

UNIVERSIDAD NACIONAL DE INGENIERÍA

FACULTAD DE INGENIERÍA MECÁNICA



**TÉCNICAS DE CONTROL NO LINEAL
MODERNO APLICADAS AL CONTROL DE
SEGUIMIENTO A UNA TRAYECTORIA DE
REFERENCIA ANGULAR DE UN
MANIPULADOR ESFÉRICO DE 3GDL**

TESIS

PARA OPTAR EL TÍTULO PROFESIONAL DE:
INGENIERO MECATRÓNICO

RODNEY ATALAYA CHÁVEZ

PROMOCIÓN 2007-I

LIMA - PERÚ

2012

A Dios.

A mis padres, Juan y María.

A mi hermano Juan Carlos.

TABLA DE CONTENIDOS

PRÓLOGO	1
CAPÍTULO 1.	
Introducción	3
1.1. Antecedentes	5
1.2. Formulación del Problema	7
1.3. Objetivos de la Tesis	7
1.4. Organización de la Tesis	8
CAPÍTULO 2.	
Modelamiento Dinámico del Robot Manipulador	9
2.1. Nociones Previas	9
2.2. Orientación de un cuerpo rígido	11
2.3. Matriz de Rotación	11
2.3.1. Matrices de Rotación Elementales	12
2.4. Derivada de Matriz de Rotación	14
2.5. Transformaciones Homogéneas	16
2.6. Procedimiento de Denavit Hartenberg	17
2.6.1. Selección de los Sistemas de Referencia según D-H	18
2.6.2. Parámetros de D-H	19
2.6.3. Matriz de Transformación Homogénea	19
2.7. Velocidad entre Sistemas de Denavit-Hartenberg	21
2.8. Aceleración entre Sistemas de Denavit-Hartenberg	24
2.9. Algoritmo Recursivo de Newton-Euler	26
2.10. Modelo del Actuador DC	28
2.10.1. Modelado del Motor N°1 de la Base del Robot	28
2.10.2. Modelado del Motor N°2 del brazo del Robot	30
2.10.3. Modelado del Motor N°3 del Antebrazo del Robot	31
2.11. Modelo en el Espacio Estado	32
CAPÍTULO 3.	
Control por Linealización Exacta por Realimentación de Estado	34
3.1. Herramientas de la Geometría Diferencial	34
3.2. Derivadas y Corchetes de Lie	35
3.3. Difeomorfismos y Transformación de Coordenadas	37
3.4. El Teorema de Frobenius	38
3.4.1. El Grado Relativo de un Sistema	39

3.4.2.	Forma Normal SISO con Linealización Exacta	40
3.5.	Linealización Exacta y Ley de Control. Caso: SISO	42
3.5.1.	Condiciones para Linealización	45
3.6.	Aplicación al Control de Velocidad de un Motor DC	46
3.6.1.	Modelo del Proceso No lineal en Espacio-Estado	48
3.6.2.	Cálculo del Grado Relativo	49
3.6.3.	Comprobación Estricta de la Linealización Exacta	49
3.6.4.	Forma Normal	50
3.6.5.	Ley de Control	51
3.6.6.	Simulación del Controlador para el Proceso No Lineal	52
3.7.	Linealización por Realimentación Exacta: Caso MIMO	55
3.7.1.	Modelando Sistemas MIMO Cuadrados	55
3.7.2.	Grado Relativo Total	56
3.7.3.	Forma Normal MIMO para Linealización Exacta	57
3.7.4.	La Ley de Control MIMO Desacoplada	58

CAPÍTULO 4.

Control Predictivo Basados en Modelos	61	
4.1.	Introducción al Control Predictivo Basado en Modelo	62
4.2.	Estrategia del Control Predictivo Basado en Modelos	63
4.3.	Aspectos de Implementación del Control Predictivo	64
4.4.	Modelado y Predicción de la Salida Controlada	64
4.4.1.	Modelado Lineal	64
4.4.2.	Modelado No Lineal	67
4.5.	Función de Coste y Optimización	68
4.6.	Control Predictivo No Lineal	68
4.7.	Control Predictivo Generalizado No Lineal en Tiempo Continuo para Sistemas MIMO	69
4.7.1.	Predicción de la Salida	70
4.7.2.	Predicción de la Trayectoria de Referencia Futura	72
4.7.3.	Minimización de la Función de Costo	73
4.8.	Aplicación al Control de Velocidad de un Motor DC	77
4.8.1.	Simulación del Controlador para el Proceso No Lineal	79

CAPÍTULO 5.

Simulaciones del Sistema	81	
5.1.	Descripción Detallada del Robot	82
5.1.1.	Primer Eslabón: Base	82
5.1.2.	Segundo Eslabón: Brazo	83
5.1.3.	Tercer Eslabón: Antebrazo	83
5.2.	Modelo Cinemático del Robot Propuesto	84
5.2.1.	Cálculo de las Matrices de Transformación Homogénea	84
5.3.	Cálculo de los Momentos de Inercia	86
5.4.	Lista de variables	88
5.5.	Control por Linealización Exacta por Realimentación de Estados (CLERE) Aplicado al Manipulador Robótico Esférico de 3GDL	89
5.5.1.	Determinación del Grado Relativo Total, r	91
5.5.2.	Forma Normal	92

5.5.3. Ley de Control	93
5.5.4. Respuesta a una Trayectoria Constante	93
5.5.5. Respuesta a una Trayectoria Senoidal	96
5.6. Control Predictivo Generalizado No Lineal Aplicado al Manipulador Robótico de 3GDL	100
5.6.1. Respuesta a una Trayectoria Constante	103
5.6.2. Respuesta a una Trayectoria Senoidal	106
5.7. Animación con la Interfaz Gráfica GUI	109
5.7.1. Uso del Panel de Control	110

CAPÍTULO 6.

Implementación en Tiempo Real del Manipulador Robótico de 3GDL	113
6.1. Tarjeta de Adquisición de Datos	114
6.2. El Amplificador	116
6.2.1. Descripción	117
6.3. Motores Usados en el Robot	119
6.3.1. Primer Eslabón: Base	119
6.3.2. Segundo Eslabón: Brazo	120
6.3.3. Tercer Eslabón: Antebrazo	120
6.4. Conexión Eléctrica con la Tarjeta PCI-6722 NI	120
6.5. Tarjeta PCI-6602 NI	120
6.6. El Sensor	122
6.6.1. Tecnología del Sensor	122
6.6.2. Descripción del Sensor Usado	124
6.6.3. Señales A, B y Z	125
6.7. Conexión con la Tarjeta PCI-6602 NI	126
6.8. Software de Control	127
CONCLUSIONES	128
BIBLIOGRAFÍA	130
APÉNDICE A.	
Programa del Cálculo de la Ecuación Dinámica del Robot	132
APÉNDICE B.	
Control por Linealización Exacta por Realimentación de Estados: Caso MIMO	136
APÉNDICE C.	
Control Predictivo Generalizado No Lineal: Caso MIMO	142
APÉNDICE D.	
TÉRMINOS DE MATRIZ DE INERCIA: Hij	147

APÉNDICE E. TÉRMINOS DE CORIOLISIS: Cj	149
---	-----

APÉNDICE F. TÉRMINOS DE GRAVEDAD: Gj	152
---	-----

LISTA DE FIGURAS

2.1. Robot manipulador	10
2.2. Tipos de articulación	10
2.3. Posición y orientación de un cuerpo rígido.	11
2.4. Rotación del sistema $O-xyz$ un ángulo α alrededor del eje \mathbf{z}	13
2.5. Sistema coordinado rotando alrededor de un eje arbitrario	15
2.6. Representación del punto P en diferentes sistemas de coordenadas	16
2.7. Sistemas de coordenadas asociadas a un eslabón	21
2.8. Modelo eléctrico y mecánico del motor reductor DC1	29
2.9. Modelo eléctrico y mecánico del motor reductor DC2	30
2.10. Modelo eléctrico y mecánico del motor reductor DC3	31
3.1. Diagrama de bloques para linealización exacta de un proceso no lineal	45
3.2. Brazo robótico de 1GDL	47
3.3. Diagrama de bloques del sistema de control para motor DC	52
3.4. Salida controlada para la posición del brazo de 1GDL con CLERE	53
3.5. Señal de control para el motor DC	54
3.6. Velocidad angular del brazo de 1GDL acoplado al motor DC	54
4.1. Estrategia del CPBM	63
4.2. Modelado y predicción	65
4.3. Comparación entre las estrategias CLERE y CPGNL	80
4.4. Señal de control para el motor DC	80
5.1. Estructura del robot	81
5.2. Grados de libertad del robot	82
5.3. Elementos del robot	83
5.4. Sistemas de referencia según D-H	84
5.5. Se muestra el programa Solidwork en el que fue diseñado el robot de 3GDL	86
5.6. Cálculo de inercia correspondiente al eslabón 1 usando Solidwork	87
5.7. Sistemas inerciales y de centro de masa de los eslabones	88
5.8. Diagrama de bloques del control por realimentación exacta	90
5.9. Salidas controladas para la posición angular de la base, brazo y antebrazo, respectivamente.	94
5.10. Error de posición angular de la base, brazo y antebrazo, respectivamente.	95
5.11. Derivada del error de posición angular de la base, brazo y antebrazo, respectivamente.	95

5.12. Señales de control para los motores que manejan a la base, brazo y antebrazo, respectivamente.	96
5.13. Salidas controladas para la posición angular de la base, brazo y antebrazo, respectivamente.	97
5.14. Error de posición angular de la base, brazo y antebrazo, respectivamente.	98
5.15. Derivada del error de posición angular de la base, brazo y antebrazo, respectivamente.	98
5.16. Señales de control para los motores que manejan a la base, brazo y antebrazo, respectivamente.	99
5.17. Salidas controladas para la posición angular de la base, brazo y antebrazo, respectivamente.	104
5.18. Error de posición angular de la base, brazo y antebrazo, respectivamente.	105
5.19. Señales de control para los motores que manejan a la base, brazo y antebrazo, respectivamente.	105
5.20. Salidas controladas para la posición angular de la base, brazo y antebrazo, respectivamente.	106
5.21. Error de posición angular de la base, brazo y antebrazo, respectivamente.	107
5.22. Señales de control para los motores que manejan a la base, brazo y antebrazo, respectivamente.	107
5.23. Salidas controladas para la posición angular de la base, brazo y antebrazo, respectivamente.	108
5.24. Señales de control para los motores que manejan a la base, brazo y antebrazo, respectivamente.	109
5.25. Panel de Control Principal para la Animación del Robot.	111
5.26. Tipos de trayectorias de referencia	112
5.27. Comparación entre las trayectorias de referencia y del control predictivo.	112
6.1. Esquema para la implementación en tiempo real del sistema robótico	113
6.2. Implementación en tiempo real del robot	114
6.3. Asignación de los pines del PCI-6722 de National Instruments	117
6.4. Amplificador serie 25A8 de Advanced Motions Control	118
6.5. Disposición de los motores que componen al sistema robótico	119
6.6. Conexión de la PCI-6722 NI con los amplificadores	121
6.7. Asignación de los pines del PCI-6602 de National Instruments para aplicaciones de lectura de encoders	123
6.8. Principio de Funcionamiento del Encoder	124
6.9. Encoder incremental HEDS9100 acoplado al motor Pittman	125
6.10. Señales A, B y Z que Proporciona el Encoder	126
6.11. Conexión de los terminales del encoder con la bornera CB-68LP	127

LISTA DE CUADROS

5.1. Parámetros D-H	84
5.2. Lista de variables de eslabones y motor	88
5.3. Lista de variables de eslabones y motor	89
6.1. Niveles Lógicos para las Entradas/Salidas Digitales	116
6.2. Niveles Lógicos para Counter/Timer y Entradas/Salidas Digitales	122

PRÓLOGO

La presente tesis trata sobre el estudio del control para el seguimiento a una trayectoria de referencia angular de un manipulador robótico esférico de 3 grados de libertad (3GDL) usando técnicas de control no lineal modernas como son, el método de Control Linealización Exacta por Realimentación de Estado (CLERE) y el Control Predictivo Generalizado No Lineal (CPGNL).

La linealización exacta por realimentación de estado se basa en la linealización del proceso mediante un cambio adecuado de coordenadas y realimentación del vector de estado de modo que el proceso resultante es lineal y desacoplado, por lo tanto pueden ser aplicadas técnicas de control lineal. Esta técnica extiende el concepto de grado relativo de sistemas lineales a sistemas no lineales. Esto se consigue valiéndose de conceptos de geometría diferencial, como son: la derivada y corchete de Lie.

Otra de las técnicas de control moderno tratada en la presente tesis es el control predictivo generalizado para sistemas no lineales. Este algoritmo se basa en obtener el error entre la salida predicha del proceso y la trayectoria de referencia futura por medio de la serie de Taylor. Luego una función de coste que depende del cuadrado del error de predicción se minimiza usando el criterio de la primera derivada, esto permite obtener una señal de control futura y de la cual sólo se envía al proceso la que corresponde al tiempo actual $u(t)$. Con esta técnica se obtiene una ley de control óptima y fuera de línea (off-line), es decir, que no requiere resolver el problema de optimización en cada instante de muestreo para hallar la señal de control óptima. Por consiguiente, esta técnica puede ser aplica-

do al control de sistemas no lineales de dinámica rápida (robots, motores, etc). Ambos algoritmos de control han sido aplicados satisfactoriamente al control angular de un manipulador y en las simulaciones se observa que las señales de control permiten a las salidas del robot seguir adecuadamente trayectorias angulares de referencia predefinidas.

Adicionalmente se estudia aspectos relacionados a la dinámica del manipulador robótico. La convención de Denavit-Hartenberg(D-H) nos permite seleccionar la ubicación de sistemas de referencias los cuales definen la posición, orientación de cada eslabón. La ecuación dinámica del manipulador se ha logrado usando el algoritmo recursivo de Newton-Euler.

Los resultados de la presente tesis fueron expuestos en el XVII Congreso Nacional de Estudiantes Ingeniería Mecánica, Eléctrica, Electrónica y Ramas Afines (CONEIMERA). El artículo presentado se adjunta al final de este documento.

CAPÍTULO 1

Introducción

En los últimos años se ha observado una tendencia hacia el desarrollo de sistemas de manufactura flexible. En este último se busca que el trabajo de re-programación, para desarrollar nuevas tareas, sea lo menor posible.[1]

Los robots industriales son componentes importantes dentro de las celdas de manufactura flexible. Éstos realizan diversas tareas tales como soldadura, posicionamiento de cargas, pintura, etc. Para realizar estas tareas eficientemente y con alta precisión muchos enfoques de control moderno han sido propuestos en los últimos años tales como control no lineal por el método de linealización exacta, control no lineal sliding, control adaptativo no lineal y el control predictivo no lineal.

Uno de los controladores ampliamente difundidos es el PID. Sin embargo, para propósitos de manufactura flexible el control PID no es el más adecuado debido a que un cambio en la tarea necesita nuevamente sintonizarlo (seleccionar otras ganancias proporcional, derivativa e integral)[2].

Además robots seriales con muchos componentes vinculados presentan una dinámica altamente no lineal y el fuerte acoplamiento entre sus articulaciones dificulta la tarea de sintonización del PID.[2]

Por consiguiente, las estrategias de control moderno que hacen uso de la dinámica no lineal del sistema se perfilan como la mejor alternativa para alcanzar un mejor desempeño en sistemas de manufactura flexible.

Las estrategias que trataremos en esta tesis son, Control por Linealización Exacta

por Realimentación de Estados (CLERE) y Control Predictivo Generalizado No Lineal (CPGNL).

La técnica CLERE se basa en la linealización del proceso mediante un cambio adecuado de coordenadas y realimentación no lineal del vector de estado, de modo que el proceso resultante es aparentemente lineal y por lo tanto pueden ser aplicadas técnicas de control lineal, como la técnica de control por ubicación de polos.

El control predictivo no lineal basado en modelo, es una de las técnicas de control moderno más potentes y goza de mucha aceptación tanto en las aplicaciones industriales como en el campo académico.[3]

Los principios de este controlador son resumidos a continuación,

1. Predecir la salida por encima de un intervalo de tiempo futuro, llamado horizonte de predicción.
2. Conocido el setpoint futuro seleccionamos un conjunto de señales de control futura que minimizan una función de costo que está en función de los errores futuros entre la salida predicha y el setpoint futuro.
3. El primer elemento de este conjunto de señales de control futuro, $u(t)$, se envía al sistema. Luego todo el procedimiento anterior se repite en cada instante de muestreo.

El uso de un modelo lineal para predecir la salida ha motivado la aparición de muchos algoritmos de Control Predictivo Lineal (CPL) ejemplo de ello son: DMC(Dynamic Matrix Control), MAC(Model Algorithm Control), GPC(Generalized Predictive Control), un estudio completo de estas técnicas de control predictivo lineal pueden ser encontradas en [4].

El Control Predictivo Lineal usa un modelo lineal para predecir la salida, pero en caso de trabajar con el modelo no lineal del sistema trae ciertas dificultades. Una de ellas, es que al usar un modelo no lineal para la predicción de las salidas convierte al problema de minimización de la función de costo en un problema de

optimización no lineal el cual requiere para su solución algoritmos numéricos de elevada carga computacional [3], este factor impide su aplicación para sistemas de dinámica rápida (robots).

Por este motivo estas técnicas están relegadas al control de sistemas de dinámica lenta por ejemplo, para el control de temperatura en reactores, en columnas de destilación.

Con el objetivo de evadir el problema de optimización no lineal, la presente tesis plantea un algoritmo control predictivo generalizado que deriva una ley de control fuera de línea (off-line), es decir no requiere hallar las señales de control óptimas en cada periodo de muestreo.

Finalmente estos algoritmos son implementados sobre la plataforma del Matlab para el control de seguimiento a una trayectoria angular de un sistema robótico de 3GDL (grados de libertad) accionado con motores de corriente continua.

1.1. Antecedentes

Los robots industriales son ampliamente usados para realizar varias tareas tales como, ensamblaje, soldadura o pintado. Para realizar estas tareas de manera precisa y eficiente, muchos enfoques de control han sido propuestos. Entre éstos tenemos al simple Control Proporcional Derivativo con Compensación de Gravedad, propuesto por Spong M. W y Vidyasagar M. en 1989.

Posteriormente, el Control por Linealización Exacta por Realimentación de Estado (CLERE), ampliamente investigado por Slotine-Li, 1991, Isidori, 1995 y Khalil, 1996, permite mejorar el comportamiento de la variable controlada. Esto se debe por que el CLERE consta de dos lazos de realimentación, un primer lazo que transforma el sistema original en un sistema lineal mediante una realimentación no lineal de las variables de estado y un segundo lazo externo que controla el sistema lineal resultante.

La técnica de control predictivo fue introducida por Richalet en 1978 con su Control Predictivo Heurístico Basado en Modelo (Model Predictive Heuristic Control,

MPHC), más tarde conocido como Control Algorítmico Basado en Modelo (Model Algorithmic Control, MAC), por otro lado, de forma independiente: Cutler y Ramaker, ingenieros de la compañía Shell, presentaron en 1980 su Control con Matriz Dinámica (Dynamic Matrix Control, DMC). Ambos algoritmos utilizan explícitamente el modelo dinámico del proceso a controlar, esto se obtuvo de la respuesta al impulso para el primer caso y de la respuesta al escalón para el segundo, para predecir el efecto de las futuras señales de control en las variables a controlar.

La formulación del control predictivo está íntimamente ligado al problema de control óptimo y a la programación lineal. El concepto de horizonte deslizante, es una de las ideas centrales del control predictivo y fue propuesto por Propoi en 1963.

El control predictivo para sistemas lineales ha recibido considerable atención en la última década debido a su robustez con respecto a un modelo con incertidumbre. En consecuencia, podemos encontrar muchos de éstos algoritmos implementados en el campo industrial.

Sin embargo, muchos sistemas son inherentemente no lineales. Por tanto el uso de modelos lineales para describir la dinámica de un proceso resulta inadecuado. Ante este problema se han realizado muchos esfuerzos con el objetivo de extender la estrategia de control predictivo a sistemas no lineales, Michalska H. y Mayne D. en 1993.

La desventaja para lograr este propósito es la alta carga computacional que se requiere para solucionar el problema de optimización en cada instante de muestreo, por ello la aplicación de esta técnica al control de robots (dinámica rápida) suena muy remota.

Para vencer la carga computacional, muchos algoritmos han sido propuestos Ping L. en 1995, Singh S. M. en 1995, Souroukh M.-Kravaris C. en 1996 y Chen W. en 2003, los cuales han tenido un objetivo común, derivar una ley de control fuera de línea, es decir, que no sea calculada en cada instante de muestreo.

El algoritmo tratado en la tesis, es el control predictivo generalizado no lineal propuesto originalmente por Siller-Alcalá, luego extendido a sistemas multivari-ables por P. J. Gawthrop, H. Demircioglu.

1.2. Formulación del Problema

El sistema a controlar es un manipulador robótico de 3 grados de libertad, que está compuesto de 3 eslabones conectados en serie por articulaciones rotativas, cada una de ellas conducida por un motorreductor DC *Pittman* con encoder. Los eslabones y acoplamientos del robot han sido considerados como rígidos. El problema que se estudia en la presente tesis es el control de seguimiento a una trayectoria angular para el manipulador. Esto se ha conseguido haciendo uso de técnicas de control moderno como son, el *control por linealización exacta por realimentación de estado* y el *control predictivo no lineal*. En ambos casos el controlador diseñado es no lineal y multivariable (MIMO) con tres entradas y tres salidas. Las entradas y salidas del robot vienen dados por el voltaje entregado a los motorreductores DC y por el ángulo que gira una articulación, respectivamente.

1.3. Objetivos de la Tesis

Los objetivos de la presente tesis son:

- a. Diseñar algoritmos de control modernos como son, el *control por linealización exacta por realimentación de estado* y el *control predictivo no lineal* para controlar el seguimiento a una trayectoria de referencia angular del manipulador robótico de 3 grados de libertad.
- b. Obtener el modelo dinámico del robot usando Newton-Euler.
- c. Simular estos controladores modernos junto a la planta bajo la plataforma de MATLAB.

1.4. Organización de la Tesis

La tesis está organizada como sigue:

En el Capítulo 2 se desarrolla el modelado matemático del robot de 3GDL. Éste ha sido obtenido basándose en el algoritmo recursivo de Newton-Euler y usando matemática simbólica de Matlab.

En el Capítulo 3 se presenta algunas herramientas matemáticas que permiten formular las condiciones por las cuales un sistema no lineal SISO o MIMO puede ser linealizado en forma exacta por realimentación de estado. Además se desarrolla el algoritmo de control. Finalmente, este algoritmo es aplicado al control de posición de una varilla que está acoplada al eje de un motor DC.

En el Capítulo 4 de control predictivo generalizado para procesos no lineales.

En el Capítulo 5 se ilustran los resultados de las simulaciones realizadas bajo la plataforma del Matlab.

En el Capítulo 6 se presenta el hardware y software del sistema usados en una posible implementación del sistema robótico en tiempo real.

Finalmente se presentan las conclusiones y recomendaciones que han sido encontradas a lo largo de la ejecución de la presente tesis.

Por último se presentan los apéndices:

En el apéndice A, se muestra .

En el apéndice B, se da a conocer conceptos como la composición de matrices de rotación, así como de transformaciones homogéneas.

En el apéndice C, se determina las relaciones cinemáticas entre los sistemas contiguos de D-H, muy útiles para determinar las velocidades lineales y angulares de cada eslabón.

CAPÍTULO 2

Modelamiento Dinámico del Robot Manipulador

2.1. Nociones Previas

En esta sección se conceptualizan algunos de los términos que se usarán a lo largo de la presente tesis.

Definición 2.1.1 *Robot Manipulador* :

Consiste en una serie de cuerpos rígidos o eslabones conectados por medio de articulaciones.

La estructura de un robot manipulador (Figura 2.1) forma una cadena cinemática abierta, donde uno de sus extremos está fijo a la base y en el extremo libre se instala la tenaza o efector final para la manipulación de objetos en el espacio.

Adicionalmente,

Definición 2.1.2 *Eslabón* :

Es un cuerpo rígido con al menos un punto en particular llamado Nodo, que soporta la conexión de otro eslabón.

Definición 2.1.3 *Eslabón Aumentado* :

Es el cuerpo formado por el eslabón propiamente dicho y el motor que se encuentra solidario a él.

Definición 2.1.4 *Articulación* :

Es la conexión entre dos o más eslabones en sus nodos. Esto restringe el movimiento relativo de los eslabones conectados.

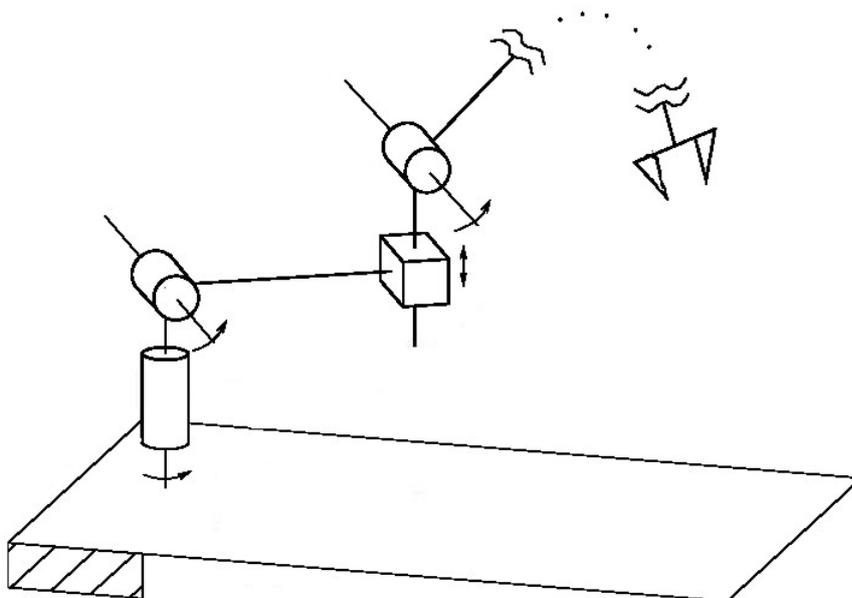
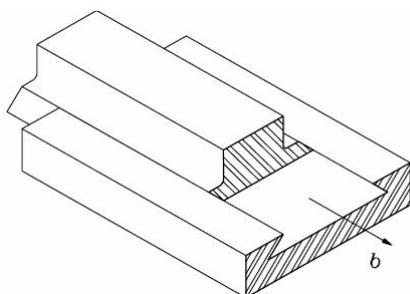


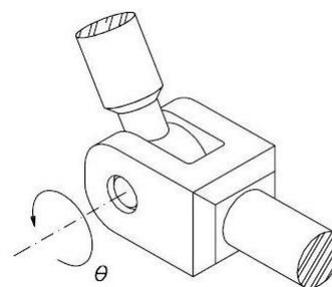
Figura 2.1: Robot manipulador

Si una articulación une a dos eslabones, esta entidad se llama **par cinemático**. Dependiendo de los grados de libertad del par cinemático, este se llama: articulación de 1GDL, articulación de 2GDL, etc.

En Robótica para facilitar el análisis cinemático y dinámico, se usan articulaciones con 1GDL. Si la articulación de 1GDL, impone un movimiento de traslación esta es llamada **Articulación Prismática** (Figura 2.2(a)) o si lo restringe a un movimiento rotacional (Figura 2.2(b)), esta es llamada **Articulación de Revolución**.



(a) Articulación prismática



(b) Articulación de revolución

Figura 2.2: Tipos de articulación

2.2. Orientación de un cuerpo rígido

La orientación¹ de un cuerpo rígido, ver Figura 2.3, es expresado en términos de los vectores unitarios del sistema solidario al cuerpo $O'-x'y'z'$, con origen en el punto O' , respecto al sistema de coordenadas $O-xyz$, tal como:

$$\begin{aligned} \mathbf{x}' &= x'_x \mathbf{x} + x'_y \mathbf{y} + x'_z \mathbf{z} = \\ \mathbf{y}' &= y'_x \mathbf{x} + y'_y \mathbf{y} + y'_z \mathbf{z} = \\ \mathbf{z}' &= z'_x \mathbf{x} + z'_y \mathbf{y} + z'_z \mathbf{z} = \end{aligned} \underbrace{\begin{bmatrix} x'_x & x'_y & x'_z \\ y'_x & y'_y & y'_z \\ z'_x & z'_y & z'_z \end{bmatrix}}_{\text{Matriz de Coordenadas}} \begin{bmatrix} \mathbf{x} \\ \mathbf{y} \\ \mathbf{z} \end{bmatrix} \quad (2.1)$$

Donde:

Los componentes x'_i , y'_i y z'_i ; $\forall i=x,y,z$ representan los cosenos directores del sistema $O'-x'y'z'$ respecto de $O-xyz$.

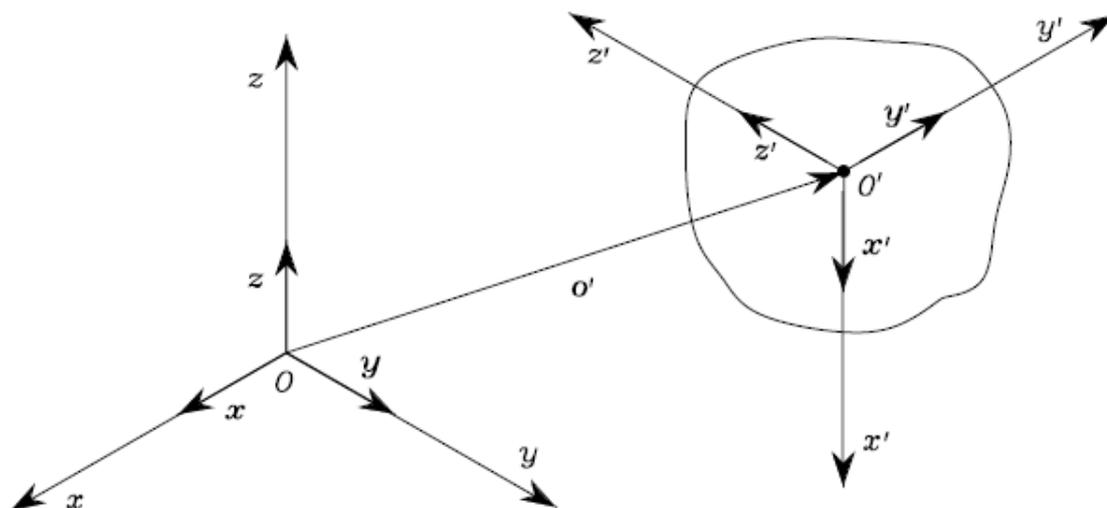


Figura 2.3: Posición y orientación de un cuerpo rígido.

2.3. Matriz de Rotación

Desde el punto de vista del Álgebra Lineal, los vectores unitarios del sistema coordinado solidario al cuerpo, forman una base dada por: $\mathbf{S}'=\{\mathbf{x}',\mathbf{y}',\mathbf{z}'\}$, de igual manera el sistema de referencia está dada por la base: $\mathbf{S}=\{\mathbf{x},\mathbf{y},\mathbf{z}\}$.

¹Existen diversos métodos para definir la orientación tales como: Matrices de Rotación, Ángulos de Euler, Parámetros de Rodriguez, Cuaternios.

Si los vectores de la base \mathbf{S}' , se expresan como combinación lineal de los vectores de la base \mathbf{S} , tal como se observa en la ecuación (2.1), entonces se define la matriz de rotación \mathbf{R} como la transpuesta de la matriz de coordenadas, y viene dada por:

$$\mathbf{R}_{\mathbf{S}'}^{\mathbf{S}} = \begin{bmatrix} \mathbf{x}' & \mathbf{y}' & \mathbf{z}' \end{bmatrix} = \begin{bmatrix} x'_x & y'_x & z'_x \\ x'_y & y'_y & z'_y \\ x'_z & y'_z & z'_z \end{bmatrix} = \begin{bmatrix} x'^T x & y'^T x & z'^T x \\ x'^T y & y'^T y & z'^T y \\ x'^T z & y'^T z & z'^T z \end{bmatrix} \quad (2.2)$$

que representa la matriz de cambio de base, es decir, transforma la base antigua \mathbf{S} a la base nueva \mathbf{S}' , a su vez, describe de rotación del sistema \mathbf{S}' respecto al sistema de referencia \mathbf{S} .

2.3.1. Matrices de Rotación Elementales

Las matrices de rotación elementales corresponden al giro del sistema de referencia alrededor de uno de sus ejes $\{\mathbf{x}, \mathbf{y}, \mathbf{z}\}$.

Por ejemplo, la rotación del sistema de referencia $O-xyz$ un ángulo α alrededor del eje \mathbf{z} da como resultado al sistema rotado $O-uvw$, tal como se muestra en la Figura 2.4.

De la ecuación (2.1) podemos describir la orientación del sistema rotado respecto al sistema de referencia $O-xyz$, y es:

$$\begin{aligned} \mathbf{x}' &= \cos(\alpha)\mathbf{x} + \sin(\alpha)\mathbf{y} + 0 \mathbf{z} \\ \mathbf{y}' &= -\sin(\alpha)\mathbf{x} + \cos(\alpha)\mathbf{y} + 0 \mathbf{z} \\ \mathbf{z}' &= 0 \mathbf{x} + 0 \mathbf{y} + 1 \mathbf{z} \end{aligned} \quad (2.3)$$

Con la ecuación (2.3) en (2.2), se obtiene la matriz de rotación del sistema $O-uvw$

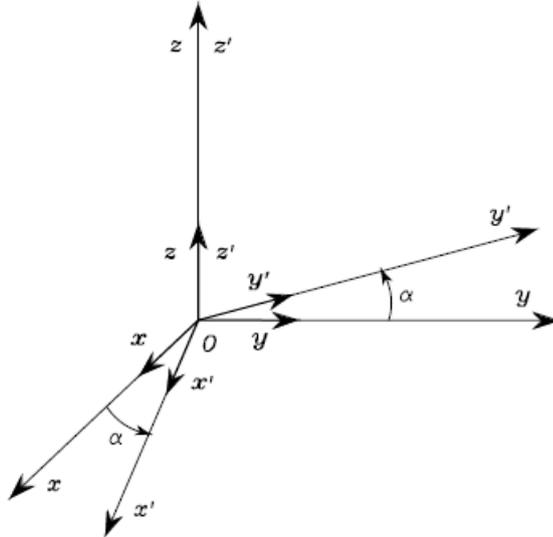


Figura 2.4: Rotación del sistema $O-xyz$ un ángulo α alrededor del eje z

con respecto a $O-xyz$, y se expresa como:

$$\mathbf{R}_{O-uvw}^{O-xyz} = \mathbf{R}_z(\alpha) = \begin{bmatrix} \mathbf{x}' & \mathbf{y}' & \mathbf{z}' \end{bmatrix} = \begin{bmatrix} \cos(\alpha) & -\sin(\alpha) & 0 \\ \sin(\alpha) & \cos(\alpha) & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (2.4)$$

De manera similar, se puede mostrar que la rotación de un ángulo β alrededor del eje y y un ángulo θ alrededor del eje x , están dados respectivamente por:

$$\mathbf{R}_y(\beta) = \begin{bmatrix} \cos(\beta) & 0 & \sin(\beta) \\ 0 & 1 & 0 \\ -\sin(\beta) & 0 & \cos(\beta) \end{bmatrix} \quad (2.5)$$

$$\mathbf{R}_x(\theta) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\theta) & -\sin(\theta) \\ 0 & \sin(\theta) & \cos(\theta) \end{bmatrix} \quad (2.6)$$

2.4. Derivada de Matriz de Rotación

Consideremos una matrix de rotación variante en el tiempo $\mathbf{R} = \mathbf{R}(t)$. En vista de la ortogonalidad de \mathbf{R} , se tiene que:

$$\mathbf{R}\mathbf{R}^T = \mathbf{I} \quad (2.7)$$

Diferenciando la ecuación de arriba con respecto al tiempo, se tiene:

$$\dot{\mathbf{R}}(t)\mathbf{R}^T(t) + \mathbf{R}(t)\dot{\mathbf{R}}^T(t) = 0 \quad (2.8)$$

Si consideramos,

$$\mathbf{S}(t) = \dot{\mathbf{R}}(t)\mathbf{R}^T(t) \quad (2.9)$$

Donde la matrix \mathbf{S} es de orden 3×3 y por la ecuación (2.8) es antisimétrica, esto es $\mathbf{S}(t) + \mathbf{S}^T(t) = 0$

Finalmente postmultiplicando a ambos lados de la ecuación (2.9) por $\mathbf{R}(t)$, se obtiene:

$$\dot{\mathbf{R}}(t) = \mathbf{S}(t)\mathbf{R}(t) \quad (2.10)$$

Consideremos un sistema que rota alrededor de un eje arbitrario a una velocidad angular ω , tal como se muestra en la Figura 2.5, y además un punto fijo \mathbf{P} en dicho sistema, por tanto $\mathbf{p}(t) = \mathbf{R}(t)\mathbf{p}'$, que derivando se obtiene:

$$\dot{\mathbf{p}}(t) = \dot{\mathbf{R}}(t)\mathbf{p}' \quad (2.11)$$

La ecuación (2.10) en (2.11), se tiene:

$$\dot{\mathbf{p}}(t) = \mathbf{S}(t)\mathbf{R}(t)\mathbf{p}' \quad (2.12)$$

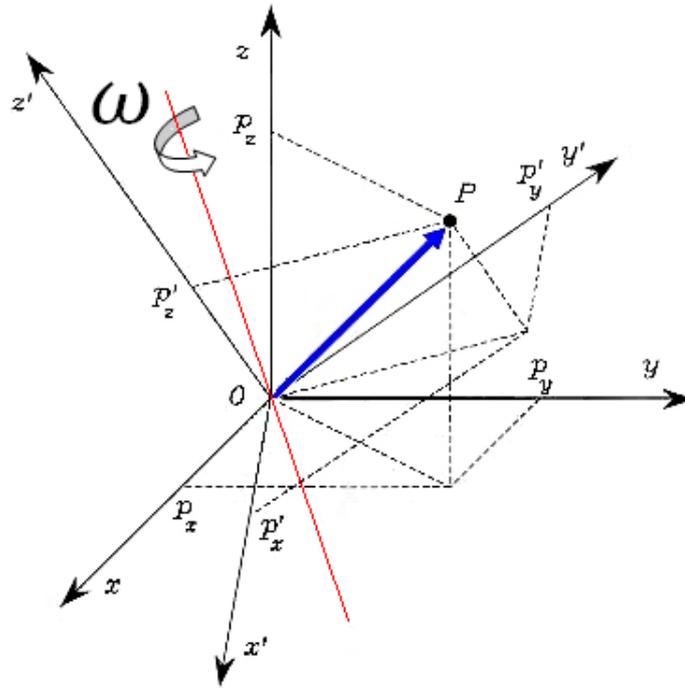


Figura 2.5: Sistema coordenado rotando alrededor de un eje arbitrario

Si $\omega(t)$ denota la velocidad angular del sistema respecto al sistema de referencia, es conocido por mecánica que:

$$\dot{\mathbf{p}}(t) = \omega(t) \times \mathbf{R}(t)\mathbf{p}' \quad (2.13)$$

Comparando las ecuaciones (2.12) y (2.13), tenemos:

$$\omega(t) \times \mathbf{R}(t) = \mathbf{S}(t)\mathbf{R}(t) \quad (2.14)$$

Si consideramos que $\omega(t) = [\omega_x \ \omega_y \ \omega_z]$, entonces la matriz $\mathbf{S}(t)$ obtenida de la ecuación (2.14) es:

$$\mathbf{S} = \begin{bmatrix} 0 & -\omega_z & \omega_y \\ \omega_z & 0 & -\omega_x \\ -\omega_y & \omega_x & 0 \end{bmatrix} \quad (2.15)$$

Si definimos que $\mathbf{S}(t) = \mathbf{S}(\omega(t))$, por tanto la ecuación (2.10), puede reescribirse como:

$$\dot{\mathbf{R}} = \mathbf{S}(\omega)\mathbf{R} \quad (2.16)$$

Además se puede demostrar que:

$$\mathbf{R}\mathbf{S}(\omega)\mathbf{R}^T = \mathbf{S}(\mathbf{R}\omega) \quad (2.17)$$

Donde: \mathbf{R} es la matriz de rotación.

2.5. Transformaciones Homogéneas

Con el objetivo de tratar conjuntamente la posición y la orientación de un sólido se introduce el concepto de coordenadas homogéneas.

Consideremos un punto P arbitrario en el espacio tal como se muestra en la

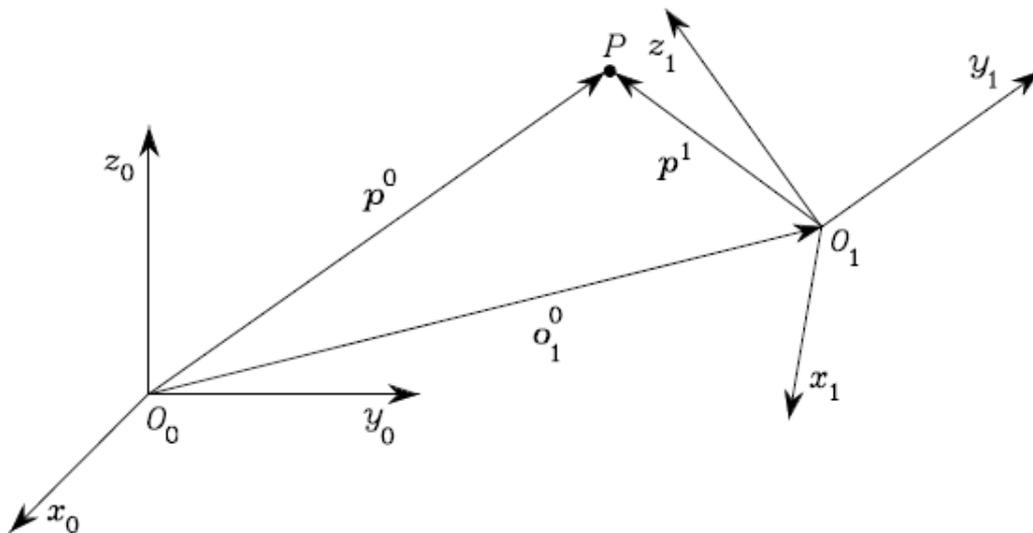


Figura 2.6: Representación del punto P en diferentes sistemas de coordenadas

Figura 2.6, y sea: \mathbf{p}^0 las coordenadas de P respecto al sistema de referencia O_0 - $x_0y_0z_0$. Además otro sistema de coordenadas O_1 - $x_1y_1z_1$, rotado \mathbf{R}_1^0 respecto a O_0 - $x_0y_0z_0$, y \mathbf{o}_1^0 es el vector que describe el origen de este sistema respecto al sistema de referencia, y sea \mathbf{p}^1 las coordenadas de P representado en el sistema

$O_1-x_1y_1z_1$. Basándose en una simple observación vemos que la posición del punto P con respecto al sistema de referencia viene expresado por:

$$\mathbf{p}^0 = \mathbf{o}_1^0 + \mathbf{R}_1^0 \mathbf{p}^1 \quad (2.18)$$

Para lograr una representación compacta entre la relación de las coordenadas de un mismo punto en diferentes sistemas de referencia, se introduce la representación homogénea de un vector \mathbf{p} como el vector $\tilde{\mathbf{p}}$, que se extiende a una cuarta componente unitaria del vector \mathbf{p} , esto es:

$$\tilde{\mathbf{p}} = \begin{bmatrix} \mathbf{p} \\ 1 \end{bmatrix} \quad (2.19)$$

De acuerdo a la representación homogénea la ecuación (2.18) se puede reescribir de la siguiente manera:

$$\tilde{\mathbf{p}}^0 = \mathbf{A}_1^0 \tilde{\mathbf{p}}^1 = \begin{bmatrix} \mathbf{R}_1^0 & \mathbf{o}_1^0 \\ \mathbf{0}^T & 1 \end{bmatrix} \tilde{\mathbf{p}}^1 \quad (2.20)$$

Donde, \mathbf{A}_1^0 es llamada Matriz de Transformación Homogénea, \mathbf{o}_1^0 es un vector de orden 3×1 y \mathbf{R}_1^0 es la matriz de rotación de orden 3×3

2.6. Procedimiento de Denavit Hartenberg

Para describir la relación geométrica existente entre dos eslabones contiguos de una cadena cinemática abierta en cada instante se coloca en cada uno de ellos un sistema coordenado solidario y arbitrario. Lo más habitual es seleccionarlos siguiendo la convención de Denavit-Hartenberg (D-H).

Denavit-Hartemberg propusieron en 1955 un método que permite seleccionar la ubicación de un sistema coordenado sobre un determinado eslabón, esto permite la obtención de las ecuaciones cinemáticas que gobiernan el movimiento.

Mediante esta selección es posible transformar un sistema coordinado a otro contiguo, por medio de 4 transformaciones básicas que dependen exclusivamente de las características geométricas del eslabón. Éstas transformaciones consisten en una sucesión de rotaciones y translaciones que permiten relacionar el sistema de coordenadas del eslabón $i-1$ con el sistema i . Un completo estudio de este procedimiento puede ser encontrado en [5], [6].

2.6.1. Selección de los Sistemas de Referencia según D-H

Para establecer los sistemas coordinados S_i , $i=0 \dots n$, de vectores unitarios $\{x_i, y_i, z_i\}$, conforme a D-H, se siguen las siguientes reglas:

1. El sistema de coordenadas inercial o de referencia S_0 , de vectores unitarios $\{x_0, y_0, z_0\}$, se fija a tierra y su vector unitario z_0 se localiza a lo largo del eje de movimiento ² del primer eslabón.
2. El vector unitario z_{i-1} se selecciona a lo largo del eje de movimiento de la i -ésima articulación.
3. El origen del sistema de coordenadas S_i se encuentra en la intersección de la normal común de los ejes de dos articulaciones consecutivas z_i y z_{i-1} con el eje z_i .
4. El eje x_i es perpendicular al eje z_{i-1} .
5. El eje y_i se determina siguiendo la regla de la mano derecha.
6. El sistema de coordenadas $\{x_n, y_n, z_n\}$, puede estar en cualquier parte de la tenaza, pero con la condición que el eje z_n esté a lo largo del eje z_{n-1} y apuntando hacia afuera del robot. El eje x_n es perpendicular a ambos ejes: z_n y z_{n-1} .

²Que puede ser translacional o rotacional, dependiendo de su tipo de articulación

2.6.2. Parámetros de D-H

Para establecer la relación entre 2 sistemas coordenados contiguos se requiere de 4 parámetros $(\theta_i, d_i, a_i, \alpha_i)$ que dependen de las características geométricas de cada eslabón y de las articulaciones que le unen con el anterior y siguiente, estos parámetros pueden ser obtenidos con las pautas que se plantean a continuación:

θ_i : Es el ángulo que forman los ejes x_{i-1} y x_i medido en un plano perpendicular al eje z_{i-1} , utilizando la regla de la mano derecha. Se trata de un parámetro variable si la articulación es giratoria.

d_i : Es la distancia a lo largo del eje z_{i-1} desde el origen del sistema de coordenadas $(i-1)$ -ésimo hasta la intersección del eje z_{i-1} con el eje x_i . Se trata de un parámetro variable si la articulación es prismática.

a_i : Es la distancia a lo largo del eje x_i que va desde la intersección del eje z_{i-1} con el eje x_i hasta el origen del sistema i -ésimo, para articulaciones giratorias. En el caso de articulaciones prismáticas, se calcula como la distancia más corta entre los ejes z_{i-1} y z_i .

α_i : Es el ángulo de separación del eje z_{i-1} y el eje z_i , medido en un plano perpendicular al eje x_i , utilizando la regla de la mano derecha.

2.6.3. Matriz de Transformación Homogénea

Después de establecer los sistemas de coordenados solidarios a los eslabones y de obtener los parámetros D-H. El procedimiento de D-H establece que será posible transformar sistema coordenado a otro contiguo por medio de 4 transformaciones básicas, éstas consisten en una sucesión de rotaciones y translaciones, las que se describen a continuación:

- Rotación alrededor del eje z_{i-1} un ángulo θ_i .
- Translación a lo largo de z_{i-1} una distancia d_{i-1} .
- Translación a lo largo de x_i una distancia a_i .

- Rotación alrededor del eje x_i un ángulo α_i .

El producto de estas cuatro transformaciones básicas produce la Matriz de Transformación Homogénea \mathbf{A}_i^{i-1} , la cual describe la localización³ del sistema i con respecto al sistema $i-1$. Por tanto:

$$\begin{aligned}
 \mathbf{A}_i^{i-1} &= \mathbf{R}(\theta_i)\mathbf{T}(d_i)\mathbf{T}(a_i)\mathbf{R}(x_i, \alpha_i) \\
 &= \begin{bmatrix} \cos(\theta_i) & -\sin(\theta_i) & 0 & 0 \\ \sin(\theta_i) & \cos(\theta_i) & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \times \begin{bmatrix} 1 & 0 & 0 & a_i \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \dots \\
 &\quad \times \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos(\alpha_i) & -\sin(\alpha_i) & 0 \\ 0 & \sin(\alpha_i) & \cos(\alpha_i) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\
 \mathbf{A}_i^{i-1} &= \begin{bmatrix} \cos(\theta_i) & -\cos(\alpha_i)\sin(\theta_i) & \sin(\alpha_i)\sin(\theta_i) & a_i\cos(\theta_i) \\ \sin(\theta_i) & \cos(\alpha_i)\cos(\theta_i) & -\sin(\alpha_i)\cos(\theta_i) & a_i\sin(\theta_i) \\ 0 & \sin(\alpha_i) & \cos(\alpha_i) & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.21)
 \end{aligned}$$

Adicionalmente la Matriz de Transformación Homogénea, que determina la localización del sistema coordenado S_i con respecto al sistema coordenado inercial o de referencia S_0 , viene dado en (2.22).

$$\mathbf{A}_i^0 = \prod_{j=1}^i \mathbf{A}_j^{j-1} = \mathbf{A}_1^0 \mathbf{A}_2^1 \dots \mathbf{A}_i^{i-1}, \quad i = 1, 2, \dots, n \quad (2.22)$$

³Posición y Orientación

2.7. Velocidad entre Sistemas de Denavit-Hartenberg

Considerando un eslabón i arbitrario de una cadena cinemática abierta y siguiendo la convención de Denavit-Hartenberg (D-H) se seleccionan los sistemas de coordenadas S_{i-1} y S_i sobre las articulaciones i e $i+1$ respectivamente, ver Figura 2.7.

Donde:

\mathbf{p}_{i-1} : Es el vector posición del origen del sistema $i - 1$.

\mathbf{p}_i : Es el vector posición del origen del sistema i .

$\mathbf{r}_{i-1,i}^{i-1}$: Denota la posición del origen del sistema i con respecto al sistema $i-1$ expresado en el sistema $i-1$.

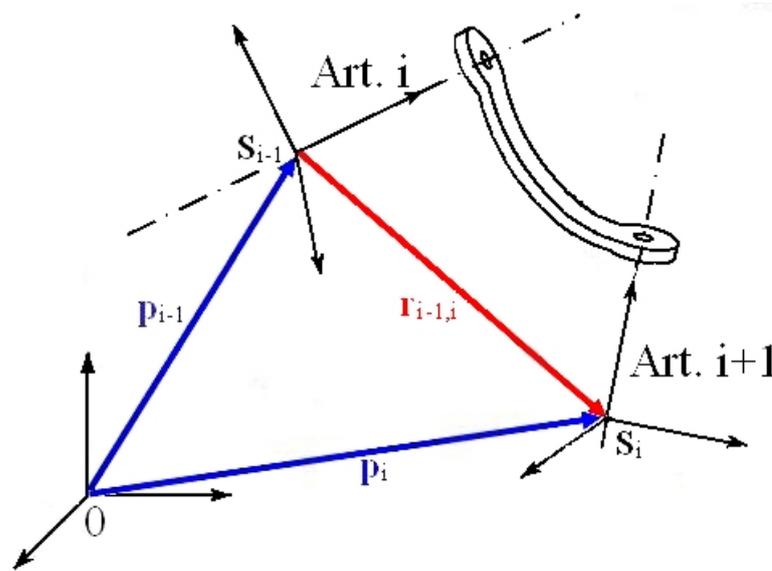


Figura 2.7: Sistemas de coordenadas asociadas a un eslabón

De la Figura 2.7 podemos escribir:

$$\mathbf{p}_i = \mathbf{p}_{i-1} + \mathbf{R}_{i-1} \mathbf{r}_{i-1,i}^{i-1} \quad (2.23)$$

Derivando la ecuación anterior obtenemos:

$$\dot{\mathbf{p}}_i = \dot{\mathbf{p}}_{i-1} + \mathbf{R}_{i-1} \dot{\mathbf{r}}_{i-1,i}^{i-1} + \boldsymbol{\omega}_{i-1} \times \mathbf{R}_{i-1} \mathbf{r}_{i-1,i}^{i-1} = \dot{\mathbf{p}}_{i-1} + v_{i-1,i} + \boldsymbol{\omega}_{i-1} \times \mathbf{r}_{i-1,i} \quad (2.24)$$

La ecuación de arriba expresa la **velocidad lineal del eslabón i** en función de la velocidad lineal y rotacional del eslabón $i-1$. Note que $v_{i-1,i}$ denota la velocidad del origen del sistema i con respecto al origen del sistema $i-1$.

Para la velocidad angular del eslabón i , recordamos la siguiente expresión:

$$\omega_i = \omega_{i-1} + \omega_{i-1,i} \quad (2.25)$$

La ecuación de arriba expresa la **velocidad angular del eslabón i** en función de las velocidades angulares del eslabón $i-1$, y del eslabón i relativa al eslabón $i-1$.

Las ecuaciones (2.24) y (2.25) dependen del tipo de articulación (*prismática o de revolución*), es decir:

Articulación Prismática En este tipo de articulación la velocidad angular relativa es nula, esto es:

$$\omega_{i-1,i} = 0$$

La velocidad relativa entre sistemas viene dado por:

$$v_{i-1,i} = \dot{d}_i \mathbf{z}_{i-1}$$

Donde \mathbf{z}_{i-1} es el vector unitario del eje de la articulación i .

Por tanto las ecuaciones (2.24) y (2.25) se reescriben como:

$$\omega_i = \omega_{i-1} \quad (2.26)$$

$$\dot{\mathbf{p}}_i = \dot{\mathbf{p}}_{i-1} + \dot{d}_i \mathbf{z}_{i-1} + \omega_i \times \mathbf{r}_{i-1,i} \quad (2.27)$$

Articulación Revolución La velocidad angular relativa entre sistemas con articulación de revolución viene dada por:

$$\omega_{i-1,i} = \dot{\theta}_i \mathbf{z}_{i-1}$$

Mientras la velocidad lineal relativa, se expresa como:

$$\mathbf{v}_{i-1,i} = \omega_{i-1,i} \times \mathbf{r}_{i-1,i}$$

Por tanto las ecuaciones (2.24) y (2.25) se reescriben como:

$$\omega_i = \omega_{i-1} + \dot{\theta}_i \mathbf{z}_{i-1} \quad (2.28)$$

$$\dot{\mathbf{p}}_i = \dot{\mathbf{p}}_{i-1} + \omega_i \times \mathbf{r}_{i-1,i} \quad (2.29)$$

Agrupando las ecuaciones anteriormente obtenidas,

$$\omega_i = \begin{cases} \omega_{i-1} & \text{para articulación prismática} \\ \omega_{i-1} + \dot{\theta}_i \mathbf{z}_{i-1} & \text{para articulación rotativa} \end{cases} \quad (2.30)$$

y

$$\dot{\mathbf{p}}_i = \begin{cases} \dot{\mathbf{p}}_{i-1} + \dot{d}_i \mathbf{z}_{i-1} + \omega_i \times \mathbf{r}_{i-1,i} & \text{para articulación prismática} \\ \dot{\mathbf{p}}_{i-1} + \omega_i \times \mathbf{r}_{i-1,i} & \text{para articulación rotativa} \end{cases} \quad (2.31)$$

Si a las ecuaciones (2.28) y (2.29) las premultiplicamos por \mathbf{R}_i^T , tendremos:

Para la Velocidad Angular

$$\omega_i^i = (\mathbf{R}_i^{i-1})^T \left\{ \omega_{i-1}^{i-1} + \dot{\theta}_i \mathbf{z}_0 \right\} \quad (2.32)$$

Para la Velocidad Lineal

$$\dot{\mathbf{p}}_i^i = (\mathbf{R}_i^{i-1})^T \dot{\mathbf{p}}_{i-1}^{i-1} + S(\omega_i^i) \mathbf{R}_i^T (\mathbf{p}_i - \mathbf{p}_{i-1}) \quad (2.33)$$

Las ecuaciones (2.32) y (2.33), permiten calcular de manera recursiva las velocidades lineales y angulares de un eslabón en función de las velocidades del que lo precede.

2.8. Aceleración entre Sistemas de Denavit-Hartenberg

Para el cálculo de las aceleraciones entre los sistemas de Denavit-Hartenberg (D-H) haremos uso de las expresiones de velocidad halladas en la sección anterior, las mismas que nuevamente escribimos:

$$\omega_i = \begin{cases} \omega_{i-1} & \text{para articulación prismática} \\ \omega_{i-1} + \dot{\theta}_i \mathbf{z}_{i-1} & \text{para articulación rotativa} \end{cases} \quad (2.34)$$

y

$$\dot{\mathbf{p}}_i = \begin{cases} \dot{\mathbf{p}}_{i-1} + \dot{d}_i \mathbf{z}_{i-1} + \omega_i \times \mathbf{r}_{i-1,i} & \text{para articulación prismática} \\ \dot{\mathbf{p}}_{i-1} + \omega_i \times \mathbf{r}_{i-1,i} & \text{para articulación rotativa} \end{cases} \quad (2.35)$$

Para calcular las aceleraciones angulares del sistema de referencia solidario al eslabón, se procede a diferenciar con respecto al tiempo las expresiones de la ecuación (2.34).

Para el caso de articulación prismática, se tiene:

$$\dot{\omega}_i = \dot{\omega}_{i-1} \quad (2.36)$$

Mientras que para articulación rotativa, se tiene:

$$\dot{\omega}_i = \dot{\omega}_{i-1} + \ddot{\theta}_i \mathbf{z}_{i-1} + \dot{\theta}_i \omega_{i-1} \times \mathbf{z}_{i-1} \quad (2.37)$$

Para hallar las expresiones de aceleración lineal de los eslabones diferenciamos con respecto al tiempo las expresiones dadas en la ecuación (2.35).

Para el caso de articulación prismática, se tiene:

$$\ddot{\mathbf{p}}_i = \ddot{\mathbf{p}}_{i-1} + \ddot{d}_i \mathbf{z}_{i-1} + \dot{d}_i \omega_{i-1} \times \mathbf{z}_{i-1} + \dot{\omega}_i \times \mathbf{r}_{i-1,i} + \omega_i \times \dot{d}_i \mathbf{z}_{i-1} + \omega_i \times (\omega_{i-1} \times \mathbf{r}_{i-1,i}) \quad (2.38)$$

para derivar la ecuación (2.38) se ha considerado $\dot{\mathbf{r}}_{i-1,i} = \dot{d}_i \mathbf{z}_{i-1} + \omega_{i-1} \times \mathbf{r}_{i-1,i}$. Sin embargo en vista de la ecuación (2.34) la ecuación (2.38) puede reescribirse como:

$$\ddot{\mathbf{p}}_i = \ddot{\mathbf{p}}_{i-1} + \ddot{d}_i \mathbf{z}_{i-1} + 2 \dot{d}_i \omega_i \times \mathbf{z}_{i-1} + \dot{\omega}_i \times \mathbf{r}_{i-1,i} + \omega_i \times (\omega_i \times \mathbf{r}_{i-1,i}) \quad (2.39)$$

Mientras que para articulación rotativa, se tiene:

$$\ddot{\mathbf{p}}_i = \ddot{\mathbf{p}}_{i-1} + \dot{\omega}_i \times \mathbf{r}_{i-1,i} + \omega_i \times (\omega_i \times \mathbf{r}_{i-1,i}) \quad (2.40)$$

Resumiendo las ecuaciones anteriores se tiene:

$$\dot{\omega}_i = \begin{cases} \dot{\omega}_{i-1} & \text{para articulación prismática} \\ \dot{\omega}_{i-1} + \ddot{\theta}_i \mathbf{z}_{i-1} + \dot{\theta}_i \omega_{i-1} \times \mathbf{z}_{i-1} & \text{para articulación rotativa} \end{cases} \quad (2.41)$$

y

$$\ddot{\mathbf{p}}_i = \begin{cases} \ddot{\mathbf{p}}_{i-1} + \ddot{d}_i \mathbf{z}_{i-1} + 2 \dot{d}_i \omega_i \times \mathbf{z}_{i-1} \\ \quad + \dot{\omega}_i \times \mathbf{r}_{i-1,i} + \omega_i \times (\omega_i \times \mathbf{r}_{i-1,i}) & \text{para articulación prismática} \\ \ddot{\mathbf{p}}_{i-1} + \dot{\omega}_i \times \mathbf{r}_{i-1,i} + \omega_i \times (\omega_i \times \mathbf{r}_{i-1,i}) & \text{para articulación rotativa} \end{cases} \quad (2.42)$$

Todas las expresiones halladas hasta este punto muestran que el movimiento de un eslabón está acoplado al movimiento de los otros eslabones a través de relaciones de velocidad y aceleraciones.

2.9. Algoritmo Recursivo de Newton-Euler

Las ecuaciones de Newton-Euler permiten encontrar las fuerzas y los momentos de inercia que actúan sobre cada eslabón, esto se consigue siguiendo los siguientes pasos, ver [5], [7]:

1. Asignar a cada eslabón un sistema de referencia de acuerdo con la convención de Denavit-Hartenberg.

2. Establecer las condiciones iniciales. Para el sistema de la base S_0 :

$$\omega_0^0 : \text{velocidad angular} = [0, 0, 0]^T.$$

$$\dot{\omega}_0^0 : \text{aceleración angular} = [0, 0, 0]^T.$$

$$\dot{\mathbf{p}}_0^0 : \text{velocidad lineal} = [0, 0, 0]^T.$$

$$\ddot{\mathbf{p}}_0^0 : \text{aceleración lineal} = -[g_{xo}, g_{yo}, g_{zo}]^T \text{ (es la gravedad expresado en el sistema } S_0), \text{ ver [5].}$$

3. Obtener las matrices de rotación \mathbf{R}_i^{i-1} , siendo ésta:

$$\mathbf{R}_i^{i-1} = \begin{bmatrix} \cos(\theta_i) & -\cos(\alpha_i)\sin(\theta_i) & \sin(\alpha_i)\sin(\theta_i) \\ \sin(\theta_i) & \cos(\alpha_i)\cos(\theta_i) & -\sin(\alpha_i)\cos(\theta_i) \\ 0 & \sin(\alpha_i) & \cos(\alpha_i) \end{bmatrix}$$

4. Obtener la velocidad angular del sistema de referencia S_i , para ello premultiplicamos \mathbf{R}_i^T con las ecuaciones (2.30).

$$\omega_i^i = \begin{cases} (\mathbf{R}_i^{i-1})^T \omega_{i-1}^{i-1} & \text{para articulación prismática} \\ (\mathbf{R}_i^{i-1})^T (\omega_{i-1}^{i-1} + \dot{\theta}_i \mathbf{z}_0) & \text{para articulación rotativa} \end{cases} \quad (2.43)$$

5. Obtener la aceleración angular del sistema de referencia S_i , para ello pre-

multiplicamos \mathbf{R}_i^T con las ecuaciones (2.41).

$$\dot{\omega}_i^i = \begin{cases} (\mathbf{R}_i^{i-1})^T \dot{\omega}_{i-1}^{i-1} & \text{para articulación prismática} \\ (\mathbf{R}_i^{i-1})^T (\dot{\omega}_{i-1}^{i-1} + \ddot{\theta}_i \mathbf{z}_0 + \dot{\theta}_i \omega_{i-1}^{i-1} \times \mathbf{z}_0) & \text{para articulación rotativa} \end{cases} \quad (2.44)$$

6. Obtener las aceleración lineal del sistema de referencia S_i , para ello premultiplicamos \mathbf{R}_i^T con las ecuaciones (2.42).

$$\ddot{\mathbf{p}}_i^i = \begin{cases} (\mathbf{R}_i^{i-1})^T (\ddot{\mathbf{p}}_{i-1}^{i-1} + \ddot{d}_i \mathbf{z}_0) + 2 \dot{d}_i \omega_i^i \times (\mathbf{R}_i^{i-1})^T \mathbf{z}_0 \\ \quad + \dot{\omega}_i^i \times \mathbf{r}_{i-1,i}^i + \omega_i^i \times (\omega_i^i \times \mathbf{r}_{i-1,i}^i) & \text{art. prismática} \\ (\mathbf{R}_i^{i-1})^T \ddot{\mathbf{p}}_{i-1}^{i-1} + \dot{\omega}_i^i \times \mathbf{r}_{i-1,i}^i + \omega_i^i \times (\omega_i^i \times \mathbf{r}_{i-1,i}^i) & \text{art. rotativa} \end{cases} \quad (2.45)$$

7. Obtener las aceleración lineal del centro de gravedad del eslabón i , dada por la siguiente expresión:

$$\ddot{\mathbf{p}}_{C_i}^i = \ddot{\mathbf{p}}_i^i + \omega_i^i \times \mathbf{r}_{i,C_i}^i + \dot{\omega}_i^i \times (\omega_i^i \times \mathbf{r}_{i,C_i}^i) \quad (2.46)$$

Donde, \mathbf{r}_{i,C_i}^i es el vector que une el origen del sistema S_i con el centro de masa del eslabón i representado en el sistema de referencia S_i .

8. Obtener la fuerza ejercida sobre cada eslabón i ,

$$\mathbf{f}_i^i = \mathbf{R}_{i+1}^i \mathbf{f}_{i+1}^{i+1} + m_i \ddot{\mathbf{p}}_{C_i}^i \quad (2.47)$$

9. Obtener el torque ejercido sobre cada eslabón i ,

$$\mu_i^i = -\mathbf{f}_i^i \times (\mathbf{r}_{i,C_i}^i + \mathbf{r}_{i-1,i}^i) + \mathbf{R}_{i+1}^i \mu_{i+1}^{i+1} + \mathbf{R}_{i+1}^i \mathbf{f}_{i+1}^{i+1} \times \mathbf{r}_{i,C_i}^i + \bar{\mathbf{I}}_i^i \dot{\omega}_i^i + \omega_i^i \times (\bar{\mathbf{I}}_i^i \omega_i^i) \quad (2.48)$$

10. Obtener la fuerza o par aplicado a la articulación i .

$$\tau_i = \begin{cases} (f_i^i)^T \mathbf{R}_i^{i-1} \mathbf{z}_0 \\ (\mu_i^i)^T \mathbf{R}_i^{i-1} \mathbf{z}_0 \end{cases} \quad (2.49)$$

Finalmente, acomodando la expresión de los torques obtenidos por Newton-Euler a la forma Lagrangiana, se tiene:

$$\tau_1 = \mathbf{M}_{11}\ddot{q}_1 + \mathbf{M}_{12}\ddot{q}_2 + \mathbf{M}_{13}\ddot{q}_3 + \mathbf{C}_1 + \mathbf{G}_1 \quad (2.50)$$

$$\tau_2 = \mathbf{M}_{21}\ddot{q}_1 + \mathbf{M}_{22}\ddot{q}_2 + \mathbf{M}_{23}\ddot{q}_3 + \mathbf{C}_2 + \mathbf{G}_2 \quad (2.51)$$

$$\tau_3 = \mathbf{M}_{31}\ddot{q}_1 + \mathbf{M}_{32}\ddot{q}_2 + \mathbf{M}_{33}\ddot{q}_3 + \mathbf{C}_3 + \mathbf{G}_3 \quad (2.52)$$

El modelo dinámico del robot se ha obtenido usando matemática simbólica de Matlab (ver Apéndice ??).

2.10. Modelo del Actuador DC

En esta sección se determinarán las ecuaciones que gobiernan la dinámica de los motores de corriente continua. Al final se obtiene una expresión que relaciona el torque desarrollado por el motor con el voltaje que se le aplica en sus extremos. En este trabajo se usarán 3 motores de la marca Pittman. Estos motores también tienen incorporados un encoder incremental para sensar la posición.

2.10.1. Modelado del Motor N°1 de la Base del Robot

En la figura 2.8, se muestra el modelo eléctrico y mecánico del motorreductor DC1 que es el responsable de dar movimiento al primer eslabón de robot.

El voltaje de control es amplificado en un factor \mathbf{K}_1 el resultado es aplicado al circuito eléctrico de la armadura y viene dado por:

$$V_{A1} = K_1 u_1 = L_1 \dot{I}_{a1} + R_1 I_{a1} + V_{b1} \quad (2.53)$$

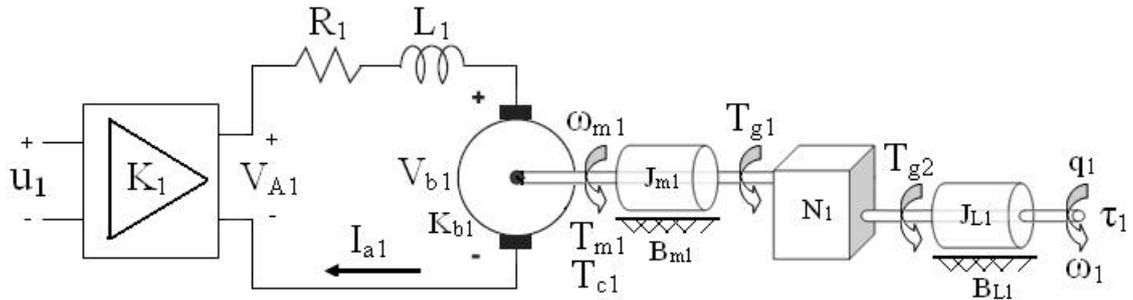


Figura 2.8: Modelo eléctrico y mecánico del motor reductor DC1

además la FEM⁴ del motor DC, es:

$$V_{b1} = K_{b1}\omega_{m1} = K_{b1}N_1\dot{q}_1 \quad (2.54)$$

donde, N_1 es la reducción de la caja de engranajes. Además se sabe que T_{m1} viene dado por:

$$T_{m1} = K_{m1}I_{a1} = J_{m1}N_1\ddot{q}_1 + B_{m1}N_1\dot{q}_1 + T_{g1} \quad (2.55)$$

en el eje de salida el torque T_{g2} está dado por:

$$T_{g2} = N_1T_{g1} = J_{L1}\ddot{q}_1 + B_{L1}\dot{q}_1 + N_1T_{c1} + \tau_1 \quad (2.56)$$

donde: J_{L1} : Inercia de la carga ($Kg\ m^2$), B_{L1} : Coeficiente de fricción ($Nm\ s/rad$) y T_{c1} : Torque de Coulumb ($N\ m$).

reemplazando la ecuación (2.56) en (2.55) y agrupando convenientemente obtenemos:

$$N_1K_{m1}I_{a1} = J_{eq1}\ddot{q}_1 + B_{eq1}\dot{q}_1 + N_1T_{c1} + \tau_1 \quad (2.57)$$

donde J_{eq1} : Inercia Equivalente ($Kg\ m^2$), B_{eq1} : Coeficiente de fricción Equivalente ($Nm\ s/rad$).

⁴Fuerza Electromotriz

despreciando la inductancia de la armadura, la ecuación (2.53) se reduce a:

$$K_{A1}u_1 = R_1I_{a1} + K_{b1}N_1\dot{q}_1 \quad (2.58)$$

despejando la I_{a1} de la ecuación (2.57) y reemplazandola en (2.58), obtenemos la ley de control para la primera entrada.

$$u_1 = R_{T1}(J_{eq1}\ddot{q}_1 + B_{eq1}\dot{q}_1 + \tau_1) + N_{T1}\dot{q}_1 + N_1R_{T1}T_{c1} \quad (2.59)$$

donde $R_{T1} : \frac{R_1}{N_1K_{m1}K_1}$, $N_{T1} : \frac{N_1K_{b1}}{K_1}$ y $T_{c1} : C_{c1}sign(\dot{q}_1)$.

2.10.2. Modelado del Motor N°2 del brazo del Robot

En la figura 2.9, se muestra el modelo electromecánico del motorreductor DC2 que es el responsable de dar movimiento al segundo eslabón.

Este motor se encuentra en la parte superior del primer eslabón sobre un disco

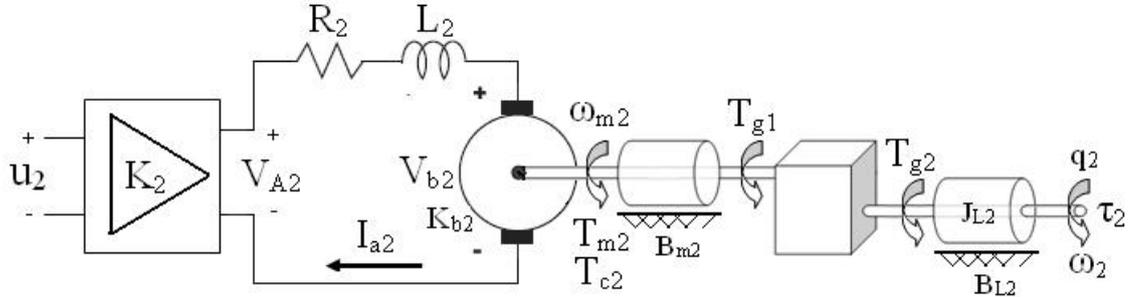


Figura 2.9: Modelo eléctrico y mecánico del motor reductor DC2

tal como se muestra en la figura 5.3. Para obtener la ley de control de la segunda entrada se procede de manera análoga como se hizo para el motor DC1, obteniendo:

$$u_2 = R_{T2}(J_{eq2}\ddot{q}_2 + B_{eq2}\dot{q}_2 + \tau_2) + N_{T2}\dot{q}_2 + N_2R_{T2}T_{c2} \quad (2.60)$$

donde $R_{T2} : \frac{R_2}{N_2K_{m2}K_2}$, $N_{T2} : \frac{N_2K_{b2}}{K_2}$ y $T_{c2} : C_{c2}sign(\dot{q}_2)$.

2.10.3. Modelado del Motor N°3 del Antebrazo del Robot

En la figura 2.10, se muestra el modelo electromecánico del motorreductor DC3 que es el responsable de dar movimiento al tercer eslabón. Este motor se encuentra en un extremo del brazo del 2do. eslabón, tal como se muestra en la figura 5.3. Para obtener la ley de control de la segunda entrada se procede de manera análoga como se hizo para el motor DC1, obteniendo:

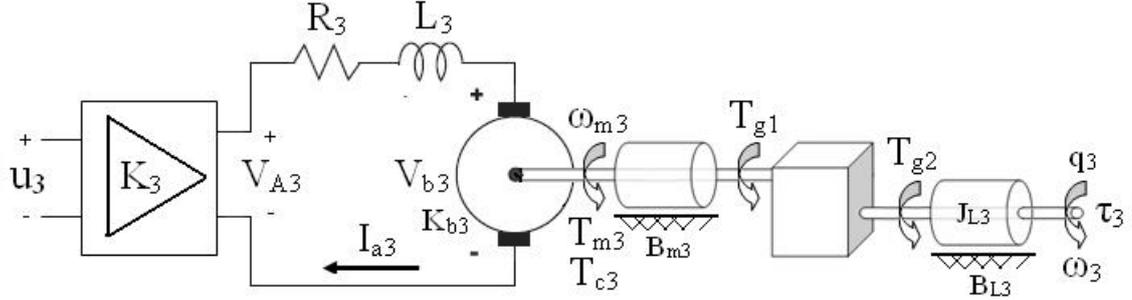


Figura 2.10: Modelo eléctrico y mecánico del motor reductor DC3

$$u_3 = R_{T3}(J_{eq3}\ddot{q}_3 + B_{eq3}\dot{q}_3 + \tau_3) + N_{T3}\dot{q}_3 + N_3R_{T3}T_{c3} \quad (2.61)$$

donde $R_{T3} : \frac{R_3}{N_3K_{m3}K_3}$, $N_{T3} : \frac{N_3K_{b3}}{K_3}$ y $T_{c3} : C_{c3}sign(\dot{q}_3)$.

Finalmente reemplazando las ecuaciones (2.50), (2.51) y (2.52) en (2.59), (2.60) y (2.61) respectivamente, se obtendrá:

$$\begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} = \begin{bmatrix} R_{T1}(M_{11} + J_{eq1}) & R_{T1}M_{12} & R_{T1}M_{13} \\ R_{T2}M_{22} & R_{T2}(M_{22} + J_{eq2}) & R_{T2}M_{23} \\ R_{T3}M_{31} & R_{T3}M_{32} & R_{T3}(M_{33} + J_{eq3}) \end{bmatrix} \begin{bmatrix} \ddot{q}_1 \\ \ddot{q}_2 \\ \ddot{q}_3 \end{bmatrix} + \begin{bmatrix} (R_{T1}B_{eq1} + N_{T1})\dot{q}_1 + R_{T1}C_1 \\ (R_{T2}B_{eq2} + N_{T2})\dot{q}_2 + R_{T2}C_2 \\ (R_{T3}B_{eq3} + N_{T3})\dot{q}_3 + R_{T3}C_3 \end{bmatrix} + \begin{bmatrix} R_{T1}G_1 \\ R_{T2}G_2 \\ R_{T3}G_3 \end{bmatrix} \quad (2.62)$$

2.11. Modelo en el Espacio Estado

La ecuación dinámica del manipulador incluyendo los motores obtenida en (2.62), para que sea representada en el espacio de estados se hacen las siguientes asignaciones de variables de estado:

donde,

Nombre de la variable	Asignación
x_1 Posición angular de la base	$x_1 = q_1$
x_2 Posición angular del brazo	$x_2 = q_2$
x_3 Posición angular del antebrazo	$x_3 = q_3$
x_4 Velocidad angular de la base	$x_4 = \dot{q}_1$
x_5 Velocidad angular del brazo	$x_5 = \dot{q}_2$
x_6 Velocidad angular del antebrazo	$x_6 = \dot{q}_3$

la ecuación en espacio estado no lineal multivariable viene dado por:

$$\dot{\mathbf{x}} = \mathbf{F}(\mathbf{x}) + \mathbf{G}(\mathbf{x})\mathbf{u} \quad (2.63)$$

la misma que en forma matricial viene dado por:

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \\ \dot{x}_5 \\ \dot{x}_6 \end{bmatrix} = \begin{bmatrix} F_1(x) \\ F_2(x) \\ F_3(x) \\ F_4(x) \\ F_5(x) \\ F_6(x) \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ G_{41}(x) & G_{42}(x) & G_{43}(x) \\ G_{51}(x) & G_{52}(x) & G_{53}(x) \\ G_{61}(x) & G_{62}(x) & G_{63}(x) \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} \quad (2.64)$$

con salidas,

$$\mathbf{y} = \mathbf{h}(x) = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} \quad (2.65)$$

fácilmente se pueden identificar las siguientes funciones:

$$F_1(x) = x_4$$

$$F_2(x) = x_5$$

$$F_3(x) = x_6$$

Las funciones $F_i(x), G_{ij} \forall i = 4, 5, 6 \wedge \forall j = 1, 2, 3$ se calculan mediante el programa del Apéndice ??.

CAPÍTULO 3

Control por Linealización Exacta por Realimentación de Estado

En este capítulo trataremos de la técnica de Control por Linealización Exacta por Realimentación de Estado (CLERE) que se inició alrededor de los años '70 con intentos de extender resultados de la teoría de sistemas lineales a sistemas no lineales; tales como controlabilidad y observabilidad del sistema.

En 1981, Isidori se encargó de mostrar que estas extensiones eran posibles. El artículo de Isidori fue seguido durante los años '80 por un gran número de resultados que generaron un fantástico *juego de herramientas* para sistemas no lineales (Isidori, 1995; Isidori, 1999).

El CLERE es concebido como dos lazos de realimentación, un primer lazo que transforma el sistema original en un sistema lineal mediante una realimentación no lineal de las variables de estado y un segundo lazo externo que controla el sistema lineal resultante. Antes de desarrollar esta técnica de control se darán algunos conceptos previos de geometría diferencial.

3.1. Herramientas de la Geometría Diferencial

En esta sección se dan a conocer ciertos conceptos básicos, como campo vectorial y gradiente de una función. La notación:

$$\mathbf{f} : R^n \rightarrow R^n \tag{3.1}$$

indica que a cada función vectorial \mathbf{f} le corresponde un campo vectorial en el espacio n -dimensional. Los campos vectoriales $\mathbf{f}(\mathbf{x})$ tratados aquí son denominados *suaves*. Se dice que un campo $\mathbf{f}(\mathbf{x})$ es suave cuando tal función vectorial posee infinitas derivadas parciales continuas.

Si $h(\mathbf{x})$ es una función escalar de variable vectorial suave, donde \mathbf{x} es el vector de estado, entonces su gradiente es:

$$\nabla h = \frac{\partial h}{\partial \mathbf{x}} = \left[\frac{\partial h}{\partial x_1} \quad \cdots \quad \frac{\partial h}{\partial x_n} \right] \quad (3.2)$$

De la misma manera el jacobiano de $\mathbf{f}(\mathbf{x})$, se define como:

$$\nabla \mathbf{f}(\mathbf{x}) = \frac{\partial \mathbf{f}(\mathbf{x})}{\partial \mathbf{x}} = \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \cdots & \frac{\partial f_1}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_n}{\partial x_1} & \cdots & \frac{\partial f_n}{\partial x_n} \end{bmatrix} \quad (3.3)$$

3.2. Derivadas y Corchetes de Lie

El método de linealización exacta por realimentación de estados se ayuda de la derivada y corchete de Lie para determinar si un sistema no lineal puede describirse mediante una dinámica que parezca lineal luego de efectuar sobre el sistema una apropiada transformación de coordenadas y una realimentación no lineal del vector de estado.

Conseguida tal descripción lineal del sistema se pueden emplear técnicas de control lineal, tal como ubicación de polos.

Definición 3.2.1 *Derivada de Lie:*

Dada una función escalar suave $h(\mathbf{x}): R^n \rightarrow R$ y un campo vectorial suave $\mathbf{f}(\mathbf{x}): R^n \rightarrow R^n$, se define una nueva función escalar $L_{\mathbf{f}}h = \nabla h \mathbf{f}$, denominada derivada de Lie, o simplemente derivada direccional de h a lo largo de la dirección del vector \mathbf{f} .

Sucesivas derivadas de Lie pueden ser formuladas de forma recursiva:

$$L_{\mathbf{f}}^0 h = h \quad (3.4)$$

$$L_{\mathbf{f}}^i h = L_{\mathbf{f}}(L_{\mathbf{f}}^{i-1} h) = \nabla(L_{\mathbf{f}}^{i-1} h)\mathbf{f}, \quad \text{para } i=1,2,\dots \quad (3.5)$$

Del mismo modo, si \mathbf{g} es otro campo vectorial, la función escalar $L_{\mathbf{g}}L_{\mathbf{f}}h(\mathbf{x})$ es:

$$L_{\mathbf{g}}L_{\mathbf{f}}h(\mathbf{x}) = \nabla(L_{\mathbf{f}}h)\mathbf{g} \quad (3.6)$$

Definición 3.2.2 Corchete de Lie:

Sean \mathbf{f} y \mathbf{g} dos campos vectoriales en R^n . El corchete de Lie de \mathbf{f} y \mathbf{g} se define como un tercer campo vectorial de la forma:

$$[\mathbf{f}, \mathbf{g}] = \nabla\mathbf{g}\mathbf{f} - \nabla\mathbf{f}\mathbf{g} = ad_{\mathbf{f}}\mathbf{g} \quad (3.7)$$

Sucesivos corchetes de Lie ($ad_{\mathbf{f}}\mathbf{g}$) se pueden formular en forma recursiva.

$$ad_{\mathbf{f}}^0 \mathbf{g} = \mathbf{g} \quad (3.8)$$

$$ad_{\mathbf{f}}^i \mathbf{g} = [\mathbf{f}, ad_{\mathbf{f}}^{i-1} \mathbf{g}] \quad \text{para } i=1,2,\dots \quad (3.9)$$

Si consideramos los siguientes campos vectoriales suaves \mathbf{f} , \mathbf{f}_1 , \mathbf{f}_2 , \mathbf{g} , \mathbf{g}_1 y \mathbf{g}_2 y α_1 y α_2 sean constantes escalares, $h(\mathbf{x})$ sea una función escalar suave, entonces se puede enunciar las siguientes propiedades del corchete de Lie:

1. Bilinealidad:

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

2. Anti-conmutatividad:

$$[\mathbf{f}, \mathbf{g}] = -[\mathbf{g}, \mathbf{f}]$$

3. Identidad de Jacobi:

$$L_{ad_{\mathbf{f}}\mathbf{g}} h = L_{\mathbf{f}}L_{\mathbf{g}}h - L_{\mathbf{g}}L_{\mathbf{f}}h$$

3.3. Difeomorfismos y Transformación de Coordenadas

El concepto de difeomorfismo puede ser tratado como una generalización del concepto de transformación de coordenadas.

Definición 3.3.1 *Difeomorfismo:*

Una función $\phi: R^n \rightarrow R^n$ definida en una región Ω , se denomina un difeomorfismo si ϕ y su inversa son suaves.

Si $\phi(\mathbf{x})$ se define en una región finita Ω en un punto $x = x_0$, entonces es un difeomorfismo local.

El siguiente lema permite determinar difeomorfismos locales.

Lema 3.3.1 *Sea $\phi(\mathbf{x})$ una función suave definida en una región Ω en R^n . Si el Jacobiano $\nabla\phi$ es no singular en un punto $\mathbf{x} = \bar{\mathbf{x}}$ de Ω , entonces $\phi(\mathbf{x})$ define un difeomorfismo local en una región de Ω .*

Empleando el concepto de difeomorfismo, podemos transformar los estados de un sistema no lineal en un nuevo conjunto de estados.

Consideremos un sistema SISO no lineal descrito por:

$$\begin{aligned}\dot{\mathbf{x}} &= \mathbf{f}(\mathbf{x}) + \mathbf{g}(\mathbf{x})u \\ y &= h(\mathbf{x})\end{aligned}\tag{3.10}$$

Definamos un nuevo conjunto de estados $\mathbf{z} = \Phi(\mathbf{x})$ y efectuando diferenciación respecto a \mathbf{z} , tenemos:

$$\begin{aligned}\dot{\mathbf{z}} &= \frac{\partial\Phi}{\partial\mathbf{x}}\dot{\mathbf{x}} = \frac{\partial\Phi}{\partial\mathbf{x}}(\mathbf{f}(\mathbf{x}) + \mathbf{g}(\mathbf{x})u) \\ \Rightarrow \dot{\mathbf{z}} &= \mathbf{f}^*(\mathbf{z}) + \mathbf{g}^*(\mathbf{z})u, \text{ y} \\ y &= h^*(\mathbf{z})\end{aligned}\tag{3.11}$$

Donde:

$$\mathbf{f}^* = \frac{\partial\Phi}{\partial\mathbf{x}}\mathbf{f} \quad \mathbf{g}^* = \frac{\partial\Phi}{\partial\mathbf{x}}\mathbf{g} \quad \mathbf{x} = \Phi^{-1}(\mathbf{z})$$

3.4. El Teorema de Frobenius

El teorema de Frobenius es una herramienta formal importante para el tratamiendo formal de la linealización por realimentación de procesos de orden n . Este teorema nos proporcionará las condiciones necesarias y suficientes para resolver una clase especial de ecuaciones con derivadas parciales. Antes de mencionar dicho teorema se darán los siguientes conceptos previos.

Definición 3.4.1 *Completamente Integrable:*

Se dice que un conjunto linealmente independiente de campos vectoriales $(\mathbf{f}_1, \dots, \mathbf{f}_m)$ en R^n es completamente integrable, si y solo si 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:

$$\nabla h_i \mathbf{f}_j = 0 \quad (3.12)$$

Donde, $1 \leq i \leq n - m, 1 \leq j \leq m$. Además, los gradientes ∇h_i son linealmente independientes.

Definición 3.4.2 *Involutivo:*

Se dice que un conjunto linealmente independiente de campos vectoriales $(\mathbf{f}_1, \dots, \mathbf{f}_m)$ en R^n es involutivo, si y sólo si existen funciones escalares α_{ijk} , tales que:

$$[\mathbf{f}_i, \mathbf{f}_j] = \sum_{k=1}^m \alpha_{ijk}(\mathbf{x}) \mathbf{f}_k(\mathbf{x}), \quad \forall i, j \quad (3.13)$$

De acuerdo a la definición anterior podemos deducir lo siguiente:

1. Un conjunto compuesto por un solo campo vectorial \mathbf{f} es involutivo porque $[\mathbf{f}, \mathbf{f}] = 0$.
2. Si un conjunto de campos vectoriales $[\mathbf{f}_1, \dots, \mathbf{f}_m]$ es involutivo, esto significa que para todo i, j :

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

Finalmente el teorema de Frobenius se enuncia como:

Teorema 3.4.1 Teorema de Frobenius:

Sea $(\mathbf{f}_1, \dots, \mathbf{f}_m)$ un conjunto de campos vectoriales linealmente independiente. Se dice que tal conjunto es integrable si y sólo si es integrable.

3.4.1. El Grado Relativo de un Sistema

Dado el siguiente proceso SISO no lineal,

$$\begin{aligned}\dot{\mathbf{x}} &= \mathbf{f}(\mathbf{x}) + \mathbf{g}(\mathbf{x})u \\ y &= h(\mathbf{x})\end{aligned}\tag{3.14}$$

Para obtener una relación directa entre la entrada u y la salida y , derivamos la salida como sigue a continuación:

$$\begin{aligned}\dot{y} &= \frac{\partial h(\mathbf{x})}{\partial \mathbf{x}} \dot{\mathbf{x}} = \frac{\partial h(\mathbf{x})}{\partial \mathbf{x}} \mathbf{f}(\mathbf{x}) + \frac{\partial h(\mathbf{x})}{\partial \mathbf{x}} \mathbf{g}(\mathbf{x})u \\ &= L_{\mathbf{f}}h(\mathbf{x}) + L_{\mathbf{g}}h(\mathbf{x})u\end{aligned}\tag{3.15}$$

Si $L_{\mathbf{g}}h(\mathbf{x}) \neq 0$, entonces se obtiene la relación deseada entre y y u . En caso contrario, debemos seguir derivando la salida $\dot{y} = L_{\mathbf{f}}h(\mathbf{x})$, tal como sigue:

$$\begin{aligned}\ddot{y} &= \frac{\partial}{\partial \mathbf{x}} L_{\mathbf{f}}h(\mathbf{x})\dot{\mathbf{x}} \\ &= \frac{\partial}{\partial \mathbf{x}} L_{\mathbf{f}}h(\mathbf{x})\mathbf{f}(\mathbf{x}) + \frac{\partial}{\partial \mathbf{x}} L_{\mathbf{f}}h(\mathbf{x})\mathbf{g}(\mathbf{x})u \\ \ddot{y} &= L_{\mathbf{f}}^2h(\mathbf{x}) + L_{\mathbf{g}}L_{\mathbf{f}}h(\mathbf{x})u\end{aligned}\tag{3.16}$$

Nuevamente, si $L_{\mathbf{g}}L_{\mathbf{f}}h(\mathbf{x}) \neq 0$, entonces el proceso finaliza. En caso contrario, se debe seguir derivando la salida hasta que en la ρ -ésima derivada la entrada u aparezca en forma explícita, tal como se muestra:

$$y^\rho = L_{\mathbf{f}}^\rho h(\mathbf{x}) + L_{\mathbf{g}}L_{\mathbf{f}}^{\rho-1}h(\mathbf{x})u; \quad L_{\mathbf{g}}L_{\mathbf{f}}^{\rho-1}h(\mathbf{x}) \neq 0\tag{3.17}$$

Definición 3.4.3 Grado Relativo:

El grado relativo del proceso no lineal dado en la ecuación (3.14), viene dado por menor orden de la derivada de la salida en que la entrada aparece en forma explícita.

Por ejemplo si ρ es el grado relativo del sistema dado en la ecuación (3.14), entonces:

$$L_g L_f^i h(\mathbf{x}) = 0, \quad i=1,2,\dots,\rho-2 \quad (3.18)$$

$$L_g L_f^{\rho-1} h(\mathbf{x}) \neq 0 \quad (3.19)$$

para todo \mathbf{x} que pertenezca a una región Ω de R^n .

3.4.2. Forma Normal SISO con Linealización Exacta

La transformación no lineal de coordenadas, cuyos conceptos fueron establecidos en la sección 3.3, que permite llevar un proceso SISO no lineal (ecuación 3.14) a su forma normal, involucra términos como: $h(\mathbf{x})$, $L_f h(\mathbf{x})$, \dots , $L_f^{\rho-1} h(\mathbf{x})$.

Por lo tanto, la transformación depende del grado relativo del proceso.

Para su estudio, empezaremos por el caso más simple: $\rho=n$, donde n es el orden del proceso.

La siguiente transformación de coordenadas:

$$\mathbf{z} = \begin{bmatrix} z_1(\mathbf{x}) \\ z_2(\mathbf{x}) \\ \vdots \\ z_n(\mathbf{x}) \end{bmatrix} = \Phi(\mathbf{x}) = \begin{bmatrix} \phi_1(\mathbf{x}) \\ \phi_2(\mathbf{x}) \\ \vdots \\ \phi_n(\mathbf{x}) \end{bmatrix} = \begin{bmatrix} h(\mathbf{x}) \\ L_f h(\mathbf{x}) \\ \vdots \\ L_f^{n-1} h(\mathbf{x}) \end{bmatrix} = \begin{bmatrix} y(\mathbf{x}) \\ \dot{y}(\mathbf{x}) \\ \vdots \\ y^n(\mathbf{x}) \end{bmatrix} \quad (3.20)$$

con $\mathbf{x}=[x_1, x_2, \dots, x_n]^T$, cumple con las condiciones de una transformación no lineal, es decir, es un difeomorfismo considerando que el orden del sistema es $\rho=n$.

Por consiguiente, dicha transformación puede llevar al proceso descrito en la

ecuación (3.14) a la forma siguiente:

$$\begin{aligned}
\dot{z}_1 &= \frac{\partial \phi_1(\mathbf{x})}{\partial(\mathbf{x})} \dot{\mathbf{x}} \\
&= \frac{\partial \phi_1(\mathbf{x})}{\partial(\mathbf{x})} (\mathbf{f}(\mathbf{x}) + \mathbf{g}(\mathbf{x})u) \\
&= L_{\mathbf{f}}h(\mathbf{x}) + L_{\mathbf{g}}h(\mathbf{x})u \\
&= L_{\mathbf{f}}h(\mathbf{x}) = \phi_2(\mathbf{x}) \\
&= z_2
\end{aligned} \tag{3.21}$$

$\vdots = \vdots$, seguimos derivando,

$$\begin{aligned}
\dot{z}_{n-1} &= \frac{\partial \phi_{n-1}(\mathbf{x})}{\partial(\mathbf{x})} \dot{\mathbf{x}} \\
&= \frac{\partial L_{\mathbf{f}}^{n-2}h(\mathbf{x})}{\partial(\mathbf{x})} (\mathbf{f}(\mathbf{x}) + \mathbf{g}(\mathbf{x})u) \\
&= L_{\mathbf{f}}^{n-1}h(\mathbf{x}) + L_{\mathbf{g}}L_{\mathbf{f}}^{n-2}h(\mathbf{x})u \\
&= L_{\mathbf{f}}^{n-1}h(\mathbf{x}) = \phi_n(\mathbf{x}) \\
&= z_n
\end{aligned} \tag{3.22}$$

$$\begin{aligned}
\dot{z}_n &= L_{\mathbf{f}}^n h(\mathbf{x}) + L_{\mathbf{g}}L_{\mathbf{f}}^{n-1}h(\mathbf{x})u \\
&= L_{\mathbf{f}}^n h(\phi^{-1}(\mathbf{z})) + L_{\mathbf{g}}L_{\mathbf{f}}^{n-1}h(\phi^{-1}(\mathbf{z}))u \\
&= b(\mathbf{z}) + a(\mathbf{z})u, \quad y,
\end{aligned} \tag{3.23}$$

$$y = z_1 \tag{3.24}$$

Notar que en la ecuación (3.23), $a(\mathbf{z}) \neq 0$, por definición de grado relativo ($\rho=n$). Cuando el proceso no lineal posee un rango $\rho < n$, entonces el cambio de coordenadas desarrollado anteriormente nos conduce a una transformación parcial, en vista que sólo se pueden definir ρ funciones de la forma $\phi_1(\mathbf{x}), \dots, \phi_\rho(\mathbf{x})$ linealmente independientes. Además, es posible agregar $n-\rho$ funciones de la forma $\phi_{\rho+1}(\mathbf{x}), \phi_{\rho+2}(\mathbf{x}), \dots, \phi_n(\mathbf{x})$, de modo que $\phi(\mathbf{x})$ sea un difeomorfismo. Con esto la forma normal del proceso descrito en la ecuación (3.14) resulta la siguiente

representación en el espacio de estado:

$$\begin{aligned}
 \dot{z}_1 &= z_2 \\
 \dot{z}_2 &= z_3 \\
 &\vdots = \vdots \\
 \dot{z}_{\rho-1} &= z_\rho \\
 \dot{z}_\rho &= L_{\mathbf{f}}^\rho h(\phi^{-1}(\mathbf{z})) + L_{\mathbf{g}} L_{\mathbf{f}}^{\rho-1} h(\phi^{-1}(\mathbf{z}))u = b(\mathbf{z}) + a(\mathbf{z})u \\
 \dot{z}_{\rho+1} &= w_1(\mathbf{z}) \\
 &\vdots = \vdots \\
 \dot{z}_n &= w_{n-\rho}(\mathbf{z}) \\
 y &= z_1
 \end{aligned} \tag{3.25}$$

donde, $w_1, w_2, \dots, w_{n-\rho}$ son funciones suaves que dependen del nuevo estado \mathbf{z} y no de la entrada manipulada u .

3.5. Linealización Exacta y Ley de Control. Caso: SISO

La linealización exacta se refiere a tratar con procesos no lineales de grado $\rho=n$. En tales procesos es posible determinar una adecuada realimentación no lineal que conduzca a una linealización exacta del proceso original. Esta técnica puede ser aplicada a un proceso SISO (Single Input Single Output) o MIMO (Multiple Input Multiple Output). En esta sección sólo nos ocuparemos del caso SISO. Se ha observado que un cambio de coordenadas de la forma $\mathbf{z}=\phi(\mathbf{x})$, descrita en la ecuación (3.20) y aplicado al proceso no lineal de la ecuación (3.14), produce un conjunto de nuevas coordenadas que poseen la forma dada en la ecuación (3.25),

pero con $\rho=n$, la misma que escribiremos,

$$\begin{aligned}
 \dot{z}_1 &= z_2 \\
 \dot{z}_2 &= z_3 \\
 &\vdots \\
 \dot{z}_{n-1} &= z_n \\
 \dot{z}_n &= L_{\mathbf{f}}^n h(\phi^{-1}(\mathbf{z})) + L_{\mathbf{g}} L_{\mathbf{f}}^{n-1} h(\phi^{-1}(\mathbf{z}))u = b(\mathbf{z}) + a(\mathbf{z})u \\
 y &= z_1
 \end{aligned} \tag{3.26}$$

Si se define una señal de realimentación u (ley de control no lineal) de la forma siguiente:

$$u = \frac{1}{a(\mathbf{z})}[-b(\mathbf{z}) + v] \tag{3.27}$$

donde, v es la nueva entrada, entonces el nuevo conjunto de ecuaciones viene dado por:

$$\begin{aligned}
 \dot{z}_1 &= z_2 \\
 \dot{z}_2 &= z_3 \\
 &\vdots \\
 \dot{z}_n &= v \\
 y &= z_1
 \end{aligned} \tag{3.28}$$

compactando la ecuación (3.28) resulta en la siguiente representación lineal en el espacio estado.

$$\dot{\mathbf{z}} = A\mathbf{z} + Bv \tag{3.29}$$

$$y = C\mathbf{z} \tag{3.30}$$

donde,

$$A = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \\ 0 & 0 & 0 & \cdots & 0 \end{bmatrix}; \quad B = \begin{bmatrix} 0 \\ 0 \\ \vdots \\ 0 \\ 1 \end{bmatrix}; \quad C = \begin{bmatrix} 1 \\ 0 \\ \vdots \\ 0 \\ 0 \end{bmatrix}^T \quad (3.31)$$

Observar que la nueva entrada del proceso lineal es v . La Figura 3.1 muestra el proceso de linealización descrito hasta ahora. El proceso lineal de la ecuación (3.29) es completamente controlable y completamente observable, pues sus matrices de controlabilidad $[B \ AB \ \cdots \ A^{n-1}B]$ y de observabilidad $[C^T \ A^T C^T \ \cdots \ (A^T)^{n-1} C^T]$ poseen rango n (rango completo).

Por otro lado, el proceso resultante es un integrador de orden n debido a que su función de transferencia es:

$$\frac{y(s)}{v(s)} = H(s) = C(sI - A)^{-1}B = \frac{1}{s^n} \quad (3.32)$$

Respecto al proceso de linealización desarrollado, debemos remarcar lo siguiente:

- El sistema de lazo cerrado resultante (ver Figura 3.1) es lineal, completamente controlable y completamente observable. Por consiguiente, el problema de control de seguimiento de trayectorias queda resuelto diseñando v con cualquier técnica de control lineal conocida.
- En general, la realimentación no lineal requiere de todo el vector de estado \mathbf{z} , por lo contrario es necesario que todos sus elementos sean medibles. De lo contrario, estamos obligados a estimar empleando un observador no lineal.

[8]

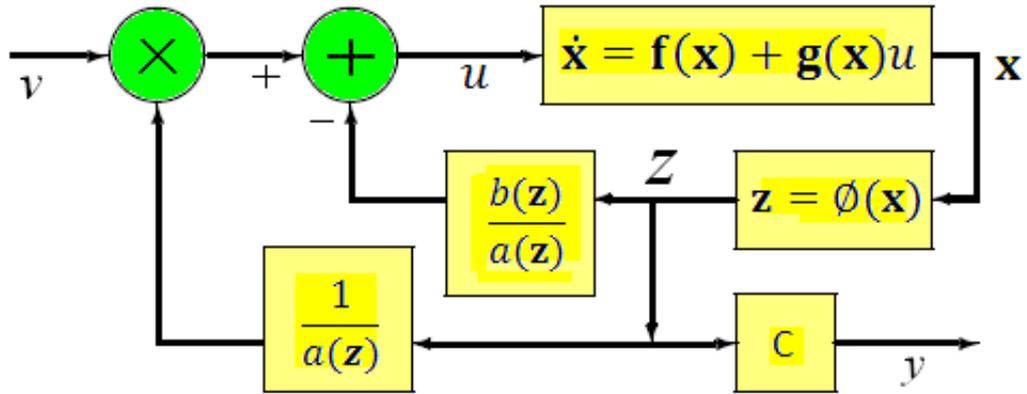


Figura 3.1: Diagrama de bloques para linealización exacta de un proceso no lineal

3.5.1. Condiciones para Linealización

La linealización desarrollada en la sección anterior fue posible debido a la existencia de una función de salida $h(\mathbf{x})$ respecto a la cual el proceso no lineal original tiene grado relativo $\rho=n$.

Finalmente, las condiciones para la linealización exacta de un proceso no lineal pueden ser formalizadas rigurosamente.

Teorema 3.5.1 *El proceso SISO no lineal,*

$$\begin{aligned}\dot{\mathbf{x}} &= \mathbf{f}(\mathbf{x}) + \mathbf{g}(\mathbf{x})u \\ y &= h(\mathbf{x})\end{aligned}\tag{3.33}$$

donde $\mathbf{f}(\mathbf{x})$ y $\mathbf{g}(\mathbf{x})$ son campos vectoriales suaves, es linealizable si y sólo si existe una región Ω en donde se cumplen las siguientes condiciones:

1. Los campos vectoriales $[\mathbf{g}, ad_{\mathbf{f}}\mathbf{g}, \dots, ad_{\mathbf{f}}^{n-1}\mathbf{g}]$ son linealmente independientes en la región Ω .
2. El conjunto $[\mathbf{g}, ad_{\mathbf{f}}\mathbf{g}, \dots, ad_{\mathbf{f}}^{n-2}\mathbf{g}]$ es involutivo en Ω .

La primera condición es una generalización de la condición de controlabilidad de sistemas lineales a sistemas no lineales, es decir, $[\mathbf{g}, ad_{\mathbf{f}}\mathbf{g}, \dots, ad_{\mathbf{f}}^{n-1}\mathbf{g}]$ aplicado a sistemas lineales se convierte en la conocida matriz de controlabilidad

$[B, AB, \dots, A^{n-1}B]$.

La condición involutiva es menos intuitiva. Para sistemas lineales, tal condición se satisface trivialmente ya que los campos vectoriales son constantes, pero en general, no se puede decir lo mismo especialmente cuando se trate de procesos no lineales.

3.6. Aplicación al Control de Velocidad de un Motor DC

Para la aplicación de la técnica de control no lineal por linealización exacta por realimentación de estados se considera un servomotor DC. Este proceso comprende un subsistema eléctrico compuesto por un generador de PWM (Pulse Width Modulation), que opera conjuntamente con un amplificador de potencia, cuyo voltaje amplificado se envía al servomotor controlado por voltaje de armadura. Además, presenta un subsistema mecánico que comprende un engranaje reductor y acoplado al eje de salida del reductor se encuentra acoplada una varilla, constituyendo en conjunto un brazo robótico de 1 grado de libertad (GDL).

El objetivo de control es diseñar e implementar un sistema de control no lineal basado en la técnica de linealización exacta por realimentación de estado de modo tal que la varilla se posicione en una referencia predeterminada o siga una trayectoria angular. Las condiciones de sobreimpulso y tiempo de establecimiento se plantearán más adelante.

La Figura 3.2 presenta la estructura del proceso no lineal a controlar.

Subsistema Mecánico La ecuación del subsistema mecánico se puede obtener aplicando la segunda ley de Newton en el punto donde se acopla el eje del servomotor con la varilla. El torque resultante nT que debe entregar el servomotor para vencer todos los torques que se le oponen es:

$$nT = nT_M + T_E + T_L + T_N \quad (3.34)$$

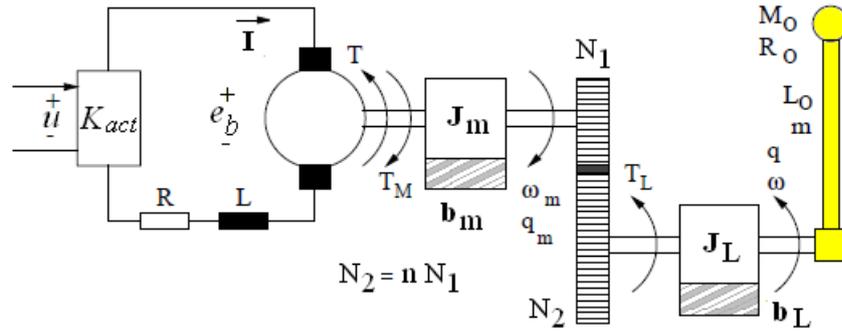


Figura 3.2: Brazo robótico de 1GDL

donde:

$$T_M = J_m \dot{\omega}_m + b_m \omega_m = n J_m \dot{\omega} + n b_m \omega \quad (3.35)$$

$$T_L = (M_o L_o^2 + \frac{2}{5} M_o R_o^2 + \frac{1}{3} m L_o^2 + J_L) \dot{\omega} + b_L \omega \quad (3.36)$$

$$T_E = g L_o (M_o + \frac{m}{2} \sin(q)) \quad (3.37)$$

Reemplazando las ecuaciones (3.35), (3.36) y (3.37) en la ecuación (3.34) se obtiene:

$$nT = M \dot{\omega} + B \omega + N \sin(q) + T_N \quad (3.38)$$

donde:

$$M = J_{eq} + M_o L_o^2 + \frac{2}{5} M_o R_o^2 + \frac{1}{3} m L_o^2 \quad (3.39)$$

$$N = g L_o (M_o + \frac{m}{2}) \quad (3.40)$$

$$J_{eq} = n^2 J_m + J_L \quad (3.41)$$

$$B = n^2 b_m + b_L \quad (3.42)$$

Subsistema Eléctrico La tensión de armadura V_a uq permite controlar la velocidad del motor viene expresada por:

$$V_a = IR + L \dot{I} + e_b \quad (3.43)$$

donde, e_b es la tensión contraelectromotriz y es igual a:

$$e_b = E\omega_m \quad (3.44)$$

y la tensión de armadura es:

$$V_a = uK_{act} \quad (3.45)$$

Energía Eléctrica a Energía Mecánica La ecuación que permite la conversión de la energía eléctrica a mecánica es:

$$T = KI \quad (3.46)$$

donde, K es la constante del par motriz. Operandi sobre las ecuaciones (3.38), (3.43), (3.45) y asumiendo que la inductancia es despreciable [8], obtenemos:

$$\dot{\omega} = - \left(\frac{n^2 KE}{RM} + \frac{B}{M} \right) \omega - \frac{N}{M} \sin(q) - \frac{T_N}{M} + \frac{nK K_{act}}{RM} u \quad (3.47)$$

3.6.1. Modelo del Proceso No lineal en Espacio-Estado

La ecuación (3.47) que describe el proceso no lineal de un motor puede ser representado en espacio estado, tal como:

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{F}(\mathbf{x}) + \mathbf{G}(\mathbf{x})u \\ y &= h(\mathbf{x}) \end{aligned} \quad (3.48)$$

Por tanto, eligiendo las variables de estado: $x_1 = q$ (posición angular), $x_2 = \dot{x}_1 = \dot{q} = \omega$ (velocidad angular), la ecuación (3.47) podemos reescribirla en espacio-estado, como:

$$\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= - \left(\frac{n^2 KE}{RM} + \frac{B}{M} \right) x_2 - \frac{N}{M} \sin(x_1) - \frac{T_N}{M} + \frac{nK K_{act}}{RM} u \end{aligned} \quad (3.49)$$

Comparando la expresión de arriba con la ecuación (3.48), se tiene:

$$\begin{aligned}
 \mathbf{F}(\mathbf{x}) &= \begin{bmatrix} x_2 \\ -\left(\frac{n^2KE}{RM} + \frac{B}{M}\right)x_2 - \frac{N}{M}\sin(x_1) - \frac{T_N}{M} \end{bmatrix} \\
 \mathbf{G}(\mathbf{x}) &= \begin{bmatrix} 0 \\ \frac{nK K_{act}}{RM} \end{bmatrix} \\
 h(\mathbf{x}) &= x_1
 \end{aligned} \tag{3.50}$$

3.6.2. Cálculo del Grado Relativo

De acuerdo a lo establecido en la sección 3.4.1, y empleando las derivadas de Lie en el modelo del proceso no lineal de segundo orden ($n=2$) hallamos el grado relativo, tal como sigue:

$$L_{\mathbf{F}}h(\mathbf{x}) = \nabla h(\mathbf{x})\mathbf{F}(\mathbf{x}) = x_2 \tag{3.51}$$

$$L_{\mathbf{G}}L_{\mathbf{F}}h(\mathbf{x}) = \nabla(L_{\mathbf{F}}h(\mathbf{x}))\mathbf{G} = \frac{nK_{act}K}{MR} \neq 0 \tag{3.52}$$

Por las ecuaciones de arriba concluimos que el grado relativo del proceso respecto a la salida $y = x_1$ es $r=2$. Esto implica que el proceso en cuestión es completamente linealizable ($r=n=2$).

3.6.3. Comprobación Estricta de la Linealización Exacta

Para aplicar las condiciones de linealización establecidas en el teorema 3.5.1, debemos hacer algunos cálculos previos, tal como:

$$ad_{\mathbf{F}}\mathbf{G} = [\mathbf{F}, \mathbf{G}] = \nabla\mathbf{G}\mathbf{F} - \nabla\mathbf{F}\mathbf{G} \tag{3.53}$$

Donde, $\nabla \mathbf{G} = 0$, pues \mathbf{G} es un vector constante (ver ecuación (3.50)), y

$$\begin{aligned} \nabla \mathbf{F} &= \begin{bmatrix} 0 & 1 \\ -\frac{N}{M} \cos(x_1) & -\left(\frac{n^2 KE}{MR} + \frac{B}{M}\right) \end{bmatrix} \\ \Rightarrow \nabla \mathbf{F} \mathbf{G} &= \begin{bmatrix} \frac{nK_{act}K}{MR} \\ -\frac{nK_{act}K}{MR} \left(\frac{n^2 KE}{MR} + \frac{B}{M}\right) \end{bmatrix} \end{aligned}$$

Con lo anterior la ecuación (3.53) viene a ser:

$$ad_{\mathbf{F}} \mathbf{G} = \begin{bmatrix} -\frac{nK_{act}K}{MR} \\ \frac{nK_{act}K}{MR} \left(\frac{n^2 KE}{MR} + \frac{B}{M}\right) \end{bmatrix} \quad (3.54)$$

Finalmente, se demuestra que:

$$\det[\mathbf{G} \quad ad_{\mathbf{F}} \mathbf{G}] = \left(\frac{nK_{act}K}{MR}\right)^2 \quad (3.55)$$

Esto significa que tal conjunto está formado por campos vectoriales linealmente independientes (1^{era} condición de linealización).

Por otra parte, el conjunto $[\mathbf{G} \quad ad_{\mathbf{F}} \mathbf{G}]$ es involutivo por que es constante. El proceso no lineal cumple con las condiciones dadas en el teorema 3.5.1, por lo tanto el proceso puede ser linealizado en forma exacta.

3.6.4. Forma Normal

La forma normal del proceso (ver sección 3.4.2) en estudio puede ser obtenida mediante la transformación no lineal $\mathbf{z} = \phi(\mathbf{x})$, como sigue:

$$\mathbf{z} = \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} = \phi(\mathbf{x}) = \begin{bmatrix} h(\mathbf{x}) \\ L_{\mathbf{F}} h(\mathbf{x}) \end{bmatrix} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \quad (3.56)$$

Por tanto, la forma normal del proceso resulta:

$$\dot{\mathbf{z}} = \begin{bmatrix} z_2 \\ -\left(\frac{n^2 KE}{RM} + \frac{B}{M}\right) z_2 - \frac{N}{M} \sin(z_1) \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{nK K_{act}}{RM} \end{bmatrix} u \quad (3.57)$$

No se considera el término T_N , pues será compensado por software.

3.6.5. Ley de Control

La ley de control siguiente:

$$u = \frac{MR}{nK_{act}K} \left[v + \left(\frac{n^2 KE}{RM} + \frac{B}{M} \right) z_2 + \frac{N}{M} \sin(z_1) \right] \quad (3.58)$$

linealiza completamente el modelo del proceso dado en la ecuación (3.57), donde v es la nueva variable de entrada por diseñarse. Cuando la señal de control es reemplazado en la ecuación (3.57), se obtiene:

$$\begin{aligned} \dot{z}_1 &= \dot{z}_2 \\ \dot{z}_2 &= v \\ y &= z_1 \end{aligned} \quad (3.59)$$

La ecuación (3.59) representa un doble integrador, es decir, $\frac{d^2 y}{dt^2} = v$. La señal v debe diseñarse de modo tal que $y = z_1$ siga una trayectoria angular de referencia $r(t)$.

La siguiente señal v :

$$v = \frac{d^2 r}{dt^2} + K_1 \frac{d(r - y)}{dt} + K_2 (r - y) \quad (3.60)$$

Permite que el sistema en lazo cerrado tenga la forma:

$$\frac{d^2 (r - y)}{dt^2} + K_1 \frac{d(r - y)}{dt} + K_2 (r - y) = 0 \quad (3.61)$$

En la ecuación (3.61) se observa que los coeficientes K_1 y K_2 determinan la dinámica del error ($r-y$). Por tanto, si se desea que $y \rightarrow r$ cuando $t \rightarrow \infty$ los eigenvalores de la ecuación (3.61) deben ubicarse en el semiplano izquierdo. Por ejemplo, si deseamos que los eigenvalores sean $s_1 = s_2 = -4$, entonces la ecuación del sistema de lazo cerrado resulta $s^2 + 8s + 16$, de donde se obtiene que $K_1 = 8$ y $K_2 = 16$.

La expresión de v podemos reescribirla como:

$$v = \frac{d^2r}{dt^2} + K_1 \frac{dr}{dt} + K_2 r - \mathbf{Kz} \quad (3.62)$$

Donde, $\mathbf{K} = [K_1 \ K_2]$ y $\mathbf{z} = [z_1 \ z_2]^T$

Finalmente, la señal de control u puede reescribirse como:

$$u = \frac{MR}{nK_{act}K} \left[\frac{d^2r}{dt^2} + K_1 \frac{dr}{dt} + K_2 r - \mathbf{Kz} + \left(\frac{n^2 KE}{RM} + \frac{B}{M} \right) z_2 + \frac{N}{M} \sin(z_1) \right] \quad (3.63)$$

3.6.6. Simulación del Controlador para el Proceso No Lineal

El algoritmo de control se ha programado MATLAB. Las condiciones para la simulación son indicadas más adelante. La Figura 3.3 muestra el diagrama de bloques del sistema de control no lineal que vamos a simular.

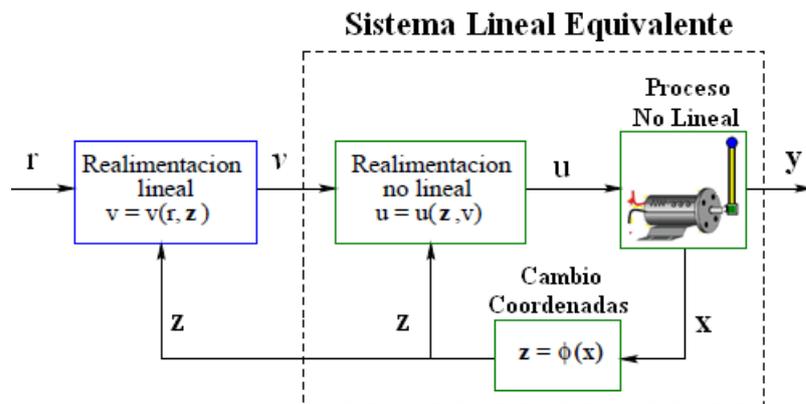


Figura 3.3: Diagrama de bloques del sistema de control para motor DC

Para Trayectoria Constante Para las simulaciones se ha considerado que todos los estados iniciales parten del reposo. Se han considerado que los polos del proceso linealizado resultante sean $s_1 = -4$ y $s_2 = -4$, de esto se obtiene que $K_1 = 8$ y $K_2 = 16$.

El tiempo de muestreo es de 10 ms, y el tiempo total de simulación es de 12 s. Además, es importante mencionar que en el programa se ha incluido la compensación de la fricción de Coulomb y se ha limitado la señal de control para evitar que el amplificador se sature.

En la Figura 3.4, se muestra la salida controlada para la posición angular del brazo de 1 grado de libertad (GDL) para una referencia de 90° . El tiempo de establecimiento es de 2 seg. y corresponde a un sistema sobreamortiguado. En la

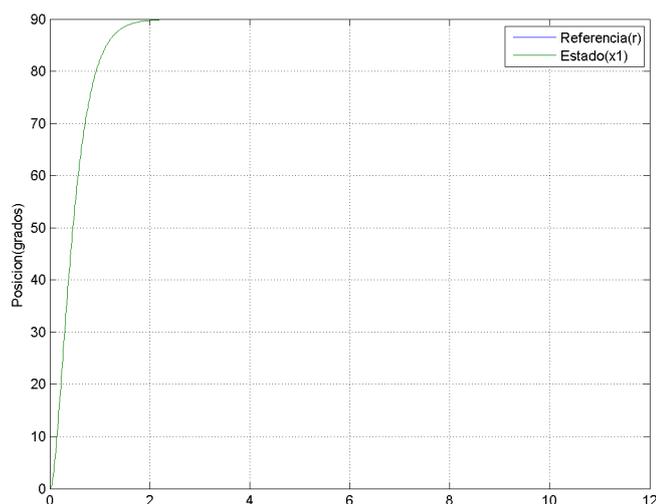


Figura 3.4: Salida controlada para la posición del brazo de 1GDL con CLERE

Figura 3.5 se muestra la señal de control enviado al motor, permitiéndole seguir la trayectoria constante tal como se muestra en Figura 3.4. Se aprecia que la señal de control alcanza un valor pico de 0.46 voltios, pasados los 2 seg. se establece a un voltaje de 0.4 voltios. La Figura 3.6 se muestra la respuesta del sistema para la velocidad angular del brazo acoplado al eje del motor DC. Se aprecia que en el arranque la velocidad crece hasta un valor pico de 2.25 rad./s para luego aproximarse a cero a los 3 seg. El aproximarse la velocidad a cero nos da a entender

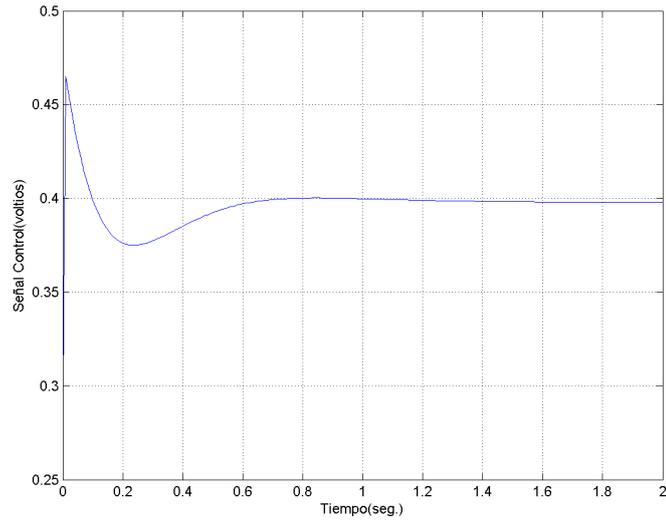


Figura 3.5: Señal de control para el motor DC

que el proceso ha sido controlado.

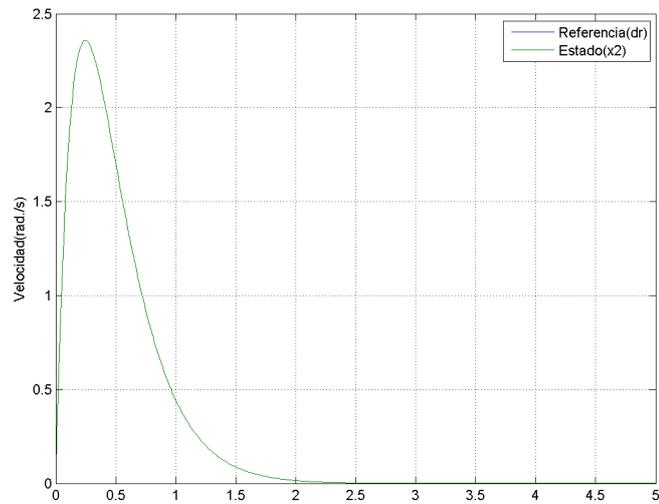


Figura 3.6: Velocidad angular del brazo de 1GDL acoplado al motor DC

3.7. Linealización por Realimentación Exacta: Caso MIMO

La estrategia de control por linealización exacta por realimentación de estado para sistemas SISO (Single Input Single Output) puede fácilmente extenderse al control de sistemas MIMO (Multiple Input Multiple Output).

En esta sección se dan los procedimientos necesarios para usar esta estrategia al control de sistemas MIMO.

3.7.1. Modelando Sistemas MIMO Cuadrados

Un sistema MIMO no lineal se denomina cuadrado cuando el número de sus entradas iguala al número de sus salidas. El sistema cuadrado en la vecindad del punto \mathbf{x}_0 , se describe por:

$$\begin{aligned}\dot{\mathbf{x}} &= \mathbf{F}(\mathbf{x}) + \mathbf{G}(\mathbf{x})\mathbf{u} = \mathbf{F}(\mathbf{x}) + \sum_{j=1}^m g_j(\mathbf{x})u_j(\mathbf{x}) \\ \mathbf{y} &= \mathbf{h}(\mathbf{x})\end{aligned}\tag{3.64}$$

Donde:

$$\begin{aligned}\mathbf{x} &= \begin{bmatrix} x_1 & x_2 & \cdots & x_n \end{bmatrix}^T \\ \mathbf{u} &= \begin{bmatrix} u_1 & u_2 & \cdots & u_m \end{bmatrix}^T \\ \mathbf{y} &= \begin{bmatrix} y_1 & y_2 & \cdots & y_m \end{bmatrix}^T \\ \mathbf{h}(\mathbf{x}) &= \begin{bmatrix} h_1(\mathbf{x}) & h_2(\mathbf{x}) & \cdots & h_m(\mathbf{x}) \end{bmatrix}^T \\ \mathbf{F}(\mathbf{x}) &= \begin{bmatrix} f_1(\mathbf{x}) & f_2(\mathbf{x}) & \cdots & f_n(\mathbf{x}) \end{bmatrix}^T \\ \mathbf{G}(\mathbf{x}) &= \begin{bmatrix} G_{11}(\mathbf{x}) & G_{12}(\mathbf{x}) & \cdots & G_{1m}(\mathbf{x}) \\ \vdots & \vdots & \ddots & \vdots \\ G_{m1}(\mathbf{x}) & G_{m2}(\mathbf{x}) & \cdots & G_{mm}(\mathbf{x}) \end{bmatrix} \\ &= [g_1(\mathbf{x}) \mid \cdots \mid g_m(\mathbf{x})]\end{aligned}$$

Además, \mathbf{x} es el vector de estado de orden $n \times 1$, \mathbf{u} es el vector de control (la entrada) de orden $m \times 1$, \mathbf{y} es el vector de salida 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 G es una matriz de dimensión $n \times m$ cuyas columnas son campos vectoriales g_j de orden $n \times 1$.

3.7.2. Grado Relativo Total

En la referencia [9], se establece que el sistema cuadrado descrito por la ecuación (3.64) posee un *grado relativo total* $\rho = \rho_1 + \dots + \rho_m$, si:

1. Para todo \mathbf{x} en una vecindad de \mathbf{x}_0

$$L_{g_j} L_{\mathbf{F}}^k h_i(\mathbf{x}) = 0 \quad j, i = 1, \dots, m \quad , k < \rho_i - 1 \quad (3.65)$$

2. La siguiente matriz de orden $m \times m$ es no singular en $x = x_0$:

$$\mathcal{A}(\mathbf{x}) = \begin{bmatrix} L_{g_1} L_{\mathbf{F}}^{\rho_1-1} h_1(\mathbf{x}) & \cdots & L_{g_m} L_{\mathbf{F}}^{\rho_1-1} h_1(\mathbf{x}) \\ L_{g_1} L_{\mathbf{F}}^{\rho_2-1} h_2(\mathbf{x}) & \cdots & L_{g_m} L_{\mathbf{F}}^{\rho_2-1} h_2(\mathbf{x}) \\ \vdots & \ddots & \vdots \\ L_{g_1} L_{\mathbf{F}}^{\rho_m-1} h_m(\mathbf{x}) & \cdots & L_{g_m} L_{\mathbf{F}}^{\rho_m-1} h_m(\mathbf{x}) \end{bmatrix} \quad (3.66)$$

de las filas de \mathcal{A} observamos que cada ρ_i está relacionado con la i -ésima salida del sistema. También debemos notar que ρ_i es el número de diferenciaciones realizadas sobre la salida y_i requeridas para que aparezca al menos uno de los componentes del vector de entrada.

3.7.3. Forma Normal MIMO para Linealización Exacta

Si el grado relativo total del sistema MIMO cuadrado descrito en la ecuación (3.64) es ρ , entonces:

$$\rho = \rho_1 + \rho_2 + \cdots + \rho_m \leq n \quad (3.67)$$

Para el sistema MIMO descrito en la ecuación (3.64) se dice que posee linealización exacta, si su grado relativo total ρ es igual al orden n del sistema.

Para la linealización exacta se examina la siguiente transformación de estados:

$$z = \Phi(x) = \begin{bmatrix} \phi_1(x) \\ \vdots \\ \phi_{\rho_1}(x) \\ \psi_1(x) \\ \vdots \\ \psi_{\rho_2}(x) \\ \vdots \\ \xi_1(x) \\ \vdots \\ \xi_{\rho_m}(x) \end{bmatrix} = \begin{bmatrix} z_1 \\ \vdots \\ z_{\rho_1} \\ z_{\rho_1+1} \\ \vdots \\ z_{\rho_1+\rho_2} \\ \vdots \\ z_{\rho_1+\rho_2+\cdots+1} \\ \vdots \\ z_{\rho_1+\rho_2+\cdots+\rho_m} \end{bmatrix} \quad (3.68)$$

donde,

$$\begin{aligned} \phi_1(\mathbf{x}) &= h_1(\mathbf{x}) = y_1(\mathbf{x}) & \psi_1(\mathbf{x}) &= h_2(\mathbf{x}) = y_2(\mathbf{x}) \\ \phi_2(\mathbf{x}) &= L_f h_1(\mathbf{x}) & \psi_2(\mathbf{x}) &= L_f h_2(\mathbf{x}) & ; & \cdots \\ \vdots &= \vdots & \vdots &= \vdots & & \\ \phi_{\rho_1}(\mathbf{x}) &= L_f^{\rho_1-1} h_1(\mathbf{x}) & \psi_{\rho_2}(\mathbf{x}) &= L_f^{\rho_2-1} h_2(\mathbf{x}) \end{aligned}$$

$$\begin{aligned} \xi_1(\mathbf{x}) &= h_m(\mathbf{x}) = y_m(\mathbf{x}) \\ \xi_2(\mathbf{x}) &= L_f h_m(\mathbf{x}) \\ \vdots &= \vdots \\ \xi_{\rho_m}(\mathbf{x}) &= L_f^{\rho_m-1} h_m(\mathbf{x}) \end{aligned} \quad (3.69)$$

lo anterior nos conduce a la siguiente descripción en el espacio de estado:

$$\begin{aligned}
y_1^{[1]} &= \phi_1^{[1]}(\mathbf{x}) = \phi_2(\mathbf{x}) \\
&\vdots \\
y_1^{[\rho_1-1]} &= \phi_{\rho_1-1}^{[1]}(\mathbf{x}) = \phi_{\rho_1}(\mathbf{x}) \\
y_1^{[\rho_1]} &= \phi_{\rho_1}^{[1]}(\mathbf{x}) = L_{\mathbf{F}}^{\rho_1} h_1(\Phi^{-1}(\mathbf{z})) + \sum_{j=1}^m L_{g_j} L_{\mathbf{F}}^{\rho_1-1} h_1(\Phi^{-1}(\mathbf{z})) u_j \\
y_2^{[1]} &= \psi_1^{[1]}(\mathbf{x}) = \psi_2(\mathbf{x}) \\
&\vdots \\
y_2^{[\rho_2-1]} &= \psi_{\rho_2-1}^{[1]}(\mathbf{x}) = \psi_{\rho_2}(\mathbf{x}) \\
y_2^{[\rho_2]} &= \phi_{\rho_2}^{[1]}(\mathbf{x}) = L_{\mathbf{F}}^{\rho_2} h_2(\Phi^{-1}(\mathbf{z})) + \sum_{j=1}^m L_{g_j} L_{\mathbf{F}}^{\rho_2-1} h_2(\Phi^{-1}(\mathbf{z})) u_j \\
&\vdots \\
y_m^{[1]} &= \xi_1^{[1]}(\mathbf{x}) = \xi_2(\mathbf{x}) \\
&\vdots \\
y_m^{[\rho_m-1]} &= \xi_{\rho_m-1}^{[1]}(\mathbf{x}) = \xi_{\rho_m}(\mathbf{x}) \\
y_m^{[\rho_m]} &= \xi_{\rho_m}^{[1]}(\mathbf{x}) = L_{\mathbf{F}}^{\rho_m} h_m(\Phi^{-1}(\mathbf{z})) + \sum_{j=1}^m L_{g_j} L_{\mathbf{F}}^{\rho_m-1} h_m(\Phi^{-1}(\mathbf{z})) u_j
\end{aligned} \tag{3.70}$$

3.7.4. La Ley de Control MIMO Desacoplada

Es importante anotar que el grado relativo $\rho_i, i = 1, \dots, m$, es el entero más pequeño requerido para que al menos una de las entradas aparezca en la salida, tal como se muestra a continuación:

$$y_i^{[\rho_i]} = L_{\mathbf{F}}^{\rho_i} h_i(\Phi^{-1}(\mathbf{z})) + \sum_{j=1}^m L_{g_j} L_{\mathbf{F}}^{\rho_i-1} h_i(\Phi^{-1}(\mathbf{z})) u_j \tag{3.71}$$

Por lo tanto se tiene que, $L_{\mathbf{F}}^{\rho_i-1} h_i(\Phi^{-1}(\mathbf{z})) u_j \neq 0$ para al menos un j en la vecindad de \mathbf{x}_0 .

Ahora consideremos la siguiente *ley de control desacoplada*:

$$\mathbf{u} = \mathcal{A}(\mathbf{z})^{-1}(\mathbf{z})[-\mathcal{B}(\mathbf{z}) + \mathbf{v}] \tag{3.72}$$

donde $\mathbf{v} = [v_1 \ \dots \ v_m]^T$ es la nueva entrada por diseñar, y $\mathcal{A}(z)$ y $\mathcal{B}(z)$ son matrices con elementos:

$$a_{ij} = L_{g_i} L_{\mathbf{F}}^{\rho_j - 1} h_j(\Phi^{-1}(\mathbf{z})), \quad i, j = 1, \dots, m \quad (3.73)$$

$$b_i = L_{\mathbf{F}}^{\rho_i} h_i(\Phi^{-1}(\mathbf{z})), \quad i = 1, \dots, m \quad (3.74)$$

Sustituyendo los elementos $u_i, i = 1, \dots, m$ de la ecuación (3.72) en (3.70) nos conduce a la representación canónica invariante con el tiempo del sistema MIMO.

Esto es:

$$y_i^{[\rho_i]} = v_i,$$

de esta relación se tiene que, $y(s) = H(s)v(s)$ fácilmente se puede demostrar que la matriz de transferencia $H(s)$ del sistema MIMO linealizado viene dado por:

$$H(s) = \begin{bmatrix} 1/s^{\rho_1} & 0 & \dots & 0 \\ 0 & 1/s^{\rho_2} & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & 1/s^{\rho_m} \end{bmatrix}$$

lo cual significa que $y_i^{(\rho_i)} = d^{\rho_i} y_i / dt^{\rho_i} = v_i, i = 1, \dots, m$, donde cada entrada v_i se diseña para ubicar los polos del correspondiente subsistema lineal equivalente.

Este sistema lineal se puede seleccionar como:

$$v_i = \frac{d^{\rho_i} y_i}{dt^{\rho_i}} = y_i^{[\rho_i]} = \omega_i^{\rho_i} + K_{i,1} \bar{y}_i^{\rho_i - 1} + \dots + K_{i,\rho_i - 1} \bar{y}_i^{[1]} + K_{i,\rho_i} \bar{y}_i \quad (3.75)$$

donde $\omega_i(t)$ es la i -ésima trayectoria deseada y, $\bar{y}_i(t) = \omega_i(t) - y_i(t)$ es la i -ésima señal de error de seguimiento.

La i -ésima ecuación característica del sistema resulta:

$$\bar{y}_i^{[\rho_i]} + K_{i,1} \bar{y}_i^{[\rho_i - 1]} + \dots + K_{i,\rho_i - 1} \bar{y}_i^{[1]} + K_{i,\rho_i} \bar{y}_i = 0 \quad (3.76)$$

Para un comportamiento asintótico estable, es decir, cuando $\bar{y}_i \rightarrow 0$ o cuando $y_i \rightarrow \omega_i$ para $t \rightarrow \infty$, ninguna de las raíces de la ecuación (3.75) debe de estar localizada en el semiplano derecho del plano-s, incluyendo el eje imaginario.

CAPÍTULO 4

Control Predictivo Basados en Modelos

El Control Predictivo Basado en Modelos (CPBM) es una tecnología que fue desarrollada inicialmente para las plantas de energía e industria petrolera, luego poco a poco fue ganando terreno en casi todas las industrias químicas, alimentarias, automovilísticas, aeroespacial y metalúrgica [10].

El éxito del control predictivo se puede atribuir a tres razones principales:

1. La primera es la incorporación de los modelos en el algoritmo de control, lo que permite al controlador operar directamente con las características dinámicas del proceso a controlar.
2. La segunda razón es la *visión* del control predictivo hacia el futuro, característica que le permite anticipar y eliminar todos los efectos de las perturbaciones.
3. Y la última razón es la consideración de restricciones en las entradas y en las salidas para el cálculo de la señal de control óptima.

El control predictivo usa el modelo del proceso para predecir su comportamiento futuro.

Las variables controladas predichas son usadas en un proceso de optimización con el objetivo de minimizar una función de coste y así calcular la secuencia de control futura que permite a la salida del sistema seguir una trayectoria de referencia.

Generalmente la función de coste depende del error cuadrático entre la trayectoria

de referencia futura y la variable controlada predicha hasta un tiempo t , llamado horizonte de predicción [2], [3].

4.1. Introducción al Control Predictivo Basado en Modelo

El **Control Predictivo Basado en Modelo** fue introducido por Richalet en 1978 con el algoritmo *Model Predictive Heuristic Control* y por otra parte independientemente Cutler y Ramaker, dos ingenieros de la compañía Shell, presentaron su controlador *Dynamic Matrix Control (DMC)* el cual controlaba de manera satisfactoria varios procesos industriales.

Estos dos algoritmos usan el modelo de la planta para predecir la evolución de las variables controladas sobre un horizonte de predicción. El cálculo de los valores futuros de las variables manipuladas (señales de control) se consiguen minimizando una función de coste que depende del error de predicción (diferencia entre la trayectoria de referencia futura y la variable controlada predicha). El proceso de optimización se repite en cada instante de muestreo actualizando las informaciones del proceso.

Posteriormente fue introducido el Generalized Predictive Controller (GPC) por Clarke, Mohtadi y Tuffs (1987). Este algoritmo surgido en un entorno académico fue probado con éxito en varias aplicaciones industriales y ha conseguido un nivel de aceptación muy importante en la industria.

Hoy en día, después de muchos cambios en la industria, varios paquetes de control predictivo han penetrado en el mercado industrial y están comercialmente disponibles, ente éstos tenemos:

- AspenTech: Dynamic Matrix Control (DMC)
- Adersa: Identification and Command (IDCOM), Hierarchical Constraint Control (HIECON), and Predictive Functional Control (PFC).
- Honeywell Profimatics: Robust Model Predictive Control Technology (RM-PCT) and Predictive Control Technology (PCT).

4.2. Estrategia del Control Predictivo Basado en Modelos

Considerando un proceso en el cual sus entradas, salidas y ley de control evolucionan tal como se muestra en la Figura 4.1, hasta un tiempo $k=k$.

La estrategia del control predictivo seguirá los siguientes pasos:

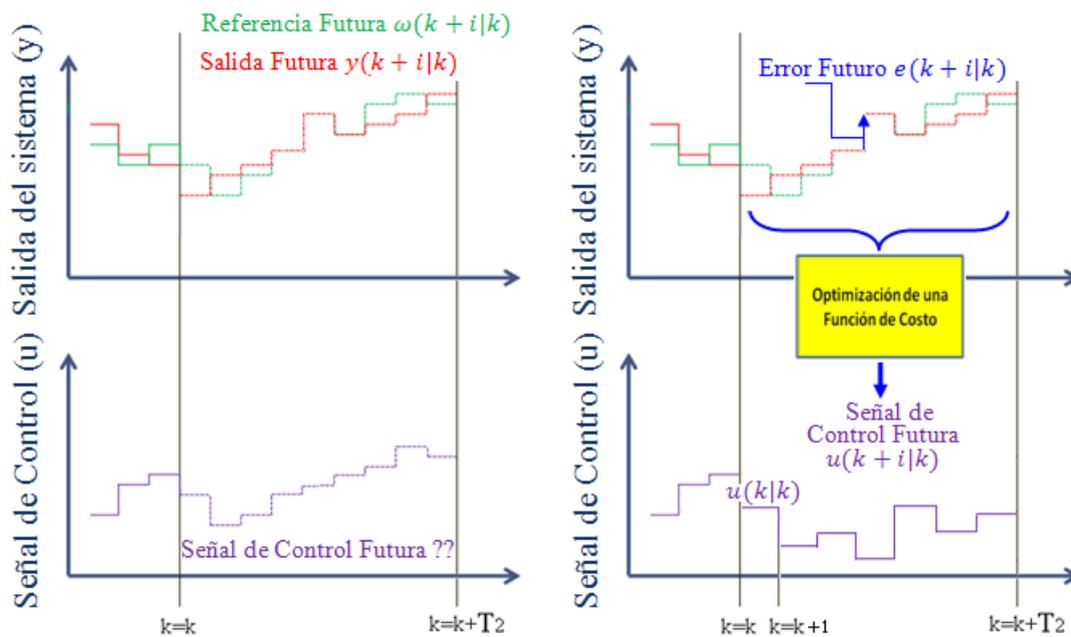


Figura 4.1: Estrategia del CPBM

- La salida futura medida en el instante k , dado por $y(k+i|k)$ $i=1, \dots, T_2$; es el resultado de usar el modelo del proceso, teniendo por información de entrada: las entradas, salidas del proceso, ambos hasta el instante $k=k$, y de una ley de control incógnita, la cual será hallada luego de un proceso de optimización. El tiempo hasta el cual la salida es predicha es llamado horizonte de predicción T_2 .
- La trayectoria de referencia futura medida en el instante k , dado por $\omega(k+i|k)$ $i=1, \dots, T_2$, nos indica cómo debemos llegar al objetivo desde el estado actual.
- La señal de control óptima $u(k+i|k)$ se calcula optimizando una función de coste para acercar al máximo posible la variable controlada a la trayectoria

de referencia. Esta función de coste es generalmente una función cuadrática del error entre la salida futura y referencia futura.

- Usa la estrategia del horizonte móvil, es decir, en cada instante de muestreo se aplica solamente la primera señal de control $u(k|k)$ de toda la secuencia de control calculada, $u(k+i|k)$. Este proceso se repite en cada instante de muestreo.

4.3. Aspectos de Implementación del Control Predictivo

El control predictivo se desarrolla siguiendo los siguientes pasos:

- Modelado y Cálculo de Predicción de la Salida Controlada.
- Optimización de la función de coste.

A continuación se describen con más detalle estos puntos.

4.4. Modelado y Predicción de la Salida Controlada

El pilar del control predictivo es el modelo, el cual debe ser bien diseñado para capturar al máximo la dinámica del proceso y debe ser capaz de permitir el cálculo de las predicciones.

El uso del modelo del proceso viene determinado por la necesidad del cálculo de la salida predicha en instantes futuros $y(k+i|k)$. Las diferentes estrategias de control predictivo basado en modelo (CPBM) usan distintos modelos para representar la relación de las salidas con las entradas. En las secciones siguientes se dará un resumen de las técnicas más importantes para modelar un sistema.

4.4.1. Modelado Lineal

Casi todas las formas posibles de modelar un proceso lineal aparecen en alguna formulación del control predictivo, siendo las más usadas las siguientes:

- Respuesta Impulsional. También conocida como modelo de convolución. La salida viene relacionada con la entrada por la siguiente expresión:

$$y(t) = \sum_{i=1}^{\infty} h_i u(t - i) \quad (4.1)$$

donde, h_i son los valores muestreados obtenidos al someter al proceso a un impulso unitario de amplitud igual al período de muestreo, ver figura 4.2(a).

Esta suma es truncada y sólo se consideran los N valores, es decir:

$$y(t) = \sum_{i=1}^N h_i u(t - i)$$

un inconveniente de este método es la gran cantidad de parámetros que necesita, ya que N suele ser un valor elevado para que describa fielmente el comportamiento del proceso. La predicción usando este modelo vendrá dado por:

$$\hat{y}(t + k | t) = \sum_{i=1}^N h_i u(t + k - i | t)$$

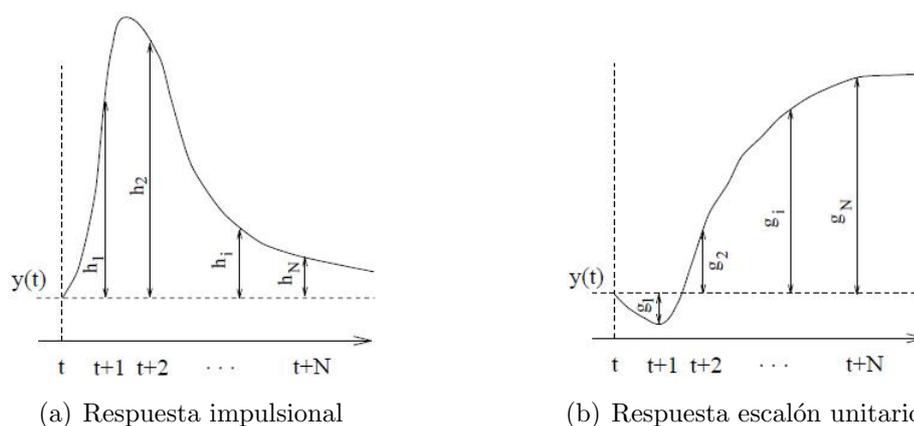


Figura 4.2: Modelado y predicción

- Respuesta al Escalón. Usado por algoritmos predictivos como control predictivo de matriz dinámica (DMC) y sus variantes. Este es muy similar a la respuesta impulsional excepto que la señal de entrada es un escalón

unitario. Para sistemas estables, la respuesta truncada viene dada por:

$$y(t) = y_o + \sum_{i=1}^N g_i \Delta u(t - i)$$

donde, g_i son los valores muestreados obtenidos al someter al proceso a un escalón unitario la figura 4.2(b) y $\Delta u(t - i) = u(t) - u(t - 1)$.

- Modelo CARIMA (Controller Auto-Regressive and Moving Average). Es un modelo muy usado y viene dado por:

$$A(q^{-1})y(t) = \left(B(q^{-1})u(t) + D(q^{-1})v(t) + \frac{T(q^{-1})}{\Delta} \right) \zeta(t) \quad (4.2)$$

donde A,B,D,T son polinomios en q^{-1} , Δ es el operador diferencial $1 - q^{-1}$, $y(t)$ es la salida del proceso, $u(t)$ es la entrada del sistema, $v(t)$ son las perturbaciones medibles y ζ ruido blanco.

Las predicciones se pueden calcular a partir del modelo de la ecuación (4.2) en el instante $t+j$.

$$Ay(t + j) = Bu(t + j) + Dv(t + j) + \frac{T}{\Delta} \zeta(t + j) \quad (4.3)$$

Para el caso de sistemas lineales, la respuesta se puede dividir en dos partes diferentes: respuesta forzada y respuesta libre. La respuesta libre representa el comportamiento predicho de la salida $y(t+j)$, considerando una acción de control constante. La respuesta forzada o controlada representa la parte de la salida que corresponde a las acciones de control futuras. El principio de superposición se puede usar en el caso de sistemas lineales, por tanto la salida predicha viene dado por:

$$y(t + j) = y_{libre}(t + j) + y_{forzada}(t + j)$$

La respuesta libre se puede calcular suponiendo que los incrementos de control son nulos $\Delta u(t) = 0$ para $j \geq 0$.

4.4.2. Modelado No Lineal

El modelado no lineal es la base para el control predictivo no lineal, este modelado contiene más información del comportamiento del proceso.

En esta sección se presenta un resumen de los diferentes métodos de modelado no lineal.

◇ Modelos en Ecuaciones Diferenciales

Estos modelos se derivan como resultado de aplicar principios físicos como de balances másicos y energéticos del proceso. Estos modelos se caracterizan por su eficacia para predecir el comportamiento del proceso, debido a que están basadas en leyes físicas, vienen descritos por ecuaciones diferenciales no lineales.

$$\begin{aligned}\dot{x}(t) &= F(x(t)) + G(x(t))u(t) \\ y(t) &= h(x(t))\end{aligned}\tag{4.4}$$

Donde $x(t)$ es el vector de estados, u es el vector de entradas, y es el vector de salidas.

La obtención de un buen modelo depende de la complejidad que presenta el proceso a controlar y el conocimiento que disponemos a la hora de modelar. Los modelos obtenidos con este método son generalmente complejos basados en ecuaciones diferenciales que en muchos casos no tienen solución analítica. Por esta razón se limita usualmente a la simulación, al estudio y diseño de plantas y no mucho al control por que conlleva a un gran esfuerzo computacional.

Sin embargo, existen algoritmos que si usan estos modelos para el control, puesto que aprovechan su gran capacidad predictiva. Existen otros métodos de modelización que usan el método de descomposición del modelo no lineal en una parte dinámica lineal y otra constituida por un polinomio no lineal estático. Dentro de esta familia podemos citar a los siguientes.

- Representaciones de Volterra.
- Modelos Hammerstein.
- Modelos Weiner.

Para mayores detalles de estos métodos de modelado se puede revisar [11].

4.5. Función de Coste y Optimización

La señal de control óptima se calcula minimizando una función de coste a lo largo de un horizonte de predicción. La función de coste es de índice cuadrático. El objetivo es minimizar el error entre la salida predicha $y(k+i|k)$ y la trayectoria de referencia futura $\omega(k+i|k)$ hasta un horizonte de predicción.

La función de costo definida en tiempo continuo es tal como se muestra a continuación:

$$J = \frac{1}{2} \int_0^{T_2} (y(t + \tau|t) - \omega(t + \tau|t))^T (y(t + d\tau|t) - \omega(t + \tau|t)) \tau \quad (4.5)$$

Donde, T_2 es el horizonte de predicción.

4.6. Control Predictivo No Lineal

Los modelos lineales describen muy bien el comportamiento del proceso pero solamente en las cercanías de su punto de operación por este hecho resultan inadecuados para describir la dinámica completa de un proceso no lineal que se desea controlar, debido a que actualmente las industrias se enfrentan a un mercado cambiante lo que les obliga a operar sus procesos productivos bajo diversos puntos de operación.

Ante ello el control predictivo no lineal (CPNL) es una buena alternativa para tratar procesos que operan en diversos puntos de operación y que presentan alto grado de no linealidad.

El CPNL resulta como extensión de las ideas propuestas del control predictivo basado en modelo para sistemas lineales, a sistemas no lineales. Sin embargo, este hecho nos conduce a un inconveniente, es que al usar un modelo no lineal para predecir las salidas convierte al problema optimización en un problema de optimización no lineal, y para solucionar este problema se requiere de algoritmos numéricos complejos de elevada carga computacional. Por este motivo estas técnicas son aplicadas generalmente al control de sistemas de dinámica lenta (procesos químicos, petroleros).

Para evadir este problema muchos algoritmos predictivos no lineal han sido desarrollados (Ping, 1995; Singh, 1995; Sourouth, 1996) basados en obtener el error de predicción con horizonte de predicción finito, por medio de la expansión serie de Taylor entre la salida del proceso y de la trayectoria de referencia futura. Luego, la minimización del error de predicción permite derivar una secuencia de señales de control de las que sólo la primera se envía al proceso.

Cabe resaltar que estos algoritmos de control predictivo no lineal no requieren de una optimización on-line, es decir en cada instante de muestreo. En consecuencia, estos algoritmos pueden ser aplicados a sistemas no lineales de dinámica rápida (robots, motores AC, etc.).

En esta tesis examinaremos el control predictivo generalizado no lineal en tiempo continuo, este enfoque está basado en la minimización de una función de costo que depende del cuadrado del error de predicción para esta tarea usamos el criterio de la primera derivada obteniendo una ley de control off-line, es decir que no requiere hallar las secuencias de control óptimas en cada instante de muestreo.

4.7. Control Predictivo Generalizado No Lineal en Tiempo Continuo para Sistemas MIMO

El presente artículo propone una estrategia de control eficiente llamado Control Predictivo Generalizado No Lineal (CPGNL) para sistemas no lineales de dinámica rápida, el cual fue desarrollado en tiempo continuo. La característica

principal del CPGNL es que no se requiere resolver el problema de optimización en cada instante de muestreo.

Consideremos el sistema no lineal,

$$\dot{x}(t) = \mathbf{F}(x) + \mathbf{G}(x)\mathbf{u}(t) \quad (4.6)$$

$$y(t) = \mathbf{h}(x), \quad (4.7)$$

donde, $x(t) \in R^n$ es el vector de estado, $\mathbf{u}(t) \in R^m$ representa el vector de control y $y = [y_1, y_2, \dots, y_m]^T \in R^m$ es la salida.

Las funciones $\mathbf{F}(x) : R^n \rightarrow R^n$, $\mathbf{G}(x) : R^n \rightarrow R^{n \times m}$ y $\mathbf{h}(x) : R^n \rightarrow R^m$ son suficientemente suaves.

La trayectoria de referencia futura viene representada por una función suave $\hat{\omega}$.

El problema consiste en diseñar una ley de control $\mathbf{u}(t)$, que permita mejorar el seguimiento de la salida controlada predicha $\hat{y}(t)$ a la trayectoria de referencia futura $\hat{\omega}(t)$ sobre un horizonte de predicción. Se define el error de predicción como $e(t + \tau|t) = \hat{y}(t + \tau|t) - \hat{\omega}(t + \tau|t)$.

4.7.1. Predicción de la Salida

La salida del sistema no lineal para un marco de predicción $\tau \in [0, T_2]$, donde T_2 es el horizonte de predicción, es aproximada por la serie de Taylor.

$$\hat{y}(t + \tau|t) \approx \hat{y}^{[0]}(t) + \tau \hat{y}^{[1]}(t) + \frac{\tau^2}{2!} \hat{y}^{[2]}(t) + \dots + \frac{\tau^\rho}{\rho!} \hat{y}^{[\rho]}(t) + \dots + \frac{\tau^{(r+\rho)}}{(r + \rho)!} \hat{y}^{[r+\rho]}(t) \quad (4.8)$$

Donde, ρ es grado relativo del sistema y r es cualquier número entero y positivo.

Las derivadas temporales $\hat{y}^{[i]}(t)$ de la salida que son requeridas en la ecuación

(4.8), vienen dadas por:

$$\hat{y}^{[0]}(t) = L_{\mathbf{F}}^0 \mathbf{h}(x) = \mathbf{h}(x) \quad (4.9)$$

$$\hat{y}^{[1]}(t) = L_{\mathbf{F}}^1 \mathbf{h}(x) \quad (4.10)$$

$$\hat{y}^{[2]}(t) = L_{\mathbf{F}}^2 \mathbf{h}(x) \quad (4.11)$$

$$\vdots = \vdots \quad (4.12)$$

$$\hat{y}^{[\rho-1]}(t) = L_{\mathbf{F}}^{\rho-1} \mathbf{h}(x) \quad (4.13)$$

$$\hat{y}^{[\rho]}(t) = L_{\mathbf{F}}^{\rho} \mathbf{h}(x) + L_{\mathbf{G}} L_{\mathbf{F}}^{\rho-1} \mathbf{h}(x) \hat{\mathbf{u}}(t) \quad (4.14)$$

Derivando la ecuación (4.14) hasta el orden $r + \rho$, se obtiene:

$$\hat{y}^{[\rho+1]}(t) = L_{\mathbf{F}}^{\rho+1} \mathbf{h}(x) + p_1(x, \hat{\mathbf{u}}) + L_{\mathbf{G}} L_{\mathbf{F}}^{\rho-1} \mathbf{h}(x) \hat{\mathbf{u}}^{[1]}(t) \quad (4.15)$$

$$\hat{y}^{[\rho+2]}(t) = L_{\mathbf{F}}^{\rho+2} \mathbf{h}(x) + p_2(x, \hat{\mathbf{u}}, \hat{\mathbf{u}}^{[1]}) + L_{\mathbf{G}} L_{\mathbf{F}}^{\rho-1} \mathbf{h}(x) \hat{\mathbf{u}}^{[2]}(t) \quad (4.16)$$

$$\vdots = \vdots \quad (4.17)$$

$$\hat{y}^{[\rho+r]}(t) = L_{\mathbf{F}}^{\rho+r} \mathbf{h}(x) + p_r(x, \hat{\mathbf{u}}, \dots, \hat{\mathbf{u}}^{[r-1]}) + L_{\mathbf{G}} L_{\mathbf{F}}^{\rho-1} \mathbf{h}(x) \hat{\mathbf{u}}^{[r]}(t) \quad (4.18)$$

donde, por ejemplo, $p_1(x, \hat{\mathbf{u}}) = L_{\mathbf{G}} L_{\mathbf{F}}^{\rho} \mathbf{h}(x) \hat{\mathbf{u}}(t) + \frac{dL_{\mathbf{G}} L_{\mathbf{F}}^{\rho-1} \mathbf{h}(x)}{dt} \hat{\mathbf{u}}(t)$, y además los p_i son funciones no lineales de alta complejidad.

Acomodando todas las derivadas de la salida, se define el siguiente vector:

$$\hat{Y}(t) = \begin{bmatrix} \hat{y}^{[0]} \\ \hat{y}^{[1]} \\ \hat{y}^{[2]} \\ \vdots \\ \hat{y}^{[\rho]} \\ \hat{y}^{[\rho+1]} \\ \vdots \\ \hat{y}^{[\rho+r]} \end{bmatrix} = \begin{bmatrix} L_{\mathbf{F}}^0 \mathbf{h}(x) \\ L_{\mathbf{F}}^1 \mathbf{h}(x) \\ L_{\mathbf{F}}^2 \mathbf{h}(x) \\ \vdots \\ L_{\mathbf{F}}^{\rho} \mathbf{h}(x) \\ L_{\mathbf{F}}^{\rho+1} \mathbf{h}(x) \\ \vdots \\ L_{\mathbf{F}}^{\rho+r} \mathbf{h}(x) \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ H(\hat{\mathbf{u}}) \end{bmatrix} \quad (4.19)$$

donde,

$$H(\hat{\mathbf{u}}) = \begin{bmatrix} L_{\mathbf{G}}L_{\mathbf{F}}^{\rho-1}\mathbf{h}(x)\hat{\mathbf{u}}^{[0]}(t) \\ p_1(x, \hat{\mathbf{u}}) + L_{\mathbf{G}}L_{\mathbf{F}}^{\rho-1}h(x)\hat{\mathbf{u}}^{[1]}(t) \\ p_2(x, \hat{\mathbf{u}}, \hat{\mathbf{u}}^{[1]}) + L_{\mathbf{G}}L_{\mathbf{F}}^{\rho-1}\mathbf{h}(x)\hat{\mathbf{u}}^{[2]}(t) \\ \vdots \\ p_r(x, \hat{\mathbf{u}}, \dots, \hat{\mathbf{u}}^{[r-1]}) + L_{\mathbf{G}}L_{\mathbf{F}}^{\rho-1}\mathbf{h}(x)\hat{\mathbf{u}}^{[r]}(t) \end{bmatrix} \quad (4.20)$$

Usando la ecuación (4.19), la ecuación (4.8) se reescribe, como:

$$\hat{y}(t + \tau|t) \approx \mathcal{T}(\tau)\hat{Y}(t) \quad (4.21)$$

donde $\mathcal{T} = \begin{bmatrix} \mathcal{I} & \bar{\tau} & \frac{\bar{\tau}^2}{2!} & \dots & \frac{\bar{\tau}^{\rho+r}}{(\rho+r)!} \end{bmatrix}$, $\bar{\tau} = \text{diag}\{\tau, \tau, \dots, \tau\}_{m \times m}$ y \mathcal{I} es la matriz identidad.

El vector $\bar{\mathbf{u}} = [\hat{\mathbf{u}}^{[0]}(t), \hat{\mathbf{u}}^{[1]}(t), \dots, \hat{\mathbf{u}}^{[r]}(t)] \in \mathbf{R}^{m(r+1)}$ se define como el conjunto de señales de control necesarias para calcular la predicción de la salida en un marco de predicción $\tau \in [0, T_2]$.

La ecuación (4.21) predice las salidas del sistema multivariable ante el conjunto de señales de control $\bar{\mathbf{u}}$.

4.7.2. Predicción de la Trayectoria de Referencia Futura

De manera similar a lo anterior procedemos a calcular la predicción de la trayectoria de referencia usando la expansion de Taylor hasta un orden $\rho + r$, para finalmente hallar el error de predicción sobre un horizonte T_2 :

$$\hat{w}(t + \tau|t) \approx \mathcal{T}(\tau)\hat{W}(t) \quad (4.22)$$

donde $\hat{W}(t) = [w^{[0]}(t) \ w^{[1]}(t) \ \dots \ w^{[\rho]}(t) \ \dots \ w^{[r+\rho]}(t)]^T$.

Ahora el índice de costo de (4.5) puede ser reformulado para luego encontrar un óptimo $\bar{\mathbf{u}}$ que lo minimice.

4.7.3. Minimización de la Función de Costo

Todo algoritmo de control predictivo contempla el uso de una función de costo de índice cuadrático, ver ecuación (4.5), que penaliza el error entre la salida predicha y la trayectoria de referencia futura. La misma puede ser reescrita usando las ecuaciones (4.21) y (4.22), como:

$$J = \frac{1}{2} \int_0^{T_2} (\hat{y}(t + \tau|t) - \hat{w}(t + \tau|t))^T (\hat{y}(t + \tau|t) - \hat{w}(t + \tau|t)) d\tau \quad (4.23)$$

$$= \frac{1}{2} (\hat{Y}(t) - \hat{W}(t))^T \int_0^{T_2} \mathcal{T}^T(\tau) \mathcal{T}(\tau) d\tau (\hat{Y}(t) - \hat{W}(t)) \quad (4.24)$$

$$= \frac{1}{2} (\hat{Y}(t) - \hat{W}(t))^T \bar{\mathcal{T}} (\hat{Y}(t) - \hat{W}(t)) \quad (4.25)$$

donde, $\bar{\mathcal{T}}$ es una matriz de $m(\rho + r + 1) \times m(\rho + r + 1)$ y viene definida por:

$$\bar{\mathcal{T}} = \int_0^{T_2} \mathcal{T}^T(\tau) \mathcal{T}(\tau) d\tau = \begin{bmatrix} \bar{\mathcal{T}}_{(1,1)} & \cdots & \bar{\mathcal{T}}_{(1,\rho+r+1)} \\ \vdots & \ddots & \vdots \\ \bar{\mathcal{T}}_{(\rho+r+1,1)} & \cdots & \bar{\mathcal{T}}_{(\rho+r+1,\rho+r+1)} \end{bmatrix} \quad (4.26)$$

La matriz $\bar{\mathcal{T}}$ puede también particionarse en submatrices, esto será de mucha utilidad como se verá más adelante:

$$\bar{\mathcal{T}} = \begin{bmatrix} \bar{\mathcal{T}}_{\rho\rho} & | & \bar{\mathcal{T}}_{\rho r} \\ - & + & - \\ \bar{\mathcal{T}}_{\rho r}^T & | & \bar{\mathcal{T}}_{rr} \end{bmatrix} \quad (4.27)$$

donde,

$$\bar{\mathcal{T}}_{rr} = \begin{bmatrix} \bar{\mathcal{T}}_{(\rho+1,\rho+1)} & \cdots & \bar{\mathcal{T}}_{(\rho+1,\rho+r+1)} \\ \vdots & \ddots & \vdots \\ \bar{\mathcal{T}}_{(\rho+r+1,\rho+1)} & \cdots & \bar{\mathcal{T}}_{(\rho+r+1,\rho+r+1)} \end{bmatrix} \quad (4.28)$$

,

$$\bar{\mathcal{T}}_{\rho r} = \begin{bmatrix} \bar{\mathcal{T}}_{(1,\rho+1)} & \cdots & \bar{\mathcal{T}}_{(1,\rho+r+1)} \\ \vdots & \ddots & \vdots \\ \bar{\mathcal{T}}_{(\rho,\rho+1)} & \cdots & \bar{\mathcal{T}}_{(\rho,\rho+r+1)} \end{bmatrix} \quad (4.29)$$

con: $\bar{T}_{\rho\rho} \in R^{m\rho \times m\rho}$, $\bar{T}_{\rho r} \in R^{m\rho \times m(r+1)}$ y $\bar{T}_{rr} \in R^{m(r+1) \times m(r+1)}$.

Los elementos ij , vienen expresados por:

$$\bar{T}_{(i,j)} = \frac{\bar{T}^{i+j-1}}{(i-1)!(j-1)!(i+j-1)}, \quad i, j = 1, \dots, \rho + r + 1 \quad (4.30)$$

con, $\bar{T} = \text{diag}\{T_2, T_2, \dots, T_2\} \in R^{m \times m}$

Además se observa que,

$$\hat{Y}(t) - \hat{W}(t) = M(x(t)) + \begin{bmatrix} [0]_{m\rho \times 1} \\ [H(\hat{\mathbf{u}})]_{m(r+1) \times 1} \end{bmatrix} \quad (4.31)$$

donde:

$$M(x(t)) = \begin{bmatrix} L_{\mathbf{f}}^0 \mathbf{h}(x) \\ L_{\mathbf{f}}^1 \mathbf{h}(x) \\ L_{\mathbf{f}}^2 \mathbf{h}(x) \\ \vdots \\ L_{\mathbf{f}}^{\rho} \mathbf{h}(x) \\ L_{\mathbf{f}}^{\rho+1} \mathbf{h}(x) \\ \vdots \\ L_{\mathbf{f}}^{\rho+r} \mathbf{h}(x) \end{bmatrix} - \begin{bmatrix} w^{[0]}(t) \\ w^{[1]}(t) \\ w^{[2]}(t) \\ \vdots \\ w^{[r+\rho]}(t) \end{bmatrix} \quad (4.32)$$

Finalmente, la condición necesaria para obtener la ley de control óptimo, viene dada por:

$$\frac{\partial J}{\partial \hat{\mathbf{u}}} = 0 \quad (4.33)$$

teniendo presente la ecuación (4.25) en (4.33), obtenemos:

$$\frac{\partial J}{\partial \hat{\mathbf{u}}} = 0 \quad (4.34)$$

$$= \frac{\partial}{\partial \hat{\mathbf{u}}} \left\{ \frac{1}{2} \left(\hat{Y}(t) - \hat{W}(t) \right)^T \bar{T} \left(\hat{Y}(t) - \hat{W}(t) \right) \right\} \quad (4.35)$$

Con la ecuación (4.31) la ecuación (4.35), se simplifica a:

$$0 = \frac{\partial}{\partial \hat{\mathbf{u}}} \left\{ \frac{1}{2} \left(\hat{Y}(t) - \hat{W}(t) \right)^T \bar{\mathcal{T}} \left(\hat{Y}(t) - \hat{W}(t) \right) \right\} \quad (4.36)$$

$$= \frac{\partial}{\partial \hat{\mathbf{u}}} \left(\hat{Y}(t) - \hat{W}(t) \right)^T \bar{\mathcal{T}} \left(\hat{Y}(t) - \hat{W}(t) \right) \quad (4.37)$$

$$= \frac{\partial}{\partial \hat{\mathbf{u}}} \left[M(x(t)) + \begin{bmatrix} [0]_{m\rho \times 1} \\ [H(\hat{\mathbf{u}})]_{m(r+1) \times 1} \end{bmatrix} \right]^T \bar{\mathcal{T}} \left(\hat{Y}(t) - \hat{W}(t) \right) \quad (4.38)$$

$$= \begin{bmatrix} 0^T & \frac{\partial}{\partial \hat{\mathbf{u}}} H^T(\hat{\mathbf{u}}) \end{bmatrix} \begin{bmatrix} \bar{\mathcal{T}}_{\rho\rho} & | & \bar{\mathcal{T}}_{\rho r} \\ - & + & - \\ \bar{\mathcal{T}}_{\rho r}^T & | & \bar{\mathcal{T}}_{rr} \end{bmatrix} \left[M(x(t)) + \begin{bmatrix} [0]_{m\rho \times 1} \\ [H(\hat{\mathbf{u}})]_{m(r+1) \times 1} \end{bmatrix} \right]$$

operando esta última ecuación, se tiene:

$$\frac{\partial}{\partial \hat{\mathbf{u}}} H^T(\hat{\mathbf{u}}) \begin{bmatrix} \bar{\mathcal{T}}_{\rho r}^T & \bar{\mathcal{T}}_{rr} \end{bmatrix} M(x(t)) + \frac{\partial}{\partial \hat{\mathbf{u}}} H^T(\hat{\mathbf{u}}) \bar{\mathcal{T}}_{rr} H(\hat{\mathbf{u}}) = 0 \quad (4.39)$$

si diferenciamos a la ecuación (4.20) respecto a $\hat{\mathbf{u}}$, vamos a obtener lo siguiente,

$$\frac{\partial}{\partial \hat{\mathbf{u}}} H(\hat{\mathbf{u}}) = \begin{bmatrix} L_{\mathbf{G}} L_{\mathbf{F}}^{\rho-1} h(x) & [0]_{m \times m} & \cdots & [0]_{m \times m} & [0]_{m \times m} \\ * & L_{\mathbf{G}} L_{\mathbf{F}}^{\rho-1} h(x) & \cdots & [0]_{m \times m} & [0]_{m \times m} \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ * & * & \cdots & L_{\mathbf{G}} L_{\mathbf{F}}^{\rho-1} h(x) & [0]_{m \times m} \\ * & * & \cdots & [0]_{m \times m} & L_{\mathbf{G}} L_{\mathbf{F}}^{\rho-1} h(x) \end{bmatrix} \quad (4.40)$$

como $L_{\mathbf{G}} L_{\mathbf{F}}^{\rho-1} h(x)$ es invertible de acuerdo a la definición de grado relativo, entonces $\frac{\partial}{\partial \hat{\mathbf{u}}} H(\hat{\mathbf{u}})$ es también invertible.

Por tanto de la ecuación (4.39) se obtiene que:

$$H(\hat{\mathbf{u}}) = - \begin{bmatrix} \bar{\mathcal{T}}_{rr}^{-1} \bar{\mathcal{T}}_{\rho r}^T & [\mathcal{I}]_{m(r+1) \times m(r+1)} \end{bmatrix} M(x(t)) \quad (4.41)$$

En este punto cabe indicar nuevamente que $H(\hat{\mathbf{u}})$ depende del conjunto de señales de control $\bar{\mathbf{u}} = \left[\hat{\mathbf{u}}^{[0]}(t), \hat{\mathbf{u}}^{[1]}(t) \dots \hat{\mathbf{u}}^{[r]}(t) \right] \in \mathbf{R}^{m(r+1)}$ y por la estrategia de hori-

zonte móvil, ver sección 4.2, sólo se envía al sistema la primera de $\bar{\mathbf{u}}$, en este caso es $\hat{\mathbf{u}}^{[0]}(t) = \hat{\mathbf{u}}(t)$.

Por lo tanto, se busca calcular $\hat{\mathbf{u}}(t)$, para esto desdoblamos la ecuación (4.41) como:

$$\begin{bmatrix} L_{\mathbf{G}}L_{\mathbf{F}}^{\rho-1}\mathbf{h}(x)\hat{\mathbf{u}}(t) \\ \text{---} \\ \vdots \end{bmatrix} = - \begin{bmatrix} & | & 1 & 0 & 0 & \cdots & 0 \\ & | & 0 & 1 & 0 & \cdots & 0 \\ \bar{\mathcal{T}}_{rr}^{-1}\bar{\mathcal{T}}_{\rho r}^T & | & \vdots & & \ddots & & 0 \\ & | & 0 & 0 & 0 & \cdots & 1 \end{bmatrix} \begin{bmatrix} L_{\mathbf{F}}^0\mathbf{h}(x) - w^{[0]}(t) \\ L_{\mathbf{F}}^1\mathbf{h}(x) - w^{[1]}(t) \\ L_{\mathbf{F}}^2\mathbf{h}(x) - w^{[2]}(t) \\ \vdots \\ L_{\mathbf{F}}^{\rho}\mathbf{h}(x) - w^{[\rho]}(t) \\ \text{---} \\ \vdots \end{bmatrix} \quad (4.42)$$

Considerando que \mathcal{K} son las m primeras filas de la matriz $\bar{\mathcal{T}}_{rr}^{-1}\bar{\mathcal{T}}_{\rho r}^T$, entonces se tiene:

$$L_{\mathbf{G}}L_{\mathbf{F}}^{\rho-1}\mathbf{h}(x)\hat{\mathbf{u}}(t) = -\mathcal{K}M_p + w^{[\rho]}(t) - L_{\mathbf{F}}^{\rho}\mathbf{h}(x) \quad (4.43)$$

donde,

$$M_p = \left[(\mathbf{h}(x) - w^{[0]}(t))^T \quad (L_{\mathbf{F}}^1\mathbf{h}(x) - w^{[1]}(t))^T \quad \cdots \quad (L_{\mathbf{F}}^{\rho-1}\mathbf{h}(x) - w^{[\rho-1]}(t))^T \right]^T \quad (4.44)$$

finalmente de la ecuación (4.43) despejamos la señal $\hat{\mathbf{u}}(t)$, obteniéndose:

$$\hat{\mathbf{u}}(t) = - (L_{\mathbf{G}}L_{\mathbf{F}}^{\rho-1}\mathbf{h}(x))^{-1} \{ \mathcal{K}M_p + L_{\mathbf{F}}^{\rho}\mathbf{h}(x) - w^{[\rho]}(t) \} \quad (4.45)$$

como podemos la ley de control es una función analítica y para obtenerla no hace uso de algún algoritmo numérico.

Si observamos la ley de control de la ecuación (4.45) nos podemos dar cuenta que ésta es parecida a la ley de control del algoritmo de linealización exacta por realimentación de estados pero la diferencia es que los coeficientes de la ley de control, dados por la matriz \mathcal{K} , son resultado de minimizar la función de costo.

4.8. Aplicación al Control de Velocidad de un Motor DC

En esta sección se aplicará el control predictivo generalizado no lineal, al control de posición ($m=1$) de un motor de corriente continua. El modelo matemático de este proceso, ver sección 3.6, es nuevamente escrito a continuación:

$$\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= -\left(\frac{n^2 KE}{RM} + \frac{B}{M}\right)x_2 - \frac{N}{M}\sin(x_1) - \frac{T_N}{M} + \frac{nKK_{act}}{RM}u \\ y &= x_1 \end{aligned} \quad (4.46)$$

Comparando la ecuación (4.46) con la ecuación de un modelo no lineal (3.48), concluimos que:

$$\begin{aligned} \mathbf{F}(\mathbf{x}) &= \begin{bmatrix} x_2 \\ -\left(\frac{n^2 KE}{RM} + \frac{B}{M}\right)x_2 - \frac{N}{M}\sin(x_1) - \frac{T_N}{M} \end{bmatrix} \\ \mathbf{G}(\mathbf{x}) &= \begin{bmatrix} 0 \\ \frac{nKK_{act}}{RM} \end{bmatrix} \\ h(\mathbf{x}) &= x_1 \end{aligned} \quad (4.47)$$

De las anteriores relaciones podemos hallar lo siguiente:

$$L_{\mathbf{F}}h(\mathbf{x}) = \frac{\partial h(\mathbf{x})}{\partial \mathbf{x}} \mathbf{F} = x_2 \quad (4.48)$$

$$L_{\mathbf{F}}^2 h(\mathbf{x}) = \frac{\partial L_{\mathbf{F}}h(\mathbf{x})}{\partial \mathbf{x}} \mathbf{F} = -\left(\frac{n^2 KE}{RM} + \frac{B}{M}\right)x_2 - \frac{N}{M}\sin(x_1) \quad (4.49)$$

$$L_{\mathbf{G}}L_{\mathbf{F}}h(\mathbf{x}) = \frac{\partial L_{\mathbf{F}}h(\mathbf{x})}{\partial \mathbf{x}} \mathbf{G} = \frac{nK_{act}K}{MR} \neq 0 \quad (4.50)$$

De la ecuación (4.50) se deduce que el grado relativo (ρ) del sistema es dos. Adicionalmente, consideremos $r=3$, ver ecuación (4.8). Con esto, las submatrices

de $\bar{\mathcal{T}}$ vienen dadas por:

$$\bar{\mathcal{T}}_{rr} = \begin{bmatrix} \bar{\mathcal{T}}_{(3,3)} & \bar{\mathcal{T}}_{(3,4)} & \bar{\mathcal{T}}_{(3,5)} & \bar{\mathcal{T}}_{(3,6)} \\ \bar{\mathcal{T}}_{(4,3)} & \bar{\mathcal{T}}_{(4,4)} & \bar{\mathcal{T}}_{(4,5)} & \bar{\mathcal{T}}_{(4,6)} \\ \bar{\mathcal{T}}_{(5,3)} & \bar{\mathcal{T}}_{(5,4)} & \bar{\mathcal{T}}_{(5,5)} & \bar{\mathcal{T}}_{(5,6)} \\ \bar{\mathcal{T}}_{(6,3)} & \bar{\mathcal{T}}_{(6,4)} & \bar{\mathcal{T}}_{(6,5)} & \bar{\mathcal{T}}_{(6,6)} \end{bmatrix} \quad (4.51)$$

, y

$$\bar{\mathcal{T}}_{\rho r} = \begin{bmatrix} \bar{\mathcal{T}}_{(1,3)} & \bar{\mathcal{T}}_{(1,4)} & \bar{\mathcal{T}}_{(1,5)} & \bar{\mathcal{T}}_{(1,6)} \\ \bar{\mathcal{T}}_{(2,3)} & \bar{\mathcal{T}}_{(2,4)} & \bar{\mathcal{T}}_{(2,5)} & \bar{\mathcal{T}}_{(2,6)} \end{bmatrix} \quad (4.52)$$

Tener presente que los elementos ij , ecuación (4.30), vienen expresados por:

$$\bar{\mathcal{T}}_{(i,j)} = \frac{\bar{\mathcal{T}}^{i+j-1}}{(i-1)!(j-1)!(i+j-1)}, \quad i, j = 1, \dots, \rho + r + 1 \quad (4.53)$$

Luego, en la ecuación (4.41) tener en cuenta que el proceso tiene sólo una salida ($y=x_1$), esto es $m=1$, además $\rho = 2$ y $r=3$.

$$H(\hat{u}) = - \begin{bmatrix} \bar{\mathcal{T}}_{33}^{-1} \bar{\mathcal{T}}_{23}^T & \mathcal{I}_{4 \times 4} \end{bmatrix} M(x(t)) \quad (4.54)$$

Luego, desdoblado la ecuación (4.54), tal como se aprecia en (4.42), tendremos:

$$\begin{bmatrix} L_{\mathbf{G}} L_{\mathbf{F}}^1 h(x) \hat{u}(t) \\ \text{---} \\ \vdots \end{bmatrix} = - \begin{bmatrix} & | & 1 & 0 & 0 & 0 \\ & | & 0 & 1 & 0 & 0 \\ \bar{\mathcal{T}}_{rr}^{-1} \bar{\mathcal{T}}_{\rho r}^T & | & 0 & 0 & 1 & 0 \\ & | & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} L_{\mathbf{F}}^0 h(x) - w^{[0]}(t) \\ L_{\mathbf{F}}^1 h(x) - w^{[1]}(t) \\ L_{\mathbf{F}}^2 h(x) - w^{[2]}(t) \\ \text{---} \\ \vdots \end{bmatrix} \quad (4.55)$$

De ésta última ecuación hallamos la ley de control, ver ecuación (4.45), que viene dada por:

$$\hat{u}(t) = - (L_{\mathbf{G}}L_{\mathbf{F}}h(x))^{-1} \{ \mathcal{K}M_p + L_{\mathbf{F}}^2h(x) - w^{[2]}(t) \} \quad (4.56)$$

Donde:

$$L_{\mathbf{F}}h(\mathbf{x}) = x_2 \quad (4.57)$$

$$L_{\mathbf{F}}^2h(\mathbf{x}) = - \left(\frac{n^2KE}{RM} + \frac{B}{M} \right) x_2 - \frac{N}{M} \sin(x_1) \quad (4.58)$$

$$L_{\mathbf{G}}L_{\mathbf{F}}h(\mathbf{x}) = \frac{nK_{act}K}{MR} \quad (4.59)$$

$$\mathcal{K} = \begin{bmatrix} \frac{280}{3T_2^2} & \frac{16}{T_2} \end{bmatrix} \quad (4.60)$$

$$M_p = \begin{bmatrix} h(x) - w^{[0]}(t) \\ L_{\mathbf{F}}h(x) - w^{[1]}(t) \end{bmatrix} \quad (4.61)$$

Tener presente que el vector \mathcal{K} viene dado por la primera ($m=1$) fila de la matriz $\bar{\mathcal{T}}_{rr}^{-1}\bar{\mathcal{T}}_{\rho r}^T$.

4.8.1. Simulación del Controlador para el Proceso No Lineal

El algoritmo de control ha sido implementado en MATLAB. Las condiciones bajo las cuales se ha realizado la simulación son mencionadas más adelante.

Para Trayectoria Constante Se ha considerado que todos los estados parten del reposo. El tiempo total de simulación es de 5 seg. y el tiempo de muestreo es de 10 ms. Además, un horizonte de predicción de 50 ms, esto es, la salida se estaría prediciendo hasta 5 muestras. Es importante mencionar que en el programa se ha incluido la compensación de la fricción de Coulomb, y se ha limitado la señal de control para evitar la saturación de los amplificadores.

En la Figura 4.3, se muestra la salida controlada para la posición angular del brazo de 1 grado de libertad (GDL) para una referencia de 90°. El tiempo de

establecimiento es de 0.5 seg. y corresponde a un sistema subamortiguado, alcanzando un máximo valor de 91° .

Además, en la misma Figura 4.3 se muestra la comparación entre las dos estrategias estudiadas en la presente tesis. Como se observa el CPGNL presenta un mejor tiempo de establecimiento pero con un sobreimpulso que alcanza los 91.3° aproximadamente. La ley de control para el CPGNL se muestra en la Figura 4.4.

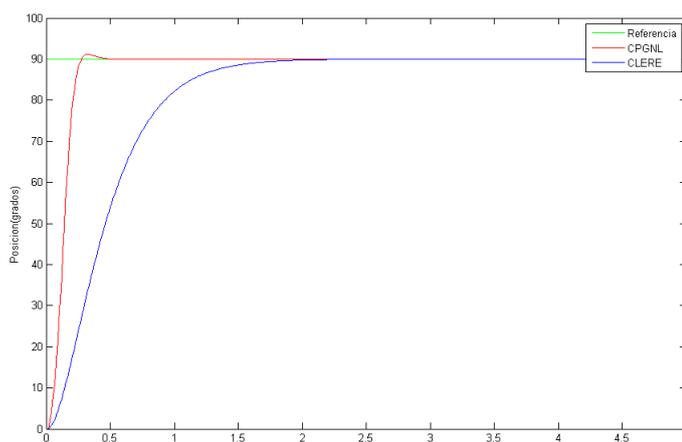


Figura 4.3: Comparación entre las estrategias CLERE y CPGNL

Se observa que al inicio la ley de control adopta un valor máximo de 1.4v, debido a un reducido tiempo de establecimiento, y luego de 2 seg. se establece en 0.4v.

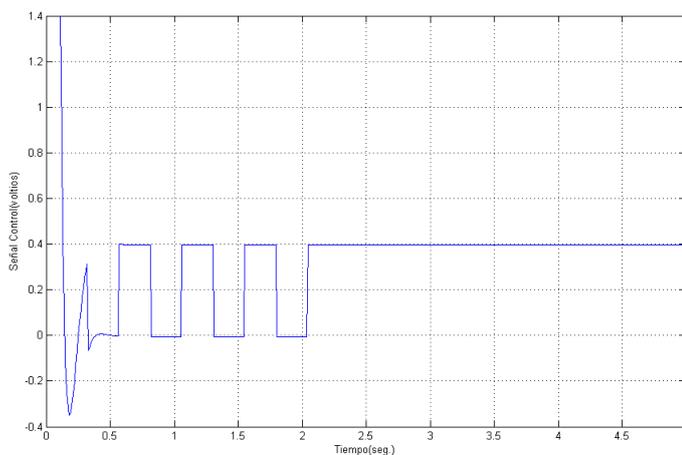


Figura 4.4: Señal de control para el motor DC

CAPÍTULO 5

Simulaciones del Sistema

El proceso a controlar corresponde a un sistema robótico, ver Figura 5.1, de cadena cinemática abierta de tres grados de libertad (GDL) en configuración esférica.

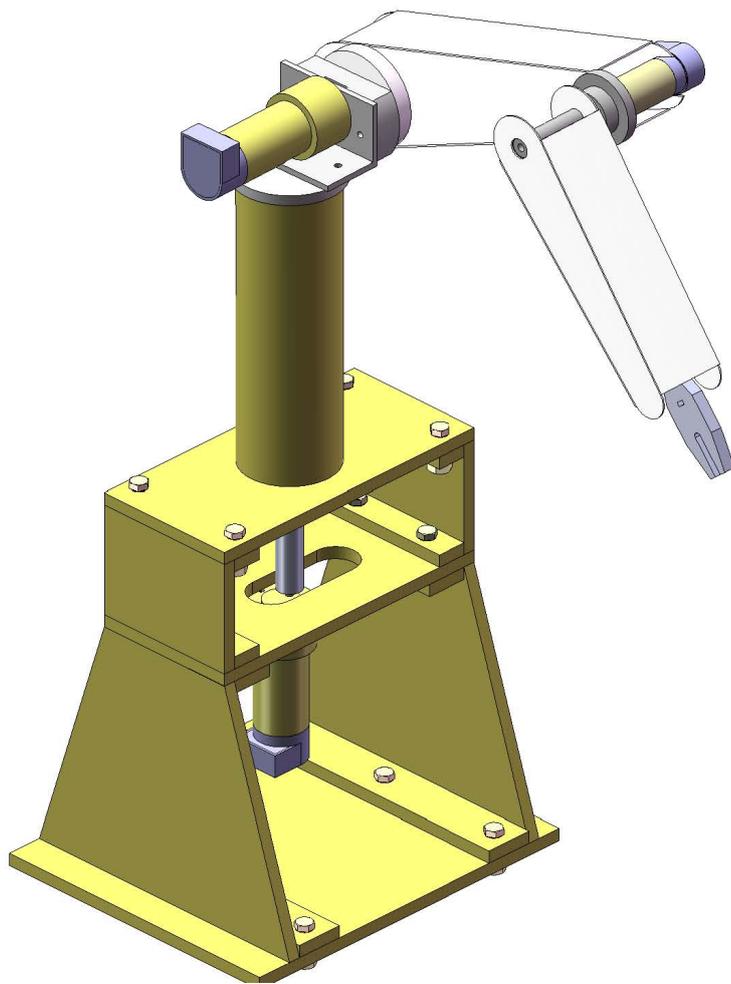


Figura 5.1: Estructura del robot

Los GDL se muestran en la Figura 5.2, y describen a continuación:

- Un primer grado, está dado por la rotación de su base alrededor de un eje perpendicular al plano de su base.
- Un segundo, ésta dado por la rotación de su brazo alrededor de su eje hombro.
- Y por último, dado por la rotación de su antebrazo alrededor de su eje codo.

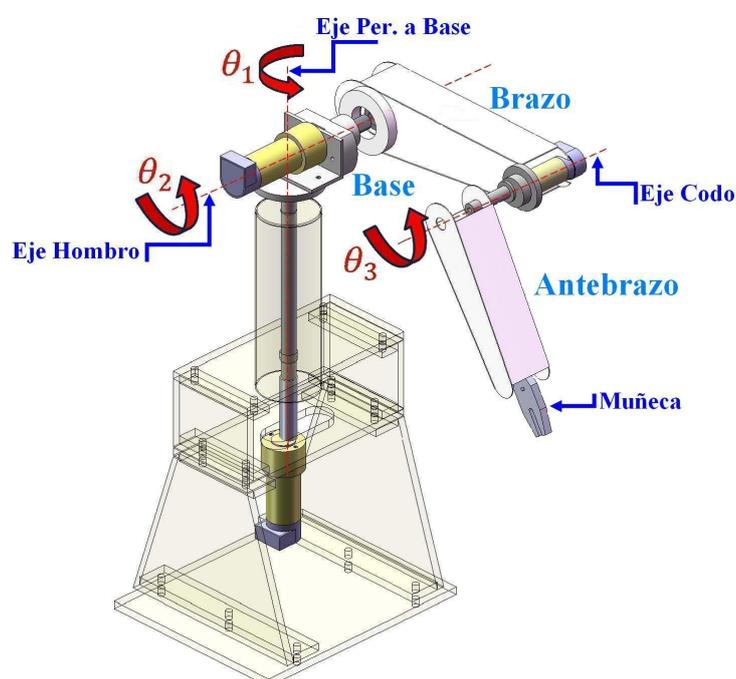


Figura 5.2: Grados de libertad del robot

5.1. Descripción Detallada del Robot

En la siguiente sección se detallan los elementos que componen al robot.

5.1.1. Primer Eslabón: Base

Es accionado por un motorreductor DC1 de 24VDC con torque pico de 18 N.m, con encoder óptico incremental de 500PPR incorporado. Éste se encuentra instalado en el eslabón 0, además este eslabón está constituido por un eje1 el cual

conecta al motor DC1 con un disco sobre el cual descanza el motorreductor DC2, ver Figura 5.3.

5.1.2. Segundo Eslabón: Brazo

Es accionado por un motorreductor DC2 de 24VDC con torque pico de 18 N.m, con encoder óptico incremental de 500PPR. Este eslabón está consituido por un brazo: en un extemo de éste se halla una bocina que se une rígidamente con el motor DC2 a través del eje2 y en el otro extremo se encuentra el motorreductor DC3, el mismo que se conecta al eslabón 3 através del eje3, ver Figura 5.3.

5.1.3. Tercer Eslabón: Antebrazo

Este eslabón es accionado por el motorreductor DC3 de 24VDC con torque pico de 2.3 N.m, con encoder óptico incremental de 500PPR. En el extremo libre de este eslabón se instala la tenaza para realizar la tarea de manipulación, ver Figura 5.3.

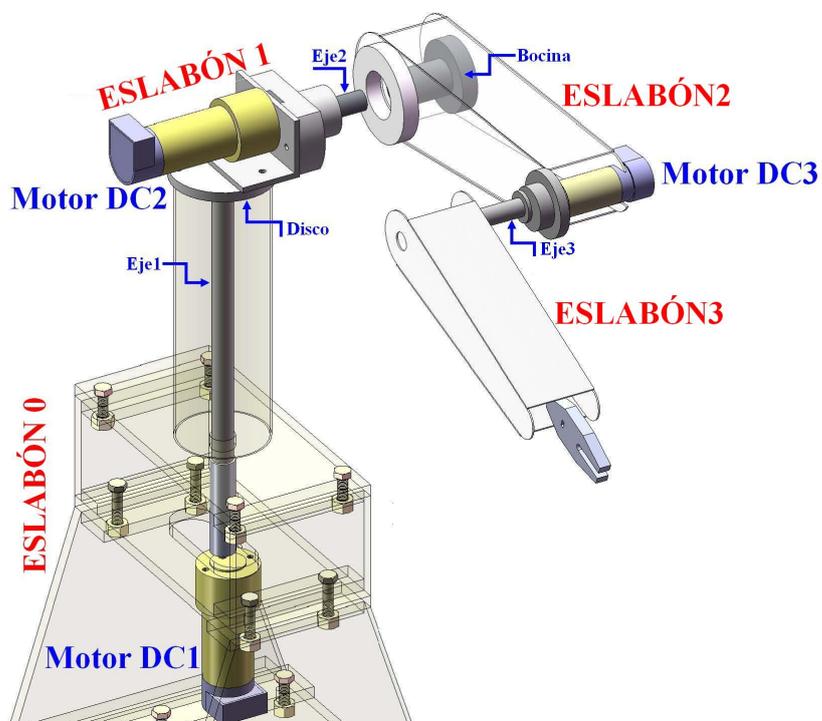


Figura 5.3: Elementos del robot

5.2. Modelo Cinemático del Robot Propuesto

Como primer paso seleccionaremos los sistemas de referencia solidarios a los eslabones mediante la Convención de Denavit Hartenberg (D-H), ver en la sección 2.6.

Éstos han sido seleccionados tal como se aprecia en Figura 5.4. Posteriormente obtenemos los parámetros D-H del robot, ver cuadro 5.1.

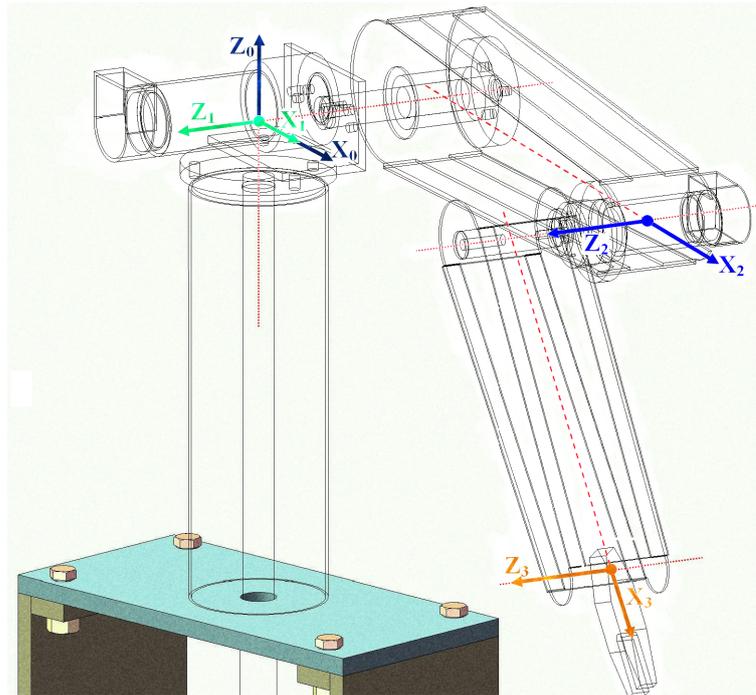


Figura 5.4: Sistemas de referencia según D-H

Cuadro 5.1: Parámetros D-H

i	α_i	θ_i	d_i	a_i
1	$\pi/2$	q_1	0	0
2	0	q_2	$-d_2$	a_2
3	0	q_3	d_3	a_3

5.2.1. Cálculo de las Matrices de Transformación Homogénea

Procedemos a calcular las matrices de transformación homogénea, ver en la sección 2.6.3.

Con esto obtenemos las matrices $\mathbf{A}_1^0, \mathbf{A}_2^1$ y \mathbf{A}_3^2 .

\mathbf{A}_1^0 : Proporciona información de posición y orientación del sistema de referencia S_1 respecto al sistema de referencia fijo S_0 , y viene dado por:

$$\mathbf{A}_1^0 = \begin{bmatrix} \cos(q_1) & 0 & \sin(q_1) & 0 \\ \sin(q_1) & 0 & -\cos(q_1) & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (5.1)$$

\mathbf{A}_2^1 : Proporciona información de posición y orientación del sistema de referencia S_2 respecto al sistema de referencia S_1 , y viene dado por:

$$\mathbf{A}_2^1 = \begin{bmatrix} \cos(q_2) & -\sin(q_2) & 0 & a_2 \cos(q_2) \\ \sin(q_2) & \cos(q_2) & 0 & a_2 \sin(q_2) \\ 0 & 0 & 1 & -d_2 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (5.2)$$

\mathbf{A}_3^2 : Proporciona información de posición y orientación del sistema de referencia S_3 respecto al sistema de referencia S_3 , y viene dado por:

$$\mathbf{A}_3^2 = \begin{bmatrix} \cos(q_3) & -\sin(q_3) & 0 & a_3 \cos(q_3) \\ \sin(q_2) & \cos(q_2) & 0 & a_3 \sin(q_3) \\ 0 & 0 & 1 & d_3 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (5.3)$$

Finalmente se recurre a la ecuación (2.22) para obtener:

\mathbf{A}_1^0 : Proporciona información de posición y orientación del sistema de referencia S_1 respecto al sistema de referencia fijo S_0 , y viene dado por:

$$\mathbf{A}_1^0 = \begin{bmatrix} \cos(q_1) & 0 & \sin(q_1) & 0 \\ \sin(q_1) & 0 & -\cos(q_1) & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (5.4)$$

\mathbf{A}_2^0 : Proporciona información de posición y orientación del sistema de referencia S_2 respecto al sistema fijo S_0 , y viene dado por:

$$\mathbf{A}_2^0 = \begin{bmatrix} \cos(q_1) \cos(q_2) & -\cos(q_1) \sin(q_2) & \sin(q_1) & a_2 \cos(q_1) \cos(q_2) - d_2 \sin(q_1) \\ \sin(q_1) \cos(q_2) & -\sin(q_1) \sin(q_2) & -\cos(q_1) & a_2 \sin(q_1) \cos(q_2) + d_2 \cos(q_1) \\ \sin(q_2) & \cos(q_2) & 0 & a_2 \sin(q_2) \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (5.5)$$

\mathbf{A}_3^0 : Proporciona información de posición y orientación del sistema de referencia S_3 respecto al sistema de referencia fijo S_0 , y viene dado por:

$$\mathbf{A}_3^0 = \begin{bmatrix} c(q_1)c(q_2+q_3) & -c(q_1)s(q_2+q_3) & s(q_1) & a_3c(q_1)c(q_2+q_3) + (d_3-d_2)s(q_1) + a_2c(q_1)*c(q_2) \\ s(q_1)c(q_2+q_3) & -s(q_1)s(q_2+q_3) & -c(q_1) & a_3s(q_1)c(q_2+q_3) + (d_2-d_3)c(q_1) + a_2s(q_1)*c(q_2) \\ s(q_2+q_3) & c(q_2+q_3) & 0 & a_3s(q_2+q_3) + a_2s(q_2) \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (5.6)$$

donde $c=\cos(\cdot)$ y $s=\sin(\cdot)$

5.3. Cálculo de los Momentos de Inercia

El robot fue diseñado en SOLIDWORK 2007, ver Figura 5.5. Este progra-

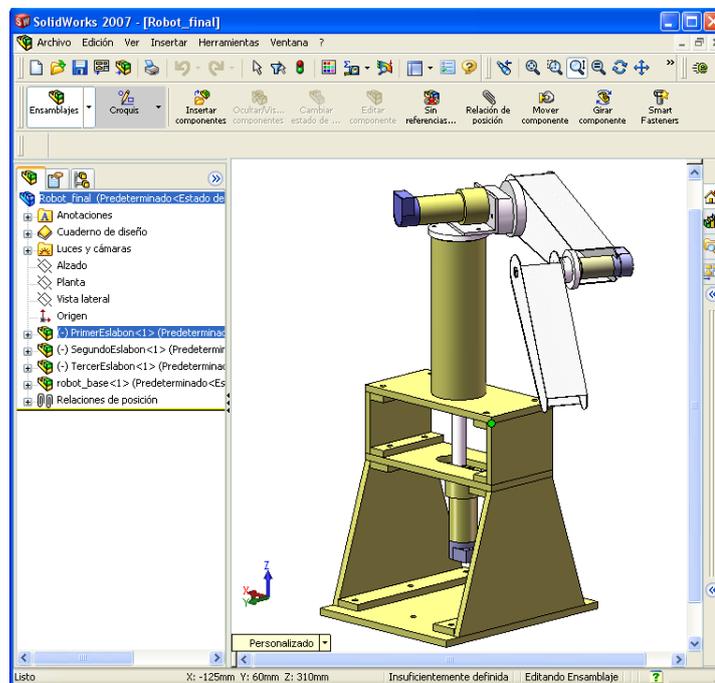


Figura 5.5: Se muestra el programa Solidwork en el que fue diseñado el robot de 3GDL

ma presenta una herramienta **Propiedades Físicas** que nos permite hallar los momentos y productos de inercia respecto a los sistemas de referencia D-H. A continuación se muestran los dibujos CAD de los eslabones, y a su derecha la tabla que contienen a sus parámetros físicos.

- Primer Eslabón, En la Figura 5.6 se muestran los sistemas de referencia de D-H (*En Azul: Eje Z, En Rojo: Eje X y En Verde: Eje Y*), adicionalmente se muestran un sistema en color rosado que corresponde a los ejes principales de inercia y su origen se halla en el centro de masa del eslabón.

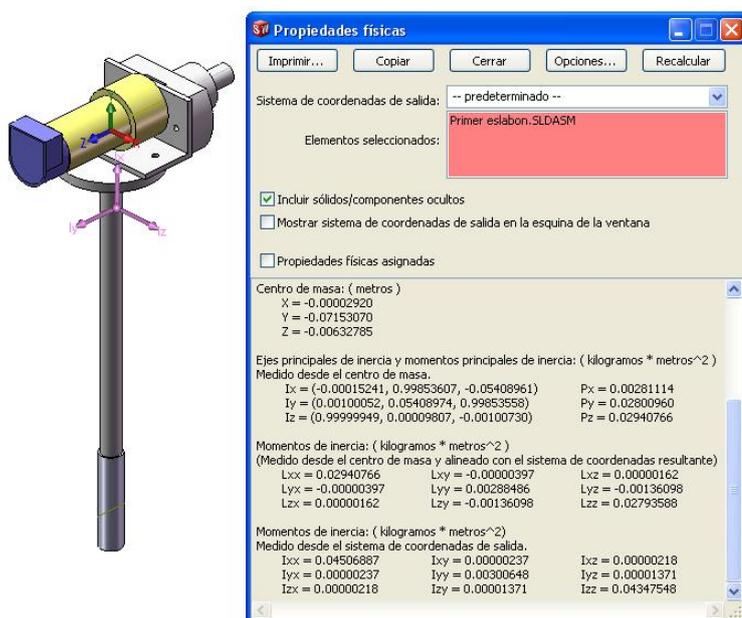


Figura 5.6: Cálculo de inercia correspondiente al eslabón 1 usando Solidwork

- Segundo Eslabón, En la Figura 5.7(a). se muestran los sistemas de referencia de D-H, así como los de sus ejes principales de inercia.
- Tercer Eslabón, En la Figura 5.7(b). se muestran los sistemas de referencia de D-H, así como los de sus ejes principales de inercia.

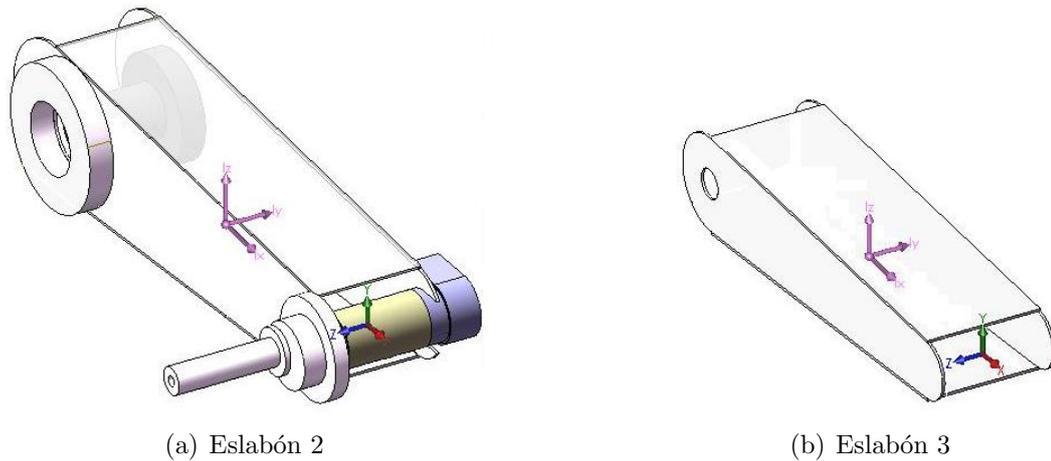


Figura 5.7: Sistemas inerciales y de centro de masa de los eslabones

5.4. Lista de variables

Las variables que intervienen se mencionan en las tablas 5.3 y 5.2, como son: Masa de los eslabones, momentos y productos de inercia cuyos valores se han obtenido con ayuda de SOLIDWORK 2007. Los valores de las variables correspondientes a los motores se han obtenido de su Datasheet.

Cuadro 5.2: Lista de variables de eslabones y motor

Símbolo	Descripción	Valor
I_{1xx}	Inercia eje x de S_1	0.02163237 Kg. m^2
I_{1yy}	Inercia al eje y de S_1	0.00404908 Kg. m^2
I_{1zz}	Inercia al eje z de S_1	0.01870108 Kg. m^2
I_{1xy}	Inercia al eje xy de S_1	0 Kg. m^2
I_{1xz}	Inercia al eje xz de S_1	0 Kg. m^2
I_{1yz}	Inercia al eje yz de S_1	-0.00020101 Kg. m^2
rCM_{1x}	Coord. x del CM en S_1	0 mt.
rCM_{1y}	Coord. y del CM en S_1	-0.05384880 mt.
rCM_{1z}	Coord. z del CM en S_1	-0.01268827 mt.
m_1	Masa del eslabón	1.993 Kg.
R_1	Resistencia del motor	2.49 Ω
K_1	Constante de ganancia del driver	1 V/V
K_{b1}	Constante FEM del motor	4.58e-2 V/rad/seg.
K_{m1}	Constante de torque	4.58e-2 N-m/A
J_{m1}	Inercia del motor	7.1e-6 Kg. m^2
B_{m1}	Constante CFV del motor	8.5e-4 Nms/rad
N_1	Reducción del motor	65.5
Cc_1	Constante de Coulumb	- Kg. m^2 /seg.

Cuadro 5.3: Lista de variables de eslabones y motor

Simbolo	Descripción	Valor
I_{2xx}	Inercia al eje x de S_2	0.00149129 Kg. m^2
I_{2yy}	Inercia al eje y de S_2	0.02538504 Kg. m^2
I_{2zz}	Inercia al eje z de S_2	0.02550188 Kg. m^2
I_{2xy}	Inercia al eje xy de S_2	0 Kg. m^2
I_{2xz}	Inercia al eje xz de S_2	0.00008690 Kg. m^2
I_{2yz}	Inercia al eje yz de S_2	0.00000407 Kg. m^2
rCM_{2x}	Coord. x del CM en S_2	-0.11997173 mt.
rCM_{2y}	Coord. y del CM en S_2	0.00070785 mt.
rCM_{2z}	Coord. z del CM en S_2	-0.00044496 mt.
m_2	Masa del eslabón	1.1068 Kg.
R_2	Resistencia del motor	2.49 Ω
K_2	Constante de ganancia del driver	1 V/V
K_{b2}	Constante FEM del motor	4.58e-2 V/rad/seg.
K_{m2}	Constante de torque	4.58e-2 N-m/A
J_{m2}	Inercia del motor	7.1e-6 Kg. m^2
B_{m2}	Constante CFV del motor	8.5e-4 Nms/rad
N_2	Reducción del motor	65.5
Cc_2	Constante de Coulumb	- Kg. m^2 /seg.
I_{3xx}	Inercia al eje x de S_3	0.00016023 Kg. m^2
I_{3yy}	Inercia al eje y de S_3	0.02538504 Kg. m^2
I_{3zz}	Inercia al eje z de S_3	0.00547069 Kg. m^2
I_{3xy}	Inercia al eje xy de S_3	0 Kg. m^2
I_{3xz}	Inercia al eje xz de S_3	0 Kg. m^2
I_{3yz}	Inercia al eje yz de S_3	0 Kg. m^2
rCM_{3x}	Coord. x del CM en S_3	-0.10640695 mt.
rCM_{3y}	Coord. y del CM en S_3	0 mt.
rCM_{3z}	Coord. z del CM en S_3	0 mt.
m_3	Masa del eslabón	0.132287 Kg.
R_3	Resistencia del motor	17 Ω
K_3	Constante de ganancia del driver	1 V/V
K_{b3}	Constante FEM del motor	4.36e-2 V/rad/seg.
K_{m3}	Constante de torque	4.36e-2 V/rad/seg.
J_{m3}	Momento de inercia del motor	1.6e-6 Kg. m^2
B_{m3}	Constante CFV del motor	1.1e-4 Nms/rad
N_3	Reducción del motor	60.5
Cc_3	Constante de Coulumb	- Kg. m^2 /seg.

5.5. Control por Linealización Exacta por Realimentación de Estados (CLERE) Aplicado al Manipulador Robótico Esférico de 3GDL

En esta sección se muestran las simulaciones de control por realimentación de estados aplicados a un sistema (robot) no lineal multivariable, de 3 entradas 3 salidas.

Para obtener las simulaciones se ha tenido en cuenta:

- El modelo en espacio estado del proceso, ver ecuación (2.64).
- El diseño del controlador no lineal multivariable basado en el método de linealización exacta por realimentación de estados, ver el capítulo 3.
- El diagrama de bloques presentado en la figura 5.8

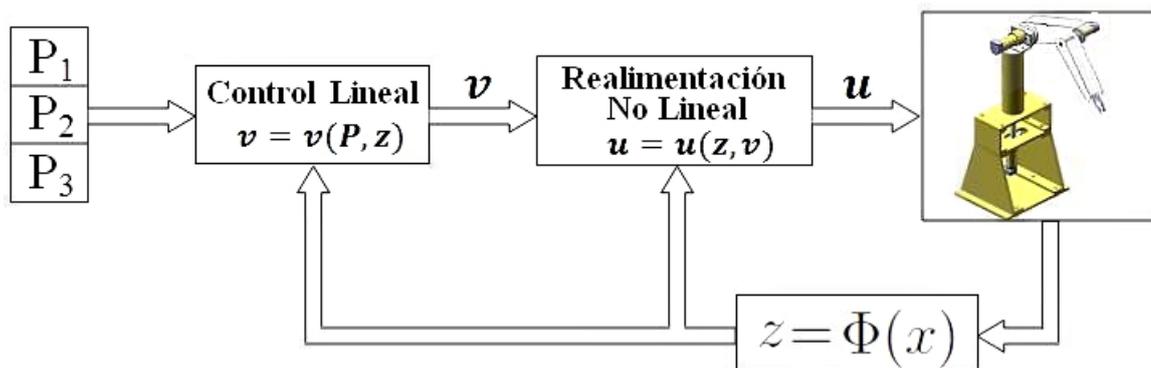


Figura 5.8: Diagrama de bloques del control por realimentación exacta

Las variables de estado x del proceso no lineal generan un cambio de coordenadas en una nueva ecuación de estados de estados z .

Con dichas variables realimentadas y las referencias P_1 , P_2 y P_3 , se determina las nuevas señales de control lineal v , mediante la técnica de ubicación de polos. Así mismo con estas señales de control v y con las variables de estado realimentados z se generan las señales de control real u que son implementadas para el control del proceso. Para la simulación de este controlador se ha utilizado el programa Matlab. Para esta simulación se ha tomado en cuenta las siguientes consideraciones:

Cond. Iniciales	Polos Usados
$x_1 = 0, x_2 = -\pi/2, x_3 = 0$	base: $-8 \wedge -8$, brazo1: $-8 \wedge -8$, brazo2: $-8 \wedge -8$

Un sistema no lineal multivariable de orden n , y con m entradas y m salidas, es completamente linealizable y desacoplado si su grado relativo total es igual a r :

$$r = r_1 + r_2 + \dots + r_m \quad (5.7)$$

Es igual al orden del sistema, esto es: $r=n$, entonces se aplica el método de linealización exacta. Aplicando los conceptos de geometría diferencial determinamos el grado relativo total de nuestro sistema.

5.5.1. Determinación del Grado Relativo Total, r

Derivando y aplicando los conceptos de geometría diferencial tenemos:

Primera Salida: $y_1(x)$

$$y_1 = h_1(x) = x_1 \quad (5.8)$$

$$\dot{y}_1 = L_{\mathbf{F}}h_1(x) = x_4 \quad (5.9)$$

$$\ddot{y}_1 = L_{\mathbf{F}}^2h_1(x) + L_{\mathbf{F}}L_{\mathbf{G}}h_1(x), \quad L_{\mathbf{F}}L_{\mathbf{G}}h_1(x) \neq 0 \quad (5.10)$$

Por tanto, el grado relativo para $y_1(x)$ es igual a 2.

Segunda Salida: $y_2(x)$

$$y_2 = h_2(x) = x_2 \quad (5.11)$$

$$\dot{y}_2 = L_{\mathbf{F}}h_2(x) = x_5 \quad (5.12)$$

$$\ddot{y}_2 = L_{\mathbf{F}}^2h_2(x) + L_{\mathbf{F}}L_{\mathbf{G}}h_2(x), \quad L_{\mathbf{F}}L_{\mathbf{G}}h_2(x) \neq 0 \quad (5.13)$$

Por tanto, el grado relativo para $y_2(x)$ es igual a 2.

Tercera Salida: $y_3(x)$

$$y_3 = h_3(x) = x_3 \quad (5.14)$$

$$\dot{y}_3 = L_{\mathbf{F}}h_3(x) = x_6 \quad (5.15)$$

$$\ddot{y}_3 = L_{\mathbf{F}}^2h_3(x) + L_{\mathbf{F}}L_{\mathbf{G}}h_3(x), \quad L_{\mathbf{F}}L_{\mathbf{G}}h_3(x) \neq 0 \quad (5.16)$$

Por tanto, el grado relativo para $y_3(x)$ es igual a 2.

Finalmente el grado relativo total es $r = r_1 + r_2 + r_3 = 6 = n$, n grado del sistema, ver ecuación (2.64).

5.5.2. Forma Normal

De los conceptos anteriormente mencionados se obtiene, la transformación de coordenadas $z = \phi(x)$, y se muestra a continuación:

$$z = \phi(x) = \begin{bmatrix} h_1(x) \\ L_{\mathbf{F}}h_1(x) \\ h_2(x) \\ L_{\mathbf{F}}h_2(x) \\ h_3(x) \\ L_{\mathbf{F}}h_3(x) \end{bmatrix} = \begin{bmatrix} x_1 & x_4 & x_2 & x_5 & x_3 & x_6 \end{bmatrix}^T = \begin{bmatrix} z_1 & z_2 & z_3 & z_4 & z_5 & z_6 \end{bmatrix}^T \quad (5.17)$$

Finalmente la nueva ecuación de estados en la variable z , viene dado por:

$$\dot{z} = \begin{bmatrix} \dot{z}_1 & \dot{z}_4 & \dot{z}_2 & \dot{z}_5 & \dot{z}_3 & \dot{z}_6 \end{bmatrix}^T = \begin{bmatrix} z_2 & v_1 & z_4 & v_2 & z_6 & v_3 \end{bmatrix}^T \quad (5.18)$$

5.5.3. Ley de Control

Para la nueva ecuación de estados (5.18) y aplicando la ecuación (3.75), obtenemos las entradas lineales v_1 y v_2 y se expresan de la siguiente forma:

$$v_1 = \dot{x}_4 = \ddot{P}_1 + K_1\dot{\bar{y}}_1 + K_2\bar{y}_2 \quad (5.19)$$

$$v_2 = \dot{x}_5 = \ddot{P}_2 + K_3\dot{\bar{y}}_2 + K_4\bar{y}_2 \quad (5.20)$$

$$v_3 = \dot{x}_6 = \ddot{P}_3 + K_5\dot{\bar{y}}_3 + K_6\bar{y}_3 \quad (5.21)$$

Donde,

P_i : Son las trayectorias deseadas.

\bar{y}_i : Error de la salida i

Así mismo,

$$\bar{y}_1 = P_1 - y_1, \quad x_1 = y_1 \quad (5.22)$$

$$\bar{y}_2 = P_2 - y_2, \quad x_2 = y_2 \quad (5.23)$$

$$\bar{y}_3 = P_3 - y_3, \quad x_3 = y_3 \quad (5.24)$$

De las anteriores ecuaciones se obtiene la ley de control correspondiente para cada motor,

$$v_1 = \ddot{P}_1 + K_1\dot{P}_1 + K_2P_1 - K_1z_2 - K_2z_1 \quad (5.25)$$

$$v_2 = \ddot{P}_2 + K_3\dot{P}_2 + K_4P_2 - K_3z_4 - K_4z_3 \quad (5.26)$$

$$v_3 = \ddot{P}_3 + K_5\dot{P}_3 + K_6P_3 - K_5z_6 - K_6z_5 \quad (5.27)$$

5.5.4. Respuesta a una Trayectoria Constante

El tiempo de simulación utilizado es de 10 seg. La Figura 5.9 muestra las salidas controladas de la posición angular de la base, brazo y antebrazo respectivamente. Luego de un tiempo de 0.9 seg. la base alcanza a la trayectoria deseada,

mientras que para el brazo y antebrazo lo alcanzan en 0.6 seg. Los valores de consigna para la posición angular de la base es de $-\pi/4$ rad., para el brazo es de $\pi/2$ rad. y el antebrazo es de $\pi/4$ rad.

La Figura 5.10 muestra la respuesta del error de posición angular de la base, brazo y antebrazo del robot, en todos los casos observamos que el error tiende a cero después de 1 seg.

La Figura 5.11 muestra la respuesta del error de velocidad de la base, brazo y antebrazo del robot, en todos los casos observamos que el error tiende a cero después de 1 seg, esto nos indica que el sistema robótico está siendo controlado.

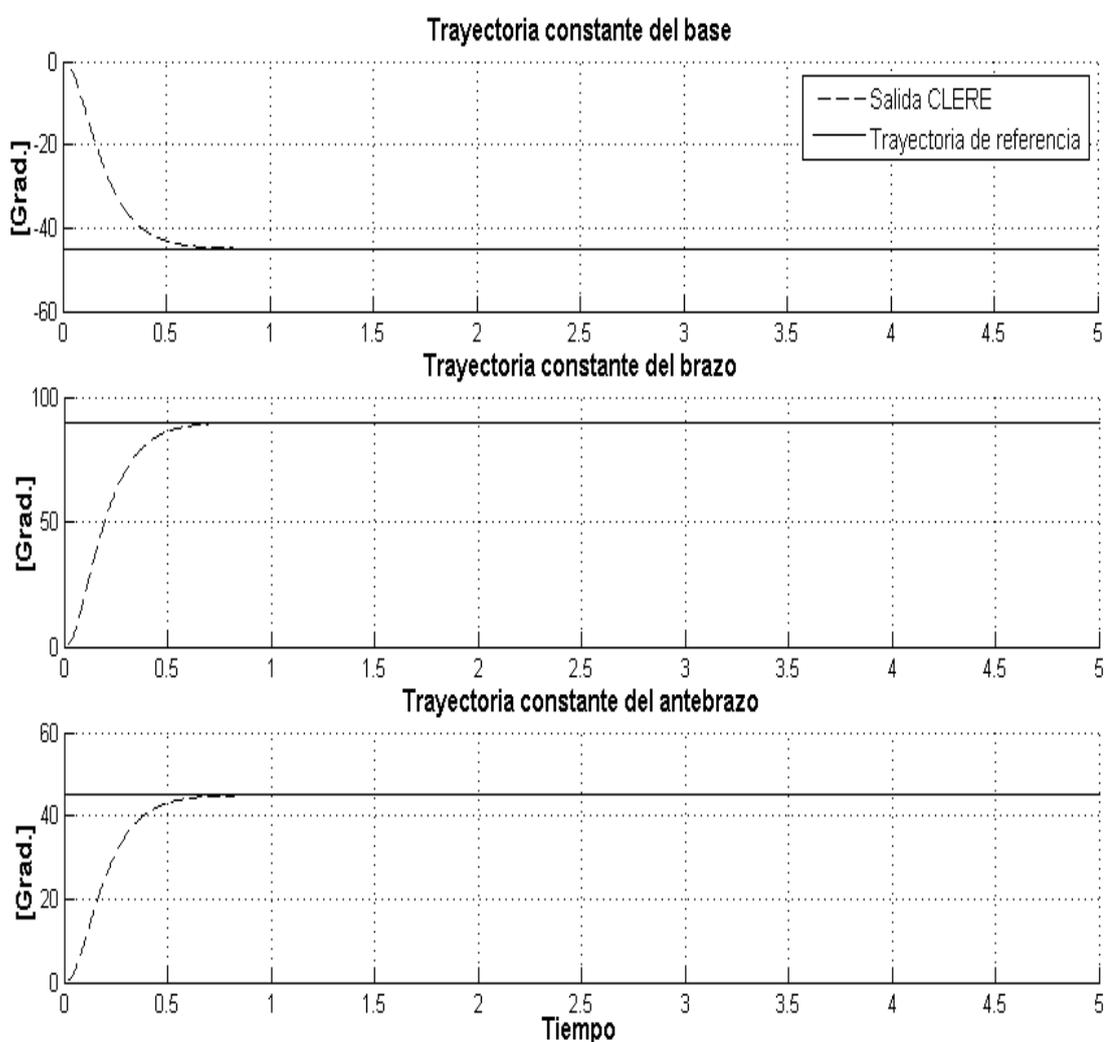


Figura 5.9: Salidas controladas para la posición angular de la base, brazo y antebrazo, respectivamente.

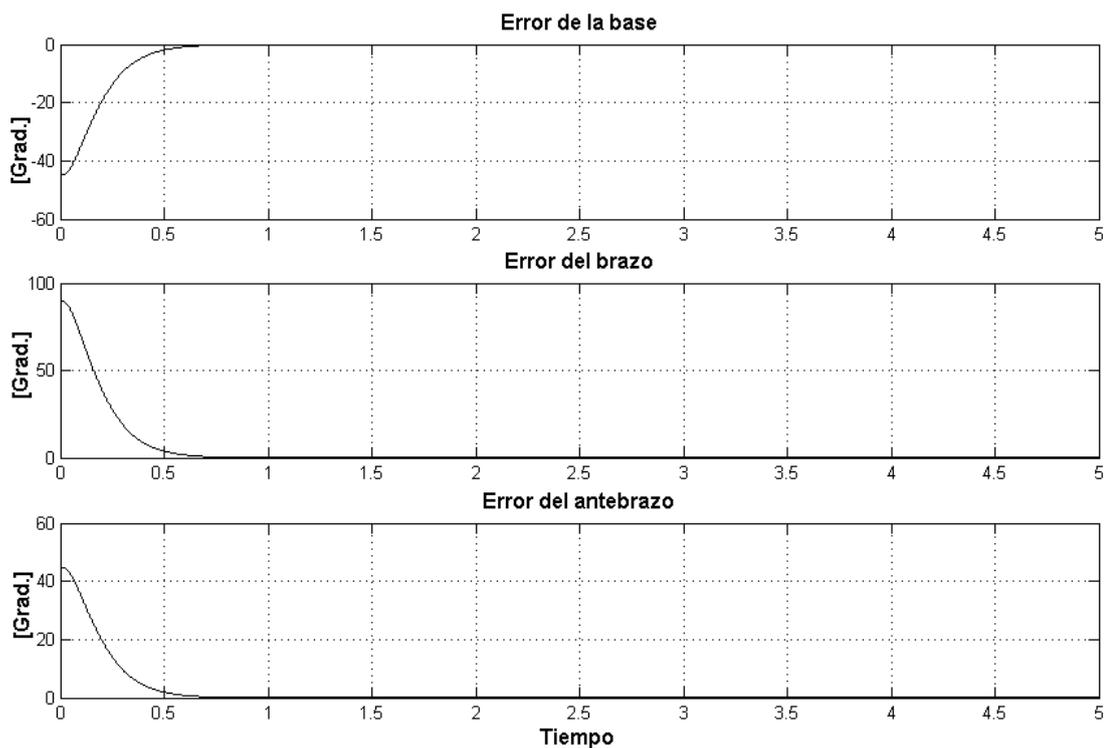


Figura 5.10: Error de posición angular de la base, brazo y antebrazo, respectivamente.

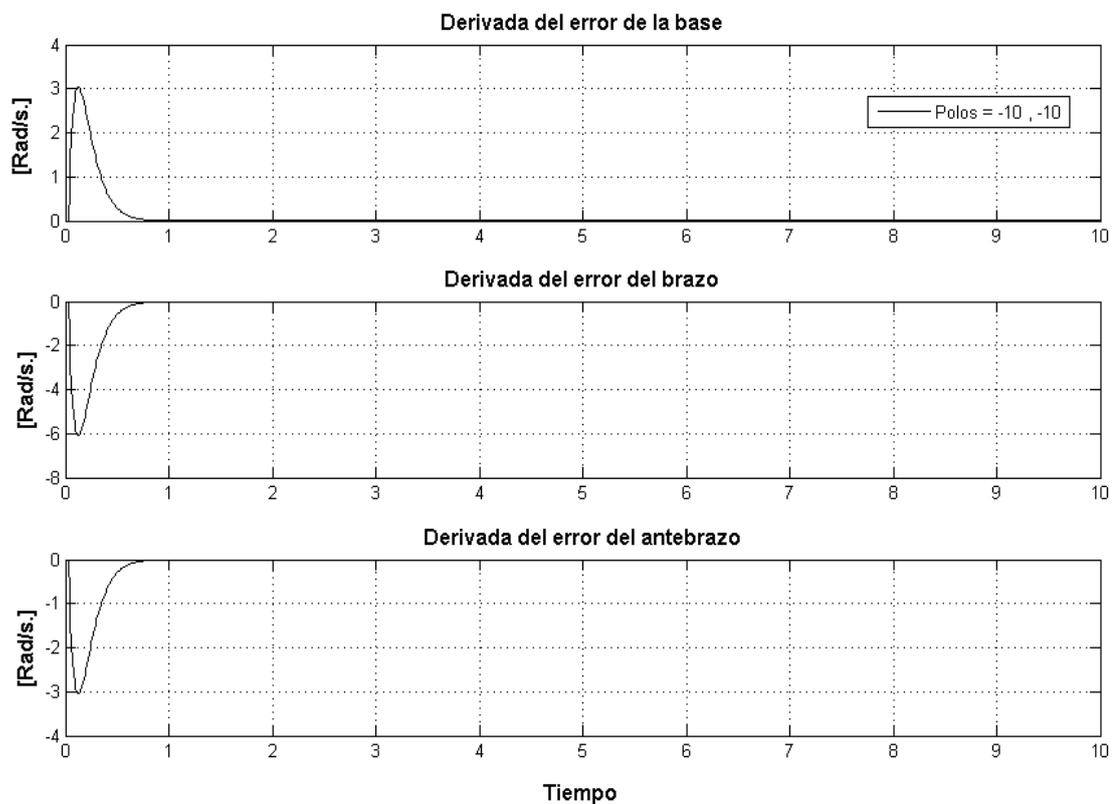


Figura 5.11: Derivada del error de posición angular de la base, brazo y antebrazo, respectivamente.

Y finalmente la Figura 5.12 muestra la ley de control que al inicio presenta una amplitud elevada pero que es limitada por el algoritmo de control. La ley de control es susceptible a los cambios en los polos, ver Figura ??, a medida que aumentamos los polos la ley de control presenta picos más elevados pero se establece a cero en menor tiempo.

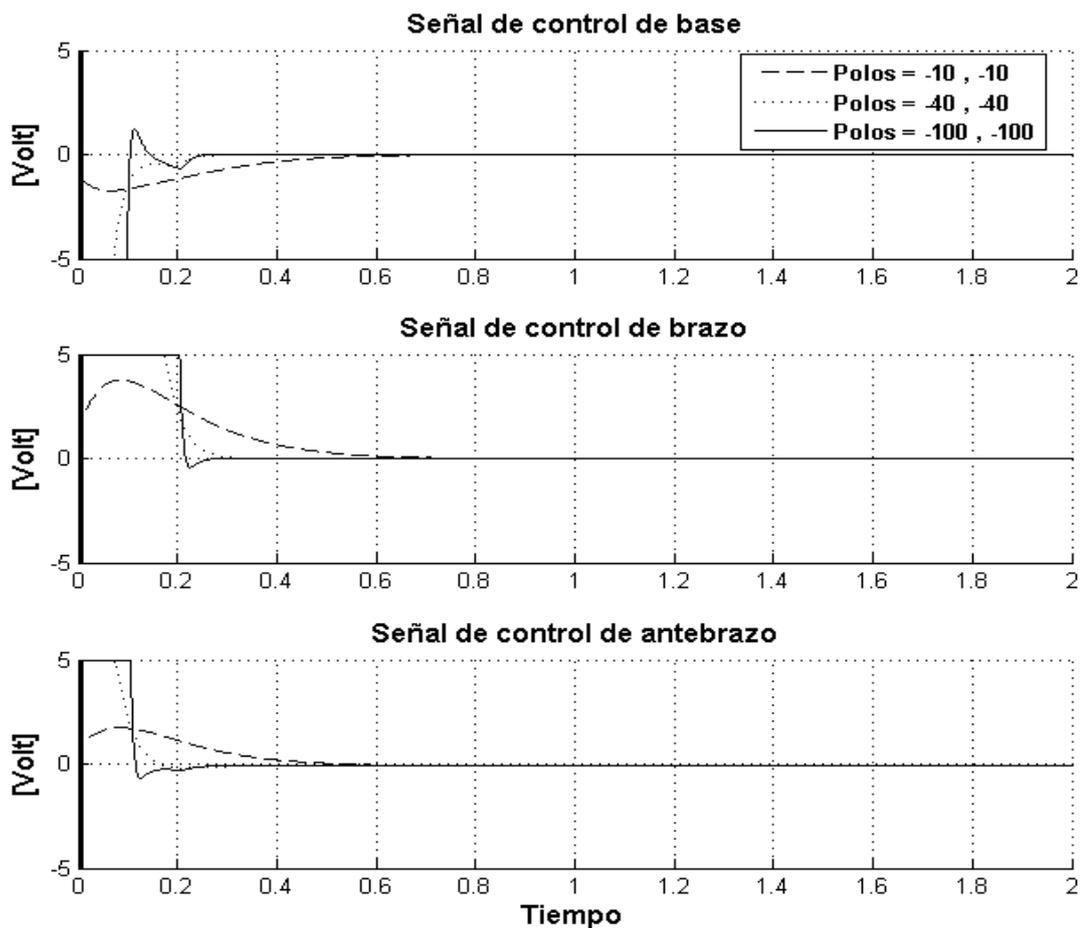


Figura 5.12: Señales de control para los motores que manejan a la base, brazo y antebrazo, respectivamente.

5.5.5. Respuesta a una Trayectoria Senoidal

Consideremos una trayectoria senoidal con las siguientes condiciones iniciales iguales a los casos anteriores.

La trayectoria para la base viene es $x_{d1} = \pi/4 \cos(0,5\pi kT) rad.$, para el brazo es de $x_{d2} = \pi/3 \cos(0,5\pi kT) rad.$ y finalmente para el antebrazo es de $x_{d3} = \pi/2 \cos(0,5\pi kT) rad.$

El tiempo de simulación es de 10 seg. La Figura 5.13 muestra las salidas controladas de la posición angular de la base, brazo y antebrazo para diversos valores de polos. Se concluye que a medida que se aumenta el valor de los polos la salida se establece en un menor tiempo que para polos de menor valor, pero también presentan una mayor amplitud que luego de aproximadamente 0.8 seg., para todos los casos, alcanzan a la trayectoria de referencia.

La Figura 5.14 muestra la respuesta del error de posición angular de la base, brazo y antebrazo del robot, se observa que el error tiende a cero después de aproximadamente 0.8 seg. para el brazo, mientras que para la base y el antebrazo se observa un error muy pequeño y periódico.

La Figura 5.11 muestra la respuesta del error de velocidad de la base, brazo y antebrazo del robot, en todos los casos observamos que el error tiende a cero después de 0.8 seg.

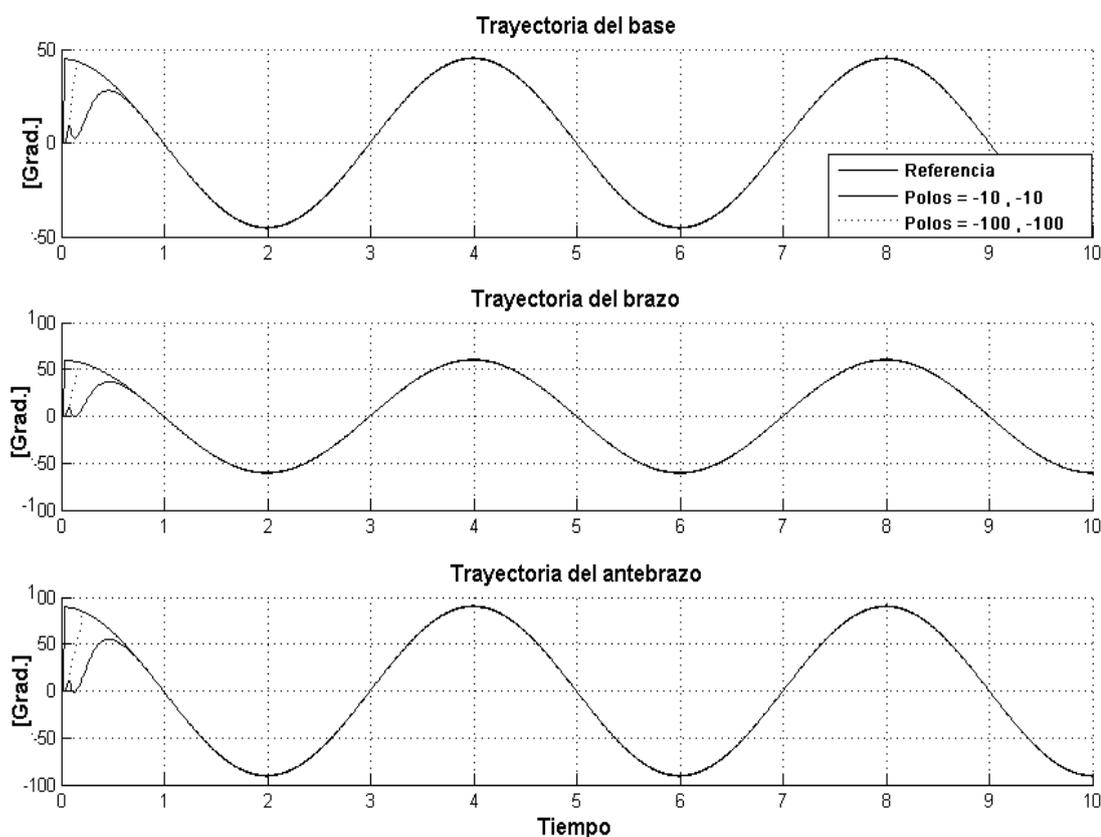


Figura 5.13: Salidas controladas para la posición angular de la base, brazo y antebrazo, respectivamente.

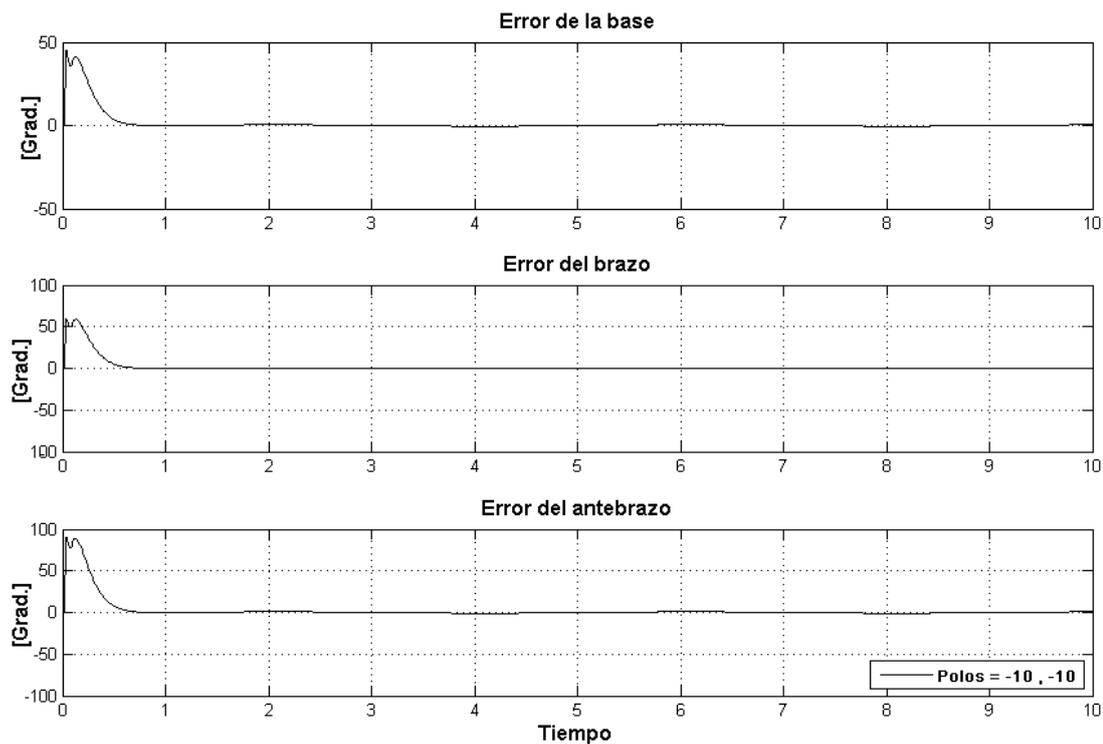


Figura 5.14: Error de posición angular de la base, brazo y antebrazo, respectivamente.

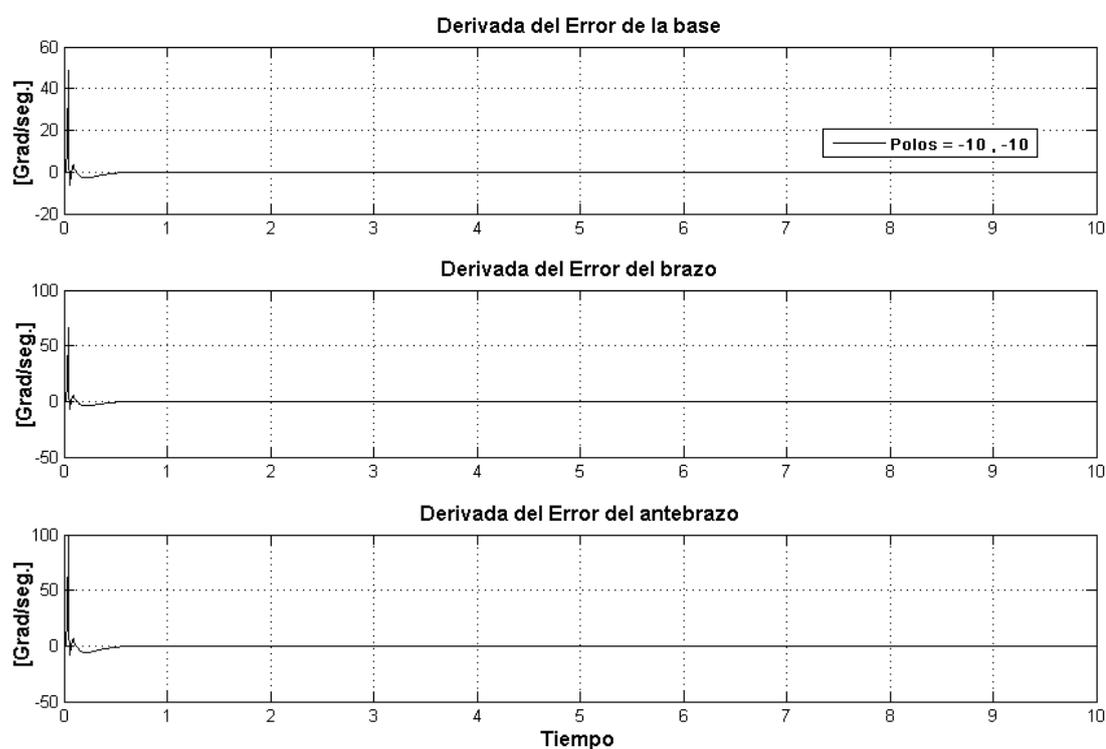


Figura 5.15: Derivada del error de posición angular de la base, brazo y antebrazo, respectivamente.

Y finalmente la Figura 5.16 muestra la ley de control que al inicio presenta

una amplitud elevada pero que es limitada por el algoritmo de control. La ley de control es susceptible a los cambios en los polos, ver Figura 5.16, a medida que aumentamos los polos la ley de control permanece más tiempo en el pico máximo, en este caso es de ± 5 volt.

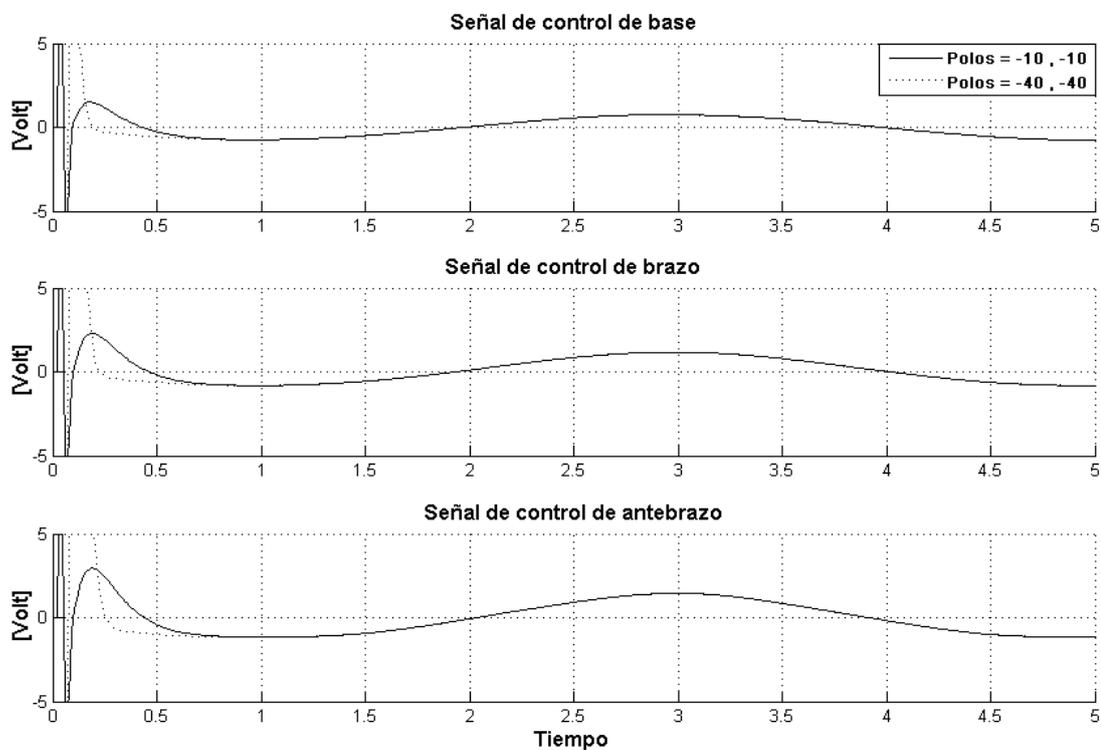


Figura 5.16: Señales de control para los motores que manejan a la base, brazo y antebrazo, respectivamente.

5.6. Control Predictivo Generalizado No Lineal Aplicado al Manipulador Robótico de 3GDL

De acuerdo a la asignación de variables de estado visto en la sección 2.11 el modelo en espacio estado viene dado por la ecuación (2.64) y es:

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \\ \dot{x}_5 \\ \dot{x}_6 \end{bmatrix} = \begin{bmatrix} F_1(x) \\ F_2(x) \\ F_3(x) \\ F_4(x) \\ F_5(x) \\ F_6(x) \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ G_{41}(x) & G_{42}(x) & G_{43}(x) \\ G_{51}(x) & G_{52}(x) & G_{53}(x) \\ G_{61}(x) & G_{62}(x) & G_{63}(x) \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \end{bmatrix} \quad (5.28)$$

con salidas,

$$y = \mathbf{h}(x) = \begin{bmatrix} x_1 & x_2 & x_3 \end{bmatrix}^T \quad (5.29)$$

fácilmente se pueden identificar las siguientes funciones:

$$F_1(x) = x_4 \quad (5.30)$$

$$F_2(x) = x_5 \quad (5.31)$$

$$F_3(x) = x_6 \quad (5.32)$$

Luego derivando las salidas, se tiene:

▷ $y^{[1]}(x)$

$$\dot{y} = \frac{\partial \mathbf{h}(x)}{\partial x} \mathbf{F} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_4 \\ x_5 \\ x_6 \\ F_4 \\ F_5 \\ F_6 \end{bmatrix} \quad (5.33)$$

$$\dot{y} = \begin{bmatrix} x_4 \\ x_5 \\ x_6 \end{bmatrix} = L_{\mathbf{F}}h(x) \quad (5.34)$$

▷ $y^{[2]}(x)$

$$\ddot{y} = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \left\{ \begin{bmatrix} x_4 \\ x_5 \\ x_6 \\ F_4 \\ F_5 \\ F_6 \end{bmatrix} + \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ G_{41}(x) & G_{42}(x) & G_{43}(x) \\ G_{51}(x) & G_{52}(x) & G_{53}(x) \\ G_{61}(x) & G_{62}(x) & G_{63}(x) \end{bmatrix} \mathbf{u} \right\} \quad (5.35)$$

$$\ddot{y} = \begin{bmatrix} F_4 \\ F_5 \\ F_6 \end{bmatrix} + \begin{bmatrix} G_{41}(x) & G_{42}(x) & G_{43}(x) \\ G_{51}(x) & G_{52}(x) & G_{53}(x) \\ G_{61}(x) & G_{62}(x) & G_{63}(x) \end{bmatrix} \mathbf{u} \quad (5.36)$$

$$\ddot{y} = L_{\mathbf{F}}^2 h(x) + L_{\mathbf{G}} L_{\mathbf{F}} h(x) \mathbf{u} \quad (5.37)$$

de aquí obtenemos el grado relativo, ¡no grado relativo total!, $\rho = 2$, pues $L_{\mathbf{G}} L_{\mathbf{F}} h(x) \neq 0$ y considerando $r=2$, tenemos para la predicción,

$$\hat{y}(t+\tau) \approx \hat{y}^{[0]}(t) + \tau \hat{y}^{[1]}(t) + \frac{\tau^2}{2!} \hat{y}^{[2]}(t) + \frac{\tau^3}{3!} \hat{y}^{[3]}(t) + \frac{\tau^4}{4!} \hat{y}^{[4]}(t), \quad 0 \leq \tau \leq T_2 \quad (5.38)$$

de la ecuación (4.41), para el sistema propuesto,

$$H(\hat{\mathbf{u}}) = - [\bar{\mathcal{T}}_{22}^{-1} \bar{\mathcal{T}}_{22}^T \quad [\mathcal{L}]_{6 \times 6}] M(x(t)) \quad (5.39)$$

reemplazando los valores de ρ y r del sistema en las ecuaciones (4.51) y (4.52), se tiene:

$$\bar{\mathcal{T}}_{rr} = \bar{\mathcal{T}}_{22} = \begin{bmatrix} \bar{\mathcal{T}}_{33} & \bar{\mathcal{T}}_{34} & \bar{\mathcal{T}}_{35} \\ \bar{\mathcal{T}}_{43} & \bar{\mathcal{T}}_{44} & \bar{\mathcal{T}}_{45} \\ \bar{\mathcal{T}}_{53} & \bar{\mathcal{T}}_{54} & \bar{\mathcal{T}}_{55} \end{bmatrix}_{9 \times 9} \quad (5.40)$$

$$\bar{\mathcal{T}}_{\rho r} = \bar{\mathcal{T}}_{22} = \begin{bmatrix} \bar{\mathcal{T}}_{13} & \bar{\mathcal{T}}_{14} & \bar{\mathcal{T}}_{15} \\ \bar{\mathcal{T}}_{23} & \bar{\mathcal{T}}_{24} & \bar{\mathcal{T}}_{25} \end{bmatrix}_{6 \times 9} \quad (5.41)$$

además se sabe que:

$$\bar{T}_{(i,j)} = \frac{\bar{T}^{i+j-1}}{(i-1)!(j-1)!(i+j-1)}, \quad i, j = 1, \dots, \rho + r + 1 \quad (5.42)$$

$$\bar{T} = \text{diag} \{T_2, T_2, \dots, T_2\} \in R^{3 \times 3}$$

sabiendo que \mathcal{K} son las m primeras filas de la matriz $\bar{\mathcal{T}}_{rr}^{-1} \bar{\mathcal{T}}_{\rho r}^T$ y teniendo presente la ecuación (5.42), se obtiene:

$$\bar{\mathcal{T}}_{rr}^{-1} \bar{\mathcal{T}}_{\rho r}^T = \begin{bmatrix} \frac{42}{T_2^2} & 0 & 0 & \frac{21}{2T_2} & 0 & 0 & | & \dots \\ 0 & \frac{42}{T_2^2} & 0 & 0 & \frac{21}{2T_2} & 0 & | & \dots \\ 0 & 0 & \frac{42}{T_2^2} & 0 & 0 & \frac{21}{2T_2} & | & \dots \\ - & - & - & - & - & - & + & - \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & | & \ddots \end{bmatrix} \quad (5.43)$$

por tanto,

$$\mathcal{K} = \begin{bmatrix} \frac{42}{T_2^2} & 0 & 0 & \frac{21}{2T_2} & 0 & 0 \\ 0 & \frac{42}{T_2^2} & 0 & 0 & \frac{21}{2T_2} & 0 \\ 0 & 0 & \frac{42}{T_2^2} & 0 & 0 & \frac{21}{2T_2} \end{bmatrix}_{3 \times 6} \quad (5.44)$$

con las anteriores consideraciones la ecuación (5.39) puede reescribirse, como (ver ecuación(4.42)):

$$\begin{bmatrix} L_{\mathbf{G}}L_{\mathbf{F}}\mathbf{h}(x)\hat{\mathbf{u}}(t) \\ \text{-----} \\ \vdots \end{bmatrix} = - \begin{bmatrix} | & 1 & 0 & 0 & \cdots \\ \mathcal{K} & | & 0 & 1 & 0 & \cdots \\ | & 0 & 0 & 1 & \cdots \\ - & + & - & - & - & \cdots \\ \vdots & | & \vdots & \vdots & \vdots & \ddots \end{bmatrix} \begin{bmatrix} L_{\mathbf{F}}^0\mathbf{h}(x) - w^{[0]}(t) \\ L_{\mathbf{F}}^1\mathbf{h}(x) - w^{[1]}(t) \\ L_{\mathbf{F}}^2\mathbf{h}(x) - w^{[2]}(t) \end{bmatrix} \quad (5.45)$$

Finalmente la señal de control viene dada por:

$$\hat{\mathbf{u}}(t) = \mathbf{u} = [L_{\mathbf{G}}L_{\mathbf{F}}\mathbf{h}(x)]^{-1} \left\{ \begin{array}{l} 42/T_2^2 \begin{bmatrix} x_1(t) - w_1(t) \\ x_2(t) - w_2(t) \\ x_3(t) - w_3(t) \end{bmatrix} + 21/(2T_2) \begin{bmatrix} x_4(t) - w_1^{[1]}(t) \\ x_5(t) - w_2^{[1]}(t) \\ x_6(t) - w_3^{[1]}(t) \end{bmatrix} \\ + \begin{bmatrix} F_4 - w_1^{[2]}(t) \\ F_5 - w_2^{[2]}(t) \\ F_6 - w_3^{[2]}(t) \end{bmatrix} \end{array} \right\} \quad (5.46)$$

La ley de control derivada en este punto es obtenida sin la necesidad de calcular los valores óptimos de la ley de control en cada instante de muestreo, esto resulta ventajoso para sistemas de dinámica rápida (robots). Además, como se aprecia la ley de control depende del factor T_2 , horizonte de predicción, el cual variaremos su valor en las simulaciones para observar el comportamiento de la salida y ley de control respecto a este cambio.

5.6.1. Respuesta a una Trayectoria Constante

El tiempo de simulación utilizado es de 10 seg. Los valores de consigna para la posición angular de la base es de $-\pi/4$ rad., para el brazo es de $\pi/2$ rad. y el antebrazo es de $\pi/4$ rad.

La Figura 5.17 muestra las salidas controladas de la posición angular de la base,

brazo y antebrazo para diversos valores del horizonte de predicción, esto nos permite concluir que para el valor $T_2 = 0,10$ se obtiene un respuesta con un tiempo de establecimiento muy corto comparado a los otros valores que adopta el horizonte de predicción, pero también se obtiene una mayor pico en el transitorio. De acuerdo a la figura, con $T_2 = 0,5$ la referencia es alcanzada después de 0.9 seg. La Figura 5.18 muestra la respuesta del error de posición angular de la base, brazo y antebrazo del robot, en todos los casos observamos que el error tiende a cero después de 1 seg.

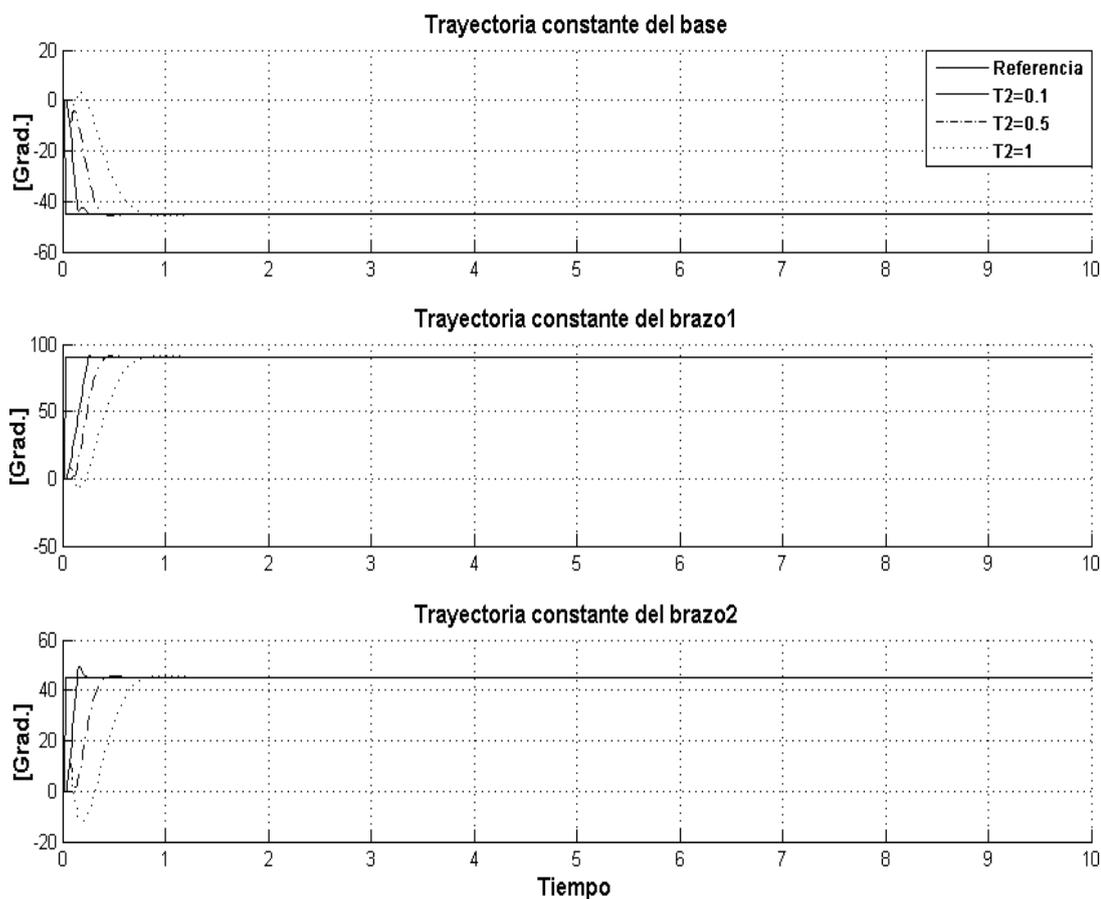


Figura 5.17: Salidas controladas para la posición angular de la base, brazo y antebrazo, respectivamente.

Finalmente la Figura 5.19 muestra la señal de control, al inicio presenta una amplitud elevada pero que es limitada por el algoritmo de control. Ésta es susceptible a los cambios en el horizonte de control, ver Figura 5.19, esto es, a medida que aumentamos el valor de T_2 la ley de control se establece a cero en menor tiempo.

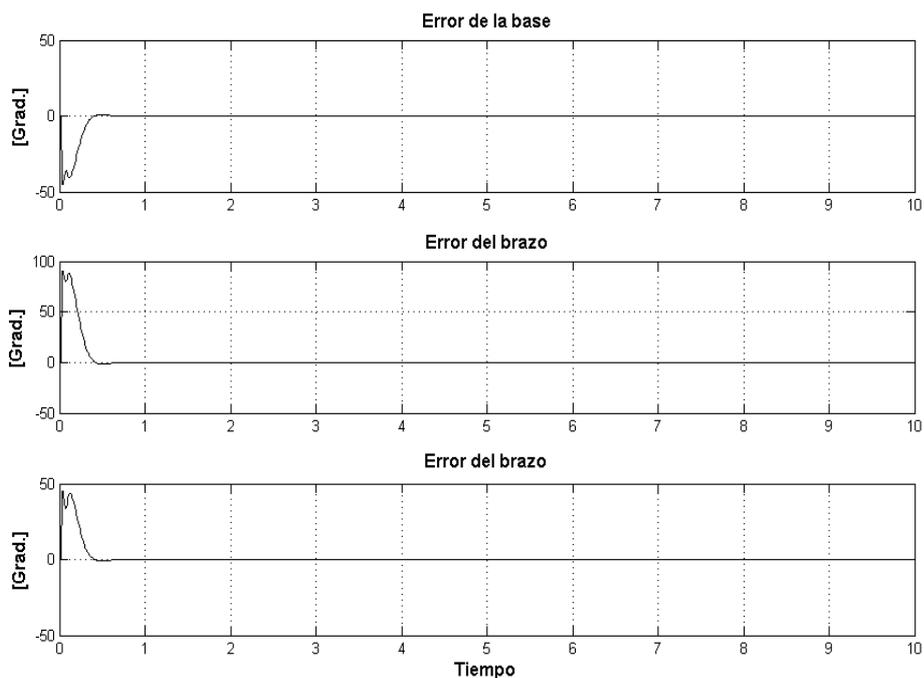


Figura 5.18: Error de posición angular de la base, brazo y antebrazo, respectivamente.

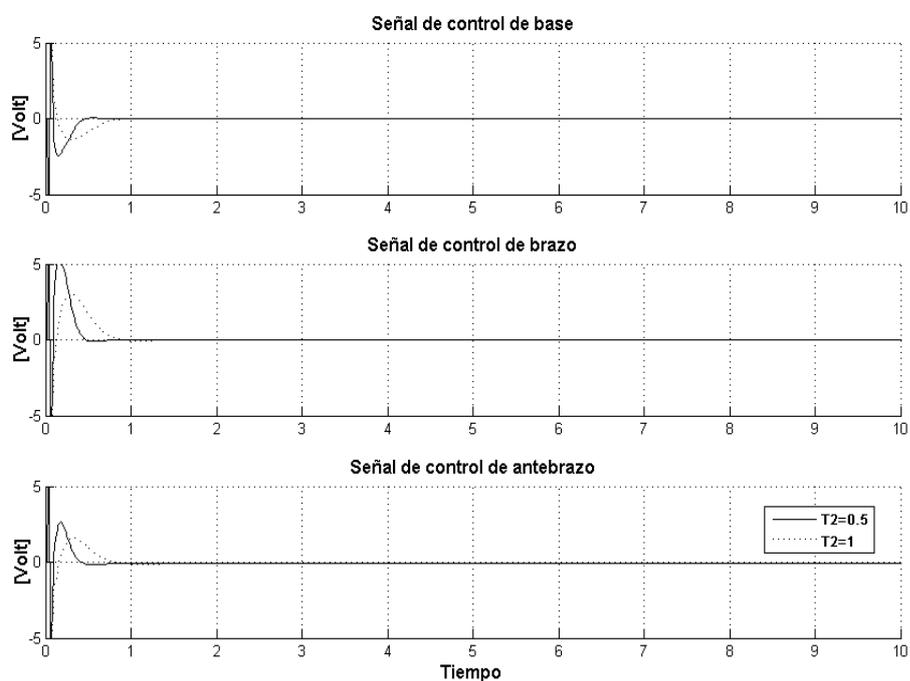


Figura 5.19: Señales de control para los motores que manejan a la base, brazo y antebrazo, respectivamente.

5.6.2. Respuesta a una Trayectoria Senoidal

Consideremos una trayectoria senoidal con las siguientes condiciones iniciales iguales a cero. La trayectoria para la base viene es $x_{d1} = \pi/4 \cos(0,5\pi kT)rad.$, para el brazo es de $x_{d2} = \pi/3 \cos(0,5\pi kT)rad.$ y para el antebrazo es de $x_{d3} = \pi/2 \cos(0,5\pi kT)rad.$ El tiempo de simulación es de 10 seg. La Figura 5.20 muestra las salidas controladas de la posición angular de la base, brazo y antebrazo para diversos valores de horizonte de predicción $T_2 = 0,1, 0,5, 1$. Se concluye que a medida que se aumenta el valor de T_2 la salida se establece en un mayor tiempo que para los de menor valor, y además para $T_2 = 0,1$, se tiene una mayor amplitud que luego de aproximadamente 0.5 seg. alcanza a la trayectoria de referencia. La Figura 5.21 muestra la respuesta del error de posición angular de la base,

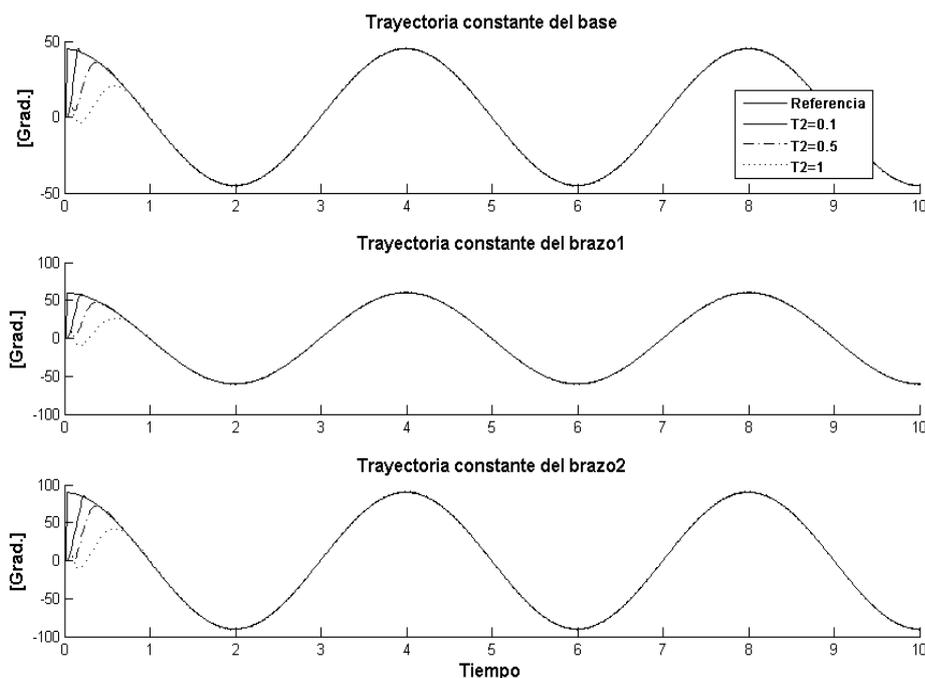


Figura 5.20: Salidas controladas para la posición angular de la base, brazo y antebrazo, respectivamente.

brazo y antebrazo del robot, se observa que el error tiende a cero después de aproximadamente 0.8 seg. todos éstos alcanzan un error cero.

Y en la Figura 5.25 se muestra la ley de control que al inicio presenta una amplitud elevada pero que es limitada por el algoritmo de control. La ley de

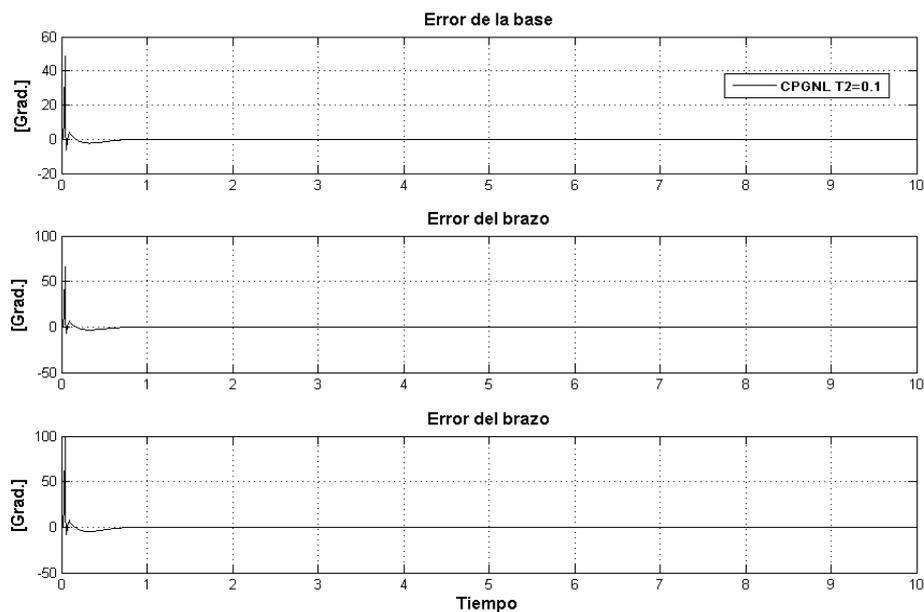


Figura 5.21: Error de posición angular de la base, brazo y antebrazo, respectivamente.

control es susceptible a los cambios de T_2 , es decir, a medida que lo aumentamos la ley de control permanece menos tiempo en el pico máximo, es decir que entrega más energía a los motores para alcanzar a la referencia en un tiempo más corto.

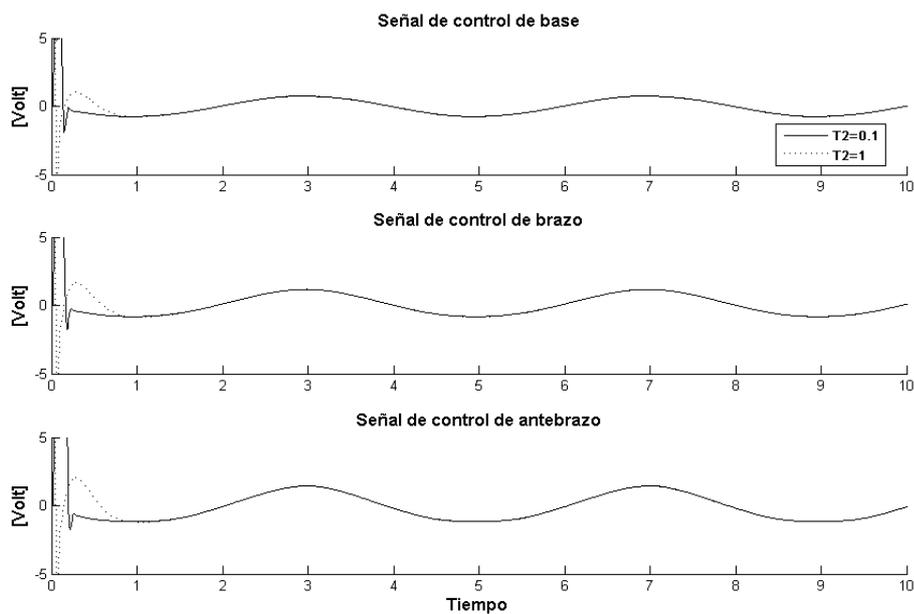


Figura 5.22: Señales de control para los motores que manejan a la base, brazo y antebrazo, respectivamente.

Hasta ese punto hemos visto las simulación del control predictivo generalizado (CPGNL) y el control por linealización exacta por realimentación de estados (CLERE) de manera independiente sin observar una comparación de éstos.

A continuación mostraremos las salidas controladas usando CLERE y CPGNL superpuestas para observar las ventajas entre éstas estrategias avanzadas de control. Para una trayectoria de referencia tal como se muestra en la Figura 5.23, observamos que el control predictivo alcanza a la referencia escalonada en menor tiempo en todos los tramos, pero esto lo consigue con un pequeño sobreimpulso. Además, en la Figura 5.24 se muestra la ley de control para cada una de las

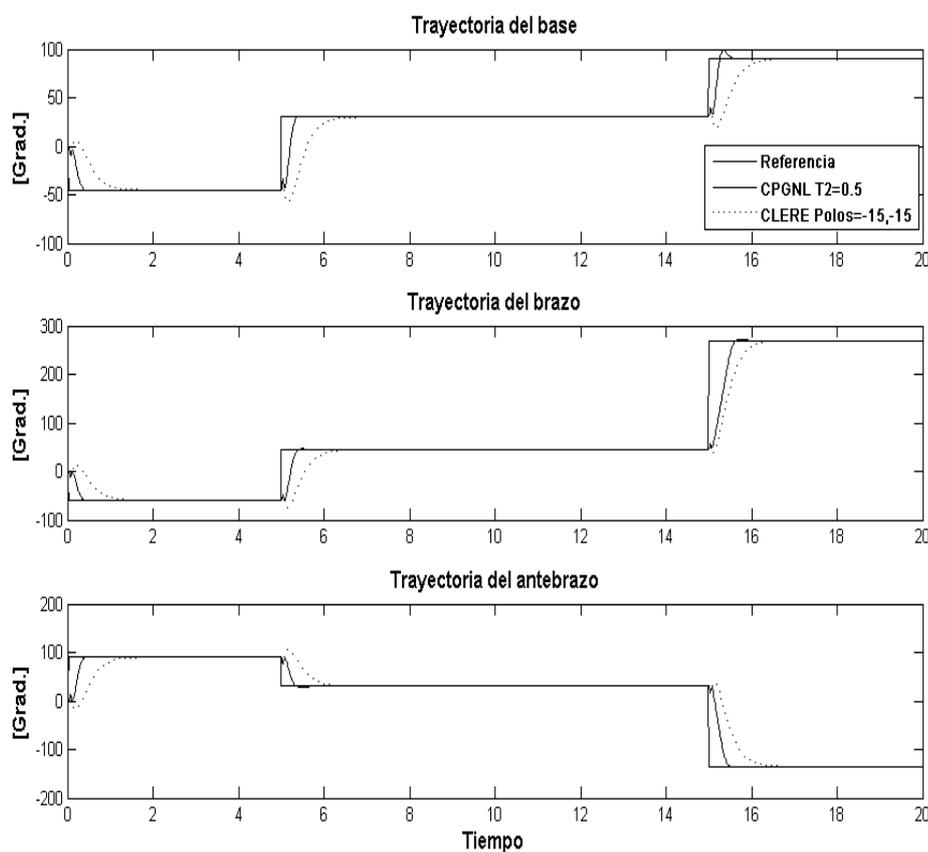


Figura 5.23: Salidas controladas para la posición angular de la base, brazo y antebrazo, respectivamente.

estrategias de control, se observa que la ley de control para el CPGNL alcanza al pico máximo de voltaje por más tiempo que para el CLERE, esto permite alcanzar con mayor rapidez a la referencia a costa de aumentar el sobreimpulso.

Las leyes de control de ambas estrategias se establecen en valores muy cercanos a cero.

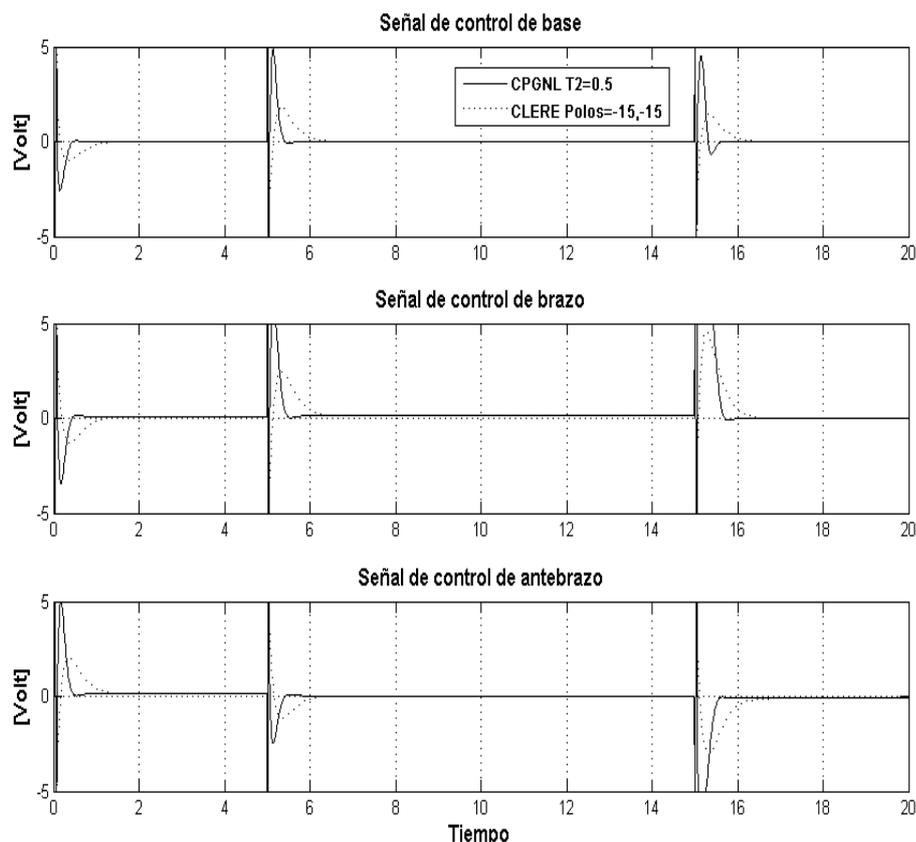


Figura 5.24: Señales de control para los motores que manejan a la base, brazo y antebrazo, respectivamente.

5.7. Animación con la Interfaz Gráfica GUI

En esta sección se presenta el diseño de la interfaz gráfica para el usuario (GUI: Graphic User Interface), esta está formada por un par de ventanas en la que uno accede para modificar ciertos valores preestablecidos para luego ejecutar el programa, y visualizar el resultado en un entorno 3D dentro del panel de control principal.

Esta interfaz permite seleccionar cualquiera de las dos estrategias tratadas en esta tesis; la primera es el control por linealización exacta por realimentación de estado (CLERE) y la segunda es el control predictivo generalizado no lineal (CPGNL)

ambas estrategias realizan la tarea de controlar al manipulador robótico de 3 GDL. El propósito del uso de la Interfaz del Usuario Gráfica (GUI), es ayudar a visualizar el posible movimiento real del robot en el espacio XYZ controlado por cualquiera de las dos estrategias de control estudiadas, así como también la posibilidad de modificar su trayectoria de referencia. Para conseguir esto el programa consta de ... archivos desarrollados en Matlab 2007, y son los siguientes:

1. El programa *Imagen.m*, es el principal que alberga a los demás. Cuando es ejecutado, nos ofrece un panel en la cual se presenta al manipulador robótico en su posición inicial, así como también controles Slider que permiten controlar cada articulación independiente bajo el espacio XYZ. Además, se presenta una área para el ploteo de los resultados, y finalmente una área donde se encuentran los controles que permiten setear la trayectoria de referencia y el tipo de controlador a usar.
2. El programa *Robot3DANI.m*, es una función permite la animación de los objetos en el espacio XYZ. Esta función setea constantemente las propiedades gráficas de los objetos para conseguir el efecto de la animación en 3D.
3. El programa *CPGNL.m*, ejecuta la rutina del control predictivo y el resultado es enviado como argumento a la función *Robot3DANI.m* para su animación en el espacio XYZ.
4. El programa *CLERE.m*, al igual que el anterior ejecuta la rutina del control por linealización exacta por realimentación de estados.

5.7.1. Uso del Panel de Control

En la Figura 5.25, apreciamos un conjunto de controles tipo slider en la parte central izquierda del panel principal de control. Cada control slider permite controlar la posición angular de cada articulación de manera independiente, finalmente si deseamos posicionar el manipulador en su posición inicial entonces procedemos a hacer click sobre el boton **INICIAL** del panel llamado **PANEL**

CONTROL.

Dentro de **PANEL CONTROL** también existen otros controles como **CON-**

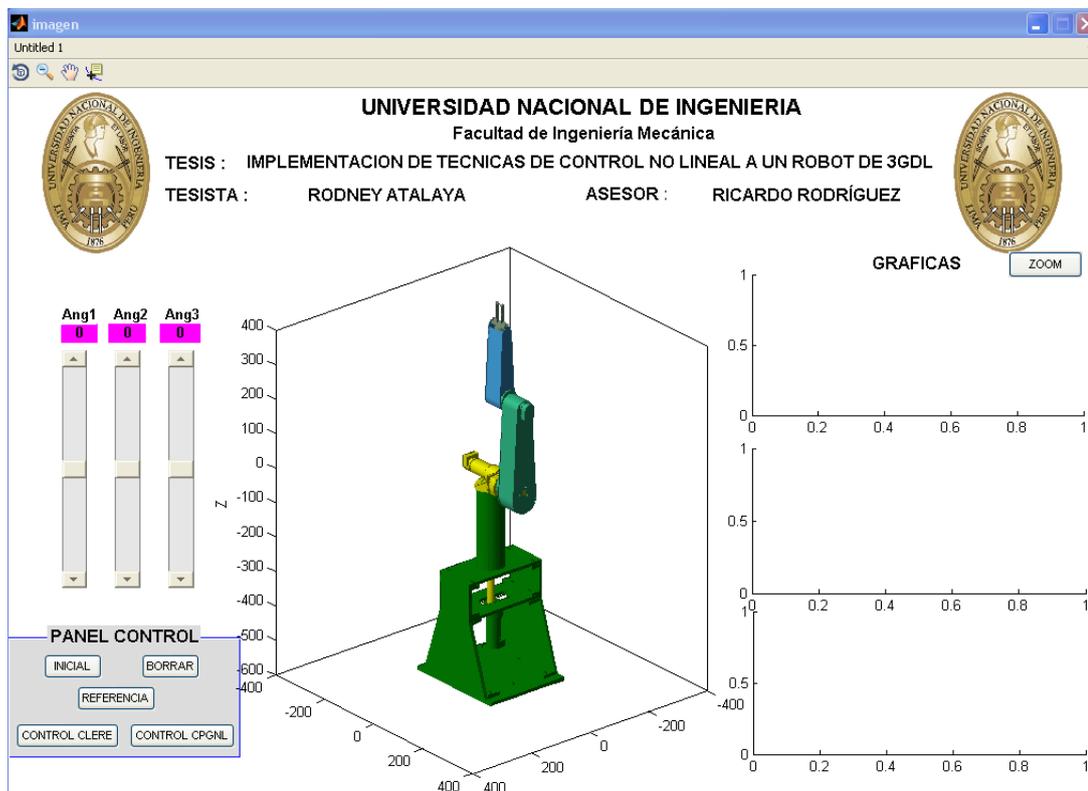
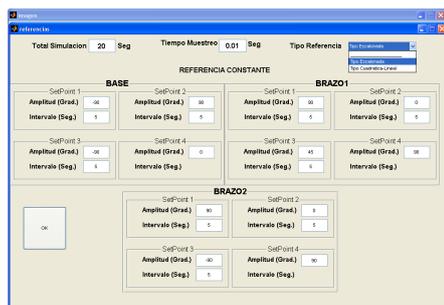


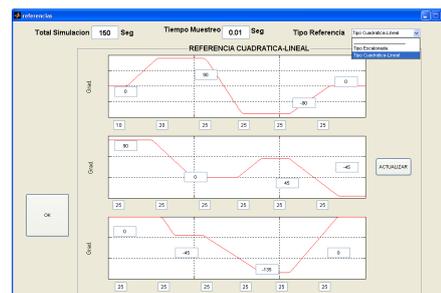
Figura 5.25: Panel de Control Principal para la Animación del Robot.

TROL CLERE y **CONTROL CPGNL** los cuales hacen el bosquejo de la trayectoria aplicando el control por linealización exacta y la trayectoria aplicando el control predictivo, respectivamente. En la Figura 5.27 nos muestra una comparación entre trayectorias, la de referencia con la del controlador predictivo. El botón **REFERENCIA** permite la ejecutar otra ventana en el cual se puede seleccionar entre 2 tipos de trayectorias de referencia y modificar sus respectivos parámetros, ver Figura 5.26(a) y 5.26(b).

En la parte central del panel de control principal se aprecia el modelo del robot en 3D, el cual adoptará movimiento conforme se manejen los demás controles. Finalmente en la parte derecha del panel principal de control se aprecia gráficas que visualizan la posición angular de las referencias comparadas con el resultado de aplicar cualquier técnica de control seleccionada en el *PANEL CONTROL*.



(a) Primer tipo de referencia



(b) Segundo tipo de referencia

Figura 5.26: Tipos de trayectorias de referencia

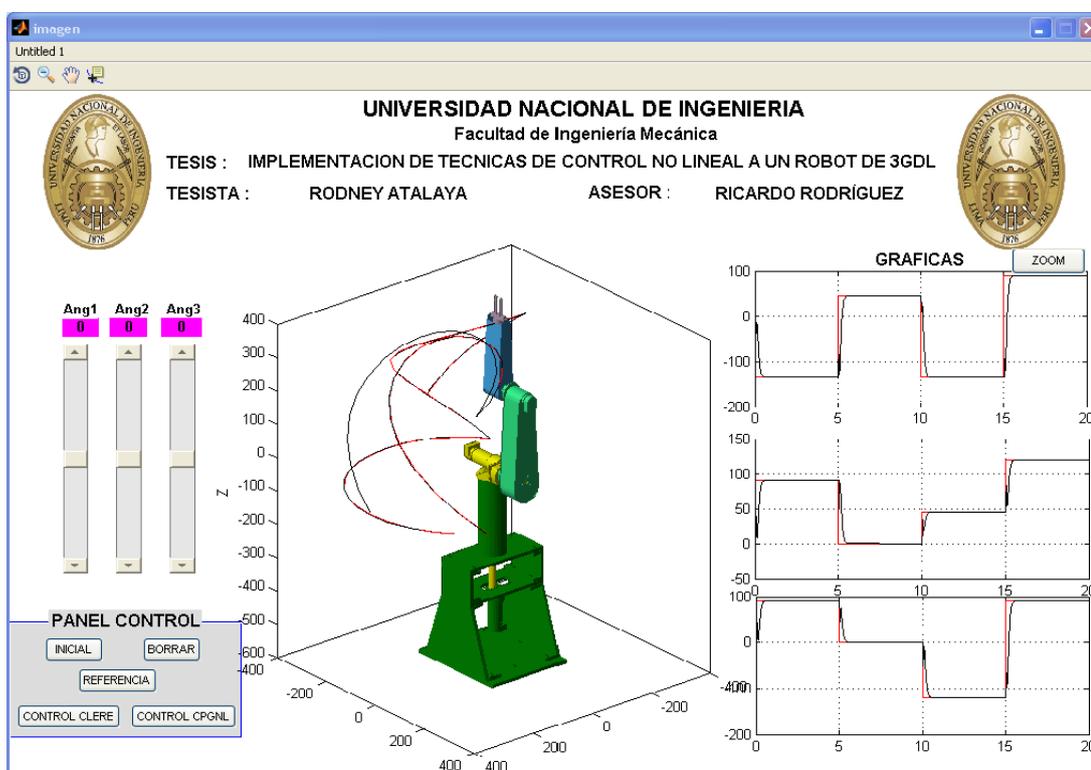


Figura 5.27: Comparación entre las trayectorias de referencia y del control predictivo.

Todos los programas empleados para animar los sistemas de control diseñados se encuentran en el CD-ROM adjunto a este trabajo de tesis.

CAPÍTULO 6

Implementación en Tiempo Real del Manipulador Robótico de 3GDL

La figura 6.1 muestra el esquema para la implementación en tiempo real del sistema robótico de 3 grados de libertad.

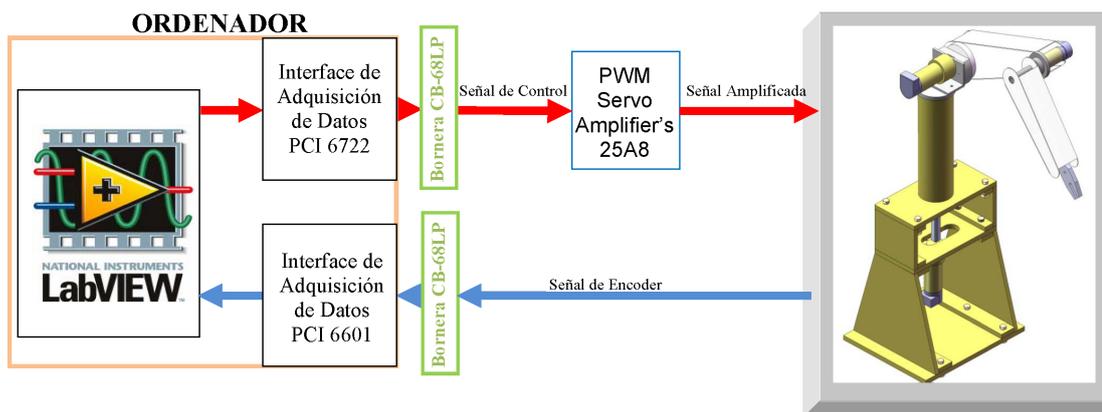


Figura 6.1: Esquema para la implementación en tiempo real del sistema robótico

Una descripción del esquema anterior se da a continuación, El algoritmo de control será implementado sobre la plataforma del Labview versión 8.5. La interface de adquisición de datos PCI-6722 de National Instruments (NI) será la encargada de enviar las señales de control provenientes del algoritmo. Éstas señales deben pasar primero por una etapa de amplificación esto se logra mediante 3 Brush Type PWM Servo Amplifier's serie 25A8 de Advance Motions Control, uno por cada señal que se amplifica. Una vez amplificada la señal de control ésta se inyecta a los motorreductores DC

de Pittman's, ver cuadro 5.2.

El movimiento del motor genera la emisión de señales (A,B y Z) que provienen del encoder acoplado a su eje. Para la lectura de las señales de los encoders se cuenta con una interface contadora de 8 canales PCI-6602 de NI.

Labview será instalado sobre un ordenador personal Procesador Dual Core Intel 1.8GHz.

El sistema robótico real se muestra en la figura 6.2. Este sistema ha sido implementado en el laboratorio de proyectos de la Sección de Postgrado de la Facultad de Ingeniería Eléctrica y Electrónica de la Universidad Nacional de Ingeniería.



Figura 6.2: Implementación en tiempo real del robot

Los dispositivos electrónicos usados para dicha implementación se detallan a continuación,

6.1. Tarjeta de Adquisición de Datos

La tarjeta de adquisición de datos PCI-6722 NI ha sido elegida como interface para manejar los datos (señal de control) que provienen del algoritmo de control implementado en Labview hacia dispositivos externos (amplificadores).

Esta tarjeta presenta las siguientes características,

- **Salida Analógica**
 - **Canales analógicos:** 8
 - **Resolución (kS/s):** 476
 - **Tipo de DAC:** Double Buffered, voltaje
 - **Tamaño de FIFO buffer (Samples):** 2047
 - **Canales DMA:** 3

- **Voltaje de Salida**
 - **Rango (V):** ± 10
 - **Impedancia de la salida (Ω):** 0.1max
 - **Corriente suministrada (mA):** ± 0.5 max

- **Entrada/Salida digital**(ver cuadro [6.1](#))
 - **Numero de canales:** 8input/output
 - **Compatibilidad:** TTL/CMOS

- **Timing I/O**
 - **Numero de canales:** 2up/down counter/timers, 1 frequency scaler
 - **Resolución:**
 - **Counter/timers (bits):** 24
 - **Frequency scaler (bits):** 4
 - **Compatibilidad:** 5V TTL/CMOS
 - **Base clock available:**
 - **Counter/timers:** 20MHz, 100kHz
 - **Frequency scaler:** 10MHz, 100kHz

- **Base clock accuracy:** $\pm 0.01\%$

En nuestro caso no se hará uso de estos canales (counter/timer) debido a que necesitamos 3 para leer los datos de los encoders (3 motores DC) y esta tarjeta sólo cuenta con 2 contadores. Por tanto se hace uso de la tarjeta contadora PCI-6602 la cual presenta 8 canales. El uso de esta tarjeta permite ahorrar tiempo en la implementación física del sistema pues una vez instalado empieza a leer la señal de los encoders y la visualiza con poca programación en el entorno del Labview.

Niveles Lógicos Digitales		
Nivel	Min	Máx
Voltaje Bajo Entrada (V)	0	0.8
Voltaje Alto Entrada (V)	2.0	5.0
Voltaje Bajo Corriente (μA)	-	-320
Voltaje Alto Corriente (μA)	-	10
Voltaje Bajo Salida (V)	-	0.4
Voltaje Alto Salida (V)	4.35	-

Cuadro 6.1: Niveles Lógicos para las Entradas/Salidas Digitales

En la figura 6.3 se aprecia la asignación de los pines de la tarjeta PCI-6722.

Para la implementación física del sistema se usó 3 pines que corresponden a *Salidas Analógicas*, en la figura 6.6 se muestran los pines usados.

6.2. El Amplificador

Las señales provenientes del algoritmo de control son enviadas al exterior por medio de la interface PCI-6722. Estas señales debido a su baja potencia no pueden ser inyectadas directamente a los motores que componen al sistema robótico, esta maniobra podría causar graves daños al ordenador o a la interface.

Por este motivo se necesita implementar una etapa de potencia entre la tarjeta PCI-6722 y el sistema robótico, esta etapa de potencia en nuestro caso está dado por 3 amplificadores Brush Type PWM Servo Amplifier serie 25A8 de Advanced Motions Control.

AO GND	34	68	NC
NC	33	67	AO GND
AO GND	32	66	AO GND
AO GND	31	65	AO 7
AO 6	30	64	AO GND
AO GND	29	63	AO GND
AO 5	28	62	NC
AO GND	27	61	AO GND
AO GND	26	60	AO 4
AO 3	25	59	AO GND
AO GND	24	58	AO GND
AO GND	23	57	AO 2
AO 0	22	56	AO GND
AO 1	21	55	AO GND
CAL	20	54	AO GND
P0.4	19	53	D GND
D GND	18	52	P0.0
P0.1	17	51	P0.5
P0.6	16	50	D GND
D GND	15	49	P0.2
+5 V	14	48	P0.7
D GND	13	47	P0.3
D GND	12	46	NC
PFI 0	11	45	EXT STROBE
PFI 1	10	44	D GND
D GND	9	43	PFI 2
+5 V	8	42	PFI 3/CTR 1 SOURCE
D GND	7	41	PFI 4/CTR 1 GATE
PFI 5/AO SAMP CLK	6	40	CTR 1 OUT
PFI 6/AO START TRIG	5	39	D GND
D GND	4	38	PFI 7
PFI 9/CTR 0 GATE	3	37	PFI 8/CTR 0 SOURCE
CTR 0 OUT	2	36	D GND
FREQ OUT	1	35	D GND

NC = No Connect

Figura 6.3: Asignación de los pines del PCI-6722 de National Instruments

6.2.1. Descripción

La figura 6.4 muestra al Brush Type PWM Servo Amplifier serie 25A8 de Advanced Motions Control el cual está encargado de manejar a los motores DC que componen al robot de 3 grados de libertad.

Aquí se muestran algunas de sus características:

- **Especificaciones Eléctricas**



Figura 6.4: Amplificador serie 25A8 de Advanced Motions Control

- Rango de voltaje DC (V): 20-80
- Máxima corriente pico de salida (A): 25
- Máxima corriente continua de salida (A): 12.5
- Máxima potencia continua de salida (W): 950
- Máxima potencia de disipación continua a corriente continua (W): 50
- Frecuencia de switcheo (kHz): 22
- Especificaciones de Control
 - Voltaje de salida analógico (V): ± 10
 - Métodos de conmutación: Brush Type
 - Modos de operación: Corriente, IR Compensación, Velocidad, Voltaje
 - Motores Soportados: Una fase (Brushed, Carga inductiva)
 - Level lógico de entrada/salida: 5V TTL
- Especificaciones Mecánicas y Ambientales
 - Tamaño(mm^3): $129.3 \times 75.8 \times 25.1$
 - Peso (g): 280

– Rango de temperatura (°C): 0-65

6.3. Motores Usados en el Robot

Los motores que componen al robot son de Pitman Motor, ver cuadro 5.2. Estos motores se han seleccionado por su torque máximo los cuales ha sido hallados mediante CosmosMotion que es una herramienta que proporciona Solidwork. La disposición de estos motores en el robot se visualiza en la figura 6.5.

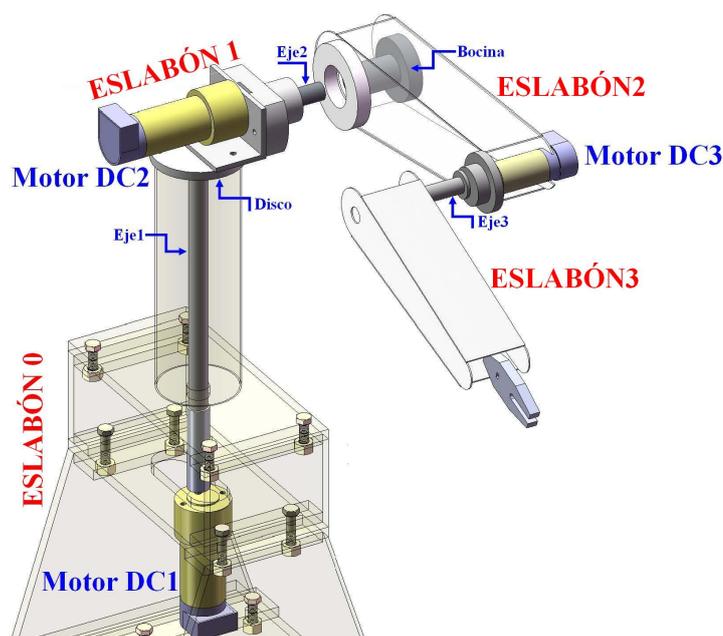


Figura 6.5: Disposición de los motores que componen al sistema robótico

6.3.1. Primer Eslabón: Base

Es accionado por un motorreductor DC1 de 24VDC con torque pico de 18 N.m, con sensor de posición incorporado (encoder óptico de 500PPR). Éste se encuentra instalado en el eslabón 0, además este eslabón está constituido por un eje1 el cual conecta al motor DC1 con un disco, sobre el cual descanza el motorreductor DC2.

6.3.2. Segundo Eslabón: Brazo

Es accionado por un motorreductor DC2 de 24VDC con torque pico de 18 N.m, con sensor de posición incorporado (encoder óptico de 500PPR). Este eslabón está consituido por un brazo: en un extemo de éste se halla una bocina que se une rígidamente con el motor DC2 a través del eje2 y en el otro extremo se encuentra el motorreductor DC3, el mismo que se conecta al eslabón 3 através del eje3.

6.3.3. Tercer Eslabón: Antebrazo

Este eslabón es accionado por el motorreductor DC3 de 24VDC con torque pico de 2.3 N.m, con sensor de posición incorporado (encoder óptico de 500PPR). En el extremo libre de este eslabón se instala la tenaza para realizar la tarea de manipulación.

6.4. Conexionado Eléctrico con la Tarjeta PCI-6722 NI

La figura 6.6 muestra las conexiones eléctricas entre la tarjeta PCI-6722, los amplificadores y finalmente los motores. Mediante unos switchs que dispone el amplificador se configura en modo voltaje, además se selecciona una determinada ganancia ($K=10$) mediante potenciómetros.

6.5. Tarjeta PCI-6602 NI

Esta tarjeta ofrece ocho contadores/temporizadores de 32 bits y hasta 32 líneas de E/S digital compatibles con TTL/CMOS. Permite realizar una amplia variedad de tareas de contador/temporizador tales como, medida de posición de codificador, conteo de eventos, medida periódica, medida de ancho de pulso, generación de pulso, generación de series de pulso y medida de frecuencia.

La PCI-6602 es elegida para la lectura de los datos que provienen de los 3 encoders (*HEDS9100*) acoplados a los ejes de cada motor.

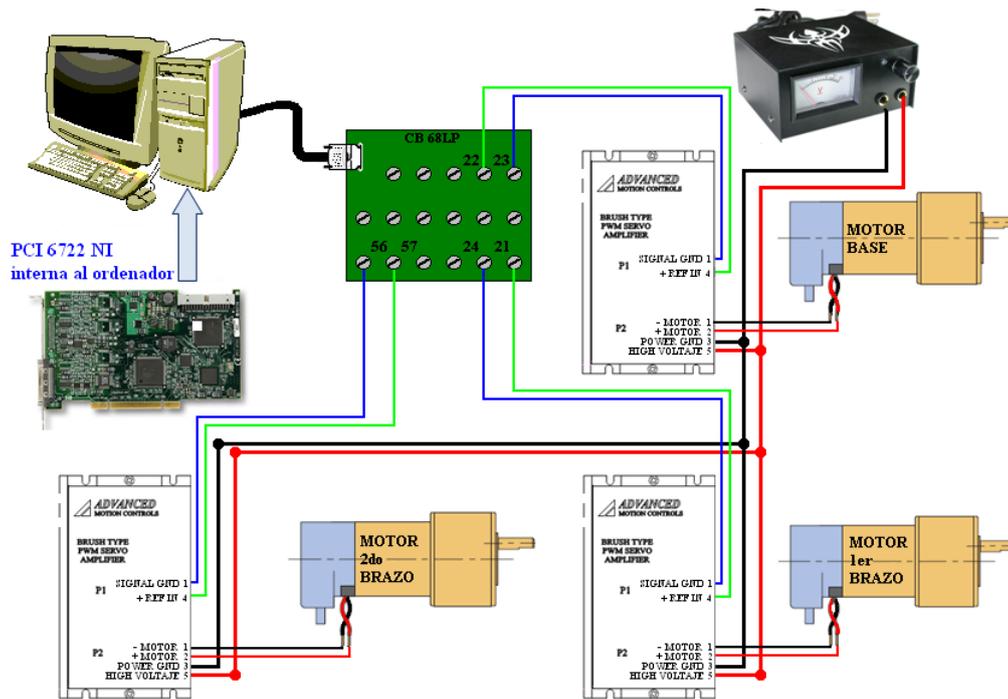


Figura 6.6: Conexión de la PCI-6722 NI con los amplificadores

- Contador/Timer de propósito general
 - Número de canales: 8
 - Número de bits del contador: 32
 - Compatibilidad: 5V TTL/CMOS
- Niveles Lógicos Digitales para Counter/Timer(ver cuadro 6.2)
 - Clock base: 100kHz, 20MHz y 80MHz
 - Precisión del clock base: $\pm 0.005\%$ Con preescalador se puede llegar hasta una frecuencia de lectura de 125MHz.
- Entradas/Salidas Digitales(ver cuadro 6.2)
 - Número de canales: Hasta 32 entradas/salidas
 - Compatibilidad: 5V TTL/CMOS
 - Histéresis: 300mV Schmitt triggers

El uso de esta tarjeta permite ahorrar tiempo en la implementación física del sistema pues una vez instalado empieza a leer la señal de los encoders y la visualiza con poca programación en el entorno del Labview.

Niveles Lógicos Digitales		
Nivel	Min	Máx
Voltaje Bajo Entrada (V)	-0.3	0.8
Voltaje Alto Entrada (V)	2.0	5.25
Voltaje Bajo Salida (V)	-	0.4
Voltaje Alto Salida (V)	2.4	-

Cuadro 6.2: Niveles Lógicos para Counter/Timer y Entradas/Salidas Digitales

En la figura 6.7 se aprecia la asignación de los pines de la tarjeta PCI-6602 de National Instruments. Para la implementación del sistema robótico se usaron ciertos pines los mismos que pueden ser vistos en la figura 6.11.

6.6. El Sensor

Para medir el desplazamiento angular del eje del motor se va a utilizar un encoder óptico incremental.

6.6.1. Tecnología del Sensor

El encoder incremental es probablemente el tipo más común de encoder utilizado en la industria, por la gran variedad de aplicaciones que su uso abarca. Este sensor está formado por tres elementos:

- Un disco con franjas transparentes y opacas alternadas, dispuestas en sentido radial.
- Un emisor de luz o fotocélula colocado en una cara del disco.
- Un receptor de luz en la cara opuesta del emisor.

Al girar el disco, figura 6.8, el haz de luz del emisor resultará interceptado por las franjas opacas y tendrá vía libre en las franjas transparentes, de modo que el

Motion Encoder Context	Signal Names	Signal Names	Motion Encoder Context
Channel_A (2)	PFI_31	34 68	GND
	GND	33 67	
Channel_A (3)	PFI_28	32 66	PFI_30
	PFI_27	31 65	
Channel_A (4)	PFI_24	29 63	PFI_29
	PFI_23	28 62	
Channel_A (5)	PFI_20	26 60	PFI_26
	PFI_19	25 59	
Channel_A (6)	PFI_16	23 57	PFI_25
	PFI_15	22 56	
Index/Z (6)	PFI_14	21 55	GND
	GND	20 54	
Channel_B (7)	RG	19 53	PFI_13
	GND	18 52	
Channel_B (7)	PFI_9	17 51	PFI_12
	PFI_8	16 50	
Channel_B (7)	PFI_7	15 49	PFI_11
	GND	14 48	
Channel_B (7)	PFI_4	13 47	PFI_10
	PFI_3	12 46	
Channel_B (7)	GND	11 45	PFI_6
	PFI_0	10 44	
Channel_B (7)	PFI_32	9 43	PFI_5
	PFI_34	8 42	
Index/Z (1)	PFI_35	7 41	GND
Channel_A (1)	PFI_33	6 40	PFI_37
Channel_B (1)	PFI_36	5 39	GND
Channel_B (1)	Reserved	4 38	Reserved
	Reserved	3 37	
Index/Z (0)	PFI_38	3 37	Reserved
Channel_A (0)	PFI_39	2 36	GND
Channel_A (0)	+5V	1 35	RG

GND: Ground

RG: 1. Reserved if using an SH68-68-D1 shielded cable
2. Ground if using an R6868 ribbon cable

Figura 6.7: Asignación de los pines del PCI-6602 de National Instruments para aplicaciones de lectura de encoders

receptor recibirá pulsos de luz.

De este modo, la señal de salida del receptor estará formada por trenes de impulsos, cuya frecuencia va a ser proporcional a la velocidad de giro del disco, y su

número proporcional al ángulo girado.

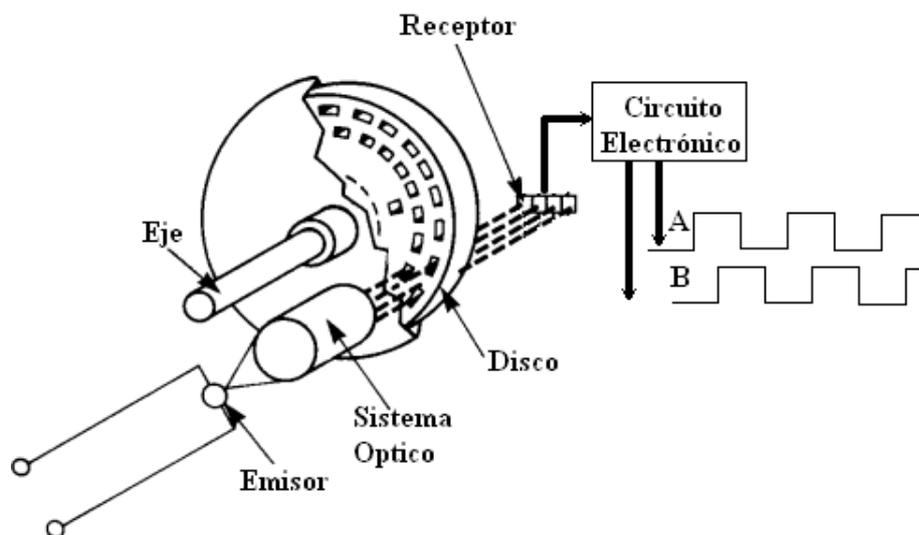


Figura 6.8: Principio de Funcionamiento del Encoder

Veamos algunas características del encoder incremental,

- Resolución. La resolución de un encoder incremental es función del número de franjas del disco, y por tanto está limitado por el tamaño del sensor.
- Presición. Depende de factores mecánicos y eléctricos entre los que cabe destacar la excentricidad del disco y la de los rodamientos, el error introducido por la electrónica de lectura, impresiones de tipo óptico, etc.
- Error de División. Viene dado por el máximo desplazamiento de dos frentes de onda consecutivos, expresado en grados eléctricos. El error de un encoder rotativo o incremental no es acumulativo, es decir no incrementa cuando el eje de más de una vuelta.

6.6.2. Descripción del Sensor Usado

En nuestro caso el encoder *HEDS9100*, ver figura 6.9, ya se encuentra acoplado al eje de cada motor Pittman, y presenta las siguientes características:

- **Características Eléctricas**

- **Tensión de alimentación:** de 4.5 a 5.5 Vcc.
 - **Intensidad máxima:** 40mA
 - **Salidas:** A,B y Z
 - **Frecuencia máxima de respuesta:** 100kHz
- **Características Mecánicas**
 - **Cuentas por revolución:** 500CPR
 - **Momento de inercia:** 10^{-6}kg.m^2
 - **Par de arranque:** $980\mu\text{N.m}$
- **Condiciones Ambientales**
 - **Temperatura ambiente de operación:** de -30 a 100°C
 - **Humedad ambiente de operación:** del 35 al 85%(Sin condensación)

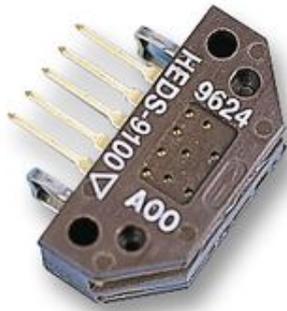


Figura 6.9: Encoder incremental HEDS9100 acoplado al motor Pittman

6.6.3. Señales A, B y Z

Con objeto de conocer el sentido de giro del eje del motor se dispone de dos secuencias de pulsos desfasadas 90° eléctricos (canales A y B), provocadas por un disco con dos pistas de franjas desfasadas 90° una respecto a la otra. Dependiendo de cuál de las dos señales de salida esté en avance de fase, se determina

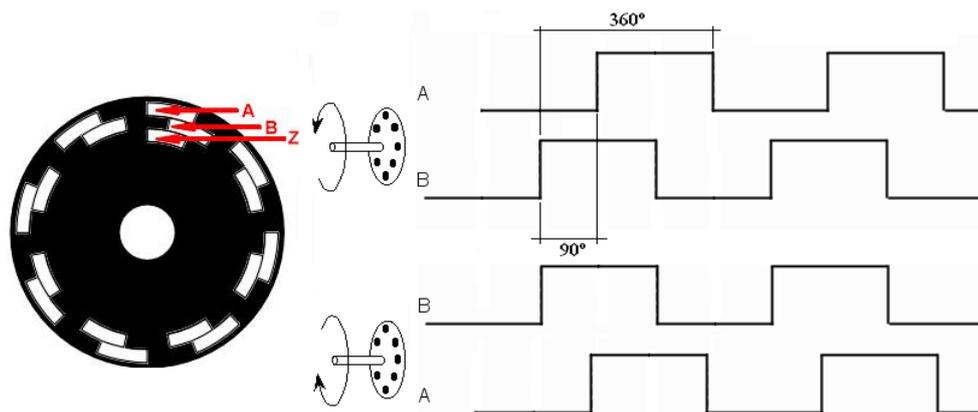


Figura 6.10: Señales A, B y Z que Proporciona el Encoder

el sentido de la rotación del disco. Con la lectura de uno sólo de los canales se obtiene información acerca de la velocidad de rotación.

Por otro lado, como el encoder incremental no determina la posición absoluta del eje, sino su posición relativa respecto a un origen, se dispone de una tercera señal (canal Z ó 0), que proporciona un impulso a cada vuelta del eje, permitiendo determinar una referencia de posición.

6.7. Conexión con la Tarjeta PCI-6602 NI

El encoder HEDS9100 tiene 5 terminales externos, 3 proporcionan la señal de salida (A, B y Z) formada por los trenes de pulsos ya vistos anteriormente y 2 para alimentar al dispositivo.

La tarea de alimentación y de lectura de las señales proporcionadas por los encoders estará a cargo de una *interfaz contadora de 8 canales* PCI-6602 de National Instruments (NI). Esta conexión se consigue por medio de la bornera CB-68LP de NI. Las conexiones son apreciadas en la figura 6.11.

o Características del PCI-6602

- **Countes/Timers:** 8
- **Maxima frecuencia:** 80MHz con preescalador puede llegar a 125MHz

- **Compatibilidad:** 5V TTL/CMOS
- **Bits:** 32

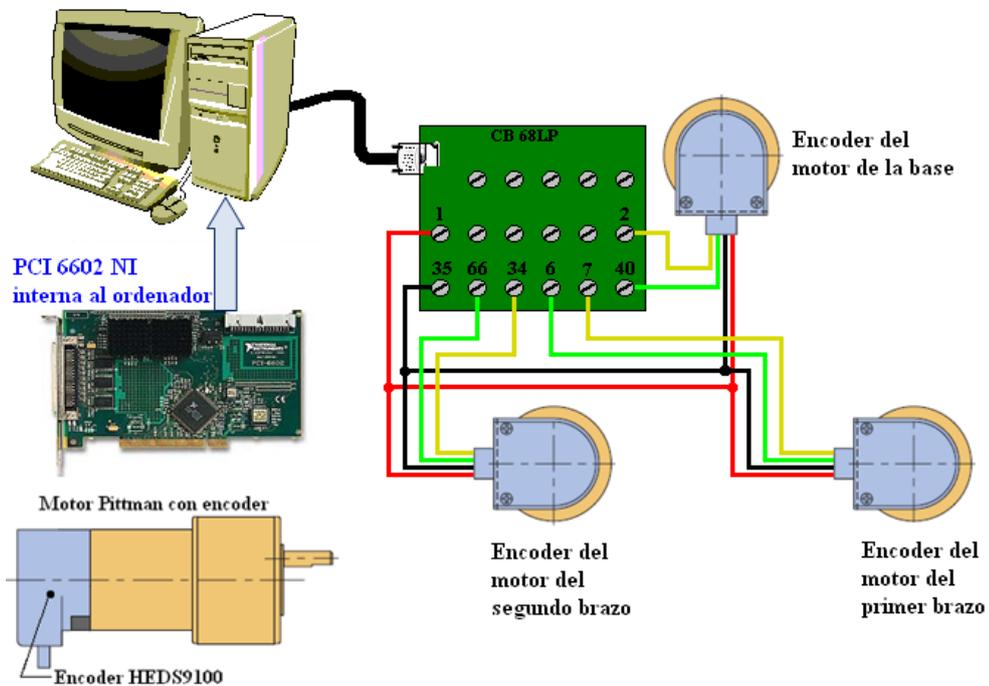


Figura 6.11: Conexión de los terminales del encoder con la bornera CB-68LP

6.8. Software de Control

El software encargado de manejar los datos que se envían (señales de control) y reciben (señales que provienen de los encoders) del sistema robótico será el lenguaje de programación gráfico Labview versión 8.5, el cual se encuentra instalado sobre un ordenador personal con Procesador Dual Core Intel 1.8GHz. Este software para su interacción con el sistema hace uso de 2 tarjetas PCI-6722 y PCI-6602 cuyas características ya fueron mencionados en secciones anteriores.

CONCLUSIONES Y RECOMENDACIONES

- De las dos técnicas desarrolladas en la presente tesis, control por linealización exacta por realimentación de estados (CLERE) y control predictivo generalizado no lineal (CPGNL), se concluye que es el CPGNL la que brinda mejor resultado para el control angular de las articulaciones del robot, simultáneamente. Esto es posible a su etapa de optimización inherente a su algoritmo de control, el cual es resuelto fuera de línea. Como se explicó anteriormente el resolver el problema de optimización en cada periodo de muestreo demandaría mucho tiempo de cálculo, el cual es crítico para sistemas de dinámica rápida, como es el caso para nuestro robot.
- Ambas técnicas de control hacen uso del modelo matemático del robot, el cual es obtenido aplicando las ecuaciones dinámicas de cuerpo rígido de Euler-Lagrange. Éste ha sido elegido para fortalecer temas como conversión de Denavit-Hartenberg (D-H), velocidad y aceleración entre sistemas de D-H, vistos en el capítulo 2.
- Para el diseño del robot se ha considerado el software Solidwork 2007. Éste nos permite obtener un dibujo de cada pieza para su posterior fabricación. En el diseño se tendrá presente evitar las interferencias las cuales pueden acarrear problemas en los resultados experimentales.
Para la fabricación del robot manipulador se considerará los siguientes materiales: hierro, acero y aluminio.

- Como trabajo futuro se plantea adicionar 3 grados de libertad formándose de esta manera la muñeca del robot. Con esto se consigue tener un sistema MIMO de 6 entradas y 6 salidas. De esta forma los algoritmos de control obtenidos en la presente tesis pueden ya ser orientados a aplicaciones industriales.
- Se estima hacer uso del software gráfico Labview 8.5 para implementar los algoritmos de control, esto por que nos permite ahorrar tiempo de programación. Para un trabajo futuro se planteará el uso de un DSP o FPGA para implementar el sistema en tiempo real.

BIBLIOGRAFÍA

- [1] Hedjar, R., Boucher, P. Nonlinear receding-horizon control of rigid link robot manipulators. *International Journal of Advanced Robotic Systems*, Volume 2, Number 1, 2005.
- [2] A. Vivas, V. Mosquera. Predictive functional control of a puma robot. *ACSE 05 Conference, Cairo, Egypt*, Volume 4, Number 2, 2005.
- [3] RAMDANE HEDJAR, REDOUANE TOUMI, PATRICK BOUCHER, DIDIER DUMU. Finite horizon nonlinear predictive control by the taylor approximation: Application to robot tracking trajectory. *International Journal of Applied Mathematics and Computer*, vol. 15, Number 4, 2005.
- [4] E.F.Camacho, C. Bordons. *Model Predictive Control (Second Edition)*. Springer, Marzo 2003.
- [5] A. Barrientos, L. Peñín., B. Balaguer. *Fundamentos de Robótica*. McGraw-Hill, Enero 2007.
- [6] K.S. Fu, R.C. González, C.S.G Lee. *Robótica, Control, Detección, Visión, Inteligencia*. McGraw-Hill, Noviembre 1990.
- [7] Bruno Siciliano, Lorenzo Sciavicco. *Robotics: Modelling, planning and control*. Springer, Julio 2008.
- [8] Arturo Rojas Moreno. *Control No Lineal Multivariable: Aplicaciones en Tiempo Real*. Marzo 2007.

- [9] Alberto Isidori. *Nonlinear Control Systems (Second Edition)*. Springer-Verlag, Noviembre 1989.
- [10] S. Joe Qin and Thomas A. Badgwell. A survey of industrial model predictive control technology. *Control Engineering Practice*, vol. 11, 2003.
- [11] Tokunbo Ogunfunmi. *Adaptive Nonlinear System Identification (The Volterra and Wiener Model Approaches)*. Springer, Septiembre 2006.

APÉNDICE A

Programa del Cálculo de la Ecuación Dinámica del Robot

```

clc,clear all
syms q1 dq1 ddq1 q2 dq2 ddq2 q3 dq3 ddq3 a2 a3 d2 d3 N1 N2 N3 Im1 Im2 Im3 I1xx I1xy I1xz I1yy I1yz ...
    I1zz m1 rCM1xx rCM1yy rCM1zz I2xx I2xy I2xz I2yy I2yz I2zz m2 rCM2xx rCM2yy rCM2zz I3xx I3xy...
    I3xz I3yy I3yz I3zz m3 rCM3xx rCM3yy rCM3zz g RT1 RT2 RT3 NT1 NT2 NT3 Jeq1 Jeq2 Jeq3 Beq1 Beq2...
    Beq3 x1 x2 x3 x4 x5 x6 T2 p1 p2 p3 dp1 dp2 dp3

%PARAMETROS DE DENAVIT-HARTEMBERG
%=====
%   i   ALPHA   THETA   D   A
%-----
%   1   pi/2   theta1   0   0
%   2    0   theta2  -d2  a2
%   3    0   theta3   d3  a3
%=====

% MATRIZ A01, R01, P01
A01=[cos(q1) 0 sin(q1) 0;
     sin(q1) 0 -cos(q1) 0;
     0 1 0 0;
     0 0 0 1];
R01=A01(1:3,1:3);P01=A01(1:3,4);
% MATRIZ A12, R12, P12
A12=[cos(q2) -sin(q2) 0 a2*cos(q2);
     sin(q2) cos(q2) 0 a2*sin(q2);
     0 0 1 -d2;
     0 0 0 1];
R12=A12(1:3,1:3);P12=A12(1:3,4);
% MATRIZ A02, R02, P02
A02=[cos(q1)*cos(q2) -cos(q1)*sin(q2) sin(q1) a2*cos(q1)*cos(q2)-d2*sin(q1);
     sin(q1)*cos(q2) -sin(q1)*sin(q2) -cos(q1) a2*sin(q1)*cos(q2)+d2*cos(q1);
     sin(q2) cos(q2) 0 a2*sin(q2);
     0 0 0 1];
R02=A02(1:3,1:3);P02=A02(1:3,4);

```

```

% MATRIZ A23, R23, P23
A23=[cos(q3) -sin(q3) 0 a3*cos(q3);sin(q3) cos(q3) 0 a3*sin(q3);0 0 1 d3;0 0 0 1];
R23=A23(1:3,1:3);P23=A23(1:3,4);
% MATRIZ A03, R03, P03
A03=[cos(q1)*cos(q2+q3) -cos(q1)*sin(q2+q3) sin(q1) a3*cos(q1)*cos(q2+q3)+(d3-d2)*sin(q1)+a2*cos(q1)*cos(q2);
      sin(q1)*cos(q2+q3) -sin(q1)*sin(q2+q3) -cos(q1) a3*sin(q1)*cos(q2+q3)+(d2-d3)*cos(q1)+a2*sin(q1)*cos(q2);
      sin(q2+q3) cos(q2+q3) 0 a3*sin(q2+q3)+a2*sin(q2);
      0 0 0 1];
R03=A03(1:3,1:3);P03=A03(1:3,4);
% CONDICIONES INICIALES
w00=[0 0 0].'; %VELOCIDAD ANGULAR
dw00=[0 0 0].'; %ACELERACION ANGULAR
p00=[0 0 0].'; %POSICION DEL SISTEMA S0
dp00=[0 0 0].'; %VELOCIDAD DEL SISTEMA S0
ddp00=[0 0 g].';%ACELERACION LINEAL DEL SISTEMA S0
f44=[0 0 0].';u44=[0 0 0].';R34=eye(3);z0=[0 0 1].';
% VECTOR QUE UNE EL SISTEMA Si-1 CON Si REPRESENTADO EN EL SISTEMA Si
r01_1=simple(R01.'*(P01-p00));
r12_2=simple(R02.'*(P02-P01));
r23_3=simple(R03.'*(P03-P02));
% VECTOR DE COORDENADAS DEL CENTRO DE MASA DEL ESLABÓN i EN Si REPRESENTADO EN Si
r1C1_1=[0 rCM1yy rCM1zz].';
r2C2_2=[rCM2xx 0 0].';
r3C3_3=[rCM3xx 0 0].';
% TENSORES DE INERCIA
I11=[I1xx 0 0;
      0 I1yy 0;
      0 0 I1zz];
I22=[I2xx 0 0;
      0 I2yy 0;
      0 0 I2zz];
I33=[I3xx 0 0;
      0 I3yy 0;
      0 0 I3zz];
% RECURSIVIDAD
% VELOCIDAD ANGULAR
w11=simple(R01.'*(w00+dq1*z0));
w22=simple(R12.'*(w11+dq2*z0));
w33=simple(R23.'*(w22+dq3*z0));
% ACCELERACIÓN ANGULAR
dw11=simple(R01.'*(dw00+ddq1*z0+dq1*S(w00)*z0));
dw22=simple(R12.'*(dw11+ddq2*z0+dq2*S(w11)*z0));
dw33=simple(R23.'*(dw22+ddq3*z0+dq3*S(w22)*z0));
% VELOCIDAD LINEAL

```

```

dp1=simple(R01.'*dp00+S(w11)*r01_1);
dp2=simple(R12.'*dp11+S(w22)*r12_2);
dp33=simple(R23.'*dp22+S(w33)*r23_3);
% ACELERACIÓN LINEAL
ddp11=simplify(R01.'*ddp00+S(dw11)*r01_1+S(w11)*S(w11)*r01_1);
ddp22=simplify(R12.'*ddp11+S(dw22)*r12_2+S(w22)*S(w22)*r12_2);
ddp33=simplify(R23.'*ddp22+S(dw33)*r23_3+S(w33)*S(w33)*r23_3);
% CÁLCULO DE ACELERACIÓN DE CENTRO DE MASA
ddpCM1_1=ddp11+S(dw11)*r1C1_1+S(w11)*S(w11)*r1C1_1;
ddpCM2_2=ddp22+S(dw22)*r2C2_2+S(w22)*S(w22)*r2C2_2;
ddpCM3_3=ddp33+S(dw33)*r3C3_3+S(w33)*S(w33)*r3C3_3;
% RECURSIVIDAD DE FUERZAS
f33=simple(R34*f44+m3*ddpCM3_3);
f22=simple(R23*f33+m2*ddpCM2_2);
f11=simple(R12*f22+m1*ddpCM1_1);
u33=simple(-S(f33)*(r23_3+r3C3_3)+R34*u44+S(R34*f44)*r3C3_3+I33*dw33+S(w33)*I33*w33);
u22=simple(-S(f22)*(r12_2+r2C2_2)+R23*u33+S(R23*f33)*r2C2_2+I22*dw22+S(w22)*I22*w22);
u11=simple(-S(f11)*(r01_1+r1C1_1)+R12*u22+S(R12*f22)*r1C1_1+I11*dw11+S(w11)*I11*w11);
T1=simple(u11.'*R01.'*z0);
T2=simple(u22.'*R12.'*z0);
T3=simple(u33.'*R23.'*z0);
% ECUACIÓN DINÁMICA DE NEWTON-EULER A LA FORMA LAGRANGIANA
% MATRIZ DE INERCIA
M11=diff(T1,ddq1);M12=diff(T1,ddq2);M13=diff(T1,ddq3);
M21=diff(T2,ddq1);M22=diff(T2,ddq2);M23=diff(T2,ddq3);
M31=diff(T3,ddq1);M32=diff(T3,ddq2);M33=diff(T3,ddq3);
% MATRIZ DE GRAVEDAD
G1=diff(T1,g)*g;G2=diff(T2,g)*g;G3=diff(T3,g)*g;
% MATRIZ DE CORIOLISIS
C1=simplify(T1-M11*ddq1-M12*ddq2-M13*ddq3-G1);
C2=simplify(T2-M21*ddq1-M22*ddq2-M23*ddq3-G2);
C3=simplify(T3-M31*ddq1-M32*ddq2-M33*ddq3-G3);
% INCLUYENDO LA DINÁMICA DE LOS MOTORES A LA ECUACIÓN DEL ROBOT
M=[RT1*(M11+Jeq1) RT1*M12 RT1*M13;
   RT2*M21 RT2*(M22+Jeq2) RT2*M23;
   RT3*M31 RT3*M32 RT3*(M33+Jeq3)];
C=[(RT1*Beq1+NT1)*dq1+RT1*C1;(RT2*Beq2+NT2)*dq2+RT2*C2;(RT3*Beq3+NT3)*dq3+RT3*C3];
G=[RT1*G1;RT2*G2;RT3*G3];
sumCG=C+G;
invM=inv(M);
MM=-invM*sumCG;
NN=invM;
F4=MM(1);F5=MM(2);F6=MM(3);
G41=NN(1,1);G42=NN(1,2);G43=NN(1,3);

```

```
G51=NN(2,1);G52=NN(2,2);G53=NN(2,3);
G61=NN(3,1);G62=NN(3,2);G63=NN(3,3);
F4=subs(F4,{q1,q2,q3,dq1,dq2,dq3},{x1,x2,x3,x4,x5,x6});
F5=subs(F5,{q1,q2,q3,dq1,dq2,dq3},{x1,x2,x3,x4,x5,x6});
F6=subs(F6,{q1,q2,q3,dq1,dq2,dq3},{x1,x2,x3,x4,x5,x6});
G41=subs(G41,{q1,q2,q3,dq1,dq2,dq3},{x1,x2,x3,x4,x5,x6});
G42=subs(G42,{q1,q2,q3,dq1,dq2,dq3},{x1,x2,x3,x4,x5,x6});
G43=subs(G43,{q1,q2,q3,dq1,dq2,dq3},{x1,x2,x3,x4,x5,x6});
G51=subs(G51,{q1,q2,q3,dq1,dq2,dq3},{x1,x2,x3,x4,x5,x6});
G52=subs(G52,{q1,q2,q3,dq1,dq2,dq3},{x1,x2,x3,x4,x5,x6});
G53=subs(G53,{q1,q2,q3,dq1,dq2,dq3},{x1,x2,x3,x4,x5,x6});
G61=subs(G61,{q1,q2,q3,dq1,dq2,dq3},{x1,x2,x3,x4,x5,x6});
G62=subs(G62,{q1,q2,q3,dq1,dq2,dq3},{x1,x2,x3,x4,x5,x6});
G63=subs(G63,{q1,q2,q3,dq1,dq2,dq3},{x1,x2,x3,x4,x5,x6});
```

APÉNDICE B

Control por Linealización Exacta por Realimentación de Estados: Caso MIMO

```

clc,clear all, close all
a2 = 0.240;a3 = 0.200;d2 = 0.113;d3 = 0.0843;
I1xx = 0.04313833;I1xy = 0;I1xz = 0;I1yy = 0.04485123;I1yz = 0;I1zz = 0.00312444;
m1 = 3.0370;rCM1xx = -0.00653211;rCM1yy = 0;rCM1zz = -0.07099550;
I2xx = 0.00151076;I2xy = 0;I2xz = -0.00065114;I2yy = 0.03065730;I2yz = 0;I2zz = 0.03006998;
m2 = 1.14084294;rCM2xx = -0.11820157;rCM2yy = 0;rCM2zz = 0.01051711;
I3xx = 0.00020817;I3xy = 0;I3xz = 0;I3yy = 0.00375672;I3yz = 0;I3zz = 0.00369897;
m3 = 0.2446;rCM3xx = -0.10496282;rCM3yy = 0;rCM3zz = 0;g = 9.8;
R1 = 2.49;K1 = 10;Imotor1 = 0.0000071;Km1 = 0.0458;Kb1 = 0.0458;Bm1 = 0.00085;N1 = 65.5;
R2 = 2.49;K2 = 10;Imotor2 = 0.0000071;Km2 = 0.0458;Kb2 = 0.0458;Bm2 = 0.00085;N2 = 65.5;
R3 = 17.0;K3 = 10;Imotor3 = 0.0000016;Km3 = 0.0436;Kb3 = 0.0436;Bm3 = 0.00011;N3 = 60.5;
JL1 = 0.00405;JL2 = 0.02552;JL3 = 0.0029;BL1 = 0.0023;BL2 = 0.0025;BL3 = 0.0030;
Jeq1=Imotor1*N1^2+JL1;Jeq2=Imotor2*N2^2+JL2;Jeq3=Imotor3*N3^2+JL3;
Beq1 = Bm1*N1^2+BL1;Beq2 = Bm2*N2^2+BL2;Beq3 = Bm3*N3^2+BL3;
RT1 = R1/(N1*Km1*K1);RT2 = R2/(N2*Km2*K2);RT3 = R3/(N3*Km3*K3);
NT1 = N1*Kb1/K1;NT2 = N2*Kb2/K2;NT3 = N3*Kb3/K3;
%% Condiciones iniciales para los estados
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;
dp1(3)=0; dp1(2)=0; dp1(1)=0;
dp2(3)=0; dp2(2)=0; dp2(1)=0;
dp3(3)=0; dp3(2)=0; dp3(1)=0;
ddp1(3)=0; ddp1(2)=0; ddp1(1)=0;
ddp2(3)=0; ddp2(2)=0; ddp2(1)=0;
ddp3(3)=0; ddp3(2)=0; ddp3(1)=0;
%% CONTROL POR REALIMENTACION EXACTA
polo1=-10;

```

```

polo2=-10;
root1=[1 -polo1];
root2=[1 -polo2];
pol1=conv(root1,root2);
Kk1=pol1(2);
Kk2=pol1(3);
polo3=-10;
polo4=-10;
root3=[1 -polo3];
root4=[1 -polo4];
pol2=conv(root3,root4);
Kk3=pol2(2);
Kk4=pol2(3);
polo5=-10;
polo6=-10;
root5=[1 -polo5];
root6=[1 -polo6];
pol3=conv(root5,root6);
Kk5=pol3(2);
Kk6=pol3(3);
%% CONTROL PREDICTIVO NO LINEAL
Nm=2000;T=0.01;
% Lazo de control
for k=4:Nm
%% REFERENCIAS
t=k*T;
    if t<5
        p1(k)=-pi/4;
    elseif t>=5 & t<15
        p1(k)=pi/6;
    elseif t>=15 & t<25
        p1(k)=pi/2;
    elseif t>=25 & t<35
        p1(k)=pi/6;
    elseif t>=35 & t<45
        p1(k)=-pi/4;
    elseif t>=45
        p1(k)=0;
    end
    %%%% REFERENCIA 2
    if t<5
        p2(k)=-pi/3;
    elseif t>=5 & t<15
        p2(k)=pi/4;

```

```

elseif t>=15 & t<25
    p2(k)=3*pi/2;
elseif t>=25 & t<35
    p2(k)=pi/4;
elseif t>=35 & t<45
    p2(k)=-pi/3;
elseif t>=45
    p2(k)=0;
end
%%%%% REFERENCIA 3
if t<5
    p3(k)=pi/2;
elseif t>=5 & t<15
    p3(k)=pi/6;
elseif t>=15 & t<25
    p3(k)=-3*pi/4;
elseif t>=25 & t<35
    p3(k)=pi/6;
elseif t>=35 & t<45
    p3(k)=pi/2;
elseif t>=45
    p3(k)=0;
end
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);
%% LEY DE CONTROL POR REALIMENTACIÓN EXACTA
% CAMBIO DE COORDENADAS PARA CONTROL POR REALIMENTACION
z1(k)=x1(k);
z2(k)=x4(k);
z3(k)=x2(k);
z4(k)=x5(k);
z5(k)=x3(k);
z6(k)=x6(k);

Hz11;Hz12;Hz13;
Hz21=Hz12;Hz22;Hz23;
Hz31=Hz13;Hz32=Hz23;Hz33;
Hz=[Hz11 Hz12 Hz13;
    Hz21 Hz22 Hz23;
    Hz31 Hz32 Hz33];

```

```

Cz1;Cz2;Cz3;
Cz=[Cz1;Cz2;Cz3];
Gz1=0;Gz2;Gz3;
Gz=[Gz1;Gz2;Gz3];
% NUEVAS ENTRADAS EQUIVALENTE : v1 - v2 - v3
v1(k)=ddp1(k)+Kk1*dp1(k)+Kk2*p1(k)-Kk1*z2(k)-Kk2*z1(k);
v2(k)=ddp2(k)+Kk3*dp2(k)+Kk4*p2(k)-Kk3*z4(k)-Kk4*z3(k);
v3(k)=ddp3(k)+Kk5*dp3(k)+Kk6*p3(k)-Kk5*z6(k)-Kk6*z5(k);
vz=[v1(k);v2(k);v3(k)];
sumCGz=Cz+Gz;
uz=Hz*vz+sumCGz;
u1(k)=uz(1,1);
u2(k)=uz(2,1);
u3(k)=uz(3,1);

% PRIMER ACTUADOR
% LIMITANDO LA FUERZA DE CONTROL
if(u1(k)>5)
    u1(k)=5;
elseif(u1(k)<-5)
    u1(k)=-5;
end

% SEGUNDO ACTUADOR
% LIMITANDO LA FUERZA DE CONTROL
if(u2(k)>5)
    u2(k)=5;
elseif(u2(k)<-5)
    u2(k)=-5;
end

% TERCER ACTUADOR
% LIMITANDO LA FUERZA DE CONTROL
if(u3(k)>5)
    u3(k)=5;
elseif(u3(k)<-5)
    u3(k)=-5;
end

%% PROCESO DE DISCRETIZACION
L2F_h=-inv(Hz)*(Cz+Gz);LGLF_h=inv(Hz);
x1(k+1)=x1(k)+T*x4(k);
x2(k+1)=x2(k)+T*x5(k);
x3(k+1)=x3(k)+T*x6(k);
F4=L2F_h(1,1);F5=L2F_h(2,1);F6=L2F_h(3,1);
G41=LGLF_h(1,1);G42=LGLF_h(1,2);G43=LGLF_h(1,3);
G51=LGLF_h(2,1);G52=LGLF_h(2,2);G53=LGLF_h(2,3);
G61=LGLF_h(3,1);G62=LGLF_h(3,2);G63=LGLF_h(3,3);

```

```

x4(k+1)=x4(k)+T*(F4+G41*u1(k)+G42*u2(k)+G43*u3(k));
x5(k+1)=x5(k)+T*(F5+G51*u1(k)+G52*u2(k)+G53*u3(k));
x6(k+1)=x6(k)+T*(F6+G61*u1(k)+G62*u2(k)+G63*u3(k));
%%
% SALIDAS %
%%
y1(k)=x1(k)*180/pi;
y2(k)=x2(k)*180/pi;
y3(k)=x3(k)*180/pi;
% ERRORES %
    e1(k)=(p1(k)-x1(k))*180/pi;
    e2(k)=(p2(k)-x2(k))*180/pi;
    e3(k)=(p3(k)-x3(k))*180/pi;
end
ejex=linspace(0,Nm*T,Nm);

figure(1)
subplot(311)
plot(ejex,p1(1:Nm)*180/pi,'k')
hold on
plot(ejex,y1(1:Nm),'-','color','k')
title('Trayectoria constante del base')
ylabel('[Grad.]')
subplot(312)
plot(ejex,p2(1:Nm)*180/pi,'k')
hold on
plot(ejex,y2(1:Nm),'-','color','k')
title('Trayectoria constante del brazo1')
ylabel('[Grad.]')
subplot(313)
hold on
plot(ejex,p3(1:Nm)*180/pi,'k')
hold on
plot(ejex,y3(1:Nm),'-','color','k')
title('Trayectoria constante del brazo2')
ylabel('[Grad.]')
xlabel('Tiempo')

figure(2)
subplot(311)
plot(ejex,u1(1:Nm),'-','color','k')
title('Señal de control de base')
ylabel('[Volt]')
subplot(312)

```

```
plot(ejex,u2(1:Nm),'-', 'color','k')
title('Señal de control de brazo')
ylabel('[Volt]')
subplot(313)
plot(ejex,u3(1:Nm),'-', 'color','k')
title('Señal de control de antebrazo')
ylabel('[Volt]')
xlabel('Tiempo')
```

APÉNDICE C

Control Predictivo Generalizado No Lineal: Caso MIMO

```

clc,clear all, close all
a2 = 0.240;a3 = 0.200;d2 = 0.113;d3 = 0.0843;
I1xx = 0.04313833;I1xy = 0;I1xz = 0;I1yy = 0.04485123;I1yz = 0;I1zz = 0.00312444;
m1 = 3.0370;rCM1xx = -0.00653211;rCM1yy = 0;rCM1zz = -0.07099550;
I2xx = 0.00151076;I2xy = 0;I2xz = -0.00065114;I2yy = 0.03065730;I2yz = 0;I2zz = 0.03006998;
m2 = 1.14084294;rCM2xx = -0.11820157;rCM2yy = 0;rCM2zz = 0.01051711;
I3xx = 0.00020817;I3xy = 0;I3xz = 0;I3yy = 0.00375672;I3yz = 0;I3zz = 0.00369897;
m3 = 0.2446;rCM3xx = -0.10496282;rCM3yy = 0;rCM3zz = 0;g = 9.8;
R1 = 2.49;K1 = 10;Imotor1 = 0.0000071;Km1 = 0.0458;Kb1 = 0.0458;Bm1 = 0.00085;N1 = 65.5;
R2 = 2.49;K2 = 10;Imotor2 = 0.0000071;Km2 = 0.0458;Kb2 = 0.0458;Bm2 = 0.00085;N2 = 65.5;
R3 = 17.0;K3 = 10;Imotor3 = 0.0000016;Km3 = 0.0436;Kb3 = 0.0436;Bm3 = 0.00011;N3 = 60.5;
JL1 = 0.00405;JL2 = 0.02552;JL3 = 0.0029;BL1 = 0.0023;BL2 = 0.0025;BL3 = 0.0030;
Jeq1=Imotor1*N1^2+JL1;Jeq2=Imotor2*N2^2+JL2;Jeq3=Imotor3*N3^2+JL3;
Beq1 = Bm1*N1^2+BL1;Beq2 = Bm2*N2^2+BL2;Beq3 = Bm3*N3^2+BL3;
RT1 = R1/(N1*Km1*K1);RT2 = R2/(N2*Km2*K2);RT3 = R3/(N3*Km3*K3);
NT1 = N1*Kb1/K1;NT2 = N2*Kb2/K2;NT3 = N3*Kb3/K3;
% CONDICIONES INICIALES PARA LOS ESTADOS
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;
dp1(3)=0; dp1(2)=0; dp1(1)=0;
dp2(3)=0; dp2(2)=0; dp2(1)=0;
dp3(3)=0; dp3(2)=0; dp3(1)=0;
ddp1(3)=0; ddp1(2)=0; ddp1(1)=0;
ddp2(3)=0; ddp2(2)=0; ddp2(1)=0;
ddp3(3)=0; ddp3(2)=0; ddp3(1)=0;
% CONTROL PREDICTIVO NO LINEAL
T2=0.5; % HORIZONTE DE PREDICCIÓN
Nm=2000;T=0.01;
% LAZO DE CONTROL

```

```

for k=4:Nm
% REFERENCIAS
t=k*T;
    if t<5
        p1(k)=-pi/4;
    elseif t>=5 & t<15
        p1(k)=pi/6;
    elseif t>=15 & t<25
        p1(k)=pi/2;
    elseif t>=25 & t<35
        p1(k)=pi/6;
    elseif t>=35 & t<45
        p1(k)=-pi/4;
    elseif t>=45
        p1(k)=0;
    end
    if t<5
        p2(k)=-pi/3;
    elseif t>=5 & t<15
        p2(k)=pi/4;
    elseif t>=15 & t<25
        p2(k)=3*pi/2;
    elseif t>=25 & t<35
        p2(k)=pi/4;
    elseif t>=35 & t<45
        p2(k)=-pi/3;
    elseif t>=45
        p2(k)=0;
    end
    if t<5
        p3(k)=pi/2;
    elseif t>=5 & t<15
        p3(k)=pi/6;
    elseif t>=15 & t<25
        p3(k)=-3*pi/4;
    elseif t>=25 & t<35
        p3(k)=pi/6;
    elseif t>=35 & t<45
        p3(k)=pi/2;
    elseif t>=45
        p3(k)=0;
    end
    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);
% ELEMENTOS DE MATRIZ DE INERCIA INCLUYENDO MOTORES.
H11;H12;H13;
H21=H12;H22;H23;
H31=H13;H32=H23;H33;
H=[H11 H12 H13;
   H21 H22 H23;
   H31 H32 H33];
C1;C2;C3;
C=[C1;C2;C3];
G1=0;G2;G3;
G=[G1;G2;G3];
L2F_h=-inv(H)*(C+G);
LGLF_h=inv(H);
% LEY DE CONTROL CONTROL PREDICTIVO
K1=42/T2^2;
K2=21/(2*T2);
u=-H*(K1*[x1(k)-p1(k);x2(k)-p2(k);x3(k)-p3(k)] + K2*[x4(k)-dp1(k);x5(k)-dp2(k);x6(k)-dp3(k)]+ ...
   L2F_h-[ddp1(k);ddp2(k);ddp3(k)]);
u1(k)=u(1,1);
u2(k)=u(2,1);
u3(k)=u(3,1);
% PRIMER ACTUADOR
% LIMITANDO LA FUERZA DE CONTROL
   if(u1(k)>5)
       u1(k)=5;
   elseif(u1(k)<-5)
       u1(k)=-5;
   end
%
% SEGUNDO ACTUADOR
% LIMITANDO LA FUERZA DE CONTROL
   if(u2(k)>5)
       u2(k)=5;
   elseif(u2(k)<-5)
       u2(k)=-5;
   end
%
% TERCER ACTUADOR
% LIMITANDO LA FUERZA DE CONTROL
   if(u3(k)>5)

```

```

        u3(k)=5;
    elseif(u3(k)<-5)
        u3(k)=-5;
    end
% PROCESO DE LINEALIZACIÓN
x1(k+1)=x1(k)+T*x4(k);
x2(k+1)=x2(k)+T*x5(k);
x3(k+1)=x3(k)+T*x6(k);
F4=L2F_h(1,1);
F5=L2F_h(2,1);
F6=L2F_h(3,1);
G41=LGLF_h(1,1);
G42=LGLF_h(1,2);
G43=LGLF_h(1,3);
G51=LGLF_h(2,1);
G52=LGLF_h(2,2);
G53=LGLF_h(2,3);
G61=LGLF_h(3,1);
G62=LGLF_h(3,2);
G63=LGLF_h(3,3);
x4(k+1)=x4(k)+T*(F4+G41*u1(k)+G42*u2(k)+G43*u3(k));
x5(k+1)=x5(k)+T*(F5+G51*u1(k)+G52*u2(k)+G53*u3(k));
x6(k+1)=x6(k)+T*(F6+G61*u1(k)+G62*u2(k)+G63*u3(k));
%% Salidas
y1(k)=x1(k)*180/pi;
y2(k)=x2(k)*180/pi;
y3(k)=x3(k)*180/pi;
%% ERRORES
e1(k)=(p1(k)-x1(k))*180/pi;
e2(k)=(p2(k)-x2(k))*180/pi;
e3(k)=(p3(k)-x3(k))*180/pi;
end
%% GRAFICAS
ejex=linspace(0,Nm*T,Nm);
figure(1)
subplot(311)
hold on
plot(ejex,p1(1:Nm)*180/pi,'k')
hold on
plot(ejex,y1(1:Nm),'-','color','k')
title('Trayectoria constante del base')
ylabel(' [Grad.] ')
subplot(312)
hold on

```

```
plot(ejex,p2(1:Nm)*180/pi,'k')
hold on
plot(ejex,y2(1:Nm),'-','color','k')
title('Trayectoria constante del brazo1')
ylabel('[Grad.]')
subplot(313)
hold on
plot(ejex,p3(1:Nm)*180/pi,'k')
hold on
plot(ejex,y3(1:Nm),'-','color','k')
title('Trayectoria constante del brazo2')
ylabel('[Grad.]')
xlabel('Tiempo')
legend('Location','SouthEast')%

figure(2)
subplot(311)
plot(ejex,u1(1:Nm),'-','color','k')
title('Señal de control de base')
ylabel('[Volt]')
subplot(312)
plot(ejex,u2(1:Nm),'-','color','k')
title('Señal de control de brazo')
ylabel('[Volt]')
subplot(313)
plot(ejex,u3(1:Nm),'-','color','k')
title('Señal de control de antebrazo')
ylabel('[Volt]')
xlabel('Tiempo')
```

APÉNDICE D

TÉRMINOS DE MATRIZ DE INERCIA: H_{ij}

$$\begin{aligned}
H_{z11}=H_{11}=RT1 * & (-2*m3*d3*d2+m1*rCM1xx^2+m1*rCM1zz^2+2*m2*a2*cos(z3(k))^2*rCM2xx \\
& +I3xx-2*m2*a2*cos(z3(k))*sin(z3(k))*rCM2yy-2*m2*cos(z3(k))*rCM2xx*sin(z3(k))*rCM2yy \\
& -I3xx*cos(z3(k)+z5(k))^2-I2xx*cos(z3(k))^2+I2yy*cos(z3(k))^2 \\
& +2*m3*a2*cos(z3(k))*cos(z3(k)+z5(k))*rCM3xx-2*m3*a2*cos(z3(k))*sin(z3(k) \\
& +z5(k))*rCM3yy+2*m3*a2*cos(z3(k))*a3*cos(z3(k)+z5(k))+I1yy-2*m2*d2*rCM2zz \\
& +I2xx+m2*rCM2yy^2+m2*d2^2+m2*rCM2zz^2+m3*rCM3zz^2+m3*d2^2+m3*d3^2 \\
& +m2*a2^2*cos(z3(k))^2-m2*cos(z3(k))^2*rCM2yy^2+2*sin(z3(k) \\
& +z5(k))*I3xy*cos(z3(k)+z5(k))+m3*cos(z3(k)+z5(k))^2*rCM3xx^2 \\
& +m3*a3^2*cos(z3(k)+z5(k))^2+2*m3*d3*rCM3zz-2*m3*d2*rCM3zz \\
& +m3*rCM3yy^2-2*m3*cos(z3(k)+z5(k))*rCM3xx*sin(z3(k)+z5(k))*rCM3yy \\
& -2*m3*a3*cos(z3(k)+z5(k))*sin(z3(k)+z5(k))*rCM3yy+I3yy*cos(z3(k) \\
& +z5(k))^2+Jeq1+m3*a2^2*cos(z3(k))^2-m3*rCM3yy^2*cos(z3(k)+z5(k))^2 \\
& +m2*cos(z3(k))^2*rCM2xx^2+2*sin(z3(k))*I2xy*cos(z3(k)))+2*m3*a3*cos(z3(k)+z5(k))^2*rCM3xx;
\end{aligned}$$

$$\begin{aligned}
H_{z12}=H_{12}=RT1 * & (m2*d2*a2*sin(z3(k))+m2*d2*sin(z3(k))*rCM2xx+m2*d2*cos(z3(k))*rCM2yy \\
& -m2*rCM2zz*a2*sin(z3(k))-m2*rCM2zz*sin(z3(k))*rCM2xx-m2*rCM2zz*cos(z3(k))*rCM2yy \\
& +I2xz*sin(z3(k))+I2yz*cos(z3(k))-m3*d3*a2*sin(z3(k))+m3*d2*a2*sin(z3(k)) \\
& -m3*rCM3zz*a2*sin(z3(k))+I3yz*cos(z3(k)+z5(k))+m3*d2*cos(z3(k)+z5(k))*rCM3yy \\
& -m3*rCM3zz*a3*sin(z3(k)+z5(k))-m3*rCM3zz*sin(z3(k)+z5(k))*rCM3xx-m3*rCM3zz*cos(z3(k) \\
& +z5(k))*rCM3yy-m3*d3*a3*sin(z3(k)+z5(k))-m3*d3*sin(z3(k)+z5(k))*rCM3xx \\
& -m3*d3*cos(z3(k)+z5(k))*rCM3yy+m3*d2*a3*sin(z3(k)+z5(k))+m3*d2*sin(z3(k) \\
& +z5(k))*rCM3xx+I3xz*sin(z3(k)+z5(k)));
\end{aligned}$$

$$\begin{aligned}
H_{z13}=H_{13}=RT1 * & (I3yz*cos(z3(k)+z5(k))+m3*d2*cos(z3(k)+z5(k))*rCM3yy-m3*rCM3zz*a3*sin(z3(k) \\
& +z5(k))-m3*rCM3zz*sin(z3(k)+z5(k))*rCM3xx-m3*rCM3zz*cos(z3(k)+z5(k))*rCM3yy \\
& -m3*d3*a3*sin(z3(k)+z5(k))-m3*d3*sin(z3(k)+z5(k))*rCM3xx-m3*d3*cos(z3(k) \\
& +z5(k))*rCM3yy+m3*d2*a3*sin(z3(k)+z5(k))+m3*d2*sin(z3(k)+z5(k))*rCM3xx+I3xz*sin(z3(k)+z5(k)));
\end{aligned}$$

$$\begin{aligned}
H_{z22}=H_{22}=RT2 * & (I2zz+2*m2*a2*rCM2xx+m2*rCM2xx^2+m2*a2^2+m2*rCM2yy^2+m3*a2^2 \\
& +2*m3*a2*cos(z3(k))*cos(z3(k)+z5(k))*rCM3xx-2*m3*a2*cos(z3(k))*sin(z3(k) \\
& +z5(k))*rCM3yy+2*m3*a2*sin(z3(k))*cos(z3(k)+z5(k))*rCM3yy \\
& +2*m3*a2*sin(z3(k))*a3*sin(z3(k)+z5(k))+2*m3*a2*cos(z3(k))*a3*cos(z3(k)+z5(k))
\end{aligned}$$

$$+I3zz+2*m3*a2*\sin(z3(k))*\sin(z3(k)+z5(k))*rCM3xx+m3*rCM3yy^2+2*m3*a3*rCM3xx+m3*a3^2+m3*rCM3xx^2+Jeq2);$$

$$\begin{aligned} Hz23=H23=RT2*(m3*a2*\cos(z3(k))*\cos(z3(k)+z5(k))*rCM3xx \\ -m3*a2*\cos(z3(k))*\sin(z3(k)+z5(k))*rCM3yy+m3*a2*\sin(z3(k))*\cos(z3(k) \\ +z5(k))*rCM3yy+m3*a2*\sin(z3(k))*a3*\sin(z3(k)+z5(k)) \\ +m3*a2*\cos(z3(k))*a3*\cos(z3(k)+z5(k))+I3zz+m3*a2*\sin(z3(k))*\sin(z3(k) \\ +z5(k))*rCM3xx+m3*rCM3yy^2+2*m3*a3*rCM3xx+m3*a3^2+m3*rCM3xx^2); \end{aligned}$$

$$Hz33=H33=RT3*(I3zz+m3*rCM3yy^2+2*m3*a3*rCM3xx+m3*a3^2+m3*rCM3xx^2+Jeq3);$$

APÉNDICE E

TÉRMINOS DE CORIOLISIS: C_j

$$\begin{aligned}
Cz_1=C_1=& (RT_1*Beq_1+NT_1)*z_2(k)+(-2*z_2(k)*m_3*\cos(z_3(k)+z_5(k))*r_{CM3xx}^2*\sin(z_3(k) \\
& +z_5(k))*z_6(k)+m_3*r_{CM3zz}*\sin(z_3(k)+z_5(k))*r_{CM3yy}*z_6(k)^2-m_3*d_3*a_3*\cos(z_3(k) \\
& +z_5(k))*z_6(k)^2-4*z_2(k)*m_3*\cos(z_3(k)+z_5(k))^2*r_{CM3xx}*r_{CM3yy}*z_6(k) \\
& -m_3*d_3*a_2*\cos(z_3(k))*z_4(k)^2+m_3*d_2*a_2*\cos(z_3(k))*z_4(k)^2 \\
& +m_2*d_2*a_2*\cos(z_3(k))*z_4(k)^2-m_3*r_{CM3zz}*\cos(z_3(k)+z_5(k))*r_{CM3xx}*z_6(k)^2 \\
& -m_3*d_3*\cos(z_3(k)+z_5(k))*r_{CM3xx}*z_6(k)^2+m_3*d_3*\sin(z_3(k) \\
& +z_5(k))*r_{CM3yy}*z_6(k)^2+m_3*d_2*a_3*\cos(z_3(k)+z_5(k))*z_6(k)^2+m_3*d_2*\cos(z_3(k) \\
& +z_5(k))*r_{CM3xx}*z_6(k)^2-2*z_4(k)*m_3*d_2*\sin(z_3(k)+z_5(k))*r_{CM3yy}*z_6(k) \\
& -m_3*d_2*\sin(z_3(k)+z_5(k))*r_{CM3yy}*z_6(k)^2+2*z_4(k)*m_3*d_2*\cos(z_3(k) \\
& +z_5(k))*r_{CM3xx}*z_6(k)-2*z_4(k)*m_3*d_3*a_3*\cos(z_3(k)+z_5(k))*z_6(k) \\
& -2*z_4(k)*m_3*d_3*\cos(z_3(k)+z_5(k))*r_{CM3xx}*z_6(k)+2*z_4(k)*m_3*d_3*\sin(z_3(k) \\
& +z_5(k))*r_{CM3yy}*z_6(k)+2*z_4(k)*m_3*d_2*a_3*\cos(z_3(k)+z_5(k))*z_6(k) \\
& -m_3*r_{CM3zz}*a_3*\cos(z_3(k)+z_5(k))*z_6(k)^2-2*z_4(k)*m_3*r_{CM3zz}*a_3*\cos(z_3(k) \\
& +z_5(k))*z_6(k)-2*z_4(k)*m_3*r_{CM3zz}*\cos(z_3(k)+z_5(k))*r_{CM3xx}*z_6(k) \\
& +2*z_4(k)*m_3*r_{CM3zz}*\sin(z_3(k)+z_5(k))*r_{CM3yy}*z_6(k)-4*z_2(k)*m_2*a_2*\cos(z_3(k))*r_{CM2xx}*\sin(z_3(k))*z_4(k) \\
& +I_2xz*\cos(z_3(k))*z_4(k)^2-2*z_2(k)*I_3xy*z_4(k)-2*z_2(k)*I_2xy*z_4(k) \\
& +4*z_2(k)*\cos(z_3(k))^2*z_4(k)*I_2xy+I_3xz*\cos(z_3(k)+z_5(k))*z_6(k)^2-I_3yz*\sin(z_3(k) \\
& +z_5(k))*z_6(k)^2-2*z_2(k)*I_3xy*z_6(k)+I_3xz*\cos(z_3(k)+z_5(k))*z_4(k)^2-I_3yz*\sin(z_3(k) \\
& +z_5(k))*z_4(k)^2+4*z_2(k)*\cos(z_3(k)+z_5(k))^2*I_3xy*z_6(k) \\
& -2*z_2(k)*I_2yy*\cos(z_3(k))*\sin(z_3(k))*z_4(k)-2*z_2(k)*m_3*a_2^2*\cos(z_3(k))*\sin(z_3(k))*z_4(k) \\
& -2*z_4(k)*I_3yz*\sin(z_3(k)+z_5(k))*z_6(k)-m_3*r_{CM3zz}*a_2*\cos(z_3(k))*z_4(k)^2 \\
& -I_2yz*\sin(z_3(k))*z_4(k)^2-2*z_2(k)*m_3*a_3^2*\cos(z_3(k)+z_5(k))*\sin(z_3(k)+z_5(k))*z_4(k) \\
& -2*z_2(k)*m_3*a_2*\sin(z_3(k))*z_4(k)*a_3*\cos(z_3(k)+z_5(k))-2*z_2(k)*m_3*a_2*\sin(z_3(k))*z_4(k)*\cos(z_3(k) \\
& +z_5(k))*r_{CM3xx}+2*z_2(k)*m_3*a_2*\sin(z_3(k))*z_4(k)*\sin(z_3(k)+z_5(k))*r_{CM3yy} \\
& -4*z_2(k)*m_2*a_2*\cos(z_3(k))^2*z_4(k)*r_{CM2yy}-4*z_2(k)*m_3*a_3*\cos(z_3(k) \\
& +z_5(k))*r_{CM3xx}*\sin(z_3(k)+z_5(k))*z_4(k)+2*z_2(k)*I_3xx*\cos(z_3(k) \\
& +z_5(k))*\sin(z_3(k)+z_5(k))*z_4(k)-2*z_2(k)*m_3*\cos(z_3(k) \\
& +z_5(k))*r_{CM3xx}^2*\sin(z_3(k)+z_5(k))*z_4(k)-2*z_2(k)*m_3*a_2*\cos(z_3(k))*\cos(z_3(k) \\
& +z_5(k))*r_{CM3yy}*z_4(k)-2*z_2(k)*m_3*a_2*\cos(z_3(k))*\sin(z_3(k)+z_5(k))*r_{CM3xx}*z_4(k) \\
& -2*z_2(k)*I_3yy*\cos(z_3(k)+z_5(k))*\sin(z_3(k)+z_5(k))*z_4(k) \\
& +2*z_2(k)*m_3*r_{CM3yy}^2*\cos(z_3(k)+z_5(k))*\sin(z_3(k)+z_5(k))*z_4(k) \\
& -4*z_2(k)*m_3*\cos(z_3(k)+z_5(k))^2*r_{CM3xx}*r_{CM3yy}*z_4(k)+2*z_4(k)*I_3xz*\cos(z_3(k)
\end{aligned}$$

$$\begin{aligned}
& +z5(k))*z6(k)-2*z2(k)*m3*a2*cos(z3(k))*a3*sin(z3(k)+z5(k))*z4(k) \\
& +m3*rCM3zz*sin(z3(k)+z5(k))*rCM3yy*z4(k)^2-m3*d3*a3*cos(z3(k)+z5(k))*z4(k)^2 \\
& -4*z2(k)*m3*a3*cos(z3(k)+z5(k))^2*rCM3yy*z4(k)-m3*d3*cos(z3(k) \\
& +z5(k))*rCM3xx*z4(k)^2+m3*d3*sin(z3(k)+z5(k))*rCM3yy*z4(k)^2 \\
& +m3*d2*a3*cos(z3(k)+z5(k))*z4(k)^2+m3*d2*cos(z3(k)+z5(k))*rCM3xx*z4(k)^2 \\
& -2*z2(k)*m2*a2^2*cos(z3(k))*sin(z3(k))*z4(k)-m3*d2*sin(z3(k)+z5(k))*rCM3yy*z4(k)^2 \\
& -m3*rCM3zz*a3*cos(z3(k)+z5(k))*z4(k)^2-2*z2(k)*m2*cos(z3(k))*rCM2xx^2*sin(z3(k))*z4(k) \\
& -4*z2(k)*m2*cos(z3(k))^2*rCM2xx*z4(k)*rCM2yy+2*z2(k)*m2*cos(z3(k))*rCM2yy^2*sin(z3(k))*z4(k) \\
& +4*z2(k)*cos(z3(k)+z5(k))^2*I3xy*z4(k)-m3*rCM3zz*cos(z3(k)+z5(k))*rCM3xx*z4(k)^2 \\
& +2*z2(k)*I2xx*cos(z3(k))*sin(z3(k))*z4(k)+2*z2(k)*m3*rCM3yy^2*cos(z3(k)+z5(k))*sin(z3(k) \\
& +z5(k))*z6(k)-4*z2(k)*m3*a3*cos(z3(k)+z5(k))*rCM3xx*sin(z3(k)+z5(k))*z6(k) \\
& -2*z2(k)*m3*a2*cos(z3(k))*sin(z3(k)+z5(k))*rCM3xx*z6(k)-4*z2(k)*m3*a3*cos(z3(k) \\
& +z5(k))^2*rCM3yy*z6(k)+2*z2(k)*I3xx*cos(z3(k)+z5(k))*sin(z3(k)+z5(k))*z6(k) \\
& +2*z2(k)*m3*rCM3xx*rCM3yy*z6(k)-2*z2(k)*m3*a3^2*cos(z3(k) \\
& +z5(k))*sin(z3(k)+z5(k))*z6(k)+2*z2(k)*m3*rCM3xx*rCM3yy*z4(k) \\
& -2*z2(k)*m3*a2*cos(z3(k))*cos(z3(k)+z5(k))*rCM3yy*z6(k) \\
& -2*z2(k)*m3*a2*cos(z3(k))*a3*sin(z3(k)+z5(k))*z6(k) \\
& -2*z2(k)*I3yy*cos(z3(k)+z5(k))*sin(z3(k)+z5(k))*z6(k) \\
& +2*z2(k)*m3*a3*rCM3yy*z6(k)+2*z2(k)*m3*a3*rCM3yy*z4(k) \\
& +2*z2(k)*m2*a2*z4(k)*rCM2yy+2*z2(k)*m2*z4(k)*rCM2xx*rCM2yy+m2*rCM2zz*sin(z3(k))*z4(k)^2*rCM2yy \\
& +m2*d2*cos(z3(k))*z4(k)^2*rCM2xx-m2*d2*sin(z3(k))*z4(k)^2*rCM2yy \\
& -m2*rCM2zz*a2*cos(z3(k))*z4(k)^2-m2*rCM2zz*cos(z3(k))*z4(k)^2*rCM2xx)*RT1;
\end{aligned}$$

$$\begin{aligned}
Cz2=C2=(RT2*Beq2+NT2)*z4(k)+(2*z2(k)^2*m3*a3*cos(z3(k)+z5(k))*rCM3xx*sin(z3(k)+z5(k)) \\
+2z2(k)^2*m3*a2*cos(z3(k))*sin(z3(k)+z5(k))*rCM3xx+z2(k)^2*m3*a2*cos(z3(k))*cos(z3(k) \\
+z5(k))*rCM3yy+z2(k)^2*m3*cos(z3(k)+z5(k))*rCM3xx^2*sin(z3(k)+z5(k))-2*z2(k)^2*cos(z3(k) \\
+z5(k))^2*I3xy-2*z2(k)^2*cos(z3(k))^2*I2xy-z2(k)^2*m3*rCM3yy^2*cos(z3(k)+z5(k))*sin(z3(k) \\
+z5(k))+2*z2(k)^2*m3*a3*cos(z3(k)+z5(k))^2*rCM3yy+z2(k)^2*m3*a3^2*cos(z3(k)+z5(k))*sin(z3(k) \\
+z5(k))-z2(k)^2*m3*rCM3xx*rCM3yy-z2(k)^2*I2xx*cos(z3(k))*sin(z3(k)) \\
+z2(k)^2*m3*a2*sin(z3(k))*cos(z3(k)+z5(k))*rCM3xx-z2(k)^2*m3*a2*sin(z3(k))*sin(z3(k) \\
+z5(k))*rCM3yy+z2(k)^2*m3*a2*sin(z3(k))*a3*cos(z3(k) \\
+z5(k))+z2(k)^2*I2xy-z2(k)^2*m2*a2*rCM2yy+z2(k)^2*I3xy+z2(k)^2*I2yy*cos(z3(k))*sin(z3(k) \\
+z2(k)^2*m2*cos(z3(k))*rCM2xx^2*sin(z3(k))+2*z2(k)^2*m2*a2*cos(z3(k))^2*rCM2yy \\
+2*z2(k)^2*m2*cos(z3(k))^2*rCM2xx*rCM2yy-z2(k)^2*m2*cos(z3(k))*rCM2yy^2*sin(z3(k)) \\
+2*z2(k)^2*m2*a2*cos(z3(k))*rCM2xx*sin(z3(k))+z2(k)^2*m2*a2^2*cos(z3(k))*sin(z3(k) \\
+z2(k)^2*m3*a2^2*cos(z3(k))*sin(z3(k))-z2(k)^2*m2*rCM2xx*rCM2yy-z2(k)^2*m3*a3*rCM3yy \\
-m3*a2*cos(z3(k))*sin(z3(k)+z5(k))*rCM3xx*z6(k)^2-m3*a2*cos(z3(k))*a3*sin(z3(k) \\
+z5(k))*z6(k)^2-2*z4(k)*m3*a2*cos(z3(k))*sin(z3(k)+z5(k))*rCM3xx*z6(k) \\
-m3*a2*sin(z3(k))*sin(z3(k)+z5(k))*rCM3yy*z6(k)^2+2*z4(k)*m3*a2*sin(z3(k))*a3*cos(z3(k) \\
+z5(k))*z6(k)-2*z4(k)*m3*a2*cos(z3(k))*a3*sin(z3(k)+z5(k))*z6(k)+m3*a2*sin(z3(k))*cos(z3(k) \\
+z5(k))*rCM3xx*z6(k)^2+2*z4(k)*m3*a2*sin(z3(k))*cos(z3(k)+z5(k))*rCM3xx*z6(k) \\
-m3*a2*cos(z3(k))*cos(z3(k)+z5(k))*rCM3yy*z6(k)^2+m3*a2*sin(z3(k))*a3*cos(z3(k)+z5(k))*z6(k)^2 \\
-2*z4(k)*m3*a2*cos(z3(k))*cos(z3(k)+z5(k))*rCM3yy*z6(k)-2*z4(k)*m3*a2*sin(z3(k))*sin(z3(k)
\end{aligned}$$

$$\begin{aligned}
&+z5(k))*rCM3yy*z6(k)+z2(k)^2*I3yy*\cos(z3(k)+z5(k))*\sin(z3(k)+z5(k))-z2(k)^2*I3xx*\cos(z3(k) \\
&+z5(k))*\sin(z3(k)+z5(k))+z2(k)^2*m3*a2*\cos(z3(k))*a3*\sin(z3(k)+z5(k))+2*z2(k)^2*m3*\cos(z3(k) \\
&+z5(k))^2*rCM3xx*rCM3yy)*RT2;
\end{aligned}$$

$$\begin{aligned}
Cz3=C3=(RT3*Beq3+NT3)*z6(k)+(-z2(k)^2*I3xx*\cos(z3(k)+z5(k))*\sin(z3(k)+z5(k)) \\
-2*z2(k)^2*\cos(z3(k)+z5(k))^2*I3xy+2*z2(k)^2*m3*\cos(z3(k)+z5(k))^2*rCM3xx*rCM3yy \\
+z2(k)^2*m3*a2*\cos(z3(k))*a3*\sin(z3(k)+z5(k))+2*z2(k)^2*m3*a3*\cos(z3(k) \\
+z5(k))*rCM3xx*\sin(z3(k)+z5(k))+z2(k)^2*m3*a2*\cos(z3(k))*\sin(z3(k) \\
+z5(k))*rCM3xx+z2(k)^2*m3*a2*\cos(z3(k))*\cos(z3(k)+z5(k))*rCM3yy \\
+z2(k)^2*m3*\cos(z3(k)+z5(k))*rCM3xx^2*\sin(z3(k)+z5(k))-z2(k)^2*m3*rCM3xx*rCM3yy \\
-z2(k)^2*m3*rCM3yy^2*\cos(z3(k)+z5(k))*\sin(z3(k)+z5(k))+2*z2(k)^2*m3*a3*\cos(z3(k) \\
+z5(k))^2*rCM3yy+z2(k)^2*m3*a3^2*\cos(z3(k)+z5(k))*\sin(z3(k)+z5(k)) \\
-z2(k)^2*m3*a3*rCM3yy+z2(k)^2*I3xy+m3*a2*\cos(z3(k))*z4(k)^2*\sin(z3(k) \\
+z5(k))*rCM3xx+m3*a2*\cos(z3(k))*z4(k)^2*\cos(z3(k)+z5(k))*rCM3yy \\
+m3*a2*\sin(z3(k))*z4(k)^2*\sin(z3(k)+z5(k))*rCM3yy-m3*a2*\sin(z3(k))*z4(k)^2*a3*\cos(z3(k) \\
+z5(k))+m3*a2*\cos(z3(k))*z4(k)^2*a3*\sin(z3(k)+z5(k))-m3*a2*\sin(z3(k))*z4(k)^2*\cos(z3(k) \\
+z5(k))*rCM3xx+z2(k)^2*I3yy*\cos(z3(k)+z5(k))*\sin(z3(k)+z5(k))*RT3;
\end{aligned}$$

APÉNDICE F

TÉRMINOS DE GRAVEDAD: G_j

$$Gz1=G1=0;$$

$$Gz2=g*(m2*(a2*\cos(z3(k))+\cos(z3(k))*rCM2xx-\sin(z3(k))*rCM2yy) \\ +m3*(a3*\cos(z3(k)+z5(k))+a2*\cos(z3(k))+\cos(z3(k)+z5(k))*rCM3xx-\sin(z3(k)+z5(k))*rCM3yy))*RT2;$$

$$Gz3=g*m3*(a3*\cos(z3(k)+z5(k))+\cos(z3(k)+z5(k))*rCM3xx-\sin(z3(k)+z5(k))*rCM3yy)*RT3;$$

APLICACIÓN DE TÉCNICAS DE CONTROL NO LINEAL A UN ROBOT ESFÉRICO DE 3 GRADOS DE LIBERTAD

Rodney Atalaya Chávez

Universidad Nacional de Ingeniería

Escuela de Postgrado de la Facultad de Ingeniería Eléctrica y Electrónica

Resumen— Este trabajo investiga la aplicación de técnicas avanzadas de control no lineal para el seguimiento a una trayectoria angular en un robot esférico de 3 grados de libertad. Las técnicas consideradas son linealización exacta por realimentación de estados (CLERE) y control predictivo generalizado no lineal (CPGNL).

El modelo dinámico del robot es obtenido usando la formulación de Euler-Lagrange.

Se demuestra que el uso de una señal de control optimizada fuera de línea (off-line) en el método CPGNL permite alcanzar un tiempo de establecimiento menor y mejora el rechazo a perturbaciones externas.

Temas claves—Control predictivo generalizado no lineal, control por linealización exacta por realimentación, Optimización y Robótica.

I. INTRODUCCIÓN

En los últimos años se ha observado una tendencia hacia el desarrollo de sistemas de manufactura flexible [1]. En este último se busca que el trabajo de reprogramación, para desarrollar nuevas tareas, sea lo menor posible. Los robots industriales son componentes importantes dentro de las celdas de manufactura flexible [1,2]. Éstos realizan diversas tareas tales como soldadura, posicionamiento de cargas, pintura, etc. Para realizar estas tareas eficientemente y con alta precisión muchos enfoques de control han sido propuestos en los últimos años. Uno de los controladores ampliamente difundidos es el PID [3]. Sin embargo, para propósitos de manufactura flexible el control PID no es el más adecuado [3]. Robots manipuladores seriales con muchos componentes vinculados presentan una dinámica altamente no lineal. El fuerte acoplamiento entre las articulaciones dificulta la tarea de sintonización del control PID. Por consiguiente, estrategias de control que hacen uso de la dinámica no lineal del sistema se perfilan como la mejor alternativa para alcanzar un mejor desempeño en sistemas de manufactura flexible.

Entre las estrategias de control más usadas para el seguimiento de trayectoria mencionamos *Control por Linealización Exacta por Realimentación de Estados* (CLERE) y *Control Predictivo no Lineal* (CPNL). El primero, requiere conocer de manera exacta el modelo dinámico del robot. Sin embargo, incertidumbres asociadas a parámetros dinámicos; e.g., cambios en la carga útil o fricción limitan el rango de aplicabilidad del CLERE [4]. CPNL ha recibido especial atención en los últimos años debido a su robustez para tratar sistemas con incertidumbres en el modelado. Sin embargo, su implementación requiere resolver un problema de optimización no lineal en cada instante de muestreo. Esto hace

que el CPNL sea principalmente usado en el control de sistemas altamente no lineales de dinámica lenta, como los que se encuentra en la industria química (control de PH) o procesamiento de petróleo.

El presente artículo propone una estrategia de control eficiente llamado *Control Predictivo Generalizado No Lineal* (CPGNL) para sistemas no lineales de dinámica rápida (robots). La característica principal del CPGNL es que no se requiere resolver el problema de optimización en cada instante de muestreo.

CLERE y CPGNL son numéricamente implementados para el control de seguimiento de trayectoria angular para un robot manipulador esférico de 3GDL, cuya morfología es robot PUMA de UNIMATION.

Este artículo está organizado como sigue. Sección 2 presenta el modelo dinámico del robot. En sección 3 se desarrolla el CPGNL. En la sección 4 se presentan resultados de las simulaciones usando CPGNL y se compara con los del CLERE. En la sección 5, se concluye con algunas observaciones.

II. MODELAMIENTO DINÁMICO DEL ROBOT

En robótica se hace uso de sistemas de coordenadas adecuados los cuales facilitan la descripción del movimiento geométrico y el análisis dinámico del robot.

Los sistemas de coordenadas han sido seleccionados usando la convención de Denavit – Hartenberg (D–H) [5]. La relación entre los sistemas coordenados $i-1$ e i está dada por:

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

donde c y s corresponden a $\cos(\cdot)$ y $\sin(\cdot)$, respectivamente.

El modelo dinámico del robot es obtenido por medio de las ecuaciones de Euler–Lagrange

$$\frac{d}{dt} \frac{\partial L}{\partial \dot{q}_i} - \frac{\partial L}{\partial q_i} = \tau_i, \quad i = 1, \dots, m, \quad (2)$$

donde q_i el ángulo de giro, \dot{q}_i la velocidad de la articulación i relativa al eslabón $i-1$, τ_i es el torque en la articulación i y m es el número de grados de libertad.

Se define L es el Lagrangiano definido por

$$L = T - U, \quad (3)$$

T y U son la energía cinética y potencial del sistema respectivamente, definidas por

$$T = \sum_{i=1}^m T_i \quad U = \sum_{i=1}^m U_i, \quad (4)$$

T_i y U_i son la energía cinética y potencial del eslabón i respectivamente.

Si el eslabón aumentado i es definido como la combinación del motor $i+1$ y del eslabón i , entonces la energía cinética del eslabón aumentado i es [8,9]

$$T_i = \frac{1}{2} m_i \dot{\mathbf{p}}_i^i T \dot{\mathbf{p}}_i^i + \dot{\mathbf{p}}_i^i T \mathbf{S}(\omega_i^i) m_i \mathbf{r}_{i,C_i}^i + \frac{1}{2} \omega_i^i T \hat{\mathbf{I}}_i^i \omega_i^i + k_{r,i+1} \dot{q}_{i+1} I_{m_{i+1}} \mathbf{z}_{m_{i+1}}^i T \omega_i^i + \frac{1}{2} k_{r,i+1} \dot{q}_{i+1}^2 I_{m_{i+1}} \quad (5)$$

Donde:

m_i : Es la masa del eslabón aumentado i .

$\dot{\mathbf{p}}_i^i$: Es la velocidad lineal del eslabón i referido al sistema coordenado i .

ω_i^i : La velocidad angular del eslabón i expresado con respecto al sistema i .

\mathbf{r}_{i,C_i}^i : Es el vector que inicia en el origen del sistema i y termina en el centro de masa del eslabón aumentado i , referido al sistema i .

$k_{r,i+1}$: Reducción del motor $i+1$.

$I_{m_{i+1}}$: Tensor de inercia del motor $i+1$ relativo a su centro de masa.

$\mathbf{z}_{m_{i+1}}^i$: Es el vector unitario a lo largo del eje del motor $i+1$ referido al sistema i .

$\mathbf{S}(\cdot)$: Es un operador definido por el vector $r = [r_x \ r_y \ r_z]^T$,

$$\mathbf{S}(r) = \begin{bmatrix} 0 & -r_z & r_y \\ r_z & 0 & -r_x \\ -r_y & r_x & 0 \end{bmatrix}^T \quad (6)$$

$\hat{\mathbf{I}}_i^i$: Tensor de inercia del eslabón aumentado i con respecto al origen del sistema i , viene dado por:

$$\hat{\mathbf{I}}_i^i = \begin{bmatrix} \hat{I}_{ixx} & -\hat{I}_{ixy} & -\hat{I}_{ixz} \\ -\hat{I}_{ixy} & \hat{I}_{iyy} & -\hat{I}_{iyz} \\ -\hat{I}_{ixz} & -\hat{I}_{iyz} & \hat{I}_{izz} \end{bmatrix} \quad (7)$$

La energía potencial del eslabón aumentado i está dada

$$U_i = -g_0^i T (m_i \mathbf{p}_i^i + m_i \mathbf{r}_{i,C_i}^i) \quad (8)$$

Donde \mathbf{p}_i^i el vector posición del link i referido al sistema i y $g_0^i T$ el vector de gravedad referido al sistema i .

Finalmente haciendo uso de la ecuación (2) se obtiene el modelo dinámico del robot, tal como se muestra en la ecuación siguiente

$$[\tau]_{m \times 1} = [H]_{m \times m} [\ddot{q}]_{m \times 1} + [C]_{m \times 1} + [G]_{m \times 1}, \quad (9)$$

donde H , es la matriz de inercia, C es el vector de coriolis, G es el vector de gravedad.

III. CONTROL PREDICTIVO GENERALIZADO NO LINEAL

El control predictivo generalizado no lineal CPGNL [10,11] ha sido planteado en tiempo continuo y se ha llevado a cabo siguiendo la estrategia de horizonte decreciente, cuyos principios son resumidos a continuación,

1. Predecir la salida por encima de un intervalo de tiempo futuro (horizonte de predicción H_p).
2. Conocido el setpoint futuro seleccionamos un conjunto de señales de control futura que minimizan una función de costo que está en función de los errores futuros entre la salida predicha y el setpoint futuro.
3. El primer elemento de este conjunto de señales de control $u(t)$, se envía al sistema.

Los pasos anteriores son repetidos en cada instante de muestreo.

A. Descripción del sistema

El control predictivo generalizado no lineal (CPGNL) considera la dinámica de un sistema multivariable representada en espacio-estado, por

$$\begin{aligned} \dot{x}(t) &= \mathbf{F}(x(t)) + \mathbf{G}(x(t))\mathbf{u} \\ y(t) &= \mathbf{h}(x(t)) \end{aligned}, \quad (10)$$

\mathbf{F} , \mathbf{G} y \mathbf{h} son diferenciables, $x \in \mathbf{R}^n$ es el vector de variables de estado, $\mathbf{u} \in \mathbf{R}^m$ es la señal de control, $y \in \mathbf{R}^m$ es la salida controlada.

B. Predicción de la salida

La salida del sistema no lineal para un marco de predicción $\tau \in [0, T_2]$, donde T_2 es el horizonte de predicción, es aproximada por la serie de Taylor

$$\begin{aligned} \hat{y}(t + \tau) &\approx \hat{y}^{[0]}(t) + \tau \hat{y}^{[1]}(t) + \frac{\tau^2}{2!} \hat{y}^{[2]}(t) \\ &+ \frac{\tau^\rho}{\rho!} \hat{y}^{[\rho]}(t) + \dots + \frac{\tau^{(\rho+r)}}{(\rho+r)!} \hat{y}^{[\rho+r]}(t) \end{aligned}, \quad (11)$$

donde ρ es grado relativo del sistema [12] y r es cualquier número entero positivo. El grado relativo viene dado por el menor orden de la derivada temporal de la salida 'y' tal que la señal de control \mathbf{u} aparezca.

Las derivadas temporales $\hat{y}^{[i]}(t)$ requeridas en la ecuación (11) vienen dadas por,

$$\begin{aligned}
\hat{y}^{[0]}(t) &= L_F^0 \mathbf{h}(x) = \mathbf{h}(x) \\
\hat{y}^{[1]}(t) &= L_F \mathbf{h}(x) \\
&\vdots \\
\hat{y}^{[\rho-1]}(t) &= L_F^{\rho-1} \mathbf{h}(x) \\
\hat{y}^{[\rho]}(t) &= L_F^\rho \mathbf{h}(x) + L_G L_F^{\rho-1} \mathbf{h}(x) \hat{\mathbf{u}}(t) \\
\hat{y}^{[\rho+1]}(t) &= L_F^{\rho+1} \mathbf{h}(x) + p_1(x, \hat{\mathbf{u}}) + L_G L_F^\rho \mathbf{h}(x) \hat{\mathbf{u}}^{[1]}(t) \\
&\vdots \\
\hat{y}^{[\rho+r]}(t) &= L_F^{\rho+r} \mathbf{h}(x) + p_r(x, \hat{\mathbf{u}}, \hat{\mathbf{u}}^{[1]}, \dots, \hat{\mathbf{u}}^{[r-1]}) + L_G L_F^{\rho-1} \mathbf{h}(x) \hat{\mathbf{u}}^{[r]}(t)
\end{aligned} \tag{12}$$

donde, $L_F \mathbf{h}(x) = \frac{\partial \mathbf{h}}{\partial x} \mathbf{F}(x(t))$, $L_G L_F \mathbf{h}(x) = \frac{\partial L_F \mathbf{h}(x)}{\partial x} \mathbf{G}(x(t))$,

$\hat{\mathbf{u}}^{[i]}(t) = \frac{d^i \hat{\mathbf{u}}(t)}{dt^i}$ y los términos de $p_i(x, \hat{\mathbf{u}}, \dots, \hat{\mathbf{u}}^{[i-1]})$ son funciones no lineales de alta complejidad [13].

Las ecuaciones anteriores se arreglan como se muestra a continuación,

$$\hat{Y} = \begin{bmatrix} \hat{y}^{[0]}(t) \\ \vdots \\ \hat{y}^{[\rho]}(t) \\ \vdots \\ \hat{y}^{[\rho+r]}(t) \end{bmatrix} = \begin{bmatrix} L_F^0 \mathbf{h}(x) \\ \vdots \\ L_F^\rho \mathbf{h}(x) \\ \vdots \\ L_F^{\rho+r} \mathbf{h}(x) \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \vdots \\ H(\hat{\mathbf{u}}) \end{bmatrix} \tag{13}$$

donde,

$$H(\hat{\mathbf{u}}) = \begin{bmatrix} L_G L_F^{\rho-1} \mathbf{h}(x) \hat{\mathbf{u}}^{[0]}(t) \\ p_1(x, \hat{\mathbf{u}}) + L_G L_F^{\rho-1} \mathbf{h}(x) \hat{\mathbf{u}}^{[1]}(t) \\ \vdots \\ p_r(x, \hat{\mathbf{u}}, \hat{\mathbf{u}}^{[1]}, \dots, \hat{\mathbf{u}}^{[r-1]}) + L_G L_F^{\rho-1} \mathbf{h}(x) \hat{\mathbf{u}}^{[r]}(t) \end{bmatrix}. \tag{14}$$

Usando la ecuación (13), la ecuación (11) se reescribe como

$$\hat{y}(t + \tau) \approx T(\tau) \hat{Y}(t), \tag{15}$$

donde

$$T(\tau) = \left[\mathbf{I} \bar{\tau} \frac{\bar{\tau}^2}{2!} \dots \frac{\bar{\tau}^{\rho+r}}{(\rho+r)!} \right], \quad \bar{\tau} = \text{diag}[\tau \ \tau \ \dots \ \tau] \in \mathbf{R}^{m \times m} \tag{16}$$

$\mathbf{I} \in \mathbf{R}^{m \times m}$, es la matriz identidad.

El vector $\bar{\mathbf{u}} = [\hat{\mathbf{u}}^{[0]}(t) \ \dots \ \hat{\mathbf{u}}^{[r]}(t)] \in \mathbf{R}^{m(r+1)}$ se define como el conjunto de señales de control necesarias para calcular la predicción de la salida en un marco de predicción $\tau \in [0, T_2]$.

La ecuación (15) predice las salidas del sistema multivariable ante el conjunto de señales de control $\bar{\mathbf{u}}$.

C. Trayectoria de referencia futura

El objetivo del control es conducir la salida predicha a lo largo de una trayectoria de referencia futura \hat{w} , ésta puede ser reescrita de manera análoga a (15),

$$\hat{w}(t + \tau) \approx T(\tau) \hat{W}(t), \tag{17}$$

donde $\hat{W}(t) = [w^{[0]}(t) \ \dots \ w^{[\rho]}(t) \ \dots \ w^{[\rho+r]}(t)]^T$

D. Minimización de la Función de Costo

Todo algoritmo de control predictivo contempla el uso de una función de costo (18) de índice cuadrático que penaliza el error entre la salida predicha \hat{y} y la trayectoria de referencia futura \hat{w} ,

$$J[\bar{\mathbf{u}}] = \frac{1}{2} \int_0^{T_2} (\hat{y}(t + \tau) - \hat{w}(t + \tau))^T (\hat{y}(t + \tau) - \hat{w}(t + \tau)) d\tau, \tag{18}$$

La cual puede ser reescrita usando las ecuaciones (15) y (17),

$$J[\bar{\mathbf{u}}] = \frac{1}{2} (\hat{Y}(t) - \hat{W}(t))^T \bar{\Gamma} (\hat{Y}(t) - \hat{W}(t)), \tag{19}$$

donde

$$\begin{aligned}
\bar{\Gamma} &= \begin{bmatrix} \bar{\Gamma}_{(1,1)} & \dots & \bar{\Gamma}_{(1,\rho+r+1)} \\ \vdots & \ddots & \vdots \\ \bar{\Gamma}_{(\rho+r+1,1)} & \dots & \bar{\Gamma}_{(\rho+r+1,\rho+r+1)} \end{bmatrix}_{m(\rho+r+1) \times m(\rho+r+1)} \\
\bar{\Gamma}_{(i,j)} &= \frac{\psi^{i+j-1}}{(i-1)!(j-1)!(i+j-1)}, \quad \psi = \text{diag}(T_2, \dots, T_2) \in \mathbf{R}^{m \times m}
\end{aligned} \tag{20}$$

$$\hat{Y}(t) - \hat{W}(t) = \begin{bmatrix} L_F^0 \mathbf{h}(x) \\ \vdots \\ L_F^{\rho+r} \mathbf{h}(x) \end{bmatrix} - \begin{bmatrix} w^{[0]}(t) \\ \vdots \\ w^{[\rho]}(t) \end{bmatrix} + \begin{bmatrix} 0 \\ \vdots \\ H(\hat{\mathbf{u}}) \end{bmatrix} = M(x(t)) + \begin{bmatrix} 0 \\ \vdots \\ H(\hat{\mathbf{u}}) \end{bmatrix} \tag{21}$$

La señal de control $\bar{\mathbf{u}} = [\hat{\mathbf{u}}^{[0]} \ \dots \ \hat{\mathbf{u}}^{[r]}(t)] \in \mathbf{R}^{m(r+1)}$ que minimiza la función de costo (18) se obtiene a partir de la siguiente condición necesaria,

$$\frac{\partial J}{\partial \bar{\mathbf{u}}} = 0. \tag{22}$$

Considerando que la matriz $\bar{\Gamma}$ se puede particionar en sub matrices tal como,

$$\bar{\Gamma} = \begin{bmatrix} \bar{\Gamma}_{\rho\rho} & \bar{\Gamma}_{\rho r} \\ \bar{\Gamma}_{r\rho} & \bar{\Gamma}_{rr} \end{bmatrix} \tag{23}$$

donde

$$\bar{\Gamma}_{rr} = \begin{bmatrix} \bar{\Gamma}_{(\rho+1,\rho+1)} & \dots & \bar{\Gamma}_{(\rho+1,\rho+r+1)} \\ \vdots & \ddots & \vdots \\ \bar{\Gamma}_{(\rho+r+1,\rho+1)} & \dots & \bar{\Gamma}_{(\rho+r+1,\rho+r+1)} \end{bmatrix}, \quad \bar{\Gamma}_{\rho r} = \begin{bmatrix} \bar{\Gamma}_{(1,\rho+1)} & \dots & \bar{\Gamma}_{(1,\rho+r+1)} \\ \vdots & \ddots & \vdots \\ \bar{\Gamma}_{(\rho,\rho+1)} & \dots & \bar{\Gamma}_{(\rho,\rho+r+1)} \end{bmatrix}$$

La ecuación (22) conduce a

$$\frac{\partial}{\partial \bar{\mathbf{u}}} \left(M(x) + \begin{bmatrix} 0 \\ \vdots \\ H(\hat{\mathbf{u}}) \end{bmatrix} \right)^T \begin{bmatrix} \bar{\Gamma}_{\rho\rho} & \bar{\Gamma}_{\rho r} \\ \bar{\Gamma}_{r\rho} & \bar{\Gamma}_{rr} \end{bmatrix} \left(M(x) + \begin{bmatrix} 0 \\ \vdots \\ H(\hat{\mathbf{u}}) \end{bmatrix} \right) = 0 \tag{24}$$

$$\begin{bmatrix} 0 \end{bmatrix}_{1 \times m\rho}^T \left(\frac{\partial}{\partial \bar{\mathbf{u}}} H(\hat{\mathbf{u}}) \right)^T \begin{bmatrix} \bar{\Gamma}_{\rho\rho} & \bar{\Gamma}_{\rho r} \\ \bar{\Gamma}_{r\rho} & \bar{\Gamma}_{rr} \end{bmatrix} \left(M(x) + \begin{bmatrix} 0 \\ \vdots \\ H(\hat{\mathbf{u}}) \end{bmatrix} \right) = 0 \tag{25}$$

$$\left(\frac{\partial}{\partial \bar{\mathbf{u}}} H(\hat{\mathbf{u}})\right)^T \left(\begin{bmatrix} \bar{\Gamma}_{\rho r}^T & | & \bar{\Gamma}_{rr} \end{bmatrix} M(x) + \bar{\Gamma}_{rr} H(\hat{\mathbf{u}}) \right) = 0 \quad (26)$$

Diferenciando la ecuación (14) con respecto a $\bar{\mathbf{u}}$, se obtiene

$$\frac{\partial}{\partial \bar{\mathbf{u}}} H(\hat{\mathbf{u}}) = \begin{bmatrix} L_g L_f^{\rho-1} \mathbf{h}(x) & 0_{m \times m} & 0_{m \times m} & \cdots & 0_{m \times m} \\ 0_{m \times m} & L_g L_f^{\rho-1} \mathbf{h}(x) & 0_{m \times m} & \cdots & 0_{m \times m} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0_{m \times m} & 0_{m \times m} & 0_{m \times m} & \cdots & L_g L_f^{\rho-1} \mathbf{h}(x) \end{bmatrix} \quad (27)$$

donde $0_{m \times m}$ denota una matriz nula de orden $m \times m$.

De acuerdo a la definición de grado relativo se tiene que

$L_g L_f^{\rho-1} \mathbf{h}(x)$ es invertible por lo tanto $\frac{\partial}{\partial \bar{\mathbf{u}}} H(\hat{\mathbf{u}})$ es invertible.

esto permite escribir las ecuaciones (22) y (26) como,

$$\begin{bmatrix} \bar{\Gamma}_{\rho r}^T & | & \bar{\Gamma}_{rr} \end{bmatrix} M(x) + \bar{\Gamma}_{rr} H(\hat{\mathbf{u}}) = 0, \quad (28)$$

y finalmente la ecuación anterior puede ser reescrita como,

$$H(\hat{\mathbf{u}}) = \begin{bmatrix} L_g L_f^{\rho-1} \mathbf{h}(x) \hat{\mathbf{u}}^{[0]}(t) \\ \vdots \end{bmatrix} = - \begin{bmatrix} \bar{\Gamma}_{rr}^{-1} \bar{\Gamma}_{\rho r}^T & | & \mathbf{I} \end{bmatrix} \begin{bmatrix} L_f^0 \mathbf{h}(x) - w^{[0]}(t) \\ \vdots \\ L_f^{\rho-1} \mathbf{h}(x) - w^{[\rho-1]}(t) \\ \vdots \end{bmatrix} \quad (29)$$

donde la matriz $\mathbf{I} \in \mathbf{R}^{m(r+1) \times m(r+1)}$ es la identidad.

La estrategia de horizonte decreciente establece que de las componentes de la señal de control $\bar{\mathbf{u}} = [\hat{\mathbf{u}}^{[0]}(t) \ \dots \ \hat{\mathbf{u}}^{[r]}(t)]$ sólo se escoge la primera, es decir $\hat{\mathbf{u}}^{[0]}(t)$.

Por lo tanto la señal de control escogida y enviada al sistema es

$$\hat{\mathbf{u}}^{[0]}(t) = - \left(L_g L_f^{\rho-1} \mathbf{h}(x) \right)^{-1} \left\{ \mathbf{K} M_p + L_f^{\rho} \mathbf{h}(x) - w^{[\rho]}(t) \right\} \quad (30)$$

donde

\mathbf{K} es la matriz formada por las primeras m filas y las primeras $m\rho$ columnas de $\bar{\Gamma}_{rr}^{-1} \bar{\Gamma}_{\rho r}^T$,

$$M_p = \begin{bmatrix} \left(\mathbf{h}(x) - w^{[0]}(t) \right)^T & \left(L_f^1 \mathbf{h}(x) - w^{[1]}(t) \right)^T & \cdots & \left(L_f^{\rho-1} \mathbf{h}(x) - w^{[\rho-1]}(t) \right)^T \end{bmatrix}.$$

IV. SIMULACIONES

Figura 1 muestra un manipulador esférico de 3 grados de libertad cuya morfología es robot PUMA de UNIMATION.

TABLA I presenta los parámetros Denavit Hartenberg del brazo robótico.

La energía cinética y potencial de cada eslabón aumentado se calcula de las ecuaciones (5) y (8), luego el torque aplicado a cada articulación se obtiene usando la ecuación de Euler-Lagrange (2).

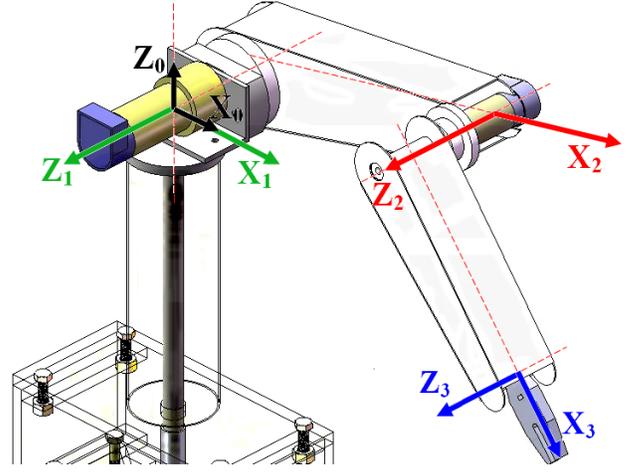


Fig. 1. Diagrama de un manipulador esférico de 3 grados de libertad. Los sistemas de coordenadas solidarios a cada eslabón siguen la convención Denavit-Hartenberg.

El modelo dinámico del robot incluyendo la dinámica de los motores es

$$\begin{bmatrix} u \\ u \\ u \end{bmatrix} = \begin{bmatrix} R_{T1}(H_{11} + J_{eq1}) & R_{T1}H_{12} & R_{T1}H_{13} \\ R_{T2}H_{21} & R_{T2}(H_{22} + J_{eq2}) & R_{T2}H_{23} \\ R_{T3}H_{31} & R_{T3}H_{32} & R_{T3}(H_{33} + J_{eq3}) \end{bmatrix} \begin{bmatrix} \ddot{q}_1 \\ \ddot{q}_2 \\ \ddot{q}_3 \end{bmatrix}, \quad (31)$$

$$+ \begin{bmatrix} (R_{T1}B_{eq1} + N_{T1})\dot{q}_1 + R_{T1}C_1 \\ (R_{T2}B_{eq2} + N_{T2})\dot{q}_2 + R_{T2}C_2 \\ (R_{T3}B_{eq3} + N_{T3})\dot{q}_3 + R_{T3}C_3 \end{bmatrix} + \begin{bmatrix} R_{T1}G_1 \\ R_{T2}G_2 \\ R_{T3}G_3 \end{bmatrix}$$

donde u es el voltaje entregado a los motores, H_{ij} , C_i y G_i son los elementos de la matriz de inercia, coriolis y gravedad

respectivamente, además $R_{Ti} = \frac{R_i}{K_{ri} K_{mi} K_i}$, $N_{Ti} = \frac{K_{ri} K_{bi}}{K_i}$ y

R_i , K_{ri} , K_{mi} , K_i , K_{bi} son constantes del motor.

TABLA I
PARÁMETROS DE DENAVIT HARTENBERG

Elemento	α_i	θ_i	d_i	a_i
1	$\pi/2$	q_1	0	0
2	0	q_2	$-d_2$	a_2
3	0	q_3	d_3	a_3

La ecuación (31) expresada en espacio de estado con variables $x_1 = q_1$, $x_2 = q_2$, $x_3 = q_3$, $x_4 = \dot{q}_1$, $x_5 = \dot{q}_2$, $x_6 = \dot{q}_3$ viene dado

$$\begin{aligned} \dot{x}(t) &= f(x) + g(x)u(t) \\ y(t) &= \mathbf{h}(x) = [x_1 \ x_2 \ x_3]^T \end{aligned}, \quad (32)$$

donde

$$f(x) = [x_4 \ x_5 \ x_6 \ F_4 \ F_5 \ F_6]^T$$

$$g(x) = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ G_{41} & G_{42} & G_{43} \\ G_{51} & G_{52} & G_{53} \\ G_{61} & G_{62} & G_{63} \end{bmatrix}, u = [u_1 \ u_2 \ u_3]^T$$

La ecuación (32) nos indica que tenemos un sistema no lineal multivariable de 3 entradas u_i y 3 salidas y_i . $F_i(x)$, $G_{ij}(x)$ son funciones no lineales que dependen de los parámetros dinámicos del robot [8].

La predicción de la salida se obtiene usando la ecuación (11). Ahora debemos hallar el grado relativo del sistema, para ello derivamos la salida $\mathbf{h}(x)$ hasta un orden en que aparezca la señal de control u_i

$$\dot{y} = \frac{\partial \mathbf{h}(x)}{\partial x} \mathbf{f} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_4 \\ x_5 \\ x_6 \\ F_4 \\ F_5 \\ F_6 \end{bmatrix} = \begin{bmatrix} x_4 \\ x_5 \\ x_6 \end{bmatrix} = L_f \mathbf{h}(x) \quad (33)$$

$$\ddot{y} = \begin{bmatrix} F_4 \\ F_5 \\ F_6 \end{bmatrix} + \begin{bmatrix} G_{41} & G_{41} & G_{41} \\ G_{51} & G_{51} & G_{51} \\ G_{61} & G_{61} & G_{61} \end{bmatrix} u = L_f^2 \mathbf{h}(x) + L_g L_f \mathbf{h}(x) u \quad (34)$$

De (34) se concluye que el sistema es de grado relativo $\rho = 2$ y además consideramos $r = 2$. Esto significa que la salida predicha del sistema (11) será aproximada por una serie de Taylor de 5 términos.

Luego hallamos las sub matrices $\bar{\Gamma}_{rr}$, $\bar{\Gamma}_{\rho r}$ de la matriz $\bar{\Gamma}$ (23) cuyos elementos vienen dados por la ecuación (20)

$$\bar{\Gamma}_{rr} = \bar{\Gamma}_{22} = \begin{bmatrix} \bar{\Gamma}_{3,3} & \bar{\Gamma}_{3,4} & \bar{\Gamma}_{3,5} \\ \bar{\Gamma}_{4,3} & \bar{\Gamma}_{4,4} & \bar{\Gamma}_{4,5} \\ \bar{\Gamma}_{5,3} & \bar{\Gamma}_{5,4} & \bar{\Gamma}_{5,5} \end{bmatrix} \in \mathbf{R}^{9 \times 9} \quad (35)$$

$$\bar{\Gamma}_{\rho r} = \bar{\Gamma}_{22} = \begin{bmatrix} \bar{\Gamma}_{1,3} & \bar{\Gamma}_{1,4} & \bar{\Gamma}_{1,5} \\ \bar{\Gamma}_{2,3} & \bar{\Gamma}_{2,4} & \bar{\Gamma}_{2,5} \end{bmatrix} \in \mathbf{R}^{6 \times 9} \quad (36)$$

La matriz \mathbf{K} formada por las primeras 3 filas y las primeras 6 columnas de $\bar{\Gamma}_{22}^{-1} \bar{\Gamma}_{22}^T$,

$$\bar{\Gamma}_{rr}^{-1} \bar{\Gamma}_{\rho r}^T = \begin{bmatrix} \frac{42}{T_2^2} & 0 & 0 & \frac{21}{2T_2} & 0 & 0 & \dots \\ 0 & \frac{42}{T_2^2} & 0 & 0 & \frac{21}{2T_2} & 0 & \dots \\ 0 & 0 & \frac{42}{T_2^2} & 0 & 0 & \frac{21}{2T_2} & \dots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix}$$

Entonces la matriz $\mathbf{K} \in \mathbf{R}^{3 \times 6}$ viene dado por,

$$\mathbf{K} = \begin{bmatrix} \frac{42}{T_2^2} & 0 & 0 & \frac{21}{2T_2} & 0 & 0 \\ 0 & \frac{42}{T_2^2} & 0 & 0 & \frac{21}{2T_2} & 0 \\ 0 & 0 & \frac{42}{T_2^2} & 0 & 0 & \frac{21}{2T_2} \end{bmatrix}. \quad (37)$$

El cómputo de la señal de control óptima se obtiene usando (30), y para nuestro caso viene dado por

$$[\hat{\mathbf{u}}^x]_{\text{opt}} = - (L_g L_f^{\rho-1} \mathbf{h}(x))^{-1} \left\{ \frac{42}{T_2^2} \begin{bmatrix} x_1 - w_1 \\ x_2 - w_2 \\ x_3 - w_3 \end{bmatrix} + \frac{21}{2T_2} \begin{bmatrix} \dot{x}_4 - w_1^{[1]} \\ \dot{x}_5 - w_2^{[1]} \\ \dot{x}_6 - w_3^{[1]} \end{bmatrix} + \begin{bmatrix} F_4 - w_1^{[2]} \\ F_5 - w_2^{[2]} \\ F_6 - w_3^{[2]} \end{bmatrix} \right\} \quad (38)$$

El control por linealización exacta por realimentación de estados y el control predictivo generalizado no lineal son implementados en el entorno de simulación de MATLAB.

El control predictivo generalizado no lineal es considerado para un horizonte de predicción $T_2 = 0.5$ s. y un tiempo de muestreo de 10 ms. El tiempo total en la simulación es 10 s.

TABLA II
CONDICIONES INICIALES

Eslabones	Condiciones Iniciales
Primer brazo	0
Segundo brazo	$-\pi/2$
Tercer brazo	0

Las condiciones iniciales se muestran en la TABLA II. Las variables de salida son las posiciones angulares de cada eslabón.

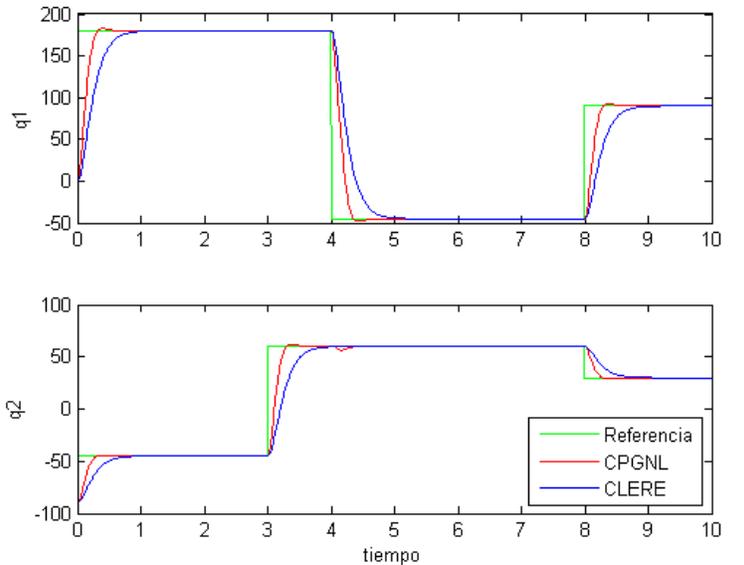


Fig. 2. Seguimiento a una trayectoria de referencia tipo step usando los controladores CPGNL y CLERE. Se observa que el CPGNL presenta un menor tiempo de establecimiento.

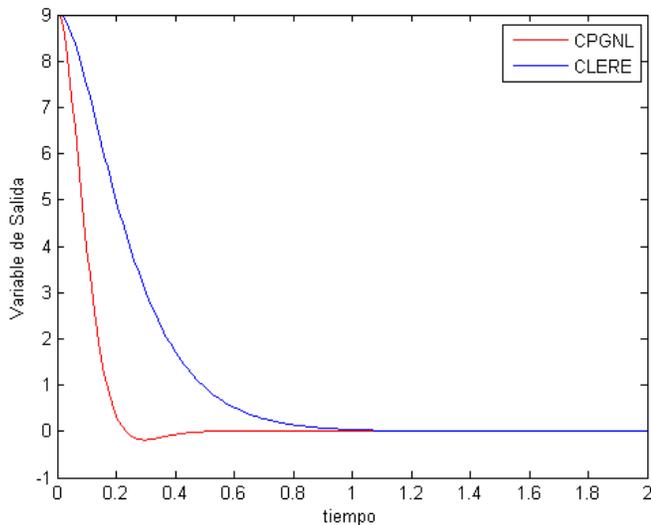


Fig.3 Rechazo del sistema a una perturbación de $q_1 = \pi/20$ usando el CPGNL y el CLERE. Se observa que el control predictivo permite rechazar la perturbación externa en un menor tiempo.

Figura 2 muestra los resultados del CPGNL y del CLERE. Se observa que el tiempo de establecimiento del CPGNL es menor.

Otra ventaja que se observa en las simulaciones del CPGNL frente al CLERE es su buena robustez contra el rechazo de perturbaciones externas. Para este test se aplicó un disturbio de $q_1 = \pi/20$ y los resultados son mostrados en la Fig. 3.

V. CONCLUSIONES

En este artículo se ha desarrollado la estrategia de control predictivo generalizado no lineal (CPGNL) e implementado para controlar un manipulador robótico de 3 grados de libertad. Además, hemos implementado el control por linealización exacta por realimentación de estados (CLERE) y comparado su desempeño con el CPGNL. Se concluye que el CPGNL permite a la salida controlada alcanzar tiempos de establecimiento menores.

En un trabajo futuro buscaremos obtener resultados experimentales que verifiquen estos resultados numéricos.

VI. REFERENCIAS

- [1] C. E. Flores. Sistema de Fabricación Flexible (FMS). México, 2004. Disponible en http://www.tec.url.edu.gt/boletin/URL_03_MEC02.pdf
- [2] Erick Vallejo R, "Sistemas Flexibles de Manufactura", Universidad del Norte, 3-4, 43-49, 1998.
- [3] A. Vivas, V. Mosquera, "Predictive Functional Control of a PUMA robot" presented ACSE 05 Conference 19-21, December 2005.
- [4] B. Siliciano, L. Sciavicco, *Robotics, Modelling, Planning and Control*. Italy: Springer, 2008, p. 39-62, 248-265.
- [5] Nikolaos A. Bompos, Panagiotis K. Artemiadis, "Modelling, full Identification and Control of the Mitsubishi PA-10 Robot Arm".
- [6] P. J. Gawthrop, H. Demircioglu and I. Siller-Alcalá, "Multivariable Continuous time Generalised Predictive Control: State Space Approach to linear and nonlinear Systems" IEE Proceedings Control Theory and Applications 1998.

[7] I. I. Siller-Alcalá "Nonlinear Continuous Time Generalized Predictive Control" Tesis para obtener el grado de Doctor en la Universidad de Glasgow, Escocia, 1998.

[8] Isidori Alberto. Nonlinear Control Systems, Second Edition, Springer-Verlag Berling, 1989, p. 146-150.