

**UNIVERSIDAD NACIONAL DE INGENIERIA
FACULTAD DE INGENIERIA MECANICA**



**“DISEÑO DE UN CONTROLADOR NO LINEAL
MULTIVARIABLE POR REALIMENTACION DE
ESTADO APLICADO A UN ROBOT MANIPULADOR
ESFERICO DE TRES GRADOS DE LIBERTAD.”**

TESIS

**PARA OPTAR EL TITULO PROFESIONAL DE
INGENIERO MECANICO ELECTRICISTA**

RONALD DIVIO MARAVI PINTO

PROMOCION 1981-I

LIMA-PERU

2005

Dedicatoria

A mis padres Viviano y Felicita

A mi hijo Ronald Juan y mi esposa Gladys

A mis hermanos Doris, Zelmira, Elmer y Rolando

Agradecimiento

Quisiera expresar mi agradecimiento a Dios por darme fé y fuerza de voluntad por concluir la presente tesis; así mismo agradezco al Ing. Freedy Sotelo, a mi amigo y compañero de promoción el Ing. Héctor Paz, por su apoyo y amistad para que mi tesis llegara a buen término.

Tabla de Contenido

Lista de Figuras	1
Lista de Tablas	3
Simbología	4
Prólogo . .	9
Capítulo 1	
Introducción	11
1.1 Objetivos	12
1.2 Alcance	12
Capítulo 2	
Planteamiento del Problema	14
2.1 Enunciado del Problema	14
2.2 Lista de las Variables	14
Capítulo 3	
Modelado de la Planta	17
3.1 Introducción	17
3.2 Descripción del Manipulador .	17
3.3 El Modelo Dinámico	19
3.3.1 El Procedimiento Denavit-Hartenberg .	21
3.3.2 El Algoritmo de Lagrange-Euler [1], [6]	24
3.4 Aplicación y Programación del Modelo	26
3.4.1 Cálculos	26
3.4.2 Programación del Algoritmo Lagrange-Euler	31
3.4.3 El Modelo de los Actuadores	33
3.4.4 Modelo en el Espacio Estado	37
3.4.5 Comportamiento del Manipulador [2],[3]	41
Capítulo 4	
Diseño Controlador No Lineal	45
4.1 Herramientas de Geometría Diferencial	45
4.1.1 Derivada de Lie y Corchete de Lie [13], [14]	46
4.1.2 Difeomorfismo y Transformación de Coordenadas [13], [14]	47
4.1.3 El Teorema de Frobenius [13], [14]	48

4.2	Linealización por Realimentación para Procesos SISO [7], [12], [13], [14]	49
4.2.1	Condiciones para Linealización	49
4.2.2	Grado Relativo del Proceso . . .	50
4.2.3	Forma Normal SISO para Linealización Exacta	51
4.2.4	Ley de Control SISO para Linealización Exacta	52
4.3	Linealización por Realimentación para Procesos MIMO [13], [14]	53
4.3.1	Modelando Procesos Cuadrados No Lineales	53
4.3.2	Grado Relativo Total	54
4.3.3	Forma Normal para Linealización Exacta MIMO .	54
4.3.4	Ley de Control de Desacoplamiento MIMO	56
Capítulo 5		
	Simulación con Matlab	58
5.1	Caso 1: Control de Trayectoria Constante	59
5.2	Caso 2: Control de Trayectoria Senoidal	66
Capítulo 6		
	Especificaciones Técnicas	70
6.1	Introducción	70
6.2	Hardware del Sistema .	70
6.2.1	La Planta .	70
6.2.2	El Actuador . .	71
6.2.3	El Sensado . . .	75
6.2.4	La Interfaz de Adquisición de Datos .	78
Conclusiones		80
Bibliografía		83
Apéndice A		
	Coeficiente de Fricción Viscosa	85
Apéndice B		
	Programas en MATLAB	90
B.1	Listado de Programas en MATLAB	90
B.2	Programa del Cálculo de la Ecuación Dinámica del Manipulador	90
B.3	Programa de Simulación con Trayectoria Constante	97
B.4	Programa de Simulación con Trayectoria Senoidal	105
Apéndice C		
	Ejemplo: Manipulador de 2DOF	113
C.1	Primer experimento: Control de Trayectoria Constante	113
C.2	Segundo Experimento: Control de Trayectoria Arbitraria	116

Lista de Figuras

3.1	Esquema del Robot Manipulador Esférico de 3DOF .	18
3.2	Esquema con las medidas del Manipulador	19
3.3	Sistemas de referencias del Manipulador .	20
3.4	Momento de inercia del disco respecto al sistema de referencia 1 (S_1).	28
3.5	Momento de inercia de la barra respecto al sistema de referencia 1 (S_1)	29
3.6	Servomotor DC1 de la base del Manipulador	33
3.7	Servomotor DC2 de la primera varilla del Manipulador	35
3.8	Servomotor DC3 de la segunda varilla del Manipulador	37
3.9	Robot de 3DOF para la animación . .	42
3.10	Menú de funciones del Robot de 3DOF	42
5.1	Diagrama de bloques del controlador no lineal	58
5.2	Trayectoria constante: salidas controladas para la base, primera y segunda varilla	64
5.3	Trayectoria constante: señales de control para los servomotores de la base, primera y segunda varilla	65
5.4	Trayectoria constante: velocidad para los servomotores de la base, primera y segunda varilla	65
5.5	Trayectoria senoidal: salidas controladas para la base, primera y segunda varilla	67
5.6	Trayectoria senoidal: señales de control para los servomotores de la base, primera y segunda varilla	68
5.7	Trayectoria senoidal: velocidad de los servomotores de la base, primera y segunda varilla	69
6.1	Diseño del Robot Manipulador Esférico de 3DOF	72
6.2	Servomotor DC	73
6.3	Circuito Amplificador de Potencia .	75

6.4	Esquema del circuito de disparo y conmutación	75
6.5	Circuito Generador PWM	76
6.6	Codificador en el eje del servomotor.	77
6.7	Estructura del codificador óptico rotatorio.	78
6.8	Detector de cuadratura.	79
6.9	Interfaz de Adquisición de Datos e Implementación del Sistema.	80
A.1	Modelo del servomotor DC.	86
A.2	Simulación para obtener la Fricción de Coulomb.	87
A.3	Respuesta tensión de entrada versus velocidad. . .	88
A.4	Experimento de la compensación de la fricción no lineal.	89
C.1	Implementación real del sistema.	114
C.2	Implementación real de la planta.	115
C.3	Trayectoria Constante Experimental: salidas controladas para la base y el brazo.	116
C.4	Trayectoria Constante Experimental: señal de control para el servomotor de la base y el brazo.	116
C.5	Trayectoria Constante Experimental: velocidad del servomotor de la base y el brazo.	117
C.6	Trayectoria Senoidal Experimental: salidas controladas para la base y el brazo.	118
C.7	Trayectoria Senoidal Experimental: señal de control para el servomotor de la base y el brazo.	118
C.8	Trayectoria Senoidal Experimental: velocidad del servomotor de la base y el brazo.	119

Lista de Tablas

2.1	valores de las variables	15
3.1	valores de los parámetros $D - H$	26
6.1	Conmutación de los MOSFET's . .	74
A.1	Parámetros del Sistema a simular	87

Simbología

- $(x_{s_0}, y_{s_0}, z_{s_0})$: sistema de coordenadas base (eje motor DC1)
 $(x_{s_1}, y_{s_1}, z_{s_1})$: sistema de coordenadas 1DOF (eje motor DC2)
 $(x_{s_2}, y_{s_2}, z_{s_2})$: sistema de coordenadas 2DOF (eje motor DC3)
 $(x_{s_3}, y_{s_3}, z_{s_3})$: sistema de coordenadas 3DOF
- S_0 : sistema de referencia base (eje motor DC1)
 S_1 : sistema de referencia 1DOF (eje motor DC2)
 S_2 : sistema de referencia 2DOF (eje motor DC3)
 S_3 : sistema de referencia 3DOF
- m_d : masa del disco
 m_p : masa de la barra
 m_2 : masa de la primera varilla
 m_3 : masa de la segunda varilla
- h_d : altura del disco
 R_d : radio del disco
 h_p : altura de la barra
 a_p : ancho y largo de la sección de la barra
- L_2 : longitud de la primera varilla
 L_3 : longitud de la segunda varilla
 $L_{x_{s_2}}$: longitud del eje de referencia S_1 al CG 1ra. varilla
 $L_{x_{s_3}}$: longitud del eje de referencia S_2 al CG 2da. varilla
 $L_{c_{s_2}}$: longitud del eje de referencia S_2 al CG 1ra. varilla
 $L_{c_{s_3}}$: longitud del eje de referencia S_3 al CG 2da. varilla
 $L_{y_{s_1}}$: distancia del CG de la barra al eje del servomotor DC2
- I_1 : momento de inercia del sólido (disco más barra)
 I_2 : momento de inercia de la 1ra. varilla
 I_3 : momento de inercia de la 2da. varilla
 J_{m1} : momento de inercia del servomotor DC1 de la base

J_{m2} : momento de inercia del servomotor DC2 de la 1ra. varilla
 J_{m3} : momento de inercia del servomotor DC3 de la 2da. varilla
 J_{eq1} : momento de inercia equivalente del servomotor DC1 de la base
 J_{eq2} : momento de inercia equivalente del servomotor DC2 de la 1ra. varilla
 J_{eq3} : momento de inercia equivalente del servomotor DC3 de la 2da. varilla
 B_{eq1} : coeficiente de fricción viscosa equivalente servomotor DC1 de la base
 B_{eq2} : coeficiente de fricción viscosa equivalente servomotor DC2 de la 1ra. varilla
 B_{eq3} : coeficiente de fricción viscosa equivalente servomotor DC3 de la 2da. varilla
 n_1 : reducción servomotor DC1 de la base
 n_2 : reducción servomotor DC2 de la 1ra. varilla
 n_3 : reducción servomotor DC3 de la 2da. varilla
 R_{a1} : resistencia armadura servomotor DC1
 R_{a2} : resistencia armadura servomotor DC2
 R_{a3} : resistencia armadura servomotor DC3
 L_{a1} : inductancia armadura servomotor DC1
 L_{a2} : inductancia armadura servomotor DC2
 L_{a3} : inductancia armadura servomotor DC3
 V_{a1} : voltaje armadura servomotor DC1
 V_{a2} : voltaje armadura servomotor DC2
 V_{a3} : voltaje armadura servomotor DC3
 K_{A1} : constante ganancia driver 1
 K_{A2} : constante ganancia driver 2
 K_{A3} : constante ganancia driver 3
 K_{m1} : constante torque servomotor DC1
 K_{m2} : constante torque servomotor DC2
 K_{m3} : constante torque servomotor DC3
 K_{b1} : constante fuerza electromotriz servomotor DC1
 K_{b2} : constante fuerza electromotriz servomotor DC2
 K_{b3} : constante fuerza electromotriz servomotor DC3
 u_1 : señal de control de entrada del 1DOF
 u_2 : señal de control de entrada del 2DOF
 u_3 : señal de control de entrada del 3DOF
 g : aceleración de la gravedad
 r_{m2} : distancia del extremo superior de la barra al servomotor DC2

C_{c1} : coeficiente de fricción de Coulomb del 1DOF y 2DOF
 C_{c3} : coeficiente de fricción de Coulomb del 3DOF
 T_1 : torque del manipulador del 1DOF
 T_2 : torque del manipulador del 2DOF
 T_3 : torque del manipulador del 3DOF
 T_{c1} : torque de fricción de Coulomb del servomotor DC1
 T_{c2} : torque de fricción de Coulomb del servomotor DC2
 T_{c3} : torque de fricción de Coulomb del servomotor DC3
 q_1 : posición angular del servomotor DC1
 q_2 : posición angular del servomotor DC2
 q_3 : posición angular del servomotor DC3
 w_1 : velocidad angular en el eje de salida del servomotor DC1
 w_2 : velocidad angular en el eje de salida del servomotor DC2
 w_3 : velocidad angular en el eje de salida del servomotor DC3
 w_{m1} : velocidad angular antes del reductor del servomotor DC1
 w_{m2} : velocidad angular antes del reductor del servomotor DC2
 w_{m3} : velocidad angular antes del reductor del servomotor DC3
 H_{11} : componente de matriz de inercia del manipulador del 1DOF
 H_{22} : componente de matriz de inercia del manipulador del 2DOF
 H_{23} : componente de matriz de inercia del manipulador del 2DOF
 H_{32} : componente de matriz de inercia del manipulador del 3DOF
 H_{33} : componente de matriz de inercia del manipulador del 3DOF
 C_1 : componente del vector fuerza centrífuga y Coriolis manipulador del 1DOF
 C_2 : componente del vector fuerza centrífuga y Coriolis manipulador del 2DOF
 C_3 : componente del vector fuerza centrífuga y Coriolis manipulador del 3DOF
 d_1 : componente del vector fuerza gravitacional del manipulador del 1DOF
 d_2 : componente del vector fuerza gravitacional del manipulador del 2DOF
 d_3 : componente del vector fuerza gravitacional del manipulador del 3DOF
 x_1 : variable de estado de la posición angular del servomotor DC1
 x_2 : variable de estado de la posición angular del servomotor DC2
 x_3 : variable de estado de la posición angular del servomotor DC3
 x_4 : variable de estado de la velocidad angular del servomotor DC1
 x_5 : variable de estado de la velocidad angular del servomotor DC2
 x_6 : variable de estado de la velocidad angular del servomotor DC3

r : grado relativo del proceso
 n : orden del proceso
 v_1 : nueva señal de entrada lineal del 1DOF
 v_2 : nueva señal de entrada lineal del 2DOF
 v_3 : nueva señal de entrada lineal del 3DOF
 K_1 : parámetro de la matriz de ganancia del 1DOF
 K_2 : parámetro de la matriz de ganancia del 1DOF
 K_3 : parámetro de la matriz de ganancia del 2DOF
 K_4 : parámetro de la matriz de ganancia del 2DOF
 K_5 : parámetro de la matriz de ganancia del 3DOF
 K_6 : parámetro de la matriz de ganancia del 3DOF
 p_1 : trayectoria deseada del 1DOF
 p_2 : trayectoria deseada del 2DOF
 p_3 : trayectoria deseada del 3DOF
 y_1 : salida del 1DOF
 y_2 : salida del 2DOF
 y_3 : salida del 3DOF
 \bar{y}_1 : señal de error de seguimiento del 1DOF
 \bar{y}_2 : señal de error de seguimiento del 2DOF
 \bar{y}_3 : señal de error de seguimiento del 3DOF
 T : tiempo de muestreo
 V_{cFE} : voltage de compensación de fricción estática
 V_{cFC} : voltage de compensación de fricción de Coulomb
 w_{th} : velocidad angular umbral
 V_{zm} : voltage en la zona muerta
 k : variable en tiempo discreto
 z : nueva variable de estado lineal
 N_m : número de muestras
 a_1 : parámetro de inercia del 2DOF
 a_2 : parámetro de inercia del 3DOF
 a_3 : parámetro de inercia del 2DOF y 3DOF
 a_4 : parámetro de inercia del 2DOF y 3DOF
 a_5 : parámetro de inercia y gravedad del 2DOF y 3DOF
 a_6 : parámetro de inercia y gravedad del 3DOF

d_{DH_i} : parámetro DENAVIT-HARTENBERG

a_{DH_i} : parámetro DENAVIT-HARTENBERG

$f(\mathbf{x})$: función vectorial

$g(\mathbf{x})$: función vectorial

$h(\mathbf{x})$: función escalar

∇ : gradiente

$L_f h$: derivada de Lie de h con respecto a f

$ad_f g$: corchete de Lie de 2 campos vectoriales

$\bar{x}, \bar{y}, \bar{z}$: coordenadas del centro de gravedad CG del cuerpo del sistema de coordenada s_i -ésimo

Prólogo

El presente trabajo trata sobre el diseño de un controlador no lineal multi-variable por realimentación de estado aplicado a un robot manipulador esférico de 3 grados de libertad (3 DOF: 3 Degree-of-freedom). Este manipulador se compone de una base giratoria y 2 eslabones, cuyas posiciones angulares son sujetos de control y las salidas controladas deben seguir la evolución de trayectorias arbitrarias.

Para este estudio, se ha organizado el trabajo en seis capítulos de la siguiente forma:

En el capítulo **I** denominado Introducción se presentan los objetivos y el alcance del trabajo.

En el capítulo **II** denominado Planteamiento del Problema se presenta el enunciado del problema.

En el capítulo **III** denominado Modelado de la Planta se desarrolla el modelado matemático usando el algoritmo Lagrange-Euler.

En el capítulo **IV** denominado Realimentación No Lineal se presenta el fundamento teórico de la técnica de control no lineal para procesos SISO (simple entrada y simple salida) y para procesos MIMO (múltiple entrada y múltiple salida).

En el capítulo **V** denominado Simulación del Sistema se ilustran los resultados obtenidos de las simulaciones hechas en MATLAB. Las simulaciones han sido realizadas para el control de trayectoria constante y para control de trayectoria senoidal. Así mismo se presenta el comportamiento del manipulador utilizando el Toolbox Robotics para MATLAB.

En el capítulo **VI** denominado Especificaciones Técnicas se define el hardware y software para una futura implementación en tiempo real.

Finalmente se presenta las conclusiones, bibliografía y los apéndices.

En el Apéndice **A** se presenta el cálculo del coeficiente de fricción viscosa del

segundo eslabón.

En el Apéndice **B** se presenta los programas del algoritmo de Lagrange-Euler y el algoritmo de control no lineal, escritos en código MATLAB.

En el Apéndice **C** se presenta los resultados experimentales de la implementación en tiempo real de un robot manipulador de 2 grados de libertad.

Capítulo 1

Introducción

El control no lineal es una ciencia que trata del diseño de sistemas de control no lineales. Los sistemas físicos del mundo real son no lineales y pueden ser descritos por ecuaciones diferenciales.

En procesos multivariables de varias entradas y varias salidas, se caracterizan por presentar un comportamiento dinámico difícil que se usa y se implementa en sistemas de control no lineal, los cuales en los últimos años ha alcanzado un elevado nivel de aceptación industrial en control de procesos.

Las técnicas de control no lineal que han logrado mayor aceptación son las siguientes: control no lineal por el método de linealización por realimentación de estado, control no lineal adaptivo, control no lineal por modos deslizantes, control predictivo no lineal, control no lineal usando backstepping y control con inteligencia artificial (control difuso, neurocontrol, etc.).

La técnica a emplear en la presente tesis es la de Control No Lineal basada en el método de Linealización por Realimentación de Estado y se basa en la linealización del proceso mediante un cambio adecuado de coordenadas y una realimentación del vector de estado de dicho proceso, de modo tal que el proceso resultante en su forma normal nos permita aplicar técnicas de control lineal en el caso de linealización exacta o bien atacar los problemas de control con un grado de dificultad menor en el caso de linealización parcial.

El método de linealización exacta se aplica cuando el proceso no lineal MIMO (multiple-input-multiple-output) de orden n , de m entradas y m salidas es completamente linealizabile y desacoplado si su grado relativo total (r) es igual al orden del proceso, o sea $r=n$, y para determinar dicho grado se aplica los conceptos de geometría diferencial. Para el diseño del controlador no lineal multivariable tienen nuevas leyes de control de entrada v_i , los cuales son diseñadas por ubicación de polos del correspondiente subsistema lineal equivalente.

Control por realimentación de estado es bien entendible para sistemas lineales con simples entradas. Sin embargo para sistemas no lineales de múltiples entradas

son tópicos de búsqueda muy lenta y debido a esto tiene que ser investigado más profundamente por el reciente desarrollo de máquinas con dinámicas no lineales de aplicación industrial como manipuladores, aeroplanos, submarinos, y vehículos espaciales.

La aplicación para la técnica propuesta y explicada en la tesis es un manipulador tres grados de libertad (3 DOF) que es un proceso inherentemente no lineal, para el cual se ha diseñado un controlador no lineal multivariable de 3 entradas y 3 salidas de tal manera que permita al robot seguir en forma asintótica una trayectoria cualquiera continua y diferenciable donde las entradas al sistema son los voltajes de armadura a los servomotores DC usados en las articulaciones y las salidas \mathbf{q}_1 , \mathbf{q}_2 y \mathbf{q}_3 son las posiciones del manipulador.

1.1 Objetivos

Los objetivos del presente trabajo son:

- a. Diseñar el modelo dinámico del Robot Manipulador Esférico de 3 Grados de Libertad empleando el algoritmo Lagrange-Euler.
- b. Explicar la técnica de control no lineal por realimentación de estado para sistemas de múltiples entradas y múltiples salidas.
- c. Diseñar el controlador no lineal y simular el sistema aplicando el método de linealización exacta por realimentación de estado aplicado al robot manipulador esférico.

1.2 Alcance

La culminación de la presente tesis es un aporte más a la Facultad en el conocimiento de esta importante técnica de control no lineal de múltiples entradas y múltiples salidas. Para demostrar la técnica expuesta se ha escogido un tema de robótica ya que es una tecnología con más auge en la actualidad y con mayor impacto en procesos de manufactura, ya que se puede lograr mayor precisión en el control de trayectorias de trabajo. De esta manera se ha escogido como aplicación al Robot Manipulador Esférico de 3 Grados de Libertad adecuado para mostrar

el funcionamiento del control no lineal. Así mismo queda explicado para poder construir e implementar dicho sistema en tiempo real ya que permitiría a la Facultad avanzar en el campo de la construcción y pruebas de funcionamiento de manipuladores robóticos con técnicas de control no lineal. Además este know-how puede luego ser trasladado a alguna aplicación industrial para ejecutar múltiples tareas.

Capítulo 2

Planteamiento del Problema

2.1 Enunciado del Problema

Los manipuladores pueden ser construídos y modelados con muchos cuerpos rígidos conectados en serie por articulaciones rotativas y conducidas cada una de ellas por actuadores eléctricos, que son servomotores DC.

El robot manipulador esférico de 3DOF está compuesto por un servomotor DC con encoder óptico incorporado y un sólido de aluminio (disco y barra) que vendría a ser la base para el primer grado de libertad; otro servomotor DC también con encoder óptico y una varilla rígida de aluminio para el segundo grado de libertad; y de la misma manera otro servomotor DC también con encoder óptico y una varilla rígida de aluminio que pueda cumplir la función de un brazo para el tercer grado de libertad, pudiéndose instalar en su extremo de esta una pinza rotórica.

El propósito de esta tesis es el control simultáneo del movimiento rotacional de la base acoplado al eje del primer servomotor DC en posición vertical; el otro movimiento rotacional de la primera varilla acoplada al eje del segundo servomotor DC en posición horizontal y finalmente el movimiento rotacional de la segunda varilla acoplada al eje del tercer servomotor DC en posición horizontal.

Para este trabajo en estudio se diseñará un controlador no lineal MIMO (Multiple-input-Multiple-output) de 3 entradas y 3 salidas basado en el método de linealización exacta por realimentación de estado de tal manera que el manipulador siga trayectorias de referencia arbitrarias empleando fuerzas de control adecuada.

2.2 Lista de las Variables

Todas las variables que intervienen en el sistema se describen en la tabla 2.1. Los valores que se presentan han sido tomados de los servomotores DC con encoder óptico MATSUSHITA modelo GMX-7MCO19B y modelo GMX-

6MPO13A. Donde: C.F.V.: coeficiente de fricción viscosa y M.I.: momento de inercia.

Tabla 2.1: valores de las variables

Símbolo	Descripción	Valor
m_d	Masa del disco	0.55 kg
m_p	Masa de la barra	1.00 kg
m_2, m_3	Masa de la 1ra. y 2da. varilla	0.20 kg
h_d	Altura del disco	0.01 m
R_d	Radio del disco	0.07 m
h_p	Altura de la barra	0.20 m
a_p	Ancho y Largo de la sección de la barra	0.044 m
L_2, L_3	Longitud de la 1ra. y 2da. varilla	0.15 m
$L_{x_{s_2}}$	Long. del eje de ref. S_1 al CG 1ra. varilla	0.10 m
$L_{x_{s_3}}$	Long. del eje de ref. S_2 al CG 2da. varilla	0.10 m
$L_{c_{s_2}}$	Long. del eje de ref. S_2 al CG 1ra. varilla	$(L_2 - L_{x_{s_2}})$ m
$L_{c_{s_3}}$	Long. del eje de ref. S_3 al CG 2da. varilla	$(L_3 - L_{x_{s_3}})$ m
I_1	M.I. del sólido (disco más barra)	0.0013 kgm^2
I_2	M.I. de la 1ra. varilla	0.0273 kgm^2
I_3	M.I. de la 2da. varilla	0.0273 kgm^2
J_{m1}, J_{m2}	M.I. servomotor DC1,DC2	0.00059 kgm^2
J_{m3}	M.I. servomotor DC3	0.000238 kgm^2
J_{eq1}	M.I. equivalente servomotor DC1	0.2177 kgm^2
B_{eq1}	C.F.V. equivalente servomotor DC1	0.054 Nms/rad
J_{eq2}	M.I. equivalente servomotor DC2	0.3954 kgm^2
B_{eq2}	C.F.V. equivalente servomotor DC2	0.0589 Nms/rad
J_{eq3}	M.I. equivalente servomotor DC3	0.000305 kgm^2
B_{eq3}	C.F.V. equivalente servomotor DC3	0.000538 Nms/rad
n_1, n_2	Reducción servomotor DC1,DC2	18.5
n_3	Reducción servomotor DC3	12.5
R_{a1}, R_{a2}	Resist. armadura servomotor DC1,DC2	3.50 Ohmios
R_{a3}	Resist. armadura servomotor DC3	5.30 Ohmios
L_{a1}, L_{a2}	Inductancia armadura servom. DC1,DC2	Despreciable
L_{a3}	Inductancia armadura servomotor DC3	Despreciable
V_{a1}, V_{a2}	Voltage armadura servomotor DC1,DC2	24 V
V_{a3}	Voltage armadura servomotor DC3	24 V
K_{A1}, K_{A2}, K_{A3}	Constante ganancia driver 1,2,3	2.17
K_{m1}, K_{m2}	Constante torque servomotor DC1,DC2	0.04364 $N - m/A$
K_{m3}	Constante torque servomotor DC3	0.04217 $N - m/A$
K_{b1}, K_{b2}	Constante fem servomotor DC1,DC2	0.04364 $V/rad/s$
K_{b3}	Constante fem servomotor DC3	0.04217 $V/rad/s$

Símbolo	Descripción	Valor
u_1, u_2, u_3	Señal de control servomotor DC1,DC2,DC3	(En simulación) V
g	Aceleración de la gravedad	$9.81m/seg^2$
r_{m2}	Distancia extremo superior de la barra al DC2	$0.05m$
C_{c1}	Coefficiente fricción de Coulomb 1 y 2DOF	$0.001488kgm^2/seg$
C_{c3}	Coefficiente fricción de Coulomb 3DOF	$0.00015kgm^2/seg$

Capítulo 3

Modelado de la Planta

3.1 Introducción

El modelado de la planta consiste en determinar la descripción matemática (usualmente un conjunto de ecuaciones diferenciales) del proceso a ser controlado, que en nuestro caso es el robot manipulador esférico. Tal modelado describe la dinámica del proceso y tiene por objetivo conocer la relación entre el movimiento del robot y las fuerzas implicadas en el mismo. Para determinar las ecuaciones del modelo usaremos el algoritmo de Lagrange-Euler [1], [4], [6], [9].

3.2 Descripción del Manipulador

El Sistema Robot Manipulador Esférico de 3 Grados de Libertad que en adelante lo llamaremos SRM (Spherical Robot Manipulator) de 3DOF es mostrado en la figura 3.1, en la figura 3.2 se muestra una vista del esquema anterior con las medidas del manipulador, y en la figura 3.3 se muestra los sistemas de referencias usados. A continuación se describe los tres grados de libertad usados en el proceso:

- a. **Primer Grado de Libertad:** compuesto por un servomotor DC1 de 24 VDC con encoder óptico incorporado, y una base conformado por un disco y una barra de aluminio; de tal manera que en la parte inferior del disco de radio R_d y altura h_d , irá acoplado con una bocina de bronce dicho servomotor en posición vertical y sobre la parte superior del disco irá una barra de sección cuadrada a_p , y de altura h_p , a su vez en la parte superior de dicha barra irá el otro servomotor en posición horizontal que corresponde al segundo grado de libertad.
- b. **Segundo Grado de Libertad:** compuesto por un servomotor DC2 de 24 VDC con encoder óptico incorporado, y una varilla rígida de aluminio, de tal manera que en el eje de dicho servomotor, en posición horizontal,

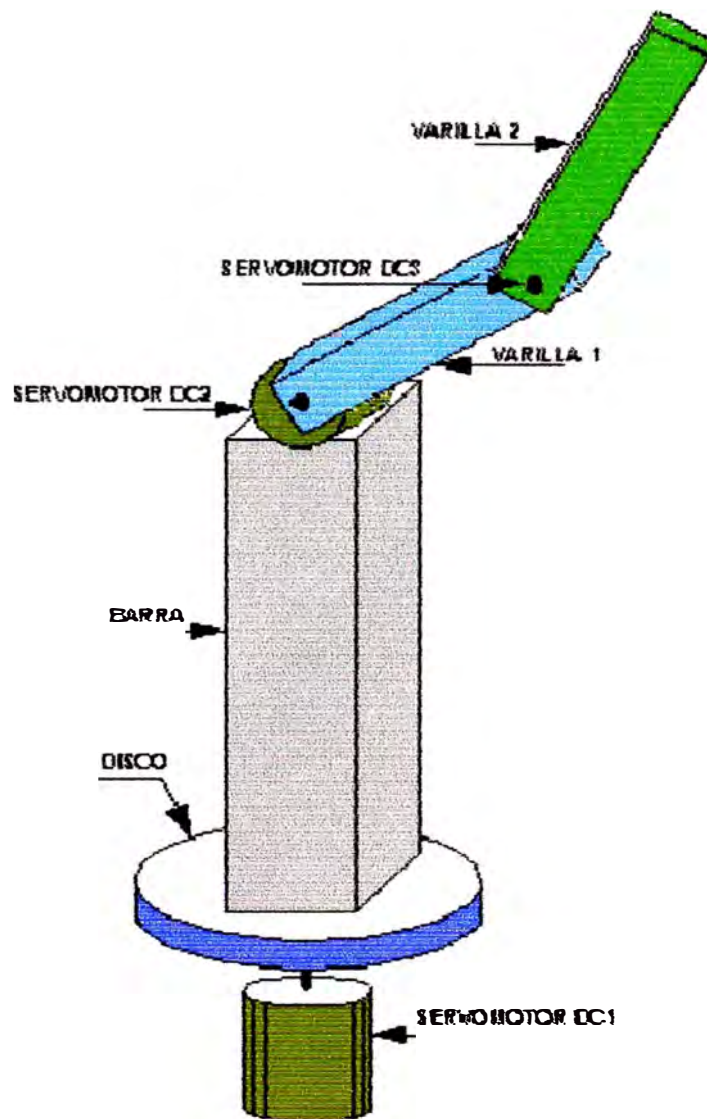


Figura 3.1: Esquema del Robot Manipulador Esférico de 3DOF

irá acoplado con una bocina de bronce dicha varilla de longitud L_2 , en el extremo de dicha varilla irá el otro servomotor en posición horizontal que corresponde al tercer grado de libertad.

- c. **Tercer Grado de Libertad:** compuesto por otro servomotor DC3 de 24 VDC con encoder óptico incorporado, y una varilla rígida de aluminio, de tal manera que en el eje de dicho servomotor, en posición horizontal, irá acoplado con una bocina de bronce dicha varilla de longitud L_3 , en el extremo de dicha varilla podrá ir un efector final.

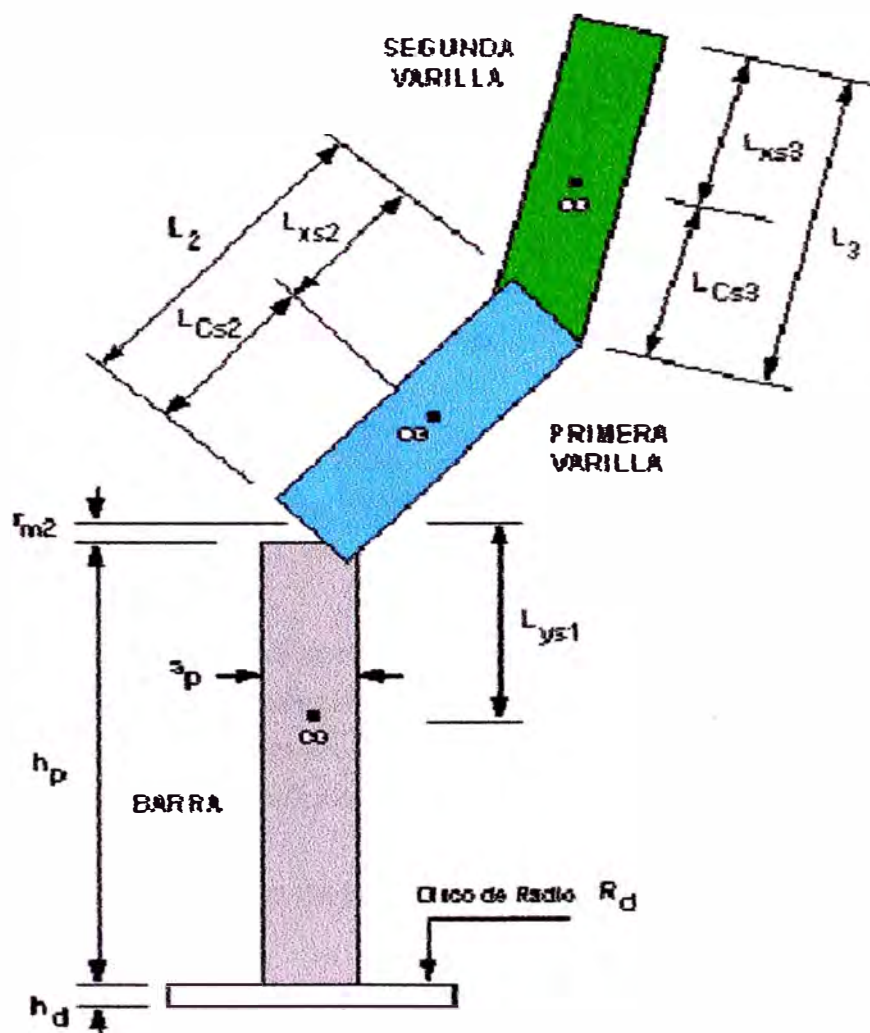
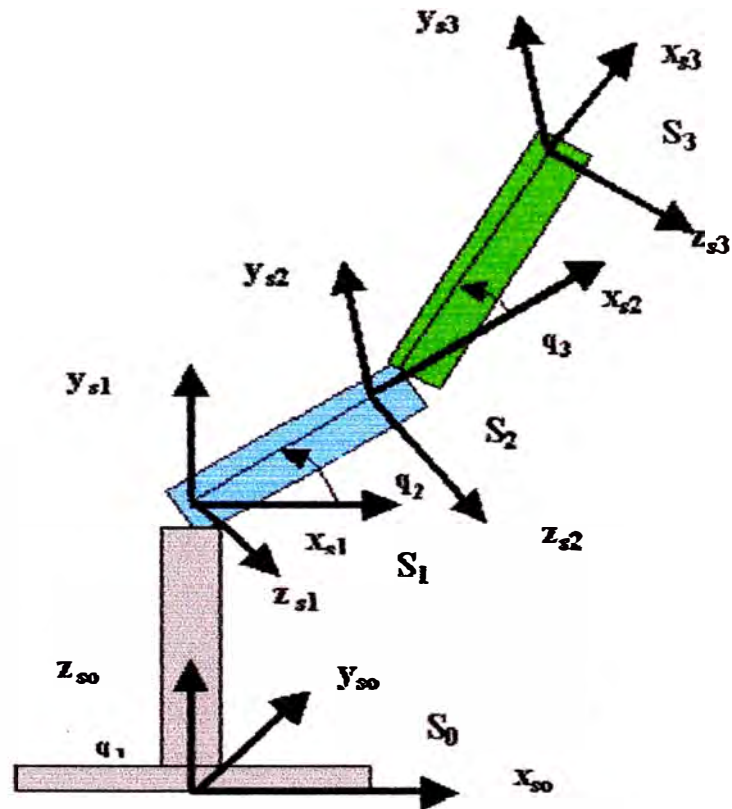


Figura 3.2: Esquema con las medidas del Manipulador



- S_0 : Sistema de referencia base (eje de motor 1)**
- S_1 : Sistema de referencia 1 (eje de motor 2)**
- S_2 : Sistema de referencia 2 (eje de motor 3)**
- S_3 : Sistema de referencia 3**

q_1, q_2, q_3 : ángulos de posición

Figura 3.3: Sistemas de referencias del Manipulador

3.3 El Modelo Dinámico

Para determinar la ecuación dinámica del manipulador existen varios métodos tales como el método de Newton-Euler, el método de Lagrange-Euler, entre otros. Para el manipulador en estudio usaremos el método de Lagrange-Euler.

El método de Lagrange-Euler emplea la fórmula Lagrangiana conjuntamente con la representación de Denavit-Hartenberg (D-H). Un completo estudio de dicho procedimiento puede ser encontrado en [1], [4], [6] [9].

3.3.1 El Procedimiento Denavit-Hartenberg

Un manipulador robótico consiste de una secuencia de cuerpos rígidos (los eslabones) articulados por juntas rotacionales o prismáticas. Cada par de articulación-eslabón representa un grado de libertad (DOF). Un sistema de coordenadas ortonormales $(x_{s_i}, y_{s_i}, z_{s_i})$ puede ser establecido usando la regla de la mano derecha para cada articulación i , donde $i = 1, \dots, n$. Así un manipulador con n DOF posee n pares de articulaciones. El último sistema de coordenadas $(x_{s_n}, y_{s_n}, z_{s_n})$ constituye el sistema de coordenadas del efector final, el cual es usualmente articulado al último eslabón. El manipulador es unido a una base de referencia 0. El sistema de coordenada base es definido como $(x_{s_0}, y_{s_0}, z_{s_0})$. Las coordenadas de la base son también las coordenadas inerciales del manipulador. El sistema de coordenadas base aumentado es definido como: $(x_{s_0}, y_{s_0}, z_{s_0}, 0)$. El procedimiento D-H origina una matriz de transformación homogénea que representa el sistema inercial del manipulador. Para describir tal matriz, requerimos establecer el sistema de coordenadas $(x_{s_i}, y_{s_i}, z_{s_i})$ del manipulador en $s_i = 0, \dots, n$, para determinar los parámetros de las articulaciones y eslabones con el fin de desarrollar la matriz de transferencia homogénea.

3.3.1.1 Estableciendo el Sistema de Coordenadas D-H

Para establecer el sistema de coordenadas D-H $(x_{s_i}, y_{s_i}, z_{s_i})$, $s_i = 0 \dots n$, se usa las siguientes reglas:

1. El sistema de coordenadas base $(x_{s_0}, y_{s_0}, z_{s_0})$ se determina usando la regla de la mano derecha. El eje z_{s_0} se escoge libremente y está localizado a lo largo del eje del movimiento del primer eslabón.

2. El eje $z_{s_{i-1}}$ es alineado con el eje del movimiento (giratorio o prismático) de la i -ésima articulación.
3. El origen del sistema de coordenadas s_i -ésima está en la intersección del eje z_{s_i} y $z_{s_{i-1}}$ o en la intersección de la normal común entre los ejes de dos articulaciones consecutivas z_{s_i} y $z_{s_{i-1}}$ y el eje z_{s_i} .
4. El eje x_{s_i} es perpendicular al eje $z_{s_{i-1}}$.
5. Aplicar la regla de la mano derecha en la articulación i -ésima para determinar y_{s_i} .
6. Generalmente, la articulación n -ésima es una articulación giratoria. El sistema de coordenadas $(x_{s_n}, y_{s_n}, z_{s_n})$, puede estar en cualquier parte del efector final con tal de que el eje z_{s_n} esté a lo largo de la dirección del eje $z_{s_{i-1}}$ y apuntando hacia afuera del manipulador. El eje x_{s_n} es perpendicular a ambos ejes: z_{s_n} y $z_{s_{n-1}}$.

3.3.1.2 Parámetros D-H

Los cuatro parámetros geométricos del manipulador (θ_i , d_{DH_i} , a_{DH_i} , y α_i), se asocian con cada par articulación-eslabón y son requeridos para describir una articulación giratoria o prismática. Después de establecer el sistema de coordenadas del manipulador tales parámetros pueden ser determinados de la siguiente manera:

1. θ_i , es el ángulo de rotación del eje $x_{s_{i-1}}$ con el eje x_{s_i} en el plano del eje $z_{s_{i-1}}$. Este es variable si la articulación es giratoria.
2. d_{DH_i} , es la distancia del origen del sistema de coordenadas $s_{(i-1)}$ -ésima a lo largo del eje $z_{s_{i-1}}$, hasta la intersección del eje $z_{s_{i-1}}$ con el eje x_{s_i} . Esta distancia d_{DH_i} es variable si la articulación i es prismática.
3. a_{DH_i} , es la distancia a lo largo del eje x_{s_i} , que va desde la intersección del eje $z_{s_{i-1}}$ con el eje x_{s_i} hasta el origen del sistema de coordenadas s_i -ésima en el caso de articulaciones giratorias.
4. α_i , es el ángulo de separación entre los ejes $z_{s_{i-1}}$ y z_{s_i} medido en un plano perpendicular al eje x_{s_i} , utilizando la regla de la mano derecha.

3.3.1.3 Matriz de Transformación Homogénea D-H

La matriz de transformación homogénea D-H caracteriza cada sistema de coordenadas del par articulación-eslabón con respecto al sistema de coordenadas previo. Por consiguiente un punto r_{s_i} del sistema de coordenadas s_i -ésimo se puede ser expresar como $(r_{s_{i-1}})$ en el sistema de coordenadas $s_{(i-1)}$ -ésimo ejecutando sucesivamente las siguientes operaciones:

1. $T(z_s, \theta_i)$: rotación de un ángulo θ_i alrededor del eje $z_{s_{i-1}}$, para alinear el eje $x_{s_{i-1}}$ con el eje x_{s_i} .
2. $T(z_s, d_{DH_i})$: traslación a lo largo del eje $z_{s_{i-1}}$ de una distancia d_{DH_i} , hasta hacer coincidir el eje $z_{s_{i-1}}$ con el eje x_{s_i} .
3. $T(x_s, a_{DH_i})$: traslación a lo largo del eje x_{s_i} de una distancia a_{DH_i} , que va desde la intersección del eje $z_{s_{i-1}}$ y el eje x_{s_i} al origen del sistema de coordenadas s_i -ésimo.
4. $T(x_s, \alpha_i)$: rotación de un ángulo α_i , del eje $z_{s_{i-1}}$ con el eje z_{s_i} alrededor del eje x_{s_i} .

El producto de estas cuatro operaciones básicas produce la matriz de transformación homogénea A_i^{i-1} , del eslabón i con respecto al eslabón $i - 1$ o articulación i con respecto a la articulación $i - 1$. Por lo tanto:

$$\begin{aligned}
 A_i^{i-1} &= T(z, \theta_i)T(0, 0, d_{DH_i})T(a_{DH_i}, 0, 0)T(x, \alpha_i) = \\
 &= \begin{bmatrix} c(\theta_i) & -s(\theta_i) & 0 & 0 \\ s(\theta_i) & c(\theta_i) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_{DH_i} \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & a_{DH_i} \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & c(\alpha_i) & -s(\alpha_i) & 0 \\ 0 & s(\alpha_i) & c(\alpha_i) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\
 &= \begin{bmatrix} c(\theta_i) & -c(\alpha_i)s(\theta_i) & s(\alpha_i)s(\theta_i) & a_{DH_i}c(\theta_i) \\ s(\theta_i) & c(\alpha_i)c(\theta_i) & -s(\alpha_i)c(\theta_i) & a_{DH_i}s(\theta_i) \\ 0 & s(\alpha_i) & c(\alpha_i) & d_{DH_i} \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.1)
 \end{aligned}$$

donde c y s , son las funciones \cos y \sin respectivamente. La siguiente matriz de transformación homogénea:

$$T_i^0 = \prod_{j=1}^i A_j^{j-1} = A_1^0 A_2^1 \dots A_i^{i-1} \quad i = 1, 2, \dots, n \quad (3.2)$$

determina la localización del sistema de coordenada (s_i)-ésima con respecto al sistema de coordenada base.

3.3.2 El Algoritmo de Lagrange-Euler [1], [6]

El procedimiento de Lagrange-Euler requiere ejecutar los siguientes pasos :

1. Asignar a cada par articulación-eslabón del manipulador un sistema de coordenadas D-H.
2. Obtener las matrices de transformación homogénea A_i^{i-1} para $i = 1, \dots, n$.
3. Obtener las matrices U_{ij} que es el efecto del movimiento de la articulación j en todos los puntos del eslabón i :

$$U_{ij} = \frac{\partial A_i^0}{\partial q_j} = \begin{cases} A_{j-1}^0 Q_j A_i^{j-1} & \text{si } j \leq i \\ 0 & \text{si } j > i \end{cases} \quad (3.3)$$

donde A_{j-1}^0 relaciona al sistema de coordenadas ($j-1$)-ésimo con el sistema de coordenadas base, y Q_j toma los siguientes valores:

$$Q_j = \begin{bmatrix} 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad \text{si la articulación es giratoria}$$

$$Q_j = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix} \quad \text{si la articulación es prismática}$$

4. Los efectos de interacción entre las articulaciones son expresadas por las siguientes matrices U_{ijk} :

$$U_{ijk} = \frac{\partial U_{ij}}{\partial q_k} = \frac{\partial}{\partial q_k} \left(\frac{\partial A_i^0}{\partial q_j} \right) \begin{cases} A_{j-1}^0 Q_j A_{k-1}^{j-1} Q_k A_i^{k-1} & \text{si } i \geq k \geq j \\ A_{k-1}^0 Q_k A_{j-1}^{k-1} Q_j A_i^{j-1} & \text{si } i \geq j \geq k \\ 0 & \text{si } k > i \text{ o } j > i \end{cases} \quad (3.4)$$

5. Obtener el tensor de inercia J_i del eslabón i -ésimo mediante:

$$J_i = \begin{bmatrix} \int x_{s_i}^2 dm & \int x_{s_i} y_{s_i} dm & \int x_{s_i} z_{s_i} dm & \int x_{s_i} dm \\ \int x_{s_i} y_{s_i} dm & \int y_{s_i}^2 dm & \int y_{s_i} z_{s_i} dm & \int y_{s_i} dm \\ \int x_{s_i} z_{s_i} dm & \int y_{s_i} z_{s_i} dm & \int z_{s_i}^2 dm & \int z_{s_i} dm \\ \int x_{s_i} dm & \int y_{s_i} dm & \int z_{s_i} dm & \int dm \end{bmatrix} \quad (3.5)$$

donde:

$\int x_{s_i}^2 dm$: momento de inercia respecto al eje x_{s_i}

$\int y_{s_i}^2 dm$: momento de inercia respecto al eje y_{s_i}

$\int z_{s_i}^2 dm$: momento de inercia respecto al eje z_{s_i}

$\int x_{s_i} y_{s_i} dm$: producto de momento de inercia respecto a los ejes x_{s_i} e y_{s_i}

$\int y_{s_i} z_{s_i} dm$: producto de momento de inercia respecto a los ejes y_{s_i} y z_{s_i}

$\int x_{s_i} z_{s_i} dm$: producto de momento de inercia respecto a los ejes x_{s_i} y z_{s_i}

$\int x_{s_i} dm$: distancia del centro de gravedad del cuerpo i al sistema x_{s_i}

$\int y_{s_i} dm$: distancia del centro de gravedad del cuerpo i al sistema y_{s_i}

$\int z_{s_i} dm$: distancia del centro de gravedad del cuerpo i al sistema z_{s_i}

$\int dm$ es la masa del cuerpo i

6. Obtener la matriz H de inercia simétrica de dimensión $n \times n$, la cual está relacionada con la aceleración y cuyos elementos se expresan como:

$$H_{ik}(\mathbf{q}) = \sum_{j=\max(i,k)}^n Tr(U_{jk} J_j U_{ji}^T) \quad i, k = 1, 2 \dots n \quad (3.6)$$

donde, Tr es la traza y n es el número de grados de libertad.

7. Obtener los términos h_{ikm} definidos por:

$$h_{ikm} = \sum_{j=\max(i,k,m)}^n Tr(U_{jkm} J_j U_{ji}^T) \quad i, k, m = 1, 2 \dots n \quad (3.7)$$

8. Obtener el vector no lineal fuerza centrífuga y Coriolis de orden $n \times 1$ cuyos elementos C_i son dados por:

$$C_i(\mathbf{q}, \dot{\mathbf{q}}) = \sum_{k=1}^n \sum_{m=1}^n h_{ikm} \dot{q}_k \dot{q}_m \quad (3.8)$$

9. Obtener el vector no lineal de fuerza de gravedad d de orden $n \times 1$ cuyos elementos son:

$$d_i(\mathbf{q}) = \sum_{j=1}^n (-m_j g U_{ji} \bar{r}_{s_j}^j) \quad i = 1, 2, \dots, n \quad (3.9)$$

donde, $g = [g_{x_{s_0}} \ g_{y_{s_0}} \ g_{z_{s_0}} \ 0]$ es el vector gravedad fila expresado en el sistema de coordenadas base y $\bar{r}_{s_j}^j = [\bar{x}_{s_j} \ \bar{y}_{s_j} \ \bar{z}_{s_j} \ 1]^T$ (el elemento 1 es un factor de escala) es el vector centro de masa del eslabón j y expresado en el sistema de coordenadas s_j -ésimo.

Finalmente la ecuación dinámica del manipulador, con el método L-E toma la forma:

$$\mathbf{T}_i = \mathbf{H}_{ik}(\mathbf{q})\ddot{\mathbf{q}}_i + \mathbf{C}_i(\mathbf{q}, \dot{\mathbf{q}}) + d_i(\mathbf{q}) \quad (3.10)$$

donde, \mathbf{T}_i es el vector de fuerzas y torques aplicado a cada coordenada generalizada q_i .

3.4 Aplicación y Programación del Modelo

3.4.1 Cálculos

La aplicación del trabajo es el robot manipulador esférico de tres grados de libertad. Para determinar la ecuación dinámica de dicho manipulador se aplica el algoritmo de Lagrange-Euler. A continuación se presenta los siguientes cálculos que usaremos para la programación con matemática simbólica de MATLAB 6.0 [11].

3.4.1.1 Cálculo de los Parámetros D-H

Primeramente se fijan los sistemas de referencia en las articulaciones como se muestra en la figura 3.3, posteriormente se determina los parámetros D-H (DENAVID-HARTENBERG), [1], [4], [6], [9], que se muestran en la Tabla 3.1:

3.4.1.2 Cálculo de las Matrices de Transformación Homogénea

Reemplazando los valores de la tabla 3.1 en la ecuación (3.1), obtenemos las matrices de transformación homogénea \mathbf{A}_1^0 , \mathbf{A}_2^1 y \mathbf{A}_3^2 :

Tabla 3.1: valores de los parámetros $D - H$

Juntura	θ_i	d_{DH_i}	a_{DH_i}	α_i
1	q_1	$h_p + h_d + r_{m2}$	0	90°
2	q_2	0	L_2	0°
3	q_3	0	L_3	0°

A_1^0 : posición y orientación del sistema de referencia 1 (S_1) de la base con respecto al sistema de referencia base (S_0):

$$A_1^0 = \begin{bmatrix} \cos(q_1) & 0 & \text{sen}(q_1) & 0 \\ \text{sen}(q_1) & 0 & -\cos(q_1) & 0 \\ 0 & 1 & 0 & h_p + h_d + r_{m2} \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

A_2^1 : posición y orientación del sistema de referencia 2 (S_2) de la primera varilla con respecto al sistema de referencia 1 (S_1):

$$A_2^1 = \begin{bmatrix} \cos(q_2) & -\text{sen}(q_2) & 0 & L_2 \cos(q_2) \\ \text{sen}(q_2) & \cos(q_2) & 0 & L_2 \text{sen}(q_2) \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

A_3^2 : posición y orientación del sistema de referencia 3 (S_3) de la segunda varilla con respecto al sistema de referencia 2 (S_2):

$$A_3^2 = \begin{bmatrix} \cos(q_3) & -\text{sen}(q_3) & 0 & L_3 \cos(q_3) \\ \text{sen}(q_3) & \cos(q_3) & 0 & L_3 \text{sen}(q_3) \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

A_2^0 : posición y orientación del sistema de referencia 2 (S_2) de la primera varilla con respecto al sistema de referencia base (S_0):

$$A_2^0 = A_1^0 A_2^1$$

A_3^0 : posición y orientación del sistema de referencia 3 (S_3) de la segunda varilla con respecto al sistema de referencia base (S_0):

$$A_3^0 = A_1^0 A_2^1 A_3^2$$

A_2^0 y A_3^0 se determina mediante matemática simbólica de MATLAB, y el programa se muestra en el apéndice B.

3.4.2.3 Cálculo del momento de inercia del disco [5], [8]

Para determinar el momento de inercia del disco respecto al sistema de referencia 1 (S_1) x_{s1} , y_{s1} , z_{s1} , se usa el Teorema de Steiner [5], (ver figura 3.4).

Cálculo del momento de inercia respecto a los ejes coordenados:

$$\int x_{s1}^2 dm = \int x_{s1CG}^2 dm + m\bar{x}^2$$

$$\int y_{s1}^2 dm = \int y_{s1CG}^2 dm + m\bar{y}^2$$

$$\int z_{s1}^2 dm = \int z_{s1CG}^2 dm + m\bar{z}^2$$

$$\bar{x} = \bar{z} = 0, \quad \bar{y} = -\left(\frac{h_d}{2} + h_p + r_{m2}\right)$$

calculando y reemplazando se obtiene:

$$\int x_{s1}^2 dm = \int z_{s1}^2 dm = \frac{m_d h_d^2}{4}$$

$$\int y_{s1}^2 dm = \frac{m_d h_d^2}{12} + m_d \left(\frac{h_d}{2} + h_p + r_{m2}\right)^2$$

cálculo del producto de los momento de inercia:

$$\int x_{s1CG} y_{s1CG} dm = \int x_{s1CG} z_{s1CG} dm = \int y_{s1CG} z_{s1CG} dm = 0 \text{ (por simetría respecto al centro de gravedad CG)}$$

reemplazando lo anterior se obtiene:

$$\int x_{s1} y_{s1} dm = \int x_{s1CG} y_{s1CG} dm + m\bar{x}\bar{y} = 0$$

$$\int y_{s1} z_{s1} dm = \int y_{s1CG} z_{s1CG} dm + m\bar{y}\bar{z} = 0$$

$$\int x_{s1} z_{s1} dm = \int x_{s1CG} z_{s1CG} dm + m\bar{x}\bar{z} = 0$$

3.4.2.4 Cálculo del momento de inercia de la barra [5], [8]

Para calcular el momento de inercia de la barra respecto al sistema de referencia 1 (S_1) x_{s1} , y_{s1} , z_{s1} , (ver figura 3.5), se procede de la misma manera que para el momento de inercia del disco, para este caso se tiene:

$$\bar{x} = \bar{z} = 0, \quad \bar{y} = -\left(\frac{h_p}{2} + r_{m2}\right)$$

$$\int x_{s1}^2 dm = \int z_{s1}^2 dm = \frac{m_p a_p^2}{12}$$

$$\int y_{s1}^2 dm = \frac{m_p h_p^2}{12} + m_p \left(\frac{h_p}{2} + r_{m2}\right)^2$$

$$\int y_{s1} z_{s1} dm = \int y_{s1} z_{s1} dm = \int x_{s1} z_{s1} dm = 0$$

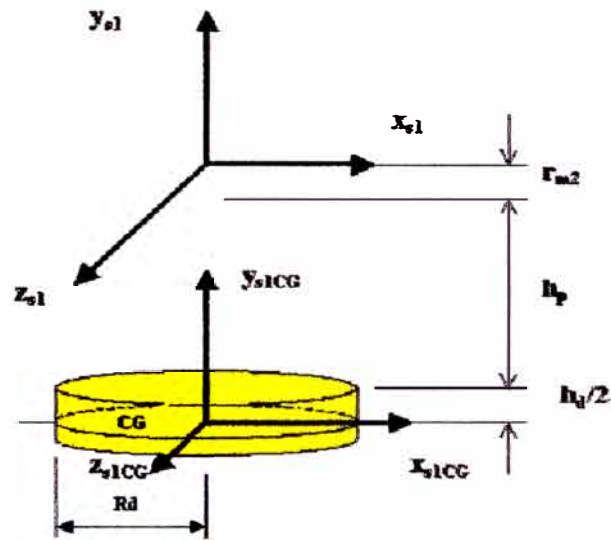


Figura 3.4: Momento de inercia del disco respecto al sistema de referencia 1 (S_1).

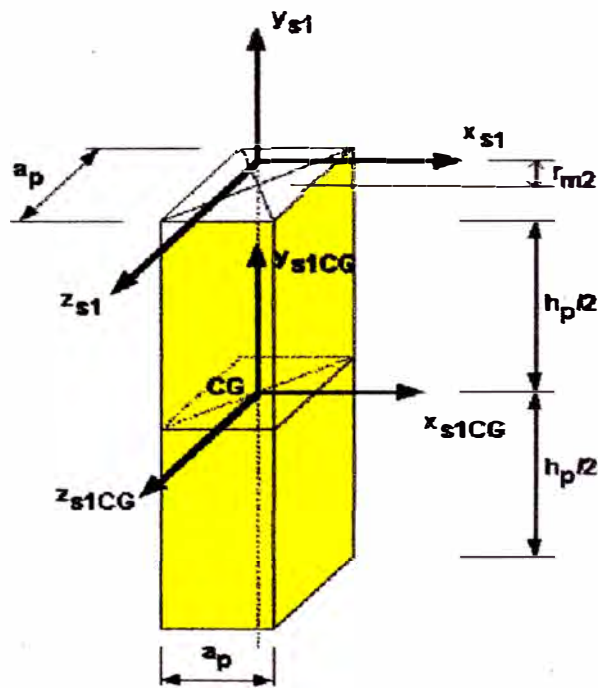


Figura 3.5: Momento de inercia de la barra respecto al sistema de referencia 1 (S_1)

3.4.2.5 Cálculo del momento de inercia de la base (disco más barra)

El momento de inercia de la base es la suma de los momentos de inercia del disco más la barra, y está dado por:

$$I_{11} = \int x_{s1}^2 dm = \int z_{s1}^2 dm = \frac{m_d h_d^2}{4} + \frac{m_p a_p^2}{12}$$

$$I_1 = \int y_{s1}^2 dm = \frac{m_p h_p^2}{12} + \frac{m_d h_d^2}{12} + m_p \left(\frac{h_p}{2} + r_{m2} \right)^2 + m_d \left(\frac{h_d}{2} + h_p + r_{m2} \right)^2$$

3.4.2.6 Cálculo de la matriz de pseudoinercia de la base J_1

De la ecuación (3.5) se obtiene la matriz de pseudoinercia de la base J_1 :

$$J_1 = \begin{bmatrix} I_{11} & 0 & 0 & 0 \\ 0 & I_1 & 0 & -m_1 L_{y_{s1}} \\ 0 & 0 & I_{11} & 0 \\ 0 & -m_1 L_{y_{s1}} & 0 & m_1 \end{bmatrix} \quad (3.11)$$

donde:

m_1 : $m_d + m_p + m_{\text{accesorios}}$

$L_{y_{s1}}$: distancia del CG de la barra al eje del servomotor DC2. No se toma, ya que más adelante en el producto de los elementos de las matrices se anula .

3.4.2.7 Cálculo de la matriz de pseudoinercia de la primera varilla J_2

Para determinar el momento de inercia de la primera varilla se obtiene de [5], [8]:

$$\int y_{s2}^2 dm = \int z_{s2}^2 dm = 0 \text{ (debido a que el 2DOF gira respecto del eje } x_{s2} \text{)}$$

$$I_2 = \int x_{s2}^2 dm = m_2 L_2^2 / 3$$

de la misma manera se calcula los productos de los momentos de inercia:

$$\int x_{s2} y_{s2} dm = \int x_{s2CG} y_{s2CG} dm + m \bar{x} \bar{y}$$

$$\int y_{s2} z_{s2} dm = \int y_{s2CG} z_{s2CG} dm + m \bar{y} \bar{z}$$

$$\int x_{s2} z_{s2} dm = \int x_{s2CG} z_{s2CG} dm + m \bar{x} \bar{z}$$

$$\bar{y} = \bar{z} = 0$$

$$\bar{x} = -L_{x_{s2}}$$

$$\int x_{s2CG} y_{s2CG} dm = \int x_{s2CG} z_{s2CG} dm = \int y_{s2CG} z_{s2CG} dm = 0 \text{ (por simetría respecto al centro de gravedad CG)}$$

reemplazando de lo anterior se obtiene

$$\int x_{s_2} y_{s_2} dm = \int y_{s_2} z_{s_2} dm = \int x_{s_2} z_{s_2} dm = 0$$

reemplazando estos valores en la ecuación (3.5), obtenemos la matriz de pseudoinercia J_2 :

$$J_2 = \begin{bmatrix} I_2 & 0 & 0 & -m_2 L_{x_{s_2}} \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ -m_2 L_{x_{s_2}} & 0 & 0 & m_2 \end{bmatrix} \quad (3.12)$$

donde:

m_2 : $m_{\text{brazo}} + m_{\text{accesorios}}$

$L_{x_{s_2}}$: long. del eje de referencia 2 (S_2) al C.G. de la primera varilla

3.4.2.8 Cálculo de la matriz de pseudoinercia de la segunda varilla J_3

Para determinar el momento de inercia de la segunda varilla se obtiene de [5], [8], y similar a lo obtenido en la primera varilla:

$$\int y_{s_3}^2 dm = \int z_{s_3}^2 dm = 0$$

$$I_3 = \int x_{s_3}^2 dm = m_3 L_3^2 / 3$$

de la misma manera los productos de los momentos de inercia son:

$$\int x_{s_3} y_{s_3} dm = \int y_{s_3} z_{s_3} dm = \int x_{s_3} z_{s_3} dm = 0$$

reemplazando estos valores en la ecuación (3.5), obtenemos la matriz de pseudoinercia J_3 :

$$J_3 = \begin{bmatrix} I_3 & 0 & 0 & -m_3 L_{x_{s_3}} \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ -m_3 L_{x_{s_3}} & 0 & 0 & m_3 \end{bmatrix} \quad (3.13)$$

donde:

m_3 : $m_{\text{brazo}} + m_{\text{accesorios}}$

$L_{x_{s_3}}$: long. del eje de referencia 3 (S_3) al C.G. de la segunda varilla

3.4.2 Programación del Algoritmo Lagrange-Euler

El programa se ha determinado aplicando Matemática Simbólica de MATLAB 6.0 [11] y se presenta en el apéndice B. Los resultados obtenidos son:

$$\begin{aligned} \mathbf{H}_{11} = & 2I_{11} + \cos(q_2)^2 * (I_2 - m_2 * L_2^2 + 2 * m_2 * L_2 * L_{e_{s_2}}) + m_3 * L_3^2 * \cos(q_3)^2 \\ & + m_3 * L_3^2 * \cos(q_2)^2 - 2 * I_3 * \sin(q_2) * \sin(q_3) * \cos(q_2) * \cos(q_3) - m_3 * L_3^2 \\ & + 2 * m_3 * L_3^2 * \sin(q_2) * \sin(q_3) * \cos(q_2) * \cos(q_3) - I_3 * \cos(q_3)^2 - I_3 * \cos(q_2)^2 \end{aligned}$$

$$\begin{aligned}
& + 2 * I_3 * \cos(q_2)^2 * \cos(q_3)^2 + 2 * m_3 * L_2 * \cos(q_2)^2 * L_{c_{s3}} * \cos(q_3) + I_3 \\
& - 4 * m_3 * L_3 * \sin(q_2) * \sin(q_3) * L_{c_{s3}} * \cos(q_2) * \cos(q_3) + m_3 * L_2^2 * \cos(q_2)^2 \\
& - 2 * m_3 * L_3 * L_{c_{s3}} * \cos(q_3)^2 - 2 * m_3 * L_3 * L_{c_{s3}} * \cos(q_2)^2 \\
& - 2 * m_3 * L_3^2 * \cos(q_2)^2 * \cos(q_3)^2 + 4 * m_3 * L_3 * L_{c_{s3}} * \cos(q_2)^2 * \cos(q_3)^2 \\
& + 2 * m_3 * L_3 * L_{c_{s3}} - 2 * m_3 * L_2 * \cos(q_2) * L_{c_{s3}} * \sin(q_2) * \sin(q_3)
\end{aligned}$$

$$\mathbf{H}_{12} = \mathbf{H}_{13} = \mathbf{H}_{21} = 0$$

$$\mathbf{H}_{22} = I_2 - L_2^2 * m_2 + 2 * L_2 * m_2 * L_{c_{s2}} + m_3 * L_2^2 - m_3 * L_3^2$$

$$+ 2 * m_3 * L_2 * L_{c_{s3}} * \cos(q_3) + 2 * m_3 * L_3 * L_{c_{s3}} + I_3$$

$$\mathbf{H}_{23} = -m_3 * L_3^2 + m_3 * L_2 * L_{c_{s3}} * \cos(q_3) + 2 * m_3 * L_3 * L_{c_{s3}} + I_3$$

$$\mathbf{H}_{31} = 0$$

$$\mathbf{H}_{32} = -m_3 * L_3^2 + m_3 * L_2 * L_{c_{s3}} * \cos(q_3) + 2 * m_3 * L_3 * L_{c_{s3}} + I_3$$

$$\mathbf{H}_{33} = -m_3 * L_3^2 + 2 * m_3 * L_3 * L_{c_{s3}} + I_3$$

$$\begin{aligned}
\mathbf{C}_1 &= -2 * \cos(q_2) * \sin(q_2) * (I_2 - L_2^2 * m_2 + 2 * L_2 * m_2 * L_{c_{s2}}) * Dq1 * Dq2 \\
& + 2 * (-1/2 * I_3 * \sin(2 * q_2 + 2 * q_3) + 1/2 * L_3^2 * m_3 * \sin(2 * q_2 + 2 * q_3) \\
& - L_3 * m_3 * L_{c_{s3}} * \sin(2 * q_2 + 2 * q_3) - 1/2 * m_3 * L_{c_{s3}} * L_2 * \sin(2 * q_2 + q_3) - 1/2 * \\
& m_3 * L_{c_{s3}} * L_2 * \sin(q_3)) * Dq1 * Dq3
\end{aligned}$$

$$\begin{aligned}
\mathbf{C}_2 &= \sin(q_2) * \cos(q_2) * (I_2 - L_2^2 * m_2 + 2 * L_2 * m_2 * L_{c_{s2}}) * Dq1^2 \\
& - 2 * m_3 * L_{c_{s3}} * L_2 * \sin(q_3) * Dq2 * Dq3 - m_3 * L_{c_{s3}} * L_2 * \sin(q_3) * Dq3^2
\end{aligned}$$

$$\begin{aligned}
\mathbf{C}_3 &= (-1/2 * L_3^2 * m_3 * \sin(2 * q_2 + 2 * q_3) + 1/2 * I_3 * \sin(2 * q_2 + 2 * q_3) \\
& + L_3 * m_3 * L_{c_{s3}} * \sin(2 * q_2 + 2 * q_3) + 1/2 * m_3 * L_{c_{s3}} * L_2 * \sin(2 * q_2 + q_3) \\
& + 1/2 * m_3 * L_{c_{s3}} * L_2 * \sin(q_3)) * Dq1^2 + m_3 * L_{c_{s3}} * L_2 * \sin(q_3) * Dq2^2
\end{aligned}$$

$$\mathbf{d}_1 = 0$$

$$\begin{aligned}
\mathbf{d}_2 &= m_2 * g * \cos(q_2) * L_{c_{s2}} - m_3 * g * \sin(q_2) * \sin(q_3) * L_{c_{s3}} \\
& + m_3 * g * \cos(q_2) * \cos(q_3) * L_{c_{s3}} + m_3 * g * L_2 * \cos(q_2)
\end{aligned}$$

$$\mathbf{d}_3 = -m_3 * g * \sin(q_2) * \sin(q_3) * L_{c_{s3}} + m_3 * g * \cos(q_2) * \cos(q_3) * L_{c_{s3}}$$

donde:

$$Dq1 = \dot{q}_1, Dq2 = \dot{q}_2, Dq3 = \dot{q}_3$$

Las componentes se pueden expresar con los parámetros \mathbf{a}_j :

$$a_1 = 2I_{11}$$

$$a_2 = I_2 - L_2^2 + 2 * m_2 L_2 * m_2 * L_{c_{s2}}$$

$$a_3 = I_3 - L_3^2 + 2 * m_3 L_3 * m_3 * L_{c_{s3}}$$

$$a_4 = m_3 L_2^2$$

$$a_5 = m_3 * L_2 * L_{c_{s3}}$$

$$a_6 = m_2 * g * L_{c_{s2}} + m_3 * g * L_2$$

$$a_7 = m_3 * g * L_{c_{s3}}$$

de tal manera que las ecuaciones se simplifiquen y se expresen:

$$\mathbf{H}_{11} = a_1 + a_2 \cos^2(q_2) + a_3 \cos^2(q_2 + q_3) + a_4 \cos^2(q_2) + 2a_5 \cos(q_2) \cos(q_2 + q_3)$$

$$\mathbf{H}_{22} = a_2 + a_3 + a_4 + 2a_5 \cos(q_3)$$

$$\mathbf{H}_{23} = \mathbf{H}_{32} = a_3 + a_5 \cos(q_3)$$

$$\mathbf{H}_{33} = a_3$$

$$\mathbf{C}_1 = -a_2 \sin(2q_2) \dot{q}_1 \dot{q}_2 - 2(a_3 \sin(q_2 + q_3) \cos(q_2 + q_3) + a_5 \sin(q_2 + q_3) \cos(q_2)) \dot{q}_1 \dot{q}_3$$

$$\mathbf{C}_2 = a_2 \sin(q_2) \cos(q_2) \dot{q}_1^2 - 2a_5 \sin(q_3) \dot{q}_2 \dot{q}_3 - a_5 \sin(q_3) \dot{q}_3^2$$

$$\mathbf{C}_3 = (a_3 \sin(q_2 + q_3) \cos(q_2 + q_3) + a_5 \sin(q_2 + q_3) \cos(q_2)) \dot{q}_1^2 + a_5 \sin(q_3) \dot{q}_2^2$$

$$\mathbf{d}_2 = a_6 \cos(q_2) + a_7 \cos(q_2 + q_3)$$

$$\mathbf{d}_3 = a_7 \cos(q_2 + q_3)$$

Finalmente la ecuación dinámica obtenida para los tres grados de libertad son:

$$\text{para el primer grado de libertad: } T_1 = H_{11} \ddot{q}_1 + C_1 \quad (3.14)$$

$$\text{para el segundo grado de libertad: } T_2 = H_{22} \ddot{q}_2 + H_{23} \ddot{q}_3 + C_2 + d_2 \quad (3.15)$$

$$\text{para el tercer grado de libertad: } T_3 = H_{32} \ddot{q}_2 + H_{33} \ddot{q}_3 + C_3 + d_3 \quad (3.16)$$

3.4.3 El Modelo de los Actuadores

3.4.3.1 Modelado del Actuador No. 1 de la Base del Manipulador

En la figura 3.6, se muestra el servomotor DC1, correspondiente a la base.

El voltage de entrada $K_{A1}u_1$ aplicado a la armadura está dado por:

$$V_{A1} = K_{A1}u_1 = L_{a1} \dot{I}_{a1} + R_{a1} I_{a1} + V_{b1} \quad (3.17)$$

la fuerza electromotriz del servomotor DC es:

$$V_{b1} = K_{b1}w_{m1} = K_{b1}n_1 \dot{q}_1 \quad (3.18)$$

en el cual n_1 es la relación de piñones. El torque T_{m1} del servomotor está dado por:

$$T_{m1} = K_{m1}I_{a1} = J_{m1}n_1 \ddot{q}_1 + B_{m1}n_1 \dot{q}_1 + T_{g1} \quad (3.19)$$

por lo tanto la ecuación del torque T_{g2} requerido por el actuador para la base tiene la forma de:

$$T_{g2} = n_1 T_{g1} = J_{g1} \ddot{q}_1 + B_{g1} \dot{q}_1 + T_1 + n_1 T_{c1} \quad (3.20)$$

donde T_1 es el torque de la carga y T_{c1} es el torque de fricción de Coulomb. La (3.20) en (3.19) queda:

$$n_1 K_{m1} I_{a1} = J_{eq1} \ddot{q}_1 + B_{eq1} \dot{q}_1 + T_1 + n_1 T_{c1} \quad (3.21)$$

donde:

$$J_{eq1} = n_1^2 J_{m1} + J_{g1}$$

$$B_{eq1} = n_1^2 B_{m1} + B_{g1}$$

$$T_1 = H_{11} \ddot{q}_1 + C_1$$

$$H_{11} = a_1 + a_2 \cos^2(q_2) + a_3 \cos^2(q_2 + q_3) + a_4 \cos^2(q_2) + 2a_5 \cos(q_2) \cos(q_2 + q_3)$$

$$C_1 = -a_2 \sin(2q_2) \dot{q}_1 \dot{q}_2 - 2(a_3 \sin(q_2 + q_3) \cos(q_2 + q_3) + a_5 \sin(q_2 + q_3) \cos(q_2)) \dot{q}_1 \dot{q}_3$$

$$a_1 = I_{11}$$

$$a_2 = I_2 + 2m_2 L_2 L_{c_{s2}} - m_2 L_2^2$$

$$a_3 = I_3 + 2m_3 L_3 L_{c_{s3}} - m_3 L_3^2$$

$$a_4 = m_3 L_2^2$$

$$a_5 = m_3 L_2 L_{c_{s3}}$$

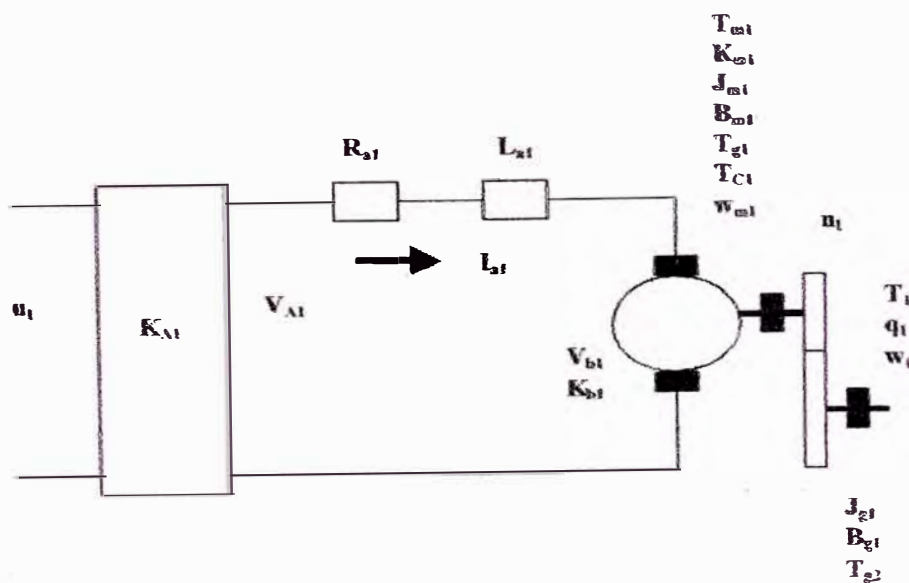


Figura 3.6: Servomotor DC1 de la base del Manipulador

despreciando la inductancia de armadura o sea $L_{a1} = 0$, obtenemos de (3.17) y (3.18)

$$K_{A1}u_1 = R_{a1}I_{a1} + K_{b1}n_1\dot{q}_1 \quad (3.22)$$

despejando I_{a1} en (3.21) y reemplazando en (3.22), obtenemos la ley de control para la primera entrada:

$$u_1 = R_{T1}(J_{eq1}\ddot{q}_1 + B_{eq1}\dot{q}_1 + T_1) + N_{T1}\dot{q}_1 + n_1R_{T1}T_{c1} \quad (3.23)$$

donde:

$$R_{T1} = \frac{R_{a1}}{n_1K_{m1}K_{A1}}$$

$$N_{T1} = \frac{n_1K_{b1}}{K_{A1}}$$

$$T_{c1} = C_{c1}sign(\dot{q}_1)$$

3.4.3.2 Modelado del Actuador No. 2 de la primera varilla del Manipulador

En la figura 3.7, se muestra el servomotor DC2 correspondiente a la primera varilla. En la misma figura se muestra el circuito eléctrico del servomotor DC2, controlado también por la armadura. Dicho servomotor se encuentra ubicado en la parte superior del prisma. Para obtener la ley de control para la segunda entrada se procede de la misma manera que para el primer servomotor, obteniendo de esta manera:

$$u_2 = R_{T2}(J_{eq2}\ddot{q}_2 + B_{eq2}\dot{q}_2 + T_2) + N_{T2}\dot{q}_2 + n_2R_{T2}T_{c2} \quad (3.24)$$

donde:

$$T_{c2} = C_{c1}sign(\dot{q}_2)$$

$$J_{eq2} = n_2^2J_{m2} + J_{g2}$$

$$B_{eq2} = n_2^2B_{m2} + B_{g2}$$

$$R_{T2} = \frac{R_{a2}}{n_2K_{m2}K_{A2}}$$

$$N_{T2} = \frac{n_2K_{b1}}{K_{A2}}$$

$$T_2 = H_{22}\ddot{q}_2 + H_{23}\ddot{q}_3 + C_2 + d_2$$

$$H_{22} = a_2 + a_3 + a_4 + 2a_5\cos(q_3)$$

$$H_{23} = a_3 + a_5\cos(q_3)$$

$$C_2 = a_2\sin(q_2)\cos(q_2)\dot{q}_1^2 - 2a_5\sin(q_3)\dot{q}_2\dot{q}_3 - a_5\sin(q_3)\dot{q}_3^2$$

$$d_2 = a_6\cos(q_2) + a_7\cos(q_2 + q_3)$$

$$a_2 = I_2 + 2m_2L_2L_{c_{s2}} - m_2L_2^2$$

$$a_3 = I_3 + 2m_3L_3L_{c_{s3}} - m_3L_3^2$$

$$a_4 = m_3L_2^2$$

$$a_5 = m_3L_2L_{c_{s3}}$$

$$a_6 = m_2gL_{c_{s2}} + m_3gL_2$$

$$a_7 = m_3gL_{c_{s3}}$$

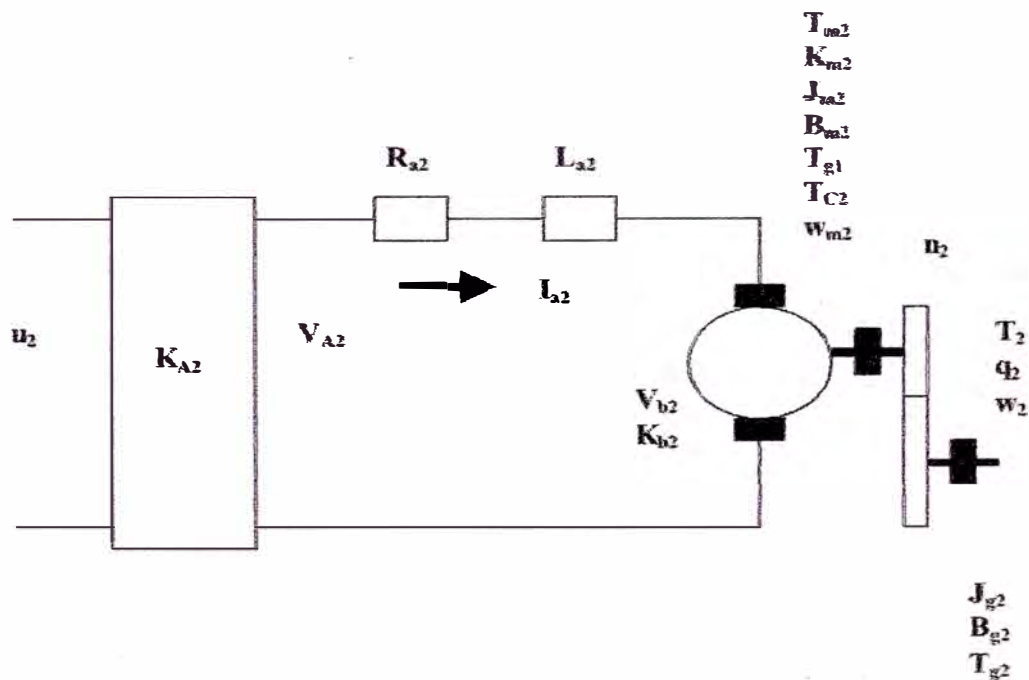


Figura 3.7: Servomotor DC2 de la primera varilla del Manipulador

3.4.3.3 Modelado del Actuador No. 3 de la segunda varilla del Manipulador

En la figura 3.8, se muestra el servomotor DC3 correspondiente a la segunda varilla. En la misma figura se muestra el circuito eléctrico del servomotor DC3, controlado también por la armadura. Dicho servomotor se encuentra ubicado en extremo de la primera varilla. Para obtener la ley de control para la tercera entrada se procede de la misma manera que para el primer servomotor, obteniendo de esta manera:

$$u_3 = R_{T3}(J_{eq3}\ddot{q}_3 + B_{eq3}\dot{q}_3 + T_3) + N_{T3}\dot{q}_3 + n_3 R_{T3} T_{c3} \quad (3.25)$$

donde:

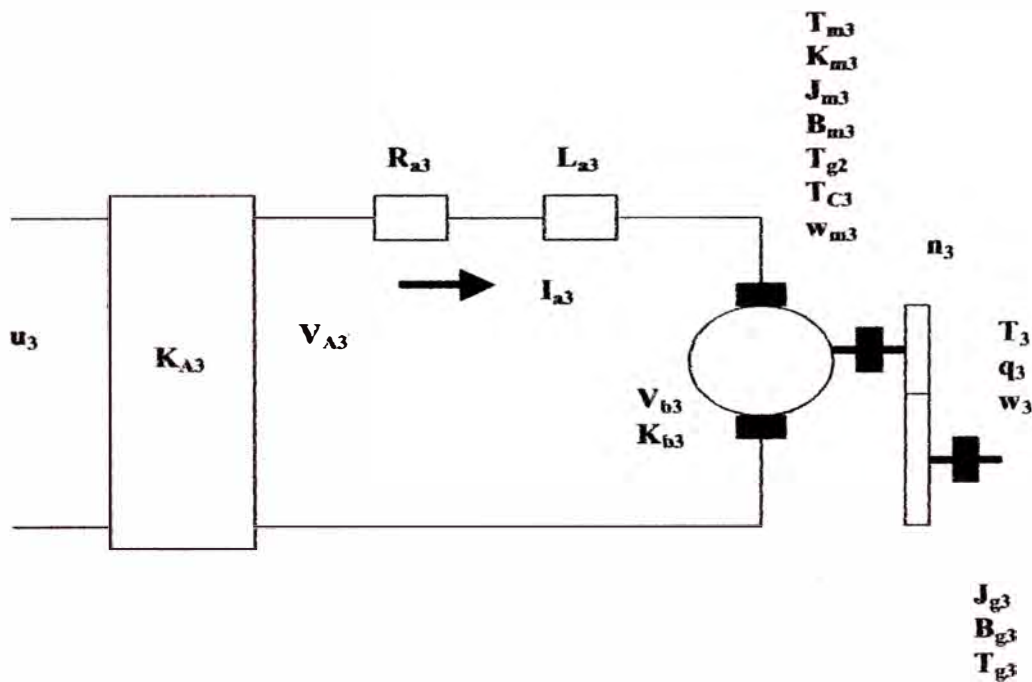


Figura 3.8: Servomotor DC3 de la segunda varilla del Manipulador

$$T_{c3} = C_{c3} \text{sign}(\dot{q}_3)$$

$$J_{eq3} = n_3^2 J_{m3} + J_{g3}$$

$$B_{eq3} = n_3^2 B_{m3} + B_{g3}$$

$$R_{T3} = \frac{R_{a3}}{n_3 K_{m3} K_{A3}}$$

$$N_{T3} = \frac{n_3 K_{b3}}{K_{A3}}$$

$$T_3 = H_{32}\ddot{q}_2 + H_{33}\ddot{q}_3 + C_3 + d_3$$

$$H_{32} = a_3 + a_5 \cos(q_3)$$

$$H_{33} = a_3$$

$$C_3 = (a_3 \sin(q_2 + q_3) \cos(q_2 + q_3) + a_5 \sin(q_2 + q_3) \cos(q_2))\dot{q}_1^2 + a_5 \sin(q_3)\dot{q}_2^2$$

$$d_3 = a_7 \cos(q_2 + q_3)$$

3.4.4 Modelo en el Espacio Estado

La ecuación dinámica del manipulador obtenida en las ecuaciones (3.14), (3.15) y (3.16), se representa en el espacio de estados mediante la siguiente asignación de variables:

$$x_1 = q_1 \tag{3.26}$$

$$x_2 = q_2 \tag{3.27}$$

$$x_3 = q_3 \tag{3.28}$$

$$x_4 = \dot{q}_1 \tag{3.29}$$

$$x_5 = \dot{q}_2 \tag{3.30}$$

$$x_6 = \dot{q}_3 \tag{3.31}$$

donde:

x_1 = posición angular de la base

x_2 = posición angular de la primera varilla

x_3 = posición angular de la segunda varilla

x_4 = velocidad angular de la base

x_5 = velocidad angular de la primera varilla

x_6 = velocidad angular de la segunda varilla

La ecuación no lineal para sistemas multivariables es:

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}) + \mathbf{G}(\mathbf{x})\mathbf{u} \quad (3.32)$$

su representación en forma matricial:

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \\ \dot{x}_5 \\ \dot{x}_6 \end{bmatrix} = \begin{bmatrix} f_1(x) \\ f_2(x) \\ f_3(x) \\ f_4(x) \\ f_5(x) \\ f_6(x) \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ g_{41}(x) & 0 & 0 \\ 0 & g_{52}(x) & g_{53}(x) \\ 0 & g_{62}(x) & g_{63}(x) \end{bmatrix} \begin{bmatrix} u_1(x) \\ u_2(x) \\ u_3(x) \end{bmatrix} \quad (3.33)$$

reemplazando las ecuaciones (3.26) y (3.29) en la ecuación (3.23) se obtiene u_1 en variables de estado:

$$u_1 = R_{T1}(J_{eq1}\dot{x}_4 + B_{eq1}x_4 + T_1) + N_{T1}x_4 + C_{c1}n_1R_{T1}sign(x_4) \quad (3.34)$$

así mismo, reemplazando las ecuaciones (3.27) y (3.30) en la ecuación (3.24) se obtiene u_2 en variables de estado:

$$u_2 = R_{T2}(J_{eq2}\dot{x}_5 + B_{eq2}x_5 + T_2) + N_{T2}x_5 + C_{c1}n_2R_{T2}sign(x_5) \quad (3.35)$$

de la misma manera, reemplazando las ecuaciones (3.28) y (3.31) en la ecuación (3.25) se obtiene u_3 en variables de estado:

$$u_3 = R_{T3}(J_{eq3}\dot{x}_6 + B_{eq3}x_6 + T_3) + N_{T3}x_6 + C_{c3}n_3R_{T3}sign(x_6) \quad (3.36)$$

la ecuación dinámica del manipulador obtenida en las ecuaciones (3.14), (3.15) y (3.16), también se representa en variables de estado:

$$T_1 = H_{11}\dot{x}_4 + C_1 \quad (3.37)$$

$$T_2 = H_{22}\dot{x}_5 + H_{23}\dot{x}_6 + C_2 + d_2 \quad (3.38)$$

$$T_3 = H_{32}\dot{x}_5 + H_{33}\dot{x}_6 + C_3 + d_3 \quad (3.39)$$

posteriormente, se reemplaza la ecuación (3.37) en (3.34) y se obtiene la primera entrada u_1 :

$$u_1 = R_{T1}(J_{eq1}\dot{x}_4 + B_{eq1}x_4 + H_{11}\dot{x}_4 + C_1) + N_{T1}x_4 + C_{c1}n_1R_{T1}sign(x_4) \quad (3.40)$$

posteriormente, se reemplaza la ecuación (3.38) en (3.35) y se obtiene la segunda entrada u_2 :

$$u_2 = R_{T2}(J_{eq2}\dot{x}_5 + B_{eq2}x_5 + H_{22}\dot{x}_5 + H_{23}\dot{x}_6 + C_2 + d_2) + N_{T2}x_5 + C_{c1}n_2R_{T2}sign(x_5) \quad (3.41)$$

finalmente, se reemplaza la ecuación (3.39) en (3.36) y se obtiene la tercera entrada u_3 :

$$u_3 = R_{T3}(J_{eq3}\dot{x}_6 + B_{eq3}x_6 + H_{32}\dot{x}_5 + H_{33}\dot{x}_6 + C_3 + d_3) + N_{T3}x_6 + C_{c3}n_3R_{T3}sign(x_6) \quad (3.42)$$

así mismo de (3.40) se obtiene:

$$\dot{x}_4 = \frac{u_1 - R_{T1}B_{eq1}x_4 - N_{T1}x_4 - R_{T1}C_1 - C_{c1}n_1R_{T1}sign(x_4)}{R_{T1}(J_{eq1} + H_{11})} \quad (3.43)$$

de (3.41) y (3.42) se obtiene \dot{x}_5 y \dot{x}_6 :

$$\dot{x}_5 = \frac{R_{T3}(J_{eq3} + H_{33})u_2 - R_{T2}H_{23}u_3 - R_{T3}(J_{eq3} + H_{33})Q_2 + R_{T2}H_{23}Q_3}{Det} \quad (3.44)$$

$$\dot{x}_6 = \frac{-R_{T3}H_{32}u_2 + R_{T2}(J_{eq2} + H_{22})u_3 + R_{T3}H_{32}Q_2 - R_{T2}(J_{eq2} + H_{22})Q_3}{Det} \quad (3.45)$$

donde:

$$Q_2 = (R_{T2}B_{eq2} + N_{T2} \pm n_2^2 R_{T2}C_{c1})x_5 + R_{T2}(C_2 + d_2)$$

$$Q_3 = (R_{T3}B_{eq3} + N_{T3} \pm n_3^2 R_{T3}C_{c3})x_6 + R_{T3}(C_3 + d_3)$$

$$Det = R_{T2}R_{T3}(J_{eq2} + H_{22})(J_{eq3} + H_{33}) - R_{T2}R_{T3}H_{23}^2$$

finalmente reemplazando las ecuaciones (3.29), (3.30), (3.31), (3.43), (3.44) y (3.45), en la ecuación (3.33) se obtiene:

$$f_1(x) = x_4$$

$$f_2(x) = x_5$$

$$f_3(x) = x_6$$

$$f_4(x) = \frac{-(R_{T1}B_{eq1} + N_{T1} \pm C_{c1}n_1^2 R_{T1})x_4 - R_{T1}C_1}{R_{T1}(J_{eq1} + H_{11})}$$

$$f_5(x) = \frac{-R_{T3}(J_{eq3} + H_{33})Q_2 + R_{T2}H_{23}Q_3}{Det}$$

$$f_6(x) = \frac{R_{T3}H_{32}Q_2 - R_{T2}(J_{eq2} + H_{22})Q_3}{Det}$$

$$g_{41}(x) = \frac{1}{R_{T1}(J_{eq1} + H_{11})}$$

$$g_{52}(x) = \frac{R_{T3}(J_{eq3} + H_{33})}{Det}$$

$$g_{53}(x) = \frac{-R_{T2}H_{23}}{Det}$$

$$g_{62}(x) = \frac{-R_{T2}H_{32}}{Det}$$

$$g_{63}(x) = \frac{R_{T2}(J_{eq2} + H_{22})}{Det}$$

como la planta tiene tres salidas se tiene:

$$y = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \end{bmatrix} \quad (3.46)$$

3.4.5 Comportamiento del Manipulador [2],[3]

Para desarrollar la simulación del comportamiento del manipulador de 3DOF se usará el Robotics Toolbox para MATLAB versión 6. Este Toolbox está basado en los parámetros de Denavit-Hartenberg y los métodos generales que representan la cinemática y dinámica de los manipuladores robóticos. En el toolbox incluye ejemplos de robots conocidos como el Puma 560 y el Stanford, que es la base para ejecutar la animación de cualquier otro manipulador. La descripción del manipulador puede ser elaborado por la matriz aumentada, la inercia de los eslabones, la inercia del servomotor y los parámetros de fricción. Tal matriz provee un significado conciso para describir y facilitar la investigación del modelo del manipulador, lo cual permitirá simular los resultados. El toolbox también provee funciones para manipular datos como vectores y transformaciones homogéneas los cuales son necesarios para representar la posición y orientación en 3 dimensiones.

3.4.5.1 Animación del Manipulador

La animación del manipulador trabaja con los comandos establecidos en el Robotics Toolbox para MATLAB [3]. Para ejecutar la animación del manipulador de 3DOF se necesitan de 2 archivos, los cuales han sido desarrollados en MATLAB 6.0, teniendo como base el robot Puma 560.

- El programa *Robot_3dof.m*, contiene los parámetros Denavit-Hartenberg, inercias y masas del manipulador, además de los valores de los momentos de inercia, coeficiente de fricción viscosa y fricción de Coulomb de los servomotores correspondientes a cada grado de libertad.
- El programa *anima_3dof.m*, contiene en su menú la animación del manipulador y otras funciones como por ejemplo cinemática directa, cinemática inversa, dinámica directa, dinámica inversa, etc

Los programas descritos se encuentran en el CD-ROM adjunto a este trabajo de tesis.

3.4.5.2 Uso del Toolbox para la animación

Al correr el programa *anima_3dof.m* se obtiene las figuras 3.9 y 3.10:

Al pulsar en la figura 3.10, el "Menú de Funciones del Robot de 3DOF" el pushbutton ANIMACION, se obtiene en la figura 3.9 la animación del robot de 3 DOF, el cual realiza el movimiento simultáneo de cada DOF. Cabe indicar que el movimiento que realiza el manipulador es de acuerdo a la posición (radianes) que se le da en el programa *Robot_3dof*, y no lo ejecuta con el algoritmo de control no lineal empleado en el trabajo de tesis.

Al pulsar el pushbutton TRAYECTORIA, el manipulador se moverá desde un ángulo cero hasta la posición dada (q_r), graficando una trayectoria polinómica para la posición velocidad, aceleración, que está dada por defecto usando la función "jtraj()", para cada grado de libertad.

Al pulsar el pushbutton TRANSFORMACIONES, se obtiene las translaciones usando la función "transl(0.5,0,0)" y rotaciones usando la función "roty(pi/2)" y "rotz(-pi/2)", en dirección de los ejes coordenados X, Y o Z, así mismo se realiza multiplicaciones entre translaciones y rotaciones. La orientación también se expresa en ángulos de Euler mediante la función "tr2eul(t)".

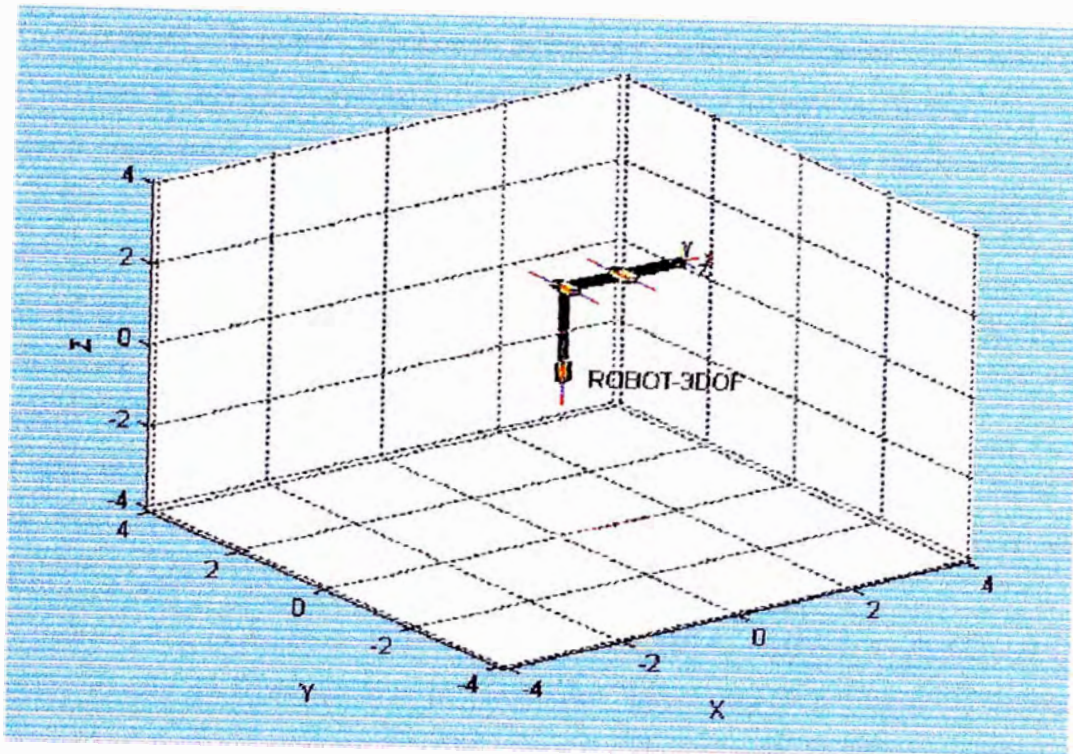


Figura 3.9: Robot de 3DOF para la animación

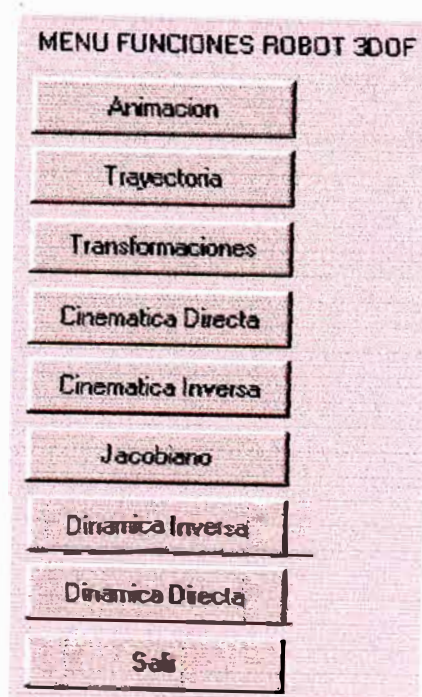


Figura 3.10: Menú de funciones del Robot de 3DOF

Al pulsar el pushbutton CINEMATICA DIRECTA, calcula la cinemática usando la función "fkine()" y retorna la matriz de transformación homogénea (T) correspondiente por cada conjunto de coordenadas. Usando la función "jtraj()" y la función "squeeze", se gráfica la trayectoria del manipulador en las coordenadas X, Y, Z vs. el tiempo; así mismo se puede graficar la trayectoria del manipulador en la coordenada X vs. la coordenada Z.

Al pulsar el pushbutton CINEMATICA INVERSA, primeramente genera la transformada homogénea (T) con la función "fkine()", para generar después con la función "ikine()" la cinemática inversa.

Al pulsar el pushbutton JACOBIANO, calcula la velocidad en la base con la función "jacob0()" y la velocidad en el extremo del manipulador con la función "jacobn()" a partir de las posiciones (q) de los eslabones.

Al pulsar el pushbutton DINAMICA INVERSA, calcula y grafica el torque para cada grado de libertad con la ecuación "tau=rne()" a partir de la posición (q), velocidad (\dot{q}) y aceleración (\ddot{q}), donde "rne" es la función que calcula el algoritmo recursivo de Newton-Euler. De igual manera se grafica los componentes que conforman el torque como la matriz de inercia usando la función "inertia()" y la fuerza de gravedad usando la función "gravload()".

Al pulsar el pushbutton DINAMICA DIRECTA, grafica y simula el movimiento del manipulador usando la ecuación " $[t \quad q \quad qd] = fdyn()$ ", donde la función "fdyn" calcula la ecuación dinámica directa del manipulador.

Capítulo 4

Diseño Controlador No Lineal

4.1 Herramientas de Geometría Diferencial

Los métodos de geometría diferencial son principalmente aplicados a procesos no lineales para determinar si tales procesos son linealizables por realimentación de estado; esto es si la descripción dinámica de un proceso en particular puede ser linealizada después de una transformación adecuada de coordenadas y la realimentación del vector de estado al proceso no lineal. De acuerdo a la terminología usada en la geometría diferencial, llamaremos a una función vectorial n dimensional:

$$\mathbf{f}(\mathbf{x}) = \begin{bmatrix} f_1(x) \\ \vdots \\ f_n(x) \end{bmatrix} = \begin{bmatrix} f_1(x_1, \dots, x_n) \\ \vdots \\ f_n(x_1, \dots, x_n) \end{bmatrix} \quad (4.1)$$

un campo vectorial, si para cada función vectorial \mathbf{f} le corresponde un campo de vectores en un espacio n dimensional \mathbf{R}^n . Posteriormente, solo trataremos con campo vectoriales suaves de la forma $\mathbf{f}(\mathbf{x})$, lo que significa que tales vectores poseen derivadas parciales continuas de cualquier orden. Suavidad también se aplica a funciones.

El gradiente de una función escalar suave $h(\mathbf{x})$ es definida como:

$$\nabla h(\mathbf{x}) = \frac{\partial h}{\partial \mathbf{x}} = \left[\frac{\partial h(x)}{\partial x_1} \quad \dots \quad \frac{\partial h(x)}{\partial x_n} \right] \quad (4.2)$$

Los elementos de ∇h , son:

$$(\nabla h)_i = \frac{\partial h}{\partial x_i}$$

Similarmente, el Jacobiano de $\mathbf{f}(\mathbf{x})$, de elementos: $(\nabla \mathbf{f})_{ij} = \partial f_i / \partial x_j$, es definido como:

$$\nabla \mathbf{f}(\mathbf{x}) = \frac{\partial \mathbf{f}(\mathbf{x})}{\partial \mathbf{x}} = \begin{bmatrix} \frac{\partial f_1(x)}{\partial x_1} & \frac{\partial f_1(x)}{\partial x_2} & \dots & \frac{\partial f_1(x)}{\partial x_n} \\ \frac{\partial f_2(x)}{\partial x_1} & \frac{\partial f_2(x)}{\partial x_2} & \dots & \frac{\partial f_2(x)}{\partial x_n} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{\partial f_n(x)}{\partial x_1} & \frac{\partial f_n(x)}{\partial x_2} & \dots & \frac{\partial f_n(x)}{\partial x_n} \end{bmatrix} \quad (4.3)$$

4.1.1 Derivada de Lie y Corchete de Lie [13], [14]

Derivada de Lie

La derivada de Lie de una función escalar $h(\mathbf{x})$ con respecto a un campo vectorial $\mathbf{f}(\mathbf{x})$ es una nueva función escalar $L_{\mathbf{f}}h$ definida como:

$$L_{\mathbf{f}}h = \nabla h \mathbf{f} = \frac{\partial h}{\partial x_1} f_1 + \dots + \frac{\partial h}{\partial x_n} f_n \quad (4.4)$$

Obsérvese que la derivada de Lie es el producto interno entre $\nabla h(\mathbf{x})$ y $\mathbf{f}(\mathbf{x})$. Las derivadas de Lie repetidas pueden ser formuladas en forma recursiva:

$$L_{\mathbf{f}}^0 h = h$$

$$L_{\mathbf{f}}^i h = L_{\mathbf{f}}(L_{\mathbf{f}}^{i-1} h) = \nabla(L_{\mathbf{f}}^{i-1} h) \mathbf{f}, \quad \text{para } i=1,2,\dots \quad (4.5)$$

Si \mathbf{f} y \mathbf{g} son campos vectoriales, luego:

$$L_{\mathbf{g}}L_{\mathbf{f}} = \nabla(L_{\mathbf{f}})\mathbf{g} \quad (4.6)$$

Corchete de Lie

El corchete de Lie de dos campos vectoriales \mathbf{f} y \mathbf{g} es otro campo vectorial $[\mathbf{f}, \mathbf{g}]$, definido como:

$$[\mathbf{f}, \mathbf{g}](x) = \nabla \mathbf{g} \mathbf{f} - \nabla \mathbf{f} \mathbf{g} \quad (4.7)$$

La notación $ad_{\mathbf{f}}\mathbf{g}$ es frecuentemente usado en vez de $[\mathbf{f}, \mathbf{g}]$. Los corchetes de Lie repetidos pueden ser formulados en forma recursiva:

$$ad_{\mathbf{f}}^0 \mathbf{g} = \mathbf{g}$$

$$ad_{\mathbf{f}}^i \mathbf{g} = [\mathbf{f}, ad_{\mathbf{f}}^{i-1} \mathbf{g}], \quad \text{para } i=1,2,\dots \quad (4.8)$$

Si \mathbf{f} , \mathbf{f}_1 , \mathbf{f}_2 , \mathbf{g} , \mathbf{g}_1 , \mathbf{g}_2 , son campo vectoriales suaves, α_1 y α_2 son constantes escalares y $h(\mathbf{x})$ es una función escalar suave de \mathbf{x} , podemos enunciar las siguientes propiedades del corchete de Lie:

1) Bilinealidad:

$$[\alpha_1 \mathbf{f}_1 + \alpha_2 \mathbf{f}_2, \mathbf{g}] = \alpha_1 [\mathbf{f}_1, \mathbf{g}] + \alpha_2 [\mathbf{f}_2, \mathbf{g}]$$

$$[\mathbf{f}, \alpha_1 \mathbf{g}_1 + \alpha_2 \mathbf{g}_2] = \alpha_1 [\mathbf{f}, \mathbf{g}_1] + \alpha_2 [\mathbf{f}, \mathbf{g}_2] \quad (4.9)$$

2) Anticonmutatividad:

$$[\mathbf{f}, \mathbf{g}] = -[\mathbf{g}, \mathbf{f}] \quad (4.10)$$

3) Identidad de Jacobi:

$$L_{ad_{\mathbf{f}}}\mathbf{g}h = L_{\mathbf{f}}L_{\mathbf{g}}h = L_{\mathbf{g}}L_{\mathbf{f}}h \quad (4.11)$$

4.1.2 Difeomorfismo y Transformación de Coordenadas [13], [14]

Una función Φ definida en una región Ω es conocida como un difeomorfismo si ambos, Φ y su inversa Φ^{-1} son funciones suaves. Si la región Ω es igual a \mathbf{R}^n (el espacio total), entonces $\Phi(\mathbf{x})$ es un difeomorfismo global. Por otro lado un difeomorfismo local es definido solo en un campo finito de un punto dado \mathbf{x}_o .

Una función suave $\Phi(\mathbf{x})$ definida en una región Ω en \mathbf{R}^n es un difeomorfismo local si la matriz Jacobiana $\nabla \mathbf{x}$ es no singular en un punto $\mathbf{x} = \mathbf{x}_o$ de la región Ω .

Empleando un difeomorfismo, podemos transformar los estados de un proceso no lineal dentro de un nuevo conjunto de estados correspondientes a la transformación de un proceso no lineal. Consideremos un proceso no lineal SISO (single-input-single-output) descrito por:

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}) + \mathbf{g}(\mathbf{x})u, \quad \mathbf{y} = h(\mathbf{x}) \quad (4.12)$$

donde la entrada simple es u y la salida simple es y . Definiremos ahora un nuevo conjunto de estados $\mathbf{z} = \Phi(\mathbf{x})$. Ejecutando la diferenciación en \mathbf{z} :

$$\dot{\mathbf{z}} = \frac{\partial \Phi}{\partial \mathbf{x}} \dot{\mathbf{x}} = \frac{\partial \Phi}{\partial \mathbf{x}} (\mathbf{f}(\mathbf{x}) + \mathbf{g}(\mathbf{x})u) = \mathbf{f}^*(\mathbf{z}) + \mathbf{g}^*(\mathbf{z})u, \quad \mathbf{y} = h^*(\mathbf{z}) \quad (4.13)$$

donde:

$$\mathbf{f}^* = \frac{\partial \Phi}{\partial \mathbf{x}} \mathbf{f}$$

$$\mathbf{g}^* = \frac{\partial \Phi}{\partial \mathbf{x}} \mathbf{g}$$

$$\mathbf{x} = \Phi^{-1}(\mathbf{z})$$

$$h^* = h$$

4.1.3 El Teorema de Frobenius [13], [14]

Sea $\mathbf{f}_1, \mathbf{f}_2, \dots, \mathbf{f}_m$ un conjunto de campos vectoriales linealmente independientes en \mathbf{R}^n . El Teorema de Frobenius establece que tal conjunto es completamente integrable sí y sólo sí es involutivo, según se explica posteriormente.

El conjunto de campos vectoriales definidos anteriormente completamente integrable sí y sólo sí existen $(n - m)$ funciones escalares $h_1(\mathbf{x}), h_2(\mathbf{x}), \dots, h_{n-m}(\mathbf{x})$, que satisfacen el siguiente sistema de ecuaciones diferenciales parciales:

$$\nabla h_i \mathbf{f}_j = 0, \quad 1 \leq i \leq n - m, \quad 1 \leq j \leq m \quad (4.14)$$

donde los gradientes $\nabla h_1, \nabla h_2, \dots, \nabla h_{n-m}$, son también linealmente independientes.

El conjunto de campos vectoriales considerado es llamado a ser involutivo sí y sólo si hay funciones escalares α_{ijk} en \mathbf{R} (el espacio unidimensional) satisfaciendo:

$$[\mathbf{f}_i, \mathbf{f}_j](\mathbf{x}) = \sum_{k=1}^m \alpha_{ijk}(\mathbf{x}) \mathbf{f}_k(\mathbf{x}), \quad \forall i, j \quad (4.15)$$

La última ecuación indica que el campo vectorial resultante $[\mathbf{f}_i, \mathbf{f}_j](\mathbf{x})$ es una combinación lineal del conjunto original de campos vectoriales $\mathbf{f}_1, \mathbf{f}_2, \dots, \mathbf{f}_m$. De (3.15), tenemos los siguientes postulados:

- 1) Los campos vectoriales constantes son siempre involutivos porque el corchete de Lie para dos vectores \mathbf{f} y \mathbf{g} resulta $[\mathbf{f}, \mathbf{g}] = 0$, el cual puede ser representado como una combinación lineal de los campos vectoriales $\mathbf{f}_1, \mathbf{f}_2, \dots, \mathbf{f}_m$.
- 2) Un conjunto compuesto por un sólo campo vectorial \mathbf{f} es involutivo porque $[\mathbf{f}, \mathbf{f}] = 0$.
- 3) El conjunto de los campos vectoriales $\mathbf{f}_1, \mathbf{f}_2, \dots, \mathbf{f}_m$ es involutivo si $\forall \mathbf{x}$ y $\forall i, j$:

$$\text{rango}[\mathbf{f}_1(\mathbf{x}) \dots \mathbf{f}_m(\mathbf{x})] = \text{rango}[\mathbf{f}_1(\mathbf{x}) \dots \mathbf{f}_m(\mathbf{x}), [\mathbf{f}_i(\mathbf{x}), \mathbf{f}_j(\mathbf{x})]]$$

donde la notación $\text{rango} [.]$ representa el rango de la matriz formada por la columna de los campos vectoriales.

4.2 Linealización por Realimentación para Procesos SISO [7], [12], [13], [14]

Esta sección presenta la metodología para diseñar controladores estables basados en la linealización entrada-salida de procesos SISO no lineales, los cuales pueden ser descritos por:

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}) + \mathbf{g}(\mathbf{x})u, \quad y = h(\mathbf{x}) \quad (4.16)$$

$$\mathbf{x} = \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix} \quad \mathbf{f} = \begin{bmatrix} f_1(\mathbf{x}) \\ \vdots \\ f_n(\mathbf{x}) \end{bmatrix} \quad \mathbf{g} = \begin{bmatrix} g_1 \\ \vdots \\ g_n \end{bmatrix}$$

donde \mathbf{x} es el vector de estado de orden $n \times 1$, u es la señal de control o entrada, y es la salida del proceso, \mathbf{f} y \mathbf{g} son campos vectoriales suaves de orden $n \times 1$ y h es una función suave que describe la salida.

4.2.1 Condiciones para Linealización

El proceso no lineal descrito en la ecuación (3.16) es linealizable sí y sólo si las siguientes condiciones son verdaderas para una región Ω :

- 1) Los campos vectoriales \mathbf{g} , $ad_{\mathbf{f}}\mathbf{g}$, ..., $ad_{\mathbf{f}}^{m-1}\mathbf{g}$, son linealmente independientes en Ω
- 2) El conjunto \mathbf{g} , $ad_{\mathbf{f}}\mathbf{g}$, ..., $ad_{\mathbf{f}}^{m-2}\mathbf{g}$ es involutivo en Ω

Para procesos lineales SISO:

$$\mathbf{f}(\mathbf{x}) = \mathbf{A}\mathbf{x}$$

$$\mathbf{g}(\mathbf{x}) = \mathbf{B}$$

donde \mathbf{A} de orden $n \times n$ y \mathbf{B} de orden $n \times 1$ son las matrices de estado y control (o distribución) respectivamente. La primera condición arriba mencionada, puede ser interpretada como la condición de controlabilidad de un proceso lineal de la forma:

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}u, \quad y = \mathbf{C}\mathbf{x} \quad (4.17)$$

El sistema (3.17) es controlable si su matriz de controlabilidad \mathcal{C} posee rango total; es decir:

$$\text{rango } \mathcal{C} = \text{rango}[\mathbf{B} \ \mathbf{A}\mathbf{B} \ \dots \ \mathbf{A}^{n-1}\mathbf{B}] = n \quad (4.18)$$

Por consiguiente la primera condición descrita arriba puede ser interpretada como la *Condición de Controlabilidad Generalizada* para procesos no lineales.

4.2.2 Grado Relativo del Proceso

Diferenciando la salida y de (3.16):

$$\dot{y} = \nabla h(\mathbf{x})(\mathbf{f} + \mathbf{g}u) = L_{\mathbf{f}}h(\mathbf{x}) + L_{\mathbf{g}}h(\mathbf{x})u$$

Si $L_{\mathbf{g}}h(\mathbf{x}) \neq 0$ para algún $\mathbf{x} = \mathbf{x}_0$, entonces la siguiente transformación de entrada:

$$u = \frac{1}{L_{\mathbf{g}}h(\mathbf{x})}(-L_{\mathbf{f}}h(\mathbf{x}) + v)$$

genera una relación diferencial lineal entre la salida y y la nueva entrada v de la forma:

$$\dot{y} = v$$

Caso contrario si $L_{\mathbf{g}}h(\mathbf{x}) = 0$ para todo \mathbf{x} , la diferenciación de \dot{y} resulta:

$$\ddot{y} = L_{\mathbf{f}}^2h(\mathbf{x}) + L_{\mathbf{g}}L_{\mathbf{f}}h(\mathbf{x})u$$

Si $L_{\mathbf{g}}L_{\mathbf{f}}h(\mathbf{x}) = 0$ para todo \mathbf{x} , necesitamos diferenciar \ddot{y} y así sucesivamente $y^{(3)}$, $y^{(4)}$, ..., $y^{(i)}$, como sigue:

$$y^{(i)} = \frac{d^i y}{dt^i} = L_{\mathbf{f}}^i h(\mathbf{x}) + L_{\mathbf{g}}L_{\mathbf{f}}^{i-1}h(\mathbf{x})u$$

hasta que para algún entero $r = i$ y para algún $\mathbf{x} = \mathbf{x}_0$:

$$L_{\mathbf{g}}L_{\mathbf{f}}^{r-1}h(\mathbf{x}) \neq 0$$

En esta situación, sustituyendo la ley de control:

$$u = \frac{1}{L_{\mathbf{g}}L_{\mathbf{f}}^{r-1}h(\mathbf{x})}(-L_{\mathbf{f}}^r h(\mathbf{x}) + v)$$

en:

$$y^{(r)} = L_{\mathbf{f}}^r h(\mathbf{x}) + L_{\mathbf{g}}L_{\mathbf{f}}^{r-1}h(\mathbf{x})u$$

genera una relación diferencial lineal entre la salida y y la nueva entrada v de la forma:

$$y^{(r)} = \frac{d^r y}{dt^r} = v$$

Por lo tanto, un proceso **SISO** posee un grado relativo r en un punto x_0 de una región Ω , si para todo $\mathbf{x} \in \Omega$:

$$L_{\mathbf{g}}L_{\mathbf{f}}^i h(\mathbf{x}) = 0, \quad 0 \leq i < r - 1 \quad (4.19)$$

$$L_{\mathbf{g}}L_{\mathbf{f}}^{r-1} h(\mathbf{x}) \neq 0 \quad (4.20)$$

Observe que r es el número de diferenciaciones ejecutadas en la salida y . En general $r \leq n$ (donde n es el orden del proceso). Si $r = n$, la linealización entrada-salida se denomina exacta.

4.2.3 Forma Normal SISO para Linealización Exacta

El proceso no lineal (4.12) puede ser transformado a una **forma normal** usando los resultados establecidos en la subsección 4.1.2. La linealización exacta de un proceso no lineal descrito por (4.12) ocurre cuando el grado relativo r es igual al orden n del proceso. Para linealización exacta, la transformación de estado:

$$\mathbf{z} = \begin{bmatrix} z_1 \\ z_2 \\ \vdots \\ z_n \end{bmatrix} = \Phi(\mathbf{x}) = \begin{bmatrix} \phi_1(\mathbf{x}) \\ \phi_2(\mathbf{x}) \\ \vdots \\ \phi_n(\mathbf{x}) \end{bmatrix} = \begin{bmatrix} h(\mathbf{x}) \\ L_{\mathbf{f}}h(\mathbf{x}) \\ \vdots \\ L_{\mathbf{f}}^{n-1}h(\mathbf{x}) \end{bmatrix} = \begin{bmatrix} y \\ \dot{y} \\ \vdots \\ y^{(n)} \end{bmatrix} \quad (4.21)$$

origina la siguiente ecuación de estado:

$$\dot{\mathbf{z}} = \begin{bmatrix} \dot{z}_1 \\ \dot{z}_2 \\ \vdots \\ \dot{z}_{n-1} \\ \dot{z}_n \end{bmatrix} = \begin{bmatrix} z_2 \\ z_3 \\ \vdots \\ z_n \\ b(\mathbf{z}) + a(\mathbf{z})u \end{bmatrix} \quad y = z_1 \quad (4.22)$$

donde:

$$a(\mathbf{z}) = L_{\mathbf{g}}L_{\mathbf{f}}^{n-1}h(\Phi^{-1}(\mathbf{z}))$$

$$b(\mathbf{z}) = L_{\mathbf{f}}^n h(\Phi^{-1}(\mathbf{z})) \quad (4.23)$$

4.2.4 Ley de Control SISO para Linealización Exacta

Consideremos la siguiente ley de control linealizada:

$$u = \frac{1}{a(\mathbf{z})} [-b(\mathbf{z}) + v] \quad (4.24)$$

donde v es una nueva entrada a ser determinada, $a(\mathbf{z})$ y $b(\mathbf{z})$ son dadas por (4.23), y \mathbf{z} es el estado linealizado. Sustituyendo (4.24) en (4.22), llegamos a la representación canónica lineal invariante en el tiempo:

$$\dot{\mathbf{z}} = \mathbf{M}\mathbf{z} + \mathbf{N}v, \quad y = \mathbf{C}\mathbf{z} \quad (4.25)$$

donde:

$$\mathbf{M} = \begin{bmatrix} 0 & 1 & 0 & \dots & 0 \\ 0 & 0 & 1 & \dots & \cdot \\ \cdot & \cdot & \cdot & \dots & \cdot \\ 0 & 0 & 0 & \dots & 1 \\ 0 & 0 & 0 & \dots & 0 \end{bmatrix} \quad \mathbf{N} = \begin{bmatrix} 0 \\ 0 \\ \cdot \\ \bullet \\ 1 \end{bmatrix} \quad \mathbf{C} = [1 \ 0 \ \dots \ 0 \ 0]$$

La función de transferencia del proceso linealizado es:

$$\frac{y(s)}{v(s)} = H(s) = \mathbf{C}[s\mathbf{I} - \mathbf{M}]^{-1}\mathbf{N} = \frac{1}{s^n} \quad (4.26)$$

donde s es el operador de Laplace. La entrada v puede ser diseñada empleando la técnica por ubicación de polos en el sistema lineal equivalente dado en (4.25). Luego, la entrada real u se calcula usando (4.24). Empleando (4.26), el sistema lineal equivalente se puede describir como:

$$v = \frac{d^n y}{dt^n} = y^{(n)} = p^{(n)} + K_1 \bar{y}^{(n-1)} + \dots + K_{n-1} \dot{\bar{y}} + K_n \bar{y} \quad (4.27)$$

donde $p(t)$ es la trayectoria deseada e $\bar{y}(t) = p(t) - y(t)$ es la señal de error de seguimiento. La ecuación característica del sistema lineal equivalente es hallada de:

$$\bar{y}^{(n)} + K_1 \bar{y}^{(n-1)} + \dots + K_{n-1} \dot{\bar{y}} + K_n \bar{y} = 0 \quad (4.28)$$

Para un comportamiento asintóticamente estable, con $\bar{y} = (p - y) \rightarrow 0$ para $t \rightarrow \infty$, ninguno de las raíces de (4.28) debe ser localizado en el lado derecho del plano s , incluyendo el eje imaginario. Los parámetros del controlador K_i ,

$i = 1, \dots, n$ son seleccionados para encontrar tales requerimientos. La entrada v toma la forma (tener presente que $p^{(1)} \equiv dp/dt$ y $p^{(0)} \equiv p$):

$$v = p^{(n)} + \sum_{i=1}^n K_i p^{(n-1)} - \mathbf{Kz} \quad (4.29)$$

donde:

$$\mathbf{K} = \begin{bmatrix} K_1 & \dots & K_n \end{bmatrix}, \quad \mathbf{z} = \begin{bmatrix} z_n \\ z_{n-1} \\ \vdots \\ z_1 \end{bmatrix} = \begin{bmatrix} y^{(n-1)} \\ y^{(n-2)} \\ \vdots \\ y \end{bmatrix}$$

4.3 Linealización por Realimentación para Procesos MIMO [13], [14]

4.3.1 Modelando Procesos Cuadrados No Lineales

Un proceso no lineal MIMO, es llamado cuadrado cuando el número de entradas del proceso es igual al número de salidas del proceso. El proceso cuadrado a ser considerado se describe como:

$$\dot{\mathbf{x}} = \mathbf{f}(\mathbf{x}) + \mathbf{G}(\mathbf{x})\mathbf{u} = \mathbf{f}(\mathbf{x}) + \sum_{j=1}^m \mathbf{g}_j(\mathbf{x})u_j(\mathbf{x}) \quad \mathbf{y} = \mathbf{h}(\mathbf{x}) \quad (4.30)$$

donde:

$$\mathbf{x} = \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix} \quad \mathbf{f} = \begin{bmatrix} f_1(\mathbf{x}) \\ \vdots \\ f_n(\mathbf{x}) \end{bmatrix} \quad \mathbf{u} = \begin{bmatrix} u_1 \\ \vdots \\ u_m \end{bmatrix} \quad \mathbf{y} = \begin{bmatrix} y_1 \\ \vdots \\ y_m \end{bmatrix}$$

$$\mathbf{h} = \begin{bmatrix} h_1(\mathbf{x}) \\ \vdots \\ h_m(\mathbf{x}) \end{bmatrix} \quad \mathbf{G} = \begin{bmatrix} \mathbf{g}_1(\mathbf{x}) & \dots & \mathbf{g}_m(\mathbf{x}) \end{bmatrix}$$

donde \mathbf{x} es el vector de estado de orden $n \times 1$, \mathbf{u} es el vector de control de entrada (de componentes u_i) de orden $m \times 1$, \mathbf{y} es la salida del vector (de componentes y_i) de orden $m \times 1$, \mathbf{f} es un campo vectorial suave de orden $n \times 1$, \mathbf{h} es un campo vectorial suave de orden $m \times 1$, y \mathbf{G} es una matriz $n \times m$ cuyas columnas son los campos vectoriales \mathbf{g}_j de orden $n \times 1$. Los componentes de \mathbf{g} y \mathbf{h} son funciones suaves.

4.3.2 Grado Relativo Total

El proceso cuadrado descrito por (4.30) posee un *grado relativo total* $r = r_1 + \dots + r_m$, [12], [13], [14] sí:

1) Para todo \mathbf{x} alrededor de \mathbf{x}_o :

$$L_{\mathbf{g}_j} L_{\mathbf{f}}^k h_i(\mathbf{x}) = 0 \quad j, i = 1, \dots, m \quad k < r_i - 1$$

2) La siguiente matriz $m \times m$ es no singular en $\mathbf{x} = \mathbf{x}_o$:

$$A(\mathbf{x}) = \begin{bmatrix} L_{\mathbf{g}_1} L_{\mathbf{f}}^{r_1-1} h_1(\mathbf{x}) & \dots & L_{\mathbf{g}_m} L_{\mathbf{f}}^{r_1-1} h_1(\mathbf{x}) \\ L_{\mathbf{g}_1} L_{\mathbf{f}}^{r_2-1} h_2(\mathbf{x}) & \dots & L_{\mathbf{g}_m} L_{\mathbf{f}}^{r_2-1} h_2(\mathbf{x}) \\ \vdots & \ddots & \vdots \\ L_{\mathbf{g}_1} L_{\mathbf{f}}^{r_m-1} h_m(\mathbf{x}) & \dots & L_{\mathbf{g}_m} L_{\mathbf{f}}^{r_m-1} h_m(\mathbf{x}) \end{bmatrix} \quad (4.31)$$

Observando las filas de A , podemos decir que cada entero r_i está relacionado con la i -ésima salida del proceso. También notamos que r_i es el mínimo número de diferenciaciones ejecutadas en la salida y_i , requeridas para que las componentes de u no desaparezcan. La no singularidad de $A(\mathbf{x})$ en $\mathbf{x} = \mathbf{x}_o$ es la versión multivariable de la condición impuesta por (4.20).

4.3.3 Forma Normal para Linealización Exacta MIMO

Si el grado relativo total de un proceso cuadrado descrito por (4.30) es r , luego:

$$r = r_1 + r_2 + \dots + r_m \leq n$$

La linealización exacta de un proceso no lineal **MIMO** descrito por (4.30), ocurre cuando el grado relativo total : $r = r_1 + \dots + r_m$, es igual al orden n del proceso; es decir, la dimensión n del vector de estado. Para linealización exacta, se tiene la siguiente transformación de estado \mathbf{z} :

$$z = \Phi(\mathbf{x}) = \begin{bmatrix} \phi_1(\mathbf{x}) \\ \vdots \\ \phi_{r_1}(\mathbf{x}) \\ \psi_1(\mathbf{x}) \\ \vdots \\ \psi_{r_2}(\mathbf{x}) \\ \xi_1(\mathbf{x}) \\ \vdots \\ \xi_{r_m}(\mathbf{x}) \end{bmatrix} = \begin{bmatrix} z_1 \\ \vdots \\ z_{r_1} \\ z_{r_1+1} \\ \vdots \\ z_{r_1+r_2} \\ z_{r_1+r_2+\dots+1} \\ \vdots \\ z_{r_1+r_2+\dots+r_m} \end{bmatrix} \quad (4.32)$$

donde:

$$\begin{aligned} \phi_1(\mathbf{x}) &= h_1(\mathbf{x}) = y_1(\mathbf{x}); \quad \phi_2(\mathbf{x}) = L_f h_1(\mathbf{x}); \quad \dots; \quad \phi_{r_1}(\mathbf{x}) = L_f^{r_1-1} h_1(\mathbf{x}) \\ \psi_1(\mathbf{x}) &= h_2(\mathbf{x}) = y_2(\mathbf{x}); \quad \psi_2(\mathbf{x}) = L_f h_2(\mathbf{x}); \quad \dots; \quad \psi_{r_2}(\mathbf{x}) = L_f^{r_2-1} h_2(\mathbf{x}) \\ &\vdots \\ \xi_1(\mathbf{x}) &= h_m(\mathbf{x}) = y_m(\mathbf{x}); \quad \xi_2(\mathbf{x}) = L_f h_m(\mathbf{x}); \quad \dots; \quad \xi_{r_m}(\mathbf{x}) = L_f^{r_m-1} h_m(\mathbf{x}) \end{aligned}$$

da lugar a las siguientes ecuaciones de estado:

$$\begin{aligned} \dot{y}_1 &= \dot{\phi}_1(\mathbf{x}) = \phi_2(\mathbf{x}) \\ &\vdots \\ y_1^{(r_1-1)} &= \dot{\phi}_{r_1-1}(\mathbf{x}) = \phi_{r_1}(\mathbf{x}) \\ y_1^{(r_1)} &= \dot{\phi}_{r_1}(\mathbf{x}) = L_f^{r_1} h_1(\Phi^{-1}(\mathbf{z})) + \sum_{j=1}^m L_{\mathbf{g}_j} L_f^{r_1-1} h_1(\Phi^{-1}(\mathbf{z})) u_j \\ \dot{y}_2 &= \dot{\psi}_1(\mathbf{x}) = \psi_2(\mathbf{x}) \\ &\vdots \\ y_2^{(r_2-1)} &= \dot{\psi}_{r_2-1}(\mathbf{x}) = \psi_{r_2}(\mathbf{x}) \\ y_2^{(r_2)} &= \dot{\psi}_{r_2}(\mathbf{x}) = L_f^{r_2} h_2(\Phi^{-1}(\mathbf{z})) + \sum_{j=1}^m L_{\mathbf{g}_j} L_f^{r_2-1} h_2(\Phi^{-1}(\mathbf{z})) u_j \\ \dot{y}_m &= \dot{\xi}_1(\mathbf{x}) = \xi_2(\mathbf{x}) \end{aligned}$$

⋮

$$y_m^{(r_m-1)} = \dot{\xi}_{r_m-1}(\mathbf{x}) = \xi_{r_m}(\mathbf{x})$$

$$y_m^{(r_m)} = \dot{\xi}_{r_m}(\mathbf{x}) = L_f^{r_m} h_m(\Phi^{-1}(\mathbf{z})) + \sum_{j=1}^m L_{g_j} L_f^{r_m-1} h_m(\Phi^{-1}(\mathbf{z})) u_j \quad (4.33)$$

4.3.4 Ley de Control de Desacoplamiento MIMO

Es importante señalar que el grado relativo r_i , $i = 1, \dots, m$, es el entero más pequeño requerido para que por lo menos una de las entradas aparezca en (ver(4.33)):

$$y_i^{(r_i)} = L_f^{r_i} h_i(\Phi^{-1}(\mathbf{z})) + \sum_{j=1}^m L_{g_j} L_f^{r_i-1} h_i(\Phi^{-1}(\mathbf{z})) u_j$$

Consideremos ahora la siguiente *ley de control desacoplada*:

$$u = A^{-1}(z)[-B(z) + \mathbf{v}] \quad (4.34)$$

donde $\mathbf{v} = [v_1 \dots v_m]^T$, es una nueva entrada a ser determinada, y $A(z)$, (ver (4.31)), y $B(z)$ son matrices con elementos:

$$b_i = L_f^{r_i} h_i(\Phi^{-1}(\mathbf{z})), \quad i = 1, \dots, m$$

$$a_{ij} = L_{g_j} L_f^{r_j-1} h_j(\Phi^{-1}(\mathbf{z})), \quad i, j = 1, \dots, m$$

respectivamente. Sustituyendo los elementos de u_i , $i = 1, \dots, m$ (ecuación (4.34)) en (4.33), se obtiene la representación canónica lineal invariante en el tiempo del proceso MIMO; es decir, la ecuación (4.33) con: $y_i^{(r_i)} = v_i$.

Por otro lado, de la relación $\mathbf{y}(s) = \mathbf{H}(s)\mathbf{v}(s)$, se puede demostrar que la transferencia $\mathbf{H}(s)$, del proceso linealizado MIMO [13]:

$$\mathbf{H}(s) = \begin{bmatrix} 1/s^{r_1} & 0 & \dots & 0 \\ 0 & 1/s^{r_2} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 1/s^{r_m} \end{bmatrix} \quad (4.35)$$

lo cual significa que $y_i^{(r_i)} = d^{r_i} y_i / dt^{r_i} = v_i$, $i = 1, \dots, m$ donde cada entrada v_i es diseñada por ubicación de polos aplicado al correspondiente subsistema lineal equivalente. Tal subsistema puede ser seleccionada como:

$$v_i = \frac{d^{r_i} y_i}{dt^{r_i}} = y_i^{(r_i)} = p_i^{(n)} + K_{i,1} \dot{y}_i^{(r_i-1)} + \dots + K_{i,r_i-1} \dot{y}_i + K_{i,r_i} y_i, \quad i = 1, \dots, m \quad (4.36)$$

donde $p_i(t)$ es la i -ésima trayectoria deseada e $\bar{y}_i(t) = p_i(t) - y_i(t)$ es el i -ésima señal de error de seguimiento. La i -ésima ecuación característica del subsistema es:

$$\bar{y}_i^{(r_i)} + K_{i,1}\bar{y}_i^{(r_i-1)} + \dots + K_{i,r_i-1}\dot{\bar{y}}_i + K_{i,r_i}\bar{y}_i = \mathbf{0}, \quad i = 1, \dots, m \quad (4.37)$$

Para comportamiento asintóticamente estable, es decir con $\bar{y}_i \rightarrow 0$ o $y_i \rightarrow p_i$ para $t \rightarrow \infty$, ninguno de las raíces de (4.37) debe ser localizado en el lado derecho del plano s , incluyendo el eje imaginario. Los parámetros del controlador $K_{i,j}$; $i = 1, \dots, m$; $j = 1, \dots, r_i$, son seleccionados para encontrar tales requerimientos, y la entrada v_i tomará la forma:

$$v_i = p_i^{(r_i)} + \sum_{j=1}^{r_i} K_{i,j} p_i^{(r_i-j)} - \mathbf{K}_i \mathbf{z}_i, \quad i = 1, \dots, m \quad (4.38)$$

donde:

$$\mathbf{K}_i = \begin{bmatrix} K_{i,1} & \dots & K_{i,r_i} \end{bmatrix} \quad \mathbf{z}_i = \begin{bmatrix} y_i^{(r_i-1)} \\ \vdots \\ y_i \end{bmatrix} \quad i = 1, \dots, m \quad (4.39)$$

Capítulo 5

Simulación con Matlab

En esta sección se muestran las simulaciones del sistema de control no lineal multivariable de tres entradas y tres salidas, por realimentación de estado para el SRM de 3DOF.

Para obtener las simulaciones respectivas se ha tenido en cuenta:

1. El modelo en el espacio-estado, visto en la subsección 3.4.4.
2. El diseño del controlador no lineal multivariable basado en el método de linealización exacta por realimentación de estado, visto en la subsección 4.3.4. Su diagrama de bloques se ilustra en la figura 5.1.
3. La programación del algoritmo de control no lineal realizada en MATLAB.

Referencias

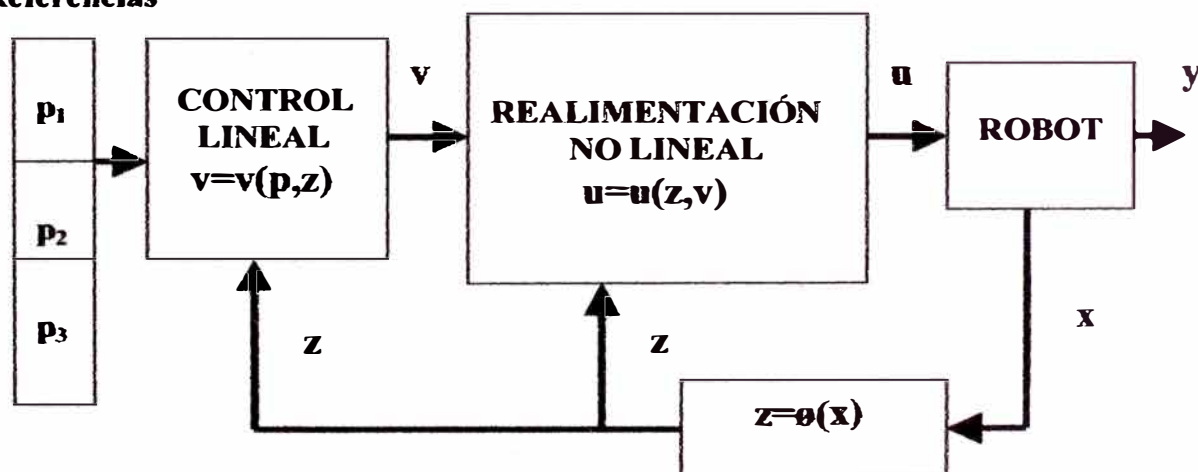


Figura 5.1: Diagrama de bloques del controlador no lineal

El diagrama de bloques de la figura 5.1 es explicado de la siguiente manera: las variables de estado x del proceso no lineal (robot manipulador esférico de 3DOF) generan un cambio de coordenadas en una nueva ecuación de estado z . Con dichas variables realimentadas y las tres referencias p_1 , p_2 y p_3 , se determinan las nuevas señales de control lineal v , mediante la técnica por ubicación de polos. Así mismo

con estas señales de control \mathbf{v} y con las variables de estado realimentadas \mathbf{z} , se generan las señales de control real \mathbf{u} que intervienen en el proceso no lineal.

Basado en los conceptos anteriores se han realizado las simulaciones utilizando el programa MATLAB [10], [11]. El desarrollo de las simulaciones se ha dividido para control de trayectoria constante y para control de trayectoria senoidal.

5.1 Caso 1: Control de Trayectoria Constante

Para esta simulación se ha asumido una trayectoria constante con los siguientes datos:

- Condiciones iniciales:
 - $x_1 = 0$, (base)
 - $x_2 = 0$, (1ra. varilla)
 - $x_3 = 0$, (2da. varilla)
- Trayectorias deseadas:
 - $p_1 = \pi/2$, posición angular de la base
 - $p_2 = 2\pi/3$, posición angular de la primera varilla
 - $p_3 = \pi$, posición angular de la segunda varilla
- Polos usados:
 - para la base: -8 y -8
 - para la primera varilla: -6 y -6
 - para la segunda varilla: -6 y -6
- Tiempo de muestreo:
 - $T = 0.010$ seg

5.1.1 Linealización por Realimentación

Un sistema no lineal multivariable de orden n de m entradas y m salidas es completamente linealizabile y desacoplado si su grado relativo total es:

$$r = r_1 + \dots + r_m$$

Si el grado relativo total r , es igual al número de orden del sistema o sea " $r = n$ ", entonces se aplica el método de linealización exacta. Aplicando los conceptos de geometría diferencial [7], [13], [14], [15], determinamos el grado relativo total.

5.1.2 Determinación del Grado Relativo Total: r

Aplicando los conceptos de las subsecciones 4.2.2 y 4.3.2, se obtiene:

- Primera salida: $y_1 = h_1(x) = x_1$. La segunda derivada de y_1 , es diferente de cero, por lo tanto el grado relativo de $r_1 = 2$
- Segunda salida: $y_2 = h_2(x) = x_2$. La segunda derivada de y_2 , es diferente de cero, por lo tanto el grado relativo de $r_2 = 2$
- Tercera salida: $y_3 = h_3(x) = x_3$. La segunda derivada de y_3 , es diferente de cero, por lo tanto el grado relativo de $r_3 = 2$
- Grado Relativo Total: $r = r_1 + r_2 + r_3 = 6$, que es igual al orden del sistema, por lo tanto se puede aplicar el método de linealización exacta.

5.1.3 Forma Normal para Linealización

Aplicando los conceptos de las subsecciones 4.2.3 y 4.3.3, se obtiene la transformación de coordenadas $z = \Phi(x)$, en la siguiente forma matricial:

$$z = \Phi(x) = \begin{bmatrix} h_1(x) \\ L_f h_1(x) \\ h_2(x) \\ L_f h_2(x) \\ h_3(x) \\ L_f h_3(x) \end{bmatrix} = \begin{bmatrix} x_1 \\ x_4 \\ x_2 \\ x_5 \\ x_3 \\ x_6 \end{bmatrix} = \begin{bmatrix} z_1 \\ z_2 \\ z_3 \\ z_4 \\ z_5 \\ z_6 \end{bmatrix}$$

por lo tanto la nueva ecuación de estado es:

$$\dot{z} = \begin{bmatrix} \dot{z}_1 \\ \dot{z}_2 \\ \dot{z}_3 \\ \dot{z}_4 \\ \dot{z}_5 \\ \dot{z}_6 \end{bmatrix} = \begin{bmatrix} z_2 \\ v_1 \\ z_4 \\ v_2 \\ z_6 \\ v_3 \end{bmatrix} \quad (5.1)$$

5.1.4 Diseño del Controlador No Lineal MIMO

Para la nueva ecuación de estado (5.1) y aplicando la ecuación (4.36), se obtiene las nuevas entradas lineales equivalentes v_1 , v_2 , y v_3 , y son expresadas de la siguiente forma, [12], [13], [14]:

$$v_1 = \dot{x}_4 = \ddot{p}_1 + K_1 \dot{\bar{y}}_1 + K_2 \bar{y}_1 \quad (5.2)$$

$$v_2 = \dot{x}_5 = \ddot{p}_2 + K_3 \dot{\bar{y}}_2 + K_4 \bar{y}_2 \quad (5.3)$$

$$v_3 = \dot{x}_6 = \ddot{p}_3 + K_5 \dot{\bar{y}}_3 + K_6 \bar{y}_3 \quad (5.4)$$

donde:

p_1 , p_2 y p_3 : trayectorias deseadas para la primera, segunda y tercera entrada

\bar{y}_1 : error en la primera salida

\bar{y}_2 : error en la segunda salida

\bar{y}_3 : error en la tercera salida

así mismo se tiene las siguientes equivalencias:

$$\bar{y}_1 = p_1 - y_1, \quad x_1 = y_1, \quad \dot{x}_1 = x_4 \quad (5.5)$$

$$\bar{y}_2 = p_2 - y_2, \quad x_2 = y_2, \quad \dot{x}_2 = x_5 \quad (5.6)$$

$$\bar{y}_3 = p_3 - y_3, \quad x_3 = y_3, \quad \dot{x}_3 = x_6 \quad (5.7)$$

reemplazando (5.5) en (5.2), se obtiene:

$$v_1 = \dot{x}_4 = \ddot{p}_1 + K_1 \dot{p}_1 + K_2 p_1 - K_1 x_4 - K_2 x_1 \quad (5.8)$$

de la misma manera, reemplazando (5.6) en (5.3), se obtiene:

$$v_2 = \dot{x}_5 = \ddot{p}_2 + K_3 \dot{p}_2 + K_4 p_2 - K_3 x_5 - K_4 x_2 \quad (5.9)$$

finalmente, reemplazando (5.7) en (5.4), se obtiene:

$$v_3 = \dot{x}_6 = \ddot{p}_3 + K_5 \dot{p}_3 + K_6 p_3 - K_5 x_6 - K_6 x_3 \quad (5.10)$$

se sabe que para trayectoria constante se tiene:

$$\dot{p}_1 = \dot{p}_2 = \dot{p}_3 = \ddot{p}_1 = \ddot{p}_2 = \ddot{p}_3 = 0$$

reemplazando estos datos en (5.8), (5.9) y (5.10) se obtiene:

$$v_1 = \dot{x}_4 = K_2 p_1 - K_1 x_4 - K_2 x_1 \quad (5.11)$$

$$v_2 = \dot{x}_5 = K_4 p_2 - K_3 x_5 - K_4 x_2 \quad (5.12)$$

$$v_3 = \dot{x}_6 = K_6 p_3 - K_5 x_6 - K_6 x_3 \quad (5.13)$$

donde, K_1, K_2, K_3, K_4, K_5 y K_6 , son los valores de la ecuación característica que depende de los polos del sistema lineal equivalente.

Con (5.11) y (3.40) se obtiene la ley control u_1 para la primera entrada:

$$u_1 = P_1 + v_1 R_{T1} (H_{11} + J_{eq1}) \pm C_{e1} n_1 R_{T1} x_4 \quad (5.14)$$

con (5.12) y (3.41) se obtiene la ley control u_2 para la segunda entrada:

$$u_2 = P_2 + v_2 R_{T2} (J_{eq2} + H_{22}) + v_3 R_{T2} H_{23} \pm C_{e1} n_2 R_{T2} x_5 \quad (5.15)$$

finalmente con (5.13) y (3.42) se obtiene la ley control u_3 para la tercera entrada:

$$u_3 = P_3 + v_2 R_{T3} H_{32} + v_3 R_{T3} (J_{eq3} + H_{33}) \pm C_{e3} n_3 R_{T3} x_6 \quad (5.16)$$

donde:

$$P_1 = R_{T1} (B_{eq1} x_4 + C_1) + N_{T1} x_4$$

$$P_2 = R_{T2} (B_{eq2} x_5 + C_2 + d_2) + N_{T2} x_5$$

$$P_3 = R_{T3} (B_{eq3} x_6 + C_3 + d_3) + N_{T3} x_6$$

$$J_{eq1} = n_1^2 J_{m1} + J_{g1}$$

$$B_{eq1} = n_1^2 B_{m1} + B_{g1}$$

$$J_{eq2} = n_2^2 J_{m2} + J_{g2}$$

$$B_{eq2} = n_2^2 B_{m2} + B_{g2}$$

$$J_{eq3} = n_3^3 J_{m3} + J_{g3}$$

$$B_{eq3} = n_3^3 B_{m3} + B_{g3}$$

$$R_{T1} = \frac{R_a}{n_1 K_{m1} K_{A1}}$$

$$R_{T2} = \frac{R_a}{n_2 K_{m2} K_{A2}}$$

$$R_{T3} = \frac{R_a}{n_3 K_{m3} K_{A3}}$$

$$N_{T1} = \frac{n_1 K_{b1}}{K_{A1}}$$

$$N_{T2} = \frac{n_2 K_{b2}}{K_{A2}}$$

$$N_{T3} = \frac{n_3 K_{b3}}{K_{A3}}$$

$$H_{11} = a_1 + a_2 \cos^2(q_2) + a_3 \cos^2(q_2 + q_3) + a_4 \cos^2(q_2) + 2a_5 \cos(q_2) \cos(q_2 + q_3)$$

$$H_{22} = a_2 + a_3 + a_4 + 2a_5 \cos(q_3)$$

$$H_{23} = H_{23} = a_3 + a_5 \cos(q_3)$$

$$H_{33} = a_3$$

$$C_1 = -a_2 \sin(2q_2) \dot{q}_1 \dot{q}_2 - 2(a_3 \sin(q_2 + q_3) \cos(q_2 + q_3) + a_5 \sin(q_2 + q_3) \cos(q_2)) \dot{q}_1 \dot{q}_3$$

$$C_2 = a_2 \sin(q_2) \cos(q_2) \dot{q}_1^2 - 2a_5 \sin(q_3) \dot{q}_2 \dot{q}_3 - a_5 \sin(q_3) \dot{q}_3^2$$

$$C_3 = (a_3 \sin(q_2 + q_3) \cos(q_2 + q_3) + a_5 \sin(q_2 + q_3) \cos(q_2)) \dot{q}_1^2 + a_5 \sin(q_3) \dot{q}_2^2$$

$$d_2 = a_6 \cos(q_2) + a_7 \cos(q_2 + q_3)$$

$$d_3 = a_7 \cos(q_2 + q_3)$$

$$a_1 = I_{11} - m_2 L_2^2$$

$$a_2 = I_2 + 2m_2 L_2 L_{c_{s2}} - m_2 L_2^2$$

$$a_3 = I_3 + 2m_3 L_3 L_{c_{s3}} - m_3 L_3^2$$

$$a_4 = m_3 L_2^2$$

$$a_5 = m_3 L_2 L_{c_{s3}}$$

$$a_6 = m_2 g L_{c_{s2}} + m_3 g L_2$$

$$a_7 = m_3 g L_{c_{s3}}$$

5.1.5 Programación del Algoritmo de Control No Lineal Multivariable

Para la programación del algoritmo de control se ha usado MATLAB [11], teniendo como dato las condiciones iniciales indicadas anteriormente y las ecuaciones (5.1), (5.11), (5.12), (5.13), (5.14), (5.15) y (5.16). Dicho programa se muestra en el apéndice B.

5.1.6 Simulaciones

Las simulaciones respectivas son mostradas desde la figura 5.2 hasta la 5.4.

En la figura 5.2 se muestra las salidas controladas para la posición angular de la base a una referencia de 90° , posición angular de la primera varilla a una referencia de 120° , posición angular de la segunda varilla a una referencia de 180° ; el tiempo de establecimiento para la base es de 1.2 seg., para la primera varilla el tiempo de establecimiento es de 2.1 seg. y para la segunda varilla el tiempo de establecimiento es de 1.5 seg.

En la figura 5.3 se muestran las señales de control para el servomotor de la base y los servomotores de las 2 varillas. La señal del servomotor de la base

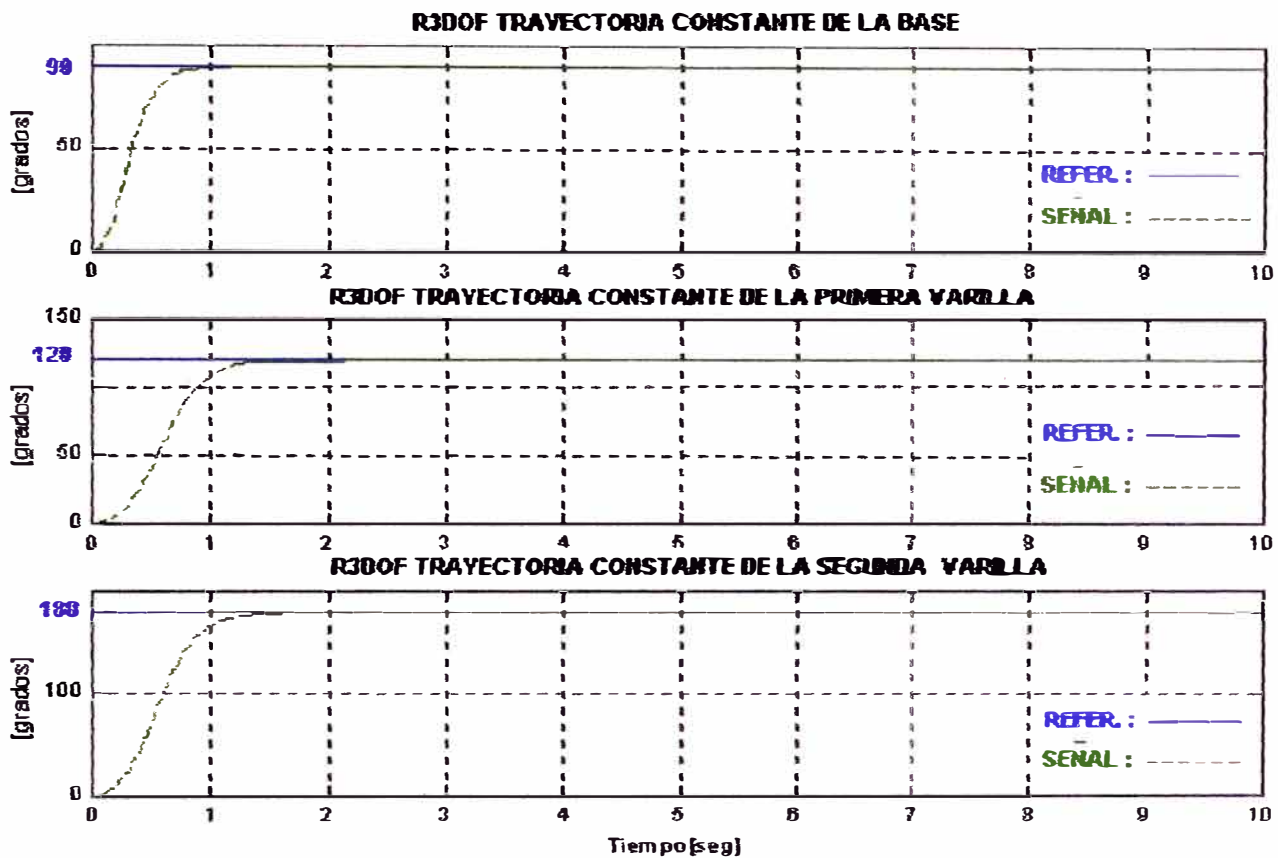


Figura 5.2: Trayectoria constante: salidas controladas para la base, primera y segunda varilla

varía de +10 voltios a -5 voltios por 0.5 seg., para luego aproximarse a cero a los 1.2 seg., mientras que la señal del servomotor de la primera varilla varía de +10 voltios a -5 voltios por 0.7 seg., para luego aproximarse a -1.5 voltios a los 1.7 seg., y finalmente la señal del servomotor de la segunda varilla varía +10 voltios a -10 voltios por 0.8 seg. para luego aproximarse a +0.25 voltios a los 1.5 seg.; de lo anterior se concluye que cuando las señales de control se hacen convergentes el proceso ha sido controlado.

En la figura 5.4 se muestra la respuesta del sistema para la velocidad angular del servomotor de la base y de las 2 varillas, se aprecia que en el arranque la velocidad para los 3 casos se incrementa, para luego aproximarse a cero a los 1 seg. para el servomotor de la base; a los 1.8 seg. para el servomotor de la primera varilla y a los 1.5 seg. para el servomotor de la segunda varilla. Esto significa que cuando dichas velocidades convergen a cero, el proceso ha sido controlado.

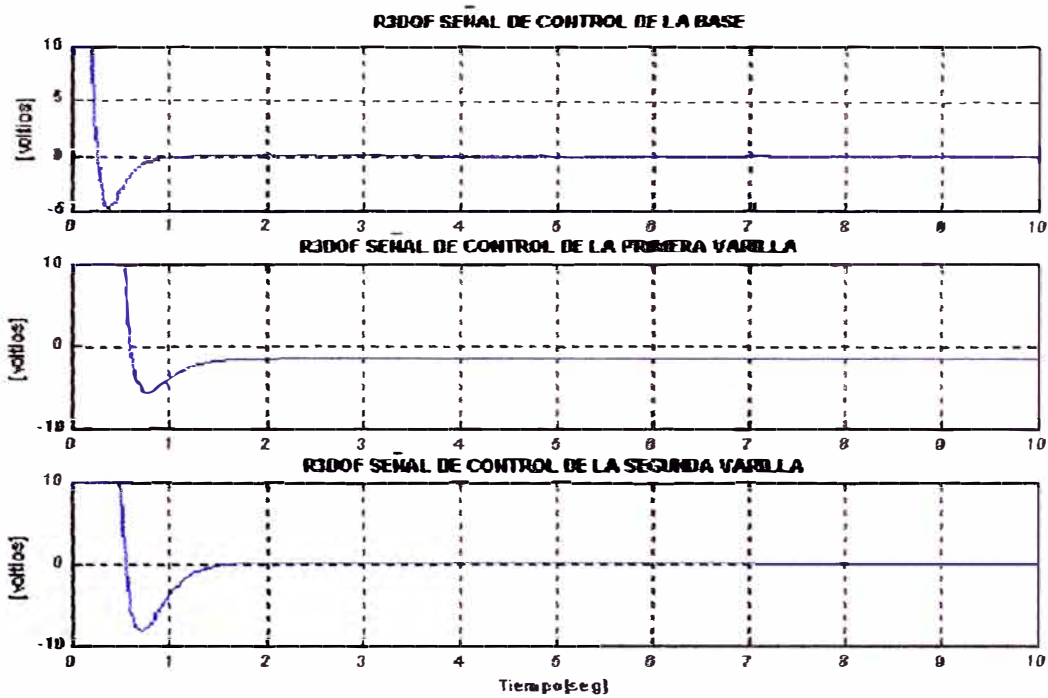


Figura 5.3: Trayectoria constante: señales de control para los servomotores de la base, primera y segunda varilla

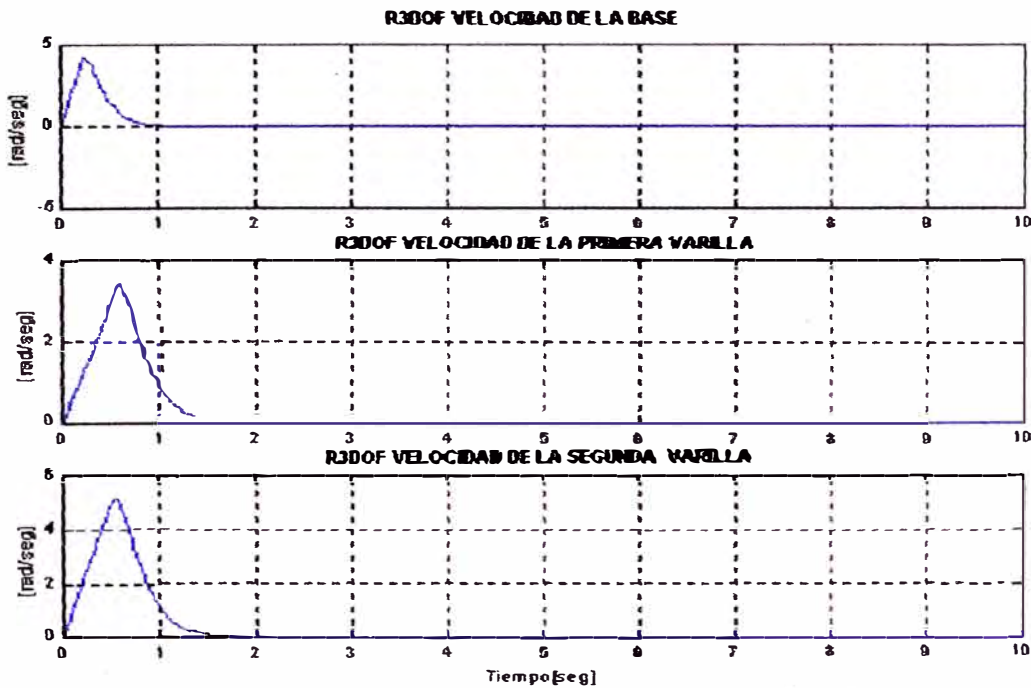


Figura 5.4: Trayectoria constante: velocidad para los servomotores de la base, primera y segunda varilla

5.2 Caso 2: Control de Trayectoria Senoidal

Para esta simulación se ha asumido una trayectoria senoidal con los siguientes datos:

- Condiciones iniciales:

$$x_1 = 0, \text{ (base)}$$

$$x_2 = 0, \text{ (1ra. varilla)}$$

$$x_3 = 0, \text{ (2da. varilla)}$$

- Trayectorias deseadas:

$$p_1 = \pi \cos(2\pi kT/11), \text{ posición angular de la base}$$

$$p_2 = 3\pi/2 \cos(2\pi kT/11), \text{ posición angular de la 1ra. varilla}$$

$$p_3 = 2\pi \cos(2\pi kT/11), \text{ posición angular de la 2da. varilla}$$

- Polos usados:

para la base: -8 y -8

para la primera varilla: -6 y -6

para la segunda varilla: -40 y -40

- Tiempo de muestreo:

$$T = 0.01 \text{ seg}$$

5.2.1 Diseño del Controlador No Lineal MIMO

Para el diseño del controlador no lineal MIMO, se usan las ecuaciones (5.8) y (5.14) para la primera entrada, las ecuaciones (5.9) y (5.15) para la segunda entrada y las ecuaciones (5.10) y (5.16) para la tercera entrada.

5.2.2 Programación del Algoritmo de Control No Lineal Multivariable

Para la programación del algoritmo de control se ha usado MATLAB [11], teniendo como dato las condiciones iniciales indicadas anteriormente y las ecuaciones (5.1) (5.8), (5.9), (5.10), (5.14), (5.15) y (5.16). Dicho programa se muestra en el apéndice B.

5.2.3 Simulaciones

Las simulaciones respectivas son mostradas desde la figura 5.5 hasta la figura 5.7.

En la figura 5.5 se muestra las salidas controladas para la posición angular de la base, posición angular de la primera varilla y posición angular de la segunda varilla, realizando un seguimiento a la señal deseada. Las señales se han fijado para un período de 11 seg. Las amplitudes de las señales senoidales indican el movimiento de la base y las 2 varillas. El tiempo de establecimiento es de 1.3 seg. para la base, 2.5 seg. para la primera varilla y 2 seg. para la segunda varilla; para después seguir a la trayectoria deseada.

En la figura 5.6 se muestran las señales de control para el servomotor de la base, el servomotor de la primera varilla y el servomotor de la segunda varilla. Las señales de los servomotores de la base y las 2 varillas llegan hasta ± 10 voltios en forma abrupta por un lapso de tiempo de 0.5, 2 y 2 seg. Posteriormente la señal de control de la base disminuye en amplitud hasta hacerse bien pequeña ± 1.0 voltios, pero siguiendo una aproximación a una trayectoria senoidal abrupta. Seguidamente la señal de control de la primera varilla disminuye en amplitud hasta hacerse pequeña ± 4 voltios, pero siguiendo una trayectoria senoidal oscilatoria abrupta. Finalmente la señal de control de la segunda varilla disminuye en amplitud hasta hacerse bien pequeña ± 0.8 voltios, pero siguiendo también una trayectoria senoidal oscilatoria abrupta.

En la figura 5.7 se muestra la respuesta del sistema para la velocidad angular del servomotor de la base, velocidad angular de la primera varilla y velocidad angular de la segunda varilla. Se aprecia que en el arranque la trayectoria de la velocidad se incrementa para los 3 casos por un lapso de tiempo de 1, 2 y 2 seg. Posteriormente la velocidad del servomotor de la base disminuye en amplitud hasta -1.8 rad/seg., para luego seguir una trayectoria senoidal de la misma amplitud. Seguidamente la señal de control de la primera varilla disminuye en amplitud hasta -4 rad/seg., para luego seguir una trayectoria senoidal de amplitud de 2.7 rad/seg. Finalmente la velocidad del servomotor de la segunda varilla disminuye en amplitud hasta -3.6 rad/seg., para luego seguir una trayectoria senoidal con la misma amplitud. Se puede notar que para los tres casos tienen el mismo período de 11 seg.

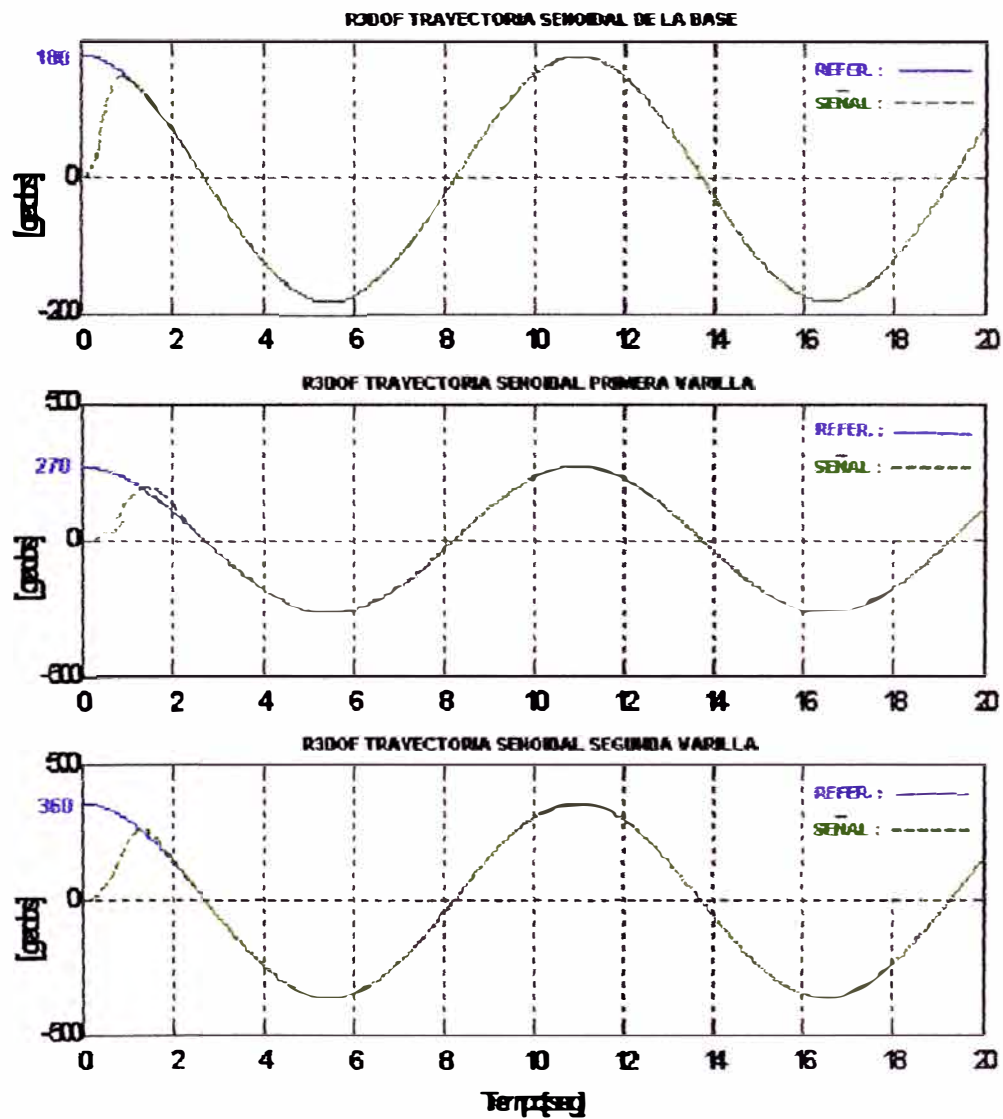


Figura 5.5: Trayectoria senoidal: salidas controladas para la base, primera y segunda varilla

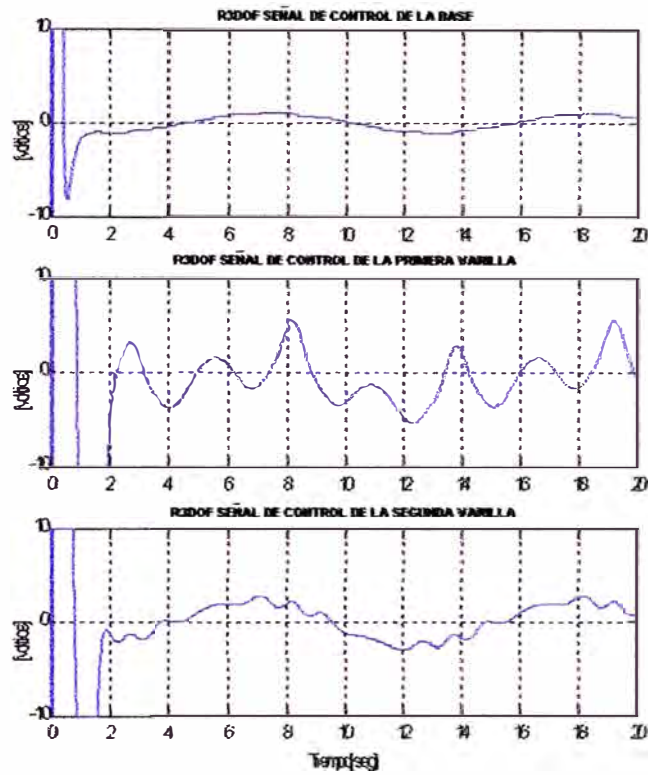


Figura 5.6: Trayectoria senoidal: señales de control para los servomotores de la base, primera y segunda varilla

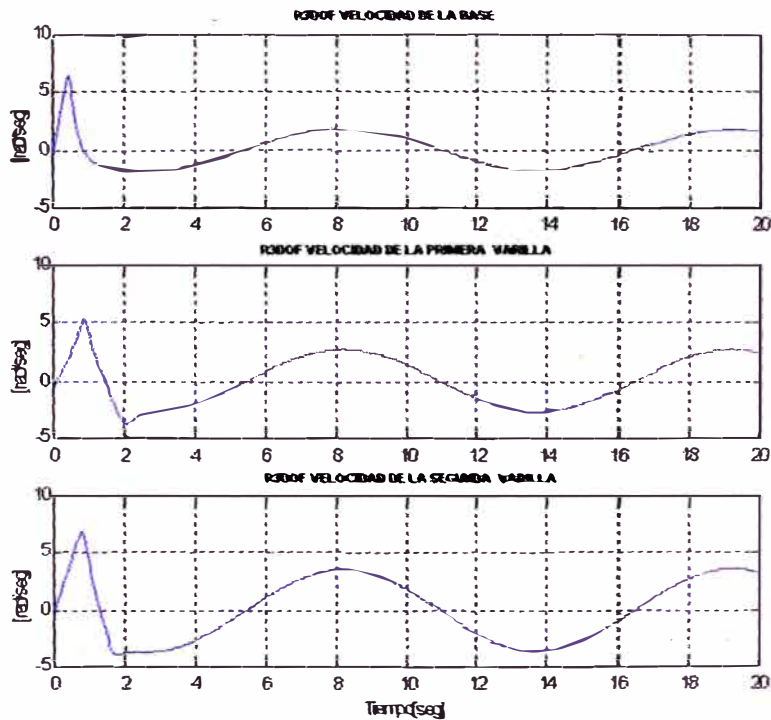


Figura 5.7: Trayectoria senoidal: velocidad de los servomotores de la base, primera y segunda varilla

Capítulo 6

Especificaciones Técnicas

6.1 Introducción

En este capítulo se tratará sobre las especificaciones técnicas para una futura implementación del sistema del Robot Manipulador Esférico de 3 grados de libertad en tiempo real, el cual se compondrá en el hardware que son los componentes físicos a usarse en el sistema de control y en el software que es el lenguaje de programación a usarse para el algoritmo de control del sistema.

6.2 Hardware del Sistema

Los componente físicos a usarse en el sistema de control serán:

- La planta
- El actuador
- El sensado
- La Interfaz de Adquisición de Datos

6.2.1 La Planta

El diseño del manipulador de 3 grados de libertad que viene a ser la planta se muestra en la figura 6.1 y estará formado por:

- Tres servomotores DC con encoder óptico incorporado
- Una base giratoria que vendrá a ser el primer grado de libertad estará formada por un disco metálico que irá acoplada con una bocina de bronce al eje del primer servomotor que estará en posición vertical y una barra metálica de aluminio de sección cuadrada que irá acoplada al disco. Sobre el extremo superior de la barra irá sujeto el segundo servomotor del segundo grado de libertad.

- El segundo grado de libertad estará formada por una varilla rígida metálica de aluminio que irá acoplada de su extremo inferior con una bocina de bronce al eje del segundo servomotor que estará en posición horizontal.
- El tercer grado de libertad estará formada por otra varilla rígida metálica de aluminio que irá acoplada de su extremo inferior con una bocina de bronce al extremo superior de la primera varilla y al eje del tercer servomotor que estará también en posición horizontal.

6.2.2 El Actuador

El actuador estará formado por:

- Tres servomotores DC de campo magnético permanente y escobillas conmutadas con encoder óptico incremental incorporado. Los servomotores DC del primer y segundo grado de libertad tendrán las siguientes características y se muestra en la figura 6.2:

Marca MATSUSHITA, modelo GMX-7MCO19B

Voltage nominal: 24 VDC

Corriente en vacío: 0.7 A

Velocidad en vacío: 255 RPM

Relación de engranajes (n_1 y n_2): 18.5:1

Torque del servomotor a plena carga: 5.65 Nm

Corriente de armadura a plena carga: 7 A

Voltage de entrada del encoder: 5 VDC

Salida del encoder: 1850 pulsos/rev. de salida de eje con 2 canales en cuadratura.

El servomotor DC del tercer grado de libertad tendrá la siguiente característica:

Marca MATSUSHITA, modelo GMX-6MPO13A

Voltage nominal: 24 VDC

Corriente en vacío: 0.2 A

Velocidad en vacío: 310 RPM

Relación de engranajes (n_3): 12.5:1

Torque del servomotor a plena carga: 2.37 in-lbs

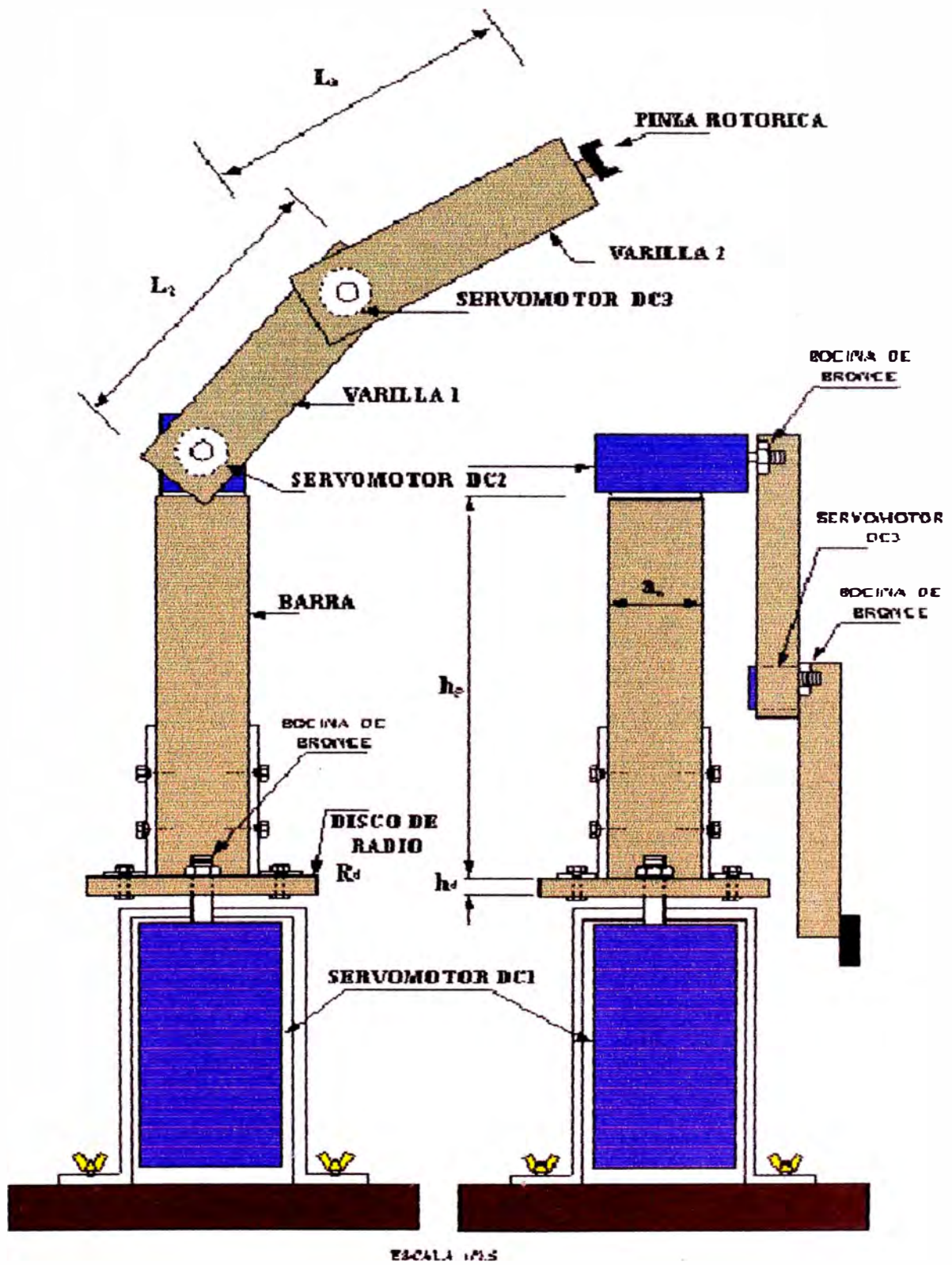


Figura 6.1: Diseño del Robot Manipulador Esférico de 3DOF

Corriente de armadura a plena carga: 4 A

Voltage de entrada del encoder: 5 VDC

Salida del encoder: 1250 con 2 canales en cuadratura



Figura 6.2: Servomotor DC

- Tres Amplificadores de Potencia, una para cada servomotor DC, en la figura 6.3 se muestra el circuito del amplificador que estará conformado por el transistor amplificador PN2222 y el transistor preamplificador PN2907 que están acoplados por un emisor común y que trabajan en la zona de corte y saturación. Así mismo en dicho circuito se muestra dos pares de conmutadores $A_{1,4}$ y $A_{2,3}$ conformado por los MOSFET's IRFZ34 cuyo esquema simplificado se muestra en la figura 6.4. El funcionamiento de este amplificador se basa en la lógica de control para activar al circuito de disparo, es decir, cuando el sistema de disparo cierra al conmutador $A_{1,4}$ y se abre $A_{2,3}$, el sentido de la corriente es la línea con puntos induciendo de esta

forma una tensión $+V_{cc}$ en el servomotor. Luego si el sistema de disparo abre al conmutador $A_{1,4}$ y cierra $A_{2,3}$, el sentido de la corriente es la línea continua induciendo así una tensión $-V_{cc}$ en el servomotor. Por lo tanto el servomotor ve en sus bornes una onda de voltage cuadrada, variando entre $\pm V_{cc}$ y la corriente que puede absorber dependerá de los conmutadores. En resumen de lo anterior podemos decir que el objetivo del sistema de disparo es realizar la conmutación de cada par $A_{1,4}$ y $A_{2,3}$, de tal forma que se eviten cortocircuitos durante el tiempo de conmutación de $A_{1,4}$ y $A_{2,3}$. El circuito amplificador de potencia de la figura 6.2 tiene un circuito construido en base en compuertas lógicas de tecnología TTL (Transistor Transistor Logic) que conformarán los circuitos integrados 7404, 7486, 7408 y dos resistencias de 500Ω . Este circuito gobernará los disparos de los MOSFET's de manera que las señales PWM que ingresan al circuito amplificador de potencia que tengan igual amplitud estarán siempre en nivel bajo (OFF) tal como se muestra en la tabla 6.1.

Tabla 6.1: Conmutación de los MOSFET's.

PWM_1	PWM_2	A_1	A_2	A_3	A_4
0	0	OFF	OFF	OFF	OFF
0	1	ON	OFF	OFF	ON
1	0	OFF	ON	ON	OFF
1	1	OFF	OFF	OFF	OFF

- Tres Generadores PWM, una para cada servomotor DC, en la figura 6.5 se muestra el circuito del generador PWM que estará conformado por el integrado LM3524 que genera una modulación por ancho de pulso a una frecuencia de 15.4 KHz, posteriormente esta señal será amplificada por el amplificador de potencia que alimentará al servomotor DC. Los circuitos osciladores 74LS123 realizarán la función de sincronización para la generación del tren de pulsos en cuadratura, cuyas salidas serán las señales PWM1 y PWM2 que serán las entradas al circuito amplificador.

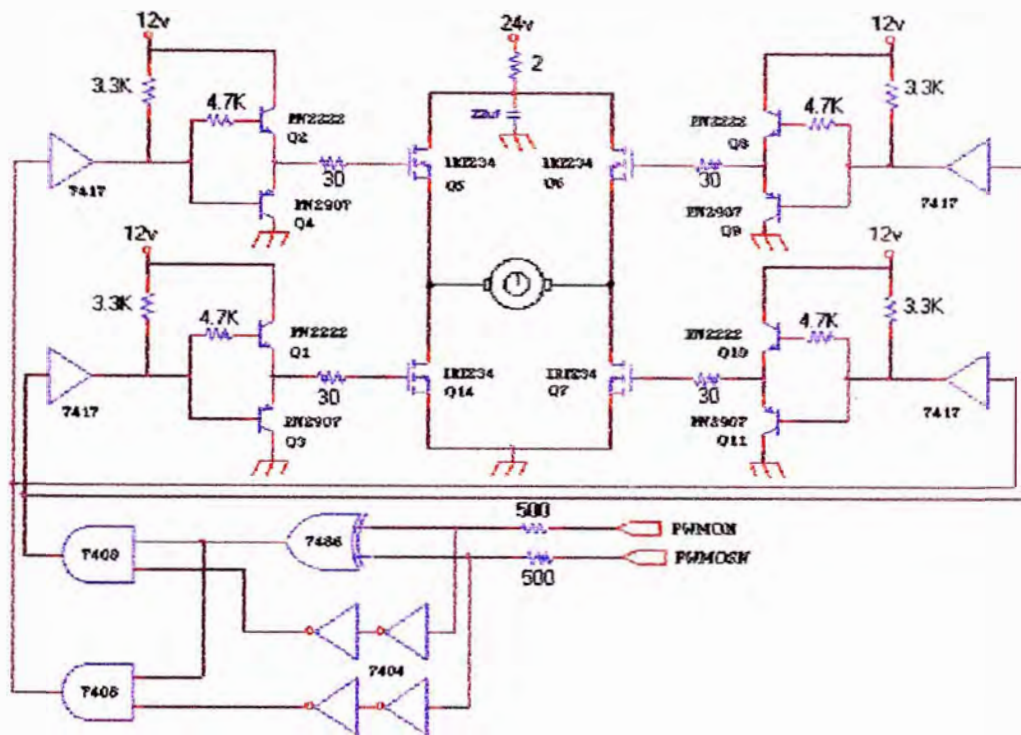


Figura 6.3: Circuito Amplificador de Potencia

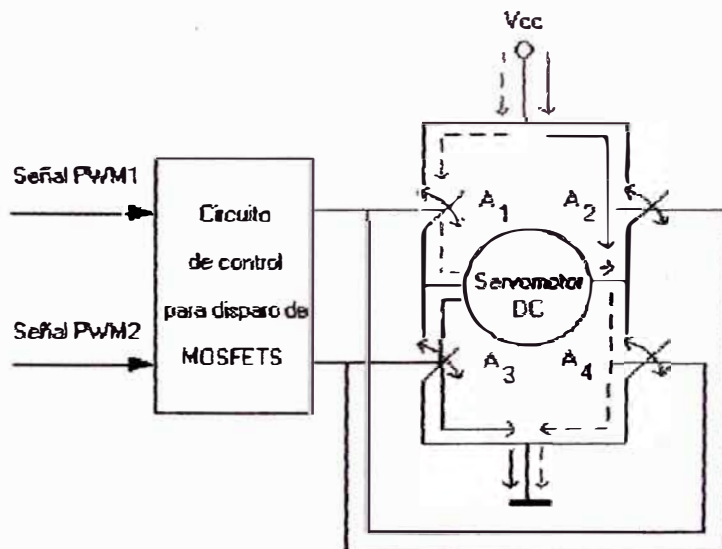


Figura 6.4: Esquema del circuito de disparo y conmutación

6.2.3 El Sensado

El sensado estará formado por el sensor de posición y el detector de cuadratura.

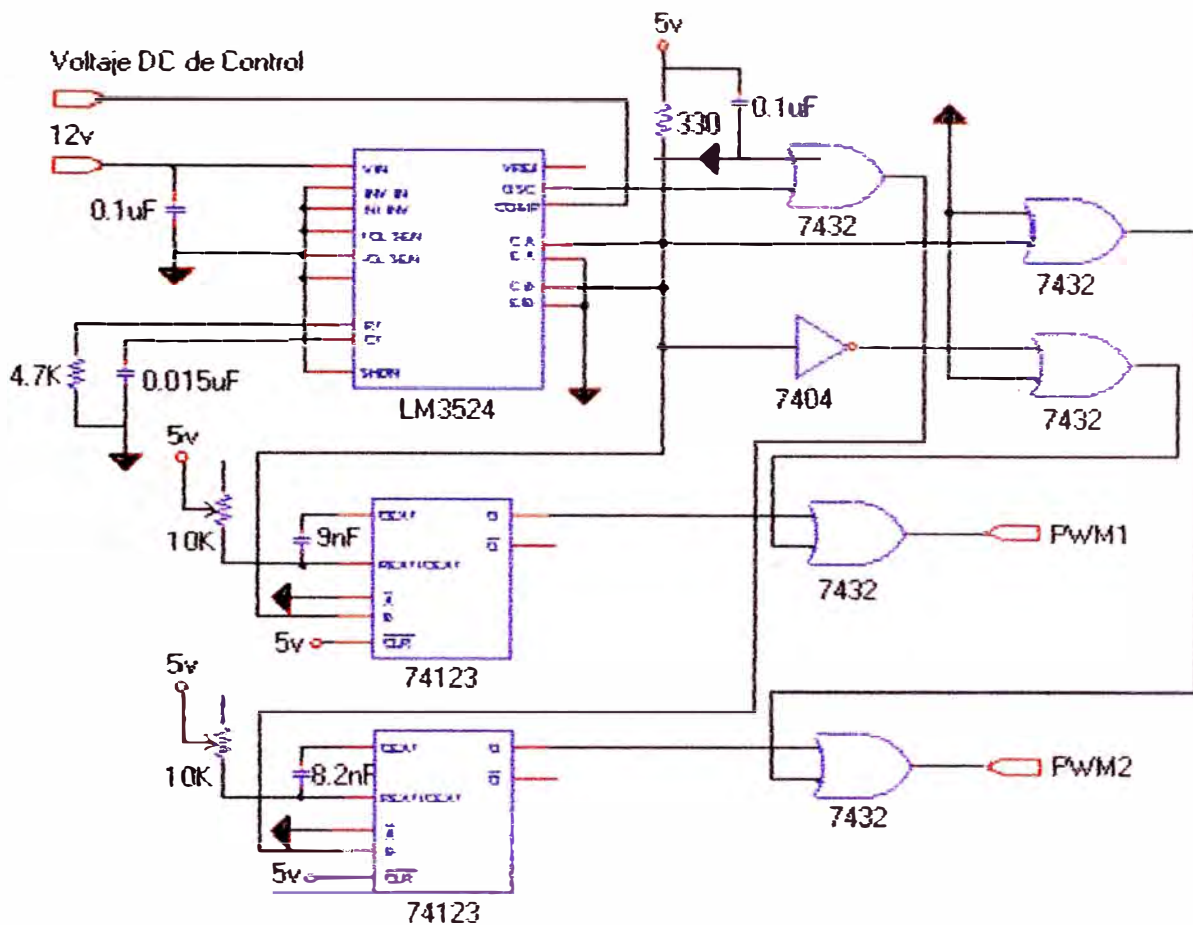


Figura 6.5: Circuito Generador PWM

El Sensor de Posición

El sensor de posición será el codificador óptico rotatorio, y estará incorporado en el servomotor DC como se muestra en la figura 6.6 y en la figura 6.7 se muestra los componentes del codificador el cual estará formado por:

- Una fuente de luz que puede ser un LED infrarrojo o láser. Es el que genera un haz de luz perpendicular al disco.
- Un disco ranurado acoplado al eje del servomotor y ubicado entre la fuente de luz y el detector.
- Un detector de luz, usualmente es un transistor óptico. Es el que genera un pulso de corriente que será la entrada a un contador de pulsos.
- Un conformador de pulsos que convierte la señal sinusoidal proveniente de la salida del detector en señal de onda cuadrada.

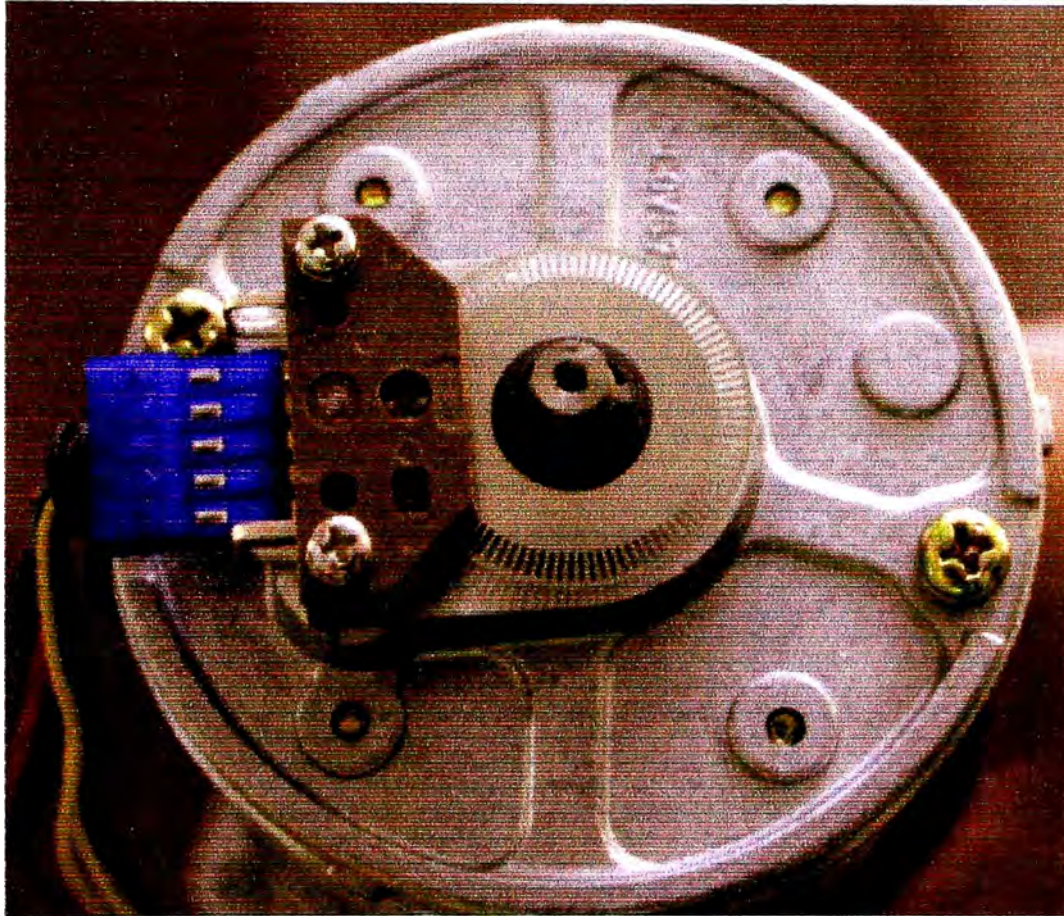


Figura 6.6: Codificador en el eje del servomotor.

El codificador que se usará, incluirá dos pistas de marcas o ranuras en cuadratura (canales A y B), cada una asociada a una fuente de luz y un transistor óptico independientes. Los canales A y B producen dos ondas cuadradas desfasadas $\pm 90^\circ$ entre sí que se utilizan para determinar la dirección del movimiento. Por ejemplo si el canal B adelanta al canal A, el eje está girando en el sentido anti-horario y viceversa.

Las características del codificador serán:

- Voltage de alimentación: 5VDC
- Resolución: 100 ppr (pulsos por revolución)
- Canales de salida: 2 canales TTL en cuadratura

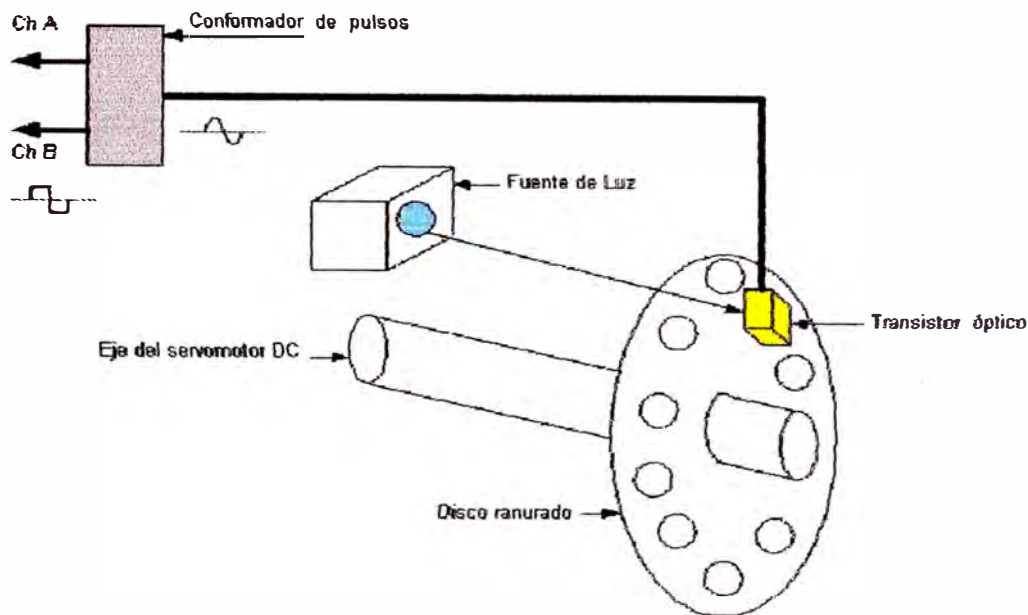


Figura 6.7: Estructura del codificador óptico rotatorio.

Detector de Cuadratura

El detector de cuadratura tendrá como propósito identificar la posición y el sentido de rotación del codificador óptico rotatorio. En la figura 6.8 se muestra el integrado LS7083, que activará la cuenta ascendente o descendente del contador UP/DOWN de 16 bits, dependiendo del giro del servomotor, captado por el desfase entre los trenes de pulso de los canales A y B. El contador UP/DOWN lo conformará dos integrados 74HC193, uno de ellos entregará un grupo de 8 bits de datos (bus A) a la interfaz de datos, en forma similar el otro grupo de igual bits conformará el bus B. X4/X1 son las entradas de resolución de multiplicación. Las salidas digitales PA, PB y PC del bloque contador son datos a procesar por la interfaz de datos.

6.2.4 La Interfaz de Adquisición de Datos

La interfaz de adquisición de datos será la encargada de manejar los datos de entrada y salida del sistema. En la figura 6.9 se muestra el esquema de la interfaz de adquisición de datos con los componentes que se podrían usar cuando se realice la implementación del sistema. Las características técnicas de la interfaz serán:

- Marca: National Instruments

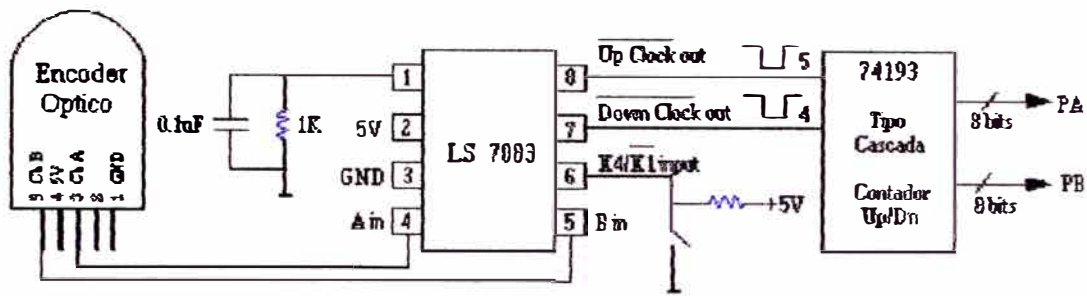


Figura 6.8: Detector de cuadratura.

- Modelo: NI 6713, insertable en la PC con Bus PCI
- Salidas analógicas: 8
- Resolución de salidas analógicas: 12 bits
- Voltage de entrada: ± 10
- Frecuencia: 1 MSamples/s
- Contador/Temporizador: 02 de 24 bits
- Entrada/Salida Digital: 8
- Cable Modelo: R6868
- Bornera Modelo: CB-68LP
- Driver Software: NI-DAQ para Windows 2000/NT
- Aplicación de Software: LabVIEW, Visual Basic, Lenguaje C++

Así mismo la interfaz que se muestra en la figura 6.7, estará compuesto por:

1. Tres salidas análogas (DAC0, DAC1 y DAC2), estas salidas serán las señales de control que se enviarán por los terminales del generador PWM (LM3524) con una resolución de 12 bits.
2. Tres puertos digitales de entrada (PA, PB y PC) de 8 bits cada uno. Estos puertos recibirán los datos provenientes del decodificador de cuadratura respectivo, que a su vez recibirá los datos de los codificadores ópticos de los servomotores.

3. El terminal OUTB0 que se usará como salida de una señal de reloj para el funcionamiento del sistema de control.
4. El terminal PD0 (bit 0 del puerto D) que se usará como entrada de la señal de reloj proveniente del OUTB0.

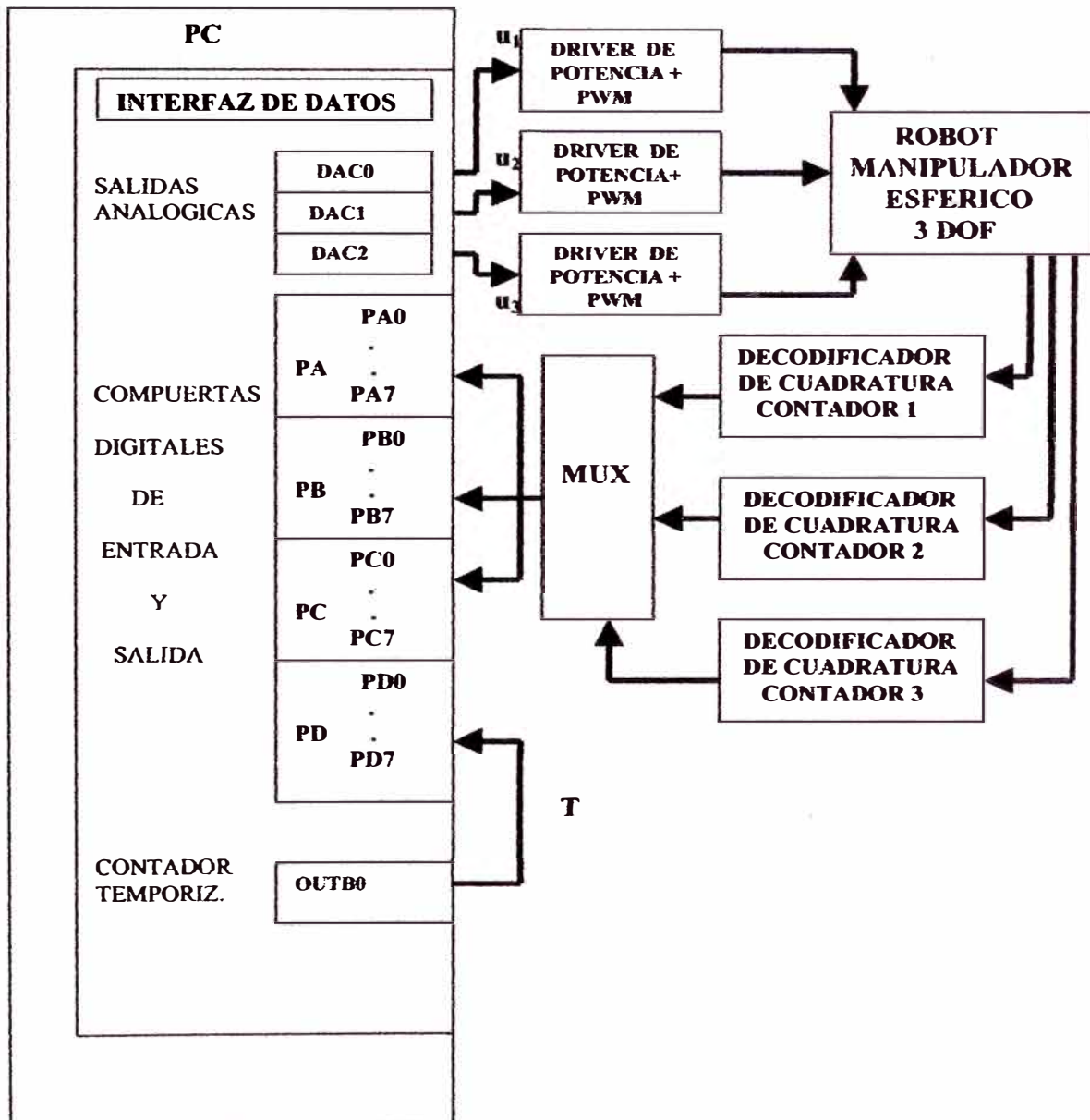


Figura 6.9: Interfaz de Adquisición de Datos e Implementación del Sistema.

Conclusiones

Finalmente se presenta las conclusiones derivadas de este trabajo, así como también algunas recomendaciones para investigaciones futuras.

Conclusiones

1. Para el diseño del sistema de control fue necesario determinar y calcular el modelo dinámico de la planta, ya que la técnica de control no lineal empleada lo requiere.
2. La ecuación dinámica del manipulador robótico se ha determinado empleando el algoritmo de Lagrange-Euler, el cual también puede ser aplicado a otros procesos físicos de dos ó más grados de libertad.
3. El planteamiento del problema fue el que las posiciones de la base y de los dos eslabones sigan trayectorias arbitrarias usando la técnica de control por realimentación de estado. Esta estrategia de control, requiere del modelo del proceso controlado. Las simulaciones realizadas en el capítulo 5 han demostrado que el modelo obtenido en el capítulo 3 es válido y cumple con las exigencias de un modelo no lineal multivariable con propósitos de control. De la misma manera, tales simulaciones también han demostrado que el sistema de control no lineal multivariable, cumple con las condiciones de diseño impuestas previamente: seguimiento de las salidas controladas con respecto a señales arbitrarias de referencia con mínimo sobreimpulso, mínimo tiempo de estabilización y error en estado estable nulo.

Recomendaciones

1. El proceso diseñado puede ser usado para implementarlo en tiempo real en hardware y software, para las áreas de control no lineal, control avanzado e inteligencia artificial, aplicado a la ingeniería de control, donde se demostraría el uso de la técnica de control en estudio así como también otras técnicas de control como por ejemplo: adaptivo, sliding, lógica difusa, etc.
2. Para que la implementación en tiempo real sea más completa se podría agregar un efector final en el extremo del segundo eslabón, que podría ser una pinza rotórica la cual podría coger y soltar algún objeto. De esta manera los diseños de sistemas de control serían más completos y podrían ser orientados a aplicaciones industriales.

Bibliografía

- [1] A.Barrientos, L. Peñín, C. Balaguer R. Aracil. *Fundamentos de Robótica*. McGraw Hill, 1997.
- [2] Baturone, Anibal Ollero. *Robótica Manipuladores y robots móviles*. Alfaomega marcombo, 2001.
- [3] Corke, Peter I. *Robotics Toolbox for MATLAB*. Pinjarra Hills, 2001.
- [4] F. Torres, J. Pomares, P. Gil S. Puente R. Aracil. *Robot y Sistemas Sensoriales*. Prentice Hall, 2002.
- [5] Ferdinand P. Berr, E. Russell Johnston. *Dinámica*. McGraw Hill, 1996.
- [6] K. S. Fu, R. C. González, C.S.G. Lee. *Robótica, Control, Detección, Visión, Inteligencia*. McGraw Hill, 1990.
- [7] Khalil, Hassan K. *NonLinear Systems*. Prentice Hall, 1996.
- [8] M., J. Díaz. *Mecánica Racional Estática*. Editorial Universo, 1993.
- [9] Moshen, Shahinpoor. *A Robot Engineering Textbook*. Harper-Row Publisher, 1987.
- [10] Nakamura, Shoichiro. *Análisis Numérico y Visualización Gráfica*. Prentice Hall, 1997.
- [11] Part Enander, Eva and Anders Sjoberg. *The Matlab 5 Handbook*. Addison Wesley, 1999.
- [12] Rojas, Arturo. *Control Avanzado*. Publicación Independiente, 2001.
- [13] Rojas, Arturo. *Applied MIMO NonLinear Control Real Time Implementations*. En construcción, 2002.

- [14] Slotine, Jean-Jacques E. *Applied NonLinear Control*. Prentice Hall, 1991.
- [15] U., José Angulo. *Robótica Práctica (Cuarta Edición)*. Editorial Paraninfo, 2000.

Apéndice A

Coeficiente de Fricción Viscosa

La obtención del coeficiente de fricción viscosa del segundo eslabón, tiene que ver con el modelamiento y compensación de la fricción de Coulomb. La fricción de Coulomb es el torque requerido para mantener el servomotor en movimiento y es una fricción que se opone al movimiento. Se ha comprobado experimentalmente que la fricción de Coulomb puede ser considerada una constante. El efecto de la fricción de Coulomb es modelado como un torque constante proporcional al signo de la velocidad, es decir:

$$T_{fc} = \begin{cases} -T_c & \dot{q} < 0 \\ T_c & \dot{q} > 0 \end{cases} \quad (\text{A.1})$$

donde T_{fc} es el torque de fricción de Coulomb, \dot{q} es la velocidad del eje del servomotor.

El modelo discreto para el servomotor DC ($L_a = 0$) considerando la fricción de Coulomb, se deducen a partir de las siguientes ecuaciones:

$$T_m = J_{eq}\dot{w} + B_{eq}w + f^*(nw) \quad (\text{A.2})$$

$$T_m = K_m I_a \quad (\text{A.3})$$

$$I_a = \frac{VK_A}{R_a} - \frac{nK_b w}{R_a} f^*(nw) \quad (\text{A.4})$$

Reemplazamos (A.3) en (A.4), luego igualamos con (A.2)

$$J_{eq}\dot{w} = \frac{VK_A K_m}{R_a} - (B_{eq} + \frac{nK_b K_m}{R_a})w - f^*(nw) \quad (\text{A.5})$$

Las señales en tiempo discreto surgen si el sistema involucra la operación de muestreo de señales continuas. Sea w_k la velocidad angular muestreada en el k -ésimo instante de tiempo, usamos ahora la aproximación de Euler:

$T_s \dot{w} = w_{k+1} - w_k$; para discretizar la ecuación (A.5), obteniendo:

$$w_{k+1} = [1 - \frac{T_s}{J_{eq}} (B_{eq} + \frac{nK_m K_b}{R_a})] w_k + \frac{T_s K_m K_A}{J_{eq} R_a} [V_{ak} - \frac{R_a f^*(nw_k)}{K_m K_A}] \quad (\text{A.6})$$

donde $f^*(nw_k)$ modela el torque de fricción de Coulomb; y podemos asumir que la aproximación es dada por $f^*(nw_k) = V_{zm} \text{sgn}(nw)$, (V_{zm} es la mitad del ancho

Tabla A.1: Parámetros del Sistema a simular

Parámetro	Descripción	Valor
G_1	$1 - (T_s/J_{eq})(B_{eq} + nK_mK_b/R_a)$	0.8488
G_2	$K_mK_A T_s/(R_a J_{eq})$	0.5661
G_3	$-C_c R_a/(K_m K_A)$	-0.0869
G_4	n	12.5
T_s		0.01

La gráfica de la velocidad angular en estado estacionario del servomotor DC vs. el voltage aplicado al amplificador de potencia, se muestra en la figura A.3, donde nos permite apreciar una zona muerta a la velocidad cero. Esta zona muerta es debido a la fricción estática del servomotor. Note el efecto no lineal en las cercanías de la zona muerta.

Para linealizar la ecuación (A.5) se debe de adicionar una tensión de entrada a la entrada del amplificador tal que compense la fricción de Coulomb cuando la velocidad del servomotor $w_M \geq w_{th}$ (w_{th} velocidad umbral). Un esquema general

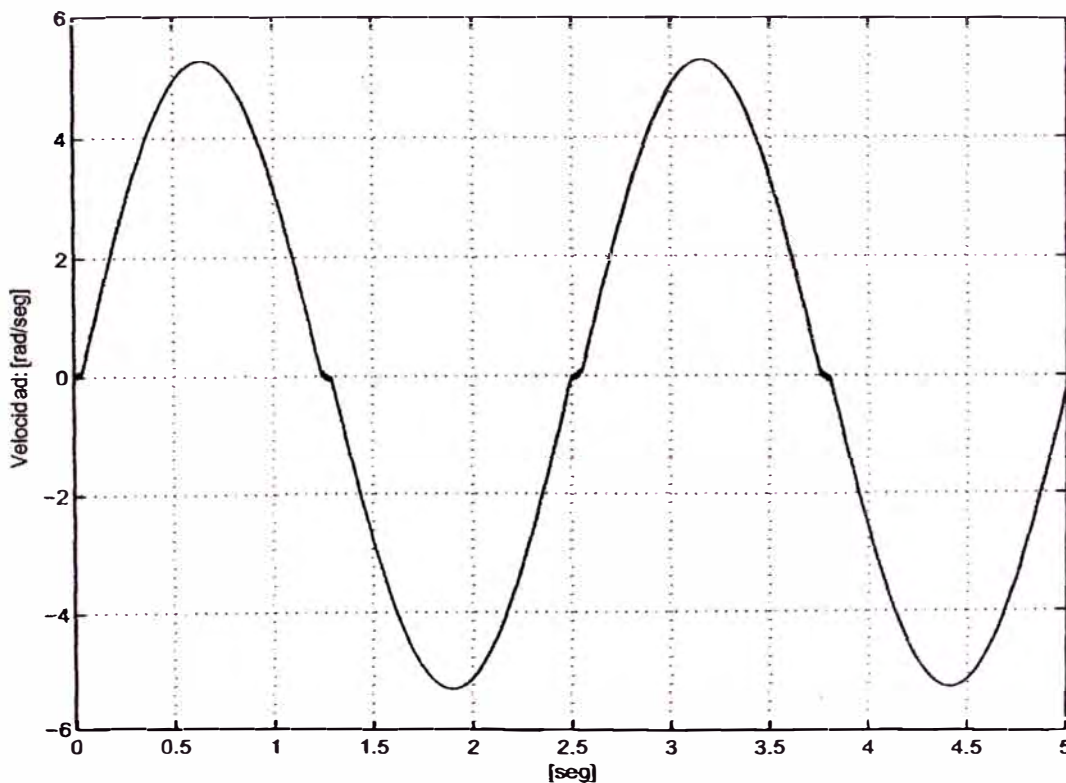


Figura A.2: Simulación para obtener la Fricción de Coulomb.

es el que presenta la compensación de las fricciones estática y de Coulomb.

$$u = v_p + V_c \tag{A.8}$$

siendo u la ley de control, v_p el voltage de compensación de la fricción estática y V_c el voltage de compensación de Coulomb. El objetivo del primer término (dependiente de la dirección del voltage de armadura) es sacar al servomotor de la zona donde la fricción estática tiene efecto. El objetivo del segundo término es compensar el torque de fricción de Coulomb presente cuando el servomotor está en movimiento a velocidades mayores al umbral w_{th} .

El voltage de compensación de Coulomb es dado por la relación:

$$V_c = \frac{R_a C_c}{K_m K_A} \text{sgn}(w_M) \quad \text{si } w_M \geq w_{th} \tag{A.9}$$

El coeficiente de fricción viscosa C_c es pequeño para servomotores DC, siendo difícil su identificación en presencia de ruido. Usamos el método de prueba y error para el cálculo de V_c , según la figura A.3, el valor inicial de V_{zm} es 0.025 este valor fue disminuído hasta encontrar el valor apropiado de $V_{zm} = 0.00868$.

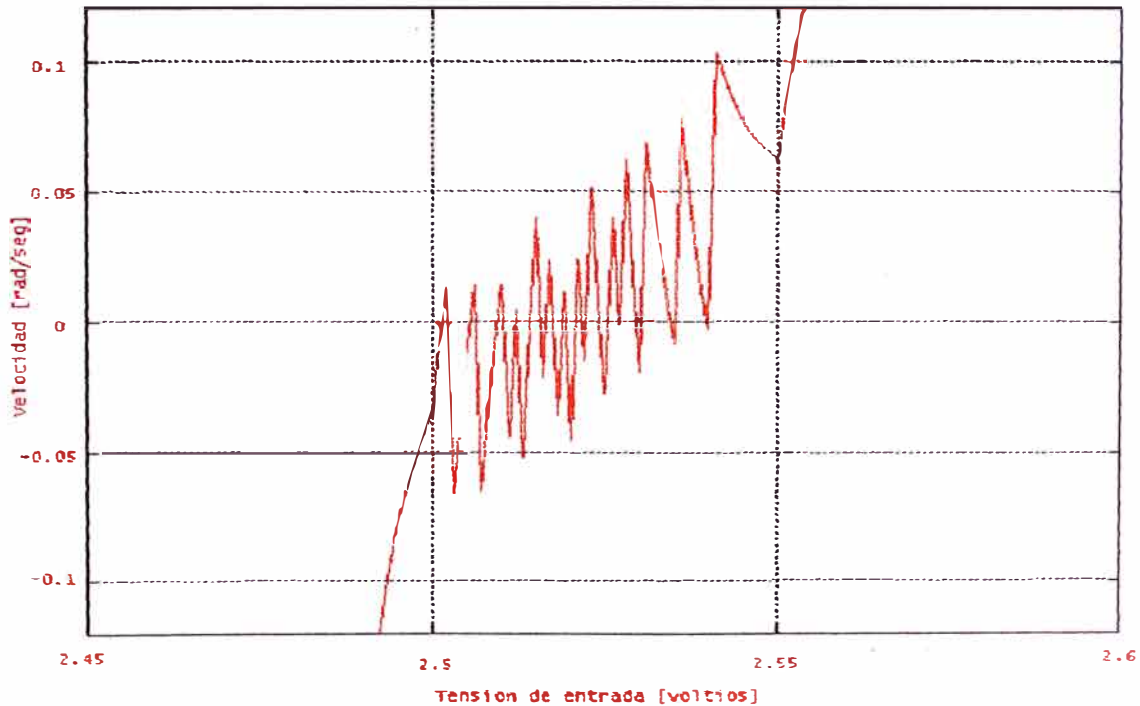


Figura A.3: Respuesta tensión de entrada versus velocidad.

Despejamos el coeficiente de fricción C_c de la ecuación A.9 para reemplazar los parámetros previamente identificados, como son:

$$R_a = 5.3 \quad \Omega$$

$$K_A = 2.17$$

$$K_m = 0.04217 \quad N - m/A$$

luego:

$$C_c = \frac{V_{zm} K_m K_A}{R_a} = \frac{0.00868(0.04217)(2.17)}{5.3} = 1.5e^{-4} \text{kgm}^2/\text{seg} \quad (\text{A.10})$$

El valor $V_{zm} = 0.00868$ permite compensar la fricción de Coulomb resultando el coeficiente de fricción de valor $C_c = 1.5e^{-4}$. Los efectos de compensación son mostrados en la figura A.4:

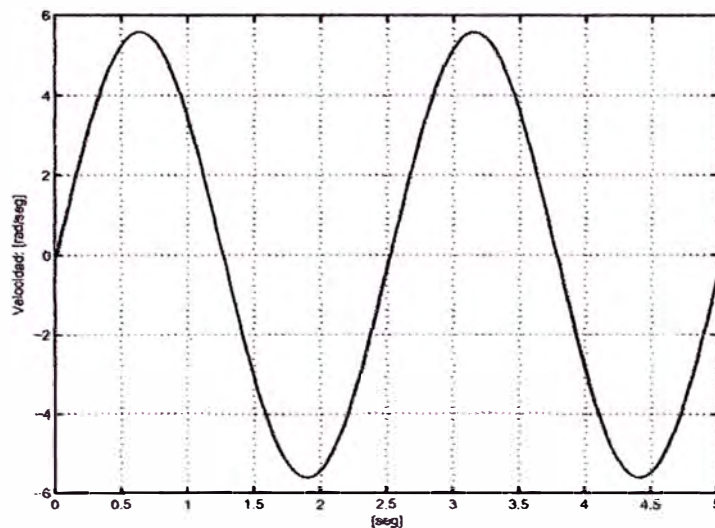


Figura A.4: Experimento de la compensación de la fricción no lineal.

Apéndice B

Programas en MATLAB

B.1 Listado de Programas en MATLAB

Los programas de simulación están escritos en código MATLAB [10], [11], y son los siguientes:

1. *modelo_r3.m*: este programa es usado en la sección 3.4.2, para determinar la ecuación dinámica del manipulador de 3DOF, mediante el algoritmo **Lagrange-Euler**.
2. *r3lex_p32.m*: este programa es usado en la sección 5.1, para realizar las simulaciones con trayectoria constante.
3. *r3lex_t3.m*: este programa es usado en la sección 5.2, para realizar las simulaciones con trayectoria senoidal.

B.2 Programa del Cálculo de la Ecuación Dinámica del Manipulador

```
%Archivo: modelo_r3.m
```

```
%Cálculo de la Ecuación Dinámica del Robot Manipulador Esférico de 3DOF aplicando el Algoritmo Lagrange-Euler con Matemática Simbólica de MATLAB.
```

```
syms q1 q2 q3 m1 hp hd rm2 hd Lys1 m2 L2 Lcs2 Lxs2 m3 L3 Lcs3 Lxs3 g g1 I1 I1 I2 Dq1 Dq2 Dq3 X Y S1 C1 S2 C2 TT1
```

```
%A.- Datos de los parámetros D-H (DENAVID-HARTEMBERG)
```

```
teta1 = q1; teta2 = q2; teta3 = q3;
```

```
dDH1 = hp + hd + rm2; dDH2 = 0; dDH3 = 0;
```

```
aDH1 = 0; aDH2 = L2; aDH3 = L3;
```

```
alfa1 = 90; alfa2 = 0; alfa3 = 0;
```

```
%B.- Cálculo de las matrices de transformación:  $A_{ij}$ 
```

```

A01 = [cos(q1)  0  sin(q1)  0
sin(q1)  0  -cos(q1)  0
0  1  0  hp + hd + rm2
0  0  0  1];
A12 = [cos(q2)  -sin(q2)  0  L2 * cos(q2)
sin(q2)  cos(q2)  0  L2 * sin(q2)
0  0  1  0
0  0  0  1];
A02 = A01 * A12;
A23 = [cos(q3)  -sin(q3)  0  L3 * cos(q3)
sin(q3)  cos(q3)  0  L3 * sin(q3)
0  0  1  0
0  0  0  1];
A03 = A02 * A23;
%X=cos(q2)*cos(q3)-sin(q2)*sin(q3)=cos(q2+q3);
%Y=sin(q2)*cos(q3)+sin(q3)*cos(q2)=sin(q2+q3);
%C.-Cálculo de las matrices de transformación homogénea: Uij
U11 = diff(A01,'q1');
U12 = diff(A01,'q2');
U13 = diff(A01,'q3');
U21 = diff(A02,'q1');
U22 = diff(A02,'q2');
U23 = diff(A02,'q3');
U31 = diff(A03,'q1');
U32 = diff(A03,'q2');
U33 = diff(A03,'q3');
A = -sin(q1) * L3 * X - sin(q1) * L2 * cos(q2);
B = cos(q1) * L3 * X + cos(q1) * L2 * cos(q2);
U31 = [-sin(q1) * X  sin(q1) * Y  cos(q1)  A
cos(q1) * X  -cos(q1) * Y  sin(q1)  B
0  0  0  0
0  0  0  0];
%U31 = [-S1 * X  S1 * Y  C1  -S1 * L3 * X - S1 * L2 * C2
%C1 * X  -C1 * Y  S1  C1 * L3 * X + C1 * L2 * C2

```



```

%0 0 0 0
%0 0 0 0];
U32 = [-cos(q1)*Y -cos(q1)*X 0 -L3*cos(q1)*Y-L2*cos(q1)*sin(q2)
-sin(q1)*Y -sin(q1)*X 0 -L3*sin(q1)*Y -L2*sin(q1)*sin(q2)
X -Y 0 L3*X + L2*cos(q2)
0 0 0 0];
U33 = [-cos(q1)*Y -cos(q1)*X 0 -L3*cos(q1)*Y
-sin(q1)*Y -sin(q1)*X 0 -L3*sin(q1)*Y
X -Y 0 L3*X
0 0 0 0];
X = cos(q2)*cos(q3) - sin(q2)*sin(q3);
Y = sin(q2)*cos(q3) + sin(q3)*cos(q2);
UU31 = [-sin(q1)*X sin(q1)*Y cos(q1) A
cos(q1)*X -cos(q1)*Y sin(q1) B
0 0 0 0
0 0 0 0];
UU32 = [-cos(q1)*Y -cos(q1)*X 0 -L3*cos(q1)*Y-L2*cos(q1)*sin(q2)
-sin(q1)*Y -sin(q1)*X 0 -L3*sin(q1)*Y -L2*sin(q1)*sin(q2)
X -Y 0 L3*X + L2*cos(q2)
0 0 0 0];
U33 = [-cos(q1)*Y -cos(q1)*X 0 -L3*cos(q1)*Y
-sin(q1)*Y -sin(q1)*X 0 -L3*sin(q1)*Y
X -Y 0 L3*X
0 0 0 0];
%D.- Matrices de seudoinercias: JJi
%Para la base (disco más barra): JJ1
JJ1 = [I11 0 0 0
0 I1 0 -m1*Lys1
0 0 I11 0
0 -m1*Lys1 0 m1];
%Para la primera varilla: JJ2
Lxs2 = L2 - Lcs2;
JJ2 = [I2 0 0 -m2*Lxs2
0 0 0 0

```

```

0 0 0 0
-m2 * Lxs2 0 0 m2];
%Para la segunda varilla: JJ3
Lxs3 = L3 - Lcs3;
JJ3 = [I3 0 0 -m3 * Lxs3
0 0 0 0
0 0 0 0
-m3 * Lxs3 0 0 m3];
%E.- Cálculo de la Matriz de Inercia Simétrica: Hik(q)
%Matriz H11
H11 = simplify(trace(U11 * JJ1 * U11.)) + simple(simplify(trace...
(U21 * JJ2 * U21.))) + simplify(trace(UU31 * JJ3 * UU31.));
%Matriz H12
H12 = simplify(trace(U22 * JJ2 * U21.)) + trace(UU32 * JJ3 * UU31.));
%Matriz H13
H13 = simplify(trace(UU33 * JJ3 * UU31.));
%Matriz H21
H21 = simplify(trace(U21 * JJ2 * U22.)) + trace(UU31 * JJ3 * UU32.));
%Matriz H22
H22 = simplify(trace(U22 * JJ2 * U22.)) + simplify(trace...
(UU32 * JJ3 * UU32.));
%Matriz H23
H23 = simplify(trace(UU33 * JJ3 * UU32.));
%Matriz H31
H31 = simplify(trace(UU31 * JJ3 * UU33.));
%Matriz H32
H32 = simplify(trace(UU32 * JJ3 * UU33.));
%Matriz H33
H33 = simplify(trace(UU33 * JJ3 * UU33.));
%F.- Cálculo de las matrices: Uijk
U111 = diff(U11,'q1');
U112 = diff(U11,'q2');
U113 = diff(U11,'q3');
U121 = diff(U12,'q1');

```

$U122 = \text{diff}(U12, 'q2');$
 $U123 = \text{diff}(U12, 'q3');$
 $U131 = \text{diff}(U13, 'q1');$
 $U132 = \text{diff}(U13, 'q2');$
 $U133 = \text{diff}(U13, 'q3');$
 $U211 = \text{diff}(U21, 'q1');$
 $U212 = \text{diff}(U21, 'q2');$
 $U213 = \text{diff}(U21, 'q3');$
 $U221 = \text{diff}(U22, 'q1');$
 $U222 = \text{diff}(U22, 'q2');$
 $U223 = \text{diff}(U22, 'q3');$
 $U231 = \text{diff}(U23, 'q1');$
 $U232 = \text{diff}(U23, 'q2');$
 $U233 = \text{diff}(U23, 'q3');$
 $U311 = \text{diff}(U31, 'q1');$
 $U312 = \text{diff}(U31, 'q2');$
 $U313 = \text{diff}(U31, 'q3');$
 $U321 = \text{diff}(U32, 'q1');$
 $U322 = \text{diff}(U32, 'q2');$
 $U323 = \text{diff}(U32, 'q3');$
 $U331 = \text{diff}(U33, 'q1');$
 $U332 = \text{diff}(U33, 'q2');$
 $U333 = \text{diff}(U33, 'q3');$

%G.- Cálculo del vector: hikm

$h111 = \text{simplify}(\text{trace}(U111 * JJ1 * U11.)) + \text{simple}(\text{simplify}(\text{trace}...$
 $(U211 * JJ2 * U21.)));$
 $h112 = \text{simple}(\text{simplify}(\text{trace}(U212 * JJ2 * U21.)));$
 $hh113 = \text{simple}(\text{trace}(U313 * JJ3 * UU31.));$
 $h121 = \text{simple}(\text{simplify}(\text{trace}(U221 * JJ2 * U21.)));$
 $h122 = \text{simple}(\text{simplify}(\text{trace}(U222 * JJ2 * U21.)));$
 $h123 = \text{simple}(\text{simplify}(\text{trace}(U323 * JJ3 * U31.)));$
 $h131 = \text{simple}(\text{simplify}(\text{trace}(U331 * JJ3 * UU31.)));$
 $h132 = \text{simple}(\text{simplify}(\text{trace}(U332 * JJ3 * UU31.)));$
 $h133 = \text{simple}(\text{simplify}(\text{trace}(U333 * JJ3 * UU31.)));$

```

h211 = simple(simplify(trace(U211 * JJ2 * U22.')));
h212 = simple(simplify(trace(U212 * JJ2 * U22.')));
h213 = simple(simplify(trace(U313 * JJ3 * U32.')));
h221 = simple(simplify(trace(U221 * JJ2 * U22.')));
h222 = simple(simplify(trace(U222 * JJ2 * U22.')));
hh223 = simple(simplify(trace(U323 * JJ3 * UU32.')));
h231 = simple(simplify(trace(U331 * JJ3 * U32.')));
hh232 = simple(simplify(trace(U332 * JJ3 * UU32.')));
hh233 = simple(simplify(trace(U333 * JJ3 * UU32.')));
hh311 = simple(simplify(trace(U331 * JJ3 * UU33.')));
h312 = simple(simplify(trace(U312 * JJ3 * UU33.')));
h313 = simple(simplify(trace(U313 * JJ3 * UU33.')));
h321 = simple(simplify(trace(U321 * JJ3 * UU33.')));
hh322 = simple(simplify(trace(U322 * JJ3 * UU33.')));
h323 = simple(simplify(trace(U323 * JJ3 * UU33.')));
h331 = simple(simplify(trace(U331 * JJ3 * UU33.')));
h332 = simple(simplify(trace(U332 * JJ3 * UU33.')));
h333 = simple(simplify(trace(U332 * JJ3 * UU33.')));

```

$$X = \cos(q2) * \cos(q3) - \sin(q2) * \sin(q3);$$

$$Y = \sin(q2) * \cos(q3) + \sin(q3) * \cos(q2);$$

$$h113 = simple(hh113);$$

$$h223 = simple(simplify(hh223));$$

$$h232 = h223;$$

$$h233 = h232;$$

$$h311 = simple(simplify(hh311));$$

$$h322 = simple(simplify(hh322));$$

%H.- Cálculo del vector de fuerza Centrífuga y de Coriolis: Ci(q,dq)

$$C1 = simple(h111 * Dq1 * Dq1 + h112 * Dq1 * Dq2 + h113 * Dq1 * Dq3 + ...$$

$$h121 * Dq1 * Dq2 + h122 * Dq2 * Dq2 + h123 * Dq2 * Dq3 + ...$$

$$h131 * Dq1 * Dq3 + h132 * Dq2 * Dq3 + h133 * Dq3 * Dq3);$$

$$C2 = simple(h211 * Dq1 * Dq1 + h212 * Dq1 * Dq2 + h213 * Dq1 * Dq3 + ...$$

$$h221 * Dq1 * Dq2 + h222 * Dq2 * Dq2 + h223 * Dq2 * Dq3 + ...$$

$$h231 * Dq1 * Dq3 + h232 * Dq2 * Dq3 + h233 * Dq3 * Dq3);$$

$$C3 = simple(h311 * Dq1 * Dq1 + h312 * Dq1 * Dq2 + h313 * Dq1 * Dq3 + ...$$

$h321 * Dq1 * Dq2 + h322 * Dq2 * Dq2 + h323 * Dq2 * Dq3 + \dots$
 $h331 * Dq1 * Dq3 + h332 * Dq2 * Dq3 + h333 * Dq3 * Dq3);$

%I.- Cálculo del vector de fuerza gravitacional: di(q)

$r11 = [0; -Lys1; 0; 1];$

$r22 = [-Lxs2; 0; 0; 1];$

$r33 = [-Lxs3; 0; 0; 1];$

$g1 = [0 \ 0 \ -g \ 0];$

$d1 = -m1 * g1 * U11 * r11 - m2 * g1 * U21 * r22 - m3 * g1 * UU31 * r33;$

$d2 = simplify(-m2 * g1 * U22 * r22 - m3 * g1 * UU32 * r33);$

$d3 = simplify(-m3 * g1 * UU33 * r33);$

%J.- Parámetros del Sistema

$\%a_1 = I2 + 2m_2L_2L_{c_{s2}} - m_2L_2^2$

$\%a_2 = I3 + 2m_3L_3L_{c_{s3}} - m_3L_3^2$

$\%a_3 = m_3L_2^2$

$\%a_4 = m_3L_2L_{c_{s3}}$

$\%a_5 = m_2gL_{c_{s2}} + m_3gL_2$

$\%a_6 = m_3gL_{c_{s3}}$

%K.- Resultados

%MATRIZ DE INERCIA SIMETRICA

$\%H_{11} = a_1\cos^2(q_2) + a_2\cos^2(q_2 + q_3) + a_3\cos^2(q_2) + 2a_4\cos(q_2)\cos(q_2 + q_3)$

$\%H_{22} = a_1 + a_2 + a_3 + 2a_4\cos(q_3)$

$\%H_{23} = a_2 + a_4\cos(q_3)$

$\%H_{32} = H_{23}$

$\%H_{33} = a_3$

%VECTOR DE FUERZA CENTRIFUGA Y CORIOLIS

$\%C_1 = -a_1\sin(2q_2)\dot{q}_1\dot{q}_2 - 2(a_2\sin(q_2 + q_3)\cos(q_2 + q_3) + a_4\sin(q_2 + q_3)\cos(q_2))\dot{q}_1\dot{q}_3$

$\%C_2 = a_1\sin(q_2)\cos(q_2)\dot{q}_1^2 - 2a_4\sin(q_3)\dot{q}_2\dot{q}_3 - a_4\sin(q_3)\dot{q}_3^2$

$\%C_3 = (a_2\sin(q_2 + q_3)\cos(q_2 + q_3) + a_4\sin(q_2 + q_3)\cos(q_2))\dot{q}_1^2 + a_4\sin(q_3)\dot{q}_2^2$

%VECTOR DE FUERZA GRAVITACIONAL

$d1 = 0;$

$d2 = a_5 * \cos(q_2) + a_6 * \cos(q_2 + q_3);$

$d3 = a_6 * \cos(q_2 + q_3);$

B.3 Programa de Simulación con Trayectoria Constante

```
%Archivo: r3lex_p3.m
%Datos del Sistema
%Sub-Sistema Mecánico
%Base
%Disco
hd=0.01; Rd=0.07; %Altura y radio
%M_DISCOTOTAL=M_DISCO+M_BOCINASERVOMOTOR1=0.3+0.25
m_dt = 0.55;
%Barra
hp=0.21;%Altura
ap=0.044;%Ancho y largo de la sección barra
%M_BARRATOTAL=1.075
m_pt=1.075-0.072;
m_hueco=0.072;
R_hueco=0.063;
%Momento de Inercia de la base
I11 = ((m_pt * (hp2)) + (3 * m_dt * (hd2)))/12;
%Primera varilla
m2=0.25;%Masa varilla
L2=0.20;%Longitud total
Lcs2=0.10;%Long. del punto de ref. 2 al centro de gravedad
I2=m2*(L2*L2)/3;%Momento de inercia
%Segunda varilla
m3=0.25;%Masa del brazo
mo=0.025;%Masa del objeto
L3=0.20;%Longitud total
Lcs3=0.10;%Long. del punto de ref. 3 al centro de gravedad
Io=mo*(ap*ap/12);%Momento de Inercia del objeto
Iv=m3*(L3*L3)/3;%Momento de Inercia segunda varilla
I3=Iv+Io;%Inercia total
g =9.81;
```

%Datos del Sub-Sistema Eléctrico

Km1=0.04364; Kb1=0.04364; Ra1=3.5; Jm1=0.00059; n1=18.5; KA1=2.17;
Km2=0.04364; Kb2=0.04364; Ra2=3.5; Jm2=0.00059; n2=18.5; KA2=2.17;
Km3=0.04217; Kb3=0.04217; Ra3=5.3; Jm3=0.000238; n3=12.5; KA3=2.17;
Cc1=0.001488;
Cc3=0.00015;

%Carga

Jeq1=0.2177; Beq1=0.054;
Jeq2=0.3954; Beq1=0.0589;
Jeq3=0.3954; Beq1=0.0589;

%Equivalencia de Resistencias

RT1=Ra1/(n1*Km1*KA1);
RT2=Ra2/(n2*Km2*KA2);
RT3=Ra3/(n3*Km3*KA3);

%Equivalencia de N

NT1 = n1*Kb1/KA1;
NT2=n2*Kb2/KA2;
NT3=n3*Kb3/KA3;

%Parámetros del Robot

$a_1 = 2 * I_{11}$
 $a_2 = I_2 + 2m_2 L_2 L_{c_{s_2}} - m_2 L_2^2$
 $a_3 = I_3 + 2m_3 L_3 L_{c_{s_3}} - m_3 L_3^2$
 $a_4 = m_3 L_2^2$
 $a_5 = m_3 L_2 L_{c_{s_3}}$
 $a_6 = m_2 g L_{c_{s_2}} + m_3 g L_2; a_7 = m_3 g L_{c_{s_3}}$

%Modelo Lineal Equivalente del Proceso de 6to Orden (Continuo)

M=[0 1 0 0 0 0;0 0 0 0 0 0;0 0 0 1 0 0;0 0 0 0 0 0; 0 0 0 0 0 1; 0 0 0 0 0 0];
N=[0 0 0;1 0 0;0 0 0;0 1 0;0 0 0;0 0 1];
C=[1 0 0 0 0 0;0 0 0 1 0 0;0 0 0 0 1 0];

%Cálculo del rango

NN = [C' M' * C' (M')^2 * C' (M')^3 * C' (M')^4 * C' (M')^5 * C'];

rangoNN rank(NN);%RANGO DEL SISTEMA r=6

%Cálculo de ganancias del Sistema realimentado: K1, K2, K3, K4, K5, K6


```

polo1 =-8;
polo2 =-8;
aa=[1  -polo1];
bb=[1  -polo2];
pol1=conv(aa,bb);
K1=pol1(2);
K2=pol1(3);
polo3=-6;
polo4=-6;
aaa=[1  -polo3];
bbb=[1  -polo4];
pol22=conv(aaa,bbb);
K3=pol22(2);
K4=pol22(3);
polo5=-6;
polo6=-6;
aaaa=[1  -polo5];
bbbb=[1  -polo6];
pol33=conv(aaaa,bbbb);
K5=pol33(2);
K6=pol33(3);

```

%Condiciones y Parámetros Iniciales

```

p1(1)=0; p1(2)=0; p1(3)=0; p1(4)=0;
p2(1)=0; p2(2)=0; p2(3)=0; p2(4)=0;
p3(1)=0; p3(2)=0; p3(3)=0; p3(4)=0;
x1(1)=0; x1(2)=0; x1(3)=0; x1(4)=0;
x2(1)=0; x2(2)=0; x2(3)=0; x2(4)=0;
x3(1)=0; x3(2)=0; x3(3)=0; x3(4)=0;
x4(1)=0; x4(2)=0; x4(3)=0; x4(4)=0;
x5(1)=0; x5(2)=0; x5(3)=0; x5(4)=0;
x6(1)=0; x6(2)=0; x6(3)=0; x6(4)=0;

```

%Tiempo de Muestreo

```

Nm=2000; T =0.010;

```

%Lazo de Control

```

for k=4:Nm;
%Trayectorias Constantes deseadas: p1, p2, p3
p1(k)=pi/2;
p2(k)=2*pi/3;
p3(k)=pi;
%Transformación de Coordenadas
z1(k)=x1(k);
z2(k)=x4(k);
z3(k)=x2(k);
z4(k)=x5(k);
z5(k)=x3(k);
z6(k)=x6(k);
%%Parámetros del sistema
%Matriz de Inercias Hij:
H11(k)=a1+a2*cos(z3(k))*cos(z3(k))+a3*cos(z3(k)+z5(k))*cos(z3(k)+z5(k)) +...
+a4*cos(z3(k))*cos(z3(k))+2*a5*cos(z3(k))*cos(z3(k)+z5(k))
H22(k)=a2+a3+a4+2*a5*cos(z5(k));
H23(k)=a3+2*a5*cos(z5(k));
H32(k)=H23(k);
H33(k)=a3;
%Vector de Fuerzas Centrífuga y de Coriolis Ci:
C1(k)=-a2*z2(k)*z4(k)*sin(2*z3(k)) -2*a3*z2(k)*z6(k)*sin(z3(k)+z5(k))*...
cos(z3(k)+z5(k)) -2*a5*z2(k)*z6(k)*sin(z3(k)+z5(k))*cos(z3(k));
C2(k)=a2*z2(k)*z2(k)*sin(z3(k))*cos(z3(k))-a54*z6(k)*z6(k)*sin(z5(k)) -...
2*a5*z4(k)*z6(k)*sin(z5(k));
C3(k)=a3*z2(k)*z2(k)*sin(z3(k)+z5(k))*cos(z3(k)+z5(k)) +...
a5*z2(k)*z2(k)*cos(z3(k))*sin(z3(k)+z5(k)) +a5*z4(k)*z4(k)*sin(z5(k));
%Vector de Fuerza Gravitacional di:
d1(k)=0;
d2(k)=a6*cos(z3(k))+a7*cos(z3(k)+z5(k));
d3(k)=a7*cos(z3(k)+z5(k));
%Nuevas entradas equivalentes: v1, v2 y v3
v1(k)=K2*p1(k)-K1*z2(k)-K2*z1(k);
v2(k)=K4*p2(k)-K3*z4(k)-K4*z3(k);

```

```

v3(k)=K6*p3(k)-K5*z6(k)-K4*z5(k);
%%Ley de Control u1, u2 y u3
%Valores de P1, P2, P3
P1(k)=(RT1*Beq1+NT1)*z2(k)+RT1*C1(k);
P2(k)=(RT2*Beq2+NT2)*z4(k)+RT2*(C2(k)+d2(k));
P3(k)=(RT3*Beq3+NT3)*z6(k)+RT3*(C3(k)+d3(k));
%%Primer Actuador: base
u1(k) = P1(k) + v1(k)*RT1*(H11(k)+Jeq1);
%Compensando fricción de Coulomb
%u1(k) = u1(k) + Cc1*n1*RT1*x4(k);
if(u1(k) > z2(k))
u1(k) = u1(k) + Cc1*n1*RT1*z2(k);
elseif(u1(k) < -z2(k))
u1(k) = u1(k) - Cc1*n1*RT1*z2(k);
end
%Limitando fuerza de control
if(u1(k) > 10)
u1(k)=10;
elseif(u1(k) < -10)
u1(k)=-10;
end
%%Segundo Actuador: primera varilla
u2(k) = P2(k) + v2(k)*RT2*(H22(k)+Jeq2) + v3(k)*RT2*H23(k);
%Compensando fricción de Coulomb
%u2(k) = u2(k) + Cc1*n2*RT2*x5(k);
if(u2(k) > z4(k))
u2(k) = u2(k) + Cc1*n2*RT2*z4(k);
elseif(u2(k) < -z4(k))
u2(k) = u2(k) - Cc1*n2*RT2*z4(k);
end
%Limitando fuerza de control
if(u2(k) > 10)
u2(k) = 10;
elseif(u2(k) < -10)

```

```

u2(k) = -10;
end
%%Tercer Actuador: segunda varilla
u3(k) = P3(k) + v2(k)*RT3*H32(k) + v3(k)*RT3*(H33(k)+Jeq3);
%Compensando fricción de Coulomb
%u3(k) = u3(k) + Cc3*n3*RT3*x6(k);
if(u3(k) > z6(k))
u3(k) = u3(k) + Cc3*n3*RT3*z6(k);
elseif(u3(k) < -z6(k))
u3(k) = u3(k) - Cc3*n3*RT3*z6(k);
end
%Limitando fuerza de control
if(u3(k) > 10)
u3(k) = 10;
elseif(u3(k) < -10)
u3(k) = -10;
end
%Proceso de discretización
Tc1=Cc1*sign(x4(k));
Tc2=Cc1*sign(x5(k));
Tc3=Cc3*sign(x6(k));
x1(k+1)=x1(k)+T*x3(k);
x4(k+1)=x4(k)+(T*(-n1*RT1*Tc1-P1(k)+u1(k)))/(RT1*(H11(k)+Jeq1));
Det(k)=RT2*RT3*((H22(k)+Jeq2)*(H33(k)+Jeq3)-H23(k)*H23(k));
x2(k+1)=x2(k)+T*x5(k);
x5(k+1)=x5(k)+(T/Det(k))*(RT3*(H33(k)+Jeq3)*u2(k)-RT2*H23(k)*u3(k)-...
RT3*(H33(k)+Jeq3)*P2(k)+RT2*H23(k)*P3(k)-...
RT3*(H33(k)+Jeq3)*n2*RT2*Tc2+ RT2*H23(k)*n3*RT3*Tc3);
x3(k+1)=x3(k)+T*x6(k);
x6(k+1)=x6(k)+(T/Det(k))*(RT2*(H22(k)+Jeq2)*u3(k)-RT3*H32(k)*u2(k)-...
RT2*(H22(k)+Jeq2)*P3(k)+RT3*H32(k)*P2(k)-...
RT2*(H22(k)+Jeq2)*n3*RT3*Tc3+ RT3*H32(k)*n2*RT2*Tc2);
end
%Simulaciones

```

```

ejex = linspace(0,Nm*T,Nm-3);
figure(1)
subplot(3,1,1)
plot(ejex,p1(4:Nm)*180/pi,ejex,x1(4:Nm)*180/pi), grid on
%title('R-3DOF: POSICION DE LA BASE')
ylabel('[grados]')
subplot(3,1,2)
plot(ejex,p2(4:Nm)*180/pi,ejex,x2(4:Nm)*180/pi), grid on
%title('R-3DOF: POSICION PRIMERA VARILLA')
ylabel('[grados]')
subplot(3,1,3)
plot(ejex,p3(4:Nm)*180/pi,ejex,x3(4:Nm)*180/pi), grid on
%title('R-3DOF: POSICION SEGUNDA VARILLA')
ylabel('[grados]')
xlabel('Tiempo [Seg.]')
print -f -deps r3lex_py3
figure(2)
subplot(3,1,1)
plot(ejex,u1(4:Nm)), grid on
%title('R-3DOF: SEÑAL DE CONTROL DE LA BASE')
ylabel('[Voltios]')
subplot(3,1,2)
plot(ejex,u2(4:Nm)), grid on
%title('R-3DOF: SEÑAL DE CONTROL PRIMERA VARILLA')
ylabel('[Voltios] )')
subplot(3,1,3)
plot(ejex,u3(4:Nm)), grid on
%title('R-3DOF: SENAL DE CONTROL SEGUNDA VARILLA')
ylabel('[Voltios]')
xlabel('Tiempo [Seg.]')
print -f -deps r3lex_pu3
figure(3)
subplot(3,1,1)
plot(ejex,x4(4:Nm)), grid on

```

```
%title('R-3DOF: VELOCIDAD DE LA BASE')
ylabel('[Rad/Seg.]')
subplot(3,1,2)
plot(ejex,x5(4:Nm)), grid on
%title('R-3DOF: VELOCIDAD PRIMERA VARILLA')
ylabel('[Rad/Seg.]')
subplot(3,1,3)
plot(ejex,x6(4:Nm)), grid on
%title('R-3DOF: VELOCIDAD SEGUNDA VARILLA')
ylabel('[Rad/Seg.]')
xlabel('Tiempo [Seg.]')
print -f -deps r3lex_pv3
```

B.4 Programa de Simulación con Trayectoria Senoidal

%Archivo: r3lex_t3.m

%Datos del Sistema

%Sub-Sistema Mecánico

%Base

%Disco

hd=0.01; Rd=0.07; %Altura y radio

%M_DISCOTOTAL=M_DISCO+M_BOCINASERVOMOTOR1-0.3+0.25

m_dt = 0.55;

%Barra

hp=0.21;%Altura

ap=0.044;%Ancho y largo de la sección barra

%M_BARRATOTAL=1.075

m_pt=1.075-0.072;

m_hueco=0.072;

R_hueco=0.063;

%Inercia de la base

$I_{11} = ((m_{pt} * (ap^2)) + (3 * m_{dt} * (Rd^2)))/12;$

%Primera varilla

m2=0.25;%Masa varilla

L2=0.20;%Longitud total

Lcs2=0.10;%Long. del punto de ref. 2 al centro de gravedad

$I2 = m2 * (L2 * L2) / 3;$ %Momento de inercia

%Segunda varilla

m3=0.25;%Masa del brazo

mo=0.025;%Masa del objeto

L3=0.20;%Longitud total

Lcs3=0.10;%Long. del punto de ref. 3 al centro de gravedad

$I_o = m_o * (ap * ap / 12);$ %Momento de Inercia del objeto

$I_v = m_3 * (L_3 * L_3) / 3;$ %Momento de Inercia segunda varilla

I3=*Iv*+*Io*;%Inercia total

g =9.81;

%Datos del Sub-Sistema Eléctrico

$K_{m1}=0.04364$; $K_{b1}=0.04364$; $R_{a1}=3.5$; $J_{m1}=0.00059$; $n_1=18.5$; $KA_1=2.17$;
 $K_{m2}=0.04364$; $K_{b2}=0.04364$; $R_{a2}=3.5$; $J_{m2}=0.00059$; $n_2=18.5$; $KA_2=2.17$;
 $K_{m3}=0.04217$; $K_{b3}=0.04217$; $R_{a3}=5.3$; $J_{m3}=0.000238$; $n_3=12.5$; $KA_3=2.17$;
 $C_{c1}=0.001488$;
 $C_{c3}=0.00015$;

%Carga

$J_{eq1}=0.2177$; $Beq1=0.054$;
 $J_{eq2}=0.3954$; $Beq1=0.0589$;
 $J_{eq3}=0.3954$; $Beq1=0.0589$;

%Equivalencia de Resistencias

$RT1=R_{a1}/(n_1*K_{m1}*KA_1)$;
 $RT2=R_{a2}/(n_2*K_{m2}*KA_2)$;
 $RT3=R_{a3}/(n_3*K_{m3}*KA_3)$;

%Equivalencia de N

$NT1=n_1*K_{b1}/KA_1$;
 $NT2=n_2*K_{b2}/KA_2$;
 $NT3=n_3*K_{b3}/KA_3$;

%Parámetros del Robot

$a_1 = 2 * I_{11}$
 $a_2 = I_2 + 2m_2L_2L_{c_{s2}} - m_2L_2^2$
 $a_3 = I_3 + 2m_3L_3L_{c_{s3}} - m_3L_3^2$
 $a_4 = m_3L_2^2$
 $a_5 = m_3L_2L_{c_{s3}}$
 $a_6 = m_2gL_{c_{s2}} + m_3gL_2$; $a_7 = m_3gL_{c_{s3}}$

%Modelo Lineal Equivalente del Proceso de 6to Orden (Continuo)

$M=[0 \ 1 \ 0 \ 0 \ 0 \ 0; 0 \ 0 \ 0 \ 0 \ 0 \ 0; 0 \ 0 \ 0 \ 1 \ 0 \ 0; 0 \ 0 \ 0 \ 0 \ 0 \ 0; 0 \ 0 \ 0 \ 0 \ 0 \ 1; 0 \ 0 \ 0 \ 0 \ 0 \ 0]$;
 $N=[0 \ 0 \ 0; 1 \ 0 \ 0; 0 \ 0 \ 0; 0 \ 1 \ 0; 0 \ 0 \ 0; 0 \ 0 \ 1]$;
 $C=[1 \ 0 \ 0 \ 0 \ 0 \ 0; 0 \ 0 \ 0 \ 1 \ 0 \ 0; 0 \ 0 \ 0 \ 0 \ 1 \ 0]$;

%Cálculo del rango

$NN = [C' \ M' * C' \ (M')^2 * C' \ (M')^3 * C' \ (M')^4 * C' \ (M')^5 * C']$;
 $rangoNN=rank(NN)$; **%RANGO DEL SISTEMA r=6**

%Cálculo de ganancias del Sistema realimentado: K1, K2, K3, K4, K5, K6

$polo1 = -8$;

```

polo2 =-8;
aa=[1  -polo1];
bb=[1  -polo2];
pol1=conv(aa,bb);
K1=pol1(2);
K2=pol2(3);
polo3=-6;
polo4=-6;
aaa=[1  -polo3];
bbb=[1  -polo4];
pol22=conv(aaa,bbb);
K3=pol22(2);
K4=pol22(3);
polo5=-6;
polo6=-6;
aaaa=[1  -polo5];
bbbb=[1  -polo6];
pol33=conv(aaaa,bbbb);
K5=pol33(2);
K6=pol33(3);
%Condiciones y Parámetros Iniciales
p1(1) =0; p1(2) =0; p1(3) =0; p1(4) =0;
dp1(1) =0; dp1(2) =0; dp1(3) =0; dp1(4) =0;
ddp1(1) =0; ddp1(2) =0; ddp1(3) =0; ddp1(4) =0;
p2(1) =0; p2(2) =0; p2(3) =0; p2(4) =0;
dp2(1) =0; dp2(2) =0; dp2(3) =0; dp2(4) =0;
ddp2(1) =0; ddp2(2) =0; ddp2(3) =0; ddp2(4) =0;
p3(1) =0; p3(2) =0; p3(3) =0; p3(4) =0;
dp3(1) =0; dp3(2) =0; dp3(3) =0; dp3(4) =0;
ddp3(1) =0; ddp3(2) =0; ddp3(3) =0; ddp3(4) =0;
x1(1)=0;x1(2)=0;x1(3)=0;x1(4)=0;
x2(1)=0;x2(2)=0;x2(3)=0;x2(4)=0;
x3(1)=0;x3(2)=0;x3(3)=0;x3(4)=0;
x4(1) =0;x4(2)=0;x4(3)=0;x4(4)=0;

```

```

x5(1)=0; x5(2)=0; x5(3)=0; x5(4)=0;
x6(1)=0; x6(2)=0; x6(3)=0; x6(4)=0;
%Tiempo de Muestreo
Nm=2000; T =0.010;
%Lazo de Control
for k=4:Nm;
%Trayectorias Senoidales deseadas: p1, p2 y p3
p1(k)=pi*cos(2pikT/11);
p2(k)=3*pi/2*cos(2pikT/11);
p3(k)=2*pi*cos(2pikT/11);
%Transformación de Coordenadas
z1(k)=x1(k);
z2(k)=x4(k);
z3(k)=x2(k);
z4(k)=x5(k);
z5(k)=x3(k);
z6(k)=x6(k);
%Aproximación de derivadas para discretización
dp1(k)=(p1(k)+3*p1(k-1)-3*p1(k-2)-p1(k-3))/(6*T);
ddp1(k)=(dp1(k)+3*dp1(k-1)-3*dp1(k-2)-dp1(k-3))/(6*T);
dp2(k)=(p2(k)+3*p2(k-1)-3*p2(k-2)-p2(k-3))/(6*T);
ddp2(k)=(dp2(k)+3*dp2(k-1)-3*dp2(k-2)-dp2(k-3))/(6*T);
dp3(k)=(p3(k)+3*p3(k-1)-3*p3(k-2)-p3(k-3))/(6*T);
ddp3(k)=(dp3(k)+3*dp3(k-1)-3*dp3(k-2)-dp3(k-3))/(6*T);
%%Parámetros del sistema
%Matriz de Inercias Hij:
H11(k)=a1+a2*cos(z3(k))*cos(z3(k))+a3*cos(z3(k)+z5(k))*cos(z3(k)+z5(k)) +...
+a4*cos(z3(k))*cos(z3(k))+2*a5*cos(z3(k))*cos(z3(k)+z5(k))
H22(k)=a2+a3+a4+2*a5*cos(z5(k));
H23(k)=a3+2*a5*cos(z5(k));
H32(k)=H23(k);
H33(k)=a4;
%Vector de Fuerzas Centrífuga y de Coriolis Ci:
C1(k)=-a2*z2(k)*z4(k)*sin(2*z3(k))-2*a3*z2(k)*z6(k)*sin(z3(k)+z5(k))*...

```

```

cos(z3(k)+z5(k)) -2*a5*z2(k)*z6(k)*sin(z3(k)+z5(k))*cos(z3(k));
C2(k)=a2*z2(k)*z2(k)*sin(z3(k))*cos(z3(k))-a5*z6(k)*z6(k)*sin(z5(k)) -...
2*a5*z4(k)*z6(k)*sin(z5(k));
C3(k)=a3*z2(k)*z2(k)*sin(z3(k)+z5(k))*cos(z3(k)+z5(k)) +...
a5*z2(k)*z2(k)*cos(z3(k))*sin(z3(k)+z5(k)) +a5*z4(k)*z4(k)*sin(z5(k));
%Vector de Fuerza Gravitacional di:
d1(k)=0;
d2(k)=a6*cos(z3(k))+a7*cos(z3(k)+z5(k));
d3(k)=a7*cos(z3(k)+z5(k));
%Nuevas entradas equivalentes: v1, v2 y v3
v1(k)=ddp1(k)+K1*dp1(k)+K2*p1(k)-K1*z2(k)-K2*z1(k);
v2(k)=ddp2(k)+K3*dp2(k)+K4*p2(k)-K3*z4(k)-K4*z3(k);
v3(k)=ddp3(k)+K5*dp3(k)+K6*p3(k)-K5*z6(k)-K4*z5(k);
%Ley de Control u1, u2 y u3
%Valores de P1, P2, P3
P1(k)=(RT1*Beq1+NT1)*z2(k)+RT1*C1(k);
P2(k)=(RT2*Beq2+NT2)*z4(k)+RT2*(C2(k)+d2(k));
P3(k)=(RT3*Beq3+NT3)*z6(k)+RT3*(C3(k)+d3(k));
%%Primer Actuador: base
u1(k) = P1(k) + v1(k)*RT1*(H11(k)+Jeq1);
%Compensando fricción de Coulomb
%u1(k) = u1(k) + Cc1*n1*RT1*x4(k);
if(u1(k) > z2(k))
u1(k) = u1(k) + Cc1*n1*RT1*z2(k);
elseif(u1(k) < -z2(k))
u1(k) = u1(k) - Cc1*n1*RT1*z2(k);
end
%Limitando fuerza de control
if(u1(k) > 10)
u1(k)=10;
elseif(u1(k) < -10)
u1(k)=-10;
end
%%Segundo Actuador: primera varilla

```

```

u2(k) = P2(k) + v2(k)*RT2*(H22(k)+Jeq2) + v3(k)*RT2*H23(k);
%Compensando fricción de Coulomb
%u2(k) = u2(k) + Cc1*n2*RT2*x5(k);
if(u2(k) > z4(k))
u2(k) = u2(k) + Cc1*n2*RT2*z4(k);
elseif(u2(k) < -z4(k))
u2(k) = u2(k) - Cc1*n2*RT2*z4(k);
end
%Limitando fuerza de control
if(u2(k) > 10)
u2(k) = 10;
elseif(u2(k) < -10)
u2(k) = -10;
end
%%Tercer Actuador: segunda varilla
u3(k) = P3(k) + v2(k)*RT3*H32(k) + v3(k)*RT3*(H33(k)+Jeq3);
%Compensando fricción de Coulomb
%u3(k) = u3(k) + Cc3*n3*RT3*x6(k);
if(u3(k) > z6(k))
u3(k) = u3(k) + Cc3*n3*RT3*z6(k);
elseif(u3(k) < -z6(k))
u3(k) = u3(k) - Cc3*n3*RT3*z6(k);
end
%Limitando fuerza de control
if(u3(k) > 10)
u3(k) = 10;
elseif(u3(k) < -10)
u3(k) = -10;
end
%Proceso de discretización
Tc1=Cc1*sign(x4(k));
Tc2=Cc1*sign(x5(k));
Tc3=Cc3*sign(x6(k));
x1(k+1)-x1(k)+T*x3(k);

```

```

x4(k+1)=x4(k)+(T*(-n1*RT1*Tc1-P1(k)+u1(k)))/(RT1*(H11(k)+Jeq1));
Det(k)=RT2*RT3*((H22(k)+Jeq2)*(H33(k)+Jeq3)-H23(k)*H23(k));
x2(k+1)=x2(k)+T*x5(k);
x5(k+1)=x5(k)+(T/Det(k))*(RT3*(H33(k)+Jeq3)*u2(k)-RT2*H23(k)*u3(k)-...
RT3*(H33(k)+Jeq3)*P2(k)+RT2*H23(k)*P3(k)-...
RT3*(H33(k)+Jeq3)*n2*RT2*Tc2+ RT2*H23(k)*n3*RT3*Tc3);
x3(k+1)=x3(k)+T*x6(k);
x6(k+1)=x6(k)+(T/Det(k))*(RT2*(H22(k)+Jeq2)*u3(k)-RT3*H32(k)*u2(k)-...
RT2*(H22(k)+Jeq2)*P3(k)+RT3*H32(k)*P2(k)-...
RT2*(H22(k)+Jeq2)*n3*RT3*Tc3+ RT3*H32(k)*n2*RT2*Tc2);
end
%Simulaciones
ejex = linspace(0,Nm*T,Nm-3);
figure(1)
subplot(3,1,1)
plot(ejex,p1(4:Nm)*180/pi,ejex,x1(4:Nm)*180/pi), grid on
%title('R-3DOF: POSICION DE LA BASE')
ylabel('[grados]')
subplot(3,1,2)
plot(ejex,p2(4:Nm)*180/pi,ejex,x2(4:Nm)*180/pi), grid on
%title('R-3DOF: POSICION PRIMERA VARILLA')
ylabel('[grados]')
subplot(3,1,3)
plot(ejex,p3(4:Nm)*180/pi,ejex,x3(4:Nm)*180/pi), grid on
%title('R-3DOF: POSICION SEGUNDA VARILLA')
ylabel('[grados]')
xlabel('Tiempo [Seg.]')
print -f -deps r3lex_ty3
figure(2)
subplot(3,1,1)
plot(ejex,u1(4:Nm)), grid on
%title('R-3DOF: SEÑAL DE CONTROL DE LA BASE')
ylabel('[Voltios]')
subplot(3,1,2)

```

```

plot(ejex,u2(4:Nm)), grid on
%title('R-3DOF: SEÑAL DE CONTROL PRIMERA VARILLA')
ylabel('[Voltios]')
subplot(3,1,3)
plot(ejex,u3(4:Nm)), grid on
%title('R-3DOF: SEÑAL DE CONTROL SEGUNDA VARILLA )
ylabel('[Voltios]')
xlabel('Tiempo [Seg.]')
print -f -deps r3lex_tu3
figure(3)
subplot(3,1,1)
plot(ejex,x4(4:Nm)), grid on
%title('R-3DOF: VELOCIDAD DE LA BASE')
ylabel('[Rad/Seg.]')
subplot(3,1,2)
plot(ejex,x5(4:Nm)), grid on
%title('R-3DOF: VELOCIDAD PRIMERA VARILLA')
ylabel('[Rad/Seg.]')
subplot(3,1,3)
plot(ejex,x6(4:Nm)), grid on
%title('R-3DOF: VELOCIDAD SEGUNDA VARILLA')
ylabel('[Rad/Seg.]')
xlabel('Tiempo [Seg.]')
print -f -deps r3lex_tv3

```


Apéndice C

Ejemplo: Manipulador de 2DOF

En este apéndice se presentan los resultados experimentales en tiempo real de la implementación del sistema robot manipulador de 2 grados de libertad (DOF) que se muestra en la figura C.1 y figura C.2, que ha sido realizado por mi persona para la elaboración de mi Tesis de Maestría desde el año 2002 al 2003, en el Laboratorio de la Sección de Postgrado de la Facultad de Ingeniería Eléctrica y Electrónica (FIEE). No se pudo realizar la implementación en tiempo real del manipulador de 3 grados de libertad (DOF) debido a que no se contaba con la Interface de Datos de más de dos salidas análogas.

El programa que he usado para controlar las trayectorias del manipulador de 2DOF en tiempo real es el Borland C++, y la técnica de control no lineal multivariable empleada es la de realimentación de estado, así como también pueden usarse otros algoritmos no lineales como deslizante, adaptivo, óptimo, etc.

Para obtener los resultados realicé dos experimentos: una para trayectoria constante y otra para trayectoria arbitraria para lo cual he usado una función senoidal.

Los datos experimentales como posición, señal de control y velocidad, que he obtenido del programa en C++ para ambas trayectorias, son ploteados en MATLAB de donde se obtiene los gráficos respectivos que se mostrarán más adelante, los cuales nos muestran el control de las trayectorias del manipulador.

C.1 Primer experimento: Control de Trayectoria Constante

Para este primer experimento he usado una trayectoria deseada constante con los siguientes datos:

- Trayectorias deseadas:

$$p_1 = 3\pi/2, \text{ (1DOF: posición angular de la base)}$$

$$p_2 = \pi, \text{ (2DOF: posición angular del brazo)}$$



Figura C.1: Implementación real del sistema.

- Polos usados:
para la base: $-7.5 + 1.9365i$ y $-7.5 - 1.9365i$
para el brazo: -7.5 y -8

Los resultados obtenidos para este experimento se muestran desde la figura C.2 hasta la figura C.4.

En la figura C.2 se muestra las salidas controladas para la posición angular de la base y posición angular del brazo, el tiempo de establecimiento es de 7 seg. para la base y 5 seg. para el brazo. Se observa que en la trayectoria de la base y del brazo no existe sobreimpulso. Se puede observar que en ambos casos la trayectoria desde el inicio hasta llegar al tiempo de establecimiento presenta ciertas oscilaciones.

En la figura C.3 se muestra las señales de control para ambos grados de libertad. La señal de control para el servomotor de la base llega hasta 3 voltios, presentando una oscilación por 7 seg. para luego converger aproximadamente a cero. Para el servomotor del brazo la señal llega hasta 3 volios presentando una oscilación por 7 seg. para luego converger aproximadamente a cero.

En la figura C.4 se muestra la respuesta al sistema para la velocidad angular del servomotor de la base y el brazo. En el gráfico se aprecia que en el arranque la velocidad en ambos se incrementa, para luego llegar a converger a cero a los 7 seg para la base y a los 7 seg para el brazo. Para ambos casos, la velocidad angular desde el inicio hasta converger a cero presenta oscilaciones abruptas.

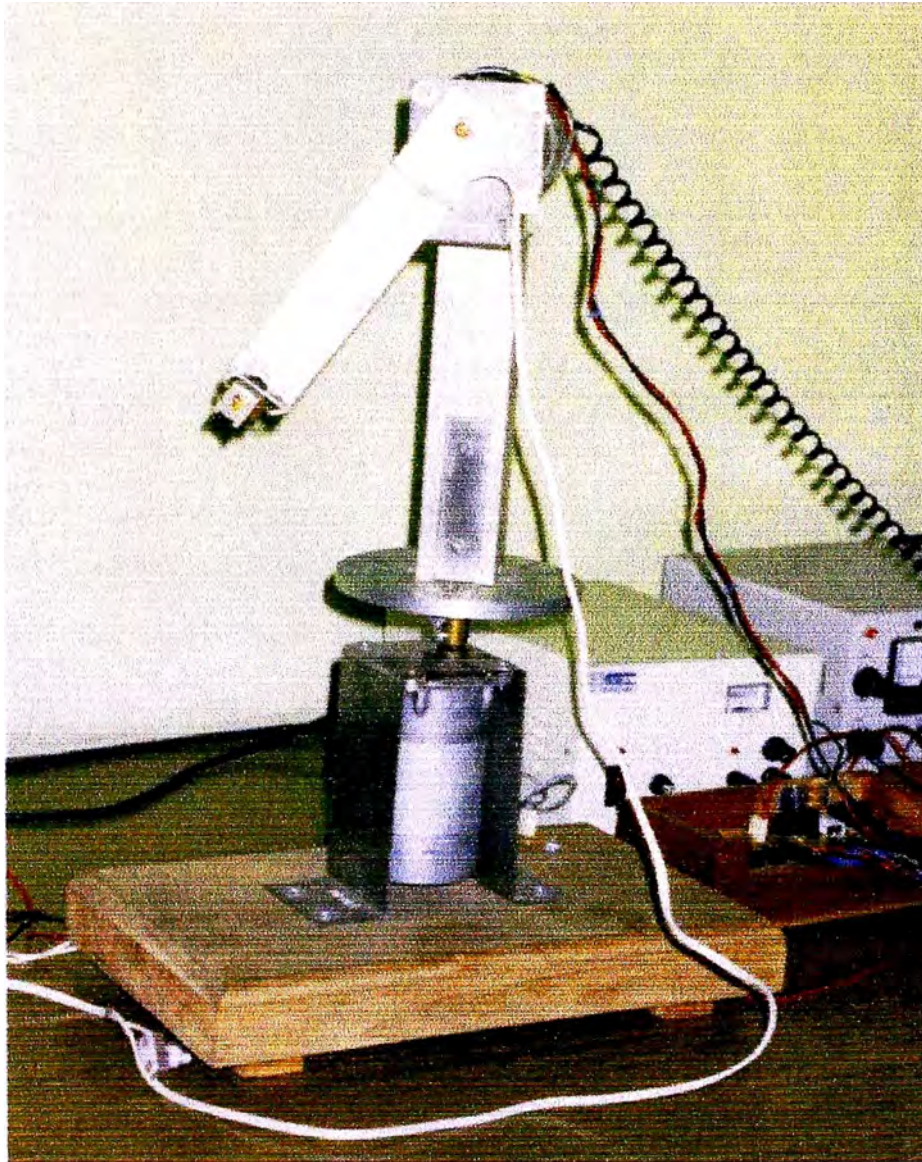


Figura C.2: Implementación real de la planta.

C.2 Segundo Experimento: Control de Trayectoria Arbitraria

Para este segundo experimento he usado una trayectoria deseada arbitraria tomando una función senoidal con los siguientes datos:

- Trayectorias deseadas:

$$p_1 = \pi/2\cos(2\pi kT/21), \text{ (1DOF: trayectoria senoidal de la base)}$$

$$p_2 = \pi/2\cos(2\pi kT/21), \text{ (2DOF: trayectoria senoidal del brazo)}$$

- Polos usados:

$$\text{para la base: } -7.5 + 2.7839i \text{ y } -7.5 - 2.7839i$$

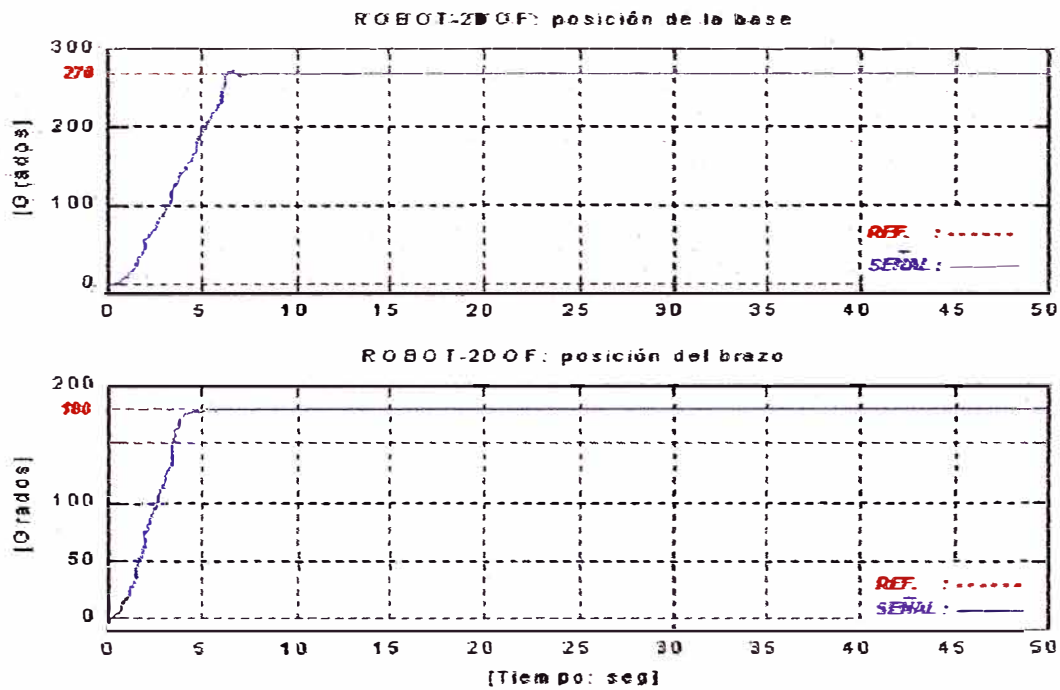


Figura C.3: Trayectoria Constante Experimental: salidas controladas para la base y el brazo.

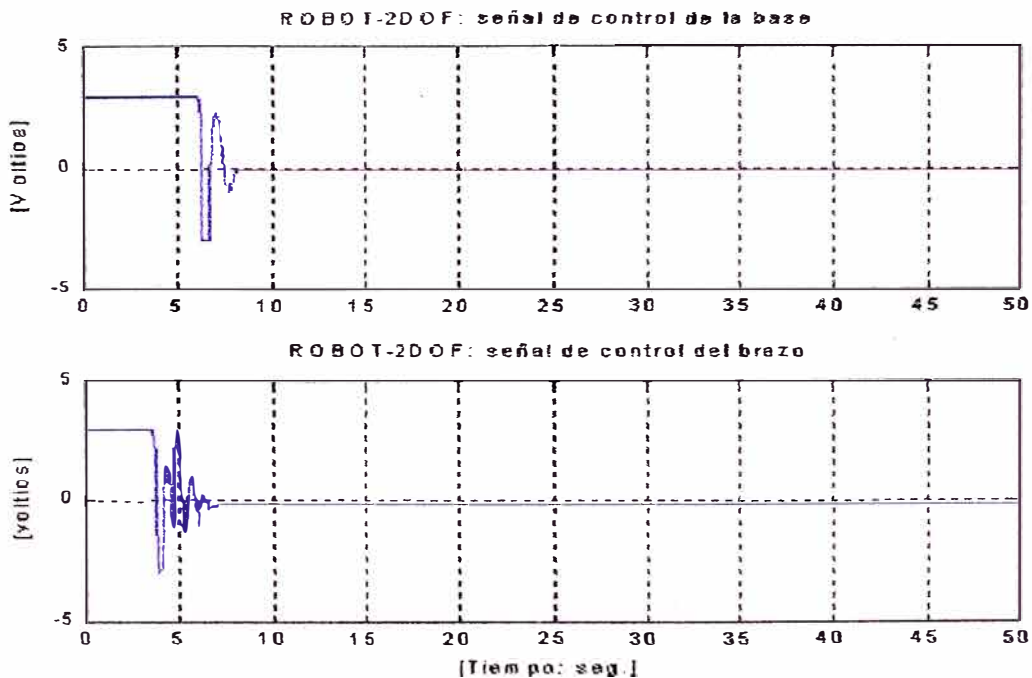


Figura C.4: Trayectoria Constante Experimental: señal de control para el servomotor de la base y el brazo.

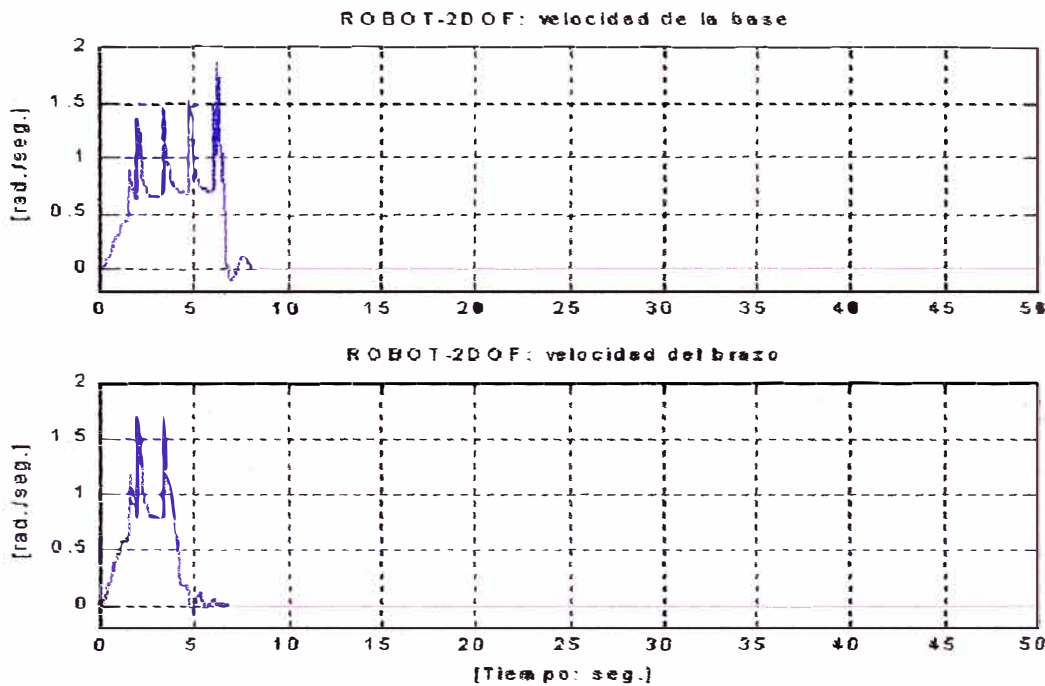


Figura C.5: Trayectoria Constante Experimental: velocidad del servomotor de la base y el brazo.

para el brazo: $-7.5 + 1.9365i$ y $-7.5 - 1.9365i$

Los resultados obtenidos para este experimento se muestran desde la figura C.5 hasta la figura C.7.

En la figura C.5 se muestra las salidas controladas para la posición angular de la base y el brazo, y que a partir de los 4 seg. para ambos casos sigue a la referencia. Se observa que en la trayectoria de la base y el brazo no existe sobreimpulso. Las señales se han fijado a una frecuencia de $1/21$ Hz de ciclo senoidal y con amplitudes de $\pi/2$ rad y $\pi/2$ rad, que indican el movimiento angular de la base y el brazo.

En la figura C.6 se muestran las señales de control para el servomotor de la base y el servomotor del brazo. La señal de control para ambos llega hasta 3 voltios por un instante de tiempo (4 seg.), posteriormente presenta oscilaciones hasta el final del tiempo del experimento siguiendo una aproximación a la trayectoria senoidal. La oscilación que presentan ambas señales es debido al gran esfuerzo que realizan los servomotores para que el sistema sea controlado.

En la figura C.7 se muestra la respuesta al sistema para la velocidad angular del servomotor de la base y el brazo. En el gráfico se aprecia que en el arranque la

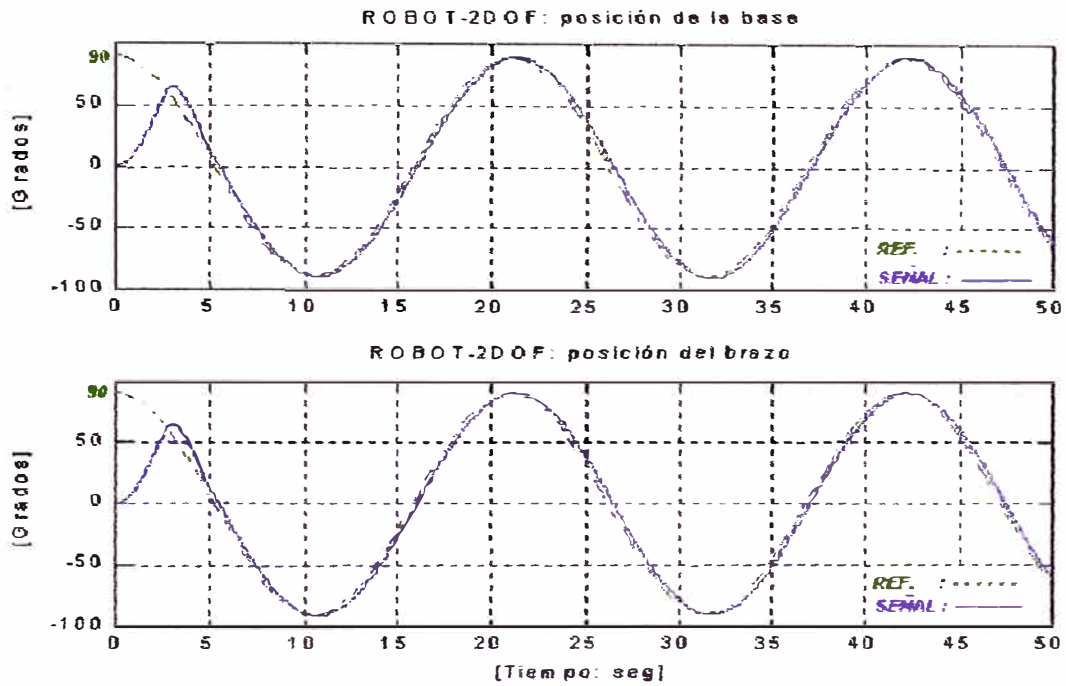


Figura C.6: Trayectoria Senoidal Experimental: salidas controladas para la base y el brazo.

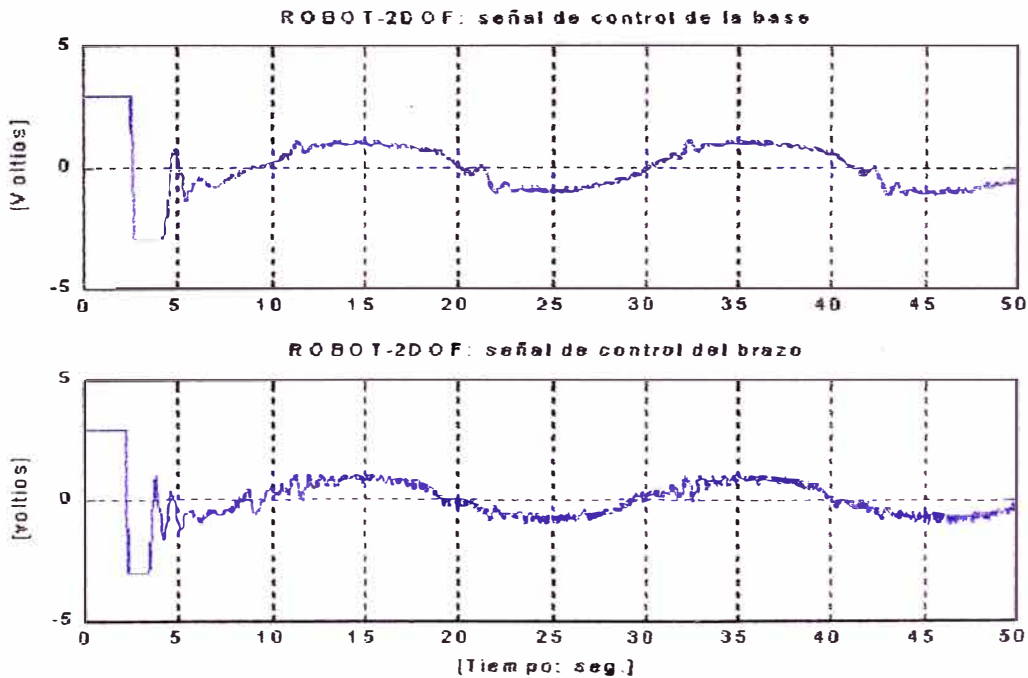


Figura C.7: Trayectoria Senoidal Experimental: señal de control para el servomotor de la base y el brazo.

velocidad en ambos casos no es senoidal presentando ciertas oscilaciones, después de 6 seg. para ambos casos siguen a la trayectoria senoidal de la referencia con pequeñas oscilaciones, debido a que los servomotores están ejerciendo fuerza de control.

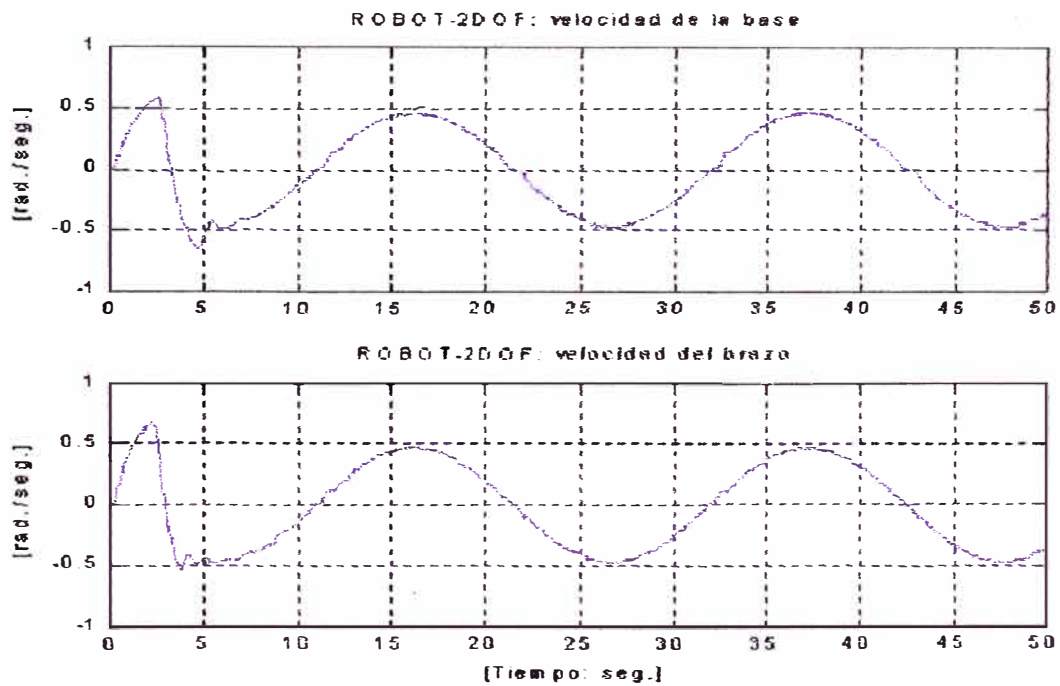


Figura C.8: Trayectoria Senoidal Experimental: velocidad del servomotor de la base y el brazo.