

UNIVERSIDAD TÉCNICA FEDERICO SANTA MARÍA
SEDE VIÑA DEL MAR - JOSÉ MIGUEL CARRERA

**DISEÑO E IMPLEMENTACIÓN DE UN CONTROLADOR PID CON PROGRAMACIÓN DE
GANANCIA BASADO EN UN SISTEMA DIFUSO**

Trabajo de Titulación para optar al
Título de Ingeniería de EJECUCIÓN EN
CONTROL E INSTRUMENTACIÓN
INDUSTRIAL

Alumno:
José Antonio Aspe Rodríguez

Profesor Guía:
Ing. Loreto Marín Carcey

2020

RESUMEN

KEYWORDS: GAIN SCHEDULING, LOGICA DIFUSA, PID, CONTROLADOR

Para la solución de problemas no lineales existen diversas técnicas como lo son el control por modelos, la lógica difusa y los controladores de estado. Aunque estas técnicas tienen buenos resultados, el análisis y diseño para cada sistema individual es largo y tedioso, por lo que se realiza una implementación más general para este tipo de problemas, buscando así minimizar los tiempos de implementación del controlador.

El tipo de algoritmo elegido para realizar lo anterior es el controlador por ganancia programada, el cual, por la forma que tiene de trabajar, se presta para una simplificación en cuanto a la parametrización e implementación. Este controlador está formado por dos partes, una es un controlador PID, que puede ser presentado en cualquiera de las formas que tiene su algoritmo, pero que para el caso de este trabajo se utilizara la forma no interactiva, esto debido a que esta representación del controlador PID está presente en la mayoría de los controladores industriales. La otra parte que forma este controlador es un modelo matemático que realiza el cambio de los parámetros, que para este caso se realizará mediante un controlador difuso que usa una generalización de las reglas de sintonía de los controladores PID.

Los resultados de las pruebas realizadas con el modelo de los tanques no interactivos muestran que el controlador diseñado cumple con las expectativas de diseño y su funcionamiento es mejor en el rango de operación con respecto al controlador PID no interactivo y el controlador de ganancia programada por tabla de estado.

ÍNDICE

RESUMEN

INTRODUCCIÓN

CAPITULO 1 – MARCO TEORICO Y DEFINICIÓN DE OBJETIVOS

1	DEFINICIÓN DE OBJETIVOS y MARCO TEÓRICO	3
1.1	MARCO TEÓRICO	3
1.1.1	Sistema de control	3
1.1.2	Controlador PID	3
1.1.3	Control adaptativo	5
1.1.4	Lógica difusa	5
1.1.5	Estructura de un controlador difuso	11
1.2	HIPÓTESIS	12
1.3	OBJETIVOS GENERAL Y ESPECÍFICOS	12
1.4	ALCANCES Y LIMITACIONES	13
2	CONTROLADOR PID CON PROGRAMACION DE GANANCIA	15
2.1	DISEÑO DEL CONTROLADOR PID	15
2.1.1	Modificaciones al algoritmo de PID	15
2.1.2	Algoritmo PID a implementar	17
2.2	DISEÑO CONTROLADOR DIFUSO	17
2.2.1	Creación de conjuntos de entrada y salida	18
2.2.2	Creación de reglas difusas	20
2.3	IMPLEMENTACIÓN CONTROLADOR PID CON PROGRAMACIÓN DE GANANCIA	22
2.4	MODELO MATEMÁTICO DE TANQUES CONECTADOS DE FORMA NO INTERACTIVA	22
2.4.1	Definiciones de modelamiento	23
2.4.2	Construcción del modelo y ajuste de parámetros	25
2.5	OBTENCION DE PARAMETROS DE CONTROL PID Y FUZZY	26
2.5.1	Linealización por Series De Taylor	27
2.5.2	Obtención de los controladores PID	29
2.5.3	Obtención parámetros controlador difuso	31
3	RESULTADOS DE LAS PRUEBAS DEL CONTROLADOR PROPUESTO	33

3.1	PREPARACIÓN PARA LAS PRUEBAS DE RENDIMIENTO DEL CONTROLADOR	33
3.2	RESULTADO DE RENDIMIENTO DEL CONTROLADOR	34
3.2.1	Pruebas de comportamiento a distintos puntos de operación	34
3.2.2	Pruebas Ante Perturbaciones	37
3.3	CONCLUSIONES A LAS PRUEBAS PARA EL CONTROL PID	38
3.4	SIMPLIFICACIÓN DEL CONTROLADOR PROPUESTO	38
3.5	COMPARATIVA CONTROLADOR PROPUESTO CON RESPECTO A LA TÉCNICA CLÁSICA	39
3.6	CONCLUSIONES A LAS PRUEBAS CON CONTROLADORES DE TABLA DE ESTADO	41
3.7	TRABAJOS FUTUROS - IMPLEMENTACION EN ENTORNOS INDUSTRIALES	42

ÍNDICE DE FIGURAS

Figura 1-1 Control en Lazo Cerrado	3
Figura 1-2 Miembro Triangular	6
Figura 1-3 Miembro Trapezoidal	7
Figura 1-4 Miembro Singleton.....	8
Figura 1-5 Ejemplo de Conjunto Difuso	8
Figura 1-6 Estructura de un controlador difuso	12
Figura 2-1 Implementación filtro derivativo	16
Figura 2-2 Implementación Controlador PID	17
Figura 2-3 Normalización de la Variable Manipulada y Razón de Cambio	18
Figura 2-4 Respuesta Escalón	19
Figura 2-5 Grafico Pseudo Color Blue - Reglas difusas : (a) Kp (b) Ki, (c) Kd.....	21
Figura 2-6 implementación final control PID con ajuste de ganancia	22
Figura 2-7 Tanques conectados de forma no interactiva	23
Figura 2-8 Implementación Modelo Tanques No Interactivos	25
Figura 2-9 Perfil de Niveles en Estado Estacionario	26
Figura 2-10 Funciones de transferencia en Estado Estacionario	29
Figura 2-11 Interfaz Aplicación "PID Tuner"	29
Figura 2-12 Diagrama de Bode cada función de transferencia.....	30
Figura 2-13 Comportamiento parámetros PID en estados estables	31
Figura 3-1 Simulación para Pruebas de Rendimiento.....	33
Figura 3-2 Simulación para Pruebas de Rendimiento.....	34
Figura 3-3 Comportamiento Controladores a Distintos Puntos de Operación.....	35
Figura 3-4 Comportamiento en caso de perturbaciones.....	38
Figura 3-5 Implementación final control PID con ajuste de ganancia	39
Figura 3-6 Implementación Controlador Por Tablas De Estado.....	40
Figura 3-7 Comparación entre tablas de estado y ajuste difuso.....	40

ÍNDICE DE ECUACIONES

Ecuación 1-1 Función matemática de un Controlador PID	4
Ecuación 1-2 Forma Triangular.....	6
Ecuación 1-3 Forma Trapezoidal	7
Ecuación 1-4 Forma Singleton	7
Ecuación 1-5 Definición matemática operación complemento	9
Ecuación 1-6 Definición matemática operación unión (OR).....	9
Ecuación 1-7 Definición matemática operación intersección.....	9
Ecuación 1-8 Or de Lukasiewicz.....	9
Ecuación 1-9 AND del Producto.....	9
Ecuación 1-10 Reglas Difusa de Mamdani	10
Ecuación 1-11 Reglas Difusas de Takagi Sugeno	10
Ecuación 1-12 Cálculo de Centroide	11
Ecuación 2-1 Controlador PID Estándar en Laplace	15
Ecuación 2-2 Implementación de Filtro Pasa bajos en la derivada	16
Ecuación 2-3 Definición de capacitancia del tanque	23
Ecuación 2-4 Ecuaciones de variación de volumen	24
Ecuación 2-5 Definición de resistencia	24
Ecuación 2-6 Ecuaciones de caudal según restricción	24
Ecuación 2-7 Ecuaciones de caudal según restricción	25
Ecuación 2-8 H1 tanque 1 (a) H2 tanque 2 (b) ambos en estado estable	26
Ecuación 2-9 Serie de Taylor - Jacobiana del Sistema	27
Ecuación 2-10 Serie Taylor tanque uno	27
Ecuación 2-11 Linealización tanque uno.....	28
Ecuación 2-12 Linealización tanque dos	28
Ecuación 2-13 Función de Transferencia del Sistema	28

ÍNDICE DE TABLAS

Tabla 2-1 Efectos de los parámetros PID	20
Tabla 2-2 Parámetros para Modelo Fenomenológico	25
Tabla 2-3 Valores en Estado Estable	27
Tabla 2-4 Tabla de Parámetros Lógica Difusa	31
Tabla 3-1 Análisis de Respuesta al Escalón	35
Tabla 3-2 Estadística Descriptiva de las Simulaciones para cada Parámetro	36
Tabla 3-3 Análisis de Respuesta al Escalón	41
Tabla 3-4 Estadística Básica de Los Controladores.....	41

SIGLAS Y SIMBOLOGÍA

A. SIGLAS

PID : Proporcionar - integrativo – derivativo

GS : Gain scheduling

PLC : Controlador Lógico Programable

ST : Texto Estructurado

B. SIMBOLOGÍA

$u(t)$: Señal de entrada al sistema en el dominio del tiempo

$e(t)$: Señal de error del sistema en el dominio del tiempo

INTRODUCCION

En el control de procesos industriales existen bastantes problemas de control no lineal debido a la naturaleza de estos, es por lo que el diseño de controladores es bastante estudiado y aunque existen varios métodos de control para los distintos tipos de problemas que se presentan en la industria el diseño y la implantación de estos sigue siendo bastante complicado.

Es este trabajo se diseñará e implementará un controlador PID con programación de ganancia basados en lógica difusa, el cual ayudará a minimizar los tiempos de implementación y eliminará la necesidad de diseño. Para cumplir con el objetivo de diseño se realizará el estudio de los sistemas no lineales, las bases que sentarán la disposición del controlador PID y los fundamentos que rigen la lógica de control difusa, todo esto se verá en el capítulo 1.

En el capítulo 2 se diseñará el controlador PID que funcionará como base para el control, para esto se realizará el estudio de los problemas de la implementación de los distintos algoritmos de los controladores PID y se seleccionará el que más se ajuste al problema que se desea resolver, junto con esto se revisarán las variables que se utilizarán para controlar el cambio de parámetros que será hecho con un controlador difuso cuyas reglas y miembros se realizarán según la teoría y la práctica adquirida en la implementación de estos controladores.

Para realizar la comparativa de comportamiento y la validación del diseño, se implementará uno de los procesos más vistos en la industria, como lo es el control de niveles conectados de forma no interactiva, estos resultados serán presentados en el capítulo 3 en donde también se podrán ver la implementación final del diseño y la comparativa con uno de los métodos más utilizados para este tipo de problemas no lineales como lo es el controlador de ganancia programada por tablas de estado.

CAPÍTULO 1: DEFINICIÓN DE OBJETIVOS Y MARCO TEÓRICO

1 DEFINICIÓN DE OBJETIVOS Y MARCO TEÓRICO

En este capítulo se presentará la base teórica de la investigación, la que será necesaria para la realización del diseño del controlador de ganancia programada, así como los objetivos e hipótesis planteados para la realización de este.

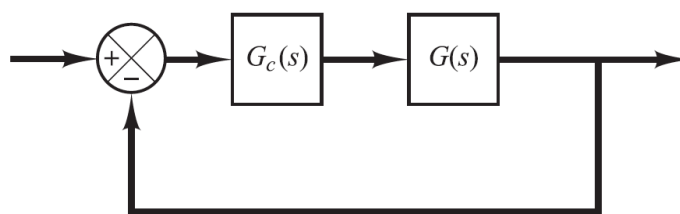
1.1 MARCO TEÓRICO

A continuación, se presentará una breve explicación de los conceptos a utilizar en este trabajo de título, para así lograr establecer las bases para un mejor entendimiento del capítulo 2.

1.1.1 Sistema de control

Un sistema de control es un conjunto de dispositivos físicos y/o de software encargado de mantener y regular el comportamiento del sistema.

Existen dos clases de sistemas de control, en lazo abierto y en lazo cerrado. Este trabajo se enfocara en el control en lazo cerrado, en estos sistemas el controles capaz de realizar una corrección mediante la retroalimentación de la salida de proceso , en la Figura 1-1 se puede apreciar el esquema de un control de lazo cerrado, siendo $G(s)$ la planta y $G_c(s)$ el controlador.



Fuente: Katsuhiko Ogata, Ingeniería de control moderna [2]

Figura 1-1 Control en Lazo Cerrado

1.1.2 Controlador PID

El control PID (Proporcional Integral Derivativo) es un mecanismo de control realimentado empleado ampliamente en la industria debido a su flexibilidad, fiabilidad y simpleza a la hora de su implementación.

En la Ecuación 1-1 se muestra el algoritmo del controlador PID no interactivo el cual es presentado por Ogata en su libro Ingeniería de control Moderna [1, p. 25]

$$u(t) = K_p * e(t) + \frac{K_p}{T_i} \int_0^t e(t) dt + K_p * T_d \frac{de(t)}{dt}$$

Fuente: Katsuhiko Ogata, Ingeniería de control moderna [2]

Ecuación 1-1 Función matemática de un Controlador PID

El funcionamiento de cada sección de la ecuación es descrito a continuación.

1.1.2.1 Acción Proporcional

La acción proporcional corresponde al producto de la señal de error y la constante proporcional. Normalmente dentro de los procesos industriales el uso de la ganancia proporcional cubre un espectro de la señal de control generando un offset durante el estado estacionario, por lo que es muy poco frecuente ver un control puramente proporcional. El offset en estado estacionario se debe a que la constante proporcional está limitada por la sobre oscilación del sistema, la cual en valores por sobre el 30% de la señal de referencia podría llegar a hacer que el sistema entre en resonancia, por lo que como recomendación general la sobre oscilación no debe sobrepasar el 25%.

1.1.2.2 Acción Integral

La acción integral tiene como propósito disminuir el error y eliminar perturbaciones que la acción proporcional no puede eliminar. La acción integral actúa sobre el error integrándolo por un periodo determinado y sumándolo a la acción proporcional, lo cual ayuda a eliminar el offset que tiene la acción proporcional.

1.1.2.3 Acción Derivativa

La acción derivativa utiliza la razón de cambio del error para realizar acciones en función de la velocidad con la que el sistema está reaccionando, normalmente este tipo de acción se utiliza cuando el actuador agrega un tiempo de retardo a la señal de control para mejorar el desempeño del controlador.

1.1.2.4 Limitaciones de un controlador PID

Aunque los controladores PID funcionan adecuadamente para la mayoría de los procesos industriales, estos no aplican en sistemas cuyas no linealidades sean importantes, los retardos de los actuadores o del mismo proceso sean muy grandes o el ruido introducido por los sensores empobrezca la acción derivativa. Para estos casos, dependiendo del proceso, existen modificaciones al modelo presentado por Ogata o alternativas como las que se presentarán en este trabajo. Además, la versión vista anteriormente (Ecuación 1-1) tiene bastantes problemas en algunos sistemas, debido al ruido introducido por los sensores y a los límites físicos de los actuadores, por lo que en el capítulo 2 se presentarán las modificaciones necesarias para robustecer el control.

1.1.3 Control Adaptativo

Un control adaptativo es aquel que, dependiendo del punto de operación en el cual se encuentre el sistema, se comporta de maneras distintas, adaptándose a circunstancias variantes de la dinámica de un sistema y a sus perturbaciones. Esto se logra ya sea modificando la salida del control mediante una corrección de la misma o modificando uno o más parámetros del controlador.

Existen diferentes tipos de controles adaptativos, en este trabajo se presentarán los que realizan un ajuste en los parámetros de sintonía de un controlador base y en particular el controlador PID con ganancia programada o en sus siglas en ingles GS (Gain Scheduling).

1.1.4 Lógica Difusa

Las notaciones y las operaciones de la lógica difusa están basadas en la lógica clásica y más específicamente en la lógica proposicional. El operando más simple de esta lógica es la “proposición” la cual actúa sobre una constante lógica, esta acción representa una operación lógica proposicional, combinando estas acciones lógicas se logra generar operaciones complejas.

[3, p. 29]

1.1.4.1 Funciones de membresía o pertenencia

Haciendo referencia a lo anterior y para generar dichas operaciones proposicionales es que los sistemas con lógica difusa cuentan con funciones de membresía, dichas funciones se encargan de cuantificar el grado de pertenencia de la variable de entrada al conjunto difuso para luego ser procesada por el sistema de inferencia.

Dentro de las funciones de membresía más utilizadas están las que entregan valores de pertenencia exponenciales, como son delta, pi, sigma, zeta y las que entregan valores lineales de pertenencia, como son las funciones triangulares y trapezoidales. Estas últimas se verán en detalle debido a que se utilizarán para la implementación del controlador PID con programación de ganancia.

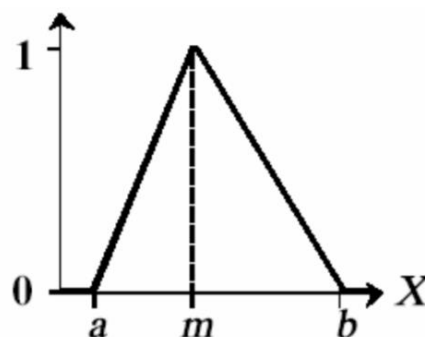
- a. **Forma Triangular:** Corresponde a la función de membresía más básica y sencilla de implementar. Como se observa en la Ecuación 1-2, su comportamiento corresponde básicamente al de la ecuación de la recta, por lo que se suele implementar tanto para conjuntos de entrada como de salida.

$$A(x) = \begin{cases} 0 & \text{si } x \leq a \\ (x-a)/(m-a) & \text{si } x \in (a, m) \\ (b-x)/(b-m) & \text{si } x \in (m, b) \\ 1 & \text{si } x \geq b \end{cases}$$

Fuente: W. Siler y J. J. Buckley [3]

Ecuación 1-2 Forma Triangular

En la Figura 1-2 se puede observar más claramente lo que representa la Ecuación 1-2



Fuente: Referencia [4]

Figura 1-2 Miembro Triangular

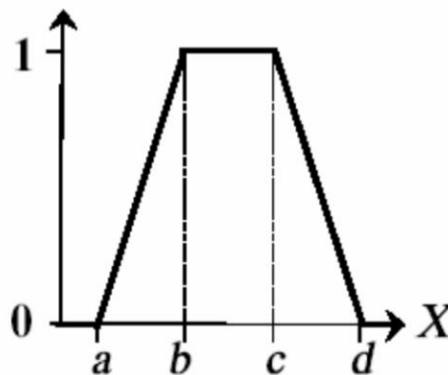
- b. **Forma Trapezoidal:** La forma trapezoidal, al igual que la triangular, se comporta como una ecuación de la recta, sin embargo, a diferencia de la triangular, la forma trapezoidal considera una banda en donde la pertenencia es completa mientras que la variable evaluada se encuentre dentro de su rango. Normalmente la implementación de este tipo de funciones de membresía se realiza en los conjuntos de entrada difusas, en muy pocas ocasiones se pueden ver estos miembros en la configuración de salida.

$$A(x) = \begin{cases} 0 & \text{si } (x \leq a) \vee (x \geq d) \\ (x-a)/(b-a) & \text{si } x \in (a, b] \\ 1 & \text{si } x \in (b, c) \\ (d-x)/(d-c) & \text{si } x \in (c, d) \end{cases}$$

Fuente: W. Siler y J. J. Buckley [3]

Ecuación 1-3 Forma Trapezoidal

En la Figura 1-3 se puede observar más claramente lo que representa la Ecuación 1-3



Fuente: J. D. C. Quisaguano [4]

Figura 1-3 Miembro Trapezoidal

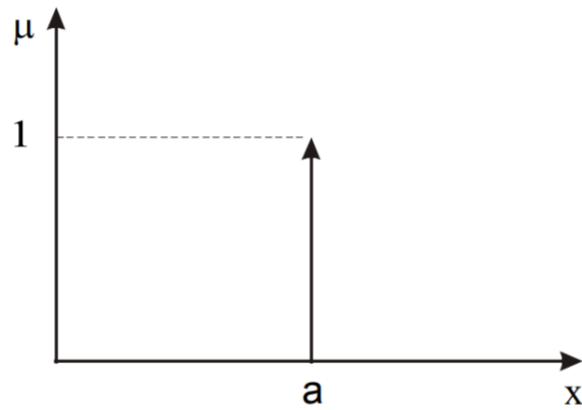
- c. **Forma Singleton:** Esta forma entrega un valor booleano y es más frecuentemente implementada en las salidas para generar valores discretos, que dependiendo del método de defusificación y la configuración del set difuso, entregarán una salida escalonada.

$$A(x) = \begin{cases} 1 & \text{si } x = a \\ 0 & \text{si } x \neq a \end{cases}$$

Fuente: W. Siler y J. J. Buckley [3]

Ecuación 1-4 Forma Singleton

En la Figura 1-4 se puede observar más claramente lo que representa la Ecuación 1-4.

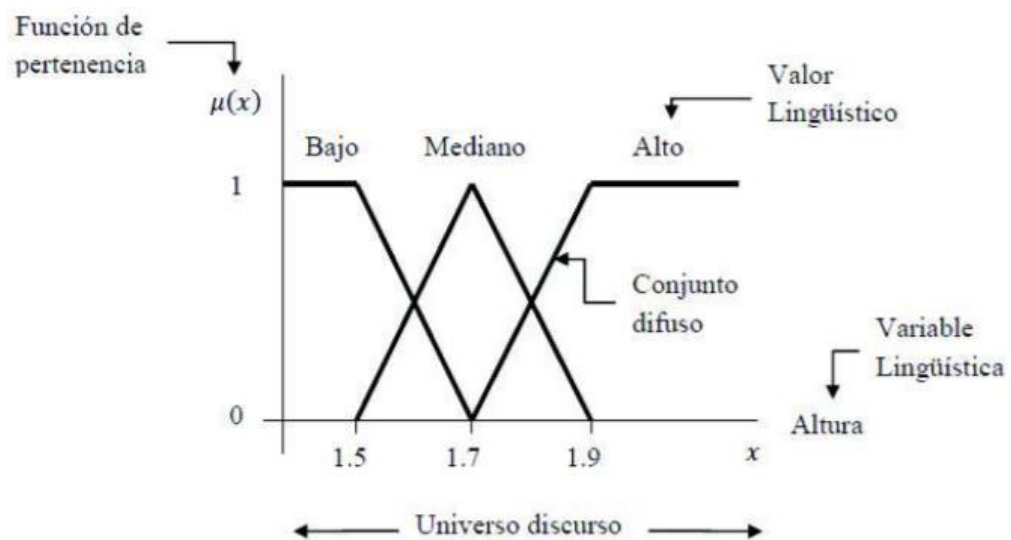


Fuente: J. D. C. Quisaguano [4]

Figura 1-4 Miembro Singleton

1.1.4.2 Sets o conjuntos difusos

Un set difuso, ya sea de entrada o de salida, es un conjunto de funciones de membresía que tiene como objetivo definir el conjunto de estados de una variable lingüística. En la Figura 1-5 se puede apreciar un ejemplo de set difuso cuya variable lingüística es la altura y sus valores lingüísticos son bajo, medio y alto.



Fuente: J. D. C. Quisaguano [4]

Figura 1-5 Ejemplo de Conjunto Difuso

1.1.4.3 Operadores difusos

Para cuantificar las funciones de pertenencia existen tres tipos de operadores difusos, los que son la unión (OR), la intersección (AND) y Complemento (NOT) y dentro de estos operandos existen distintas formas matemáticas de realizar la cuantificación.

Teniendo un set difuso con dos funciones de membresía llamadas A y B se puede definir matemáticamente dichas operaciones de la siguiente forma:

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

Fuente: W. Siler y J. J. Buckley [3]

Ecuación 1-5 Definición matemática operación complemento

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

Fuente: W. Siler y J. J. Buckley [3]

Ecuación 1-6 Definición matemática operación unión (OR)

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

Fuente: W. Siler y J. J. Buckley [3]

Ecuación 1-7 Definición matemática operación intersección

Las formas presentadas para cada operación corresponden a las propuestas por Zadeh y son las más utilizadas al momento de la implementación de software para el control industrial, otras definiciones que se pueden encontrar son la OR de Lukasiewicz Ecuación 1-8 y la AND del producto Ecuación 1-9

$$\mu_{A \cup B}(x) = \max[\mu_A(x) + \mu_B(x), 1]$$

Fuente: W. Siler y J. J. Buckley [3]

Ecuación 1-8 Or de Lukasiewicz

$$\mu_{A \cap B} = \mu_A(x) * \mu_B(x)$$

Fuente: W. Siler y J. J. Buckley [3]

Ecuación 1-9 AND del Producto

1.1.4.4 Reglas difusas

Las reglas difusas son proposiciones que permiten expresar el conocimiento disponible sobre una acción y sus consecuencias, para esto se realiza una combinación entre las funciones de pertenencia y las operaciones difusas que buscan determinar el valor de esta proposición.

Existen variados métodos de creación de reglas difusas, los cuales pueden ser variaciones o mejoras de los presentados en este trabajo.

- a. **Reglas difusas de Mamdani:** Las reglas de Mamdani son altamente implementadas en control industrial debido a su simplicidad y amplia aceptación, lo anterior debido a lo bien adaptado que está este modelo con respecto a la incorporación del conocimiento experto y la experiencia del diseñador. La forma de implementación de Mamdani es la siguiente:

$$\text{If } x_1 \text{ is } A \text{ AND } x_2 \text{ is } B \text{ THEN } u_1 \text{ is } C$$

Fuente: W. Siler y J. J. Buckley [3]

Ecuación 1-10 Reglas Difusa de Mamdani

Siendo x_1 y x_2 entradas del proceso, A y B miembros de un set difuso de entrada y C de un miembro de un set difuso de salida que entrega un valor a la salida u_1 . La expresión anterior si x_1 es A y x_2 es B entonces la salida u_1 toma el valor de C.

- b. **Reglas difusas de Takagi-Sugeno:** Las reglas de Takagi Sugeno son mayormente implementadas para el modelamiento de procesos debido el resultado difuso a diferencia de mamdani es el valor de una función no lineal con respecto a las entradas, aunque esto último dependiendo del tipo de proceso y su complejidad puede ser modificado para expresar la función de salida de manera lineal. La forma de implementación de Sugeno es la siguiente:

$$\text{If } x_1 \text{ is } A \text{ AND } x_2 \text{ is } B \text{ THEN } u_1 = f(x_1, x_2)$$

Fuente: W. Siler y J. J. Buckley [3]

Ecuación 1-11 Reglas Difusas de Takagi Sugeno

Siendo x_1 y x_2 entradas del proceso, A y B miembros de una función de pertenencia difusa de entrada y la salida u_1 el valor de la función $f(x)$ para ese punto en concreto. Al igual que

mandami la resolución de la lógica es, si x_1 es A y x_2 es B entonces u_1 toma el valor de la función para ese punto.

1.1.4.5 Defusificación

La defusificación es el proceso matemático que se realiza para obtener el valor de un conjunto de pertenencia difusa de salida, los distintos métodos de defusificación arrojarán resultados diferentes para un mismo conjunto de reglas, por lo que es importante definir qué tipo de método se ajusta más a los requerimientos, el más común y más utilizado es el cálculo de centroide, Ecuación 1-12, que entrega el centro de gravedad de la coordenada x de la función de pertenencia de salida.

$$y_d = \frac{\int y\mu_y(y)dy}{\int \mu_y(y)dy}$$

Fuente: W. Siler y J. J. Buckley [3]

Ecuación 1-12 Cálculo de Centroide

Debido a que el costo computacional de este método es demasiado alto, también se suele implementar la defusificación por centro de área, la cual a su vez es un punto de partida para las demás formas de defusificación, como lo son el criterio del máximo y el método del centro de área.

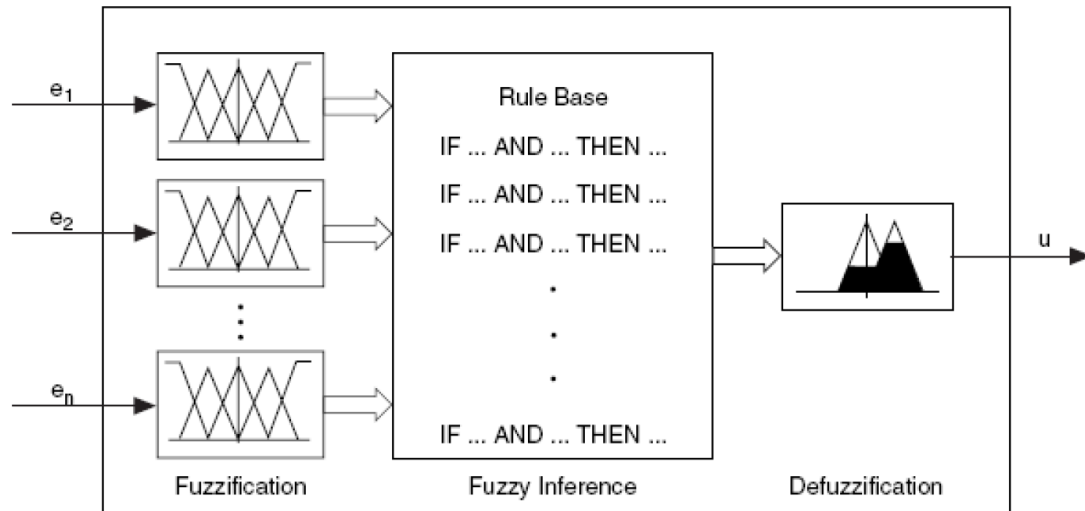
1.1.5 Estructura de un controlador difuso

Un controlador difuso tiene como estructura lógica tres pasos, fusificación, base de reglas y defusificación, como se muestra en la Figura 1-6, cada una de estas etapas tiene sus subsistemas ya definidos.

A continuación, se definirá cada etapa tomando en consideración lo expuesto con anterioridad en la Sección 1.1.4

- I. Fusificación: Se tienen los sets de membresías difusas de cada entrada y sus respectivas funciones de pertenencia.
- II. Inferencia Difusa: Se encuentra la definición de la base de reglas y los valores que tomará cada conjunto de salida dependiendo de la pertenencia del conjunto.

- III. Defusificación: Se procesan las funciones de pertenencia de la o las salidas para generar un valor numérico según el método de defusificación seleccionado.



Fuente: J. D. C. Quisaguano [4]

Figura 1-6 Estructura de un controlador difuso

1.2 HIPÓTESIS

Para esta investigación se plantea la siguiente pregunta de investigación: Utilizando el método propuesto para el diseño e implementación de controladores PID con ganancia programada, se logra una mejor performance en el proceso y una mayor flexibilidad para generar cambios en el sistema.

1.3 OBJETIVOS GENERAL Y ESPECÍFICOS

El objetivo general de este trabajo es demostrar que el diseño propuesto y la implementación de los algoritmos de un controlador PID con programación de ganancia basados en lógica difusa mejoran la performance de un proceso. Para cumplir con el objetivo general se pueden identificar los siguientes objetivos específicos:

- I. Diseñar e implementar controlador PID base para la modificación de ganancia
- II. Diseñar e implementar controlador Difuso para la modificación de parámetros en línea del controlador PID
- III. Implementar el control de ganancia programada de tal forma que los parámetros a modificar sean los mínimos para cada implementación de este.

- IV. Generar set de funciones de transferencia para las pruebas de generalización del control para determinar los parámetros del controlador difuso
- V. Implementar la simulación de la planta de dos estanques por llenado gravitatorio para poner a prueba el control PID de ganancia programada final.

1.4 ALCANCES Y LIMITACIONES

En este trabajo se busca establecer un esquema para un controlador PID de ganancia programada “genérico” que pueda ser implementado sin mayores cambios en los parámetros en cualquier sistema que cumpla con lo definido en el capítulo 2.

El foco de este trabajo no es la simulación ni la implementación de la planta como tal, sino más bien esta simulación es un instrumento para validar la hipótesis.

CAPÍTULO 2: CONTROLADOR PID CON PROGRAMACION DE GANANCIA

2 CONTROLADOR PID CON PROGRAMACION DE GANANCIA

En las siguientes subsecciones se detallará el diseño de todas y cada una de las partes que componen el controlador PID con programación de ganancia. Además, se realizará el modelo matemático de tanques conectados de forma no interactiva, el cual se utilizará para corroborar el funcionamiento del sistema controlado.

2.1 DISEÑO DEL CONTROLADOR PID

En la sección 1.1.2 se mostró la ecuación del controlador estándar, la que esta implementada en un gran número de aplicaciones industriales, en la Ecuación 2-1 se muestra la misma ecuación pero en el dominio de Laplace, de esta forma se trabajará en todo este capítulo.

$$C(s) = Kc * \left(1 + \frac{1}{Ti * S} + Td * s \right)$$

Fuente: Katsuhiko Ogata, Ingeniería de control moderna [2]

Ecuación 2-1 Controlador PID Estándar en Laplace

La implementación de esta fórmula en sistemas de control reales tiene una serie de problemas debido a las limitantes físicas de los actuadores, sensores y del medio en el que se implemente, es por lo que para robustecer el algoritmo se realizarán las siguientes modificaciones.

2.1.1 Modificaciones al algoritmo de PID

Para robustecer el algoritmo que se presentó anteriormente y debido a las distintas limitaciones que se presentan al implementar estos sistemas en medios físicos, se deben realizar ciertas modificaciones:

2.1.1.1 Limitación de ganancia derivativa

En los sistemas físicos las mediciones de las señales manipuladas se ven afectadas por el ruido, lo que provoca problemas en la parte derivativa del PID. Para solucionar este problema y

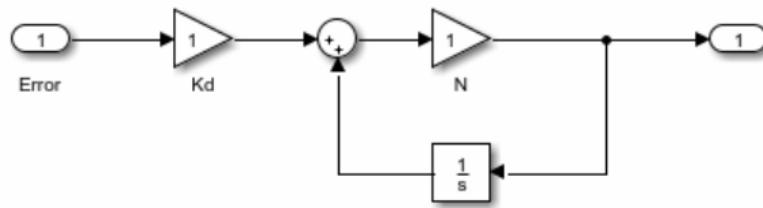
debido a que frecuentemente los ruidos que más afectan la medición son los de alta frecuencia, se debe implementar un filtro pasa bajos en la parte derivativa.

$$C(s) = \frac{N}{1 + N/s}$$

Fuente: Katsuhiko Ogata, Ingeniería de control moderna [2]

Ecuación 2-2 Implementación de Filtro Pasa bajos en la derivada

En la Ecuación 2-2 se muestra el filtro pasa bajos en el dominio de Laplace en donde N es la frecuencia de corte en radianes por segundos cuyos valores más comunes son 10 y 100. Si se despeja la ecuación se puede acomodar para poder realizar la implementación como un sistema retroalimentado, lo que simplifica el proceso de diseño como lo muestra la Figura 2-1.



Fuente: Elaboración Propia Software Matlab

Figura 2-1 Implementación filtro derivativo

2.1.1.2 Limitación de acción integral

La acción integral se ve afectada por las limitaciones físicas de los actuadores, si el sistema no logra regular el error a causa de que el actuador llega a su límite físico, la acción integral seguirá variando después del desbordamiento, lo que hará que al realizar la acción contraria exista una banda de no acción, a este fenómeno se le conoce como windup.

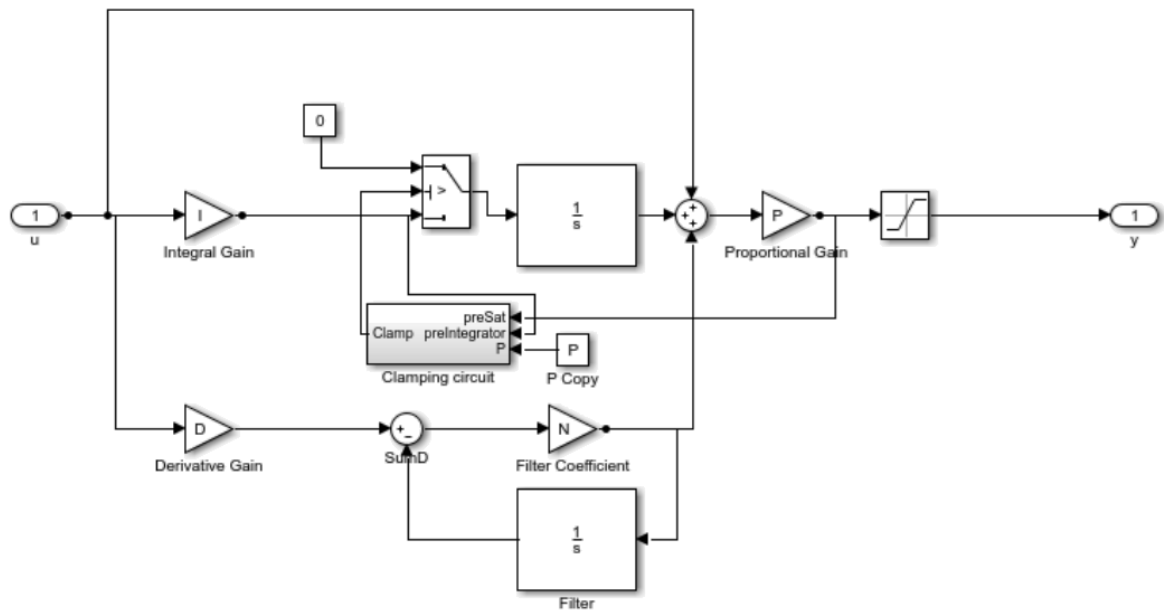
Para evitar este fenómeno existen algoritmos de anti-windup los cuales se describen a continuación.

- a. Clamping: Consiste en detectar el desbordamiento de la salida con respecto a los límites máximos y mínimos del sistema. Este método es el más común debido a que funciona en todos los casos y es sencillo de configurar.
- b. Back-Calculation: Al igual que en el método de clamping se busca limitar el desbordamiento, esto se logra mediante la retroalimentación de la diferencia entre la

salida limitada y la acción integral a la que se le aplica una ganancia (K_p). Este método depende en gran medida de la correcta sintonización de la ganancia K_p y en los casos en que esta no se encuentre bien sintonizada puede no tener efecto o en el peor caso afectar la respuesta del sistema.

2.1.2 Algoritmo PID a Implementar

Finalmente tomando en consideración lo presentado en los puntos anteriores es que se implementará el PID con el filtro derivativo y la limitación integral por clamping que se muestra en la Figura 2-2.



Fuente: Elaboración Propia Software Matlab

Figura 2-2 Implementación Controlador PID

2.2 DISEÑO CONTROLADOR DIFUSO

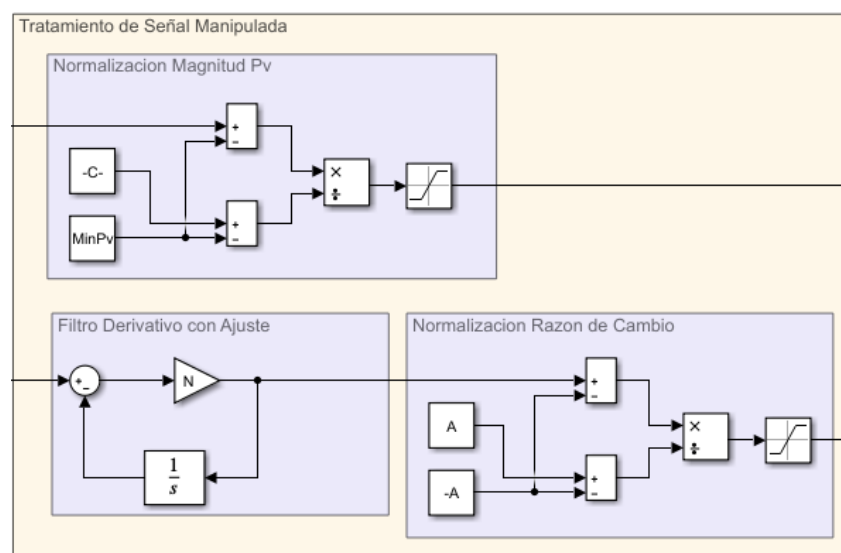
En esta sección se realizará el diseño del controlador difuso que se encargará de modificar los parámetros de sintonía del PID no interactivo.

2.2.1 Creación de conjuntos de entrada y salida

Para la creación de los conjuntos de entrada, en primer lugar, se deben definir las entradas al sistema difuso, en este caso se trabajará con la variable controlada y la derivada de esta, esto debido a las siguientes razones:

- I. Al utilizar la variable controlada y la razón de cambio se tiene un vector de magnitud y dirección respectivamente, lo cual da un nivel de detalle mayor del comportamiento de la señal.
- II. Debido a que se busca un enfoque general es bueno trabajar con la variable controlada, esto ya que si se trabajase con el error se deben ajustar los conjuntos según la magnitud y cambio de esta última.

Para poder generalizar el control es necesario llevar las magnitudes tanto de la variable controlada como la razón de cambio a porcentajes, para esto se debe realizar una normalización de la señal, el método de normalización que se utilizará para este controlador será el método de Scaling que consta de normalizar la variable de entrada entre dos rangos conocidos. Aunque este método tiene el problema de amplificar los errores de medición es una buena forma de mantener las proporciones sin necesidad de mayores análisis como en otros métodos, como lo puede ser la normalización por la media. En la Figura 2-3 se puede observar la implementación de dicha normalización en Matlab Simulink.



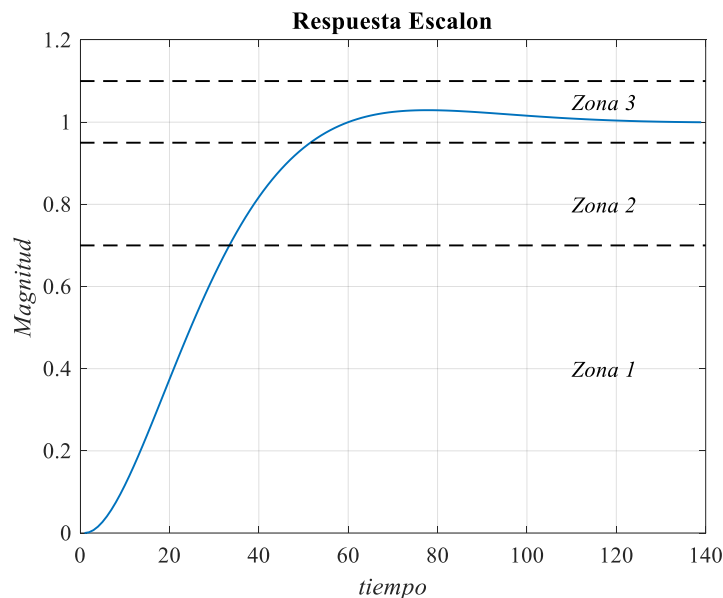
Fuente: Elaboración Propia Software Matlab

Figura 2-3 Normalización de la Variable Manipulada y Razón de Cambio

Ya con la normalización definida, se pueden generar los conjuntos difusos para la variable manipulada y su razón de cambio, para esto se utilizará el conocimiento que se tiene de los sistemas de primer y segundo orden donde se sabe que la respuesta ideal del sistema es exponencial por lo que se utilizarán las funciones de membresía delta, pi, sigma, zeta, las cuales se vieron en la sección 1.1.4.1.

Para la definición de la cantidad de funciones de membresía se utilizará la Figura 2-4 donde se muestra la respuesta de escalón unitario en un sistema de segundo orden, como se puede observar es posible definir tres zonas:

- Zona 1: Esta zona se define tomando en cuenta que desde cierto punto el error es tan grande que deja de importar su magnitud
- Zona 2: Una zona más cercana a la zona de estabilidad donde el error empieza a tener menos peso con respecto a la acción.
- Zona 3: Zona de estabilidad donde la acción de control disminuye notablemente y solo busca mantener el error de estado estacionario



Fuente: Elaboración Propia Software Matlab

Figura 2-4 Respuesta Escalón

Se analiza la respuesta a escalón debido a que con esto es posible observar los cambios de la señal en el tiempo, lo que permite determinar la cantidad de miembros necesarios para reconstruir el comportamiento de la señal. Si se toma en consideración que la Zona 3 es el lugar en donde el cambio es menor o nulo y debido a que la razón de cambio es tanto positiva como negativa se pueden definir un conjunto de 5 miembros para la razón de cambio, Muy positivos (PB), Positivo (P), Sin Cambio (SC), Negativo (N), Muy Negativo (NB).

Con respecto al conjunto para la variable manipulada se representará de forma tal que se genere una matriz de orden 5, esto es debido a que de esta forma se puede lograr una función de transferencia lineal si se definen reglas estándares y los miembros difusos tanto de entrada como de salida son equidistantes.

Para ajustar la ganancia del controlador PID las salidas del controlador difuso deben ser tres, el ajuste proporcional (Kp), el ajuste integrativo (Ki) y el ajuste derivativo (Kd), desde ahora en adelante cada vez que se nombre Kp, Ki y Kd se hará referencia a los valores de ajuste. Debido a que la matriz de conocimiento es cuadrada se puede aplicar uno de los criterios de diseño utilizados en la industria para la simplificación en la implementación de este tipo de sistemas que consta en agregar un miembro de salida a cada extremo, es decir, como la matriz de conocimiento es de orden 5 la salida debería tener 7 miembros.

2.2.2 Creación de reglas difusas

Para la creación de las reglas difusas y debido a que lo que se busca es ajustar los parámetros se seguirán las reglas generales de sintonización manual de controladores PID [2] que se muestra en la Tabla 2-1.

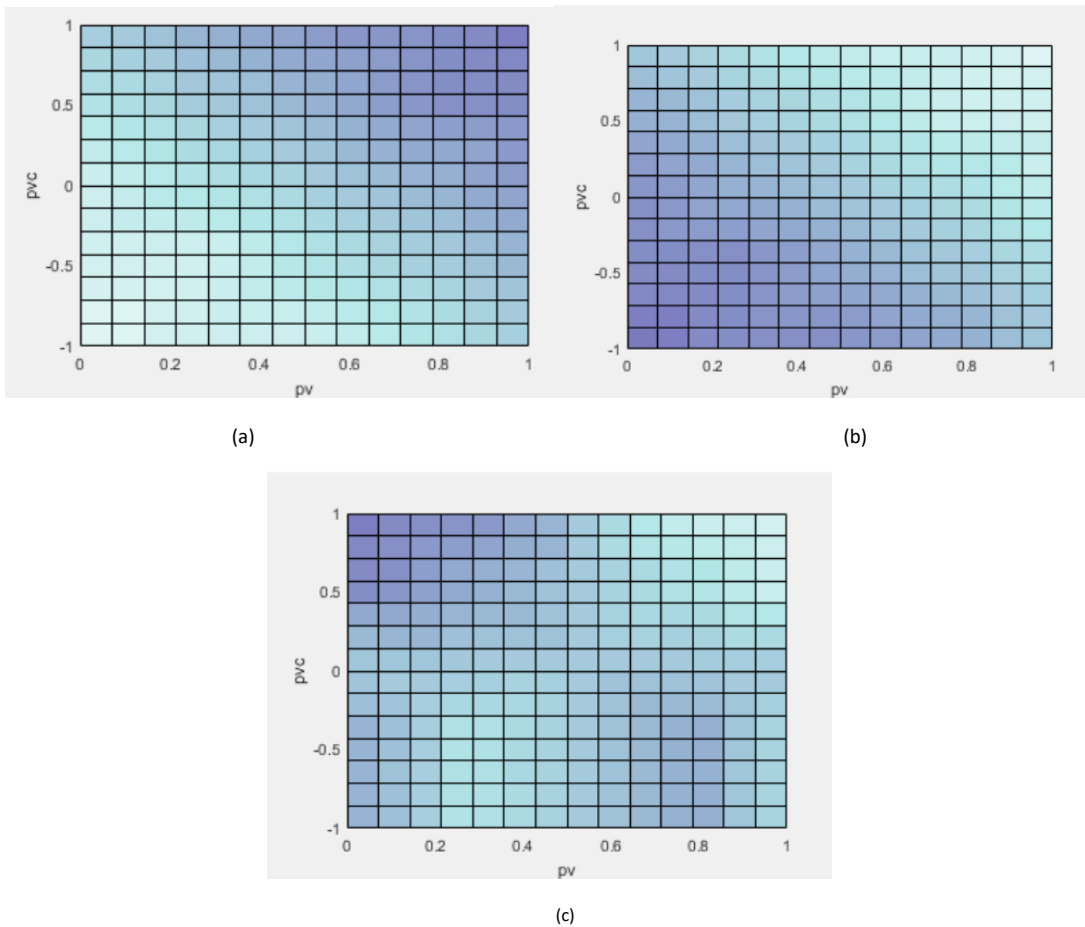
Tabla 2-1 Efectos de los parámetros PID

Action	Rise Time	Overshoot	Stability	Settling Time
Kc Increase	faster	Increases	gets worse	little Change
1/ti Increase	faster	Increases	gets worse	Increases
td Increase	slow	decreases	improves	Decreases

Fuente: J. Jantzen, «Tuning Of Fuzzy PID Controllers»

Como se puede ver en la tabla y tomando en consideración que la ganancia integral se define según la Ecuación 2-1 como $\frac{K_c}{T_i}$ y observando que este es el inverso de Kc las reglas difusas pueden invertirse en magnitud lo que simplifica el diseño. Con respecto al parámetro Kd es más difícil llegar a determinar las reglas, esto debido a que los cambios de parámetros en la derivada del PID fuera de la Zona 3 (zona de estabilización) podrían provocar que el margen de fase del sistema llegue a valores cercanos a los -180° , lo que haría oscilar el sistema, sumado a lo anterior también se presentan problemas importantes en el cálculo del factor N del filtro pasa bajos, el que también altera la fase del sistema controlado. Por esto, la modificación del parámetro Kd se realizará en la zona de estabilización y el factor de filtro N no se modificará.

Con las apreciaciones anteriores y los datos de la tabla 2-1, se llega a las reglas difusas, las cuales se grafican en la Figura 2-5.



Fuente: Elaboración Propia Software Matlab

Figura 2-5 Grafico Pseudo Color Blue - Reglas difusas : (a) K_p (b) K_i , (c) K_d

En la Figura 2-5 (a) se ven las reglas de K_p donde la acción proporcional aumenta a medida que la variable manipulada aumenta (azul oscuro) y disminuye en caso contrario (blanco). Esto se debe a que en un sistema de control directo de segundo orden a medida que aumenta la variable manipulada alejándose del punto de operación mayor es el error que se produce, por lo que el ajuste del parámetro K_p debe ser mayor para disminuir el tiempo de estabilización.

En la Figura 2-5 (b) se ven las reglas de la acción integral que, como se revisó anteriormente, son el inverso de las reglas del parámetro K_p .

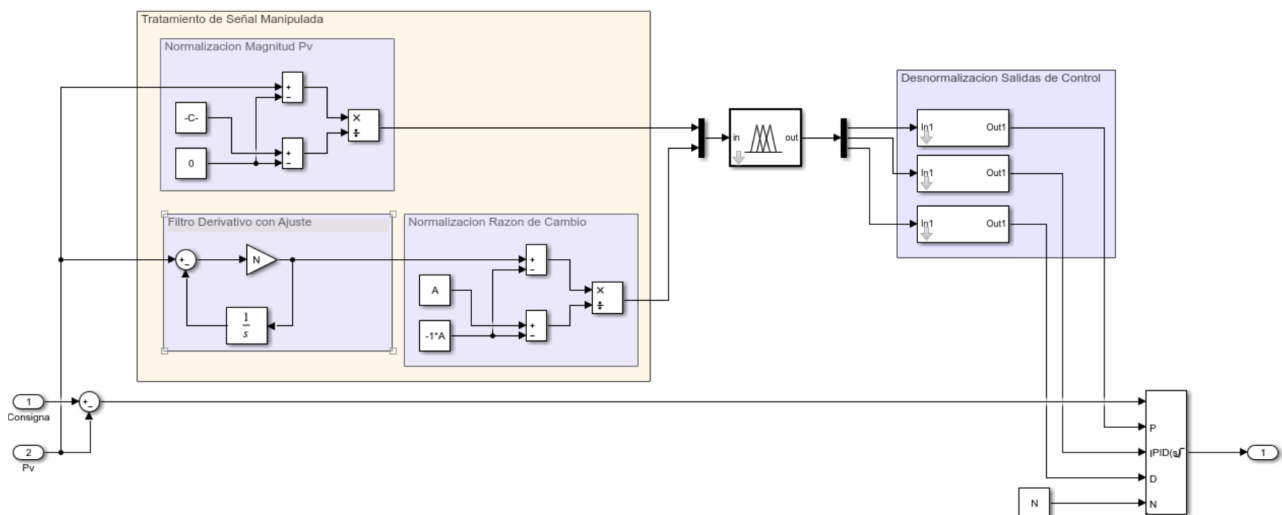
Para la acción derivativa que se muestran en la Figura 2-5 (c), los cambios son dentro de los miembros difusos P, SC, N y con acciones limitadas o nulas dependiendo fuertemente de la razón de cambio de la variable manipulada.

Todas las reglas se presentan en forma detallada en la sección de Anexos del documento

2.3 IMPLEMENTACIÓN CONTROLADOR PID CON PROGRAMACIÓN DE GANANCIA

Para la implementación se realiza la unión de los controladores con un esquema sencillo realizado en Matlab Simulink que se muestra en la Figura 2-6, se puede dividir la implementación en cuatro.

- Tratamiento de señal: En esta sección se realiza el cálculo de la razón de cambio de la variable manipulada, así como la normalización de las señales de entrada del controlador difuso de ajuste paramétrico
- Controlador difuso: Encargado de calcular el ajuste de parámetros para el controlador PID de la planta.
- Manejo de parámetros de salida: Se realiza la desnormalización de la señal de salida para el ajuste de parámetros al controlador PID dentro del rango configurado en la aplicación.
- Controlador PID: Controlador PID principal del sistema



Fuente: Elaboración Propia Software Matlab

Figura 2-6 implementación final control PID con ajuste de ganancia

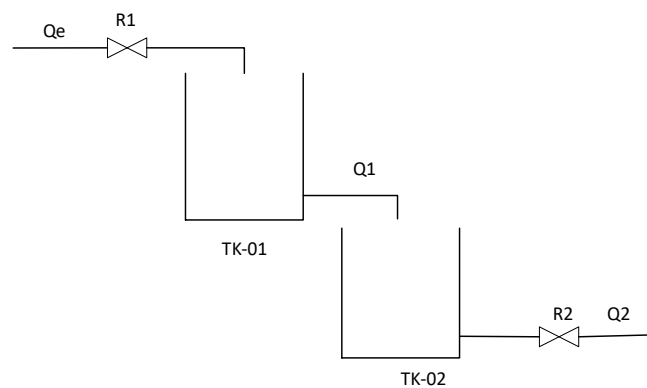
2.4 MODELO MATEMÁTICO DE TANQUES CONECTADOS DE FORMA NO INTERACTIVA

En un sistema hidráulico, que es el sistema a modelar, existen tres tipos de elementos básicos a considerar: la resistencia, que en este caso es la dificultad que encuentra el fluido para circular a través del circuito; la capacitancia, que indica el cambio que se debe efectuar en el sistema para que este tenga un cambio de potencial y la inercia, que es el cambio necesario

para romper la inercia para poder producir un cambio. En las siguientes secciones se mostrará el proceso de modelado y la correspondencia de dichos factores en el proceso.

2.4.1 Definiciones de modelamiento

Para la simulación del proceso de tanques con conexión no interactiva que se muestra en la Figura 2-7 se define que las distancias entre tuberías que conectan los tanques y alimentan a estos son cercanos por lo que no se generara un retraso en el cambio del nivel, además se define que el líquido a transportar será agua la cual no tiene suficiente densidad para generar un retraso en el traslado de esta de un tanque a otro por lo que teniendo en cuenta estas definiciones el factor inercia es despreciable.



Fuente: Elaboración Propia

Figura 2-7 Tanques conectados de forma no interactiva

Para el cálculo de la capacitancia del sistema se sigue la definición que se muestra en la siguiente ecuación.

$$C = \frac{\text{Cambio en el Líquido Almacenado}}{\text{cambio en la Altura}}$$

Fuente: Katsuhiko Ogata, *Ingeniería de control moderna* [2]

Ecuación 2-3 Definición de capacitancia del tanque

Si se observa la ecuación esta corresponde a la derivada del volumen dividida en la derivada de la altura del tanque. Si se considera que el tanque es cilíndrico y se reemplaza el volumen por su definición matemática se obtiene que la capacidad \$C\$ es el área transversal del

tanque, esta definición es aplicable para ambos estanques debido a que no depende de ningún otro factor.

Si se realiza un despeje de la Ecuación 2-3 trabajando en el dominio del tiempo se pueden obtener las siguientes ecuaciones para ambos tanques.

$$C_1 \frac{dH_1}{dt} = (Q_e - Q_1)$$

$$C_2 \frac{dH_2}{dt} = (Q_1 - Q_2)$$

Fuente: Elaboración Propia – Software wolfram mathematica

Ecuación 2-4 Ecuaciones de variación de volumen

La resistencia del sistema como se vio anteriormente no es más que la restricción que encuentra el fluido para avanzar como se muestra en la siguiente ecuación.

$$R = \frac{\text{Cambio en la Diferencia de Nivel}}{\text{cambio en la velocidad del Flujo}}$$

Fuente: Katsuhiko Ogata, *Ingeniería de control moderna* [2]

Ecuación 2-5 Definición de resistencia

En la Figura 2-7 Tanques conectados de forma no interactiva se observa que las restricciones R1 y R2 están dadas por las válvulas de control y la resistencia que existe entre la comunicación de los dos tanques. La resistencia del sistema entonces dependerá de la geometría de la válvula, por lo que, si se presume que el comportamiento de las válvulas es proporcional al flujo y que este no es laminar, es posible definir un factor de descarga en metros cúbicos por segundo para obtener un modelo de válvula de control. Lo mismo ocurre para la resistencia del orificio de salida del tanque uno. Según estas definiciones se obtienen las siguientes ecuaciones

$$Q_{\text{SecTransversal}} = K\sqrt{H}$$

$$Q_{\text{Valvula}} = K * a * \sqrt{H}$$

Fuente: [10]

Ecuación 2-6 Ecuaciones de caudal según restricción

Definido lo anterior se pueden despejar los flujos de salida desde la Ecuación 2-5, como se puede observar en las siguientes ecuaciones

$$Qe = K_1 a_1$$

$$Q_2 = K_2 \sqrt{H_1}$$

$$Q_2 = K_3 a_2 \sqrt{H_2}$$

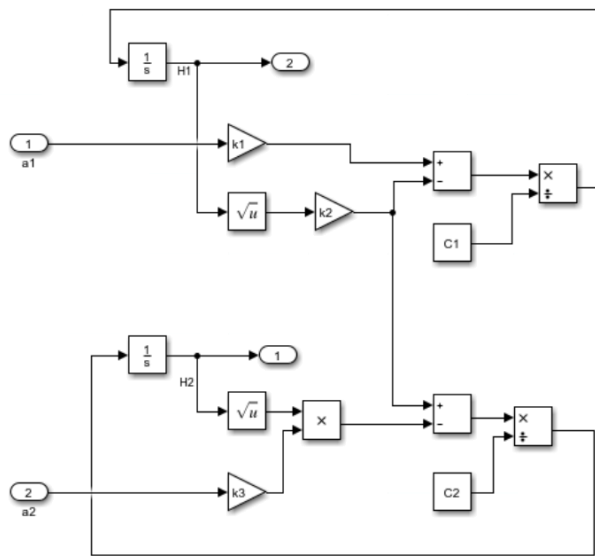
Fuente: Elaboración Propia – Software wolfram mathematica

Ecuación 2-7 Ecuaciones de caudal según restricción

2.4.2 Construcción del modelo y ajuste de parámetros

Con las definiciones antes realizadas se han obtenido las ecuaciones que describen un modelo fenomenológico del sistema de dos tanques no interactivos por lo que se procede a realizar la construcción del modelo en Matlab/Simulink.

En la Figura 2-8 se muestra la implementación de las ecuaciones descritas en la sección 2.4.1 sustituyendo para dejar todo en función de la altura.



Fuente: Elaboración Propia Software Matlab

Figura 2-8 Implementación Modelo Tanques No Interactivos

Para la parametrización se designan valores de manera arbitraria teniendo en cuenta que el modelo responde a variaciones pequeñas de caudal. Los parámetros seleccionados se muestran en la siguiente tabla.

Tabla 2-2 Parámetros para Modelo Fenomenológico

k1	k2	k3	a1	a2	C1	C2
0,15	0,04	0,07	0 a 1	0,45	1	1,5

Fuente: Elaboración Propia

2.5 OBTENCION DE PARAMETROS DE CONTROL PID Y FUZZY

Para la obtención de los parámetros de ajuste del PID es necesario realizar la linealización del modelo fenomenológico, para esto se necesita calcular los valores de las variables de salida en estado estacionario, para este caso particular el sistema es estable cuando Q_e es igual a Q_2 , esto se cumple cuando la derivada del nivel se vuelve cero.

Si se vuelven a igualar las ecuaciones de la sección 2.4.1 y se considera que R_2 se mantiene constante, se obtienen las siguientes ecuaciones.

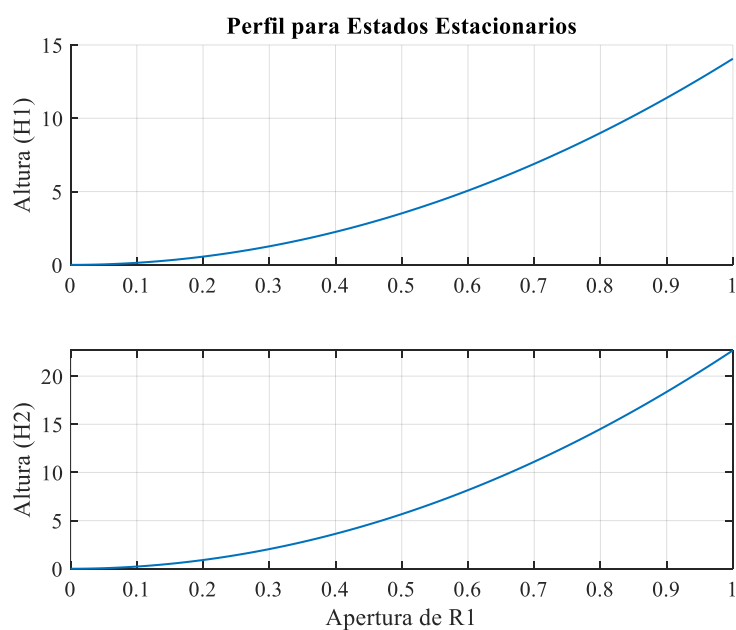
$$H_{1s} = \left(\frac{K_1 a_1}{K_2} \right)^2 \quad H_{2s} = \frac{K_2^2 H_{1s}}{(K_3 a_2)^2}$$

(a) (b)

Fuente: Elaboración Propia – Software wolfram mathematica

Ecuación 2-8 H1 tanque 1 (a) H2 tanque 2 (b) ambos en estado estable

Con estas ecuaciones es posible construir un perfil de estados a diferentes aperturas de válvula R1, como se puede observar en la Figura 2-9 el perfil es exponencial, por regla general para recrear una señal de esta características se tomen de tres a cinco puntos, tomando en cuenta que la válvula se mueve de 0% a 100% (0 a 1) para obtener cinco puntos de operación distintos se deberían crear cinco modelos lineales con aperturas del válvula del 20%



Fuente: Elaboración Propia Software Matlab

Figura 2-9 Perfil de Niveles en Estado Estacionario

En la tabla se pueden apreciar los valores en estado estacionario para lo anteriormente comentado, estos valores se condicen con los valores de la Figura 2-9 y serán utilizados más adelante para linealizar la planta en los distintos puntos de operación

Tabla 2-3 Valores en Estado Estable

Valores del sistema en estado estable		
Válvula %	H1	H2
0,2	0,56	0,91
0,4	2,25	3,63
0,6	5,06	8,16
0,8	9,00	14,51
1	14,06	22,68

Fuente: Elaboración Propia

2.5.1 Linealización por Series De Taylor

Las series de Taylor son una aproximación lineal de funciones mediante una serie de ecuaciones parciales que describen el comportamiento lineal del sistema, debido a que se necesita encontrar sintonías para distintos puntos de operación se utilizará la serie solo hasta la primera derivada del sistema, lo que se conoce como Jacobiana del sistema.

$$F(x) = F(x^*) + \nabla F(x^*) \Delta x$$

Fuente: libro zotero se mario

Ecuación 2-9 Serie de Taylor - Jacobiana del Sistema

Sabiendo que la variable de entrada al primer tanque es a_1 y la salida es H_1 y tomando en cuenta las ecuaciones de la sección 2.4, se tiene que para el TK01 la ecuación de Taylor es la siguiente:

$$F(a_1, H_1) \approx F(a_{1s}, H_{1s}) + \left. \frac{dF(a_1, H_1)}{da_1} \right|_{(a_{1s}, H_{1s})} * (a_1 - a_{1s}) + \left. \frac{dF(a_1, H_1)}{dH_1} \right|_{(a_{1s}, H_{1s})} * (H_1 - H_{1s})$$

Fuente: Elaboración Propia – Software wolfram mathematica

Ecuación 2-10 Serie Taylor tanque uno

Gracias a que se tiene la función $F(a_1, H_1)$ solo hace falta realizar las operaciones de derivada evaluadas en los puntos de operación y luego despejar para la variable H_{1s} lo que se muestra a continuación.

$$C_1 \frac{d\Delta H_1}{dx} = K_1 \Delta a_1 - \frac{K_2}{2\sqrt{H_{1s}}} \Delta H_1$$

Fuente: Elaboración Propia – Software wolfram mathematica

Ecuación 2-11 Linealización tanque uno

Se debe hacer notar que tanto $(a_1 - a_{1s})$ y $(H_1 - H_{1s})$ se expresan como variables de desviación, esto con el fin de hacer más entendible la ecuación. Si se sigue el mismo procedimiento se puede obtener para el tanque dos la siguiente ecuación.

$$C_2 \frac{d\Delta H_2}{dx} = \frac{k_2}{2\sqrt{H_{1s}}} \Delta H_1 - \frac{k_3 a_2}{2\sqrt{H_{2s}}} \Delta H_2$$

Fuente: Elaboración Propia – Software wolfram mathematica

Ecuación 2-12 Linealización tanque dos

Para obtener el modelo en función de H_2/a_1 solo es necesario representar ambos modelos en el dominio de la Laplace y multiplicarlos, esto dará origen a la siguiente función de transferencia de segundo orden con la que se trabajará de ahora en adelante.

$$\frac{H_2(s)}{a_1(s)} = \frac{\frac{k_1 k_2}{2\sqrt{H_{1s}}}}{\left(C_1 s + \frac{k_2}{2\sqrt{H_{1s}}}\right) \left(C_2 s + \frac{k_3 a_2}{2\sqrt{H_{2s}}}\right)}$$

Fuente: Elaboración Propia – Software wolfram mathematica

Ecuación 2-13 Función de Transferencia del Sistema

Con la función de transferencia del sistema ya es posible generar con ayuda de la Tabla 2-3 las diferentes funciones de transferencia para los distintos puntos de operación, esto se realiza mediante un pequeño script que toma los valores calculados anteriormente y los ingresa a la Ecuación 2-13

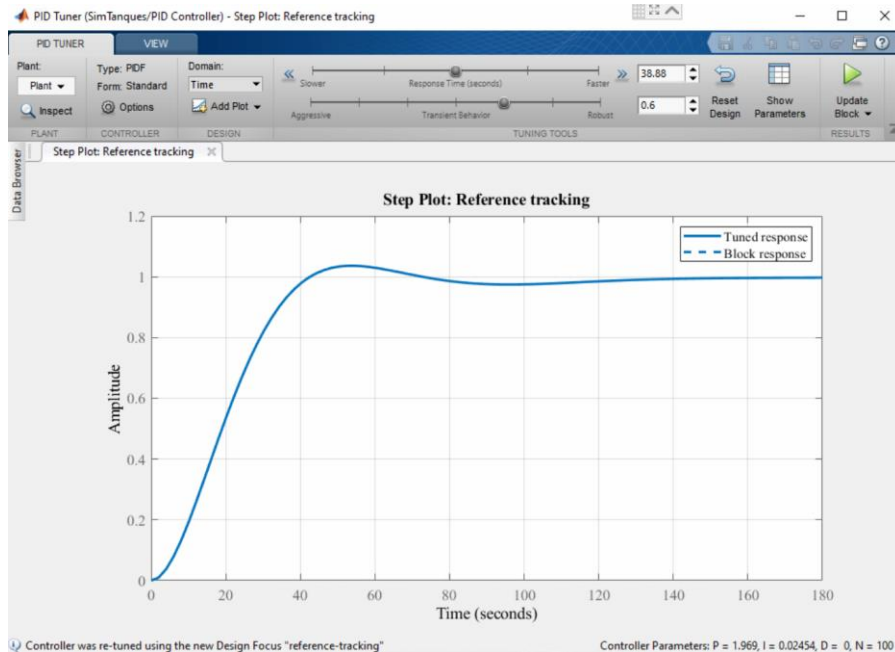
<p>From input "a1" to output "H2": 0.002662</p> <p>-----</p> <p>$s^2 + 0.03763 s + 0.000293$</p>	<p>From input "a1" to output "H2": 0.0006667</p> <p>-----</p> <p>$s^2 + 0.009424 s + 1.838e-05$</p>	<p>From input "a1" to output "H2": 0.001333</p> <p>-----</p> <p>$s^2 + 0.01884 s + 7.346e-05$</p>
<p>From input "a1" to output "H2": 0.0008891</p> <p>-----</p> <p>$s^2 + 0.01257 s + 3.268e-05$</p>	<p>From input "a1" to output "H2": 0.0005333</p> <p>-----</p> <p>$s^2 + 0.007538 s + 1.176e-05$</p>	

Fuente: Elaboración Propia Software Matlab

Figura 2-10 Funciones de transferencia en Estado Estacionario

2.5.2 Obtención de los controladores PID

El ajuste del controlador será realizado con el software Matlab cuya interfaz de usuario para su aplicación de sintonización se muestra en la Figura 2-11. El software está programado para lograr un equilibrio entre rendimiento y robustez.



Fuente: Elaboración Propia Software Matlab

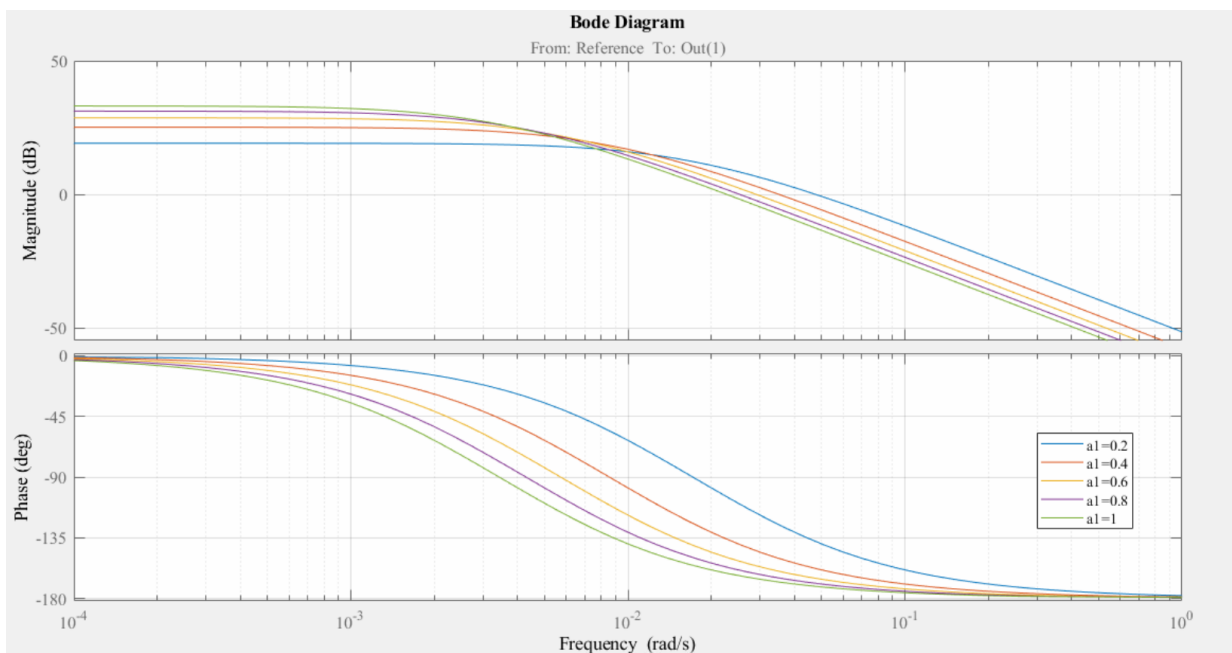
Figura 2-11 Interfaz Aplicación "PID Tuner"

Por defecto, el algoritmo elige el ancho de banda con la frecuencia de corte de la dinámica de la planta, y diseña para un margen de fase objetivo de 60 °, con esto se logra la robustez necesaria en caso de una variación en la dinámica de la planta o en algún error de modelado. Matlab implementa este algoritmo modificando el tiempo de respuesta, el ancho de banda, la respuesta transitoria o el margen de fase según sea solicitado en base a dos criterios seleccionables, rechazo a la perturbación y seguimiento de la referencia. Para ambos casos el

software siempre intenta obtener un valor del margen de fase mínimo lo más bajo posible, es decir, que para cada nuevo controlador cambiará el margen de fase. Es por esto por lo que se utilizará la línea de comandos de la herramienta ya que da mayor flexibilidad a la hora de sintonizar.

Si se utilizara la opción por defecto el software intentaría mantener la fase mínima de cada punto de operación que se esté sintonizando lo cual haría que algunas definiciones que se utilizaron al crear el controlador difuso se invalidaran, es por esto que se analizará el comportamiento frecuencial de cada uno de los puntos de operación calculados en la sección 2.5.

En la Figura 2-12 se puede apreciar el comportamiento frecuencial de cada uno de los modelos calculado anteriormente en un diagrama de bode, como se aprecia para todos los modelos el sistema es estable debido a que tanto el margen de fase como el margen de ganancia se encuentra dentro de la zona de estabilidad, es decir ninguno se encuentra sobre los 0 db ni por debajo de los -180° . Esto hace que el proceso de selección de fase mínima sea más sencillo, debido a que solo se debe elegir la mayor de las frecuencias de corte (W_{cp}), en este caso el valor de esta variable sería 0,04761. En caso de que uno de los sistemas fuese inestable se tendría que ajustar la sintonía a la fase mínima a dicho sistema o en caso de haber más de un sistema inestable, ajustar la fase mínima a un sistema cuya fase mínima comprenda al resto.



Fuente: Elaboración Propia Software Matlab

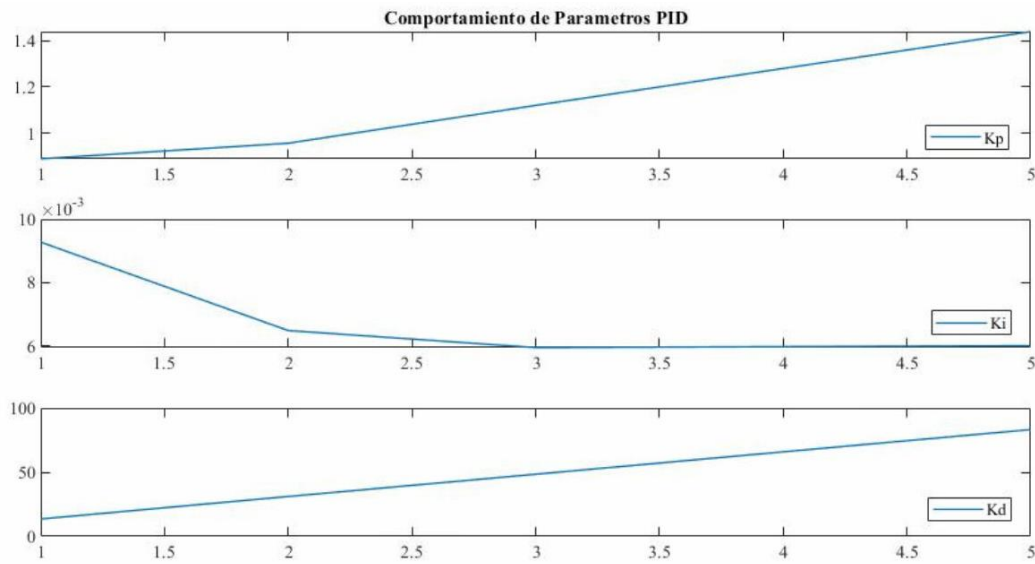
Figura 2-12 Diagrama de Bode cada función de transferencia

Ya habiendo seleccionado la frecuencia se pueden obtener los controladores para las cinco plantas ya diseñadas para esto se utiliza el comando "pidtune" desde la consola de Matlab

seleccionando cada sistema por separado. Este proceso está también dentro del script preparado con anterioridad el cual puede ser visto en el anexo A.

2.5.3 Obtención Parámetros Controlador Difuso

En la Figura 2-13 se puede apreciar el comportamiento que tienen los parámetros de los controladores PID obtenidos en la sección anterior donde se aprecia que es el mismo comportamiento que simula la configuración del controlador difuso, lo que confirma que las suposiciones en base a los datos teóricos son correctas mientras se mantenga el margen de fase para los controladores.



Fuente: Elaboración Propia Software Matlab

Figura 2-13 Comportamiento parámetros PID en estados estables

Es por lo anterior que solo es necesario utilizar el primer y último controlador obtenido en la sección 2.5.2 con lo que se obtendrían los parámetros que se muestran en la Tabla 2-4

Tabla 2-4 Tabla de Parámetros Lógica Difusa

Controladores Difuso		
Parámetro	Mínimo	Máximo
Kp	0,692	0,876
Ki	0,00403	0,00185
Kd	12,1	96,2
N	21	14,8

Fuente: Elaboración Propia

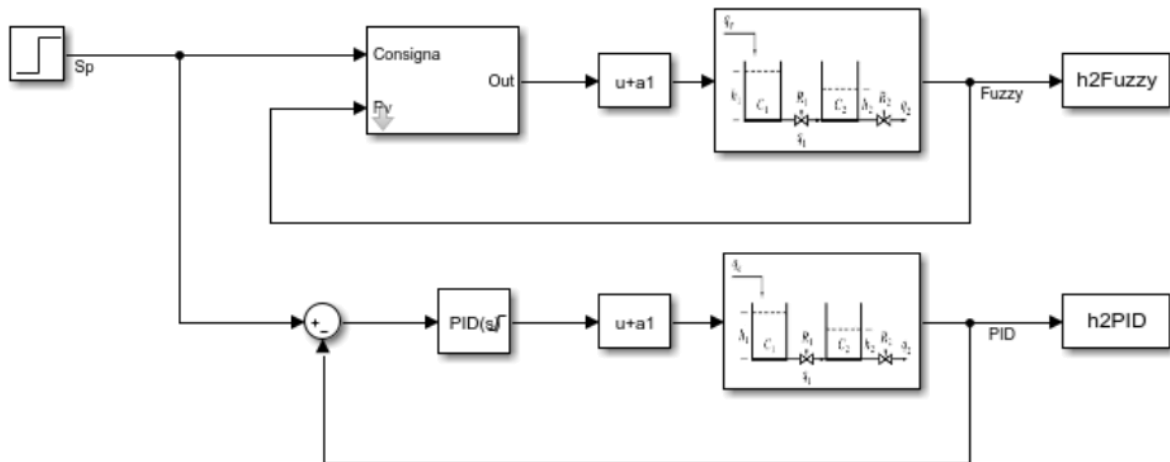
CAPÍTULO 3: RESULTADOS DE LAS PRUEBAS DEL CONTROLADOR PROPUESTO

3 RESULTADOS DE LAS PRUEBAS DEL CONTROLADOR PROPUESTO

En las siguientes secciones se mostrarán los resultados y las modificaciones realizadas al controlador propuesto según las especificaciones descritas en el Capítulo 2.

3.1 PREPARACIÓN PARA LAS PRUEBAS DE RENDIMIENTO DEL CONTROLADOR

Para la realización de las pruebas se hace una implementación conjunta de los componentes diseñados en el capítulo dos y un controlador PID en Simulink como se puede apreciar en la Figura 3-1.



Fuente: Elaboración Propia Software Matlab

Figura 3-1 Simulación para Pruebas de Rendimiento

El controlador PID servirá para realizar una comparativa entre el diseño del controlador con ganancia programada y un controlador PID con un solo ajuste de ganancia. Para la sintonización de este PID fue necesario calcular el punto de operación medio dentro del rango de operación total, como se definió en el capítulo dos el rango completo de operación corresponde al nivel tomado por H2 cuando a1 está entre 0 y 1, por lo tanto, se realiza la sintonía para a1 igual a 0,5.

Controlador (mask)

Bloque Controlador PID con Ajuste de Ganancia Difuso

Ajuste Pv

Maximo 22 Minimo 0

Ajuste Razon de Cambio 0.05

Limites de Salida

Maximo 1 Minimo 0

Parameters

Parametros Fuzzy

Controller

FuzzyControl

Filtro Pasabajos 15

Banda de Control

kp max C05.Kp kp min C01.Kp

ki max C05.Ki ki min C01.Ki kd min C05.Kd kd min C01.Kd

Fuente: Elaboración Propia Software Matlab

Figura 3-2 Simulación para Pruebas de Rendimiento

Junto con lo anterior se genera un conjunto de script que sirven para automatizar las pruebas los que pueden verse en el Anexo B. Dicho conjunto de script tiene el objetivo de generar para el sistema simulados los puntos de operación antes calculados y realizar un ajuste de setpoint de un 20% por sobre el valor inicial del modelo.

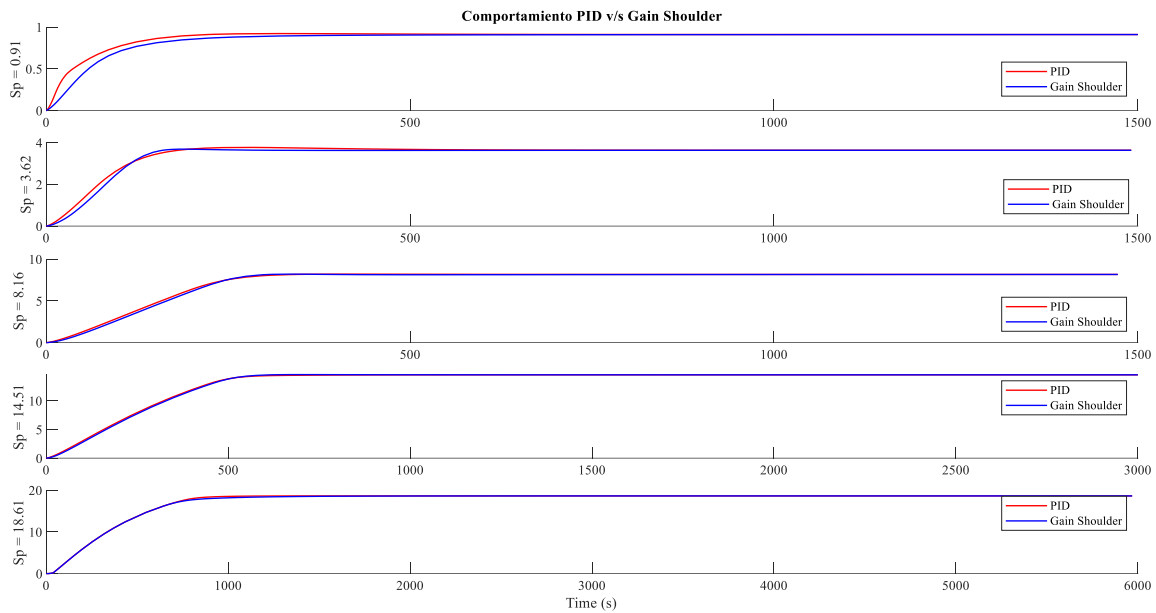
3.2 RESULTADO DE RENDIMIENTO DEL CONTROLADOR

Para determinar si los resultados del controlador diseñado en el capítulo dos son satisfactorios se realizarán dos pruebas de las cuales se analizará el comportamiento de manera gráfica para luego revisar los indicadores básicos de performance de cualquier sistema de control.

3.2.1 Pruebas de comportamiento a distintos puntos de operación

Para esta prueba se han considerado los cinco puntos de operación antes calculados y que se pueden observar en la Tabla 2-3. En la Figura 3-3 se muestran los resultados de dichas pruebas en donde se puede apreciar y como era de esperar a medida que el controlador PID se aleja del punto de operación para el cual fue sintonizado su comportamiento es menos eficiente, se debe recordar que ambos controladores están sintonizados para seguir la referencia por sobre la rapidez para alcanzar el valor de referencia por lo que los overshoot que sufre en los niveles más bajos de H₂ no deberían ser más altos que el 5% por diseño.

Por otra parte, el comportamiento del controlador con ganancia programada está dentro de los parámetros establecidos y aunque demora más tiempo en algunas ocasiones en llegar al estado estable para todos los valores de H2 este se comporta según lo diseñado.



Fuente: Elaboración Propia Software Matlab

Figura 3-3 Comportamiento Controladores a Distintos Puntos de Operación

En la Tabla 3-1 se puede ver el análisis de los controladores a la respuesta escalón de la Figura 3-3 para los diferentes puntos de operación mientras que en la Tabla 3-2 se ven las estadísticas descriptivas de los valores de la Tabla 3-1, con estas dos tablas se puede realizar un análisis más detallado para los distintos parámetros.

Tabla 3-1 Análisis de Respuesta al Escalón

Simulación	Rise Time		Settling Time		Overshoot	
	PID	Gain Shoulder	PID	Gain Shoulder	PID	Gain Shoulder
1	115,64	145,72	153,51	217,96	1,29	0,00
2	114,32	102,45	151,68	140,42	3,74	1,46
3	204,16	199,55	266,49	261,77	0,45	0,45
4	397,92	398,32	498,20	500,54	0,00	0,27
5	601,38	602,93	766,34	804,24	0,00	0,00

Fuente: Elaboración Propia

Tabla 3-2 Estadística Descriptiva de las Simulaciones para cada Parámetro

Datos Estadísticos	Rise Time		Settling Time		Overshoot	
	PID	Gain Shoulder	PID	Gain Shoulder	PID	Gain Shoulder
Media	286,68	289,79	367,24	384,99	1,10	0,44
Desviación estándar	210,47	208,47	263,97	270,19	1,57	0,60
Rango	487,06	500,48	614,66	663,82	3,74	1,46
Mínimo	114,32	102,45	151,68	140,42	0,00	0,00
Máximo	601,38	602,93	766,34	804,24	3,74	1,46

Fuente: Elaboración Propia

3.2.1.1 Análisis del tiempo de subida

El tiempo de subida se define como el tiempo en que el sistema tarda en llegar desde el 10% al 90% del valor final de la señal, la media de este tiempo en el controlador de ganancia con respecto al controlador PID es bastante similar, solo hay aproximadamente 3 segundos de diferencia, lo que sumado a la desviación estándar de este parámetro que es prácticamente idéntico para ambos, por lo que se puede decir que en cuanto al tiempo de subida ambos controladores se comportan de forma similar.

3.2.1.2 Análisis del tiempo de estabilización

Este tiempo está definido como el tiempo que se tarda el error en llegar al estado estable, dicho estado fue definido para estas pruebas como una variación del valor final de la señal de un 5%. En este caso la diferencia entre ambos controladores, aunque no es significativa marca una diferencia en el comportamiento del sistema, para el controlador con ganancia programada el tiempo de establecimiento es entre un 6% y un 7% más lento.

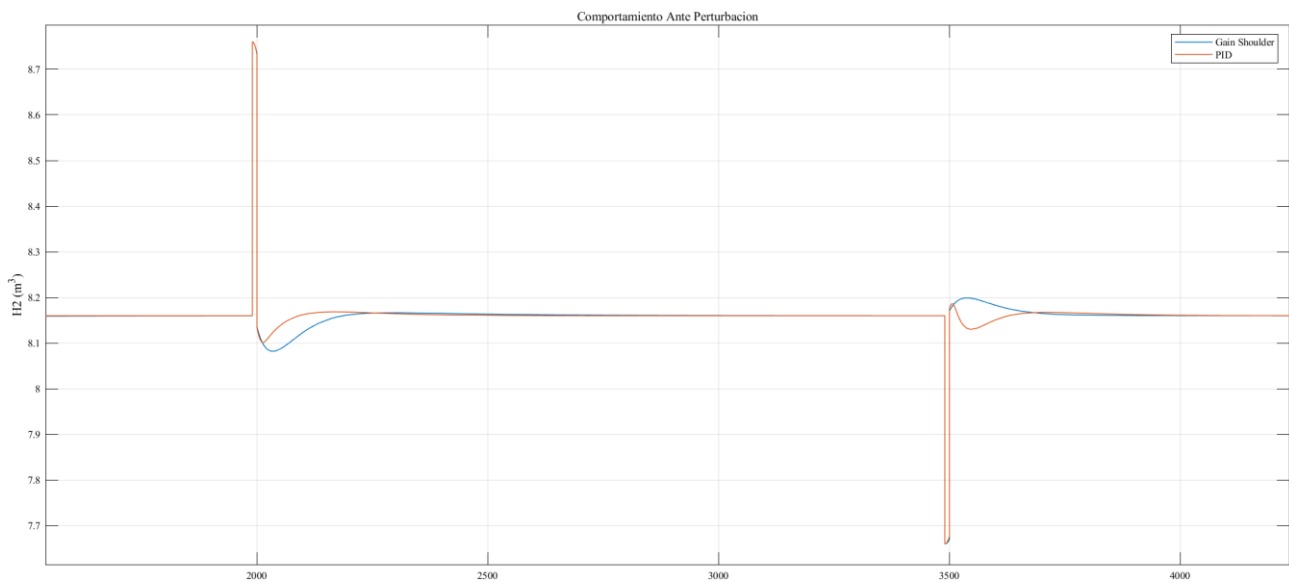
3.2.1.3 Análisis del tiempo de Overshoot

Se define como el porcentaje entre el valor máximo por sobre el setpoint alcanzado por la señal y el valor final de esta, para este diseño se sintonizaron los controladores para un overshoot no mayor al 5%. Este parámetro afecta notablemente los dos parámetros anteriores, debido a que la forma de controlarlo es disminuyendo los tiempos de subida lo que aumenta el tiempo de establecimiento.

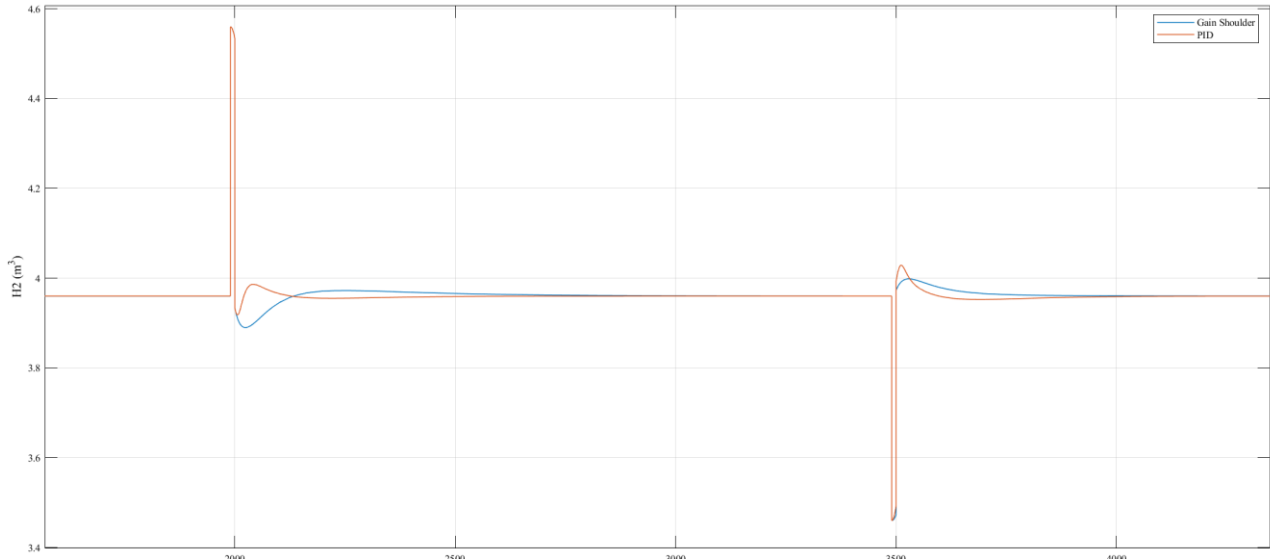
Aunque ninguno de los dos controladores excede el 5% de overshoot establecido al momento de elegir los parámetros de sintonía con el controlador de ganancia programada se nota una mejora significativa en lo que anteriormente se definió como zona 1 (Figura 2-4) que corresponde a la zona con mayores no linealidades del sistema y a su vez una mejora dentro del rango de control con un overshoot máximo de 1,47 un poco menos de la mitad del overshoot del controlador PID estándar.

3.2.2 Pruebas Ante Perturbaciones

En la Figura 3-4 se muestra el comportamiento ante una perturbación en el nivel H2, en la Figura 3-4 (a) se muestra el comportamiento que se tiene para el punto de operación para el cual se sintonizó el PID donde se ve claramente que el tiempo de restablecimiento y overshoot del sistema son mucho mejores que el del sistema con ganancia programada, mientras que para el caso dos (Figura 3-4(b)) se muestra un punto de operación fuera del rango operativo del PID estándar en donde se ve el aumento del overshoot y la aparición de undershoot aunque los tiempos de establecimiento son menores y los overshoot no superan los márgenes establecidos.



(a)



(b)

Fuente: Elaboración Propia Software Matlab

Figura 3-4 Comportamiento en caso de perturbaciones

3.3 CONCLUSIONES A LAS PRUEBAS PARA EL CONTROL PID

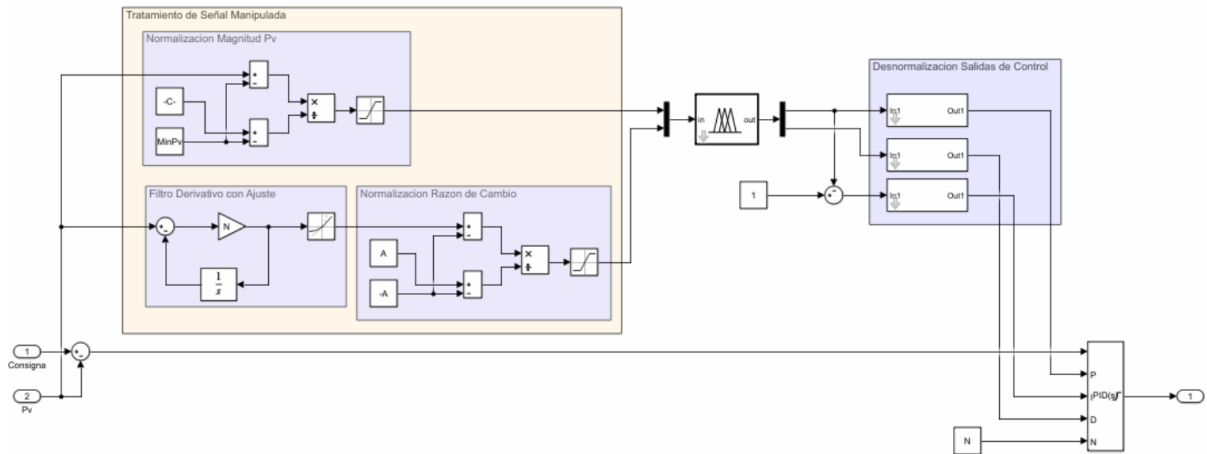
Como se vio en las tablas los resultados para el controlador con ganancia programada este es bastante lineal en todos los puntos de operación, lo que demuestra que es una mejor opción en caso de que el comportamiento del sistema sea no lineal.

Aunque el controlador propuesto tiene una mejor performance en general, si se realiza una comparación con el controlador PID Estándar para un punto de operación en concreto el controlador PID Estándar este disminuye su efectividad, esto es debido a que en la banda de puntos de operación el controlador con ganancia programada utiliza una curva de ajuste determinada por las reglas del controlador difuso lo que hace que el ajuste, aunque dentro de las especificaciones sea de menor performance.

3.4 SIMPLIFICACIÓN DEL CONTROLADOR PROPUESTO

Con las pruebas anteriores se demuestra que el sistema cumple con las características para las cuales fue diseñado y que las reglas basadas en la teoría de comportamiento de los sistemas de segundo orden son funcionales. Por lo que se puede realizar una simplificación del mismo con el objetivo de obtener los mismos resultados con un menor costo en la implementación del controlador.

En la Figura 3-5 se muestra la nueva configuración del controlador, como se puede apreciar, ya no existe la salida del conjunto difuso de K_i , esta fue reemplazada por una ecuación sencilla que invierte el resultado de K_p generando la misma salida que anteriormente tenía K_i . Junto con esto también se eliminan las reglas y conjuntos pertenecientes a este parámetro lo que hace más sencillo realizar cambios a las reglas manteniendo la premisa de que K_i se comporta inversamente a K_p .



Fuente: Elaboración Propia Software Matlab

Figura 3-5 Implementación final control PID con ajuste de ganancia

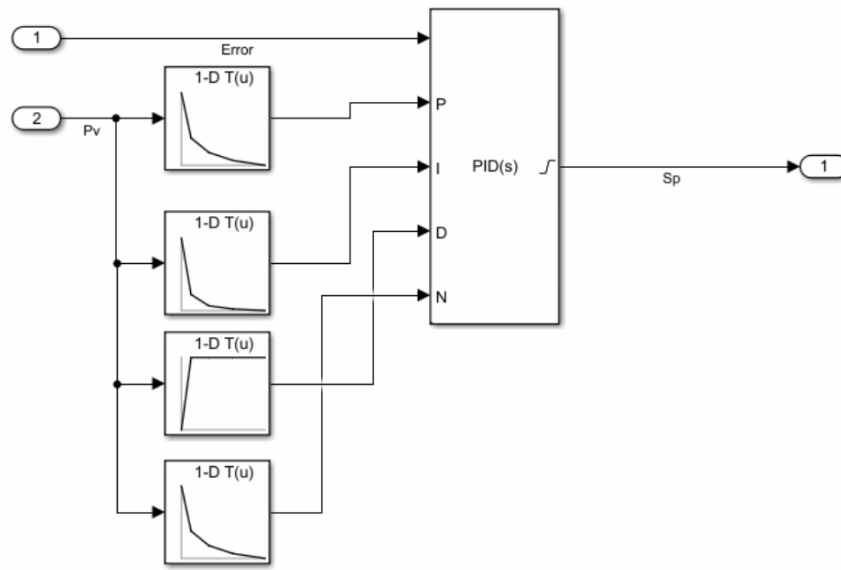
También se mejora el tratamiento de las señales de entrada y salida saturando la entrada hacia los conjuntos difusos, este límite no afecta el correcto funcionamiento de la lógica difusa debido a que esta considera cualquier resultado mayor a 1 con la máxima acción de control y menor a 0 con la mínima acción de control y obedece más a una optimización de código para eliminar warning y errores por valores infinitos.

3.5 COMPARATIVA CONTROLADOR PROPUESTO CON RESPECTO A LA TÉCNICA CLÁSICA

Una de las estrategias más usadas para afrontar este tipo de problemas de control en donde las no linealidades del sistema afectan el rendimiento de manera notable en distintos puntos de operación son los controladores de ganancia programada por tabla dinámica o de estados. Estos sistemas son simples y están formados por una serie de tablas con valores predefinidos para cada punto de operación, la salida de dichas tablas puede ser interpolada entre cada punto o sencillamente ir cambiando dentro del umbral definido por la sintonía.

En la Figura 3-6 se muestra la implementación del sistema en Matlab/Simulink de un control de ganancia programada por tablas de estados, se han utilizado los mismos puntos de

operación para la creación de las tablas y se ha decidido interpolar los datos en vez de generar saltos entre los valores de quiebre de la tabla.

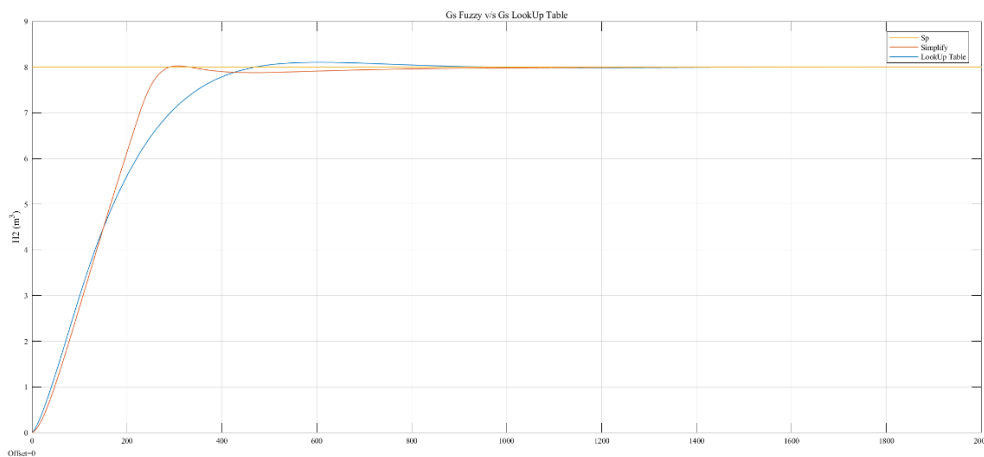


Fuente: Elaboración Propia Software Matlab

Figura 3-6 Implementación Controlador Por Tablas De Estado

Con respecto a la sintonización es mucho más flexible que el diseño presentado anteriormente debido a que no existen tantas restricciones que cumplir para que se den las condiciones de operación del controlador, por lo que es posible utilizar métodos empíricos como los de Ziegler-Nichols o computacionales como los de minimización de parámetros como el IEA, para este caso se utiliza la misma técnica que con los controladores anteriores con la salvedad que no se restringe el ancho de bando ni el margen de fase mínima

En la Figura 3-7 se ve el comportamiento a escalón de los controladores, en rojo la señal del controlador GS difuso y en azul el controlador por tablas de estado. Se puede apreciar que ambos tienen un buen comportamiento con respecto al seguimiento de la señal.



Fuente: Elaboración Propia Software Matlab

Figura 3-7 Comparación entre tablas de estado y ajuste difuso

Si se observan los parámetros principales de la respuesta escalón estos muestran un comportamiento similar en los primeros 3 segmentos de simulación aumentando el tiempo de subida en los últimos dos rangos simulados, pero disminuyendo el overshoot para el caso del controlador por tabla de estados.

Tabla 3-3 Análisis de Respuesta al Escalón

Simulación	Rise Time		Settling Time		Overshoot	
	PID	Gain Shoulder	PID	Gain Shoulder	PID	Gain Shoulder
1	147.24	145,46	185,74	221,45	1.08	0,00
2	107.28	102,33	147,02	140,32	1,06	1,04
3	278.08	199,42	364,74	261,86	1,44	0,45
4	506,39	398,32	672,21	499.22	0,45	0,00
5	506.38	602,93	950,35	766.81	0,16	0,00

Fuente: Elaboración Propia

Con respecto a la variabilidad de los parámetros para los distintos puntos de operación estos son presentado en la Tabla 3-4 donde se observa que los resultados son muy similares a la a lo ocurrido con el PID Estándar, el tiempo de subida es mayor en el controlador por tabla de estados pero la estabilidad y sobre impulsos son menores en el controlador propuesto.

Tabla 3-4 Estadística Básica de Los Controladores

Datos Estadísticos	Rise Time		Settling Time		OverShoot	
	GS Table	GS Fuzzy	GS Table	GS Fuzzy	PID	GS Fuzzy
Media	309,07	289,69	464,01	377,93	0,84	0,30
Desviación estándar	190,88	208,55	341,92	255,13	0,52	0,46
Rango	399,11	500,60	803,33	626,49	1,28	1,04
Mínimo	107,28	102,33	147,02	140,32	0,16	0,00
Máximo	506,39	602,93	950,35	766,81	1,44	1,04

Fuente: Elaboración Propia

3.6 CONCLUSIONES A LAS PRUEBAS CON CONTROLADORES DE TABLA DE ESTADO

De este conjunto de pruebas para medir el desempeño de los controladores se puede ver que el controlador diseñado cumple con mayor eficiencia los requerimientos de diseño, aunque el controlador por tabla de estado también tiene una buena performance. La diferencia radica en la forma que tiene el controlador de tabla de estado para obtener un valor de salida, en donde se interpreta la magnitud de la señal controlada y se interpola entre dos puntos de operación lo que castiga enormemente las zonas de no linealidad, esto se puede ver en la pérdida de eficiencia entre las simulaciones 3 a 5 en donde para obtener una mejor performance deberían

agregarse más valores a la tabla de estado. Por otra parte, el controlador propuesto sopesa este problema agregando tanto el vector magnitud como el vector dirección de la señal controlada lo que da mayor fidelidad a la salida del controlador.

3.7 TRABAJOS FUTUROS - IMPLEMENTACION EN ENTORNOS INDUSTRIALES

Para la implementación del modelo en un entorno industrial es necesario llevar el diseño propuesto a uno de los cinco lenguajes de programación de los PLC definidos en el estándar IEC 61131, dentro de los lenguajes el más propicio para realizar la implementación es el Texto Estructurado (ST).

Matlab cuenta con herramientas de generación de código para PLC de las marcas más utilizadas en el mercado, por lo que sería necesario realizar la simplificación del modelo de control propuesto para mejorar las matemáticas utilizadas y así disminuir el tiempo de ejecución lo cual es importante debido a que estos controladores funcionan de forma cíclica.

Respecto a la simplificación del modelo difuso, por la forma en la que es presentado es posible generar una tabla de estados que simplificaría notablemente el tiempo de ejecución del programa al eliminar el cálculo de centroide, mientras que la simplificación del controlador PID podría únicamente limitarse a la utilización de los PID que ya traen en sus librerías la mayoría de los PLC.

CONCLUSIONES

En los resultados anteriormente mostrados se puede concluir que el controlador propuesto es superior en rendimiento a los controladores PID estándar y a la solución estándar para este problema de control como lo son los controladores por tablas de estado. Es posible listar los siguientes pros y contras del diseño propuesto.

- I. La implementación del diseño del controlador de ganancia programada requiere un estudio para la variable controlada que, aunque sencillo es inexistente en el resto de los controles.
- II. La sintonización del controlador de ganancia programada es bastante complicada y es necesario un análisis mucho mayor al del controlador PID y PID por tablas de estado, siendo estos últimos dos controladores fácilmente sintonizables por métodos de sintonización altamente conocidos y estudiados.
- III. Para la sintonización del controlador propuesto es necesario realizar un modelo del sistema a controlar bastante preciso lo que a veces no es posible en plantas reales. Este problema no se presenta con los controladores PID Estándar y por tabla de estados.
- IV. Para poder aplicar el controlador propuesto el proceso a controlar debe ser estable para todos los puntos de operación que se requieran. En caso de no ser así invalidaría las reglas y la forma de sintonía lo que haría aún más difícil el proceso de ajuste. Este problema no se tiene con el método estándar de tabla de estados debido a que se podrían generar tantos puntos como se deseen para la interpolación

Como conclusiones finales el Controlador por ganancia controlada propuesto sería aplicable en casos puntuales y su performance más elevada no compensaría la cantidad de análisis que es necesario realizar para su implementación en comparación con otras técnicas mucho más sencillas de implementar, esto desde el punto de vista de un único proceso a controlar.

Desde el punto de vista de la implementación dentro de un sistema de control distribuido en donde existen muchos más procesos que controlar y la implementación del diseño propuesto podría darse como una macro (planilla con lógica pre-configurada) podría llegar a ser útil, aunque con la cantidad de análisis necesario para realizar la sintonización y las consideraciones que se deben tener en cuenta podrían llegar a ser mejor algún controlador personalizado.

BIBLIOGRAFÍA

- [1] M. A. P. Cando, «CONTROL DE UN MODELO AERODINÁMICO APLICANDO SISTEMAS DIFUSOS.», p. 148, 2008.
- [2] Katsuhiko Ogata, *Ingeniería de control moderna*, 5.^a ed. Prentice Hall, 2010.
- [3] W. Siler y J. J. Buckley, *Fuzzy expert systems and fuzzy reasoning*. Hoboken, N.J: Wiley, 2005.
- [4] J. D. C. Quisaguano, «Diseño y Simulación de Esquemas de control difuso optimizados mediante algoritmos evolutivos aplicados a sistemas lineales», p. 118, 2018.
- [5] J. Jantzen, «Tuning Of Fuzzy PID Controllers», p. 22.
- [6] A. Roca, *Control automático de procesos industriales: con prácticas de simulación y análisis por ordenador PC*. 2014.
- [7] A. Matute y W. B. Suárez, «Técnicas de Lógica Difusa en Ingeniería de Control», p. 10, 2017.
- [8] M. Orellana y M. Enrique, «Diseño e implementación de algoritmos de control difuso y Pid adaptativo STR para el sistema mimo de doble rotor 33-220 de feedback para el laboratorio de control automático de la universidad politécnica salesiana sede Guayaquil», p. 95, 2018.
- [9] K. Ogata, *Dinámica de Sistemas 1ra Edición*. Prentice-Hall, 1987.
- [10] «PID Tuning Algorithm - MATLAB & Simulink - MathWorks América Latina». [En línea]. Disponible en: <https://la.mathworks.com/help/control/getstart/pid-tuning-algorithm.html>. [Accedido: 28-dic-2019].

ANEXOS

ANEXO A: SCRIPT PARA CALULOS DE PUNTOS DE OPERACIÓN

```

% Limpieza del area de trabajo
clc
clear all
close all

% Definicion estandar ploteo
set(0,'DefaultTextFontName','Times',...
'DefaultTextFontSize',12,...
'DefaultAxesFontName','Times',...
'DefaultAxesFontSize',12,...
'DefaultLineLineWidth',1,...
'DefaultLineMarkerSize',7.75)

%Parametros modelo tanques

k1=0.15;
k2=0.04;
k3=0.07;
a2=0.45;
C1=1;
C2=1.5;

% Definicion del modelo en simulink para calculos siguientes
mdl = 'ModelTank';
open_system(mdl)

% Definicion de puntos de operacion

H2 = [0.91 3.63 8.16 14.51 22.68];
operPoint = operSpec(mdl,numel(H2));

for i = 1:numel(H2)
    operPoint(i).Outputs.Known = true;
    operPoint(i).Outputs.y = H2(i);
end

% Busqueda del punto de equilibrio para los puntos de operacion

% eqPoint = findop(mdl,operPoint,findopOptions('DisplayReport','off'));
eqPoint = findop(mdl,operPoint);

% Linealizacion y Estabilidad del modelo
G = linearize(mdl,eqPoint);
estable = isstable(G,'elem')

figure
hold on
for i = 1:numel(H2)
    sys = G(:, :, i);
    sys.Name = ['H2=', num2str(H2(i))];
    sys.InputName = 'Reference';
    bode(sys);
end
legend('show','location','southeast')
hold off

% Busqueda de controladores para puntos de operacion
[C,info] = pidtune(G,'pidf',0.02555);

```

```

clsys = feedback(G*C,1);
figure
hold on
for i = 1:length(H2)
    % Select a system from the LTI array
    sys = clsys(:, :, i);
    set(sys, 'Name', ['H2=', num2str(H2(i))], 'InputName', 'Reference');
    % Plot step response
    stepplot(sys, 3000);
end
legend('show', 'location', 'southeast')
hold off

```

ANEXO B: SCRIPT PARA PRUEBAS DE RENDIMIENTO

```

% Inicializacion de puntos de operacion
H1init = [0.56 2.25 5.06 9.00 14.06];
H2init = [0.91 3.63 8.16 14.51 22.68];
alinit = 0.2:0.2:1;

% Generacion de valores de simulacion
for i=1: numel(H2init)
    H1s = H1init(i);
    H2s = H2init(i);
    a1 = alinit(i);
    Sp = H2s*0.2 + H2s

    % Simulacion Simulink
    sim('PruebasRendimiento', 1000);

    % Generacion de graficas
    subplot(5,1,i)
    hold on
    plot(h2PID.time, h2PID.signals.values)
    hold on
    plot(h2Fuzzy.time, h2Fuzzy.signals.values)
    hold on
    plot(SpPlot.time, SpPlot.signals.values)
end

```

ANEXO C: REGLAS CONTROLADOR DIFUSO

1. If (pv is NB) and (pvc is NB) then (Kp is PB)(Ki is NB)(Kd is N)
2. If (pv is NB) and (pvc is N) then (Kp is PM)(Ki is NM)(Kd is N)
3. If (pv is NB) and (pvc is Z) then (Kp is PM)(Ki is NM)(Kd is Z)
4. If (pv is NB) and (pvc is P) then (Kp is P)(Ki is N)(Kd is NM)
5. If (pv is NB) and (pvc is PB) then (Kp is Z)(Ki is Z)(Kd is NB)
6. If (pv is N) and (pvc is NB) then (Kp is PM)(Ki is NM)(Kd is P)
7. If (pv is N) and (pvc is N) then (Kp is PM)(Ki is NM)(Kd is P)
8. If (pv is N) and (pvc is Z) then (Kp is P)(Ki is N)(Kd is Z)
9. If (pv is N) and (pvc is P) then (Kp is Z)(Ki is Z)(Kd is N)
10. If (pv is N) and (pvc is PB) then (Kp is N)(Ki is P)(Kd is NM)

11. If (pv is Z) and (pvc is NB) then (Kp is PM)(Ki is NM)(Kd is Z)
12. If (pv is Z) and (pvc is N) then (Kp is P)(Ki is N)(Kd is Z)
13. If (pv is Z) and (pvc is Z) then (Kp is Z)(Ki is Z)(Kd is Z)
14. If (pv is Z) and (pvc is P) then (Kp is N)(Ki is P)(Kd is Z)
15. If (pv is Z) and (pvc is PB) then (Kp is NM)(Ki is PM)(Kd is Z)
16. If (pv is P) and (pvc is NB) then (Kp is P)(Ki is N)(Kd is N)
17. If (pv is P) and (pvc is N) then (Kp is Z)(Ki is Z)(Kd is N)
18. If (pv is P) and (pvc is Z) then (Kp is N)(Ki is P)(Kd is Z)
19. If (pv is P) and (pvc is P) then (Kp is NM)(Ki is PM)(Kd is P)
20. If (pv is P) and (pvc is PB) then (Kp is NM)(Ki is PM)(Kd is PM)
21. If (pv is PB) and (pvc is NB) then (Kp is Z)(Ki is Z)(Kd is P)
22. If (pv is PB) and (pvc is N) then (Kp is N)(Ki is P)(Kd is P)
23. If (pv is PB) and (pvc is Z) then (Kp is NM)(Ki is PM)(Kd is Z)
24. If (pv is PB) and (pvc is P) then (Kp is NM)(Ki is PM)(Kd is PM)
25. If (pv is PB) and (pvc is PB) then (Kp is NB)(Ki is PB)(Kd is PB)