

ESCUELA POLITÉCNICA DEL EJÉRCITO

DEPARTAMENTO DE ELÉCTRICA Y
ELECTRÓNICA

CARRERA DE INGENIERÍA EN
ELECTRÓNICA,
AUTOMATIZACIÓN Y CONTROL

PROYECTO DE GRADO PARA LA
OBTENCIÓN DEL TÍTULO DE
INGENIERÍA

DISEÑO E IMPLEMENTACIÓN DE UN
CONTROLADOR DIFUSO PARA EL
SISTEMA CARRO-PÉNDULO DEL
LABORATORIO DE SERVOMEKANISMOS

HUGO MARCELO PÉREZ GAIBOR

SANGOLQUÍ – ECUADOR

2012

Declaración de Responsabilidad

ESCUELA POLITÉCNICA DEL EJÉRCITO

INGENIERÍA EN ELECTRÓNICA, AUTOMATIZACIÓN Y CONTROL

DECLARACIÓN DE RESPONSABILIDAD

HUGO MARCELO PÉREZ GAIBOR

DECLARO QUE:

El proyecto de grado denominado “Diseño e implementación de un controlador difuso para el sistema carro-péndulo del laboratorio de servomecanismos”, ha sido desarrollado con base a una investigación exhaustiva, respetando derechos intelectuales de terceros, conforme las citas que constan al pie, de las páginas correspondientes, cuyas fuentes se incorporan en la bibliografía.

Consecuentemente este trabajo es de mi autoría.

En virtud de esta declaración, me responsabilizo del contenido, veracidad y alcance científico del proyecto de grado en mención.

Sangolquí, 10 de Septiembre de 2012

Hugo Marcelo Pérez Gaibor

Autorización de publicación

ESCUELA POLITÉCNICA DEL EJÉRCITO

INGENIERÍA EN ELECTRÓNICA, AUTOMATIZACIÓN Y CONTROL

AUTORIZACIÓN

Yo, Hugo Marcelo Pérez Gaibor

Autorizo a la Escuela Politécnica del Ejército la publicación, en la biblioteca virtual de la Institución del trabajo “Diseño e implementación de un controlador difuso para el sistema carro-péndulo del laboratorio de servomecanismos”, cuyo contenido, ideas y criterios son de mi exclusiva responsabilidad y autoría

Sangolquí, 10 de Septiembre de 2012

Hugo Marcelo Pérez Gaibor

Certificado de tutoría

ESCUELA POLITÉCNICA DEL EJÉRCITO

INGENIERÍA EN ELECTRÓNICA, AUTOMATIZACIÓN Y CONTROL

CERTIFICADO

Ing. Víctor Proaño, M.Sc
Ing. Paúl Ayala

CERTIFICAN

Que el trabajo titulado “Diseño e implementación de un controlador difuso para el sistema carro-péndulo del laboratorio de servomecanismos”, realizado por Hugo Marcelo Pérez Gaibor, ha sido guiado y revisado periódicamente y cumple normas estatutarias establecidas por la ESPE, en el Reglamento de Estudiantes de la Escuela Politécnica del Ejército.

Debido a que se trata de un trabajo de investigación recomiendan su publicación.

El mencionado trabajo consta de un documento empastado y un disco compacto el cual contiene los archivos en formato portátil de Acrobat (pdf). Autorizan a Hugo Marcelo Pérez Gaibor que lo entregue al Ingeniero Víctor Proaño, en su calidad de Coordinador de la Carrera.

Sangolquí, 10 de Septiembre de 2012

Ing. Víctor Proaño

DIRECTOR

Ing. Paúl Ayala

CODIRECTOR

CERTIFICACIÓN

Certificamos que el siguiente proyecto titulado “Diseño e implementación de un controlador difuso para el sistema carro – péndulo del laboratorio de servomecanismos” fue desarrollado en su totalidad por el Sr. Hugo Marcelo Pérez Gaibor con C. I. 1803513645 bajo nuestra dirección.

Ing. Víctor Proaño

DIRECTOR

Ing. Paúl Ayala

CODIRECTOR

AGRADECIMIENTO

Mi agradecimiento a Dios por siempre haberme guiado por el buen camino, iluminarme ante todas las dificultades que existió durante toda esta vida universitaria, a mi padres y hermana por haberme guiado y educado con valores que me van a servir toda la vida.

De igual manera al Ing. Víctor Proaño por su acertada conducción y por haber depositado en mí su entera confianza en el desarrollo de este proyecto. Al Ing. Paúl Ayala quien también estuvo a cargo de la conducción de este proyecto por haber demostrado entereza y profesionalismo a carta cabal en el desarrollo de este proyecto.

De igual manera a las personas a cargo de los laboratorios de servomecanismos que siempre facilitaron el acceso para el desarrollo de la parte práctica.

Finalmente a mis amigos los cuales estuvieron a mi lado de una u otra forma sabiendo apoyarme en momentos malos y buenos.

DEDICATORIA

A mis padres Hugo y Sara, a mi hermana Gaby y a todos aquellos que confiaron en mí.

PRÓLOGO

La Escuela Politécnica del Ejército posee en el laboratorio de servomecanismos el sistema Carro – Péndulo que forma parte de una gama de sistemas que permiten aplicar la teoría de control en forma práctica. Este sistema como tal justifica de manera idónea la aplicación de una lógica difusa para su control en modo péndulo invertido debido a la naturaleza de su funcionamiento.

El presente proyecto complementa al manual de usuario de este sistema ya que no detalla ciertos aspectos que permitan entender al alumno el *hardware* así como el *software* que vienen por defecto como base para poder realizar nuevas pruebas sobre el mismo. Parte del estudio del sistema Carro – Péndulo juntamente con la teoría de lógica difusa. Explica la obtención y desarrollo del modelo no lineal matemático que se encuentra en el manual de usuario. Identifica las constantes que se encuentran inmersas en dicho modelo matemático. Expone los controladores que se encuentran en los modelos de simulación y experimentación del sistema. Con base a todo lo mencionado anteriormente se diseña un controlador de lógica difusa para verificarlo en forma simulada y experimental.

ÍNDICE DE CONTENIDO

AGRADECIMIENTO.....	6
DEDICATORIA	7
PRÓLOGO.....	8
ÍNDICE DE CONTENIDO.....	9
CAPÍTULO I.....	12
INTRODUCCIÓN	12
1.1 JUSTIFICACIÓN E IMPORTANCIA	12
1.2 ALCANCE DEL PROYECTO	13
1.3 REAL TIME WINDOWS TARGET (RTWT)	14
1.4 REAL TIME WORKSHOP (RTW).....	16
1.5 LÓGICA DIFUSA	17
1.5.1 Variable y valor lingüístico.....	18
1.5.2 Conjuntos difusos.....	19
1.5.3 Funciones de pertenencia	22
1.5.4 Operaciones entre conjuntos difusos.....	26
1.5.5 Tipos de operadores	29
1.5.6 Implicación difusa.....	30
1.5.7 Inferencia difusa.....	30
1.5.8 Aplicación de la lógica difusa en sistemas de control.....	31
1.5.9 Etapa de fusificación.....	33
1.5.10 Base de reglas	33
1.5.11 Mecanismo de inferencia	34
1.5.12 Etapa de defusificación	35
CAPÍTULO II	37
MODELO MATEMÁTICO DEL SISTEMA CARRO – PÉNDULO.....	37
2.1 FUNCIONAMIENTO DEL SISTEMA CARRO-PÉNDULO	37
2.1.1 Descripción del hardware.....	41
2.1.1.1 Fuente de alimentación.....	42

2.1.1. Circuito de potencia	42
2.1.1. Tarjeta de interfaz RT-DAC4/PCI-D	44
2.1.1. Sensor de posición.....	46
2.1.1. Motor DC	49
2.1.2 Descripción del software	50
2.2 MODELO MATEMÁTICO	52
2.2.1 Cálculo del momento de inercia	59
2.3 DESCRIPCIÓN DEL MODELO MATEMÁTICO DESARROLLADO EN SIMULINK DE MATLAB ®	61
2.4 CUANTIFICACIÓN Y TIEMPO DE MUESTREO EN EL SISTEMA DE ADQUISICIÓN.....	68
CAPÍTULO III.....	72
IDENTIFICACIÓN DE PARÁMETROS DEL SISTEMA, CONTROLADOR BASADO EN REGLAS, CONTROLADOR PID.....	72
3.1 IDENTIFICACIÓN DE PARÁMETROS FÍSICOS DEL SISTEMA.....	72
3.1.1 Restablecimiento de encoders	74
3.1.2 Comprobación de las señales de medición.....	74
3.1.3 Identificación de la magnitud de control.....	76
3.1.4 Mínima fuerza necesaria para mover el carro	79
3.1.5 Identificación de la fricción del péndulo	82
3.1.6 Presentación de parámetros.....	88
3.2 CONTROLADOR BASADO EN REGLAS	89
3.2.1 Etapa de balanceo del péndulo	94
3.2.2 Etapa de control de posición del péndulo.....	95
3.2.3 Etapa de control de posición del carro	96
3.3 CONTROLADOR PID	96
3.3.1 Diseño del controlador PID.....	97
CAPÍTULO IV.....	99
DISEÑO E IMPLEMENTACIÓN DEL CONTROLADOR DIFUSO.....	99
4.1 CONTROLADOR DIFUSO PROPORCIONAL DERIVATIVO CON BALANCEO	100
4.1.1 Definición de variables y valores lingüísticos.....	101
4.1.2 Rango de variables	102
4.1.3 Funciones de pertenencia	103
4.1.4 Base de reglas.....	108

4.1.5 Implementación del controlador difuso PD con balanceo en SIMULINK	113
4.1.6 Simulación del controlador difuso PD con balanceo en SIMULINK	116
4.1.7 Ajuste del controlador difuso PD con balanceo en SIMULINK	117
4.1.8 Implementación del controlador difuso PD con balanceo en el modelo experimental .	119
4.1.9 Experimentación del controlador difuso PD con balanceo	122
4.1.10 Ajuste experimental del controlador difuso PD con balanceo	123
4.2 CONTROLADOR DIFUSO PD	126
4.2.1 Implementación del controlador difuso PD en SIMULINK	126
4.2.2 Simulación del controlador difuso PD en SIMULINK	128
4.2.3 Implementación del controlador difuso PD en el modelo experimental	129
4.2.4 Experimentación del controlador difuso PD	130
4.3 CONTROLADOR DIFUSO PI.....	131
4.3.1 Definición de variables y valores lingüísticos.....	131
4.3.2 Rango de variables	132
4.3.3 Funciones de pertenencia	133
4.3.4 Base de reglas.....	135
4.3.5 Implementación del controlador difuso PI en SIMULINK.....	136
4.3.6 Simulación del controlador difuso PI en SIMULINK.....	136
4.3.7 Implementación del controlador difuso PI en el modelo experimental.....	138
4.3.8 Experimentación del controlador difuso PI.....	138
CAPÍTULO V	141
CONCLUSIONES Y RECOMENDACIONES	141
5.1 Conclusiones	141
5.2 Recomendaciones.....	144
REFERENCIAS BIBLIOGRÁFICAS.....	146
ÍNDICE DE FIGURAS.....	150
ÍNDICE DE TABLAS	152
GLOSARIO.....	153

CAPÍTULO I

INTRODUCCIÓN

1.1 JUSTIFICACIÓN E IMPORTANCIA

El Departamento de Eléctrica y Electrónica tiene como uno de sus objetivos el desarrollo de investigación. El proyecto que se pretende realizar se enfoca dentro de la línea de investigación de Automática y Control, específicamente en el campo del Control Inteligente.

Cuando el sistema es complejo o menos conocido con precisión es difícil de caracterizar el conocimiento adecuado por medios convencionales para poder llegar a un mecanismo para la toma de decisiones. Hay que encontrar formas alternativas de codificar el conocimiento disponible sobre el sistema, que puede resultar en el desarrollo de controladores inteligentes^[1]. Por lo que la identificación precisa de la dinámica del modelo de laboratorio es esencial para la generación de un algoritmo de control apropiado.

^[1] Inteco, PENDULUM – CART SYSTEM User's Manual. 1997. (1)

En el manual de usuario del sistema carro-péndulo, no se detallan ciertos aspectos que permitan entender al alumno su funcionamiento como por ejemplo los valores de sintonización de los parámetros de los controladores que se usan aparecen sin ninguna justificación matemática y por tanto no contribuyen al aprendizaje de la teoría de control que es el aspecto más importante para el cual se realiza las prácticas de laboratorio. Es así que este proyecto complementará el manual mediante un análisis y explicación de los cálculos de distintos controladores.

Finalmente, la conclusión del proyecto permitirá disponer de una herramienta más para verificar la teoría de control inteligente de tal manera que se incremente el número de alumnos que puedan realizar experimentos en el laboratorio de servomecanismos.

1.2 ALCANCE DEL PROYECTO

Se realizará el estudio del sistema carro-péndulo para determinar las posibilidades y limitaciones del sistema que serán descritas en el documento del proyecto.

Se realizará el análisis de la teoría involucrada para resolver el problema de control del sistema, para lo cual es necesario realizar una correcta identificación de los parámetros de la planta.

Se analizarán los problemas de compromiso entre cuantización y velocidad de muestreo, el diseño de controladores basados en reglas, el cálculo de los parámetros de sintonización del controlador PID. Se analizará el compensador DIFUSO que viene descrito en la guía. En este controlador se comprueba el funcionamiento del control para el carro en el modo grúa.

La parte más importante del trabajo será el diseño e implementación de un controlador difuso para el sistema carro-péndulo en el modo péndulo invertido. Se realizará la simulación del sistema de control en MATLAB y se implementará el controlador en el sistema, registrando los resultados que se obtienen.

Cabe destacar que la fácil implementación de los controladores es una característica propia del sistema ya que posee los elementos necesarios como son la tarjeta de adquisición, y la posibilidad de trabajo en tiempo real con la herramienta RTW de MATLAB.

1.3 REAL TIME WINDOWS TARGET (RTWT)

Real Time Windows Target es un componente de la caja de herramientas de Matlab que permite realizar simulaciones en tiempo real. Así como control en tiempo real, simulación en tiempo real para plantas físicas por ejemplo turbinas de avión, modelamiento de sistemas físicos, etc.

El RTWT se considera como la tarjeta de adquisición de datos en el software Simulink ya que posee bloques de entradas analógicas, salidas analógicas, contadores, temporizadores.

Real Time Windows Target es una solución aplicada en computadoras para prototipos y pruebas de sistemas en tiempo real, para realizar visualizaciones de señales con el bloque de osciloscopio de Simulink, modificación de parámetros y control en tiempo real con un conjunto de bloques que proporcionan conexiones entre las entradas y salidas físicas de la tarjeta de adquisición de datos y el modelo en tiempo real^[2].

Algunas de sus características son:

- Kernel en tiempo real, es decir hace de interfaz con el sistema operativo Windows para asegurar que la aplicación en tiempo real se está ejecutando en el tiempo de muestreo seleccionado.
- Aplicación en tiempo real
- Adquisición y análisis de señales
- Sintonización de parámetros

El proceso de desarrollo es el siguiente:

1. Diseñar un sistema de control
2. Crear el modelo en Simulink

^[2] Acuña, Byron e Ibarra, Oswaldo. *Diseño e implementación de un sistema controlador de temperatura pid para la unidad air flow temperature control system mediante la utilización de la herramienta rtw de matlab*. 2010. (2)

3. Correr la simulación en tiempo no real
4. Crear una aplicación en tiempo real
5. Correr la aplicación en tiempo real
6. Analizar y visualizar las señales de datos.

1.4 REAL TIME WORKSHOP (RTW)

Real Time Workshop es una parte integral del entorno de Simulink que genera y ejecuta archivos en código C para desarrollar y probar los algoritmos modelados en Simulink y códigos en Matlab. El código en C resultante puede ser utilizado para aplicaciones en tiempo real porque se basa en una programación por interrupciones haciendo notar a la CPU la aparición de alguna circunstancia que requiera su intervención^[3].

El RTW trabaja sobre la plataforma de Matlab y Simulink interpretando los códigos .m o los diagramas de bloques de Simulink .mdl para generar un modelo en tiempo real .rtw el cual mediante un compilador (Visual C++) y un programa identificador del lenguaje del compilador (TLC) genera los archivos .c con sus respectivas librerías .h para de esta manera generar un ejecutable que pueda ser interpretado fácilmente en lenguaje de bajo nivel y que a su vez sea compatible con la interfaz gráfica de Simulink en modo externo. El archivo ejecutable .exe generado por la herramienta RTW se forma a partir del tipo de archivo que hayamos seleccionado y su función principal es poder ejecutarse en plataforma

^[3] Cilento, Augusto. *Sistemas de control de motores en Tiempo Real mediante Matlab*. 2007. (3)

de Windows o Unix independientemente pudiendo ser llamada y controlada por Matlab. Este archivo es llamado durante la ejecución del modelo en tiempo real para trabajar conjuntamente con la interfaz de usuario^[4].

El proceso de desarrollo es el siguiente:

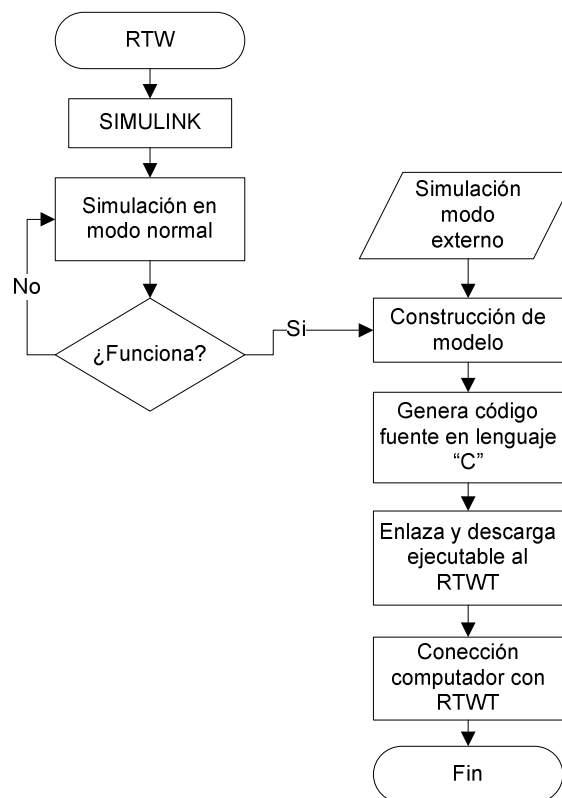


Figura. 1. 1. Algoritmo RTW

1.5 LÓGICA DIFUSA

La lógica difusa es una técnica de la Inteligencia Artificial que permite trabajar con información de entrada ambigua, imprecisa, o incompleta que no es exacta para poder obtener una conclusión.

^[4] Universidad Nacional de Quilmes, IACI. *Real Time Workshop*, Laboratorio de Automatización. (4)

Expresa el conocimiento común, el cual es de tipo lingüístico cualitativo, en un lenguaje matemático (cuantitativo) regido por la teoría de conjuntos difusos y funciones de pertenencia asociados a estos^[5].

La lógica difusa utiliza expresiones que no son ni totalmente ciertas ni completamente falsas, es decir aplica conceptos que pueden tomar un valor cualesquiera de veracidad dentro de un conjunto de valores que oscilan entre dos extremos, la verdad absoluta y la falsedad total. De este modo la lógica difusa puede funcionar con expresiones como: “mucho”, “muy” y “un poco”, pero conociendo el valor que representan como por ejemplo: una puerta no tiene porque estar necesariamente abierta (valor 1), ni cerrada (valor 0), sino que además puede estar a medias (valor 0.5), bastante (valor 0.8), casi cerrada (0.1), etc.

1.5.1 Variable y valor lingüístico

Una variable permite especificar la incertidumbre o subjetividad de un determinado concepto, es decir puede tomar por valor términos del lenguaje natural, como: temperatura ambiente, estatura, velocidad, etc., lo que se define dentro de conjuntos difusos como universo de discurso.

^[5] Pérez, Rosanna. *Procesado y optimización de espectros gamma mediante técnicas de lógica difusa: aplicación a la identificación de materiales pictóricos*. 2005. (5)

El valor lingüístico es la función de pertenencia de una variable, presenta una colección de atributos que puede adquirir la variable, y cada atributo está representado por un conjunto difuso. Como por ejemplo: la variable *Estatura* que tiene tres atributos *Bajo*, *Mediano* y *Alto*, y cada uno de estos atributos estaría representado por un conjunto difuso respectivo. Estos atributos reciben el nombre de *Valores Lingüísticos*.

1.5.2 Conjuntos difusos

Un conjunto clásico se define como una colección de elementos que existen dentro de un universo. Cada uno de los elementos del universo pertenecen o no a un determinado conjunto. Por lo tanto, cada conjunto puede definirse completamente por una función de pertenencia, que opera sobre los elementos del universo, y que le asigna un valor de 1 si el elemento pertenece al conjunto, y de 0 si no pertenece. Por ejemplo si A es el conjunto de vehículos, las funciones de pertenencia μ de un carro y un bolígrafo al conjunto difuso de vehículos es: $\mu_{vehículos}(carro) = 1$ y $\mu_{vehículos}(bolígrafo) = 0$.

Ahora bien, un conjunto difuso se define de forma similar, con una diferencia conceptual importante: un elemento puede pertenecer parcialmente a más de un conjunto.

El universo de los conjuntos clásicos, toma el nombre de universo de discurso en los conjuntos difusos, ya que involucran una serie de variables que se encuentran asociadas con un determinado valor lingüístico definido por una palabra, adjetivo o etiqueta

lingüística^[6]. Matemáticamente se asigna a cada posible elemento que existe en el universo un valor que representa su grado de pertenencia o membrecía en el conjunto difuso.

Cada elemento de un conjunto difuso presenta un grado de pertenencia a un conjunto difuso que puede tomar cualquier valor entre 0 y 1, como se puede ver en la Figura. 1. 2.

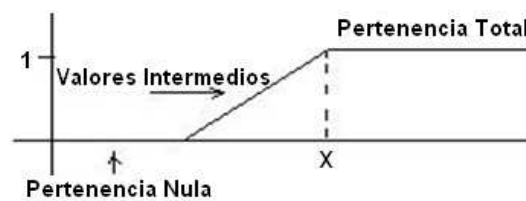


Figura. 1. 2. Conjunto Difuso

A un conjunto difuso se lo puede caracterizar de acuerdo a lo siguiente:

1. Los conjuntos difusos pueden tener cualquier valor entre el intervalo 0 y 1.
2. Un elemento puede pertenecer (parcialmente) a un conjunto difuso y simultáneamente pertenecer (parcialmente) al complemento de dicho conjunto.
3. Elementos en las fronteras de los conjuntos difusos que pueden pertenecer al mismo tiempo a más de un conjunto difuso con diferente grado de pertenencia.

Para observar la diferencia que existe entre los conjuntos clásicos y los conjuntos difusos se representa en la Figura. 1. 3. la estatura de una persona: “baja”, “media”, “alta”,

^[6] Maldonado, Casas, Canas. *Fuzzy sets application to assess the parameters of the bridge seismic vulnerability*. 2002. (6)

por lo que si una persona mide 179 cm pertenece al conjunto estatura media en los conjuntos clásicos mientras en los conjuntos difusos pertenece tanto al conjunto estatura media como al conjunto estatura alta en un 5% y 80% respectivamente.

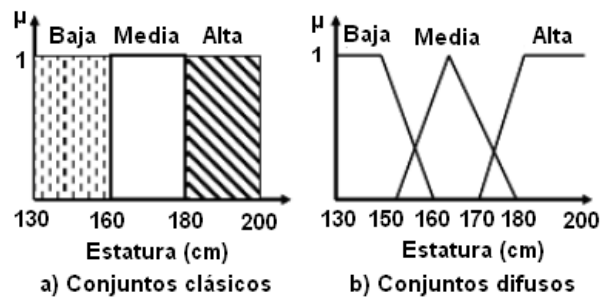


Figura. 1. 3. Representación de la estatura de una persona

Algunas propiedades de los conjuntos difusos se mencionan a continuación:

- Si un conjunto difuso A contiene todos los elementos de U con un valor de pertenencia distinto de cero en A , se tiene un conjunto difuso “crisp” (numérico).
- Si un conjunto difuso no contiene ningún elemento de U , se tiene un conjunto difuso vacío.
- Si un conjunto difuso es un solo elemento de U , se tiene un conjunto difuso “singleton”.

1.5.3 Funciones de pertenencia

El grado de pertenencia de un elemento a un conjunto difuso se define mediante una función característica asociada al conjunto difuso, llamada *función de pertenencia*.

La función de pertenencia permite representar gráficamente un conjunto difuso. Asocia a cada elemento del conjunto difuso el grado con que pertenece al valor lingüístico, 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], como se muestra en la Figura. 1. 4. Así, si se define un conjunto difuso A sobre un universo de discurso U con x elementos: la función de pertenencia queda definida por $\mu_A(x)$, donde para cada valor que pueda tomar x, la función de pertenencia $\mu_A(x)$ proporciona el grado de pertenencia de este valor x al conjunto difuso A.

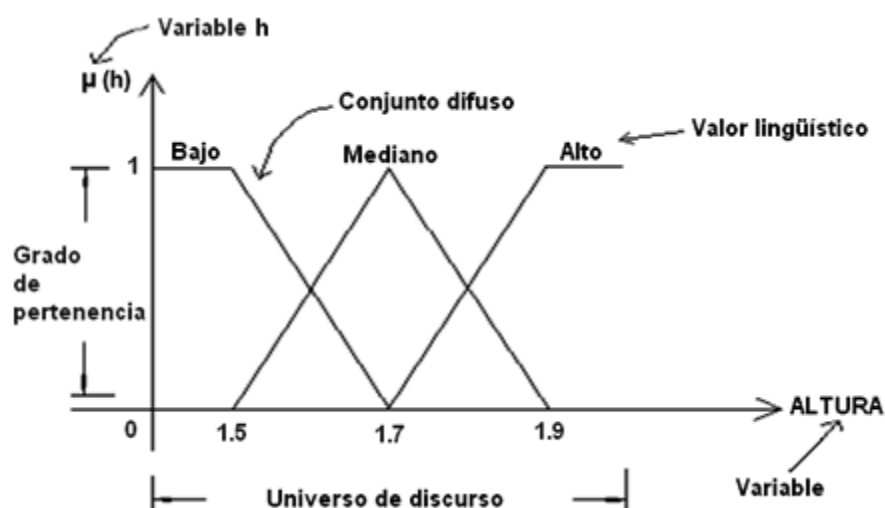


Figura. 1. 4 Función de pertenencia

A las funciones de pertenencia se las puede caracterizar de la siguiente manera:

- La función de pertenencia proporciona una medida del grado de pertenencia de un elemento de U con el conjunto difuso.
- La forma de la función de pertenencia utilizada, depende del criterio aplicado en la resolución de cada problema.
- La única condición que debe cumplir una función de pertenencia es que tome valores entre 0 y 1, con continuidad.

Algunas propiedades de las funciones de pertenencia se mencionan a continuación:

- El punto de cruce de un conjunto difuso es el punto de U cuyo valor de pertenencia al conjunto es igual a 0.5 (cuando son equidistantes).
- Dos conjuntos difusos A y B son iguales si y sólo si sus funciones de pertenencia $\mu_A(x)$ y $\mu_B(x)$ son iguales.
- El conjunto difuso B contiene al conjunto difuso A , esto es $A \subset B$, si y sólo si $\mu_A(x) \leq \mu_B(x)$ para todo $x \in U$.

Existen diferentes tipos de funciones de pertenencia: función gamma, función sigma, función trapezoidal, función gaussiana, como se muestra en la Figura. 1. 5.

Para propósitos prácticos, la función triangular y trapezoidal son las estándares especialmente en aplicaciones en tiempo real, debido a la sencillez de sus fórmulas y a su eficiencia computacional^[7]. Estas funciones pueden ser descritas con 3 o 4 puntos.

El uso de funciones de pertenencia complejas no aportan mayor precisión, puesto que se está definiendo un concepto difuso^[8].

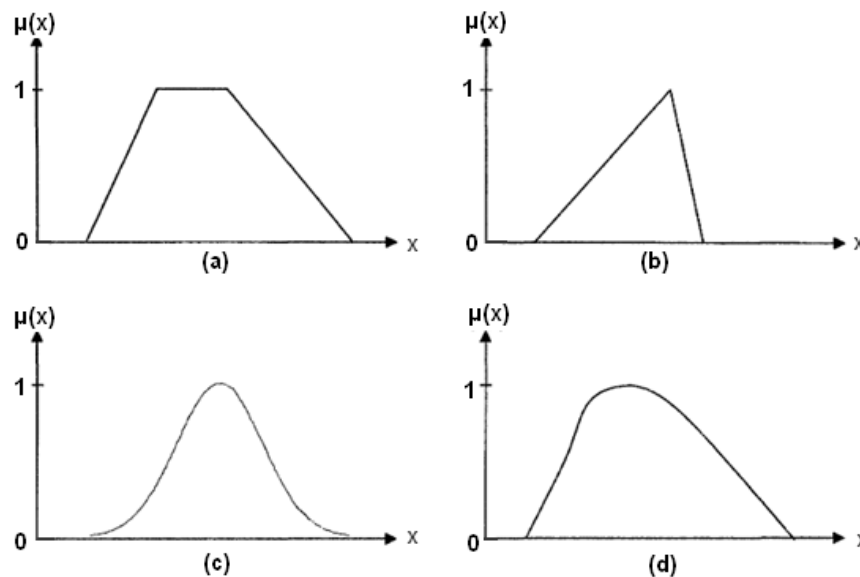


Figura. 1. 5. Funciones de pertenencia: (a) trapezoidal, (b) triangular, (c) gaussiana, (d) sigma

La función trapezoidal se utiliza habitualmente en sistemas difusos sencillos, pues permite definir un conjunto difuso con pocos datos, y calcular su valor de pertenencia con pocos cálculos, resultando adecuada para modelar propiedades que comprenden un rango de valores^[9].

^[7] Rodríguez, Alejandra. *Astrofísica Computacional: Aplicación de técnicas de inteligencia artificial en la clasificación y parametrización de espectros estelares*. 2010. (19)

^[8] Universidad Politécnica de Madrid, eMathTeacher. *Método de Mamdani de Inferencia Borrosa*. (18)

^[9] Cárdenas, Moncayo. *Técnicas de control difuso aplicadas a la navegación de un robot móvil*. 2006. (7)

La función triangular es adecuada para modelar propiedades con un valor de inclusión distinto de cero para un rango de valores estrecho en torno a un punto b . La función gaussiana tiene forma de campana, es adecuada para los conjuntos definidos en torno a un valor c , modelando propiedades como *medio*, *normal*, *cero*.

La función sigma resulta adecuada para modelar propiedades como *grande*, *mucho*, *positivo*, se caracteriza por tener un valor de inclusión distinto de 0 para un rango de valores por encima de cierto punto a siendo 0 por debajo de a y 1 para valores mayores que c , su punto de cruce (valor 0.5) es $b=(a+c)/2$; y entre los puntos a y c es de tipo cuadrático (suave).

Estas funciones pueden estar solapadas o no, el hecho de estar solapadas pone de manifiesto un aspecto clave de la lógica difusa: una variable puede pertenecer con diferentes grados de pertenencia a varios conjuntos difusos a la vez.

Por otra parte a mayor número de funciones de pertenencia asociadas a una misma variable se tiene mayor resolución, pero también mayor complejidad computacional.

Existen varias formas para elegir la función de pertenencia que depende de la aplicación en particular. El método *Horizontal* es el más sencillo que se basa en el

conocimiento humano de los expertos. El método *Vertical* es utilizando la colección de datos para diseñar la función^[10].

1.5.4 Operaciones entre conjuntos difusos

Cuando una variable cubre el dominio de más de un valor lingüístico, la variable difusa final es deducida por alguna operación que toma en cuenta el grado de pertenencia de cada uno de los valores lingüísticos.

Al igual que en la lógica clásica, en la lógica difusa existen tres operaciones básicas sobre conjuntos: unión, intersección y complemento. La definición de estas operaciones se hace empleando el concepto de función de pertenencia de los conjuntos.

Unión: Para dos conjuntos difusos A y B definidos sobre el mismo universo de discurso U , y con funciones de pertenencia $\mu_A(x)$ y $\mu_B(x)$; el resultado de efectuar la operación de unión entre estos dos conjuntos (en lógica binaria es el equivalente de una operación OR), es un nuevo conjunto difuso $A \cup B$ definido sobre el mismo universo de discurso, y con función de pertenencia $\mu_{A \cup B}(x)$ como se muestra en la Figura. 1. 6., dada por:

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

^[10] Gómez, Galindo. *Conjuntos y sistemas difusos*. Departamento de Lenguajes y Ciencias de la Computación, Universidad de Málaga. (8)

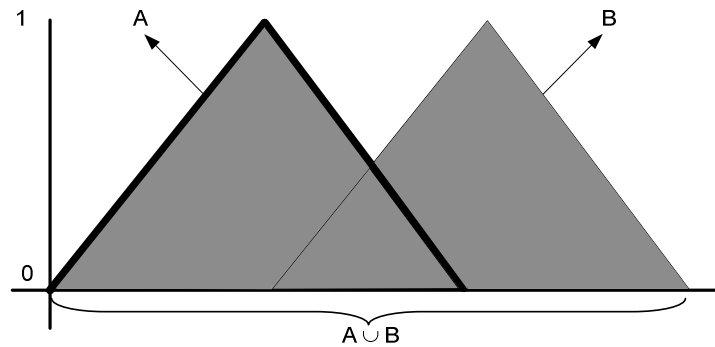


Figura. 1. 6. Operación unión de conjuntos difusos

Intersección: Para dos conjuntos difusos A y B definidos sobre el mismo universo de discurso U , y con funciones de pertenencia $\mu_A(x)$ y $\mu_B(x)$; el resultado de efectuar la operación de intersección entre estos dos conjuntos (en lógica binaria es el equivalente de una operación AND), es un nuevo conjunto difuso $A \cap B$ definido sobre el mismo universo, y con función de pertenencia $\mu_{A \cap B}(x)$, como se muestra en la Figura. 1. 7., dada por:

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

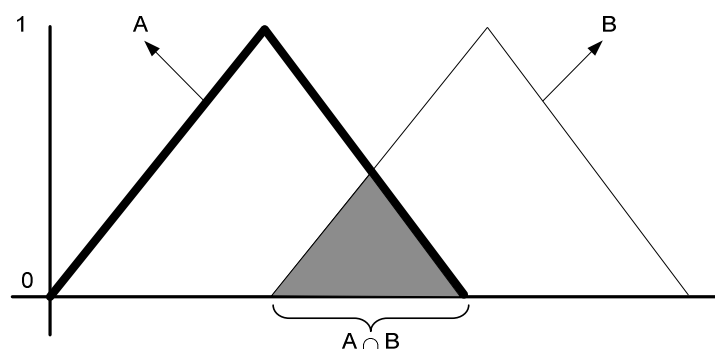


Figura. 1. 7. Operación intersección de conjuntos difusos

Complemento: Para un conjunto difuso A definido sobre un universo de discurso U , y cuya función de pertenencia es $\mu_A(x)$; el resultado de efectuar la operación de complemento (en lógica binaria es el equivalente de la operación NOT), es un nuevo conjunto difuso A' definido sobre el mismo universo, y con función de pertenencia $\mu_{A'}(x)$, como se muestra en la Figura. 1. 8., dada por:

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

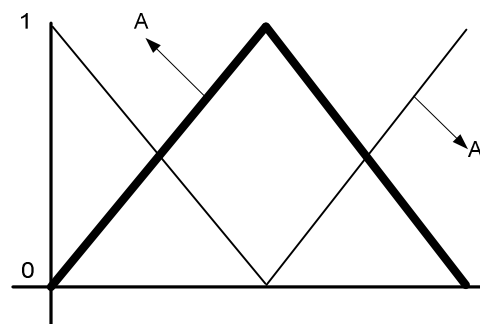


Figura. 1. 8. Operación complemento de conjuntos difusos

Estas tres operaciones definidas para conjuntos difusos cumplen al igual que en la teoría clásica de conjuntos: asociatividad, conmutatividad y distributividad así como las leyes de Morgan^[11].

Sin embargo hay que destacar que existen dos leyes fundamentales de la teoría clásica de conjuntos como son el principio de contradicción: $A \cup \bar{A} = U$ y el principio de exclusión: $A \cap \bar{A} = \emptyset$ que no se cumplen en la teoría de conjuntos difusos^[12].

^[11] Medina, Pablo. *Estudio del desempeño del motor de inducción usando un controlador difuso*. 2007. (9)

1.5.5 Tipos de operadores

El establecimiento de una similitud matemática entre la teoría de conjuntos, la lógica tradicional y el álgebra booleana, que garantice que cada teorema enunciado en una de ellas tenga un parecido en las otras dos permitirá traducir las reglas difusas a relaciones entre conjuntos difusos y éstas a términos de operadores algebraicos con los que se puede trabajar, como se muestra en la Tabla. 1.1.

Teoría de Conjuntos	Álgebra Booleana	Lógica Tradicional
Intersección	Conjunción	AND
Unión	Disyunción	OR
Complemento	Negación	NOT

Tabla. 1. 1. Similitud entre operadores

En lógica difusa una proposición puede representarse por un conjunto difuso: "*X es A*" donde *X* corresponde a un conjunto *A* con función de pertenencia $\mu_A(x)$, mientras que "*Y es B*" donde *Y* corresponde a un conjunto *B* con función de pertenencia $\mu_B(y)$, y la combinación de estas dos proposiciones con el operador AND, es decir la proposición "*X es A AND Y es B*" corresponde a un nuevo conjunto difuso con función de pertenencia $\mu_{A \text{ AND } B}(x, y) = \min \{\mu_A(x), \mu_B(y)\}$.

^[12] Pineda, Vivas. *Control de un modelo aerodinámico aplicando sistemas difusos*. 2008. (10)

1.5.6 Implicación difusa

Un análisis especial debe hacerse con el operador lógico de implicación, que combina dos proposiciones con la expresión *SI... Entonces... (If... Then...)*, y que es el fundamento de las inferencias realizadas en sistemas de lógica difusa.

El operador lógico de implicación, permite encontrar un camino matemático para evaluar proposiciones como por ejemplo: "*Si las vibraciones son altas Entonces el rodamiento está desgastado*", o "*Si los ingresos del cliente son bajos Entonces su capacidad de endeudamiento es poca*". A la relación entre dos proposiciones a través del operador lógico de implicación, se las llama también **REGLAS**, las mismas que servirán en lo posterior para definir la base de reglas de los sistemas de control difuso.

1.5.7 Inferencia difusa

La inferencia difusa es el conjunto de proposiciones *IF-THEN* que modelan el problema que se quiere resolver. Una regla difusa simple tiene la forma:

“Si u es A entonces v es B”

Dónde A y B son conjuntos difusos definidos en los rangos de “u” y “v” respectivamente. Una regla expresa un tipo de relación entre conjuntos A y B cuya función

característica sería $\mu_{A \rightarrow B}(x, y)$ y representa lo que se conoce como implicación lógica. La elección apropiada de esta función está sujeta a las reglas de la lógica proposicional.

Entonces como se ve las reglas expresan un tipo de relación entre dos o más proposiciones y representan una implicación lógica, donde el conocimiento se expresa por reglas.

Debido a que un Sistema de Lógica Difusa puede, en general, tener varias entradas y varias salidas, la forma genérica de las reglas presentes en la Base de Reglas es la siguiente:

IF X_1 es A_1 AND X_2 es A_2 AND...AND X_m es A_m THEN Y_1 es B_1 AND Y_2 es B_2 AND... AND Y_n es B_n

En estas reglas, $A_1, A_2, \dots, A_m, B_1, B_2, \dots, B_n$ son los valores Lingüísticos de las Variables respectivas^[13].

1.5.8 Aplicación de la lógica difusa en sistemas de control

Un sistema de control difuso trabaja de manera muy diferente a los sistemas de control convencionales. Usan el conocimiento experto para generar una base de conocimientos que dará al sistema la capacidad de tomar decisiones sobre ciertas acciones que se presentan en su funcionamiento^[14]. Los sistemas de control difuso

^[13] Duarte, Oscar. *Sistemas de lógica difusa*. Fundamentos. (11)

^[14] Passino, Kevin y Yurkovich, Stephen. *Fuzzy control*. 1998. (12)

permiten describir un conjunto de reglas que utilizaría una persona para controlar un proceso y a partir de estas reglas generar acciones de control.

A continuación se muestra en la Figura. 1. 9. la arquitectura del controlador difuso en un proceso.

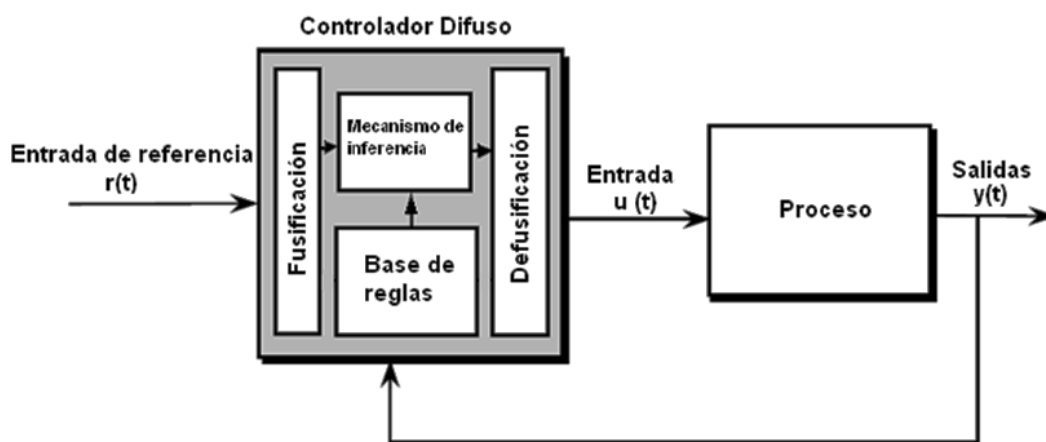


Figura. 1. 9. Arquitectura del Controlador Difuso

Como se puede observar en la Figura. 1. 9., el controlador difuso cuenta con cuatro bloques:

- Etapa de Fusificación
- Base de Reglas
- Mecanismo de Inferencia
- Etapa de Defusificación

Se detallan cada una de ellas a continuación.

1.5.9 Etapa de fusificación

Tiene como objetivo transformar cada dato de la señal de entrada del modelo en variables difusas, donde a cada dato de entrada se le asigna un grado de pertenencia a uno o varios de los conjuntos difusos en que se divide un rango de valores posibles para dicha entrada.

Para esta etapa se deben tener definidos los rangos de variación de las variables de entrada y los conjuntos difusos asociados con sus respectivas funciones de pertenencia.

1.5.10 Base de reglas

Contiene las reglas lingüísticas del control y la información referente a las funciones de pertenencia de los conjuntos difusos. Como ya se ha visto en la inferencia difusa, las reglas tienen la forma:

$$\textit{Si } u_1 \textit{ es } A \textit{ y } u_2 \textit{ es } B \textit{ entonces } y \textit{ es } C$$

Donde A, B son los conjuntos difusos de las variables de entrada u_1 y u_2 , mientras C es el de la variable de salida y .

Para determinar las reglas difusas existen varias maneras, entre las que se destacan las basadas en:

- La experiencia de expertos y el conocimiento de ingeniería de control. La base de reglas se determina a partir de entrevistas con el operador o a través del conocimiento de la dinámica del proceso.
- La modelación del proceso. Los parámetros de la base de conocimiento se obtienen a partir de datos de entrada y salida del proceso.

1.5.11 Mecanismo de inferencia

El mecanismo de inferencia interpreta cada regla como una implicación difusa. Es decir el bloque de inferencia es aquel en el que se realiza la “traducción matemática” de las reglas difusas. Las reglas difusas modelan el sistema para poder trabajar con ellas y extraer un resultado.

La máquina de inferencia realiza la tarea de calcular las variables de salida a partir de las variables de entrada, mediante las reglas del controlador y la inferencia difusa, entregando conjuntos difusos de salida.

La secuencia de cálculos que realiza el mecanismo de inferencia^[15] incluye:

- Determinar el grado de cumplimiento de las variables difusas para cada regla, a partir de los grados de pertenencia de las variables de entrada obtenidos en la etapa de fusificación.

^[15] Pineda, Vivas. *Control de un modelo aerodinámico aplicando sistemas difusos*. 2008. (10)

- A partir del grado de cumplimiento de las variables difusas, en cada regla se tiene un conjunto difuso de salida, que tiene asociado una función de pertenencia. Por lo tanto se tiene un conjunto de salida con una función de pertenencia.

- Para evaluar el conjunto total de reglas, se unen los conjuntos difusos resultantes de cada regla, generándose un conjunto de salida con una nueva función de pertenencia.

De esta forma, se obtiene una salida difusa del controlador, con una función de pertenencia.

1.5.12 Etapa de defusificación

En este bloque a partir de los conjuntos difusos obtenidos en el bloque de inferencia y mediante métodos matemáticos de defusificación, se obtiene el valor numérico y determinante de la variable de salida.

Existen diferentes métodos de defusificación, algunos de los cuales se describen a continuación:

- **Método del máximo.** La salida corresponde al valor para el cual la función de pertenencia alcanza su máximo.

-
- **Media del máximo.** La salida es el promedio entre los elementos del conjunto que tienen un grado de pertenencia máximo.

 - **Centro de gravedad.** Genera como salida el valor correspondiente al centro de gravedad de la función de pertenencia del conjunto de salida.

CAPÍTULO II

MODELO MATEMÁTICO DEL SISTEMA CARRO – PÉNDULO

2.1 FUNCIONAMIENTO DEL SISTEMA CARRO-PÉNDULO

El sistema mecánico carro-péndulo es un sistema listo para realizar experimentos de control. Los componentes mecánicos y electrónicos se muestran en la Figura. 2. 1.: un péndulo (formado por una carga y un polo), un carro, un motor dc, un sensor de posición, un sensor de ángulo, una fuente de alimentación, un circuito de potencia y una tarjeta de interfaz.

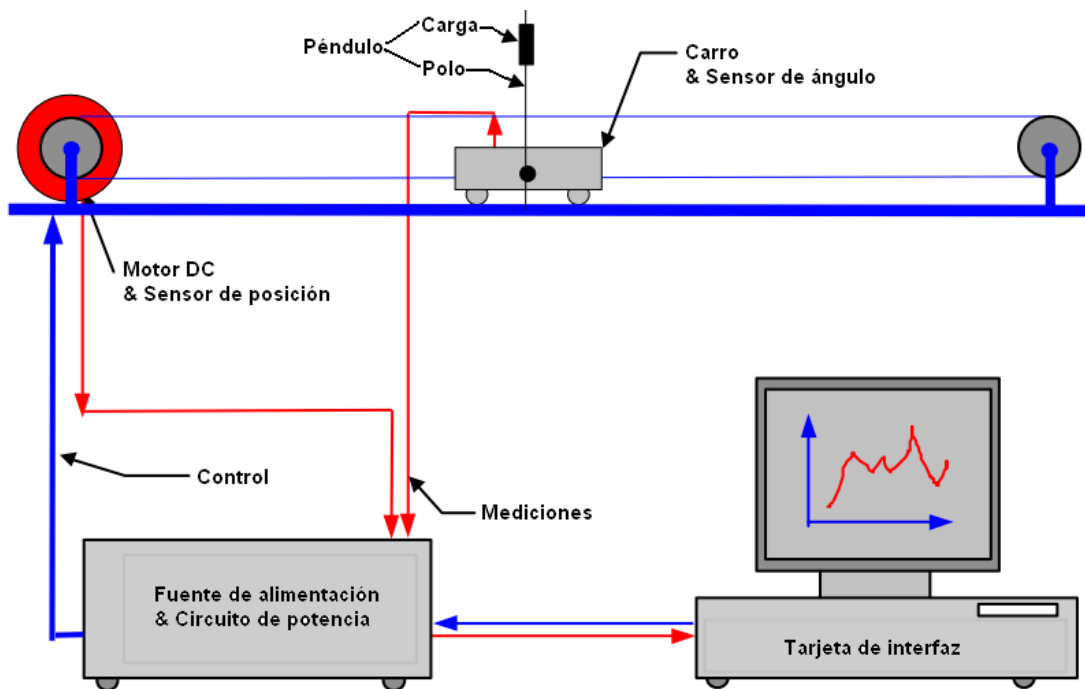


Figura. 2. 1. Componentes del sistema carro - péndulo

Este sistema trabaja de tal manera que el péndulo montado en el carro pueda oscilar libremente en el plano vertical. El carro se desliza sobre un riel de longitud limitada mediante ruedas, impulsado por un motor de corriente continua a través de una banda de transmisión unida a dos poleas como se muestra en la Figura. 2. 2. Para balancear y equilibrar el péndulo se empuja el carro de un lado a otro sobre este riel, la posición del motor se mide con un sensor y esta información es enviada mediante el circuito de potencia a la tarjeta de interfaz para luego pasar al ordenador para realizar su debido control de posición.

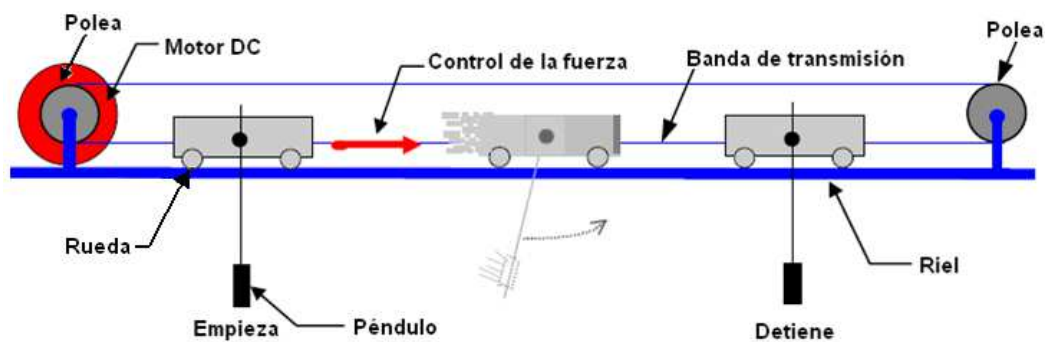


Figura. 2. 2. Sistema de control del péndulo

El sistema carro - péndulo es una aplicación de control de tipo abierto, es decir el usuario desarrolla en Simulink el bloque de control y lo conecta al modelo de experimentación. Con ello se tiene acceso al control y a la medición de las variables de posición del carro y del péndulo.

El sistema carro-péndulo realiza la transformación de las señales de control a una variación de la señal de voltaje correspondiente y para ello utiliza la tarjeta de interfaz que

aplica la técnica PWM (Modulación por Ancho de Pulso), que es la generación de pulsos cuadrados con modificación de su ciclo de trabajo controlando de esta manera la cantidad de energía que se envía al proceso^[16].

La posición y ángulo son medidos e ingresados al entorno de trabajo de Matlab mediante la tarjeta de adquisición RT-DAC4/PCI-D equipado con un XILINX® Spartan-II FPGA dedicado al diseño lógico. La Figura. 2. 3., muestra detalles de la configuración del hardware de control para el sistema carro-péndulo.

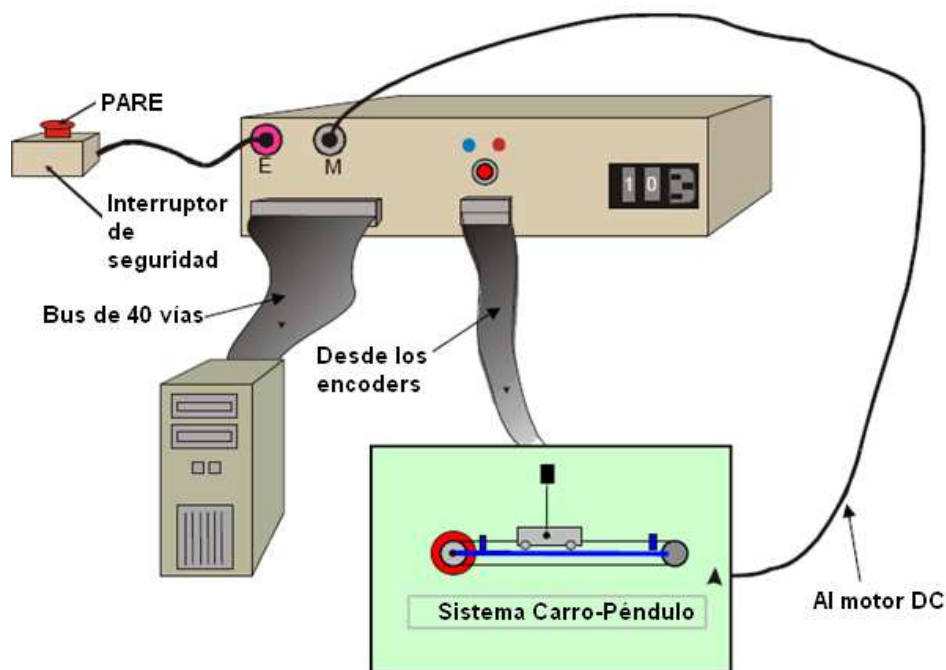


Figura. 2. 3. Configuración de Hardware del sistema carro - péndulo

El *software* de control para el sistema carro-péndulo está desarrollado en Matlab específicamente en la caja de herramientas definido como “p1”, misma que tiene una

^[16] Díaz, Juan. *Inversores pwm*, Tecnología Electrónica, Universidad de Oviedo. 1999. (13)

colección de funciones M, modelos MDL y código C, archivos DLL que producen un ambiente de trabajo para desarrollar diseños y problemas de control.

Es así que Matlab trabaja con los siguientes componentes: Real Time Windows Target (RTWT) que es un componente de la caja de herramientas de Matlab que permite hacer simulaciones en tiempo real es decir realizar visualizaciones de señales con el bloque de osciloscopios de Simulink, modificación de parámetros y control en tiempo real. El Real Time Workshop (RTW) que es una parte integral del entorno de Simulink para generar y ejecutar archivos en código C para desarrollar y aprobar los algoritmos modelados en Simulink y códigos en Matlab^[17]. Cabe señalar que el RTW es un componente utilizado por el RTWT, siendo este último diseñado para operar con gran flexibilidad y rapidez durante la creación de modelos.

El software integrado en este sistema soporta todas las fases del desarrollo de un sistema de control:

- Proceso de identificación en línea,
- Modelamiento de sistema de control, diseño y simulación,
- Implementación de algoritmos de control.

Las herramientas del sistema carro-péndulo proveen al usuario una variedad de capacidades de software:

^[17] Ortíz, Floriberto. *Modelado y control PD-difuso en tiempo real para el sistema barra-esfera*. Departamento de control automático, Centro de investigación y de estudios avanzados del instituto politécnico nacional. 2004. (14)

- Información que puede fluir entre el proceso y el ambiente de Matlab en línea,
- Experimentar el control en tiempo real de los algoritmos demostrativos,
- Desarrollo, simulación y aplicación de algoritmos de control definidos por el usuario.

Las herramientas del sistema carro-péndulo vienen incluidas en el software y manuales del péndulo.

2.1.1 Descripción del hardware

Los componentes mecánicos y electrónicos que conforman el sistema carro-péndulo son:

- Péndulo (carga + polo),
- Carro,
- Fuente de alimentación,
- Circuito de potencia,
- Tarjeta de interfaz RT-DAC4/PCI-D adquisición y control I/O,
- Sensor de posición lineal,
- Sensor de posición angular,
- Motor DC,
- Interruptor de seguridad,

2.1.1. Fuente de alimentación

La fuente de alimentación es el dispositivo encargado de convertir la tensión alterna de 115 VAC disponible en el toma corriente a 24 VDC. Con este voltaje de corriente directa permite el funcionamiento del ventilador y el circuito de potencia que muestran en la Figura. 2. 4.

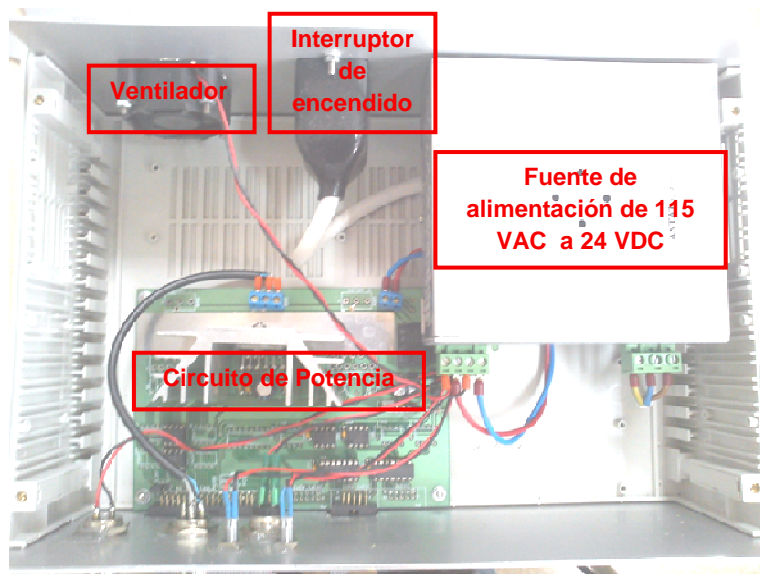


Figura. 2. 4. Fuente de alimentación

2.1.1. Circuito de potencia

Para encender el circuito de potencia se debe en primera instancia presionar el interruptor de encendido que se encuentra en la parte posterior de la caja logrando con ello habilitar la fuente de alimentación que a la vez enciende el indicador azul, para luego presionar el pulsador de encendido de este circuito de potencia que a la vez enciende el indicador rojo.

El circuito de potencia permanece encendido hasta que se lo apague desde el interruptor de encendido o al presionar el interruptor de seguridad que al mismo tiempo detiene al motor DC y apaga el indicador rojo dejando encendido el indicador azul es decir únicamente la fuente de alimentación.

El circuito de potencia es el encargado de elevar el nivel de la señal de control que se transmite desde el computador para hacer trabajar la bobina del motor a 22.5 [V]. Esta etapa de potencia es controlada por el LMD18200T que es un Puente-H de 3[A] diseñado para aplicaciones de control de movimiento tal como se muestra en la Figura. 2. 5., está construido utilizando tecnología multiproceso que combina circuitería bipolar e ideal para la conducción de los motores DC y motores paso a paso.

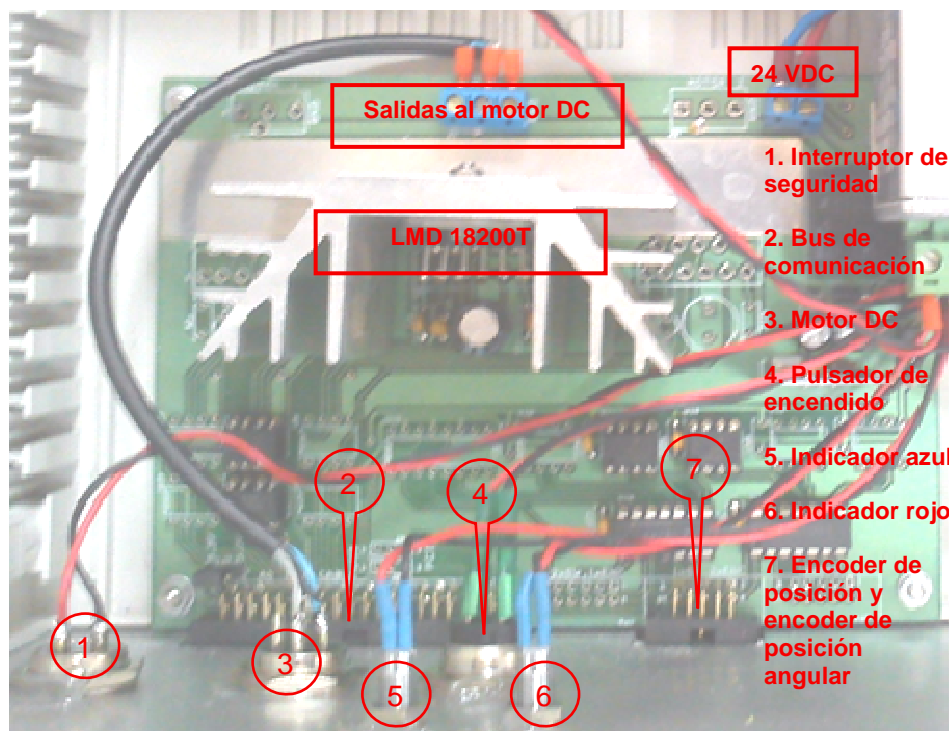


Figura. 2. 5. Circuito de potencia

Un esquema general de la etapa de potencia que se utiliza para girar al motor DC se muestra en la Figura. 2. 6.

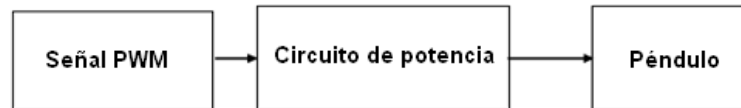


Figura. 2. 6. Esquema de la etapa de potencia del sistema carro – péndulo

2.1.1. Tarjeta de interfaz RT-DAC4/PCI-D

La tarjeta de interfaz de adquisición de datos es la denominada “RT-DAC4/PCI-D” con multifuncionalidad analógica y digital de I/O, dedicada a la adquisición de datos en tiempo real y control bajo un ambiente Windows 2000/XP. La tarjeta utiliza un bus PCI (3.3V y 5.5V) y soporta operaciones en tiempo real aún con los retardos causados por Windows, algunas de sus características para entradas/salidas son:

- *Entradas analógicas:* 16 canales multirango multiplexadas, resolución de 12 bits, rango de entrada $\pm 10V$, ganancias programables de (x1, x2, x4, x8, x16). Tiempo de conversión 1,6 [us].
- *Salidas analógicas:* 4 canales, resolución de 12 bits, tiempo de establecimiento de 6 [us], rango de salida 0V a +10V, -10V a 0V, $\pm 10V$.
- *Entradas y salidas digitales:* 32 canales bidireccionales, se direccionan individualmente vía software, voltaje de entrada para niveles altos $V_{IH} = 2,0V$ a

3,6V, para niveles bajos $V_{IL} = -0,5V$ a $0,8V$, voltaje de salida: $V_{OH} = 2,4V$ (min), $V_{OL} = 0,4V$ (máx.), corriente de salida en el rango de 2mA a 24mA por canal.

- 4 canales de salidas PWM, con resolución de 8/12 bits seleccionado por software.
- 4 canales de entrada para encoders incrementales, los cuales transforman el movimiento angular a una serie de impulsos digitales^[18].

El esquema interior de la tarjeta RT-DAC4/PCI-D se presenta en la Figura. 2. 7.

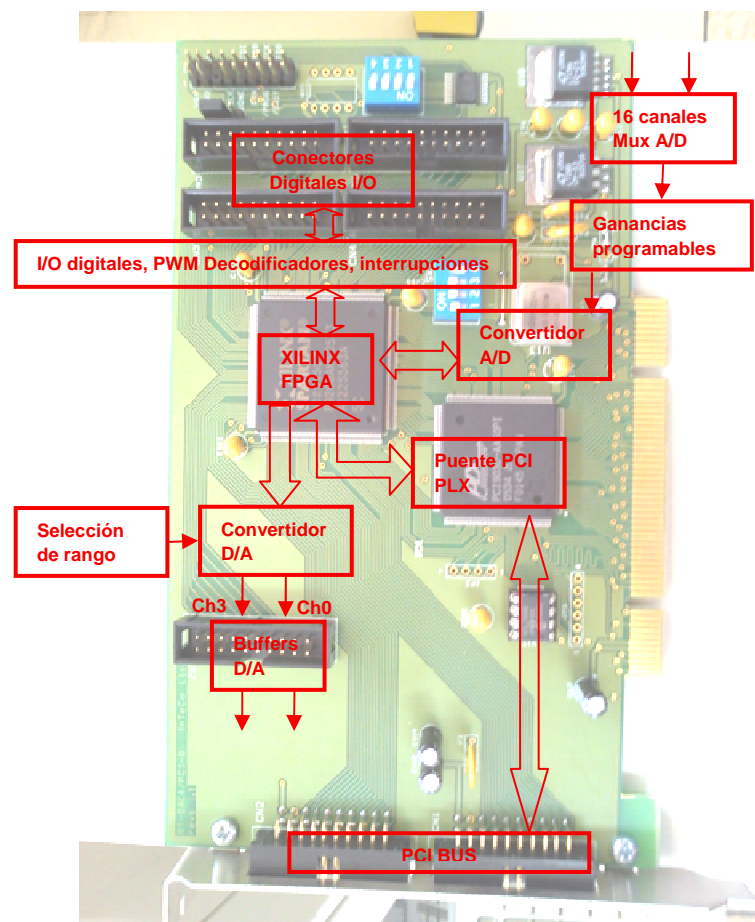


Figura. 2. 7. Esquema interior de la tarjeta RT-DAC4/PCI-D

^[18] Cruz, Agustín. *Multiplicador electrónico para encoder incremental*. 2008. (15)

La tarjeta contiene un multiplexor de entrada analógica conectada a 16 canales unipolares, el rango de voltaje está definido desde -10V a 10 V.

La tarjeta está equipada con un convertidor A/D de 12 bits de aproximaciones sucesivas que entrega una resolución de 5mV con un rango de entrada de $\pm 10V$. El tiempo de conversión del convertidor A/D de la tarjeta RT-DAC4/PCI-D es igual a 1,6 μ s.

La tarjeta contiene cuatro convertidores D/A de 12 bits, conectado a cuatro canales de salida analógicos, los canales pueden ser configurados por hardware, cada canal de salida analógica puede entregar hasta 10mA, existen 32 líneas digitales de I/O en la tarjeta RT-DAC4/ PCI-D, las direcciones pueden ser configuradas independientemente. La tarjeta RT-DAC4/PCI-D está equipada con convertidores A/D paralelos de 12 bits.

2.1.1. Sensor de posición

Para sensar la posición del carro y el ángulo que describe el péndulo se usan un encoder óptico incremental el mismo que se encarga de enviar los datos a través de la tarjeta de interfaz RT-DAC4/PCI-D como se observa en la Figura. 2. 8.

El funcionamiento se basa en la rotación de un disco perforado. Las perforaciones forman dos anillos concéntricos con diferencia en el radio que los describe. Un haz de luz emitida por una fuente LED pasa a través del disco perforado hacia un receptor LED que lo traduce a dos señales eléctricas con formas de onda rectangulares con desfase de 45° como se muestra en la Figura. 2. 9.

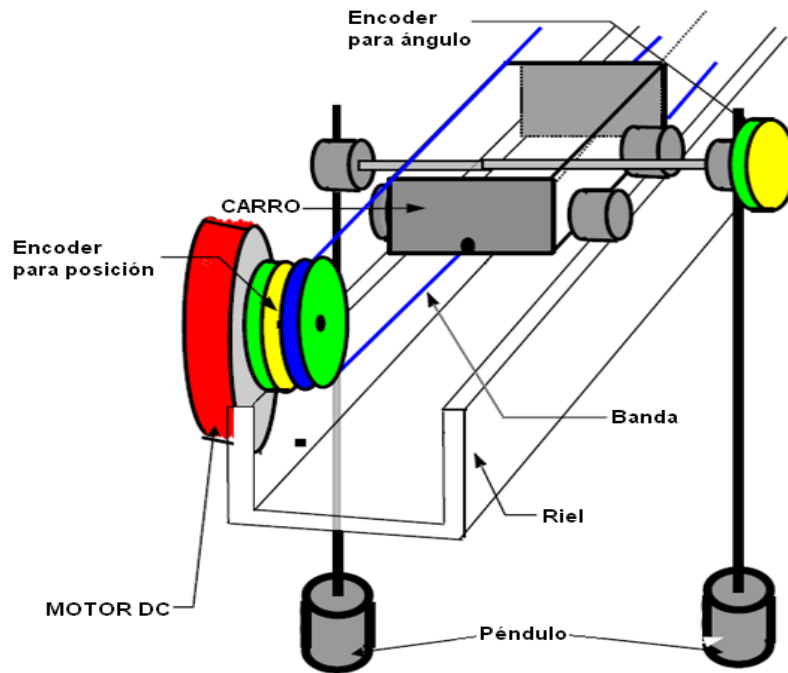


Figura. 2. 8. Partes mecánicas y sensores en la configuración carro – péndulo

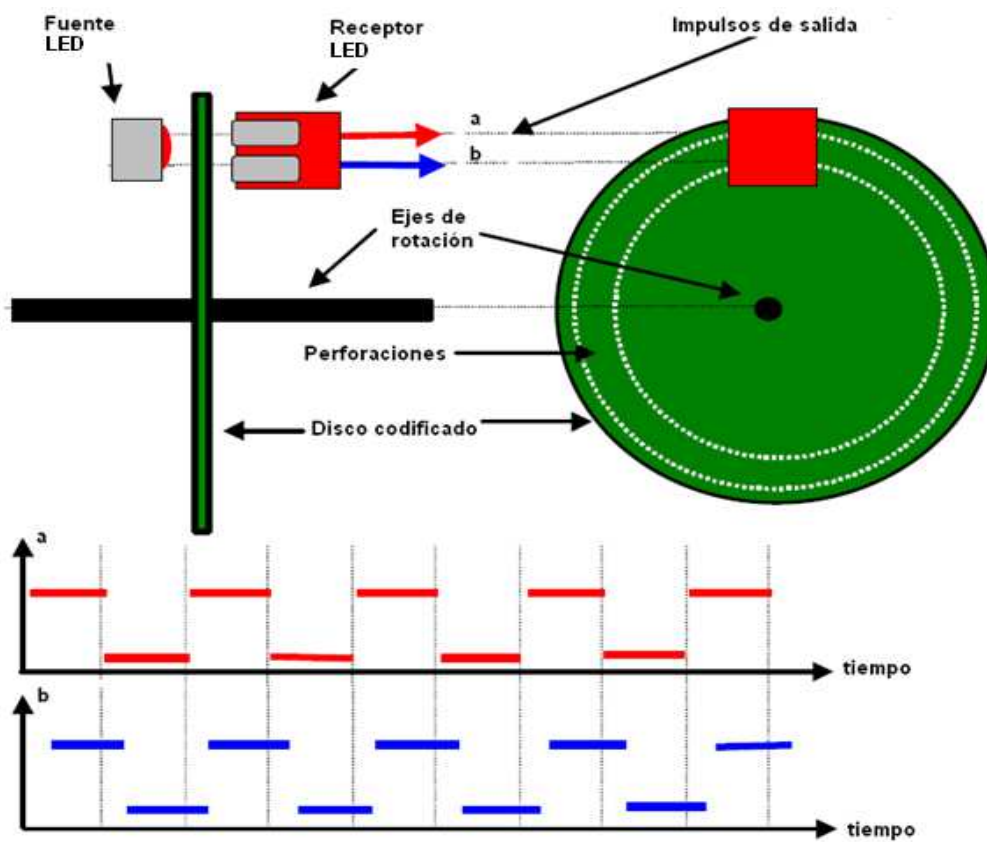


Figura. 2. 9. Señales eléctricas producidas por el disco codificado del encoder incremental

Codifica información correspondiente a la dirección de giro ya sea en sentido horario o anti horario, de manera que se requiere ambas señales como se muestra en la Figura. 2. 10., se tiene un circuito a través del cual se determina el sentido del desplazamiento.

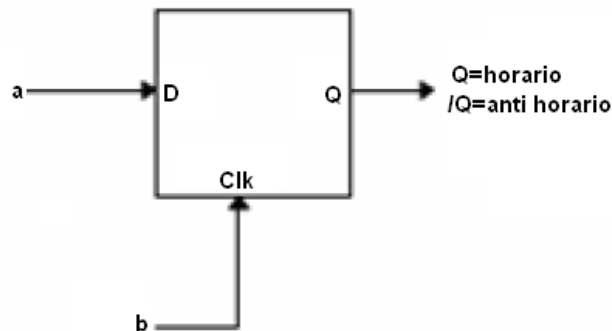


Figura. 2. 10. Determinación del sentido de giro

Se implementa con un Flip-Flop “D”, la señal “a” se emplea como dato y “b” como señal de reloj, en el sentido horario (izquierda a derecha) se captura continuamente un nivel alto, esto, porque el flanco de subida de “b” coincide con el nivel alto de “a”. Para el sentido anti horario (derecha a izquierda) el flanco de subida de “b” coincide ahora con el nivel bajo de “a”.

También codifica información correspondiente al desplazamiento realizado obteniéndose directamente de cualquiera de las dos señales eléctricas, un ciclo de la señal corresponde al mínimo avance, se puede usar como referencia el flanco de subida o bajada en este caso el mínimo avance corresponde a $360^\circ/4096=0.087^\circ$.

Las características del encoder incremental son: señal de salida de -0.5 a 5 VDC con una resolución de 4096 cuentas por revolución. La alimentación eléctrica es de -0.5 a 7 VDC, cuenta con 5 pines (canal A, canal B, VDC, tierra, sentido de rotación).

2.1.1. Motor DC

Para mover el carro sobre el riel se usa un motor DC el mismo que se encarga de balancear el péndulo sobre el plano vertical. Es un motor de corriente continua de 94 [W] totalmente cerrado en un diseño ultra delgado con un perfil de panqueque como se muestra en la Figura. 2. 11., usa imanes permanentes en forma de panqueques de alta potencia, proporcionando un mayor rendimiento, inercia baja en la rotación, operación uniforme, regulación amplia de velocidad.



Figura. 2. 11. Motor DC

Tiene como características una señal para trabajar de 22.5 VDC, con un torque de 30 [Ncm] y una velocidad de 3000 RPM.

2.1.2 Descripción del software

El software del sistema carro-péndulo es versátil, permite diseñar nuevos sistemas de control con Simulink y aplicarlos en tiempo real como se muestra en la Figura. 2. 12.

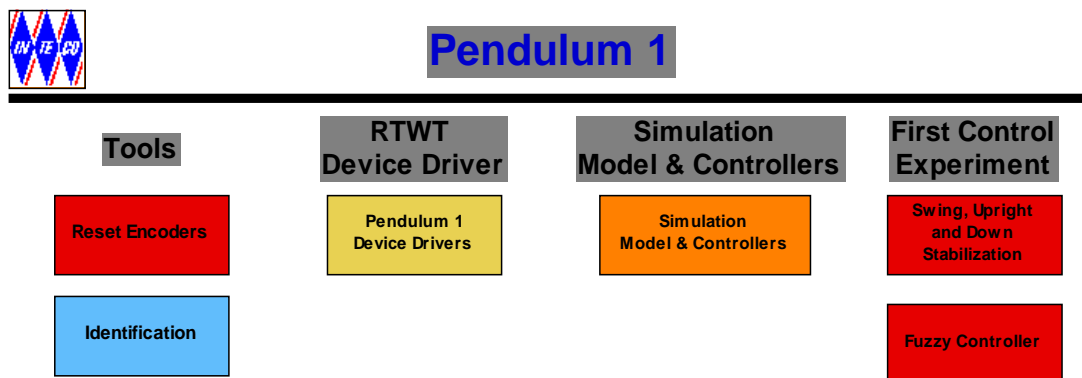


Figura. 2. 12. Software carro-péndulo

El software contiene herramientas para:

- Restablecer los encoders

Ajusta los registros de medición de los encoders del sistema carro-péndulo a un valor cero. Toma la nueva posición como punto de referencia.

- Identificación

Es una herramienta importante en el sistema basado en la medición de datos en forma experimental de entradas y salidas. Permite obtener el valor de la frecuencia

de amortiguamiento del péndulo como parámetro más elemental dentro del funcionamiento del sistema, para luego en base a este parámetro lograr caracterizar al modelo matemático de simulación.

Además permite conocer las acciones que realiza el sistema al enviar órdenes a través del computador y también observar las mediciones de los sensores de posición.

- Simulación

El software dedicado para este sistema posee el modelo matemático dinámico de simulación para el computador el mismo que permite simular el funcionamiento real del sistema. Permitirá su funcionamiento en MODO GRÚA o MODO PÉNDULO INVERTIDO.

Además trae consigo el modelo demostrativo de control para el sistema en MODO GRÚA y MODO PÉNDULO INVERTIDO realizado por el fabricante.

- Experimentación

Esta herramienta contiene el controlador del dispositivo (*device driver*) como parte esencial para poder interactuar entre el computador y el sistema real. Permitirá su funcionamiento en MODO GRÚA o MODO PÉNDULO INVERTIDO.

Además trae consigo el modelo demostrativo de control para el sistema en MODO GRÚA Y MODO PÉNDULO INVERTIDO realizado por el fabricante.

2.2 MODELO MATEMÁTICO

El modelo matemático representa al modelo físico de este sistema, describiendo matemáticamente el comportamiento dinámico del mismo. Este modelo dinámico está expresado en ecuaciones diferenciales de primer orden las mismas que están modeladas en Simulink de Matlab. Con este modelo se puede analizar el comportamiento del sistema o bien predecir su comportamiento.

Existen algunos modelos matemáticos que describen al sistema carro – péndulo ya que depende de la perspectiva que se la realice. Por tal motivo se considera el modelo matemático no lineal, al igual que sus parámetros de ingreso que se muestran en el manual de usuario, mismos que han sido obtenidos en trabajos previos sobre el sistema.

Los parámetros que corresponden al modelo matemático original del sistema carro - péndulo han sido obtenidos por el fabricante, se muestran en la Tabla. 2. 1. A estos se los puede tomar como referencia al momento de realizar el proceso de identificación de manera experimental ya que pueden variar en equivalencia. Las variaciones de los parámetros dependen del desempeño físico del sistema.

Nombre	Descripción	Unidad
m	Masa equivalente al carro y al péndulo	0.872 [kg]
l	Distancia desde el centro de masa del péndulo al eje de rotación	0.011 [m]
f_c	Coefficiente de fricción dinámico del carro	0.5 [N.s/m]
f_s	Fricción estática del carro	1.203 [N]
f_p	Coefficiente de fricción del péndulo	$6.65 \cdot 10^{-5}$ [N.m.s/rad]
J_p	Momento de inercia del péndulo con respecto al eje de rotación	0.00292 [kg.m ²]
g	Gravedad	9.81 [m/s ²]
p_1	Fuerza de control relacionada a la señal PWM	9.4 [N]
p_2	Fuerza de control relacionada a la velocidad del carro	-0.548 [N.s/m]
u_{max}	Máximo valor de la señal PWM	0.5
m_c	Masa equivalente al carro	0.768 [kg]
m_{ps}	Masa del polo	0.038 [kg]
m_{pw}	Masa de la carga	0.014 [kg]
R_l	Longitud del riel	1.8 [m]
l_p	Longitud del polo	0.5 [m]
l_{po}	Distancia entre el centro de masa del polo y el eje de rotación del péndulo	0.1075 [m]
l_c	Longitud de la carga	0.03 [m]
l_{co}	Distancia entre el centro de masa de la carga y el eje de rotación del péndulo	0.3540 [m]
T	Período del péndulo	1.17 [s]
J	Momento de inercia relacionado al centro de masa del péndulo	0.00282 [kg.m ²]

Tabla. 2. 1. Parámetros del modelo matemático tomados por el fabricante

Ahora los parámetros presentes en las ecuaciones que gobiernan el movimiento del sistema mecánico y que son ingresadas para la simulación del modelo matemático se muestran en la Tabla. 2. 2., los mismos que son obtenidas experimentalmente en la identificación a través del código que se muestra en la Tabla. 2. 3. numerado en código del 1 al 27. La variación de cualquiera de ellos influye en la simulación del modelo matemático.

Parámetro	Descripción	Valor
$[x1(0) x2(0) x3(0) x4(0)]$	Vector de estado inicial: posición carro, posición péndulo, velocidad carro, velocidad péndulo.	-0.93 [m], -pi [rad], 0 [m/s], 0 [rad/s]
mc, mp	Masa del carro, Masa del péndulo	1.0423 [kg], 0.104 [kg]
f_p, FS, FC	Coefficiente de fricción del péndulo, Fricción estática del carro, Coeficiente de fricción dinámica del carro	0.00004949 [N.m.s/rad], 1.63322 [N], 0.5 [N.s/m]
l, J, g	Distancia desde el centro de masa del péndulo al eje de rotación, Momento de inercia total, Gravedad	0.00826190 [m], 0.00290756 [kg.m ²], 9.81 [m/s ²]
M, DZ_u	Control: magnitud M (motor), DZ_u (señal de control mínima)	12.86 [N], 0.127
DZ_{cv}, DZ_{cp}	Velocidades mínimas: carro, péndulo	0.1 [m/s], 1.5 [rad/s]

Tabla. 2. 2. Parámetros del modelo matemático

	UNIDAD	CÓDIGO	DESCRIPCIÓN
<i>CONSTANTE</i>			
$g = 9.81$	$\frac{m}{s^2}$	No. 1	Gravedad
<i>MEDIDO</i>			
$m_1 = 0.768$	kg	No. 2	Masa del carro
$m_2 = 0.024$	kg	No. 3	Masa del péndulo
$m_{zl} = 0.1458$	kg	No. 4	Masa izquierda (motor DC + polea)
$m_{zp} = 0.0805$	kg	No. 5	Masa derecha (polea)
$m_c = m_1 + 2 * m_2 + m_{zl} + m_{zp}$	kg	No. 6	Masa izquierda (motor DC + polea)
$m_{ps} = 0.038$	kg	No. 7	Masa del polo
$m_{pw} = 0.014$	kg	No. 8	Masa de la carga
$m_p = 2(m_{ps} + m_{pw})$	kg	No. 9	Masa total del péndulo
$l_p = 0.5$	m	No. 10	Longitud del polo
$r_p = 0.00305$	m	No. 11	Radio del polo
$l_c = 0.019$	m	No. 12	Longitud de la carga
$r_c = 0.0060$	m	No. 13	Radio de la carga
$l_c = 0.019$	m	No. 14	Longitud de la carga
$l_{co} = 0.3540$	m	No. 15	Longitud desde el centro de masa de la carga al eje de rotación del péndulo
$l_{po} = 0.1075$	m	No. 16	Longitud desde el centro de masa

del polo al eje de rotación del
péndulo

IDENTIFICADO

$M = 12.86$	N	No. 17	Magnitud de control
$FS_{pwm} = 0.127$	ciclo PWM	No. 18	Mínimo
$FS = FS_{pwm} * M$	N	No. 19	Fricción estática
$FC = 0.5$	$\frac{N.s}{m}$	No. 20	Fricción dinámica
$DZ_u = FS_{pwm}$	ciclo PWM	No. 21	Mínimo
$DZ_{cv} = 0.1$	$\frac{m}{s}$	No. 22	Velocidad mínima del carro
$DZ_{cp} = 1.5$	$\frac{rad}{s}$	No. 23	Velocidad mínima del péndulo
<i>CALCULADO</i>			
$J_p = mpw * (\frac{1}{12} * l_c^2 + \frac{1}{4} * r_c^2 + l_{co}^2 + \frac{1}{4} * r_p^2) + m_{ps} * (\frac{1}{12} * l_p^2 + l_{po}^2)$	$kg.m^2$	No. 24	Momento de inercia del péndulo con respecto al eje de rotación
$f_p = J_p * f_{pc}$	$\frac{N.m.s}{rad}$	No. 25	Coefficiente de fricción del péndulo
$l = \frac{l_{po} * m_{ps} + l_{co} * m_{pw}}{m_c + m_{ps} + m_{pw}}$	m	No. 26	Distancia desde el centro de masa del péndulo al eje de rotación
$J = J_p - l * l * (m_c + mp)$	$kg.m^2$	No. 27	Momento de inercia relacionado al centro de masa del péndulo

Tabla. 2. 3. Código para presentación de parámetros identificados

Las ecuaciones que rigen al modelo matemático, se basan en el esquema que se muestra en la Figura. 2. 13. Está respaldado en principios teóricos de las operaciones que ocurren en el proceso es decir es un MODELO PARAMÉTRICO. Tomando como partida aquello se considera cuatro ecuaciones fundamentales que van a constituir el modelo matemático, basadas en la *traslación* del centro de masa del sistema y la

rotación de todo el sistema alrededor del centro de masa como se expresa en las ecuaciones 2.1, 2.3 y 2.2, 2.4 respectivamente.

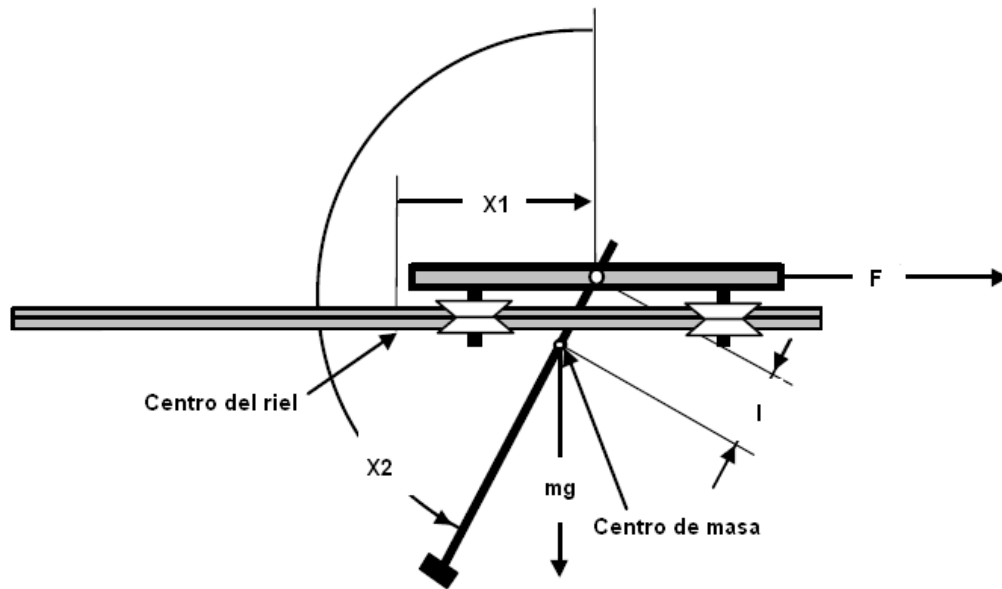


Figura. 2. 13. Esquema para el modelo matemático sistema carro-péndulo

Se considera girar al péndulo en un plano vertical alrededor de un eje situado en el carro, el mismo que puede moverse a lo largo del riel horizontal, situado en el plano de rotación. El estado del sistema es un vector:

$$x = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix}$$

- x_1 , es la *posición del carro*
- x_2 , es el *ángulo entre la posición vertical del péndulo y el péndulo en ese instante*, medido en sentido antihorario ($x_2 = 0$ para la posición vertical del péndulo)
- x_3 , es la *velocidad del carro*

- x_4 , es la *velocidad angular del péndulo*
- F , *fuerza de control* aplicada al carro paralela al riel

Las ecuaciones diferenciales ordinarias de primer orden que constituyen el modelo matemático del sistema son las siguientes^[19]:

$$\dot{x}_1 = x_3 \quad (2.1)$$

$$\dot{x}_2 = x_4 \quad (2.2)$$

$$\dot{x}_3 = \frac{a_1 w_1(x, u) + w_2(x) \cos x_2}{d(x)} \quad (2.3)$$

$$\dot{x}_4 = \frac{w_1(x, u) \cos x_2 + a_2 w_2(x)}{d(x)} \quad (2.4)$$

$$|u(t)| \leq u_{max}$$

Donde,

$u(t)$, señal de control

u_{max} , magnitud de control máxima

$$w_1(x, u) = k_1 u - x_4^2 \sin x_2 - k_2 x_3$$

$$w_2(x) = g \sin x_2 - k_3 x_4$$

$$d(x) = b - \cos^2 x_2$$

^[19] Turnau, Korytowski, Szymkat. *Time optimal control for the pendulum-cart system in real time*. Departamento de automatic, AGH. 1999. (16)

$$a_1 = \frac{J_p}{ml}, a_2 = \frac{1}{l}, b = a_1 a_2 = \frac{J_p}{ml^2}$$

$$k_1 = \frac{p_1}{ml}, k_2 = \frac{f_c - p_2}{ml}, k_3 = \frac{f_p}{ml}$$

La fuerza F es generada por el motor DC, controlada por una señal de voltaje u con modulación de pulso (PWM), donde $F = p_1 u + p_2 x_3$.

p_1 , es la fuerza de control relacionada a la señal PWM

p_2 , es la fuerza de control relacionada a la velocidad del carro

u , es la señal de control

El software de Matlab controla la variable u tomando valores en el intervalo $[-0.2, 0.2]$. La masa total equivalente al péndulo y al carro es nombrada por m . l es la distancia desde el eje de rotación del péndulo al centro de masa del sistema. J_p es el momento de inercia del péndulo con respecto a su eje de rotación en el carro.

Cabe señalar que dentro del modelo matemático también se incluye la **fricción** del carro que está compuesta de dos fuerzas: la fricción estática y la fricción dinámica proporcional a la velocidad del carro $f_c x_3$. También una fricción por torque en el movimiento angular del péndulo, proporcional a la velocidad angular $f_p x_4$.

2.2.1 Cálculo del momento de inercia

El péndulo de este sistema como tal está suspendido verticalmente de un eje horizontal, alrededor del cual puede oscilar por acción de la gravedad. Si se desplaza un pequeño ángulo desde su posición de equilibrio y se lo suelta, el péndulo oscila con movimiento armónico simple. Por tal motivo se genera un momento de inercia respecto al eje de rotación en el que se da a conocer la importancia de la masa en el movimiento de rotación del péndulo alrededor de ese eje.

La Figura. 2. 14. que se muestra a continuación se usa para ilustrar los parámetros involucrados en el cálculo del momento de inercia del péndulo con respecto a su eje de rotación en el carro.

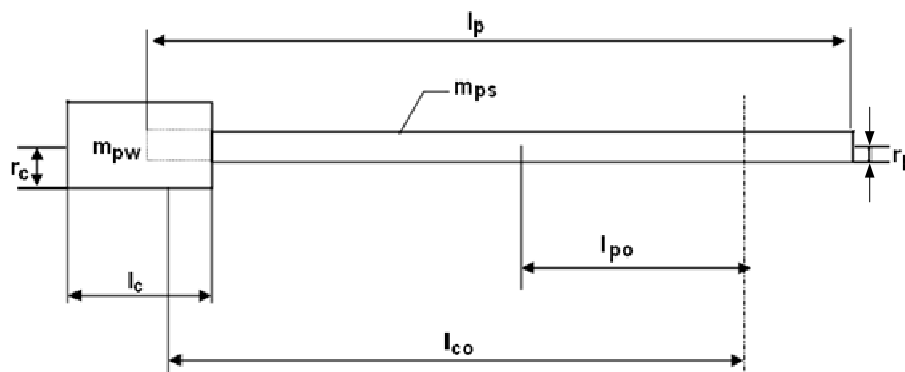


Figura. 2. 14. Parámetros en el cálculo del momento de inercia del péndulo

Para el desarrollo se usa la siguiente notación:

- m_{pw} , masa de la carga [kg]
- m_{ps} , masa del polo [kg]
- l_p , longitud del polo [m]
- l_{po} , distancia entre el centro de masa del polo y el eje de rotación del péndulo [m]
- l_c , longitud de la carga [m]
- l_{co} , distancia entre el centro de masa de la carga y el eje de rotación del péndulo [m]
- r_p , radio del polo [m]
- r_c , radio de la carga [m]
- m_c , masa equivalente al carro [kg]

El momento de inercia del péndulo respecto al eje de rotación es el siguiente:

$$J_p = \frac{1}{12} m_{pw} l_c^2 + \frac{1}{4} m_{pw} r_c^2 + m_{pw} l_{co}^2 + \frac{1}{12} m_{ps} l_p^2 + \frac{1}{4} m_{pw} r_p^2 + m_{ps} l_{po}^2 \quad (2.5)$$

Aplicando el teorema de *Steiner* se determina el momento de inercia J . Es decir se determina el momento de inercia J del péndulo respecto la distancia l desde el centro de masa del péndulo.

J puede ser expresado por J_p

$$J = J_p - l^2(m_c + m_p) \quad (2.6)$$

La distancia desde el centro de masa del péndulo al eje de rotación es igual a:

$$l = \frac{l_{po}m_{ps} + l_{co}m_{pw}}{m_c + m_{ps} + m_{pw}} \quad (2.7)$$

2.3 DESCRIPCIÓN DEL MODELO MATEMÁTICO DESARROLLADO EN SIMULINK DE MATLAB ®

El software dedicado para el sistema carro – péndulo posee un bloque dinámico que es donde se encuentra desarrollado el modelo matemático como se muestra en la Figura. 2. 15., mismo que es de gran utilidad para realizar simulaciones con nuevos controladores.

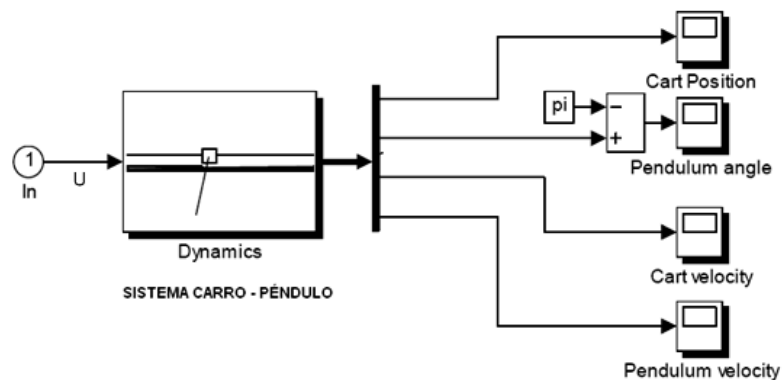


Figura. 2. 15. Bloque dinámico del sistema carro-péndulo

A partir de este bloque se pueden realizar nuevas pruebas de control. Funciona de modo que la señal cuadrada de control “U” ingresa al modelo dinámico por el puerto de entrada “1”, pasa por bloque dinámico, luego cuatro señales se muestran en cuatro

visualizadores para: la posición del carro, la posición angular del péndulo, la velocidad del carro y la velocidad angular del péndulo.

Abriendo la máscara del bloque dinámico se observa todo el tratamiento realizado para implantar el modelo matemático regido por las cuatro ecuaciones diferenciales mencionadas anteriormente, además de varios bloques de la biblioteca de Simulink que están dedicados para el sistema carro - péndulo: *Zona muerta y Saturación (1)*, y la *Fricción (2)* como se muestra en la Figura. 2. 16. Un *bloque de memoria (3)* está presente para evitar generación de lazos algebraicos. Por otro lado existen cuatro bloques integradores que permiten resolver las ecuaciones para dar a conocer: la posición del carro, la velocidad del carro, la posición angular del péndulo y la velocidad angular del péndulo.

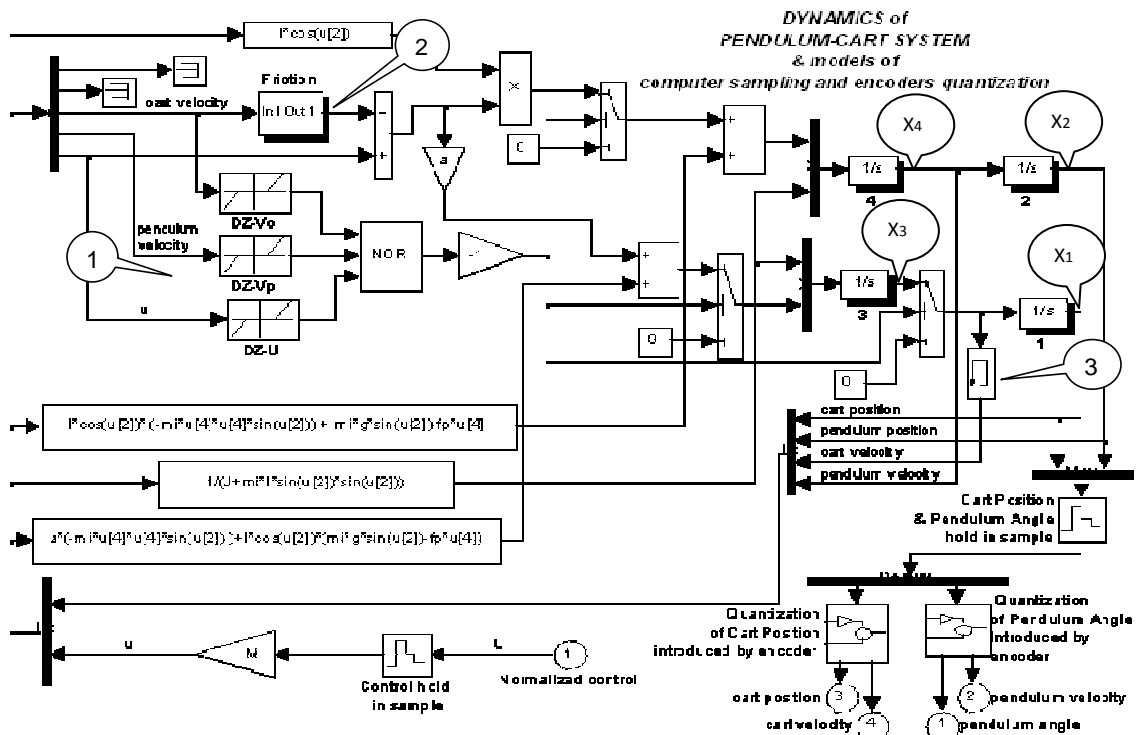


Figura. 2. 16. Interior del bloque dinámico del sistema carro-péndulo

Para representar al Motor DC usualmente no se usa la máxima fuerza M generada por el motor DC^[20]. Por lo tanto se reduce la fuerza U proporcionalmente ($0 \leq U \leq 1$). La señal de control mediante el bloque “*hold in sample*” se discretiza con un tiempo de muestreo igual a 0.01 [s]. Luego esta señal pasa a tomar una ganancia “ M ” igual a 12.8 que es la magnitud de control.

Se usa $U_{max} = 0.2$ es decir solo el 20% de la potencia nominal del motor DC, con esto pasa a controlar el modelo dinámico.

Para ubicar las ecuaciones en el modelo de Simulink se procede a reemplazar los valores que corresponden en las ecuaciones 2.3 y 2.4 para poderlos ubicar de manera sencilla.

Reemplazando a_1 , $w_1(x, u)$, $w_2(x)$, $d(x)$, k_1 , k_2 , k_3 , b y J_p en la ecuación 2.3 se tiene:

$$\dot{x}_3 = \frac{a_1(uk_1 - x_4^2 \sin x_2 - k_2 x_3) + (g \sin x_2 - k_3 x_4) \cos x_2}{b - \cos^2 x_2}$$

$$\dot{x}_3 = \frac{a_1 \left[u \left(\frac{p_1}{ml} \right) - x_4^2 \sin x_2 - \left(\frac{f_c - p_2}{ml} \right) x_3 \right] + \left[g \sin x_2 - \left(\frac{f_p}{ml} \right) x_4 \right] \cos x_2}{\frac{J + l^2(m_c + m_p)}{ml^2} - \cos^2 x_2}$$

$$\dot{x}_3 = \frac{a_1 \left[u \left(\frac{p_1}{ml} \right) - x_4^2 \sin x_2 - \left(\frac{f_c - p_2}{ml} \right) x_3 \right] + \left[g \sin x_2 - \left(\frac{f_p}{ml} \right) x_4 \right] \cos x_2}{\frac{J + ml^2}{ml^2} - \cos^2 x_2}$$

^[20] Inteco, PENDULUM – CART SYSTEM User’s Manual. 1997. (1)

$$\dot{x}_3 = \frac{la_1(up_1+x_3p_2-f_c x_3)+a_1(-ml^2x_4^2\sin x_2)+l\cos x_2(gml\sin x_2-f_p x_4)}{J+ml^2\sin^2 x_2}$$

$$\dot{x}_3 = \frac{a(up_1+x_3p_2-f_c x_3)+a(-mlx_4^2\sin x_2)+l\cos x_2(gml\sin x_2-f_p x_4)}{J+ml^2\sin^2 x_2} \quad (2.8)$$

Reemplazando $w_1(x, u)$, a_2 , $w_2(x)$, $d(x)$, k_1 , k_2 , k_3 , b y J_p en la ecuación 2.4 se tiene:

$$\dot{x}_4 = \frac{uk_1\cos x_2 - x_4^2\sin x_2\cos x_2 - x_3k_2\cos x_2 + a_2g\sin x_2 - a_2k_3x_4}{b - \cos^2 x_2}$$

$$\dot{x}_4 = \frac{u\left(\frac{p_1}{ml}\right)\cos x_2 - x_4^2\sin x_2\cos x_2 - x_3\left(\frac{f_c - p_2}{ml}\right)\cos x_2 + \left(\frac{1}{l}\right)g\sin x_2 - \left(\frac{1}{l}\right)\left(\frac{f_p}{ml}\right)x_4}{\frac{J_p}{ml^2} - \cos^2 x_2}$$

$$\dot{x}_4 = \frac{l\cos x_2(up_1+x_3p_2-f_c x_3)+l\cos x_2(-mlx_4^2\sin x_2)+mlg\sin x_2-f_p x_4}{J_p - ml^2\cos^2 x_2}$$

$$\dot{x}_4 = \frac{l\cos x_2(up_1+x_3p_2-f_c x_3)+l\cos x_2(-mlx_4^2\sin x_2)+mlg\sin x_2-f_p x_4}{J+l^2(m_c+m_p)-ml^2\cos^2 x_2}$$

$$\dot{x}_4 = \frac{l\cos x_2(up_1+x_3p_2-f_c x_3)+l\cos x_2(-mlx_4^2\sin x_2)+mlg\sin x_2-f_p x_4}{J+ml^2\sin^2 x_2} \quad (2.9)$$

- BLOQUE ZONA MUERTA Y SATURACIÓN

Formado por tres bloques “Dead Zone”, cada uno funciona de modo que:

- Si la entrada está dentro de la zona muerta (mayor que el límite inferior y menor que el límite superior), la salida es cero.
- Si la entrada es mayor o igual al límite superior, la salida es la entrada menos el límite superior.
- Si la entrada es menor o igual al límite inferior, la salida es la entrada menos el límite inferior.

Los tres bloques de *Zona muerta* logran sensibilizar al modelo es decir para trabajar con todo los valores de las señales: velocidad del carro, velocidad del péndulo y de control. El límite mínimo para cuando las señales sean cero está en el rango dado a continuación:

$DZ - V_c \rightarrow DZ_{cv}$ (velocidad mínima del carro)= está en el rango de [-0.1, 0.1] [m/s]

$DZ - V_p \rightarrow DZ_{cp}$ (velocidad mínima del péndulo)= está en el rango de [-1.5, 1.5] [rad/s]

$DZ - U \rightarrow DZ_u$ (señal de control mínima)= está en el rango de [-0.127, 0.127]

Posterior a aquello, las tres señales de salida de los bloques *Zona muerta* se comparan en el operador lógico NOR, activándolo cuando las tres señales sean cero. Luego este resultado adquiere una ganancia de -1. Todo este proceso se realiza con el fin de negar el paso de las expresiones correspondientes a las ecuaciones 2.8 y 2.9 y así poder inhabilitar el funcionamiento de la simulación. Este proceso es una condición de habilitación en tres *conmutadores* del diseño.

- BLOQUE FRICCIÓN

La nomenclatura usada en el módulo de “*fricción viscosa y de coulomb*” son las siguientes:

Coefficiente de fricción estática del carro $\rightarrow FS = 1.63322$. Es una fuerza que se opone al inicio del movimiento de un objeto por una superficie y depende de su peso y del tipo de materiales que están en contacto^[21].

Coefficiente de fricción dinámica del carro $\rightarrow FC = 0.5$. La fricción dinámica existe cuando un cuerpo ya se encuentra en movimiento, normalmente el coeficiente de fricción dinámico es más grande que el coeficiente de fricción estático^[22].

Bajo la máscara se tiene el siguiente esquema como se muestra en la Figura. 2. 17.

^[21] Ogata, Katsuhiko. *Ingeniería de Control Moderna*. 2002. (17)

^[22] Ogata, Katsuhiko. *Ingeniería de Control Moderna*. 2002. (17)

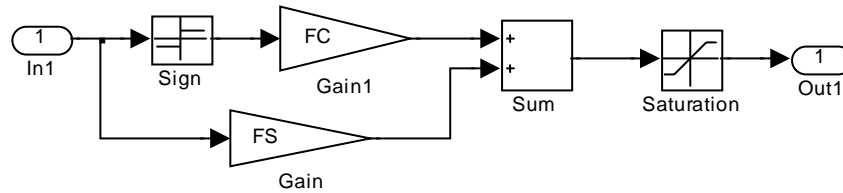


Figura. 2. 17. Interior del bloque fricción

El bloque “*sign*” funciona de modo que:

- La salida es 1 cuando la entrada es mayor que cero.
- La salida es 0 cuando la entrada es igual a cero.
- La salida es -1 cuando la entrada es menor que cero.

Como se había mencionado anteriormente la fricción del carro está compuesta de la fricción estática y la fricción dinámica. Entonces la señal velocidad del carro es evaluada en primera instancia por el bloque “*sign*” para mostrar el signo correspondiente para luego multiplicarse con el valor de FC, en segunda instancia se multiplica con el valor de FS. Luego los dos valores se suman, el resultado se normaliza en el bloque de *saturación* para que permanezca entre [-1, 1]. Todo este proceso se realiza con el fin de emular la fricción dinámica real del sistema.

Tras obtener las ecuaciones 2.8 y 2.9 en Simulink, se colocan cuatro bloques integradores para obtener las siguientes señales:

- Tercer bloque integrador entrega la señal correspondiente a la velocidad del carro.
- Primer bloque integrador entrega la señal correspondiente a la posición del carro.
- Cuarto bloque integrador entrega la señal correspondiente a la velocidad angular del péndulo.
- Segundo bloque integrador entrega la señal correspondiente a la posición angular del péndulo.

Estas señales permiten monitorear al proceso. En el modelo dinámico estas señales regresan continuamente al proceso y a su vez se muestran fuera de este bloque tras realizar un proceso de cuantificación.

2.4 CUANTIFICACIÓN Y TIEMPO DE MUESTREO EN EL SISTEMA DE ADQUISICIÓN

El bloque dinámico está equipado adicionalmente con el modelo del equipo de muestreo, el modelo cuantificador de la posición del carro y el modelo cuantificador de la posición angular del péndulo, para de este modo representar las condiciones reales de los encoders.

Abriendo la máscara de los dos bloques de cuantificación se muestra la Figura. 2. 18.

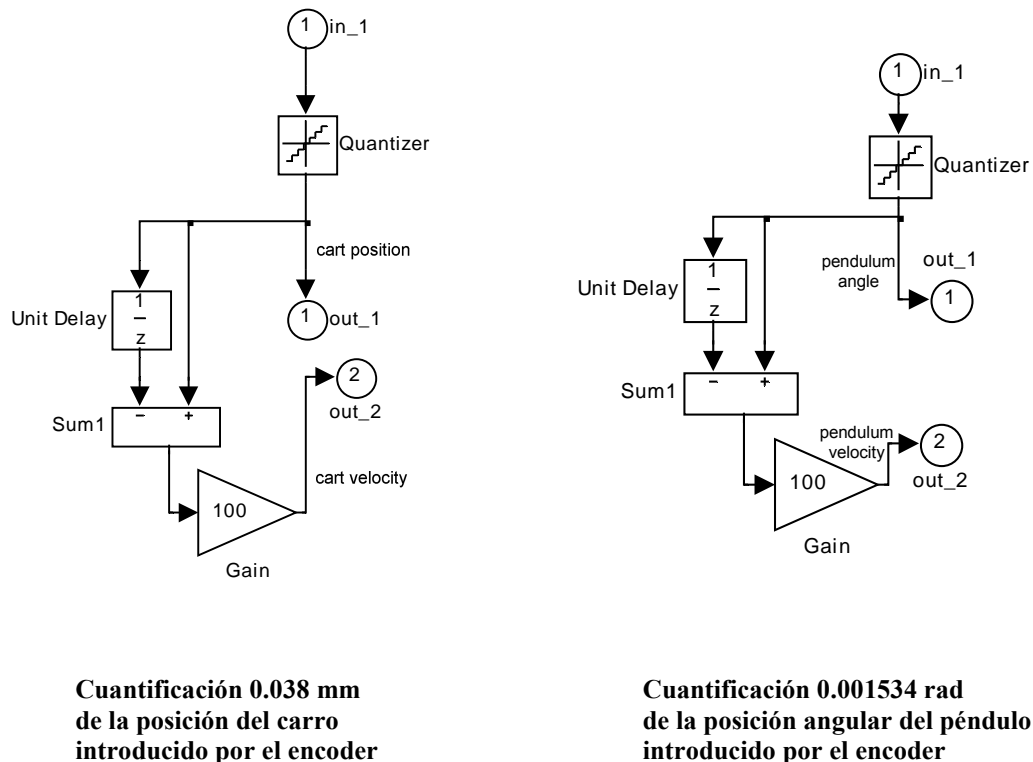


Figura. 2. 18. Bloques de cuantificación

En el sistema carro – péndulo real existen dos variables de estado: la posición angular del péndulo y la posición del carro las mismas que son medidas. Existen dos encoders montados en el eje del péndulo y el eje del motor respectivamente. Ambos encoders tienen idéntica resolución equivalente a 4096 cuentas por revolución.

La posición angular del péndulo es medido con una resolución igual a 0.001534 rad (0.087°) y la posición del carro con una resolución de 0.038 [mm] es decir una revolución del encoder del motor DC corresponde a 0.235 [m] de desplazamiento del carro sobre el riel. Cuando el carro viaja desde un extremo del riel al otro, el encoder realiza ocho revoluciones.

En el modelo dinámico las señales correspondientes a la posición del carro y la posición angular del péndulo pasan por el bloque “*hold in sample*” donde se discretizan con un tiempo de muestreo igual a 0.01 [s]. Esta pareja última de señales discretas pasan en primera instancia a dos bloques de *cuantificación*, para obtener la posición del carro y la posición angular del péndulo, en segunda instancia se les aplica un retardo para que esta nueva señal pase a restar a la señal cuantificada y este resultado multiplicarlo por una ganancia de cien (frecuencia de muestreo) obteniéndose la señal correspondiente a la velocidad del carro y a la velocidad angular del péndulo en cada módulo respectivamente, es decir estas señales de velocidad resultan del cambio de valor en sus señales de posición.

Cuando las señales ingresan al módulo de *cuantificación* pasan primero al módulo *cuantificador* para ser cuantizada en el intervalo de 0.038 [mm] y 0.001534 [rad] para la posición del carro y la posición angular del péndulo respectivamente.

La **cuantificación** es definida por la construcción de dispositivos de detección en este caso los encoders. El período de muestreo del computador tiene también una influencia significativa en las señales medidas y señal de control. El período de muestreo está fijado en 0.01 [s].

La elección de este valor tiene algunas razones:

- Partiendo de que el muestreo consiste en obtener muestras equiespaciadas temporalmente de una señal analógica. Estos es, se almacena el valor de la señal de entrada cada T_s segundos, siendo T_s el período de muestreo equivalente a 0.01 [s] y

$f_s = \frac{1}{T_s}$ la frecuencia de muestreo equivalente a 100 [Hz] (el número de muestras por segundo que se toman).

- Es suficiente para el sistema con la frecuencia natural de 1 [Hz]. Aunque la disminución del período de muestreo no es recomendado puesto que debe haber un compromiso entre el muestreo y la cuantificación^[23].

^[23] Inteco, PENDULUM – CART SYSTEM User’s Manual. 1997. (1).

CAPÍTULO III

IDENTIFICACIÓN DE PARÁMETROS DEL SISTEMA, CONTROLADOR BASADO EN REGLAS, CONTROLADOR PID

3.1 IDENTIFICACIÓN DE PARÁMETROS FÍSICOS DEL SISTEMA

Con la identificación como método experimental y mediante Matlab, se obtienen los parámetros del modelo matemático que reproducen con suficiente exactitud las características dinámicas del sistema carro – péndulo, los cuales son decisivos a la hora de definir el comportamiento dinámico del sistema de un modo simple. El principal valor a identificar es el período y frecuencia de amortiguamiento del péndulo, los demás valores que se mostrarán proceden de cálculos realizados.

La identificación consiste en variar la entrada del sistema y registrar las variaciones producidas en las salidas asociadas.

La pantalla de identificación consta en el menú principal de la caja de herramientas “p1” de Matlab como se muestran en la Figura. 3. 1.

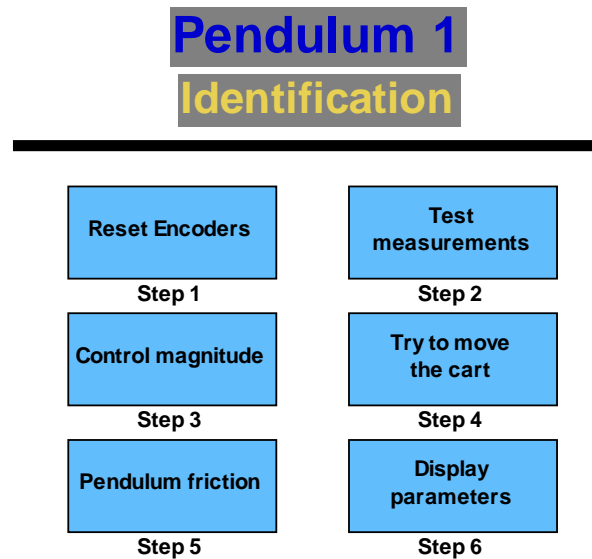


Figura. 3. 1. Pasos en la identificación del sistema carro-péndulo

A través de esta opción se realiza el proceso de identificación para verificar y cambiar valores que vienen por defecto para el sistema carro – péndulo si es necesario. Por lo que la identificación puede mejorar la exactitud de las características dinámicas del sistema carro-péndulo en el modelo matemático.

Los parámetros del sistema son guardados en el archivo “*pl_parameters.m*” que está localizado en el directorio M de la caja de herramientas “Pendulum1”. Este archivo contiene todos los cálculos necesarios para el modelo de simulación y se lo puede modificar durante la identificación del proceso.

3.1.1 Restablecimiento de encoders

Para poder establecer una nueva posición de inicio para el conteo de los encoders es necesario restablecerlos. A través de la opción “*reset encoders*” se puede realizar este procedimiento.

3.1.2 Comprobación de las señales de medición

Se experimenta con señales del sistema carro – péndulo correspondientes al encoder de posición del carro y al encoder de posición angular del péndulo.

Mediante la interfaz que se muestra en Matlab se puede monitorear la dirección y la posición del carro así como del péndulo.

Para iniciar el chequeo de señales se ubica al carro en el centro del riel y al péndulo se lo deja inmóvil, se restablece los encoders, para luego iniciar en la segunda opción “*test measurements*” como se muestra en la Figura. 3. 2.

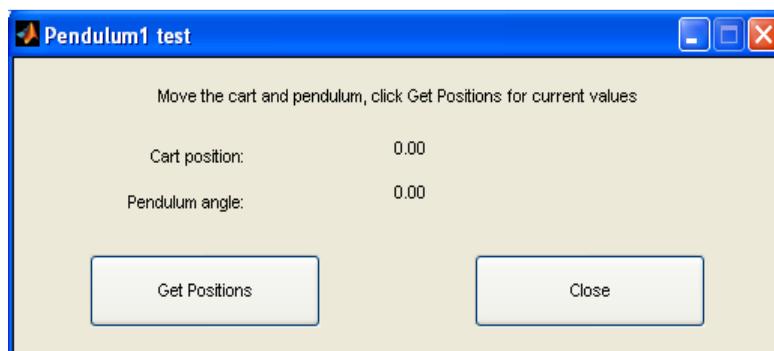


Figura. 3. 2. Ventana de comprobación de señales de los encoders

Como ejemplo de comprobación se realiza algunas operaciones:

- Mover el carro a la izquierda aproximadamente 10 [cm]. Se obtiene la posición y la dirección del carro igual a -0.09893.
- Mover el carro a la derecha aproximadamente 10 [cm]. Se obtiene la posición y la dirección del carro igual a +0.09725.
- Balancear al péndulo con la mano hacia la derecha. Se obtiene valores positivos.
- Balancear al péndulo con la mano hacia la izquierda. Se obtiene valores negativos.

Para determinar la funcionalidad de los encoders se muestran los rangos de trabajo:

- **Rango del encoder para la posición del carro:** desde el extremo izquierdo hasta el extremo derecho del riel con una longitud de 180 cm [0, +1.7823]. Se entiende el conteo del encoder que va de 0 [m] a 1.7823 [m].
- **Rango del encoder para la posición angular del péndulo:** en sentido horario desde la posición de reposo hacia abajo hasta completar una revolución [0, +6.2879]. Se entiende el conteo del encoder que va de 0 [rad] a 6.2879 [rad].

3.1.3 Identificación de la magnitud de control

En este paso se procede a encontrar los valores correspondientes entre la señal de control (PWM) que se envía desde el computador específicamente desde la interfaz de Matlab y el motor DC (Torque). La magnitud de control es imprescindible saberla ya que con ella se puede conocer el valor a partir del cual el carro empieza a moverse o el valor con el cual el carro puede adquirir la velocidad máxima.

Para iniciar la identificación se debe ingresar a la opción “*control magnitude*”, donde se muestra el rango de valores de ingreso para el PWM que es $[-1,1]$ como se muestra en la Figura. 3. 3.

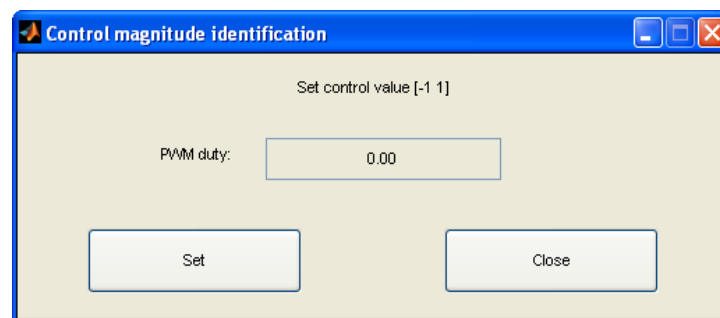


Figura. 3. 3. Ajuste del ciclo de duración PWM

Así en este proceso se puede identificar que al ingresar un valor PWM negativo el carro se mueve hacia la derecha del riel y si el valor es positivo el carro se mueve hacia la izquierda del riel.

Con el propósito de obtener estos valores se ubica una balanza electrónica con un extremo fijo en el extremo izquierdo del riel y el otro extremo fijo en el carro como se

muestra en la Figura. 3. 4. Los valores de fuerza estática que ejerce el motor DC en relación a la señal PWM se muestra en la Tabla. 3. 1.



Figura. 3. 4. Montaje de la balanza

PWM	Fuerza [kgf]	Fuerza [N]
0	0	0
0.1	0.07	0.98
0.2	0.195	1.96
0.3	0.335	2.94
0.4	0.55	3.92
0.5	0.72	4.9
0.6	0.69	5.88
0.7	0.81	6.86
0.8	0.74	7.84
0.9	0.82	8.82
1	0.9	9.8

Tabla. 3. 1. Fuerza del motor DC

Se grafican los valores de la fuerza estática con sus respectivos valores PWM, como se muestra la Figura. 3. 5.

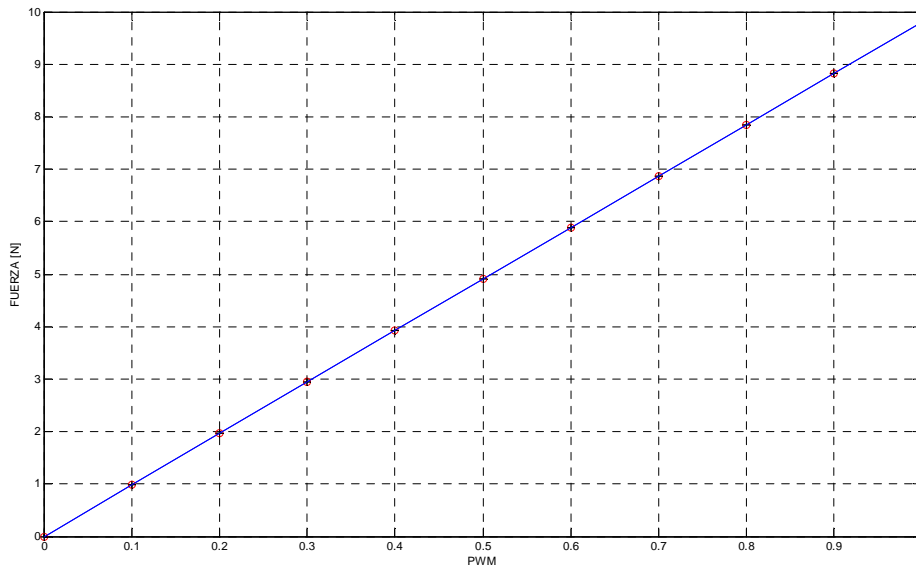


Figura. 3. 5. Distribución de valores PWM vs. Fuerza

Estos valores muestran claramente una tendencia lineal por lo que se usa la herramienta *polyfit* de Matlab para obtener la ecuación (3.1) que describe este comportamiento. Además esta última ecuación permite graficar y obtener cualquier valor de fuerza según el PWM ingresado de una manera exacta.

$$y = 9.8x \quad (3.1)$$

Ahora se procede a graficar todo el rango de valores PWM disponibles partiendo de la ecuación 3.1 tal como se muestra en la Figura. 3. 6.

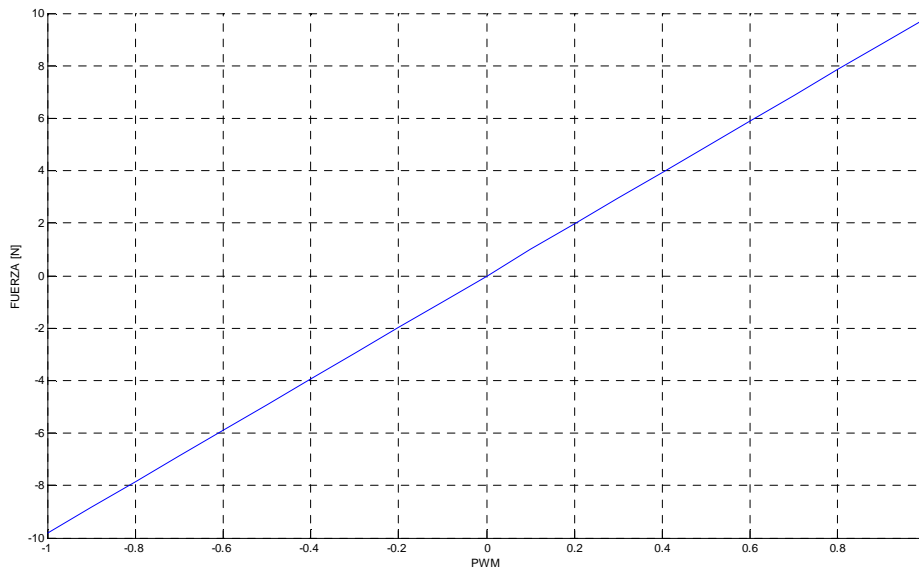


Figura. 3. 6. Valores PWM vs. Fuerza

Entendiéndose que la máxima fuerza que ejerce el motor DC en el sistema es de 10 [N] y como mínima 0[N] al ingresar como valor PWM 1 y 0 respectivamente.

3.1.4 Mínima fuerza necesaria para mover el carro

Cuando el carro trata de moverse sobre el riel existe una fuerza de rozamiento estática f_s que se opone a este movimiento. Partiendo de aquello es necesario conocer el valor mínimo de la fuerza F necesaria para vencer f_s y así mover el carro.

Antes de iniciar con la identificación se debe ubicar el carro al extremo izquierdo y presionar el botón de encendido para la alimentación del motor DC. En el menú principal presionar la opción “*Try to move the cart*”, luego dar una respuesta positiva en la ventanas

de dialogo activas de la Figura. 3. 7. Rápidamente aparece una ventana de información como se muestra en la Figura. 3. 8.



Figura. 3. 7. Ventanas de diálogo

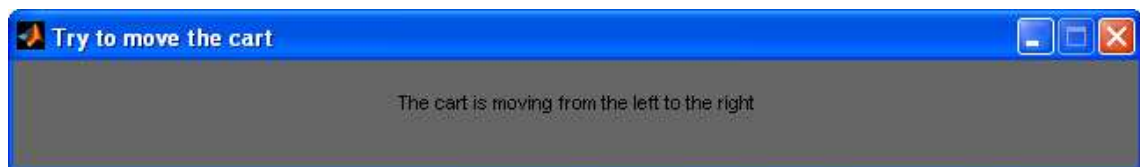


Figura. 3. 8. Información sobre el movimiento del carro

Inmediatamente el algoritmo de identificación empieza a trabajar, el carro se mueve automáticamente desde el extremo izquierdo hacia la derecha del riel. El control de la fuerza es lentamente incrementada para detectar el movimiento del carro. Finalmente el carro alcanza el lado derecho del riel, un poco más del centro. Cuando se observe la Figura. 3. 9. se ubica el carro con la mano al extremo derecho del riel, una vez ahí se da clic en OK.



Figura. 3. 9. Ubicación del carro

Enseguida el experimento inicia otra vez, pero ahora el carro se mueve en dirección opuesta como muestra la ventana de información de la Figura. 3. 10.



Figura. 3. 10. Información sobre el movimiento del carro

Cuando este procedimiento finalice la gráfica de la Figura. 3. 11. aparece. Es entonces que el carro termina su movimiento al extremo izquierdo del riel.

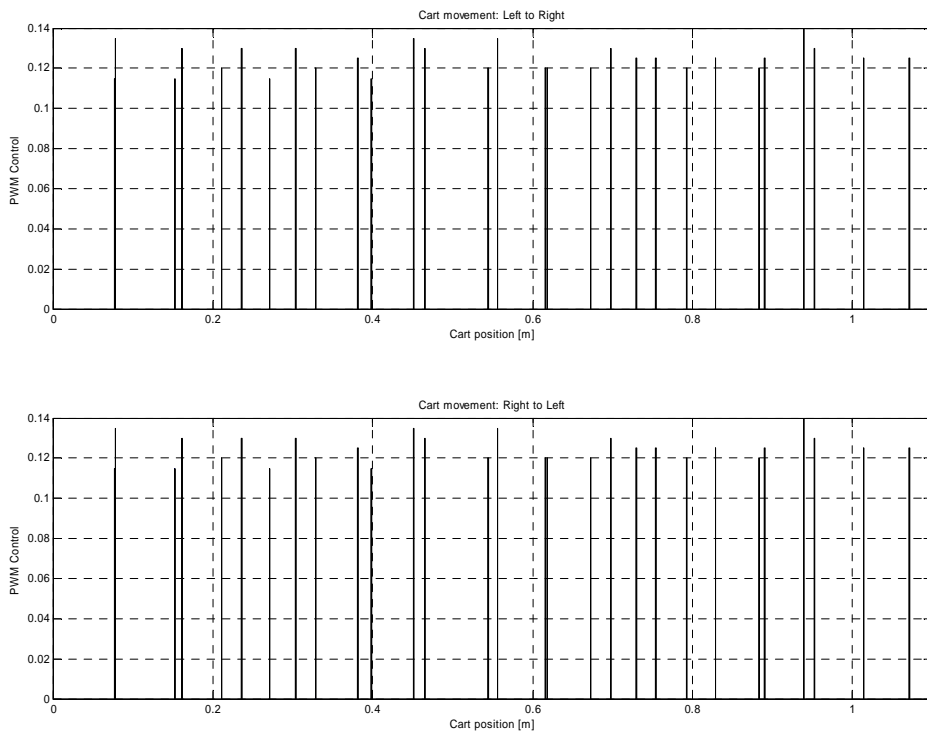


Figura. 3. 11. Detección de la mínima fuerza

Las barras de la Figura. 3. 11. muestran los valores de fuerza requerida para mover el carro a diferentes posiciones en el riel. En tal razón el valor medio de la señal de control (duración del ciclo PWM) es igual a 0.127. Es decir la mínima fuerza con la que el motor rompe la inercia y empieza a moverse es de 1.244 [N] que corresponde a un valor PWM de 0.127.

3.1.5 Identificación de la fricción del péndulo

Antes de iniciar con esta identificación se debe fijar el carro al extremo izquierdo del riel usando un resorte que permita mantenerlo inmóvil como se muestra en la Figura. 3. 12., puesto que se va a trabajar con el péndulo. Además se debe verificar el encendido de la fuente de alimentación y circuito de potencia.



Figura. 3. 12. Fijación del carro

Una vez que el carro se encuentre fijo se inicia la identificación a través del botón “*Pendulum friction*” tras lo cual aparece la ventana que se muestra en la Figura. 3. 13.

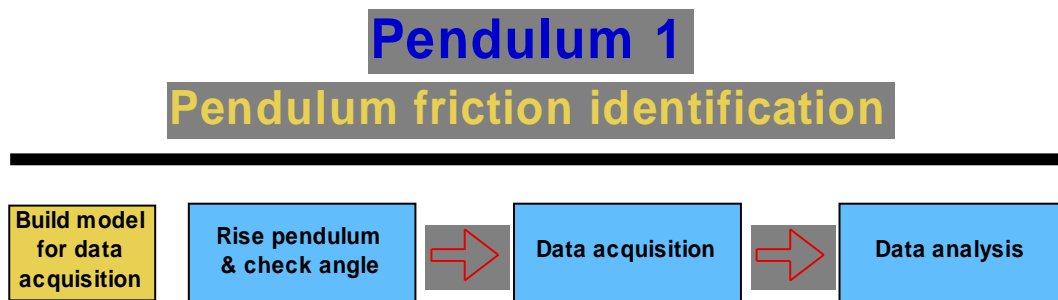


Figura. 3. 13. Ventana de identificación de fricción del péndulo

Aquí se procede a tomar datos del péndulo usando el controlador del dispositivo obteniéndose al final la frecuencia de amortiguamiento y período del péndulo. Al dar clic en la botón “*Build model for data acquisition*” se construye el modelo RTW de experimentación del modelo que se muestra en la Figura 3. 14., necesario para obtener la fricción del péndulo, mismo que está nombrado como “*PI_ExpIdentPF*”. Se debe verificar que el directorio de trabajo actual no sea uno de los directorios que se encuentra en el MATLAB Path ya que de ser el caso no se dará la construcción del modelo de adquisición apareciendo un mensaje error que indica que el RTW no permite construir programas en las direcciones donde esté instalado Matlab, por lo tanto se lo debe cambiar al directorio de trabajo o carpeta *work*.

Pendulum 1 Data acquisition for Pendulum friction

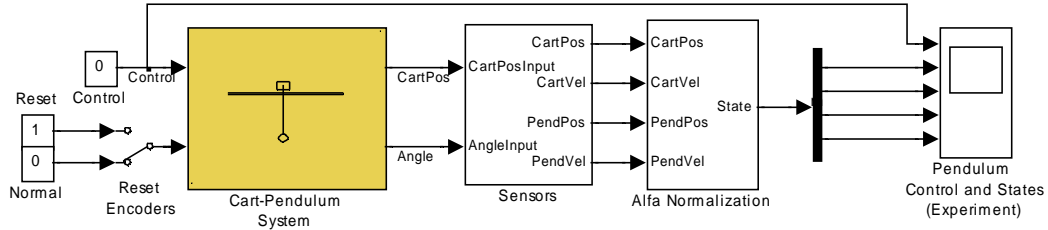


Figura. 3. 14. Ventana de adquisición de datos para la fricción del péndulo

En la Figura. 3. 14. se muestra el bloque “Alfa Normalization” que toma la señal de posición angular del péndulo realizando con ella el cálculo del ángulo suplementario para el caso en el que su valor se encuentre entre el tercer y cuarto cuadrante con valor negativo y cuando se encuentre entre el primero y segundo cuadrante obteniéndose el mismo ángulo con valor positivo todo esto en radianes usando la función *atan2* de Matlab como se muestra en la Figura. 3. 15. Todo este proceso se realiza con el objetivo de obtener la señal referida al origen de coordenadas y comprendida entre los valores límites de π y $-\pi$. A esta señal se suma el valor π como condición inicial de posición angular del péndulo.

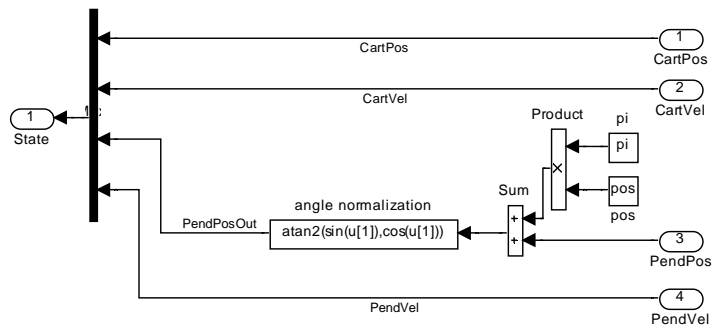


Figura. 3. 15. Normalización posición angular del péndulo, Modelo simulación

El encoder que sensa la posición del carro toma como punto de referencia el extremo izquierdo del riel, lo que debería mostrarse como 0 [m] siendo esto compensado con un *offset* añadido al interior del controlador del dispositivo de -0.93 [m] es decir empieza detectando este valor, el encoder que sensa la posición angular del péndulo toma como punto de referencia la posición vertical hacia abajo del péndulo empezando como punto de referencia el valor π incrementando su valor si este gira en sentido horario y decrementando su valor si este gira en sentido antihorario.

Al dar clic en el botón “*Rise pendulum & check angle*” se observa el valor actual de la posición angular del péndulo en la posición vertical hacia abajo como se muestra en la Figura. 3. 16. Es decir para este proceso el valor desde el cual se empieza a medir es 3.1416, si se lo mueve en sentido horario su valor se incrementa pero si se lo mueve en sentido antihorario su valor se decrementa.

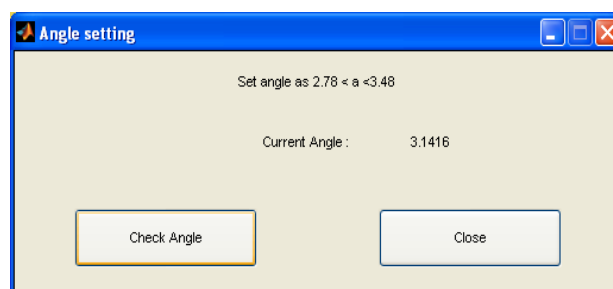


Figura. 3. 16. Ventana de comprobación de posición angular del péndulo

Para dar inicio con el proceso es necesario subir el péndulo un ángulo $\pi \pm \pi/9$ (20 grados arriba de la posición estable hacia abajo) y soltarlo. Rápidamente se da clic en el botón “*Data acquisition*” iniciándose la obtención de datos con un tiempo de simulación

externa de 40 [s]. Estos datos son recibidos mientras el péndulo está oscilando, los cuales son usados para calcular el coeficiente de fricción.

Al dar clic en “*Data analysis*” se procesan los datos y se obtiene automáticamente la gráfica que se muestra en la Figura. 3. 17.

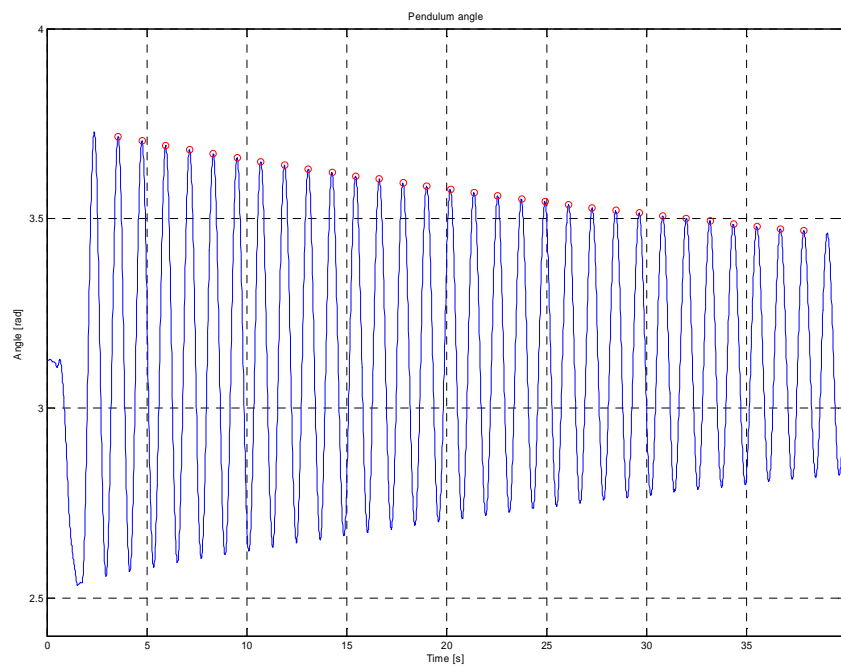


Figura. 3. 17. Trazado de datos del péndulo

En el procesamiento de datos se realizan los siguientes cálculos:

- Cálculo del valor principal del período del péndulo T
- Cálculo de la frecuencia de amortiguamiento f_{pc}

La fórmula de cálculo para la constante de amortiguamiento es la siguiente:

$$f_{pc} = \frac{2}{(2n-1)T} \log \left(\frac{A_1}{A_n} \right) \quad (3.2)$$

$$f_p = J_p \cdot f_{pc} \quad (3.3)$$

Donde:

f_{pc} , frecuencia de amortiguamiento [1/s]

T , período oscilaciones del péndulo [s]

f_p , coeficiente de fricción del péndulo [N.m.s/rad]

J_p , momento de inercia del péndulo [kg.m²]

n , número de máximos sobre el valor π

A_1 , A_n , valores máximos, después de 1 oscilación y 'n' oscilaciones respectivamente

Los valores máximos locales A_1 y A_n son tomados de la Figura. 3.17., de acuerdo a la distribución que muestra la gráfica de la Figura. 3. 18.

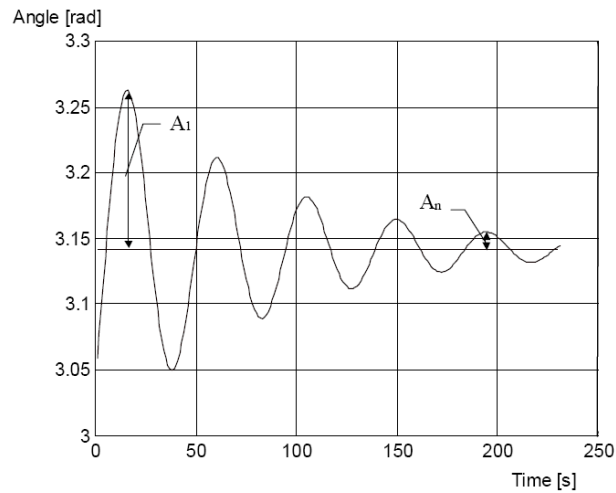


Figura. 3. 18. Oscilaciones amortiguadas del péndulo

Finalmente en la ventana de comandos de Matlab se muestra automáticamente los siguientes resultados:

Período del péndulo: 1.182963 [s]

Frecuencia de amortiguamiento: 0.016574 [Hz]

3.1.6 Presentación de parámetros

Al dar clic en el botón “*Display parameters*” se calcula y se despliega todos los parámetros requeridos en el modelo matemático de simulación, al igual que los parámetros identificados en forma experimental.

Cabe recalcar que los valores presentes por defecto en el modelo de simulación no coinciden con los identificados en tal razón se los cambió en el archivo “*p1_parameters.m*” y se obtiene la Tabla. 3. 2.

Parámetro	Descripción	Unidad
m_c	Masa del carro	1.0423 [kg]
m_p	Masa del péndulo	0.104 [kg]
f_p	Coefficiente de fricción del péndulo	0.00004949 [N.m.s/rad]
FS	Fricción Estática	1.63322 [N]
FC	Fricción de Coulomb	0.5 [N.s/m]
l	Distancia	0.00826190 [m]
J	Momento de inercia	0.00290756 [kg.m ²]
g	Aceleración de la gravedad	9.81 [m/s ²]
M	Magnitud de control	12.86 [N]
DZ_u	Zona muerta de Control	0.127
DZ_{cv}	Zona muerta del Carro	0.1 [m/s]
DZ_{cp}	Zona muerta del Péndulo	1.5 [rad/s]

Tabla. 3. 2. Valores Identificados para el modelo matemático de simulación

3.2 CONTROLADOR BASADO EN REGLAS

El controlador basado en reglas no tiene correspondencia con un tipo de controlador basado en criterios de calidad como tiempos de respuesta, error, etc, sino que más bien está enfocado a cumplir con el objetivo de control de manera robusta, insensible a cambios en parámetros y perturbaciones.

El objetivo de control está definido para balancear el péndulo hacia arriba y estabilizarlo en posición vertical implicando un ajuste entre el hecho de mover el carro y a la vez balancear al péndulo por lo que se lo ha clasificado como un método de control inteligente.

Dentro del modelo de simulación y experimentación realizados por el fabricante se encuentra el *controlador basado en reglas* y el *controlador PID* como se muestra en la Figura. 3. 19.

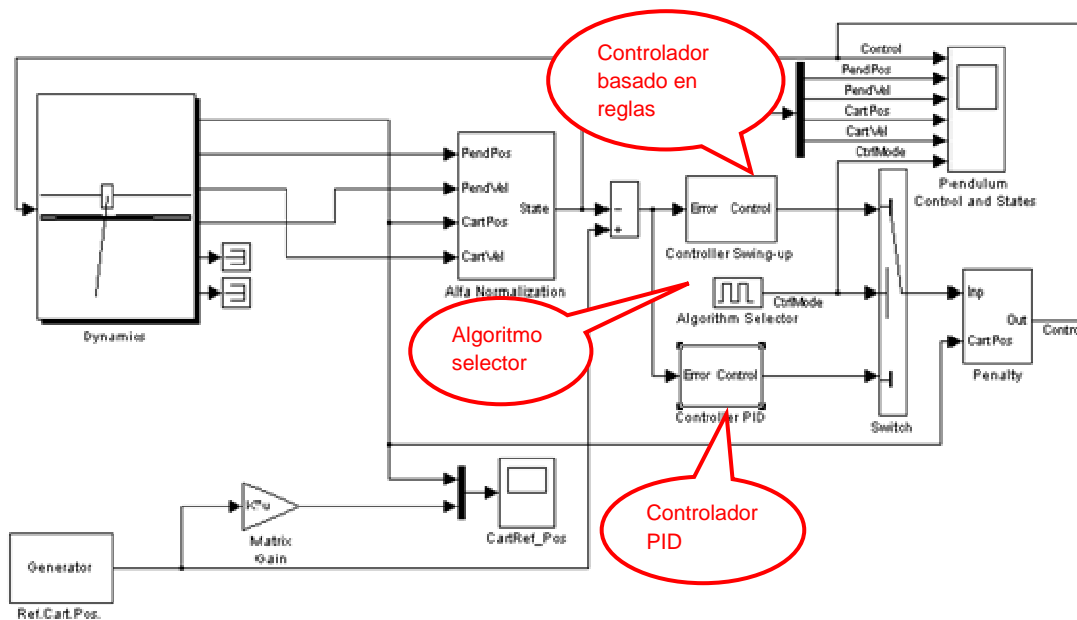


Figura. 3. 19. Modelo Simulación - Controlador basado en reglas y Controlador PID

Estos dos controladores pasan a funcionar acorde con la elección que realice el bloque *algoritmo selector* para el funcionamiento en modo péndulo invertido (bloque controlador basado en reglas) cuando sea *uno* o el funcionamiento en modo grúa (bloque controlador PID) cuando sea *cero*.

El algoritmo del control basado en reglas se muestra en la Tabla. 3. 3. numerado en código del 1 al 12.

<i>REGLAS</i>	<i>CÓDIGO</i>
<i>If</i> $ x_2 - ZonaEstabilización < 0$	<i>No. 1</i>
<i>then</i> $u =: K_1x_1 + K_2x_2 + K_3x_3 + K_4x_4$	<i>No. 2</i>
<i>else if</i> $\{0.5 * (x_4)^2 + 9.81 * k * R[\cos(x_2) - 1]\} > 0$	<i>No. 3</i>
<i>then</i> $u =: 0$	<i>No. 4</i>
<i>then</i> $u = sign [x_4 * (x_2 - \pi/2)]$	<i>No. 5</i>
<i>if</i> $u > 0$	<i>No. 6</i>
<i>then</i> $u =: U * u + Fricción$	<i>No. 7</i>
<i>then</i> $u =: U * u - Fricción$	<i>No. 8</i>
<i>if</i> $ u > 1$	<i>No. 9</i>
<i>then</i> $u =: sign (u)$	<i>No. 10</i>
<i>if</i> $ x_1 - \frac{R_l}{2} < 0$	<i>No. 11</i>
<i>then</i> $u =: sign (x_1)$	<i>No. 12</i>

Tabla. 3. 3. Algoritmo del controlador basado en reglas

Este código simboliza las operaciones internas que realiza todo el controlador. Si el péndulo está en la vecindad de la posición vertical estable dentro de la zona de estabilización (No. 1) entonces un *linear controller* es aplicado (No. 2) siendo esta una matriz con realimentación de ganancias.

Si el péndulo se encuentra fuera de la zona de estabilización (no se cumple No.1) este debe balancearse hacia arriba nuevamente. El algoritmo comprueba si el balanceo del péndulo no resulta en un sobrepaso de la posición vertical estable.

Para lograr una llegada suave a la vecindad de la posición vertical estable, el algoritmo comprueba si la energía cinética del péndulo menos la energía perdida debido a las fuerzas de fricción es suficiente para levantar el centro de gravedad del péndulo hasta la posición vertical condición No. 3 (bloque *soft landing arbiter*). Si la condición es satisfecha entonces el control es puesto a cero (No. 4).

La regla para el balanceo del péndulo tiene una forma muy simple (No. 5). Se trata del “*bang-bang*” relacionado con momentos de cambio del signo de la velocidad angular y la posición angular (“*swinging controller*”).

Los efectos relacionados con la fricción de Coulomb del carro se incorporan directamente al control. Si el control es positivo (No. 6) se lo hace *más positivo* mediante la adición de una cantidad de control para superar la fricción de Coulomb (No. 7). Del mismo modo cuando se realiza un control es negativo se lo hace *más negativo* (No. 8). El

bloque que realiza esta acción es el *Coulomb Friction & Control Gain*. Entonces el control es normalizado a permanecer en el rango $[-1,+1]$ (No. 9 y No. 10).

Finalmente el bloque *Limiter* es añadido para reforzar el centrado del carro y prevenir que este llegue a los extremos del carril. La regla es simple, si el rango dado por el parámetro de este bloque es superado (No. 11) entonces el opuesto a la dirección de la fuerza de movimiento máxima se aplica al carro (No. 12).

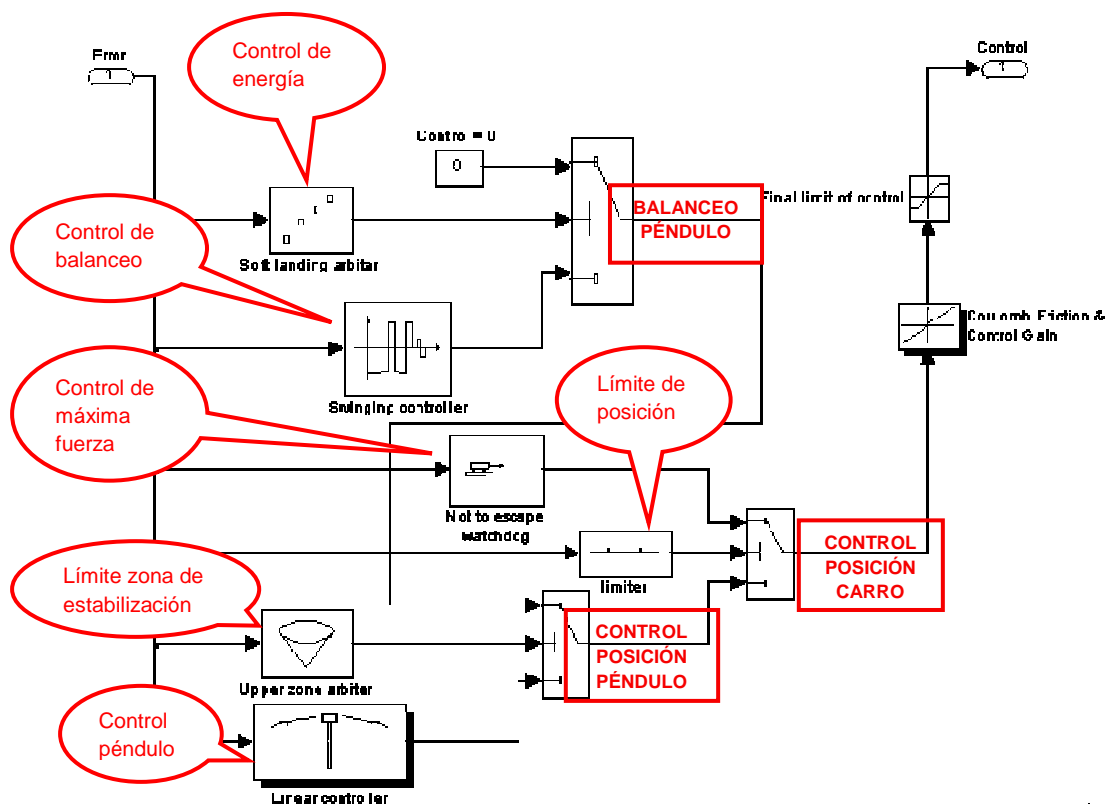


Figura. 3. 20. Modelo Simulación - Controlador basado en reglas

Una vez revisado el algoritmo de control se muestra el interior del controlador en la Figura. 3. 20., donde se aprecian tres etapas las cuales trabajan conjuntamente:

- Etapa de balanceo del péndulo.
- Etapa de control de posición del péndulo.
- Etapa de control de posición del carro.

3.2.1 Etapa de balanceo del péndulo

Esta etapa es la encargada de balancear el péndulo mediante el bloque *control de balanceo*, en la zona de balanceo hasta los límites de la zona de estabilización y ubicar el carro al centro del riel como se muestra en la Figura. 3. 21.

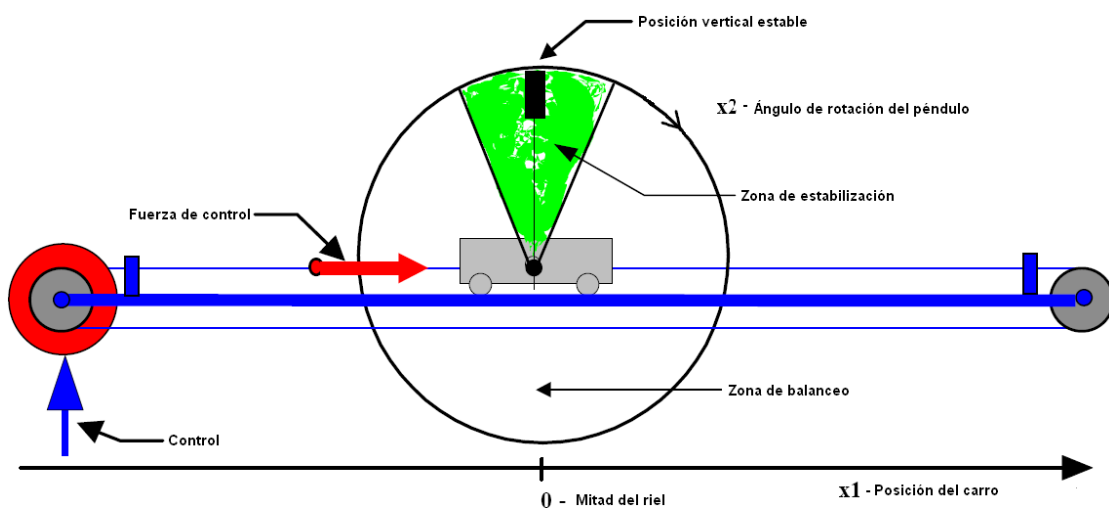


Figura. 3. 21. Zonas de actividad para los algoritmos de control

Para lograr el balanceo del péndulo se fundamenta en la ecuación 3.1, a partir de la cual se consigue una salida de control normalizada entre ± 1 .

$$u = \text{sign} \left[x_4 \left(|x_2| - \frac{\pi}{2} \right) \right] \quad (3.4)$$

Donde:

u , señal de control

x_4 , velocidad angular del péndulo

x_2 , posición angular del péndulo

En el interior del bloque *control de energía* se calcula el valor de la energía total con el que se controla la conmutación entre el bloque *control de balanceo* y la constante cero.

La energía total del péndulo es calculada en cada paso numérico que realiza el encoder de posición angular del mismo.

3.2.2 Etapa de control de posición del péndulo

Esta etapa es la encargada de estabilizar al péndulo en la posición vertical hacia arriba y también estabilizar el carro en la mitad del riel.

El bloque *límite zona de estabilización* fija el valor 0.3 [rad] desde el cual empieza la zona de estabilización, controlando la conmutación entre el bloque *control péndulo* y la constante cero.

3.2.3 Etapa de control de posición del carro

Esta etapa es la encargada de limitar el recorrido que puede realizar el carro sobre el riel.

El bloque *límite de posición* fija el límite máximo de recorrido con un valor de 0.35 [m], controlando la conmutación entre el control de posición del péndulo y el bloque *control de máxima fuerza* donde el valor máximo de control es generado con signo opuesto al de la posición del carro.

3.3 CONTROLADOR PID

Este controlador es el más usado en la industria por ser el más robusto en muchas aplicaciones, su diagrama se muestra en la Figura. 3. 22.

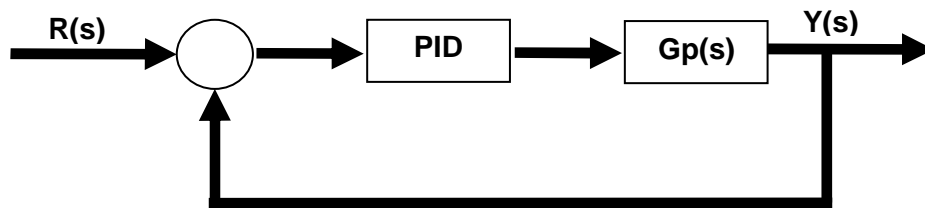


Figura. 3. 22. Diagrama bloque PID

Es un controlador digital ya que está implementado con el FPGA que dispone la tarjeta de interfaz.

3.3.1 Diseño del controlador PID

El controlador PID tiene tres acciones que son: proporcional (P), integral (I) y derivativa (D). La función de transferencia discreta parte de la expresión general de un controlador PID, aplicado una aproximación rectangular para llegar a la siguiente forma:

$$u = K_p \varepsilon_1(k) + K_i \sum_{k=0}^n \varepsilon_1(k) + K_d [\varepsilon_1(k) - \varepsilon_1(k - 1)] \quad (3.5)$$

Donde:

ε_1 , señal de error (resta entra la posición deseada del carro y la posición medida del carro)

k_p , ganancia proporcional

k_i , ganancia de integración

k_d , ganancia de derivación

Las constantes k_p , k_i , k_d pueden ser escogidas de acuerdo la regla de Ziegler – Nichols para oscilaciones sostenidas.

Como se puede apreciar en la Figura. 3. 19. el controlador está presente en el modelo demostrativo de simulación y también en el modelo demostrativo de experimentación para el funcionamiento del sistema en modo grúa.

Tiene como variables de control la posición angular del péndulo y la posición del carro.

CAPÍTULO IV

DISEÑO E IMPLEMENTACIÓN DEL CONTROLADOR DIFUSO

El diagrama de bloques del sistema de control de posición para la planta carro-péndulo se muestra en la Figura. 4. 1, en un control en lazo cerrado. Aquí el controlador automático mantiene la posición del carro y el péndulo comparando sus posiciones reales con las posiciones deseadas y corrigiendo cualquier error mediante un ajuste en la rotación del motor DC.

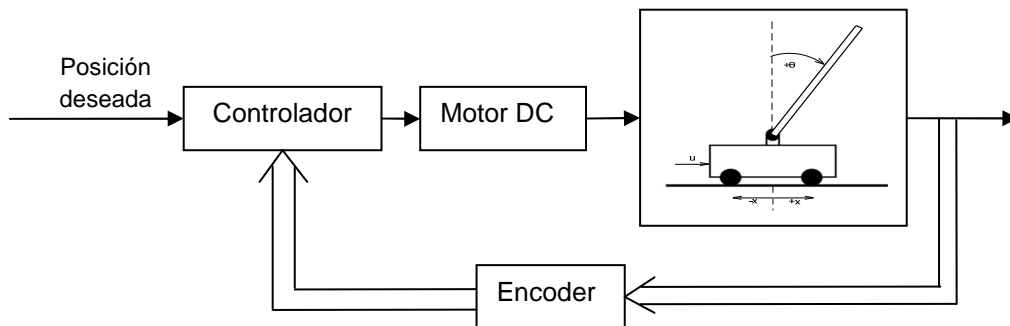


Figura. 4. 1. Diagrama de bloques del sistema de control

Son dos las variables que se controla, usando únicamente una señal de control como se muestran en la Tabla 4.1.

TIPO	CONTROL PÉNDULO	CONTROL CARRO
Variable controlada	Posición péndulo [rad]	Posición carro [m]
Variable manipulada	Ciclo de trabajo del motor DC	Ciclo de trabajo del motor DC

Tabla. 4. 1. Variables sistema de control

El controlador difuso para la planta carro – péndulo consta de las siguientes partes: fusificación, mecanismo de inferencia, base de reglas difusas, defusificación. Se ha diseñado usando la herramienta *Fuzzy Logic Toolbox* de Matlab para un sistema de inferencia tipo Mamdani, con un operador difuso AND para las dos proposiciones inmersas en el “antecedente” sobre cada regla difusa, con un método de Implicación MIN aplicado entre el valor obtenido por el operador AND y el peso que representa el “consecuente” sobre cada regla difusa, con un método de Agregación MAX aplicado a la combinación de las funciones asociadas a las salidas de cada regla difusa en una única función de salida, y con un método de defusificación CENTROIDE a la función de salida.

El modelo matemático del sistema carro – péndulo se usa para el diseño y la simulación del controlador difuso.

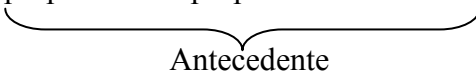
4.1 CONTROLADOR DIFUSO PROPORCIONAL DERIVATIVO CON BALANCEO

En esta sección se tiene como objetivo complementar el accionar del controlador *Swing-up* mediante el diseño de un controlador difuso PD. En una primera etapa trabaja el controlador *swing-up* logrando balancear el péndulo y ubicándolo en la zona de estabilización es entonces que en una segunda etapa trabaja el controlador difuso PD dentro de los límites de esta zona para ubicar el péndulo en la posición vertical estable y el carro en el centro del riel.

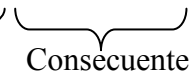
Se realiza un controlador difuso PD para la posición angular del péndulo y un controlador difuso PD para la posición del carro. Cada uno de los controladores tiene dos entradas y una salida. Estas señales de salida se suman conformando una sola señal de control.

Las reglas difusas tienen la siguiente forma:

- *SI* proposición *Y* proposición *entonces* consecuencia



Antecedente



Consecuente

4.1.1 Definición de variables y valores lingüísticos

Las variables asignadas para el control de posición angular del péndulo, se indican en la Tabla. 4. 2.

CONTROLADOR DIFUSO PD PÉNDULO		
Tipo	Variable	Etiqueta
Entrada	Error de posición angular del péndulo	Errorpen
Entrada	Cambio del error de posición angular del péndulo (velocidad angular)	Cerrorpen
Salida	Control	Controlpen

Tabla. 4. 2. Variables del controlador difuso PD péndulo

El mismo esquema de definición se usa para el control de posición del carro, como se indica en la Tabla. 4. 3.

CONTROLADOR DIFUSO PD CARRO		
Tipo	Variable	Etiqueta
Entrada	Error de posición del carro	Errorcar
Entrada	Cambio del error de posición del carro (velocidad)	Cerrorcar
Salida	Control	Controlcar

Tabla. 4. 3. Variables del controlador difuso PD carro

Los valores lingüísticos que han sido escogidos para las variables de entrada y salida de los controladores difusos PD péndulo y PD carro son cinco, como se indica en la Tabla.

4. 4.

Valor lingüístico	Etiqueta
Grande negativo	GN
Pequeño negativo	PN
Cero	Z
Pequeño positivo	PP
Grande positivo	GP

Tabla. 4. 4. Valores lingüísticos de los controladores difusos PD

4.1.2 Rango de variables

El conocimiento detallado del proceso facilita la definición de los universos de discurso, es así que para cada variable se han definido los rangos en los que trabajan los controladores para el sistema carro – péndulo.

Las variables del controlador difuso PD péndulo se muestran en la Tabla. 4. 5.

CONTROLADOR DIFUSO PD PÉNDULO	
Variable	Rango
Error de posición angular del péndulo	[-0.3, 0.3] [rad]
Cambio del error de posición angular del péndulo	[-2, 2] [rad/s]
Control	[-0.2, 0.2]

Tabla. 4. 5. Rango de variables del controlador difuso PD péndulo

Las variables del controlador difuso PD carro se muestran en la Tabla. 4. 6.

CONTROLADOR DIFUSO PD CARRO	
Variable	Rango
Error de posición del carro	[-0.35, 0.35] [m]
Cambio del error de posición del carro	[-1, 1] [m/s]
Control	[-0.2, 0.2]

Tabla. 4. 6. Rango de variables del controlador difuso PD carro

4.1.3 Funciones de pertenencia

Las funciones de pertenencia utilizadas en los conjuntos difusos del *controlador difuso PD péndulo* son de tipo triangular y exponencial. Estas funciones de pertenencia son asignadas a los conjuntos difusos de entrada y salida en este controlador.

Como se puede observar en la Figura. 4. 2.:

- *Errorpen* y *Cerrorpen*: se ha elegido funciones de pertenencia de tipo triangular para los valores lingüísticos medios, mientras que para las fronteras de tipo exponencial con lo cual se asegura una cobertura de control fuera de

los límites establecidos aplicándose los valores máximos de la variable *controlpen*. Los valores de esta señal de control cumplen con el objetivo de ubicar el péndulo en posición vertical. Esta señal de control se contrapone con la tarea que realiza el controlador *swing-up* por lo que permite disminuir en parte la velocidad con la que ingresa el péndulo hacia la zona de estabilización.

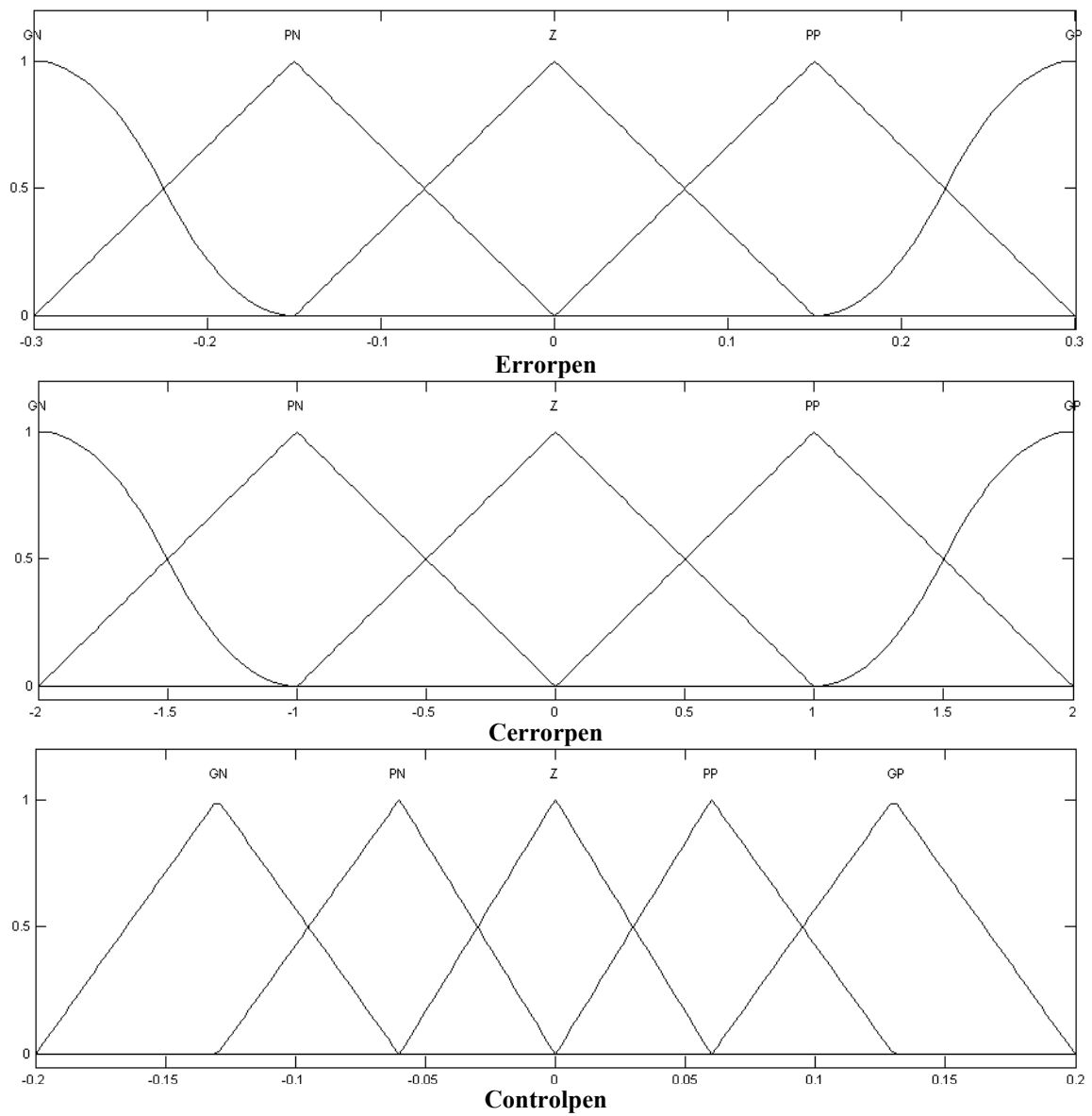


Figura. 4. 2. Funciones de pertenencia, Controlador difuso PD péndulo

- *Controlpen*: se ha elegido funciones de pertenencia de tipo triangular para todos los valores lingüísticos. Estas funciones están dispuestas de forma equidistante hasta el límite establecido por la variable consiguiendo la superficie de control que se muestra en la Figura. 4. 3.

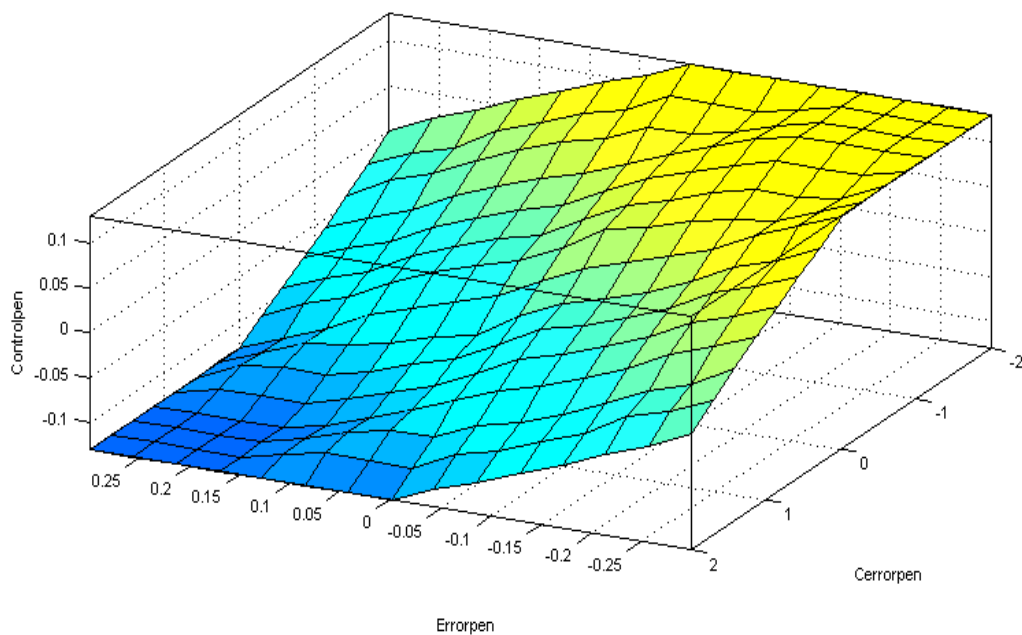


Figura. 4. 3. Superficie de control del controlador difuso PD péndulo

La Figura. 4. 3. muestra la superficie de control que representa la salida del controlador difuso PD péndulo como función de las dos señales de entrada. Se observa una linealidad en su extensión que será óptimo para el control de posición angular.

Las funciones de pertenencia utilizadas en los conjuntos difusos del *controlador difuso PD carro* son de tipo triangular y exponencial. Estas funciones de pertenencia son asignadas a los conjuntos difusos de entrada y salida en este controlador.

Como se puede observar en la Figura. 4. 4.:

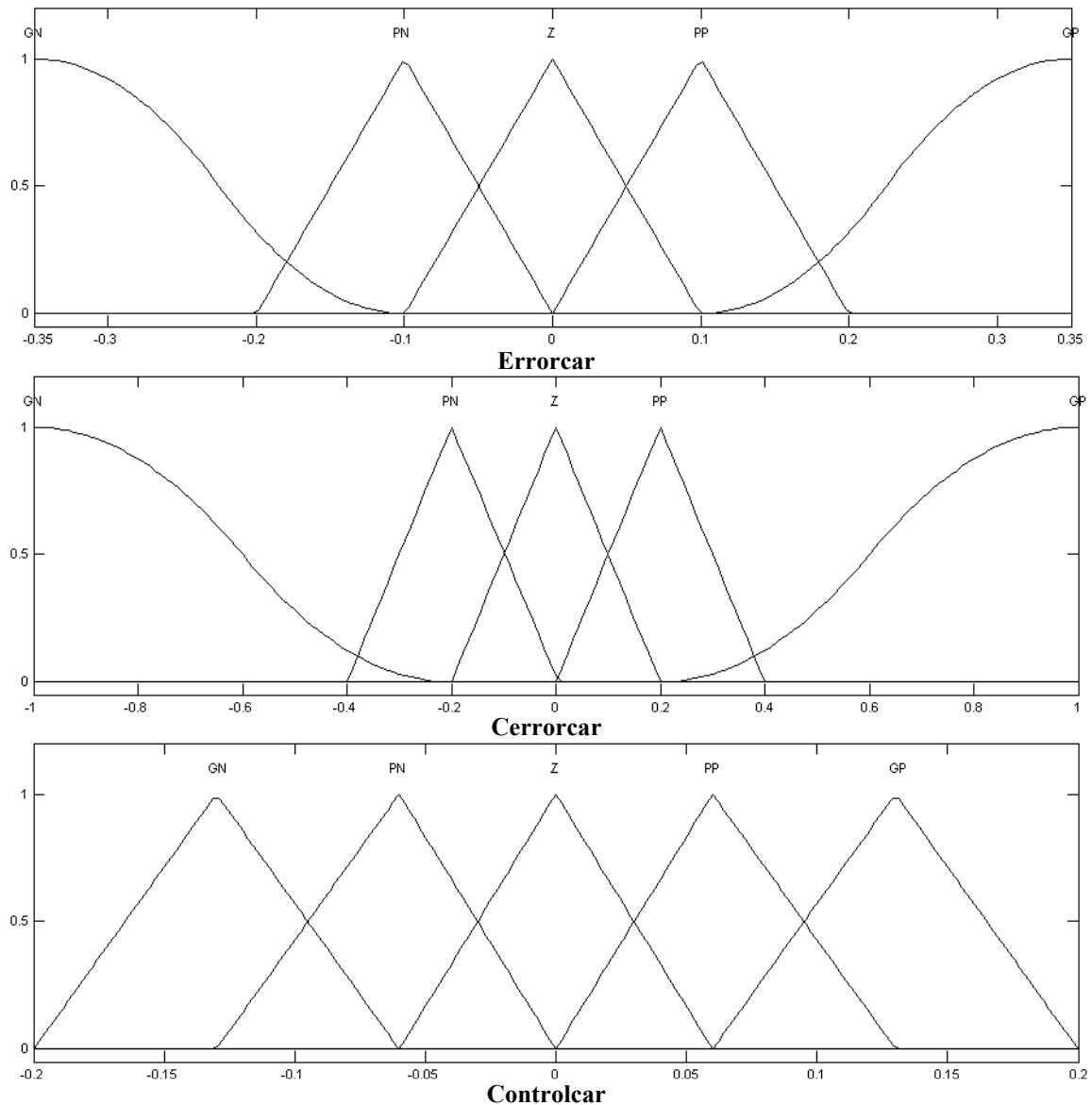


Figura. 4. 4. Funciones de pertenencia, Controlador difuso PD carro

- *Errorcar y Cerrorcar:* se ha elegido funciones de pertenencia de tipo triangular para los valores lingüísticos medios, mientras que para las fronteras de tipo exponencial asegurando una cobertura de control fuera de los límites establecidos. A las funciones de pertenencia que se encuentra en la mitad se las ha contraído a valores más cercanos a cero para de este modo enfocar el control sobre el carro hacia el centro del riel.

- *Controlcar*: se ha elegido funciones de pertenencia de tipo triangular para todos los valores lingüísticos. Estas funciones están dispuestas de forma equidistante hasta el límite establecido por la variable consiguiendo la superficie de control que se muestra en la Figura. 4. 5.

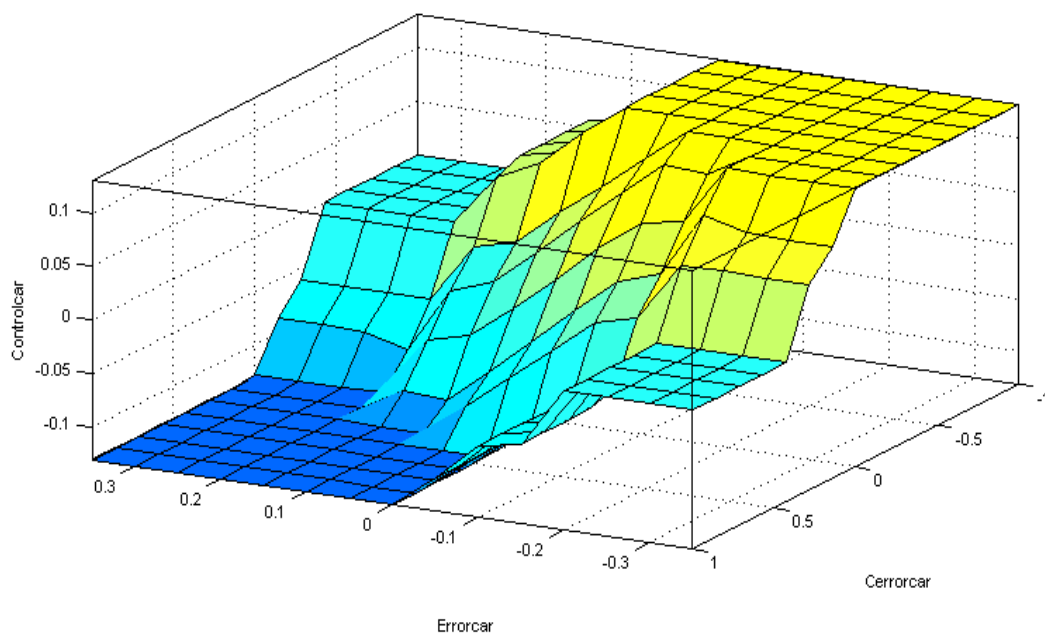


Figura. 4. 5. Superficie de control del controlador difuso PD carro

La superficie de control que muestra la Figura. 4. 5. indica una linealidad en su extensión, enfocada alrededor del centro de la misma debido a la agrupación en las funciones de pertenencia de las variables Errorcar y Cerrorcar, que no es el caso de los extremos, consiguiendo de este modo una señal de control enfocada en un rango menor al establecido por estas variables.

4.1.4 Base de reglas

Se ha dispuesto trabajar con 25 reglas que se presentan en la Tabla. 4. 7., las cuales guardan la misma estructura de composición en los dos controladores difusos PD.

1	Si	Errorpen es Grande negativo	Y	Cerrorpen es Grande negativo	Entonces	Controlpen es Grande positivo
2	Si	Errorpen es Grande negativo	Y	Cerrorpen es Pequeño negativo	Entonces	Controlpen es Grande positivo
3	Si	Errorpen es Grande negativo	Y	Cerrorpen es Cero	Entonces	Controlpen es Grande positivo
4	Si	Errorpen es Grande negativo	Y	Cerrorpen es Pequeño positivo	Entonces	Controlpen es Pequeño positivo
5	Si	Errorpen es Grande negativo	Y	Cerrorpen es Grande positivo	Entonces	Controlpen es Cero
6	Si	Errorpen es Pequeño negativo	Y	Cerrorpen es Grande negativo	Entonces	Controlpen es Grande positivo
7	Si	Errorpen es Pequeño negativo	Y	Cerrorpen es Pequeño negativo	Entonces	Controlpen es Grande positivo
8	Si	Errorpen es Pequeño negativo	Y	Cerrorpen es Cero	Entonces	Controlpen es Pequeño positivo
9	Si	Errorpen es Pequeño negativo	Y	Cerrorpen es Pequeño positivo	Entonces	Controlpen es Cero
10	Si	Errorpen es Pequeño negativo	Y	Cerrorpen es Grande positivo	Entonces	Controlpen es Pequeño negativo
11	Si	Errorpen es Cero	Y	Cerrorpen es Grande negativo	Entonces	Controlpen es Grande positivo
12	Si	Errorpen es Cero	Y	Cerrorpen es Pequeño negativo	Entonces	Controlpen es Pequeño positivo
13	Si	Errorpen es Cero	Y	Cerrorpen es Cero	Entonces	Controlpen es Cero
14	Si	Errorpen es Cero	Y	Cerrorpen es Pequeño positivo	Entonces	Controlpen es Pequeño negativo
15	Si	Errorpen es Cero	Y	Cerrorpen es Grande positivo	Entonces	Controlpen es Grande negativo
16	Si	Errorpen es Pequeño positivo	Y	Cerrorpen es Grande negativo	Entonces	Controlpen es Pequeño positivo
17	Si	Errorpen es Pequeño positivo	Y	Cerrorpen es Pequeño negativo	Entonces	Controlpen es Cero
18	Si	Errorpen es Pequeño positivo	Y	Cerrorpen es Cero	Entonces	Controlpen es Pequeño negativo
19	Si	Errorpen es Pequeño positivo	Y	Cerrorpen es Pequeño positivo	Entonces	Controlpen es Grande negativo
20	Si	Errorpen es Pequeño positivo	Y	Cerrorpen es Grande positivo	Entonces	Controlpen es Grande negativo
21	Si	Errorpen es Grande positivo	Y	Cerrorpen es Grande negativo	Entonces	Controlpen es Cero
22	Si	Errorpen es Grande positivo	Y	Cerrorpen es Pequeño negativo	Entonces	Controlpen es Pequeño negativo
23	Si	Errorpen es Grande positivo	Y	Cerrorpen es Cero	Entonces	Controlpen es Grande negativo
24	Si	Errorpen es Grande positivo	Y	Cerrorpen es Pequeño positivo	Entonces	Controlpen es Grande negativo
25	Si	Errorpen es Grande positivo	Y	Cerrorpen es Grande positivo	Entonces	Controlpen es Grande negativo

Tabla. 4. 7. Base de reglas del controlador difuso PD péndulo

Donde:

- *Errorpen*: diferencia entre la posición angular deseada del péndulo y la posición angular real del péndulo.

$$Errorpen = posición_deseada(\theta') - posición_real(\theta) \quad (4.1)$$

De esta manera los diferentes valores que puede adquirir *Errorpen* representan el momento que el péndulo tiene un ángulo a la izquierda o derecha con respecto a la línea vertical de referencia.

- *Cerrorpen* (velocidad angular ω): derivada del error (*Errorpen*).
- *Controlpen*: señal de control.

Como se puede observar en la Tabla. 4. 7., las proposiciones están conectadas con el operador “Y”, además de esto se encuentran los valores lingüísticos “grande negativo”, “pequeño negativo”, “cero”, “pequeño positivo”, “grande positivo”. Así también a la base de reglas se la puede representar como se muestra en la Tabla. 4. 8.

"Fuerza" u		"Cambio de Error"				
		GN	PN	Z	PP	GP
"Error"	GN	GP	GP	GP	PP	Z
	PN	GP	GP	PP	Z	PN
	Z	GP	PP	Z	PN	GN
	PP	PP	Z	PN	GN	GN
	GP	Z	PN	GN	GN	GN

Tabla. 4. 8. Base de reglas, controlador difuso PD péndulo

Al observar la Tabla. 4. 8. se puede resaltar el hecho de que existe una diagonal de ceros y a su vez una simetría en el cuerpo de la misma. Esta simetría está dada por la simetría dinámica que existe en el funcionamiento del sistema.

Para el *controlador difuso PD péndulo* cada una de las reglas difusas califica una diferente configuración del péndulo, como se muestra en la Figura. 4. 6.

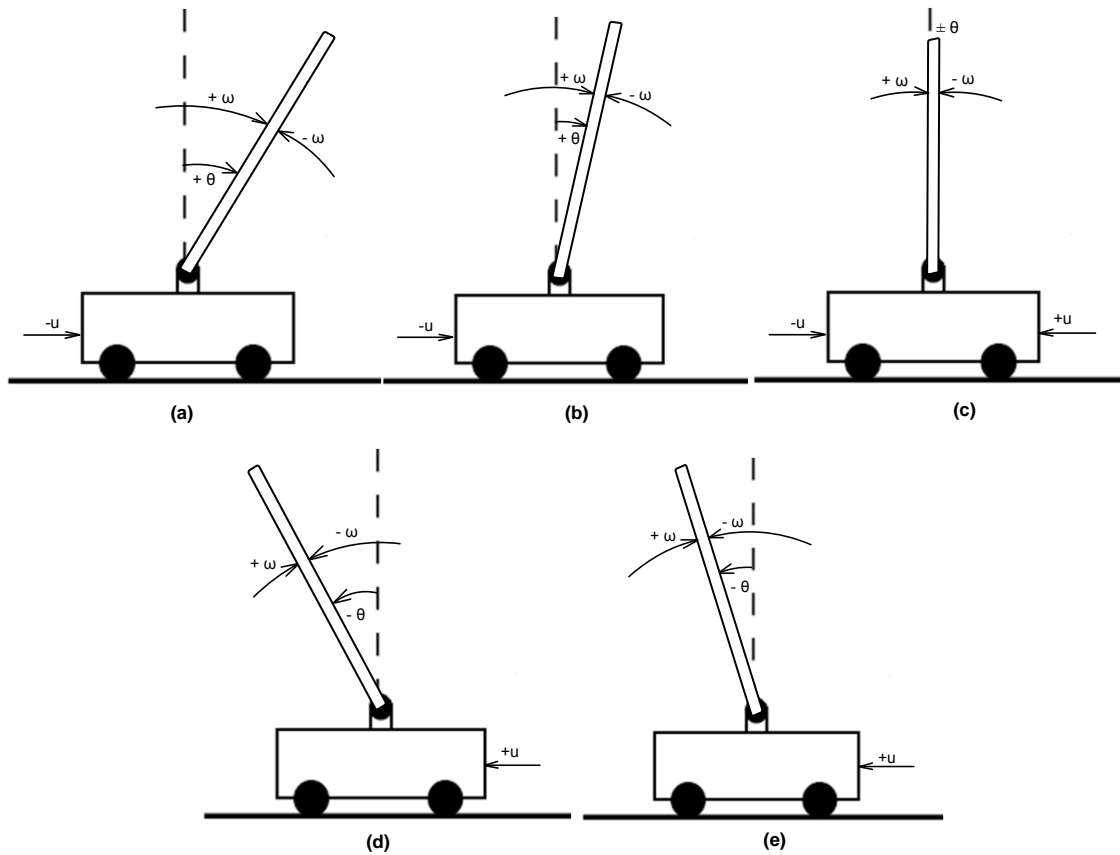


Figura. 4. 6. Posiciones del péndulo

Para poder balancear al péndulo y que llegue a la zona de estabilización, el carro debe moverse de tal manera que si el péndulo está al lado derecho del carro este debe moverse hacia el lado izquierdo del riel y cuando el péndulo está al lado izquierdo del carro este debe moverse hacia el lado derecho del riel.

Ahora el análisis que se procede a realizar es cuando el péndulo alcanza la zona de estabilización que es cuando empieza a funcionar el *controlador difuso PD péndulo*. Es así que cuando el péndulo está al lado derecho del carro este debe moverse hacia el lado derecho de riel y cuando el péndulo está al lado izquierdo del carro este debe moverse

hacia el lado izquierdo del riel. Pero cada vez que el péndulo se balancea este adquiere una velocidad angular ($Cerrorpen$) ya sea positiva o negativa para llegar a determinado punto dentro de la zona de estabilización por lo que se procede analizar su influencia dentro del proceso de control.

1. La Figura. 4. 6. (a) es la representación cuando $Errorpen$ es GN y $Cerrorpen$ es GN entonces se ejerce sobre el carro una fuerza GP. Es decir cuando el péndulo tiene un ángulo θ grande positivo respecto a la vertical de referencia y se está desplazando con una velocidad angular ω grande positivo en sentido de las manecillas del reloj entonces se aplica una fuerza GN hacia la derecha del riel para que el péndulo se pueda ubicar en la posición correcta. Ahora si $Errorpen$ es GN y $Cerrorpen$ es GP entonces se ejerce sobre el carro una fuerza Z que permite una llegada suave a la posición vertical estable.
2. La Figura. 4. 6. (b) es la representación cuando $Errorpen$ es PN y $Cerrorpen$ es PN entonces se ejerce sobre el carro una fuerza GP. Es decir cuando el péndulo tiene un ángulo θ pequeño positivo y una velocidad angular ω pequeño positivo entonces se aplica una fuerza GN hacia la derecha del riel. Ahora si $Errorpen$ es PN y $Cerrorpen$ es PP entonces se ejerce sobre el carro una fuerza Z.
3. La Figura. 4. 6. (c) es la representación cuando $Errorpen$ es Z y $Cerrorpen$ es PN o PP entonces se ejerce sobre el carro una fuerza PP y PN respectivamente. Es decir cuando el péndulo tiene un ángulo θ cero (la cuantificación lingüística Z no implica que $Errorpen$ sea exactamente cero) y una velocidad angular ω pequeño positivo o

pequeño negativo entonces se aplica una fuerza PN hacia la derecha del riel y PP hacia la izquierda del riel respectivamente, que son de magnitud pequeña puesto que en este instante el péndulo se encuentra en la posición vertical estable.

4. La Figura. 4. 6. (d) es la representación cuando Errorpen es GP y Cerrorpen es GN entonces se ejerce sobre el carro una fuerza Z. Ahora si Errorpen es GP y Cerrorpen es GP entonces se ejerce sobre el carro una fuerza GN. Es decir cuando el péndulo tiene un ángulo θ grande negativo y una velocidad angular ω grande negativo entonces se aplica una fuerza GP hacia la izquierda del riel.
5. La Figura. 4. 6. (e) es la representación cuando Errorpen es PP y Cerrorpen es PN entonces se ejerce sobre el carro una fuerza Z. Ahora si Errorpen es PP y Cerrorpen es PP entonces se ejerce sobre el carro una fuerza GN. Es decir cuando el péndulo tiene un ángulo θ pequeño negativo y una velocidad angular ω pequeño negativo entonces se aplica una fuerza GP hacia la izquierda del riel.

Si bien es cierto no se mencionan a todas las reglas, se ha considerado analizar algunas de ellas para obtener un conocimiento más claro del funcionamiento del controlador difuso.

La estructura de trabajo de las reglas difusas en el *controlador difuso PD carro* es idéntico al explicado en el *controlador difuso PD péndulo*, con la diferencia que ahora se considera al carro en la representación de los valores lingüísticos y no al péndulo (Figura. 4. 6.).

4.1.5 Implementación del controlador difuso PD con balanceo en SIMULINK

Se procede a explicar los bloques inmersos en la implementación de los controladores difusos PD, en el modelo de simulación del sistema carro - péndulo que se muestran en la Figura. 4. 7.

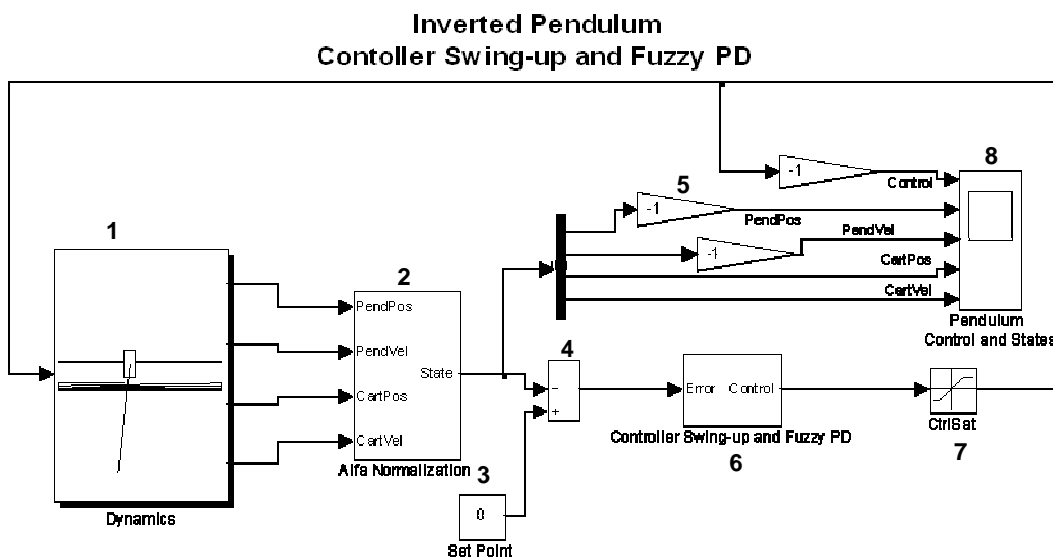


Figura. 4. 7. Implementación controlador difuso PD con balanceo, Modelo simulación

1. Bloque que contiene el modelo matemático de simulación del sistema carro – péndulo.
2. Bloque donde se toma la señal de posición angular del péndulo realizando con ella el cálculo del ángulo suplementario para el caso en el que su valor se encuentre entre el tercer y cuarto cuadrante con valor negativo y cuando se encuentre entre el primero y segundo cuadrante obteniéndose el mismo ángulo con valor positivo todo esto en radianes usando la función *atan2* de Matlab como se muestra en la Figura. 4. 8. Todo este proceso se realiza con el objetivo de obtener la señal referida al origen de coordenadas y comprendida entre

los valores límites de π y $-\pi$ obteniendo así una señal manejable para llegar al set point deseado.

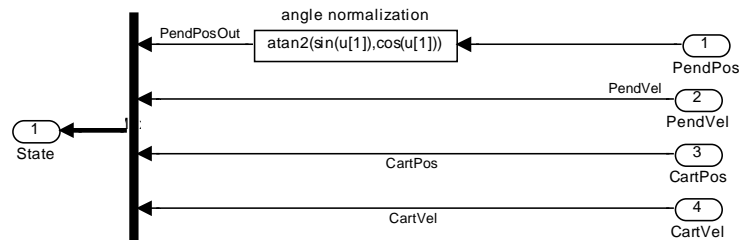


Figura. 4. 8. Normalización posición angular del péndulo, Modelo simulación

3. Set Point de valor cero para ubicar al carro y al péndulo, en la mitad del riel y en la posición vertical estable respectivamente.

4. Bloque que realiza la resta entre el valor obtenido en la posición del carro, la posición angular del péndulo con el valor deseado, es decir que a la salida de este bloque de substracción se va a obtener el error que se obtuvo al poner un valor deseado con el valor que se obtiene en la simulación. Además se incluye en este proceso las señales velocidad angular del péndulo y velocidad del carro para de este modo guardar coherencia en la lógica que se dispone para las cuatro señales que ingresan a los controladores difusos respectivamente.

5. Ganancias ubicadas en las señales de monitoreo del proceso, para simular el funcionamiento del modelo experimental.

6. Bloque controlador basado en reglas para controlar la posición del carro y del péndulo. Al interior de este bloque se encuentra el controlador difuso PD péndulo y el controlador

difuso PD carro los mismos que tienen en sus entradas a más de los errores de posición, las señales del cambio de error de posición del péndulo y el cambio de error de posición del carro respectivamente, como se muestra en la Figura. 4. 9.

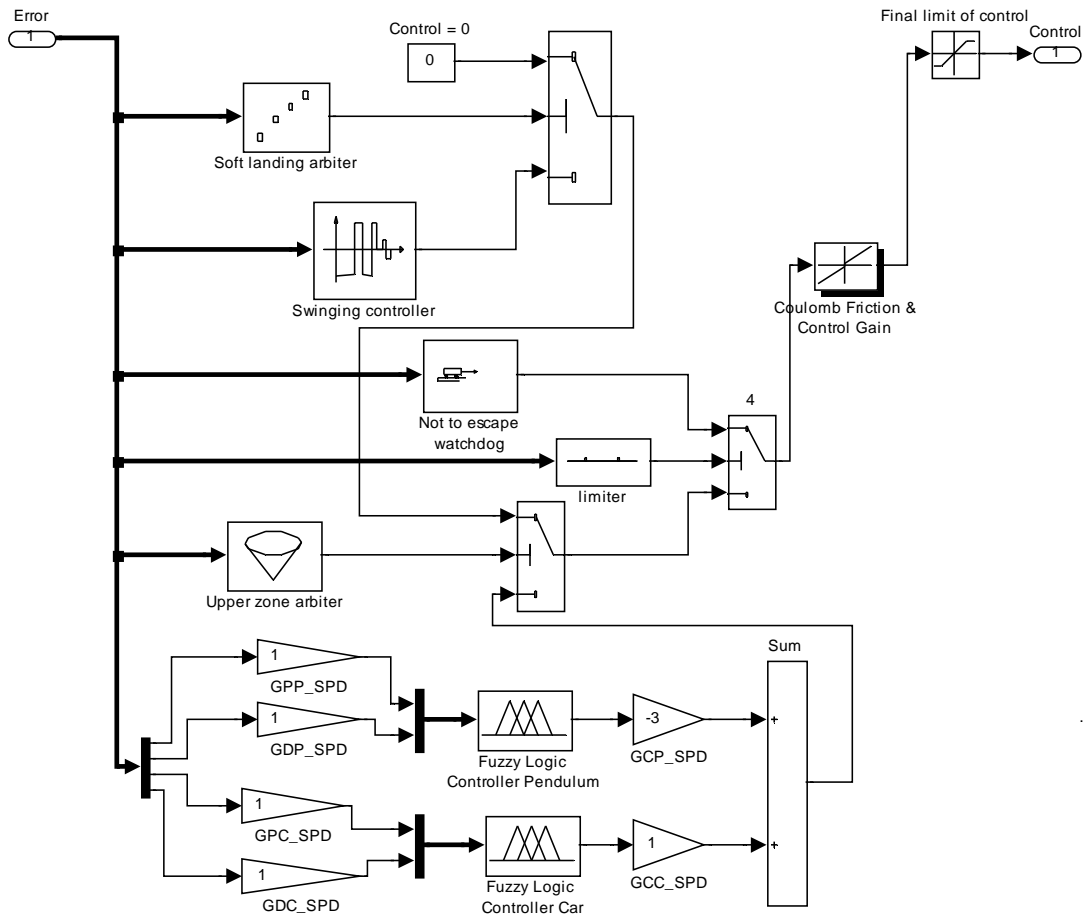


Figura. 4. 9. Implementación controlador difuso PD con balanceo, Modelo simulación

7. Bloque encargado de limitar la señal de control en el rango [-0.2, 0.2].

8. Visualizador que indica la señal de control, la señal posición angular del péndulo, la señal velocidad del péndulo, la señal posición del carro, la señal velocidad del carro.

4.1.6 Simulación del controlador difuso PD con balanceo en SIMULINK

Los controladores difusos PD tienen como misión cumplir el objetivo de control, corrigiendo el error con la parte proporcional y además previsorio lo que hace más rápida la acción de control, aunque exista error en estado estacionario.

Antes de iniciar la simulación se debe escribir en el espacio de trabajo de Matlab PCS lo que permite cargar la interfaz gráfica y los archivos de extensión “.fis” realizados para este proyecto. La interfaz gráfica es amigable con el usuario permitiendo un fácil manejo de sus opciones.

Los resultados de simulación se observa en la Figura. 4. 10.

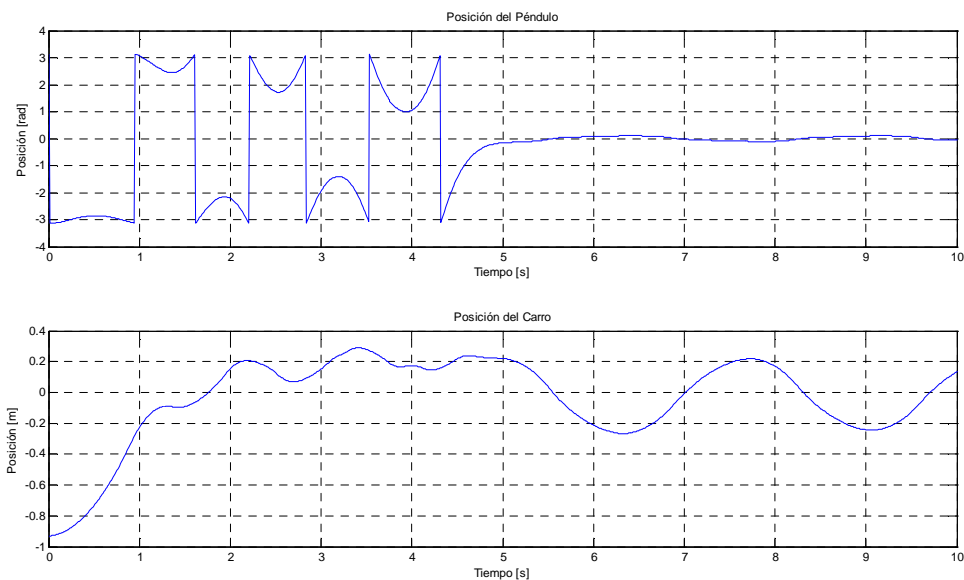


Figura. 4. 10. Difuso PD con balanceo, Modelo Simulación, Posición péndulo y Posición carro

Como se puede observar en la Figura. 4. 10. el objetivo de control se ha logrado. La señal de posición del péndulo presenta error en estado estacionario. La señal de posición del carro también presenta error en estado estacionario que se manifiesta en oscilaciones.

Esta simulación está realizada en un tiempo de 10 [s] con los siguientes valores en las ganancias:

Controlador difuso PD péndulo:

- Ganancia proporcional (GPP_SPD)= 1
- Ganancia derivativa (GDP_SPD) = 1
- Ganancia de control (GCP_SPD) = -3

Controlador difuso PD carro:

- Ganancia proporcional (GPC_SPD) = 1
- Ganancia derivativa (GDC_SPD) = 1
- Ganancia de control (GCC_SPD) = 1

4.1.7 Ajuste del controlador difuso PD con balanceo en SIMULINK

Las ganancias asociadas a cada una de las variables son necesarias ya que permiten realizar un ajuste en el universo de discurso asociadas a ellas.

Antes de realizar cambios en cualquier ganancia de cualquier controlador se puede observar que cuando todas las ganancias tienen valor unitario, en la posición del péndulo que este gire por instantes en sentido horario o antihorario demostrando que existe un control sobre el péndulo, siendo necesario aumentar la ganancia de control.

El ajuste se realiza en dos etapas en la primera etapa se empieza con variaciones en la ganancia de control para el *controlador difuso péndulo*, en la segunda etapa si no se logra el objetivo de control, entonces se realiza la variación en la ganancia de control para el *controlador difuso carro*. La variación de las ganancias restantes no tiene mayor influencia en **simulación**. Las ganancias empiezan a ser evaluadas desde el valor unitario usando el modo manual (prueba y error) e incrementándose en pasos de uno.

Para el caso del *controlador difuso PD péndulo* la ganancia de control debe ser negativa ya que esta cumple con el funcionamiento experimental del motor DC del sistema, la ganancia para la parte proporcional y derivativa debe ser positiva ya que estas corresponden con la lógica dispuesta en las reglas difusas.

Para el caso del *controlador difuso PD carro* la ganancia de control debe ser negativa cumpliendo con el funcionamiento experimental del motor DC del sistema, sin embargo en el modelo de simulación se ha probado que debe trabajar con valor positivo, la ganancia para la parte proporcional y derivativa debe ser positiva ya que estas corresponden con la lógica dispuesta en las reglas difusas.

Tras ajustar las ganancias en los dos controladores se obtiene los siguientes valores:

Controlador difuso PD péndulo:

- Ganancia proporcional (GPP_SPD) = 1
- Ganancia derivativa (GDP_SPD) = 1
- Ganancia de control (GCP_SPD) = -3

Controlador difuso PD carro:

- Ganancia proporcional (GPC_SPD) = 1
- Ganancia derivativa (GDC_SPD) = 1
- Ganancia de control (GCC_SPD) = 1

Estos valores de ganancias serán tomados como punto de referencia para el ajuste en el modelo experimental

4.1.8 Implementación del controlador difuso PD con balanceo en el modelo experimental

Se procede a explicar los bloques inmersos en la implementación de los controladores difusos PD, en el modelo de experimentación del sistema carro - péndulo que se muestran en la Figura. 4. 11.

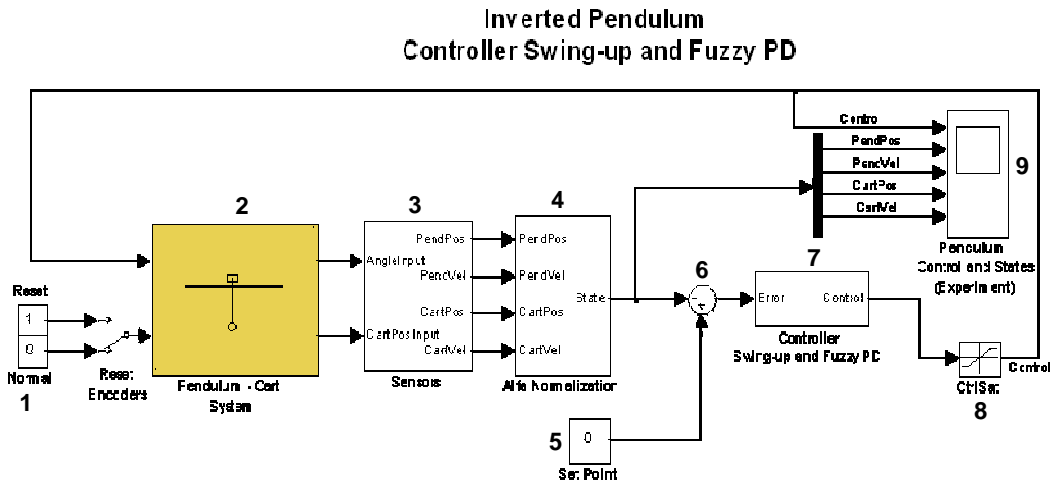


Figura. 4. 11. Implementación controlador difuso PD con balanceo, Modelo experimentación

1. Bloque donde se puede seleccionar el restablecimiento de los encoders o continuar con el experimento.
2. Bloque que contiene el controlador del dispositivo (*device driver*) que sirve de medio de enlace para la comunicación de datos desde Matlab hacia la tarjeta de adquisición de datos RT-DAC4/PCI-D y viceversa.
3. Bloque de sensores que tiene como entradas la posición del carro y la posición angular del péndulo, para obtener de ellas la velocidad del carro y la velocidad angular del péndulo respectivamente.
4. Bloque donde se toma la señal de posición angular del péndulo realizando con ella el cálculo del ángulo suplementario para el caso en el que su valor se encuentre entre el tercer y cuarto cuadrante con valor negativo y cuando se encuentre entre el primero y segundo cuadrante obteniéndose el mismo ángulo con valor positivo todo esto en radianes usando la

función *atan2* de Matlab como se muestra en la Figura. 4. 12. Todo este proceso se realiza con el objetivo de obtener la señal referida al origen de coordenadas y comprendida entre los valores límites de π y $-\pi$. A esta señal se suma el valor π como condición inicial de posición angular del péndulo.

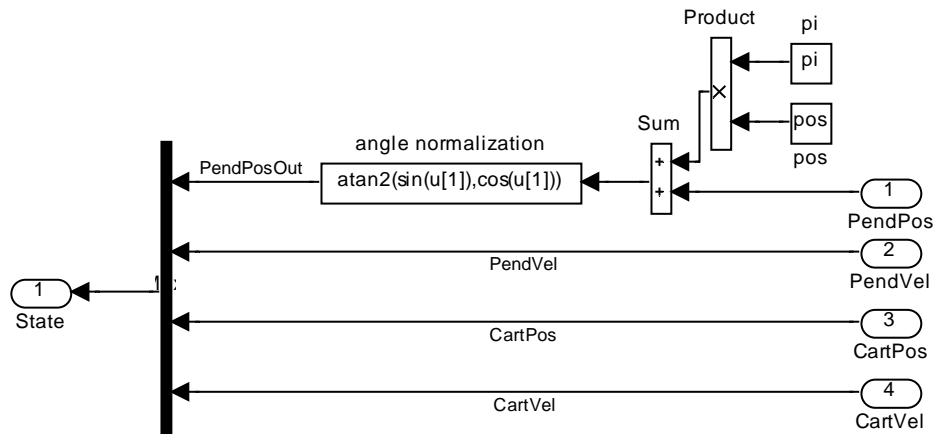


Figura. 4. 12. Normalización posición angular del péndulo, Modelo experimentación

5. Set Point de valor cero para ubicar al carro y al péndulo, en la mitad del riel y en la posición vertical estable respectivamente.

6. Bloque que realiza la resta entre el valor obtenido en la posición del carro, la posición angular del péndulo con el valor deseado, es decir que a la salida de este bloque de substracción se va a obtener el error que se obtuvo al poner un valor deseado con el valor que se obtiene en la simulación. Además se incluye en este proceso las señales velocidad angular del péndulo y velocidad del carro para de este modo guardar coherencia en la lógica que se dispone para las cuatro señales que ingresan a los controladores difusos respectivamente.

7. Bloque controlador basado en reglas para controlar la posición del carro y posición angular del péndulo. Al interior de este bloque se encuentra el controlador difuso PD péndulo y el controlador difuso PD carro con la misma estructura que se realizó en la simulación.

El bloque *limiter* tiene como valor límite 0.5 [m], puesto que debe existir una brecha entre este valor y el valor límite de la variable error de posición del carro que en este caso es 0.35 [m] para que cuando el carro llegue a este punto no permanezca oscilando indefinidamente.

El bloque *upper zone arbiter* tiene por defecto el valor 0.2 [rad], lo que para el rango de la variable error de posición angular del péndulo es incorrecto, decidiéndose ubicar el valor 0.3 [rad].

8. Bloque encargado de limitar la señal de control en el rango [-0.4, 0.4].

9. Visualizador que indica la señal de control, la señal posición angular del péndulo, la señal velocidad del péndulo, la señal posición del carro, la señal velocidad del carro.

4.1.9 Experimentación del controlador difuso PD con balanceo

Antes poner en marcha cualquier modelo de experimentación se debe construir el modelo RTW para luego conectar con el objetivo e iniciar.

Los valores de referencia con los que se empieza el experimento corresponden a los encontrados en simulación. Tras realizar algunas pruebas sobre el sistema se determina que los valores de ganancias tratan de cumplir con el objetivo de control sobre el sistema pero es necesario realizar algunos ajustes.

4.1.10 Ajuste experimental del controlador difuso PD con balanceo

Antes de realizar un ajuste en las ganancias de los controladores difusos PD se analiza el cumplimiento del objetivo de control con los valores de ganancias obtenidas en la simulación. Tras varias pruebas realizadas con distintos valores de ganancia en los dos controladores, se concluye que es necesario variar las ganancias únicamente en el *controlador difuso PD péndulo* más no en el *controlador difuso PD carro*, es decir dar **mayor prioridad** al posicionamiento del péndulo como parte esencial para llegar al objetivo de control.

Partiendo del hecho de querer obtener ciertos valores de ganancias fijas para poder sintonizar al controlador difuso PD péndulo, se realizan varios cambios en las ganancias obteniéndose distintos comportamientos del sistema que si bien por un parte cumplen con el objetivo de control, por otra no se llega a realizarlo de manera correcta. Surge entonces la necesidad de poder obtener ciertos valores de sintonización más óptimos para el sistema, apoyándose en la toma de valores correspondientes al criterio de desempeño *integral del error cuadrático (CIEC)* basado en la siguiente ecuación:

$$\int_0^{\infty} e^2(t). dt \quad (4.2)$$

Este criterio de desempeño permite evaluar el comportamiento del sistema y son la base para optimizarlos respecto del comportamiento de su error en estado transitorio. La aplicación de este criterio en el control del sistema es el equivalente a la suma del CIEC generado por el péndulo y el carro, como se observa en la Tabla. 4. 9.

Ganancia proporcional (GPP SPD)	Ganancia derivativa (GDP SPD)	Ganancia de control (GCP SPD)	CIEC posición angular del péndulo	CIEC posición del carro	CIEC TOTAL
1	1	-3	20.61	1.464	22.07
1.7	0.5	-3	9.383	0.5292	9.912
1.6	0.5	-3	9.387	0.7054	10.09
1.5	0.5	-3	9.198	0.7015	9.899
1.5	0.7	-3	9.265	0.6479	9.912
1.2	0.6	-3	9.426	0.5483	9.974
1.3	0.7	-3	9.226	0.5334	9.76
1.7	0.8	-3	9.393	0.5998	9.993
1.5	0.9	-3	9.322	0.5396	9.862
1.8	1	-2.5	9.287	0.553	9.84
1.6	1	-2.5	9.388	0.5175	9.906

Tabla. 4. 9. CIEC del controlador difuso PD con balanceo

Como se observa en la Tabla. 4. 9. el CIEC más óptimo corresponde al valor 9.76 con sus respectivas ganancias, mismo que sirve como referencia al momento de demostrar el control sobre la planta.

Todos los valores que presenta la Tabla. 4. 9. cumplen con el objetivo de control aunque muestran diferentes tipos de respuesta notorios al momento de la experimentación. El tiempo de experimentación para estas pruebas es de 10 [s] con el siguiente rango de variación: ganancia proporcional [0.2, 2], ganancia derivativa [0.2, 2], ganancia de control [-2.5, -3.5].

Ahora considerando el criterio *Ajuste a través de los universos de discurso*^[24]. Se señala que las ganancias en las entradas del controlador difuso afectan el desempeño de los universos de discurso de la siguiente manera:

- Si $g=1$, no existe efecto en las funciones de pertenencia.
- Si $g<1$, las funciones de pertenencia son extendidas horizontalmente de manera uniforme por un factor de $\frac{1}{g}$.
- Si $g>1$, las funciones de pertenencia son contraídas horizontalmente de manera uniforme por un factor de $\frac{1}{g}$.

También se señala que el universo de discurso en la salida del controlador difuso tiene el siguiente comportamiento:

- Si $g=1$, no existe efecto en las funciones de pertenencia.
- Si $g<1$, las funciones de pertenencia son contraídas horizontalmente de manera uniforme.
- Si $g>1$, las funciones de pertenencia son extendidas horizontalmente de manera uniforme.

^[24] Passino, Kevin y Yurkovich, Stephen. *Fuzzy control*. 1998. (12).

Entonces se procede a realizar un análisis de las ganancias obtenidas en la Tabla. 4.9. Los valores de ganancia proporcional son mayores que “1”, indicando la disminución del universo de discurso es decir menor a 0.3 [rad]. Los valores de ganancia derivativa son menores que “1”, indicando el aumento del universo de discurso es decir realiza un control más rápido. Los valores de ganancia de control tienden a permanecer en “3” para de este modo funcionar con el desempeño que brindan el resto de ganancias del controlador, puesto que si se lo cambia por arriba o debajo de este valor también cambia el resto de ganancias.

4.2 CONTROLADOR DIFUSO PD

El objetivo en esta sección es probar el *controlador difuso PD péndulo* y el *controlador difuso PD carro* sin el controlador *Swing – up*. Estos controladores trabajan juntos de modo que al soltar el péndulo desde la posición vertical estable y desde el centro del riel, deben proporcionar la permanencia en estas posiciones.

Los controladores difusos que se utilizan en esta sección son los mismos que se definieron en la sección 4.1.1., 4.1.2, 4.1.3, 4.1.4.

4.2.1 Implementación del controlador difuso PD en SIMULINK

Los bloques inmersos en la implementación de los controladores difusos PD, en el modelo de simulación del sistema carro – péndulo son los mismos que se explicaron en la

sección 4.1.5. con la diferencia que ahora no se encuentra el controlador *Swing – up* como se muestra en la Figura. 4. 13.

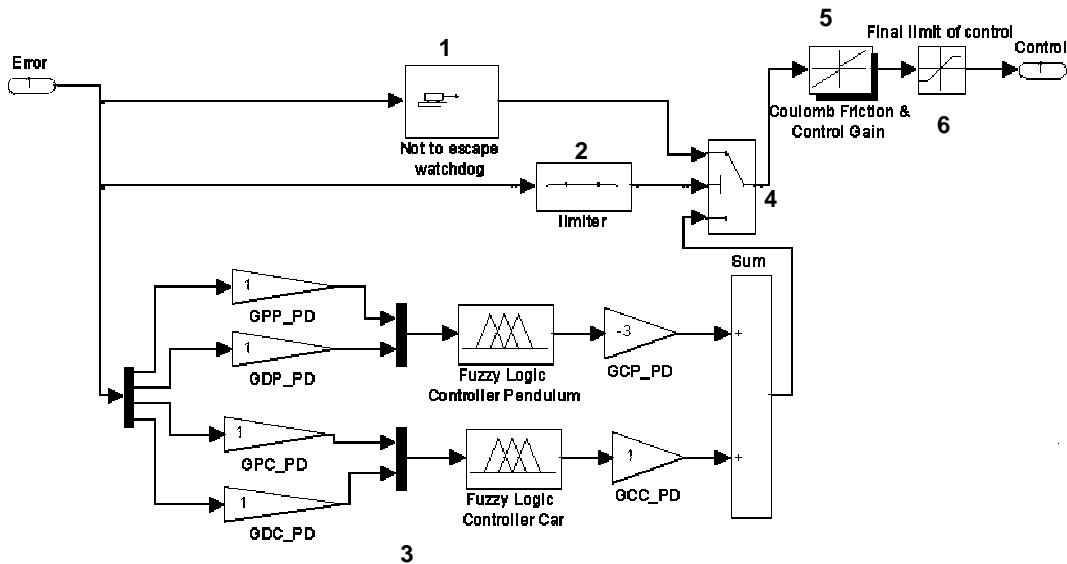


Figura. 4. 13. Implementación controlador difuso PD, Modelo simulación

1. Bloque que genera el opuesto a la dirección de la fuerza de movimiento máxima aplicada al carro.
2. Bloque que refuerza el centrado del carro y previene que este llegue a los extremos del riel.
3. Controlador difuso PD péndulo y Controlador difuso PD carro.
4. Bloque conmutador entre los datos de entrada “1” y “3”, dependiendo de la señal “2”. Es decir si el carro está dentro de 0.35 [m] actúa el control difuso, caso contrario actúa la máxima fuerza contraria al movimiento del carro.

5. Bloque que genera un offset en la señal total de control para iniciar desde la mínima fuerza con la que el motor rompe la inercia.
6. Bloque encargado de limitar la señal de control en el rango $[-1, 1]$, que son los valores máximos a los cuales se puede alcanzar en todo el proceso.

4.2.2 Simulación del controlador difuso PD en SIMULINK

Los resultados de la simulación se observa en la Figura. 4. 14.

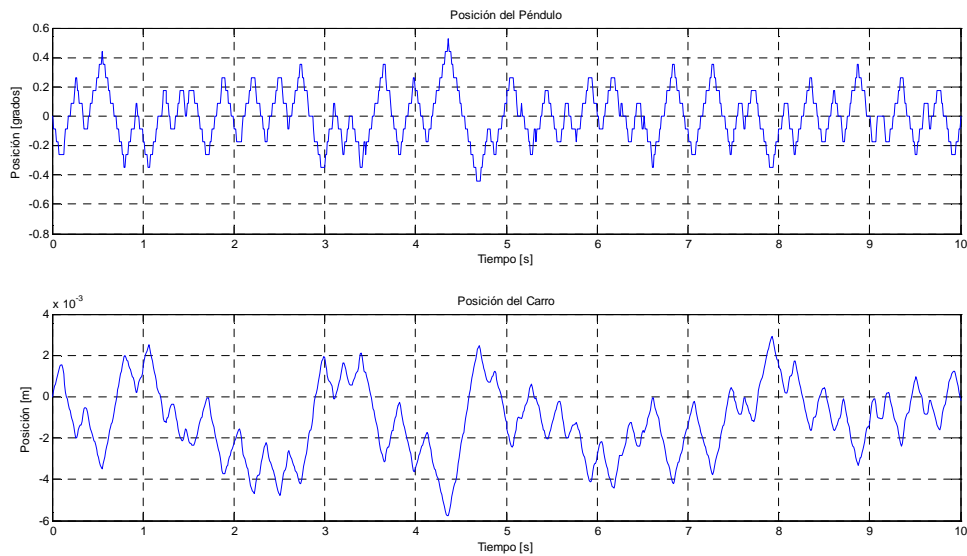


Figura. 4. 14. Difuso PD, Modelo Simulación, Posición péndulo y Posición carro

Como se puede observar en la Figura. 4. 14. el objetivo de control se ha logrado. El error presente en las dos señales de posición son despreciables.

Esta simulación está realizada en un tiempo de 10 [s] con los siguientes valores en las ganancias:

Controlador difuso PD péndulo:

- Ganancia proporcional (GPP_PD) = 1
- Ganancia derivativa (GDP_PD) = 1
- Ganancia de control (GCP_PD) = -3

Controlador difuso PD carro:

- Ganancia proporcional (GPC_PD) = 1
- Ganancia derivativa (GDC_PD) = 1
- Ganancia de control (GCC_PD) = 1

4.2.3 Implementación del controlador difuso PD en el modelo experimental

La implementación de los controladores difusos PD en el modelo de experimentación del sistema carro – péndulo, es semejante al realizado para el modelo de simulación (sección 4. 2. 1.), con la diferencia que ahora contiene el controlador del dispositivo (*device driver*) a cambio del modelo matemático.

4.2.4 Experimentación del controlador difuso PD

Antes de realizar cualquier cambio en las ganancias de los controladores difusos PD se analiza el cumplimiento del objetivo de control con los valores de ganancias impuestas en simulación. En efecto se cumple con el objetivo de control aunque con oscilaciones alrededor del centro del riel.

Se varían únicamente las ganancias en el *controlador difuso PD péndulo* más no en el *controlador difuso PD carro*, ya que tiene **mayor prioridad** para la ubicación del péndulo en la posición vertical estable.

El criterio de desempeño *integral del error cuadrático (CIEC)* aplicado al controlador difuso PD, da como resultado los valores que se muestran en la Tabla. 4. 10.

Ganancia proporcional (GPP PD)	Ganancia derivativa (GDP PD)	Ganancia de control (GCP PD)	CIEC posición angular del péndulo	CIEC posición del carro	CIEC TOTAL
1	1	-3	0.003703	0.03892	0.04262
1	0.5	-3	0.001961	0.01449	0.01646
1	0.8	-3	0.006229	0.04871	0.05494
1.2	1	-3	0.006475	0.09828	0.1048
1.3	1	-3	0.01166	0.1807	0.1924
1	0.7	-3.5	0.003497	0.05584	0.05934
1	0.6	-3.5	0.003797	0.05128	0.05508
1.7	0.5	-3	0.001061	0.02457	0.02563
1.6	0.5	-3	0.002258	0.03009	0.03235
1.8	1.0	-2.5	0.002493	0.04975	0.05224
1.6	1.0	-2.5	0.002552	0.03806	0.04061

Tabla. 4. 10. CIEC del controlador difuso PD

Como se observa en la Tabla. 4. 10. el CIEC más óptimo corresponde al valor 0.01646 con sus respectivas ganancias, mismo que sirve como referencia al momento de demostrar este control sobre la planta.

4.3 CONTROLADOR DIFUSO PI

El objetivo en esta sección es probar un controlador difuso PD para la posición angular del péndulo y un controlador difuso PI para la posición del carro sin el controlador *Swing – up*. Estos controladores trabajan juntos de modo que al soltar el péndulo desde la posición vertical estable y desde el centro del riel, estos deben proporcionar la permanencia en estas posiciones.

El controlador difuso PD para la posición angular del péndulo es el mismo que se ha definido en la sección 4.1.1., 4.1.2, 4.1.3, 4.1.4. Entonces se procede a diseñar un controlador difuso PI para la posición del carro. Este controlador tiene dos entradas y una salida. Las señales de salida de los dos controladores se suman conformando una sola señal de control.

Las reglas difusas tiene la siguiente forma:

- ***SI*** proposición ***Y*** proposición ***entonces*** consecuencia

4.3.1 Definición de variables y valores lingüísticos

Las variables asignadas para el control de posición del carro, se indican en la Tabla.

4. 11.

CONTROLADOR DIFUSO PI CARRO		
Tipo	Variable	Etiqueta
Entrada	Error de posición del carro	Erroricar
Entrada	Integral del error de posición del carro	Integralicar
Salida	Control	Controlicar

Tabla. 4. 11. Variables del controlador difuso PI carro

Los valores lingüísticos que han sido escogidos para las variables de entrada y salida del controlador difuso PI carro son cinco como se indica en la Tabla. 4.12.

Valor lingüístico	Etiqueta
Grande negativo	GN
Pequeño negativo	PN
Cero	Z
Pequeño positivo	PP
Grande positivo	GP

Tabla. 4. 12. Valores lingüísticos del controlador difuso PI carro

4.3.2 Rango de variables

Para cada variable del controlador difuso PI carro se han definido los rangos en los que trabajan, como se muestra en la Tabla. 4. 13.

CONTROLADOR DIFUSO PI CARRO	
Variables	Rango
Error de posición del carro	[-0.35, 0.35] [m]
Integral del error de posición del carro	[-0.5, 0.5] [m.s]
Control	[-0.2, 0.2]

Tabla. 4. 13. Rango de variables del controlador difuso PI carro

4.3.3 Funciones de pertenencia

Las funciones de pertenencia utilizadas en los conjuntos difusos del *controlador difuso PI carro* son de tipo triangular y exponencial. Estas funciones de pertenencia son asignadas a los conjuntos difusos de entrada y salida.

Como se puede observar en la Figura. 4. 15.:

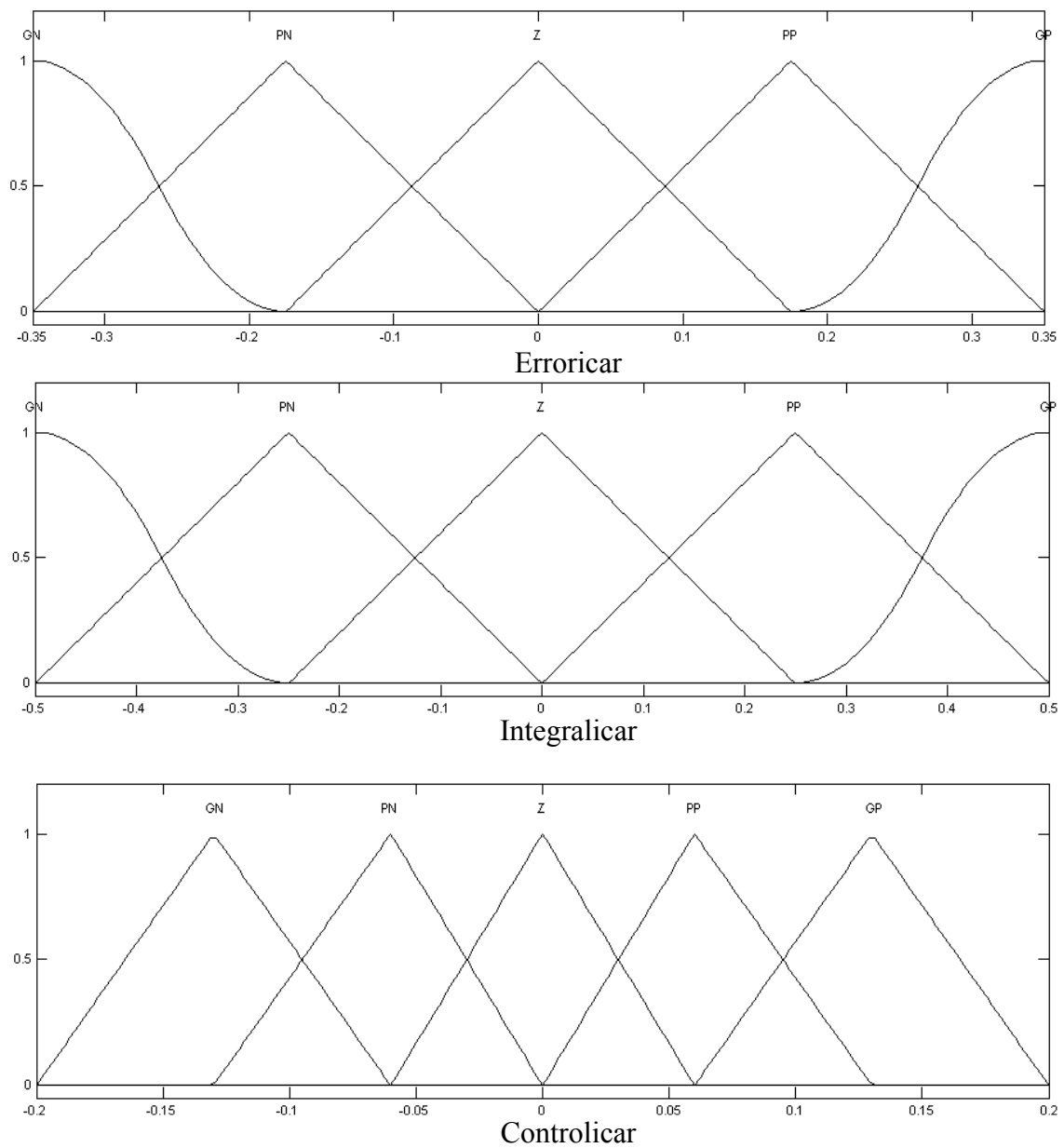


Figura. 4. 15. Funciones de pertenencia, Controlador difuso PI carro

- *Erroricar e Integralicar*: se ha elegido funciones de pertenencia de tipo triangular para los valores lingüísticos medios, mientras que para las fronteras de tipo exponencial, asegurando una cobertura de control fuera de los límites establecidos.
- *Controlicar*: se ha elegido funciones de pertenencia de tipo triangular para todos los valores lingüísticos. Estas funciones están dispuestas de forma equidistante hasta el límite establecido por la variable consiguiendo la superficie de control que se muestra en la Figura. 4. 16.

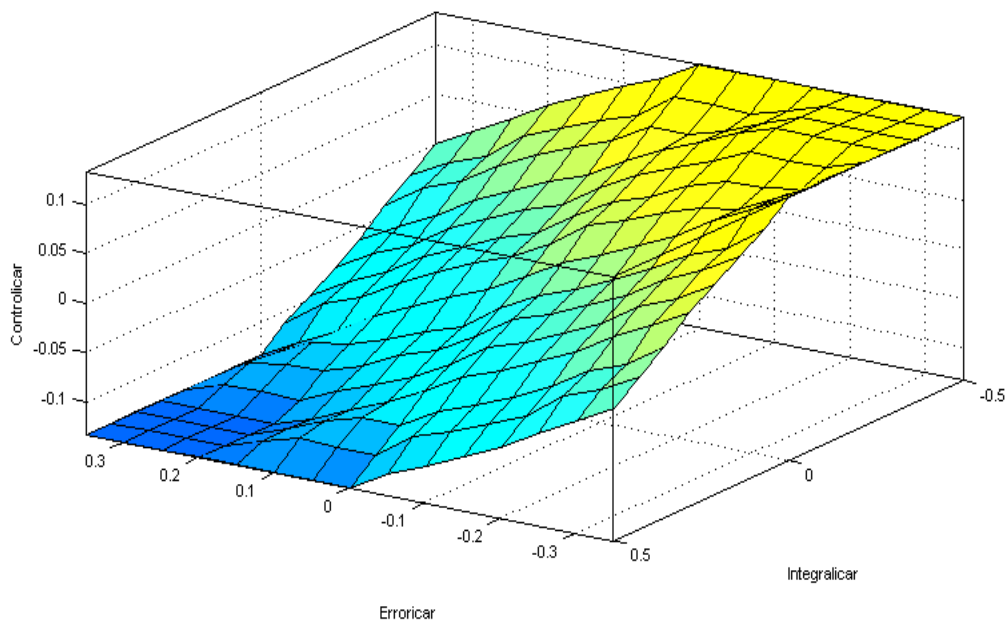


Figura. 4. 16. Superficie de control del controlador difuso PI carro

La superficie de control que muestra la Figura. 4. 16. indica una linealidad en su extensión.

4.3.4 Base de reglas

En este controlador difuso PI se ha dispuesto trabajar con 25 reglas que se presentan en la Tabla. 4. 14.

1	Si	Erroricar es Grande negativo	Y	Integralicar es Grande negativo	Entonces	Controlicar es Grande positivo
2	Si	Erroricar es Grande negativo	Y	Integralicar es Pequeño negativo	Entonces	Controlicar es Grande positivo
3	Si	Erroricar es Grande negativo	Y	Integralicar es Cero	Entonces	Controlicar es Grande positivo
4	Si	Erroricar es Grande negativo	Y	Integralicar es Pequeño positivo	Entonces	Controlicar es Pequeño positivo
5	Si	Erroricar es Grande positivo	Y	Integralicar es Grande positivo	Entonces	Controlicar es Cero
6	Si	Erroricar es Pequeño negativo	Y	Integralicar es Grande negativo	Entonces	Controlicar es Grande positivo
7	Si	Erroricar es Pequeño negativo	Y	Integralicar es Pequeño negativo	Entonces	Controlicar es Grande positivo
8	Si	Erroricar es Pequeño negativo	Y	Integralicar es Cero	Entonces	Controlicar es Pequeño positivo
9	Si	Erroricar es Pequeño negativo	Y	Integralicar es Pequeño positivo	Entonces	Controlicar es Cero
10	Si	Erroricar es Pequeño negativo	Y	Integralicar es Grande positivo	Entonces	Controlicar es Pequeño negativo
11	Si	Erroricar es Cero	Y	Integralicar es Grande negativo	Entonces	Controlicar es Grande positivo
12	Si	Erroricar es Cero	Y	Integralicar es Pequeño negativo	Entonces	Controlicar es Pequeño positivo
13	Si	Erroricar es Cero	Y	Integralicar es Cero	Entonces	Controlicar es Cero
14	Si	Erroricar es Cero	Y	Integralicar es Pequeño positivo	Entonces	Controlicar es Pequeño negativo
15	Si	Erroricar es Cero	Y	Integralicar es Grande positivo	Entonces	Controlicar es Grande negativo
16	Si	Erroricar es Pequeño positivo	Y	Integralicar es Grande negativo	Entonces	Controlicar es Pequeño positivo
17	Si	Erroricar es Pequeño positivo	Y	Integralicar es Pequeño negativo	Entonces	Controlicar es Cero
18	Si	Erroricar es Pequeño positivo	Y	Integralicar es Cero	Entonces	Controlicar es Pequeño negativo
19	Si	Erroricar es Pequeño positivo	Y	Integralicar es Pequeño positivo	Entonces	Controlicar es Grande negativo
20	Si	Erroricar es Pequeño positivo	Y	Integralicar es Grande positivo	Entonces	Controlicar es Grande negativo
21	Si	Erroricar es Grande positivo	Y	Integralicar es Grande negativo	Entonces	Controlicar es Cero
22	Si	Erroricar es Grande positivo	Y	Integralicar es Pequeño negativo	Entonces	Controlicar es Pequeño negativo
23	Si	Erroricar es Grande positivo	Y	Integralicar es Cero	Entonces	Controlicar es Grande negativo
24	Si	Erroricar es Grande positivo	Y	Integralicar es Pequeño positivo	Entonces	Controlicar es Grande negativo
25	Si	Erroricar es Grande positivo	Y	Integralicar es Grande positivo	Entonces	Controlicar es Grande negativo

Tabla. 4. 14. Base de reglas del controlador difuso PI carro

Donde:

- *Erroricar*: diferencia entre la posición deseada del carro y la posición real del carro.
- *Integralicar*: integral del error (*Erroricar*).
- *Controlicar*: señal de control.

4.3.5 Implementación del controlador difuso PI en SIMULINK

Los bloques inmersos en la implementación del controlador difuso PI, en el modelo de simulación del sistema carro – péndulo son los mismos que se explicaron en la sección 4.1.5. y 4.2.1 con la diferencia que ahora no se encuentra el controlador *Swing – up* como se muestra en la Figura. 4. 17.

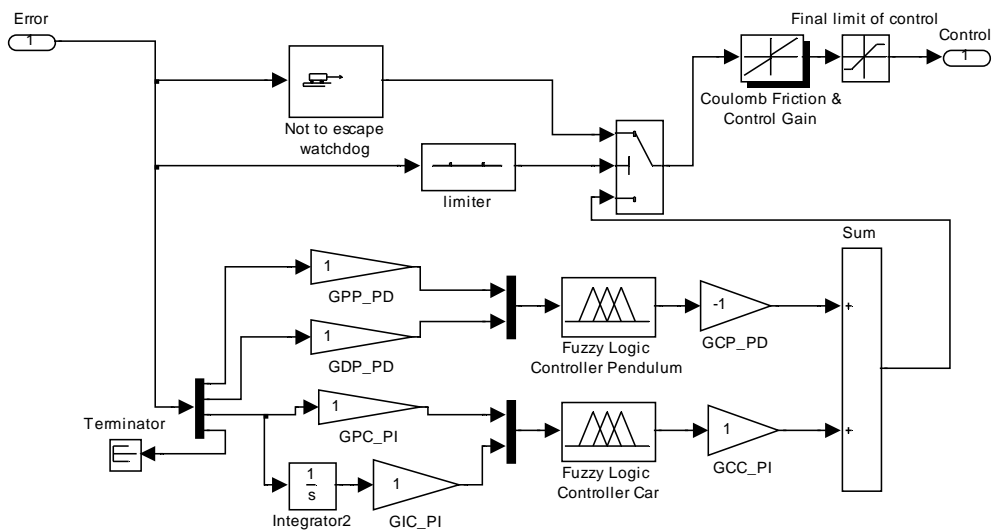


Figura. 4. 17. Implementación controlador difuso PI, Modelo simulación

4.3.6 Simulación del controlador difuso PI en SIMULINK

El sistema carro – péndulo como proceso se lo puede catalogar como rápido, puesto que para lograr la estabilidad únicamente del péndulo las acciones del controlador deben ser rápidas lo que es prioridad en este control, aunque cabe la posibilidad de implementar un controlador lento pero que también corrija el error en estado estacionario respecto de la

posición del carro sin afectar el control sobre el péndulo. En tal motivo se aplica un controlador PI únicamente al carro.

La variable integral del error de posición del carro es proporcional al error acumulado, lo que implica que es un modo de control lento.

Los resultados de simulación se observa en la Figura. 4. 18.

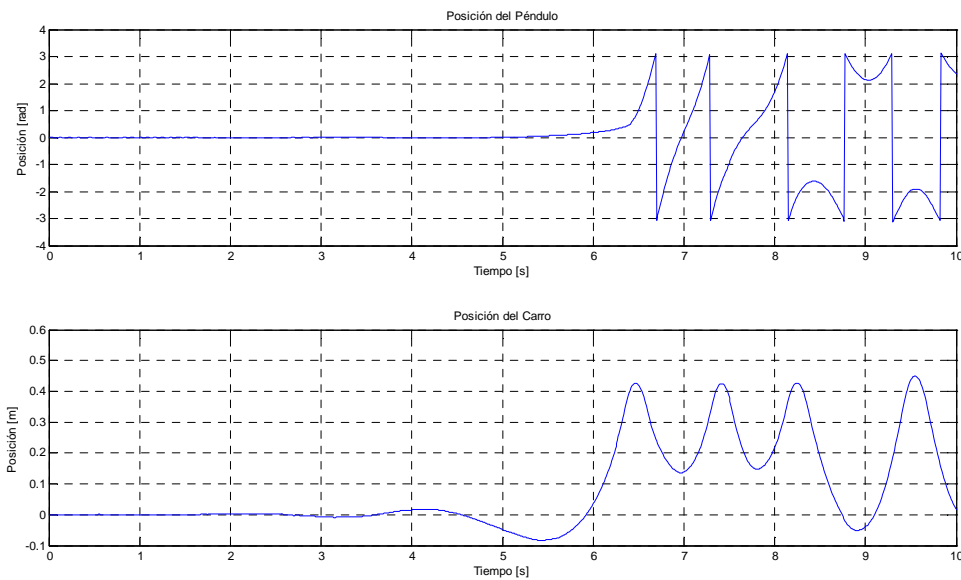


Figura. 4. 18. Difuso PI, Modelo Simulación, Posición péndulo y Posición carro

Como se puede observar en la Figura. 4. 18. el objetivo de control se logra por un corto intervalo de tiempo. La señal de posición del péndulo juntamente con la señal de posición del carro presentan error en estado estacionario conforme avanza el tiempo de simulación lo que provoca la caída del péndulo.

Esta simulación está realizada en un tiempo de 10 [s] con los siguientes valores en las ganancias:

Controlador difuso PD péndulo:

- Ganancia proporcional (GPP_{PD})= 1
- Ganancia derivativa (GDP_{PD})= 1
- Ganancia de control (GCP_{PD}) = -1

Controlador difuso PI carro:

- Ganancia proporcional (GPC_{PI}) = 1
- Ganancia derivativa (GDC_{PI}) = 1
- Ganancia de control (GCC_{PI}) = 1

4.3.7 Implementación del controlador difuso PI en el modelo experimental

La implementación del controlador difuso PI, en el modelo de experimentación del sistema carro – péndulo, es semejante al realizado para el modelo de simulación (sección 4.3.5.), con la diferencia que ahora contiene el controlador del dispositivo (*device driver*).

4.3.8 Experimentación del controlador difuso PI

Antes de realizar cualquier cambio en las ganancias de los controladores difusos PD y PI se analiza el cumplimiento del objetivo de control con los valores de ganancias impuestas en simulación. En consecuencia estos valores demuestran un comportamiento

similar a la simulación ya que el objetivo de control se cumple por cierto intervalo de tiempo, es decir el péndulo en este intervalo de tiempo se encuentra en posición vertical hacia arriba pero con el carro deslizándose hacia un extremo del riel, lo que posteriormente ocasiona la caída del péndulo.

De igual manera se varían únicamente las ganancias en el controlador difuso PD péndulo más no en el controlador difuso PI carro, ya que tiene **mayor prioridad** para la ubicación del péndulo en la posición vertical estable.

El criterio de desempeño *integral del error cuadrático (CIEC)* aplicado al controlador difuso PI se muestra en la Tabla. 4. 15.

Ganancia proporcional (GPP PD)	Ganancia derivativa (GDP PD)	Ganancia de control (GCP PD)	CIEC posición angular del péndulo	CIEC posición del carro	CIEC TOTAL
1	1	-1	32.45	1.742	34.19
1	0.5	-3	31.62	1.272	32.89
1	0.8	-3	34.19	1.808	36
1.2	1	-3	30.57	1.503	32.13
1.3	1	-3	11.87	1.565	13.43
1	0.7	-3.5	17.47	1.707	19.18
1	0.6	-3.5	13.44	1.447	14.89
1.7	0.5	-3	11.92	1.3	13.25
1.6	0.5	-3	45.9	1.772	47.67
1.8	1.0	-2.5	29.03	1.835	30.87
1.7	1.0	-2.5	16.74	1.342	18.08

Tabla. 4. 15. CIEC del controlador difuso PI

Como se observa en la Tabla. 4. 15. el CIEC más óptimo corresponde al valor 13.25 con sus respectivas ganancias, mismo que sirve como referencia al momento de demostrar el control que existe sobre la planta.

Todos los valores que presenta la Tabla. 4. 15. cumplen con el objetivo de control de manera parcial ya que como se había mencionado llega a realizarlo por un cierto intervalo de tiempo demostrando que el tipo de controlador que debe existir sobre el carro también debe ser rápido más no lento que es el caso del controlador difuso PI. El tiempo de experimentación para estas pruebas es de 10 [s].

CAPÍTULO V

CONCLUSIONES Y RECOMENDACIONES

5.1 Conclusiones

1. La identificación y cuantificación de las reglas difusas que tienen correspondencia con la dinámica de la planta física del laboratorio ha sido esencial en la generación de un algoritmo de control apropiado.
2. En el diseño del control difuso PD o PI, la elección apropiada de las variables y valores lingüísticos así como las funciones de pertenencia han sido pilares fundamentales en el desempeño del mismo.
3. La conexión en tiempo real entre el computador y el sistema carro – péndulo es primordial puesto que en las acciones de control que demandan sus componentes no debe haber retardos de tiempo.
4. El modelo matemático de simulación para el sistema carro – péndulo ha sido una herramienta de gran ayuda, ya que conjuntamente con la guía de funcionamiento ha permitido familiarizarse y evitar inconvenientes en la operación del sistema. Además de poder crear en base al modelo matemático el sistema de control.

5. Los valores identificados para el modelo matemático han sido obtenidos de manera experimental, los cuales por razones físicas del sistema varían al momento de encontrarlos. En consecuencia los valores ubicados en el modelo matemático guardan cierta similitud con la planta física, pero no se garantiza una equivalencia exacta.

6. El modelo matemático que describe al sistema carro – péndulo trata de emular en su totalidad el comportamiento del mismo, es así que en este proyecto se comprobó al simular los controladores diseñados, la existencia de un cierto grado de error en el modelo matemático.

7. Se ha logrado con éxito el control del sistema carro – péndulo en modo péndulo invertido mediante dos controladores difusos proporcional derivativo con balanceo para el péndulo y el carro e implementarlos en Matlab.

8. Se consiguió realizar el control del sistema carro – péndulo en modo péndulo invertido mediante dos controladores difusos proporcional derivativo para el péndulo y el carro, cuando el péndulo inicia desde la posición vertical estable.

9. Se logró realizar el control del sistema carro – péndulo en modo péndulo invertido de manera parcial mediante un controlador difuso proporcional derivativo para el péndulo y un controlador difuso proporcional integral para el carro, cuando el péndulo inicia desde la posición vertical estable.

10. Se determinó que al implementar un controlador difuso para el péndulo y otro para el carro es conveniente dar prioridad al realizado sobre el péndulo puesto que este tiene mayor contribución en el logro del modo péndulo invertido.

11. Se analizó el funcionamiento en detalle del sistema carro – péndulo ya que no se disponía suficiente información del mismo.

12. La interfaz realizada en Matlab permite al usuario, mantener una interacción amigable y a la vez funcional de todo el comportamiento del sistema tanto en simulación como experimentación.

5.2 Recomendaciones

1. Es necesario realizar un estudio detallado de las partes y el funcionamiento del hardware así como del software del sistema carro – péndulo antes de realizar cualquier prueba de control sobre el mismo. Puesto que en el manual de usuario no existe toda la información.
2. Seguir detenidamente los pasos para el proceso de identificación ya que de esto depende los valores obtenidos y a su vez un acertado modelo de simulación.
3. Al momento de realizar el restablecimiento de los encoders se debe recordar que existe una única posición para este proceso, que es en el extremo izquierdo del riel y el péndulo en posición vertical hacia abajo.
4. Verificar siempre el estado del indicador rojo (circuito de potencia), ya que en caso de presiona el interruptor de seguridad se debe primero desenclavar y después nuevamente pulsar para el encendido del circuito de potencia.
5. Cada vez que se realice cambios en el modelo de experimentación, se debe construir el modelo (RTW) antes de usarlo.
6. Revisar el ajuste correcto del elemento de sujeción para la banda de transmisión y el carro.

7. La construcción del modelo experimental en Matlab, es de fácil ejecución por parte del usuario, lo que conlleva a un mal funcionamiento si el control no es el correcto, pudiendo provocar daños en la estructura mecánica. Por lo mismo se recomienda que se ejecute cualquier control primero en el modelo de simulación.

8. Mantener un área despejada alrededor del sistema carro – péndulo, puesto que llegado al caso el péndulo puede girar en los extremos de riel afectando a elementos que estén en su rango de acción.

REFERENCIAS BIBLIOGRÁFICAS

1. **Inteco**, PENDULUM – CART SYSTEM User's Manual. [En línea] 1997.
<http://www.inteco.com.pl>.

2. **Acuña, Byron e Ibarra, Oswaldo**. *Diseño e implementación de un sistema controlador de temperatura pid para la unidad air flow temperature control system mediante la utilización de la herramienta rtw de matlab*. [En línea] 2010.
<http://repositorio.espe.edu.ec/handle/21000/237>.

3. **Cilento, Augusto**. *Sistemas de control de motores en Tiempo Real mediante Matlab*. [En línea] 2007. <http://cde05.etse.urv.es/pub/pdf/1088pub.pdf>.

4. **Universidad Nacional de Quilmes, IACI**. *Real Time Workshop*, Laboratorio de Automatización. [En línea].
<http://iaci.unq.edu.ar/materias/laboratorio2/transparencias%5CReal%20Time%20Workshop.pdf>.

5. **Pérez, Rosanna.** *Procesado y optimización de espectros gamma mediante técnicas de lógica difusa: aplicación a la identificación de materiales pictóricos.* [En línea] 2005. http://biblioteca.universia.net/html_bura/ficha/params/title/procesado-optimizacion-espectros-raman-tecnicas-logica-difusa-aplicacion-identificacion-materiales/id/905539.html.

6. **Maldonado, Casas, Canas.** *Fuzzy sets application to assess the parameters of the bridge seismic vulnerability.* [En línea] 2002. <http://upcommons.upc.edu/revistes/bitstream/2099/4604/1/RR182B.pdf>

7. **Cárdenas, Moncayo.** *Técnicas de control difuso aplicadas a la navegación de un robot móvil.* [En línea] 2006. <http://bibdigital.epn.edu.ec/handle/15000/58>.

8. **Gómez, Galindo.** *Conjuntos y sistemas difusos.* Departamento de Lenguajes y Ciencias de la Computación, Universidad de Málaga. [En línea]. <http://www.lcc.uma.es/~ppgg/FSS/FSSindex.pdf>

9. **Medina, Pablo.** *Estudio del desempeño del motor de inducción usando un controlador difuso.* [En línea] 2007. <http://bibdigital.epn.edu.ec/handle/15000/4219>.

10. **Pineda, Vivas.** *Control de un modelo aerodinámico aplicando sistemas difusos.* [En línea] 2008. <http://bibdigital.epn.edu.ec/handle/15000/1082>

11. **Duarte, Oscar.** *Sistemas de lógica difusa. Fundamentos.* [En línea]. <http://www.profesaulosuna.com/data/files/ELECTRONICA/LOGICA%20DIFUSA/TextoAplicaciones.pdf>

12. **Passino, Kevin y Yurkovich, Stephen.** *Fuzzy control.* [En línea] 1998. <http://www2.ece.ohio-state.edu/~passino/books.html>

13. **Díaz, Juan.** *Inversores pwm,* Tecnología Electrónica, Universidad de Oviedo. [En línea] 1999. <http://pels.edv.uniovi.es/pels/pels/Pdf/Leccion%20Inversores%20Juan%20Diaz.pdf>

14. **Ortíz, Floriberto.** *Modelado y control PD-difuso en tiempo real para el sistema barra-esfera.* Departamento de control automático, Centro de investigación y de estudios avanzados del instituto politécnico nacional. [En línea] 2004. <http://www.ctrl.cinvestav.mx/~yuw/pdf/MaTesFOR.pdf>

15. **Cruz, Agustín.** *Multiplicador electrónico para encoder incremental.* [En línea] 2008. http://polibits.gelbukh.com/38_07.pdf

16. **Turnau, Korytowski, Szymkat.** *Time optimal control for the pendulum-cart system in real time.* Departamento de automatic, AGH. [En línea] 1999.
http://aq.ia.agh.edu.pl/mse/CCA99_211.pdf
17. **Ogata, Katsuhiko.** *Ingeniería de Control Moderna*, Cuarta Edición, Prentice Hall. 2002.
18. **Universidad Politécnica de Madrid, eMathTeacher.** *Método de Mamdani de Inferencia Borrosa.* [En línea].
<http://www.dma.fi.upm.es/java/fuzzy/fuzzyinf/funpert.htm>.
19. **Rodríguez, Alejandra.** *Astrofísica Computacional: Aplicación de técnicas de inteligencia artificial en la clasificación y parametrización de espectros estelares.* [En línea] 2010.
http://ruc.udc.es/dspace/bitstream/2183/7322/6/RodriguezFernandez_AlejandraJunio2010_TD.pdf.

ÍNDICE DE FIGURAS

Figura. 1. 1. Algoritmo RTW	17
Figura. 1. 2. Conjunto Difuso.....	20
Figura. 1. 3. Representación de la estatura de una persona.....	21
Figura. 1. 4 Función de pertenencia	22
Figura. 1. 5. Funciones de pertenencia: (a) trapezoidal, (b) triangular, (c) gaussiana, (d) sigma	24
Figura. 1. 6. Operación unión de conjuntos difusos.....	27
Figura. 1. 7. Operación intersección de conjuntos difusos.....	27
Figura. 1. 8. Operación complemento de conjuntos difusos	28
Figura. 1. 9. Arquitectura del Controlador Difuso	32
Figura. 2. 1. Componentes del sistema carro - péndulo	37
Figura. 2. 2. Sistema de control del péndulo	38
Figura. 2. 3. Configuración de Hardware del sistema carro - péndulo.....	39
Figura. 2. 4. Fuente de alimentación	42
Figura. 2. 5. Circuito de potencia	43
Figura. 2. 6. Esquema de la etapa de potencia del sistema carro – péndulo.....	44
Figura. 2. 7. Esquema interior de la tarjeta RT-DAC4/PCI-D	45
Figura. 2. 8. Partes mecánicas y sensores en la configuración carro – péndulo.....	47
Figura. 2. 9. Señales eléctricas producidas por el disco codificado del encoder incremental	47
Figura. 2. 10. Determinación del sentido de giro	48
Figura. 2. 11. Motor DC.....	49
Figura. 2. 12. Software carro-péndulo.....	50
Figura. 2. 13. Esquema para el modelo matemático sistema carro-péndulo	56
Figura. 2. 14. Parámetros en el cálculo del momento de inercia del péndulo	59
Figura. 2. 15. Bloque dinámico del sistema carro-péndulo.....	61
Figura. 2. 16. Interior del bloque dinámico del sistema carro-péndulo.....	62
Figura. 2. 17. Interior del bloque fricción	67
Figura. 2. 18. Bloques de cuantificación.....	69
Figura. 3. 1. Pasos en la identificación del sistema carro-péndulo.....	73
Figura. 3. 2. Ventana de comprobación de señales de los encoders.....	74
Figura. 3. 3. Ajuste del ciclo de duración PWM	76
Figura. 3. 4. Montaje de la balanza	77
Figura. 3. 5. Distribución de valores PWM vs. Fuerza	78
Figura. 3. 6. Valores PWM vs. Fuerza	79
Figura. 3. 7. Ventanas de diálogo.....	80

Figura. 3. 8. Información sobre el movimiento del carro.....	80
Figura. 3. 9. Ubicación del carro.....	80
Figura. 3. 10. Información sobre el movimiento del carro.....	81
Figura. 3. 11. Detección de la mínima fuerza	81
Figura. 3. 12. Fijación del carro	82
Figura. 3. 13. Ventana de identificación de fricción del péndulo.....	83
Figura. 3. 14. Ventana de adquisición de datos para la fricción del péndulo.....	84
Figura. 3. 15. Normalización posición angular del péndulo, Modelo simulación.....	84
Figura. 3. 16. Ventana de comprobación de posición angular del péndulo.....	85
Figura. 3. 17. Trazado de datos del péndulo	86
Figura. 3. 18. Oscilaciones amortiguadas del péndulo.....	88
Figura. 3. 19. Modelo Simulación - Controlador basado en reglas y Controlador PID	90
Figura. 3. 20. Modelo Simulación - Controlador basado en reglas.....	93
Figura. 3. 21. Zonas de actividad para los algoritmos de control.....	94
Figura. 3. 22. Diagrama bloque PID	97
Figura. 4. 1. Diagrama de bloques del sistema de control.....	99
Figura. 4. 2. Funciones de pertenencia, Controlador difuso PD péndulo.....	104
Figura. 4. 3. Superficie de control del controlador difuso PD péndulo.....	105
Figura. 4. 4. Funciones de pertenencia, Controlador difuso PD carro	106
Figura. 4. 5. Superficie de control del controlador difuso PD carro	107
Figura. 4. 6. Posiciones del péndulo.....	110
Figura. 4. 7. Implementación controlador difuso PD con balanceo, Modelo simulación	113
Figura. 4. 8. Normalización posición angular del péndulo, Modelo simulación.....	114
Figura. 4. 9. Implementación controlador difuso PD con balanceo, Modelo simulación	115
Figura. 4. 10. Difuso PD con balanceo, Modelo Simulación, Posición péndulo y Posición carro	116
Figura. 4. 11. Implementación controlador difuso PD con balanceo, Modelo experimentación ...	120
Figura. 4. 12. Normalización posición angular del péndulo, Modelo experimentación.....	121
Figura. 4. 13. Implementación controlador difuso PD, Modelo simulación	127
Figura. 4. 14. Difuso PD, Modelo Simulación, Posición péndulo y Posición carro	128
Figura. 4. 15. Funciones de pertenencia, Controlador difuso PI carro.....	133
Figura. 4. 16. Superficie de control del controlador difuso PI carro	134
Figura. 4. 17. Implementación controlador difuso PI, Modelo simulación.....	136
Figura. 4. 18. Difuso PI, Modelo Simulación, Posición péndulo y Posición carro.....	137

ÍNDICE DE TABLAS

Tabla. 1. 1. Similitud entre operadores	29
Tabla. 2. 1. Parámetros del modelo matemático tomados por el fabricante.....	53
Tabla. 2. 2. Parámetros del modelo matemático	54
Tabla. 2. 3. Código para presentación de parámetros identificados.....	55
Tabla. 3. 1. Fuerza del motor DC	77
Tabla. 3. 2. Valores Identificados para el modelo matemático de simulación.....	89
Tabla. 3. 3. Algoritmo del controlador basado en reglas	91
Tabla. 4. 1. Variables sistema de control	99
Tabla. 4. 2. Variables del controlador difuso PD péndulo	101
Tabla. 4. 3. Variables del controlador difuso PD carro.....	102
Tabla. 4. 4. Valores lingüísticos de los controladores difusos PD	102
Tabla. 4. 5. Rango de variables del controlador difuso PD péndulo	103
Tabla. 4. 6. Rango de variables del controlador difuso PD carro.....	103
Tabla. 4. 7. Base de reglas del controlador difuso PD péndulo.....	108
Tabla. 4. 8. Base de reglas, controlador difuso PD péndulo	109
Tabla. 4. 9. CIEC del controlador difuso PD con balanceo	124
Tabla. 4. 10. CIEC del controlador difuso PD	130
Tabla. 4. 11. Variables del controlador difuso PI carro	132
Tabla. 4. 12. Valores lingüísticos del controlador difuso PI carro.....	132
Tabla. 4. 13. Rango de variables del controlador difuso PI carro	132
Tabla. 4. 14. Base de reglas del controlador difuso PI carro.....	135
Tabla. 4. 15. CIEC del controlador difuso PI.....	139

GLOSARIO

- **A/D (Analog/Digital)**, Conversor Analógico/Digital: codificación.
- **Cuantificación**, Número de símbolos que se utiliza para guardar una medida de una señal.
- **D/A (Digital/Analog)**, Conversor Digital/Anal: decodificación.
- **DC (Direct Current)**, Corriente Directa.
- **Difuso**, Técnica de razonamiento que aplica valores múltiples de verdad a las restricciones durante la resolución de problemas.
- **Driver (Controlador)**, Programa conocido como controlador, que permite la gestión de los dispositivos conectados al computador.
- **Encoder (Codificador)**, Dispositivo que convierte la información de un formato o el código a otro.
- **FIS (fuzzy inference system)**, Sistema de inferencia difuso.
- **Kernel**, Interface con el sistema operativo Windows. Asigna altas prioridades de ejecución a un código ejecutable.
- **NOR**, Puerta lógica digital.
- **RPM**, Revoluciones por minuto.
- **Simulink**, Entorno para simulación multidominio y diseño basado en modelos para sistemas dinámicos y embebidos.
- **VAC (Voltage altern current)**, Voltaje de corriente alterna.
- **VDC (Voltage direct current)**, Voltaje de corriente directa.

ACTA DE ENTREGA

El proyecto fue entregado al Departamento de Eléctrica y Electrónica y reposa en la Escuela Politécnica del Ejército desde:

Sangolquí, a _____

Ing. Víctor Proaño

COORDINADOR DE CARRERA

Hugo Marcelo Pérez Gaibor