



**UNIVERSIDAD AUTÓNOMA
DEL ESTADO DE HIDALGO**

INSTITUTO DE CIENCIAS BÁSICAS E INGENIERÍA

**INGENIERÍA EN ELECTRÓNICA Y
TELECOMUNICACIONES**



**“Control PD y PID difuso para un robot de dos
grados de libertad”**

T E S I S

**QUE PARA OBTENER EL TÍTULO DE INGENIERO
EN ELECTRÓNICA Y TELECOMUNICACIONES**

P R E S E N T A:

Israel Hernández Campero

ASESOR:

M. en C. Elías Varela Paz

MINERAL DE LA REFORMA, HIDALGO, MAYO DEL 2008.

Agradecimientos

A ti señor, por haberme dado la fortaleza que necesitaba, a mi familia que me han conducido con amor y sabiduría, hoy ven forjado un anhelo, una ilusión y un deseo.

Gracias por enseñarme lo que han recogido a su paso por la vida y por darme la libertad de elegir mi futuro, por brindarme con las manos abiertas su apoyo y su confianza en mi preparación para un futuro mejor.

Por que hoy he recibido su más valiosa herencia, mi profesión por ustedes la obtuve, y a ustedes se las brindo con admiración y respeto.

A todas las personas que directa e indirectamente colaboraron para la conclusión de este gran proyecto.

Sinceramente, gracias.

Contenido

Agradecimientos.....	ii
Contenido.....	iii
Índice de tablas.....	iv
Índice de Figuras.....	V
1. Introducción.	
1.1 Objetivos.....	1
1.2 Justificación.....	1
1.3 Planteamiento del problema.....	2
1.4 Solución propuesta.....	2
1.5 Metodología.....	2
1.6 Contribución de la tesis.....	3
1.7 Estado del arte.....	3
2. Lógica difusa y control difuso.	
2.1 Introducción.....	6
2.2 Lógica difusa.....	7
2.2.1 Conjuntos difusos y operaciones básicas.....	7
2.3 Operaciones de conjuntos.....	14
2.4 Funciones de membresía y parametrización.....	17
2.5 Razonamiento difuso.....	21
2.5.1 Principio de extensión.....	21
2.6 Control difuso.....	22
2.6.1 Controlador difuso.....	22
2.6.2 Fuzzy toolbox de matlab.....	23
2.7 Conclusiones.....	26
3. Modelo de un robot de dos grados de libertad.	
3.1 Introducción.....	27
3.2 Formulación Euler-Lagrange.....	27
3.3 Configuración del robot de dos grados de libertad.....	29
3.4 Energía cinética.....	30
3.5 Energía potencial.....	32
3.6 Lagrangiano.....	33
3.7 Ecuaciones de movimiento de Euler-Lagrange.....	33
3.8 Modelo cinemático directo e inverso de posición y velocidad.....	35

3.9 Evaluación del modelo dinámico y cinemático.....	36
3.9.1 Evaluación del sistema sin fricción.....	36
3.9.2 Evaluación del sistema con poca fricción.....	38
3.9.3 Evaluación del sistema con mucha fricción.....	40
3.10 Conclusiones.....	42
4. Control clásico de un robot.....	43
4.1 Introducción.....	43
4.2 Regulación del robot 2GDL.....	43
4.2.1 Controlador P y PD.....	44
4.2.2 Controlador PID.....	49
4.3 Seguimiento de trayectorias.....	51
4.3.1 Planificación de la trayectoria.....	51
4.3.2 Controlador PID.....	52
4.4 Controlador clásico con compensación de gravedad.....	55
4.5 Conclusiones.....	60
5. Control mediante lógica difusa de un robot de dos grados de libertad	
5.1 Introducción.....	61
5.2 Regulación del robot 2GDL mediante lógica difusa.....	61
5.3 Seguimiento de trayectorias del robot 2GDL mediante lógica difusa.....	81
5.4 Conclusiones.....	89
Conclusiones.....	90
Trabajos futuros.....	92
Glosario.....	93
Bibliografía.....	96

Tabla 2.1 Operación de conjuntos.....	15
Tabla 3.1 Parámetros y valores del sistema péndulo rotatorio.....	30
Tabla 5.1 Parámetros de las funciones de membresía de las variables de control.....	64
Tabla 5.2 Parámetros de las membresías de las variables de control respecto a las velocidades del sistema.....	64
Tabla 5.3 parámetros de las funciones de membresía de las salidas de control.....	65
Tabla 5.4 Matriz de control.....	66
Tabla 5.5 Parámetros de las funciones de membresía de las variables para cambio de variable.....	71
Tabla 5.6 Parámetros de las funciones de membresía de las salidas para el cambio de variable.....	72
Tabla 5.7 Parámetros de las funciones de membresía para nuevas reglas.....	74
Tabla 5.8 Parámetros de las funciones de membresía de salida para nuevas reglas.....	75
Tabla 5.9 Matriz de control.....	77
Tabla 5.10 Parámetros de las funciones de membresía de salida para nuevas reglas.....	85

Fig. 2.1 Función de membresía con universo discreto.....	9
Fig. 2.2 Función de membresía.....	10
Fig. 2.3 Función de membresía.....	11
Fig. 2.4 Propiedades de los conjuntos difusos.....	12
Fig. 2.5 Conjuntos difusos convexos y no convexos.....	13
Fig. 2.6 Concepto $A \subseteq B$	16
Fig. 2.7 Operaciones de conjuntos difusos.....	17
Fig. 2.8 Ejemplos de las funciones de membresía.....	19
Fig. 2.9 Varias funciones generadas con la función sigmoideal.....	20
Fig. 2.10 Interfaz de usuario de la herramienta de lógica difusa de matlab.....	24
Fig. 2.11 Interfaz de usuario de la herramienta de lógica difusa de matlab en sus diferentes funciones.....	25
Fig. 2.12 Consolidación de la lógica difusa sobre matlab simulink.....	25
Fig. 2.13 Control de un sistema masa-resorte en matlab simulink.....	26
Fig. 3.1 Robot de dos grados de libertad.....	29
Fig. 3.2 Movimiento y velocidades de las articulaciones del robot (sin fricción).....	37
Fig. 3.3 Movimientos en el espacio de trabajo (ws) del robot sin fricción.....	38
Fig. 3.4 Movimientos y velocidades de las articulaciones del robot (poca fricción).....	39
Fig. 3.5 Movimiento en el espacio de trabajo (ws) del robot (poca fricción).....	39
Fig. 3.6 Movimientos y velocidades de las articulaciones del robot (mucho fricción).....	41
Fig. 3.7 Movimientos en el espacio de trabajo (ws) del robot (mucho fricción).....	41
Fig. 4.1 Movimientos y velocidades de las articulaciones del robot con un controlador tipo P.....	45
Fig. 4.2 Movimiento en el espacio de trabajo del robot con un controlador tipo P.....	45
Fig. 4.3 Entradas de control o par generado por el controlador tipo P.....	46
Fig. 4.4 Movimiento y velocidades de las articulaciones del robot con un controlador PD.....	47
Fig. 4.5 Movimientos en el espacio de trabajo del robot con un controlador tipo PD.....	48
Fig. 4.6 Entradas de control o par generado por el controlador PD.....	48
Fig. 4.7 Movimientos y velocidades de las articulaciones del robot con un controlador tipo PID.....	50
Fig. 4.8 Movimiento en el espacio de trabajo del robot con un controlador PID... ..	50
Fig. 4.9 Entradas de control o par generado por un controlador PID.....	51
Fig. 4.10 Trayectoria propuesta para el controlador del manipulador.....	52
Fig. 4.11 Movimiento y velocidades de las articulaciones del robot con un controlador tipo PID para el caso de seguimiento de trayectorias.....	53

Fig. 4.12 Movimientos en el espacio de trabajo del robot con un control de tipo PID para el caso de seguimiento de trayectorias.....	54
Fig. 4.13 Entradas de control o par generado por el controlador PID para el caso de seguimiento de trayectorias.....	54
Fig. 4.14 Movimiento y velocidades de las articulaciones del robot con un controlador PD con compensación de gravedad.....	56
Fig. 4.15 Movimientos en el espacio de trabajo del robot con un control de tipo PD con compensación de gravedad.....	57
Fig. 4.16 Entradas de control o par generado por el controlador PD con compensación de gravedad.....	57
Fig. 4.17 Movimientos y velocidades de las articulaciones del robot con un controlador PD con compensación de gravedad.....	58
Fig. 4.18 Movimientos en el espacio de trabajo del robot con un controlador PD con compensación de gravedad.....	59
Fig. 4.19 Entradas de control o par generado por el controlador PD con compensación de gravedad para el seguimiento de trayectorias.....	59
Fig. 5.1 Determinación de las variables lingüísticas respecto al sistema.....	62
Fig. 5.2 Determinación de las variables lingüísticas respecto al sistema, segundo eslabón.....	63
Fig. 5.3 Función de membresía para el primer y segundo eslabón.....	63
Fig. 5.4 Función de membresía para la velocidad, primer y segundo eslabón.....	64
Fig. 5.5 Función de membresía para la generación de señales.....	65
Fig. 5.6 Proceso de defusificación.....	67
Fig. 5.7 Movimiento y velocidad de las articulaciones del robot con un controlador tipo PD difuso.....	68
Fig. 5.8 Movimientos en el espacio de trabajo del robot con un controlador tipo PD difuso.....	68
Fig. 5.9 Entrada de control o par generado por el controlador PID.....	69
Fig. 5.10 Movimientos y velocidades de articulación del robot con un controlador tipo PD difuso.....	70
Fig. 5.11 Movimiento en el espacio de trabajo del robot con un controlador tipo PD difuso.....	70
Fig. 5.12 Entradas de control o par generado por el controlador PID.....	71
Fig. 5.13 Movimientos y velocidades de las articulaciones del robot con un controlador tipo PD difuso.....	72
Fig. 5.14 Movimientos en el espacio de trabajo del robot con un controlador tipo PD difuso.....	73
Fig. 5.15 Entradas de control o par generado por el controlador PID.....	73
Fig. 5.16 Determinación de variables lingüísticas respecto al sistema.....	74
Fig. 5.17 Determinación de las variables lingüísticas respecto al sistema, segundo eslabón.....	75
Fig. 5.18 Función de membresía para la generación de señales de entrada.....	77
Fig. 5.19 Función de membresía para la generación de señales de entrada.....	78
Fig. 5.20 Proceso de defusificación con las nuevas reglas.....	78
Fig. 5.21 Movimiento y velocidades de las articulaciones del robot con un controlador tipo PD difuso.....	79

Fig. 5.22 Movimientos en el espacio de trabajo del robot con un controlador tipo PD difuso.....	80
Fig. 5.23 Entradas de control o par generado por el controlador PID.....	80
Fig. 5.24 Movimiento y velocidades de las articulaciones del robot con un controlador tipo PID difuso en seguimiento de trayectoria.....	81
Fig. 5.25 Movimiento en el espacio de trabajo del robot con un controlador tipo PID difuso para el seguimiento de trayectorias.....	82
Fig. 5.26 Entradas de control o par generado por el controlador PID para el seguimiento de trayectorias.....	83
Fig. 5.27 Movimiento y velocidades de las articulaciones del robot con un controlador PD difuso en seguimiento de trayectoria.....	84
Fig. 5.28 Movimiento en el espacio de trabajo del robot con un controlador tipo PD difuso para el seguimiento de trayectorias.....	84
Fig. 5.29 Entradas de control o par generado por el controlador PD para el seguimiento de trayectorias.....	85
Fig. 5.30 Movimiento y velocidades de las articulaciones del robot con un controlador tipo PD difuso en seguimiento de trayectorias.....	86
Fig. 5.31 Movimiento en el espacio de trabajo del robot con un controlador tipo PD difuso para el seguimiento de trayectorias.....	87
Fig. 5.32 Entradas de control o par generado por el controlador PD para el seguimiento de trayectorias.....	87
Fig. 5.33 Movimientos y velocidades de las articulaciones del robot con un controlador tipo PD difuso en seguimiento de trayectorias.....	88
Fig. 5.34 Movimiento en el espacio de trabajo del robot con un controlador tipo PD difuso para el seguimiento de trayectorias.....	89
Fig. 5.35 Entradas de control o par generado por el controlador PD para el seguimiento de trayectorias.....	89
Fig. 6.1 Movimiento en el espacio de trabajo del robot con un controlador PD con compensación de gravedad.....	91
Fig. 6.2 Movimiento en el espacio de trabajo con un controlador tipo PD difuso para el seguimiento de trayectorias.....	92

Capítulo 1

Introducción

1.1 Objetivos

Objetivo general: Desarrollar de una forma sencilla y eficiente el control de un robot de dos grados de libertad.

Objetivos específicos:

- Hacer un estudio de control mediante lógica difusa.
- Obtener el modelo dinámico de un robot de dos grados de libertad.
- Obtener las cinemáticas directas e inversas de sistema.
- Realizar un controlador mediante lógica difusa para el sistema en cuestión.
- Hacer un estudio comparativo con un controlador PD sobre el sistema bajo las mismas condiciones iniciales.

1.2 Justificación

En la actualidad el control de sistemas mediante técnicas clásicas es algo realmente sencillo, sin embargo el análisis de estabilidad para los mismos involucra una serie de cálculos muy complejos, que el usuario puede pasar por alto una vez obteniendo cierta experiencia del funcionamiento del sistema dinámico, por tal motivo el conocimiento empírico es una gran herramienta en la aplicación de controladores, ya que en ocasiones la elección de ganancias de control son encontradas a prueba y error. Por tal motivo el uso de un controlador empírico facilita el manejo de sistemas y por lo cual se piensa que es una

herramienta precisa y sencilla de aplicar, de ahí que este trabajo de investigación es enfocado al diseño de un controlador empírico mejor conocido como control difuso.

Los modelos con base en ecuaciones diferenciales lineales y no lineales son muy útiles para tener un antecedente del comportamiento de un sistema, en ingeniería la forma más común de verificar este comportamiento es mediante la simulación numérica, en la actualidad estas simulaciones se hacen mediante el software de Matlab en su versión de Simulink o en programación estructurada. Sin embargo la interpretación de las salidas de estos sistemas comúnmente se hace mediante un análisis gráfico, lo cual dificulta la comprensión del cómo se comporta el sistema en cierto tiempo, por lo que es necesaria una animación del sistema, y así visualizar de forma más cómoda dicho comportamiento.

1.3 Planteamiento del problema

Actualmente el uso de controladores clásicos es aplicado en un sinnúmero de procesos, estos controladores son diseñados para el cambio de un proceso a otro, clásicamente las operaciones de un robot pueden ser vistas como un proceso, es decir el cambio de una posición a otra. Sin embargo cuando se requiere que un robot realice una operación específica, se requiere de la planeación de tareas, es decir planificar una trayectoria, entonces el control clásico no es fiable para esta aplicación, por lo que se requiere del diseño de controladores muy complejos que realicen este trabajo sin problemas.

A principios de los años 60s Lofti A. Zadeh crea un nuevo concepto para el diseño de controladores, el cual se basa solo en el conocimiento de un experto, a esta lógica la llama lógica difusa, y sus aplicaciones aunque muy variadas son sencillas. De aquí que nace la idea de tratar de solucionar el problema de seguimiento de trayectorias mediante este novedoso tipo de control.

1.4 Solución propuesta

En esta tesis se propone desarrollar un controlador difuso para sistemas dinámicos, en este caso un robot de dos grados de libertad. Este controlador tendrá la ventaja de que será validado mediante un modelo dinámico, para así darle un mayor realismo a la implementación. Para ello se realizará el modelo de estos sistemas dinámicos, su simulación numérica, esta simulación se realiza mediante el software Matlab Simulink 2007b.

1.5 Metodología

Metodología para el diseño de un controlador difuso mediante la Fuzzy toolbox (herramienta de lógica difusa) en Matlab Simulink 2007b.

1. Obtener el modelo dinámico del sistema.
2. Validar el modelo mediante simulación numérica (en Matlab Simulink).
3. Diseñar las ganancias de un control clásico y aplicarlo a un manipulador.
4. Mediante la herramienta de lógica difusa de Matlab realizar un controlador con las características de un PD y un PID.
5. Comprobar los resultados mediante una simulación y discutir los resultados.

1.6 Contribuciones de la tesis

Las contribuciones de este trabajo de investigación se muestran a continuación:

- Obtener la simulación numérica del sistema dinámico.
- Desarrollo de un controlador difuso PD y PID
- Desarrollar un bosquejo de control basado en lógica difusa el cual pueda ser una herramienta extra de los controladores clásicos.
- Realizar un estudio comparativo de los controladores industriales más comunes con un controlador difuso.

1.7 Descripción del estado del arte

Esta simple idea nació en un artículo de Lotfi A. Zadeh publicado en 1965 y titulado "Fuzzy Sets" (Conjuntos Difusos). La lógica difusa permite representar de forma matemática conceptos o conjuntos imprecisos, tales como días fríos, meses calurosos, personas altas, salarios bajos, guisos con mucho condimento, profesores poco valorados.

Pero hay que tener en cuenta que la idea en sí de que las cosas no son blancas o negras, sino que existen infinitos matices de grises viene ya desde la época de los primeros grandes filósofos como Platón.

Posteriormente a ellos, otros grandes pensadores como David Hume o Kant apoyaban esta idea manteniendo que el razonamiento venía dado por las observaciones de las que somos testigos a lo largo de nuestra vida y la detección de algunos principios contradictorios en la lógica clásica.

Tras la publicación de Lotfi A. Zadeh, se comenzó rápidamente a usar la lógica difusa en distintas aplicaciones prácticas, llegando a su máximo auge a principios de los años 90, y continuando éste hasta la época actual.

A continuación se presenta una breve reseña histórica del control difuso.

- En los años 30's, Lukasiewicz Define la Lógica Multivaluada, como generalización de su lógica trivaluada (0, $\frac{1}{2}$, 1).
- En 1964, La noción de Conjunto Fuzzy aparece por primera vez en un memorándum de la universidad de Berkeley en California y es debida al ucraniano nacionalizado americano, Lofti Zadeh.
- En 1965, la revista "*Information and Control*" publica el memorándum anterior, en donde aparece el artículo de Zadeh, "*Fuzzy Sets*".
- En 1971, Zadeh publica el artículo, "Quantitative Fuzzy Semantics", en donde Introduce los elementos formales que acabarían componiendo el cuerpo de la doctrina de la lógica Difusa y sus aplicaciones tal como se conocen en la actualidad.
- En 1974, el Británico Ebrahim Mandani, demuestra la aplicabilidad de la lógica difusa en el campo del control. Desarrolla el primer sistema de control Fuzzy práctico, la regulación de un motor de vapor.
- A finales de los 70's, Los ingenieros daneses Lauritz Peter Holmbland y jens-Jurgen Ostergaard desarrollan el primer sistema de control difuso comercial, destinado a una planta de cemento.
- Los japoneses empiezan a explotar la lógica difusa de forma masiva. Los occidentales asumieron una actitud reacia principalmente por dos razones: la primera era porque la palabra "Fuzzy" sugería algo confuso y sin forma, y la segunda porque no había forma de probar analíticamente que la teoría funcionaba correctamente, ya que el control difuso no estaba basado en modelos matemáticos.
- Aparecen toda una serie de investigadores japoneses en el campo de la lógica difusa tales como Sugeno, Togai, Bart Kosko (el fuzsensei), entre otros.
- En 1986, Yamakawa, publica el artículo, "Fuzzy Controller hardward system". Desarrolla controladores difusos en circuitos integrados.
- En 1987, se inaugura en Japón el **subterráneo de Sendai**, uno de los más espectaculares sistemas de control difuso creados por el hombre. Desde entonces el controlador inteligente ha mantenido los trenes rodando eficientemente.
- En 1987, "**FUZZY BOOM**", Se comercializan multitud de productos basados en la lógica difusa (sobre todo en Japón).

Productos basados en la Lógica Difusa: Cámaras Fotográficas y de Vídeo, Electrodomésticos, alarmas, etc., controladores industriales, dispositivos médicos, y otros sistemas relativamente complejos. Dada la facilidad de la lógica difusa para representar conocimientos, se ha empleado también en la solución de problemas sociológicos, sicológicos, políticos, administrativos, económicos, epidemiológicos y de otras disciplinas.

Existen paquetes como el Fuzzy Decision Maker, que ayudan a las personas a tomar decisiones de todo tipo, como por ejemplo solucionar un problema familiar.

Sin embargo las aplicaciones de lógica difusa no se limitan a estos ejemplos, ya que se maneja un ambiente ambiguo esto tiene gran potencial en el desarrollo de controladores. Los mismos que pueden ser implementados a una gran variedad de sistemas, la gran ventaja es que el diseño se aproxima mediante el conocimiento de un experto (un conocimiento empírico), a diferencia de otros controladores que requiere el conocimiento de las dinámicas del sistema.

Actualmente el desarrollo de controladores basados en lógica difusa es muy abundante, no solamente para sistemas biológicos o bioquímicos, sino que también es aplicable a sistemas electromecánicos [10,11,12], pero la aplicación de robots manipuladores es basada simplemente a robots móviles limitando el uso de estas técnicas a robots manipuladores [13]. Por lo tanto las ventajas de usar sistemas controlados por lógica difusa pueden ser expresadas de la siguiente forma:

- Sistemas complejos que son difíciles o imposibles de modelar por métodos convencionales.
- Sistemas controlados por expertos Humanos.
- Sistemas que utilizan la observación humana como entrada o como base de las reglas.
- Sistemas que son confusos por naturaleza, como los encontrados en las ciencias sociales y del comportamiento

Capítulo 2

Lógica difusa y control difuso.

2.1 Introducción

En principios de siglo la introducción de lógica binaria fue un gran avance ya que comprende principios que pueden ser determinados de forma verdadera (1) y falsa (0), sin embargo el lenguaje natural del ser humano no es tan trivial, ya que la comunicación presenta una serie de combinaciones no ambiguas, es decir la toma de decisiones no es directamente relacionada con lógica binaria, si no que hace el uso de más de una opción para la toma de decisiones.

En 1964, L. Zadeh aplicó la lógica multivaluada a teoría de conjuntos, estableciendo la posibilidad de que los elementos de cierto conjunto pudieran tener diferentes grados de pertenencia. Zadeh introdujo el término *fuzzy* (borroso, difuso) y desarrollo un álgebra completa para los conjuntos difusos, aunque estos conjuntos no tenían aplicación.

A mediados de los años sesenta, cuando E. H. Mamdani diseñó el primer controlador basado en lógica difusa aplicado a un motor de vapor, y dando así aplicaciones prácticas al diseño de Zadeh.

2.2 Lógica difusa.

En la lógica difusa se trabaja con conjuntos, los cuales son definidos por funciones de membresía o pertenencia, que se denotan como $\mu_A(x)$ e indican el grado de pertenencia (entre 0 y 1) del elemento con valor x al conjunto A .

La lógica difusa permite tratar información de todo tipo e imprecisa, como *temperatura, fuerza, estatura, altura*; en términos de conjuntos difusos (definitivamente imprecisos). Estos conjuntos difusos se combinan con una serie de reglas para definir las acciones a realizar, por ejemplo, Si la temperatura es alta entonces enfriar demasiado, así la lógica difusa tiene gran impacto sobre el control de procesos ya que esté mismo puede ser diseñado por una persona experta la cual de las condiciones de operación.

Por lo tanto se puede decir que los sistemas basados en lógica difusa pueden ser aplicados a todo tipo de problemas, por lo que se facilita la aplicación a sistemas no lineales de control. Los sistemas difusos, al igual que las redes neuronales, pueden modelar procesos no lineales o no bien definidos y aprender de los datos haciendo el uso de determinados algoritmos de aprendizaje (como las propias de redes neuronales). Estos sistemas permiten hacer uso del conocimiento de un experto en un tema, así fortalece el conocimiento, a veces ambiguo del experto (o por sentido común). Y gracias a la simplicidad de cálculos, normalmente pueden aplicarse de forma barata y rápida.

Desde los resultados obtenidos por Zadeh y sus primera aplicaciones en procesos de control demostradas por Mamdani, el control difuso ha probado ser una excelente herramienta para el control de sistemas complejos.

A continuación se presentan aspectos básicos de la lógica difusa.

2.2.1 Conjuntos difusos y operaciones básicas.

La teoría de conjuntos clásicos es acotada, por ejemplo, un conjunto clásico A de números acotados por 6 puede ser expresado de la forma:

$$A = \{x \mid x > 6\} \quad (2.1)$$

donde es claro que la cota superior de x es 6, entonces x es generado por números y estos números son pertenecía del conjunto A . Sin embargo la naturaleza del ser humano, en la determinación de conjuntos, es diferente ya que puede ser abstracta e imprecisa, como en la aplicación de la estatura de una persona, está puede ser descrita por “alto, mediano o bajo” mientras que dichas estaturas pueden oscilar de 1.60 a 2 metros, y de esta manera poder clasificar a las personas.

En contraste un conjunto difuso es más general al establecer los elementos de un conjunto, de forma que se pueden establecer reglas que cumplan los elementos que pertenecen al conjunto, y este puede ser representado de la forma siguiente:

$$A = \{x \in U \mid \text{cumple ciertas condiciones}\}$$

donde U es denominado como universo de discurso o conjunto universal, el cual por definición contiene todos los elementos posibles que conciernen a cada contexto en particular. Recordando que un conjunto clásico (*crisp*) A , o simplemente el conjunto A , en el universo de discurso U se puede definir al especificar las propiedades que pueden satisfacer los elementos del conjunto, y para la comprensión del presente capítulo será necesaria una introducción a la terminología de conjuntos difusos, la cual se presenta a continuación:

Definición 2.3 *Conjuntos difusos y funciones de membresía.*

Si X es la colección de objetos denotados generalmente por x , entonces un **conjunto difuso** A en X es definido como un conjunto de pares ordenados de la forma:

$$A = \{(x, \mu_A(x)) \mid x \in X\} \tag{2.2}$$

donde $\mu_A(x)$ es la **función de membresía** (o **FM de forma corta**) para un conjunto difuso A , La FM mapea cada elemento de X a un grado de membresía (o valor de membresía) entre 0 y 1.

Obviamente, la definición de un conjunto difuso es una extensión de la definición de conjunto clásico en el que la función característica tiene valores entre 0 y 1. Si el valor de la MF $\mu_A(x)$ es estricta sobre 0 y 1, entonces A es reducido a un conjunto clásico y $\mu_A(x)$ es la función característica de A , Donde X se refiere al universo de discurso, o simplemente al universo y este puede consistir de objetos discretos (ordenados o no) o espacios continuos. Esto puede ser visto más claramente en el siguiente ejemplo.

Ejemplo 2.1 *Conjuntos difusos con universo no ordenado discreto.*

Sea $X = \{\text{México, Puebla, Monterrey}\}$ un conjunto de ciudades donde uno puede vivir. El conjunto difuso $C = \text{''Ciudad donde deseo vivir''}$ puede ser descrita de la siguiente forma:

$$C = \{(\text{México}, 0.6), (\text{Puebla}, 0.4), (\text{Monterrey}, 0.8)\}$$

Aparentemente el universo de discurso X es discreto y contiene objetos no ordenados, en este caso, tres grandes ciudades de la República Mexicana.

Como se puede ver en el ejemplo anterior, el universo no es definido de una forma ordenada, entonces se pueden obtener conjuntos difusos ordenados.

Ejemplo 2.2 *Conjunto difuso con universo discreto ordenado.*

Sea $X = \{0,1,2,3,4,5,6\}$ el número de niños que una familia puede tener. Entonces el conjunto $A = \text{''el número de niños que tiene una familia''}$ puede ser descrito de la siguiente forma:

$$A = \{(0,0.1), (1,0.3), (2,0.7), (3,1), (4,0.7), (5,0.3), (6,0.1)\}$$

Aquí se tiene un universo X ordenado; la MF para el conjunto difuso puede ser visto en la figura 2.1

Como se puede apreciar el nivel o grado de membresía puede ser dado por cualquier persona, sin embargo pueden existir conjuntos difusos en los que el grado de membresía es muy similar para todos, esto se puede ver en el siguiente ejemplo.

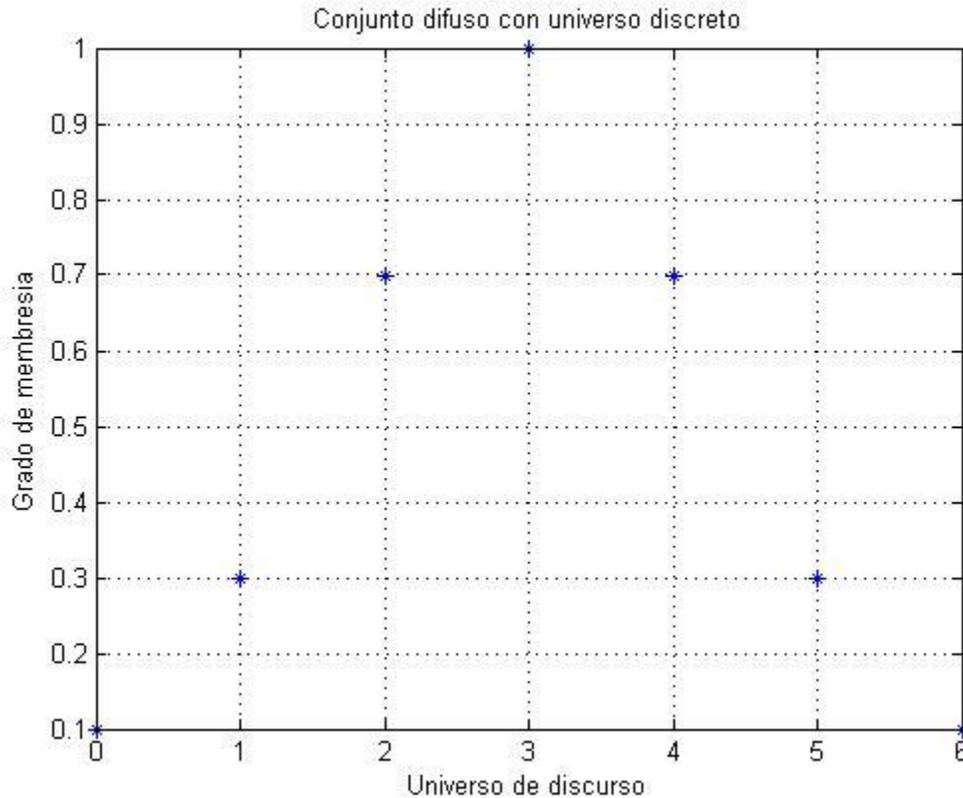


Fig. 2.1: Función de membresía con universo discreto.

Ejemplo 2.3 *Conjunto difuso de un universo continuo.*

Sea $X = \mathbb{R}^+$ el conjunto de las edades posibles de los seres humanos. Entonces el conjunto difuso $B =$ "edad alrededor de 50 años" puede ser expresada como:

$$B = \left\{ (x, \mu_B(x)) \mid x \in X \right\} \tag{2.3}$$

donde

$$\mu_B(x) = \frac{1}{1 + \left(\frac{x - 50}{10} \right)^4} \tag{2.4}$$

Esto es ilustrado por la figura 2.2

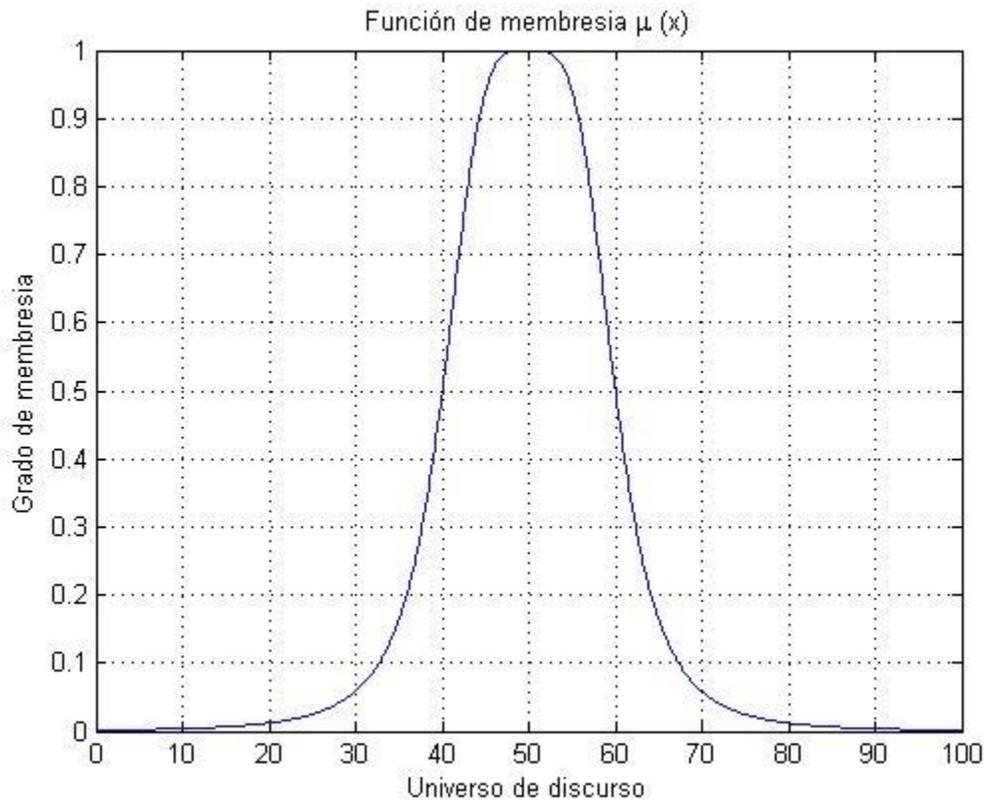


Fig. 2.2: Función de membresía.

De los ejemplos anteriores puede ser visto que es posible construir conjuntos difusos los cuales dependen de dos variables: la identificación del universo de discurso y las especificaciones de una función de membresía apropiada. La especificación de una función de membresía es del tipo *subjetivo*, conceptos aparentemente similares pero con pequeñas diferencias, como lo es la edad, (es decir, el número de niños que puede tener una familia”) la cuales son definidas según una encuesta lo cual da un fenómeno del tipo aleatorio.

Para simplificar la notación, puede ser introducido un camino alternativo el cual denota un conjunto difuso. Un conjunto difuso A puede ser expresado de la siguiente manera:

La primera se cumple si X es una colección de objetos discretos, y la segunda se cumple si X es una colección de espacios continuos. La sumatoria e integral de la función anterior indican la intersección de los pares $(x, \mu_A(x)/x)$, donde de forma similar el operador $/$ no indica la división explícita.

$$A = \left\{ \begin{array}{ll} \sum_{x_i \in X} \mu_A(x_i) / x_i & 1^a \\ \int_X \mu_A(x) / x & 2^a \end{array} \right\} \quad (2.5)$$

Ejemplo 2.4 *Expresión alternativa.*

Usando la notación de la expresión (2.5), puede ser reescrita de forma de un conjunto difuso como en el ejemplo 2.1, 2.2 y 2.3 como:

$$C = \{(México, 0.6), (Puebla, 0.4), (Monterrey, 0.8)\}$$

$$A = \{(0, 0.1), (1, 0.3), (2, 0.7), (3, 1), (4, 0.7), (5, 0.3), (6, 0.1)\}$$

y

$$B = \int_{\mathbb{R}^+} \frac{1}{1 + \left(\frac{x-50}{10}\right)^4} / x,$$

Respectivamente.

Prácticamente cuando el universo de discurso X , es un espacio continuo, a las funciones de membresía de ese universo de discurso se les representa con adjetivos o también denominados como variables lingüísticas, tal es el caso de grande, muy grande o pequeño. Un ejemplo de esto puede ser visto en la figura 2.3.

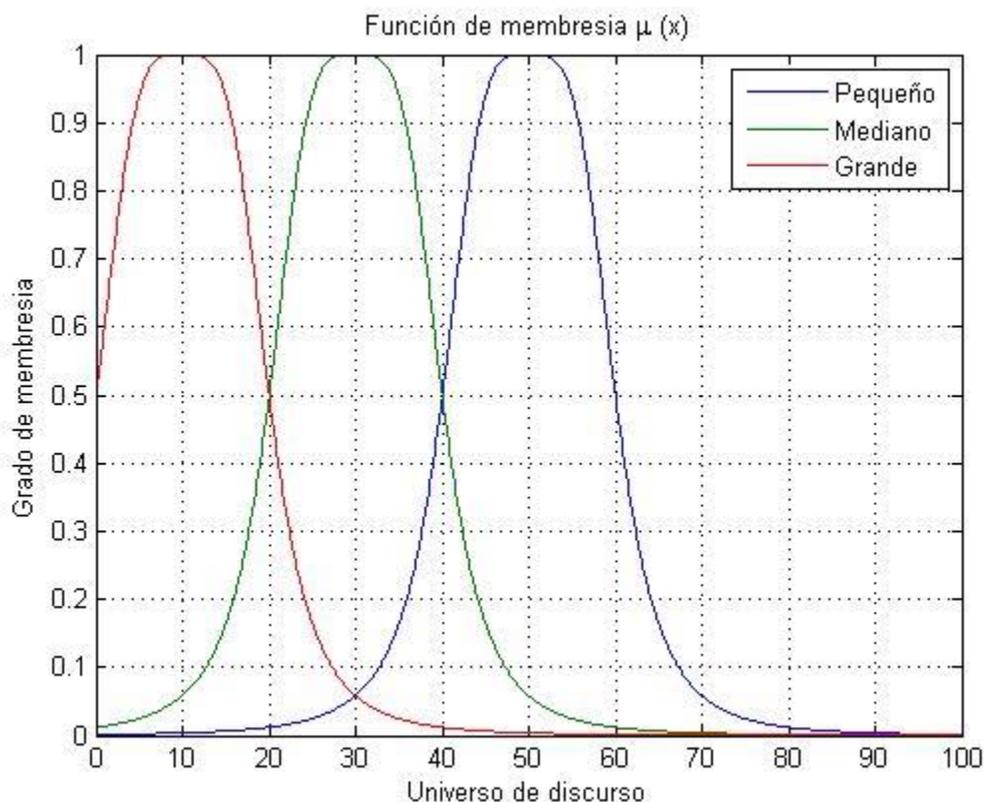


Fig. 2.3: Función de membresía.

Estas funciones de membresía, de en la figura anterior, tienen ciertas características que definen un conjunto difuso, de las cuales se mencionan algunas definiciones.

Definición 2.2 Soporte.

El soporte de un conjunto difuso A es el conjunto de todos los puntos x en X tal que $\mu_A(x) > 0$.

$$\text{soporte}(A) = \{x \mid \mu_A(x) > 0\} \tag{2.6}$$

Definición 2.3 Cresta.

La cresta de un conjunto difuso A es el conjunto de todos los puntos x en X , tal que $\mu_A(x) = 1$.

$$\text{cresta}(A) = \{x \mid \mu_A(x) = 1\} \tag{2.7}$$

Definición 2.4 Normal.

Un conjunto difuso A es normal si la cresta es no vacía. En otras palabras, siempre que se puede encontrar un punto x en X , tal que $\mu_A(x) = 1$.

Definición 2.5 Puntos de cruce.

Un punto de cruce de un conjunto difuso A es un punto $x \in X$ donde $\mu_A(x) = 0.5$

$$\text{punto de cruce}(A) = \{x \mid \mu_A(x) = 0.5\} \tag{2.8}$$

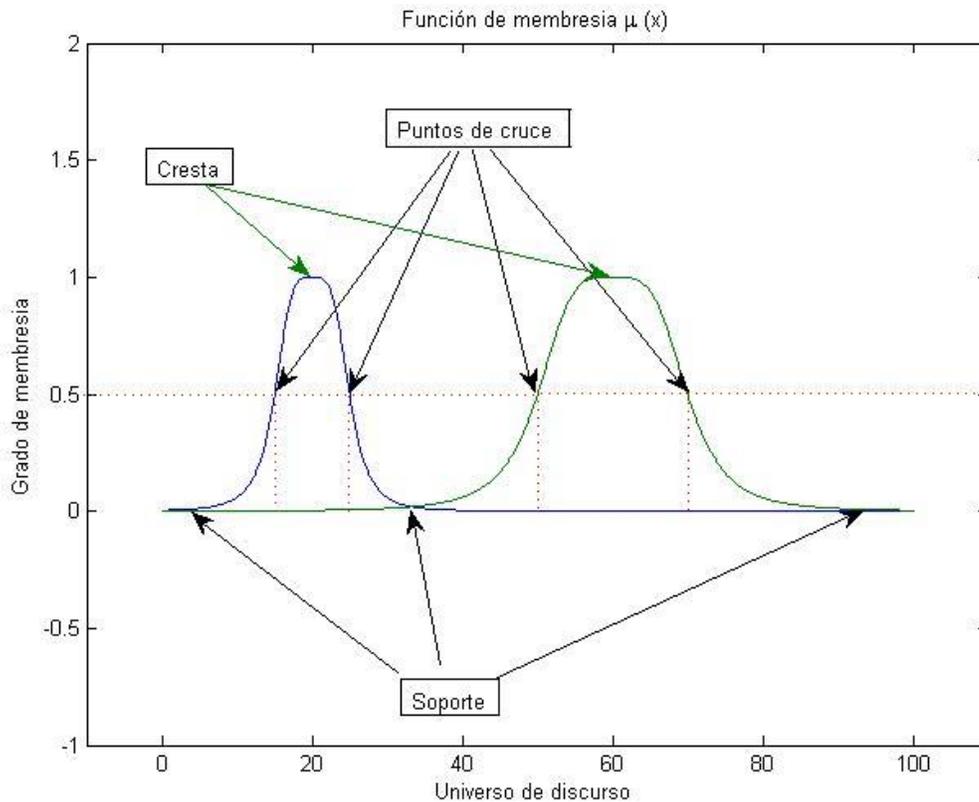


Fig. 2.4: Propiedades de los conjuntos difusos.

Las propiedades definidas anteriormente, pueden ser vistas en la figura 2.4, la cual muestra el soporte, la cresta y los puntos de cruce de un conjunto difuso.

Definición 2.6 *Impulso difuso.*

Un conjunto difuso que tiene un soporte un punto en X con $\mu_A(x)=1$ es llamado impulso difuso.

Definición 2.7 α de corte, α de corte fuerte.

El α de corte o nivel α de corte de un conjunto difuso A es un conjunto cresta definido por:

$$A_\alpha = \{x | \mu_A(x) \geq \alpha\} \tag{2.9}$$

Mientras que el nivel α de corte fuerte, se define por el conjunto siguiente:

$$A_\alpha = \{x | \mu_A(x) > \alpha\} \tag{2.10}$$

Definición 2.8 *Conjunto Convexo.*

Un conjunto difuso A es convexo si y solo si para cualquier $x_1, x_2 \in X$ y cualquier $\lambda \in [0, 1]$,

$$\mu_A(\lambda x_1 + (1-\lambda)x_2) \geq \min\{\mu_A(x_1), \mu_A(x_2)\} \tag{2.11}$$

Alternadamente, A es convexo si todo conjunto de nivel α son convexos.

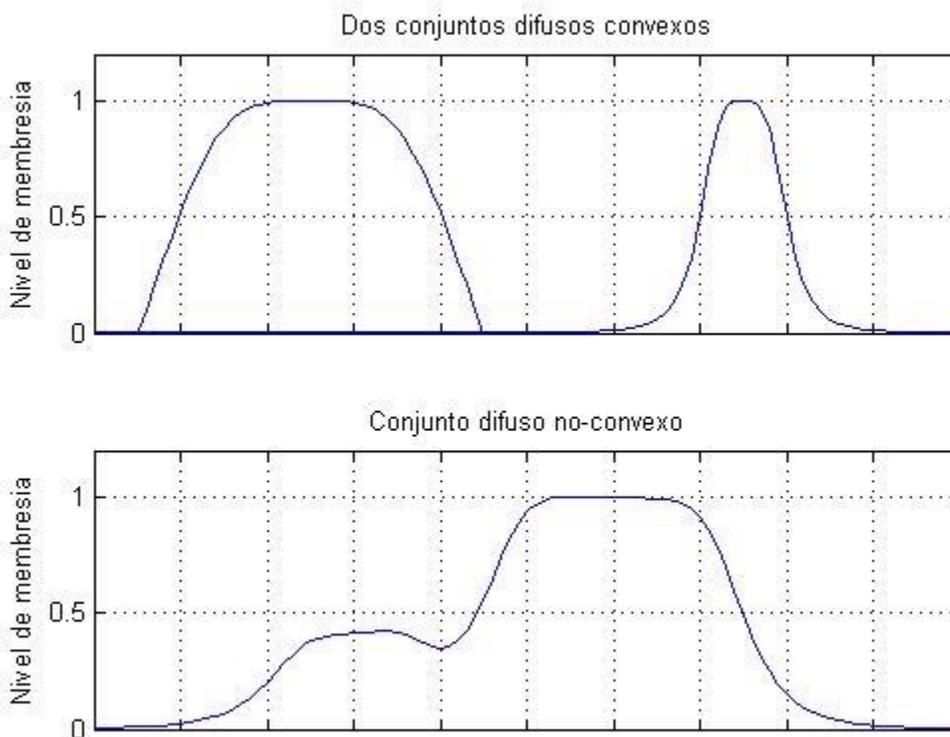


Fig. 2.5: Conjuntos difusos convexos y no convexo.

Definición 2.9 *Números difusos.*

Un número difusa A es un conjunto difuso en la línea real (\mathbb{R}) tal que cumple con las condiciones de convexidad y normalidad.

Definición 2.10 *Ancho de banda de un conjunto difuso normal y convexo.*

Para un conjunto difuso normal y convexo, el ancho de banda o banda es definido como la distancia entre los dos únicos puntos de cruce:

$$\text{banda}(A) = |x_2 - x_1| \tag{2.12}$$

Donde $\mu_A(x_1) = \mu_A(x_2) = 0.5$.

Definición 2.11 *Conjuntos abiertos, por la derecha e izquierda, y conjuntos cerrados.*

Un conjunto difuso A es abierto por la derecha, si $\lim_{x \rightarrow -\infty} \mu_A(x) = 1$ y $\lim_{x \rightarrow +\infty} \mu_A(x) = 0$; abierto

por la izquierda si $\lim_{x \rightarrow -\infty} \mu_A(x) = 0$ y $\lim_{x \rightarrow +\infty} \mu_A(x) = 1$; y cerrado si

$$\lim_{x \rightarrow -\infty} \mu_A(x) = \lim_{x \rightarrow +\infty} \mu_A(x) = 0.$$

Una vez entendidas las definiciones de conjuntos difusos, entonces se debe estudiar las operaciones que existen entre ellos, de igual manera que la teoría clásica de conjuntos, los conjuntos difusos se caracterizan por las mismas operaciones fundamentales. En la siguiente sección se muestran dichas operaciones.

2.3 Operaciones de conjuntos.

La unión, intersección, y complemento son algunas de las operaciones básicas de conjuntos clásicos. De la base de estas operaciones un número de identidades puede ser definido, como una lista (ver tabla 2.1). Estas tablas pueden ser verificadas al usar diagramas de ven.

Ley de contradicción	$A \cap \bar{A} = 0$
Ley de exclusión media	$A \cup \bar{A} = X$
Idempotencia	$A \cap A = A, A \cup A = A$
Involución	$\bar{\bar{A}} = A$
Conmutativa	$A \cup B = B \cup A, A \cap B = B \cap A$
Asociativa	$(A \cup B) \cup C = A \cup (B \cup C)$ $(A \cap B) \cap C = A \cap (B \cap C)$
Distributiva	$A \cup (B \cap C) = (A \cup B) \cap (A \cup C)$ $A \cap (B \cup C) = (A \cap B) \cup (A \cap C)$
Absorción	$A \cup (A \cap B) = A$ $A \cap (A \cup B) = A$
Absorción de complementos	$A \cup (\bar{A} \cap B) = A \cup B$ $A \cap (\bar{A} \cup B) = A \cap B$

Tabla 2.1: Operaciones de conjuntos.

Después de ver las operaciones básicas de conjuntos (unión, intersección u complemento), Zadeh introdujo los mismos conceptos en lógica difusa. Antes de introducir estos tres conceptos de operaciones de conjuntos, primero se definirán algunas nociones de conjuntos difusos.

Definición 2.12 *Subconjunto* (\subseteq).

Un conjunto difuso A contenido en un conjunto B (esto equivale a decir, A es subconjunto de B) si y solo si $\mu_A(x) \leq \mu_B(x)$ para todo x . En notación matemática esto es:

$$A \subseteq B \Leftrightarrow \mu_A(x) \leq \mu_B(x) \tag{2.13}$$

El concepto de la definición 2.12 se puede ver en la figura 2.6.

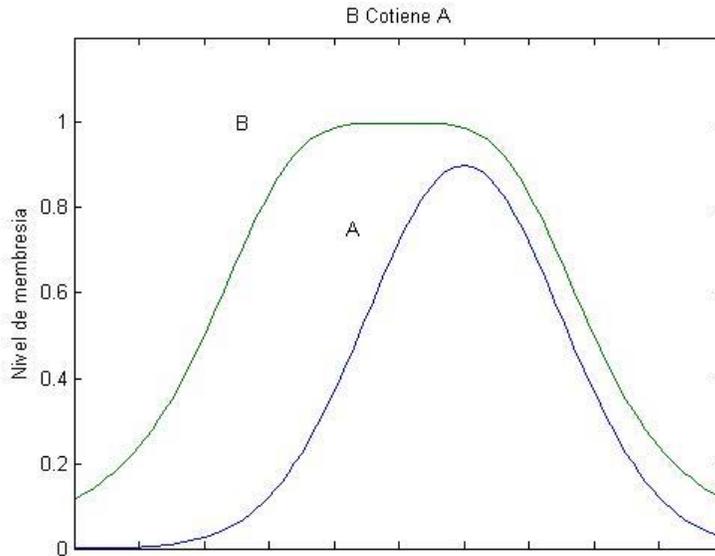


Fig. 2.6: Concepto $A \subseteq B$.

Definición 2.13 Unión.

La unión de dos conjuntos difusos A y B es un conjunto difuso C , escrito como $C = A \cup B$ o $C = A \text{ OR } B$, donde la función de membresía es la relación entre A y B dada por:

$$\mu_C(x) = \max(\mu_A(x), \mu_B(x)) = \mu_A(x) \vee \mu_B(x) \tag{2.14}$$

Definición 2.15 Intersección.

La intersección de dos conjuntos difusos A y B es un conjunto difuso C , escrito como $C = A \cap B$ o $C = A \text{ AND } B$, donde la función de membresía de es la relación de A y B de la forma

$$\mu_C(x) = \min(\mu_A(x), \mu_B(x)) = \mu_A(x) \wedge \mu_B(x) \tag{2.15}$$

Definición 2.16 Complemento.

El complemento de un conjunto difuso A , denotado por \bar{A} , $(\neg A, \text{NOTA})$, es definido como:

$$\mu_{\bar{A}}(x) = 1 - \mu_A(x) \tag{2.16}$$

La figura 2.7 muestra en detalle las definiciones 2.14, 2.15 y 2.16.

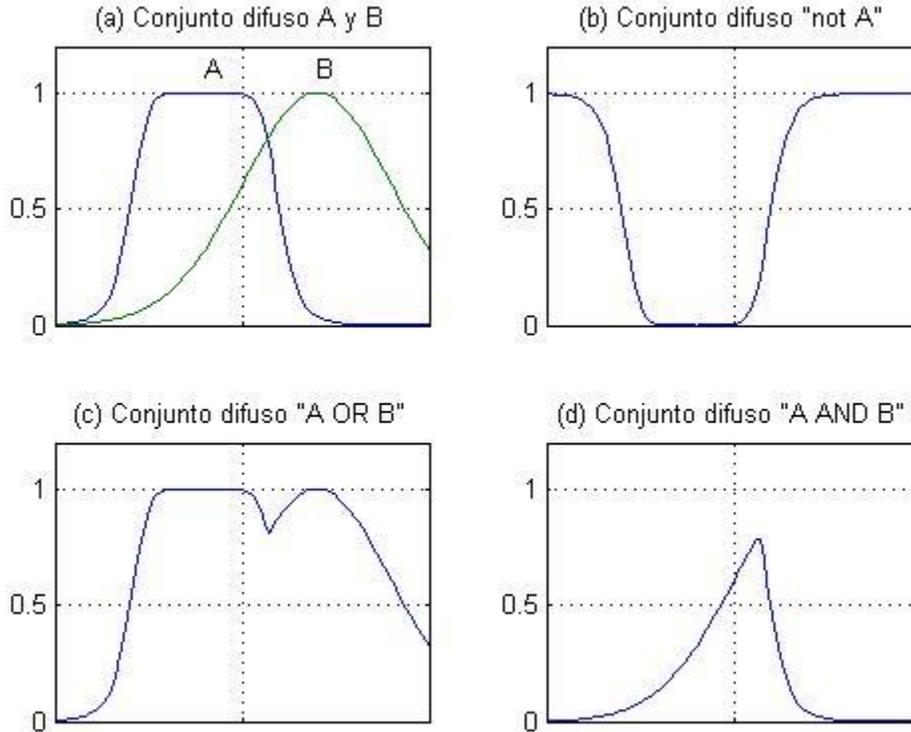


Fig. 2.7: Operaciones de conjuntos difusos.

Definición 2.17 *Producto cartesiano y co-producto.*

Sean A y B conjuntos difusos en X e Y , respectivamente. El producto cartesiano de A y B , denotado por $A \times B$, es un conjunto difuso en el producto espacial $X \times Y$ con un función de membresía

$$\mu_{A \times B}(x, y) = \min(\mu_A(x), \mu_B(x)). \tag{2.17}$$

De forma similar, el co-producto cartesiano $A + B$ es un conjunto difuso con función de membresía:

$$\mu_{A+B}(x, y) = \max(\mu_A(x), \mu_B(x)). \tag{2.18}$$

Ambos productos $A \times B$ y $A + B$ son caracterizado por dos funciones de membresía bidimensionales.

2.4 Funciones de membresía y parametrización.

Como se menciono anteriormente, la caracterización de una función de membresía es dada por ciertos aspectos básicos, como el tener un universo de discurso, una cresta, entre otras características, sin embargo estas funciones pueden tener distintas formas, que hasta cierto punto son representativas por la forma de las mismas. Actualmente se han definido diferentes tipos de funciones de membresía (de una y dos dimensiones). como este proyecto de tesis solo se enfoca a una dimensión solo se darán las definiciones pertinentes es este sentido.

Definición 2.18 *Función de membresía tipo triangular.*

Una función de membresía tipo triangular es especificada por tres parámetros $\{a, b, c\}$, como sigue:

$$\text{Triangular}(x; a, b, c) = \begin{cases} 0, & x \leq a \\ \frac{x-a}{b-a}, & a \leq x \leq b \\ \frac{c-x}{c-d}, & b \leq x \leq c \\ 0, & c \leq x \end{cases} \quad (2.19)$$

Al usar el máximo y mínimo, se tiene una expresión alternativa para la ecuación anterior:

$$\text{Triangular}(x; a, b, c) = \max \left(\min \left(\frac{x-a}{b-a}, \frac{c-x}{c-d} \right), 0 \right) \quad (2.20)$$

donde los parámetros $\{a, b, c\}$ determinan las coordenadas en x de las tres esquinas de la función de membresía (la función de membresía tipo triangular definida en $\{x; a, b, c\} = \{x; 20, 60, 80\}$ puede ser vista en la figura 2.8 a).

Definición 2.19 *Función de membresía tipo Trapezoidal.*

Una función de membresía tipo trapezoidal es definida por cuatro parámetros $\{a, b, c, d\}$ de la siguiente forma:

$$\text{Trapezoidal}(x; a, b, c, d) = \begin{cases} 0, & x \leq a \\ \frac{x-a}{b-a}, & a \leq x \leq b \\ 1, & b \leq x \leq c \\ \frac{d-x}{d-c}, & c \leq x \leq d \\ 0 & d \leq x \end{cases} \quad (2.21)$$

De forma similar a la anterior, esta función de membresía coincide con las expresiones de máximo y mínimo, y esta puede ser representada de la siguiente forma:

$$\text{Triangular}(x; a, b, c, d) = \max \left(\min \left(\frac{x-a}{b-a}, 1, \frac{d-x}{d-c} \right), 0 \right) \quad (2.22)$$

Los parámetros $\{a, b, c, d\}$ (con $a < b \leq c < d$) determinan las coordenadas en x de las cuatro esquinas de la función tipo trapezoidal, en la figura 2.8 b se puede ver una

función de membresía tipo trapezoidal cuyas esquinas están en las coordenadas $\{x; a, b, c, d\} = \{x; 10, 50, 60, 95\}$.

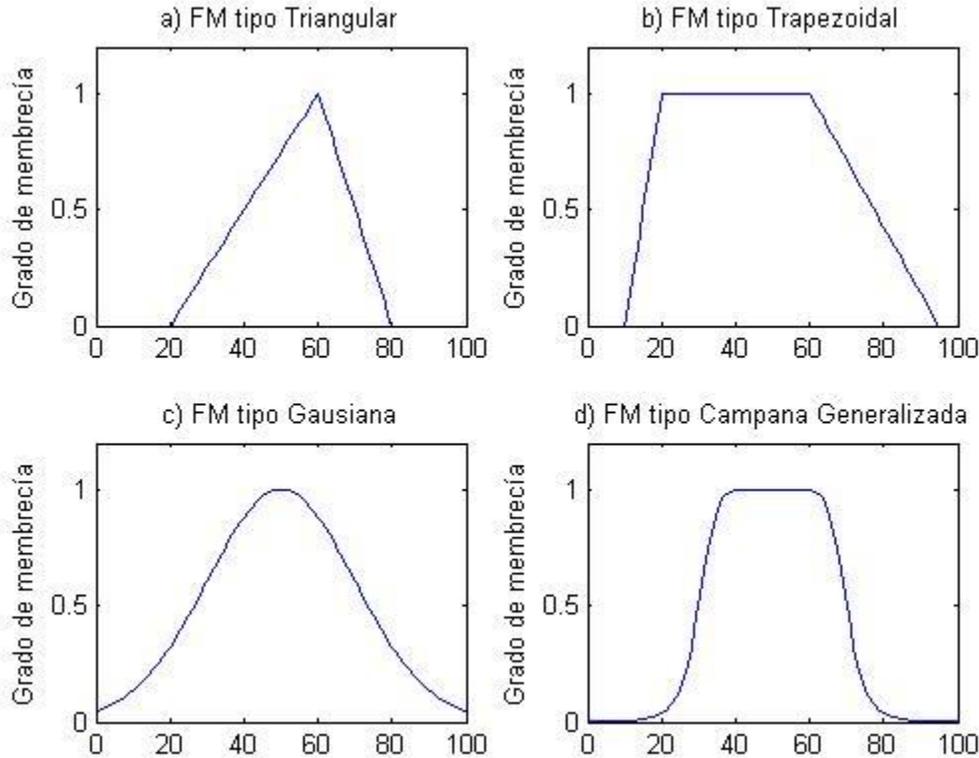


Fig. 2.8: Ejemplos de las funciones de membresía.

Definición 2.20 *Función de membresía tipo Gausiana.*

Una función de membresía del tipo Gausiana es especificada por dos parámetros $\{c, \sigma\}$:

$$\text{Gausiana}(x; c, \sigma) = e^{-\frac{1}{2} \left(\frac{x-c}{\sigma} \right)^2} \tag{2.23}$$

Una función de membresía del tipo Gausiana es determinada por los componentes c y σ los cuales representan el centro de y el ancho de banda de la función. En la figura 2.8 c se puede ver una función del tipo Gausiana con centro $c = 50$ y ancho de banda $\sigma = 20$ expresada por $\{x; 50, 20\}$.

Definición 2.21 *Función de membresía tipo campana generalizada.*

Una función de membresía del tipo campana generalizada es especificada por tres parámetros $\{a, b, c\}$:

$$\text{Campana}(x; a, b, c) = \frac{1}{1 + \left| \frac{x-c}{a} \right|^{2b}}, \quad (2.24)$$

donde el parámetro b usualmente es positivo.

En la figura 2.8 d se ilustra un ejemplo de la función de membresía del tipo campana generalizada, cuyos parámetros son $(x; 20, 4, 50)$.

Definición 2.22 *Función de membresía tipo sigmoidal.*

Una función de membresía del tipo sigmoidal se define da la siguiente forma:

$$\text{Sigmoidal}(x; a, c) = \frac{1}{1 + e^{-a(x-c)}}, \quad (2.24)$$

donde a es el control de la pendiente en el punto de cruce $x = c$.

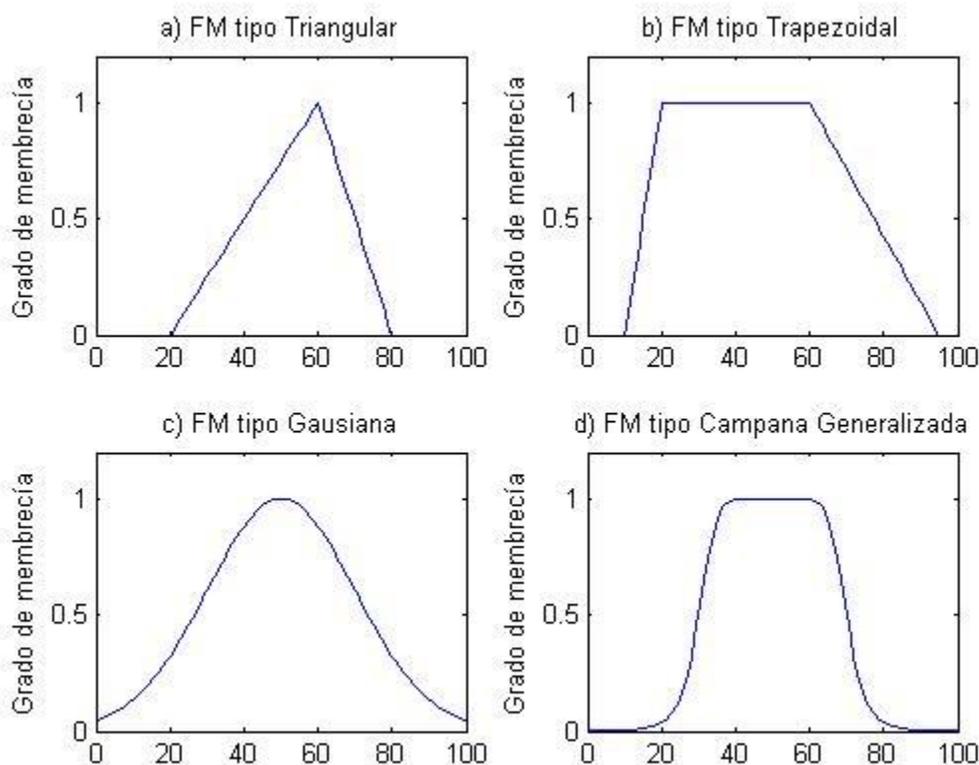


Fig. 2.9: Varias funciones de membresía generadas con la función Sigmoidal.

En la figura 2.9 se pueden ver varias funciones del tipo sigmoidal, donde los parámetros de ajuste son los siguientes, figura 2.9 a $y_1 = \text{sig}(x; 1, 5)$, figura 2.9 b $y_2 = \text{sig}(x; 2, 5)$, figura 2.9 c $y_3 = \text{sig}(x; 2, -5)$ y por último la figura 2.9 d como el producto de y_1 con y_3 .

2.5 Razonamiento difuso.

Las reglas y razonamiento difuso son un bloque del sistema de inferencia difusa, las cuales tienen propiedades particulares sobre los sistemas difusos y son muy importantes para las herramientas de la base de teoría difusa. Esta tiene gran importancia sobre la aplicación de control automático, sistemas expertos, reconocimiento de patrones, etc.

2.5.1 Principio de extensión.

El principio de extensión [] es una base de los conceptos de teoría difusa la cual provee la extensión de crestas de un dominio matemático a un dominio difuso. Estos procedimientos generalizan un mapeo punto a punto común de un dominio difuso.

Para ser más específicos, suponemos que f es una función de X en Y , y A es un conjunto difuso en X definido como:

$$A = \mu_A(x_1)/x_1 + \mu_A(x_2)/x_2 + \dots + \mu_A(x_n)/x_n \quad (2.25)$$

Entonces el principio de extensión indica que la imagen de un conjunto difuso A , bajo el mapeo $f(\cdot)$ puede ser expresado como un conjunto difuso B

$$B = f(A) = \mu_A(x_1)/y_1 + \mu_A(x_2)/y_2 + \dots + \mu_A(x_n)/y_n \quad (2.26)$$

donde $y_i = f(x_i)$, $i = 1, \dots, n$. En otras palabras el conjunto difuso B , puede ser encontrado al calcular los valores de $f(\cdot)$ en x_1, \dots, x_n . Si $f(\cdot)$ contiene más de un mapeo, entonces existirán $x_1, x_2 \in X$, $x_1 \neq x_2$, tal que $f(x_1) = f(x_2) = y^*$, $y^* \in Y$. En este caso, el grado de membresía de B en $y = y^*$ es el máximo de los grados de membresía de A en $x = x_1$ y en $x = x_2$, dado que $f(x) = y^*$ puede resultar que $x = x_1$ o $x = x_2$. De forma más general:

$$\mu_B(y) = \max_{x \rightarrow f^{-1}(y)} (\mu_A(x)) \quad (2.27)$$

Ejemplo 2.5 Aplicación del principio de extensión.

Sea

$$A = 0.1/-3 + 0.4/-1 + 0.8/0 + 0.9/1 + 0.3/2$$

y sea

$$f(x) = x^3 - 3$$

Bajo el principio de extensión se tiene:

$$B = 0.1/1 + 0.4/-2 + 0.8/-3 + 0.9/-2 + 0.3/1$$

$$B = (0.4 \vee 0.9)/-2 + 0.8/-3 + (0.1 \vee 0.3)/1$$

$$B = 0.8/-3 + 0.9/-2 + 0.3/1$$

2.6 Control difuso.

En esta sección se verifica la importancia de la lógica difusa sobre problemas de control, para esto se presenta la composición del controlador que es básicamente la siguiente:

1. Una base de reglas (un conjunto de sentencias denominadas reglas IF-THEN), que contiene la cuantificación de la lógica difusa dada por una descripción lingüística del un experto para hacer un mejor control.
2. Un mecanismo de inferencia (también llamada máquina de inferencia o modulo de inferencia difusa), el cual emula la experiencia del experto en la toma de decisiones haciendo una aproximación al aplicar el conocimiento de control.
3. Una interfase de fusificación, la cual convierte las entradas de control en información para que el mecanismo de inferencia pueda ser fácil de aplicar y activar las reglas de control.
4. Una interfase de defusificación, la cual convierte la conclusión del mecanismo de inferencia de la actual entrada en la salida de control del proceso.

Estos cuatro aspectos que se involucran dentro del control difuso, fueron vistos de manera inherente en las secciones 2.2 y 2.3. La primera parte de esta sección se hará una introducción a un controlador difuso, la segunda parte a la herramienta difusa de Matlab (fuzzy toolbox).

2.6.1 Controlador difuso.

Para el proceso de control, como primer paso se deben identificar las salidas y entradas del sistema. Así como los rangos de operación (dando el uso del conocimiento de un experto sobre el proceso de control).

Una vez identificadas estas variables, se sigue con definir las funciones de error, es decir que variables se deben controlar y que posiciones o referencias se desean, de esta manera el uso del conocimiento del experto es aplicado de nueva forma. Después se sigue con el diseño de las funciones de membresía, para definir los rangos de operación, que la función de error sea pequeña positiva, positiva o muy positiva, esta función puede ser cero (es decir que el sistema está en el lugar indicado), pequeña negativa, negativa y muy negativa. De igual forma definir la salida del sistema que puede ser manejada de forma similar a las funciones de membresía de las entradas con el simple cambio de que el conocimiento del experto es aplicado nuevamente para definir la fuerza, par de fuerza, incremento de biomasa, entre otras salidas de control.

Después de tener estos dos pasos, nuevamente es requerida la ayuda del experto de la planta para poder identificar las acciones de control, es decir, un experto toma las decisiones de control de la forma o con las sentencias “*si ocurre este caso, además ocurre este otro... entonces la salida de control será la siguiente*” este proceso en el campo de la lógica difusa es mejor conocido como el proceso de fusificación.

El último paso del proceso de control mediante lógica difusa es el convertir esta serie de condiciones en salidas que el sistema pueda interpretar como salidas de control “*fuerza, par de fuerza, incremento de biomasa, entre otras salidas de control*” este proceso puede ser determinado (como se vio en la sección anterior) con el método de “*centro de gravedad, bisector de área o principal máximo*”, este proceso es mejor conocido como la interfase de defusificación.

Una vez visto a detalle el procedimiento que se debe seguir para el control de sistemas mediante lógica difusa, lo siguiente es programar esta base de conocimientos en algún lenguaje de programación. Los lenguajes de programación más comunes son C, C++, visual C, basic, visual basic, entre otros, sin embargo actualmente por comodidad el uso de programación basada en el lenguaje C es básico, ya que muchos de los compiladores experto en algún tipo de área son basados en este compilador (Multisim, PS-SPICE, Matemática, Matlab, LaTeX, entre otros), este proyecto se usa el compilador Matlab con su herramienta Simulink y su librería de lógica difusa.

2.6.2 Fuzzy toolbox de Matlab.

Como se ha visto en este capítulo, las operaciones difusas entre conjuntos, el proceso de defusificación, entre otros, es mediante programación estructurada, sin embargo el software científico Matlab con su herramienta de Matlab incluye librerías que pueden ser utilizadas con este fin. En esta sección se presenta una breve descripción del funcionamiento de la herramienta para control difuso de Matlab.

El uso de esta herramienta, como primer paso se debe inicializar Matlab (con Simulink en sus versiones 6.1 o superiores), e introducir el comando fuzzy lo cual permite que se despliegue la siguiente ventana tipo interfaz de usuario (GUI).

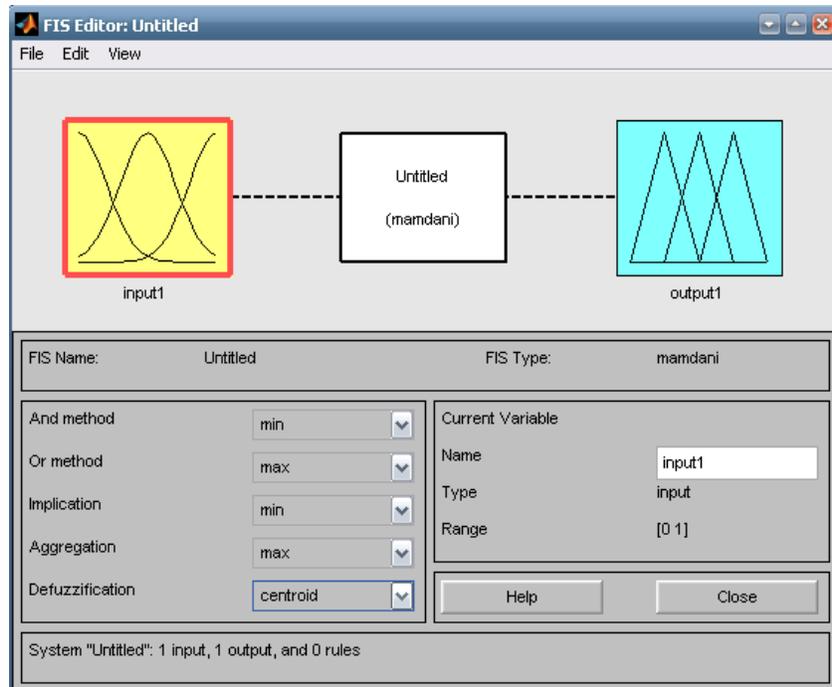


Fig. 2.10: Interfaz de usuario de la herramienta de lógica difusa de Matlab.

Esta ventana tiene diferentes funciones, por ejemplo se pueden generar nuevas funciones de entrada o salida, así como cambiar el tipo de lógica (Tipo Mamdani o Sugeno), de igual forma se pueden introducir o modificar las funciones de membresía, los nombres de las mismas, el método (para este caso como se ha manejado en secciones anteriores se maneja el método máx-mín), modificar los rangos de el universo de discurso, crear o modificar reglas y el método de defusificación (estos cambios son producidos directamente en el menú principal y puede ser visto en la figura 2.10).

Una vez modificados los parámetros de lógica difusa, es decir, después de que se verifica que la lógica diseñada sea la adecuada para su aplicación, entonces este modulo se guarda o exporta (de forma permanente a un archivo y para poder aplicarlo sobre Simulink a workspace), esto se puede ver en la figura 2.11. El siguiente paso es abrir o crear un archivo de Simulink, después sobre la librería de las herramientas seleccionar la fuzzy toolbox de la forma en que se muestra en la figura 2.12., una vez creada esta función sobre el nuevo archivo, el siguiente paso es dar doble clic sobre el bloque (fuzzy logic controller) e introducir el nombre con el que se exporto al modulo de lógica difusa sobre el workspace de Matlab.

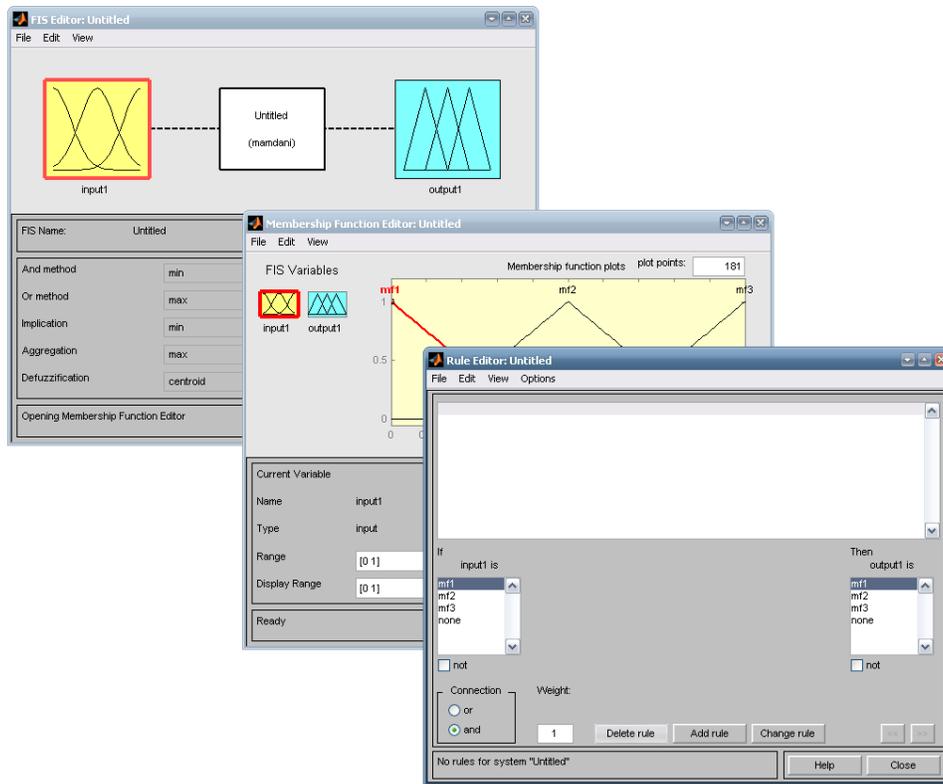


Fig. 2.11: Interfaz de usuario de la herramienta de lógica difusa de Matlab en sus diferentes funciones (la primera es la función de entrada, la segunda corresponde al editor de funciones de membresía, la tercera corresponde al editor de reglas).

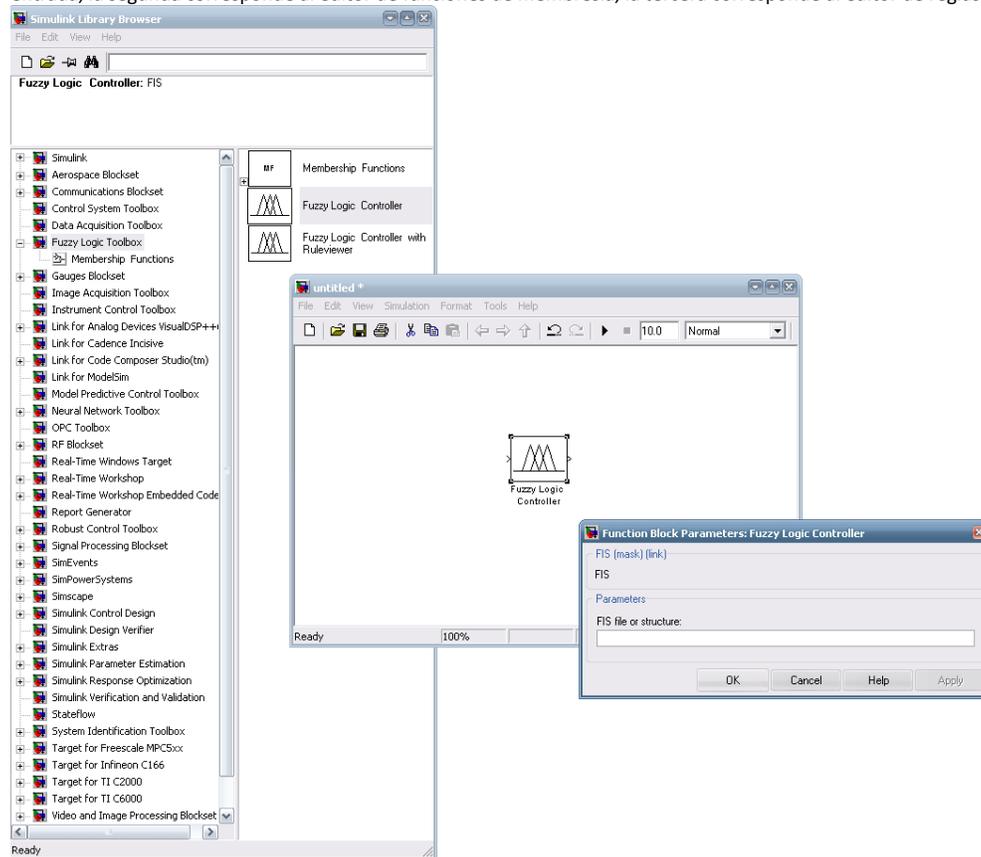


Fig. 2.12: Consolidación de la lógica difusa sobre Matlab Simulink.

Como último paso se conectan las entradas y salidas del controlador y se ejecuta el programa (para este caso solo se presenta un control difuso del sistema masa-resorte-amortiguador en la figura 2.13)

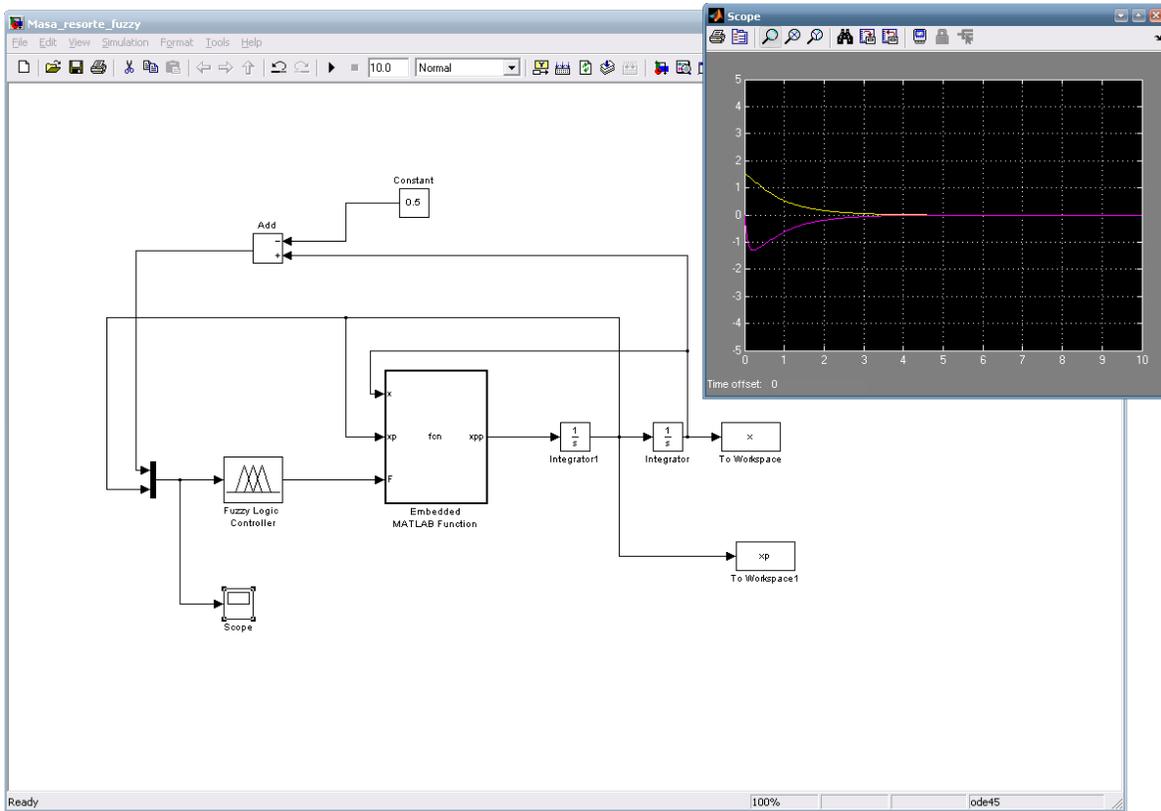


Fig. 2.13: Control de un sistema masa resorte amortiguador con matlab Simulink.

Donde se pretende llevar la posición del sistema a 0.5 (referencia) y como se puede ver en la figura 2.13, las ecuaciones de error llegan a cero que es lo que se desea en un problema de control.

2.7 Conclusiones.

En este capítulo se presenta teoría, así como las operaciones básicas de lógica difusa. Se puede observar que la lógica difusa puede tener muchas aplicaciones, como control, logística, automatización.

Capítulo 3

Modelo de un robot de dos grados de libertad.

3.1 Introducción.

Los robots manipuladores son sistemas mecánicos articulados formados por eslabones conectados entre sí a través de uniones o articulaciones. Las articulaciones son básicamente de dos tipos rotacionales y trasnacionales. Cada unión se controla independientemente a través de un actuador, que se coloca generalmente en dicha unión y el movimiento de las uniones provoca un movimiento relativo en las articulaciones (referencia pendiente).

3.2 Formulación Euler-Lagrange.

Las ecuaciones de un robot manipulador pueden obtenerse a partir de las ecuaciones de movimiento de Newton. El inconveniente que presenta este método es que el análisis se complica notablemente, cuando aumenta el número de articulaciones del robot. En esos casos, es conveniente emplear las ecuaciones de movimiento de Lagrange, esta ecuación viene del Lagrangiano que es dado por la suma de la energía potencial y cinética del sistema, la energía cinética es definida de acuerdo a la siguiente relación:

$$K = \frac{1}{2} \sum_{i=1}^n m_i v_i^2 = \frac{1}{2} \dot{q}^T D(q) \dot{q} \quad (3.1)$$

donde m_i es la i -ésima masa del i -ésimo eslabón, v_i corresponde a la velocidad del i -ésimo eslabón, $q \in R^n$ es el vector relacionado a las posiciones articulares, $\dot{q} \in R^n$

corresponde a las velocidades articulares, y $D(q)$ es denominado como matriz de inercia. La energía cinética es descrita por la siguiente ecuación:

$$U = \frac{1}{2} \sum_{i=1}^n m_i h_i \quad (3.2)$$

donde h_i es la i -ésima altura respecto al centro de gravedad, y n representa el número de grados de libertad del sistema, como se observa en el anteriormente de estas ecuaciones de desprenden las dinámicas de Euler-Lagrange, las cuales son un conjunto de ecuaciones que describen el comportamiento de un sistema, este conjunto de ecuaciones es descrito por:

$$\frac{d}{dt} \left[\frac{\partial L}{\partial \dot{q}} \right] - \frac{\partial L}{\partial q} = \tau \quad (3.3)$$

donde τ es la entrada de control o par de fuerza aplicado, L el Lagrangiano del sistema y es dado por la siguiente relación:

$$L = K - U \quad (3.4)$$

De la ecuación (5.3) se puede hacer una factorización, la cual nos lleva a la ecuación clásica de un robot manipulador:

$$D(q)\ddot{q} + C(q, \dot{q}) + G(q) = \tau \quad (3.5)$$

Cuando se presentan dinámicas dadas por fuerzas tribológicas, esta última queda expresada como:

$$D(q)\ddot{q} + C(q, \dot{q}) + G(q) + F(\dot{q}) = \tau \quad (3.6)$$

Donde:

$D(q) \in \mathbf{R}^{n \times n}$:	Matriz de inercias.
$C(q, \dot{q}) \in \mathbf{R}^{n \times n}$:	Matriz de coriolis y fuerzas centrífugas
$G(q) \in \mathbf{R}^{n \times 1}$:	Vector de gravedad
$F(\dot{q}) \in \mathbf{R}^{n \times 1}$:	Vector de fricción
$\tau \in \mathbf{R}^{n \times 1}$:	Vector de pares generalizados
$q \in \mathbf{R}^{n \times 1}$:	Vector de coordenadas generalizadas o variables articulares
$\dot{q} \in \mathbf{R}^{n \times 1}$:	Vector de velocidades articulares
$\ddot{q} \in \mathbf{R}^{n \times 1}$:	Vector de aceleraciones articulares

Entonces se puede seguir con el modelo del sistema planteado en esta tesis.

3.3 Configuración del robot de dos grados de libertad.

El sistema Péndulo doble o robot de dos grados de libertad (2GDL) es un sistema con eslabones tipo rotacionales, el movimiento del hombro (primer grado de libertad) se realiza en un plano horizontal girando alrededor de un eje perpendicular al plano xy , mientras que el codo (segundo eslabón) se encuentra colocado en un extremo del hombro y su eje de giro es lineal al eje del brazo y su movimiento se realiza en un plano paralelo al de este último.

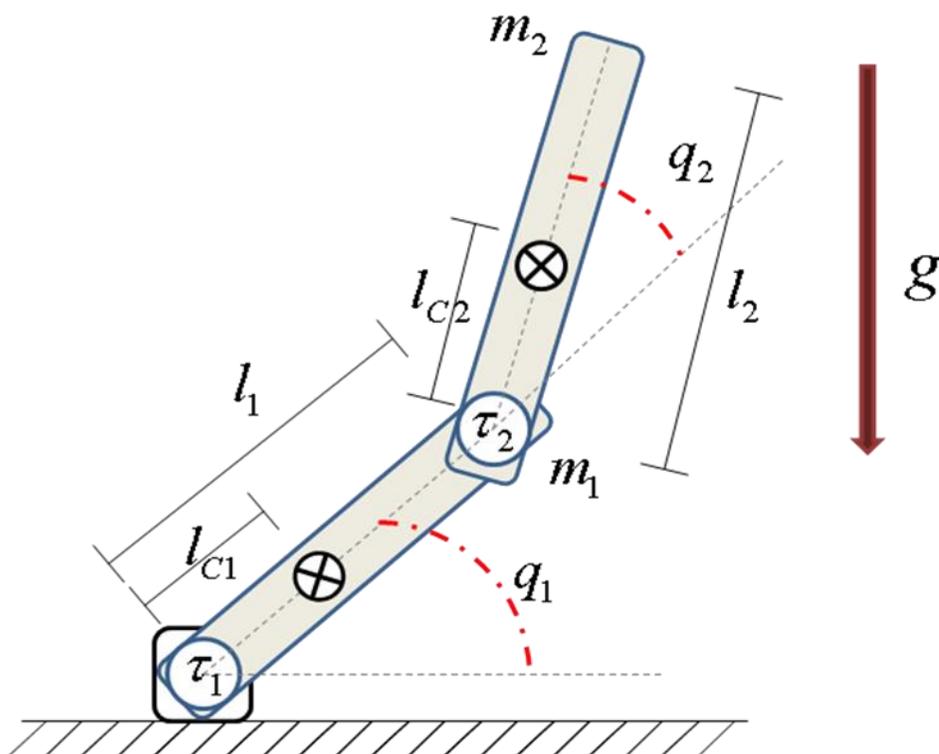


Fig. 3.1 Robot de dos grados de libertad.

El sistema Péndulo doble, este sistema se traslada de manera rotatoria el cual es descrito por la Figura 3.1 y los parámetros del sistema son descritos en la Tabla 3.1.

Descripción	Notación	Valor	Unidades
Distancia al centro de masa del primer eslabón	l_{c1}	0.1551	m
Distancia al centro de masa del segundo eslabón	l_{c2}	0.1635	m
Distancia del primer eslabón	l_1	0.2032	m
Distancia del segundo eslabón	l_2	0.3841	m
Masa del primer eslabón	m_1	0.8293	Kg
Masa del segundo eslabón	m_2	0.3402	Kg
Inercia del primer eslabón	I_1	0.005	Kgm^2
Inercia del segundo eslabón	I_2	0.0043	Kgm^2
Aceleración de gravedad.	g	9.81	m/seg^2

Tabla 3.1 Parámetros y valores del sistema Péndulo rotatorio.

Las dinámicas de este sistema vienen dadas por el siguiente conjunto de ecuaciones:

$$\ddot{q} = \begin{bmatrix} \ddot{q}_1 \\ \ddot{q}_2 \end{bmatrix} \quad \dot{q} = \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \end{bmatrix} \quad q = \begin{bmatrix} q_1 \\ q_2 \end{bmatrix} \quad (3.7)$$

El modelo matemático del sistema puede ser obtenido del diagrama de cuerpo libre mediante la segunda ley de Newton o por metodología de Euler-Lagrange, para nuestro caso de estudio se desarrolla por medio de la metodología de Euler-Lagrange. El péndulo está asociado con las siguientes variables operacionales:

- q_1 : mide el ángulo de rotación del brazo.
- q_2 : mide el ángulo de posición del péndulo respecto a la vertical.

La energía cinética asociada a este sistema es dada por la energía cinética del brazo del hombro y del codo, esta energía es dada de la siguiente forma [].

3.4 Energía cinética.

El sistema se analiza como dos masas independientes, primero se procede a calcular la energía cinética del primer eslabón. En la Figura 3.1 se muestra su diagrama de cuerpo libre.

Las coordenadas del centro de masa del primer eslabón expresadas en el plano X – Y son:

$$\begin{aligned} x_1 &= l_{c1} \cos(q_1) \\ y_1 &= l_{c1} \sin(q_1) \end{aligned} \quad (3.8)$$

Y las del segundo eslabón están dadas por:

$$\begin{aligned} x_2 &= l_1 \cos(q_1) + l_{c2} \cos(q_1 + q_2) \\ y_2 &= l_1 \sin(q_1) + l_{c2} \sin(q_1 + q_2) \end{aligned} \quad (3.9)$$

El vector de velocidad v_1 del centro de masa del primer eslabón partiendo de (3.8) en consecuencia es:

$$v_1 = \begin{bmatrix} \dot{x}_1 \\ \dot{y}_1 \end{bmatrix} = \begin{bmatrix} -l_{c1} \sin(q_1) \dot{q}_1 \\ l_{c1} \cos(q_1) \dot{q}_1 \end{bmatrix} \quad (3.10)$$

Por lo tanto la velocidad al cuadrado $v_1^T v_1$ del centro de masa de dicho eslabón se obtiene como:

$$\begin{aligned} v_1^T v_1 &= \begin{bmatrix} -l_{c1} \sin(q_1) \dot{q}_1 & l_{c1} \cos(q_1) \dot{q}_1 \end{bmatrix} \begin{bmatrix} -l_{c1} \sin(q_1) \dot{q}_1 \\ l_{c1} \cos(q_1) \dot{q}_1 \end{bmatrix} \\ v_1^T v_1 &= l_{c1}^2 \dot{q}_1^2 \left[\sin^2(q_1) + \cos^2(q_1) \right] \end{aligned} \quad (3.10)$$

Al tomar la identidad trigonométrica $\sin^2(q_1) + \cos^2(q_1) = 1$ se obtiene la velocidad al cuadrado del centro de masa, y esta resulta ser:

$$v_1^T v_1 = l_{c1}^2 \dot{q}_1^2 \quad (3.11)$$

Finalmente, la energía cinética correspondiente al movimiento del primer eslabón se obtiene como:

$$K_1 = \frac{1}{2} m_1 v_1^T v_1 + \frac{1}{2} I_1 \dot{q}_1^2 = \frac{1}{2} m_1 l_{c1}^2 \dot{q}_1^2 + \frac{1}{2} I_1 \dot{q}_1^2 \quad (3.12)$$

Por otro lado, la velocidad del centro de masa del segundo eslabón expresada en el plano X-Y está dada por:

$$v_2 = \begin{bmatrix} \dot{x}_2 \\ \dot{y}_2 \end{bmatrix} = \begin{bmatrix} -l_1 \sin(q_1) \dot{q}_1 - l_{c2} \sin(q_1 + q_2) (\dot{q}_1 + \dot{q}_2) \\ l_{c1} \cos(q_1) \dot{q}_1 + l_{c1} \cos(q_1 + q_2) (\dot{q}_1 + \dot{q}_2) \end{bmatrix} \quad (3.13)$$

Finalmente, la velocidad al cuadrado respecto al centro de masa del segundo eslabón está dada por:

$$\begin{aligned} v_2^T v_2 &= \begin{bmatrix} \dot{x}_2 & \dot{y}_2 \end{bmatrix} \begin{bmatrix} \dot{x}_2 \\ \dot{y}_2 \end{bmatrix} \\ &= \begin{bmatrix} -l_1 \sin(q_1) \dot{q}_1 - l_{c2} \sin(q_1 + q_2) (\dot{q}_1 + \dot{q}_2) \\ l_{c1} \cos(q_1) \dot{q}_1 + l_{c1} \cos(q_1 + q_2) (\dot{q}_1 + \dot{q}_2) \end{bmatrix}^T \begin{bmatrix} -l_1 \sin(q_1) \dot{q}_1 - l_{c2} \sin(q_1 + q_2) (\dot{q}_1 + \dot{q}_2) \\ l_{c1} \cos(q_1) \dot{q}_1 + l_{c1} \cos(q_1 + q_2) (\dot{q}_1 + \dot{q}_2) \end{bmatrix} \end{aligned} \quad (3.14)$$

donde al emplear las identidades trigonométricas

$$\begin{aligned}\cos^2(A) + \sin^2(A) &= 1, \\ \sin(A)\sin(A+B) + \cos(A)\cos(A+B) &= \cos(B)\end{aligned}\quad (3.15)$$

La operación anterior resulta en lo siguiente:

$$v_2^T v_2 = l_1^2 \dot{q}_1^2 + l_{c1} [\dot{q}_1^2 + 2\dot{q}_1 \dot{q}_2 + \dot{q}_2^2] + 2l_1 l_{c1} [\dot{q}_1^2 + \dot{q}_1 \dot{q}_2] \cos(q_2) \quad (3.16)$$

De esta forma la energía cinética de dicho eslabón es:

$$\begin{aligned}K_2 &= \frac{1}{2} m_2 v_2^T v_2 + \frac{1}{2} I_2 [\dot{q}_1^2 + \dot{q}_2^2], \\ &= \frac{m_2 l_1^2}{2} \dot{q}_1^2 + \frac{m_2 l_{c1}}{2} [\dot{q}_1^2 + 2\dot{q}_1 \dot{q}_2 + \dot{q}_2^2] + m_2 l_1 l_{c1} [\dot{q}_1^2 + \dot{q}_1 \dot{q}_2] \cos(q_2) + \frac{1}{2} I_2 [\dot{q}_1^2 + \dot{q}_2^2].\end{aligned}\quad (3.17)$$

De esta manera se puede obtener la energía potencial del sistema, que en resumidas cuentas es la suma de K_1 y K_2 , obteniendo lo siguiente:

$$\begin{aligned}K &= K_1 + K_2 = \frac{1}{2} m_1 l_{c1}^2 \dot{q}_1^2 + \frac{1}{2} I_1 \dot{q}_1^2 + \frac{m_2 l_1^2}{2} \dot{q}_1^2 \\ &+ \frac{m_2 l_{c1}}{2} [\dot{q}_1^2 + 2\dot{q}_1 \dot{q}_2 + \dot{q}_2^2] + m_2 l_1 l_{c1} [\dot{q}_1^2 + \dot{q}_1 \dot{q}_2] \cos(q_2) + \frac{1}{2} I_2 [\dot{q}_1^2 + \dot{q}_2^2].\end{aligned}\quad (3.18)$$

Como el siguiente paso para obtener el Lagrangiano del robot 2GDL, es obtener la energía potencial del mismo, la cual se describe en la siguiente sección.

3.5 Energía potencial.

De forma similar, la energía potencial puede descomponerse como la suma de dos partes: $U = U_1 + U_2$ donde U_1 y U_2 son las energías potenciales asociadas a las masas m_1 y m_2 respectivamente. De donde se obtiene que:

$$U_1 = m_1 l_{c1} g \sin(q_1) \quad (3.19)$$

Y

$$U_2 = m_2 l_1 g \sin(q_1) + m_2 l_{c2} g \sin(q_1 + q_2) \quad (3.20)$$

Por lo tanto, la energía potencial total del sistema es la suma de ambas resultando en lo siguiente:

$$U_1 + U_2 = m_1 l_{c1} g \sin(q_1) + m_2 l_1 g \sin(q_1) + m_2 l_{c2} g \sin(q_1 + q_2) \quad (3.21)$$

3.6 Lagrangiano.

Así el Lagrangiano dado por (3.4), resulta en la resta de la energía cinética con la energía potencial, entonces para el caso de estudio el Lagrangiano se obtiene con (3.21) restado de (3.18) de la siguiente forma:

$$L = \frac{1}{2} m_1 l_{c1}^2 \dot{q}_1^2 + \frac{1}{2} I_1 \dot{q}_1^2 + \frac{m_2 l_1^2}{2} \dot{q}_1^2 + \frac{m_2 l_{c1} l_1}{2} [\dot{q}_1^2 + 2\dot{q}_1 \dot{q}_2 + \dot{q}_2^2] + m_2 l_1 l_{c1} [\dot{q}_1^2 + \dot{q}_1 \dot{q}_2] \cos(q_2) + \frac{1}{2} I_2 [\dot{q}_1^2 + \dot{q}_2^2] - m_1 l_{c1} g \sin(q_1) - m_2 l_1 g \sin(q_1) - m_1 l_{c2} g \sin(q_1 + q_2). \quad (3.22)$$

de esta última ecuación se puede obtener el modelo dinámico del sistema, entonces como último paso se procede a obtener las ecuaciones de movimiento de Euler-Lagrange.

3.7 Ecuaciones de movimiento de Euler-Lagrange.

Al utilizar las ecuaciones de movimiento de Euler-Lagrange (3.3), y desarrollarlas se tiene:

$$\frac{\partial L}{\partial \dot{q}_1} = [m_1 l_{c1}^2 + m_2 l_1^2] \dot{q}_1 + m_2 l_{c2}^2 \dot{q}_1 + m_2 l_{c2}^2 \dot{q}_2 + 2m_2 l_1 l_{c2} \cos(q_2) \dot{q}_1 + m_2 l_1 l_{c2} \cos(q_2) \dot{q}_2 + I_1 \dot{q}_1 + I_2 [\dot{q}_1 + \dot{q}_2], \quad (3.23)$$

$$\frac{\partial L}{\partial q_1} = -g [m_1 l_{c1} + m_2 l_1] \cos(q_1) - g m_2 l_{c2} \cos(q_1 + q_2) \quad (3.24)$$

$$\frac{\partial L}{\partial \dot{q}_2} = m_2 l_{c2}^2 \dot{q}_1 + m_2 l_{c2}^2 \dot{q}_2 + m_2 l_1 l_{c2} \cos(q_2) \dot{q}_1 + I [\dot{q}_1 + \dot{q}_2] \quad (3.25)$$

$$\frac{\partial L}{\partial q_2} = m_2 l_1 l_{c1} [\dot{q}_1^2 + \dot{q}_1 \dot{q}_2] \sin(q_2) + m_1 l_{c2} g \cos(q_1 + q_2) \quad (3.26)$$

$$\frac{d}{dt} \left[\frac{\partial L}{\partial \dot{q}_1} \right] = [m_1 l_{c1}^2 + m_2 l_1^2 + m_2 l_{c2}^2 + 2m_2 l_1 l_{c2} \cos(q_2)] \ddot{q}_1 + [m_2 l_{c2}^2 + m_2 l_1 l_{c2} \cos(q_2)] \ddot{q}_2 \quad (3.27)$$

$$-2m_2 l_1 l_{c2} \sin(q_2) \dot{q}_1 \dot{q}_2 - m_2 l_1 l_{c2} \sin(q_2) \dot{q}_2^2 + I_1 \ddot{q}_1 + I_2 [\ddot{q}_1 + \ddot{q}_2],$$

$$\frac{\partial L}{\partial \dot{q}_2} = [m_2 l_{c2}^2 + m_2 l_1 l_{c2} \cos(q_2)] \ddot{q}_1 + m_2 l_{c2}^2 \ddot{q}_2 - m_2 l_1 l_{c2} \sin(q_2) \dot{q}_1 \dot{q}_2 + I [\ddot{q}_1 + \ddot{q}_2] \quad (3.28)$$

Debido a que el sistema puede ser representado como (3.5), de las ecuaciones (3.24), (3.26), (3.27) y (3.28), el modelo dinámico se simplifica como:

$$D(q) = \begin{bmatrix} \theta_1 + \theta_2 + \theta_3 \cos(q_2) & \theta_2 + \theta_3 \cos(q_2) \\ \theta_2 + \theta_3 \cos(q_2) & \theta_2 \end{bmatrix} \quad (3.29)$$

$$C(q, \dot{q}) = \theta_3 \text{sen}(q_2) \begin{bmatrix} -\dot{q}_2 & -\dot{q}_1 - \dot{q}_2 \\ -\dot{q}_1 & 0 \end{bmatrix} \quad (3.30)$$

$$G(q) = \begin{bmatrix} g\theta_4 \cos(q_1) + g\theta_5 \cos(q_1 + q_2) \\ g\theta_5 \cos(q_1 + q_2) \end{bmatrix} \quad (3.31)$$

$$F(\dot{q}) = \begin{bmatrix} v_1 \dot{q} + c_1 \text{sgn } \dot{q} \\ v_2 \dot{\theta} + c_2 \text{sgn } \dot{\theta} \end{bmatrix} \quad (3.32)$$

donde los siguientes parámetros fueron introducidos para reducir las ecuaciones:

$$\begin{cases} \theta_1 = m_1 l_{c1}^2 + m_2 l_1^2 + I_1 \\ \theta_2 = m_2 l_{c2}^2 + I_2 \\ \theta_3 = m_2 l_1 l_{c2} \\ \theta_4 = m_1 l_1 l_{c2} \\ \theta_5 = m_2 l_{c2} \end{cases}$$

donde las constantes v y c son coeficientes de fricción seca y viscosa respectivamente. De esta manera se obtiene el modelo matemático, para el robot en lazo abierto, el cual es dado por:

$$\ddot{q} = -D^{-1}(q)[C(q, \dot{q}) + G(q) + F(\dot{q})] \quad (3.33)$$

Ya que el sistema es de dos grados de libertad y de segundo orden se hace el siguiente cambio de variable:

$$\begin{aligned} q_1 &= x_1, & \dot{q}_1 &= \dot{x}_1 = x_3, & \ddot{q}_1 &= \dot{x}_3, \\ q_2 &= x_2, & \dot{q}_2 &= \dot{x}_2 = x_4, & \ddot{q}_2 &= \dot{x}_4 \end{aligned} \quad (3.34)$$

Así entonces el sistema queda representado de la siguiente manera:

$$\frac{dx}{dt} = f(x) \quad (3.35)$$

donde $\frac{dx}{dt} = f(x)$ tiene las siguientes dinámicas:

$$\frac{dx}{dt} = \frac{\begin{bmatrix} \theta_2 & -\theta_2 - \theta_3 \cos(q_2) \\ -\theta_2 - \theta_3 \cos(q_2) & \theta_1 + \theta_2 + \theta_3 \cos(q_2) \end{bmatrix}}{\theta_1 \theta_2 + \theta_3^2 \cos^2(q_2)} \left\{ \begin{array}{l} \left[\begin{array}{l} \tau_1 - F_1 \\ \tau_2 - F_1 \end{array} \right] + \theta_3 \text{sen}(q_2) \left[\begin{array}{l} \dot{q}_2 \quad \dot{q}_1 + \dot{q}_2 \\ \dot{q}_1 \quad 0 \end{array} \right] \\ - \left[\begin{array}{l} g\theta_4 \cos(q_1) + g\theta_5 \cos(q_1 + q_2) \\ g\theta_5 \cos(q_1 + q_2) \end{array} \right] \end{array} \right\} \quad (3.36)$$

donde F_1 y F_2 son los parámetros de fricción.

3.8 Modelo cinemático directo e inverso de posición, y velocidad.

Como se puede ver en la sección anterior, el modelo matemático solo es referido a posiciones articulares, las cuales son indispensables para el punto de vista de regulación y seguimiento de trayectorias, sin embargo cuando se planifica un proceso dentro del espacio de trabajo es necesario del conocimiento de las posiciones de las coordenadas generalizadas, por tal motivo se requiere del conocimiento del modelo cinemático de posición.

De acuerdo a lo anterior, el modelo cinemático directo de posición, es la posición del efector final del robot 2GDL determinado por la figura 3.1, por lo tanto este modelo también puede ser expresado de la forma siguiente:

$$\begin{aligned} x &= l_1 \cos(q_1) + l_2 \cos(q_1 + q_2) \\ y &= l_1 \text{sen}(q_1) + l_2 \text{sen}(q_1 + q_2) \end{aligned} \quad (3.37)$$

Con base al marco de referencia cartesiano, de la figura 3.1, se obtienen los ángulos q_2 , ϕ , y β , mismos que son empleados para definir a q_1 .

$$q_2 = \cos^{-1} \left\{ \frac{(x^2 + y^2) - (l_1^2 + l_2^2)}{2l_1 l_2} \right\} \quad (3.38)$$

$$\phi = \text{sen}^{-1} \left\{ \frac{l_2 \text{sen}(q_2)}{\sqrt{x^2 + y^2}} \right\} \quad (3.39)$$

$$\beta = \theta_1 + \phi \quad (3.40)$$

Por lo tanto, el cálculo de q_1 es al considerar las dos posibilidades de q_2 , de la siguiente manera:

1. Si $q_2 > 0$ entonces $q_2 = \beta - \phi$.
2. Si $q_2 < 0$ entonces $q_2 = \beta + \phi$.

El modelo cinemático directo de velocidad se obtiene directamente del modelo cinemático directo de posición, por lo tanto hay que derivar (3.37) y se obtiene lo siguiente:

$$\begin{aligned}\dot{x} &= -(l_1 \sin(q_1) + l_2 \sin(q_1 + q_2)) \dot{q}_1 - (l_2 \sin(q_1 + q_2)) \dot{q}_2 \\ \dot{y} &= (l_1 \cos(q_1) + l_2 \cos(q_1 + q_2)) \dot{q}_1 + (l_2 \cos(q_1 + q_2)) \dot{q}_2\end{aligned}\quad (3.41)$$

O de forma matricial se obtiene lo siguiente:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \end{bmatrix} = \underbrace{\begin{bmatrix} -(l_1 \sin(q_1) + l_2 \sin(q_1 + q_2)) & -(l_2 \sin(q_1 + q_2)) \\ (l_1 \cos(q_1) + l_2 \cos(q_1 + q_2)) & (l_2 \cos(q_1 + q_2)) \end{bmatrix}}_{J_x} \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \end{bmatrix}\quad (3.42)$$

donde J_x es el Jacobiano del robot. De (3.42) se obtiene el modelo cinemático inverso de velocidad, el cual es determinado por la siguiente ecuación:

$$\dot{q} = J_x^{-1} \dot{x}\quad (3.43)$$

donde el Jacobiano inverso para el robot 2GDL descrito en la figura 3.1 es lo siguiente:

$$J_x^{-1} = \frac{1}{\Delta(q)} \begin{bmatrix} (l_2 \cos(q_1 + q_2)) & (l_2 \sin(q_1 + q_2)) \\ -(l_1 \cos(q_1) + l_2 \cos(q_1 + q_2)) & -(l_1 \sin(q_1) + l_2 \sin(q_1 + q_2)) \end{bmatrix}\quad (3.44)$$

donde $\Delta(q)$ es el determinante del Jacobiano, tal expresión respecto al robot 2GDL es la siguiente:

$$\Delta(q) = l_1 l_2 (\sin(q_1) \sin(q_1 + q_2) + \cos(q_1) \cos(q_1 + q_2))\quad (3.45)$$

De esta forma se obtienen los modelos cinemáticos tanto de posición y velocidad (directos e inversos), y el modelo dinámico del sistema (3.36), y como último se requiere evaluar dichos modelos, este proceso se describe en la siguiente sección.

3.9 Evaluación del modelo dinámico y cinemático.

Se ha sustituido el vector de fuerzas de fricción para cada uno de los modelos de fricción como se ha visto en la sección anterior y se hace un análisis de su comportamiento de acuerdo a su posición, velocidad de cada articulación.

3.9.1 Evaluación del sistema sin fricción.

Las condiciones iniciales para las simulaciones del modelo evaluado en el robot de dos grados de libertad se muestran a continuación (caso sin fricción):

Tiempo de simulación: 20 segundos.

Paso de integración: 0.1 milisegundos.

Tipo de integrador: ODE 45

Condiciones iniciales:

$$\begin{aligned}
 q_1(0) &= 0, \\
 \dot{q}_1(0) &= 0, \\
 q_2(0) &= \pi/2, \\
 \dot{q}_2(0) &= 0.
 \end{aligned}$$

De las condiciones anteriores se obtuvieron los siguientes resultados.

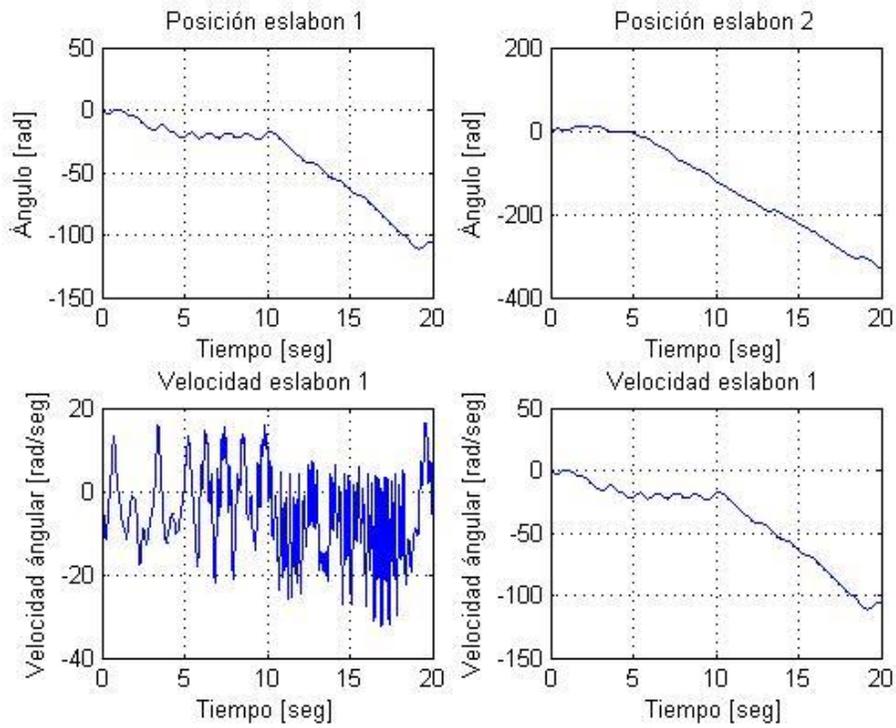


Fig. 3.2: Movimientos y velocidades de las articulaciones del robot (sin fricción).

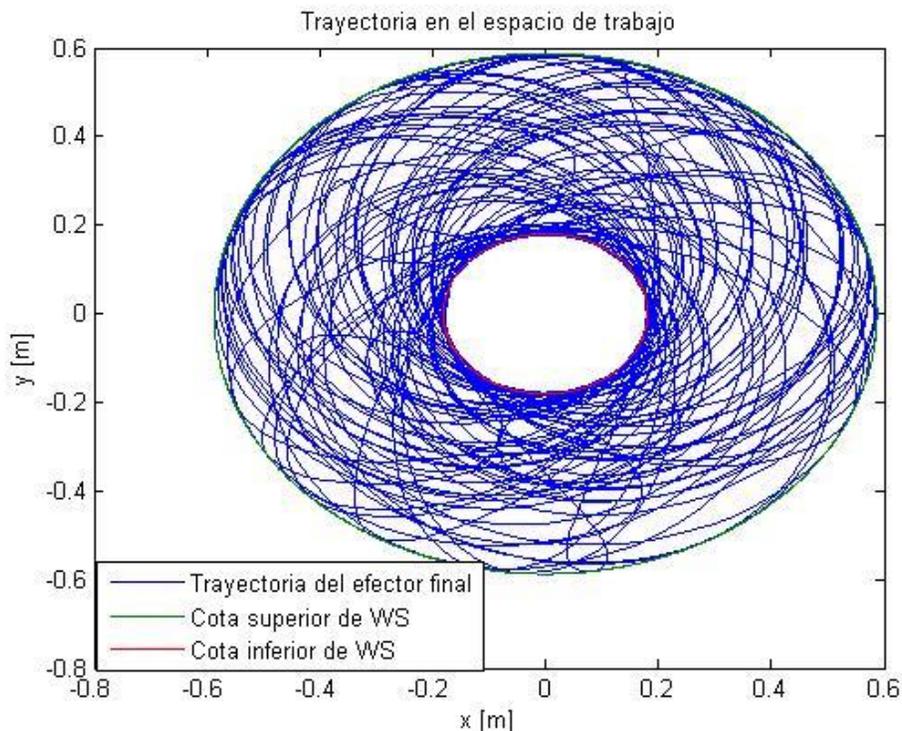


Fig. 3.3: Movimientos en el espacio de trabajo (WS) del robot (sin fricción).

Como se puede ver en la figura 3.3 el sistema oscila en función de las dinámicas, sin embargo como es el caso en que el sistema no presenta fricción, en la simulación numérica el sistema seguirá oscilando de forma indeterminada.

En la figura 3.4 se observa el comportamiento del sistema, en esta figura se muestra el desempeño del mismo pero sobre el espacio de trabajo, es decir, se muestra la evolución del efector final del sistema descrito en la figura 3.1 en las coordenadas cartesianas X-Y.

3.9.2 Evaluación del sistema con poca fricción.

Ahora bien, si el sistema cuenta con fricción (esto para que tenga un comportamiento más aproximado a la realidad), ya que los sistemas dinámicos reales del tipo mecánicos cuentan con parámetros y dinámicas que en las que se ejercen fuerzas de fricción, al aplicar estas fuerzas de fricción y con las condiciones de simulación siguientes:

Tiempo de simulación: 20 segundos.
 Tipo de integrador: ODE 45
 Fricción viscosa: $c_1=0.0023$, $c_2=0.0025$.
 Fricción seca: $v_1=0.00545$, $v_2=0.00047$.

Paso de integración: 0.1 milisegundos.
 Condiciones iniciales:
 $q_1(0) = 0$,
 $\dot{q}_1(0) = 0$,
 $q_2(0) = \pi / 2$,
 $\dot{q}_2(0) = 0$.

De las condiciones anteriores se obtuvieron los siguientes resultados.

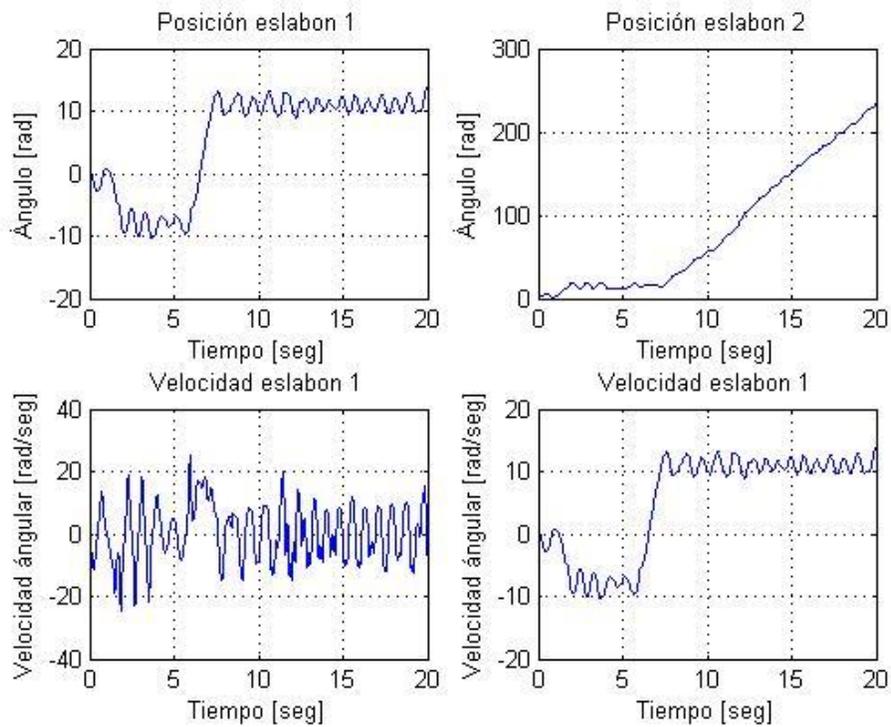


Fig. 3.4: Movimientos y velocidades de las articulaciones del robot (poca fricción).

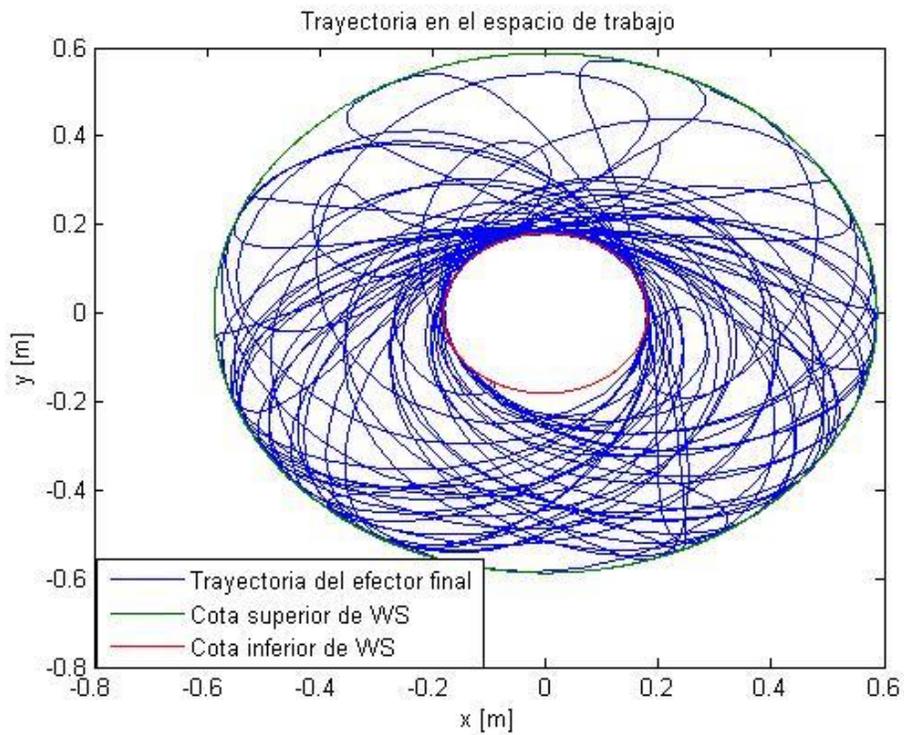


Fig. 3.5: Movimientos en el espacio de trabajo (WS) del robot (poca fricción).

Como se puede observar en la figura 3.4 en comparación con la figura 3.2, el sistema se comporta de forma diferente, esto por el simple hecho, que en esta simulación numérica se incorporan parámetros de fricción viscosa y seca, de igual manera se puede ver dicho efecto al comparar las figuras 3.5 y 3.3 que son el caso en que se muestra el desempeño del robot 2GDL en su espacio de trabajo.

Como en los dos casos anteriores no se puede visualizar de una forma fácil, como las fuerzas de fricción seca y viscosa influyen directamente sobre el comportamiento de un sistema dinámico (como lo es el sistema mecánico robot 2GDL), se propone realizar una simulación numérica más, en la cual estos parámetros de fricción sean tan grandes que se pueda observar la diferencia al comparar resultados.

3.9.3 Evaluación del sistema con mucha fricción.

Ya que un sistema real, las consideraciones de diseño son realizadas de tal manera que la fricción sea mínima, entonces en la siguiente simulación se hace esta consideración.

Tiempo de simulación: 20 segundos.
Tipo de integrador: ODE 45
Fricción viscosa: $c_1=0.23$, $c_2=0.25$.
Fricción seca: $v_1=0.545$, $v_2=0.047$.

Paso de integración: 0.1 milisegundos.

Condiciones iniciales:

$$q_1(0) = 0,$$

$$\dot{q}_1(0) = 0,$$

$$q_2(0) = \pi / 2,$$

$$\dot{q}_2(0) = 0.$$

De las condiciones anteriores se obtuvieron los siguientes resultados.

Como se puede ver en las figuras 3.2, 3.4 y 3.6 el comportamiento más aceptable es el descrito por la figura 3.6 ya que dicho comportamiento es el más aproximado a una planta real, ya que como lo muestra la figura 3.4 el sistema permanece en movimiento, dicho comportamiento es imposible en la vida real. El comportamiento de la figura 3.6 es bueno, sin embargo en la realidad se trata de tener menor fricción ya que esta genera pérdidas, como lo es el desgaste en las piezas.

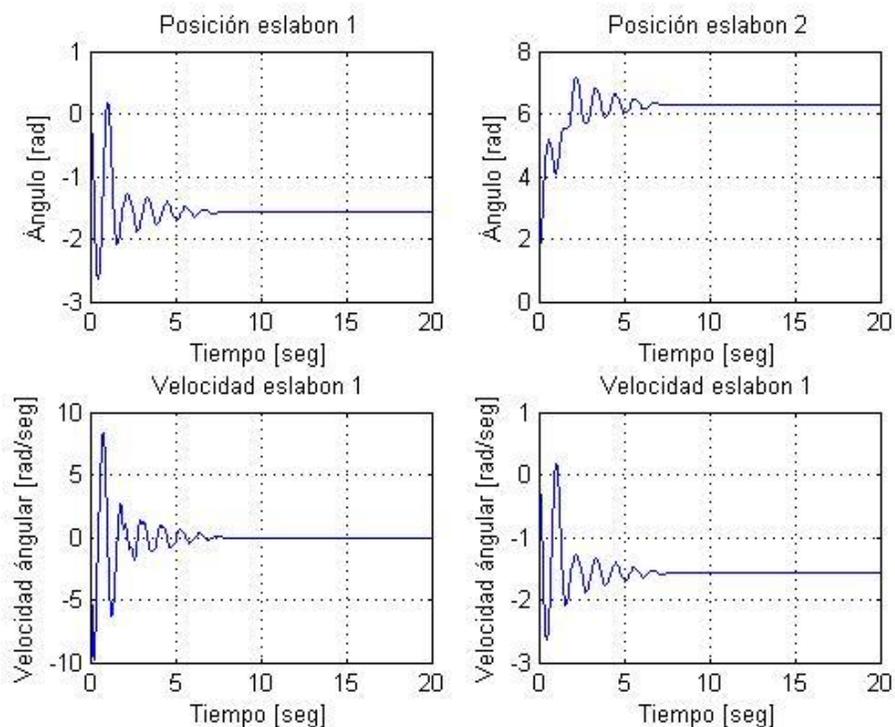


Fig. 3.6: Movimientos y velocidades de las articulaciones del robot (mucho fricción).

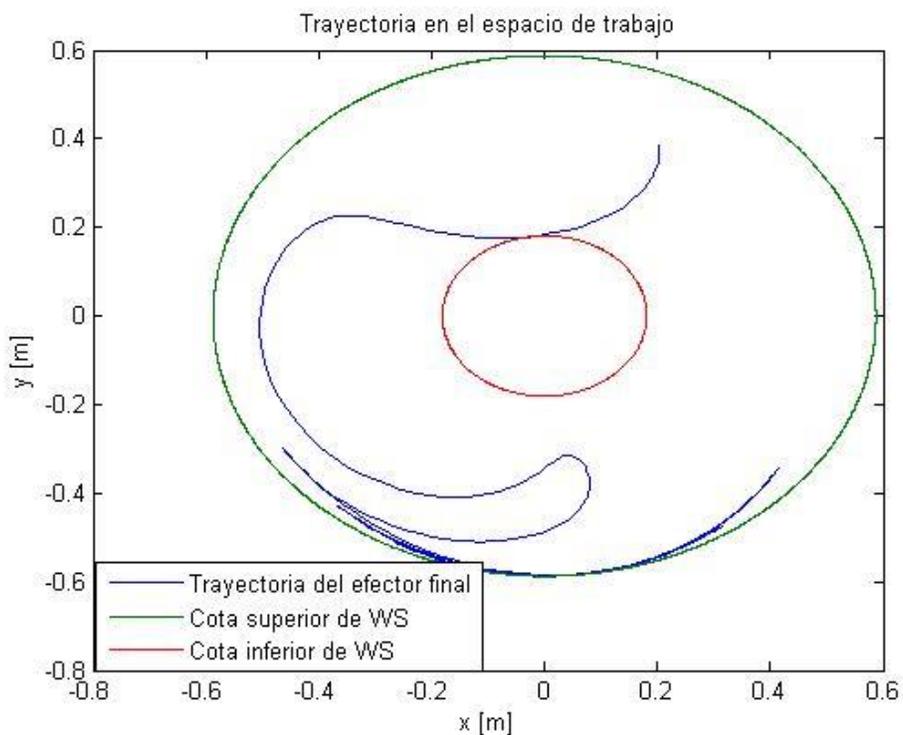


Fig. 3.7: Movimientos en el espacio de trabajo (WS) del robot (mucho fricción).

3.10 Conclusiones.

En este capítulo se puede observar como un sistema dinámico de dos grados de libertad (robot 2GDL) puede tener diferentes comportamientos, esto dependiendo de los parámetros inherentes del sistema (como lo son las masas, las longitudes de los eslabones, las inercias y los coeficientes de fricción). También se puede observar que no es necesario tener un sistema de forma física para poder determinar su comportamiento, ya que con el uso del modelo dinámico y cinemático se puede obtener dicho comportamiento, sin embargo hay que tener un amplio conocimiento de programación así como de métodos numéricos para la solución del sistema de ecuaciones diferenciales no-lineales.

Capítulo 4

Control clásico de un robot.

4.1 Introducción.

Clásicamente el control de sistemas robóticos es mediante la aplicación del también clásico control PD y PID, sin embargo estos controladores no compensan de forma eficiente las no linealidades, ya que los robots manipuladores en su totalidad cuentan con dinámicas no lineales (esto se puede ver en el capítulo anterior), por lo tanto la generación de un controlador que compense dichas no linealidades es necesario. Una de las aproximaciones clásicas de un controlador el cual compensa de forma eficiente es el controlador PD+ el cual es un simple controlador que compensa términos de energía potencial. La existencia y prueba de estos controladores es muy compleja, ya que para poder demostrar el funcionamiento de estos controladores es necesaria de una prueba de estabilidad, clásicamente esta prueba es basada en la teoría de estabilidad en el sentido de Lyapunov en el cual se involucran conceptos de funciones definidas positivas, definidas negativas, funciones semi-definidas positivas, funciones semi-definidas negativas, matrices simétricas, entre otras definiciones. Lo cual hace que dichas pruebas sean difíciles por tal motivo este tema de investigación solo se enfoca a la comparación de estas técnicas con un controlador propuesto, y por tal motivo solo se presentaran las simulaciones numéricas de los controladores.

4.2 Regulación del robot 2GDL.

Clásicamente el controlador PD es aplicado a sistemas lineales en si, sin embargo estudios recientes han proporcionado información que involucra el uso de este controlador sobre plantas no lineales.

La configuración básica del controlador PD es basada en funciones de error, por lo tanto para la aplicación de este controlador sobre la dinámica del sistema se requiere de una dinámica de error la cual es dada del las dinámicas a controlar del sistema.

4.2.1 Controlador P y PD.

Para el control de posición de un robot manipulador es necesario el conocimiento de las posiciones articulares o la posición angular de las coordenadas generalizadas, en otras palabras el conocimiento de las variables q , por lo tanto en error de posición será generado por la siguiente ecuación:

$$\tilde{q} = q - q_d \quad (4.1)$$

donde $\tilde{q} \in \mathbb{R}^{n \times 1}$ es el error, $q \in \mathbb{R}^{n \times 1}$ es la posición real del sistema y $q_d \in \mathbb{R}^{n \times 1}$ es la posición deseada. Por lo tanto la construcción de un controlador con esta simple estructura seria de la forma:

$$\tau = K_p \tilde{q} \quad (4.2)$$

donde $\tau \in \mathbb{R}^{n \times 1}$ es la entrada de control de la ecuación (3.5), $K_p \in \mathbb{R}^{n \times n}$ es una matriz de ganancias proporcionales, y a esta estructura de control se le conoce como controlador tipo proporcional o controlador P.

Al aplicar la estructura de control (4.2) sobre el sistema robótico de dos grados de libertad dado en la figura 3.1 con los parámetros de la tabla 3.1 se presentan los siguientes resultados (para todas las simulaciones numéricas se toma el caso de poca fricción visto en el capítulo 3, sección 5.2):

Tiempo de simulación: 20 segundos.

Paso de integración: 0.1 milisegundos.

Tipo de integrador: ODE 45

Condiciones iniciales:

Posición deseada:

$$q_1(0) = 0,$$

$$q_{1d} = \pi / 2,$$

$$\dot{q}_1(0) = 0,$$

$$q_{2d} = 0,$$

$$q_2(0) = \pi / 2,$$

$$\dot{q}_2(0) = 0.$$

Las salidas del sistema se comportan como lo muestra la figura 4.1, y las entradas de control se muestran en la figura 4.2, en estas figuras se puede observar como el controlador proporcional no compensa las dinámicas de forma eficiente, ya que se genera un error demasiado grande (ver figura 4.1) lo cual es de esperarse, ya que este tipo de controladores en ineficiente para el controlador de sistemas no lineales como lo es el robot de dos grados de libertad.

Para hacer un mejor control sobre el sistema se propone realizar un controlador clásico PD (proporcional derivativo) el cual tiene ciertas ventajas sobre el controlador tipo proporcional, ya que este además de tener la ventaja de ser un controlador proporcional este tiene la propiedad de tener involucrada un error de velocidad, es decir un sistema que se anticipa al cambio de error.

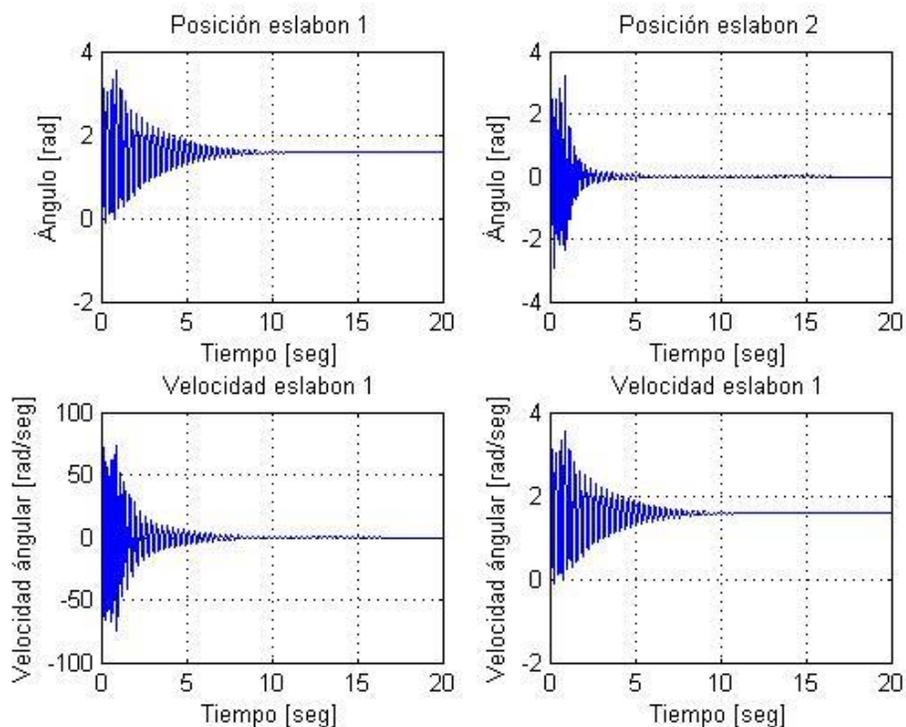


Fig. 4.1: Movimientos y velocidades de las articulaciones del robot con un controlador tipo P.

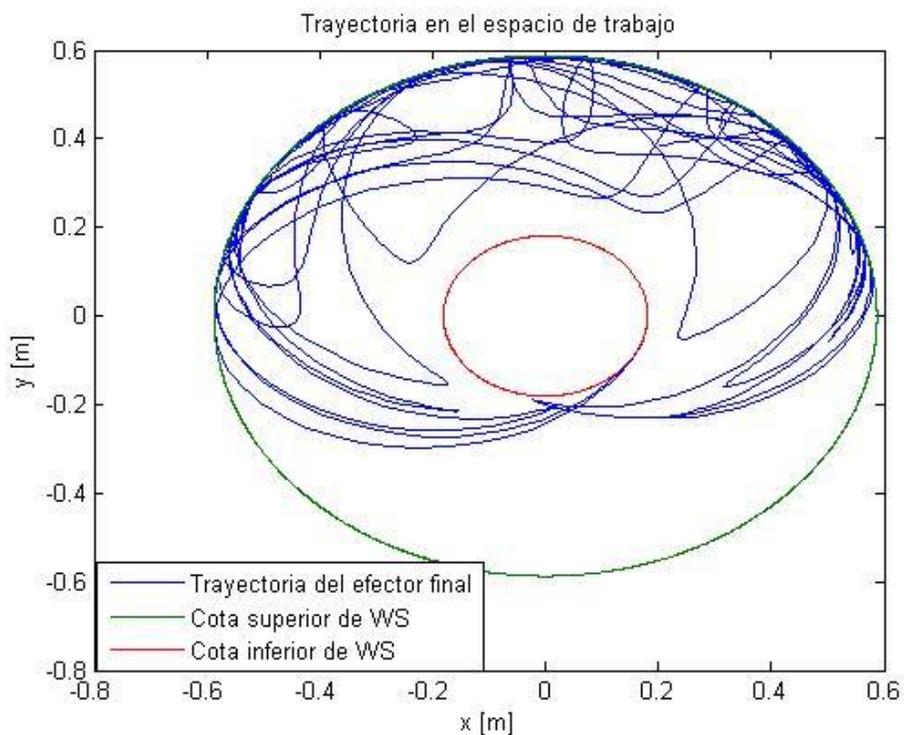


Fig. 4.2: Movimientos en el espacio de trabajo del robot con un controlador tipo P.

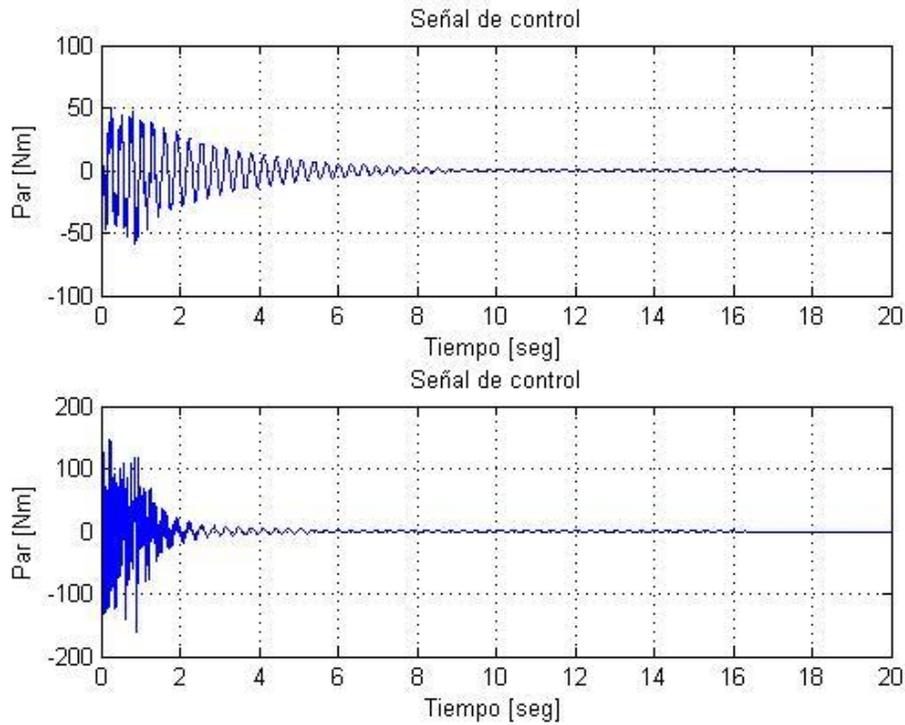


Fig. 4.3: Entradas de control o par generado por el controlador Proporcional.

En el desarrollo del controlador PD, como se menciono anteriormente es necesario del conocimiento de los estados que involucran la velocidad articular del sistema, en otras palabras del error de velocidad o derivada del error de la forma:

$$\dot{\tilde{q}} = \dot{q} - \dot{q}_d \quad (4.3)$$

donde $\tilde{q} \in \mathbb{R}^{n \times 1}$ es el error de velocidad articular, $\dot{q} \in \mathbb{R}^{n \times 1}$ es la velocidad real producida por sistema, $\dot{q}_d \in \mathbb{R}^{n \times 1}$ es la velocidad deseada. Entonces al introducir estas dinámicas de error sobre el control P, este puede reescribirse de la siguiente forma:

$$\tau = K_D \dot{\tilde{q}} + K_P \tilde{q} \quad (4.4)$$

donde $\tau \in \mathbb{R}^{n \times 1}$ es la entrada de control, $K_D \in \mathbb{R}^{n \times n}$ y $K_P \in \mathbb{R}^{n \times n}$ son matrices de ganancias las cuales tienen la propiedad de ser definidas positivas y simétricas.

Al aplicar la estructura de control (4.4) sobre el sistema robótico de dos grados de libertad dado en la figura 3.1 con los parámetros de la tabla 3.1 se presentan los siguientes resultados de simulación numérica:

Tiempo de simulación: 20 segundos.
 Tipo de integrador: ODE 45
 Posición deseada:

Paso de integración: 0.1 milisegundos.
 Condiciones iniciales:
 $q_1(0) = 0,$
 $\dot{q}_1(0) = 0,$
 $q_2(0) = \pi / 2,$
 $\dot{q}_2(0) = 0.$

$$q_{1d} = \pi/2,$$

$$q_{2d} = 0,$$

$$\dot{q}_{1d} = 0,$$

$$\dot{q}_{2d} = 0,$$

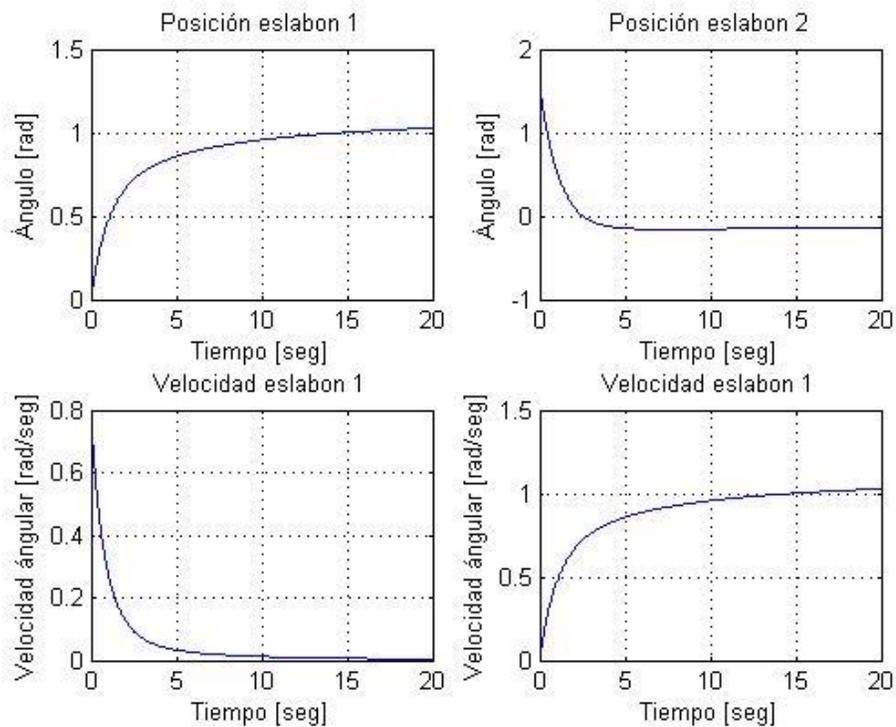


Fig. 4.4: Movimientos y velocidades de las articulaciones del robot con un controlador tipo PD.

En las figuras 4.4, 4.5 y 4.6 se puede ver el comportamiento del sistema ante la respuesta del controlador PD descrito por (4.6), sin embargo como se puede ver este control no es el indicado para resolver el problema de regulación, ya que se presenta un error en estado estacionario.

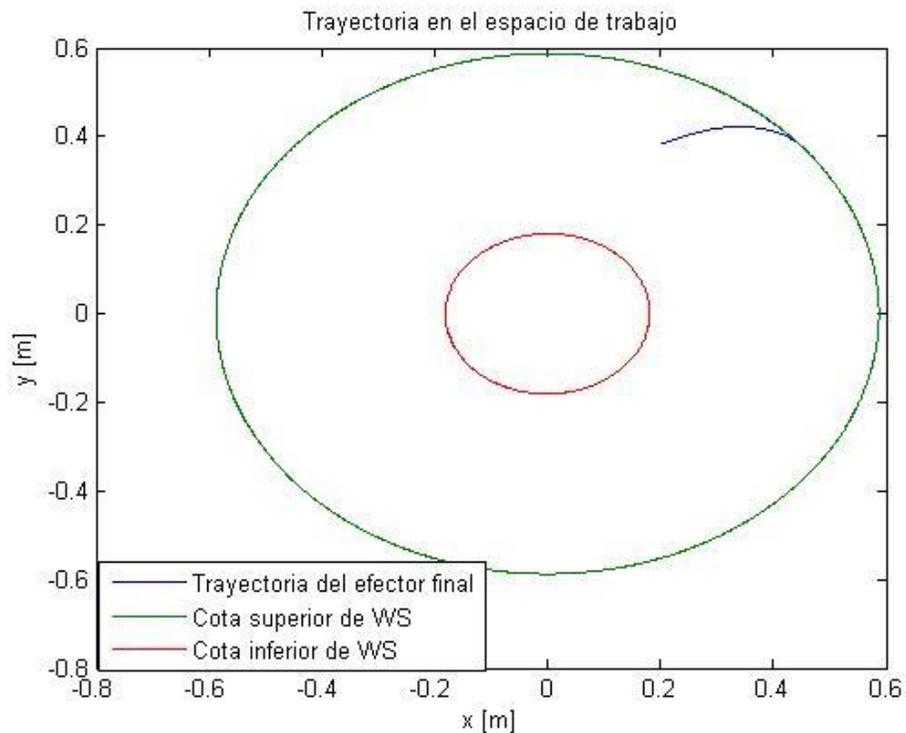


Fig. 4.5: Movimientos en el espacio de trabajo del robot con un controlador tipo PD.

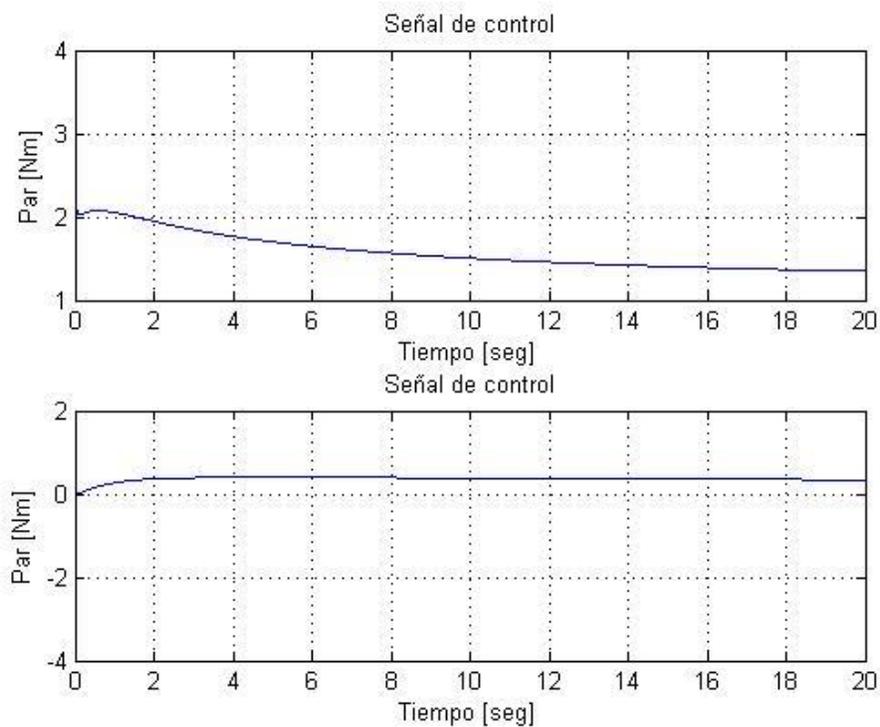


Fig. 4.6: Entradas de control o par generado por el controlador PD.

Por tal motivo se requiere de un controlador que resuelva este problema, y el controlador clásico más indicado es el controlador PID.

4.2.2 Controlador PID.

Como se menciona en la sección anterior, el controlador indicado para resolver el problema de regulación de un robot manipulador es el controlador PID ya que por su estructura este control reduce el error en estado estacionario. Esto se debe al simple hecho de que cuando un sistema mantiene un error en estado estacionario (este problema puede ser visto en la figura 4.3) siempre se mantiene un error constante, ya que la propiedad de la acción integral es la de dar un orden mayor, i.e. cuando se integra una constante en relación al tiempo de ejecución este se transforma en la ecuación de una recta con pendiente proporcional a la constante:

$$\int a dt = a \int dt = at + c \quad (4.5)$$

Si este simple hecho es aplicado a la dinámica de error, entonces cuando se encuentre un error en estado estacionario, el control aplicara una fuerza o par de tal forma que elimine el mismo.

Para aplicar esta acción entonces se propone una acción tipo integral sobre las dinámicas de error, es decir:

$$\int \tilde{q} dt = \int (q - q_d) dt \quad (4.5)$$

donde $\tilde{q} \in \mathbb{R}^{n \times 1}$ es el error de posición articular, $\dot{q} \in \mathbb{R}^{n \times 1}$ es la posición real producida por sistema, $\dot{q}_d \in \mathbb{R}^{n \times 1}$ es la posición deseada. Entonces al introducir estas dinámicas de error sobre el control PD, este puede reescribirse de la siguiente forma:

$$\tau = K_D \dot{\tilde{q}} + K_P \tilde{q} + K_I \int \tilde{q} dt \quad (4.6)$$

donde $\tau \in \mathbb{R}^{n \times 1}$ es la entrada de control, $K_D \in \mathbb{R}^{n \times n}$, $K_P \in \mathbb{R}^{n \times n}$ y $K_I \in \mathbb{R}^{n \times n}$ son matrices de ganancias las cuales tienen la propiedad de ser definidas positivas y simétricas.

Aplicando la estructura de control (4.6) sobre el sistema robótico de dos grados de libertad dado en la figura 3.1 con los parámetros de la tabla 3.1 el controlador PID genera los siguientes resultados de simulación numérica:

Tiempo de simulación: 20 segundos.

Tipo de integrador: ODE 45

Posición deseada:

$$q_{1d} = \pi / 2,$$

$$q_{2d} = 0,$$

$$\dot{q}_{1d} = 0,$$

$$\dot{q}_{2d} = 0,$$

Paso de integración: 0.1 milisegundos.

Condiciones iniciales:

$$q_1(0) = 0,$$

$$\dot{q}_1(0) = 0,$$

$$q_2(0) = \pi / 2,$$

$$\dot{q}_2(0) = 0.$$

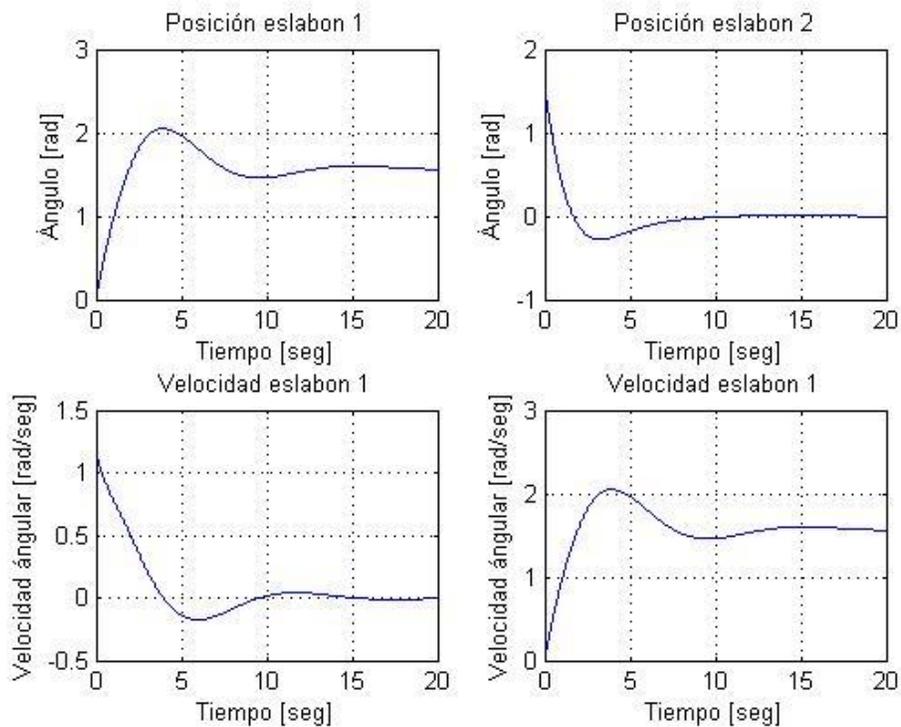


Fig. 4.7: Movimientos y velocidades de las articulaciones del robot con un controlador tipo PID.

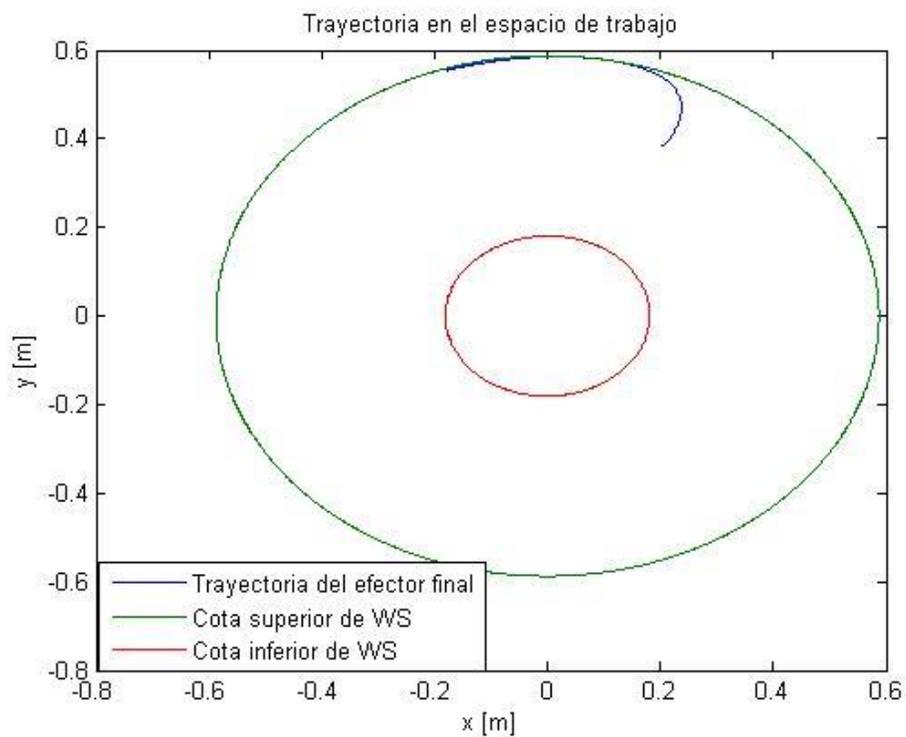


Fig. 4.8: Movimientos en el espacio de trabajo del robot con un controlador tipo PID.

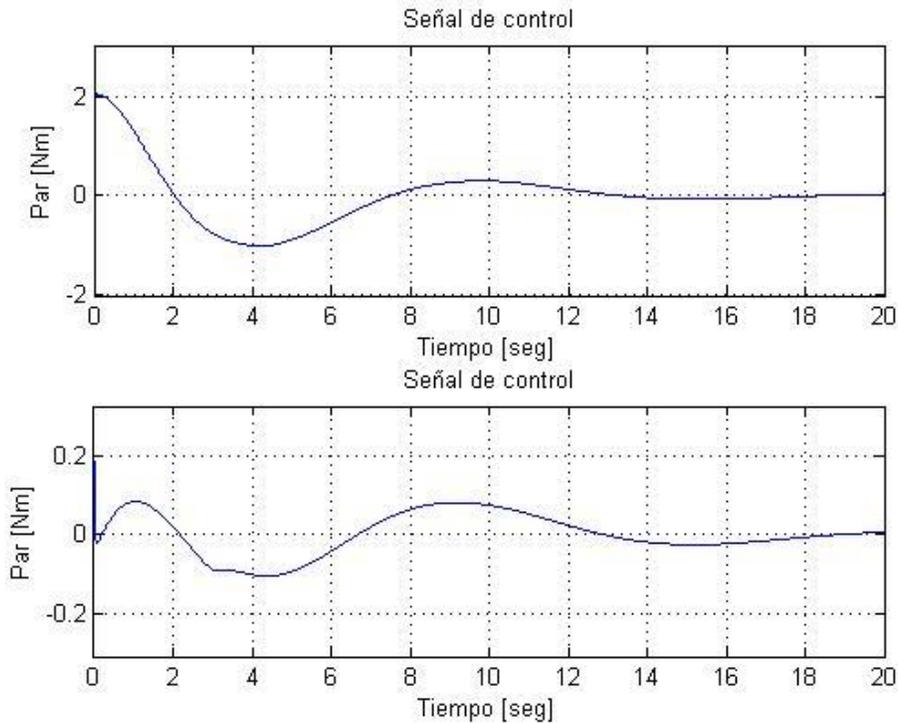


Fig. 4.9: Entradas de control o par generado por el controlador PID.

Como se puede ver en las figuras 4.5 y 4.6 la acción del controlador PID reduce prácticamente en cero el error en estado estacionario lo cual es bueno para el problema de regulación punto a punto, sin embargo el problema propuesto en este proyecto es el de hacer seguimiento de trayectorias, lo cual indica que se debe planificar una trayectoria a seguir en relación a las coordenadas generalizadas, es decir, sobre el plano cartesiano xy.

4.3 Seguimiento de trayectorias.

Para el seguimiento de trayectorias primero se debe diseñar la trayectoria a seguir, es decir la estructura de la tarea que realiza el robot manipulador.

4.3.1 Planificación de la trayectoria.

Los movimientos que se desean para la implementación de las estrategias de control son dadas a partir de lo siguiente:

- La tarea que se desea es determinada por las ecuaciones paramétricas siguientes:

$$\begin{aligned}
 p_x &= r \cos(t)^3 + h, \\
 p_y &= r \sin(t)^3 + k,
 \end{aligned}
 \tag{4.7}$$

donde $h=0.3$, $k=0.3$ y $r=0.4$.

- La velocidad de referencia es determinada por la derivada temporal de la ecuación anterior. La planificación de los movimientos corresponden al espacio de trabajo.
- Donde la trayectoria definida por (4.7) corresponde a la visualización de una estrella de 4 puntas como se puede ver en la figura 4.7

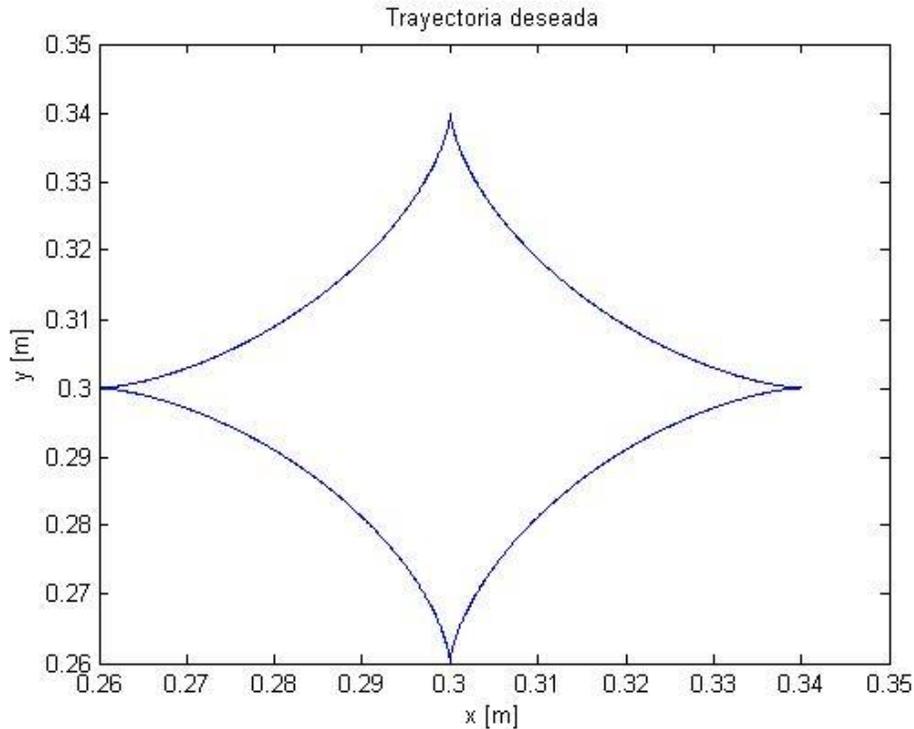


Fig. 4.10: Trayectoria propuesta para el control del manipulador.

Una vez planificada la trayectoria a seguir, se propone aplicar el controlador que obtuvo una mejor respuesta en estado estacionario, es decir, el controlador PID.

4.3.2 Controlador PID.

Como en la sección 4.2 el controlador tendrá la estructura del PID, sin embargo la propuesta de control requiere de parámetros extra, es decir consignas que involucren la derivada temporal e integral respecto al tiempo de la trayectoria propuesta. Para esto se realiza cada operación de forma individual.

Para la parte proporcional el controlador tendrá como consigna de posición deseada la trayectoria descrita en (4.7), sin embargo para la acción derivativa se requiere de la derivada temporal de esta misma, obteniendo lo siguiente:

$$\begin{aligned} \frac{d}{dt}(px) &= \frac{d}{dt}(r \cos(t)^3 + h) = 6r \cos(t)^2 \cos(t) - 3r \cos(t), \\ \frac{d}{dt}(py) &= \frac{d}{dt}(r \sin(t)^3 + k) = 6r \cos(t)^2 \sin(t) - 3r \sin(t). \end{aligned} \quad (4.8)$$

De la misma manera que para la parte derivativa, para la parte integral se requiere de la integral respecto al tiempo de la referencia, es decir:

$$\int p_x dt = \int (r \cos(t)^3 + h) dt = 00,$$

$$\int p_y dt = \int (r \text{sen}(t)^3 + k) dt = 00. \tag{4.9}$$

Una vez obtenidos los parámetros que son requeridos para la aplicación del controlador PID en el caso de seguimiento de trayectorias, el siguiente paso es aplicar el control descrito en (4.6), el cual entrega los siguientes resultados en simulación numérica:

Tipo de integrador: ODE 45

Condiciones iniciales:

$$q_1(0) = 0,$$

$$\dot{q}_1(0) = 0,$$

$$q_2(0) = \pi/2,$$

$$\dot{q}_2(0) = 0.$$

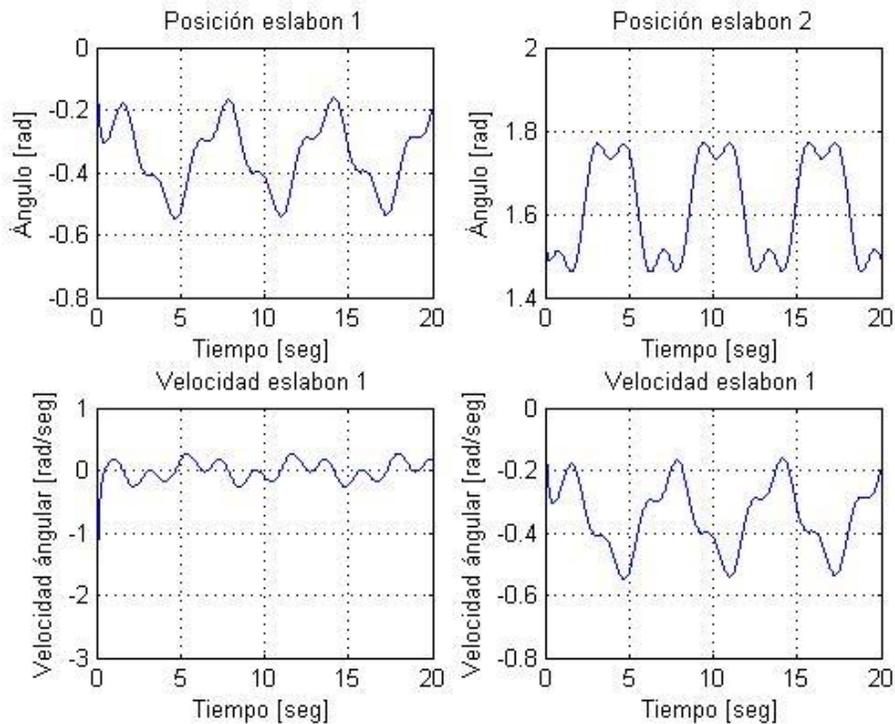


Fig. 4.11: Movimientos y velocidades de las articulaciones del robot con un controlador tipo PID para el caso de seguimiento de trayectorias.

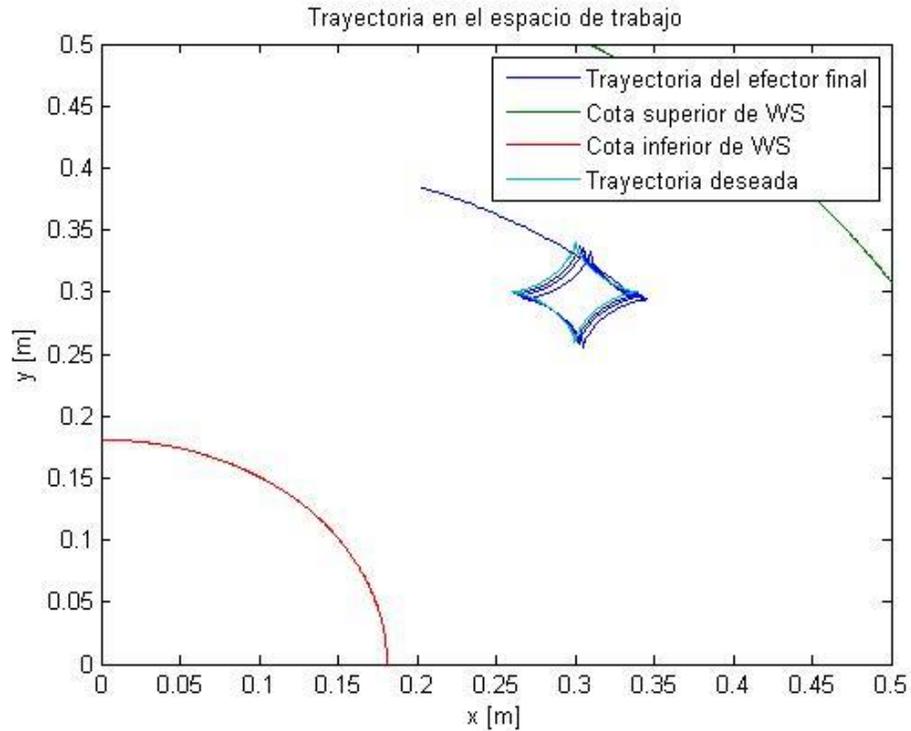


Fig. 4.12: Movimientos en el espacio de trabajo del robot con un controlador tipo PID para el caso de seguimiento de trayectorias.

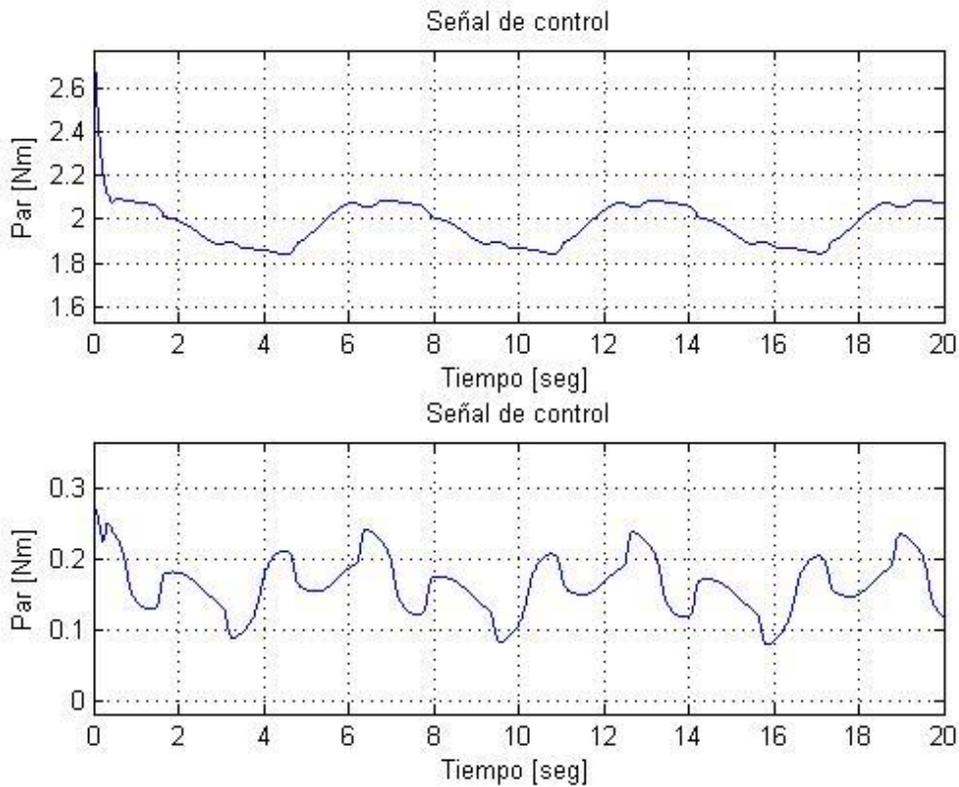


Fig. 4.13: Entradas de control o par generado por el controlador PID para el caso de seguimiento de trayectorias.

Como se puede ver en las figuras 4.8 y 4.9, donde se puede ver el desempeño de las posiciones y las velocidades que se generan al aplicar la ley de control PID, el comportamiento del controlador PID sobre el manipulador de 2GDL es algo deficiente para el problema de seguimiento de trayectorias, también se puede concluir que si el controlador PID es deficiente para el seguimiento de trayectorias, con mucha más razón sus predecesores tendrán un comportamiento más errático, por tal motivo se requiere de un controlador que compense de forma mucho más eficiente este problema

4.4 Controlador clásico con compensación de gravedad.

Como se puede observar en la sección anterior, la estructura clásica de controladores P, PD y PID es deficiente para solucionar el problema de seguimiento de trayectorias, por tal motivo que se requiere de la aplicación de un controlador más avanzado. El controlador aplicado en robots manipuladores con un nivel más complejo, de forma estructural, es el controlador con compensación de gravedad, sin embargo la desventaja en comparación a los controladores anteriores, es que este tipo de controladores requiere del conocimiento de las dinámicas del sistema, que hasta el momento es parte del problema planteado, ya que este trabajo de investigación propone que el sistema sea visto como una caja negra.

La estructura de este controlador es la siguiente:

$$\tau = K_D \dot{\tilde{q}} + K_P \tilde{q} + G(q) \quad (4.10)$$

donde $\tau \in \mathbb{R}^{n \times 1}$ es la entrada de control, $K_D \in \mathbb{R}^{n \times n}$, $K_P \in \mathbb{R}^{n \times n}$ son matrices de ganancias las cuales tienen la propiedad de ser definidas positivas y simétricas, sin embargo el término $G(q) \in \mathbb{R}^{n \times 1}$ involucra las dinámicas del sistema.

Para el caso de estudio del robot manipulador de 2GDL, las dinámicas del término $G(q) \in \mathbb{R}^{n \times 1}$ son las dinámicas generadas por la energía potencial del sistema, es decir se el controlador suprime estas dinámicas, el sistema se comportará como un manipulador de 2GDL cuya dinámica será parecida o en este caso igual a un sistema robótico tipo Scara horizontal y no vertical, realizando así un mejor desempeño al tratar de llegar a las referencias. Por tal motivo en esta sección se presentaran los dos casos de que se han manejado, es decir regulación a un punto y seguimiento de trayectorias.

En el primer caso de estudio donde se aplica el controlador PD con compensación de gravedad se presentan los siguientes parámetros de simulación:

Tiempo de simulación: 20 segundos.

Paso de integración: 0.1 milisegundos.

Tipo de integrador: ODE 45

Condiciones iniciales:

Posición deseada:

$$\begin{aligned} q_{1d} &= \pi/2, \\ q_{2d} &= 0, \\ \dot{q}_{1d} &= 0, \\ \dot{q}_{2d} &= 0, \end{aligned}$$

$$\begin{aligned} q_1(0) &= 0, \\ \dot{q}_1(0) &= 0, \\ q_2(0) &= \pi/2, \\ \dot{q}_2(0) &= 0. \end{aligned}$$

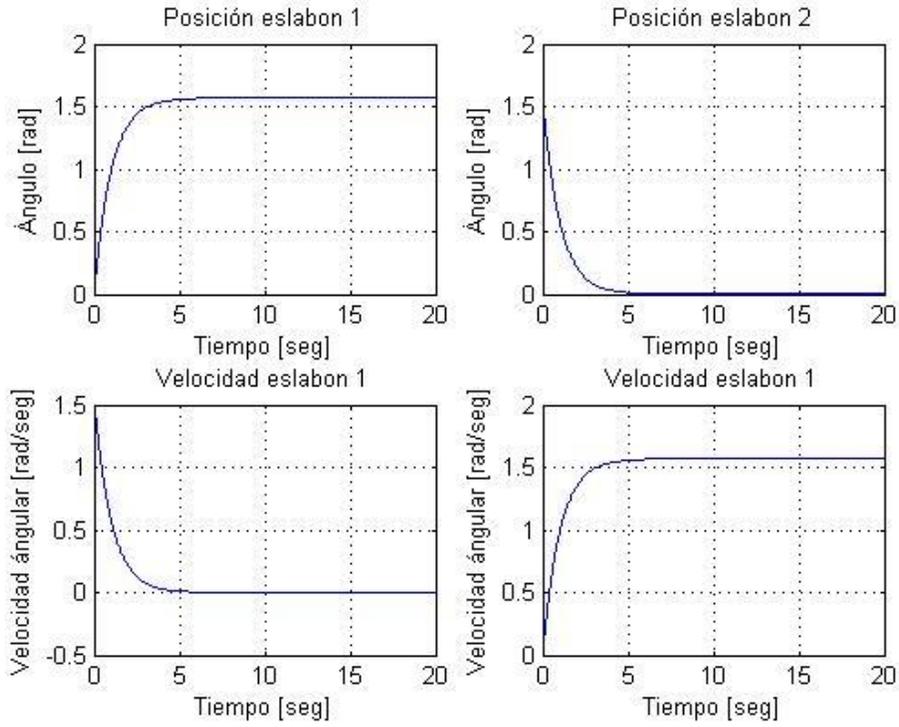


Fig. 4.14: Movimientos y velocidades de las articulaciones del robot con un controlador PD con compensación de gravedad.

donde las dinámicas de gravedad que serán compensadas tienen la siguiente forma:

$$G(q) = \begin{bmatrix} g\theta_4 \cos(q_1) + g\theta_5 \cos(q_1 + q_2) \\ g\theta_5 \cos(q_1 + q_2) \end{bmatrix} \quad (4.11)$$

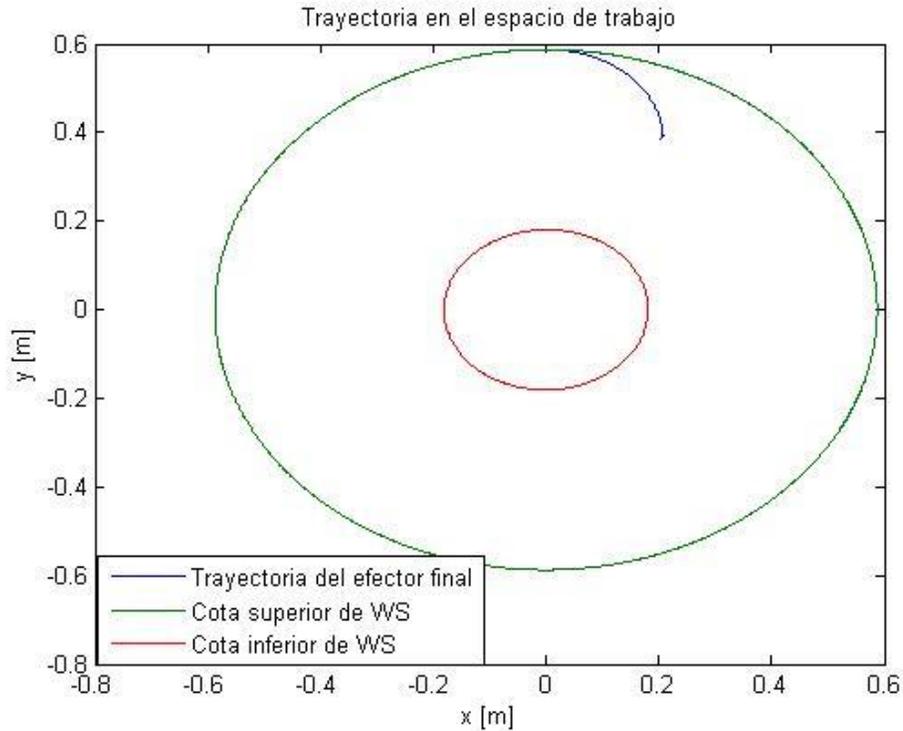


Fig. 4.15: Movimientos en el espacio de trabajo del robot con un controlador PD con compensación de gravedad.

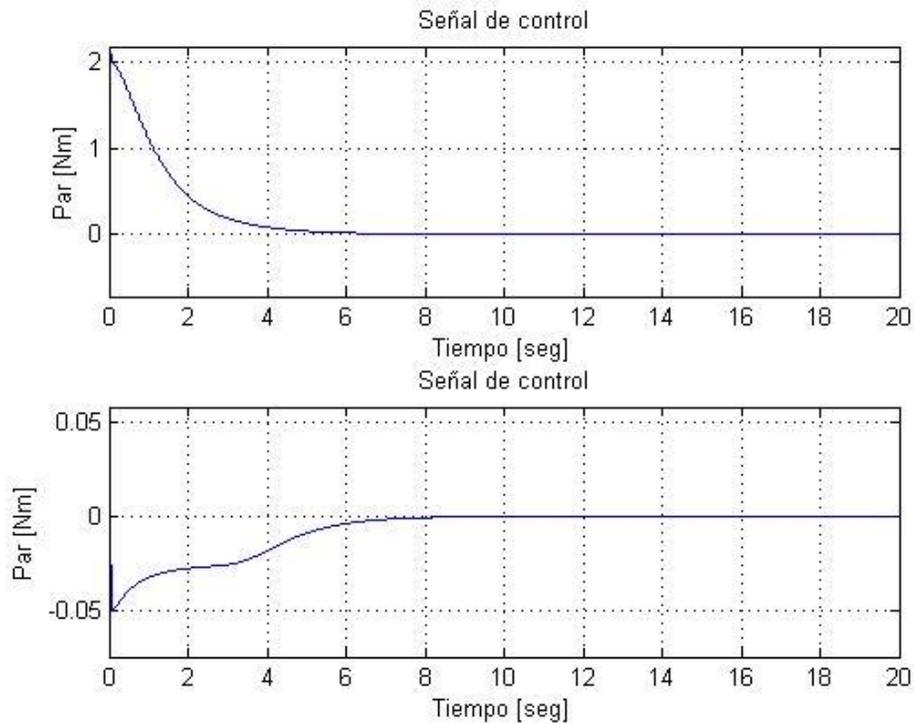


Fig. 4.16: Entradas de control o par generado por el controlador PD con compensación de gravedad.

Tiempo de simulación: 20 segundos.

Paso de integración: 0.1 milisegundos.

Tipo de integrador: ODE 45

Condiciones iniciales:

$$q_1(0) = 0,$$

$$\dot{q}_1(0) = 0,$$

$$q_2(0) = \pi/2,$$

$$\dot{q}_2(0) = 0.$$

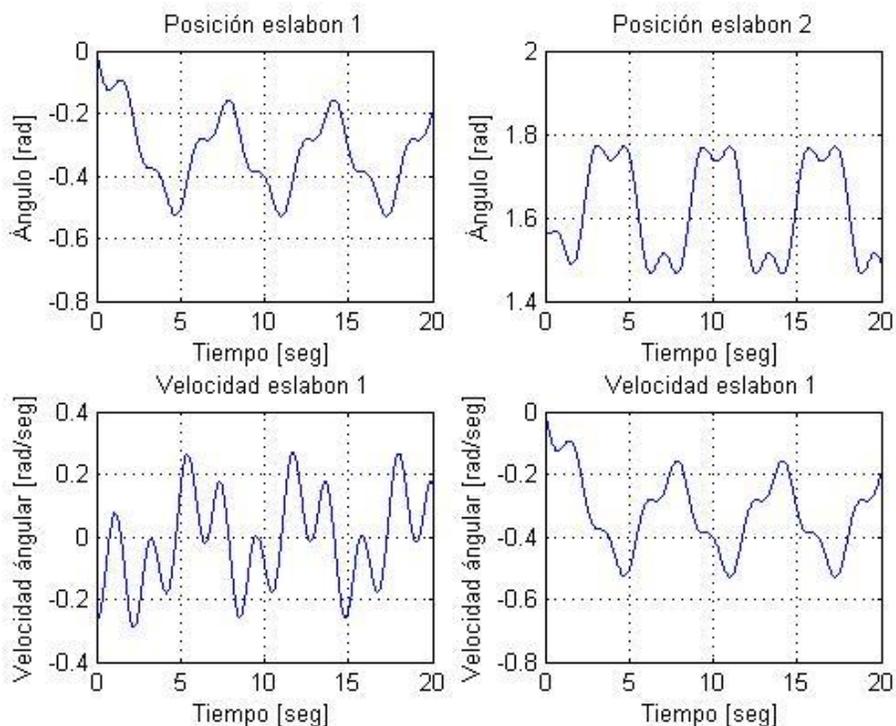


Fig. 4.17: Movimientos y velocidades de las articulaciones del robot con un controlador PD con compensación de gravedad.

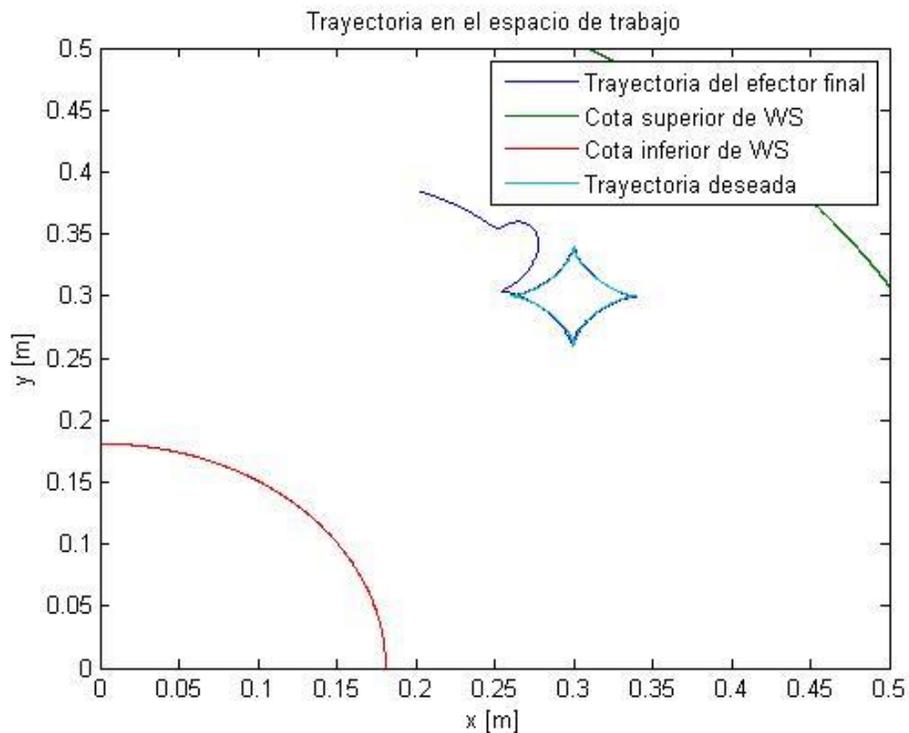


Fig. 4.18: Movimientos en el espacio de trabajo del robot con un controlador PD con compensación de gravedad.

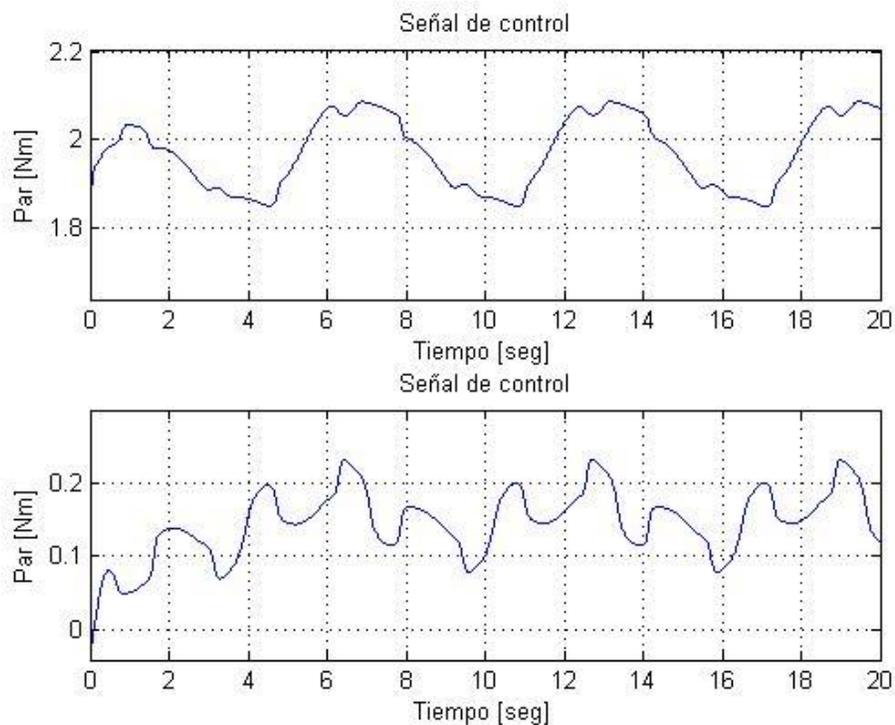


Fig. 4.19: Entradas de control o par generado por el controlador PD con compensación de gravedad para el seguimiento de trayectorias.

4.5 Conclusiones.

Como se puede observar en las diferentes técnicas de control, cada vez que se agregan términos en la dinámica de control, este se modifica de tal manera que se mejora su desempeño esto sucede a costa del esfuerzo sobre el par aplicado. Como se puede ver para el caso de regulación punto a punto, el control tiene un buen desempeño sin embargo cuando se requiere de seguir una trayectoria, el control que funciona de forma eficiente para el caso de regulación, este no funciona para el siguiente caso.

Esto se debe a que el control para regulación está diseñado para funcionar en una región muy cercana al punto de control, sin embargo cuando se cambia a una trayectoria, el control debe funcionar para una región mucho mayor, la cual hace que las dinámicas no lineales se intervengan de forma evidente sobre el desempeño del mismo.

Capítulo 5

Control mediante lógica difusa de un robot de dos grados de libertad.

5.1 Introducción.

La creación de controladores difusos a tenido gran auge actualmente, esto es debido a su naturaleza, y facilidad de diseño, sin embargo la aplicación solo se limitado a sistemas biotecnológicos o bioquímicos ya que es muy difícil obtener un modelo de dichos procesos. Por tal motivo se ha limitado la aplicación de este tipo de controladores.

En años recientes pocos trabajos han dedicado su aplicación a sistemas electromecánicos más a robots manipuladores.

En el presente capítulo se diseñaran controladores difusos, los cuales tengan las características de los controladores diseñados en la sección pasada. Es decir se diseñaran controladores que emulen el controlador PD, y el controlador PID para el caso de regulación punto a punto como para el caso de seguimiento de trayectorias.

5.2 Regulación del robot 2GDL mediante lógica difusa.

Del capítulo 2 se puede ver que el proceso de control difuso puede ser descrito mediante una serie de pasos, así que primero hay que identificar el sistema a controlar (en este caso un robot de 2GDL definido en la figura 3.1), para el cual hay que definir los parámetros del universo de discurso para cada variable a controlar del sistema, esto con el objetivo de crear la base de reglas.

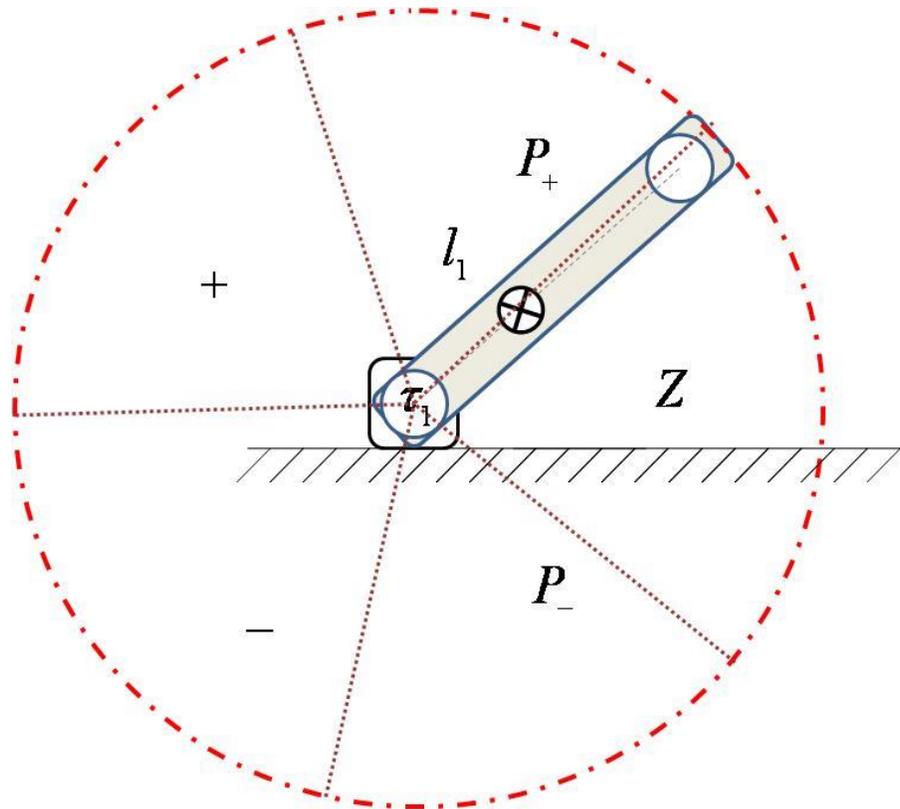


Fig. 5.1: Determinación de las variables lingüísticas respecto al sistema.

Donde los valores Z , P_+ , $+$, P_- , y $-$ son valores determinados por el experto (para este caso la variable Z que es cero, está sobre el origen del sistema), y estas variables son cero, poco positivo, positivo, poco negativo, y negativo.

Por lo cual ya están definidas las variables lingüísticas respecto a la posición del primer eslabón, y seguimos con las variables del segundo eslabón.

De las figuras 5.1 y 5.2 se pueden determinar las variables lingüísticas y los valores de las mismas, entonces haciendo referencia al capítulo 2 sección 2 y 3 la creación de las funciones de membresía son referidas a las posiciones del sistema, para lo cual se propone que estas funciones se comporten de la siguiente manera, dentro de un universo descrito por los 360° que tiene en las posiciones cada eslabón.

Por tal motivo, se pretende que el universo de discurso se desarrolle en un rango de $\pm 180^\circ$ o $\pm \pi$ sobre el espacio de trabajo de cada eslabón. De tal manera que se obtienen 3 funciones de membresía en un rango de $\pm \pi$, para este control se propone que las funciones de membresía sean del tipo trapezoidal (ver figura 5.3) esto se debe a que solo se requiere de regulación punto a punto.

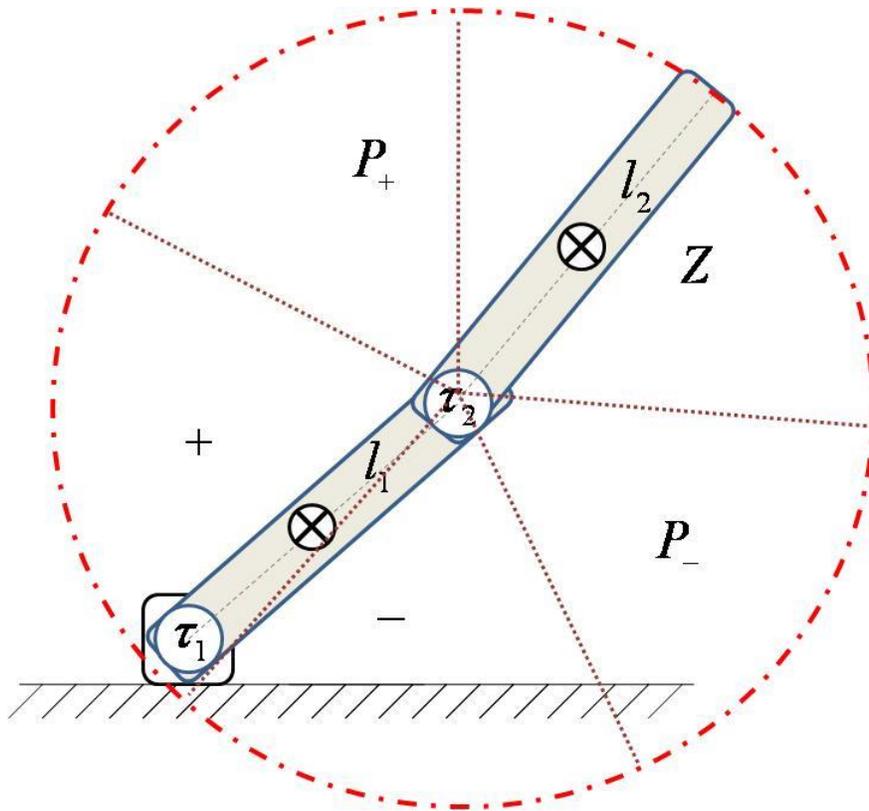


Fig. 5.2: Determinación de las variables lingüísticas respecto al sistema (segundo eslabón).

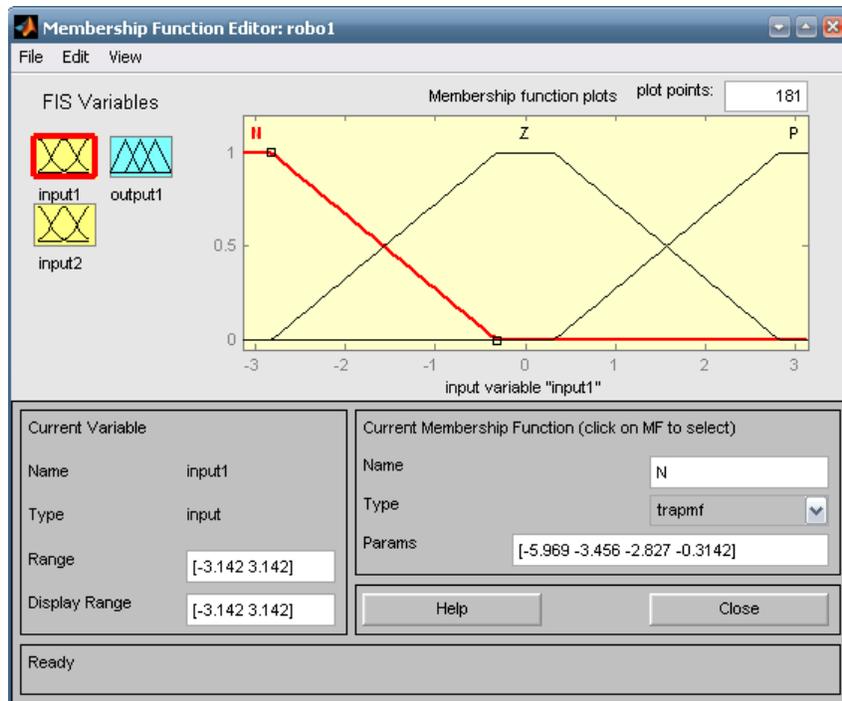


Fig. 5.3: Función de membresía para el primer y segundo eslabón.

Donde las funciones de membresía fueron diseñadas (como se menciono anteriormente) del tipo trapezoidal, las cuales tienen los siguientes parámetros:

Función de membresía	Primer parámetro (a)	Segundo parámetro (b)	Tercer parámetro (c)	Cuarto parámetro (d)
Negativa	-5.96	-3.456	-2.87	-0.3
Cero	-2.87	-0.3	0.3	3.87
Positiva	0.3	2.87	3.46	5.96

Tabla. 5.1: Parámetros de las funciones de membresía de las variables de control.

De la misma forma en que se diseñan las funciones de membresía para la posición se diseñan las funciones de membresía para la velocidad (esta corresponde a las figura 5.4),

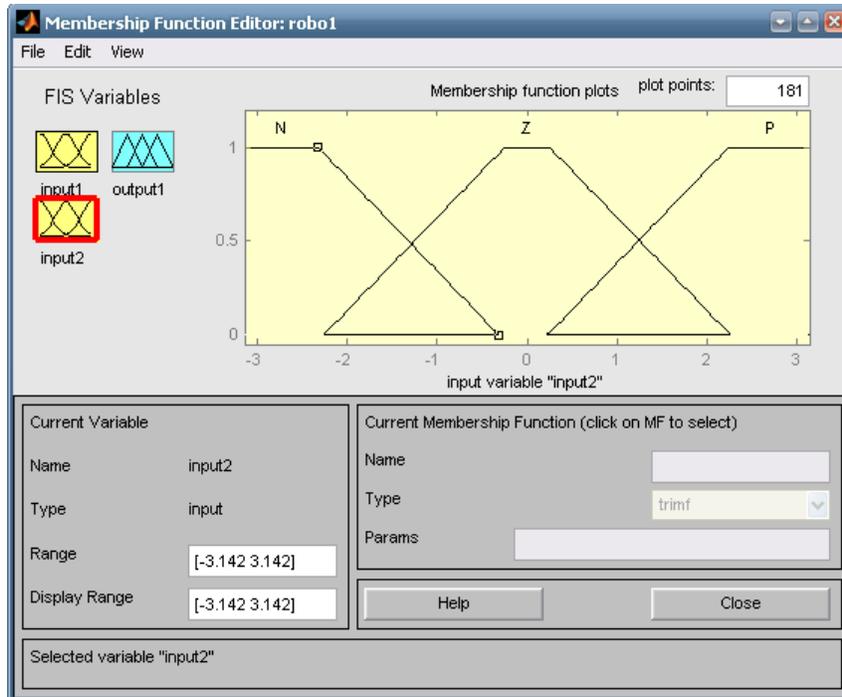


Fig. 5.4: Función de membresía para la velocidad para el primer y segundo eslabón.

Para el caso de las variables de velocidad, se proponen funciones de membresía (tipo trapezoidal) relativas a velocidad de la siguiente manera:

Función de membresía	Primer parámetro (a)	Segundo parámetro (b)	Tercer parámetro (c)	Cuarto parámetro (d)
Negativa	-5.96	-3.8	-2.5	0
Cero	-2.5	-0.4	0.4	2.5
Positiva	0	2.5	3.8	5.96

Tabla. 5.2: Parámetros de las funciones de membresía de las variables de control, respecto a las velocidades del sistema.

De acuerdo con estas funciones de entrada se determina que las funciones de membresía para las salidas de control para cada eslabón en particular sean descritas de la misma forma que las de entrada, sin embargo, para este caso en vez de definir 3 parámetros de control para las variables lingüísticas se dice que existan 5 en las salidas, esto con el

objetivo de hacer más variado el control al momento de crear las reglas, por lo tanto las funciones de membresía para la salida son ajustadas de la siguiente manera:

Función de membresía	Primer parámetro (a)	Segundo parámetro (b)	Tercer parámetro (c)	Cuarto parámetro (d)
Negativa	-50	-40	-30	-10
Poco negativa	-19	-11	-9	-1
Cero	-9	-1	1	9
Poco positiva	1	9	11	19
Positiva	10	30	40	50

Tabla. 5.3: Parámetros de las funciones de membresía de las salidas de control.

De las premisas anteriores se puede generar las funciones de membresía para la salida del control en la herramienta de Matlab de la siguiente forma:

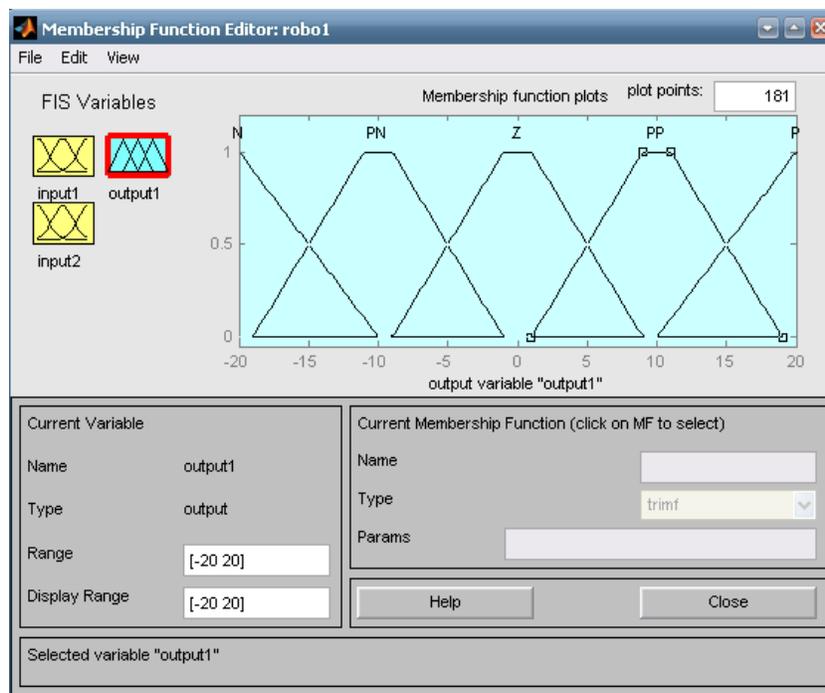


Fig. 5.5: Función de membresía para la generación de señales de control.

Para el control de este sistema se propone que la salida sea acotada en un rango de $\pm 20 Nm$, lo cual es razonable para el control de un motor de CD.

Entonces una vez obtenidas las funciones de membresía que emulan el conocimiento de los casos que puede generar el sistema, el siguiente paso es el de emular las decisiones del experto, por lo que hay que generar una serie de reglas con las sentencias “*Si la posición es A y Si la velocidad es B, entonces la Salida de control es C*”, para lo cual como experto se proponen la siguiente base de reglas o sentencias:

1. Si la posición es N y tiene velocidad N entonces la Salida es P
2. Si la posición es N y tiene velocidad Z entonces la Salida es PP
3. Si la posición es N y tiene velocidad P entonces la Salida es Z
4. Si la posición es Z y tiene velocidad N entonces la Salida es PP
5. Si la posición es Z y tiene velocidad Z entonces la Salida es Z
6. Si la posición es Z y tiene velocidad P entonces la Salida es PN
7. Si la posición es P y tiene velocidad N entonces la Salida es Z
8. Si la posición es P y tiene velocidad Z entonces la Salida es PN
9. Si la posición es P y tiene velocidad P entonces la Salida es N

Esta serie de reglas también puede ser vista como una matriz la cual puede verse de la siguiente forma:

<i>Pocisión</i>	Negativa	Cero	Positiva
<i>Velocidad</i>			
Negativa	Positiva	Poco Positiva	Cero
Cero	Poco Positiva	Cero	Poco Negativa
Positiva	Cero	Poco Negativa	Negativa

Tabla. 5.4: Matriz de control.

Lo cual es una ventaja respecto a la base de reglas, ya que la matriz de reglas puede verse por campos De esta misma forma y como las variables lingüísticas respecto al segundo eslabón son de forma similar, entonces la base de reglas propuesta para este caso tendrá la misma estructura que en la tabla 5.4. El procedimiento anterior da la cuantificación lógica del experto sobre la planta.

El paso siguiente del control difuso, es elegir la interfase de fusificación, el cual tiene el objetivo de aplicar las decisiones del experto y transformarlas a la activación de las reglas para la toma de decisiones. En este caso la herramienta de Matlab lo hace directamente, sin embargo de forma lógica se toma el procedimiento de máx.-min, T-norma u S-norma (este procedimiento puede verse en la figura 5.6).

Y como último paso se propone el método de defusificación, este método transforma la toma de decisiones en variables numéricas las cuales hacen que el proceso de control sea el correcto, es decir, el control no envía variables tipo lingüísticas como poco control, sin control, etc. por lo que envira valores numéricos como 5.2, 0.3, etc.

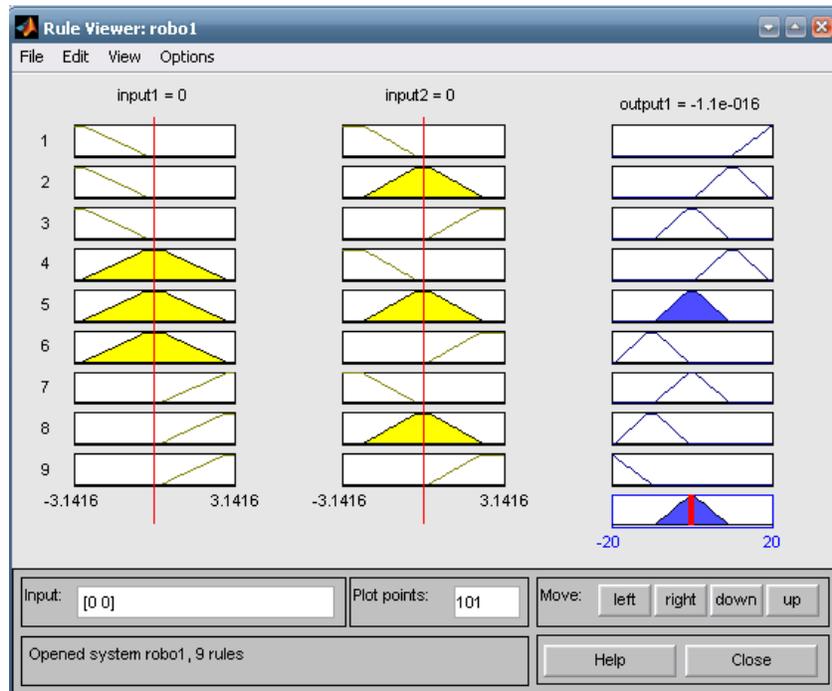


Fig. 5.6: Proceso de defusificación.

El proceso de defusificación que se elige es el clásico, es decir defusificación por centro de gravedad. Entonces al aplicar las consignas de control difuso sobre la plataforma propuesta (robot 2GDL) bajo las siguientes condiciones de simulación numérica, se obtienen los resultados que a continuación se describen.

Como en los casos de control del capítulo 4, se propone un sistema con baja fricción, de acuerdo a esto las simulaciones generan lo siguiente:

Tiempo de simulación: 20 segundos.

Tipo de integrador: ODE 45

Posición deseada:

$$q_{1d} = \pi / 2,$$

$$q_{2d} = 0,$$

$$\dot{q}_{1d} = 0,$$

$$\dot{q}_{2d} = 0,$$

Paso de integración: 0.1 milisegundos.

Condiciones iniciales:

$$q_1(0) = 0,$$

$$\dot{q}_1(0) = 0,$$

$$q_2(0) = \pi / 2,$$

$$\dot{q}_2(0) = 0.$$

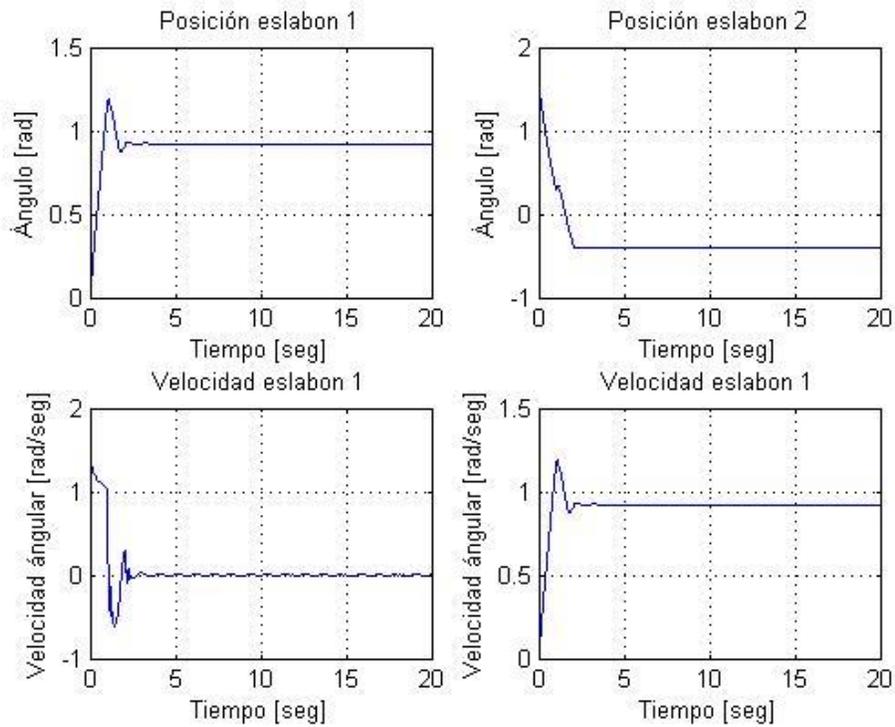


Fig. 5.7: Movimientos y velocidades de las articulaciones del robot con un controlador tipo PD difuso.

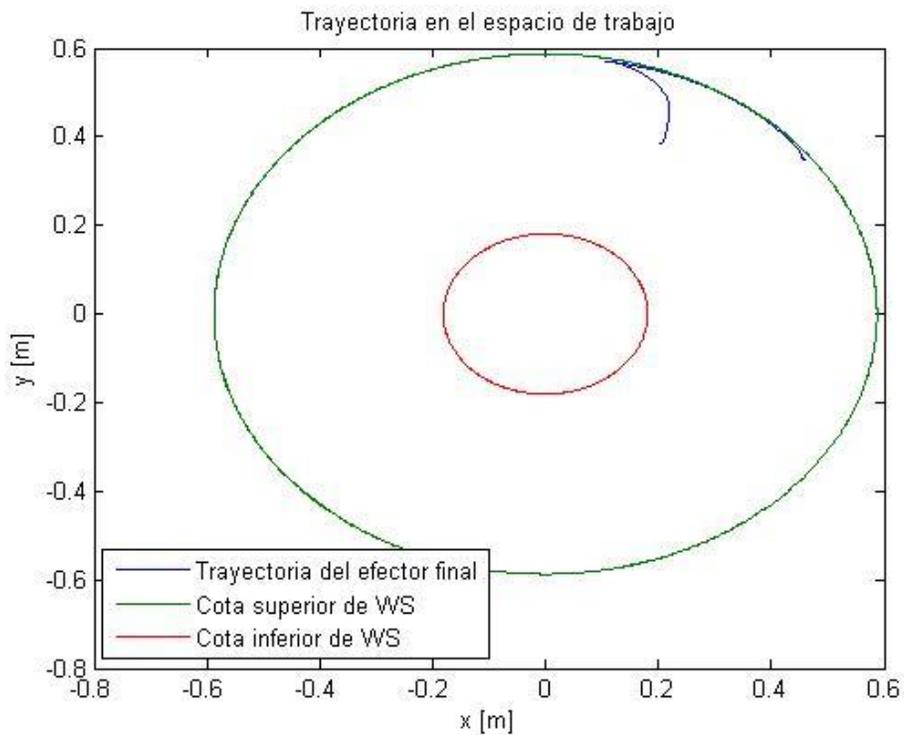


Fig. 5.8: Movimientos en el espacio de trabajo del robot con un controlador tipo PD difuso.

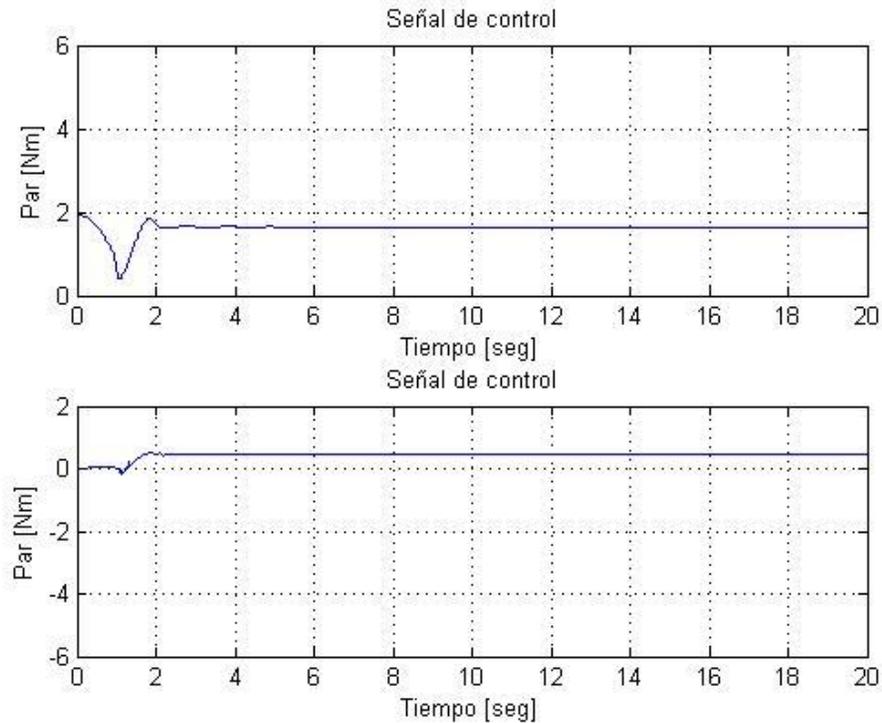


Fig. 5.9: Entradas de control o par generado por el controlador PID.

Como se puede observar en las figuras 5.7, 5.8 y 5.9, el controlador heurístico en configuración PD no estabiliza al sistema en el punto deseado, por tal motivo se propone que dentro de las dinámicas de error, estas estén sujetas a una ganancia proporcional, entonces si el error es pequeño, este será incrementado, de esta manera enviará una consigna de control más fuerte de este modo estabilizará al sistema en el punto deseado (de esta propuesta se obtienen los siguientes resultados).

Tiempo de simulación: 20 segundos.

Tipo de integrador: ODE 45

Posición deseada:

$$q_{1d} = \pi / 2,$$

$$q_{2d} = 0,$$

$$\dot{q}_{1d} = 0,$$

$$\dot{q}_{2d} = 0,$$

Paso de integración: 0.1 milisegundos.

Ganancias de Control:

$$K1=5, K2=4.$$

Condiciones iniciales:

$$q_1(0) = 0,$$

$$\dot{q}_1(0) = 0,$$

$$q_2(0) = \pi / 2,$$

$$\dot{q}_2(0) = 0.$$

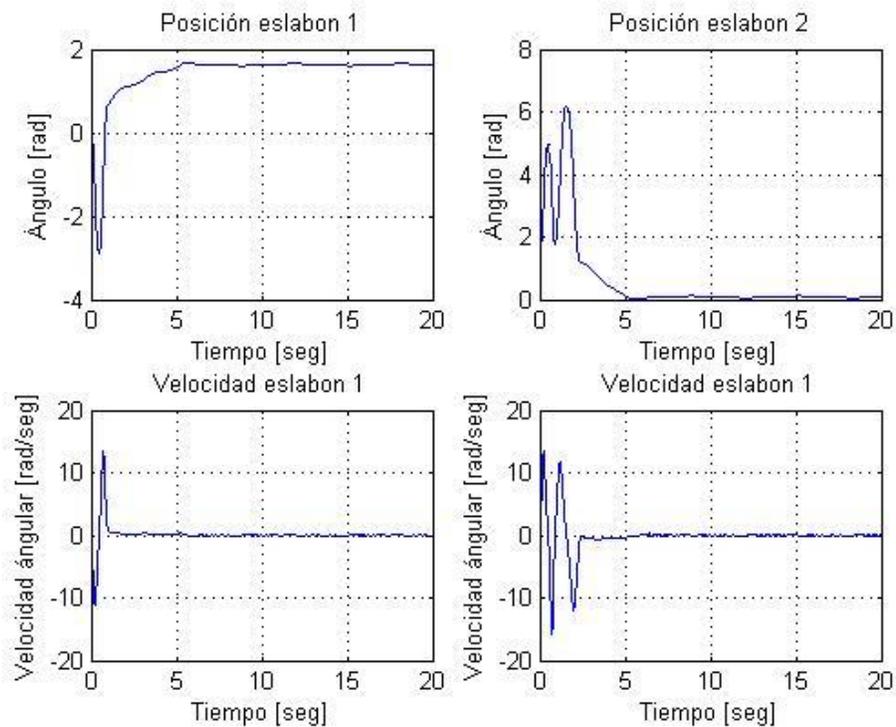


Fig. 5.10: Movimientos y velocidades de las articulaciones del robot con un controlador tipo PD difuso.

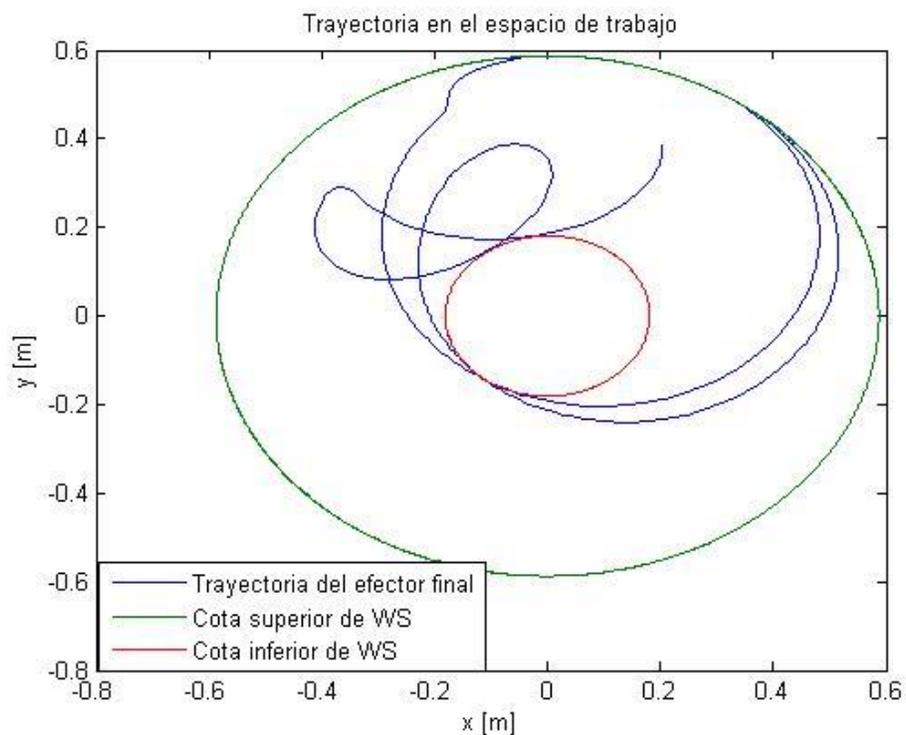


Fig. 5.11: Movimientos en el espacio de trabajo del robot con un controlador tipo PD difuso.

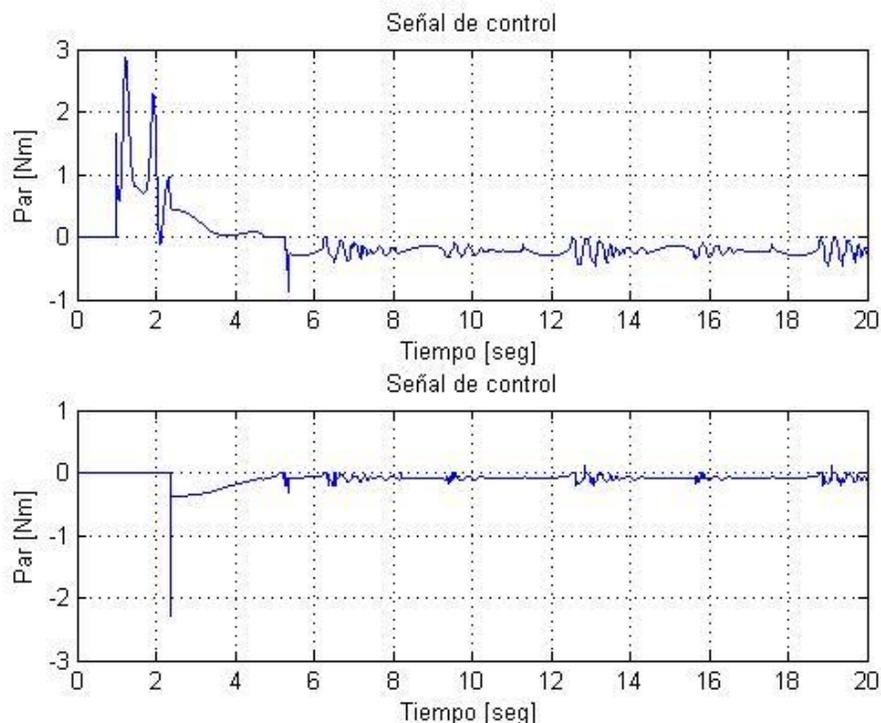


Fig. 5.12: Entradas de control o par generado por el controlador PID.

Como se puede observar, a este control se le dificulta llegar a la referencia ya que alcanza la misma en un tiempo aproximado en 5 segundos (figura 5.10 y 5.11), mientras que la ley de control tiene un comportamiento errático (figura 5.12), por lo que se propone hacer un controlador que contenga una configuración de reglas más variada. Por tal motivo se requiere de un controlador que resuelva este problema, y el controlador clásico más indicado es el controlador PD difuso con ganancias de error. Por lo que se propone agregar dos nuevas funciones de control el cual sea del tipo integral, es decir incluir una serie de reglas que realice la acción integral para cada control. Para lograr tal efecto se propone lo siguiente:

Definimos funciones de membresía de tal forma que los errores sean incrementados de forma lineal, obteniendo una entrada, una salida, donde las entradas son determinadas de la siguiente forma:

Función de membresía	Primer parámetro (a)	Segundo parámetro (b)	Tercer parámetro (c)	Cuarto parámetro (d)
Negativa	-6.28	-6.28/3	-3.14	-3.14/2
Cero	-3.14	-3.14/2	3.14/2	3.14
Positiva	3.14/2	3.14	6.28/3	6.28

Tabla. 5.5: Parámetros de las funciones de membresía de las variables para el cambio de variable.

De la misma manera se definen las funciones de membresía para la salida de la siguiente forma:

Función de membresía	Primer parámetro (a)	Segundo parámetro (b)	Tercer parámetro (c)
Salida	Tipo lineal $f(x) = x + 3$	Tipo lineal $f(x) = 2x + 6$	Tipo lineal $f(x) = x + 4$

Tabla. 5.6: Parámetros de las funciones de membresía de las salidas para el cambio de variable.

A este tipo de control se le llama, controlador difuso tipo Sugeno, ya que las salidas son funciones, para el caso de estudio se proponen lineales, de tal manera que emule el control como una acción integral, y de esta forma se obtienen los siguientes resultados:

Tiempo de simulación: 20 segundos.

Paso de integración: 0.1 milisegundos.

Tipo de integrador: ODE 45

Ganancias de Control:

Posición deseada:

$K1=5, K2=4, Ki=2.$

$$q_{1d} = \pi / 2,$$

Condiciones iniciales:

$$q_{2d} = 0,$$

$$q_1(0) = 0,$$

$$\dot{q}_{1d} = 0,$$

$$\dot{q}_1(0) = 0,$$

$$\dot{q}_{2d} = 0,$$

$$q_2(0) = \pi / 2,$$

$$\dot{q}_2(0) = 0.$$

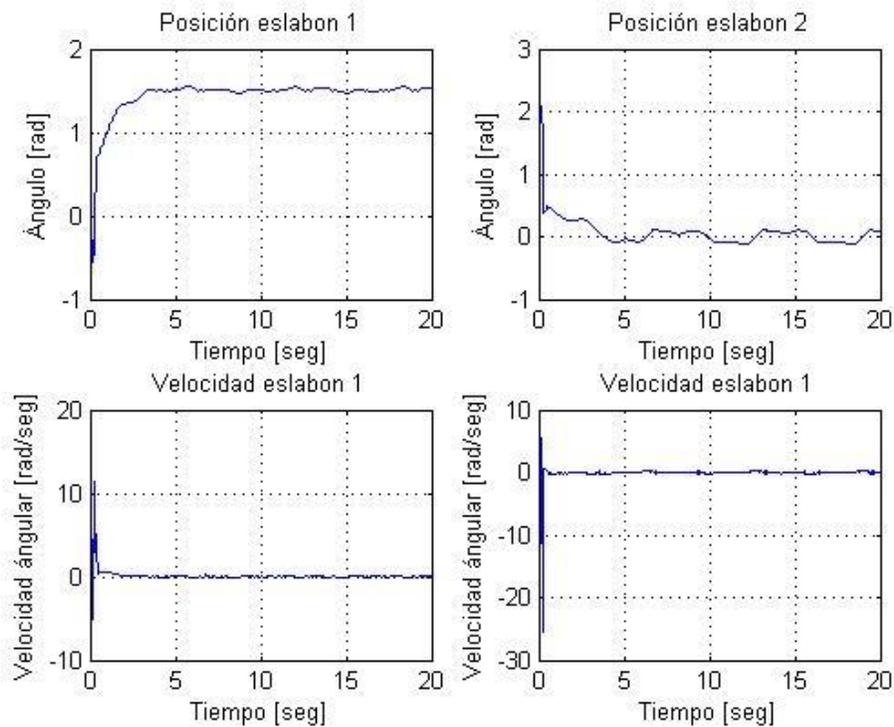


Fig. 5.13: Movimientos y velocidades de las articulaciones del robot con un controlador tipo PD difuso.

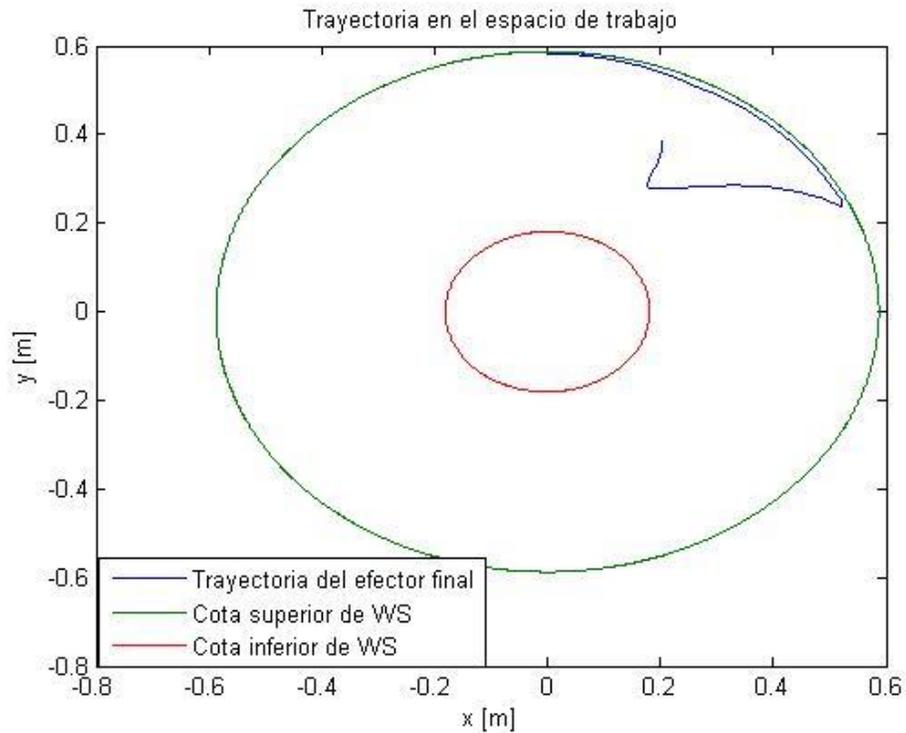


Fig. 5.14: Movimientos en el espacio de trabajo del robot con un controlador tipo PD difuso.

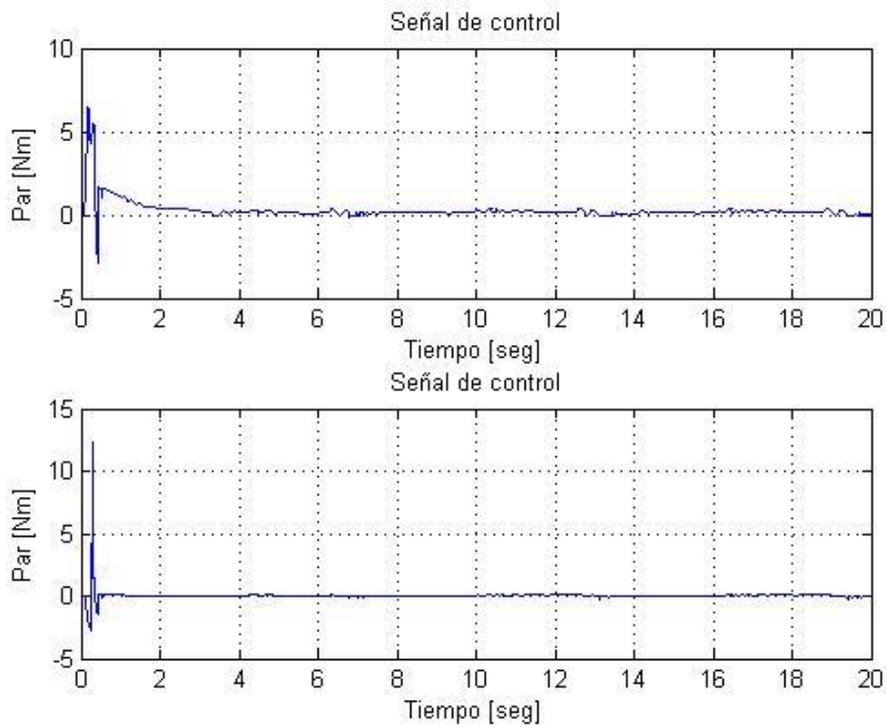


Fig. 5.15: Entradas de control o par generado por el controlador PID.

Como se puede observar en la figura 5.13 el control llega a la referencia sin problemas, sin embargo esté control aún tiene una serie de perturbaciones, entonces hay que resolver este problema con la asignación de nuevas funciones de membresía y en consecuencia una serie de nuevas reglas. Por lo que las variables pueden ser descritas como lo muestran las figuras 5.16 y 5.17.

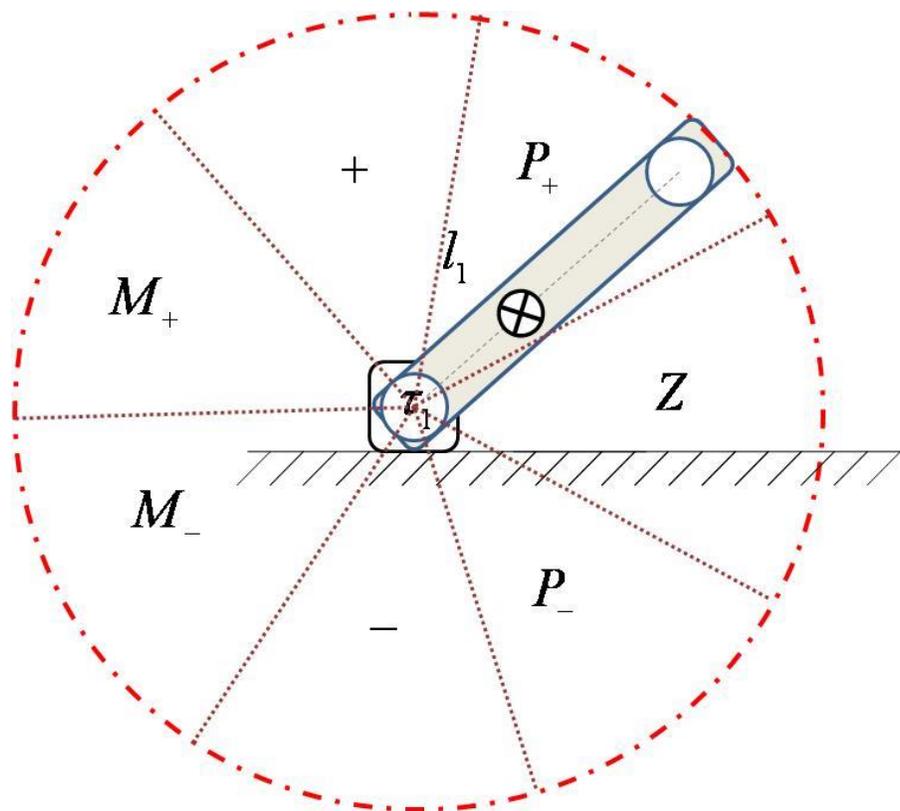


Fig. 5.16: Determinación de las variables lingüísticas respecto al sistema.

Las nuevas funciones de membresía son propuestas del tipo triangular, y sus parámetros están determinados de la siguiente forma:

Función de membresía	Primer parámetro	Segundo parámetro	Tercer parámetro
Negativa	-4.17	-3.142	-1.571
Poco negativa	-3.142	-1.571	0
Cero	-1.571	0	1.571
Poco positiva	0	1.571	3.142
Positiva	1.571	3.142	4.17

Tabla. 5.7: Parámetros de las funciones de membresía para nuevas reglas.

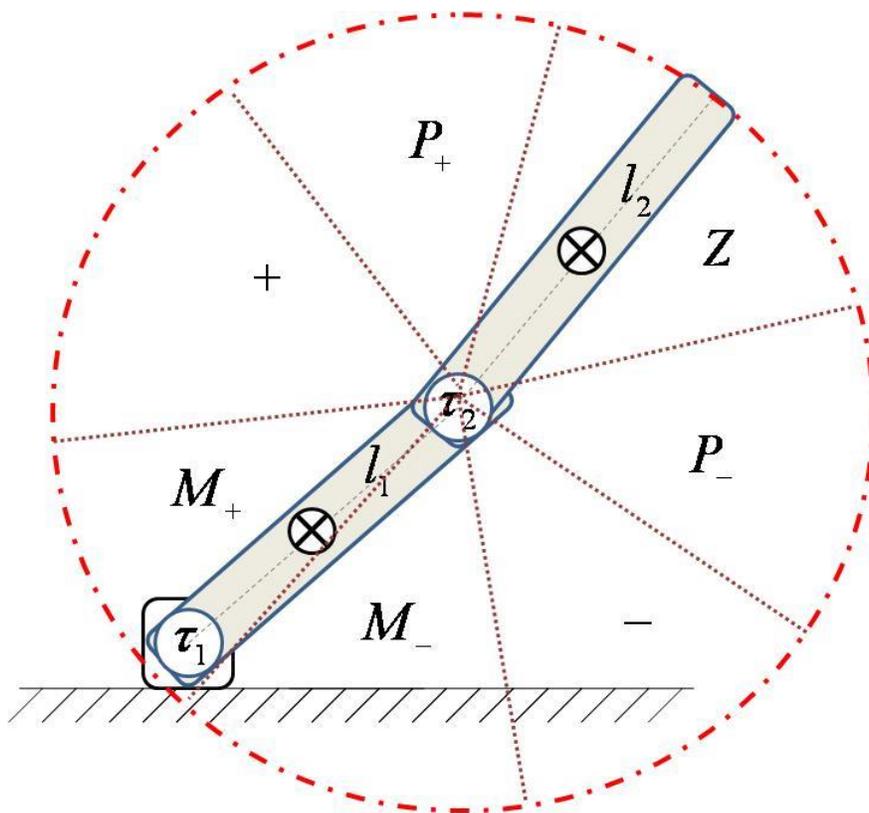


Fig. 5.17: Determinación de las variables lingüísticas respecto al sistema (segundo eslabón).

De igual forma para las funciones de salida, en este caso se tomen otras dos funciones de membresía nuevas, y en consecuencia cambian los valores de las mismas sobre el universo de discurso. Obteniendo lo siguiente:

Función de membresía	Primer parámetro	Segundo parámetro	Tercer parámetro
Muy Negativa	-40	-30	-20
Negativa	-30	-19	-10
Poco Negativa	-20	-10	0
Cero	-10	0	10
Poco Positiva	0	10	20
Positiva	10	20	30
Muy Positiva	20	30	40

Tabla. 5.8: Parámetros de las funciones de membresía de salida para nuevas reglas.

Así una nueva generación de reglas difusas es creada, de tal manera que estas pueden ser escritas de la siguiente forma:

Entonces una vez obtenidas las funciones de membresía que emulan el conocimiento de los casos que puede generar el sistema, el siguiente paso es el de emular las decisiones del experto, por lo que hay que generar una serie de reglas con las sentencias “*Si la posición es A y Si la velocidad es B, entonces la Salida de control es C*”, para lo cual como experto se proponen la siguiente base de reglas o sentencias:

1. Si la posición es N	y tiene velocidad N	entonces	la Salida es MP
2. Si la posición es N	y tiene velocidad PN	entonces	la Salida es MP
3. Si la posición es N	y tiene velocidad Z	entonces	la Salida es P
4. Si la posición es N	y tiene velocidad PP	entonces	la Salida es PP
5. Si la posición es N	y tiene velocidad P	entonces	la Salida es Z
6. Si la posición es PN	y tiene velocidad N	entonces	la Salida es MP
7. Si la posición es PN	y tiene velocidad PN	entonces	la Salida es P
8. Si la posición es PN	y tiene velocidad Z	entonces	la Salida es PP
9. Si la posición es PN	y tiene velocidad PP	entonces	la Salida es Z
10. Si la posición es PN	y tiene velocidad P	entonces	la Salida es PN
11. Si la posición es Z	y tiene velocidad N	entonces	la Salida es P
12. Si la posición es Z	y tiene velocidad PN	entonces	la Salida es PP
13. Si la posición es Z	y tiene velocidad Z	entonces	la Salida es Z
14. Si la posición es Z	y tiene velocidad PP	entonces	la Salida es PN
15. Si la posición es Z	y tiene velocidad P	entonces	la Salida es N
16. Si la posición es PP	y tiene velocidad N	entonces	la Salida es PP
17. Si la posición es PP	y tiene velocidad PN	entonces	la Salida es Z
18. Si la posición es PP	y tiene velocidad Z	entonces	la Salida es PN
19. Si la posición es PP	y tiene velocidad PP	entonces	la Salida es N
20. Si la posición es PP	y tiene velocidad P	entonces	la Salida es MN
21. Si la posición es P	y tiene velocidad N	entonces	la Salida es Z
22. Si la posición es p	y tiene velocidad PN	entonces	la Salida es PN
23. Si la posición es p	y tiene velocidad Z	entonces	la Salida es N
24. Si la posición es p	y tiene velocidad PP	entonces	la Salida es MN
25. Si la posición es P	y tiene velocidad P	entonces	la Salida es MN

Esta serie de reglas también puede ser vista como una matriz la cual puede verse de la siguiente forma:

<i>Pocisión</i> <i>Velocidad</i>	Negativa	Poco Negativa	Cero	Poco Positiva	Positiva
Negativa	Muy positiva	Muy positiva	Poco positiva	Positiva	Cero
Poco Negativa	Muy positiva	Poco positiva	Positiva	Cero	Poco Negativa
Cero	Poco positiva	Positiva	Cero	Poco Negativa	Negativa
Poco Positiva	Positiva	Cero	Poco Negativa	Negativa	Muy Negativa
Positiva	Cero	Poco Negativa	Negativa	Muy Negativa	Muy Negativa

Tabla. 5.9: Matriz de control.

De esta forma las nuevas funciones de membresía quedan estructuradas sobre la herramienta de lógica difusa de la siguiente manera:

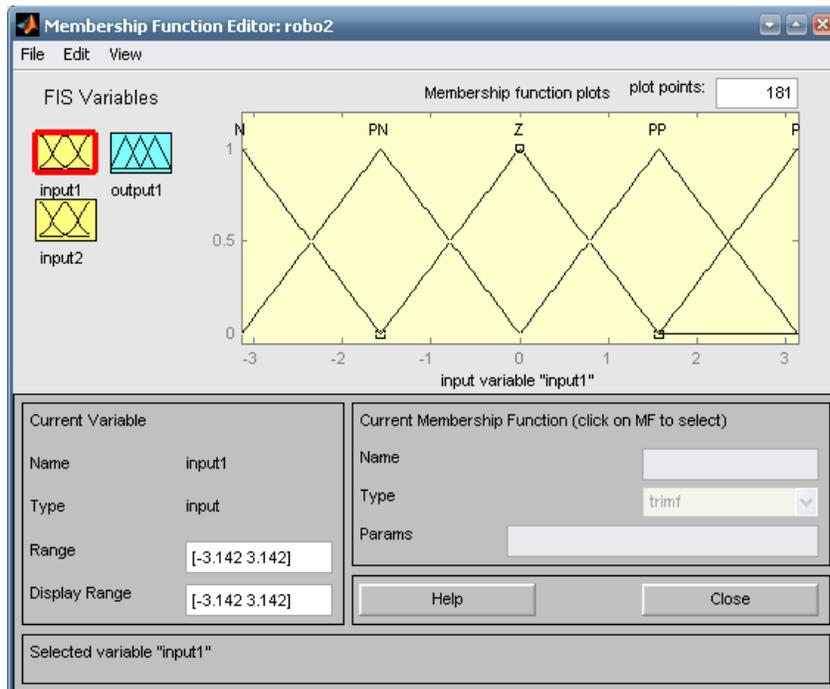


Fig. 5.18: Función de membresía para la generación de señales de entrada.

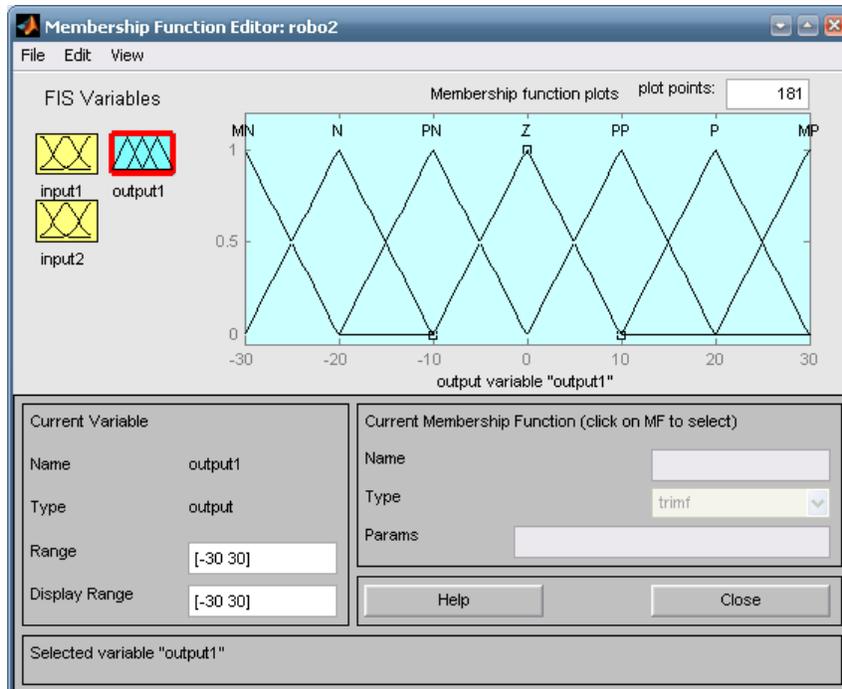


Fig. 5.19: Función de membresía para la generación de señales de entrada.

Y el método de defusificación, este método transforma la toma de decisiones en variables numéricas las cuales hacen que el proceso de control sea el correcto, es decir, el control no envía variables tipo lingüísticas como poco control, sin control, etc. por lo que envira valores numéricos como 5.2, 0.3, etc.

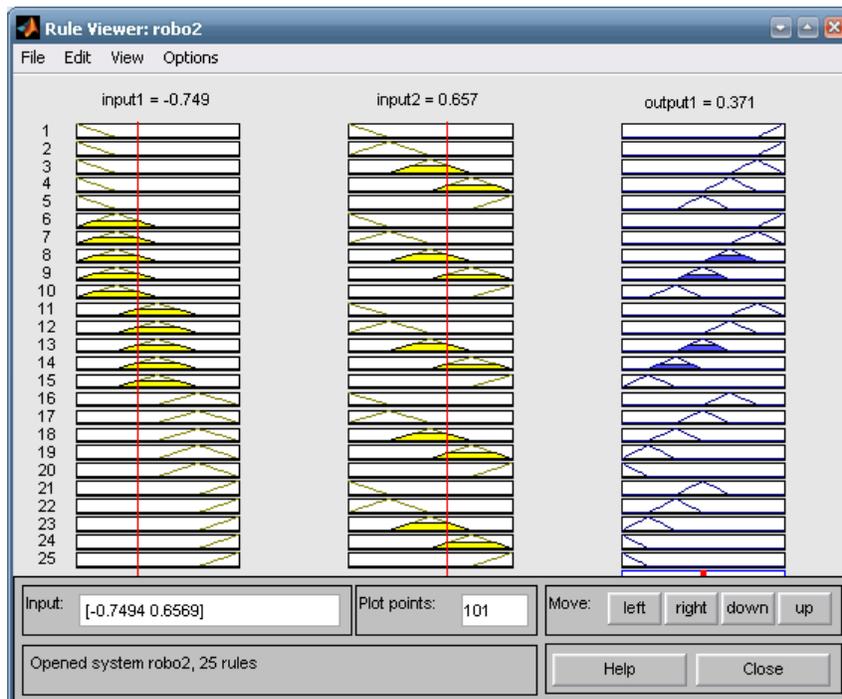


Fig. 5.20: Proceso de defusificación con las nuevas reglas.

El proceso de defusificación que se elige es el clásico, es decir defusificación por centro de gravedad. Entonces al aplicar las consignas de control difuso sobre la plataforma propuesta (robot 2GDL) bajo las siguientes condiciones de simulación numérica, se obtienen los resultados que a continuación se describen.

Como en los casos de control del capítulo 4, se propone un sistema con baja fricción, de acuerdo a esto las simulaciones generan lo siguiente:

Tiempo de simulación: 20 segundos.

Tipo de integrador: ODE 45

Posición deseada:

$$q_{1d} = \pi / 2,$$

$$q_{2d} = 0,$$

$$\dot{q}_{1d} = 0,$$

$$\dot{q}_{2d} = 0,$$

Paso de integración: 0.1 milisegundos.

Ganancias de Control:

$$K1=5, K2=4.$$

Condiciones iniciales:

$$q_1(0) = 0,$$

$$\dot{q}_1(0) = 0,$$

$$q_2(0) = \pi / 2,$$

$$\dot{q}_2(0) = 0.$$

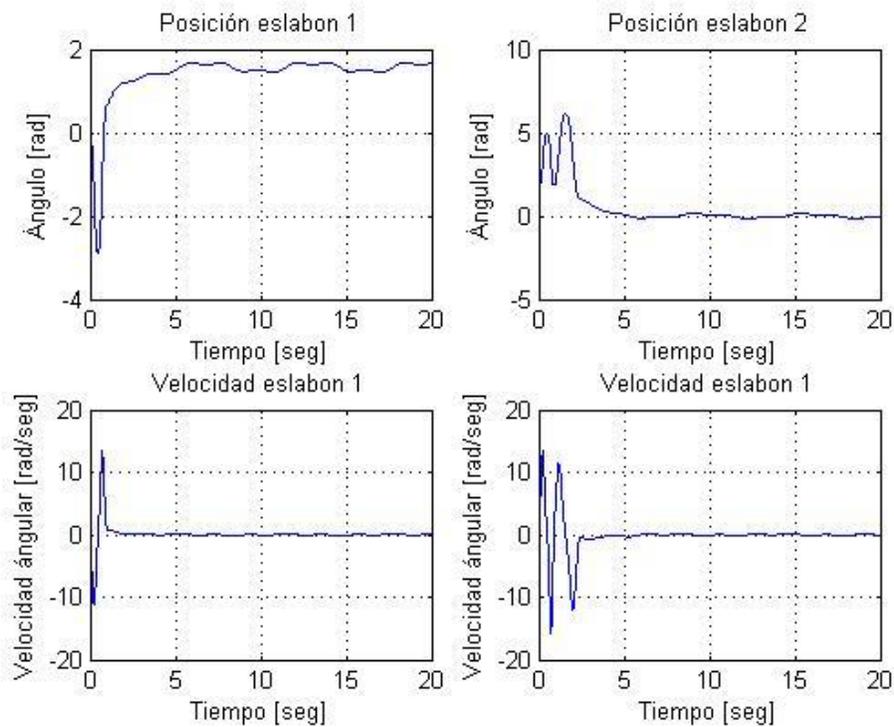


Fig. 5.21: Movimientos y velocidades de las articulaciones del robot con un controlador tipo PD difuso.

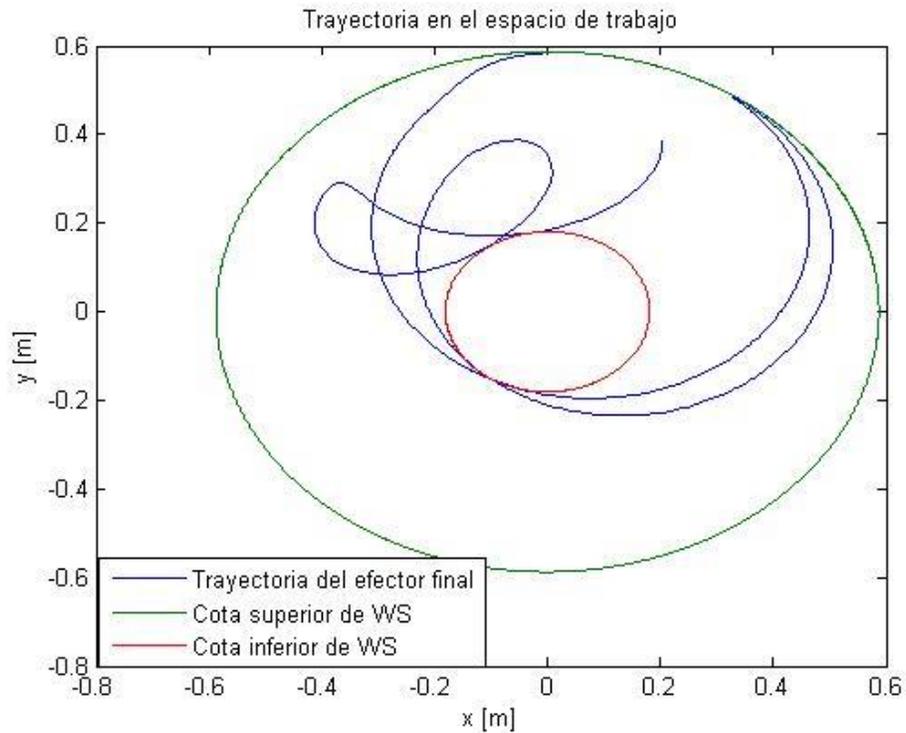


Fig. 5.22: Movimientos en el espacio de trabajo del robot con un controlador tipo PD difuso.

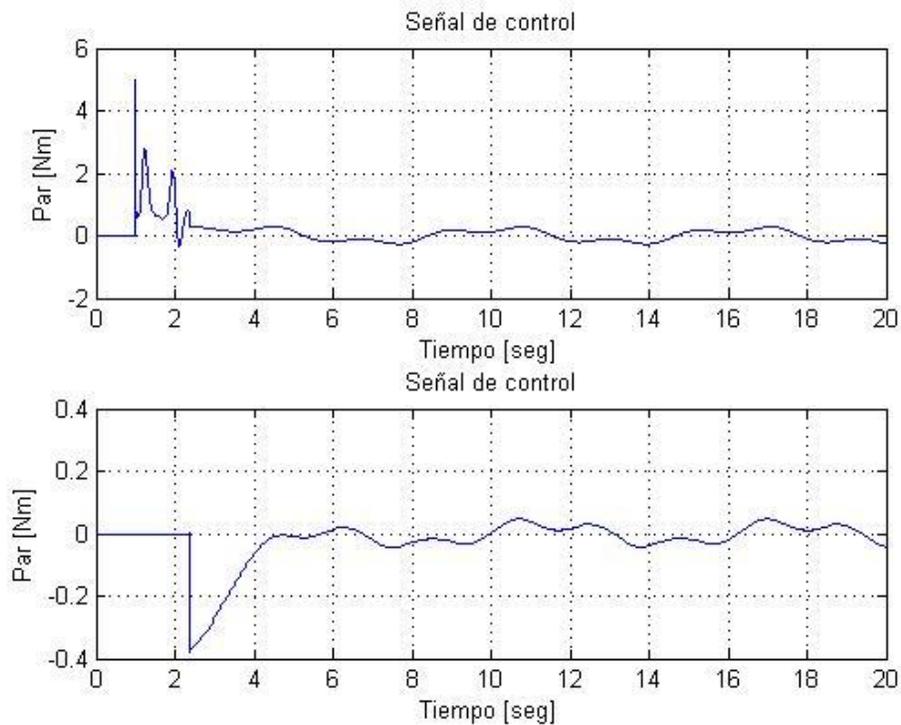


Fig. 5.23: Entradas de control o par generado por el controlador PID.

Como se puede observar el control es un tipo PD, y como se puede apreciar en las figuras 5.21, 5.22 y 5.23 el control hace que el sistema se comporte de forma similar al control PID definido anteriormente (5.16, 5.17 y 5.18). Entonces lo siguiente es verificar el comportamiento de estos sistemas cuya referencia deseada sea la trayectoria propuesta en el capítulo anterior.

5.3 Seguimiento de trayectorias del robot 2GDL mediante Lógica difusa.

Con base a lo anterior y haciendo referencia al capítulo 4, se propone que el sistema siga una referencia, entonces seguimos con proponer el control PID desarrollado en la sección anterior, en este caso se propone que siga una trayectoria, de tal forma que el sistema arroja los siguientes resultados en simulación numérica:

Tiempo de simulación: 20 segundos.

Tipo de integrador: ODE 45

Posición deseada:

$$q_{1d} = \pi / 2,$$

$$q_{2d} = 0,$$

$$\dot{q}_{1d} = 0,$$

$$\dot{q}_{2d} = 0,$$

Paso de integración: 0.1 milisegundos.

Ganancias de Control: $K_1=5$, $K_2=4$, $K_i=2$.

Condiciones iniciales:

$$q_1(0) = 0,$$

$$\dot{q}_1(0) = 0,$$

$$q_2(0) = \pi / 2,$$

$$\dot{q}_2(0) = 0.$$

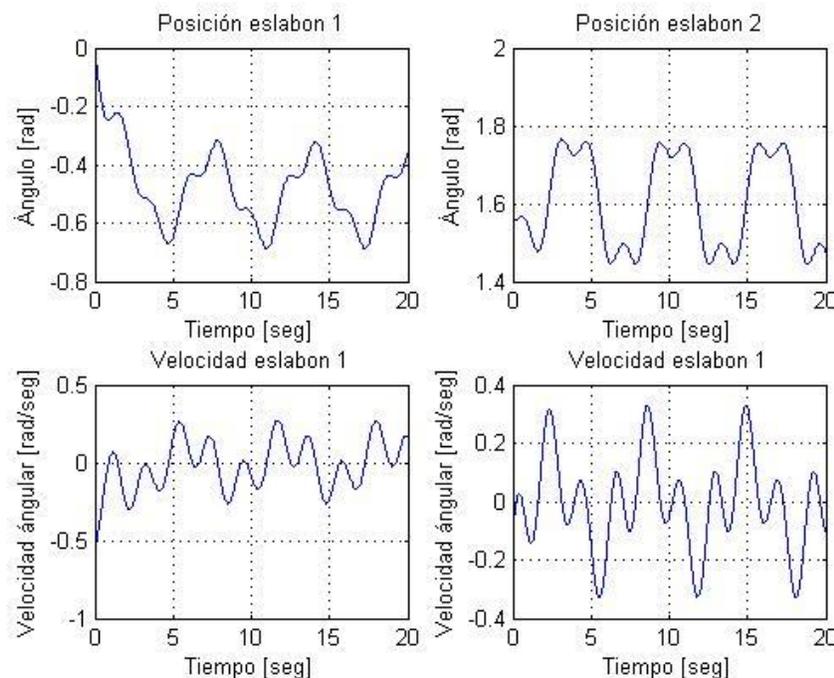


Fig. 5.24: Movimientos y velocidades de las articulaciones del robot con un controlador tipo PID difuso en seguimiento de trayectorias.

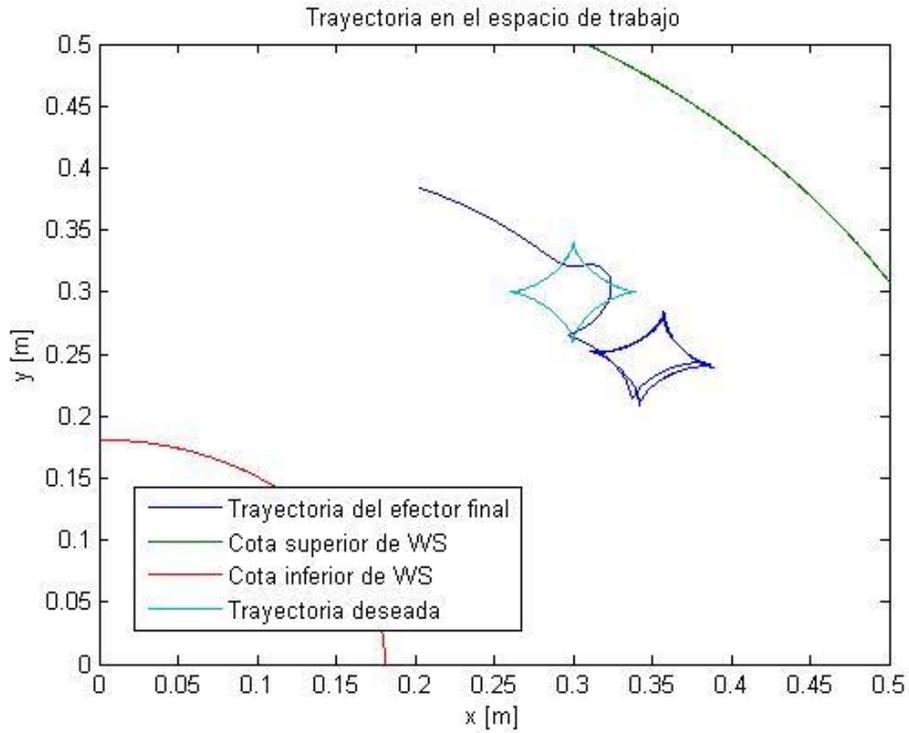


Fig. 5.25: Movimientos en el espacio de trabajo del robot con un controlador tipo PID difuso para el seguimiento de trayectorias.

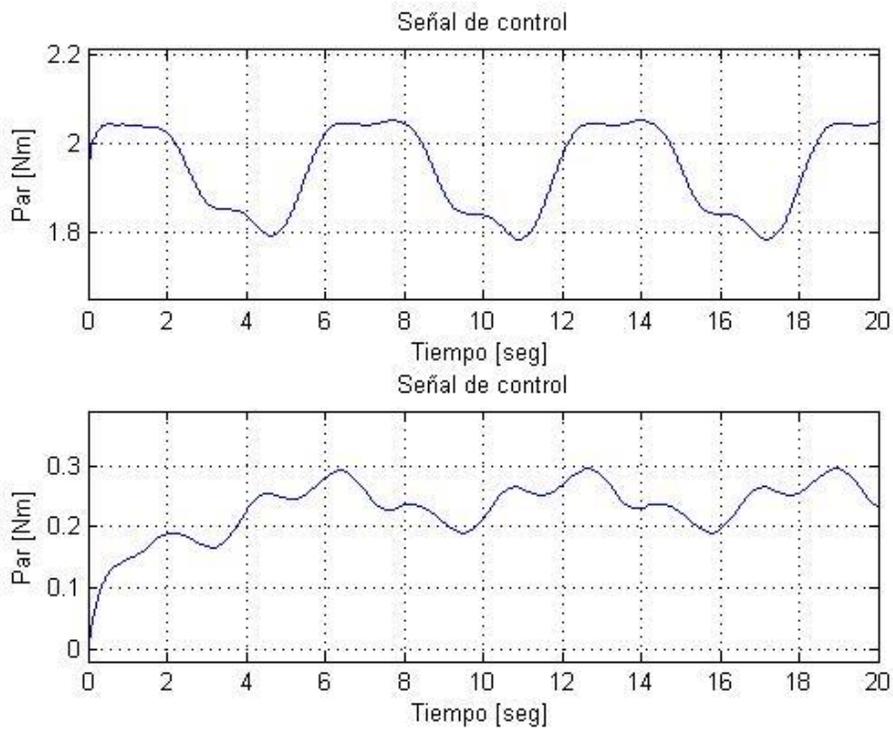


Fig. 5.26: Entradas de control o par generado por el controlador PID para el seguimiento de trayectorias.

Como se puede observar en las figuras 5.24, 5.25 y 5.26, este controlador no sigue la referencia de forma efectiva, por lo cual descartamos al controlador PID con la primera base de reglas como un buen control. Entonces seguimos con la siguiente estructura de control que es el controlador PD difuso con 25 reglas, el descrito en las figuras 5.21, 5.22 y 5.23, para el caso de seguimiento de trayectorias, obteniendo lo siguiente:

Tiempo de simulación: 20 segundos.

Tipo de integrador: ODE 45

Posición deseada:

$$q_{1d} = \pi / 2,$$

$$q_{2d} = 0,$$

$$\dot{q}_{1d} = 0,$$

$$\dot{q}_{2d} = 0,$$

Paso de integración: 0.1 milisegundos.

Ganancias de Control: $K_1=5$, $K_2=4$.

Condiciones iniciales:

$$q_1(0) = 0,$$

$$\dot{q}_1(0) = 0,$$

$$q_2(0) = \pi / 2,$$

$$\dot{q}_2(0) = 0.$$

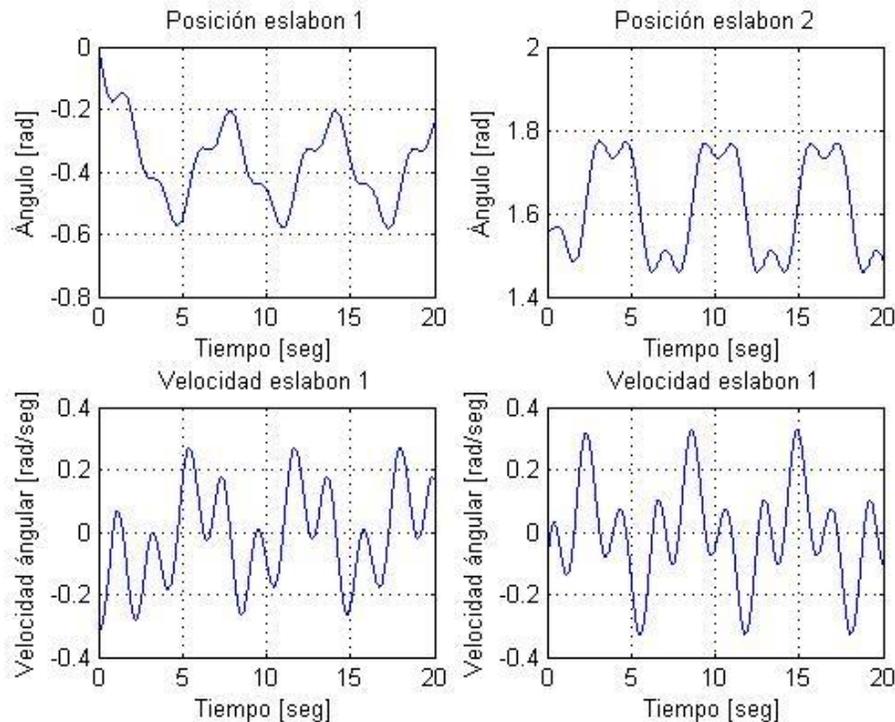


Fig. 5.27: Movimientos y velocidades de las articulaciones del robot con un controlador tipo PD difuso en seguimiento de trayectorias.

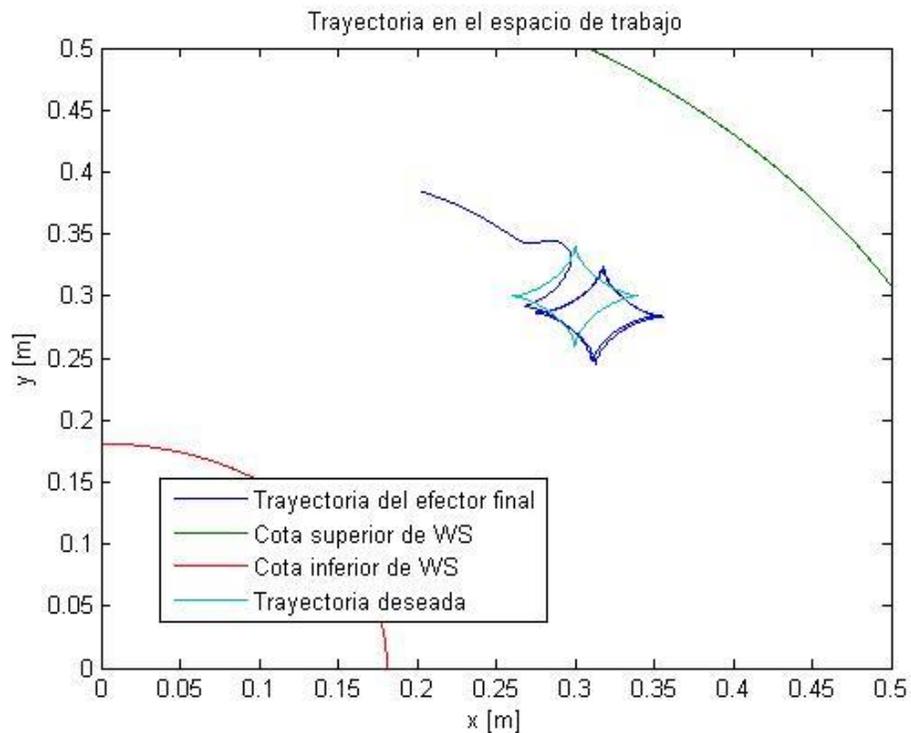


Fig. 5.28: Movimientos en el espacio de trabajo del robot con un controlador tipo PD difuso para el seguimiento de trayectorias.

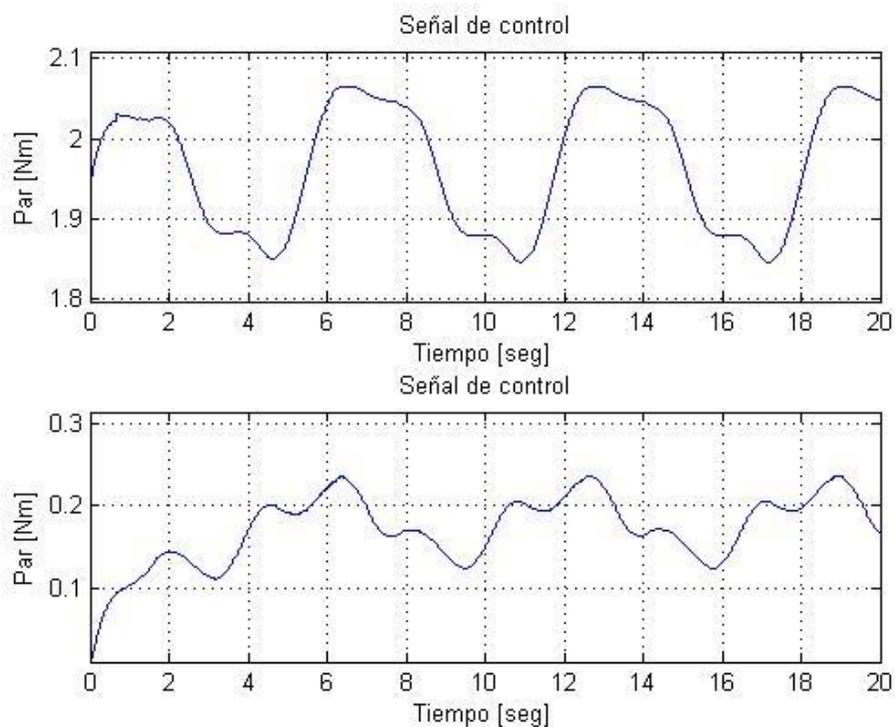


Fig. 5.29: Entradas de control o par generado por el controlador PD para el seguimiento de trayectorias.

En la figura 5.28 se puede observar claramente que el controlador tiene un comportamiento mucho mejor que el controlador PID, sin embargo existe un error muy grande por lo que se propone que a esta base de reglas se le añada la parte integral como sucedió en la sección anterior, por lo tanto se proponen nuevas funciones de salida, las cuales representen la acción integral, obteniendo lo siguiente:

Para la obtención de la forma lineal, para una entrada, una salida, las entradas son determinadas de la siguiente forma:

Función de membresía	Primer parámetro	Segundo parámetro	Tercer parámetro
Negativa	-4.16	-3.14	-1.52
Poco Negativa	-3.14	-1.52	0
Cero	-1.52	0	1.52
Poco Positiva	0	1.52	3.14
Positiva	1.52	3.14	4.16

Tabla. 5.10: Parámetros de las funciones de membresía de salida para nuevas reglas.

De la misma manera se definen las funciones de membresía para la salida de la siguiente forma:

Función de membresía	Negativa	Poco Negativa	Cero	Poco Positiva	Positiva
salida	$f(x) = -3x - 9$	$f(x) = -x - 3$	$f(x) = x^3$	$f(x) = x + 3$	$f(x) = x + 9$

Al integrar estas nuevas salidas de error sobre las dinámicas de control y sumarlas al controlador PD anterior y haciendo uso de esta entrada con la posición para el control tipo PI, entonces el controlador se aproxima a un controlador del tipo PID, si a este control se le hace una asignación de ganancia de forma unitaria, entonces se obtienen los siguientes resultados en simulación numérica:

Tiempo de simulación: 20 segundos.

Tipo de integrador: ODE 45

Posición deseada:

$$q_{1d} = \pi / 2,$$

$$q_{2d} = 0,$$

$$\dot{q}_{1d} = 0,$$

$$\dot{q}_{2d} = 0,$$

Paso de integración: 0.1 milisegundos.

Ganancias de Control: $K_1=1, K_2=1, K_i=1$.

Condiciones iniciales:

$$q_1(0) = 0,$$

$$\dot{q}_1(0) = 0,$$

$$q_2(0) = \pi / 2,$$

$$\dot{q}_2(0) = 0.$$

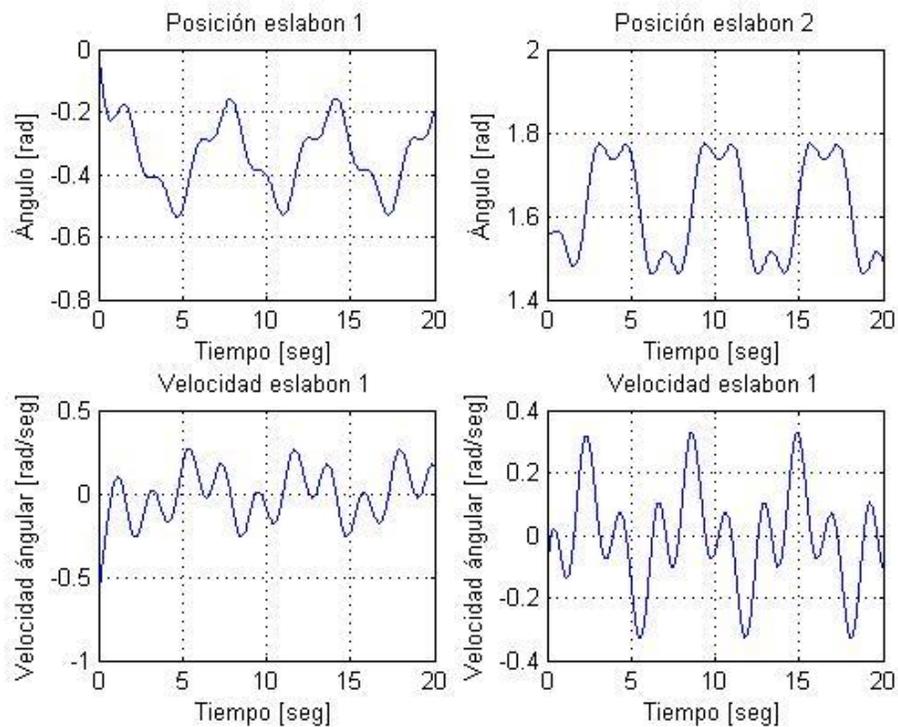


Fig. 5.30: Movimientos y velocidades de las articulaciones del robot con un controlador tipo PD difuso en seguimiento de trayectorias.

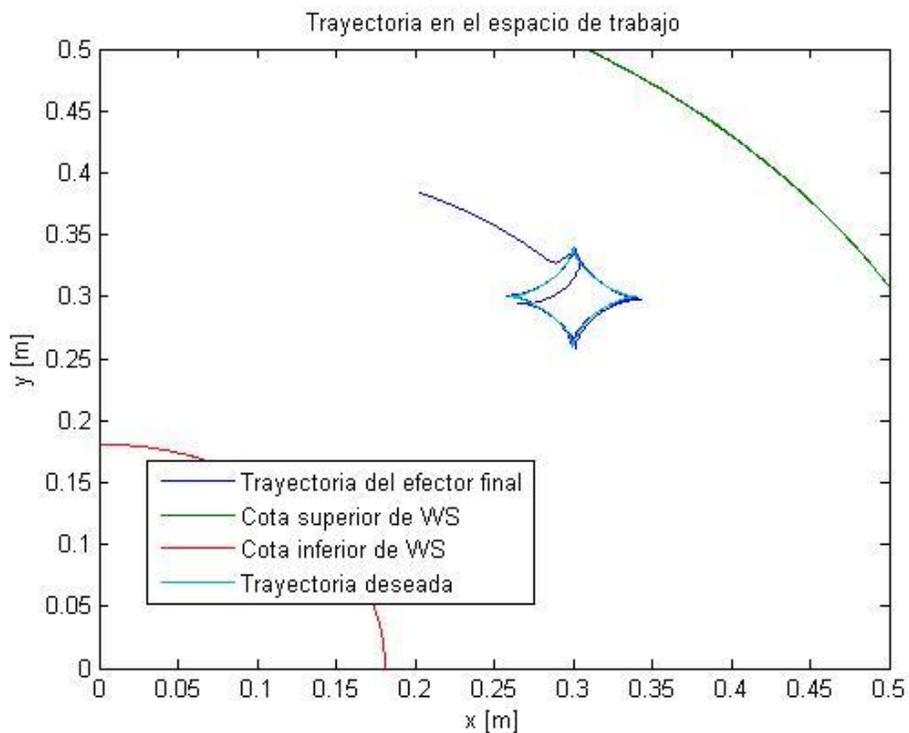


Fig. 5.31: Movimientos en el espacio de trabajo del robot con un controlador tipo PD difuso para el seguimiento de trayectorias.

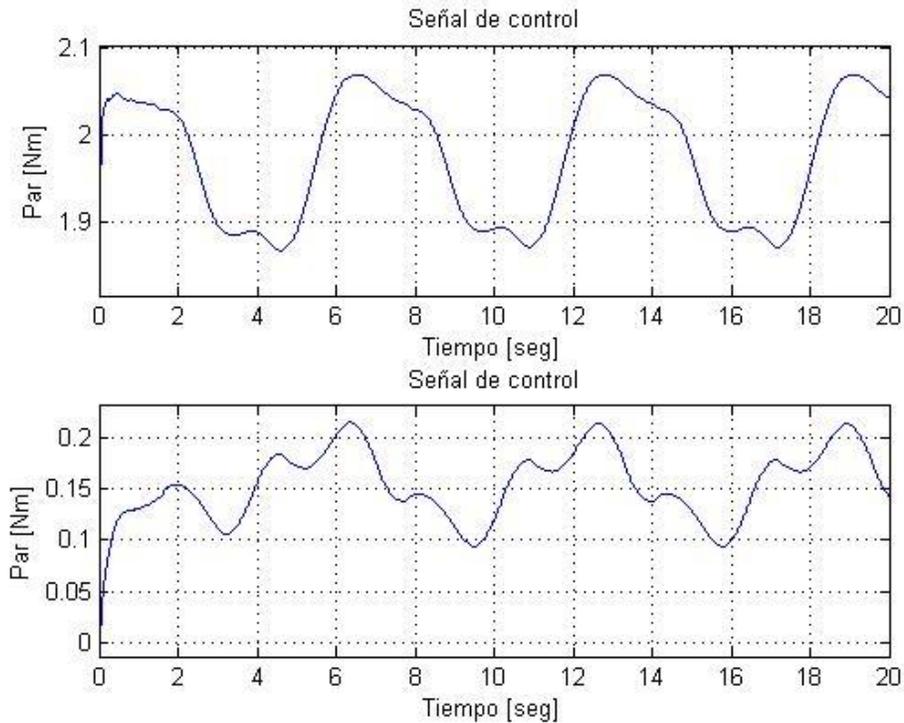


Fig. 5.32: Entradas de control o par generado por el controlador PD para el seguimiento de trayectorias.

Como se puede ver este control se comporta como el PD con ganancias, además el error que se presenta es mucho menor que en el PD (ver figuras 28 y 31), entonces al agregar ganancias a este control obteniendo lo siguiente:

Tiempo de simulación: 20 segundos.

Tipo de integrador: ODE 45

Posición deseada:

$$q_{1d} = \pi / 2,$$

$$q_{2d} = 0,$$

$$\dot{q}_{1d} = 0,$$

$$\dot{q}_{2d} = 0,$$

Paso de integración: 0.1 milisegundos.

Ganancias de Control: $K_1=3$, $K_2=4$, $K_i=3$.

Condiciones iniciales:

$$q_1(0) = 0,$$

$$\dot{q}_1(0) = 0,$$

$$q_2(0) = \pi / 2,$$

$$\dot{q}_2(0) = 0.$$

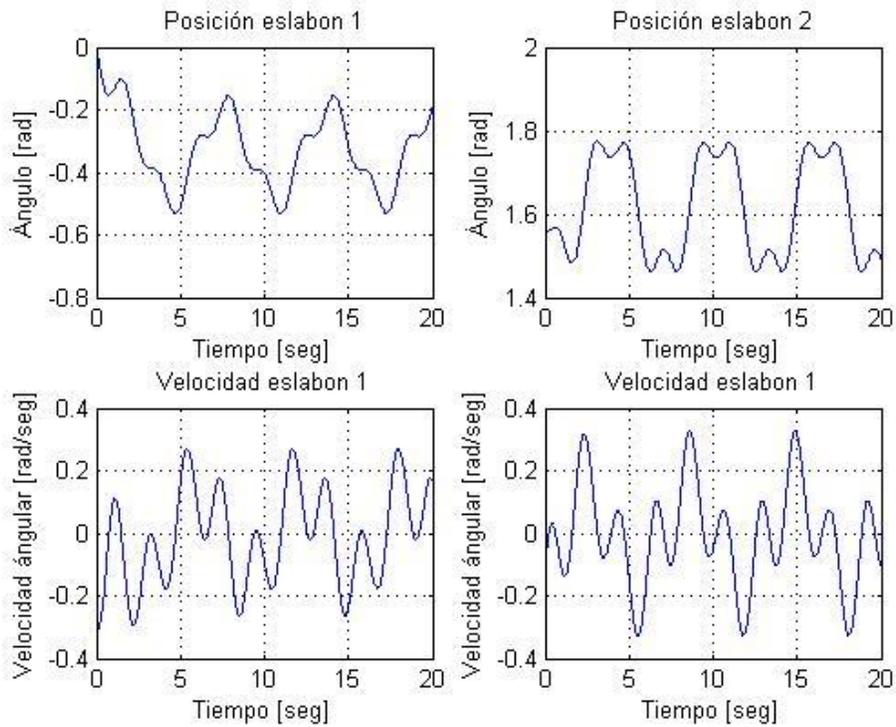


Fig. 5.33: Movimientos y velocidades de las articulaciones del robot con un controlador tipo PD difuso en seguimiento de trayectorias.

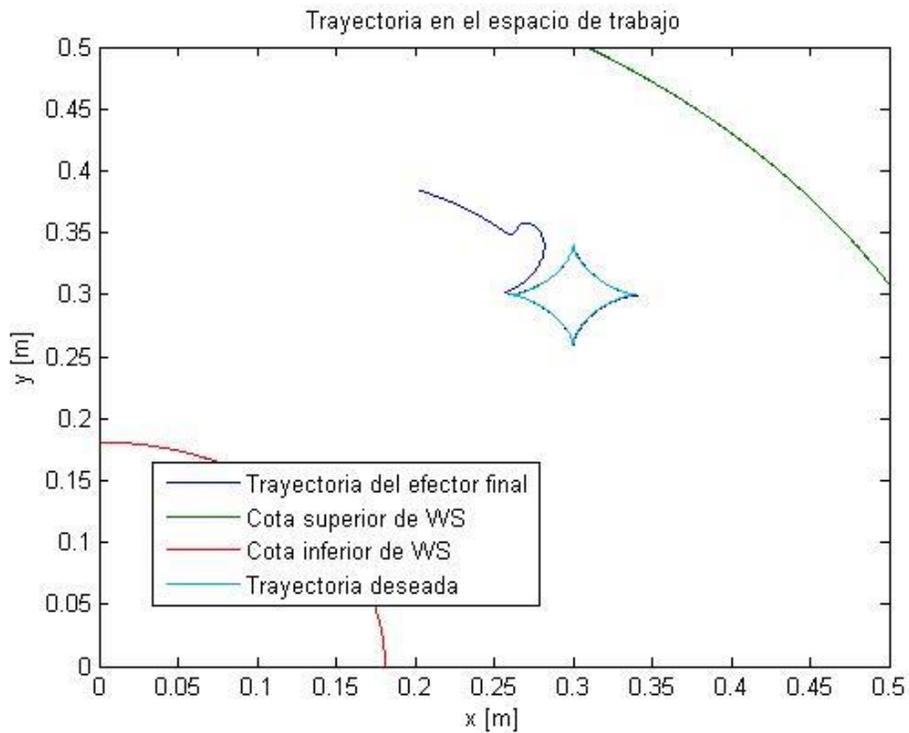


Fig. 5.34: Movimientos en el espacio de trabajo del robot con un controlador tipo PD difuso para el seguimiento de trayectorias.

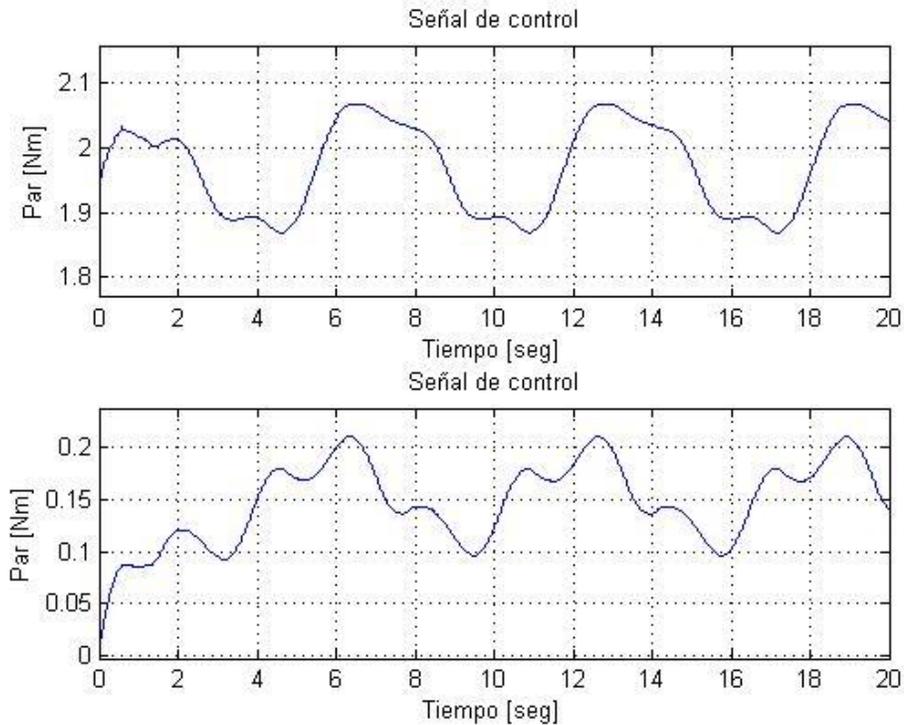


Fig. 5.35: Entradas de control o par generado por el controlador PD para el seguimiento de trayectorias.

Como se puede observar este controlador tiene un mejor desempeño que cualquiera de los controladores diseñados en esta sección, ya que como se puede ver en la figura 3.34, la posición real del sistema es la misma que la deseada, además que el sistema alcanza la referencia en un tiempo no menor que un segundo, mientras que la señal de control (figura 5.35) no demanda un torque excesivo haciendo de este control una muy buena opción para el seguimiento de trayectorias.

5.4 Conclusiones.

El diseño de un controlador difuso, como se puede ver en el transcurso de este capítulo, es muy sencillo y como es un controlador que requiere del conocimiento de un experto. Como se observa en la evolución de este control aplicado sobre regulación a un punto este es mejorado con la adquisición de más reglas, de forma similar que en el seguimiento de trayectorias, sin embargo en este caso se proporcionaron ganancias sobre las dinámicas de error (para el error proporcional, integral y derivativo), haciendo de este el mejor control que se presentó en el capítulo.

Conclusiones

Conclusiones generales.

En esta tesis se presenta el cómo elaborar de controladores clásicos y difusos para un sistema dinámico, como se puede apreciar en el capítulo 3 para una interpretación más real de las dinámicas al modelo se le inyecta fricción, se puede observar que los controladores propuestos en el capítulo 5 son buenos para el caso de regulación punto a punto, sin embargo no son lo suficientemente fiables para el seguimiento de trayectorias, sin embargo en el capítulo 6 se puede apreciar que con un controlador diseñado por un experto (en este caso por experiencias del comportamiento del error) son solo se puede hacer un controlador con una buena aproximación al seguimiento de trayectorias, sino que este controlador resulta ser muy práctico y tiene la ventaja de los controladores clásicos que pueden ser aplicados a cualquier tipo de sistema (obviamente que el sistema sea

completamente actuado, ya que en otro caso se deberán modificar la base de reglas, así como los universos de discurso del mismo).

Como estudio comparativo se puede decir que los controladores difusos son mucho más fáciles de implementar para el caso de seguimiento de trayectorias, ya que no se requiere del conocimiento del modelo matemático para el control que es el caso del controlador por compensación de gravedad.

Sin embargo para el caso de regulación punto a punto se puede decir que el control clásico es mucho más fácil de implementar, ya que no se requiere del conocimiento de un experto para el diseño de control, el control es posible con un controlador PD y con una simple asignación de ganancias.

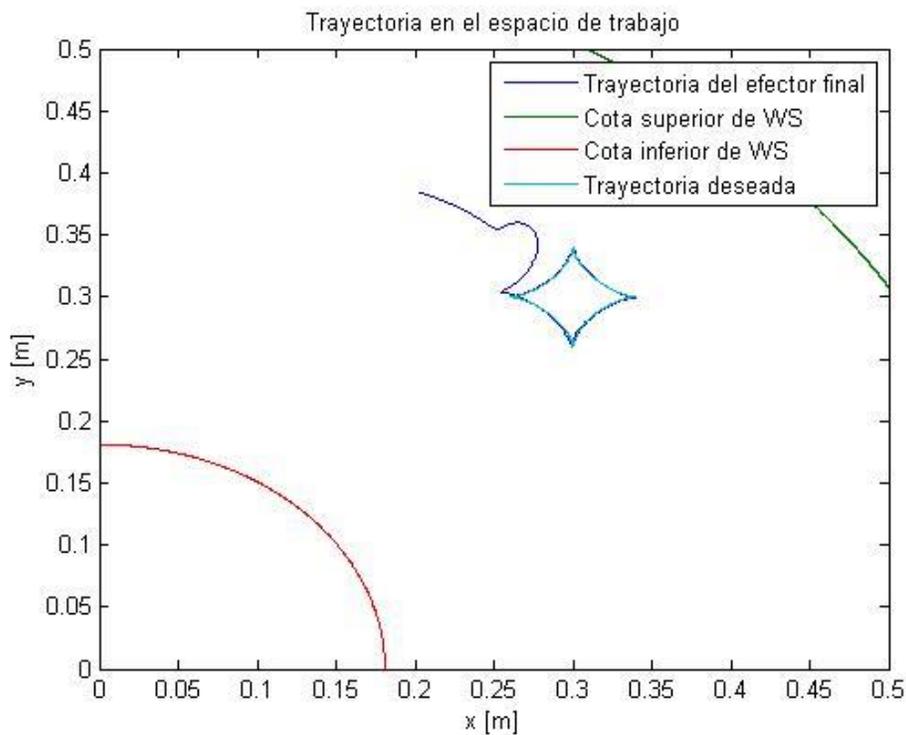


Fig. 6.1: Movimientos en el espacio de trabajo del robot con un controlador PD con compensación de gravedad.

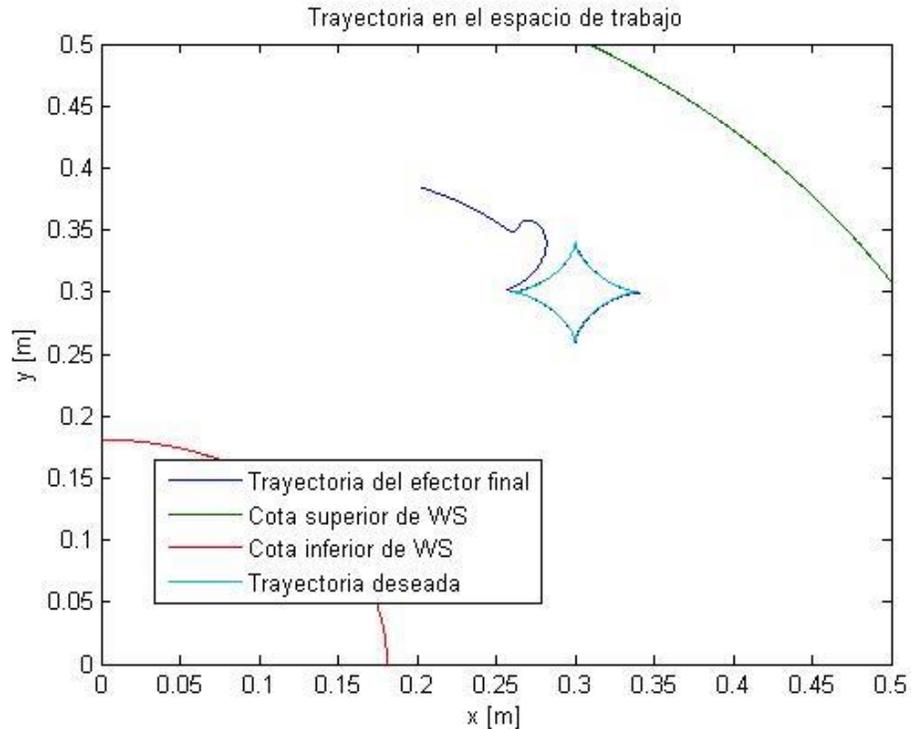


Fig. 6.2: Movimientos en el espacio de trabajo del robot con un controlador tipo PD difuso para el seguimiento de trayectorias.

Lo mencionado anteriormente, puede ser visto en las figuras 6.1 y 6.2 en las que se puede ver el desempeño del robot sobre el espacio de trabajo, donde en la primera se muestra el control relacionado al PD más compensación de gravedad y el segundo el PID difuso. En las mismas también se puede observar que el controlador PID difuso tiene un mejor desempeño al momento de reproducir la trayectoria, en consecuencia se puede concluir que el diseño de controladores mediante lógica difusa es más fácil (en términos de generales) y mejor en su desempeño.

Trabajos futuros.

Como trabajos futuros se pretende realizar un controlador que trate de minimizar los esfuerzos electromecánicos que pueden producirse cuando el efector final del manipulador está en los límites del espacio de trabajo, para esto se pretende crear una base de reglas que mediante una transformación de las coordenadas generalizadas, estas se comporten como las coordenadas cartesianas.

Glosario

- Actuador:** Es el dispositivo que proporciona la fuerza motriz real para las articulaciones de un robot. El actuador suele obtener su energía a partir de una de estas tres fuentes: aire comprimido, fluido por presión o electricidad; por lo que reciben el nombre de actuadores neumáticos, hidráulicos o eléctricos.
Los actuadores tales como los relés, solenoides y motores, posibilitan que la computadora, controle la operación de los sistemas del vehículo.
- Articulación:** Una articulación es la conexión que existe entre dos o más eslabones, la cual se encuentra en los nodos de los eslabones y permite algún movimiento o movimiento potencial, entre los eslabones conectados. Cada articulación provee al robot de al menos un “grado de libertad”. En otras palabras, las articulaciones permiten al manipulador realizar movimientos.
- Centro de masa:** Las fuerzas se distribuyen sobre una línea, un área o un volumen. Por lo común, no es muy difícil encontrar una resultante de estas fuerzas distribuidas. Para tener el mismo efecto esta resultante debe actuar sobre el centroide del cuerpo, el cual es un punto en el que se puede considerar que un sistema de fuerzas distribuidas está concentrado, con el mismo efecto que la fuerza distribuida.
- Dinámica:** Se denomina dinámica a la parte de la mecánica que estudia conjuntamente el movimiento y las fuerzas que lo originan.
- Efecto de Coriolis:** Efecto debido al movimiento rotacional de la tierra, que se manifiesta en todo cuerpo en movimiento, de tal forma que lo desvía de su trayectoria recta. En el hemisferio norte la desviación ocurre hacia la derecha de la dirección del cuerpo y mientras que en el hemisferio sur la desviación es hacia la izquierda.
- Energía:** Capacidad de realizar un trabajo.
- Energía Cinética:** Es la energía que posee un cuerpo por razón de su movimiento.

-
- Energía potencial:** La que posee un cuerpo por el hecho de hallarse en un campo de fuerzas, se dice que está en reposo.
- Eslabón:** El eslabón es un cuerpo rígido que posee por lo menos dos nodos, que son los puntos de unión con otros eslabones. Los eslabones se pueden nombrar de acuerdo al número de nodos que poseen, por ejemplo: Eslabón binario: Es el que contiene dos nodos. Eslabón ternario: Es el que contiene tres nodos. Eslabón cuaternario: Es el que contiene cuatro nodos.
- Fricción:** La fricción aparece cuando dos superficies se frotan una en contra de la otra presentando desprendimiento de calor.
- Fuerza de Coriolis:** Efecto debido al movimiento rotacional de la tierra, que se manifiesta en todo cuerpo en movimiento, de tal forma que lo desvía de su trayectoria recta.
- Fuerza de fricción:** Fuerza aparente hacia el exterior experimentada por un cuerpo que gira alrededor de su eje.
- Fuerza de gravedad:** Fuerza de atracción entre dos masas. La magnitud de esta fuerza es directamente proporcional al producto de ambas masas e inversamente proporcional al cuadrado de la distancia que las separa.
- Grado de libertad:** Son los posibles movimientos básicos independientes, ya sean giratorios o de desplazamiento, que el robot puede realizar.
- Inercia:** Propiedad de un cuerpo que tiende a oponerse a toda variación en su estado de reposo o de movimiento.
- Longitud:** Dimensión que expresa el valor de una distancia.
- Manipulador:** Es un mecanismo que usualmente consiste en una serie de eslabones articulados o desplazamiento relativo a otro eslabón, para agarrar y mover objetos, por lo regular tiene varios grados de libertad.
- Matriz:** Conjunto de $m \times n$ números distribuidos en m *filas* y n *columnas*. Cada número se designa con dos índices: el primero señala el número de filas contadas de arriba a abajo, y el segundo de la columna contando de izquierda a derecha.

-
- Mecánica:** Es la rama de la Física y de la ingeniería que se ocupa de las relaciones mutuas entre fuerza, materia y movimiento.
- Modelo matemático:** Es simplemente una representación matemática de sistemas del mundo real. Este modelo se desarrolla a través de la aplicación, a los elementos de un sistema, de las reglas conocidas del comportamiento.
- Radianes:** Unidades angulares que corresponden a un arco de longitud igual a su radio.
- Robot:** Es un dispositivo mecánico que realiza acciones basadas en movimientos. Sus acciones más comunes son moverse autónómicamente entre otras manipulaciones demasiado precisas, pesadas, repetitivas o riesgosas para el humano.
- Robot manipulador:** Son esencialmente brazos articulados. De forma más precisa, un manipulador convencional es una cadena cinemática abierta formada por un conjunto de eslabones o elementos de la cadena interrelacionados mediante articulaciones o pares cinemáticos.
- Servomotor:** Aparato mecánico gobernado por la rueda del timón y que a su vez acciona la caña del timón en los buques grandes.
- Simulación digital:** Representación del comportamiento de algo, expresado en magnitudes numéricas mediante el computador.

Bibliografía

- [1] Hung V. Vu Ramin S. Esfandiari “Dinamic Systems, Modeling and Análisis” Mc Graw Hill 1997, ISBN-13:978-0-07-296661-9.
- [2] Benjamín C. Kuo “Sistemas de Control Digital” CECSA, 2000 ISBN 968-26-1292-6.
- [3] Benjamín C. Kuo “Sistemas de Control Automático” Séptima edición, Pearson Educación, 1996, ISBN 968-880-723-0.
- [4] Jesús Patricio Ordaz Oliver “Control basado en energía y estabilización para sistemas subactuados” M. En C. Tesis, Universidad Autónoma del Estado de Hidalgo, Pachuca de Soto, Hidalgo, 2007.
- [5] Javier García de Jalón, José Ignacio Rodríguez, Jesús Vidal “Aprenda Matlab 7.0 Como si estuviera en primero” Madrid, Diciembre 2005, Escuela Técnica Superior de Ingenieros Industriales.
- [6] Lofti A. Zadeh. Fuzzy logic, neural networks and soft computing. One-page course announcement of CS 294-4, Spring 1993, the Univerity of califrnia at Berkeley Nobember 1992.
- [7] <http://www.mathworks.com>
- [8] <http://www.mathworks.co.uk/company/events/webinars/wbnr30380.html?id=30380&p1=40588&p2=40588>
- [9] <http://www.mathworks.co.uk/products/virtualreality/demos.html>
- [10] Neuro-Fuzzy and Soft Computing, Jyh-Shin Roger Jang, Chuen-Tsai Sun, Eiji Mizutani, Prentice Hall 1997.
- [11] Introduction to fuzzy logic using Matlab, S. N. Sivanandam, S. Sumathi, S. N. Deepa, Spinger-Verlang 2007.