

BIBLIOTECA — CENTRO DE
GRADUADOS E INVESTIGACION
A. T. L.

DIVISIÓN DE ESTUDIOS DE POSGRADO E INVESTIGACIÓN

**“ Control Descentralizado Difuso Adaptable:
Aplicación al Robot Mitsubishi PA10-7CE ”**

POR

Ing. José Alejandro Flores Avila

TESIS

**PRESENTADA COMO REQUISITO PARCIAL PARA OBTENER EL
GRADO DE MAESTRO EN CIENCIAS EN INGENIERÍA ELÉCTRICA**

DIRECTOR DE TESIS

Dr. Miguel Ángel Llamas Leal

CODIRECTOR DE TESIS

Dr. Ramón García Hernández

ISSN: 0188-9060



RIITEC: (10)-TMCIE-2014

Torreón, Coahuila, México
Julio, 2014

SEP

SECRETARÍA DE
EDUCACIÓN PÚBLICA



DIRECCIÓN GENERAL DE EDUCACIÓN SUPERIOR TECNOLÓGICA
Instituto Tecnológico de la Laguna



2014, Año de Octavio Paz

Torreón, Coah. **03/Julio/2014**
Dependencia: DEPI/CPCIE
Oficio: DEPI/CPCIE/062/2014
Asunto: Autorización de
impresión de tesis.

C. JOSÉ ALEJANDRO FLORES ÁVILA
CANDIDATO AL GRADO DE MAESTRO EN CIENCIAS EN INGENIERÍA ELÉCTRICA.
PRESENTE

Después de haber sometido a revisión su trabajo de tesis titulado:

**"CONTROL DESCENTRALIZADO DIFUSO ADAPTABLE:
APLICACIÓN AL ROBOT MITSUBISHI PA10-7CE"**

Habiendo cumplido con todas las indicaciones que el jurado revisor de tesis hizo, se le comunica que se le concede la autorización con número de registro **RIITEC: (10)-TMCIE-2014**, para que proceda a la impresión del mismo.

ATENTAMENTE
EDUCACIÓN TECNOLÓGICA FUENTE DE INNOVACIÓN

DR. JOSÉ LUIS MEZA MEDINA
Jefe de la División de Estudios de Posgrado e Investigación
del Instituto Tecnológico de la Laguna

J.L.M.M. / J.M.M.
J.L.M.M. / J.M.M.



SECRETARÍA DE
EDUCACIÓN PÚBLICA
INSTITUTO TECNOLÓGICO
de la Laguna
División de Estudios de Posgrado
e Investigación



SEP

SERVICIOS DE
EDUCACIÓN SUPERIOR



DIRECCIÓN GENERAL DE EDUCACIÓN SUPERIOR TECNOLÓGICA
Instituto Tecnológico de la Laguna



2014, Año de Octavio Paz

Torreón, Coah., 03/Julio/2014

DR. JOSE LUIS MEZA MEDINA
JEFE DE LA DIVISIÓN DE ESTUDIOS DE POSGRADO E INVESTIGACIÓN

Por medio de la presente, hacemos de su conocimiento que después de haber sometido a revisión el trabajo de tesis titulado:

**"CONTROL DESCENTRALIZADO DIFUSO ADAPTABLE:
APLICACIÓN AL ROBOT MITSUBISHI PA10-7CE"**

Desarrollado por el **C. José Alejandro Flores Ávila**, con número de control **M1213028** y habiendo cumplido con todas las correcciones que se le indicaron, estamos de acuerdo que se le conceda la autorización de la fecha de examen de grado para que proceda a la impresión de la misma.

ATENTAMENTE
EDUCACIÓN TECNOLÓGICA FUENTE DE INNOVACIÓN

Dr. Miguel A. Llama Leal
Asesor/Director de Tesis

Dr. Ramón García Hernández
Coasesor de Tesis

Dr. Víctor A. Santibáñez Dávila
Comité Tutorial

Dr. Francisco Jurado Zamarripa
Comité Tutorial



Dedicatorias

A mi padre, José Flores Basurto ejemplo de hombre honrado, trabajador e incansable, un padre admirable que desde siempre ha estado a mi lado apoyándome y entregando todo de sí para hacerme crecer.

A mi madre, Yolanda Ávila Villagraza, la mujer más maravillosa que pudo haberme educado, modelo de madre amorosa y amada, mujer inteligente y capaz, que al igual que mi padre, me ha dado todo en esta vida.

Gracias a ambos por su perseverancia, su constancia y trabajo; sin ustedes nada de esto sería posible, es un privilegio ser su hijo.

A mis hermanos Ricaro Arturo y Lizeth Flores Ávila, por compartir su vida a mi lado y acompañarme siempre, por ser motivación y razón de gran orgullo, por su apoyo incondicional.

A Blanca Yudiria Arellano López, por ser sinónimo de amor y lealtad, por su compromiso y su comprensión, y por permitirme el honor de estar a su lado. You will never walk alone.

Agradecimientos

A Dios, por todo lo que me ha dado en esta vida.

Al Dr. Miguel Ángel Llama Leal, por su asesoría, sus contribuciones y su entera disposición, por brindarme la confianza, el apoyo y la dirección necesaria para lograr satisfactoriamente esta meta.

Al Dr. Víctor Adrián Santibáñez Dávila, por sus colaboraciones, su importante ayuda y su actitud siempre comprometida al desarrollo de las nuevas generaciones de investigadores.

Al Dr. Ricardo Emmanuel Campa Cocóm, por ir más allá del deber, porque sin ser su responsabilidad ha aportado su tiempo, conocimiento y entrega a este trabajo de investigación.

Al Dr. Francisco Jurado Zamuripa, por sus palabras de aliento, su motivación y por entregar en el aula, conocimiento que hoy forma parte importante de este proyecto.

Al Instituto Tecnológico de la Laguna y CONACYT por permitirme continuar con mis estudios a través de la entrega de programas de posgrado y apoyos para la investigación en México respectivamente.

Control Descentralizado Difuso Adaptable: Aplicación al Robot Mitsubishi PA10-7CE

José Alejandro Flores Ávila

Resumen

En este trabajo de tesis se propone un algoritmo de control que emplea sistemas difusos adaptables para aproximar cierta porción de la dinámica de robots manipuladores a fin de resolver el problema de seguimiento de trayectorias. Dicho esquema se caracteriza por no requerir del conocimiento del modelo dinámico, y en contraste con algunos controladores difuso adaptables previamente desarrollados, el aquí propuesto se encuentra en una configuración descentralizada, en la que cada articulación es considerada como un subsistema y es controlada de manera independiente a partir de sus variables locales. Además, se presenta el estudio realizado para garantizar la estabilidad y acotamiento de soluciones del sistema en lazo cerrado vía teoría de Lyapunov y se incluye un análisis funcional para demostrar que se satisface el objetivo de control de movimiento. La teoría aquí expuesta es verificada en simulación aplicando el algoritmo diseñado sobre el robot Mitsubishi PA10-7CE y se reportan los resultados.

Decentralized Adaptive Fuzzy Control: Application to the Mitsubishi PA10-7CE Robot

José Alejandro Flores Ávila

Abstract

In this thesis a control algorithm that uses fuzzy adaptive systems is proposed to approximate certain portion of the manipulator robots dynamics in order to solve the trajectory tracking problem. This scheme is characterized by not requiring of the dynamic model knowledge, and in contrast to some fuzzy adaptive controllers previously developed, the one proposed here is on a decentralized configuration, wherein each joint is considered as a subsystem and it is independently controlled only through its local variables. Furthermore, a carried out study to guarantee the stability and the boundedness of solution of the closed loop system via Lyapunov theory is presented, and a functional analysis is included to prove that the objective of motion control is satisfied. The theory exposed here is verified via simulation by applying the designed algorithm to the Mitsubishi PA10-7CE robot and the results are reported.

Índice general

1	Introducción	1
1.1	Objetivo general	3
1.1.1	Objetivos específicos	3
1.2	Motivación	3
1.3	Planteamiento del problema	4
1.4	Organización del documento	5
2	Marco teórico	6
2.1	Preliminares matemáticos	6
2.1.1	Vectores	6
2.1.2	Matrices	8
2.1.3	Espacios funcionales	12
2.1.4	Funciones específicas	13
2.2	Sistemas dinámicos	13
2.2.1	Sistemas lineales	14
2.2.2	Sistemas no lineales	16
2.3	Estabilidad en el sentido de Lyapunov	16
2.3.1	Acotamiento de soluciones	18
2.4	Modelo dinámico de robots manipuladores	18
2.4.1	Propiedades	19
2.5	Control de robots manipuladores	20
2.5.1	Especificaciones de control	21
2.6	Control adaptable	22
3	Lógica difusa	24
3.1	Conjuntos difusos	25
3.1.1	Funciones de membresía	25

3.2	Variable lingüística	27
3.3	Fusificación	28
3.4	Reglas difusas	29
3.4.1	Base de reglas difusas	29
3.5	Inferencia difusa	30
3.5.1	Inferencia producto	31
3.6	Defusificación	32
3.6.1	Promedio de centros	33
3.7	Sistema difuso	33
3.8	Aproximador universal	34
3.9	Sistemas de control basados en lógica difusa	35
3.9.1	Características	35
3.9.2	Estructura	36
4	Control descentralizado difuso adaptable	38
4.1	Sistema descentralizado	38
4.2	Sistema difuso adaptable	42
4.3	Diseño y estructura del controlador	44
4.3.1	Ecuación de lazo cerrado	45
4.3.2	Análisis de estabilidad	46
5	Aplicación al robot Mitsubishi PA10-7CE	51
5.1	Antecedentes y estudio del robot	51
5.1.1	Nivel 1: Mecanismo	52
5.1.2	Nivel 2: Controlador de servos	53
5.1.3	Nivel 3: Control de movimiento	54
5.1.4	Nivel 4: Control de operación	56
5.2	Resultados en simulación	56
5.2.1	Parámetros de diseño	57
5.2.2	Resultados obtenidos	58
5.3	Interfaz gráfica de control	63
5.3.1	Operación de la interfaz de control	65
5.3.2	Errores comunes	66
6	Conclusiones y trabajo futuro	68
6.1	Conclusiones	68

6.2 Trabajo futuro	69
APÉNDICE	70
A Parámetros del robot Mitsubishi PA10-7CE	70
B Datos técnicos de la interfaz gráfica de usuario	71
B.1 Generación de nuevos controladores	74
Referencias	76

Índice de figuras

3.1	Función de pertenencia pseudo-trapezoidal F .	26
3.2	Función de pertenencia pseudo-trapezoidal L .	26
3.3	Función de pertenencia triangular Λ .	27
3.4	Función de pertenencia impulsiva δ .	27
3.5	Variable lingüística.	28
3.6	Proceso de fusificación <i>singleton</i> .	29
3.7	Método de inferencia producto usando fusificador <i>singleton</i> .	32
3.8	Combinación de consecuentes.	32
3.9	Diagrama a bloques de la estructura general de un sistema difuso.	37
4.1	Diagrama a bloques del controlador propuesto.	44
5.1	Niveles funcionales actuales del Mitsubishi PA10-7CE.	52
5.2	Parte mecánica del PA10-7CE.	53
5.3	Caja de control de los servomotores del robot.	54
5.4	Modificación de la tarjeta ARCNET PCI20U-4000.	54
5.5	Circuito del transceptor óptico.	55
5.6	Conexión entre la tarjeta ARCNET y el transceptor óptico.	55
5.7	Trayectorias deseadas de movimiento.	57
5.8	Funciones de membresía de entrada.	58
5.9	Gráficas de las posiciones actuales y deseadas de cada articulación.	59
5.10	Gráficas de los pares aplicados a cada articulación.	60
5.11	Errores de posición obtenidos con los controladores difuso adaptable descentralizado (DFAC) y PID.	62
5.12	Interfaz gráfica del usuario para el manejo del PA10-7CE.	63
B.1	Bloque PA10 Control.	71
B.2	Parámetros del bloque PA10 Control.	73
B.3	Bloques para la generación de nuevos controladores.	74

B.4 Diagrama a bloques de Simulink para su manejo con la interfaz gráfica.	75
--	----

Índice de tablas

3.1	Tabla de consulta de reglas difusas	30
5.1	Parámetros de diseño.	57
5.2	Ganancias del controlador PID.	61
A.1	Parámetros del robot Mitsubishi PA10-7CE.	70

Introducción

La ingeniería de control, la robótica y la mecatrónica son áreas interdisciplinarias que a menudo interactúan entre sí y que poseen una identidad propia; la ingeniería de control por ejemplo, se basa en el modelado matemático de sistemas y el análisis de su dinámica a fin de producir un comportamiento deseado del sistema; la robótica por su parte es la ciencia que estudia los robots, su diseño, manufactura, y aplicación, mientras que la mecatrónica abarca la combinación sinérgica de las ingenierías mecánica, eléctrica, electrónica de control y de sistemas computacionales.

Uno de los resultados que emerge de la interacción de estas tres áreas es lo que se conoce como *control de robots*, que en términos generales consiste en hacer que un manipulador lleve a cabo una determinada tarea. No obstante, el control de un brazo robótico de varios eslabones resulta un problema difícil y desafiante debido a las altas no linealidades y a la complejidad de su dinámica. En particular, para el caso típico en que un manipulador se desplaza libremente dentro de su espacio de trabajo, sin que tenga una interacción con el entorno, la posición y velocidad articular son las salidas del robot y el objetivo de control asociado al seguimiento de trayectorias recibe el nombre de control de movimiento e involucra, entre otras, una convergencia del error a cero.

En años recientes, diversos esfuerzos han sido desarrollados en el diseño de controladores para robots manipuladores basados en metodologías de linealización por realimentación, control adaptable y control robusto, con un enfoque en el conocimiento del modelo dinámico o su estructura. Algunos de estos estudios están reportados en textos como [4], [8], [11], [19] y [22].

Por otra parte, una clase de esquema para el control de movimiento de robots que ha recibido una creciente atención e importancia son los basados en lógica difusa (*e.g.* ver [10], [13]). Una de las ventajas del control difuso es que la acción de control requerida para cada articulación es determinada sin tener necesariamente que basarse en un modelo matemático y se ha buscado mejorar estos controladores con la incorporación de otras técnicas como el control difuso adaptable, que es un esquema no lineal que utiliza sistemas difusos con parámetros ajustables como aproximadores de funciones, que a su vez, son utilizados para calcular la ley de control utilizando un mínimo conocimiento de la planta; *e.g.* en [16] se aborda el problema de seguimiento de trayectoria mediante un control difuso adaptable, identificando la dinámica del robot a través de sistemas difusos, obteniendo un buen desempeño; sin embargo este controlador como muchos otros tiene una estructura de control *centralizado*, por lo que cuando el número de grados de libertad crece, la complejidad de los sistemas difusos crece exponencialmente, por lo que su diseño e implementación pueden ser tareas bastante difíciles.

Un enfoque alternativo es el control *descentralizado*, en el que el robot se concibe como un conjunto de subsistemas interconectados, así cada articulación se toma como un subsistema con la finalidad de desarrollar controladores independientes, los cuales sólo consideran mediciones de las variables locales de cada articulación, es decir, la posición y velocidad angular, y se compensan los efectos de interconexión como inercias y las fuerzas de Coriolis, que se suponen como perturbaciones; algunos ejemplos de esto son abordados en [5] y [6] empleando redes neuronales.

En la literatura previa, se han desarrollado trabajos sobre el control descentralizado difuso adaptable con importantes logros en el análisis de estabilidad y en el desempeño de los controladores, *e.g.* en [13] y [25] la estabilidad y acotamiento de soluciones para el seguimiento de trayectorias es presentado, sin embargo no se cumple con el objetivo de control de movimiento.

En este trabajo se propone un controlador descentralizado difuso adaptable para el problema de seguimiento, y combinando conceptos de robótica, lógica difusa, control no lineal y control adaptable, se prueba la estabilidad y acotamiento de soluciones del sistema total vía un diseño por síntesis de Lyapunov. Además, haciendo uso del lema de Barbalat se concluye que el objetivo de control de movimiento de robots manipuladores se satisface. Los resultados teóricos obtenidos son validados en simulación considerando el modelo dinámico del robot Mitsubishi PA10-7CE tomado de [17], [18] y se evalúa el desempeño del algoritmo de control.

1.1. Objetivo general

El desarrollo de este proyecto de tesis tiene como principal objetivo proponer un esquema de control descentralizado difuso adaptable que permita resolver el problema de seguimiento de trayectorias en robots manipuladores.

1.1.1. Objetivos específicos

- Realizar una revisión del estado del arte a fin de tener una visión global del área de investigación a abordar.
- Diseñar un controlador descentralizado difuso adaptable y aplicarlo al robot Mitsubishi PA10-7CE para la validación de esta propuesta.
- Llevar a cabo el análisis que garantiza la estabilidad del esquema de control propuesto y el cumplimiento del objetivo de control de movimiento de robots manipuladores.

1.2. Motivación

La robótica y la mecatrónica representan en la actualidad áreas estratégicas y claves para todo país que aspire a la modernidad, el desarrollo constante de dispositivos mecánicos, electrónicos, mecatrónicos y electromecánicos tanto de uso doméstico como de uso industrial, se ha ido incrementando con el paso de los años, dichos dispositivos, como pueden ser robots o máquinas programables, requieren de controladores para su óptimo funcionamiento. El desarrollo científico no solo ha ido creciendo en el número de dispositivos que salen al mercado cada día, sino que también se ha incrementado en su complejidad de diseño, esto da lugar al constante desarrollo de nuevas técnicas de control inteligente debido a que en algunos casos, la teoría de control clásico no es suficiente para obtener resultados adecuados en el control de estos dispositivos.

Por otra parte, el uso de robots manipuladores se ha generalizado en muchas aplicaciones que requieren de movimientos repetitivos y relativamente sencillos, mientras que las capacidades de un robot manipulador están determinadas entre otras, por la precisión de movimiento, el grado de manipulación del robot, la sofisticación y confiabilidad del controlador. Lo anterior aunado al hecho de que los esquemas descentralizados para

el control de estos sistemas no lineales interconectados no han recibido una amplia atención, como en el caso de esquemas centralizados, motivan al diseño de controladores descentralizados difusos adaptables utilizando sólo la información local y garantizando la estabilidad del sistema completo.

1.3. Planteamiento del problema

Como se explica en [17], el robot Mitsubishi PA10 es un robot manipulador industrial que cambia por completo la visión de robots industriales convencionales. Su nombre es acrónimo de "Portable General-Purpose Intelligent Arm" (Brazo Inteligente Portátil de Propósito General). Existen dos versiones: el PA10-6C y el PA10-7C, donde el dígito del sufijo indica el número de grados de libertad del brazo. Este trabajo se enfocará al estudio del modelo PA10-7CE que es la versión mejorada del PA10-7C.

El robot PA10 es un robot de arquitectura abierta; esto significa que posee:

- Una estructura jerárquica con varios niveles de control.
- Una comunicación entre niveles por medio de alguna interfaz estandarizada.
- Una interfaz abierta y de propósito general en el nivel más alto.

Este esquema permite al operador dedicarse exclusivamente a la programación de tareas en el nivel más alto del sistema PA10, sin tener que preocuparse por el funcionamiento de los niveles más bajos. Para la programación se utiliza un lenguaje de alto nivel como Visual BASIC o Visual C++, desde una computadora personal con el sistema operativo MS-Windows. El robot PA10, es actualmente el robot de arquitectura abierta más utilizado en el campo de la investigación.

El problema consiste en hacer que el extremo final del manipulador siga una trayectoria especificada en el espacio de trabajo del robot. El seguimiento debe ser lo más cercano posible. En algunos casos, uno de los problemas para la implementación de sistemas de control de movimiento de robots manipuladores radica en que el modelo del proceso a controlar no está disponible o es demasiado complicado para propósitos de diseño.

Con base en lo anterior, es importante aproximar el modelo del proceso y conservar las características esenciales para que combinado con la estrategia de control permita

resolver el problema de seguimiento de trayectorias. En el caso de este trabajo de tesis, se propone diseñar y estudiar un esquema de control descentralizado difuso adaptable para resolver dicho problema.

1.4. Organización del documento

A continuación se describe cada uno de los capítulos que componen este trabajo de tesis. En el Capítulo 2 se exponen los fundamentos teóricos en los cuales está basado el estudio y análisis del controlador propuesto, tales como herramientas matemáticas, teoremas de estabilidad y tópicos selectos de control de robots. En el Capítulo 3 se tratan conceptos básicos sobre lógica difusa, definiciones, módulos de los que se compone un sistema difuso y su propiedad como aproximadores universales. El diseño y la estructura del controlador descentralizado difuso adaptable son abordados en el Capítulo 4, así como el análisis de estabilidad que conlleva a demostrar que se satisface el objetivo de control de movimiento de robots manipuladores. Posteriormente, en el Capítulo 5 se reportan los resultados en simulación de la aplicación del controlador difuso al robot Mitsubishi PA10-7CE y se describe la interfaz de control que permite el manejo de este robot a través de MATLAB/Simulink. Luego, el Capítulo 6 es dedicado a las conclusiones obtenidas en este trabajo de tesis, así como los comentarios sobre el trabajo futuro a realizar. Por último se incluyen el apéndice y las referencias bibliográficas.

Marco teórico

En este capítulo se presentan las herramientas matemáticas básicas que serán empleadas a lo largo del documento. Básicamente se resumen algunos conceptos de álgebra lineal como vectores, matrices y espacio de funciones. Posteriormente se presentan descripciones de sistemas dinámicos lineales y no lineales, así como las definiciones y teoremas de estabilidad en el sentido de Lyapunov y acotamiento de soluciones; luego se describe el modelo dinámico de robots manipuladores y algunas de sus propiedades. Finalmente se brindan algunas definiciones relevantes de control de robots y control adaptable.

2.1. Preliminares matemáticos

2.1.1. Vectores

El conjunto de los *números reales* [8] es denotado por el símbolo \mathbb{R} mientras que el espacio euclidiano real de dimensión n será denotado por \mathbb{R}^n , es decir el conjunto de todos los vectores \boldsymbol{x} de dimensión n formados por un arreglo de n números reales en forma de columna

$$\boldsymbol{x} = \begin{bmatrix} x_1 \\ \vdots \\ x_n \end{bmatrix} = \left[x_1 \quad \cdots \quad x_n \right]^T$$

donde $x_1, x_2, \dots, x_n \in \mathbb{R}$ son los elementos del vector \boldsymbol{x} y el superíndice T denota transposición.

Producto de vectores

El producto interno o producto punto de dos vectores $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$ se define como

$$\mathbf{x}^T \mathbf{y} = \begin{bmatrix} x_1 & \cdots & x_n \end{bmatrix} \begin{bmatrix} y_1 \\ \vdots \\ y_n \end{bmatrix} = \sum_{i=1}^n x_i y_i = x_1 y_1 + \cdots + x_n y_n$$

Se puede verificar que el producto punto cumple con las siguientes dos propiedades

- $\mathbf{x}^T \mathbf{y} = \mathbf{y}^T \mathbf{x}$, para todo $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$.
- $\mathbf{x}^T (\mathbf{y} + \mathbf{z}) = \mathbf{x}^T \mathbf{y} + \mathbf{x}^T \mathbf{z}$, para todo $\mathbf{x}, \mathbf{y}, \mathbf{z} \in \mathbb{R}^n$.

Norma euclidiana

La norma euclidiana $\|\mathbf{x}\|$ de un vector $\mathbf{x} \in \mathbb{R}^n$ está definida como

$$\|\mathbf{x}\| = \sqrt{\sum_{i=1}^n x_i^2} = \sqrt{\mathbf{x}^T \mathbf{x}}$$

donde sólo se considera la parte positiva de la raíz cuadrada y tiene las siguientes propiedades:

- $\|\mathbf{x}\| = 0$ si y sólo si $\mathbf{x} = \mathbf{0} \in \mathbb{R}^n$,
- $\|\mathbf{x}\| > 0$ para todo $\mathbf{x} \in \mathbb{R}^n$ con $\mathbf{x} \neq \mathbf{0} \in \mathbb{R}^n$,
- $\|\alpha \mathbf{x}\| = |\alpha| \|\mathbf{x}\|$ para todo $\alpha \in \mathbb{R}$ y $\mathbf{x} \in \mathbb{R}^n$,
- $\|\mathbf{x}\| - \|\mathbf{y}\| \leq \|\mathbf{x} + \mathbf{y}\| \leq \|\mathbf{x}\| + \|\mathbf{y}\|$ para todo $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$ (desigualdad del triángulo)
- $|\mathbf{x}^T \mathbf{y}| \leq \|\mathbf{x}\| \|\mathbf{y}\|$ para todo $\mathbf{x}, \mathbf{y} \in \mathbb{R}^n$ (desigualdad de Schwarz)

Normas p

Se considera la clase de *normas p* [9] definidas como

$$\|\mathbf{x}\|_p = (|x_1|^p + |x_2|^p + \cdots + |x_n|^p)^{1/p}, \quad \forall \mathbf{x} \in \mathbb{R}^n, \quad 1 \leq p \leq \infty$$

y

$$\|\mathbf{x}\|_{\infty} = \max_i |x_i|.$$

Las tres normas más comunes son $\|\mathbf{x}\|_1$, $\|\mathbf{x}\|_{\infty}$ y la norma euclidiana

$$\|\mathbf{x}\|_2 = (|x_1|^2 + |x_2|^2 + \dots + |x_n|^2)^{1/2} = (\mathbf{x}^T \mathbf{x})^{1/2}.$$

Todas las normas p son equivalentes, en el sentido de que si $\|\cdot\|_{\alpha}$ y $\|\cdot\|_{\beta}$ son dos normas p diferentes, entonces existen constantes positivas C_1 y C_2 tales que

$$C_1 \|\mathbf{x}\|_{\alpha} \leq \|\mathbf{x}\|_{\beta} \leq C_2 \|\mathbf{x}\|_{\alpha}, \quad \forall \mathbf{x} \in \mathbb{R}^n.$$

Para las tres normas más comunes, esta desigualdad toma la forma de

$$\|\mathbf{x}\|_2 \leq \|\mathbf{x}\|_1 \leq \sqrt{n} \|\mathbf{x}\|_2,$$

$$\|\mathbf{x}\|_{\infty} \leq \|\mathbf{x}\|_2 \leq \sqrt{n} \|\mathbf{x}\|_{\infty},$$

$$\|\mathbf{x}\|_{\infty} \leq \|\mathbf{x}\|_1 \leq n \|\mathbf{x}\|_{\infty}.$$

Un resultado importante que concierne a las normas p es la *desigualdad de Hölder*

$$\mathbf{x}^T \mathbf{y} \leq \|\mathbf{x}\|_p \|\mathbf{y}\|_q, \quad \text{con } \frac{1}{p} + \frac{1}{q} = 1, \quad \forall \mathbf{x}, \mathbf{y} \in \mathbb{R}^n.$$

2.1.2. Matrices

Se denota por $\mathbb{R}^{n \times m}$ al conjunto de matrices reales [8] \mathbf{A} de dimensión $n \times m$ formadas por un arreglo de números reales ordenados en n renglones y m columnas de la forma

$$\mathbf{A} = \{a_{ij}\} = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1m} \\ a_{21} & a_{22} & \dots & a_{2m} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n1} & a_{n2} & \dots & a_{nm} \end{bmatrix}$$

La matriz transpuesta $\mathbf{A}^T = \{a_{ji}\} \in \mathbb{R}^{m \times n}$ es obtenida al intercambiar los renglones por las columnas de $\mathbf{A} = \{a_{ij}\} \in \mathbb{R}^{n \times m}$. El producto de matrices satisface las siguientes propiedades:

- $(\mathbf{AB})^T = \mathbf{B}^T \mathbf{A}^T$ para toda $\mathbf{A} \in \mathbb{R}^{m \times p}$ y $\mathbf{B} \in \mathbb{R}^{p \times n}$,

- en general $AB \neq BA$,
- para toda $A \in \mathbb{R}^{m \times p}$, $B \in \mathbb{R}^{p \times n}$:

$$A(B + C) = AB + AC \quad \text{con } C \in \mathbb{R}^{p \times n},$$

$$ABC = A(BC) = (AB)C \quad \text{con } C \in \mathbb{R}^{n \times r}.$$

Matrices particulares

Una matriz A es *cuadrada* si $n = m$, i.e. si el número de renglones es igual al número de columnas. Una matriz cuadrada $A \in \mathbb{R}^{n \times n}$ es *simétrica* si es igual a su transpuesta, es decir, si $A = A^T$. A es *antisimétrica* si $A = -A^T$. La siguiente propiedad de matrices antisimétricas es de particular importancia en control de robots

$$x^T A x = 0, \quad \text{para todo } x \in \mathbb{R}^n.$$

Una matriz cuadrada $A = \{a_{ij}\} \in \mathbb{R}^{n \times n}$ es *diagonal* si $a_{ij} = 0$ para $i \neq j$. Una matriz diagonal A se denota por $A = \text{diag}\{a_{11}, a_{22}, \dots, a_{nn}\} \in \mathbb{R}^{n \times n}$. En el caso particular en que $a_{11} = a_{22} = \dots = a_{nn}$, la correspondiente matriz diagonal es denotada por $A = \text{diag}\{a\} \in \mathbb{R}^{n \times n}$. Dos matrices diagonales de particular importancia son la matriz *identidad* de dimensión n , denotada por $I = \text{diag}\{1\} \in \mathbb{R}^{n \times n}$, y la matriz *nula* de dimensión n definida como $0 = \text{diag}\{0\} \in \mathbb{R}^{n \times n}$.

Una matriz cuadrada $A \in \mathbb{R}^{n \times n}$ es *singular* si su determinante es cero, es decir, $\det[A] = 0$, en caso contrario se dice que A es *no singular*. La inversa de la matriz A , i.e. A^{-1} existe si y sólo si A es no singular.

Una matriz cuadrada $A \in \mathbb{R}^{n \times n}$ no necesariamente simétrica, es *definida positiva* si

$$x^T A x > 0, \quad \forall x \in \mathbb{R}^n, \quad \text{con } x \neq 0 \in \mathbb{R}^n.$$

Cualquier matriz simétrica definida positiva $A = A^T > 0$ es no singular. Además, $A = A^T > 0$ si y sólo si $A^{-1} = (A^{-1})^T > 0$. También se puede demostrar que la suma de dos matrices definidas positivas da como resultado una matriz definida positiva.

Una matriz $A \in \mathbb{R}^{n \times n}$ no necesariamente simétrica, es *semidefinida positiva* si

$$x^T A x \geq 0, \quad \forall x \in \mathbb{R}^n, \quad \text{con } x \neq 0 \in \mathbb{R}^n.$$

Una matriz $\mathbf{A} \in \mathbb{R}^{n \times n}$ es *definida negativa* si $-\mathbf{A}$ es definida positiva, y es *semidefinida negativa* si $-\mathbf{A}$ es semidefinida positiva.

Valores propios

Para cada matriz cuadrada $\mathbf{A} \in \mathbb{R}^{n \times n}$ existen n valores propios o eigenvalores denotados por $\lambda_1\{\mathbf{A}\}, \lambda_2\{\mathbf{A}\}, \dots, \lambda_n\{\mathbf{A}\}$. Los eigenvalores de la matriz $\mathbf{A} \in \mathbb{R}^{n \times n}$ satisfacen

$$\det[\lambda_i\{\mathbf{A}\}\mathbf{I} - \mathbf{A}] = 0, \quad \text{para } i = 1, 2, \dots, n$$

donde $\mathbf{I} \in \mathbb{R}^{n \times n}$ es la matriz identidad. Para el caso de matrices simétricas, sus valores propios son tales que:

- $\lambda_1\{\mathbf{A}\}, \lambda_2\{\mathbf{A}\}, \dots, \lambda_n\{\mathbf{A}\} \in \mathbb{R}$.
- expresando el eigenvalor más grande de \mathbf{A} como $\lambda_{\text{Max}}\{\mathbf{A}\}$ y el eigenvalor más pequeño como $\lambda_{\text{Min}}\{\mathbf{A}\}$ respectivamente, el teorema de Rayleigh-Ritz establece que para todo $\mathbf{x} \in \mathbb{R}^n$ se tiene

$$\lambda_{\text{Min}}\{\mathbf{A}\}\|\mathbf{x}\|^2 \leq \mathbf{x}^T \mathbf{A} \mathbf{x} \leq \lambda_{\text{Max}}\{\mathbf{A}\}\|\mathbf{x}\|^2$$

Una matriz simétrica $\mathbf{A} = \mathbf{A}^T \in \mathbb{R}^{n \times n}$ es definida positiva si y sólo si todos sus eigenvalores son positivos.

Norma espectral

La *norma espectral* $\|\mathbf{A}\|$ de una matriz $\mathbf{A} \in \mathbb{R}^{n \times m}$ se define como

$$\|\mathbf{A}\| = \sqrt{\lambda_{\text{Max}}\{\mathbf{A}^T \mathbf{A}\}}$$

donde $\lambda_{\text{Max}}\{\mathbf{A}^T \mathbf{A}\}$ denota el eigenvalor máximo de la matriz simétrica $\mathbf{A}^T \mathbf{A} \in \mathbb{R}^{m \times m}$.

Para el caso particular de matrices simétricas $\mathbf{A} = \mathbf{A}^T \in \mathbb{R}^{n \times n}$ se tiene que

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

En las expresiones anteriores el valor absoluto resulta redundante si \mathbf{A} es simétrica y definida positiva.

La norma espectral satisface las siguientes propiedades:

- $\|\mathbf{A}\| = 0$, si y sólo si $\mathbf{A} = \mathbf{0} \in \mathbb{R}^{n \times m}$
- $\|\mathbf{A}\| > 0$, para todo $\mathbf{A} \in \mathbb{R}^{n \times m}$ con $\mathbf{A} \neq \mathbf{0} \in \mathbb{R}^{n \times m}$
- $\|\mathbf{A} - \mathbf{B}\| \leq \|\mathbf{A}\| + \|\mathbf{B}\|$, para toda $\mathbf{A}, \mathbf{B} \in \mathbb{R}^{n \times m}$
- $\|\alpha \mathbf{A}\| = |\alpha| \|\mathbf{A}\|$, para toda $\alpha \in \mathbb{R}$ y $\mathbf{A} \in \mathbb{R}^{n \times m}$
- $\|\mathbf{A}^T \mathbf{B}\| \leq \|\mathbf{A}\| \|\mathbf{B}\|$, para toda $\mathbf{A}, \mathbf{B} \in \mathbb{R}^{n \times m}$

Otras desigualdades útiles son las siguientes. Considérese la matriz $\mathbf{A} \in \mathbb{R}^{n \times m}$ y el vector $\mathbf{x} \in \mathbb{R}^m$. La norma euclidiana del vector \mathbf{Ax} satisface que

$$\|\mathbf{Ax}\| \leq \|\mathbf{A}\| \|\mathbf{x}\|,$$

lógicamente $\|\mathbf{A}\|$ es la norma espectral de \mathbf{A} y $\|\mathbf{x}\|$ es la norma euclidiana de \mathbf{x} , asimismo, sea $\mathbf{y} \in \mathbb{R}^n$, el valor absoluto de $\mathbf{y}^T \mathbf{Ax}$ satisface que

$$\|\mathbf{y}^T \mathbf{Ax}\| \leq \|\mathbf{A}\| \|\mathbf{y}\| \|\mathbf{x}\|$$

Norma p inducida

La *norma p inducida* [9] de una matriz $\mathbf{A} \in \mathbb{R}^{n \times m}$ está definida por

$$\|\mathbf{A}\|_p = \sup_{\mathbf{x} \neq \mathbf{0}} \frac{\|\mathbf{Ax}\|_p}{\|\mathbf{x}\|_p} = \max_{\|\mathbf{x}\|_p=1} \|\mathbf{Ax}\|_p$$

para el caso de $p = 1, 2, \infty$ se tiene que

$$\begin{aligned} \|\mathbf{A}\|_1 &= \max_j \sum_{i=1}^n |a_{ij}| \\ \|\mathbf{A}\|_2 &= [\lambda_{\max} \{\mathbf{A}^T \mathbf{A}\}]^{1/2} \\ \|\mathbf{A}\|_\infty &= \max_i \sum_{j=1}^m |a_{ij}| \end{aligned}$$

Algunas propiedades útiles de las normas inducidas para las matrices reales $\mathbf{A} \in \mathbb{R}^{n \times m}$ y $\mathbf{B} \in \mathbb{R}^{m \times l}$, son las siguientes

$$\begin{aligned} \frac{1}{\sqrt{m}} \|\mathbf{A}\|_{\infty} &\leq \|\mathbf{A}\|_2 \leq \sqrt{n} \|\mathbf{A}\|_{\infty} \\ \frac{1}{\sqrt{n}} \|\mathbf{A}\|_1 &\leq \|\mathbf{A}\|_2 \leq \sqrt{m} \|\mathbf{A}\|_1 \\ \|\mathbf{A}\|_2 &\leq \sqrt{\|\mathbf{A}\|_1 + \|\mathbf{A}\|_{\infty}} \\ \|\mathbf{AB}\|_p &\leq \|\mathbf{A}\|_p \|\mathbf{B}\|_p \end{aligned}$$

2.1.3. Espacios funcionales

Una clase especial de espacios vectoriales son los llamados \mathcal{L}_p^n [8] donde n es un entero positivo y $p \in (0, \infty]$. Los espacios lineales denotados por \mathcal{L}_2^n y \mathcal{L}_{∞}^n , los cuales son definidos a continuación, son a menudo empleados en el análisis de sistemas dinámicos interconectados en la teoría de estabilidad.

Una función $\mathbf{f} : \mathbb{R}^n \rightarrow \mathbb{R}^m$ es *continua* si

$$\lim_{\mathbf{x} \rightarrow \mathbf{x}_0} \mathbf{f}(\mathbf{x}) = \mathbf{f}(\mathbf{x}_0) \quad \forall \mathbf{x}_0 \in \mathbb{R}^n$$

Entonces el espacio \mathcal{L}_2^n consiste en el conjunto de todas las funciones continuas $\mathbf{f} : \mathbb{R}_+ \rightarrow \mathbb{R}^n$ tal que

$$\int_0^{\infty} \mathbf{f}(t)^T \mathbf{f}(t) dt = \int_0^{\infty} \|\mathbf{f}(t)\|^2 dt < \infty$$

En otras palabras, una función \mathbf{f} pertenece al espacio \mathcal{L}_2^n ($\mathbf{f} \in \mathcal{L}_2^n$) si la integral de su norma euclidiana al cuadrado está acotada superiormente. Se dice también que \mathbf{f} es cuadrático integrable.

El espacio \mathcal{L}_{∞}^n consiste en el conjunto de todas las funciones continuas $\mathbf{f} : \mathbb{R}_+ \rightarrow \mathbb{R}^n$ tal que su norma euclidiana está acotada superiormente como

$$\sup_{t \geq 0} \|\mathbf{f}(t)\| < \infty$$

Ahora se presenta un resultado de la literatura de análisis funcional el cual provee condiciones suficientes para que funciones pertenecientes al espacio \mathcal{L}_2^n tengan un límite en cero. Este resultado es a menudo utilizado en la literatura de control adaptable.

Lema 2.1 *Considérese una función continuamente diferenciable $f : \mathbb{R}_+ \rightarrow \mathbb{R}^n$. Supóngase que f y su derivada temporal \dot{f} satisfacen lo siguiente*

- $f, \dot{f} = \frac{d}{dt}f \in \mathcal{L}_{\infty}^n$,
- $f \in \mathcal{L}_2^n$.

entonces, necesariamente $\lim_{t \rightarrow \infty} f(t) = \mathbf{0} \in \mathbb{R}^n$.

Prueba: Ver [8].

2.1.4. Funciones específicas

A continuación se presentan las definiciones de algunas funciones específicas [9] conocidas como funciones clase \mathcal{K} y clase \mathcal{KR} , que con frecuencia son empleadas en el estudio de las propiedades de la solución de sistemas no lineales no autónomos, los cuales serán abordados más adelante.

Definición 2.1 *Una función $\varphi : [0, r] \rightarrow \mathbb{R}_+$ (o una función continua $\varphi : [0, \infty) \rightarrow \mathbb{R}_+$) pertenece a la clase \mathcal{K} , i.e., $\varphi \in \mathcal{K}$ si*

1. $\varphi(0) = 0$
2. φ es estrictamente creciente en $[0, r]$ (o en $[0, \infty)$).

Definición 2.2 *Una función continua $\varphi : [0, \infty) \rightarrow \mathbb{R}_+$ pertenece a la clase \mathcal{KR} , i.e., $\varphi \in \mathcal{KR}$ si*

1. $\varphi(0) = 0$
2. φ es estrictamente creciente en $[0, \infty)$
3. $\lim_{r \rightarrow \infty} \varphi(r) = \infty$.

2.2. Sistemas dinámicos

Un *sistema* es la combinación de componentes o partes que es percibido como una sola entidad, estas partes están relacionadas unas con otras a través de un particular

conjunto de variables llamadas *estados* del sistema, que determinan completamente el comportamiento del sistema en cualquier instante dado. Un *sistema dinámico* es un sistema cuyos estados cambian con el tiempo. Específicamente, los estados del sistema pueden ser considerados como información almacenada de los eventos pasados del sistema.

Un sistema dinámico puede ser considerado como un modelo matemático que involucre en general entradas, estados y salidas que capturan la descripción dinámica de una determinada clase de sistemas físicos. Un sistema físico debe ser modelado dependiendo del enfoque o estudio que se desca realizar y es esencial un vasto conocimiento del sistema físico a fin de obtener un modelo adecuado del sistema, dado que las ecuaciones que lo describen pueden asumir diferentes formas. En ingeniería de control dos representaciones matemáticas de sistemas físicos que tienen una relevante importancia son los *sistemas lineales* y los *sistemas no lineales*. A continuación se presenta una breve descripción de estos conceptos, además de las representaciones y propiedades más importantes de este tipo de sistemas que han sido empleadas para el desarrollo de este trabajo de investigación.

2.2.1. Sistemas lineales

Los sistemas aquí abordados, se limitan a los llamados *sistemas lineales invariantes en el tiempo* [1] cuya representación en general está dada por

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t), \quad (2.1)$$

$$\mathbf{y}(t) = \mathbf{C}\mathbf{x}(t) + \mathbf{D}\mathbf{u}(t), \quad (2.2)$$

donde $\mathbf{x}(t) \in \mathbb{R}^n$ es el vector de estados, $\mathbf{u}(t) \in \mathbb{R}^m$ es el vector de entradas, $\mathbf{y}(t)$ es el vector de salida de dimensión p , mientras que $\mathbf{A} \in \mathbb{R}^{n \times n}$, $\mathbf{B} \in \mathbb{R}^{n \times m}$ son denominadas matriz del sistema y matriz de entrada respectivamente, finalmente $\mathbf{C} \in \mathbb{R}^{p \times n}$ es la matriz de salida y $\mathbf{D} \in \mathbb{R}^{p \times m}$ que en la mayoría de los sistemas es una matriz nula. Vale la pena mencionar que \mathbf{C} y \mathbf{D} al igual que \mathbf{A} y \mathbf{B} son matrices constantes, es decir, no son funciones del tiempo. La matriz de función de transferencia $\mathbf{H}(s)$ del sistema está definida como $\mathbf{H}(s) = \mathbf{C}(s\mathbf{I} - \mathbf{A})^{-1}\mathbf{B}$ donde s es un número complejo, cabe señalar que cada polo de $\mathbf{H}(s)$ es un eigenvalor de la matriz \mathbf{A} , así, si cada valor propio de \mathbf{A} tiene parte real negativa, entonces cada polo de $\mathbf{H}(s)$ tiene parte real negativa.

La ecuación (2.1) es un conjunto de ecuaciones diferenciales lineales de primer orden y la ecuación (2.2) es un conjunto de ecuaciones algebraicas; ambas ecuaciones dan lugar al llamado *espacio de estados* o simplemente *ecuación de estados*.

La solución de la ecuación (2.1) es obtenida de la forma

$$\mathbf{x}(t) = e^{At} \mathbf{x}_0 + \int_0^t e^{A(t-\tau)} \mathbf{B} \mathbf{u}(\tau) d\tau \quad (2.3)$$

donde \mathbf{x}_0 es la condición inicial de $\mathbf{x}(t)$ en $t = t_0$. De lo anterior resulta claro que para sistemas lineales invariantes en el tiempo la solución es independiente del tiempo inicial t_0 ; todo lo que importa es cuanto tiempo ha pasado desde entonces, i.e. $t - t_0$. Sin pérdida de generalidad, asúmase que $t_0 = 0$ y se define

$$\mathbf{K}(t) = \begin{cases} e^{At} \mathbf{B} & \text{para } t \geq 0 \\ \mathbf{0} & \text{para } t < 0 \end{cases}$$

Ahora, nótese que la integral en (2.3) es simplemente la convolución de la entrada $\mathbf{u}(\cdot)$, con la matriz respuesta al impulso del sistema, entonces la solución a la ecuación (2.1) puede reescribirse como

$$\mathbf{x}(t) = e^{At} \mathbf{x}_0 + (\mathbf{K} * \mathbf{u})(t) \quad (2.4)$$

donde $*$ es el operador de convolución. Un resultado importante en análisis real que puede ser útil en el estudio de sistemas lineales es la llamada desigualdad de Young [23], la cual establece lo siguiente

Suponga que $\mathbf{f} \in \mathcal{L}^p(\mathbb{R}^n)$ y $\mathbf{g} \in \mathcal{L}^q(\mathbb{R}^n)$, si $1 \leq p, q, r \leq \infty$, $\frac{1}{p} + \frac{1}{q} = 1 + \frac{1}{r}$ entonces

$$\|\mathbf{f} * \mathbf{g}\|_r \leq \|\mathbf{f}\|_p \|\mathbf{g}\|_q.$$

Lema 2.2 *La respuesta a entrada cero del sistema (2.1) o bien, la solución a la ecuación $\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t)$ tiende a $\mathbf{0}$ conforme $t \rightarrow \infty$ si y sólo si todos los eigenvalores de \mathbf{A} tienen parte real negativa.*

Prueba: Ver [1].

Lema 2.3 *Considere la matriz función de transferencia de dimensión n , $\mathbf{H}(s) \in \mathbb{K}^{n \times n}(s)$ cuyos elementos son funciones racionales estrictamente propias¹ de la variable*

¹Es decir, el grado del denominador es estrictamente mayor que el grado del numerador.

compleja s . Asuma que el denominador de todos sus elementos tiene todas sus raíces en el lado izquierdo del plano complejo (i.e. tienen parte real negativa).

1. si $\mathbf{u} \in \mathcal{L}_2^n$ entonces $\mathbf{y} \in \mathcal{L}_2^n \cap \mathcal{L}_\infty^n$, $\dot{\mathbf{y}} \in \mathcal{L}_2^n$ y $\mathbf{y}(t) \rightarrow \mathbf{0}$ conforme $t \rightarrow \infty$.
2. si $\mathbf{u} \in \mathcal{L}_\infty^n$ entonces $\mathbf{y} \in \mathcal{L}_\infty^n$, $\dot{\mathbf{y}} \in \mathcal{L}_\infty^n$.

Prueba: Ver [8], [2].

2.2.2. Sistemas no lineales

Ya que la mayoría de los sistemas físicos son inherentemente no lineales, el modelado no lineal de sistemas puede exhibir un comportamiento dinámico más rico. Aquí se presenta una breve descripción de una clase de sistemas no lineales no autónomos (variantes en el tiempo) los cuales satisfacen ciertas propiedades como existencia y unicidad de soluciones, dependencia continua de las condiciones iniciales y parámetros del sistema, las cuales pueden ser consultadas en textos especializados como [3] y [9]. Específicamente, se considera el sistema no lineal variante en el tiempo

$$\dot{\mathbf{x}}(t) = \mathbf{f}(t, \mathbf{x}(t)), \quad \mathbf{x}(t_0) = \mathbf{x}_0, \quad t \geq t_0 \quad (2.5)$$

donde $\mathbf{x}(t) \in D \subset \mathbb{R}^n$, $\mathbf{f} : [t_0, \infty) \times D \rightarrow \mathbb{R}^n$ y $D = \{\mathbf{x} \in \mathbb{R}^n \mid |\mathbf{x}| < r\}$. Se asume además que \mathbf{f} es de tal naturaleza que para cada $\mathbf{x}_0 \in D$ y cada $t_0 \in \mathbb{R}_+$, la solución $\mathbf{x}(t, t_0, \mathbf{x}_0) = \mathbf{x}(t)$ de (2.5) existe y es única.

2.3. Estabilidad en el sentido de Lyapunov

Uno de los problemas más básicos en la teoría de sistemas es la estabilidad de sistemas dinámicos, caracterizada por el análisis de la respuesta del sistema a pequeñas perturbaciones en sus estados. Existen diferentes clases de problemas de estabilidad que surgen del estudio de sistemas dinámicos. Una de las herramientas matemáticas más poderosas y utilizadas para analizar la estabilidad de un sistema dinámico es la teoría de Lyapunov, cuyo método está basado en la construcción de una función de las coordenadas del estado del sistema que sirve como una norma generalizada de la solución del sistema dinámico. Su atractivo parte del hecho de que las conclusiones

acerca del comportamiento dinámico del sistema pueden ser obtenidas sin tener que calcular las trayectorias o soluciones del sistema.

En esta sección se presentan algunas de las generalizaciones y extensiones de la teoría de estabilidad de Lyapunov, tal como el teorema de acotamiento de soluciones que puede ser empleado cuando el sistema no tiene puntos de equilibrio [3].

Definición 2.3 *El sistema no lineal no autónomo (2.5) es estable en el sentido de Lyapunov si, para cada $\epsilon > 0$ y $t_0 \in [0, \infty)$, existe $\delta = \delta(\epsilon, t_0) > 0$ tal que $\|\mathbf{x}_0\| < \delta$ implica que $\|\mathbf{x}(t)\| < \epsilon$ para todo t_0 .*

Definición 2.4 *El sistema no lineal no autónomo (2.5) es uniformemente estable en el sentido de Lyapunov si, para cada $\epsilon > 0$ y $t_0 \in [0, \infty)$, existe $\delta = \delta(\epsilon) > 0$ tal que $\|\mathbf{x}_0\| < \delta$ implica que $\|\mathbf{x}(t)\| < \epsilon$ para todo t_0 y para todo $t \in [0, \infty)$.*

Teorema 2.1 *Considérese el sistema no lineal no autónomo dado por la ecuación (2.5). Entonces las siguientes afirmaciones se cumplen:*

1. *Si existe una función $V(t, \mathbf{x}) : [0, \infty) \times D \rightarrow \mathbb{R}$ continuamente diferenciable y una función $\alpha(\cdot)$ clase \mathcal{K} tal que*

$$V(t, 0) = 0, \quad t \in [0, \infty), \quad (2.6)$$

$$\alpha(\|\mathbf{x}\|) \leq V(t, \mathbf{x}), \quad (t, \mathbf{x}) \in [0, \infty) \times D, \quad (2.7)$$

$$\dot{V}(t, \mathbf{x}) \leq 0, \quad (t, \mathbf{x}) \in [0, \infty) \times D, \quad (2.8)$$

entonces el sistema no lineal no autónomo dado por (2.5) es estable en el sentido de Lyapunov.

2. *Si existe una función $V(t, \mathbf{x}) : [0, \infty) \times D \rightarrow \mathbb{R}$ continuamente diferenciable y funciones $\alpha(\cdot)$, $\beta(\cdot)$ clase \mathcal{K} que satisfacen (2.7), (2.8) y*

$$V(t, \mathbf{x}) \leq \beta(\|\mathbf{x}\|), \quad (t, \mathbf{x}) \in [0, \infty) \times D, \quad (2.9)$$

entonces el sistema no lineal no autónomo dado por (2.5) es uniformemente estable en el sentido de Lyapunov.

Prueba: Ver [3].

2.3.1. Acotamiento de soluciones

Previamente fueron introducidos conceptos de estabilidad en el sentido de Lyapunov, ahora se tratará con el concepto de *acotamiento de soluciones*, el cual parte de la idea de indagar si para cada condición inicial del sistema en una bola de radio δ la solución del sistema no lineal está acotada. Esta noción está ampliamente relacionada a lo que se conoce en la literatura como *estabilidad práctica* y es utilizada también en estudios de control adaptable como una herramienta parcial para la determinación de convergencia de soluciones.

Definición 2.5 *El sistema no lineal no autónomo (2.5) está uniformemente acotado si existe $\gamma > 0$ tal que, para cada $\delta \in (0, \gamma)$ existe $\epsilon = \epsilon(\delta) > 0$ tal que $\|\mathbf{x}_0\| < \delta$ implica que $\|\mathbf{x}(t)\| < \epsilon$, $t \geq t_0$.*

Teorema 2.2 *Asuma que (2.5) posee solución única para toda $\mathbf{x}_0 \in \mathbb{R}^n$. Si existe una función $V(t, \mathbf{x})$ definida en $\|\mathbf{x}\| \geq \mu$ (donde μ puede ser grande) y $t \in [0, \infty)$ con derivadas parciales de primer orden continuas respecto a \mathbf{x} y a t , si existen funciones $\alpha_1, \alpha_2 \in \mathcal{KR}$ tal que*

$$\alpha_1(\|\mathbf{x}\|) \leq V(t, \mathbf{x}) \leq \alpha_2(\|\mathbf{x}\|), \quad (2.10)$$

$$\dot{V}(t, \mathbf{x}) \leq 0, \quad (2.11)$$

para toda $\|\mathbf{x}\| \geq \mu$ y $t \in [0, \infty)$, entonces las soluciones de (2.5) están uniformemente acotadas.

2.4. Modelo dinámico de robots manipuladores

Esta sección brinda una descripción concisa del modelo dinámico de robots manipuladores seriales, su obtención queda fuera del alcance de este documento y puede ser consultada en textos especializados como [8], [19], [22]. Las ecuaciones dinámicas de un manipulador describen explícitamente la relación entre las fuerzas que actúan en las articulaciones y el movimiento de la estructura. El cálculo del *modelo dinámico* de un robot manipulador juega un papel importante en la simulación de movimiento y el diseño de algoritmos de control. La simulación de movimiento de un brazo robótico permite a las estrategias de control y a la planificación de técnicas de movimiento ser

sometidas a ensayo sin la necesidad de usar un sistema físico disponible. Un método para el cálculo de las ecuaciones de movimiento de un manipulador en el *espacio articular*, es decir en función de las variables articulares (*coordenadas generalizadas*) q_i ($i = 1, \dots, n$) que definen la posición de los eslabones de un manipulador de n grados de libertad, es el basado en la *formulación Euler-Lagrange* que es conceptualmente simple y sistemático. Considérese entonces un robot articulado de n grados de libertad de cadena cinemática abierta, compuesto de eslabones rígidos interconectados, su modelo dinámico puede ser expresado en forma compacta como

$$\mathbf{B}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{g}(\mathbf{q}) + \mathbf{f}(\dot{\mathbf{q}}) = \boldsymbol{\tau} \quad (2.12)$$

la cual es una ecuación diferencial vectorial no lineal en el estado $[\mathbf{q}^T \ \dot{\mathbf{q}}^T]^T$ donde en general $\mathbf{q} \in \mathbb{R}^n$ es el vector de posiciones articulares y $\dot{\mathbf{q}}$ su derivada temporal, es decir el vector de velocidades articulares, $\boldsymbol{\tau} \in \mathbb{R}^n$ es el vector de pares aplicados, $\mathbf{B}(\mathbf{q}) = \mathbf{B}^T(\mathbf{q}) > 0 \in \mathbb{R}^{n \times n}$ es la matriz de inercias, $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) \in \mathbb{R}^{n \times n}$ es la matriz de fuerzas centrífugas y de Coriolis, $\mathbf{g}(\mathbf{q}) \in \mathbb{R}^n$ representa el vector de pares gravitacionales y finalmente $\mathbf{f}(\dot{\mathbf{q}}) \in \mathbb{R}^n$ es el vector de pares debidos a la fricción.

A pesar de la complejidad de la ecuación dinámica (2.12), la cual describe el comportamiento de robots manipuladores, esta ecuación y los términos que lo constituyen satisfacen ciertas propiedades notables [8] y de particular relevancia para el estudio de control y análisis de estabilidad del presente trabajo de investigación. Tales propiedades son enunciadas a continuación.

2.4.1. Propiedades

Suponiendo que el robot está dotado exclusivamente de articulaciones del tipo revolución el modelo dado por la ecuación (2.12) cumple lo siguiente:

Propiedad 2.1 La matriz de inercia es tal que satisface

$$b'_1 \leq \|\mathbf{B}(\mathbf{q})\| \leq b'_2$$

para algunas constantes positivas b'_1, b'_2 .

Propiedad 2.2 La matriz $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})$ obtenida usando los llamados coeficientes de

Christoffel, está relacionada con la matriz de inercia por la expresión

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

Propiedad 2.3 Existe una constante $k_{C_1} > 0$ tal que

$$\|\mathbf{C}(\mathbf{q}, \mathbf{x})\mathbf{y}\| \leq k_{C_1} \|\mathbf{x}\| \|\mathbf{y}\|$$

para todo $\mathbf{q}, \mathbf{x}, \mathbf{y} \in \mathbb{R}^n$.

Propiedad 2.4 Existe una constante k' tal que

$$\|\mathbf{g}(\mathbf{q})\| \leq k' \quad \forall \mathbf{q} \in \mathbb{R}^n$$

Propiedad 2.5 El vector de fricción está limitado superiormente de la forma

$$\|\mathbf{f}(\dot{\mathbf{q}})\| \leq c_1 + c_2 \|\dot{\mathbf{q}}\|$$

para algunas constantes positivas c_1, c_2 .

2.5. Control de robots manipuladores

El problema de control de un manipulador puede ser formulado como: determinar el registro en el tiempo de las fuerzas generalizadas (fuerzas o pares) a desarrollar por los actuadores, con el fin de garantizar la ejecución de las tareas asignadas, al tiempo que se satisfacen los requisitos del estado transitorio y el estado estacionario. La tarea asignada debe considerar la ejecución de los movimientos especificados para un manipulador operando en el espacio libre o la ejecución de movimientos especificados y las fuerzas de contacto para un manipulador cuyo efector final está limitado por el entorno. En vista de la complejidad del problema, ambos aspectos se tratan de manera separada [19]: el control de la interacción con el ambiente, el cual es analizado por medio de técnicas de control en el *espacio operacional*, y el control de movimiento en el espacio libre, el cual es abordado a través de técnicas de control en el *espacio articular*. Estas últimas pueden ser distinguidas entre esquemas de *control descentralizado*; esto es, cada articulación del manipulador es controlada de manera individual e independiente de las otras empleando

sólo las variables locales de dicha articulación, donde las interconexiones entre eslabones son consideradas como una perturbación; y esquemas de *control centralizado*, que es cuando los efectos de interacción dinámica entre las articulaciones son tomados en cuenta.

En esta sección se abordará exclusivamente el estudio de controladores de movimiento para robots manipuladores que se desplazan libremente en su espacio de trabajo, esto es, sin interactuar con el entorno, asimismo se considerarán robots manipuladores provistos de actuadores ideales sin dinámica, es decir, los actuadores proporcionarán pares y fuerzas proporcionales a sus entradas.

2.5.1. Especificaciones de control

En la etapa de control de un sistema cualquiera se deben determinar las variables físicas del sistema cuyo comportamiento se desea gobernar, estas variables reciben el nombre de *salidas del sistema*. Además, también deben identificarse claramente aquellas variables físicas que se encuentran disponibles y que influyen en la evolución de las salidas del sistema denominadas *variables de entrada*.

En el caso particular de robots manipuladores, la variable de salida, denotada momentáneamente por \mathbf{y} cuya conducta se desea controlar, ofrece un amplio espectro de elecciones tal como se explica a continuación.

En el caso de robots que se desplazan libremente dentro de su espacio de trabajo sin interactuar con el medio ambiente, la salida \mathbf{y} puede corresponder simplemente a las posiciones articulares \mathbf{q} y velocidades articulares $\dot{\mathbf{q}}$, o también a la posición y orientación del órgano terminal.

Para robots manipuladores que involucran su interacción con el medio ambiente por contacto físico para realizar las tareas asignadas, la salida \mathbf{y} puede incluir los pares y fuerzas \mathbf{f} ejercidos por el extremo del último eslabón del robot sobre su medio ambiente.

Dicho esto, se aprecia que la salida \mathbf{y} correspondiente a un robot, asociado a una clase de tareas, en general puede tener la forma funcional

$$\mathbf{y} = \mathbf{y}(\mathbf{q}, \dot{\mathbf{q}}, \mathbf{f})$$

Por otra parte, las variables de entrada, aquellas que pueden ser modificadas para

alterar la evolución de las salidas, son básicamente los pares y fuerzas τ aplicados por los actuadores sobre las articulaciones del robot.

Por ejemplo, para un manipulador el cuál se desplaza libremente dentro de su espacio de trabajo, sin que tenga una interacción con el entorno, la posición y velocidad articular q y \dot{q} , respectivamente, son las salidas del robot, es decir

$$y = y(q, \dot{q}, f) = \begin{bmatrix} q & \dot{q} \end{bmatrix}^T$$

mientras τ es su entrada. En esta situación, nótese que para robots con n articulaciones se tendrán en general $2n$ salidas y n entradas.

En particular, en el caso típico que la salida y de un robot manipulador corresponda a la posición articular q y velocidad articular \dot{q} , los objetivos de control asociados a *regulación* y *seguimiento de trayectoria* reciben los nombres de posición pura en coordenadas articulares y control de movimiento en coordenadas articulares, respectivamente. A éstos se hará mención en lo sucesivo simplemente como *control de posición* y *control de movimiento* [8], respectivamente.

Entonces, el problema de control de movimiento para robots manipuladores radica en encontrar una función vectorial τ tal que las posiciones q , asociadas a las coordenadas articulares del robot sigan a las trayectorias deseadas q_d con precisión.

En términos más formales, el *objetivo del control de movimiento*, consiste en encontrar τ tal que

$$\lim_{t \rightarrow \infty} \tilde{q}(t) = 0 \quad (2.13)$$

donde $\tilde{q} \in \mathbb{R}^n$ representa el error de posición articular o simplemente error de posición y está definido como

$$\tilde{q}(t) := q_d(t) - q(t).$$

2.6. Control adaptable

El área de control adaptable ha crecido hasta ser una de las más ricas en términos de algoritmos, técnicas de diseño y herramientas analíticas, entre otras. Partiendo del hecho de que la salida de un sistema contiene información acerca de los estados de dicho sistema, así como de sus parámetros los que se consideran desconocidos, un sofisticado controlador por realimentación podría ser capaz de aprender acerca de los cambios en

los parámetros a través del procesamiento de la salida y utilizar ganancias apropiadas para ajustarlos. Este argumento lleva a la estructura de control por realimentación en la cuál está basada el control adaptable [7]. La estructura del controlador consiste en un lazo de realimentación y un controlador con parámetros ajustables. La manera de modificar los parámetros del controlador en respuesta a la salida del sistema distingue un esquema de otro.

Un controlador adaptable está formado entonces por la combinación de un estimador de parámetros en línea denominado *ley adaptable*, el cual provee estimados de los parámetros desconocidos de la planta a cada instante, con una ley de control que es motivada a partir del caso de parámetros conocidos. En el *control adaptable indirecto*, los parámetros de la planta son estimados en línea y usados para calcular los parámetros del controlador; mientras que en el *control adaptable directo*, el modelo de la planta es parametrizado en términos de los parámetros del controlador que son estimados directamente sin cálculos intermedios que envuelvan los parámetros de la planta.

El diseño de la ley de adaptación es crucial para las propiedades de estabilidad del controlador adaptable, dado que la ley de adaptación introduce una no linealidad que hace al sistema en lazo cerrado un sistema no lineal y variante en el tiempo. Un método para garantizar la estabilidad del sistema en lazo cerrado es conocido como diseño por síntesis de Lyapunov y está basado en el método directo de Lyapunov. En este enfoque el problema de diseñar una ley adaptable es formulado como un problema de estabilidad donde la ecuación diferencial de la ley de adaptación se escoge tal que ciertas condiciones de estabilidad basadas en la teoría de Lyapunov sean satisfechas.

Lógica difusa

El campo de sistemas de control difuso ha tenido un rápido progreso en años recientes, en parte debido a que el control difuso es una alternativa práctica para una variedad de desafiantes aplicaciones de control, ya que provee un método conveniente para la construcción de controladores no lineales vía el uso de información heurística, utilizando lenguaje natural para la cuantificación, razonamiento e inferencia de este tipo de información. Actualmente los investigadores están tratando de sistematizar los enfoques existentes y de desarrollar nuevos algoritmos más poderosos.

El concepto de lógica difusa fue introducido por primera vez en 1965 por el profesor Lofti A. Zadeh de la Universidad de California en Berkeley en su artículo titulado *Fuzzy Sets*[26] (conjuntos difusos), como una alternativa para describir conjuntos que no pueden definirse con facilidad, o bien para describir el grado de pertenencia de ciertos elementos en dichos conjuntos.

La lógica difusa o teoría de conjuntos difusos es una herramienta matemática basada en grados de membresía que permite modelar información que contiene ambigüedad, imprecisión e incertidumbre, al medir el grado en que un evento ocurre, empleando para esto una base de conocimiento o razonamiento humano y está basada en la teoría de posibilidad, que a su vez está relacionada con la incertidumbre de eventos pasados y/o actuales.

Un sistema basado en lógica difusa puede modelar un sistema no lineal basándose en un conocimiento previo sin necesidad de una amplia descripción matemática. De igual forma, se puede construir un controlador utilizando lógica difusa para calcular una

salida de control adecuada según la experiencia disponible sobre el sistema a controlar. Este capítulo está basado en [14] y presenta algunos conceptos y definiciones propios de lógica difusa, así como los fundamentos teóricos en los que se ha basado parte del proyecto de investigación desarrollado.

3.1. Conjuntos difusos

Por definición, un conjunto consiste en una colección de objetos que pueden ser tratados como una sola entidad y cumplen con ciertas características o condiciones. El conjunto universal el cual contiene todos los posibles elementos concernientes a un contexto en particular es llamado *universo de discurso*.

Sea $X \in U$ el universo de discurso, o conjunto universal, con elementos de X denotados por x . Un conjunto difuso A de X es definido por una función de membresía o pertenencia $\mu_A(x)$ definida como $\mu_A(x) : X \in U \rightarrow [0, 1]$ que asocia cada elemento x con un grado o valor de membresía en A .

Entonces, un conjunto difuso A en $X \subset U$ puede ser representado como un conjunto de pares ordenados de un elemento genérico x y su valor de pertenencia $\mu_A(x)$ que representa el grado en el que el elemento x pertenece al conjunto difuso A , esto es

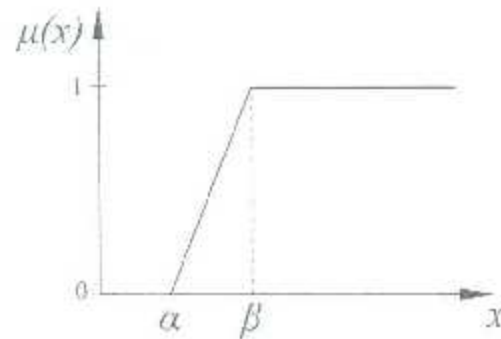
$$A = \{(x, \mu_A(x) | x \in X)\}.$$

3.1.1. Funciones de membresía

Las funciones de membresía o funciones de pertenencia que caracterizan conjuntos difusos pueden tener varias formas. Enseguida se examinan los tipos de funciones empleadas en este trabajo de investigación.

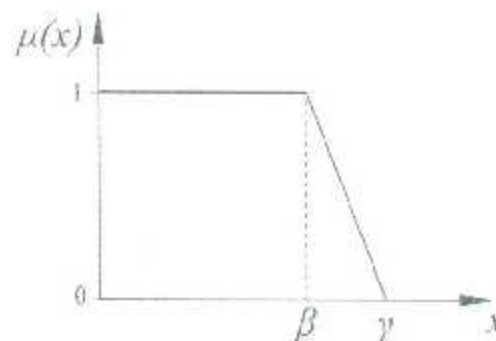
La función pseudo-trapezoidal $\Gamma : U \rightarrow [0, 1]$ mostrada en la figura 3.1 es una función con dos parámetros α y β definida como:

$$\Gamma(x; \alpha, \beta) = \begin{cases} 0 & x < \alpha, \\ (x - \alpha) / (\beta - \alpha) & \alpha \leq x \leq \beta, \\ 1 & x > \beta. \end{cases}$$

Figura 3.1: Función de pertenencia pseudo-trapezoidal Γ .

La función pseudo-trapezoidal $L : U \rightarrow [0, 1]$ que se ilustra en la figura 3.2 es una función con dos parámetros α y β definida como:

$$L(x; \alpha, \beta) = \begin{cases} 1 & x < \alpha, \\ (\beta - x)/(\beta - \alpha) & \alpha \leq x \leq \beta, \\ 0 & x > \beta. \end{cases}$$

Figura 3.2: Función de pertenencia pseudo-trapezoidal L .

La función triangular $\Lambda : U \rightarrow [0, 1]$ de la figura 3.3 es una función con tres parámetros α , β y γ dada por:

$$\Lambda(x; \alpha, \beta, \gamma) = \begin{cases} 0 & x < \alpha, \\ (x - \alpha)/(\beta - \alpha) & \alpha \leq x \leq \beta, \\ (\gamma - x)/(\beta - \alpha) & \beta \leq x \leq \gamma, \\ x > \gamma. \end{cases}$$

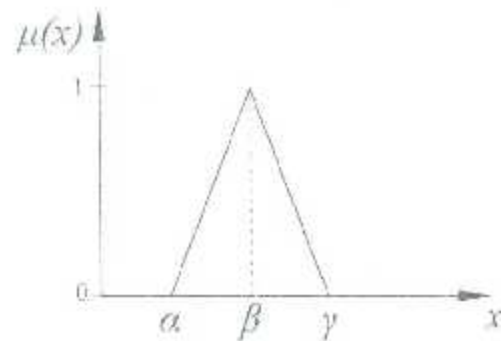


Figura 3.3: Função de pertinência triangular A.

Função de pertinência impulsiva δ com parâmetro \bar{x} , mostrada em la figura 3.4 y definida por

$$\delta(x, \bar{x}) = \begin{cases} 1 & x = \bar{x} \\ 0 & x \neq \bar{x} \end{cases}$$

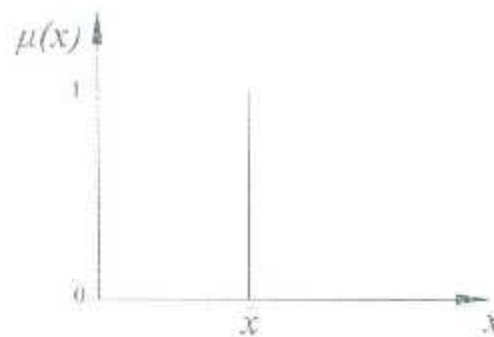


Figura 3.4: Função de pertinência impulsiva δ.

3.2. Variable lingüística

Si una variable puede tomar palabras en lenguaje natural como sus valores, entonces puede ser llamada "variable lingüística". Esas palabras son caracterizadas por conjuntos difusos definidos en un universo de discurso específico. Las variables lingüísticas permiten entonces expresar conceptos vagos en un lenguaje natural en términos matemáticos precisos. Esto permite incorporar conocimiento humano a los sistemas de control.

Una variable lingüística es definida por los siguientes parámetros:

- X es el nombre de la variable lingüística,
- T es el conjunto de valores lingüísticos que la variable X puede tomar,
- U es el universo de discurso de la variable; es decir el rango de valores reales que la variable puede tomar,
- M es una regla semántica que relaciona cada valor lingüístico en T con un conjunto difuso en U .

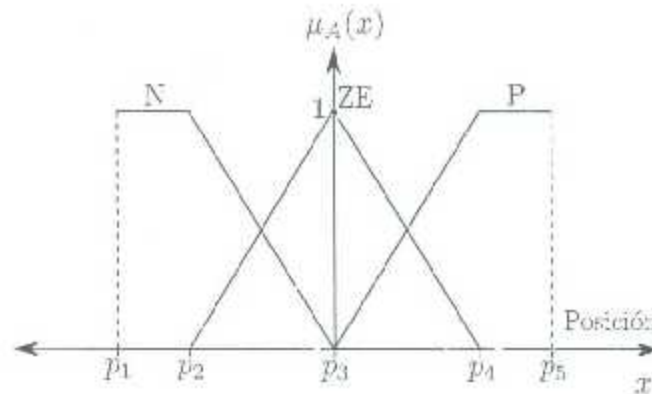


Figura 3-5: Variable lingüística.

En la figura 3.5 por ejemplo, la variable lingüística está definida por los siguientes parámetros: $X = \text{Posición}$, $T = \{N, ZE, P\}$, $U = [p_1, \dots, p_5]$ donde p_1, \dots, p_5 son los parámetros de soporte que definen las funciones de pertenencia de cada conjunto difuso. Finalmente, M relaciona los valores lingüísticos de T con cada función de pertenencia mostrada.

3.3. Fusificación

En el proceso de fusificación se reciben las n variables de entrada x_i ($i = 1, 2, \dots, n$) y a cada valor real x_i^* (valor *crisp*) se asigna un grado de pertenencia $\mu_{A_i^b}(x_i^*)$ para cada uno de los p_i conjuntos difusos A_i^b . Este proceso puede ser visto como un mapeo definido por

$$\mu_{A_i^b}(x_i^*) : U \rightarrow [0, 1] \subset \mathbb{R}_1 \quad (3.1)$$

La fusificación impulsiva o *singleton* (figura 3.6) es el tipo de fusificación más utilizada en control automático para aplicaciones en tiempo real, dado que reduce significativamente el tiempo de cómputo y está definida como

$$\mu_{A_i^j}(x_i^*) = \begin{cases} \mu_{A_i^j}(x_i^*) & x_i = x_i^* \\ 0 & x_i \neq x_i^* \end{cases}$$

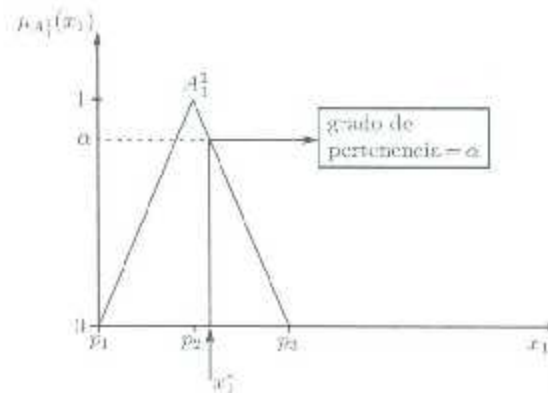


Figura 3.6. Proceso de fusificación *singleton*

3.4. Reglas difusas

Una regla difusa está dada por la siguiente estructura [14]:

IF proposición difusa *THEN* proposición difusa

donde una *proposición difusa* puede ser del tipo atómica; es decir de un solo enunciado tal como *IF* x es A *THEN* y es B , o del tipo compuesto en el cual se utilizan dos o más enunciados atómicos conectados por operadores *AND*, *OR* y *NOT*, llamados conectivos.

3.4.1. Base de reglas difusas

La base de reglas difusas es una de las partes principales de un sistema difuso ya que almacena el conocimiento heurístico y cualitativo en el que se ha de basar un sistema difuso. La base de reglas consiste de un conjunto de reglas difusas del tipo *IF THEN* (3.2) denominada $Ru^{(1-\alpha)}$ cuya estructura es descrita por la siguiente ecuación

$$Ru^{(l_1, \dots, l_n)} : \text{IF } x_1 \text{ es } A_1^{l_1} \text{ AND } \dots \text{ AND } x_n \text{ es } A_n^{l_n} \text{ THEN } y \text{ es } B^{l_1, \dots, l_n} \quad (3.2)$$

donde x_i ($i = 1, 2, \dots, n$) son las variables lingüísticas de entrada que pueden estar contenidas en $\mathbf{x} = (x_1, x_2, \dots, x_n)^T \in U \subset \mathbb{R}^n$, mientras que $y \in V \subset \mathbb{R}$ es la variable lingüística de salida; $A_i^{l_i}$ ($i = 1, 2, \dots, p_i$) son los conjuntos difusos definidos en U y B^{l_1, \dots, l_n} son los conjuntos difusos definidos en V . $M = \prod_{i=1}^n p_i$ es el número de reglas difusas en la base de reglas. A esta estructura de reglas se le llama comúnmente *forma canónica difusa de reglas IF-THEN*.

Los sistemas difusos almacenan reglas como asociaciones difusas. El almacenamiento se hace en tablas (matrices). A esta tabla se le conoce como *Fuzzy Associative Memory* (FAM), tabla de consulta (*look-up table*) o simplemente tabla de base de reglas.

Tabla 3.1: Tabla de consulta de reglas difusas

$x_1 \backslash x_2$	$l_1 = 1$ A_1^1	$l_1 = 2$ A_1^2	$l_1 = 3$ A_1^3	$l_1 = 4$ A_1^4	$l_1 = 5$ A_1^5
$l_2 = 1$ A_2^1	B^{11}	B^{21}	B^{31}	B^{41}	B^{51}
$l_2 = 2$ A_2^2	B^{12}	B^{22}	B^{32}	B^{42}	B^{52}
$l_2 = 3$ A_2^3	B^{13}	B^{23}	B^{33}	B^{43}	B^{53}
$l_2 = 4$ A_2^4	B^{14}	B^{24}	B^{34}	B^{44}	B^{54}
$l_2 = 5$ A_2^5	B^{15}	B^{25}	B^{35}	B^{45}	B^{55}

La tabla 3.1 es un ejemplo de las asociaciones difusas que forman la tabla de consulta para una base de reglas con estructura (3.2) para el caso de dos entradas y cinco conjuntos difusos definidos para cada una de ellas.

3.5. Inferencia difusa

El proceso de inferencia difusa evalúa el conjunto de reglas (obtenidas del conocimiento heurístico), de acuerdo a un tipo de implicación (la manera en que se interpretan las reglas *IF-THEN*), para obtener un conjunto difuso de salida inferido. En otras palabras, la inferencia puede entenderse como el proceso que: dada una relación difusa,

previamente establecida, entre un conjunto de entrada A y un conjunto de salida B , se puede concluir cómo sería una salida B' dada una nueva entrada A' .

Este proceso de inferencia es llevado a cabo por la *máquina de inferencia* o *máquina inferencial*. Existen en la literatura muchos tipos de máquinas de inferencia en base a varios factores tales como la secuencia en que se evalúan las reglas, el tipo de implicación seleccionada y la manera en que se combinan las reglas en la tabla de consulta para obtener la salida final inferida. Estos factores así como los diversos tipos de máquinas de inferencia pueden ser consultados en textos especializados como [24].

En este documento se trata con la máquina inferencial producto ya que es una de las más ampliamente usadas en control difuso debido a que presenta simplicidad en la implementación y bajo tiempo de cálculo, características que la hacen ideal para control en tiempo real.

3.5.1. Inferencia producto

Dado un conjunto de reglas con estructura (3.2), funciones de pertenencia de entrada $\mu_{A_i^j}(x_i)$ y funciones de pertenencia de salida $\mu_{B_i^j \rightarrow i_n}(y)$ para toda $\mathbf{x} = (x_1, x_2, \dots, x_n)^T \in U \subset \mathbb{R}^n$, $y \in V \subset \mathbb{R}$, la inferencia o conclusión sobre entradas tipo *singleton* que se puede obtener por el método del producto está dada por

$$\mu_{B'}(y) = \max_{i_1=1}^{p_1} \cdots \max_{i_n=1}^{p_n} [\mu_{B_{i_1 \dots i_n}}(y)] \quad (3.3)$$

$$= \mu_{B_{i_1}}(y) \cup \cdots \cup \mu_{B_{i_1 \dots i_n}}(y) \quad (3.4)$$

donde el símbolo \cup denota el operador unión y

$$\mu_{B_{i_1 \dots i_n}}(y) = \prod_{i=1}^n \mu_{A_i^j}(x_i^*) \mu_{B_i^j \rightarrow i_n}(y). \quad (3.5)$$

En la inferencia producto el valor escalar $\omega = \prod_{i=1}^n \mu_{A_i^j}(x_i^*)$ es multiplicado punto a punto por la función $\mu_{B_i^j \rightarrow i_n}(y)$, lo cual resulta en una nueva función $\mu_{B_{i_1 \dots i_n}}(y)$ que es similar a $\mu_{B_i^j \rightarrow i_n}(y)$, pero escalada al valor de ω (ver la figura 3.7). A este método se le conoce también como el *método de escalamiento*.

Una vez que se ha obtenido el consecuente de cada una de las reglas es decir, M funciones $\mu_{B_{i_1 \dots i_n}}(y)$, es necesario combinarlas para obtener una conclusión difusa final o salida

de la máquina inferencial (ver la figura 3.8); esto es, $\mu_{B'}(y)$, lo que se hace mediante la operación unión.

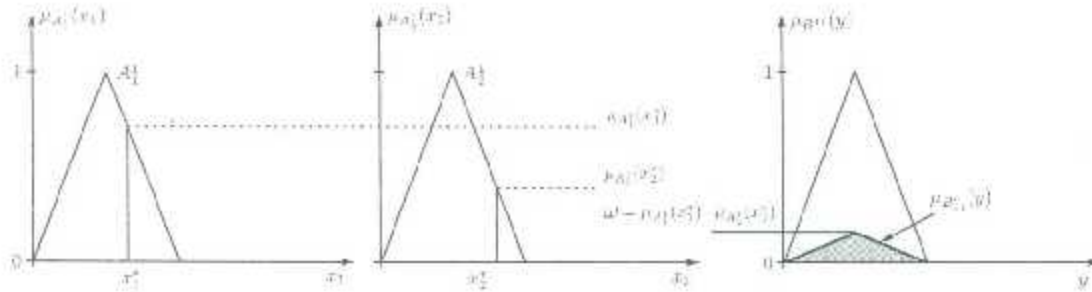


Figura 3.7: Método de inferencia "producto usando fusificador singleton".

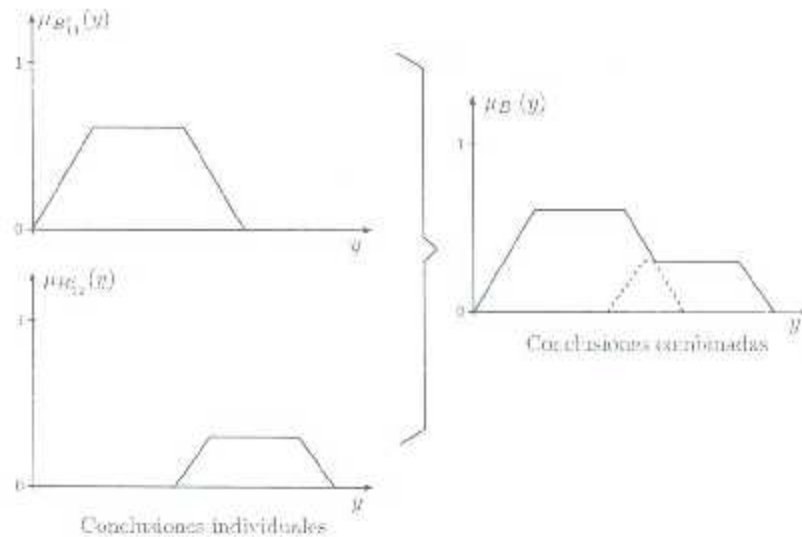


Figura 3.8: Combinación de consecuentes

3.6. Defusificación

La defusificación transforma la conclusión o conjunto difuso obtenido de la máquina inferencial *i.e.*, $\mu_{B'}(y)$ en un valor real (o valor *crisp*); es decir, realiza un mapeo de un conjunto difuso $B' \in V \subset \mathbb{R}$ a un valor real $y^* \in V \subset \mathbb{R}$. Se puede decir que es el proceso inverso de la fusificación ya que el defusificador debe obtener un punto en V que mejor represente a B' . En la literatura existen una variedad de estrategias de defusificación, entre los que se encuentran: centro de área o gravedad, promedio de centros, centro de sumas, centro del área más grande, defusificador máximo, primero del máximo, último del máximo; entre otras.

La técnica más ampliamente utilizada en aplicaciones de control en tiempo real es el método promedio de centros, debido a su bajo costo computacional y poco tiempo de cálculo y a su propiedad de continuidad, lo que significa que un pequeño cambio en la salida de la máquina de inferencia difusa no resulta en un cambio grande en la salida defusificada.

3.6.1. Promedio de centros

Es una aproximación del método centro de gravedad y es un promedio de centros de los M conjuntos difusos de que se compone $\mu_{B^*}(y)$. Para el caso en que se usa la base de reglas del tipo (3.2) este método está definido por

$$y^*(\mathbf{x}^*) = \frac{\sum_{l_1=1}^{p_1} \cdots \sum_{l_n=1}^{p_n} \bar{y}^{l_1 \cdots l_n} w_{l_1 \cdots l_n}(\mathbf{x}^*)}{\sum_{l_1=1}^{p_1} \cdots \sum_{l_n=1}^{p_n} w_{l_1 \cdots l_n}(\mathbf{x}^*)} \quad (3.6)$$

donde $\bar{y}^{l_1 \cdots l_n}$ es el centro del $l_1 \cdots l_n$ -ésimo conjunto difuso del que se compone la salida a defusificar, y $w_{l_1 \cdots l_n}$ es su altura mientras que $\mathbf{x}^* = (x_1^*, x_2^*, \dots, x_n^*)^T \in \mathbb{R}^n$ es el conjunto de valores *crisp* de entrada. Como puede verse, en este método no importa la forma del conjunto de salida, pues sólo se toma en cuenta el centro y altura del conjunto, por esta razón es muy rápido en tiempo real y con una buena aproximación.

3.7. Sistema difuso

Se ha mencionado que la máquina de inferencia producto (3.3) es la más utilizada en control automático debido a su sencillez y bajo tiempo de cálculo; igualmente se ha mencionado que el defusificador promedio de centros (3.6) es también el más usado por las mismas razones; entonces, es posible combinar ambas ecuaciones para obtener un poderoso algoritmo que puede ser implementado fácilmente en sistemas de control difuso en tiempo real.

Considérese un conjunto de reglas difusas con la forma (3.2), esta forma asegura que todas las posibles combinaciones entre conjunto son tomadas en cuenta. Si específicamente se emplean funciones de membresía de entrada del tipo triangular y de salida del tipo impulsiva (*singleton*), inferencia producto y defusificador promedio de centros, la salida real de un sistema difuso dado puede ser escrita como

$$y = \frac{\sum_{l_1=1}^{p_1} \sum_{l_2=1}^{p_2} \cdots \sum_{l_n=1}^{p_n} \bar{y}^{l_1 l_2 \cdots l_n} \left(\prod_{i=1}^n \mu_{A_i^{l_i}}(x_i) \right)}{\sum_{l_1=1}^{p_1} \sum_{l_2=1}^{p_2} \cdots \sum_{l_n=1}^{p_n} \left(\prod_{i=1}^n \mu_{A_i^{l_i}}(x_i) \right)} \quad (3.7)$$

donde $\bar{y}^{l_1 l_2 \cdots l_n}$ es el $l_1 \cdots l_n$ -ésimo conjunto difuso de la conclusión dada por la máquina inferencial.

Una de las contribuciones principales de los sistemas difusos como el de la ecuación (3.7), el cual representa un sistema basado en reglas, es proveer un algoritmo sistemático para transformar un conjunto de reglas lingüísticas del tipo *IF-THEN* en un mapeo no lineal. Esta característica es muy importante pues encuentra aplicación en sistemas de control complejos en los que no es sencillo calcular un modelo matemático para aplicar controladores clásicos basados en el modelo.

3.8. Aproximador universal

En la Sección 3.7 se mostró un tipo de sistema difuso que puede ser escrito como fórmulas no lineales compactas. Por un lado, estas fórmulas compactas simplifican el cálculo de los sistemas difusos; por otro lado, brindan la oportunidad de analizar el sistema a mayor detalle. Una propiedad interesante a estudiar es la capacidad de los sistemas lógico difusos de aproximar funciones no lineales, dado que si un sistema difuso puede aproximar cualquier tipo de función no lineal con una precisión arbitraria, entonces éstos pueden ser útiles en una amplia variedad de aplicaciones, incluyendo aplicación en control automático.

El siguiente teorema muestra que un sistema difuso, con máquina de inferencia producto, fusificador singleton, defusificador promedio de centros, tiene una capacidad de aproximación universal, es decir, esta clase de sistemas lógico difusos son *aproximadores universales*.

Teorema 3.1 *Supóngase que el universo de discurso U es un conjunto compacto en \mathbb{R}^n . Entonces, para cualquier función real continua $g(\mathbf{x})$ en U y un $\epsilon > 0$ arbitrario, existe un sistema difuso $f(\mathbf{x})$ en la forma de (3.7) tal que*

$$\sup_{\mathbf{x} \in U} |f(\mathbf{x}) - g(\mathbf{x})| < \epsilon \quad (3.8)$$

Prueba: Ver [24]

3.9. Sistemas de control basados en lógica difusa

El éxito de los controladores por lógica difusa está dado principalmente por la capacidad que tienen para lidiar con el conocimiento representado en una forma lingüística o cualitativa, en vez de una representación en el marco matemático convencional. El control difuso es una mezcla de varias disciplinas como son los sistemas expertos tradicionales basados en reglas, la teoría de conjuntos difusos y la teoría de control. Mientras la mayoría de los controladores tradicionales se derivan modelando cuantitativamente y matemáticamente el proceso a controlar, un controlador lógico difuso (CLD) se deriva directamente del conocimiento heurístico y cualitativo que un diseñador experto tiene acerca de la planta a controlar; por lo tanto, un CLD convierte una estrategia de control lingüístico en una estrategia de control automático. El conocimiento es expresado a través de un conjunto de reglas lingüísticas del tipo *IF antecedente THEN consecuente* que puede ser interpretado desde un punto de vista de control automático como *IF condición THEN acción*. Lo anterior puede también ser interpretado como una forma de inferir una conclusión (acción) respecto a una entrada determinada.

3.9.1. Características

Enseguida se exponen algunas de las características más importantes e interesantes que poseen los sistemas de control basados en lógica difusa.

- Los controladores difusos son tolerantes a datos imprecisos, incertidumbres y parámetros no modelados, es decir, presentan un comportamiento robusto.
- Pueden modelar o aproximar sistemas no lineales; de hecho, la solución de control para sistemas no lineales algunas veces es mucho más sencilla que con otras estrategias de control al no requerir ecuaciones diferenciales para calcular la ley de control.
- Permiten la combinación con otras técnicas de control clásicas o no lineales de manera que pueden coexistir en un mismo proceso.

- los sistemas difusos pueden ser empleados en sistemas de control para ejercer la acción de control directamente, o bien para ajustar los parámetros de la ley de control.
- Pueden basarse en conocimiento obtenido a partir de la experiencia del diseñador empleando un lenguaje natural lo que permite una fácil incorporación de dicho conocimiento.
- Sin embargo, los controladores basados en lógica difusa requieren mayor tiempo de cálculo, lo que constituye una de sus principales desventajas, por lo que se debe considerar la capacidad de los sistemas digitales en que se implementan.
- Otra aspecto que puede ser visto como una desventaja es que este tipo de controladores poseen numerosos parámetros de configuración, lo que en ocasiones vuelve la sintonización del controlador una tarea laboriosa.

3.9.2. Estructura

Los controladores difusos pueden aplicarse a alto o bajo nivel. En el primer caso se trata de autosintonizadores difusos que calculan las ganancias de algún tipo de controlador convencional al cual se le permite variar sus parámetros de acuerdo al error de estado o la salida. En el segundo caso, el controlador difuso se encuentra en la trayectoria directa ejerciendo la acción de control sobre la planta a controlar; por esta razón se les conoce como controladores difusos directos.

Por otro lado, los controladores lógicos difusos, como todo controlador, ofrecen una salida de manipulación calculada a partir de las variables de entrada, que pueden ser las salidas de la planta o errores del sistema en lazo cerrado y como ya se ha mencionado, este tipo de algoritmos permiten la combinación con otras técnicas de control clásicas o no lineales, adaptables, entre otras.

Sin importar cuales son las variables de entrada, o si el controlador difuso se aplica a nivel alto o bajo, o si éste comparte la acción de control con algún otro algoritmo, un controlador difuso siempre está constituido de las siguientes partes principales (Figura 3.9): a) fusificador, b) base de reglas y mecanismo de inferencia difusa y c) defusificador, procesos que se han abordado en secciones previas de este mismo capítulo.



Figura 3.9: Diagrama a bloques de la estructura general de un sistema difuso.

Por otro lado, para diseñar un CLD se requiere contar con las siguientes especificaciones:

a) *Fusificador*

- Funciones de pertenencia de entrada,
- Método de fusificación.

b) *Base de reglas y máquina de inferencia difusa*

- Base de reglas,
- Conexión entre proposiciones de antecedentes,
- Método de inferencia.

c) *Defusificador*

- Funciones de membresía de salida,
- Método de defusificación.

Nótese que la base de datos es un auxiliar que contiene los tipos de funciones de membresía de las entradas y la salida, la cantidad de ellas, así como los parámetros que definen sus formas (triangulares, Gaussianas, impulsivas entre otras).

Capítulo 4

Control descentralizado difuso adaptable

En este capítulo se presenta el desarrollo teórico del controlador propuesto. En primer lugar se describe el modelo y la dinámica descentralizada de robots manipuladores. Posteriormente se brinda una breve descripción del sistema difuso adaptable, sus propiedades y estructura, luego se aborda concretamente la ley de control descentralizada difusa adaptable que se ha diseñado; sus elementos son descritos a detalle y finalmente se muestra el análisis basado en la teoría de Lyapunov, que garantiza la estabilidad del sistema total así como el cumplimiento del objetivo de control de movimiento de robots manipuladores.

4.1. Sistema descentralizado

Considérese el modelo dinámico de un robot manipulador serial de n eslabones rígidos, descrito por las siguientes ecuaciones de movimiento [8]

$$B(q)\ddot{q} + C(q, \dot{q})\dot{q} + g(q) + f(\dot{q}) = \tau \quad (4.1)$$

En lo subsecuente se supondrá que tanto las referencias deseadas de posición $q_d = q_d(t) \in \mathbb{R}^n$ como sus derivadas temporales \dot{q}_d, \ddot{q}_d están acotadas. Ahora se denota el error de posición articular como $\tilde{q} = q_d - q$ y por tanto el error de velocidad articular $\dot{\tilde{q}} = \dot{q}_d - \dot{q}$, a fin de presentar el error generalizado de seguimiento [20]

$$s = \Lambda \tilde{q} + \dot{\tilde{q}} \quad (4.2)$$

que bien puede ser visto como una superficie de deslizamiento [9] donde $\Lambda = \text{diag}\{\lambda_1, \dots, \lambda_n\} > 0$ es una matriz constante. Además se tiene que $\dot{s} = \Lambda \dot{\bar{q}} + \ddot{\bar{q}}$, y $\ddot{q} = \ddot{q}_d - \ddot{\bar{q}}$ con lo que (4.1) puede ser reescrita como

$$\begin{aligned}\tau &= B(q)[\ddot{q}_d + \Lambda \dot{\bar{q}} - \dot{s}] + C(q, \dot{q})[\dot{q}_d + \Lambda \bar{q} - s] \\ &\quad + g(q) + f(\dot{q}) \\ \tau &= B(q)[\ddot{q}_d + \Lambda \dot{\bar{q}}] + C(q, \dot{q})[\dot{q}_d + \Lambda \bar{q}] \\ &\quad + g(q) + f(\dot{q}) - B(q)\dot{s} - C(q, \dot{q})s\end{aligned}$$

o bien

$$B(q)\dot{s} + C(q, \dot{q})s = -\tau + \chi \quad (4.3)$$

que representa la ecuación dinámica del error con

$$\chi = B(q)[\ddot{q}_d + \Lambda \dot{\bar{q}}] + C(q, \dot{q})[\dot{q}_d + \Lambda \bar{q}] + g(q) + f(\dot{q}) \quad (4.4)$$

Ahora, se busca acotar superiormente al vector χ en (4.4), para lo cual se hace

$$\begin{aligned}\|\chi\| &= \|B(q)[\ddot{q}_d + \Lambda \dot{\bar{q}}] + C(q, \dot{q})[\dot{q}_d + \Lambda \bar{q}] + g(q) + f(\dot{q})\| \\ &\leq \|B(q)[\ddot{q}_d + \Lambda \dot{\bar{q}}]\| + \|C(q, \dot{q})[\dot{q}_d + \Lambda \bar{q}]\| + \|g(q)\| + \|f(\dot{q})\| \\ &\leq \|B(q)\ddot{q}_d\| + \|B(q)\Lambda \dot{\bar{q}}\| + \|C(q, \dot{q})\dot{q}_d\| + \|C(q, \dot{q})\Lambda \bar{q}\| \\ &\quad + k' + c_1 + c_2 \|\dot{q}\|\end{aligned} \quad (4.5)$$

donde se han empleado las propiedades 2.4 y 2.5 del modelo dinámico descritas en el Capítulo 2. Dado que las velocidades y aceleraciones deseadas se suponen acotadas i.e., $\dot{q}_d, \ddot{q}_d \in \mathcal{L}_\infty^n$ se puede establecer que $\|\dot{q}_d\| \leq \zeta_1$, $\|\ddot{q}_d\| \leq \zeta_2$, para algunas $\zeta_1, \zeta_2 > 0$. Luego de (4.5) se tiene

$$\begin{aligned}\|\chi\| &\leq \|B(q)\|\|\ddot{q}_d\| + \|B(q)\|\|\Lambda \dot{\bar{q}}\| + k_{c_1} \|\dot{q}\|\|\dot{q}_d\| + k_{c_1} \|\dot{q}\|\|\Lambda \bar{q}\| \\ &\quad + k' + c_1 + c_2 \|\dot{q}\| \\ &\leq b'_2 \zeta_2 + b'_2 \|\Lambda\|\|\dot{\bar{q}}\| + k_{c_1} \zeta_1 \|\dot{q}\| + k_{c_1} \|\Lambda\|\|\dot{\bar{q}}\|\|\bar{q}\| \\ &\quad + k' + c_1 + c_2 \|\dot{q}\|\end{aligned} \quad (4.6)$$

para lo que se han empleado las propiedades 2.1 y 2.3 del modelo dinámico de manipuladores. Dada la definición del error de velocidad se tiene que $\dot{\bar{q}} = \dot{q}_d - \dot{\bar{q}}$ y

en consecuencia $\|\dot{\tilde{\mathbf{q}}}\| \leq \|\dot{\tilde{\mathbf{q}}_d}\| + \|\dot{\tilde{\mathbf{q}}}\|$, así (4.6) queda como

$$\begin{aligned} \|\chi\| &\leq b'_2 \zeta_2 - b'_2 \|\Lambda\| \|\dot{\tilde{\mathbf{q}}}\| + k_{C_1} \zeta_1 \|\dot{\tilde{\mathbf{q}}_d}\| + k_{C_1} \zeta_1 \|\dot{\tilde{\mathbf{q}}}\| + k_{C_1} \|\Lambda\| \|\dot{\tilde{\mathbf{q}}_d}\| \|\tilde{\mathbf{q}}\| \\ &\quad + k_{C_1} \|\Lambda\| \|\dot{\tilde{\mathbf{q}}}\| \|\tilde{\mathbf{q}}\| + k' + c_1 + c_2 \|\dot{\tilde{\mathbf{q}}_d}\| - c_2 \|\dot{\tilde{\mathbf{q}}}\| \\ &\leq b'_2 \zeta_2 - b'_2 \|\Lambda\| \|\dot{\tilde{\mathbf{q}}}\| + k_{C_1} \zeta_1^2 + k_{C_1} \zeta_1 \|\dot{\tilde{\mathbf{q}}}\| + k_{C_1} \zeta_1 \|\Lambda\| \|\tilde{\mathbf{q}}\| \\ &\quad + k_{C_1} \|\Lambda\| \|\dot{\tilde{\mathbf{q}}}\| \|\tilde{\mathbf{q}}\| + k' + c_1 + c_2 \zeta_1 + c_2 \|\dot{\tilde{\mathbf{q}}}\|, \end{aligned}$$

o bien

$$\|\chi\| \leq \alpha'_1 + \alpha'_2 \|\tilde{\mathbf{q}}\| + \alpha'_3 \|\dot{\tilde{\mathbf{q}}}\| + \alpha'_4 \|\tilde{\mathbf{q}}\| \|\dot{\tilde{\mathbf{q}}}\|, \quad (4.7)$$

donde

$$\begin{aligned} \alpha'_1 &= b'_2 \zeta_2 + k_{C_1} \zeta_1^2 + k' + c_1 + c_2 \zeta_1 \\ \alpha'_2 &= k_{C_1} \zeta_1 \|\Lambda\| \\ \alpha'_3 &= b'_2 \|\Lambda\| + k_{C_1} \zeta_1 + c_2 \\ \alpha'_4 &= k_{C_1} \|\Lambda\| \end{aligned}$$

son constantes positivas determinadas por los parámetros dinámicos del robot y por las trayectorias de referencia dadas.

El siguiente lema, el cual es introducido en forma original, brinda una cota superior de $\|\chi\|$ en términos de $\|\mathbf{s}\|$.

Lema 4.1 *Para la norma del vector χ expresado en (4.4), existen constantes $\alpha_i > 0$ con $i = 1, 2, 3$ tal que*

$$\|\chi\| \leq \alpha_1 + \alpha_2 \|\mathbf{s}\| + \alpha_3 \|\mathbf{s}\|^2 \quad (4.8)$$

donde

$$\begin{aligned} \alpha_1 &= \alpha'_1 + \alpha'_2 \bar{a} + \alpha'_3 \bar{c} + \alpha'_4 \bar{a} \bar{c} \\ \alpha_2 &= \alpha'_2 \bar{b} + \alpha'_3 \bar{d} + \alpha'_4 (\bar{a} \bar{d} + \bar{b} \bar{c}) \\ \alpha_3 &= \alpha'_4 \bar{b} \bar{d} \end{aligned}$$

Prueba: Nótese que (4.2) puede ser tomada como un arreglo de ecuaciones diferenciales lineales de primer orden donde $\dot{\tilde{\mathbf{q}}} = -\Lambda \tilde{\mathbf{q}} + \mathbf{s} = \begin{bmatrix} \dot{\tilde{q}}_1 \\ \vdots \\ \dot{\tilde{q}}_n \end{bmatrix}$, con $\dot{\tilde{q}}_i = -\lambda_i \tilde{q}_i + s_i$ para $i = 1, \dots, n$ cuya solución general está dada por

$$\tilde{q}_i(t) = \tilde{q}_{i0}e^{-\lambda_i t} + \int_0^t e^{-\lambda_i(t-\tau)} s_i(\tau) d\tau \quad (4.9)$$

o bien,

$$\tilde{q}_i(t) = \tilde{q}_{i0}e^{-\lambda_i t} + e^{-\lambda_i t} * s_i(t) \quad (4.10)$$

donde $*$ es el operador de convolución y \tilde{q}_{i0} es la condición inicial de $\tilde{q}_i(t)$ en el tiempo inicial t_0 . Por otro lado, se tiene que la norma del vector $\tilde{\mathbf{q}}$ satisface que

$$\|\tilde{\mathbf{q}}\| \leq (|\tilde{q}_1| + \dots + |\tilde{q}_n|) \quad (4.11)$$

por lo cual se aplica el valor absoluto a ambos lados de la ecuación (4.10) y empleando la desigualdad de Young se obtiene una cota superior sobre ésta, expresada de la siguiente manera

$$|\tilde{q}_i| \leq a_i + b_i |s_i|$$

donde $a_i \geq |\tilde{q}_{i0}e^{-\lambda_i t}|$ y $b_i \geq |e^{-\lambda_i t}|$ siendo esto posible debido al hecho de que todos los elementos de la matriz \mathbf{A} son mayores que cero. Entonces (4.11) puede ser reescrita como

$$\|\tilde{\mathbf{q}}\| \leq \sum_{i=1}^n a_i + \sum_{i=1}^n b_i |s_i|$$

o bien, empleando la desigualdad $\|\mathbf{s}\|_1 \leq \sqrt{n}\|\mathbf{s}\|$

$$\|\tilde{\mathbf{q}}\| \leq \bar{a} + \bar{b}\|\mathbf{s}\| \quad (4.12)$$

donde $\bar{a} = a_{i_{\max}} n$, $\bar{b} = b_{i_{\max}} \sqrt{n}$. Luego la norma del error de velocidad puede ser acotada por

$$\|\dot{\tilde{\mathbf{q}}}\| \leq \|\mathbf{A}\|\|\tilde{\mathbf{q}}\| + \|\mathbf{s}\|$$

y empleando el resultado obtenido en (4.12) se llega finalmente a

$$\|\dot{\tilde{\mathbf{q}}}\| \leq \bar{c} + \bar{d}\|\mathbf{s}\| \quad (4.13)$$

donde $\bar{c} = \|\mathbf{A}\|\bar{a}$, $\bar{d} = \|\mathbf{A}\|\bar{b} + 1$. La sustitución de (4.12) y (4.13) en (4.7) concluye la prueba.

Para ajustar el sistema dentro de la configuración de control descentralizado adoptada por la mayoría de los manipuladores, se separa la dinámica del error en (4.3) y (4.1) en

n subsistemas dados por

$$\sum_{j=1}^n b_{ij}(\mathbf{q})\dot{s}_j + \sum_{j=1}^n c_{ij}(\mathbf{q}, \dot{\mathbf{q}})s_j + \tau_i = \chi_i \quad (4.14)$$

donde

$$\begin{aligned} \chi_i = & \sum_{j=1}^n b_{ij}(\mathbf{q})[\dot{q}_{d_j} + \lambda_j \tilde{q}_j] + \sum_{j=1}^n c_{ij}(\mathbf{q}, \dot{\mathbf{q}})[\dot{q}_{d_j} + \lambda_j \tilde{q}_j] \\ & + g_i(\mathbf{q}) + f_i(\dot{\mathbf{q}}) \end{aligned} \quad (4.15)$$

Así, se tiene que χ_i en (4.15) representa los términos debidos a las interconexiones entre eslabones actuando sobre el i -ésimo subsistema. De hecho (ver [13]), χ_i puede también ser expresado como

$$\chi_i = \eta_i(\mathbf{x}_i) + \delta'(\mathbf{q}, \dot{\mathbf{q}}, \mathbf{v}, \dot{\mathbf{v}}) \quad (4.16)$$

con $\mathbf{v} = \dot{\mathbf{q}}_d - \Lambda \tilde{\mathbf{q}}$, $\mathbf{x}_i = \begin{bmatrix} q_i & \dot{q}_i \end{bmatrix}^T$ y donde $\eta_i(\mathbf{x}_i)$ representa la porción local relacionada solamente a la posición y velocidad articular y a los parámetros del i -ésimo subsistema, mientras que $\delta'(\mathbf{q}, \dot{\mathbf{q}}, \mathbf{v}, \dot{\mathbf{v}})$ es la porción global (relacionada al resto de los subsistemas) de las interconexiones, y para fines prácticos de control descentralizado puede ser considerado como un término de perturbaciones. Finalmente, es posible deducir que existen también constantes positivas $\beta_{i_1}, \beta_{i_2}, \beta_{i_3}$ tal que χ_i está acotado de la misma manera que en (4.8) i.e.,

$$|\chi_i| \leq \beta_{i_1} + \beta_{i_2} \|\mathbf{s}\| + \beta_{i_3} \|\mathbf{s}\|^2 \quad (4.17)$$

Por otro lado, el problema de control de movimiento para robots manipuladores radica en encontrar una función vectorial $\boldsymbol{\tau}$ tal que las posiciones \mathbf{q} asociadas a las coordenadas articulares del robot sigan con precisión a las trayectorias deseadas \mathbf{q}_d .

En términos más formales, el *objetivo del control de movimiento* consiste en encontrar $\boldsymbol{\tau}$ tal que

$$\lim_{t \rightarrow \infty} \tilde{\mathbf{q}}(t) = \mathbf{0} \quad (4.18)$$

4.2. Sistema difuso adaptable

Considérese un sistema difuso $\hat{\eta}_i(\mathbf{x}_i)$ similar al dado en (3.7) para el caso de dos entradas, i.e., $\mathbf{x}_i = \begin{bmatrix} x_{i1} & x_{i2} \end{bmatrix}^T = \begin{bmatrix} q_i & \dot{q}_i \end{bmatrix}^T \in U_i \subset \mathbb{R}^2$ y una salida $y_i \in V_i \subset \mathbb{R}$, en el

que para cada variable de entrada x_{ij} ($j = 1, 2$) se definen p_j conjuntos difusos $A_{ij}^{l_j}$ ($l_j = 1, 2, \dots, p_j$), cada uno caracterizado por una función de membresía $\mu_{A_{ij}^{l_j}}(x_{ij})$, y para cada una de las funciones de membresía de salida $\bar{y}_i^{l_1 l_2}$ corresponde una función del tipo impulsiva (*singleton*).

Dicho sistema es construido a partir del conjunto de $\prod_{j=1}^2 p_j$ reglas difusas *IF-THEN* de la forma

$$\text{IF } x_{i1} \text{ is } A_{i1}^{l_1} \text{ AND } x_{i2} \text{ is } A_{i2}^{l_2} \text{ THEN } \hat{\eta}_i(\mathbf{x}_i) \text{ is } \bar{y}_i^{l_1 l_2}$$

Además, si específicamente se emplean funciones de membresía de entrada del tipo triangular, inferencia producto y defusificador promedio de centros, el sistema difuso $\hat{\eta}_i(\mathbf{x}_i)$ puede ser representado como

$$\hat{\eta}_i(\mathbf{x}_i) = \frac{\sum_{l_1=1}^{p_1} \sum_{l_2=1}^{p_2} \bar{y}_i^{l_1 l_2} \left(\prod_{j=1}^2 \mu_{A_{ij}^{l_j}}(x_{ij}) \right)}{\sum_{l_1=1}^{p_1} \sum_{l_2=1}^{p_2} \left(\prod_{j=1}^2 \mu_{A_{ij}^{l_j}}(x_{ij}) \right)} \quad (4.19)$$

Sean $\bar{y}_i^{l_1 l_2}$ los parámetros libres (parámetros ajustables) los cuales pueden ser contenidos dentro de un vector $\theta_i \in \mathbb{R}^{\prod_{j=1}^2 p_j}$, i.e., $\theta_i = \left[\bar{y}_i^{l_1 l_2} \dots \bar{y}_i^{l_1 l_2} \right]^T$, entonces (4.19) puede ser reescrita como

$$\hat{\eta}_i(\mathbf{x}_i | \theta_i) = \theta_i^T \xi_i(\mathbf{x}_i) \quad (4.20)$$

donde $\xi_i(\mathbf{x}_i) \in \mathbb{R}^{\prod_{j=1}^2 p_j}$ es el vector regresor cuyos $l_1 l_2$ -ésimos elementos están dados por

$$\xi_{l_1 l_2}(\mathbf{x}_i) = \frac{\prod_{j=1}^2 \mu_{A_{ij}^{l_j}}(x_{ij})}{\sum_{l_1=1}^{p_1} \sum_{l_2=1}^{p_2} \left(\prod_{j=1}^2 \mu_{A_{ij}^{l_j}}(x_{ij}) \right)}$$

Empleando el resultado obtenido en [24], se tiene que para cualquier función real continua $\eta_i(\mathbf{x}_i)$ definida en un conjunto compacto $U_i \subset \mathbb{R}^n$ con un $\epsilon_i > 0$ arbitrario, existe un sistema difuso (4.19) tal que

$$\sup_{\mathbf{x}_i \in U_i} |\eta_i(\mathbf{x}_i) - \hat{\eta}_i(\mathbf{x}_i)| < \epsilon_i \quad (4.21)$$

4.3. Diseño y estructura del controlador

A fin de satisfacer el objetivo de control de movimiento de robots manipuladores, se propone una ley de control descentralizada difusa adaptable para el error dinámico de seguimiento dado por (4.14) y (4.16) la cual es expresada por

$$\tau_i = k_i s_i + \rho_i s_i^3 + \hat{\eta}_i(\mathbf{x}_i | \hat{\theta}_i) + \sigma_i \text{sign}(s_i) \quad (4.22)$$

donde k_i , ρ_i , σ_i son constantes positivas (ganancias de control) y $\hat{\eta}_i(\mathbf{x}_i | \hat{\theta}_i) = \hat{\theta}_i^T \xi_i(\mathbf{x}_i)$ es la acción difusa adaptable que estima la porción local de la dinámica en (4.16), y cuyos parámetros ajustables $\hat{\theta}_i$ son actualizados empleando la ley de adaptación

$$\dot{\hat{\theta}}_i = -\gamma_i \xi_i(\mathbf{x}_i) s_i \quad (4.23)$$

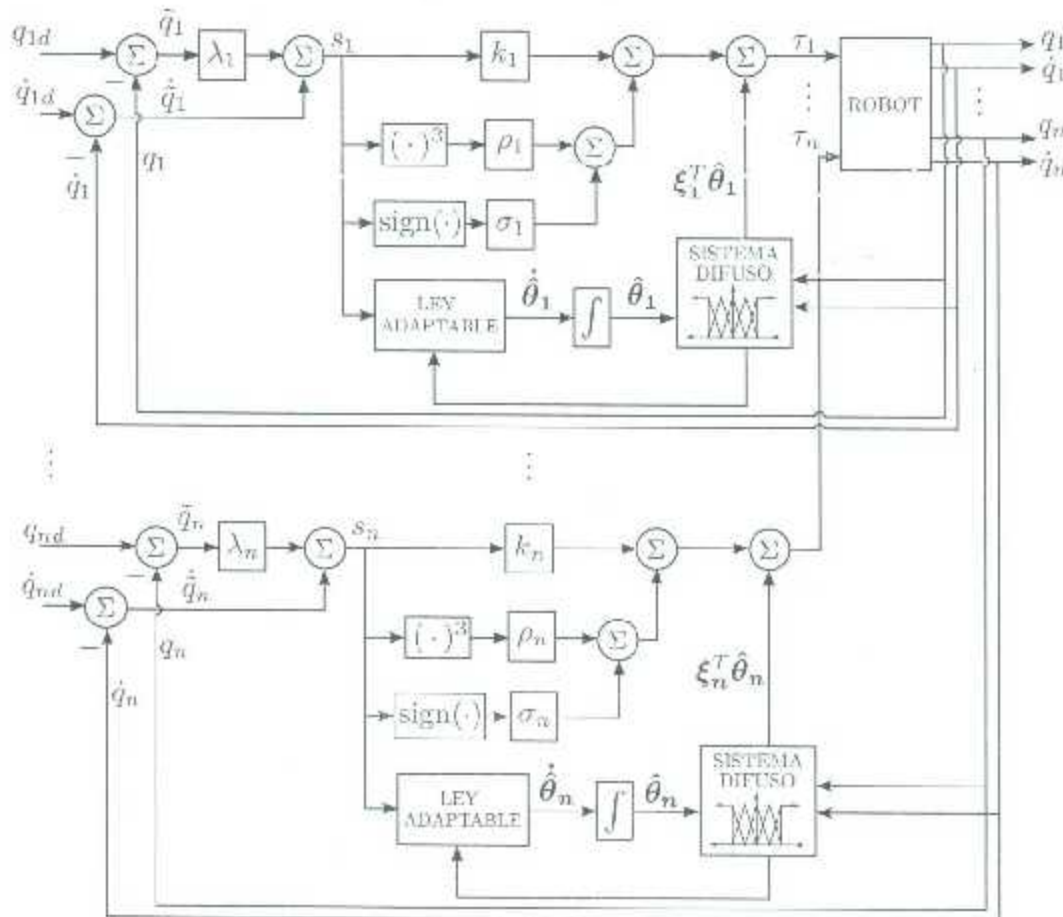


Figura 4.1: Diagrama a bloques del controlador propuesto.

conde $\gamma_i > 0$ es una ganancia de adaptación que adelanta o retrasa la estimación de los parámetros libres $\hat{\theta}_i$ del sistema difuso $\hat{\eta}_i$, el cual cabe remarcar que sólo emplea como entrada las variables locales q_i, \dot{q}_i del i -ésimo subsistema. La figura 4.1, muestra el diagrama a bloques del controlador difuso adaptable descentralizado dado por (4.22).

Por otro lado, el término $k_i s_i$ es una simple acción de control PD, mientras que los términos $\rho_i s_i^3$ y $\sigma_i \text{sign}(s_i)$ representan una acción de amortiguamiento y un control por modos deslizantes ideal respectivamente, que suprimen los efectos de incertidumbres despreciadas y perturbaciones resumidas en χ_i , además de tener un rol importante para asegurar la estabilidad del sistema, tal como se presentará más adelante.

4.3.1. Ecuación de lazo cerrado

Considérese el sistema difuso en (4.20) con un vector de parámetros $\theta_i \in \Omega_i$ como el mejor aproximador de la porción local de la dinámica χ_i en (4.16), cuyo error de aproximación definido como

$$\omega_i = \eta_i(\mathbf{x}_i) - \hat{\eta}_i(\mathbf{x}_i|\theta_i)$$

está acotado y es arbitrariamente pequeño en vista de (4.21); bajo estas condiciones (4.16) puede reescribirse como

$$\begin{aligned} \chi_i &= \eta_i(\mathbf{x}_i) + \delta'(\mathbf{q}, \dot{\mathbf{q}}, \mathbf{v}, \dot{\mathbf{v}}) + \hat{\eta}_i(\mathbf{x}_i|\theta_i) - \hat{\eta}_i(\mathbf{x}_i|\theta_i) \\ &= \hat{\eta}_i(\mathbf{x}_i|\theta_i) + \delta'(\mathbf{q}, \dot{\mathbf{q}}, \mathbf{v}, \dot{\mathbf{v}}) + \omega_i \\ &= \hat{\eta}_i(\mathbf{x}_i|\theta_i) + \delta(\mathbf{q}, \dot{\mathbf{q}}, \mathbf{v}, \dot{\mathbf{v}}) \end{aligned} \quad (4.24)$$

conde $\delta(\mathbf{q}, \dot{\mathbf{q}}, \mathbf{v}, \dot{\mathbf{v}}) = \delta'(\mathbf{q}, \dot{\mathbf{q}}, \mathbf{v}, \dot{\mathbf{v}}) + \omega_i$ será considerado como un término de perturbaciones y además se satisface que

$$\theta_i = \arg \min_{\theta_i \in \Omega_i} \left[\sup_{\mathbf{x}_i \in U_i} |\eta_i(\mathbf{x}_i) - \hat{\eta}_i(\mathbf{x}_i|\theta_i)| \right].$$

Sea $\delta = [\delta_1 \ \cdots \ \delta_n]^T$ el vector de perturbación total, el cual de acuerdo a [13] puede ser acotado bajo una tasa similar a la de χ en (4.8), *i.e.*,

$$\|\delta\| \leq \beta_1 + \beta_2 \|s\| + \beta_3 \|s\|^2 \quad (4.25)$$

para algunas constantes $\beta_1, \beta_2, \beta_3 > 0$. Ahora sustituyendo (4.22) en (4.14) y tomando en cuenta (4.24) se obtiene la ecuación en lazo cerrado del sistema

$$\begin{aligned}
 & \sum_{j=1}^n b_{ij}(\mathbf{q})\dot{s}_j + \sum_{j=1}^n c_{ij}(\mathbf{q}, \dot{\mathbf{q}})s_j + k_i s_i + \rho_i s_i^3 + \sigma_i \text{sign}(s_i) \\
 &= \hat{\eta}_i(\mathbf{x}_i|\boldsymbol{\theta}_i) - \hat{\eta}_i(\mathbf{x}_i|\hat{\boldsymbol{\theta}}_i) - \delta(\mathbf{q}, \dot{\mathbf{q}}, \mathbf{v}, \dot{\mathbf{v}}) \\
 &= \boldsymbol{\xi}_i(\mathbf{x}_i)^T [\boldsymbol{\theta}_i - \hat{\boldsymbol{\theta}}_i] - \delta(\mathbf{q}, \dot{\mathbf{q}}, \mathbf{v}, \dot{\mathbf{v}}) \\
 &= \boldsymbol{\xi}_i(\mathbf{x}_i)^T \tilde{\boldsymbol{\theta}}_i(t) + \delta(\mathbf{q}, \dot{\mathbf{q}}, \mathbf{v}, \dot{\mathbf{v}})
 \end{aligned} \tag{4.26}$$

con $\tilde{\boldsymbol{\theta}}_i(t) = \boldsymbol{\theta}_i - \hat{\boldsymbol{\theta}}_i$ y que puede ser expresada vectorialmente como

$$\mathbf{B}(\mathbf{q})\dot{\mathbf{s}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\mathbf{s} + \mathbf{K}\mathbf{s} + \boldsymbol{\tau}_e + \boldsymbol{\Sigma}\text{sign}(\mathbf{s}) = \boldsymbol{\xi}^T \tilde{\boldsymbol{\theta}}(t) + \boldsymbol{\delta} \tag{4.27}$$

donde $\mathbf{K} = \text{diag}\{k_1, \dots, k_n\}$, $\boldsymbol{\tau}_e = [\rho_1 s_1^3 \ \dots \ \rho_n s_n^3]^T$, $\boldsymbol{\Sigma} = \text{diag}\{\sigma_1, \dots, \sigma_n\}$, $\text{sign}(\mathbf{s}) = [\text{sign}(s_1) \ \dots \ \text{sign}(s_n)]^T$ y finalmente $\boldsymbol{\xi} = \text{diag}\{\boldsymbol{\xi}_1(\mathbf{x}_1), \dots, \boldsymbol{\xi}_n(\mathbf{x}_n)\}$.

4.3.2. Análisis de estabilidad

A partir de un diseño por síntesis de Lyapunov, se muestra que los estados del sistema en lazo cerrado (4.27), i.e. el error generalizado de seguimiento $\mathbf{s} = [s_1 \ \dots \ s_n]^T$ y el error de estimación difuso $\tilde{\boldsymbol{\theta}} = [\tilde{\boldsymbol{\theta}}_1 \ \dots \ \tilde{\boldsymbol{\theta}}_n]^T$ están acotados, y que además el límite del error de posición $\tilde{\mathbf{q}}$ tiende a cero cuando el tiempo tiende a infinito, es decir se satisface el objetivo de control.

A fin de llevar a cabo el análisis de estabilidad se propone la siguiente función candidata de Lyapunov

$$V(\mathbf{s}, \tilde{\boldsymbol{\theta}}) = \frac{1}{2} \mathbf{s}^T \mathbf{B}(\mathbf{q})\mathbf{s} + \frac{1}{2} \tilde{\boldsymbol{\theta}}^T \boldsymbol{\Gamma}^{-1} \tilde{\boldsymbol{\theta}} \tag{4.28}$$

la cuál es definida positiva globalmente y radialmente desacotada debido a las propiedades de la matriz de inercias y a que $\boldsymbol{\Gamma} = \text{diag}\{\gamma_1 \mathbf{I}, \dots, \gamma_n \mathbf{I}\} > 0$ por diseño. Nótese además que $V(\mathbf{s}, \tilde{\boldsymbol{\theta}})$ satisface

$$\lambda_{\min}\{\mathbf{B}\} \|\mathbf{s}\|^2 + \lambda_{\min}\{\boldsymbol{\Gamma}^{-1}\} \|\tilde{\boldsymbol{\theta}}\|^2 \leq V(\mathbf{s}, \tilde{\boldsymbol{\theta}}) \leq \lambda_{\max}\{\mathbf{B}\} \|\mathbf{s}\|^2 + \lambda_{\max}\{\boldsymbol{\Gamma}^{-1}\} \|\tilde{\boldsymbol{\theta}}\|^2$$

Luego, la derivada temporal de $V(\mathbf{s}, \tilde{\boldsymbol{\theta}})$ evaluada a lo largo de las trayectorias del sistema en lazo cerrado en (4.27) da como resultado

$$\begin{aligned}\dot{V}(\mathbf{s}, \tilde{\boldsymbol{\theta}}) &= \mathbf{s}^T \mathbf{B}(\mathbf{q})\dot{\mathbf{s}} + \frac{1}{2} \mathbf{s}^T \dot{\mathbf{B}}(\mathbf{q})\mathbf{s} + \tilde{\boldsymbol{\theta}}^T \boldsymbol{\Gamma}^{-1} \dot{\tilde{\boldsymbol{\theta}}} \\ &= \mathbf{s}^T [\boldsymbol{\xi}^T \tilde{\boldsymbol{\theta}}(t) + \boldsymbol{\delta} - \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\mathbf{s} - \mathbf{K}\mathbf{s} - \boldsymbol{\tau}_c - \boldsymbol{\Sigma} \text{sign}(\mathbf{s})] \\ &\quad + \frac{1}{2} \mathbf{s}^T \dot{\mathbf{B}}(\mathbf{q})\mathbf{s} + \tilde{\boldsymbol{\theta}}^T \boldsymbol{\Gamma}^{-1} \dot{\tilde{\boldsymbol{\theta}}}\end{aligned}$$

Eliminando términos semejantes y tomado en consideración la propiedad 2.2 del modelo dinámico y la representación vectorial de la ley de adaptación (4.23), *i.e.*

$$\dot{\tilde{\boldsymbol{\theta}}} = -\boldsymbol{\Gamma} \boldsymbol{\xi} \mathbf{s}$$

se llega a

$$\dot{V}(\mathbf{s}, \tilde{\boldsymbol{\theta}}) = \mathbf{s}^T \boldsymbol{\delta} - \mathbf{s}^T \mathbf{K} \mathbf{s} - \mathbf{s}^T \boldsymbol{\tau}_c - \mathbf{s}^T \boldsymbol{\Sigma} \text{sign}(\mathbf{s}) \quad (4.29)$$

Ahora se buscan cotas superiores sobre cada uno de los términos de (4.29). Para el primer término, empleando (4.25) se llega a

$$\begin{aligned}\mathbf{s}^T \boldsymbol{\delta} &\leq |\mathbf{s}^T \boldsymbol{\delta}| \\ &\leq \|\mathbf{s}\| \|\boldsymbol{\delta}\| \\ &\leq \beta_1 \|\mathbf{s}\| + \beta_2 \|\mathbf{s}\|^2 + \|\mathbf{s}\|^3.\end{aligned} \quad (4.30)$$

Luego por el teorema de Rayleigh-Ritz el segundo término es acotado como

$$\begin{aligned}\mathbf{s}^T \mathbf{K} \mathbf{s} &\geq \lambda_{\min}\{\mathbf{K}\} \|\mathbf{s}\|^2 \\ -\mathbf{s}^T \mathbf{K} \mathbf{s} &\leq -\lambda_{\min}\{\mathbf{K}\} \|\mathbf{s}\|^2\end{aligned} \quad (4.31)$$

donde $\lambda_{\min}\{\mathbf{K}\}$ expresa el mínimo eigenvalor de \mathbf{K} . De [25] se tiene que $\sum_{i=1}^n s_i^4 \geq \frac{1}{n} \|\mathbf{s}\|^4$; por lo tanto, de la definición de $\boldsymbol{\tau}_c$ en (4.27) se tiene

$$\begin{aligned}\mathbf{s}^T \boldsymbol{\tau}_c &= \rho_1 s_1^4 + \dots + \rho_n s_n^4 \\ \mathbf{s}^T \boldsymbol{\tau}_c &\geq \rho_{\min} \sum_{i=1}^n s_i^4 \\ -\mathbf{s}^T \boldsymbol{\tau}_c &\leq -\frac{\rho_{\min}}{n} \|\mathbf{s}\|^4,\end{aligned} \quad (4.32)$$

con ρ_{\min} como la constante ρ_i mínima. Finalmente,

$$\begin{aligned}
 \text{sign}(\mathbf{s})^T \Sigma \mathbf{s} &= \sigma_1 \text{sign}(s_1) s_1 + \cdots + \sigma_n \text{sign}(s_n) s_n \\
 \text{sign}(\mathbf{s})^T \Sigma \mathbf{s} &= \sigma_1 |s_1| + \cdots + \sigma_n |s_n| \\
 \text{sign}(\mathbf{s})^T \Sigma \mathbf{s} &\geq \lambda_{\min}\{\Sigma\} \|\mathbf{s}\| \\
 -\text{sign}(\mathbf{s})^T \Sigma \mathbf{s} &\leq -\lambda_{\min}\{\Sigma\} \|\mathbf{s}\|
 \end{aligned} \tag{4.33}$$

donde $\lambda_{\min}\{\Sigma\}$ es el eigenvalor mínimo de Σ . Entonces $\dot{V}(\mathbf{s}, \tilde{\boldsymbol{\theta}})$ en (4.29) puede ser acotada superiormente por

$$\begin{aligned}
 \dot{V}(\mathbf{s}, \tilde{\boldsymbol{\theta}}) &\leq \beta_1 \|\mathbf{s}\| + \beta_2 \|\mathbf{s}\|^2 + \beta_3 \|\mathbf{s}\|^3 - \lambda_{\min}\{\mathbf{K}\} \|\mathbf{s}\|^2 \\
 &\quad - \left(\frac{\rho_{\min}}{n} \right) \|\mathbf{s}\|^4 - \lambda_{\min}\{\Sigma\} \|\mathbf{s}\|
 \end{aligned}$$

y aplicando la desigualdad $\|\mathbf{s}\|^3 \leq \|\mathbf{s}\|^2 + \|\mathbf{s}\|^4$ se llega a

$$\begin{aligned}
 \dot{V}(\mathbf{s}, \tilde{\boldsymbol{\theta}}) &\leq (\beta_1 - \lambda_{\min}\{\Sigma\}) \|\mathbf{s}\| + (\beta_4 - \lambda_{\min}\{\mathbf{K}\}) \|\mathbf{s}\|^2 \\
 &\quad - \left(\beta_3 - \frac{\rho_{\min}}{n} \right) \|\mathbf{s}\|^4
 \end{aligned} \tag{4.34}$$

donde $\beta_4 = \beta_2 + \beta_3$, con lo que es posible seleccionar constantes $\lambda_{\min}\{\Sigma\}$, $\lambda_{\min}\{\mathbf{K}\}$ y ρ_{\min} tal que $\dot{V}(\mathbf{s}, \tilde{\boldsymbol{\theta}}) \leq 0$, lo cual por el teorema 2.2 implica que \mathbf{s} y $\tilde{\boldsymbol{\theta}}$ están acotadas. Además se satisfacen las condiciones necesarias dadas por el teorema 2.1, por lo cual se concluye que el sistema en lazo cerrado es estable en el sentido de Lyapunov.

Con el propósito de demostrar la convergencia del error de posición $\tilde{\mathbf{q}}$ a cero cuando el tiempo t tiende a infinito, se estudian las propiedades del error generalizado de seguimiento y su derivada temporal. Primero, se considera nuevamente al vector \mathbf{s} dado en (4.2) como un sistema de ecuaciones diferenciales lineales de primer orden y, se emplean argumentos estándares de control lineal como el Lema 2.3 para establecer que

$$\mathbf{s} \in \mathcal{L}_{\infty}^n \Rightarrow \dot{\tilde{\mathbf{q}}}, \tilde{\mathbf{q}} \in \mathcal{L}_{\infty}^n$$

Lo anterior también implica que $\mathbf{q}, \dot{\mathbf{q}} \in \mathcal{L}_{\infty}^n$ puesto que $\|\mathbf{q}\| \leq \|\mathbf{q}_d\| + \|\tilde{\mathbf{q}}\|$, $\|\dot{\mathbf{q}}\| \leq \|\dot{\mathbf{q}}_d\| + \|\dot{\tilde{\mathbf{q}}}\|$, es decir, las normas de los vectores de posición y velocidad son menores que la suma de funciones acotadas.

Ahora se busca acotar por arriba a la norma de $\dot{\mathbf{s}}$ empleando la ecuación de lazo cerrado

expresada en (4.27)

$$\begin{aligned}
 \|B(\mathbf{q})\dot{s}\| &= \|\xi^T \tilde{\theta} + \delta - C(\mathbf{q}, \dot{\mathbf{q}})s - \mathbf{K}s - \tau_c - \Sigma \text{sign}(s)\| \\
 &\leq \|\xi^T \tilde{\theta}\| + \|\delta\| + \|C(\mathbf{q}, \dot{\mathbf{q}})s\| + \|\mathbf{K}s\| + \|\tau_c\| + \|\Sigma \text{sign}(s)\| \\
 &\leq \|\xi\| \|\tilde{\theta}\| + \beta_1 + \beta_2 \|s\| + \beta_3 \|s\|^2 + k_{C_1} \|\dot{\mathbf{q}}\| \|s\| \\
 &\quad + \lambda_{\text{Max}}\{\mathbf{K}\} \|s\| + \|\tau_c\| + \|\Sigma \text{sign}(s)\|
 \end{aligned} \tag{4.35}$$

donde la norma del vector regresor ξ es finita. Para τ_c se tiene

$$\begin{aligned}
 \|\tau_c\|^2 &= \rho_1^2 r_1^6 + \dots + \rho_n^2 r_n^6 \\
 &\leq \rho_{\text{Max}}^2 (r_1^6 + \dots + r_n^6) \\
 &\leq \rho_{\text{Max}}^2 \|s\|_6^6
 \end{aligned} \tag{4.36}$$

aplicando la raíz cuadrada a ambos lados de la ecuación (4.36) y empleando la equivalencia de normas

$$\|\tau_c\| \leq \rho_{\text{Max}} C_1 \|s\|^3 \tag{4.37}$$

para alguna constante $C_1 > 0$ adecuada. Por otro lado

$$\begin{aligned}
 \|\Sigma \text{sign}(s)\|^2 &= \sigma_1^2 \text{sign}(s_1)^2 + \dots + \sigma_n^2 \text{sign}(s_n)^2 \\
 &= \sigma_1^2 + \dots + \sigma_n^2 \\
 &\leq n \lambda_{\text{Max}}\{\Sigma\}^2
 \end{aligned}$$

y en consecuencia

$$\|\Sigma \text{sign}(s)\| \leq \sqrt{n} \lambda_{\text{Max}}\{\Sigma\} \tag{4.38}$$

Sustituyendo los resultados (4.37) y (4.38) en (4.35) y haciendo uso de la propiedad 2.1 del modelo dinámico, se llega a

$$\begin{aligned}
 \|\dot{s}\| &\leq \frac{1}{b_2} \left[\|\xi\| \|\tilde{\theta}\| + \beta_1 + \beta_2 \|s\| + \beta_3 \|s\|^2 + k_{C_1} \|\dot{\mathbf{q}}\| \|s\| \right. \\
 &\quad \left. + \lambda_{\text{Max}}\{\mathbf{K}\} \|s\| + \rho_{\text{Max}} C_1 \|s\|^3 + \sqrt{n} \lambda_{\text{Max}}\{\Sigma\} \right]
 \end{aligned} \tag{4.39}$$

con lo que se determina que

$$\dot{s} \in \mathcal{L}_{\infty}^n$$

puesto que la norma de $\dot{\mathbf{s}}$ en (4.39) está limitada superiormente por la suma de funciones acotadas.

Finalmente nótese que $\dot{V}(\mathbf{s}, \tilde{\boldsymbol{\theta}})$ en (4.34) también satisface

$$\dot{V}(\mathbf{s}, \tilde{\boldsymbol{\theta}}) \leq (\beta_4 - \lambda_{\min}\{\mathbf{K}\})\|\mathbf{s}\|^2.$$

Así, integrando a ambos lados de la desigualdad anterior desde $t = 0$ hasta $t = \infty$, i.e.

$$\int_0^\infty \dot{V}(\mathbf{s}, \tilde{\boldsymbol{\theta}}) dt < \int_0^\infty (\beta_4 - \lambda_{\min}\{\mathbf{K}\})\|\mathbf{s}\|^2 dt$$

se obtiene

$$\int_0^\infty \|\mathbf{s}\|^2 dt \leq \frac{V_\infty - V_0}{\lambda_{\min}\{\mathbf{K}\} - \beta_4}$$

lo cual es la definición de una función cuadrático integrable. Recapitulando, se tiene entonces que $\mathbf{s}, \dot{\mathbf{s}} \in \mathcal{L}_{\infty}^n$, $\mathbf{s} \in \mathcal{L}_2^n$ y usando el Lema 2.1 se prueba que

$$\lim_{t \rightarrow \infty} \mathbf{s}(t) = \mathbf{0} \quad (4.40)$$

Lema 4.2 Para el sistema (4.2), el resultado obtenido en (4.40) implica que

$$\lim_{t \rightarrow \infty} \tilde{\mathbf{q}}(t) = \mathbf{0} \quad (4.41)$$

Prueba: Sea

$$\dot{\tilde{\mathbf{q}}} = -\Lambda \tilde{\mathbf{q}} + \mathbf{s}$$

una ecuación diferencial vectorial de primer orden manejada por la entrada \mathbf{s} . Ya que todos los valores propios de $-\Lambda$ tienen parte real negativa, es decir satisfacen $\text{Re}[-\lambda_i] < 0$ ($i = 1, 2, \dots, n$), se tiene que la respuesta $\tilde{\mathbf{q}}(t)$ a entrada cero es asintóticamente estable, i.e., la solución $\tilde{\mathbf{q}}(t)$ tiende a $\mathbf{0}$ cuando $t \rightarrow \infty$.

Capítulo 5

Aplicación al robot Mitsubishi PA10-7CE

En este capítulo se presenta una descripción del robot manipulador Mitsubishi PA10-7CE que se encuentra en el Instituto Tecnológico de la Laguna, así como su estructura y componentes. Posteriormente se expone la interfaz gráfica desarrollada como parte del trabajo de tesis, la cual permite manejar al robot a través de MATLAB/Simulink. Finalmente, se aborda la aplicación en simulación del controlador descentralizado difuso adaptable al robot Mitsubishi PA10-7CE y se exponen los resultados. Cabe señalar que la aplicación experimental del controlador queda fuera del alcance de este trabajo.

5.1. Antecedentes y estudio del robot

El robot manipulador industrial Mitsubishi PA10-7CE es una de las versiones del "Brazo Inteligente Portátil de Propósito General (Portable General-Purpose Intelligent Arm PA10)" de arquitectura abierta desarrollado por Mitsubishi Heavy Industries (MHI). Esta versión se encuentra disponible en el Laboratorio de Mecatrónica y Control del Instituto Tecnológico de la Laguna, y consiste actualmente de *cuatro niveles funcionales* (Figura 5.1). Usualmente, en los robots manipuladores industriales estos niveles comparten una única tarjeta dedicada al control, lo cual implica una arquitectura cerrada.

Como se ha mencionado anteriormente, el sistema PA10 es abierto ya que está separado claramente por niveles, los cuales estén comunicados a través de alguna estandarizada,

y se puede acceder a ellas con relativa facilidad al conocer la forma en que se comunican [17].

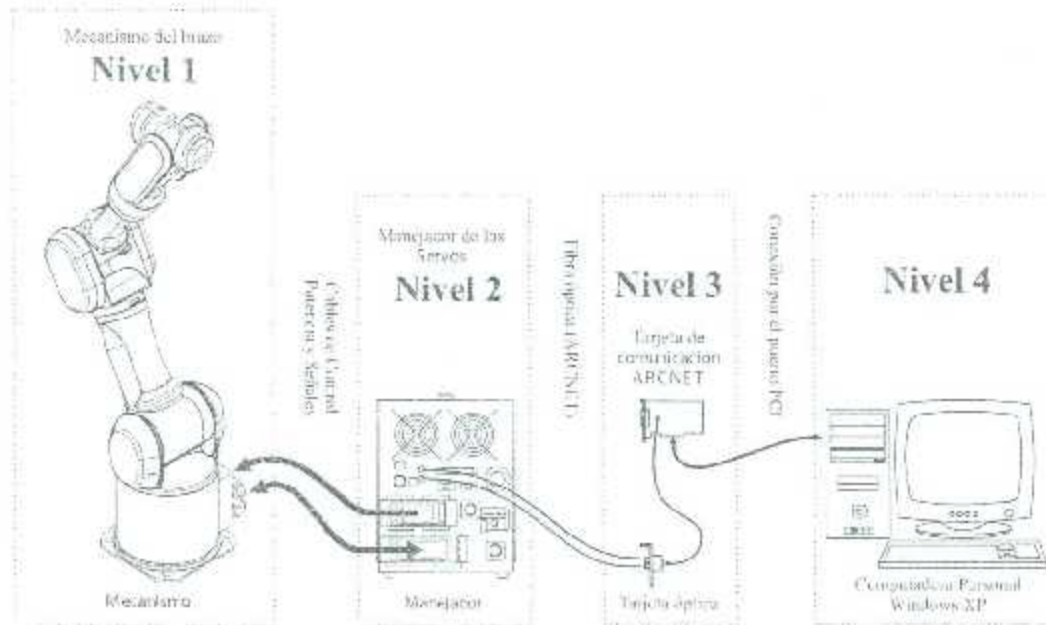


Figura 5.1: Niveles funcionales actuales del Mitsubishi PA10-7CE.

A continuación se presenta una breve descripción de los componentes más relevantes de cada uno de los niveles del robot PA10-7CE.

5.1.1. Nivel 1: Mecanismo

Es el nivel más bajo del robot y corresponde a toda la estructura mecánica. Los servomotores del PA10 son trifásicos del tipo CD sin escobillas (*brushless DC*) y están acoplados a los eslabones por medio de transmisiones armónicas (*harmonic drives*). Aquí se pueden apreciar dos componentes importantes

Brazo

Es la parte mecánica del sistema, y está compuesto de siete articulaciones conectadas a través de eslabones tal y como se muestra en la figura 5.2. Además, cuenta con frenos electromagnéticos en cada una de las articulaciones, cuyo propósito es el de sostener el robot, no de frenarlo, ni de sostener cargas externas cuando está apagado.

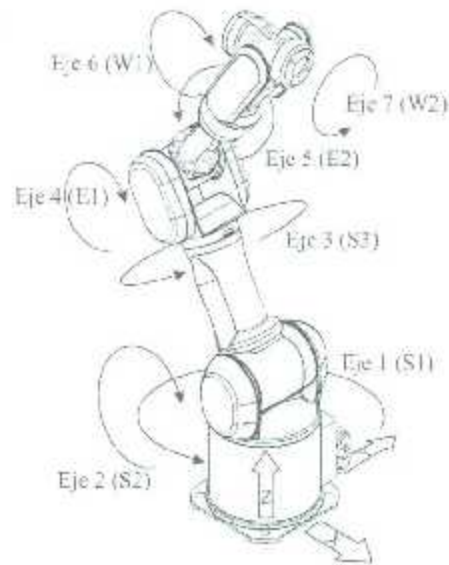


Figura 5.2: Parte mecánica del PA10-7CE.

Base

El robot, se puede instalar en una base con ruedas que dispone de cuatro brazos desplegados que le dan estabilidad a la base. Gracias a esta base el PA10 puede trasladarse con relativa facilidad.

5.1.2. Nivel 2: Controlador de servos

Este manipulador posee un tipo de conexión que permite a los usuarios tener acceso a los manejadores de los servos, gracias a una red de área local (LAN) que usa un protocolo de paso de turno "ARCNET", que asegura el desempeño en tiempo real. El manejador de los servos se puede configurar en *modo par* o *modo velocidad* dependiendo de cuál sea la consigna de control de movimiento para los servomotores.

Caja de control

Dentro de la caja de control (figura 5.3) se encuentran los manejadores de los servos y la fuente de potencia, que son los que controlan y proporcionan energía a los servomotores del robot. Además la caja de control cuenta con un transceptor que maneja el protocolo de comunicación ARCNET.

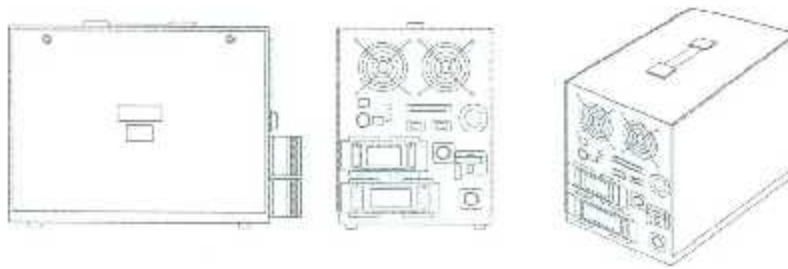


Figura 5.3. Caja de control de los servomotores del robot.

botón de paro de emergencia

Si se acciona este botón inmediatamente se activan los frenos y se apagan los servomotores de las articulaciones. Para poder manejar el robot después de presionar el botón de paro, es necesario liberar el botón y reiniciar la caja de control.

5.1.3. Nivel 3: Control de movimiento

A fin de manejar el robot tanto en modo par como en modo velocidad este nivel está dotado de una tarjeta con bus PCI que maneja el protocolo ARCNET, y una tarjeta o transceptor óptico que convierte las señales eléctricas en ópticas, lo que permite enviar y recibir datos entre los niveles 2 y 3 a través de fibra óptica plástica.

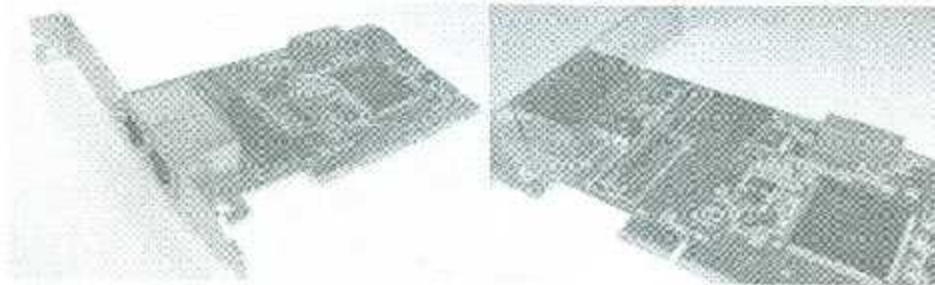


Figura 5.4: Modificación de la tarjeta ARCNET PCI20U-4000.

Tarjeta controladora de ARCNET

Actualmente se emplea la tarjeta PCI20U-4000 (figura 5.4) de la marca Contemporary Controls, la cual fue modificada como se muestra en [17] reemplazando el transceptor original por la tarjeta de conversión óptica, ya que originalmente no está diseñada para

transmitir datos por fibra óptica plástica. Esta tarjeta cuenta con el circuito integrado COM20022, mismo circuito controlador de ARCNET que emplea el PA10-7CE.

Tarjeta óptica

Esta tarjeta o transceptor fue diseñado en [17] tal que sea el equivalente del transceptor óptico MHI-D7210 de Mitsubishi Heavy Industries, utilizando los mismos modelos de transmisor HFBR-1528 y receptor HFBR-2528 de la tarjeta de conversión de Mitsubishi. El circuito usado se muestra en la figura 5.5.

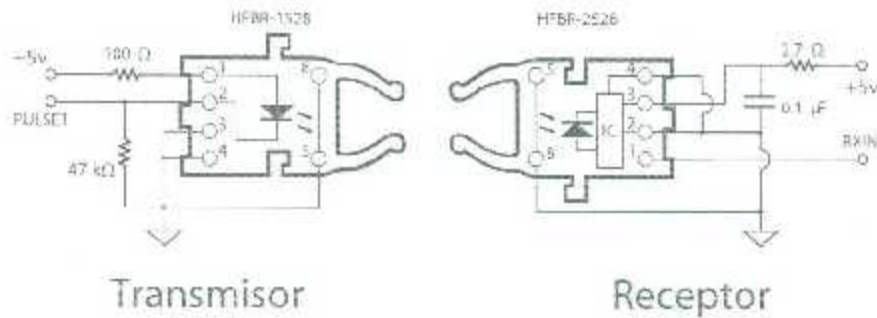


Figura 5.5: Circuito del transceptor óptico.

En la figura 5.6 se muestra la conexión entre la tarjeta controladora de ARCNET y la tarjeta de conversión óptica.

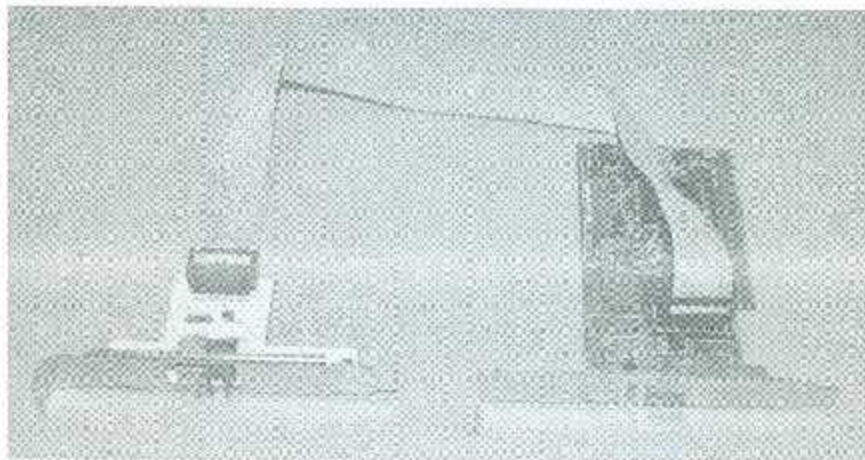


Figura 5.6: Conexión entre la tarjeta ARCNET y el transceptor óptico.

5.1.4. Nivel 4: Control de operación

El usuario interactúa con este nivel, el cuál está formado por una PC de control que tiene insertada la tarjeta de comunicación ARCNET a través de un puerto o bus PCI y la tarjeta de fibra óptica descrita anteriormente.

Esta computadora de control cuenta con los programas necesarios (ver [30]) para manejar el robot a través de una interfaz gráfica la cuál es descrita a detalle más adelante.

5.2. Resultados en simulación

Se llevó a cabo un estudio en simulación considerando el modelo dinámico del robot Mitsubishi PA10-7CE, tomado de [17] y [18] que debido a su extensión no es presentado en este documento, con el propósito de verificar los resultados teóricos del Capítulo 4 y de evaluar el desempeño del algoritmo de control descentralizado difuso adaptable dado por

$$\tau_i = k_i s_i + \rho_i s_i^3 + \hat{\eta}_i(\mathbf{x}_i, \hat{\theta}_i) + \sigma_i \text{sign}(s_i) \quad (5.1)$$

que emplea la siguiente ley adaptable para la estimación de los parámetros libres del sistema difuso

$$\dot{\hat{\theta}}_i = -\gamma_i \xi_i(\mathbf{x}_i) s_i \quad (5.2)$$

Las referencias deseadas de posición q_{id} y de velocidad \dot{q}_{id} con $i = 1, 2, \dots, 7$ están determinadas por las expresiones siguientes

$$\begin{aligned} q_{id} &= c(1 - e^{-pt^3}) \sin(\omega t) \\ \dot{q}_{id} &= \omega c(1 - e^{-pt^3}) \cos(\omega t) + 3cpt^2 \sin(\omega t) e^{-pt^3} \end{aligned}$$

Los parámetros c , p , ω utilizados para cada una de las articulaciones fueron seleccionados iguales y tienen los siguientes valores $c = \pi/4$, $p = .001$, $\omega = .285$ [rad/s], la figura 5.7 muestra las gráficas en el tiempo de las referencias deseadas.

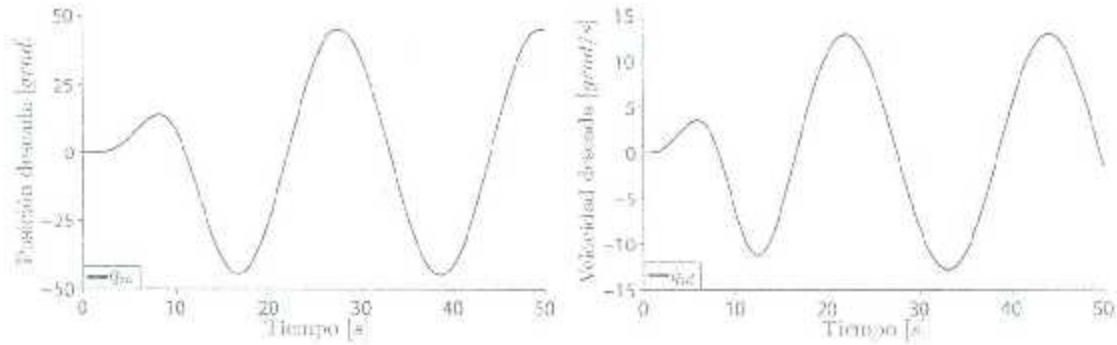


Figura 5.7: Trayectorias deseadas de movimiento.

5.2.1. Parámetros de diseño

Las ganancias involucradas en la ley de control (5.1) y en la ley de adaptación (5.2) para cada una de las articulaciones están contenidas en la tabla 5.1 y se ha empleado $\Lambda = \text{diag} \{ 57, 56, 50, 80, 50, 133, 335 \} [s^{-1}]$.

Tabla 5.1: Parámetros de diseño.

i	$k_i [\frac{Nm \cdot s}{rad}]$	$\rho_i [\frac{Nm \cdot s^2}{rad \cdot s}]$	$\delta_i [Nm]$	$\gamma_i [\frac{Nm}{rad}]$
1	70	3	10	10
2	90	5	25	10
3	40	.5	5	10
4	25	3	10	10
5	6	.12	5	5
6	6	.1	.15	10
7	1.2	.001	.1	1

Para aproximar la porción local de la dinámica del manipulador, cada uno de los sistemas difusos $\hat{\eta}_i(\mathbf{x}_i | \hat{\theta}_i)$ recibe dos entradas $x_{i1} = y_i$; $x_{i2} = \dot{q}_i$ ($j = 1, 2$) cuyos universos de discurso fueron particionados en $p_j = 5$ conjuntos difusos: $A_{ij}^1 = \text{NB}$ (*Negative Big*), $A_{ij}^2 = \text{NS}$ (*Negative Small*), $A_{ij}^3 = \text{ZE}$ (*Zero*), $A_{ij}^4 = \text{PS}$ (*Positive Small*), $A_{ij}^5 = \text{PB}$ (*Positive Big*), caracterizados por funciones de membresía triangulares $\mu_{A_{ij}^k}(x_{ij})$, que en todos los casos son simétricas respecto a cero.

La figura 5.8, esquematiza el tipo de funciones de membresía de entrada utilizadas donde $\mathbf{P} = \{ p_1, p_2, p_3, p_4, p_5 \}$ para la subfigura 5.8 (a) y $\mathbf{P} =$

$\{ p_1, p_2, p_3, p_4, p_5, p_6, p_7 \}$ para la subfigura 5.8 (b) denotan los soportes de estas funciones, también llamados particiones difusas.

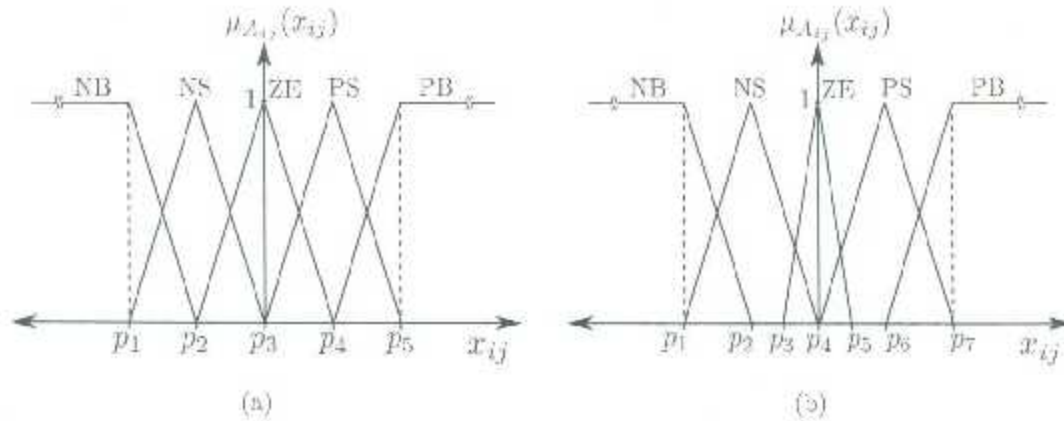


Figura 5.8: Funciones de membresía de entrada.

Para la posición, las funciones de membresía de entrada de las articulaciones 3, 4, 5 y 7 adoptan la forma de las mostradas en la subfigura 5.8 (b) y sus soportes o particiones difusas son $\mathbf{P} = \{ -2, -1, -0.5, 0, 0.5, 1, 2 \}$ [rad], y para las articulaciones 1, 2, y 6 las funciones de membresía de entrada son como las mostradas en la subfigura 5.8 (a) cuyos soportes están dados por $\mathbf{P} = \{ -6, -3, 0, 3, 6 \}$ [rad].

Por otro lado, los conjuntos difusos de entrada correspondientes a la velocidad tienen todos la forma de las funciones contenidas en la subfigura 5.8 (a). Las particiones del universo de discurso para las articulaciones 1 y 2 son $\mathbf{P} = \{ -2, -1.2, 0, 1.2, 2 \}$ [rad/s]; para las articulaciones 3, 4, 7 se tiene $\mathbf{P} = \{ -2.8, -1.6, 0, 1.6, 2.8 \}$ [rad/s] y finalmente los soportes de las funciones para las articulaciones 5 y 6 son $\mathbf{P} = \{ -3, -2, 0, 2, 3 \}$ [rad/s].

5.2.2. Resultados obtenidos

La figura 5.9 muestra los gráficos en el tiempo de las posiciones articulares y referencias deseadas. Se puede apreciar que cada articulación sigue con precisión a su respectiva trayectoria, lo cual indica que el controlador descentralizado difuso adaptable propuesto tiene un desempeño satisfactorio puesto que cumple el objetivo de control de movimiento, verificando así los resultados teóricos desarrollados previamente.

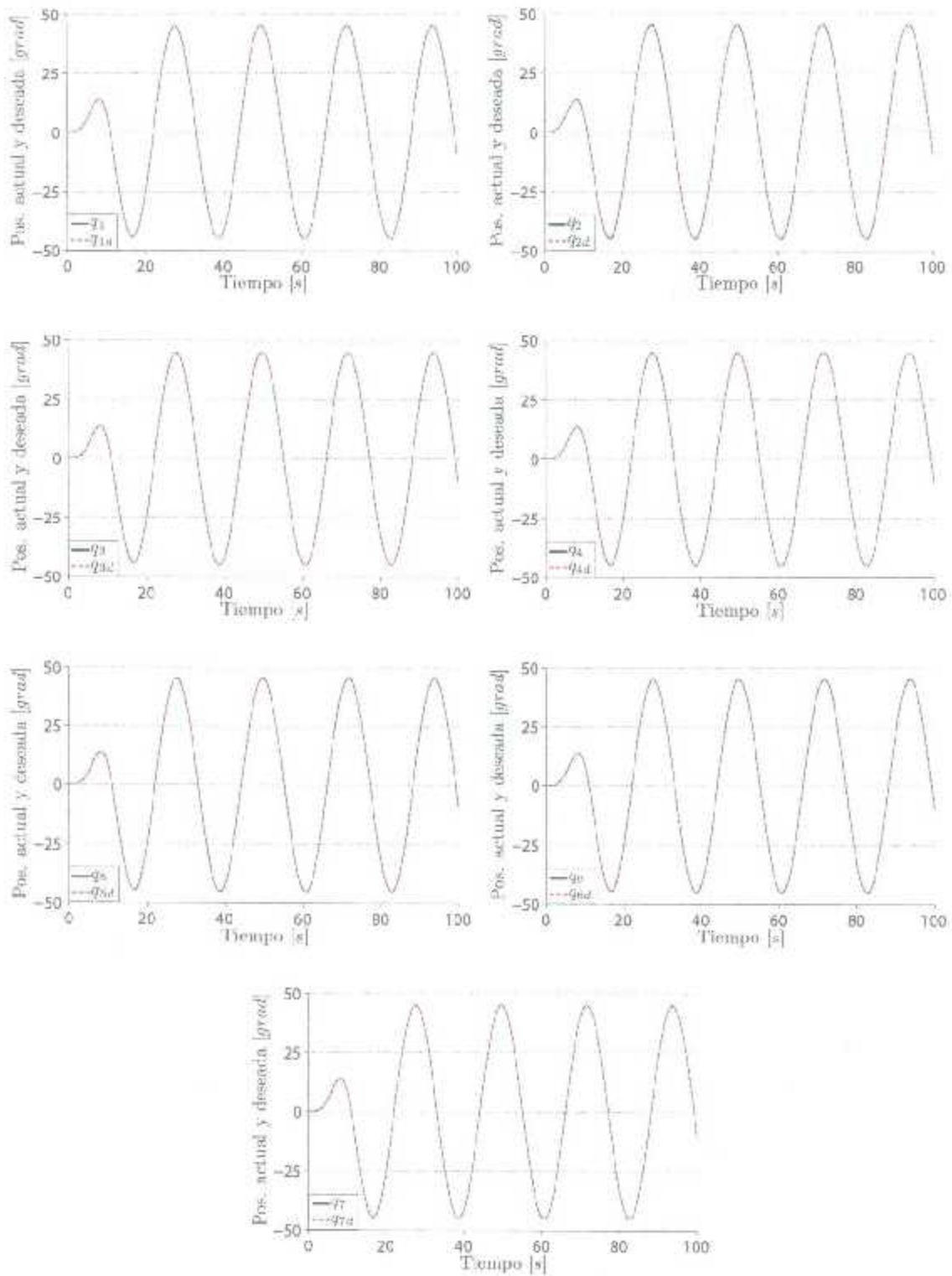


Figura 5.9: Gráficas de las posiciones actuales y deseadas de cada articulación

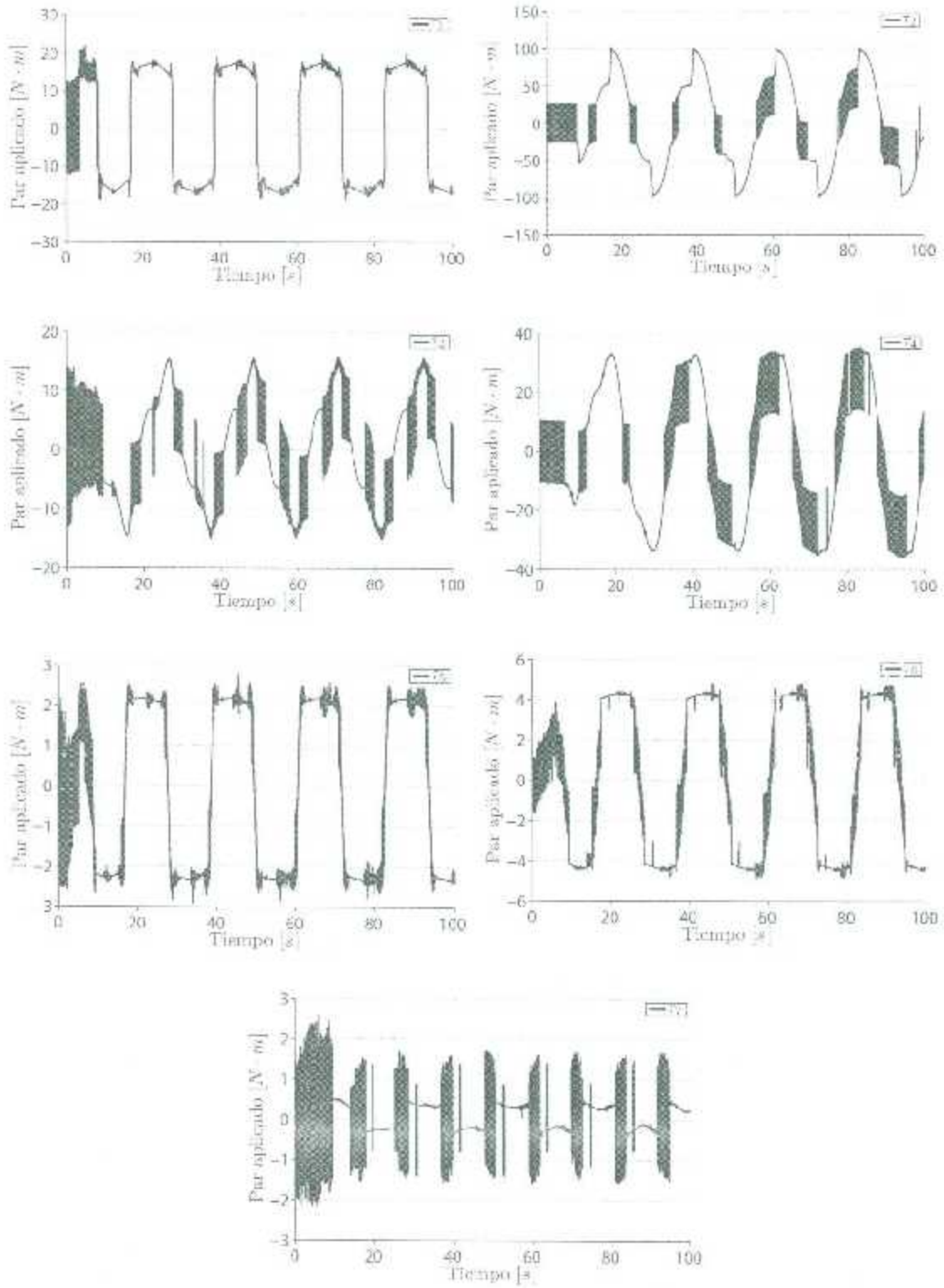


Figura 3.10: Gráficas de los pares aplicados a cada articulación.

Por otra parte, la figura 5.10 muestra los gráficos en el tiempo de los pares aplicados por la ley de control (5.1) a cada una de las articulaciones.

Para evaluar de cierta forma el desempeño del algoritmo desarrollado en este trabajo de tesis, se propuso realizar una comparación entre los errores de posición obtenidos con el controlador descentralizado difuso adaptable y los errores de posición obtenidos con un esquema diferente, por lo cual se llevó a cabo la simulación de un controlador Proporcional-Integral-Derivativo (PID) dado por la expresión

$$\tau = \mathbf{K}_p \bar{q} + \mathbf{K}_v \dot{\bar{q}} + \mathbf{K}_i \int_0^t \bar{q}(\xi) d\xi \quad (5.3)$$

Empleando ganancias proporcionales y derivativas equivalentes a la acción de control $k_i s_i$ en (5.1) siendo esto posible debido a la selección de las ganancias PD como $\mathbf{K}_p = \text{diag}\{k_{p1}, \dots, k_{pm}\}$, $\mathbf{K}_v = \text{diag}\{k_{v1}, \dots, k_{vm}\}$. Por otro lado \mathbf{K}_i también se define como una matriz diagonal $\mathbf{K}_i = \text{diag}\{k_{i1}, \dots, k_{im}\}$ y la selección de estas ganancias se realizó de manera heurística a fin de encontrar el mejor desempeño del controlador.

Los valores de las ganancias utilizadas para el controlador PID están contenidos en la tabla 5.2

Tabla 5.2: Ganancias del controlador PID.

Articulación	k_p [$\frac{Nm}{rad}$]	k_v [$\frac{Nm.s}{rad}$]	k_i [$\frac{Nm}{s}$]
1	3990	70	1600
2	5040	90	3000
3	2000	40	760
4	2000	25	1000
5	300	6	500
6	798	6	400
7	402	1.2	300

Los resultados de esta comparación se muestran en la figura 5.11, en la cuál se puede apreciar que el controlador difuso adaptable tiene un mejor desempeño en las articulaciones 1 a 4 mientras que en las articulaciones 5 a 7 se nota un mejor desempeño del controlador PID aunque la diferencia es relativamente pequeña. En general se considera que el desempeño del controlador propuesto es bastante satisfactorio.

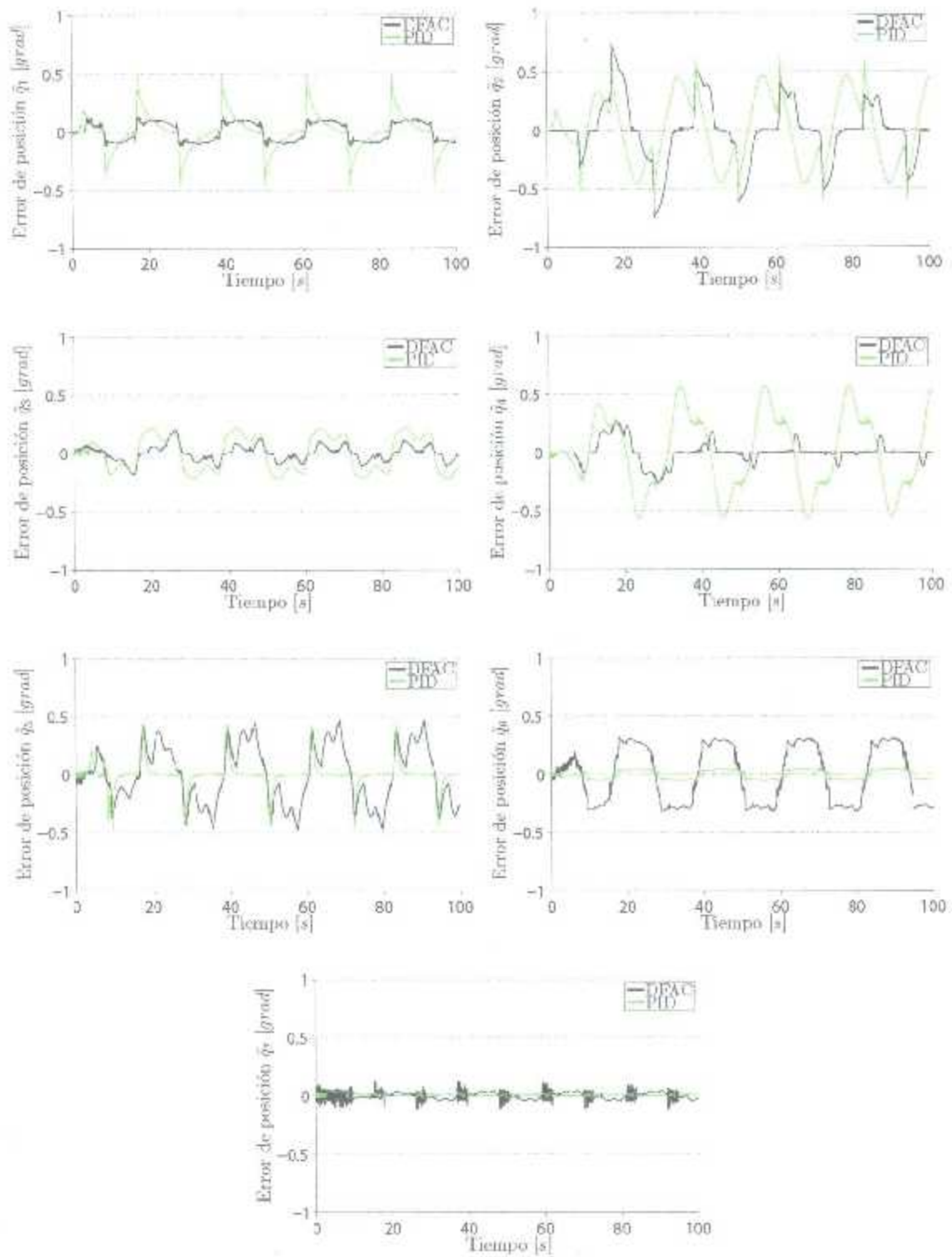


Figura 5.11: Errores de posición obtenidos con los controladores difuso adaptable descentralizado (DFAC) y PID.

5.3. Interfaz gráfica de control

A pesar de la existencia de programas para el manejo del Mitsubishi PA10-7CE como el reportado en [17], el hecho de poder manejarlo a través de MATLAB/Simulink motivó y justificó el desarrollo de una interfaz gráfica (figura 5.12) que permita al usuario tener acceso de una forma sencilla y sistemática a las funciones del robot.

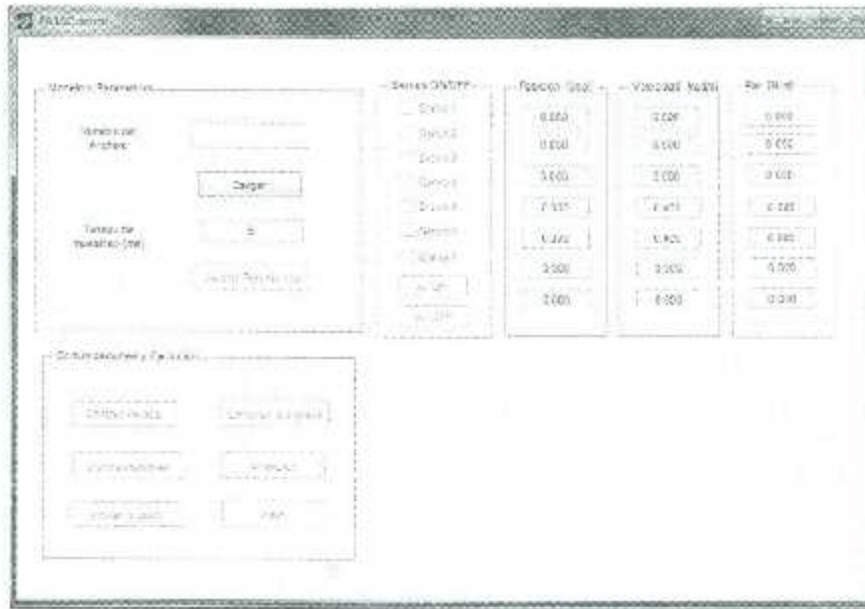


Figura 5.12: Interfaz gráfica del usuario para el manejo del PA10-7CE.

Para el funcionamiento en MATLAB/Simulink de la interfaz de control del Mitsubishi PA10 se hace uso de las siguientes herramientas:

- Quarc versión 2.3 “Quanser Real-Time Rapid Control Prototyping Software”. Es un software desarrollado por la compañía Quanser, que está diseñado para acelerar el diseño e implementación de procesos de control. Se integra con Simulink para crear prototipos de control y realizar pruebas en tiempo real y está totalmente integrado con Simulink Coder. Al instalar este programa se generan la librería en la que se encuentra el bloque PA10 Control.
- Microsoft Visual Studio Professional Edition 2010 (versión 10.0). Es un entorno de desarrollo integrado (IDE, por sus siglas en inglés) para sistemas operativos Windows que soporta múltiples lenguajes de programación tales como C++ y

es utilizado debido a que el programa Quare requiere de un compilador de C++ soportado por MATLAB.

- Matlab R2011a con: Simulink, Simulink Coder, MATLAB Coder, Control System Toolbox.

Estos programas fueron instalados en una computadora del Laboratorio de Mecatrónica y Control del IITL, con Windows 7 de 64-bits.

Cabe señalar que la interfaz desarrollada es una versión preliminar la cuál a la fecha, sólo contempla su uso para experimentación en tiempo real, bajo ciertas condiciones de diseño que son expuestas a detalle en el Apéndice B. A continuación se describen los controles y demás componentes que se encuentran en la ventana de la interfaz gráfica.

- *Nombre del archivo:* esta etiqueta indica que en el cuadro de texto adyacente se debe ingresar el nombre del archivo de simulink el cual contiene el algoritmo de control.
- *Cargar:* Al presionar este botón se abre el archivo de Simulink que se ha indicado. Si no existe un archivo con el nombre dado, se despliega una ventana de error.
- *Tiempo de muestreo:* esta etiqueta indica que en el cuadro de texto adyacente se debe ingresar el valor del tiempo de muestreo a utilizar en milisegundos, el cuál debe estar entre los 5 y 100 ms. Por defecto este cuadro de texto tiene un valor asignado de 5 ms.
- *Ajustar parámetros:* Este botón establece por defecto los siguientes parámetros en la ventana de configuración de parámetros del archivo de Simulink (*Simulation/Configuration Parameters*): Integrador odel (Euler), paso fijo, tiempo final de ejecución infinito, modo de ejecución externo. Si el tiempo de muestreo se encuentra fuera de los límites, se despliega una ventana de error.
- *Construir modelo:* Al presionar este botón se crean los archivos necesarios para la ejecución del modelo en tiempo real, y es equivalente al comando *Tools/Code Generation/Build Model* de Simulink.
- *Conectar a objetivo:* La acción de este botón es equivalente al comando *Simulation/Connect to Target* de Simulink para modo externo.

- *Comunicaciones:* Al presionar este botón se establece la comunicación con los manejadores de los servos y el tiempo de simulación de Simulink comienza a correr, si la comunicación no es llevada a cabo con éxito, una ventana de error de Simulink es desplegada. La acción de este botón es equivalente al comando *Simulation/Start Real-Time code* de Simulink para modo externo.
- *Servos ON/OFF:* al marcar o desmarcar alguno de los check box contenidos en este panel, se enciende o apaga respectivamente el servomotor correspondiente. El botón *all ON* enciende todos los servomotores del PA10, mientras que el botón *all OFF* apaga todos los servomotores.
- *Arranque:* Una vez que se inició la comunicación con el robot y se activaron los servos, es posible iniciar el controlador del PA10 al presionar este botón; en este momento el tiempo de inicio de control empieza a correr.
- *Enviar a casa:* Al presionar este botón se inicia el algoritmo para llevar al robot a su posición de casa sin importar lo que se encuentre haciendo el robot.
- *Paro:* Al presionar este botón se detiene el robot, los servos son apagados y los frenos magnéticos que detienen al PA10 se activan; esta acción es equivalente al comando *Simulation/Stop Real-Time Code* para modo externo de Simulink.
- *Paneles adicionales:* Son los paneles ubicados en la parte superior derecha de la interfaz (véase la Figura 5.12) e indican la posición, velocidad y par actuales de cada una de las articulaciones.

5.3.1. Operación de la interfaz de control

A continuación se brinda una lista de los pasos para operar la interfaz de control del robot Mitsubishi PA10-7CE

1. Encender los manejadores de los servomotores. Esto se realiza mediante el interruptor que se encuentra en la cara posterior de la caja de control.
2. Iniciar el programa. A la fecha, esto se lleva a cabo al ejecutar el archivo de matlab (PA10Control.m) que se encuentra en la computadora de control.
3. Ingresar el nombre del archivo de Simulink que contiene el algoritmo de control.

4. Cargar el programa.
5. Si es necesario, ingresar un tiempo de muestreo diferente al establecido por defecto y presionar el botón *ajustar parámetros*.
6. Construir el modelo.
7. Conectar a objetivo.
8. Establecer comunicaciones con el robot, presionando el botón *comunicaciones*.
9. Encender los servocontroladores que serán utilizados, a través del panel *Servos ON/OFF*. Al menos uno de los servomotores debe estar encendido para continuar al siguiente paso.
10. Iniciar el controlador empleando el botón *Arranque*. Nótese que al establecer comunicaciones, el tiempo de simulación de Simulink empieza a correr, sin embargo el tiempo t de ejecución del controlador empieza a correr hasta este paso; es decir, el tiempo inicial t_0 se da al presionar el botón *Arranque*.
11. Enviar a casa.

Si en algún punto es necesario detener al robot completamente, es posible realizarlo al presionar el botón *Paro* en la interfaz gráfica.

5.3.2. Errores comunes

En algunas ocasiones después de establecer comunicaciones con el robot, no es posible encender ninguno de los servomotores, lo cual se resuelve fácilmente presionando el botón *paro* de la interfaz gráfica y volviendo a establecer comunicaciones con el PA10. En ciertas ocasiones al intentar construir el modelo, la interfaz despliega una ventana de error notificando que esta acción no se ha llevado a cabo, lo que se resuelve borrando las variables del workspace de MATLAB y presionando nuevamente el botón *construir modelo* en el programa. Finalmente, otro error común aunque menos frecuente que los anteriores ocurre al tratar de establecer comunicaciones con el robot, si este error pasa, Simulink despliega una ventana de error. Para solucionar lo anterior es necesario cerrar MATLAB, volverlo a abrir y empezar nuevamente la secuencia de operación de la interfaz gráfica.

Actualmente se están realizando las pruebas de la aplicación en tiempo real del controlador propuesto al robot PA10-7CE. Sin embargo por cuestiones de falta de tiempo no se han alcanzado a presentar en este trabajo de tesis.

Conclusiones y trabajo futuro

6.1. Conclusiones

En este trabajo de tesis se estudió el problema de seguimiento de trayectoria de robots manipuladores empleando un controlador difuso adaptable descentralizado, y vía un diseño por síntesis de Lyapunov se demostró la estabilidad del sistema en lazo cerrado lo que implica que los estados del sistema en conjunto están acotados. Además, a través de un análisis funcional típico de sistemas adaptables, se probó que se satisface el objetivo de control de movimiento, es decir, se demostró la convergencia global del error de posición.

A fin de verificar los resultados teóricos obtenidos se llevó a cabo una simulación del controlador difuso propuesto, considerando el robot Mitsubishi PA10 de siete grados de libertad, y se compararon cualitativamente los resultados contra un controlador PID mostrando un buen desempeño. La implementación exitosa del algoritmo propuesto muestra una de sus ventajas respecto a esquemas de control centralizado, ya que para el caso en que el número de articulaciones es grande un esquema descentralizado no es factible debido a su alto costo computacional.

Con los resultados descritos previamente se desarrolló el artículo "Convergencia Global de un Controlador Difuso Adaptable Descentralizado para el Movimiento de Robots Manipuladores" y se sometió al XVI Congreso Latinoamericano de Control Automático (CLCA 2014), a realizarse en Octubre de 2014 en la ciudad de Cancún, Quintana Roo, México.

Otra de las contribuciones resultado de este proyecto de investigación, consiste en el desarrollo de la interfaz gráfica de usuario, que permite manejar el robot FA10-7CE a través de MATLAB/Simulink en tiempo real como una alternativa a los programas *WinMechLab* y *Mitsubishi Main Control*. Sin embargo, se trata de una versión preliminar dado que aún existen diferentes funciones y mejoras que deben ser realizadas a futuro.

6.2. Trabajo futuro

Una de las mejoras más importantes que se han planteado, es la de permitir que el usuario sólo genere propiamente el archivo que contiene el algoritmo de control, a diferencia de como se hace actualmente generando un archivo que contiene el controlador y el resto de bloques necesarios para el manejo de la interfaz. También es necesario desarrollar áreas de graficación definidas por el usuario a fin de facilitar el seguimiento de los experimentos.

Finalmente, la implementación del controlador difuso adaptable para su experimentación en tiempo real y el empleo de estas pruebas para el desarrollo de un artículo de revista indizada son otros de los trabajos futuros que podrían complementar o mejorar las metas a alcanzar en este proyecto de investigación.

Parámetros del robot Mitsubishi PA10-7CE

La masa total del robot Mitsubishi PA10-7CE es de aproximadamente 40 [kg], y puede sostener en su efector final una carga de hasta 10 [kg], como se explica en [17].

En la tabla A.1 se muestran las consignas máximas de velocidad y par que se pueden aplicar a cada una de las articulaciones del robot así como los límites de posición.

La reducción de engranajes de la transmisión armónica que tienen los motores de las articulaciones es de 50. Es decir, si se le aplica al motor un par de 1 [N · m], en realidad son 50 [N · m] (después de la transmisión) los que se aplican a la articulación.

Articulación	Velocidad [rad/s]	Par [N · m]	Posición [°]
S1	1	232	±165
S2	1	232	±85
S3	2	100	±165
E1	2	100	±125
W1	2π	14.5	+245
W2	2π	14.5	±145
W3	2π	14.5	+245

Tabla A.1: Parámetros del robot Mitsubishi PA10-7CE.

Apéndice **B**

Datos técnicos de la interfaz gráfica de usuario

El motor de la interfaz gráfica de usuario abordada en la sección 5.3, es el bloque de simulink "PA10 Control" desarrollado por Quanser y mostrado en la figura B.1, el cuál es el vínculo de comunicación entre MATLAB/Simulink y el robot Mitsubishi PA10-7CE. Este bloque está encargado de enviar las consignas de control de par o velocidad dependiendo del modo de operación, y de recibir los ángulos, velocidades y pares de cada articulación y puede ser encontrado en la siguiente librería de simulink.

QUARC Targets/Devices/Third Party/Mitsubishi/Robots/PA10/Interfacing

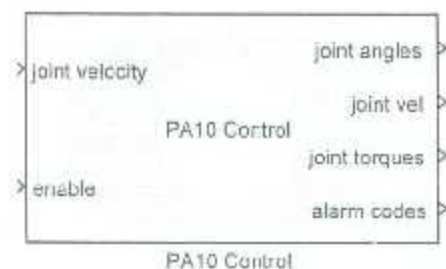


Figura B.1: Bloque PA10 Control.

El bloque PA10 Control envía las consignas de control (par o velocidad) para cada articulación del robot PA10-7CE. Cada articulación puede ser habilitada, deshabilitada y controlada independientemente. Las salidas del bloque PA10 Control, son las posiciones, velocidades y pares de cada articulación así como códigos de alarma.

Nótese que este bloque utiliza el protocolo ARCNET para comunicarse con el robot. A continuación se presenta una breve descripción de los puertos de entrada y de salida del bloque PA10 Control, suponiendo como es el caso que el robot PA10 a controlar es la versión de 7 grados de libertad.

a) Puertos de entrada

- **joint velocity/torque:** Es un vector de 7 elementos que contiene las consignas de velocidad en $[rad/s]$ o las consignas de par en $[N \cdot m]$ dependiendo del modo de operación del robot.
- **enable:** Es un vector de 7 elementos de valor booleano indicando si cada articulación está habilitada (valor verdadero 1) o deshabilitada (valor falso 0).

b) Puertos de salida

- **joint angles:** Es un vector de 7 elementos que contiene las posiciones articulares actuales en $[rad]$.
- **joint vel:** Es un vector de 7 elementos que contiene la velocidad de cada articulación en $[rad/s]$.
- **joint torques:** Es un vector de 7 elementos de los pares articulares actuales en $[N \cdot m]$.
- **alarm codes:** Es un vector de 7 elementos de los códigos de alarma que entrega el sistema PA10.

Los comandos **joint velocity/torque** son del tipo *double*, mientras que los elementos del vector **enable** son del tipo *boolean*. Las salidas del bloque son todas del tipo *double* excepto por el vector **alarm codes**, el cuál es del tipo *uint16*.

Ahora se describen los parámetros que deben ser configurados en el bloque PA10 Control en el cuadro de dialogo mostrado en la figura B.2.

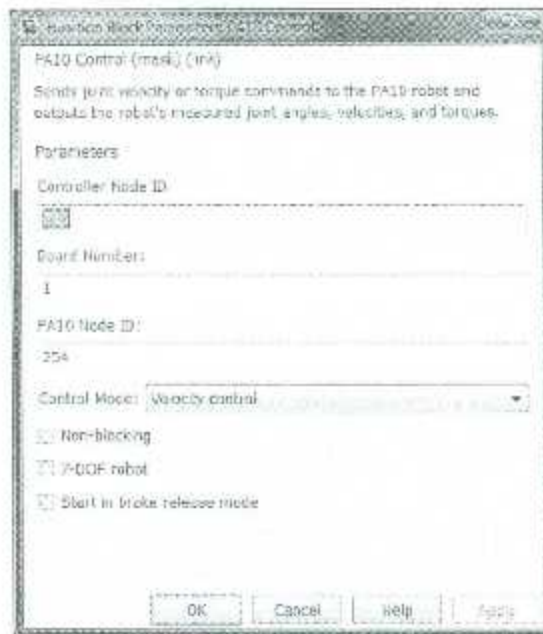


Figura B.2: Parámetros del bloque PA10 Control.

Puertos de entrada

- **Controller Node ID:** se refiere al nodo ID de la tarjeta ARCNET de la computadora local que envía los comandos al PA10. Se requiere que este ID sea 255.
- **Board Number:** El número de la tarjeta ARCNET en la computadora local. El valor por defecto es 1.
- **PA10 Node ID:** El nodo ID de la tarjeta ARCNET de la caja de control del PA10. Este ID debe ser 254.
- **Control Mode:** Esta opción selecciona si los comandos dados como entrada a este bloque se van a utilizar para el control en modo velocidad o en modo par. Nótese que cada modo de control utiliza la misma señal de entrada por lo que es fundamental que el usuario compruebe que está utilizando el modo de control correcto.
- **Non-blocking:** Esta opción selecciona si el bloque funciona con métodos de comunicación no bloqueante. La operación predeterminada es bloqueante, y el bloque puede ser colocado en un subproceso asíncrono para evitar muestras perdidas si la frecuencia de muestreo es alta.

- **7-DOF robot:** Esta opción debe ser seleccionada para trabajar con la versión de 7 grados de libertad del sistema PA10. Si esta opción no es seleccionada se asume que la versión utilizada es la de 6 grados de libertad.
- **Start in brake release mode:** Seleccione esta opción para iniciar el PA10 en el modo de liberación del freno. Si se selecciona esta opción, con el vector **enable** activará o desactivará el freno de forma conjunta y todos las consignas de control serán ignoradas.

B.1. Generación de nuevos controladores

Para crear un nuevo controlador que pueda ser manejado a través de la interfaz gráfica de usuario, es necesario incluir en el archivo de simulink los bloques “Trayectorias de Movimiento” (figura B.3 (a)) y “PA10-7CE Simulink” (figura B.3 (b)) los cuales están dentro del archivo de simulink “BloquesPA10control.mdl” que se encuentra en la computadora de control.

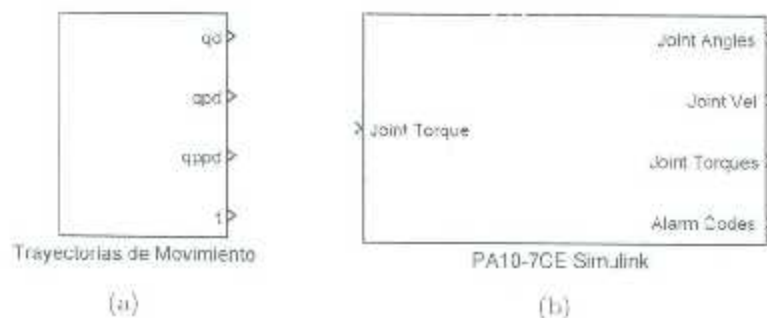


Figura B.3: Bloques para la generación de nuevos controladores

A continuación se describen los bloques

- Trayectorias de Movimiento:** Dentro de este bloque se encuentra una función integrada de MATLAB (*MATLAB Function*) llamada “Referencias Deseadas” en la que se debe ingresar el código de las posiciones, velocidades, y aceleraciones deseadas. Las salidas del bloque Trayectorias de Movimiento son
 - **qd:** vector de 7 elementos que contiene las posiciones deseadas de cada articulación en $[rad]$.

- **qpd**: vector de 7 elementos que contiene las velocidades articulares deseadas en $[rad/s]$.
- **qppd**: vector de 7 elementos que contiene las aceleraciones deseadas para cada articulación en $[rad/s^2]$.
- **t**: valor escalar que se refiere al tiempo de ejecución del controlador. Esta salida tiene un valor cero hasta que el botón *Arranque* de la interfaz gráfica es presionado, momento en el que comienza a incrementar su valor. Debe ser utilizado para restablecer los integradores del controlador, es decir debe ser empleado como la variable *External Reset* de los bloques "Integrator" de Simulink.

b) **PA10-7CE Simulink**: En su interior se encuentra el bloque PA10 Control descrito con anterioridad y el resto de elementos necesarios para el manejo del controlador a través de la interfaz. Las entradas y salidas de este bloque son exactamente iguales a las del bloque PA10 Control.

En la figura B.4 se muestra un controlador PID de seguimiento implementado en Simulink para su manejo con la interfaz gráfica de usuario, donde es posible apreciar la incorporación de los bloques tratados en esta sección y su funcionamiento.

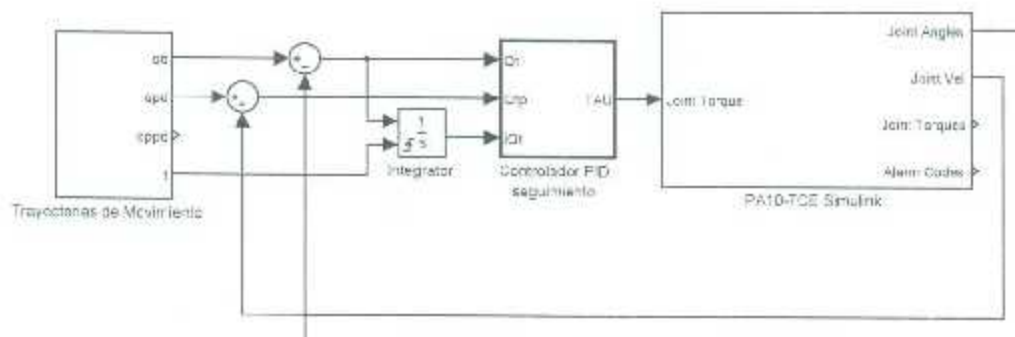


Figura B.4: Diagrama a bloques de Simulink para su manejo con la interfaz gráfica.

Bibliografía

- [1] Chen C.-T. (1999), *Linear System Theory and Design*, Oxford University Press, New York Oxford.
- [2] Desoer C. A. and Vidyasagar M. (1975), *Feedback Systems: Input-Output Properties*, New York: Academic.
- [3] Haddad W., and Chellaboina V. (2008) *Nonlinear Dynamical Systems and Control: A Lyapunov-Based Approach*, Princeton University Press.
- [4] Huang A. C., Chien M. C. (2010), *Adaptive Control of Robot Manipulators: A Unified Regressor-Free Approach*, World Scientific Publishing Co.
- [5] García R., Sánchez E., Llama M. and Ruz J. (2011), Decentralized Neural Backstepping Control for an Industrial PA10 7CE Robot Arm, *International Conference on Neural Computation: Theory and Applications*, Paris, France, October 21-24, 2011.
- [6] García R., Sánchez E., Bayro E., Santibañez V., and Ruz J., (2011), Real-time decentralized neural block control: Application to a two DOF robot manipulator, *International Journal of Innovative Computing, Information and Control*, Vol. 7, No. 3, pp. 1075-1085.
- [7] Ioannou P., and Sun J. (2012), *Robust Adaptive Control*, Dover Publications Inc.
- [8] Kelly R., Santibañez V., and Loría A. (2005), *Control of Robot Manipulators in Joint Space*, Springer-Verlag.

-
- [9] Khalil H. (2001). *Nonlinear Systems*. Prentice Hall, Upper Saddle River, New Jersey.
- [10] Kim Y. T. (2002). Independent Joint Adaptive Fuzzy Control of Robot Manipulator, *Automation Congress, 2002 Proceedings of the 5th Biannual World*, Vol. 14, pp. 645 - 652.
- [11] Lewis F. L., and Munro N. (2004). *Robot Manipulator Control Theory and Practice*, Marcel Dekker Inc.
- [12] Liu M. (1999), Decentralized control of robot manipulators: nonlinear and adaptive approaches, *IEEE Transactions on Automatic Control*, Vol.44, No.2, pp. 357-363.
- [13] Liu M. (2000), Stability Analysis of Decentralized Adaptive Fuzzy Logic Control for Robot Arm Tracking, *IEEE Proceedings of the 39th IEEE Conference on Decision and Control*, Vol. 1 pp. 883-888.
- [14] Llama M., (2001) *Diseño de Controladores Inteligentes Aplicados a Robots Manipuladores*, Tesis de Doctorado, Instituto Tecnológico de la Laguna, México.
- [15] Llama M., Kelly R., and Santibáñez V., (2001) A Stable Motion Control System for Manipulators via Fuzzy Self-Tuning, *Fuzzy Sets and Systems* Vol. 124, No. 2, pp. 133-154.
- [16] Llama M., Kelly R., Santibáñez V. and Centeno H. (2009) An adaptive fuzzy controller for robot manipulators: Theory and experimentation, *International Journal of Factory Automation, Robotics and Soft Computing*, PP. 122-131.
- [17] Ramírez C., (2008) *Modelado y Control en Modo Par del Robot Mitsubishi PA10-7CE*, Tesis de Maestría, Instituto Tecnológico de la Laguna, México.
- [18] Sakanassi J., Santibáñez V., Mazuca S. y Campa R. (2011), Identificación de los Pares de Fricción en el Robot Mitsubishi PA10-7CE, *Congreso anual AMCA 2011*.
- [19] Siciliano B., Sciavicco L., Villani L. and Oriolo G. (2009), *Robotics Modelling, Planning and Control*, Springer-Verlag.
- [20] Slotine J. J., and Li W. (1987), On the Adaptive Control of Robot Manipulators, *The International Journal of Robotics Research*, Vol. 6, pp. 49-59.

-
- [21] Slotine J. J., and Li W. (1990), *Applied Nonlinear Control*, Prentice Hall.
- [22] Spong M. W., Vidyasagar M., and Hutchinson S. (2005), *Robot Modeling and Control*, John Wiley & Sons.
- [23] Spooover J., Maggiore M., Ordóñez R., and Passino K. (2002), *Stable Adaptive Control and Estimation for Nonlinear Systems: Neural and Fuzzy Approximator Techniques*, John Wiley & Sons.
- [24] Wang L. X. (1996), *A Course In Fuzzy Systems and Control*, Prentice Hall International Inc.
- [25] Yang Z-J., Fukushima Y., and Qin P. (2011), Decentralized Adaptive Robust Control of Robot Manipulators, *IEEE Transactions on Control Systems Technology*, Vol. 20, pp. 1357 - 1365.
- [26] Zadeh Lofti A. (1965), Fuzzy Sets, *Information and Control*, 8:338-353.
- [27] Contemporary Control Systems, Inc., "ARCNET Interface Cards from Contemporary Controls for the PA10 Robot Arm from Mitsubishi Heavy Industries".
- [28] Contemporary Control Systems, Inc., "COM20020/22 Null Stack Driver, Kernel Mode Driver for Windows 2K/XP version 1.4a".
- [29] http://www.quarcservice.com/ReleaseNotes/files/pa10_control_block.html
- [30] Quanser Quick Installation Guide: QUARC 2.3, Quanser Real-Time Rapid Control Prototyping Software for Windows.

