricos que limitan la aplicación de los métodos en espacio cartesiano para la generación de trayectorias.

- Puntos intermedios inalcanzables
- Velocidades articulares altas cerca de las singularidades
- Puntos alcanzables solo con algunas orientaciones

Puntos intermedios inalcanzables: conociendo los puntos inicial A, final B, y requiriendo alcanzar B en línea recta desde A, para el caso representado en la Figura 4.1, en el que el segundo eslabón es más corto que el primero, por lo que el espacio de trabajo, demarcado en azul, tiene un agujero en el centro el cual contiene algunos “puntos intermedios inalcanzables” de la trayectoria por tanto la no se puede ejecutar. Nótese que el movimiento de A a B es posible en el espacio articular.

²² CRAIG, Jhon J. Introduction to robotics: Mechanics and control. Addison-Wesley. 1989

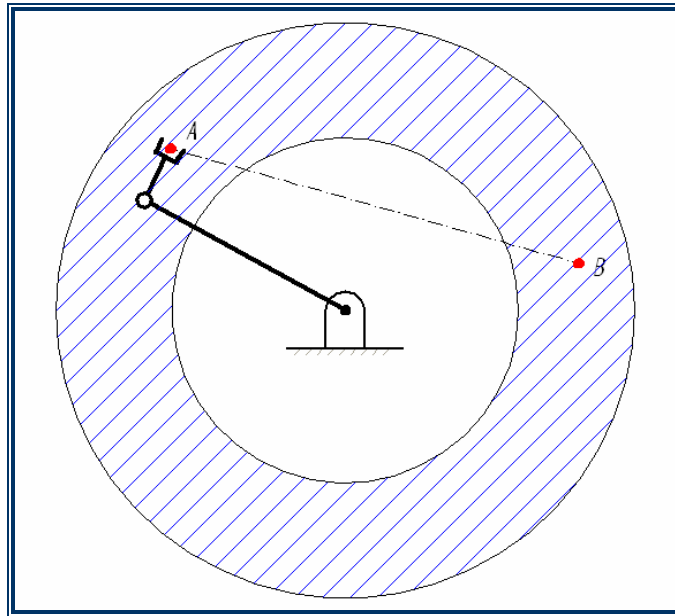


Figura 4.1. Trayectoria con puntos intermedios inalcanzables

Velocidades articulares altas cerca de las singularidades: si, un manipulador siguiendo una ruta de línea recta en el espacio cartesiano se aproxima a la configuración de singularidad del mecanismo, una o más velocidades articulares pueden aumentar hacia infinito. Debido a que estas velocidades del mecanismo son mayores al los limites propios del manipulador, lo mas seguro es que esto implique una desafortunada desviación de la ruta planeada.

Para ejemplificar este problema observe la Figura 4.2 donde se trata de mover un manipulador planar de dos eslabones (con longitudes iguales) del punto {A} al punto {B} con velocidad lineal constante describiendo una línea recta. Evidentemente todos los puntos son alcanzables, pero cuando el manipulador pasa por la mitad, la velocidad de la primera articulación es demasiado alta. Esta velocidad será más alta cuanto mas la trayectoria se aproxime al eje de la primera articulación. Una forma de solucionar esto sería

limitar la velocidad de la trayectoria al máximo de velocidad articular, innegablemente esto haría perder los atributos temporales deseados.

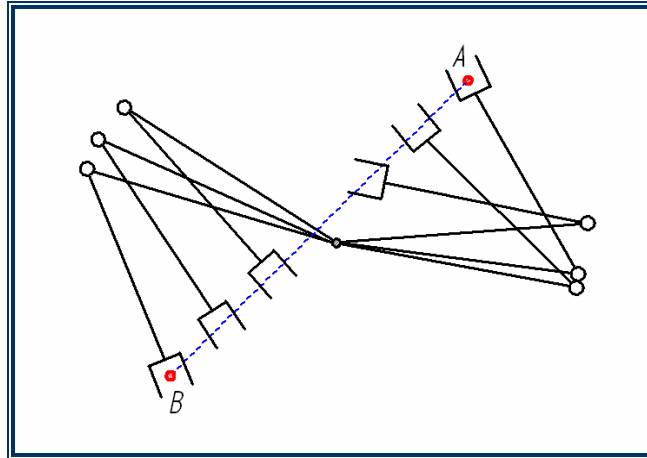


Figura 4.2. Velocidades altas cerca de las singularidades

Puntos alcanzables solo con algunas orientaciones: esto ocurre cuando la trayectoria requiere el paso por puntos alcanzables pero en los que el manipulador tiene restringidas algunas orientaciones, si se especifica una de estas orientaciones limitadas la trayectoria será desviada.

Por ejemplo, en la Figura 4.3 se observa un manipulador de dos eslabones de longitudes iguales, que se desplaza en el plano. Se traza una trayectoria rectilínea para ir del punto {A} al punto {B}. El manipulador puede alcanzar todos los puntos de la ruta planeada, pero no la puede seguir ya que hacia la mitad deberá cambiar su orientación y no llegará al punto {B} con la orientación deseada.

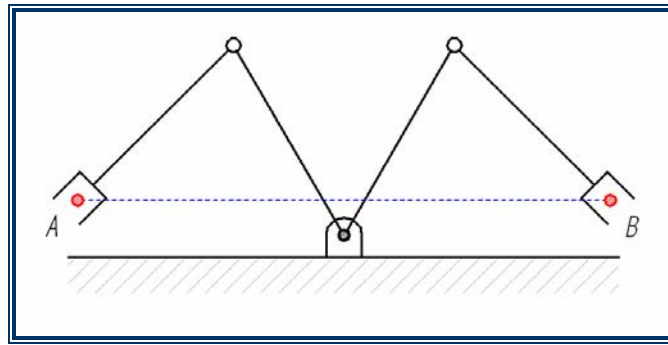


Figura 4.3. Puntos alcanzables con algunas orientaciones

Debido a los problemas indicados anteriormente, la mayor parte de manipuladores industriales emplea, en tiempo real, métodos de generación en el espacio articular. Consecuentemente para la generación de trayectorias en el RobotArm MR999 se usa un método de generación en el espacio articular.

4.3.2. PLANEADO EN EL ESPACIO ARTICULAR

En el planeado en el espacio articular, no existe un control sobre las posiciones u orientaciones cartesianas entre puntos de paso consecutivos, lo que se trata de lograr es una función suave de interpolación para cada articulación.

Aquí cada punto de la ruta esta comúnmente especificado en términos de la posición y orientación deseadas del efector final, $\{T\}$, relativa al marco de referencia fijo $\{S\}$. Cada uno de estos puntos de la trayectoria debe ser convertido a un grupo de parámetros articulares, aplicando la cinemática inversa. En aquel momento, se busca una función suave para cada una de las n articulaciones, que pase por los puntos vía y llegue al punto objetivo.

El tiempo requerido por cada segmento es el mismo para cada articulación, y cada articulación debe alcanzar los puntos al mismo tiempo, así será encontrado cada punto de la posición cartesiana de {T} requerida. Cabe anotar que la función angular de cada articulación no depende de las funciones de las otras articulaciones.

El planeado en el espacio articular es mas económico computacionalmente, asimismo como no se hace una correspondencia continua entre el espacio articular y el cartesiano no hay problemas de singularidad del mecanismo. Ahora bien, el problema se reduce a encontrar una función para cada articulación que tome el valor t_0 en la posición inicial de la articulación y el valor t_f en la posición deseada. Existen varios métodos para encontrar funciones suaves, $\theta(t)$, que pueden ser usadas para interpolar los valores articulares.

Algunos de estos métodos son²³:

- Empleo de polinomios cúbicos
- Empleo de polinomios cúbicos con puntos de paso
- Empleo de polinomios de orden superior
- Empleo de funciones lineales con enlace parabólico cobertura

En nuestro caso, utilizaremos polinomios de orden superior, específicamente de grado 5 (quínticos), por ser más versátiles al permitir definir posición, velocidad y aceleración.

²³ Para profundizar en métodos de interpolación ver [1], [2]. La cobertura completa esta mas allá del alcance de este trabajo

4.4. GENERACIÓN DE TRAYECTORIAS EMPLEANDO POLINOMIOS DE QUINTO GRADO

Para obtener movimientos suaves de los actuadores en las articulaciones es conveniente especificar posición, velocidad y aceleración al comienzo y al final de cada segmento; esto puede hacerse empleando polinomios de quinto grado (quínticos). Cada función articular del tiempo $\theta(t)$ esta dada por:

$$\theta(t) = a_0 + a_1t + a_2t^2 + a_3t^3 + a_4t^4 + a_5t^5 \quad (4.4)$$

Las condiciones de frontera son:

Para $t = 0$

$$\theta_0 = a_0, \quad (4.5)$$

Para $t = t_f$

$$\theta_f = a_0 + a_1t_f + a_2t_f^2 + a_3t_f^3 + a_4t_f^4 + a_5t_f^5, \quad (4.6)$$

Para $\left. \frac{d\theta}{dt} \right|_{t=t_0}$

$$\dot{\theta}_0 = a_1, \quad (4.7)$$

Para $\left. \frac{d\theta}{dt} \right|_{t=t_f}$

$$\dot{\theta}_f = a_1 + 2a_2t_f + 3a_3t_f^2 + 4a_4t_f^3 + 5a_5t_f^4, \quad (4.8)$$

Para $\left. \frac{d^2\theta}{dt^2} \right|_{t=t_0}$

$$\ddot{\theta}_0 = 2a_2, \quad (4.9)$$

Para $\left. \frac{d^2\theta}{dt^2} \right|_{t=t_f}$

$$\ddot{\theta}_f = 2a_2 + 6a_3t_f + 12a_4t_f^2 + 20a_5t_f^3. \quad (4.10)$$

Este conjunto de restricciones conforman un sistema de seis ecuaciones con seis incógnitas cuya solución es:

$$\begin{aligned}
 a_0 &= \theta_o, \\
 a_1 &= \dot{\theta}_o, \\
 a_2 &= \frac{\ddot{\theta}_o}{2}, \\
 a_3 &= \frac{20\theta_f - 20\theta_o - (8\dot{\theta}_f + 12\dot{\theta}_o)t_f - (3\ddot{\theta}_o - \ddot{\theta}_f)t_f^2}{2t_f^3}, \\
 a_4 &= \frac{30\theta_o - 30\theta_f + (14\dot{\theta}_f + 16\dot{\theta}_o)t_f - (3\ddot{\theta}_o - 2\ddot{\theta}_f)t_f^2}{2t_f^4}, \\
 a_5 &= \frac{12\theta_f - 12\theta_o - (6\dot{\theta}_f + 6\dot{\theta}_o)t_f - (\ddot{\theta}_o - \ddot{\theta}_f)t_f^2}{2t_f^5}.
 \end{aligned} \tag{4.11}$$

Estas ecuaciones junto con las condiciones de frontera ofrecen la interpolación de quinto grado para encontrar una función articular del tiempo $\theta(t)$ para cada articulación del RobotArm MR 999.

En MATLAB se emplea la función *pol5.m* y *pol5aux.m*, incluidas en el anexo, para encontrar las trayectorias articulares al resolver las ecuaciones (4.11), junto con la función *InversaRoboArm.m*, para encontrar los valores articulares de interpolación. A estas funciones deben introducirse las posiciones-orientaciones (inicial, intermedias y final), las velocidades y aceleraciones.

Es posible, además, introducir los valores de duración de intervalos entre puntos vía y el tamaño de paso para la trayectoria generada.

La sintaxis de la función *pol5.m* es:

[qcoef, time, q_t, qd_t, qdd_t]=pol5(q, qd, qdd, t0, tij, step)

Los parámetros de esta función son:

Entradas:

q: matriz con tantas filas como articulaciones, que contiene en cada columna las posiciones articulares deseadas para el punto inicial, los puntos de paso y el punto final.

qd: matriz con tantas filas como articulaciones, que contiene en cada columna las velocidades articulares deseadas para el punto inicial, los puntos de paso y el punto final.

qdd: matriz con tantas filas como articulaciones, que contiene en cada columna las aceleraciones articulares deseadas para el punto inicial, los puntos de paso y el punto final.

t0: parámetro que contiene el instante inicial.

tij: vector que contiene la duración de intervalos entre puntos.

step: paso con el que se calculan posición, velocidad y aceleración resultantes.

Salidas:

qcoef: matriz de tres dimensiones, que contiene en cada fila los coeficientes del polinomio quintico solución para cada una de las variables articulares. La primera columna se corresponde con la potencia quintica y la última con el término independiente. El tercer índice de la matriz hace referencia a la articulación a la que corresponden los coeficientes.

time: vector tiempo con los instantes para los que se calculan las matrices q_t , qd_t y qdd_t que contienen los valores de posición, velocidad y aceleración resultantes.

q_t: matriz que contiene en cada fila los valores de posición correspondientes a cada articulación.

qd_t: matriz que contiene en cada fila los valores de velocidad correspondientes a cada articulación.

qdd_t: matriz que contiene en cada fila los valores de aceleración correspondientes a cada articulación.

CONCLUSIONES

- Se obtuvieron los modelos dinámicos y de generación de trayectorias esperados y planteados en los objetivos del proyecto.
- Se construyó una serie de aplicaciones computacionales para facilitar el cálculo y recálculo de los modelos, basados en funciones MATLAB y en algunos “toolbox” encontrados en la literatura e incluidos como anexo
- Se extrajo, del manipulador físico y su manual, un modelo 3D detallado, modelado en Solid Edge V12 con sus respectivas piezas, planos y ensamble; y un modelo 3D simplificado, modelado en Roboworks cuyo fin principal es recibir información proveniente del software de cálculo (MATLAB/SimuLink) y generar la subsiguiente animación basada en estos datos.
- Los modelos desarrollados junto con los útiles computacionales construidos, proporcionan una potente herramienta de simulación del RobotArm MR 999
- Los desarrollos se presentan de forma ordenada y secuencial. Así, se podrán utilizar en posteriores estudios de profundización en este tema.

RECOMENDACIONES

- Promover el estudio de la mecatrónica como fuente de recursos renovadores para la ingeniería e impulsor de la creatividad. Como lugar de convergencia de las destrezas del ingeniero mecánico de nuestra época.
- Continuar con el estudio del manipulador RobotArm MR 999, extendiéndose hasta el control. De esta forma, se podrá claramente utilizar el material de este proyecto como soporte de las estrategias de control a implementar.
- Tratar de poner en funcionamiento un curso o un laboratorio de robótica en la Escuela de Ingeniería Mecánica de la Universidad Industrial de Santander, que podría, en un principio, tomar este material como una guía tangible.

BIBLIOGRAFÍA

- ✓ **ACEVEDO OLAYA, Carlos Gonzalo-CACERES MARTÍNEZ, Hernando.** Diseño y construcción de un brazo robot IPT-CNC modulo 1. Trabajo de grado. Escuela de ingeniería mecánica. Universidad Industrial de Santander. 1998.
- ✓ **ANGELES, Jorge.** Fundamentals of Robotic Mechanical Systems. Springer. 2003. [3]
- ✓ **CRAIG, Jhon J.** Introduction to robotics: Mechanics and control. Addison-Wesley. 1989. [1]
- ✓ **FLOREZ FRANCO, Marco Fidel-GONZÁLEZ, Fabio Alfonso.** Programa ejecutivo para el control de un robot industrial. Trabajo de grado. Escuela de ingenierías eléctrica y electrónica. Universidad Industrial de Santander. 1999
- ✓ **FU, King Sun.** Robótica: Control, detección, Visión e inteligencia. McGraw-Hill. 1988 [2]
- ✓ **GONZÁLEZ, Fabio Alfonso.** Algoritmos genéticos aplicados al planeamiento de trayectorias de robots móviles, modelamiento y simulación. Trabajo de grado. Maestría en potencia eléctrica. Universidad Industrial de Santander. 2004.

- ✓ **JOYA CASTELLANOS, Juan Carlos.** Modulo didáctico para el diseño de modelos geométricos de robots manipuladores. Trabajo de grado. Escuela de ingeniería de sistemas. Universidad Industrial de Santander. 1993.

- ✓ **OLLERO, Aníbal.** Robótica, manipuladores y robots móviles. Alfaomega. 2003. [7]

- ✓ **SIERRA BUENO, Daniel Alfonso.** Planeación de trayectorias para un robot en una celda de manufactura. Trabajo de grado. Escuela de ingenierías eléctrica y electrónica. Universidad Industrial de Santander. 2001.

ANEXOS

ANEXO A. COMO UTILIZAR LA INTERFASE PARA SIMULAR EL ROBOT ARM MR999

A.1 PRELIMINAR

Para poder utilizar la interfase de simulación deben seguir las instrucciones del archivo *LEER Interfase.txt*, incluido dentro del CD anexo.

A.2 INICIAR INTERFASE

Para abrir la interfase debe ejecutar en la ventana de comandos de MATLAB7 la sentencia: `>>principal`

Dicha sentencia debe mostrar la ventana del menú principal, como se observa en la Figura A.1.



Figura A.1 Ventana del menú principal.

A.3. SIMULACIÓN DE LA CINEMÁTICA.

Estando en el menú principal (Figura A.1) elija la opción “*CINEMÁTICA*” con lo que aparecerá el submenú “*CINEMÁTICA DEL ROBOT ARM MR999*” donde enfrente de cada opción (botón) hay una breve explicación de la función que se encontrará al presionar dicho botón, como se puede ver en la Figura A.2.

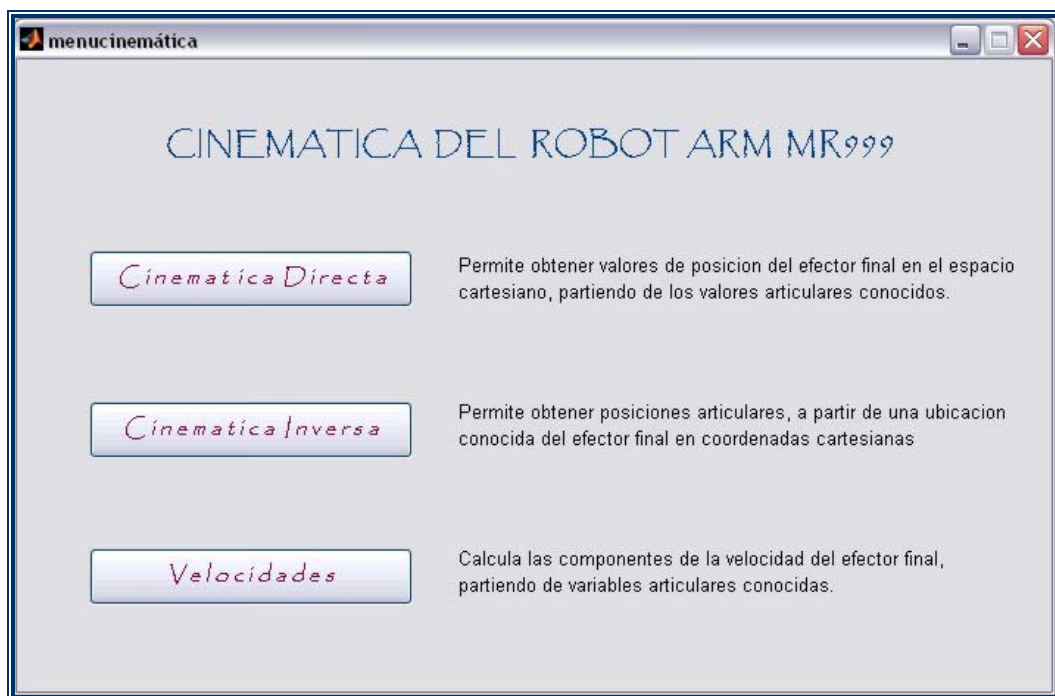


Figura A.2. Submenú para simular la CINEMÁTICA

A.3.1. SIMULACIÓN DE LA CINEMÁTICA DIRECTA

Al pulsar el botón “*Cinemática Directa*” deberá verse el formato para entrada y salida de datos mostrado en la Figura A.3. donde en las casillas bajo el rotulo “*VARIABLES DE ENTRADA*” se introducen los valores de cada una de las posiciones articulares en grados, para que en las casillas bajo el rotulo

“*VARIABLES DE SALIDA*” se visualicen los valores de posición y orientación del efector final en coordenadas cartesianas respecto a la base del manipulador.



Figura A.3. Simulación de la cinemática directa.

El botón “*Animación*” tiene como fin la visualización de la simulación del movimiento del RobotArm MR999, bajo el entorno graficador de MATLAB.

Una imagen de esta animación puede observarse en la Figura A.4. donde se aprecia un esquema del manipulador con los sistemas de coordenadas atados a su respectiva articulación según la notación utilizada en el desarrollo.

El botón “*Mostrar Imagen*” sirve para observar una imagen del manipulador en la posición donde todas sus articulaciones tienen un valor de rotación de

0° considerada como la posición inicial para efectos de la animación(ver Figura A.5), y se representa la orientación de los marcos de referencia utilizados. El sentido de rotación de los ángulos se rige por la regla de la mano derecha, es decir una rotación del eje X_i al eje Y_i se considera positiva ($+\theta_i$).

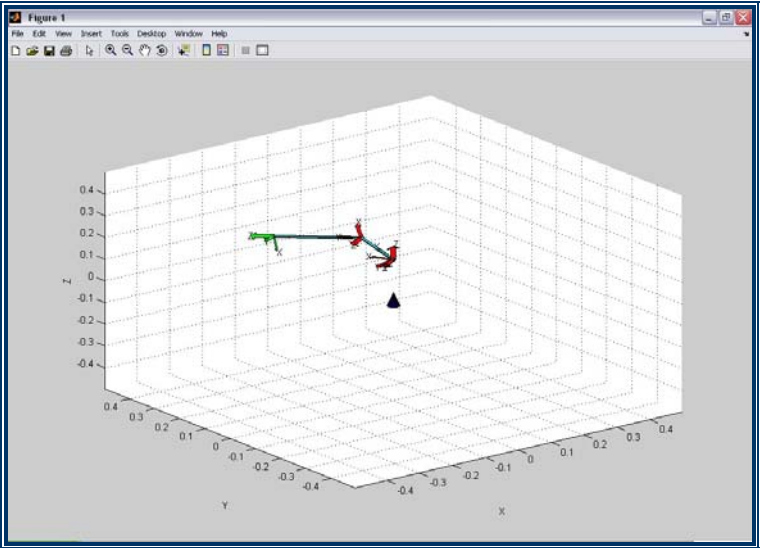


Figura A.4. Entorno de simulación en MATLAB

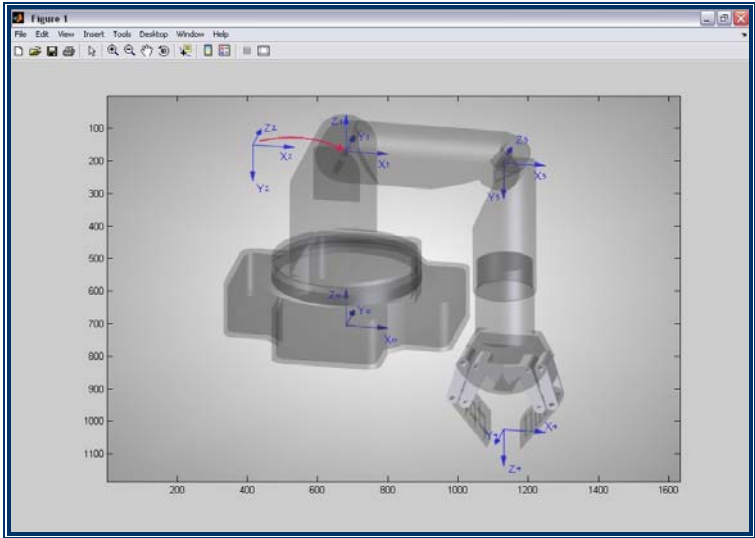


Figura A.5. Posición inicial con sistemas de referencia

A.3.2. SIMULACIÓN DE LA CINEMÁTICA INVERSA

Al pulsar el botón “*Cinemática Inversa*” deberá verse el formato para entrada y salida de datos mostrado en la Figura A.6. donde en las casillas bajo el rotulo “*VARIABLES DE ENTRADA*” se introducen los valores de posición y orientación del efector final en coordenadas cartesianas respecto a la base del manipulador para que en las casillas bajo el rotulo “*VARIABLES DE SALIDA*” se visualicen los valores de cada una de las posiciones articulares en grados.

VARIABLES DE ENTRADA		VARIABLES DE SALIDA	
Posicion X (cm)	-26.42	TETA 1 (°)	160.012
Posicion Y (cm)	9.61	TETA 2 (°)	-60.022
Posicion Z (cm)	31.17	TETA 3 (°)	-44.9671
Gama (°)	74.08	TETA 4 (°)	249.998
Beta (°)	19.29		
Alfa (°)	-115.38		

Figura A.6. Simulación de la cinemática inversa.

La funcionalidad de este formato se rige por el mismo esquema explicado en el apartado sobre cinemática directa, es decir, los botones ejecutan el mismo proceso, pero para el caso de la cinemática inversa.

A.3.3. CALCULO DE LA VELOCIDAD DEL EFECTOR FINAL

Al pulsar el botón “*Velocidades*” deberá verse el formato para entrada y salida de datos mostrado en la Figura A.7. donde en las casillas bajo el rotulo “*VARIABLES DE ENTRADA*” se introducen los valores de posición y velocidad angular de cada articulación para que en las casillas bajo el rotulo “*VARIABLES DE SALIDA*” se visualicen los valores de cada componente de velocidad tanto lineal como rotacional del efector final respecto a la base.

The screenshot shows a software window titled "VELOCIDAD DEL EFECTOR FINAL" with a sub-window titled "jacobiano". The window is divided into two main sections: "VARIABLES DE ENTRADA" and "VARIABLES DE SALIDA".

VARIABLES DE ENTRADA:

Variable	Value
TETA 1 (°)	160
W1 (rad/s)	1.2
TETA 2 (°)	-60
W2 (rad/s)	0.8
TETA 3 (°)	-45
W3 (rad/s)	0.75
TETA 4 (°)	250
W4 (rad/s)	1.7

VARIABLES DE SALIDA:

Variable	Value
Vx (m/s)	-0.278923
Wx (rad/s)	-2.07318
Vy (m/s)	-0.257529
Wy (rad/s)	-0.894901
Vz (m/s)	-0.391553
Wz (rad/s)	1.63999

An "OK" button is visible between the input and output sections.

Figura A.7. Calculo de velocidad del efector final

A.4. SIMULACIÓN DE LA CINÉTICA

Estando en el menú principal (Figura A.1) elija la opción “*CINÉTICA*” con lo que aparecerá el submenú “*CINÉTICA DEL ROBOT ARM MR999*” donde enfrente de cada opción (botón) hay una breve explicación de la función que se encontrara al presionar dicho botón, como se puede ver en la Figura A.8.

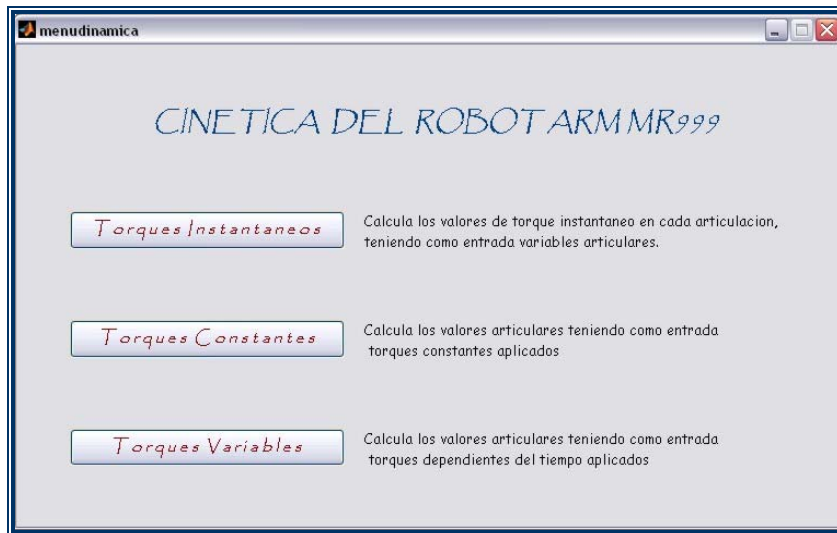


Figura A.8. Menú para simular la cinética

A.4.1. SIMULACIÓN DE LA CINÉTICA INVERSA

Al pulsar el botón “*Torques Instantáneos*” deberá verse el formato para entrada y salida de datos mostrado en la Figura A.9. donde en las casillas bajo el rotulo “*VARIABLES DE ENTRADA*” se introducen los valores de posición, velocidad y aceleración angular de cada articulación para que en las casillas bajo el rotulo “*VARIABLES DE SALIDA*” se visualicen los valores de cada torque que se esta ejerciendo sobre cada articulación en el instante indicado por las variables de entrada.

A.4.2. SIMULACIÓN DE LA CINÉTICA DIRECTA

A.4.2.1. APLICANDO TORQUES CONSTANTES

Al pulsar el botón “*Torques Constantes*” deberá verse el diagrama de bloques en SimuLink mostrado en la Figura A.10. para introducir los valores de torque constante en cada una de las articulaciones, debe hacer doble click en

el bloque "Torques de entrada" y digitar los valores deseados. Para iniciar la simulación basta con seleccionar "start" del menú "simulation".

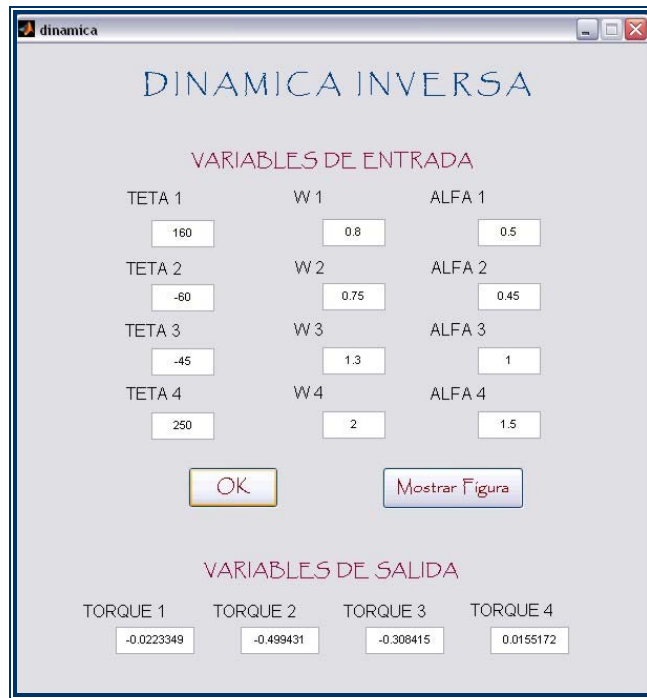


Figura A.9. Cálculo de torques instantáneos.

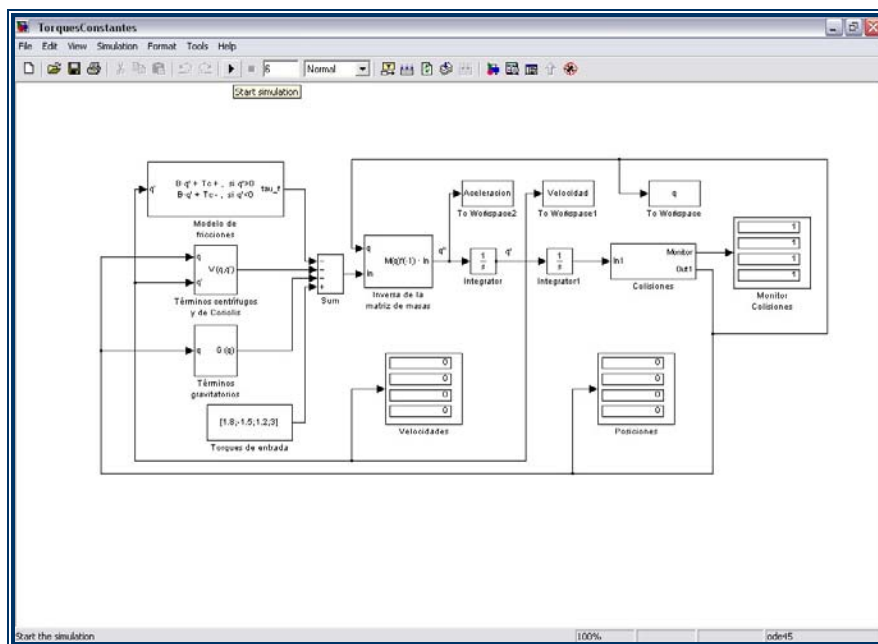


Figura A.10. Diagrama de bloques en Simulink con torques constantes

A.4.2.2. APLICANDO TORQUES VARIABLES

Al pulsar el botón “*Torques Variables*” deberá verse el diagrama de bloques en SimuLink mostrado en la Figura A.11. para introducir las funciones de torque variable en cada una de las articulaciones, debe hacer doble click en cada bloque “*Torque1, 2, 3, 4*” y digitar la función del tiempo deseada. Para iniciar la simulación basta con seleccionar “*start*” del menú “*simulation*”.

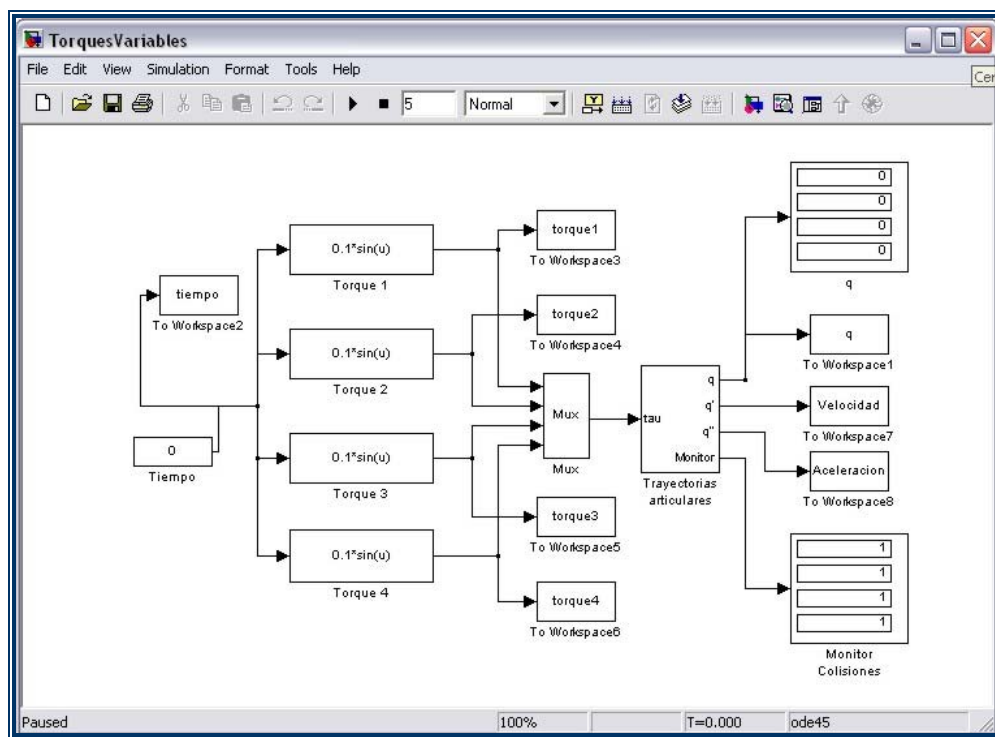


Figura A.11. Diagrama de bloques en SimuLink con entrada de torques variables

A.5. SIMULACIÓN DE TRAYECTORIAS

Estando en el menú principal (Figura A.1) elija la opción “*GENERACION DE TRAYECTORIAS*” con lo que aparecerá el formato para entrada de datos mostrado en la Figura A.12.



Figura A.12. Formato de entrada de valores para la generación de trayectorias

Al presionarse cualquiera de las opciones se obtendrá en forma gráfica el resultado respectivo. En la figura A.13 se observa el resultado para el caso de las posiciones articulares de acuerdo a la generación de trayectorias, donde las curvas azul, verde, rojo y cyan corresponden a la base, hombro, codo y muñeca respectivamente.

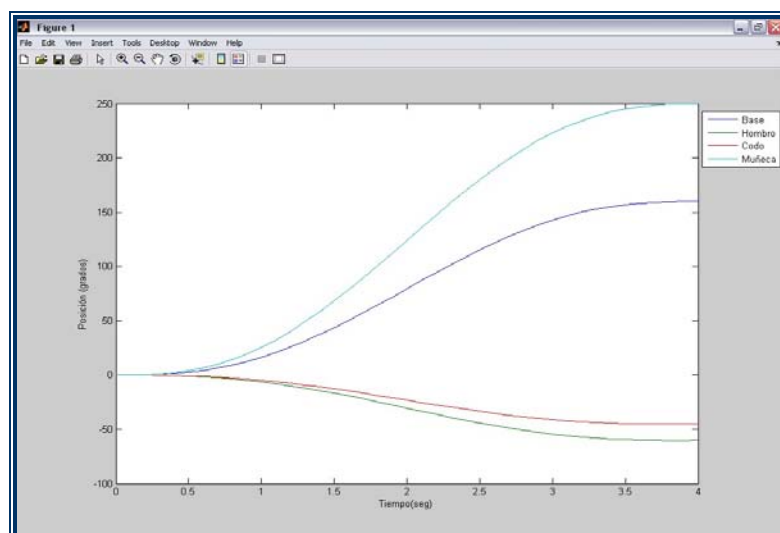


Figura A.13. Resultado grafico para posiciones articulares según generación de trayectoria

ANEXO B. ANIMACIÓN EN ROBOWORKS

B.1 PRELIMINAR

Para poder visualizar las simulaciones de MATLAB con una mejor calidad grafica se dispone del uso de Roboworks, para ello debe seguir las instrucciones del archivo *LEER RoboWorks.txt*, incluido dentro del CD anexo.

B.2 INICIAR ANIMACION

Para abrir el modelo del RobotArm MR999, en el menú archivo seleccione abrir y busque el archivo con nombre *Roboarm999.scn* y ábralo.

Para ver una animación previamente generada por MATLAB seleccione “*From File*” del menú “*Animation*” y enrute la búsqueda al directorio ...MATLAB7\work para seleccionar el archivo *.dat que se desee animar.

Los archivos *.dat se generan al realizar la animación correspondiente en la interfase de MATLAB, es decir si se quiere ver la animación para la cinemática directa, debe abrirse el archivo *Directa.dat*.

Una imagen del entorno RoboWorks se observa en la Figura B.1 donde se animo el archivo *Trayectoria.dat* creado por MATLAB al simular la generación de trayectorias.

Cabe advertir que en este entorno existe la posibilidad de realizar movimientos bajo la voluntad del usuario directamente, para esto debe presionar las teclas de la forma indicada en la Tabla B.1.

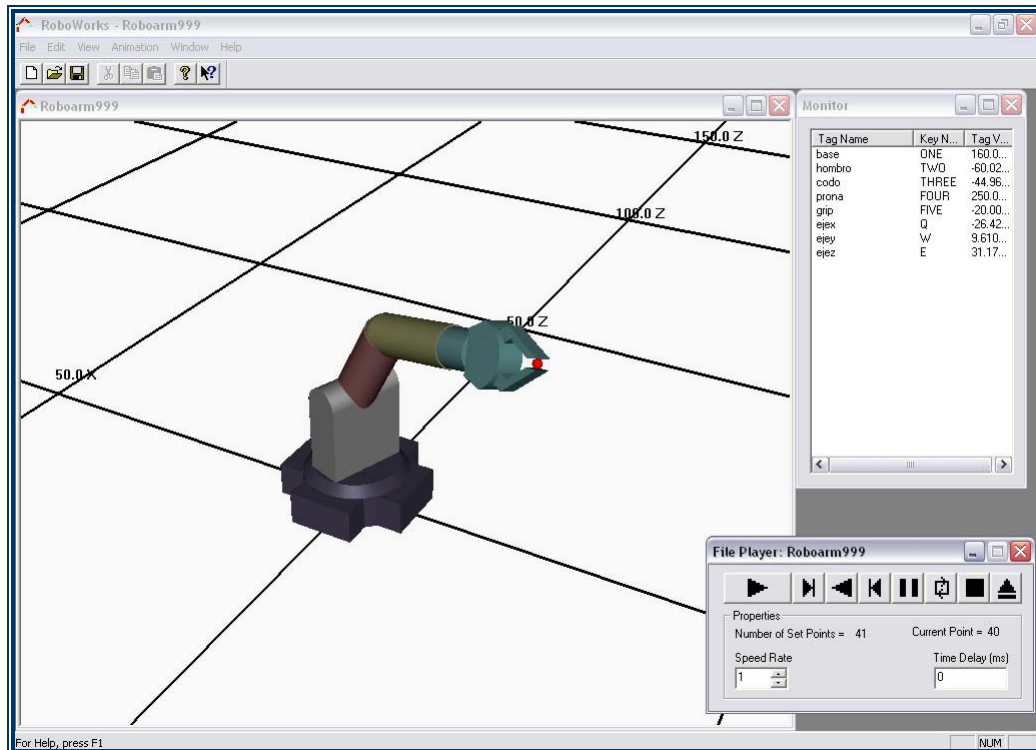


Figura B.1. Imagen entorno RoboWorks

Elemento a mover	Teclas de control	
	Sentido positivo	Sentido negativo
Base	1	Shift+1
Hombro	2	Shift+2
Codo	3	Shift+3
Muñeca	4	Shift+4
Gripper	5	Shift+5
Objetivo en X	Q	Shift+Q
Objetivo en Y	W	Shift+W
Objetivo en Z	E	Shift+E

Tabla B.1. Animación Manual