



UNIVERSIDAD AUTÓNOMA DE QUERÉTARO
FACULTAD DE INGENIERÍA
Maestría en ciencias en Instrumentación y Control Automático

DISEÑO DE UN CONTROLADOR POR LÓGICA DIFUSA APLICADO A
UN ROBOT MANIPULADOR

TESIS

Que como parte de los requisitos para obtener el grado de

**MAESTRO EN CIENCIAS EN INSTRUMENTACIÓN Y
CONTROL AUTOMÁTICO**

PRESENTA

MONTUFAR VILLAFUERTE FABIOLA

Expediente 207742

DIRIGIDO POR:

M EN C. ALFONSO NORIEGA PONCE

C.U. QUERÉTARO, QRO. MARZO DE 2012



Universidad Autónoma de Querétaro
 Facultad de Ingeniería
 Maestría en ciencias en Instrumentación y Control Automático

Diseño de un controlador por lógica difusa aplicado a un robot manipulador

TESIS

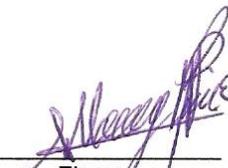
Que como parte de los requisitos para obtener el grado de
 Nm., Maestro en Ciencias en Instrumentación y Control Automático

Presenta:
 Montufar Villafuerte Fabiola

Dirigido por:
 M en C. Alfonso Noriega Ponce

SINODALES

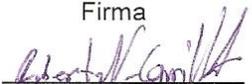
M. en C. Alfonso Noriega Ponce
 Presidente


 Firma

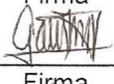
Dr. Víctor Manuel Hernández Guzmán
 Secretario


 Firma

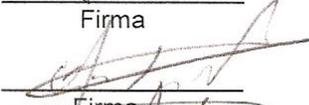
Dr. Roberto Valentín Carrillo Serrano
 Vocal


 Firma

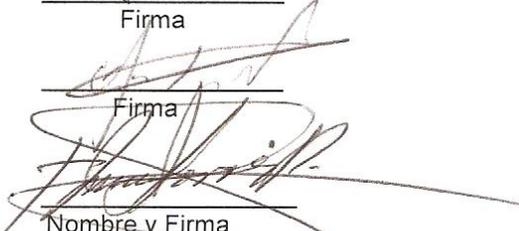
Dr. Luis Alfonso Franco Gasca
 Suplente


 Firma

Dr. Efrén Gorrostieta Hurtado
 Suplente


 Firma


 Nombre y Firma
 Dr. Aurelio Domínguez González
 Director de la Facultad


 Nombre y Firma
 Dr. Irineo Torres Pacheco
 Director de Investigación y Posgrado

RESUMEN

Debido a la variación de la carga y dinámicas no modeladas, el modelo de un robot manipulador es clasificado como un sistema dinámico no lineal. En este trabajo de investigación, se desarrolló un controlador lógico difuso, el cual es robusto a incertidumbres y perturbaciones. Se realizaron simulaciones y pruebas experimentales en un robot manipulador de dos grados de libertad. También se realizó un análisis metódico del controlador difuso desarrollado, basándose para esto, en los controladores de estructura variable. Se compararon los resultados obtenidos con controladores clásicos tales como el PID y el control PD compensado.

(Palabras clave: Lógica Difusa, Robustez, Robot Manipulador, Controladores de Estructura Variable)

SUMMARY

Due to variation of load and unmodeled dynamics, a robot manipulator is classified as a nonlinear dynamical system. In this research, we developed a fuzzy logic controller which is robust to uncertainties and disturbances. Simulations and experimental verification had been developed in a robot manipulator of two degrees of freedom. We also performed a methodical analysis of the fuzzy controller based on variable structure controllers. We compared the results obtained with classical controllers such as PID and PD control compensated.

(Key words: Fuzzy Logic, Robustness, Robot Manipulator, Variable Structure Controller)

AGRADECIMIENTOS

A Dios por guiarme en los momentos difíciles y por estar a mi lado cuando más lo he necesitado, por darme una familia cariñosa y amistades inigualables. A mis padres Isabel y Amilcar, gracias por el apoyo y amor que a lo largo de la vida me han brindado. Sobre todo gracias por confiar en mí y ayudarme en esta etapa de mi vida, sin ellos no sería la persona que soy ahora. A mis hermanas Jessica y Mercedes por su cariño y apoyo, gracias por sus consejos y ánimos cuando lo he necesitado.

A Víctor por brindarme su amor y cariño, por apoyarme en todo momento y regañarme cuando me desanimaba. A su familia por hacerme sentir como en casa.

Al M. en C. Alfonso Noriega Ponce, director de esta tesis, por sus consejos y conocimientos brindados de control inteligente y sin los cuales no se hubiera elaborado el proyecto de investigación aquí presentado. Al Dr. Víctor M. Hernández Guzmán por los conocimientos que compartió con nosotros en el área de control, por sus consejos y recomendaciones para mejorar este trabajo; así también por facilitarnos el uso de la plataforma de experimentación. Al Dr. Valentín Carrillo S. por sus consejos y sus conocimientos brindados durante el transcurso de la maestría, así también por su valiosa colaboración en la etapa de instrumentación del proyecto presentado. Al Dr. Luis A. Franco y al Dr. Efrén Gorrostieta por sus consejos, comentarios y tiempo brindado para mejorar este trabajo de investigación.

A mis compañeros de clase, en especial a Francisco y Anselmo, por los momentos divertidos y de compañerismo que pasamos a lo largo de la maestría.

Al Consejo Nacional de Ciencia y Tecnología por el apoyo brindado para realizar mis estudios de maestría. A la Universidad Autónoma de Querétaro por dar las herramientas necesarias para empezar y culminar con esta etapa.

ÍNDICE DE CONTENIDO

RESUMEN	i
SUMMARY	ii
AGRADECIMIENTOS	iii
ÍNDICE DE CONTENIDO	iv
ÍNDICE DE FIGURAS.....	vii
ÍNDICE DE TABLAS	ix
I INTRODUCCIÓN	1
1.1 Descripción del problema	1
1.2 Justificación	4
1.3 Objetivo general	4
1.4 Hipótesis.....	4
1.5 Alcances	5
1.6 Organización del trabajo.....	5
II REVISIÓN DE LITERATURA.....	7
2.1 Estado del arte.....	7
2.2 Conceptos básicos de lógica difusa	10
2.2.1 Conjuntos difusos	10

2.2.2	Funciones de pertenencia de conjuntos difusos	11
2.2.3	Variables lingüísticas y valores lingüísticos	13
2.2.4	Operaciones entre conjuntos difusos.....	13
2.2.5	Reglas difusas SI- ENTONCES.....	14
2.2.6	Controladores difusos	15
2.3	Control de robots manipuladores	19
2.3.1	Cinemática Directa	19
2.3.2	Cinemática inversa	20
2.3.3	Dinámica de manipuladores.....	20
III	METODOLOGÍA	24
3.1	Obtención del modelo dinámico del manipulador	24
3.1.1	Cálculo de la energía cinética	24
3.1.2	Cálculo de la energía potencial.....	27
3.1.3	Cálculo del Lagrangiano.....	27
3.2	Simulación del modelo encontrado	29
3.3	Diseño del controlador	32
3.3.1	Introducción	32
3.3.2	Sistemas difusos basados en controladores de estructura variable	33
3.3.3	Controlador difuso propuesto.....	35
3.3.4	Análisis del controlador lógico difuso	43
IV	RESULTADOS	54
4.1	Descripción de las pruebas realizadas	54
4.2	Controlador robusto difuso	55
4.2.1	Parámetros y ganancias del controlador.....	55
4.2.2	Simulaciones.....	58
4.2.3	Resultados experimentales	64
4.3	Control PD con compensación	68
4.3.1	Simulaciones.....	69

4.3.2	Resultados experimentales	71
4.4	Control PID	73
4.4.1	Simulaciones.....	74
4.4.2	Resultados experimentales	76
4.5	Comparación entre los algoritmos de control desarrollados.	78
V	CONCLUSIONES.....	81
VI	BIBLIOGRAFÍA.....	83
Apéndice A	Preliminares matemáticos y propiedades.....	88
A.1	Propiedades del modelo dinámico de robots manipuladores.....	88
A.2	Definiciones de álgebra lineal	92
Apéndice B	Sistemas de control de Estructura Variable (CEV).....	95
B.1	Fundamentos del CEV	95
B.2	Descripción matemática del CEV con MD	98
B.3	Reducción del ruido continuo.....	99
Apéndice C	CÓDIGOS DE PROGRAMAS.....	100
C.1	Código utilizado para las simulaciones en Matlab.	100
C.2	Código en Builder C++ para el controlador difuso.....	111
Apéndice D	Plataforma de experimentación	114
D.1	Descripción General del sistema	114

ÍNDICE DE FIGURAS

Figura	Página.
1.1 Sistema de control de manipulador basado en el modelo.....	3
2.1 Tipos de funciones de membresía: a) trapezoidal b) triangular c) singleton y d)gaussiana	12
2.2 Representación de la variable lingüística velocidad y sus valores lingüísticos	13
2.3 Esquema de control difuso.....	15
2.4 Método del centro de área	17
2.5 Método promedio de centros	18
3.1 Análisis geométrico del manipulador de 2 grados de libertad	25
3.2 Trayectoria obtenida con $\tau_1=0$ N-m y $\tau_2=0.15$ N-m.....	31
3.3 Trayectoria obtenida por Maldonado del toro, et al. (2011).....	31
3.4 Trayectoria correspondiente a $\tau_1=1.3$ N-m y $\tau_2=0$ N-m.....	31
3.5 Trayectoria obtenida por Maldonado del toro, et al. (2011) $\tau_1=1.3$ N-m y $\tau_2=0$ N-m.....	31
3.6 Trayectoria en el plano XY con $\tau_1=1.6$ N-m y $\tau_2=-0.3$ N-m.....	32
3.7 Trayectoria presentada por Maldonado del toro, et al. (2011) $\tau_1=1.6$ N-m y $\tau_2=-0.3$ N-m.....	32
3.8 Sistema de estructura variable	34
3.9 Esquema general del controlador.....	37
3.10 Funciones de pertenencia para la variable de entrada error de posición y error de velocidad	37
3.11 Funciones de pertenencia para la variable de entrada "error de velocidad"	38
3.12 Funciones de pertenencia para la salida	38
3.13 Funciones de membresía en el plano de fase del error.....	40
3.14 Reglas difusas y el plano de fase del error.....	44
3.15 Conjunto de puntos en el plano de fase del error normalizado	46
4.1 Diagrama a bloques del controlador robusto difuso	58
4.2 Simulación del CLD robusto para la articulación 1	59
4.3 Gráfica del error de posición, eslabón 1	59
4.4 Simulación CLD robusto eslabón 2.....	60
4.5 Error de seguimiento eslabón 2.....	60
4.6 Simulaciones variando la masa en el eslabón 2	61
4.7 Error de seguimiento articulación 1.....	62
4.8 Simulación articulación 2 con cambio de masa	62
4.9 Error de seguimiento articulación 2.....	63
4.10 Respuesta del eslabón 1	63
4.11 Respuesta del eslabón 2	64
4.12 Resultados experimentales de la articulación 1	65

4.13 Error de seguimiento articulación 1.....	65
4.14 Resultados experimentales articulación 2.....	66
4.15 Resultados experimentales con carga.....	66
4.16 Resultados experimentales articulación 2.....	67
4.17 Resultados experimentales articulación 1.....	68
4.18 Resultados experimentales articulación 2.....	68
4.19 Simulación PD compensado articulación 1.....	70
4.20 Simulación PD compensado articulación 2.....	70
4.21 Simulación PD compensado con variación de la masa 2.....	71
4.22 Simulación PD Compensado articulación 2.....	72
4.23 Resultados experimentales PD compensado articulación 1.....	72
4.24 Resultados experimentales PD compensado.....	73
4.25 Resultados experimentales PD compensado variación de masa.....	73
4.26 PD compensado variando la masa.....	74
4.27 Simulación PID articulación 1.....	75
4.28 Simulación PID articulación 2.....	75
4.29 PID con variación en la masa de eslabón 2.....	764
4.30 PID articulación 2.....	77
4.31 Resultados experimentales PID articulación 1.....	77
4.32 Resultados experimentales PID articulación 2.....	78
4.33 Gráfico comparativo resultados experimentales.....	79
4.34 Gráfico comparativo resultados experimentales prueba 2.....	80
B.1 Sistema de estructura variable.....	96
B.2 trayectorias en cadena cerrada del sistema.....	97
B.3 Las trayectorias del sistema como función del tiempo: (A) La condición inicial en la región I; (B) La condición inicial en la región II.....	97
D.1 Robot manipulador de 2 g.d.l.....	114
D.2 Esquema general para el control del robot.....	117
D.3 Etapa 1.....	117
D.4 Etapa 2 conversión de señales.....	118
D.5 Etapa 3 potencia.....	119

ÍNDICE DE TABLAS

Tabla	Página
<i>3-1 Parámetros del robot de 2 gdl.....</i>	<i>30</i>
<i>3-2 Base de reglas de un CLD.....</i>	<i>34</i>
<i>3-3 Reglas difusas con el modelo UNLP.....</i>	<i>41</i>
<i>4-1 Parámetros nominales utilizados del robot de 2 gdl.....</i>	<i>55</i>
<i>4-2 Valor parámetros nominales.....</i>	<i>57</i>
<i>4-3 Centros de las FM de entrada.....</i>	<i>57</i>
<i>4-4 Centros de las FM de salida.....</i>	<i>57</i>
<i>4-5 Ganancias del CLD.....</i>	<i>57</i>
<i>D-1 Parámetros calculados del robot manipulador.....</i>	<i>57</i>
<i>D-2 Especificaciones técnicas GM14904S014.....</i>	<i>57</i>
<i>D-3 Especificaciones técnicas GM8724S015.....</i>	<i>57</i>

I INTRODUCCIÓN

En el presente capítulo, se dará una descripción general del proyecto de investigación realizado. Para una mejor comprensión del tema se presenta el problema a resolver, los objetivos que se pretenden alcanzar, las razones por las cuales se decidió trabajar en este tema, así como, los alcances que tendrá el trabajo de investigación efectuado. Al final del capítulo se muestra una descripción de la estructura del presente documento de tesis.

1.1 Descripción del problema

Un robot manipulador industrial es una máquina formada por eslabones donde cada uno de éstos representa un grado de libertad; otra característica que posee este tipo de robots es su flexibilidad, ya que son reprogramables y de usos múltiples, por lo que son utilizados en tareas repetitivas y que requieren de precisión.

Las principales ventajas para el uso de robots manipuladores en la industria son la reducción de los costos de producción y el incremento de la precisión, la calidad y la productividad.

Para lograr que un robot realice una tarea, es necesario un dispositivo de control, el cual asegure que el mecanismo realizará de forma autónoma dicha actividad. Es importante resaltar que uno de los propósitos del controlador, además del mencionado anteriormente, es maximizar la velocidad de ejecución de la tarea y la repetitividad en la medida de lo que sea posible, tomando en cuenta las limitaciones del sistema físico.

El sistema de control, es entonces el cerebro del manipulador robótico, que dará instrucciones mandando las señales necesarias a los actuadores. Para realizar esto, existen diferentes esquemas de control clásico, como el control PD con pre-compensación, PID o el control PD con par calculado, los cuales utilizan parte de la dinámica del manipulador. La ecuación 1.1 explica esto con más detalle, describiendo la dinámica del robot manipulador.

$$\tau = M(q)\ddot{q} + C(q, \dot{q})\dot{q} + g(q) \quad (1.1)$$

donde: q , \dot{q} y \ddot{q} representan las posiciones, velocidades y aceleraciones de los eslabones respectivamente, $M(q)$ es la matriz de inercia de $n \times n$ donde n define el número de grados de libertad del manipulador, $C(q, \dot{q})$ es una matriz de $n \times n$ de términos centrífugos y de Coriolis, y $g(q)$ es un vector de $n \times 1$ de términos de gravedad.

Cada elemento de $M(q)$ y $g(q)$ depende de q , la posición de todas las articulaciones del manipulador, así como de la masa, inercia, y longitud de cada eslabón. Cada elemento de $C(q, \dot{q})$ es una función tanto de la posición, como de la velocidad. A su vez, cada matriz y vector $M(q)$, $g(q)$ y $C(q, \dot{q})$ están formados por funciones seno y coseno, lo que convierte al robot manipulador en un sistema no lineal.

Otra característica que posee la dinámica del manipulador, es que los parámetros tales como la masa e inercia pueden variar al agregarle carga al robot por ejemplo: al mover un objeto de un lugar a otro, o bien que la masa no sea estimada correctamente. Además, se podría incorporar un término de fricción, $F(q, \dot{q})$, suponiendo que éste es una función de las posiciones y las velocidades de las articulaciones, y agregarlo a la ecuación (1.1), por lo tanto el resultado sería la ecuación (1.2):

$$\tau = M(q)\ddot{q} + C(q, \dot{q})\dot{q} + g(q) + F(q, \dot{q}) \quad (1.2)$$

La mayoría de los autores resuelve el problema de controlar un sistema como el que describe la ecuación (1.2), utilizando un esquema de control compensado que reduce el sistema no lineal a uno lineal donde las no linealidades son compensadas -de ahí el nombre de control compensado-. La ecuación (1.2) se representa entonces como:

$$\tau = \alpha\tau' + \beta \quad (1.3)$$

donde

$$\begin{aligned}\alpha &= M(q) \\ \beta &= C(q, \dot{q})\dot{q} + g(q) + F(q, \dot{q})\end{aligned}\tag{1.4}$$

β representa un par calculado, por lo tanto, la ley de control a utilizar estaría dada únicamente por τ' :

$$\tau' = \ddot{q}_d + k_v \dot{\tilde{q}} + k_p \tilde{q}\tag{1.5}$$

donde q_d , \dot{q}_d y \ddot{q}_d corresponde a la posición, velocidad y aceleración deseada respectivamente y \tilde{q} es el error de posición definido como:

$$\tilde{q} = q_d - q\tag{1.6}$$

por ello, el sistema de control resultante es el que se muestra en la figura 1.1.

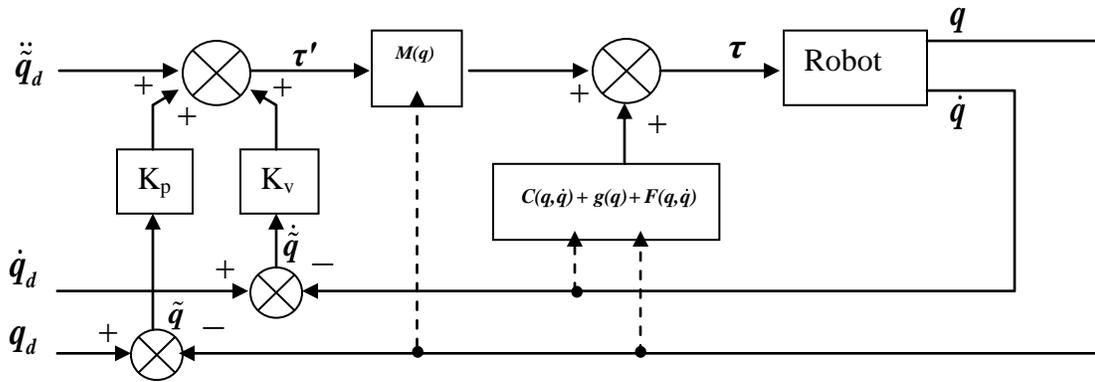


Figura 1.1 Sistema de control de manipulador basado en el modelo

A simple vista, se observa que el controlador cumple con su objetivo, reduciendo el sistema a uno lineal de segundo orden, sin embargo, el rendimiento ideal de este tipo de controlador no se obtiene en la práctica por diversas razones, entre las cuales se puede mencionar el hecho de que es necesario un conocimiento exacto de las masas, inercias y longitudes de cada eslabón, es decir, un modelo matemático perfecto del modelo real. Este modelo es algo imposible de obtener ya que algunos factores no se pueden conocer; además se debe suponer que el sistema es invariante en el tiempo, sin embargo, si se requiere que el robot tome un objeto, la masa de los eslabones cambia, por lo tanto, habrá un error en la compensación y, por ende, un error en la posición (la tarea a realizar tendrá un margen de error).

1.2 Justificación

La principal dificultad es la necesidad de identificar, de forma precisa, el modelo del robot manipulador; algunos métodos sólo suponen el conocimiento de ciertos términos del modelo dinámico, tal como la matriz de inercia. Cuando los parámetros en el modelo no concuerdan con los parámetros del dispositivo real, no se tiene un control eficiente del mismo.

Por todo lo explicado anteriormente, se deduce la necesidad de obtener un controlador que no necesite los parámetros exactos del manipulador. Un algoritmo de control inteligente, como el control difuso, es una herramienta útil para este tipo de problemas, ya que es muy versátil y permite encontrar diferentes métodos de solución para un mismo problema.

El control difuso utilizando información lingüística de un proceso posee varias ventajas tales como robustez, no necesita de un modelo matemático y el algoritmo de control se basa principalmente en la definición de las reglas que describen al sistema (Kuo, 2010) . Debido a estas razones, se desarrollará un controlador difuso, el cual no requiera de valores exactos de la dinámica del manipulador y que permita controlar al robot con un margen de error menor en comparación con otros controladores.

1.3 Objetivo general

Diseñar un controlador con base en lógica difusa para el manejo de un robot manipulador sujeto a perturbaciones externas y variaciones en sus parámetros.

1.4 Hipótesis

Con el desarrollo de un algoritmo de control por lógica difusa se obtendrá un error de posición menor en comparación con otros algoritmos, cuando se desconocen los parámetros del robot manipulador y éste está sujeto a perturbaciones externas

1.5 Alcances

Se pretende comparar los resultados obtenidos de un control por lógica difusa con base en otros controladores convencionales y además, verificar si es factible utilizar un controlador de este tipo para robots manipuladores.

Una posible aplicación que se podría dar al controlador es adecuarlo a un manipulador industrial, o bien, a un robot manipulador didáctico, el cual podría servir de referencia para futuras investigaciones realizadas en el campo de control dentro de la Facultad de Ingeniería.

1.6 Organización del trabajo

La organización de la presente tesis es la siguiente: en el capítulo 2 se presenta una descripción de los trabajos que se han realizado en el área; en éste se incluyen los temas que permiten la comprensión del proyecto realizado. Entre los tópicos de interés se encuentra: la lógica difusa y el control por lógica difusa, así como también los trabajos realizados en cuanto a robótica para el control de un manipulador.

En el capítulo 3 se describe, como primer paso, el desarrollo para obtener el modelo matemático de un robot de 2 grados de libertad; como segundo paso, se sigue con el desarrollo del controlador lógico difuso; posteriormente, se realiza un análisis del controlador obtenido.

En el capítulo 4 se muestran los resultados obtenidos del controlador difuso aplicado al robot de dos grados de libertad, se muestran simulaciones y experimentos realizados en un prototipo.

Finalmente, en el capítulo 5 se dan a conocer las conclusiones obtenidas del trabajo realizado.

II REVISIÓN DE LITERATURA

2.1 Estado del arte

A continuación se presenta el estado del arte que sustenta este trabajo, comenzando con algunos trabajos realizados en el área de lógica difusa como objeto de investigación y posteriormente se muestra la aplicación para control de sistemas no lineales, enfocándose primordialmente al control de robots manipuladores.

Existen diferentes descripciones acerca del concepto de la lógica difusa. López (2001), menciona que la lógica difusa es una teoría rigurosa, matemáticamente bien fundamentada, en donde se utiliza el concepto de verdad parcial y que además, proporciona una herramienta poderosa para aplicaciones en control y supervisión de procesos industriales. El trabajo realizado por López (2001) da una breve introducción acerca de la lógica difusa además de algunos puntos importantes a tomar en cuenta al momento de decidir utilizarla.

Una descripción más detallada la muestra Jiménez (2000). Él menciona que la lógica difusa es un concepto de control, que permite deducir una acción a realizar a partir de datos imprecisos, similar al que tiene el razonamiento humano.

Otro concepto de la lógica difusa lo manejan Juárez y Zavala (2004), describiendo a la lógica difusa como una metodología que proporciona una manera simple y elegante de obtener una conclusión a partir de una información de entrada, ambigua, imprecisa o incompleta.

El primer trabajo aplicativo de la lógica difusa fue realizado en 1974 por el Profesor Mandani de la escuela Queen Mary de Londres, quien aplicó las ideas básicas de

Zadeh sobre una caldera de vapor. La primera gran aplicación fue el tren electromagnético de Sendai construido en 1986 por la sociedad Hitachi.

Desde entonces la lógica difusa ha sido implementada en muchas aplicaciones ya que presenta algunas características como lo mencionan Rezoug *et al.* (2009) dentro de las cuales las más relevantes son: no necesita de un modelo matemático; es relativamente simple, rápida y adaptable; puede alcanzar los objetivos de diseño. Por lo tanto, el control por lógica difusa es aplicado en sistemas no lineales, donde no es posible hacer la identificación de la planta por métodos convencionales (López y Muñoz, 2007).

Existen en la actualidad diferentes aplicaciones de controladores inteligentes dentro de los cuales está la lógica difusa. Es interesante ver las comparaciones que existen entre estos diferentes algoritmos de control, por ejemplo, Alzate y Suarez (2006) presentan un análisis de la estabilidad de los sistemas no lineales a partir del plano de fase. En esta investigación, se aplican diferentes esquemas de control inteligentes sobre un sistema no lineal, utilizando como técnicas de control el control difuso, las redes neuronales y el control neurodifuso. Encontrando un mejor resultado al aplicar el control neurodifuso.

Sala y Ariño (2009) muestran algunas de las desventajas de las metodologías difusas que han sido estudiadas y superadas recientemente en algunos aspectos, sin embargo muchos sistemas no lineales pueden ser modelados como sistemas difusos de modo que el control difuso puede considerarse como una técnica para el control no lineal.

Si bien, la lógica difusa presenta aplicaciones simples como lo son en motores de CD, es aún más relevante la aplicación en robots de diferentes tipos como: industriales, didácticos, etc.

Yeong y Chung (1993) desarrollaron un controlador por lógica difusa para un robot manipulador basado en esquemas de sistemas de estructura variable. Las reglas difusas y las funciones de membresía de entrada y salida del controlador difuso, son diseñadas para garantizar la estabilidad total, en este caso, realizan una simulación en un

robot de dos grados de libertad. Además, mencionan que un robot manipulador es un caso de dinámica incierta debido a variaciones de carga, fricción y disturbios externos.

Otra aplicación interesante es la realizada por Dac Loc, *et al.* (2004) en el cual un controlador difuso adaptable es diseñado para un robot manipulador, el controlador difuso utilizado es sintetizado a partir de un conjunto de reglas IF- THEN. Lo interesante del trabajo realizado es que las funciones de pertenencia cambian de acuerdo a algunas leyes de adaptación con el propósito de controlar la trayectoria de un robot de dos grados de libertad.

Zeinali y Notash (2010) presentan una metodología por lógica difusa basado en el modelo dinámico inverso de un robot manipulador. La metodología fue implementada para construir un modelo difuso adaptativo usado en un prototipo manipulador paralelo de 4 grados de libertad.

Martínez, *et al.* (1996) desarrollaron un brazo robótico, controlando la posición por medio de un algoritmo difuso. Mencionan que el principal problema en el pasado con los brazos articulados, ha sido la dificultad en calcular la solución de las ecuaciones de la cinemática inversa del brazo. Sin embargo, sistema de posicionamiento difuso desarrollado en este artículo muestra que una simple representación del modelo cinemático puede ser utilizado. En los resultados que presentan el controlador difuso demuestra ser extremadamente rápido y flexible.

Rezoug, *et al.* (2009) proponen el control de un robot manipulador por lógica difusa, el cual es accionado por músculos neumáticos artificiales. El objetivo del control es la posición y la velocidad de rastreo del robot. Los resultados de esta investigación se muestran a nivel de simulación en la cual demuestran la viabilidad y ventajas de esta propuesta, ya que mencionan que el utilizar actuadores neumáticos presenta algunas ventajas como lo es el bajo costo, de uso robusto y sin necesidad de mantenimiento. De los resultados, se puede observar que el control de la lógica difusa puede ser aplicado al brazo robot.

Pires y Godinho (2008) presentan una aplicación de control de fuerza de un manipulador basada en una estrategia de lógica difusa. Describen el desarrollo y realización de una arquitectura de control indirecto de fuerza basada en un controlador difuso-PI, también describen el desarrollo de una aplicación en API win 32, la cual hace lecturas de los valores de la fuerza y permite efectuar ensayos de trayectorias imponiendo diversas condiciones a la fuerza deseada.

La lógica difusa tiene una historia corta, pero un rápido crecimiento debido a su capacidad de resolver problemas relacionados con la incertidumbre de la información utilizando el conocimiento de los expertos. Además, proporciona un método formal para la expresión del conocimiento en forma entendible por los humanos. Estas cualidades le aseguran un amplio campo de aplicabilidad y un alto interés para las aplicaciones industriales, presentes y futuras.

2.2 Conceptos básicos de lógica difusa

El primer concepto propuesto en lógica difusa fue en 1965, en la Universidad de California, Berkeley por Lofti A. Zadeh, quien combinó los conceptos de la lógica y de los conjuntos de Lukasiewicz mediante la definición de grados de pertenencia. Así, la Lógica Difusa ha cobrado una gran fama por la variedad de sus aplicaciones las cuales van desde el control de procesos industriales complejos hasta el diseño de dispositivos artificiales de deducción automática, pasando por la construcción de artefactos electrónicos de uso doméstico y de entretenimiento así como también de sistemas de diagnóstico.

2.2.1 Conjuntos difusos

Un concepto básico para introducirnos a la lógica difusa son los conjuntos difusos, para ello es necesario compararlo con los conjuntos clásicos. Un conjunto clásico es una colección de objetos de cualquier tipo que se caracteriza por los elementos que contiene, si X es un conjunto, entonces " $x \in X$ ", donde x es un elemento del conjunto X . Un conjunto

clásico x en un universo de discurso U se puede definir con la utilización de su función característica $\mu(x)$, que permite definir X en el dominio $[0,1]$, es decir, si un elemento del conjunto X pertenece a dicho conjunto se asigna un 1 a su función característica $\mu(x)$, y un 0 en caso contrario.

Un conjunto borroso \tilde{A} en X (universo de discurso), queda caracterizado por una función de inclusión que toma valores en el rango $[0,1]$, en otras palabras, en comparación con los conjuntos clásicos un conjunto borroso puede tomar valores intermedios (Martín del Brío y Sanz, 2007). En términos formales:

Definición 1. *La función de pertenencia $\mu_{\tilde{A}}$ de un conjunto difuso \tilde{A} es una función*

$$\mu_{\tilde{A}} : U \rightarrow [0,1]$$

Este conjunto difuso \tilde{A} en U puede ser representado como un conjunto de pares ordenados de elementos x y su grado de pertenencia o membresía $\mu_{\tilde{A}}$ (Lee, 1990).

$$\tilde{A} = \{(x, \mu_{\tilde{A}}(x)) \mid x \in X\} \tag{1.7}$$

Cada elemento $x \in U$ tiene un grado de pertenencia $\mu_{\tilde{A}} \in [0,1]$ el cual es completamente determinado por la ecuación (2.1). El grado de pertenencia se cuantifica mediante la función de pertenencia $\mu_{\tilde{A}}$ como sigue: si $\mu_{\tilde{A}}$ indica el grado de pertenencia de x al conjunto \tilde{A} , entonces toma valores entre 0 y 1; si $\mu_{\tilde{A}}=1$, x pertenece totalmente a \tilde{A} ; si $\mu_{\tilde{A}}=0$, x no pertenece a \tilde{A} .

2.2.2 Funciones de pertenencia de conjuntos difusos

Existen diferentes formas que puede tomar una función de pertenencia. Las más frecuentes son la función tipo trapezoidal, singleton, triangular, gaussiana. Como se muestra en la

figura 2.1.

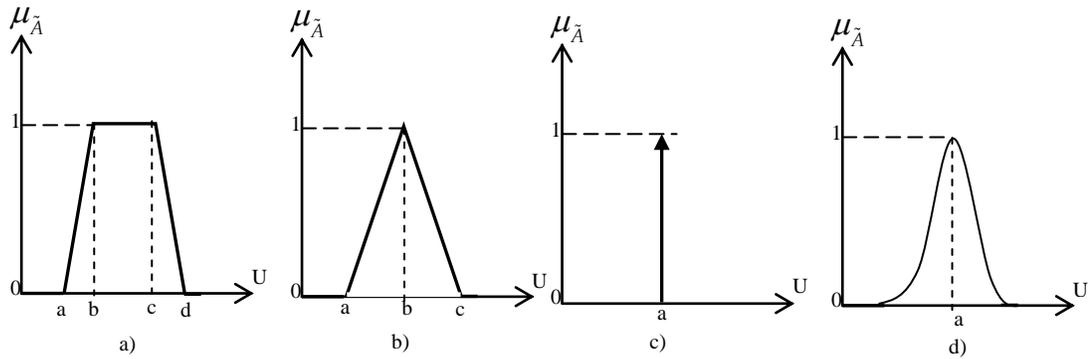


Figura 2.1 Tipos de funciones de membresía: a) trapezoidal b) triangular c) singleton y d) gaussiana

Para evitar hacer tediosa esta parte introductoria, únicamente se describirán las funciones de membresía que se utilizarán a lo largo del trabajo de investigación, las cuales son la triangular y singleton.

Una función de membresía de tipo triangular se puede expresar de forma matemática como,

$$\mu_{\tilde{A}}(x) = \begin{cases} 0 & x < a \\ \frac{x-a}{b-a} & a \leq x \leq b \\ \frac{c-x}{c-b} & b \leq x \leq c \\ 0 & x > c \end{cases} \quad (1.8)$$

Esta función es adecuada para modelar propiedades con un valor de inclusión distinto de cero para un rango de valores estrecho en torno a un punto b.

La función de membresía tipo singleton tiene valor 1 sólo para un punto a y 0 para el resto. Se utiliza habitualmente en sistemas difusos simples para definir los conjuntos difusos de las particiones de las variables de salida, pues permite simplificar cálculos y requiere menos memoria para almacenar la base de reglas. La figura 2.1 c) muestra a la función singleton, la cual se define como:

$$\mu_{\bar{A}}(x) = \begin{cases} 1 & x = a \\ 0 & x \neq a \end{cases} \quad (1.9)$$

2.2.3 Variables lingüísticas y valores lingüísticos

Una variable lingüística es aquella que se puede representar en términos del lenguaje natural, por ejemplo: mucho, poco, positivo, negativo, etc., estas palabras describen el comportamiento de un determinado sistema.

Definición 2. Si una variable puede describirse con palabras del lenguaje natural, ésta es llamada variable lingüística, donde las palabras son caracterizadas por conjuntos difusos definidos dentro del universo de discurso en el cual la variable es descrita (Wang, 1997).

Por ejemplo, la velocidad de un carro, el término o la palabra velocidad sería la variable lingüística ya que puede ser descrita como baja, media o alta. Tomando el ejemplo anterior, se obtiene entonces la definición de un valor lingüístico, es decir, es la descripción de una variable lingüística y toma valores del lenguaje natural, su representación en los sistemas difusos es mediante las funciones de pertenencia. En la figura 2.2 se muestra el ejemplo descrito anteriormente.

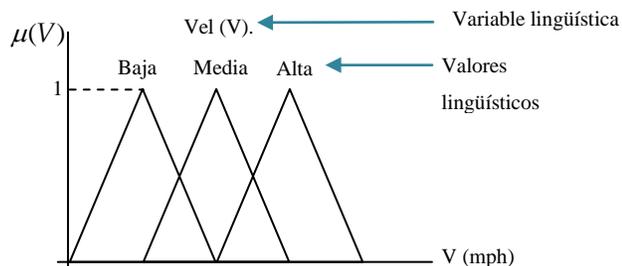


Figura 2.2 Representación de la variable lingüística velocidad y sus valores lingüísticos

2.2.4 Operaciones entre conjuntos difusos

Las operaciones básicas que se realizan entre conjuntos difusos son: unión, intersección y complemento.

Definición 3. Unión. La unión de dos conjuntos difusos A y B es un conjunto difuso C , escrito como $c = A \cup B$ ó $C = A \text{ or } B$, cuya función de membresía se relaciona a las de A y B por (Lee, 1990):

$$\begin{aligned}\mu_c(x) &= \max(\mu_A(x), \mu_B(x)) \\ &= \mu_A(x) \vee \mu_B(x)\end{aligned}\tag{1.10}$$

Definición 4. Intersección. La intersección de dos conjuntos difusos A y B es un conjunto difuso C , escrito como $C = A \cap B$ ó $C = A \text{ and } B$, cuya función de membresía se relaciona a las de A y B por (Lee, 1990):

$$\begin{aligned}\mu_c(x) &= \min(\mu_A(x), \mu_B(x)) \\ &= \mu_A(x) \wedge \mu_B(x)\end{aligned}\tag{1.11}$$

Definición 5. Complemento. El complemento del conjunto difuso A , denotado por $\bar{A}(\neg A, \text{not } A)$ se define como (Lee, 1990):

$$\mu_{\bar{A}}(x) = 1 - \mu_A(x)\tag{1.12}$$

2.2.5 Reglas difusas SI- ENTONCES

Las reglas difusas combinan uno o más conjuntos borrosos de entrada, llamados antecedentes o premisas y les asocian un conjunto borroso de salida, llamado consecuente o consecuencia. Los conjuntos difusos se asocian utilizando un conector lógico “o/or” ó “y/and”. Éstas permiten expresar el conocimiento que se tiene sobre la relación antecedente consecuente. Para expresar este conocimiento de forma completa normalmente se precisa de varias reglas que se agrupan formando lo que se conoce como base de reglas difusas. Normalmente tienen la siguiente forma:

SI < antecedente o premisa > ENTONCES < consecuente o conclusión >

Este formato de reglas se conoce como difuso puro o tipo Mandani por ser quien primero las propuso. Otro formato de reglas es el llamado de tipo Sugeno. En este caso la función de salida es una combinación lineal de las variables de entrada (Martín del Brío y Sanz, 2007).

2.2.6 Controladores difusos

Presentadas las bases de los conjuntos difusos, se muestra a continuación la aplicación de todos los conceptos planteados en los párrafos anteriores. Un controlador difuso o sistema difuso tiene como finalidad modelar un sistema específico. La estructura básica de éste, consiste principalmente de cuatro componentes:

Fuzificación: Se encarga de convertir los valores físicos de una variable lingüística de entrada en valores lingüísticos, los cuales son representados en conjuntos difusos. Se asignan los grados de membresía de todos los valores de entrada.

Base de reglas: contiene las reglas difusas con estructura SI-ENTONCES que definen la relación entre las variables lingüísticas.

Mecanismo de inferencia. Obtiene una conclusión utilizando la base de reglas y las determinadas entradas.

Defuzificación: se encarga de convertir el resultado o acción de control, de términos lingüísticos a términos que puedan ser empleados por un actuador o dispositivo electrónico. La figura 2.3 muestra el esquema completo del controlador difuso. A continuación se presenta con más detalle cada etapa del controlador difuso.



Figura 2.3 Esquema de control difuso

Fuzificación

Los conjuntos difusos son usados para cuantificar la información en la base de reglas y el mecanismo de inferencia opera sobre estos conjuntos para obtener otros conjuntos difusos; por lo tanto, se debe especificar como convertirá el sistema difuso sus entradas numéricas $x \in X$ en conjuntos difusos, para entonces ser utilizados por el sistema difuso (Passino y Yurkovich, 1998). La función de fuzificación más simple es la tipo singleton, es decir, el valor crisp será el único valor del conjunto difuso. Formalmente se puede representar como:

$$\mu_{\bar{A}}(x') = \begin{cases} 1 & x = x' \\ 0 & x \neq x' \end{cases} \quad (1.13)$$

Donde x' es el valor actual de la entrada.

Las razones del porque no han sido utilizados otros métodos de fuzificación (por ejemplo la gaussiana) son: agregan mayor complejidad computacional en el proceso de inferencia y además la necesidad de utilizarlas no ha sido justificada. Esto se debe en parte al hecho de que se alcanzan buenas capacidades funcionales en el sistema difuso, cuando sólo se utiliza fuzificación singleton.

Mecanismo de inferencia

La inferencia calcula el valor de verdad para la premisa de cada regla y de esto se infiere la conclusión de la regla, el resultado es un subconjunto difuso que se asigna a cada variable de salida para cada regla. Hay dos procedimientos en inferencia difusa: “*generalized modus ponens*” (GMP) y “*generalized modus tollens*” (GMT).

El razonamiento en lógica difusa se hace en base al GMP ya que permite calcular el consecuente B de una regla difusa:

Si x es A entonces y es B

x es A' **premisa**

Se deduce y es B'

Defuzificación

La defuzificación convierte los valores difusos de las variables de salida en valores concretos dentro del universo de discurso correspondiente, es decir, genera una acción no difusa a partir de la acción difusa resultante del sistema de inferencia. Supongamos que tenemos M reglas difusas del tipo: SI x es L_x ENTONCES y es L_y , donde si se introducen valores de entrada x_i , se obtienen como salida un conjunto asociado a todas las reglas \tilde{y}_j .

La asociación del conjunto difuso \tilde{y}_j y el valor crisp y_j es la defuzificación donde $\mu_{\tilde{y}_j}(y_n)$ es el grado de pertenencia del valor de entrada actual para la n -ésima regla. Existen diversos métodos de defuzificación, a continuación se presentan los dos más utilizados:

Método del centro de área (CDA). En este método se calcula el centro de gravedad del área limitada por la curva de la función de membresía, para ser el valor de salida representativo de la cantidad difusa.

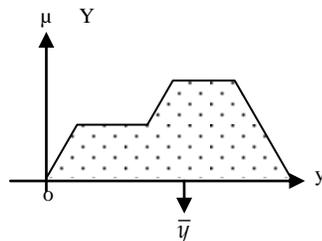


Figura 2.4 Método del centro de área

$$\bar{y} = \frac{\int \mu_y(y) \cdot y \, dy}{\int \mu_y(y) \, dy} \quad (1.14)$$

Entre las propiedades positivas de este método hay que destacar que es continuo (un pequeño cambio en las entradas no implica un cambio brusco en las salidas) y no es

ambiguo (obtiene un valor único como resultado del proceso). Sin embargo el método presenta algunos inconvenientes. En primer lugar, no es plausible (ya que la salida puede corresponder a un punto del soporte del conjunto con bajo grado de activación). Otra característica, a veces indeseable, es que no se trata de un método lineal. Por otra parte, la principal limitación de este método es la necesidad de recorrer todo el universo de discurso para evaluar el centroide. Esto hace que el cálculo del valor representativo sea muy costoso en número de operaciones.

Método del promedio de centros. Es un método frecuentemente utilizado debido a su reducida complejidad computacional comparada con el método anterior ya que es independiente del soporte del conjunto difuso inferido por cada regla y de su forma. Este método toma en consideración sólo el punto en que las funciones de pertenencia individuales de la variable de salida tienen valor igual a la unidad (en el caso de estar normalizadas y ser éste único). Para el caso común de utilizar representaciones triangulares, trapezoidales o gaussianas, el valor considerado es el centro de la función. Si llamamos α_i al grado con el que el valor de entrada activa el antecedente de la regla i y c_i es el valor de pico del conjunto difuso de salida que se encuentra en el consecuente de la regla i , la salida defuzificada mediante este método sería:

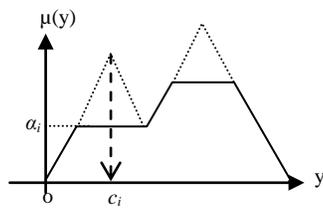


Figura 4.5 Método promedio de centros

$$\bar{y} = \frac{\sum_{i=1}^l \alpha_i \cdot c_i}{\sum_{i=1}^l \alpha_i} \quad (1.15)$$

Los procesos de inferencia, fuzificación y defuzificación son esenciales para definir los módulos de control difuso.

Centro de Sumas: Es un método similar al CDA, pero en este caso se trata de evitar el complejo cálculo del conjunto de salida y . Para la obtención del elemento numérico se considera el área de cada uno de los valores lingüísticos de salida individualmente. En el

caso frecuente de que exista solapamiento entre funciones de pertenencia adyacentes en el dominio de salida, dichas regiones serán reflejadas más de una vez.

$$\bar{y} = \frac{\sum_{i=1}^l \alpha_i \cdot S_i \cdot c_i}{\sum_{i=1}^l \alpha_i \cdot S_i} \quad (1.16)$$

2.3 Control de robots manipuladores

En la industria la mayoría de los robots son brazos articulados. Según la definición del “Robot Institute of America”, un robot industrial es un manipulador programable multifuncional diseñado para mover materiales, piezas, herramientas o dispositivos especiales (Ollero, 2007)

Otra definición de robot manipulador -en términos mecánicos- es un brazo mecánico articulado formado por eslabones conectados a través de uniones o articulaciones que permiten un movimiento relativo entre dos eslabones consecutivos. El movimiento de cada articulación puede ser traslacional, rotacional o una combinación de ambos. Bajo condiciones razonables, el número de articulaciones en un manipulador determina su número de grados de libertad (Kelly y Santibañez, 2003).

2.3.1 Cinemática Directa

La cinemática es la ciencia que estudia el movimiento, sin considerar las fuerzas que lo ocasionan. Por lo tanto, se estudian, la posición, la velocidad y la aceleración. En un manipulador se estudian todas las propiedades geométricas, así como la manera en que la posición cambia a medida que se articula el mecanismo.

Un manipulador consiste de una serie de cuerpos rígidos conectados por vínculos, éstos pueden ser de dos tipos: rotacionales y prismáticos. La estructura completa forma la cadena cinemática. Al final de la cadena cinemática se encuentra el efector final que puede

ser una pinza, herramienta, etc. El cual permite la manipulación de objetos en el espacio. (Sciavicco y Siciliano, 2000)

El problema de la cinemática directa consiste en determinar cuál es la posición y orientación del extremo final del robot con respecto a un sistema de coordenadas que se toma como referencia, conociendo los ángulos de sus articulaciones y sus parámetros geométricos.

2.3.2 Cinemática inversa

El objetivo del problema cinemático inversa consiste en encontrar los valores que deben adoptar las coordenadas articulares del robot para que su extremo se posicione y oriente según una determinada localización espacial (Barrientos *et al.*, 2007). Se puede decir entonces que la cinemática inversa hace lo inverso a la cinemática directa, es decir, busca determinar los ángulos que debe formar cada articulación para llevar el brazo a una posición (o coordenadas) conocida o deseada.

Existen diferentes métodos para resolver el problema de la cinemática inversa como lo hace la solución geométrica, este método permite obtener los valores de las primeras variables articulares, éstas consiguen posicionar el robot. Para ello utilizan relaciones trigonométricas y geométricas sobre los elementos del robot. Se recurre a la resolución de triángulos formados por los elementos y articulaciones del robot.

2.3.3 Dinámica de manipuladores

La dinámica es la ciencia que se ocupa de estudiar la relación entre las fuerzas que actúan sobre un cuerpo y el movimiento que en él se origina. En robótica, el modelo dinámico tiene por objetivo conocer la relación entre el movimiento del robot y las fuerzas implicadas en el mismo. Este modelo relaciona matemáticamente: la localización del robot definida por sus variables articulares o por las coordenadas de localización de su extremo y

sus derivadas: velocidad y aceleración, las fuerzas y pares aplicados en las articulaciones, los parámetros dimensionales del robot (Barrientos *et al.*, 2007).

Sin embargo, la obtención de este modelo para mecanismos de uno o dos grados de libertad no es excesivamente compleja, pero si el número de grados de libertad aumenta, el planteamiento y obtención del modelo se complica enormemente.

Por lo tanto, la obtención de este modelo es uno de los aspectos más complejos de la robótica, además es de suma utilidad porque con el modelo se obtiene: la simulación del movimiento del robot; el diseño y evaluación de la estructura mecánica del robot y el diseño y evaluación del control dinámico del robot. Este último aspecto es de gran importancia ya que de la calidad del control depende la precisión y velocidad de sus movimientos.

Existen diversos métodos para obtener el modelo dinámico del robot como lo son el método de Newton- Euler, el cual se basa en las formulas dinámicas elementales y el análisis de fuerzas y momentos que actúan sobre cada vínculo. También se encuentra el método de Euler-Lagrange, éste último está basado en la diferencia de la energía cinética y la energía potencial del mecanismo.

La energía cinética es la energía que se genera en un cuerpo en movimiento y depende directamente de la velocidad, por lo tanto, la energía cinética k del i -ésimo vínculo puede expresarse como:

$$k_i = \frac{1}{2} m_i v_{C_i}^T v_{C_i} + \frac{1}{2} \omega_i^T I_i \omega_i \quad (1.17)$$

En donde el primer término es la energía cinética debida a la velocidad lineal del centro de masas del vínculo, el segundo término es la energía cinética debida a la velocidad angular del vínculo. Las v_{C_i}, ω_i en la ec. (1.17), son funciones de la posición y de la

velocidad. La energía cinética total es la suma de la energía cinética de cada vínculo del robot manipulador como lo muestra la ecuación (1.18).

$$k = \sum_{i=1}^n k_i \quad (1.18)$$

La energía potencial es la energía asociada con la fuerza gravitatoria. Esta dependerá de la altura relativa de un objeto a algún punto de referencia, la masa, y la fuerza de la gravedad. La energía potencial del i -ésimo vínculo puede expresarse como:

$$u_i = m_i g P_{ci} \quad (1.19)$$

en donde g es el vector de gravedad, P_{ci} es el vector que ubica al centro de masas del i -ésimo vínculo (Craig, 2006). La energía potencial total almacenada en el robot manipulador es la suma de la energía potencial de cada vínculo.

$$u = \sum_{i=1}^n u_i \quad (1.20)$$

La formulación dinámica lagrangiana provee los medios para derivar las ecuaciones de movimiento a partir de una función escalar llamada lagrangiano que se define como la diferencia entre la energía cinética y potencial de un sistema mecánico. El lagrangiano del manipulador es:

$$L(q, \dot{q}) = k(q, \dot{q}) - u(q) \quad (1.21)$$

Y las ecuaciones de movimiento serían

$$\frac{d}{dt} \frac{\partial L}{\partial \dot{\theta}} - \frac{d}{dt} \frac{\partial L}{\partial \theta} = \tau, \quad (1.22)$$

donde τ es el vector $n \times 1$ de momentos de torsión de los actuadores. En general, la ecuación que rige la dinámica del manipulador es:

$$\tau = M(q)\ddot{q} + C(q, \dot{q}) + g(q) \quad (1.23)$$

donde M es la matriz de masas de $n \times n$, C es un vector de términos centrífugos y Coriolis y g es un vector de términos de gravedad.

III METODOLOGÍA

3.1 Obtención del modelo dinámico del manipulador

Las ecuaciones dinámicas de un robot manipulador pueden obtenerse a partir de las ecuaciones de movimiento de Newton, pero presenta un inconveniente ya que se complica notablemente cuando aumenta el número de articulaciones del robot (Kelly y Santibáñez, 2003). Es por esta razón que se recurre a emplear las ecuaciones de movimiento de Lagrange.

El método de Euler-Lagrange se basa en una interpretación de la energía involucrada, esto es, requiere del cálculo las energías cinética k y Potencial u , de igual forma requiere del cálculo del Lagrangiano que es la diferencia entre la energía cinética y potencial.

En un robot manipulador la energía cinética, será la suma de la energía de cada eslabón, del mismo modo se calcula la energía potencial. A continuación se muestra el procedimiento para obtener el modelo dinámico del robot manipulador de dos grados de libertad.

3.1.1 Cálculo de la energía cinética

La energía cinética de un cuerpo rígido se determina como la suma de: el producto de la mitad de su masa por el cuadrado de la rapidez del centro de masa (energía cinética traslacional) y el producto de la mitad de su momento de inercia (referido al centro de masa) por el cuadrado de su velocidad angular (energía cinética rotacional).

Esta descripción, se puede representar matemáticamente como se muestra en la ecuación (2.1).

$$k_i(q, \dot{q}) = \frac{1}{2} m_i v^2 + \frac{1}{2} I_i \omega^2 \quad (2.1)$$

Donde:

m_i Masa del i-ésimo eslabón

ω Velocidad angular

I_i Momento de inercia del i-ésimo eslabón

El problema de calcular la energía cinética se reduce a encontrar el cuadrado de la rapidez del centro de masa representada por v^2 por lo que es necesario utilizar el esquema geométrico del manipulador presentado en la figura 3.1, además haciendo uso del procedimiento desarrollado por Kelly y Santibañez (2003), se presenta el desarrollo para obtener la energía cinética.

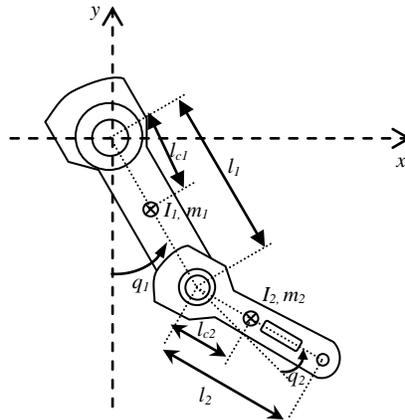


Figura 3.1 Análisis geométrico del manipulador de 2 grados de libertad

Por lo tanto para el eslabón 1 se obtienen las coordenadas para la masa m_1 expresadas en el plano x-y

$$\begin{aligned} x_1 &= l_{c1} \text{sen}(q_1) \\ y_1 &= -l_{c1} \text{cos}(q_1) \end{aligned} \quad (2.2)$$

el vector velocidad v_1 es por lo tanto:

$$v_1 = \begin{bmatrix} \dot{x}_1 \\ \dot{y}_1 \end{bmatrix} = \begin{bmatrix} l_{c1}\dot{q}_1 \cos(q_1) \\ l_{c1}\dot{q}_1 \sin(q_1) \end{bmatrix} \quad (2.3)$$

Para el eslabón 2 se obtienen las coordenadas para la masa m_2 :

$$\begin{aligned} x_2 &= l_1 \sin(q_1) + l_{c2} \sin(q_1 + q_2) \\ y_2 &= -l_1 \cos(q_1) - l_{c2} \cos(q_1 + q_2) \end{aligned} \quad (2.4)$$

el vector velocidad v_2 sería:

$$v_2 = \begin{bmatrix} \dot{x}_2 \\ \dot{y}_2 \end{bmatrix} = \begin{bmatrix} l_1 \dot{q}_1 \cos(q_1) + l_{c2} \cos(q_1 + q_2)(\dot{q}_1 + \dot{q}_2) \\ l_1 \dot{q}_1 \sin(q_1) + l_{c2} \sin(q_1 + q_2)(\dot{q}_1 + \dot{q}_2) \end{bmatrix} \quad (2.5)$$

La rapidez al cuadrado para ambas articulaciones resulta ser:

$$\begin{aligned} v_1^T v_1 &= l_{c1}^2 \dot{q}_1^2 \\ v_2^T v_2 &= l_1^2 \dot{q}_1^2 + l_{c2}^2 (\dot{q}_1^2 + 2\dot{q}_1 \dot{q}_2 + \dot{q}_2^2) + 2l_1 l_{c2} [\cos(q_2)] (\dot{q}_1^2 + \dot{q}_1 \dot{q}_2) \end{aligned} \quad (2.6)$$

Obtenidas las velocidades, se calcula la energía cinética total del manipulador, la cual está dada por la suma de las energías cinéticas de cada vínculo esto es:

$$\begin{aligned} K_1 &= \frac{1}{2} m_1 v_1^T v_1 + \frac{1}{2} I_1 \dot{q}_1^2 \\ K_2 &= \frac{1}{2} m_2 v_2^T v_2 + \frac{1}{2} I_2 \dot{q}_2^2 \end{aligned} \quad (2.7)$$

Por lo tanto sustituyendo (3.6) en la ecuación (2.7)

$$\begin{aligned} K_1 &= \frac{1}{2} m_1 l_{c1}^2 \dot{q}_1^2 + \frac{1}{2} I_1 \dot{q}_1^2 \\ K_2 &= \frac{1}{2} m_2 l_1^2 \dot{q}_1^2 + \frac{1}{2} m_2 l_{c2}^2 [\dot{q}_1^2 + 2\dot{q}_1 \dot{q}_2 + \dot{q}_2^2] + m_2 l_1 l_{c2} [\cos(q_2)] (\dot{q}_1^2 + \dot{q}_1 \dot{q}_2) + \frac{1}{2} I_2 (\dot{q}_1 + \dot{q}_2)^2 \end{aligned} \quad (2.8)$$

3.1.2 Cálculo de la energía potencial

La energía potencial se define como la energía que poseen los cuerpos que se encuentran a una altura, y depende de la masa del cuerpo y de la gravedad. La ecuación (2.9) define a la energía potencial.

$$U = mgh \quad (2.9)$$

Donde:

m masa del cuerpo

g gravedad

h altura del cuerpo

De forma similar a la energía cinética, la energía potencial total del sistema se puede obtener sumando la energía potencial de cada eslabón, obteniéndose:

$$\begin{aligned} U_1 &= -m_1 g l_{c1} \cos(q_1) \\ U_2 &= -m_2 g l_1 \cos(q_1) + m_2 g l_{c2} \cos(q_1 + q_2) \end{aligned} \quad (2.10)$$

3.1.3 Cálculo del Lagrangiano

La función Lagrangiana se define como

$$\begin{aligned} L &= K - U \\ L &= K_1 + K_2 - U_1 - U_2 \end{aligned} \quad (2.11)$$

Por lo tanto sustituyendo las ecuaciones (2.8) y (2.10) se obtiene:

$$\begin{aligned} L &= \frac{1}{2} [m_1 l_{c1}^2 m_2 l_1^2 + I_1 + I_2] \dot{q}_1^2 + \frac{1}{2} m_2 l_{c2} [\dot{q}_1^2 + 2\dot{q}_1 \dot{q}_2 + \dot{q}_2^2] + m_2 l_1 l_{c2} [\cos(q_2)] (\dot{q}_1^2 + \dot{q}_1 \dot{q}_2) + \\ & [m_1 l_{c1} + m_2 l_1] g \cos(q_1) + m_2 g l_{c2} \cos(q_1 + q_2) + \frac{1}{2} I_2 \dot{q}_2^2 \end{aligned} \quad (2.12)$$

Dado que la formulación dinámica se basa en la ecuación (2.13)

$$\tau_i = \frac{d}{dt} \frac{\partial L}{\partial \dot{q}_i} - \frac{\partial L}{\partial q_i} \quad (2.13)$$

Siendo τ_i el par aplicado e i el número de articulación, se requiere del cálculo de las derivadas con respecto a cada articulación por lo tanto:

$$\begin{aligned} \frac{\partial L}{\partial \dot{q}_1} = & [m_1 l_{c1}^2 + m_2 l_1^2 + I_1 + I_2] \dot{q}_1 + m_2 l_{c2}^2 \dot{q}_1 + m_2 l_{c2}^2 \dot{q}_2 + 2m_2 l_1 l_{c2} \cos(q_2) \dot{q}_1 \\ & + m_2 l_1 l_{c2} \cos(q_2) \dot{q}_2 + I_2 \dot{q}_2 \end{aligned} \quad (2.14)$$

$$\frac{\partial L}{\partial \dot{q}_2} = m_2 l_{c2}^2 \dot{q}_1^2 + m_2 l_{c2} \dot{q}_2 + m_2 l_1 l_{c2} \cos(q_2) \dot{q}_1 + I_2 [\dot{q}_1 + \dot{q}_2]$$

$$\frac{\partial L}{\partial q_1} = -[m_1 l_{c1} + m_2 l_1] g \text{sen}(q_1) - m_2 g l_{c2} \text{sen}(q_1 + q_2)$$

$$\frac{\partial L}{\partial q_2} = -m_2 l_1 l_{c2} \text{sen}(q_2) [\dot{q}_1 \dot{q}_2 + \dot{q}_1^2] - m_2 g l_{c2} \text{sen}(q_1 + q_2)$$

Utilizando los resultados de la ecuación (2.14)

$$\begin{aligned} \frac{d}{dt} \frac{\partial L}{\partial \dot{q}_1} = & [m_1 l_{c1}^2 + m_2 l_1^2 + m_2 l_{c2}^2 + 2m_2 l_1 l_{c2} \cos(q_2) + I_1 + I_2] \ddot{q}_1 + [m_2 l_{c2}^2 + m_2 l_1 l_{c2} \cos(q_2) + I_2] \ddot{q}_2 \\ & - 2m_2 l_1 l_{c2} \text{sen}(q_2) \dot{q}_1 \dot{q}_2 - m_2 l_1 l_{c2} \text{sen}(q_2) \dot{q}_2^2 \end{aligned} \quad (2.15)$$

$$\frac{d}{dt} \frac{\partial L}{\partial \dot{q}_2} = m_2 l_{c2}^2 \ddot{q}_1 + m_2 l_{c2}^2 \ddot{q}_2 + m_2 l_1 l_{c2} \cos(q_2) \ddot{q}_1 - m_2 l_1 l_{c2} \text{sen}(q_2) \dot{q}_1 \dot{q}_2 + I_2 [\ddot{q}_1 + \ddot{q}_2] \quad (2.16)$$

Por lo tanto se tiene que para la articulación 1 y 2 se obtiene:

$$\begin{aligned} \tau_1 = & [m_1 l_{c1}^2 + m_2 l_1^2 + m_2 l_{c2}^2 + 2m_2 l_1 l_{c2} \cos(q_2) + I_1 + I_2] \ddot{q}_1 + [m_2 l_{c2}^2 \\ & + m_2 l_1 l_{c2} \cos(q_2) + I_2] \ddot{q}_2 - 2m_2 l_1 l_{c2} \text{sen}(q_2) \dot{q}_1 \dot{q}_2 - m_2 l_1 l_{c2} \text{sen}(q_2) \dot{q}_2^2 \\ & + [m_1 l_{c1} + m_2 l_1] g \text{sen}(q_1) + m_2 g l_{c2} \text{sen}(q_1 + q_2) \end{aligned} \quad (2.17)$$

$$\tau_2 = [m_2 l_{c2}^2 + m_2 l_1 l_{c2} \cos(q_2) + I_2] \ddot{q}_1 + [m_2 l_{c2}^2 + I_2] \ddot{q}_2 + m_2 l_1 l_{c2} \text{sen}(q_2) \dot{q}_1^2 + m_2 g l_{c2} \text{sen}(q_1 + q_2) \quad (2.18)$$

Agrupando términos y utilizando la notación $s_i = \text{sen}(q_i)$; $c_i = \text{cos}(q_i)$

$$\tau_1 = [m_1 l_{c1}^2 + m_2 l_1^2 + m_2 l_{c2}^2 + 2m_2 l_1 l_{c2} c_2 + I_1 + I_2] \ddot{q}_1 + [m_2 l_{c2}^2 + m_2 l_1 l_{c2} c_2 + I_2] \ddot{q}_2 - 2m_2 l_1 l_{c2} s_2 \dot{q}_1 \dot{q}_2 - m_2 l_1 l_{c2} s_2 \dot{q}_2^2 + [m_1 l_{c1} + m_2 l_1] g s_1 + m_2 g l_{c2} s_{12} \quad (2.19)$$

$$\tau_2 = [m_2 l_{c2}^2 + m_2 l_1 l_{c2} c_2 + I_2] \ddot{q}_1 + [m_2 l_{c2}^2 + I_2] \ddot{q}_2 + m_2 l_1 l_{c2} s_2 \dot{q}_1^2 + m_2 g l_{c2} s_{12} \quad (2.20)$$

También se pueden expresar en forma matricial

$$\tau = M(q) \ddot{q} + C(q, \dot{q}) \dot{q} + g(q) \quad (2.21)$$

donde M representa la matriz de masas, G es el vector de gravedad y C el vector de términos centrífugos y de Coriolis.

$$\begin{bmatrix} \tau_1 \\ \tau_2 \end{bmatrix} = \begin{bmatrix} m_1 l_{c1}^2 + 2m_2 l_1 l_{c2} c_2 + (l_1^2 + l_{c2}^2) m_2 + I_1 + I_2 & m_2 l_{c2}^2 + m_2 l_1 l_{c2} c_2 + I_2 \\ m_2 l_{c2}^2 + m_2 l_1 l_{c2} c_2 + I_2 & m_2 l_{c2}^2 + I_2 \end{bmatrix} \begin{bmatrix} \ddot{q}_1 \\ \ddot{q}_2 \end{bmatrix} + \begin{bmatrix} -m_2 l_1 l_{c2} s_2 \dot{q}_2 & -m_2 l_1 l_{c2} s_2 (\dot{q}_1 + \dot{q}_2) \\ m_2 l_1 l_{c2} s_2 \dot{q}_1 & 0 \end{bmatrix} \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \end{bmatrix} + \begin{bmatrix} (m_1 l_{c1} + m_2 l_1) g s_1 + m_2 g l_{c2} s_{12} \\ m_2 g l_{c2} s_{12} \end{bmatrix} \quad (2.22)$$

3.2 Simulación del modelo encontrado

El modelo encontrado relaciona los pares de control en las articulaciones con la evolución de las variables articulares. Si se pretende simular el comportamiento dinámico del manipulador ante determinados pares, es necesario resolver el modelo para obtener las trayectorias articulares (Ollero, 2007). A partir de la ecuación (2.21) se obtiene:

$$\ddot{q} = M^{-1}(q)[\tau - C(q, \dot{q})\dot{q} - g(q)] \quad (2.23)$$

Las simulaciones se realizaron en Matlab® para tres diferentes valores de par, con los siguientes parámetros del robot (ver apéndice D).

Tabla 3-1 Parámetros del robot de 2 gdl

Parámetros	Valor
l_1 (longitud eslabón 1)	0.1933 m
l_2 (longitud eslabón 2)	0.1458 m
m_1 (masa eslabón 1)	0.763 kg
m_2 (longitud eslabón 2)	0.343 kg
l_{c1} (longitud del centro de masa eslabón 1)	0.136 m
l_{c2} (longitud del centro de masa eslabón 2)	0.04849 m
I_1 (Inercia del eslabón 1)	0.0186 kgm ²
I_2 (Inercia del eslabón 2)	0.00269kgm ²
Constante g (gravedad)	9.81 m/s ²

Así también se utilizó la siguiente trayectoria en coordenadas cartesianas:

$$\begin{aligned} x &= l_1 \sin(q_1) + l_2 \sin(q_1 + q_2) \\ y &= -l_1 \cos(q_1) - l_2 \cos(q_1 + q_2) \end{aligned} \quad (2.24)$$

se tomó como referencia el trabajo expuesto por Maldonado del toro, *et al.* (2011), en el cual se presentan las pruebas que se muestran a continuación.

Prueba 1. La primer simulación se realizo con $\tau_1=0$ N-m y $\tau_2=0.15$ N-m. La Figura 3.2 muestra el resultado obtenido con el modelo desarrollado en esta sección y en la Figura 3.3 se muestra la trayectoria lograda por Maldonado del Toro, *et al.* (2011).

Dado que únicamente se mueve el eslabón 2, la trayectoria que se genera son varias eliEPPs sin un centro fijo, esto por la inercia que genera el eslabón 2 al moverse en el plano X-Y.

Prueba 2. Para esta segunda simulación se consideran los pares de $\tau_1=1.3$ N-m y $\tau_2=0$ N-m. Las figuras Figura 3.4 Figura 3.5 muestran el resultado obtenido y el presentado por Maldonado del toro, *et al.* (2011) respectivamente.

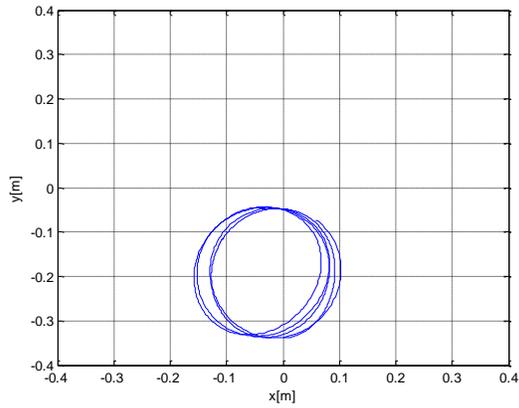


Figura 3.2 Trayectoria obtenida con $\tau_1=0$ N-m y $\tau_2=0.15$ N-m

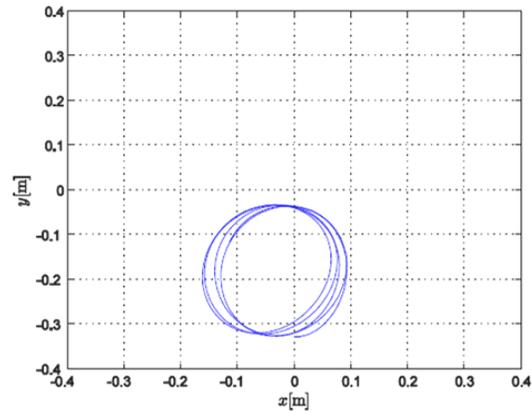


Figura 3.3 Trayectoria obtenida por Maldonado del toro, *et al.* (2011)

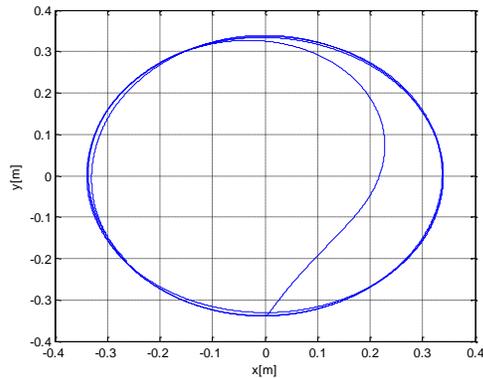


Figura 3.4 Trayectoria correspondiente a $\tau_1=1.3$ N-m y $\tau_2=0$ N-m

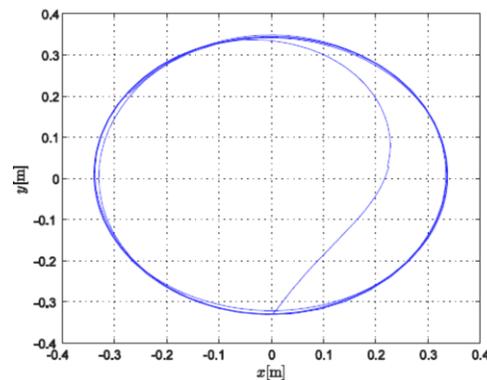


Figura 3.5 Trayectoria obtenida por Maldonado del toro, *et al.* (2011) $\tau_1=1.3$ N-m y $\tau_2=0$ N-m

El eslabón 1 genera una trayectoria en el plano XY de forma elíptica; en comparación con el anterior, la elipse es de mayor tamaño que la obtenida aplicando únicamente un valor de par a la articulación 2.

Prueba 3. En esta última prueba se aplicaron valores de par de $\tau_1=1.6$ N-m y $\tau_2=-0.3$ N-m, a cada eslabón.

La Figura 3.6 muestra la trayectoria obtenida en el plano XY, dado que los valores de par son de signo contrario para cada articulación, se genera una elipse, - realizada por la articulación 1- formada por dobleces, esto debido a la articulación 2.

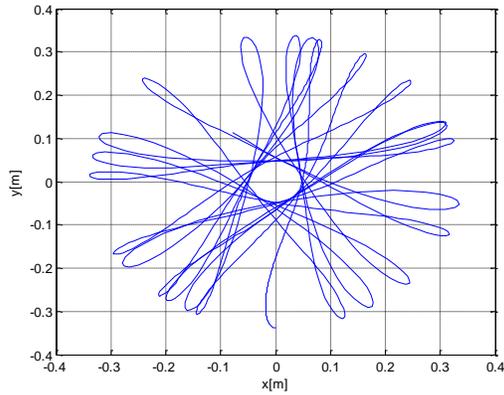


Figura 3.6 Trayectoria en el plano XY con $\tau_1=1.6$ N-m y $\tau_2=-0.3$ N-m

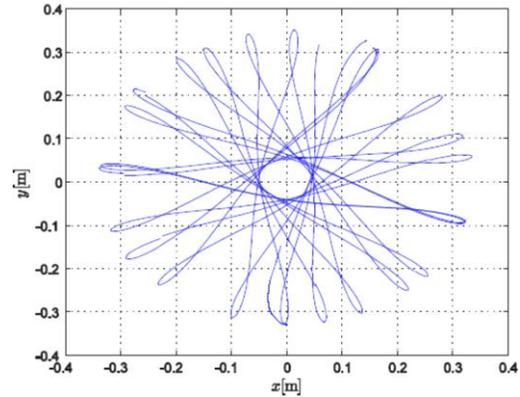


Figura 3.7 Trayectoria presentada por Maldonado del toro, *et al.* (2011) $\tau_1=1.6$ N-m y $\tau_2=-0.3$ N-m

Todas las pruebas se simularon en un tiempo 1.2 s., con un periodo de muestreo de 1ms. Es importante mencionar que los resultados obtenidos de la comparación que se realiza entre el modelo desarrollado en este trabajo con el presentado por Maldonado del toro, *et al.* (2011) son satisfactorios ya que se obtiene el comportamiento esperado, reflejado en las coordenadas X-Y.

3.3 Diseño del controlador

3.3.1 Introducción

Los sistemas expertos de control difuso basados en reglas, conocido como controladores difusos o CLD son, sin duda, la aplicación más extendida de la lógica difusa (Martin y Sanz, 2007).

Hoy en día, el control difuso ha sido sugerido como una alternativa a las técnicas de control convencionales para sistemas de control complejos. Durante la última década, el

control difuso ha atraído una gran atención de las comunidades académica e industrial, especialmente en el control de manipuladores.

Los CLD's han ampliado sus aplicaciones a muchos campos de la ingeniería, enfocándose especialmente en la robustez de los controladores difusos a las incertidumbres de un sistema no lineal o a perturbaciones externas (Yeong y Chung, 1994).

3.3.2 Sistemas difusos basados en controladores de estructura variable

Los sistemas de control de estructura variable (SCEV) con modos deslizantes (MD) se caracterizan por la facilidad con la que tratan no linealidades e incertidumbres en los sistemas de control. A parte de su aplicación principal que es la estabilización de sistemas, las técnicas del CEV comienzan a aplicarse al control de sistemas de seguimiento de trayectoria, control óptimo y control adaptable (ver el apéndice B).

En distintos trabajos de investigación, un gran grupo de controladores difusos de sistemas no lineales se diseñan en el plano de fase determinado por el error (e) y su derivada (\dot{e}), (Ray y Mayumder, 1984), (Tang y Mullholand, 1987) y (Wakileh y Gill, 1988), calculando un valor difuso de la variable de control según los valores difuso e y \dot{e} .

El enfoque general del diseño de control se hace por la división de plano de fase en dos semiplanos por una línea de conmutación. Dentro de los semiplanos, se producen salidas positivas y negativas de control respectivamente, lo que pone de manifiesto la similitud entre las reglas difusas del CLD y la salida del controlador de estructura variable, como se observa en la **¡Error! No se encuentra el origen de la referencia.** y la figura 3.8 sistema de estructura variable

La tabla 3.8 representa la base de reglas de un controlador difuso en el cual las entradas son el error y su derivada. Los valores lingüísticos de la variable error son: error negativo grande (ENG), error negativo pequeño (ENP), cero (ECR), error positivo pequeño (EPP) y error positivo grande (EPG). De igual manera para la derivada del error los valores lingüísticos son: derivada del error positiva grande (DEPG), derivada del error positiva

pequeña (DEPG), cero (DECR), derivada del error negativa pequeña y negativa grande (NPDE y NGDE).

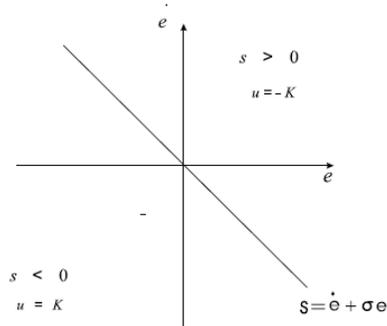


Figura 3.8 Sistema de estructura variable

Tabla 3-2 Base de reglas de un CLD

e' / e	ENG	ENP	ECR	EPP	EPG
DEPG	CRK	NGK	NGK	NGK	NGK
DEPP	PGK	CRK	NPK	NPK	NGK
DECR	PGK	PPK	CRK	NPK	NGK
NPDE	PGK	PPK	PPK	CRK	NGK
NGDE	PGK	PGK	PGK	PGK	CRK

La desventaja del MD está en los cambios drásticos de las variables de control que provocan alto estrés en los actuadores, como por ejemplo, el ruido continuo. Mientras que en CLD la magnitud de las salidas de control depende de la distancia del vector de estado a la línea de conmutación, suavizando el control. Si los límites de las incertidumbres del modelo son conocidos, se puede garantizar la estabilidad y la alta calidad del comportamiento del sistema controlado. En principio, el controlador difuso funciona como un controlador de modo deslizante modificado. Las reglas están, en general condicionadas en tal manera que por encima de la línea de conmutación $s=0$ se genera un control negativo y por debajo de la misma se genera un control positivo, similar al MD.

Basándose en la similitud entre el CLD y el CEV se ha analizado la robustez del CLD para una clase de sistemas dinámicos no lineales con incertidumbres de un proceso, en el sentido de Lyapunov en Yi y Chung (1994).

En el presente trabajo se propone un controlador robusto difuso para robots manipuladores con incertidumbres, basándonos en el análisis de estabilidad de un CLD propuesto por Yi y Chung (1997).

3.3.3 Controlador difuso propuesto

Generalidades

Las ecuaciones dinámicas de un robot manipulador de n grados de libertad puede ser descrita como:

$$M(q)\ddot{q} + C(\dot{q}, q)\dot{q} + g(q) + \Delta = \tau \quad q, \tau, \Delta \in \mathbb{R}^n \quad (2.25)$$

donde $M(q) \in \mathbb{R}^{n \times n}$ es la matriz de inercia, $C(\dot{q}, q)\dot{q} \in \mathbb{R}^n$ representa las fuerzas centrífugas y de Coriolis, y $g(q) \in \mathbb{R}^n$ es el vector de fuerzas gravitacionales. Δ representa incertidumbres no estructuradas, las cuales están acotadas de forma superior como:

$$|\Delta_i| \leq \Delta_{\max, i} \quad i = 1, 2, \dots, n \quad (2.26)$$

Una propiedad que presenta la matriz de inercia y de Coriolis es la siguiente (ver apéndice A, propiedad 7 de la matriz de Coriolis):

$$N(q, \dot{q}) = \frac{1}{2} \dot{M}(q) - C(\dot{q}, q) \quad (2.27)$$

$N(q, \dot{q})$ es antisimétrica y además se cumple que el modelo dinámico del robot puede ser representado por:

$$M(q)\ddot{q} + C(\dot{q}, q)\dot{q} + g(q) = Y(q, \dot{q}, \ddot{q}) \quad (2.28)$$

$$Y \in \mathbb{R}^{n \times p}, \theta \in \mathbb{R}^p$$

donde θ es un vector de parámetros de inercia (*masa, centro de masa e inercia*) y $Y(q, \dot{q}, \ddot{q})$ consiste de funciones conocidas de las coordenadas generalizadas. Suponiendo que se conoce los parámetros de inercia, la incertidumbre estructural y su cota superior puede ser representada como:

$$|\theta_{e,i}| = |\theta_{0,i} - \theta_i| \leq \rho_i \quad i = 1, 2, \dots, p \quad (2.29)$$

donde $\theta_{0,i}$ es el valor nominal conocido de θ_i y $\theta_{e,i}$ denota el error de modelado. Para una trayectoria representada por q_d ¹ el error de seguimiento estaría dado por:

$$\tilde{q} = q - q_d \quad (2.30)$$

Procedimiento de diseño del controlador difuso

Existen varios elementos de diseño en los CLD tales como: 1) definición de las variables de entrada con la forma de sus conjuntos difusos; 2) definición de las variables de salida con la forma de sus conjuntos difusos; 3) método de razonamiento y defuzificación; 4) reglas del controlador.

Definición de las variables de entrada y salida.

Se han considerado diferentes variables para utilizarse en el controlador, pero se decidió utilizar como variables el *error de seguimiento* (\tilde{q}) y *el error de velocidad* ($\dot{\tilde{q}}$). Se utilizó un controlador difuso desacoplado, por lo tanto, dos controladores son implementados uno para cada articulación, cada controlador tiene dos entradas ($\tilde{q}_i, \dot{\tilde{q}}_i$) y una salida ($\tau_{\tilde{f}_i}$) como se muestra en la figura 3.9. El término desacoplado es usado debido a que los controladores operan independientes uno del otro, es decir, no hay transferencia de información entre ellos (Passino y Yurkovich, 1998).

¹ En adelante, las letras en negritas indican vectores y matrices, mientras que las letras normales indican escalares.

Funciones de pertenencia para las entradas y la salida

Establecidas las entradas del controlador, como siguiente paso del diseño es asignar el tipo de funciones de pertenencia para cada conjunto difuso.

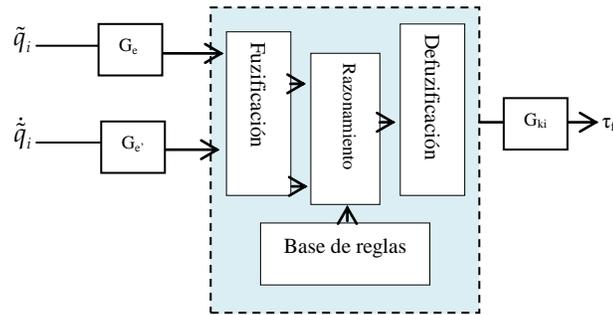


Figura 3.9 Esquema general del controlador

Para la variable error de seguimiento los conjuntos difusos usados son de tipo triangular y los valores lingüísticos son los siguientes: error positivo grande (EPG), error positivo pequeño (EPP), error negativo grande (ENG), error negativo pequeño (ENP) y cero (ECR). Para facilitar el diseño se eligen los centros de cada función de membresía, por lo tanto se tiene que:

$$C_{ENP} = -C_{EPP}, \quad C_{ENG} = -C_{EPG} \quad (2.31)$$

Por simplicidad, cada función de membresía es elegida de tal forma que únicamente dos funciones se crucen entre el valor central de sus funciones vecinas. La figura 3.10 muestra los conjuntos difusos descritos anteriormente.

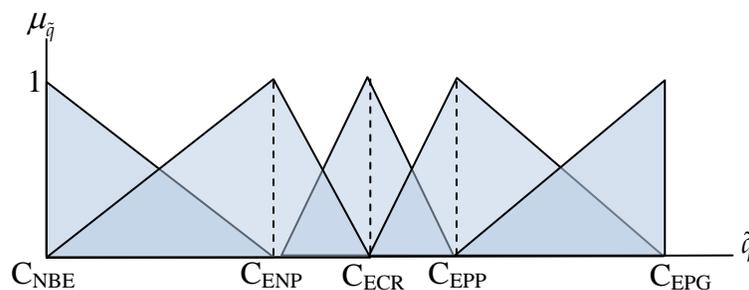


Figura 3.10 Funciones de pertenencia para la variable de entrada error de posición y error de velocidad

Para la variable **error de velocidad**, se utilizaron funciones de tipo triangular y las variables lingüísticas son las siguientes: derivada del error positivo grande (DEPG), derivada del error positivo pequeño (DEPP), derivada del error negativo grande (NGDE), derivada del error negativo pequeño (NPDE) y cero (DECR). La variable error de velocidad está definida como:

$$\dot{\tilde{q}} = \dot{q} - \dot{q}_d \quad (2.32)$$

$$C_{DENP} = -C_{DEPP}, \quad C_{DENG} = -C_{DEPG} \quad (2.33)$$

La figura 3.11 muestra las funciones de pertenencia para los conjuntos difusos descritos anteriormente.

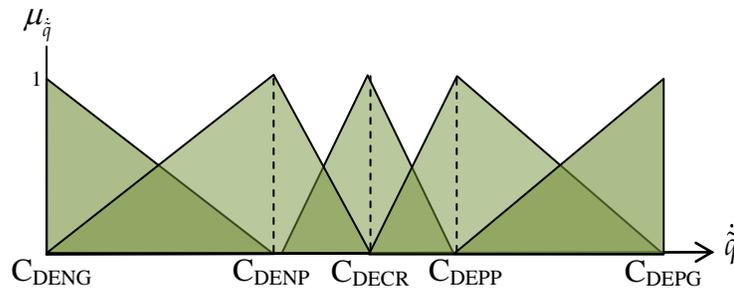


Figura 3.11 Funciones de pertenencia para la variable de entrada "error de velocidad"

Para la variable de salida **par**, se utilizaron funciones de tipo singleton, dado que no se requiere demasiado costo computacional. Los valores lingüísticos son los siguientes: positivo grande (PGK), positivo pequeño (PPK), negativo grande (NGK), negativo pequeño (NPK) y cero (CRK). La figura 3.12 muestra las funciones de pertenencia para los conjuntos difusos de salida.

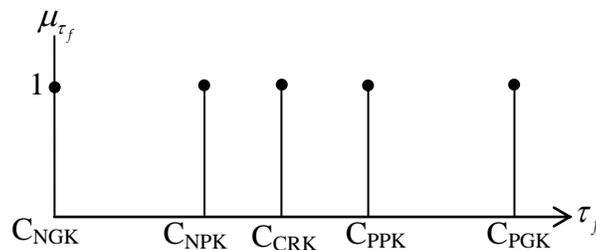


Figura 3.12 Funciones de pertenencia para la salida

Todos los conjuntos difusos de entrada y salida se diseñaron en el espacio normalizado, es decir, en el intervalo de -1 a 1. Las ganancias $G_e, G_{\dot{e}}$ y G_k que se observan en la Figura 3.9 a la entrada y salida del controlador difuso sirven como factor de escalamiento entre el espacio normalizado y el correspondiente espacio real,

$$\begin{aligned}\tilde{q}_n &= G_e \cdot \tilde{q} \\ \dot{\tilde{q}}_n &= G_{\dot{e}} \cdot \dot{\tilde{q}} \\ \tau_f &= G_k \cdot \tau_n\end{aligned}\tag{2.34}$$

además, los centros de las funciones de membresía se eligen de tal forma que se utilice completamente el espacio normalizado, por lo tanto se debe cumplir que:

$$\begin{aligned}C_{ENG}(=-1) &< C_{ENP} < C_{ECR} < C_{EPP} < C_{EPG}(=1) \\ C_{DENG}(=-1) &< C_{DENP} < C_{DECR} < C_{DEPP} < C_{DEPG}(=1) \\ C_{NGK}(=-1) &< C_{NPK} < C_{CRK} < C_{PPK} < C_{PGK}(=1)\end{aligned}\tag{2.35}$$

Definición de la base de conocimiento.

Una vez que se han descrito las variables de entrada con sus respectivos conjuntos difusos, es necesario establecer las reglas que conforman la base de conocimiento que regirán el comportamiento del controlador. Para obtener las reglas difusas se recurrió al presentado por Yi y Chung (1997), denominado UNLP por sus siglas en ingles *upper-diagonal-negative, lower-diagonal-positive*.

El procedimiento para obtener la base de conocimiento se consigue utilizando la teoría de modos deslizantes, primero se parte de colocar los conjuntos difusos de la entrada y la salida en el plano de fase del error, como se muestra en la Figura 3.13, donde se observa que los pares de valores $C_{ENG}, C_{DEPG}, C_{EPG}, C_{NGDE}$ determinan una línea diagonal en el plano de fase de error normalizado, donde:

$$\begin{aligned}\dot{\tilde{q}}_n + \tilde{q}_n &= 0 \\ G_{\dot{e}} * \dot{\tilde{q}} + G_e \tilde{q} &= 0\end{aligned}\tag{2.36}$$

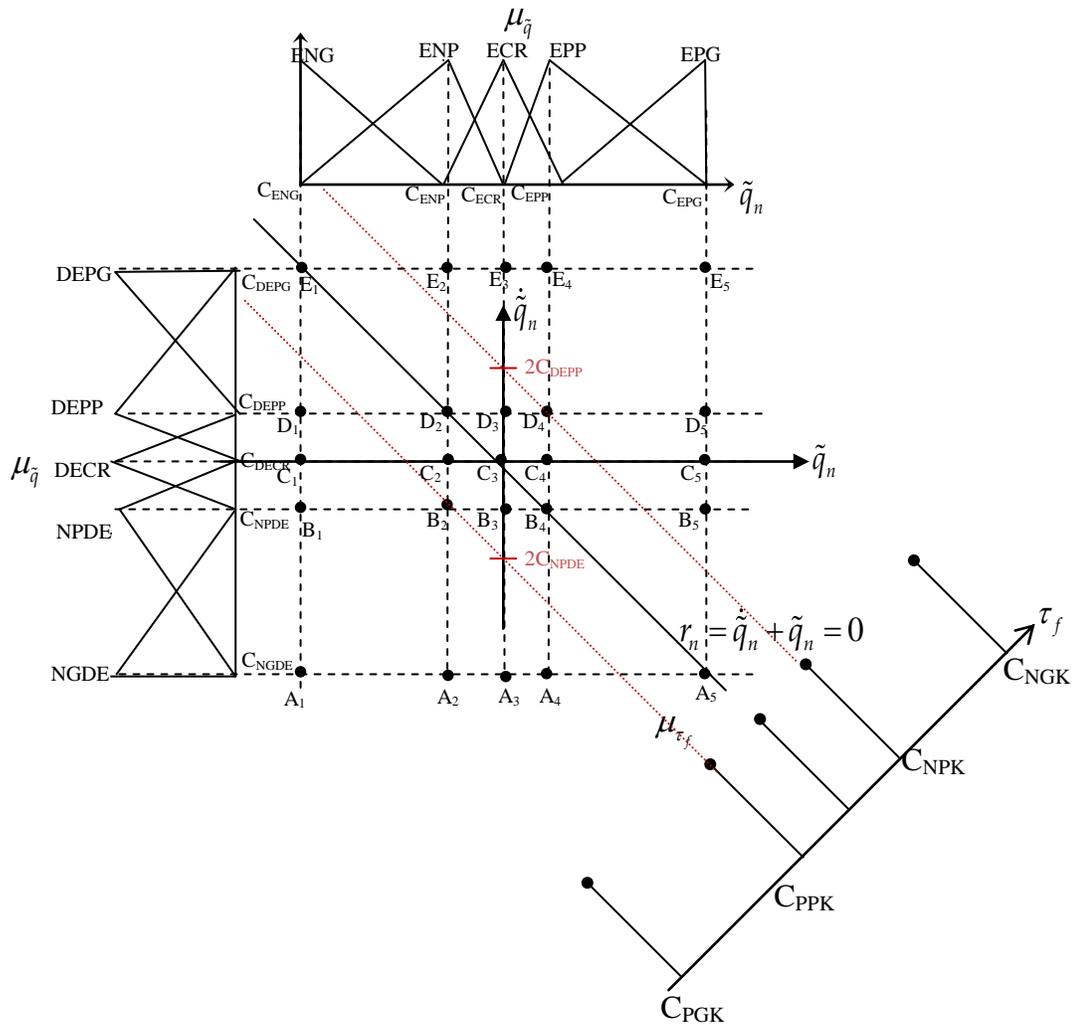


Figura 3.13 Funciones de membresía en el plano de fase del error

Dado que los universos de discurso del error de posición y el error de velocidad tienen 5 conjuntos difusos cada uno, el plano de fase se divide en 25 áreas de acuerdo al cruce entre el error de seguimiento y el de velocidad como lo muestra la Figura 3.13. El centro de cada área es expresado con un punto etiquetado desde A_1 a E_5 . Dichos puntos corresponden a cada regla difusa, de la forma si- entonces, que conforman a la base de conocimiento.

Para asignar la salida correspondiente a determinadas entradas y obtener la base de reglas difusas se parte del siguiente procedimiento:

- 1) Trazar dos líneas paralelas a r_n en los puntos donde $\dot{\tilde{q}}_n = 2C_{DEPP}$ en el plano de fase del error. En la Figura 3.13 están marcadas con las líneas en rojo.
- 2) Asignar el conjunto difuso de salida CRK al punto donde $r_n=0$.
- 3) Asignar el conjunto difuso de salida NPK a la región donde $0 < r_n < 2C_{DEPP}$.
- 4) Asignar el conjunto difuso de salida PPK a la región donde $2C_{DENP} < r_n < 0$.
- 5) Asignar el conjunto difuso de salida NGK arriba de la diagonal donde $r = 2C_{DEPP}$
- 6) Asignar el conjunto difuso de salida PGK debajo de la diagonal donde $r = 2C_{DENP}$

Con este procedimiento se obtienen la Tabla 3-3, donde se resumen las 25 reglas de la forma si-entonces.

Tabla 3-3 Reglas difusas con el modelo UNLP

$\dot{\tilde{q}} \setminus \tilde{q}$	ENG	ENP	ECR	EPP	EPG
DEPG	CRK	NGK	NGK	NGK	NGK
DEPP	PGK	CRK	NPK	NPK	NGK
DECR	PGK	PPK	CRK	NPK	NGK
DENP	PGK	PPK	PPK	CRK	NGK
DENG	PGK	PGK	PGK	PGK	CRK

Por ejemplo, la regla formada por la fila 1 y columna 5, indica que: **si** el error de posición es positivo grande **y** el error de velocidad es positivo grande, **entonces**, el par será negativo grande. La señal de control negativa revierte este comportamiento, dado que si fuera positiva, el error incrementaría más y más. Este proceder de la base de conocimiento implica la similitud entre los controladores de estructura variable y el CLD.

Hay una relación entre el error de plano de fase real y el normalizado con las ganancias del CLD, multiplicando la ecuación (2.36) por $1/G_c$ se obtiene lo expresado en la ecuación (2.37).

$$\begin{aligned}\dot{\tilde{q}} + \frac{G_e}{G_{\dot{e}}} \tilde{q} &= 0 \\ \dot{\tilde{q}} + \lambda \tilde{q} &= 0, \quad \lambda = \frac{G_e}{G_{\dot{e}}} > 0\end{aligned}\tag{2.37}$$

El parámetro $r = \dot{\tilde{q}} + \lambda \tilde{q}$ representa una medición del error del sistema de control en lazo cerrado.

Método de inferencia y defuzificación.

La inferencia es el proceso mediante el cual se genera un mapeo para asignar a una entrada una salida utilizando lógica difusa. El proceso de inferencia provee las bases para la toma de decisiones del sistema. Este proceso involucra la utilización de funciones de membresía y las reglas generadas en la base de conocimiento. En este proyecto se utilizó la inferencia del mínimo, es decir se calcula la relación que existe entre las entradas generadas, utilizando el mínimo valor del grado de membresía de dichas entradas, es decir:

$$\mu_{i,prem} = \min\{\mu_{\tilde{q}_m}, \mu_{\tilde{q}_n}\} \quad i = 1, \dots, l\tag{2.38}$$

Con el subconjunto difuso encontrado $\mu_{i,prem}$, donde i indica el número de regla difusa, se obtiene la salida correspondiente de acuerdo a la regla activada, donde el proceso de defuzificación convierte este valor difuso en un valor físico de salida.

Por último se realiza la defuzificación mediante el método del promedio de centros, cuya implementación es muy práctica.

$$\tau_{f_n} = \frac{\sum_{k=1}^l \mu_k(k) \cdot k}{\sum_{k=1}^l \mu_k(k)}\tag{2.39}$$

Donde k representa el centro de las funciones de membresía de salida las cuales son singleton ($C_{NGK} \dots C_{PGK}$) y μ_k el grado de membresía que se obtuvo en la inferencia. Si se

utilizan funciones triangulares en la salida, el resultado es igual dado que con este método no importa la forma de la función de membresía.

Aquí se concluye con el diseño del controlador, en los epígrafes que siguen se utilizó el análisis desarrollado por Yi y Chung (1997), así también se tomaron como referencias los teoremas necesarios de dichos autores para demostrar la convergencia del sistema en lazo cerrado a una vecindad cercana al punto de equilibrio.

3.3.4 Análisis del controlador lógico difuso

De la **¡Error! No se encuentra el origen de la referencia.**, se pueden resumir las siguientes propiedades del controlador difuso propuesto (Palm, 1992).

- * $\tau_f \approx 0$, cerca de la línea diagonal, $r=0$.
- * $\tau_f < 0$, arriba de la línea diagonal, $r > 0$.
- * $\tau_f > 0$, debajo de la línea diagonal, $r < 0$.
- * La magnitud de la salida tiende a incrementar de acuerdo a la distancia entre $r=0$ y los estados del sistema.

El siguiente teorema describe las propiedades anteriores de manera formal:

Teorema 1. Existencia de ϕ : para cualquier $k_\alpha \in [0, G_k]$, existe un límite ϕ tal que la salida del controlador difuso τ_f (Yi y Chung, 1997), satisface la ecuación:

$$\tau_f = -k_f \frac{r}{|r|} \quad \text{en} \quad |r| \geq \phi \tag{2.40}$$

donde $k_f > k_\alpha$ y $r = e + \lambda \dot{e}$

Demostración. Considérese una línea perpendicular s^\perp a $r=0$ como se muestra en la Figura 3.14.

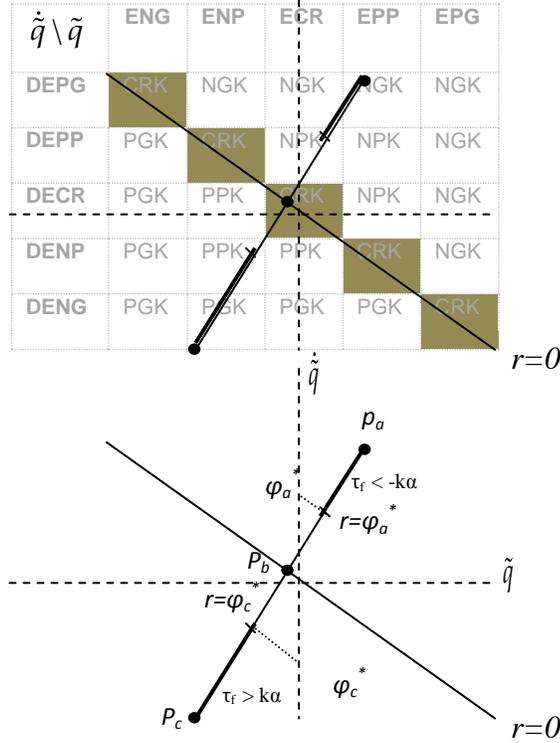


Figura 3.14 Reglas difusas y el plano de fase del error

Observando a la Figura 3.14, es claro que los puntos p_a , p_c y el punto de cruce p_b de s^\perp , la salida del CLD corresponde a:

$$\begin{aligned}
 \tau_f(\tilde{q}, \dot{q}) \Big|_{p_a} &= G_{k_{\text{CRK}}} = -G_K \\
 \tau_f(\tilde{q}, \dot{q}) \Big|_{p_b} &= G_{k_{\text{CRK}}} = 0 \\
 \tau_f(\tilde{q}, \dot{q}) \Big|_{p_c} &= G_{k_{\text{PGK}}} = G_K
 \end{aligned} \tag{2.41}$$

dado que la salida es continua, para cualquier $k_\alpha \in [0 G_K]$, existen puntos ϕ_a^* y ϕ_c^* en s^\perp , tal que:

$$\begin{aligned}\tau_f(\tilde{q}, \dot{\tilde{q}}) &< -k_\alpha \quad \text{en } r > \phi_a^* \\ \tau_f(\tilde{q}, \dot{\tilde{q}}) &> k_\alpha \quad \text{en } r < \phi_c^*\end{aligned}\tag{2.42}$$

estos argumentos son aplicables para todo s^\perp . Como una consecuencia de $\phi = \max\left\{\left|\phi_a^*\right|, \left|\phi_c^*\right|\right\}$, $\tau_f(\tilde{q}, \dot{\tilde{q}})$ se expresa en todo el plano de fase como sigue:

$$\begin{aligned}\tau_f(\tilde{q}, \dot{\tilde{q}}) &< -k_\alpha \quad \text{en } r > \phi \\ \tau_f(\tilde{q}, \dot{\tilde{q}}) &> k_\alpha \quad \text{en } r < \phi\end{aligned}\tag{2.43}$$

La ecuación (2.43) puede ser reescrita como la ecuación (2.40). El teorema 1, muestra que el CLD con la base de reglas UNLP, puede ser considerado como una extensión de un controlador de estructura variable. Adicionalmente, si los valores centrales de las funciones de pertenencia de la entrada y la salida satisfacen la ecuación (2.44):

$$\begin{aligned}C_{ECR} = 0, C_{EPP} = -C_{ENP}, C_{EPG} = -C_{ENG}, &= 1, \\ C_{DECR} = 0, C_{DEPP} = -C_{DENP}, C_{DEPG} = -C_{DENG}, &= 1\end{aligned}\tag{2.44}$$

$$C_{EPP} = C_{DEPP}, \quad C_{ENP} = C_{DENP}\tag{2.45}$$

entonces las magnitudes de ϕ y k_α del teorema 1 pueden ser descritas por los parámetros de diseño del CLD de acuerdo al siguiente teorema.

Teorema 2. Magnitud de k_α y ϕ : si $C_{PSK} < \frac{1}{2}C_{PBK}$, entonces para $k_\alpha = G_k C_{PPK}$, τ_f puede caracterizarse con respecto a $\phi = 2C_{PSE} / G_{\dot{q}}$ y $k_f \geq k_\alpha$ como (Yi y Chung ,1997):

$$\tau_f = -k_f \frac{r}{|r|} \quad \text{en } |r| \geq \phi\tag{2.46}$$

Demostración.

El plano de fase del error normalizado se muestra en la **x** el cual corresponde al plano de fase del error de la Figura 3.14 con las ganancias $G_{\dot{q}}$ y $G_{\ddot{q}}$. De la se extrajeron un conjunto

de puntos ϕ_n^* y $-\phi_n^*$ para $k_{n,\alpha} = C_{NSK}$ y $k_{n,\alpha} = C_{PSK}$, representados por la línea continua en negritas, en el que $\tau_{n,f}(\tilde{q}_n, \dot{\tilde{q}}_n)$ está dado por:

$$\begin{aligned} \tau_{n,f}(\tilde{q}_n, \dot{\tilde{q}}_n) &= C_{NSK} \quad \text{en} \quad r_n = \phi_n^* \\ \tau_{n,f}(\tilde{q}_n, \dot{\tilde{q}}_n) &= C_{PSK} \quad \text{en} \quad r_n = -\phi_n^* \end{aligned} \quad (2.47)$$

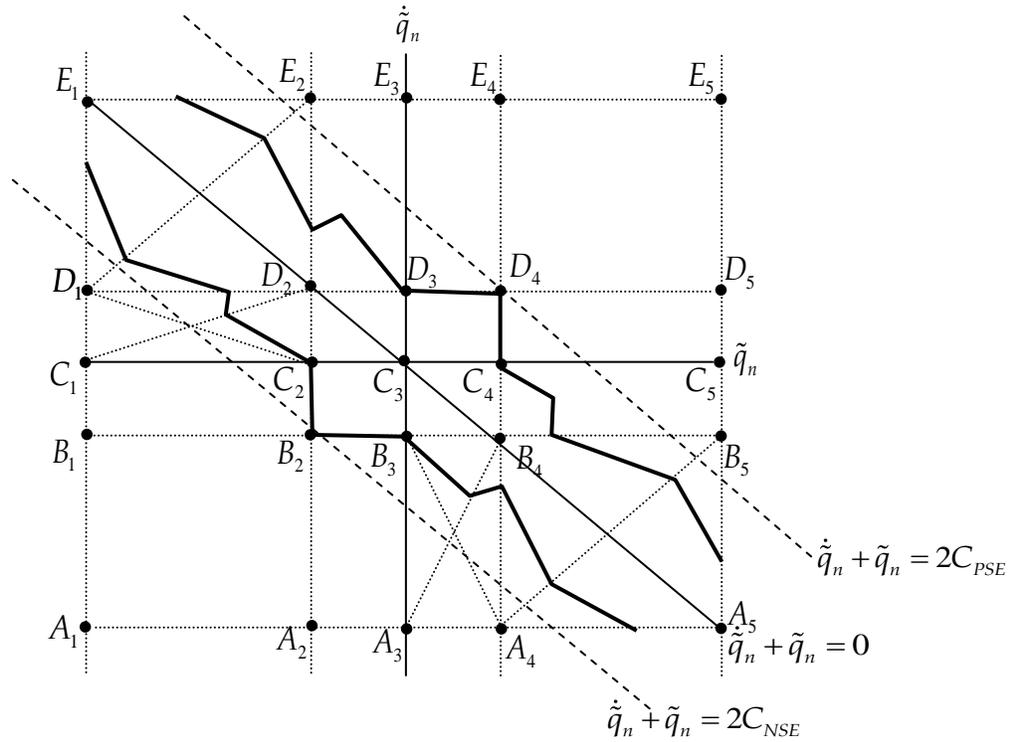


Figura 3.15 Conjunto de puntos en el plano de fase del error normalizado

donde $r_n = \dot{\tilde{q}}_n + \tilde{q}_n$, aquí se considera la relación entre las ecuaciones (2.44) y (2.45), por lo tanto el conjunto de puntos ϕ_n^* y $-\phi_n^*$ son simétricos con respecto a $r_n = 0$. Aplicando los mismos argumentos presentados en el teorema 1, $\tau_{n,f}(\tilde{q}_n, \dot{\tilde{q}}_n)$ es escrito como

$$\begin{aligned}
\tau_{n,f}(\tilde{q}_n, \dot{\tilde{q}}_n) &< C_{NSK} \quad \text{en } r_n > \phi_n^* \\
\tau_{n,f}(\tilde{q}_n, \dot{\tilde{q}}_n) &> C_{PSK} \quad \text{en } r_n < -\phi_n^*
\end{aligned} \tag{2.48}$$

donde $\phi_n = \max\left\{\left|\phi_n^*\right|, \left|-\phi_n^*\right|\right\} = 2C_{EPP}$, como lo muestra la figura 3.15 y dado que

$\tilde{q}_n = Ge \cdot \tilde{q}$, $\dot{\tilde{q}}_n = Ge \cdot \dot{\tilde{q}}$ y $\tau_f = G_k \cdot \tau_{n,f}(\tilde{q}_n, \dot{\tilde{q}}_n)$ corresponde a

$$\begin{aligned}
\tau_f &< G_k \cdot C_{NPK} \quad \text{en } r > \phi \\
\tau_f &> G_k \cdot C_{PPK} \quad \text{en } r < -\phi
\end{aligned} \tag{2.49}$$

donde $\phi = \frac{2C_{EPP}}{G_e}$. De acuerdo al teorema 2, es posible ajustar al parámetro ϕ con los parámetros de diseño del CLD.

Análisis

Supóngase ahora que se tiene un CLD para cada eslabón de un manipulador de n vínculos², es decir,

$$\begin{aligned}
\mathbf{r} &= \dot{\tilde{\mathbf{q}}} + \mathbf{A}\tilde{\mathbf{q}} \quad \text{donde } \mathbf{A} = \text{diag}\{\lambda_i\} \\
\lambda_i &= \frac{G_e}{G_e} > 0, \quad i = 1, \dots, n
\end{aligned} \tag{2.50}$$

Para analizar el sistema en lazo cerrado utilizando la caracterización del CLD (2.40), la ecuación (2.25) se puede expresar en términos de la ecuación (2.51) (Spong, 1992).

² De aquí en adelante, un parámetro de cada CLD será denotado por el subíndice i , $i=1, \dots, n$

$$\begin{aligned}
r &= \ddot{\tilde{q}} + \Lambda \dot{\tilde{q}} \quad \text{dado que } \tilde{q} = q - q_d \text{ y } \dot{\tilde{q}} = \dot{q} - \dot{q}_d, \text{ sustituyendo} \\
r &= \dot{q} - \dot{q}_d + \Lambda \tilde{q} \\
\text{ademas si } \dot{q}_d - \Lambda \tilde{q} &= v \text{ y } a = \dot{v} \text{ entonces } r = \dot{q} - v \quad \dot{r} = \ddot{q} - a
\end{aligned} \tag{2.51}$$

Restando el término $M(q)a + C(q, \dot{q})v + g(q)$ a ambos lados de la ecuación (2.25) se obtiene

$$M(\ddot{q}) + C(q, \dot{q})\dot{q} + \Delta_d + [M(q)a + C(q, \dot{q})v + g(q)] = \tau \tag{2.52}$$

agrupando términos se obtiene

$$M(q)(\ddot{q} - a) + C(q, \dot{q})(\dot{q} - v) + \Delta_d + [M(q)a + C(q, \dot{q})v + g(q)] = \tau \tag{2.53}$$

Utilizando las expresiones mostradas en (2.51) para r y la propiedad presentada en la ecuación (2.28),

$$M(q)\dot{r} + C(q, \dot{q})r + Y(q, \dot{q}, v, a)\theta + \Delta_d = \tau \tag{2.54}$$

El siguiente teorema describe la robustez del CLD propuesto consistiendo de la ley de control (Yi y Chung, 1997), (2.55) y la ecuación (2.54)

$$\tau = -Kr + Y(q, \dot{q}, v, a)\theta_o + \tau_f \tag{2.55}$$

donde $K \in \mathbb{R}^{n \times n}$, es una matriz positiva definida, θ_o es un vector de parámetros nominales conocidos de θ y τ_f es el vector de salida del CLD para eliminar las incertidumbres paramétricas y perturbaciones, es decir $Y(\theta - \theta_o) + \Delta$. Para analizar la robustez del sistema en lazo cerrado, se utiliza el teorema de la cota final (*ultimate boundedness*) de un sistema dinámico, el cual implica que a pesar de las incertidumbres, los estados de un sistema

ingresan eventualmente y permanecen en una región pequeña cercana a un punto de equilibrio en el espacio de estados.

Teorema 3. Cota final: Si la ganancia de salida $G_{k,i} = 1, \dots, n$ del CLD satisface (2.56), entonces el sistema en lazo cerrado tiene una cota final (Yi y Chung ,1997).

$$G_{k,i} C_{PPK,i} = |Y(q, \dot{q}, a, v) \rho|_i + \Delta_{\max,i} \quad i = 1, \dots, n \quad (2.56)$$

donde $\rho = [\rho_1, \dots, \rho_p]^T$ y $|Y(q, \dot{q}, a, v) \rho|_i$ es la magnitud del i-esimo elemento de $Y(q, \dot{q}, a, v) \rho \in \mathbb{R}^n$.

Demostración.

Sustituyendo (2.55) en la ecuación (2.54)

$$M(q)\dot{r} + C(q, \dot{q})r + \Delta_d = -Kr + Y(q, \dot{q}, v, a)(\theta_o - \theta) + \tau_f \quad (2.57)$$

se elige la siguiente función candidata de Lyapunov:

$$V = \frac{1}{2} r^T M(q)r + \tilde{q}^T AK\tilde{q} \quad (2.58)$$

diferenciando (2.58)

$$\dot{V} = r^T M(q)\dot{r} + \frac{1}{2} r^T \dot{M}(q)r + 2\tilde{q}^T AK\dot{\tilde{q}} \quad (2.59)$$

insertando (2.57) se obtiene

$$\dot{V} = r^T \left[-(C(q, \dot{q}) + K)r + Y(q, \dot{q}, a, v)\theta_e + \tau_f - \Delta \right] + \frac{1}{2} r^T \dot{M}(q)r + 2\tilde{q}^T AK\dot{\tilde{q}} \quad (2.60)$$

Utilizando la propiedad presentada en la ecuación (2.27) da como resultado:

$$\dot{V} = \mathbf{r}^T \left[-\mathbf{K}\mathbf{r} + \mathbf{Y}(\mathbf{q}, \dot{\mathbf{q}}, \mathbf{a}, \mathbf{v})\boldsymbol{\theta}_e + \boldsymbol{\tau}_f - \boldsymbol{\Delta} \right] + 2\tilde{\mathbf{q}}^T \boldsymbol{\Lambda} \mathbf{K} \tilde{\mathbf{q}} \quad (2.61)$$

Sustituyendo $\mathbf{r} = \dot{\tilde{\mathbf{q}}} + \boldsymbol{\Lambda} \tilde{\mathbf{q}}$ en (2.61) y reduciendo términos se obtiene

$$\dot{V} = -\mathbf{x}^T \mathbf{Q} \mathbf{x} + \mathbf{r}^T [\mathbf{Y}(\mathbf{q}, \dot{\mathbf{q}}, \mathbf{a}, \mathbf{v})\boldsymbol{\theta}_e - \boldsymbol{\Delta}] + \mathbf{r}^T \boldsymbol{\tau}_f \quad (2.62)$$

donde $\mathbf{x} = [\tilde{\mathbf{q}}^T, \dot{\tilde{\mathbf{q}}}^T]^T$ y $\mathbf{Q} = \begin{bmatrix} \boldsymbol{\Lambda}^T \mathbf{K} \boldsymbol{\Lambda} & 0 \\ 0 & \mathbf{K} \end{bmatrix}$.

Para obtener las cotas superiores de \dot{V} en cada intervalo de tiempo, se realiza una partición de \mathbf{R}^+ en $L_1 = \{t \mid |r_i| \geq \phi_i \forall i\}$ y $L_2 = \{t \mid |r_i| < \phi_i \exists i\}$.

i) Cuando $t \in L_1$, de la ecuación (2.62) junto con (2.40) y (2.56), la cota sobre \dot{V} estaría dada por:

$$\begin{aligned} \dot{V} &\leq -\mathbf{x}^T \mathbf{Q} \mathbf{x} + \sum_{i=1}^n |r_i| \{ |\mathbf{Y}(\mathbf{q}, \dot{\mathbf{q}}, \mathbf{a}, \mathbf{v})\boldsymbol{\rho}|_i + \Delta_{\max,i} \} + \sum_{i=1}^n |r_i| \cdot -k_{f,i} \frac{|r_i|}{|r_i|} \\ &\leq -\lambda_{\min}(\mathbf{Q}) \cdot \|\mathbf{x}\|^2 \quad \text{utilizando la propiedad A.16 del apéndice A} \end{aligned} \quad (2.63)$$

donde λ_{\min} es el eigenvalor mínimo de la matriz \mathbf{Q} .

ii) Considérese ahora el caso cuando $t \in L_2$. Dado que los conjuntos difusos de salida están definidos en el espacio normalizado, está claro que $|\tau_{f,i}| \leq G_{k,i} \quad \forall i$, entonces de (2.56) se adquiere:

$$|\tau_{f,i}| \leq \frac{1}{C_{PPK,i}} \left\{ \|\mathbf{Y}(\mathbf{q}, \dot{\mathbf{q}}, \mathbf{a}, \mathbf{v})\boldsymbol{\rho}\|_i + \Delta_{\max,i} \right\} \quad \forall i. \quad (2.64)$$

insertando (2.64) y (2.56) en (2.62), \dot{V} puede ser escrita como

$$\begin{aligned}
\dot{V} &\leq -\mathbf{x}^T \mathbf{Q} \mathbf{x} + \sum_{i \in l} r_i \{ \mathbf{Y}(\mathbf{q}, \dot{\mathbf{q}}, \mathbf{a}, \mathbf{v}) \rho + \Delta \}_i + \sum_{i \in l} r_i \tau_{f,i} + \sum_{i \notin l} r_i \{ \mathbf{Y}(\mathbf{q}, \dot{\mathbf{q}}, \mathbf{a}, \mathbf{v}) \rho + \Delta \}_i + \sum_{i \notin l} r_i \tau_{f,i} \\
&\leq -\mathbf{x}^T \mathbf{Q} \mathbf{x} + \sum_{i \in l} |r_i| \{ | \mathbf{Y}(\mathbf{q}, \dot{\mathbf{q}}, \mathbf{a}, \mathbf{v}) \rho |_i + \Delta_{\max,i} \} + \sum_{i \in l} |r_i| | \tau_{f,i} | + \sum_{i \notin l} |r_i| \{ | \mathbf{Y}(\mathbf{q}, \dot{\mathbf{q}}, \mathbf{a}, \mathbf{v}) \rho |_i + \Delta_{\max,i} \} \\
&\quad + \sum_{i \notin l} r_i \cdot -k_{f,i} \frac{r_i}{|r_i|}
\end{aligned}$$

dado que $|r_i| < \phi_i$, entonces

$$\begin{aligned}
\dot{V} &< -\mathbf{x}^T \mathbf{Q} \mathbf{x} + \sum_{i \in l} \phi_i \{ | \mathbf{Y}(\mathbf{q}, \dot{\mathbf{q}}, \mathbf{a}, \mathbf{v}) \rho |_i + \Delta_{\max,i} \} + \sum_{i \in l} \phi_i | \tau_{f,i} | + \sum_{i \notin l} |r_i| \{ | \mathbf{Y}(\mathbf{q}, \dot{\mathbf{q}}, \mathbf{a}, \mathbf{v}) \rho |_i + \Delta_{\max,i} \} \\
&\quad + \sum_{i \notin l} r_i \cdot -k_{f,i} \frac{r_i}{|r_i|}
\end{aligned} \tag{2.65}$$

ya que $\sum_{i \notin l} |r_i| \{ | \mathbf{Y}(\mathbf{q}, \dot{\mathbf{q}}, \mathbf{a}, \mathbf{v}) \rho |_i + \Delta_{\max,i} \} + \sum_{i \notin l} r_i \cdot -k_{f,i} \frac{r_i}{|r_i|} < 0$ es posible acotar a \dot{V} como

$$\dot{V} \leq -\mathbf{x}^T \mathbf{Q} \mathbf{x} + \sum_{i \in l} \phi_i \{ | \mathbf{Y}(\mathbf{q}, \dot{\mathbf{q}}, \mathbf{a}, \mathbf{v}) \rho |_i + \Delta_{\max,i} \} + \sum_{i \in l} \phi_i | \tau_{f,i} | \tag{2.66}$$

y utilizando la ecuación (2.64) en (2.66)

$$\dot{V} \leq -\mathbf{x}^T \mathbf{Q} \mathbf{x} + \sum_{i \in l} \phi_i \cdot \frac{C_{PPK,i} + 1}{C_{PPK}} \{ | \mathbf{Y}(\mathbf{q}, \dot{\mathbf{q}}, \mathbf{a}, \mathbf{v}) \rho |_i + \Delta_{\max,i} \} \tag{2.67}$$

Cabe señalar que a partir de las propiedades de la dinámica del robot, se puede acotar a las incertidumbres como:

$$| \mathbf{Y}(\mathbf{q}, \dot{\mathbf{q}}, \mathbf{a}, \mathbf{v}) \rho |_i + \Delta_{\max,i} \leq k_{0,i} + k_{1,i} \| \dot{\mathbf{x}} \| + k_{2,i} \| \ddot{\mathbf{x}} \|^2 \tag{2.68}$$

donde $k_{0,i}, k_{1,i}$ y $k_{2,i}$ son constantes positivas y $Y(\mathbf{q}, \dot{\mathbf{q}}, \mathbf{a}, \mathbf{v})$ se define como en la ecuación (2.28). Es importante notar que $C_{PPK} + 1/C_{PPK}$ es una constante, por lo tanto

$$\frac{C_{PSK,i} + 1}{C_{PSK,i}} (k_{0,i} + k_{1,i} \|\mathbf{x}\| + k_{2,i} \|\mathbf{x}\|^2) = k_{0,i} + k_{1,i} \|\mathbf{x}\| + k_{2,i} \|\mathbf{x}\|^2 \quad (2.69)$$

y por lo tanto $k_{0,i} + k_{1,i} \|\mathbf{x}\| + k_{2,i} \|\mathbf{x}\|^2 < k_{0,i} + k_{1,i} \|\mathbf{x}\| + k_{2,i} \|\mathbf{x}\|^2$

Sustituyendo lo anterior en (2.67)

$$\dot{V} \leq -\lambda_{\min}(Q) \|\mathbf{x}\|^2 + \sum_{i \in I} \phi_i \left\{ k_{0,i} + k_{1,i} \|\mathbf{x}\| + k_{2,i} \|\mathbf{x}\|^2 \right\} \quad (2.70)$$

agregando y quitando el término $\lambda_{\min}(Q) \eta \|\mathbf{x}\|^2$ a la ecuación (2.70)

$$\dot{V} \leq -\lambda_{\min}(Q) \eta \|\mathbf{x}\|^2 - \lambda_{\min}(Q) (1 - \eta) \|\mathbf{x}\|^2 + \sum_{i \in I} \phi_i \left\{ k_{0,i} + k_{1,i} \|\mathbf{x}\| + k_{2,i} \|\mathbf{x}\|^2 \right\} \quad (2.71)$$

donde $0 < \eta < 1$ y si $\Phi_l = \left\{ \|\mathbf{x}\| \mid -\lambda_{\min}(Q) (1 - \eta) \|\mathbf{x}\|^2 = \sum_{i \in I} \phi_i \left[k_{0,i} + k_{1,i} \|\mathbf{x}\| + k_{2,i} \|\mathbf{x}\|^2 \right] \right\}$

$$\dot{V} \leq -\lambda_{\min}(Q) \eta \|\mathbf{x}\|^2, \quad \text{si } \|\mathbf{x}\| \geq \Phi_l \quad (2.72)$$

Finalmente, como resultado de evaluar ambos casos i) y ii), \dot{V} esta acotada de forma superior como

$$\dot{V} \leq -\lambda_{\min}(Q) \eta \|\mathbf{x}\|^2, \quad \text{si } \|\mathbf{x}\| \geq \Phi \quad (2.73)$$

Φ representa el peor de los casos de Φ_l . La ecuación (2.73) implica una cota última del sistema en lazo cerrado como lo define Khalil (1992).

El teorema desarrollado anteriormente presenta una construcción sistemática de un controlador robusto difuso, únicamente se necesita tener cierto conocimiento de las cotas sobre las incertidumbres, tal como lo mencionan Yi y Chung (1997).

Aquí se concluye el capítulo 3, en el cual se realizó el diseño del controlador y su posterior análisis. En el siguiente capítulo, se presentan las simulaciones y pruebas experimentales realizadas en un robot manipulador prototipo de dos grados de libertad, utilizando las herramientas descritas en este capítulo.

IV RESULTADOS

El capítulo se encuentra dividido en dos categorías, las simulaciones realizadas en el entorno Matlab® para el robot manipulador prototipo de dos grados de libertad y las pruebas experimentales realizadas sobre el mismo. Dado que parte de los objetivos es realizar una comparación de los algoritmos de control clásicos más implementados, el PD con compensación y el PID con el controlador robusto difuso, se presentan las simulaciones de cada controlador y sus resultados experimentales.

Al final del capítulo se muestra una tabla comparativa de acuerdo a la medición del error eficaz obtenido de cada controlador. El error cuadrático medio o eficaz se obtiene de acuerdo a la ecuación (4.1):

$$\tilde{q}_{rms} = \sqrt{\frac{1}{N} \sum_{j=1}^N (q_{i,j} - q_{d,i,j})^2} \quad (3.1)$$

donde N representa el número de muestras, q_i representa la posición actual y $q_{d,i}$ representa la posición deseada (Sharkawy, *et al.*, 2011).

4.1 Descripción de las pruebas realizadas

Las siguientes pruebas fueron realizadas para cada algoritmo de control a excepción de la última, ya que en este caso aplica solo para el control lógico difuso.

Prueba 1. Robot sin carga: se supone al robot sin carga en el eslabón 2, para el caso del PD Compensado también se supone al modelo dinámico conocido.

Prueba 2. Robot con carga: consistió de conjeturar que la masa del eslabón 2 ha cambiado debido a que el manipulador tomo un objeto con un peso del 30% más de la masa de dicho

eslabón en un tiempo de 25 a 85 s., dado que se modifica su masa, la inercia y el centro de masa cambiaran.

Prueba 3. Método de defuzificación por promedio de áreas y funciones triangulares a la salida. Aplica únicamente el controlador lógico difuso, se cambio el tipo de defuzificador de promedio de centros a promedio de áreas, utilizando funciones triangulares en el consecuente.

4.2 Controlador robusto difuso

4.2.1 Parámetros y ganancias del controlador

El método de diseño descrito en el capítulo 3 se aplicó a un robot manipulador de dos grados de libertad. Las pruebas experimentales y las simulaciones, se obtuvieron utilizando los parámetros reales del robot, la **¡Error! No se encuentra el origen de la referencia.** muestra dichos datos (ver apéndice D).

Tabla 4-1 Parámetros nominales utilizados del robot de 2 gdl

Parámetros	Valor
l_1 (longitud eslabón 1)	0.193 m
l_2 (longitud eslabón 2)	0.145 m
m_1 (masa eslabón 1)	0.763 kg
m_2 (longitud eslabón 2)	0.343 kg
l_{c1} (longitud del centro de masa eslabón 1)	0.136 m
l_{c2} (longitud del centro de masa eslabón 2)	0.048 m
I_1 (Inercia del eslabón 1)	0.0186
I_2 (Inercia del eslabón 2)	0.0026987
Constante g (gravedad)	9.81

La trayectoria deseada fue la siguiente:

$$\begin{aligned}
q_{d_1} &= -\pi \cdot \text{sen}(0.1 \cdot t) \\
q_{d_2} &= \pi \cdot \text{sen}(0.1 \cdot t)
\end{aligned} \tag{3.2}$$

De acuerdo a las ecuaciones obtenidas del modelo del robot de dos grados de libertad presentada en la sección 3.1 y la ecuación (2.22); una parametrización del robot es la siguiente:

$$\begin{aligned}
\theta_1 &= m_1 l_{c_1}^2 + m_2 l_1^2 + I_1 & \theta_2 &= m_2 l_{c_2}^2 + I_2 \\
\theta_3 &= m_2 l_1 l_{c_2} & \theta_4 &= m_1 l_{c_1} \\
\theta_5 &= m_2 l_1 & \theta_6 &= m_2 l_{c_2}
\end{aligned} \tag{3.3}$$

con esta parametrización, la ecuación (2.22) que representa el modelo, puede ser reescrita como:

$$Y(q, \dot{q}, \ddot{q})\theta = \tau \quad Y \in \mathbb{R}^{2 \times 6}, \theta \in \mathbb{R}^6 \tag{3.4}$$

Donde los componentes y_{ij} de $Y(q, \dot{q}, \ddot{q})$, estarían dados por :

$$\begin{aligned}
y_{11} &= \ddot{q}_1 & y_{12} &= \ddot{q}_1 + \ddot{q}_2 \\
y_{13} &= \cos(q_2)(2\ddot{q}_1 + \ddot{q}_2) - \text{sen}(q_2)(\dot{q}_2^2 + 2\dot{q}_1\dot{q}_2) \\
y_{14} &= g \text{sen}(q_1) & y_{15} &= g \text{sen}(q_1) \\
y_{16} &= g \text{sen}(q_1 + q_2) & y_{21} &= 0 \\
y_{22} &= \ddot{q}_1 + \ddot{q}_2 & y_{23} &= \cos(q_2)\ddot{q}_1 + \text{sen}(q_2)\dot{q}_1^2 \\
y_{24} &= 0 & y_{25} &= 0 \\
y_{26} &= g \text{sen}(q_1 + q_2)
\end{aligned} \tag{3.5}$$

Mientras que los componentes de la matriz $Y(q, \dot{q}, a, v)$ estarían dados por:

$$\begin{aligned}
y_{11} &= a_1 & y_{12} &= a_1 + a_2 \\
y_{13} &= \cos(q_2)(2a_1 + a_2) - \text{sen}(q_2)(\dot{q}_2 v_2 + \dot{q}_1 v_2 + \dot{q}_2 v_1) \\
y_{14} &= -g \text{sen}(q_1) & y_{15} &= g \text{sen}(q_1) \\
y_{16} &= g \text{sen}(q_1 + q_2) & y_{21} &= 0 \\
y_{22} &= a_1 + a_2 & y_{23} &= \cos(q_2)a_1 + \text{sen}(q_2)\dot{q}_1 v_1 \\
y_{24} &= 0 & y_{25} &= 0 \\
y_{26} &= g \text{sen}(q_1 + q_2)
\end{aligned} \tag{3.6}$$

Los valores del vector de parámetros θ_0 de acuerdo a la ecuación (4.3) corresponden a:

Tabla 4-2 Valor parámetros nominales

$\theta_{0,i}$	$\theta_{0,1}$	$\theta_{0,2}$	$\theta_{0,3}$	$\theta_{0,4}$	$\theta_{0,5}$	$\theta_{0,6}$
Valor	0.0455	0.0035	0.0032	0.1038	0.0662	0.0165

La estructura del controlador difuso y sus FM de entrada y salida se muestran en las figuras 3.9, 3.10, 3.11 y 3.12. De acuerdo al teorema 2 y las ecuaciones (3.44) y (3.45), se seleccionaron los centros de las FM como parámetros de diseño, las Tabla 4-3 y Tabla 4-4 resumen dichos parámetros en el espacio normalizado $[-1,1]$.

Tabla 4-3 Centros de las FM de entrada

Parámetro <i>Eslabón i</i>	$C_{ENG,i}$	$C_{ENP,i}$	$C_{ECR,i}$	$C_{EPP,i}$	$C_{EPG,i}$
1	-1.0	-0.01	0.0	0.01	1
2	-1.0	-0.001	0.0	0.001	1

Tabla 4-4 Centros de las FM de salida

Parámetro <i>Eslabón i</i>	$C_{NGK,i}$	$C_{NPK,i}$	$C_{CRK,i}$	$C_{PPK,i}$	$C_{PGK,i}$
1	-1.0	-0.1	0.0	0.1	1
2	-1.0	-0.15	0.0	0.15	1

El esquema completo del controlador lógico difuso se muestra en la Figura 4.1; las ganancias de escalamiento seleccionadas se muestran en la tabla 4-5.

Tabla 4-5 Ganancias del CLD

Ganancias <i>Eslabón i</i>	Ge_i	Ge'_i	Gk_i
1	4.0	0.32	3.5
2	0.19	0.02	2.1

En consecuencia, la pendiente λ_i de r_i y la magnitud del límite φ_i , los cuales determinan las características del error y precisión de seguimiento en estado estable, respectivamente, de acuerdo al teorema 2 y la ecuación (2.37), están dadas como:

$$\lambda_1 = \frac{G_{e,1}}{G_{\dot{e},1}} = 12.5, \quad \lambda_2 = \frac{G_{e,2}}{G_{\dot{e},2}} = 9.5, \quad (3.7)$$

$$\phi_1 = \frac{2C_{PPE,1}}{G_{\dot{e},1}} = 0.0625, \quad \phi_2 = \frac{2C_{PPE,2}}{G_{\dot{e},2}} = 0.1$$

La matriz \mathbf{K} debe cumplir que sea positiva definida por lo tanto se eligió de tal forma que el controlador tuviera el mejor desempeño posible.

$$\mathbf{K} = \begin{bmatrix} 0.3 & 0 \\ 0 & 0.1 \end{bmatrix} \quad (3.8)$$

A continuación se presentan los resultados simulados y experimentales.

4.2.2 Simulaciones

Siguiendo el esquema presentado en la figura 4.1, se realizó la simulación del controlador en Matlab®, con los programas que se muestran en el apéndice C. Las simulaciones se realizaron con un periodo de muestreo de 0.01 s., durante 120 s., utilizando los valores de ganancias y de diseño del CLD presentados en el apartado anterior.

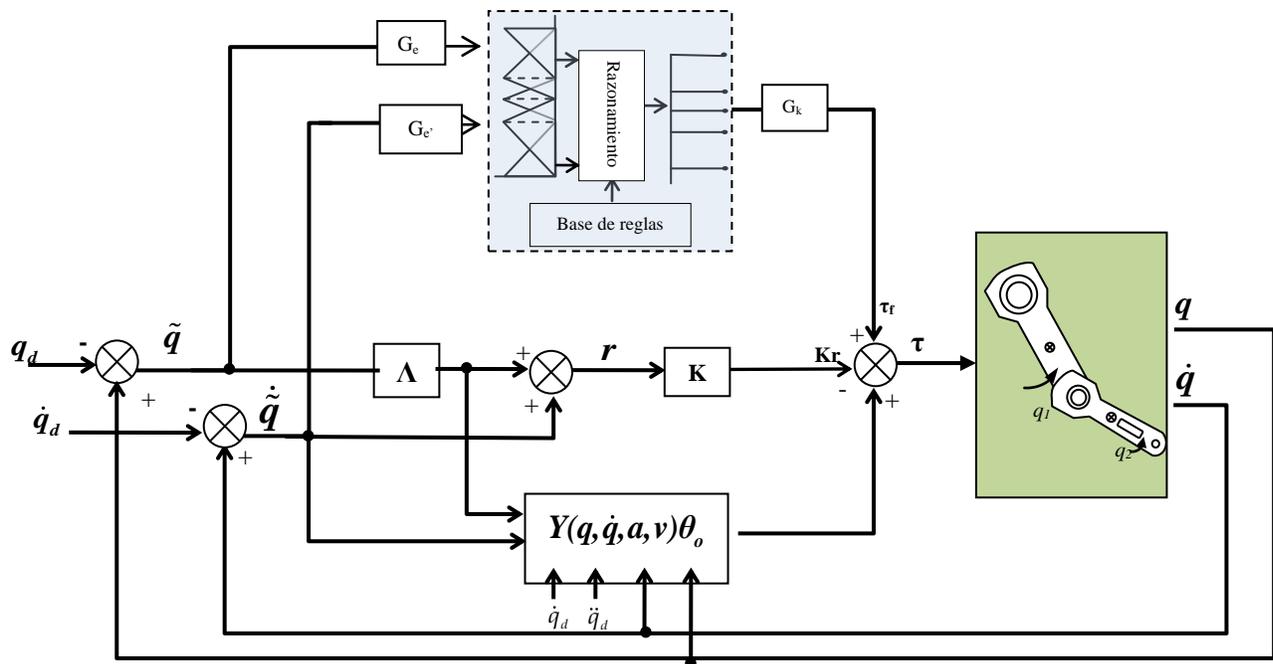


Figura 4.1 Diagrama a bloques del controlador robusto difuso

Prueba 1. Robot sin carga: La primera prueba simulada consistió en suponer al robot sin carga en el eslabón 2. Las gráficas obtenidas de dicha prueba son las siguientes:

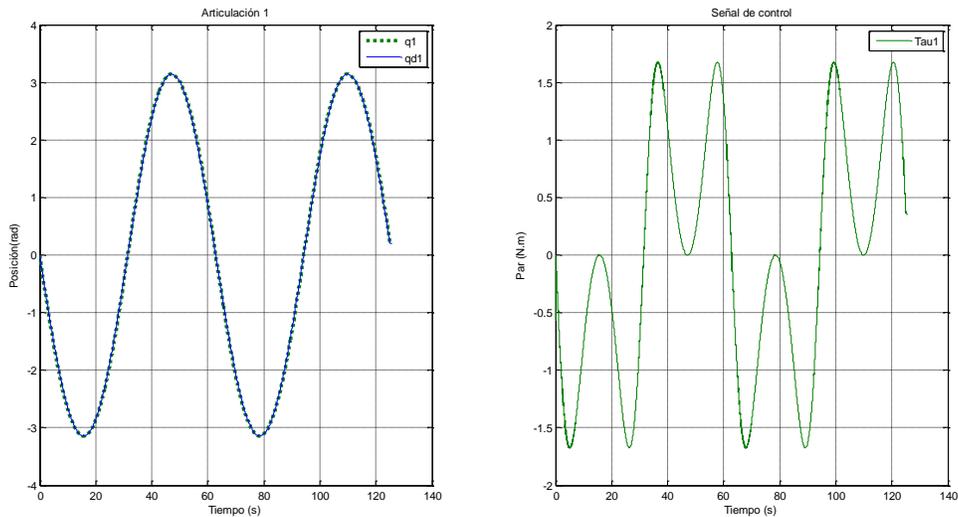


Figura 4.2 Simulación del CLD robusto para la articulación 1

Por lo que se observa en la Figura 4.2, el eslabón 1 sigue la trayectoria deseada con un error mínimo. Utilizando el valor eficaz del error se obtuvo que para el eslabón 1 esté fue de $\tilde{q}_{1,rms} = 27.2 \times 10^{-3}$

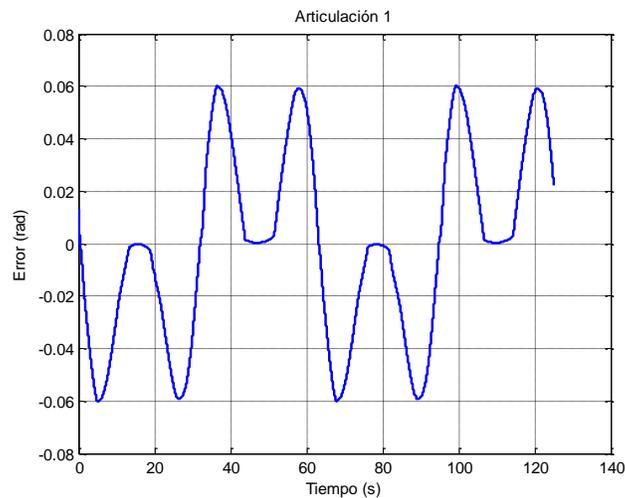


Figura 4.3 Gráfica del error de posición, eslabón 1

El par máximo fue de alrededor de ± 1.66 N.m, el cual está representado por la figura derecha de Figura 4.2. En la figura 4.3 se muestra la grafica del error de seguimiento de la articulación 1. Para el eslabón 2, se obtuvieron los resultados mostrados en la figura 4.4.

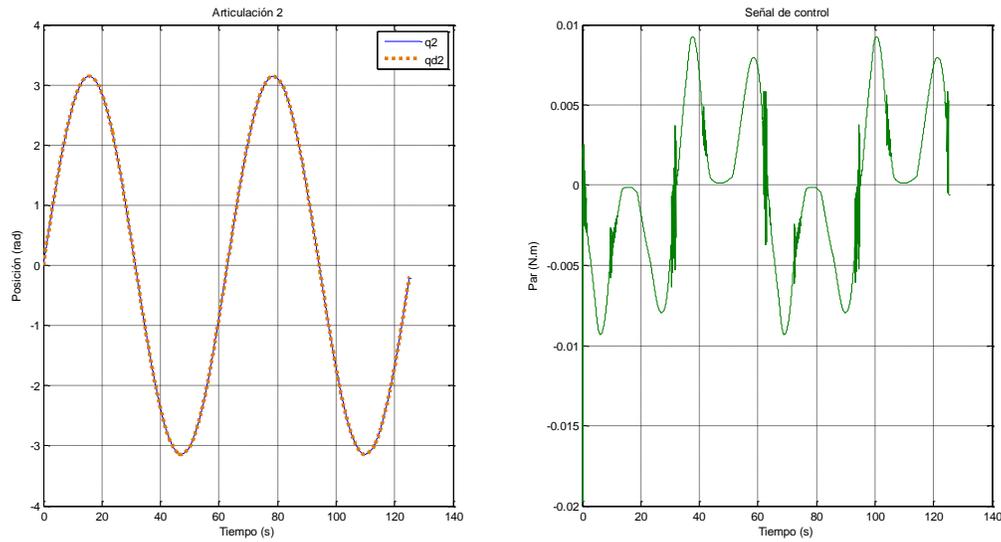


Figura 4.4 Simulación CLD robusto eslabón 2

El error de seguimiento eficaz resultante es de $\tilde{q}_{2,rms} = 3.7 \times 10^{-3}$ rad y la gráfica correspondiente al error de seguimiento en el tiempo de 126 s se presenta en la figura 4.5.

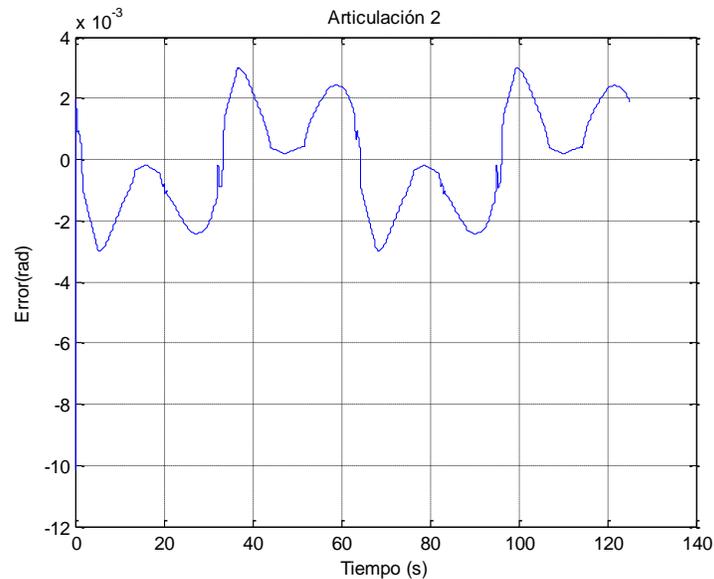


Figura 4.5 Error de seguimiento eslabón 2

Prueba 2. Robot con carga: La finalidad es demostrar la característica de robustez de los CLD, los resultados obtenidos son los siguientes:

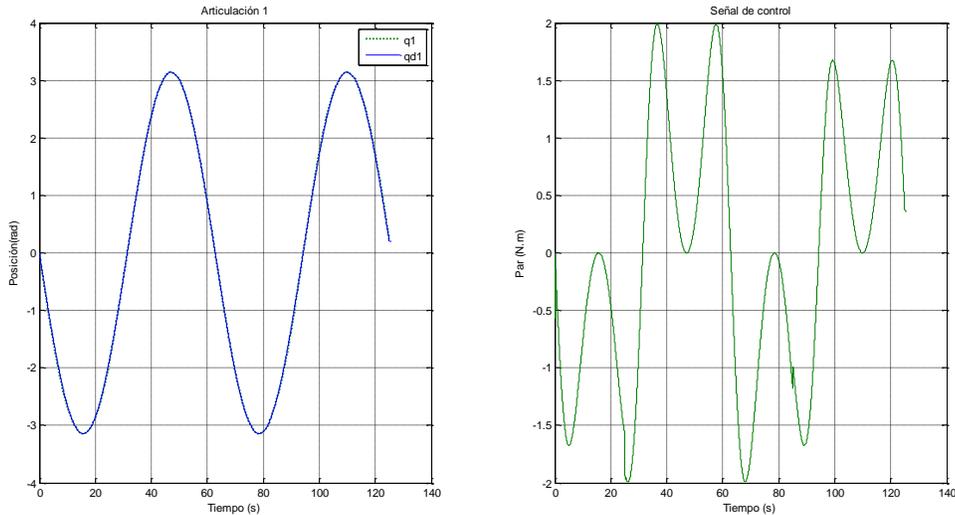


Figura 4.6 Simulaciones variando la masa en el eslabón 2

En la figura 4.6 se observa a la articulación 1 y su correspondiente señal de control, donde se aprecia que en el tiempo $t=25$ el par sufre un aumento en comparación con la señal obtenida en la figura 4.2. El error eficaz es de $\tilde{q}_{1,rms} = 20 \times 10^{-3}$ rad, que como era de esperarse es mayor al obtenido en la prueba 1. La figura 4.7 muestra la gráfica del error de seguimiento, donde es claro que el error disminuye a un valor cercano a cero al momento de cambiar la masa, inercia y centro de masa del eslabón 2, demostrando la robustez del sistema en lazo cerrado.

Para la articulación 2 se obtuvo un error eficaz de $\tilde{q}_{2,rms} = 3.7 \times 10^{-3}$ rad, las gráficas correspondientes a la posición y la señal de control se muestra en la Figura 4.8. La variación en la masa se aprecia en la señal de control donde se observa que esta disminuye dado que al agregarle la masa, el nuevo vector de parámetros θ calculados cambia y se aproxima al nominal θ_0 .

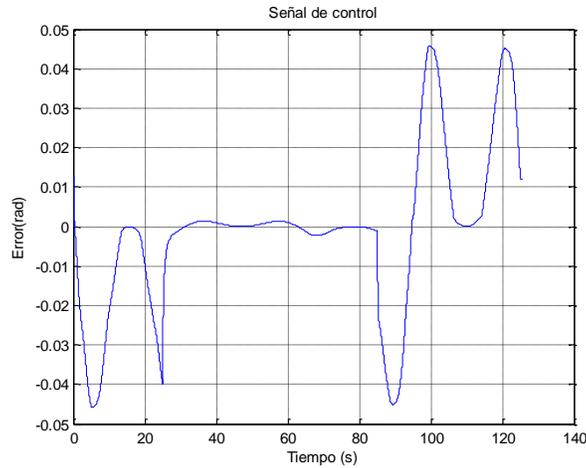


Figura 4.7 Error de seguimiento articulación 1

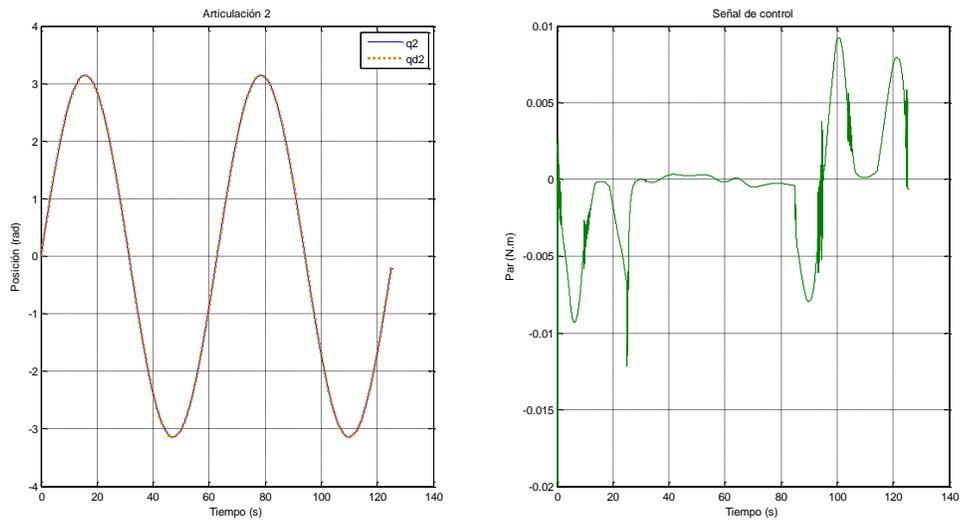


Figura 4.8 Simulación articulación 2 con cambio de masa

La figura 4.9 muestra el error de seguimiento obtenido, donde se observa mejor el cambio de masa aplicada al manipulador en el intervalo de 25 a 85 s.

Prueba 3. Método de defuzificación por promedio de áreas y funciones triangulares a la salida. La base de las funciones de membresía se eligió de 0.25, utilizando los centros presentados en la tabla 4-4. Los resultados obtenidos se muestran en las figuras 4.10 y 4.11 para las articulaciones 1 y 2, respectivamente.

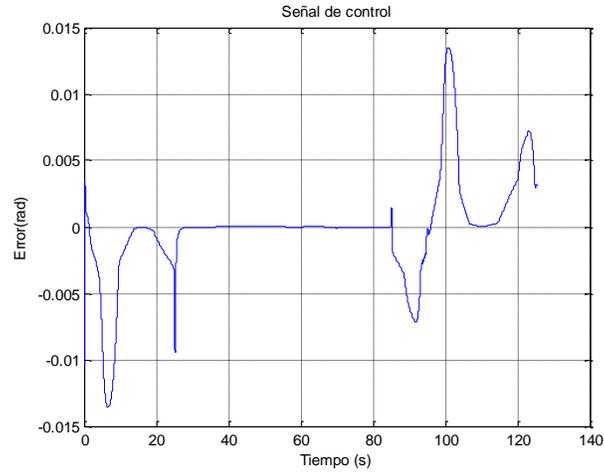


Figura 4.9 Error de seguimiento articulación 2

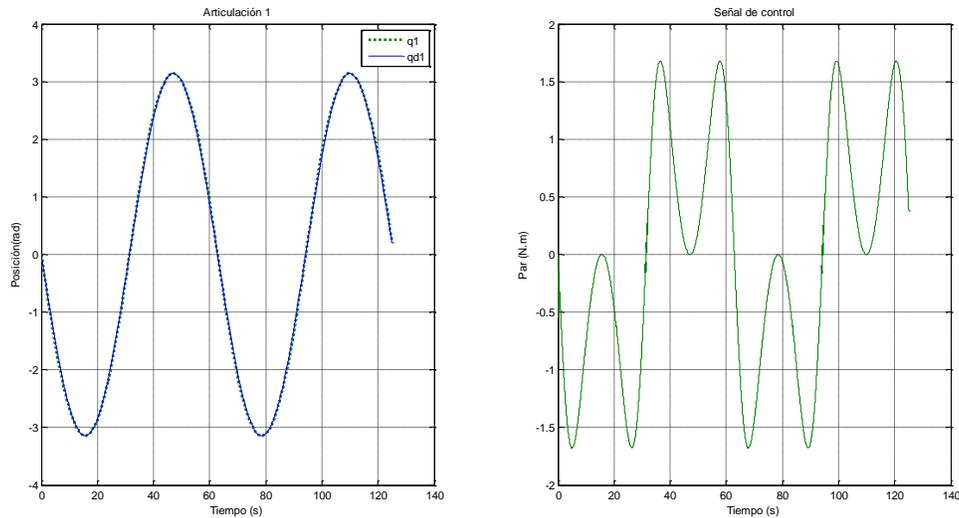


Figura 4.10 Respuesta del eslabón 1

El error cuadrático para la articulación 1 corresponde a $\tilde{q}_{1,rms} = 32.2 \times 10^{-3}$ rad, el cual es mayor al obtenido por el método de defuzificación por promedio de centros. Con la articulación 2 se obtuvo un error eficaz $\tilde{q}_{2,rms} = 14.5 \times 10^{-3}$ rad.

El método del promedio de centros presenta un mejor desempeño en cuanto al índice de error obtenido (en esta aplicación), además realiza menos operaciones

computacionales a diferencia de método del promedio de áreas. El optar por uno u otro método depende del diseñador y el proceso a controlar.

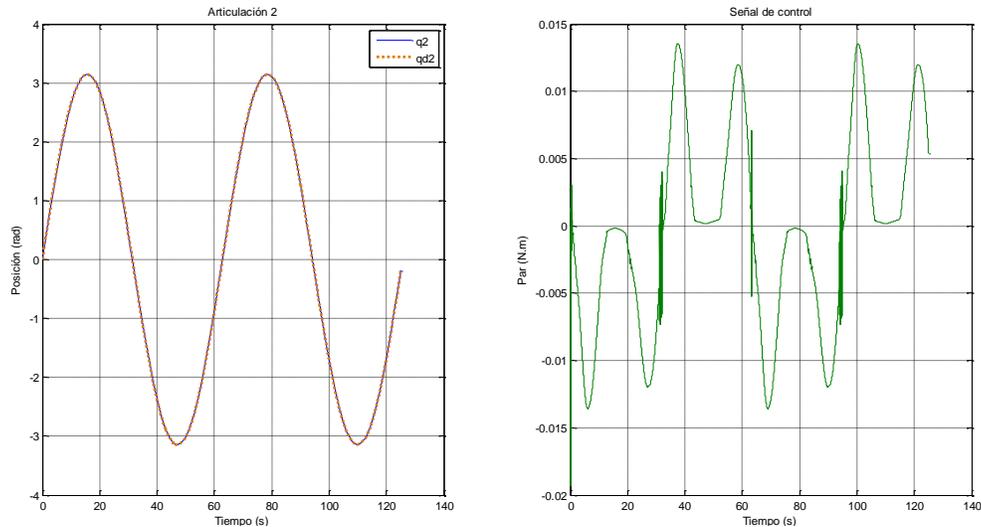


Figura 4.11 Respuesta del eslabón 2

4.2.3 Resultados experimentales

Para las pruebas experimentales se utilizaron las mismas ganancias seleccionadas en el epígrafe 4.2.1. Se utilizó el vector de parámetros θ , que corresponde a los valores calculados mostrados en la tabla 4-2. El periodo de muestreo es de 0.01s y las pruebas se realizaron en 125 s. La descripción de la plataforma experimental se muestra en el apéndice D.

Prueba 1. Robot sin carga. Para la articulación 1 se obtuvo la respuesta presentada en la figura 4.10, donde la figura izquierda representa la trayectoria actual y la trayectoria deseada. La articulación 1 sigue la trayectoria deseada con un error eficaz de 16.7×10^{-3} rad. La gráfica del lado derecho de la figura 4.12 representa el par proporcionado por el algoritmo de control.

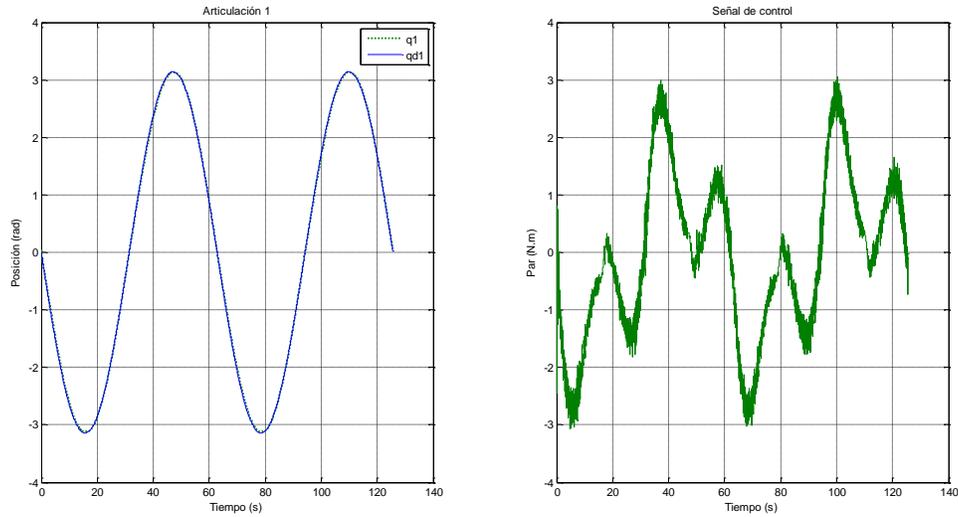


Figura 4.12 Resultados experimentales de la articulación 1

En la figura 4.13 se muestra la gráfica del error de seguimiento, el valor absoluto máximo del error es de 0.06 rad aproximadamente.

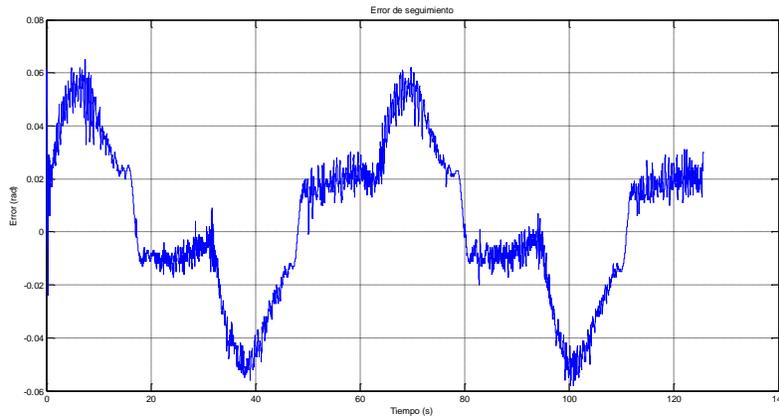


Figura 4.13 Error de seguimiento articulación 1

La figura 4.14 muestra la posición actual y la posición deseada, además de la señal de control correspondiente a la articulación 2. El eslabón 2 sigue la trayectoria deseada con un error eficaz de 13.0×10^{-3} rad

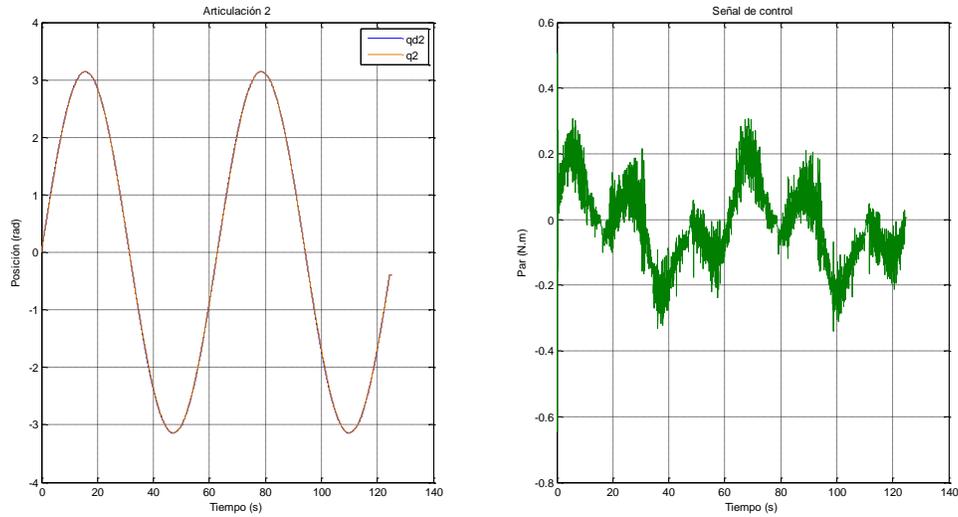


Figura 4.14 Resultados experimentales articulación 2

Prueba 2. Robot con carga. Por seguridad se optó por cambiar la masa del eslabón 2 en el programa de control del robot y no agregarle la carga de manera física para evitar que los engranes de los motores se dañaran.

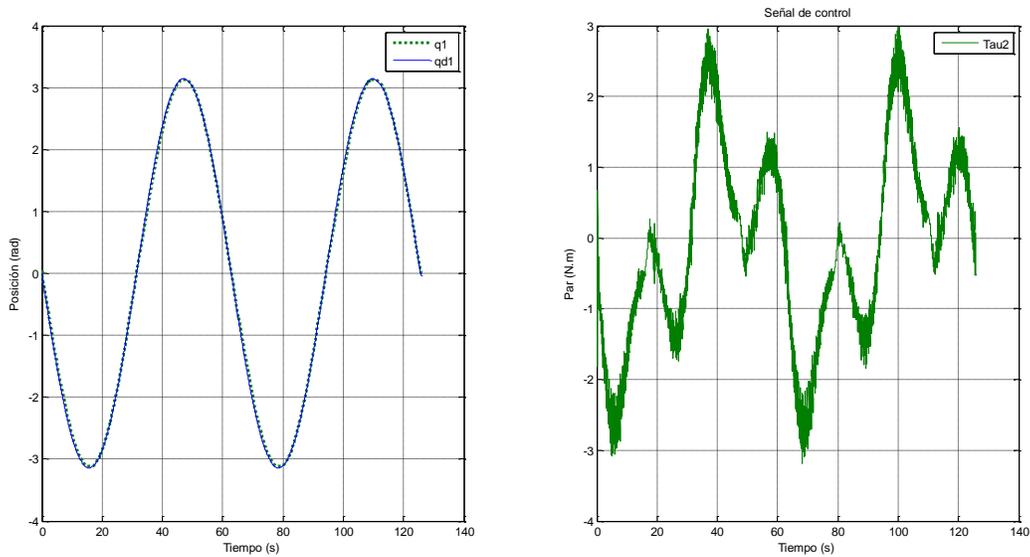


Figura 4.15 Resultados experimentales con carga

La figura 4.15 muestra la trayectoria de la articulación 1, el error obtenido al cambiar la masa fue de $\tilde{q}_{1,rms} = 29.7 \times 10^{-3}$, de aquí se observa que aumento su valor en comparación con el obtenido en la prueba sin carga.

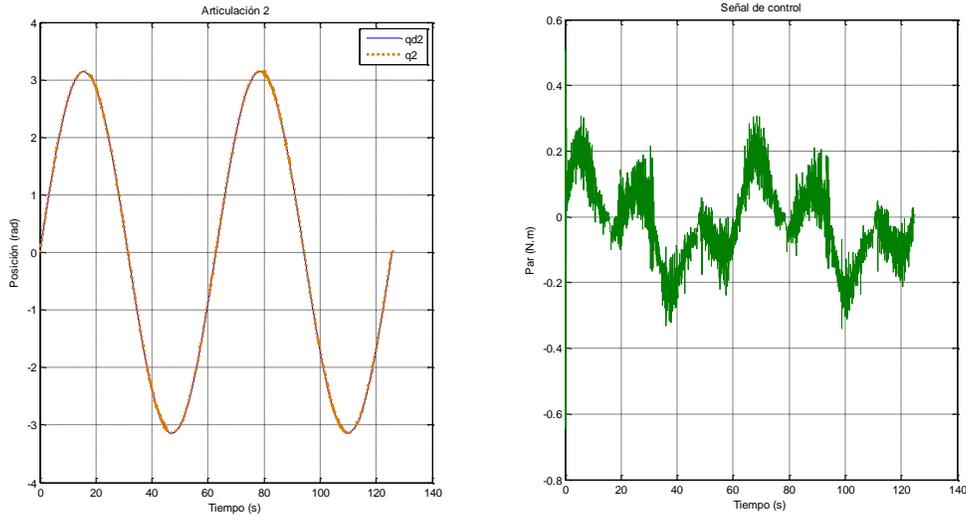


Figura 4.16 Resultados experimentales articulación 2

La figura 4.16 presenta la trayectoria de la articulación 2, el error obtenido fue de $\tilde{q}_{2,rms} = 14.1 \times 10^{-3}$ rad, incrementándose en comparación con el obtenido anteriormente.

Prueba 3. Método de defuzificación por promedio de áreas y funciones triangulares a la salida. Utilizando las ganancias de la tabla 4-5 y un ancho de base en las funciones triangulares de 0.25, se obtuvieron los resultados presentados en las figuras 4.17 y 4.18.

El error eficaz resultante fue de $\tilde{q}_{1,rms} = 38.4 \times 10^{-3}$ rad, en esta prueba en comparación con la prueba 1, el error fue más grande, sin embargo, se tuvo un mejor desempeño en el seguimiento de la trayectoria ya que era de forma suave y continúa. El error eficaz en la articulación 2 fue de $\tilde{q}_{2,rms} = 20.6 \times 10^{-3}$ rad. Con lo anterior y comparando con los resultados derivados de las simulaciones se concluye que se tiene un menor índice de error con el método de defuzificación por media de centros.

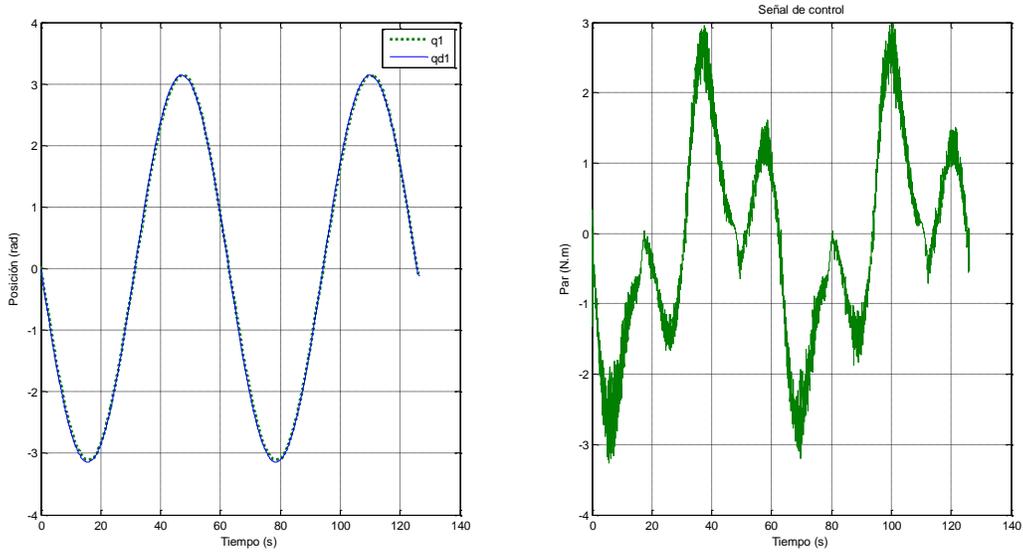


Figura 4.17 Resultados experimentales articulación 1

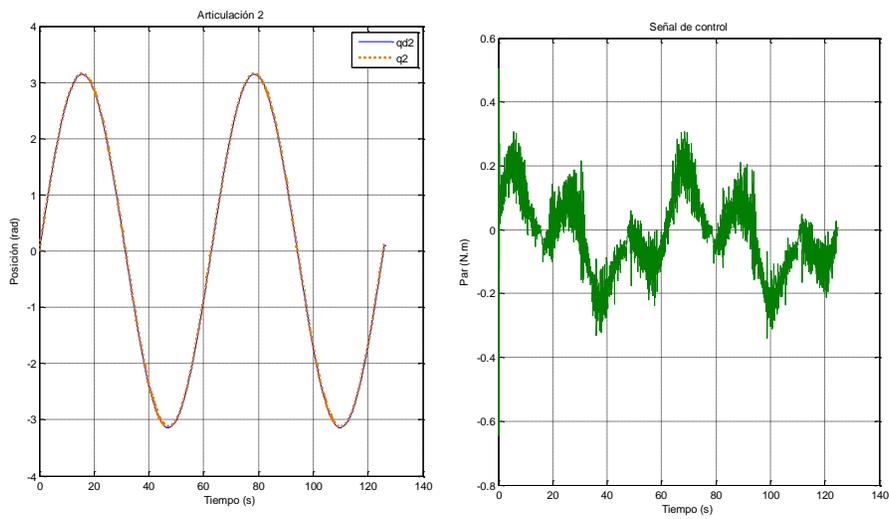


Figura 4.18 Resultados experimentales articulación 2 con el CLD

4.3 Control PD con compensación

El controlador PD con compensación puede expresarse como:

$$\tau = K_p \tilde{q} + K_v \dot{\tilde{q}} + M(q)[\ddot{q}_d + \Lambda \tilde{q}] + C(q, \dot{q})[\dot{q}_d + \Lambda \tilde{q}] + g(q) \quad (3.9)$$

donde $K_p, K_v \in \mathbb{R}^{n \times n}$ son matrices definidas positivas de diseño, $\tilde{q} = q_d - q$ denota el error de posición y $\Lambda = K_p / K_v$.

El control PD con compensación es del tipo basado en el modelo, es decir, que la ley de control emplea explícitamente los términos del modelo. Kelly y Santibáñez (2003), muestran el diseño de este controlador, dado que no es objetivo de esta investigación, no se presenta el desarrollo, simplemente se tomo como referencia los parámetros de diseño obtenidos por los autores mencionados.

4.3.1 Simulaciones

Las simulaciones se realizaron en Matlab®, utilizando el programa mostrado en el apéndice C; se tomaron los parámetros del robot de la tabla 4-1. El tiempo de muestreo es de 0.01 s, las ganancias elegidas son las siguientes:

$$\begin{aligned}
 k_{p_1} &= 3.6, & k_{v_1} &= 0.3 \\
 k_{p_2} &= 0.6, & k_{v_2} &= 0.03 \\
 \lambda_1 &= \frac{k_{p_1}}{k_{v_1}} = 12 \\
 \lambda_2 &= \frac{k_{p_2}}{k_{v_2}} = 20
 \end{aligned} \tag{3.10}$$

dado que:

$$M(q)[\ddot{q}_d + \Lambda \dot{\tilde{q}}] + C(q, \dot{q})[\dot{q}_d + \Lambda \tilde{q}] + g(q) = Y(q, \dot{q}, a, v)$$

donde

$$\begin{aligned}
 a &= \ddot{q}_d + \Lambda \dot{\tilde{q}} = \dot{v} \\
 v &= \dot{q}_d + \Lambda \tilde{q}
 \end{aligned} \tag{3.11}$$

La ley de control expresada en la ecuación 4.9 puede ser implementada, empleando la ecuación 4.12.

$$\tau = K_p \tilde{q} + K_v \dot{\tilde{q}} + Y(q, \dot{q}, a, v)\theta \quad (3.12)$$

Prueba 1. Robot sin carga. La figura 4.19 muestra la respuesta de la articulación 1 con la trayectoria deseada presentada en la ecuación 4.1. El error eficaz es de $\tilde{q}_1 = 7.03 \times 10^{-4}$ rad. Dado que se considera al modelo del robot conocido o exacto el error es muy pequeño.

Las gráficas resultantes de la articulación 2 son las presentadas por la figura 4.20. El error de seguimiento eficaz es de $\tilde{q}_2 = 3.67 \times 10^{-4}$ rad.

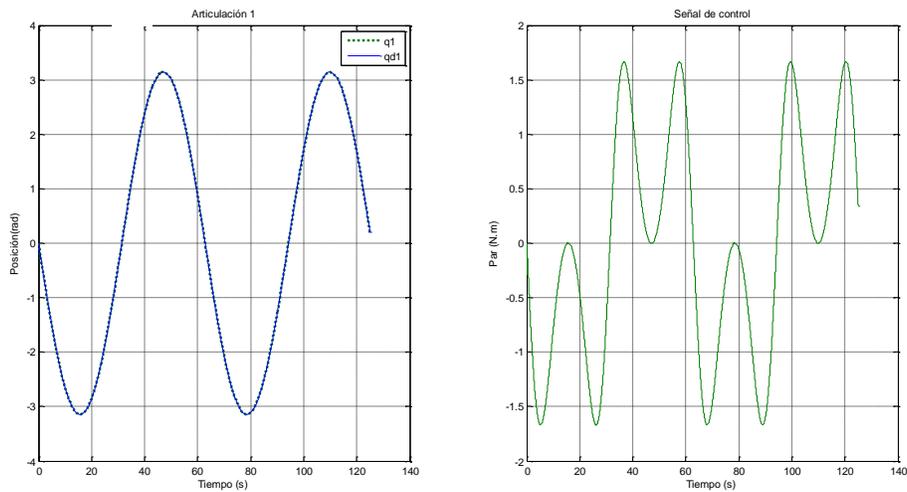


Figura 4.19 Simulación PD compensado articulación 1

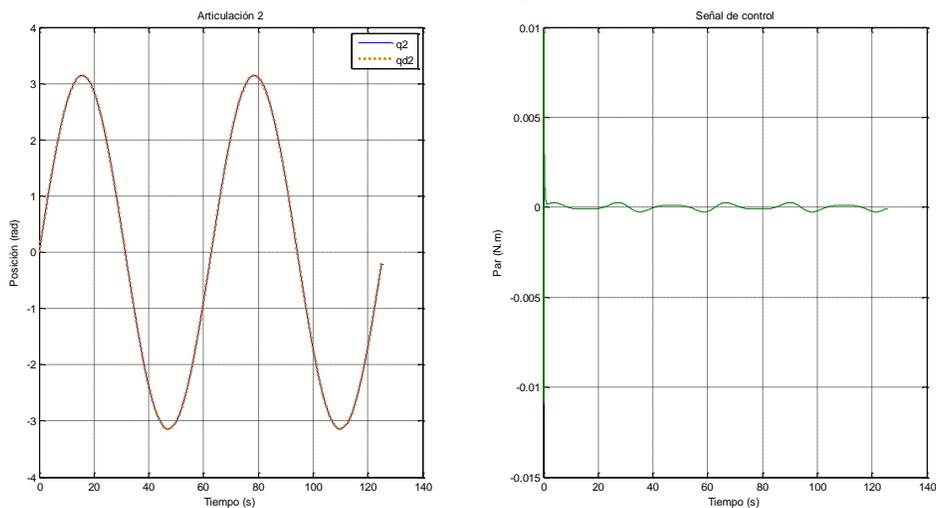


Figura 4.20 Simulación PD compensado articulación 2

Prueba 2. Robot con carga. Los resultados obtenidos para el PD compensado en esta prueba se muestran a continuación.

El error eficaz resultante para la articulación 1 fue de $\tilde{q}_{1,rms} = 36.94 \times 10^{-3}$ rad, mucho mayor en comparación al obtenido considerando al robot sin carga. La señal de control también aumenta en el intervalo de tiempo en el que es agregada la carga, como lo muestra la figura 4.21.

Para la articulación 2 se obtuvo un error eficaz de $\tilde{q}_{2,rms} = 9.4 \times 10^{-3}$ rad, las gráficas correspondientes a la posición y la señal de control se muestra en la figura 4.22. La señal de control aumenta de manera considerable, siendo el par máximo de 0.024 N.m; pasados los 85 seg., el par disminuye a casi cero

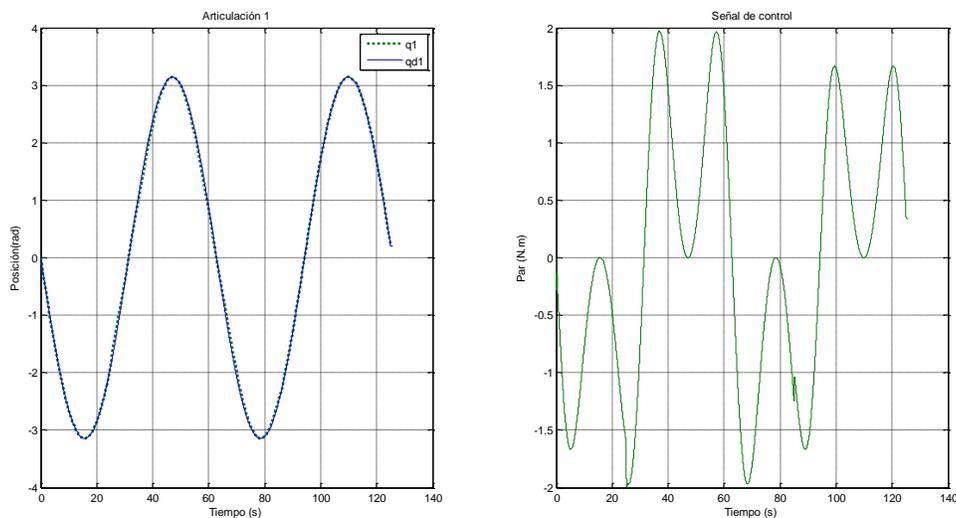


Figura 4.21 Simulación PD compensado con variación de la masa 2

4.3.2 Resultados experimentales

Se utilizó la misma plataforma experimental utilizada para el CLD robusto, el periodo de muestreo fue de 0.01s en un tiempo de 63 s. Las trayectorias deseadas utilizadas se presentan en la ecuación 4.1.

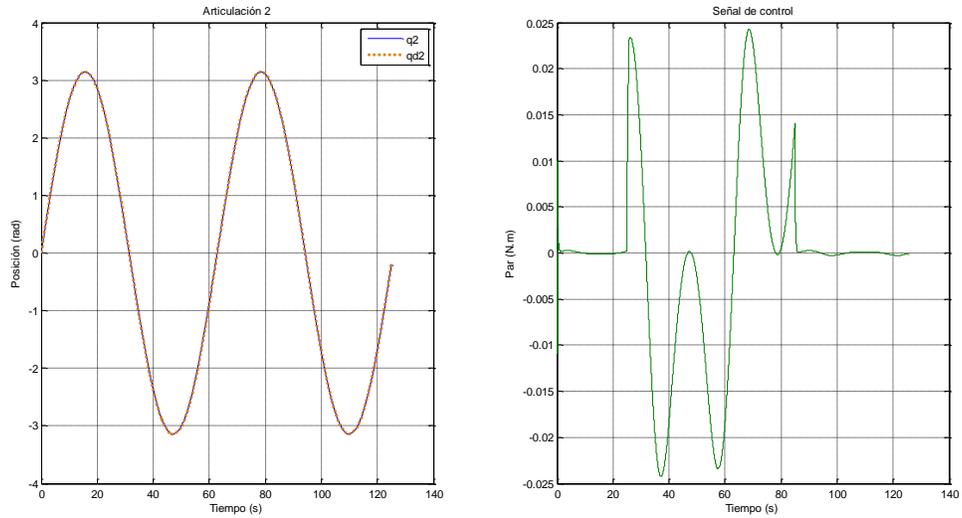


Figura 4.22 Simulación PD Compensado articulación 2

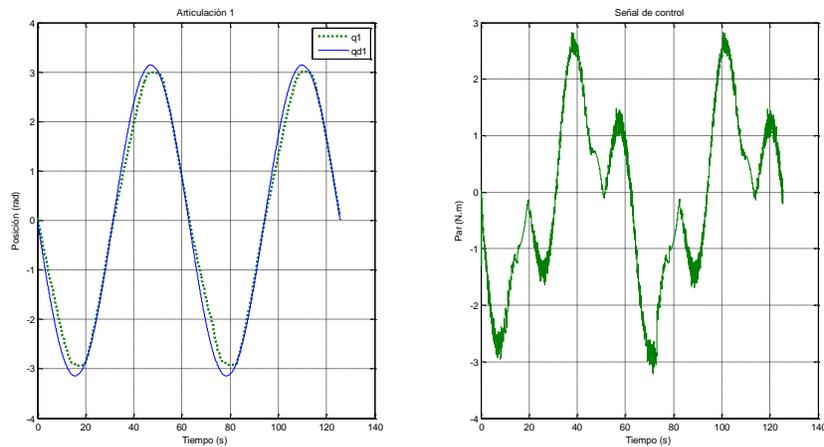


Figura 4.23 Resultados experimentales PD compensado articulación 1

Prueba 1. Robot sin carga. Para la articulación 1 se obtuvo la respuesta presentada en la figura 4.23, el error de seguimiento eficaz es de 0.2427 rad. La gráfica del lado derecho de la figura 4.22 representa el par proporcionado por el algoritmo de control, el cual tiene un par máximo aproximado de 2.5 N.m a los 38 s.

La figura 4.24 muestra la posición actual y la posición deseada, además de la señal de control correspondiente a la articulación 2. El eslabón 2 sigue la trayectoria deseada con un error eficaz de 0.1449 rad.

Prueba 2. Robot con carga. El error obtenido para la articulación 1 fue de 0.2579 rad. Los resultados obtenidos se muestran en la figura 4.25.

Para la articulación 2 se obtuvo un error de 0.1590 rad y los resultados se presentan en la figura 4.26.

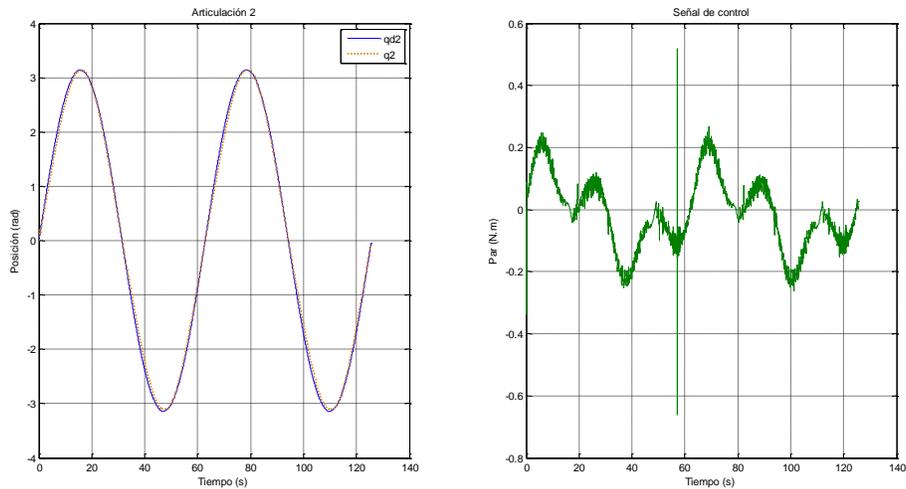


Figura 4.24 Resultados experimentales PD compensado articulación 2

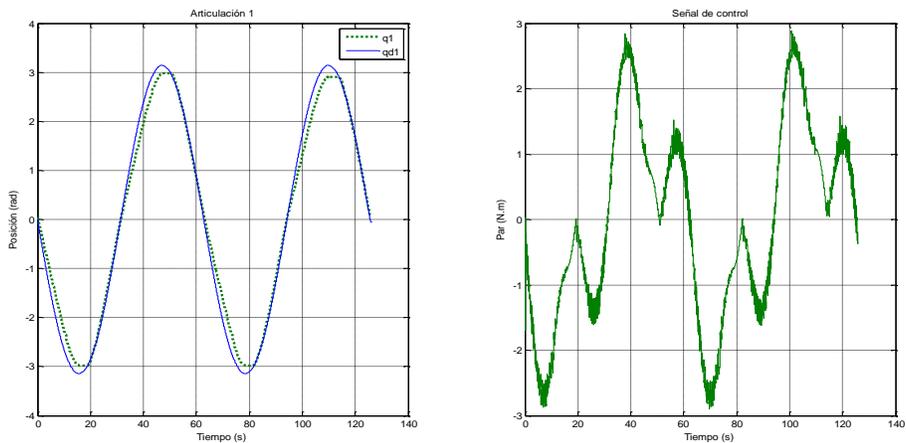


Figura 4.25 Resultados experimentales PD compensado variación de masa eslabón 1

4.4 Control PID

El controlador PID para robots manipuladores puede expresarse como:

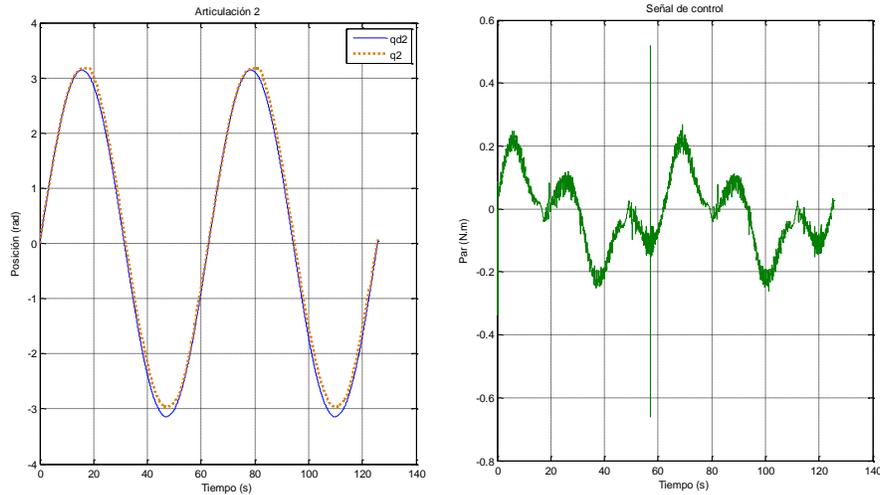


Figura 4.26 PD compensado variando la masa

$$\tau = K_p \tilde{q} + K_I \int_0^t q(s) ds + K_D \dot{\tilde{q}} \quad (3.13)$$

donde K_p, K_I y $K_d \in \mathbb{R}^{n \times n}$ son matrices definidas positivas de diseño, $\tilde{q} = q_d - q$ denota el error de posición.

La mayoría de los robots manipuladores son controlados mediante controladores PID. El objetivo de agregar un término integral al PD es intentar llevar a cero el error de posición. Kelly y Santibáñez (2003), muestran el diseño de este controlador, dado que no es objetivo de esta investigación, no se presenta el desarrollo, simplemente se tomo como referencia los parámetros de diseño obtenidos por los autores mencionados.

4.4.1 Simulaciones

Las simulaciones se realizaron en Matlab®, utilizando el programa mostrado en el apéndice C; se tomaron los parámetros del robot mostrados en la tabla 4-1. El tiempo de muestreo es de 0.01, las ganancias elegidas son las siguientes:

$$\begin{aligned}
 k_{p_1} &= 10, & k_{I_1} &= 9 & k_{d_1} &= 0.8 \\
 k_{p_2} &= 4, & k_{I_2} &= 2 & k_{d_2} &= 0.3
 \end{aligned}
 \tag{3.14}$$

Prueba 1. Robot sin carga. La figura 4.27 muestra la respuesta de la articulación 1 con la trayectoria deseada presentada en la ecuación 4.1. El error de seguimiento eficaz es de $\tilde{q}_1 = 0.0256\text{rad}$. El valor de par máximo es de aproximadamente 1.6 N.m. Las gráficas resultantes de la articulación 2 se presentan en la figura 4.28. El error de seguimiento eficaz es de $\tilde{q}_2 = 9.015 \times 10^{-4}\text{rad}$.

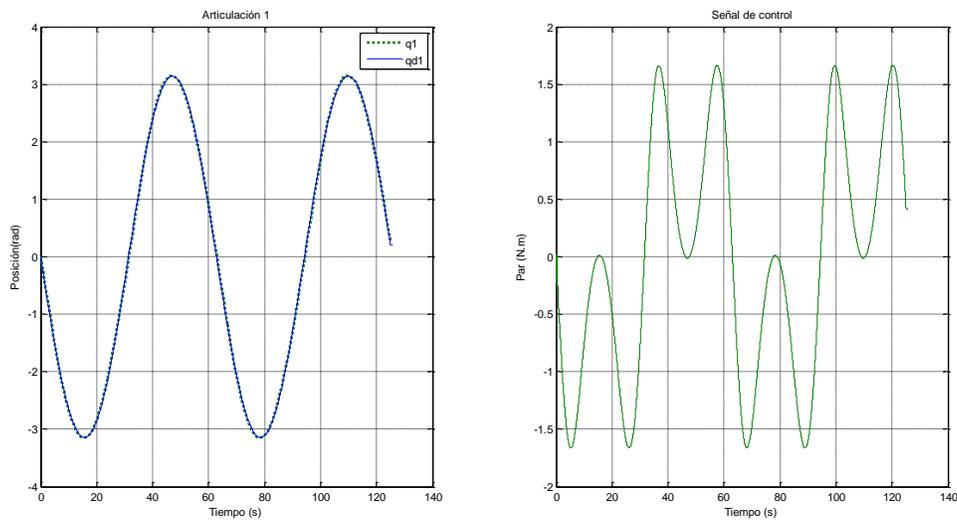


Figura 4.27 Simulación PID articulación 1

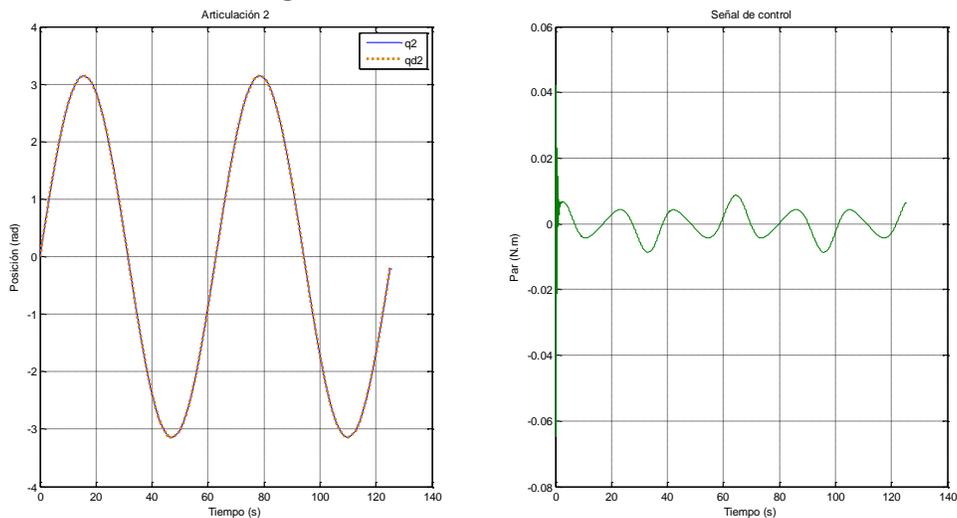


Figura 4.28 Simulación PID articulación 2

Prueba 2. Robot con carga. En la figura 4.28 se observa a la articulación 1 y su correspondiente señal de control, donde se aprecia que en el tiempo $t=25$, hay una perturbación por el cambio en la masa del eslabón 2. El error eficaz es de 0.0312 rad y el par máximo es de aproximadamente 2.4 N.m. Para la articulación 2 se obtuvo un error eficaz de 0.0033 rad, las gráficas correspondientes a la posición y la señal de control se muestra en la figura 4.29. La variación en la masa se aprecia en la señal de control, donde se observa un pico de aproximadamente 0.14 N.m.

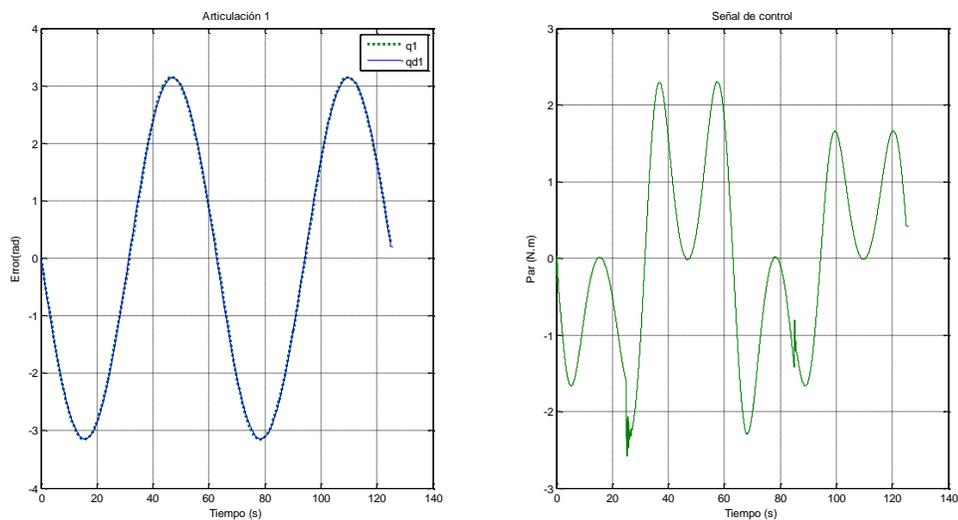


Figura 4.29 PID con variación en la masa de eslabón 2

4.4.2 Resultados experimentales

Se utilizó la misma plataforma experimental utilizada para el CLD robusto y el PD compensado, el periodo de muestreo fue de 0.01s en un tiempo de 125 s. Las trayectorias deseadas utilizadas se presentan en la ecuación 4.1.

Prueba 1. Robot sin carga. Para la articulación 1 se obtuvo la respuesta presentada en la figura 4.31, donde la figura izquierda representa la posición actual y la posición deseada.

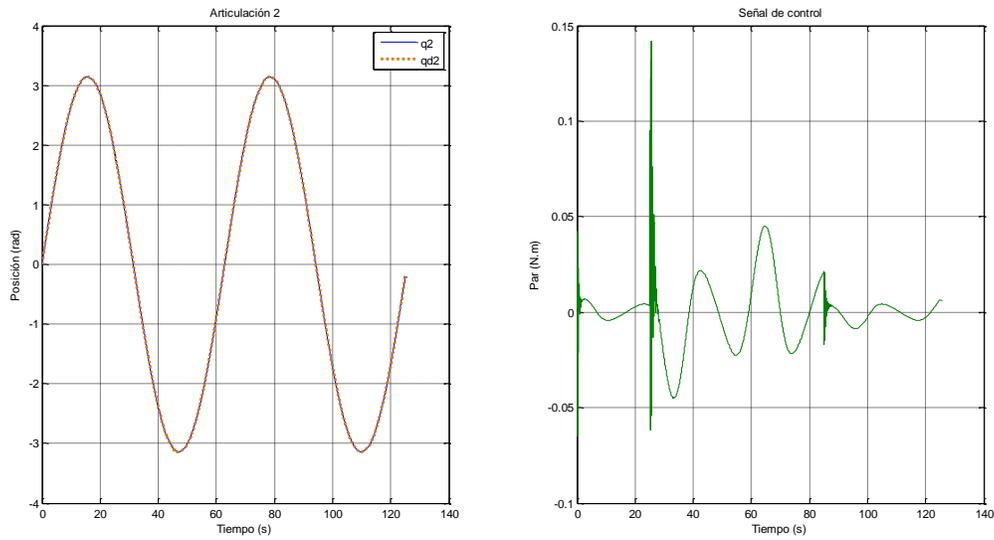


Figura 4.30 PID articulación 2

La articulación 1 sigue la trayectoria deseada con un error de seguimiento eficaz de 0.0328 rad. La gráfica del lado derecho de la figura 4.31 representa el par proporcionado por el algoritmo de control, el cual tiene un par máximo aproximado de 2.5 N.m. La figura 4.32 muestra la posición actual y la posición deseada, además de la señal de control correspondiente a la articulación 2. El eslabón 2 sigue la trayectoria deseada con un error eficaz de 0.0108 rad. La prueba 2 no fue posible realizarla mediante el programa de control ya que una ventaja del control PID es que no utiliza el modelo dinámico del robot manipulador.

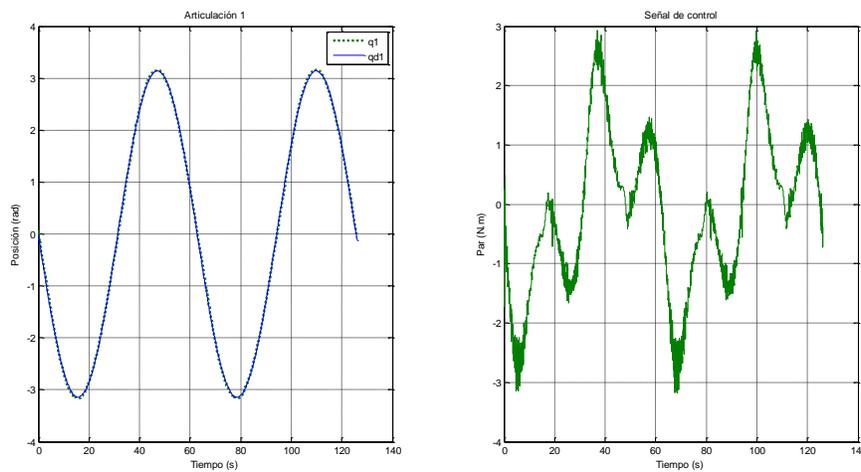


Figura 4.31 Resultados experimentales PID articulación 1

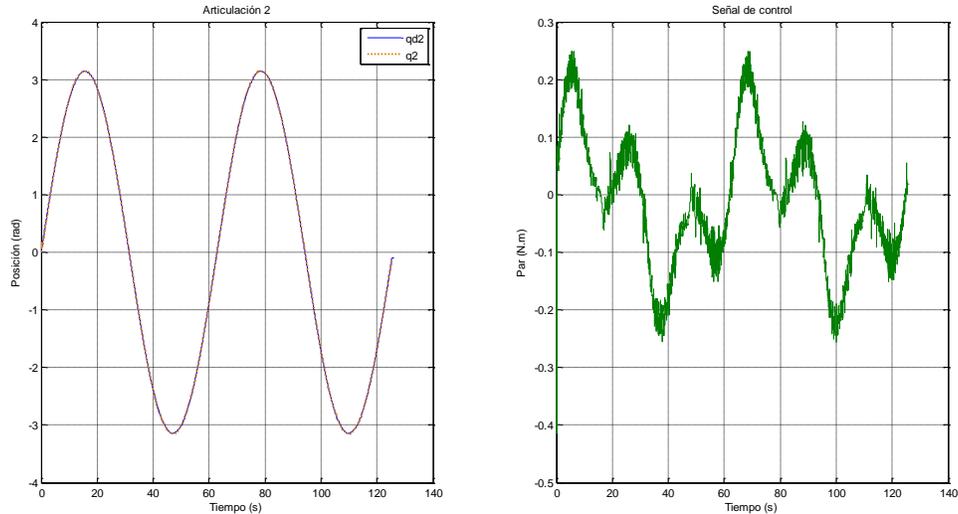


Figura 4.32 Resultados experimentales PID articulación 2

4.5 Comparación entre los algoritmos de control desarrollados.

A continuación se presenta una tabla comparativa de las pruebas experimentales realizadas de los tres algoritmos citados, de acuerdo al error eficaz o rms, el cual se calcula como se mostró en la sección 4.1. Los resultados experimentales se resumen en la siguiente tabla 4-6 del error eficaz, para la prueba 1.

Algoritmo de control	$\tilde{q}_{1,rms} (rad)$	$\tilde{q}_{2,rms} (rad)$
CLD robusto	16.7×10^{-3}	13.0×10^{-3}
PD compensado	242.7×10^{-3}	144.9×10^{-3}
PID	32.8×10^{-3}	10.6×10^{-3}

Tabla 4-6 Resumen del error eficaz resultados experimentales

En la gráfica siguiente se muestran los valores de la tabla de forma comparativa. En los resultados experimentales, se comprobó que el CLD robusto tiene un mejor seguimiento, ya que tiene el menor error eficaz, sin embargo, el PID presenta una buena respuesta. El PD compensado por otro lado, tuvo el peor seguimiento, dado que el modelo dinámico compensado, no corresponde con el modelo real del robot, esto debido a fricciones y los parámetros no modelados.

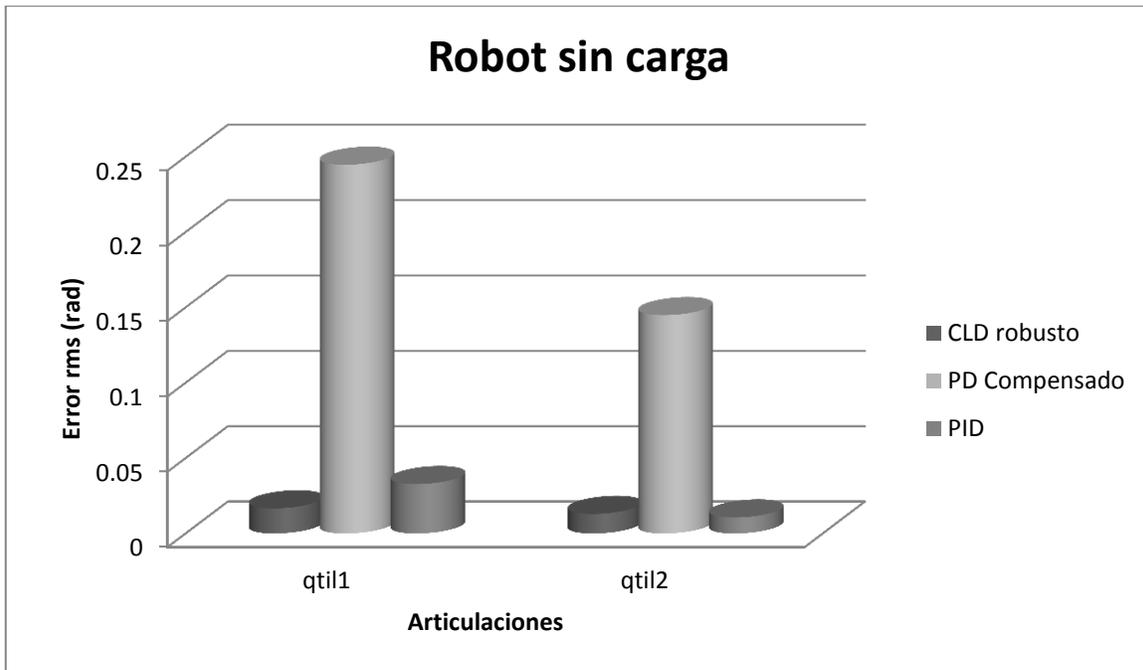


Figura 4. 33 Gráfico comparativo resultados experimentales

Para la prueba 2, únicamente se incluyen los resultados obtenidos del CLD y del PD compensado, esto debido a que no realizo la prueba experimental con el algoritmo PID.

Algoritmo de control	$\tilde{q}_{1,rms} (rad)$	$\tilde{q}_{2,rms} (rad)$
CLD robusto	38.3×10^{-3}	14.8×10^{-3}
PD compensado	257.9×10^{-3}	159.0×10^{-3}
PID	---	---

Tabla 4-7 Resumen del error eficaz resultados experimentales prueba 2

Estos resultados indican que el CLD está cumpliendo con el objetivo del control, que es reducir el error ante incertidumbres o cambios paramétricos. En especial muestra que el control PD compensado se puede complementar con un CLD difuso para mejorar su respuesta, sin embargo se complica la etapa de sintonización ya que aumenta el número de ganancias y parámetros a ajustar.

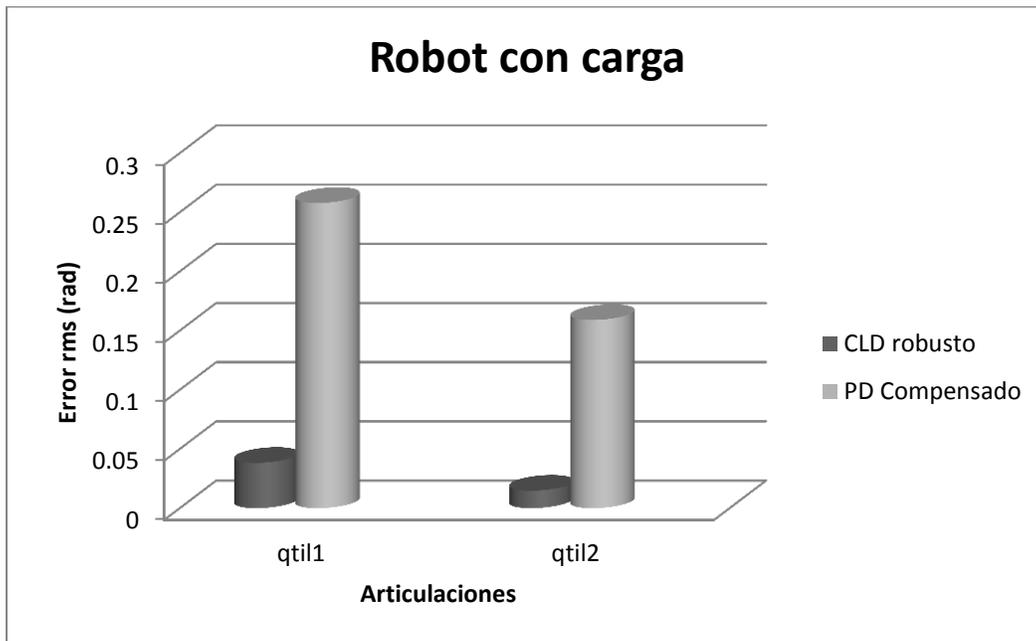


Figura 4. 34 Gráfico comparativo resultados experimentales prueba 2

V CONCLUSIONES

Hay varias fuentes de incertidumbre en la dinámica de los robots manipuladores, tales como; variación en la carga del manipulador, fricciones, dinámicas no modeladas, y además el desgaste que sufren los mecanismos actuadores, que provocan hacer imposible obtener de manera exacta su modelo matemático. La mayoría de los controladores tales como el PD compensado, par computado, PD precompensado, necesitan tener conocimiento a priori de la dinámica del manipulador de manera exacta, siendo esto difícil de obtener, los controladores resultan no tener un buen desempeño.

El trabajo de investigación realizado se basó principalmente en diseñar un controlador lógico difuso, el cual, ante las incertidumbres mencionadas en el párrafo anterior, tuviera un mejor desempeño en comparación con algoritmos de control clásico: PD con compensación y el PID.

De acuerdo a los resultados presentados en la implementación práctica de cada controlador, se obtuvo un buen desempeño del controlador difuso propuesto en comparación con los algoritmos PD compensado y el PID, siendo el factor de comparación el error rms o eficaz de posición, cumpliéndose con el objetivo principal de la tesis.

Otra aportación del trabajo realizado es que se presenta una metodología para la elaboración de la base de reglas, la cual se basó principalmente en la teoría de controladores con estructura variable en específico los modos deslizantes, esto se consiguió gracias a que existe una similitud entre los controladores de estructura variable (CEV) y los controladores difusos, pero a su vez no se presenta el problema de “castaño”, principal desventaja de los CEV debido a términos discontinuos, dado que la salida de controlador difuso es continua, como se demuestra en los teoremas 1 y 2.

En la implementación experimental, el principal problema al que se enfrentó el controlador difuso fue la sintonización del mismo, ya que son 5 ganancias a ajustar para cada eslabón, en comparación con los algoritmos PD y PID, los cuales únicamente cuentan con 2 y 3 ganancias de sintonía, respectivamente, pudiendo ser esta una desventaja para el controlador difuso ya que requeriría mayor tiempo de sintonización, para lograr un desempeño perfecto.

A pesar de que el control difuso ha sido blanco de varias críticas porque no cuenta con un estudio de estabilidad, en este trabajo se presentó un análisis metódico para determinar la “robustez” que presenta ante incertidumbres, utilizando el teorema de la cota final (por sus siglas en inglés U.U.B.) y parte de la teoría de control robusto.

VI BIBLIOGRAFÍA

- Al-Hadithi, B. M., F. Matía y A. Jiménez. 2005. Controladores borrosos basados en estructura variable con modos deslizantes: aspectos y similitudes. *RIAI*. 2: 19-35.
- Alzate, A. y G. Suarez. 2006. Aplicación de controles inteligentes sobre sistemas no lineales análisis de estabilidad en el plano de fase. *Sientia et Technica*. 12: 169-174
- Barrientos, A., L.F. Peñin, C. Balaguer, y R. Aracil. 2007. *Fundamentos de robótica*. México, D.F. McGraw-Hill
- Craig, J.J. *Robótica*. 2006. *Fundamentos de robótica*. Mexico, McGraw-Hill
- Dac Loc, H., y C. Cuong. 2004. An adaptative fuzzy logic controller for robot manipulator. *International Journal of Advanced Robotic Systems*. 2: 115-117
- Jiménez, B.L. 2000. Algunas aplicaciones de lógica difusa a teoría de control. Tesis de maestría, Instituto Politécnico Nacional, Tijuana.
- Juárez, D. y J. Zavala. 2004. Simulación de sistema difuso para el control de velocidad de un motor de CD. Páginas 25-34. Memorias de 3er. Congreso de computo AGECOMP, UAEM, México.
- Khalil, H. K. 1992. *Nonlinear Systems*. New York. Macmillan
- Kelly, R., y V. Santibáñez. 2003. *Control de Movimiento de Robots Manipuladores*. Madrid. Pearson Educación
- Kuo, T.C. 2010. Trajectory control of a robotic manipulator utilizing an adaptive fuzzy sliding mode. *World Academy of Science, Engineering and Technology*. 65:913-917

- Lee, C.C. 1990. Fuzzy Logic in control Systems: Fuzzy Logic Controller-Part I. IEEE transactions on systems, man and cybernetics. 20:404-418
- López, A.A. 2001. Usos y Abusos de la logica difusa. Con mantenimiento Productivo 8:12-17.
- López, J., y P. Muñoz. 2007. Control de un motor utilizando lógica difusa con reglas sintonizadas por algoritmos genéticos. Scientia et Technica 13: 121-127.
- Maldonado del toro, H. M., R. Silva, E.R. Ramos, V. M. Hernandez, y J.C. Rivera. 2011. Modelado y simulacion de un robot rígido de dos grado de libertad. Lat. Am. J. Phys. Educ.5:321-327.
- Martin del Brio, B. y A. Sanz. 2007. Redes Neuronales y Sistemas Borrosos. 3a edicion. Mexico. Alfaomega.
- Martínez, J., J. Bowles, y P.Mills. 1996. A fuzzy logic positioning system for an articulated Robot arm: 251-257
- Ollero, A. 2007. Manipuladores y robots móviles. Mexico, D.F. Alfaomega
- Palm, R. 1992. Sliding mode fuzzy control. IEEE Int. Conf. Fuzzy Systems. San Diego, C.A. pp. 519-526.
- Passino, K.M. y S. Yurkovich. 1998. Fuzzy control Book. Ohio U.S. Addison-Wesley Longman Inc.
- Pires, J.N., y T. Godinho. 2008. Control difuso de fuerza en robotica industrial. Ingenierias. 21: 55-62.

- Ray, K.S., y D.D. Majunder. 1984. Application of circle criterion for stability analysis of linear siso and mimo systems associated with fuzzy logic controler. IEEE Transactions on Systems, Man and Cybernetics. 14: 345-349.
- Rezoug, A. S., S. Boudoua y F. Harmerlain. 2009. Fuzzy Logic Control for Manipulator Robot actuated by Pneumatic Artificial Muscles. Journal of electrical systems. 01: 1-6.
- Sala, A. y C.V. Ariño. 2009. Reduciendo distancias entre el control difuso y el control no lineal: luces y sombras. Revista Iberoamericana de automática e informática industrial. 6: 26-35
- Sciavicco, L.y B. Siciliano. 2000. Modelling and Control of robot manipulator. London. Springer.
- Spong, M. W. 1992. On the robust control of robot manipulators. IEEE trans. Automat. Contr. 37:1782-1786
- Sharkawy, A. B., M.M. Othman y A. M. Khalil. 2011. A robust fuzzy tracking control scheme for robotic manipulators with experimental verification. Intelligent control and automation. 2: 100-111.
- Tang, K. L. y R. J. Mullholand. 1988. Comparing fuzzy logic with classical controllers designs. IEEE Transactions on Systems, Man and Cybernetics. 17:1085-1087.
- Wakileh, B. A. M. y K. F. Gill. 1988. Use of fuzzy logic in robotics. Computers in Industry. 17: 35-46
- Wang, X.L. 1995. A course in fuzzy systems and control. Upper Saddle River, N.J. Prentice-Hall, PTR.

Yeong, Y.S. y M. J. Chung. 1997. A robust fuzzy logic controller for robot manipulator with uncertainties. IEEE transactions on systems, Man and Cybernetics. 27: 706-713.

Yeong, Y.S. y M.J. Chung. 1994. Systematic design and stability analysis of a fuzzy controller. Fuzzy Sets and systems. 91: 271-298

Yeong, Y.S. y M.J. Chung. 1993. Design of a Fuzzy Logic Controller for robot manipulators in VSS control Scheme. Proc. IFSA. 11: 1207-1210.

Zeinali, M. y L. Notash. 2010. Fuzzy logic-based inverse dynamic modelling robot manipulators. Transactions of the Canadian Society for Mechanical Engineering. 34:137-15

APÉNDICES

Apéndice A Preliminares matemáticos y propiedades

A.1 Propiedades del modelo dinámico de robots manipuladores³.

En este apéndice se presentan las propiedades del modelo dinámico de robots manipuladores caracterizados por la ecuación:

$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} + g(q) = \tau \quad (\text{A.1})$$

Las propiedades de cada componente de la ecuación (A.1) serán revisadas a lo largo de este epígrafe, las cuales están divididas en: matriz de inercia $M(q)$, matriz centrífuga y de Coriolis ($C(q, \dot{q})$), el vector de gravedad $g(q)$, además de la propiedad de linealidad en los parámetros dinámicos (de principal importancia para el desarrollo de esta investigación).

Propiedades de la Matriz de inercia.

La matriz de inercia es parte importante tanto en el modelado, así como en el diseño de controladores para robots, debido a que algunas de sus propiedades se utilizan para el análisis de estabilidad de los sistemas de control. A continuación se listan algunas propiedades

La matriz $M(q)$ es de $n \times n$, simétrica y definida positiva, además sus elementos son funciones únicamente de q .

1.- Existe una constante positiva α , real tal que:

$$M(q) \geq \alpha I \quad \forall q \in \mathbb{R}^n \quad (\text{A.2})$$

³ Kelly y Santibáñez (2003)

donde I denota la matriz identidad de dimensión $n \times n$. Además la inversa de la matriz de inercia existe y es definida positiva.

2.- únicamente para el caso de robots provistos de articulaciones rotacionales, existe una constante $\beta > 0$ tal que:

$$\lambda_{\max}(M(q)) \leq \beta, \quad \forall q \in R^n \quad (\text{A.3})$$

Una manera de calcular β está dada por

$$\beta \geq n \left(\max_{i,j,q} |M_{ij}(q)| \right).$$

3.- Para robots con articulaciones rotativas, existe una constante $K_M > 0$ tal que:

$$\|M(x)z - M(y)z\| \leq K_M \|x - y\| \|z\| \quad \forall x, y, z \in R^n \quad (\text{A.4})$$

Una manera de calcular K_M es:

$$K_M \geq n^2 \left(\max_{i,j,k,q} \left| \frac{\partial M_{ij}(q)}{\partial q_k} \right| \right)$$

Propiedades de la matriz de Coriolis $C(q, \dot{q})$

La importancia de las propiedades de la matriz centrífuga y de Coriolis se hace patente en el estudio de estabilidad de los sistemas de control de robot. Las principales propiedades de dicha matriz se listan a continuación.

La matriz de fuerzas centrífugas y de Coriolis es una matriz de $n \times n$ cuyos elementos son funciones de q y \dot{q} y satisface las siguientes propiedades

1. La matriz $C(q, \dot{q})$ puede ser no única pero el vector $C(q, \dot{q}) \dot{q}$ es único.
2. $C(q, 0) = 0, \forall q \in R^n$
3. Para todo vector $q, x, y, z \in R^n$ y escalar α , se tiene que:

$$\begin{aligned}
C(q, x)y &= C(q, y)x, \\
C(q, z + \alpha x)y &= C(q, z)y + \alpha C(q, x)y
\end{aligned} \tag{A.5}$$

4. Para robots con articulaciones rotativas, existe una constante $K_{C1} > 0$ tal que:

$$\|C(q, x)y\| \leq K_{C1} \|x\| \|y\| \quad \forall q, x, y \in \mathbb{R}^n \tag{A.6}$$

El cual se puede representar como:

$$K_{C1} \geq n^2 \left(\max_{i, j, k, q} |C_{kij}(q)| \right)$$

5. Para el caso de robots con articulaciones rotativas, existen constantes $K_{C1} > 0$ y $K_{C2} > 0$ tales que:

$$\|C(x, z)w - C(y, v)w\| \leq k_{C1} \|z - v\| \|w\| + k_{C2} \|x - y\| \|w\| \|z\| \quad \forall v, x, y, z, w \in \mathbb{R}^n \tag{A.7}$$

6. La matriz $C(q, \dot{q})$ está relacionada con la matriz de inercia $M(q)$ por la expresión:

$$x^T \left[\frac{1}{2} \dot{M}(q) - C(q, \dot{q}) \right] x = 0 \quad \forall q, \dot{q}, x \in \mathbb{R}^n \tag{A.8}$$

y de hecho $\frac{1}{2} \dot{M}(q) - C(q, \dot{q})$ es una matriz antisimétrica. En forma análoga la matriz

$\dot{M}(q) - 2C(q, \dot{q})$ es antisimétrica, y también resulta cierto que:

$$\dot{M}(q) = C(q, \dot{q}) + C^T(q, \dot{q}) \tag{A.9}$$

Propiedades del vector de gravedad $g(q)$

El vector de pares gravitacionales $g(q)$ está presente en robots que no han sido diseñados desde un punto de vista mecánica, con compensación de pares de gravedad, es decir, sin contrapesos o resortes. Algunas de sus propiedades más importantes se citan continuación.

El vector de pares gravitacionales $g(q)$ de $n \times 1$ depende solo de las posiciones articulares q . El vector $g(q)$ esta acotado si q lo está también.

1. El vector $g(q)$ y el vector de velocidad \dot{q} pueden relacionarse mediante:

$$\int_0^T g(q)^T \dot{q} dt = U(q(t)) - U(q(0)) \quad \forall T \in \mathbb{R}_+ \quad (\text{A.10})$$

2. Para el caso de robots provistos únicamente de articulaciones rotacionales, existe una constante finita k_U tal que:

$$\int_0^T g(q)^T \dot{q} dt + U(q(0)) \geq k_U \quad \forall T \in \mathbb{R}_+ \quad y \quad k_U = \min_q \{U(q)\} \quad (\text{A.11})$$

3. Para robots con articulaciones rotativas, el vector $g(q)$ es Lipschitz lo cual significa que existe una constante $k_g > 0$ tal que:

$$\|g(x) - g(y)\| \leq K_g \|x - y\| \quad \forall x, y \in \mathbb{R}^n \quad (\text{A.12})$$

Una manera sencilla de calcular k_g es:

$$K_g \geq n \left(\max_{i,j,q} \left| \frac{\partial g_i(q)}{\partial q_j} \right| \right)$$

Más aun, K_g satisface:

$$k_g \geq \left\| \frac{\partial g(q)}{\partial q} \right\| \geq \lambda_{\max} \left\{ \frac{\partial g(q)}{\partial q} \right\}$$

3.-Para robots que solo tienen uniones rotativas, existe una constante $K' > 0$ tal que:

$$\|g(q)\| \leq k, \quad \forall q \in \mathbb{R}^n \quad (\text{A.13})$$

A.2 Definiciones de álgebra lineal

Producto de vectores

El producto interno de dos vectores $x, y \in \mathbb{R}^n$ está definido como:

$$x^T y = [x_1 \quad x_2 \quad \dots \quad x_n] \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{bmatrix}^T \begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_n \end{bmatrix} \quad (\text{A.14})$$

y además satisface las propiedades

$$* x^T y = y^T x$$

$$* x^T (y+z) = x^T y + x^T z$$

Norma euclidiana

La norma euclidiana $\|x\|$ de un vector x se define como:

$$\|x\| = \sqrt{\sum x_i^2} = \sqrt{x^T x} \quad (\text{A.15})$$

la cual satisface las siguientes propiedades

$$* \|x\| = 0 \text{ si y solo si } x=0 \in \mathbb{R}^n.$$

$$* \|x\| > 0 \text{ para todo } x \in \mathbb{R}^n \text{ con } x \neq 0 \in \mathbb{R}^n.$$

$$* \|\alpha x\| = |\alpha| \|x\| \text{ para todo } \alpha \in \mathbb{R} \text{ y si } x \in \mathbb{R}^n.$$

$$* \|x\| - \|y\| \leq \|x + y\| \leq \|x\| + \|y\|$$

$$* \left| x^T y \right| = \|x\| \|y\|$$

Matrices particulares

Una matriz A es cuadrada si $n=m$. una matriz cuadrada es simétrica si esta es igual a su transpuesta, es decir, $A=A^T$, mientras que es antisimétrica si $A=-A^T$. una propiedad de las matrices antisimétricas es: $x^T Ax = 0$

Una matriz cuadrada es singular si su determinante es nulo, es decir, si $\det[A]=0$, en caso contrario es no singular. A su vez, una matriz cuadrada, es definida positiva, sin ser necesariamente simétrica, si: $x^T Ax > 0$. Una matriz cuadrada, no necesariamente simétrica, es semidefinida positiva si: $x^T Ax \geq 0$

Cualquier matriz simétrica y definida positiva $A=A^T > 0$ es no singular; por tanto su inversa existe. Una matriz cuadrada es definida negativa si $-A$ es definida positiva mientras que es semidefinida negativa si $-A$ es semidefinida positiva.

Valores propios

Para cada matriz cuadrada A existen n valores propios o eigenvalores denotados por $\lambda_i\{A\}$.

Para el caso de una matriz simétrica, sus valores propios son tales que:

* $\lambda_1\{A\}, \lambda_2\{A\}, \dots, \lambda_n\{A\}$ son números reales.

*denotando los valores máximos y mínimos de A por λ_{\max} y λ_{\min} , respectivamente, el

teorema de Rayleigh-Ritz establece que para todo $x \in \mathbb{R}^n$ se tiene:

$$\lambda_{\max}\{A\} \|x\|^2 \geq x^T Ax \geq \lambda_{\min}\{A\} \|x\|^2 \tag{A.16}$$

Norma espectral

La norma espectral $\|A\|$ de una matriz $A \in \mathbb{R}^{n \times m}$ se define como: $\|A\| = \sqrt{\lambda_{\max}\{A^T A\}}$ donde

$\lambda_{\max}\{A^T A\}$ denota el eigenvalor máximo de la matriz simétrica. Para el caso particular de matrices simétricas, se tiene que:

$$* \|A\| = \max_i |\lambda_i\{A\}|$$

$$* \|A^{-1}\| = \frac{1}{\min_i |\lambda_i\{A\}|}$$

Apéndice B Sistemas de control de Estructura Variable⁴ (CEV)

B.1 Fundamentos del CEV

Un sistema de estructura variable se compone de dos o más subsistemas continuos y una lógica que realiza la conmutación entre ellos en función de los estados del sistema. La acción de control que resulta de esta ley de conmutación es una función discontinua de los estados. Un modo de operación particular se obtiene cuando las conmutaciones ocurren a frecuencia muy elevada, restringiendo la trayectoria de los estados del sistema a una variedad en el espacio de estados. Este modo de operación es llamado modo o régimen deslizante (MD) y presenta propiedades muy atractivas. Entre otras, es robusto a perturbaciones externas y La incertidumbre en los parámetros, el sistema de lazo cerrado es de orden reducido, y su dinámica es gobernada mediante el diseño de la función de conmutación.

Entre los sistemas susceptibles de aplicar la teoría del CEV se tienen: sistemas no lineales, sistemas con retraso, sistemas de gran escala, sistemas multivariables , etc. Aparte de su aplicación principal, que es la estabilización de sistemas, las técnicas del VSC comienzan a aplicarse al control de sistemas de seguimiento de movimientos, control óptimo y al control adaptativo.

El siguiente ejemplo puede dar una idea básica del CEV. Dado el sistema de segundo orden:

$$\ddot{x} + a\dot{x} + (b+k)x = 0$$

Si la ley de control kx se elige de forma que k cambie entre valores diferentes, modificando la estructura del sistema en cadena cerrada, se obtienen trayectorias diferentes de las que se

⁴ Al-Hadithi, *et al.* (2005)

podrían obtener con un valor de k fijo. Si se supone que el sistema no tiene amortiguamiento ($a=0$) y que k toma valores:

$$k = \begin{cases} \alpha - b \\ -\alpha - b \end{cases}$$

El sistema en cadena cerrada tendrá dos estructuras:

$$\begin{aligned} \ddot{x} + \alpha x &= 0 \\ \ddot{x} - \alpha x &= 0 \end{aligned}$$

Las trayectorias en los dos casos se muestran en la figura B.1 (A) y (B). En el caso de la última ecuación, existe un único sub-espacio estable definido por la siguiente ecuación: $\dot{x} + \sqrt{\alpha x} = 0$, utilizando esta línea y la línea $x=0$, se divide el plano de fases en cuatro regiones como se ve en la figura B.1 (C). Se elige el control de la siguiente forma:

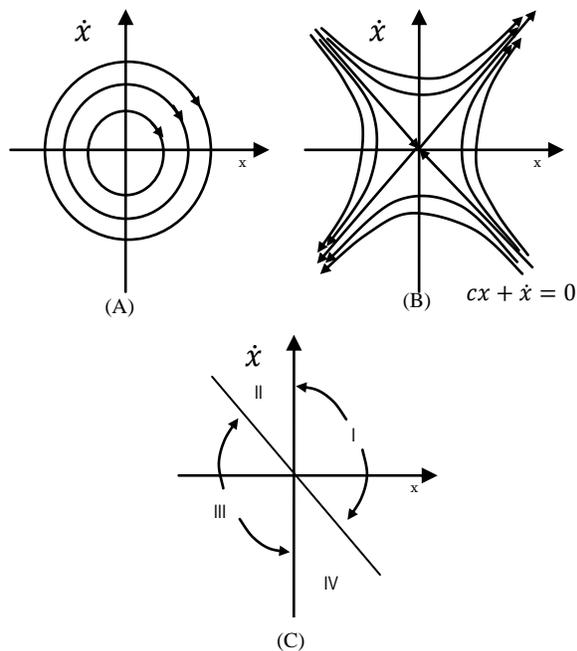
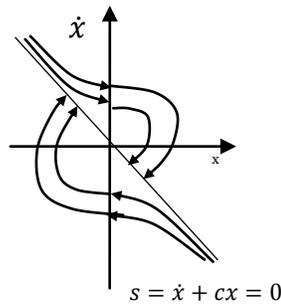


Figura B.1 Sistema de estructura variable

$$u(t) = \begin{cases} (\alpha - b)x & \text{en las regiones I y III} \\ (-\alpha - b)x & \text{en las regiones II y IV} \end{cases}$$

Las trayectorias del sistema en cadena cerrada se muestran en la figura B.2 y el sistema resultante es globalmente estable, a pesar de que los dos sistemas realimentados de los cuales se formó son críticamente estable e inestable respectivamente. Las respuestas de $x(t)$ para condiciones iniciales en la región I y II se muestran en la figura B.3



B.2 trayectorias en cadena cerrada del sistema

En el ejemplo anterior, las trayectorias del sistema resultante con conmutación constan de trozos de las trayectorias de los sistemas realimentados sin conmutación. Es también posible obtener movimientos en el sistema de conmutación que no tengan relación con los sistemas sin conmutación de los cuales se forma.

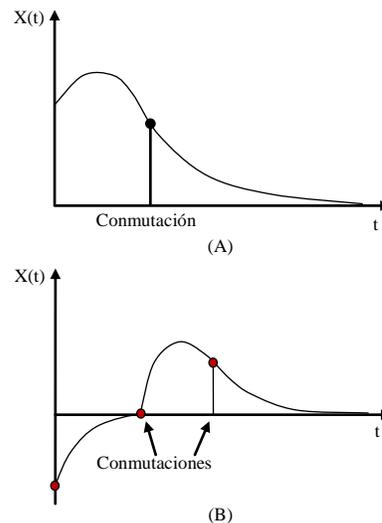


Figura B.3 Las trayectorias del sistema como función del tiempo: (A) La condición inicial en la región I; (B) La condición inicial en la región II

B.2 Descripción matemática del CEV con MD

En el CEV, los algoritmos de diseño incluyen la selección de las funciones deslizantes $s(x)$ deseadas, esto es, la elección de sus parámetros. Posteriormente se implementa un control discontinuo que asegura la existencia de los modos deslizantes en cada punto del plano deslizante $s(x)=0$. En la etapa final, el control debe dirigir los estados del sistema hacia el plano deslizante.

En general, las dinámicas del CEV constan de dos condiciones: la condición de alcance y la condición deslizante. En la primera, el objetivo es llegar al plano de conmutación en un tiempo finito desde cualquier punto. También se denomina modo no deslizante. La segunda parte se denomina el modo deslizante, durante la cual la trayectoria tiende asintóticamente al origen del espacio de estados. Los parámetros de la función de conmutación condicionan el comportamiento final del sistema durante el modo deslizante.

Se pueden observar cuatro características del sistema de estructura variable:

- * Como el origen del espacio de estados representa el estado de equilibrio del sistema, el MD representa el comportamiento del mismo durante el periodo transitorio. En otras palabras, la superficie que describe $s(x)=0$ define la respuesta transitoria del sistema durante MD.
- * Durante el MD; las dinámicas de la trayectoria son de un orden menor que el orden del sistema original.
- * Durante el MD, la dinámica de la trayectoria está gobernada solamente por los parámetros que describen la función deslizante.
- * EL MD es una trayectoria que no es inherente a ninguna de las estructuras de control.

La ley de control del sistema variable se cambia cuando su estado cruza la línea de conmutación. En general, la entrada del control puede que no sea suave, e incluso a veces

es discontinua durante la conmutación y puede excitar dinámicas no modeladas de alta frecuencia.

B.3 Reducción del ruido continuo

Una de las características del diseño y análisis de sistemas de CEV es que la acción de control puede conmutar de un valor a otro de forma infinitamente rápida. Sin embargo, en sistemas prácticos, el ruido continuo es indeseable, porque es imposible conseguir el control de alta conmutación necesario en la mayoría de diseños de CEV, debido a muchas razones. Una de ellas es la existencia de un retardo en el cómputo del control. La segunda razón es la limitación de los actuadores físicos. El ruido continuo siempre ocurre en los modos deslizantes y en régimen permanente. En régimen permanente, el ruido continuo aparece como una oscilación de alta frecuencia alrededor del punto deseado de equilibrio y puede resultar una fuente de excitación de las dinámicas de alta frecuencia del sistema no modeladas. Por tanto, el ruido continuo debe eliminarse de modo que el control actúe adecuadamente.

Existen distintas aproximaciones para eliminar el ruido continuo. En varios diseños del CEV, el control tiene términos similares a un relé. Las propiedades ideales de un relé son prácticamente imposibles de conseguir. Por ello un enfoque para reducir el ruido continuo es reemplazar el relé por una saturación (aproximación continua). Otro enfoque sugiere añadir una capa límite alrededor de la superficie de conmutación para alisar la discontinuidad del control. Dentro de esta capa, el control se elige para ser una aproximación continua de la función de conmutación.

Apéndice C CÓDIGOS DE PROGRAMAS

C.1 Código utilizado para las simulaciones en Matlab.

Para la simulación de los controladores se utilizaron los siguientes códigos.

CONTROL PID

```
%robot 2 grados de libertad CONTROL PID
clear all;
close all;
%clc;

%% parámetros del robot
l1=0.193;           %longitud de eslabón 1
l2=0.145;           %longitud de eslabón 2
lc1=.136;           %distancia al centro de masa 1
lc2=.048;           %distancia al centro de masa 2
m1=0.763;           %masa 1
m2=0.343;           %masa 2
I1=0.0186;          %inercia eslabón 1
I2=0.0026987;       %inercia eslabón 2
g=9.81              %aceleración de gravedad

A=m1*lc1^2+m2*(l1^2+lc2^2)+I1+I2;
B=m2*l1*lc2;
C=m2*lc2^2+I2;
H=m1*g*lc1+m2*g*l1;
F=m2*g*lc2;
G=m2*l1*lc2;

%% valores iniciales
x1(1)=0;           %posición articular 1
x2(1)=0;           %posición articular 2
x3(1)=0;           %velocidad articular 1
x4(1)=0;           %velocidad articular 2
pd1(1)=0;
pd2(1)=0;
tau1(1)=0;         %par 1 en instante t cero
tau2(1)=0;         %par 2 en instante t cero

step=.01; %dimensión del paso de integración
kT(1)=1*step;

%% ganancias control PID
kp1=12; ki1=8.8; kd1=0.45;
kp2=3.42; ki2=1.3; kd2=0.05;

auxe=0; %auxiliares para almacenar la suma del error cuadrático
```

```

auxe2=0;
intel=0; %auxiliares para almacenar la suma del área
inte2=0;
areal=0;
area2=0;
%% inicio de simulación
for b=2:12500
    % asignación de posiciones, velocidades y aceleraciones deseadas
    pd1(b)=-pi*sin(b*0.1*step);
    pdd1(b)=-pi*0.1*cos(b*0.1*step);
    pd2(b)=pi*sin(b*0.1*step);
    pdd2(b)=pi*0.1*cos(b*0.1*step);
    pdd1d(b)=pi*0.1^2*sin(b*0.1*step);
    pdd2d(b)=-pi*0.1^2*sin(b*0.1*step);

    q1(b)=x1(b); %salida posición
    q2(b)=x2(b);
    q1p(b)=x3(b);
    q2p(b)=x4(b);

    %cálculo del err de posición
    e1(b)=pd1(b)-q1(b);
    e2(b)=pd2(b)-q2(b);

    %cálculo del error de velocidad
    de1(b)=pdd1(b)-q1p(b);
    de2(b)=pdd2(b)-q2p(b);

    %cálculo de la integral
    areal=(e1(b)*step);
    intel=intel+areal;
    area2=(e2(b)*step);
    inte2=inte2+area2;

    %señal de control
    tau1(b)=kp1*e1(b)+kd1*de1(b)+ki1*intel;
    tau2(b)=kp2*e2(b)+kd2*de2(b)+ki2*inte2;

    %%modelo del robot

    m11=A+2*B*cos(q2(b));
    m12=C+B*cos(q2(b));
    m21=m12;
    m22=C;
    C1=-G*(2*q1p(b)+q2p(b))*q2p(b)*sin(q2(b));
    C2=G*q1p(b)*q1p(b)*sin(q2(b));
    g1=H*sin(q1(b))+F*sin(q1(b)+q2(b));
    g2=F*sin(q1(b)+q2(b));

    aux1=m11*m22-m12*m21;

    v1= tau1(b)-C1-g1;
    v2= tau2(b)-C2-g2;
    f1=q1p(b);
    f2=q2p(b);

```

```

f3=( (m22*v1)-(m12*v2))/aux1;
f4=( (-m12*v1)+(m11*v2))/aux1;
x1(b+1)=f1*step+q1(b);
x2(b+1)=f2*step+q2(b);
x3(b+1)=f3*step+q1p(b);
x4(b+1)=f4*step+q2p(b);
KT(b)=(b)*step;
auxe=auxe+e1(b)^2;
auxe2=auxe2+e2(b)^2;
%cambio de masa en eslabón 2 a los 25 seg.
if KT(b)==25
    m2=0.343+0.5*(0.343);
    lc2=.048+0.5*(0.048);
    I2=0.0026987+1*(0.0026987);
    A=m1*lc1^2+m2*(l1^2+lc2^2)+I1+I2;
    B=m2*l1*lc2;
    C=m2*lc2^2+I2;
    H=m1*g*lc1+m2*g*l1;
    F=m2*g*lc2;
    G=m2*l1*lc2;
end
%cambio de masa en eslabón 2 a los 85 seg.
if(KT(b)==85)
    m2=0.343;
    lc2=.048;
    I2=0.0026987;
    A=m1*lc1^2+m2*(l1^2+lc2^2)+I1+I2;
    B=m2*l1*lc2;
    C=m2*lc2^2+I2;
    H=m1*g*lc1+m2*g*l1;
    F=m2*g*lc2;
    G=m2*l1*lc2;
end
end
%cálculo del error eficaz
errorrms1=(auxe/6000)^(1/2)
errorrms2=(auxe2/6000)^(1/2)

%instrucciones para obtener las gráficas
subplot(1,2,1),
plot(KT,q1,'b',KT,pd1,'r');
grid on;
legend('q1','qd1');
xlabel('Tiempo (s)');
ylabel('Posición (rad)');
title('Articulación 1')
subplot(1,2,2),plot(KT,taul,'b');
xlabel('Tiempo (s)');
ylabel('Par (N.m)');
title('Señal de control')
grid on

```

CONTROL PD

Para el control PD se utilizó el código siguiente

```
%robot 2 grados de libertad CONTROL PD
clear all;
close all;
%clc;

%% parámetros del robot
l1=0.193;           %longitud de eslabón 1
l2=0.145;           %longitud de eslabón 2
lc1=.136;           %distancia al centro de masa 1
lc2=.048;           %distancia al centro de masa 2
m1=0.763;           %masa 1
m2=0.343;           %masa 2
I1=0.0186;          %inercia eslabón 1
I2=0.0026987;       %inercia eslabón 2
g=9.81              %aceleración de gravedad

A=m1*lc1^2+m2*(l1^2+lc2^2)+I1+I2;
B=m2*l1*lc2;
C=m2*lc2^2+I2;
H=m1*g*lc1+m2*g*l1;
F=m2*g*lc2;
G=m2*l1*lc2;

%% valores iniciales
x1(1)=0;           %posición articular 1
x2(1)=0;           %posición articular 2
x3(1)=0;           %velocidad articular 1
x4(1)=0;           %velocidad articular 2
pd1(1)=0;
pd2(1)=0;
tau1(1)=0;         %par 1 en instante t cero
tau2(1)=0;         %par 2 en instante t cero

step=.01; %dimensión del paso de integración
kT(1)=1*step;

%% ganancias control PD
k1=0.3;lambda1=12.5;
k2=0.1;lambda2=11;
%%
auxe=0; %auxiliares para almacenar la suma del error cuadrático
auxe2=0;

%% inicio de simulación por 125 sg.
for b=2:12500
    % asignación de posiciones, velocidades y aceleraciones deseadas
    pd1(b)=-pi*sin(b*0.1*step);
    pdd1(b)=-pi*0.1*cos(b*0.1*step);
    pd2(b)=pi*sin(b*0.1*step);
    pdd2(b)=pi*0.1*cos(b*0.1*step);
    pddld(b)=pi*0.1^2*sin(b*0.1*step);
```

```

pdd2d(b)=-pi*0.1^2*sin(b*0.1*step);

q1(b)=x1(b); %SALIDA posición
q2(b)=x2(b);
q1p(b)=x3(b);
q2p(b)=x4(b);

%cálculo del error de posición
e1(b)=pd1(b)-q1(b);
e2(b)=pd2(b)-q2(b);

%cálculo del error de velocidad
de1(b)=pdd1(b)-q1p(b);
de2(b)=pdd2(b)-q2p(b);

%cálculo de V y a para el cambio de variable
V1=pdd1(b)-lambda1*e1(b);
V2=pdd2(b)-lambda2*e2(b);

a11=pdd1d(b)-lambda1*de1(b);
a12=pdd2d(b)-lambda2*de2(b);

%Matriz Y
Yav=[a11 a11+a12 cos(q2(b))*(2*a11+a12) sin(q2(b))*(q2p(b)*V2+q1p(b)
      *V2+q2p(b)*V1) g*sin(q1(b)) g*sin(q1(b)) g*sin(q1(b)+q2(b));
      0 a11+a12 cos(q2(b))*a11+sin(q2(b))*q1p(b)*V1 0 0
      g*sin(q1(b)+q2(b))];

r1(b)=de1(b)+lambda1*e1(b);
r2(b)=de2(b)+lambda2*e2(b);

%calculó de Y*th
T2=Yav*th;

tau1(b)=T2(1,1)-k1*r1(b);
tau2(b)=T2(2,1)-k2*r2(b);

%señal de control
tau1(b)=kp1*e1(b)+kd1*de1(b)+ki1*inte1;
tau2(b)=kp2*e2(b)+kd2*de2(b)+ki2*inte2;

%modelo del robot

m11=A+2*B*cos(q2(b));
m12=C+B*cos(q2(b));
m21=m12;
m22=C;
C1=-G*(2*q1p(b)+q2p(b))*q2p(b)*sin(q2(b));
C2=G*q1p(b)*q1p(b)*sin(q2(b));
g1=H*sin(q1(b))+F*sin(q1(b)+q2(b));
g2=F*sin(q1(b)+q2(b));

```

```

aux1=m11*m22-m12*m21;

v1= tau1(b)-C1-g1;
v2= tau2(b)-C2-g2;
f1=q1p(b);
f2=q2p(b);
f3=(m22*v1)-(m12*v2)/aux1;
f4=(-m12*v1)+(m11*v2)/aux1;
x1(b+1)=f1*step+q1(b);
x2(b+1)=f2*step+q2(b);
x3(b+1)=f3*step+q1p(b);
x4(b+1)=f4*step+q2p(b);
KT(b)=(b)*step;
auxe=auxe+e1(b)^2;
auxe2=auxe2+e2(b)^2;
%cambio de masa en eslabón 2 a los 25 seg.
if KT(b)==25
    m2=0.343+1*(0.343);
    lc2=.048+1*(0.048);
    I2=0.0026987+1*(0.0026987);
    A=m1*lc1^2+m2*(l1^2+lc2^2)+I1+I2;
    B=m2*l1*lc2;
    C=m2*lc2^2+I2;
    H=m1*g*lc1+m2*g*l1;
    F=m2*g*lc2;
    G=m2*l1*lc2;
end
if(KT(b)==85)
    m2=0.343;
    lc2=.048;
    I2=0.0026987;
    A=m1*lc1^2+m2*(l1^2+lc2^2)+I1+I2;
    B=m2*l1*lc2;
    C=m2*lc2^2+I2;
    H=m1*g*lc1+m2*g*l1;
    F=m2*g*lc2;
    G=m2*l1*lc2;
end
end
%cálculo del error eficaz
errorrms1=(auxe/12500)^(1/2)
errorrms2=(auxe2/12500)^(1/2)

%instrucciones para obtener las gráficas
subplot(1,2,1),
plot(KT,q1,'b',KT,pd1,'r');
grid on;
legend('q1','qd1');
xlabel('Tiempo (s)');
ylabel('Posición (rad)');
title('Articulación 1')
subplot(1,2,2),plot(KT,tau1,'b');
xlabel('Tiempo (s)');
ylabel('Par (N.m)');
title('Señal de control')
grid on

```

CONTROL DIFUSO

```
%robot 2 grados de libertad CONTROL DIFUSO
clear all;
close all;
%clc;

%%
l1=0.193;           %longitud de eslabón 1
l2=0.145;           %longitud de eslabón 2
lc1=.136;           %distancia al centro de masa 1
lc2=.048;           %distancia al centro de masa 2
m1=0.763;           %masa 1
m2=0.343;           %masa 2
I1=0.0186;         %inercia eslabón 1
I2=0.0026987;      %inercia eslabón 2
g=9.81              %aceleración de gravedad

th=[m1*lc1^2+m2*l1^2+I1;m2*lc2^2+I2;m2*l1*lc2;m1*lc1;m2*l1;m2*lc2];

A=m1*lc1^2+m2*(l1^2+lc2^2)+I1+I2;
B=m2*l1*lc2;
C=m2*lc2^2+I2;
H=m1*g*lc1+m2*g*l1;
F=m2*g*lc2;
G=m2*l1*lc2;
%% valores iniciales
x1(1)=0;           %posición articular 1
x2(1)=0;           %posición articular 2
x3(1)=0;           %velocidad articular 1
x4(1)=0;           %velocidad articular 2

pd1(1)=0;
pd2(1)=0;
tau1(1)=0;         %par 1 en instante t cero
tau2(1)=0;         %par 2 en instante t cero

step=.01; %dimensión del paso de integración
kT(1)=1*step;
%% parámetros del controlador difuso
%número de funciones de membresía de las entradas
ep1=5; %número de funciones de membresía de la variable lingüística error
de pos. articulación 1
ev1=5; %número de funciones de membresía de la variable lingüística error
de velocidad Articulación 1
ep2=5; %número de funciones de membresía de la variable lingüística error
de pos. articulación 2
ev2=5; %número de funciones de membresía de la variable lingüística error
de velocidad. articulación 2

%%ganancias de escalamiento
gep1=4.0; %10;%ganancias de escalamiento de la entrada ep2
```

```

gev1=0.32;%0.025;%ganancias de escalamiento de la entrada ev1 0.09
ts=0.01
gep2=0.282; %7;%ganancias de escalamiento de la entrada ep2
gev2=0.02;%0.02;%ganancias de escalamiento de la entrada ev2 0.03 ts=0.01

%ganancias control PD
k1=0.3;lambda1=12.5;
k2=0.1;lambda2=11;

%centros de las funciones de membresía de entrada y salida
cep1=[-1 -0.1 0 0.1 1]; %vector de centros de las funciones de membresía
ep1
cev1=[-1 -0.1 0 0.1 1]; %vector de centros de las funciones de membresía
ev1
cep2=[-1 -0.01 0 0.01 1]; %vector de centros de las funciones de
membresía ep2
cev2=[-1 -0.01 0 0.01 1]; %vector de centros de las funciones de
membresía ev2
%%
%distancia de los centros de las funciones de membresía
MD1=[0.9 0.1 0.1 0.9 0.1];
MI1=[0.1 0.9 0.9 0.1 0.9];
MDe1=[0.9 0.1 0.1 0.9 0.1];
MIe1=[0.1 0.9 0.9 0.1 0.9];
MD2=[0.99 0.01 0.01 0.99 0.01];
MI2=[0.01 0.99 0.99 0.01 0.99];
MDe2=[0.99 0.01 0.01 0.99 0.01];
MIe2=[0.01 0.99 0.99 0.01 0.99];
%% matriz de reglas
reglas=[1 1 1 1 0;
        1 0.1 0.1 0 -1;
        1 0.1 0 -0.1 -1;
        1 0 -0.1 -0.1 -1;
        0 -1 -1 -1 -1];
%%
auxe=0;
auxe2=0;
for b=2:6000

    pd1(b)=-pi*sin(b*0.1*step);
    pdd1(b)=-pi*0.1*cos(b*0.1*step);
    pd2(b)=pi*sin(b*0.1*step);
    pdd2(b)=pi*0.1*cos(b*0.1*step);
    pdd1d(b)=pi*0.1^2*sin(b*0.1*step);
    pdd2d(b)=-pi*0.1^2*sin(b*0.1*step);
    q1(b)=x1(b); %SALIDA posición
    q2(b)=x2(b);
    q1p(b)=x3(b);
    q2p(b)=x4(b);
    %cálculo del error de posición
    e1(b)=q1(b)-pd1(b);
    e2(b)=q2(b)-pd2(b);
    %cálculo del error de velocidad
    edif1(b)=e1(b)*gev1;
    edif2(b)=e2(b)*gev2;

```

```

de1 (b)=q1p (b) -pdd1 (b) ;
de2 (b)=q2p (b) -pdd2 (b) ;
dedif1 (b)=de1 (b) *gev1;
dedif2 (b)=de2 (b) *gev2;

V1=pdd1 (b) -lambda1*e1 (b) ;
V2=pdd2 (b) -lambda2*e2 (b) ;
a11=pdd1d (b) -lambda1*de1 (b) ;
a12=pdd2d (b) -lambda2*de2 (b) ;

Yav=[a11 a11+a12 cos (q2 (b)) * (2*a11+a12) -
sin (q2 (b)) * (q2p (b) *V2+q1p (b) *V2+q2p (b) *V1) g*sin (q1 (b)) g*sin (q1 (b))
g*sin (q1 (b)+q2 (b)) ;
0 a11+a12 cos (q2 (b)) *a11+sin (q2 (b)) *q1p (b) *V1 0 0
g*sin (q1 (b)+q2 (b)) ] ;

gk1 (b)=3.2;
gk2 (b)=2.1;
%señal de control

tf1 (b)=difusol (edif1 (b) , dedif1 (b) , cep1, cev1, ep1, ev1, reglas, MD1, MI1, MDe1, M
Ie1) *gk1 (b) ;

tf2 (b)=difusol (edif2 (b) , dedif2 (b) , cep2, cev2, ep2, ev2, reglas, MD2, MI2, MDe2, M
Ie2) *gk2 (b) ;

r1 (b)=de1 (b) +lambda1*e1 (b) ;
r2 (b)=de2 (b) +lambda2*e2 (b) ;

%calculó del vector Y*th
T2=Yav*th;

tau1 (b)=T2 (1, 1) -k1*r1 (b) +tf1 (b) ;
tau2 (b)=T2 (2, 1) -k2*r2 (b) +tf2 (b) ;

%%modelo del robot

m11=A+2*B*cos (q2 (b)) ;
m12=C+B*cos (q2 (b)) ;
m21=m12;
m22=C;
C1=-G* (2*q1p (b) +q2p (b)) *q2p (b) *sin (q2 (b)) ;
C2=G*q1p (b) *q1p (b) *sin (q2 (b)) ;
g1=H*sin (q1 (b)) +F*sin (q1 (b) +q2 (b)) ;
g2=F*sin (q1 (b) +q2 (b)) ;

aux1=m11*m22-m12*m21;

v1= tau1 (b) -C1-g1;
v2= tau2 (b) -C2-g2;
f1=q1p (b) ;
f2=q2p (b) ;
f3= ( (m22*v1) - (m12*v2) ) /aux1;
f4= ( (-m12*v1) + (m11*v2) ) /aux1;

```

```

x1(b+1)=f1*step+q1(b);
x2(b+1)=f2*step+q2(b);
x3(b+1)=f3*step+q1p(b);
x4(b+1)=f4*step+q2p(b);
KT(b)=(b)*step;
auxe=auxe+e1(b)^2;
auxe2=auxe2+e2(b)^2;

if KT(b)==25
    m2=0.343+1*(0.343);
    lc2=.048+1*(0.048);
    I2=0.0026987+1*(0.0026987);
    A=m1*lc1^2+m2*(l1^2+lc2^2)+I1+I2;
    B=m2*l1*lc2;
    C=m2*lc2^2+I2;
    H=m1*g*lc1+m2*g*l1;
    F=m2*g*lc2;
    G=m2*l1*lc2;
end
if(KT(b)==85)
    m2=0.343;
    lc2=.048;
    I2=0.0026987;
    A=m1*lc1^2+m2*(l1^2+lc2^2)+I1+I2;
    B=m2*l1*lc2;
    C=m2*lc2^2+I2;
    H=m1*g*lc1+m2*g*l1;
    F=m2*g*lc2;
    G=m2*l1*lc2;
end

end

errorrms1=(auxe/6000)^(1/2)
errorrms2=(auxe2/6000)^(1/2)
subplot(1,2,1),
plot(KT,q2,'b',KT,pd2,'r');
grid on;
legend('q2','qd2');
xlabel('Tiempo (s)');
ylabel('Posición (rad)');
title('Articulación 2')
subplot(1,2,2),plot(KT,tau2,'b');% ,KT,tau2,'r');
xlabel('Tiempo (s)');
ylabel('Par (N.m)');
%legend('Taul','Tau2');
title('Señal de control')
grid on

```

Función difusol.m

```
function [ui]=difusol(e,de,cep,cev,ep,ev,reglas,MD,MI,MDe,MIE)
```

```

e_cont=0;% almacena número de veces que se activan las funciones
de_cont=0;
base=0.25;% base del triangulo de las FM de salida para el caso del
defuzificador por promedio de áreas.
num=0; %numerador
den=0; %denominador
%% etapa de fuzificación
%busca el grado de membresía de acuerdo a la entrada de la variable lingüística
error de posición
if e<=cep(1)
    mfe=[1 zeros(1,ep-1)];%asegura el máximo si la entrada es mayor a la FM ENG
    e_cont=e_cont+1;
    e_int=1;%almacena posición de FM activa
elseif e>=cep(ev)
    mfe=[zeros(1,ep-1) 1];%asegura el máximo si la entrada es mayor a la FM EPG
    e_cont=e_cont+1;
    e_int=ep;
else
    for i=1:ep
        if e<=cep(i)
            mfe(i)=max([0 1+(e-cep(i))/MI(i)]);%Calcula grado de membresía a la
izquierda de las FM
            if mfe(i)~=0
                e_cont=e_cont+1;
                e_int=i;
            end
        else
            mfe(i)=max([0,1+(cep(i)-e)/MD(i)]);%Calcula grado de membresía a la
derecha de las FM
            if mfe(i)~=0
                e_cont=e_cont+1;
                e_int=i;
            end
        end
    end
end
%para la variable linguistica error de velocidad (ev)
if de<=cev(1)
    mfc=[1 zeros(1,ev-1)];%asegura el maximo si la entrada es mayor a la FM NGDE
    de_cont=de_cont+1;
    de_int=1;
elseif de>=cev(ev)%asegura el maximo si la entrada es mayor a la FM DEPG
    mfc=[zeros(1,ev-1) 1];
    de_cont=de_cont+1;
    de_int=ev;
else
    for i=1:ev
        if de<cev(i)
            mfc(i)=max([0,1+(de-cev(i))/MIe(i)]);%Calcula grado de membresía a
la izquierda de las FM
            if mfc(i)~=0
                de_cont=de_cont+1;
                de_int=i;
            end
        else
            mfc(i)=max([0,1+(cev(i)-de)/MDe(i)]);%Calcula grado de membresía a
la derecha de las FM
            if mfc(i)~=0
                de_cont=de_cont+1;
                de_int=i;
            end
        end
    end
end
end

```

```

end
%inferencia del mínimo y etapa de defuzificación.
i=1;
for m=(e_int-e_cont+1):e_int
    for n=(de_int-de_cont+1):de_int
        prem=min([mfe(m) mfc(n)]);%calcula el mínimo de los grados de membresía

        %método de defuzificación por promedio de áreas
        %area=base*(prem-prem^2/2)
        %num=num+reglas(n,m)*area*prem;
        %den=den+area*prem;

        %método de defuzificación por promedio de centros
        num=num+reglas(n,m)*prem;
        den=den+prem;
    end
end
ui=num/den;

```

C.2 Código en Builder C++ para el controlador difuso

A continuación se presenta el código implementado en C++ del algoritmo de control difuso.

```

float difuso(float qtil, float qptil, float cep[6], float cev[6], int ep, int ev, float reglas[6][6], float MD[6], float
MI[6], float MDe[6], float MLe[6])
{
    int e_cont=0,e_int=0; //registros para almacenar FM activas
    int de_cont=0,de_int=0;
//numerador y denominador del defuzificador
    float num=0;
    float den=0;
//variables para almacenar grado de membresía
    float mfe[6]={0.0};
    float mfc[6]={0.0};
    float prem=0.0;//guarda el resultado de la inferencia
    ui=0.0;//salida del defuzificador
    float base=0.25 //base del triangulo para la defuzificación.

//----- fusificación-----
//para la variable lingüística error de posición

    if (qtil<=cep[1]) //asegura máximo a la izquierda de la variable error
    {
        for(j=2;j<6;j++)
            {mfe[j]=0.0;}
        mfe[1]=1.0;
        e_cont=e_cont+1;
        e_int=1;
    }
    Else //asegura máximo a la derecha de la variable error
    {
        if (qtil>=cep[ep])
            {

```



```

if (qptil<cev[i])
{
mfc[i]=max(0.0,1+(qptil-cev[i])*MLe[i]); //calcula grado de membresía a la izquierda de la FM
if (mfc[i]!=0)
{
de_cont=de_cont+1;
de_int=i;
}
}
else
{
mfc[i]=max(0.0,1+(cev[i]-qptil)*MDe[i]); //calcula grado de membresía a la derecha de la FM
if (mfc[i]!=0)
{
de_cont=de_cont+1;
de_int=i;
}
}
}
}

//-----inferencia del mínimo y defuzificación-----
fun=e_int-e_cont+1;
fund=de_int-de_cont+1;
for(m=fun;m<=e_int;m++)
{
for (n=fund;n<=de_int;n++)
{

premi=min(mfe[m],mfc[n]); //valor mínimo de las entradas

//método de defuzificación por promedio de centros.
num=num+(reglas[n][m]*premi);
den=den+premi;

//método de defuzificación por promedio de áreas
//area=base*(premi-premi^2/2)
//num=num+reglas(n,m)*area*premi;
//den=den+area*premi;
}
}

ui=num/den;

return ui;

```

Apéndice D Plataforma de experimentación

D.1 Descripción General del sistema

El sistema a controlar es un robot manipulador de dos grados de libertad, tipo serial. El material del que está construido es de aluminio, la figura D.1 muestra al robot utilizado en las pruebas experimentales presentadas en esta tesis.

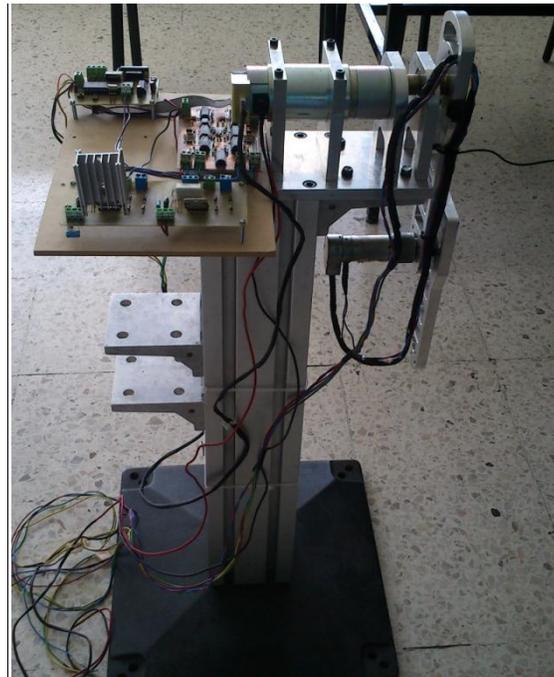


Figura D.1 Robot manipulador de 2 g.d.l.

En la tabla D.1 se presentan los parámetros calculados del robot manipulador. Los actuadores utilizados son dos motores de cd con encoder de la marca Pittman . Para el eslabón 1 el motor utilizado es el GM14904S014 (motor 1) y para el eslabón 2 es el GM8724S015 (motor 2), de 12 v. La tabla D.2 muestra las especificaciones técnicas del motor 1, tomadas de la hoja de datos proporcionada por el fabricante. Las especificaciones técnicas del motor 2 se muestran en la tabla D.3.

Tabla D.1 Parámetros calculados del robot manipulador

Parámetros	Valor
l_1 (longitud eslabón 1)	0.193 m
l_2 (longitud eslabón 2)	0.145 m
m_1 (masa eslabón 1)	0.763 kg
m_2 (longitud eslabón 2)	0.343 kg
l_{c1} (longitud del centro de masa eslabón 1)	0.136 m
l_{c2} (longitud del centro de masa eslabón 2)	0.048 m
I_1 (Inercia del eslabón 1)	0.0186
I_2 (Inercia del eslabón 2)	0.0026987
Constante g (gravedad)	9.81

Tabla D.2 Especificaciones técnicas GM14904S014
Fuente: Pittman® 2001 Express datasheet

Assembly Data	Symbol	Units	Value
Reference Voltage	E	V	12
No-Load Speed	S_{NL}	rpm (rad/s)	179 (18.7)
Continuous Torque (Max.) ¹	T_C	oz-in (N-m)	374 (2.6)
Peak Torque (Stall) ²	T_{PK}	oz-in (N-m)	2934 (20.7)
Weight	W_M	oz (g)	44.7 (1268)
Motor Data			
Torque Constant	K_T	oz-in/A (N-m/A)	4.33 (3.06E-02)
Back-EMF Constant	K_E	V/krpm (V/rad/s)	3.21 (3.06E-02)
Resistance	R_T	Ω	0.27
Inductance	L	mH	0.40
No-Load Current	I_{NL}	A	0.52
Peak Current (Stall) ²	I_P	A	44.4
Motor Constant	K_M	oz-in/ \sqrt{W} (N-m/ \sqrt{W})	8.63 (6.09E-02)
Friction Torque	T_F	oz-in (N-m)	1.6 (1.1E-02)
Rotor Inertia	J_M	oz-in-s ² (kg-m ²)	3.7E-03 (2.6E-05)
Electrical Time Constant	τ_E	ms	1.58
Mechanical Time Constant	τ_M	ms	7.0
Viscous Damping	D	oz-in/krpm (N-m-s)	0.18 (1.2E-05)
Damping Constant	K_D	oz-in/krpm (N-m-s)	55 (3.7E-03)
Maximum Winding Temperature	θ_{MAX}	°F (°C)	311 (155)
Thermal Impedance	R_{TH}	°F/watt (°C/watt)	45.9 (7.7)
Thermal Time Constant	τ_{TH}	min	28.8
Gearbox Data			
Reduction Ratio			19.7
Efficiency ³			0.84
Maximum Allowable Torque		oz-in (N-m)	500 (3.53)
Encoder Data			
Channels			3
Resolution		CPR	500
<small>1 - Specified at max. winding temperature at 25°C ambient without heat sink. 2 - Theoretical values supplied for reference only. 3 - Effective gearbox efficiency for this unit improved by use of ball bearings.</small>			

La instrumentación que permitió el control del robot manipulador se dividió en tres partes, adquisición y comunicación (1), conversión de señales (2) y etapa de potencia (3). El diagrama D.2 muestra las etapas citadas.

Tabla D.3 Especificaciones técnicas GM8724S015
Fuente: Pittman® 2001 Express datasheet

Assembly Data	Symbol	Units	Value
Reference Voltage	E	V	12
No-Load Speed	S_{NL}	rpm (rad/s)	230 (24.1)
Continuous Torque (Max.) ¹	T_C	oz-in (N-m)	42 (2.9E-01)
Peak Torque (Stall) ²	T_{PK}	oz-in (N-m)	117 (8.3E-01)
Weight	W_M	oz (g)	11.3 (320)
Motor Data			
Torque Constant	K_T	oz-in/A (N-m/A)	3.09 (2.18E-02)
Back-EMF Constant	K_E	V/krpm (V/rad/s)	2.29 (2.18E-02)
Resistance	R_T	Ω	4.33
Inductance	L	mH	2.34
No-Load Current	I_{NL}	A	0.18
Peak Current (Stall) ²	I_P	A	2.77
Motor Constant	K_M	oz-in/ \sqrt{W} (N-m/ \sqrt{W})	1.49 (1.05E-02)
Friction Torque	T_F	oz-in (N-m)	0.35 (2.5E-03)
Rotor Inertia	J_M	oz-in-s ² (kg-m ²)	2.3E-04 (1.6E-06)
Electrical Time Constant	τ_E	ms	0.54
Mechanical Time Constant	τ_M	ms	14.7
Viscous Damping	D	oz-in/krpm (N-m-s)	0.020 (1.4E-06)
Damping Constant	K_D	oz-in/krpm (N-m-s)	1.6 (1.1E-04)
Maximum Winding Temperature	θ_{MAX}	°F (°C)	311 (155)
Thermal Impedance	R_{TH}	°F/watt (°C/watt)	70.5 (21.4)
Thermal Time Constant	τ_{TH}	min	10.7
Gearbox Data			
Reduction Ratio			19.5
Efficiency ³			0.87
Maximum Allowable Torque		oz-in (N-m)	175 (1.24)
Encoder Data			
Channels			3
Resolution		CPR	500

1 - Specified at max. winding temperature at 25°C ambient without heat sink. 2 - Theoretical values supplied for reference only.
3 - Effective gearbox efficiency for this unit improved by use of ball bearings.

El algoritmo de control se desarrolló en una PC en el programa Builder® C++ el cual además de permitir desarrollar la programación del algoritmo de control, también sirve como interfaz de usuario, el código del programa utilizado se muestra en el apéndice C. La etapa 1 está compuesta por el microcontrolador y la PC, el microcontrolador utilizado es un PIC16F877A de Microchip® el cual tiene 5 puertos de entrada/salida. El microcontrolador lee los datos de los encoders por el puerto B, mediante interrupciones y los envía a la PC, la señal de control es recibida por el microcontrolador, proveniente de la PC. Dado que los

niveles de voltaje difieren entre las señales PIC-PC, se utilizó un CI MAX 232 que se encarga de nivelarlos. La figura D.3 muestra a la etapa 1.

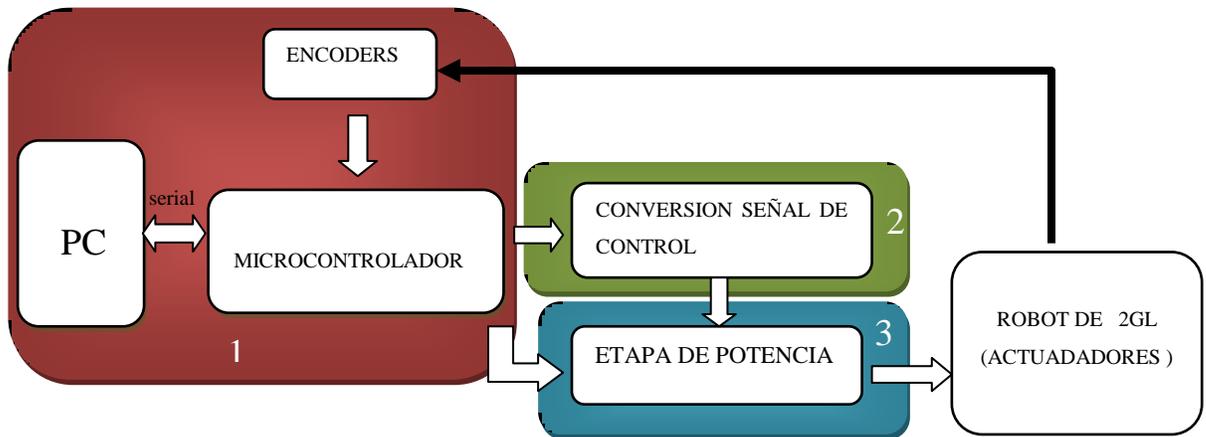


Figura D.2 Esquema general para el control del robot

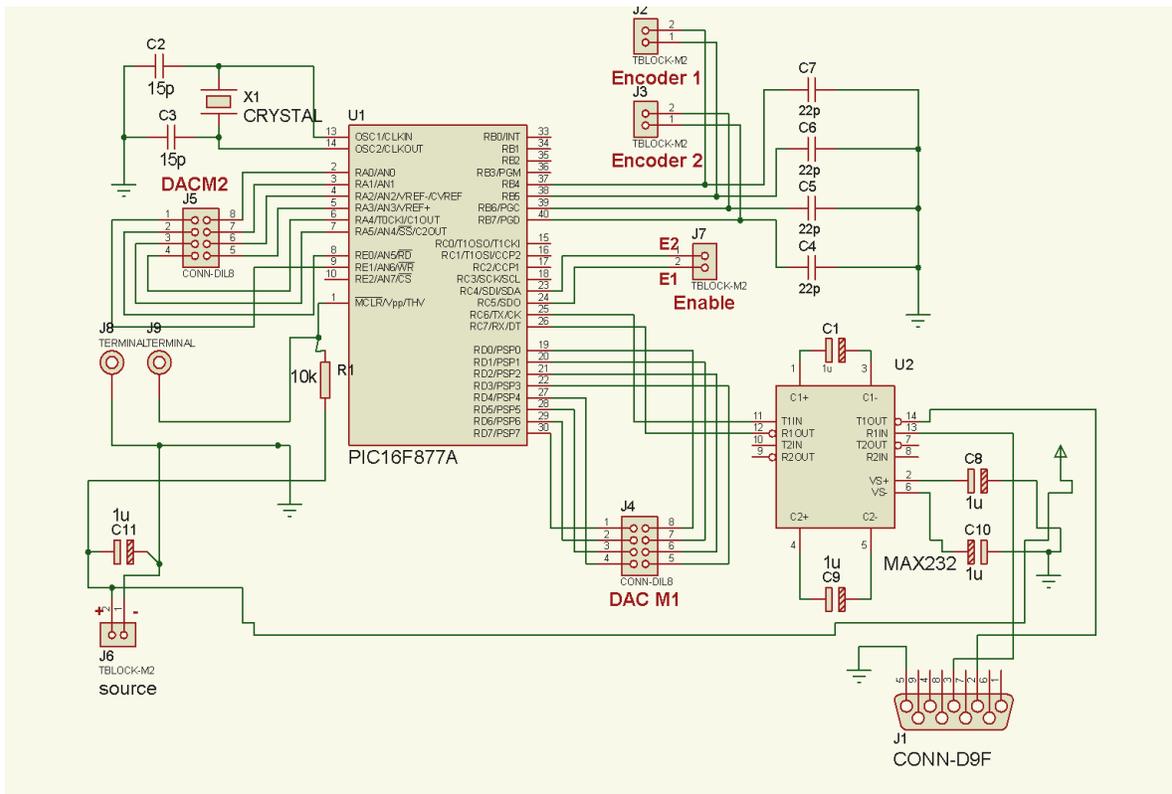


Figura D.3 Etapa 1

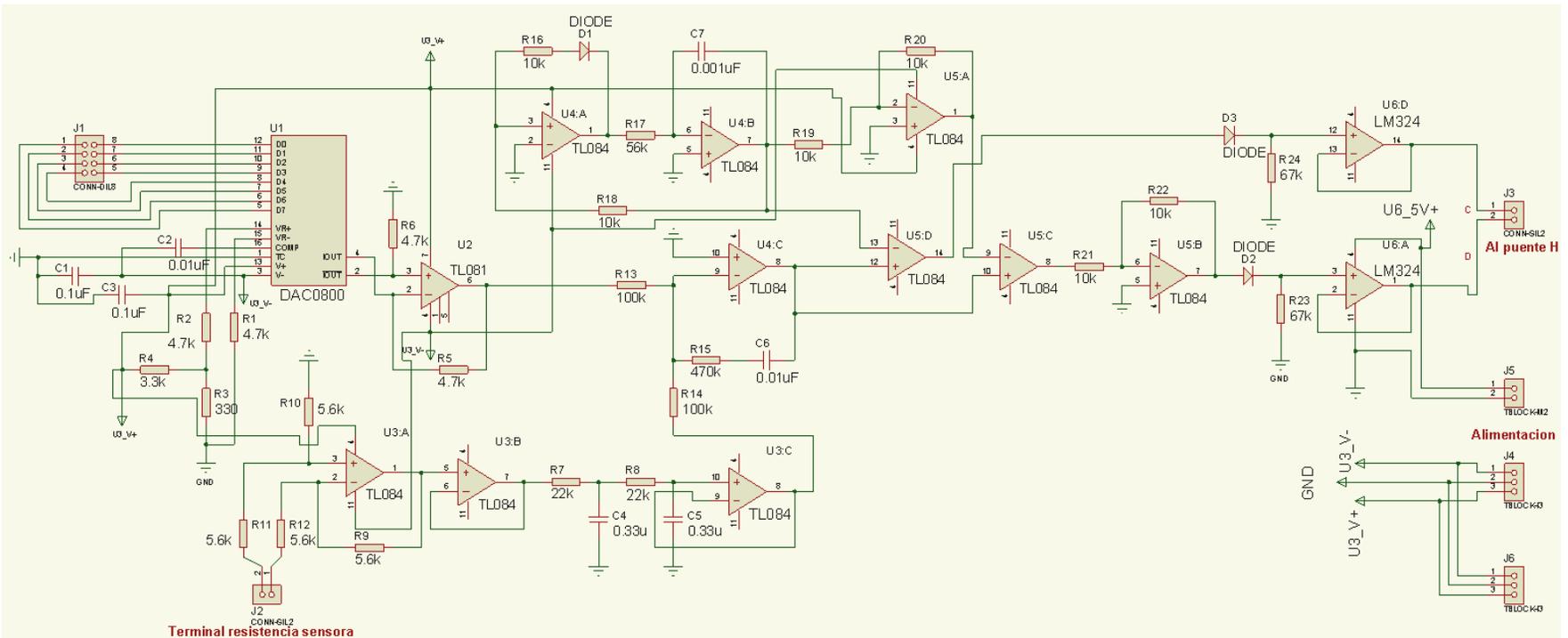


Figura D.4 Etapa 2 Conversión de señales

Para manipular los datos recibidos (señal de control) por el microcontrolador de la PC, se utilizaron dos convertidores digitales analógicos DAC0800, uno para cada motor, los puertos del PIC utilizados fueron el puerto A, D y parte del E. La señal obtenida a la salida del DAC era de -1 a 1 V, esta señal era comparada con la señal obtenida del motor mediante una resistencia sensora colocada en serie, para realizar esto se acondiciono la señal con amplificadores operacionales. El resultado de la comparación, era procesada por otros amplificadores que se encargaban de convertirla en una seña PWM . La figura C.4 representa el circuito eléctrico descrito.

La etapa 3 esta compuesta por el puente H L298N.

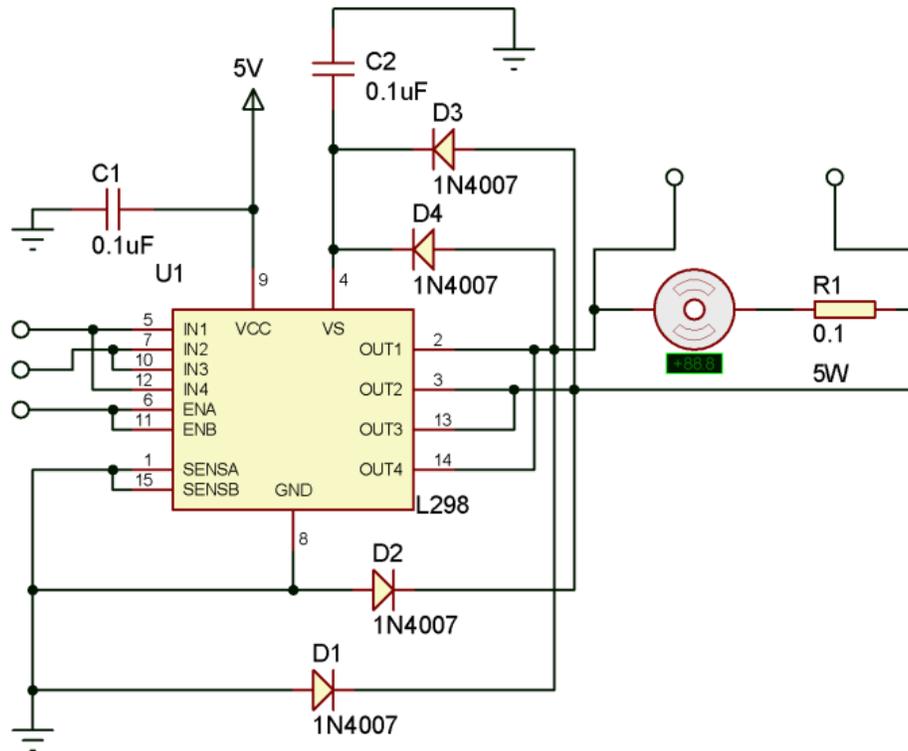


Figura D.5 Etapa 3 potencia

Éste se encarga de la potencia, el cual es controlado por los bloques explicados anteriormente, el puente H amplifica la corriente de alimentación del motor para realizar el movimiento. Este dispositivo tiene la capacidad de invertir el movimiento del motor, ya que puede cambiar el sentido de la corriente que pasa por el motor, mediante las señales D y C. La señal de enable, E, es obtenida por software. En los circuitos anteriores se crea un modulador de ancho de pulsos, el cual ayuda a controlar al puente-H.