

UNIVERSIDAD NACIONAL DEL CALLAO
ESCUELA DE POSGRADO

SECCIÓN DE POSGRADO DE LA FACULTAD DE
INGENIERÍA ELÉCTRICA Y ELECTRÓNICA



“DISEÑO DEL SISTEMA DE CONTROL POR
LÓGICA DIFUSA APLICADO AL BRAZO
ROBÓTICO TRANSLACIONAL DE 2DOF”

TESIS PARA OPTAR EL GRADO ACADÉMICO DE MAESTRO EN
CIENCIAS DE LA ELECTRÓNICA
MENCIÓN: CONTROL Y AUTOMATIZACIÓN

AUTOR: JACOB ASTOCONDOR VILLAR

CALLAO – PERU

2011

A handwritten signature in black ink, appearing to be "Jacob Astocondor Villar".

A second handwritten signature in black ink, appearing to be "Jacob Astocondor Villar".

JURADO SUSTENTACIÓN DE TESIS DE MAESTRÍA

DR. JUAN HERBER GRADOS GAMARRA	Presidente
DR. MARCELO NEMESIO DAMAS NIÑO	Secretario
MG. FRANCO IVÁN VÉLIZ LIZÁRRAGA	Miembro
DR. CIRO ITALO TERÁN DIANDERAS	Miembro
MG. NICANOR RAÚL BENITES SARAVIA	Asesor

Nº DE LIBRO: 01

Nº DE ACTA: 13

FECHA: julio 20, 2011.

RESOLUCIÓN DE LA SECCIÓN POSGRADO DE LA FIEE Nº 049-2011-DSPG-FIEE

DEDICATORIA

A Dios que me ha dado entendimiento y sabiduría, a mi amada esposa Maruja + que es mi ayuda idónea y que me ha sido de gran apoyo para alcanzar las metas propuestas, a mis hijos Lizbeth y Jacob Jr. Que es la que me inspira seguir adelante, a mis padres Liberato + y Fortunata por el amor derramado en mi.

AGRADECIMIENTO

Quisiera expresar mi agradecimiento el apoyo total que recibí de mis profesores y personal administrativo de la Sección de Postgrado de la FIEE-UNAC, y en particular a mi asesor al MSc. Nicanor Raúl Benites Saravia por sus enseñanzas y sugerencias para este trabajo de tesis. A todo el personal de la Sección de Postgrado de la FIEE UNAC por darme el apoyo durante mi formación académica y un especial agradecimiento al Msc Franco Veliz Lizárraga , al Dr. Ing. Cesar Augusto Rodríguez Aburto, al Dr. Juan Grados Gamarra al Dr. Marcelo Nemesio Damas Niño por darme todas las facilidades y apoyo para hacer posible esta tesis.

ÍNDICE

CARATULA	I
HOJA DE REFERENCIA DEL JURADO	II
DEDICATORIA	III
AGRADECIMIENTO	IV
ÍNDICE	01
PRÓLOGO	04
RESUMEN	05
ABSTRACT	06

Capítulo I

PLANTEAMIENTO INICIAL DE LA INVESTIGACIÓN	07
1.1 Identificación del problema	07
1.2 Formulación del problema	08
1.3 Objetivos de la investigación	09
1.3.1 Objetivo general	09
1.3.2 Objetivo Especifico	09
1.4 Justificación	09
1.5 Limitaciones y Facilidades	10
1.6 Hipótesis de partida	11

Capítulo II

MARCO TEÓRICO	12
2.1 Antecedentes del estudio	12
2.2 Bases teóricas	14
2.2.1 Introducción	14
2.2.2 Conjuntos Difusos	15
2.2.3 Universo	20
2.2.4 Dominio	20
2.2.5 Funciones de pertenencia ó membresía	20
2.2.6 Controlador lógico difuso	22
2.3 Glosario de términos	28

Capítulo III	
METODOLOGÍA	30
3.1 Relación entre las variables de la investigación	30
3.2 Tipo de investigación	31
3.3 Diseño de la investigación	32
3.3.1 Modelado de la Planta	32
3.3.2 Requerimiento de diseño del controlador del Brazo Robótico Translacional g de 2DOF	43
3.3.3 Diseño del control difuso en tiempo continuo	48
3.3.4 Diseño del control difuso en tiempo discreto	64
3.4 Metodica de cada momento de la investigación	70
3.5 Operacionalización de variables	71
3.6 Población y muestra	71
3.7 Técnica e instrumentos de recolección de datos	71
3.8 Procedimientos de recolección de datos	71
3.9 Procedimiento estadístico y análisis de datos	71
Capítulo IV	
RESULTADOS	72
4.1 Resultados parciales	72
4.1.1 Resultados del sistema de control difuso en tiempo continuo en lazo cerrado	72
4.1.2 Resultados del sistema de control difuso en tiempo discreto en lazo cerrado	74
4.1.3 Señales de los sensores para desplazar linealmente y angular del Brazo Robótico	75
4.2 Resultados Finales	77
Capítulo V	
DISCUSIÓN DE RESULTADOS	80
5.1 Constatación de hipótesis con los resultados	80
5.2 Contratación de resultados con otros estudios similares	80
5.2.1 Experimento	81

CONCLUSIONES	84
RECOMENDACIONES	84
REFERENCIALES	87
ANEXOS	
Anexo A	
Matriz de Consistencia	89
Anexo B	
Identificación de Parámetros	90
B.1 Identificación de Parámetros del Servomotor	90
B.2 Curva de la velocidad para la identificación	91
B.3 Coeficiente de Fricción del carro	94
B.4 Coeficiente de Fricción Viscosa del Brazo	96
Anexo C	
Programas en MATLAB	101
C.1 Listado de Programas	101
C.1.1 Simulación de la planta en lazo abierto	101
C.1.2 Simulación del sistema de control en tiempo continuo en lazo cerrado	103
C.1.3 Simulación del sistema de control en tiempo discreto en lazo cerrado	108
C.1.4 Simulación de la discretización de los sensores de posición lineal y angular	114
C.1.5 Simulación del sistema de control en tiempo discreto en lazo cerrado con señal de prueba sensorial	117

PRÓLOGO

El objetivo de este trabajo es mostrar en detalle el proceso de implementación del núcleo principal de un sistema de control difuso de un sistema denominado Brazo Robótico translacional de segundo orden (BRT2DOF). En esta tesis se presentan las bases teóricas generales de los controladores difusos, a continuación se presenta el modelo matemático completo del sistema de los elementos utilizados en el sistema de control y luego se presenta el proceso de diseño considerando los bloques de fuzificación, defuzificación e inferencia basada en reglas, junto con los resultados obtenidos y su simulación correspondiente.

Teniendo en cuenta que, hasta ahora, en el ámbito local se ha producido escasa literatura en el área de controladores difusos, este trabajo es importante ya que constituye un aporte para la construcción de una base de conocimiento más amplia en este campo.

Se presentan guías y referencias puntuales sobre el proceso práctico diseño y simulación de controladores difusos las cuales pueden resultar de utilidad para personas pertenecientes a diferentes entornos académicos o productivos que posean conocimientos sobre el tema y deseen iniciarse o profundizar en el campo de la experimentación asociada con el control difuso.

Este trabajo está enmarcado dentro del área de control, específicamente fue motivado y además está sustentado en los conocimientos adquiridos en las asignaturas: Control Automático, Instrumentación y Control Inteligente, cursadas en la Sección de Postgrado de Control y Automatización, de la Universidad Nacional del Callao, de la Facultad de Ingeniería Eléctrica y Electrónica.

RESUMEN

La presente tesis trata sobre el control de trayectoria de un Brazo Robótico translacional de 2DOF (Two-Degrees-of-Freedoms) que consta de un móvil accionado por una polea y un eslabón articulado en el CG (centro de gravedad) de dicho móvil. Este proceso será controlado mediante la técnica de control fuzzy.

La acción de control está orientada a controlar el movimiento translacional del móvil y el movimiento angular del brazo que es libre de girar en ambas direcciones. Las metas impuestas en la tesis son: Diseño, modelado, simulación en tiempo real del sistema, controlado con la ley de control fuzzy.

Los sistemas convencionales de control son diseñados tradicionalmente usando modelos matemáticos de sistemas físicos para posteriormente aplicar técnicas de diseño para obtener controladores apropiados para el sistema. Sin embargo, en la realidad, el modelo y sus parámetros son con frecuencia desconocidos. Esto se debe a cambios en el ambiente de trabajo, dinámicas no modeladas y la presencia de no linealidades e incertidumbres que son difícil de tratar con los controladores convencionales los cuales no siempre son capaces de aprender o de adaptarse a nuevas situaciones a continuación se plantea este problema y la solución del mismo utilizando la técnica de lógica difusa.

Actualmente las técnicas avanzadas de control basadas en modelos, como son el control predictivo, el control por modos deslizantes, el control adaptable, entre otras, están siendo combinadas con técnicas de control basadas en los sistemas difusos

ABSTRACT

This thesis deals with the trajectory control of a 2DOF arm translational Robotic (Two-Degrees-of-Freedoms) consisting of a cell phone powered by a LINK pulley and articulated in the CG (center of gravity) said Cell This process controlled by techniques will fuzzy control.

The control action oriented this translational motion control and movement cell angle of the arm that is free to rotate in both directions. The goals involved in the thesis are: design, modeling, real-time simulation system, controlled by the fuzzy control law.

Conventional systems control are traditionally designed using models mathematical physical systems, to apply design techniques appropriate for drivers system. However, in reality, the model and its parameters are often unknown. This is because changes in the work environment, and unmodeled dynamics presence of nonlinearities and uncertainties are difficult of treated with conventional controllers which do not always are able to learn or adapt to new situations Below this is a problem and the solution of the using fuzzy logic.

Techniques currently advanced model based control, such as predictive control, the sliding mode control, adaptive control, among others, are being combined with techniques of control based on fuzzy systems

Capítulo I

I.-PLANTEAMIENTO INICIAL DE LA INVESTIGACIÓN

1.1. Identificación del problema

El sistema Brazo Robótico Translacional de 2DOF (**BRT2DOF**) es un sistema no lineal lo cual con lleva a problemas en la modelación matemática y fundamentalmente en el control que de este se quiere hacer.

Para lograr obtener un buen control del sistema Brazo Robótico Translacional de 2DOF (**BRT2DOF**) se aplicara la estrategia de control difuso e identificación de sistemas.

El problema que se estudia en esta tesis es el control simultáneo de la posición angular del brazo y la posición de translación del carro, que es resuelto por medio de la técnica de control por logica difusa.

Para esto se requiere que el manipulador robótico realice el seguimiento de trayectorias de referencia arbitrarias empleando la fuerza de control adecuada. Por consiguiente, el proceso a controlar posee dos entradas y dos salidas, es decir, es un proceso MIMO (Multiple-Input-Multiple-Output).

El Brazo robótico translacional de 2DOF(**BRT2DOF**) , es un proceso multivariable que posee un eslabón (el brazo del robot) articulado al eje del motor DC. Este motor está montado sobre uncarro que desarrolla su movimiento de translación por la acción de dos poleas. Una de las poleas está acoplada al eje de un segundo motor DC. Este último motor va a gobernar el movimiento del carro.

En nuestro caso el control difuso se presenta como una alternativa de control ya que nos permite ajustar automáticamente las variables a controlar: el avance y la velocidad del carro, además del ángulo del brazo o varilla así como su velocidad angular.

Como condición de diseño, se supone que el controlador (regulador) debe ser tan eficiente o confiable que el Brazo (varilla)debe mantenerse en equilibrio (estable), aún cuando se halle sometido a una situación externa, tal como una perturbación debida a la aplicación de una fuerza en el extremo de la varilla o a un cambio en la

dinámica (modelo) del sistema.

Debido a lo particular del sistema **BRT2DOF** denominado Brazo Robótico Translacional es posible plantear algunos problemas interesantes para la teoría de control. Algunos de estos son:

1. El sistema es no lineal.
2. Presenta cuatro variables de estado.
3. El control es acotado
4. La longitud del riel es finita, por lo tanto la posición del carro es acotada.

Fue necesario modelar la dinámica de la planta mediante las ecuaciones de energía de Lagrange. El modelo requiere de la identificación de los parámetros del servomotor DC cuyos resultados experimentales son presentados en el apéndice A.

En definitiva, Teniendo en cuenta todas estas consideraciones , el problema a resolver es controlar la posición lineal y angular del Brazo Robótico Translacional de 2DOF (**BRT2DOF**, a una referencia dada de posición lineal y angular.

1.2. Formulación del problema

El Brazo robótico Translacional de 2DOF (**BRT2DOF** es un sistema complejo de articulaciones que interactúan con una dinámica no lineal. A medida que la velocidad de los manipuladores se incrementa la cantidad de interacciones entre sus articulaciones complican el problema de control.

Las estrategias de control son limitadas cuando se utilizan técnicas de control convencionales siendo confiables cuando presentan un funcionamiento a baja velocidad, dónde la interacción de la dinámica es mínima y el modelo puede ser eficazmente linealizado y desacoplado.

Luego para mejorar la eficiencia y rango de operación del manipulador, se debe de considerar métodos de control más sofisticados, siendo uno de ellos el método de control por lógica difusa.

Ante las exigencias requeridas en el diseño de controladores inteligentes, para un brazo robótico translacional de 2DOF (*BRT2DOF*), es que se formula el problema en forma de pregunta:

¿El diseño del controlador difuso permitirá un óptimo control del brazo robótico translacional de 2DOF?

1.3. Objetivos de la investigación

1.3.1. Objetivo general

Diseñar un controlador difuso para un sistema Brazo Robótico Translacional de 2DOF (*BRT2DOF*).

1.3.2. Objetivo Específico

El objetivo específico del presente trabajo es aplicar estrategias de **Control fuzzy**, para controlar la posición angular θ del brazo y la posición **lineal** del *BRT2DOF* y de esta manera poder comprobar que el controlador diseñado, resuelve el problema de no linealidad del sistema en estudio, así como comprobar que el sistema Brazo Robótico Translacional de 2DOF es estable dentro del rango para el cual fue diseñado.

1.4. Justificación

En la naturaleza existen sistemas físicos como el péndulo invertido del cual parte este trabajo, que no pueden ser operados de forma lineal debido a las características no-lineales que estos presentan, y puesto que las técnicas clásicas de control, como el PID, son diseñadas para controlar sistemas lineales; lo cual implica que si estas técnicas son aplicadas a sistemas de este tipo, el rango de operación del sistema sería muy limitado; por lo que es necesario desarrollar técnicas avanzadas de control, como son los controles inteligentes, y entre los cuales se encuentra el control difuso.

Por lo tanto en este trabajo se diseña un control difuso aplicado a un sistema denominado Brazo robótico Translacional de 2DOF (*BRT2DOF*). Para verificar que el control diseñado funcione satisfactoriamente, se hace uso de la simulación

digital mediante el paquete computacional Matlab y Simulink.

El controlador difuso no sólo compensa el efecto de gravedad, también compensa otros efectos no lineales, como la fricción, etc. Para aplicaciones donde el margen de error es crítico esta contribución de añadir el control difuso a los controladores PID es importante.

1.5. Limitaciones y Facilidades

1.5.1. Limitaciones

Debido a los problemas propios de un sistema real, es posible que se llegue al resultado esperado a partir de los métodos estudiados y aplicados con error permitido en estado estacionario, lo que no desmerita el trabajo, ya que éste incluye importantes aspectos teóricos. Es por esto que se realizarán inicialmente pruebas sobre un sistema simulado.

El trabajo utiliza el sistema *BRT2DOF* en estudio, no se dispone en la FIEE-UNAC un prototipo *BRT2DOF* actualmente se está construyendo un prototipo donde se pueden probar y experimentar diferentes algoritmos de control y programación en tiempo real, que ejemplifican y justifican el desarrollo teórico.

1.5.2. Facilidades

Para el diseño y simulación de los sistemas de control propuestos, se ha utilizado Matlab y Simulink para el cálculo matemático y la validación del modelo del Brazo Robótico Translacional de segundo grado de libertad (*BRT2DOF*), así como para la simulación de la respuesta de los sistemas de control usados en este trabajo, y validar su desempeño.

Existe literatura importante sobre las estrategias de control Fuzzy utilizadas en esta tesis.

1.6. Hipotesis de partida

En función del planteamiento del problema, de las interrogantes planteadas del problema, de los antecedentes técnicos, así como de los objetivos generales y específicos que se persigue, es que se plantea la siguiente hipótesis:

La aplicación de estrategias de control fuzzy permitirán controlar la posición angular del Brazo Robótico θ y la posición lineal del Brazo Robótico r a niveles de referencia deseados y dentro de valores acotados.

Capítulo II

II. MARCO TEÓRICO

2.1. Antecedentes del estudio

El desarrollo de esta lógica tiene sus orígenes a principio de la década de 1920 con los trabajos de lógica multivaluada realizados por Jan Lukasiewicz. En 1937 se publicó el artículo *Philosophy of Science* de Max Black en el cual utiliza la teoría de conjuntos difusos pero él usó el término vaguedad (vagueness).

La lógica difusa fue investigada a mediados de la década de los años sesenta en la Universidad de California en Berkeley por el ingeniero Lofty A. Zadeh cuando se dio cuenta de lo que él llamo principio de incompatibilidad. **«Conforme la complejidad de un sistema aumenta, nuestra capacidad para ser precisos y construir instrucciones sobre su comportamiento disminuye hasta el umbral más allá del cual la precisión y el significado son características excluyentes».**

Y es así que en 1965 L. A. Zadeh, profesor de teoría de sistemas de la Universidad de California en Berkeley publica un artículo llamado conjuntos difusos "Fuzzy Sets"[1]. En el que establece, a partir de la teoría de Jan Lukasiewicz, leyes algebraicas para conjuntos difusos, y además introduce el concepto de conjunto difuso (Fuzzy Set) y pone el nombre de lógica difusa (Fuzzy Logic), bajo el que reside la idea de que los elementos bajo los cuales se construye el pensamiento humano no son nmeros sino etiquetas lingüísticas [2].

Ya que la lógica difusa permite representar el conocimiento común, que es mayoritariamente del tipo lingüístico cualitativo y no necesariamente cuantitativo, en un lenguaje matemático a través de la teoría de conjuntos difusos y funciones características asociadas a ellos, además de que nos permite trabajar de manera simultanea con términos lingüísticos y datos matemáticos; si bien los términos lingüísticos son inherentemente menos precisos que los datos matemáticos, los primeros nos pro-

porcionan información que es más compatible con el modo de razonamiento humano.

A partir de aquí se da origen a innumerables artículos sobre lógica difusa, la cual permitió tener una alternativa programable de control y proceso tecnológico que trata con el tipo de descripciones subjetivas o ambiguas que son comúnmente usadas en el mundo real y que ha encontrado un gran campo de aplicación en el control automático.

Aunque al principio la comunidad científica no aceptó totalmente la teoría de la lógica difusa, Lofti A. Zadeh continuó ampliando y asentando nuevos principios de conjuntos difusos. Podemos mencionar a algunos científicos que realizaron aportaciones para el desarrollo de estas teorías, como lo son Bellman, Lakoff, Sugeno, Smith Goguen, Kohout, Chan, Bezdek, Dunn, Tanaka, Zimmerman y Kandel entre otros.

Es durante este periodo, que abarca en su totalidad la década de los años sesenta, y gracias a las aportaciones de los científicos mencionados anteriormente, que gran parte de las estructuras lógicas y matemáticas son generalizadas en términos de lógica difusa : relaciones lógicas, grupos, operaciones, funciones, operadores algoritmos etc. Y es a principio de la década de los setenta que se presentan en algunas universidades japonesas grupos de investigación enfocados al desarrollo de la teoría de lógica difusa como al estudio de su implementación.

La primera demostración práctica de un controlador por lógica difusa la dieron Ebrahim Mamdani y Assilian en el Queen Mary College en Londres, Inglaterra, [24] controlando un pequeño modelo de máquina de vapor y demostrando así la aplicación de la lógica difusa en la automatización.

La primera implementación comercial relevante tomó parte también en Europa para un controlador de temperatura de un horno de cemento desarrollado por F.L.Smidth y Cia. en Dinamarca [Holmblad, 1982], se dice que esta fue la primera implementación industrial.

En 1983 Fuji aplica la lógica difusa para el control de inyección química en plantas depuradoras de agua por primera vez en Japón y en 1987 Hitachi pone en marcha un controlador difuso para el tren de Sendai y la empresa Omron desarrolla los primeros controladores difusos comerciales.

El primer microprocesador de inferencia difusa fue desarrollado en los laboratorios *AT&T* Bell durante los primeros años de la década de los 80.

Sin embargo, los grandes propulsores de la masificación en aplicaciones de control por lógica difusa en operación están en Japón. Debido al boom que se presentó en 1987 y que por esta razón que los fabricantes japoneses han sido líderes en esta área, con aplicaciones como:

- **Robots.**
- **Ferrocarril subterráneo:** permitiendo detener suavemente los trenes.
- **Máquinas lavarropas:** usan sensores ópticos o electro-ópticos para determinar la suciedad de la ropa y determinar la cantidad óptima de detergente, agua y ciclos de lavado necesarios.
- **Refrigeradores:** sistema de auto-descongelado.
- **Cámaras:** para compensar en situaciones de luz complicadas (entre otras aplicaciones)
- **Frenos antibloqueo (ABS).**

También existen aplicaciones de lógica difusa en que ésta se utiliza como herramienta en la modelación de procesos complejos o pobremente definidos, construyendo modelos difusos (sin embargo, las aplicaciones prácticas actuales son generalmente controladores difusos).

2.2. Bases teóricas

2.2.1. Introducción

La metodología que hace uso de conjuntos difusos definidos por funciones de membresía en expresiones lógicas es llamada Lógica Difusa [2].

La lógica difusa fue diseñada para representar y razonar sobre una forma particular de conocimiento, expresado de forma verbal.

Lo difuso de una propiedad subyace en la ausencia de límites bien definidos del conjunto de objetos a los cuales se aplica dicha propiedad.

El pensamiento humano generalmente toma decisiones evaluando el grado de verdad de una variable, esto es debido a que no vemos variables discretas, es decir, aquellas variables cuyo valor es si ó no. Este tipo de razonamiento es difícil de reproducir mediante métodos normales, en las que las variables tienen que pertenecer completamente a algún grupo y por lo tanto no pertenecer a otro. La lógica difusa es una manera de resolver este problema.

En la lógica difusa una aseveración puede ser más o menos cierta, mientras que en la lógica clásica, una aseveración o es cierta o es falsa (nada intermedio). La lógica difusa extiende la lógica clásica de tal manera que permite valores intermedios entre cero y uno [7].

Una computadora puede interpretar una secuencia lingüística tal como, **si la lavadora está la mitad de llena, entonces usa menos agua**. La lógica difusa agrega inteligencia a la lavadora, ya que la computadora infiere una acción desde un conjunto de reglas *IF – THEN* . Por lo que "La lógica difusa es calcular con palabras" según Lofti A. Zadeh

2.2.2. Conjuntos Difusos

Los conjuntos difusos son un desarrollo que va más allá del concepto matemático que se tiene sobre un conjunto. Los conjuntos fueron estudiados por primera vez de manera formal por el matemático alemán George Cantor. Durante su vida encontró mucha oposición a su teoría de conjuntos, pero hoy en día la mayoría de los matemáticos creen que es posible representar una gran parte, no todas, las matemáticas por medio del lenguaje de la teoría de conjuntos.

Para el ingeniero en control, la lógica difusa y las relaciones difusas son de suma importancia para poder entender de qué manera trabajan las reglas difusas.

Conjuntos clásicos

Un conjunto es una colección de objetos de cualquier tipo. Para cualquier elemen-

to en el universo de discurso, puede determinarse si este pertenece o no al conjunto. De acuerdo a Cantor un conjunto X es una colección de objetos definidos y distinguibles de nuestra intuición que pueden ser tratados como un todo. Los objetos son los miembros de X .

El concepto "objetos de nuestra intuición", nos permite tener la libertad de elegir incluso a los conjuntos que tienen muchos miembros de manera infinita. Los objetos deben ser "definidos" dando un objeto y un conjunto, debe ser posible determinar si el objeto es miembro ó no del conjunto.

Los objetos también deben ser "distinguibles", dado un conjunto y sus miembros, debe ser posible determinar si cualesquiera de los miembros son diferentes ó el mismo.

Los miembros definen completamente un conjunto. Para determinar la membresía es necesario, que la sentencia " x es un miembro de X ", donde x es remplazada por un objeto y X por el nombre del conjunto, sea verdadera o falsa.

Teoría de Conjuntos Difusos

Los conjuntos difusos se distinguen de los conjuntos clásicos, en que un objeto puede pertenecer parcialmente a un conjunto ó puede pertenecer parcialmente a varios conjuntos.

Otro punto importante que cabe mencionar es que los conjuntos difusos no requieren tener una ecuación matemática precisa de su función de pertenencia, sino que pueden tener descripciones lingüísticas. Esto es parte fundamental de la utilidad que tiene que usar la lógica difusa en controladores, puesto que una planta se puede describir funcionalmente y no matemáticamente. Con esto se puede simular la experiencia, conocimiento y forma de pensar de un controlador humano utilizando una máquina.

En cuanto pertenecé ó no un objeto a un conjunto difuso, se llama su **grado de pertenencia o membresía** y este valor está definido por la función de pertenencia o membresía del conjunto, que por lo general se representa mediante la letra griega

” μ ”.

La definición de un conjunto difuso extiende la definición de un conjunto clasico, ya que los valores de membresía μ estan dentro del intervalo:

$$0 \leq \mu \leq 1$$

Un conjunto clasico es en consecuencia un caso especial de un conjunto difuso con valores de membresía restringidos a $\mu \in \{0, 1\}$ ”según L. A. Zadeh.

$$\mu(x) = \text{grado de pertenencia}$$

Un conjunto difuso A de un universo de discurso U es caracterizado por una función de membresía $\mu = U \rightarrow (0, 1)$, que asocia a cada elemento u de U , un número $\mu(u)$ que puede tomar todos los valores reales comprendidos dentro del intervalo $(0, 1)$, que representa el grado de membresía de u en A . El conjunto difuso A de $U = u_1, u_2, \dots, u_n$ y es expresado por:

$$A = \sum_{i=1}^n \frac{\mu_A u_i}{u_i} = \sum_{1,n} \mu_A u_i$$

Donde \sum representa unión.

El primer ejemplo utilizado por Lofti A. Zadeh, para ilustrar el concepto de conjunto difuso, fue el conjunto ”hombres altos”.

Según la teoría de la lógica clásica el conjunto ”hombres altos” es un conjunto al que pertenecerían los hombres con una estatura mayor a un cierto valor, que podemos establecer en 1.80 metros, por ejemplo, y todos los hombres con una altura inferior a este valor quedarían fuera del conjunto.

Así tendríamos que un hombre que mide 1.81 metros de estatura pertenecería al conjunto hombre altos, y en cambio un hombre que mida 1.79 metros de altura ya no pertenecería a ese conjunto. Sin embargo, no parece muy lógico decir que un hombre es alto y otro no lo es cuando su altura difiere en dos centímetros.

El enfoque de la lógica difusa considera que el conjunto ”hombres altos” es un conjunto que no tiene una frontera clara para pertenecer o no pertenecer a él me-

diante una función que define la transición de "alto" a "no alto" se asigna a cada valor de altura un grado de pertenencia al conjunto, entre "0 y 1". Así por ejemplo, un hombre que mida 1,79 podría pertenecer al conjunto difuso "hombres altos" con un grado 0.8 de pertenencia, uno que mida 1.81 con un grado 0,85, y uno que mida 1,50 m con un grado 0,1.

Visto desde esta perspectiva se puede considerar que la lógica clásica es un caso límite de la lógica difusa en el que se asigna un grado de pertenencia 1 a los hombres con una altura mayor o igual a 1,80 y un grado de pertenencia 0 a los que tienen una altura menor.

Así pues, los conjuntos difusos pueden ser considerados como una generalización de los conjuntos clásicos, la teoría clásica de conjuntos sólo contempla la pertenencia o no pertenencia de un elemento a un conjunto, sin embargo la teoría de conjuntos difusos contempla la pertenencia parcial de un elemento a un conjunto, es decir, cada elemento presenta un grado de pertenencia a un conjunto difuso que puede tomar cualquier valor entre 0 y 1.

Este grado de pertenencia se define mediante la función característica asociada al conjunto difuso, para cada valor que pueda tomar un elemento o variable de entrada x la función característica $\mu_A(x)$ proporciona el grado de pertenencia de este valor de x al conjunto difuso A .

Podemos además decir que el conjunto A es matemáticamente equivalente a su función de pertenencia o característica $\mu_A(x)$, ya que conocer $\mu_A(x)$ es lo mismo que conocer A .

Un conjunto difuso en el universo U se caracteriza por una función de pertenencia $\mu_A(x)$ que toma valores en el intervalo $[0, 1]$ y puede representarse como un conjunto de pares ordenados de un elemento x y su valor de pertenencia al conjunto:

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

Muchos conceptos de teoría clásica de conjuntos se pueden hacer extensivos a los conjuntos difusos, otros son exclusivos e inherentes a la teoría de conjuntos difusos.

Algunos de los más utilizados son los siguientes:

El soporte de un conjunto difuso A en el universo de discurso U es un conjunto crisp ”(numérico) que contiene todos los elementos de U que tienen un valor de pertenencia distinto de cero en A , esto es:

$$\text{sop}(x) = \{x \in U \mid \mu_A(x) > 0\}$$

Si el soporte de un conjunto difuso no contiene ningún elemento, tendremos un conjunto difuso vacío. Si el soporte de un conjunto difuso es un solo punto tendremos lo que se conoce como singleton difuso.

Un simple par $\langle x, \mu(x) \rangle$ es un singleton difuso debido a que todo el conjunto puede ser visto como la unión de sus singletons.

El uso de *singletons* simplifica considerablemente el proceso de inferencia y posibilita la implementación electrónica eficiente de los sistemas de inferencia difusos.

La mayoría de las veces es conveniente pensar en un conjunto A como un vector:

$$\mathbf{a} = (\mu(\mathbf{x}_1), \mu(\mathbf{x}_2), \dots, \mu(\mathbf{x}_n))$$

Se entiende entonces que cada posición ($i = 1, 2, 3, \dots, n$) corresponde a un punto en el universo de n puntos.

Operaciones con Conjuntos Difusos

Las operaciones más comunes realizadas con conjuntos difusos son las siguientes:

El conjunto complementario \bar{A} de un conjunto difuso A es aquel cuya función característica viene dada por:

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

La unión de dos conjuntos difusos A y B es un conjunto difuso $A \cup B$ en U está dada por:

$$\mu_{A \cup B}(\mathbf{x}) = \max[\mu_A(\mathbf{x}), \mu_B(\mathbf{x})]$$

La intersección de dos puntos difusos A y B en un conjunto difuso $A \cap B$ en U cuya función de pertenencia esta dada por:

$$A \cap B = \min[\mu_A(x), \mu_B(x)]$$

2.2.3. Universo

Los miembros de un conjunto difuso son tomados de un universo de discurso ó simplemente de un universo.

El universo contiene todos los objetos que puedan caer dentro de la consideración de comparar al conjunto U con la definición de conjunto difuso. Dependiendo del contexto que se de, por ejemplo:

El *universo* depende de las unidades de medida, una relación en el tiempo depende de sí el clima es medido en días, horas ó semanas .

Por lo tanto el universo es el rango de todos los posibles valores aplicables a una variable de un sistema.

2.2.4. Dominio

El rango de valores sobre el cual se ubica el ancho de una función de membresía se conoce como dominio.

2.2.5. Funciones de pertenencia ó membresía

Cada elemento dentro del universo de discurso, es un miembro del conjunto difuso, en cierto grado, inclusive cero .

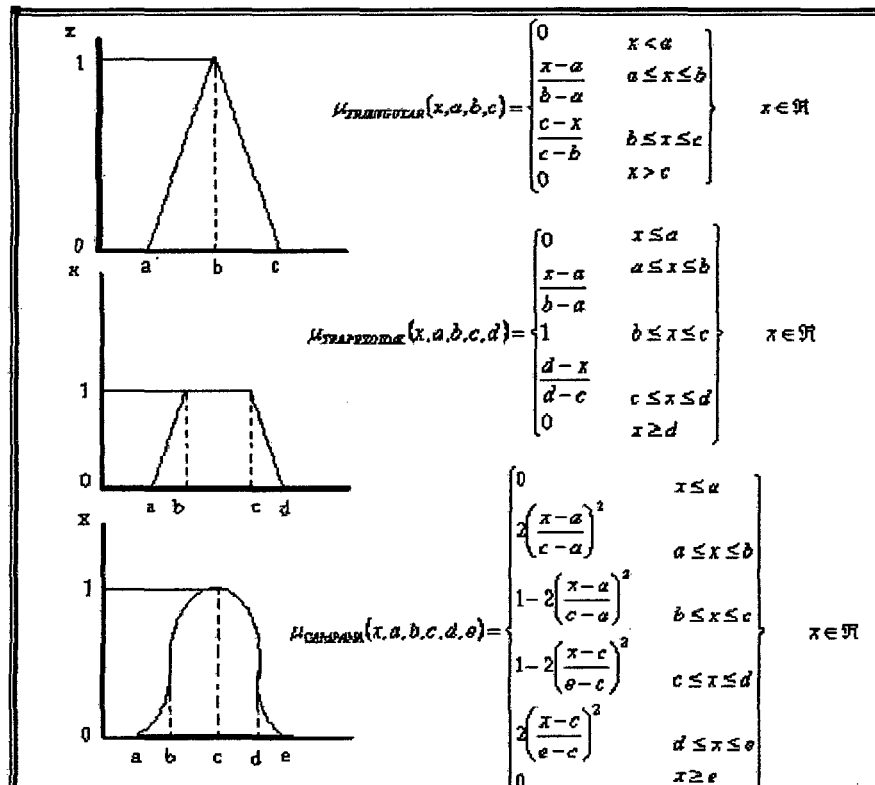
La función que asocia un número con cada elemento de x dentro del universo de discurso es llamada función de membresía .

El grado de membresía en los conjuntos difusos puede ser representado por medio de una función matemática la cual nos indica el grado de pertenencia que presenta un elemento dentro de un conjunto difuso.

En otras palabras, si A es un conjunto difuso, entonces la función de membresía $\mu_A(x)$ mide el grado con el cual el elemento x pertenece al conjunto A .

Las funciones de membresía se pueden clasificar según su forma, siendo las más comunes por su simplicidad las siguientes: trapezoidal, triangular, campana de gauss, gamma, pi, singleton etc. Las más importantes se presentan en la figura(2,1)

Figura 2.1: Funciones de membresía más comunes.



Existen dos formas alternativas para representar una función de membresía. La manera continua y la manera discreta.

Un conjunto difuso continuo A se define usando una función de membresía continua que generalmente es una función matemática.

Una función de membresía trapezoidal es una función continua lineal controlada por cuatro parámetros, $\{a, b, c, d\}$.

De la manera discreta la función de membresía y el universo de discusión son puntos en una lista (vector).

Donde el conjunto difuso discreto se define por pares ordenados:

$$A = \{\langle x_1, \mu(x_1) \rangle, \langle x_2, \mu(x_2) \rangle, \dots, |x_i \in u, i = 1, 2, \dots, n\}$$

Cada valor de membresía es una evaluación de la función de membresía en el punto discreto x_i en el universo u , y el conjunto completo es una colección usualmente finita de pares:

$$\langle x_i, \mu(x_i) \rangle$$

Cabe mencionar que las funciones de membresía no deben ser necesariamente simétricas.

2.2.6. Controlador lógico difuso

La finalidad de un controlador difuso es la de modificar el comportamiento de la planta mediante el cambio de una o varias entradas del sistema, de acuerdo a un conjunto de reglas y un proceso de inferencia que permita obtener las salidas deseadas.

Un controlador lógico difuso (*CLD*), emplea principios de lógica difusa y nos permite convertir estrategias de control lingüístico, basado en conocimiento experto, en una estrategia de control automático.

Los sistemas de control difuso se basan en reglas difusas que representan un mecanismo de decisión de control, para ajustar los cambios no deseados provenientes de la planta.

Normalmente, los sistemas de control difuso sustituyen las habilidades de un operador humano por un sistema basado en reglas difusas.

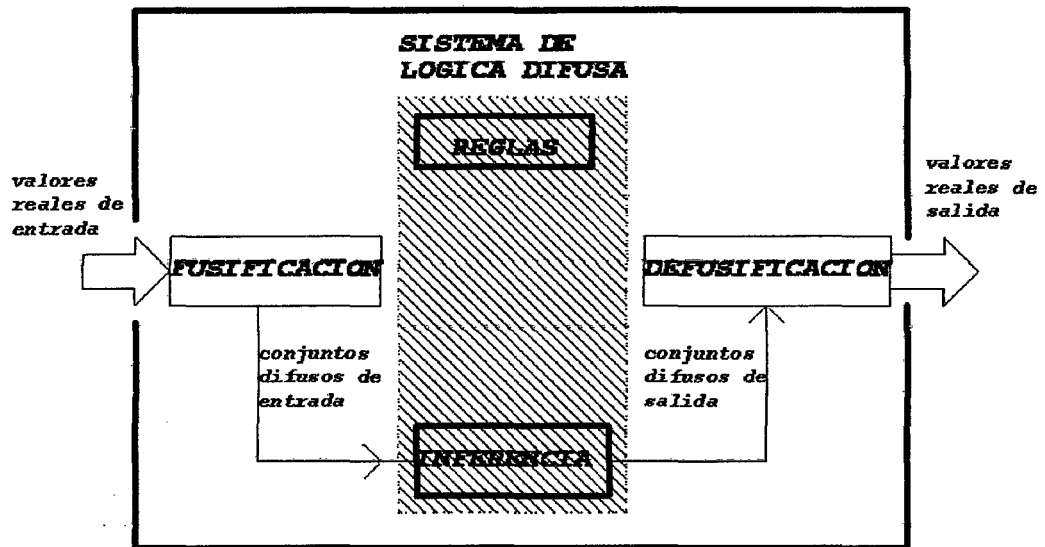
Un sistema difuso reemplaza las ecuaciones diferenciales del modelo matemático, por un modelo construido en base de un número de reglas.

Hasta la fecha no existe una metodología única, para elaborar un controlador difuso, sin embargo nos basamos en el método propuesto por C.C. Lee , el cual est compuesto por las siguientes partes:

- Fusificación.
- Base de conocimiento..
- Lógica de decisiones..
- Defusificación.

Las interconexiones entre estos módulos son mostradas en la figura 2.2)

Figura 2.2: Partes de un sistema de lógica difusa



Fusificación

Es el proceso de asignar valores de membresía o pertenencia a un valor numérico de entrada para cada una de las etiquetas difusas que forman la variable lingüística. El primer paso en la fusificación es dividir el universo de discurso, asignando etiquetas en cada variable del controlador difuso.

Posteriormente se establecen las funciones de membresía o pertenencia (conjuntos difusos) para dar significado numérico a cada etiqueta. Cada función de membresía

identifica el rango de los valores de entrada que corresponden a una etiqueta.

En el proceso de fusificación, las funciones de membresía definidas en relación a las variables de entrada mapean a los datos presentes para determinar el grado de verdad en la premisa de cada regla, por lo que es necesario elaborar primeramente las funciones de membresía correspondientes para cada entrada.

En general el proceso de fusificación lleva a cabo la transformación de valores crisp (reales) a valores difusos, realizando los siguientes pasos.

- Medir las magnitudes de las variables físicas de entrada.
- Efectuar un mapeo escalado, que transfiere el rango de valores de entrada a su correspondiente universo de razonamiento.
- Convertir valores de entrada en valores lingüísticos, los cuales son vistos como etiquetas pertenecientes a conjuntos difusos.

Base de conocimiento

Se obtiene de la experiencia desarrollada por un operador y de conocimientos sobre Ingeniería de control, y depende del proceso a controlar y los requerimientos de diseño. Está compuesta de dos partes:

- Una base de datos de control difuso
- Una base de reglas de control difuso (utiliza variables lingüística).

Base de datos

Incluye la discretización y normalización de los universos de entrada y salida, la definición de subconjuntos (partición de los universos y funciones de pertenencia) y satisfacer la propiedad de completitud.

1. **Discretización:** Debido a la necesidad de discretizar, ya que los datos se procesan en forma digital, este proceso genera niveles cuantizados, cada uno de los cuales representa un elemento genérico en un universo de discurso.
2. **Normalización:** Es el acondicionamiento de los valores reales, a valores de pertenencia entre cero y uno la normalización del universo discreto puede ser lineal o no.

3. **Partición de los universos.:** Se refiere al número de etiquetas que toma una variable lingüística; por ejemplo la variable "velocidad del motor", puede tomar los valores "muy baja", "baja", "media" y "alta". El número de términos está determinado por las características del sistema a controlar y la calidad de control.
4. **Funciones de pertenencia:** Como ya se ha mencionado anteriormente las funciones de pertenencia o membresía, representan gráficamente la relación que mantienen los elementos de un subconjunto difuso, dentro de un universo de discurso, con el grado de pertenencia al conjunto en cuestión. La representación de estas figuras puede tomar diferentes formas, siendo las más utilizadas las de forma triangular y trapezoidal. La utilización de alguna de ellas puede ser de manera arbitraria, dependiendo de la aplicación en particular.
5. **Completitud:** Esta propiedad indica que el algoritmo debe ser capaz de inferir una acción correcta para cada estado del proceso.

Base de Reglas

La estrategia de control, derivada de la experiencia se expresa mediante el uso de algoritmos difusos. Las reglas de control que forman el algoritmo difuso pueden definirse usando los siguientes criterios:

1. **Selección de las variables:** Las variables de entrada se seleccionan basándose en la experiencia y en conocimientos de Ingeniería y el cambio de error (derivada del error). La importancia de usar la variación de error se ilustra con un caso cotidiano, como es el hecho de atravesar una avenida, para lo cual no solo consideramos la distancia entre un auto y la persona que va cruzar la calle, sino también consideramos muy relevante la rapidez con la que el auto se desplaza.
2. **Origen y obtención de las reglas de control:** La experiencia y los conocimientos en Ingeniería de control. Es la que más se utiliza. En un controlador lógico difuso, su operación dinámica está caracterizada por un conjunto de reglas, compuestas por variables lingüísticas, basada en conocimiento experto es usualmente de la forma: *IF* (un conjunto de condiciones son satisfechas) *Then* (un conjunto de consecuentes que pueden inferir). Donde los antecedentes y los consecuentes de las reglas *IF – Then* (**Sí-Entonces**) son asociados con

conceptos difusos (términos lingüísticos), formando lo que se conoce como declaración condicional difusa en donde el antecedente es una condición sobre la base del estado de las variables del proceso y el consecuente es una acción de control para el sistema a controlar (proceso).

Lógica de decisiones

Como ya se ha mencionado un *CLD* puede emular a un hábil experto operador humano , mediante funciones de implicación difusa, y mecanismos de inferencia principalmente.

Inferencia difusa

Después que las variables de entrada han sido convertidas a valores de variables lingüísticas, el paso de inferencia difusa identifica las reglas que se aplican a cada situación, y mediante un método llamado *MIN/MAX*, ya que toma el mínimo peso de los antecedentes para determinar el peso de las reglas, y el máximo peso de las reglas para determinar la salida difusa, las cuales son empleadas en la entrada del generador de valores reales.

Defusificación

Una vez que hemos evaluado las reglas, los valores que obtuvimos son difusos, en otras palabras es una expresión lingüística que si la consideramos como salida para nuestro control, tendríamos el problema de que nuestro sistema a controlar no pueda interpretar tal orden lingüístico, lo que hace necesario convertir estas salidas difusas en un valor crisp (valor real), a esto se le llama defusificación.

La cuál combina todas las salidas difusas en un resultado específico para cada variable de salida, para esto se emplean diferentes métodos, dentro de los cuales los más comunes son los siguientes:

- Método del máximo.
- Método de centro de gravedad.
- Método de la altura.

Método del máximo

En este método la salida se toma como el valor de la conclusión de la regla que obtuvo mayor grado de pertenencia. Usar este de método no resulta tan atractivo puesto que se pierde algo de las ventajas de los conjuntos difusos.

Método de centro de gravedad

Este método es un tipo de "promedio" y por esto se utiliza mucho. Así no se pierde el aspecto "difuso" del controlador usado, como si sucede con el método del Valor Máximo.

En este método lo que se realiza es el cálculo del centro de masa del gráfico de la función de pertenencia, dándonos un sólo valor de salida, el inconveniente es que el cálculo del centro de masa se puede complicar bastante.

Para el caso de un Universo discreto, la salida defusificada se puede obtener mediante la siguiente expresión:

$$Z_0 = \frac{\sum_j^n \mu_{zz}(W_j) \cdot W_j}{\sum_j^n \mu_z(W_j)}$$

Donde:

- n es el numero de niveles de cuantizacion del universo de salida
- μ_z es una función de pertenencia de salida resultante.

Este método corta la función membresía de los respectivos terminos lingüísticos, de acuerdo con sus valores de pertenencia, formando áreas nuevas las cuales son sobrepuestas para formar una sola. Balanceando esta área se obtiene el valor preciso de salida. Lo anterior se ilustra por medio de la siguiente figura.

Método de la altura.

Se calcula para cada regla el centro de gravedad del conjunto difuso de salida, posteriormente se procede a calcular la salida del sistema como la media ponderada.

Concretamente en el caso de un universo discreto, la salida de control puede definirse por medio de la siguiente ecuación:

$$Z_0 = \sum_j^n \frac{W_j}{n}$$

Donde

- W_j es el valor central de la etiqueta lingüística j para la cual la función de membresía alcanza el máximo valor $\mu_z(W_j)$
- n es el número de etiquetas que alcanzan el valor máximo.

2.3. Glosario de términos

No-Lineal.-

Un proceso es no-lineal cuando no hay una simple relación proporcional entre causa y efecto.

Lazo cerrado.-

Configuración mediante la cual una muestra de la salida es restada a la entrada.

Lazo abierto.-

Configuración de un sistema que opera sin realimentación.

Función Característica.-

Relación entre dos o ms magnitudes variables que describen el funcionamiento de un dispositivo en condiciones especificadas.

Estabilización

Calidad de estable, que no se modifican las características que tiene un determinado cuerpo o sistema.

Acción de control.-

Algoritmo matemático que se genera en el controlador y acta sobre el proceso para minimizar o anular el error entre la referencia y la señal medida o sensada.

Algoritmo de control

Abstracción lógica de un sistema distribuido, se habla de un conjunto de procesos y de líneas de comunicación virtuales, con la finalidad de realizar una acción de control.

Sistema difuso

Es el sistema que usa la lógica difusa.

Sistema Lineal.-

Un sistema es lineal si se cumple el principio de superposición. Es decir, la relación entre la entrada y salida del sistema es una línea recta.

Variable controlada.-

Variable dinámica que se regula. Esta variable es medida por los sensores.

Variable de control.-

Variable entregada por el controlador para ser aplicada al actuador, después de ser transformada y amplificada. Esta variable depende de la variable error.

Setpoint ó punto de referencia.-

Valor deseado de la variable controlada, expresada en las mismas unidades de ésta

Capítulo III

III. METODOLOGÍA

Para desarrollar la técnica de lógica difusa en el sistema de estudio, como ya se ha mencionado, es un sistema que por su naturaleza no lineal, y algo complejo al momento de obtener su modelo matemático. A continuación obtendremos un modelo general que partirá del esquema mostrado figura 3.1, usando la ecuación dinámica de Lagrange, que nos va a proporcionar la exactitud del modelo,

3.1. Relación entre las variables de la investigación

El sistema del Brazo Robótico Translacional de 2DOF que lo denominamos (**BRT2DOF**) es mostrado en la figura 3.1.

M_1 es un servomotor de DC con un mecanismo de reducción y un encoder óptico articulado a una polea de radio R_p . Esta polea usa un cable para transmitir la fuerza F_1 que maneja el movimiento corredizo de un carro de masa M_c montado a un par de rieles a lo largo del eje de x .

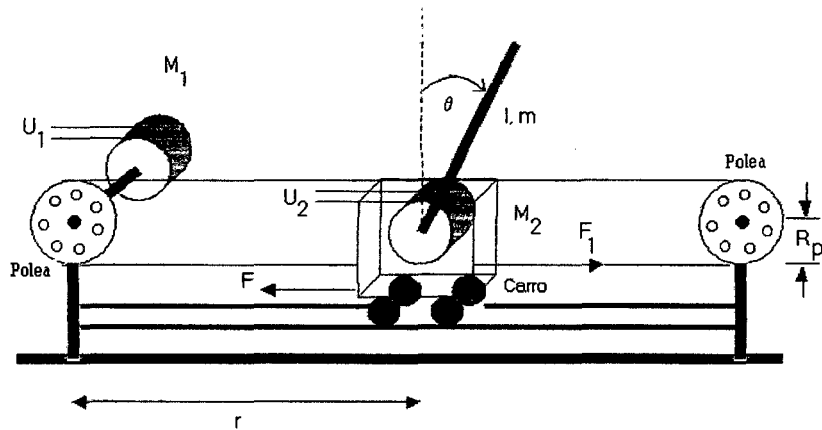
$\{M_2$ también es un servomotor DC con un encoder óptico empleado para manejar el movimiento rotatorio del brazo (libre de girar en ambas direcciones) del(**BRT2DOF**) alrededor de un punto del pivote; se asume que tal punto es el CG (centro de gravedad) del carro.

En la figura 3.1, θ es la posición angular del brazo de longitud l y masa m , r es la posición del carro, F es la fuerza friccional opuesta al movimiento del carro.

El proceso a ser controlado representa un proceso con dos entradas; los voltajes de control $K_{A1}u_1$ y $K_{A2}u_2$ aplicados a los terminales de armadura de M_1 y M_2 respectivamente, y dos salidas: r y θ . Los parámetros K_{A1} y K_{A2} son las ganancias de amplificación.

Teniendo en cuenta todas estas consideraciones, las variables para nuestra planta o proceso son:

Figura 3.1: Brazo Robótico Translacional de segundo orden (BRT2DOF).



Variables independientes (variables de entrada):

- Voltaje de entrada del motor M_1 denominado u_1 .
- Voltaje de entrada del motor M_2 denominado u_2 .

Estos voltajes son amplificados por K_{A1} y K_{A2} para darle la potencia necesaria a los motores M_1 y M_2 .

Variables dependientes (variables de salida)

son las variables ha ser controladas y se indica a continuación.

- Posición lineal del Brazo , que es la posición del carro denominado r .
- Posición angular del brazo de longitud L y masa m denominado θ .

3.2. Tipo de investigación

Investigación transversal experimental. Se aplican herramientas matemáticas para el modelado, simulación, diseño de los controladores y la simulación de la respuesta del sistema de control en lazo cerrado.

3.3. Diseño de la investigación

3.3.1. Modelado de la Planta

El propósito de este modelo matemático es desarrollar y probar una ley de control que pueda hacer que la trayectoria del carro y la trayectoria del brazo sigan trayectorias de referencia arbitrarias. Nosotros asumimos que las dos componentes de la salida $y(r, \theta)$ son medidas y que las fuerzas de control $u(u_1, u_2)$ es la entrada de control.

Nuestro primer objetivo es encontrar la relación dinámica entre $u(u_1, u_2)$, $y(r, \theta)$ para esto podemos usar las ecuaciones de Lagrange. Para expresar la energía en este sistema, introducimos las variables de la ecuación de Lagrange. El principio de Lagrange nos permite escribir en forma directa las ecuaciones de energía cinética T , energía potencial U y energía disipada D . En otras palabras, una vez que modelamos T , U y D , tendremos las ecuaciones dinámicas del sistema.

Ecuaciones de Energía para el Carro

El carro es restringido para un movimiento en la dirección horizontal, las ecuaciones dinámicas para el carro comprenden el desarrollo de las energías cinética (T_1), potencial (U_1) y de disipación (D_1) dadas por.

$$T_1 = \frac{1}{2} M_c \dot{r}^2 \quad (3.1)$$

$$U_1 = 0 \quad (3.2)$$

$$D_1 = \frac{1}{2} F_c \dot{r}^2 \quad (3.3)$$

Ecuaciones de Energía para el Brazo

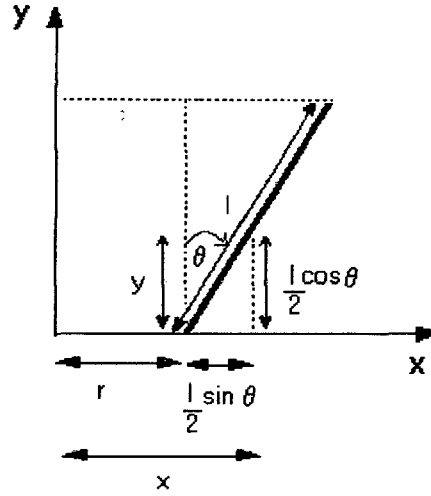
Sea la energía cinética (T_2) del brazo dada por.

$$T_2 = \frac{1}{2} J \dot{\theta}^2 + \frac{1}{2} m (\dot{x}^2 + \dot{y}^2) \quad (3.4)$$

donde J es el momento de inercia referido al brazo de masa de m de la figura 3.1.

El desarrollo de la ecuación (3.4) se explica con ayuda de la figura 3.2 que muestra las coordenadas (x, y) en función de los parámetros del sistema del brazo robótico.

Figura 3.2: Subsistema del brazo robótico.



$$x = r + \frac{l}{2} \sin \theta \quad (3.5)$$

$$y = \frac{l}{2} \cos \theta \quad (3.6)$$

derivando las ecuaciones (3.5) y (3.6).

$$\dot{x} = \dot{r} + \frac{l}{2} \cos(\theta) \dot{\theta} \quad (3.7)$$

$$\dot{y} = -\frac{l}{2} \sin(\theta) \dot{\theta} \quad (3.8)$$

elevando al cuadrado.

$$\dot{x}^2 = \dot{r}^2 + \dot{r}l \cos(\theta) \dot{\theta} + \frac{l^2}{4} \cos^2(\theta) \dot{\theta}^2 \quad (3.9)$$

$$\dot{y}^2 = \frac{l^2}{4} \sin^2(\theta) \dot{\theta}^2 \quad (3.10)$$

sumando las ecuaciones (3.9) y (3.10).

$$\dot{x}^2 + \dot{y}^2 = \dot{r}^2 + \dot{r}l \cos(\theta) \dot{\theta} + \frac{l^2}{4} \dot{\theta}^2 \quad (3.11)$$

reemplazando (3.11) en (3.4) obtenemos la energía cinética.

$$T_2 = \frac{1}{2} J \dot{\theta}^2 + \frac{1}{2} m (\dot{r}^2 + \dot{r}l \cos(\theta) \dot{\theta} + \frac{l^2}{4} \dot{\theta}^2) \quad (3.12)$$

La energía potencial y la energía de disipación del brazo, es dada por las ecuaciones.

$$U_2 = \frac{m_l g}{2} \cos\theta \quad (3.13)$$

$$D_2 = \frac{1}{2} C \dot{\theta}^2 \quad (3.14)$$

Ecuaciones de Lagrange del Proceso BRT2DOF

La ecuación de Lagrange generalizada es.

$$\frac{\partial}{\partial t} \left(\frac{\partial L}{\partial \dot{q}} \right) - \frac{\partial L}{\partial q} + \frac{\partial D}{\partial \dot{q}} = F_q \quad (3.15)$$

donde, L es el conjunto de energías del sistema ($L = T - U$), D es la disipación de energía externa del sistema, q es la coordenada de variable independiente del sistema, \dot{q} es la derivada de variable independiente del sistema y F_q es la fuerza externa que actúa en la dirección de q .

Para el sistema BRT de 2DOF, los términos de las energías individuales componen en su conjunto la totalidad energética del sistema.

Energía cinética del sistema

$$T = T_1 + T_2 = \frac{1}{2} M_c \dot{r}^2 + \frac{1}{2} J \dot{\theta}^2 + \frac{1}{2} m (\dot{r}^2 + \dot{r} l \cos(\theta) \dot{\theta} + \frac{l^2}{4} \dot{\theta}^2)$$

Energía potencial del sistema

$$U = U_1 + U_2 = \frac{m_l g}{2} \cos\theta$$

Energía disipada del sistema

$$D = D_1 + D_2 = \frac{1}{2} F_c \dot{r}^2 + \frac{1}{2} C \dot{\theta}^2$$

Podemos generalizar la ecuación de Lagrange para el vector q en función de coordenadas generales (r, θ) del proceso BRT de 2DOF es formulado de acuerdo con (3.15).

$$\frac{\partial}{\partial t} \left(\frac{\partial T}{\partial \dot{r}} \right) - \frac{\partial T}{\partial r} + \frac{\partial U}{\partial r} + \frac{\partial D}{\partial \dot{r}} = F_1 \quad (3.16)$$

$$\frac{\partial}{\partial t} \left(\frac{\partial T}{\partial \dot{\theta}_1} \right) - \frac{\partial T}{\partial \theta_1} + \frac{\partial U}{\partial \theta_1} + \frac{\partial D}{\partial \dot{\theta}_1} = T_L \quad (3.17)$$

Para el planteo de las ecuaciones de Lagrange, se observa que el término de energía potencial U no es una función del tiempo derivativo de r . Las ecuaciones dinámicas del Lagrange serán desarrolladas para las variables en donde interviene el sistema.

Desarrollamos las derivadas parciales que indica la ecuación (3.16) para determinar la fuerza F_1 .

$$\frac{\partial T}{\partial \dot{r}} = (M_c + m)\dot{r} + \frac{ml}{2}\cos(\theta)\dot{\theta} \quad (3.18)$$

$$\frac{d}{dt}\left(\frac{\partial T}{\partial \dot{r}}\right) = (M_c + m)\ddot{r} - \frac{ml}{2}\sin(\theta)\dot{\theta}^2 + \frac{ml}{2}\cos(\theta)\ddot{\theta} \quad (3.19)$$

$$\frac{\partial T}{\partial r} = 0; \quad \frac{\partial U}{\partial r} = 0; \quad \frac{\partial D}{\partial \dot{r}} = F_c\dot{r} \quad (3.20)$$

Obtenemos la primera ecuación parcial del sistema para F_1 .

$$F_1 = (M_c + m)\ddot{r} - \frac{ml}{2}\sin(\theta)\dot{\theta}^2 + \frac{ml}{2}\cos(\theta)\ddot{\theta} + F_c\dot{r} \quad (3.21)$$

Desarrollamos las derivadas parciales que indica la ecuación (3.17) para determinar el torque T_L .

$$\frac{\partial T}{\partial \dot{\theta}} = J\dot{\theta}_1 + \frac{ml}{2}\cos(\theta)\dot{r} + \frac{ml^2}{4}\dot{\theta} \quad (3.22)$$

$$\frac{d}{dt}\left(\frac{\partial T}{\partial \dot{\theta}}\right) = J\ddot{\theta} + \frac{ml}{2}\cos(\theta)\ddot{r} - \frac{ml}{2}\sin(\theta)\dot{\theta}\dot{r} + \frac{ml^2}{4}\ddot{\theta} \quad (3.23)$$

$$\frac{\partial T}{\partial \theta} = -\frac{ml}{2}\sin(\theta)\dot{r}; \quad \frac{\partial U}{\partial \theta} = -\frac{ml}{2}g\sin(\theta); \quad \frac{\partial D}{\partial \dot{\theta}_1} = C\dot{\theta} \quad (3.24)$$

Obtenemos la segunda ecuación parcial del sistema para T_L .

$$T_L = \frac{ml}{2}\cos(\theta)\ddot{r} + \left(\frac{ml^2}{4} + J\right)\ddot{\theta} + C\dot{\theta} - \frac{mlg}{2}\sin(\theta) \quad (3.25)$$

El modelo Lagrangiano puede ser derivado usando las ecuaciones (3.21) y (3.25). Sin embargo si nosotros estamos interesados en tener como fuerzas de control a los voltajes de armadura u_1 y u_2 en lugar de F_1 y T_L , respectivamente, para esto es necesario modelar los subsistemas eléctricos del BRT de 2DOF.

Modelo del Servomotor DC M_1 - Polea

La figura 3.3 muestra el servomotor de DC M_1 que es articulado a una polea. Despreciando la inductancia de la armadura L_a , el voltaje de entrada $K_A u_1$ aplicado a la armadura se da por.

$$R_a I_{a1} + V_{b1} = K_A u_1 \quad (3.26)$$

El voltaje de la fuerza electromotriz es proporcional a la velocidad de motor.

$$V_{b1} = K_b \dot{\theta}_{m1} \quad (3.27)$$

La ecuación del torque T_{m1} del servomotor es dado por.

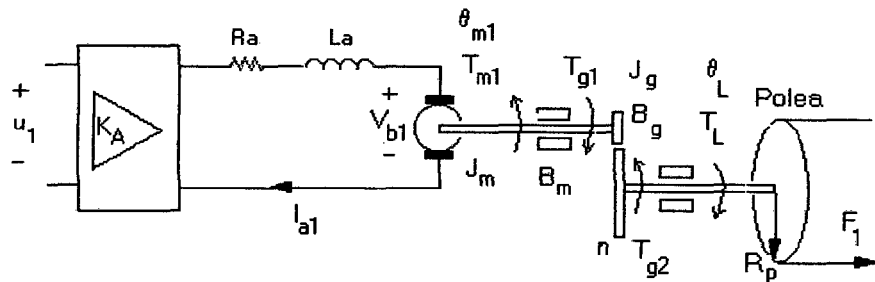
$$T_{m1} = J_m \ddot{\theta}_{m1} + B_m \dot{\theta}_{m1} + T_{g1} \quad (3.28)$$

El torque T_{g2} requiere la transmisión de la polea, y el torque en la polea es directamente proporcional a la fuerza F_1 , luego formulamos.

$$T_{g2} = n T_{g1} = (J_g + J_p) \ddot{\theta}_L + (B_g + B_p) \dot{\theta}_L + F_1 R_p \quad (3.29)$$

donde J_m , J_g y J_p son los momentos de inercia de la armadura, mecanismo de reducción de engranaje y polea, respectivamente, mientras B_m , B_g y B_p son las constantes de fricción de la armadura, mecanismo de reducción de engranaje y polea respectivamente.

Figura 3.3: Subsistema motor - polea.



Nosotros también hemos usado la relación para la conservación de energía de engranajes $T_{g1} \dot{\theta}_{m1} = T_{g2} \dot{\theta}_L$ con $(n \theta_L = \theta_{m1})$, de este modo podemos expresar (3.28) como.

$$T_{m1} = J_m n \ddot{\theta}_L + B_m n \dot{\theta}_L + T_{g1} \quad (3.30)$$

despejando T_{g1} de (3.29) y reemplazamos en (3.30), luego agrupando términos tendremos.

$$nT_{m1} = J_{eq1} \ddot{\theta}_L + B_{eq1} \dot{\theta}_L + F_1 R_p \quad (3.31)$$

donde

$$J_{eq1} = n^2 J_m + J_g + J_p$$

$$B_{eq1} = n^2 B_m + B_g + B_p$$

el torque del servomotor T_{m1} es proporcional a la corriente I_{a1} .

$$T_{m1} = K_m I_{a1} \quad (3.32)$$

reemplazando (3.32) en (3.31), luego despejamos I_{a1} .

$$I_{a1} = \frac{J_{eq1}}{nK_m} \ddot{\theta}_L + \frac{B_{eq1}}{nK_m} \dot{\theta}_L + \frac{R_p F_1}{nK_m} \quad (3.33)$$

usando las ecuaciones (3.27), (3.33) en (3.26), luego despejamos F_1 .

$$F_1 = \frac{nK_m K_A}{R_a R_p} u_1 - \frac{J_{eq1}}{R_p} \ddot{\theta}_L - \left(\frac{B_{eq1}}{R_p} + \frac{n^2 K_m K_b}{R_a R_p} \right) \dot{\theta}_L \quad (3.34)$$

El movimiento rotacional de la polea puede ser transformado como un movimiento translacional del carro usando la siguiente relación (ver figura 3.3).

$$\theta_L = \frac{r}{R_p} \quad (3.35)$$

derivando (3.35) y sustituyendo en (3.34), obtenemos la ecuación general del subsistema del motor.

$$F_1 = \frac{nK_m K_A}{R_a R_p} u_1 - \frac{J_{eq1}}{R_p^2} \ddot{r} - \left(\frac{B_{eq1}}{R_p^2} + \frac{n^2 K_m K_b}{R_a R_p^2} \right) \dot{r} \quad (3.36)$$

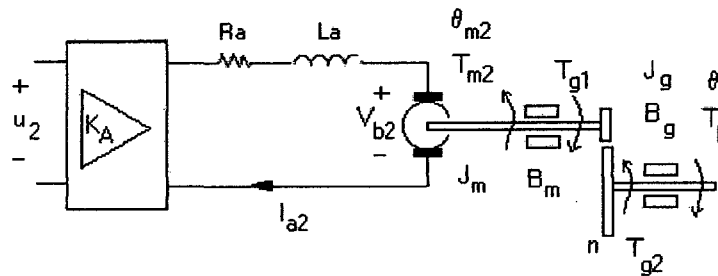
igualando (3.36) con (3.21), se obtiene la primera ecuación dinámica de Lagrange del sistema.

$$u_1 = \frac{R_a R_p}{n K_m K_A} \left\{ (M_c + m + \frac{J_{eq1}}{R_p^2}) \ddot{r} + (F_c + \frac{B_{eq1}}{R_p^2} + \frac{n^2 K_m K_b}{R_a R_p^2}) \dot{r} + \frac{ml}{2} \cos(\theta) \ddot{\theta} + \dots - \frac{ml}{2} \sin(\theta) \dot{\theta}^2 \right\} \quad (3.37)$$

Modelo del Servomotor DC M_2 - Brazo

La figura 3.4 muestra el servomotor de DC M_2 que articula a un punto del pivote localizado en el CG del carro. Teniendo en cuenta que M_1 y M_2 tiene los mismos parámetros y despreciando la inductancia de la armadura L_a .

Figura 3.4: Subsistema del torque del brazo.



El voltaje de entrada $K_A u_2$ aplicado a la armadura se da por.

$$R_a I_{a2} + V_{b2} = K_A u_2 \quad (3.38)$$

donde:

$$V_{b2} = K_b \dot{\theta}_{m2} \quad (3.39)$$

La relación de desplazamiento angular del eje del motor y eje de carga es.

$$\theta_{m2} = n\theta \quad (3.40)$$

La ecuación de torque T_{m2} del servomotor (ver figura 3.4) será.

$$T_{m2} = J_m n \ddot{\theta} + B_m n \dot{\theta} + T_{g1} \quad (3.41)$$

El torque T_{g2} requiere manejar el brazo que se expresa.

$$T_{g2} = nT_{g1} = J_g\ddot{\theta} + B_g\dot{\theta} + T_L \quad (3.42)$$

reemplazamos (3.42) en (3.41), siendo T_L el torque de carga.

$$nT_{m2} = J_{eq2}\ddot{\theta} + B_{eq2}\dot{\theta} + T_L \quad (3.43)$$

donde:

$$J_{eq2} = n^2 J_m + J_g$$

$$B_{eq2} = n^2 B_m + B_g$$

El torque T_{m2} del servomotor es proporcional a la corriente I_{a2} .

$$T_{m2} = K_m I_{a2} \quad (3.44)$$

sustituyendo (3.40), (3.39) en (3.38) y despejando I_{a2} .

$$I_{a2} = \frac{K_A}{R_a} u_2 - \frac{nK_b}{R_a} \dot{\theta} \quad (3.45)$$

sustituyendo (3.45), (3.44) en (3.43) y despejando T_L .

$$T_L = -J_{eq2}\ddot{\theta} - (B_{eq2} + \frac{n^2 K_b K_m}{R_a})\dot{\theta} + \frac{nK_m K_A}{R_a} u_2 \quad (3.46)$$

igualando (3.46) con (3.25), se obtiene la segunda ecuación dinámica de Lagrange del sistema:

$$u_2 = \frac{R_a}{nK_m K_A} \left\{ \frac{ml}{2} \cos(\theta) \ddot{r} + \left(\frac{ml^2}{4} + J + J_{eq2} \right) \ddot{\theta} + \left(C + B_{eq2} + \frac{n^2 K_m K_b}{R_a} \right) \dot{\theta} + \dots - \frac{mlg}{2} \sin(\theta) \right\} \quad (3.47)$$

Modelo Lagrangiano

La ecuación (3.21) con (3.37) y (3.25) con (3.47) conforman el proceso Lagrangiano del BRT2 de 2DOF que toma la forma:

$$u_1 = m_{11}\ddot{r} + p_{11}\dot{r} + m_{12}\cos\theta\ddot{\theta} + p_{12}\sin\theta\dot{\theta}^2 \quad (3.48)$$

$$u_2 = m_{22}\ddot{\theta} + p_{22}\dot{\theta} + m_{21}\cos\theta\ddot{r} + d_{21}\sin\theta \quad (3.49)$$

Las ecuaciones (3.48) y (3.49) pueden ser transformadas en su representación de la matriz como sigue:

$$M(x)\ddot{x} + P(x, \dot{x})\dot{x} + d(x) = u \quad (3.50)$$

donde x y u son los estados:

$$x = \begin{bmatrix} r \\ \theta \end{bmatrix} \quad u = \begin{bmatrix} u_1 \\ u_2 \end{bmatrix}$$

las matrices M , P y Q del modelo del sistema son.

$$M = \begin{bmatrix} m_{11} & m_{12}\cos\theta \\ m_{21}\cos\theta & m_{22} \end{bmatrix} \quad P = \begin{bmatrix} p_{11} & p_{12}\dot{\theta}\sin\theta \\ 0 & p_{22} \end{bmatrix} \quad Q = \begin{bmatrix} 0 \\ d_{21}\sin\theta \end{bmatrix}$$

La tabla 3.1 muestra los coeficientes de las matrices M , P y Q del sistema, mientras que la tabla 3.2 muestra los parámetros del servomotor DC. El método de identificación de los parámetros del motor y la obtención de las constantes de fricción del carro F_c y del brazo C se explican en el anexo A. Se han considerado las siguientes equivalencias para el momento de inercia y la fricción viscosa $J_{eq} = J_{eq1} = J_{eq2}$ y $B_{eq} = B_{eq1} = B_{eq2}$, debido a que reflejan valores similares, tal como lo demuestran los resultados experimentales obtenidos en la identificación de los parámetros para los dos servomotores DC.

Discretización del sistema Brazo Robótico Transnacional de 2DOF

Después de haber llegado a determinar el modelo matemático del sistema Brazo Robótico Translacional de 2DOF denominado **BRT2DOF**, de manera analítica,

Tabla 3.1: Coeficientes de las matrices M , P y Q .

$m_{11} = \left(\frac{R_a R_p}{n K_A K_m}\right) (M_c + m + \frac{J_{eq}}{R_p^2})$	$p_{11} = \left(\frac{R_a R_p}{n K_A K_m}\right) (F_c + \frac{B_{eq}}{R_p^2} + \frac{n^2 K_m K_b}{R_a R_p^2})$
$m_{12} = \left(\frac{R_a R_p}{n K_A K_m}\right) \frac{ml}{2}$	$p_{12} = -\left(\frac{R_a R_p}{n K_A K_m}\right) \frac{ml}{2}$
$m_{21} = \left(\frac{R_a}{n K_A K_m}\right) \frac{ml}{2}$	$p_{21} = 0$
$m_{22} = \left(\frac{R_a}{n K_A K_m}\right) \left(\frac{ml^2}{4} + J + J_{eq}\right)$	$p_{22} = \left(\frac{R_a}{n K_A K_m}\right) (C + B_{eq} + \frac{n^2 K_m K_b}{R_a})$
$d_{12} = 0$	$d_{21} = -\left(\frac{R_a}{n K_A K_m}\right) \frac{mlg}{2}$

Tabla 3.2: Parámetros estimados del sistema.

Símbolo	Descripción	Valor	Unidades
M_c	masa del carro + masa de la polea	1.6492	Kg
m	masa del brazo	0.0519	Kg
l	longitud del brazo	0.30	m
J	momento de inercia del brazo robótico	$2,38e^{-3}$	Kgm^2
J_{eq}	momento de inercia equivalente	3.0551×10^{-4}	Kgm^2
B_{eq}	fricción viscosa equivalente	5.3848×10^{-4}	$Nm/rad/s$
C	coeficiente de fricción viscosa brazo	$9,5e^{-4}$	Kgm^2/seg
F_c	coeficiente de fricción del carro	2.81	Kg/seg
g	aceleración de la gravedad	9.81	$mseg^{-2}$
K_b	constante de fuerza contraelectromotriz	0.0565	$V/rad/s$
K_m	constante de torque de motor	0.0421	Nm/A
R_p	radio de la polea	0.048	m
R_a	resistencia de armadura	5.3	Ω
n	factor de reducción	12.5	
K_A	constante de ganancia de amplificación	5.5	

procederemos a ingresar dicho modelo matemático al programa Matlab, a fin de poder realizar varios análisis adicionales, tales como comportamiento natural de la posición angular y lineal del Brazo en lazo abierto es decir sin un controlador, luego realizar el análisis con el controlador difuso en tiempo continuo y su posterior discretización del modelo matemático. Todos los análisis anteriores podran realizarse analíticamente, pero resultaría sumamente engorroso y tedioso por la complejidad

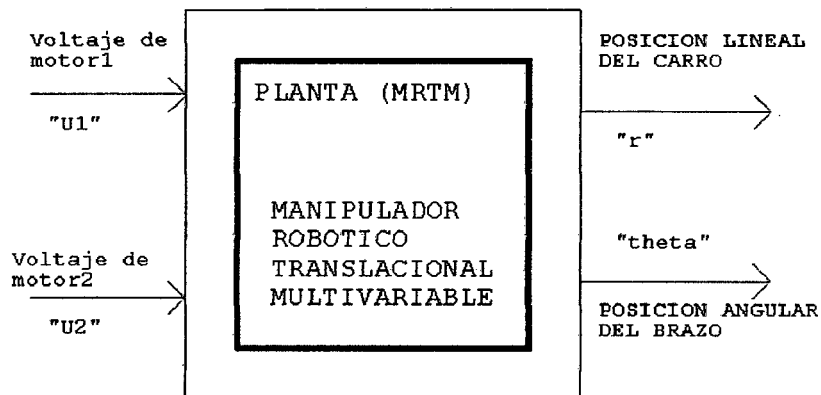
matemática del modelo , por ello a fin de no detenernos sólo en la parte matemática es que utilizamos como herramienta fundamental para esta tarea, al software Matlab. El Programa correspondiente se encuentra en el anexo B, y tiene como nombre Programa `openlooptest_mrtm2dof.m` que para su ejecución requiere la función denominado `f_mrtm2dof.m` . Se presenta tres casos tal como se describe el comportamiento del sistema.

Simulación del BRT2DOF en lazo abierto

En la *fig3,5* se muestra el diagrama de bloque del proceso BRT de 2DOF en lazo abierto, en el que se muestra la planta con sus entradas y salidas.

Las entradas del sistema nos representa los voltajes aplicados a cada motor DC y se simboliza como u_1 y u_2 ; las salidas son en este caso las posición lineal (r) del carro y las posición angular del del segundo motor θ .

Figura 3.5: Planta no lineal a controlar con inductancia de valor cero.



La figura 3.6 en la parte superior se describen las posiciones para el carro descrita por la variable r y para el giro angular del brazo descrito por la variable θ .

Para realizar la simulación del sistema en lazo abierto se realizó las siguientes rutinas denominadas

- **Función `f_mrtm2dof.m`.** Esta función devuelve los valores de las aceleraciones para posición y el ángulo respectivamente considerando el modelo matemático de la ecuación de EULER_LAGRANGE.

- **Programa openlooptest_mrtm2dof.m.** Este programa hace uso de la función anterior y la rutina para la simulación , así mismo la rutina para mostrar en forma animada la simulación del sistema

Se presenta la simulación para varios casos considerando los voltajes de entradas de los motores M_1 y M_2 , por lo que en los gráficos obtenidos se muestra la evolución del desplazamiento lineal r y angular θ y también la posición final de la animación del sistema

Primer caso: Simulación del sistema con una entrada escalón unitario

$u_1 = 1V$ y $u_2 = 0,0000000001V$ valor muy pequeño aplicada en $t = 0s$.

Observar la figura 3.6 que para esta simulación están representadas por una línea recta con una pendiente constante para la evolución de la posición del carro y la posición de la varilla $\theta = \frac{\pi}{6}$ posición inicial , se observa la evolución de la varilla hasta un valor final de $\theta = \pi$.

El carro presenta una velocidad autosostenida en $0,2m/s$, mientras que el brazo comienza a girar en sentido horario por inercia de gravedad.

Segundo Caso: Simulación del sistema para $u_1=1.0V$ $u_2=0.5V$

Para esta condición con $u_1 = 1,0$, $u_2 = 0,5$ la posición del carro r y la posición angular de la varilla θ se muestra en La figura 3,7 Se observa en la figura 3,7 que la posición del carro en $ts = 10s$, el recorrido es 2 metros y la posición angular de la varilla es mas de 1000 grados , ó sea a dado cerca de tres vueltas la varilla.

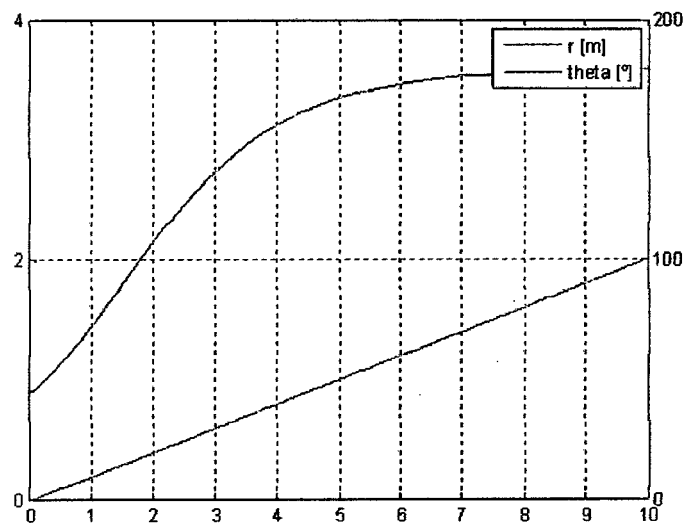
Tercer Caso: Simulación del sistema para $u_1 = 1,0V$, $u_2 = 1,0V$

para esta condición con $u_1 = 1,0$, $u_2 = 1,0$ se muestra en la figura 3,8 las respuesta del sistema. En la figura se muestra que el desplazamiento lineal del carro es de 2 metros y el desplazamiento angular de la varilla es de aproximadamente de 10 vueltas

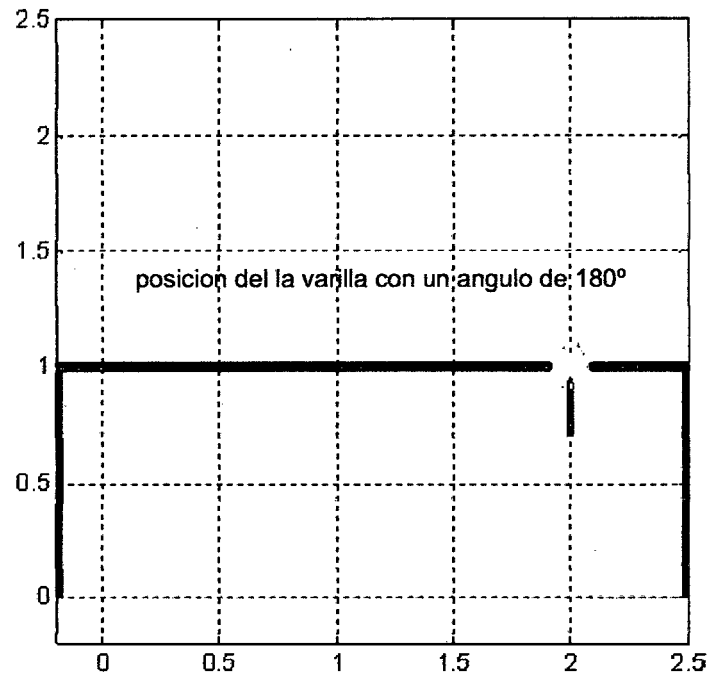
3.3.2. Requerimientos de diseño del controlador del Brazo Robótico Translacionl de 2DOF

Los requerimientos que debe satisfacer el controlador basado en Lógica Difusa son:

Figura 3.6: Respuesta de la posición lineal r y angular θ debido a un escalón.

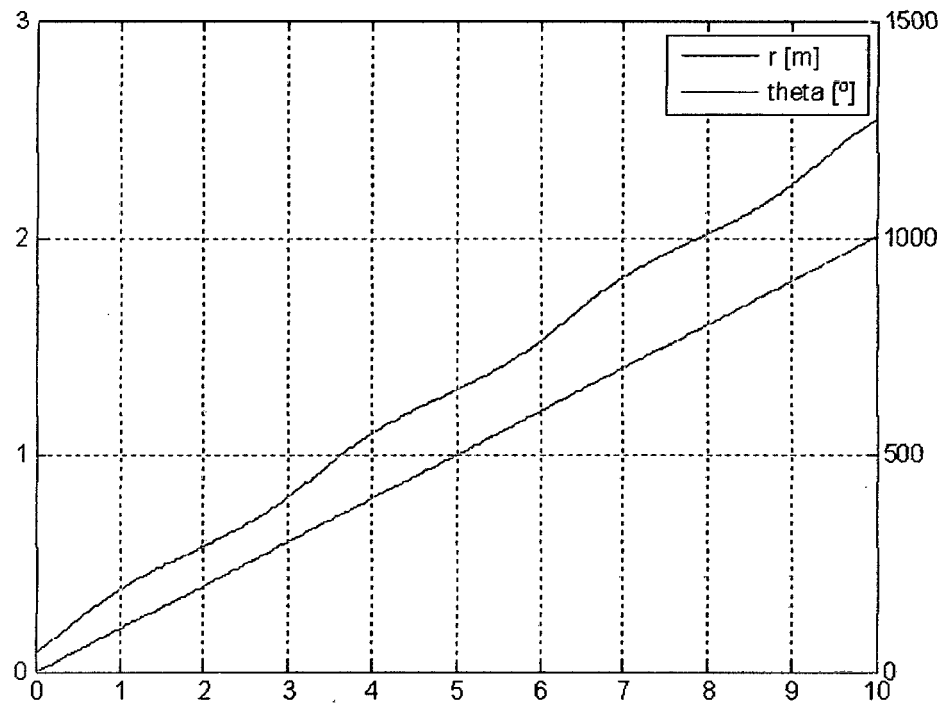


$u_1=1$ $u_2=0.0000001$

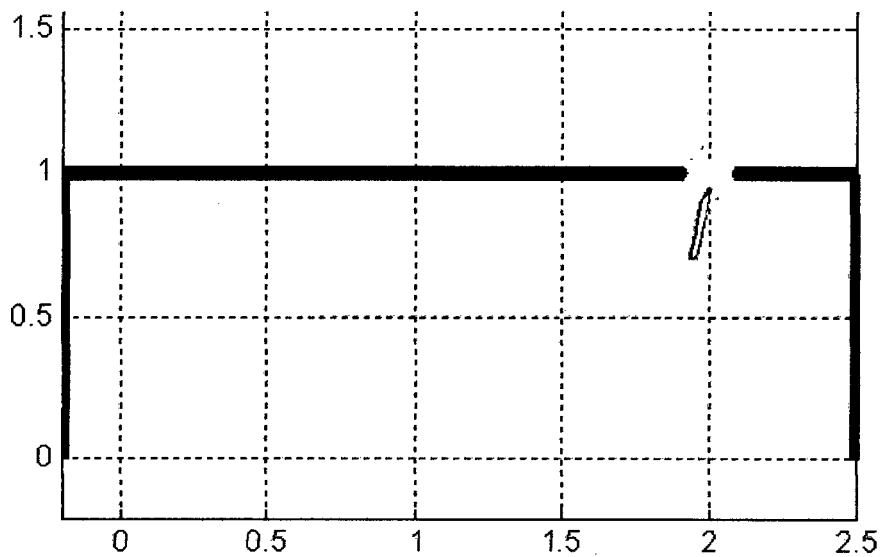


- **Trabajar con la característica No Lineal del BRT2DOF.**- Vimos que el modelo dinámico de la planta es no lineal. En este caso no se necesita linealizar el modelo.
- **Absorber las imperfecciones de modelado.**- Tales como perturbaciones, pequeñas cargas, ruido, y aproximaciones.

Figura 3.7: Respuesta de la posición lineal r y angular θ con $u_1 = 1,0v$, $u_2 = 0,5V$

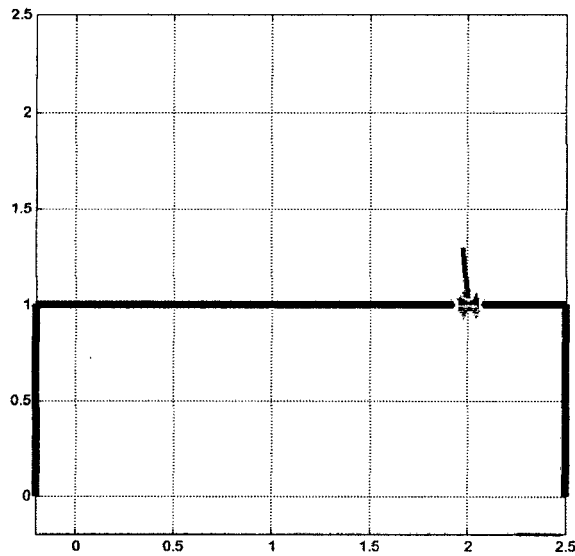
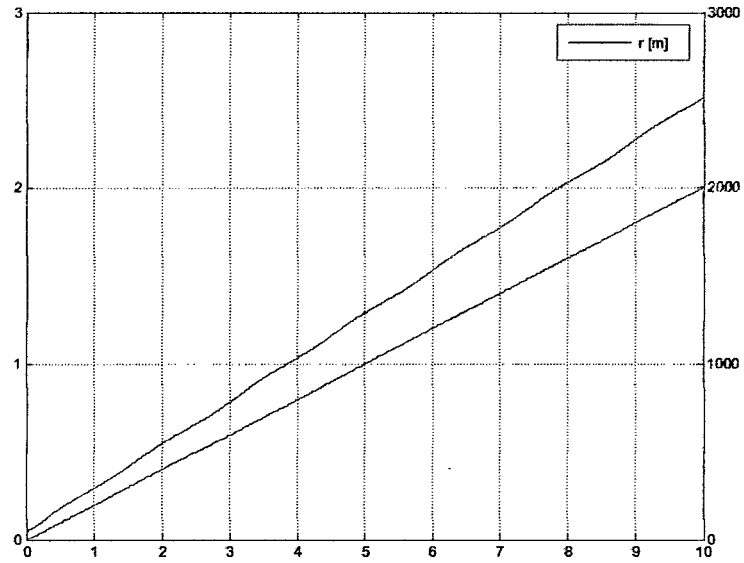


$u_2 u_1 = 1.0 = 0.5$



- **Adaptarse a condiciones de operación no modeladas.**- Se refiere a variables no modeladas dentro del proceso, variación de parámetros con el tiempo, imperfecciones mecánicas y eléctricas, etc.

Figura 3.8: Respuesta de la posición lineal r y angular θ con $u_1 = 1,0, u_2 = 1,0V$.



A.-Controladores basados en Lógica Difusa.

En un controlador difuso, los datos pasan a través de un bloque de **preprocesamiento**, a través de un controlador, y luego a través de un bloque de **postprocesamiento**.

El preprocesamiento

Puede ser un escalamiento lineal o no lineal, así como una cuantización en caso que las funciones de pertenencia sean discretizadas (vectores). Si no fuera así, entonces las funciones de pertenencia deben ser implementadas mediante una función sencilla.

Cuando se diseña la base de reglas, el diseñador necesita considerar el número de conjuntos difusos de las entradas, su forma, y su traslape. Las reglas deben ser implementadas por el diseñador, a menos que se empleen medios más avanzados como auto-organización o redes neuronales.

También debemos tomar una decisión sobre el método de activación, que puede ser el método Producto o el método Mínimo.

Luego debemos seleccionar el método de **defusificación** (entendiéndose este término como la transición de la respuesta difusa a la respuesta real), y también el método de agregación de las inferencias.

El postprocesamiento

Consiste en escalar la salida. En caso que el controlador sea del tipo incremental, el postprocesamiento debe incluir un integrador.

La siguiente es una lista de decisiones de diseño:

- **Decisiones relacionadas con la base de reglas.**- Número de entradas/salidas, reglas de inferencia, universos de discurso continuo/discreto, el número de funciones de pertenencia, ancho de base y traslape de las funciones de pertenencia, salida "singleton" (basada en funciones de pertenencia tipo impulso unitario).
- **Decisiones relacionadas con el motor de inferencia.**- Conectivos, modificadores, operación de activación, operación de agregación y operación de acumulación.
- **Método de Defusificación.**- COG (Center of Gravity), COGS (COG con Singleton), MOM (Mean of Maximums), Yager, etc.

- **Pre y Postprocesamiento.**-Escalamiento, factores de ganancia, cuantización, y tiempo de muestreo.

Algunos de las decisiones mencionadas anteriormente deben ser siempre consideradas. Otros, según el caso.

3.3.3. Diseño del control difuso en tiempo continuo

La lógica difusa (Fuzzy Logic) ha surgido como una herramienta importante para el control de subsistemas y procesos industriales complejos, así como también para la electrónica de entretenimiento y hogar, sistemas de diagnóstico y otros sistemas expertos. Aunque la lógica difusa se inventó en Estados Unidos, el crecimiento rápido de esta tecnología ha comenzado desde Japón y ahora nuevamente ha alcanzado USA y también Europa.

La lógica difusa es todavía un boom en Japón, el número de cartas patentando aplicaciones aumenta exponencialmente. Principalmente se trata de aplicaciones más bien simples de lógica difusa. Lo difusa ha llegado a ser una palabra clave para vender. Los artículos electrónicos sin componentes difusos se están quedando gradualmente desfasados. Como una mordaza, que muestra la popularidad de la lógica difusa, cada vez es más frecuente un sello con **fuzzy logic** impreso sobre el producto.

En Japón la investigación sobre lógica difusa es apoyada ampliamente con un presupuesto enorme. En Europa y USA se están realizando esfuerzos para alcanzar al tremendo éxito japonés. Por ejemplo, la NASA emplea lógica difusa para el complejo proceso de maniobras de acoplamiento.

La lógica difusa es básicamente una lógica multivaluada que permite valores intermedios para poder definir evaluaciones convencionales como **s/no**, **verdadero/falso**, **negro/blanco**, etc. Las nociones como **más bien caliente** o **poco fro** pueden formularse matemáticamente y ser procesados por computadoras. De esta forma se ha realizado un intento de aplicar una forma más humana de pensar en la programación de computadoras.

La lógica difusa se inició en 1965 por Lotfi A. Zadeh, profesor de ciencia de computadoras en la Universidad de California en Berkeley.

En esencia un **controlador lógico difuso**, contiene un algoritmo que es capaz de convertir una estrategia de control lingüística en una estrategia de control automático.

Con la lógica difusa se pueden diseñar aplicaciones para que las máquinas respondan con mayor inteligencia a la imprecisión y a las condiciones del mundo exterior, con lo que se busca imitar el comportamiento humano.

La creación de una máquina con lógica difusa, es forjar un sistema experto, en donde el comportamiento de la máquina, va a estar basado totalmente en el conocimiento del experto o de la persona que aporta sus conocimientos empíricos para el funcionamiento de ésta. El conocimiento del experto es el conocimiento empírico de cómo controlar el fenómeno, sin conocer ningún modelo del sistema a controlar.

El introductor de la idea en la ingeniería actual fué Lotfi Zadeh, un profesor matemático iraní residente en los Estados Unidos que en 1965 en la Universidad de California en Berkeley, Zadeh publicó un reporte llamado Conjuntos difusos, donde presentó formalmente la teoría de estados multivaluados, y se introdujo el término difuso en la literatura técnica. Lotfi Zadeh expande la idea de los estados clásicos de lógica Booleana a los que nombró estados difusos.

A diferencia de la lógica Booleana, la lógica difusa es multivaluada, por lo que se considera un número infinito de grados de valor entre cierto y falso.

La lógica difusa define grados de relación y grados de verdad de algo, algunas cosas pueden ser parcialmente verdaderas y parcialmente falsas simultáneamente.

Luego teniendo en cuenta las consideraciones indicadas se realizó los siguientes procedimientos de diseo:

I.-Diseño del controlador en tiempo continuo de posición horizontal

Lamentablemente no existe un procedimiento estándar, pero es posible definir

ciertos pasos de trabajo para llegar a un buen diseño del controlador con lógica fuzzy.

La metodología que se propone sigue los lineamientos generales. Como se verá, en cada uno de ellos no existe una solución única, y su diseño se asienta en conocimiento heurístico sobre el proceso y pruebas de ensayo-error hasta conseguir una respuesta adecuada a nuestras especificaciones previas, en un proceso de refinamiento que la mayoría de las veces requiere una nueva definición en el criterio de selección usado en estos pasos. Esta **sintonización** del controlador puede hacerse también en forma automática, por ejemplo con algoritmos genéticos, o redes neuronales, de modo tal de encontrar un controlador óptimo. Incluso esta optimización puede hacerse en línea, una vez que el controlador está en funcionamiento normal en planta, aunque deben contemplarse especialmente los aspectos de estabilidad. Sin embargo, este tipo especial de controlador difuso (CLD) (denominados adaptivos, será objeto de otro trabajo.

La metodología propuesta se resume en los siguientes pasos:

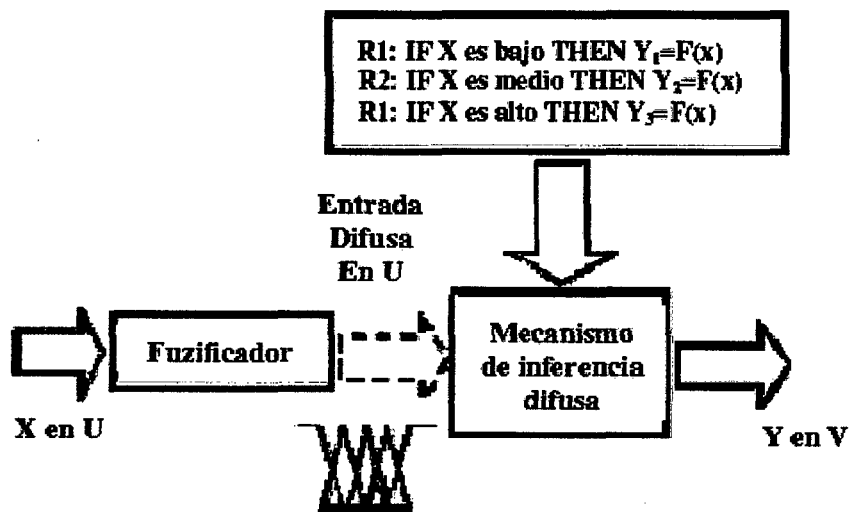
1. Selección de las variables y del Universo de Discurso
2. Selección de la estrategia de emborronamiento
 - (a) Particionamiento de los espacios de entrada y salida
 - (b) Elección de las funciones de pertenencia
 - (c) Discretización/normalización de los Universos de Discurso
 - (d) completitud
3. Construcción de la base de reglas
 - (a) Elección de las variables de entrada y las de control
 - (b) Elección de las reglas de control
4. Selección de lógica de toma de decisiones
 - (a) Definición del tipo de implicancia a utilizar
 - (b) Interpretación de los conectivos AND y ALSO
 - (c) Definición del operador de composición
 - (d) Mecanismo de inferencia
5. Selección de la estrategia de desemborronamiento

Para nuestro caso se consideró el controlador modelo difuso TAKAGI- SUGENO ; este modelo se ha convertido en un método de razonamiento lógico muy conocido y utilizado , debido a sus numerosas aplicaciones en sistemas difusos ,en controles difusos y en ingeniería difusas.

La mayor ventaja del modelo Takagi-sugeno es que es un modelo simple y puede ser utilizado en muchas aplicaciones practicos para el modelado y de sistemas complejos.

Este modelo se caracteriza por relaciones basadas en reglas difusas, donde las premisas de cada regla representan subespacios difusos y las consecuencias es una relacion lineal de entrada - salida (*Takagi,sugeno*1995).Las variables de entrada en las premisas de cada regla son relacionadas por operadores "Y", y las variables de salida es función de las variables de estado en general una función lineal figura 3.9

Figura 3.9: Procesamiento general sistema difuso Takagi Sugeno



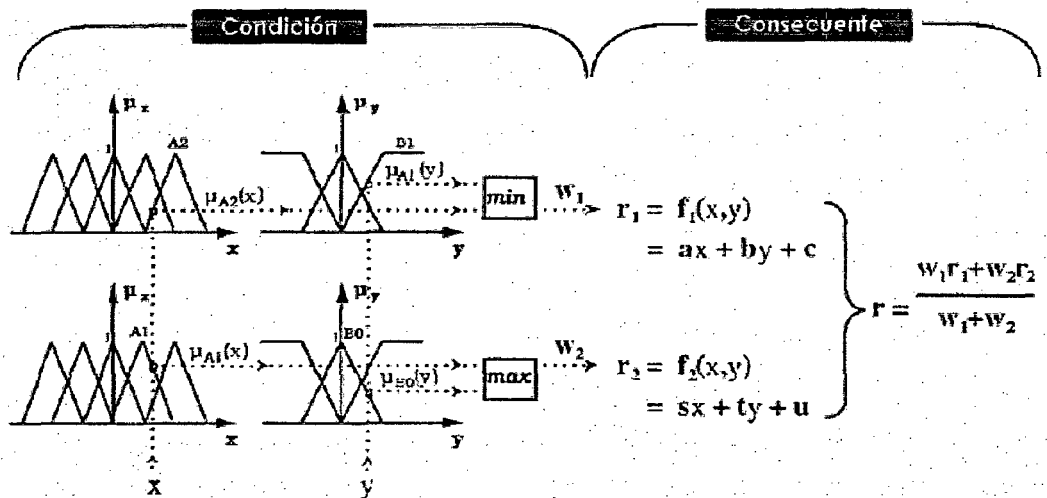
Las reglas de base de conocimiento de un sistema sugeno es diferente a las de sistemas Mandami pues el consecuente de estas reglas ya no es una etiqueta lingüística si no es una función de entrada que tenga el sistema en un momento dado esto se ilustra a continuación.

IF La entrada es baja THEN La salida $Y = F(\text{entrada})$

En los sistemas difusos tipo sugeno, los valores que arrojan los consecuentes de las diferentes reglas que se han activado en un momento determinado ya son valores numéricos, por lo que no se necesita una etapa de defuzzyficacion en la figura 3.10 se presentan el modelo *takagi – Sugeno* de dos regla difusas , con dos variables de entrada y una de salida.

Debido a una dependencia lineal en el sistema de variables de entrada , este modelo es ideal para actuar como supervisor interpolador de múltiples controladores lineales que aplican en diferentes condiciones de operaciones de un sistema dinámico no lineal.

Figura 3.10: Modelo Takagi Sugeno



De lo indicado el sistema a controlar denominado Manipulador robótico Traslacional multivariable , se considero el controlador FLC tipo *Takagi – sugeno* doble sistema de inferencia en paralelo defuzzyficacion de orden 0 ;Teniendo dos variables a controlar como son :

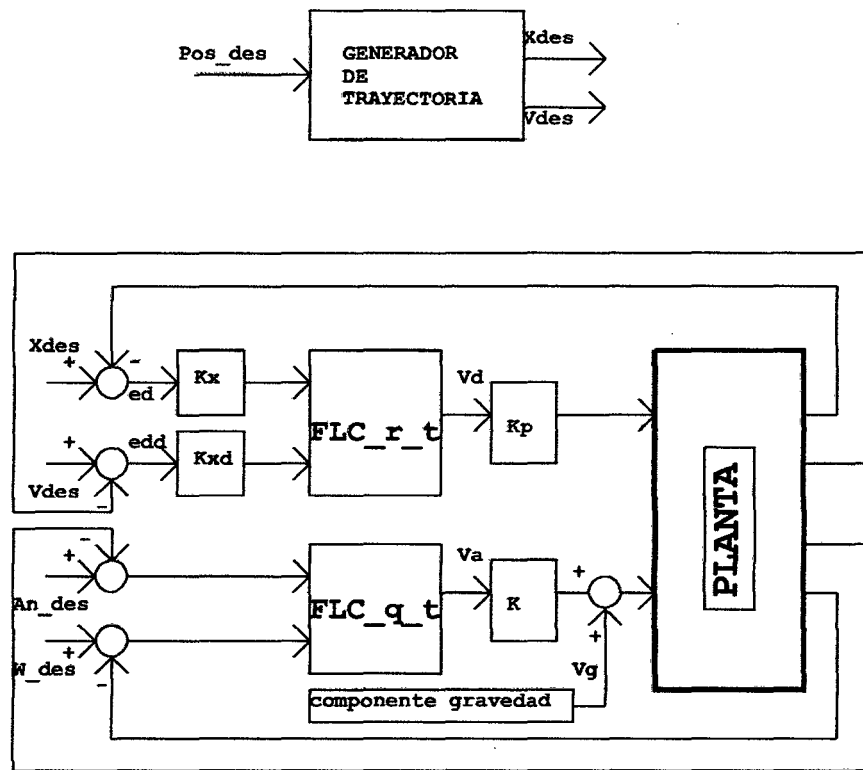
La posición del carro r y la posición angular θ de la varilla.

El sistema de control en tiempo continuo se puede expresar mediante el diagrama de bloques de la figura 3.11 en el cual se muestra las diferentes etapas a tomar en cuenta .

Teniendo en cuenta los lazos de control sobre las variables.

Para realizar el control sobre el sistema debe considerarse que si el operador

Figura 3.11: Sistema de control en tiempo continuo



tiene que equilibrar la varilla, entonces éste debe descomponer su control según las variables en juego del proceso, e.g., mantener a cero el ángulo de la varilla, o mantener la posición de referencia (o "Set Point"), la cual puede también ser variante en el tiempo (problema de seguimiento), también se debe considerar la posición del carro. Los controladores basados en Lógica Difusa ofrecen un razonamiento para el control basado en un Set Point Difuso de cada variable que controlan. Este Set Point es inherente y conlleva a dos opciones de control sobre las variables del sistema. Así, podemos establecer

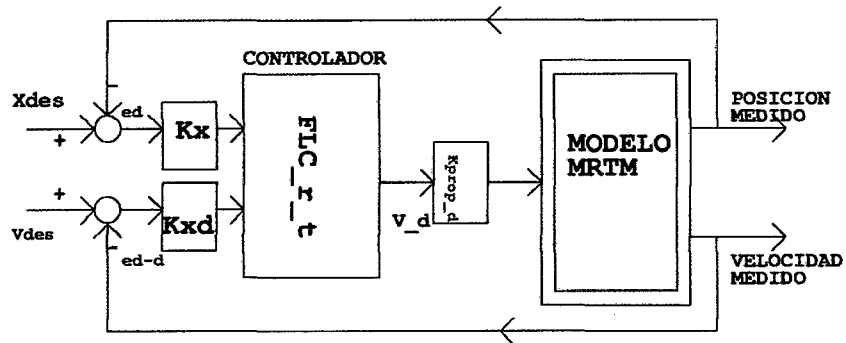
1. Control de posición del carro: X con setpoint constante ó variable
2. Control de posición angular: θ ángulo de la varilla
3. Compensador de gravedad : V_g

Dichas opciones de control se aprecian mejor en el esquema de la Figura 3.11 , en donde se ha incluido el modelo del BRT2DOF dentro del diagrama de bloques con fines de simulación.

Controlador de posición Horizontal del Subsistema Motor polea

La estructura del controlador de posición del carro se muestra en la figura 3.12 con la nomenclatura que utilizaremos. donde:

Figura 3.12: Sistema de control de posición



- e_d = señal error de posición
- $ed.d$ = señal error de la derivada ó señal error de velocidad
- K_x = constante de adaptación de entrada antes de la fuzzificación de la posición para darle el nivel adecuado al error de posición
- $K_{x.d}$ = constante de adaptación de entrada antes de la fuzzificación de la velocidad para darle el nivel adecuado al error de velocidad
- $K_{prop.d}$ = constante proporcional de la salida para la posición para dar el nivel de tensión adecuado al motor de posicionamiento del brazo
- $V.d$ = señal de control , como voltaje de salida

Elección del Universo de Discurso de las Variables de Entrada.

Es usual escoger universos de discurso estandarizados, p.e. $[-1,+1]$ y luego aplicar factores de escala para cada variable según la planta a controlar . En nuestro caso hemos supuesto que las referencias a seguir por la planta estarán comprendidas entre $[-1,+1]$:

1. El error de posición variará normamente entre $[-15,15]$

2. La derivada lo hace entre [-15,15]

a esto tiene que aplicarse factores de escala.

FUNCIONES DE PERTENENCIA

Partición y definición difusa de las variables de entrada

a.-Variable de entrada distancia:distance

El rango de esta variable es de (-15, 15). Esta variable va a tomar 5 términos lingüísticos, ya que son suficientes para caracterizar esta variable de entrada distancia . Las funciones de pertenencia son gaussianas respectivamente . Los términos lingüísticos utilizados tienen la siguiente matriz

Entrada matriz de parámetros de la MSFS de distancia

<i>distance</i>		
[-15.0	-0.400	1
0.11	-0.250	0
0.11	0.000	0
0.11	0.250	0
15	0.350	1]

b.-Variable de entrada distancia-d:distance_d

La variable de entrada distance_d es la derivada de la distancia osea la velocidad ; sea considerado el rango de [-15,15], esta variable va a tomar tres términos lingüísticos , suficiente para caracterizarla,se considera la siguiente matriz

Entrada matriz de parámetros de la MSFS de distancia_d ó error de velocidad

<i>distance_d</i>		
[-15.0	-0.350	1
0.200	0.000	0
15	0.350	1]

Elección del Universo de Discurso de las Variables de Salida:Votaje de Actuación

El rango de este variable es de $[-21,21]$; Lo que se pretende con la fusificación de los términos lingüísticos de la variable de salida no es lo mismo que con las variables de entrada. Para la variable de salida lo que se pretende es que una vez truncado un término lingüístico, se sepa a qué valor de la variable de salida pertenece ese truncamiento y se pueda construir el consecuente, para posteriormente, mediante el método de defusificación elegido, obtener el valor concreto de salida. se ha considerado un vector de voltajes tales:

Vector de voltajes: $Voltaje_r$							
$MSFs = [1$	2	3	4	5	6	7]	
$Voltaje_q = [12$	-14	-7	0	7	14	21]	

Base de reglas

A la hora de definir las funciones de pertenencia hay que tener en cuenta que la mínima salida defusificada que vamos a tener va a ser el centro del término lingüístico situado más a la izquierda, y la máxima salida va a ser el centro del término lingüístico situado más a la derecha. Esta variable va a tomar 5 términos lingüísticos, ya que son suficientes para caracterizar esta variable. Todas las funciones de pertenencia son del tipo triangular, ya que para las variables lingüísticas de salida, en la mayoría de las aplicaciones, sólo se utilizan funciones de pertenencia triangulares, ya que estas ofrecen buenos resultados. Los términos lingüísticos utilizados tienen el siguiente significado y parámetros de definición:

RB_r			
$NL=1$	2	3	
$NS=2$	3	4	
$-Z=3$	4	5	
$PS=4$	5	6	
$PL=5$	6	7	

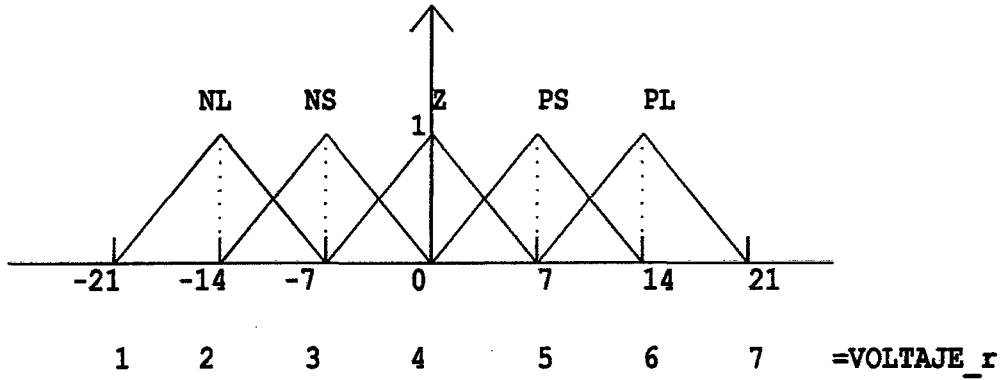
Haciendo una representación gráfica de las funciones de pertenencia de la variable de salida 'tensión de actuación' queda tal como se muestra en la figura 4.5

Diseño de las reglas del controlador de posición

Para definir las reglas se ha tenido en cuenta que:

- Un error negativo significa que se obtiene una salida mayor que la referencia, por lo que se debe disminuir la tensión de actuación a la planta.

Figura 3.13: Funciones de Pertenencia de la variable de salida tensión de actuación (v)



- Un error positivo significa que se obtiene una salida menor que la referencia, por lo que se debe aumentar la tensión de actuación a la planta.
- Si la derivada del error es del mismo signo que el error, significa que el error está aumentando.
- Si la derivada del error es de signo contrario al error, significa que el error está disminuyendo.
- Si la derivada del error es cero, significa que el error permanece constante.
- Si el error es negativo grande (NL) ó positivo grande (PL) la derivada del error da igual, ya que hay que aplicar la mínima tensión o la máxima tensión respectivamente.

Considerando lo anterior y sabiendo que las reglas se definen con la clásica sentencia (SI - ENTONCES), las reglas definidas son las siguientes considerando la figura 4.6

Figura 3.14: Representación compacta de las reglas del controlador de posición

Ed		NL	NS	Z	PS	PL
		NM	NS	PS		
Ed_d	N					
	Z	NL	NS	Z	PS	PL
	P		NS	PS	PM	

Defusificación

El método de defusificación que se ha elegido es el de el **centro del área** ó **centroide**, ya que proporciona buenos resultados en aplicaciones de control, requiere un tiempo computacional mediano y es sencillo de implementar. La fórmula que emplea este método para obtener el valor del consecuente (número real de la salida) se indica en la ecuación 4.1:

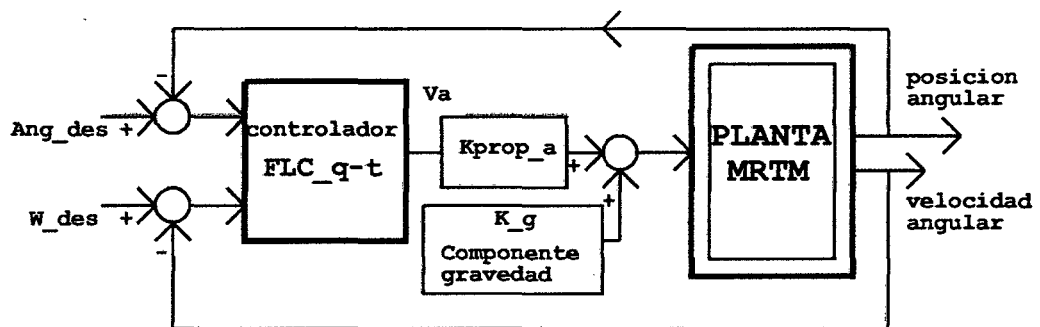
$$Salida = \frac{\sum_{i=1}^n A_i \cdot u_i}{\sum_{i=1}^n A_i} \quad (3.51)$$

donde u_i es la posición del centroide de cada término lingístico de la salida, y A_i es el área activa bajo la función de pertenencia de cada término lingístico.

Controlador de posición angular (brazo)

El esquema general del controlador de posición angular del brazo del sistema BRT2DOF se muestra en la figura 3.15 con la nomenclatura que utilizaremos.

Figura 3.15: Sistema de control de posición angular del brazo del BRT2DOF



donde:

- ang_{des} = ángulo deseado.
- W_{des} = velocidad angular referencial.
- K_g = componente de gravedad del brazo del BRT2DOF.
- K_{prop_a} = constante proporcional de salida para la señal de control del ángulo.

Diseño del controlador de posición angular del brazo BRT2DOF

En primer lugar se identifican las variables de entrada y salida del controlador. La información de la que se dispone es el valor de salida de la planta a controlar, que

mediante comparación con la referencia introducida, proporciona el error existente y la derivada del error (velocidad y sentido del cambio del error). Estas serán las entradas del controlador borroso. La salida del controlador será la tensión que se debe aplicar a la planta en función de las entradas, para llegar al objetivo. Para el cálculo del error, se resta la referencia menos la salida de la planta.

$$error = Referencia - salida = ang_d - ang_c$$

donde:

- ang_d = ángulo deseado
- ang_c = ángulo controlado

El cálculo de la derivada del error se realiza a partir de dos muestras consecutivas del error.

$$d_e = \frac{e_{(actual)} - e_{(anterior)}}{Tiempo_{muestreo}}$$

Fusificación

conocidas las variables de entrada y salida hay que elaborar el número de términos lingüísticos que van a representar a cada variable y qué tipo de función de pertenencia va a emplear cada uno.

Variable de entrada error de posición (e)

Esta variable va a tomar 5 términos lingüísticos, ya que son suficientes para caracterizar esta variable de entrada, Los términos lingüísticos utilizados tienen el siguiente significado.

error de posicion			
NL	[-4.0	-2.00	1
NS	0.8	-0.50	0
Z	0.8	0.0	0
PS	0.8	0.50	0
PL	4	2.0	1]

Variable de entrada derivada del error de posición (angle_d)

Esta variable va a tomar 3 términos lingüísticos, suficientes para caracterizarla. Los términos lingüísticos utilizados tienen el siguiente significado y parámetros:

derivada del error de posición angle.d			
N[-0.75	-0.75	1
Z	3.0	0.0	0
P	0.75	0.75	1]

Variable de salida (voltaje_q) para posición angular del brazo

El rango de esta variable es de (-12, 12). Lo que se pretende con la fusificación de los términos lingüísticos de la variable de salida no es lo mismo que con las variables de entrada. Para la variable de salida lo que se pretende es que una vez truncado un término lingüístico, se sepa a qué valor de la variable de salida pertenece ese truncamiento y se pueda construir el consecuente, para posteriormente, mediante el método de defusificación elegido, obtener el valor concreto de salida. Para las pruebas de simulación se ha considerado un vector de voltajes tales:

Vector de voltajes: $Voltaje_q$						
MSFs =[-1	2	3	4	5	6	7]
$Voltaje_q$ =[-12	-4.5	-2	0	2	4.5	12]

Esta variable va a tomar 5 términos lingüísticos, ya que son suficientes para caracterizar esta variable. Todas las funciones de pertenencia son del tipo triangular, ya que para las variables lingüísticas de salida, en la mayoría de las aplicaciones, sólo se utilizan funciones de pertenencia triangulares, ya que éstas ofrecen buenos resultados. Los términos lingüísticos utilizados tienen el siguiente significado y parámetros de definición.

Base de reglas : Angle			
NL	1	2	3
NS	2	3	4
Z	3	4	5
PS	4	5	6
PL	5	6	7

Generador de Trayectoria de Referencia para el Sistema de Control BRT2DOF

Para obtener la posición deseada del desplazamiento lineal del carro en eje "x" se

genera una trayectoria tipo rampa para ello se considera la ecuación.

- $x_{des(actual)} = x_{des(anterior)} + v_{des(anterior)} * T_s$
- x_{des} = posición deseada x
- v_{des} = velocidad deseada
- T_s = Tiempo de muestreo virtual

Para aplicar la ecuación anterior se consideró

- T_{rise} = Tiempo de la rampa = 1
- pos_{des} = Posición deseada = dato de referencia del usuario
- $V_{rise} = (pos_{des}) / T_{rise}$

Relaciones Matemáticas del Controlador

Se considera las siguientes relaciones:

1. t_t = tiempo del controlador coincidente con el de la simulación del modelo .
2. p_p = posición particular en el eje x
3. $v_p = (p_p - p_{pant}) / t_s$; velocidad para posición particular x
4. $a_{sex} = X(i, 4) * 57,2958$,posición angular para una posición x particular en radianes.
5. $w_{sex} = X(i, 5) * 57,2958$;velocidad angular para una posición x particular en radianes/seg

Cálculo de las señales de error

Las señales de error para la posición x ,la señal error de la velocidad para el carro y señal error de la posición angular del brazo θ se expresa:

1. $e_d = K_x * (x_{des,pp})$, señal error de posición en x .
2. $ed_d = K_{xd} v_{des} - V_p$, error de velocidad en x
3. $e_q = (ang_{des}) - a_{sex}$, error de posición angular θ

donde:

- K_x = constante de adaptación de error de posición lineal
- K_{xd} = constante de adaptación de velocidad
- p_p = posición x medido del carro
- x_{des} = posición deseada del carro
- V_p = velocidad de posición lineal = $p_p - p_{pan} / T_s$

Evaluación del Controlador Fuzzy En Tiempo Continuo

Para implementar y evaluar el controlador fuzzy se ha realizado los cálculos para obtener la señal de control como voltaje para el motor de posición lineal ó denominado V_d (voltios) y para el motor de posición angular del brazo denominado V_a (voltios) .

Cálculo de la señal de control para la posición lineal del carro: V_d

Para obtener la señal de control de posición x denominado V_d se ha considerado los siguientes parámetros

1. e_d = error de posición
2. ed_d = error de velocidad
3. $distance$ = función de membresía de posición
4. $distance_d$ = función de membresía para velocidad
5. BR_r = base de reglas de posición lineal
6. $Voltaje_r$ = base de reglas para la salida como voltaje

La función que realiza la evaluación es:

$$v_d = \text{FLC} - t(e_d, ed_d, distance, distance_d, RB_r, voltaje_r)$$

Esta función determina el valor de la tensión para controlar al motor que desplaza el carro .

Cálculo de la señal de control para la posición angular del brazo: V_q

Para obtener la señal de control de posición angular θ denominado V_q se considera los siguientes parámetros.

1. e_q = error de posición angular
2. w_{sex} = error de velocidad angular
3. $angle$ = función de membresía de posición angular
4. $angle_d$ = función de membresía para velocidad angular
5. BR_q = base de reglas de posición angular
6. $Voltaje_q$ = base de reglas para la salida como voltaje de posición angular

La función que realiza la evaluación es:

$$v_q = \text{FLC} - t(e_q, -w_{sex}, angle, angle_d, RB_q, voltaje_q)$$

Adaptación de los niveles de tensión de las señales de control

La señal de control para la posición lineal y angular se debe realizar una adaptación, luego se considera las constantes de adaptación para adecuar el nivel para cada caso:

- **Adaptación de la señal de control de posición Lineal- salida del controlador de posición lineal: u_1**

La ecuación matemática para adaptar la señal de control de posición angular se considera la ecuación:

$$u_1 = K_{propd} * V_d$$

donde:

1. K_{propd} = constante de adaptación de posición de salida
2. V_d = voltaje de control para la posición lineal del brazo

- **Adaptación de la señal de control de posición angular- salida del controlador de posición angular: u_2**

La ecuación matemática para adaptar la señal de control de posición angular se considera la ecuación (3.52)

$$u_2 = (K_{prop_a}) * (V - a) - Kg * \sin(\theta) \quad (3.52)$$

donde:

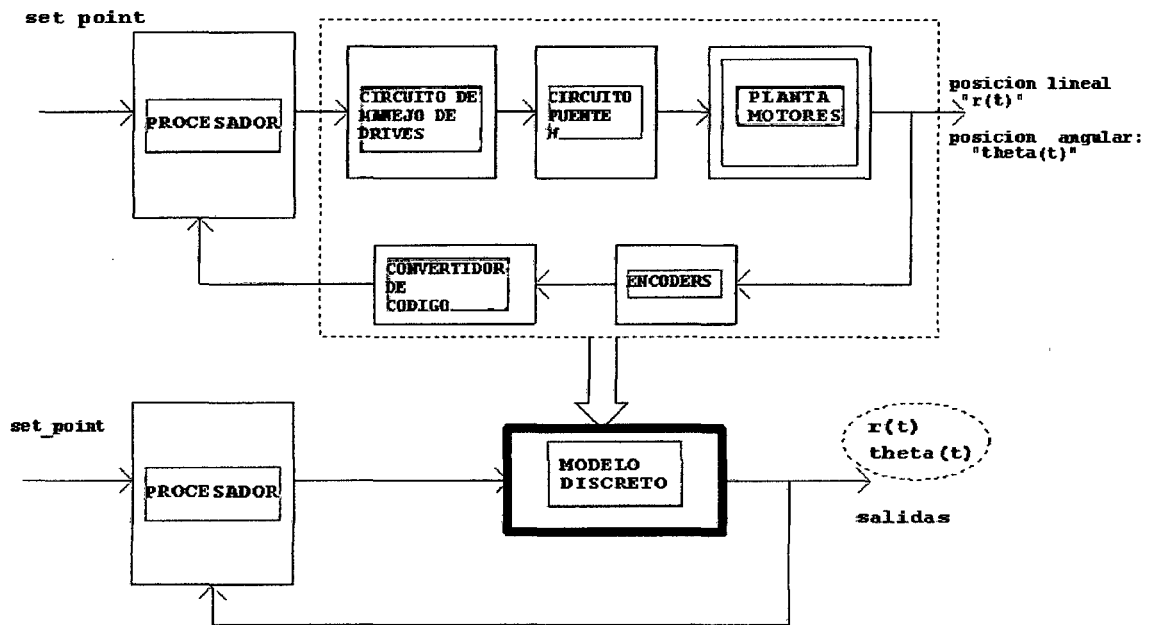
K_g = es la constante de componente de gravedad de la varilla acoplado al eje del motor, esta dado por la ecuacion (3.53)

$$Kg = \frac{L * m * g * r_a}{2 * K_m} \quad (3.53)$$

3.3.4. Diseño del control difuso en tiempo discreto

Esta sección se muestra la discretización del sistema de control de tiempo continuo y se presenta el diagrama de bloques del sistema figura 3.16

Figura 3.16: Sistema de control en tiempo discreto



El sistema dispone de dos servomotores DC (modelos iguales) con encoder óptico incremental de 1250 cuentas por vuelta con salida a dos canales en cuadratura.

El sensado tiene el decodificador de cuadratura que se encarga de leer los datos del encoder óptico para emitir salidas *UP/DOWN* luego conectar al puerto de entradas digitales del DSP320LF2812.

El codificador como la mayoría de los casos, incluye dos pistas de marcas o ranuras en cuadratura (canales A y B), cada una asociada a una fuente de luz y un fotodetector independientes. Los canales A y B producen dos ondas cuadradas desfasadas en 90 grados entre sí que se utilizan para determinar la dirección del movimiento. Por ejemplo si el canal B adelanta al canal A, el eje estará girando en el sentido antihorario y viceversa. Si no se utiliza una detección de cuadratura, la vibración puede causar el envío de pulsos con error al contador. El codificador usado tiene una resolución de 100 ppr (pulsos por revolución) para cada motor se considera las siguientes relaciones:

I.Sensor motor1:Posición lineal

- $npulse_encoder1=512$; número de pulsos del encoder para la posición.
- $qeq = 2$; multiplicador del QEP.
- $dpolea=0.084*2$; diámetro de la polea.
- $m2pulse=npulse_encoder1*qeq*19.745/(pi*dpolea)$; número de pulsos considerando factores

II.Sensor motor2 : Posición angular

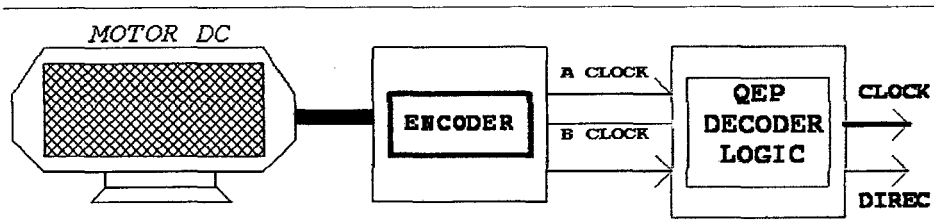
- $npulse_motor2=512$; número de pulse de encoder2 para la posición.
- $ls714=4$; multiplicador del tacómetro 2.
- $gsex2pulse=npulse_encoder2*ls7184/360$; número de pulsos considerado factores

Muestreo de las señales sensadas

Se considera la frecuencia de muestreo fsc ó $tsc = 1/fsc$ las relaciones de velocidad del motor 1 y del motor 2 serán.

- $V2pulse= npulse_encoder1*qeq*5.9/(pi*polea)*tsc$.

Figura 3.17: Sensores de motores 1 y 2



- $wsex2pulse=npulse_encoder2*ls7184/360*tsc;$

Control difuso de posición del carro lineal y de posición angular del brazo

En esta etapa se convierte todas las variables de magnitudes físicas a número de pulsos (desritizar) luego se tiene :

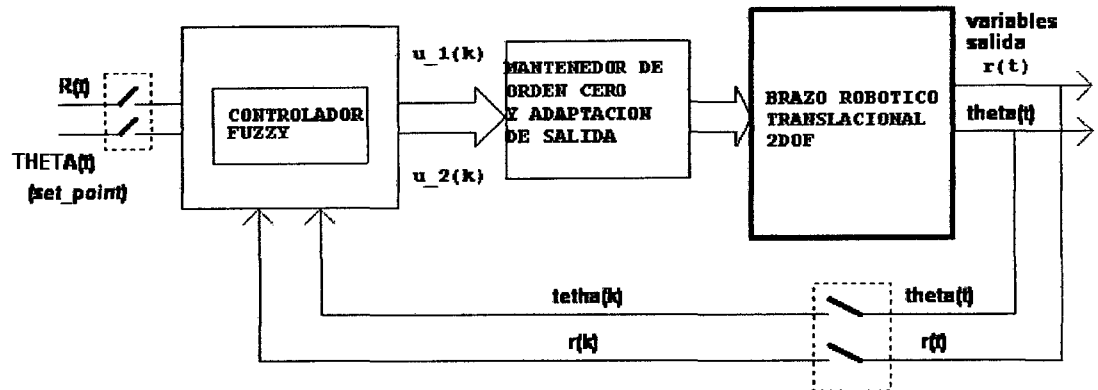
- $x(k)$ posición lineal del carro actual.
- $x(k - 1)$ posición anterior del carro
- $v(k) = x(k) - x(k - 1)/ts$.
- $\theta(k)$ posición angular del brazo o de la varilla
- $w(k)$ velocidad angular

Relación de las señales de error en tiempo discreto

Las ecuaciones que se tienen para determinar las señales de error del sistema que son señales de entrada al controlador difuso se expresan:

1. $e_d(k)=Kx*(x_des(k)-pp(k));$ error de posición en x .
2. $x_des(k)= m2pulse*x_des.$
3. $e_{d-d}(k)=Kxd*(v_des_(k)-Vp_(k));$ error de velocidad lineal en x .
4. $v_des(k)=v_des*v2pulse.$
5. $e_\theta(k)=\theta_des(k) - \theta(k);$ error de posición angular θ .

Figura 3.18: Señales del Controlador logico difuso entrada y salida



En el gráfico se muestra las señales de entrada y salida del controlador difuso discreto figura 3.18

Funciones de membresía del controlador difuso de posición lineal del carro y posición angular del brazo

Considerando las funciones de membresía del sistema de control continuo , para el sistema discreto se tiene que realizar la transformación de la funciones de membresía del sistema continuo al universo de números enteros.

Constantes de transformacion

Las constantes de transformación para la posición angular y lineal son:

1. Secuencia de pulsos del motor1 para posición lineal denominado: $m2pulse$
2. Secuencia de pulsos del motor2 para la posición angular denominado : $gsex2pulse$.
3. Velocidad de pulsos para posición lineal denominado: $V2pulse = m2pulse * Tsc$.
4. Velocidad de pulsos para la posición angular denominado: $Wsex2pulse = gsex2pulse * tsc$.

Parámetros para calcular las constantes

Los parámetros son los siguientes :

1. Número de pulsos del encoder del motor1 denominado $npulse_encoder1=512$.
2. Número de pulsos del encoder del motor2 denominado $npulse_encoder2=512$.
3. Diámetro de la polea de la faja : $dpolea= 0.048*2$.
4. Relación de engranaje para la posición $n1=19.745$.

Tamaños de las funciones de membresía

Se tiene que determinar el tamaño de las funciones de membresía de las entradas del controlador difuso de posición lineal del carro y de la posición angular del brazo de la varilla, luego podemos indicar lo siguiente:

1. Tamaño de la función de membresía de posición : $sizep = size(distance); sisep = sisep(1, 1)$.
2. Tamaño de la función de membresía de velocidad : $sizev = size(distance_d); sisev = sisev(1, 1)$.
3. Tamaño de la función de membresía de posición angular : $sizea = size(angle); sisea = sisea(1, 1)$.
4. Tamaño de la función de membresía de velocidad angular: $sizew = size(angle_d); sizew = sizew(1, 1)$.

Para discretizar las funciones de membresía se debe conservar la última columna de cada función de membresía (FM) luego se tiene.

- $distance_pulsos(:,3) = distance(:,3)$.
- $distance_d_pulsos(:,3) = distance_d(:,3)$.
- $angle_pulsos(:,3) = angle(:,3)$
- $angle_d_pulsos(:,3) = angle_d(:,3)$

Después se transforma el offset ("C") de cada función de membresía (MF) ; entonces debe realizarse las siguientes relaciones.

- $distance_pulsos(:,2) = distance(:,2) * m2pulse$.
- $distance_d_pulsos(:,2) = distance_d(:,2) * v2pulse$

- $\text{angle_pulsos}(:,2) = \text{angle}(:,2) * \text{gsex2pulse}$
- $\text{angle_d_pulsos}(:,2) = \text{angle_d}(:,2) * \text{wsex2pulse}$

Luego debe transformarse las ganancias de cada función de membresía mediante las siguientes relaciones.

- $\text{distance_pulsos}(K,1) = \text{distance}(k,1) * \text{m2pulse} * 10000 / \text{m2pulse}$, donde k varía de 1 a 5.
- $\text{distance_d_pulsos}(k,1) = \text{distance_d}(k,1) * 10000 / \text{v2pulse}$, donde K varía de 0 a 5
- $\text{angle_pulsos}(k,1) = \text{angle}(k,1) * 10000 / \text{gsex2pulse}$, donde k varía de 0 a 5
- $\text{angle_d_pulsos}(k,1) = \text{angle_d}(:,2) * 10000 / \text{wsex2pulse}$, k varía de 0 a 5

Evaluación del controlador fuzzy discreto

Las señales de salida del controlador fuzzy para la posición lineal del carro y de posición angular de la varilla se obtiene considerando las relaciones siguientes.

I.voltaje para la Posición lineal del carro-Salida del controlador fuzzy

Se determina mediante la siguiente función.

- $V_d = \text{FLC_k1}\{(\text{ed}(i, 1), 0 * \text{ed_d}(i, 1), \text{distance_pulsos}, \text{distance_d_pulsos}, \text{RB}_r, \text{Voltaje}_r)\}$

Donde los parámetros para evaluar son.

1. $e_d(i,1)$ error de posición discreto lineal .
2. $e_d_d(i,1)$ =error de velocidad lineal
3. distance_pulsos = función de membresia de posición lineal
4. distance_d_pulsos = funcion de membresia de velocidad lineal.
5. RB_r = bases de reglas para posición lineal del carro
6. Voltaje_r = bases de reglas como voltaje de salida para posición lineal

Donde el V_d es el voltaje de salida del controlador fuzzy y el voltaje para actuar sobre el proceso está dado por:

- $u_1 = K_{prop} * V_d$, voltaje para controlar el motor 1

II. voltaje para la posición angular de la varilla -Salida del controlador fuzzy

Para calcular el voltaje de salida del controlador fuzzy para la posición angular del brazo está dado por la función.

- $V_a = FLC_k1(e_{q(i,1)}, 0, angle_pulsos, angle_d_pulsos, RB_q, Voltaje_q)$

los parámetros para evaluar son:

1. $e_{q(i,1)}$ = error de posición angular discreto.
2. $e_{q-d(i,1)}$ = error de velocidad angular = 0
3. $angle_pulsos$ = función de membresía de posición angular
4. $angle_d_pulsos$ = función de membresía de velocidad angular.
5. RB_q = bases de reglas para posición angular del brazo
6. $Voltaje_q$ = función de membresía del voltaje de salida para posición angular

La adaptación de los niveles de tensión al motor 2 tiene la siguiente relación.

- $u_2 = 0,5 * V_a - 0,1398 * sen\theta$, para controlar el motor 2
- u_g = voltaje de compensación de gravedad

3.4. Metodología de cada momento de la investigación

La metodología empleada es la siguiente:

- Modelar matemáticamente el *BRT2DOF*, y efectuar su simulación en lazo abierto.
- Diseñar y simular la estrategia de Control fuzzy en lazo cerrado en tiempo continuo.

- Diseñar y simular la estrategia de Control fuzzy en lazo cerrado en tiempo discreto.
- Efectuar las conclusiones en base a los resultados obtenidos y efectuar algunas recomendaciones para trabajos futuros.

3.5. Operacionalización de las variables

En función a las variables dependientes e independientes, al operacionalizar se determinan los indicadores de cada variable ; posición lineal del brazo y posición angular del brazo.

3.6. Población y muestra

Por el tipo de investigación no corresponde determinar el tamaño de la muestra.

3.7. Técnica e instrumentos de recolección de datos

Debido a que no se ha determinado el tamaño de la muestra, no corresponde utilizar las técnicas e instrumentos para recolección de datos.

3.8. Procedimientos de recolección de datos

No corresponde al presente trabajo.

3.9. Procedimiento estadístico y análisis de datos

No corresponde utilizar procesamiento y análisis de datos.

Capítulo IV

IV. RESULTADOS

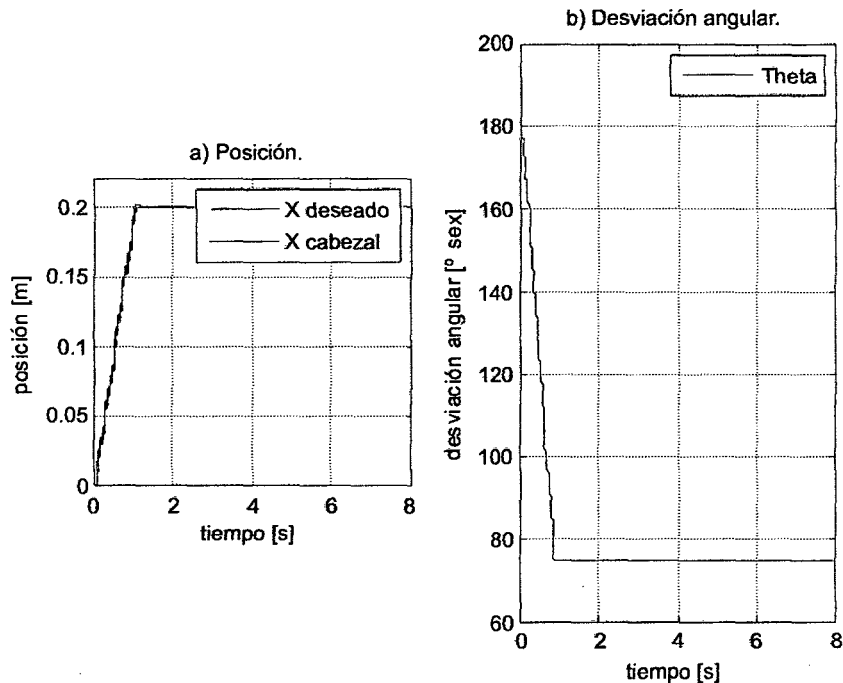
4.1. Resultados parciales

4.1.1. Resultados del sistema de control difuso en tiempo continuo en lazo cerrado

Mediante el programa en Matlab, denominado `mrtm_FLC_c.m`, nos permite obtener la respuesta mostrada en la figura 4.1. el programa `mrtm_FLC_c.m` requiere de las siguientes funciones:

- `FLC_t.m` : Que a su vez necesita de: `MF_t.m`
- `f_mrtm2dof`: funcion que determina la aceleración para la posición y ángulo

Figura 4.1: Posición lineal y angular del brazo robótico en lazo cerrado tiempo continuo



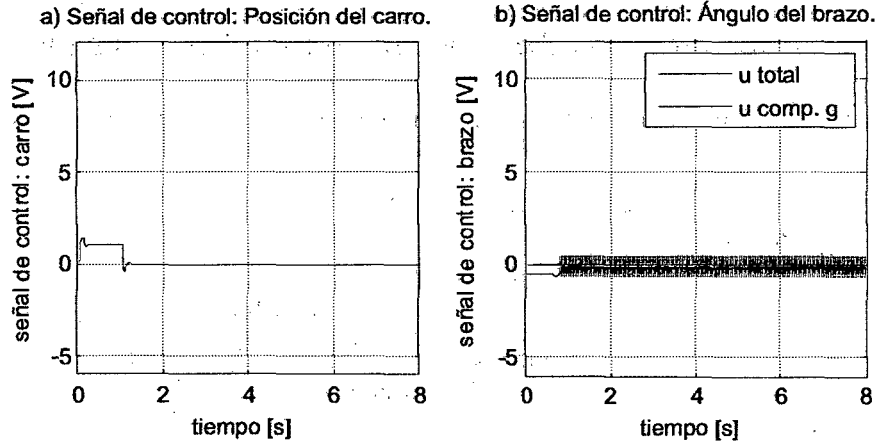
La evaluación se realizó considerando los valores de referencia:

- $X_{deseado} = 0,2m$

- $\theta_{deseado} = 75\text{grados}$

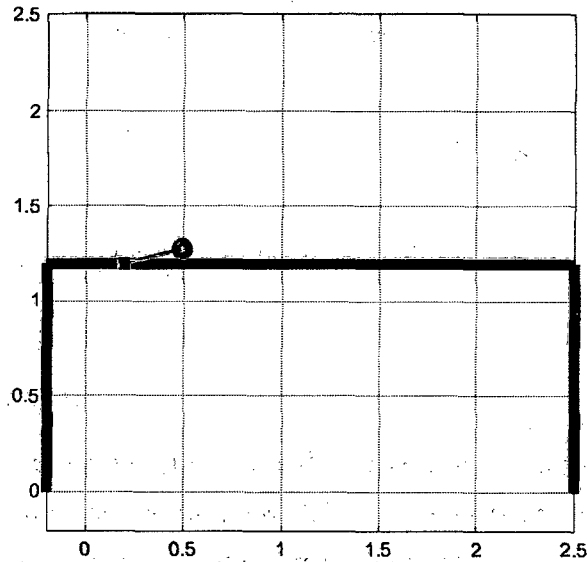
El programa también nos muestra la señales de control de posición lineal del carro o brazo y la señalal de control para la posición angular del brazo. ver figura 4.2. El programa incluye la simulación en forma animada del sistema tal como se

Figura 4.2: Señales de control de posición lineal y angular del brazo robótico



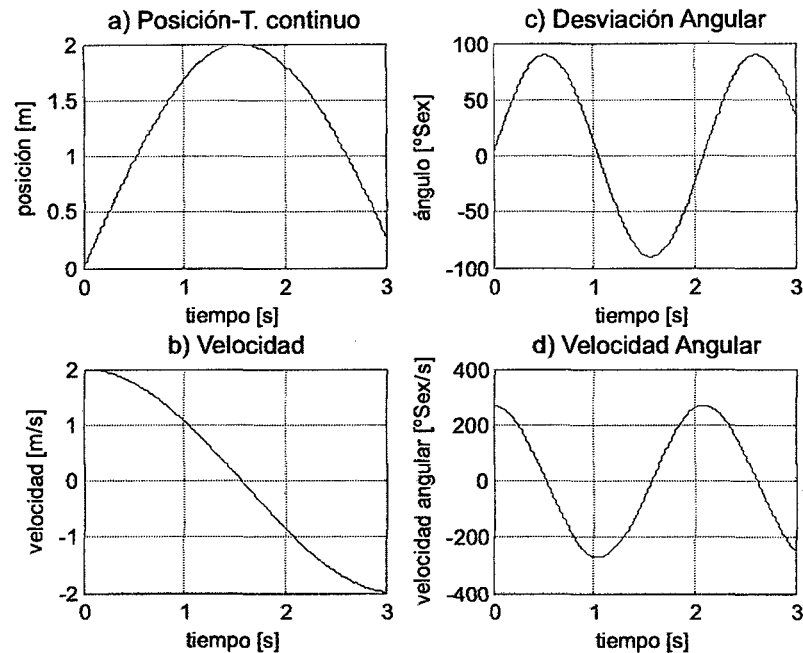
muestra en alfigura 4.3

Figura 4.3: simulación del sistema en forma animada



Se considera como referencia las señales en tiempo continuo la función senoidal para la posición lineal y angular del brazo tal como se muestra en la figura 4.6

Figura 4.8: Señales de los sensores en tiempo continuo para posición lineal y angular del brazo



En la figura 4.9 se muestra las señales discretas de los sensores de posición lineal y angular para la frecuencia de muestreo $f_{sc} = 50HZ$.

En la figura 4.10 se muestra las señales discretas de los sensores de posición lineal y angular para la frecuencia de muestreo $f_{sc} = 150HZ$.

4.2. Resultados Finales

De los resultados parciales podemos concluir que nos permite evaluar el desempeño del sistema en lazo abierto y lazo cerrado en tiempo continuo y discreto considerando la estrategia de control fuzzy.

En la figura 4.11 se muestra la evaluación del sistema en lazo abierto; en la figura ve que la variable desplazamiento lineal y angular crece indefinidamente, el sistema es incontrolable.

En la figura 4.1 y figura 4.4 el sistema es de lazo cerrado en tiempo continuo y discreto las variables desplazamiento lineal y angular del brazo sigue a la referencia

Figura 4.9: Señales de los sensores en tiempo discreto

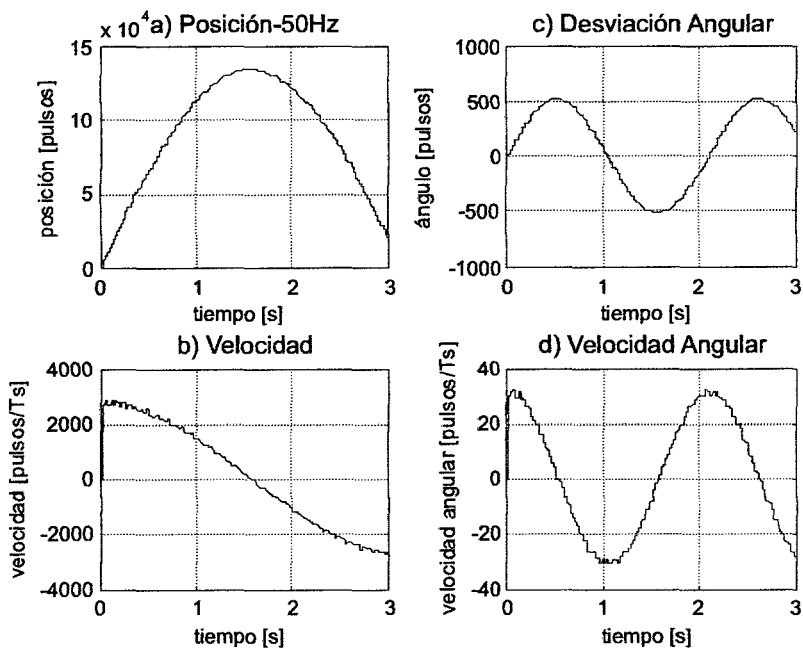
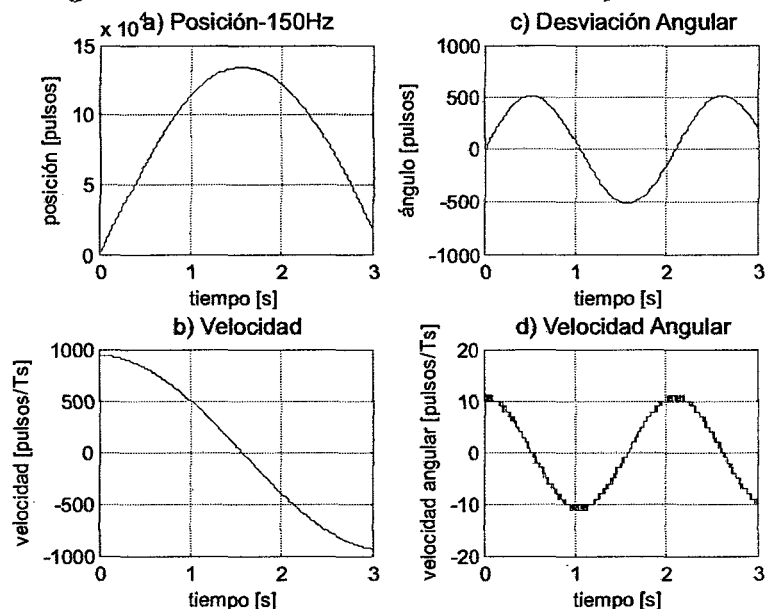


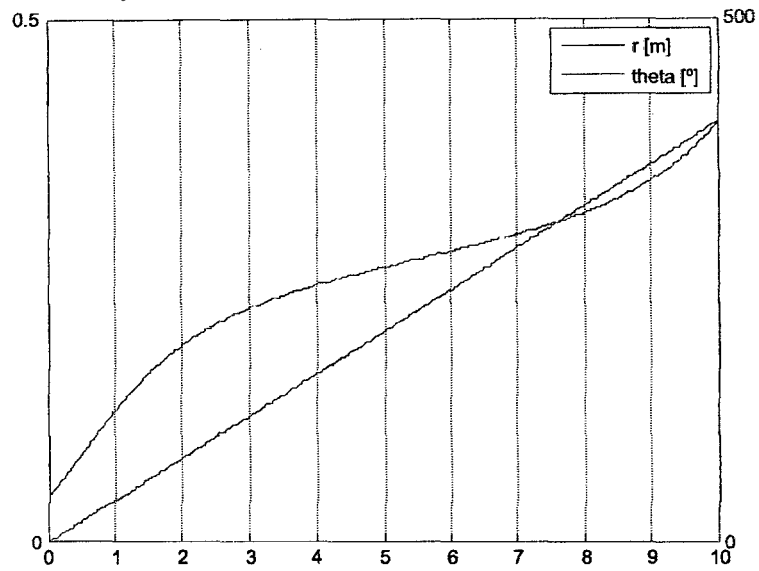
Figura 4.10: Señales de los sensores en tiempo discreto



, el error prácticamente es cero y el tiempo de respuesta transitoria es menor a 1 seg.

Con respecto al comportamiento de las señales de control para la posición lineal del brazo y angular se muestra en la figura 4.2 y 4.5 que inicialmente es grande y luego tiende a cero cuando la variable controlada y deseada son iguales.

Figura 4.11: sistema de control en lazo abierto



El tiempo transitorio $T_s = 0,95s$

La frecuencia de muestreo vemos que el sistema se comporta mejor para una frecuencia $F_{sc} = 100Hz$. con respecto a la frecuencia de 50Hz.

Capítulo V

V. DISCUSIÓN Y RESULTADOS

5.1. Contrastación de hipótesis con los resultados

De los resultados mostrados en el cuadro 4.1, se puede observar que la técnica de Control Fuzzy es el adecuado para esta aplicación, ya que se cumple con los requerimientos de funcionamiento, con un sobrepico nulo, y un tiempo de establecimiento 0.95seg, cumpliéndose con lo previsto. En tal sentido, se ha cumplido con la hipótesis planteada en el trabajo, el cual paso a anotar:

”La aplicación de estrategias de control fuzzy permitirán controlar la posición lineal y angular del Brazo Robótico translacional de segundo orden denominado BRT2DOF a niveles de referencia deseados, y error estacionario nulo en tiempo continuo y tiempo discreto.

5.2. Contrastación de resultados con otros estudios similares

En esta sección mostramos las simulaciones del sistema de control por modo deslizante para el sistema BRT2DOF y las simulaciones del sistema de control por lógica difusa. En la primera parte se encuentra el desarrollo de las simulaciones utilizando la ley de control original, es decir, sin aplicar el método de supresión de chattering.

Todas las simulaciones del sistema controlado son respuesta a una referencia arbitraria constante, con el propósito de realizar una mejor comparación de los resultados de las simulaciones. El desarrollo de las simulaciones se estructuran en tres experimentos:

5.2.1. Experimento

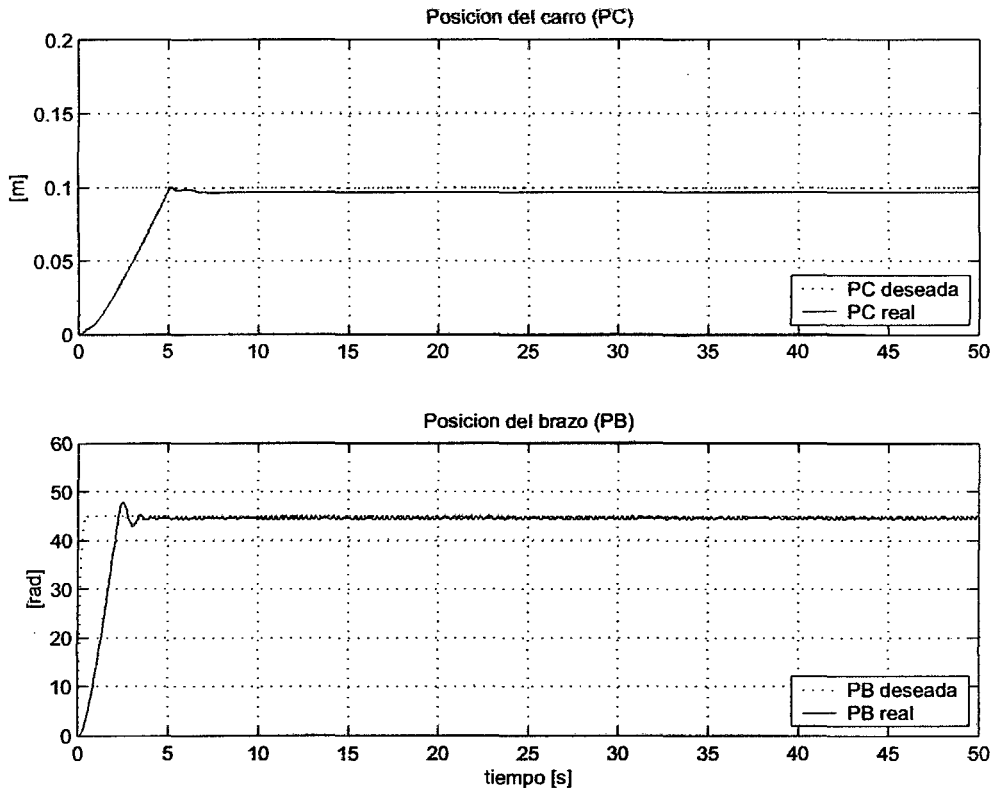
I.-sistema de control por modo deslizante para el sistema BRT2DOF

Las condiciones iniciales son $x_1 = 0\text{m}$ para el carro y $x_2 = 0\text{rad}$ para la posición angular del brazo, estas condiciones serán utilizadas para los demás experimentos.

Las trayectorias constantes deseadas para el carro y para el brazo son $x_{d1} = 0,1\text{m}$ y $x_{d2} = \pi/4\text{rad}$ respectivamente. El tiempo utilizado en la prueba es de 50 seg.

La figura 5.1 muestra las salidas controladas para la posición del carro y posición angular brazo, luego de un tiempo de 4.5 seg y 2.5 seg respectivamente. Se observa que en la posición del carro y posición del brazo no existe sobrepaso, debido al incremento de los parámetros de sintonía c_1 y c_2 que se realizó en la experiencia en tiempo real.

Figura 5.1: Control de posición para en carro y el brazo.



La figura 5.2 muestra la respuesta del sistema para la velocidad del carro y velocidad angular del brazo, en ambos casos la velocidad decae al valor cero en un tiempo breve de 4.5 seg y 2.5 seg respectivamente. La oscilación que esta presente a

medida que el sistema se controla se debe a que los motores están ejerciendo fuerza de control a medida que el sistema se estabiliza hasta encontrar la posición deseada.

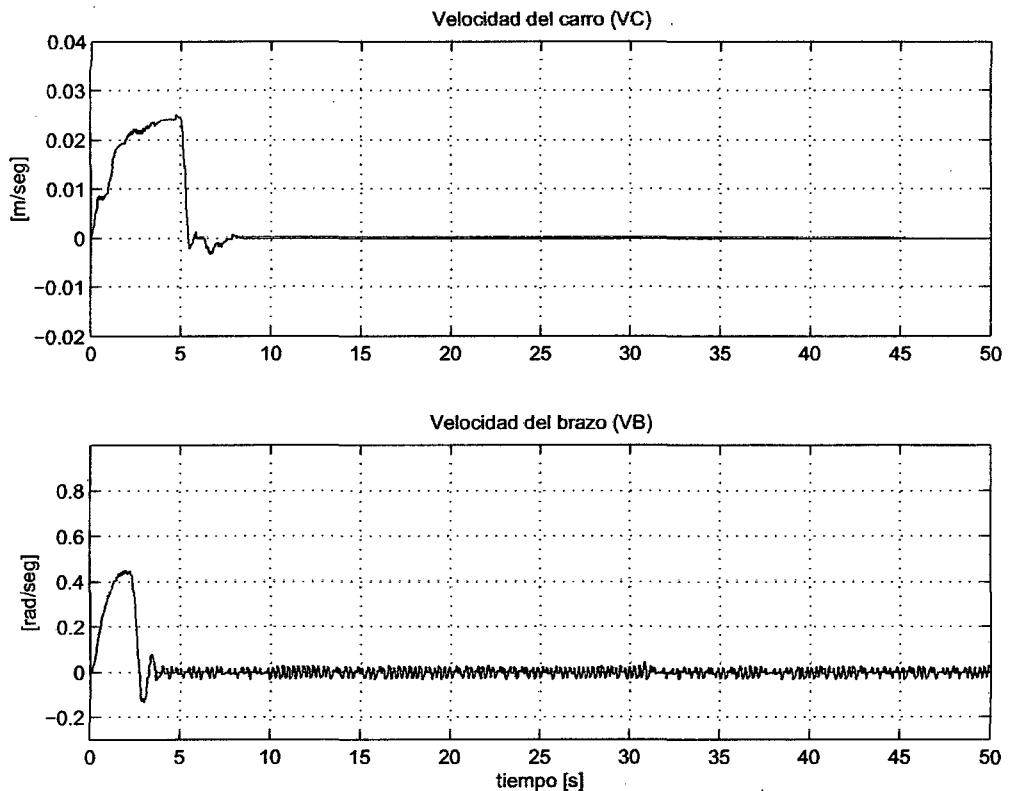


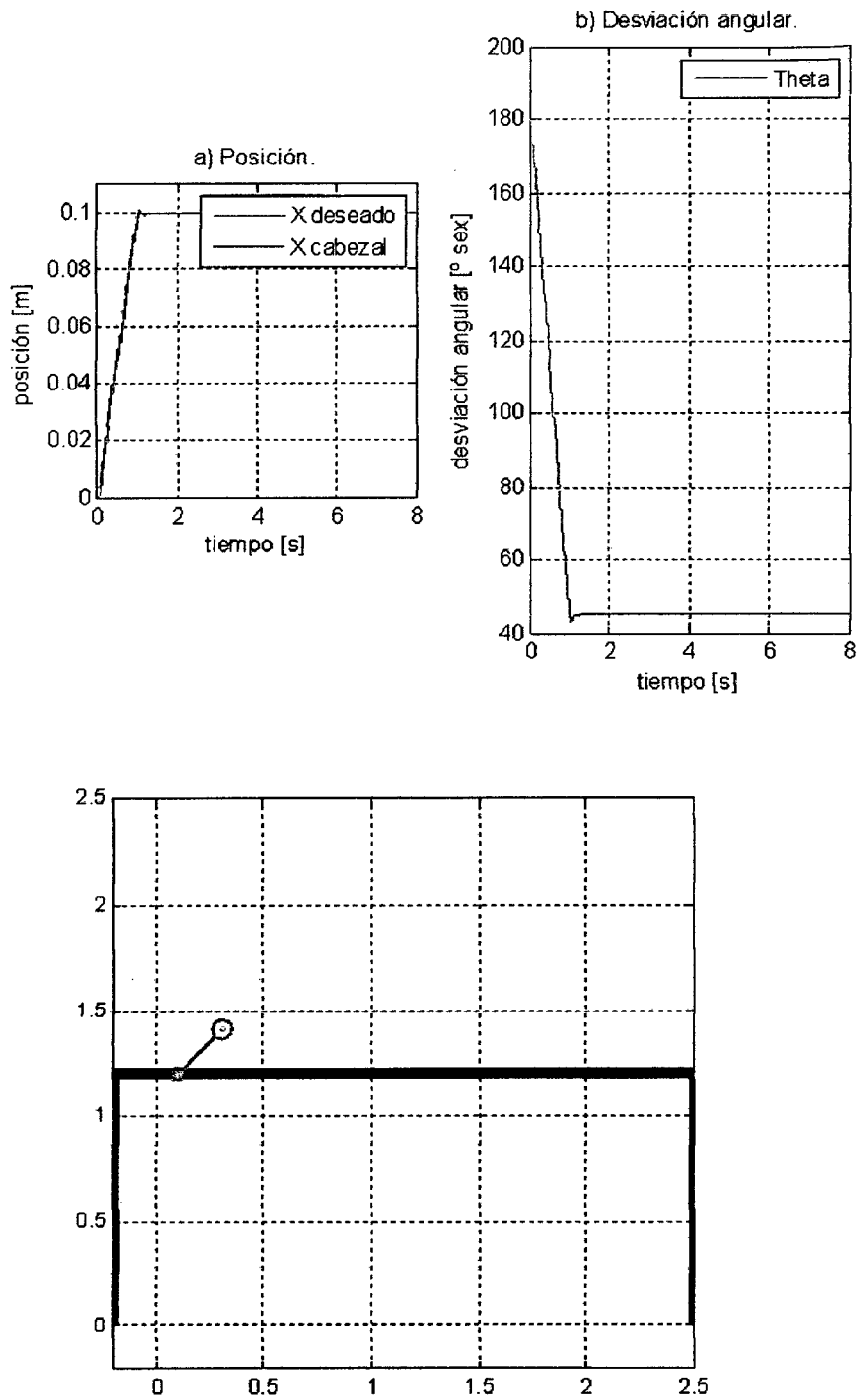
Figura 5.2: Curva de la velocidad para en carro y velocidad del brazo.

II.-Sistema de control fuzzy para el sistema BRT2DOF

Las trayectorias constantes deseadas para el carro y para el brazo son $x_{d1} = 0,1m$ y $x_{d2} = \pi/4rad$ respectivamente. El tiempo utilizado en la prueba es de 8 seg. La figura 5.3 muestra las salidas controladas para la posición del carro y posición angular, y la figura animada del sistema brazo, luego de un tiempo 8 seg respectivamente. Observando las respuestas de ambos sistemas se puede concluir que el sistema con control fuzzy para la posición del carro es $t_s = 0.95$ seg , para el sistema de control con ley modo deslizante es $t_s = 5$ seg. luego podemos concluir que el sistema con control difuso es más rápido

Las respuesta del sistema para la posición angular , el sistema con control fuzzy es mas rápido; el error en estado estacionado del sistema con control modo deslizante

Figura 5.3: Curva de la posición lineal y angular BRT2DOF.



es mayor.

Luego se indica que el sistema con control fuzzy es más eficiente por las consideraciones indicadas

CONCLUSIONES

Concluimos que en esta tesis se presentan los conceptos fundamentales de la lógica difusa, luego se realiza el modelo matemático del sistema a controlar, se realiza las simulaciones en lazo abierto y lazo cerrado para tiempo continuo y luego para discreto considerando diferentes entradas de referencia tipo escalón,

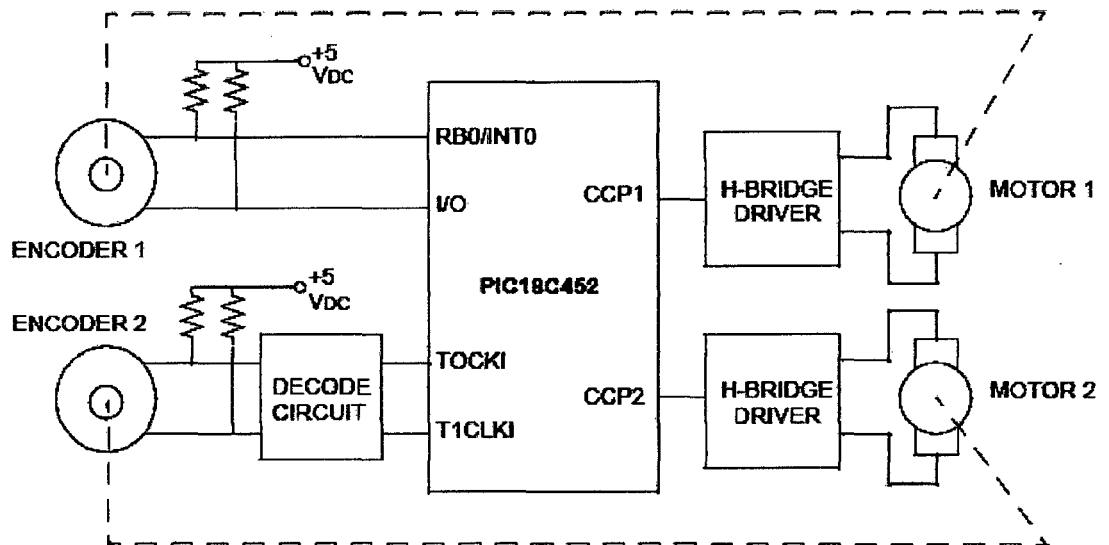
Las conclusiones de la tesis son:

1. De los resultados de simulación se puede observar que el controlador lógico difuso diseñado, presenta un buen desempeño, como se puede apreciar en la figura 4.1 La repuesta del sistema de lazo cerrado tiempo continuo , a los cambios de referencia, son rápidos; así mismo para el sistema de control difuso discreto ,tal como se observa en al figura 4.4 la respuesta completa no presenta error en estado estable.
2. El controlador por lógica difusa diseñado resuelve el problema de control propuesto, obteniendo para todas las simulaciones, mostradas en el capítulo 4, desempeños satisfactorios con sobreimpulsos nulos , tiempos de establecimiento aproximados de 1 seg: y errores en estado estable muy cercanos a cero.
3. Como pruebas complementarias de simulación del controlador difuso se aplicaron otras señales de prueba como son las señales senoidal ,rampa , las cuales realizan un buen seguimiento de la señal de referencia, aunque se presenta un ligero error en estado estacionario anexo C.
4. Incorpora lenguaje común al diseño
5. No requiere un modelo riguroso de la planta.
6. Es fácil realizar modificaciones al controlador.
7. Puede llegar a ser lento debido al muestreo.
8. Depende de la aplicación.

RECOMENDACIONES

Se recomienda la implementación del sistema BRT2DOF considerando la figura 5.4.

Figura 5.4: Diagramas de bloques de la implementación del sistema.



Donde los componentes físicos está constituido por dos servomotores (modelos iguales) con encoder óptico incremental de 1250 cuentas por vuelta con salida a dos canales en cuadratura.

Los actuadores se implementa con dos tarjetas similares basados en el drive L6302, este driver integra un puente H completo realizado en Multipower-BCD tecnología que combina transistores de potencia aislados DMOS con CMOS y Bipolar circuitos en el mismo chip. Mediante el uso de tecnología mixta que ha sido posible optimizar los circuitos de la lógica y la etapa de potencia a lograr el mejor rendimiento posible. la Transistores DMOS de salida puede operar en el suministro voltajes de hasta 42V y 5 amperios y de manera eficiente a altas frecuencias de conmutación velocidades. Todas las entradas lógicas son TTL, CMOS y MC compatibles. Cada canal (medio puente) de el dispositivo es controlado por una entrada lógica por separado, mientras que los controles comunes permiti a ambos canales. ver figura 5.5

En particular, este sistema BRT2DOF se recomienda implementarlo tanto en el PIC18F4550 y PIC16F877 por el costo y también porque disponen de los recursos

REFERENCIALES

- [1] Acosta ,Gerardo.. **Taller de Controlador Basado en Lgica Difusa..** *Universidad Nacional del Centro de la Provincia de Buenos Aires. Publicación Independiente, Noviembre 2001.*
- [2] *Angulo Usategui,Jose María,Aritz Etxebarria Ruiz,Ignacio Angulo Martinez y Ivaán Trueba Parra.* **dsPIC Diseño Práctico de Aplicaciones.** McGRAW-HILL/INTERAMERICANA S.A.U. España -Madrid, 2006.
- [3] Benajamin C. Kuo. **Sistemas de Control Digital.** *Prentice Hall Hispanoamericana S.A. México,New York ,1997.*
- [4] *Bonifacio Martin del Rio y Alfredo Sanz Molina.* **Redes Neuronales y Sistemas Difusos. 2a. edicion ampliada y actualizada.** AlfaOmega grupo editor S.A. México, 2002.
- [5] Garcia Breijo,Eduardo . **Compilador C CCs y Simulador PROTEUS para Microcontroladores PIC.** , *AlfaOmega grupo editor S.A.de C.V. México, 2008.*
- [6] *Lofti A. Zadeh.* **Fuzzy sets Department of electrical Engineering research laboratory.** University of California, Berkeley California, 1965.
- [7] Lofti A. Zadeh. **Fuzzy sets.** *Information and Control, Vol 8, No.3 pp.338-353, june 1965.*
- [8] *Ogata, Katsuhiko.* **Ingeniería de Control Moderna.** Prentice Hall,Hispanoamericana S.A. México,New York ,1990.
- [9] Ogata, Katsuhiko. **Sistemas de Control en Tiemp Discreto.** *Compañía Editorial Continental S.A. México ,1999.*
- [10] *Ollero Bayurone,Aníbal.* **Control Por computadora Descripcion interna y diseño óptimo.** Alfaomega S.A. de C.V. S.A. México ,1991.

- [11] Passino M.K., Yurkovich S. **Fuzzy Control**,. *Addison-Wesley, USA,1998.*
- [12] Reynoso Oscar ,José María Sebastian , Zuñiga y Rafael Aracil Santoja y Fernando Torres Medina . **Control De Sistemas Discretos**. McGRAW-HILL/INTERAMERICANA S.A.U. España -Madrid, 2004.
- [13] Rojas Moreno, Arturo. **Control Avanzado-Diseño y Aplicaciones en Tiempo Real**. *Universidad Nacional de Ingeniería ,Publicación Independiente,Lima- Perú, 2001.*
- [14] Mohammad Jamshidi,. **Fuzzy logic and control Software and hardware applications**,. Ed. Prentice Hall, Estados Unidos,1994.
- [15] Ying, Hao. **Fuzzy Control and Modeling. Analytical Foundations and Applications**.. *IEEE Press. 2000*

ANEXO A

MATRIZ DE CONSISTENCIA

Título: “DISEÑO DE UN SISTEMA DE CONTROL POR LÓGICA DIFUSA APLICADO A UN BRAZO ROBÓTICO TRANSLACIONAL DE 2DOF”

PROBLEMAS	OBJETIVOS	HIPOTESIS	VARIABLES	MÉTODOS
<p>General</p> <p>El sistema Brazo Robótico Translacional de 2DOF (BRT2DOF) es un sistema no lineal e inestable (ya que el Brazo Robótico Transnacional es inestable), lo cual con lleva a problemas en la modelación matemática y fundamentalmente en el control.</p> <p>Específico</p> <p>Construcción del sistema de control utilizando lógica difusa y su aplicación del sistema utilizando otras leyes de control, en el área de control.</p>	<p>Objetivo general</p> <p>Controlar el sistema Brazo Robótico Translacional utilizando lógica difusa de manera que el sistema de control funcione adecuadamente.</p> <p>Objetivo específico</p> <p>a. Diseñar, simular y construir el sistema de Control aplicado a un Brazo Robótico Translacional utilizando lógica difusa de manera que el sistema de control funcione adecuadamente a pesar de variaciones en el modelo matemático del sistema.</p> <p>b. Determinar el estado del arte y los problemas en el control de un sistema no lineal, en general, y de un Brazo Robótico Translacional , en particular</p> <p>c. Documentar las prácticas, métodos, modelos, resultados, análisis y diseño</p>	<p>Un control difuso no causará inestabilidades en el Brazo Robótico Translacional (BRT), la ley de control por lógica difusa funciona mejor en sistemas no lineales en este caso el sistema denominado Brazo Robótico Translaciona de 2DOF1 (BRT2DOF), permitirá controlar dentro del margen de valores, las variables de salida posición angular del brazo ó Varilla ('θ') y posición lineal del Brazo ó posición del carro (r)</p>	<p>Variables dependientes:</p> <ul style="list-style-type: none"> • Posición angular del Brazo "θ" , esta variable controlada posiciona la varilla un ángulo \square • Posición Lineal del brazo "r" esta variable dependiente controlada posiciona en el eje horizontal la posición del carro <p>Variables independientes: son los voltajes aplicados a los motores M1 y M2</p> <ul style="list-style-type: none"> • μ_1 : <i>voltaje...de...motor1</i> • μ_2 : <i>voltaje...de...motor2</i> 	<p>General</p> <p>A través de este método se conocerá el diseño de un sistema de control con lógica difuso</p> <p>Específicos</p> <p>Se aplicara los métodos de investigación : Experimental y Matematizacion (modelación matemática del sistema de control considerando la ley fuzzy Logic)</p>

Identificación de Parámetros

Por identificación se entiende la formulación de un modelo que permita representar al sistema con todas sus propiedades y características relevantes. En nuestro caso determinaremos los parámetros de los servomotores para el sistema BRTM de 2DOF, estos parámetros hallados, nos permitan validar los resultados experimentales. En la sección desarrollaremos el modelo matemático del sistema para realizar la identificación, basándonos en el análisis del servomotor DC expuesto en la sección anterior. Un modelo matemático es un conjunto de ecuaciones que envuelven las variables del sistema que proporcionan una representación óptima al sistema. En general, si realizamos un experimento en que aplicamos ciertos valores a las entradas del sistema, obtendremos ciertas respuestas en sus salidas; de ese modo dando los mismos valores que en el experimento a las variables independientes de nuestro modelo, podemos esperar que después de resolver el conjunto de ecuaciones la respuesta del modelo deberá tener respuestas similares a aquellas salidas del sistema físico.

B.1. Identificación de Parámetros del Servomotor

Para identificar los parámetros de un servomotor DC (sin carga) es necesario tener la edición de un programa en C para la generación de un archivo de datos de la velocidad del servomotor en lazo abierto. Los archivos usados para la experiencia en tiempo real son:

- *adqs.cpp*, contiene el programa principal para adquisición de la data de la velocidad. Este programa es desarrollado en lenguaje C.
- *adq_d1.cpp*, contiene el programa de la configuración de la tarjeta LABPC+ que trabaja con la librería *Nidaqbc.lib*.
- *adq_d1.h*, contiene las funciones para el manejo de la tarjeta LABPC+.
- *Nidaqbc.lib* (contiene las librerías de funciones para el manejo de la tarjeta LABPC+).

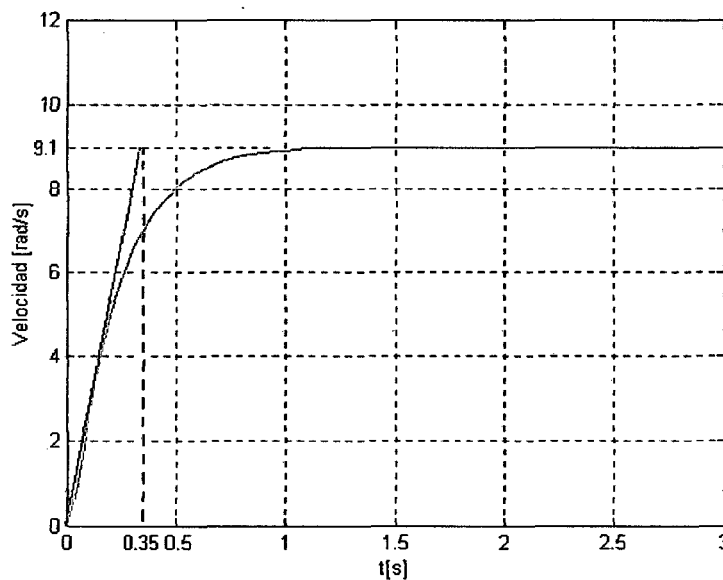
La creación del proyecto se realiza por medio del archivo *adqs.prj*, incluye los archivos: {*adqs.cpp*, *adq1.cpp* y *Nidaqbc.lib*}.

La ejecución del archivo *prj* (ejecución en tiempo real), generan el archivo de datos *adqs24.dat* para el servomotor. El archivo de datos es utilizado para obtener la curva de velocidad que se muestra en la figura B.1. El uso de un método de identificación para determinar los parámetros del servomotor nos conduce a hallar la resistencia de armadura R_a , constante de torque de servomotor K_m , constante de fuerza contraelectromotriz K_b , fricción viscosa B y momento de inercia J .

B.2. Curva de la Velocidad para la Identificación

Usaremos el método gráfico de aproximación de la respuesta del proceso a un sistema de primer orden, cuando es despreciada la inductancia de armadura L_a . Los servomotores que utilizaremos son de fabricación Matsushita modelo GMX 6MPO13A con encoder incremental, disponemos de la información (ver tabla B.1) del servomotor Matsushita con encoder incremental. Para obtener los datos necesarios como son la amplitud K_{med} y a constante de tiempo T_{med} de la curva de velocidad, trazamos una recta tangente en el punto de inflexión, tal como se muestra en la figura B.1, posteriormente trazamos las perpendiculares hacia los ejes coordenados que cortaran a estos en dos puntos, cuyos valores son: $K_{med} = 9,1$ y $T_{med} = 0,35$.

Figura B.1: Curva S.



Disponemos de la información del servomotor Matsushita Modelo GMX-6MPO13A

con encoder incremental; Alimentación 24VDC/0.2A; Velocidad sin carga 310RPM; Reducción 12.5:1. La tabla B.1 muestra los valores que corresponden a las pruebas de laboratorio a 24VDC.

Tabla B.1: Especificaciones del servomotor DC.

Velocidad (W) [rad/seg]	Torque (nT_m) [Nm]	Corriente [A]
32.46	N.L.	0.2
29.84	0.2118	0.5
27.22	0.5507	1.0
24.39	0.8826	1.5
21.46	1.2710	2.0
19.37	1.5534	2.5
15.70	1.8712	3.0
10.47	2.0336	3.5
6.28	2.1466	4.0
STALL	2.3725	4.5

Siguiendo el mismo análisis realizado para los subsistemas de los modelos de los servomotores M_1 y M_2 , y tratándose de servomotores de la mismo modelo de fabricación, se realizó la experiencia en tiempo real para el servomotor M_2 que está acoplado al carro, es decir, los parámetros para el servomotor M_2 serán los mismos para el servomotor M_1 .

Usualmente un modelo lineal de segundo orden para un servomotor DC se expresa mediante la función de transferencia de la velocidad de un servomotor DC dada por.

$$\frac{W(s)}{V_a(s)} = \frac{K_m}{J_{eq}L_a s^2 + (L_a B_{eq} + R_a J_{eq})s + (R_a B_{eq} + K_m K_b)} \quad (B.1)$$

Si omitimos la inductancia, es decir $L_a = 0$, entonces (B.1) se convierte es una función de transferencia de primer orden de la forma.

$$\frac{W(s)}{V_a(s)} = \frac{K_m}{R_a J_{eq} s + (R_a B_{eq} + K_m K_b)} = \frac{K_{med}}{T_{med} s + 1} \quad (B.2)$$

donde.

$$K_{med} = \frac{K_m}{R_a B_{eq} + K_m K_b} \quad (B.3)$$

y

$$T_{med} = \frac{R_a J_{eq}}{R_a B_{eq} + K_m K_b} \quad (B.4)$$

La resistencia R_a se obtiene de la siguiente relación.

$$R_a = \frac{V}{I_{aSTALL}} = \frac{24V}{4,5A} = 5,3\Omega \quad (B.5)$$

La constante contraelectromotriz K_b se obtienen de acuerdo con la tabla B.1 y considerando la inductancia $L_a = 0$.

$$K_b = \frac{V_a - R_a I_a}{nW} = \frac{24V - 5,3\Omega \times 0,2A}{12,5 \times 32,46 \text{rad/s}} = 0,0565 \text{V/rad/s} \quad (B.6)$$

La constante de torque K_m se obtienen de acuerdo con la tabla B.1;

$$nT_{m|STALL} = 2,3725 \text{Nm}$$

El factor de reducción es 12.5, entonces.

$$T_{m|STALL} = 0,1898 \text{Nm}$$

La constante del motor se determina a partir de la relación.

$$K_m = \frac{T_{mSTALL}}{I_{aSTALL}} = \frac{0,1898 \text{Nm}}{4,5A} = 0,0421 \text{Nm/A} \quad (B.7)$$

Reemplazamos los valores $K_{med} = 9,1$ y $T_{med} = 0,35$ que fueron obtenidos a partir de las pruebas de laboratorio para la velocidad en lazo abierto, utilizando la tarjeta de adquisición de datos LABPC+, usando los resultados de R_a y K_m de las ecuaciones (B.5) y (B.7) y dividiendo (B.4) de (B.3) obtenemos la expresión para J_{eq} .

$$J_{eq} = \frac{T_{med} K_m}{K_{med} R_a} = \frac{0,35 \times 0,0421}{9,1 \times 5,3} = 3,0551 \times 10^{-4} \text{Kgm}^2 \quad (B.8)$$

La constante de fricción viscosa obtenemos reemplazando los valores de K_{med} , K_m y R_a en la ecuación (B.3), luego despejamos B_{eq} .

$$B_{eq} = \frac{K_m/K_{med} - K_m^2}{R_a} = \frac{0,0421/9,1 - 0,0421^2}{5,3} = 5,3848 \times 10^{-4} \text{Nm/rad/s} \quad (B.9)$$

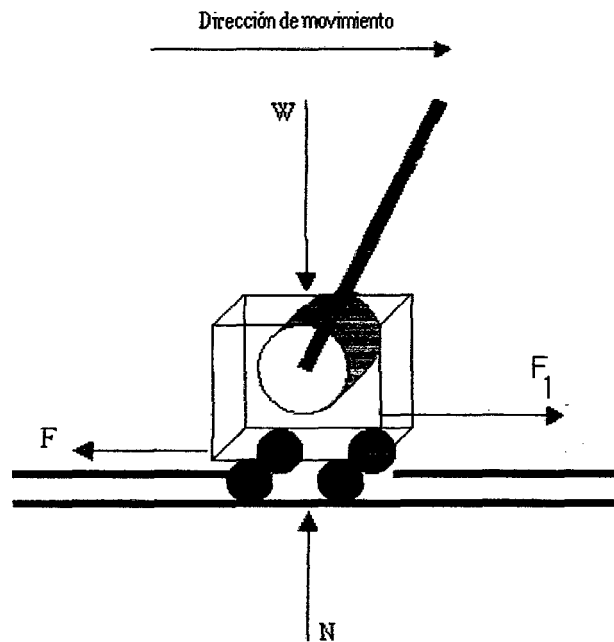
Finalmente se puede determinar la ganancia del amplificador K_A , por medición directa de la tensión en los terminales de armadura del servomotor (salida del amplificador), y considerando la tensión de entrada u de 1.4 voltios.

$$K_A = \frac{V_{aMEDIDO}}{u} = \frac{9,1V}{1,4V} = 5,5 \quad (B.10)$$

B.3. Coeficiente de Fricción del Carro

El concepto de fricción estática puede entenderse mejor considerando el diagrama mostrado en figura B.2. La figura B.2 representa el carro de peso W que descansan en un plano horizontal (el peso W considera los pesos del carro, servomotor y brazo). Una fuerza F_1 que tiende a producir el movimiento hacia la derecha. El movimiento se opone a una fuerza F conocida como fuerza de fricción del carro. F es siempre paralela al plano y opuesta al movimiento. El peso es soportado por la fuerza normal N (el peso de N es siempre perpendicular a la superficie de contacto). Todas las fuerzas son mostradas mediante vectores simples y el objetivo es el equilibrio.

Figura B.2: Dirección de movimiento.



F es una fuerza pasiva, es decir, aumenta F_1 hasta un máximo F_m conocido como la fuerza friccional máxima. F_m es determinado por el peso del objeto y la naturaleza de las superficies en el contacto. Si F_1 es mayor que F_m , el cuerpo se moverá. F_m es igual a la fuerza mínima, mientras se omiten las fuerzas inerciales, necesarias para que el objeto empiece a moverse.

Fricción Deslizante

La fricción deslizante es la fuerza requerida para mantener un movimiento relativo entre dos cuerpos. La figura B.2 puede ser usada para representar el bloque

deslizante con una velocidad constante en el plano horizontal. A velocidades bajas, la fricción es independiente de la velocidad deslizante. A velocidades más altas, la fricción disminuye como el aumento de velocidad. La fricción deslizante normalmente es menor que la fricción estática; por consiguiente, los coeficientes de fricción deslizante son normalmente más pequeños que los coeficientes de fricción estática.

Valor Experimental del Coeficiente de Fricción

El coeficiente de fricción es determinado experimentalmente y puede variar dependiendo de las superficies de contacto y las condiciones particulares bajo la cual se han medido. La tabla B.2 es considerada como una guía general [?].

Sean las ecuaciones de movimiento independientes.

$$\begin{aligned} F_1 &= F_m \\ F_m &= fN \end{aligned} \tag{B.11}$$

la ecuación en el equilibrio.

$$\sum F_y = 0 = N - W \tag{B.12}$$

la ecuación de la fuerza friccional máxima queda expresada en función de la constante de fricción F_c y la velocidad v .

$$vF_c = fN \tag{B.13}$$

luego, la constante de fricción queda expresada.

$$F_c = \frac{fN}{v} \tag{B.14}$$

Los valores de W y v se asignan de acuerdo a las mediciones realizadas experimentalmente, tal es el caso del peso $W = M_c + m = 1,7011Kg$ (ver tabla 2.2) y para calcular la velocidad, tenemos la longitud que recorre el carro de $0,9m$ (longitud máxima de la riel) que recorre en un tiempo de $2,8seg$, de este modo tenemos la velocidad promedio de $0,32m/seg$. Para el coeficiente de fricción, se considera el

caso de material metal sobre metal con una condición seca y para un tipo de fricción deslizando, el valor asignado es $f = 0,054$ (ver tabla B.2). Con estos datos reemplazamos en (B.14) para obtener el coeficiente de fricción del carro F_c .

$$F_c = \frac{0,054(1,7011)(9,8)}{0,32} = 2,81 \text{ Kg/seg}$$

Tabla B.2: Coeficiente de fricción f .

Material	Condición	Estática	Deslizante
Madera sobre madera	Seco	0.30-0.60	0.20-0.50
Madera sobre madera	Mojado	0.40-0.70	0.20-0.40
Madera sobre metal	Seco	0.20-0.60	0.30-0.50
Madera sobre metal	Mojado	0.50-0.70	0.20-0.40
Metal sobre metal	Seco	0.20-1.10	0.03-0.50
Metal sobre metal	Engrasado	0.10-0.50	0.03-0.30
Cuero sobre metal	Seco	0.30-0.50	0.30-0.50
Piedra sobre concreto	Seco	0.60-1.00	
Acero sobre hielo	Seco	0.02-0.03	0.01-0.02

B.4. Coeficiente de Fricción Viscosa del Brazo

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

$$T_{fc} = \begin{cases} -T_c & \dot{q} < 0 \\ T_c & \dot{q} > 0 \end{cases} \quad (\text{B.15})$$

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

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

$$T_m = J_{eq}\dot{w} + B_{eq}w + f^*(nw) \quad (\text{B.16})$$

$$T_m = K_m I_a \quad (\text{B.17})$$

$$I_a = \frac{V_b K_A}{R_a} - \frac{n K_b w}{R_a} \quad (\text{B.18})$$

reemplazamos (A.18) en (A.17), luego igualamos con (A.16).

$$J_{eq} \dot{w} = \frac{V_b K_A K_m}{R_a} - (B_{eq} + \frac{n K_b w}{R_a}) - f^*(nw) \quad (\text{B.19})$$

Las señales en tiempo discreto surgen si el sistema involucra la operación de muestreo de señales continuas. sumimos a w_k como la velocidad angular muestreada en el k -ésimo instante de tiempo.

Usamos la aproximación de Euler $T_s \dot{w} = w_{k+1} - w_k$ para discretizar la ecuación (B.19), luego obtenemos.

$$w_{k+1} = [1 - \frac{T_s}{J_{eq}} (B_{eq} + \frac{n K_m K_b}{R_a})] w_k + \frac{T_s K_m K_A}{J_{eq} R_a} [V_{bk} - \frac{R f^*(n w_k)}{K_m K_A}] \quad (\text{B.20})$$

$f^*(n w_k)$ modela el torque de fricción de Coulomb; podemos indicar que la aproximación es dada por $f^*(n w_k) = V_{zm} \text{sgn}(n w)$ (V_{zm} es la mitad del ancho de la zona muerta que será determinada más adelante). Si la velocidad es distinta de cero, el torque tomará dos valores constantes dependiendo del sentido de giro del servomotor.

La ecuación (B.20) podemos expresar como.

$$w_{k+1} = [1 - \frac{T_s}{J_{eq}} (B_{eq} + \frac{n K_m K_b}{R_a})] w_k + \frac{T_s K_m K_A}{J_{eq} R_a} [V_{bk} - \frac{RC \text{sgn}(n w_k)}{K_m K_A}] \quad (\text{B.21})$$

El diagrama de bloques de la figura B.3 muestra al modelo del servomotor discreto. El diagrama involucra la compensación de la fricción de Coulomb determinada por el parámetro G_3 , que será objeto de análisis para determinar el coeficiente de fricción viscosa C .

Los parámetros para el sistema sin compensar se describen en la tabla B.3, donde G_i , $i = 1, 2, 3, 4$.

Las simulaciones han sido desarrolladas en SIMULINK, cuya entrada V_b es una entrada senoidal de amplitud $0,5V$ y frecuencia de 3rad/seg . El resultado de la simulación para la velocidad angular w es mostrada en la figura B.4

La gráfica de la velocidad angular en estado estacionario del servomotor DC versus el voltaje aplicado al amplificador de potencia, se muestra en la figura B.5,

Figura B.3: Modelo discreto del servomotorDC.

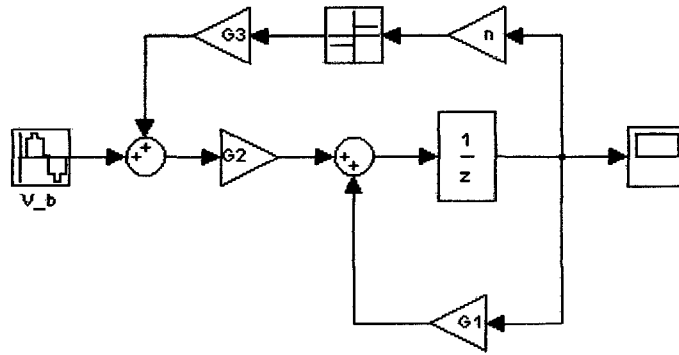
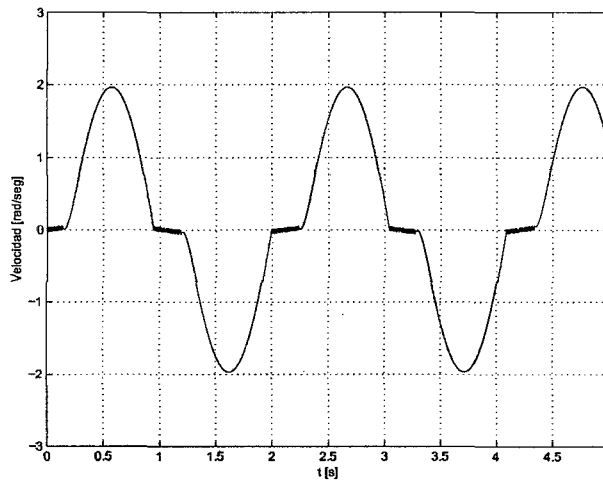


Tabla B.3: Parámetros del sistema a simular.

Parámetro	Descripción	Valor
G_1	$1 - (T_s/J_{eq})(B_{eq} + nK_mK_b/R_a)$	0.9799
G_2	$K_mK_A T_s / (R_a J_{eq})$	0.1430
G_3	$-CR_a / (K_mK_A)$	-0.2174
G_4	n	12.5
T_s		0.001

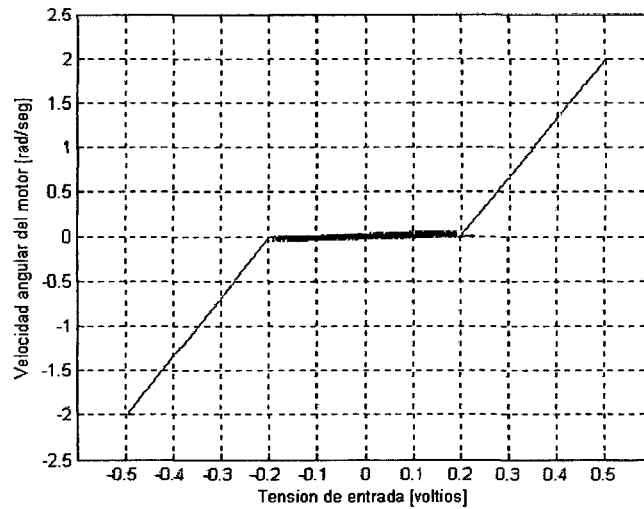
Figura B.4: Simulación del modelo simplificado $L = 0$.



nos permite apreciar una zona muerta a la velocidad cero. Esta zona muerta es debida a la fricción estática del servomotor. Note que el efecto no lineal en las cercanías de la zona muerta.

Para linealizar la ecuación (B.19) se debe de adicionar una tensión de entrada a la entrada del amplificador tal que compense la fricción de Coulomb cuando la

Figura B.5: Tensión de salida versus tensión de entrada.



velocidad del servomotor $w_M \geq w_{th}$ (w_{th} velocidad umbral). Un esquema general es el que presenta la compensación de las fricciones estática y de Coulomb.

$$u = v + V_c \quad (B.22)$$

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

El voltaje de compensación de Coulomb es dado por la relación.

$$V_c = \frac{R_a C}{K_m K_A} \text{sgn}(V_b) \quad \text{si } w_M \geq w_{th} \quad (B.23)$$

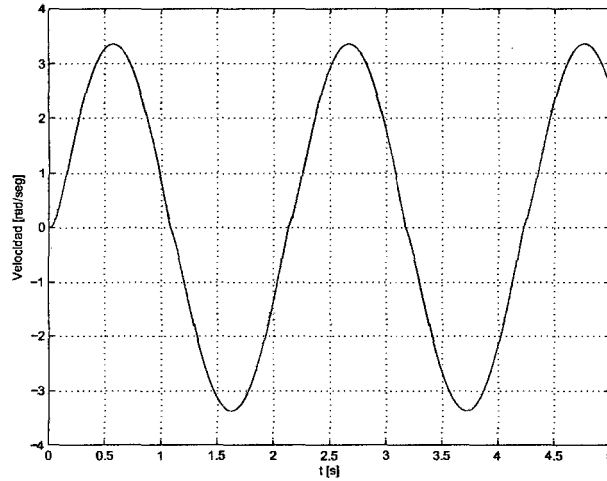
El coeficiente de fricción viscosa C es pequeño para servomotores DC, siendo difícil su identificación en presencia de ruido. Usamos el método de prueba y error para el cálculo de V_c , según la figura B.5, el valor inicial de V_{zm} es 0.2, este valor fue disminuido hasta encontrar el valor apropiado de $V_{zm} = 0,0217$.

Despejamos el coeficiente de fricción C de la ecuación (B.23) para reemplazar los parámetros previamente identificados, como son: $R_a = 5,3\Omega$, $K_A = 5,5$ y $K_m = 0,0421Nm/A$, luego.

$$C = \frac{V_{zm} K_m K_A}{R_a} = \frac{0,0217(0,0421)(5,5)}{5,3} = 9,5e^{-4} \text{ Kg m}^2 / \text{seg} \quad (\text{B.24})$$

El valor $V_{zm} = 0,0217$ que conduce a compensar la fricción de Coulomb resultando el coeficiente de fricción de valor $C = 9,5e^{-4}$. Los efectos de compensación son mostrados en la figura B.6.

Figura B.6: Experimento de la compensación de la fricción no lineal.



Programas en MATLAB

C.1. Listado de Programas

C.1.1. Simulación de la planta en lazo abierto

El programa de simulación están escritos en código MATLAB. Se presentan el programa llamado **openlooptest_mrtm2dof.m** que nos permite obtener la respuesta del sistema

El programa **openlooptest_mrtm2dof.m** es utilizado en las simulaciones realizadas en la sección 3. La característica principal del programa es el uso de la función *f_mrtm2dof.m* para obtener las respuesta de posición del carro y la posición angular de la varilla.

El código del programa de control se lista a continuación.

```
%-----programa openlooptest_mrtm2dof.m-----
clear all; close all; ts=0.01; simtime=0:ts:10;
nsamples=size(simtime);nsamples=nsamples(1,2);

x(1,:)=[0 0 0 pi/4 0 0];
u1=1.0; u2=0.0;

timecontrol=0;u=0;hh=1;v_d=0;v_a=0;inted=0;pp=0;ap=0;ppant=0;apant=0;
for i=1:nsamples
    tt=(i-1)*ts;
%-----
    u1ss(i,1)=u1;
    u2ss(i,1)=u2;

%%SIMULACION DE LA RESPUESTA DEL SISTEMA%%
    xx=x(i,1);
    v=x(i,2);
    a=x(i,3);
    theta=x(i,4);
    w=x(i,5);
    ag=x(i,6);
%%INICIO DE LA RUTINA DE RESPUESTA DEL MODELO%%
    h=ts;
    k1x=h*v;
    k1t=h*w;
    [m1xx,m1tx]=f_mrtm2dof(xx,v,a,theta,w,ag,u1,u2);
    mix=h*m1xx;
```

```

mit=h*mitx;

k2x=h*(v+m1x);
k2t=h*(w+m1t);
[m2x,m2t]=f_mrtm2dof(xx+k1x,v+m1x,a,theta+k1t,w+m1t,ag,u1,u2);
m2x=h*m2x;
m2t=h*m2t;

xx = xx + 0.5*(k1x+k2x);
theta = theta + 0.5*(k1t+k2t);

v = v + 0.5*(m1x+m2x);
w = w + 0.5*(m1t+m2t);

a = mixx; %aceleracion de posicion o desplazamiento lineal
ag = mltx; % aceleracion angular

x(i+1,:)= [xx v a theta w ag];
%%%%%%%%%%
end figure(1)
plotyy(simtime,[x(1:nsamples,1)],simtime,[x(1:nsamples,4)*57.2958]);
grid legend('r [m]','theta [ ]'); figure(2) subplot(121)
plot(simtime,[u1ss(1:nsamples,1)]); grid legend('u1 [V]');
axis([0,simtime(1,nsamples),0,u1*1.25]); subplot(122)
plot(simtime,[u2ss(1:nsamples,1)]); grid legend('u2 [V]');
axis([0,simtime(1,nsamples),0,u2*1.25]);

%%%%%%%%%% ANIMACION%%%%%%%%%%
sim=1;

%Datos para animacion
hei=1; le=0.3; offset=0.2; ddx=0.03; xc(1)=0; yc(1)=hei; xe(1)=0;
ye(1)=hei-le;

xc = x(:,1); yc = hei*ones(nsamples+1,1); xe = x(:,1) +
le*sin(x(:,4)); ye = hei + le*cos(x(:,4));

if sim==1
    figure(4)
    for i=1:10:nsamples
        pause(0.005);
        plot([-offset,2.5],[hei,hei],'-k',[-offset,-offset],[0,hei],
'-k',[2.5,2.5],[hei,0],'-k','linewidth',5);
        grid;
        hold on;
        plot([xc(i),xe(i)],[yc(i),ye(i)],'-b','linewidth',3);
        plot([xc(i)-ddx,xc(i)+ddx],[hei,hei],'-hy','linewidth',6);
        axis([-offset,2.5,-offset,2.5]);
        axis square
        hold off;
    end
end

-> function f_mrtm2dof
[xpp,thetapp]=f_mrtm2dof(x,xp,xpp,theta,thetapp,thetapp,u1,u2);
%=====
% Multivariable Translational Robot Manipulator Dynamic Model

```

```

% -----
% This function returns the values for the accelerations
% of the position and angle, respectively. This function
% is usually a part of a main program to simulate the
% dynamic behavior of the system.
%
% [xpp,thetapp]=f_mrtm2dof(x, xp, xpp, theta, thetap, thetapp, u1, u2)
%
% where:
%      thetapp(k)=ag;
%      thetap(k) =w;
%      theta(k)  =theta;
%      xpp(k)   =a;
%      xp(k)    =v;
%      x(k)     =x;
%=====
%
Mc=1.6492;      %Kg
m=0.0519;      %Kg
l=0.3;         %m
J=2.38e-3;     %Kgm^2
Jeq=3.0551e-4; %Kgm^2
Beq=5.3848e-2; %Nm/rad/s
C=9.5e-4;      %Kgm^2/s
Fc=2.81;       %Kg/s
g=9.81;        %m/s^2
Rp=0.048;      %m

%%%%%%%%Constante motor DC%%%%%%%%
Ra=5.3; Ka=5.5; n=12.5; Km=0.0421; Kb=0.0565;

%%%%%%%% M,P & Q Quoeeficients del modelo del sistema%%%%%%%%

m_11=((Ra*Rp)/(n*Ka*Km))*(Mc+m+(Jeq/Rp^2));
m_12=((Ra*Rp)/(n*Ka*Km))*m*1/2; m_21=(Ra/(n*Ka*Km))*m*1/2;
m_22=(Ra/(n*Ka*Km))*((m*l^2)/4+J+Jeq);
p_11=((Ra*Rp)/(n*Ka*Km))*(Fc+(Beq/Rp^2)+((n^2*Km*Kb)/(Ra*Rp^2)));
p_12=-((Ra*Rp)/(n*Ka*Km))*m*1/2; p_21=0;
p_22=(Ra/(n*Ka*Km))*(C+Beq+((n^2*Km*Kb)/Ra)); d_12=0;
d_21=- (Ra/(n*Ka*Km))*m*1*g/2;

%-----
% Mathematical model using the Euler-Lagrange Formulation
%-----
xpp1=(u1-p_11*xp-m_12*cos(theta)*thetapp-p_12*sin(theta)*thetap^2)/m_11;
thetapp=(u2-p_22*thetap-m_21*cos(theta)*xpp-d_21*sin(theta))/m_22;
xpp=xpp1;

```

C.1.2. Simulación del sistema de control en tiempo continuo en lazo cerrado

Se presenta los programas y subprogramas (funciones) para realizar la simulación del sistema de control BRT2DOF ; el programa principal se denomina: **mrtm_FLC_c.m**.

El código del programa de control se lista a continuación.

```

%Programa de simulacin en tiempo continuo
%Sistema: Manipulador Robtico Traslacional Multivariable
%Controlador: FLC
%
%      Defuzzificacin de orden 0.
%Funciones necesarias:
%   FLC_t.m : Que a su vez necesita de: MF_t.m
%   f_mrtm2dof

clear all; close all;

ts=0.005;           %Tiempo de muestreo virtual para el modelo
simtime=0:ts:8;     %Duracin de la simulacin
nsamples=size(simtime);nsamples=nsamples(1,2); %Nmero de muestras
Kprop_d=14;         %Const. proporcional de salida para la posicin
Kprop_a=0.25;       %Const. proporcional de salida para el ngulo
K_g=0.1398;        %Constante de la compensacin de gravedad

KX=0.60; KXd=0.30; %Ganancias anteriores a la fuzzificacin

x(1,:)=[0 0 0 pi 0 0]; %Estados iniciales
xdes(1,1)=0;          %Posicin inicial deseada
pos_des=input('Ingrese la posicin deseada = '); %Pos. deseada final
ang_des=input('Ingrese el ngulo deseado = '); %Angulo deseado
T_rise=1;             %Tiempo de la rampa
Vrise=(pos_des/T_rise);

%CONTROLADOR DE POSICIN
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Funciones de Pertenencia
%Entradas:
%-----
%distance=Matriz de parmetros de las MSFs de Distance
distance=[ -15.00 -0.400 1; % [ a c sig;
           0.11 -0.250 0; % sigma c gauss;
           0.11 0.000 0;
           0.11 0.250 0;
           15.00 0.400 1];

%Distance=Matriz de parmetros de las MSFs de Distance
distance_d=[ -15.000 -0.350 1;
             0.200 0.000 0;
             15.000 0.350 1];

%Salida:
%-----
%voltage=vector de voltajes
%MSFs      1 2 3 4 5 6 7
voltage_r=[-21 -14 -7 0 7 14 21];

%Base de Reglas

%distance-d
%NL NM NS Z PS PM PL
RB_r=...
[1 2 3; %NL
 2 3 4; %NS
 3 4 5; %Z      distance
 4 5 6; %PS

```

```

5 6 7]; %PL
%CONTROLADOR DEL BRAZO
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Funciones de Pertenencia
%Entradas:
%-----
%angle=Matriz de parmetros de las MSFs de Angle
angle=[-4.0 -2.0 1;
       0.8 -0.5 0;
       0.8 0.0 0;
       0.8 0.5 0;
       4.0 2.0 1];

%angle_d=Matriz de parmetros de las MSFs de Angle_d
angle_d=[-0.75 -7.5 1;
         3.00 0.0 0;
         0.75 7.5 1];

%Salida:
%-----
%voltage=vector de voltajes
%MSFs      1 2 3 4 5 6 7
voltage_q=[-12 -4.5 -2 0 2 4.5 12];

%Base de Reglas

%angle-d
%NL Z PL
RB_q=...
[1 2 3 ; %NL
 2 3 4 ; %NS
 3 4 5 ; %Z      angle
 4 5 6 ; %PS
 5 6 7 ]; %PL

timecontrol=0;u=0;hh=0;v_d=0;v_a=0;inted=0;pp=0;asex=0;ppant=0;asexant=0;
for i=1:nsamples
    tt=(i-1)*ts;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    if (tt<T_rise);
        vdes(i,1)=Vrise;
        xdes(i+1,1)=xdes(i,1)+vdes(i,1)*ts;    %Se genera una rampa
    else
        vdes(i,1)=0;
        xdes(i+1)=pos_des;
    end
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    CONTROLADOR %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
    timecontrol=tt; %Tiempo del controlador coincidente con el de la
    hh=hh+1; %simulacin del modelo, para el caso continuo.
    ppant=pp;
    pp=(x(i,1));
    vp=(pp-ppant)/ts;

    asexant=asex; %Registro de variables para uso posterior
    asex=(x(i,4)*57.2958);
    wsex=(x(i,5)*57.2958);wphh(hh,1)=wsex;

    ed(i,1)=KX*(xdes(i,1)-pp); %Ciculo de las seales de error.

```

```

    ed_d(i,1)=KXd*(vdes(i,1)-vp);
    eq(i,1)=ang_des-ases;

    %Evaluación Fuzzy:
    v_d=FLC_t(ed(i,1),ed_d(i,1),distance,distance_d,RB_r,voltage_r);
    v_a=FLC_t(eq(i,1),-wsex,angle,angle_d,RB_q,voltage_q);

    xhh(hh,1)=x(i,1);
    thetahh(hh,1)=x(i,4);
    thh(hh,1)=tt;
%end
v_dmat(i,1)=v_d; v_amat(i,1)=v_a; %intedmat(i,1)=inted;
%----- deadzone -----
%dzone=4.1111;
%if u<-dzone
%   u=u+dzone;
%elseif u>dzone
%   u=u-dzone;
%else
%   u=0;
%end
%-----
uss(i,1)=Kprop_d*v_d;
uss(i,2)=Kprop_a*v_a-K_g*sin(x(i,4));
usg(i,1)=-K_g*sin(x(i,4));
u1=uss(i,1);%Kprop*v_d;
u2=uss(i,2);%0.1;%0*0.01*v_a+0.0519*0.15*9.81*sin(x(i,4))*0.0002;
%%%%%%%% SIMULACION DE LA RESPUESTA DEL SISTEMA
xx=x(i,1);
v=x(i,2);
a=x(i,3);
theta=x(i,4);
w=x(i,5);
ag=x(i,6);
%%%%%%%% INICIO DE LA RUTINA DE RESPUESTA DEL MODELO
%%%%%%%% RUNGE KUTTA DE 2DO. ORDEN
h=ts;

k1x=h*v;
k1t=h*w;
[m1xx,m1tx]=f_mrtm2dof(xx,v,a,theta,w,ag,u1,u2);
m1x=h*m1xx;
m1t=h*m1tx;

k2x=h*(v+m1x);
k2t=h*(w+m1t);
[m2x,m2t]=f_mrtm2dof(xx+k1x,v+m1x,a,theta+k1t,w+m1t,ag,u1,u2);
m2x=h*m2x;
m2t=h*m2t;

xx = xx + 0.5*(k1x+k2x);
theta = theta + 0.5*(k1t+k2t);

v = v + 0.5*(m1x+m2x);
w = w + 0.5*(m1t+m2t);

a = m1xx;
ag = m1tx;

```

```

    x(i+1,:)= [xx v a theta w ag];
    %%%%%%%%%%
end

figure(1) subplot(121)
plot(simtime,xdes(1:nsamples,1),'r',simtime,x(1:nsamples,1),'b',...
    'LineWidth',1); grid
legend('X deseado','X cabezal'); axis([0,8,0,pos_des*1.1]);axis
square; xlabel('tiempo [s]');ylabel('posicin [m]'); title('a
Posicin. '); subplot(122) plot(simtime,[x(1:nsamples,4)*57.2958],...
    'LineWidth',1); grid
legend('Theta');
%axis([0,8,-10,10]);axis square;
xlabel('tiempo [s]');ylabel('desviacin angular [ sex]'); title('b
Desviacin angular. ');

xxxx=1; if xxxx==1 figure(2) subplot(121)
plot(simtime,uss(:,1),'r','LineWidth',1); grid
%legend('u total','u seguimiento');
xlabel('tiempo [s]');ylabel('seal de control: carro [V]');
axis([0,8,-6,12]);axis square; title('a Seal de control: Posicin
del carro. '); subplot(122)
plot(simtime,uss(:,2),'r',simtime,usg(:,1),'b','LineWidth',1); grid
legend('u total','u comp. g'); axis([0,8,-6,12]);axis square;
xlabel('tiempo [s]');ylabel('seal de control: brazo [V]');
title('b Seal de control: ngulo del brazo. '); end

%figure(3);
%plot(thh,[xhh*10 thetahh*57.2958 u_track u_damp]); grid;
%legend('X x10','theta (deg)','u_tracking','u_damping');
%%%%%%%%% ANIMACION %%%%%%%%%%
if pos_des<0
    sim=2;
else
    sim=1;
end
%sim=3;
%Datos para animacion
hei=1.2; le=0.3; offset=0.2; ddx=0.03; xc(1)=0; yc(1)=hei; xe(1)=0;
ye(1)=hei-le;

xc = x(:,1); yc = hei*ones(nsamples+1,1); xe = x(:,1) +
le*sin(x(:,4)); ye = hei + le*cos(x(:,4));

if sim==1
    figure(4)
    for i=1:10:nsamples
        pause(0.005);
        plot([-offset,2.5],[hei,hei],'-k',[-offset,-offset],[0,hei],'-k',
            [2.5,2.5],[hei,0],'-k','linewidth',5);
        grid;
        hold on;
        plot([xc(i),xe(i)],[yc(i),ye(i)],'-g','linewidth',2);
        plot([xc(i)-ddx,xc(i)+ddx],[hei,hei],'-r','linewidth',6);
        plot(xe(i),ye(i),'ob','linewidth',4);
        axis([-offset,2.5,-offset,2.5]); axis square;
        hold off;
    end
end

```

```

        end
    end

    if sim==2
        figure(4)
        for i=1:10:nsamples
            pause(0.005);
            plot([offset,-2.5],[hei,hei],'-k',[offset,offset],[0,hei],'-k',
                [-2.5,-2.5],[hei,0],'-k','linewidth',5);
            grid;
            hold on;
            plot([xc(i),xe(i)],[yc(i),ye(i)],'-g','linewidth',2);
            plot([xc(i)-ddx,xc(i)+ddx],[hei,hei],'-r','linewidth',6);
            plot(xe(i),ye(i),'ob','linewidth',4);
            axis([-2.5,offset,-offset,2.5]); axis square;
            hold off;
        end
    end
end

```

C.1.3. Simulación del sistema de control en tiempo discreto en lazo cerrado

Se presenta los programas y subprogramas (funciones) para realizar la simulación del sistema de control **BRT2DOF** ; el programa principal se denomina: **mrtm_FLC_d3.m** El código del programa del sistema de control discreto se lista a continuación

```

%Programa de simulacin en tiempo discreto
%Sistema: Manipulador Robtico Traslacional Multivariable
%Controlador: FLC
% Defuzzificacin de orden 0.
%Funciones necesarias:
% FLC_k1.m : Que a su vez necesita de: MF_k.m
% f_mrtm2dof

clear all; close all;

fsc=100; %Frecuencia de muestreo del controlador
tsc=1/fsc; %Periodo de muestreo del controlador
ts=0.005; %Periodo de muestreo para la simulacin
simtime=0:ts:8; %Tiempo de prueba
nsamples=size(simtime);nsamples=nsamples(1,2);
Kprop=14; %Constante proporcional (posicin)

KX=0.60; KXd=0.30; %Ganancias para la posicin
% y velocidad lineal

x(1,:)=[0 0 0 pi 0 0]; %Estados iniciales
xdes(1,1)=0; %Inicializacin de la posicin deseada.
longpend=0.3; Tpend=2.0061*sqrt(longpend); %Parmetros para
%formar la trayectoria.
pos_des=input('Ingrese la posicin deseada = ');
ang_des=input('Ingrese el ngulo deseado = ');
Vmax=(pos_des/Tpend)*8/13; %Parmetro para la trayectoria.
T_rise=1; Vrise=(pos_des/T_rise);

```



```

%CONTROLADOR DE POSICIN
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Funciones de Pertenencia

%Entradas:
%-----
%distance=Matriz de parmetros de las MFs de posicin
distance=[ -15.00 -0.400 1; % [ a    c sig;
           0.11 -0.250 0; %   sigma c gauss;
           0.11  0.000 0;
           0.11  0.250 0;
           15.00 0.400 1];

%Distance=Matriz de parmetros de las MSFs de Distance
distance_d=[ -15.000 -0.350 1;
             0.200  0.000 0;
             15.000  0.350 1];

%Salida:
%-----
%voltage=vector de voltajes
%MSFs      1  2  3 4 5 6 7
voltage_r=[-21 -14 -7 0 7 14 21]; %Valores singleton

%Base de Reglas

%velocidad lineal
% N Z P
RB_r=...
[1 2 3; %NL
 2 3 4; %NS
 3 4 5; %Z      posicion
 4 5 6; %PS
 5 6 7]; %PL

%CONTROLADOR DEL BRAZO

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Funciones de Pertenencia

%Entradas:
%-----
%angle=Matriz de parmetros de las MFs de Angle
angle=[-4.0 -2.0 1;
       0.8 -0.5 0;
       0.8  0.0 0;
       0.8  0.5 0;
       4.0  2.0 1];
sizerangle=5; mbflabels_q=strvcat('NG','NP','Z','PP','PG');

%angle_d=Matriz de parmetros de las MSFs de Angle_d
angle_d=[-0.75 -7.5 1;
         3.00  0.0 0;
         0.75  7.5 1];
sizerangle_d=3;

```

```

%Salida:
%-----
%voltage=vector de voltajes
%MSFs      1  2  3  4  5  6  7
voltage_q=[-12 -4.5 -2 0 2 4.5 12];

%Base de Reglas

%angle-d
% N Z P
RB_q=...
[1 2 3 ; %NL
 2 3 4 ; %NS
 3 4 5 ; %Z      angle
 4 5 6 ; %PS
 5 6 7 ]; %PL

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%TRANSFORMACION DE LAS FUNCIONES DE MEMBRESIA AL
%UNIVERSO DE NUMEROS ENTEROS

%%Constantes de transformacin
npulse_encoder1=512; npulse_encoder2=512; %Parametros para
qep=2; ls7184=4; dpolea=0.048*2; n1=19.745; %calcular las constantes

m2pulse=npulse_encoder1*qep*n1/(pi*dpolea);
gsex2pulse=npulse_encoder2*ls7184/360; v2pulse=m2pulse*tsc;
wsex2pulse=gsex2pulse*tsc;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Tamaos de las MFs
sizep=size(distance); sizep=sizep(1,1); sizev=size(distance_d);
sizev=sizev(1,1); sizea=size(angle); sizea=sizea(1,1);
sizew=size(angle_d); sizew=sizew(1,1);
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Se conserva la ltima columna de cada MF.
distance_pulsos(:,3)=distance(:,3);
distance_d_pulsos(:,3)=distance_d(:,3);
angle_pulsos(:,3)=angle(:,3); angle_d_pulsos(:,3)=angle_d(:,3);
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Se transforma el offset ("c") de cada MF.
distance_pulsos(:,2)=distance(:,2).*m2pulse;
distance_d_pulsos(:,2)=distance_d(:,2).*v2pulse;
angle_pulsos(:,2)=angle(:,2).*gsex2pulse;
angle_d_pulsos(:,2)=angle_d(:,2).*wsex2pulse;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Se transforma la ganancia en cada MF.
for k=1:sizep
    if distance(k,3)==1
        distance_pulsos(k,1)=distance(k,1)*10000/m2pulse;
    else
        distance_pulsos(k,1)=10000/(distance(k,1))/m2pulse;
    end
end

for k=1:sizev
    if distance_d(k,3)==1

```

```

        distance_d_pulsos(k,1)=distance_d(k,1)*10000/v2pulse;
    else
        distance_d_pulsos(k,1)=10000/(distance_d(k,1))/v2pulse;
    end
end

for k=1:sizea
    if angle(k,3)==1
        angle_pulsos(k,1)=angle(k,1)*10000/gsex2pulse;
    else
        angle_pulsos(k,1)=10000/(angle(k,1))/gsex2pulse;
    end
end

for k=1:sizew
    if angle_d(k,3)==1
        angle_d_pulsos(k,1)=angle_d(k,1)*10000/wsex2pulse;
    else
        angle_d_pulsos(k,1)=10000/(angle_d(k,1))/wsex2pulse;
    end
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%%%%%%%%%% INICIO DE LA SIMULACION %%%%%%%%%%%
timecontrol=0;u=0;hh=0;v_d=0;v_a=0;inted=0;pp=0;asex=0;ppant=0;asexant=0;ap=0;apant=0;
for i=1:nsamples
    tt=(i-1)*ts; %tt=valor del tiempo en segundos.

    %%%%%%%%%%% GENERADOR DE TRAYECTORIA
    if (tt<T_rise);
        vdes(i,1)=Vrise;
        xdes(i+1,1)=xdes(i,1)+vdes(i,1)*ts;
    else
        vdes(i,1)=0;
        xdes(i+1)=pos_des;
    end

    %%%%%%%%%%% CONTROLADOR
    controlar=floor((tt-timecontrol)/tsc);
    if controlar==1; %Detecta el momento en que el controlador debe
        timecontrol=tt; %muestrear las entradas y actualizar la seal
        hh=hh+1; % de control.
        controltime(hh)=tt;

        %EN ADELANTE TODAS LAS VARIABLES SE TRANSFORMAN DE MAGNITUDES FSICAS
        %A NMEMRO DE PULSOS
        ppant=pp; %Se guarda el valor de la posicin
        pp=floor(m2pulse*x(i,1)); %prdidada de info. debida a la discretizacin
        vp=(pp-ppant);

        apant=ap;
        ap=floor(gsex2pulse*x(i,4)*57.2958);%prdidada de info. debida a la discretizacin
        wp=0*(ap-apant);
        wphh(hh,1)=wp;

```

```

ed(i,1)=KX*(xdes(i,1)*m2pulse-pp);
ed_d(i,1)=KXd*(vdes(i,1)*v2pulse-vp);
eq(i,1)=ang_des*gsex2pulse-ap;

v_d=FLC_k1(ed(i,1),0*ed_d(i,1),distance_pulsos,distance_d_pulsos,RB_r,voltage_r);
v_a=FLC_k1(eq(i,1),0,angle_pulsos,angle_d_pulsos,RB_q,voltage_q);

xhh(hh,1)=x(i,1);
thetahh(hh,1)=x(i,4);
thh(hh,1)=tt;
end
v_dmat(i,1)=v_d; v_amat(i,1)=v_a; %intedmat(i,1)=inted;
%----- deadzone -----
%dzone=4.1111;
%if u<-dzone
% u=u+dzone;
%elseif u>dzone
% u=u-dzone;
%else
% u=0;
%end
%-----
uss(i,1)=Kprop*v_d;
uss(i,2)=0.5*v_a-0.1398*sin(x(i,4));
usg(i,1)=-0.1398*sin(x(i,4));
u1=uss(i,1);
u2=uss(i,2);

%%%%%%%%% SIMULACION DE LA RESPUESTA DEL SISTEMA
xx=x(i,1);
v=x(i,2);
a=x(i,3);
theta=x(i,4);
w=x(i,5);
ag=x(i,6);

% INICIO DE LA RUTINA DE RESPUESTA DEL MODELO

%RUNGE KUTTA DE 2DO. ORDEN
h=ts;
k1x=h*v;
k1t=h*w;
[m1xx,m1tx]=f_mrtm2dof(xx,v,a,theta,w,ag,u1,u2);
m1x=h*m1xx;
m1t=h*m1tx;

k2x=h*(v+m1x);
k2t=h*(w+m1t);
[m2x,m2t]=f_mrtm2dof(xx+k1x,v+m1x,a,theta+k1t,w+m1t,ag,u1,u2);
m2x=h*m2x;
m2t=h*m2t;

xx = xx + 0.5*(k1x+k2x);
theta = theta + 0.5*(k1t+k2t);

v = v + 0.5*(m1x+m2x);

```

```

w = w + 0.5*(m1t+m2t);

a = mixx;
ag = mitx;

x(i+1,:)= [xx v a theta w ag];
%%%%%%%%%%
end

figure(1) subplot(121)
plot(simtime,xdes(1:nsamples,1),'r',simtime,x(1:nsamples,1),'b',...
'LineWidth',1); grid
legend('X deseado','X cabezal'); axis([0,8,0,pos_des*1.1]);axis
square; xlabel('tiempo [s]');ylabel('posicin [m]'); title('a)
Posicin. '); subplot(122) plot(simtime,[x(1:nsamples,4)*57.2958],...
'LineWidth',1); grid
legend('Theta');
%axis([0,8,-10,10]);axis square;
xlabel('tiempo [s]');ylabel('desviacin angular [ sex]'); title('b)
Desviacin angular. ');

xxxx=1; if xxxx==1 figure(2) subplot(121)
plot(simtime,uss(:,1),'r','LineWidth',1); grid
%legend('u total','u seguimiento');
xlabel('tiempo [s]');ylabel('seal de control: carro [V]');
axis([0,8,-6,12]);axis square; title('a) Seal de control: Posicin
del carro. '); subplot(122)
plot(simtime,uss(:,2),'r',simtime,usg(:,1),'b','LineWidth',1); grid
legend('u total','u comp. g'); axis([0,8,-6,12]);axis square;
xlabel('tiempo [s]');ylabel('seal de control: brazo [V]');
title('b) Seal de control: ngulo del brazo. '); end

%figure(3);
%plot(thh,[xhh*10 thetahh*57.2958 u_track u_damp]); grid;
%legend('X x10','theta (deg)','u_tracking','u_damping');
%%%%%%%%%% ANIMACION
if pos_des<0
sim=2;
else
sim=1;
end
%sim=3;
%Datos para animacion
hei=1.2; le=longpend; offset=0.2; ddx=0.03; xc(1)=0; yc(1)=hei;
xe(1)=0; ye(1)=hei-le;

xc = x(:,1); yc = hei*ones(nsamples+1,1); xe = x(:,1) +
le*sin(x(:,4)); ye = hei + le*cos(x(:,4));

if sim==1
figure(4)
for i=1:10:nsamples
pause(0.005);
plot([-offset,2.5],[hei,hei],'-k',[ -offset,-offset],[0,hei],'-k',
[2.5,2.5],[hei,0],'-k','linewidth',5);
grid;
hold on;
plot([xc(i),xe(i)],[yc(i),ye(i)],'-g','linewidth',2);

```

```

        plot([xc(i)-ddx,xc(i)+ddx],[hei,hei],'-r','linewidth',6);
        plot(xe(i),ye(i),'ob','linewidth',4);
        axis([-offset,2.5,-offset,2.5]); axis square;
        hold off;
    end
end

if sim==2
    figure(4)
    for i=1:10:nsamples
        pause(0.005);
        plot([offset,-2.5],[hei,hei],'-k',[offset,offset],[0,hei],'-k',
            [-2.5,-2.5],[hei,0],'-k','linewidth',5);
        grid;
        hold on;
        plot([xc(i),xe(i)],[yc(i),ye(i)],'-g','linewidth',2);
        plot([xc(i)-ddx,xc(i)+ddx],[hei,hei],'-r','linewidth',6);
        plot(xe(i),ye(i),'ob','linewidth',4);
        axis([-2.5,offset,-offset,2.5]); axis square;
        hold off;
    end
end
end

```

C.1.4. Simulación de la discretización de los sensores de posición lineal y angular

Se presenta el programa y subrutinas (funciones) para realizar la simulación de discretización los sensores sistema de control **BRT2DOF** ; el programa principal se denomina: **mrtmedof_discret_sensors.m**

El código del programa se lista a continuación

```

clear all; close all;
%Programa de simulacin en tiempo discreto
%Sistema: Manipulador Robtico Translacional Multivariable
%de 2 grados de libertad.
%Controlador: FLC
%          Defuzzificacin de orden 0.

npulse_encoder1=512;%numero de pulsos del encoder para la posicion
qep=2;%multiplicador del QEP
npulse_encoder2=512;
ls7184=4;%multiplicador del t2
dpolea=0.048*2;
m2pulse=npulse_encoder1*qep*19.745/(pi*dpolea);
gsex2pulse=npulse_encoder2*ls7184/360;

ts=0.001;
simtime=0:ts:3;
nsamples=size(simtime);nsamples=nsamples(1,2);
x(:,1)=2*sin(1*simtime');
x(:,2)=2*cos(1*simtime');
x(:,4)=90*sin(3*simtime');
x(:,5)=270*cos(3*simtime');

```

```

fsc=150;
tsc=1/fsc;
v2pulse=npulse_encoder1*qep*5.9/(pi*dpolea)*tsc;
wsex2pulse=npulse_encoder2*ls7184/360*tsc;

timecontrol=0;hh=0;ppac=0;apac=0;ppant=0;apant=0;

for i=1:nsamples
    tt=(i-1)*ts;
    controlar=floor((tt-timecontrol)/tsc);
    if controlar==1;
        timecontrol=tt;
        hh=hh+1;
        controltime(hh)=tt;
        ppant=ppac;
        ppac=floor(m2pulse*x(i,1));
        apant=apac;
        apac=floor(gsex2pulse*x(i,4));
    end
    pp(i)=ppac;
    vp(i)=(ppac-ppant);
    ap(i)=apac;
    wp(i)=(apac-apant);
end

figure(1)
subplot(221)
plot(simtime,x(:,1),'k','LineWidth',1);grid;
xlabel('tiempo [s]');ylabel('posicin [m]');
title('a) Posicin-T. continuo','FontSize',11);
subplot(223)
plot(simtime,x(:,2),'k','LineWidth',1);grid;
xlabel('tiempo [s]');ylabel('velocidad [m/s]');
title('b) Velocidad','FontSize',11);
subplot(222)
plot(simtime,x(:,4),'k','LineWidth',1);grid;
xlabel('tiempo [s]');ylabel('ngulo [Sex]');
title('c) Desviacin Angular','FontSize',11);
subplot(224)
plot(simtime,x(:,5),'k','LineWidth',1);grid;
xlabel('tiempo [s]');ylabel('velocidad angular [Sex/s]');
title('d) Velocidad Angular','FontSize',11);

figure(2)
subplot(221)
plot(simtime,pp,'k','LineWidth',1);grid;
xlabel('tiempo [s]');ylabel('posicin [pulsos]');
title('a) Posicin-150Hz','FontSize',11);
subplot(223)
plot(simtime,vp,'k','LineWidth',1);grid;
xlabel('tiempo [s]');ylabel('velocidad [pulsos/Ts]');
title('b) Velocidad','FontSize',11);
subplot(222)
plot(simtime,ap,'k','LineWidth',1);grid;
xlabel('tiempo [s]');ylabel('ngulo [pulsos]');

```

```

title('c) Desviacin Angular', 'FontSize', 11);
subplot(224)
plot(simtime, wp, 'k', 'LineWidth', 1); grid;
xlabel('tiempo [s]'); ylabel('velocidad angular [pulsos/Ts]');
title('d) Velocidad Angular', 'FontSize', 11);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
fsc=100;
tsc=1/fsc;
ts=0.001;
v2pulse=npulse_encoder1*qep*5.9/(pi*dpolea)*tsc;
wsex2pulse=npulse_encoder2*ls7184/360*tsc;

timecontrol=0; hh=0; ppac=0; apac=0; ppant=0; apant=0;

for i=1:nsamples
    tt=(i-1)*ts;
    controlar=floor((tt-timecontrol)/tsc);
    if controlar==1;
        timecontrol=tt;
        hh=hh+1;
        controltime(hh)=tt;
        ppant=ppac;
        ppac=floor(m2pulse*x(i,1));
        apant=apac;
        apac=floor(gsex2pulse*x(i,4));
    end
    pp(i)=ppac;
    vp(i)=(ppac-ppant);
    ap(i)=apac;
    wp(i)=(apac-apant);
end

figure(3)
subplot(221)
plot(simtime, pp, 'k', 'LineWidth', 1); grid;
xlabel('tiempo [s]'); ylabel('posicin [pulsos]');
title('a) Posicin-100Hz', 'FontSize', 11);
subplot(223)
plot(simtime, vp, 'k', 'LineWidth', 1); grid;
xlabel('tiempo [s]'); ylabel('velocidad [pulsos/Ts]');
title('b) Velocidad', 'FontSize', 11);
subplot(222)
plot(simtime, ap, 'k', 'LineWidth', 1); grid;
xlabel('tiempo [s]'); ylabel('ngulo [pulsos]');
title('c) Desviacin Angular', 'FontSize', 11);
subplot(224)
plot(simtime, wp, 'k', 'LineWidth', 1); grid;
xlabel('tiempo [s]'); ylabel('velocidad angular [pulsos/Ts]');
title('d) Velocidad Angular', 'FontSize', 11);

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
fsc=50;
tsc=1/fsc;
ts=0.001;
v2pulse=npulse_encoder1*qep*5.9/(pi*dpolea)*tsc;
wsex2pulse=npulse_encoder2*ls7184/360*tsc;

```



```

timecontrol=0;hh=0;ppac=0;apac=0;ppant=0;apant=0;

for i=1:nsamples
    tt=(i-1)*ts;
    controlar=floor((tt-timecontrol)/tsc);
    if controlar==1;
        timecontrol=tt;
        hh=hh+1;
        controltime(hh)=tt;
        ppant=ppac;
        ppac=floor(m2pulse*x(i,1));
        apant=apac;
        apac=floor(gsex2pulse*x(i,4));
    end
    pp(i)=ppac;
    vp(i)=(ppac-ppant);
    ap(i)=apac;
    wp(i)=(apac-apant);

end

figure(4)
subplot(221)
plot(simtime,pp,'k','LineWidth',1);grid;
xlabel('tiempo [s]');ylabel('posicin [pulsos]');
title('a) Posicin-50Hz','FontSize',11);
subplot(223)
plot(simtime,vp,'k','LineWidth',1);grid;
xlabel('tiempo [s]');ylabel('velocidad [pulsos/Ts]');
title('b) Velocidad','FontSize',11);
subplot(222)
plot(simtime,ap,'k','LineWidth',1);grid;
xlabel('tiempo [s]');ylabel('ngulo [pulsos]');
title('c) Desviacin Angular','FontSize',11);
subplot(224)
plot(simtime,wp,'k','LineWidth',1);grid;
xlabel('tiempo [s]');ylabel('velocidad angular [pulsos/Ts]');
title('d) Velocidad Angular','FontSize',11);

```

C.1.5. Simulación del sistema de control en tiempo discreto en lazo cerrado con señal de prueba senoidal

Se presenta los programas y subprogramas (funciones) para realizar la simulación del sistema de control **BRT2DOF** ; el programa principal se denomina: **mrtm.FLC_d4.m** El código del programa del sistema de control discreto se lista a continuación

```

%Programa de simulacin en tiempo discreto
%Sistema: Manipulador Robtico Traslacional Multivariable
%Controlador: FLC
%          Defuzzificacin de orden 0.
%Funciones necesarias:
% FLC_k1.m : Que a su vez necesita de: MF_k.m
% f_mrtm2dof

```

```

clear all; close all;

fsc=100; %Frecuencia de muestreo del controlador
tsc=1/fsc; %Periodo de muestreo del controlador
ts=0.005; %Periodo de muestreo para la simulacin
simtime=0:ts:8; %Tiempo de prueba
nsamples=size(simtime);nsamples=nsamples(1,2);
Kprop=14; %Constante proporcional (posicin)

KX=0.60; KXd=0.30; %Ganancias para la posicin
% y velocidad lineal

x(1,:)=0 0 0 pi 0 0]; %Estados iniciales
xdes(1,1)=0; %Inicializacin de la posicin deseada.
longpend=0.3; Tpend=2.0061*sqrt(longpend); %Parametros para
%formar la trayectoria.
pos_des=input('Ingrese la amplitud de posicin deseada = ');
ang_des=input('Ingrese la amplitud angular deseada = ');
Vmax=(pos_des/Tpend)*8/13; %Parametro para la trayectoria.

%CONTROLADOR DE POSICIN
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Funciones de Pertenencia

%Entradas:
%-----
%distance=Matriz de parmetros de las MFs de posicin
distance=[ -15.00 -0.400 1; % [ a c sig;
           0.11 -0.250 0; % sigma c gauss;
           0.11 0.000 0;
           0.11 0.250 0;
           15.00 0.400 1];

%Distance=Matriz de parmetros de las MSFs de Distance
distance_d=[ -15.000 -0.350 1;
             0.200 0.000 0;
             15.000 0.350 1];

%Salida:
%-----
%voltage=vector de voltajes
%MSFs 1 2 3 4 5 6 7
voltage_r=[-21 -14 -7 0 7 14 21]; %Valores singleton

%Base de Reglas

%velocidad lineal
% N Z P
RB_r=...
[1 2 3; %NL
 2 3 4; %NS
 3 4 5; %Z posicin
 4 5 6; %PS
 5 6 7]; %PL

```

```

%CONTROLADOR DEL BRAZO

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Funciones de Pertenencia

%Entradas:
%-----
%angle=Matriz de parmetros de las MFs de Angle
angle=[-4.0 -2.0 1;
       0.8 -0.5 0;
       0.8 0.0 0;
       0.8 0.5 0;
       4.0 2.0 1];
sizerangle=5;
mbflabels_q=strvcat('NG','NP','Z','PP','PG');

%angle_d=Matriz de parmetros de las MSFs de Angle_d
angle_d=[-0.75 -7.5 1;
         3.00 0.0 0;
         0.75 7.5 1];
sizerangle_d=3;

%Salida:
%-----
%voltage=vector de voltajes
%MSFs   1 2 3 4 5 6 7
voltage_q=[-12 -4.5 -2 0 2 4.5 12];

%Base de Reglas

%angle-d
% N Z P
RB_q=...
[1 2 3 ; %NL
 2 3 4 ; %NS
 3 4 5 ; %Z angle
 4 5 6 ; %PS
 5 6 7 ]; %PL

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

%TRANSFORMACION DE LAS FUNCIONES DE MEMBRESIA AL
%UNIVERSO DE NUMEROS ENTEROS

%%Constantes de transformacin
npulse_encoder1=512; npulse_encoder2=512; %Parametros para
qep=2; ls7184=4; dpolea=0.048*2; n1=19.745; %calcular las constantes

m2pulse=npulse_encoder1*qep*n1/(pi*dpolea);
gsex2pulse=npulse_encoder2*ls7184/360;
v2pulse=npulse_encoder1*qep*n1/(pi*dpolea)*tsc;
wsex2pulse=npulse_encoder2*ls7184/360*tsc;
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%Tamaos de las MFs
sizep=size(distance); sizeep=sizeep(1,1);

```


%%%%%%%%% GENERADOR DE TRAYECTORIA

```
xdes(i,1)=sin(pi*tt/2)*pos_des/2+1;
vdes(i,1)=cos(pi*tt/2)*pos_des/2*(pi/2);
angdes(i,1)=sin(pi/4*tt)*ang_des;
```

%%%%%%%%% CONTROLADOR

```
controlar=floor((tt-timecontrol)/tsc);
if controlar==1; %Detecta el momento en que el controlador debe
    timecontrol=tt; %muestrear las entradas y actualizar la seal
    hh=hh+1; % de control.
    controltime(hh)=tt;
    ppant=pp; %Se guarda el valor de la posicin
    pp=floor(m2pulse*x(i,1)); %Prdida de informacin debida a la digitalizacin
    vp=(pp-ppant);

    apant=ap;
    ap=floor(gsex2pulse*x(i,4)*57.2958);%Prdida de informacin debida a la digitalizacin
    wp=0*(ap-apant);
    wphh(hh,1)=wp;

    ed(i,1)=KX*(xdes(i,1)*m2pulse-pp);
    ed_d(i,1)=KXd*(vdes(i,1)*v2pulse-vp);
    eq(i,1)=angdes(i,1)*gsex2pulse-ap;

    v_d=FLC_k1(ed(i,1),0*ed_d(i,1),distance_pulsos,distance_d_pulsos,RB_r,voltage_r);
    v_a=FLC_k1(eq(i,1),0,angle_pulsos,angle_d_pulsos,RB_q,voltage_q);

    xhh(hh,1)=x(i,1);
    thetahh(hh,1)=x(i,4);
    thh(hh,1)=tt;
end
v_dmat(i,1)=v_d; v_amat(i,1)=v_a; %intedmat(i,1)=inted;
%----- deadzone -----
%dzone=4.1111;
%if u<-dzone
% u=u+dzone;
%elseif u>dzone
% u=u-dzone;
%else
% u=0;
%end
%-----
uss(i,1)=Kprop*v_d;
uss(i,2)=0.5*v_a-0.1398*sin(x(i,4));
usg(i,1)=-0.1398*sin(x(i,4));
u1=uss(i,1);
u2=uss(i,2);

%%%%%%%%% SIMULACION DE LA RESPUESTA DEL SISTEMA
xx=x(i,1);
```

```

v=x(i,2);
a=x(i,3);
theta=x(i,4);
w=x(i,5);
ag=x(i,6);

% INICIO DE LA RUTINA DE RESPUESTA DEL MODELO

%RUNGE KUTTA DE 2DO. ORDEN
h=ts;

k1x=h*v;
k1t=h*w;
[m1xx,m1tx]=f_mrtm2dof(xx,v,a,theta,w,ag,u1,u2);
m1x=h*m1xx;
m1t=h*m1tx;

k2x=h*(v+m1x);
k2t=h*(w+m1t);
[m2x,m2t]=f_mrtm2dof(xx+k1x,v+m1x,a,theta+k1t,w+m1t,ag,u1,u2);
m2x=h*m2x;
m2t=h*m2t;

xx = xx + 0.5*(k1x+k2x);
theta = theta + 0.5*(k1t+k2t);

v = v + 0.5*(m1x+m2x);
w = w + 0.5*(m1t+m2t);

a = m1xx;
ag = m1tx;

x(i+1,:)= [xx v a theta w ag];
%%
end

figure(1)
subplot(121)
plot(simtime,xdes(1:nsamples,1),'r',simtime,x(1:nsamples,1),'b',...
'LineWidth',1); grid
legend('X deseado','X cabezal');
axis([0,8,0,pos_des+1.2]);
xlabel('tiempo [s]');ylabel('posicin [m]');
title('a) Posicin. ');
subplot(122)
plot(simtime,angdes(1:nsamples,1),'r',simtime,[x(1:nsamples,4)*57.2958],'b',...
'LineWidth',1); grid
legend('Theta');
%axis([0,8,-10,10]);axis square;
xlabel('tiempo [s]');ylabel('desviacin angular [ sex]');
title('b) Desviacin angular. ');

xxxx=1;
if xxxx==1
figure(2)
subplot(121)

```

```

plot(simtime,uss(:,1),'r','LineWidth',1); grid
%legend('u total','u seguimiento');
xlabel('tiempo [s]');ylabel('seal de control: carro [V]');
axis([0,8,-20,20]);axis square;
title('a) Seal de control: Posicin del carro.');
```

```

subplot(122)
plot(simtime,uss(:,2),'r',simtime,usg(:,1),'b','LineWidth',1); grid
legend('u total','u comp. g');
axis([0,8,-20,20]);axis square;
xlabel('tiempo [s]');ylabel('seal de control: brazo [V]');
title('b) Seal de control: ngulo del brazo.');
```

```

end

%figure(3);
%plot(thh,[xhh*10 thetahh*57.2958 u_track u_damp]); grid;
%legend('X x10','theta (deg)','u_tracking','u_damping');
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%% ANIMACION
if pos_des<0
    sim=2;
else
    sim=1;
end
%sim=3;
%Datos para animacion
hei=1.2;
le=longpend;
offset=0.2;
ddx=0.03;
xc(1)=0; yc(1)=hei;
xe(1)=0; ye(1)=hei-le;

xc = x(:,1);
yc = hei*ones(nsamples+1,1);
xe = x(:,1) + le*sin(x(:,4));
ye = hei + le*cos(x(:,4));

if sim==1
figure(4)
    for i=1:10:nsamples
        pause(0.005);
        plot([-offset,2.5],[hei,hei],'-k',[-offset,-offset],[0,hei],'-k',[2.5,2.5],[hei,0],
            '-k','linewidth',5);
        grid;
        hold on;
        plot([xc(i),xe(i)],[yc(i),ye(i)],'-g','linewidth',2);
        plot([xc(i)-ddx,xc(i)+ddx],[hei,hei],'-r','linewidth',6);
        plot(xe(i),ye(i),'ob','linewidth',4);
        axis([-offset,2.5,-offset,2.5]); axis square;
        hold off;
    end
end

if sim==2
figure(4)
    for i=1:10:nsamples
        pause(0.005);
        plot([offset,-2.5],[hei,hei],'-k',[offset,offset],[0,hei],'-k',[-2.5,-2.5],[hei,0],
            '-k','linewidth',5);
    end
end

```

```
grid;
hold on;
plot([xc(i),xe(i)],[yc(i),ye(i)],'-g','linewidth',2);
plot([xc(i)-ddx,xc(i)+ddx],[hei,heil],'-r','linewidth',6);
plot(xe(i),ye(i),'ob','linewidth',4);
axis([-2.5,offset,-offset,2.5]); axis square;
hold off;
end
end
```