

UNIVERSIDAD CATÓLICA DE SANTA MARÍA

FACULTAD DE CIENCIAS E INGENIERÍAS FÍSICAS Y FORMALES

**PROGRAMA PROFESIONAL DE INGENIERIA
MECÁNICA, MECANICA ELÉCTRICA Y MECATRONICA**



**CONTROL DIFUSO PID DE UN SISTEMA DE CONTROL PARA UNA GRÚA
PUENTE**

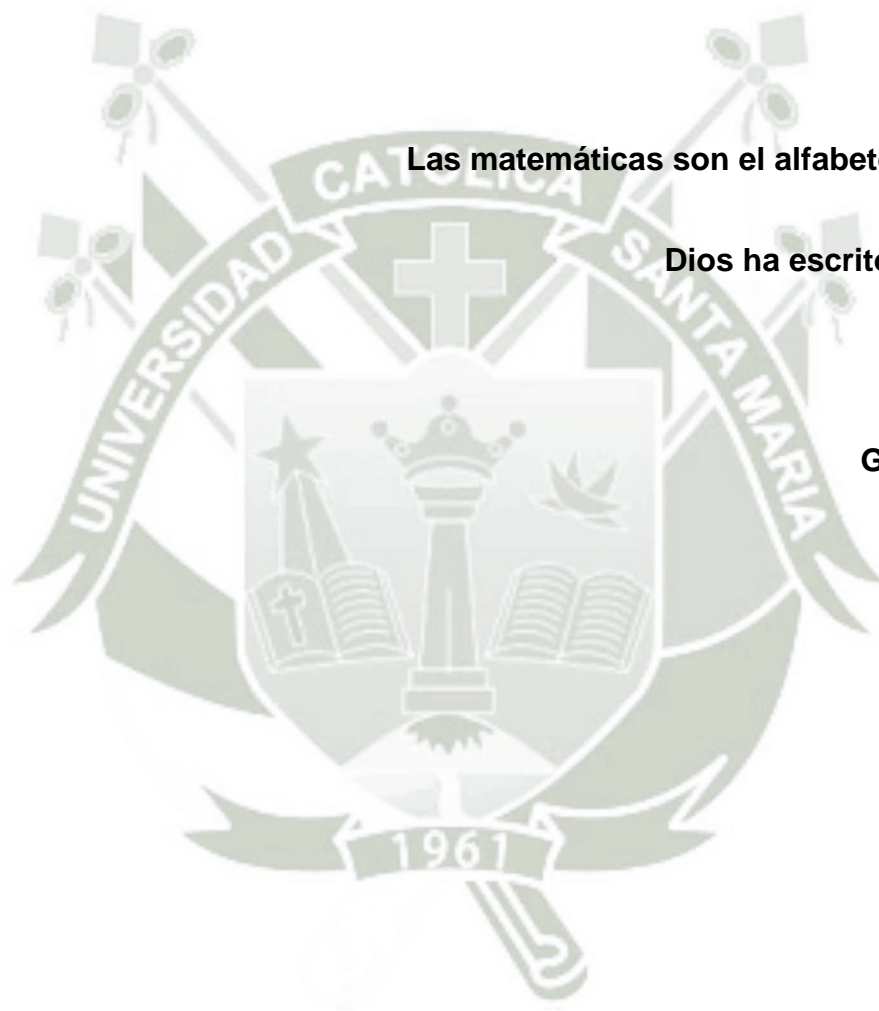
TESIS PRESENTADA POR EL BACHILLER:

JUAN JOSÉ SALAS CERVANTES

**PARA OBTENER EL TÍTULO PROFESIONAL
DE INGENIERO MECATRÓNICO**

AREQUIPA – PERÚ

2013



Las matemáticas son el alfabeto con el cual

Dios ha escrito el universo

Galileo Galilei

DEDICATORIA

A Dios por brindarme la oportunidad y la dicha de la vida, al brindarme los medios necesarios para continuar mi formación como profesional y siendo un apoyo incondicional para lograrlo ya que sin él no hubiera podido.

A mis padres, a mis hermanas y sobrinos dedicarle este presente documento quienes permanentemente me apoyaron con su espíritu alentador, contribuyendo incondicionalmente a lograr mis metas y objetivos propuestos y que al brindarme con su ejemplo a ser perseverante y darme la fuerza que me impulsó a conseguirlo.

Y a mi abuelo Abel Salas, que siempre deseó que sea ingeniero y una pena que la vida no le haya permitido ver eso realizado, pero sé que desde el cielo me ve y se alegra por este triunfo.

AGRADECIMIENTOS

A los docentes, al Ing. Sergio Mestas que me han acompañado durante este largo camino, brindándome siempre su orientación con profesionalismo ético en la adquisición de conocimientos y afianzando mi formación.

Igualmente a mi gran amigo Dennis Barrios quien me ayudo y enseñó con paciencia a culminar este gran proyecto, que a pesar de las dificultades siempre encontrar el camino hacia un lugar mejor.



INDICE GENERAL

ÍNDICE GENERAL	5
ÍNDICE DE IMÁGENES	9
RESUMEN	12
ABSTRACT	13
INTRODUCCIÓN	14
CAPITULO I: MARCO METODOLOGICO	
1.1. Identificación del Problema	15
1.2. Descripción del Problema	15
1.3. Antecedentes	16
1.4. Objetivos	
1.4.1. Objetivo General	20
1.4.2. Objetivos Específicos	20
1.5. Hipótesis	20
CAPÍTULO II: MARCO TEÓRICO	
2.1. Grúas Punte	21
2.1.1. Generalidades	21
2.1.2. Componentes	21
2.1.3. Parámetros	23
2.1.4. Potencia de los Motores Eléctricos	25
2.1.4.1 Motores de Elevación	25
2.1.4.2. Motores de Traslación	26
2.1.5. Puestos de Operación	26
2.2. Tipos de Soluciones Utilizando Grúas Punte	27
2.3. Arquitectura integrada para el control de una Grúa Punte	31

2.4. Productividad y Eficiencia	33
2.5. Riesgos y Factores de Riesgo	34
2.5.1. Riesgos en grúas puente	35
2.5.2. Medidas de prevención. Equipamiento de las grúas	37
2.5.2.1. Dispositivos de seguridad	37
2.5.2.2. Disposiciones e instalaciones básicas de seguridad en las grúas	39
2.5.2.3. Elementos de seguridad básicos para una Grúa Puente	39
2.5.3. Método de evaluación del riesgo	40
2.6. Control de Procesos	40
2.6.1. Conceptos Básicos	41
2.6.2. Características del Proceso	42
2.6.2.1. Lazo abierto y lazo cerrado	44
2.6.3. Tipos de Control	44
2.6.4. Características del Sistema de Control	45
2.6.5. Controladores PID	46
2.6.5.1. Reglas de sintonía de controladores PID	48
2.6.5.1.1. Reglas de Ziegler-Nichols para Sintonizar Controladores PID	49
2.7. Consideraciones Finales	55
CAPÍTULO III: CONTROLADORES DIFUSOS	
3.1. Control con Lógica Difusa	56
3.1.1. Conceptos Básicos	56
3.1.2. Funciones de pertenencia	58
3.1.3. Tipos De Sistemas de Lógica Difusa	62
3.1.3.1. Sistema Tipo Mandani	62
3.1.3.2. Sistema Tipo Sugeno	65

3.1.4. Procedimiento de Diseño de un Control con Lógica Difusa	68
3.2. Controlador Proporcional Integral Derivativo (PID) Difuso	68
3.2.1. Entradas de un controlador PID difuso	68
3.2.2. Salidas de un controlador PID difuso	71
3.2.3. Funciones de Pertenencia	72
3.2.4. Reglas Difusas	72
3.3. Consideraciones finales	75
CAPÍTULO IV: PROPUESTA DE CONTROL	
4.1. Modelado	77
4.1.1. Diagrama de cuerpo libre del carro	78
4.1.2. Diagrama de cuerpo libre de la carga	78
4.2. Propuesta de Control PID Difuso	83
4.2.1. Entradas Difusas	83
4.2.1.1. Entrada Error del Angulo	83
4.2.1.2. Entrada Derivada del Error del Angulo	84
4.2.1.3. Entrada Error del Carro	85
4.2.1.4. Entrada Derivada del Error del Carro	86
4.2.2. Salidas Difusas	87
4.2.2.1. Salida del Angulo	88
4.2.2.1. Salida del Carro	89
4.2.3. Reglas Difusas	90
CAPÍTULO V: SIMULADOR GRÚA PUENTE	
5.1. Método de Euler	95
5.2. Diseño del simulador	97

CAPÍTULO VI: ANÁLISIS DEL SISTEMA USANDO SIMULINK

6.1. Resultados	105
6.1.1. Resultados Control PD	105
6.1.2. Resultados Control PI	107
6.1.3. Resultados Control PID	108
6.1.4. Resultados Control Difuso	110

CAPITULO VII: ANALIZANDO LOS RESULTADOS CON EL SIMULADOR

7.1. Resultados	113
7.1.1. Resultados Control PD	113
7.1.2. Resultados Control PID	114
7.1.4. Resultados Control Difuso	116
CONCLUSIONES	118
ANEXOS	119
REFERENCIAS BIBLIOGRÁFICAS	150



ÍNDICE DE IMÁGENES

CAPÍTULO II: MARCO TEÓRICO

Figura 2.1. Carro	22
Figura 2.2 Grúa Puente	23
Figura 2.3. Parámetros Grúa Puente	25
Figura 2.4. Ejemplos emplazamientos de cabinas	27
Figura 2.5. Grúa Puente a diferentes alturas	28
Figura 2.6. Grúa Puente para cargas pesadas y dos grúas puente para cargas livianas.	29
Figura 2.7. Disposición de grúas puente para carga pesado y dos grúas consola.	29
Figura 2.8. Disposición de Grúa Puente con viga horizontal corrediza.	30
Figura 2.9. Disposición con pluma giratoria interior.	30
Figura 2.10. Disposición de dos grúas consola.	31
Figura 2.11. Arquitectura Integrada.	32
Figura 2.12. Sistema de Control con Bucle de Control.	43
Figura 2.13. Control PID de una planta.	48
Figura 2.14. Sobreimpulso.	49
Figura 2.15. Respuesta escalón unitario de una planta.	51
Figura 2.16. Curva de respuesta en forma de S.	51
Figura 2.17. Sistema de lazo cerrado con un controlador proporcional.	53

CAPÍTULO III: CONTROLADORES DIFUSOS

Figura 3.1. Función de Perteneceía Trapezoidal.	59
Figura 3.2. Función de Pertenencia Triangular.	60
Figura 3.3. Función de Pertenencia Pseudo-Exponencial.	60
Figura 3.4. Función de Pertenencia Gausiana.	61
Figura 3.5. Función de Pertenencia Sigmoidea.	61

Figura 3.6. Sistema Difuso Mamdani, Procesamiento General.	62
Figura 3.7. Reglas Difusas Tipo Mamdani.	64
Figura 3.8. Sistema Difuso Mamdani, Procesamiento Detallado.	65
Figura 3.9. Sistema Difuso Sugeno, Procesamiento General.	65
Figura 3.10. Reglas Difusas Tipo Sugeno.	66
Figura 3.11. Sistema Difuso Sugeno, Procesamiento Detallado.	67
Figura 3.12. Una entrada PID difuso.	69
Figura 3.13. Dos entradas PID difuso.	69
Figura 3.14. Tres entradas PID difuso. (tipo I)	70
Figura 3.15. Tres entradas PID difuso. (tipo II)	70
Figura 3.16. Distribución de las funciones de membrecía (todo tipo de controladores)	72
Figura 3.17. Comportamiento de las reglas difusas.	73
Figura 3.18. Comportamiento de las reglas difusas.	74
Figura 3.19. Comportamiento de las reglas difusas.	75
CAPÍTULO IV: PROPUESTA DE CONTROL	
Figura 4.1. Diagrama de Cuerpo Libre de una Grúa Puente.	77
Figura 4.2. Diagrama de cuerpo libre.	78
Figura 4.3. Diagrama de cuerpo libre de la carga.	79
Figura 4.4. Propuesta Control PID Difuso.	83
Figura 4.5. Funciones de membrecía de la entrada error del ángulo.	84
Figura 4.6. Funciones de membrecía de la entrada derivada del error del ángulo.	85
Figura 4.7. Funciones de membrecía de la entrada error del carro.	86
Figura 4.8. Funciones de membrecía de la entrada derivada del error del carro.	87
Figura 4.9. Función de membrecía de salida del ángulo.	89
Figura 4.10. Función de membrecía de salida del carro.	90

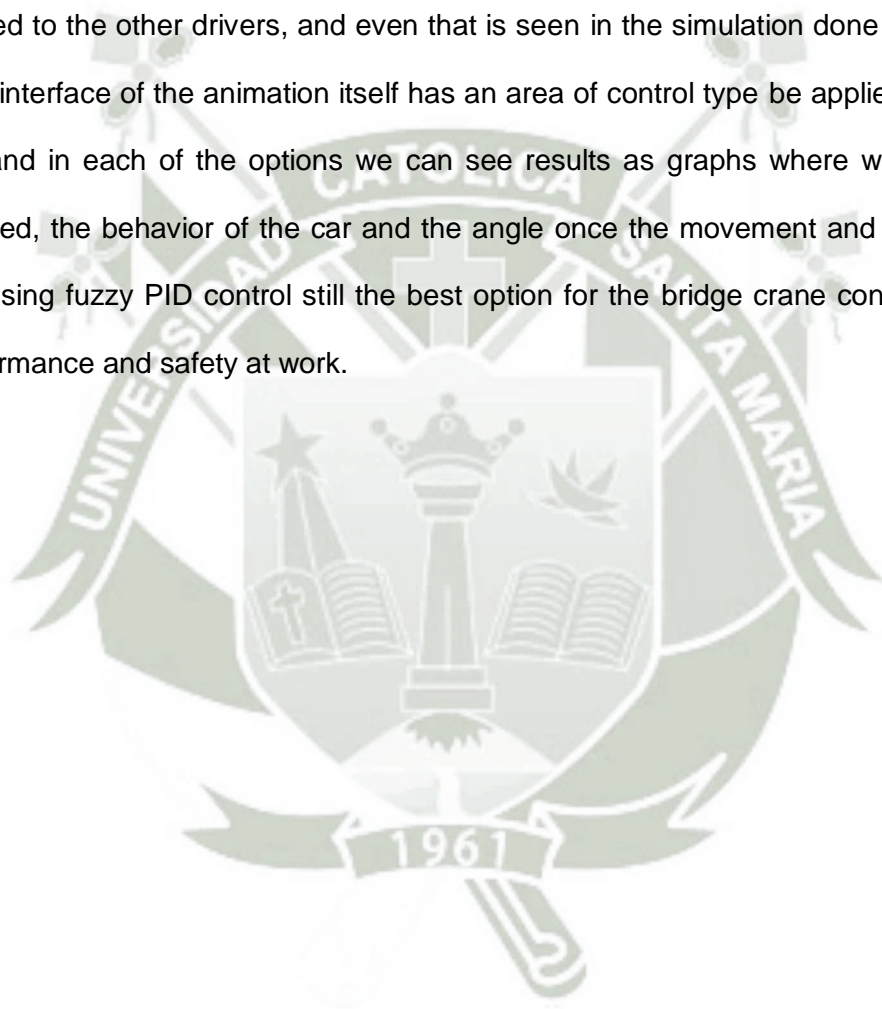
Figura 4.11. Comportamiento de las reglas difusas.	93
Figura 4.12. Reglas difusas. (forma lingüística)	94
CAPÍTULO V: SIMULADOR GRÚA PUENTE	
Figura 5.1. Predicción de un nuevo valor en la solución.	96
Figura 5.2. Interface Simulador.	100
CAPÍTULO VI: ANÁLISIS DEL SISTEMA USANDO SIMULINK	
Figura 6.1. Diagrama de control PD convencional.	103
Figura 6.2. Diagrama de control PI convencional.	104
Figura 6.3. Diagrama de control PID convencional.	104
Figura 6.4. Diagrama de control PID Difuso.	105
Figura 6.5. Comportamiento de la posición, control PD.	106
Figura 6.6. Comportamiento del ángulo, control PD.	106
Figura 6.7. Comportamiento de la posición, control PI.	107
Figura 6.8. Comportamiento del ángulo, control PI.	108
Figura 6.9. Comportamiento de la posición, control PID.	109
Figura 6.10. Comportamiento de la posición, control PID.	109
Figura 6.11. Comportamiento de la posición, control difuso.	110
Figura 6.12. Comportamiento de la posición, control difuso.	111
CAPITULO VII: ANALIZANDO LOS RESULTADOS CON EL SIMULADOR	
Figura 7.1. Comportamiento de la posición, control PD.	113
Figura 7.2. Comportamiento del ángulo, control PD.	114
Figura 7.3. Comportamiento de la posición, control PID.	115
Figura 7.4. Comportamiento de la posición, control PID.	115
Figura 7.5. Comportamiento de la posición, control difuso.	116
Figura 7.6. Comportamiento de la posición, control difuso	116

RESUMEN

En este proyecto se evalúa el desarrollo del controlador PID Difuso para el control de una Grúa Puente, en el cual también se realiza el control convencional como PD, PI y PID para su posterior comparación en la cual se evaluara los valores de sobreimpulso, los tiempo de estabilidad, el error en estado estable y el valor del tiempo de subida. Una vez realizada dicha operación el control PID Difuso responde con mejor cualidades de los en comparación a los otros controladores, e incluso eso se ve en la simulación realizada donde se puede apreciar la interface de la propia animación se tiene un sector del tipo de control que puede aplicar al Grúa Puente y en cada una de las opciones se puede ver resultados como graficas donde se aprecia la fuerza empleada, el comportamiento del carro y del ángulo una vez realizado el movimiento y por lo cual se opta por elegir a el control PID difuso siendo la mejor opción para el control del Grúa Puente; mejorando su desempeño y su seguridad en el trabajo.

ABSTRACT

In this project evaluates the development of fuzzy PID controller for controlling a crane bridge, in which control is also performed conventional PD, PI and PID for later comparison in which overshoot values evaluated, the time stability, steady state error and the value of the rise time. Once this operation Fuzzy PID control responds best qualities compared to the other drivers, and even that is seen in the simulation done where we can see the interface of the animation itself has an area of control type be applied to the crane bridge and in each of the options we can see results as graphs where we can see the force used, the behavior of the car and the angle once the movement and which you opt for choosing fuzzy PID control still the best option for the bridge crane control, improving its performance and safety at work.



INTRODUCCIÓN

Capítulo I: Se plantea el problema y a justificación del mismo, donde se expone el Control Difuso PID de un Sistema de Control para una Grúa Puentes con sus respectivos objetivos y alcances para el desarrollo del mismo.

Capítulo II: Comprende el marco teórico, las descripción de todos los conceptos teóricos sé que tienen que tomar en cuenta para así poder entender y realizar los objetivos propuestos anteriormente.

Capítulo III: Se encuentra todo los conceptos básicos sobre controladores difusos, como los tipos de sistemas de lógica difusa, sus funciones de membresía, entradas, salidas, reglas.

Capítulo IV: Se muestra el diagrama de cuerpo libre del carro y de la carga, además la demostración de cómo se obtienen las funciones de transferencia respectivas del carro y del ángulo. Además de contar con la propuesta de control para darle solución al control de la Grúa Puentes.

Capítulo V: Se tiene como se calculas las aceleración angular y lineal, además te determinar el método de iteración del simulador.

Capítulo VI: Se analizan los resultados obtenido mediante el Simulink, en base a los cuatro criterios de evaluación para la posición y dos para el ángulo y su posterior comparación.

Capítulo VII: Se analizan los resultados obtenido mediante el simulador, en base a los cuatro criterios de evaluación para la posición y dos para el ángulo y su posterior comparación.

Juan José Salas Cervantes.

Arequipa, Mayo del 2013.

CAPITULO I

MARCO METODOLOGICO

1.1. Identificación del Problema

Las grúas puente tienen el problema que al movilizar cargas de un lugar a otro presentan una oscilación, la cual produce pérdida de tiempo ya que el operador tiene que estabilizar la carga por seguridad para proseguir con el movimiento de la misma. Lo anterior se traduce en pérdidas de tiempo y dinero para la empresa; es así que se recomienda implementar un sistema de control automático para este tipo de grúas, lo que nos otorga una forma más segura para el trabajo.

1.2. Descripción del Problema

En los procesos industriales secuenciales, donde es necesario transportar y manipular materias primas, productos semielaborados o productos terminados, entre otros diferentes procesos de transporte y manipulación; se hace cada vez más necesario optimizar el proceso de transporte y manipulación de cargas pesadas, dotando de un mayor grado de automatización y de “inteligencia” a las soluciones basadas en grúas puente.

Esta necesidad aparece debido a que controlar el balanceo presente en la carga es una tarea de dificultad elevada. Esto significa que para poder realizar esta operación es necesario contar con un operario con experiencia en este trabajo, o en su defecto contar con un dispositivo para ayudar a los operadores sin experiencia a realizar sus

trabajo sin problemas; de la misma forma que realiza la empresa Schneider Electric¹ que suministra dispositivos de antibalaneo, cruzamiento, sincronización de elevación, control de sobre carga, etc. Otra opción es implementar controladores automáticos que evitarían la necesidad de contar con un controlador humano.

1.3. Antecedente

En la actualidad se hace cada vez menos necesario disponer del trabajo humano ya que por diferentes factores se producen accidentes o incluso es difícil mantener profesionales con experiencia en determinada área, llegando muchas veces a contratarse personal con poca o nula experiencia. La afirmación anterior no es ajena a los procesos donde se utilicen grúas puente; es por eso que constantemente se desarrollan o adecuan formas de ayudar al trabajador en su labor, lo que, además de mejorar la productividad y eficiencia en ambientes industriales, brinda mayor seguridad. Un ejemplo de esta generación constante de nuevas tecnologías que minimicen los riesgos para el ser humano es la realización de un control inteligente para resolver el problema de control cinemático de una Grúa Puente dedicado a la carga/descarga de mercancías, propuesto por Valera García et al². En esta propuesta, aspectos tales como el control de posición (seguidor de trayectoria), antibalaneo de la mercancía transportada, así como la generación de trayectorias óptimas multiobjetivo, teniendo en cuenta criterios de minimización de tiempos y recorridos, vida útil de los elementos de transmisión mecánica, etc., son analizados y abordados por medio de la aplicación de Técnicas de Control Inteligente basadas en Algoritmos Genéticos y Redes Neuronales.

¹ http://www.schneiderelectric.es/documents/original-equipment-manufacturers/local/440704_C10-soluciones-gruas.pdf

² Aplicaciones de Técnicas Neuro-genéticas en la resolución del problema de control cinemático en grúa puente. Juanjo Valera García, Eloy Irigoyen, Vicente Gómez-Garay y Mikel Larrea.

No limitándonos a esa clase de solución, tenemos la propuesta de González Álvarez³, quien, teniendo en cuenta el tema de carga/descarga y el posicionamiento de material radioactivo en una central nuclear, desarrollo un simulador basado en Marilou Robotic Studios y Matlab que permite probar diversidad de algoritmos antibalanceo. Además, desarrolló una versión simple del algoritmo de control “Input Shaping”.

Otras soluciones al problema de controlar automáticamente las grúas puente, fueron propuestas por Betancur Parra, Tafur Corrales⁴ y por Didier Giraldo, Eduardo Giraldo⁵. Parra y Corrales controlan una Grúa Puente utilizando el controlador LQR para el cálculo de la ganancias fijas; la utilización de este sistema es útil sobre sistemas SISO y MIMO que hacen que un sistema sea estable y controlable ya que esto lo realiza de manera automática hasta encontrar dichas constantes mínimas que cumplen con lo requerido para el sistema. Betancur y Tafur hacen una comparación entre el algoritmo de ajuste de las matrices Q y R, tradicionalmente utilizados en el controlador LQR, con un algoritmo genético para realizar el mismo trabajo. Esta propuesta la realiza basándose en que los algoritmos genéticos se inspiran en la evolución genética para encontrar las mejores soluciones a un problema. Teniendo estas notables comparaciones entre estos dos algoritmos de ajuste de los controladores LQR se llega a la conclusión de que usando algoritmos genéticos se puede solucionar problemas mucho más específicos, con buenos resultados. Así mismo Didier Giraldo y Eduardo Giraldo presentan una variante en los controladores

³ Entorno de simulación basado en Marilou y Matlab para control antibalanceo de Grúas puente. Silvia González Álvarez.

⁴ Sintonización de un control óptimo cuadrático con computación evolutiva para una grúa puente. Carlos Eduardo Betancur P. y Leonardo Taffurht C.

⁵ Aplicación de un controlador lineal cuadráticos usando un observador de orden mínimo sobre una grúa puente. Didier Giraldo y Eduardo Giraldo.

LQR; en la misma incluye el uso de un observador, ya que en la mayoría de los sistemas reales no se puede acceder a todas las variables de estado; si todas las variables son lo mínimas posibles se tiene un observador de orden reducido. Teniendo todo eso se puede calcular el vector de ganancias K , el cual nos proporciona las matrices Q y R . En las cuales restringen el ángulo y la velocidad angular; los valores de la diagonal Q se seleccionan de acuerdo a las variables que se quieran penalizar, lo cual reduce la oscilación con el mismo tiempo de respuesta pero sin restricciones.

Como última y no menos importante solución se tiene la propuesta de Sánchez Bermúdez et al⁶, la cual fue desarrollada para controlar la oscilación en todo tipo de grúas viajeras. Estos autores controlaron la oscilación utilizando controladores difusos ya que estos constituyen una de las técnicas más avanzadas en la actualidad. Para este caso se aplicaron 3 controladores difusos: El primero para actuar sobre las oscilaciones de la carga que tiene como entradas las oscilaciones y las variaciones de esta; el segundo para las oscilaciones del carro, el cuál toma como entrada el desplazamiento del carro respecto al punto de equilibrio y las variaciones de este; el tercero es el encargado de elaborar la señal de control que se entregará al sistema para compensar ambas oscilaciones.

Finalmente, al analizar detenidamente los antecedentes antes nombrados, se observa que, varios de ellos presentan deficiencias. Dentro de estas deficiencias podemos nombrar el hecho de no incluir en el modelo a la cadena que sostiene la carga, o a todas las variables que hacen parte del movimiento dinámico, como la inercia de la carga y las fricciones de rotación y translación. Además, podemos incluir el hecho de

⁶ Control de oscilaciones en grúas viajeras con Reguladores Fuzzy. Jorge Sánchez Bermúdez, Gilberto Machado Burguesa y Concepción Gómez.

que algunas propuestas (E.G. Sánchez Bermúdez et al) no se cuenta con un modelo real y mucho menos con los parámetros de la Grúa Puentes.

Por otro lado, es importante observar que en la actualidad los controladores difusos se han convertido en los más utilizados y confiables tanto en la academia como en la industria. Ejemplo de tal auge de este tipo de controladores son la innumerable cantidad de aplicaciones existentes en la literatura y de forma comercial; en los cuales se observa un mayor énfasis en los controladores PID difusos. Se puede apreciar aplicaciones como el control de iluminación en una planta⁷, el control de un horno rotatorio para producir carbón activado⁸, el control de un horno eléctrico⁹, el control de la posición de un motor de corriente continua (DC)¹⁰, el control de crucero en un vehículo a bajas velocidades¹¹, el control de un sistema de generación de energía eléctrica basado en celdas de combustible tipo PEM¹², el control de una trituradora de cono¹³, el control de turbinas de gas en centrales de ciclo combinado¹⁴, el control de estabilidad de un sistema de péndulo invertido¹⁵, el control cinemático directo de un robot manipulador¹⁶, el control de turbinas a gas¹⁷, entre otros. Por lo anteriormente expuesto y por la falta de aplicaciones del control PID difuso a grúas puente, es

⁷ <http://www.slideshare.net/diegomejia230290/-pid>

⁸ <ftp://ftp-urgell.upc.es/Electronica/H.Martinez/Controlador%20PID.pdf>

⁹ <http://tesis.dpucuto.edu.bo/attachments/article/1744/Diseño%20controlador%20PID%20difuso%20aplicado%20horno%20electrico.pdf>

¹⁰ http://www.scielo1.unal.edu.co/scielo.php?script=sci_arttext&pid=S0123-21262010000100007&lng=en&nrm=iso

¹¹ <http://www.car.upm-csic.es/autopia/publicaciones/r2010010.pdf>

¹² http://www.cenidet.edu.mx/subaca/web-elec/tesis_mc/153MC_mvr.pdf

¹³ <http://www.agregados.biz/fag/-control-pid-inmune-en-la-aplicacion-del-sistema-de-con.php>

¹⁴ <http://www.iie.org.mx/2001f/art2.pdf>

¹⁵ <http://www.itsh.edu.mx/noticias2/articulo%20pedro.pdf>

¹⁶ http://www.scielo.cl/scielo.php?pid=S0718-33052011000300002&script=sci_arttext

¹⁷ <http://www.buenastareas.com/ensayos/Turbinas-Con-Pid-Difuso/6290176.html>

que se considera prioritario el desarrollo de este tipo de controladores para su aplicación en grúas puente.

1.4. Objetivos

1.4.1. Objetivo General

- Diseñar y simular un sistema de Control PID Difuso para una Grúa Puente.

1.4.2. Objetivos Específicos

- Desarrollar un controlador de oscilación de la carga suspendida, minimizando balanceos de la misma, en el proceso de transporte; en todos los regímenes de operación.
- Evaluar la respuesta de un sistema de un Control PID Difuso en un simulador de una Grúa Puente.
- Comparar las respuestas del sistema con un control de lazo cerrado, aplicando los controladores convencionales y el PID difuso.

1.5. Hipótesis

- Es probable que al diseñar el sistema de Control PID difuso para una Grúa Puente, se pueda alcanzar las mejoras en el balanceo de carga, seguridad del proceso, productividad, eficiencia y disminución de riesgos de manipulación para la industria.

CAPÍTULO II

MARCO TEÓRICO

2.1. Grúas Puente

2.1.1. Generalidades

Las Grúa Puente son máquinas utilizadas para la elevación y transporte, en el ámbito de su campo de acción, de materiales. Lo anterior generalmente se aplica a procesos de almacenamiento o curso de fabricación. Este tipo de grúa está compuesto por una viga simple o doble, biapoyada sobre dos carriles, elevados sobre unos postes, dispuestos a tal efecto; también la viga puede estar biapoyada en los componentes de la estructura resistente de la nave o edificación.¹⁸

El movimiento longitudinal se lleva a cabo mediante la translación de la viga principal o puente a través de los carriles elevados. En la práctica totalidad de los casos, la rodadura es por ruedas metálicas sobre carriles también metálicos.

El movimiento transversal se realiza mediante el desplazamiento de un polipasto o carro sobre uno o dos carriles dispuestos sobre la viga principal. Y el movimiento vertical se ejecuta a través del mecanismo de elevación.

2.1.2. Componentes

Una Grúa Puente se compone por las siguientes partes:

¹⁸GRUAS, Larrade Emilio, Miravete Antonio. Pág. 317

- a. **Mecanismo de elevación:** Conjunto de motores y aparejos (sistema de poleas y cables destinados a variar fuerzas y velocidades) que se aplican en el movimiento vertical de la carga. Figura 2.1¹⁹ (1)

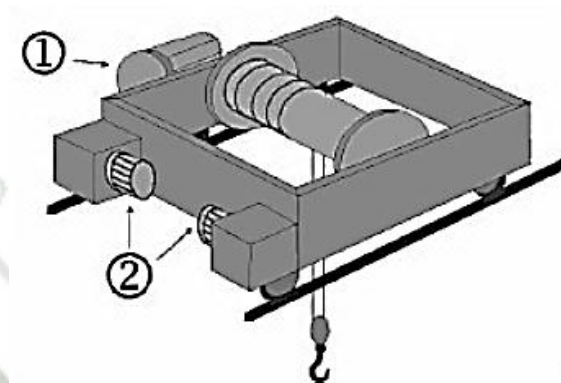


Figura 2.1. Carro

- b. **Mecanismo de translación del carro:** Conjunto de motores que se aplican en el movimiento longitudinal del carro (sistema mecánico con los mecanismos de elevación). Figura 2.1 (2)
- c. **Camino de rodadura:** Elemento estructural por el que se desplaza longitudinalmente la grúa. Figura 2.2²⁰(2) Mecanismo de giro: conjunto mecánico que realiza el desplazamiento angular del brazo o bien de la posición de los ganchos de un carro.
- d. **Botonera:** Dispositivo eléctrico o electrónico unido físicamente mediante una manguera de cables eléctricos a la grúa, para el manejo de la misma desde el exterior de la cabina.
- e. **Telemando:** dispositivo electrónico inalámbrico (sin unión física a la grúa), para el manejo de la grúa.

¹⁹ Norma Técnica Peruana N° 736: Generalidades. pág. 3

²⁰ Norma Técnica Peruana N° 736: Generalidades. pág. 2

- f. **Cabina:** habitáculo destinado, si existe, a la conducción de la grúa y que alberga los dispositivos fijos de mando y al operador o gruista. Figura 2.2 (3)
- g. **Accesorios o útiles de prensión:** elementos auxiliares cuya función es la de sujetar la carga, tales como: pinzas, pulpos, electroimanes, ventosas, cucharas, etc.

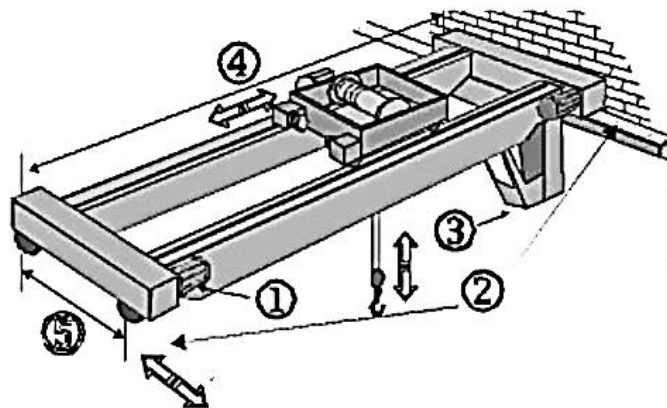


Figura 2.2 Grúa Puente

2.1.3. Parámetros

Para poder definir las características de los diferentes componentes de una Grúa Puente, es necesario tener en consideración ciertos parámetros relacionados a las condiciones en que la misma operara; estos parámetros se lista a continuación:

- a. **Altura máxima de recorrido del gancho:** distancia vertical entre el nivel más bajo del suelo (incluido el foso, si existe) y el gancho de carga, cuando éste se encuentra en la posición más elevada de trabajo. Figura 2.3²¹ (1)

²¹ Norma Técnica Peruana N° 736: Generalidades Pág. 4

- b. **Luz:** es la distancia horizontal entre los ejes de los carriles de la vía de rodadura. Figura 2.3 (2)
- c. **Distancia entre ejes de las ruedas de los testeros:** es la distancia medida paralelamente al eje longitudinal de desplazamiento. Figura 2.2 (5).
- d. **Voladizo total:** distancia máxima horizontal entre el eje del camino de rodadura más próximo al voladizo y el extremo de la estructura emplazada sobre el voladizo. Figura 2.3 (3)
- e. **Voladizo útil:** distancia máxima horizontal entre el eje del camino de rodadura más próximo al voladizo y el eje del elemento de prensión emplazado sobre el voladizo. Figura 2.3 (4)
- f. **Brazo útil:** distancia horizontal entre el eje vertical de la parte giratoria o eje de rodadura y el eje vertical del elemento de prensión.
- g. **Brazo total:** distancia horizontal entre el eje vertical de la parte giratoria o eje de rodadura y el eje vertical del extremo de la estructura.
- h. **Carga nominal o máxima:** valor de la carga fijado por el fabricante e indicado en la placa de características (incluye los accesorios de elevación y aprehensión originales).
- i. **Carga útil:** carga bajo el aparejo o accesorios si los hay.
- j. **Placa de características:** fija en cada grúa indica: el fabricante, año de fabricación, número, carga nominal y útil en función de los alcances, si le es aplicable. Si la grúa dispone de varios mecanismos de elevación, se indicarán las características de cada uno.

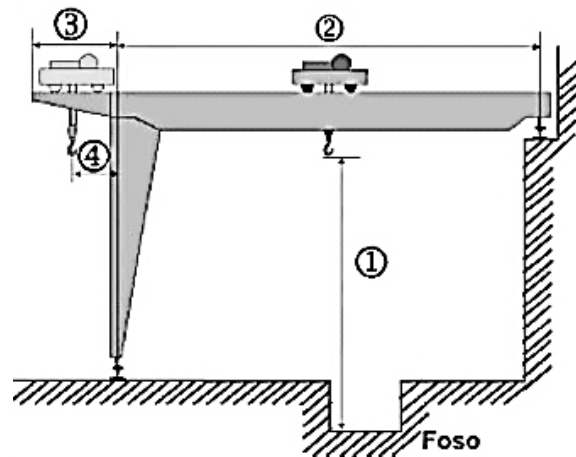


Figura 2.3. Parámetros Grúa Puentes

2.1.4. Potencia de los Motores Eléctricos

Es importante tener en cuenta que la estructura del mecanismo de elevación de cargas debe ser robusta, confiable y capaz de soportar las fuertes cargas verticales y de transmitir los esfuerzos generados por la carga útil a los motores de dicha grúa.²²

2.1.4.1 Motores de Elevación²³

La potencia de los Motores de Elevación está representada por la siguiente ecuación.

$$P_{elev} = \frac{G2 \cdot V_{elev}}{\eta}$$

²² GRUAS, Larrade Emilio, Miravete Antonio. Pág. 325

²³ GRUAS, Larrade Emilio, Miravete Antonio. Pág. 343

Dónde:

G2: Es la carga a elevar. (carga útil + Peso elem. susp.)

V_{elev} : Velocidad de elevación. (m/min)

η : Rendimiento: 0.85

2.1.4.2. Motores de Traslación²⁴

En caso de la traslación es preciso calcular la potencia a régimen permanente y de aceleración.

La potencia de régimen permanente tiene la siguiente expresión:

$$P_{tras} = \frac{G1 + G2 \cdot W \cdot V_{tras}}{4.5 \cdot 10^6 \cdot \eta}$$

Dónde:

G1: Carga muerta a trasladar.

W: 7 para rodamiento, 20 para casquillo de bronce.

V_{tras} : Velocidad de traslación (m/min)

η : Rendimiento: 0.85

2.1.5. Puestos de Operación

Es el lugar del cual el operador maneja los órganos de control de la grúa, las grandes grúas generalmente se operan desde una cabina situada en la estructura con diferentes posiciones (Figura 2.4²⁵) y también se pueden operar desde el suelo mediante una botonera o telemando.

²⁴ GRUAS, Larrade Emilio, Miravete Antonio. Pág. 343

²⁵ Norma Técnica Peruana N° 737: Utilización y formación de operadores, pág. 2

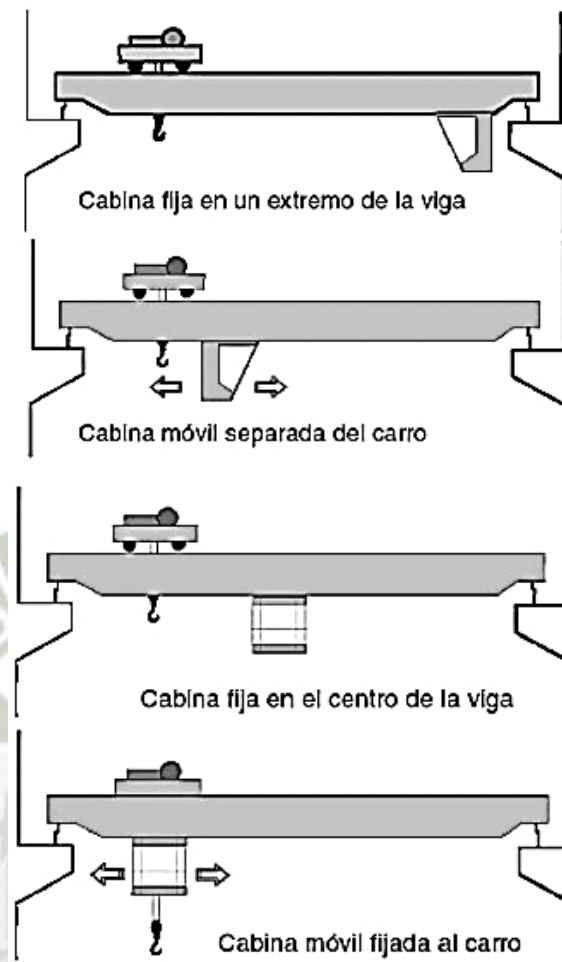


Figura 2.4. Ejemplos emplazamientos de cabinas

2.2. Tipos de Soluciones Utilizando Grúas Puente.

En la práctica, cuando se implementan soluciones basadas en grúas puente, es importante que se diseñe e instale todo un sistema de grúas que cumplan con las exigencias del trabajo al cual serán sometidas, incluyendo la previsión de las diferentes formas en las que se podría utilizar el sistema de grúas instalado. Por ejemplo en el caso de tener una gran diferencia entre la carga normal y la carga máxima se tendrá el problema de que la velocidad de traslación de la grúa será excesivamente reducida por

las cargas; en este sentido se esbozan 6 soluciones a este problema las cuales se muestran a continuación:

- a. Disponer dos Grúa Puente en la misma vía, cada una de ellas con una capacidad de carga igual a la mitad de la carga máxima a levantar; existiendo un problema de solape entre ambas grúas que trabajan de forma independiente.
- b. Con el objeto de poder realizar diferentes trabajos a condiciones aceptables se realizan 2 o más instalaciones para una Grúa Puente con una carga máxima y el resto para cargas normales con mayor velocidad de traslación del carro, ambas instalaciones se ejecutan a diferentes alturas (Figura 2.5.²⁶). y si existen varias zonas de trabajo se puede adaptar grúas ligeras de luz limitada. (Figura 2.6.²⁷)

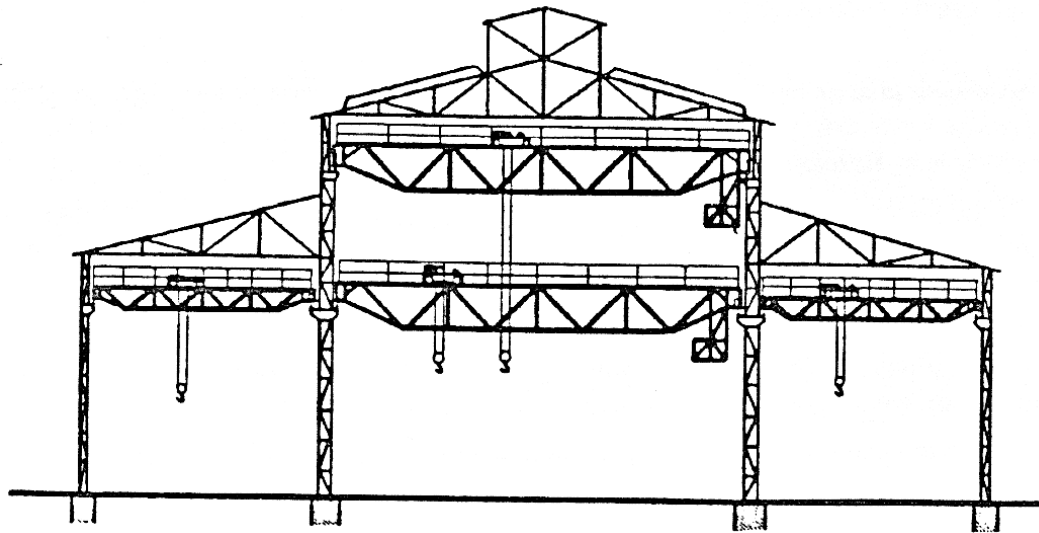


Figura 2.5. Grúa Puente a diferentes alturas.

²⁶ GRUAS, Larrade Emilio, Miravete Antonio. Pág. 318

²⁷ GRUAS, Larrade Emilio, Miravete Antonio. Pág. 319

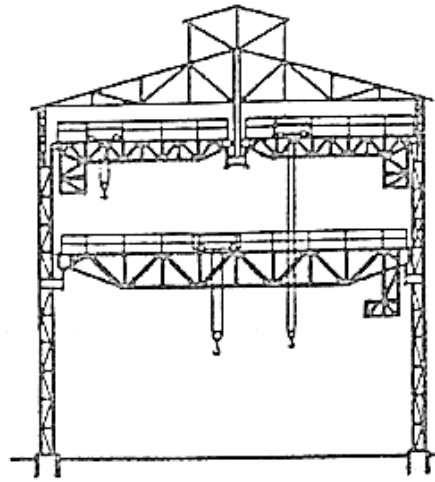


Figura 2.6. Grúa Puente para cargas pesadas y dos grúas puente para cargas livianas.

- c. Se pueden cambiar o reemplazar las grúas para cargas livianas y de luz limitada por unas grúas consola. El momento generado por la carga en voladizo en las citadas grúas, es absorbido por dos esfuerzos horizontales soportados por la pared de la nave²⁸ (Figura 2. 7.²⁹)

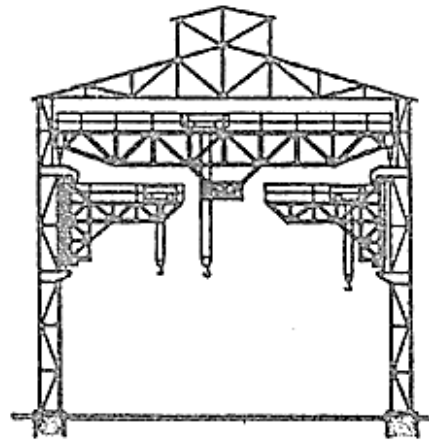


Figura 2.7. Disposición de grúas puente para carga pesado y dos grúas consola.

²⁸ GRUAS, Larrade Emilio, Miravete Antonio. Pág. 319

²⁹ GRUAS, Larrade Emilio, Miravete Antonio. Pág. 319

- d. Cuando se tiene varias zonas de trabajo es necesario trasladar la carga a una zona donde se pueda trasladar con una viga horizontal corrediza que con esto posibilite y haga más fácil la traslación de la carga. (Figura 2.8³⁰)

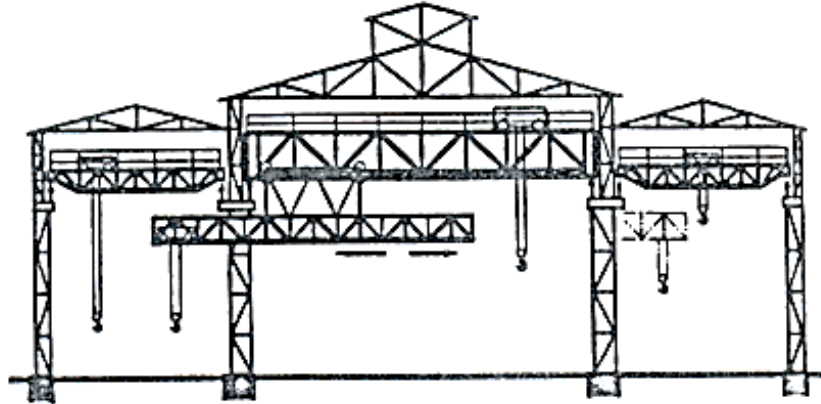


Figura 2.8. Disposición de Grúa Puente con viga horizontal corrediza.

- e. Otra solución es en la que se tiene la posibilidad de disponer de un carro trasladable con pluma giratoria inferior (Figura 2.9³¹)

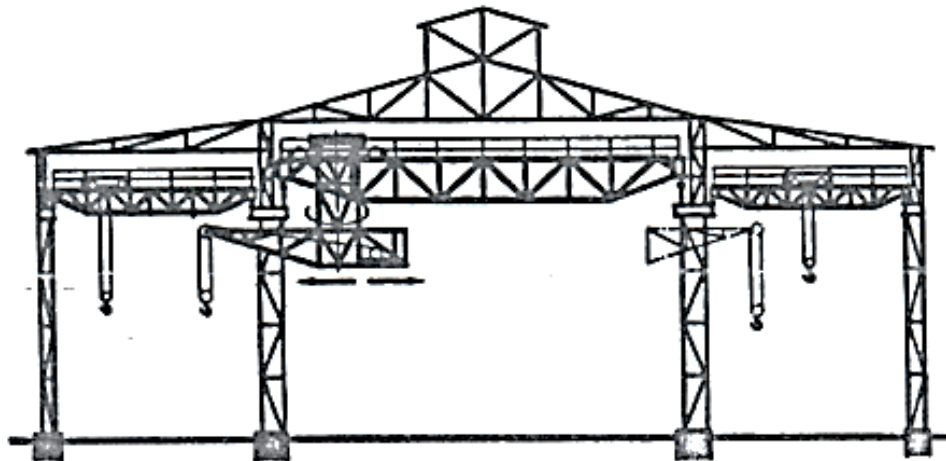


Figura 2.9. Disposición con pluma giratoria interior.

³⁰ GRUAS, Larrade Emilio, Miravete Antonio. Pág. 320

³¹ GRUAS, Larrade Emilio, Miravete Antonio. Pág. 320

- f. Por último, otra solución es la implementación de dos grúas consola situadas en los postes de separación entre ambas zonas de trabajo. (Figura 2.10³²)

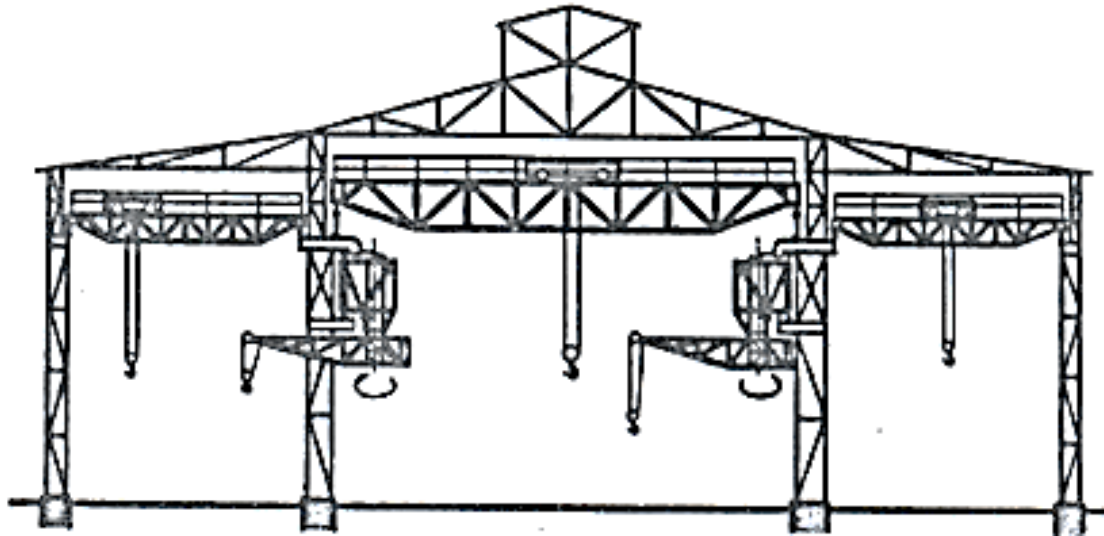


Figura 2.10. Disposición de dos grúas consola

2.3. Arquitectura integrada para el control de una Grúa Puente³³

Las grúas puente cuentan con un arquitectura de control, la cual siempre supervisa las acciones que realiza la misma; por ejemplo este sistema identificara la inminencia de una colisión con otra Grúa Puente o la llegada al final del riel, para lo cual se emitirá una alarma de alerta indicando un posible accidente.

Una arquitectura típicamente utilizada en este tipo de control en grúas puente se puede apreciar en la Figura 2.11.

³² GRUAS, Larrade Emilio, Miravete Antonio. Pág. 321

³³ Soluciones garantizadas para grúas industriales, Schneider Electric, pág. 29

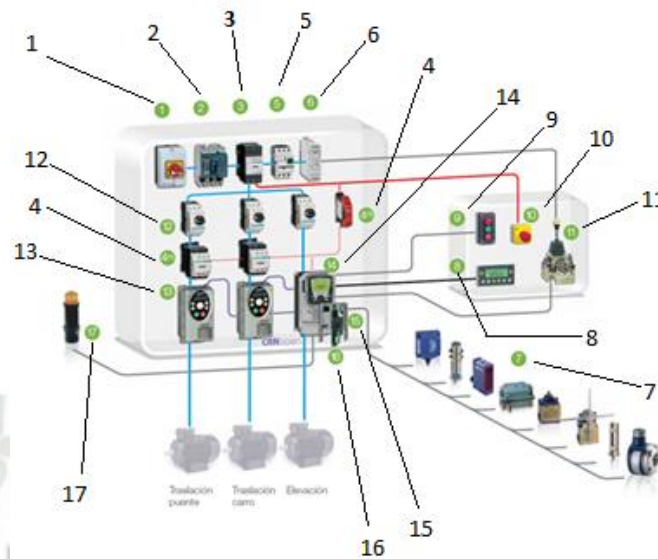


Figura 2.11. Arquitectura Integrada

- Interruptor seccionador | Vario (1)
- Interruptor automático | NS (2)
- Contactor general | TeSys D (3)
- Módulo de seguridad(1) + contactor(1)| Preventa XPS + TeSys D (4)
- Interruptor automático | TeSys GV2R (5)
- Fuente de alimentación para modo de conmutación | Phaseo (6)
- Sensor de proximidad, sensor fotoeléctrico, selector, limitador, codificador |
Osiprox + Osiris + Osiswitch + Osicoder célula de carga (7)
- Pantalla HMI | Magelis XBT (8)
- Inicio parada / envolvente de alarma-zumbador | Harmony XALD (9)
- Parada de emergencia | Harmony XAL (10)
- Combinador | XK (11)
- Interruptor automático | TeSys GV2L (12)

- Variador de velocidad | Altivar 31 (13)
- Variador de velocidad | Altivar 71 (14)
- Controlador | Tarjeta Control Inside VW (15)
- Tarjeta de interface de codificador | VW (16)
- Baliza indicadora | Harmony XVB (17)

2.4. Productividad y Eficiencia

La productividad se define como la cantidad de producción de una unidad de producto o servicio por insumo de cada factor utilizado por unidad de tiempo.

La eficiencia se define como la capacidad de disponer de alguien o de algo para conseguir un objetivo determinado. No debe confundirse con eficacia que se define como la capacidad de lograr el efecto que se desea o se espera.

- **Productividad**

Relación entre la cantidad de bienes y servicios producidos y los recursos necesarios para hacerlo. Es más importante mejorar la productividad que la producción.

$$Productividad = \frac{Cantidad\ de\ productos}{Cantidad\ recursos\ utilizados}$$

La relación mostrada en la ecuación anterior, es una ecuación genérica. La misma hay que reformularla al contexto de las grúas puente. En las grúas puente en lugar de medir la productividad en función de la cantidad de los productos producidos se deberá medir en función de los objetos trasladados. Además, los recursos utilizados en grúas puente se entenderían como el tiempo

en el cual la grúa está en funcionamiento. En este sentido, la fórmula para medir la productividad de grúas puente queda como sigue.

$$Productividad = \frac{Cantidad\ de\ objetos\ trasladados}{Tiempo\ de\ trabajo}$$

- **Eficiencia**

Eficiencia significa alcanzar el mejor grado de cumplimiento de objetivos, al menor costo posible y con los recursos indispensables.

$$Eficiencia = \frac{Resultados}{Objetivos}$$

Pero para poder medir la eficiencia en una Grúa Puente se debe de redefinir la anterior formula dado que los resultados en grúas puente serán medidos en función en la cantidad de los objetos realmente trasladados y es claro que el objetivo de este tipo de dispositivos es trasladar objetos. Por lo tanto, a la hora de medir la eficiencia en grúas puente se deberá considerar la cantidad de objetos que se desean trasladar con una determinada grúa.

$$Eficiencia = \frac{Objetos\ realmente\ trasladados}{Objetos\ deseados\ a\ trasladar}$$

2.5. Riesgos y Factores de Riesgo³⁴

Riesgo se denomina a la probabilidad de que un objeto material, sustancia o fenómeno pueda, potencialmente, desencadenar perturbaciones en la salud o integridad física del trabajador, así como en materiales y equipos.

³⁴ Norma Técnica Peruana N° 736 pág. 5

Factor de riesgo se entiende a la existencia de elementos, fenómenos, ambiente y acciones humanas que encierran una capacidad potencial de producir lesiones o daños materiales, y cuya probabilidad de ocurrencia depende de la eliminación y/o control del elemento agresivo.

2.5.1. Riesgos en grúas puente

El riesgo laboral es la posibilidad de que un trabajador sufra daños por la exposición a los peligros asociados al trabajo que realiza. Las técnicas preventivas son las actuaciones o medidas que se toman en todas las actividades de la empresa para eliminar o reducir los riesgos y, en su defecto, minimizar sus consecuencias, si estos se materializan.

Los factores de riesgo específicos de las grúas, lo constituyen el desplazamiento del equipo (sea en carga o en vacío) y su posible interacción con el personal o con otras máquinas u objetos que se encuentren dentro de la zona de desplazamiento de la grúa.

Los riesgos y factores de riesgo más importantes son los mecánicos, eléctricos, ergonómicos, por fallo de energía y por falta o inadecuación de medidas de seguridad.

Riesgos mecánicos

- De arrastre o atrapamiento por la carga o por la propia grúa.
- De impacto por la carga o por la propia grúa.

- De pérdida de estabilidad (de la carga, de la máquina o de sus elementos).
- De rotura de elementos de la máquina (por envejecimiento, fatiga, etc.)

Riesgos eléctricos

- Pueden ser debidos a contactos eléctricos directos o indirectos.
- Riesgo térmico producido por las resistencias de puesta en marcha que pueden producir quemaduras por contacto.

Riesgos producidos por defectos ergonómicos en el diseño

- Posturas forzadas o esfuerzos excesivos (especialmente en las grúas con cabina para el operador).
- Inadecuada iluminación localizada.

Riesgos producidos por fallo en la alimentación de energía, y otros trastornos funcionales

- Fallo en la alimentación de energía (de los circuitos de potencia y/o de mando).
- Fallo del sistema de mando (puesta en marcha o aceleración intempestivos).

Riesgos producidos por la ausencia y/o inadecuación de medidas de seguridad

- Inexistencia o diseño inadecuado de resguardos o dispositivos de protección.
- Diseño inadecuado de dispositivos de marcha y paro.
- Ausencia o inadecuación de señales y pictogramas seguridad.

- Ausencia o inadecuación de los dispositivos de paro de emergencia.
- Medios inadecuados de carga/descarga.
- Ausencia y/o inadecuación de accesorios en las operaciones de ajuste y/o mantenimiento.
- Ausencia o inadecuación de equipos de protección individual.

A estos riesgos deben añadirse aquellos que son propios del entorno de trabajo de las grúas, como por ejemplo:

- Caída de personas a nivel y desde altura
- Inhalación de sustancias nocivas
- Estrés térmico por calor o frío
- Trauma sonoro
- Etc.

2.5.2. Medidas de prevención. Equipamiento de las grúas³⁵

Las lesiones por accidentes constituyen a nivel mundial una de las cinco primeras causas de muerte e incapacidad.

2.5.2.1. Dispositivos de seguridad

No todos los dispositivos relacionados a continuación deben equipar una grúa. Cada grúa debe equiparse con los dispositivos de seguridad que se requieran, en función del resultado de su evaluación específica de riesgos realizada según su uso específico.

- **Dispositivo anti-choque:** dispositivo que impide la colisión entre grúas con zonas de acción coincidentes. Enclavamiento de la grúa: dispositivo de

³⁵ Norma Técnica Peruana N° 736 pág. 6

anclaje que evita su desplazamiento una vez situada en una posición de paro o reposo.

- **Limitador de aflojamiento del cable:** dispositivo que evita los posibles riesgos generados por el aflojamiento del cable.
- **Limitador de altura de elevación superior e inferior:** dispositivo para impedir que el accesorio de prensión de la carga se eleve/descienda de forma que, pueda golpear la estructura de la grúa o sobrepasar el límite superior o inferior establecido.
- **Limitador de altura de elevación de seguridad:** dispositivo redundante, como elemento de seguridad, que impide que la carga golpee la estructura de la grúa, si fallan el dispositivo anterior
- **Limitador de la carga:** dispositivo automático que impide el manejo de cargas que excedan de la capacidad nominal establecida.
- **Limitador de traslación o de giro:** dispositivos que impiden todo movimiento, a lo largo de los caminos de rodadura o de giro respecto a su eje, que superen los límites establecidos.
- **Pestillo de seguridad:** dispositivo mecánico que impide el desenganche involuntario de los elementos de sujeción de la carga.
- **Señal acústica:** elemento de señalización sonoro que indica el funcionamiento de la grúa.
- **Señales ópticas luminosas:** elementos de señalización luminosos que indican el funcionamiento de la grúa.

- **Paro de emergencia:** dispositivo de accionamiento manual, que al ser pulsado, asegura la inmediata desconexión de todos los elementos motores de la grúa e impide su puesta en marcha intempestiva al ser desenclavado.

2.5.2.2. Disposiciones e instalaciones básicas de seguridad en las grúas.

Con independencia de que la grúa posea o no marcado CE, el usuario deberá realizar la evaluación de riesgos en función de las operaciones y de la ubicación real de la grúa, indicando los elementos de seguridad adicionales precisos para asegurar su funcionamiento seguro.

2.5.2.3. Elementos de seguridad básicos para una Grúa Puente

- Botonera de control, con clara señalización diferenciada de los mandos.
- Dispositivo de paro de emergencia, claramente identificado, que corta el circuito eléctrico de todos los elementos de la grúa excepto los dispositivos de sujeción de la carga, está dotado de un dispositivo que impide su rearme involuntario.
- Las botoneras de control móviles, serán de mando sensitivo, deteniéndose automáticamente la maniobra si se dejan de pulsar.
- Dispositivo de bloqueo de seguridad, con llave, para evitar la utilización de los controles por personal no autorizado.
- Dispositivos de final de carrera superior e inferior en el mecanismo de elevación.
- Finales de carrera de traslación del carro.
- Finales de carrera de traslación del puente y pórtico.

- Limitadores de carga y de par.
- Dispositivo de seguridad que evite la caída de la carga durante su manipulación.
- Ganchos de elevación provistos de pestillo de seguridad.
- Indicación, claramente visible, de la carga nominal.
- Barandillas adecuadas de protección en todos los pasos elevados.
- Carteles de señalización de los riesgos residuales.

2.5.3. Método de evaluación del riesgo³⁶

Para la evaluación de los riesgos propios de las grúas reseñadas en la Norma Técnica Peruana N° 736³⁷, "Grúas tipo puente: Generalidades", se aconseja seguir las indicaciones que se establecen en la UNE EN 1050:1997; mientras que para la evaluación de las condiciones de trabajo asociadas a las mismas se recomiendan las indicaciones contenidas en el manual "Evaluación de las condiciones de trabajo en pequeñas y medianas empresas"³⁸ y en la NTP 330³⁷ "Sistema simplificado de evaluación de riesgos de accidente", ambas publicadas por el INHST³⁹

2.6. Control de Procesos

El control de procesos es una especialidad de la ingeniería que combina, a su vez, distintas ramas, entre las que destacan, sistemas de control, automatización, electrónica e informática.

³⁶ Norma Técnica Peruana N° 736 pág. 6

³⁷ http://www.indecopi.gob.pe/0/modulos/JER/JER_Interna.aspx?ARE=0&PFL=14&JER=71

³⁸ <http://www.insht.es/portal/site/Insht/menuitem.1f1a3bc79ab34c578c2e8884060961ca/?vgnnextoid=ad6c41941b086110VgnVCM100000dc0ca8c0RCRD&vgnnextchannel=cfc0c465c5f13110VgnVCM100000dc0ca8c0RCRD>

³⁹ <http://www.insht.es/portal/site/Insht/>

Su principal aplicación y propósito es el análisis, diseño y automatización de procesos de manufactura de la mayor parte de las áreas industriales como: petróleo y gas, generación de energía eléctrica, textil, alimentaria, automovilística, etc.

2.6.1. Conceptos Básicos⁴⁰

La ingeniería de control se basa en los fundamentos de la teoría de la realimentación y el análisis de sistemas lineales, e integrar los conceptos de las teorías de redes y de comunicación. Por lo tanto, la ingeniería de control no está limitada de la ingeniería, sino que es igualmente aplicable a las ingenierías aeronáuticas, química, mecánica, del medio ambiente, civil y eléctrica. Por ejemplo un sistema de control incluye a menudo componentes eléctricos, mecánicos y químicos. Además, al aumentar el conocimiento de la dinámica de los sistemas comerciales, sociales y políticos, también incrementa la capacidad de control de esos sistemas.

El objeto de todo proceso industrial será la obtención de un producto final, de unas características determinadas de forma que cumpla con las especificaciones y niveles de calidad exigidos por el mercado, cada día más restrictivos. Esta constancia en las propiedades del producto sólo será posible gracias a un control exhaustivo de las condiciones de operación, ya que tanto la alimentación al proceso como las condiciones del entorno son variables en el tiempo. La misión del sistema de control de proceso será corregir las desviaciones surgidas en las variables de proceso respecto de unos valores determinados, que se consideran óptimos para conseguir las propiedades requeridas en el producto producido.

⁴⁰ Sistema de control moderno, Dorf, Bishop, pág. 2

El sistema de control nos permitirá una operación del proceso más fiable y sencilla, al encargarse de obtener unas condiciones de operación estables, y corregir toda desviación que se pudiera producir en ellas respecto a los valores de ajuste.

Las principales características que se deben buscar en un sistema de control serán:

1. Mantener el sistema estable, independiente de perturbaciones y desajustes.
2. Conseguir las condiciones de operación objetivo de forma rápida y continua.
3. Trabajar correctamente bajo un amplio abanico de condiciones operativas.
4. Manejar las restricciones de equipo y proceso de forma precisa.

La implantación de un adecuado sistema de control de proceso, que se adapte a las necesidades de nuestro sistema, significará una sensible mejora de la operación.

Principalmente los beneficios obtenidos serán:

- Incremento de la productividad.
- Mejora de los rendimientos.
- Mejora de la calidad.
- Ahorro energético.
- Control medioambiental.
- Seguridad operativa.
- Optimización de la operación del proceso/ utilización del equipo.
- Fácil acceso a los datos del proceso.

2.6.2. Características del Proceso

El control del proceso consistirá en la recepción de unas entradas, variables del proceso, su procesamiento y comparación con unos valores predeterminados por el

usuario, y posterior corrección en caso de que se haya producido alguna desviación respecto al valor preestablecido de algún parámetro de proceso.

- Elementos de medida (Sensores) Generan una señal indicativa de las condiciones de proceso. bucle de control típico estará formado por los siguientes elementos, a los que habrá Elementos de control lógico (Controladores): Leen la señal de medida, comparan la variable medida con la deseada (punto de consigna) para determinar el error, y estabilizan el sistema realizando el ajuste necesario para reducir o eliminar el error.
- Elementos de actuación (Válvulas y otros elementos finales de control): Reciben la señal del controlador y actúan sobre el elemento final de control, de acuerdo a la señal recibida.

Esta serie de operaciones de medida, comparación, calculo y corrección, constituyen una cadena cerrada constituyen ciclo cerrado. El conjunto de elementos que hacen posible este control reciben el nombre de bucle de control. (Figura 2.12)

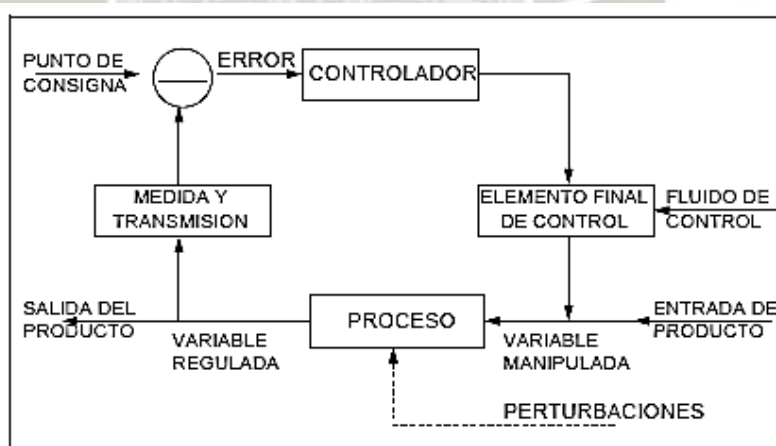


Figura 2.12. Sistema de Control con Bucle de Control

El controlar un proceso, se refiere a como se controlan variables inherentes al mismo para⁴¹:

- Reducir la variabilidad del producto final
- Incrementar la eficiencia
- Reducir impacto ambiental
- Mantener el proceso dentro de los límites de seguridad que corresponda

2.6.2.1. Lazo abierto y lazo cerrado⁴²

Lazo cerrado existe cuando la variable de proceso es medida, comparada con el setpoint y se genera una acción tendiente a corregir cualquier desviación respecto del mismo.

A lazo abierto, la variable de proceso no es comparada y se genera una acción independientemente de las condiciones de la misma.

2.6.3. Tipos de Control⁴³

Se puede hacer una clasificación de los sistemas de control atendiendo al procedimiento lógico usado por el controlador del sistema para regular la evolución del proceso. Los principales tipos de control utilizados en los procesos industriales serán:

Normales

- Sistemas de realimentación. (Feed-back): Proporcional, Integral, Derivativo
- Sistema anticipativo (Feed-Foward)

⁴¹ Control de Procesos, José Roberto Vignoni, pág. 4

⁴² Control de Procesos, José Roberto Vignoni, pág. 9

⁴³ Sistemas de Control Automático. Guía 5. Pág. 3

- Sistema en cascada (Cascade)
- Sistema selectivo (Over-Ride)

Avanzados

- Control de restricciones (Constraint Control)
- Control del modelo de referencia (Model Reference Control)
- Optimización de unidades.

2.6.4. Características del Sistema de Control

Es necesario definir ciertos términos básicos.

- **Señal de salida:** Es la variable que se desea controlar posición, velocidad, presión, temperatura, etc. También se denomina variable controlada.
- **Señal de referencia:** Es el valor que se desea que alcance la señal de salida.
- **Error:** Es la diferencia entre la señal de referencia y la señal de salida real.
- **Señal de control:** es la señal que produce el controlador para modificar la variable controlada de tal forma que se disminuya, o elimine, el error.
- **Señal análoga:** Es una señal continua en el tiempo.
- **Señal digital:** Es una señal que solo toma valores de 1 y 0. La computadora solo envía y/o recibe señales digitales.
- **Convertor análogo/digital:** Es un dispositivo que convierte una señal analógica en una señal digital (1 y 0).
- **Convertor digital/análogo:** Es un dispositivo que convierte una señal digital en una señal analógica (corriente o voltaje).

- **Planta:** Es el elemento físico que se desea controlar. Planta puede ser: un motor, un horno, un sistema de disparo, un sistema de navegación, un tanque de combustible, etc.
- **Proceso:** Operación que conduce a un resultado determinado.
- **Sistema:** Consiste en un conjunto de elementos que actúan coordinadamente para realizar un objetivo determinado.
- **Perturbación:** Es una señal que tiende a afectar la salida del sistema, desviándola del valor deseado.
- **Sensor:** Es un dispositivo que convierte el valor de una magnitud física como presión, flujo, temperatura, etc. En una señal eléctrica codificada ya sea en forma analógica o digital. También es llamado transductor. Los sensores, o transductores, analógicos envían, por lo regular, señales normalizadas de 0 a 5 voltios, 0 a 10 voltios o 4 a 20 mA.
- **Sistema de control en lazo cerrado:** Es aquel en el cual continuamente se está monitoreando la señal de salida para compararla con la señal de referencia y calcular la señal de error, la cual a su vez es aplicada al controlador para generar la señal de control y tratar de llevar la señal de salida al valor deseado. También es llamado control realimentado.
- **Sistema de control en lazo abierto:** En estos sistemas de control la señal de salida no es monitoreada para generar una señal de control.

2.6.5. Controladores PID

Un PID (Proporcional Integral Derivativo) es un mecanismo de control por realimentación que calcula la desviación o error entre un valor medido y el valor que

se quiere obtener, para aplicar una acción correctora que ajuste el proceso. El algoritmo de cálculo del control PID se da en tres parámetros distintos: el proporcional, el integral, y el derivativo. El valor Proporcional determina la reacción del error actual. El Integral genera una corrección proporcional a la integral del error, esto nos asegura que aplicando un esfuerzo de control suficiente, el error de seguimiento se reduce a cero. El Derivativo determina la reacción del tiempo en el que el error se produce. La suma de estas tres acciones es usada para ajustar al proceso vía un elemento de control como la posición de una válvula de control o la energía suministrada a un calentador, por ejemplo. Ajustando estas tres variables en el algoritmo de control del PID, el controlador puede proveer un control diseñado para lo que requiera el proceso a realizar. La respuesta del controlador puede ser descrita en términos de respuesta del control ante un error, el grado el cual el controlador llega al "set point", y el grado de oscilación del sistema. Nótese que el uso del PID para control no garantiza control óptimo del sistema o la estabilidad del mismo⁴⁴. Algunas aplicaciones pueden solo requerir de uno o dos modos de los que provee este sistema de control. Un controlador PID puede ser llamado también PI, PD, P o I en la ausencia de las acciones de control respectivas. Los controladores PI son particularmente comunes, ya que la acción derivativa es muy sensible al ruido, y la ausencia del proceso integral puede evitar que se alcance al valor deseado debido a la acción de control.

Actualmente en el ambiente laboral se usan formas modificadas del control PID, como por ejemplo el control I-PD y el control PID con dos grados de libertad. La utilidad de los controladores se aplica de gran manera a la mayoría de los sistemas

⁴⁴ Sistema de Control Automático, Benjamin C. Kuo, 7ma Edic. pág. 709

de control. En particular, cuando el modelo matemático de la planta no se conoce y por lo tanto, no se puede emplear métodos de diseño analítico, es cuando los controladores PID resultan más útiles⁴⁵.

En el campo de control de procesos es bien conocido que los esquemas de un control PID sencillo y los complejos demuestran una gran utilidad para el control satisfactorio, pero no en todos los casos presente un control óptimo del sistema.

2.6.5.1. Reglas de sintonía de controladores PID

Control PID de plantas. En la Figura 2.13⁴⁶ se muestra un control PID.

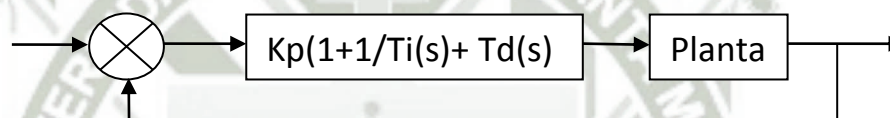


Figura 2.13. Control PID de una planta

El proceso de seleccionar los parámetros del controlador que cumplan con las especificaciones de comportamiento dadas se conocen como sintonía del controlador. Ziegler y Nichols sugirieron reglas para sintonizar los controladores PID lo que significa dar valores a $(K_p, T_i$ y $T_d)$ basándose en las respuesta escalón experimentales o en el valor de K_p que produce estabilidad marginal cuando solo se usa la acción de control proporcional.

Las reglas de Ziegler-Nichols, que se presentan a continuación son muy convenientes cuando no se conoce el modelo matemático de las plantas, pero

⁴⁵ Ingeniería de Control Moderno, Ogata, Katsuniko. 4ta Edic. Pág. 681

⁴⁶ Ingeniería de Control Moderno 4ta Edic" Ogata, Katsuniko. Pág. 682

también se puede aplicar al diseño de sistemas de modelos matemáticos conocidos⁴⁷.

2.6.5.1.1. Reglas de Ziegler-Nichols para Sintonizar Controladores PID

Ziegler y Nichols propusieron unas reglas para determinar los valores de la ganancia proporcional K_p , del tiempo integral T_i y del tiempo derivativo T_d , con base en las características de respuesta transitoria de una planta específica. Tal determinación de los parámetros de los controladores PID o de la sintonización de los controles PID la realizan los ingenieros en el sitio mediante experimentos sobre la planta.

Existen dos métodos denominados reglas de sintonización de Ziegler-Nichols. En ambos se pretende obtener un 25% de sobrepaso máximo en la respuesta escalón como se muestra en la siguiente Figura 2.14.

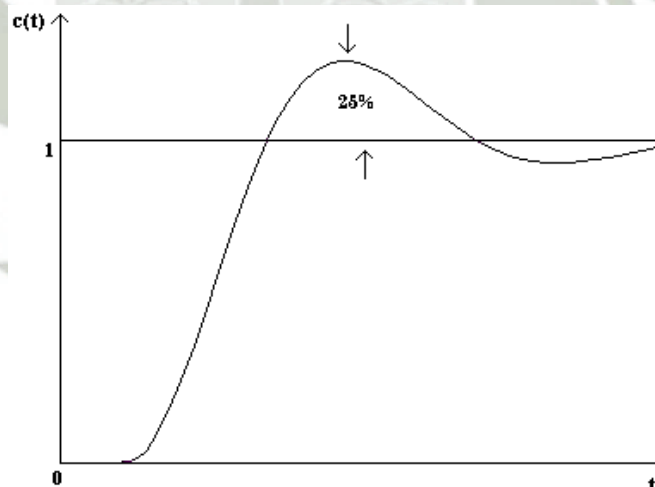


Figura 2.14. Sobreimpulso

⁴⁷ Ingeniería de Control Moderno 4ta Edic" Ogata, Katsuniko. Pág. 683

- **Primero Método:** En el primer método, la respuesta de la planta a una entrada escalón unitario se obtiene de manera experimental, como se muestra en la Figura 2.15⁴⁸. Si la planta no contiene integradores ni polos dominantes complejos conjugados, la curva de respuesta escalón unitario puede tener forma de S, como se observa en la Figura 2.16. Si la respuesta no exhibe una curva con forma de S, este método no es pertinente. Tales curvas de respuesta escalón se generan experimentalmente o a partir de una simulación dinámica de la planta.

La curva con forma de S se caracteriza por dos parámetros: el tiempo de retardo L y la constante de tiempo T. El tiempo de retardo y la constante de tiempo se determinan dibujando una recta tangente en el punto de inflexión de la curva con forma de S y determinando las intersecciones de esta tangente con el eje del tiempo y la línea $c(t)=K$, como se aprecia en la Figura 2.17⁴⁹.

En este caso, la función de transferencia $C(s)/U(s)$ se aproxima mediante un sistema de primer orden con un retardo de transporte del modo siguiente:

$$\frac{C(s)}{U(s)} = \frac{K e^{-Ls}}{Ts + 1}$$

⁴⁸ Ingeniería de Control Moderno 4ta Edic" Ogata, Katsuniko. Pág. 683

⁴⁹ Ingeniería de Control Moderno 4ta Edic" Ogata, Katsuniko. Pág. 683

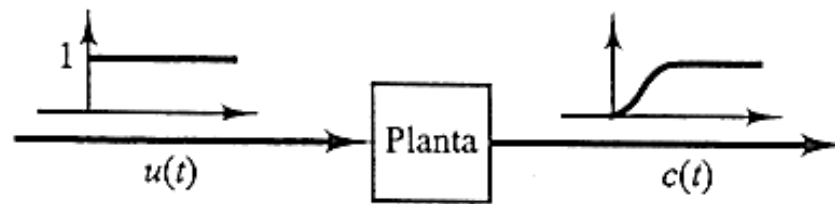


Figura 2.15. Respuesta escalón unitario de una planta.

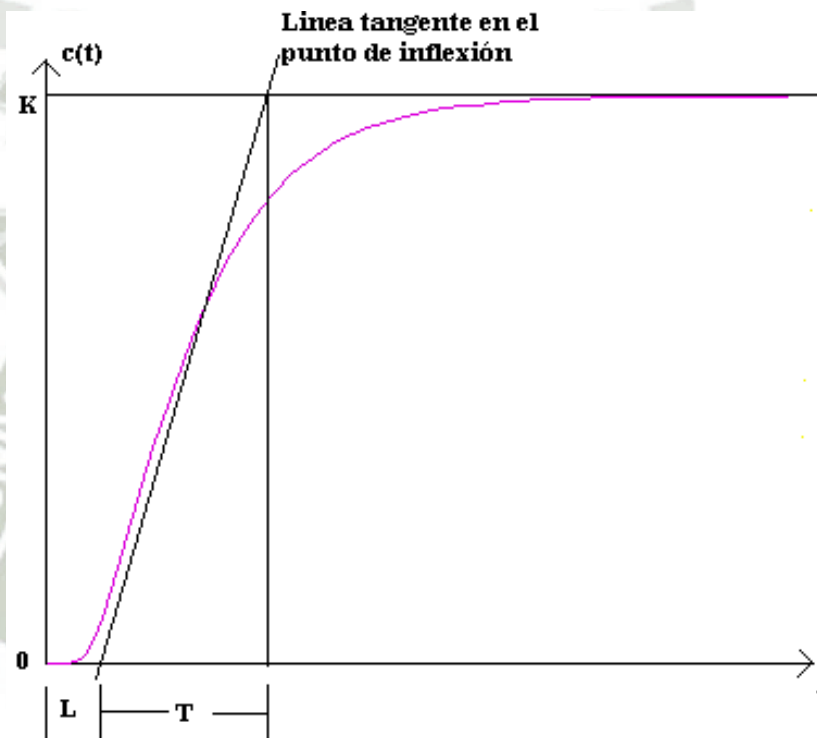


Figura 2.16. Curva de respuesta en forma de S.

Ziegler y Nichols sugirieron establecer los valores de K_p , T_i y T_d de acuerdo con la fórmula que aparece en la siguiente Tabla 2.1⁵⁰.

⁵⁰ Ingeniería de Control Moderno 4ta Edic" Ogata, Katsuniko. Pág. 684

Tabla 2.1. Regla de sintonía de Ziegler-Nichols basada en la repuesta escalón de la planta (Primer Método)

Tipo de controlador	K_p	T_i	T_d
P	$\frac{T}{L}$	∞	0
PI	$0.9 \frac{T}{L}$	$\frac{L}{0.3}$	0
PID	$1.2 \frac{T}{L}$	$2L$	$0.5L$

Observe que el controlador PID sintonizado mediante el primer método de las reglas de Ziegler-Nichols produce:

$$G_c s = K_p \left(1 + \frac{1}{T_i s} + T_d s \right)$$

$$G_c s = 1.2 \frac{T}{L} \left(1 + \frac{1}{2Ls} + 0.5Ls \right)$$

$$G_c s = 0.6T \frac{s + \frac{1}{2}}{s}$$

Por lo tanto el controlador PID tiene un polo en el origen y un cero $s = -\frac{1}{L}$

- **Segundo Método:** En el segundo método, primero establecemos $T_i = \infty$ y $T_d = 0$. Usando sólo la acción de control proporcional (véase la

Figura 2.17⁵¹, se incrementa K_p de 0 a un valor crítico K_{cr} en donde la salida exhiba primero oscilaciones sostenidas. Si la salida no presenta oscilaciones sostenidas para cualquier valor que pueda tomar K_p , no se aplica este método. Por tanto, la ganancia crítica K_{cr} y el periodo P_{cr} correspondiente se determinan experimentalmente. Ziegler-Nichols sugirieron que se establecieran los valores de los parámetros K_p , T_i y T_d de acuerdo con la fórmula que aparece en la siguiente Tabla 2.2⁵².

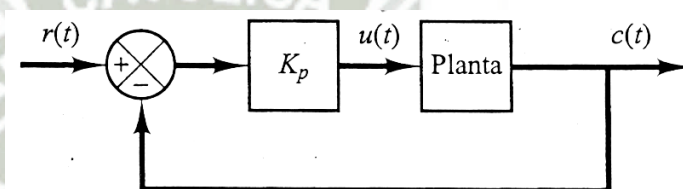


Figura 2.17. Sistema de lazo cerrado con un controlador proporcional

Tabla 2.2. Regla de sintonía de Ziegler-Nichols basada en la ganancia crítica K_{cr} y periodo crítico P_{cr} (Segundo Método)

Tipo de controlador	K_p	T_i	T_d
P	$0.5K_{cr}$	∞	0
PI	$0.45K_{cr}$	$\frac{1}{1.2}P_{cr}$	0
PID	$0.6K_{cr}$	$0.5P_{cr}$	$0.125P_{cr}$

⁵¹ Ingeniería de Control Moderno 4ta Edic" Ogata, Katsuniko. Pág. 684

⁵² Ingeniería de Control Moderno 4ta Edic" Ogata, Katsuniko. Pág. 685

Se debe observar que el controlador PID sintonizado mediante el segundo método de las reglas de Ziegler-Nichols produce:

$$G_c(s) = K_p \left(1 + \frac{1}{T_i s} + T_d s \right)$$

$$G_c(s) = 0.6K_{cr} \left(1 + \frac{1}{0.5P_{cr}s} + 0.125P_{cr}s \right)$$

$$G_c(s) = 0.075K_{cr}P_{cr} \frac{s + \frac{4}{P_{cr}}}{s^2}$$

Por lo tanto, el controlador PID tiene un polo en el origen y cero doble en

$$s = -\frac{4}{P_{cr}}$$

Si se conoce la función de transferencia de la planta, se calcula la respuesta escalón unitario o la ganancia crítica K_{cr} y el periodo crítico P_{cr} . Sin embargo, la utilidad real de las reglas de sintonización de Ziegler-Nichols se vuelve evidente cuando no se conoce la dinámica de la planta, por lo que no se cuenta con enfoques analíticos o gráficos para el diseño de controladores.

En general, para aquellas plantas con una dinámica complicada y sin integradores, se han aplicado las reglas de sintonización de Ziegler-Nichols. Sin embargo, si la planta tiene un integrador, en algunos casos estas reglas no son pertinentes.

Para ilustrar una situación en las que las reglas de Ziegler-Nichols no se aplican, consideremos el caso donde un sistema de control con realimentación

unitaria tiene una planta cuya función de transferencia es de la siguiente manera:

$$G_{(s)} = \frac{s + 2}{s} \frac{s + 3}{s + 1} \frac{1}{s + 5}$$

2.7. Consideraciones Finales

Todo lo anterior mostrado nos sirve para saber qué elementos componen una Grúa Puente, tener el conocimiento de las posibles soluciones a diversos problemas como de carga/descargar y de transporte de cargas pesadas, así también como los riesgos que con lleva el operar una Grúa Puente o simple estar parado o trabajado cerca de una Grúa Puente así mismo también para comprender y saber que elementos como ponen un sistema de control, lo que nos permite poder realizar análisis de todos los sistemas lineales y no lineales, y también con sistemas dinámicos.

La lógica nos proporciona un tipo de control que más se adapta mejor al mundo real en el que vivimos ya que nos permite funcionar y comprender con nuestras propias experiencias y poder plasmarlas en las reglas difusas. El cual nos permite resolver problemas varias tan solo con la experiencia aprendida a lo largo de los años de trabajo y lo cual se explica con mayor detenimiento en el siguiente capítulo III.

CAPÍTULO III

CONTROLADORES DIFUSOS

3.1. Control con Lógica Difusa

3.1.1. Conceptos Básicos⁵³

La lógica difusa ha cobrado una fama grande por la variedad de sus aplicaciones, las cuales van desde el control de complejos procesos industriales, hasta el diseño de dispositivos artificiales de deducción automática, pasando por la construcción de artefactos electrónicos de uso doméstico y de entretenimiento, así como también de sistemas de diagnóstico. De hecho, desde hace ya, al menos, década y media, la expedición de patentes industriales de mecanismos basados en la lógica difusa tiene un crecimiento sumamente rápido en todas las naciones industrializadas del orbe. Se ha considerado de manera general que el concepto de lógica difusa apareció en 1965, en la Universidad de California en Berkeley, introducido por Lotfi A. Zadeh . Las lógicas difusas, pues de hecho hay que hablar de ellas en plural, son esencialmente lógicas multivaluadas que extienden a las lógicas clásicas. Estas últimas imponen a sus enunciados únicamente valores falso o verdadero. Bien que éstas han modelado satisfactoriamente a una gran parte del razonamiento natural, es cierto que el razonamiento humano utiliza valores de verdad que no necesariamente son tan deterministas. Por ejemplo, al calificar que el cielo es azul uno está tentado a graduar qué tan azul, en efecto, es el cielo, e igualmente, si un vehículo se mueve rápido, también se está obligado a considerar qué tan rápido es

⁵³ Introducción a la Lógica Difusa, Tomas Arredondo, pág. 4

el vehículo, aunque esto último no implique necesariamente cuantificar la velocidad del vehículo con toda precisión. Las lógicas difusas procuran crear aproximaciones matemáticas en la resolución de ciertos tipos de problemas. Pretenden producir resultados exactos a partir de datos imprecisos, por lo cual son particularmente útiles en aplicaciones electrónicas o computacionales. El adjetivo difuso aplicado a ellas se debe a que los valores de verdad no-deterministas utilizados en ellas tienen, por lo general, una connotación de incertidumbre. Un vaso medio lleno, independientemente de que también esté medio vacío, no está lleno completamente ni está vacío completamente. Qué tan lleno puede estar es un elemento de incertidumbre, es decir, de difusidad, entendida esta última como una propiedad de indeterminismo. Ahora bien, los valores de verdad asumidos por enunciados aunque no son deterministas, no necesariamente son desconocidos. Por otra parte, desde un punto de vista optimista, lo difuso puede entenderse como la posibilidad de asignar más valores de verdad a los enunciados que los clásicos "falso" o "verdadero". Así pues, reiteramos, las lógicas difusas son tipos especiales de lógicas multivaluadas. Las lógicas difusas han tenido aplicaciones de suma relevancia en el procesamiento electrónico de datos. En determinadas áreas de conocimiento, a sus enunciados se les asocia valores de verdad que son grados de veracidad o falsedad, mucho más amplios que los meros "verdadero" y "falso". En un sistema deductivo se distinguen enunciados de entrada y enunciados de salida. El objetivo de todo sistema manejador de una lógica difusa es describir los grados de los enunciados de salida en términos de los de entrada. Más aún, algunos sistemas son capaces de refinar los grados de veracidad de los enunciados de salida conforme se refinan los de los de entrada. Por estas propiedades es que ciertos sistemas de lógica difusa aparentan una labor de aprendizaje, y son excelentes

mecanismos de control de procesos. Desde el punto de vista tecnológico, las lógicas difusas se encuadran en el área de la llamada Inteligencia Artificial y han dado origen a sistemas expertos de tipo difuso y a sistemas de control automático. En esta presentación haremos énfasis en el carácter multivaluado de las lógicas difusas. Introduciremos primero la noción de conjunto difuso, y las operaciones usuales en ese tipo de conjuntos. Inmediatamente después, presentaremos ciertos tipos de cálculos proposicionales de tipo difuso y de cuantificación difusa.

3.1.2. Funciones de pertenencia⁵⁴

Una función de pertenencia de un conjunto borroso A sobre un universo de discurso X es de la forma $\mu_A: X \rightarrow [0,1]$, donde a cada elemento de X le corresponde un valor entre 0 y 1. Este valor, llamado valor de pertenencia o grado de pertenencia, representa el grado en el que el elemento de X pertenece al conjunto borroso A .

Las funciones de pertenencia nos permiten representar gráficamente un conjunto borroso. En el eje "x" (abscisas) se representa el universo de discurso, mientras que en el eje "y" (ordenadas) se sitúan los grados de pertenencia en el intervalo $[0,1]$.

Para construir funciones de pertenencia se suelen utilizar funciones sencillas, ya que al estar definiendo conceptos borrosos el uso de funciones complejas no aporta mayor precisión.

A continuación, se presentan las funciones de pertenencia que utilizaremos:

⁵⁴ Introducción a la Lógica Difusa, Tomas Arredondo, pág. 16

- **Función de Pertenencia Trapezoidal**

Definida por sus límites inferior a , superior d , y los límites de soporte inferior b y superior c , tal que $a < b < c < d$

$$\mu_A(x) = \begin{cases} 0, & \text{si } (x < a) \text{ ó } (x > d) \\ \frac{x-a}{b-a}, & \text{si } a \leq x \leq b \\ 1, & \text{si } b \leq x \leq c \\ \frac{d-x}{d-c}, & \text{si } c \leq x \leq d \end{cases}$$

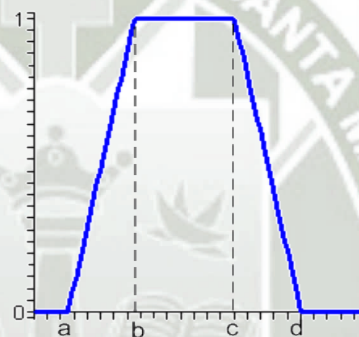


Figura 3.1. Función de Pertenencia Trapezoidal

- **Función de Pertenencia Triangular:**

Viene definida por un límite inferior a , un límite superior b , y un valor m tal que $a < m < b$. La función no tiene porqué ser simétrica.

$$\mu_A(x) = \begin{cases} 0, & \text{si } x \leq a \\ \frac{x-a}{m-a}, & \text{si } a < x \leq m \\ \frac{b-x}{b-m}, & \text{si } m < x < b \\ 0, & \text{si } x \geq b \end{cases}$$

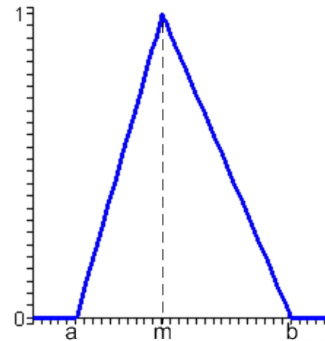


Figura 3.2. Función de Pertenencia Triangular

- **Función de Pertenencia Pseudo-Exponencial:**

Definida por el valor medio m y el parámetro $k > 1$.
Cuanto mayor es el valor de k , el crecimiento es más rápido y la campana es más estrecha

$$\mu_A(x) = \frac{1}{1 + k(x - m)^2}$$

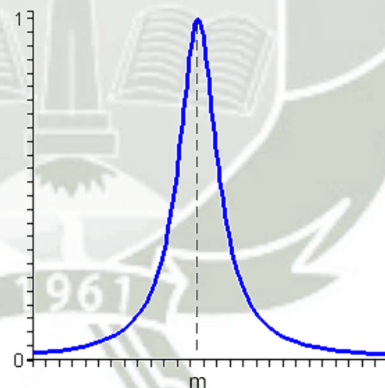


Figura 3.3. Función de Pertenencia Pseudo-Exponencial

- **Función de Pertenencia Gaussiana:**

Viene definida por su valor medio m y una desviación estándar $k > 0$. Se cumple que cuanto menor es k , más estrecha es la “campana”.

$$\mu_A(x) = e^{-\frac{(x-m)^2}{2k^2}}$$

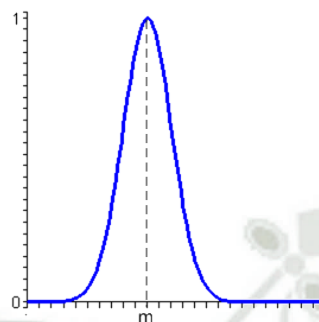


Figura 3.4. Función de Pertenencia Gaussiana.

- **Función de Pertenencia Sigmoide**

Definida por sus límites inferior **a**, superior **b** y el valor **m** o punto de inflexión, tales que **a < m < b**. El crecimiento es más lento cuanto mayor sea la distancia **a-b**. Para el caso concreto de **m=(a+b)/2**, que es lo usual

$$\mu_A(x) = \begin{cases} 0, & \text{si } x \leq a \\ 2 \left[\frac{x-a}{b-a} \right]^2, & \text{si } a < x \leq m \\ 1 - 2 \left[\frac{x-b}{b-a} \right]^2, & \text{si } m < x < b \\ 1, & \text{si } x \geq b \end{cases}$$

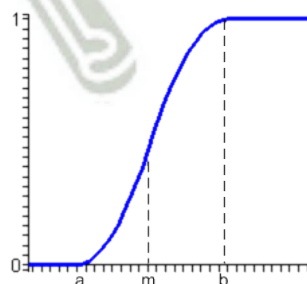


Figura 3.5. Función de Pertenencia Sigmoidea

3.1.3. Tipos De Sistemas de Lógica Difusa

Los sistemas de lógica difusa tiene una estrecha relación con los conceptos difusos tales como conjuntos difusos, variables lingüísticas y demás. Los más populares sistemas de lógica difusa que se encuentra en la literatura se encuentran en unos de los siguientes tipos: sistemas difusos tipo Mamdani (con fuzzificador y defuzzificador) Sistemas difusos tipo Takagi-Sugeno

3.1.3.1. Sistema Tipo Mamdani

En la Figura 3.6., se muestra la configuración básica de un sistema tipo Mamdani

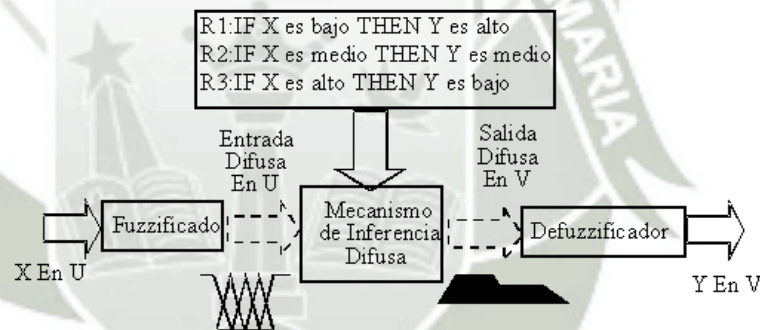


Figura 3.6. Sistema Difuso Mamdani, Procesamiento General

En un sistema difuso tipo Mamdani se distinguen las siguientes partes:

- **Fuzzificador**

La entrada de un sistema de lógica difusa tipo Mamdani normalmente es un valor numérico proveniente, por ejemplo, de un sensor; para que

este valor pueda ser procesado por el sistema difuso se hace necesario convertirlo a un "lenguaje" que el mecanismo de inferencia pueda procesar. Esta es la función del fuzzificador, que toma los valores numéricos provenientes del exterior y los convierte en valores "difusos" que pueden ser procesados por el mecanismo de inferencia. Estos valores difusos son los niveles de pertenencia de los valores de entrada a los diferentes conjuntos difusos en los cuales se ha dividido el universo de discurso de las diferentes variables de entrada al sistema.

- **Mecanismo de Inferencia Difusa**

Teniendo los diferentes niveles de pertenencia arrojados por el fuzzificador, los mismos deben ser procesados para generar una salida difusa. La tarea del sistema de inferencia es tomar los niveles de pertenencia y apoyado en la base de reglas generar la salida del sistema difuso.

- **Base de Reglas Difusas**

La base de reglas son la manera que tiene el sistema difuso de guardar el conocimiento lingüístico que le permiten resolver el problema para el cual ha sido diseñado. Estas reglas son del tipo IF-THEN. Una regla de la base de reglas o base de conocimiento tiene dos partes, el antecedente y la conclusión como se observa en la siguiente Figura 3.7:

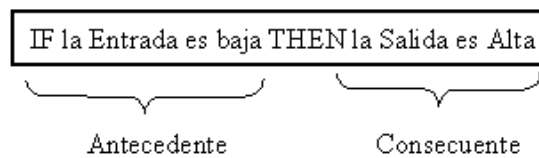


Figura 3.7. Reglas Difusas Tipo Mamdani

En un sistema difuso tipo Mamdani tanto el antecedente como el consecuente de las reglas están dados por expresiones lingüísticas.

- **Defuzzificador**

La salida que genera el mecanismo de inferencia es una salida difusa, lo cual significa que no puede ser interpretada por un elemento externo (por ejemplo un controlador) que solo manipule información numérica.

Para lograr que la salida del sistema difuso pueda ser interpretada por elementos que solo procesen información numérica, hay que convertir la salida difusa del mecanismos de inferencia; este proceso lo realiza el fuzzificador.

La salida del mecanismo de inferencia es un conjunto difuso resultante, para generar la salida numérica a partir de estos conjuntos existen varias opciones como el Centro de Gravedad, los Centros Promediados entre otros.

- **Centro de Gravedad**

$$Centroide = \frac{\sum_{i=0}^n f(x)_i x_i}{\sum_{i=0}^n f(x)_i}$$

- **Centros Promediados**

$$y = \frac{i b_i \mu_{premisas}(i)}{i \mu_{premisas}(i)}$$

En el siguiente gráfico se muestra de manera más detallada como es el funcionamiento de un sistema difuso Mamdani (Figura 3.8)

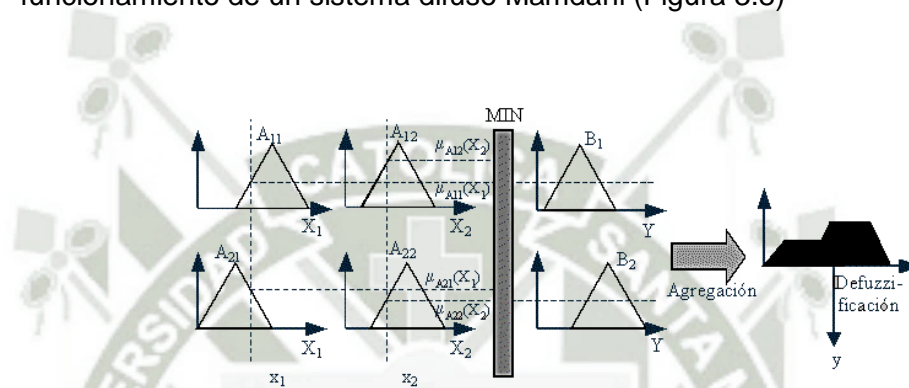


Figura 3.8. Sistema Difuso Mamdani, Procesamiento Detallado

3.1.3.2. Sistema Tipo Sugeno

Otra alternativa de procesamiento en los sistemas difusos fue la propuesta por Sugeno en los sistemas difusos que llevan su nombre. (Figura 3.9)

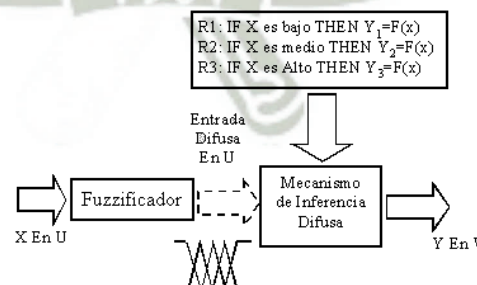


Figura 3.9. Sistema Difuso Sugeno, Procesamiento General

En los sistemas difusos Sugeno se distinguen las siguientes partes:

- **Fuzzificador**

Realiza la misma función que en los sistemas Mamdani explicados anteriormente

- **Mecanismo de inferencia difusa**

Realiza la misma función que en los sistemas Mamdani explicados anteriormente

- **Base de Reglas Difusas**

Las reglas de la base de conocimiento de un sistema Sugeno es diferente a las de los sistemas Mamdani pues el consecuente de estas reglas ya no es una etiqueta lingüística sino que es una función de la entrada que tenga el sistema en un momento dado, esto se ilustra en la Figura 3.10 a continuación:



Figura. 3.10. Reglas Difusas Tipo Sugeno

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 defuzzificación.

Para calcular la salida del sistema difuso se ponderan los diferentes consecuentes teniendo en cuenta el valor que se activó el antecedente de cada una de las reglas, para un sistema con dos reglas la salida del sistema difuso sería:

$$y = \frac{w_1 y_1 + w_2 y_2}{w_1 + w_2}$$

- **Cálculo de la Salida de un Sistema Difuso Sugeno**

En este caso:

$$y_1 = f_1 x$$

$$y_2 = f_2 x$$

Serían las funciones que permiten calcular el consecuente de cada una de las dos reglas implicadas.

En el siguiente gráfico se muestra de manera más detallada como es el funcionamiento de un sistema difuso Sugeno. (Figura 3.11)

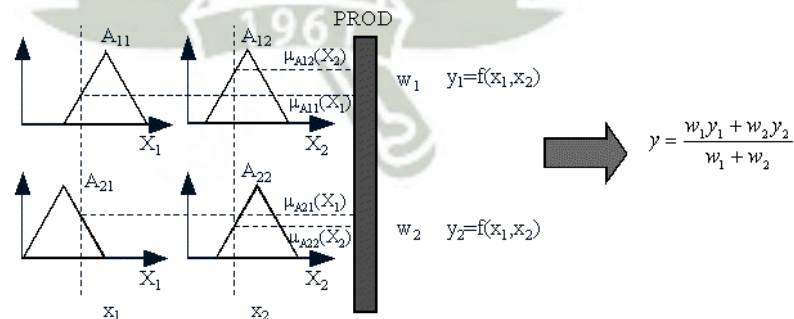


Figura 3.11. Sistema Difuso Sugeno, Procesamiento Detallado

3.1.4. Procedimiento de Diseño de un Control con Lógica Difusa

El procedimiento de diseño involucra los pasos siguientes⁵⁵:

1. Formulación del problema a resolver.
2. Determinación del modelo difuso a controlar.
3. Fuzzificación.
4. Diseño de la base de reglas.
5. Calculo de las inferencias
6. Desfuzzificación.
7. Simulación del sistema difuso.
8. Implementación del sistema de control difuso.

3.2. Controlador Proporcional Integral Derivativo (PID) Difuso

Para poder entender cómo se debe de implementar un controlador PID difuso, se procede a analizar las propuestas existentes en la literatura. En base a aspectos excelentes de un controlar difuso como son: características de las entradas, características de las salidas, características de las funciones de pertenencia y las características de las reglas difusas. Lo que explicamos a seguir:

3.2.1. Entradas de un controlador PID difuso

Teniendo en cuenta toda la teoría actual sobre controles difusos se ve el tipo de entradas que llegan al controlador difuso y las cuales son:

- **Con una entradas:** El error (e).

La señal de error es el control esencial y fundamental componente en el control PID. Por lo tanto, mediante el uso de esta entrada, en este caso

⁵⁵ Antología Lógica Difusa Aplicada al Control. Sulla Torres, Raúl Ricardo. Pág. 4

como se trata de una sola entrada de un sistema difuso de control PID.

Tenemos como ejemplo a la Figura 3.12⁵⁶

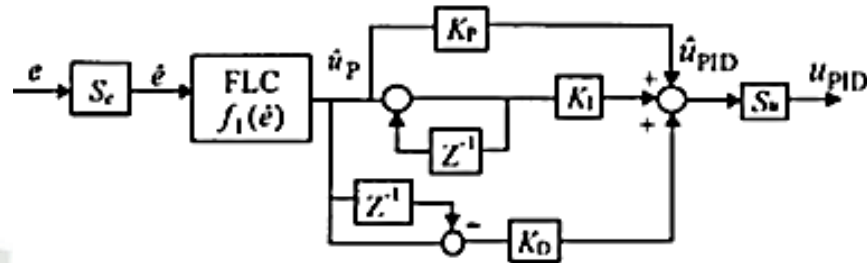


Figura 3.12. Una entrada PID difuso

- **Con dos entradas:** El error (e) y la derivada del error ($d.e$).

Los elementos de control de dos entradas seleccionados tienen las entradas (e , $d.e$) que forman un control PID difuso. Se puede obtener opciones de combinación PI y PD que se muestra en la Figura. 3.13, un controlador de dos entradas PID difuso pueden ser formado. Tenemos como ejemplo a la Figura 3.13⁵⁷

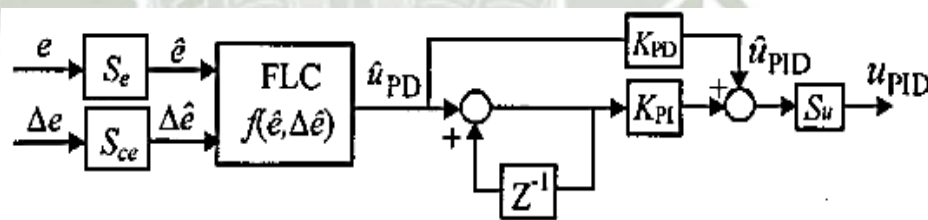


Figura 3.13. Dos entradas PID difuso

- **Con tres entradas:** El error (e), la derivada del error ($d.e$) y la segunda derivada del error ($d^2.e$).

⁵⁶ Analysis of Direct Action Fuzzy PID Controller Structures George K. I. Mann, Bao-Gang Hu, Member, IEEE, and Raymond G. Gosine, Member, IEEE. Pág. 4

⁵⁷ Analysis of Direct Action Fuzzy PID Controller Structures George K. I. Mann, Bao-Gang Hu, Member, IEEE, and Raymond G. Gosine, Member, IEEE. Pág. 4

La idea basada en el conocimiento hasta ahora se puede seleccionar tres de entrada correspondientes a generar un control PID. Tenemos como ejemplo a las Figura 3.14 y Figura 3.15⁵⁸

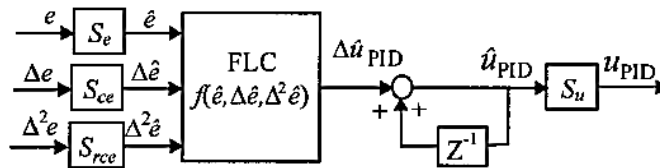


Figura 3.14. Tres entradas PID difuso (tipo I)

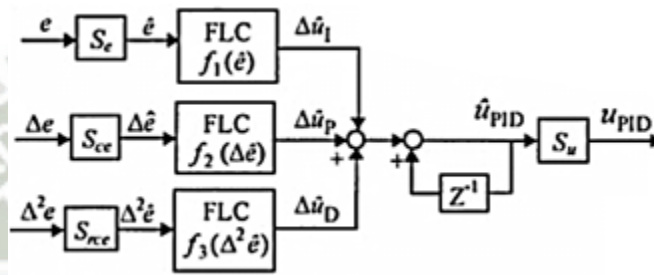


Figura 3.15. Tres entradas PID difuso (tipo II)

Como hemos podido apreciar en ninguno de los casos anteriores, se ve que la integral no forma parte de las señales de entrada que ingresan al bloque difuso, por lo que con la tabla a continuación lo explicaremos mejor.

	Kp aumenta	Ti aumenta	Td aumenta
Estabilidad	Se reduce	Aumenta	Aumenta
Velocidad	Aumenta	Disminuye	Aumenta
Error est. Estacionario	No eliminado	No eliminado	No eliminado

Entonces teniendo la anterior tabla podemos apreciar que la integral produce los mismo efectos que la derivada, entonces se tendrían dos entradas que realizan el

⁵⁸ Analysis of Direct Action Fuzzy PID Controller Structures George K. I. Mann, Bao-Gang Hu, Member, IEEE, and Raymond G. Gosine, Member, IEEE. Pág. 3

mismo trabajo lo cual produce una cierta complicación con las variables y es por eso que no se toma en cuenta como entrada, sino que se utiliza de mejor manera ubicándola después de la salida del bloque difuso, y es ahí donde cumple la función de suavizar la señal de salida, lo que proporciona un mejor control del proceso.

3.2.2. Salidas de un controlador PID difuso

Como se puede apreciar en cuanto a las salidas, obviamente se relacionan con el tipo entradas, lo que ocasiona varias maneras de obtener la salida desea del bloque difuso como ejemplo cuando se trata de una sola entrada, la salida del bloque difuso se divide en 3, una que es directa y solamente se multiplica por una constante proporcional, la segunda tiene un bloque que retrasa la señal y la suma y a su vez es multiplicada por una constante integral y de igual manera con la tercera, solo que con la diferencia de que el retraso de la señal se resta y se multiplica por una constante derivativa. Al final de todo este proceso se obtiene la señal desea que ingrese al bloque de planta (Figura 3.12.).

Cuando se tiene dos entradas al bloque difuso, las salidas de este son dos, una que directa y se multiplica por las constaste proporcional y derivada, la segunda salida se multiplica por una combinación de constaste proporcional e integral y que también cuanta con un retraso de la señal que se suma, lo que esto al final nos da la señal desea para que ingrese al bloque de planta (Figura 3.13).

Y por último cuando se tiene tres entradas, en un caso no se tiene que realizar ningún proceso adicional para obtener la señal desea solo un retrasado de la señal para suavizarla y esa es la señal que va directo al bloque difuso (Figura 3.14). Y en otro caso cada señal de entradas tiene su propio bloque difuso, que luego se

suman y se le es aplicado un retrasado para suavizar la señal que ha de entrar al bloque de planta (Figura 3.15).

3.2.3. Funciones de Pertenencia

Como ya se mencionó y dado el carácter investigativo de este proyecto, se procedió a analizar las funciones de membrecía más comúnmente utilizadas en controladores PID difusos. Después de este proceso de análisis se concluyó que la mejor opción de función de pertenencia es la función triangular pues reduce el tiempo de operación, aumentando la velocidad de respuesta y además las funciones están solapadas en un 50% lo cual nos brinda una banda constante para entrada y salida de cada tipo de controlador. (Figura 3.16)⁵⁹

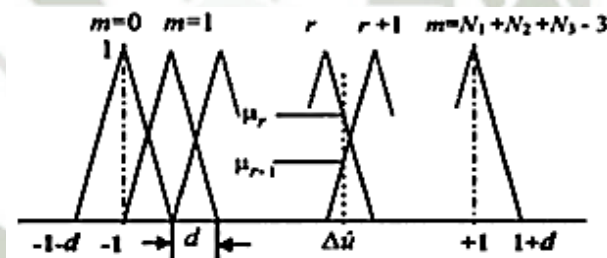


Figura 3.16. Distribución de las funciones de membrecía (todo tipo de controladores)

3.2.4. Reglas difusas

Para poder comprender aún mejor la lógica difusa y su aplicación debemos analizar el comportamiento de las reglas difusas y la lógica con la cual se realizan las mismas, en lo que encontramos que varios autores tienen la misma base para realizar sus reglas, teniendo como guía la Figura 3.17, en la que los autores

⁵⁹Analysis of Direct Action Fuzzy PID Controller Structures George K. I. Mann, Bao-Gang Hu, Member, IEEE, and Raymond G. Gosine, Member, IEEE. Pág. 6

Godjevac⁶⁰, Xionh Li y Gatland⁶¹ proponen, teniendo una diagonal que va de arriba a abajo y de izquierda a derecha, en la que se puede apreciar que toda la parte superior de la diagonal es positiva y que la va en aumento conforme se aleja de la diagonal y que toda la parte de debajo de la diagonal es negativa y se hace más negativa con forme se aleja de la diagonal. Lo que significa que es la acción de que debe de tomar el controlador en cualquiera sea el área en que se encuentre.

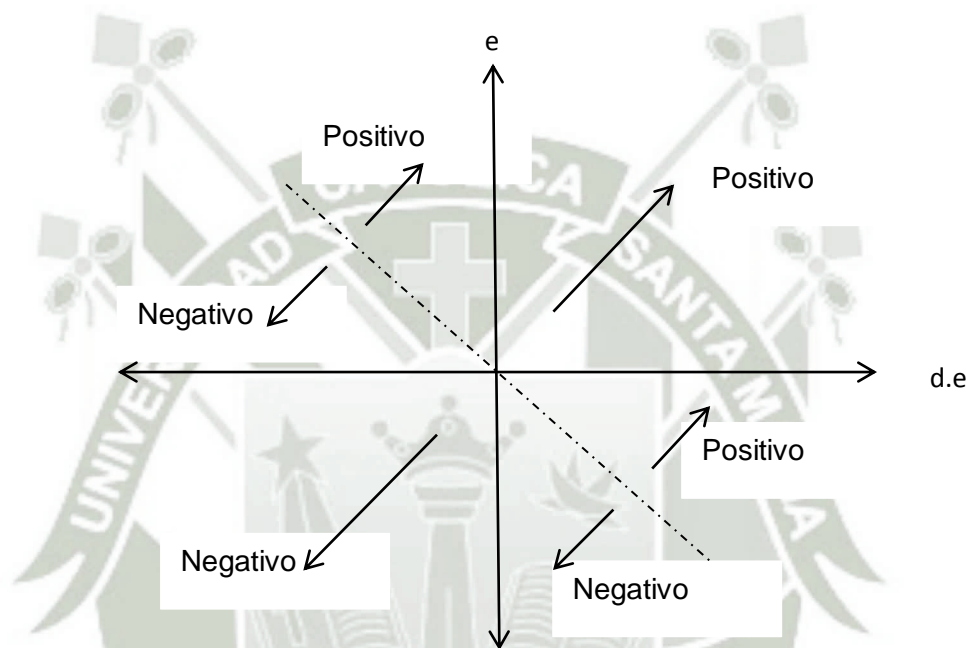


Figura 3.17. Comportamiento de las reglas difusas

También tenemos más propuestas Ngamroo⁶², Weibin y Qingjian⁶³ y por ultimo Yang et al⁶⁴ de igual manera que el apartado anterior se aprecia el mismo tipo de grafico (Figura 3.18) con la única diferencia que se aprecia que toda la parte

⁶⁰ Comparison between PID and fuzzy control. Jelena Godjevac

⁶¹ Conventional Fuzzy Control and Its Enhancement. Han-Xiong Li and H. B. Gatland

⁶² Application of electrolyzer to alleviate power fluctuation in a stand alone microgrid based on an optimal fuzzy PID control. Issarachai Ngamroo

⁶³ Base don PLC temperatura PID – Fuzzy control system design and simulation. Cao Weibin and Meng Qingjian.

⁶⁴ Modeling and Fuzzy-PID Control of Hydraulic-Rolls Coupled System in Cold Rolling Mills. Xu Yang, Bo Wen and Chao-nan Tong.

superior de la diagonal es negativa y que se hace más negativa conforme se va separando de la diagonal y en la parte de abajo de la diagonal es positiva y se va haciendo cada vez más positiva conforme se va separando de la diagonal, lo cual significa que es la acción que debe de tomar el controlador en cualquiera sea el área que se encuentre .

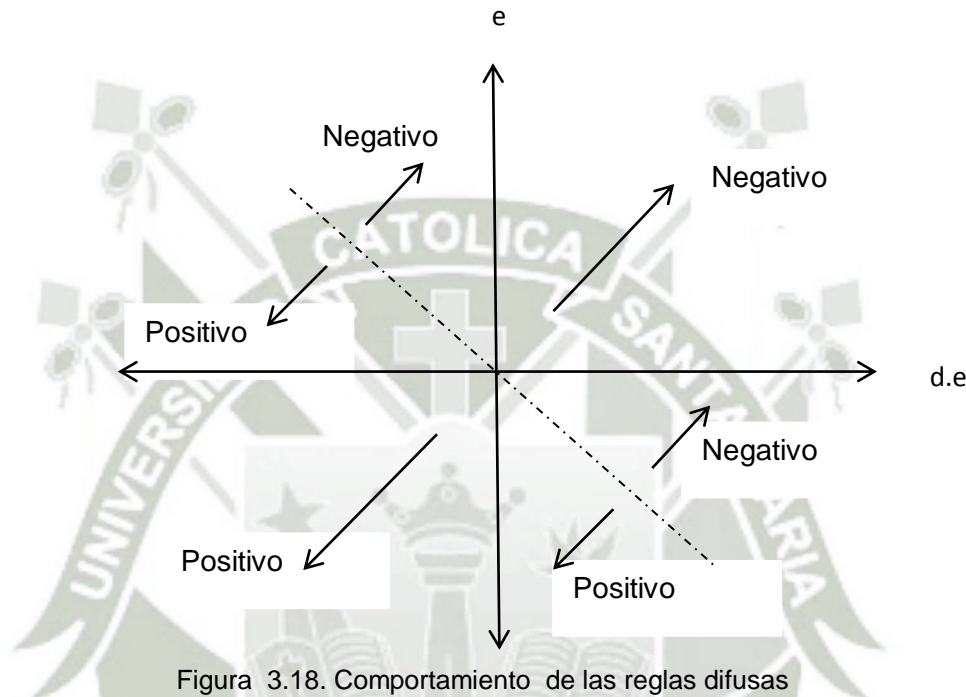


Figura 3.18. Comportamiento de las reglas difusas

Ahora como se ha visto en los dos anteriores postulados, hay una diferencia en cuanto a los signos de la acción que ha de tomar el controlador, como se aprecia están invertidos, eso no quiere decir que uno esté bien y el otro mal, sino está relacionado al tipo de aplicación en la que se encuentre el controlador como ejemplo en la propuesta de Weibin y Qingjian en la que ellos aplican el controlador en un control de temperatura, como se puede apreciar en la gráfica (3.18), que cuando el error es muy negativo lo que quiere decir que la temperatura es muy baja, con lo que se necesita una acción muy positiva lo mismo decir que se requiere que aumente la temperatura, es por eso las dos graficas tiene esa particular diferencia.

Para todo esto podemos observar una excepción en la propuesta de Xiong Li y Gatland que si bien mantienen la parte superior es positiva y la inferior negativa, se puede observar también que presenta áreas que son estables como la área superior izquierda y la área inferior derecha que son positiva y negativa respectivamente, en estas dos zonas se mantiene la acción tomada anteriormente o simplemente no se realiza ninguna acción. Lo que significa que este tipo de conjunto de reglas hacen que la acción del controlador será un poco conservadora (Figura 3.19)

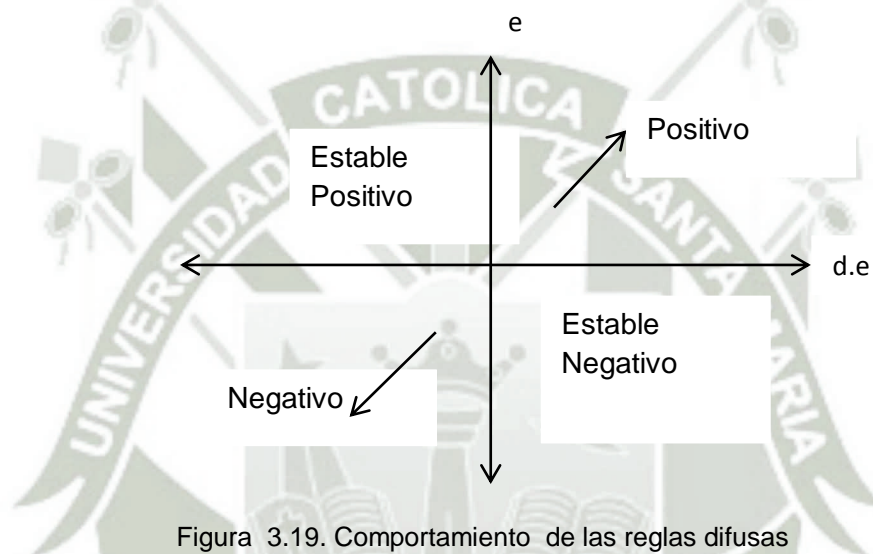


Figura 3.19. Comportamiento de las reglas difusas

3.3. Consideraciones finales

Todo lo que hemos aprendido en este capítulo, nos ayuda a entender como realmente funciona un controlador difuso y todas las partes de que está compuesto, como la forma en que se procesan las entradas y salidas para dar un respuesta, la manera en que se escogen las funciones de membresía, la forma de las funciones de membresía, la manera en que se desarrollan las reglas y la lógica en que se construye ese conjunto de reglas y para qué tipo de problema en específico.

Todo esto nos proporciona las herramientas necesarias para poder realizar nuestro propio controlador difuso, teniendo todas las características de un controlador de alto nivel. En este sentido en el siguiente capítulo proponemos un controlador PID difuso para una Grúa Puente tomando como base el análisis realizado en este capítulo.



CAPÍTULO IV

PROPUESTA DE CONTROL

En este capítulo procedemos a realizar los cálculos para poder obtener la función de transferencia a partir del diagrama de cuerpo libre de la Grúa Puentes, teniendo en cuenta todas las fuerzas que intervienen en el movimiento de esta, también procedemos a definir las funciones de membresía como su forma y tamaño, el rango de cada entrada y salida, y como último como las reglas difusas

4.1. Modelado

En la Figura 4.1 siguiente se representa el diagrama de cuerpo libre para una Grúa Puentes

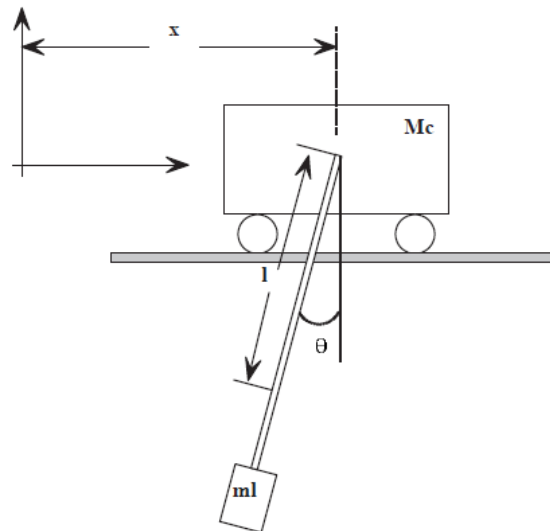


Figura 4.1. Diagrama de Cuerpo Libre de una Grúa Puentes

4.1.1. Diagrama de cuerpo libre del carro

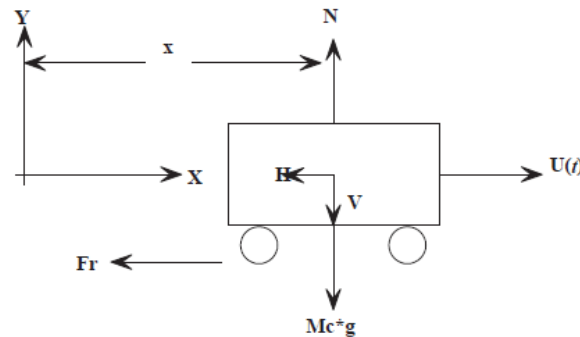


Figura 4.2. Diagrama de cuerpo libre

De aquí en adelante podemos encontrar las fórmulas que definen el sistema y que gobiernan los estados.

Realizando suma de fuerzas en x en el carro se tiene:

$$+\rightarrow Fx = ma$$

$$U_t - Fr - H = Mc \cdot x \quad (1)$$

Donde Fr es la fricción de las ruedas del carro contra el riel

$$Fr = fc \cdot x \quad (2)$$

4.1.2. Diagrama de cuerpo libre de la carga

Se toma en esta parte solo el diagrama de fuerzas que se encuentran aplicadas solo a la carga, y lo que podemos observar a continuación:

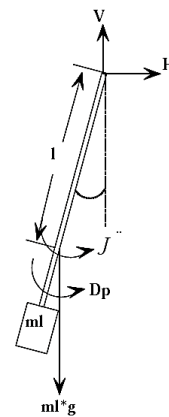


Figura 4.3. Diagrama de cuerpo libre de la carga

Realizado suma de fuerzas en x en la carga se tiene:

$$+\rightarrow F_x = ma$$

$$H = ml \frac{d^2}{dt^2} (x - l \sin \theta) \quad (3)$$

Realizando suma de fuerzas en y en la carga se tiene:

$$+\uparrow F_y = ma$$

$$V - ml \cdot g = ml \frac{d^2}{dt^2} (l \cos \theta) \quad (4)$$

Dónde:

$$\frac{d^2}{dt^2} x - l \sin \theta = x - l \theta \cos \theta + l \theta^2 \sin \theta$$

$$\frac{d^2}{dt^2} l \cos \theta = -l \theta \sin \theta + l \theta^2 \cos \theta$$

Haciendo suma de momentos en el centro de masas de la carga se tiene:

$$M = 0$$

$$J\theta + Dp - Hl\cos\theta + Vlsen\theta = 0 \quad (5)$$

Donde Dp es la fricción rotacional, la que se define de esta manera:

$$Dp = fp\theta$$

Reemplazando las anteriores formulas en (1) y (5) y lo que tenemos es:

$$U t - Fr = Mcx + ml(x - l\theta\cos\theta + l\theta^2\sin\theta)$$

$$U t = Mc + ml x - ml * l\theta\cos\theta + ml * l\theta^2\sin\theta + Fr \quad (6)$$

$$J\theta + Dp - Hl\cos\theta + Vlsen\theta = 0$$

$$J\theta + fp\theta - ml * l\cos\theta x - l\theta\cos\theta + \theta^2lsen\theta + lsen\theta mlg - ml * l(\theta\sin\theta - \theta^2\cos\theta) = 0 \quad (7)$$

Linealizando alrededor del punto de operación $\theta = 0$

$$\sin^2\theta \cong 0 \quad \sin\theta \cong \theta \quad \cos\theta \cong 1 \quad \theta^2 \cong 0 \quad \theta \cong 0$$

Entonces por lo tanto las ecuaciones (6) y (7) será de la siguiente manera

$$U t = Mc + ml x - ml * l\theta + fcx \quad (8)$$

$$J\theta + fp\theta - ml * lx + ml * l^2\theta + l.ml * g.\theta = 0 \quad (9)$$

A partir de las siguientes fórmulas se procede a calcular las funciones de transferencia del carro y del ángulo.

Entonces resolviendo las ecuaciones (8) y (9) obtenemos lo siguiente:

$$U(s) = Mc + ml s^2 X(s) - ml.l.s^2.\theta(s) + fc.s X(s)$$

$$U(s) = X(s) [Mc + ml s^2 + fc.s - ml.l.s^2.\theta(s)] \quad (10)$$

$$J\theta + fp\theta - ml * lx + ml * l^2\theta + l.ml * g.\theta = 0$$

$$J.s^2.\theta(s) + fp.s.\theta(s) - ml * l.s^2.X(s) + ml * l^2.s^2.\theta(s) + l.ml * g.\theta(s) = 0$$

$$\theta(s) [J.s^2 + fp.s + ml * l^2.s^2 + l.ml * g - ml * l.s^2.X(s)] = 0$$

$$X(s) = \frac{\theta(s) [J.s^2 + fp.s + ml * l^2.s^2 + l.ml * g]}{ml * l.s^2} \quad (11)$$

Entonces reemplazamos (11) en (10), podemos obtener la función de transferencia correspondiente al ángulo

$$U(s) = \frac{Mc + ml s^2 + fc.s [J.s^2 + fp.s + ml.l^2.s^2 + l.ml.g \theta(s)]}{ml.l.s^2} - ml.l.s^2.\theta(s)$$

$$U(s) = \frac{Mc + ml s^2 + fc.s [J.s^2 + fp.s + ml.l^2.s^2 + l.ml.g \theta(s)] - ml^2.l^2.s^2.\theta(s)}{ml.l.s^2}$$

Como resultado del reemplazo de las fórmulas anteriores obtenemos la función de transferencia correspondiente al ángulo que es la siguiente.

$$\frac{\theta(s)}{U(s)} = \frac{ml.l.s^2}{(J + ml.l^2) s^4 + \frac{fp}{-ml^2.l^2} [Mc + ml] s^3 + \frac{fc}{+ml.l^2.fc} [Mc + ml] s^2 + \frac{l.ml.g}{+fp.fc} [Mc + ml] s}$$

Ordenando la función de transferencia del ángulo

$$\frac{\theta s}{U(s)} = \frac{ml.l.s^2}{\frac{Mc + ml}{l^2} J + \frac{Mc + ml}{l^2} fp s^4 + \frac{Mc + ml}{l^2} (J + ml * l^2) s^3 + \frac{Mc + ml}{l^2} l.ml.g s^2 + \frac{Mc + ml}{l^2} fc.ml.l.g s}$$

Como paso final solo nos falta obtener la función de transferencia para el carro

Entonces se despeja $\theta(s)$ de la ecuación (11) y se reemplaza en la ecuación (10) y obtenemos lo siguiente:

$$U s = X s \frac{Mc + ml s^2 + fc.s}{J.s^2 + fp.s + ml.l^2.s^2 + ml.l.g} - \frac{X s (ml^2.l^2.s^4)}{J.s^2 + fp.s + ml.l^2.s^2 + ml.l.g}$$

$$\frac{U s}{X s} = \frac{[Mc + ml s^2 + fc.s J.s^2 + fp.s + ml.l^2.s^2 + ml.l.g - ml^2.l^2.s^4]}{J.s^2 + fp.s + ml.l^2.s^2 + ml.l.g}$$

Como resultado obtenemos la función de transferencia del carro que es la siguiente:

$$\frac{X s}{U s} = \frac{(J + ml.l^2)s^2 + fp.s + ml.l.g}{\frac{Mc + ml}{(J + ml.l^2) s^4 + \frac{fp}{+J.fc} s^3 + \frac{l.ml.g}{+fp.fc} s^2 + [l.ml.g.fc]s}}$$

Ordenando la función de transferencia del carro o de la posición

$$\frac{X s}{U(s)} = \frac{(J + ml.l^2)s^2 + fp.s + ml.l.g}{\frac{Mc + ml}{l^2} J + \frac{Mc + ml}{l^2} fp s^4 + \frac{Mc + ml}{l^2} (J + ml * l^2) s^3 + \frac{Mc + ml}{l^2} l.ml.g s^2 + \frac{Mc + ml}{l^2} fc.ml.l.g s}$$

4.2. Propuesta de Control PID Difuso

Teniendo en cuenta la manera de que se manejan las entradas, para el controlador difuso y aún más siendo un controlador difuso PID, tenemos esta propuesta para el control de una Grúa Puento. Como se puede apreciar en la Figura 4.4, se tiene un control tanto como para el ángulo y para el carro, en los dos casos cada uno tiene como entrada el error y la derivada del error; en cuanto a la integral se tiene después del controlador que tiene la función de suavizar la salida de los dos controladores, que luego se les promedia y es repartida para cada función de transferencia como la del carro y del ángulo.

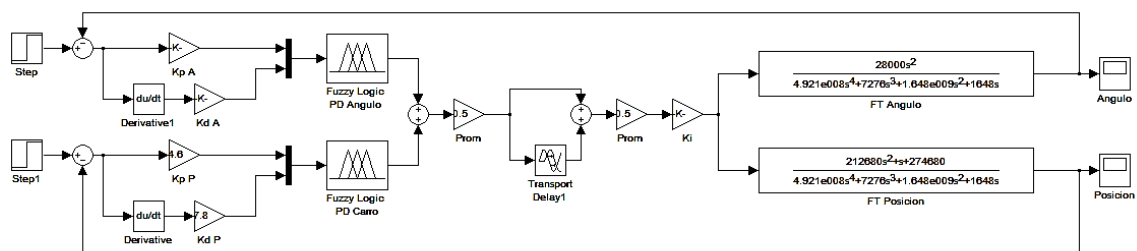


Figura 4.4. Propuesta Control PID Difuso

4.2.1. Entradas Difusas

Las funciones de membresía para el controlador PID difuso de la posición del carro y del ángulo, la única diferencia es el rango de cada una. En este sentido las mismas se describen a continuación.

4.2.1.1. Entrada Error del Angulo

Como hemos podido apreciar en los anteriores postulados, podemos ya definir cuáles son las funciones de membresía para las entradas, de todas las mencionadas elegimos las funciones triangulares.

Se encuentran dentro del rango $[-4.5 \ 4.5]$ y contamos con 7 funciones de membrecía, las que son a continuación

- Big Negative (BN), siendo de tamaño $[-6 \ -4.5 \ -3.002]$
- Medium Negative (MN), siendo de $[-4.5 \ -3.002 \ -1.5]$
- Small Negative (SN), siendo de tamaño $[-3.002 \ -1.5 \ 0]$
- Zero (ZR), siendo de tamaño $[-1.5 \ 0 \ 1.5]$
- Small Positive (SP), siendo de tamaño $[0 \ 1.5 \ 3.002]$
- Medium Positive (MP), siendo de tamaño $[1.5 \ 3.002 \ 4.5]$
- Big Positive (BP), siendo de tamaño $[3.002 \ 4.5 \ 6.003]$

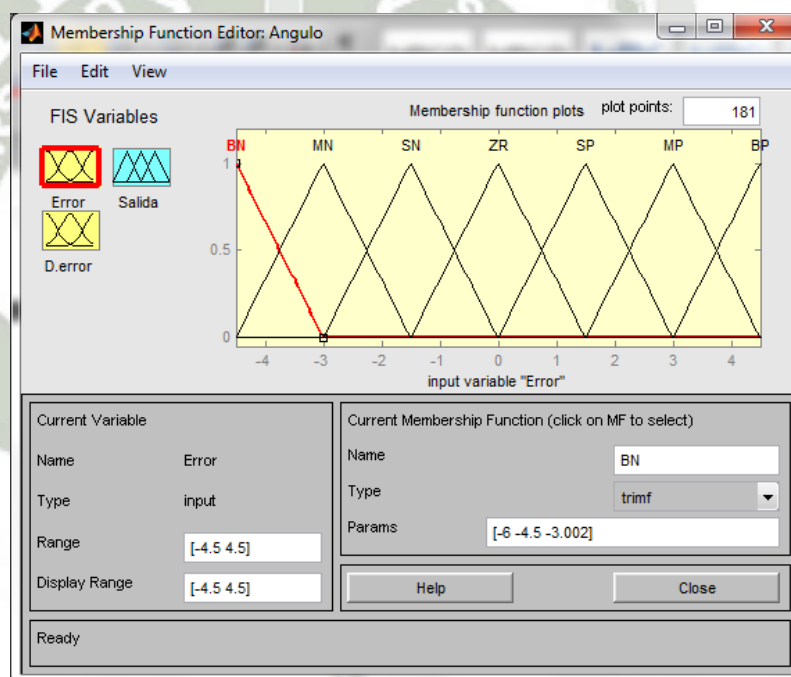


Figura 4.5. Funciones de membrecía de la entrada error del ángulo

4.2.1.2. Entrada Derivada del Error del Angulo

Para la derivada del error, se encuentra dentro del rango $[-0.9 \ 0.9]$ y también cuenta con 7 funciones de membrecía las cuales vemos a continuación:

- Big Negative (BN), siendo de tamaño [-1.2 -0.9 -0.6001]
- Medium Negative (MN), siendo de tamaño [-0.9 -0.6001 -0.3]
- Small Negative (SN), siendo de tamaño [-0.6001 -0.3 0]
- Zero (ZR), siendo de tamaño [-0.3 0 0.3]
- Small Positive (SP), siendo de tamaño [0 0.3 0.6001]
- Medium Positive (MP), siendo de tamaño [0.3 0.6001 0.9]
- Big Positive (BP), siendo de tamaño [0.6001 0.9 1.201]

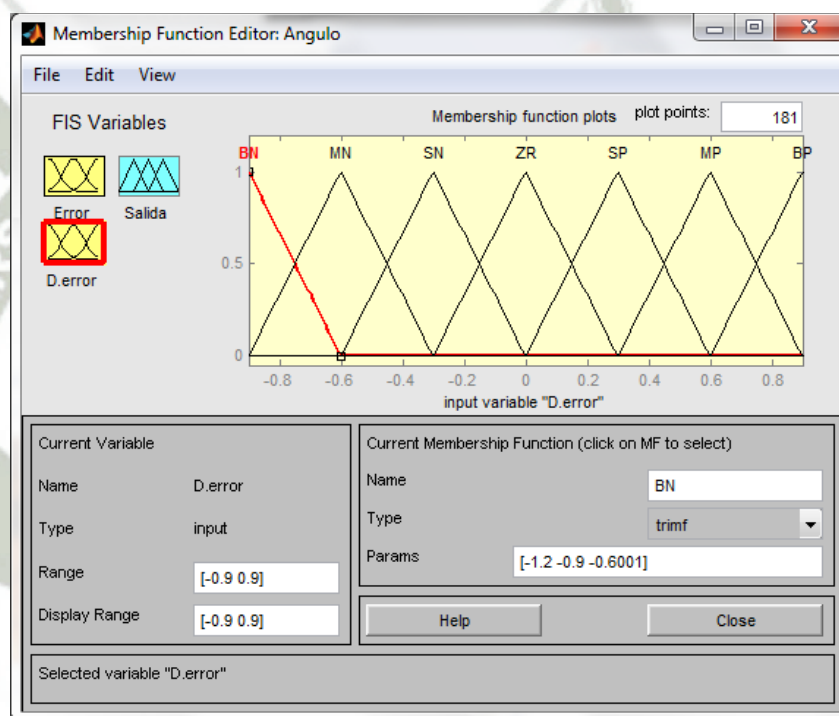


Figura 4.6. Funciones de membrecía de la entrada derivada del error del ángulo

4.2.1.3. Entrada Error del Carro

Como hemos podido apreciar en los anteriores postulados, podemos ya definir cuáles son las funciones de membrecía para las entradas, de todas las mencionadas elegimos las funciones triangulares.

Se encuentran dentro del rango $[-18.75 \ 18.75]$ y contamos con 7 funciones de membrecía, las que son a continuación

- Big Negative (BN), siendo de tamaño $[-25 \ -18.75 \ -12.5]$
- Medium Negative (MN), siendo de tamaño $[-18.75 \ -12.5 \ -6.251]$
- Small Negative (SN), siendo de tamaño $[-12.5 \ -6.251 \ 0]$
- Zero (ZR), siendo de tamaño $[-6.251 \ 0 \ 6.251]$
- Small Positive (SP), siendo de tamaño $[-6.251 \ 0 \ 6.251]$
- Medium Positive (MP), siendo de tamaño $[6.251 \ 12.5 \ 18.75]$
- Big Positive (BP), siendo de tamaño $[12.5 \ 18.75 \ 25.02]$

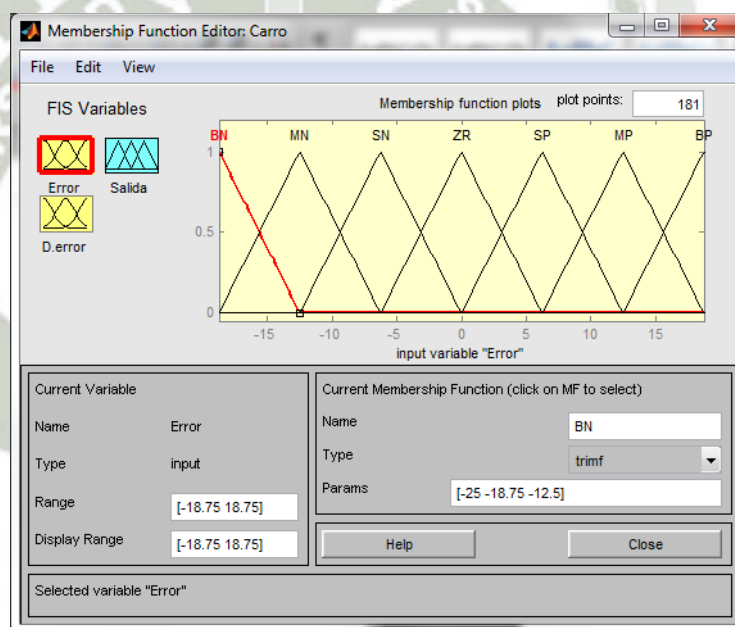


Figura 4.7. Funciones de membrecía de la entrada error del carro

4.2.1.4. Entrada Derivada del Error del Carro

Para la derivada del error, se encuentra dentro del rango $[-9.375 \ 9.375]$ y también cuenta con 7 funciones de membrecía las cuales vemos a continuación:

- Big Negative (BN), siendo de tamaño $[-12.5 \ -9.375 \ -6.248]$
- Medium Negative (MN), siendo de tamaño $[-9.375 \ -6.248 \ -3.127]$

- Small Negative (SN), siendo de tamaño [-6.248 -3.127 0]
- Zero (ZR), siendo de tamaño [-3.127 0 3.127]
- Small Positive (SP), siendo de tamaño [0 3.127 6.248]
- Medium Positive (MP), siendo de tamaño [3.127 6.248 9.375]
- Big Positive (BP), siendo de tamaño [6.248 9.375 12.52]

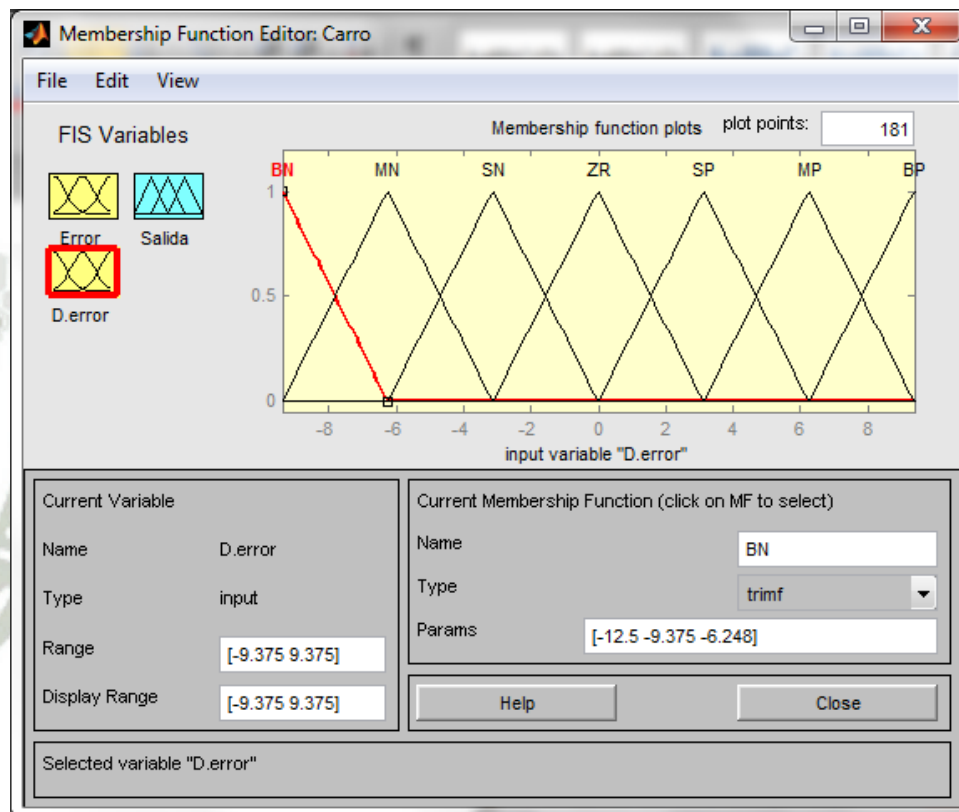


Figura 4.8. Funciones de membresía de la entrada derivada del error del carro.

4.2.2. Salidas Difusas

Las salidas para nuestro controlador PID Difuso, como son la del carro y del ángulo, en estos dos casos la salida se encuentra retrasada y sumada la cual es integrada para poder suavizar la señal de salida la cual entra al bloque de planta.

4.2.2.1. Salida del Angulo

La salida se encuentra dentro del rango $[-93750 \ 93750]$ y también cuenta con 7 funciones de membrecía las cuales vemos a continuación:

- Big Negative (BN), siendo de tamaño
 $[-1.25e+005 \ -9.375e+004 \ -6.249e+004]$
- Medium Negative (MN), siendo de tamaño
 $[-9.375e+004 \ -6.249e+004 \ -3.126e+004]$
- Small Negative (SN), siendo de tamaño
 $[-6.249e+004 \ -3.126e+004 \ 0]$
- Zero (ZR), siendo de tamaño
 $[-3.077e+004 \ 496.1 \ 3.176e+004]$
- Small Positive (SP), siendo de tamaño
 $[0 \ 3.126e+004 \ 6.249e+004]$
- Medium Positive (MP), siendo de tamaño
 $[3.126e+004 \ 6.249e+004 \ 9.375e+004]$
- Big Positive (BP), siendo de tamaño
 $[6.249e+004 \ 9.375e+004 \ 1.251e+005]$

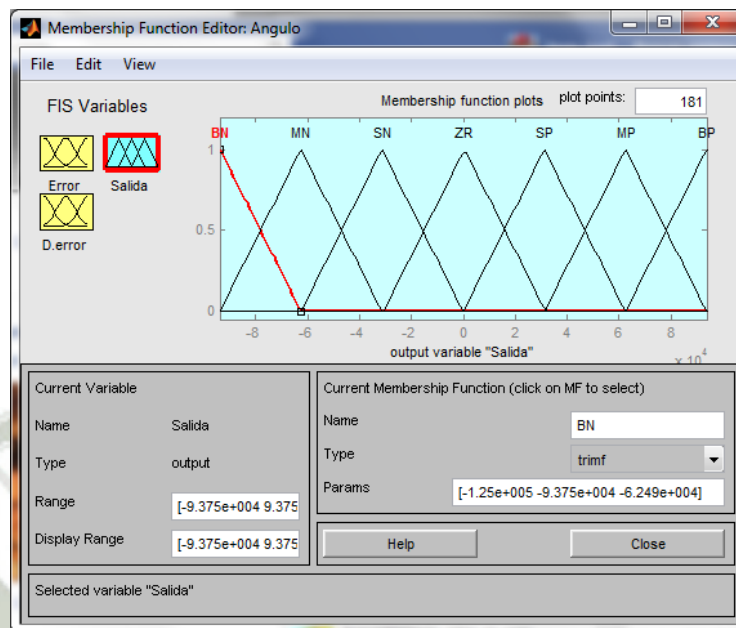


Figura 4.9. Función de membresía de salida del ángulo.

4.2.2.1. Salida del Carro

La salida se encuentra dentro del rango $[-93750 \ 93750]$ y también cuenta con 7 funciones de membresía las cuales vemos a continuación:

- Big Negative (BN), siendo de tamaño $[-1.25e+005 \ -9.375e+004 \ -6.249e+004]$
- Medium Negative (MN), siendo de tamaño $[-9.375e+004 \ -6.249e+004 \ -3.126e+004]$
- Small Negative (SN), siendo de tamaño $[-6.249e+004 \ -3.126e+004 \ 0]$
- Zero (ZR), siendo de tamaño $[-3.077e+004 \ 496.1 \ 3.176e+004]$
- Small Positive (SP), siendo de tamaño $[0 \ 3.126e+004 \ 6.249e+004]$
- Medium Positive (MP), siendo de tamaño $[3.126e+004 \ 6.249e+004 \ 9.375e+004]$
- Big Positive (BP), siendo de tamaño $[6.249e+004 \ 9.375e+004 \ 1.251e+005]$

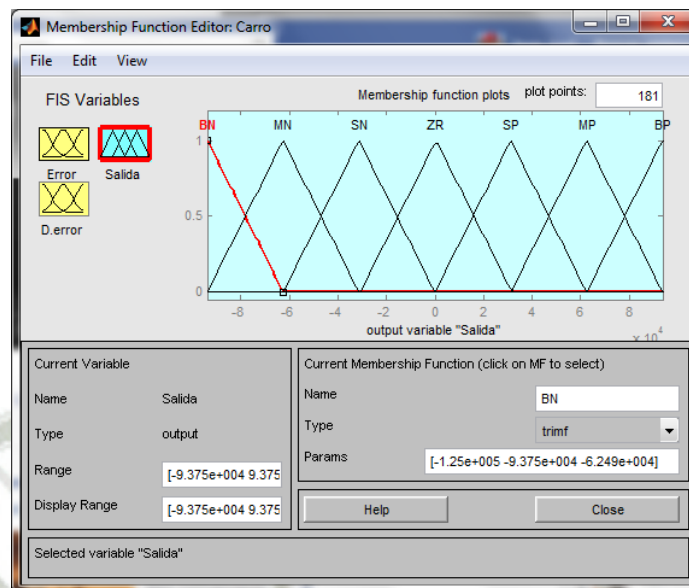


Figura 4.10. Función de membresía de salida del carro

4.2.3. Reglas Difusas

Teniendo en cuenta el respectivo análisis de las reglas visto en el capítulo anterior, como siendo dos entradas, cada una cuanta con 7 funciones de membresía, siendo muy parecidas a nuestra situación y lo cual tomamos como ejemplo la manera, la lógica en que esos autores realizan sus propias reglas para dar solución a sus respectivos problemas, lo cual nos brinda mayor seguridad en realizar nuestras propias reglas.

Entonces tenemos que el error es el valor deseado menos el valor obtenido, por lo tanto si el error es positivo, nos falta llegar y si es error es negativo significa que no pasamos. Como se puede ver en las siguientes tablas se entiende mejor la acción que se debe de tomar para lograr nuestro objetivo. (Tabla 4.1 y Tabla 4.2)

Tabla N°4.1.

Error	D. error	
	-1	1
1	-1	1
-1	-1	1

Dónde:

1 para avanzar

-1 para retroceder

Tabla N°4.2.

Error	D. error	
	-1	1
1	PE	FE
-1	PM	FM

Dónde:

PM me pase del objetivo pero estoy mejorando

FE aun no llego al objetivo y estoy empeorando

PE me pasé del objetivo y estoy empeorando

FM aún no llego al objetivo y estoy mejorando

Así mismo la derivada del error es el error actual menos el error en el tiempo anterior. De igual manera para poder entender mejor el concepto de la derivada y como se determina la accion que se debe realizar se muestras en la tabla 4.3.

Tabla 4.3. Derivada del Error

d.e (t)	e(t)	e(t-1)
+	+ grande	+ pequeño
-	+ pequeño	+ grande
+	+	-
-	-	+
-	- grande	- pequeño
+	- pequeño	- grande

Ahora bien después de haber visto las anteriores tablas, la lógica que hay que seguir y la acción que debemos tomar, es entonces que procedemos a definir nuestras reglas, en la tabla 4.4.

Tabla N°4.4. Reglas Difusas

error	D. error						
	-3	-2	-1	0	1	2	3
3	-1	-1	-1	1	2	3	3
2	-1	-1	-1	1	2	3	3
1	-2	-1	-1	1	1	3	3
0	-3	-2	-1	0	1	2	3
-1	-3	-3	-1	-1	1	1	2
-2	-3	-3	-2	-1	1	1	1
-3	-3	-3	-2	-1	1	1	1

De igual manera que se hizo el análisis de comportamiento de las reglas difusas lo mismo realizamos con nuestras reglas y del cual obtenemos el siguiente grafico en la Figura 3.11.

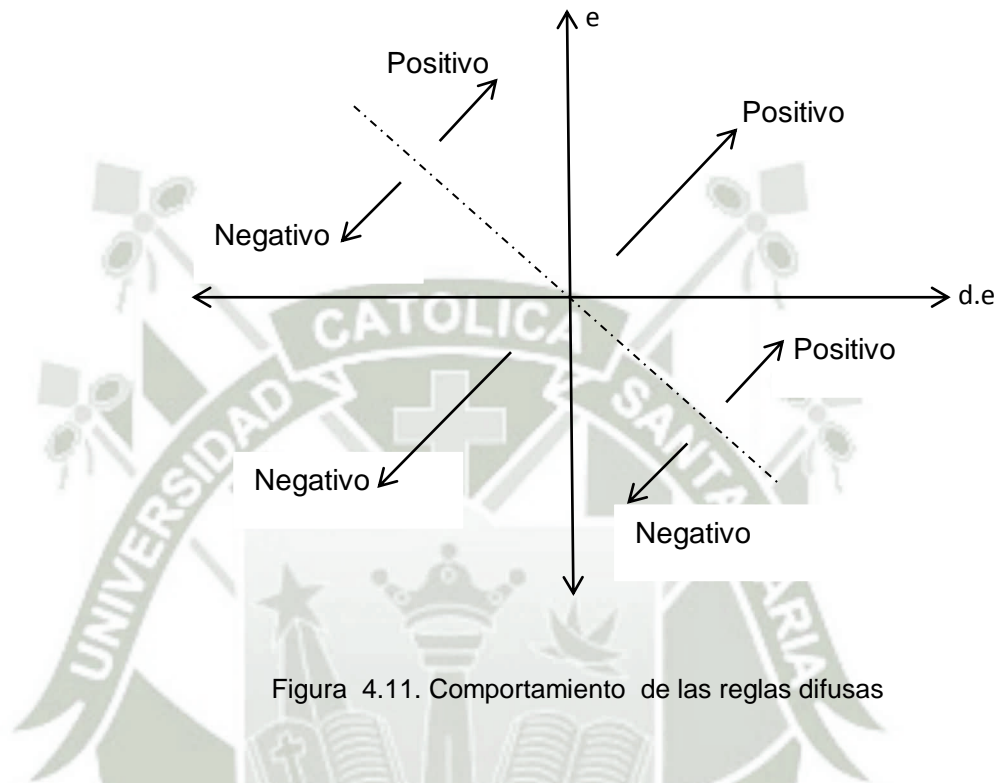


Figura 4.11. Comportamiento de las reglas difusas

Pero no solo es la única manera en que se presentan las reglas, la forma lingüística a continuación:

1. If (Error is BN) and (D.error is BN) then (Salida is BN) (1)
2. If (Error is BN) and (D.error is MN) then (Salida is BN) (1)
3. If (Error is BN) and (D.error is SN) then (Salida is BN) (1)
4. If (Error is BN) and (D.error is ZR) then (Salida is BN) (1)
5. If (Error is BN) and (D.error is SP) then (Salida is MN) (1)
6. If (Error is BN) and (D.error is MP) then (Salida is SN) (1)
7. If (Error is BN) and (D.error is BP) then (Salida is SN) (1)
8. If (Error is MN) and (D.error is BN) then (Salida is BN) (1)
9. If (Error is MN) and (D.error is MN) then (Salida is BN) (1)
10. If (Error is MN) and (D.error is SN) then (Salida is BN) (1)

11. If (Error is MN) and (D.error is ZR) then (Salida is MN) (1)
12. If (Error is MN) and (D.error is SP) then (Salida is SN) (1)
13. If (Error is MN) and (D.error is MP) then (Salida is SN) (1)
14. If (Error is MN) and (D.error is BP) then (Salida is SN) (1)
15. If (Error is SN) and (D.error is BN) then (Salida is MN) (1)
16. If (Error is SN) and (D.error is MN) then (Salida is MN) (1)
17. If (Error is SN) and (D.error is SN) then (Salida is SN) (1)
18. If (Error is SN) and (D.error is ZR) then (Salida is SN) (1)
19. If (Error is SN) and (D.error is SP) then (Salida is SN) (1)
20. If (Error is SN) and (D.error is MP) then (Salida is SN) (1)
21. If (Error is SN) and (D.error is BP) then (Salida is SN) (1)
22. If (Error is ZR) and (D.error is BN) then (Salida is SN) (1)
23. If (Error is ZR) and (D.error is MN) then (Salida is SN) (1)
24. If (Error is ZR) and (D.error is SN) then (Salida is SN) (1)
25. If (Error is ZR) and (D.error is ZR) then (Salida is ZR) (1)
26. If (Error is ZR) and (D.error is SP) then (Salida is SP) (1)
27. If (Error is ZR) and (D.error is MP) then (Salida is SP) (1)
28. If (Error is ZR) and (D.error is BP) then (Salida is SP) (1)
29. If (Error is SP) and (D.error is BN) then (Salida is SP) (1)
30. If (Error is SP) and (D.error is MN) then (Salida is SP) (1)
31. If (Error is SP) and (D.error is SN) then (Salida is SP) (1)
32. If (Error is SP) and (D.error is ZR) then (Salida is SP) (1)
33. If (Error is SP) and (D.error is SP) then (Salida is SP) (1)
34. If (Error is SP) and (D.error is MP) then (Salida is MP) (1)
35. If (Error is SP) and (D.error is BP) then (Salida is MP) (1)
36. If (Error is MP) and (D.error is BN) then (Salida is SP) (1)
37. If (Error is MP) and (D.error is MN) then (Salida is SP) (1)
38. If (Error is MP) and (D.error is SN) then (Salida is SP) (1)
39. If (Error is MP) and (D.error is ZR) then (Salida is MP) (1)
40. If (Error is MP) and (D.error is SP) then (Salida is BP) (1)
40. If (Error is MP) and (D.error is SP) then (Salida is BP) (1)
41. If (Error is MP) and (D.error is MP) then (Salida is BP) (1)
42. If (Error is MP) and (D.error is BP) then (Salida is BP) (1)
43. If (Error is BP) and (D.error is BN) then (Salida is SP) (1)
44. If (Error is BP) and (D.error is MN) then (Salida is SP) (1)
45. If (Error is BP) and (D.error is SN) then (Salida is MP) (1)
46. If (Error is BP) and (D.error is ZR) then (Salida is BP) (1)
47. If (Error is BP) and (D.error is SP) then (Salida is BP) (1)
48. If (Error is BP) and (D.error is MP) then (Salida is BP) (1)
49. If (Error is BP) and (D.error is BP) then (Salida is BP) (1)

Figura 4.12. Reglas difusas (forma lingüística)

CAPÍTULO V

SIMULADOR GRÚA PUENTE

Simulación es una técnica numérica para realizar un modelo de un sistema real con la finalidad de realizar experimentos en una computadora digital. Estos experimentos involucran ciertos tipos de modelos matemáticos y lógicos que describen el comportamiento de sistemas o para evaluar nuevas estrategias, dentro de los límites impuestos por un cierto criterio o un conjunto de ellos para el funcionamiento del sistema en períodos de tiempo.

Para poder desarrollar una simulación es importante elegir un método de integración numérica el cual permita calcular iterativamente los valores de las variables de estado de un sistema. En este sentido en esta tesis se adopta el método de integración de Euler.

5.1. Método de Euler

Este método se aplica para encontrar la solución a ecuaciones diferenciales ordinaria, esto es, cuando la función involucra solo una variable independiente

$$\frac{dy}{dx} = f(x, y)$$

El método se basa de forma general en la pendiente estimada de la función para extrapolar desde un valor anterior a un nuevo valor:

Nuevo valor = valor anterior + pendiente x tamaño del paso. O bien

$$y_{i+1} = y_1 + \phi h \quad (1)$$

De esta manera, la fórmula (1), se aplica paso a paso para encontrar un valor en el futuro y así trazar la trayectoria de la solución. La Figura 5.1., muestra el procedimiento aplicado con la fórmula (1)

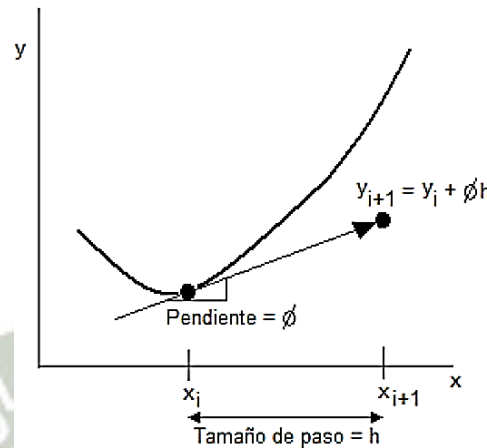


Figura 5.1. Predicción de un nuevo valor en la solución

El método de Euler utiliza la pendiente al inicio del intervalo como una aproximación de la pendiente promedio sobre todo el intervalo. La primera derivada proporciona una estimación directa de la pendiente en x_i .

$$\phi = f(x, y)$$

$f(x_i, y_i)$, es la ecuación diferencial evaluada en x_i y y_i . Sustituyendo esta estimación de la pendiente en la ecuación (1) se tiene:

$$y_{i+1} = y_i + f(x_i, y_i)h \quad (2)$$

La ecuación (2), se le conoce como el método de Euler. En esta fórmula se predice un nuevo valor de y por medio de la pendiente que es igual a la primera derivada en el valor original de x , este nuevo valor habrá que extrapolarse en forma sobre el tamaño de paso h .

5.2. Diseño del simulador

Para realizar la simulación tenemos en cuenta las siguientes formulas (6) y (7) del capítulo 4 las cuales nos permiten obtener las componentes x y θ ya que esto nos proporciona el cambio de la velocidad lineal y el de la velocidad angular. Y se reemplaza en el siguiente sistema de ecuaciones de espacio de estados

$$\dot{\theta} = \omega + \theta dt$$

$$\dot{\theta} = \omega + \theta dt$$

$$\dot{x} = v + x dt$$

$$\dot{x} = v + x dt$$

Para obtener x tenemos:

$$J\dot{\theta} + fp\theta - ml * l\cos\theta \ddot{x} - l\theta\cos\theta + \theta^2 l\sin\theta + l\sin\theta mlg - ml * l(\theta\sin\theta - \theta^2\cos\theta) = 0$$

$$J\dot{\theta} + fp\theta + ml * l^2\cos\theta\ddot{\theta} - ml * l\cos\theta \ddot{x} + \theta^2 l\sin\theta - ml * l^2\sin^2\theta + l\sin\theta mlg + ml * l\theta^2\cos\theta = 0$$

$$J\dot{\theta} + ml * l^2\cos\theta\ddot{\theta} - ml * l^2\sin^2\theta + fp\theta - ml * l\cos\theta \ddot{x} + \theta^2 l\sin\theta + l\sin\theta(mlg + ml * l\theta^2\cos\theta) = 0$$

$$\theta(J + ml * l^2\cos\theta - ml * l^2\sin^2\theta) + fp\theta - ml * l\cos\theta \ddot{x} + \theta^2 l\sin\theta + l\sin\theta(mlg + ml * l\theta^2\cos\theta) = 0$$

$$\theta = \frac{-fp\theta + ml * l\cos\theta \ddot{x} + \theta^2 l\sin\theta - l\sin\theta(mlg + ml * l\theta^2\cos\theta)}{J + ml * l^2(\cos\theta - \sin^2\theta)}$$

Remplazando en U(t) para obtener x :

$$U t = Mc + ml \ddot{x} - ml * l\theta\cos\theta + ml * l\theta^2\sin\theta + fc * x$$

$$U t = Mc + ml x - ml * l \cos \theta - \frac{-fp\theta + ml * l \cos \theta x + \theta^2 l \sin \theta}{J + ml * l^2 (\cos \theta - \sin^2 \theta)} + ml * l \theta^2 \sin \theta + fc * x$$

$$U t = \frac{Mc + ml x J + ml * l^2 \cos \theta - \sin^2 \theta - ml * l \cos \theta - fp\theta + ml * l \cos \theta x + \theta^2 l \sin \theta - l \sin \theta mlg + ml * l \theta^2 \cos \theta + (ml * l \theta^2 \sin \theta + fc * x)(J + ml * l^2 (\cos \theta - \sin^2 \theta))}{J + ml * l^2 (\cos \theta - \sin^2 \theta)}$$

$$U t = \frac{Mc + ml x J + ml * l^2 \cos \theta - \sin^2 \theta - ml^2 * l^2 \cos^2 \theta - ml * l \cos \theta - fp\theta + ml * l^2 \cos \theta \sin \theta \theta^2 - l \sin \theta mlg + ml * l \theta^2 \cos \theta + (ml * l \theta^2 \sin \theta + fc * x)(J + ml * l^2 (\cos \theta - \sin^2 \theta))}{J + ml * l^2 (\cos \theta - \sin^2 \theta)}$$

$$U t = \frac{Mc + ml x [J + ml * l^2 \cos \theta - \sin^2 \theta - ml^2 * l^2 \cos^2 \theta] - ml * l \cos \theta - fp\theta + ml * l^2 \cos \theta \sin \theta \theta^2 - l \sin \theta mlg + ml * l \theta^2 \cos \theta + (ml * l \theta^2 \sin \theta + fc * x)(J + ml * l^2 (\cos \theta - \sin^2 \theta))}{J + ml * l^2 (\cos \theta - \sin^2 \theta)}$$

$$x = \frac{U t J + ml * l^2 \cos \theta - \sin^2 \theta + ml * l \cos \theta - fp\theta + ml * l^2 \cos \theta \sin \theta \theta^2 - l \sin \theta mlg + ml * l \theta^2 \cos \theta - (ml * l \theta^2 \sin \theta + fc * x)(J + ml * l^2 (\cos \theta - \sin^2 \theta))}{Mc + ml [J + ml * l^2 \cos \theta - \sin^2 \theta - ml^2 * l^2 \cos^2 \theta]}$$

Pero teniendo en cuenta las siguientes condiciones:

$$\sin^2 \theta \cong 0 \quad \sin \theta \cong \theta \quad \cos \theta \cong 1 \quad \theta^2 \cong 0 \quad \theta \cong 0$$

$$x = \frac{U t J + ml * l^2 + ml * l - fp\theta - ml * l g \theta - fc * x (J + ml * l^2)}{Mc + ml J + ml * l^2 - ml^2 * l^2}$$

$$x = \frac{J + ml * l^2 (U t - fc * x) + ml * l - fp\theta - ml * l g \theta}{Mc J + ml * l^2 + ml * J}$$

Para obtener θ tenemos la siguiente ecuación:

$$J\theta + fp\theta - ml * l \cos \theta x - l\theta \cos \theta + \theta^2 l \sin \theta + l \sin \theta mlg - ml * l (\theta \sin \theta - \theta^2 \cos \theta) = 0$$

$$J\theta + fp\theta - ml * l \cos \theta x - ml * l \cos \theta - l\theta \cos \theta + \theta^2 l \sin \theta + l \sin \theta mlg - ml * l (\theta \sin \theta - \theta^2 \cos \theta) = 0$$

$$x = \frac{J\theta + fp\theta - ml * l \cos \theta - l\theta \cos \theta + \theta^2 l \sin \theta + l \sin \theta mlg - ml * l (\theta \sin \theta - \theta^2 \cos \theta)}{ml * l \cos \theta}$$

Remplazando en U(t) para obtener θ :

$$U t = M c + m l x - m l * l \theta \cos \theta + m l * l \theta^2 \operatorname{sen} \theta + f c * x$$

$$U t = M c + m l \frac{J \theta + f p \theta - m l * l \cos \theta - l \theta \cos \theta + \theta^2 l \operatorname{sen} \theta + l \operatorname{sen} \theta m l g - m l * l (\theta \operatorname{sen} \theta - \theta^2 \cos \theta)}{m l * l \cos \theta} - m l * l \theta \cos \theta + m l * l \theta^2 \operatorname{sen} \theta + f c * x$$

$$U t = \frac{M c + m l J \theta + f p \theta - m l * l \cos \theta - l \theta \cos \theta + \theta^2 l \operatorname{sen} \theta + l \operatorname{sen} \theta m l g - m l * l (\theta \operatorname{sen} \theta - \theta^2 \cos \theta) - (m l * l \theta \cos \theta + m l * l \theta^2 \operatorname{sen} \theta + f c * x)(m l * l \cos \theta)}{m l * l \cos \theta}$$

$$U t = \frac{M c + m l \theta (J + m l * l^2 \cos^2 \theta - m l * l^2 \operatorname{sen}^2 \theta) - m l^2 * l^2 \cos^2 \theta + (M c + m l)[f p \theta - m l * l \cos \theta \operatorname{sen} \theta \theta^2 + l \operatorname{sen} \theta (m l g + m l * l \cos \theta \theta^2) - (m l * l \theta^2 \operatorname{sen} \theta + f c * x)(m l * l \cos \theta)]}{m l * l \cos \theta}$$

$$\theta = \frac{U t m l * l \cos \theta - (M c + m l)[f p \theta - m l * l \cos \theta \operatorname{sen} \theta \theta^2 + l \operatorname{sen} \theta m l g + m l * l \cos \theta \theta^2 + (m l * l \theta^2 \operatorname{sen} \theta + f c * x)(m l * l \cos \theta)]}{M c + m l (J + m l * l^2 \cos^2 \theta - m l * l^2 \operatorname{sen}^2 \theta) - m l^2 * l^2 \cos^2 \theta}$$

Pero teniendo en cuenta las siguientes condiciones

$$\operatorname{sen}^2 \theta \cong 0 \quad \operatorname{sen} \theta \cong \theta \quad \cos \theta \cong 1 \quad \theta^2 \cong 0 \quad \theta \cong 0$$

$$\theta = \frac{U t m l * l - (M c + m l)[f p \theta + m l * l g \theta] + f c * x(m l * l)}{M c + m l (J + m l * l^2) - m l^2 * l^2}$$

$$\theta = \frac{m l * l (U t - f c * x) - (M c + m l)[f p \theta + m l * l g \theta]}{M c J + m l * l^2 + m l * J}$$

Una vez ya obtenidas las fórmulas que determinan el espacio de estados y determinado el método de integración con todo eso ya definido procedemos a programar el simulador. Como se puede apreciar en la Figura 5.2. Se tiene la interface

de dicho simulador, donde se tiene que en la imagen izquierda superior se tiene el movimiento de la Grúa Puente, en la imagen derecha superior se tiene la señal de control, en la imagen izquierda inferior se tiene el comportamiento del carro y en la imagen derecha inferior se tiene el comportamiento del ángulo.

Así también se puede apreciar en la columna de la izquierda ahí tenemos dos botones uno para ejecutar la simulación y el otro para reiniciar el proceso de simulación siendo el verde y el amarillo respectivamente, mas debajo de dichos botones tenemos una secciones que nos permite que tipo de control hemos de usar, así como sus respectivas ganancias y por ultimo tenemos el cuadro de más abajo, en el cual podemos ingresar los parámetros que definen a la Grúa Puente, los que son: la longitud de la cuerda o cadena, la masa del carro, la masa de la carga, el momento de inercia, la fricción del carro, la fricción rotacional, el tamaño de la viga y como parte de la simulación se tiene la posición inicial, posición final y el tiempo de simulación

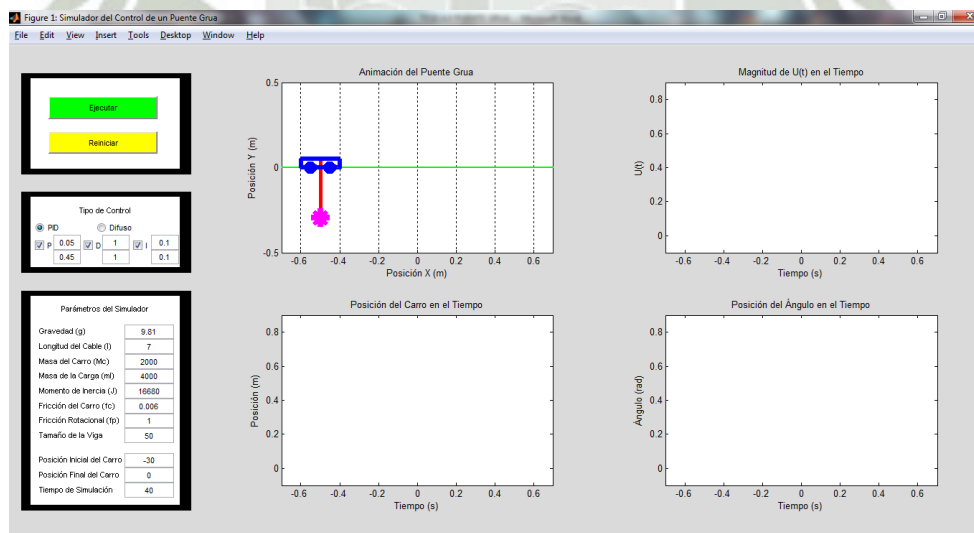


Figura 5.2. Interface Simulador

Para poder realizar dicha simulación, esta depende de 4 archivos realizados en Matlab:

1. ControlPGruaFinal, en este archivo se encuentran, el control mismo de la Grúa Puente, además se encuentra la manera de cómo se calcula las funciones de transferencia tanto para el control convencional y el difuso, además de las reglas difusas y por último el propio gráfico de la Grúa Puente. **(anexo 1)**.
2. PuenteGruaFinal, en él se encuentra todo el diseño de la interface, como los botones y sus colores, las tablas para elegir el tipo de control, la tabla de ingreso de los parámetros y la ubicación de gráficas. **(anexo 2)**
3. FuncionPertenenciaTriangular, nos permite realizar todas las funciones de membresía, ya que en este archivo se encuentra el forma para realizar las funciones de membresía desde el inicio, medio y final. **(anexo 3)**
4. Selcbk, la que nos permite utilizar los botones que hace llamado a las diferentes funciones para realizar la simulación, como la de ejecutar y reiniciar la simulación. **(anexo 4)**

CAPÍTULO VI

ANÁLISIS DEL SISTEMA USANDO SIMULINK

En este capítulo se realiza el análisis de la Grúa Puentes la cual tiene los siguientes parámetros, lo cual se presentan en la siguiente tabla 6.1.

Tabla 6.1. Parámetros Grúa Puentes

Parámetro	Valor
Límites del riel	50 [m]
Gravedad (g)	9.81 [m/s ²]
Longitud del cable (l)	7 [m]
Masa del carro (Mc)	2000 [Kg]
Masa de la carga (ml)	4000 [Kg]
Momento de inercia (J)	16680 [Kg·m ²]
Coefficiente de fricción de rotación (fp)	1
Coefficiente de fricción del carro (fc)	0.006

Teniendo estos parámetros podemos las funciones de transferencia:

Función de transferencia de la posición.

$$\frac{X(s)}{U(s)} = \frac{212680s^2 + s + 274680}{4.921e008s^4 + 7276s^3 + 1.648e009s^2 + 1648s}$$

Función de transferencia del ángulo.

$$\frac{\theta(s)}{U(s)} = \frac{28000s^2}{4.921e008s^4 + 7276s^3 + 1.648e009s^2 + 1648s}$$

Entonces, teniendo los parámetros y las respectivas funciones de transferencia se procede a analizar cada sistema de control; es decir se analizarán los tres controles convencionales (PD, PI y PID) y el control difuso, este análisis se realiza teniendo en cuenta que a cada uno de los sistemas tienen como señal de entrada al escalón unitario, lo que nos permite apreciar cual es el comportamiento de la posición y del ángulo en cada uno de ellos.

En todos estos controles se aprecia que hay dos controladores, uno para el control de la posición y el otro para el control del ángulo, como cada uno depende del otro es que se promedia la acción de los dos controladores, esta ingresa a cada una de las funciones de transferencia y por medio del bloque "scope" del Simulink es que se obtiene dichas gráficas de comportamiento del sistema.

Entonces en primer lugar, tenemos al control de la Grúa Puente con un controlador PD tanto para el control de posición, como para el control del ángulo como es el que se ve en la Figura 6.1.

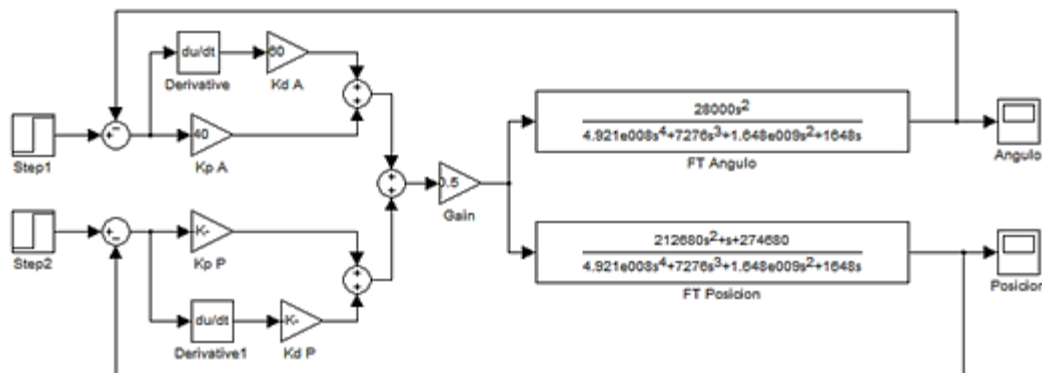


Figura 6.1. Diagrama de control PD convencional.

En segundo lugar tenemos el control de la Grúa Puentes con un controlador PI, tanto para la posición, como para el control del ángulo, el que se ve en la Figura 6.2.

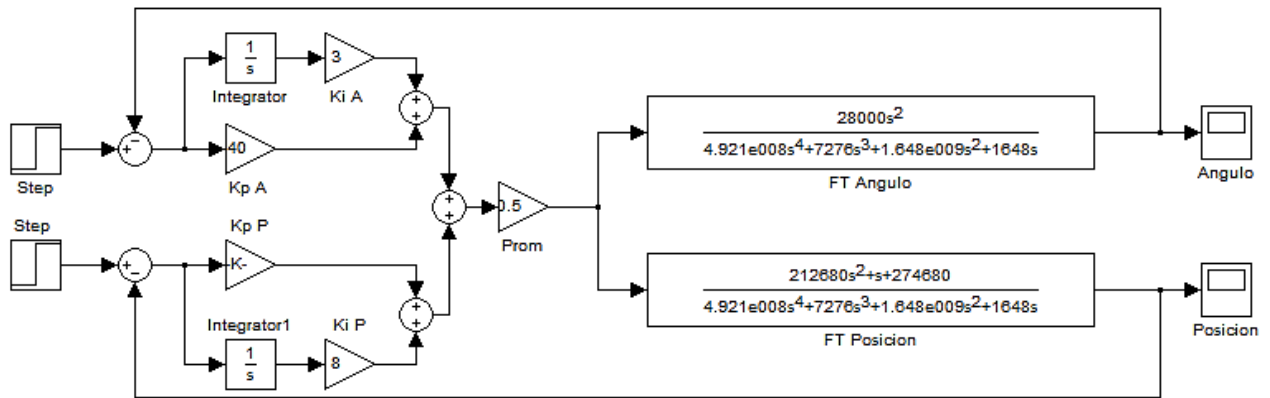


Figura 6.2. Diagrama de control PI convencional.

En tercer lugar tenemos el control de la Grúa Puentes con un controlador PID, tanto para la posición, como para el control del ángulo, el que se ve en la Figura 6.3.

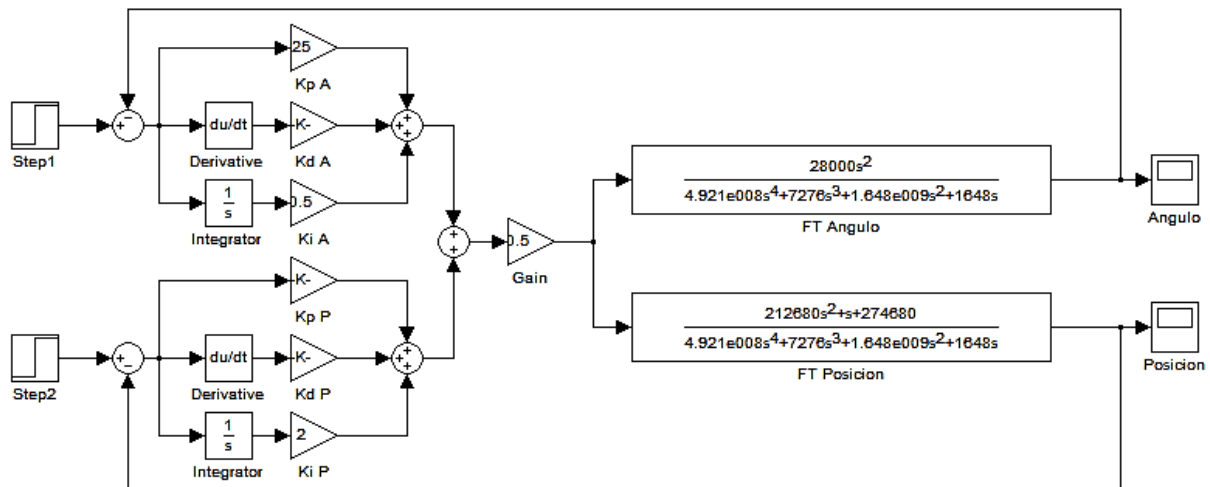


Figura 6.3. Diagrama de control PID convencional.

En último lugar tenemos el control de la Grúa Puentes con un control PID, tanto para la posición, como para el control del ángulo, el que se ve en la Figura 6.4

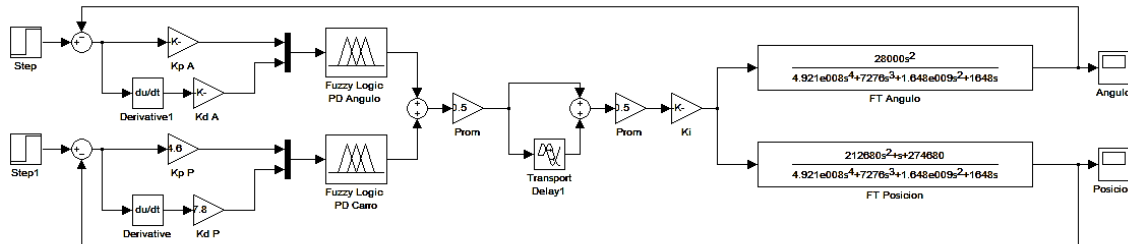


Figura 6.4. Diagrama de control PID Difuso

Ya definido todo en cuando a los controladores procedemos a presentar los resultados de cada controlador, lo que significa es obtener las gráficas de comportamiento de la posición y del comportamiento del ángulo.

6.1. Resultados

Para analizar las gráficas proporcionadas por el Simulink que siguen, se toma en cuenta tres, ítems para el control de la posición los cuales son, el sobre impulso, el tiempo de subida, el tiempo de establecimiento y el error en estado estable y para el control del ángulo son dos, la magnitud de las oscilaciones y el tiempo de establecimiento.

6.1.1. Resultados Control PD

Las siguientes gráficas se obtuvieron gracias a las ganancias aplicadas en dicho controlador, para poder la estabilidad desea.

Las ganancias obtenidas fueron para la posición son $K_p = 8500$ $K_d = 19500$ y para el ángulo son $K_p = 40$, $K_d = 60$

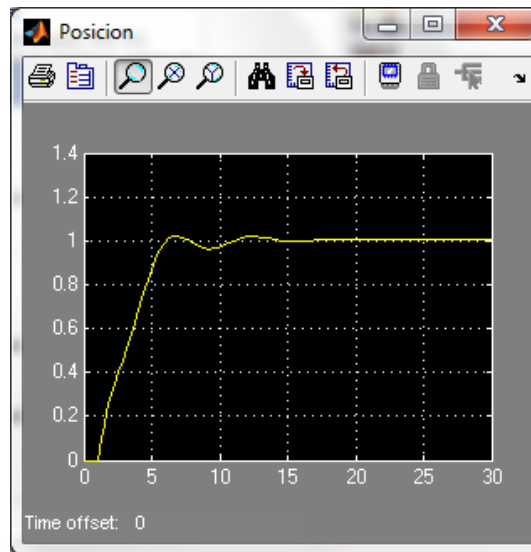


Figura 6.5. Comportamiento de la posición, control PD

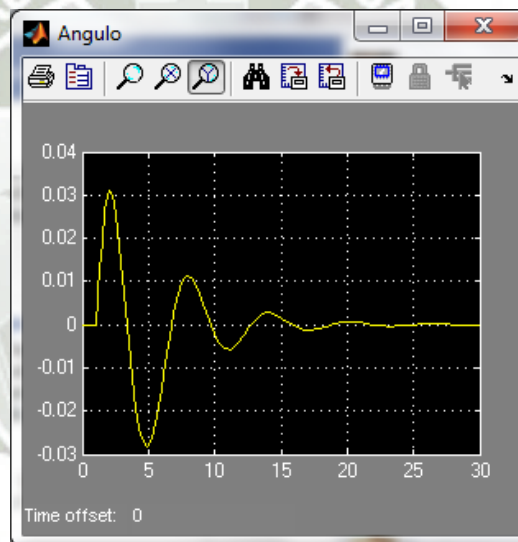


Figura 6.6. Comportamiento del ángulo, control PD

Resultados de la posición

- Sobreimpulso: 0 %
- Tiempo de subida: 5 segundos
- Tiempo de establecimiento: a los 15 segundos
- Error estado estable: 0

Resultados del ángulo

- Magnitud de la oscilación: 0.03 radianes
- Tiempo de establecimiento: a los 25 segundos

Todos estos resultados se pueden apreciar en la Figuras 6.5 y 6.6

6.1.2. Resultados Control PI

Las siguientes graficas se obtuvieron gracias a las ganancias aplicadas en dicho controlador, para poder la estabilidad desea.

Las ganancias obtenidas fueron para la posición son $K_p = 1400$, $K_i = 8$ y para el ángulo son $K_p = 40$, $K_i = 3$

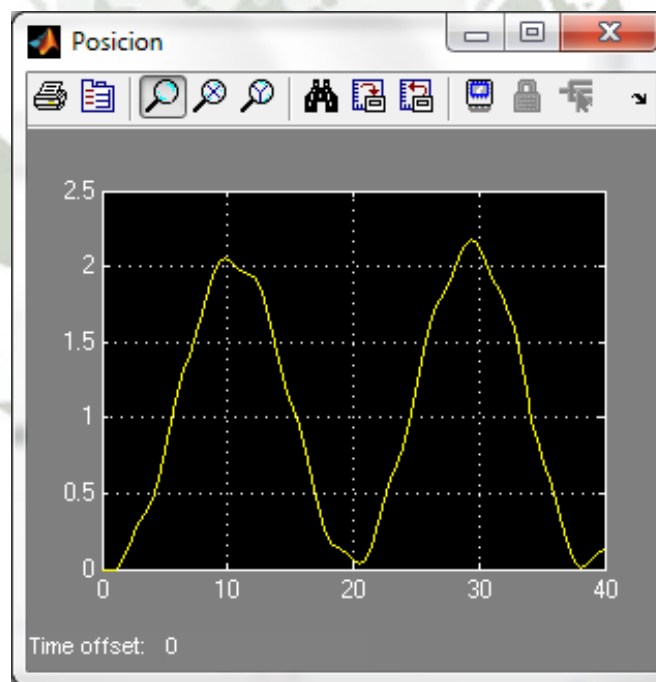


Figura 6.7. Comportamiento de la posición, control PI

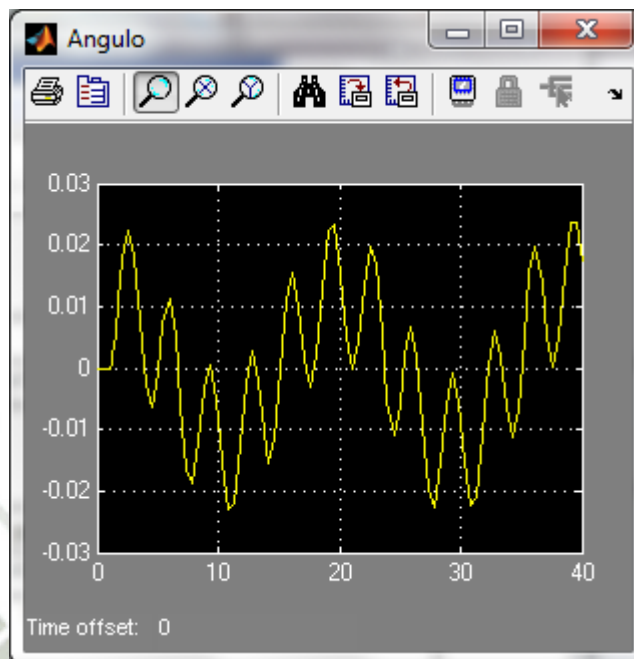


Figura 6.8. Comportamiento del ángulo, control PI

Resultados de la posición

- Sobreimpulso: 100%
- Tiempo de subida: 8 segundos
- Tiempo de establecimiento: no se estabiliza
- Error estado estable: muy alto

Resultados del ángulo

- Magnitud de la oscilación: 0.02 radianes
- Tiempo de establecimiento: no llega a estabilizarse

Todos estos resultados se pueden apreciar en la Figuras 6.7 y 6.8.

6.1.3. Resultados Control PID

Las siguientes graficas se obtuvieron gracias a las ganancias aplicadas en dicho controlador, para poder la estabilidad desea.

Las ganancias obtenidas fueron para la posición son $K_p = 5000$, $k_d = 1400$, $K_i = 2$
y para el ángulo son $K_p = 25$, $k_d = 6000$, $K_i = 0.5$

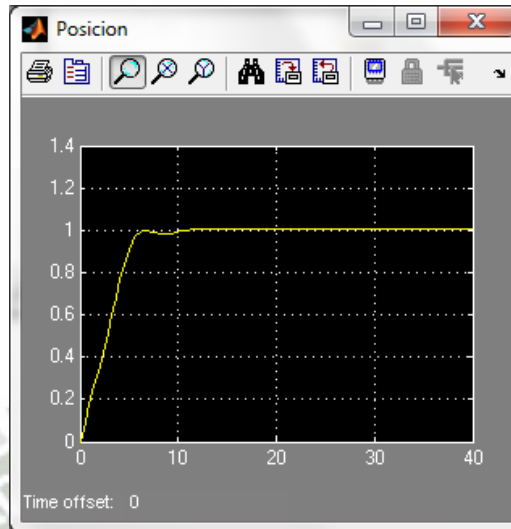


Figura 6.9. Comportamiento de la posición, control PID

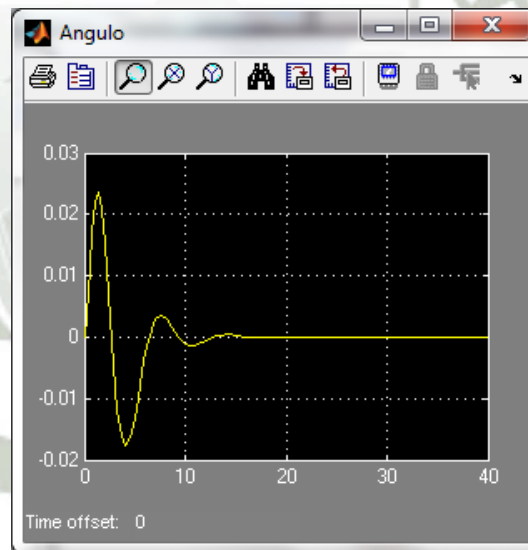


Figura 6.10. Comportamiento de la posición, control PID

Resultados de la posición

- Sobreimpulso: 0 %
- Tiempo de subida: 6 segundos

- Tiempo de establecimiento: a los 12 segundos
- Error estado estable: 0

Resultados del ángulo

- Magnitud de la oscilación: 0.025 radianes
- Tiempo de establecimiento: a los 16 segundos

Todos estos resultados se pueden apreciar en la Figuras 6.9 y 6.10.

6.1.4. Resultados Control Difuso

Las siguientes graficas se obtuvieron gracias a las ganancias aplicadas en dicho controlador, para poder la estabilidad desea.

Las ganancias obtenidas fueron para la posición son $K_p = 4.6$ $k_d = 7.8$, y para el ángulo son $K_p = 0.0001$, $k_d = 0.001$ y para los dos casos el valor de k_i es el mismo para los dos, $k_i = 0.13$

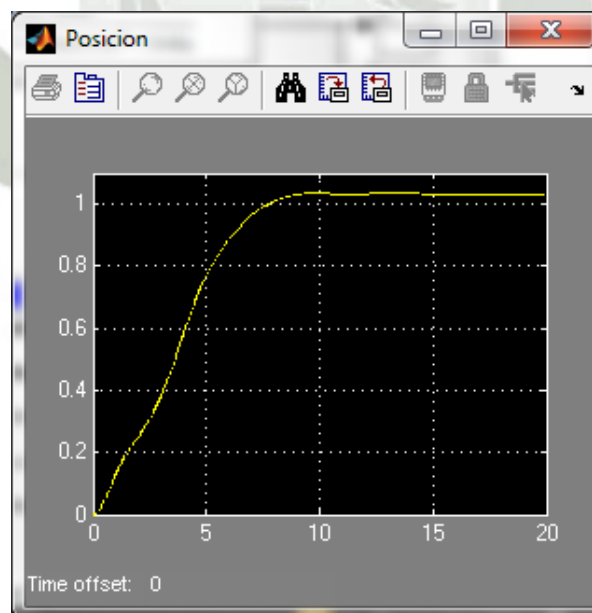


Figura 6.11. Comportamiento de la posición, control difuso

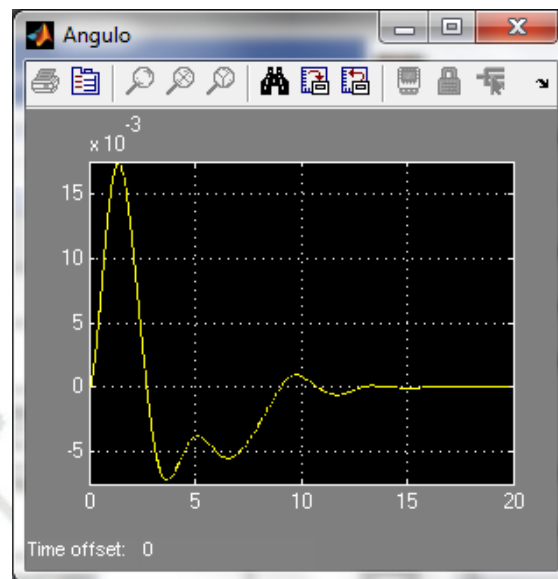


Figura 6.12. Comportamiento de la posición, control difuso

Resultados de la posición

- Sobreimpulso: 0 %
- Tiempo de subida: 5 segundos
- Tiempo de establecimiento: a los 10 segundos
- Error estado estable: 0

Resultados del ángulo

- Magnitud de la oscilación: 0.015 radianes
- Tiempo de establecimiento: a los 15 segundos

Todos estos resultados se pueden apreciar en la Figuras 6.11 y 6.12.

Como se ha podido apreciar en las Figuras anteriores y en los resultados obtenidos del Simulink, el control PID difuso tiene mejores cualidades para dar solución al control de la

Grúa Puente ya que no presenta sobre impulso, el tiempo de subida es menor y el tiempo de establecimiento son mucho menores a los demás y no presenta error en estado estable. Así mismo la magnitud de oscilación y el tiempo de establecimiento de la carga en comparación con los otros controladores.



CAPITULO VII

ANALIZANDO LOS RESULTADOS CON EL SIMULADOR

Como se ha visto en el capítulo anterior el análisis de las gráficas obtenidas mediante Simulink, en este capítulo se hace lo mismo solo con la diferencia que se utiliza el simulador para obtener las mismas gráficas y su posterior análisis.

7.1. Resultados

Así mismo en el capítulo anterior se procede a analizar las gráficas que siguen, se toma en cuenta tres, ítems para el control de la posición los cuales son, el sobre impulso, el tiempo de subida, el tiempo de establecimiento y el error en estado estable y para el control del ángulo son dos, la magnitud de las oscilaciones y el tiempo de establecimiento

7.1.1. Resultados Control PD

Las siguientes graficas se obtuvieron gracias a las ganancias aplicadas en dicho controlador, para poder la estabilidad desea.

Las ganancias obtenidas fueron para la posición son $K_p = 9000$, $K_d = 30000$ y para el ángulo son $K_p = 150$, $K_d = 2500$

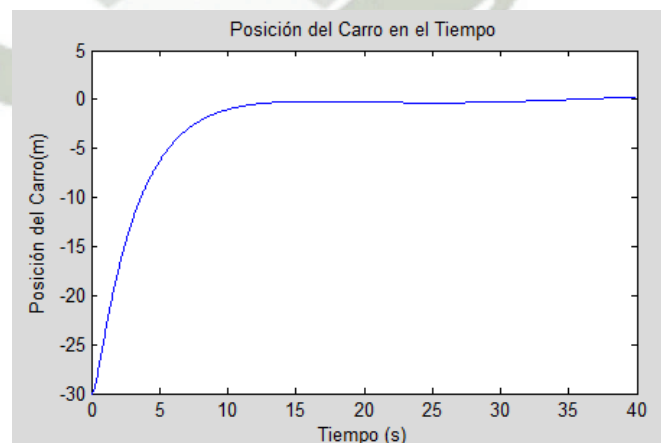


Figura 7.1. Comportamiento de la posición, control PD

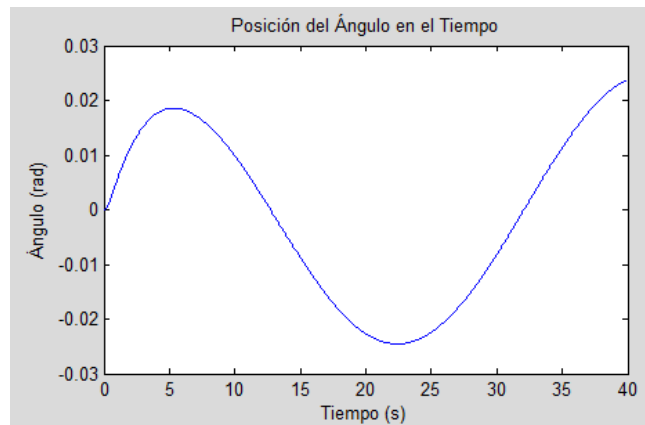


Figura 7.2. Comportamiento del ángulo, control PD

Resultados de la posición

- Sobreimpulso: 0 %
- Tiempo de subida: 5 segundos
- Tiempo de establecimiento: a los 15 segundos
- Error estado estable: 0

Resultados del ángulo

- Magnitud de la oscilación: 0.02 radianes
- Tiempo de establecimiento: es estable todo el recorrido

Todos estos resultados se pueden apreciar en la Figuras 7.1. y 7.2.

7.1.2. Resultados Control PID

Las siguientes graficas se obtuvieron gracias a las ganancias aplicadas en dicho controlador, para poder la estabilidad desea.

Las ganancias obtenidas fueron para la posición son $K_p = 16000$, $k_d = 45000$, $K_i = 3900$ y para el ángulo son $K_p = 1500$, $k_d = 2000$, $K_i = 500$

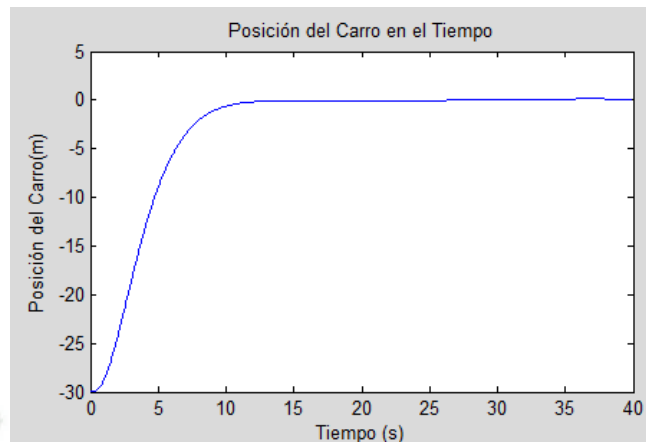


Figura 7.3. Comportamiento de la posición, control PID

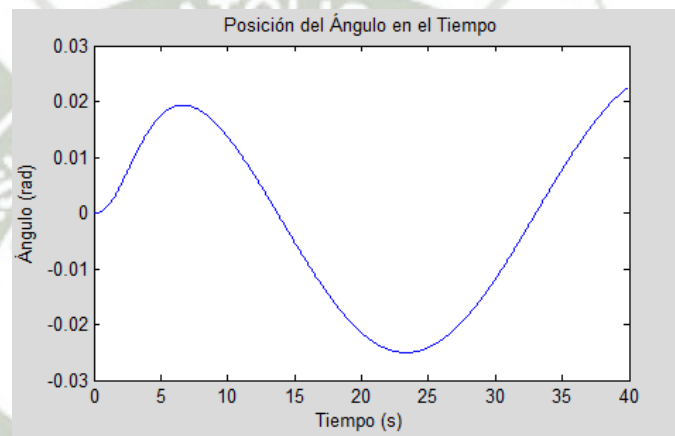


Figura 7.4. Comportamiento de la posición, control PID

Resultados de la posición

- Sobreimpulso: 0 %
- Tiempo de subida: 7 segundos
- Tiempo de establecimiento: a los 12 segundos
- Error estado estable: 0

Resultados del ángulo

- Magnitud de la oscilación: 0.02 radianes
- Tiempo de establecimiento: es estable todo el recorrido

Todos estos resultados se pueden apreciar en la Figuras 7.3 y 7.4.

7.1.3. Resultados Control Difuso

Las siguientes graficas se obtuvieron gracias a las ganancias aplicadas en dicho controlador, para poder la estabilidad desea.

Las ganancias obtenidas fueron para la posición son $K_p = 0.198$, $k_d = 2.4$, y para el ángulo son $K_p = 0.001$, $k_d = 1.1$ y para los dos casos el valor de k_i es el mismo para los dos, $k_i = 1.45$

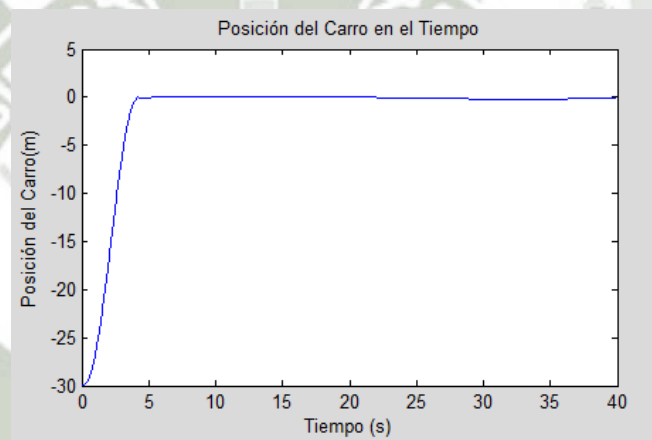


Figura 7.5. Comportamiento de la posición, control difuso

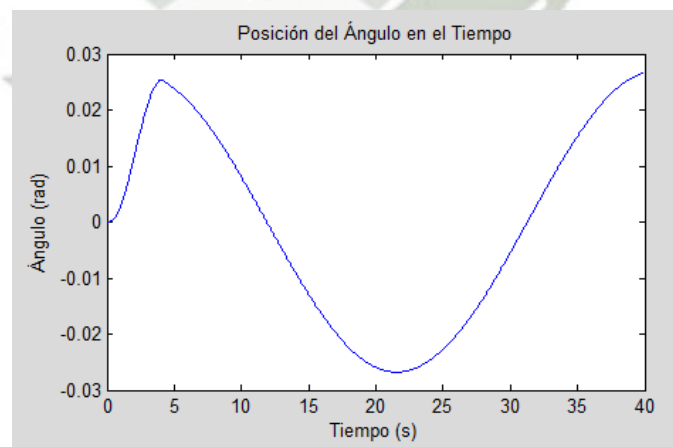


Figura 7.6. Comportamiento de la posición, control difuso

Resultados de la posición

- Sobreimpulso: 0 %
- Tiempo de subida: 3 segundos
- Tiempo de establecimiento: a los 5 segundos
- Error estado estable: 0

Resultados del ángulo

- Magnitud de la oscilación: 0.025 radianes
- Tiempo de establecimiento: es estable todo el recorrido

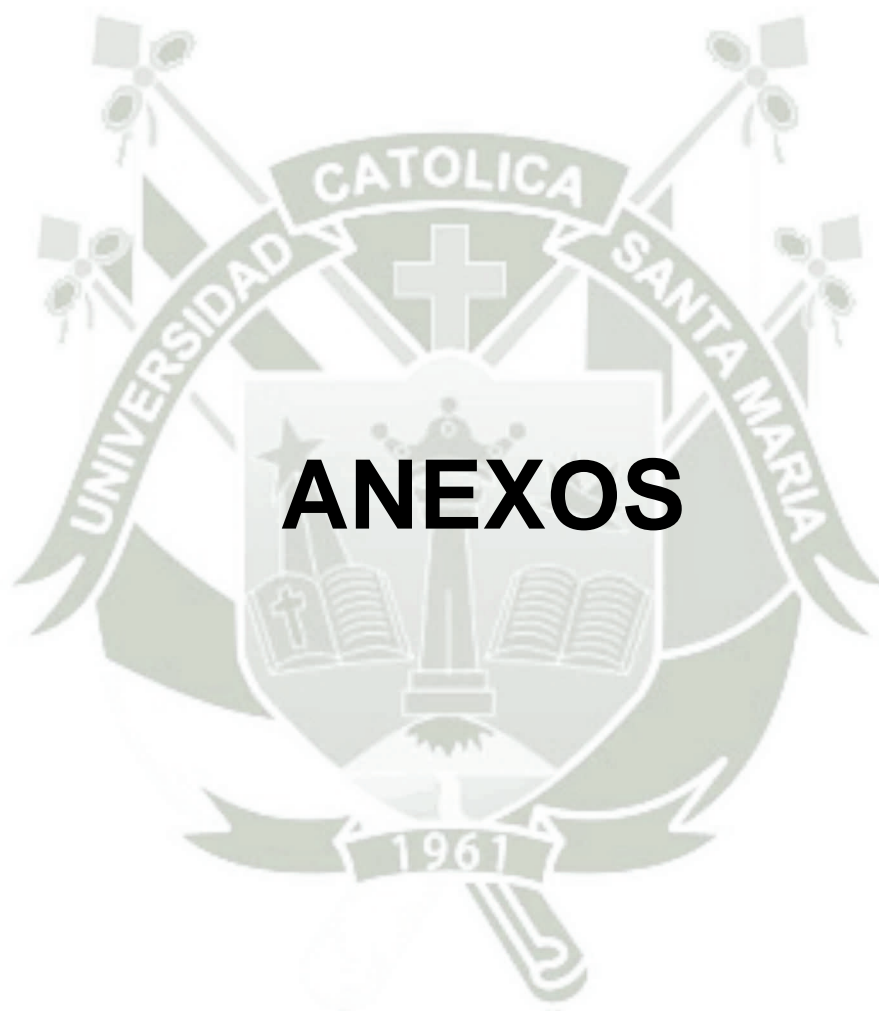
Todos estos resultados se pueden apreciar en la Figuras 7.5 y 7.6.

Como se ha podido apreciar en las Figuras anteriores y en los resultados obtenidos en el simulador, el control PID difuso tiene mejores cualidades para dar solución al control de la Grúa Puente ya que no presenta sobre impulso, el tiempo de subida es menor y el tiempo de establecimiento son mucho menores a los demás y no presenta error en estado estable. Así mismo la magnitud de oscilación y el tiempo de establecimiento de la carga en comparación con los otros controladores.

CONCLUSIONES

El desarrollo de este trabajo nos llevó a importantes conclusiones sobre el desarrollo de controlares PID difusos en comparación en comparación con controladores tradicionales; teniendo especial énfasis en su aplicación al control de grúas puente. Estas conclusiones se listan a seguir:

1. Que el controlar PID difuso cumple mejor con la tarea de minimizar la oscilación de la carga en el proceso de transporte.
2. Que la respuesta del controlador PID difuso era lo que se esperaba, gracias a las reglas difusas que emulan el juicio de un operario con experiencia.
3. En el momento de comparar los resultados, se vio que el controlador difuso resultó ser la mejor opción de solución para el control de las grúas puente.
4. Al comparar las gráficas del Simulink y las del simulador se nota leves diferencias, ya que las primeras son con una señal de entrada del tipo escalón unitario, y la segunda es cómo se comporta realmente.
5. Las ganancias obtenidas en el Simulink son el punto de partida para obtener las ganancias en el simulador.
6. Teniendo en cuenta los resultados, se puede observar que el tiempo de estabilidad de la posición y de la carga es mucho menor en relación a los otros controladores, con lo significa que el tiempo de trabajo se reduce, es decir la productividad aumenta.
7. Teniendo en cuenta que la productividad aumenta al utilizar un controlador PID difuso se puede concluir que también aumenta la cantidad de objetos realmente trasladados; es decir la eficiencia del proceso también aumenta.



ANEXO 1

```

function ControlPGruaFinal(comando)
if comando == 1 % Boton Ejecutar

    %parametros generales de simulador
    dt = 0.001; %tiempo de integración (debe ser lo menor posible)
    TControl = 0.033; %tiempo de muestreo de los sensores y el controlador (30 cuadros por
segundo es un buen valor)

    %Recogiendo los valores de los parámetros del simulador
    GravedadEdit = findobj('Tag','GravedadEdit');
    Gravedad = str2double(get(GravedadEdit,'String'))
    LongitudEdit = findobj('Tag','LongitudEdit');
    Longitud = str2double(get(LongitudEdit,'String'))
    MasaCarroEdit = findobj('Tag','MasaCarroEdit');
    MasaCarro = str2double(get(MasaCarroEdit,'String'))
    MasaCargaEdit = findobj('Tag','MasaCargaEdit');
    MasaCarga = str2double(get(MasaCargaEdit,'String'))
    MomentoInerciaEdit = findobj('Tag','MomentoInerciaEdit');
    MomentoInercia = str2double(get(MomentoInerciaEdit,'String'))
    FriccionCarroEdit = findobj('Tag','FriccionCarroEdit');
    FriccionCarro = str2double(get(FriccionCarroEdit,'String'))
    FriccionRotacionalEdit = findobj('Tag','FriccionRotacionalEdit');
    FriccionRotacional = str2double(get(FriccionRotacionalEdit,'String'))
    TamanhoVigaEdit = findobj('Tag','TamanhoVigaEdit');
    TamanhoViga = str2double(get(TamanhoVigaEdit,'String'))
    PosInicialEdit = findobj('Tag','PosInicialEdit');
    PosInicial = str2double(get(PosInicialEdit,'String'))
    PosFinalEdit = findobj('Tag','PosFinalEdit');
    PosFinal = str2double(get(PosFinalEdit,'String'))
    TSimulacionEdit = findobj('Tag','TSimulacionEdit');
    TSimulacion = int32(str2double(get(TSimulacionEdit,'String')))

    %Calculando las Funciones de Transferencia
    %Función de transferencia del angulo
    s4 = (MasaCarro+MasaCarga)*MomentoInercia + MasaCarro*MasaCarga*(Longitud^2);
    s3 = (MasaCarro+MasaCarga)*FriccionRotacional +
(MomentoInercia+MasaCarga*(Longitud^2))*FriccionCarro;
    s2 =
(MasaCarro+MasaCarga)*MasaCarga*Longitud*Gravedad+FriccionCarro*FriccionRotacional;
    s1 = FriccionCarro*MasaCarga*Longitud*Gravedad;
    TransferTheta = tf([MasaCarga*Longitud 0 0],[s4 s3 s2 s1 0])
    %
    TransferX = tf([MomentoInercia+MasaCarga*(Longitud^2) FriccionRotacional
MasaCarga*Longitud*Gravedad],[s4 s3 s2 s1 0])
    %
    %Identificando el tipo de control Elegido por el Usuario
    % 0 --> Tipo no identificado
    % 1 --> Control P
    % 2 --> Control PD
    % 3 --> Control PDI

```

```

% 4 --> Control Difuso
PIDRadio = findobj('Tag','PIDRadio');
PControlCheck = findobj('Tag','PControlCheck');
DControlCheck = findobj('Tag','DControlCheck');
IControlCheck = findobj('Tag','IControlCheck');
PID = get(PIDRadio,'Value');
PControl = get(PControlCheck,'Value');
DControl = get(DControlCheck,'Value');
IControl = get(IControlCheck,'Value');

Tipo = 0;
if PID == 1
    if PControl==1
        if DControl==1
            if IControl==1
                Tipo = 3; %Control PDI
                Estado = [0,0,PosInicial,0]; %[theta,dtheta,x,dx]
                Deseado = [0, PosFinal]; %[theta,x]
                SenalControl = 0; %u(t)
                tiempo = 0;
                PControlEditTheta = findobj('Tag','PControlEditTheta');
                PControlTheta = str2double(get(PControlEditTheta,'String'));
                Kp_theta = PControlTheta
                PControlEditPosicion = findobj('Tag','PControlEditPosicion');
                PControlPosicion = str2double(get(PControlEditPosicion,'String'));
                Kp_posicion = PControlPosicion

                DControlEditTheta = findobj('Tag','DControlEditTheta');
                DControlTheta = str2double(get(DControlEditTheta,'String'));
                Td_theta = DControlTheta
                DControlEditPosicion = findobj('Tag','DControlEditPosicion');
                DControlPosicion = str2double(get(DControlEditPosicion,'String'));
                Td_posicion = DControlPosicion

                IControlEditTheta = findobj('Tag','IControlEditTheta');
                IControlTheta = str2double(get(IControlEditTheta,'String'));
                Ti_theta = IControlTheta
                IControlEditPosicion = findobj('Tag','IControlEditPosicion');
                IControlPosicion = str2double(get(IControlEditPosicion,'String'));
                Ti_posicion = IControlPosicion

                Xs = [];
                Thetas = [];
                Uts = [];
                Tiempos = [];
                error_angulo = Deseado(1)-Estado(1);
                error_posicion = Deseado(2)-Estado(3);
                error_angulo_ant = error_angulo;
                error_posicion_ant = error_posicion;
                SenalControl_ant = 0;
                while (tiempo < TSimulacion)
                    tiempo = tiempo + dt;
            end
        end
    end
end

```

```

    aceleracion_theta = (MasaCarga*Longitud*(SenalControl-FriccionCarro*Estado(4))-
(MasaCarro+MasaCarga)*(FriccionRotacional*Estado(2)+MasaCarga*Longitud*Gravedad*Estado(
1)))/(MasaCarro*(MomentoInercia+MasaCarga*(Longitud^2)+MasaCarga*MomentoInercia));
    aceleracion_posicion = ((MomentoInercia + MasaCarro*(Longitud^2))*(SenalControl-
FriccionCarro*Estado(4))+MasaCarga*Longitud*(MasaCarga*Longitud*Gravedad*Estado(1)-
FriccionRotacional*Estado(2)))/(MasaCarro*(MomentoInercia+MasaCarga*(Longitud^2))+MasaCar
ga*MomentoInercia);
    Estado(4) = Estado(4)+aceleracion_posicion*dt; %dx
    Estado(3) = Estado(3)+Estado(4)*dt; %x
    Estado(2) = Estado(2)+aceleracion_theta*dt; %dtheta
    Estado(1) = Estado(1)+Estado(2)*dt; %theta
    if (mod(int32(tiempo*1000),int32(TControl*1000))==0)
        error_angulo_ant2 = error_angulo_ant;
        error_angulo_ant = error_angulo;
        error_angulo = Deseado(1)-Estado(1);
        error_posicion_ant2 = error_posicion_ant;
        error_posicion_ant = error_posicion;
        error_posicion = Deseado(2)-Estado(3);
        SenalControl_ant2 = SenalControl_ant;
        SenalControl_ant = SenalControl;
        K1_theta = Kp_theta + Td_theta/TControl + Ti_theta*TControl;
        K2_theta = Ti_theta*TControl - 2*Td_theta/TControl;
        K3_theta = Td_theta/TControl - Kp_theta;
        K1_posicion = Kp_posicion + Td_posicion/TControl + Ti_posicion*TControl;
        K2_posicion = Ti_posicion*TControl - 2*Td_posicion/TControl;
        K3_posicion = Td_posicion/TControl - Kp_posicion;

        SenalControl = SenalControl_ant2 + (K1_theta*error_angulo +
K2_theta*error_angulo_ant + K3_theta*error_angulo_ant2 + K1_posicion*error_posicion +
K2_posicion*error_posicion_ant + K3_posicion*error_posicion_ant2)/2;

        Xs = [Xs Estado(3)];
        Thetas = [Thetas Estado(1)];
        Uts = [Uts SenalControl];
        Tiempos = [Tiempos tiempo];

        CatetoOpuesto = sin(Estado(1))*Longitud;
        CatetoAdyacente = cos(Estado(1))*Longitud;
        PosCarga = [Estado(3)-CatetoOpuesto,Longitud*0.1456-CatetoAdyacente];
        PGruaA = findobj('Tag','PGruaA');
        axes(PGruaA);
        plot([-TamanhoViga*0.7 TamanhoViga*0.7],[0 0],'g','linewidth',2) % Linea
Representando la viga
        hold on
        plot([Estado(3) PosCarga(1)],[Longitud*0.1456 PosCarga(2)],'r','linewidth',4) %
Linea representando el cable de la carga
        hold on
        plot([Estado(3)+TamanhoViga/10 Estado(3)-TamanhoViga/10 Estado(3)-
TamanhoViga/10 Estado(3)+TamanhoViga/10 Estado(3)+TamanhoViga/10],[Longitud*0.1456
Longitud*0.1456 0 0 Longitud*0.1456],'b','linewidth',5) % Representacion del carro
        hold on
        plot(PosCarga(1),PosCarga(2),'*m','linewidth',20) %Representación de la Carga
        hold on
        plot(Estado(3)+TamanhoViga/20,0,'*b','linewidth',13) % Rueda delantera del Carro
    
```

```

hold on
plot(Estado(3)-TamanhoViga/20,0,'*b','linewidth',13) % Rueda Trasera del Carro
hold off
set(PGruaA,'XLim', [-TamanhoViga*0.7 TamanhoViga*0.7]);
set(PGruaA,'YLim', [-Longitud*1.5 Longitud*1.5]);
set(PGruaA,'Tag','PGruaA');
grid on
set(PGruaA,'Title',text('String','Animación del Puente Grua'));
set(PGruaA,'XLabel',text('String','Posición X (m)'));
set(PGruaA,'YLabel',text('String','Posición Y (m)'));
drawnow;
end
end

U_t = findobj('Tag','U_t');
axes(U_t);
plot(Tiempos,Uts);
set(U_t,'Tag','U_t');
set(U_t,'Title',text('String','Magnitud de U(t) en el Tiempo'));
set(U_t,'XLabel',text('String','Tiempo (s)'));
set(U_t,'YLabel',text('String','U(t)'));

posCarro = findobj('Tag','posCarro');
axes(posCarro);
plot(Tiempos,Xs);
set(posCarro,'Tag','posCarro');
set(posCarro,'Title',text('String','Posición del Carro en el Tiempo'));
set(posCarro,'XLabel',text('String','Tiempo (s)'));
set(posCarro,'YLabel',text('String','Posición del Carro(m)'));

posAngulo = findobj('Tag','posAngulo');
axes(posAngulo);
plot(Tiempos,Thetas)
set(posAngulo,'Tag','posAngulo');
set(posAngulo,'Title',text('String','Posición del Ángulo en el Tiempo'));
set(posAngulo,'XLabel',text('String','Tiempo (s)'));
set(posAngulo,'YLabel',text('String','Ángulo (rad)'));
else
Tipo = 2; %Contro PD
Estado = [0,0,PosInicial,0]; %[theta,dtheta,x,dx]
Deseado = [0, PosFinal]; %[theta,x]
SenalControl = 0; %u(t)
tiempo = 0;
PControlEditTheta = findobj('Tag','PControlEditTheta');
PControlTheta = str2double(get(PControlEditTheta,'String'));
Kp_theta = PControlTheta
PControlEditPosicion = findobj('Tag','PControlEditPosicion');
PControlPosicion = str2double(get(PControlEditPosicion,'String'));
Kp_posicion = PControlPosicion

DControlEditTheta = findobj('Tag','DControlEditTheta');
DControlTheta = str2double(get(DControlEditTheta,'String'));
Td_theta = DControlTheta
DControlEditPosicion = findobj('Tag','DControlEditPosicion');

```

```

DControlPosicion = str2double(get(DControlEditPosicion,'String'));
Td_posicion = DControlPosicion

Xs = [];
Thetas = [];
Uts = [];
Tiempos = [];
error_angulo = Deseado(1)-Estado(1);
error_posicion = Deseado(2)-Estado(3);
while (tiempo < TSimulacion)
    tiempo = tiempo + dt;
    aceleracion_theta = (MasaCarga*Longitud*(SenalControl-FriccionCarro*Estado(4))-
(MasaCarro+MasaCarga)*(FriccionRotacional*Estado(2)+MasaCarga*Longitud*Gravedad*Estado(
1)))/(MasaCarro*(MomentoInercia+MasaCarga*(Longitud^2)+MasaCarga*MomentoInercia));
    aceleracion_posicion = ((MomentoInercia + MasaCarro*(Longitud^2))*(SenalControl-
FriccionCarro*Estado(4))+MasaCarga*Longitud*(MasaCarga*Longitud*Gravedad*Estado(1)-
FriccionRotacional*Estado(2)))/(MasaCarro*(MomentoInercia+MasaCarga*(Longitud^2))+MasaCar
ga*MomentoInercia);
    Estado(4) = Estado(4)+aceleracion_posicion*dt; %dx
    Estado(3) = Estado(3)+Estado(4)*dt; %x
    Estado(2) = Estado(2)+aceleracion_theta*dt; %dtheta
    Estado(1) = Estado(1)+Estado(2)*dt; %theta
    if (mod(int32(tiempo*1000),int32(TControl*1000))==0)
        error_angulo_ant = error_angulo;
        error_angulo = Deseado(1)-Estado(1);
        error_posicion_ant = error_posicion;
        error_posicion = Deseado(2)-Estado(3);
        SenalControl = (Kp_theta*error_angulo + Td_theta*(error_angulo-
error_angulo_ant)/TControl + Kp_posicion*error_posicion + Td_posicion*(error_posicion-
error_posicion_ant)/TControl)/2;

    Xs = [Xs Estado(3)];
    Thetas = [Thetas Estado(1)];
    Uts = [Uts SenalControl];
    Tiempos = [Tiempos tiempo];

    CatetoOpuesto = sin(Estado(1))*Longitud;
    CatetoAdyacente = cos(Estado(1))*Longitud;
    PosCarga = [Estado(3)-CatetoOpuesto,Longitud*0.1456-CatetoAdyacente];
    PGruaA = findobj('Tag','PGruaA');
    axes(PGruaA);
    plot([-TamanhoViga*0.7 TamanhoViga*0.7],[0 0],'g','linewidth',2) % Linea
Representando la viga
    hold on
    plot([Estado(3) PosCarga(1)],[Longitud*0.1456 PosCarga(2)],'r','linewidth',4) %
Linea representando el cable de la carga
    hold on
    plot([Estado(3)+TamanhoViga/10 Estado(3)-TamanhoViga/10 Estado(3)-
TamanhoViga/10 Estado(3)+TamanhoViga/10 Estado(3)+TamanhoViga/10],[Longitud*0.1456
Longitud*0.1456 0 0 Longitud*0.1456],'b','linewidth',5) % Representacion del carro
    hold on
    plot(PosCarga(1),PosCarga(2),'*m','linewidth',20) %Representación de la Carga
    hold on
    plot(Estado(3)+TamanhoViga/20,0,'*b','linewidth',13) % Rueda delantera del Carro

```

```

        hold on
        plot(Estado(3)-TamanhoViga/20,0,'*b','linewidth',13) % Rueda Trasera del Carro
        hold off
        set(PGruaA,'XLim', [-TamanhoViga*0.7 TamanhoViga*0.7]);
        set(PGruaA,'YLim', [-Longitud*1.5 Longitud*1.5]);
        set(PGruaA,'Tag','PGruaA');
        grid on
        set(PGruaA,'Title',text('String','Animación del Puente Grua'));
        set(PGruaA,'XLabel',text('String','Posición X (m)'));
        set(PGruaA,'YLabel',text('String','Posición Y (m)'));
        drawnow;
    end
end

U_t = findobj('Tag','U_t');
axes(U_t);
plot(Tiempos,Uts);
set(U_t,'Tag','U_t');
set(U_t,'Title',text('String','Magnitud de U(t) en el Tiempo'));
set(U_t,'XLabel',text('String','Tiempo (s)'));
set(U_t,'YLabel',text('String','U(t)'));

posCarro = findobj('Tag','posCarro');
axes(posCarro);
plot(Tiempos,Xs);
set(posCarro,'Tag','posCarro');
set(posCarro,'Title',text('String','Posición del Carro en el Tiempo'));
set(posCarro,'XLabel',text('String','Tiempo (s)'));
set(posCarro,'YLabel',text('String','Posición del Carro(m)'));

posAngulo = findobj('Tag','posAngulo');
axes(posAngulo);
plot(Tiempos,Thetas)
set(posAngulo,'Tag','posAngulo');
set(posAngulo,'Title',text('String','Posición del Ángulo en el Tiempo'));
set(posAngulo,'XLabel',text('String','Tiempo (s)'));
set(posAngulo,'YLabel',text('String','Ángulo (rad)'));
end
else
if IControl==1
    Tipo = 0; %Tipo no identificado
else
    Tipo = 1; %Control P
    Estado = [0,0,PosInicial,0]; %[theta,dtheta,x,dx]
    Deseado = [0, PosFinal]; %[theta,x]
    SenalControl = 0; %u(t)
    tiempo = 0;
    PControlEditTheta = findobj('Tag','PControlEditTheta');
    PControlTheta = str2double(get(PControlEditTheta,'String'));
    Kp_theta = PControlTheta
    PControlEditPosicion = findobj('Tag','PControlEditPosicion');
    PControlPosicion = str2double(get(PControlEditPosicion,'String'));
    Kp_posicion = PControlPosicion

```

```

Xs = [];
Thetas = [];
Uts = [];
Tiempos = [];
while (tiempo < TSimulacion)
    tiempo = tiempo + dt;
    aceleracion_theta = (MasaCarga*Longitud*(SenalControl-FriccionCarro*Estado(4))-
(MasaCarro+MasaCarga)*(FriccionRotacional*Estado(2)+MasaCarga*Longitud*Gravedad*Estado(
1)))/(MasaCarro*(MomentoInercia+MasaCarga*(Longitud^2))+MasaCarga*MomentoInercia));
    aceleracion_posicion = ((MomentoInercia + MasaCarro*(Longitud^2))*(SenalControl-
FriccionCarro*Estado(4))+MasaCarga*Longitud*(MasaCarga*Longitud*Gravedad*Estado(1)-
FriccionRotacional*Estado(2)))/(MasaCarro*(MomentoInercia+MasaCarga*(Longitud^2))+MasaCar
ga*MomentoInercia);
    Estado(4) = Estado(4)+aceleracion_posicion*dt; %dx
    Estado(3) = Estado(3)+Estado(4)*dt; %x
    Estado(2) = Estado(2)+aceleracion_theta*dt; %dtheta
    Estado(1) = Estado(1)+Estado(2)*dt; %theta
    if (mod(int32(tiempo*1000),int32(TControl*1000))==0)
        error_angulo = Deseado(1)-Estado(1);
        error_posicion = Deseado(2)-Estado(3);
        SenalControl = (Kp_theta*error_angulo + Kp_posicion*error_posicion)/2;

    Xs = [Xs Estado(3)];
    Thetas = [Thetas Estado(1)];
    Uts = [Uts SenalControl];
    Tiempos = [Tiempos tiempo];

    CatetoOpuesto = sin(Estado(1))*Longitud;
    CatetoAdyacente = cos(Estado(1))*Longitud;
    PosCarga = [Estado(3)-CatetoOpuesto,Longitud*0.1456-CatetoAdyacente];
    PGruaA = findobj('Tag','PGruaA');
    axes(PGruaA);
    plot([-TamanhoViga*0.7 TamanhoViga*0.7],[0 0],'g','linewidth',2) % Linea
Representando la viga
    hold on
    plot([Estado(3) PosCarga(1)],[Longitud*0.1456 PosCarga(2)],'r','linewidth',4) %
Linea representando el cable de la carga
    hold on
    plot([Estado(3)+TamanhoViga/10 Estado(3)-TamanhoViga/10 Estado(3)-
TamanhoViga/10 Estado(3)+TamanhoViga/10 Estado(3)+TamanhoViga/10],[Longitud*0.1456
Longitud*0.1456 0 0 Longitud*0.1456],'b','linewidth',5) % Representacion del carro
    hold on
    plot(PosCarga(1),PosCarga(2),'*m','linewidth',20) %Representación de la Carga
    hold on
    plot(Estado(3)+TamanhoViga/20,0,'*b','linewidth',13) % Rueda delantera del Carro
    hold on
    plot(Estado(3)-TamanhoViga/20,0,'*b','linewidth',13) % Rueda Trasera del Carro
    hold off
    set(PGruaA,'XLim', [-TamanhoViga*0.7 TamanhoViga*0.7]);
    set(PGruaA,'YLim', [-Longitud*1.5 Longitud*1.5]);
    set(PGruaA,'Tag','PGruaA');
    grid on
    set(PGruaA,'Title',text('String','Animación del Puente Grua'));
    set(PGruaA,'XLabel',text('String','Posición X (m)'));

```

```

        set(PGruaA,'YLabel',text('String','Posición Y (m)'));
        drawnow;
    end

end

U_t = findobj('Tag','U_t');
axes(U_t);
plot(Tiempos,Uts);
set(U_t,'Tag','U_t');
set(U_t,'Title',text('String','Magnitud de U(t) en el Tiempo'));
set(U_t,'XLabel',text('String','Tiempo (s)'));
set(U_t,'YLabel',text('String','U(t)'));

posCarro = findobj('Tag','posCarro');
axes(posCarro);
plot(Tiempos,Xs);
set(posCarro,'Tag','posCarro');
set(posCarro,'Title',text('String','Posición del Carro en el Tiempo'));
set(posCarro,'XLabel',text('String','Tiempo (s)'));
set(posCarro,'YLabel',text('String','Posición del Carro(m)'));

posAngulo = findobj('Tag','posAngulo');
axes(posAngulo);
plot(Tiempos,Thetas);
set(posAngulo,'Tag','posAngulo');
set(posAngulo,'Title',text('String','Posición del Ángulo en el Tiempo'));
set(posAngulo,'XLabel',text('String','Tiempo (s)'));
set(posAngulo,'YLabel',text('String','Ángulo (rad)'));

end
end
end
else
    Tipo = 4; %Control PDI Difuso
    Estado = [0,0,PosInicial,0]; %[theta,dtheta,x,dx]
    Deseado = [0, PosFinal]; %[theta,x]
    SenalControl = 0; %u(t)
    tiempo = 0;
    PControlEditTheta = findobj('Tag','PControlEditTheta');
    PControlTheta = str2double(get(PControlEditTheta,'String'));
    Kp_theta = PControlTheta;
    PControlEditPosicion = findobj('Tag','PControlEditPosicion');
    PControlPosicion = str2double(get(PControlEditPosicion,'String'));
    Kp_posicion = PControlPosicion;

    DControlEditTheta = findobj('Tag','DControlEditTheta');
    DControlTheta = str2double(get(DControlEditTheta,'String'));
    Td_theta = DControlTheta;
    DControlEditPosicion = findobj('Tag','DControlEditPosicion');
    DControlPosicion = str2double(get(DControlEditPosicion,'String'));
    Td_posicion = DControlPosicion;

    IControlEdit = findobj('Tag','IControlEditTheta');

```

```

IControl = str2double(get(IControlEdit,'String'));
Ti = IControl;

Xs = [];
Thetas = [];
Uts = [];
Tiempos = [];
error_angulo = Deseado(1)-Estado(1);
error_posicion = Deseado(2)-Estado(3);

while (tiempo < TSimulacion)
    tiempo = tiempo + dt;
    aceleracion_theta = (MasaCarga*Longitud*(SenalControl-FriccionCarro*Estado(4))-
(MasaCarro+MasaCarga)*(FriccionRotacional*Estado(2)+MasaCarga*Longitud*Gravedad*Estado(
1)))/(MasaCarro*(MomentoInercia+MasaCarga*(Longitud^2)+MasaCarga*MomentoInercia));
    aceleracion_posicion = ((MomentoInercia + MasaCarro*(Longitud^2))*(SenalControl-
FriccionCarro*Estado(4))+MasaCarga*Longitud*(MasaCarga*Longitud*Gravedad*Estado(1)-
FriccionRotacional*Estado(2)))/(MasaCarro*(MomentoInercia+MasaCarga*(Longitud^2))+MasaCar
ga*MomentoInercia);
    Estado(4) = Estado(4)+aceleracion_posicion*dt; %dx
    Estado(3) = Estado(3)+Estado(4)*dt; %x
    Estado(2) = Estado(2)+aceleracion_theta*dt; %dtheta
    Estado(1) = Estado(1)+Estado(2)*dt; %theta
    if (mod(int32(tiempo*1000),int32(TControl*1000))==0)
        error_angulo_ant = error_angulo;
        error_angulo = Deseado(1)-Estado(1);
        error_posicion_ant = error_posicion;
        error_posicion = Deseado(2)-Estado(3);

    TamEspacioDecision = 12; %Radianes
    e_inicios_theta = [-TamEspacioDecision/2 -(3*TamEspacioDecision/8) -
(TamEspacioDecision/4) -(TamEspacioDecision/8) 0 (TamEspacioDecision/8)
(TamEspacioDecision/4)];
    e_centros_theta = [- (3*TamEspacioDecision/8) -(TamEspacioDecision/4) -
(TamEspacioDecision/8) 0 (TamEspacioDecision/8) (TamEspacioDecision/4)
3*TamEspacioDecision/8];
    e_finales_theta = [- (TamEspacioDecision/4) -(TamEspacioDecision/8) 0
(TamEspacioDecision/8) (TamEspacioDecision/4) 3*TamEspacioDecision/8
TamEspacioDecision/2];
    TamEspacioDecision = 2.4; %Radianes
    de_inicios_theta = [- (TamEspacioDecision/2) -(3*TamEspacioDecision/8) -
(TamEspacioDecision/4) -(TamEspacioDecision/8) 0 (TamEspacioDecision/8)
(TamEspacioDecision/4)];
    de_centros_theta = [- (3*TamEspacioDecision/8) -(TamEspacioDecision/4) -
(TamEspacioDecision/8) 0 (TamEspacioDecision/8) (TamEspacioDecision/4)
3*TamEspacioDecision/8];
    de_finales_theta = [- (TamEspacioDecision/4) -(TamEspacioDecision/8) 0
(TamEspacioDecision/8) (TamEspacioDecision/4) 3*TamEspacioDecision/8
TamEspacioDecision/2];
    TamEspacioDecision = TamanhoViga;
    e_inicios_posicion = [- (TamEspacioDecision/2) -(3*TamEspacioDecision/8) -
(TamEspacioDecision/4) -(TamEspacioDecision/8) 0 (TamEspacioDecision/8)
(TamEspacioDecision/4)];

```

```

    e_centros_posicion = [-(3*TamEspacioDecision/8) -(TamEspacioDecision/4) -
(TamEspacioDecision/8) 0 (TamEspacioDecision/8) (TamEspacioDecision/4)
3*TamEspacioDecision/8];
    e_finales_posicion = [-(TamEspacioDecision/4) -(TamEspacioDecision/8) 0
(TamEspacioDecision/8) (TamEspacioDecision/4) 3*TamEspacioDecision/8
TamEspacioDecision/2];
    TamEspacioDecision = TamanoViga*0.2;
    de_inicios_posicion = [-(TamEspacioDecision/2) -(3*TamEspacioDecision/8) -
(TamEspacioDecision/4) -(TamEspacioDecision/8) 0 (TamEspacioDecision/8)
(TamEspacioDecision/4)];
    de_centros_posicion = [-(3*TamEspacioDecision/8) -(TamEspacioDecision/4) -
(TamEspacioDecision/8) 0 (TamEspacioDecision/8) (TamEspacioDecision/4)
3*TamEspacioDecision/8];
    de_finales_posicion = [-(TamEspacioDecision/4) -(TamEspacioDecision/8) 0
(TamEspacioDecision/8) (TamEspacioDecision/4) 3*TamEspacioDecision/8
TamEspacioDecision/2];
    TamEspacioDecision = 250000;
    U_inicios = [-(TamEspacioDecision/2) -(3*TamEspacioDecision/8) -
(TamEspacioDecision/4) -(TamEspacioDecision/8) 0 (TamEspacioDecision/8)
(TamEspacioDecision/4)];
    U_centros = [-(3*TamEspacioDecision/8) -(TamEspacioDecision/4) -
(TamEspacioDecision/8) 0 (TamEspacioDecision/8) (TamEspacioDecision/4)
3*TamEspacioDecision/8];
    U_finales = [-(TamEspacioDecision/4) -(TamEspacioDecision/8) 0
(TamEspacioDecision/8) (TamEspacioDecision/4) 3*TamEspacioDecision/8
TamEspacioDecision/2];

    %FUZZIFICANDO
    %Evaluando Pertenencia de errores de angulo
    EAngulo_BN =
FuncionPertenenciaTriangular(Kp_theta*error_angulo,e_inicios_theta(1),e_centros_theta(1),e_finales_theta(1),2);
    EAngulo_MN =
FuncionPertenenciaTriangular(Kp_theta*error_angulo,e_inicios_theta(2),e_centros_theta(2),e_finales_theta(2),1);
    EAngulo_SN =
FuncionPertenenciaTriangular(Kp_theta*error_angulo,e_inicios_theta(3),e_centros_theta(3),e_finales_theta(3),1);
    EAngulo_Z =
FuncionPertenenciaTriangular(Kp_theta*error_angulo,e_inicios_theta(4),e_centros_theta(4),e_finales_theta(4),1);
    EAngulo_SP =
FuncionPertenenciaTriangular(Kp_theta*error_angulo,e_inicios_theta(5),e_centros_theta(5),e_finales_theta(5),1);
    EAngulo_MP =
FuncionPertenenciaTriangular(Kp_theta*error_angulo,e_inicios_theta(6),e_centros_theta(6),e_finales_theta(6),1);
    EAngulo_BP =
FuncionPertenenciaTriangular(Kp_theta*error_angulo,e_inicios_theta(7),e_centros_theta(7),e_finales_theta(7),3);
    EAngulo =
[ones(7,1)*EAngulo_BN,ones(7,1)*EAngulo_MN,ones(7,1)*EAngulo_SN,ones(7,1)*EAngulo_Z,ones(7,1)*EAngulo_SP,ones(7,1)*EAngulo_MP,ones(7,1)*EAngulo_BP];

```

```

%Evaluando Pertenencia de derivada del error de angulo
dEAngulo_BN = FuncionPertenenciaTriangular(Td_theta*(error_angulo-
error_angulo_ant)/TControl,de_inicios_theta(1),de_centros_theta(1),de_finales_theta(1),2);
dEAngulo_MN = FuncionPertenenciaTriangular(Td_theta*(error_angulo-
error_angulo_ant)/TControl,de_inicios_theta(2),de_centros_theta(2),de_finales_theta(2),1);
dEAngulo_SN = FuncionPertenenciaTriangular(Td_theta*(error_angulo-
error_angulo_ant)/TControl,de_inicios_theta(3),de_centros_theta(3),de_finales_theta(3),1);
dEAngulo_Z = FuncionPertenenciaTriangular(Td_theta*(error_angulo-
error_angulo_ant)/TControl,de_inicios_theta(4),de_centros_theta(4),de_finales_theta(4),1);
dEAngulo_SP = FuncionPertenenciaTriangular(Td_theta*(error_angulo-
error_angulo_ant)/TControl,de_inicios_theta(5),de_centros_theta(5),de_finales_theta(5),1);
dEAngulo_MP = FuncionPertenenciaTriangular(Td_theta*(error_angulo-
error_angulo_ant)/TControl,de_inicios_theta(6),de_centros_theta(6),de_finales_theta(6),1);
dEAngulo_BP = FuncionPertenenciaTriangular(Td_theta*(error_angulo-
error_angulo_ant)/TControl,de_inicios_theta(7),de_centros_theta(7),de_finales_theta(7),3);
dEAngulo = [ones(1,7)*dEAngulo_BN;
ones(1,7)*dEAngulo_MN;ones(1,7)*dEAngulo_SN;ones(1,7)*dEAngulo_Z;ones(1,7)*dEAngulo_SP;
ones(1,7)*dEAngulo_MP;ones(1,7)*dEAngulo_BP];

```

```

%Evaluando Pertenencia de errores de posicion
EPosicion_BN =
FuncionPertenenciaTriangular(Kp_posicion*error_posicion,e_inicios_posicion(1),e_centros_posicio
n(1),e_finales_posicion(1),2);
EPosicion_MN =
FuncionPertenenciaTriangular(Kp_posicion*error_posicion,e_inicios_posicion(2),e_centros_posicio
n(2),e_finales_posicion(2),1);
EPosicion_SN =
FuncionPertenenciaTriangular(Kp_posicion*error_posicion,e_inicios_posicion(3),e_centros_posicio
n(3),e_finales_posicion(3),1);
EPosicion_Z =
FuncionPertenenciaTriangular(Kp_posicion*error_posicion,e_inicios_posicion(4),e_centros_posicio
n(4),e_finales_posicion(4),1);
EPosicion_SP =
FuncionPertenenciaTriangular(Kp_posicion*error_posicion,e_inicios_posicion(5),e_centros_posicio
n(5),e_finales_posicion(5),1);
EPosicion_MP =
FuncionPertenenciaTriangular(Kp_posicion*error_posicion,e_inicios_posicion(6),e_centros_posicio
n(6),e_finales_posicion(6),1);
EPosicion_BP =
FuncionPertenenciaTriangular(Kp_posicion*error_posicion,e_inicios_posicion(7),e_centros_posicio
n(7),e_finales_posicion(7),3);
EPosicion =
[ones(7,1)*EPosicion_BN,ones(7,1)*EPosicion_MN,ones(7,1)*EPosicion_SN,ones(7,1)*EPosicion_
Z,ones(7,1)*EPosicion_SP,ones(7,1)*EPosicion_MP,ones(7,1)*EPosicion_BP];

```

```

%Evaluando Pertenencia de derivada del error de posicion
dEPosicion_BN = FuncionPertenenciaTriangular(Td_posicion*(error_posicion-
error_posicion_ant)/TControl,de_inicios_posicion(1),de_centros_posicion(1),de_finales_posicion(1),
2);
dEPosicion_MN = FuncionPertenenciaTriangular(Td_posicion*(error_posicion-
error_posicion_ant)/TControl,de_inicios_posicion(2),de_centros_posicion(2),de_finales_posicion(2),
1);

```

```

dEPosicion_SN = FuncionPertenenciaTriangular(Td_posicion*(error_posicion-
error_posicion_ant)/TControl,de_inicios_posicion(3),de_centros_posicion(3),de_finales_posicion(3),
1);
dEPosicion_Z = FuncionPertenenciaTriangular(Td_posicion*(error_posicion-
error_posicion_ant)/TControl,de_inicios_posicion(4),de_centros_posicion(4),de_finales_posicion(4),
1);
dEPosicion_SP = FuncionPertenenciaTriangular(Td_posicion*(error_posicion-
error_posicion_ant)/TControl,de_inicios_posicion(5),de_centros_posicion(5),de_finales_posicion(5),
1);
dEPosicion_MP = FuncionPertenenciaTriangular(Td_posicion*(error_posicion-
error_posicion_ant)/TControl,de_inicios_posicion(6),de_centros_posicion(6),de_finales_posicion(6),
1);
dEPosicion_BP = FuncionPertenenciaTriangular(Td_posicion*(error_posicion-
error_posicion_ant)/TControl,de_inicios_posicion(7),de_centros_posicion(7),de_finales_posicion(7),
3);
dEPosicion = [ones(1,7)*dEPosicion_BN;
ones(1,7)*dEPosicion_MN;ones(1,7)*dEPosicion_SN;ones(1,7)*dEPosicion_Z;ones(1,7)*dEPosicio
n_SP;ones(1,7)*dEPosicion_MP;ones(1,7)*dEPosicion_BP];

```

%EVALUANDO LAS REGLAS DIFUSAS

```

%Para el Angulo
% IF EAngulo_BN AND dEAngulo_BN THEN U_BN
% IF EAngulo_BN AND dEAngulo_MN THEN U_BN
% IF EAngulo_BN AND dEAngulo_SN THEN U_BN
% IF EAngulo_BN AND dEAngulo_Z THEN U_BN
% IF EAngulo_BN AND dEAngulo_SP THEN U_MN
% IF EAngulo_BN AND dEAngulo_MP THEN U_SN
% IF EAngulo_BN AND dEAngulo_BP THEN U_SN
% IF EAngulo_MN AND dEAngulo_BN THEN U_BN
% IF EAngulo_MN AND dEAngulo_MN THEN U_BN
% IF EAngulo_MN AND dEAngulo_SN THEN U_BN
% IF EAngulo_MN AND dEAngulo_Z THEN U_MN
% IF EAngulo_MN AND dEAngulo_SP THEN U_SN
% IF EAngulo_MN AND dEAngulo_MP THEN U_SN
% IF EAngulo_MN AND dEAngulo_BP THEN U_SN
% IF EAngulo_SN AND dEAngulo_BN THEN U_MN
% IF EAngulo_SN AND dEAngulo_MN THEN U_MN
% IF EAngulo_SN AND dEAngulo_SN THEN U_SN
% IF EAngulo_SN AND dEAngulo_Z THEN U_SN
% IF EAngulo_SN AND dEAngulo_SP THEN U_SN
% IF EAngulo_SN AND dEAngulo_MP THEN U_SN
% IF EAngulo_SN AND dEAngulo_BP THEN U_SN
% IF EAngulo_Z AND dEAngulo_BN THEN U_SN
% IF EAngulo_Z AND dEAngulo_MN THEN U_SN
% IF EAngulo_Z AND dEAngulo_SN THEN U_SN
% IF EAngulo_Z AND dEAngulo_Z THEN U_Z
% IF EAngulo_Z AND dEAngulo_SP THEN U_SP
% IF EAngulo_Z AND dEAngulo_MP THEN U_SP
% IF EAngulo_Z AND dEAngulo_BP THEN U_SP
% IF EAngulo_SP AND dEAngulo_BN THEN U_SP
% IF EAngulo_SP AND dEAngulo_MN THEN U_SP
% IF EAngulo_SP AND dEAngulo_SN THEN U_SP
% IF EAngulo_SP AND dEAngulo_Z THEN U_SP

```

```
% IF EAngulo_SP AND dEAngulo_SP THEN U_SP
% IF EAngulo_SP AND dEAngulo_MP THEN U_MP
% IF EAngulo_SP AND dEAngulo_BP THEN U_MP
% IF EAngulo_MP AND dEAngulo_BN THEN U_SP
% IF EAngulo_MP AND dEAngulo_MN THEN U_SP
% IF EAngulo_MP AND dEAngulo_SN THEN U_SP
% IF EAngulo_MP AND dEAngulo_Z THEN U_MP
% IF EAngulo_MP AND dEAngulo_SP THEN U_BP
% IF EAngulo_MP AND dEAngulo_MP THEN U_BP
% IF EAngulo_MP AND dEAngulo_BP THEN U_BP
% IF EAngulo_BP AND dEAngulo_BN THEN U_SP
% IF EAngulo_BP AND dEAngulo_MN THEN U_SP
% IF EAngulo_BP AND dEAngulo_SN THEN U_MP
% IF EAngulo_BP AND dEAngulo_Z THEN U_BP
% IF EAngulo_BP AND dEAngulo_SP THEN U_BP
% IF EAngulo_BP AND dEAngulo_MP THEN U_BP
% IF EAngulo_BP AND dEAngulo_BP THEN U_BP
U_theta = min(EAngulo,dEAngulo);
```

```
%Para la posicion
% IF EPosicion_BN AND dEPosicion_BN THEN U_BN
% IF EPosicion_BN AND dEPosicion_MN THEN U_BN
% IF EPosicion_BN AND dEPosicion_SN THEN U_BN
% IF EPosicion_BN AND dEPosicion_Z THEN U_BN
% IF EPosicion_BN AND dEPosicion_SP THEN U_MN
% IF EPosicion_BN AND dEPosicion_MP THEN U_SN
% IF EPosicion_BN AND dEPosicion_BP THEN U_SN
% IF EPosicion_MN AND dEPosicion_BN THEN U_BN
% IF EPosicion_MN AND dEPosicion_MN THEN U_BN
% IF EPosicion_MN AND dEPosicion_SN THEN U_BN
% IF EPosicion_MN AND dEPosicion_Z THEN U_MN
% IF EPosicion_MN AND dEPosicion_SP THEN U_SN
% IF EPosicion_MN AND dEPosicion_MP THEN U_SN
% IF EPosicion_MN AND dEPosicion_BP THEN U_SN
% IF EPosicion_SN AND dEPosicion_BN THEN U_MN
% IF EPosicion_SN AND dEPosicion_MN THEN U_MN
% IF EPosicion_SN AND dEPosicion_SN THEN U_SN
% IF EPosicion_SN AND dEPosicion_Z THEN U_SN
% IF EPosicion_SN AND dEPosicion_SP THEN U_SN
% IF EPosicion_SN AND dEPosicion_MP THEN U_SN
% IF EPosicion_SN AND dEPosicion_BP THEN U_SN
% IF EPosicion_Z AND dEPosicion_BN THEN U_SN
% IF EPosicion_Z AND dEPosicion_MN THEN U_SN
% IF EPosicion_Z AND dEPosicion_SN THEN U_SN
% IF EPosicion_Z AND dEPosicion_Z THEN U_Z
% IF EPosicion_Z AND dEPosicion_SP THEN U_SP
% IF EPosicion_Z AND dEPosicion_MP THEN U_SP
% IF EPosicion_Z AND dEPosicion_BP THEN U_SP
% IF EPosicion_SP AND dEPosicion_BN THEN U_SP
% IF EPosicion_SP AND dEPosicion_MN THEN U_SP
% IF EPosicion_SP AND dEPosicion_SN THEN U_SP
% IF EPosicion_SP AND dEPosicion_Z THEN U_SP
% IF EPosicion_SP AND dEPosicion_SP THEN U_SP
% IF EPosicion_SP AND dEPosicion_MP THEN U_MP
```

```
% IF EPosicion_SP AND dEPosicion_BP THEN U_MP
% IF EPosicion_MP AND dEPosicion_BN THEN U_SP
% IF EPosicion_MP AND dEPosicion_MN THEN U_SP
% IF EPosicion_MP AND dEPosicion_SN THEN U_SP
% IF EPosicion_MP AND dEPosicion_Z THEN U_MP
% IF EPosicion_MP AND dEPosicion_SP THEN U_BP
% IF EPosicion_MP AND dEPosicion_MP THEN U_BP
% IF EPosicion_MP AND dEPosicion_BP THEN U_BP
% IF EPosicion_BP AND dEPosicion_BN THEN U_SP
% IF EPosicion_BP AND dEPosicion_MN THEN U_SP
% IF EPosicion_BP AND dEPosicion_SN THEN U_MP
% IF EPosicion_BP AND dEPosicion_Z THEN U_BP
% IF EPosicion_BP AND dEPosicion_SP THEN U_BP
% IF EPosicion_BP AND dEPosicion_MP THEN U_BP
% IF EPosicion_BP AND dEPosicion_BP THEN U_BP
U_posicion = min(EPosicion,dEPosicion);
```

%DEFUSIFICANDO

%Calculando la fuerza de las reglas del angulo usando RSS(Root Sum Squared)

U_BN_theta =

$\text{sqrt}(U_theta(1,1)^2+U_theta(1,2)^2+U_theta(2,1)^2+U_theta(2,2)^2+U_theta(3,1)^2+U_theta(3,2)^2+U_theta(4,1)^2);$

U_MN_theta = $\text{sqrt}(U_theta(1,3)^2+U_theta(2,3)^2+U_theta(4,2)^2+U_theta(5,1)^2);$

U_SN_theta =

$\text{sqrt}(U_theta(1,4)^2+U_theta(2,4)^2+U_theta(3,3)^2+U_theta(3,4)^2+U_theta(4,3)^2+U_theta(5,2)^2+U_theta(5,3)^2+U_theta(6,1)^2+U_theta(6,2)^2+U_theta(6,3)^2+U_theta(7,1)^2+U_theta(7,2)^2+U_theta(7,3)^2);$

U_Z_theta = $\text{sqrt}(U_theta(4,4)^2);$

U_SP_theta =

$\text{sqrt}(U_theta(1,5)^2+U_theta(1,6)^2+U_theta(1,7)^2+U_theta(2,5)^2+U_theta(2,6)^2+U_theta(2,7)^2+U_theta(3,5)^2+U_theta(3,6)^2+U_theta(4,5)^2+U_theta(5,4)^2+U_theta(5,5)^2+U_theta(6,4)^2+U_theta(7,4)^2);$

U_MP_theta = $\text{sqrt}(U_theta(3,7)^2+U_theta(4,6)^2+U_theta(6,5)^2+U_theta(7,5)^2);$

U_BP_theta =

$\text{sqrt}(U_theta(4,7)^2+U_theta(5,6)^2+U_theta(5,7)^2+U_theta(6,6)^2+U_theta(6,7)^2+U_theta(7,6)^2+U_theta(7,7)^2);$

%Calculando la fuerza de las reglas de la posicion usando RSS(Root Sum Squared)

U_BN_posicion =

$\text{sqrt}(U_posicion(1,1)^2+U_posicion(1,2)^2+U_posicion(2,1)^2+U_posicion(2,2)^2+U_posicion(3,1)^2+U_posicion(3,2)^2+U_posicion(4,1)^2);$

U_MN_posicion =

$\text{sqrt}(U_posicion(1,3)^2+U_posicion(2,3)^2+U_posicion(4,2)^2+U_posicion(5,1)^2);$

U_SN_posicion =

$\text{sqrt}(U_posicion(1,4)^2+U_posicion(2,4)^2+U_posicion(3,3)^2+U_posicion(3,4)^2+U_posicion(4,3)^2+U_posicion(5,2)^2+U_posicion(5,3)^2+U_posicion(6,1)^2+U_posicion(6,2)^2+U_posicion(6,3)^2+U_posicion(7,1)^2+U_posicion(7,2)^2+U_posicion(7,3)^2);$

U_Z_posicion = $\text{sqrt}(U_posicion(4,4)^2);$

U_SP_posicion =

$\text{sqrt}(U_posicion(1,5)^2+U_posicion(1,6)^2+U_posicion(1,7)^2+U_posicion(2,5)^2+U_posicion(2,6)^2+U_posicion(2,7)^2+U_posicion(3,5)^2+U_posicion(3,6)^2+U_posicion(4,5)^2+U_posicion(5,4)^2+U_posicion(5,5)^2+U_posicion(6,4)^2+U_posicion(7,4)^2);$

U_MP_posicion =

$\text{sqrt}(U_posicion(3,7)^2+U_posicion(4,6)^2+U_posicion(6,5)^2+U_posicion(7,5)^2);$

```

    U_BP_posicion =
    sqrt(U_posicion(4,7)^2+U_posicion(5,6)^2+U_posicion(5,7)^2+U_posicion(6,6)^2+U_posicion(6,7)^
    2+U_posicion(7,6)^2+U_posicion(7,7)^2);

    %Usando RSS-Centroide para calcular la variacion en la salida en base a theta
    U_fuerzas_theta = [U_BN_theta U_MN_theta U_SN_theta U_Z_theta U_SP_theta
    U_MP_theta U_BP_theta];
    SenalControl_theta = sum(U_centros.*U_fuerzas_theta)/sum(U_fuerzas_theta);

    %Usando RSS-Centroide para calcular la variacion en la salida en base a la posicion
    U_fuerzas_posicion = [U_BN_posicion U_MN_posicion U_SN_posicion U_Z_posicion
    U_SP_posicion U_MP_posicion U_BP_posicion];
    SenalControl_posicion = sum(U_centros.*U_fuerzas_posicion)/sum(U_fuerzas_posicion);

    SenalControl = Ti*(SenalControl+(SenalControl_theta + SenalControl_posicion)/2)/2;

    Xs = [Xs Estado(3)];
    Thetas = [Thetas Estado(1)];
    Uts = [Uts SenalControl];
    Tiempos = [Tiempos tiempo];

    CatetoOpuesto = sin(Estado(1))*Longitud;
    CatetoAdyacente = cos(Estado(1))*Longitud;
    PosCarga = [Estado(3)-CatetoOpuesto,Longitud*0.1456-CatetoAdyacente];
    PGruaA = findobj('Tag','PGruaA');
    axes(PGruaA);
    plot([-TamanhoViga*0.7 TamanhoViga*0.7],[0 0],'g','linewidth',2) % Linea Representando
la viga
    hold on
    plot([Estado(3) PosCarga(1)],[Longitud*0.1456 PosCarga(2)],'r','linewidth',4) % Linea
representando el cable de la carga
    hold on
    plot([Estado(3)+TamanhoViga/10 Estado(3)-TamanhoViga/10 Estado(3)-TamanhoViga/10
Estado(3)+TamanhoViga/10 Estado(3)+TamanhoViga/10],[Longitud*0.1456 Longitud*0.1456 0 0
Longitud*0.1456],'b','linewidth',5) % Representacion del carro
    hold on
    plot(PosCarga(1),PosCarga(2),'*m','linewidth',20) %Representación de la Carga
    hold on
    plot(Estado(3)+TamanhoViga/20,0,'*b','linewidth',13) % Rueda delantera del Carro
    hold on
    plot(Estado(3)-TamanhoViga/20,0,'*b','linewidth',13) % Rueda Trasera del Carro
    hold off
    set(PGruaA,'XLim', [-TamanhoViga*0.7 TamanhoViga*0.7]);
    set(PGruaA,'YLim', [-Longitud*1.5 Longitud*1.5]);
    set(PGruaA,'Tag','PGruaA');
    grid on
    set(PGruaA,'Title',text('String','Animación del Puente Grua'));
    set(PGruaA,'XLabel',text('String','Posición X (m)'));
    set(PGruaA,'YLabel',text('String','Posición Y (m)'));
    drawnow;
end
end

U_t = findobj('Tag','U_t');

```

```

axes(U_t);
plot(Tiempos,Uts);
set(U_t,'Tag','U_t');
set(U_t,'Title',text('String','Magnitud de U(t) en el Tiempo'));
set(U_t,'XLabel',text('String','Tiempo (s)'));
set(U_t,'YLabel',text('String','U(t)'));

posCarro = findobj('Tag','posCarro');
axes(posCarro);
plot(Tiempos,Xs);
set(posCarro,'Tag','posCarro');
set(posCarro,'Title',text('String','Posición del Carro en el Tiempo'));
set(posCarro,'XLabel',text('String','Tiempo (s)'));
set(posCarro,'YLabel',text('String','Posición del Carro(m)'));

posAngulo = findobj('Tag','posAngulo');
axes(posAngulo);
plot(Tiempos,Thetas);
set(posAngulo,'Tag','posAngulo');
set(posAngulo,'Title',text('String','Posición del Ángulo en el Tiempo'));
set(posAngulo,'XLabel',text('String','Tiempo (s)'));
set(posAngulo,'YLabel',text('String','Ángulo (rad)'));

end
Tipo
elseif comando == 2 % Boton Reiniciar
%Devolviendo los valores originales a los parámetros del simulador
GravedadEdit = findobj('Tag','GravedadEdit');
set(GravedadEdit,'String','9.81')
LongitudEdit = findobj('Tag','LongitudEdit');
set(LongitudEdit,'String','7')
MasaCarroEdit = findobj('Tag','MasaCarroEdit');
set(MasaCarroEdit,'String','2000')
MasaCargaEdit = findobj('Tag','MasaCargaEdit');
set(MasaCargaEdit,'String','4000')
MomentoInerciaEdit = findobj('Tag','MomentoInerciaEdit');
set(MomentoInerciaEdit,'String','16680')
FriccionCarroEdit = findobj('Tag','FriccionCarroEdit');
set(FriccionCarroEdit,'String','0.006')
FriccionRotacionalEdit = findobj('Tag','FriccionRotacionalEdit');
set(FriccionRotacionalEdit,'String','1')
TamanhoVigaEdit = findobj('Tag','TamanhoVigaEdit');
set(TamanhoVigaEdit,'String','50')
PosInicialEdit = findobj('Tag','PosInicialEdit');
set(PosInicialEdit,'String','-30')
PosFinalEdit = findobj('Tag','PosFinalEdit');
set(PosFinalEdit,'String','0')
TSimulacionEdit = findobj('Tag','TSimulacionEdit');
set(TSimulacionEdit,'String','40')
elseif comando == 4 % PID Radio
PControlCheck = findobj('Tag','PControlCheck');
set(PControlCheck,'Value',0);
set(PControlCheck,'Enable','on');
PControlEditTheta = findobj('Tag','PControlEditTheta');

```

```

set(PControlEditTheta,'String','1');
set(PControlEditTheta,'Enable','off');
PControlEditPosicion = findobj('Tag','PControlEditPosicion');
set(PControlEditPosicion,'String','1');
set(PControlEditPosicion,'Enable','off');
DControlCheck = findobj('Tag','DControlCheck');
set(DControlCheck,'Enable','on');
set(DControlCheck,'Value',0);
DControlEditTheta = findobj('Tag','DControlEditTheta');
set(DControlEditTheta,'String','1');
set(DControlEditTheta,'Enable','off');
DControlEditPosicion = findobj('Tag','DControlEditPosicion');
set(DControlEditPosicion,'String','1');
set(DControlEditPosicion,'Enable','off');
IControlCheck = findobj('Tag','IControlCheck');
set(IControlCheck,'Enable','on');
set(IControlCheck,'Value',0);
IControlEditTheta = findobj('Tag','IControlEditTheta');
set(IControlEditTheta,'String','1');
set(IControlEditTheta,'Enable','off');
IControlEditPosicion = findobj('Tag','IControlEditPosicion');
set(IControlEditPosicion,'String','1');
set(IControlEditPosicion,'Enable','off');
elseif comando == 5 % Fuzzy Radio
PControlCheck = findobj('Tag','PControlCheck');
set(PControlCheck,'Enable','off');
set(PControlCheck,'Value',1);
PControlEditTheta = findobj('Tag','PControlEditTheta');
set(PControlEditTheta,'String','1');
set(PControlEditTheta,'Enable','on');
PControlEditPosicion = findobj('Tag','PControlEditPosicion');
set(PControlEditPosicion,'String','1');
set(PControlEditPosicion,'Enable','on');
DControlCheck = findobj('Tag','DControlCheck');
set(DControlCheck,'Enable','off');
set(DControlCheck,'Value',1);
DControlEditTheta = findobj('Tag','DControlEditTheta');
set(DControlEditTheta,'String','1');
set(DControlEditTheta,'Enable','on');
DControlEditPosicion = findobj('Tag','DControlEditPosicion');
set(DControlEditPosicion,'String','1');
set(DControlEditPosicion,'Enable','on');
IControlCheck = findobj('Tag','IControlCheck');
set(IControlCheck,'Enable','off');
set(IControlCheck,'Value',1);
IControlEditTheta = findobj('Tag','IControlEditTheta');
set(IControlEditTheta,'String','1');
set(IControlEditTheta,'Enable','on');
IControlEditPosicion = findobj('Tag','IControlEditPosicion');
set(IControlEditPosicion,'String','1');
set(IControlEditPosicion,'Enable','off');
elseif comando == 6 % P Check
PControlCheck = findobj('Tag','PControlCheck');
PControlCheckValue = get(PControlCheck,'Value');

```

```

PControlEditTheta = findobj('Tag','PControlEditTheta');
PControlEditPosicion = findobj('Tag','PControlEditPosicion');
if PControlCheckValue == 1
    set(PControlEditTheta,'String','1');
    set(PControlEditTheta,'Enable','on');
    set(PControlEditPosicion,'String','1');
    set(PControlEditPosicion,'Enable','on');
else
    set(PControlEditTheta,'String','1');
    set(PControlEditTheta,'Enable','off');
    set(PControlEditPosicion,'String','1');
    set(PControlEditPosicion,'Enable','off');
end
elseif comando == 7 % D Check
DControlCheck = findobj('Tag','DControlCheck');
DControlCheckValue = get(DControlCheck,'Value');
DControlEditTheta = findobj('Tag','DControlEditTheta');
DControlEditPosicion = findobj('Tag','DControlEditPosicion');
if DControlCheckValue == 1
    set(DControlEditTheta,'String','1');
    set(DControlEditTheta,'Enable','on');
    set(DControlEditPosicion,'String','1');
    set(DControlEditPosicion,'Enable','on');
else
    set(DControlEditTheta,'String','1');
    set(DControlEditTheta,'Enable','off');
    set(DControlEditPosicion,'String','1');
    set(DControlEditPosicion,'Enable','off');
end
elseif comando == 8 % I Check
IControlCheck = findobj('Tag','IControlCheck');
IControlCheckValue = get(IControlCheck,'Value');
IControlEditTheta = findobj('Tag','IControlEditTheta');
IControlEditPosicion = findobj('Tag','IControlEditPosicion');
if IControlCheckValue == 1
    set(IControlEditTheta,'String','0.1');
    set(IControlEditTheta,'Enable','on');
    set(IControlEditPosicion,'String','0.1');
    set(IControlEditPosicion,'Enable','on');
else
    set(IControlEditTheta,'String','0.1');
    set(IControlEditTheta,'Enable','off');
    set(IControlEditPosicion,'String','0.1');
    set(IControlEditPosicion,'Enable','off');
end
end
end

```

ANEXO 2

```

function PuenteGruaFinal()

Pantalla = figure('Color',[1 1 1], ...
    'Tag','PuenteGrua', 'Name','Simulador del Control de una Grua Puente');
whitebg
whitebg

FrameBotones_a = uicontrol('Parent',Pantalla, ...
    'Units','normalized', ...
    'BackgroundColor',[0 0 0], ...
    'Position',[0.0126 0.7393 0.1733 0.2044], ...
    'Style','frame', ...
    'Tag','Nbarframe');

FrameBotones_b = uicontrol('Parent',Pantalla, ...
    'Units','normalized', ...
    'BackgroundColor',[1 1 1], ...
    'Position',[0.0215 0.7496 0.1533 0.1853], ...
    'Style','frame', ...
    'Tag','Kframe');

Ejecutar = uicontrol('Parent',Pantalla, ...
    'Units','normalized', ...
    'BackgroundColor',[0 1 0], ...
    'Callback','ControlPGruaFinal(1)', ...
    'Position',[0.04037 0.85037 0.111111 0.0459259], ...
    'String','Ejecutar', ...
    'Tag','Runbutton');

Reiniciar = uicontrol('Parent',Pantalla, ...
    'Units','normalized', ...
    'BackgroundColor',[1 1 0], ...
    'Callback','ControlPGruaFinal(2)', ...
    'Position',[0.04037 0.78037 0.111111 0.0459259], ...
    'String','Reiniciar', ...
    'Tag','clear');

FrameTipoControl_a = uicontrol('Parent',Pantalla, ...
    'Units','normalized', ...
    'BackgroundColor',[0 0 0], ...
    'Position',[0.0126 0.5393 0.1733 0.1644], ...
    'Style','frame', ...
    'Tag','Nbarframe');

FrameTipoControl_b = uicontrol('Parent',Pantalla, ...
    'Units','normalized', ...
    'BackgroundColor',[1 1 1], ...
    'Position',[0.0215 0.5496 0.1533 0.1453], ...
    'Style','frame', ...
    'Tag','Kframe');
    
```

```
TextTipoControl = uicontrol('Parent',Pantalla, ...
    'Units','normalized', ...
    'BackgroundColor',[1 1 1], ...
    'Position',[0.023 0.645926 0.15 0.0311111], ...
    'String','Tipo de Control', ...
    'Style','text', ...
    'Tag','TipoControltitle');
```

```
BGTipoControl = uibuttongroup('Parent',Pantalla, ...
    'visible','on', ...
    'BorderType','none', ...
    'BackgroundColor',[1 1 1], ...
    'SelectionChangeFcn', @selcbk, ...
    'Position',[0.023 0.555926 0.15 0.1]);
```

```
PIDRadio = uicontrol('Parent',BGTipoControl, ...
    'Units','normalized', ...
    'BackgroundColor',[1 1 1], ...
    'Position',[0.025 0.570926 0.4 0.4], ...
    'String','PID', ...
    'Style','Radio', ...
    'Tag','PIDRadio');
```

```
FuzzyRadio = uicontrol('Parent',BGTipoControl, ...
    'Units','normalized', ...
    'BackgroundColor',[1 1 1], ...
    'Position',[0.445 0.570926 0.4 0.4], ...
    'String','Difuso', ...
    'Style','Radio', ...
    'Tag','FuzzyRadio');
```

```
PControlCheck = uicontrol('Parent',Pantalla, ...
    'Units','normalized', ...
    'BackgroundColor',[1 1 1], ...
    'Position',[0.025741 0.575926 0.02 0.04], ...
    'String','P', ...
    'Style','checkbox', ...
    'Callback','ControlPGruaFinal(6)', ...
    'Tag','PControlCheck', ...
    'Value',0);
```

```
PControlEditTheta = uicontrol('Parent',Pantalla, ...
    'Units','normalized', ...
    'BackgroundColor',[1 1 1], ...
    'Position',[0.045148 0.587 0.028 0.03], ...
    'String','1', ...
    'Style','edit', ...
    'Enable','off', ...
    'Tag','PControlEditTheta', ...
    'Value',1);
```

```
PControlEditPosicion = uicontrol('Parent',Pantalla, ...
    'Units','normalized', ...
```

```
'BackgroundColor',[1 1 1], ...
'Position',[0.045148 0.557 0.028 0.03], ...
'String','1', ...
'Style','edit', ...
'Enable','off', ...
'Tag','PControlEditPosicion', ...
'Value',1);
```

```
DControlCheck = uicontrol('Parent',Pantalla, ...
'Units','normalized', ...
'BackgroundColor',[1 1 1], ...
'Position',[0.075741 0.575926 0.02 0.04], ...
'String','D', ...
'Style','checkbox', ...
'Callback','ControlPGruaFinal(7)', ...
'Tag','DControlCheck', ...
'Value',0);
```

```
DControlEditTheta = uicontrol('Parent',Pantalla, ...
'Units','normalized', ...
'BackgroundColor',[1 1 1], ...
'Position',[0.095148 0.587 0.028 0.03], ...
'String','1', ...
'Style','edit', ...
'Enable','off', ...
'Tag','DControlEditTheta', ...
'Value',1);
```

```
DControlEditPosicion = uicontrol('Parent',Pantalla, ...
'Units','normalized', ...
'BackgroundColor',[1 1 1], ...
'Position',[0.095148 0.557 0.028 0.03], ...
'String','1', ...
'Style','edit', ...
'Enable','off', ...
'Tag','DControlEditPosicion', ...
'Value',1);
```

```
IControlCheck = uicontrol('Parent',Pantalla, ...
'Units','normalized', ...
'BackgroundColor',[1 1 1], ...
'Position',[0.125741 0.575926 0.02 0.04], ...
'String','I', ...
'Style','checkbox', ...
'Callback','ControlPGruaFinal(8)', ...
'Tag','IControlCheck', ...
'Value',0);
```

```
IControlEditTheta = uicontrol('Parent',Pantalla, ...
'Units','normalized', ...
'BackgroundColor',[1 1 1], ...
'Position',[0.145148 0.587 0.028 0.03], ...
'String','1', ...
'Style','edit', ...
```

```
'Enable','off', ...
'Tag','IControlEditTheta', ...
'Value',1);

IControlEditPosicion = uicontrol('Parent',Pantalla, ...
'Units','normalized', ...
'BackgroundColor',[1 1 1], ...
'Position',[0.145148 0.557 0.028 0.03], ...
'String','1', ...
'Style','edit', ...
'Enable','off', ...
'Tag','IControlEditPosicion', ...
'Value',1);

FrameParametros_a = uicontrol('Parent',Pantalla, ...
'Units','normalized', ...
'BackgroundColor',[0 0 0], ...
'Position',[0.0126 0.0593 0.1733 0.4444], ...
'Style','frame', ...
'Tag','Nbarframe');

FrameParametros_b = uicontrol('Parent',Pantalla, ...
'Units','normalized', ...
'BackgroundColor',[1 1 1], ...
'Position',[0.0215 0.0696 0.1533 0.4253], ...
'Style','frame', ...
'Tag','Kframe');

TextParametros = uicontrol('Parent',Pantalla, ...
'Units','normalized', ...
'BackgroundColor',[1 1 1], ...
'Position',[0.023 0.445926 0.15 0.0311111], ...
'String','Parámetros del Simulador', ...
'Style','text', ...
'Tag','TextParametros');

GravedadText = uicontrol('Parent',Pantalla, ...
'Units','normalized', ...
'BackgroundColor',[1 1 1], ...
'Position',[0.03037 0.402 0.05 0.03], ...
'HorizontalAlignment','left', ...
'String','Gravedad (g)', ...
'Style','text', ...
'Tag','GravedadText');

GravedadEdit = uicontrol('Parent',Pantalla, ...
'Units','normalized', ...
'BackgroundColor',[1 1 1], ...
'Position',[0.118148 0.405926 0.05 0.03], ...
'String','9.81', ...
'Style','edit', ...
'Tag','GravedadEdit', ...
'Value',9.81);
```

```
LongitudText = uicontrol('Parent',Pantalla, ...
    'Units','normalized', ...
    'BackgroundColor',[1 1 1], ...
    'Position',[0.03037 0.372 0.10 0.03], ...
    'HorizontalAlignment','left', ...
    'String','Longitud del Cable (l)', ...
    'Style','text', ...
    'Tag','LongitudText');
```

```
LongitudEdit = uicontrol('Parent',Pantalla, ...
    'Units','normalized', ...
    'BackgroundColor',[1 1 1], ...
    'Position',[0.118148 0.375926 0.05 0.03], ...
    'String','7', ...
    'Style','edit', ...
    'Tag','LongitudEdit', ...
    'Value',7);
```

```
MasaCarroText = uicontrol('Parent',Pantalla, ...
    'Units','normalized', ...
    'BackgroundColor',[1 1 1], ...
    'Position',[0.03037 0.342 0.10 0.03], ...
    'HorizontalAlignment','left', ...
    'String','Masa del Carro (Mc)', ...
    'Style','text', ...
    'Tag','MasaCarroText');
```

```
MasaCarroEdit = uicontrol('Parent',Pantalla, ...
    'Units','normalized', ...
    'BackgroundColor',[1 1 1], ...
    'Position',[0.118148 0.345926 0.05 0.03], ...
    'String','2000', ...
    'Style','edit', ...
    'Tag','MasaCarroEdit', ...
    'Value',2000);
```

```
MasaCargaText = uicontrol('Parent',Pantalla, ...
    'Units','normalized', ...
    'BackgroundColor',[1 1 1], ...
    'Position',[0.03037 0.312 0.10 0.03], ...
    'HorizontalAlignment','left', ...
    'String','Masa de la Carga (ml)', ...
    'Style','text', ...
    'Tag','MasaCargaText');
```

```
MasaCargaEdit = uicontrol('Parent',Pantalla, ...
    'Units','normalized', ...
    'BackgroundColor',[1 1 1], ...
    'Position',[0.118148 0.315926 0.05 0.03], ...
    'String','4000', ...
    'Style','edit', ...
    'Tag','MasaCargaEdit', ...
    'Value',4000);
```

```
MomentolNerciaText = uicontrol('Parent',Pantalla, ...
    'Units','normalized', ...
    'BackgroundColor',[1 1 1], ...
    'Position',[0.03037 0.282 0.10 0.03], ...
    'HorizontalAlignment','left', ...
    'String','Momento de Inercia (J)', ...
    'Style','text', ...
    'Tag','MomentolNerciaText');
```

```
MomentolNerciaEdit = uicontrol('Parent',Pantalla, ...
    'Units','normalized', ...
    'BackgroundColor',[1 1 1], ...
    'Position',[0.118148 0.285926 0.05 0.03], ...
    'String','16680', ...
    'Style','edit', ...
    'Tag','MomentolNerciaEdit', ...
    'Value',16680);
```

```
FriccionCarroText = uicontrol('Parent',Pantalla, ...
    'Units','normalized', ...
    'BackgroundColor',[1 1 1], ...
    'Position',[0.03037 0.252 0.10 0.03], ...
    'HorizontalAlignment','left', ...
    'String','Fricción del Carro (fc)', ...
    'Style','text', ...
    'Tag','FriccionCarroText');
```

```
FriccionCarroEdit = uicontrol('Parent',Pantalla, ...
    'Units','normalized', ...
    'BackgroundColor',[1 1 1], ...
    'Position',[0.118148 0.255926 0.05 0.03], ...
    'String','0.006', ...
    'Style','edit', ...
    'Tag','FriccionCarroEdit', ...
    'Value',0.006);
```

```
FriccionRotacionalText = uicontrol('Parent',Pantalla, ...
    'Units','normalized', ...
    'BackgroundColor',[1 1 1], ...
    'Position',[0.03037 0.222 0.10 0.03], ...
    'HorizontalAlignment','left', ...
    'String','Fricción Rotacional (fp)', ...
    'Style','text', ...
    'Tag','FriccionRotacionalText');
```

```
FriccionRotacionalEdit = uicontrol('Parent',Pantalla, ...
    'Units','normalized', ...
    'BackgroundColor',[1 1 1], ...
    'Position',[0.118148 0.225926 0.05 0.03], ...
    'String','1', ...
    'Style','edit', ...
    'Tag','FriccionRotacionalEdit', ...
    'Value',1);
```

```
TamanhoVigaText = uicontrol('Parent',Pantalla, ...
    'Units','normalized', ...
    'BackgroundColor',[1 1 1], ...
    'Position',[0.03037 0.192 0.10 0.03], ...
    'HorizontalAlignment','left', ...
    'String','Tamaño de la Viga', ...
    'Style','text', ...
    'Tag','TamanhoVigaText');
```

```
TamanhoVigaEdit = uicontrol('Parent',Pantalla, ...
    'Units','normalized', ...
    'BackgroundColor',[1 1 1], ...
    'Position',[0.118148 0.195926 0.05 0.03], ...
    'String','50', ...
    'Style','edit', ...
    'Tag','TamanhoVigaEdit', ...
    'Value',50);
```

```
PosInicialText = uicontrol('Parent',Pantalla, ...
    'Units','normalized', ...
    'BackgroundColor',[1 1 1], ...
    'Position',[0.03037 0.142 0.10 0.03], ...
    'HorizontalAlignment','left', ...
    'String','Posición Inicial del Carro', ...
    'Style','text', ...
    'Tag','PosInicialText');
```

```
PosInicialEdit = uicontrol('Parent',Pantalla, ...
    'Units','normalized', ...
    'BackgroundColor',[1 1 1], ...
    'Position',[0.118148 0.145926 0.05 0.03], ...
    'String','-30', ...
    'Style','edit', ...
    'Tag','PosInicialEdit', ...
    'Value',-30);
```

```
PosFinalText = uicontrol('Parent',Pantalla, ...
    'Units','normalized', ...
    'BackgroundColor',[1 1 1], ...
    'Position',[0.03037 0.112 0.10 0.03], ...
    'HorizontalAlignment','left', ...
    'String','Posición Final del Carro', ...
    'Style','text', ...
    'Tag','PosFinalText');
```

```
PosFinalEdit = uicontrol('Parent',Pantalla, ...
    'Units','normalized', ...
    'BackgroundColor',[1 1 1], ...
    'Position',[0.118148 0.115926 0.05 0.03], ...
    'String','0', ...
    'Style','edit', ...
    'Tag','PosFinalEdit', ...
    'Value',0);
```

```

TSimulacionText = uicontrol('Parent',Pantalla, ...
    'Units','normalized', ...
    'BackgroundColor',[1 1 1], ...
    'Position',[0.03037 0.082 0.10 0.03], ...
    'HorizontalAlignment','left', ...
    'String','Tiempo de Simulación', ...
    'Style','text', ...
    'Tag','TSimulacionText');

TSimulacionEdit = uicontrol('Parent',Pantalla, ...
    'Units','normalized', ...
    'BackgroundColor',[1 1 1], ...
    'Position',[0.118148 0.085926 0.05 0.03], ...
    'String','40', ...
    'Style','edit', ...
    'Tag','TSimulacionEdit', ...
    'Value',40);

PGruaA = axes('Parent',Pantalla, ...
    'Box','on', ...
    'Color',[1 1 1], ...
    'Position',[0.277977 0.581098 0.277023 0.343902], ...
    'Tag','PGruaA', ...
    'XColor',[0 0 0], ...
    'YColor',[0 0 0]);

plot([-1 1],[0 0],'g','linewidth',2) % Línea Representando la viga
hold on
plot([-0.5 -0.5],[0.05 -0.2934],'r','linewidth',4) % Línea representando el cable de la carga
hold on
plot([-0.4 -0.6 -0.6 -0.4 -0.4],[0.05 0.05 0 0 0.05],'b','linewidth',5) % Representación del carro
hold on
plot(-0.5,-0.2934,'*m','linewidth',20) % Representación de la Carga
hold on
plot(-0.45,0,'*b','linewidth',13) % Rueda delantera del Carro
hold on
plot(-0.55,0,'*b','linewidth',13) % Rueda Trasera del Carro
hold off
set(PGruaA,'XLim', [-0.7 0.7]);
set(PGruaA,'YLim', [-0.5 0.5]);
set(PGruaA,'Tag','PGruaA');
grid on

TitlePGruaA = text('Parent',PGruaA, ...
    'Color',[0 0 0], ...
    'HorizontalAlignment','center', ...
    'Position',[0.3 1.03247 0], ...
    'String','Animación del Puente Grua', ...
    'Tag','TitlePGruaA', ...
    'VerticalAlignment','bottom');
set(get(TitlePGruaA,'Parent'),'Title',TitlePGruaA);

XLabelPGruaA = text('Parent',PGruaA, ...
    'Color',[0 0 0], ...

```

```
'HorizontalAlignment','center', ...
'Position',[0.3 -0.0844156 0], ...
'String','Posición X (m)', ...
'Tag','XLabelPGruaA', ...
'VerticalAlignment','cap');
set(get(XLabelPGruaA,'Parent'),'XLabel',XLabelPGruaA);
```

```
YLabelPGruaA = text('Parent',PGruaA, ...
'Color',[0 0 0], ...
'HorizontalAlignment','center', ...
'Position',[-0.121005 0.5 0], ...
'String','Posición Y (m)', ...
'Rotation',90, ...
'Tag','YLabelPGruaA', ...
'VerticalAlignment','baseline');
set(get(YLabelPGruaA,'Parent'),'YLabel',YLabelPGruaA);
```

```
U_t = axes('Parent',Pantalla, ...
'Box','on', ...
'Color',[1 1 1], ...
'Position',[0.67 0.581098 0.277023 0.343902], ...
'Tag','U_t', ...
'XColor',[0 0 0], ...
'XLim',[-0.7 0.7], ...
'YColor',[0 0 0], ...
'YLim',[-0.1 0.9]);
```

```
TitleU_t = text('Parent',U_t, ...
'Color',[0 0 0], ...
'HorizontalAlignment','center', ...
'Position',[-2.22045e-16 0.932468 0], ...
'String','Magnitud de U(t) en el Tiempo', ...
'Tag','Text3', ...
'VerticalAlignment','bottom');
set(get(TitleU_t,'Parent'),'Title',TitleU_t);
```

```
XLabelU_t = text('Parent',U_t, ...
'Color',[0 0 0], ...
'HorizontalAlignment','center', ...
'Position',[-2.22045e-16 -0.184416 0], ...
'String','Tiempo (s)', ...
'Tag','Text4', ...
'VerticalAlignment','cap');
set(get(XLabelU_t,'Parent'),'XLabel',XLabelU_t);
```

```
YLabelU_t = text('Parent',U_t, ...
'Color',[0 0 0], ...
'HorizontalAlignment','center', ...
'Position',[-0.920548 0.4 0], ...
'Rotation',90, ...
'String','U(t)', ...
'Tag','Text8', ...
'VerticalAlignment','baseline');
set(get(YLabelU_t,'Parent'),'YLabel',YLabelU_t);
```

```

posCarro = axes('Parent',Pantalla, ...
'Box','on', ...
'Color',[1 1 1], ...
'Position',[0.277977 0.11 0.277023 0.343902], ...
'Tag','posCarro', ...
'XColor',[0 0 0], ...
'XLim',[-0.7 0.7], ...
'YColor',[0 0 0], ...
'YLim',[-0.1 0.9]);

TitleposCarro = text('Parent',posCarro, ...
'Color',[0 0 0], ...
'HorizontalAlignment','center', ...
'Position',[-2.22045e-16 0.932468 0], ...
'String','Posición del Carro en el Tiempo', ...
'Tag','Text31', ...
'VerticalAlignment','bottom');
set(get(TitleposCarro,'Parent'),'Title',TitleposCarro);

XLLabelposCarro = text('Parent',posCarro, ...
'Color',[0 0 0], ...
'HorizontalAlignment','center', ...
'Position',[-2.22045e-16 -0.184416 0], ...
'String','Tiempo (s)', ...
'Tag','Text41', ...
'VerticalAlignment','cap');
set(get(XLLabelposCarro,'Parent'),'XLabel',XLLabelposCarro);

YLabelposCarro = text('Parent',posCarro, ...
'Color',[0 0 0], ...
'HorizontalAlignment','center', ...
'Position',[-0.920548 0.4 0], ...
'Rotation',90, ...
'String','Posición (m)', ...
'Tag','Text81', ...
'VerticalAlignment','baseline');
set(get(YLabelposCarro,'Parent'),'YLabel',YLabelposCarro);

posAngulo = axes('Parent',Pantalla, ...
'Box','on', ...
'Color',[1 1 1], ...
'Position',[0.67 0.11 0.277023 0.343902], ...
'Tag','posAngulo', ...
'XColor',[0 0 0], ...
'XLim',[-0.7 0.7], ...
'YColor',[0 0 0], ...
'YLim',[-0.1 0.9]);

TitleposAngulo = text('Parent',posAngulo, ...
'Color',[0 0 0], ...
'HorizontalAlignment','center', ...
'Position',[-2.22045e-16 0.932468 0], ...
'String','Posición del Ángulo en el Tiempo', ...

```

```
'Tag','Text32', ...  
'VerticalAlignment','bottom');  
set(get(TitleposAngulo,'Parent'),'Title',TitleposAngulo);  
  
XLabelposAngulo = text('Parent',posAngulo, ...  
'Color',[0 0 0], ...  
'HorizontalAlignment','center', ...  
'Position',[-2.22045e-16 -0.184416 0], ...  
'String','Tiempo (s)', ...  
'Tag','Text42', ...  
'VerticalAlignment','cap');  
set(get(XLabelposAngulo,'Parent'),'XLabel',XLabelposAngulo);  
  
YLabelposAngulo = text('Parent',posAngulo, ...  
'Color',[0 0 0], ...  
'HorizontalAlignment','center', ...  
'Position',[-0.920548 0.4 0], ...  
'Rotation',90, ...  
'String','Ángulo (rad)', ...  
'Tag','Text82', ...  
'VerticalAlignment','baseline');  
set(get(YLabelposAngulo,'Parent'),'YLabel',YLabelposAngulo);
```



ANEXO 3

```

function val_Pert = FuncionPerteneanciaTriangular(val, Inicio, Medio, Fin, tipo)

if tipo == 1
    if val <= Inicio
        val_Pert = 0;
    elseif val <= Medio
        val_Pert = (val - Inicio)/(Medio-Inicio);
    elseif val <= Fin
        val_Pert = (Fin - val)/(Fin - Medio);
    else
        val_Pert = 0;
    end
elseif tipo == 2
    if val <= Medio
        val_Pert = 1;
    elseif val <= Fin
        val_Pert = (Fin - val)/(Fin - Medio);
    else
        val_Pert = 0;
    end
else
    if val <= Inicio
        val_Pert = 0;
    elseif val <= Medio
        val_Pert = (val - Inicio)/(Medio-Inicio);
    else
        val_Pert = 1;
    end
end
end
    
```

ANEXO 4

```

function selcbk(source, eventdata)
    anterior = get(eventdata.OldValue,'String');
    nuevo = get(eventdata.NewValue,'String');
    if (strcmp(anterior,nuevo) == 0)
        if strcmp('PID',nuevo) == 1
            ControlIPGruaFinal(4);
        elseif strcmp('Difuso',nuevo) == 1
            ControlIPGruaFinal(5);
        end
    end
end
end
    
```

REFERENCIAS BIBLIOGRÁFICAS

- Sulla Torres, Raúl Ricardo, “Antología Lógica Difusa Aplicada al Control”.
- Miravete, Antonio, Larrade, Emilio, “Grúas”. Centro Politécnico Superior Universidad de Zaragoza. Edición 1996
- Ogata, Katsuniko, “Ingeniería de Control Moderno” Pearson Educacion, S.A. Madrid 2003. 4ta Edición.
- Ogata, Katsuniko, “Problemas de Ingeniería de Control Utilizando Matlab, Un Enfoque Práctico” Prentice Hall, 1998. Edition Illustrated.
- Kuo Benjamín C. “Sistema de Control Automático” Prentice Hall 1996. 1era Edición.
- Dorf, Bishop, “Sistema de Control Moderno” Pearson Prentice Hall 2010.10ma Edicion.
- José Roberto Vignoni, “Control de Procesos” Instrumentación y Comunicaciones Industriales 2002.
- Tomas Arredondo,” Introducción a la Lógica Difusa” Edición 2012
- Benjamin C. Kuo, “Sistema de Control Automático”, Prentice Hall. 7ma Edición
- Tomas Arredondo, Pedro Ponce Cruz, “Introducción a la Lógica Difusa”, Marcombo, 2011.
- Schneider Electric, “Soluciones garantizadas para grúas industriales”
- Norma Técnica Peruana 737: Utilización y formación de operadores
- Norma Técnica Peruana 736: Generalidades
- Dave Misir, Heidar A. Malki, Guanrong Chen, “Design and analysis of a proportional-integral-derivative controller”. Department of Electrical Engineering, University of Houston, 1995.

- George K. I. Mann, Bao-Gang Hu and Raymond G. Gosine, “Analysis of Direct Action PID Controller Structures”. IEEE Transactions On Systems, Man, And Cybernetics—Part B: Cybernetics, Vol. 29, No. 3, June 1999.
- Han-Xiong Li and H. B. Gatland, “Conventional Control and Its Enhancement”. IEEE Transactions On Systems, Man, And Cybernetics-Part B: Cybernetics, Vol. 26, No. 5, October 1996
- Cao Weibin, Meng Qingjian. “Based on PLC temperature PID - Fuzzy control system design and simulation”. International Conference on Information, Networking and Automation (ICINA) 2010.
- Xu Yang, Bo Wen, Chao-nan Tong, “Modeling and Fuzzy-PID Control of Hydraulic-Rolls Coupled System in Cold Rolling Mills”. School of Automation & Electrical Engineering, Key Laboratory of Advanced Control of Iron and Steel Process (Ministry of Education), University of Science and Technology Beijing.
- Didier Giraldo y Eduardo Giraldo, “Aplicación de un controlador lineal cuadráticos usando un observador de orden mínimo sobre una Grúa Puente”. Scientia Et Technica, vol. XIII, núm. 37, diciembre, 2007, pp. 91-95, Universidad Tecnológica de Pereira, Colombia.
- Jorge Sánchez Bermúdez, Gilberto Machado Burguesa y Concepción Gómez, “Control de oscilaciones en grúas viajeras con Reguladores Fuzzy.” Scientia Et Technica, vol. XIII, núm. 37, diciembre, 2007, pp. 91-95, Universidad Tecnológica de Pereira, Colombia.
- Juanjo Valera García, Eloy Irigoyen, Vicente Gómez-Garay y Mikel Larrea, “Aplicaciones de Técnicas Neuro-genéticas en la resolución del problema de control cinemático en Grúa Puente.” Departamento de Ingeniería de Sistemas y Automática, (UPV/EHU)
- Silvia Gonzáles Álvarez, “Entorno de simulación basado en Marilou y Matlab para control antibalanceo de Grúas puente”. Universidad Carlos III de Madrid. Departamento de Ingeniería de Sistemas y Automática. Jul-2012.

- Carlos Eduardo Betancur P. y Leonardo Taffurht C. “Sintonización de un control óptimo cuadrático con computación evolutiva para una Grúa Puente”. Universidad Tecnológica de Pereira, Ingeniería Eléctrica. Nov 2008.
- http://www.insht.es/InshtWeb/Contenidos/Documentacion/FichasTecnicas/NTP/Ficheros/201a300/ntp_253.pdf
- http://repositorio.utp.edu.co/tesisdigitales/texto/anexos/6298B562_anexo.pdf
- <http://www.cea-ifac.es/actividades/jornadas/XXIX/pdf/255.pdf>
- http://repositorio.utp.edu.co/tesisdigitales/texto/anexos/6298B562_anexo.pdf
- <http://es.scribd.com/doc/56286619/Control-de-un-puente-grua>
- http://www.automatas.org/hardware/teoria_pid.htm
- <http://delta.cs.cinvestav.mx/~gmorales/ldifll/node1.html>
- http://members.tripod.com/jesus_alfonso_lopez/Intro2.html
- <http://personal.telefonica.terra.es/web/mir/ferran/kosko.htm>
- http://catarina.udlap.mx/u_dl_a/tales/documentos/lmt/ramirez_r_o/capitulo3.pdf
- <http://profesores.elo.utfsm.cl/~tarredondo/info/soft-comp/Introduccion%20a%20la%20Logica%20Difusa.pdf>